



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**SISTEMA PARA LA ADMINISTRACIÓN DE
EQUIPOS DE CÓMPUTO
(SAECOM)**

T É S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

PRESENTA:

ERICK OCTAVIO BENÍTEZ LÓPEZ

DIRECTOR DE TESIS: ING. CRUZ SERGIO AGUILAR DIAZ

CIUDAD UNIVERSITARIA, MÉXICO D.F. OCTUBRE 2012





Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



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.

ÍNDICE

Introducción	I
--------------	---

CAPÍTULO 1 ANTECEDENTES

1.1 Panorama General	1
1.2 Objetivo General	1
1.3 Finalidad de la Aplicación	1

CAPÍTULO 2 IDENTIFICACIÓN DE LA PROBLEMÁTICA

2.1 Introducción	2
2.2 Definición de los Alcances del Sistema	2
2.3 Definición de los Requerimientos	4

CAPÍTULO 3 FUNDAMENTOS TEÓRICOS

3.1 Desarrollo de Sistemas Informáticos	5
3.1.1 Sistema	5
3.1.2 Ingeniería de Software	6
3.2 Metodologías de Desarrollo	7
3.2.1 Ciclo de vida	7
3.2.2 Ciclo de vida en cascada	9
3.2.3 Ciclo de vida en V	10
3.2.4 Ciclo de vida por prototipos	11
3.2.5 Ciclo de vida en espiral	13
3.2.6 Herramientas de desarrollo	15
3.2.7 Diagramas de flujo de dato	15
3.3 Técnicas de implementación de sistemas	17
3.3.1 Metodología estructurada	17
3.3.2 Metodología UML	20
3.4 Sistemas Operativos	21
3.4.1 Definición y clasificación	21
3.4.2 Software libre y propietario	22
3.4.3 Aplicaciones de los sistemas operativos	23
3.5 Lenguajes de desarrollo de sistemas	24
3.5.1 Conceptos	24

3.5.2 Lenguaje HTML	27
3.5.3 Lenguaje Javascript	28
3.5.4 Lenguaje PHP	28
3.5.5 Lenguaje JSP	30
3.5.6 Lenguaje Python	32
3.5.7 Lenguaje ASP	34
3.5.8 Lenguaje ASP.NET	35
3.6 Bases de Datos	36
3.6.1 Conceptos	36
3.6.2 MySQL	43
3.6.3 PostgreSQL	44
3.6.4 Oracle	45
3.6.5 SQL Server	47
3.7 Servidores	52
3.7.1 Tipos de servidores y sus aplicaciones	52
3.8 Seguridad	54
3.8.1 Conceptos	55
3.8.2 Políticas	69
3.8.3 Herramientas de seguridad y protección	60
3.8.3.1 Firewall	60
3.8.3.2 Antivirus	66
3.8.3.3 Respaldos	70
3.9 Navegadores	71
3.9.1 Tipos de navegadores	72
CAPÍTULO 4 ANÁLISIS Y DISEÑO DEL SISTEMA	75
4.1 Desarrollo del sistema	75
4.2 Dimensionamiento de la base de datos	76
4.3 Diagrama entidad-relación	78
4.4 Diccionario de datos	79
4.5 Diseño de pantallas	81
CAPÍTULO 5 IMPLEMENTACIÓN DEL SISTEMA	84
5.1 Creación de la base de datos	84
5.2 Creación y diseño de módulos	87
5.2.1 Módulo de acceso	87
5.2.2 Módulo de administración y mantenimiento	87

5.2.3 Módulo de captura, consulta e impresión	88
5.4 Creación del portal web del sistema	89
5.5 Configuración y puesta en marcha	98
CAPÍTULO 6 PRUEBAS Y DEPURACIÓN DEL SISTEMA	103
6.1 Pruebas de lógica	103
6.2 Pruebas unitarias y de integración	104
6.3 Detección de errores de validación	106
6.4 Pruebas de carga y de recorrido de la información	107
6.5 Pruebas de funcionalidad	111
6.6 Pruebas de volumen	112
CAPÍTULO 7 DOCUMENTACIÓN Y RESULTADOS	113
7.1 Creación de la documentación del sistema	113
7.2 Creación del manual de operación del sistema	113
7.3 Creación del manual de usuario	114
7.4 Resultados del sistema	127
CONCLUSIONES	
APÉNDICES	
GLOSARIO	
BIBLIOGRAFÍA	
MESOGRAFÍA	

INTRODUCCIÓN

Hoy día tenemos muchas herramientas tecnológicas de las cuales podemos hacer uso como el software libre el cual podemos utilizarlo, como su nombre lo menciona, sin necesidad de realizar algún pago o adquirir alguna licencia. En el cual podemos implementar herramientas que faciliten las actividades que realizamos profesionalmente.

Este trabajo muestra como fue implementando el sistema para la administración de equipos de cómputo, basado en software libre principalmente, así como las bases teóricas que se utilizaron para su implementación con la finalidad de que su funcionamiento cumpla con las necesidades.

En el capítulo uno se presenta un panorama general dando a conocer las desventajas que se tienen actualmente en algunos lugares con la administración de sus equipos de cómputo. Además de los objetivos y la finalidad del sistema.

En el capítulo dos se analiza los requerimientos y alcances del sistema así como las herramientas necesarias tanto de hardware como de software para el funcionamiento del sistema.

En el tercer capítulo se especifican los fundamentos teóricos para la realización del sistema, ya que son importantes para el desarrollo adecuado de todo proyecto.

En el capítulo cuatro se lleva a cabo el análisis del sistema de acuerdo a los requerimientos del cliente, en base a ello se hará la elección de software, hardware y se realiza el diseño de los módulos que se adapten a las necesidades del lugar donde se utilizará dicho sistema.

En el capítulo quinto con base en el análisis realizado al sistema, se lleva a cabo la creación de la base de datos y los módulos planteados para la implementación del sistema.

En el capítulo seis, se realizará una serie de pruebas “testing” que nos permitirán verificar y revelar la calidad del sistema identificando posibles fallos de implementación, calidad, o usabilidad del mismo. Comprobando el grado de cumplimiento respecto de las especificaciones iniciales del sistema.

El en capítulo siete se realizará la documentación del sistema, el manual de operación, manual del usuario final y por último los resultados del sistema.

CAPÍTULO 1 ANTECEDENTES

1.1 PANORAMA GENERAL

En algunas instituciones o empresas la administración de los equipos de cómputo es muy deficiente pues no cuentan con la información completa o actualizada simplemente es un mero inventario el cual muchas veces su elaboración es arcaica su forma de almacenamiento en papel dificulta el proceso de consulta, es lento, y dicho material en el que se asienta la información tiende a deteriorarse por el uso y pensar en realizar un respaldo, ya que podría ocurrir algún incidente natural o humano que dañe de manera parcial o total la información, sería muy complejo.

Si ya se cuenta con dichos equipos es conveniente la introducción de medios informáticos para el almacenamiento y manipulación de la información. Con esto se pretende tener eficiencia en el manejo, consulta y administración de los equipos.

1.2 OBJETIVO GENERAL

- El sistema para la administración de equipos de cómputo tiene como objetivo agilizar el registro y consultar la información que se vaya generando durante su periodo de uso.
- El sistema debe proporcionar una interfaz de fácil manejo y amigable, que permita intuitivamente su utilización.
- Seguridad en la información, mediante el uso de un manejador de base de datos.
- Sólo se almacenará la información correcta cuidando así la integridad de esta.
- Proporcione información mediante reportes generales y particulares que cubran las necesidades de los usuarios.

1.3 FINALIDAD DE LA APLICACIÓN

Con la finalidad de agilizar la administración de información se proporcionará un software basado en un ambiente gráfico el cual será de fácil utilización, adaptándose a las necesidades del cliente.

El cual permita el acceso a la información a los usuarios para su consulta, modificación, adición y creación de la misma.

Para la seguridad dicho software funcionara de manera interna y solo se podrá consultar cierta información de acuerdo a las políticas de privacidad y seguridad de la información a través de la web.

El acceso será de acuerdo al usuario. Por lo que el sistema estará estructurado en base a perfiles, los cuales permitirán el acceso a ciertos módulos de la aplicación dependiendo de las actividades que el usuario desarrolle en el lugar donde se esté utilizando dicho sistema o fuera si se requiere.

CAPÍTULO 2 IDENTIFICACIÓN DE LA PROBLEMÁTICA.

2.1 INTRODUCCIÓN

De acuerdo al análisis que se ha realizado hemos podido observar que algunas instituciones o empresas no cuentan con la información completa o actualizada de sus equipos de cómputo. Información que nos permita saber el número de equipos que se tiene, el estado en el que se encuentran, su ubicación, la persona responsable de los mismos.

2.2 DEFINICIÓN DE LOS ALCANCES DEL SISTEMA

Es necesario determinar los alcances y límites que debe cumplir el sistema en cuanto al manejo de la información ya que es importante contar con ella, en caso que se desee adquirir nuevos equipos, en alguna auditoría o por seguridad de los mismo en caso de que ocurran pérdidas o desastres.

Se contemplará cuatro tipos de perfiles

- Cliente
- Técnico/Ingeniero
- Administrador
- Mesa de ayuda

El perfil **Cliente** únicamente tendrá acceso para realizar reportes o comentarios de su equipo.

El perfil **Técnico/Ingeniero** puede registrar en inventario equipo nuevo, bajas de equipo, modificación de información de los mismos y agregar reportes; cerrar reportes.

El perfil **Administrador** será el encargado de crear cuentas de usuarios, eliminar cuentas, soporte, mantenimiento a la base de datos y generar reportes. Tendrá la posibilidad de modificar el sistema de acuerdo a las necesidades que se presenten a lo largo del tiempo.

El perfil del usuario **Mesa de ayuda** permitirá agregar reportes de manera rápida que le soliciten vía telefónica u otro medio cabe señalar que dicho perfil deberá contar con las herramientas necesarias que permitan al usuario de este perfil ingresar en el menor tiempo posible las incidencias que le sean reportadas.

Mesa de ayuda

Ofrece la posibilidad de gestionar y solucionar todas las posibles incidencias de manera integral, junto con la atención de requerimientos relacionados con las TICs (Tecnologías de Información y Comunicaciones).

Los términos y su uso se extienden con la popularización del estándar ITIL para el gerenciamiento de TI.

Contenido

- 1 Descripción
- 2 Analista
- 3 Componentes
- 4 Funciones
- 5 Organización
- 6 Enlaces externos

Descripción

Como su nombre lo dice, es un servicio de Mesa de Ayuda, donde se ofrecen Servicios acerca de soporte técnico (bugs, consultas, etc.). Ayuda a incrementar la productividad y aumenta la satisfacción de los usuarios internos y externos de una empresa u organización.

Analista

El analista de la Mesa de Ayuda debe tener habilidades, conocimientos y capacidades, necesarias para la atención eficiente al usuario. La primera, debe explicar una solución a los requerimientos del usuario, usando la lógica y razonamiento para identificar las fortalezas y debilidades de soluciones alternativas brindadas a los usuarios.

También debe poseer conocimientos de software, hardware y telecomunicaciones, todo relacionado con el area, además de las políticas de la organización, así como también tener capacidades comunicacionales fundamentales, tales como escuchar y comprender la información, y expresar las ideas en forma oral, a problemas específicos, para así lograr respuestas coherentes. Lo más importante es dar una solución, dejar al usuario satisfecho con las gestiones del analista.

Componentes

La *Mesa de Ayuda* se basa en un conjunto de recursos técnicos y humanos que permiten dar soporte a diferentes niveles de usuarios informáticos de una empresa, tales como:

- Servicio de soporte a usuarios de “sistemas microinformáticos”.
- Soporte telefónico centralizado online
- Atendido de forma inmediata e individualizada por Técnicos Especializados.
- Apoyado sobre un Sistema informático de última generación.
- Permite asignar tareas a técnicos propios o externos a su empresa.

Funciones

Una Mesa de Ayuda tiene varias funciones. Este provee a los usuarios un punto central para recibir ayuda en varios temas referentes a la computadora. El help desk típicamente administra sus peticiones vía software que permite dar seguimiento a las peticiones del usuario con un único número de Ticket. Esto también puede ser llamado "Seguimiento Local de Fallos" o LBT por sus siglas en inglés (Local Bug Tracker). Este software, a menudo puede ser una herramienta extremadamente benéfica cuando se usa para encontrar, analizar y eliminar problemas comunes en un ambiente computacional de la organización.

En una Mesa de ayuda¹, el usuario notifica su problema, y este emite un ticket que contiene los detalles del problema; si el primer nivel es capaz de resolver el problema, el ticket es cerrado y actualizado con la documentación de la solución para permitir a otros técnicos de servicio tener una referencia. Si el problema necesita ser escalado, este será despachado a un segundo nivel.

Los grandes Centros de Servicio tienen diferentes niveles para manejar los diferentes tipos de preguntas.

2.3 DEFINICIÓN DE LOS REQUERIMIENTOS

El sistema que se implementará, para su funcionamiento, requiere un servidor web para su publicación, si se pretende tener en línea, un motor de bases de datos, un equipo de cómputo preferentemente sólo para el sistema.

El sistema a desarrollar deberá contar para su implementación, con los siguientes requisitos:

- Un servidor web
- Un manejador de base de datos

El servidor debe contar con las características que a continuación se mencionan:

- Procesador como a 2 GHz o superior
- Disco duro de 120 GB o superior
- Sistema Operativo Linux distribución Ubuntu ya que el ambiente es muy amigable.

CAPÍTULO 3 FUNDAMENTOS TEÓRICOS

3.1 DESARROLLO DE SISTEMAS INFORMÁTICOS

Podemos definir el desarrollo de sistemas informáticos como el proceso mediante el cual el conocimiento humano y el uso de las ideas son llevados a las computadoras; de manera que pueda realizar las tareas para la cual fue desarrollada.

Para que esto sea utilizado deberán, funcionar adecuadamente, ser de fácil manejo, adecuarse a la empresa para la que fué diseñada y debe ayudar al personal a realizar su trabajo de forma eficiente

3.1.1 SISTEMA

Un sistema se define como un conjunto de elementos relacionados entre sí para llegar a un fin común. Enfocándonos al área de sistemas informáticos, dichos elementos son el *hardware*, el *software* y los usuarios.

Aunque es muy común usar “Sistema Informático” y “Sistema de Información” como sinónimos son términos diferentes. Un sistema de información comprende elementos como personas, datos, técnicas y recursos materiales para la administración de los datos y su posterior aprovechamiento. En el sistema informático se ven involucrados equipos de cómputo que permitirán el almacenamiento, procesamiento y acceso a dicha información.

Los sistemas, aunque tengan diferentes objetivos, comparten elementos de los que dependerá su buen funcionamiento.

Uno de los agentes más importantes que forma parte de un sistema informático son las personas, ya que estas son las que ingresan los datos por medio de periféricos de entrada, dando pie a una serie de procesos y generando la información que puede ser consultada.

Otro de los elementos que integran los sistemas son los procedimientos, los cuales atienden a las actividades del usuario, procesando los datos y generando la información que posteriormente será consultada, limitando este acceso solo a las personas autorizadas. Un tercer componente que podemos encontrar es el equipo, que hace referencia a cuestiones técnicas como los dispositivos de cómputo y en

general, la infraestructura necesaria para que el sistema funcione de manera correcta.

3.2.1 INGENIERÍA DE SOFTWARE

El *software* se puede definir como el conjunto de componentes lógicos (procedimientos, reglas, datos, etc.) que hacen funcionar y realizar tareas específicas a una computadora, pueden ir desde lo que es el sistema operativo hasta aplicaciones específicas como editores de texto, videojuegos, editores de imágenes y más.

En un principio, el software era desarrollado por quien tenía una necesidad específica y lo producía conforme a su experiencia o mejor dicho, de acuerdo a lo que su intuición le dictara, más que nada, la programación era un arte.

Para generar un producto de *software* se requieren de ciertos métodos y técnicas para que el desarrollo sea de calidad, de ahí que haya surgido la Ingeniería de Software, rama de la ingeniería que cuida los aspectos mediante la aplicación de procesos previamente corroborados.

La Ingeniería de Software se puede definir como el estudio de principios y metodologías para generar el conocimiento necesario para el buen diseño, desarrollo, operación y mantenimiento del *software*.

Algunos problemas que se llegaron a detectar en la producción de *software* incluyen:

- Retrasos considerables en la entrega.
- Poca productividad.
- Elevados costos de producción y cargas de mantenimiento.
- Baja calidad y fiabilidad del producto.
- Gran dificultad en el mantenimiento.

A lo anterior se le conoce como crisis del *software* y es mediante el desarrollo de ciertas metodologías y la implementación de nuevas herramientas con las que se pretende evitar problemas.

Lo que se busca con estos cambios es solucionar problemas de administración, calidad, productividad y sobre todo, facilidad de mantenimiento, actividad que demanda una gran cantidad de recursos de toda índole (humanos, materiales y económicos).

Dentro de la Ingeniería de Software se han desarrollado diferentes metodologías que ayudan a mejorar la producción, este proceso se denomina ciclo de desarrollo

o ciclo de vida e incluye varias fases que van desde el diseño, pasando por la codificación y pruebas, hasta el mantenimiento del producto.

3.2 METODOLOGÍA DE DESARROLLO

La Ingeniería de Software es un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo de sistemas de información

3.2.1 CICLO DE VIDA

El ciclo de vida del software se utiliza para estructurar las actividades que se llevarán a cabo en el desarrollo de un producto.

En todo sistema se deben considerar tiempo y asignación de recursos para el desarrollo y posteriormente, para el mantenimiento y que el producto siga teniendo una vida útil.

Existen varios modelos de ciclo de vida, dependiendo de esto serán las etapas que lo componen y la forma en cómo se llevará a cabo la realización del proyecto.

Independientemente del modelo que se trate, hay etapas en común:

- Análisis: Estudio de los requerimientos para determinar su viabilidad.
- Diseño: Abstracción del sistema y elaboración de la interfaz de usuario.
- Codificación: Propiamente la elaboración del sistema como tal.
- Pruebas: Verificación del correcto funcionamiento del producto.
- Mantenimiento: Correcciones y/o modificaciones al sistema.

Análisis

Tanto el desarrollador como el usuario tienen un papel activo en la Ingeniería de Software, es aquí, en esta interacción usuario - desarrollador donde nacen un conjunto de actividades llamadas análisis. El usuario intentará planear un sistema confuso a nivel de descripción de datos, funciones y comportamiento. El desarrollador debe actuar como interrogador, como un consultor que resuelve problemas y como negociador.

El análisis de requisitos permite al ingeniero de sistemas especificar las características operacionales del *software* (función, datos y rendimientos), indica la

relación de la interfaz con otros elementos del sistema y establece las restricciones que debe cumplir el producto.

El análisis de requisitos del *software* se enfoca en cinco áreas de trabajo:

- Reconocimiento del problema
- Evaluación del problema
- Modelado
- Especificación
- Revisión

Diseño

El diseño del *software* es realmente un proceso multipasos que se enfoca sobre cuatro atributos distintos del problema:

- Estructura de datos.
- Arquitectura del *software*.
- Representaciones de interfaz.
- Detalle procedimental (algoritmos).

El proceso de diseño traduce requisitos de una representación del *software* donde se pueda evaluar su calidad antes de que comience la codificación. Si el diseño se ejecuta de una manera detallada, la codificación puede realizarse mecánicamente.

Codificación

El diseño debe traducirse en una forma legible para la máquina, para este fin se utilizan lenguajes de programación que pueden ser procesados a diferentes niveles por los equipos de cómputo. La etapa de codificación puede variar en función de la tecnología en la que se va a implementar la solución de *software*.

Pruebas

Una vez que se ha generado el código, comienzan las pruebas del programa. Las cuales se enfocan en la lógica interna del *software*, asegurando que todas las sentencias se han probado, y sobre las funciones externas. Es decir, se realizan pruebas integrales para asegurar que la entrada definida producirá los resultados que realmente se requieren.

Mantenimiento

El *software*, indudablemente sufrirá cambios después de que se entregue al usuario final. Esto se deberá a diversas razones, por ello, el mantenimiento del *software* se aplica a cada uno de los pasos precedentes.

A continuación se describen algunos modelos de ciclo de vida:

3.2.2 CICLO DE VIDA EN CASCADA

También se le conoce como modelo lineal secuencial. Es una técnica que se derivó de otras ramas de la ingeniería, tenía como objetivo el mejorar la calidad del software y reducir sus costos.

Es el modelo que más ha sido utilizado. La forma en cómo opera consiste en concluir cada etapa con un alto grado de exactitud y verificar si es posible continuar con la siguiente. Una vez que se ha llegado al periodo de mantenimiento es posible regresar a etapas anteriores para hacer cambios, haciendo nuevamente el recorrido hasta llegar al final. Ver figura 3.1

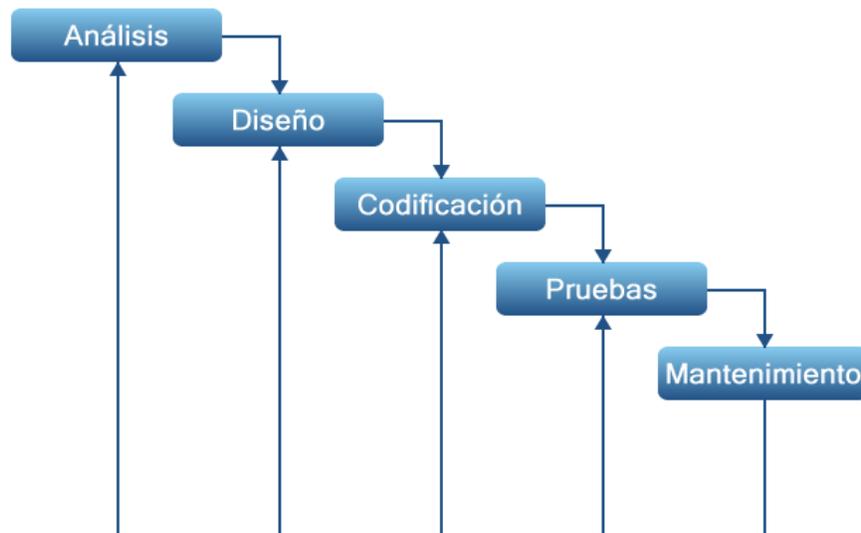


Figura 3.1 Ciclo de vida en cascada

Ventajas

- Facilidad en la planificación.
- Producto resultante de alta calidad.

Desventajas

- Disponibilidad de todos los requisitos del sistema.
- Resultados perceptibles hasta que el proyecto ya se encuentre en una etapa avanzada.

- Gran impacto de errores cuando el proyecto va avanzado.
- Repercusión en los tiempos de entrega cuando hay retrasos en etapas tempranas.
- Mayor costo del desarrollo.

Un ejemplo de aplicación de este modelo lo podemos encontrar en un proyecto de reingeniería, donde ya están establecidas las especificaciones.

3.2.3 CICLO DE VIDA EN V

Está basado en el modelo en cascada y consta de las mismas fases, con la ventaja de que una fase también nos sirve para verificar o validar otras etapas.

En la figura 3.2 podemos ver que el análisis y diseño se encuentran del lado izquierdo y del derecho tenemos las pruebas y el mantenimiento y el vértice lo hace la fase de codificación.

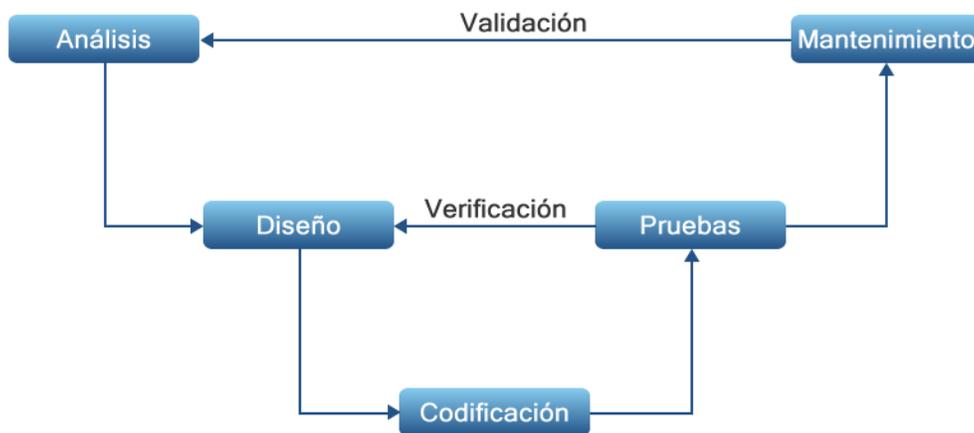


Figura 3.2 Ciclo de vida en “V”

Ventajas

- Facilidad en la detección de fallos gracias a la relación entre desarrollo y pruebas.
- El modelo se puede ajustar completamente a las necesidades del proyecto.
- Ofrece información detallada (instrucciones, recomendaciones, explicaciones, etc.) sobre cómo implementar cada etapa y el camino a seguir.

Desventajas

- Necesidad de todos los requerimientos por parte del usuario.
- Producto disponible hasta el final del ciclo de vida.
- Costo elevado en el periodo de pruebas.

3.2.4 CICLO DE VIDA POR PROTOTIPOS

Forma parte de los modelos de desarrollo evolutivo y es de gran ayuda cuando los requerimientos no están claramente establecidos, cuando no se sabe cómo implementar la interfaz de usuario o en algún otro caso que implique una gran cantidad de modificaciones y/o interacción con el usuario.

El prototipo nos ofrece un mejor medio para comunicarnos con el usuario y así poder identificar los requerimientos. Gracias a la retroalimentación se afinan detalles para empezar con el desarrollo formal del proyecto.

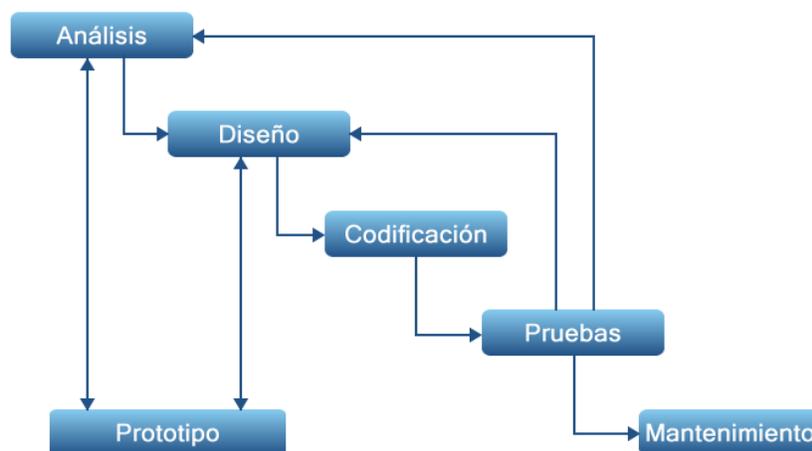


Figura 3.3 Ciclo de vida por prototipos

Ventajas

- No es necesaria la descripción completa de los requisitos.
- Se pueden realizar modificaciones en etapas tempranas del desarrollo.
- Entendimiento más claro por parte del usuario ya que el planteamiento no se encuentra solamente en la documentación.

Desventajas

- Se puede llegar a extender excesivamente en el tiempo de desarrollo del proyecto.
- Difícil administración si no se tiene un objetivo claro del *software* a desarrollar.
- El usuario e incluso los desarrolladores pueden considerar al prototipo como un sistema terminado.

Modelo DRA

El Desarrollo Rápido de Aplicaciones (DRA) es un modelo de proceso del desarrollo del *software* lineal secuencial que enfatiza un ciclo de desarrollo extremadamente corto, como se muestra en la siguiente figura:

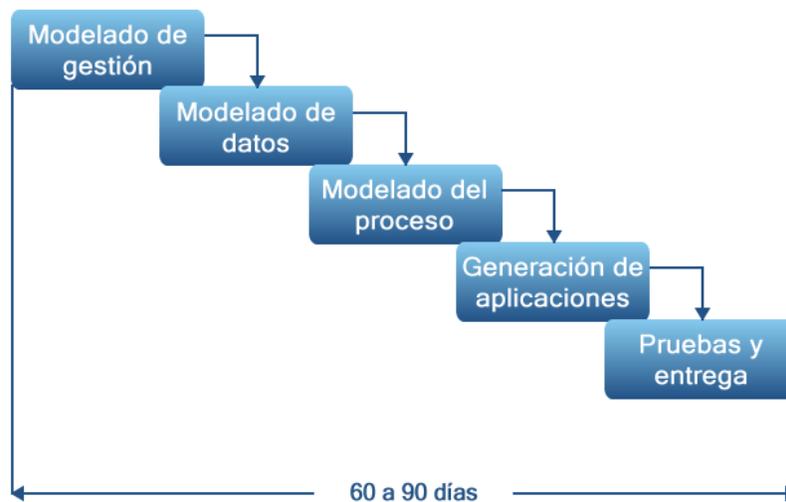


Figura 3.4 Modelo DRA

En el DRA se logra el desarrollo rápido utilizando un enfoque de construcción basado en componentes. Si se comprenden bien los requisitos y se limita el alcance del proyecto, el proceso DRA permite al equipo de desarrollo crear un sistema completamente funcional dentro de periodos cortos de tiempo. Cuando se utiliza principalmente para aplicaciones de sistemas de información, el enfoque DRA comprende las siguientes fases:

Modelado de gestión

El flujo de información entre las funciones de gestión se modela de forma que responda a las siguientes preguntas:

- ¿Qué información conduce el proceso de gestión?
- ¿Qué información se genera?
- ¿Quién la genera?
- ¿A dónde va la información?
- ¿Quién la procesó?

Modelado de datos

El flujo de información definido como parte de la fase de modelado de gestión se refina como un conjunto de objetos de datos necesarios para apoyar la empresa. Se definen las características, llamadas atributos, de cada uno de los objetos y las relaciones entre ellos.

Modelado de proceso

Los objetos de datos definidos en la fase de modelado de datos quedan transformados para lograr el flujo de información necesario para implementar una función de gestión. Las descripciones del proceso se crean para añadir, modificar, suprimir, o recuperar un objeto de datos.

Generación de aplicaciones

El DRA asume la utilización de técnicas de cuarta generación. En lugar de crear *software* con lenguajes de programación de tercera generación, el proceso DRA trabaja para volver a utilizar componentes de programas ya existentes (cuando es posible) o crear componentes reutilizables (cuando sea necesario). En todos los casos se utilizan herramientas automáticas para facilitar la construcción del *software*.

Pruebas de entrega

Como el proceso DRA enfatiza la reutilización ya se han comprobado muchos de los componentes de los programas, esto reduce tiempo de pruebas. Sin embargo, se deben probar los componentes nuevos y se deben ejercitar todas las interfaces a fondo.

3.2.5 CICLO DE VIDA EN ESPIRAL

El modelo en espiral consta de una serie de ciclos que se repiten dando como resultado un producto con mejoras respecto del anterior.

Este modelo es similar al de prototipo pero añade un componente denominado análisis de riesgos. Éstos abarcan aspectos como la mala interpretación de requerimientos, errores de diseño, problemas de implementación, etc.

El primer paso es definir los objetivos y posteriormente se realiza el análisis de los riesgos. Si después de haber hecho dicho análisis se observa que hay incertidumbre sobre el problema, entonces se hace uso de un prototipo con el fin de entenderlo mejor y depurar los requerimientos proporcionados por el usuario.

Una vez terminada esta parte, el usuario evalúa los productos obtenidos y hace la retroalimentación para hacer las modificaciones necesarias y mejorar el desarrollo.

En cada vuelta de la espiral, la actividad de ingeniería se realiza mediante el ciclo de cascada o por medio del modelo de prototipos.

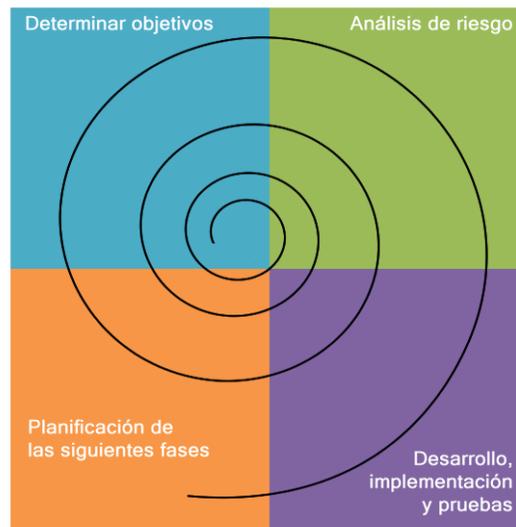


Figura 3.5 Ciclo de vida en espiral

Ventajas

- No se necesita una definición completa de los requerimientos para iniciar el proyecto.
- Validación de requerimientos desde etapas tempranas.
- Facilidad para detectar errores de análisis y diseño.

Desventajas

- Participación constante por parte del usuario.

- Mucho tiempo de desarrollo y alto costo.
- Requiere experiencia en la identificación de riesgos.

3.2.6 HERRAMIENTAS DE DESARROLLO

Las herramientas CASE (*Computer Aided Software Engineering* - Ingeniería de Software Asistida por Computadora) son un conjunto de utilidades, métodos y técnicas empleadas por los profesionales del desarrollo de *software* para facilitar y agilizar el ciclo de vida en un sistema, ya sea completamente o solo enfocándose en alguna de las etapas que lo componen. Es útil en tareas como el diseño del proyecto, cálculo de costos, generación de código a partir del diseño, generación automática de documentación, etc.

Actualmente son varias las instituciones públicas que hacen uso de estas herramientas puesto que facilitan las tareas del ciclo de vida, como las etapas de codificación e implementación de pruebas, permitiendo un desarrollo más rápido de los sistemas, a su vez reducen los costos de producción, haciéndolas más productivas y ofreciendo una mayor calidad de servicio al público.

Algunas de las características y ventajas de las herramientas CASE sobre el desarrollo del *software* son:

- Permiten la comprobación de errores en etapas tempranas del desarrollo.
- Tienen soporte de reusabilidad.
- Mejoran la productividad en el desarrollo y mantenimiento.
- Aumentan la calidad del producto.
- Reducen el tiempo y el costo de desarrollo mediante la generación de código, documentación, automatización de pruebas, etc.
- Ayudan a tener una mejor planificación de un proyecto.
- Incrementan la biblioteca de conocimiento informático de una organización ayudando a la búsqueda de soluciones para los futuros desarrollos.
- Facilitan el uso de las distintas metodologías propias de la Ingeniería del *Software*.

3.2.7 DIAGRAMA DE FLUJO DE DATOS

Es una representación gráfica para la maceta del "flujo" de datos a través de un sistema de información. Un diagrama de flujo de datos también se puede utilizar para la visualización de procesamiento de datos (diseño estructurado). Es una práctica común para un diseñador dibujar un contexto a nivel de diagrama de flujo de datos (DFD) que primero muestra la interacción entre el sistema y las entidades externas. Este contexto a nivel de DFD se "explotó" para mostrar más detalles del sistema que se está modelando.

Los diagramas de flujo de datos fueron inventados por Larry Constantine, el desarrollador original del diseño estructurado, basado en el modelo de computación de Martin y Estrin: "flujo gráfico de datos".

Los DFD son una de las tres perspectivas esenciales de Análisis de Sistemas Estructurados y Diseño por Método SSADM. El patrocinador de un proyecto y los usuarios finales tendrán que ser informados y consultados en todas las etapas de una evolución del sistema. Con un diagrama de flujo de datos, los usuarios van a poder visualizar la forma en que el sistema funcione, lo que el sistema va a lograr, y cómo el sistema se pondrá en práctica.

El antiguo sistema de diagramas de flujo de datos puede ser elaborado y se comparó con el nuevo sistema de diagramas de flujo para establecer diferencias y mejoras a aplicar para desarrollar un sistema más eficiente. Los diagramas de flujo de datos pueden ser usados para proporcionar al usuario final una idea física de cómo resultarán los datos a última instancia, y cómo tienen un efecto sobre la estructura de todo el sistema. La manera en que cualquier sistema es desarrollado puede determinarse a través de un diagrama de flujo de datos. El desarrollo de un DFD ayuda en la identificación de los datos de la transacción en el modelo de datos.

Niveles, los cuales son:

- **Nivel 0:** Diagrama de contexto.
- **Nivel 1:** Diagrama de nivel superior.
- **Nivel 2:** Diagrama de detalle o expansión.

Diagrama de Contexto: Nivel 0

En el diagrama de contexto se caracterizan todas las interacciones que realiza un sistema con su entorno (entidades externas), estas pueden ser otros sistemas, sectores internos a la organización, o factores externos a la misma. Se dibuja un sólo proceso que representa al sistema en cuestión y se escribe su nombre en dicha burbuja como un sustantivo común más adjetivos. De él solamente parten los flujos de datos que denotan las interrelaciones entre el sistema y sus agentes externos, no admitiéndose otros procesos ni almacenamientos en el dibujo.

Resulta de gran utilidad para los niveles posteriores de análisis como herramienta de balanceo.

Diagrama de Nivel Superior: Nivel 1

En el diagrama de nivel superior se plasman todos los procesos que describen al proceso principal. En este nivel los procesos no suelen interrelacionarse directamente, sino que entre ellos debe existir algún almacenamiento o entidad externa que los una. Esta regla de construcción sirve como ayuda al analista para

contemplar que en un nivel tan elevado de abstracción (DFD Nivel 1) es altamente probable que la información que se maneja requiera ser almacenada en el sistema aunque no esté especificado por un requisito funcional, siendo en realidad un requisito no-funcional.

Diagrama de Detalle o Expansión: Nivel 2

En un diagrama de nivel 2 o mayor, comienzan a explotarse las excepciones a los caminos principales de la información dado que aumenta progresivamente el nivel de detalle. De aquí en adelante se permiten los flujos entre procesos.

El DFD nivel 2 puede considerarse el máximo para ser validado en forma conjunta con el usuario dado que en los niveles posteriores el alto grado de complejidad del diagrama puede resultar de muy difícil lectura para personas ajenas al equipo de sistemas. También se recomienda el diagrama de nivel superior.

3.3 TÉCNICAS DE IMPLEMENTACIÓN DE SISTEMAS

Para el desarrollo de sistemas hacemos uso de alguna de las metodologías que se han tratado. Una metodología se puede definir como el conjunto de procedimientos necesarios para llegar a un objetivo. Enfocándonos en la parte de desarrollo de sistemas, la metodología es un modo sistemático de producir *software*, donde el proyecto se divide en etapas y cada una de estas tiene tareas a realizar, además de tener bien definidas las entradas que recibe y las salidas que produce.

Lo que se busca mediante el uso de una metodología son cuatro puntos principales:

- Satisfacer las necesidades del usuario.
- Realizar cambios fácilmente una vez que el sistema esté en funcionamiento.
- Proporcionar una interfaz intuitiva y amigable para usar el sistema.
- Producir un sistema capaz de funcionar sin fallos y con la mínima cantidad de correcciones posibles.

Al igual que en el proceso de producción de un carro, o en la elaboración de un refresco, en las metodologías podemos encontrar distintos caminos dependiendo de la forma como se quieran satisfacer los requerimientos dados: la metodología estructurada y la metodología orientada a objetos.

3.3.1 METODOLOGÍA ESTRUCTURADA

La metodología estructurada busca crear modelos de forma descendente, es decir, descompone el sistema de forma que se puedan tener partes lo más pequeñas posibles. La metodología estructurada se divide a su vez en orientada a procesos y orientada a datos.

Existen varias herramientas que se utilizan en la metodología orientada a procesos, entre las que encontramos:

Diagramas de Flujo de Datos (DFD): representan la forma en la que los datos se mueven y se transforman e incluye:

- Procesos
- Flujos de datos
- Almacenamiento de datos

Los procesos individuales se pueden a su vez descomponer en otros DFD de nivel superior.

Especificaciones de procesos: es una herramienta que nos define las acciones que lleva a cabo una función en el sistema, es decir, lo que hay en medio de las entradas y las salidas.

Diccionario de datos (DD)

Contiene las características lógicas (si es una cadena, entero, decimal, la longitud, rango de valores, si debe o no tener un valor, etc.) para ser utilizados en un sistema.

Diagrama de Transición de Estados (DTE)

Son herramientas que modelan procesos que dependen del tiempo, se basan en el estado que tiene el sistema y en los cambios de estado.

Diagramas Entidad - Relación (DER)

Nos ayuda al modelado de datos mostrando las entidades principales, sus propiedades y la interrelación que hay entre ellas.

Metodología Orientada a Objetos (MOO)

Es una metodología de más reciente creación que está enfocada al uso de componentes, teniendo la ventaja de reutilizar código y de que su mantenimiento sea más fácil debido a la mejor localización de los cambios.

El paradigma de orientación a objetos se centra en la entidad principal (objeto), la cual contiene los datos y la forma en cómo estos datos pueden ser manipulados.

El objeto es una abstracción de los entes del mundo real y se puede utilizar por medio de la instanciación de una clase, que no es otra cosa más que la esencia del objeto. En otras palabras, la clase nos define las características del objeto.

Algunos de los rasgos más característicos de la orientación a objetos son:

Herencia

Como si se tratase de una familia, las clases también pueden heredar métodos y atributos de una clase padre.

Polimorfismo

Es la capacidad de nombrar a un método (que puede hacer cosas diferentes) de la misma forma para objetos de diferente clase.

Encapsulamiento

Se le llama así a la propiedad de los objetos que hace que aquellas características no necesarias para interactuar con otros no sean visibles.

Metodología OMT

La metodología OMT (Object Modeling Technique Técnica de Modelado de Objetos) es una práctica que nos sirve para el diseño y modelado de un sistema de *software*.

Algunas de sus características principales es que considera la comunicación con el usuario, muestra la información de una forma diferente y disminuye la complejidad del problema mediante la segmentación de procesos.

La etapa de diseño se descompone en:

Modelo de objeto

Representa los artefactos del sistema, es decir las clases y asociaciones con atributos y operaciones.

Modelo dinámico

Constituye las interacciones entre los componentes ya mencionados, estas interacciones son eventos, estados y transiciones.

Modelo funcional

Hace referencia a los métodos del sistema desde el punto de vista de flujo de datos.

Esta metodología pone especial interés en las etapas de análisis y diseño, siendo la precursora de UML.

3.3.2 METODOLOGÍA UML

UML (*Unified Modeling Language* - Lenguaje de Modelado Unificado) es un lenguaje de modelado de sistemas de software muy utilizado actualmente y aprobado en el 2005 por la ISO como un estándar.

Este lenguaje busca hacer una descripción completa del modelo y los artefactos del sistema, contempla formas de representación para procesos de negocios, funciones del sistema, expresiones del lenguaje de programación, esquemas de bases de datos, etc.

Existen 2 versiones de UML, la que podemos encontrar actualmente es la 2.3 y cuenta con 13 tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas y se agrupan en diagramas de estructura, comportamiento e interacción.

Los diagramas de estructura enfatizan en los elementos que deben existir en el sistema modelado:

- Diagrama de clases
- Diagrama de componentes
- Diagrama de objetos
- Diagrama de estructura compuesta
- Diagrama de despliegue
- Diagrama de paquetes

Los diagramas de comportamiento hacen hincapié en lo que debe suceder en el sistema modelado:

- Diagrama de actividades
- Diagrama de casos de uso
- Diagrama de estados

Los diagramas de interacción son un subtipo de diagramas de comportamiento, que prestan especial importancia sobre el flujo de control y de datos entre los elementos del sistema modelado:

- Diagrama de secuencia

- Diagrama de comunicación
- Diagrama de tiempos
- Diagrama global de interacciones

3.4 SISTEMAS OPERATIVOS

El **sistema operativo** es el programa (o software) más importante de un ordenador. Para que funcionen los otros programas, cada ordenador de uso general debe tener un sistema operativo. Los sistemas operativos realizan tareas básicas, tales como reconocimiento de la conexión del teclado, enviar la información a la pantalla, no perder de vista archivos y directorios en el disco, y controlar los dispositivos periféricos tales como impresoras, escáner, etc.

En sistemas grandes, el sistema operativo tiene incluso mayor responsabilidad y poder, es como un policía de tráfico, se asegura de que los programas y usuarios que están funcionando al mismo tiempo no interfieran entre ellos. El sistema operativo también es responsable de la seguridad, asegurándose de que los usuarios no autorizados no tengan acceso al sistema.

3.4.1 DEFINICIÓN Y CLASIFICACIÓN

Los sistemas operativos surgen con la necesidad de agilizar y manipular de una manera eficaz las computadoras y crear un medio para que el usuario pueda operarla con un entorno, lenguaje y operación bien definido para hacer un verdadero uso y explotación de ésta.

Un sistema operativo es el encargado de brindar al usuario una forma amigable y sencilla de operar, interpretar, codificar y emitir las ordenes al procesador central para que este realice las tareas necesarias y específicas para completar una orden.

El sistema operativo, es el instrumento indispensable para hacer de la computadora un objeto útil. Bajo este nombre se agrupan todo aquel software que permite a los usuarios la utilización de estos, que de otra manera serian difíciles de controlar. Un sistema operativo se define como un conjunto de procedimientos manuales y automáticos, que permiten a un grupo de usuarios compartir una instalación de computadora eficazmente

Los sistemas operativos pueden ser clasificados de la siguiente forma:

- **Multiusuario:** Permite que dos o más usuarios utilicen sus programas al mismo tiempo. Algunos sistemas operativos permiten a centenares o millares de usuarios al mismo tiempo.
- **Multiprocesador:** soporta el abrir un mismo programa en más de una CPU.

- **Multitarea:** Permite que varios programas se ejecuten al mismo tiempo.
- **Multitramo:** Permite que diversas partes de un solo programa funcionen al mismo tiempo.
- **Tiempo Real:** Responde a las entradas inmediatamente. Los sistemas operativos como DOS y UNIX, no funcionan en tiempo real.

3.4.2 SOFTWARE LIBRE Y PROPIETARIO

Definamos antes que es un software libre y un propietario

El software libre es un programa o secuencia de instrucciones usada por un dispositivo de procesamiento digital de datos para llevar a cabo una tarea específica o resolver un problema determinado, sobre el cual su dueño renuncia a la posibilidad de obtener utilidades por las licencias, patentes, o cualquier forma que adopte su derecho de propiedad sobre él (o sea, el software libre puede estar licenciado, o protegido por una patente autoral), por lo que puede utilizarse o transferirse sin pago alguno al licenciante, o a su creador. Otra característica es que se encuentra disponible el código fuente del software, por lo que puede modificarse el software sin ningún límite, y sin pago a quien lo inventó o lanzó al mercado.

El opuesto del Software libre es el denominado software propietario, aquel que es imposible de utilizar en otro hardware, o terminal modificar, o transferir sin pagar derechos a su inventor o creador.

Para ello, quienes han patentado software libre, lo han hecho permitiendo las actividades recién nombradas. Así nace el Copyleft (el opuesto del Copyright de los derechos autorales), que es básicamente el principio de que cualquier modificación o cambio (“derivative works”), debe quedar disponible para el resto de la comunidad bajo el mismo licenciamiento original. Con ello se fomenta el trabajo colaborativo en el desarrollo de la informática.

También es importante recalcar algunos puntos importantes para un buen desarrollo del mismo:

El software se desarrolla o construye; no se manufactura en el sentido clásico. A pesar de que existen similitudes entre el desarrollo del software y la manufactura del hardware, las dos actividades serían diferentes en lo fundamental. En ambas la alta calidad se alcanza por medio del buen diseño, la fase de manufactura del hardware puede incluir problemas de calidad existentes en el software.

El software no se desgasta. El software es inmune a los males ambientales que desgasten el hardware. Por lo tanto la curva de tasas de fallas para el software debería tener la forma de la “curva idealizada”. Los defectos sin descubrir causan tasas de fallas altas en las primeras etapas de vida de un programa. Sin embargo,

los errores se corrigen y la curva se aplanan: el software no se desgasta, pero si se deteriora.

A pesar de que la industria tiene una tendencia hacia la construcción por componentes, la mayoría del software aun se construye a la medida. Un componente de software se debe diseñar e implementar de forma que puede utilizarse en muchos programas diferentes.

Características de un buen software:

- Corrección
- Fiabilidad
- Eficiencia
- Integridad
- Facilidad de uso
- Facilidad de mantenimiento
- Flexibilidad
- Facilidad de prueba
- Portabilidad
- Facilidad de reuso
- Interoperabilidad

3.4.3 APLICACIONES DE LOS SISTEMAS OPERATIVOS

Un sistema operativo es un programa o conjunto de programas que sirven para gestionar los recursos de un equipo de cómputo. Algunos ejemplos de plataformas de sistemas operativos más conocidos son Microsoft Windows, Mac OS, Linux y Unix. Para funcionar bien, un sistema operativo tiene carpetas y directorios. Si los borramos, se estropeará. Es lo mismo que nos sucedería si quitáramos las carpetas de un programa como Word o Paint.

Algunos ejemplos de las aplicaciones de los sistemas operativos son:

- Servidores Firewall
- Desarrollo
- Servidores dedicados y no dedicados
- Servidores de antivirus
- Servidores de dominio
- Servidores para la administración y almacenamiento de bases de datos

Así como también existe software para uso específico como:

- Software de desarrollo
- Software de servidor

- Software para manejo de base de datos
- Software para servicios de web
- Software educativo
- Software de seguridad

3.5 LENGUAJES DE DESARROLLO DE SISTEMAS

3.5.1 CONCEPTOS

Un lenguaje de programación es un lenguaje que puede ser utilizado para controlar el comportamiento de una máquina, particularmente una computadora. Consiste en un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones.

Un lenguaje de programación permite especificar de manera precisa: sobre qué datos una computadora debe operar, cómo deben ser estos almacenados, transmitidos y qué acciones debe tomar bajo una variada gama de circunstancias. Todo esto, a través de un lenguaje que intenta estar relativamente próximo al lenguaje humano o natural, tal como sucede con el lenguaje léxico. Una característica relevante de los lenguajes de programación es que puedan tener un conjunto común de instrucciones que puedan ser comprendidas para realizar la construcción del programa.

Los procesadores usados en las computadoras son capaces de entender y actuar según lo indican programas escritos en un lenguaje fijo llamado lenguaje máquina.

Todo programa escrito en otro lenguaje puede ser ejecutado de dos maneras:

- Mediante un programa que va adaptando las instrucciones conforme son encontradas. A este proceso se lo llama interpretar y a los programas que lo hacen se los conoce como intérpretes.
- Traduciendo este programa al programa equivalente escrito en lenguaje de máquina. A ese proceso se lo llama compilar y al traductor se lo conoce como compilador.

CLASIFICACIÓN DE LOS LENGUAJES DE PROGRAMACIÓN

Los lenguajes de programación se determinan según el nivel de abstracción, Según la forma de ejecución y según el paradigma de programación que poseen cada uno de ellos y esos pueden ser:

SEGÚN SU NIVEL DE ABSTRACCIÓN

LENGUAJES DE BAJO NIVEL

Los lenguajes de bajo nivel son lenguajes de programación que se acercan al funcionamiento de una computadora. El lenguaje de más bajo nivel es el código máquina y el lenguaje ensamblador.

- El lenguaje máquina: Este lenguaje ordena a la máquina las operaciones fundamentales para su funcionamiento. Consiste en la combinación de 0's y 1's para formar las ordenes entendibles por el hardware de la máquina. Este lenguaje es mucho más rápido que los lenguajes de alto nivel.
- El lenguaje ensamblador: Es un derivado del lenguaje máquina y está formado por abreviaturas de letras y números llamadas mnemotécnicos. Una de las ventajas con respecto al código máquina es que los códigos fuentes son más cortos y los programas creados ocupan menos memoria.

LENGUAJES DE MEDIO NIVEL

Hay lenguajes de programación que son considerados como lenguajes de medio nivel ya que puede acceder a los registros del sistema, trabajar con direcciones de memoria, todas ellas características de lenguajes de bajo nivel y a la vez realizar operaciones de alto nivel.

LENGUAJES DE ALTO NIVEL

Son aquellos que se encuentran más cercanos al lenguaje natural que al lenguaje máquina.

Se tratan de lenguajes independientes de la arquitectura del ordenador. Por lo que, en principio, un programa escrito en un lenguaje de alto nivel, lo puedes migrar de una máquina a otra sin ningún tipo de problema. Los lenguajes de alto nivel son normalmente fáciles de aprender porque están formados por elementos de lenguajes naturales, como el inglés.

A diferencia de los programas de ensamble, los programas en lenguaje de alto nivel se pueden utilizar con diferentes marcas de computadores sin tener que hacer modificaciones considerables.

Otras ventajas de los lenguajes de alto nivel son:

- Son más fáciles de aprender que los lenguajes ensambladores
- Se pueden escribir más rápidamente
- Permiten tener mejor documentación

- Son más fáciles de mantener

SEGÚN LA FORMA DE EJECUCIÓN

LENGUAJES COMPILADOS

Naturalmente, un programa que se escribe en un lenguaje de alto nivel también tiene que traducirse a un código que pueda utilizar la máquina. Los programas traductores que pueden realizar esta operación se llaman compiladores. Éstos, como los programas ensambladores avanzados, pueden generar muchas líneas de código de máquina por cada proposición del programa fuente.

Los compiladores son aquellos cuya función es traducir un programa escrito en un determinado lenguaje a un idioma que la computadora entienda (lenguaje máquina con código binario).

Al usar un lenguaje compilado, el programa desarrollado nunca se ejecuta mientras haya errores, sino hasta que luego de haber compilado el programa.

LENGUAJES INTERPRETADOS

Se puede también utilizar una alternativa diferente de los compiladores para traducir lenguajes de alto nivel. En lugar de traducir el programa fuente y grabar en forma permanente el código objeto que se produce durante la compilación, el usuario sólo carga el programa fuente en la computadora junto con los datos que se van a procesar, para que posteriormente un programa intérprete, almacenado en el sistema operativo del disco, o incluido de manera permanente dentro de la máquina, convierta cada proposición del programa fuente en lenguaje de máquina conforme vaya siendo necesario durante el proceso de los datos.

En el caso de que el procesamiento sea repetitivo cada instrucción tendrá que volver a ser interpretado cada vez que se ejecute, lo cual hace que el programa sea más lento en tiempo de ejecución pero más rápido en tiempo de diseño porque no se tiene que estar compilando a cada momento el código completo. El intérprete elimina la necesidad de realizar la compilación después de cada modificación del programa cuando se quiere agregar funciones o corregir errores; pero es obvio que un programa objeto compilado con antelación deberá ejecutarse con mucha mayor rapidez que uno que se debe interpretar a cada paso de su producción.

SEGÚN EL PARADIGMA DE PROGRAMACIÓN

Un paradigma de programación representa un enfoque particular o filosofía para la construcción del software. No es mejor uno que otro sino que cada uno tiene ventajas y desventajas. También hay situaciones donde un paradigma resulta más apropiado que otro.

Atendiendo al paradigma de programación, se pueden clasificar los lenguajes en:

Lenguajes imperativos

- BASIC
- C
- C++
- Java
- C#
- Perl

Lenguajes funcionales

- Haskell
- Miranda

Híbridos

- Lisp
- Scheme
- Ocaml
- Standard ML
- ML
- Scala

Lenguajes lógicos

- Prolog

3.5.2 LENGUAJE HTML

HTML, sigla de HyperText Markup Language (Lenguaje de Etiquetas de Hipertexto), es el lenguaje de marcado predominante para la construcción de páginas web. Es usado para describir la estructura y el contenido en forma de texto, así como para complementar el texto con objetos tales como imágenes. HTML se escribe en forma de etiquetas. HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un (Javascript).

El lenguaje HTML puede ser creado y editado con cualquier editor de textos básico, como puede ser Gedit, el Bloc de Notas de Windows, o cualquier otro editor que admita texto sin formato como GNU Emacs, Microsoft Wordpad, TextPad, Vim, Note pad++, etc.

Existen además, otros programas para la realización de sitios Web o edición de código HTML, como Microsoft FrontPage, el cual tiene un formato básico parecido al resto de los programas de Office. También existe el famoso software de Macromedia llamado Dreamweaver, siendo uno de los más utilizados en el ámbito de diseño y programación Web. Estos programas se les conoce como editores WYSIWYG.

3.5.3 LENGUAJE JAVASCRIPT

JavaScript es un lenguaje de programación interpretado que no requiere compilación, utilizado principalmente en páginas web, con una sintaxis semejante a la del lenguaje Java y el lenguaje C.

Al contrario que Java, JavaScript no es un lenguaje orientado a objetos, ya que no dispone de Herencia, es un lenguaje basado en prototipos, por lo que las nuevas clases se generan clonando las clases base (prototipos) y extendiendo su funcionalidad.

Todos los navegadores modernos interpretan el código JavaScript integrado dentro de las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

El lenguaje fue inventado por Brendan Eich en la empresa Netscape Communications, que es la que desarrolló los primeros navegadores web comerciales. Apareció por primera vez en el producto de Netscape llamado Netscape Navigator 2.0.

Tradicionalmente, se venía utilizando en páginas web HTML, para realizar tareas y operaciones en el marco de la aplicación únicamente cliente, sin acceso a funciones del servidor. JavaScript se ejecuta en el agente de usuario al mismo tiempo que las sentencias van descargándose junto con el código HTML.

3.5.4 LENGUAJE PHP

PHP es un lenguaje de programación interpretado usado normalmente para la creación de páginas web dinámicas. PHP es un acrónimo recursivo que significa "PHP Hypertext Pre-processor"³. Actualmente también se puede utilizar para la creación de otros tipos de programas incluyendo aplicaciones con interfaz gráfica usando las bibliotecas Qt o GTK+.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos, tales como UNIX (y de ese tipo, como Linux o Mac OS X) y Windows, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

PHP es una alternativa a las tecnologías de Microsoft ASP y ASP.NET (que utiliza C#/VB.NET como lenguajes), a ColdFusion de la compañía Adobe (antes Macromedia), a JSP/Java de Sun Microsystems, y a CGI/Perl.

Usos de PHP

Los principales usos del PHP son los siguientes:

- Programación de páginas web dinámicas, habitualmente en combinación con el motor de base datos MySQL, aunque cuenta con soporte nativo para otros motores, incluyendo el estándar ODBC, lo que amplía en gran medida sus posibilidades de conexión.
- Programación en consola, al estilo de Perl o Shell scripting.
- Creación de aplicaciones gráficas independientes del navegador, por medio de la combinación de PHP y Qt/GTK+, lo que permite desarrollar aplicaciones de escritorio en los sistemas operativos en los que está soportado.

Características de PHP

Ventajas

- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Biblioteca nativa de funciones sumamente amplia e incluida.
- No requiere definición de tipos de variables.
- Tiene manejo de excepciones.



Figura 3.6 Funcionamiento del PHP

Desventajas

- No posee una abstracción de base de datos estándar, sino bibliotecas especializadas para cada motor
- No posee adecuado manejo de internacionalización, unicode, etc.
- Por su diseño dinámico no puede ser compilado y es muy difícil de optimizar.
- Por sus características promueve la creación de código desordenado y complejo de mantener.

3.5.5 LENGUAJE JSP

Es una tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

Esta tecnología es un desarrollo de la compañía Sun Microsystems. La Especificación JSP 1.2 fue la primera que se liberó y en la actualidad está disponible la Especificación JSP 2.1.

Las JSP's permiten la utilización de código Java mediante scripts. Además, es posible utilizar algunas acciones JSP predefinidas mediante etiquetas. Estas

etiquetas pueden ser enriquecidas mediante la utilización de Bibliotecas de Etiquetas (TagLibs o Tag Libraries) externas e incluso personalizadas.

JSP puede considerarse como una manera alternativa, y simplificada, de construir servlets. Es por ello que una página JSP puede hacer todo lo que un servlet puede hacer, y viceversa. Cada versión de la especificación de JSP está fuertemente vinculada a una versión en particular de la especificación de servlets.

El funcionamiento general de la tecnología JSP es que el Servidor de Aplicaciones interpreta el código contenido en la página JSP para construir el código Java del servlet a generar. Este servlet será el que genere el documento (típicamente HTML) que se presentará en la pantalla del Navegador del usuario.

JSP -> Servidor Aplicaciones (Servlets) -> Cliente (Navegador)

Es posible enriquecer el lenguaje de etiquetas utilizado por JSP. Para ello debemos extender la capa de alto nivel JSP mediante la implementación de Bibliotecas de Etiquetas (Tags Libraries). Un ejemplo de estas bibliotecas son las proporcionadas por Sun bajo la denominación de JSTL o las distribuidas por Apache junto con el Framework de Struts.

TagLibs -> JSP -> Servidor Aplicaciones (Servlets) -> Cliente (Navegador)

El rendimiento de una página JSP es el mismo que tendría el servlet equivalente, ya que el código es compilado como cualquier otra clase Java. A su vez, la máquina virtual compilará dinámicamente a código de máquina las partes de la aplicación que lo requieran. Esto hace que JSP tenga un buen desempeño y sea más eficiente que otras tecnologías web que ejecutan el código de una manera puramente interpretada.

La principal ventaja de **JSP** frente a otros lenguajes es que el lenguaje Java es un lenguaje de propósito general que excede el mundo web y que es apto para crear clases que manejen lógica de negocio y acceso a datos de una manera prolija. Esto permite separar en niveles las aplicaciones web, dejando la parte encargada de generar el documento HTML en el archivo JSP.

Otra ventaja es que JSP hereda la portabilidad de Java, y es posible ejecutar las aplicaciones en múltiples plataformas sin cambios. Es común incluso que los desarrolladores trabajen en una plataforma y que la aplicación termine siendo ejecutada en otra.

Los servlets y Java Server Pages (JSPs) son dos métodos de creación de páginas web dinámicas en servidor usando el lenguaje Java. En ese sentido son similares a otros métodos o lenguajes tales como el PHP, ASP o los CGIs, programas que generan páginas web en el servidor. Sin embargo, se diferencian de ellos en otras cosas.

Para empezar, los JSPs y servlets se ejecutan en una máquina virtual Java, lo cual permite que, en principio, se puedan usar en cualquier tipo de ordenador, siempre que exista una máquina virtual Java para él. Cada servlet (o JSP, a partir de ahora lo usaremos de forma indistinta) se ejecuta en su propia hebra, es decir, en su propio contexto; pero no se comienza a ejecutar cada vez que recibe una petición, sino que persiste de una petición a la siguiente, de forma que no se pierde tiempo en invocarlo (cargar programa + intérprete). Su persistencia le permite también hacer una serie de cosas de forma más eficiente: conexión a bases de datos y manejo de sesiones, por ejemplo.

Las JSPs son en realidad una forma alternativa de crear servlets ya que el código JSP se traduce a código de servlet Java la primera vez que se le invoca y en adelante es el código del nuevo servlet el que se ejecuta produciendo como salida el código HTML que compone la página web de respuesta.

3.5.6 LENGUAJE PYTHON

Es un lenguaje de programación de alto nivel cuya filosofía hace hincapié en una sintaxis muy limpia y que favorezca un código legible.

Se trata de un lenguaje de programación multiparadigma ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, usa tipado dinámico, es fuertemente tipado y multiplataforma.

Es administrado por la Python Software Foundation. Posee una licencia de código abierto, denominada Python Software Foundation License, que es compatible con la Licencia pública general de GNU a partir de la versión 2.1.1, e incompatible en ciertas versiones anteriores.

Python fue creado a finales de los ochenta por Guido van Rossum en el Centro para las Matemáticas y la Informática (CWI, Centrum Wiskunde & Informatica), en los Países Bajos, como un sucesor del lenguaje de programación ABC, capaz de manejar excepciones e interactuar con el sistema operativo Amoeba.

El nombre del lenguaje proviene de la afición de su creador original, Guido van Rossum, por los humoristas británicos Monty Python.

Van Rossum es el principal autor de Python, y su continuo rol central en decidir la dirección de Python es reconocido, refiriéndose a él como *Benevolente Dictador Vitalicio* (en inglés: *Benevolent Dictator for Life*, BDFL).

En 1991, van Rossum publicó el código de la versión 0.9.0 en alt.sources. En esta etapa del desarrollo ya estaban presentes clases con herencia, manejo de excepciones, funciones y los tipos modulares, como: str, list, dict, entre otros. Además en este lanzamiento inicial aparecía un sistema de módulos adoptado de

Modula-3; van Rossum describe el módulo como “una de las mayores unidades de programación de Python”. El modelo de excepciones en Python es parecido al de Modula-3, con la adición de una cláusula else. En el año 1994 se formó comp.lang.python, el foro de discusión principal de Python, marcando un hito en el crecimiento del grupo de usuarios de este lenguaje.

Python alcanzó la versión 1.0 en enero de 1994. Una característica de este lanzamiento fueron las herramientas de la programación funcional: lambda, reduce, filter y map. Van Rossum explicó que “hace 12 años, Python adquirió lambda, reduce(), filter() y map(), cortesía de un hacker de Lisp que las extrañaba y que envió parches”. El donante fue Amrit Prem; no se hace ninguna mención específica de cualquier herencia de Lisp en las notas de lanzamiento.

La última versión liberada proveniente de CWI fue Python 1.2. En 1995, van Rossum continuó su trabajo en Python en la Corporation for National Research Initiatives (CNRI) en Reston, Virginia, donde lanzó varias versiones del software.

Python 2.0 tomó una característica mayor del lenguaje de programación funcional Haskell: listas por comprensión. La sintaxis de Python para esta construcción es muy similar a la de Haskell, salvo por la preferencia de los caracteres de puntuación en Haskell, y la preferencia de Python por palabras claves alfabéticas. Python 2.0 introdujo además un sistema de recolección de basura capaz de recolectar referencias cíclicas. .

Python 2.1 fue un trabajo derivado de Python 1.6.1, así como también de Python 2.0. Su licencia fue renombrada a: Python Software Foundation License. Todo el código, documentación y especificaciones añadidas, desde la fecha del lanzamiento de la versión alfa de Python 2.1, tiene como dueño a Python Software Foundation (PSF), una organización sin ánimo de lucro fundada en el año 2001, tomando como modelo la Apache Software Foundation. Incluido en este lanzamiento fue una implementación del scoping más parecida a las reglas de static scoping

Python es un lenguaje de programación multiparadigma. Esto significa que más que forzar a los programadores a adoptar un estilo particular de programación, permite varios estilos: programación orientada a objetos, programación imperativa y programación funcional. Otros paradigmas están soportados mediante el uso de extensiones.

Python usa tipado dinámico y conteo de referencias para la administración de memoria.

Una característica importante de Python es la resolución dinámica de nombres; es decir, lo que enlaza un método y un nombre de variable durante la ejecución del programa (también llamado ligadura dinámica de métodos).

Otro objetivo del diseño del lenguaje es la facilidad de extensión. Se pueden escribir nuevos módulos fácilmente en C o C++. Python puede incluirse en aplicaciones que necesitan una interfaz programable.

3.5.7 LENGUAJE ASP

Active Server Pages (ASP), también conocido como ASP clásico, es una tecnología de Microsoft del tipo "lado del servidor" para páginas web generadas dinámicamente, que ha sido comercializada como un anexo a Internet Information Services (IIS).

- 1 Descripción
- 2 Versiones
- 3 Ejemplo
- 4 Véase también
- 5 Enlaces externos

La tecnología ASP está estrechamente relacionada con el modelo tecnológico y de negocio de su fabricante. Intenta ser solución para un modelo de programación rápida ya que "programar en ASP es como programar en Visual Basic y C#", por supuesto con muchas limitaciones y algunas ventajas específicas en entornos web.

Lo interesante de este modelo tecnológico es poder utilizar diversos componentes ya desarrollados como algunos controles ActiveX así como componentes del lado del servidor, tales como CDONTS, por ejemplo, que permite la interacción de los scripts con el servidor SMTP que integra IIS.

Se facilita la programación de sitios web mediante varios objetos integrados, como por ejemplo un objeto de sesión basada en cookies, que mantiene las variables mientras se pasa de página a página.

Es limitado a solo funcionar con IIS, por lo que su uso es cuestionado por la mayoría de los programadores web quienes prefieren otros lenguajes de programación del lado del servidor como por ejemplo PHP, Perl, Java etc.

Ha pasado por cuatro versiones mayores:

- ASP 1.0 (distribuido con IIS 3.0)
- ASP 2.0 (distribuido con IIS 4.0)
- ASP 3.0 (distribuido con IIS 5.0)
- ASP.NET (parte de la plataforma .NET de Microsoft).

Las versiones pre-.NET se denominan actualmente (desde 2002) como ASP *clásico*.

En el último ASP clásico, ASP 3.0, hay siete objetos integrados disponibles para el programador: Application, ASPError, Request, Response, Server, Session yObjectContext. Cada objeto tiene un grupo de funcionalidades frecuentemente usadas y útiles para crear páginas web dinámicas.

Desde 2002, el ASP *clásico* está siendo reemplazado por ASP.NET, que entre otras cosas, reemplaza los lenguajes interpretados como VBScript o JScript por lenguajes compilados a código intermedio (llamado MSIL o Microsoft Intermediate Language) como Visual Basic .NET, C#, o cualquier otro lenguaje que soporte la plataforma .NET. El código MSIL se compila con posterioridad a código nativo.

3.5.8 LENGUAJE ASP.NET

Herramienta de desarrollo web comercializado por Microsoft. Es usado por programadores para construir sitios web dinámicos, aplicaciones web y servicios XML. Forma parte de la plataforma .NET de Microsoft y es la tecnología sucesora de la tecnología Active Server Pages (ASP).

Microsoft desarrolló una nueva tecnología denominada ASP.NET como parte de su estrategia.NET para el desarrollo Web, con el objetivo de resolver las limitaciones de ASP y posibilitar la creación de software como servicio. ASP.NET es la plataforma unificada de desarrollo web que proporciona los servicios necesarios para crear aplicaciones web.

En la actualidad una aplicación ASP.NET puede ejecutarse de dos formas distintas:

Aplicaciones cliente/servidor: Estas aplicaciones están típicamente en formato de ejecutables compilados. Estos pueden integrar una interfaz de usuario, Pero acostumbran ser menos gestionables y escalables que las demás aplicaciones.

Aplicaciones que utilizan el navegador: Dichas aplicaciones están caracterizadas por contar con una interfaz de web . La interfaz gráfica integra varias tecnologías, las cuales son el HTML, XHTML, scripting, etc; siempre y cuando el navegador que se esté utilizando soporte estas tecnologías.

3.6 BASES DE DATOS

Una base de datos es un “almacén” que nos permite guardar grandes cantidades de información de forma organizada para que luego podamos encontrar y utilizar fácilmente. A continuación se presenta una guía que te explicará el concepto y características de las bases de datos.



3.7 Bases de Datos

El término de bases de datos fue escuchado por primera vez en 1963, en un simposio celebrado en California, USA. Una **base de datos**⁴ se puede definir como un conjunto de información relacionada que se encuentra agrupada ó estructurada.

Desde el punto de vista informático, la base de datos es un sistema formado por un conjunto de datos almacenados en discos que permiten el acceso directo a ellos y un conjunto de programas que manipulen ese conjunto de datos.

Cada base de datos se compone de una o más tablas que guarda un conjunto de datos. Cada tabla tiene una o más **columnas** y **filas**. Las columnas guardan una parte de la información sobre cada elemento que queramos guardar en la tabla, cada fila de la tabla conforma un registro.

3.6.1 CONCEPTOS

Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular. De la cual podemos obtener información específica consultada mediante un lenguaje estructurado llamado SQL.

Características

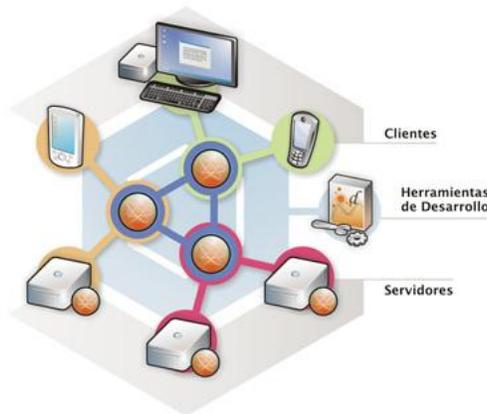
Entre las principales características de los sistemas de base de datos podemos mencionar:

- Independencia lógica y física de los datos

- Redundancia mínima
- Acceso concurrente por parte de múltiples usuarios
- Integridad de los datos
- Consultas complejas optimizadas
- Seguridad de acceso y auditoria
- Respaldo y recuperación
- Acceso a través de lenguajes de programación estándar

Sistema de Gestión de Base de Datos

Los Sistemas de Gestión de Base de Datos (SGBD) (en inglés DataBase Management System) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta.



3.8 Sistema Gestor de Bases de Datos

Ventajas de las bases de datos

Control sobre la redundancia de datos:

Los sistemas de ficheros almacenan varias copias de los mismos datos en ficheros distintos. Esto hace que se desperdicie espacio de almacenamiento, además de provocar la falta de consistencia de datos.

En los sistemas de bases de datos todos estos ficheros están integrados, por lo que no se almacenan varias copias de los mismos datos. Sin embargo, en una base de datos no se puede eliminar la redundancia completamente, ya que en ocasiones es necesaria para modelar las relaciones entre los datos.

Consistencia de datos:

Eliminando o controlando las redundancias de datos se reduce en gran medida el riesgo de que haya inconsistencias. Si un dato está almacenado una sola vez, cualquier actualización se debe realizar sólo una vez, y está disponible para todos los usuarios inmediatamente. Si un dato está duplicado y el sistema conoce esta redundancia, el propio sistema puede encargarse de garantizar que todas las copias se mantienen consistentes.

Compartición de datos:

En los sistemas de ficheros, los ficheros pertenecen a las personas o a los departamentos que los utilizan. Pero en los sistemas de bases de datos, la base de datos pertenece a la empresa y puede ser compartida por todos los usuarios que estén autorizados.

Mantenimiento de estándares:

Gracias a la integración es más fácil respetar los estándares necesarios, tanto los establecidos a nivel de la empresa como los nacionales e internacionales. Estos estándares pueden establecerse sobre el formato de los datos para facilitar su intercambio, pueden ser estándares de documentación, procedimientos de actualización y también reglas de acceso.

Mejora en la integridad de datos:

La integridad de la base de datos se refiere a la validez y la consistencia de los datos almacenados. Normalmente, la integridad se expresa mediante restricciones o reglas que no se pueden violar. Estas restricciones se pueden aplicar tanto a los datos, como a sus relaciones, y es el SGBD quien se debe encargar de mantenerlas.

Mejora en la seguridad:

La seguridad de la base de datos es la protección de la base de datos frente a usuarios no autorizados. Sin unas buenas medidas de seguridad, la integración de datos en los sistemas de bases de datos hace que éstos sean más vulnerables que en los sistemas de archivos.

Mejora en la accesibilidad a los datos:

Muchos SGBD proporcionan lenguajes de consultas o generadores de informes que permiten al usuario hacer cualquier tipo de consulta sobre los datos, sin que sea necesario que un programador escriba una aplicación que realice tal tarea.

Mejora en la productividad:

El SGBD proporciona muchas de las funciones estándar que el programador necesita escribir en un sistema de archivos. A nivel básico, el SGBD proporciona todas las rutinas de manejo de ficheros típicas de los programas de aplicación.

El hecho de disponer de estas funciones permite al programador centrarse mejor en la función específica requerida por los usuarios, sin tener que preocuparse de los detalles de implementación de bajo nivel.

Mejora en el mantenimiento:

En los sistemas de archivos, las descripciones de los datos se encuentran inmersas en los programas de aplicación que los manejan.

Esto hace que los programas sean dependientes de los datos, de modo que un cambio en su estructura, o un cambio en el modo en que se almacena en disco, requiere cambios importantes en los programas cuyos datos se ven afectados.

Sin embargo, los SGBD separan las descripciones de los datos de las aplicaciones. Esto es lo que se conoce como independencia de datos, gracias a la cual se simplifica el mantenimiento de las aplicaciones que acceden a la base de datos.

Aumento de la concurrencia:

En algunos sistemas de archivos, si hay varios usuarios que pueden acceder simultáneamente a un mismo fichero, es posible que el acceso interfiera entre ellos de modo que se pierda información o se pierda la integridad. La mayoría de los SGBD gestionan el acceso concurrente a la base de datos y garantizan que no ocurran problemas de este tipo.

Mejora en los servicios de copias de seguridad:

Muchos sistemas de archivos dejan que sea el usuario quien proporcione las medidas necesarias para proteger los datos ante fallos en el sistema o en las aplicaciones. Los usuarios tienen que hacer copias de seguridad cada día, y si se produce algún fallo, utilizar estas copias para restaurarlos.

En este caso, todo el trabajo realizado sobre los datos desde que se hizo la última copia de seguridad se pierde y se tiene que volver a realizar. Sin embargo, los SGBD actuales funcionan de modo que se minimiza la cantidad de trabajo perdido cuando se produce un fallo.

Desventajas de las bases de datos

Complejidad:

Los SGBD son conjuntos de programas que pueden llegar a ser complejos con una gran funcionalidad. Es preciso comprender muy bien esta funcionalidad para poder realizar un buen uso de ellos.

Coste del equipamiento adicional:

Tanto el SGBD, como la propia base de datos, pueden hacer que sea necesario adquirir más espacio de almacenamiento. Además, para alcanzar las prestaciones deseadas, es posible que sea necesario adquirir una máquina más grande o una máquina que se dedique solamente al SGBD. Todo esto hará que la implantación de un sistema de bases de datos sea más cara.

Vulnerable a los fallos:

El hecho de que todo esté centralizado en el SGBD hace que el sistema sea más vulnerable ante los fallos que puedan producirse. Es por ello que deben tenerse copias de seguridad (Backup).

Tipos de Campos

Cada Sistema de Base de Datos posee tipos de campos que pueden ser similares o diferentes. Entre los más comunes podemos nombrar:

- **Numérico:** Entre los diferentes tipos de campos numéricos podemos encontrar enteros “sin decimales” y reales “decimales”.
- **Booleanos:** Poseen dos estados: Verdadero “Si” y Falso “No”.
- **Memos:** Son campos alfanuméricos de longitud ilimitada. Presentan el inconveniente de no poder ser indexados.
- **Fechas:** Almacenan fechas facilitando posteriormente su explotación. Almacenar fechas de esta forma posibilita ordenar los registros por fechas o calcular los días entre una fecha y otra.
- **Alfanuméricos:** Contienen cifras y letras. Presentan una longitud limitada (255 caracteres).
- **Autoincrementables:** Son campos numéricos enteros que incrementan en una unidad su valor para cada registro incorporado. Su utilidad resulta: Servir de identificador ya que resultan exclusivos de un registro.

Modelo Entidad-Relación (E-R)

Los diagramas o modelos entidad-relación (denominado por sus siglas, ERD “Diagram Entity relationship”) son una herramienta para el modelado de datos de

un sistema de información. Estos modelos expresan entidades relevantes para un sistema de información, sus inter-relaciones y propiedades.

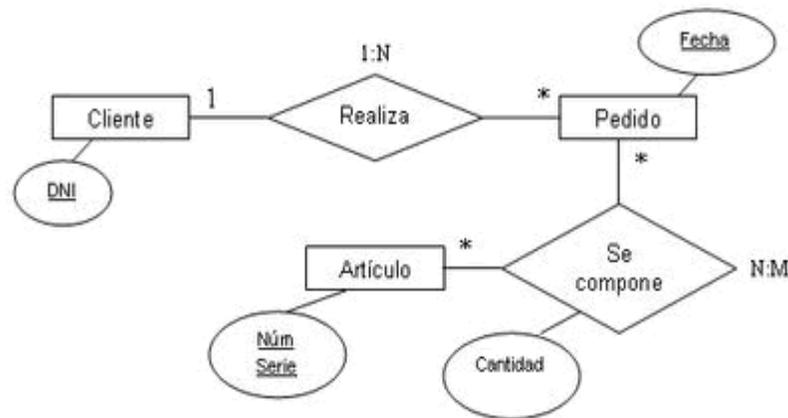


Figura 3.9 Diagrama entidad relación

Cardinalidad de las Relaciones

El diseño de relaciones entre las tablas de una base de datos puede ser la siguiente:

- **Relaciones de uno a uno:** Una instancia de la entidad A se relaciona con una y solamente una de la entidad B.
- **Relaciones de uno a muchos:** Cada instancia de la entidad A se relaciona con varias instancias de la entidad B.
- **Relaciones de muchos a muchos:** Cualquier instancia de la entidad A se relaciona con cualquier instancia de la entidad B.

Estructura de una Base de Datos

Una base de datos, a fin de ordenar la información de manera lógica, posee un orden que debe ser cumplido para acceder a la información de manera coherente. Cada base de datos contiene una o más tablas, que cumplen la función de contener los campos.

En el siguiente ejemplo mostramos una tabla “comentarios” que contiene 4 campos.

	Field	Type
<input type="checkbox"/>	<u>id</u>	int(11)
<input type="checkbox"/>	titulo	varchar(100)
<input type="checkbox"/>	texto	blob
<input type="checkbox"/>	fecha	varchar(10)

Los datos quedarían organizados como mostramos en siguiente ejemplo:

	<u>id</u>	<u>titulo</u>	<u>texto</u>	<u>fecha</u>
<input type="checkbox"/>	1	saludos	[BLOB - 0 B]	22-10-2007
<input type="checkbox"/>	2	como estas ???	[BLOB - 0 B]	23-10-2007

Por consiguiente una base de datos posee el siguiente orden jerárquico:

- Tablas
- Campos
- Registros
- Lenguaje SQL (Structure Query Lenguaje)

El lenguaje SQL es el más universal en los sistemas de base de datos. Este lenguaje nos permite realizar consultas a nuestras bases de datos para mostrar, insertar, actualizar y borrar datos.

A continuación veremos un ejemplo de ellos:

- **Mostrar:** Para mostrar los registros se utiliza la instrucción Select. `Select * From comentarios.`
- **Insertar:** Los registros pueden ser introducidos a partir de sentencias que emplean la instrucción Insert. `Insert Into comentarios (titulo, texto, fecha) Values ('saludos', 'como esta', '22-10-2007')`
- **Borrar:** Para borrar un registro se utiliza la instrucción Delete. En este caso debemos especificar cuál o cuáles son los registros que queremos borrar. Es por ello necesario establecer una selección que se llevara a cabo mediante la cláusula Where. `Delete From comentarios Where id='1'.`
- **Actualizar:** Para actualizar los registros se utiliza la instrucción Update. Como para el caso de Delete, necesitamos especificar por medio de Where cuáles son los registros en los que queremos hacer efectivas nuestras modificaciones. Además, tendremos que especificar cuáles son los nuevos valores de los campos que deseamos actualizar. `Update comentarios Set titulo='Mi Primer Comentario' Where id='1'`

3.6.2 MySQL



3.10 MySQL

Es un sistema de gestión de bases de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems y ésta a su vez de Oracle Corporation desde abril de 2009— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

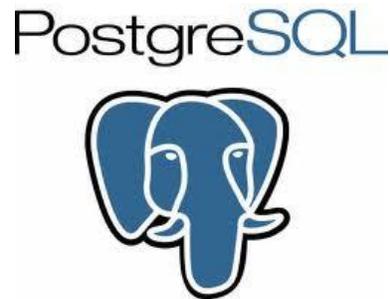
Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero para aquellas empresas que quieran incorporarlo en productos privativos deben comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario de proyectos como Apache, donde el software es desarrollado por una comunidad pública y los derechos de autor del código están en poder del autor individual, MySQL es patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet. MySQL AB fue fundado por David Axmark, Allan Larsson y Michael Widenius.

MySQL es muy utilizado en aplicaciones web, como Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones. Sea cual sea el entorno en el que va a utilizar MySQL, es importante monitorizar de antemano el rendimiento para detectar y corregir errores tanto de SQL como de programación.

3.6.3 POSTGRESQL



3.11 PostgresAQL

Postgres es una base de datos de plataforma múltiple, y se puede encontrar tanto en forma de fuente como compilada para los siguientes sistemas: Solaris, SunOS, HPUX, AIX, Linux, Irix, Digital Unix, BSD, NetBSD, FreeBSD, SCO unix, NEXTSTEP y en general para todos los dialectos Unix, además de para Windows NT.

Los sistemas de mantenimiento de Bases de Datos relacionales tradicionales (DBMS,s) soportan un modelo de datos que consisten en una colección de relaciones con nombre, que contienen atributos de un tipo específico. En los sistemas comerciales actuales, los tipos posibles incluyen numéricos de punto flotante, enteros, cadenas de caracteres, cantidades monetarias y fechas. Está generalmente reconocido que este modelo será inadecuado para las aplicaciones futuras de procesamiento de datos. El modelo relacional sustituyó modelos previos en parte por su "simplicidad espartana". Sin embargo, como se ha mencionado, esta simplicidad también hace muy difícil la implementación de ciertas aplicaciones. Postgres ofrece una potencia adicional sustancial al incorporar los siguientes cuatro conceptos adicionales básicos en una vía en la que los usuarios pueden extender fácilmente el sistema

- clases
- herencia
- tipos
- funciones

Otras características aportan potencia y flexibilidad adicional:

- Restricciones (Constraints)
- Disparadores (triggers)
- Reglas (rules)
- Integridad transaccional

Estas características colocan a Postgres en la categoría de las Bases de Datos identificadas como objeto-relacionales. Nótese que éstas son diferentes de las referidas como orientadas a objetos, que en general no son bien aprovechables para soportar lenguajes de Bases de Datos relacionales tradicionales. Postgres tiene algunas características que son propias del mundo de las bases de datos orientadas a objetos. De hecho, algunas Bases de Datos comerciales han incorporado recientemente características en las que Postgres fue pionera.

3.6.4 ORACLE



3.11 Oracle

Es un manejador de base de datos relacional que hace uso de los recursos del sistema informático en todas las arquitecturas de hardware, para garantizar su aprovechamiento al máximo en ambientes cargados de información.

Es el conjunto de datos que proporciona la capacidad de almacenar y acudir a estos de forma recurrente con un modelo definido como relacional. Además es una suite de productos que ofrece una gran variedad de herramientas.

Es el mayor y más usado Sistema Manejador de Base de Dato Relacional (RDBMS) en el mundo. La Corporación Oracle ofrece este RDBMS como un producto incorporado a la línea de producción. Además incluye cuatro generaciones de desarrollo de aplicación, herramientas de reportes y utilitarios.

Oracle corre en computadoras personales (PC), microcomputadoras, mainframes y computadoras con procesamiento paralelo masivo. Soporta unos 17 idiomas, corre automáticamente en más de 80 arquitecturas de hardware y software distinto sin tener la necesidad de cambiar una sola línea de código. Esto es porque más el 80% de los códigos internos de Oracle son iguales a los establecidos en todas las plataformas de sistemas operativos.

Evolución

El poderoso modelo relacional ha evolucionado desde herramientas y los modelos de datos de redes. La mayor aceptación y uso de un modelo de datos es el

modelo relacional que fue conocido en 1969 con la revisión hecha por IBM, Dr. E. F. Codd.

Un modelo relacional posee tres grandes aspectos:

- Estructuras: Definición de objetos que contengan datos y que son accesibles a los usuarios.
- Operaciones: Definir acciones que manipulen datos u objetos.
- Reglas: Leyes para gobernar la información, cómo y qué manipular.

Una base de datos relacional definida como un modelo de información es estrictamente visualizable por los usuarios mediante tablas. Una tabla está compuesta por una matriz bidimensional de filas y columnas. En cualquier ocasión la información es cambiada en una base de datos relacional, cualquier información es el resultado de una consulta presentada por el usuario en el formato de filas/columnas.

Versiones de ORACLE

Oracle ha presentado cuatro generaciones para desarrollo de aplicación:

Oracle 5 y Oracle 6

Fueron las dos primeras versiones de Oracle, quedando aún rezagadas por las versiones sucesoras.

Oracle 7

La base de datos relacional componentes de Oracle Universal Server. Posee además las versiones 7.1, 7.1.2, y 7.1.3.

Oracle 7 Parallel

Ofrece a los usuarios un método seguro y administrable para incrementar la performance de sus bases de datos existentes introduciendo operaciones en paralelo y sincrónicas dentro de sus ambientes informáticos.

Oracle 8

Incluye mejoras de rendimiento y de utilización de recursos. Independiente de que se necesite dar soporte a decenas de miles de usuarios y cientos de terabytes de datos, o se disponga de un sistema mucho más pequeño, pero igualmente crítico, todos se benefician del rendimiento de Oracle8. Este soporta aplicaciones de procesamiento de transacciones on line (OLTP) y de data warehousing mayores y más exigentes.

Oracle Fail Safe

Protege al sistema de caída de entornos Cluster Windows NT. Este producto es para que el sistema operativo Windows NT disponga de una mayor oferta de soluciones en entornos cluster.

Oracle Universal Server

Con soporte completo de Web, mensajería y datos multimedia, Oracle Universal Server es el eslabón fundamental en el camino de transición de la industria de la tecnología a la informática de red, donde la complejidad de software es transferida desde la PC de escritorio a poderosos servidores y redes.

3.6.5 SQL SERVER



3.12 SQL Sever

Microsoft SQL Server 7.0 constituye un lanzamiento determinante para los productos de bases de datos de Microsoft, continuando con la base sólida establecida por SQL Server 6.5. Como la mejor base de datos para Windows NT, SQL Server es el RDBMS de elección para una amplia gama de clientes corporativos y Proveedores Independientes de Software (ISVs) que construyen aplicaciones de negocios. Las necesidades y requerimientos de los clientes han llevado a la creación de innovaciones de producto significativas para facilitar la utilización, escalabilidad, confiabilidad y almacenamiento de datos.

Objetivos del Diseño de SQL Server

Los clientes están buscando soluciones para sus problemas de negocios. La mayoría de las "soluciones" de bases de datos solamente traen múltiples niveles de costos y complejidad. La estrategia de Microsoft es la de hacer que SQL Server sea la base de datos más fácil de utilizar para construir, administrar e implementar aplicaciones de negocios. Esto significa tener que poner a disposición un modelo de programación rápido y sencillo para desarrolladores, eliminando la administración de base de datos para operaciones estándar, y suministrando herramientas sofisticadas para operaciones más complejas.

SQL Server disminuye el costo total de propiedad a través de características como administración multi-servidor y con una sola consola; ejecución y alerta de trabajos basadas en eventos; seguridad integrada; y scripting administrativo. Esta versión también libera al administrador de base de datos para aspectos más sofisticados del trabajo al automatizar las tareas de rutina. Al combinar estos poderosos servicios de administración con las nuevas características de configuración automática, Microsoft SQL Server es la elección ideal de automatización de sucursales y aplicaciones de base de datos insertadas.

Los clientes invierten en sistemas de administración de bases de datos, en forma de aplicaciones escritas para esa base de datos y la educación que implica para la implementación y administración

Características

Microsoft SQL Server revoluciona el concepto de Base de Datos para la Empresa. Reúne en un sólo producto la potencia necesaria para cualquier aplicación empresarial crítica junto con unas herramientas de gestión que reducen al mínimo el coste de propiedad. Con Microsoft SQL Server, la empresa tiene todo de serie.

Miles de Soluciones Disponibles: Tendrá libertad de elección, ya que todas las aplicaciones de gestión del mercado corren sobre Microsoft SQL Server

- Escalabilidad: Se adapta a las necesidades de la empresa, soportando desde unos pocos usuarios a varios miles. Empresas centralizadas u oficinas distribuidas, replicando cientos de sites.
- Potencia: Microsoft SQL Server es la mejor base de datos para Windows NT Server. Posee los mejores registros de los benchmarks independientes (TCP) tanto en transacciones totales como en coste por transacción.
- Gestión: Con un completo interfaz gráfico que reduce la complejidad innecesaria de las tareas de administración y gestión de la base de datos.
- Orientada al desarrollo: Visual Basic, Visual C++, Visual J++, Visual Interdev, Microfocus Cobol y muchas otras herramientas son compatibles con Microsoft SQL Server.

La mejor base de datos para Internet, Intranet y Extranet

- Diseñada desde su inicio para trabajar en entornos Internet e Intranet, Microsoft SQL Server es capaz de integrar los nuevos desarrollos para estos entornos específicos con los desarrollos heredados de aplicaciones "tradicionales". Es más, cada aplicación que desarrollemos para ser empleada en entornos de red local puede ser utilizada de forma transparente -en parte o en su totalidad- desde entornos Internet, Intranet o Extranet.
- Plataforma de desarrollo fácil y abierta: Integrada con las mejores tecnologías de Internet como ActiveX, ADC y Microsoft Transaction Server

y con las mejores herramientas de gestión y desarrollo para Internet como FrontPage97, Microsoft Office97 y Visual Interdev.

- Diseñada para INTERNET: Es el único gestor de base de datos que contiene de forma integrada la posibilidad de generar contenido HTML de forma automática.
- La Base de Soluciones Integradas: La Integración total con BackOffice permite resolver toda las necesidades de infraestructura de la empresa con un sólo paquete.
- Potente y Escalable: Microsoft SQL Server es la única base de datos cuyo rendimiento sobre Internet está publicado, ofreciendo registros espectaculares.
- Mínimo coste de Propiedad: La sencillez de la instalación, y la potencia de sus herramientas de gestión y el menor coste de toda la industria para entornos Internet, hacen de Microsoft SQL Server la mejor opción con el menor coste.

Arquitectura RDBMS

- Arquitectura de servidor simétrico y paralelo con balanceo automático de carga en múltiples procesadores.
- Kernel multithread real para mejor rendimiento transaccional y escalabilidad.
- Soporte grandes bases de datos (VLDB) (+1 TB).
- Completo proceso transaccional interactivo con rollback automático y recuperación de roll-forward.
- Optimizador de consultas mejorado basado en coste.
- Checkpointing mejorado para un mejor throughput de datos y tiempo de respuesta.
- Soporte E/S asíncrono para acceso en paralelo a múltiples dispositivos de disco para un mejor throughput.
- Bloqueo a nivel fija y página con escalación de bloqueos; resolución automática de deadlocks.

Datos distribuidos y replicación

- Llamadas a procedimientos remotos servidor-a-servidor (procedimientos almacenados remotos).
- Replicación asíncrona o continua basada en registros, o sincronización planificada de tablas point-in-time.
- Configuración de replicación gráfica y características de gestión.
- Replicación de subscritores ODBC, incluyendo IBM DB2, ORACLE, SYBASE y Microsoft Access.
- El Distributed Transaction Coordinator gestiona transacciones que involucran a dos o más servidores SQL (proceso Two Phase Commit 2PC) transparente.
- Replicación de tipos de datos Texto e Imagen.

Data Warehousing y amplio soporte de datos

- Ejecución transaccional en paralelo, carga e indexación de datos , y verificación de integridad.
- Backup/restore en paralelo de alta velocidad que soporta hasta 32 discos o dispositivos de cinta.
- Las extensiones de consulta OLAP, CUBE, y ROLLUP permiten conjuntos de resultados multidimensionales.
- EXEC e INTO ahora permiten almacenar en tablas resultados desde procedimientos almacenados.
- Copias de seguridad en cinta y recuperación de tablas individuales.
- Reconstrucción de índice sin realizar dropping y recreación del índice.

Integración Internet y correo electrónico

- MAPI, permitiendo aplicaciones de flujo de trabajo y notificación de cambio de datos automática.
- Compatibilidad con Microsoft Internet Information Server y otros servidores Web populares.
- SQL Web Assistant, para el retorno automático de datos en formato HTML.
- Procedimientos almacenados para generar páginas HTML o actualizar datos en plantillas Web.
- Posibilidad de poblar automáticamente carpetas públicas de Microsoft Exchange con datos.

Gestión y administración centralizada de bases de datos

- SQL Enterprise Manager, una consola de gestión y motorización 32-bit visual basada en Windows.
- Un único punto de configuración y gestión de control de datos remotas.
- SQL Executive, planificador de trabajos y monitor para gestión proactiva de servidores distribuidos.
- Operaciones remotas desatendidas mediante un "agente inteligente" que incluye evento-alerta- respuesta.
- Scripts Visual Basic a través de SQL-Distributed Management Objects (SQL-DMO) basados en OLE.
- DBA Assistant, para el mantenimiento automático rutinario en una única tarea planificada.
- SQL Trace, para monitorizar consultas cliente-servidor mediante SQL almacenadas en archivos de registros.
- Soporte de MIB y traps SNMP para monitorizar SQL Server desde herramientas de gestión basadas en SNMP.

Disponibilidad, fiabilidad y tolerancia a fallos

- Mirroring de dispositivos de base de datos con failover automático para tolerancia a fallos de dispositivos.
- Copias de seguridad online desatendidas garantizando la consistencia de datos para la más alta disponibilidad.
- Contextos de usuario protegidos, que pueden aislar los fallos a un thread de un único usuario.
- Recuperación point-in-time para restaurar bases de datos o transacción logs en un intervalo de tiempo.
- Tolerancia a fallos de servidor, permitiendo failover automático a un servidor de backup o en espera.

Mejoras en programabilidad y lenguaje

- Triggers, procedimientos almacenados (autoexec), disparador de eventos antes y después de conexiones.
- Procedimientos almacenados extendidos (funciones definidas por el usuario) utilizando C/C++.
- Cursores basados en el motor con scrolling hacia adelante y atrás; posicionamiento absoluto y relativo.
- Sentencias DLL permitidas dentro de transacciones.
- Transacciones distribuidas dentro de interfaces DB-Library, ODBC, Transact-SQL, XA y OLE Transaction.
- Procedimientos almacenados OLE Automation.

Seguridad

- Un único ID de login tanto para red como para la DB para mejorar la seguridad y facilitar la administración.
- Password y encriptación de datos en red para mejorar la seguridad.
- Encriptación de procedimientos almacenados para la integridad y seguridad de código de aplicación.
- Interoperabilidad e integración con desktops.
- API estándar DB-Library totalmente soportada: estándar ODBC Nivel 2 totalmente soportado como API nativa.
- Gateway Open Data Services (ODS) programable para acceso transparente a fuentes de datos externas.
- Gateways de Microsoft y de terceros para fuentes de datos relacionales y no-relacionales, incluyendo IBM DB2.
- Soporte de importantes estándares de mercado como ANSI SQL-92, FIPS 127-2, XA, SNMP.

3.7 SERVIDORES

En informática, un servidor es un tipo de software que realiza ciertas tareas en nombre de los usuarios. El término servidor ahora también se utiliza para referirse al ordenador físico en el cual funciona ese software, una máquina cuyo propósito es proveer datos de modo que otras máquinas puedan utilizar esos datos.

Este uso dual puede llevar a confusión. Por ejemplo, en el caso de un servidor web, este término podría referirse a la máquina que almacena y maneja los sitios web, y en este sentido es utilizada por las compañías que ofrecen hosting o hospedaje. Alternativamente, el servidor web podría referirse al software, como el servidor de http de Apache, que funciona en la máquina y maneja la entrega de los componentes de las páginas web como respuesta a peticiones de los navegadores de los clientes.

Los archivos para cada sitio de Internet se almacenan y se ejecutan en el servidor. Hay muchos servidores en Internet y muchos tipos de servidores, pero comparten la función común de proporcionar el acceso a los archivos y servicios.

Un servidor sirve información a los equipos que se conecten a él. Cuando los usuarios se conectan a un servidor pueden acceder a programas, archivos y otra información del servidor.

En la web, un servidor web es un cliente que realiza una solicitud mediante el protocolo http para enviar páginas web a la computadora de un usuario cuando el usuario las solicita.

Los servidores web, servidores de correo y servidores de bases de datos son a lo que tiene acceso la mayoría de la gente al usar Internet.

Algunos servidores manejan solamente correo o solamente archivos, mientras que otros hacen más de un trabajo, ya que un mismo ordenador puede tener diferentes programas de servidor funcionando al mismo tiempo.

Los servidores se conectan a la red mediante una interfaz que puede ser una red verdadera o mediante conexión vía línea telefónica o digital.

3.7.2 TIPOS DE SERVIDORES Y SUS APLICACIONES

Esta lista categoriza los diversos tipos de servidores del mercado actual:

Plataformas de Servidor (*Server Platforms*): Un término usado a menudo como sinónimo de sistema operativo, la plataforma es el hardware o software subyacentes para un sistema, es decir, el motor que dirige el servidor.

Servidores de Aplicaciones (*Application Servers*): Designados a veces como un tipo de *middleware* (software que conecta dos aplicaciones), los servidores de aplicaciones ocupan una gran parte del territorio entre los servidores de bases de datos y el usuario, y a menudo los conectan.

Servidores de Audio/Video (*Audio/Video Servers*): Los servidores de Audio/Video añaden capacidades multimedia a los sitios web permitiéndoles mostrar contenido multimedia en forma de flujo continuo (*streaming*) desde el servidor.

Servidores de Chat (*Chat Servers*): Los servidores de chat permiten intercambiar información a una gran cantidad de usuarios ofreciendo la posibilidad de llevar a cabo discusiones en tiempo real.

Servidores de Fax (*Fax Servers*): Un servidor de fax es una solución ideal para organizaciones que tratan de reducir el uso del teléfono pero necesitan enviar documentos por fax.

Servidores FTP (*FTP Servers*): Uno de los servicios más antiguos de Internet, File Transfer Protocol permite mover uno o más archivos

Servidores Groupware (*Groupware Servers*): Un servidor groupware es un software diseñado para permitir colaborar a los usuarios, sin importar la localización, vía Internet o vía Intranet corporativo y trabajar juntos en una atmósfera virtual.

Servidores IRC (*IRC Servers*): Otra opción para usuarios que buscan la discusión en tiempo real, Internet Relay Chat consiste en varias redes de servidores separadas que permiten que los usuarios conecten el uno al otro vía una red IRC.

Servidores de Listas (*List Servers*): Los servidores de listas ofrecen una manera mejor de manejar listas de correo electrónico, bien sean discusiones interactivas abiertas al público o listas unidireccionales de anuncios, boletines de noticias o publicidad.

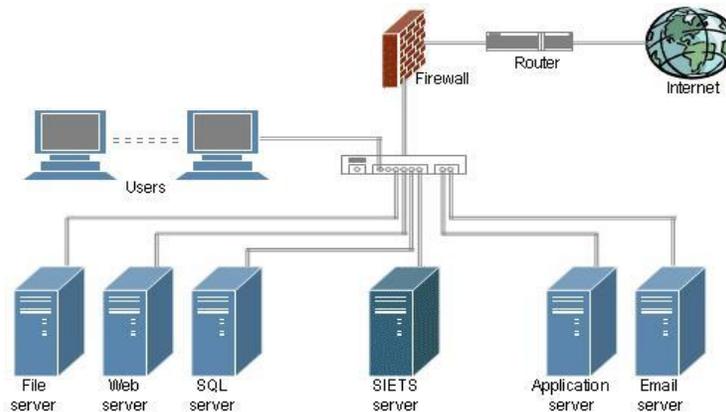
Servidores de Correo (*Mail Servers*): Casi tan ubicuos y cruciales como los servidores web, los servidores de correo mueven y almacenan el correo electrónico a través de las redes corporativas (vía LANs y WANs) y a través de Internet.

Servidores de Noticias (*News Servers*): Los servidores de noticias actúan como fuente de distribución y entrega para los millares de grupos de noticias públicos actualmente accesibles a través de la red de noticias USENET.

Servidores Proxy (*Proxy Servers*): Los servidores proxy se sitúan entre un programa del cliente (típicamente un navegador) y un servidor externo (típicamente otro servidor web) para filtrar peticiones, mejorar el funcionamiento y compartir conexiones.

Servidores Telnet (*Telnet Servers*): Un servidor telnet permite a los usuarios entrar en un ordenador huésped y realizar tareas como si estuviera trabajando directamente en ese ordenador.

Servidores Web (*Web Servers*): Básicamente, un servidor web sirve contenido estático a un navegador, carga un archivo y lo sirve a través de la red al navegador de un usuario. Este intercambio es mediado por el navegador y el servidor que hablan el uno con el otro mediante HTTP. Se pueden utilizar varias tecnologías en el servidor para aumentar su potencia más allá de su capacidad de entregar páginas HTML; éstas incluyen scripts CGI, seguridad SSL y páginas activas del servidor (ASP).



3.13 Tipos de Servidores

3.8 SEGURIDAD

La seguridad informática es el área de la informática⁵ que se enfoca en la protección de la infraestructura computacional y todo lo relacionado con esta (incluyendo la información contenida). Para ello existen una serie de estándares,

protocolos, métodos, reglas, herramientas y leyes concebidas para minimizar los posibles riesgos a la infraestructura o a la información. La seguridad informática comprende software, bases de datos, metadatos, archivos y todo lo que la organización valore (activo) y signifique un riesgo si ésta llega a manos de otras personas. Este tipo de información se conoce como información privilegiada o confidencial.

El concepto de seguridad de la información no debe ser confundido con el de seguridad informática, ya que este último sólo se encarga de la seguridad en el medio informático, pudiendo encontrar información en diferentes medios o formas.

3.8.1 CONCEPTOS

La seguridad informática está concebida para proteger los activos informáticos, entre los que se encuentran:

- **La información contenida**

Se ha convertido en uno de los elementos más importantes dentro de una organización. La seguridad informática debe ser administrada según los criterios establecidos por los administradores y supervisores, evitando que usuarios externos y no autorizados puedan acceder a ella sin autorización. De lo contrario la organización corre el riesgo de que la información sea utilizada maliciosamente para obtener ventajas de ella o que sea manipulada, ocasionando lecturas erradas o incompletas de la misma. Otra función de la seguridad informática en esta área es la de asegurar el acceso a la información en el momento oportuno, incluyendo respaldos de la misma en caso de que esta sufra daños o pérdida producto de accidentes, atentados o desastres.

- **La infraestructura computacional**

Una parte fundamental para el almacenamiento y gestión de la información, así como para el funcionamiento mismo de la organización. La función de la seguridad informática en esta área es velar que los equipos funcionen adecuadamente y prever en caso de falla planes de robos, incendios, boicot, desastres naturales, fallas en el suministro eléctrico y cualquier otro factor que atente contra la infraestructura informática.

- **Los usuarios**

Son las personas que utilizan la estructura tecnológica, zona de comunicaciones y que gestionan la información. La seguridad informática debe establecer normas

que minimicen los riesgos a la información o infraestructura informática. Estas normas incluyen horarios de funcionamiento, restricciones a ciertos lugares, autorizaciones, denegaciones, perfiles de usuario, planes de emergencia, protocolos y todo lo necesario que permita un buen nivel de seguridad informática minimizando el impacto en el desempeño de los funcionarios y de la organización en general y como principal contribuyente al uso de programas realizados por programadores.

- **Las amenazas**

Una vez que la programación y el funcionamiento de un dispositivo de almacenamiento (o transmisión) de la información se consideran seguras, todavía deben ser tenidas en cuenta las circunstancias "no informáticas" que pueden afectar a los datos, las cuales son a menudo imprevisibles o inevitables, de modo que la única protección posible es la redundancia (en el caso de los datos) y la descentralización -por ejemplo mediante estructura de redes- (en el caso de las comunicaciones).

- **Estos fenómenos pueden ser causados por:**

- El usuario: Causa del mayor problema ligado a la seguridad de un sistema informático (porque no le importa, no se da cuenta o a propósito).
- Programas maliciosos: Programas destinados a perjudicar o a hacer un uso ilícito de los recursos del sistema. Es instalado (por inatención o maldad) en el ordenador abriendo una puerta a intrusos o bien modificando los datos. Estos programas pueden ser un virus informático, un gusano informático, un troyano, una bomba lógica o un programa espía o Spyware.
- Un intruso: Persona que consigue acceder a los datos o programas de los cuales no tiene acceso permitido (cracker, defacer, script kiddie o *Script boy, viruxer, etc.*).
- Un siniestro (robo, incendio, inundación): Una mala manipulación o una malintención derivan a la pérdida del material o de los archivos.
- El personal interno de Sistemas: Las pujas de poder que llevan a disociaciones entre los sectores y soluciones incompatibles para la seguridad informática.

- **Tipos de amenaza**

El hecho de conectar una red a un entorno externo nos da la posibilidad de que algún atacante pueda entrar en ella, con esto, se puede hacer robo de información o alterar el funcionamiento de la red. Sin embargo el hecho de que la red no sea conectada a un entorno externo no nos

garantiza la seguridad de la misma. De acuerdo con el Computer Security Institute (CSI) de San Francisco aproximadamente entre 60 y 80 por ciento de los incidentes de red son causados desde adentro de la misma. Basado en esto podemos decir que existen 2 tipos de amenazas:

- **Amenazas internas:** Generalmente estas amenazas pueden ser más serias que las externas por varias razones como son:
 - ✓ Los usuarios conocen la red y saben cómo es su funcionamiento.
 - ✓ Tienen algún nivel de acceso a la red por las mismas necesidades de su trabajo.
 - ✓ Los IPS y Firewalls son mecanismos no efectivos en amenazas internas.

Esta situación se presenta gracias a los esquemas ineficientes de seguridad con los que cuentan la mayoría de las compañías a nivel mundial, y porque no existe conocimiento relacionado con la planeación de un esquema de seguridad eficiente que proteja los recursos informáticos de las actuales amenazas combinadas.

El resultado es la violación de los sistemas, provocando la pérdida o modificación de los datos sensibles de la organización, lo que puede representar un daño con valor de miles o millones de dólares.

- **Amenazas externas:** Son aquellas amenazas que se originan de afuera de la red. Al no tener información certera de la red, un atacante tiene que realizar ciertos pasos para poder conocer qué es lo que hay en ella y buscar la manera de atacarla. La ventaja que se tiene en este caso es que el administrador de la red puede prevenir una buena parte de los ataques externos.

La amenaza informática del futuro

Si en un momento el objetivo de los ataques fue cambiar las plataformas tecnológicas ahora las tendencias cibercriminales indican que la nueva modalidad es manipular los significados de la información digital. El área semántica, era reservada para los humanos, se convirtió ahora en el núcleo de los ataques debido a la evolución de la Web 2.0 y las redes sociales, factores que llevaron al nacimiento de la generación 3.0.

- Se puede afirmar que “la Web 3.0 otorga contenidos y significados de manera tal que pueden ser comprendidos por las computadoras, las cuales por medio de técnicas de inteligencia artificial- son capaces de emular y

mejorar la obtención de conocimiento, hasta el momento reservada a las personas”.

- Es decir, se trata de dotar de significado a las páginas Web, y de ahí el nombre de Web semántica o Sociedad del Conocimiento, como evolución de la ya pasada Sociedad de la Información

En este sentido, las amenazas informáticas que viene en el futuro ya no son con la inclusión de troyanos en los sistemas o softwares espías, sino con el hecho de que los ataques se han profesionalizado y manipulan el significado del contenido virtual.

- “La Web 3.0, basada en conceptos como elaborar, compartir y significar, está representando un desafío para los hackers que ya no utilizan las plataformas convencionales de ataque, sino que optan por modificar los significados del contenido digital, provocando así la confusión lógica del usuario y permitiendo de este modo la intrusión en los sistemas”, La amenaza ya no solicita la clave de homebanking del desprevenido usuario, sino que directamente modifica el balance de la cuenta, asustando al internauta y, a partir de allí, sí efectuar el robo del capital”.

Para no ser presa de esta nueva ola de ataques más sutiles, Se recomienda:

- Mantener las soluciones activadas y actualizadas.
- Evitar realizar operaciones comerciales en computadoras de uso público.
- Verificar los archivos adjuntos de mensajes sospechosos y evitar su descarga en caso de duda.

Existe un viejo dicho en la seguridad informática que dicta: *“lo que no está permitido debe estar prohibido”* y ésta debe ser la meta perseguida.

Los medios para conseguirlo son:

1. Restringir el acceso (de personas de la organización y de las que no lo son) a los programas y archivos.
2. Asegurar que los operadores puedan trabajar pero que no puedan modificar los programas ni los archivos que no correspondan (sin una supervisión minuciosa).
3. Asegurar que se utilicen los datos, archivos y programas correctos en/y/por el procedimiento elegido.
4. Asegurar que la información transmitida sea la misma que reciba el destinatario al cual se ha enviado y que no le llegue a otro.
5. Asegurar que existan sistemas y pasos de emergencia alternativos de transmisión entre diferentes puntos.

6. Organizar a cada uno de los empleados por jerarquía informática, con claves distintas y permisos bien establecidos, en todos y cada uno de los sistemas o aplicaciones empleadas.
7. Actualizar constantemente las contraseñas de accesos a los sistemas de cómputo.

Elementos de un análisis de riesgo

Cuando se pretende diseñar una técnica para implementar un análisis de riesgo informático se pueden tomar los siguientes puntos como referencia a seguir:

- Planes para reducir los riesgos.

Análisis de impacto al negocio

El reto es asignar estratégicamente los recursos para equipo de seguridad y bienes que intervengan, basándose en el impacto potencial para el negocio, respecto a los diversos incidentes que se deben resolver. Para determinar el establecimiento de prioridades, el sistema de gestión de incidentes necesita saber el valor de los sistemas de información que pueden ser potencialmente afectados por incidentes de seguridad. Esto puede implicar que alguien dentro de la organización asigne un valor monetario a cada equipo y un archivo en la red o asignar un valor relativo a cada sistema y la información sobre ella. Dentro de los valores para el sistema se pueden distinguir: Confidencialidad de la información, la Integridad (aplicaciones e información) y finalmente la Disponibilidad del sistema. Cada uno de estos valores es un sistema independiente del negocio, supongamos el siguiente ejemplo, un servidor Web público pueden poseer los requisitos de confidencialidad de baja (ya que toda la información es pública), pero de alta disponibilidad y los requisitos de integridad. En contraste, un sistema de planificación de recursos empresariales (ERP), sistema puede poseer alta puntaje en los tres variables. Los incidentes individuales pueden variar ampliamente en términos de alcance e importancia.

3.8.2 POLÍTICAS

Actualmente las legislaciones nacionales de los Estados, obligan a las empresas, instituciones públicas a implantar una política de seguridad. Ej: En España la Ley Orgánica de Protección de Datos o también llamada LOPD y su normativa de desarrollo.

Generalmente se ocupa exclusivamente a asegurar los derechos de acceso a los datos y recursos con las herramientas de control y mecanismos de identificación.

Estos mecanismos permiten saber que los operadores tienen sólo los permisos que se les dio.

La seguridad informática debe ser estudiada para que no impida el trabajo de los operadores en lo que les es necesario y que puedan utilizar el sistema informático con toda confianza. Por eso en lo referente a elaborar una política de seguridad, conviene:

- Elaborar reglas y procedimientos para cada servicio de la organización.
- Definir las acciones a emprender y elegir las personas a contactar en caso de detectar una posible intrusión
- Sensibilizar a los operadores con los problemas ligados con la seguridad de los sistemas informáticos.

Los derechos de acceso de los operadores deben ser definidos por los responsables jerárquicos y no por los administradores informáticos, los cuales tienen que conseguir que los recursos y derechos de acceso sean coherentes con la política de seguridad definida. Además, como el administrador suele ser el único en conocer perfectamente el sistema, tiene que derivar a la directiva cualquier problema e información relevante sobre la seguridad, y eventualmente aconsejar estrategias a poner en marcha, así como ser el punto de entrada de la comunicación a los trabajadores sobre problemas y recomendaciones en término de seguridad informática.

3.8.3 HERRAMIENTAS DE SEGURIDAD Y PROTECCIÓN

3.8.3.1 FIREWALL

Un firewall es una parte de un sistema o una red que está diseñada para bloquear el acceso no autorizado, permitiendo al mismo tiempo comunicaciones autorizadas.

Se trata de un dispositivo o conjunto de dispositivos configurados para permitir, limitar, cifrar, descifrar, el tráfico entre los diferentes ámbitos sobre la base de un conjunto de normas y otros criterios.

Los Firewall pueden ser implementados en hardware o software, o una combinación de ambos. Los cortafuegos se utilizan con frecuencia para evitar que los usuarios de Internet no autorizados tengan acceso a redes privadas conectadas a Internet, especialmente intranets. Todos los mensajes que entren o salgan de la intranet pasan a través del firewall, que examina cada mensaje y

bloquea aquellos que no cumplen los criterios de seguridad especificados. También es frecuente conectar al firewall a una tercera red, llamada Zona desmilitarizada o DMZ, en la que se ubican los servidores de la organización que deben permanecer accesibles desde la red exterior.

Un Firewall correctamente configurado añade una protección necesaria a la red, pero que en ningún caso debe considerarse suficiente. La seguridad informática abarca más ámbitos y más niveles de trabajo y protección.

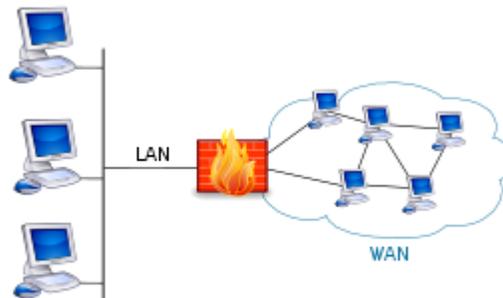


Figura 3.14 Esquema de una red de computadoras que utiliza un firewall

Historia de los cortafuegos

El término "Firewall / Fireblock" significaba originalmente una pared para confinar un incendio o riesgo potencial de incendio en un edificio. Más adelante se usa para referirse a las estructuras similares, como la hoja de metal que separa el compartimiento del motor de un vehículo o una aeronave de la cabina. La tecnología de los cortafuegos surgió a finales de 1980, cuando Internet era una tecnología bastante nueva en cuanto a su uso global y la conectividad. Los predecesores de los cortafuegos para la seguridad de la red fueron los routers utilizados a finales de 1980, que mantenían a las redes separadas unas de otras. La visión de Internet como una comunidad relativamente pequeña de usuarios con máquinas compatibles, que valoraba la predisposición para el intercambio y la colaboración, terminó con una serie de importantes violaciones de seguridad de Internet que se produjo a finales de los 80:

- Clifford Stoll, que descubrió la forma de manipular el sistema de espionaje alemán.
- Bill Cheswick, cuando en 1992 instaló una cárcel simple electrónica para observar a un atacante.

- En 1988, un empleado del Centro de Investigación Ames de la NASA, en California, envió una nota por correo electrónico a sus colegas que decía:

"Estamos bajo el ataque de un virus de Internet! Ha llegado a Berkeley, UC San Diego, Lawrence Livermore, Stanford y la NASA Ames."

- El Gusano Morris, que se extendió a través de múltiples vulnerabilidades en las máquinas de la época. Aunque no era malicioso, el gusano Morris fue el primer ataque a gran escala sobre la seguridad en Internet; la red no esperaba ni estaba preparada para hacer frente a su ataque.

Primera generación – Firewall de red: filtrado de paquetes

El primer documento publicado para la tecnología firewall data de 1988, cuando el equipo de ingenieros Digital Equipment Corporation (DEC) desarrolló los sistemas de filtro conocidos como cortafuegos de filtrado de paquetes. Este sistema, bastante básico, fue la primera generación de lo que se convertiría en una característica más técnica y evolucionada de la seguridad de Internet. En AT&T Bell, Bill Cheswick y Steve Bellovin, continuaban sus investigaciones en el filtrado de paquetes y desarrollaron un modelo de trabajo para su propia empresa, con base en su arquitectura original de la primera generación.

El filtrado de paquetes actúa mediante la inspección de los paquetes (que representan la unidad básica de transferencia de datos entre computadoras en Internet). Si un paquete coincide con el conjunto de reglas del filtro, el paquete se reducirá (descarte silencioso) o será rechazado (desprendiéndose de él y enviando una respuesta de error al emisor). Este tipo de filtrado de paquetes no presta atención a si el paquete es parte de una secuencia existente de tráfico. En su lugar, se filtra cada paquete basándose únicamente en la información contenida en el paquete en sí (por lo general utiliza una combinación del emisor del paquete y la dirección de destino, su protocolo, y, en el tráfico TCP y UDP, el número de puerto). Los protocolos TCP y UDP comprenden la mayor parte de comunicación a través de Internet, utilizando por convención puertos bien conocidos para determinados tipos de tráfico, por lo que un filtro de paquetes puede distinguir entre ambos tipos de tráfico (ya sean navegación web, impresión remota, envío y recepción de correo electrónico, transferencia de archivos...); a menos que las máquinas a cada lado del filtro de paquetes son a la vez utilizando los mismos puertos no estándar.

El filtrado de paquetes llevado a cabo por un firewall actúa en las tres primeras capas del modelo de referencia OSI, lo que significa que todo el trabajo lo realiza

entre la red y las capas físicas. Cuando el emisor origina un paquete y es filtrado por el cortafuegos, éste último comprueba las reglas de filtrado de paquetes que lleva configuradas, aceptando o rechazando el paquete en consecuencia. Cuando el paquete pasa a través de cortafuegos, éste filtra el paquete mediante un protocolo y un número de puerto base (GSS). Por ejemplo, si existe una norma en el firewall para bloquear el acceso telnet, bloqueará el protocolo IP para el número de puerto 23.

Segunda generación – Firewall de estado

Durante 1989 y 1990, tres colegas de los laboratorios AT&T Bell, Dave Presetto, Janardan Sharma, y Nigam Kshitiij, desarrollaron la tercera generación de servidores de seguridad. Esta tercera generación firewall tiene en cuenta además la colocación de cada paquete individual dentro de una serie de paquetes. Esta tecnología se conoce generalmente como la inspección de estado de paquetes, ya que mantiene registros de todas las conexiones que pasan por el firewall, siendo capaz de determinar si un paquete indica el inicio de una nueva conexión, es parte de una conexión existente, o es un paquete erróneo. Este tipo de firewall pueden ayudar a prevenir ataques contra conexiones en curso o ciertos ataques de denegación de servicio.

Tercera generación - Firewall de aplicación

Son aquellos que actúan sobre la capa de aplicación del modelo OSI. La clave de un cortafuegos de aplicación es que puede entender ciertas aplicaciones y protocolos (por ejemplo: protocolo de transferencia de archivos, DNS o navegación web), y permite detectar si un protocolo no deseado se coló a través de un puerto no estándar o si se está abusando de un protocolo de forma perjudicial.

Un cortafuegos de aplicación es mucho más seguro y fiable cuando se compara con un cortafuegos de filtrado de paquetes, ya que repercute en las siete capas del modelo de referencia OSI. En esencia es similar a un cortafuegos de filtrado de paquetes, con la diferencia de que también podemos filtrar el contenido del paquete. El mejor ejemplo de cortafuegos de aplicación es ISA (Internet Security and Acceleration).

Un firewall de aplicación puede filtrar protocolos de capas superiores tales como FTP, TELNET, DNS, DHCP, HTTP, TCP, UDP y TFTP (GSS). Por ejemplo, si una organización quiere bloquear toda la información relacionada con una palabra en concreto, puede habilitarse el filtrado de contenido para bloquear esa palabra en

particular. No obstante, los cortafuegos de aplicación resultan más lentos que los de estado.

Acontecimientos posteriores

En 1992, Bob Braden y DeSchon Annette, de la Universidad del Sur de California (USC), dan forma al concepto de cortafuegos. Su producto, conocido como "Visas", fue el primer sistema con una interfaz gráfica con colores e iconos, fácilmente implementable y compatible con sistemas operativos como Windows de Microsoft o MacOS de Apple. En 1994, una compañía israelí llamada Check Point Software Technologies lo patentó como software denominándolo FireWall-1.

La funcionalidad existente de inspección profunda de paquetes en los actuales cortafuegos puede ser compartida por los sistemas de prevención de intrusiones (IPS).

Actualmente, el Grupo de Trabajo de Comunicación Middlebox de la Internet Engineering Task Force (IETF) está trabajando en la estandarización de protocolos para la gestión de cortafuegos.

Otro de los ejes de desarrollo consiste en integrar la identidad de los usuarios dentro del conjunto de reglas del firewall. Algunos cortafuegos proporcionan características tales como unir a las identidades de usuario con las direcciones IP o MAC. Otros, como el firewall NuFW, proporcionan características de identificación real solicitando la firma del usuario para cada conexión.

Tipos de Firewall

Nivel de aplicación de pasarela

Aplica mecanismos de seguridad para aplicaciones específicas, tales como servidores FTP y Telnet. Esto es muy eficaz, pero puede imponer una degradación del rendimiento.

Circuito a nivel de pasarela

Aplica mecanismos de seguridad cuando una conexión TCP o UDP es establecida. Una vez que la conexión se ha hecho, los paquetes pueden fluir entre los anfitriones sin más control. Permite el establecimiento de una sesión que se origine desde una zona de mayor seguridad hacia una zona de menor seguridad.

Firewall de capa de red o de filtrado de paquetes

Funciona a nivel de red (capa 3 del modelo OSI, capa 2 del stack de protocolos TCP/IP) como filtro de paquetes IP. A este nivel se pueden realizar filtros según los distintos campos de los paquetes IP: dirección IP origen, dirección IP destino. A menudo en este tipo de firewall se permiten filtrados según campos de nivel de transporte (capa 3 TCP/IP, capa 4 Modelo OSI), como el puerto origen y destino, o a nivel de enlace de datos (no existe en TCP/IP, capa 2 Modelo OSI) como la dirección MAC.

Firewall de capa de aplicación

Trabaja en el nivel de aplicación (capa 7 del modelo OSI), de manera que los filtrados se pueden adaptar a características propias de los protocolos de este nivel. Por ejemplo, si se trata de tráfico HTTP, se pueden realizar filtrados según la URL a la que se está intentando acceder.

Un firewall a nivel 7 de tráfico HTTP suele denominarse proxy, y permite que los computadores de una organización entren a Internet de una forma controlada. Un proxy oculta de manera eficaz las verdaderas direcciones de red.

Firewall personal

Es un caso particular de firewall que se instala como software en una computadora, filtrando las comunicaciones entre dicho computador y el resto de la red. Se usa por tanto, a nivel personal.

Ventajas de un Firewall

Bloquea el acceso a personas no autorizadas a redes privadas.

Limitaciones de un Firewall

Las limitaciones se desprenden de la misma definición del firewall: filtro de tráfico. Cualquier tipo de ataque informático que use tráfico aceptado por el firewall (por usar puertos TCP abiertos expresamente, por ejemplo) o que sencillamente no use la red, seguirá constituyendo una amenaza. La siguiente lista muestra algunos de estos riesgos:

- No puede proteger contra aquellos ataques cuyo tráfico no pase a través de él.

- No puede proteger de las amenazas a las que está sometido por ataques internos o usuarios negligentes. El firewall no puede prohibir a espías corporativos copiar datos sensibles en medios físicos de almacenamiento (discos, memorias, etc.) y sustraerlas del edificio.
- No puede proteger contra los ataques de ingeniería social.
- No puede proteger contra los ataques posibles a la red interna por virus informáticos a través de archivos y software. La solución real está en que la organización debe ser consciente en instalar software antivirus en cada máquina para protegerse de los virus que llegan por cualquier medio de almacenamiento u otra fuente.
- No protege de los fallos de seguridad de los servicios y protocolos cuyo tráfico esté permitido. Hay que configurar correctamente y cuidar la seguridad de los servicios que se publiquen en Internet.

Políticas del Firewall

Hay dos políticas básicas en la configuración de un firewall que cambian radicalmente la filosofía fundamental de la seguridad en la organización:

- Política restrictiva: Se deniega todo el tráfico excepto el que está explícitamente permitido. El firewall obstruye todo el tráfico y hay que habilitar expresamente el tráfico de los servicios que se necesiten. Esta aproximación es la que suelen utilizar las empresas y organismos gubernamentales.
- Política permisiva: Se permite todo el tráfico excepto el que esté explícitamente denegado. Cada servicio potencialmente peligroso necesitará ser aislado básicamente caso por caso, mientras que el resto del tráfico no será filtrado. Esta aproximación la suelen utilizar universidades, centros de investigación y servicios públicos de acceso a internet.

La política restrictiva es la más segura, ya que es más difícil permitir por error tráfico potencialmente peligroso, mientras que en la política permisiva es posible que no se haya contemplado algún caso de tráfico peligroso y sea permitido por omisión.

3.8.3.2 ANTIVIRUS

En informática los antivirus son programas cuyo objetivo es detectar y/o eliminar virus informáticos. Nacieron durante la década de 1980.

Con el transcurso del tiempo, la aparición de sistemas operativos más avanzados e Internet, ha hecho que los antivirus hayan evolucionado hacia programas más

avanzados que no sólo buscan detectar virus informáticos, sino bloquearlos, desinfectarlos y prevenir una infección de los mismos, y actualmente ya son capaces de reconocer otros tipos de malware, como spyware, rootkits, etc.



Figura 3.15 Antivirus más utilizados

Métodos de contagio

Existen dos grandes grupos de propagación: los virus cuya instalación el usuario en un momento dado ejecuta o acepta de forma inadvertida, o los gusanos, con los que el programa malicioso actúa replicándose a través de las redes.

En cualquiera de los dos casos, el sistema operativo infectado comienza a sufrir una serie de comportamientos anómalos o no previstos. Dichos comportamientos son los que dan la traza del problema y tienen que permitir la recuperación del mismo.

Dentro de las contaminaciones más frecuentes por interacción del usuario están las siguientes:

- Mensajes que ejecutan automáticamente programas (como el programa de correo que abre directamente un archivo adjunto).
- Ingeniería social, mensajes como: «Ejecute este programa y gane un premio».
- Entrada de información en discos de otros usuarios infectados.

- Instalación de software que pueda contener uno o varios programas maliciosos.
- Unidades extraíbles de almacenamiento (USB).

Seguridad y métodos de protección

Existen numerosos medios para combatir el problema; sin embargo, a medida que nuevos programas y sistemas operativos se introducen en el mercado, más difícil es tener controlados a todos y más sencillo va a ser que a alguien se le ocurran nuevas formas de infectar sistemas.

Los métodos para contener o reducir los riesgos asociados a los virus pueden ser los denominados activos o pasivos.

Copias de seguridad (pasivo)

Mantener una política de copias de seguridad garantiza la recuperación de los datos y la respuesta cuando nada de lo anterior ha funcionado.

Así mismo las empresas deberían disponer de un plan y detalle de todo el software instalado para tener un plan de contingencia en caso de problemas.

Planificación

La planificación consiste en tener preparado un plan de contingencia en caso de que una emergencia de virus se produzca, así como disponer al personal de la formación adecuada para reducir al máximo las acciones que puedan presentar cualquier tipo de riesgo. Cada antivirus puede planear la defensa de una manera, es decir, un antivirus puede hacer un escaneado completo, rápido o de vulnerabilidad según elija el usuario.

Consideraciones de software

El software es otro de los elementos clave en la parte de planificación. Se debería tener en cuenta la siguiente lista de comprobaciones:

1. Tener el software imprescindible para el funcionamiento de la actividad, nunca menos pero tampoco más. Tener controlado al personal en cuanto a la instalación de software es una medida que va implícita. Así mismo tener controlado el software asegura la calidad de la procedencia del mismo (no debería permitirse software pirata o sin garantías). En todo caso un

inventario de software proporciona un método correcto de asegurar la reinstalación en caso de desastre.

2. Disponer del software de seguridad adecuado. Cada actividad, forma de trabajo y métodos de conexión a Internet requieren una medida diferente de aproximación al problema. En general, las soluciones domésticas, donde únicamente hay un equipo expuesto, no son las mismas que las soluciones empresariales.
3. Métodos de instalación rápidos. Para permitir la reinstalación rápida en caso de contingencia.
4. Asegurar licencias. Determinados softwares imponen métodos de instalación de una vez, que dificultan la reinstalación rápida de la red. Dichos programas no siempre tienen alternativas pero ha de buscarse con el fabricante métodos rápidos de instalación.
5. Buscar alternativas más seguras. Existe software que es famoso por la cantidad de agujeros de seguridad que introduce. Es imprescindible conocer si se puede encontrar una alternativa que proporcione iguales funcionalidades pero permitiendo una seguridad extra.

Consideraciones de la red

Disponer de una visión clara del funcionamiento de la red permite poner puntos de verificación, filtrado y detección ahí donde la incidencia es más claramente identificable. Sin perder de vista otros puntos de acción es conveniente:

1. Mantener al máximo el número de recursos de red en modo de sólo lectura. De esta forma se impide que computadoras infectadas los propaguen.
2. Centralizar los datos. De forma que detectores de virus en modo batch puedan trabajar durante la noche.
3. Realizar filtrados de firewall de red. Eliminar los programas que comparten datos, como pueden ser los *peer-to-peer* (P2P); Mantener esta política de forma rigurosa, y con el consentimiento de la gerencia.
4. Reducir los permisos de los usuarios al mínimo, de modo que sólo permitan el trabajo diario.
5. Controlar y monitorizar el acceso a Internet. Para poder detectar en fases de recuperación cómo se ha introducido el virus, y así determinar los pasos a seguir.

Sistemas operativos más atacados

Las plataformas más atacadas por virus informáticos son la línea de sistemas operativos Windows⁷ de Microsoft. Respecto a los sistemas derivados

de Unix como GNU/Linux, BSD, Solaris, Mac OS X, estos han corrido con mayor suerte debido en parte al sistema de permisos. No obstante en las plataformas derivadas de Unix han existido algunos intentos que más que presentarse como amenazas reales no han logrado el grado de daño que causa un virus en plataformas Windows.

Plataformas Unix, inmunes a los virus de Windows

Un virus informático sólo atacará la plataforma para la que fue desarrollado.

3.8.3.3 RESPALDOS

La información constituye el activo más importante de las empresas, pudiendo verse afectada por muchos factores tales como robos, incendios, fallas de disco, virus u otros. Desde el punto de vista de la empresa, uno de los problemas más importantes que debe resolver es la protección permanente de su información crítica.

La medida más eficiente para la protección de los datos es determinar una buena política de copias de seguridad o backups: Este debe incluir copias de seguridad completa (los datos son almacenados en su totalidad la primera vez) y copias de seguridad incrementales (sólo se copian los ficheros creados o modificados desde el último backup). Es vital para las empresas elaborar un plan de backup en función del volumen de información generada y la cantidad de equipos críticos.

Un buen sistema de respaldo debe contar con ciertas características indispensables:

- Continuo
El respaldo de datos debe ser completamente automático y continuo. Debe funcionar de forma transparente, sin intervenir en las tareas que se encuentra realizando el usuario.
- Seguro
Muchos softwares de respaldo incluyen cifrado de datos (128-448 bits), lo cual debe ser hecho localmente en el equipo antes del envío de la información.
- Remoto
Los datos deben quedar alojados en dependencias alejadas de la empresa.
- Mantención de versiones anteriores de los datos

Se debe contar con un sistema que permita la recuperación de versiones diarias, semanales y mensuales de los datos.

Hoy en día los sistemas de respaldo de información online (Servicio de backup remoto) están ganando terreno en las empresas y organismos gubernamentales. La mayoría de los sistemas modernos de respaldo de información online cuentan con las máximas medidas de seguridad y disponibilidad de datos. Estos sistemas permiten a las empresas crecer en volumen de información sin tener que estar preocupados de aumentar su dotación física de servidores y sistemas de almacenamiento.

3.8 NAVEGADORES

Un **navegador** o **navegador web**⁶ (del inglés, web browser) es una aplicación que opera a través de Internet, interpretando la información de archivos y sitios web para que podamos leerla, (ya se encuentre ésta alojada en un servidor dentro de la World Wide Web o en un servidor local).

El navegador interpreta el código, HTML generalmente, en el que está escrita la página web y lo presenta en pantalla permitiendo al usuario interactuar con su contenido y navegar hacia otros lugares de la red mediante enlaces o hipervínculos.

La funcionalidad básica de un navegador web es permitir la visualización de documentos de texto, posiblemente con recursos multimedia incrustados. Los documentos pueden estar ubicados en la computadora en donde está el usuario, pero también pueden estar en cualquier otro dispositivo que esté conectado a la computadora del usuario o a través de Internet, y que tenga los recursos necesarios para la transmisión de los documentos (un software servidor web).

Tales documentos, comúnmente denominados páginas web, poseen hipervínculos que enlazan una porción de texto o una imagen a otro documento, normalmente relacionado con el texto o la imagen.

El seguimiento de enlaces de una página a otra, ubicada en cualquier computadora conectada a la Internet, se llama navegación, de donde se origina el nombre navegador (aplicado tanto para el programa como para la persona que lo utiliza, a la cual también se le llama cibernauta). Por otro lado, hojeador es una traducción literal del original en inglés, browser, aunque su uso es minoritario.

Historia

El primer navegador, desarrollado en el CERN a finales de 1990 y principios de 1991 por Tim Berners-Lee, era bastante sofisticado y gráfico, pero sólo funcionaba en estaciones NeXT.

El navegador **Mosaic**, que funcionaba inicialmente en entornos UNIX sobre X11, fue el primero que se extendió debido a que pronto el NCSA preparó versiones para Windows y Macintosh. Sin embargo, poco más tarde entró en el mercado **Netscape Navigator** que rápidamente superó en capacidades y velocidad a Mosaic. Este navegador tuvo la ventaja de funcionar en casi todos los UNIX, así como en entornos Windows.



Figura 3.16 Navegadores

3.9.1 TIPOS DE NAVEGADORES

- **Internet Explorer**

(Anteriormente Spyglass Mosaic) fue la apuesta tardía de Microsoft para entrar en el mercado y consiguió desbancar al Netscape Navigator entre los usuarios de Windows, debido a la integración del navegador con el sistema operativo, llegando a poseer cerca del 95% de la cuota de mercado. Netscape Communications Corporation liberó el código fuente de su navegador, naciendo así el proyecto **Mozilla**.

- **Mozilla Firefox**

Fue reescrito desde cero tras decidirse a desarrollar y usar como base un nuevo conjunto de widgets multiplataforma basado en XML llamado XUL y esto hizo que tardara bastante más en aparecer de lo previsto inicialmente, apareciendo una versión 1.0 de gran calidad y para muchísimas plataformas a la vez el 5 de junio del 2002.

A finales de 2004 aparece en el mercado **Firefox**, una rama de desarrollo de Mozilla que pretende hacerse con parte del mercado de Internet Explorer. Se trata de un navegador más ligero que su hermano mayor.

- **Google Chrome**

El 2 de septiembre vio la luz. El navegador web desarrollado por Google y compilado con base en componentes de código abierto como el motor de renderizado de WebKit y su estructura de desarrollo de aplicaciones (Framework). Google Chrome es el tercer navegador más utilizado en Internet y actualmente posee una cuota de mercado del 7,05%.¹ Está disponible gratuitamente bajo condiciones de servicio específicas. El nombre del navegador deriva del término usado para el marco de la interfaz gráfica de usuario ("chrome").

- **Safari**

Es un navegador web de código cerrado desarrollado por Apple Inc. Está disponible para Mac OS X, iOS (el sistema usado por el iPhone, el iPod Touch y el iPad) y Microsoft Windows.

Incluye navegación por pestañas, corrector ortográfico, búsqueda progresiva, vista del historial en CoverFlow, administrador de descargas y un sistema de búsqueda integrado.

- **Opera**

Es un navegador web y suite de Internet creado por la empresa noruega Opera Software, capaz de realizar múltiples tareas como navegar por sitios web, gestionar correo electrónico, contactos, fuentes web, charlar vía IRC y funcionar como cliente BitTorrent. Opera ha sido pionero en incorporar muchas de las características que hoy en día se pueden ver en el resto de los navegadores web como el zoom y la búsqueda integrada y popularizar otras como las pestañas.

Sus características de seguridad incluyen protección integrada contra phishing y malware, cifrado de alta seguridad al navegar por sitios web seguros, y la capacidad de eliminar fácilmente datos privados, como por ejemplo cookies, caché y datos de inicios de sesión.

Opera funciona en una gran variedad de sistemas operativos, incluyendo Microsoft Windows, Mac OS X, GNU/Linux y FreeBSD. Algunas versiones de Opera están disponibles para dispositivos que utilizan sistemas operativos como Maemo, BlackBerry, Symbian, Windows Mobile, Android o iOS, así como aquellos que soportan Java ME.

Funcionamiento de los navegadores

La comunicación entre el servidor web⁸ y el navegador se realiza mediante el protocolo HTTP, aunque la mayoría de los navegadores soportan otros protocolos como FTP, Gopher, y HTTPS (una versión cifrada de HTTP basada en Secure Socket Layer o Capa de Conexión Segura (SSL)).

La función principal del navegador es descargar documentos HTML y mostrarlos en pantalla. En la actualidad, no solamente descargan este tipo de documentos sino que muestran con el documento sus imágenes, sonidos e incluso vídeos *streaming* en diferentes formatos y protocolos. Además, permiten almacenar la información en el disco o crear marcadores (*bookmarks*) de las páginas más visitadas.

Algunos de los navegadores web más populares se incluyen en lo que se denomina una Suite. Estas Suite disponen de varios programas integrados para leer noticias de Usenet y correo electrónico.

Los primeros navegadores web sólo soportaban una versión muy simple de HTML. El rápido desarrollo de los navegadores web propietarios condujo al desarrollo de dialectos no estándares de HTML y a problemas de interoperabilidad en la web.

Los más modernos (como Google Chrome, Amaya, Mozilla, Netscape, Opera e Internet Explorer 9.0) soportan los estándares HTML y XHTML (comenzando con HTML 4.01, los cuales deberían visualizarse de la misma manera en todos ellos).

Los estándares web son un conjunto de recomendaciones dadas por el World Wide Web consortium W3C) y otras organizaciones internacionales acerca de cómo crear e interpretar documentos basados en la web. Su objetivo es crear una web que trabaje mejor para todos, con sitios accesibles a más personas y que funcionen en cualquier dispositivo de acceso a Internet.

CAPÍTULO 4 ANÁLISIS Y DISEÑO DEL SISTEMA

4.1 DESARROLLO DEL SISTEMA

En base a las herramientas mencionadas en el capítulo anterior, para el desarrollo e implementación del mismo, se utilizará lo siguiente:

Metodología

Ciclo de vida en cascada, ya que es un modelo de desarrollo de fácil planeación, permite el regreso a etapas anteriores para hacer cambios.

Modelado de la DB

Como herramienta de diseño para la BD, se utilizó DER, se aplicó hasta la Tercera Regla de Normalización.

Normalización

Son una serie de reglas que permiten a los diseñadores de bases de datos desarrollar un esquema que minimice los problemas de lógica. Cada regla está basada en la que le antecede. La normalización se adoptó porque el viejo estilo de poner todos los datos en un solo lugar, como un archivo o una tabla de la base de datos, era ineficiente y conducía a errores de lógica cuando se trataban de manipular los datos.

Reglas	Descripción
Primera Forma Normal (1FN)	Incluye la eliminación de todos los grupos repetidos
Segunda Forma Normal (2FN)	Asegura que todas las columnas que no son llave sean completamente dependientes de la llave primaria (PK).
Tercera Forma Normal (3FN)	Elimina cualquier dependencia transitiva. Una dependencia transitiva en aquella en la cual las columnas que no son llave son dependientes de otras columnas que tampoco son llave.

Las primeras tres formas normales son suficientes para cubrir las necesidades de la mayoría de las bases de datos.

Manejador de la Base de Datos

El manejador que se utilizó fue MySQL que soporta gran portabilidad entre sistemas, aprovecha la potencia de sistemas multiprocesador, condición de open source de MySQL hace que la utilización sea gratuita y se puede modificar con total libertad.

Lenguaje de Desarrollo

Como entorno de desarrollo se utilizó PHP ya que es un lenguaje multiplataforma, completamente orientado al desarrollo de aplicaciones web dinámicas con acceso a información almacenada en una Base de Datos.

Como se menciona en el capítulo anterior el código fuente escrito en PHP es invisible al navegador y al cliente ya que es el servidor el que se encarga de ejecutar el código y enviar su resultado HTML al navegador. Esto hace que la programación en PHP sea segura y confiable, capacidad de conexión con la mayoría de los motores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL y PostgreSQL.

A continuación y en base a las herramientas ya mencionadas se comienza con la implementación del sistema.

4.2 DIMENSIONAMIENTO DE LA BASE DE DATOS

Recordando un poco los resultados esperados; El sistema busca ordenar, mantener y facilitar el manejo de la información, con la finalidad ofrecer un mejor servicio a quien lo requiera. Además de contar con la información organizada y estructurada de tal manera que siempre se encuentre disponible cada que se requiera.

En base a las necesidades por lo que se pretende implementar dicho sistema, hablare brevemente del análisis que se realizo.

Primero que nada el sistema debe de funcionar en cualquier, empresa pública o privada, hospital o instituto, enfocado principalmente a los equipos de cómputo.

Dichos dispositivos pueden ser; impresoras, CPU, monitores, teclados, escáner, laptops, proyectores, switches, servidores, nobreak, cámaras ip.

De los cuales es conveniente llevar un control del inventario que nos permita saber en momento dado su ubicación, mantenimientos realizados, fallas en los mismos, usuario responsable, etc.

Lo que se pretende es tener la siguiente información que a continuación se enlista.

Usuarios

- Nombre
- Login

- Password
- Área de Soporte
- Tipo

Cliente

- Número de referencia
- Nombre
- Login
- Password
- Área
- Teléfono
- Departamento
- Ubicación
- Comentario

Reportes

- Número de referencia asignado al cliente y a los equipos
- Nombre del cliente que hizo el reporte
- Departamento
- Área
- Teléfono
- Ubicación
- Fecha del reporte
- Estado en que se encuentra el reporte
- Descripción de la falla
- Personal que atendió o le fue asignado el reporte
- Diagnóstico
- Comentarios

Equipo

- Número de referencia asignado
- Equipo (monitor, cpu, escáner, impresora, nobreak)
- Número de serie
- Número de inventario
- Comentario

Avisos

- Aviso
- A quien va dirigido
- Fecha de creación
- Nombre del equipo donde fue realizado
- IP del equipo donde fue realizado

Actividades

- ¿Quién?

- Descripción de la actividad
- Fecha en que se realizó
- Fecha de creación
- Nombre del equipo donde fue realizado
- IP del equipo donde fue realizado

Un paso importante es la conversión del esquema conceptual a un esquema lógico adecuado al modelo relacional. Para ello, se deben hacer algunas transformaciones: eliminar las relaciones de muchos a muchos, eliminar las relaciones complejas, eliminar las relaciones recursivas, eliminar las relaciones con atributos, eliminar los atributos multievaluados, reconsiderar las relaciones de uno a uno y eliminar las relaciones redundantes.

La normalización garantiza que el esquema resultante está más próximo al modelo de la empresa, es consistente, tiene la mínima redundancia y la máxima estabilidad.

Basándonos en las consideraciones anteriores, en los requerimientos del sistema y en los fundamentos de diseño de base de datos y aplicándolos para obtener un modelo que nos permita simplificar y relacionar dicha información enlistada anteriormente

Llegamos al siguiente modelo del Diagrama Entidad-Relación (DER)

4.3 DIAGRAMA ENTIDAD-RELACIÓN

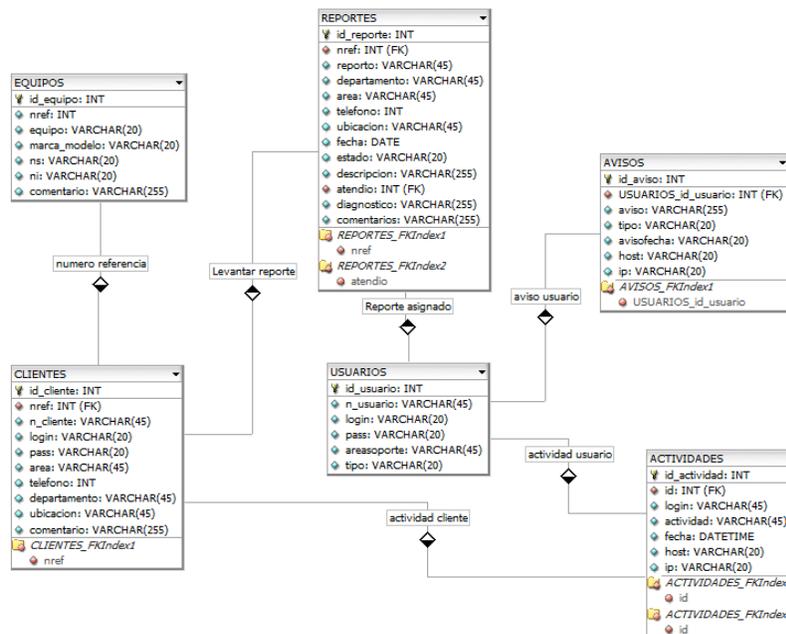


Figura 4.1 Diagrama entidad-relación

4.4 DICCIONARIO DE DATOS

Tabla	Nombre de los datos	Tipo	Longitud	Descripción
EQUIPOS	Id_equipo	INT	4	Llave primaria
	nref	INT	4	Tipo de equipo
	marca_modelo	VARCHAR	20	Marca y modelo
	ns	VARCHAR	20	Número de serie del equipo
	ni	VARCHAR	20	Número de inventario
	comentario	VARCHAR	255	Comentario
REPORTES	id_reporte	INT	4	Llave primaria
	nref	INT	4	Llave secundaria
	reporto	VARCHAR	45	Ciente que levanto el reporte
	departamento	VARCHAR	45	Departamento
	area	VARCHAR	45	Área
	telefono	INT	12	Teléfono
	ubicacion	VARCHAR	45	Ubicación
	fecha	DATE		Fecha
	estado	VARCHAR	20	Estado del reporte abierto o cerrado
	descripcion	VARCHAR	255	Descripción de la falla o anomalía que presenta el equipo
	atendio	VARCHAR	45	Ingeniero/ Técnico que resolvió la falla
diagnostico	VARCHAR	255	Descripción de cómo se dio solución a la falla	

	comentarios	VARCHAR	255	Comentarios
USUARIOS	Id_usuario	INT	4	Llave primaria
	n_usuario	VARCHAR	20	Nombre del empleado
	login_usuario	VARCHAR	20	Login
	pass_usuario	VARCHAR	20	Password
	areasoporte	VARCHAR	45	Area de soporte
	tipo	VARCHAR	20	Perfil del usuario
AVISOS	id_aviso	INT	4	Llave primaria
	aviso	VARCHAR	20	Aviso a publicar
	tipo	VARCHAR	45	Para quien es el aviso
	avisofecha	VARCHAR	45	Fecha de creación
	host	VARCHAR	time	Nombre del equipo
	ip	VARCHAR	20	Ip del equipo
ACTIVIDADES	id_actividades	INT	4	Llave primaria
	Id_cliente	INT	4	Llave secundaria
	login	VARCHAR	20	Login de usuario o cliente
	actividad	VARCHAR	255	Descripción de la actividad
	fecha	Date		Fecha
	host	VARCHAR	20	Nombre de la equipo
	ip	VARCHAR	20	Ip del equipo
CLIENTES	Id_cliente	INT	4	Llave primaria
	nref	INT	4	Llave secundaria
	n_cliente	VARCHAR	45	Nombre del cliente
	login	VARCHAR	20	Login
	pass	VARCHAR	20	Password
	area	VARCHAR	45	Área
	departamento	VARCHAR	45	Departamento
	telefono	INT	12	Telefono
	ubicacion	VARCHAR	45	Ubicación
comentario	VARCHAR	255	Comentario	

4.5 DISEÑO DE PANTALLAS

Pantalla de Acceso

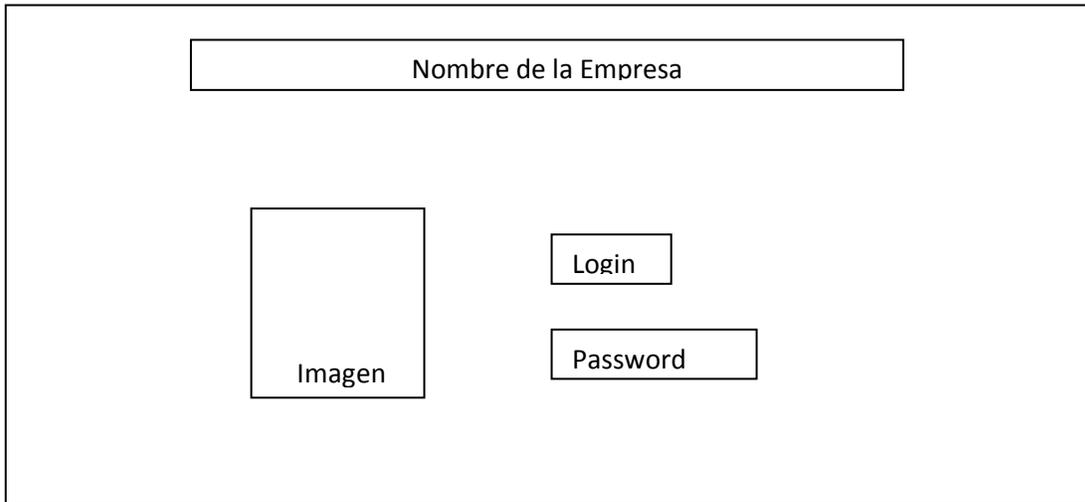


Figura 4.2 Pantalla de acceso

Pantalla de Administración

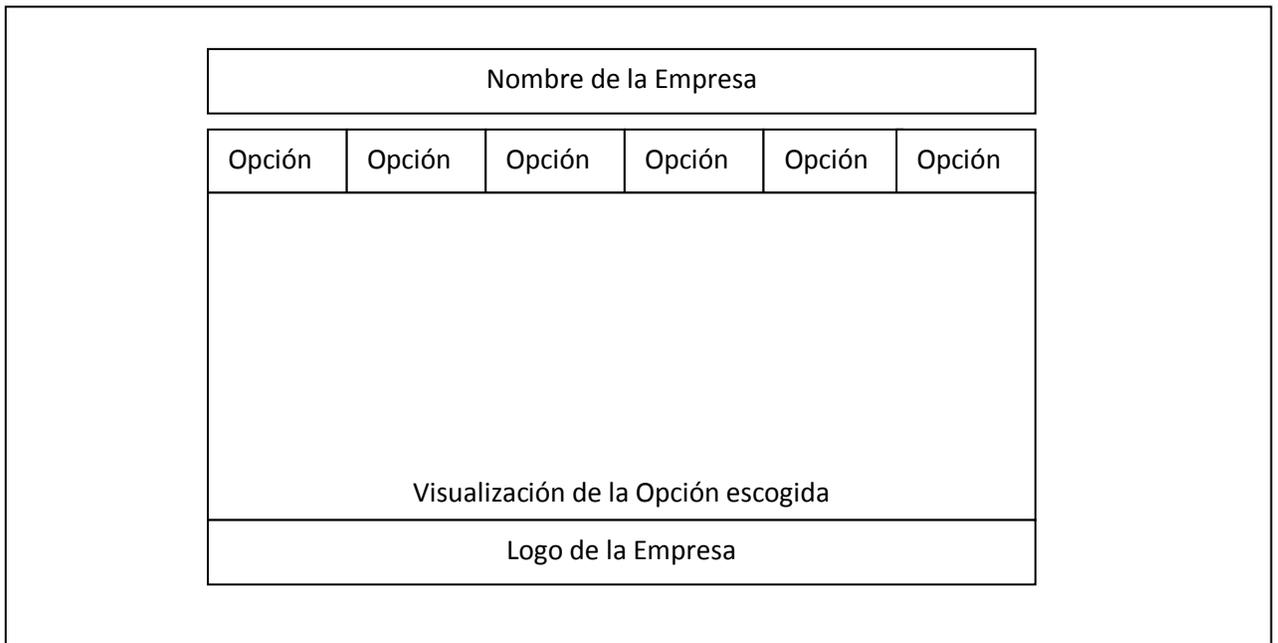


Figura 4.3 Pantalla de administración

Pantalla de cliente, mesa de ayuda e ingeniero/técnico

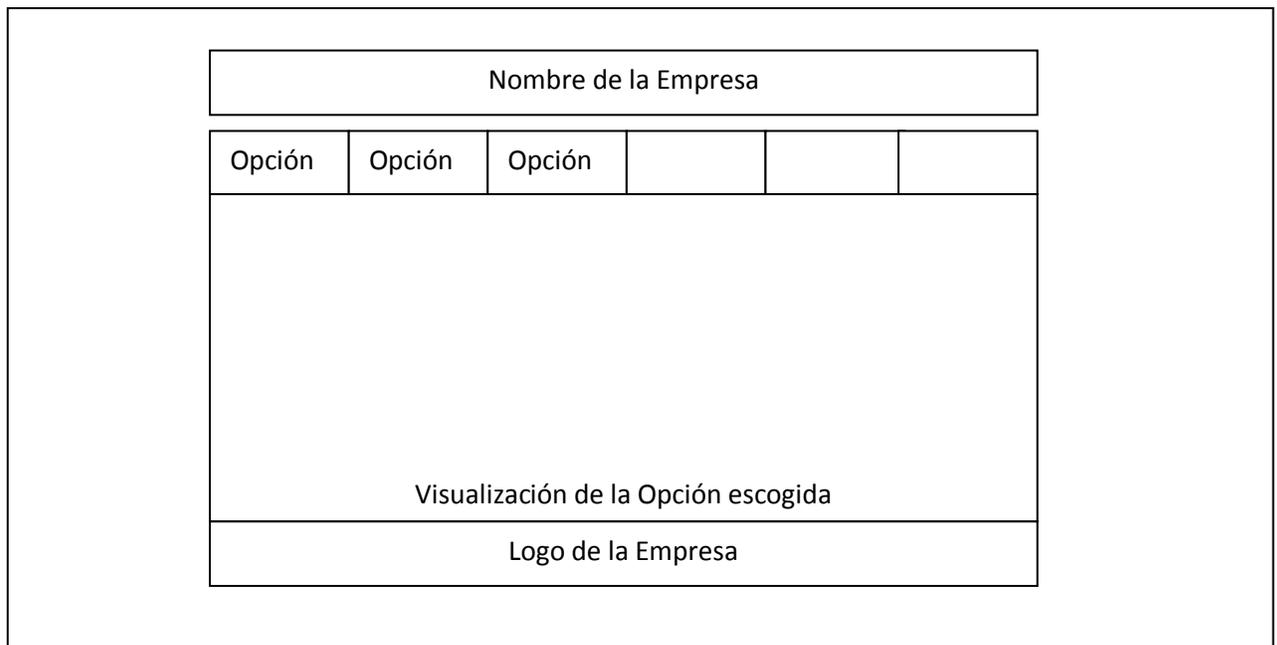


Figura 4.4 Otros perfiles

Para los usuarios que tengan perfil diferente a administrador las opciones que visualicen una vez que hayan iniciado sesión dependerán de las actividades que realicen

Como se había mencionado en el capítulo dos se crearon cuatro perfiles

- Cliente
- Técnico/Ingeniero
- Administrador
- Mesa de ayuda

El perfil **Cliente** únicamente tendrá acceso para realizar reportes o comentarios de su equipo.

El perfil **Técnico/Ingeniero** puede registrar en inventario equipo nuevo, bajas de equipo, modificación de información de los mismos y agregar reportes; cerrar reportes.

El perfil **Administrador** será el encargado de crear cuentas de usuarios, eliminar cuentas, soporte, mantenimiento a la base de datos y generar reportes. Tendrá la posibilidad de modificar el sistema de acuerdo a las necesidades que se presenten a lo largo del tiempo.

El perfil del usuario **Mesa de ayuda** permitirá agregar reportes de manera rápida que le soliciten vía telefónica u otro medio, cabe señalar que dicho perfil deberá contar con las herramientas necesarias que permitan al usuario de este perfil ingresar en el menor tiempo posible las incidencias que le sean reportadas.

CAPÍTULO 5 IMPLEMENTACIÓN DEL SISTEMA

5.1 CREACIÓN DE LA BASE DE DATOS

Una vez implementada la base de datos podemos para su fácil creación podemos utilizar un software para su implementación en el manejador de bases de datos para este caso mysql y el software que utilizaremos será DBDesigner 4

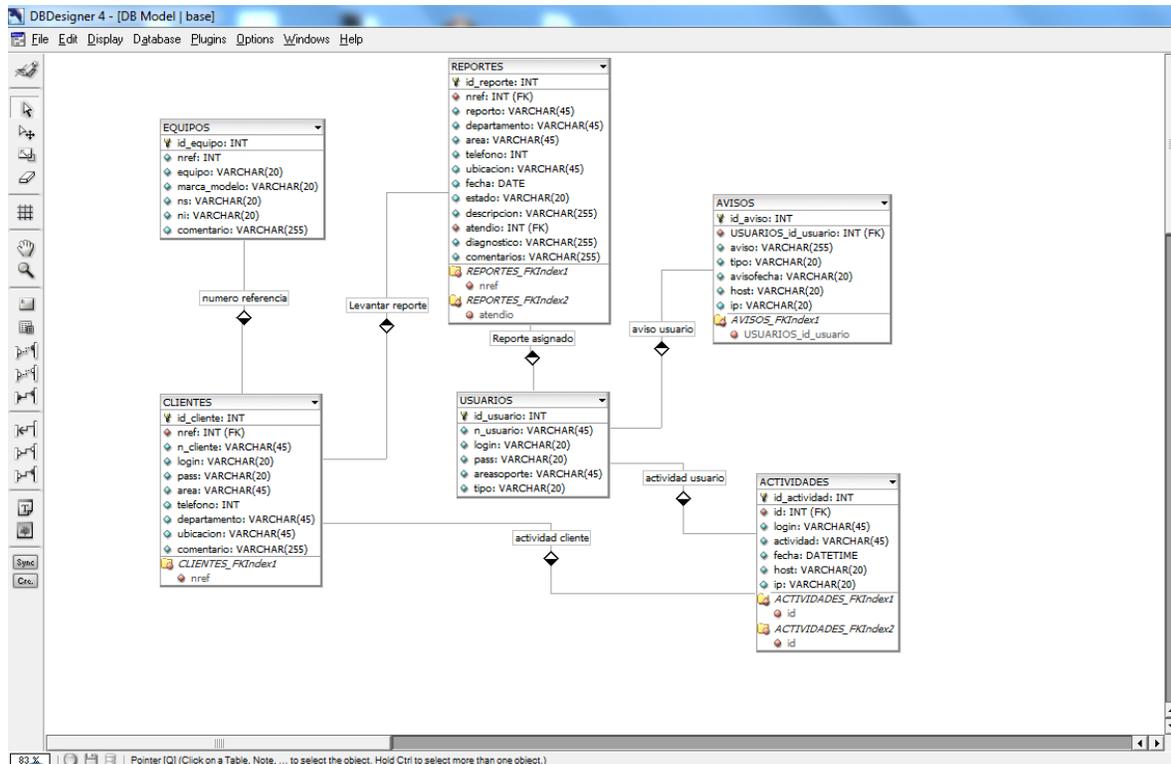


Figura 5.1 DBDesigner 4

Dicho software nos puede implementar el código sql e incluso realizar una conexión al manejador de base de datos Mysql para la implementación de la base de datos.

```
CREATE TABLE ACTIVIDADES (
  id_actividad INT NOT NULL AUTO_INCREMENT,
  id INT NOT NULL,
  login VARCHAR(45) NOT NULL,
  actividad VARCHAR(45) NOT NULL,
  fecha DATETIME NOT NULL,
  host VARCHAR(20) NOT NULL,
  ip VARCHAR(20) NOT NULL,
  PRIMARY KEY(id_actividad),
  INDEX ACTIVIDADES_FKIndex1(id),
```

```
INDEX ACTIVIDADES_FKIndex2(id)
);

CREATE TABLE AVISOS (
  id_avisos INT NOT NULL AUTO_INCREMENT,
  aviso VARCHAR(255) NOT NULL,
  tipo VARCHAR(20) NOT NULL,
  avisofecha VARCHAR(20) NOT NULL,
  host VARCHAR(20) NOT NULL,
  ip VARCHAR(20) NOT NULL,
  PRIMARY KEY(id_avisos)
);

CREATE TABLE CLIENTES (
  id_cliente INT NOT NULL AUTO_INCREMENT,
  nref INT NOT NULL,
  n_cliente VARCHAR(45) NOT NULL,
  login VARCHAR(20) NULL,
  pass VARCHAR(20) NULL,
  area VARCHAR(45) NOT NULL,
  telefono INT NULL,
  departamento VARCHAR(45) NOT NULL,
  ubicacion VARCHAR(45) NOT NULL,
  comentario VARCHAR(255) NULL,
  PRIMARY KEY(id_cliente),
  INDEX CLIENTES_FKIndex1(nref)
);

CREATE TABLE EQUIPOS (
  id_equipo INT NOT NULL AUTO_INCREMENT,
  nref INT NULL,
  equipo VARCHAR(20) NOT NULL,
  marca_modelo VARCHAR(20) NOT NULL,
  ns VARCHAR(20) NOT NULL,
  ni VARCHAR(20) NOT NULL,
  comentario VARCHAR(255) NULL,
  PRIMARY KEY(id_equipo)
);

CREATE TABLE REPORTES (
  id_reporte INT NOT NULL AUTO_INCREMENT,
  nref INT NOT NULL,
  reporto VARCHAR(45) NULL,
  departamento VARCHAR(45) NULL,
  area VARCHAR(45) NULL,
  telefono INT NULL,
```

```
ubicacion VARCHAR(45) NULL,
fecha DATE NOT NULL,
estado VARCHAR(20) NOT NULL,
descripcion VARCHAR(255) NOT NULL,
atendio INT NOT NULL,
diagnostico VARCHAR(255) NOT NULL,
comentarios VARCHAR(255) NOT NULL,
PRIMARY KEY(id_reporte),
INDEX REPORTES_FKIndex1(nref),
INDEX REPORTES_FKIndex2(atendio)
);
```

```
CREATE TABLE USUARIOS (
id_usuario INT NOT NULL AUTO_INCREMENT,
n_usuario VARCHAR(45) NOT NULL,
login VARCHAR(20) NOT NULL,
pass VARCHAR(20) NOT NULL,
areasoporte VARCHAR(45) NULL,
tipo VARCHAR(20) NOT NULL,
PRIMARY KEY(id_usuario)
);
```

Mysql nos permite crear rápidamente la base SAECOM copiando el código sql o importando el archivo que nos genero ABDesigner 4

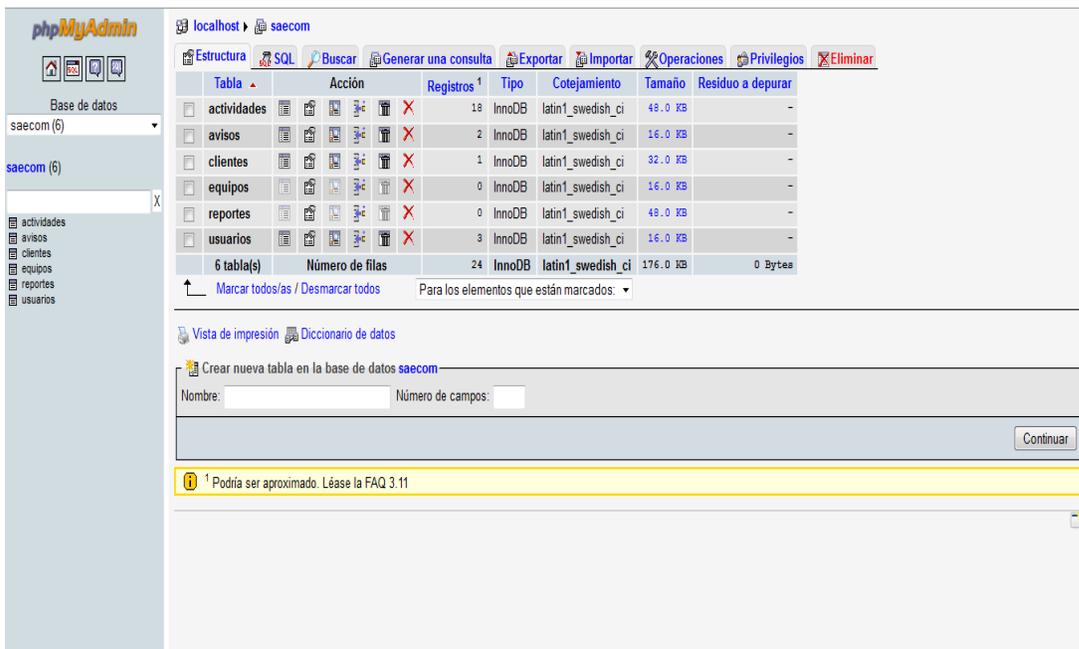


Figura 5.2 Phpmyadmin en Wampserver

5.2 CREACIÓN Y DISEÑO DE MÓDULOS

5.2.1 MÓDULO DE ACCESO



Figura 5.3 Módulo de Acceso

5.2.2 MÓDULO DE ADMINISTRACIÓN Y MANTENIMIENTO



Figura 5.4 Módulo de Administración

5.2.3 MÓDULO DE CAPTURA, CONSULTA E IMPRESIÓN

REPORTES DE EQUIPOS

Número de Reporte [Limpiar Búsqueda](#) [ACTUALIZAR](#)

Selección	Número de Reporte	Número de Referencia	Cliente	Descripción	Fecha	Estado	Atendido
<input type="radio"/>	25	9656	Benjamin Ayala Pérez	se apaga	2012-06-12 09:21:40	abierto	José Braulio Moreno López
<input type="radio"/>	24	9667	Jacinto Pérez Martínez	No hay internet	2012-05-11 23:50:47	terminado	Cruz Sergio Aguilar Díaz
<input type="radio"/>	20	9667	Jacinto Pérez Martínez	no tengo internet	2012-05-11 03:37:10	abierto	Mesa de Ayuda
<input type="radio"/>	13	4634	Beatriz Morales Mendoza	no prende	2012-05-11 00:24:36	abierto	Mesa de Ayuda
<input type="radio"/>	12	9439	Carlos Garcia Blanco	Nobreak emitiendo ruido	2012-06-20 15:00:22	abierto	José Braulio Moreno López
<input type="radio"/>	10	6495	Carlos Ventura Juárez	No imprime	2012-08-07 12:05:00	abierto	José Braulio Moreno López
<input type="radio"/>	8	5600	Julieta Pérez Tucan	Se reinicia equipo	0000-00-00 00:00:00	abierto	José Braulio Moreno López
<input type="radio"/>	7	5805	Tómas Mendoza Villegas	no imprime	2012-04-16 17:55:00	abierto	José Braulio Moreno López
<input type="radio"/>	6	9656	Benjamin Ayala Pérez	Se apaga monitor	2012-05-14 13:22:00	abierto	José Braulio Moreno López

Páginas 1

5.5 Módulo de captura perfil mesa de ayuda

REPORTES DE EQUIPOS

Número de Reporte [Limpiar Búsqueda](#) [ACTUALIZAR](#)

Selección	No de Reporte	No Referencia	Cliente	Descripción	Fecha	Estado	Atendido
<input type="radio"/>	25	9656	Benjamin Ayala Pérez	se apaga	2012-06-12 09:21:40	abierto	José Braulio Moreno López
<input type="radio"/>	20	9667	Jacinto Pérez Martínez	no tengo internet	2012-05-11 03:37:10	abierto	Mesa de Ayuda
<input type="radio"/>	13	4634	Beatriz Morales Mendoza	no prende	2012-05-11 00:24:36	abierto	Mesa de Ayuda
<input type="radio"/>	12	9439	Carlos Garcia Blanco	Nobreak emitiendo ruido	2012-06-20 15:00:22	abierto	José Braulio Moreno López
<input type="radio"/>	10	6495	Carlos Ventura Juárez	No imprime	2012-08-07 12:05:00	abierto	José Braulio Moreno López
<input type="radio"/>	8	5600	Julieta Pérez Tucan	Se reinicia equipo	0000-00-00 00:00:00	abierto	José Braulio Moreno López
<input type="radio"/>	7	5805	Tómas Mendoza Villegas	no imprime	2012-04-16 17:55:00	abierto	José Braulio Moreno López
<input type="radio"/>	6	9656	Benjamin Ayala Pérez	Se apaga monitor	2012-05-14 13:22:00	abierto	José Braulio Moreno López

Páginas 1

5.6 Módulo de captura perfil soporte



GENERAR REPORTE

Número de Reporte	Número de Referencia	Cliente	Descripción	Fecha	Estado	Diagnóstico
24	9667	Jacinto Pérez Martínez	No hay internet	2012-05-11 23:50:47	terminado	Cable desconectado
20	9667	Jacinto Pérez Martínez	no tengo internet	2012-05-11 03:37:10	abierto	Reporte realizado por el cliente

Páginas 1

Generar



5.7 Módulo de captura perfil cliente

5.4 CREACIÓN DEL PORTAL WEB DEL SISTEMA

Una vez creados los diferentes módulos estos deben seguir una estructura dependiendo de qué usuario es el que utilice el sistema. La información que introduzca en los diferentes módulos deberá ser almacenada de forma correcta para que cuando sea consultada se muestre de forma correcta y sea verídica

Para montar el portal web lo haremos en un sistema operativo Linux distribución Ubuntu. Como fue mencionado anteriormente utilizaremos software libre. Podemos adquirir sistema de la página de descarga <http://ubuntumexico.org/>.

Una vez realizada la descarga procederemos a instalarlo.

Iniciamos nuestro equipo dedicado en donde instalaremos nuestro sistema. Veremos la siguiente pantalla



Figura 5.8 Inicio de livecd

Escogemos el idioma y damos clic en instalar Ubuntu

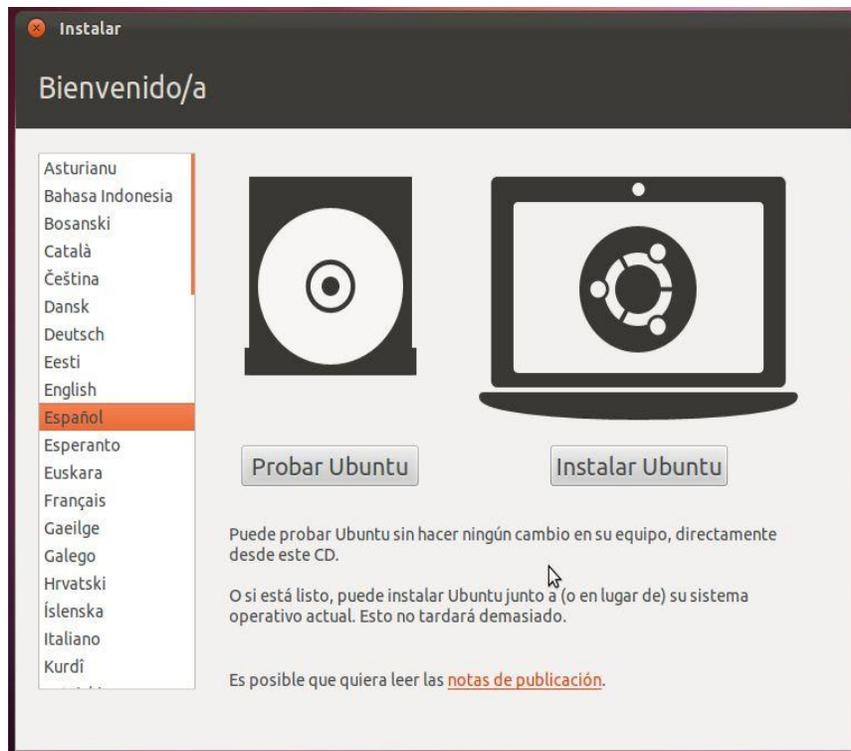


Figura 5.9 Pantalla de bienvenido/a

Seleccionamos Descargar actualizaciones mientras se instala

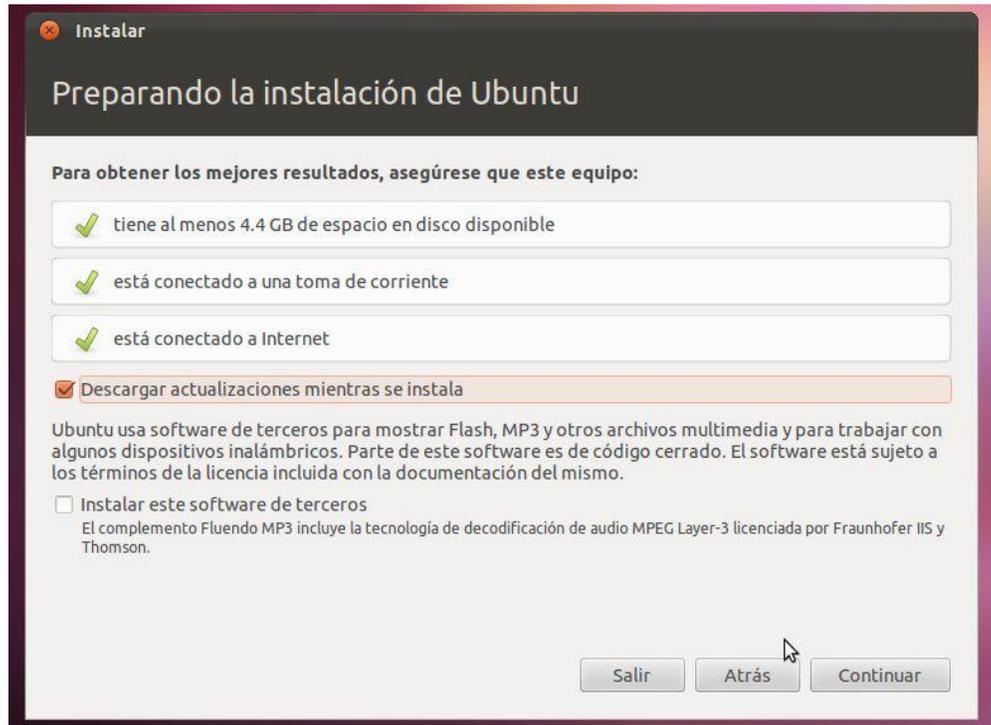


Figura 5.10 Preparando instalación

En la siguiente pantalla seleccionamos Borrar disco e instalar Ubuntu

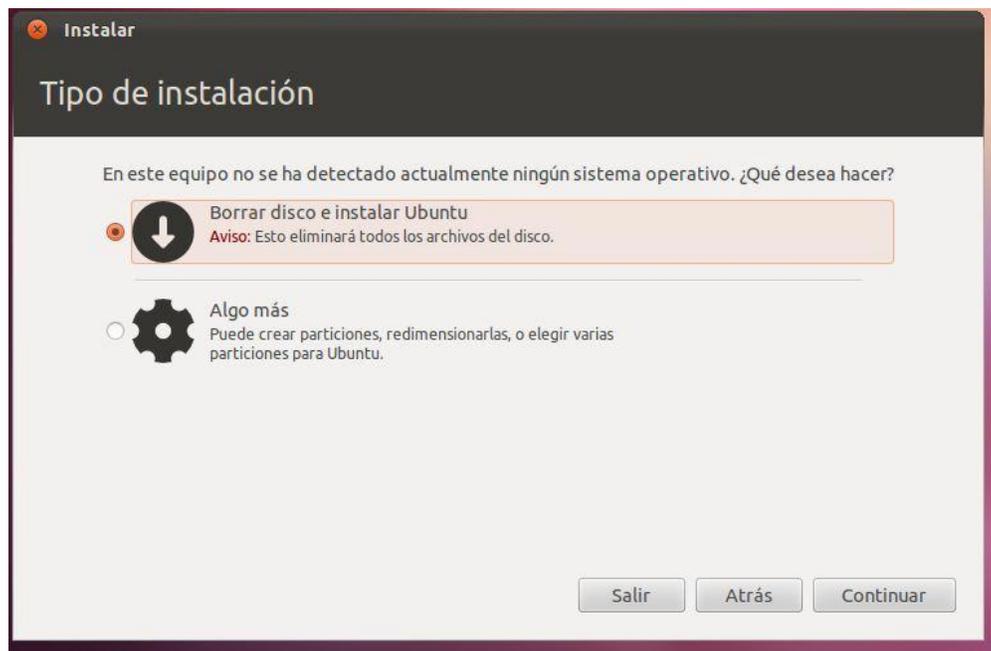


Figura 5.11 Tipo de instalación

Colocamos nuestra ubicación y damos clic en continuar



Figura 5.12 Ubicación

Escogemos la distribución del teclado y damos clic en continuar

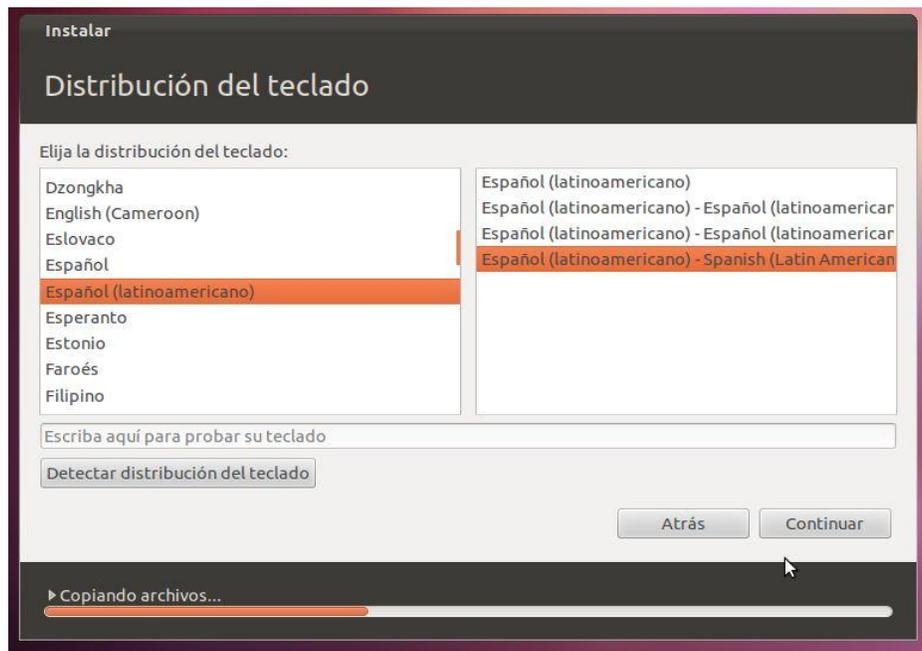


Figura 5.13 Distribución del teclado

Llenamos nuestros datos y damos clic en continuar

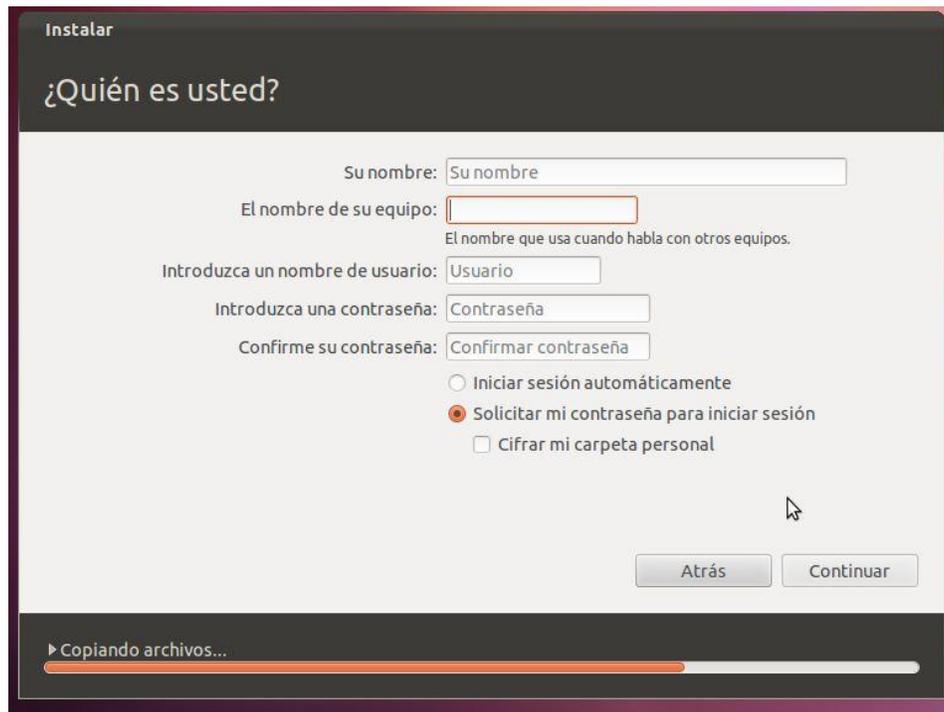


Figura 5.14 Datos del equipo

Esperamos el proceso de instalación



Figura 5.14 Proceso de instalación

La instalación finaliza con la siguiente ventana

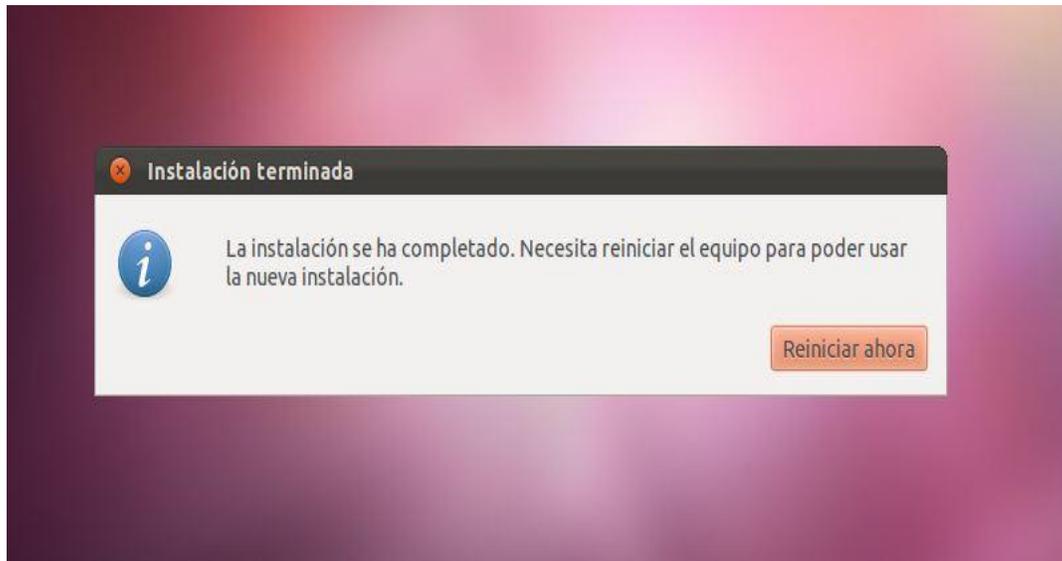


Figura 5.15 Instalación finalizada

Después del reinicio nos aparecerá la pantalla donde podremos iniciar sesión

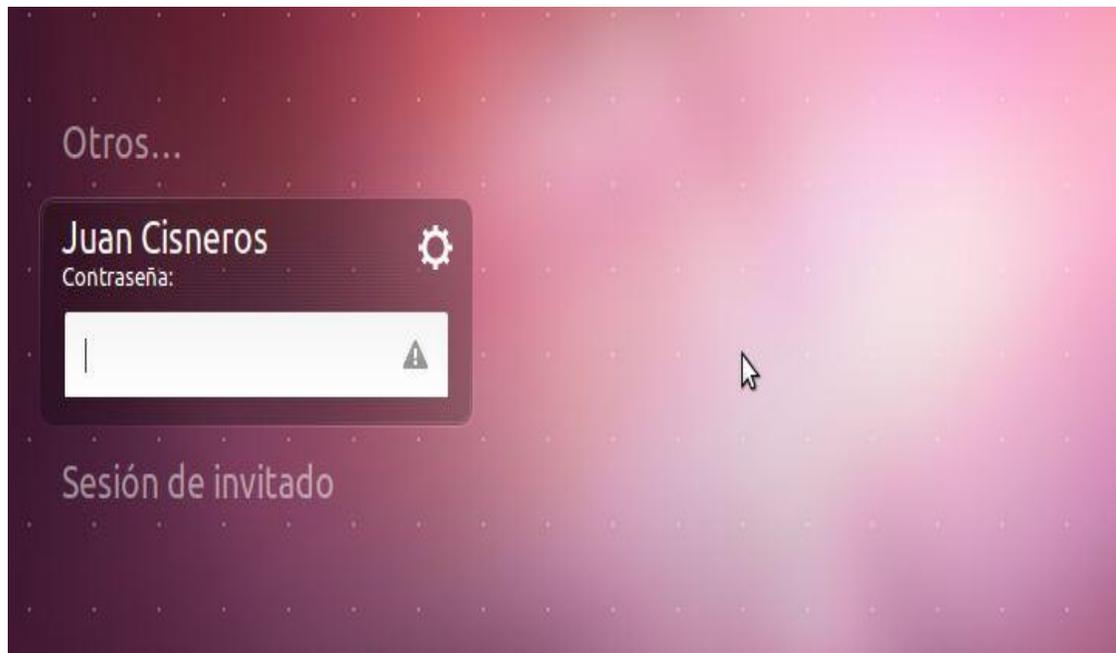


Figura 5.16 Pantalla de logueo al sistema Ubuntu

Después que reinicio el sistema iniciamos sesión y vamos a instalar LAMP abrimos una terminal de Ubuntu

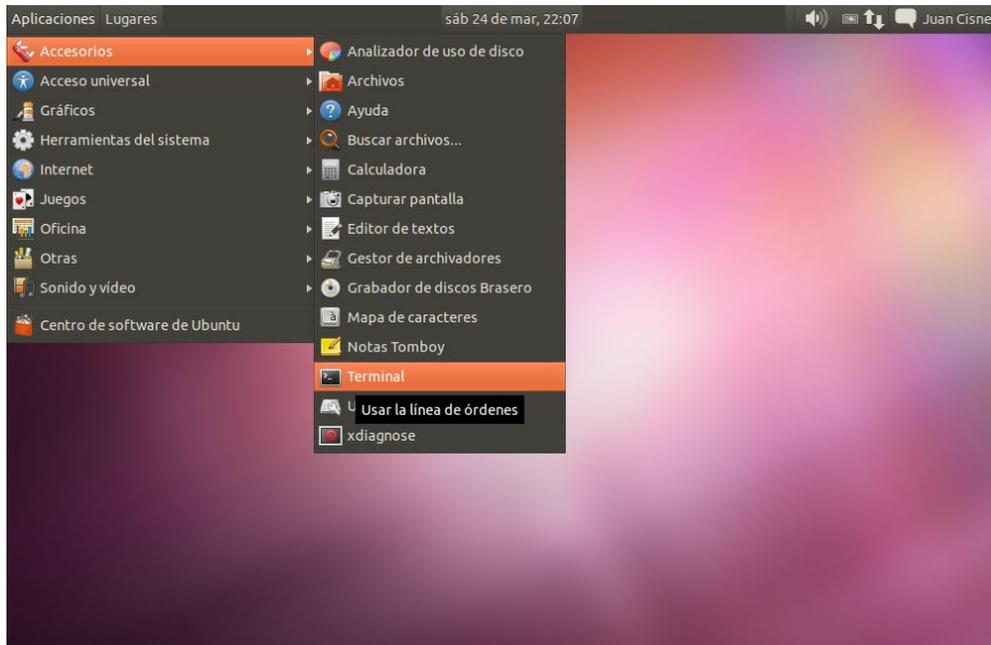


Figura 5.17 Instalar LAMP

Colocamos en la terminal el siguiente comando

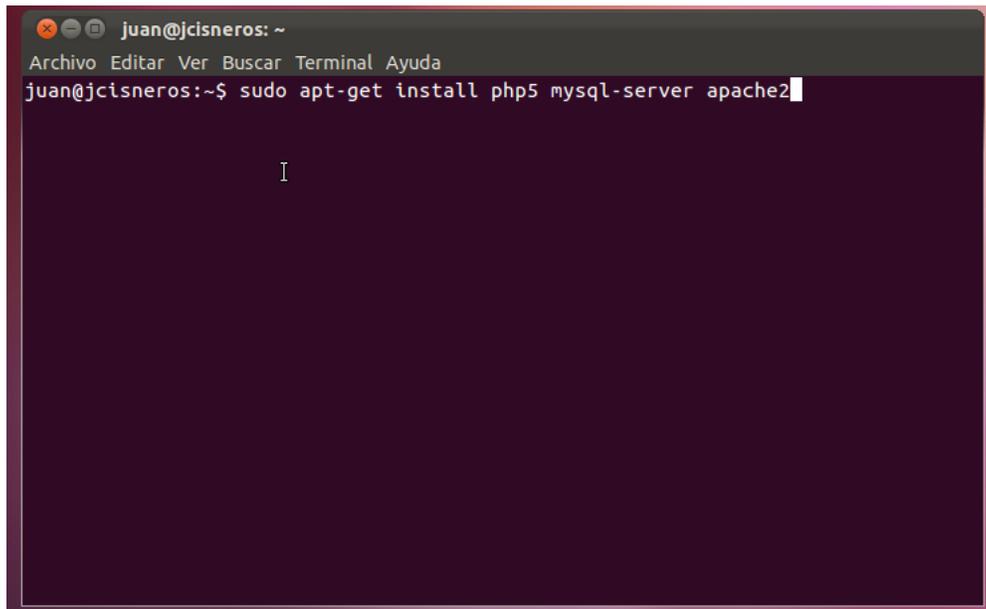


Figura 5.18 Comando para instalar LAMP

Colocamos la contraseña del usuario root de mysql

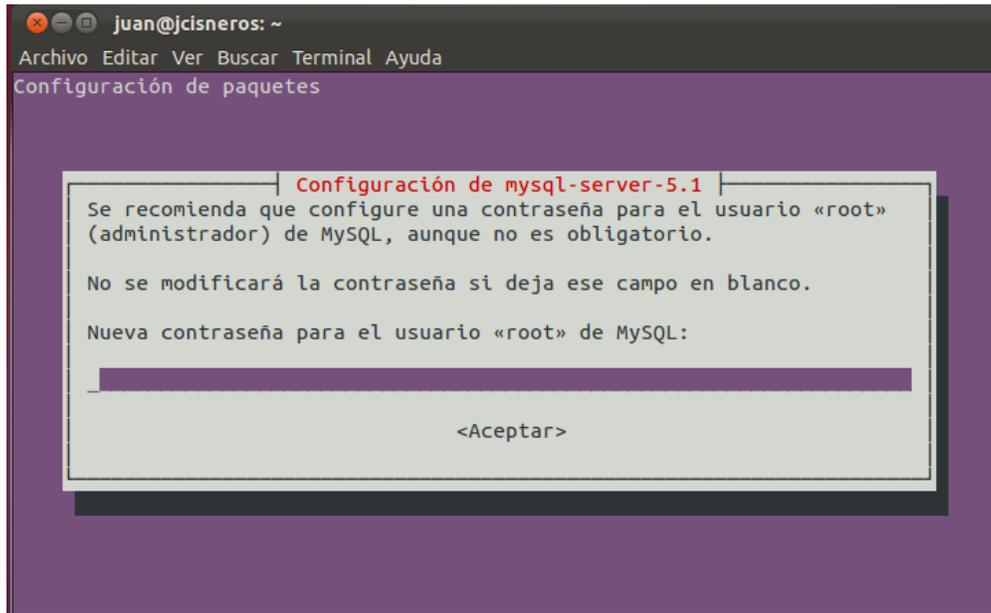


Figura 5.19 Contraseña de usuario root

Al terminar la instalación abrimos nuestro navegador y colocamos localhost si visualizamos la siguiente página la instalación se realizó sin problemas

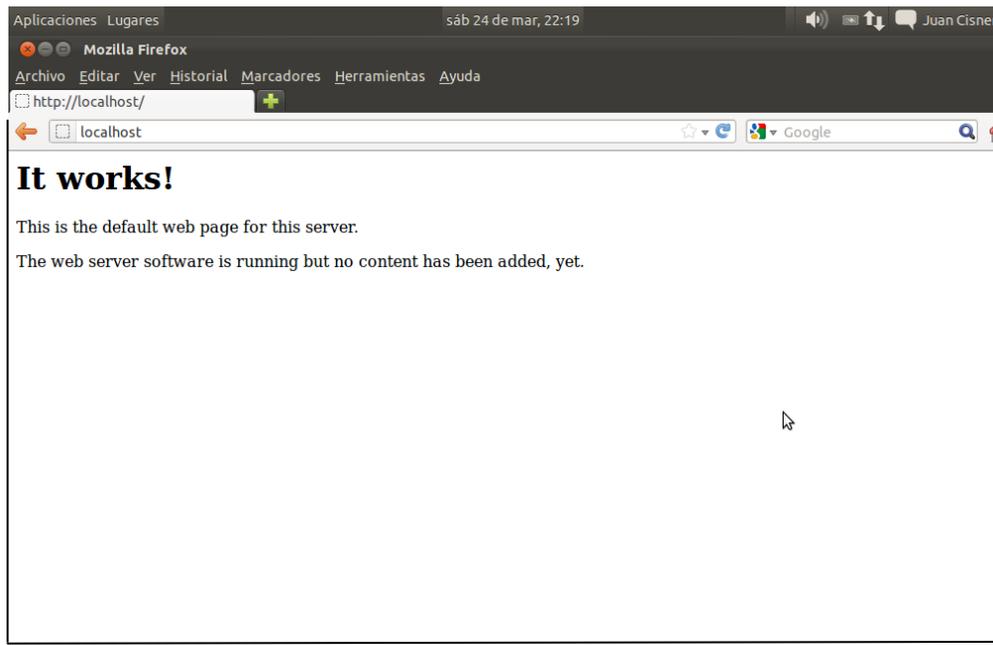
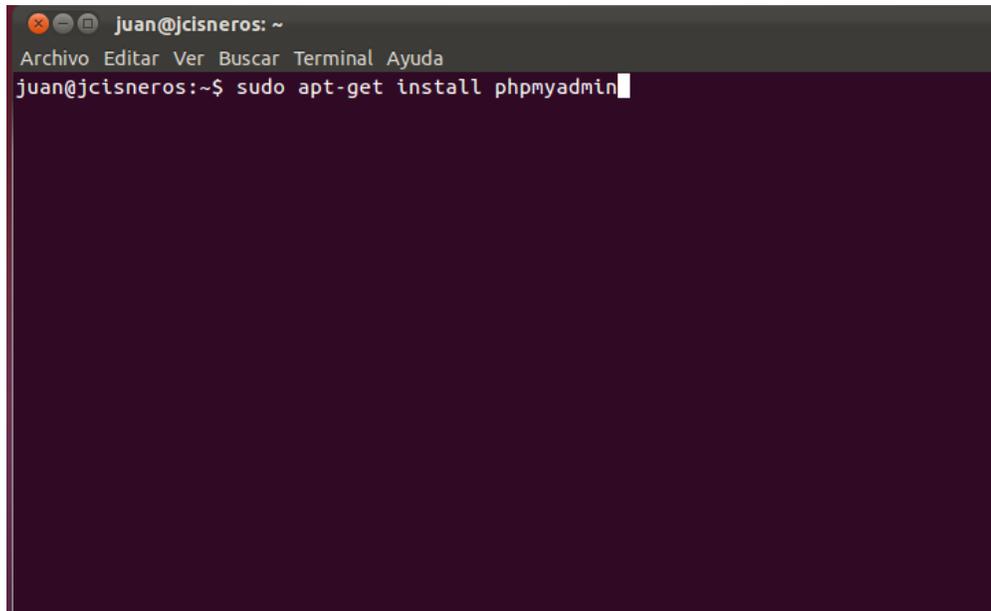


Figura 5.20 Instalación correcta

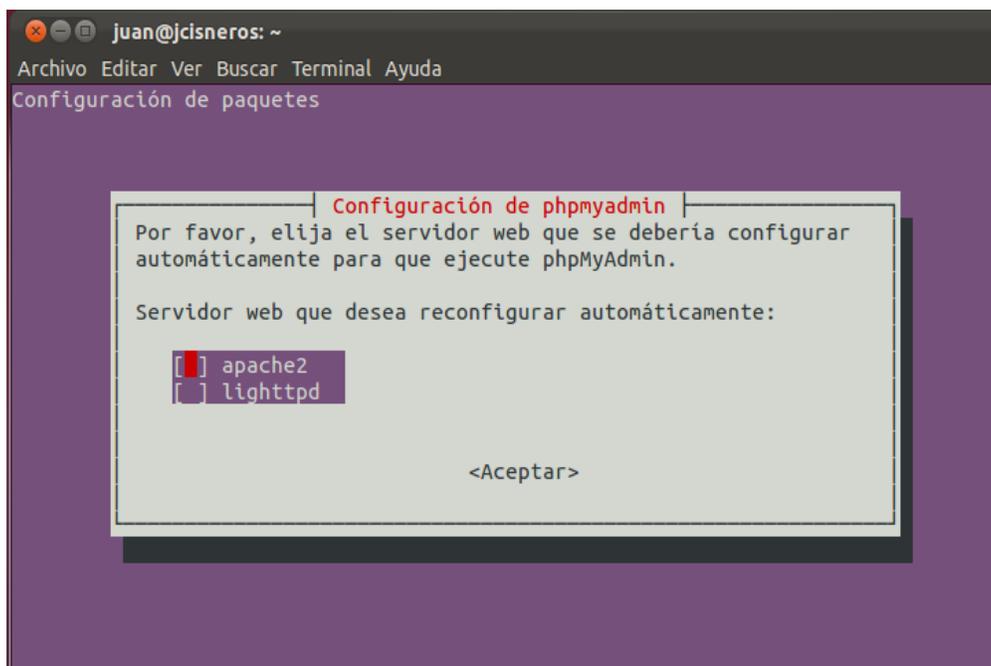
Ahora vamos a instalar phpmyadmin para la base de datos. Abrimos nuevamente una terminal y digitamos el siguiente comando.



```
juan@jcsneros: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
juan@jcsneros:~$ sudo apt-get install phpmyadmin
```

Figura 5.21 Comando para instalar phpmyadmin

En la siguiente pantalla seleccionamos apache2



```
juan@jcsneros: ~  
Archivo Editar Ver Buscar Terminal Ayuda  
Configuración de paquetes  
  
Configuración de phpmyadmin  
Por favor, elija el servidor web que se debería configurar automáticamente para que ejecute phpMyAdmin.  
Servidor web que desea reconfigurar automáticamente:  
  
[ ] apache2  
[ ] lighttpd  
  
<Aceptar>
```

Figura 5.22 instalación de apache

Al terminar la instalación abrimos nuestro navegador y colocamos localhost/phpmyadmin si visualizamos la siguiente página la instalación se realizó sin problemas

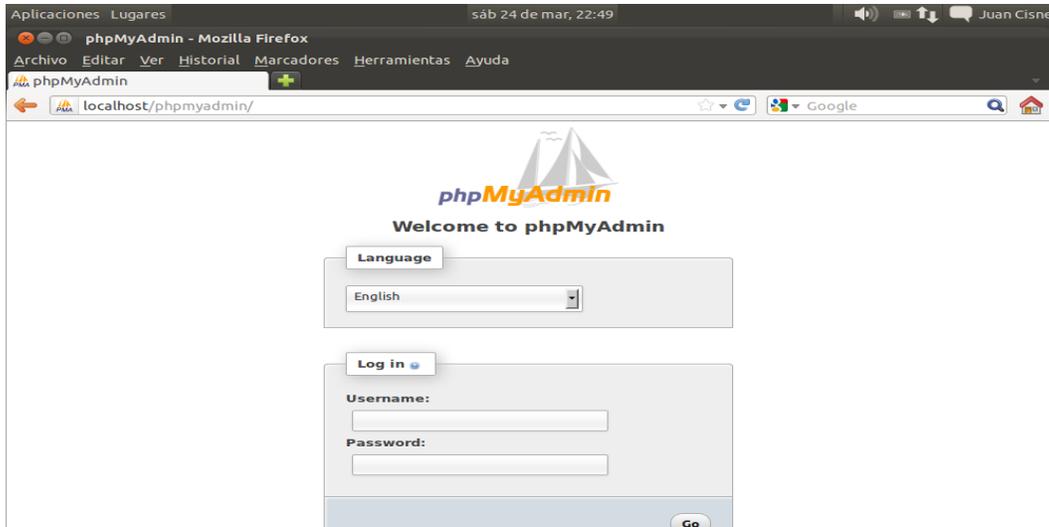


Figura 5.23 instalación correcta de phpmyadmin

5.5 CONFIGURACIÓN Y PUESTA EN MARCHA

Iniciamos nuestro navegador y nos dirigimos a phpmyadmin. Iniciamos sesión con el usuario y contraseña que utilizamos para configurar mysql en la instalación anterior.

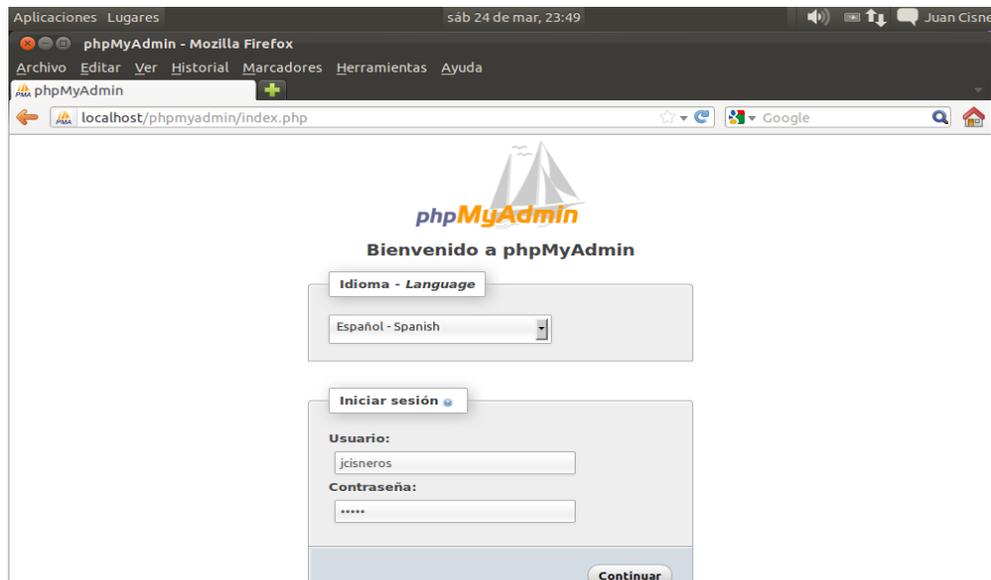


Figura 5.24 pantalla de inicio phpmyadmin

Creamos una nueva base de datos en la pestaña Bases de Datos le llamaremos saecom.

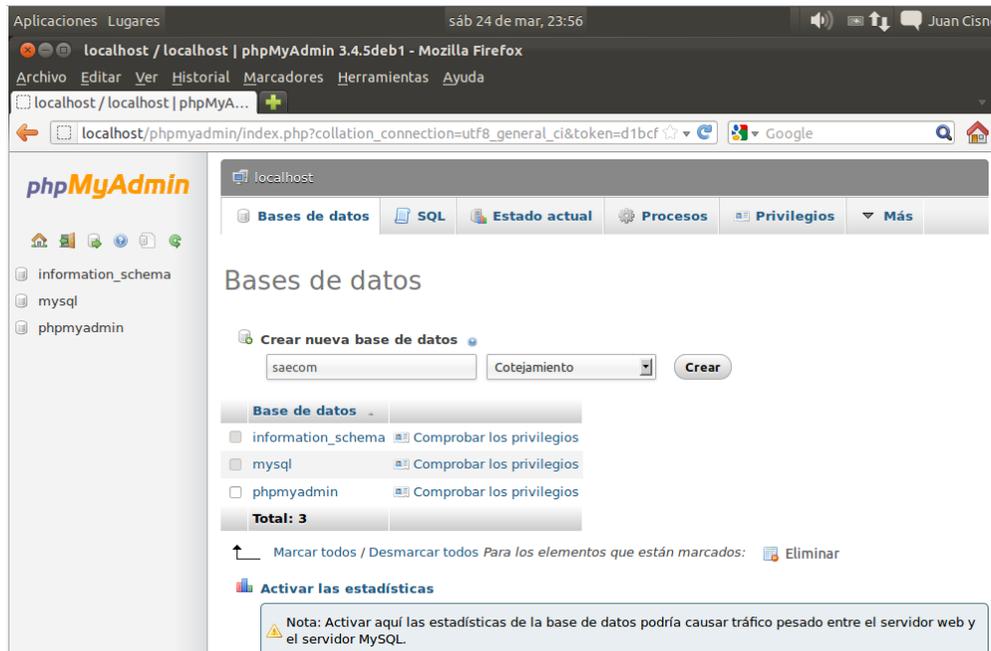


Figura 5.25 creación de la base de datos en phpmyadmin

Una vez creada nos aparecerá del lado izquierdo la seleccionamos he importamos nuestra base de datos

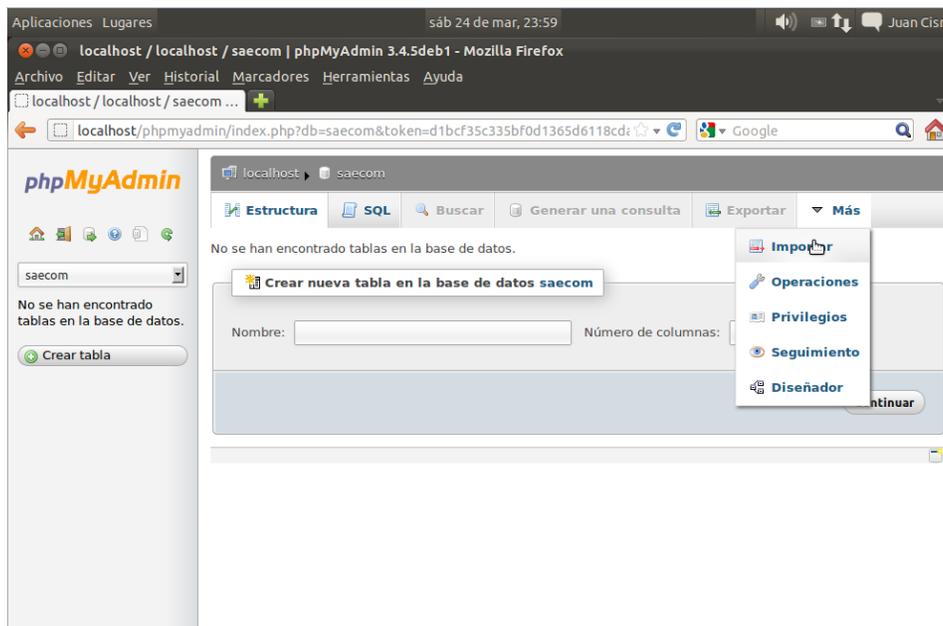


Figura 5.26 Importando base de datos

Si visualizamos la siguiente pantalla la importación se realizó sin problemas

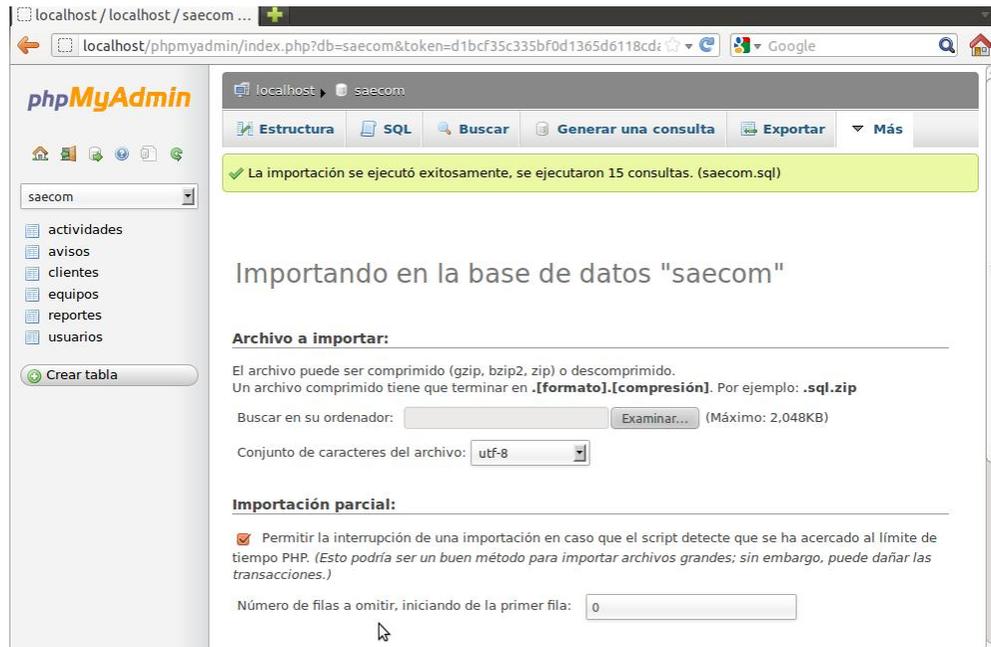


Figura 5.27 Importación de la base de datos correcta

Ahora vamos a colocar nuestros programas del sitio en la carpeta www de nuestro lamp. Sistema de Archivos → var → www

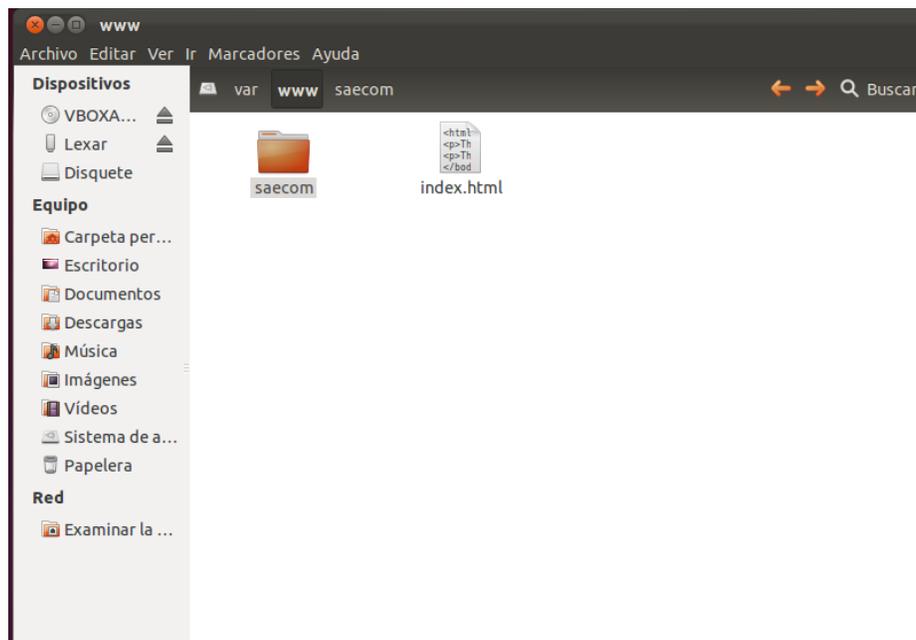


Figura 5.28 Carpeta del sitio

Desde phpmyadmin creamos un usuario que es con el que se conectar la aplicación a la base de datos.

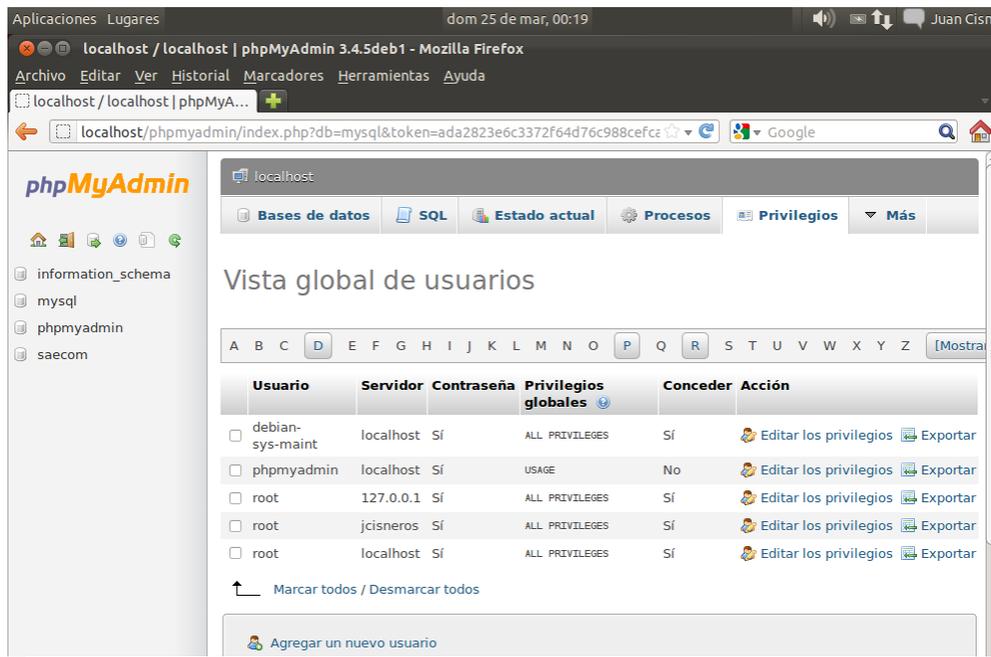


Figura 5.29 Usuario nuevo en phpmyadmin

El usuario y contraseña que creamos en phpmyadmin lo colocamos en el archivo conectar.php

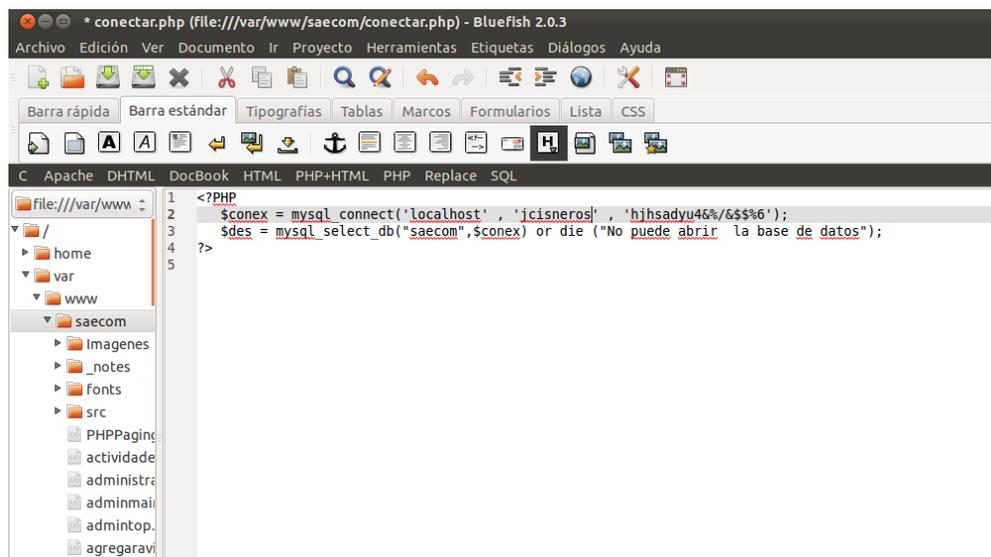


Figura 5.30 Configuración del archivo conectar.php

Desde nuestro navegador digitamos la siguiente dirección para comprobar que nuestra aplicación este funcionando



Figura 5.31 Pantalla de inicio del sistema saecom

Para utilizar el sistema desde otro equipo conectado a la misma red primero investigamos que IP tiene asignado el equipo donde instalamos el sistema SAECOM de la siguiente forma

```

juan@jcsisneros: ~
Archivo Editar Ver Buscar Terminal Ayuda
juan@jcsisneros:~$ ifconfig
eth0      Link encap:Ethernet  direcciónHW 08:00:27:16:a6:25
          Direc. inet:192.168.56.102  Difus.:192.168.56.255  Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:fe16:a625/64  Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST  MTU:1500  Métrica:1
          Paquetes RX:223 errores:1 perdidos:0 overruns:0 frame:0
          Paquetes TX:88 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:31749 (31.7 KB)  TX bytes:26133 (26.1 KB)
          Interrupción:11 Dirección base: 0xd020
    
```

Figura 5.32 IP del sistema saecom

En el equipo de donde nos vamos a conectar abrimos nuestro navegador y digitamos la dirección IP que obtuvimos. IP/saecom si visualizamos la siguiente pantalla nuestro sistema SAECOM lo tenemos funcionando en cualquier equipo que se encuentre dentro del segmento de red en el que tenemos nuestro equipo donde montamos nuestro sistema.

CAPÍTULO 6 PRUEBAS Y DEPURACIÓN DEL SISTEMA

6.1 PRUEBAS DE LÓGICA

El más utilizado al iniciar sesión

Mensaje sin escribir usuario y contraseña

The image shows two parts of a web application. On the left is a login form titled "BIENVENIDO". It has two input fields: "Login" and "Password", both of which are empty. Below the fields is a button labeled "Enviar". On the right is a modal dialog box with a light gray background. The text inside the dialog says "Olvidaste escribir tu usuario y contraseña". At the bottom right of the dialog is a button labeled "Aceptar".

Figura 6.1 Campos vacíos

Escribiendo sólo el usuario

The image shows two parts of a web application. On the left is a login form titled "BIENVENIDO". The "Login" field contains the text "admin" and the "Password" field is empty. Below the fields is a button labeled "Enviar". On the right is a modal dialog box with a light gray background. The text inside the dialog says "Olvidaste escribir tu contraseña". At the bottom right of the dialog is a button labeled "Aceptar".

Figura 6.2 Sólo usuario

Escribiendo sólo la contraseña

The image shows two parts of a web application. On the left is a login form titled "BIENVENIDO". The "Login" field is empty and the "Password" field contains five dots. Below the fields is a button labeled "Enviar". On the right is a modal dialog box with a light gray background. The text inside the dialog says "Olvidaste escribir tu login". At the bottom right of the dialog is a button labeled "Aceptar".

Figura 6.3 Sólo contraseña

Al borrar un registro



Figura 6.4 Borrar registro

6.2 PRUEBAS UNITARIAS Y DE INTEGRACIÓN

Agregaremos un cliente Miguel Sánchez Martínez

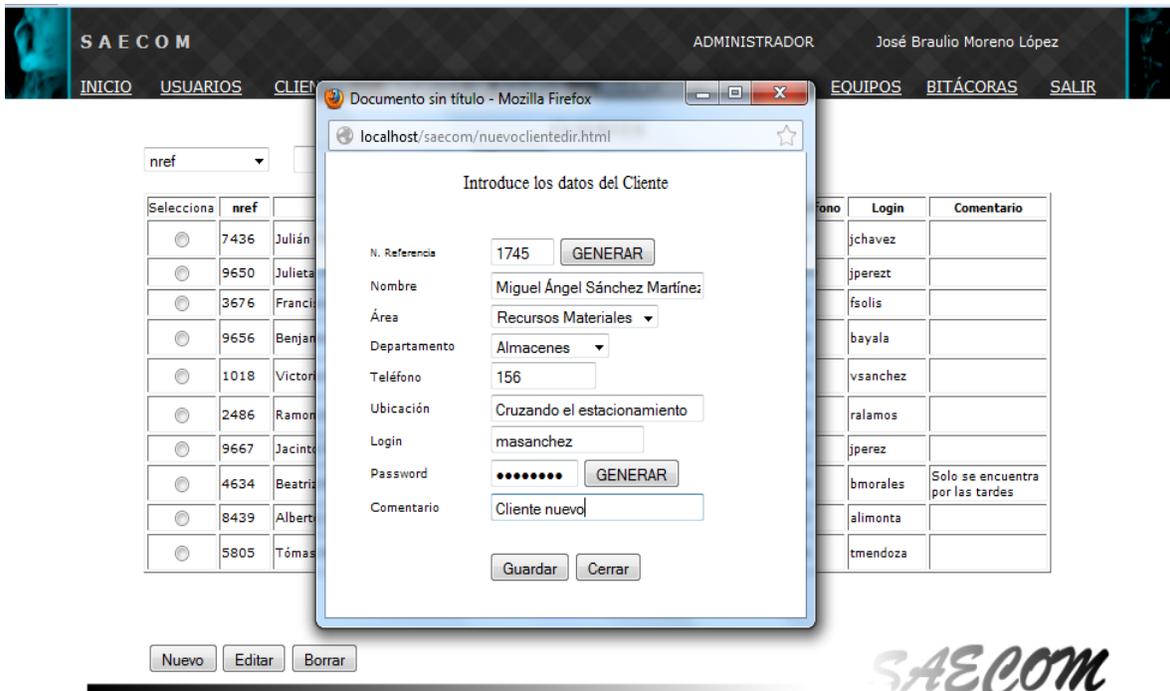


Figura 6.5 Agregar cliente

Podemos verificar que se ha registrado en la lista de cliente



Figura 6.6 Cliente agregado

Verificamos desde la base de datos



Figura 6.7 Cliente visualizado en la base de datos

No se le ha asignado ningún equipo

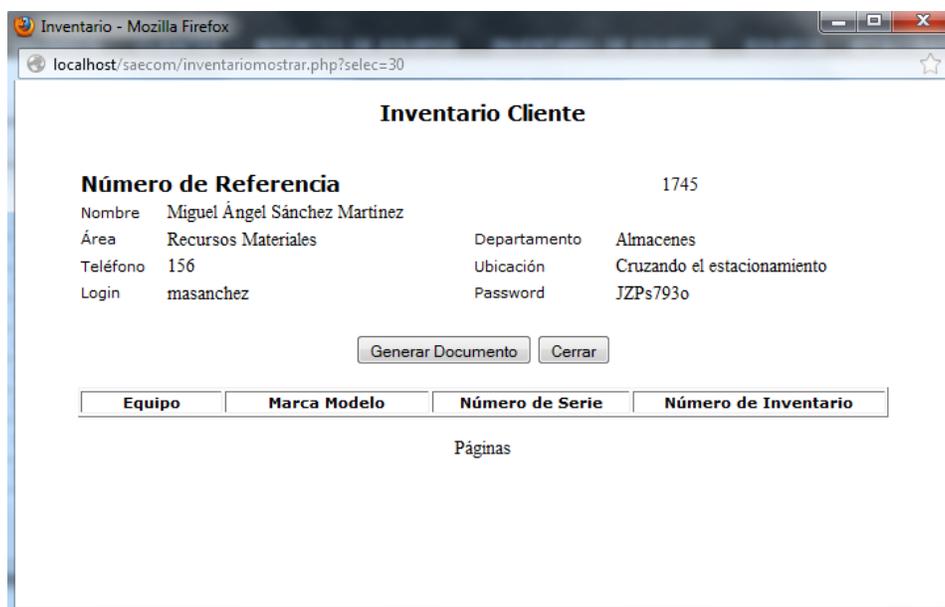


Figura 6.8 Inventario cliente

6.3 DETECCIÓN DE ERRORES DE VALIDACIÓN

Error al guardar la fecha de reporte



SAECOM

Figura 6.9 Error en la fecha de reporte

Se presenta error al editar reportes

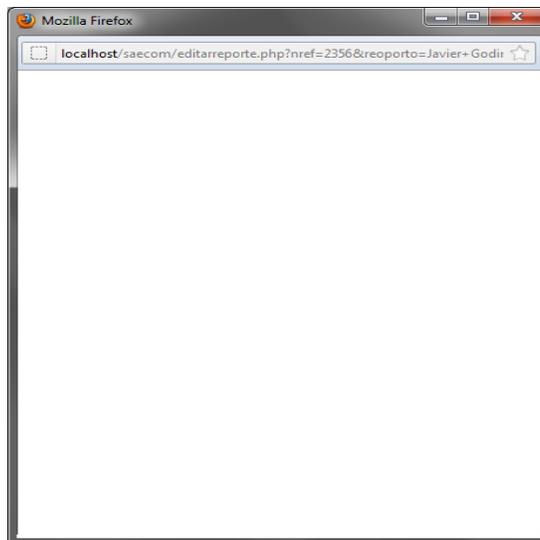


Figura 6.10 Error al editar reporte

6.4 PRUEBAS DE CARGA Y RECORRIDO DE LA INFORMACIÓN

Inicio de sesión



Figura 6.11 Pantalla de inicio

Registro de actividad



Figura 6.12 Bitácora

Publicación de avisos para cada perfil



Figura 6.13 Publicación de avisos

Perfil cliente

The screenshot shows the SAECOM client profile interface. The header includes the SAECOM logo, the user role 'CLIENTE', and the name 'Jacinto Pérez Martínez'. Navigation links include 'INICIO', 'REPORTAR EQUIPO', and 'SALIR'. The main content area is titled 'AVISOS' and contains a table with the following data:

Aviso	Fecha
El próximo jueves se realiza actualización de aplicación	2012-09-19 04:07:10

Below the table, it indicates 'Páginas 1'.

Figura 6.14 Aviso cliente

Perfil soporte

The screenshot shows the SAECOM support profile interface. The header includes the SAECOM logo, the user role 'SOPORTE', and the name 'Erick Octavio Benítez López'. Navigation links include 'INICIO', 'REPORTES DE EQUIPOS', 'INVENTARIO DE EQUIPOS', 'EQUIPOS', and 'SALIR'. The main content area is titled 'AVISOS' and contains a table with the following data:

Aviso	Fecha
Comenzar los mantenimientos del 1er piso de edificio A	2012-05-04 04:03:12
Recabar firmas de los clientes del mes de marzo	2012-03-25 03:51:13

Below the table, it indicates 'Páginas 1'.

Figura 6.15 Aviso soporte

Perfil Mesa de Ayuda

The screenshot shows the SAECOM help desk profile interface. The header includes the SAECOM logo, the user role 'MESA DE AYUDA', and the name 'Melissa Juárez Reyes'. Navigation links include 'INICIO', 'REPORTES DE EQUIPOS', and 'SALIR'. The main content area is titled 'AVISOS' and contains a table with the following data:

Aviso	Fecha
El próximo viernes daran mantenimiento los ingenieros al 1er piso favor de avisar a las áreas correspondientes	2012-09-19 04:07:40
Próximo martes actualización de antivirus	2012-09-19 04:07:57

Below the table, it indicates 'Páginas 1'.

Figura 6.16 Aviso mesa de ayuda

Agregar usuario

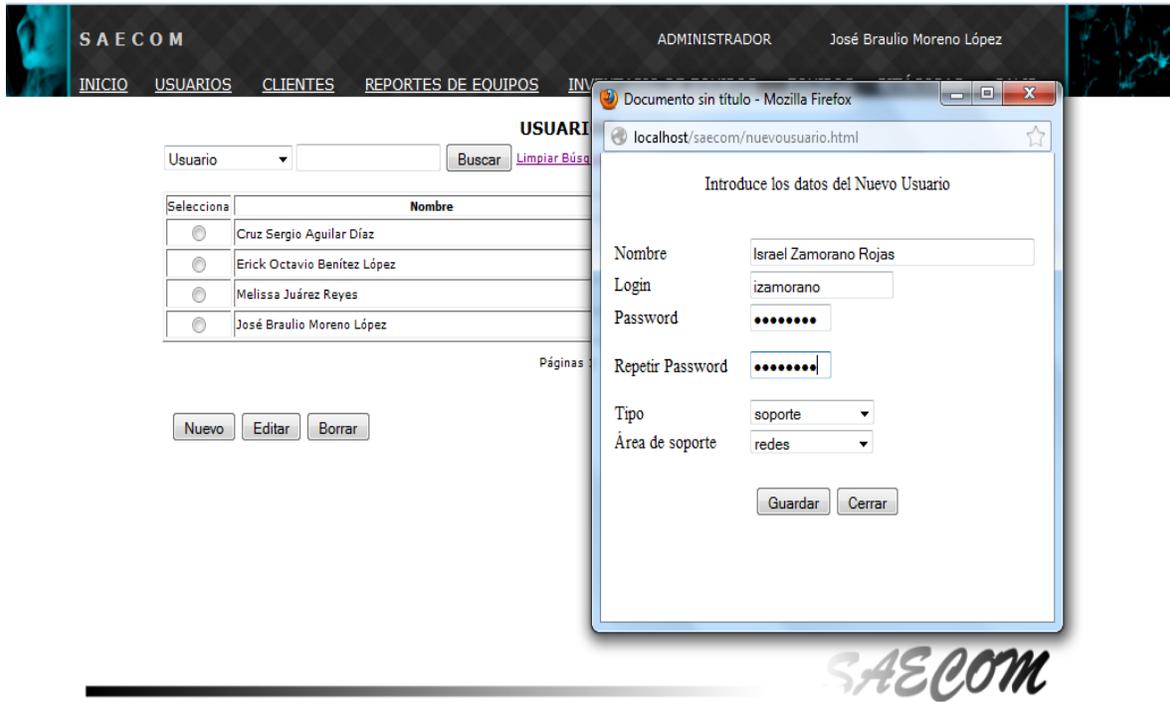


Figura 6.17 Agregar usuario

Usuario agregado

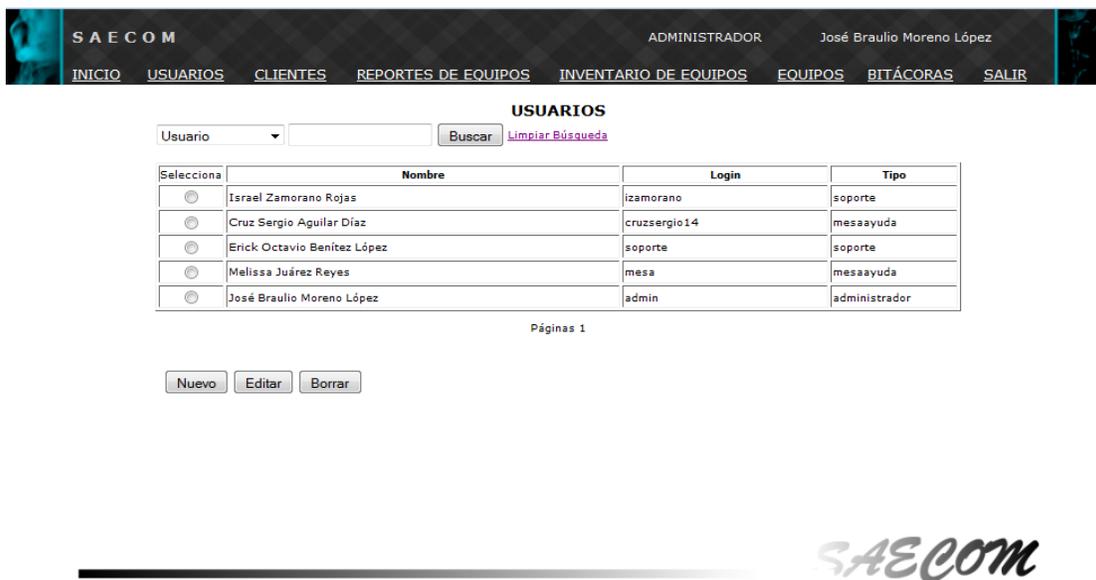


Figura 6.18 usuario agregado

Agregar cliente

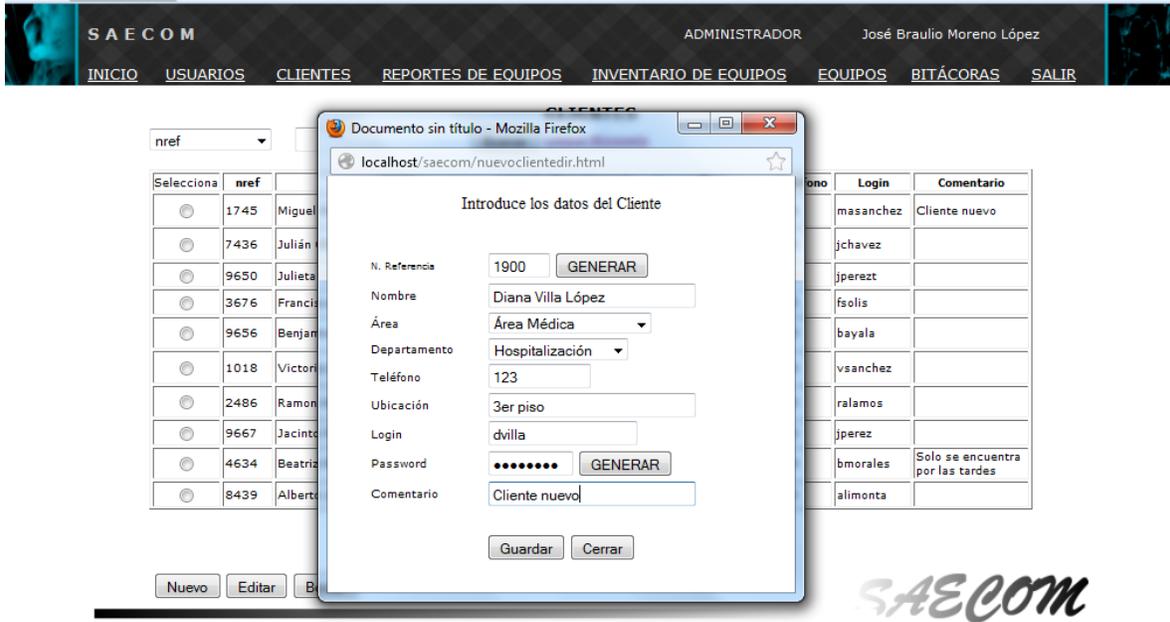


Figura 6.19 Agregar cliente

Cliente agregado

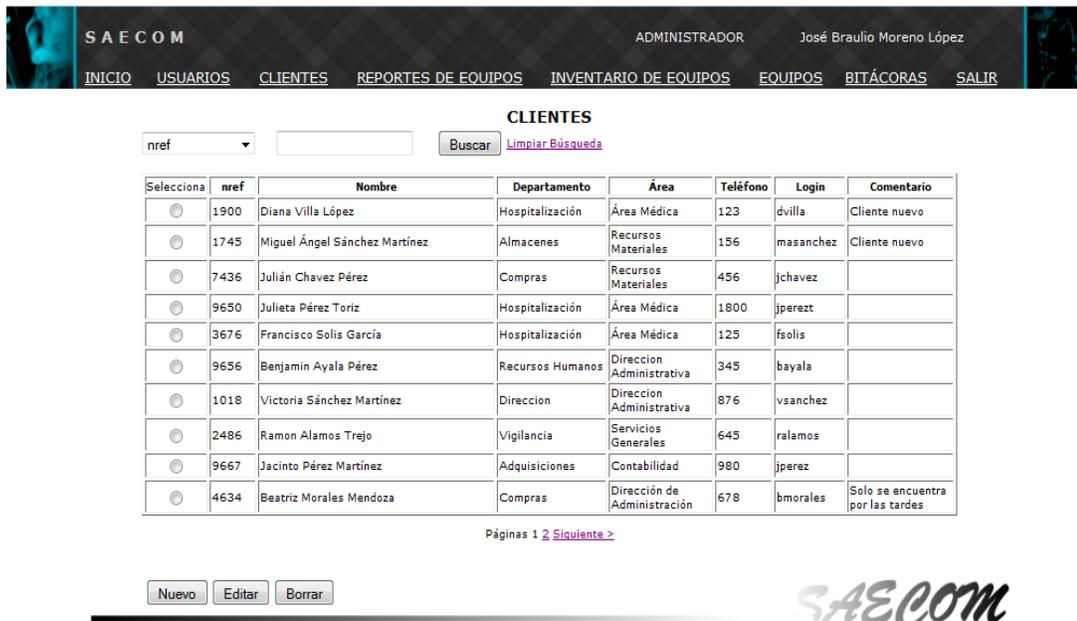


Figura 6.20 Cliente agregado

6.5 PRUEBAS DE FUNCIONALIDAD

El sistema sólo permite el acceso a usuarios que se encuentran registrados.



Figura 6.21 Pantalla de logue

El sistema sólo permite el acceso según el perfil del usuario.



Figura 6.22 Perfil de administrador

6.6 PRUEBAS DE VOLUMEN

Descripción

Verificar que la aplicación funciona adecuadamente bajo los siguientes escenarios de volumen:

- Determinan la cantidad de datos con la cual el sistema falla.
- Carga máxima que el sistema soporta en un período dado.
- Es una prueba costosa (en tiempo de máquina y personal), se debe tratar de no exceder los límites.

Técnica

Usar múltiples clientes, corriendo las mismas pruebas o pruebas complementarias para producir el peor caso de volumen por un período extendido.

Utilizar un tamaño máximo de Base de datos (actual o con datos representativos) y múltiples clientes para correr consultas simultáneamente para períodos extendidos.

Se realizaron ocho conexiones simultáneas con diferentes perfiles.

- ✓ Dos sesiones iniciadas como administrador
- ✓ Tres sesiones como clientes
- ✓ Dos sesiones iniciadas como Ingeniero/Técnico
- ✓ Una sesión iniciada como Mesa de Ayuda.

CAPÍTULO 7 DOCUMENTACIÓN Y RESULTADOS

7.1 CREACIÓN DE LA DOCUMENTACIÓN DEL SISTEMA

El sistema de administración de equipos de cómputo pretende llevar un inventario de los equipos de cómputo que se tienen en una institución, empresa u organización. Tener un control sobre dichos equipos ya que el sistema nos almacena la siguiente información.

- ✓ Usuario responsable del equipo
- ✓ Cuantos equipos tiene bajo reguardo dicho usuario
- ✓ Donde se encuentra ubicado dicho equipo
- ✓ Que fallas a presentado el equipo
- ✓ IP y hostname del equipo

El sistema permite al usuario del equipo levantar un reporte sobre la misma vía web o vía telefónica (Mesa de ayuda).

El sistema cuenta con el perfil Mesa de Ayuda que permite crear reportes los cuales son atendidos por personal calificado (soporte técnico)

El sistema cuenta con un perfil Soporte Técnico en el cual permite al personal encargado de proporcionar dicho servicio llevar de manera ordenada los reportes que se generan por el usuario o la Mesa de Ayuda.

7.2 CREACIÓN DEL MANUAL DE OPERACIÓN DEL SISTEMA

El sistema tiene por nombre SISTEMA DE ADMINISTRACIÓN DE EQUIPOS DE CÓMPUTO, es recomendable que esté bajo la administración del departamento de informática o de tecnologías de la información.



Figura 7.1 Pantalla de inicio de saecom

7.3 CREACIÓN DEL MANUAL USUARIO

Manual del Usuario

Introducción

El sistema estará estructurado en base a perfiles, los cuales permitirán el acceso a ciertos módulos de la aplicación dependiendo de las actividades que el usuario desarrolle dentro de la institución.

Inicialmente se contemplará cuatro tipos de perfiles.

- Administrador
- Ingeniero / Técnico
- Cliente
- Mesa de ayuda

Perfil	Actividad
Administrador	Será el encargado de crear cuentas de usuarios soporte y mantenimiento a la base de datos.
Ingeniero/Técnico	Tendrá acceso a la mayoría de las características que tenga el sistema, esto es, agregar información sobre los equipos de cómputo nuevos o bajas, mantenimientos preventivos o correctivos. Dicha información se irá anexando al historial del equipo.
Cliente	Sólo agregará cierta información; levantar reportes, comentarios.
Mesa de Ayuda	El perfil mesa de ayuda permite agregar reportes que se hacen vía telefónica.

El departamento encargado de la administración y puesta en marcha del sistema proporcionará una dirección web la cual al digitarla en nuestro navegador visualizaremos la ventana de acceso al sistema



Figura 7.2 Pantalla de acceso del sistema

Administrador

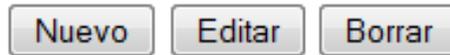
Cuenta con todos los privilegios del sistema la sesión se inicia con la siguiente pantalla. En la parte superior se visualizan todas a las opciones del sistema y a las cuales se pueden acceder dando un clic en la opción deseada.



Figura 7.3 Perfil administrado

Opción INICIO

El usuario puede publicar avisos para ser vistos por los demás usuarios al iniciar sesión. Se cuenta con tres botones Nuevo, Editar y borrar



El botón Nuevo nos permite crear un aviso y seleccionar al usuario que visualizará dicho aviso.

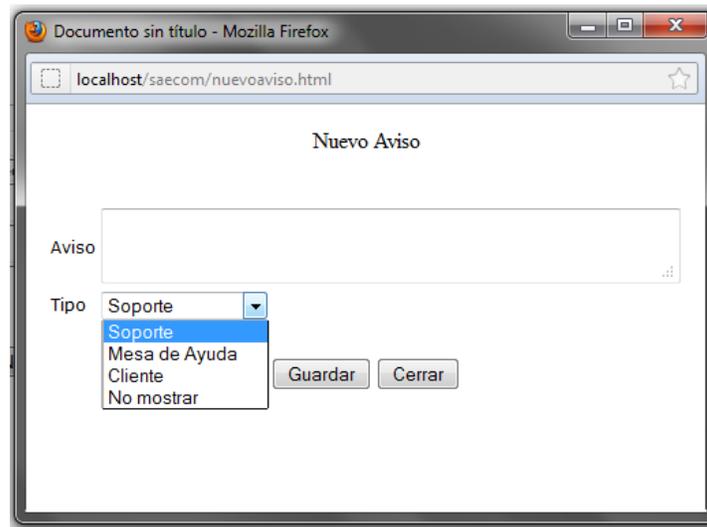
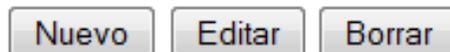


Figura 7.4 Pantalla para crear nuevo aviso

Podemos editar o borrar dicho aviso seleccionándolo y dando clic en el botón Editar o Borrar según sea el caso.

Opción USUARIOS

El administrador da de ALTA al personal de soporte o de mesa de ayuda u otro administrador. Se cuenta con tres botones



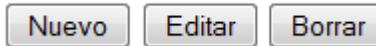
El botón Nuevo nos permite asignar usuario y contraseña personal antes mencionado.

Figura 7.5 Pantalla para crear nuevo usuario

Podemos editar o borrar los datos del personal seleccionándolo y dando clic en el botón Editar o Borrar según sea el caso.

Opción CLIENTES

El administrador del sistema puede dar de alta si es necesario a los clientes. Se cuenta con tres botones



El botón Nuevo nos permite asignar usuario y contraseña al cliente.

Figura 7.6 Pantalla para crear nuevo cliente

Podemos editar o borrar los datos del cliente seleccionándolo y dando clic en el botón Editar o Borrar según sea el caso.

Opción REPORTES DE EQUIPOS

El administrador del sistema puede crear reportes si es necesario. Se cuenta con cuatro botones.



El botón Nuevo nos permite crear un nuevo reporte. Nos arroja una ventana donde podemos buscar al cliente para visualizar sus datos y así el levantamiento del reporte sea más rápido.



Figura 7.7 Pantalla para generar un reporte

Realizamos la búsqueda del cliente una vez encontrado, seleccionamos y damos un clic en el botón generar



Figura 7.8 Reporte generado

Visualizamos la información del cliente rápidamente.

Figura 7.9 Generando un reporte

Podemos editar o borrar el reporte seleccionándolo y dando clic en el botón Editar o Borrar según sea el caso.

El botón Mostrar nos permite visualizar el reporte e imprimirlo seleccionando el reporte y dando clic en dicho botón.

Opción INVENTARIO DE EQUIPOS

El administrador del sistema puede agregar un registro al inventario. Se cuenta con cuatro botones.



El botón de Asignar nos arroja la siguiente ventana que nos permite

- Agregar clientes y asignarles equipos
- Buscar clientes y agregar editar o eliminar equipos asignados
- Buscar clientes y verificar que equipos tiene asignados

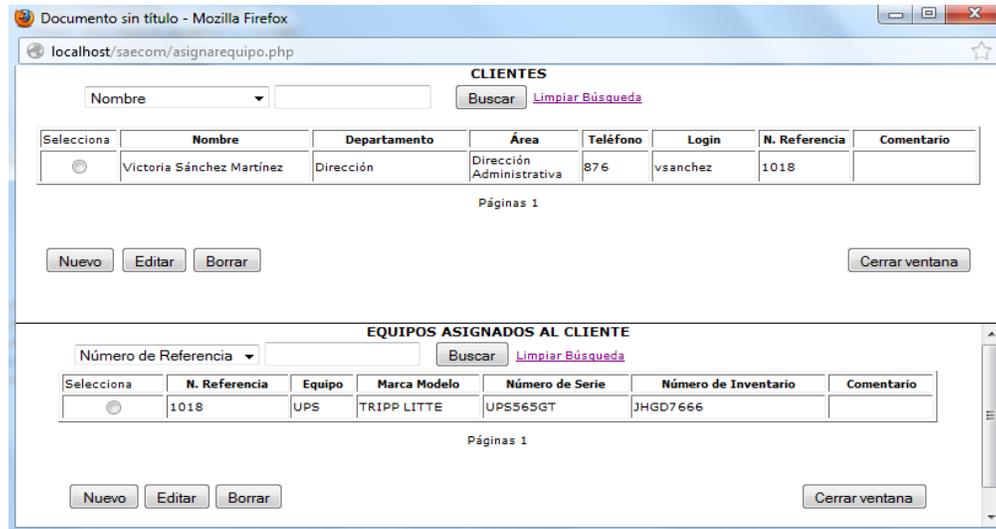


Figura 7.10 Pantalla para asignar equipos a los clientes

Podemos editar o borrar el registro de inventario dando clic en el botón Editar o Borrar según sea el caso.

El botón Mostrar nos permite visualizar los equipos asignados al cliente e imprimir el formato de resguardo, seleccionando el cliente y dando clic en dicho botón.



Figura 7.11 Inventario por cliente

S A E C O M
RESGUARDO DE EQUIPO

Número de Referencia 1018

Se hace entrega de equipo a Victoria Sanchez Martinez en el departamento de Direccion ubicado en 7to Piso Edificio G y pertenece al área de Direccion Administrativa. Teléfono 878

Usuario vsanchez Contraseña 8172r7a2

El equipo se enlista a continuación

Equipo	Marca y Modelo	Número de Serie	Número de Inventario
UPS	TRIPP LITTE	LPS666GT	JHGD7888

El equipo entregado queda bajo resguardo del Cliente cualquier falla o desperfecto lo puede reportar con el número de referencia asignado al Departamento XXXXXXXX al Número XXXXXXXX, o bien podrá levantar un reporte desde la página <http://www.xxxx> utilizando su usuario y contraseña asignado.

Victoria Sanchez Martinez

Ing Javier Gómez Salinas
Jefe de Departamento de XXXXXXXXXX

Ing Israel Zamorano Rojas
Sujeto del Departamento de XXXXXXXXXX

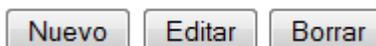
Fecha: 27/04/2012 Hora: 22:00:47

Figura 7.12 Formato de resguardo de equipo generado por el sistema

Opción Equipos

El administrador del sistema puede agregar registros de equipos si es necesario con la finalidad de que al momento de asignar equipos al cliente únicamente se realice la búsqueda del equipo y se le asigne al cliente agilizando el proceso.

Se cuenta con tres botones



El botón de Nuevo permite al administrador agregar un registro de un equipo.

Podemos editar o borrar el registro del equipo seleccionado, dando clic en el botón Editar o Borrar según sea el caso.

Opción BITÁCORA

Permite al administrador visualizar los inicios de sesión de los usuarios, del personal así como de los clientes

login	Actividad	Fecha	Hostname	Ip
admin	inicio sesion	2012-09-30 05:12:05	redis	127.0.0.1
mesa	cerro sesion	2012-09-30 05:11:59	redis	127.0.0.1
mesa	inicio sesion	2012-09-30 05:07:56	redis	127.0.0.1
admin	cerro sesion	2012-09-30 05:07:46	redis	127.0.0.1
admin	inicio sesion	2012-09-30 04:57:22	redis	127.0.0.1
mesa	inicio sesion	2012-09-30 04:56:29	redis	127.0.0.1
admin	cerro sesion	2012-09-30 04:56:24	redis	127.0.0.1
admin	inicio sesion	2012-09-30 04:43:34	redis	127.0.0.1
cliente	cerro sesion	2012-09-30 04:43:29	redis	127.0.0.1
cliente	inicio sesion	2012-09-30 04:42:30	redis	127.0.0.1



Figura 7.13 Bitácora

Opción SALIR

Salir del sistema

Mesa de Ayuda

Para este perfil tiene la opción de INICIO donde visualiza los avisos que el administrador publicó y la opción de REPORTE

REPORTES DE EQUIPOS

Número de Reporte: [Limpiar Búsqueda](#) [ACTUALIZAR](#)

Selecciona	Número de Reporte	Número de Referencia	Cliente	Descripción	Fecha	Estado	Atendio
<input type="radio"/>	25	9656	Benjamin Ayala Pérez	se apaga	0000-00-00 00:00:00	abierto	José Braulio Moreno López
<input type="radio"/>	24	9667	Jacinto Pérez Martínez	No hay internet	2012-05-11 23:50:47	terminado	Cruz Sergio Aguilar Diaz
<input type="radio"/>	21	9667	Jacinto Pérez Martínez	no sirve teclado	2012-05-11 03:46:29	abierto	Mesa de Ayuda
<input type="radio"/>	20	9667	Jacinto Pérez Martínez	no tengo internet	2012-05-11 03:37:10	abierto	Mesa de Ayuda
<input type="radio"/>	13	4634	Beatriz Morales Mendoza	no prende	2012-05-11 00:24:36	abierto	Mesa de Ayuda
<input type="radio"/>	12	9439	Carlos Garcia Blanco	ghjgh	2012-05-11 00:23:28	abierto	Mesa de Ayuda
<input type="radio"/>	10	6495	Jose Moreno López	hjhkgdhjas	2012-05-05 01:31:19	abierto	Mesa de Ayuda
<input type="radio"/>	8	5600	Julietta Pérez Tucan	Se reinicia equipo	0000-00-00 00:00:00	abierto	José Braulio Moreno López
<input type="radio"/>	7	5805	Tómas Mendoza Villegas	no imprime	0000-00-00 00:00:00	abierto	José Braulio Moreno López
<input type="radio"/>	6	9656	Benjamin Ayala Pérez	Se apaga monitor	0000-00-00 00:00:00	abierto	José Braulio Moreno López

Páginas 1 2 [Siguiente >](#)

Figura 7.14 Perfil mesa de ayuda

Opción REPORTE DE EQUIPOS

Funciona igual que la opción de REPORTE DE EQUIPOS en el perfil de administrador sólo que el perfil Mesa de Ayuda no puede eliminar los registros

REPORTES DE EQUIPOS

Número de Reporte: [Limpiar Búsqueda](#) [ACTUALIZAR](#)

Selecciona	Número de Reporte	Número de Referencia	Cliente	Descripción	Fecha	Estado	Atendio
<input type="radio"/>	25	9656	Benjamin Ayala Pérez	se apaga	2012-05-12 01:00:45	abierto	Mesa de
<input type="radio"/>	24						
<input type="radio"/>	23						
<input type="radio"/>	22						
<input type="radio"/>	21						
<input type="radio"/>	20						
<input type="radio"/>	19						
<input type="radio"/>	18						
<input type="radio"/>	17						
<input type="radio"/>	16						

GENERAR REPORTE

Nombre: [Limpiar Búsqueda](#)

Selecciona	Nombre	Departamento	Área	Teléfono	Número de Referencia
<input type="text"/>					

Figura 7.15 Reporte de equipos perfil mesa de ayuda

Soporte

Para este perfil tiene la opción de INICIO donde visualiza los avisos que el administrador publicó, las opciones de REPORTE DE EQUIPOS, INVENTARIO DE EQUIPOS Y EQUIPOS.

Opción REPORTE DE EQUIPOS

El perfil soporte permite crear reportes de la misma forma que el perfil administrador solo que no permite eliminar los registros.

The screenshot shows the SAECOM support system interface. At the top, there is a navigation menu with options: INICIO, USUARIOS, CLIENTES, REPORTE DE EQUIPOS, INVENTARIO DE EQUIPOS, EQUIPOS, BITÁCORAS, and SALIR. The user is logged in as ADMINISTRADOR José Braulio Moreno López. The main section is titled 'REPORTE DE EQUIPOS' and includes a search bar with 'Número de Reporte' and buttons for 'Buscar', 'Limpiar Búsqueda', and 'ACTUALIZAR'. Below the search bar is a table with the following data:

Selección	Número de Reporte	Número de Referencia	Cliente	Descripción	Fecha	Estado	Atendio
<input type="radio"/>	25	9656	Benjamin Ayala Pérez	se apaga	2012-06-12 09:21:40	abierto	José Braulio Moreno López
<input type="radio"/>	24	9667	Dacinto Pérez Martínez	No hay internet	2012-05-11 23:50:47	terminado	Cruz Sergio Aguilar Díaz
<input type="radio"/>	20	9667	Dacinto Pérez Martínez	no tengo internet	2012-05-11 03:37:10	abierto	Mesa de Ayuda
<input type="radio"/>	13	4634	Beatriz Morales Mendoza	no prende	2012-05-11 00:24:36	abierto	Mesa de Ayuda
<input type="radio"/>	12	9439	Carlos Garcia Blanco	Nobreak emitiendo ruido	2012-06-20 15:00:22	abierto	José Braulio Moreno López
<input type="radio"/>	10	6495	Carlos Ventura Juárez	No imprime	2012-08-07 12:05:00	abierto	José Braulio Moreno López
<input type="radio"/>	8	5600	Julieta Pérez Tucan	Se reinicia equipo	0000-00-00 00:00:00	abierto	José Braulio Moreno López
<input type="radio"/>	7	5805	Tómas Mendoza Vitegas	no imprime	2012-04-16 17:55:00	abierto	José Braulio Moreno López
<input type="radio"/>	6	9656	Benjamin Ayala Pérez	Se apaga monitor	2012-05-14 13:22:00	abierto	José Braulio Moreno López

Below the table, there are buttons for 'Nuevo', 'Editar', 'Borrar', and 'Mostrar'. The SAECOM logo is visible in the bottom right corner.

Figura 7.16 Reporte de equipos perfil administrador

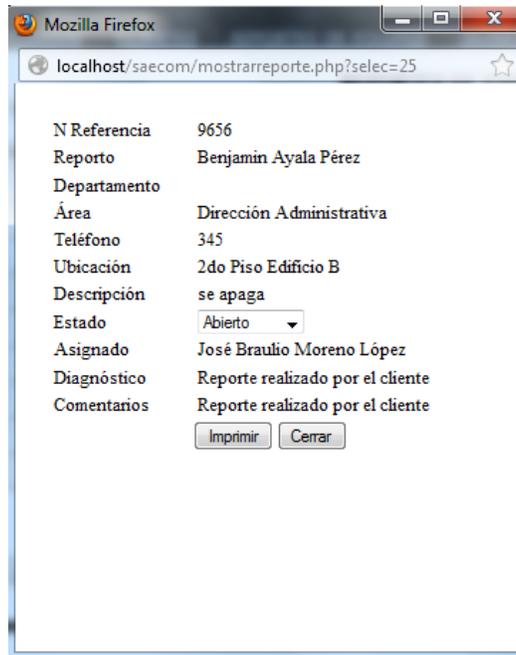
El botón de Atender nos muestra una ventana donde permite especificar el diagnóstico o la reparación que se le realizó al equipo. Seleccionamos el cliente y damos clic en dicho botón.

The screenshot shows a Mozilla Firefox browser window displaying the 'editarreportesoporte.php?selec=25' form. The form contains the following fields and values:

- N Referencia: 9656
- Reporto: Benjamin Ayala Pérez
- Departamento: Dirección Administrativa
- Área: 345
- Teléfono: 2do Piso Edificio B
- Ubicación: se apaga
- Descripción: Abierto
- Estado: Abierto
- Asignado: José Braulio Moreno López
- Diagnóstico: Reporte realizado por el cliente
- Comentarios: Reporte realizado por el cliente

At the bottom of the form, there are 'Guardar' and 'Cerrar' buttons.

El botón de Mostrar permite visualizar la información correspondiente al reporte creado. Seleccionamos el cliente y damos clic en dicho botón.



Opción INVENTARIO DE EQUIPOS

El perfil soporte permite crear registros en el inventarios de la misma forma que el perfil administrador sólo que no permite eliminar dichos registros.



Figura 7.17 Inventario de equipos perfil administrador

Opción EQUIPOS

El perfil soporte permite crear registros de equipos de la misma forma que el perfil administrador solo que no permite eliminar dichos registros.

Selección	Equipo	Marca Modelo	Número de Serie	Número de Inventario	Comentario
<input type="radio"/>	Monitor	hp	S46564f	4356fghf	Recursos humanod
<input type="radio"/>	Escaner	DELL S6700	TYU655	ESC765N	
<input type="radio"/>	UPS	TRIPP LITTE	UPS565GT	JHGD7666	
<input type="radio"/>	UPS	TRIPP LITTE	UYU1122	UPSTY6	
<input type="radio"/>	Monitor	Astra 5500	UJHY6590	MON432	
<input type="radio"/>	Monitor	Astra 5500	MOKI8765	MON549	
<input type="radio"/>	Monitor	Astra 5500	YHNG1265	MON876	
<input type="radio"/>	Impresora	HP Lasejet 987	OKJ876	IMP675	
<input type="radio"/>	Impresora	HP Lasejet 987	UIOP123	IMP1267	
<input type="radio"/>	CPU	HP TL3500	ERTY3454	QWE712	

SAECOM

Figura 7.18 Equipos perfil administrador

Cliente

Para este perfil tiene la opción de INICIO donde visualiza los avisos que el administrador público y la opción de REPORTAR EQUIPO

Opción REPORTAR EQUIPO

El perfil cliente puede crear su propio reporte desde el sistema o vía telefónica. Desde el sistema selecciona la opción REPORTAR EQUIPO. Automáticamente se visualizan los reportes que ha generado el cliente.

Número de Reporte	Número de Referencia	Cliente	Descripción	Fecha	Estado	Diagnóstico
24	9667	Jacinto Pérez Martínez	No hay internet	2012-05-11 23:50:47	terminado	Cable desconectado
20	9667	Jacinto Pérez Martínez	no tengo internet	2012-05-11 03:37:10	abierto	Reporte realizado por el cliente

SAECOM

Figura 7.19 reportar equipo perfil cliente

Como sus datos ya se tienen el sistema, en el cliente sólo tendrá que describir la falla del equipo y dar clic en enviar para general el reporte.

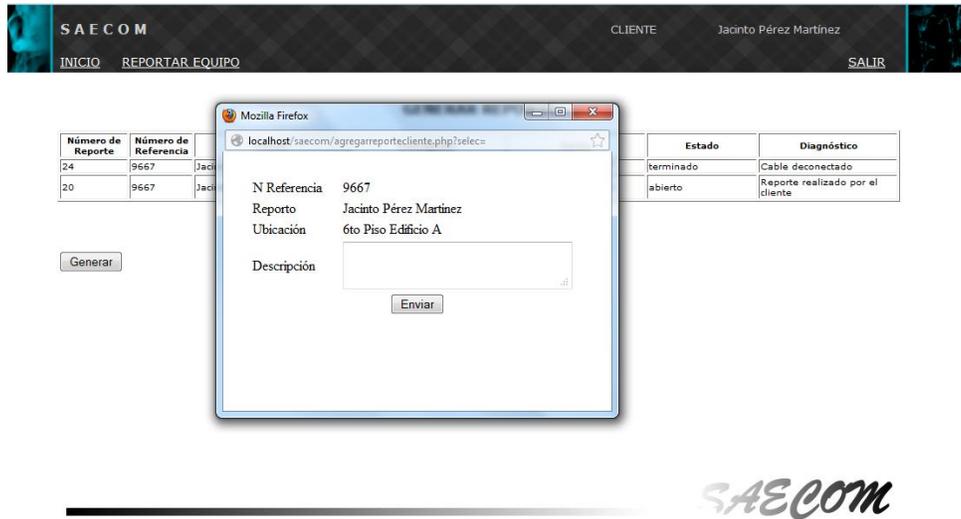


Figura 7.20 Generando reporte perfil cliente

7.4 RESULTADOS DEL SISTEMA

El sistema busca ordenar, mantener y facilitar el manejo de la información, con la finalidad ofrecer un mejor servicio a quien lo requiera. Además de contar con la información organizada y estructurada de tal manera que siempre se encuentre disponible cada que se requiera.

CONCLUSIONES

En la actualidad existen un gran número de herramientas de las cuales podemos hacer uso, con las cuales podemos implementar alguna aplicación que optimice nuestras actividades laborales.

Muchas de ellas son de uso libre y algunas tienen costo pero finalmente podemos adquirir y conocer dichas herramientas mediante internet.

El sistema cumplió con los objetivos específicos ya que nos proporciona.

- Una interfaz de fácil manejo
- Proporciona varios niveles de seguridad de acuerdo al perfil
- La información siempre está disponible, segura y protegida
- Además de proporcionar reportes generales de manera particular

Los conocimientos que he aplicado para la realización de esta tesis los obtuve a lo largo de mi formación universitaria, en las asignaturas de Bases de Datos, Ingeniería de Software, Computadoras y Programación, Sistemas Operativos, Ingeniería de Programación, Redes de Computadoras.

Al realizar este sistema me he dado cuenta que es muy importante entender claramente las necesidades del usuarios final ya que este será quien utilice el sistema.

Este trabajo me permitió aplicar conocimientos adquiridos durante la carrera, así como reforzar y conocer algunos que no tenía claro o bien que desconocía.

Por otra parte creo que la formación adquirida durante la carrera nos hace ser más analíticos, y por consiguiente tratar de siempre obtener los mayores y mejores resultados cuando los aplicamos en nuestras actividades personal y profesionales.

El sistema para la administración de equipos de cómputo agiliza el registro y permite consultar la información que se ha generando durante el periodo de uso.

Seguridad en la información, el usuario sólo puede ingresar al sistema si se encuentra registrado y dicho usuarios sólo podrá consultar o modificar la información dependiendo del perfil que le sea asignado.

Sólo se almacena la información correcta cuidando así la integridad de ésta.

El sistema proporciona información mediante reportes generales y particulares.

GLOSARIO

La Biblioteca de Infraestructura de Tecnologías de Información (ITIL), frecuentemente abreviada **ITIL** (del inglés Information Technology Infrastructure Library), es un conjunto de conceptos y prácticas para la gestión de servicios de tecnologías de la información, el desarrollo de tecnologías de la información y las operaciones relacionadas con la misma en general. ITIL da descripciones detalladas de un extenso conjunto de procedimientos de gestión ideados para ayudar a las organizaciones a lograr calidad y eficiencia en las operaciones de TI. Estos procedimientos son independientes del proveedor y han sido desarrollados para servir como guía que abarque toda infraestructura, desarrollo y operaciones de TI.

Base de Datos (DB).- Cualquier conjunto de datos organizados para su almacenamiento en la memoria de un ordenador o computadora, diseñado para facilitar su mantenimiento y acceso de una forma estándar. Los datos suelen aparecer en forma de texto, números o gráficos. Hay cuatro modelos principales de bases de datos: el modelo jerárquico, el modelo en red, el modelo relacional (el más extendido hoy en día).

Base de Datos Relacional (DBR).- Tipo de base de datos o sistema de administración de bases de datos, que almacena información en tablas (filas y columnas de datos) y realiza búsquedas utilizando los datos de columnas especificadas de una tabla para encontrar datos adicionales en otra tabla.

Datos Elementales.- Un dato elemental, tal como indica su nombre, es una pieza elemental de información. El primer paso en el diseño de una base de datos debe ser un análisis detallado y exhaustivo de los datos elementales requeridos.

Campos y Subcampos.- Los datos elementales pueden ser almacenados en campos o en subcampos. Un campo es identificado por un rótulo numérico que se define en la FDT de la base de datos. A diferencia de los campos, los subcampos no se identifican por medio de un rótulo, sino por un delimitador de subcampo.

Delimitador de Subcampo.- Un delimitador de subcampo es un código de dos caracteres que precede e identifica un subcampo de longitud variable dentro de un campo.

DBMS: Data Base Management System (SISTEMA DE MANEJO DE BASE DE DATOS).- Consiste de una base de datos y un conjunto de aplicaciones (programas) para tener acceso a ellos.

Modelo de Datos.- es un conjunto de herramientas conceptuales para describir los datos, las relaciones entre ellos, su semántica y sus limitantes.

Redundancia.- Esta se presenta cuando se repiten innecesariamente datos en los archivos que conforman la base de datos.

Inconsistencia.- Ocurre cuando existe información contradictoria o incongruente en la base de datos.

Dificultad en el Acceso a los Datos.- Debido a que los sistemas de procesamiento de archivos generalmente se conforman en distintos tiempos o épocas y ocasionalmente por distintos programadores, el formato de la información no es uniforme y se requiere de establecer métodos de enlace y conversión para combinar datos contenidos en distintos archivos.

Aislamiento de los Datos.- Se refiere a la dificultad de extender las aplicaciones que permitan controlar a la base de datos, como pueden ser, nuevos reportes, utilerías y demás debido a la diferencia de formatos en los archivos almacenados.

Anomalías en el Acceso Concurrente.- Ocurre cuando el sistema es multiusuario y no se establecen los controles adecuados para sincronizar los procesos que afectan a la base de datos. Comúnmente se refiere a la poca o nula efectividad de los procedimientos de bloqueo.

Problemas de Seguridad.- Se presentan cuando no es posible establecer claves de acceso y resguardo en forma uniforme para todo el sistema, facilitando así el acceso a intrusos.

Problemas de Integridad.- Ocurre cuando no existe a través de todo el sistema procedimientos uniformes de validación para los datos.

Nivel Físico.- Es aquel en el que se determinan las características de almacenamiento en el medio secundario. Los diseñadores de este nivel poseen un amplio dominio de cuestiones técnicas y de manejo de hardware.

Nivel Conceptual.- Es aquel en el que se definen las estructuras lógicas de almacenamiento y las relaciones que se darán entre ellas. Ejemplos comunes de este nivel son el diseño de los registros y las ligas que permitirán la conexión entre registros de un mismo archivo, de archivos distintos incluso, de ligas hacia archivos.

Document Object Model o DOM.- ('Modelo de Objetos del Documento' o 'Modelo en Objetos para la Representación de Documentos') es esencialmente una interfaz de programación de aplicaciones (API) que proporciona un conjunto estándar de objetos para representar documentos HTML y XML, un modelo estándar sobre cómo pueden combinarse dichos objetos, y una interfaz estándar para acceder a ellos y manipularlos. A través del DOM, los programas pueden acceder y modificar el contenido, estructura y estilo de los documentos HTML y XML, que es para lo que se diseñó principalmente.

Diccionario de datos (DD).- Cconjunto de metadatos que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.

ISO.-La Organización Internacional para la Estandarización es una federación de alcance mundial integrada por cuerpos de estandarización nacionales de 130 países, uno por cada país. Es una organización no gubernamental establecida en 1947. La misión de la ISO es promover el desarrollo de la estandarización y las actividades con ella relacionada en el mundo con la mira en facilitar el intercambio de servicios y bienes, y para promover la cooperación en la esfera de lo intelectual, científico, tecnológico y económico.

BIBLIOGRAFÍA

Ingeniería del Software: Un Enfoque Práctico, Roger Pressman, Sexta Edición
Sistemas Operativos: Andrew S. Tanenbaum Pearson Educación, 2003
PHP y MySQL: Tecnología para el desarrollo de aplicaciones web. Ángel Cobo

Pfleeger S., "Ingeniería de Software, Teoría y Práctica" - Primera Edición Editorial Prentice Hall - 2002.

Stallings, William SISTEMAS OPERATIVOS. Prentice Hall, 2da edición
Piattini Mario, Adoración de Miguel, Marcos Esperanza.

DISEÑO DE BASES DE DATOS RELACIONALES. Ed. Alfaomega

GUÍA DE APRENDIZAJE: MYSQL Larry Ullman (Pearson Educación) Idioma: Español 352 páginas

Welling, Luke; Thomson, Laura (ANAYA MULTIMEDIA) 976 páginas. Idioma: Español

Creación de un portal con PHP y MySQL Pavón Puertas, Jacobo, (aut.)
Ra-Ma Editorial, S.A. 1ª ed., 1ª imp.(06/2004) 240 páginas; Idiomas: Español

Creación de sitios web con PHP Gil Rubio, Javier; Yagüe Panadero, Agustín; Tejedor Cerbel, Jorge & Alonso Villaverde, Santiago (Editorial McGraw-Hill) 592 páginas.

PHP a través de ejemplos GUTIÉRREZ, A. y BRAVO, G. (Ra-ma) 576 páginas.

H. M. Deitel. Introducción a los Sistemas Operativos. Addison-Wesley Iberoamericana, México, 1987

A. S. Tanenbaum. Sistemas Operativos Modernos. Prentice Hall Hispanoamericana, S.A., México, 1993.

MESOGRAFÍA

- 1 [http://es.wikipedia.org/wiki/Mesa de ayuda](http://es.wikipedia.org/wiki/Mesa_de_ayuda)
- 2 [http://es.wikipedia.org/wiki/Document Object Model](http://es.wikipedia.org/wiki/Document_Object_Model)
- 3 <http://www.php.net/manual/en/preface.php>
- 4 <http://www3.uji.es/~mmarques/f47/apun/node68>
- 5 [http://es.wikipedia.org/wiki/Seguridad inform%C3%A1tica](http://es.wikipedia.org/wiki/Seguridad_inform%C3%A1tica)
- 6 [http://es.wikipedia.org/wiki/Navegador web](http://es.wikipedia.org/wiki/Navegador_web)
- 7 <http://es.wikipedia.org/wiki/Antivirus>
- 8 [http://es.wikipedia.org/wiki/Navegador web](http://es.wikipedia.org/wiki/Navegador_web)