



---

---

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**

**DESARROLLO DE UN SISTEMA PARA EL CONTROL DE ÓRDENES DE COMPRA E INVENTARIOS PARA UNA CADENA DE RESTAURANTES DE COMIDA RÁPIDA.**

**TESIS**

**QUE PARA OBTENER EL TÍTULO DE**

**INGENIERO EN COMPUTACIÓN**

**P R E S E N T A N :**

**CASTRO PEDRERA SANDRA  
FLORES RANGEL CARLOS HERIBERTO  
PEREA SANDOVAL CESAR AUGUSTO  
SANDOVAL REYES BEATRIZ  
VÁZQUEZ CAMACHO MIRIAM**

**DIRECTOR DE TESIS: M. EN I. NORMA ELVA CHÁVEZ RODRÍGUEZ**



**CIUDAD UNIVERSITARIA**

**ABRIL 2007**



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## Agradecimientos

A Dios, porque a pesar de todas las adversidades siempre me dio la fuerza necesaria y el ánimo para seguir adelante.

A Esteban, gracias mi amor por darme los ánimos, por aguantar mis arranques y sobre todo por creer en mí. Eres el hombre más maravilloso que existe y espero compartir el resto de mi vida contigo y que crezcamos juntos como hasta ahora.

A mi mamachita y Mario por recibirme en su casa en el momento en que más lo necesitaba.

A mi suegra y a Titi porque gracias a su apoyo, su ejemplo de fortaleza y gran amor me hicieron sentir segura y me proporcionaron los medios para continuar.

A Tere y Yola por mostrarme que siempre hay cosas buenas que ver en las demás personas y sobre todo por ser unos mujerones.

A mi amadísima Universidad con la cual se ha forjado mucho de mí como persona y como profesional.

A mis compañeros de tesis por su dedicación y tiempo que dedicaron para lograr por fin poner “la cereza del pastel”.

A todos mis maestros, amigos, compañeros y personas que me rodearon y que fueron capaces de enseñarme mucho más de lo que decían los libros.

**Sandra Castro Pedrera**

Este trabajo esta dedicado para todas aquellas personas que creyeron en mí, a todas aquellas que estuvieron conmigo y que aun están pues gracias a sus alegrías, sus tristezas y aquellas emociones que compartimos es que hoy se cumple un ciclo en esta amada universidad.

Gracias a mi abuelito que siempre estuvo apoyándome y al pendiente de mi, fue un gran maestro, él siempre quiso ver este trabajo y esfuerzo completado, se que donde esté debe sentirse muy contento y orgulloso por este triunfo en el cual esta incluido.

A mis padres agradezco por todo el apoyo, por todo el cariño y por estar conmigo tanto en las buenas y en las malas. Gracias a su entusiasmo dedicación y comprensión he podido cerrar un ciclo mas, lo cual les agradezco de todo corazón pues cuando se tienen unas guías como lo fueron ellos es muy fácil poder convertirse en un gran persona y a sus grandes consejos que siempre fueron para poder mejorar.

A mis hermanos por su cariño, apoyo y por su preocupación durante toda mi vida y especialmente por las nuevas alegrías que me dieron con mis sobrinos. Pues ellos fueron una fuente de inspiración para seguir adelante.

A Celeste con su cariño, amor, apoyo y comprensión hizo que el tiempo durante el desarrollo de este proyecto fuera mucho más corto y fácil, gracias a su interés es que se termina un ciclo más dentro de los muchos que existen.

Al Ingeniero Luis César Vázquez por todos los consejos, apoyo y confianza que me brindo durante mi formación académica, gracias.

Gracias a personas como Gonzalo, Iliana, y muchos otras amistades que he conocido y que han estado presentes en mi vida tanto académica como en la vida diaria y que me han dejado enseñanzas.

Gracias a mis compañeros de tesis por permitirme participar y juntos poder salir adelante en este proyecto.

CARLOS HERIBERTO FLORES RANGEL

A mis amigos por darme la oportunidad de estar ahí con ellos  
A mis enemigos por estar mí contra y oponerse a mi decisión  
A mis padres Mario y Socorro que son el ejemplo de dedicación  
A mis hermanos por ser como son  
Al amor de mi vida que en ella encontré pasión por seguir  
Por la risa de una bebé en la que con sus manos encontré la fé de la vida  
A todos ellos...

Gracias...

César

A mi Madre, por que siempre ha estado a mi lado en todos los momentos de mi vida, por su amor incondicional y por enseñarme el valor de la vida.

A Daniel, por su gran amor, por su paciencia, por su apoyo y por elegir su camino junto al mío.

A mi Bebé, porque desde que me enteré que venia en camino, supe que debía terminar lo que algún día empecé, fue su presencia la que me impulsó a no quedarme a la "mitad", por llegar siempre hasta el final.

A mis Hermanos por su apoyo, por estar en el momento justo cuando los he necesitado.

A mi Padre por albergar en su enorme corazón un lugar dedicado especialmente para mí.

A Miriam por su amistad sincera y desinteresada, por compartir conmigo tantos momentos alegres y porque no, también los tristes en toda esta aventura que fue nuestra carrera universitaria.

A mis compañeros de esta Tesis, sin ellos simplemente no hubiera sido posible, gracias por su dedicación y compromiso.

Al PAT y a nuestra asesora de Tesis Norma Elva, que gracias a ellos pude convertir en realidad el sueño de tener mi Título Universitario.

y desde luego agradezco a mi UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO por estar hecha en CU.

"Por mi raza hablará el espíritu"

Goooy!!!!!!!!!!!!!!!!!!!!!!!!!!!!

*Beatriz Sandoval Reyes*

## **Agradecimientos**

Dedico esta tesis a mi madre por el enorme apoyo que me brindo a lo largo de mi carrera, por sus horas de desvelo acompañándome mientras estudiaba para algún examen, por su consejo insistente para que me titulara y por ser la mujer más valiente y emprendedora que conozco, te amo mamá.

A mis hijas Kelly y Sarahi por sus abrazos y besos incondicionales, porque sin saberlo han sido fuente de energía e inspiración para vivir y ser mejor cada día. Siempre las llevo en mi corazón.

De igual forma quiero agradecerle a mi padre y a Aída por su apoyo y consejos, no solo en mi trayectoria escolar, sino en toda mi vida.

A mi tía Pi por toda la sabiduría que carga en esa cabecita y que comparte día a día conmigo y con mis hijas, gracias mil por tu valioso tiempo y por toda la ayuda brindada.

A César y Vero por todo el cariño y comprensión que siempre me han brindado.

A mis queridas tías Lala, Mare, Eva, Eme y Chabe porque en mi verán reflejados parte de los valores y preceptos con los que me educaron. Gracias por la confianza.

A mi querida Bety por todos los momentos compartidos en la facultad, por ser una gran persona, mi mejor amiga y por que cumplimos el sueño de titularnos juntas.

A la profesora Norma Elva y a mis compañeros de Tesis por su esfuerzo y empeño en la realización de ésta.

A mis jefes Vicente Contreras y Boris Franco por apoyarme para que realizara el seminario de titulación.

Y en especial quiero agradecerle a Dios la oportunidad de aprender y crecer dentro de la máxima casa de estudios, UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO, y por siempre ser de piel dorada y sangre azul.

*Sinceramente Miriam Vázquez Camacho*

## Capítulo I. Introducción.

1.1	Resumen	6
1.2	Introducción General.	8
1.3	Descripción del problema.	10
1.4	Objetivo General.	11
1.4.1	Objetivos Específicos.	11
1.5	Justificación.	12
1.6	Alcances y limitaciones.	14

## Capítulo II. Marco teórico.

2.1	Bases de Datos.	17
2.1.1	Introducción.	17
2.1.2	Modelo de Datos.	19
2.1.3	Modelos lógicos.	20
2.1.4	Modelo entidad-relación (E-R).	21
2.1.5	Modelo de red.	22
2.1.6	Modelo jerárquico.	23
2.1.7	Modelo orientado a objetos.	23
2.1.8	Comparación entre modelos.	23
2.1.9	Modelos físicos de datos.	24
2.1.10	Instancias y esquemas.	24
2.1.11	Independencia de datos.	24
2.1.12	Lenguaje de definición de datos (DDL).	25
2.1.13	Lenguaje de manipulación de datos (DML).	25
2.1.14	Sistema gestor de base de datos (DBMS).	26
2.1.15	Administrador de base de datos (DBA).	27
2.1.16	Usuarios de bases de datos.	28
2.1.17	Estructura general del sistema de bases de datos.	29
2.1.18	Normalización.	31
2.1.19	Acid.	33
2.2	Redes.	35
2.2.1	Definición.	35
2.2.2	Clasificación de redes.	36
2.2.3	Topologías.	37
2.2.3.1	Topología de Red.	37
2.2.3.2	Topología de Bus.	37
2.2.3.3	Topología de Estrella.	38
2.2.3.4	Topología de Anillo.	39
2.2.3.5	Topología de Árbol.	39
2.2.3.6	Topología de Malla.	40
2.2.4	Modelos de referencia.	41
2.2.4.1	Modelo OSI.	41
2.2.4.1.1	Capa física (Capa 1).	41
2.2.4.1.2	Capa de Enlace (Capa 2).	42
2.2.4.1.3	Capa de Red (Capa 3).	42



2.2.4.1.4	Capa de Transporte (Capa 4).	42
2.2.4.1.5	Capa de Sección (Capa 5).	43
2.2.4.1.6	Capa de Presentación (Capa 6).	44
2.2.4.1.7	Capa de Aplicación (Capa 7).	44
2.2.4.2	Cliente/Servidor.	44
2.2.4.2.1	Definiciones.	44
2.2.4.2.2	Modelo Cliente/Servidor.	45
2.2.4.2.3	Hardware Cliente/Servidor.	47
2.2.4.2.4	Servidor.	47
2.2.4.2.5	Cliente.	48
2.2.4.2.6	Cliente/Servidor en Web.	48
2.2.4.2.7	El cliente Web.	49
2.2.5	Modelo TCP/IP (Transport Control Protocol / Internet Protocol).	50
2.2.5.1	Capa de Aplicación.	51
2.2.5.2	Capa de Transporte.	51
2.2.5.3	Capa de Red.	51
2.2.5.4	Capa Física.	52
2.2.6	Seguridad.	53
2.2.6.1	Introducción.	53
2.2.6.2	La seguridad informática en la actualidad.	53
2.2.6.3	Situaciones que involucran un estado de emergencia dentro de la seguridad informática.	54
2.2.6.3.1	Amenazas y ataques.	56
2.2.6.3.2	Riesgos.	57
2.2.6.3.3	Condiciones de Salvaguarda de la información ( políticas de seguridad ).	59
2.2.6.4	Criptología informática (documentos en resguardo).	63
2.2.6.4.1	Definición.	63
2.2.6.4.2	Historia.	64
2.2.6.5	Encriptación de documento y su transmisión.	68
2.2.6.6	Diseño de contraseñas seguras.	68
2.3	Ingeniería del software.	70
2.3.1	Ciclo de vida del software.	70
2.3.1.1	Definición de un Modelo de ciclo de vida.	70
2.3.2	Alternativas de Modelos de Ciclo de Vida de software.	71
2.3.2.1	Modelo Cascada.	71
2.3.2.2	Modelo de desarrollo incremental.	71
2.3.2.3	Modelo de desarrollo evolutivo.	72
2.3.2.4	Modelo de prototipo de requerimientos.	73
2.3.2.5	Modelo Espiral.	74
2.3.2.6	Modelo Concurrente.	74
2.3.2.7	Modelo Rup.	75
2.3.2.8	Modelo Xp.	77

### Capítulo III. Análisis.

3.1	Situación actual.	81
3.2	Requerimientos del usuario.	86

3.2.1	Módulo para la carga de información en la Base de Datos.	87
3.2.2	Módulo para la toma de pedido.	88
3.2.3	Módulo de consultas de remisiones pendientes.	89
3.2.4	Módulo para el registro de las recepciones.	89
3.2.5	Módulo para el registro de transferencias.	91
3.2.6	Módulo para la captura del inventario semanal.	93
3.3	Presupuestos.	95
3.3.1	Hardware.	95
3.3.2	Software.	95
3.4	Análisis de los riesgos de seguridad.	96
3.4.1	Riesgo de Paso.	96
3.4.1.1	Acceso libre de usuario.	96
3.4.1.2	Omisión de resguardo de usuario y contraseña.	97
3.4.1.3	Omisión de encriptación de paso ( contraseñas simples.	97
3.4.2	Riesgos de Uso.	97
3.4.2.1	Solicitudes sin autorización.	97
3.4.2.2	Observadores sin autorización.	98
3.4.2.3	Usuarios sin autorización.	98
3.4.3	Violación de la integridad.	98
3.4.3.1	Ataques a las instalaciones lógicas.	98
3.4.4	Políticas de seguridad Básicas.	99
3.4.4.1	Políticas de uso.	100
3.4.4.2	Políticas de acceso.	100
3.4.4.3	Políticas de respaldo.	101

## Capítulo IV. Diseño.

4.1	Herramientas de desarrollo.	103
4.1.1	Linux.	103
4.1.1.1	Introducción.	103
4.1.1.2	Breve Historia.	104
4.1.1.3	Características principales.	104
4.1.1.4	Características adicionales.	105
4.1.2	Sistema manejador de bases de datos Posgresql.	107
4.1.2.1	Introducción e Historia.	107
4.1.2.2	Características Principales.	107
4.1.3	Conceptos Básicos de Programación Orientada a Objetos.	110
4.1.3.1	Introducción.	110
4.1.3.2	Terminología.	110
4.1.4	Java.	112
4.1.4.1	Breve Historia.	112
4.1.4.2	Características Principales.	115
4.1.4.3	Entorno de Ejecución.	117
4.1.4.4	Tipo de aplicaciones.	117
4.1.5	Apache Tomcat.	118
4.1.5.1	Introducción.	118
4.1.5.2	Principales Características.	119

4.1.6	Requerimientos para el desarrollo de la aplicación.	119
4.1.6.1	Requerimientos mínimos de Hardware.	119
4.1.6.2	Requerimientos mínimos de Software.	119
4.2	Casos de uso.	121
4.2.1.	Detección de Actores.	122
4.2.2	Detección de casos de uso – CU	122
4.2.3	Diagramas generales de Casos de Uso	123
4.2.4	Documentación de Casos de Uso.	124
4.3	Diseño de la Base de Datos.	
4.3.1	Puntos considerados.	134
4.3.2	Diseño de la BD empleando el modelo entidad-relación.	135
4.3.2.1	Identificar las entidades.	135
4.3.2.2	Identificar los atributos de cada entidad.	137
4.3.2.3	Identificar las relaciones	137
4.3.2.4	Creación del diagrama entidad-relación	137

## Capítulo V. Implementación.

5.1	Instalación de la aplicación.	139
5.2	Caso práctico.	143

## Capítulo VI. Pruebas.

6.1	Ingreso al sistema.	164
6.2	Elaboración de pedido.	164
6.3	Consulta de remisión.	168
6.4	Registro de recepción.	169
6.5	Registro de transferencia de entrada.	172
6.6	Registro de transferencia de salida.	174
6.7	Registro de inventario semanal.	175

## Capítulo VII. Conclusiones y Recomendaciones

7.1	Conclusiones	178
7.1	Trabajo a futuro	180

## Apéndices

A.	Diagrama entidad-relación de la base de datos.	182
	Diccionario de datos.	183
B.	Manual de usuario.	238
C.	Estándares de programación.	276

<b>Bibliografía</b>	284
---------------------	-----

# Capítulo I. Introducción.

- 1.1 Resumen.
- 1.2 Introducción General.
- 1.3 Descripción del problema.
- 1.4 Objetivo General.
  - 1.4.1 Objetivos Específicos.
- 1.5 Justificación.
- 1.6 Alcances y limitaciones.



## **1.1 Resumen.**

### Capítulo I. Introducción.

El primer capítulo introduce al lector en la importancia de la implementación de un sistema informático para el control de Órdenes de Compra e Inventarios para una cadena de restaurantes de comida rápida. Esto seguido del planteamiento del problema, objetivo, justificación, alcances y limitaciones.

### Capítulo II. Marco Teórico.

En este capítulo se abordarán temas básicos de conceptos computacionales que servirán para desarrollar el proyecto de tesis.

### Capítulo III. Análisis.

Este capítulo se analiza el problema actual, para determinar las causas que lo generan, así como para definir los medios posibles para solucionarlo.

### Capítulo IV. Diseño

Dentro de este capítulo, se determina la organización de la base de datos, incluidos su estructura y contenido. Dentro de la estructura se describe la definición de las tablas.

También se mencionan las herramientas empleadas en el diseño de base de datos, tales como el modelo de casos de uso, que se utilizó para identificar los flujos funcionales que debe realizar el sistema, así como roles de usuario y verificar que los requerimientos funcionales sean tomados en cuenta.

## Capítulo V. Implementación

Bajo el entorno de java, se ha desarrollado un Sistema de Control de Inventarios y Órdenes de Compra, el cual consta de tres módulos principales, los cuales son:

- Control de Órdenes de Compra.
  - Pedidos
  - Remisiones
  - Recepciones
  - Consultas
- Control de Inventario
  - Transferencias de entrada/salida
  - Inventario Semanal
- Reportes
  - Inventario Semanal
  - Transferencias de entrada
  - Transferencias de salida

## Capítulo VI. Pruebas

En el marco de este capítulo, es donde se llevan a cabo las pruebas y funcionamiento general del sistema.

## Capítulo VII. Conclusiones

Aquí se encuentran las conclusiones de cada uno de los participantes de este proyecto de Tesis.

## **1.2 Introducción General.**

El desarrollo de empresas que pretendan ser realmente competitivas en el plano mundial por su nivel de calidad y atención al cliente, requiere de una plataforma que se encuentre en la posición de aplicar nuevos métodos y tecnologías enfocadas a encaminar dicho desarrollo.

Los cambios en la política comercial y económica centrados en el principio de la globalización obligan, no sólo como sociedad sino primordialmente como empresas, a ser más competentes en todos los ámbitos. Se debe contar con una mayor cantidad y calidad de información, para poder tomar decisiones efectivas, usar tecnologías más avanzadas tanto de proceso como de información y comunicación, así como muchas otras habilidades específicas que nunca se terminarían de enumerar pues la lista crece día a día. Todo esto implica una necesidad de cambiar constantemente, a desarrollar otra cultura social y de trabajo, pues cada vez con mayor frecuencia se tendrán que aprender nuevos conceptos y desarrollar nuevas habilidades.

En el caso particular de una cadena de restaurantes de comida rápida, basta con echar un vistazo a los principios con los que se rigen:

### **El Cliente es lo más importante**

Los equipos de Restaurantes y los Centros de Apoyo a Restaurantes escuchamos y respondemos a la voz de los clientes. Siempre con una actitud positiva de ¡Sí se puede!

### **Confianza en las personas**

Confiamos en las buenas intenciones de las personas, creemos en ellas, tomamos en cuenta las ideas de todos y desarrollamos equipos diversos en estilos y experiencias.

### **Apoyo y Capacitación**

Nos apoyamos y capacitamos unos a otros, primero tomando lo mejor de los demás y posteriormente en cómo podemos ser más efectivos.

### **Congruencia**

Hacemos lo que decimos.

### **Superación**

Rebasar los resultados del año anterior mejorando continuamente.

### **Optimismo**

Trabajamos con optimismo e intensidad.

### **Trabajo en equipo**

Tenemos la seguridad de que "juntos podemos".

Como se puede observar, todos estos principios están enfocados a que la empresa sea mejor cada día, brindando al cliente y a su familia la mejor opción para deleitar el paladar y servirle en el menor tiempo posible, con calidad y cortesía, de aquí la importancia de que esté obligada a ser más competitiva.



Por todo ello este proyecto de Tesis propone un análisis del actual manejo de materias primas e insumos con los que cuentan este tipo de restaurantes y propone mejoras, al implementar un sistema informático integral, que ayude a sus empleados a realizar sus tareas de manera mas eficiente, así como disminuir las pérdidas por mermas, desperdicios y substracción ilegal de los insumos, teniendo un fuerte impacto sobre las utilidades de la empresa.

### **1.3 Descripción del problema.**

Actualmente en los restaurantes existen problemas que dificultan y encarecen los procesos internos relacionados con los pedidos de materia prima, la falta de ésta, es ocasionada por dos factores principales: pronósticos inexactos de uso de materia prima y sistemas de inventarios poco eficientes.

La forma en que la empresa realiza los pronósticos actualmente es por medio de un sistema ya existente que genera un archivo con la relación de productos y las cantidades de los mismos como orden de compra, sin embargo dicha información implica un método muy complejo para realizar un pedido, dicho método se explicará más a detalle en capítulos subsecuentes.

Para la elaboración de este proyecto de Tesis se estudiaron y analizaron los problemas a los que se enfrentan los restaurantes de comida rápida, con ayuda del personal. Posteriormente se recopiló, organizó y analizó la información y se identificaron las herramientas que permitieron solucionar el problema. La evaluación de los resultados obtenidos fue de gran importancia para conocer su impacto dentro del restaurante y para saber si realmente representaban una alternativa atractiva de mejora.

Ante esta problemática, y en la búsqueda de conseguir una mayor participación en el mercado surge la necesidad de contar con un sistema informático a la

medida, el cual agilizará sus procesos, tanto administrativos como operativos; con el objetivo primordial de ahorrar tiempo, disminuir mermas y costos, permitiendo un flujo eficiente de información, entre otras ventajas, que finalmente repercuten en la satisfacción del cliente y le brindan una ventaja competitiva a la empresa.

#### **1.4 Objetivo General.**

Desarrollar un sistema que permita hacer un manejo eficiente del proceso de compra de materia prima y utensilios, recepción e inventario de los mismos, dentro de una cadena de restaurantes de comida rápida con el fin de disminuir las pérdidas provocadas por su inexistencia.

##### **1.4.1 Objetivos Específicos.**

Los objetivos específicos de este proyecto de tesis son:

- Mejorar el proceso de compra de materia prima y utensilios a través de un sistema gráfico amigable que genere ahorros.
- Control y manejo de tiempos límite para realizar el pedido.
- Automatizar y optimizar la generación del pedido.
- Ayudar a prevenir la falta de insumos y tomar las medidas pertinentes para no afectar la calidad del servicio.
- Verificar si el pedido solicitado coincide con el producto recibido.
- Controlar adecuadamente lo que el restaurante recibe de cada proveedor y compararlo contra lo facturado por dicho proveedor.

Dichos objetivos surgen ante la enorme importancia que representa para el restaurante, ofrecer un alto nivel de servicio a sus clientes de manera que le permita seguir contando con su preferencia sobre las demás empresas líderes en su ramo.

## 1.5 Justificación.

En vista de que existe un mercado restaurantero extremadamente capaz, surge la necesidad de implementar sistemas informáticos eficientes para obtener una ventaja competitiva, lo que significa satisfacer las necesidades del cliente. Todo esto buscando beneficios: como el mejoramiento en la eficiencia de los procesos tanto administrativos como operativos; con el objetivo primordial de ahorrar tiempo, disminuir costos y fortalecer significativamente la venta de productos en condiciones óptimas.

También en la operación propia del restaurante, los beneficios son:

- Planeación de compras. Al controlar el inventario se crea información precisa, que será útil para aprovisionarse de producto sin excesos y sin faltantes, ya que podrá conocerse a ciencia cierta las fluctuaciones de las existencias dependiendo de la época del año en la que se encuentren.
- Planeación del flujo de efectivo. Se conocerá el monto de la compra ya sea diaria o semanalmente, lo que permitirá saber cuánto capital se habrá de invertir. Así se eliminarán o postergarán gastos no propios a la operación del restaurante, para no minar su liquidez.
- Sencillez en el proceso y consulta de orden-remisión-recepción de productos a proveedores, ya que se hace una orden, de ésta se recibe la remisión, y solo se verifica en la recepción diferencias contra lo remisionado.
- Pagar al proveedor por los productos que realmente se reciben en cada restaurante.
- Posibilidad de obtener estadísticas de la calidad de servicio de los proveedores, para controlar el proceso.
- Minimizar errores de captura derivados conversiones erróneas entre las unidades de presentación y las unidades de inventario.
- Evitar recaptura en la recepción, ya que solo se realizan correcciones a la información con respecto a la orden de compra.

- Se considera el pedido sugerido de SAI<sup>1</sup> (Sistema Administrador de Información) y los pedidos en tránsito para sugerir la cantidad de producto que se debe ordenar, evitando que el gerente haga cálculos manuales.
- Conocer de forma sencilla y oportuna las diferencias entre lo ordenado y lo remisionado, es decir si el proveedor podrá surtir la orden completa. Esto le brinda al restaurante la oportunidad de solicitar apoyo a otro restaurante para abastecerse con tiempo de insumos que pudieran ser críticos para la producción y que el proveedor no tuviera en existencia.
- Declarar las diferencias, en caso de haberlas, entre lo remisionado por el proveedor y lo recibido por el restaurante. Esto le permite llevar un mejor control en las solicitudes de notas de crédito.
- A nivel restaurante, a los gerentes se les proporcionará una herramienta completa que les permita realizar de una forma más simple y eficaz la captura de transferencias de entrada y de salida, y del inventario semanal.
- Aplicación Todo en Uno. Una sola aplicación que permite realizar el proceso completo: Orden-Remisión-Recepción-Inventario con una misma funcionalidad en los distintos módulos como se muestra en la Figura 1.5.

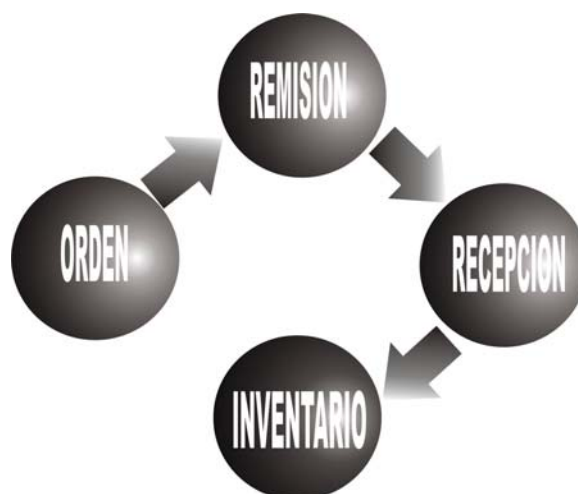


Figura 1.5 Proceso Completo Orden-Remisión-Recepción-Inventario.

<sup>1</sup> SAI Sistema Administrador de Información.

## **1.6 Alcances y limitaciones.**

- El estudio se realizará en restaurantes de comida rápida del D.F.
- Se detectarán los diferentes métodos de alta de pedidos.
- La investigación no realizará un plan de aplicación para cada restaurante en particular.
- No se abordará el desarrollo y funcionamiento del SAI.

Este proyecto de Tesis vislumbra no solo una aplicación de órdenes y recepción de producto, sino un módulo que forma parte de una nueva aplicación de inventario que en un inicio va a interactuar con el de SAI y que en el futuro puede sustituirlo.

# Capítulo II.

## Marco Teórico.

- 2.1 Bases de Datos.
  - 2.1.1 Introducción.
  - 2.1.2 Modelo de Datos.
  - 2.1.3 Modelos lógicos.
  - 2.1.4 Modelo entidad-relación (E-R).
  - 2.1.5 Modelo de red.
  - 2.1.6 Modelo jerárquico.
  - 2.1.7 Modelo orientado a objetos.
  - 2.1.8 Comparación entre modelos.
  - 2.1.9 Modelos físicos de datos.
  - 2.1.10 Instancias y esquemas.
  - 2.1.11 Independencia de datos.
  - 2.1.12 Lenguaje de definición de datos (DDL).
  - 2.1.13 Lenguaje de manipulación de datos (DML).
  - 2.1.14 Sistema gestor de base de datos (DBMS).
  - 2.1.15 Administrador de base de datos (DBM).
  - 2.1.16 Usuarios de bases de datos.
  - 2.1.17 Estructura general del sistema de bases de datos.
  - 2.1.18 Normalización.
  - 2.1.19 Acid.
- 2.2 Redes.
  - 2.2.1 Definición.
  - 2.2.2 Clasificación de redes.
  - 2.2.3 Topologías.
    - 2.2.3.1 Topología de Red.
    - 2.2.3.2 Topología de Bus.
    - 2.2.3.3 Topología de Estrella.
    - 2.2.3.4 Topología de Anillo.
    - 2.2.3.5 Topología de Árbol.
    - 2.2.3.6 Topología de Malla.
  - 2.2.4 Modelos de referencia.
    - 2.2.4.1 Modelo OSI.
      - 2.2.4.1.1 Capa física (Capa 1).
      - 2.2.4.1.2 Capa de Enlace (Capa 2).
      - 2.2.4.1.3 Capa de Red (Capa 3).
      - 2.2.4.1.4 Capa de Transporte (Capa 4).
      - 2.2.4.1.5 Capa de Sección (Capa 5).
      - 2.2.4.1.6 Capa de Presentación (Capa 6).

- 2.2.4.1.7 Capa de Aplicación (Capa 7).
- 2.2.4.2 Cliente/Servidor.
  - 2.2.4.2.1 Definiciones.
  - 2.2.4.2.2 Modelo Cliente/Servidor.
  - 2.2.4.2.3 Hardware Cliente/Servidor.
  - 2.2.4.2.4 Servidor.
  - 2.2.4.2.5 Cliente.
  - 2.2.4.2.6 Cliente/Servidor en Web.
  - 2.2.4.2.7 El cliente Web.
- 2.2.5 **Modelo TCP/IP (Transport Control Protocol / Internet Protocol).**
  - 2.2.5.1 Capa de Aplicación.
  - 2.2.5.2 Capa de Transporte.
  - 2.2.5.3 Capa de Red.
  - 2.2.5.4 Capa Física.
- 2.2.6 Seguridad.
  - 2.2.6.1 Introducción.
  - 2.2.6.2 La seguridad informática en la actualidad.
  - 2.2.6.3 Situaciones que involucran un estado de emergencia dentro de la seguridad informática.
    - 2.2.6.3.1 Amenazas y ataques.
    - 2.2.6.3.2 Riesgos.
    - 2.2.6.3.3 Condiciones de Salvaguarda de la información ( políticas de seguridad ).
  - 2.2.6.4 Criptología informática (documentos en resguardo).
    - 2.2.6.4.1 Definición.
    - 2.2.6.4.2 Historia.
  - 2.2.6.5 Encriptación de documento y su transmisión.
  - 2.2.6.6 Diseño de contraseñas seguras.
- 2.3 Ingeniería del software.
  - 2.3.1 Ciclo de vida del software.
    - 2.3.1.1 Definición de un Modelo de ciclo de vida.
  - 2.3.2 Alternativas de Modelos de Ciclo de Vida de software.
    - 2.3.2.1 Modelo Cascada.
    - 2.3.2.2 Modelo de desarrollo incremental.
    - 2.3.2.3 Modelo de desarrollo evolutivo.
    - 2.3.2.4 Modelo de prototipo de requerimientos.
    - 2.3.2.5 Modelo Espiral.
    - 2.3.2.6 Modelo Concurrente.
    - 2.3.2.7 Modelo Rup.
    - 2.3.2.8 Modelo Xp.

## **2.1 Bases de Datos (BD).**

### **2.1.1 Introducción.**

Una base de datos consiste en una colección de datos interrelacionados que pertenecen a un mismo contexto, se almacenan sistemáticamente para su posterior uso. En la actualidad, y gracias al desarrollo tecnológico de campos como la informática y la electrónica, la mayoría de las bases de datos tienen formato electrónico, que ofrece un amplio rango de soluciones al problema de almacenamiento de datos.

Los sistemas de bases de datos están diseñados para gestionar grandes bloques de información, que implica tanto la definición de estructuras para el almacenamiento como de mecanismos para la gestión de la información. Además deben ofrecer mayor seguridad de la información almacenada pese a la caída del sistema o accesos no autorizados.

Las bases de datos, son creadas con un Sistema Manejador de Bases de Datos (DBMS). Los tres componentes principales de un sistema de base de datos son el hardware, el software DBMS y los datos a manejar, así como el personal encargado del manejo del sistema.

Los objetivos principales de un sistema de base de datos consisten en disminuir los siguientes aspectos:

- *Redundancia e inconsistencia de datos.* Puesto que los archivos que mantienen almacenada la información son creados por diferentes tipos de programas de aplicación existe la posibilidad de que si no se controla detalladamente el almacenamiento, se pueda originar un duplicado de información. Esto aumenta los costos de almacenamiento y acceso a los datos, además de que puede originar la inconsistencia de estos - es decir diversas copias de un mismo dato no concuerdan entre si -, por ejemplo: que se actualiza la dirección de un cliente en un archivo y que en otros archivos permanezca la anterior.



- *Dificultad para tener acceso a los datos.* Un sistema de base de datos debe contemplar un entorno de datos que le facilite al usuario el manejo de los mismos.
- *Anomalías del acceso concurrente.* Para mejorar el funcionamiento global del sistema y obtener un tiempo de respuesta más rápido, muchos sistemas permiten que múltiples usuarios actualicen los datos simultáneamente. En un entorno así la interacción de actualizaciones concurrentes puede dar por resultado datos inconsistentes. Para prevenir esta posibilidad debe mantenerse alguna forma de supervisión en el sistema.
- *Problemas de seguridad.* La información de toda empresa es importante, aunque unos datos lo son más que otros, por tal motivo se debe considerar el control de acceso a los mismos, no todos los usuarios pueden visualizar alguna información, por tal motivo para que un sistema de base de datos sea confiable debe mantener un grado de seguridad que garantice la autenticación y protección de los datos.
- *Problemas de integridad.* Los valores de datos almacenados en la base de datos deben satisfacer cierto tipo de restricciones de consistencia. Estas restricciones se hacen cumplir en el sistema añadiendo códigos apropiados en los diversos programas de aplicación.

Para describir la estructura de una BD es necesario definir el concepto de modelo de datos, y los diversos modelos de datos que se dividen en tres grupos: modelos lógicos basados en objetos, modelos lógicos basados en registros y modelos físicos de datos. Es por ello que para el desarrollo del presente trabajo se planteará primeramente el modelo de datos y después se profundizará en los modelos de bases de datos más utilizados. Finalmente, debido a que en el desarrollo de este proyecto se utilizará el modelo entidad relación, se considera necesario ver el tema de normalización.

### 2.1.2 Modelo de Datos.

Uno de sus objetivos más importantes es proporcionar a los usuarios una visión abstracta de los datos, es decir, el sistema esconde ciertos detalles de como se almacenan y mantienen los datos, sin embargo su estructura y organización deben permitir una extracción eficientemente. Este requerimiento ha llevado al diseño de estructuras de datos complejas para la representación de datos en la BD. Esta complejidad se maneja a través de distintos niveles de abstracción para simplificar la interacción con el sistema.

1. *Nivel físico.* Es el nivel más bajo de abstracción y describe como se almacenan realmente los datos, se describen en detalle las estructuras de datos complejas del nivel bajo.
2. *Nivel conceptual.* Describe qué datos son realmente almacenados en la BD y las relaciones que existen entre ellos. Aquí se describe la BD completa en términos de un número pequeño de estructuras sencillas. El nivel conceptual de abstracción lo usan los administradores de BD, quienes deben decidir que información se va a guardar en la BD.
3. *Nivel de visión.* El nivel más alto de abstracción describe sólo parte de la BD completa. Muchos usuarios no se interesan por toda la información, sólo necesitan una parte de la BD. Para simplificar su interacción con el sistema se define el nivel de abstracción de visión. El sistema puede proporcionar muchas visiones para la misma BD.

En el nivel físico, un registro puede describirse como un bloque de posiciones de memoria consecutivas. En el nivel conceptual, cada registro se describe por medio de una definición de tipo, y se define la interrelación entre los tipos.

Finalmente, en el nivel de visión, se definen varias visiones de la BD. Por ejemplo, los cajeros sólo ven la información sobre las cuentas de los clientes, sin poder acceder a los salarios de los empleados.

La interrelación entre estos tres niveles de abstracción se ilustra en la figura 2.1.2.

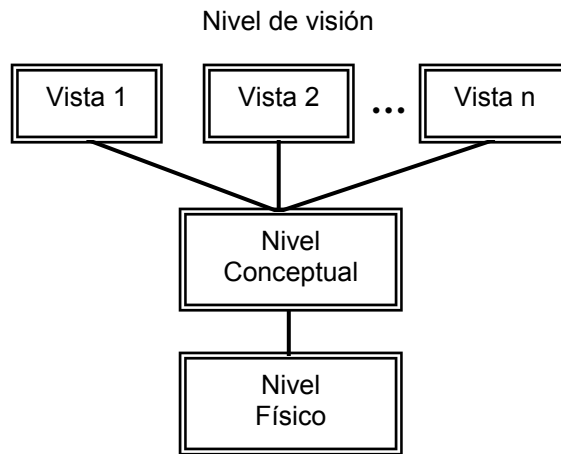


Figura 2.1.2. Niveles de Abstracción

### 2.1.3 Modelos lógicos.

- *Modelos lógicos basados en objetos.* Los modelos lógicos basados en objetos se usan para describir datos en el nivel conceptual y de visión. Se caracterizan porque proporcionan capacidad de estructuración bastante flexible y permiten especificar restricciones de datos explícitamente. Hay muchos modelos diferentes, y seguramente aparecerán más. Algunos de los más conocidos son el modelo entidad-relación, el orientado a objetos, el binario, el semántico de datos y el modelo funcional de datos. El modelo entidad-relación ha ganado aceptación y se utiliza ampliamente en la práctica, el modelo orientado a objetos incluye muchos conceptos del anterior, y esta ganando aceptación rápidamente.
- *Modelos lógicos basados en registros.* Los modelos lógicos basados en registros se utilizan para describir datos en los modelos conceptual y físico. A diferencia de los modelos lógicos basados en objetos, se usan para especificar la estructura lógica global de la BD y para proporcionar una descripción a nivel más alto de la implementación. Los modelos basados en registros se llaman así porque la BD está estructurada en registros de formato fijo de varios tipos.

Cada tipo de registro define un número fijo de campos, o atributos, y cada campo normalmente es de longitud fija. La estructura más rica de estas BD a menudo lleva a registros de longitud variable en el nivel físico. Los modelos basados en registros no incluyen un mecanismo para la representación directa de código de la BD, en cambio, hay lenguajes separados que se asocian con el modelo para expresar consultas y actualizaciones. Los tres modelos de datos más aceptados son los modelos relacional, de red y jerárquico. El modelo relacional ha ganado aceptación por encima de los otros.

#### **2.1.4 Modelo entidad-relación (E-R).**

El modelo entidad-relación fue inventado por Peter Chen en los años setenta. El propósito de este modelo es simplificar el diseño de bases de datos a partir de descripciones textuales de los requerimientos.

Los elementos esenciales del modelo son las entidades, los atributos y las relaciones entre las entidades. Una entidad es un objeto que existe y que es distinguible de otros objetos. Una entidad puede ser concreta (como una persona o un libro) o abstracta (como un algoritmo, un curso o una dirección en Internet).

Las entidades tienen atributos. Un atributo de una entidad es una característica interesante sobre ella, es decir, representa alguna propiedad que nos interesa almacenar. Por ejemplo, una persona tiene un nombre, una fecha de nacimiento y una CURP; algunos de los valores de estos atributos los comparte con otras personas, y otros son exclusivos de él. La persona además tiene otras propiedades, como su peso o estatura, que no son interesantes. Podemos agrupar las entidades dependiendo de la clasificación que hagamos de los objetos que representan; entidades que representen objetos del mismo tipo tendrán los mismos atributos (con valores distintos o iguales). Llamaremos a estos grupos conjuntos de entidades. Los conjuntos de entidades se designan en el diagrama de E-R con un cuadrado que lleva su nombre dentro. Los atributos de cada uno de ellos se grafican usando círculos o elipses que "cuelgan" del cuadrado que

representa el conjunto respectivo. A modo de ejemplo, la representación de una entidad persona con sus atributos se presenta en la figura 2.1.4.1.

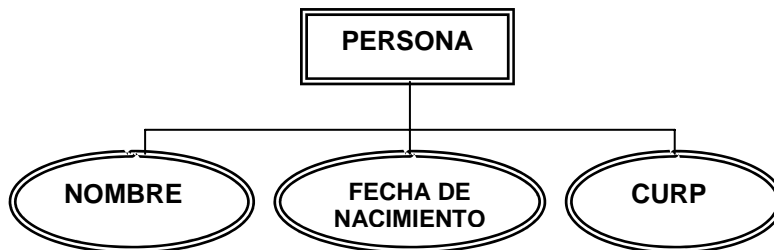


Figura 2.1.4.1. Entidad Persona y sus atributos

Una relación es una asociación entre entidades, generalmente dos. Una relación puede ocurrir entre dos entidades de un mismo conjunto de entidades (por ejemplo, un empleado es supervisado por su jefe, quien a su vez es otro empleado), o, en otras palabras, entre entidades de conjuntos distintos (por ejemplo, un curso es dictado por un profesor). Pueden existir relaciones entre más de dos conjuntos de entidades. Las relaciones se grafican en el diagrama de E-R con un rombo, unido con una línea a cada conjunto de entidades que relacionan; la relación entre las hipotéticas entidades curso y profesor se ilustra en la figura 2.4.1.2.



Figura 2.4.1.2. Dos entidades y una relación entre ellas

### 2.1.5 Modelo de red.

Los datos en el modelo de red se representan mediante colecciones de registros y las relaciones entre los datos se representan mediante enlaces, los cuales pueden verse como punteros.

### **2.1.6 Modelo jerárquico.**

El modelo jerárquico es similar al modelo de red, los datos y las relaciones se representan mediante registros y enlaces. Se diferencia del modelo de red en que los registros están organizados como colecciones de árboles.

### **2.1.7 Modelo orientado a objetos.**

Al igual que el modelo Entidad-Relación, el modelo orientado a objetos se basa en una colección de objetos. Un objeto contiene valores acumulados en variables instancia dentro de él, y estos valores son objetos por sí mismos. Así los objetos contienen objetos a un nivel de anidamiento arbitrario. Un objeto también contiene partes de código, que operan sobre el objeto, que se denominan métodos.

Los objetos que contienen los mismos tipos de valores y los mismos métodos se agrupan en clases. Una clase puede ser vista como una definición de tipo para objetos. La única forma en que un objeto puede acceder a los datos de otro objeto es invocando a un método de éste otro objeto. Esto se llama envío de un mensaje al objeto. Así la interfaz de llamada de los métodos de un objeto define su parte visible externamente, la parte interna de un objeto (las variables de instancia y el código de método) no son visibles externamente.

### **2.1.8 Comparación entre modelos.**

Los modelos relacionales se diferencian de los modelos de red y jerárquico en que no usan punteros o enlaces.

En cambio, el modelo relacional conecta registros mediante los valores que éstos contienen. Esta libertad del uso de punteros permite que se defina una base matemática formal.

### **2.1.9 Modelos físicos de datos.**

Los modelos físicos de datos se usan para describir datos en el nivel más bajo. Hay muy pocos modelos físicos de datos en uso, siendo los más conocidos el modelo unificador y de memoria de elementos.

### **2.1.10 Instancias y Esquemas.**

Las BD cambian a lo largo del tiempo según se añade y se suprime información. La colección de información almacenada en un determinado momento en el tiempo, se llama instancia de la BD. El diseño global de la BD se llama esquema de la BD, y se cambian muy raras veces.

El concepto de esquema corresponde a la noción de definición de tipo en un lenguaje de programación, y el concepto del valor de una variable corresponde al concepto de una instancia de un esquema de la BD.

Los sistemas de BD tienen varios esquemas, divididos de acuerdo a los niveles de abstracción tratados, el esquema físico, el esquema conceptual y los subesquemas. En general, los sistemas de BD soportan un esquema físico, otro conceptual y varios subesquemas.

### **2.1.11 Independencia de datos.**

La capacidad de modificar una definición de un esquema de un nivel sin afectar la definición de un esquema en el nivel superior siguiente se llama independencia de datos. Hay dos niveles de independencia de datos:

1. Independencia física de datos. Es la capacidad de modificar el esquema físico sin provocar que se vuelvan a escribir los programas de aplicación.
2. Independencia lógica de datos. Es la capacidad de modificar el esquema conceptual sin provocar que se vuelvan a escribir los programas de aplicación.

La independencia lógica de datos es más difícil de lograr, ya que los programas de aplicación son fuertemente dependientes de la estructura lógica de los datos a los que acceden.

El concepto de independencia de datos es similar al concepto de tipos abstractos de datos, ambos ocultan detalles de implementación.

### **2.1.12 Lenguaje de Definición de datos (DDL).**

Un esquema de BD se especifica por medio de un conjunto de definiciones que se expresan mediante un lenguaje especial llamado lenguaje de definición de datos (Data Definition Language, DDL). El resultado de la compilación de sentencias de DDL es un conjunto de tablas que se almacenan en un archivo especial que llamado diccionario de datos o directorio.

Un directorio de datos es un archivo que contiene metadatos, es decir, datos sobre datos. Este archivo se consulta antes de leer o modificar los datos reales en el sistema de BD.

La estructura de almacenamiento y los métodos de acceso se especifican por medio de un conjunto de definiciones en un tipo especial de DDL llamado lenguaje de almacenamiento y definición de datos. El resultado de la compilación de estas definiciones es un conjunto de instrucciones que especifican los detalles de implementación de los esquemas que normalmente se esconden a los usuarios.

### **2.1.13 Lenguaje de Manipulación de Datos (DML).**

Por manipulación de datos se entiende la recuperación y modificación de la información almacenada y la inserción y supresión de información.

A nivel físico, se definen algoritmos que permitan acceso eficiente a los datos. En los niveles de abstracción más altos, se pone énfasis en la facilidad de uso. El objetivo es proporcionar una interacción eficiente entre las personas y el sistema.



Un lenguaje de manipulación de datos (data manipulation language, DML) es un lenguaje que capacita a los usuarios para acceder o manipular los datos.

Existen básicamente dos tipos:

1. Procedimentales: requieren que el usuario especifique qué datos se necesitan y cómo conseguirlos.
2. No procedimentales: el usuario debe especificar qué datos se necesitan, pero no cómo obtenerlos.

Los DML no procedimentales normalmente son más sencillos de aprender y usar, sin embargo pueden generar código que no sea tan eficiente como el producido por los lenguajes procedimentales. Esta dificultad puede remediarse a través de varias técnicas de optimización.

Una consulta es una sentencia que solicita la recuperación de información. El trozo de un DML que implica recuperación de información se llama lenguaje de consultas. Aunque es incorrecto, suelen utilizarse los términos lenguaje de consultas y lenguaje de manipulación de datos como sinónimos.

#### **2.1.14 Sistema Gestor de Bases de Datos (DBMS).**

El sistema gestor de bases de datos es la porción más importante del software de un sistema de base de datos. Un DBMS es una colección de numerosas rutinas de software interrelacionadas, cada una de las cuales es responsable de alguna tarea específica. En otras palabras un DBMS es un módulo de programa que proporciona el interfaz entre los datos de bajo nivel almacenados y los programas de aplicación y consultas, y es responsable de las siguientes tareas:

1. Interacción con el gestor de archivos. El DBMS traduce las distintas sentencias DML a comandos del sistema de archivos de bajo nivel. Así, es responsable del verdadero almacenamiento de los datos.
2. Implantación de la integridad. Los valores de los datos que se almacenan deben satisfacer ciertos tipos de restricciones de consistencia, que debe

especificar explícitamente el administrador de la BD. El DBMS entonces puede determinar si se produce una violación de la restricción, si es así, se debe tomar la acción apropiada.

3. Implantación de la seguridad. Es trabajo del DBMS hacer que se cumplan los requisitos de seguridad.
4. Copia de seguridad y recuperación. Un sistema informático, como cualquier otro dispositivo, está sujeto a fallos. Es responsabilidad del DBMS detectar fallos y restaurar la BD al estado que existía antes de ocurrir el fallo.

Esto se lleva a cabo con procedimientos de copias de seguridad y recuperación.

5. Control de concurrencia. Cuando varios usuarios actualizan la BD de forma concurrente, es posible que no se conserve la consistencia de los datos. Controlar la interacción entre los usuarios concurrentes es otra responsabilidad del gestor de la BD.

Los sistemas de BD diseñados para computadoras personales pequeñas pueden no tener todas las características especificadas anteriormente.

### **2.1.15 Administrador de Bases de Datos (DBA).**

Una de las razones principales para tener DBMS es tener control central de los datos y de los programas que acceden a esos datos. La persona que tiene dicho control central sobre el sistema se llama administrador de la base de datos (DataBase Administrator, DBA).

Las funciones del DBA son:

1. Definición de esquema. El esquema original de la BD se crea escribiendo un conjunto de definiciones que son traducidas por el compilador de DDL a un conjunto de tablas, almacenadas permanentemente en el diccionario de datos.

2. Definición de la estructura de almacenamiento y del método de acceso. Se crean escribiendo un conjunto de definiciones traducidas por el compilador del lenguaje de almacenamiento y definición de datos.
3. Modificación del esquema y de la organización física. Las modificaciones son poco comunes, pero se logran escribiendo un conjunto de definiciones usadas por el compilador DDL para generar modificaciones a las tablas internas apropiadas.
4. Concesión de autorización para el acceso a los datos. La concesión de diferentes tipos de autorización permite al DBA regular qué partes de la BD van a poder ser accedidas por varios usuarios.
5. Especificación de las restricciones de integridad. Las restricciones de integridad se mantienen en una estructura especial del sistema que consulta el gestor de la BD cada vez que tiene lugar una actualización.

#### **2.1.16 Usuarios de Bases de Datos.**

Se pueden definir cuatro tipos de usuarios de BD, diferenciados por la forma de interactuar con el sistema:

1. Programadores de aplicaciones. Son quienes interactúan con el sistema por medio de llamadas en DML, incorporadas en un programa escrito en un lenguaje principal (como Pascal o C). Estos programas se denominan comúnmente programas de aplicación.

Hay tipos especiales de lenguajes de programación que combinan estructuras de control de lenguajes como Pascal con estructuras de control para el manejo de un objeto de la BD, como relaciones. Estos lenguajes. A veces llamados lenguajes de cuarta generación, a menudo incluyen características especiales para facilitar la generación de formas y la presentación de datos en pantalla. La mayoría de los DBMS comerciales incluyen un lenguaje de cuarta generación.

2. Usuarios sofisticados. Interaccionan con el sistema sin escribir programas, en cambio escriben sus preguntas en un lenguaje de consultas. Cada consulta se somete a un procesador de consultas, cuya función es tomar una sentencia en DML y descomponerla en instrucciones que entienda el DBMS.
3. Usuarios especializados. Algunos usuarios sofisticados escriben aplicaciones de BD especializadas que no encajan en el marco tradicional de procesamiento de datos, como diseño asistido por computador, sistemas basados en el conocimiento, etc.
4. Usuarios ingenuos. Los usuarios no sofisticados interactúan con el sistema invocando a uno de los programas de aplicación permanentes que se han escrito anteriormente.

#### **2.1.17 Estructura General del Sistema.**

Un sistema de base de datos se encuentra dividido en módulos cada uno de los cuales controla una parte de la responsabilidad total de sistema. En la mayoría de los casos, el sistema operativo proporciona únicamente los servicios más básicos y el sistema de la base de datos debe partir de esa base y controlar además el manejo correcto de los datos. Así el diseño de un sistema de base de datos debe incluir la interfaz entre el sistema de base de datos y el sistema operativo.

Los componentes funcionales de un sistema de base de datos, son:

- *Gestor de archivos*. Gestiona la asignación de espacio en la memoria del disco y de las estructuras de datos usadas para representar información.
- *Manejador de base de datos*. Sirve de interfaz entre los datos y los programas de aplicación.
- *Procesador de consultas*. Traduce las proposiciones en lenguajes de consulta a instrucciones de bajo nivel. Además convierte la solicitud de usuario en una forma más eficiente.

- *Compilador de DDL*. Convierte las proposiciones DDL en un conjunto de tablas que contienen metadatos, estas se almacenan en el diccionario de datos.

Además se requieren varias estructuras de datos como parte de la implementación del sistema físico, como:

- *Archivo de datos*. En él se encuentran almacenados físicamente los datos de una organización.
- *Diccionario de datos*. Contiene la información referente a la estructura de la base de datos.
- *Índices*. Permiten un rápido acceso a registros que contienen valores específicos.

Una forma gráfica de representar los componentes antes mencionados y la relación que existe entre ellos se muestra en la figura 2.1.17.

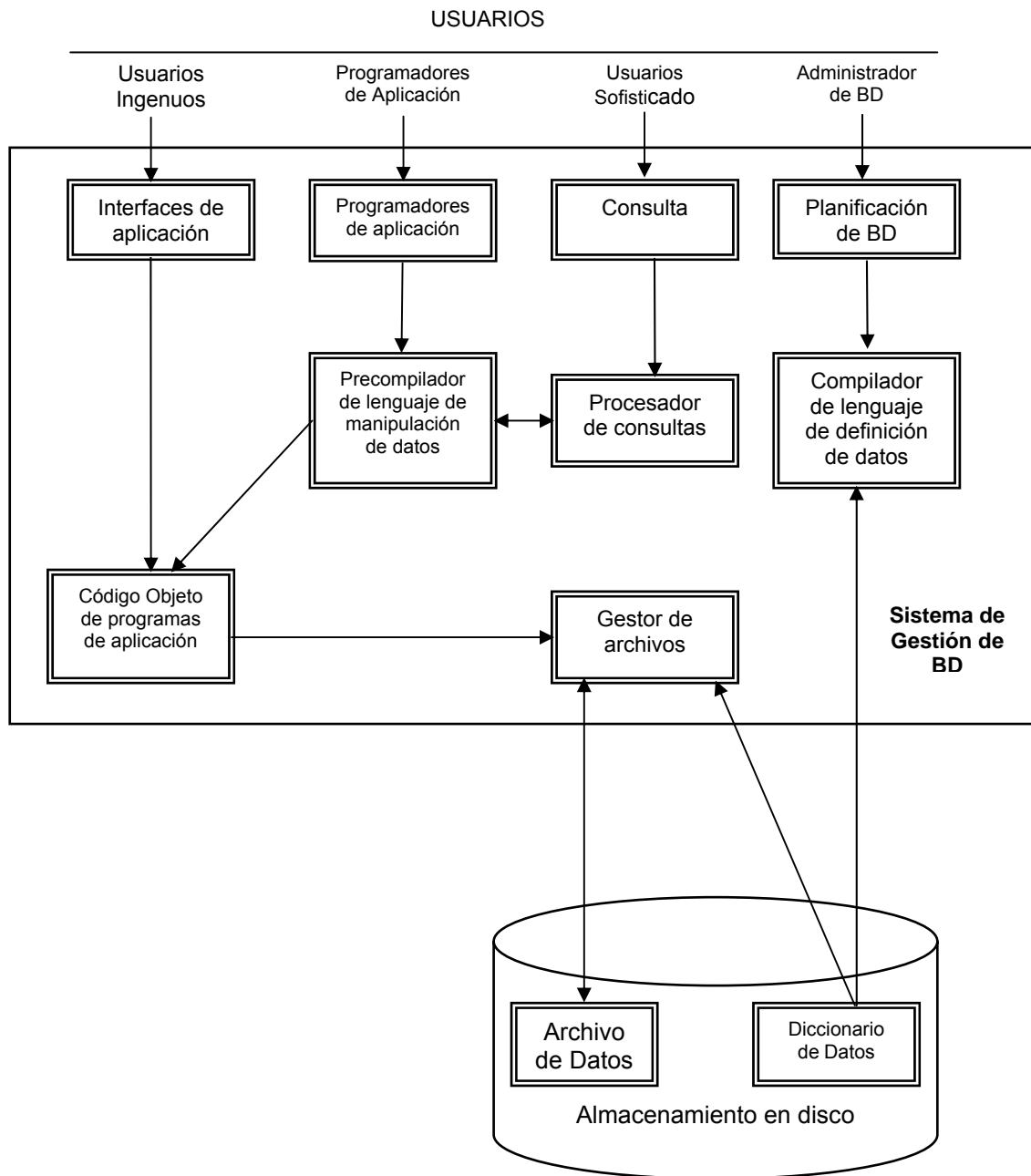


Figura 2.1.17. Estructura General de un sistema de Bases de Datos

### 2.1.18 Normalización

Dentro del análisis de un sistema de base de datos, se debe descomponer dicha base en grupos y segmentos de registros. Este proceso es la descomposición; el mismo es necesario independientemente de la arquitectura de la base de datos relacional, red o jerárquica. Sin embargo, para la base de datos relacional, la

acción correspondiente puede dividirse y expresarse en términos formales y se denomina normalización a la misma.

La normalización de datos, o normalización como tal, es un proceso de refinamiento de las estructuras de la base de datos para mejorar la velocidad a la que los datos puedan accederse, así como mejorar su integridad. La normalización convierte una relación en varias subrelaciones, cada una de las cuales obedece a reglas. Estas reglas se describen en términos de dependencia. Una vez que se definan las distintas formas de dependencia, se encontrarán procedimientos a aplicar a las relaciones de modo tal que las mismas puedan descomponerse de acuerdo a la dependencia que prevalece. Esto implicará la formación de varias subrelaciones a partir de la única relación preexistente.

Otra ventaja de la normalización de una base de datos es el consumo de espacio. Una base de datos normalizada puede ocupar menos espacio en disco que una no normalizada. Hay menos repetición de datos, lo que tiene como consecuencia un mucho menor uso de espacio en disco.

#### Grados de normalización

Existen básicamente tres niveles de normalización: Primera Forma Normal (1NF), Segunda Forma Normal (2NF) y Tercera Forma Normal (3NF). Cada una de estas formas tiene sus propias reglas.

- *Primera Forma Normal.* La regla de la Primera Forma Normal establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas.
- *Segunda Forma Normal.* La regla de la Segunda Forma Normal establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Una dependencia parcial es un término que describe a aquellos datos que no dependen de la clave de la tabla para identificarlos.
- *Tercera Forma Normal.* La regla de la Tercera Forma Normal señala que se debe eliminar y separar cualquier dato que no sea clave. El valor de esta

columna debe depender de la clave. Todos los valores deben identificarse únicamente por la clave.

Existen tres niveles más de normalización que no se han discutido aquí. Ellos son Forma Normal Boyce-Codd, Cuarta Forma Normal (4NF), Quinta Forma Normal (5NF) o Forma Normal de Proyección-Unión, Forma Normal de Proyección-Unión Fuerte, Forma Normal de Proyección-Unión Extra Fuerte y Forma Normal de Clave de Dominio. Estas formas de normalización pueden llevar las cosas más allá de lo que necesita. Éstas existen para hacer una base de datos realmente relacional. Tienen que ver principalmente con dependencias múltiples y claves relacionales.

La normalización es una técnica que se utiliza para crear relaciones lógicas apropiadas entre tablas de una base de datos.

Ayuda a prevenir errores lógicos en la manipulación de datos. La normalización facilita también agregar nuevas columnas sin romper el esquema actual ni las relaciones.

Normalizar demasiado puede conducir a tener una base de datos ineficiente y hacer a su esquema demasiado complejo para trabajar.

#### **2.1.19 ACID.**

En el contexto de bases de datos se denomina ACID a la propiedad de una base de datos para realizar transacciones seguras. Así pues ACID define a un sistema de gestión de bases de datos que puede realizar transacciones seguras.

ACID es un acrónimo de Atomicity, Consistency, Isolation and Durability: Durabilidad, Aislamiento, Consistencia e Indivisibilidad.

- *Atomicidad (Indivisible)*. La atomicidad de una transacción garantiza que todas sus acciones sean realizadas o ninguna sea ejecutada, en el caso de



la transacción bancaria o se ejecuta tanto el "deposito-deducción" o ninguna acción será realizada. Esta propiedad es de suma importancia ante un fallo del sistema, ya que asegura que ninguna operación quede a medias.

- *Consistencia*. Es la propiedad que asegura que sólo se empieza aquello que se puede acabar. Por lo tanto se ejecutan aquellas operaciones que no van a romper la reglas y directrices de integridad de la base de datos. Muy similar a la "Atomicidad", la consistencia garantiza que las reglas que hayan sido declaradas para una transacción sean cumplidas, regresando a la transacción bancaria, supongamos que cada vez que se realice una transferencia inter-bancaria de \$100,000 sea necesario notificar a la sucursal del tarjeta-habiente, si no es posible comunicarse y actualizar la información en la sucursal del cliente, toda la transacción será abortada.
- *Aislamiento*. Es la propiedad que asegura que una operación no puede afectar a otras. Esto garantiza que las transacciones que se estén realizando en el sistema sean invisibles a todos los usuarios hasta que éstas hayan concluido, en la transacción bancaria es posible que el sistema este programado para intentar en 5 o 10 ocasiones más antes de abortar una transacción por completo. Este aislamiento garantiza que los usuarios del sistema no observen estos cambios intermedios hasta que sea finalizada la última acción de actualización, asegurando que dos transacciones sobre la misma información nunca generará ningún tipo de error.
- *Durabilidad*. La durabilidad de una transacción garantiza que al instante en el que se finaliza la transacción esta perdure a pesar de otras consecuencias, esto es, si la energía eléctrica falla, el sistema aún será capaz de recordar todas las transacciones que han sido realizadas en el sistema.

## **2.2 Redes.**

### **2.2.1 Definición.**

- Una red de computadoras consiste en una o más computadoras conectadas por un medio físico y que ejecutan un software que permite a las computadoras comunicarse unas con las otras.
- Red es un sistema de comunicaciones, ya que permite comunicarse con otros usuarios y compartir archivos y periféricos. Es decir es un sistema de comunicaciones que conecta a varias unidades y que les permite intercambiar información.
- En tecnología de información, una red es una serie de puntos o de nodos interconectados por las trayectorias de comunicación. Las redes pueden interconectar con otras redes y contener subredes.

La topología más común, o configuración general de redes, incluye el bus, la estrella, y las topologías token-ring. Las redes se pueden clasificar también en términos de la separación física entre nodos, como redes de área local (LAN, local area network), redes de área metropolitana (MAN, metropolitan area network), y redes de área amplia (WAN, Wide area network).

Una cierta clase de redes puede también ser clasificada por el tipo de tecnología de la transmisión de datos que se emplea. Por ejemplo, una red TCP/IP (Transport Control Protocol/Internet Protocol), o una red del tipo SNA (Systems Network Architecture); si transporta voz, datos, o ambas clases de señales; por quién puede utilizar la red (pública o privada); por la naturaleza de sus conexiones (conmutada, dedicada o no dedicada, o por conexión virtual); y por los tipos de conexiones físicas (por ejemplo, fibra óptica, cable coaxial, y par trenzado sin blindaje UTP). Las grandes redes de telefonía y las redes que usan su infraestructura (tal como el Internet) disponen de acuerdos para compartir e intercambiar recursos con otras compañías para formar redes mucho más grandes.

### 2.2.2 Clasificación de redes.

Podemos clasificar las redes en las dimensiones de la tecnología de transmisión y por su alcance geográfico.

#### Por su Tecnología de Transmisión:

- *Broadcast*. Las redes de difusión (broadcast) tienen un solo canal de comunicación, por lo que todas las máquinas de la red lo comparten.
- *Point to point (punto a punto)*. Constan de muchas conexiones entre pares individuales de máquinas.

#### Por su alcance geográfico:

- *LAN (Local Area Network)*: Una red de área local es un sistema de comunicaciones constituido por un conjunto de hardware (cableado, dispositivos, PC's, servidores, etc.) y software que se distribuyen por una extensión limitada (planta, edificio, grupo de edificios) en el que existen una serie de recursos compatibles (discos, impresoras, bases de datos, etc.), a los que pueden tener acceso los usuarios para compartir información de trabajo.
- *Redes Ethernet*. A la cual también se conoce como IEEE 802.3, es actualmente el estándar más popular para redes LAN. El estándar 802.3 emplea una topología de estrella o de bus. Ethernet permite transmitir datos a través de la red a una velocidad de 10 Mbps, usa un método de transmisión de datos conocido como Acceso Múltiple con Detección de Portadora y Detección de Colisiones (CSMA/CD).
- *MAN (Metropolitan Area Network)*: Las redes de área metropolitana cubren extensiones mayores como pueden ser ciudades.
- *WAN (Wide Area Network)*: Redes de Cobertura Amplia son redes que cubren una amplia región geográfica, a menudo un país o un continente.

### 2.2.3 Topologías.

#### 2.2.3.1 Topología de red.

La topología de red es la disposición física en la que se conectan los nodos de una red de ordenadores o servidores. Estas computadoras pueden conectarse de muchas y muy variadas maneras. La conexión más simple es un enlace unidireccional entre dos nodos. Se puede añadir un enlace de retorno para la comunicación en ambos sentidos. Los cables de comunicación modernos normalmente incluyen más de un cable para facilitar esto, aunque redes muy simples basadas en buses tienen comunicación bidireccional en un solo cable.

La topología de red la determina únicamente la configuración de las conexiones entre nodos. La distancia entre los nodos, las interconexiones físicas, las tasas de transmisión y/o los tipos de señales no pertenecen a la topología de la red, aunque pueden verse afectados por la misma.

#### 2.2.3.2 Red en bus.

Red en bus se refiere a la topología de red en la que todas las estaciones están conectadas a un único canal de comunicaciones por medio de unidades interfaz y multiplexores. Las estaciones utilizan este canal para comunicarse con el resto. Es la más sencilla por el momento. Como se muestra en la figura 2.2.3.2.

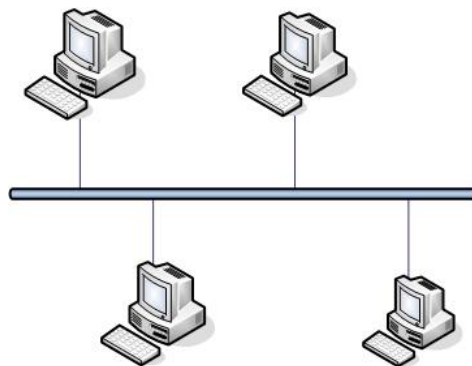


Figura 2.2.3.2. Topología de red.

### 2.2.3.3 Topología en estrella.

La topología en estrella es una topología de red muy común en las redes de computadoras. En su forma más simple, una topología en estrella consta de un switch central, o hub que actúa como un router para retransmitir los mensajes. Como se muestra en la figura 2.2.3.3.

Cuando se aplica a redes basadas en bus, este hub central retransmite todas las transmisiones recibidas desde cualquier nodo periférico a todos los nodos periféricos de la red, a veces incluso al nodo original. Todos los nodos periféricos se pueden comunicar con los demás transmitiendo o recibiendo del nodo central solamente. Un fallo en la línea de conexión de cualquier nodo con el nodo central provocaría el aislamiento de ese nodo respecto a los demás, pero el resto de sistemas permanecería intacto.

Se utiliza sobre todo para redes locales. La mayoría de las redes de área local que tienen un router, un switch o un hub siguen esta topología. El nodo central en estas sería el hub, el router o el switch, por el que pasan todos los paquetes.

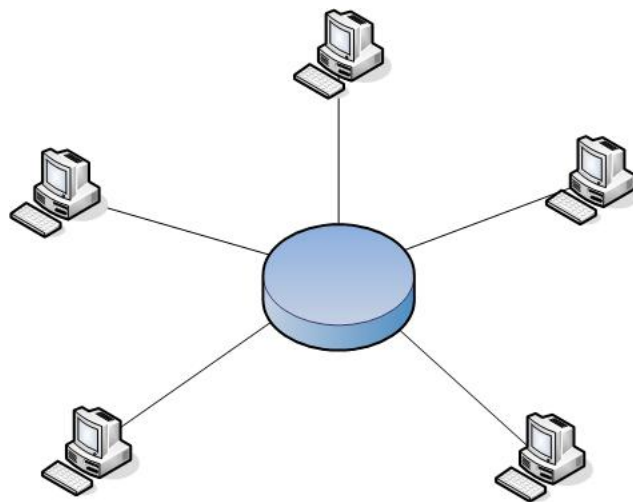


Figura 2.2.3.3. Topología Estrella

#### 2.2.3.4 Red en anillo.

Topología de red en la que las estaciones se conectan formando un anillo. Cada estación está conectada a la siguiente y la última está conectada a la primera. Cada estación tiene un receptor y un transmisor que hace la función de repetidor, pasando la señal a la siguiente estación del anillo. Como se muestra en la figura 2.2.3.4.

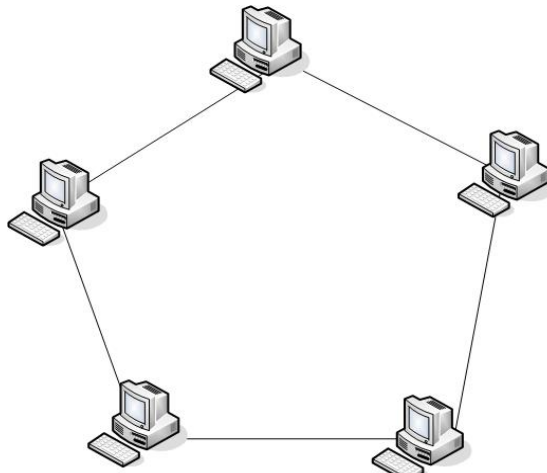


Figura 2.2.3.4. Topología anillo

#### 2.2.3.5 Red de árbol.

Topología de red en la que los nodos están colocados en forma de árbol. Desde una visión topológica, la conexión en árbol es parecida a una serie de redes en estrella interconectadas salvo en que no tiene un nodo central. En cambio, tiene un nodo de enlace troncal, generalmente ocupado por un hubs o switch, desde el que se ramifican los demás nodos. Es una variación de la red en bus, la falla de un nodo no implica interrupción en las comunicaciones. Se comparte el mismo canal de comunicaciones. Como se muestra en la figura 2.2.3.5.

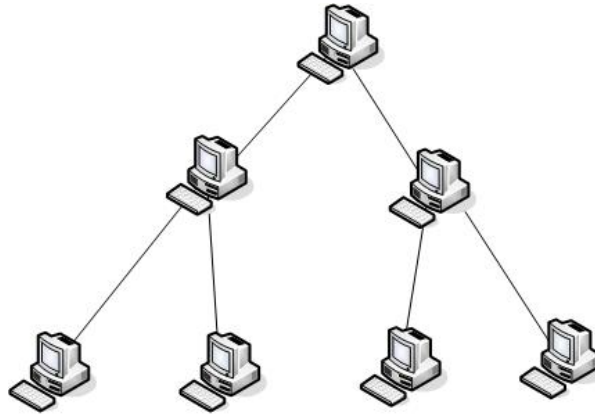


Figura. 2.2.3.5. Topología Árbol

### 2.2.3.6 Red de malla.

La Topología en malla es una topología de red en la que cada nodo está conectado a uno o más de los otros nodos. De esta manera es posible llevar los mensajes de un nodo a otro por diferentes caminos.

Si la red de malla está completamente conectada no puede existir absolutamente ninguna interrupción en las comunicaciones.

Cada servidor tiene sus propias conexiones con todos los demás servidores.

Esta topología, a diferencia de otras (como topología en árbol y topología en estrella), no requiere de un servidor o nodo central, con lo que se reduce el mantenimiento (un error en un nodo, sea importante o no, no implica la caída de toda la red).

Una red con topología en malla ofrece una redundancia y fiabilidad superiores. En una topología en malla, cada equipo está conectado a todos los demás equipos. Aunque la facilidad de solución de problemas y el aumento de la fiabilidad son ventajas muy interesantes, estas redes resultan caras de instalar, ya que utilizan mucho cableado. Como se puede observar en la figura 2.2.3.6.

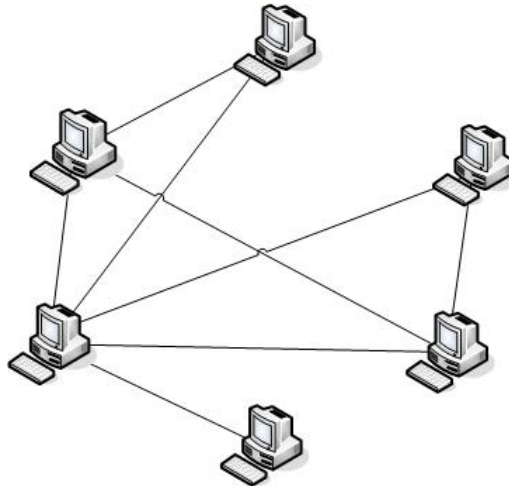


Figura. 2.2.3.6 Topología Malla

## 2.2.4 Modelos de referencia.

### 2.2.4.1 Modelo OSI. (Open System Interconnection)

El modelo de referencia de Interconexión de Sistemas Abiertos (OSI, Open System Interconnection) lanzado en 1984 fue el modelo de red descriptivo creado por ISO. Proporcionó a los fabricantes un conjunto de estándares que aseguraron una mayor compatibilidad e interoperabilidad entre los distintos tipos de tecnología de red producidos por las empresas a nivel mundial. Como se puede observar en la tabla 2.2.4.1.

Capa Física	(Capa 1)
Capa de enlace de datos	(Capa 2)
Capa de red	(Capa 3)
Capa de transporte	(Capa 4)
Capa de sesión	(Capa 5)
Capa de presentación	(Capa 6)
Capa de aplicación	(Capa 7)

*Tabla 2.2.4.1. Capas que contiene el Modelo OSI*

#### 2.2.4.1.1 Capa Física (Capa 1).

La Capa Física del modelo de referencia OSI es la que se encarga de las conexiones físicas de la computadora hacia la red, tanto en lo que se refiere al medio; característica del medio y la forma en la que se transmite la información.



Es la encargada de transmitir los bits de información a través del medio utilizado para la transmisión.

#### **2.2.4.1.2 Capa de enlace de datos (Capa 2).**

A partir de cualquier medio de transmisión debe ser capaz de proporcionar una transmisión sin errores, es decir, un tránsito de datos fiable a través de un enlace físico. Debe crear y reconocer los límites de las tramas, así como resolver los problemas derivados del deterioro, pérdida o duplicidad de las tramas. También puede incluir algún mecanismo de regulación del tráfico que evite la saturación de un receptor que sea más lento que el emisor.

La capa de enlace de datos se ocupa del direccionamiento físico, de la topología de la red, del acceso a la red, de la notificación de errores, de la distribución ordenada de tramas y del control del flujo.

#### **2.2.4.1.3 Capa de red (Capa 3).**

El cometido de la capa de red es hacer que los datos lleguen desde el origen al destino, aún cuando ambos no estén conectados directamente. Es decir que se encarga de encontrar un camino manteniendo una tabla de enrutamiento y atravesando los equipos que sea necesario, para hacer llegar los datos al destino. Los equipos encargados de esto se denominan routers.

Adicionalmente la capa de red debe administrar la congestión de red, que es el fenómeno que se produce cuando una saturación de un nodo tira abajo toda la red.

#### **2.2.4.1.4 Capa de transporte (Capa 4).**

Su función básica es aceptar los datos enviados por las capas superiores, dividirlos en pequeñas partes si es necesario, y pasarlos a la capa de red. En el caso del modelo OSI, también se asegura que lleguen correctamente al otro lado de la comunicación. Otra característica a destacar es que debe aislar a las capas

superiores de las distintas posibles implementaciones de tecnologías de red en las capas inferiores, lo que la convierte en el corazón de la comunicación. En esta capa se proveen servicios de conexión para la capa de sesión que serán utilizados finalmente por los usuarios de la red al enviar y recibir paquetes. Estos servicios estarán asociados al tipo de comunicación empleada, la cual puede ser diferente según el requerimiento que se le haga a la capa de transporte.

Podemos definir a la capa de transporte como:

“Capa encargada de efectuar el transporte de los datos (que se encuentran dentro del paquete) de la máquina origen a la destino, independizándolo del tipo de red física que se esté utilizando”.

#### **2.2.4.1.5 Capa de sesión (Capa 5).**

Esta capa ofrece varios servicios que son cruciales para la comunicación, como son:

- Control de la sesión a establecer entre el emisor y el receptor (quién transmite, quién escucha y seguimiento de ésta).
- Control de la concurrencia (que dos comunicaciones a la misma operación crítica no se efectúen al mismo tiempo).

Mantener puntos de verificación (checkpoints), que sirven ante una interrupción de transmisión por cualquier causa, la misma se pueda reanudar desde el último punto de verificación en lugar de repetirla desde el principio.

El servicio provisto por esta capa es la capacidad de asegurar que, dada una sesión establecida entre dos máquinas, la misma se pueda efectuar para las operaciones definidas de principio a fin, reanudándolas en caso de interrupción.

#### **2.2.4.1.6 Capa de presentación (Capa 6).**

El objetivo de la capa de presentación es encargarse de la representación de la información, de manera que aunque distintos equipos puedan tener diferentes representaciones internas.

Esta capa es la primera en trabajar más el contenido de la comunicación que cómo se establece la misma. En ella se tratan aspectos tales como la semántica y la sintaxis de los datos transmitidos, ya que distintas computadoras pueden tener diferentes formas de manejarlas.

Se puede definir esta capa como la encargada de manejar las estructuras de datos abstractas y realizar las conversiones de representación de datos necesarias para la correcta interpretación de los mismos.

Esta capa también permite cifrar los datos y comprimirlos.

#### **2.2.4.1.7 Capa de aplicación (Capa 7).**

Ofrece a las aplicaciones (de usuario o no) la posibilidad de acceder a los servicios de las demás capas y define los protocolos que utilizan las aplicaciones para intercambiar datos, como correo electrónico (POP y SMTP), gestores de bases de datos y servidor de ficheros (FTP). Hay tantos protocolos como aplicaciones distintas y puesto que continuamente se desarrollan nuevas aplicaciones el número de protocolos crece sin parar.

### **2.2.4.2 Modelo Cliente/Servidor.**

#### **2.2.4.2.1 Definiciones.**

- Cualquier combinación de sistemas que pueden colaborar entre si para dar a los usuarios toda la información que ellos necesiten sin que tengan que saber donde esta ubicada.

- Es una arquitectura de procesamientos cooperativo donde uno de los componentes pide servicios a otro.
- Es un procesamiento de datos de índole colaborativo entre dos o más computadoras conectadas a una red.
- El término cliente/servidor es originalmente aplicado a la arquitectura de software que describe el procesamiento entre dos o más programas: una aplicación y un servicio.
- IBM define al modelo Cliente/Servidor. "Es la tecnología que proporciona al usuario final el acceso transparente a las aplicaciones, datos, servicios de cómputo o cualquier otro recurso del grupo de trabajo y/o, a través de la organización, en múltiples plataformas. El modelo soporta un medio ambiente distribuido en el cual los requerimientos de servicio hechos por estaciones de trabajo inteligentes o "clientes", resultan en un trabajo realizado por otras computadoras llamadas servidores".
- La arquitectura cliente/servidor llamado modelo cliente/servidor o servidor-cliente es una forma de dividir y especializar programas y equipos de cómputo a fin de que la tarea que cada uno de ellos realizada se efectúe con la mayor eficiencia, y permita simplificarlas.
- "Es un modelo para construir sistemas de información, que se sustenta en la idea de repartir el tratamiento de la información y los datos por todo el sistema informático, permitiendo mejorar el rendimiento del sistema global de información".

#### **2.2.4.2.2 Modelo Cliente/Servidor.**

Bajo esta filosofía, en absoluto exclusiva de los sistemas operativos, se esconde la idea de liberar al núcleo del sistema operativo de una gran parte de las funciones asociadas a éste. Dado la mayor parte de esas funciones se definen como servicios ofrecidos por el sistema, aparecen los conceptos de proceso servidor (el que ofrecer un servicio) y proceso cliente (el que lo utiliza). De esta manera, se consigue separar el sistema operativo en partes, controladas por procesos de usuario, donde cada parte se encarga de una faceta del sistema. El sistema

operativo en si, se limita a poco más que controlar la comunicación entre los clientes y los servidores.

Los sistemas clientes-servidor pueden ser de muchos tipos, dependiendo de las aplicaciones que el servidor pone a disposición de los clientes. Entre otros existen:

- *Servidores de Impresión*, mediante el cual los usuarios comparte impresoras
- *Servidores de Archivos*, con el cual los clientes comparten discos duros
- *Servidores de Bases de Datos*, donde existe una única base de datos

La tecnología denominada cliente/servidor es utilizada por todas las aplicaciones de Internet/Intranet. Un único servidor típicamente sirve a una multitud de clientes, ahorrando a cada uno de ellos el problema de tener la información instalada y almacenadas localmente.

En resumen, un servidor comienza su ejecución antes del inicio de cualquier interacción y (usualmente) continua aceptando peticiones y enviando respuestas indefinidamente. Un cliente es cualquier programa que efectúa una petición y espera una respuesta; (generalmente) termina después de de usar un servidor un numero finito de veces.

Un servidor espera una petición sobre un puerto bien conocido, el cual ha sido usado para poder comunicarse. Como se muestra en la siguiente figura 2.2.4.2.2.

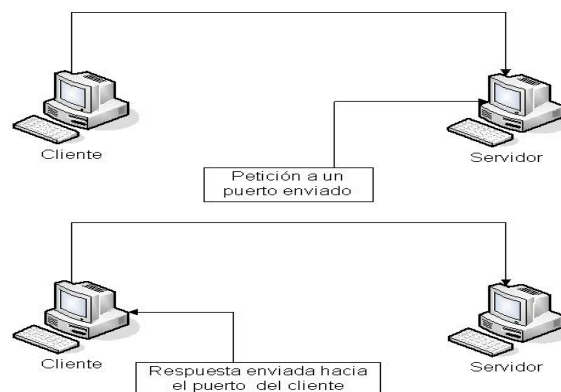


Figura 2.2.4.2.2. Modelo Cliente/Servidor

### **2.2.4.2.3 Hardware cliente/servidor**

La arquitectura cliente/servidor permite al usuario en una máquina, llamado cliente, requerir algún tipo de servicio de una máquina a la que está unido llamado el servidor, mediante una red como una LAN (Red de Área Local) o una WAN (Red de Área Mundial). Estos servicios pueden ser peticiones de datos de una base de datos, de información contenida en archivos o los archivos en sí mismos o peticiones de imprimir datos en una impresora asociada. Aunque los clientes y servidores suelen verse como máquinas separadas, pueden de hecho, ser dos áreas separadas en la misma máquina.

Una máquina cliente unida a un servidor puede ser a su vez servidor de otro cliente y el servidor puede ser otro cliente de otro servidor en red. También es posible tener el servidor corriendo en un sistema operativo y el cliente en otro distinto.

Hay varios tipos de máquinas clientes en entornos cliente/servidor. Uno de los clientes más populares es una computadora personal basada en Intel que ejecuta aplicaciones de DOS en un entorno Windows. Otro cliente popular es la terminal X; de hecho, el sistema X Windows es un modelo cliente/servidor clásico. Hay también clientes UNIX que ejecutan sistemas operativos como UNIXWare. Un servidor que pide cosas a otro servidor es un cliente de la máquina a la que está pidiendo. Sin considerar el tipo de cliente que se está usando en una red cliente/servidor, se está realizando al menos una de las funciones básicas descritas aquí como funciones del cliente.

### **2.2.4.2.4 Servidor.**

Los servidores de una red cliente/servidor son los procesos que proporcionan información, recursos y servicios a los clientes de la red. Cuando un cliente pide un recurso como, por ejemplo, un archivo, datos de una base, accesos a aplicaciones remotas o impresión centralizada, el servidor proporciona estos

recursos al cliente. Como se menciona antes, los procesos del servidor pueden residir en una máquina que también actúa como cliente de otro servidor.

Además de proporcionar este tipo de recursos, un servidor puede dar acceso a otras redes, actuando como un servidor de comunicaciones que conecta a otros servidores o microcomputadoras que actúan como host de la red. También puede permitir enviar información como correo electrónico desde un cliente de una red a otro cliente de una red. Puede actuar como servidor de seguridad, como servidor de gestión de la red, como servidor multimedia, etc.

#### **2.2.4.2.5 Cliente.**

Los clientes en una red cliente/servidor son las máquinas o procesos que piden información, recursos y servicios a un servidor unido. Estas peticiones pueden ser cosas como proporcionar datos de una base de datos, aplicaciones, partes de archivos o archivos completos a la máquina cliente. Los datos, aplicaciones o archivos pueden residir en un servidor y ser simplemente accedidos por el cliente o pueden ser copiados o movidos físicamente a la máquina cliente. Esta disposición permite a la máquina cliente ser relativamente pequeña. Para cada tipo de entorno de cliente, hay habitualmente software específico (y a veces hardware) en el cliente, con algún software y hardware análogo en el servidor.

Los servidores pueden ser sistemas operativos diferentes como Windows NT, Windows, OS/2, UNIX, etc. Unix es popular porque como sistema operativo de servidores puede ser utilizado en muchos tipos de configuraciones sobre máquinas servidor además de como servidores de archivos y servidores de impresión.

#### **2.2.4.2.6 Cliente/Servidor en Web.**

Los servidores Web son aquellos que permiten a los clientes compartir datos, documentos, etc. Aunque parte de la tecnología cliente/servidor el servidor Web aporta unas ventajas adicionales en aspectos muy importantes como son:

- El Web se crea normalmente como un sistema abierto al que cualquiera puede contribuir y acceder desde cualquier punto de la red de la empresa: no requiere nombre de usuario ni contraseña como los sistemas tradicionales cliente/servidor.
- Los servidores Web muestran información con un simple click del ratón a través de un proceso de hipervínculo.
- La información servida puede ser de cualquier tipo gracias a la utilización de los estándares de Internet. Esa información es de solo lectura pues, a diferencia de los sistemas normales cliente/servidor, el usuario no puede hacer cambios en el dispositivo original de los datos.
- Dado que el servidor Web es de tan fácil acceso, ello hace posible publicar información de forma instantánea en toda la compañía mediante un simple almacenamiento de la misma en el servidor.
- El servidor Web del Intranet puede servir de la misma copia de un archivo a toda la organización de la misma forma que un único servidor Web de Internet puede servir al mundo entero. Así que ahora solo hay una única copia del archivo a actualizar, y cuando se actualiza, la nueva versión es servida a toda la compañía.
- La amplitud de la red suele ser mayor que otros sistemas cliente/servidor, pues la mayor parte de intranet se construyen WAN, mas que sobre redes LAN.

#### **2.2.4.2.7 El cliente Web.**

La herramienta o aplicación utilizada para explorar y navegar por el ámbito Web se conoce como Browser o cliente Web. Se trata de un cliente capaz de acceder, recibir, montar y mostrar documentos Web. Hay que recordar que también en este caso se está tratando con una arquitectura cliente/servidor, lo que supone la existencia de un servidor Web que recibe una petición y envía una respuesta, y un cliente que envía la petición, recibe la respuesta y la muestra al usuario. El cliente Web recibe en respuesta a su requerimiento un conjunto de archivos y las instrucciones para organizarlos y mostrarlos al usuario tal como desea el creador



del documento. Conforme va recibiendo los componentes, el cliente Web los organiza y los va mostrando progresivamente al usuario.

Las aplicaciones clientes que resultan ser en estos momentos sumamente avanzadas, y sus funcionalidades no se limitan a la obtención y proceso de montaje de las páginas Web. Son capaces de almacenar en la computadora local las páginas, las imágenes, sonidos, etc.

Esto se une a la existencia en mayor número, de sitios Web para la consulta de servidores de bases de datos y de catálogos en línea de bibliotecas, y esto se convierte en una aplicación universal para la navegación en Internet.

Cabe indicar que se utiliza el browser para mostrar las páginas Web. El protocolo HTTP utiliza la notación especial para localizar y acceder a páginas Web.

### **2.2.5 Modelo TCP/IP (Transport Control Protocol / Internet Protocol).**

El TCP / IP es la base del Internet que sirve para enlazar computadoras que utilizan diferentes sistemas operativos, incluyendo PC, microcomputadoras y computadoras centrales sobre redes de área local y área extensa. TCP / IP fue desarrollado y demostrado por primera vez en 1972 por el departamento de defensa de los Estados Unidos, ejecutándolo en el ARPANET una red de área extensa del departamento de defensa. Las capas de este modelo podemos observarlas en la siguiente tabla 2.2.5.

Aplicación
Transporte
Red
Físico

*Tabla 2.2.5. Capas del modelo TCP/IP*

El segundo modelo mayor de estratificación por capas no se origina de un comité de estándares, sino que proviene de las investigaciones que se realizan respecto

al conjunto de protocolos de TCP/IP. Con un poco de esfuerzo, el modelo ISO puede ampliarse y describir el esquema de estratificación por capas del TCP/IP, pero los presupuestos subyacentes son lo suficientemente distintos para distinguirlos como dos diferentes.

#### **2.2.5.1 Capa de Aplicación.**

Esta capa corresponde a las aplicaciones que están disponibles para los usuarios, como TELNET, FTP, SNMP.

#### **2.2.5.2 Capa de transporte.**

Provee comunicación extremo a extremo desde un programa de aplicación a otro. Puede proveer un transporte confiable asegurándose de que los datos lleguen sin errores y en la secuencia correcta. Coordina a múltiples aplicaciones que se encuentren interactuando con la red simultáneamente de tal manera que los datos que envíe una aplicación sean recibidos correctamente por la aplicación remota.

#### **2.2.5.3 Capa de red.**

Controla la comunicación entre un equipo y otro. Conformar los paquetes IP que serán enviados por la capa inferior. Extrae los paquetes recibidos pasando a la capa superior la información dirigida a una aplicación.

El protocolo IP especifica que la unidad básica de transferencia de datos en el TCP/IP es el paquete.

Los paquetes pueden ser retrasados, perdidos, duplicados, enviados en una secuencia incorrecta o fragmentada intencionalmente para permitir que un nodo con un buffer limitado pueda coger todo el paquete. Es la responsabilidad del protocolo IP reensamblar los fragmentos del paquete en el orden correcto. En algunas situaciones de error los paquetes son descartados sin mostrar ningún mensaje mientras que en otras situaciones los mensajes de error son recibidos por la máquina origen (esto lo hace el protocolo ICMP).

El protocolo IP también define cual será la ruta inicial por la que serán mandados los datos.

#### **2.2.5.4 Capa Física**

Este nivel corresponde al hardware. En este nivel están los protocolos ARP y RARP.

## **2.2.6 SEGURIDAD.**

### **2.2.6.1 Introducción.**

A lo largo del tiempo, las sociedades han temido por la intromisión dentro de lo más importante que como individuos se tiene: la información, la identidad. La información que se concentra en una posición determinada o bien planteada en el resguardo de algún tipo manifiesta posibles agresiones que pueden modificarla y/o destruirla situación que infiere en la disposición de la misma.

La seguridad informática se encuentra condicionada por la elección del sistema operativo y de una versión y configuración estables. Podría definirse como el conjunto de procedimientos y actuaciones destinados al funcionamiento del sistema de información. Su gran avance esta determinando las condiciones y reglamentaciones que se deben llevar en la actualidad, las amenazas crecientes y las catástrofes de índole natural han puesto en funcionamiento una gran cantidad de maquinarias lógicas que son enfocadas en el resguardo de la información en forma remota o en el sitio de emisión de la misma. Dicha maquinaria cada vez más compleja determina una serie de pasos que cada uno y todos los usuarios de una red o bien sistema computacional deben seguir bajo una serie de políticas que tratan de armonizar el mismo convivir de cada usuario.

La determinación de la maquinaria lógica depende ahora de las condiciones que en la red disponemos y de los alcances que cada individuo dentro de la misma va a tener, situación que involucra una serie de usos y de condiciones que se revisarán a lo largo del presente capítulo.

### **2.2.6.2 La seguridad informática en la actualidad.**

Contemplar la situación de la seguridad informática en las épocas recientes nos deja una percepción algo obsoleta y esto es debido a todo lo que los diferentes sistemas en cómputo han ido evolucionando, la aparición de medios de comunicación mas rápidos y la transferencia de datos mas exacta y mas completa

adicionando la concentración de una gama muy importante de datos y temáticas en el vertiginoso desarrollo de la Internet, la implantación del comercio electrónico y el impulso de la denominada "Comunidad Informática", los involucrados con algún sistema de computo comen, viven y respiran con un bombardeo continuo del tema de seguridad en estos ámbitos.

De una infraestructura computacional constituida en sistemas dueños de la información y de grandes servidores manejados por los grandes concedores de los mismos (personal que con una serie de habilidades han hecho parecer que han salido del plano terrenal), y en conjunto se van de alguna forma degradando hasta llegar al individuo o usuario que hará del sistema final un utensilio de uso común y menos complejo. Ello ha supuesto que los niveles iniciales de conocimiento sean rápidamente adquiridos por cualquier persona interesada, sin especiales conocimientos técnicos en la materia. La "globalización" en el conocimiento ha supuesto una quiebra de la seguridad de tiempos pasados amparada, en gran medida, en una cierta ignorancia de la importancia de la información.

Cuando se habla de "Seguridad Informática" se suele pensar en la implantación de una serie de soluciones enfocadas a solventar un problema o situación específica, sin estudiar todos los elementos que puedan estar relacionados, dejando una estela de preocupación por una posible situación peligrosa para el manejo de la información y no como un método de previsión en ese sentido.

### **2.2.6.3 Situaciones que involucran un estado de emergencia dentro de la seguridad informática.**

Cuando empezamos a tratar la información no como una entidad aislada sino como un conjunto de datos con un sentido, un propósito y una disposición, nos topamos con una serie de características que se deben de estar contemplando, las cuales conforman la seguridad de los activos de información dentro de una

organización o empresa las cuales se describen como *confidencialidad, la integridad y la disponibilidad*.

- *Confidencialidad*: Toda la información debe contener un estado de confidencia, en donde el creador de la misma o un grupo selecto de usuarios solo podrán tener acceso a la misma. Tiene relación con la protección de información frente a posibles accesos no autorizados, con independencia del lugar en que reside la información o la forma en que se almacena. La información valiosa dentro de una organización, necesita ser protegida mediante estrictas medidas de control de las cuales podemos disponer de la verificación y la autorización son dos mecanismos que se emplean para asegurar la confidencialidad de la información.
- *Integridad*: Los datos contenidos en un sitio determinado deben estar en condiciones de soportar alteraciones por condiciones no previstas o por intromisiones de naturaleza desconocidas. Se refiere a la protección de información, datos, sistemas contra cambios o alteraciones en su estructura o contenido ya sean intencionadas, no autorizadas o casuales. Los datos no son los únicos que deben estar bajo un control de la integridad, los manejadores de datos, software o procesos que hacen uso de los datos deben estar bajo una supervisión que controle su uso y posibles modificaciones. La información se debe preservar y poner a disposición de los usuarios autorizados de una forma precisa, completa y oportuna.
- *Disponibilidad*: es la garantía de que los usuarios autorizados puedan acceder a la información y recursos cuando los necesiten. La falta de disponibilidad se manifiesta principalmente de dos formas:
  - La denegación, o repudio, del servicio debido a la falta de garantías de la prestación del mismo, tanto por parte del prestador del servicio como del solicitante o tomador (controles de identificación fehaciente, falta de prestaciones de los equipos, congestión de líneas, etc.).

- La pérdida de servicios de los recursos de información por causa de catástrofes naturales o por fallos de equipos, averías, acción de virus, etc.

### **2.2.6.3.1 Amenazas y ataques.**

En términos muy generales podemos determinar 2 situaciones que pueden violar la integridad de la información dentro:

- *Amenazas*: las amenazas se consideran como la intromisión de un ente lógico-computacional cuya misión es observar la mayor cantidad de información para en su momento poder hacer uso de ella en las condiciones que mejor le sean favorables. Se puede decir que es el estado previo antes de cualquier ataque.
- *Ataques*: una vez dentro de las estadias de la información el intruso tiene en sus manos la facultad de poder hacer uso de la misma, manipularla en beneficio de si mismo de un grupo en particular o bien de ser sometida a un deterioro que ponga en peligro su estado de integridad, disponibilidad y confidencialidad.

La información se ve sometida a distintas amenazas que pueden clasificarse en intencionadas, no intencionadas y naturales.

En condiciones de intromisiones de la información las “amenazas intencionadas” las ejercen entes no autorizados que acceden de forma indebida a los datos o información sensible. Los usuarios no autorizados pueden ser externos o pertenecientes a la propia organización y se pueden clasificar como curiosos o maliciosos. Los curiosos normalmente ojean un poco y no siempre entran con pretensiones concretas, ni saben lo que van a encontrar. Los maliciosos entran a los sistemas para apropiarse de datos o información por intereses económicos, o bien con ánimo de dañar o destruir recursos. El acceso no autorizado de usuarios ya sean curiosos o maliciosos significa siempre una violación de la confidencialidad y con frecuencia acarrea violaciones de la integridad y de la disponibilidad.

Por otro lado las “amenazas no intencionadas” provienen típicamente de empleados con poca formación o negligentes que no han seguido los pasos para proteger sus contraseñas, asegurar adecuadamente sus ordenadores o actualizar con la frecuencia debida el programa antivirus. Las amenazas no intencionadas también implican a veces a los programadores o personal de proceso de datos cuando no se siguen las normas y procedimientos de seguridad establecidos, cuando existen. Este entorno operativo es especialmente sensible ya que sencillos errores en un programa pueden afectar a la integridad de la aplicación global y de cualquier otra aplicación con la que comparta información en común.

Aunque no son las mas frecuentes las “amenazas naturales” incluyen fallos de equipos y calamidades tales como incendios, inundaciones y terremotos que pueden causar la pérdida de equipos y datos. Las amenazas naturales suelen afectar a la disponibilidad de los recursos y de la información.

Aunque la información esta bajo amenazas constantes las condiciones de uso han complementado una serie de perspectivas o de políticas de uso que establecen ahora las restricciones y los pasos para la protección de la misma.

#### **2.2.6.3.2 Riesgos.**

Los riesgos asociados a la pérdida de la confidencialidad, integridad o disponibilidad son de diversos tipos y casi siempre implican daños económicos incluyendo responsabilidad contractual, falsos datos financieros, mayores costes, pérdidas de activos, pérdida de negocios, descrédito y pérdida de imagen pública. En la tabla 2.2.6.3.2. se muestra una matriz de amenazas con sus respectivos puntos débiles.



Amenazas	Cualidad	Riesgo
Usuario no autorizado	Confidencialidad	Acceso no autorizado
(curioso)	Integridad	Divulgación no autorizada
Observación no autorizada	Confidencialidad	
Monitorización no autorizada	Confidencialidad	
Copias no autorizadas	Integridad	
Sustracción de información	Integridad	
Usuario no autorizado	Confidencialidad	Acceso indebido, copia o sustracción de datos
(malicioso)	Integridad	Alteración de datos, falsificación, fraude
Destrucción,	Disponibilidad	Pérdida, repudio del servicio
Usuario autorizado	Integridad	Pérdida o alteración de información
Daños.	Disponibilidad	Denegación o alteración de servicio
Catástrofe natural	Disponibilidad	Destrucción, daños, averías, pérdida de servicio

Tabla 2.2.6.3.2. Matriz de Amenazas

### **2.2.6.3.3 Condiciones de salvaguarda de la información, (Políticas de Seguridad).**

Estudios realizados en los últimos años demuestran que un alto porcentaje de organizaciones han experimentado algún tipo de pérdida de información o soporte físico, siendo las causas más frecuentes los errores uso y los virus.

En el uso diario de los sistemas computacionales debiera ser impensable tener PC's o servidores sin programas de protección contra virus. El aumento en el número de delitos informáticos está haciendo replantearse a los profesionales de las tecnologías de la información las medidas de seguridad y mecanismos de control. Los delitos informáticos se han convertido en un mayor riesgo debido por una parte a que hay un mayor número de personas que conocen la informática y tienen acceso a recursos informáticos, y por otra al alto nivel de interconexiones entre redes tanto internas como externas. Pero sigue ocurriendo que la mayoría de los incidentes se originan interiormente a la organización, tanto los no intencionados como los maliciosos.

Las políticas de seguridad no son uniformes para todos los sistemas. El nivel del riesgo debiera determinar el nivel de control adecuado. Cada riesgo se puede tratar mediante la aplicación de uno o varias políticas de seguridad de seguridad. Dichas políticas podrían verse bajo la siguiente clasificación:

- Administrativas
- Físicas
- Técnicas

Las políticas de seguridad administrativas incluyen las políticas y procedimientos de seguridad. Las políticas establecen lo que los usuarios pueden y no pueden hacer al utilizar los recursos informáticos de la organización. También incluyen procedimientos para la renovación de claves de acceso, autorizaciones para

acceder a sus recursos, la revisión y validación periódica de la vigencia del tipo acceso, la asignación de responsabilidades, conocimientos de la seguridad y formación técnica, gestión y supervisión de las tecnologías y soluciones aplicadas, la recuperación tras averías o fallos y la realización y aplicación de planes de contingencia. Los controles administrativos también incluyen la revisión de seguridad e informes de auditoría, que se utilizan para identificar si los usuarios siguen las políticas y procedimientos. Finalmente, los controles administrativos incluyen la asignación de propiedad de los datos y los recursos. Cada persona de la organización debe tener claras sus responsabilidades relativas a la seguridad de cada componente a su cuidado, y estas responsabilidades se deben incluir en los objetivos de cada persona para asegurar que se les valore por la responsabilidad asignada y se les evalúe según el cumplimiento de sus obligaciones.

Las políticas de seguridad físicas limitan el acceso físico directo a los equipos. Las políticas físicas consisten en cerraduras, bloqueos para teclados, vigilantes de seguridad, alarmas y sistemas ambientales para la detección de agua, fuego y humo. También incluyen sistemas de respaldo y alimentación de reserva, tales como baterías y fuentes de alimentación ininterrumpida (UPS).

Las políticas de seguridad técnicas son controles que se implantan a través de soportes físicos o lógicos que típicamente son difíciles de vencer y, una vez implantados, pueden funcionar sin la intervención humana...El soporte lógico específico incluye antivirus, firmas digitales, cifrado, programas de control de biblioteca, herramientas de gestión de red, contraseñas, tarjetas inteligentes, control de acceso de llamadas, sistemas de re-llamada, seguimiento de huellas o trazas de auditoría y sistemas expertos de detección de intrusiones.

Los controles de acceso protegen contra la utilización o manipulación no autorizada de recursos mediante la verificación y la autorización. La verificación asegura la identidad del usuario o sistema que solicita acceso. Los controles de

autorización aseguran que el acceso a la información y los recursos informáticos se limiten de acuerdo con las directrices de la dirección. Otro término relacionado a la seguridad es el no-repudio. El no-repudio previene la posibilidad de que una de las partes de un intercambio o transacción niegue en falso después que haya tenido lugar la misma. El control técnico que se emplea para garantizar el no-repudio es la firma digital. Allí donde antes se empleaba una firma manuscrita para verificar que se había hecho una compra o celebrado un determinado contrato, ahora existe una alternativa digital para verificar que una persona cumplirá con sus obligaciones. La firma digital consiste en una clave privada que sólo conoce o puede duplicar el contenedor de la clave, parecida a su firma y rúbrica manuscrita. Las firmas digitales verifican la fuente, autenticidad e integridad de mensajes electrónicos.

Las políticas de seguridad administrativas, físicas y técnicas se pueden subdividir en preventivas y correctivas. Las preventivas intentan evitar la ocurrencia de acontecimientos indeseados, mientras que las políticas de seguridad correctivas intentan identificar los incidentes y reducir sus efectos después de que hayan sucedido. Es siempre preferible evitar un incidente de seguridad en lugar que tener que hacerle frente a posteriori. Las políticas de seguridad correctivas se han diseñado para identificar el problema con suficiente rapidez para reducir los daños al mínimo y para poder valorar con precisión la magnitud de los daños causados por el incidente. En la tabla 2.2.6.3.3. podemos visualizar los controles administrativos, físicos y técnicos específicos, y una forma de considerarlos dentro clasifica además en mecanismos preventivos o correctivos

<b>Preventivos Correctivos</b>
<b>Controles Físicos</b>
<ul style="list-style-type: none"><li>• Archivos y documentación de reserva Detectores de movimiento</li><li>• Verjas Detectores de humo y fuego</li><li>• Guardias de seguridad Monitorización por televisión de circuito cerrado</li><li>• Sistemas de tarjetas de identificación Sensores y alarmas</li><li>• Cerraduras y llaves</li><li>• Candados cifrados</li><li>• Potencia de reserva</li><li>• Controles biométricos de acceso</li><li>• Selección de emplazamiento</li><li>• Extintores de incendios</li><li>• Bloqueo de teclados</li></ul>
<b>Controles Administrativos</b>
<ul style="list-style-type: none"><li>• Conocimientos de seguridad y formación técnica</li><li>• Revisiones y auditorias de seguridad</li><li>• Separación de obligaciones Control de calidad</li><li>• Procedimientos sancionadores Investigaciones de antecedentes</li><li>• Políticas y procedimientos de seguridad Rotación de responsabilidades</li><li>• Gestión y supervisión</li><li>• Recuperación de averías y planes de contingencia</li><li>• Administración de accesos de usuarios</li><li>• Gestión de propietarios de datos y recursos</li></ul>
<b>Controles Técnicos</b>

- Programas de control de acceso y trazas para auditoria.
- Programas *firewall* (cortafuegos) y antivirus Sistemas expertos de detección de intromisiones.
- Gestión de contraseñas.
- Tarjetas inteligentes.
- Cifrado.

Tabla 2.2.6.3.3 Controles de Seguridad de Información

#### **2.2.6.4 Criptología informática (documentos en resguardo).**

##### **2.2.6.4.1 La definición.**

A lo largo del tiempo se han desarrollado varios formatos y técnicas para preservar ocultar o bien mejorar la confidencialidad de la información. Estas técnicas que tratan de mantener confidencial la información entran en la Criptología. Aparece la necesidad de transmitir y almacenar información de forma tal que se pueda preservar su confidencialidad. Esta necesidad es el motor de desarrollo de la Criptología.

*Definición:* “La criptografía es la técnica, ciencia o arte de la escritura secreta”

Se entiende por criptología el método sistemático en el que los procesos de cifrado están enfocados a ocultar el contenido de mensajes enviados entre dos partes: emisor y receptor.

El militar Kerckhoffs en su libro “La criptografía militar” describe las reglas que debe cumplir un buen sistema criptográfico.

- “No debe existir ninguna forma de recuperar mediante el criptograma el texto inicial o la clave. Esta regla se considera cumplida siempre que la complejidad del proceso de recuperación del texto original sea suficiente para mantener la seguridad del sistema.

- Todo sistema criptográfico debe estar compuesto por dos tipos distintos de información.
  - Pública: como es la familia de algoritmos que lo definen.
  - Privada: como es la clave que se usa en cada cifrado particular. En los sistemas de clave pública, parte de la clave es también información pública.
- La forma de escoger la clave debe ser fácil de recordar y modificar.
- Debe ser factible la comunicación del criptograma por los medios de transmisión habituales.
- La complejidad del proceso de recuperación del texto original debe corresponderse con el beneficio obtenido. ”

Aunque esto es el precedente de todo buen sistema de cifrado, no son las únicas reglas a seguir, es de ser considerado además las condiciones en las cuales los sistemas criptológicos son aplicados y el grado de “profundidad” (numero de procesos criptológicos implicados) que se desea aplicar.

#### **2.2.6.2.2 La historia**

Aunque la historia marca los primeros antecedentes de la Criptología en la antigua civilización egipcia, las aplicaciones de más profundas son presentadas en la primera década del siglo XX.

Durante las guerras mundiales se crearon los primeros sistemas criptológicos modernos detonándose la aparición de los métodos de transposición en los que se consideraban bloques de un tamaño determinado para su encriptación, apareciendo el sistema ADFGX en las filas alemanas con una transposición en bloques de longitud 20. En el periodo entreguerras surgen la época de pánico en las relaciones internacionales y la importancia de las alianzas y la confidencialidad de dichos pactos se generaliza y se impone una serie de reglas no escritas en las que la información y su transmisión debe ser lo mas segura posible.

La primera patente registrada de una máquina encriptadora surge en Europa por parte de un Alexander Koch y Arthur Scherbius, este último de origen alemán quien más tarde en los laboratorios de la Alemania Nazi dio vida a la máquina Enigma que durante algún tiempo fue considerada como inviolable, aunque después del 42 provocaría un sin número de derrotas que provocaron la pérdida de Atlántico.

Mientras que en el Atlántico aparecía la Enigma, en el Pacífico aparecía el Código Púrpura, la versión japonesa de la Enigma. Turing había logrado descifrar la Enigma, el Comandante J.J. Rochefort junto a un grupo de altos científicos describió los secretos del Código Púrpura logrando la más sonada victoria en el atolón de Midway. Los americanos no se quedaron con las ganas y en su búnker un modelo llamado Sigaba (figura 2.2.6.4.2) apodada por los alemanes como “la gran máquina” la cual se convirtió en la única que no reveló sus secretos durante su vida útil.

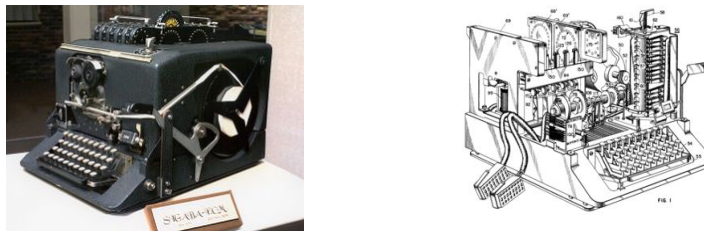


Figura 2.2.6.4.2 Sigaba “La gran Máquina”

Para los años 70's, la combinación de los intelectos de Diffie y Hellman establecieron las bases de los algoritmos de la *llave Pública*, la cual no se podía imaginar hasta ese momento debido a que toda la teoría estaba basada en una “palabra secreta”.

En 1976 aparece el código DES siendo favorecido por el desarrollo de las computadoras digitales. Fue el primero en ser un sistema de forma compleja, pero la existencia de la clave secreta, que debía ser muy guardada si se deseaba mantener la robustez del sistema dejaba un hueco en la seguridad. Diffie y Hellman lograron cubrir ese hueco con su teoría de la llave pública.



Un año después Rivest, Shamir y Adleman diseñaron un sistema basado en buscar números primos, un reto casi sin poder ser cumplido, y más utilizado hasta la fecha el cual llamaron RSA y para el año de 1991 aparece el sistema PGP de Phil Zimmerman que más que un sistema criptográfico es un software que reúne una serie de métodos como el DES.

En la tabla 2.2.6.4.2. muestra una clasificación general de los métodos de encriptación que a lo largo de la historia se han podido ir conociendo.

<b>Cifrado de Sustitución</b>				
<b>Sustitución monoalfabética</b>	<b>Sustituciones polialfabeticas</b>	<b>Recifrado o supercifrado</b>	<b>Cifrado de Transposición</b>	<b>Cifrado sustitución-transposición</b>
<ul style="list-style-type: none"> <li>• Cifras hebreas</li> <li>• Código Polibio</li> <li>• Cifrado César</li> <li>• Cifrado Kama-Sutra</li> <li>• Cifrado de templarios</li> <li>• Cifrado Pig Pen</li> <li>• Cifrado Babou</li> <li>• Cifrado de bacon</li> </ul>	<ul style="list-style-type: none"> <li>• Disco de Alberti</li> <li>• Tabla recta de Trithemious</li> <li>• Bellaso y la palabra clave</li> <li>• Cifrado de Della porta</li> <li>• Cifrado Vigenére</li> <li>• Cifrado de Beaufort</li> <li>• Cilindro de Jefferson</li> <li>• Cifrado Playfair</li> </ul>	<ul style="list-style-type: none"> <li>• Cifrado de Bazeris</li> </ul>	<ul style="list-style-type: none"> <li>• Baston de Licurgo (Skytale)</li> <li>• Formula de Sator o cuadrado latino</li> <li>• Trasposiciones geométricas</li> <li>• Rail fence</li> <li>• Reja giratoria de Flissner</li> <li>• Cifrado Nihlist</li> <li>• Cifrado de Ubchi</li> </ul>	<ul style="list-style-type: none"> <li>• Cifrado ADFGVX</li> </ul>
<b>Sistemas de Códigos</b>	<b>Esteganografía</b>	<b>Cifrado homofónico</b>	<b>Criptografía Moderna</b>	<b>Máquinas</b>
<ul style="list-style-type: none"> <li>• Código Braille</li> <li>• Código Morse</li> <li>• Código Navajo</li> <li>• Código ISBN</li> </ul>	<ul style="list-style-type: none"> <li>• Ave Maria de Trithemius</li> <li>• Parilla de cardano</li> <li>• Cifrado de Bacon</li> <li>• Seguridad monetaria Suiza</li> <li>• Semografía</li> <li>• El código de la Biblia</li> <li>• Cartas de George Sand</li> </ul>	<ul style="list-style-type: none"> <li>• El disco del ejercito Mexicano</li> <li>• Cifrado de inversión de frecuencias</li> <li>• Sistema diccionario</li> <li>• Criptograma del asesino del Zodiaco</li> <li>• Multiple E</li> </ul>	<ul style="list-style-type: none"> <li>• Cifrado de bloque ECB y CBC</li> <li>• Algoritmo de Deffi-Hellman</li> <li>• DES</li> <li>• Algoritmo de llave publica</li> <li>• RSA</li> <li>• AES</li> <li>• Criptografía cuántica.</li> </ul>	<ul style="list-style-type: none"> <li>• La bomba de Turing</li> <li>• La máquina Enigma</li> <li>• El secreto de Bletchley Park</li> <li>• Sigaba</li> </ul>

Tabla 2.2.6.4.2 Clasificación de los métodos de la encriptación

### **2.2.6.5 Encriptación de documento y su transmisión.**

En el concepto de criptografía tradicional, o de "clave secreta" tanto el emisor como el receptor poseen una misma clave o password. El emisor utiliza esta clave para cifrar el mensaje obteniéndose el "mensaje cifrado", que es ilegible. El receptor utiliza la misma clave utilizada para el cifrado para descifrar el mensaje y obtener así el mensaje original. Si esta clave es conocida únicamente por emisor y receptor, se asegura que el mensaje recibido es el original y que es enviado por la única persona (además del receptor), que tiene la clave.

Aunque el sistema simple con clave conocida por emisor y receptor son de condiciones individuales (solo conocida por ellos) no los exime de que dicha clave pueda quedar o ser repetida por algún usuario distinto, o bien ser interceptado el mensaje adicionando la violentación de la contraseña lo cual implicaría que dicho mensaje podría ser leído ya sea por coincidencia de la contraseña o bien por ser violada la privacidad. De cualquier modo estos métodos tradicionales no funcionaron bajo el criterio anterior y tuvieron que abrir paso a métodos con mayor consideración de seguridad.

Y con este punto cabe aclarar, criptología no implica seguridad total, hay que considerar que la criptología como tal es solo un instrumento para que una serie de consideraciones de seguridad (políticas) lleven a cabo su función como tal.

### **2.2.6.6 Diseño de Contraseñas Seguras**

Uno de los métodos más comunes y certeros para controlar el acceso a información en los sistemas es usar contraseñas. Es de suma importancia fortalecer las contraseñas y mantenerlas siempre como algo verdaderamente confidencial. Uno de los errores más comunes es elaborar una clave que sea fácil de recordar, precisamente con la intención de no olvidarla, pero no es recomendable diseñarlas en base a datos personales como teléfono, número de

cartilla, fecha de cumpleaños, etc. Palabras de diccionario en ningún idioma, los nombres de mascotas, aficiones o gustos.

Si una contraseña violada, pierde su contundencia para proteger la información que se quiere mantener de manera confidencial. No existe una contraseña ideal pero se puede considerar una que sea para el usuario fácil de recordar, pero que nadie más pueda dar con ella.

No sirve de mucho implementar reglas si no se coloca énfasis en la conservación de la confidencialidad. Es necesario considerar:

- No compartir contraseñas, implementando sistemas aleatorios de tiempo para que sean cambiadas.
- No repetir las contraseñas en diferentes sistemas.
- No considerar el almacenaje de las mismas en la computadora.
- No considerar enviarlas por correo electrónico ni por mensajero instantáneo.
- En caso de que tenga indicios de que alguien más ya las conoce, haga aplicar un reemplazo inmediato de la contraseña.

## **2.3 Ingeniería del software.**

### **2.3.1 Ciclo de Vida del Software.**

Un modelo de ciclo de vida define el estado de las fases a través de las cuales se mueve un proyecto de desarrollo de software.

El primer ciclo de vida del software, "Cascada", fue definido por Winston Royce a fines del 70. Desde entonces muchos equipos de desarrollo han seguido este modelo. Sin embargo, ya desde 10 a 15 años atrás, el modelo cascada ha sido sujeto a numerosas críticas, debido a que es restrictivo y rígido, lo cual dificulta el desarrollo de proyectos de software moderno. En su lugar, muchos modelos nuevos de ciclo de vida han sido propuestos, incluyendo modelos que pretenden desarrollar software más rápidamente, o más incrementalmente o de una forma más evolutiva, o precediendo el desarrollo a escala total con algún conjunto de prototipos rápidos.

#### **2.3.1.1 Definición de un Modelo de Ciclo de Vida**

Un modelo de ciclo de vida de software es una vista de las actividades que ocurren durante el desarrollo de software, intenta determinar el orden de las etapas involucradas y los criterios de transición asociadas entre estas etapas.

Un modelo de ciclo de vida del software:

Describe las fases principales de desarrollo de software.

- Define las fases primarias esperadas de ser ejecutadas durante esas fases.
- Ayuda a administrar el progreso del desarrollo.
- Provee un espacio de trabajo para la definición de un detallado proceso de desarrollo de software.

Así, los modelos por una parte suministran una guía para los ingenieros de software con el fin de ordenar las diversas actividades técnicas en el proyecto, por otra parte suministran un marco para la administración del desarrollo y el

mantenimiento, en el sentido en que permiten estimar recursos, definir puntos de control intermedios, monitorear el avance, etc.

## **2.3.2 Alternativas de Modelos de Ciclo de Vida.**

### **2.3.2.1 Modelo Cascada**

Este es el más básico de todos los modelos, y sirve como bloque de construcción para los demás modelos de ciclo de vida. La visión del modelo cascada del desarrollo de software es muy simple; dice que el desarrollo de software puede ser a través de una secuencia simple de fases. Cada fase tiene un conjunto de metas bien definidas, y las actividades dentro de una fase contribuye a la satisfacción de metas de esa fase o quizás a una subsecuencia de metas de la fase. Las flechas muestran el flujo de información entre las fases. La flecha de avance muestra el flujo normal. Las flechas hacia atrás representan la retroalimentación.

El modelo de ciclo de vida cascada, captura algunos principios básicos:

- Planear un proyecto antes de embarcarse en él.
- Definir el comportamiento externo deseado del sistema antes de diseñar su arquitectura interna.
- Documentar los resultados de cada actividad.
- Diseñar un sistema antes de codificarlo.
- Testear un sistema después de construirlo.

### **2.3.2.2 Modelo de Desarrollo Incremental**

Los riesgos asociados con el desarrollo de sistemas largos y complejos son enormes. Una forma de reducir los riesgos es construir sólo una parte del sistema, reservando otros aspectos para niveles posteriores. El desarrollo incremental es el proceso de construcción siempre incrementando subconjuntos de requerimientos del sistema. Típicamente, un documento de requerimientos es escrito al capturar todos los requerimientos para el sistema completo.

El modelo de desarrollo incremental provee algunos beneficios significativos para los proyectos:

- Construir un sistema pequeño es siempre menos riesgoso que construir un sistema grande.
- Al ir desarrollando parte de las funcionalidades, es más fácil determinar si los requerimientos planeados para los niveles subsiguientes son correctos.
- Si un error importante es realizado, sólo la última iteración necesita ser descartada.
- Reduciendo el tiempo de desarrollo de un sistema (en este caso en incremento del sistema) decrecen las probabilidades que esos requerimientos de usuarios puedan cambiar durante el desarrollo.
- Si un error importante es realizado, el incremento previo puede ser usado.

Los errores de desarrollo realizados en un incremento, pueden ser arreglados antes del comienzo del próximo incremento.

### **2.3.2.3 Modelo De Desarrollo Evolutivo**

El modelo de desarrollo evolutivo (algunas veces denominado como prototipo evolutivo) construye una serie de grandes versiones sucesivas de un producto. El modelo evolutivo asume que los requerimientos no son completamente conocidos al inicio del proyecto.

El modelo evolutivo, los requerimientos son cuidadosamente examinados, y sólo esos que son bien comprendidos son seleccionados para el primer incremento. Los desarrolladores construyen una implementación parcial del sistema que recibe sólo estos requerimientos.

El sistema es entonces desarrollado, los usuarios lo usan, y proveen retroalimentación a los desarrolladores.

Basada en esta retroalimentación, la especificación de requerimientos es actualizada, y una segunda versión del producto es desarrollada y desplegada. El proceso se repite indefinidamente.

Todo lo que uno tiene que hacer es construir un subconjunto de requerimientos conocidos (incremental), y comprender al principio que muchos nuevos requerimientos es probable que aparezcan cuando el sistema sea desplegado o desarrollado.

El desarrollo de software en forma evolutiva requiere un especial cuidado en la manipulación de documentos, programas, datos de test, etc. desarrollados para distintas versiones del software. Cada paso debe ser registrado, la documentación debe ser recuperada con facilidad, los cambios deben ser efectuados de una manera controlada.

#### **2.3.2.4 Modelo de Prototipo de Requerimientos.**

El prototipo de requerimientos es la creación de una implementación parcial de un sistema, para el propósito explícito de aprender sobre los requerimientos del sistema. Un prototipo es construido de una manera rápida tal como sea posible. Esto es dado a los usuarios, clientes o representantes de ellos, posibilitando que ellos experimenten con el prototipo. Estos individuos luego proveen la retroalimentación sobre lo que a ellos les gustó y no les gustó acerca del prototipo proporcionado, quienes capturan en la documentación actual de la especificación de requerimientos la información entregada por los usuarios para el desarrollo del sistema real. El prototipo puede ser usado como parte de la fase de requerimientos (determinar requerimientos) o justo antes de la fase de requerimientos (como predecesor de requerimientos).

El Prototipo ha sido usado frecuentemente en los 90, porque la especificación de requerimientos para sistemas complejos tiende a ser relativamente dificultoso de cursar. Muchos usuarios y clientes encuentran que es mucho más fácil proveer



retroalimentación convenientemente basado en la manipulación, leer una especificación de requerimientos potencialmente ambigua y extensa.

### **2.3.2.5 Modelo Espiral**

El modelo espiral de los procesos software es un modelo del ciclo de *meta-vida*. En este modelo, el esfuerzo de desarrollo es iterativo. Tan pronto como uno completa un esfuerzo de desarrollo, otro comienza. Además, en cada desarrollo ejecutado, puedes seguir estos cuatro pasos:

- Determinar qué quieres lograr.
- Determinar las rutas alternativas que puedes tomar para lograr estas metas. Por cada una, analizar los riesgos y resultados finales, y seleccionar la mejor.
- Seguir la alternativa seleccionada en el paso 2.
- Establecer qué tienes terminado.

### **2.3.2.6 Modelo Concurrente**

El modelo concurrente provee una meta-descripción del proceso software.

La contribución del modelo concurrente es su capacidad de describir las múltiples actividades del software ocurriendo simultáneamente.

Esto no sorprende a nadie que ha estado involucrado con las diversas actividades que ocurren en algún tiempo del proceso de desarrollo de software.

Los requerimientos son usualmente "líneas de base", cuando una mayoría de los requerimientos comienzan a ser bien entendidos, en este tiempo se dedica un esfuerzo considerable al diseño. Sin embargo, una vez que comienza el diseño, cambios a los requerimientos son comunes y frecuentes (después de todo, los problemas reales cambian, y el entendimiento de los problemas desarrollados también). No se aconseja detener el diseño en este camino cuando los

requerimientos cambian; en su lugar, existe una necesidad de modificar y rehacer líneas de base de los requerimientos mientras progresa el diseño. Por supuesto, dependiendo del impacto de los cambios de los requerimientos el diseño puede no ser afectado, medianamente afectado o se requerirá comenzar todo de nuevo.

Durante el diseño de arquitectura, es posible que algunos componentes comiencen a ser bien definidos antes que la arquitectura completa sea estabilizada. En tales casos, puede ser posible comenzar el diseño detallado en esos componentes estables. Similarmente, durante el diseño detallado, puede ser posible proceder con la codificación y quizás regular testeando en forma unitaria o realizando testeos de integración previo a llevar a cabo el diseño detallado de todos los componentes.

En algunos proyectos, múltiples etapas de un producto se han desarrollado concurrentemente. Por ejemplo, no es inusual estar haciendo mantenimiento de la etapa 1 de un producto, y al mismo tiempo estar haciendo mantenimiento sobre un componente 2, mientras que se está haciendo codificación sobre un componente 3, mientras se realiza diseño sobre una etapa 4, y especificación de requisitos sobre un componente 5.

En todos estos casos, diversas actividades están ocurriendo simultáneamente. Eligiendo seguir un proyecto usando técnicas de modelación concurrente, se posibilita el conocimiento del estado verdadero en el que se encuentra el proyecto.

### **2.3.2.7 Modelo RUP**

RUP es uno de los procesos más utilizados actualmente, ya que en realidad esta pensado para adaptarse a cualquier proyecto, y no tan solo de software.

Un proyecto realizado siguiendo RUP se divide en cuatro fases:

1. Intercepción (puesta en marcha)

2. Elaboración (definición, análisis, diseño)
3. Construcción (implementación)
4. Transición (fin del proyecto y puesta en producción)

En cada fase se ejecutarán una o varias iteraciones (de tamaño variable según el proyecto), y dentro de cada una de ellas seguirá un modelo de cascada o *waterfal* para los flujos de trabajo que requieren las nueva actividades anteriormente citadas.

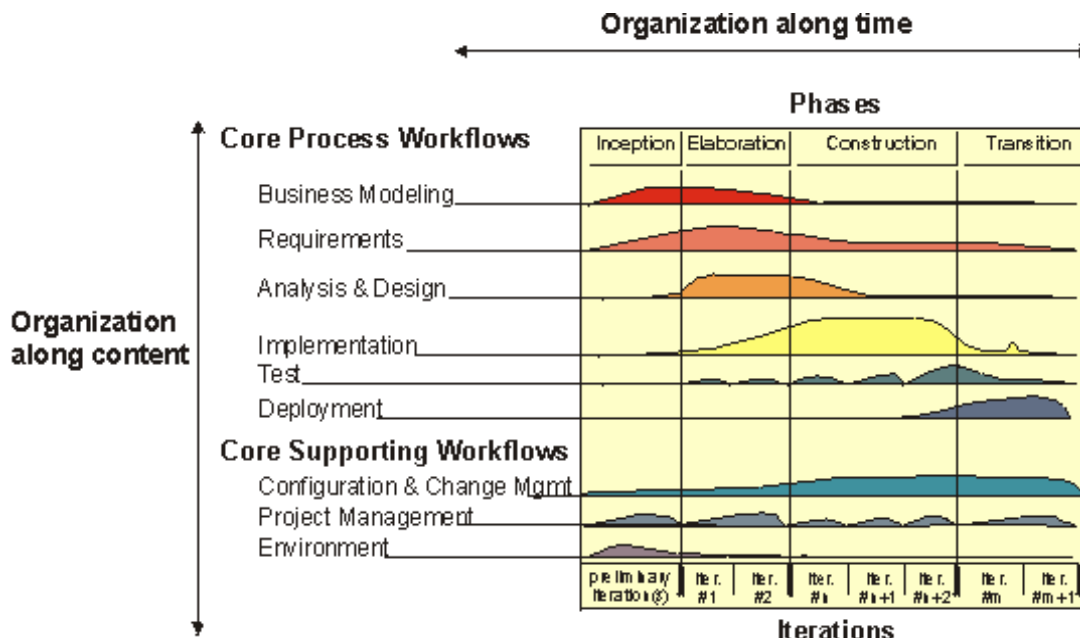


Figura 2.7.1.1. Actividades de RUP

RUP define nueve actividades a realizar en cada fase del proyecto como se muestra en la figura 2.7.1.1. 2.7.1.2.

1. Modelado del negocio
2. Análisis de requisitos
3. Análisis y diseño
4. Implementación
5. Test
6. Distribución
7. Gestión de configuración y cambios

8. Gestión del proyecto

9. Gestión del entorno

El flujo de trabajo (*workflow*) entre ellas en base a los llamados diagramas de actividad.

El proceso define una serie de roles que se distribuyen entre los miembros del proyecto y que definen las tareas de cada uno y el resultado (*artefactos* en la jerga de RUP) que se espera de ellos.



Figura 2.7.1.2. Procesos de RUP

RUP se basa en casos de uso para describir lo que se espera del software y esta muy orientado a la arquitectura del sistema, documentándose lo mejor posible, basándose en UML (*Unified Modeling Language*) como herramienta principal.

RUP es un proceso muy general y muy grande, por lo que antes de usarlo habrá que adaptarlo a las características de la empresa. Por suerte ya hay muchos procesos descritos en internet que son versiones reducidas del RUP.

### 2.3.2.8 Modelo XP

Este modelo intenta por medio de un trabajo orientado directamente al objetivo, basado en las relaciones interpersonales y la velocidad de reacción.

XP intenta minimizar el riesgo de fallo del proceso por medio de la disposición permanente de un representante *competente* del cliente a disposición del equipo de desarrollo. Este representante debería estar en condiciones de contestar rápida y correctamente a cualquier pregunta del equipo de desarrollo de forma que no se retrase la tomar de decisiones, de ahí lo de *competente*.

XP define *UserStories* como base del software a desarrollar. Estas historias las escribe el cliente y describen escenarios sobre el funcionamiento del software, que no solo se limitan a la *GUI* si no también pueden describir el modelo, dominio, etc. A partir de las *UserStories* y de la arquitectura perseguida se crea un plan de *releases* entre el equipo de desarrollo y el cliente.

Para cada release se discutirán los objetivos de la misma con el representante del cliente y se definirán las iteraciones (de pocas semanas de duración) necesarias para cumplir con los objetivos de la release. El resultado de cada iteración es un programa que se transmite al cliente para que lo juzgue. En base a su opinión se definen las siguientes iteraciones del proyecto y si el cliente no está contento se adaptará el plan de releases e iteraciones hasta que el cliente de su aprobación y el software este a su gusto.

Junto a los *UserStories* están los escenarios de pruebas que describen el escenario contra el que se comprueba la realización de las *UserStories*. *UserStories* y casos de pruebas son la base sobre la que se asienta el trabajo del desarrollador.

Como primer paso de cada iteración se escribirán las pruebas, de tal forma que puedan ser ejecutadas automáticamente, de manera que pueda comprobarse la corrección del software antes de cada release. Esto es de vital importancia en XP debido a su apuesta por las iteraciones cortas que generan software que el cliente puede ver y por la refactorización para mejorar el código constantemente, que hacen más que deseable una cantidad considerable de test lo más

automatizables posible. Así pues, la funcionalidad concreta del software solo se escribe cuando las pruebas para su corrección estén preparadas. En la figura 2.3.2.8. podemos observar el diagrama de XP

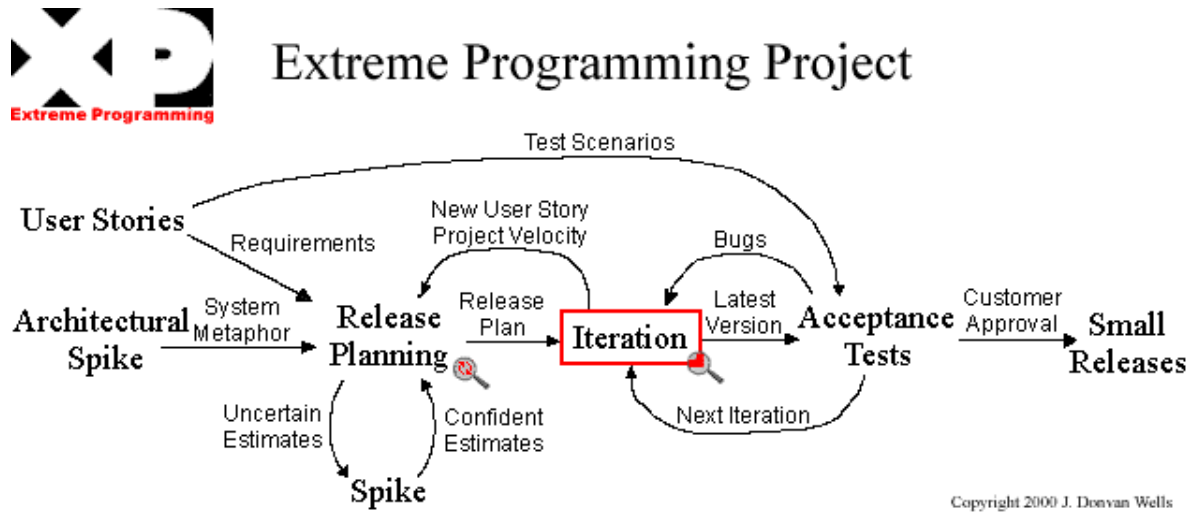


Figura 2.3.8.2 Diagrama de XP

# Capítulo III.

## Análisis.

- 3.1** Situación actual.
- 3.2** Requerimientos del usuario.
  - 3.2.1** Módulo para la carga de información en la Base de Datos.
  - 3.2.2** Módulo para la toma de pedido.
  - 3.2.3** Módulo de consultas de remisiones pendientes.
  - 3.2.4** Módulo para el registro de las recepciones.
  - 3.2.5** Módulo para el registro de transferencias.
  - 3.2.6** Módulo para la captura del inventario semanal.
- 3.3** Presupuestos.
  - 3.3.1** Hardware.
  - 3.3.2** Software.
- 3.4** Análisis de los riesgos de seguridad.
  - 3.4.1** Riesgo de Paso.
    - 3.4.1.1** Acceso libre de usuario.
    - 3.4.1.2** Omisión de resguardo de usuario y contraseña.
    - 3.4.1.3** Omisión de encriptación de paso (contraseñas simples).
  - 3.4.2** Riesgos de Uso.
    - 3.4.2.1** Solicitudes sin autorización.
    - 3.4.2.2** Observadores sin autorización.
    - 3.4.2.3** Usuarios sin autorización.
  - 3.4.3** Violación de la integridad.
    - 3.4.3.1** Ataques a las instalaciones lógicas.
  - 3.4.4** Políticas de seguridad Básicas.
    - 3.4.4.1** Políticas de uso.
    - 3.4.4.2** Políticas de acceso.
    - 3.4.4.3** Políticas de respaldo.

### 3.1 Situación Actual

Actualmente los restaurantes siguen una serie de pasos fijos y consecutivos para realizar el pedido de materia prima y utensilios. Dichos pasos tienen como objetivo hacer esta tarea metódicamente y se pretende que con la experiencia la persona que la realiza lo haga de manera eficiente. A continuación se presenta un resumen del proceso a seguir por una persona para generar un pedido y señalaremos los puntos perfectibles en el proceso actual. De aquí en adelante se entenderá como encargado a la persona interactúa con el sistema.

Como primer paso el encargado entra al SAI y obtiene de él la lista de productos a incluirse en el pedido, esta lista tiene los productos ordenados por familias y algunos de estos productos tienen una cantidad sugerida que es manejada en unidades de inventario. El número de productos incluidos en la lista pueden variar, pero el encargado prefiere imprimir la lista para utilizarla en el resto del proceso ya que ocupa hasta 21 cuartillas. La figura 3.1 muestra un segmento de lo que imprime el encargado para obtener el pedido sugerido:

Nombre_de Producto	Costo Medida	Cantidad Disponid	Pronost Requerimient	Sugerido Pedid	Costo Unit.
3020/22POLLO	CBZA	57.15	2392.77	2336.00	19.30
3054TIRAS EMPAN	PZA	2554.00	4202.57	1649.00	2.15
3453COR.SUPREME	PZA	447.00	541.54	95.00	6.46
4032 BIGCRUNCH	PZA	622.00	1189.90	568.00	5.38
4036POPCORN	CHI BLSA	24.06	48.96	25.00	87.29
4058/59MINICRUN	PZA	564.00	600.44	37.00	2.76
5645QUEECRE	CHIP PZA	0.00	0.00	0.00	2.02
3221SAZONADO	TS PQTE	50.47	16.97	0.00	54.50
3230 HARINA	T.S SACO	18.83	38.93	21.00	41.87
3239SAL	TS TMOR PQTE	32.83	12.98	0.00	7.26
3251MEZCLA	LECH PQTE	57.37	14.97	0.00	9.81
3830 SAZONADOR	PQTE	7.09	24.96	18.00	13.56
3962MARINADR	TS PQTE	205.98	67.88	0.00	4.84
3964 MARINADOR	PQTE	11.56	28.33	17.00	5.94
5310PRE.HUTCHEE	SACO	0.00	0.00	0.00	26.64

Figura 3.1 Pantalla de pedido sugerido



La siguiente tarea consiste en seleccionar los productos de la lista que realmente se requieran. Una vez seleccionados los productos y auxiliándose de un señalador, el restaurante toma el número de unidades de inventario<sup>1</sup> (columna Sugerido Pedido de la figura 3.1) que se necesitan según el sistema, enseguida convierte estas unidades a las unidades de proveedor<sup>2</sup> mediante un número denominado factor de equivalencia<sup>3</sup> y anota el resultado de la operación al lado del producto. Si el restaurante además tiene dos pedidos a la semana, debe de restar el producto que tiene en tránsito<sup>4</sup> para obtener la cantidad sugerida que realmente requiere.

Un ejemplo de esto sería el siguiente:

El sistema sugiere que se soliciten 850 vasos de unigel de 32 OZ (unidad de inventario), cada caja (unidad de proveedor) contiene 1000 vasos (factor de equivalencia).

El encargado necesita entonces hacer la división de  $850/1000 = 0.85$  de caja, pero como no se puede solicitar incompleta la unidad de proveedor, el encargado deberá solicitar en su pedido 1000 vasos que son iguales a una caja.

Generalizando la fórmula que se tiene que aplicar para hacer los cálculos de conversión entre unidades de inventario y proveedor tenemos:

$$\text{Unidad proveedor} = \text{Unidad inventario} / \text{Factor de equivalencia}$$

El punto de las conversiones es muy delicado en el proceso, ya que se deja toda la responsabilidad al encargado y éste se debe saber de memoria los factores de equivalencia ó debe estarlo verificando en los empaques que entregue el proveedor para hacerlo de manera adecuada.

---

<sup>1</sup> Unidades de inventario se refieren a las unidades que ocupa el restaurante para llevar a cabo el control de sus productos.

<sup>2</sup> Unidades de proveedor se refieren a las unidades que ocupa el proveedor para surtir un producto.

<sup>3</sup> Factor de equivalencia es aquel que se ocupa para hacer la conversión entre unidades de inventario y unidades de proveedor.

<sup>4</sup> Producto en tránsito se refiere a producto que ya se ordenó anteriormente, pero aún no ha sido recibido cuando ya se tiene que volver a ordenar. Este producto generalmente se tiene en restaurantes que tienen 2 pedidos a la semana.

El proceso de conversión, como ya se mencionó se lleva a cabo para cada uno de los productos por lo que es obvio que resulta un proceso agobiante al cual hay que dedicarle mucho tiempo y que genera muchos errores por depender del encargado exclusivamente.

Es necesario aclarar que muchas veces el proceso de la conversión es sustituido por la experiencia que tenga el encargado sobre el pedido que debe realizar para cada ocasión, lo que obviamente lo hace mucho más eficiente, no siendo así para aquella persona que aún no cuenta con la suficiente experiencia y toma como base los valores proporcionados por el sistema.

El siguiente paso consiste en realizar el pedido propiamente, para esto es necesario aclarar que uno de los proveedores (al cual llamaremos principal) surte entre el 85 y 90 por ciento de los insumos que requiere el restaurante, el resto de los proveedores surten el porcentaje restante. Esto se menciona porque el proveedor principal tiene la infraestructura necesaria para realizar el pedido a través de una aplicación suya dentro del sistema de cada restaurante y recibir los pedidos por medio del proceso de poleo<sup>5</sup>. El pedido a los demás proveedores se hace telefónicamente.

Cada restaurante tiene instalado el programa del proveedor principal dentro de la computadora gerencial, de manera que se accede a ella a través de un icono en el escritorio.

La aplicación es tipo texto y tiene una serie de opciones en las cuales no se profundizará demasiado, excepto en la que es relevante para el nuevo sistema que se planea desarrollar *Orders* (Ordenes), por medio de ésta opción el encargado transcribe todo el pedido que tiene anotado en papel, ya con las respectivas operaciones de conversión para que en el campo *Order Qty* (Cantidad a Ordenar) como se puede apreciar en la figura 3.2.

---

<sup>5</sup> Poleo es el proceso de intercambio de información entre el restaurante y las oficinas administrativas y para este caso específico entre las oficinas administrativas y el proveedor principal.

Item#	Description	Unit of Measure	Last Order	Order Qty	Est. Amount	Days Since Last Order
89037	JABON BACTERICIDA KA	CJ-1CJ	-	1	341.78	-
89040	JALADOR DE HULE	PZ-1PZ	-	2	221.58	-
89048	PAPEL ECONOMICO JUMB	CJ-1CJ	-	1	179.03	-
89049	DESPACHADOR DE SERVI	PZ-1PZ	-	3	21.33	-
89053	RECOGEDOR DE LAMINA	PZ-1PZ	-	1	177.86	-
89055	F-2706 ROCIADOR P/CR	PZ-1PZ	-	5	40.10	-
89064	BOTE PARA BASURA GRA	PZ-1PZ	-	1	215.75	-
89066	TOALLA INTERDOBLADA	CJ-1CJ	-	1	127.88	-

[F11]=Help      [F31]=Sug Order Calc      [F61]=Crit. Items      [F9]=Done  
 [F21]=Order History      [F5]=Copy Suggested      [F81]=Add/Search      [Esc]=Cancel

Figura 3.2 Aplicación del proveedor principal para realizar el pedido

Ya finalizado el proceso únicamente se pulsa F9 y con esto, el encargado confirma el envío de su orden. Aunque en realidad se genera un archivo para ser poleado a oficinas.

Ahora al encargado sólo le resta esperar al día en que el proveedor principal surta esa orden, los cual será de 3 ó 4 días después.

Ya cuando el proveedor surte el producto, le entrega al encargado la hoja de la remisión, en donde están escritos los productos que se están entregando, por lo que ahora tiene que ingresarlo a su sistema a través del SAI, en la opción de "Inventario y PCA" → "Recepciones de Proveedores" ó eligiendo las opciones de "Inventario y PCA" → Inventario de Entradas y Salidas. La figura 3.3 muestra la primera opción y la figura 3.4 la segunda.

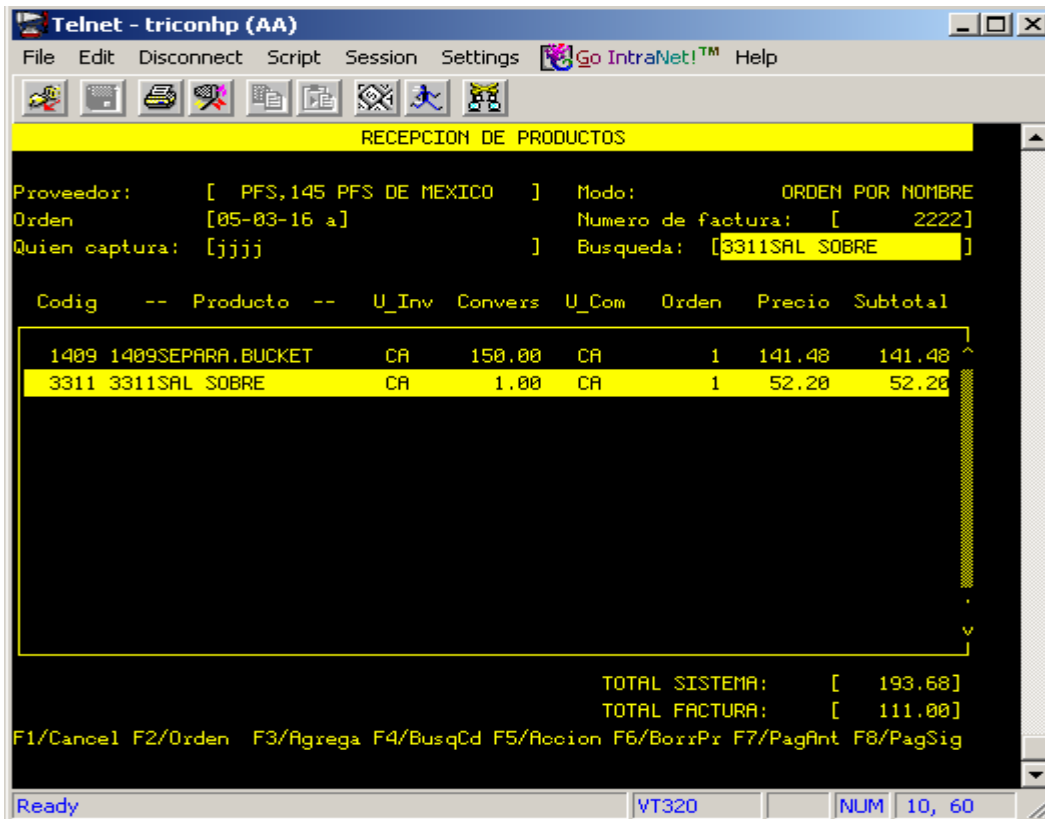


Figura 3.3 Recepción de proveedores

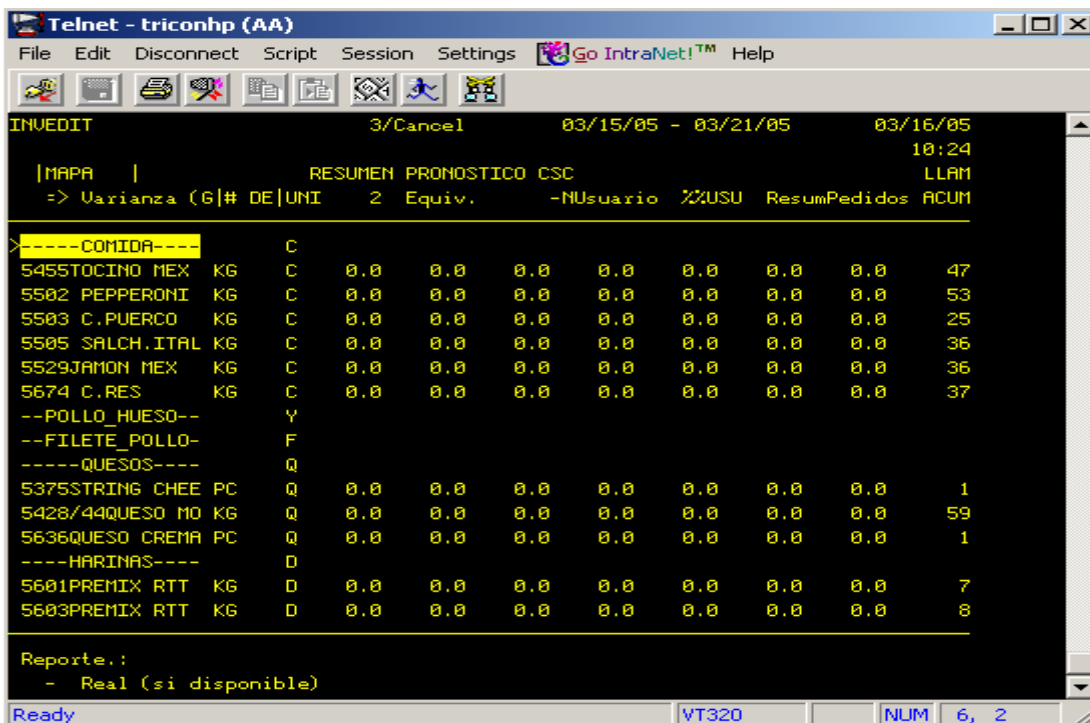


Figura 3.4 Inventario de entradas y salidas

En cualquiera de las dos opciones para registrar su recepción tiene que buscar producto por producto para añadirlo ó editarlo, además de que las recepciones deben registrarse en unidades de inventario y en la hoja de remisión vienen escritas en unidades de proveedor, por lo que nuevamente hay que hacer las conversiones de forma manual sobre la hoja de papel.

La pantalla de la Figura 3.4 además sirve para registrar las transferencias tanto de entrada como de salida, de modo que al ver las cantidades de producto en existencia no se sabe si es una transferencia ó una recepción. Finalmente ésta misma pantalla es la que se usa para hacer el cierre semanal capturando los valores finales de producto.

Es notorio que el proceso actual está repleto de puntos que pueden hacer al proceso difícil de operar, susceptible a errores y difícil de controlar lo que puede ocasionar muchas pérdidas monetarias al restaurante reduciendo las utilidades de forma importante, por lo que los puntos débiles encontrados, son los que se intenta corregir con el nuevo sistema, además de añadir más puntos de control del proceso.

### **3.2 Requerimientos del usuario**

Después de algunas reuniones con los usuarios implicados en la creación de este nuevo sistema, se obtienen una serie de requerimientos, los cuales se han dividido de acuerdo a los distintos módulos a desarrollar. Estos módulos son:

- Módulo para la carga de información en la Base de Datos
- Módulo para la toma del pedido.
- Módulo de consulta de remisiones pendientes.
- Módulo para el registro de las recepciones.
- Módulo para el registro de transferencias.
- Módulo para la captura del inventario semanal.

La descripción de los requerimientos para los módulos mencionados se tratará en los siguientes puntos.

### **3.2.1 Módulo para la carga de información en la Base de Datos**

Este módulo se requiere desarrollar porque es importante que cierta información se encuentre disponible en la base de datos para el correcto funcionamiento de la aplicación. Ésta información varía dependiendo de ciertas circunstancias como productos de venta, disponibilidad de producto por parte del proveedor, precios, etc. Los requerimientos que se listan a continuación son para carga de catálogos y remisiones:

- La información de los catálogos debe ser actualizada cada vez que sea requerido, esto se puede dar generalmente por cambio de precios y/o productos de los proveedores.
- La información de los catálogos acerca de los productos de inventario, productos de proveedor y proveedores sólo debe ser actualizada y/o insertada, no debe ser eliminada.
- La forma de hacer que no se tomen en cuenta los productos de inventario y/o de proveedor se debe manejar por medio de un indicador que indique el estatus del producto.
- Se debe tener las remisiones de cada restaurante en un archivo cuyo nombre deberá ser la fecha de carga.
- La remisión proporcionada por el proveedor debe ser cargada al sistema del restaurante únicamente si se tiene la referencia del pedido al que pertenece y/o si es un pedido forzado<sup>6</sup>.
- La remisión proporcionada por el proveedor debe ser cargada en el momento de hacer el poleo a los restaurantes.

---

<sup>6</sup> Pedido forzado se refiere a un pedido surtido al restaurante sin que éste haya sido solicitado, aunque en realidad sí es solicitado por el departamento de logística de las oficinas administrativas.

### 3.2.2 Módulo para la toma del pedido

Este módulo se requiere desarrollar para procesar generar y procesar la información de los pedidos. La lista de requerimientos identificados es la siguiente:

- El pedido sugerido que se genera en el sistema original, también debe ser mostrado en esta nueva aplicación, pero únicamente para los productos que tienen Pedido Sugerido diferente de cero.
- Se deben mostrar todos los productos de todos los proveedores que estén activos.
- Se debe permitir filtrar los productos asociados a cada proveedor.
- Se debe permitir la selección de todos los productos para ordenar, se debe tener la alternativa de sólo seleccionar los productos que tengan pedido sugerido distinto de cero y también se debe permitir añadir productos de manera individual (sin importar el valor de su pedido sugerido).
- Los productos elegidos deben ser desplegados en una pantalla en la cuál se le da la posibilidad al usuario de capturar la cantidad recibida en unidad de inventario ó en unidad de proveedor. Las conversiones entre ambas unidades se deben hacer de manera automática.
- La pantalla donde se capturan las cantidades que se desean ordenar del producto debe tener una columna que muestre los valores de producto en tránsito y si éste valor es diferente de cero, se debe restar al pedido sugerido original de manera que se tenga un nuevo pedido sugerido más aproximado a lo que realmente se necesita.
- Al terminar de realizar el pedido, debe aparecer una pantalla donde se pueda revisar el pedido antes de confirmarlo, además se debe permitir hacer las correcciones necesarias en caso de haber errores.
- Al confirmar que el pedido es correcto, el sistema debe mostrar la fecha en la cuál el proveedor se llevará su pedido y le debe permitir adelantar sus pedidos hasta 7 días. Después de elegir la fecha, el sistema debe mostrar una pantalla con el pedido confirmado. Además se debe generar un archivo

plano que sirva para hacer la carga de la información de la orden a las oficinas administrativas.

- El sistema debe permitir imprimir tanto la Revisión del Pedido como la Confirmación del mismo.

### **3.2.3 Módulo de consulta de remisiones pendientes**

Este módulo se requiere desarrollar para obtener la comparación entre lo que se ordenó en el pedido y lo que el proveedor dice que surtirá en la remisión, de manera que se tenga la previsión para obtener los suficientes insumos críticos para la venta por medio de transferencias de los restaurantes cercanos. Los requerimientos son:

- El sistema debe permitir elegir el proveedor del cuál se desea consultar la remisión.
- El sistema debe permitir elegir el número de remisión que desea consultar.
- El sistema debe mostrar un reporte donde se vea una comparación entre el pedido y la remisión, estas diferencias deberán ser realizadas para una mejor visualización.
- El sistema debe permitir imprimir este reporte.

### **3.2.4 Módulo para el registro de las recepciones**

Este módulo se requiere desarrollar para llevar a cabo el registro de la recepción dentro del sistema de manera que se afecte al inventario. Los requerimientos para este módulo son:

- El sistema debe permitir elegir el proveedor del cual se desea recibir producto.
- El sistema debe permitir hacer tres diferentes tipos de recepción:
  - I. De remisión. Este tipo de recepción se debe hacer cuando en el sistema se carga la remisión que entrega el proveedor. Para este caso, el sistema debe permitir elegir el número de remisión del cuál



desea recibir. Al elegirlo se deben mostrar los productos y las cantidades que se encuentran en la remisión.

II. De orden. Este tipo de recepción se debe hacer cuando en el sistema no se encuentra cargada la remisión que entrega el proveedor, pero sí se tiene registrada la orden dentro del mismo. Para este caso, el sistema debe permitir elegir el número de orden de la cuál se desea recibir. Al elegirlo se deben mostrar los productos y las cantidades que se encuentran en la orden.

III. Sin ninguna de las anteriores. Este tipo de recepción se debe hacer cuando en el sistema no se encuentra cargada la remisión que entrega el proveedor y tampoco se tiene la orden dentro del mismo. Este es el caso de un pedido levantado telefónicamente por estar fuera de tiempo.

- Para cualquiera de los tres tipos de recepción el sistema debe permitir añadir ó eliminar los productos que requieran.
- El sistema sólo debe permitir recibir de un proveedor a la vez, por lo tanto, los productos a añadir deberán estar asociados con él.
- Se debe contar con un código de discrepancia en la pantalla donde están los productos que se están recibiendo que deberá ser utilizado en el caso de que se modifique alguna cantidad del producto.
- Para evitar errores en la captura de cantidades (por la confusión de unidades de inventario y presentación), el sistema debe permitir capturar como máximo la cantidad de la suma de las últimas cuatro recepciones registradas para el producto. Si no se tienen estas recepciones, entonces el sistema permitirá capturar la cantidad que se desee hasta reunir las cuatro recepciones requeridas para “topar” la cantidad.
- Al terminar de registrar la recepción, debe aparecer una pantalla donde se pueda revisar la recepción antes de confirmarla, además se debe permitir hacer las correcciones necesarias en caso de haber errores.

- Al confirmar que la recepción es correcta, el sistema debe mostrar una pantalla con la recepción confirmada. Además de esto se deben generar los siguientes archivos:
  - I. Archivo plano con la información de los productos y cantidades recibidas para hacer la carga de la información de la recepción a las oficinas administrativas.
  - II. Archivo plano con información de la recepción en un formato entendible para el sistema actual, éste se genera para alterar el archivo donde se almacena el inventario y que sirve para cálculos de algunos índices útiles para el restaurante. Después de que se genera este archivo se debe ejecutar un comando existente en el sistema actual para insertar los datos del inventario, de otra forma, el archivo antes mencionado no se altera.
  - III. Archivo plano con información de la recepción con un formato definido para la carga en oficinas administrativas para otros propósitos.
- Después de confirmar la recepción, se debe desplegar un reporte que muestra las diferencias que existen entre el pedido y la recepción, por lo que éste reporte sólo se deberá desplegar si la recepción se hace de la remisión cargada al sistema ó de la orden.
- Tanto la revisión de la recepción, confirmación de la recepción y reporte de diferencias se deben poder imprimir.

### **3.2.5 Módulo para el registro de transferencias**

Este módulo se requiere desarrollar para llevar a cabo el registro de las transferencias de entrada y salida dentro del sistema de manera que se afecte al inventario. Los requerimientos para este módulo son:

Para las transferencias de entrada:

- El sistema debe solicitar el restaurante de origen para el producto, si este restaurante no existe, entonces el sistema no debe permitir recibir la

transferencia de él. Sólo se debe permitir hacer la transferencia de entrada de un restaurante a la vez.

- El sistema permitirá añadir los productos que se recibirán por medio de un listado de productos sin importar el proveedor al que pertenezcan.
- La captura de la cantidad de producto a recibir en la transferencia se dará únicamente en unidades de inventario.
- Para evitar errores en la captura de cantidades (por la confusión de unidades de inventario y presentación), el sistema debe permitir capturar como máximo la cantidad del 25% de la suma de las últimas cuatro recepciones registradas para el producto. Si no se tienen estas recepciones, entonces el sistema permitirá capturar la cantidad que se desee hasta reunir las cuatro recepciones requeridas para “topar” la cantidad.
- Al terminar de registrar la transferencia, el sistema debe desplegar una pantalla de revisión de la transferencia, también se debe permitir hacer las correcciones necesarias en caso de detectar algún error en esta pantalla.
- Al confirmar que la transferencia es correcta, el sistema debe mostrar una pantalla con la transferencia de entrada confirmada. Además de esto se deben generar los siguientes archivos:
  - I. Archivo plano con información de la transferencia en un formato entendible para el sistema actual, éste se genera para alterar el archivo donde se almacena el inventario y que sirve para cálculos de algunos índices útiles para el restaurante. Después de que se genera este archivo se debe ejecutar un comando existente en el sistema actual para insertar los datos del inventario, de otra forma, el archivo antes mencionado no se altera.
  - II. Archivo plano con información de la transferencia de entrada con un formato definido para la carga en oficinas administrativas para otros propósitos.

Para las transferencias de salida:

- El sistema debe solicitar el restaurante destino para el producto, si este restaurante no existe, entonces el sistema no debe permitir la transferencia a él. Sólo se debe permitir hacer la transferencia de salida de un restaurante a la vez.
- El sistema permitirá añadir los productos que se recibirán por medio de un listado de productos sin importar el proveedor al que pertenezcan.
- La captura de la cantidad de producto a recibir en la transferencia se dará únicamente en unidades de inventario.
- El sistema no debe permitir transferir más de la cantidad que tiene disponible del producto.
- Al terminar de registrar la transferencia, el sistema debe desplegar una pantalla de revisión de la transferencia, también se debe permitir hacer las correcciones necesarias en caso de detectar algún error en esta pantalla.
- Al confirmar que la transferencia es correcta, el sistema debe mostrar una pantalla con la transferencia de salida confirmada. Además debe generar un archivo plano con información de la transferencia de salida con un formato definido para la carga en oficinas administrativas para otros propósitos. Además se debe modificar el archivo del inventario directamente.

### **3.2.6 Módulo para la captura del inventario semanal**

Este módulo se requiere desarrollar para capturar la cantidad de inventario final de la semana, este módulo debe considerar los movimientos realizados dentro del sistema: recepciones, transferencias de entrada y de salida. Los requerimientos de este módulo son:

- El sistema debe desplegar una pantalla con las siguientes características:
  - I. Debe mostrar los productos que se deben inventariar clasificados por familia al igual que el SAI.

- II. Debe mostrar los movimientos que se han hecho del producto a lo largo de la semana, estos son: inventario inicial, recepciones, transferencias de entrada y transferencias de salida.
  - III. Debe permitir la captura de inventario final en unidades de proveedor, unidades de inventario y unidades de receta.
  - IV. Se debe guardar el inventario en unidades de inventario, por lo que las unidades de receta y de proveedor se deben convertir en unidades de inventario automáticamente.
  - V. Debe mostrar los valores de uso real, uso ideal las varianzas en el producto y en dinero.
  - VI. Debe permitir capturar las mermas de producto en unidades de inventario.
  - VII. Finalmente debe calcular conforme a los valores capturados de inventario final y merma, la cantidad faltante de producto en unidades de inventario.
- El sistema no debe permitir la captura de inventario final de cantidades mayores a las que se tienen en existencia (esto es la suma del inventario inicial + recepciones + transferencias de entrada – transferencias de salida)
  - Cuando se tenga una eficiencia (uso ideal/ uso real) menor del 90% ó mayor al 100% se debe hacer notoria para llamar la atención.
  - Al terminar la captura del inventario final de los productos se debe desplegar una pantalla donde se pueda revisar las cantidades capturadas para corregir en caso de ser necesario hacerlo.
  - Finalmente si todo se realizó correctamente se debe mostrar una pantalla donde se puede imprimir el comprobante del inventario capturado. Además se deben escribir los valores en el archivo donde se encuentra el inventario.

### 3.3. Presupuestos

#### 3.3.1 Hardware

En la tabla 3.1 se muestra el equipo que se necesita para el desarrollo de la aplicación y el costo de cada elemento:

<b>Elemento de Hardware</b>	<b>Costo en USD (Dólares)</b>
APC Smart UPS 1000VA 670W	400
HP DC7600, 1 GB con Tarjeta de Expansión PCI y Puerto Serial Adicional	1230
Disco Duro 40 GB	110
Modem TrendNet 560x External	80
Monitor SVGA Color	150
Regulador Sola Basic 2KVA	400
Tarjeta de Red 10/100 3COM 3C905B-TX	50
Tarjeta de video nVidia GeForce Mx4000 PCI	70

*Tabla 3.1 Costos del hardware empleado*

#### 3.3.2 Software

El software a utilizar para el desarrollo del sistema no tiene costo alguno ya que se usará software abierto, sin embargo, el sistema tiene un costo de desarrollo el cual se cotizará por horas teniendo el siguiente desglose de actividades con el tiempo de desarrollo y el subtotal del costo. El costo por análisis y diseño será de 50 USD la hora, el de la programación, pruebas y documentación en 30 USD la hora, todo esto se muestra en la tabla 3.2.

<b>Módulo a Desarrollar</b>	<b>Horas</b>	<b>Costo subtotal en USD (Dólares)</b>
Reuniones con el usuario para definición de requerimientos	8 horas	400
Diseño y creación de la Base de Datos	48 horas	2,400
Desarrollo del módulo para la carga de información en la Base de Datos	56 horas	1,680
Desarrollo del módulo para la toma del pedido	64 horas	1,920
Desarrollo del módulo de consulta de remisiones pendientes.	32 horas	960
Desarrollo del módulo para el registro de las recepciones.	80 horas	2,400
Desarrollo del módulo para el registro de transferencias	64 horas	1,920
Desarrollo del módulo para la captura del inventario semanal	80 horas	2,400
Desarrollo del programa de instalación	24 horas	720
Etapa de pruebas	40 horas	1,200
Manual de usuario	32 horas	960
<b>Costo total (aproximado)</b>		<b>16,960</b>

*Tabla 3.2 Costos aproximados del desarrollo del sistema*

### **3.4 Análisis de Riesgos de Seguridad**

#### **3.4.1 RIESGO DE PASO**

##### **3.4.1.1 Acceso libre de usuario**

El titular del servicio no prevé el resguardo de su contraseña y su nombre de usuario, dejando libre acceso al personal que bajo discreción del titular estén en posibilidades de apoyarle, siendo el titular de la contraseña el responsable del mal uso de la misma aunque este antepone la ética del usuario del momento.

### **3.4.1.2 Omisión de resguardo de usuario y contraseña**

El titular del servicio omite el apoyo de sus adjuntos, hace buen uso del servicio pero pierde la consideración del buen resguardo de sus claves, dejando a la vista de cualquier individuo en tránsito de su lugar de trabajo. Regularmente dejándola en papeles o bien en ubicaciones como libretas o agendas.

### **3.4.1.3 Omisión de encriptación de paso(contraseñas simples)**

El titular resguarda ahora su contraseña y su usuario, pero las características de su contraseña se vuelven comunes que ante la intromisión de un ente lógico-computacional y con un poco de trabajo puede ser descubierta y violentada. Esto ocurre de forma recurrente cuando el titular y el personal de soporte no tienen un procedimiento de encriptación de la información, siendo el titular de la contraseña el que coloca su propuesta de contraseña la cual queda en condiciones “blancas” (sin modificación alguna) siendo las más recurridas las fechas de nacimiento, fechas importantes para el titular, nombres de familiares o conocidos al igual que nombres de mascotas o marcas comerciales incluyendo a las personalidades a las cuales es aficionado.

## **3.4.2 RIESGO DE USO**

### **3.4.2.1 Solicitudes sin autorización**

El titular de la cuenta bajo omisión de desconexión deja en acceso libre su sesión lo cual provocaría que cualquiera que este en posibilidades de hacer uso no autorizado de la cuenta tenga la oportunidad de hacerlo colocando una solicitud de pedido sin más restricciones dado que esta bajo un cargo no autorizado lo cual le estaría facilitando la autenticación del solicitante dado que quien es el titular ya lo esta.



### **3.4.2.2 Observadores sin autorización**

Un observador no autorizado estaría contemplado, en las condiciones de ocasión o de intención, en el momento de uso de software, esto es, que un observador por casualidad esta siendo participe del uso de software esto por la omisión del titular de la cuenta que al momento de estar en uso del mismo software no toma atención si el personal que esta a su lado esta o no autorizado para ello; en el otro caso es el observador quien provoca la ocasión para ver la utilización del software lo cual esta provocando una situación de abuso de confianza por su parte aunque el titular de la cuenta no se percate de la misma.

### **3.4.2.3 Usuario sin autorización.**

En este caso es un usuario que esta violentando el acceso a la cuenta, ya sea por medios violentos o bien de intención lógica, esto es, el usuario empieza a jugar con posibles claves de acceso (como ya se había comentado pensando en posibilidades de comunes como las fechas de nacimiento), o bien se encuentra en un estado frenético en el cual hace omisión de la estipulaciones básicas de tranquilidad e intenta acceder de alguna otra forma al software ya sea atentando contra la integridad física del usuario titular de la cuenta o bien del medio de transmisión en donde esta instalado el software.

## **3.4.3 VIOLACIÓN DE LA INTEGRIDAD.**

### **3.4.3.1 Ataque a las instalaciones lógicas.**

Aunque la situación de integridad en el plano informático hace mas referencia a la situación del tratamiento lógico de la información, es de considerarse que esta integridad esta considerada en dos etapas, la física con el resguardo del hardware y la lógica con el software, en el caso de la primera es indudable que debe haber un contacto física para poder hacer uso del mismo, pero en el lógico puede verse afectado por distintas fuentes, desde la presencia de conexiones no certificadas

para estar en uso provocando el paso de software indeseable como spam, virus informáticos, o software malicioso. En el caso del spam, aunque de menor peligro por su llegada al equipo de trabajo este software puede servir de puerta “interdimensional” (acceso entre portales de conexión) con los muy desagradables virus.

Aunque en el caso de los virus esta reconocido una clasificación mas extensa (y por tal motivo no es de interés para esta tesis su indagación mas afondo) solo nos quedaremos con la base genérica de los mismos. Entre tanto a la existencia de virus informáticos podemos generalizar la presencia de virus Troyanos, en cuyo caso su método infeccioso esta presentado por ocultarse en la transmisión de algún o de algunos tipos de mensajes eléctricos; por otro lado están las Bombas de Tiempo, en cuyo método infeccioso esta muy restringido a la fecha u hora de detonación de la misma, esto da una ventaja al usuario o al personal encargado de actuar; y por último consideramos a los Gusanos informáticos los cuales se dedican a la destrucción de la información en el equipo huésped.

En el caso de nuestro usuario, dejar una puerta abierta significaría que algún software malicioso pudiera tomar posesión de la transmisión o recepción de la información en cuyo caso la situación de la información que fue corrompida.

#### **3.4.4 Políticas de seguridad básicas.**

Es evidente que el resguardo de la información puede quedar sellado con el seguimiento de pasos simples que nos ayuden al control del uso, de su alcance y su respaldo, de estos pasos podemos establecer condiciones generales o bien particulares para cada propietario de información o bien de su tráfico común. Es necesario también entender que dichas condiciones de trabajo no sean omitidas y esto con llevaría a un buen uso de los recursos existentes dentro del centro de trabajo.

#### **3.4.4.1 Políticas de uso.**

- El usuario en turno dará de alta su sesión verificando que el equipo al cual se este conectando no posea una sesión previa abierta, de ser ese el caso procederá a cerrar dicha sesión y reporteara al personal responsable para que pueda tomar cartas en el asunto y aplicar las sanciones correspondientes. Una vez cerrada la sesión anterior podrá habilitar la propia.
- Iniciada su sesión el usuario, deberá tomarse el tiempo requerido sin interrupciones de preferencia para realizar su pedido, si es el caso de que por causas de fuerza mayor el usuario tenga que dejar su posición, este deberá cerrar sin excepciones su sesión haya o no concluido con su solicitud.
- Retomada su solicitud o bien, una vez concluida el usuario deberá de cerrar su sesión y deberá de verificar que esto si haya ocurrido para evitar el mal uso de su cuenta y no verse sancionado por su omisión.

#### **3.4.4.2 Políticas de acceso.**

- Establecer claves de acceso en condiciones seguras (considerando su encriptación para evitar el ser violentadas de forma sencilla).
- Disponer e implementar acciones que permitan el buen guardado de las claves por parte del usuario, esto implica tener en consideración la habilidad del usuario para retener su clave provista.
- Activar y/o dar de alta cortafuegos para impedir el paso de software malicioso y de igual forma disponer de filtros de paso para la IP de cada usuario conectado a nuestro servidor del servicio, esto puede ser fácilmente implementado con iptables dentro de las diferentes distribuciones de Linux.

#### **3.4.4.3 Políticas de respaldo.**

- Establecer periodos en los cuales los diferentes usuario estén fuera del servicio para hacer una copia del periodo completo del servicio.

# Capítulo IV.

## Diseño.

- 4.1** Herramientas de desarrollo.
  - 4.1.1** Linux.
    - 4.1.1.1** Introducción.
    - 4.1.1.2** Breve Historia.
    - 4.1.1.3** Características principales.
    - 4.1.1.4** Características adicionales.
  - 4.1.2** Posgresql.
    - 4.1.2.1** Introducción e Historia.
    - 4.1.2.2** Características Principales.
  - 4.1.3** Conceptos Básicos de POO.
    - 4.1.3.1** Introducción.
    - 4.1.3.2** Terminología.
  - 4.1.4** Java.
    - 4.1.4.1** Breve Historia.
    - 4.1.4.2** Características Principales.
    - 4.1.4.3** Entorno de Ejecución.
    - 4.1.4.4** Tipo de aplicaciones.
  - 4.1.5** Apache Tomcat.
    - 4.1.5.1** Introducción.
    - 4.1.5.2** Principales Características.
  - 4.1.6** Requerimientos para el desarrollo de la aplicación.
    - 4.1.6.1** Requerimientos mínimos de Hardware.
    - 4.1.6.2** Requerimientos mínimos de Software.
- 4.2** Casos de uso.  
Diseño de la Base de Datos.

## **Definición de la Plataforma de Desarrollo**

### **4.1 Herramientas de desarrollo**

#### **4.1.1 Linux**

##### **4.1.1.1 Introducción**

La mayor parte de las computadoras que existen en la actualidad están diseñadas de forma que puedan ejecutar diversas tareas o programas. Estos programas pueden ir desde un procesador de textos, a un programa para la animación de gráficos tridimensionales o distintos tipos de juegos.

Para su correcto funcionamiento deben ser además capaces de acceder a los recursos de que dispone la máquina, como por ejemplo escribir o leer datos en un disco duro, mostrar un gráfico, etc. Es evidente, que si cada programa actuase de una forma independiente, existirían graves problemas y conflictos, puesto que, por ejemplo, tendrían libertad para escribir sus datos sobre los de otro, etc.

Para solucionar este tipo de problemas se desarrollaron los Sistemas Operativos, los cuales aportan unos mecanismos y reglas básicas de funcionamiento, de forma que los programas puedan acceder a los recursos de la computadora adecuada. Aunque ésta fue la funcionalidad inicial de los sistemas operativos, con el tiempo se han añadido otras muchas, como la ejecución de programas, el control de la memoria, la creación y control de interfaces gráficas de usuario, etc.

En la actualidad existen una gran cantidad de sistemas operativos dependiendo del tipo de computadora en la que se va a ejecutar.

Linux es un sistema operativo para equipos personales y estaciones de trabajo que incorpora una interfaz gráfica de usuario. Se distingue por su potencia y flexibilidad, además se distribuye gratuitamente bajo la licencia GPL (General Public License) de GNU (Gnu's Not UNIX) gestionado por Free Software Foundation a diferencia de UNIX del cual se basa.

#### 4.1.1.2 Breve historia

Linux fue desarrollado a principios de la década de los noventa por Linus Torvalds. Torvalds tenía la convicción de diseñar un sistema operativo que superara las limitaciones de MINIX y se acercara más a UNIX tanto en apariencia como en el kernel (núcleo del sistema operativo). Y sobre todo a equipos PC y compatibles.

Este se trataba de un proyecto universitario por lo que Torvalds pidió ayuda a través de Internet el 3 de Julio de 1991. Durante el desarrollo Torvalds encontró muchos problemas a lo largo de la programación del kernel. Pero Linux empezó a disponer de controladores para los dispositivos internos de la PC y un funcionamiento correcto del disco el 3 de julio, unas horas después de enviar su primer e-mail informado sobre su proyecto. Dos meses más tarde Linux empezaba a funcionar y el código fuente de la primera versión 0.01 ya estaba disponible. La versión 0.01 incluía un bash shell 1.08 y el compilador gcc 1.40.

El 5 de octubre de 1991, Torvalds anuncio la primera versión "Oficial" de Linux (versión 0.02). Con esta versión se pudo ejecutar bash (GNU Bourne Again Shell) y gcc (El compilador GNU de C). En esta etapa de desarrollo aún no se pensaba en los términos: soporte, documentación o distribución. Después de la versión 0.03, Torvalds salto en la numeración hasta la 0.10, más y más programadores a lo largo y ancho de Internet empezaron a trabajar en el proyecto y después de sucesivas revisiones, se incremento el número de versión hasta la 0.95 (Marzo 1992). Más de un año después (diciembre 1993) el núcleo del sistema estaba en la versión 0.99 y la versión 1.0 llegó hasta el 14 de marzo de 1994.

#### 4.1.1.3 Características principales

Dentro de las características más significativas que distinguen a Linux están:

1. *Multitarea*. Un sistema es multitarea cuando es capaz de ejecutar varias tareas al mismo tiempo, Linux utiliza la llamada multitarea preventiva, es decir, el sistema es el encargado de asignar tiempo de microprocesador, de

esta manera se asegura que todos los programas utilizados en un momento serán ejecutados.

2. *Multiusuario*. Linux permite que varios usuarios puedan utilizar la maquina y sus recursos al mismo tiempo
3. *Multiplataforma*. Es la capacidad de poder funcionar en diversas plataformas de hardware.
4. *Multiprocesador*. Soporta sistemas que contienen más de un procesador.
5. *Memoria Virtual*. Memoria virtual usando paginación (sin intercambio de procesos completos) a disco. A una partición o un archivo en el sistema de archivos, o ambos, con la posibilidad de añadir más áreas de intercambio sobre la marcha. Un total de 16 zonas de intercambio de 128Mb de tamaño máximo pueden ser usadas en un momento dado con un límite teórico de 2Gb para intercambio. Este límite se puede aumentar fácilmente con el cambio y compilación de unas cuantas líneas en el código.
6. *Consolas virtuales múltiples*. Varias sesiones de nombre del usuario a través de la consola entre las que se puede cambiar con las combinaciones adecuadas de teclas (totalmente independiente del hardware de video). Y se crean dinámicamente.
7. *Shells programables*. El shell es un programa que se encuentra en el sistema que interpreta los comandos que son la interfaz con el sistema operativo y facilitan su control, el shell es el encargado de comunicarnos con el kernel o núcleo del sistema operativo. Al iniciar una sesión cada usuario ejecuta un shell particular. El sistema nos muestra un carácter tal como: \$ o # o > u otro, siendo éstos los más comunes y dependerá del tipo de shell que iniciamos.

#### **4.1.1.4 Características adicionales**

Como algunas características adicionales tenemos:

- Uso de nombres largos, hasta 254 caracteres.
- Linux considera diferentes las letras minúsculas y mayúsculas. Lo que permite mayor número de posibilidades para nombres de archivos.



- Completado de palabra por el tabulador.
- Los archivos ocultos se manejan poniendo un punto al inicio del nombre.
- El shell contiene dos caracteres comodines que son ? y \* ; Por medio de ? podemos sustituir una letra y por medio de \* sustituimos toda una cadena de caracteres.
- La entrada estándar de los comandos es el teclado.
- La salida estándar es el monitor.
- Algunos comandos se manejan como expresiones regulares.
- Linux maneja rutas absolutas y relativas; una ruta relativa se refiere a tomar en cuenta a partir del directorio actual y una ruta absoluta se refiere a partir de la raíz (/home/usuario/directorio/).
- Manejos, configuraciones, dispositivos son manejadas a través de archivos, es decir todo en Linux es representado como un archivo dentro del sistema.

Otro concepto importante asociado con Linux son las distribuciones. Una *distribución* es un agrupamiento del núcleo del sistema operativo Linux (la parte desarrollada por Linus Torvalds) y otra serie de aplicaciones de uso general o no tan general. En principio las empresas que desarrollan las distribuciones de Linux están en su derecho al cobrar una cierta cantidad por el software que ofrecen, aunque en la mayor parte de las ocasiones se pueden conseguir estas distribuciones desde Internet, de *revistas* o de *amigos*, siendo todas estas formas gratuitas y legales.

Alguna de las distribuciones más conocidas son Fedora, *Debian*, *Slackware* y *SUSE*, *Ubuntu* todas ellas incluyen el software más reciente y empleado lo cual incluye compiladores de C/C++, editores de texto, juegos, programas para el acceso a *Internet*, así como el entorno gráfico de *Linux: X Window*.

## **4.1.2 Postgresql.**

### **4.1.3.1 Introducción e historia.**

PostgreSQL es un sistema manejador de base de datos denominado Objeto-Relacional que ha sido desarrollado desde 1977. Comenzó como un proyecto denominado Ingres en la Universidad Berkeley de California.

En 1986 otro equipo dirigido por Michael Stonebraker de Berkeley continuó el desarrollo del código de Ingres para crear Postgres. En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgres fue renombrado a PostgreSQL, tras un breve periodo en que fue conocido como Postgres95.

PostgreSQL está considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre.

### **4.1.3.1 Características principales.**

PostgreSQL puede ser usado, modificado y distribuido gratuitamente para cualquier propósito que puede ser privado, comercial ó académico.

Postgresql soporta casi toda la sintaxis SQL (incluyendo subconsultas, transacciones, tipos y funciones definidas por el usuario), contando también con un amplio conjunto de enlaces con lenguajes de programación (incluyendo C, C++, Java, Perl, PHP y Python).

PostgreSQL usa el modelo cliente/servidor, una sesión consiste en los siguientes procesos:

- Un proceso servidor que maneja los archivos de la base de datos, acepta las conexiones de las aplicaciones cliente, y ejecuta acciones a nombre de los clientes. Este programa es llamado postmaster.
- La aplicación del cliente que requiere ejecutar las operaciones en la base de datos. Las aplicaciones cliente pueden ser de diferente naturaleza:

podría ser una herramienta orientada a texto, una aplicación gráfica, un servidor web que acceda a la base de datos para desplegar páginas web o una herramienta especializada de mantenimiento de base de datos. Algunas aplicaciones cliente son proveídas por la distribución de PostgreSQL, otras deben ser desarrolladas por los usuarios.

Como una aplicación típica cliente/servidor, el cliente y el servidor pueden estar en diferentes máquinas, en ese caso se comunicarán sobre el protocolo TCP/IP. Esto es muy importante contemplarlo debido a que los archivos accedidos en una máquina cliente podrían no estar accesibles en la máquina servidor.

Las principales características de este gestor de bases de datos son:

- Implementación del estándar SQL92/SQL99.
- Soporta distintos tipos de datos: además del soporte para los tipos base, también soporta datos de tipo fecha, monetarios, elementos gráficos, datos sobre redes (MAC, IP), cadenas de bits, etc. También permite la creación de tipos propios.
- Incorpora una estructura de datos array (arreglo).
- Incorpora funciones de diversa índole: manejo de fechas, geométricas, orientadas a operaciones con redes, etc.
- Permite la declaración de funciones propias, así como la definición de triggers (disparadores).
- Soporta el uso de índices, reglas y vistas.
- Incluye herencia entre tablas (aunque no entre objetos, ya que no existen), por lo que a este gestor de bases de datos se le incluye entre los gestores objeto-relacionales.
- Permite la gestión de diferentes usuarios, como también los permisos asignados a cada uno de ellos.

La siguiente tabla 4.1.2.2. muestra una comparación entre Postgresql, Mysql y manejadores de bases de datos comerciales:

	Postgresql	MySQL	Comercial
<b>Integridad de Datos</b>			
Apegado a ACID	X	X	X
Bloqueo a nivel registro	X	X	X
Deshacer cambios parciales	X	X	X
<b>Características avanzadas</b>			
Procedimientos almacenados	X	En 5.0	X
Vistas	X	En 5.0	X
Triggers (Disparadores)	X	En 5.1	X
Secuencias	X	X	X
Cursores	X	In 5.0	X
Tipos de dato definidos por el usuario	X	?	X
<b>Índices</b>			
Columna	X	X	X
Multi-columnas	X	X	X
Llave primaria	X	X	X
Texto completo	X	X	X
<b>Replicación</b>			
Single-master	X*	X	X
Multi-master	X**		X
<b>Métodos de Interfase</b>			
ODBC/JDBC	X	X	X
C/C++, Java	X	X	X

\* Open Source, pero se tiene con otro vendedor

\*\* Las soluciones existen, pero son comerciales.

Tabla 4.1.2.2 Comparación entre Postgresql, Mysql y manejadores de bases de datos comerciales

### **4.1.3 Conceptos Básicos de Programación Orientada a Objetos.**

#### **4.1.3.1 Introducción**

El análisis es la base de la programación estructurada, es decir, la descomposición de una tarea en tareas más pequeñas. Un programa no es otra cosa que una colección de funciones que son llamadas sucesivamente por la función principal y única del programa. Cada función tiene sus propias variables, es un módulo independiente. La Programación Orientada a Objetos es el siguiente paso en la evolución de los lenguajes de programación, que combina funciones y datos en una unidad autoconsistente denominada clase.

La orientación a objetos es un paradigma de programación que facilita la creación de software de calidad por sus factores que potencian el mantenimiento, la extensión y la reutilización del software generado bajo este paradigma.

La programación orientada a objetos trata de amoldarse al modo de pensar del hombre y no al de la máquina. Esto es posible gracias a la forma racional con la que se manejan las abstracciones que representan las entidades del dominio del problema, y a propiedades como la jerarquía o el encapsulamiento.

El elemento básico de este paradigma no es la función (elemento básico de la programación estructurada), sino un ente denominado objeto. Un objeto es la representación de un concepto para un programa, y contiene toda la información necesaria para abstraer dicho concepto: los datos que describen su estado y las operaciones que pueden modificar dicho estado, y determinan las capacidades del objeto.

#### **4.1.3.2 Terminología**

Los siguientes son algunos términos importantes de este paradigma de programación:

*Objeto.* Se trata de un ente abstracto usado en programación que permite separar los diferentes componentes de un programa, simplificando así su elaboración, depuración y posteriores mejoras.

Los objetos se componen de 3 partes fundamentales: métodos, eventos y atributos.

*Métodos.* Son aquellas funciones que permite efectuar el objeto y que nos rinden algún tipo de servicio durante el transcurso del programa. Determinan a su vez como va a responder el objeto cuando recibe un mensaje.

*Eventos.* Son aquellas acciones mediante las cuales el objeto reconoce que se está interactuando con él. De esta forma el objeto se activa y responde al evento según lo programado en su código.

*Atributos.* Características que aplican al objeto solo en el caso en que el sea visible en pantalla por el usuario; entonces sus atributos son el aspecto que refleja, tanto en color, tamaño, posición, si está o no habilitado, etc.

*Clases.* Son abstracciones que representan a un conjunto de objetos con un comportamiento e interfaz común.

Una clase no es más que una plantilla para la creación de objetos. Cuando se crea un objeto (*instanciación*) se ha de especificar de qué clase es el objeto instanciado, para que el compilador comprenda las características del objeto.

*Abstracción.* Concepto en el cual la mente humana “modeliza” la realidad en forma de objetos. Para ello busca parecidos entre la realidad y la posible implementación de objetos de programa que simulen el funcionamiento de los objetos reales.

La abstracción humana se gestiona de una manera jerárquica, dividiendo sucesivamente sistemas complejos en conjuntos de subsistemas, para así entender más fácilmente la realidad. Esta es la forma de pensar que la orientación a objeto intenta cubrir.

*Encapsulamiento.* Es el principio que permite a los objetos elegir qué información es pública y qué información es oculta para el resto de los objetos. Para ello los objetos suelen presentar sus métodos como interfaces públicas y sus atributos como datos privados e inaccesibles desde otros objetos.

*Herencia.* Mecanismo mediante el cual a partir de una clase base, se puede crear una ó varias clases hijas, de manera que las clases hijas heredan atributos y métodos de su clase padre. Esto ayuda a la reutilización de código.

*Polimorfismo.* Esta propiedad permite que un objeto presente diferentes comportamientos en función del contexto en que se encuentre. Por ejemplo un método puede presentar diferentes implementaciones en función de los argumentos que recibe, recibir diferentes números de parámetros para realizar una misma operación, y realizar diferentes acciones dependiendo del nivel de abstracción en que sea llamado.

## **4.1.4 Java**

### **4.1.4.1 Breve historia**

Java fue diseñado en 1990 por James Gosling, de Sun Microsystems, como software para electrodomésticos.

Inicialmente Java se llamó Oak (roble en inglés), aunque tuvo que cambiar de denominación, debido a que dicho nombre ya estaba registrado por otra empresa.

Tres de las principales razones que llevaron a crear Java son:

- Creciente necesidad de interfaces mucho más cómodas e intuitivas que los sistemas de ventanas que proliferaban hasta el momento.
- Fiabilidad del código y facilidad de desarrollo. Gosling observó que muchas de las características que ofrecían C o C++ aumentaban de forma alarmante el gran coste de pruebas y depuración. Por ello en los sus ratos libres creó un lenguaje de programación donde intentaba solucionar los fallos que encontraba en C++.
- Enorme diversidad de controladores electrónicos. Los dispositivos electrónicos se controlan mediante la utilización de microprocesadores de bajo precio y reducidas prestaciones, que varían cada poco tiempo y que utilizan diversos conjuntos de instrucciones. Java permite escribir un código común para todos los dispositivos.

El proyecto Green fue el primero en el que se aplicó Java, y consistía en un sistema de control completo de los aparatos electrónicos y el entorno de un hogar. Con este fin se construyó una computadora experimental denominada \*7 (Star Seven). El sistema presentaba una interfaz basada en la representación de la casa de forma animada y el control se llevaba a cabo mediante una pantalla sensible al tacto.

Más tarde Java se aplicó a otro proyecto denominado VOD (Video On Demand) en el que se empleaba como interfaz para la televisión interactiva que se pensaba iba a ser el principal campo de aplicación de Java. Ninguno de estos proyectos se convirtió nunca en un sistema comercial, pero fueron desarrollados enteramente en un Java primitivo

Una vez que en Sun se dieron cuenta de que a corto plazo la televisión interactiva no iba a ser un gran éxito, instaron a FirstPerson a desarrollar nuevas estrategias que produjeran beneficios. Entre ellas se encontraba la aplicación de Java a Internet, la cual no se consideró productiva en ese momento.



Bill Joy juzgó que Internet podría llegar a ser el campo adecuado para disputar a Microsoft su primacía en el terreno del software, y vió en Oak el instrumento idóneo para llevar a cabo estos planes.

Para poder presentarlo en sociedad se tuvo que modificar el nombre de este lenguaje de programación y se tuvo que realizar una serie de modificaciones de diseño para poderlo adaptar al propósito mencionado. Así Java fue presentado en sociedad en agosto de 1995.

#### **4.1.4.2 Características principales**

Algunas de las características principales de java son:

- *Es un lenguaje sencillo y potente.* Java ofrece toda la funcionalidad de un lenguaje potente, pero sin las características menos usadas y más confusas de otros lenguajes como C y C++. Como ejemplos de estas características tenemos: aritmética de apuntadores, referencias, registros (struct), definición de tipos (typedef), macros (#define), necesidad de liberar memoria (free). Aunque, en realidad, lo que hace es eliminar las palabras reservadas (struct, typedef), ya que las clases son algo parecido.

La sintaxis de Java es similar a la de C++.

- *Es un lenguaje orientado a objetos.* Java trabaja con sus datos como objetos y con interfaces a esos objetos. Soporta las características del paradigma de la orientación a objetos mencionadas en la sección anterior. Las plantillas de objetos son llamadas, como en C++, *clases* y sus objetos, *instancias*. Estas instancias, como en C++, necesitan ser construidas y destruidas en espacios de memoria.
- *Es distribuido.* Java se ha construido con extensas capacidades de interconexión TCP/IP. Existen librerías de rutinas para acceder e interactuar con protocolos como *http* y *ftp*.

La verdad es que Java en sí no es distribuido, sino que proporciona las librerías y herramientas para que los programas puedan ser distribuidos, es decir, que se corran en varias máquinas, interactuando.

- *Es robusto.* Java realiza verificaciones en busca de problemas tanto en tiempo de compilación como en tiempo de ejecución. La comprobación de tipos en Java ayuda a detectar errores en el ciclo de desarrollo. Java obliga a la declaración explícita de métodos. Maneja la memoria para eliminar las preocupaciones por parte del programador de la liberación o corrupción de memoria. También implementa los *arreglos auténticos*, en vez de listas enlazadas de apuntadores.
- *Es de arquitectura neutral.* El compilador Java compila su código a un archivo objeto de formato independiente de la arquitectura de la máquina en que se ejecutará. Cualquier máquina que tenga el sistema de ejecución (*run-time*) puede ejecutar ese código objeto, sin importar en modo alguno la máquina en que ha sido generado.

El código fuente Java se "compila" a un código de bytes de alto nivel independiente de la máquina. Este código (byte-code) está diseñado para ejecutarse en una máquina virtual que es implementada por un sistema run-time, que sí es dependiente de la máquina.

Podemos ver que lo verdaderamente dependiente del sistema es la *Máquina Virtual Java (JVM)* y las librerías fundamentales, que también nos permitirían acceder directamente al hardware de la máquina. Además, habrá APIs (Application Program Interface) de Java que también entren en contacto directo con el hardware y serán dependientes de la máquina.

- *La ejecución del código Java es segura y fiable:* Los programas no acceden directamente a la memoria de la computadora, siendo imposible que un programa escrito en Java pueda acceder a los recursos de la computadora sin que esta operación le sea permitida de forma explícita. De este modo, los datos del usuario quedan a salvo de la existencia de virus escritos en Java. La ejecución segura y controlada del código Java es una característica única, que no puede encontrarse en ninguna otra tecnología. El código Java pasa muchas pruebas antes de ejecutarse en una máquina. El código se pasa a través de un verificador de byte-codes que comprueba el formato de los fragmentos de código y aplica un probador de teoremas

para detectar fragmentos de código ilegal. Si los byte-codes pasan la verificación sin generar ningún mensaje de error, entonces sabemos que: El código no produce desbordamiento de operandos en la pila, el tipo de los parámetros de todos los códigos de operación son conocidos y correctos, no ha ocurrido ninguna conversión ilegal de datos, tal como convertir enteros en apuntadores, el acceso a los campos de un objeto se sabe que es legal, no hay ningún intento de violar las reglas de acceso y seguridad establecidas

- *Es portable.* Más allá de la portabilidad básica por ser de arquitectura independiente, Java implementa otros estándares de portabilidad para facilitar el desarrollo. Los enteros son siempre *enteros* y además, enteros de 32 bits en complemento a 2. Además, Java construye sus interfaces de usuario a través de un sistema abstracto de ventanas de forma que las ventanas puedan ser implantadas en entornos Unix, Pc o Mac.
- *Es interpretado.* El intérprete Java puede ejecutar directamente el código objeto, enlazar un programa, que normalmente, consume menos recursos que compilarlo.

La verdad es que Java para conseguir ser un lenguaje independiente del sistema operativo y del procesador que incorpore la máquina utilizada, es tanto interpretado como compilado. El código fuente escrito con cualquier editor se compila generando el byte-code. Este código intermedio es de muy bajo nivel, pero sin alcanzar las instrucciones máquina propias de cada plataforma. El byte-code corresponde al 80% de las instrucciones de la aplicación. Ese mismo código es el que se puede ejecutar sobre cualquier plataforma. Para ello hace falta el run-time, que sí es completamente dependiente de la máquina y del sistema operativo, que interpreta dinámicamente el byte-code y añade el 20% de instrucciones que faltaban para su ejecución. Con este sistema es fácil crear aplicaciones multiplataforma, pero para ejecutarlas es necesario que exista el run-time correspondiente al sistema operativo utilizado.

- *Es multithreaded (multihilo)*. Java permite muchas actividades simultáneas en un programa. Los hilos (a veces llamados, procesos ligeros), son básicamente pequeños procesos o piezas independientes de un gran proceso. El beneficio de ser multithreaded consiste en un mejor rendimiento interactivo y mejor comportamiento en tiempo real. Aunque el comportamiento en tiempo real está limitado a las capacidades del sistema operativo subyacente

#### **4.1.4.3 Entornos de ejecución**

Cuando Sun decidió lanzar su nuevo estándar Java, llamado Java2, creó tres diferentes entornos para desarrollo y ejecución de aplicaciones. Estos son:

- *J2SE* (Java 2 Standard Edition) son los entornos de ejecución, las herramientas y APIs que requieren los desarrolladores para escribir, desplegar y ejecutar applets y aplicaciones en Java.
- *J2EE* (Java 2 Enterprise Edition ) es un estándar para el desarrollo, despliegue y gestión de aplicaciones distribuidas multicapa, basadas en componentes. J2EE combina un conjunto de tecnologías en una arquitectura con un modelo de programación de aplicaciones razonable, así como un conjunto de pruebas de compatibilidad para el desarrollo de aplicaciones en el lado del servidor.
- *J2ME* (Java 2 Micro Edition) es un entorno de ejecución Java muy pequeño, altamente optimizado para poder ser ejecutado en dispositivos consumibles, desde teléfonos, móviles, PDAs, etc. Las APIs utilizables en J2ME son un subconjunto de las que incluye J2SE.

#### **4.1.4.4 Tipos de aplicaciones**

Las aplicaciones básicas para desarrollar en java son:

- Programas de aplicación general ejecutados en consolas.
- Programas que se ejecuten en ambientes gráficos.

- Applets son programas que se mandan a una pc remota en donde esta aplicación se carga y corre. Los applets se ejecutan en el navegador de la máquina remota, siempre que éste incluya la máquina virtual.
- Servlets son programas en java que se ejecutan en el web server, que al ser llamado por una máquina remota genera los resultados y los envía a la pc remota en forma de una página html.
- JSP's (Java Server Pages) combina código html y código java en un solo programa o aplicación que también es ejecutado por el web server, ya que lo compila como un servlet.
- Beans y Java Beans tecnologías de tipo servlet pero que están especializadas principalmente en mantener enlaces constantes a bases de datos de cualquier marca, esto permite que una serie de varios programas o aplicaciones a la vez estén interactuando con el mismo java bean y por consiguiente con la base de datos.
- Existen otra serie de aplicaciones diferentes que se pueden construir con java, tales como aplicaciones RMI (Remote Method Invocation), aplicaciones en tiempo real, EJB (Entrepiso Java Beans), etc que se aplican en sistemas empresariales de gran tamaño y complejidad.

#### **4.1.5 Apache Tomcat**

##### **4.1.5.1 Introducción**

Un contenedor de servlets es parte del servidor web o servidor de aplicaciones que se encarga de realizar todo el trabajo de las conexiones, decodificaciones y gestiones de la vida del servlet.

Tomcat funciona como un contenedor de servlets con un entorno JSP que implementa las especificaciones de los servlets y JSP's de SUN Microsystems, se le considera un servidor de aplicaciones.

#### 4.1.5.2 Principales características

Las características relevantes de Tomcat son:

- Corre como un servicio (Windows) ó demonio (Linux, UNIX).
- Puede funcionar como web server por sí mismo o integrado dentro de Apache.
- Provee herramientas de administración de la aplicación: **deploy** (proceso de instalación para la ejecución de los servlets en los servidores), **undeploy** (proceso de desinstalación en los servidores), parar e iniciar las aplicaciones.
- Maneja múltiples aplicaciones (sites).
- Todos sus componentes son configurables.
- Tiene jerarquía anidada de componentes.

#### 4.1.6 Requerimientos para desarrollo del Sistema.

##### 4.1.6.1 Requerimientos mínimos de Hardware

Los requerimientos mínimos para que la aplicación funcione de manera adecuada son:

- Pentium III a 800 MHz (Recomendado Pentium IV con velocidad mayor a 800 MHz por la demanda de recursos de varios procesos dentro del sistema).
- Disco duro de 20 Gb (Recomendado 40 Gb por el crecimiento de la información de la Base de Datos).
- 128 Mb de RAM (Recomendado 256 por la demanda de recursos de varios procesos dentro del sistema).

##### 4.1.6.2 Requerimientos de software

Para que la aplicación funcione dentro de la máquina donde será instalada necesitamos tener los siguientes programas instalados:

- Linux Knoppix/Debian con Kernel de Suse 2.4.20. El sistema es una combinación de dos distribuciones por las siguientes razones:

- Knoppix/Debian es una distribución de Linux que es muy fácil de instalar y configurar ya que reconoce automáticamente la mayoría del hardware en la PC
- El kernel de la distribución SUSE se usa debido a un emulador de binarios de SCO (Open Systems Software) llamado Linux A. B. I., el cual es necesario, ya que muchos programas y procesos dentro del sistema del restaurante lo necesitan para ejecutarse.
- El sistema de ventanas de esta distribución se denomina XFce 4 ya que es un sistema muy ligero y muy configurable.
- J2SE Versión 1.4
- Apache Tomcat 5.0.28 (Configurado para ser un servidor standalone (autónomo)).
- Postgresql Versión 7.4.5

## 4.2 CASOS DE USO.

Los casos de uso, son operaciones específicas que se realizan tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación de otro caso de uso.

El modelo de casos de uso, es un modelo de las funciones deseadas del sistema (casos de uso) y lo que lo rodea (actores) y se utiliza para capturar los requerimientos funcionales del sistema.

El propósito principal de este modelo es comunicar la funcionalidad y comportamiento del sistema al cliente y/o usuario final para que éste lo valide, además de impulsar el análisis, diseño, implementación y pruebas del sistema.

El modelo de casos de uso:

- Se usa para identificar
  - Los flujos funcionales que deben realizarse con el sistema.
  - Los roles de usuarios que deben interactuar con el sistema.
  - Las interfaces que debe tener el sistema hacia sistemas externos.
- Se usa para verificar
  - Que todos los requerimientos funcionales se han capturado.
  - Que todos los desarrolladores han entendido estos requerimientos.
- Facilita la comunicación con los usuarios finales y expertos del dominio.
  - Proporciona aceptación desde las primeras etapas de desarrollo del sistema.



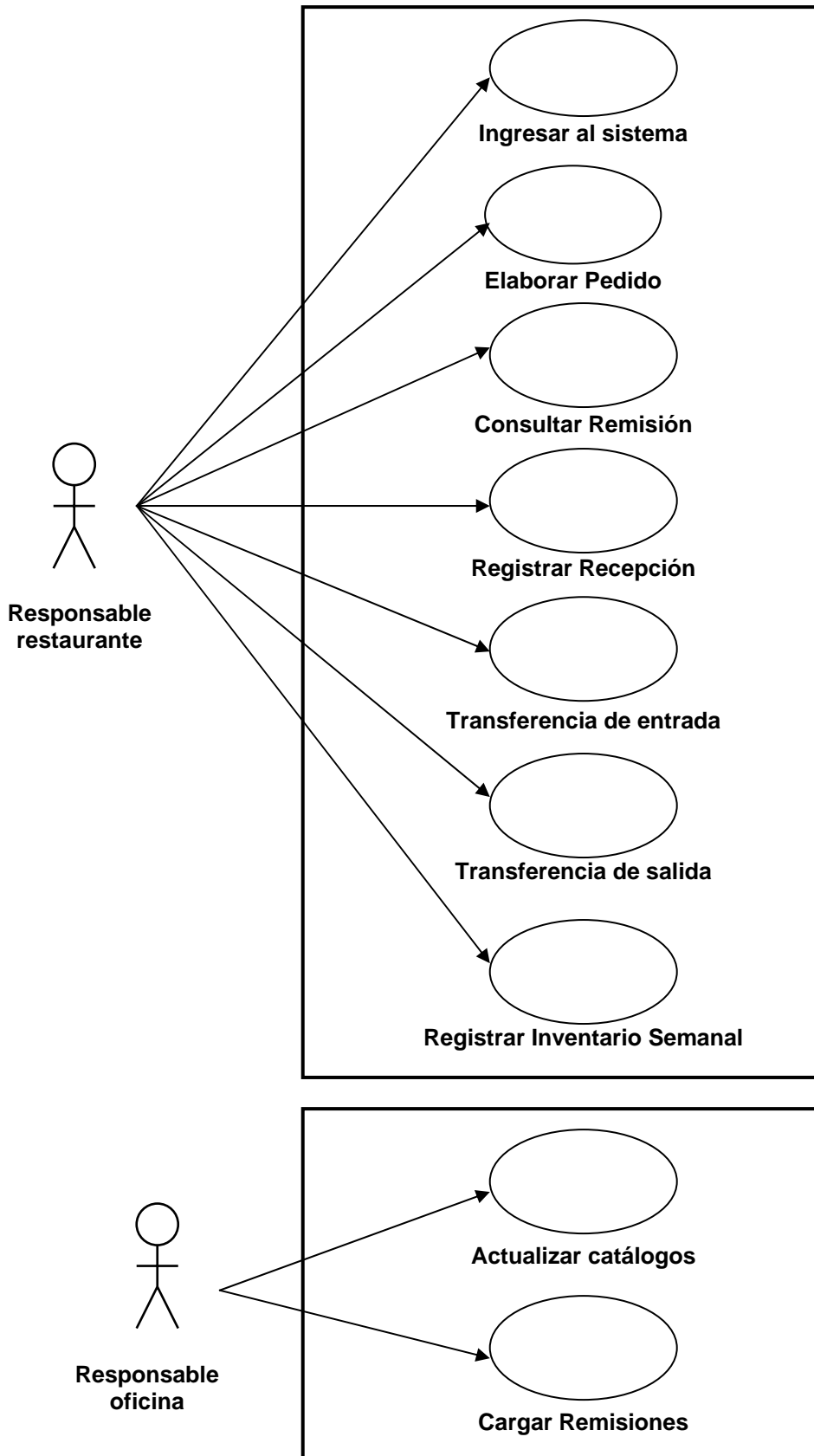
#### 4.2.1 Detección de Actores.



#### 4.2.2 Detección de Casos de Uso – CU.

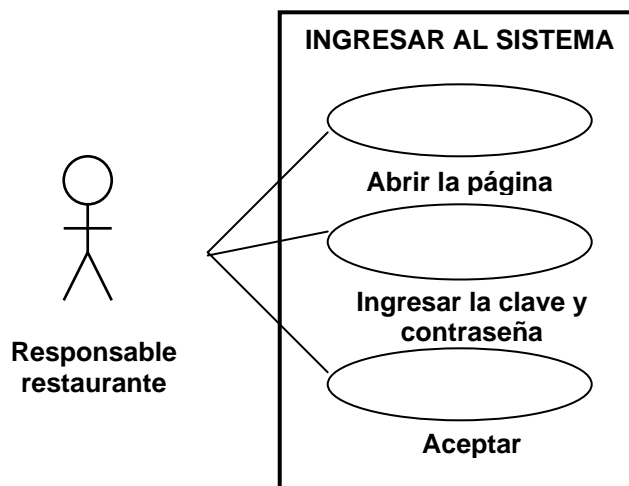
- Ingresar al sistema
- Elaborar Pedido
- Consultar Remisión
- Registrar Recepción
- Transferencia de Entrada
- Transferencia de Salida
- Registrar Inventario Semanal
- Actualizar Catálogos
- Cargar Remisiones

### 4.2.3 Diagramas generales de Casos de Uso.

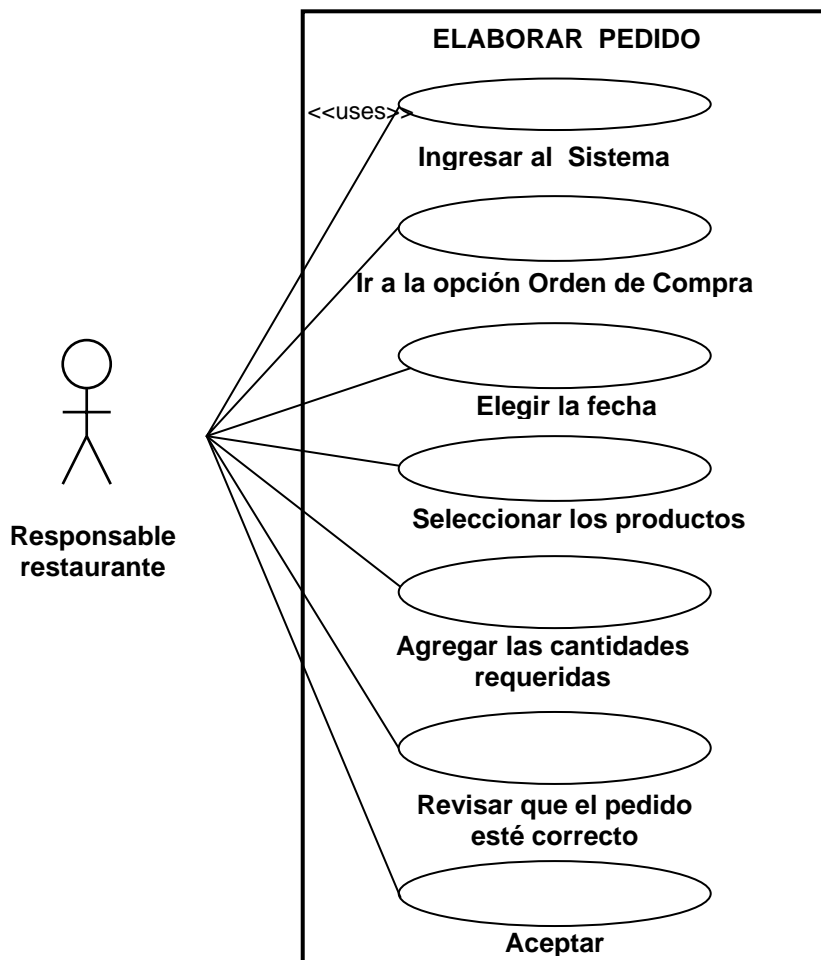


#### 4.2.4 Documentación de Casos de Uso

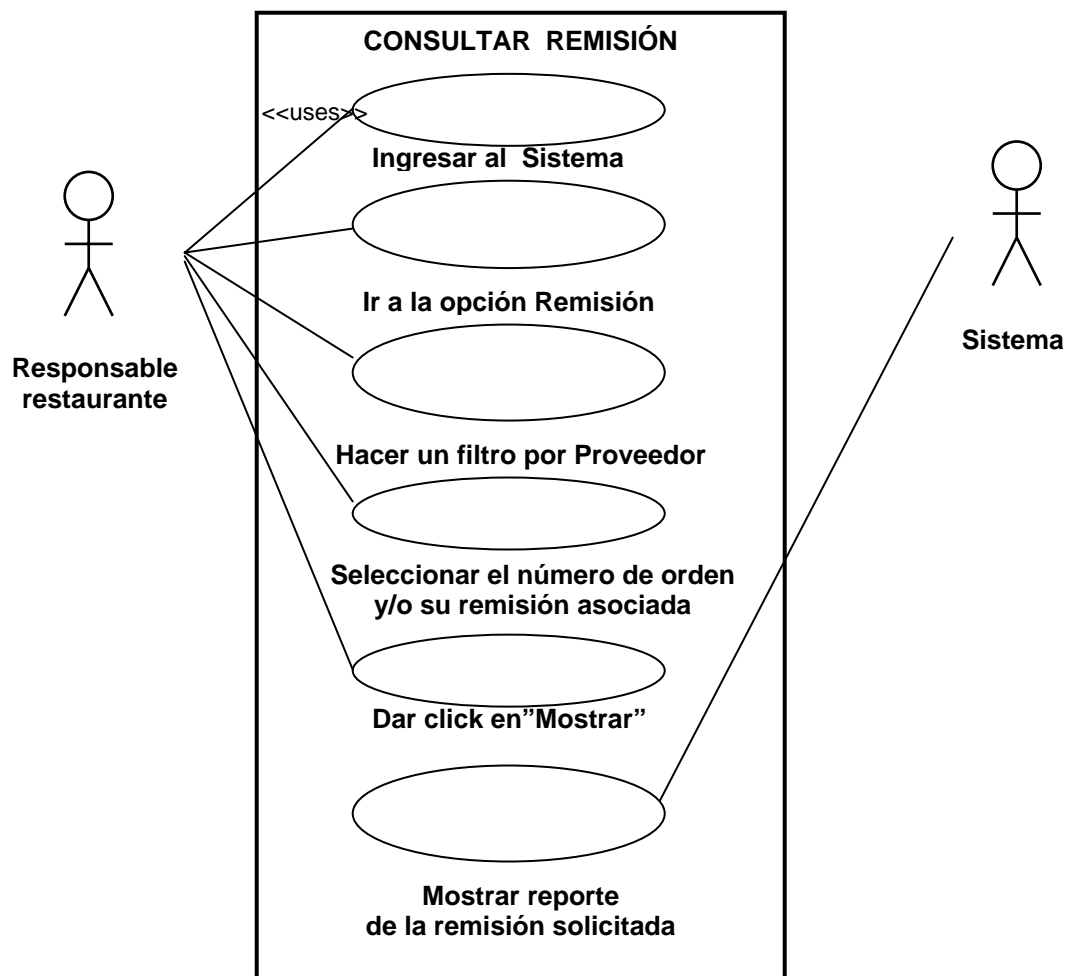
<b>Caso de uso:</b>	Ingresar al sistema
<b>Actores:</b>	Responsable restaurante
<b>Propósito:</b>	El responsable de oficina ingresa al sistema con la finalidad de realizar un pedido, o consultar alguna remisión, o realizar los registros de recepción o inventarios.
<b>Descripción:</b>	<ol style="list-style-type: none"><li>1. Dar doble click sobre el ícono del sistema que se encuentra en el escritorio, esto abrirá la página de la aplicación.</li><li>2. Se mostrará una pantalla en la que deberá ingresar <i>Usuario y Contraseña</i>.</li><li>3. Dar click en el botón de Aceptar.</li></ol>



<b>Caso de uso:</b>	Elaborar pedido
<b>Actores:</b>	Responsable restaurante.
<b>Propósito:</b>	El pedido se elabora para surtir al restaurante con materia prima e insumos suficientes para su correcta operación.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. <i>Ingresar al sistema.</i></li> <li>2. Ir a la opción Orden de Compra.</li> <li>3. Elegir la fecha de la cual se desea generar el pedido sugerido.</li> <li>4. Elegir los productos que desea ordenar.</li> <li>5. Agregar las cantidades requeridas de los productos deseados.</li> <li>6. Revisar que el pedido sea correcto.</li> <li>7. Aceptar la operación.</li> </ol>



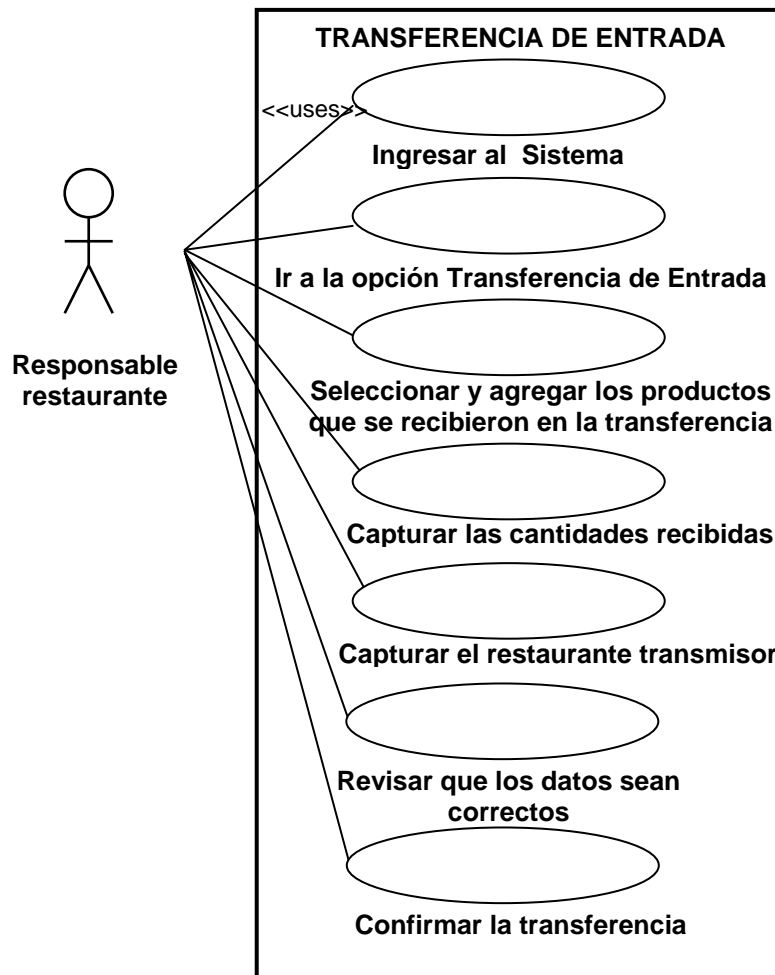
<b>Caso de uso:</b>	Consultar remisión
<b>Actores:</b>	Responsable restaurante, Sistema
<b>Propósito:</b>	Comparar la orden del pedido y la remisión que el proveedor teóricamente surtirá, de tal modo que se prevenga la obtención de los insumos críticos requeridos para la venta, mediante transferencias de los restaurantes más cercanos.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. <i>Ingresar al sistema.</i></li> <li>2. Ir a la opción "Remisión"</li> <li>3. Hacer un filtro por Proveedor.</li> <li>4. Seleccionar el número de orden y/o su remisión asociada.</li> <li>5. Dar click en "Mostrar".</li> <li>6. El Sistema generará y mostrará el reporte de la remisión, en el cual estarán resaltadas las diferencias con un color diferente</li> </ol>



<b>Caso de uso:</b>	Registrar recepción
<b>Actores:</b>	Responsable restaurante, Sistema
<b>Propósito:</b>	Registrar los productos realmente recibidos de una remisión, añadiendo o eliminando productos según el caso.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. <i>Ingresar al sistema.</i></li> <li>2. Ir a la opción “Recepción”.</li> <li>3. Seleccionar Proveedor.</li> <li>4. Se pueden hacer cuatro tipos de recepciones: <ol style="list-style-type: none"> <li>a) Con orden y remisión.</li> <li>b) Con orden sin remisión.</li> <li>c) Con remisión sin orden.</li> <li>d) Sin orden ni remisión.</li> </ol> </li> <li>5. Para los tres primeros tipos se debe seleccionar el número de orden y/o remisión correspondiente y dar click en el botón “Mostrar”.</li> <li>6. Para el último tipo (caso d) deberá dar click en el botón de “Agregar Productos” y seleccionar los productos recibidos.</li> <li>7. Modificar las cantidades de los registros de acuerdo a lo recibido físicamente.</li> <li>8. En el caso de haber recibido productos que no se solicitaron en la remisión agregarlos.</li> <li>9. Dar click en el botón de “Actualizar”. Se muestra la ventana de Confirmación de la Recepción.</li> <li>10. Capturar número de remisión (si se cuenta con él) y nombre de quien captura.</li> <li>11. Dar click en el botón “Aceptar”.</li> <li>12. El Sistema generará y mostrará el reporte de cómo quedaría finalmente la recepción.</li> <li>13. Revisar los datos registrados.</li> <li>14. Confirmar la recepción.</li> <li>15. Consultar el “Reporte de Diferencias”.</li> <li>16. El Sistema generará y mostrará el “Reporte de Diferencias”.</li> </ol>

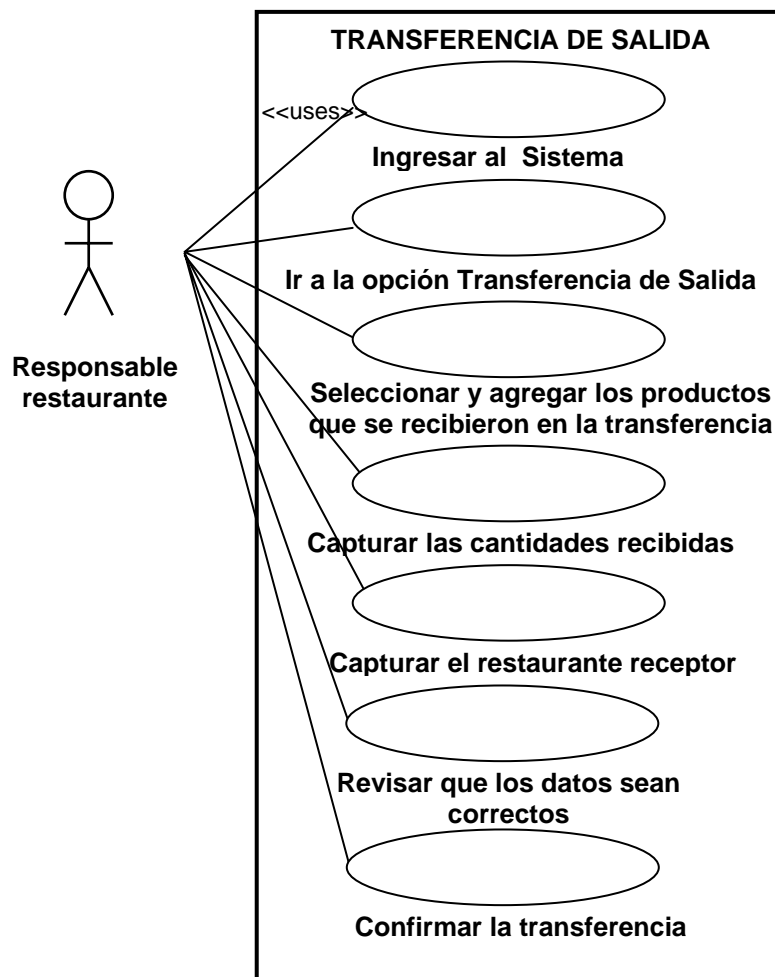


<b>Caso de uso:</b>	Transferencia de Entrada
<b>Actores:</b>	Responsable restaurante
<b>Propósito:</b>	Registrar en el inventario los productos recibidos por medio de una transferencia de otro restaurante.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Ingresar al sistema.</li> <li>2. Ir a la opción "Transferencias de Entrada".</li> <li>3. Seleccionar y agregar los productos que se van a registrar.</li> <li>4. Capturar las cantidades de acuerdo a lo recibido.</li> <li>5. Capturar el restaurante del cual se recibió la transferencia.</li> <li>6. Dar click en el botón "Aceptar".</li> <li>7. Revisar que los datos registrados sean correctos.</li> <li>8. Confirmar la transferencia.</li> </ol>

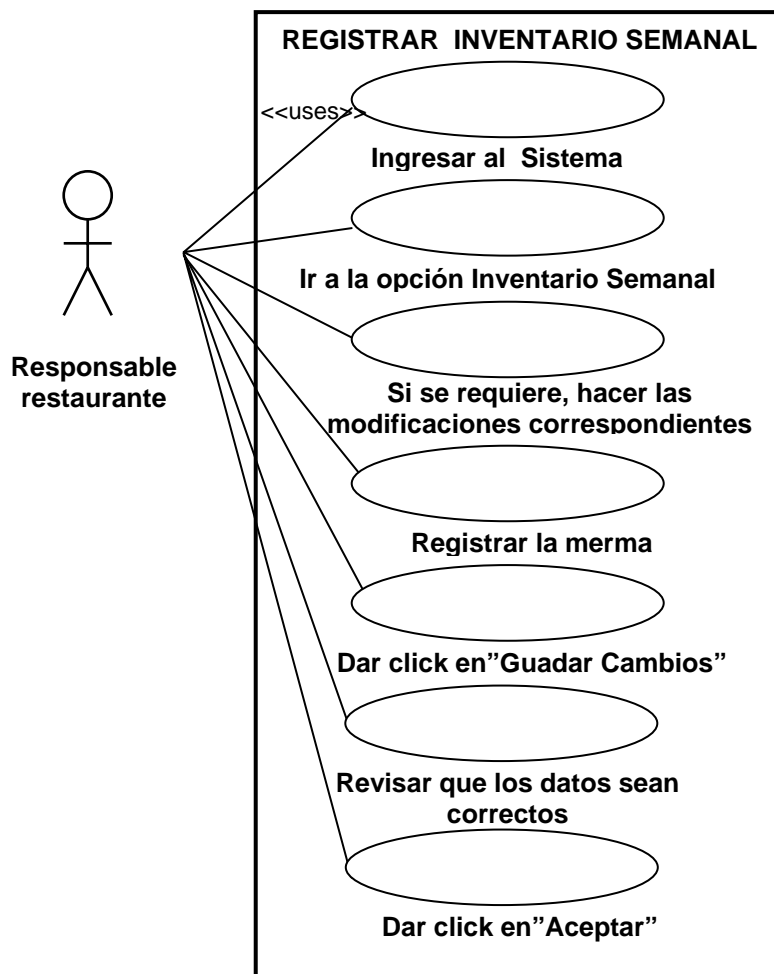




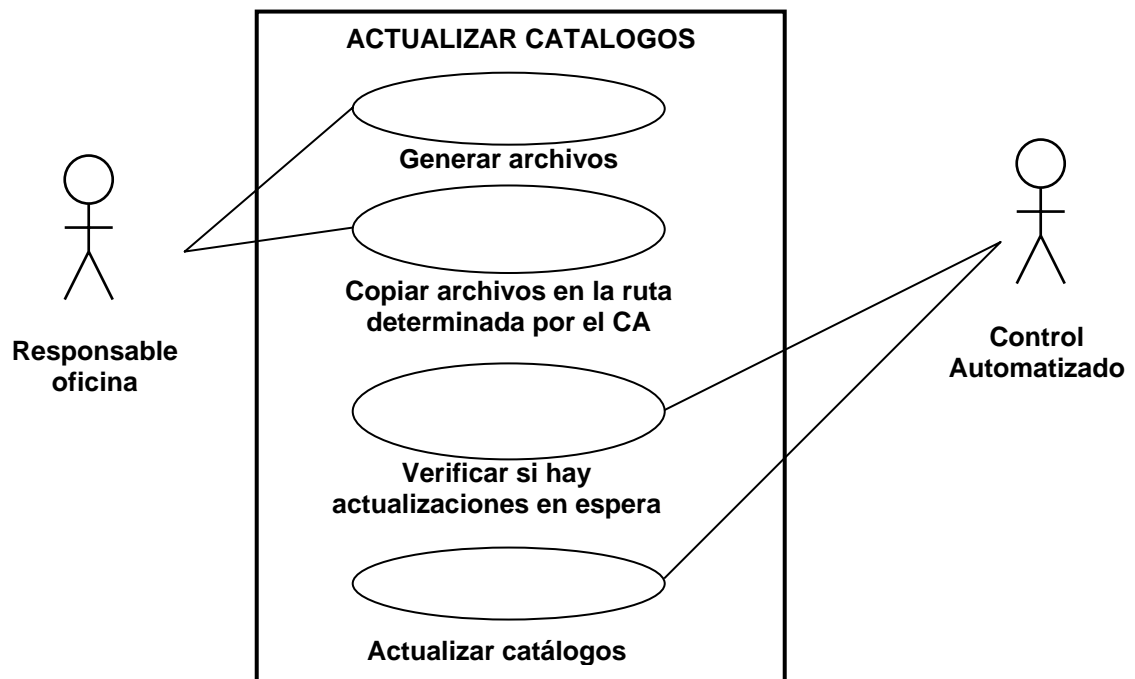
<b>Caso de uso:</b>	Transferencia de Salida
<b>Actores:</b>	Responsable restaurante
<b>Propósito:</b>	Descargar en el inventario la salida de los productos transferidos a otro restaurante.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. Ingresar al sistema.</li> <li>2. Ir a la opción "Transferencias de Salida"</li> <li>3. Seleccionar y agregar los productos que se van a transferir.</li> <li>4. Capturar las cantidades de acuerdo a lo transferido.</li> <li>5. Capturar el restaurante al cual se hará la transferencia.</li> <li>6. Dar click en el botón "Aceptar"</li> <li>7. Revisar que los datos registrados sean correctos.</li> <li>8. Confirmar la transferencia.</li> </ol>



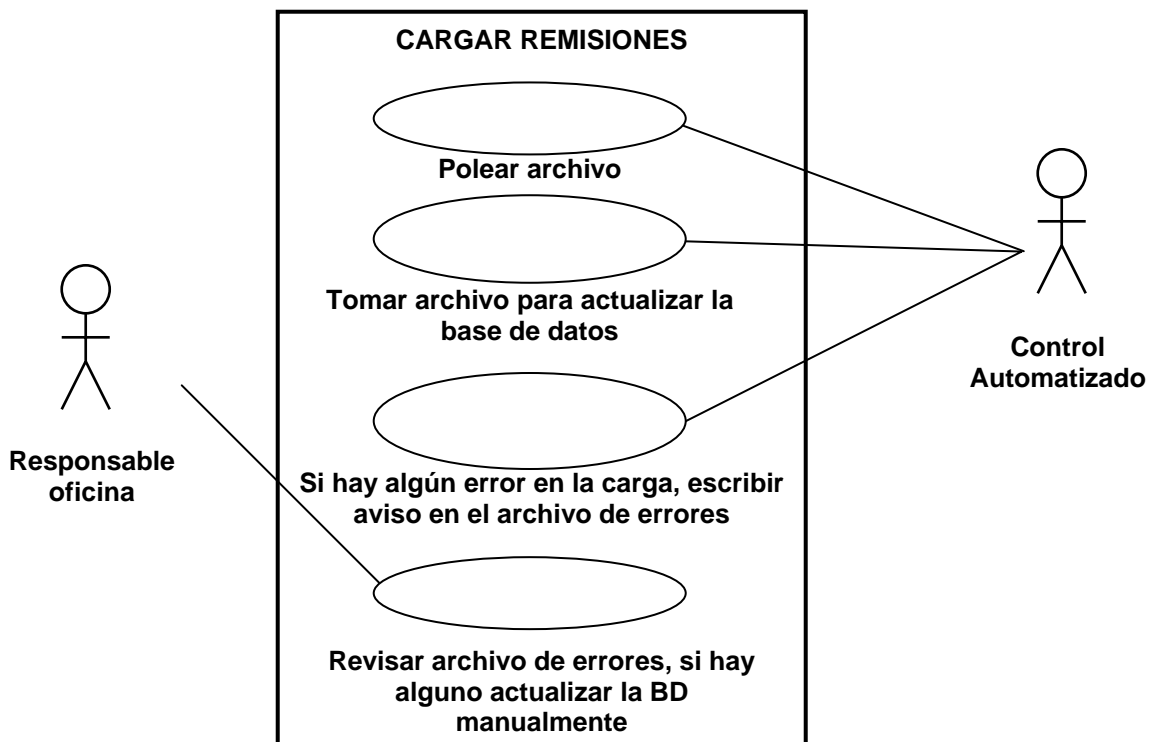
<b>Caso de uso:</b>	Registrar inventario semanal
<b>Actores:</b>	Responsable restaurante
<b>Propósito:</b>	Capturar la cantidad de inventario final de la semana, a partir de los movimientos realizados dentro del sistema: recepciones, transferencias de entrada y de salida.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1. <i>Ingresar al sistema</i></li> <li>2. Ir a la opción Inventario semanal.</li> <li>3. Modificar las cantidades de acuerdo a como quedaría el inventario final.</li> <li>4. Registrar la Merma.</li> <li>5. Dar click en el botón "Guardar cambios".</li> <li>6. Revisar si los datos son correctos.</li> <li>7. Confirmar el registro de inventario final dando click en el botón "Aceptar".</li> </ol>



<b>Caso de uso:</b>	Actualizar catálogos
<b>Actores:</b>	Responsable oficina, Control automatizado
<b>Propósito:</b>	Mantener actualizados los catálogos en los casos en que cambien los precios de los productos, se agreguen o desactiven proveedores o bien cambios en los productos.
<b>Descripción:</b>	<ol style="list-style-type: none"> <li>1) Generar los archivos que contienen los cambios de los catálogos.</li> <li>2) Copiar dichos archivos en la ruta de donde los tomará el Control automatizado (CA).</li> <li>3) El CA revisará diariamente a una hora específica (que será fuera del horario de operación del restaurante) si existen actualizaciones en espera.</li> <li>4) Si detecta que hay actualizaciones pendientes iniciará el proceso de actualización de catálogos en cada restaurante.</li> <li>5) Si la actualización no puede llevarse a cabo o es interrumpida por cualquier circunstancia, se generará un aviso para que se actualice manualmente.</li> </ol>



<b>Caso de uso:</b>	Cargar Remisiones
<b>Actores:</b>	Responsable oficina, Control automatizado
<b>Propósito:</b>	Actualizar la base de datos con los archivos de las remisiones del proveedor.
<b>Descripción:</b>	<p>1) El Control Automatizado (CA) polea los archivos del restaurante a la oficina</p> <p>2) El CA se activa a una hora especifica para realizar la carga en la base de datos</p> <p>3) El CA va leyendo cada archivo poleado y lo va insertando en la Base de Datos</p> <p>4) Si la carga no puede llevarse a cabo o es interrumpida por cualquier circunstancia, se generará en un archivo un aviso para que se realice la carga manualmente.</p>



## 4.3 DISEÑO DE LA BASE DE DATOS

### 4.3.1 Puntos considerados.

Con la finalidad de que el software desarrollado fuera totalmente funcional, se estudiaron detalladamente los requerimientos de los usuarios y los productos específicos que se esperan del sistema. Entre los factores que se encontraron de importancia para el diseño de la base de datos para el control de órdenes de compra e inventarios, se listan los siguientes:

- Número de registros almacenados en el sistema.
- Número estimado de transacciones.
- Número de usuarios con acceso al sistema.
- Información en línea actualizada.
- Qué equipo va a utilizar el sistema.
- Si el sistema va a operar en red.
- Clasificación y manejo de información.
- Elegir una plataforma de programas de cómputo apropiada para la tarea.
- Asegurar que el equipo utilizado tenga el suficiente poder de procesamiento y velocidad para optimizar el desempeño de la base de datos.
- Utilizar un diseño de base de datos relacional para optimizar su flexibilidad.
- Evitar el almacenaje de información repetida, empleando la normalización de la BD.
- Usar estándares de programación que sigan las convenciones de la industria, de tal forma que una persona distinta al programador original pueda corregir, modificar, actualizar o auditar la base de datos.
- Documentar la base de datos para los programadores, incluyendo descripciones de los códigos o campos en el diseño de la base de datos (Diccionario de datos).
- Utilizar nombres convencionales para todos los objetos incluidos en la base de datos (como tablas, consultas, reportes, formatos, etc.)
- Utilizar códigos de programación de diseño modular (es decir, conformados por módulos que puedan ser probados y evaluados por separado).

- Utilizar índices e identificadores únicos para acelerar las búsquedas y permitir ligar tablas de datos.

Uno de los puntos más importantes en el diseño fue considerar las necesidades en el manejo de información. Por ejemplo, jamás se borra información de los catálogos acerca de los productos en el inventario, simplemente se modifica un atributo para indicar que el producto ya no existe, esto con el fin de tener un registro completo, el cual podría ser utilizado en una auditoria con fines estadísticos. Todos esos aspectos a considerar fueron detalladamente analizados.

La base de datos de este proyecto de Tesis es relacional y por lo tanto el diseño conceptual es el modelo entidad – relación.

#### **4.3.2 Diseño de la BD, empleando el modelo entidad-relación.**

El modelo conceptual utilizado es el modelo entidad-relación, que posee los siguientes conceptos: entidades, relaciones, atributos (Ver Capítulo 2, sección 2.1.3)

Las tareas realizadas para lograr dicho modelo fueron:

##### **4.3.2.1 Identificar las entidades.**

Al examinar las especificaciones de requisitos de usuario, se buscaron los nombres de las entidades (por ejemplo: catálogo de productos, catálogo de familias, catálogo de proveedores, etc.)

Entity name	Description
op_grl_cat_config_difference	Tabla que almacena el orden en que deben desplegarse en la lista de código de discrepancia dependiendo el caso de la recepción.
op_grl_cat_confirmation	Catálogo de los tipos de confirmación. Solo puede almacenar los siguientes valores: 1. Electrónica 2. Telefónica 3. Fax 4. E - mail
op_grl_cat_difference	Catálogo con los códigos de discrepancia disponibles al hacer una recepción.
op_grl_cat_family	Catálogo de las familias en las que se clasifican los productos de inventario.
op_grl_cat_frecuency	Catálogo de las frecuencias en las que se debe inventariar un producto de inventario. Los valores que puede tener: 1 - Diario 2 - Semanal 3 - Periódico 4 - No se inventaria

op_grl_cat_inventory	Catálogo de los productos de inventario.
op_grl_cat_product_company	Catálogo de qué producto se aplica a cada marca.
op_grl_cat_provider	Catálogo de proveedores disponibles.
op_grl_cat_providers_product	Catálogo de los productos asociados a cada proveedor.
op_grl_cat_recep_cases	Catálogo de los casos de recepción que se pueden llevar a cabo. Sus valores son: 1 - Sin orden y sin remisión 2 - Sin orden y con remisión 3 - Con orden y sin remisión 4 - Con orden y con remisión 5 - Nuevos elementos 6 - Diff ord/recep no rem 7 - Dif ord/recep inex
op_grl_cat_unit_measure	Catalogos de unidades de medida, tanto de proveedor como de inventario.
op_grl_days_advance_order	Tabla de configuración para conocer el número de días que es permitido adelantar un pedido dentro de un restaurante.
op_grl_difference	Tabla para almacenar las diferencias entre orden y recepción encontradas. Los registros se obtienen al confirmar la recepción.
op_grl_document_type	Catálogo con los tipos de documento a recibir. Tiene los valores de : 1 - Remisión 2 - Factura
op_grl_order	Tabla que almacena los pedidos realizados, ya confirmados.
op_grl_order_detail	Tabla que almacena los productos que se requieren en un pedido al confirmarlo. Parte de la llave primaria de esta tabla.
op_grl_order_limit	Tabla que almacena los días límite para que PFS reciba el pedido del restaurante.
op_grl_reception	Tabla que registra una recepción de producto cuando ésta es confirmada.
op_grl_reception_detail	Tabla que almacena los datos de los productos que se registran en la recepción después de ser confirmada.
op_grl_remission	Tabla que almacena a remisión de los productos a recibir por parte del restaurante.
op_grl_remission_detail	Tabla que almacena los productos que se tienen en la remisión.
op_grl_step_order	Tabla de paso cuando una orden aún no es confirmada. Esta tabla solo almacena una orden que es eliminada al momento de confirmarla.
op_grl_step_order_detail	Tabla que almacena los productos que se desea ordenar, pero solo temporalmente hasta que se confirme el pedido.
op_grl_step_reception	Tabla que almacena temporalmente los datos de una recepción, antes de que sea confirmada.
op_grl_step_reception_detail	Tabla que almacena de manera temporal los productos que se registran en la recepción, antes de ser confirmada.
op_grl_suggested_order	Tabla que almacena el pedido sugerido generado desde la aplicación, este pedido sugerido es el mismo que se genera dentro de FMS.
op_grl_way_order	Tabla que almacena la información de los productos que se encuentran en tránsito para recalcular el pedido sugerido.
op_grl_existence	Tabla que almacena la existencia actual de un producto en el inventario.
op_grl_financial_mov.	Tabla que almacena información financiera de ventas, transacciones, gastos, depósitos, movimientos de caja, etc.
op_grl_step_transfer	Tabla que almacena temporalmente los datos de una transferencia, antes de que sea confirmada
op_grl_step_transfer_detail	Tabla que almacena de manera temporal los productos que se registran en la transferencia, antes de ser confirmada
op_grl_suggested_transfer	Tabla que almacena los valores de pedido sugerido para la siguiente semana de la fecha en la que se invoque su carga.
op_grl_transfer	Tabla que almacena temporalmente los datos de una transferencia cuando ha sido confirmada.

op_grl_transfer_detail	Tabla que almacena de manera temporal los productos que se registran en la transferencia cuando ha sido confirmada.
op_inv_conversion_factor	Tabla que almacena los factores de conversión entre las unidades de proveedor y las de las recetas.
op_inv_exceptions	Tabla que almacena parámetros para determinar la eficiencia de un producto.
op_inv_inventory_begin	Tabla que almacena los valores del inventario inicial de la semana tomándolos del archivo de inventario.
op_inv_inventory_detail	Tabla que almacena la información relacionada con el inventario semanal cuando ya fueron guardados los cambios en el campo de inventario final
op_inv_inventory_mov	Tabla que almacena temporalmente los movimientos de recepciones y transferencias en el inventario hasta pasarlos a op_inv_inventory_detail.
op_inv_step_inventory_detail	Tabla que almacena la información relacionada con el inventario semanal cuando no han guardados los cambios en el campo de inventario final
ss_cat_yum_day	Tabla que almacena los días de la semana. Iniciando en martes.
ss_cat_time	Tabla que almacena toda la información relacionada con el calendario del restaurante (que no es el mismo que el normal) de forma anual.
ss_cat_neighbor_store	Tabla que almacena el catálogo de todos los restaurantes a los que se puede hacer ó se puede recibir una transferencia.

#### 4.3.2.2 Identificar los atributos de cada entidad.

Una vez definidas las entidades, se procede a asociar todos y cada uno de los atributos que serán importantes para cada una de ellas. Además se define el que será su atributo llave, el cual no puede repetirse, de manera que cada registro sea único dentro de una entidad. Además se aplican reglas de normalización para eliminar redundancia en la información.

#### 4.3.2.3 Identificar las relaciones.

Después se definen las relaciones existentes entre las entidades de manera que quedan definidas las llaves foráneas para cada entidad por medio de estas relaciones.

#### 4.3.2.4 Creación del diagrama entidad-relación.

Se dibuja el diagrama entidad-relación donde se verá claramente el diseño hasta el momento de la base de datos. El diagrama se dibuja a través de una herramienta especializada para esta labor y por medio de ella se genera el script



que sirve para la generación física de la Base de Datos y finalmente se genera el diccionario de datos con esta misma herramienta.

Tanto el Diagrama Entidad- Relación como el diccionario de datos se incluyen en el apéndice I de esta tesis.

# Capítulo V. Implementación.

- 5.1 Instalación de la aplicación.
- 5.2 Caso práctico.

### 5.1. Instalación de la aplicación

Para instalar la aplicación se crea un archivo empaquetado y comprimido con extensión .tar.gz donde se incluyen todos y cada uno de los archivos relacionados con la aplicación (incluyendo los de creación de la base de datos), incluyendo un Shell Script (Script de comandos de UNIX ó Linux) en el cual se llevarán a cabo los siguientes pasos:

1. Creación de los directorios de la aplicación.
  - a) /usr/local/tomcat/webapps/ROOT/CSS: En este directorio se almacenan las hojas de estilo de la aplicación.
  - b) /usr/local/tomcat/webapps/ROOT/Images: En este directorio se almacenan las imágenes de la aplicación.
  - c) /usr/local/tomcat/webapps/ROOT/Include: En este directorio se almacenan algunos programas que se incluyen en toda la aplicación.
  - d) /usr/local/tomcat/webapps/ROOT/applets: En este directorio se almacenan los programas que guardan el applet del calendario para hacer la consulta de los reportes por fecha.
  - e) /usr/local/tomcat/webapps/ROOT/Scripts: En este directorio se almacenan shell scripts que sirven para realizar algunos procesos alternativos como la generación de archivos de texto.
  - f) /usr/local/tomcat/webapps/ROOT/WEB-INF/classes/generals: En este directorio se almacenan las clases compiladas que corresponden al diseño de la plantilla de desarrollo de la aplicación.
  - g) /usr/local/tomcat/webapps/ROOT/WEB-INF/classes/generals/servlets: En este directorio se almacenan las clases que son los servlets de la plantilla de desarrollo de la aplicación.
  - h) /usr/local/tomcat/webapps/ROOT/WEB-INF/jinvtran: Este directorio almacena las clases de apoyo para el módulo de captura de inventario semanal.
  - i) /usr/local/tomcat/webapps/ROOT/Inventory: En este directorio se almacena los directorios de la aplicación desarrollada.

- j) `/usr/local/tomcat/webapps/ROOT/Inventory/PurchaseOrder`: En este directorio se almacenan los programas que componen el módulo de Ordenes de Compra (Este módulo se compone de la toma del pedido, consulta de la remisión e ingreso de la recepción).
- k) `/usr/local/tomcat/webapps/ROOT/Inventory/PurchaseOrder/Transfers`: En este directorio se almacenan los programas que componen el módulo de Transferencias.
- l) `/usr/local/tomcat/webapps/ROOT/Inventory/PurchaseOrder/WeeklyInventory`: En este directorio se almacenan los programas que componen el módulo de Inventario Semanal.
- m) Dentro de cada uno de los directorios (j, k, l) se crean los siguientes subdirectorios:
- `Entry`: Almacena el código JSP de las formas que requieran captura por parte del usuario.
  - `Proc.`: Almacena código JSP de uso general para la aplicación.
  - `Rpt`: Almacena código JSP de los reportes requeridos en el sistema.
  - `Scripts`: Almacena código de javascript de uso general para la aplicación.
- n) `/usr/bin/ph/databases`: Este directorio almacena los directorios por módulo para cargas de información en la base de datos como los catálogos, las remisiones, usos ideales, etc.
- o) `/usr/bin/ph/databases/purchase_order`: Este directorio almacena los scripts de carga de información para el módulo de Órdenes de Compra.
- p) `/usr/bin/ph/databases/transfers`: Este directorio almacena los scripts de carga de información para el módulo de Transferencias.
- q) `/usr/bin/ph/databases/inventory`: Este directorio almacena los scripts de carga de información para el módulo de Inventario Semanal
- r) Dentro de cada uno de los directorios (o, p, q) se crean los siguientes subdirectorios:
- `bin`: Almacena el código en Perl de la carga de información.

- dat: Almacena los archivos de texto con los que llevará a cabo la carga de información.
  - lib: Almacenan el código en Perl de las funciones de uso general para la carga de información.
  - log: Almacena los archivos de log generados para saber los estatus de las cargas.
- s) /usr/bin/ph/3w\_orden: Directorio donde se almacena los archivos de texto que representan la orden de compra y que serán poleados a oficinas.
- t) /usr/bin/ph/3w\_recepcion: Directorio donde se almacena los archivos de texto que representan la recepción y que serán poleados a oficinas.
- u) /usr/bin/ph/3w\_discrem: Directorio donde se almacena los archivos de texto que representan las diferencias entre la remisión y la recepción y que serán poleados a oficinas.
- v) /usr/bin/ph/3w\_discord: Directorio donde se almacena los archivos de texto que representan las diferencias entre la orden y la recepción y que serán poleados a oficinas
2. Copia de todos los archivos en los directorios que les correspondan
  3. Cambio de propietario de los archivos recién copiados
  4. Cambio de permisos en los archivos recién copiados
  5. Ejecución de los archivos de creación y/o modificación de la base de datos.
  6. Ejecución de los scripts de carga de catálogos a la BDD
  7. En todo el proceso se crea un archivo de log para poder dar seguimiento a la instalación y para ver si fue llevada a cabo con éxito.

Listo el directorio empaquetado, se envía por poleo a los restaurantes en donde se tiene la orden de desempaquetar el directorio y ejecutar todo lo que este en el Shell Script de instalación. Finalmente se envía a oficinas el archivo de log generado por el Shell de instalación para conocer el estatus de la misma.

## 5.2 Caso práctico

La siguiente pretende ser una guía para mostrar la forma en que quedó la implementación del sistema y los pasos básicos sobre los diferentes procedimientos que se pueden llevar a cabo dentro de él.

La figura 5.2.1 muestra la pantalla de inicio del sistema. Para poder ingresar al sistema se debe dar clic en donde dice “Entrar”.

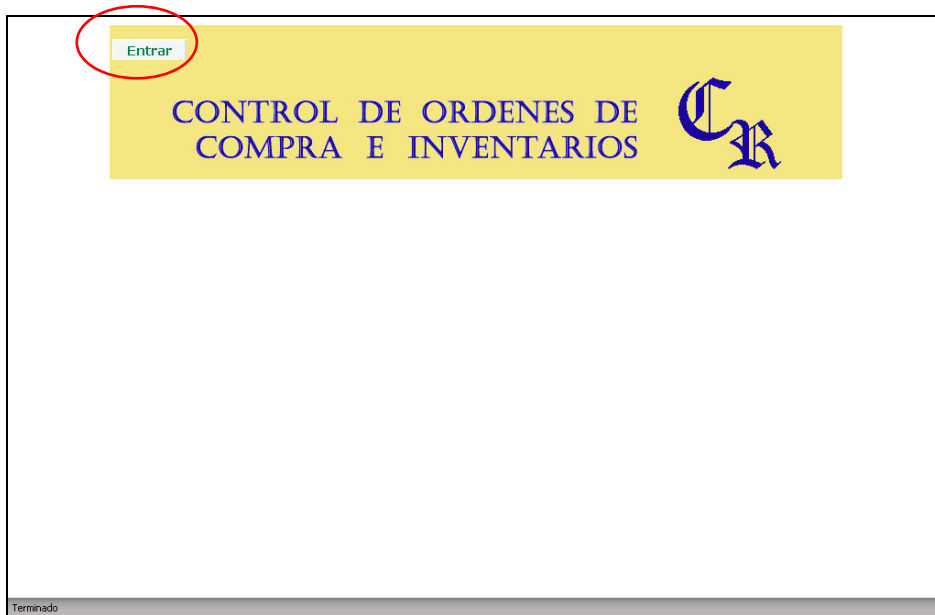


Figura 5.2.1 Pantalla de Inicio

El siguiente paso es proporcionar el Usuario y Contraseña y se da clic en “Accesar” para entrar a la aplicación, como se muestra en la figura 5.2.2.

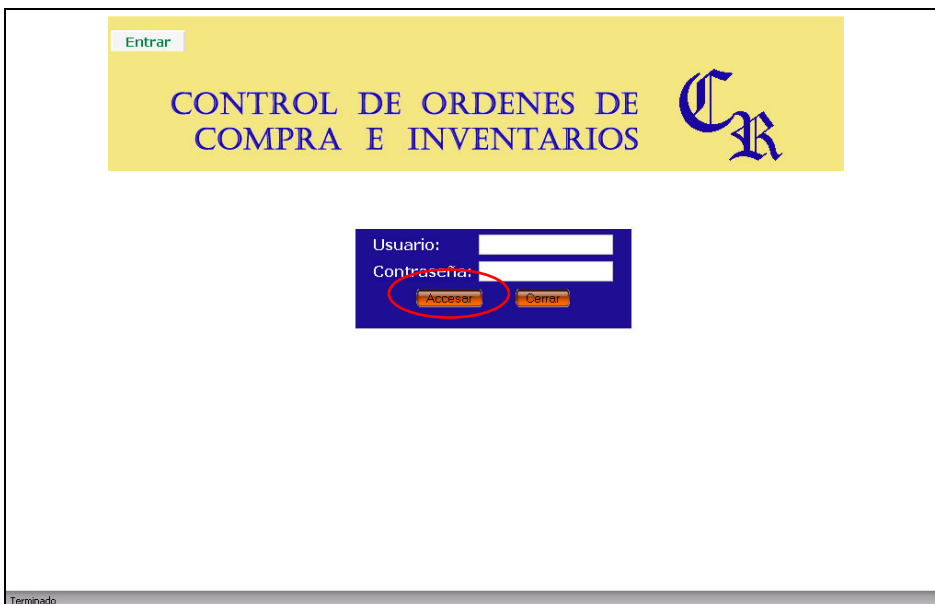


Figura 5.2.2 Pantalla de Acceso

Si el Usuario o Contraseña son incorrectos se mostrará un mensaje de error como el de la figura 5.2.3.

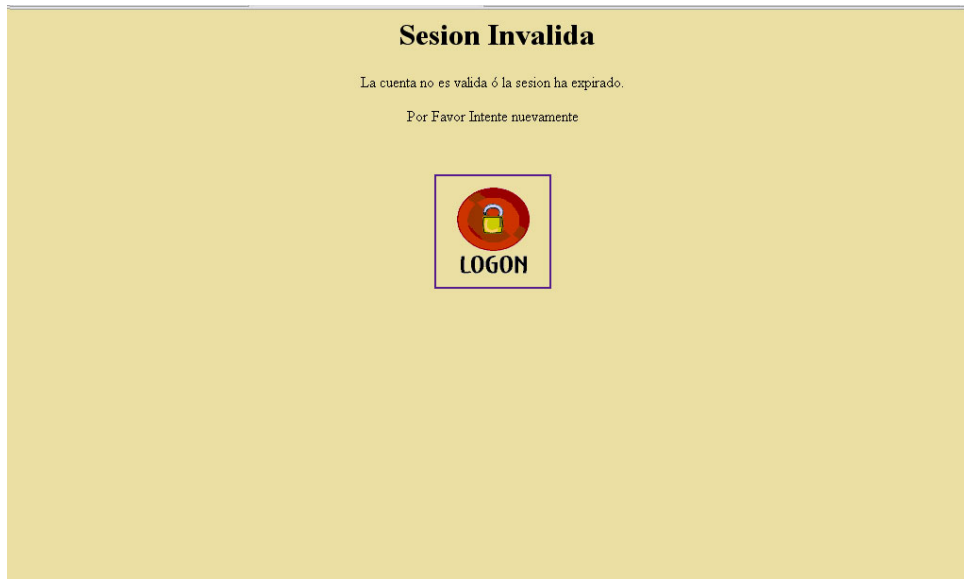


Figura 5.2.3 Pantalla de Error en contraseña o Usuario

Una vez que se haya ingresado al sistema aparece la pantalla Principal dando la bienvenida y muestra indicaciones a seguir. Como se muestra en la figura 5.2.4



Figura 5.2.4 Pantalla de Bienvenida

Al dar clic en el botón Izquierdo que dice "Inventario" se desplegarán las opciones del menú principal, como se observa en la figura 5.2.5.

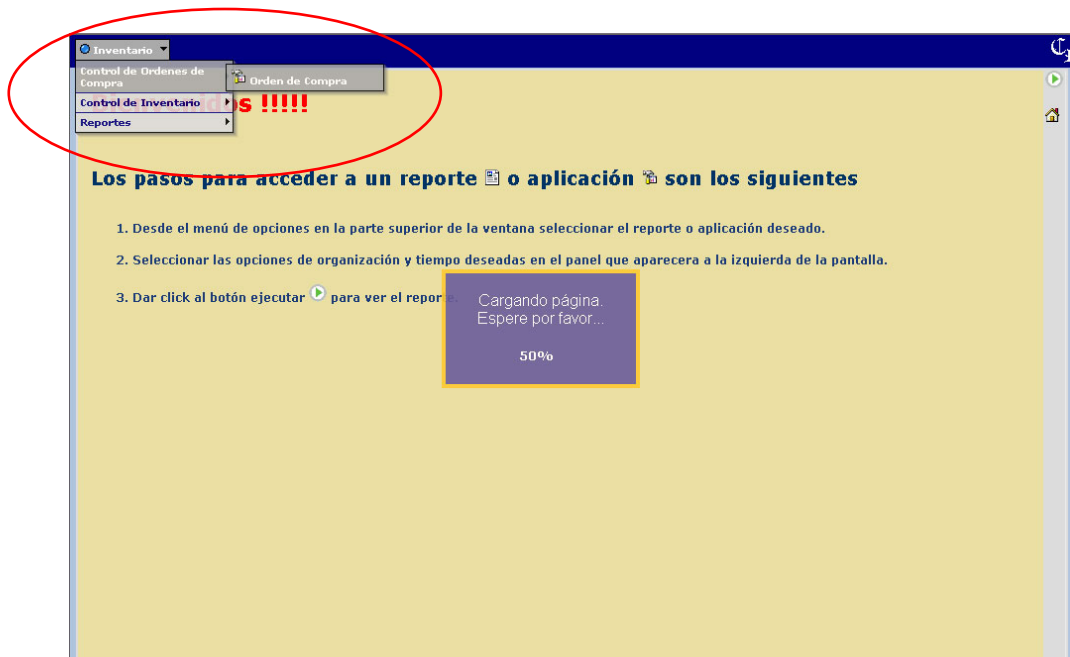


Figura 5.2.5 Pantalla Inicial

Al dar clic en la opción “Control de Orden de Compra” se muestran 4 apartados: Pedido, Remisión, Recepción y Consultas. Como se puede ver en la figura 5.2.6.

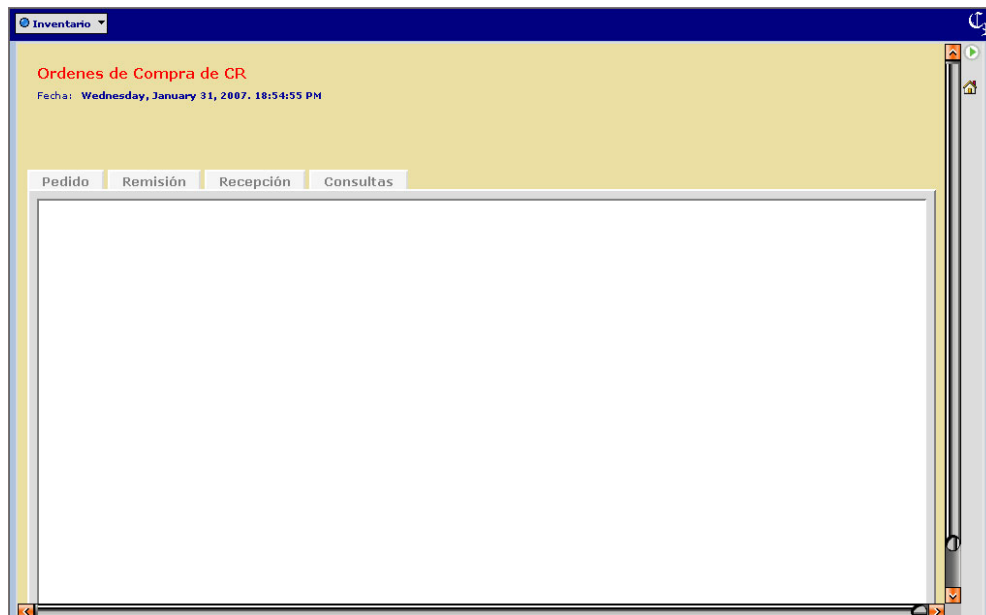


Figura 5.2.6 Control de Órdenes de Compra.

### 5.2.1 PEDIDO

Para elaborar un pedido dar clic en la pestaña de Pedido. Aparecerá en la pantalla un calendario en el cual se deberá elegir la fecha en la que será calculado el pedido sugerido tal como se muestra en la figura 5.2.7.





Figura 5.2.7 Generación de pedido.

Una vez seleccionada la fecha dar clic en el botón Generar.

Para continuar dar clic en el botón Agregar Productos. En esta pantalla se seleccionarán los productos que se desean ordenar, estos pueden ser filtrados por proveedor, si se desea, tal como se observa en la figura 5.2.8.

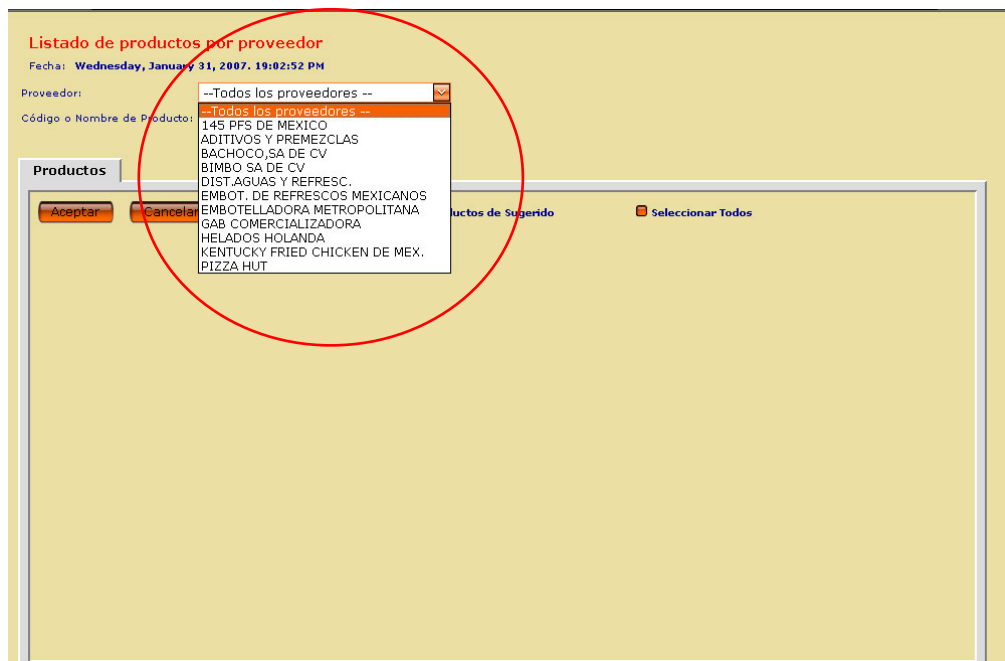


Figura 5.2.8 Selección de proveedor

Para seleccionar los productos deseados se debe dar clic en el checkbox (casilla de verificación) que se encuentra al inicio de cada renglón. También hay una opción para elegir todos los productos del pedido sugerido, lo cual cambia a color rosa todos los productos donde haya una cantidad diferente de cero. Si se desea descartar algún producto dar clic sobre la casilla activa para desactivar el checkbox.

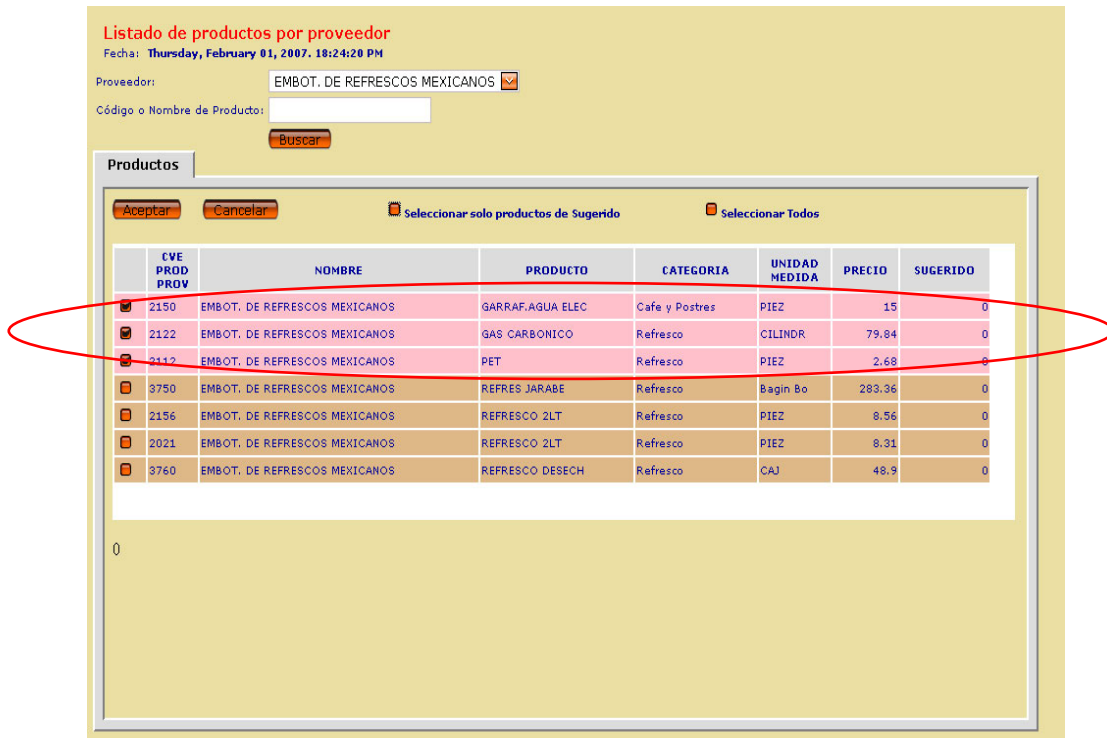


Figura 5.2.9 Productos seleccionados.

Al seleccionar los productos estos se mostrarán en la pantalla de Órdenes de Compra de CR en la cual se deberán agregar las cantidades deseadas de cada producto. Se puede capturar en unidades de inventario o bien en unidades de presentación. Una vez concluida esta captura dar clic en aceptar para que aparezca la pantalla de revisión y confirmación como se muestra en las figura 5.2.10 y figura 5. 2.11

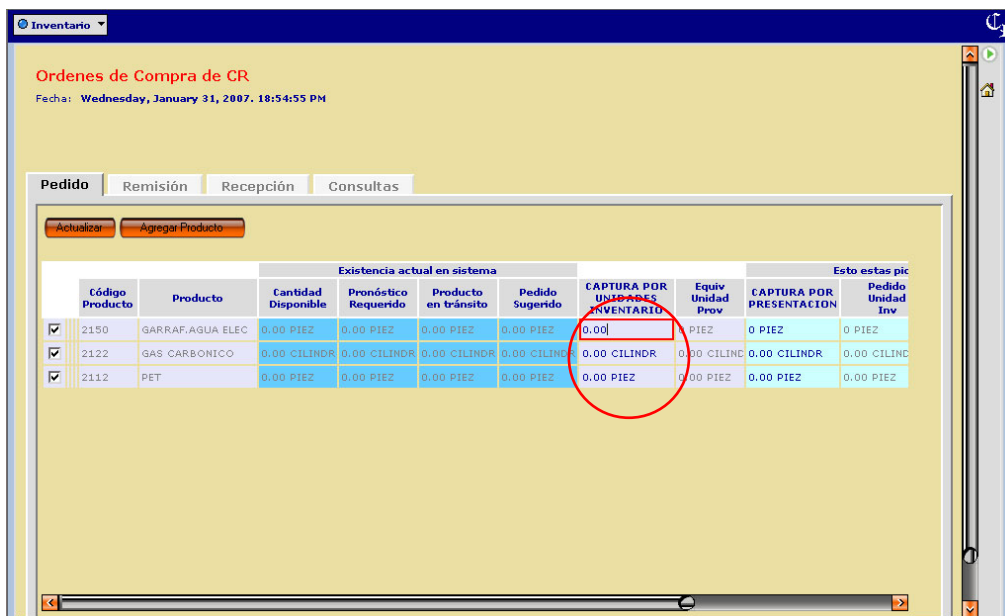


Figura 5.2.10 Pantalla principal de toma de pedido.

Al terminar de capturar las cantidades de producto que se desea ordenar, se da clic en el botón Actualizar para que se muestre la figura 5.2.11 que sirve para revisar que el pedido realizado sea correcto. Si se desea modificar algo en cantidades ó productos se da clic en el botón Cancelar.

**Revisión de pedido**  
 Fecha: Friday, January 26, 2007. 17:16:16 PM  
 Centro de contribución: 240 - VALLADOLID  
 Número de pedido: 10096  
 Desea confirmar el pedido?

Codigo Producto	Producto	Proveedor	Datos del Sistema				Esto estas Pidiendo					Dif vs sug	Costo
			Cantidad Disponible	Pronóstico Requerido	Producto en tránsito	Sugerido	Cantidad Requerida	Equiv Unidad Prov	Pedido Unidad Prov	Pedido Unidad Inv			
33819	PURE DE PAPA	145 PFS DE MEXICO	69.35	84.29	0.00	15.00	15.00 KILOGRAM	0.98 CAJ	1.00 CAJ	16.00 KILOGRAM	0.28	295.1	
33830	SAZONADOR E.T.C	145 PFS DE MEXICO	10.74	12.54	0.00	2.00	2.00 PAQUET	0.08 CAJ	1.00 CAJ	24.00 PAQUET	22.00	381.4	
33942	MARGARINA	145 PFS DE MEXICO	108.67	126.38	0.00	18.00	18.00 PIEZ	0.25 CAJ	1.00 CAJ	72.00 PIEZ	54.00	108.4	
33963	PREM.BISCU AWO	145 PFS DE MEXICO	42.88	67.20	0.00	25.00	25.00 BOLS	4.17 CAJ	5.00 CAJ	30.00 BOLS	5.00	760.5	
33966	PAN MINICRUNCH	145 PFS DE MEXICO	128.00	232.69	0.00	105.00	105.00 PIEZ	0.66 CAJ	1.00 CAJ	160.00 PIEZ	55.00	239.5	
34055	POPCORN CHICKEN	145 PFS DE MEXICO	21.21	26.70	0.00	6.00	6.00 BOLS	0.75 CAJ	1.00 CAJ	8.00 BOLS	2.00	732	
34057	JITOMATE	145 PFS DE MEXICO	32.65	48.07	0.00	16.00	16.00 KILOGRAM	1.45 CAJ	2.00 CAJ	22.00 KILOGRAM	6.00	349.6	
34184	PLATO	145 PFS DE MEXICO	0.81	2.11	0.00	2.00	2.00 CAJ	2.00 PAQUET	2.00 PAQUET	2.00 CAJ	0.00	389.1	
38084	MAGNESOL	145 PFS DE MEXICO	40.00	57.67	0.00	18.00	18.00 PIEZ	0.20 CAJ	1.00 CAJ	90.00 PIEZ	72.00	719.1	
38713	CJ.PAPA GD	145 PFS DE MEXICO	330.00	651.94	0.00	322.00	322.00 PIEZ	0.16 CAJ	1.00 CAJ	2000.00 PIEZ	1678.00	679.1	
38715	CJ.ECONOMICA	145 PFS DE MEXICO	477.00	534.09	0.00	58.00	58.00 PIEZ	0.19 CAJ	1.00 CAJ	300.00 PIEZ	242.00	189.1	
38723	STICKER DELIVER	145 PFS DE MEXICO	43.00	238.21	0.00	196.00	196.00 PIEZ	0.20 ROLL	1.00 ROLL	1000.00 PIEZ	804.00	145.4	
89583	CUCHARA PLAST.	145 PFS DE MEXICO	0.84	1.20	0.00	1.00	1.00 CAJ	1.00 CAJ	1.00 CAJ	1.00 CAJ	0.00	323.0	
											<b>Sub-Total:</b>	\$ 5,314.5	
											<b>TOTAL:</b>	\$ 5,314.5	

Figura 5.2.11 Pantalla de revisión de pedido.

Si después de revisar el pedido, todo esta correcto se da clic en el botón Aceptar, lo cual mostrará la figura 5.2.12 que es la pantalla donde se muestra la fecha para la cual el proveedor se llevará el pedido para surtirlo posteriormente.

**Revisión de pedido**  
 Fecha: Friday, January 26, 2007. 17:18:26 PM  
 Centro de contribución: 240 - VALLADOLID  
 Número de pedido: 10096  
 Desea confirmar el pedido?

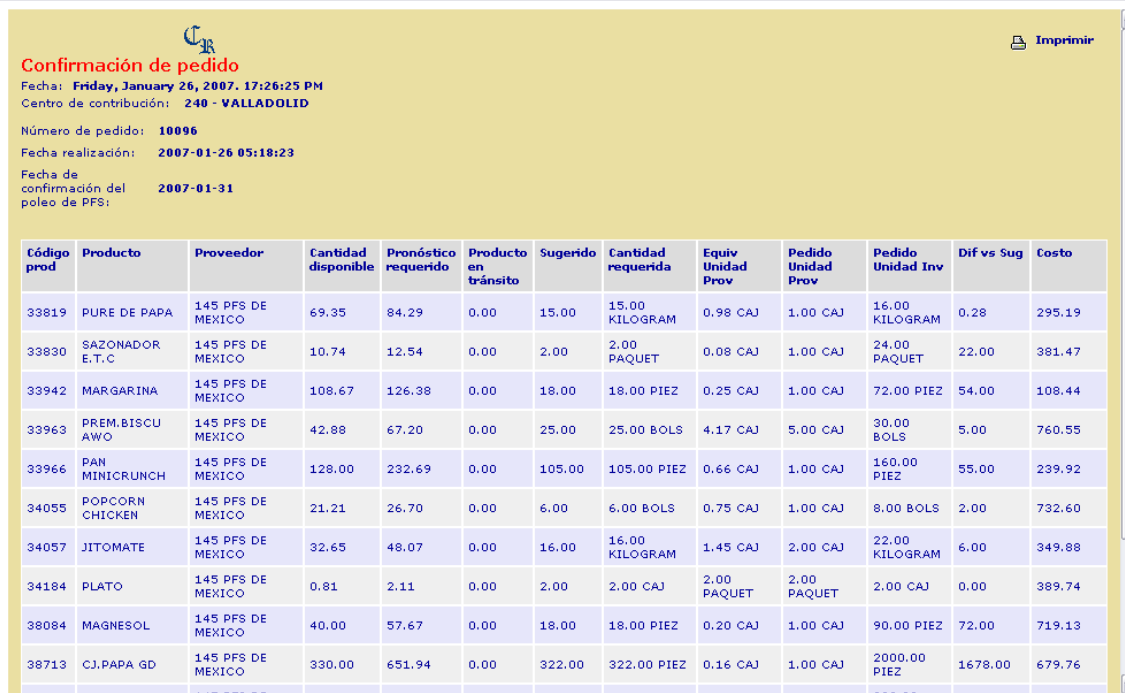
Este pedido es el que Pacific espera polear para:  
 2007-01-31  
   
 Terminado

\* Si deseas adelantar el pedido y la fecha no es la correcta pulsa el botón del "Siguiente pedido" para elegir la siguiente fecha

Codigo Producto	Producto	Proveedor	Datos del Sistema				Esto estas Pidiendo					Dif vs Sug	Costo
			Cantidad Disponible	Pronóstico Requerido	Producto en tránsito	Sugerido	Cantidad Requerida	Equiv Unidad Prov	Pedido Unidad Prov	Pedido Unidad Inv			
33819	PURE DE PAPA	145 PFS DE MEXICO	69.35	84.29	0.00	15.00	15.00 KILOGRAM	0.98 CAJ	1.00 CAJ	16.00 KILOGRAM	0.28	295.1	
33830	SAZONADOR E.T.C	145 PFS DE MEXICO	10.74	12.54	0.00	2.00	2.00 PAQUET	0.08 CAJ	1.00 CAJ	24.00 PAQUET	22.00	381.4	
33942	MARGARINA	145 PFS DE MEXICO	108.67	126.38	0.00	18.00	18.00 PIEZ	0.25 CAJ	1.00 CAJ	72.00 PIEZ	54.00	108.4	
33963	PREM.BISCU AWO	145 PFS DE MEXICO	42.88	67.20	0.00	25.00	25.00 BOLS	4.17 CAJ	5.00 CAJ	30.00 BOLS	5.00	760.5	
33966	PAN MINICRUNCH	145 PFS DE MEXICO	128.00	232.69	0.00	105.00	105.00 PIEZ	0.66 CAJ	1.00 CAJ	160.00 PIEZ	55.00	239.5	
34055	POPCORN CHICKEN	145 PFS DE MEXICO	21.21	26.70	0.00	6.00	6.00 BOLS	0.75 CAJ	1.00 CAJ	8.00 BOLS	2.00	732	
34057	JITOMATE	145 PFS DE MEXICO	32.65	48.07	0.00	16.00	16.00 KILOGRAM	1.45 CAJ	2.00 CAJ	22.00 KILOGRAM	6.00	349.6	
34184	PLATO	145 PFS DE MEXICO	0.81	2.11	0.00	2.00	2.00 CAJ	2.00 PAQUET	2.00 PAQUET	2.00 CAJ	0.00	389.1	
38084	MAGNESOL	145 PFS DE MEXICO	40.00	57.67	0.00	18.00	18.00 PIEZ	0.20 CAJ	1.00 CAJ	90.00 PIEZ	72.00	719.1	
38713	CJ.PAPA GD	145 PFS DE MEXICO	330.00	651.94	0.00	322.00	322.00 PIEZ	0.16 CAJ	1.00 CAJ	2000.00 PIEZ	1678.00	679.1	
38715	CJ.ECONOMICA	145 PFS DE MEXICO	477.00	534.09	0.00	58.00	58.00 PIEZ	0.19 CAJ	1.00 CAJ	300.00 PIEZ	242.00	189.1	
38723	STICKER DELIVER	145 PFS DE MEXICO	43.00	238.21	0.00	196.00	196.00 PIEZ	0.20 ROLL	1.00 ROLL	1000.00 PIEZ	804.00	145.4	
89583	CUCHARA PLAST.	145 PFS DE MEXICO	0.84	1.20	0.00	1.00	1.00 CAJ	1.00 CAJ	1.00 CAJ	1.00 CAJ	0.00	323.0	
											<b>Sub-Total:</b>	\$ 5,314.5	
											<b>TOTAL:</b>	\$ 5,314.5	

Figura 5.2.12 Pantalla de elección de fecha de envío.

Finalmente, al confirmar la fecha de envío al proveedor, se despliega la figura 5.2.13 que muestra la Confirmación del Pedido.



**Confirmación de pedido**  
 Fecha: Friday, January 26, 2007. 17:26:25 PM  
 Centro de contribución: 240 - VALLADOLID  
 Número de pedido: 10096  
 Fecha realización: 2007-01-26 05:18:23  
 Fecha de confirmación del poleo de PFS: 2007-01-31

Código prod	Producto	Proveedor	Cantidad disponible	Pronóstico requerido	Producto en tránsito	Sugerido	Cantidad requerida	Equiv Unidad Prov	Pedido Unidad Prov	Pedido Unidad Inv	Dif vs Sug	Costo
33819	PURE DE PAPA	145 PFS DE MEXICO	69.35	84.29	0.00	15.00	15.00 KILOGRAM	0.98 CAJ	1.00 CAJ	16.00 KILOGRAM	0.28	295.19
33830	SAZONADOR E.T.C	145 PFS DE MEXICO	10.74	12.54	0.00	2.00	2.00 PAQUET	0.08 CAJ	1.00 CAJ	24.00 PAQUET	22.00	381.47
33942	MARGARINA	145 PFS DE MEXICO	108.67	126.38	0.00	18.00	18.00 PIEZ	0.25 CAJ	1.00 CAJ	72.00 PIEZ	54.00	108.44
33963	PREM.BISCU AWO	145 PFS DE MEXICO	42.88	67.20	0.00	25.00	25.00 BOLS	4.17 CAJ	5.00 CAJ	30.00 BOLS	5.00	760.55
33966	PAN MINICRUNCH	145 PFS DE MEXICO	128.00	232.69	0.00	105.00	105.00 PIEZ	0.66 CAJ	1.00 CAJ	160.00 PIEZ	55.00	239.92
34055	POPCORN CHICKEN	145 PFS DE MEXICO	21.21	26.70	0.00	6.00	6.00 BOLS	0.75 CAJ	1.00 CAJ	8.00 BOLS	2.00	732.60
34057	JITOMATE	145 PFS DE MEXICO	32.65	48.07	0.00	16.00	16.00 KILOGRAM	1.45 CAJ	2.00 CAJ	22.00 KILOGRAM	6.00	349.88
34184	PLATO	145 PFS DE MEXICO	0.81	2.11	0.00	2.00	2.00 CAJ	2.00 PAQUET	2.00 PAQUET	2.00 CAJ	0.00	389.74
38084	MAGNESOL	145 PFS DE MEXICO	40.00	57.67	0.00	18.00	18.00 PIEZ	0.20 CAJ	1.00 CAJ	90.00 PIEZ	72.00	719.13
38713	CJ.PAPA GD	145 PFS DE MEXICO	330.00	651.94	0.00	322.00	322.00 PIEZ	0.16 CAJ	1.00 CAJ	2000.00 PIEZ	1678.00	679.76

Figura 5.2.13 Pantalla de confirmación de pedido.

## REMISIÓN

La opción Remisión es para consultar los productos y cantidades que el proveedor teóricamente surtirá. Es la respuesta al pedido levantado.



Figura 5.2.14 Pantalla para elección de la operación de consulta Remisión.

Una vez que se ingresa a esta opción, se deberá elegir el proveedor y posteriormente el número de pedido que tiene asociado el número de remisión asignado por el proveedor. Dar clic en el botón Mostrar, se desplegará el listado de los productos asociados a dicha remisión mostrando las diferencias

entre los productos que se solicitaron y los que el proveedor teóricamente surtirá. Tal y como lo podemos observar en las figura 5.2.15 y 5.2.16.

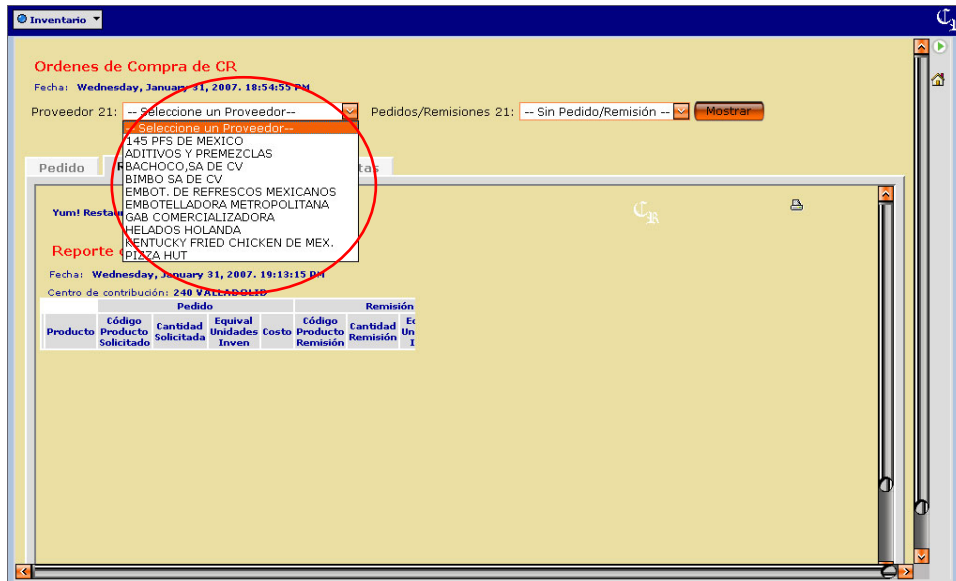


Figura 5.2.15 Proveedores de Remisión

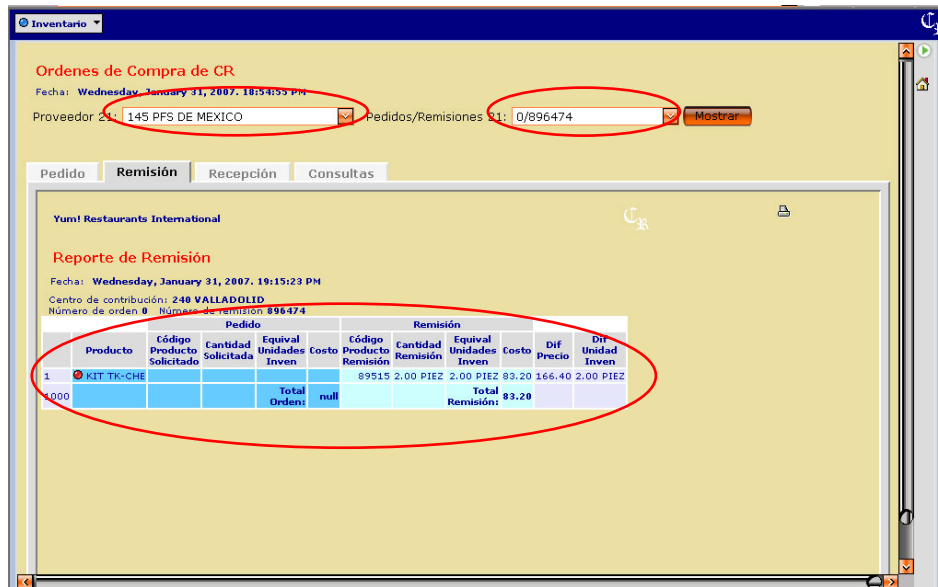


Figura 5.2.16 Reporte de remisión

## RECEPCION

La recepción de los productos es el apartado en el cual se registra cuando el pedido ha sido abastecido.

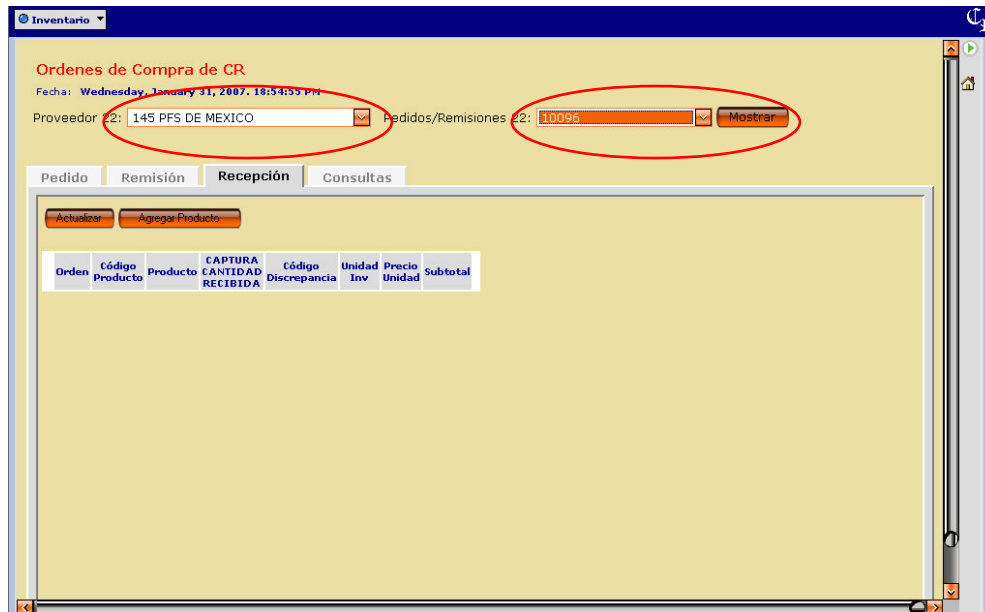


Figura 5.2.17 Recepción

Para poder registrar la recepción se deberá seleccionar el proveedor con el cual se realizó el pedido, como se puede observar en la figura 5.2.17. Al proporcionar estos datos se observa en pantalla la remisión electrónica que fue entregada por el proveedor, ésta debe coincidir con la que se entrega físicamente en el restaurante. En la figura 5.2.18 se ve a detalle lo mencionado anteriormente.

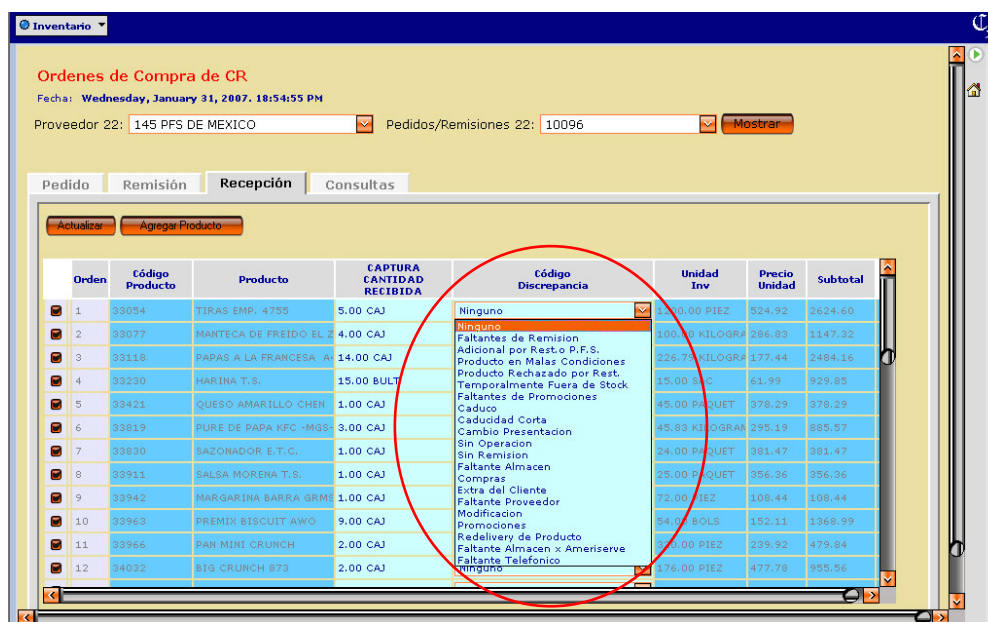


Figura 5.2.18 Productos de Recepción

Si dentro lo que se solicita y lo que se recibe no hay coincidencia, en la columna código de discrepancia se podrán registrar las diferencias.

Cuando ya se realizaron los ajustes necesarios para registrar los productos que realmente se están recibiendo, se da clic en el botón Actualizar para que despliegue la figura 5.2.19 que permite revisar lo que se está registrando en la recepción. Si se detecta algún error ó es necesario realizar ajustes, se da clic en el botón Cancelar.

Orden	Codigo Producto	Producto	Cantidad Recibida	Discrepancia	Unidad Inv	Precio Unit	Subtotal
1	33054	TIRAS EMP. 4755	7.00 CAJ	Ninguno	1680.00 PIEZ	524.92	3,674.44
2	33118	PAPAS A LA FRANCESA A49	16.00 CAJ	Ninguno	259.20 KILOGRAM	177.44	2,839.04
3	34055	POP CORN BACHOCO	3.00 CAJ	Ninguno	24.00 BOLS	732.60	2,197.8
4	34059	MINI CRUNCH BACHOCO	1.00 CAJ	Ninguno	180.00 PIEZ	503.75	503.75
5	34032	BIG CRUNCH 873	4.00 CAJ	Ninguno	352.00 PIEZ	477.78	1,911.12
6	33630	PAY DE LIMON -117803-	1.00 CAJ	Ninguno	48.00 PIEZ	290.24	290.24
7	33966	PAN MINI CRUNCH	1.00 CAJ	Ninguno	160.00 PIEZ	239.92	239.92
8	34057	JITOMATE BIONATUR	4.00 CAJ	Ninguno	44.00 KILOGRAM	174.94	699.76
9	33942	MARGARINA BARRA GRMS 90 GM	2.00 CAJ	Ninguno	144.00 PIEZ	108.44	216.88
10	33421	QUESO AMARILLO CHEN	1.00 CAJ	Ninguno	45.00 PAQUET	378.29	378.29
11	38550	MARCADOR DE CERA BEROL NEGRO	5.00 PIEZ	Ninguno	5.00 PIEZ	5.74	28.7
12	38745	TICKET DE COMPRA KFC DELIVERY	1.00 PAQUET	Ninguno	1.00 PAQUET	98.14	98.14
13	88128	MARCO PARA BISCUITS	1.00 PIEZ	Ninguno	1.00 PIEZ	418.67	418.67
14	89019	DESPACHADOR TOALLA INTERDOB	1.00 PIEZ	Ninguno	1.00 PIEZ	32.00	32
15	89082	SINK SANITIZER 542	40.00 PIEZ	Ninguno	40.00 SOBR	1.07	42.8

Figura 5.2.19 Pantalla de revisión de pedido.

Si al contrario todo es correcto, se captura el nombre de la persona que esta registrando la recepción y se da clic en el botón Aceptar, de manera que se despliegue la figura 5.2.20 que indica que la recepción ha sido confirmada.

**Confirmación de recepción**  
 Fecha: **Friday, January 26, 2007. 17:49:44 PM**  
 Centro de contribución: **240 VALLADOLID**  
 No. de Recepción: **10443**  
 Número de documento: **899293**  
 Capturada por: **NICHEL DEL VALLE**

[Ver reporte de diferencias](#)

Código producto	Producto	Cantidad recibida	Discrepancia	Unidad Inv	Precio unit	Subtotal
1 33054	TIRAS EMP. 4755	7.00 CAJ	Ninguno	1,680.00 PIEZ	524.92	3,674.44
2 33118	PAPAS A LA FRANCESA A49	16.00 CAJ	Ninguno	259.20 KILOGRAM	177.44	2,839.04
3 34055	POP CORN BACHOCO	3.00 CAJ	Ninguno	24.00 BOLS	732.60	2,197.80
4 34059	MINI CRUNCH BACHOCO	1.00 CAJ	Ninguno	180.00 PIEZ	503.75	503.75
5 34032	BIG CRUNCH 873	4.00 CAJ	Ninguno	352.00 PIEZ	477.78	1,911.12
6 33630	PAY DE LIMON -117803-	1.00 CAJ	Ninguno	48.00 PIEZ	290.24	290.24
7 33966	PAN MINI CRUNCH	1.00 CAJ	Ninguno	160.00 PIEZ	239.92	239.92
8 34057	JITOMATE BIONATUR	4.00 CAJ	Ninguno	44.00 KILOGRAM	174.94	699.76
9 33942	MARGARINA BARRA GRMS 90 GM	2.00 CAJ	Ninguno	144.00 PIEZ	108.44	216.88
10 33421	QUESO AMARILLO CHEN	1.00 CAJ	Ninguno	45.00 PAQUET	378.29	378.29
11 38550	MARCADOR DE CERA BEROL NEGRO	5.00 PIEZ	Ninguno	5.00 PIEZ	5.74	28.70
12 38745	TICKET DE COMPRA KFC DELIVERY	1.00 PAQUET	Ninguno	1.00 PAQUET	98.14	98.14
13 88128	MARCO PARA BISCUITS	1.00 PIEZ	Ninguno	1.00 PIEZ	418.67	418.67
14 89019	DESPACHADOR TOALLA INTERDOB	1.00 PIEZ	Ninguno	1.00 PIEZ	32.00	32.00
15 89082	SINK SANITIZER 542	40.00 PIEZ	Ninguno	40.00 SOBR	1.07	42.80
16 39050	PAPEL TOALLA	3.00 ROLL	Ninguno	3.00 ROLL	10.18	30.54

Figura 5.2.20 Pantalla de confirmación de la recepción.

Si existen diferencias entre lo ordenado y lo recibido se generará un reporte de discrepancias y al dar clic en el enlace Ver reporte de diferencias de la figura anterior se mostrará la figura 5.2.21

**Reporte de Discrepancias**  
 Fecha: **Friday, January 26, 2007. 17:55:25 PM**  
 Centro de contribución: **240 VALLADOLID**  
 No. Recepción: **10443**

Desea confirmar el número de Reporte del proveedor?  
 No. Reporte del Proveedor:

Producto	Pedido				Recepción				Codigo Discr.	Dif
	Codigo Producto Solicitado	Cantidad Requerida	Equiv Unidad Inv	Costo	Codigo Producto Recibido	Cantidad Recibida	Equiv Unidad Inv	Costo		
SALSA JALAPE	34157	3.00 CAJ	3.00 CAJ	1015.86				0.00	Faltante Proveedor	-3.00
PAPA FRANCESA	33118	16.00 CAJ	259.00 KILOGRAM	2839.04	33118	15.00 CAJ	243.00 KILOGRAM	2661.60	Faltantes de Remision	-1.00
			Total Orden:	<b>3854.90</b>			Total Recepción:	<b>2661.60</b>		

Figura 5.2.21 Pantalla de reporte de discrepancias.



## CONSULTAS

La parte de Consultas es donde se pueden verificar todos los datos del pedido, la recepción y en caso de existir las discrepancias entre ellas. Como puede observarse en la Figura 5.2.22 este apartado es muy parecido a las pantallas de Órdenes de Compra.

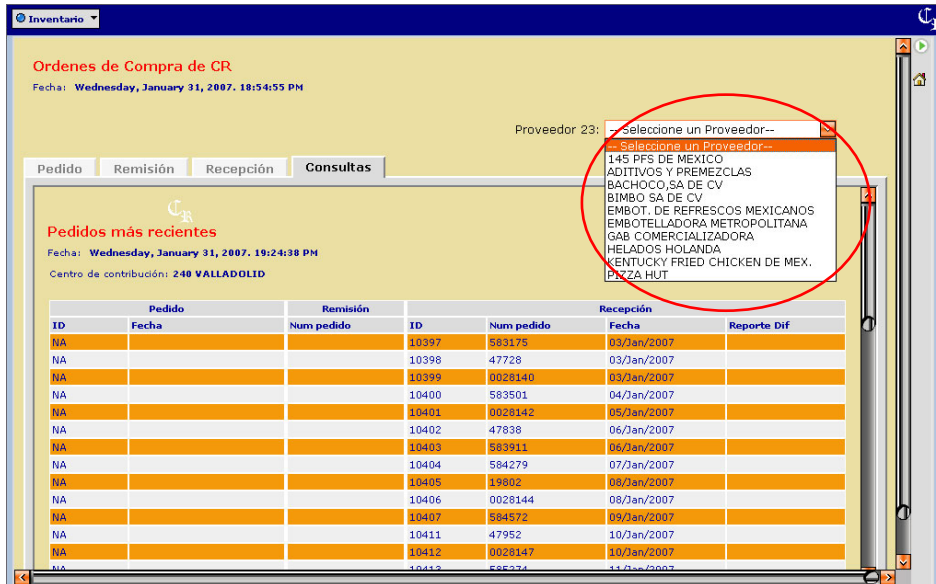


Figura 5.2.22 Proveedores y Productos de Consulta

En la Figura 5.2.23 se observa que al elegir un proveedor y ver sus datos al colocar el ratón sobre cada pedido aparece un mensaje que dice Ver detalle de la recepción. Al darle clic aparece una ventana en la cual se desplegarán los detalles del pedido solicitado figura 5.2.24.

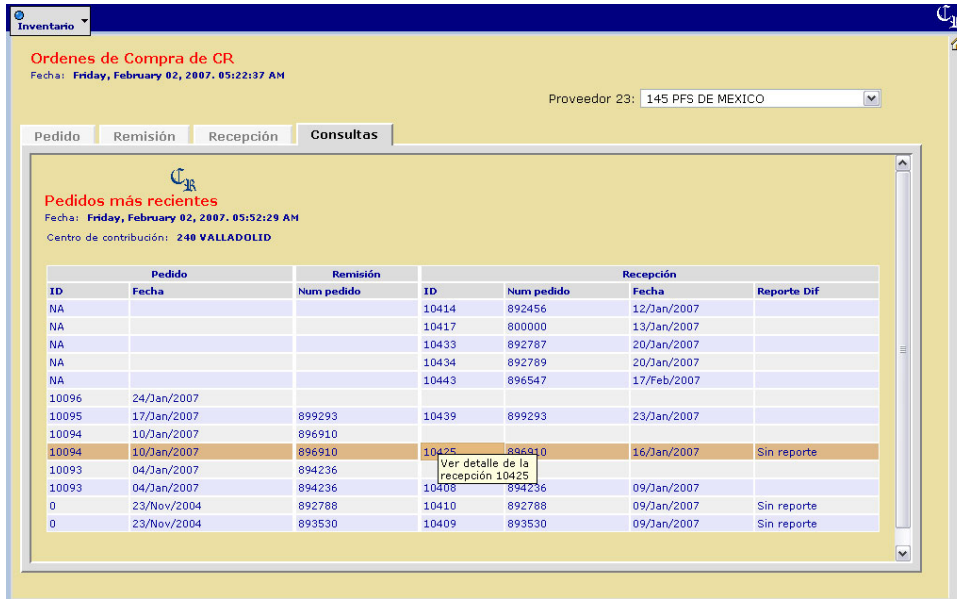


Figura 5.2.23 Consultas de un proveedor



Figura 5.2.24 Detalles de Consulta

## TRANSFERENCIA DE ENTRADA

Como se muestra en la figura 5.2.25 en el menú Control de Inventarios se encuentran las opciones: transferencias de entrada/salida y el inventario semanal.



Figura 5.2.25

Al dar clic en Transferencias de Entrada aparecerá un recuadro que dice Calculando la existencia actual en inventario, espere por favor... Al desaparecer este recuadro se puede iniciar la captura. Capture el número de la Tienda de origen que se solicita.

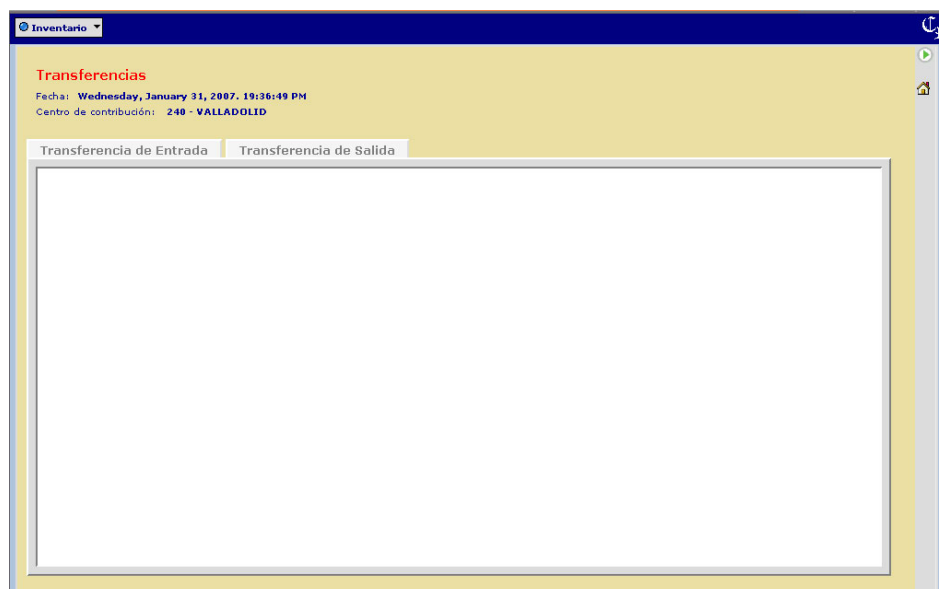


Figura 5.2.26 transferencias pagina principal

A continuación dar clic en el botón Agregar productos para seleccionar los productos que se recibieron de la transferencia. Una vez seleccionados dar

click en Aceptar se mostrarán los productos para poder capturar las unidades recibidas.



Figura 5.2.27 Transferencias.

Dar click en Actualizar para pasar a la revisión de la operación como se muestra en la Figura 5.2.28. Una vez que se encuentre seguro de la transferencia dar click en Aceptar, si hay alguna modificación dar click en Cancelar. Si se acepta la revisión se pasa a la pantalla de confirmación de la transferencia, la cual se puede imprimir si se desea. Ver Figura 5.2.29.



Figura 5.2.28 Revisión Transferencia de entrada.

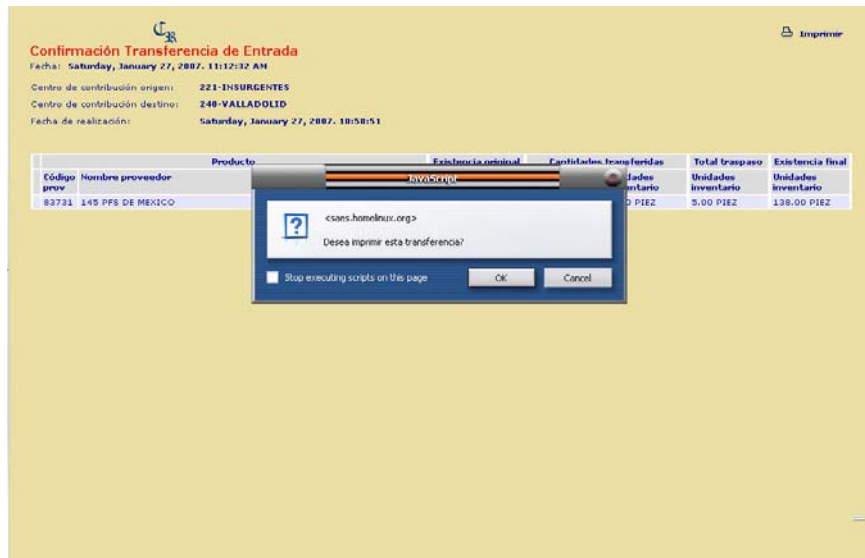


Figura 5.2.29 Confirmación Transferencia Entrada.

## TRANSFERENCIA DE SALIDA

Al dar clic en Transferencias de Salida aparecerá un recuadro que dice Calculando la existencia actual en inventario, espere por favor... Al desaparecer este recuadro se puede iniciar la captura. Capture el número de la Tienda de destino que se solicita, que es el número de tienda que recibirá los productos transferidos.

Dar clic en el botón Agregar Productos, se mostrará la pantalla con los productos que puede transferir, como se observa en la pantalla 5.2.30. Seleccionar los productos deseados dando clic en el checkbox correspondiente.



Figura 5.2.30

Una vez seleccionados todos los productos a transferir dar clic en Aceptar, se mostrarán los productos para poder capturar las cantidades transferidas en la columna Unidades Inventario, tal como se observa en la Figura 5.2.31. Es importante señalar que no podrá transferir una cantidad mayor a la de existencias.

Producto		Existencia original	Cantidades a traspasar		Total traspaso	Existencia final	Pronóstico requerido
Código prov	Nombre proveedor	Unidades inventario	Unidades proveedor	Unidades inventario	Unidades inventario	Unidades inventario	Una semana
<input checked="" type="checkbox"/>	38713 145 PFS DE MEXICO	CAJA GRANDE P/ PAPAS	170.00 PIEZ	0 CAJ	10	170.00	731.28 PIEZ

Disponde de 170.00 PIEZ más que puede transferir

Figura 5.2.31 Transferencias

Dar clic en Actualizar para pasar a la revisión de la operación, como se observa en la Figura 5.2.32. Una vez que se encuentre seguro de la transferencia dar clic en Aceptar, si hay alguna modificación dar clic en Cancelar.

¿ Desea confirmar esta transferencia de salida a la unidad 221-INSURGENTES ?

Producto		Existencia original	Cantidades a traspasar		Total traspaso	Existencia final
Código prov	Nombre proveedor	Unidades inventario	Unidades proveedor	Unidades inventario	Unidades inventario	Unidades inventario
38713	145 PFS DE MEXICO	CAJA GRANDE P/ PAPAS	170.00 PIEZ	0.00 CAJ	2.00 PIEZ	168.00 PIEZ

Figura 5.2.32 Revisión Transferencia de Salida

Si se acepta la revisión se pasa a la pantalla de confirmación de la transferencia, la cual se puede imprimir si se desea.



Figura 5.2.33 Confirmación Transferencia de Salida

### INVENTARIO SEMANAL

En la opción inventario semanal se podrán revisar las existencias de los productos, como se observa en la Figura 5.2.34. En el sistema se muestran los productos clasificados por categoría así como los movimientos que se han hecho del producto a lo largo de la semana, estos son: inventario inicial, recepciones, transferencias de entrada y transferencias de salida, la captura puede realizarse en unidades de proveedor o en unidades de inventario.

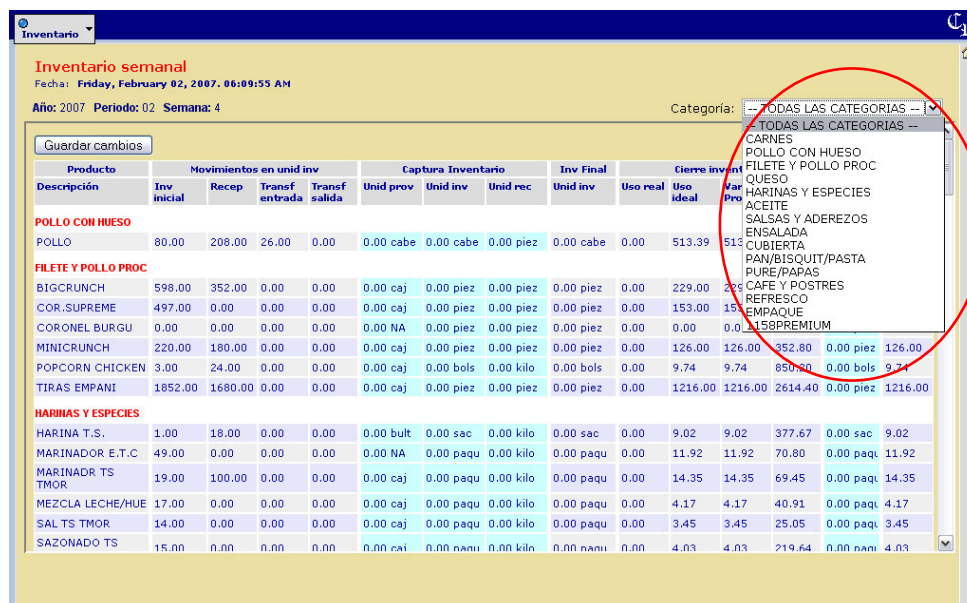
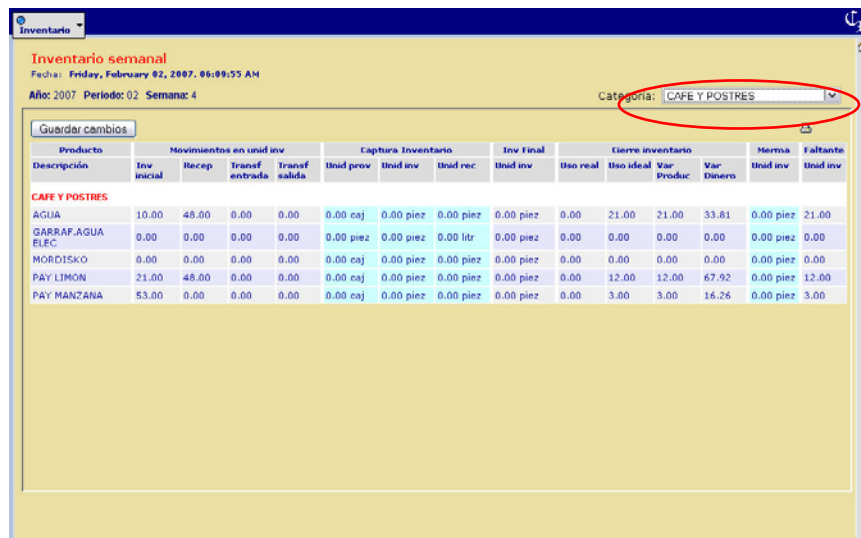


Figura 5.2.34 Inventario Semanal

El campo Categoría es un filtro que permite hacer más sencilla la búsqueda de algún producto, como se muestra en la Figura 5.2.35.



**Inventario semanal**  
Fecha: Friday, February 02, 2007, 06:09:55 AM  
Año: 2007 Período: 02 Semanas: 4

Categoría: CAFE Y POSTRES

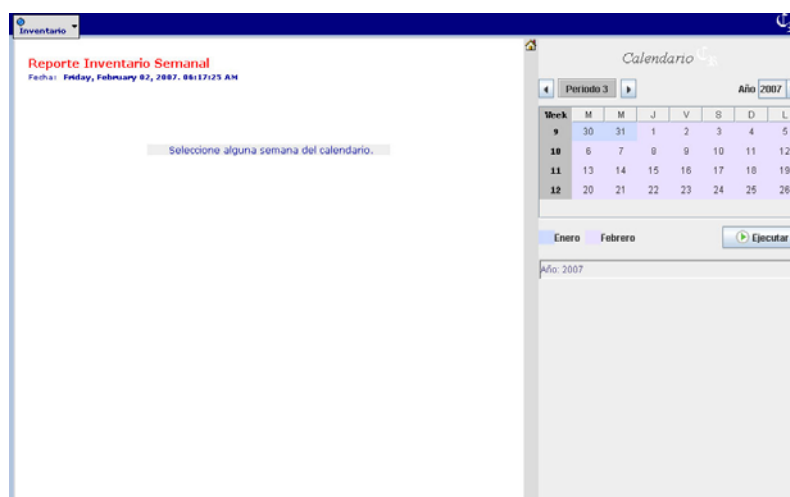
Guardar cambios

Producto	Movimientos en unid inv				Captura Inventario			Inv Final	Cierre inventario				Merma		Faltante
	Inv inicial	Recep	Transf entrada	Transf salida	Unid prov	Unid inv	Unid rec	Unid inv	Bao real	Bao ideal	Var Produc	Var Dinero	Unid inv	Unid inv	
<b>CAFE Y POSTRES</b>															
AGUA	10.00	48.00	0.00	0.00	0.00 caj	0.00 piez	0.00 piez	0.00 piez	0.00	21.00	21.00	33.81	0.00 piez	21.00	
GARRAFAGUA ELEC	0.00	0.00	0.00	0.00	0.00 piez	0.00 piez	0.00 litr	0.00 piez	0.00	0.00	0.00	0.00	0.00 piez	0.00	
MORDISKO	0.00	0.00	0.00	0.00	0.00 caj	0.00 piez	0.00 piez	0.00 piez	0.00	0.00	0.00	0.00	0.00 piez	0.00	
PAY LIMON	21.00	48.00	0.00	0.00	0.00 caj	0.00 piez	0.00 piez	0.00 piez	0.00	12.00	12.00	67.92	0.00 piez	12.00	
PAY MANZANA	53.00	0.00	0.00	0.00	0.00 caj	0.00 piez	0.00 piez	0.00 piez	0.00	3.00	3.00	16.26	0.00 piez	3.00	

Figura 5.2.35 Inventario Semanal

## REPORTES.

Este menú contiene tres opciones de reportes: Inventario Semanal, Transferencia de Entrada y Transferencia de Salida. La opción Inventario semanal nos muestra un calendario para consultar los Inventarios semanales, de acuerdo al periodo, al año y el día. Una vez seleccionados estos parámetros dar click en el botón Ejecutar, como se observa en la Figura 5.2.36.



**Reporte Inventario Semanal**  
Fecha: Friday, February 02, 2007, 06:17:25 AM

Seleccione alguna semana del calendario.

Calendario

Período 3 Año 2007

Week	M	M	J	V	S	D	L
9	30	31	1	2	3	4	5
10	6	7	8	9	10	11	12
11	13	14	15	16	17	18	19
12	20	21	22	23	24	25	26

Enero Febrero Ejecutar

Año: 2007

Figura 5.2.36 Reporte Inventario Semanal



La opción Transferencias de Entrada nos muestra un calendario para consultarlas de acuerdo al periodo, al año y el día. Una vez seleccionados estos parámetros dar click en el botón Ejecutar, como se observa en la Figura 5.2.37.

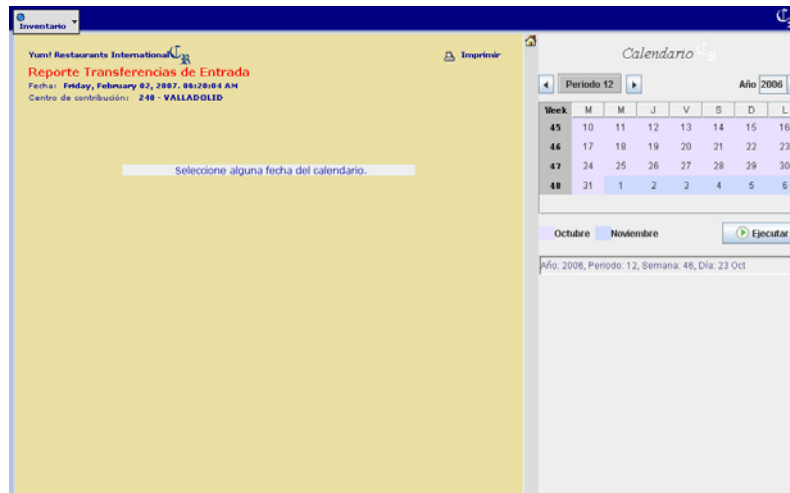


Figura 5.2.37 Reporte Transferencias de Entrada

La opción Transferencias de Salida nos muestra un calendario para consultarlas de acuerdo al periodo, al año y el día. Una vez seleccionados estos parámetros dar click en el botón Ejecutar, como se observa en la Figura 5.2.38.

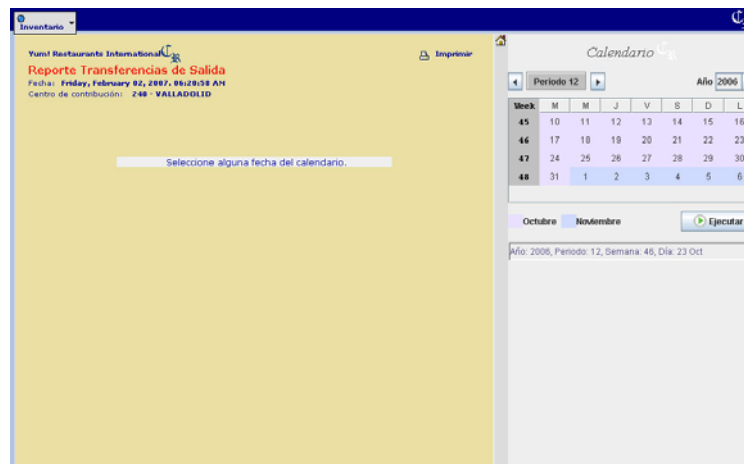


Figura 5.2.38 Reporte Transferencia de Salida

# Capítulo VI. Pruebas.

- 6.1 Ingreso al sistema.
- 6.2 Elaboración de pedido.
- 6.3 Consulta de remisión.
- 6.4 Registro de recepción.
- 6.5 Registro de transferencia de entrada.
- 6.6 Registro de transferencia de salida.
- 6.7 Registro de inventario semanal.

En este capítulo se explica como llevar a cabo el ciclo de petición y recepción de productos, el cual se divide en los siguientes procedimientos: ingresar al sistema, elaboración de pedidos, consulta de remisiones en caso de contar con ellas, registro de la recepción del pedido dentro del sistema, registro de las transferencias de entrada, registro de las transferencias de salida y finalmente el registro del inventario semanal.

### **6.1. Ingresar al sistema**

A continuación se detallan los pasos a seguir para ingresar al sistema, cabe aclarar que siendo un sistema para Web puede ser accedido desde cualquier navegador de Internet, sin embargo la programación de la aplicación contiene características que sólo con Mozilla Firefox funcionará correctamente, debido a esto las pruebas se realizarán en dicho navegador.

1. Se debe abrir un navegador de Internet Mozilla Firefox e ir a la dirección <http://saes.homelinux.org:8080/>, esto desplegará la pantalla principal de la aplicación.
2. La pantalla presenta una pestaña que dice “Acceder”, dar click sobre ella.
3. Al hacer lo anterior se muestra una pequeña ventana que pide *Usuario* y *Contraseña*, los cuales son *admin* y *admin*, respectivamente, a continuación oprimir la tecla *Enter* o dar click el botón de Acceder.

### **6.2. Elaboración de pedido**

Dentro de este proceso existe una fase que se denomina “Pedido Sugerido” que es un listado de los productos que probablemente requiera el restaurante. Este listado lo determina un algoritmo que, como ya se señaló en capítulos anteriores, ya existía en el proceso y fue retomado como tal sin hacerle ninguna modificación, razón por la cuál no se ahondará en esta fase del proceso. A continuación se detallan los pasos para realizar un pedido.

1. Ingresar al sistema con el usuario y contraseña adecuados, como se indica en el punto 6.1.

2. Elegir la siguiente secuencia de menús: Inventario ► Control de Ordenes de Compra ► Orden de Compra.
3. Al dar click sobre la pestaña *Pedido* se despliega un calendario en donde se solicita la fecha del pedido sugerido. Elegir la fecha de la cual se desea generar el pedido sugerido, dando click sobre el día en el calendario.
4. Una vez elegida la fecha para Generar el Pedido Sugerido dar click en el botón “Generar”. Con esto se muestra un mensaje que dice “Generando pedido sugerido. Espere por favor”. Hasta que el mensaje desaparezca de la pantalla se procede al paso siguiente.
5. Dar click en el botón de “Agregar Producto” se mostrará una pantalla con todos los productos disponibles, de todos los proveedores. Aquí tenemos tres alternativas para la elección de productos:
  - a) Se puede seleccionar el pedido sugerido, activando el checkbox “Seleccionar solo productos de Sugerido”. Esto marcará en rosa todos los productos del Pedido Sugerido. Si algún producto no se desea, se puede quitar de la selección dando click sobre el checkbox correspondiente (el que esta a la izquierda del producto).
  - b) Se puede seleccionar todos los productos disponibles activando el checkbox “*Seleccionar Todos*”. Esto marcará con rosa todos los productos. Si algún producto no se desea, se puede quitar de la selección dando click sobre el checkbox correspondiente (el que esta a la izquierda del producto).
  - c) Se pueden seleccionar los productos de manera individual, dando click sobre el checkbox correspondiente al producto deseado (el que esta a la izquierda del producto). Esto marcará en rosa el producto seleccionado.
6. Al quitar la selección de *Todos los productos* cada registro vuelve a su color original y se desactivan los checkbox.
7. Seleccionar solo productos del pedido sugerido. Los registros que en la columna del pedido sugerido tengan un valor diferente de cero se pondrán en color rosa y se activa el checkbox que se encuentra a la izquierda de

- cada registro, se selecciona el pedido sugerido de “Todos” los proveedores.
8. Al quitar la selección de *Solo productos del sugerido* cada registro vuelve a su color original y se desactivan los checkbox.
  9. En la parte superior Elegir al Proveedor *145 PFS* y dar click en el botón “Buscar”. Este filtro mostrará en pantalla únicamente los productos de este proveedor.
  10. Seleccionar los productos del sugerido y dar click en el botón “Aceptar”
  11. Modificar algunas cantidades en la columna “Captura por unidades de inventario”, observar que al salirse del campo se modifiquen correctamente las columnas de “Equiv Unidad Prov”, “Captura por Presentación”, “Pedido Unidad Inv” y “Costo”.
  12. Modificar algunas cantidades en la columna “Captura por Presentación”, observar que al salirse del campo se deben modificar las columnas “Captura Por Unidades De Inventario”, de “Equiv Unidad Prov”, “Pedido Unidad Inv” y “Costo”.
  13. Anotar el código de algún producto que se encuentre en la pantalla.
  14. Dar click en el botón de “Agregar Producto” nuevamente.
  15. El código anotado en el punto número 13, escribirlo en el campo de “Código o Nombre de Producto” y dar click en “Buscar”.
  16. La pantalla posiciona al inicio dicho código, si se escribe parte del nombre del producto, la pantalla posiciona en la parte superior la primera ocurrencia del patrón de búsqueda.
  17. Dar click en “Aceptar”. Observar que el código NO se vuelve a insertar porque ya se había ingresado con anterioridad.
  18. Dar click en el botón de “Agregar Producto” nuevamente.
  19. Añadir un par de productos que no estén en el pedido sugerido (cualquier producto que tenga como valor cero en esa columna)
  20. Dar click en “Aceptar”. Los productos se añadirán en la parte inferior de la pantalla.

21. Modificar la columna “Captura Por Unidades De Inventario” ó “Captura Por Presentación” en uno de los productos.
22. Dejar el otro producto como esta.
23. Dar click en el botón “Actualizar”. Debe mostrar una advertencia de que no se deben capturar cantidades en cero y se enmarca en rojo el campo que es necesario capturar.
24. Capturar una cantidad en cualquiera de los campos: “Captura por unidades de inventario” ó “Captura por presentación”.
25. Desactivar el checkbox (que se encuentra a la izquierda de cada registro) en un par de productos.
26. Dar click en el botón “Actualizar”. (Se despliega la ventana con el título “Revisión de Pedido”).
27. Dar click en el ícono de la impresora que se encuentra en la esquina superior derecha de la pantalla. (Debe mostrarse la misma pantalla, pero reducida, como si estuviera en vista previa).
28. Dar click en el enlace “Continuar con el pedido”.
29. Dar click en el botón de “Cancelar”, con esto todos los productos deben regresar a la pantalla de captura.
30. Dar click en el botón de “Actualizar” nuevamente.
31. Cerrar la ventana (dando click en la cruz de la esquina superior derecha de la ventana).
32. Dar click en el botón de “Actualizar” nuevamente.
33. Dar click en el botón de “Aceptar” de la ventana. Se muestra una vista previa de la confirmación de la orden, lista para imprimirse.
34. Verificar que los datos se hayan insertado correctamente en la base de datos. (tablas op\_grl\_order, op\_grle\_order\_detail, op\_grl\_suggested\_order).
35. Verificar que se genera el archivo ascii relacionado con el proceso en el directorio /usr/bin/ph/3w\_orden.

### 6.3. Consulta de remisión

En este módulo se obtiene la comparación entre la orden del pedido y la remisión que el proveedor teóricamente surtirá, de tal modo que se prevenga la obtención de los insumos críticos requeridos para la venta, mediante transferencias de los restaurantes más cercanos. A continuación se detallan los pasos a seguir para consultar la remisión del pedido. Se elaborará un ejemplo específico para así poder obtener los 4 tipos de diferencias que pueden presentarse en una remisión.

1. Hacer una orden con los siguientes códigos de producto: 34957, 34953, 34158, 34157 y 33962. Todos los productos se ordenan con cantidad de uno, excepto el 34052.
2. Anotar el número de orden que asigna el sistema.
3. Ejecutar el script que genera la remisión de prueba llamado `/usr/local/tomcat/webapps/ROOT/Inventory/PurchaseOrder/Scripts/GenerateRemission.pl` que recibe como parámetro el número de orden anotado en el punto 2.
4. Verificar en la Base de Datos que se haya generado una remisión (en las tablas `op_grl_remission` y `op_grl_remission_detail`).
5. Cerrar todas las ventanas del navegador.
6. Ingresar nuevamente al sistema.
7. Elegir la siguiente secuencia de menús: Inventario ► Control de Ordenes de Compra ► Orden de Compra.
8. Elegir la pestaña “Remisión”.
9. Seleccionar al proveedor “145 PFS”, en el otro combo debe aparecer, si existe, el número de orden y su remisión asociada, seleccionarla y dar click en “Mostrar”.
10. Se muestra una remisión con los 4 tipos de diferencias que tenemos, todas las diferencias se denotan con un círculo rojo al inicio del registro. Las diferencias presentadas deben ser en los siguientes códigos de producto:
  - a. Diferencia en código:

33116, se entrega en la remisión como código 33118.

b. Diferencia en cantidad:

34052, se entrega en la remisión con una unidad menos de la que pedimos.

c. Diferencia por producto extra:

83731, se entrega en la remisión, pero no fue ordenado.

d. Diferencia por producto faltante:

34157, no se entrega en la remisión.

#### **6.4. Registro de recepción**

El registro de recepción permite hacer tres diferentes tipos de recepción. El primero se hace mediante la elección del número de remisión; el segundo se aplica en el caso en que la remisión no se encuentra cargada en el sistema, pero sí se tiene registrada la orden dentro del mismo. Para este caso, se elige el número de orden de la cuál se desea recibir. Finalmente si no es ninguno de los casos anteriores, implica que el pedido fue levantado telefónicamente por estar fuera de tiempo y la recepción debe permitir añadir ó eliminar los productos requeridos. A continuación se detallan los pasos a seguir para realizar este proceso.

1. Ingresar al sistema con el usuario y contraseña adecuados.
2. Elegir la siguiente secuencia de menús: Inventario ► Control de Ordenes de Compra ► Orden de Compra.
3. Seleccionar la pestaña "Recepción".
4. Seleccionar al proveedor "145 PFS".
5. Se hacen los cuatro tipos de recepciones como se describe en los siguientes puntos:
  - a) Con orden y remisión
  - b) Con orden sin remisión
  - c) Con remisión sin orden
  - d) Sin orden ni remisión.



6. Para los tres primeros tipos (a, b, c) se debe seleccionar el número de orden y/o remisión correspondiente y dar click en el botón “Mostrar”. Para el último tipo (d) solo dar click en el botón de “Agregar Productos”.
7. Si elegimos cualquiera de la tres primera opciones de recepción, Se despliega la orden ó remisión de la cual se registra la recepción. Si no, hay que añadir los productos al igual que en la orden por medio del boton “Agregar producto”.
8. Modificar algunas cantidades en el campo de “Captura cantidad recibida”. Observar cómo se actualizan las columnas “Unidad Inv” y “Subtotal”. Modificar la columna de “Código de Discrepancia” asociado al registro.
9. Dar click en “Agregar Producto”, se despliega la pantalla con los productos del proveedor que se eligió para la recepción.
10. Añadir un producto que ya este en la pantalla donde se modifican cantidades y un producto que no este. (También se puede buscar por código o nombre del producto)
11. Dar click en el botón “Aceptar”. El elemento que ya se encontraba en esta pantalla y se intentó agregar nuevamente, no se añade, pero el nuevo sí, hasta el final, con un código de discrepancia elegido por defecto.
12. Dar click en botón “Actualizar”. Sale una advertencia de que no se puede recibir producto en cero y se enmarca en rojo el campo donde la cantidad esta en cero.
13. Modificar la cantidad señalada.
14. Elegir código de discrepancia ninguno en algún producto y modificar cantidad recibida.
15. Dar click en botón “Actualizar”. Sale una advertencia de que se debe capturar código de discrepancia al cambiar la cantidad recibida y se enmarca en rojo donde el código de discrepancia esta seleccionado con “Ninguno”
16. Modificar el código de discrepancia señalado.
17. Quitar la paloma del checkbox a la izquierda de cada registro de un par de productos.

18. Anotar todos los cambios realizados en la pantalla.
19. Dar click en el botón de “Actualizar”. Se muestra la ventana de Revisión de la Recepción, en donde se tienen un par de campos:
  - a) Número de documento, Este no podrá ser editado si se trata de una recepción hecha de una remisión, por default aparecerá ese documento, de cualquier otra forma hay que capturarlo.
  - b) Quien captura. Este campo hay que capturarlo para saber quién registro la recepción dentro del sistema.
20. Dar click en el botón de la impresora que se encuentra en la esquina superior derecha de la ventana. Se muestra una vista previa de la pantalla donde se confirma o cancela la impresión.
21. Dar click en el enlace de “Continuar con la recepción”, y regresamos a la pantalla de revisión.
22. Dar click en el botón de Cancelar ó cerrar la ventana (dando click en la cruz en la esquina superior derecha de la ventana). Los productos recibidos se regresan a la pantalla de captura.
23. Dar click de nuevo en el botón de “Actualizar”.
24. Capturar número de documento (de ser necesario) y nombre de quien captura.
25. Dar click en el botón “Aceptar”.
26. Se muestra la vista previa de la confirmación de la recepción lista para imprimir y un enlace para consultar el “Reporte de Diferencias” dar click en dicha liga.
27. Comparar las diferencias mostradas en el reporte con las anotadas en el punto 18.
28. Cerrar la ventana.
29. Verificar que los datos estén dentro de la base de datos en las tablas `op_grl_reception_detail`, `op_grl_reception` y `op_grl_difference`.
30. Verificar que se hayan generado los archivos asociados con este proceso, los cuales son:
  - a) Archivo en `/usr/bin/ph/3w_recepción` – Es el ascii de la recepción.

- b) Archivo en /usr/bin/ph/3w\_discrem – Ascii de diferencias entre remisión y recepción.
- c) Archivo en /usr/bin/ph/3w\_discord – Ascii de diferencias entre orden y recepción.
- d) Archivo en /usr/bin/ph/rpcost – Ascii para que se altere el inventario en SAI.
- e) Archivo en /usr/fms/op/rpts/sdc\_drt – Archivo para SDC (otro sistema contable en las oficinas).

### **6.5. Registro de transferencia de entrada**

Este módulo realiza una afectación al inventario en el sistema, en el caso en que se ingresen productos a este restaurante transferidos de otro restaurante y no del proveedor. A continuación se detallan los pasos a seguir para realizar dicho proceso.

1. Antes de iniciar esta prueba se debe garantizar la existencia de datos en el directorio /usr/fms/data para poder generar el pedido sugerido de SAI y de allí tomar el pronóstico requerido. También es importante que exista el archivo de inventario (invtran) de SAI con datos reales del año, periodo y semana actual, ya que es afectado directamente al realizar una transferencia de entrada.
2. Ingresar a la aplicación con el usuario y contraseña adecuados.
3. Elegir la secuencia siguiente de menús: Inventario ► Control de inventario ► Transferencias de entrada/salida.
4. Dar click sobre la pestaña “Transferencia de entrada”.
5. Dar click sobre el botón “Agregar productos” (Se abre una nueva ventana con todos los productos del inventario).
6. Seleccionar los productos que se van a recibir (utilizar los checkbox) y presionar el botón “Aceptar”.

7. En la columna titulada *Cantidades a traspasar*, escribir las cantidades de los productos que se van a recibir. Se pueden escribir cantidades en unidades de inventario (ej. piezas).
8. Repetir los pasos 5, 6 y 7 para incorporar más productos a la transferencia de entrada.
9. Al terminar de capturar las cantidades de todos los productos, escribir también el número del centro de contribución del cual se reciben los productos (cuadro de texto con la leyenda “Tienda de origen”), y presionar el botón “Actualizar”.
10. Se muestra una nueva ventana llamada Revisión de transferencia de entrada en la que se tiene la opción de confirmar la transferencia de entrada.
11. Para modificar las cantidades a recibir, eliminar alguno de los productos, o bien agregar un producto a la transferencia, presionar el botón “Cancelar”.
12. Para confirmar la transferencia, presionar el botón “Aceptar”.
13. Al realizar la confirmación, para imprimir un reporte de la transferencia realizada presionar el botón “Imprimir comprobante”.
14. Finalmente, presionar el botón “Cerrar ventana”.
15. Se muestra nuevamente la pantalla que se despliega al elegir la secuencia del punto 3.
16. Verificar que los datos de la transferencia han sido guardados en la base de datos. Las tablas a revisar son: `op_grl_transfer` y `op_grl_transfer_detail`.
17. Verificar que se hayan generado los archivos asociados a este proceso. Dichos archivos se encuentran en:
  - a) `/usr/fms/op/rpts/sdc_dim` (ej. 00021dim.811, 00021dim.821).  
Información de la transferencia de entrada para SDC.
  - b) `/usr/bin/ph/rpcost` (ej. 05-08-11.fms, 05-08-21.fms). Información de la transferencia de entrada para cargarse en el archivo de inventario de SAI.

## 6.6. Registro de transferencia de salida

En el caso en que un restaurante externo solicite apoyo de otro restaurante (local) para surtir productos faltantes, este módulo realiza la afectación al inventario descontando los productos que se transfieren del local al externo. A continuación se detallan los pasos a seguir para realizar dicho proceso.

1. Hay que asegurarse que existen datos en el directorio /usr/fms/data para poder generar el pedido sugerido de SAI y de allí tomar el pronóstico requerido. También es importante que exista el archivo de inventario (invtran) de SAI con datos reales del año, periodo y semana actual, ya que es afectado directamente al realizar una transferencia de salida.
2. Ingresar a la aplicación con el usuario y contraseña adecuados.
3. Elegir la secuencia siguiente de menús: Inventario ► Control de inventario ► Transferencias de entrada/salida.
4. Presionar sobre la pestaña Transferencia de salida.
5. Presionar el botón “Agregar productos” (Se abre una nueva ventana con todos los productos del inventario).
6. Seleccionar los productos que se van a transferir (utilizar los checkbox) y presionar el botón “Aceptar”.
7. En la columna titulada Cantidades a traspasar, escribir las cantidades de los productos que se van a transferir. Se pueden escribir cantidades en unidades de inventario (ej. piezas).
8. Repetir los pasos 5, 6 y 7 para incorporar más productos a la transferencia de salida.
9. Al terminar de capturar las cantidades de todos los productos, escribir también el número del centro de contribución al que se van a enviar los productos (cuadro de texto con la leyenda “Tienda de destino”), y presionar el botón “Actualizar”.
10. Se muestra una nueva ventana llamada Revisión de Transferencia de salida en la que se tiene la opción de confirmar la transferencia de salida.

11. Para modificar las cantidades a transferir, eliminar alguno de los productos, o bien agregar un producto a la transferencia, presionar el botón “Cancelar”.
12. Para confirmar la transferencia, presionar el botón “Aceptar”.
13. Al realizar la confirmación, para imprimir un reporte de la transferencia realizada presionar el botón “Imprimir comprobante”.
14. Finalmente, presionar el botón “Cerrar ventana”.
15. Se muestra nuevamente la pantalla que se despliega al elegir la secuencia del punto 3.
16. Verificar que los datos de la transferencia han sido guardados en la base de datos. Las tablas a revisar son: op\_grl\_transfer y op\_grl\_transfer\_detail.
17. Verificar que se hayan generado los archivos asociados a este proceso. Dichos archivos se encuentran en:
  - a) /usr/fms/op/rpts/sdc\_dex (ej 00021dex.829, 00021dex.830). Información de la transferencia de salida para SDC.

### **6.7. Registro de inventario semanal**

En este módulo se captura la cantidad de inventario final de la semana, a partir de los movimientos realizados dentro del sistema: recepciones, transferencias de entrada y de salida. En el sistema se muestran los productos clasificados por familia así como los movimientos que se han hecho del producto a lo largo de la semana, estos son: inventario inicial, recepciones, transferencias de entrada y transferencias de salida, la captura puede realizarse en unidades de proveedor, unidades de inventario o unidades de receta. A continuación se detallan los pasos a seguir para llevar a cabo dicho proceso.

1. Antes de iniciar esta prueba, debe garantizarse la existencia de datos en el directorio /usr/fms/data, principalmente el archivo de inventario (invtran) de SAI con datos reales del año, periodo y semana actual, ya que dicho archivo es utilizado para obtener información del inventario inicial y uso ideal, además de que es afectado al guardar los valores del inventario final.

2. Es importante también que existan los archivos de reportes de movimientos financieros en `/usr/fms/op/rpts/sdc_dft`, y que el directorio del `/usr/fms/op/print1` (datos de venta) tenga datos reales. Esto es debido a que se tienen que obtener los valores de las ventas netas diarias, pues son utilizadas en el módulo de inventario.
3. Ingresar a la aplicación con el usuario y contraseña adecuados.
4. Elegir la secuencia siguiente de menús: Inventario ► Control de inventario ► Inventario semanal.
5. Se muestra la pantalla de captura de inventario, con los productos del inventario agrupados por familia.
6. En la columna titulada Captura de Inventario, escribir las cantidades para el inventario final. Se pueden escribir cantidades en unidades de proveedor (ej. cajas), en unidades de inventario (ej. paquetes), y en unidades de receta (ej. piezas).
7. En la columna titulada Merma, escribir las cantidades de los productos mermados.
8. Al terminar de escribir las cantidades de los productos a inventariar, presionar el botón “Guardar cambios”.
9. Para hacer algún cambio en las cantidades de los productos que se están inventariando, presionar el botón “Cancelar”.
10. Para confirmar los valores capturados del inventario final, presionar el botón “Aceptar”.
11. Al realizar la confirmación, para imprimir un reporte del inventario final capturado, presionar el botón “Imprimir comprobante”.
12. Finalmente, presionar el botón “Cerrar ventana”.
13. Se muestra nuevamente la pantalla inicial de captura de inventario.
14. Verificar que los datos del inventario final han sido guardados en la base de datos. Las tablas a revisar son: `op_inv_inventory_detail`.
15. Revisar que en el archivo de inventario de SAI (`invtran`) se han escrito los valores del inventario final.



# Capítulo VII Conclusiones.



Los sistemas de cómputo, buscan de forma general, aportar una mejora notoria en los procesos de negocio llevados a cabo por cualquier organización, esto incluye la automatización y control de los mismos. Desarrollando este proyecto de Tesis se ha intentado cumplir con este punto primordialmente y además se han obtenido una serie de conclusiones relevantes acerca de este mejoramiento, debido a que los objetivos planteados al inicio han sido cumplidos de manera satisfactoria denotando una amplia mejoría en los siguientes aspectos:

- El sistema permite realizar todas las operaciones con una misma interfaz para el usuario y con un solo sistema, lo que facilita la forma de operación ya que no tiene que consultar información de distintas fuentes.
- Se logra una mejora considerable en el tiempo de elaboración de un pedido para los proveedores debido a que una tarea realizada manualmente tal como lo es el cálculo de pedido sugerido y la conversión de unidades, ahora es realizada a través del sistema.
- Se tiene un control de los pedidos debido al proceso que permite adelantarlos hasta siete días de manera que no se recurra a confusiones por parte del proveedor a la hora de surtirlo.
- Los restaurantes pueden conocer con anticipación los productos que probablemente no sean surtidos por el proveedor de manera que pueda prever esta situación y que no se convierta en una situación crítica por falta de insumos.
- Se logra una mejora en tiempo en el registro de productos recibidos, debido a que ya se tiene esta información almacenada en el sistema, de manera que sólo se hagan los ajustes necesarios para ingresar esta información correctamente dentro del inventario.
- Se le presenta al usuario un reporte de diferencias entre lo que ordena y recibe, de manera que pueda solicitar fácilmente un reenvío de producto al proveedor por los productos faltantes.

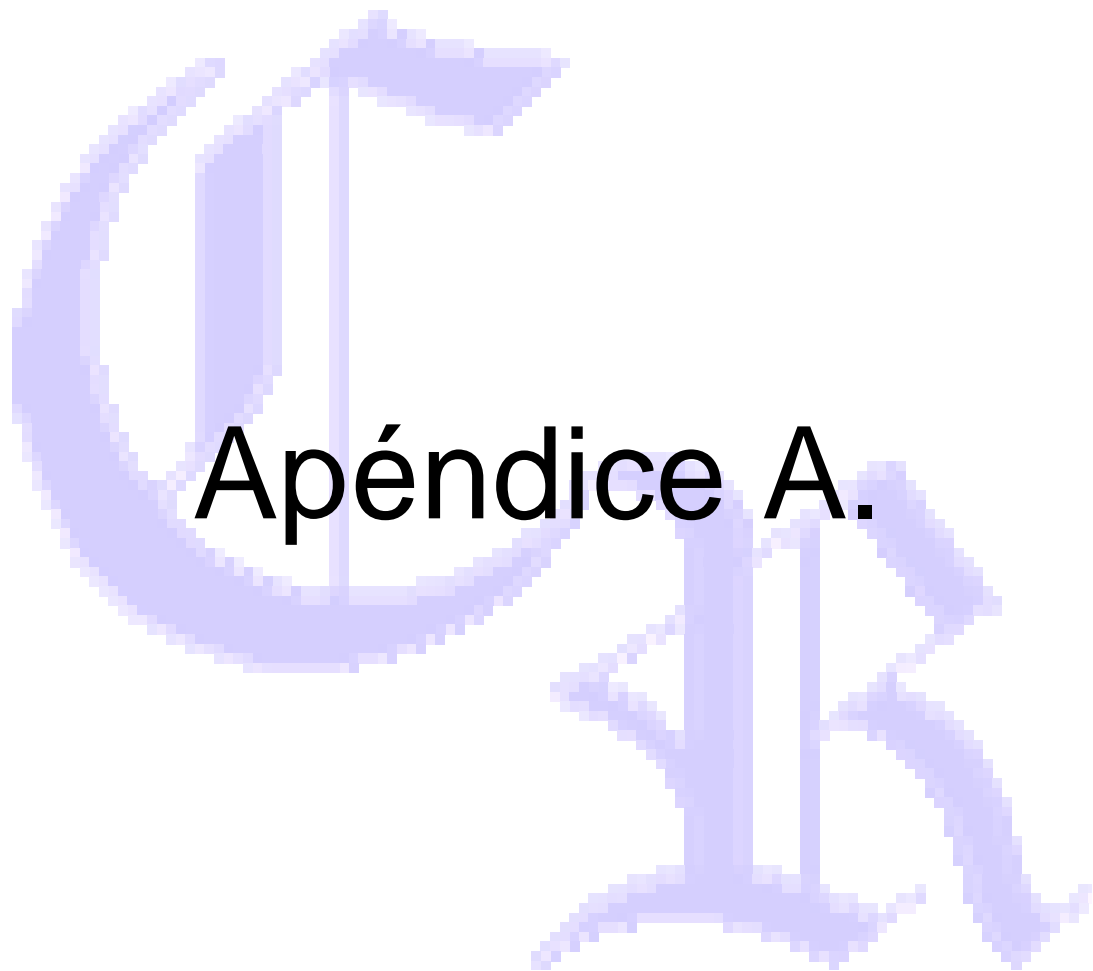
- En oficinas ahora se tiene el control de lo que cada restaurante esta recibiendo por cada proveedor, de manera que sólo se pague por lo que se recibe y no por lo que se factura, ya que ahí se generaban grandes diferencias en dinero que ahora representa un ahorro para la compañía.
- Se tiene una diferenciación entre las entradas de producto ingresadas por recepción de proveedor y las transferencias de entrada de otro restaurante, lo que permite un mayor control en el inventario.
- La información de inventario ingresada a través de este nuevo sistema, se incluye también en los archivos de control del SAI para el cálculo de índices control y pronósticos de venta de los restaurantes.

Debido a todos los puntos mencionados anteriormente, el proyecto de Tesis ha sido una experiencia enriquecedora y de gran aprendizaje, ya que abarco todo lo que es el ciclo de desarrollo de un sistema y su puesta en marcha, lo cual implica una gran satisfacción por haber conseguido el objetivo principal que dio origen al presente trabajo.

### **7.1 Trabajo a futuro.**

Durante la realización de este proyecto de Tesis se detectaron nuevas necesidades para el mejoramiento y ampliación de este sistema, de manera que se sigan logrando beneficios con el control de inventario tales como ahorros significativos tanto en tiempo como en dinero, disminución de mermas, obtención de los insumos en tiempo y forma adecuadas, mejora en la calidad de servicio tanto de los proveedores como de los restaurantes, etc. Estas necesidades dan pie a ampliar la investigación y desarrollo del presente sistema y se listan a continuación:

- Cierre de inventario semanal a través del sistema desarrollado, ya que se hace a través del SAI, lo que implica que se tenga que acceder a otro sistema que busca suplirse en un futuro.
- Diseño de algoritmos para la generación de pedido sugerido de manera que se logre una independencia entre el SAI y esta nueva plataforma.
- Desarrollo de un módulo para la captura diaria de inventario de productos críticos.
- Desarrollo de reportes que permitan medir la calidad de servicio de los proveedores, consultar información histórica de los conceptos de inventario y pedidos realizados.
- Optimización de la plantilla de desarrollo utilizada para la programación de este sistema.





## Diccionario de Datos

## Table details

## Tabla: op\_grl\_cat\_config\_difference

## Detalles de la tabla:

Descripción Tabla que almacena el orden en que deben desplegarse en la lista de código de discrepancia dependiendo el caso de la recepción.

Nombre de la restricción de llave primaria PK\_op\_grl\_cat\_config\_difference

## Columnas:

Llave	Nombre del campo	Tipo de dato	No nulo	Default
PK,FK	case_id	SMALLINT	Yes	
PK,FK	difference_id	CHARACTER(2)	Yes	
	sort_seq	SMALLINT	No	

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_difference_op_grl_cat_config_difference	op_grl_cat_difference	difference_id	op_grl_cat_config_difference	difference_id	Cero o Más
op_grl_cat_recep_cases_op_grl_cat_config_difference	op_grl_cat_recep_cases	case_id	op_grl_cat_config_difference	case_id	Cero o Más

## Tabla: op\_grl\_cat\_confirmation

## Detalles de la tabla:

Descripción Catálogo de los tipos de confirmación. Solo puede almacenar los siguientes valores: 1. Electrónica 2. Telefónica 3. Fax 4. E - mail

Nombre de la restricción de llave primaria confirmation\_id

## Columnas:

Llave	Nombre del campo	Tipo de dato	No nulo	Default
PK	confirmation_id	SMALLINT	Yes	
	Descripción	CHARACTER VARYING(20)	No	

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_confirmation_op_grl_cat_provider	op_grl_cat_confirmation	confirmation_id	op_grl_cat_provider	confirmation_id	Cero o Más

## Tabla: op\_grl\_cat\_difference

## Detalles de la tabla:

Descripción Catálogo con los códigos de discrepancia disponibles al hacer una recepción.

Nombre de la restricción de llave primaria PK\_op\_grl\_cat\_difference

## Columnas:

Llave	Nombre del campo	Tipo de dato	No nulo	Default
PK	difference_id	CHARACTER(2)	Yes	
	dif_desc	CHARACTER VARYING(30)	No	

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_difference_op_grl_cat_config_difference	op_grl_cat_difference	difference_id	op_grl_cat_config_difference	difference_id	Cero o Más
op_grl_cat_difference_op_grl_reception_detail	op_grl_cat_difference	difference_id	op_grl_reception_detail	difference_id	Cero o Más
op_grl_cat_difference_op_grl_step_reception_detail	op_grl_cat_difference	difference_id	op_grl_step_reception_detail	difference_id	Cero o Más

## Tabla: op\_grl\_cat\_family

## Detalles de la tabla:

Descripción Catálogo de las familias en las que se clasifican los productos de inventario.

Nombre de la restricción de llave primaria PK\_op\_grl\_cat\_family

## Columnas:

Llave	Nombre del campo	Tipo de dato	No nulo	Default
PK	family_id	CHARACTER(10)	Yes	
	family_desc	CHARACTER VARYING(20)	No	
	family_order	SMALLINT	No	

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_family_op_grl_cat_inventory	op_grl_cat_family	family_id	op_grl_cat_inventory	family_id	Cero o Más

**Tabla: op\_grl\_cat\_frequency**

## Detalles de la tabla:

Descripción: Catálogo de las frecuencias en las que se debe inventariar un producto de inventario. Los valores que puede tener: 1 - Diario 2 - Semanal 3 - Periódico 4 - No se inventaría

Nombre de la restricción de llave primaria: PK\_op\_grl\_cat\_frequency

## Columnas:

Llave	Nombre del campo	Tipo de dato	No nulo	Default
PK	frequency_id	SMALLINT	Yes	
	frec_desc	CHARACTER VARYING(20)	No	

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_frequency_op_grl_cat_inventory	op_grl_cat_frequency	frequency_id	op_grl_cat_inventory	frequency_id	Cero o Más

**Tabla: op\_grl\_cat\_inventory**

## Detalles de la tabla:

Descripción: Catálogo de los productos de inventario.

Nombre de la restricción de llave primaria: PK\_op\_grl\_cat\_inventory

## Columnas:

Llave	Nombre del campo	Tipo de dato	No nulo	Default
PK	inv_id	CHARACTER(6)	Yes	
	inv_desc	CHARACTER VARYING(30)	No	
FK	family_id	CHARACTER(10)	Yes	
FK	inv_unit_measure	CHARACTER(4)	Yes	
FK	frequency_id	SMALLINT	No	
FK	recipe_unit_measure	CHARACTER(4)	Yes	
	rcp_conversion_factor	NUMERIC(12,2)	No	
FK	company_id	CHARACTER(3)	Yes	
	active_flag	BOOLEAN	No	

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_family_op_grl_cat_inventory	op_grl_cat_family	family_id	op_grl_cat_inventory	family_id	Cero o Más
op_grl_cat_frequency_op_grl_cat_inventory	op_grl_cat_frequency	frequency_id	op_grl_cat_inventory	frequency_id	Cero o Más
op_grl_cat_inventory_op_grl_cat_providers_product	op_grl_cat_inventory	inv_id	op_grl_cat_providers_product	inv_id	Cero o Más
op_grl_cat_inventory_op_grl_suggested_order	op_grl_cat_inventory	inv_id	op_grl_suggested_order	inv_id	Cero o Más
op_grl_cat_inventory_op_grl_suggested_transfer	op_grl_cat_inventory	inv_id	op_grl_suggested_transfer	inv_id	Cero o Más
op_grl_cat_inventory_op_inv_inventory_detail	op_grl_cat_inventory	inv_id	op_inv_inventory_detail	inv_id	Cero o Más
op_grl_cat_inventory_op_inv_step_inventory_detail	op_grl_cat_inventory	inv_id	op_inv_step_inventory_detail	inv_id	Cero o Más
op_grl_cat_product_company_op_grl_cat_inventory	op_grl_cat_product_company	company_id	op_grl_cat_inventory	company_id	Cero o Más
op_grl_cat_unit_measure_op_grl_cat_inventory	op_grl_cat_unit_measure	unit_id	op_grl_cat_inventory	inv_unit_measure	Cero o Más
op_grl_cat_unit_measure_op_grl_cat_inventory	op_grl_cat_unit_measure	unit_id	op_grl_cat_inventory	recipe_unit_measure	Cero o Más

**Tabla: op\_grl\_cat\_product\_company**

## Detalles de la tabla:

Descripción: Catálogo de qué producto se aplica a cada marca, sus valores únicamente pueden ser: KFC - KFC PH - PH COM - COMBO AMB - AMBOS IND - INDEFINIDO OBS - OBSOLETO

Nombre de la restricción de llave primaria: PK\_op\_grl\_cat\_product\_company

## Columnas:

Llave	Nombre del campo	Tipo de dato	No nulo	Default
PK	company_id	CHARACTER(3)	Yes	
	company_desc	CHARACTER	No	

VARYING(20)

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_product_company_op_grl_cat_inventory	op_grl_cat_product_company	company_id	op_grl_cat_inventory	company_id	Cero o Más

**Tabla: op\_grl\_cat\_provider**

## Detalles de la tabla:

Descripción	Catálogo de proveedores disponibles.
Nombre de la restricción de llave primaria	PK_op_grl_cat_provider

## Columnas:

Llave	Nombre del campo	Tipo de dato	No nulo	Default
PK	provider_id	CHARACTER(10)	Yes	
	provider_code_before	CHARACTER(10)	No	
	name	CHARACTER VARYING(30)	No	
	street	CHARACTER VARYING(30)	No	
	neighbourhood	CHARACTER VARYING(30)	No	
	delegation	CHARACTER VARYING(30)	No	
	document_type_id	SMALLINT	No	
FK	confirmation_id	SMALLINT	Yes	
	contact	CHARACTER VARYING(25)	No	
	phone	CHARACTER VARYING(20)	No	
	email	CHARACTER VARYING(40)	No	
	ip_phone	CHARACTER VARYING(6)	No	
	bank_account	CHARACTER VARYING(20)	No	
	active_flag	BOOLEAN	No	

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_confirmation_op_grl_cat_provider	op_grl_cat_confirmation	confirmation_id	op_grl_cat_provider	confirmation_id	Cero o Más
op_grl_cat_provider_op_grl_cat_providers_product	op_grl_cat_provider	provider_id	op_grl_cat_providers_product	provider_id	Cero o Más
op_grl_cat_provider_op_grl_reception	op_grl_cat_provider	provider_id	op_grl_reception	provider_id	Cero o Más
op_grl_cat_provider_op_grl_remission	op_grl_cat_provider	provider_id	op_grl_remission	provider_id	Cero o Más
op_grl_cat_provider_op_grl_step_reception	op_grl_cat_provider	provider_id	op_grl_step_reception	provider_id	Cero o Más
op_grl_document_type_op_grl_cat_provider	op_grl_document_type	document_type_id	op_grl_cat_provider	document_type_id	Cero o Más

**Tabla: op\_grl\_cat\_providers\_product**

## Detalles de la tabla:

Descripción	Catálogo de los productos asociados a cada proveedor.
Nombre de la restricción de llave primaria	PK_op_grl_cat_providers_product

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	provider_id	CHARACTER(10)		Yes		
	provider_product_code	CHARACTER(10)		Yes		
FK	inv_id	CHARACTER(6)		Yes		
	stock_code_id	CHARACTER(6)		No		
FK	provider_unit_measure	CHARACTER(4)		No		
	conversion_factor	NUMERIC(12,2)		No		
	provider_price	NUMERIC(12,2)		No		
	provider_product_desc	CHARACTER VARYING(40)		No		
	active_flag	BOOLEAN		No		
	reception_limit	INTEGER		No		



## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_inventory_op_grl_cat_providers_product	op_grl_cat_inventory	inv_id	op_grl_cat_providers_product	inv_id	Cero o Más
op_grl_cat_providers_product_op_grl_order_detail	op_grl_cat_providers_product	provider_id,provider_product_code	op_grl_order_detail	provider_id,provider_product_code	Cero o Más
op_grl_cat_providers_product_op_grl_remission_detail	op_grl_cat_providers_product	provider_id	op_grl_remission_detail	provider_id	Cero o Más
op_grl_cat_providers_product_op_grl_remission_detail2	op_grl_cat_providers_product	provider_id,provider_product_code	op_grl_reception_detail	provider_id,provider_product_code	Cero o Más
op_grl_cat_providers_product_op_grl_step_order_detail	op_grl_cat_providers_product	provider_id,provider_product_code	op_grl_step_order_detail	provider_id,provider_product_code	Cero o Más
op_grl_cat_providers_product_op_grl_step_reception_detail	op_grl_cat_providers_product	provider_id,provider_product_code	op_grl_step_reception_detail	provider_id,provider_product_code	Cero o Más
op_grl_cat_providers_product_op_grl_way_order	op_grl_cat_providers_product	provider_id,provider_product_code	op_grl_way_order	provider_id,provider_product_code	Cero o Más
op_grl_cat_provider_op_grl_cat_providers_product	op_grl_cat_provider	provider_id	op_grl_cat_providers_product	provider_id	Cero o Más
op_grl_cat_unit_measure_op_grl_cat_providers_product	op_grl_cat_unit_measure	unit_id	op_grl_cat_providers_product	provider_unit_measure	Exactly One

**Tabla: op\_grl\_cat\_recep\_cases**

## Detalles de la tabla:

Descripción	Catálogo de los casos de recepción que se pueden llevar a cabo. Sus valores son: 1 - Sin orden y sin remisión 2 - Sin orden y con remisión 3 - Con orden y sin remisión 4 - Con orden y con remisión 5 - Nuevos elementos 6 - Diff ord/recep no rem 7 - Dif ord/recep inex
Nombre de la restricción de llave primaria	PK_op_grl_cat_recep_cases

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	case_id	SMALLINT		Yes		
	case_desc	CHARACTER VARYING(25)		No		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_recep_cases_op_grl_cat_config_difference	op_grl_cat_recep_cases	case_id	op_grl_cat_config_difference	case_id	Cero o Más

**Tabla: op\_grl\_cat\_unit\_measure**

## Detalles de la tabla:

Descripción	Catálogos de unidades de medida, tanto de proveedor como de inventario.
Nombre de la restricción de llave primaria	PK_op_grl_cat_unit_measure

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	unit_id	CHARACTER(4)		Yes		
	unit_name	CHARACTER VARYING(50)		No		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_unit_measure_op_grl_cat_inventory	op_grl_cat_unit_measure	unit_id	op_grl_cat_inventory	recipe_unit_measure	Cero o Más
op_grl_cat_unit_measure_op_grl_cat_inventory	op_grl_cat_unit_measure	unit_id	op_grl_cat_inventory	inv_unit_measure	Cero o Más
op_grl_cat_unit_measure_op_grl_cat_providers_product	op_grl_cat_unit_measure	unit_id	op_grl_cat_providers_product	provider_unit_measure	Exactly One
op_grl_cat_unit_measure_op_grl_order_detail	op_grl_cat_unit_measure	unit_id	op_grl_order_detail	provider_unit	Cero o Más
op_grl_cat_unit_measure_op_grl_remission_detail	op_grl_cat_unit_measure	unit_id	op_grl_remission_detail	unit_measure_remissions	Cero o Más
op_grl_cat_unit_measure_op_grl_step_order_detail	op_grl_cat_unit_measure	unit_id	op_grl_step_order_detail	provider_unit	Cero o Más

**Tabla: op\_grl\_days\_advance\_order**

## Detalles de la tabla:

Descripción Tabla de configuración para conocer el número de días que es permitido adelantar un pedido dentro de un restaurante.

Nombre de la restricción de llave primaria

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
	num_days_advance	CHARACTER VARYING(40)		No		

**Tabla: op\_grl\_difference**

## Detalles de la tabla:

Descripción Tabla para almacenar las diferencias entre orden y recepción encontradas. Los registros se obtienen al confirmar la recepción.

Nombre de la restricción de llave primaria

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
	reception_id	INTEGER		No		
	product_name	CHARACTER VARYING(30)		No		
	order_product	CHARACTER(10)		No		
	qty_required	TEXT		No		
	order_equivalent	TEXT		No		
	ord_cost	NUMERIC		No		
	recep_product	CHARACTER(10)		No		
	qty_received	TEXT		No		
	recep_equivalent	TEXT		No		
	recep_cost	NUMERIC		No		
	dif_desc	CHARACTER VARYING(40)		No		
	diff_prv	TEXT		No		
	dif_inv	TEXT		No		

**Tabla: op\_grl\_document\_type**

## Detalles de la tabla:

Descripción Catálogo con los tipos de documento a recibir. Tiene los valores de : 1 - Remisión 2 - Factura

Nombre de la restricción de llave primaria PK\_op\_grl\_document\_type

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	document_type_id	SMALLINT		Yes		
	Descripción	CHARACTER VARYING(20)		No		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_document_type_op_grl_cat_provider	op_grl_document_type	document_type_id	op_grl_cat_provider	document_type_id	Cero o Más
op_grl_document_type_op_grl_reception	op_grl_document_type	document_type_id	op_grl_reception	document_type_id	One Or More
op_grl_document_type_op_grl_step_reception	op_grl_document_type	document_type_id	op_grl_step_reception	document_type_id	Cero o Más

**Tabla: op\_grl\_existence**

## Detalles de la tabla:

Descripción Tabla que almacena la existencia actual de un producto en el inventario.

Nombre de la restricción de llave primaria PK\_op\_grl\_existence

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	inv_id	CHARACTER(6)		Yes		
PK	family_id	CHARACTER(10)		Yes		

**Tabla: op\_grl\_financial\_mov**

## Detalles de la tabla:

Descripción Tabla que almacena información financiera de ventas, transacciones, gastos, depósitos, movimientos de caja, etc.

Nombre de la restricción de llave primaria PK\_op\_grl\_financial\_mov

## llave primaria

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	finantial_code	CHARACTER(40)		Yes		
	quantity	NUMERIC(12,2)		No		
	store_id	TIME WITH TIME ZONE		No		
PK	date_id	TIMESTAMP		Yes		

**Tabla: op\_grl\_order**

## Detalles de la tabla:

Descripción	Tabla que almacena los pedidos realizados, ya confirmados.
Nombre de la restricción de llave primaria	PK_op_grl_order

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	order_id	INTEGER		Yes		
PK	store_id	SMALLINT		Yes		
	date_id	TIMESTAMP		No		
	date_limit	TIMESTAMP		No		

## Relaciones:

Nombre de la restricción de llave foránea	Nombre de la restricción de llave Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_order_op_grl_order_detail	op_grl_order	order_id,store_id	op_grl_order_detail	order_id,store_id	One Or More
op_grl_order_op_grl_remission	op_grl_order	order_id,store_id	op_grl_remission	order_id,store_id	One Or More
op_grl_order_op_grl_suggested_order	op_grl_order	order_id,store_id	op_grl_suggested_order	order_id,store_id	Cero o Más
op_grl_order_op_grl_way_order	op_grl_order	order_id,store_id	op_grl_way_order	order_id,store_id	Cero o Más

**Tabla: op\_grl\_order\_detail**

## Detalles de la tabla:

Descripción	Tabla que almacena los productos que se requieren en un pedido al confirmarlo. Parte de la llave primaria de esta tabla.
Nombre de la restricción de llave primaria	PK_op_grl_order_detail

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	order_id	INTEGER		Yes		
	store_id	SMALLINT		Yes		
PK,FK	provider_product_code	CHARACTER(10)		Yes		
PK,FK	provider_id	CHARACTER(10)		Yes		
FK	provider_unit	CHARACTER(4)		No		
	unit_cost	NUMERIC(12,2)		No		
	inv_required_quantity	NUMERIC(12,2)		No		
	prv_required_quantity	NUMERIC(12,2)		No		

## Relaciones:

Nombre de la restricción de llave foránea	Nombre de la restricción de llave Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_providers_product_op_grl_order_detail	op_grl_cat_providers	provider_id,provider_product_code	op_grl_order_detail	provider_id,provider_product_code	Cero o Más
op_grl_cat_unit_measure_op_grl_order_detail	op_grl_cat_unit_measure	unit_id	op_grl_order_detail	provider_unit	Cero o Más
op_grl_order_op_grl_order_detail	op_grl_order	order_id,store_id	op_grl_order_detail	order_id,store_id	One Or More

**Tabla: op\_grl\_order\_limit**

## Detalles de la tabla:

Descripción	Tabla que almacena los días límite para que PFS reciba el pedido del restaurante.
Nombre de la restricción de llave primaria	PK_op_grl_order_limit

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	store_id	CHARACTER VARYING(40)		Yes		
PK,FK	weekday_id	SMALLINT		Yes		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
ss_cat_yum_day_op_grl_order_limit	ss_cat_yum_day	weekday_id	op_grl_order_limit	weekday_id	Cero o Más

**Tabla: op\_grl\_reception**

## Detalles de la tabla:

Descripción	Tabla que registra una recepción de producto cuando ésta es confirmada.
Nombre de la restricción de llave primaria	PK_op_grl_reception

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	reception_id	INTEGER		Yes		
PK	store_id	SMALLINT		Yes		
FK	document_num	CHARACTER(10)		No		
	document_type_id	SMALLINT		No		
	remission_id	CHARACTER(10)		No		
FK	order_id	SMALLINT		No		
	date_id	TIMESTAMP		No		
	provider_id	CHARACTER(10)		No		
	report_num	CHARACTER VARYING(10)		No		
	responsible	CHARACTER VARYING(30)		No		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_provider_op_grl_reception	op_grl_cat_provider	provider_id	op_grl_reception	provider_id	Cero o Más
op_grl_document_type_op_grl_reception	op_grl_document_type	document_type_id	op_grl_reception	document_type_id	One Or More
op_grl_reception_op_grl_reception_detail	op_grl_reception	store_id,reception_id	op_grl_reception_detail	store_id,reception_id	One Or More

**Tabla: op\_grl\_reception\_detail**

## Detalles de la tabla:

Descripción	Tabla que almacena los datos de los productos que se registran en la recepción después de ser confirmada.
Nombre de la restricción de llave primaria	PK_op_grl_reception_detail

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	reception_id	INTEGER		Yes		
PK,FK	store_id	SMALLINT		Yes		
PK,FK	provider_product_code	CHARACTER(10)		Yes		
	received_quantity	NUMERIC(12,2)		No		
FK	difference_id	CHARACTER(2)		Yes		
	unit_cost	NUMERIC(12,2)		No		
PK,FK	provider_id	CHARACTER(10)		Yes		
	sort_num	INTEGER		No		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_difference_op_grl_reception_detail	op_grl_cat_difference	difference_id	op_grl_reception_detail	difference_id	Cero o Más
op_grl_cat_providers_product_op_grl_remission_detail2	op_grl_cat_providers_product	provider_id,provider_product_code	op_grl_reception_detail	provider_id,provider_product_code	Cero o Más
op_grl_reception_op_grl_reception_detail	op_grl_reception	store_id,reception_id	op_grl_reception_detail	store_id,reception_id	One Or More

**Tabla: op\_grl\_remission**

## Detalles de la tabla:

Descripción	Tabla que almacena a remisión de los productos a recibir por parte del restaurante.
Nombre de la restricción de llave primaria	PK_op_grl_remission

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	remission_id	CHARACTER(10)		Yes		
PK,FK	store_id	SMALLINT		Yes		
PK,FK	provider_id	CHARACTER(10)		Yes		
FK	order_id	INTEGER		Yes		
	date_id	TIMESTAMP		No		

delivey_date	TIMESTAMP	No			
Relaciones:					
Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_provider_op_grl_remission	op_grl_cat_provider	provider_id	op_grl_remission	provider_id	Cero o Más
op_grl_order_op_grl_remission	op_grl_order	order_id,store_id	op_grl_remission	order_id,store_id	One Or More
op_grl_remission_op_grl_remission_detail	op_grl_remission	remission_id,store_id,provider_id	op_grl_remission_detail	remission_id,store_id,provider_id	One Or More

**Tabla: op\_grl\_remission\_detail**

Detalles de la tabla:

Descripción	Tabla que almacena los productos que se tienen en la remisión.
Nombre de la restricción de llave primaria	PK_op_grl_remission_detail

Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	remission_id	CHARACTER(10)		Yes		
PK,FK	store_id	SMALLINT		Yes		
PK	provider_product_code_order	CHARACTER(10)		Yes		
PK	provider_product_code_remission	CHARACTER(10)		Yes		
	required_quantity	NUMERIC(12,2)		No		
FK	unit_measure_remission	CHARACTER(4)		No		
	unit_cost	NUMERIC(12,2)		No		
	difference	BOOLEAN		No		
PK,FK	provider_id	CHARACTER(10)		Yes		
	sort_num	INTEGER		No		

Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_providers_pro duct_op_grl_remission_detail	op_grl_cat_providers_pro duct	provider_id	op_grl_remission_detail	provider_id	Cero o Más
op_grl_cat_unit_measure_op_grl_remission_detail	op_grl_cat_unit_measure	unit_id	op_grl_remission_detail	unit_measure_remission	Cero o Más
op_grl_remission_op_grl_remission_detail	op_grl_remission	remission_id,store_id,provider_id	op_grl_remission_detail	remission_id,store_id,provider_id	One Or More

**Tabla: op\_grl\_step\_order**

Detalles de la tabla:

Descripción	Tabla de paso cuando una orden aún no es confirmada. Esta tabla solo almacena una orden que es eliminada al momento de confirmarla.
Nombre de la restricción de llave primaria	PK_op_grl_step_order

Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	order_id	INTEGER		Yes		
PK	store_id	SMALLINT		Yes		
	date_id	TIMESTAMP		No		
	date_limit	TIMESTAMP		No		

Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_step_order_op_grl_step_order_detail	op_grl_step_order	order_id,store_id	op_grl_step_order_detail	order_id,store_id	One Or More

**Tabla: op\_grl\_step\_order\_detail**

Detalles de la tabla:

Descripción	Tabla que almacena los productos que se desea ordenar, pero solo temporalmente hasta que se confirme el pedido.
Nombre de la restricción de llave primaria	PK_op_grl_step_order_detail

Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	order_id	INTEGER		Yes		
PK,FK	store_id	SMALLINT		Yes		
PK,FK	provider_product_code	CHARACTER(10)		Yes		
PK,FK	provider_id	CHARACTER(10)		Yes		
FK	provider_unit	CHARACTER(4)		No		

unit_cost	NUMERIC(12,2)	No
inv_required_quantity	NUMERIC(12,2)	No
prv_required_quantity	NUMERIC(12,2)	No

## Relaciones:

Nombre de la restricción de llave foránea	de Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_providers_product_op_grl_step_order_detail	op_grl_cat_providers_product	provider_id,provider_duct_code	op_grl_step_order_detail	provider_id,provider_duct_code	Cero o Más
op_grl_cat_unit_measure_op_grl_step_order_detail	op_grl_cat_unit_measure	unit_id	op_grl_step_order_detail	provider_unit	Cero o Más
op_grl_step_order_op_grl_step_order_detail	op_grl_step_order	order_id,store_id	op_grl_step_order_detail	order_id,store_id	One Or More

**Tabla: op\_grl\_step\_reception**

## Detalles de la tabla:

Descripción	Tabla que almacena temporalmente los datos de una recepción, antes de que sea confirmada.
Nombre de la restricción de llave primaria	PK_op_grl_step_reception

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	reception_id	INTEGER		Yes		
PK	store_id	SMALLINT		Yes		
	document_num	CHARACTER(10)		No		
FK	document_type_id	SMALLINT		No		
	remission_id	CHARACTER(10)		No		
	order_id	INTEGER		No		
	date_id	TIMESTAMP		No		
FK	provider_id	CHARACTER(10)		No		
	report_num	CHARACTER VARYING(10)		No		
	responsible	CHARACTER VARYING(30)		No		

## Relaciones:

Nombre de la restricción de llave foránea	de Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_provider_op_grl_step_reception	op_grl_cat_provider	provider_id	op_grl_step_reception	provider_id	Cero o Más
op_grl_document_type_op_grl_step_reception	op_grl_document_type	document_type_id	op_grl_step_reception	document_type_id	Cero o Más
op_grl_step_reception_op_grl_step_reception_detail	op_grl_step_reception	store_id,reception_id	op_grl_step_reception_detail	store_id,reception_id	Cero o Más

**Tabla: op\_grl\_step\_reception\_detail**

## Detalles de la tabla:

Descripción	Tabla que almacena de manera temporal los productos que se registran en la recepción, antes de ser confirmada.
Nombre de la restricción de llave primaria	PK_op_grl_step_reception_detail

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	reception_id	INTEGER		Yes		
PK,FK	store_id	SMALLINT		Yes		
PK,FK	provider_product_code	CHARACTER(10)		Yes		
	received_quantity	NUMERIC(7,2)		No		
	unit_cost	NUMERIC(7,2)		No		
PK,FK	provider_id	CHARACTER(10)		Yes		
	sort_num	INTEGER		No		
FK	difference_id	CHARACTER(2)		No		

## Relaciones:

Nombre de la restricción de llave foránea	de Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_difference_op_grl_step_reception_detail	op_grl_cat_difference	difference_id	op_grl_step_reception_detail	difference_id	Cero o Más
op_grl_cat_providers_product_op_grl_step_reception_detail	op_grl_cat_providers_product	provider_id,provider_product_code	op_grl_step_reception_detail	provider_id,provider_product_code	Cero o Más
op_grl_step_reception_op_grl_step_reception_detail	op_grl_step_reception	store_id,reception_id	op_grl_step_reception_detail	store_id,reception_id	Cero o Más

**Tabla: op\_grl\_step\_transfer**

Detalles de la tabla:

Descripción	Tabla que almacena temporalmente los datos de una transferencia, antes de que sea confirmada
Nombre de la restricción de llave primaria	PK_op_grl_step_transfer

Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	transfer_id	INTEGER		Yes		
	local_store_id	SMALLINT		No		
	neighbor_store_id	SMALLINT		No		
	date_id	TIMESTAMP		No		
	transfer_type	SMALLINT		No		

Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_step_transfer_op_grl_step_transfer_detail	op_grl_step_transfer	transfer_id	op_grl_step_transfer_detail	transfer_id	Cero o Más

**Tabla: op\_grl\_step\_transfer\_detail**

Detalles de la tabla:

Descripción	Tabla que almacena de manera temporal los productos que se registran en la transferencia, antes de ser confirmada
Nombre de la restricción de llave primaria	PK_op_grl_step_transfer_detail

Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	transfer_id	INTEGER		Yes		
PK	inv_id	CHARACTER(6)		Yes		
	stock_code_id	CHARACTER(6)		Yes		
	provider_product_code	CHARACTER(10)		No		
	provider_quantity	NUMERIC(12,2)		Yes		
	inventory_quantuty	NUMERIC(12,2)		Yes		
	prv_conversion_factor	NUMERIC(12,2)		Yes		
	provider_unit_measure	CHARACTER(50)		No		
	inventory_unit_measure	CHARACTER(50)		No		
	provider_id	CHARACTER(10)		No		
	existence	DOUBLE PRECISION		No		

Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_step_transfer_op_grl_step_transfer_detail	op_grl_step_transfer	transfer_id	op_grl_step_transfer_detail	transfer_id	Cero o Más

**Tabla: op\_grl\_suggested\_order**

Detalles de la tabla:

Descripción	Tabla que almacena el pedido sugerido generado desde la aplicación, este pedido sugerido es el mismo que se genera dentro de FMS.
Nombre de la restricción de llave primaria	PK_op_grl_suggested_order

Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	store_id	SMALLINT		Yes		
PK,FK	inv_id	CHARACTER(6)		Yes		
	available_quantity	NUMERIC(12,2)		No		
	required	NUMERIC(12,2)		No		
	suggested_quantity	NUMERIC(12,2)		No		
PK,FK	order_id	INTEGER		Yes		

Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_inventory_op_grl_suggested_order	op_grl_cat_inventory	inv_id	op_grl_suggested_order	inv_id	Cero o Más
op_grl_order_op_grl_suggested_order	op_grl_order	order_id,store_id	op_grl_suggested_order	order_id,store_id	Cero o Más

**Tabla: op\_grl\_suggested\_transfer**

Detalles de la tabla:

Descripción	Tabla que almacena los valores de pedido sugerido para la siguiente semana de la fecha en la que se invoque su carga.
Nombre de la restricción de llave primaria	PK_op_grl_suggested_transfer

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	local_store_id	SMALLINT		Yes		
PK,FK	inv_id	CHARACTER(6)		Yes		
	available_quantity	NUMERIC(12,2)		No		
	required	NUMERIC(12,2)		No		
	suggested_quantity	NUMERIC(12,2)		No		
PK	transfer_id	INTEGER		Yes		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_inventory_op_grl_suggested_transfer	op_grl_cat_inventory	inv_id	op_grl_suggested_transfer	inv_id	Cero o Más

**Tabla: op\_grl\_transfer**

## Detalles de la tabla:

Descripción	Tabla que almacena temporalmente los datos de una transferencia cuando ha sido confirmada.
Nombre de la restricción de llave primaria	PK_op_grl_transfer

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	transfer_id	INTEGER		Yes		
	local_store_id	SMALLINT		Yes		
	neighbor_store_id	SMALLINT		Yes		
	date_id	TIMESTAMP		No		
	transfer_type	SMALLINT		No		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_transfer_op_grl_transfer_detail	op_grl_transfer	transfer_id	op_grl_transfer_detail	transfer_id	Cero o Más
ss_cat_neighbor_store_op_grl_transfer	ss_cat_neighbor_store		op_grl_transfer		Cero o Más

**Tabla: op\_grl\_transfer\_detail**

## Detalles de la tabla:

Descripción	Tabla que almacena de manera temporal los productos que se registran en la transferencia cuando ha sido confirmada.
Nombre de la restricción de llave primaria	PK_op_grl_transfer_detail

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	transfer_id	INTEGER		Yes		
PK	inv_id	CHARACTER(6)		Yes		
	stock_code_id	CHARACTER(6)		Yes		
	provider_product_code	CHARACTER(10)		Yes		
	provider_quantity	NUMERIC(12,2)		Yes		
	inventory_quantity	NUMERIC(12,2)		Yes		
	prv_conversion_factor	NUMERIC(12,2)		Yes		
	provider_unit_measure	CHARACTER(50)		No		
	inventory_unit_measure	CHARACTER(50)		No		
	provider_id	CHARACTER(10)		No		
	existence	DOUBLE PRECISION		No		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_transfer_op_grl_transfer_detail	op_grl_transfer	transfer_id	op_grl_transfer_detail	transfer_id	Cero o Más

**Tabla: op\_grl\_way\_order**

## Detalles de la tabla:

Descripción	Tabla que almacena la información de los productos que se encuentran en tránsito para recalcular el pedido sugerido.
Nombre de la restricción de llave primaria	PK_op_grl_way_order

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	order_id	INTEGER		Yes		
PK,FK	store_id	SMALLINT		Yes		
PK,FK	provider_product_code	CHARACTER(10)		Yes		
PK,FK	provider_id	CHARACTER(10)		Yes		
	way_quantity	NUMERIC(12,2)		Yes		



## Relaciones:

Nombre de restricción de la Padre de llave foránea	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_providers_product roduct_op_grl_way_order	provider_id,provider_pro uct_code	op_grl_way_ order	provider_id,provider_pro uct_code	Cero o Más
op_grl_order_op_grl_w ay_order	order_id,store_id	op_grl_way_ order	order_id,store_id	Cero o Más

**Tabla: op\_inv\_conversion\_factor**

## Detalles de la tabla:

Descripción	Tabla que almacena los factores de conversión entre las unidades de proveedor y las de las recetas.
Nombre de la restricción de llave primaria	PK_op_inv_conversion_factor

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	inv_id	CHARACTER(6)		Yes		
	rcp_conversion_factor	NUMERIC(12,2)		No		
	prv_conversion_factor	NUMERIC(12,2)		No		
	prv_unit_measure	CHARACTER(4)		No		
	inv_unit_measure	CHARACTER(4)		No		

**Tabla: op\_inv\_exceptions**

## Detalles de la tabla:

Descripción	Tabla que almacena parámetros para determinar la eficiencia de un producto.
Nombre de la restricción de llave primaria	PK_op_inv_exceptions

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	inv_id	CHARACTER(6)		Yes		
PK	family_id	CHARACTER(10)		Yes		
	min_efficiency	DOUBLE PRECISION		No		
	max_efficiency	DOUBLE PRECISION		No		
	max_variance	DOUBLE PRECISION		No		

**Tabla: op\_inv\_inventory\_begin**

## Detalles de la tabla:

Descripción	Tabla que almacena los valores del inventario inicial de la semana tomándolos del archivo de inventario.
Nombre de la restricción de llave primaria	PK_op_inv_inventory_begin

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	inv_id	CHARACTER(6)		Yes		
	inv_begin	DOUBLE PRECISION		No		
	week_no	SMALLINT		No		

**Tabla: op\_inv\_inventory\_detail**

## Detalles de la tabla:

Descripción	Tabla que almacena la información relacionada con el inventario semanal cuando ya fueron guardados los cambios en el campo de inventario final.
Nombre de la restricción de llave primaria	PK_op_inv_inventory_detail

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	inv_id	CHARACTER(6)		Yes		
	inv_beg	NUMERIC(12,2)		No		
	receptions	NUMERIC(12,2)		No		
	itransfers	NUMERIC(12,2)		No		
	otransfers	NUMERIC(12,2)		No		
	prv_inv_end	NUMERIC(12,2)		No		
	inv_inv_end	NUMERIC(12,2)		No		
	rec_inv_end	NUMERIC(12,2)		No		
	inv_end	NUMERIC(12,2)		No		

	decreases	NUMERIC(12,2)	No
	ideal_use	NUMERIC(12,2)	No
	real_use	NUMERIC(12,2)	No
	unit_cost	NUMERIC(12,2)	No
	prv_conversion_factor	NUMERIC(12,2)	No
	rcp_conversion_factor	NUMERIC(12,2)	No
	prv_unit_measure	CHARACTER(4)	No
	inv_unit_measure	CHARACTER(4)	No
	misc	BOOLEAN	No
	date_id	TIMESTAMP	Yes
PK	year_no	SMALLINT	Yes
PK	period_no	SMALLINT	Yes
PK	week_no	SMALLINT	Yes

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_inventory_op_inv_inventory_detail	op_grl_cat_inventory	inv_id	op_inv_inventory_detail	inv_id	Cero o Más

**Tabla: op\_inv\_inventory\_mov**

## Detalles de la tabla:

Descripción Tabla que almacena temporalmente los movimientos de recepciones y transferencias en el inventario hasta pasarlos a op\_inv\_inventory\_detail.

Nombre de la restricción de llave primaria PK\_op\_inv\_inventory\_mov

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	inv_id	CHARACTER(6)		Yes		
PK	provider_product_code	CHARACTER(10)		Yes		
	receptions	CHARACTER(40)		No		
	input_transfers	CHARACTER(40)		No		
	output_transfers	CHARACTER(40)		No		

**Tabla: op\_inv\_step\_inventory\_detail**

## Detalles de la tabla:

Descripción Tabla que almacena la información relacionada con el inventario semanal cuando no han guardados los cambios en el campo de inventario final

Nombre de la restricción de llave primaria PK\_op\_inv\_step\_inventory\_detail

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK,FK	inv_id	CHARACTER(6)		Yes		
	inv_beg	NUMERIC(12,2)		No		
	receptions	NUMERIC(12,2)		No		
	itransfers	NUMERIC(12,2)		No		
	otransfers	NUMERIC(12,2)		No		
	prv_inv_end	NUMERIC(12,2)		No		
	inv_inv_end	NUMERIC(12,2)		No		
	rec_inv_end	NUMERIC(12,2)		No		
	inv_end	NUMERIC(12,2)		No		
	decreases	NUMERIC(12,2)		No		
	ideal_use	NUMERIC(12,2)		No		
	real_use	NUMERIC(12,2)		No		
	unit_cost	NUMERIC(12,2)		No		
	prv_conversion_factor	NUMERIC(12,2)		No		
	rcp_conversion_factor	NUMERIC(12,2)		No		
	prv_unit_measure	CHARACTER(4)		No		
	inv_unit_measure	CHARACTER(4)		No		
	misc	BOOLEAN		No		
	date_id	TIMESTAMP		Yes		
PK	year_no	CHARACTER(40)		Yes		
PK	period_no	CHARACTER(40)		Yes		
PK	week_no	CHARACTER(40)		Yes		

## Relaciones:

Nombre de la restricción de llave foránea	Padre	Columnas	Hijo	Columnas	Cardinalidad
op_grl_cat_inventory_op_inv_step_inventory_detail	op_grl_cat_inventory	inv_id	op_inv_step_inventory_detail	inv_id	Cero o Más

**Tabla: ss\_cat\_neighbor\_store**

## Detalles de la tabla:

Descripción Tabla que almacena el catálogo de todos los restaurantes a los que se puede hacer ó se puede recibir una transferencia.

Nombre de la restricción de llave primaria PK\_ss\_cat\_neighbor\_store

## llave primaria

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	store_id	SMALLINT		Yes		
	store_desc	CHARACTER VARYING(50)		No		
	area_id	CHARACTER VARYING(10)		No		

## Relaciones:

Nombre de la restricción de llave foránea	Nombre de la restricción de llave Padre	Columnas	Hijo	Columnas	Cardinalidad
ss_cat_neighbor_store_op_grl_transfer	ss_cat_neighbor_store		op_grl_transfer		Cero o Más

**Tabla: ss\_cat\_time**

## Detalles de la tabla:

Descripción Tabla que almacena toda la información relacionada con el calendario del restaurante (que no es el mismo que el normal) de forma anual.

Nombre de la restricción de llave primaria PK\_ss\_cat\_time

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	date_id	TIMESTAMP		Yes		
FK	weekday_id	SMALLINT		Yes		0
	year_no	SMALLINT		Yes		0
	quarter_no	SMALLINT		Yes		0
	period_no	SMALLINT		Yes		0
	week_no	SMALLINT		Yes		0

## Relaciones:

Nombre de la restricción de llave foránea	Nombre de la restricción de Padre	Columnas	Hijo	Columnas	Cardinalidad
ss_cat_yum_day_ss_cat_time	ss_cat_yum_day	weekday_id	ss_cat_time	weekday_id	Cero o Más

**Tabla: ss\_cat\_yum\_day**

## Detalles de la tabla:

Descripción Tabla que almacena los días de la semana. Iniciando en martes.

Nombre de la restricción de llave primaria PK\_ss\_cat\_yum\_day

## Columnas:

Llave	Nombre del campo	Tipo de dato	Dominio	No nulo	Check	Default
PK	weekday_id	SMALLINT		Yes		0
	weekday_desc	CHARACTER(15)		No		

## Relaciones:

Nombre de la restricción de llave foránea	Nombre de la restricción de llave Padre	Columnas	Hijo	Columnas	Cardinalidad
ss_cat_yum_day_op_grl_order_limit	ss_cat_yum_day	weekday_id	op_grl_order_limit	weekday_id	Cero o Más
ss_cat_yum_day_ss_cat_time	ss_cat_yum_day	weekday_id	ss_cat_time	weekday_id	Cero o Más

**Detalles de las columnas****Columna: active\_flag**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
Descripción	Bandera para indicar si un producto se encuentra activo (t) o es obsoleto (f)
Llave primaria	No
Referencia a	
Tipo de dato	BOOLEAN
Tipo de dato en el script	BOOLEAN
Dominio	

**Columna: active\_flag**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Bandera que nos indica si un proveedor se encuentra activo (t) ó no (f).
Llave primaria	No
Referencia a	
Tipo de dato	BOOLEAN
Tipo de dato en el script	BOOLEAN
Dominio	

**Columna: active\_flag**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
Descripción	Bandera que indica si el producto esta activo (t) ó no (f)
Llave primaria	No
Referencia a	
Tipo de dato	BOOLEAN
Tipo de dato en el script	BOOLEAN
Dominio	

**Columna: area\_id**

Detalles de las columnas:

Nombre de la tabla	ss_cat_neighbor_store
Descripción	Área a la que pertenece el restaurante.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(10)
Tipo de dato en el script	CHARACTER VARYING(10)
Dominio	

**Columna: available\_quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_transfer
Descripción	Cantidad disponible del producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: available\_quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_order
Descripción	Cantidad disponible del producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: bank\_account**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Número de cuenta bancaria del proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(20)
Tipo de dato en el script	CHARACTER VARYING(20)
Dominio	

**Columna: case\_desc**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_recep_cases
Descripción	Descripción del caso de la recepción
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(25)
Tipo de dato en el script	CHARACTER VARYING(25)
Dominio	

**Columna: case\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_recep_cases
Descripción	Identificador único para el caso.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: case\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_config_difference
Descripción	
Llave primaria	Yes
Referencia a	case_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: company\_desc**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_product_company
Descripción	Nombre de la compañía
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(20)
Tipo de dato en el script	CHARACTER VARYING(20)
Dominio	

**Columna: company\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_product_company
Descripción	
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(3)
Tipo de dato en el script	CHARACTER(3)
Dominio	

**Columna: company\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
Descripción	Llave foránea con la tabla op_grl_cat_product_company para determinar a que marca puede ser surtido este producto de inventario.
Llave primaria	No
Referencia a	company_id
Tipo de dato	CHARACTER(3)
Tipo de dato en el script	CHARACTER(3)

Dominio

**Columna: confirmation\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_confirmation
Descripción	
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: confirmation\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Llave foránea con la tabla op_grl_cat_confirmation.
Llave primaria	No
Referencia a	confirmation_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: contact**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Nombre del contacto hacia el proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(25)
Tipo de dato en el script	CHARACTER VARYING(25)
Dominio	

**Columna: conversion\_factor**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
Descripción	Factor de conversión entre las unidades de proveedor y las unidades de inventario. Ejemplo, 1 caja (unidad de proveedor) son 1000 vasos (unidad de inventario), por lo que el factor de conversión sería 1000.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: date\_id**

## Detalles de las columnas:

Nombre de la tabla	ss_cat_time
Descripción	Fecha del calendario. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

## Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Fecha en la que se registra los valores de inventario final.
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Fecha en la que se registra los valores de inventario final.
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP

Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_transfer
Descripción	Fecha en la que se registra la transferencia.
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer
Descripción	Fecha de la transferencia.
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Fecha en la que se registra la recepción.
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_order
Descripción	Fecha en la que se realiza la orden.
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_remission
Descripción	Fecha en la que es cargada la remisión dentro del sistema.
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception
Descripción	Fecha en la que fue realizada la recepción.
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_order
Descripción	Fecha de confirmación de la recepción.
Llave primaria	No
Referencia a	

Tipo de dato	TIMESTAMP
--------------	-----------

Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_financial_mov
Descripción	Fecha del movimiento financiero.
Llave primaria	Yes
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_limit**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_order
Descripción	
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: date\_limit**

Detalles de las columnas:

Nombre de la tabla	op_grl_order
Descripción	
Llave primaria	No
Referencia a	
Tipo de dato	TIMESTAMP
Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: decreases**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Merma del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: decreases**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Merma del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: delegation**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	delegation
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: delivey\_date**

Detalles de las columnas:

Nombre de la tabla	op_grl_remision
Descripción	Fecha en la que se entregará el producto de esta remisión
Llave primaria	No
Referencia a	

Tipo de dato	TIMESTAMP
--------------	-----------



Tipo de dato en el script	TIMESTAMP
Dominio	

**Columna: Descripción**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_confirmation
Descripción	Descripción del tipo de confirmación.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(20)
Tipo de dato en el script	CHARACTER VARYING(20)
Dominio	

**Columna: Descripción**

Detalles de las columnas:	
Nombre de la tabla	op_grl_document_type
Descripción	Tipo de documento.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(20)
Tipo de dato en el script	CHARACTER VARYING(20)
Dominio	

**Columna: difference**

Detalles de las columnas:	
Nombre de la tabla	op_grl_remission_detail
Descripción	Bandera para indicar si hay una diferencia entre la orden y la remisión (t) ó no la hay.
Llave primaria	No
Referencia a	
Tipo de dato	BOOLEAN
Tipo de dato en el script	BOOLEAN
Dominio	

**Columna: difference\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_difference
Descripción	
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(2)
Tipo de dato en el script	CHARACTER(2)
Dominio	

**Columna: difference\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_step_reception_detail
Descripción	
Llave primaria	No
Referencia a	difference_id
Tipo de dato	CHARACTER(2)
Tipo de dato en el script	CHARACTER(2)
Dominio	

**Columna: difference\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_config_difference
Descripción	
Llave primaria	Yes
Referencia a	difference_id
Tipo de dato	CHARACTER(2)
Tipo de dato en el script	CHARACTER(2)
Dominio	

**Columna: difference\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_reception_detail
Descripción	Llave foránea con la tabla op_grl_cat_difference para determinar el código de discrepancia del producto recibido.
Llave primaria	No
Referencia a	difference_id

Tipo de dato	CHARACTER(2)
Tipo de dato en el script	CHARACTER(2)
Dominio	

**Columna: diff\_prv**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Diferencia en unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	TEXT
Tipo de dato en el script	TEXT
Dominio	

**Columna: dif\_desc**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_difference
Descripción	
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: dif\_desc**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Código de discrepancia del producto.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(40)
Tipo de dato en el script	CHARACTER VARYING(40)
Dominio	

**Columna: dif\_inv**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Diferencia en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	TEXT
Tipo de dato en el script	TEXT
Dominio	

**Columna: document\_num**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Número de documento del cual se recibe (si se tiene la remisión ese es el número).
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: document\_num**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception
Descripción	Número de documento del cual se hizo la recepción (Factura o Remisión)
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: document\_type\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Clave del tipo de documento.
Llave primaria	No

Referencia a	document_type_id
--------------	------------------

Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: document\_type\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception
Descripción	Llave foránea con la tabla op_grl_document_type para determinar si se captura de una factura ó remisión.
Llave primaria	No
Referencia a	document_type_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: document\_type\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Tipo de documento que emite el proveedor. Llave foránea con la tabla op_grl_document_type
Llave primaria	No
Referencia a	document_type_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: document\_type\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_document_type
Descripción	Identificador único del tipo de documento.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: email**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Email del contacto hacia el proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(40)
Tipo de dato en el script	CHARACTER VARYING(40)
Dominio	

**Columna: existence**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Existencia actual del producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	DOUBLE PRECISION
Tipo de dato en el script	DOUBLE PRECISION
Dominio	

**Columna: existence**

Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Existencia actual del producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	DOUBLE PRECISION
Tipo de dato en el script	DOUBLE PRECISION
Dominio	

**Columna: family\_desc**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_family
Descripción	Nombre de la familia
Llave primaria	No

Referencia a	
Tipo de dato	CHARACTER VARYING(20)
Tipo de dato en el script	CHARACTER VARYING(20)
Dominio	

**Columna: family\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
Descripción	Llave foránea con la table op_grl_cat_family para determinar a qué familia pertenece el producto.
Llave primaria	No
Referencia a	family_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: family\_id**

Detalles de las columnas:

Nombre de la tabla	op_inv_exceptions
Descripción	Llave foránea con la table op_grl_cat_family para determinar a qué familia pertenece el producto. Parte de la llave primaria.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: family\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_family
Descripción	
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: family\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_existence
Descripción	Clave de la familia a la que pertenece el producto del inventario.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: family\_order**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_family
Descripción	
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: finantial\_code**

Detalles de las columnas:

Nombre de la tabla	op_grl_finantial_mov
Descripción	Código financiero del movimiento.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(40)
Tipo de dato en el script	CHARACTER(40)
Dominio	

**Columna: frequency\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_frequency
--------------------	----------------------

Descripción	
-------------	--

Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: frequency\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
Descripción	Llave foránea con la tabla op_grl_cat_frequency para determinar la frecuencia con que este producto es inventariado.
Llave primaria	No
Referencia a	frequency_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: frec\_desc**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_frequency
Descripción	Llave primaria del catálogo.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(20)
Tipo de dato en el script	CHARACTER VARYING(20)
Dominio	

**Columna: ideal\_use**

## Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Uso ideal del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: ideal\_use**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Uso ideal del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: input\_transfers**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_mov
Descripción	Transferencias de entrada de la semana.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(40)
Tipo de dato en el script	CHARACTER(40)
Dominio	

**Columna: inventory\_quantity**

## Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Cantidad transferida ó a transferir en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inventory\_quantity**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
--------------------	-----------------------------

Descripción	Cantidad transferida ó a transferir en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inventory\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Unidad de medida del inventario.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(50)
Tipo de dato en el script	CHARACTER(50)
Dominio	

**Columna: inventory\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Unidad de medida del inventario.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(50)
Tipo de dato en el script	CHARACTER(50)
Dominio	

**Columna: inv\_beg**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Inventario inicial del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inv\_beg**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Inventario inicial del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inv\_begin**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_begin
Descripción	Inventario inicial del producto.
Llave primaria	No
Referencia a	
Tipo de dato	DOUBLE PRECISION
Tipo de dato en el script	DOUBLE PRECISION
Dominio	

**Columna: inv\_desc**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
Descripción	Descripción del producto de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: inv\_end**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
--------------------	------------------------------

Descripción	Total de inventario final en unidad de inventario (Sumando las cantidades de los 3 campos anteriores con sus respectivas conversiones).
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inv\_end**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Total de inventario final en unidad de inventario (Sumando las cantidades de los 3 campos anteriores con sus respectivas conversiones).
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inv\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Clave del producto de inventario que se esta transfiriendo. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

## Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Código de inventario de un producto. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	inv_id
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_mov
Descripción	Código de inventario de un producto. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
Descripción	Código de inventario de un producto. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Código de inventario de un producto. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	inv_id
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_begin
--------------------	------------------------

Descripción	Código de inventario de un producto. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

Detalles de las columnas:	
Nombre de la tabla	op_inv_exceptions
Descripción	Código de inventario de un producto. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

Detalles de las columnas:	
Nombre de la tabla	op_inv_conversion_factor
Descripción	Código de inventario de un producto. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_existence
Descripción	Código de inventario de un producto. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_transfer_detail
Descripción	Clave del producto de inventario que se esta transfiriendo. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_suggested_transfer
Descripción	Código de inventario de un producto. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	inv_id
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_suggested_order
Descripción	Código de inventario del producto que esta en el pedido sugerido.
Llave primaria	Yes
Referencia a	inv_id
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_providers_product



Descripción	Llave foránea con la tabla op_grl_cat_inventory, para saber a qué producto de inventario se liga este producto de proveedor.
Llave primaria	No
Referencia a	inv_id
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: inv\_inv\_end**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Cantidad de producto final en unidad de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inv\_inv\_end**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Cantidad de producto final en unidad de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inv\_required\_quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_order_detail
Descripción	Cantidad requerida de producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inv\_required\_quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_order_detail
Descripción	Cantidad de producto requerido en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: inv\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Clave de la unidad de medida del inventario.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: inv\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Clave de la unidad de medida del inventario.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: inv\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
--------------------	----------------------

Descripción	No
Llave primaria	unit_id
Referencia a	
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: inv\_unit\_measure**

Detalles de las columnas:	
Nombre de la tabla	op_inv_conversion_factor
Descripción	Clave de la unidad de medida del inventario
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: ip\_phone**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_provider
Descripción	Teléfono IP del contacto hacia el proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(6)
Tipo de dato en el script	CHARACTER VARYING(6)
Dominio	

**Columna: itransfers**

Detalles de las columnas:	
Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Transferencias de entrada del producto acumuladas a la semana.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: itransfers**

Detalles de las columnas:	
Nombre de la tabla	op_inv_inventory_detail
Descripción	Transferencias de entrada del producto acumuladas a la semana.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: local\_store\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_transfer
Descripción	Clave del restaurante local.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: local\_store\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_suggested_transfer
Descripción	Clave del restaurante local. Parte de la llave primaria de la tabla
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: local\_store\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_step_transfer

Descripción	Clave del restaurante local.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: max\_efficiency**

## Detalles de las columnas:

Nombre de la tabla	op_inv_exceptions
Descripción	Porcentaje de eficiencia máxima permitida en el producto.
Llave primaria	No
Referencia a	
Tipo de dato	DOUBLE PRECISION
Tipo de dato en el script	DOUBLE PRECISION
Dominio	

**Columna: max\_variance**

## Detalles de las columnas:

Nombre de la tabla	op_inv_exceptions
Descripción	Rango de varianza máxima para considerar al producto de muy baja ó muy alta eficiencia en su uso.
Llave primaria	No
Referencia a	
Tipo de dato	DOUBLE PRECISION
Tipo de dato en el script	DOUBLE PRECISION
Dominio	

**Columna: min\_efficiency**

## Detalles de las columnas:

Nombre de la tabla	op_inv_exceptions
Descripción	Porcentaje de eficiencia mínima permitida en el producto.
Llave primaria	No
Referencia a	
Tipo de dato	DOUBLE PRECISION
Tipo de dato en el script	DOUBLE PRECISION
Dominio	

**Columna: misc**

## Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Bandera que indica si un producto es misceláneo (no tiene uso ideal).
Llave primaria	No
Referencia a	
Tipo de dato	BOOLEAN
Tipo de dato en el script	BOOLEAN
Dominio	

**Columna: misc**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Bandera que indica si un producto es misceláneo (no tiene uso ideal).
Llave primaria	No
Referencia a	
Tipo de dato	BOOLEAN
Tipo de dato en el script	BOOLEAN
Dominio	

**Columna: name**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Nombre del proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: neighbor\_store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_transfer
--------------------	-----------------

Descripción	Clave del restaurante de donde se hace ó se recibe una transferencia.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: neighbor\_store\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_step_transfer
Descripción	Clave del restaurante de donde se hace ó se recibe una transferencia.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: neighbourhood**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_provider
Descripción	Referencias de calles en donde se ubica el proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: num\_days\_advance**

Detalles de las columnas:	
Nombre de la tabla	op_grl_days_advance_order
Descripción	Número de días para permitir el adelanto de un pedido.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(40)
Tipo de dato en el script	CHARACTER VARYING(40)
Dominio	

**Columna: order\_equivalent**

Detalles de las columnas:	
Nombre de la tabla	op_grl_difference
Descripción	Equivalencia de la cantidad requerida en la orden del producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	TEXT
Tipo de dato en el script	TEXT
Dominio	

**Columna: order\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_order_detail
Descripción	Llave foránea con la tabla op_grl_order para determinar a qué pedido pertenecen los productos almacenados en esta tabla. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	order_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: order\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_reception
Descripción	Clave de la orden de la cual se hizo la recepción en caso de existir.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: order\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_remission

Descripción	Llave foránea con la tabla op_grl_order para determinar si la remisión de asocia a una orden. De no estar asociada, este campo tendrá el valor de cero.
Llave primaria	No
Referencia a	order_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: order\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_order
Descripción	Identificador único del pedido.
Llave primaria	Yes
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: order\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_way_order
Descripción	Llave foránea con la tabla op_grl_order. Parte de la llave primaria.
Llave primaria	Yes
Referencia a	order_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: order\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_order
Descripción	Orden a la que se asociará el pedido sugerido.
Llave primaria	Yes
Referencia a	order_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: order\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Número de orden de la cual se hace la recepción (en caso de existir).
Llave primaria	No
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: order\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_order
Descripción	Identificador único de la tabla, número de orden asignada por el sistema.
Llave primaria	Yes
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: order\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_order_detail
Descripción	Número de orden asignada por el sistema.
Llave primaria	Yes
Referencia a	order_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: order\_product**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
--------------------	-------------------

Descripción	Código del producto en la orden.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: ord\_cost**

Detalles de las columnas:	
Nombre de la tabla	op_grl_difference
Descripción	Costo en la orden del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC
Tipo de dato en el script	NUMERIC
Dominio	

**Columna: otrangers**

Detalles de las columnas:	
Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Transferencias de salida del producto acumuladas a la semana.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: otrangers**

Detalles de las columnas:	
Nombre de la tabla	op_inv_inventory_detail
Descripción	Transferencias de salida del producto acumuladas a la semana.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: output\_transfers**

Detalles de las columnas:	
Nombre de la tabla	op_inv_inventory_mov
Descripción	Transferencias de salida de la semana.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(40)
Tipo de dato en el script	CHARACTER(40)
Dominio	

**Columna: period\_no**

Detalles de las columnas:	
Nombre de la tabla	ss_cat_time
Descripción	Periodo del calendario
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: period\_no**

Detalles de las columnas:	
Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Periodo del cierre de inventario.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(40)
Tipo de dato en el script	CHARACTER(40)
Dominio	

**Columna: period\_no**

Detalles de las columnas:	
Nombre de la tabla	op_inv_inventory_detail

Descripción	Periodo del cierre de inventario.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: phone**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_provider
Descripción	Teléfono del contacto hacia el proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(20)
Tipo de dato en el script	CHARACTER VARYING(20)
Dominio	

**Columna: product\_name**

Detalles de las columnas:	
Nombre de la tabla	op_grl_difference
Descripción	Nombre del producto del cual se tienen diferencias.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: provider\_code\_before**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_provider
Descripción	Código del proveedor anterior.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Clave del proveedor del producto.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_remission
Descripción	Llave foránea con la tabla op_grl_provider para determinar a que proveedor se le asocia esta remisión. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_step_order_detail
Descripción	Clave del proveedor al que se le hace el pedido.
Llave primaria	Yes
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:	
Nombre de la tabla	op_grl_order_detail

Descripción	Llave foránea con la tabla op_grl_cat_providers_product para determinar a que proveedor se le hace el pedido. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception
Descripción	Llave foránea con op_grl_cat_provider para determinar de qué proveedor se hizo la recepción.
Llave primaria	No
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_way_order
Descripción	Llave foránea con la tabla op_grl_cat_providers_product. Parte de la llave primaria.
Llave primaria	Yes
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Código del proveedor del cual se hace la recepción.
Llave primaria	No
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_remission_detail
Descripción	Llave foránea con la tabla op_grl_provider para determinar de que proveedor proviene la remisión.
Llave primaria	Yes
Referencia a	provider_id,provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Clave del proveedor del producto.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception_detail
Descripción	Llave foránea con op_grl_cat-providers_product para determinar el proveedor del cual se esta haciendo la recepción.
Llave primaria	Yes
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
--------------------	------------------------------



Descripción	Identificador único para el proveedor. Parte de la llave primaria de la tabla
Llave primaria	Yes
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Código del proveedor e identificador único.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception_detail
Descripción	Código del proveedor del cual se hace la recepción.
Llave primaria	Yes
Referencia a	provider_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_price**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
Descripción	Precio del proveedor para el producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: provider\_product\_code**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Clave del producto del proveedor de l que se esta transfiriendo.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_mov
Descripción	Código del producto para el proveedor. Parte de la llave primaria.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
Descripción	Código de proveedor para un producto. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code**

## Detalles de las columnas:

Nombre de la tabla	op_grl_order_detail
--------------------	---------------------

Descripción	Llave foránea con la table op_grl_cat_providers_product para determinar qué producto se esta ordenando. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	provider_product_code
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_order_detail
Descripción	Clave del producto del proveedor que se ordena.
Llave primaria	Yes
Referencia a	provider_product_code
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception_detail
Descripción	Llave foránea con la tabla de op_grl_cat_providers_product para determinar qué producto es el que se esta recibiendo. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	provider_product_code
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code**

Detalles de las columnas:

Nombre de la tabla	op_grl_way_order
Descripción	Llave foránea con la tabla op_grl_cat_providers_product. Parte de la llave primaria.
Llave primaria	Yes
Referencia a	provider_product_code
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code**

Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Clave del producto del proveedor de l que se esta transfiriendo.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception_detail
Descripción	Codigo del producto del proveedor. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	provider_product_code
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code\_order**

Detalles de las columnas:

Nombre de la tabla	op_grl_remission_detail
Descripción	Llave foránea con la tabla op_grl_providers_product para determinar la clave del producto de proveedor que se ordenó. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_code\_remis**

## Detalles de las columnas:

Nombre de la tabla	op_grl_remission_detail
Descripción	Llave foránea con la tabla op_grl_providers_product para determinar la clave del producto de proveedor que se recibirá. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: provider\_product\_desc**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
Descripción	Descripción del producto del proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(40)
Tipo de dato en el script	CHARACTER VARYING(40)
Dominio	

**Columna: provider\_quantity**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Cantidad transferida ó a transferir en unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: provider\_quantity**

## Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Cantidad transferida ó a transferir en unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: provider\_unit**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_order_detail
Descripción	Clave de la unidad de medida del producto del proveedor.
Llave primaria	No
Referencia a	unit_id
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: provider\_unit**

## Detalles de las columnas:

Nombre de la tabla	op_grl_order_detail
Descripción	Llave foránea con la tabla op_grl_cat-unit_measure para determinar la unidad de medida del producto del proveedor.
Llave primaria	No
Referencia a	unit_id
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: provider\_unit\_measure**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Unidad de medida del proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(50)
Tipo de dato en el script	CHARACTER(50)

Dominio

**Columna: provider\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
Descripción	Llave foránea con la tabla de op_grl_cat_unit_measure, para saber la unidad asociada al producto del proveedor.
Llave primaria	No
Referencia a	unit_id
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: provider\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Unidad de medida del proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(50)
Tipo de dato en el script	CHARACTER(50)
Dominio	

**Columna: prv\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Factor de conversión entre las unidades de inventario y las unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Factor de conversión de unidad de proveedor a unidad de receta.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Factor de conversión de unidad de proveedor a unidad de receta.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_inv_conversion_factor
Descripción	Factor de conversión entre las unidades de inventario y las unidades de proveedor
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Factor de conversión entre las unidades de inventario y las unidades de proveedor.
Llave primaria	No
Referencia a	

Tipo de dato	NUMERIC(12,2)
--------------	---------------

Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_inv\_end**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Cantidad de producto final en unidad de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_inv\_end**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Cantidad de producto final en unidad de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_required\_quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_order_detail
Descripción	Cantidad de producto requerida en unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_required\_quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_order_detail
Descripción	Cantidad requerida en unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: prv\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Clave de la unidad de medida del proveedor
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: prv\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Clave de la unidad de medida del proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: prv\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_inv_conversion_factor
Descripción	Clave de la unidad de medida del proveedor
Llave primaria	No
Referencia a	

Tipo de dato	CHARACTER(4)
--------------	--------------

Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: qty\_received**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Cantidad de producto recibida en unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	TEXT
Tipo de dato en el script	TEXT
Dominio	

**Columna: qty\_required**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Cantidad requerida del producto en la orden en unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	TEXT
Tipo de dato en el script	TEXT
Dominio	

**Columna: quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_financial_mov
Descripción	Cantidad en pesos del movimiento financiero.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: quarter\_no**

Detalles de las columnas:

Nombre de la tabla	ss_cat_time
Descripción	Número de cuarto de año en que cae la fecha.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: rcp\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
Descripción	Factor de conversión entre unidades de medida y unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: rcp\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Factor de conversión entre unidades de medida y unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: rcp\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Factor de conversión entre unidades de receta y unidades de inventario.
Llave primaria	No
Referencia a	

Tipo de dato	NUMERIC(12,2)
--------------	---------------

Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: rcp\_conversion\_factor**

Detalles de las columnas:

Nombre de la tabla	op_inv_conversion_factor
Descripción	Factor de conversión entre unidades de las recetas y unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: real\_use**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Uso real del producto
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: real\_use**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Uso real del producto
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: received\_quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception_detail
Descripción	Cantidad de producto recibida.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(7,2)
Tipo de dato en el script	NUMERIC(7,2)
Dominio	

**Columna: received\_quantity**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception_detail
Descripción	Cantidad de producto recibida.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: receptions**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Recepciones del producto acumuladas a la semana.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: receptions**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_mov
Descripción	Recepciones registradas en la semana.
Llave primaria	No
Referencia a	

Tipo de dato	CHARACTER(40)
--------------	---------------

Tipo de dato en el script	CHARACTER(40)
Dominio	

**Columna: receptions**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Recepciones del producto acumuladas a la semana.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: reception\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Número de recepción al que esta relacionado el reporte de diferencias.
Llave primaria	No
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: reception\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Número de recepción asignado por el sistema de manera única.
Llave primaria	Yes
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: reception\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception_detail
Descripción	Llave foránea con la tabla op_grl_step_reception para determinar el número de recepción al que están asociados los productos. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	reception_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: reception\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception_detail
Descripción	Llave foránea con la tabla op_grl_reception para saber a qué número de recepción estan asociados los productos. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	reception_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: reception\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception
Descripción	Número de recepción asignado por el sistema cuando es confirmada. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: reception\_limit**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
Descripción	



Llave primaria	No
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: recep\_cost**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Costo del producto en la recepción.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC
Tipo de dato en el script	NUMERIC
Dominio	

**Columna: recep\_equivalent**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Cantidad recibida del producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	TEXT
Tipo de dato en el script	TEXT
Dominio	

**Columna: recep\_product**

Detalles de las columnas:

Nombre de la tabla	op_grl_difference
Descripción	Código del producto en la recepción.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: recipe\_unit\_measure**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_inventory
Descripción	
Llave primaria	No
Referencia a	unit_id
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: rec\_inv\_end**

Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Cantidad de producto final en unidad de receta
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: rec\_inv\_end**

Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Cantidad de producto final en unidad de receta.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: remission\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception
--------------------	------------------

Descripción	Clave de la remisión de la cual se recibió, en caso de existir.
-------------	---

Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: remission\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Número de remisión de la cual se hace la recepción (en caso de existir esta remisión).
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: remission\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_remission
Descripción	Número que identifica la remisión de manera única.. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: remission\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_remission_detail
Descripción	Llave foránea con la tabla op_grl_remission para determinar a qué remisión de le asocian los productos a recibir. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	remission_id
Tipo de dato	CHARACTER(10)
Tipo de dato en el script	CHARACTER(10)
Dominio	

**Columna: report\_num**

## Detalles de las columnas:

Nombre de la tabla	op_grl_reception
Descripción	Número de reporte que captura el gerente en caso de existir diferencias entre orden y recepción y reportarlo.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(10)
Tipo de dato en el script	CHARACTER VARYING(10)
Dominio	

**Columna: report\_num**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Número de reporte en caso de obtener diferencias entre orden y recepción y de reportarlo al proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(10)
Tipo de dato en el script	CHARACTER VARYING(10)
Dominio	

**Columna: required**

## Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_order
Descripción	Cantidad requerida de producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: required**

## Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_transfer
--------------------	---------------------------

Descripción	Cantidad requerida del producto en la orden en unidades de proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: required\_quantity**

Detalles de las columnas:	
Nombre de la tabla	op_grl_remission_detail
Descripción	Cantidad de producto que viene registrado en la remisión que es lo que supuestamente debe recibir el restaurante.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: responsible**

Detalles de las columnas:	
Nombre de la tabla	op_grl_step_reception
Descripción	Nombre de la persona que hace la captura de la recepción.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: responsible**

Detalles de las columnas:	
Nombre de la tabla	op_grl_reception
Descripción	Nombre de la persona que hace la captura de la recepción.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: sort\_num**

Detalles de las columnas:	
Nombre de la tabla	op_grl_remission_detail
Descripción	Número para el que el ordenamiento de los productos se de de la misma forma en la remisión física que en la electrónica.
Llave primaria	No
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: sort\_num**

Detalles de las columnas:	
Nombre de la tabla	op_grl_step_reception_detail
Descripción	Número para ordenar los elementos de la recepción.
Llave primaria	No
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: sort\_num**

Detalles de las columnas:	
Nombre de la tabla	op_grl_reception_detail
Descripción	Número para tener ordenada la recepción.
Llave primaria	No
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: sort\_seq**

Detalles de las columnas:	
Nombre de la tabla	op_grl_cat_config_difference

Descripción	
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: stock\_code\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Código de zona de stock para el producto.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: stock\_code\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Código de zona de stock para el producto.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: stock\_code\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_providers_product
Descripción	Código de zona de stock para el producto.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(6)
Tipo de dato en el script	CHARACTER(6)
Dominio	

**Columna: store\_desc**

## Detalles de las columnas:

Nombre de la tabla	ss_cat_neighbor_store
Descripción	Nombre del restaurante
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(50)
Tipo de dato en el script	CHARACTER VARYING(50)
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_order
Descripción	Llave foránea con la tabla ss_cat_store para determinar el CC del restaurante y/o su nombre.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	ss_cat_neighbor_store
Descripción	Clave del restaurante
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_order
--------------------	------------------------

Descripción	Llave foránea con la tabla ss_cat_store para determinar el CC y/o nombre del restaurante.
-------------	---

Llave primaria	Yes
Referencia a	store_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception
Descripción	Número de CC del restaurante. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_order_detail
Descripción	Llave foránea con la tabla ss_cat_store para determinar el CC y/o nombre del restaurante el cual tiene asociada esta orden. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	store_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_financial_mov
Descripción	Clave del restaurante.
Llave primaria	No
Referencia a	
Tipo de dato	TIME WITH TIME ZONE
Tipo de dato en el script	TIME WITH TIME ZONE
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_remission_detail
Descripción	Llave foránea con la tabla ss_cat_store para determinar el número del CC y/o nombre del restaurante. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	store_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_order_limit
Descripción	Llave foránea con la tabla ss_cat_store de la plantilla, para determinar el CC y/o nombre del restaurante.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER VARYING(40)
Tipo de dato en el script	CHARACTER VARYING(40)
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_order_detail
Descripción	Número de CC del restaurante asociado a la orden.
Llave primaria	Yes
Referencia a	store_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_reception_detail
--------------------	-------------------------

Descripción	Llave foránea con la tabla ss_cat_store para determinar el número del CC y/o nombre del restaurante. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	store_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_reception
Descripción	Llave foránea con la tabla ss_cat_store para determinar el número de CC y/o nombre del restaurante. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_remission
Descripción	Llave foránea con la tabla ss_cat_store para determinar el número de CC y/o nombre del restaurante. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	store_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_order
Descripción	Número de CC del restaurante para el que se hace la orden.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_way_order
Descripción	Llave foránea con la tabla op_grl_order. Parte de la llave primaria.
Llave primaria	Yes
Referencia a	store_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: store\_id**

Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception_detail
Descripción	Número de CC del restaurante. Parte de la llave primaria de esta tabla.
Llave primaria	Yes
Referencia a	store_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: street**

Detalles de las columnas:

Nombre de la tabla	op_grl_cat_provider
Descripción	Calle de la dirección del proveedor.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(30)
Tipo de dato en el script	CHARACTER VARYING(30)
Dominio	

**Columna: suggested\_quantity**

## Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_transfer
Descripción	Cantidad sugerida del producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: suggested\_quantity**

## Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_order
Descripción	Cantidad sugerida del producto en unidades de inventario.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: transfer\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer
Descripción	Identificador de la transferencia. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: transfer\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_suggested_transfer
Descripción	Identificador de la transferencia. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: transfer\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_transfer
Descripción	Clave de la transferencia. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: transfer\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_transfer_detail
Descripción	Clave de la transferencia. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	transfer_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: transfer\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer_detail
Descripción	Clave de la transferencia. Parte de la llave primaria de la tabla.
Llave primaria	Yes
Referencia a	transfer_id
Tipo de dato	INTEGER
Tipo de dato en el script	INTEGER
Dominio	

**Columna: transfer\_type**

## Detalles de las columnas:

Nombre de la tabla	op_grl_transfer
Descripción	Tipo de transferencia. Con valor de 1 para las transferencias de entrada y 0 para las de salida.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: transfer\_type**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_transfer
Descripción	Tipo de transferencia. Con valor de 1 para las transferencias de entrada y 0 para las de salida.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: unit\_cost**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_reception_detail
Descripción	Costo unitario del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(7,2)
Tipo de dato en el script	NUMERIC(7,2)
Dominio	

**Columna: unit\_cost**

## Detalles de las columnas:

Nombre de la tabla	op_grl_step_order_detail
Descripción	Costo unitario del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: unit\_cost**

## Detalles de las columnas:

Nombre de la tabla	op_grl_order_detail
Descripción	Costo unitario del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: unit\_cost**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Costo unitario del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: unit\_cost**

## Detalles de las columnas:

Nombre de la tabla	op_grl_remission_detail
Descripción	Costo unitario del producto registrado en la remisión.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	



**Columna: unit\_cost**

## Detalles de las columnas:

Nombre de la tabla	op_grl_reception_detail
Descripción	Costo unitario del producto recibido.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: unit\_cost**

## Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Costo unitario del producto.
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: unit\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_unit_measure
Descripción	Identificador único de 4 letras para la unidad de medida.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: unit\_measure\_remis**

## Detalles de las columnas:

Nombre de la tabla	op_grl_remission_detail
Descripción	Llave foránea con la tabla op_grl_cat_uit_measure para determinar la unidad de medida que se tiene en el producto de la remisión.
Llave primaria	No
Referencia a	unit_id
Tipo de dato	CHARACTER(4)
Tipo de dato en el script	CHARACTER(4)
Dominio	

**Columna: unit\_name**

## Detalles de las columnas:

Nombre de la tabla	op_grl_cat_unit_measure
Descripción	Nombre de la unidad de medida.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER VARYING(50)
Tipo de dato en el script	CHARACTER VARYING(50)
Dominio	

**Columna: way\_quantity**

## Detalles de las columnas:

Nombre de la tabla	op_grl_way_order
Descripción	Cantidad del producto que esta en tránsito (En unidad de proveedor).
Llave primaria	No
Referencia a	
Tipo de dato	NUMERIC(12,2)
Tipo de dato en el script	NUMERIC(12,2)
Dominio	

**Columna: weekday\_desc**

## Detalles de las columnas:

Nombre de la tabla	ss_cat_yum_day
Descripción	Nombre del día de la semana.
Llave primaria	No
Referencia a	
Tipo de dato	CHARACTER(15)
Tipo de dato en el script	CHARACTER(15)

Dominio

**Columna: weekday\_id**

## Detalles de las columnas:

Nombre de la tabla	op_grl_order_limit
Descripción	Llave foránea con la tabla ss_cat_yum_day de la plantilla para determinar el día de la semana que es el límite en que PFS debe recibir el pedido.
Llave primaria	Yes
Referencia a	weekday_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: weekday\_id**

## Detalles de las columnas:

Nombre de la tabla	ss_cat_yum_day
Descripción	Clave del día de la semana. Llave primaria de la tabla.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: weekday\_id**

## Detalles de las columnas:

Nombre de la tabla	ss_cat_time
Descripción	Día de la semana. Parte de la llave primaria de la tabla.
Llave primaria	No
Referencia a	weekday_id
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: week\_no**

## Detalles de las columnas:

Nombre de la tabla	ss_cat_time
Descripción	Semana del calendario.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: week\_no**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Semana de cierre de inventario.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: week\_no**

## Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Semana de cierre de inventario.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(40)
Tipo de dato en el script	CHARACTER(40)
Dominio	

**Columna: week\_no**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_begin
Descripción	Semana del inventario inicial para el producto.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT

Dominio

**Columna: year\_no**

## Detalles de las columnas:

Nombre de la tabla	op_inv_step_inventory_detail
Descripción	Año del cierre de inventario.
Llave primaria	Yes
Referencia a	
Tipo de dato	CHARACTER(40)
Tipo de dato en el script	CHARACTER(40)
Dominio	

**Columna: year\_no**

## Detalles de las columnas:

Nombre de la tabla	ss_cat_time
Descripción	Año del calendario.
Llave primaria	No
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	

**Columna: year\_no**

## Detalles de las columnas:

Nombre de la tabla	op_inv_inventory_detail
Descripción	Año del cierre de inventario.
Llave primaria	Yes
Referencia a	
Tipo de dato	SMALLINT
Tipo de dato en el script	SMALLINT
Dominio	







# MANUAL DE USUARIO

Sistema para el Control  
de Ordenes de Compra  
e Inventarios

---



## CONTENIDO.

<b>CONTENIDO</b> .....	<b>239</b>
<b>PEDIDOS</b> .....	<b>240</b>
<b>INTRODUCCIÓN</b> .....	<b>240</b>
<b>RESUMEN DE PROCEDIMIENTOS</b> .....	<b>240</b>
<b>PROCEDIMIENTOS</b> .....	<b>241</b>
Hacer un Pedido.....	241
Consulta de Remisiones Pendientes.....	245
Captura de Recepción .....	246
Consultar Reportes .....	250
<b>PEDIDOS ADELANTADOS</b> .....	<b>252</b>
<b>INVENTARIOS</b> .....	<b>256</b>
<b>INTRODUCCIÓN</b> .....	<b>256</b>
<b>NORMAS</b> .....	<b>257</b>
Recibo.....	257
Almacenaje de Productos .....	258
Normas de Control Diario y Semanal.....	259
Pedido/Orden .....	259
<b>RESUMEN DE PROCEDIMIENTOS</b> .....	<b>259</b>
Congelar Transacciones.....	261
Hacer pedidos .....	263
Recibir .....	264
Almacenar .....	264
Control diario y semanal .....	267
Conteo de inventario crítico semanal .....	31
Control de Inventario Semanal .....	33
Control de Inventario .....	33
Transferencias de Inventarios .....	34
Procedimientos de transferencia de inventario.....	36
Procedimientos en el sistema para una transf. de inventario ..	36
Control de Merma.....	274

# PEDIDOS

## INTRODUCCIÓN

Las siguientes páginas explican como llevar a cabo el ciclo de petición y recepción de productos, el cual incluye: procedimiento para hacer pedidos, revisión de remisiones en caso de contar con ellas, registro de la recepción del pedido dentro del sistema y finalmente consultas de reportes. Todo esto llevado a cabo a través de una aplicación desarrollada para este propósito.

## RESUMEN DE PROCEDIMIENTOS

	Procedimiento	Pasos
1	Hacer un pedido	<ul style="list-style-type: none"><li>• Ingresar al sistema</li><li>• Generar pedido sugerido</li><li>• Agregar productos</li><li>• Modificar valores en el formulario</li><li>• Revisión y confirmación de la orden</li></ul>
2	Consulta de remisiones pendientes	<ul style="list-style-type: none"><li>• Ingresar al sistema</li><li>• Ingresar a la consulta de las remisiones</li><li>• Seleccionar proveedor y remisión a consultar</li></ul>
3	Captura de recepción de producto	<ul style="list-style-type: none"><li>• Ingresar al sistema</li><li>• Determinar el tipo de recepción a realizar</li><li>• Modificar valores en el formulario</li><li>• Revisión y confirmación de la recepción</li><li>• Consulta del reporte de diferencias</li><li>• Captura de número de reporte de proveedor</li></ul>
4	Consultar reportes	<ul style="list-style-type: none"><li>• Ingresar al sistema</li><li>• Seleccionar proveedor</li><li>• Ordenar columnas de manera ascendente o descendente</li><li>• Seleccionar Orden, Remisión ó Recepción a consultar.</li></ul>



## PROCEDIMIENTOS

1	Hacer un Pedido	
	1.1 Ingresar al Sistema	<ul style="list-style-type: none"><li>▪ En la pantalla principal de la computadora donde se encuentran todos los íconos, buscar uno que dice <b>Ereports</b> y pulsar un click sobre él. Si existe otro navegador abierto como el del Reporteador, es necesario cerrarlo y volver a pulsar <b>Ereports</b>. Esto desplegará la pantalla principal de la aplicación.</li><li>▪ La pantalla presenta un par de pestañas, elegir la que dice “<i>Restaurante</i>”.</li><li>▪ Al hacer lo anterior se despliega una pequeña ventana que pide <i>Usuario</i> y <i>Contraseña</i>, los cuales son <i>admin</i> y <i>admin</i>, respectivamente. Pulsar botón de <i>Accesar</i>.</li><li>▪ Elegir en la parte superior la siguiente secuencia de menús: <i>Inventario -&gt;Control de Ordenes de Compra -&gt;Orden de Compra.</i></li></ul> <p><b>NOTA:</b> Si ya se ingresó al sistema para realizar otro de los procedimientos descritos en el resumen, ya no es necesario volver a ingresar, a menos que transcurran más de tres horas de la última vez que se ingreso ó se haya cerrado el navegador.</p>
	1.2 Generar Pedido Sugerido	<ul style="list-style-type: none"><li>▪ Elegir la pestaña que dice <i>Pedido</i>. Se despliega una ventana con un calendario para generar el Pedido Sugerido.</li><li>▪ Elegir la fecha de la cual se desea generar el pedido sugerido, pulsando el número de día en el calendario.</li><li>▪ Pulsar el botón de “<i>Generar</i>”. Con esto se muestra un mensaje que dice “<i>Generando pedido sugerido. Espere por favor</i>”. Hasta que este mensaje se deje de mostrar se procede al paso siguiente.</li></ul>

### 1.3 Agregar Productos

- Pulsar el botón de “*Agregar Producto*”. Esto despliega una ventana con una tabla que muestra todos los productos disponibles para ordenar de los diferentes proveedores disponibles.
- Aquí tenemos diferentes alternativas para la elección de productos:
  - Se puede seleccionar el pedido sugerido, checando “*Seleccionar solo productos de Sugerido*” . Esto marcará en rojo todos los productos del Pedido Sugerido. Si algún producto no se desea, se puede quitar la paloma al cuadrado que esta a la izquierda del producto.
  - Se puede seleccionar todos los productos disponibles checando “*Seleccionar Todos*”. Esto marcará con rojo todos los productos. Si algún producto no se desea, se puede quitar la paloma al cuadrado que esta a la izquierda del producto.
  - Se puede seleccionar productos individualmente únicamente seleccionando el cuadrado que aparece a la izquierda del producto. Esto marcará en rojo el producto seleccionado.
- Finalmente, ya elegidos los productos que se desean ordenar, pulsar el botón “*Aceptar*”. Si no se desea agregar los productos elegidos pulsar botón “*Cancelar*”. En el primer caso, los productos elegidos se desplegarán en la pantalla principal (donde están las pestañas), en ambos casos la ventana de selección de productos se cerrará.

#### **NOTAS:**

- Para la selección de varios productos, la búsqueda por proveedor solo funciona la primera vez que se despliega esta pantalla.
- Si en la elección inicial de productos se detecta que falta algún producto por añadir, se puede hacer posteriormente repitiendo el procedimiento, sin hacer uso de la búsqueda por proveedor, ya que al hacer esto, permitirá agregar solo un producto a la vez, de lo contrario se podrán agregar varios productos.

#### 1.4 Modificar valores en el formulario

- Los campos que se pueden modificar dentro de la tabla son “CAPTURA POR UNIDADES INVENTARIO” y/o “CAPTURA POR PRESENTACION”
- No es necesario modificar ambos, ya que el sistema lo hace automáticamente haciendo uso del factor de equivalencia entre ambas unidades (la de proveedor y la de inventario).
- Al elegir uno de los campos modificables, estos se denotan enmarcados por un cuadro rojo.
- Se puede desplazar a través de los campos modificables del mismo tipo por medio de las “flechas” en el teclado ó la tecla “enter”
- La información mostrada en esta pantalla se divide básicamente en dos segmentos:
  - a) El primero que muestra los datos del sistema (estos valores estarán en ceros en caso de no tener pedido sugerido para algún producto), además de venir una columna de Producto en tránsito en casi de tener producto que se ordenó y aún no se recibe, esto afectará al pedido sugerido original de FMS porque resta el valor del pedido en tránsito de manera que el pedido sugerido se verá modificado.
  - b) El segundo se refiere a las cantidades de producto que se están pidiendo, para hacer comparativas.
- Si el producto se ordena en unidades de inventario, dependiendo del factor de equivalencia, se redondearan las unidades de proveedor para el pedido y se mostrará la equivalencia de este redondeo en unidades de inventario, para estudiar la conveniencia del pedido, ya que no se puede pedir por ejemplo: Media caja de Vasos # 16, esto se redondeará a uno, aunque se requieran 500 vasos, ya que no se puede ordenar media caja.
- La pantalla muestra otros datos informativos como Clave del producto, nombre del producto, costo, proveedor.
- No se permite ordenar productos con cantidad igual a cero, en caso de intentarlo. Y pulsar el botón “Actualizar” se desplegará el mensaje siguiente: “Es necesario capturar la cantidad requerida ó el pedido en unidad de proveedor de todos los productos”. Si esto sucede, pulsar “OK” y el producto que esta en cero se enmarcará en rojo para capturar el valor.
- Si un producto fue añadido por error y no se desea ordenar, solo es necesario quitar la palomita en el cuadro de la izquierda del producto para que no sea tomado en cuenta.
- Si falta añadir algún producto, se pulsa de nuevo el botón de “Agregar producto” y se elige de la lista.
- Ya teniendo los productos que necesitamos, para hacer la orden pulsar el botón “Actualizar”.

## 1.5 Revisión y confirmación del Pedido

- Se despliega una ventana con los productos que fueron ordenados, segmentados también por datos del sistema y por lo que se esta pidiendo, además se subtotaes en caso de hacer la orden a varios proveedores y el total de la orden.
- El título de esta ventana es “*Revisión de Pedido*”, esto nos indica que aún **no es definitivo el pedido** y es sujeto a modificaciones en caso de pulsar el botón “*Cancelar*”. Al hacer esto, la ventana se cerrará y los datos capturados regresarán al formulario de captura, donde se pueden hacer todas las modificaciones que se consideren necesarias en el pedido.
- Los datos de la revisión se pueden imprimir a través del icono en forma de impresora ubicado en la esquina superior derecha de la ventana, al pulsarlo se mostrará el siguiente mensaje “*Desea Imprimir este Documento?*” que se puede aceptar o cancelar y se muestra una vista previa de lo que se va a imprimir. Esta vista previa tiene una línea en color rojo que dice “<< **Continuar con el pedido**”, esta línea se pulsa para regresar a la pantalla donde se puede confirmar el pedido.

**NOTA IMPORTANTE:** Esta ventana no se debe cerrar por ningún motivo a través de la “X” de la parte superior derecha de la ventana, ya que implicaría que se pierden todos los datos capturados en el pedido y habría que repetir todo el proceso de captura de pedido.

- Si después de hecha la revisión todo esta correcto y no se requiere hacer modificaciones, pulsar el botón de “*Aceptar*”.
- Al confirmar la orden se mostrará una vista previa de el Pedido Confirmado con el título “*Confirmación de Pedido*”, con esto el Pedido **ya no puede ser modificado**, además se mostrará el mensaje “*Desea Imprimir este Documento?*”, lo cual se puede aceptar o cancelar.
- Cerrar la ventana a través de la “X” en la parte superior derecha de la ventana, ya que los datos están almacenados correctamente.

## 2 Consulta de Remisiones Pendientes

2.1 Ingresar al sistema

2.2 Ingresar a la consulta de las remisiones

- Repetir el paso 1.1 de ser necesario.

- Elegir en la parte superior la siguiente secuencia de menús:  
*Inventario ->Control de Ordenes de Compra ->Orden de Compra*

Y elegir la pestaña que dice “Remisión”

Ó únicamente elegir la pestaña que dice “Remisión” si ya se encuentra dentro del sistema.

- En la parte superior de la pantalla que se despliega existen un par de listas desplegables, de las cuales hay que elegir en primer lugar el proveedor, de manera que en la segunda lista se desplieguen las remisiones pendientes asociadas al proveedor elegido.
- Después de elegir proveedor y la remisión que se desea consultar, pulsar el botón de “Mostrar”.
- Si este botón se pulsa sin haber elegido proveedor, se desplegará el siguiente mensaje “*Debe seleccionar un proveedor para consultar la remisión*”, y si se pulsa sin elegir la remisión se despliega lo siguiente: “*Debe seleccionar una remisión a consultar*”.
- Si se hizo correctamente lo anterior, se despliega la remisión que proporcionó el proveedor. La particularidad de esta consulta es que en la columna de la descripción del proveedor se indican las diferencias con respecto a la orden con una especie de semáforo en rojo. Las posibles diferencias a desplegarse son:
  - Productos que fueron ordenados, pero que no están en la remisión, productos que no están en el pedido, pero sí aparecen en la remisión, productos que tienen una diferencia de código entre el pedido y la remisión, productos que tienen diferencias en cantidades entre el pedido y la remisión.
- Esta remisión se puede imprimir presionando el icono de la impresora que aparece en la esquina superior derecha de la ventana, al pulsarlo se mostrará el siguiente mensaje “*Desea Imprimir este Documento?*” que se puede aceptar o cancelar y una vista previa de lo que se va a imprimir.
- Cerrar la ventana a través de la “X” en la parte superior derecha de la ventana

### 3 Captura de Recepción

3.1 Ingresar al sistema

3.2 Determinar el tipo de recepción a realizar

- Repetir el paso 1.1 de ser necesario.
- Elegir en la parte superior la siguiente secuencia de menús:  
*Inventario ->Control de Ordenes de Compra -> Orden de Compra*  
*Y elegir la pestaña que dice "Recepción"*  
Ó únicamente elegir la pestaña que dice "Recepción" si ya se encuentra dentro del sistema.
- Se despliega una pantalla que en la parte superior tiene un par de listas desplegables: una para elegir Proveedor y la segunda para Pedidos/Remisiones. Se debe elegir obligatoriamente un proveedor para capturar la recepción, sin importar que tipo de recepción se necesite.
- En el sistema tenemos 3 alternativas para realizar el registro de recepción:
  - a) Hacer la recepción de un Pedido** (Sin remisión). Para este caso se elige el *Proveedor* de la lista y en la lista de *Pedidos/Remisiones* se elige el número de pedido del cual se desea capturar la recepción. Para identificar un pedido en esta lista viene el número de pedido solamente.
  - b) Hacer la recepción de una remisión.** Para este caso se elige el *Proveedor* de la lista y en la lista de *Pedidos/Remisiones* se elige el número de pedido y remisión de la cual se desea capturar la recepción. Para identificar las remisiones, en la lista aparecen asociadas con su número de pedido:  
*Núm. Pedido/Remisión*, ejemplo: 10000/627054
  - c) Si no tenemos una orden ni una remisión.** En caso de realizar una recepción de este tipo se elige el Proveedor de la lista que le corresponda.

**NOTAS:**

- Si no se elige proveedor y se pulsa el botón de "Mostrar" ó el de "Agregar Producto" se muestra el siguiente mensaje: "Debe seleccionar un proveedor para realizar la recepción".
- Si no se elige proveedor y se pulsar el botón de actualizar no se realiza ninguna operación.
- Si ya se eligió proveedor y se desea hacer un tipo de recepción **a) o b)** y no se eligió ni pedido ni remisión y se pulsa el botón "Mostrar" aparece el siguiente mensaje: "Debe seleccionar una orden para realizar la recepción".

### 3.3 Modificar valores en el formulario

- Si el tipo de recepción es **a)**, se despliega los productos que se capturaron en el pedido.
- Si el tipo de recepción es **b)**, se despliegan los productos que se tienen en la remisión en el mismo orden en que se encuentra en la factura que entrega el proveedor con las cantidades que se tengan en esa remisión
- Si el tipo de recepción es **c)** se procede a Agregar productos.
- Para agregar productos sólo es necesario pulsar el botón de “Agregar Producto”, lo cual despliega una ventana de productos como la de el registro de un Pedido, sólo que esta lista sólo contiene los productos del proveedor que se eligió en el paso anterior. Seleccionar los que se deseen añadir a través de poner una paloma al cuadrado que esta al lado del producto y pulsar el botón de aceptar. Esto se puede realizar para los 3 tipos de recepciones mencionadas anteriormente.
- Teniendo todos los productos desplegados en la pantalla, se puede modificar la cantidad recibida a través del campo “CAPTURADA CANTIDAD RECIBIDA” con sólo un click se enmarca en rojo para permitir la captura. El desplazamiento a través de estos campos también puede ser llevado a través de la “flechas” en el teclado ó la tecla “enter”. Si se cambia dicho valor será necesario elegir un valor de la lista de códigos de discrepancia que se encuentra al lado del campo que se modificó, esta lista tiene múltiples valores que nos ayudarán a denotar las diferencias entre lo que se ordeno (o esta en la remisión) y lo que se recibe, en caso de hacer recepción tipo **a)** ó **b)**. Es recomendable que en cuanto se modifique el valor de la cantidad recibida se elija el código de discrepancia ya que el sistema al actualizar mandará el mensaje “Es necesario capturar el código de discrepancia en caso de que la cantidad recibida cambie” por cada uno de los valores modificados.
- Cuando se añaden productos en la pantalla, el código de discrepancia que aparece por defecto es “Adicional por Restaurante ó PFS”, este puede ser modificado si se desea. Por lo tanto para recepciones tipo **c)**, el código de discrepancia ya tendrá este valor porque añadimos todos los productos que se están recibiendo.
- El sistema no permite hacer recepciones con cantidades igual a cero, por lo que si algún producto no se recibe, hay que quitarle la paloma que se encuentra en el cuadrado de la izquierda de cada producto. Si se intenta recibir en cero algún producto el sistema enviará el siguiente mensaje: “Es necesario capturar la cantidad recibida de todos los productos” y el producto que este con ceros se enmarca en rojo.
- Ya teniendo los productos que recibimos, para registrar la recepción pulsar el botón “Actualizar”.

### 3.4 Revisión y Confirmación de la Recepción

- Se despliega una ventana con los productos que fueron recibidos mostrando las mismas columnas que la pantalla de captura del punto anterior.
- El título de esta ventana es “*Revisión de la Recepción*”, esto nos indica que aún **no es definitivo la recepción y no se ha afectado FMS**, por lo tanto es sujeto a modificaciones en caso de pulsar el botón “*Cancelar*”. Al hacer esto, la ventana se cerrará y los datos capturados regresarán al formulario de captura, donde se pueden hacer todas las modificaciones que se consideren necesarias en la recepción.
- Los datos de la revisión se pueden imprimir a través del icono en forma de impresora ubicado en la esquina superior derecha de la ventana, al pulsarlo se mostrará el siguiente mensaje “*Desea Imprimir este Documento?*” que se puede aceptar o cancelar y se muestra una vista previa de lo que se va a imprimir. Esta vista previa tiene una línea en color rojo que dice “**<< Continuar con la recepción**”, esta línea se pulsa para regresar a la pantalla donde se puede confirmar la recepción.
- **NOTA IMPORTANTE:** Esta ventana no se debe cerrar por ningún motivo a través de la “X” de la parte superior derecha de la ventana, ya que implicaría que se pierden todos los datos capturados en la recepción y habría que repetir todo el proceso de registro de la recepción.
- Esta pantalla además muestra un par de campos más los cuales son:
  - El primero es el número de documento que entrega el proveedor, este campo no se capturará si se hace una recepción tipo **b)**, de otra forma se escribe el número que se encuentre en la factura que entregue el proveedor.
  - El segundo campo se refiere a quién captura la recepción y aquí se debe escribir el nombre de quién registra la recepción en el sistema.
- Es necesario capturar ambos campos, de no hacerlo el sistema muestra los siguientes mensajes: “*Es necesario capturar el número de Pedido del Proveedor*” para el primer campo y “*Es necesario capturar el nombre de la persona que captura la recepción*” para el segundo campo.
- Si después de hecha la revisión todo esta correcto y no se requiere hacer modificaciones, pulsar el botón de “*Aceptar*”.
- Al confirmar la recepción se mostrará una vista previa de el Recepción Confirmada con el título “*Confirmación de Recepción*”, con esto la Recepción **ya no puede ser modificada**, además se mostrará el mensaje “*Desea Imprimir este Documento?*”, lo cual se puede aceptar o cancelar.
- Entra a consultar a FMS y ya deben estar registrados los productos recibidos en Inventario de Entradas y Salidas.



### 3.5 Consulta del Reporte de Diferencias

- En esta pantalla de “*Confirmación de Recepción*” se encuentra un vínculo que dice: “*Ver reporte de diferencias*” abajo del texto “*Recepción Terminada*”. Pulsar este vínculo.

**NOTA:** Si este vínculo no aparece es porque la recepción fue tipo **c)** e implica que todo son diferencias.

### 3.6 Captura de Número de Reporte del Proveedor

- Lo anterior despliega una pantalla con todas las diferencias presentadas entre el Pedido y la Recepción, tenemos 4 tipos de diferencias dentro de este reporte:
  - Productos que se ordenaron y no se recibieron.
  - Productos que se recibieron y no se ordenaron ó se ordenaron por otro medio como el teléfono.
  - Productos que tienen diferencias en código del proveedor.
  - Productos que tienen diferencias en cantidades entre Orden y Recepción
- Reportar como se tiene acostumbrado los productos faltantes y/o defectuosos y capturar el número de reporte que proporcione el proveedor en el campo “*No. Reporte del Proveedor*” de la pantalla.
- Los datos de el Reporte de Diferencias se pueden imprimir a través del icono en forma de impresora ubicado en la esquina superior derecha de la ventana, al pulsarlo se mostrará el siguiente mensaje “*Desea Imprimir este Documento?*” que se puede aceptar o cancelar y se muestra una vista previa de lo que se va a imprimir. Esta vista previa tiene una línea en color rojo que dice “**<< Continuar con la captura del reporte del proveedor**”, esta línea se pulsa para regresar a la pantalla donde se puede continuar con la captura del reporte del Proveedor.
- **NOTA IMPORTANTE:** Esta ventana no se debe cerrar por ningún motivo a través de la “X” de la parte superior derecha de la ventana, ya que implicaría que se pierde la oportunidad de registrar el reporte del proveedor en el reporte.
- Cuando ya se capture pulsar el botón de “*Aceptar*”.
- Al hacer lo anterior se mostrará una vista previa del reporte de diferencias con el número de reporte del Proveedor capturado, además se mostrará el mensaje “*Desea Imprimir este Documento?*”, lo cual se puede aceptar o cancelar.

## 4 Consultar Reportes

4.1 Ingresar al sistema

- Repetir el paso 1.1 de ser necesario.

4.2 Seleccionar Proveedor

- Elegir en la parte superior la siguiente secuencia de menús:  
*Inventario ->Control de Ordenes de Compra -> Orden de Compra*

*Y elegir la pestaña que dice "Consultas"*

Ó únicamente elegir la pestaña que dice "Consultas" si ya se encuentra dentro del sistema.

- Se despliega una pantalla que en la parte superior tiene una lista desplegable para elegir proveedor en caso de querer hacer consultas por un solo proveedor. Si no se selecciona ningún proveedor, se despliegan todos los pedidos, remisiones y recepciones relacionadas de las últimas 15 órdenes de 30 días atrás de TODOS los proveedores, si se elige proveedor de la lista se despliega la misma información, es decir las últimas 15 órdenes de 30 días pasados, con remisiones y recepciones asociadas (en caso de existir) pero de el proveedor elegido.

4.3 Ordenar columnas de manera ascendente o descendente

- La pantalla principal de esta sección se divide en una tabla que se divide en 3 secciones:

a) **Pedido.** Esta sección es en la que se basa la pantalla para el ordenamiento en un principio, ya que el número de orden denotada por la columna ID se despliega de manera descendente. Las columnas que pertenecen a esta sección son: "ID" y "Fecha", la primera es el número de la orden y la segunda la fecha en que fue realizada, a ambas columnas se les puede cambiar el orden.

b) **Remisión.** La única columna de esta sección es: "Num pedido", que es el número de remisión, a esta columna se le puede modificar el orden.

c) **Recepción.** Las columnas de esta sección son: "ID", "Num Pedido", "Fecha", "Reporte Dif". Las cuales significan: Número de recepción cargada en el sistema, número de remisión asociada, fecha de la recepción y esta columna simboliza si existe reporte de diferencias, respectivamente. A las columnas que se les puede cambiar el orden son "Num pedido" y "Fecha".

- La forma de cambiar el orden en las columnas anteriormente mencionadas es ponerse en el título de la columna y dar un click en él, de manera que las columnas pasarán de orden ascendente a descendente y viceversa si repetimos el proceso.

4.4 Seleccionar Orden, Remisión ó Recepción a consulta

- Para consultar una orden, hay que elegir el ID ó número de orden que se desee, en ese momento se despliega una ayuda del sistema que dice “*Ver detalle de la orden ID*” donde **ID** es el número de la Orden. Dar un clic sobre el número y se desplegará una ventana con el título de “*Consulta orden de compra*” y tiene los mismos datos de la “*Confirmación de la Orden*” del procedimiento **1.5** del presente manual.
- Para consultar una remisión, hay que elegir el número de pedido ó remisión, se despliega una ayuda del sistema que dice “*Ver detalle de la remisión NUM PEDIDO*” donde **NUM PEDIDO** es el número que se esta eligiendo. Dar un clic sobre el número y se desplegará una ventana con el título de “*Consulta de remisión*” y tiene los mismos datos que el “*Reporte de Remisión*” del procedimiento **2.3** del presente manual.
- Para consultar una recepción, hay que elegir el ID ó número de recepción que se desee, en ese momento se despliega una ayuda del sistema que dice “*Ver detalle de la recepción NUM RECEPCION*” donde **NUM RECEPCION** es el número de la recepción que se esta eligiendo. Dar un clic sobre el número y se desplegará una ventana con el título de “*Consulta de recepción*” y tiene los mismos datos que la “*Confirmación de la Recepción*” del procedimiento **3.4** del presente manual.
- Para consultar un Reporte de Diferencias, hay que elegir el texto que es la intersección entre la columna de “*Reporte Dif*” y el renglón de la orden que se desee consultar. Si en la columna dice “*Sin reporte*” simboliza que si hay diferencias, pero no tienen el número de reporte del proveedor al avisarle de los faltantes, si trae un número, ese número es el del reporte del proveedor.
- Para saber si una orden esta en proceso aún, se puede observar que en la pantalla no tiene una recepción asociada, si ya hay una recepción simboliza que esta finalizado el ciclo.

## PEDIDOS ADELANTADOS

¿Cómo se manejarán los pedidos adelantados dentro de la aplicación?

Cuando quiere confirmar el pedido aparece la siguiente pantalla, que es la de confirmación del pedido:

Codigo Producto	Producto	Proveedor	Datos del Sistema				Esto estas Pidiendo					Dif vs Sug	Costo
			Cantidad Disponible	Pronóstico Requerido	Producto en transito	Sugerido	Cantidad Requerida	Equiv Unidad Prov	Pedido Unidad Prov	Pedido Unidad Inv			
3395	3395 ENSALADA DE COL	GAB COMERCIALIZADORA	11.95	152.51	120.00	21.00	21.00 Kilogram	1.00 Cas	1.00 Cas	21.00 Kilogram	0.00	111.13	
4015	4015LECHUGA PROCES	GAB COMERCIALIZADORA	10.61	22.85	10.00	3.00	3.00 Kilogram	0.33 Cas	1.00 Cas	10.00 Kilogram	6.08	83.87	
4000	4000 BOLLO GRANDE	BIMBO SA DE CV	1.00	11.99	10.00	1.00	1.00 Ba	1.00 Ba	1.00 Ba	1.00 Ba	0.00	11.5	
4001	4001BOLLO BIG CRUNCH	BIMBO SA DE CV	17.00	413.00	339.00	57.00	57.00 Piece	2.85 Ba	3.00 Ba	60.00 Piece	3.00	87.6	
3020	3020/22POLLO	BACHOCO,SA DE CV	94.79	486.56	324.00	68.00	68.00 MX Hea	68.00 MX Hea	68.00 MX Hea	68.00 MX Hea	0.00	1,353.88	
33054	3054TIRAS EMPANI	145 PFS DE MEXICO	611.00	1578.60	747.00	221.00	221.00 Piece	0.92 Bo	1.00 Bo	240.00 Piece	19.00	524.45	
33116	3116PAPA FRANCESA	145 PFS DE MEXICO	68.51	126.80	41.00	18.00	18.00 Kilogram	1.11 Bo	2.00 Bo	33.00 Kilogram	14.40	341.22	
33118	3116PAPA FRANCESA	145 PFS DE MEXICO	68.51	126.80	41.00	18.00	18.00 Kilogram	1.11 Cas	2.00 Cas	33.00 Kilogram	14.40	339	
33230	3230 HARINA T.S.	145 PFS DE MEXICO	12.00	13.03	0.00	2.00	2.00 MX Sac	2.00 Bushie	2.00 Bushie	2.00 MX Sac	0.00	93.6	
33819	3819 PURE DE PAPA	145 PFS DE MEXICO	23.50	39.62	11.00	6.00	6.00 Kilogram	0.39 Bo	1.00 Bo	16.00 Kilogram	9.28	271.33	
33963	3963PREM.BISCU AWO	145 PFS DE MEXICO	12.03	34.04	18.00	5.00	5.00 Ba	0.83 Bo	1.00 Bo	6.00 Ba	1.00	165.29	
34015	4015LECHUGA PROCES	145 PFS DE MEXICO	10.61	22.85	10.00	3.00	3.00 Kilogram	0.33 Bo	1.00 Bo	10.00 Kilogram	6.08	79.87	
34032	4032 BIGCRUNCH	145 PFS DE MEXICO	155.00	234.56	47.00	33.00	33.00 Piece	0.38 Bo	1.00 Bo	88.00 Piece	55.00	463.69	
34036	4036POPCORN CHICKEN	145 PFS DE MEXICO	4.53	20.40	14.00	2.00	2.00 Ba	0.25 Bo	1.00 Bo	8.00 Ba	6.00	697.09	
34632	4632 VASO 16 OZ	145 PFS DE MEXICO	667.00	926.52	130.00	130.00	130.00 Piece	0.13 Bo	1.00 Bo	1000.00 Piece	870.00	304.21	
34730	4730 BOLSA # 6	145 PFS DE MEXICO	302.60	505.28	132.00	71.00	71.00 Piece	0.07 Rol	1.00 Rol	1000.00 Piece	929.00	48.55	
34953	4953 VASO 706 KFC	145 PFS DE MEXICO	168.00	348.91	133.00	48.00	48.00 Piece	0.10 Bo	1.00 Bo	500.00 Piece	452.00	174.42	
34957	4957 TAPA # 406	145 PFS DE MEXICO	662.00	1313.99	468.00	184.00	184.00 Piece	0.18 Bo	1.00 Bo	1000.00 Piece	816.00	105.15	
34960	4960 TAPA 416 KFC	145 PFS DE MEXICO	348.00	391.46	0.00	44.00	44.00 Piece	0.04 Bo	1.00 Bo	1000.00 Piece	956.00	184.02	
38680	8680BUCKET	145 PFS DE MEXICO	74.00	85.37	0.00	12.00	12.00 Piece	0.10 Bo	1.00 Bo	125.00 Piece	113.00	309.87	
38714	8714/91CJ.PICNIC	145 PFS DE MEXICO	75.00	109.03	19.00	16.00	16.00 Piece	0.11 Bo	1.00 Bo	150.00 Piece	134.00	227.53	
83400	ADEREZ.ENSALA	145 PFS DE MEXICO	18.59	20.48	0.00	2.00	2.00 MX Packag	0.25 Bo	1.00 Bo	8.00 MX Packag	6.00	186.76	
84048	JITOMATE	145 PFS DE MEXICO	10.74	34.31	19.00	5.00	5.00 Kilogram	0.50 Bo	1.00 Bo	10.00 Kilogram	5.00	251.02	
Sub-Total: € 1 353.88													

FIGURA 1

Al pulsar el botón de aceptar de la pantalla anterior, se abre una nueva ventana más pequeña en donde se pueden tener los siguientes casos:

1) Cuando el pedido esta siendo realizado antes de la fecha límite de poleo para Pacific. Hagamos la suposiciones necesarias para explicarlo por medio de un ejemplo:

- a) Los límites para el restaurante son lunes y jueves.
- b) El día en que se esta haciendo el pedido es viernes 13 de mayo de 2005, lo cual implica que sería un pedido para el límite del próximo lunes 16 de mayo de 2005.

Dado esto la pantalla que se despliega es la siguiente:



FIGURA 2

Si se pulsa el botón de “Aceptar” de la FIGURA 2, se despliega el siguiente mensaje:



FIGURA 3

Si se pulsa el botón de “OK” de la FIGURA 3, se confirma la orden que debe ser poleada por PFS para el 16/mayo/2005 y con esto finalizamos el proceso imprimiendo el pedido confirmado.

Si se pulsa el botón de “Cancel” se regresa a la pantalla mostrada en la FIGURA 2

2) Cuando el pedido este siendo realizado el mismo día que es la fecha límite de poleo para PFS y son antes de las 12 hrs. Hagamos las suposiciones pertinentes igual que en el caso anterior:

a) Los límites para el restaurante son lunes y jueves.

b) El día en que se esta haciendo el pedido es jueves 12 de mayo de 2005 a las 9:30 am, lo cual implica que sería un pedido para el límite del 12 de mayo de 2005.

Con esta situación la pantalla que se desplegaría sería:



FIGURA 4

Si se pulsa el botón de “Aceptar” de la FIGURA 4, se despliega el siguiente mensaje:

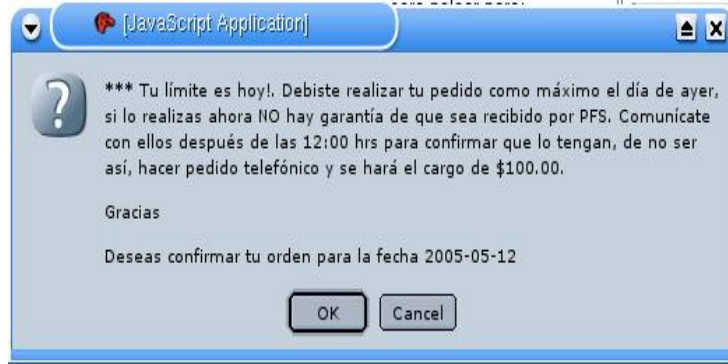


FIGURA 5

Como se puede observar el mensaje trae una advertencia acerca de que la orden debió hacerse como máximo el día anterior, ya que no se tiene la garantía de que PFS lo reciba, ya que el pedido no fue realizado con suficiente anticipación.

Al pulsar el botón “OK”, se confirmará la orden y se finaliza el proceso imprimiendo el pedido confirmado.

Si se pulsa el botón de “Cancel” se regresa a la pantalla mostrada en la FIGURA 4

**3)** Cuando el pedido este siendo realizado el mismo día que es la fecha límite de poleo para PFS y son después de las 12 hrs. Las suposiciones para este caso serían:

- a) Los límites para el restaurante son lunes y jueves.
- b) El día en que se esta haciendo el pedido es jueves 12 de mayo de 2005 a las 15:30 hrs.

En este caso tendríamos las mismas pantallas del caso **1)** (FIGURA 2 Y FIGURA 3), ya que caemos en el caso de realizar el pedido antes de la fecha en la que Pacific espera polear el pedido.

### ¿Cómo adelantar un pedido?

Esto hay veces que se requiere adelantar un pedido porque quizás el gerente quiera preveer el pedido antes de salir de vacaciones, por ejemplo. Para hacer esto lo único que se haría sería pulsar el botón de “Siguiente pedido” en las pantallas de la FIGURA 2 ó FIGURA 4, dependiendo el caso en el que nos encontremos. Al hacer lo anterior se despliega el siguiente mensaje:

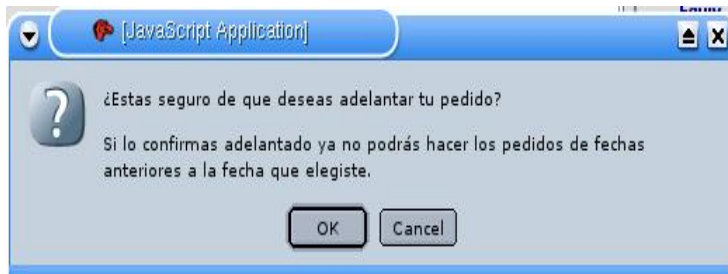


FIGURA 6

Si se pulsa el botón de “Cancel” se regresa a la pantalla de la FIGURA 2 ó FIGURA 4, dependiendo el caso en el que nos encontremos.

Si se pulsa el botón de “OK”, se muestra una pantalla que es muy similar a las de la FIGURA 2 y la FIGURA 4, solo que ahora presenta una lista desplegable con las fechas que tenemos disponibles para hacer el pedido (ya sea adelantado o a tiempo)

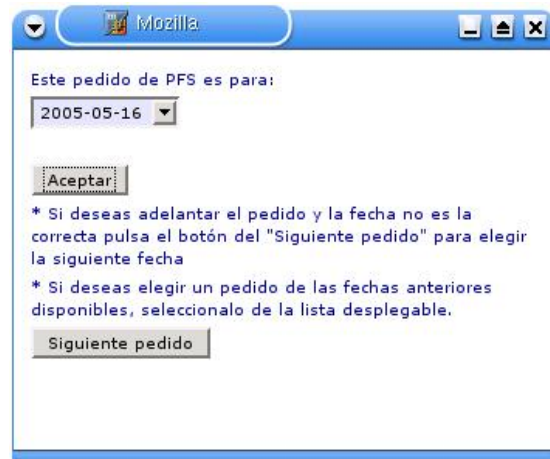


FIGURA 7

Si se pulsa el botón de “*Siguiente Pedido*” hasta el límite de días que se les permite adelantar, la pantalla se mostrará de la siguiente forma:

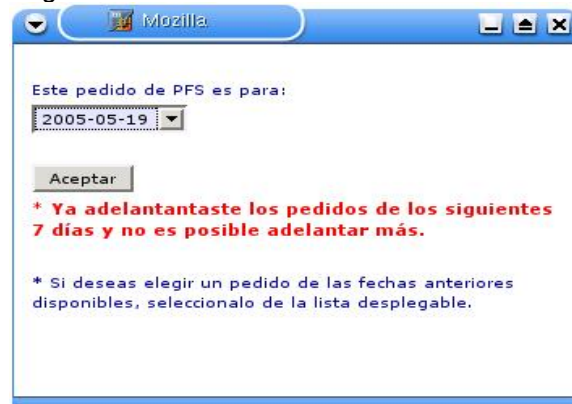


FIGURA 8

Finalmente si ya no se tienen fechas disponibles para adelantar el pedido porque ya hicieron todos los que se tenían probables, la pantalla aparece de la siguiente manera:

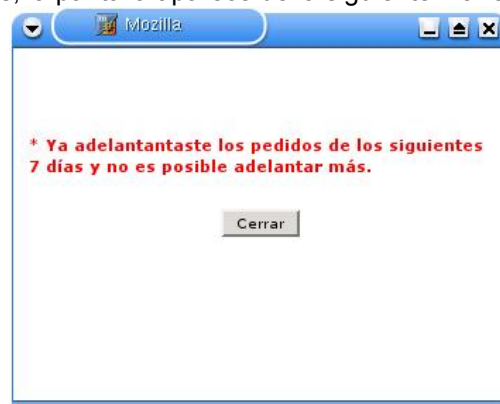


FIGURA 9



## NOTAS:

1) Si al hacer un pedido, detectan que se les olvidó registrar algún producto más, deben hacerlo telefónicamente y no levantar un nuevo pedido en el sistema porque lo tomaría como otro pedido para otro límite de poleo.

2) Se debe tener cuidado al adelantar los pedidos, porque si “saltamos” alguno de ellos, ya no se podrá realizar después. Ejemplo:

Supongamos que puedo hacer pedidos para el 16/mayo y 19/mayo, el pedido debería ser para el 16/mayo por ser la primera fecha disponible, si se llegara a elegir la del 19/mayo por error, el pedido para el límite del 16/mayo **ya no se podrá realizar**.

3) Esta nueva secuencia para adelantar pedidos, sólo se tendrá cuando en el pedido se encuentre por lo menos un producto de PFS.

## INVENTARIOS

### *INTRODUCCIÓN*

El costo del inventario es por lo general el costo más controlable a nivel de restaurante.

El control de inventario incluye:

- Pronosticar transacciones.
- Ordenar Insumos de acuerdo a lo pronosticado.
- Recibir los productos.
- Almacenar los productos.
- Sacar existencias.
- Transferir existencias.
- Documentar mermas.

El control de inventario es una parte muy importante para manejar lucrativamente un restaurante.

Un sistema de rutina para supervisar las ventas de productos y el uso de inventario es esencial para un mejor control del inventario y para controlar los costos.

Es importante para controlar el inventario diaria y semanalmente.



## NORMAS

### Recibo

Los productos e ingredientes se inspeccionan y se almacenan correctamente al ser recibidos.
Se verifican y anotan las temperaturas de los productos congelados y refrigerados al momento de entrega
No se acepta el producto si está fuera del límite de temperatura <ul style="list-style-type: none"><li>- Los productos congelados están a <math>-10^{\circ}\text{F}</math> a <math>+10^{\circ}\text{F}</math> (<math>-12^{\circ}\text{C}</math> a <math>-23^{\circ}\text{C}</math>)</li><li>- Los productos refrigerados están a <math>33^{\circ}\text{F}</math> a <math>40^{\circ}\text{F}</math> (<math>0^{\circ}\text{C}</math> a <math>5^{\circ}\text{C}</math>)</li><li>- Los productos rechazados están claramente marcados, reportados, y devueltos.</li></ul>
Los productos rechazados son claramente rotulados, reportados y devueltos.
Los productos congelados están entre <b><math>-23^{\circ}\text{C}</math> a <math>-12^{\circ}\text{C}</math></b> $-10^{\circ}$ a $10^{\circ}\text{F}$ .
Los productos refrigerados están entre <b><math>1^{\circ}</math> a <math>5^{\circ}\text{C}</math></b> $33^{\circ}$ a $40^{\circ}\text{F}$ .
Los productos refrigerados y congelados son almacenados inmediatamente.

**Cualquier producto entregado fuera del rango de temperatura aceptable debe ser rechazado.**



## Almacenaje de Productos

***Todos los productos se almacenan para asegurar la calidad e higiene de la comida.***

Está cubierta la comida almacenada.

Se rota el inventario para que lo más viejo se use primero.

Se anotan las fechas "Usar antes de..." en toda comida  
- Se anota la fecha de vencimiento o recibo en todos los artículos que no son comida

La comida pasada de fecha o dañada se desecha de inmediato o se sella, se marca y se fecha para evitar que se use.

Todos los materiales se guardan por lo menos 6 pulgadas (15 cm) arriba del piso en tablillas aprobadas.

Los paquetes de los alimentos están en buenas condiciones cuando:

- No hay latas deformes, goteando o infladas.
- Las tapas no están rajadas y cierran bien los recipientes.
- Los artículos apilan de forma segura para prevenir accidentes.

Las fechas de vencimiento o recibo son anotadas en todos los artículos no alimenticios.

Todos los artículos en el restaurante provienen de proveedores aprobados.

El piso, las paredes y los de las áreas de almacenaje están limpios y organizados

**Normas de Control Diario y Semanal**

Norma
Las cuentas del inventario son verificadas por el gerente del restaurante.
El inventario crítico es realizado por lo menos una vez al día.
El inventario semanal es realizado una vez por semana como mínimo, antes de comenzar las ventas para la próxima semanal comercial.

**Pedido/Orden**

Norma
El sistema Pedido Sugerido se usa correctamente.
La gerencia verifica el recibo y la transferencia de los productos.
El pedido es preparado por adelantado.

**Resumen de Procedimientos**

Procedimiento	Pasos a seguir
1 Congelar Transacciones	<ul style="list-style-type: none"> <li>■ Objetivo del congelado</li> <li>■ Recomendaciones para congelar</li> </ul>
2 Hacer pedidos	<ul style="list-style-type: none"> <li>■ Usar el sistema pedido sugerido.</li> <li>■ Efectuar los cálculos.</li> <li>■ Calcular la cantidad de pedido sugerido.</li> <li>■ Calcular la cantidad a pedir.</li> <li>■ Realizar el pedido.</li> </ul>

3	Recibir	<ul style="list-style-type: none"><li>■ Verificar el envase y la calidad apropiada.</li><li>■ Verificar la etiqueta y la fuente.</li><li>■ Verificar las temperaturas.</li><li>■ Verificar la fecha de vencimiento.</li><li>■ Verificar las cantidades y descripciones contra la factura.</li><li>■ Entrar lo recibido en el sistema de inventario.</li></ul>
4	Almacenar	<ul style="list-style-type: none"><li>■ Marcar las fechas de los productos.</li><li>■ Rotar el inventario actual.</li><li>■ Almacenar los productos.</li><li>■ Seguir los procedimientos de almacenaje de los productos químicos.</li><li>■ Descongelar los productos congelados.</li></ul>
5	Control diario y semanal	<p><b>Contar diariamente el inventario crítico</b></p> <p><b>Controlar semanalmente el inventario</b></p> <p><b>Control del inventario</b></p> <p><b>Transferencias de inventario</b></p> <p><b>Procedimientos de transferencia de inventario</b></p>
6	Control de mermas	<p><b>Factores de la merma</b></p> <p><b>Análisis del uso</b></p> <p><b>Identificar si es merma o un problema del mantenimiento de registro</b></p> <p><b>Corregir los problemas COS debido a la merma</b></p> <p><b>Corregir los problemas de mantenimiento del registro</b></p> <p><b>Tabla para resolver problemas COS</b></p>

1	<b>Congelar Transacciones</b>
<p>Suministros y medios:</p> <p>Calculadora Hoja con transacciones congeladas. Tablilla con sujetapapeles y pluma</p>	
<p>1 Objetivo</p>	<p>El objetivo de la aplicación de pronósticos es dar al gerente la posibilidad de pronosticar la cantidad de pedidos que el restaurante venderá diariamente a la hora de la comida y de la cena.</p> <p>El sistema genera un pronóstico basado en información actual, y permite al gerente ajustarlo si fuera necesario.</p> <p>El sistema utiliza el total de pedidos para generar:</p> <p>Planes de preparación</p> <p>Cantidad sugerida para ordenar producto a proveedores como Pacific Star</p> <p>Por otro lado, usted utiliza el Congelado de transacciones para la programación de horarios gráficos diarios de acuerdo al modelo de labor.</p> <p><b><i>Su objetivo es pronosticar, de la manera mas precisa posible, la cantidad de pedidos que venderá el restaurante.</i></b></p> <p>EL PRONOSTICAR ES UNA DE LAS ACTIVIDADES MAS IMPORTANTES DEL SISTEMA SAI. ENTRE MEJOR APRENDA A PRONOSTICAR, MAS EFICIENTE SERA EL FUNCIONAMIENTO DE SU RESTAURANTE.</p> <p><b><i>Se considera un buen pronóstico cuando la diferencia entre lo real y lo pronosticado tiene 5 % de variación máximo.</i></b></p>

2 Recomendaciones para congelar

Revise el pronóstico del sistema.

Imprima el reporte de pedidos reales vs pronosticados por día de la semana y por fecha.

Identifique las tendencias actuales y la de años anteriores

Revise los planes de mercadotecnia nacional y local para las siguientes semanas. (pueden generar aumento en transacciones o cambios de mixes de productos unicamente)

Calcule e introduzca el pronóstico (mínimo 17 días).

Oprimir F9 para comenzar a modificar transacciones

Moverse al campo de comida o cena de la fecha a modificar

Capturar las transacciones deseadas para el detalle de comida o cena, o bien modificar el total de transacciones.

El sistema se encargará de distribuir el incremento en ambas partes del día (comida o cena). Se pueden modificar las transacciones a más detalle por cada semana, por destino y por partes del día, es decir, puedes indicarle al sistema cuántas transacciones deseadas para comedor, llevar o Entrega a Domicilio, para la comida y cuántas para la cena.

Para realizar esto, en la pantalla donde te aparecen las 2 semanas, oprimir F10 para que muestre el sistema el detalle de la semana 1.

Ya en esta pantalla, te mostrará las transacciones para cada destino y para cada parte del día (daypart), bastará con moverte al campo a modificar e indicar las nuevas transacciones.

Con F10 podrás ahora irte al detalle de la segunda semana y realizar el mismo procedimiento. Con F10 regresarás al resumen de las 2 semanas.

Una vez que has terminado de realizar tus modificaciones, oprimir F12 para salir

Seleccionar Salir Actualizando

El sistema te solicitará la fecha hasta la cual congelará las transacciones.

Ingresar la fecha.El sistema trabajará generando los planes de preparación para los días congelados menos 1.

<b>2</b>	<b>Hacer pedidos</b>
<p>Suministros y medios:</p>	<p>Hoja de trabajo Pedido Sugerido</p> <p>Calculadora</p> <p>Reportes de inventario y hojas de trabajo</p> <p>Tablilla con sujetapapeles y pluma</p>
<p>1 Usar el sistema Pedido Sugerido</p>	<p>Consultar el congelado de transacciones y verificar que se hayan congelado 17 días por adelantado.</p> <ul style="list-style-type: none"> <li>■ El sistema “pedido sugerido” se usa para determinar los niveles adecuados de inventario que se deben mantener de cada producto para evitar que falte o que se eche a perder. La cantidad de pedido sugerido está basada en el uso del producto y en la mezcla de venta. La cantidad es cambiada sólo si la mezcla del producto o las ventas cambia significativamente.</li> </ul>
<p>2 Efectuar los cálculos</p>	<ul style="list-style-type: none"> <li>■ <b>Usar el Sistema Pedido sugerido.</b></li> <li>■ Calcular (por producto del inventario):</li> <li>■ La cantidad del producto a usar para los próximos 14 días</li> <li>■ Porcentaje esperado del aumento o reducción de las ventas basado en los programas locales de mercadeo, en las promociones regionales, eventos especiales, etc. (Consultar la tabla de Congelado para tener más información ya que ahí viene incluido.)</li> <li>■ Nivel de inventario actual del producto.</li> </ul>

**La cantidad Pedido Sugerido no tiene que determinarse todas las semanas; cambiará sólo si las mezclas del producto o las ventas cambian significativamente. Diferentes productos pueden tener diferentes reservas, tales como perecederos y productos que pueden tener una reserva para 1 día. El objetivo es tener por lo menos 1 día el mayor valor del inventario si al día siguiente se entregará el pedido.**

3	Recibir				
		Suministros y medios:	Tablilla con sujetapapeles y pluma Hojas y formularios de entrega		
1	Verificar el envase y la calidad apropiados	<ul style="list-style-type: none"> <li>■ Verificar que las cajas, recipientes o envases estén en buenas condiciones.</li> </ul> <table border="0" style="width: 100%;"> <tr> <td style="width: 50%; vertical-align: top;"> <b>Artículo:</b>                              Cajas de cartón/bolsas                              Latas                              Productos lácteos                              Frutas frescas/vegetales                              No alimenticios                         </td> <td style="width: 50%; vertical-align: top;"> <b>Verificar si está:</b>                              Dañada, gotea, rasgados, abierto                              Huecos, abollada, hendida                              Fecha de caducidad, lote.                              Color, aroma, frescura, tamaño.                              Envase y condición                         </td> </tr> </table> <ul style="list-style-type: none"> <li>■ Prestar atención especial a los envases dañados/goteantes porque el producto dentro también puede estar dañado o contaminado.</li> </ul> <p><b>RECHAZAR todos los artículos que no son aceptables.</b></p>		<b>Artículo:</b> Cajas de cartón/bolsas Latas Productos lácteos Frutas frescas/vegetales No alimenticios	<b>Verificar si está:</b> Dañada, gotea, rasgados, abierto Huecos, abollada, hendida Fecha de caducidad, lote. Color, aroma, frescura, tamaño. Envase y condición
<b>Artículo:</b> Cajas de cartón/bolsas Latas Productos lácteos Frutas frescas/vegetales No alimenticios	<b>Verificar si está:</b> Dañada, gotea, rasgados, abierto Huecos, abollada, hendida Fecha de caducidad, lote. Color, aroma, frescura, tamaño. Envase y condición				
2	Verificar la etiqueta y la fuente	<ul style="list-style-type: none"> <li>■ Verificar la etiqueta y que todos los productos sean de los proveedores aprobados.</li> </ul> <p>Requisitos mínimos de la etiqueta:</p> <ul style="list-style-type: none"> <li>■ Nombre del artículo</li> <li>■ Fecha de caducidad.</li> <li>■ Número de pedido del artículo</li> <li>■ Nombre del fabricante/número de identificación y localidad</li> <li>■ Peso neto</li> <li>■ Artículos por envase</li> <li>■ Cualquiera otra marca/condiciones exigidas por la ley</li> </ul> <p><b>RECHAZAR todos los artículos que no reúnan esas exigencias.</b></p>			

4	Almacenar		
		Suministros y medios:	Plumón o Pluma Registro de merma



1 Marcar las fechas de los artículos

- Marcar las fechas en todos los artículos alimenticios recibidos con una pluma (no tóxica) y etiquetas de HLD incluyendo la siguiente información:

Producto_____	Nombre del producto
Llegada_____	Fecha de recepción
Salida_____	Fecha en que sale a descongelar
Listo_____	Fecha en que se puede usar
Caduca_____	Fecha en que se tiene que descartar.

2 Rotar el inventario actual

Rotar y marcar el inventario de modo que el producto más viejo se use primero.

- Para reducir la confusión, los productos pueden ser utilizado el metodo PEPS. Primeras Entredas Primeras Salidas.

Normalmente, los artículos acabados de entregar deben tener una vigencia más larga que los bienes en existencia en el restaurante. Sin embargo, las fechas para usar deben verificarse contra las fechas para usar del inventario existente con el objetivo de almacenar los artículos apropiadamente.

**Desechar los productos no vigentes y los artículos que se vean o huelan mal. Actualizar el registro de merma con la cantidad y la descripción del producto desechado.**

**Es posible que un artículo vigente pueda dañarse debido a manejo o almacenaje impropios.**



**5 Control diario y  
semanal**

Suministros y  
medios:

Tablilla con sujetapapeles y pluma

Hojas de control de inventario

Bascula calibrada para kilogramos

Formularios de transferencia de inventario

1 Conteo de inventario crítico diario

- Los artículos claves del inventario deben contarse todos los días después que el restaurante haya cerrado por la noche, y si es necesario una vez por la mañana antes de que el restaurante abra, para verificar la cuenta de la noche.

**Cualquier diferencia entre los conteos del inventario del cierre y la apertura deben anotarse e investigarse.**

- Los artículos incluidos en el conteo diario incluyen los más costosos, los más difíciles de controlar, los más fáciles de robar, tales como:
  - Carnes.
  - Quesos.
  - Vegetales (Champiñones y pimiento verde)
  - Refrescos
  - Harina.
  - Artículos promocionales.( Premios, Juguetes, etc..)
  - Cajas de pizza (especialmente en Restaurantes de Entrega).
  - Artículos catalogados de tener una variante inaceptable.
  - **Los conteos o pesos reales deben hacerse utilizando una balanza calibrada en kilogramos**
  - **Se debe restar el peso del recipiente cuando se hace un inventario físico.**
  - **El peso de los recipientes metalicos de la mesa de producción son:**

Entero -	<b>1. 300 Kg</b>
$\frac{1}{2}$ -	<b>800 grs</b>
$\frac{1}{4}$ -	<b>550 grs</b>
$\frac{1}{8}$ -	<b>400 grs</b>
  - Asegurar que los recibos y transferencias han sido debidamente contabilizados en las hojas de control de inventario crítico.
  - Las grandes variantes inexplicables pueden atribuirse a problemas de entrenamiento, robo, o porque el exceso de un producto en inventario conduce a que se eche a perder, etc.

Si es necesario debido a grandes variantes del producto, el inventario crítico debe hacerse varias veces al día incluyendo en la tarde después del período de más venta, después del cambio de turno, y después del período de más venta de la noche.

## 2 Control de Inventario Semanal

- El inventario semanal es una extensión del inventario crítico diario. Es realizado todos los días lunes por la noche después que el restaurante ha cerrado. Incluye todos los productos utilizados en el restaurante que son controlados semanalmente.
- Como en los inventarios críticos diarios, es esencial archivar todos los recibos del inventario, transferencias de entrada y salida, y la cuenta del inventario físico final.
- El uso real de todos los productos es comparado con el uso ideal de todos los productos, y las variantes son calculadas en términos de cantidad y costo.
- Es esencial que el conteo y los costos de los productos no elaborados sean exactos para obtener una información para el reporte financiero (P&L), así como para proporcionar información para los pedidos.

## Control de Inventario

- El Gerente de Area realiza comprobaciones de rutina al azar en los inventarios diarios. El error permisible en los conteos está en el rango de  $\pm 1\%$  de las cifras reportadas.
- El Gerente de Area debe realizar por lo menos 1 inventario completo en un período y realizar por lo menos 1 conteo de inventario en cada restaurante una vez al año.
- Todos los artículos del inventario son para utilizar sólo con fines comerciales.
- Los departamentos de contabilidad y auditoría interna pueden realizar inventarios de rutina al final del período para verificar la exactitud del informe y completar Auditorias.

### 3 Transferencias de Inventarios

- **Transferencias de inventario entre restaurantes:**
- Debe mantenerse a un mínimo y hacerse sólo en situaciones de emergencia.
- No debe considerarse como una alternativa para hacer proyecciones propias al hacer los pedidos.
- Sólo se debe hacer dentro de la misma compañía (es decir, ninguna compañía a franquicia o vice versa) a menos que esté autorizado previamente por el Gerente de Area.

Todas las mercancías refrigeradas o congeladas deben ser transportadas en una unidad de transporte refrigerada o congelada, en condiciones apropiadas para la seguridad de los alimentos y de higiene. **Donde esto no se pueda hacer, no se podrán transferir las mercancías.**

- Todas las transferencias, independientemente del valor monetario, deben hacerse usando la documentación de transferencia de inventario.

**No se permiten préstamos de mercancías (por eje. artículos prestados entre restaurantes sin la apropiada documentación).**

Transferencias  
de inventario  
(Continuación)

- **Los formularios de transferencia de inventario deben contener la siguiente información:**
- Fecha de la transferencia.
- Descripción del producto.
- Cantidad de unidades transferidas.
- Costo por unidad.
- Costo total.
- Número de identificación de 3 dígitos del restaurante que envía. (Centro de Contribución C.C.)
- Número de identificación de 3 dígitos del restaurante que recibe. (Centro de Contribución C.C.)
- Números de cuenta.
- Cantidad total transferida.
- Firmas del integrante apropiado del equipo tanto del restaurante que envía y del que recibe.
- La mercancía que sale del restaurante debe tener 2 firmas por procedimiento.

4 Procedimientos de transferencia de inventario

- Deben seguirse todos los parámetros para la seguridad y la higiene de los alimentos.
- Todas las transferencias entradas y salida deben ser registradas en el sistema de entradas y salidas de FMS.
- La mercancía que sale requiere de dos firmas autorizadas. Los formularios de transferencia deben ser firmados por el integrante apropiado del Equipo en el restaurante que recibe.
- Todas las transferencias deben entrarse en los sistemas de administración de inventario del que restaurante que envía y del que recibe el mismo día en que ocurra la transferencia.
- Debe enviarse el número de copias apropiado al restaurante que recibe (es decir, 1 para poner en el archivo del restaurante que recibe). El número apropiado también se debe guardar en el restaurante que envía (es decir, 1 para guardar en el archivo del restaurante que envía). Consultar los detalles en el apéndice.
- Las transferencias de mercancía son verificadas por la area y/o departamento de Finanzas para confirmar que las transferencias salidas concuerdan con las transferencias entradas. Cualquier discrepancia puede motivar una medida disciplinaria, hasta e incluyendo el despido del empleo.



4 Procedimientos en el sistema para una transferencia de inventario

Instrucciones para realizar transferencias:

- Seleccionar Inventario, Entradas y Salidas
- Seleccionar el ingrediente deseado
- Moverse hasta la columna del día en el que se está transfiriendo o bien el día en que se está recibiendo en caso de ser una nota de entrada. Si estás capturando el inventario final de la semana muévete hasta la columna de inventario final.
- Oprimir F6 Transfer, para el caso de una nota de entrada e inventario final, continuar con el paso siguiente.
- Indicar la cantidad a transferir
- Indicar el centro de costos que lo recibirá, el centro de costos deberá introducirse en 6 dígitos, por ejemplo 000804
- Oprimir F6 para continuar
- Confirmar con S
- Oprimir F12 para salir
- Seleccionar 2 para salir actualizando. La opción 1 Hacer cambios adicionales te regresará a la pantalla donde estabas trabajando, la opción 3 Salir sin actualizar anulará lo que hayas registrado.
- Acontinuación ejemplo de transferencia.

## 6 Control de Merma

<p>Producto</p> <p>Suministros y medios:</p>	<p>Higienizar</p> <p>Informes sobre el uso</p> <p>Hojas de cuenta manual</p> <p>Bascula calibrada, higienizada, si es aplicable</p> <p>Termómetros, si se necesitan</p> <p>Libro para registrar las temperaturas</p>
<p>1 Factores de merma</p>	<ul style="list-style-type: none"> <li>■ Si un restaurante quiere ser rentable, el Gerente General del Restaurante debe controlar la merma.</li> <li>■ Alguna merma es inevitable si un restaurante quiere tener una alta calidad de producto en existencia y estar listo para atender rápidamente los pedidos de los clientes.</li> <li>■ Estos pedidos generan las ventas que crean las ganancias del restaurante.</li> <li>■ Si un restaurante tiene demasiados alimentos en existencia o comete errores en la preparación, el exceso de merma reduce la ganancia.</li> <li>■ El exceso de merma puede conducir a que un restaurante opere con pérdida.</li> </ul>
<p>2 Análisis del uso</p>	<ul style="list-style-type: none"> <li>■ La merma puede incrementarse por malas prácticas al recibir, almacenar y preparar los productos.</li> <li>■ Es responsabilidad del Gerente General del Restaurante eliminar esas malas prácticas y evitar problemas tales como:             <ul style="list-style-type: none"> <li>■ No rotar el producto en el almacén.</li> <li>■ No documentar los ajustes en la proyección del producto.</li> </ul> </li> </ul>



# Apéndice C.

## Estándares de programación

<b>BASE DE DATOS</b>	
BASE DE DATOS GENERAL	dbeyum
TABLAS GENERALES	ss+_+ cat +_+ nombre
TABLAS DE APLICACIONES	id_direcciones + _ + id_aplicaciones + _ + nombre
NOMBRE DE LA LLAVE PRIMARIA	nombre+_+id
NOMBRE DEL CAMPO	nombre
PROCEDIMIENTOS ALMACENADOS	id_direcciones + _ + id_aplicaciones + _ + sp + _+ nombre
DISPARADORES	id_direcciones + _ + id_aplicaciones + _ + tr + _+ nombre
ÍNDICES	idx_id_direcciones + _ + nombre de la tabla + _ + 99
ÍNDICES ÚNICOS	udx_id_direcciones + _ + nombre de la tabla + _ + 99
VISTAS	id_direcciones + _ + id_aplicaciones + _ + vw + _+ nombre

## EJEMPLOS

## TABLAS GENERALES

IDENTIFICADOR	DESCRIPCIÓN
ss_cat_store	Almacén
ss_cat_area	Área
ss_cat_zone	Zona
ss_cat_region	Región
ss_cat_company	Compañía
ss_cat_employee	Empleado
ss_cat_period	Periodo

## TABLAS DE APLICACIONES

IDENTIFICADOR	DESCRIPCIÓN
op_grl_sale	Tabla de venta
hr crt_detail	Tabla de detalle de la orden
hr crt_cat_course	Tabla de catalogo de curso
rd_chm_survey	Tabla de champs survey
fi_grl_forecast	Tabla de pronostico

## NOMBRE DE LA LLAVE PRIMARIA

<b>IDENTIFICADOR</b>	<b>DESCRIPCIÓN</b>
area_id	Identificador de área
region_id	Identificador de región
employee_id	Identificador de empleado
inventory_id	Identificador de inventario

<b>NOMBRE DEL CAMPO</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCIÓN</b>
area_desc	Descripción del área
region_desc	Descripción de la región
company_desc	Descripción de la compañía
Amount	Descripción de la equivalencia
amt_sale	Descripción equivalente de la venta
Adress	Dirección
Street	Calle
Quantity	Cantidad
qty_items_sold	Cantidad vendida
amt_items_sold	Equivalencia vendida
d_date	Día de venta
d_date	Día de menú
d_date	Día de inventario
d_date	Día de orden

<b>VISTAS</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCIÓN</b>
op_grl_vw_sale	Vista venta
hr crt_vw_detail	Vista detalle de la orden
hr crt_vw_course	Vista curso
rd_chm_vw_survey	Vista champs survey
fi_grl_vw_forecast	Vista pronostico

<b>PROCEDIMIENTOS ALMACENADOS</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCIÓN</b>
op_grl_sp_sale	Procedimiento almacenado de venta
hr crt_sp_detail	Procedimiento almacenado del detalle de la orden
hr crt_sp_course	Procedimiento almacenado del curso
rd_chm_sp_survey	Procedimiento almacenado del champs survey
fi_grl_sp_forecast	Procedimiento almacenado del pronostico

<b>DISPARADORES</b>
---------------------

IDENTIFICADOR	DESCRIPCIÓN
op_grl_tr_sale	Disparador de venta
op_crt_tr_detail	Disparador de detalle de la orden
hr_crt_tr_course	Disparador de curso
rd_chm_tr_survey	Disparador de champs survey
fi_grl_tr_forecast	Disparador de pronostico

ÍNDICES	
IDENTIFICADOR	DESCRIPCIÓN
idx_op_grl_sale_01	Índice de tabla de venta (almacén)
idx_op_grl_sale_02	Índice de tabla de venta (día)
idx_hr_crt_detail_01	Índice de tabla de detalle de la orden
idx_hr_crt_cat_course_01	Índice de tabla de catálogo curso
idx_rd_chm_survey_01	Índice de tabla de champs survey
idx_fi_grl_forecast_01	Índice de tabla de pronostico

ÍNDICE UNICO	
IDENTIFICADOR	DESCRIPCION
udx_op_grl_sale_01	Índice único de tabla de venta (almacén)
udx_op_grl_sale_02	Índice único de tabla de venta (día)
udx_hr_crt_detail_01	Índice único de tabla de detalle de la orden
udx_hr_crt_cat_course_01	Índice único de tabla de catálogo e curso
udx_rd_chm_survey_01	Índice único de tabla de champs survey
udx_fi_grl_forecast_01	Índice único de tabla de pronostico

## CONVENCIONES : NOMBRE DE VARIABLE

### CATÁLOGOS

ALCANCE	
IDENTIFICADOR	DESCRIPCIÓN
G	Global
L	Local
I	Instancia
M	Modulo
P	Parámetro
S	Sesión

TIPO
------

IDENTIFICADOR	DESCRIPCIÓN
A	Array
S	String
C	character
V	varchar
F	Decimal/Double/float
L	Long
I	Integer
F	flotante
B	boolean
D	Date

### VARIABLES

<b>VARIABLES GLOBALES</b>	<i>alcance+tipo+nombre</i>
<b>CONSTANTES</b>	<b>NOMBRE</b>

### EJEMPLOS

VARIABLES GLOBALES	
IDENTIFICADOR	NOMBRE
giPeriod	Periodo CR
giYear	Año CR
giStore	Almacen
gsUser	Usuario
gsPass	Contraseña
giYearMonth	Año CR
msFirstNombre	Primer nombre

CONSTANTES	
IDENTIFICADOR	NOMBRE
HEAD_COUNT	Mínimo tamaño
MAXIMUM_SIZE	Máximo tamaño

**CONVENCIONES : NOMBRE DE PROGRAMAS****CATÁLOGOS**

<b>DIRECTORIO DE NOMBRES</b>	
Entry	Entrada
Rpt	Reporte
Quero	Consulta
Proc	Proceso
Include	Inclusion

<b>PROGRAMAS</b>	
<b>PROGRAMAS</b>	<i>Nombre.Programa Extensión</i>
<b>FUNCIONES</b>	<i>nombre1Nombre2</i>
<b>PAQUETES</b>	<i>nombre1nombre2</i>
<b>CLASES</b>	<i>Nombre1Nombre2</i>
<b>INTERFASES</b>	<i>Nombre1Nombre2</i>
<b>MÉTODOS</b>	<i>nombre1Nombre2</i>

**EJEMPLOS**

<b>PROGRAMA NOMBRE</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCION</b>
StoreCr.php	Mantenimiento a la estructura Organizacional ( Centro de Costo)
KpiKfc.php	Reporte de KPI
HomeServicePh.php	Reporte de Home Service
SaleCr.jsp	Venta

<b>FUNCIONES</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCIÓN</b>
loadFileChamps	Carga de la función Champs
updateEmployee	Función alta de empleado
changePrices	Función cambio de precios

<b>PAQUETES</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCION</b>
Loadfilechamps	Carga del paquete Champs
Updateemployee	Paquete alta de empleado
Changeprices	Paquete cambio de precios



<b>CLASES</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCION</b>
LoadFileChamps	Carga de la Clase Champs
UpdateEmployee	Clase alta de empleado
ChangePrices	Clase cambio de precios

<b>INTERFASES</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCION</b>
LoadFileChamps	Carga de la interfase Champs
UpdateEmployee	Interfase alta de empleado
ChangePrices	Interfase cambio de precios

<b>MÉTODOS</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCIÓN</b>
loadFileChamps	Carga del método Champs
updateEmployee	Método alta de empleado
changePrices	Método cambio de precios

## **CONVENCIONES : CONTROLES HTML**

### **CATÁLOGOS**

<b>CONTROLES</b>	
<b>IDENTIFICADOR</b>	<b>DESCRIPCIÓN</b>
Txt	Cuadro de texto
Lst	Lista desplegable
Tbl	Tabla
Frm	Forma
Fra	Frase
Div	División
Ifr	Frase interactiva
Cmd	Botón
Txt	Area de texto
Img	Imágenes
Opt	Radio Botón
Chk	Botón de chequeo
Hid	Campo oculto

**EJEMPLOS** <http://mexwsv02/template/StandardObjectCR.htm>

<b>Cuadro de texto</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
txtPeriod	Cuadro de texto para Periodo
txtYear	Cuadro de texto para Año

<b>Lista desplegable</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
lstPeriod	Cuadro de listado para Periodo
txtYear	Cuadro de listado para Año

<b>Tabla</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
tblPeriod	Tabla para Periodo
tblYear	Tabla para Año

<b>Forma</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
frmPeriod	Forma para Periodo
frmYear	Forma para Año

<b>Marco</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
fraPeriod	Marco para Periodo
fraYear	Marco para Año

<b>División</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
divPeriod	División para Periodo
divYear	División para Año

<b>Marco Interactiva</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
ifrPeriod	Marco interactivo para Periodo
ifrYear	Marco interactivo para Año

<b>Botón</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
cmdPeriod	Botón para Periodo
cmdYear	Botón para Año

<b>Área de texto</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
txtPeriod	Área de texto para Periodo
txtYear	Área de texto para Año

<b>Imágenes</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
imgPeriod	Botón para Periodo
imgYear	Botón para Año

<b>Radio Botón</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
optPeriod	Radio Botón para Periodo
optYear	Radio Botón para Año

<b>Botón de Chequeo</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
chkPeriod	Botón de Chequeo para Periodo
chkYear	Botón de Chequeo para Año

<b>Campo Oculto</b>	
<b>IDENTIFICADOR</b>	<b>NOMBRE</b>
hidPeriod	Campo oculto para Periodo
hidYear	Campo oculto para Año



- Procesamiento de Bases de Datos,  
Fundamentos, diseño e instrumentación  
Quinta edición  
David M. Kroenke  
Ed. Prentice Hall  
México 1996
- Sistemas de Bases de Datos,  
Administración y uso  
Alice Y. H. Tsai  
Editorial Prentice Hall  
México 1990
- DB2/SQL,  
Manual para programadores  
Tim Martyn  
Tim Hartley  
Editorial Mc.Graw Hill  
España 1991
- Fundamentos de Bases de Datos  
Segunda edición  
Henry F. Korth  
Abraham Silberschatz  
Editorial Mc.Graw Hill
- Revista PC WORLD. Los Secretos del nuevo [Office](#). Edición julio 2001  
[Panamá](#), año 8 número 101.
- Revista PC WORLD. Proteja su PC, como evitar ataque de virus. Edición  
julio 2002. Panamá, año 10, número 112.
- Linux (Manual de Referencia)  
Segunda edición  
Osborne McGraw-Hill  
España, 2000
- Practical PostgreSQL (O`Reilly)  
John Worsley, Joshua Drake (Comand Prompt, Inc.)  
Editado por Andrew Brookins, Michael Holloway  
Copyright © 2001 Commandprompt, Inc.
- Java Programming Language SSL-257-V052  
Student Guide  
Sun Educational Services  
USA, 2002

## Referencias.

<http://es.wikipedia.org/wiki/>

<http://cache.fdo-may.ubiobio.cl/decom/doc/VIRUS2.htm>

[http://www.geocities.com/diana\\_m\\_alvarez/principal.htm](http://www.geocities.com/diana_m_alvarez/principal.htm)

<http://www.monografias.com/trabajos/auditoinfo/auditoinfo.shtml>

<http://dmi.uib.es/~bbuades/auditoria/auditoria.PPT>

<http://www.delitosinformaticos.com/propiedadindustrial/auditoria.shtml>

<http://www.monografias.com/trabajos5/audi/audi.shtml>

<http://www.monografias.com/trabajos11/breverres/breverres.shtml>

<http://www.monografias.com/trabajos/sequinfo/sequinfo.shtml>

<http://www.geocities.com/Athens/Olympus/7428/virus1.html>

<http://ciberconta.unizar.es/LECCION/SEGURO/101.HTM>

[http://www.criptored.upm.es/guiateoria/qt\\_m142a.htm](http://www.criptored.upm.es/guiateoria/qt_m142a.htm)

[http://www.google.com/search?hl=es&ie=UTF-8&oe=UTF-8&q=%22políticas+de+seguridad+informática%22&btnG=B%C3%BAsqueda+en+Google&lr=lang\\_es](http://www.google.com/search?hl=es&ie=UTF-8&oe=UTF-8&q=%22políticas+de+seguridad+informática%22&btnG=B%C3%BAsqueda+en+Google&lr=lang_es)

<http://www.notariado.org/noticias/escriturapublica/16%20escriturapublica/la@2.htm>

<http://pp.terra.com.mx/hugalde/virussoy.html>

<http://www.belt.es/articulos/articulo.asp?id=65>

<http://www.ctv.es/USERS/mpq/estrado/estrado004.html>

<http://web.bemarnet.es/seguridad.html>

<http://www.filetopia.org/es/politics.htm>

<http://www.ispjae.edu.cu/eventos/citel/articulos/seguridad.htm>

[http://216.239.53.100/search?q=cache:X69gAEDxg5QC:europa.eu.int/informati on+society/eeurope/news+library/pdf+files/netsec.es.pdf+%22seguridad+de+redes%22&hl=es&lr=lang\\_es&ie=UTF-8](http://216.239.53.100/search?q=cache:X69gAEDxg5QC:europa.eu.int/informati on+society/eeurope/news+library/pdf+files/netsec.es.pdf+%22seguridad+de+redes%22&hl=es&lr=lang_es&ie=UTF-8)

[http://www.netpecos.org/docs/mysql\\_postgres/index.html](http://www.netpecos.org/docs/mysql_postgres/index.html)

<http://www.devx.com/dbzone/Article/20743/1954?pf=true>

<http://www.postgresql.org/docs/7.4/interactive/index.html>

<http://scalab.uc3m.es/~docweb/pr-inf/manuales/Tutoriales/java20.pdf>

<http://www.upv.es/amiga/43.htm>

<http://www.manual-java.com/manualjava/caracteristicas-java.html>

[http://www.germinus.com/consultorio/desarrollo/cons\\_soft\\_2.htm](http://www.germinus.com/consultorio/desarrollo/cons_soft_2.htm)

<http://www.programacionfacil.com/javacon/nueve1.htm>

<http://www.informaticamilenium.com.mx/Paginas/espanol/sitioweb.htm#dred>

<http://usuarios.lycos.es/janjo/janjo1.html>

<http://www.ilustrados.com/publicaciones/EpypkEkApAeZpzKVme.php>

[http://www.pchardware.org/redes/redes\\_intro.php](http://www.pchardware.org/redes/redes_intro.php)

<http://www.monografias.com/trabajos24/arquitectura-cliente-servidor/arquitectura-cliente-servidor.shtml>

[www.elrincondelvago.com](http://www.elrincondelvago.com)

[www.pcm.gob.pe/portal\\_ongei/publicaciones/cultura/Lib5038](http://www.pcm.gob.pe/portal_ongei/publicaciones/cultura/Lib5038)