



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE CIENCIAS

Implementando la Factura Digital mediante SCRUM

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

Licenciado en Ciencias de la Computación

P R E S E N T A :

Pérez León Miguel Angel



**DIRECTOR DE TESIS:
M. en C. María Guadalupe Elena Ibargüengoitia
González
2013**



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.....	4
1 Conceptos Básicos.....	5
1.1 Facturación Digital.....	5
1.2 Elementos indispensables de la Factura Electrónica.....	6
1.2.1 Firma Electrónica Avanzada (FEA).....	6
1.2.2 Comprobante fiscal digital (CFD).....	7
1.2.3 Cadena Original.....	7
1.2.4 Extensible Markup Language (XML).....	7
1.2.5 Extensible Stylesheet Language Transformations (XSLT).....	8
1.2.6 Portable Document Format (PDF).....	8
1.3 Metodologías.....	8
1.4 SCRUM.....	9
2 Open Enterprise Resource Planning (OpenERP).....	11
2.1 Definición.....	11
2.1.1 Ventajas de OpenERP.....	12
2.1.1.1 Modularidad.....	13
2.1.2 Desventajas de OpenERP.....	13
2.2 Diseño y Arquitectura de OpenERP.....	13
2.3 Diagrama de un sistema tipo OpenERP.....	14
2.4 Componentes.....	15
2.4.1 Python.....	15
2.4.2 Arquitectura de OpenERP.....	15
2.4.3 Sistema Manejador de Bases de Datos (SMBD).....	16
2.4.3.1 Ventajas PostgreSQL.....	16
2.5 Instalación.....	16
2.5.1 Cliente-Servidor.....	17
2.5.2 Sobre OpenERP-Ubuntu.....	17
3 Planteamiento del problema.....	18
3.1 Contexto general.....	18
3.2 Definición del problema.....	21
4 Instalación del ambiente de desarrollo.....	22
4.1 Especificación de requerimientos.....	22
4.2 Análisis del módulo.....	22
4.2.1 Definición de Actores y Casos de Uso.....	22
4.3 Ciclo 0: Instalación OpenERP.....	25
4.4 Ciclo 1: Administrar Base de Datos.....	38
5 Implementación de las funciones del módulo SiFaDig.....	49
5.1 Ciclo 2: Administrar Usuarios.....	49
5.2 Ciclo 3: Modificación de Usuarios y Permisos.....	57
5.3 Ciclo 4: Revisión de Ciclos y Product BackLog.....	74
5.4 Ciclo 5: Administración de Productos.....	75
5.5 Ciclo 6: Administración de Clientes.....	89
5.6 Ciclo 7: Administración de Ventas.....	98
5.7 Ciclo 8: Instalación del módulo de Facturación Electrónica.....	109
Conclusiones.....	119
Bibliografía.....	120

Implementando la Factura Digital mediante SCRUM

1.Datos del alumno	1.Datos del alumno
Apellido paterno	Pérez
Apellido materno	León
Nombre(s)	Miguel Angel
Teléfono	55-20-69-96-50
Universidad Nacional Autónoma de México	Universidad Nacional Autónoma de México
Facultad de Ciencias	Facultad de Ciencias
Carrera	Ciencias de la Computación
Número de cuenta	301081223
2, Datos del tutor	2, Datos del tutor
Grado	M. en C.
Nombre(s)	María Guadalupe Elena
Apellido paterno	Ibargüengoitia
Apellido materno	González
3, Datos del sinodal 1	3, Datos del sinodal 1
Grado	Mat.
Nombres(s)	Manuel
Apellido paterno	López
Apellido materno	Mateos
4, Datos del sinodal 2	4, Datos del sinodal 2
Grado	L. en C. C.
Nombres(s)	Sonia Josefina
Apellido paterno	Valery
Apellido materno	Lobo
5, Datos del sinodal 3	5, Datos del sinodal 3
Grado	M. en C.
Nombres(s)	Gustavo Arturo
Apellido paterno	Márquez
Apellido materno	Flores
6, Datos del sinodal 4	6, Datos del sinodal 4
Grado	M. en A.
Nombres(s)	Karla
Apellido paterno	Ramírez
Apellido materno	Pulido
7. Datos del trabajo escrito	7. Datos del trabajo escrito
Título	Implementando la Facturación Digital mediante SCRUM
Subtítulo	Implementando la Facturación Digital mediante SCRUM
Número de páginas	120 p
Año	2013

Introducción

Esta tesis tiene como objetivo implementar paso a paso, mediante SCRUM¹ un sistema de facturación digital, que pueda ofrecer una solución a las empresas que necesiten adoptar la facturación digital dentro de sus estándares, así como mostrar una guía para todo aquel que este interesado en incluir SCRUM dentro de sus prácticas laborales.

Como principales objetivos en esta tesis se tiene:

- ♣ Desarrollar el módulo de facturación electrónica.
- ♣ Mostrar cómo es que SCRUM es una metodología ágil y es un verdadero aliado para la gestión de proyectos.
- ♣ Mediante el uso de OpenERP² mostrar los elementos que intervienen en la generación de facturas electrónicas. A lo largo de esta tesis se mostrará cómo haciendo uso de software libre y de herramientas al alcance de cualquier usuario estándar, es posible generar un sistema que sea capaz de administrar los recursos de una empresa, además de permitir generar facturas electrónicas con base en las ventas generadas a través del sistema.

Es bien conocido que el marco que rige la contabilidad de cada país es muy diferente y mas aún en el caso de un OpenERP que pretende ser utilizado por cualquier usuario sin importar su nacionalidad. Sin embargo, en el caso particular de este trabajo, nos enfocaremos en el marco legal y gubernamental que establece el SAT³ para poder genera una factura electrónica que sea válida.

De manera mas detallada también se analizarán los componentes principales que integran la estructura de OpenERP, tales como su arquitectura, diseño e implementación y que esto sirva como base para que, en caso de ser necesario, se desarrolle o modifique de manera sencilla cualquier módulo que así lo requiera.

Para lograr los puntos antes mencionados haremos uso de SCRUM que es un método ágil de desarrollo de software para poder cumplir con los objetivos en tiempo y forma, además de mantener un orden adecuado y controlar de manera clara y detallada el tiempo y los recursos de los cuales se dispone. Al inicio de esta tesis se revisa de manera somera los componentes de SCRUM, cómo funcionan y como interactúan entre sí.

La estructura de esta tesis se divide en 2 bloques principales, en el primero de ellos se realiza una pequeña introducción a los conceptos básicos de factura digital y de SCRUM y a los elementos que integran OpenERP, además de revisar algunas de sus ventajas y desventajas. También en esta primer bloque se ilustra cómo generar el entorno necesario para que el sistema se desempeñe de manera óptima, además de prepara el entorno de desarrollo en caso de ser necesario. El segundo bloque inicia en el capítulo 4 y este está dividido en ciclos, cada ciclo define una porción del sistema que amerita ser analizado de manera mas detallada. Al inicio de cada ciclo se indican las tareas que comprende dicho ciclo, además de indicarse la cantidad de tiempo y esfuerzo que se requiere y la cantidad de tareas remanentes. La cantidad de esfuerzo es un estimado del grado de complejidad que involucra cada tarea. Estos elementos son la base de SCRUM y además nos sirven para poder generar un diagnóstico claro y una retrospectiva del trabajo realizado. Por último y antes de iniciar de lleno con el contenido de la tesis, es importante recalcar la importancia del Product BackLog, ya que este documento es la columna vertebral de SCRUM y sin este no tiene sentido hacer uso de esta metodología.

¹ Metodología ágil que toma su nombre del juego de Rugby, está basada en un proceso iterativo e incremental. A su vez se define una serie de roles que determinan las responsabilidades de los involucrados en el proceso de desarrollo[1].

² Conjunto de módulos informáticos dedicados a la administración y planeamiento de los recursos que forman parte de una empresa determinada [2].

³ Sistema de Administración Tributaria de la república mexicana y es el órgano encargado de dictar las normas y estándares que regulan la forma en la que se emiten facturas [4].

Capítulo 1: Conceptos Básicos.

En este capítulo se dará un panorama general de lo que es la **Factura Digital** y el por qué es una herramienta que, además de ser útil a la sociedad ya que simplifica y transforma el proceso de facturación en un proceso más seguro, es un excelente proyecto en el cual se pueden aplicar metodologías ágiles como lo es SCRUM.

1.1 Facturación Digital

Es un sistema informático que se encarga de generar facturas electrónicas siguiendo los estándares impuestos por el SAT. Una factura electrónica es un documento digital, que por sus características en cuanto a seguridad y a manejabilidad puede ser usado para el fin que el usuario lo considere necesario, ya sea para fines contables, administrativos declarativos etc., es decir, es la versión digital de una factura física que hoy por hoy es válida ante cualquier institución gubernamental y que está avalada por el SAT.

Actualmente en nuestro país a este tipo de facturas no se les ha dado la importancia que ameritan y no se ha visto el potencial que estas poseen ya que cuentan con múltiples ventajas ante sus similares tradicionales que se mencionarán más adelante. Sin embargo a partir del año 2011 la factura digital será obligatoria para aquellas empresas que facturen más de \$4,000,000 anualmente y desde esta perspectiva pareciera que un sistema de facturación digital, no tendría demasiada importancia y podría pasar desapercibido como una solución integral a las empresas que emiten facturas, sin embargo después de mencionar las ventajas más importantes de la facturación digital será evidente que a pesar de que la factura digital no sea obligatoria para todo tipo de empresas, aquellas que no adopten este tipo de facturación dentro de sus estándares estarán compitiendo en desigualdad de condiciones.

¿Que ventajas trae consigo hacer uso de un sistema de facturación digital?

Como se mencionó anteriormente los beneficios que trae consigo la facturación digital dentro de una empresa es directamente proporcional al número de facturas emitidas, y es una herramienta útil para cualquier empresa que emita facturas.

- 1 El manejo de un documento digital es mucho mejor y más sencillo que el manejo de un documento físico.
³⁵/₁₇ Una factura digital es más sencilla de transportar de un lado a otro y no es necesario llevar una gran cantidad de hojas de papel, basta con almacenar los archivos en algún dispositivo digital.
- 2 Hoy en día el envío de un documento digital es más rápido, ágil, barato y seguro que tratar de enviar un documento físico.
³⁵/₁₇ Enviar una factura digital vía correo electrónico es instantáneo, a diferencia del envío de facturas de la manera tradicional.
³⁵/₁₇ El envío de correos electrónicos no genera costos excesivos que puedan encarecer el manejo de documentos digitales.
³⁵/₁₇ Una factura digital es más segura que una factura tradicional, debido a varios candados que la hacen prácticamente infalsificable.
- 3 El almacenaje de los documentos digitales es más eficiente y barato, lo que se traduce en la reducción de gastos para la empresa.
³⁵/₁₇ Al momento de generar una factura de manera tradicional, además de generar gastos por la cantidad de hojas que son necesarias y demás elementos que están involucrados en la generación de una factura de este tipo, el almacenaje de las mismas es de igual manera un elemento que encarece el costo de emitir una factura tradicional.
³⁵/₁₇ A su vez al momento que ya no sea útil almacenar dichas facturas tradicionales, es necesario pagar un costo por la destrucción de las mismas. Con una factura digital no se tiene este tipo de inconvenientes.
- 4 La automatización de los procesos que dan como resultado una factura digital traen como resultado, que el desempeño de la empresa en general sea óptimo.
³⁵/₁₇ Al automatizar el proceso de generar facturas digitales, así como algunos otros procesos que pueden verse envueltos al generar una factura digital, se reducen tiempos dentro de la

empresa.

³⁵₁₇ El reducir tiempos de producción se traduce en menores costos tanto para la empresa como para sus clientes.

La unión de estas ventajas tiene como resultado que el uso de un sistema para generar facturas digitales se vuelva indispensable en la mayoría de las empresas que expiden facturas.

¿Cómo se justifica el que el tema fundamental de esta tesis sea la facturación digital?

Durante mi estancia como estudiante de la facultad de ciencias me pude percatar de lo útil y prácticamente indispensable que resulta ser un sistema integral que resuelva la gran mayoría de procesos que pueden ser automatizados como en este caso la expedición de facturas electrónicas. Una vez concluida la etapa de estudiante y al momento de incorporarme al mercado laboral fue evidente que un sistema que incorpore en sus estándares la facturación digital sería de gran ayuda para cualquier empresa que desea generar facturas digitales. En suma, ambas experiencias fueron los justificantes para el tema de esta tesis.

Después de haber cursado el seminario de Facturación Digital impartido en la Facultad de Ciencias de la U.N.A.M. por el profesor Manuel López Mateos y posteriormente haber colaborado con él como su ayudante impartiendo la misma materia y con la experiencia laboral que he reunido hasta el momento de escribir esta tesis, habrá múltiples empresas que estarán dispuestas a poner en producción un sistema de facturación digital.

Por otro lado en mi experiencia laboral la tendencia de la ingeniería de software hoy por hoy es el desarrollo de software basado en iteraciones cíclicas en el cual se considera al software como un ente cambiante, susceptible a modificaciones continuas por lo que SCRUM es una metodología ideal para el desarrollo de un sistema de esta naturaleza. Además el tema de las metodologías ágiles que se trata en este trabajo es un tema al que, en mi experiencia, aun le hace falta estandarizarlo y darle la importancia que este merece como eje de cualquier desarrollo de software, por lo que lo hace un tema muy atractivo para la investigación y divulgación.

Además a esto se suma el factor económico y la estandarización, con esto me refiero a que, hasta el momento los sistemas de facturación digital que se utilizan en nuestro país, son sistemas desarrollados en el extranjero y además los costos de estos sistemas son elevados, el cliente tiene que adaptarse al sistema, siendo que una de las premisas del desarrollo de software es que el sistema se adapte a las necesidades del usuario.

Objetivo de la tesis

Esta tesis tiene como objetivo implementar paso a paso mediante SCRUM, un sistema de facturación digital mismo que pueda ofrecer una solución a las empresas que necesiten adoptar la facturación digital dentro de sus estándares, así como servir de guía para todo aquel que este interesado en incluir SCRUM dentro de sus prácticas laborales.

1.2 Elementos indispensables de la Factura Electrónica

Antes de entrar de fondo al planteamiento del problema se definirán algunos conceptos básicos.

La factura electrónica en México es la representación digital de un tipo de comprobante fiscal digital (CFD), que está apegada a los estándares definidos por el Servicio de Administración Tributaria (SAT) en el anexo 20 de la Resolución de Miscelánea Fiscal, y la cual puede ser generada, transmitida y resguardada utilizando medios electrónicos. Cada factura electrónica emitida cuenta con un certificado digital y sello digital que corrobora su origen y le da validez ante el SAT; una cadena original que funciona como un resumen del contenido de la factura; y un folio que indica el número de la transacción.

1.2.1 Firma Electrónica Avanzada (FEA)

Con la aprobación de la Miscelánea Fiscal del 2004, el SAT pretende facilitar a los contribuyentes la realización de varios trámites incluyendo la emisión de Facturas Electrónicas mediante el uso de la Firma

Electrónica Avanzada (FEA). La Firma Electrónica Avanzada permite a los contribuyentes cumplir con sus obligaciones a través de internet de manera fácil, gratuita y segura. La Firma Electrónica Avanzada (sello digital en caso de las facturas electrónicas) son datos adjuntos a un mensaje electrónico vinculados a la información contenida y a la identidad de la persona que lo origina.

¿Cuáles son los pasos para tramitar la Firma Electrónica Avanzada?

Tramitar la Firma Electrónica Avanzada es un proceso sencillo. A grandes rasgos el proceso consiste en 4 pasos:

- 1) Concertar una cita en la oficina del SAT de su elección.
- 2) Generar un requerimiento de certificación.
- 3) Reunir la documentación necesaria.
- 4) Acudir a su cita con el SAT a generar su firma.

Toda aquella empresa que tenga pensado implementar la facturación digital, debe tomar en cuenta que un prerequisite indispensable es contar con un representante legal, que a su vez cuente con su FEA para poder generar un CFD o Factura Electrónica.

1.2.2 Comprobante Fiscal Digital o CFD

Un Comprobante Fiscal Digital es un documento digital que se recibe al adquirir un bien, servicio o usar temporalmente bienes inmuebles. Dentro de los comprobantes fiscales digitales se encuentran: factura electrónica, recibo de honorarios, recibo de arrendamiento, nota de débito, nota de crédito, recibo de donativo, y carta porte.

1.2.3 Cadena Original

La cadena original, es uno de los principales elementos de una factura digital y esta consta de la información que contiene la factura, como: datos del emisor, cantidad, año, folio, etc., agrupada en una sola cadena de texto plano y haciendo uso de pipes “|” para separar cada campo, ejemplo:

```
||2.0|ABCD|2|03-05-2010T14:11:36|49|2008|INGRESO|UNA SOLA EXHIBICIÓN|  
2000.00|00.00|2320.00|PAMC660606ER9|CONTRIBUYENTE PRUEBA SEIS PATERNOSEIS  
MATERNOSEIS|PRUEBA SEIS|6|6|PUEBLA CENTRO|PUEBLA|PUEBLA|PUEBLA||MÉXICO|72000|  
CAUR390312S87|ROSA MARÍA CÁLDERON URIEGAS|TOPOCHICO|52|JARDINES DEL VALLE|  
NUEVO LEÓN|MEXICO|95465|1.00|SERVICIO|01|ASESORIA FISCAL Y ADMINISTRATIVA|  
2000.00|IVA|16.00|320.00||
```

1.2.4 Extensible Markup Language (XML)

XML o *eXtensible Markup Language* es un metalenguaje⁴ que sirve para estructurar de mejor manera la información. XML es utilizado por ejemplo en sitios web y en conjunto con HTML el resultado es conocido como XHTML⁵, sin embargo XML no surgió únicamente para su aplicación en internet, si no que busca convertir se en un estándar para el intercambio de información en medios informáticos. Y dado que la interacción entre el sistema de facturación y el SAT será constante, lo ideal es contar con un estándar, ejemplo:

⁴ Metalenguaje es un tipo de lenguaje que sirve para hacer referencia a otro lenguaje, denominado lenguaje objeto [6].

⁵ Siglas del inglés *eXtensible HyperText Markup Language*. XHTML es básicamente HTML expresado como XML válido. Es más estricto a nivel técnico, pero esto permite que posteriormente sea más fácil al hacer cambios o buscar errores entre otros [8].


```

<?xml version="1.0" encoding="ISO-8859-1"?>
<XML-CCI xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<Cabecera>
  <VersionCodificacion>String</VersionCodificacion>
  <Modalidad>String</Modalidad>
  <TipoEmisorFactura>EM</TipoEmisorFactura>
  <EmisorTercero>
    <IdentificacionFiscal>
      <TipoPersona>F</TipoPersona>
      <TipoResidencia>E</TipoResidencia>
      <NumeroDocumento>String</NumeroDocumento>
    </IdentificacionFiscal>
  <PersonaJuridica>
    <RazonSocial>String</RazonSocial>
    <NombreComercial>String</NombreComercial>
    <DatosRegistrales>
      <Libro/>
      <RMercantil/>
    </DatosRegistrales>
  </PersonaJuridica>
</Cabecera>
<...../>

```

Este ejemplo corresponde a la descripción de una factura y como se puede ver en el ejemplo por cada etiqueta que abre hay una que cierra, de tal manera que cada conjunto de etiquetas representa un campo que puede ser mapeado en una factura. Como ya se explico esta una forma de organizar y acceder a la información de manera estructurada

1.2.5 Extensible Stylesheet Language Transformations (XSLT)

XSLT (siglas de Extensible Stylesheet Language Transformations, expresión inglesa traducible como "lenguaje extensible de hojas de estilo") es un conjunto de reglas normalmente conocidas como plantilla la cual permite transformar un documento XML en otro de la misma clase o incluso en documentos con otro tipo de formato como textos planos, páginas web o incluso documentos tipo pdf.

1.2.6 PDF

Portable Document Format, es el formato de archivo ampliamente aceptado en medios informáticos y que sirve para mostrar información de manera visual, gracias a su portabilidad es sumamente compatible. Es el producto final del sistema y en éste se plasman todos y cada uno de los elementos que conforman la factura electrónica. En este documento se puede encontrar contenida toda la información que hace de la factura electrónica un medio altamente seguro.

1.3 Metodologías

Desde el momento en el que el desarrollo de software se volvió una parte importante de la sociedad surgió la necesidad de darle una estructura a este desarrollo. Como es bien conocido, cualquier actividad que involucre la interacción de varios individuos para lograr una meta en común, es de vital importancia definir las reglas del juego. Por ejemplo al momento de construir un edificio es importante que el arquitecto identifique los elementos que integran dicha estructura y se comunique con el ingeniero para determinar la organización y la interacción de dichos elementos.

A través de la historia de la computación se ha visto que el proceso de desarrollo de software no es muy diferente de otros procesos de desarrollo como por ejemplo, el de la construcción de un edificio y debido a este tipo de experiencia también se ha llegado a la conclusión de que es muy importante definir desde un inicio la metodología que será usada en el desarrollo de software.

A primera vista no es tan evidente el por qué es necesario definir dicha metodología en el desarrollo de software, como lo sería en el proceso de construcción ya que en el proceso de construcción es tangible y hasta cierto punto obvio, ya que sin el orden que proporciona una metodología el edificio en cuestión se derrumba o simplemente no es funcional. De igual manera en el proceso de desarrollo de software, si no hay un orden en este proceso, está destinado a fracasar o a volverse incosteable. Por experiencia propia

me he dado cuenta que la tendencia principal de cualquier programador, es precisamente esa, programar y no tomarse tiempo para definir las reglas con lo que el desarrollo se vuelve más fácil, más fluido y al final más ordenado.

Para fines de esta tesis y en general en lo que concierne el desarrollo de software, una metodología hace referencia al conjunto de métodos que regirá el desarrollo del software. Dentro de las metodologías para el desarrollo de software hay dos corrientes principales:

Metodologías Formales: Este tipo de metodologías, son muy rígidas en cuanto a los métodos que se utilizan y demandan una gran cantidad de documentación y trámites laborales.

Metodologías Ágiles: Surgen como una alternativa a las metodologías formales, están basadas en el *manifiesto ágil*⁶ y sus principales características son:

1. Valorar más a los individuos y su interacción que a los procesos y las herramientas.
2. Valorar más el software que funciona que la documentación exhaustiva.
3. Valorar más la colaboración con el cliente que la negociación contractual.
4. Valorar más la respuesta al cambio que el seguimiento de un plan.

Por sus características las metodologías ágiles se adaptan más a las necesidades de hoy en día y satisfacen de mejor manera las cambiantes necesidades del usuario.

1.4 SCRUM

Scrum debe su nombre al juego de pelota llamado rugby, en el cual el objetivo principal es llevar la pelota de un extremo al otro y esto puede ser logrado mediante el scrum, en el cual los jugadores de un equipo se alinean de manera horizontal y de manera unitaria forman una falange que tiene como propósito pasar el balón de forma diagonal a su siguiente compañero libre. De tal forma que el equipo completo forma una sola unidad y trabaja de manera conjunta para lograr su objetivo, que en este caso es llevar el balón hasta el terreno del contrincante.

La analogía con este juego (rugby) y el desarrollo de software es fácil de ver, ya que el objetivo de SCRUM dentro del desarrollo de software es lograr que el equipo de trabajo funcione de manera integral y como una sola unidad fuerte y responsable y dando pequeños pasos pero firmes, de igual manera que en el rugby, y obtener un producto de software de calidad.

¿Cómo funciona Scrum?

Como la mayoría de las metodologías, si no es que todas, SCRUM define los roles que desempeña cada miembro. Mismos que pueden ser subdivididos en dos grandes grupos, los cerdos y las gallinas, esta subdivisión responde a la necesidad de definir que elementos del SCRUM son de vital importancia y que sin ellos no puede funcionar esta metodología y cuales a pesar de ser importantes no son indispensables. Como regla nemotécnica se tiene una pequeña fábula en la cual hay un cerdo y una gallina platicando y la gallina le comenta al cerdo “Oye sería muy buena idea que juntos abriéramos un restaurante” a lo que el cerdo responde, “Me parece una excelente idea, ¿cómo le pondremos de nombre?” y la gallina comenta “¿Qué te parece Huevos con Jamón” a lo que el cerdo se niega y responde “Con ese nombre tú estás involucrada, pero yo estoy **comprometido**”. Esta pequeña fábula tiene como objetivo mostrar que

⁶ En marzo de 2001 diecisiete críticos de los modelos de mejora del desarrollo de software basados en procesos, convocados por Kent Beck, se reunieron en Salt Lake City para tratar sobre técnicas y procesos para desarrollar software. En la reunión se acuñó el término “Métodos Ágiles” para definir a los métodos que estaban surgiendo como alternativa a las metodologías formales (CMMI, SPICE) a las que consideraban excesivamente “pesadas” y rígidas por su carácter normativo y fuerte dependencia de planificaciones detalladas previas al desarrollo. Los integrantes de la reunión resumieron los principios sobre los que se basan los métodos alternativos en cuatro postulados, lo que ha quedado denominado como **Manifiesto Ágil** [10].

hay roles dentro del equipo que son de vital importancia, que a la vez están comprometidos con el producto final y de manera inmediata con los objetivos de la agenda y por otro lado hay roles que a pesar de estar involucrados no son indispensables como la gallina de la fábula.

Puercos

1. Propietario del Producto conoce y marca las prioridades del proyecto o producto. Es el interesado en que el producto sea completado de manera adecuada y de acuerdo a las necesidades establecidas por los usuarios.
2. El Scrum Manager es la persona que asegura el seguimiento de la metodología guiando las reuniones y ayudando al equipo ante cualquier problema que pueda aparecer. Su responsabilidad es entre otras, la de hacer de paraguas ante las presiones externas.
3. Team son las personas responsables de implementar la funcionalidad o funcionalidades elegidas por el Product Owner. Dichas funcionalidades deben ser completadas en la duración del sprint.

Gallinas

1. Los Usuarios o Interesados, son los beneficiarios finales del producto, y son quienes viendo los progresos, pueden aportar ideas, sugerencias o necesidades.

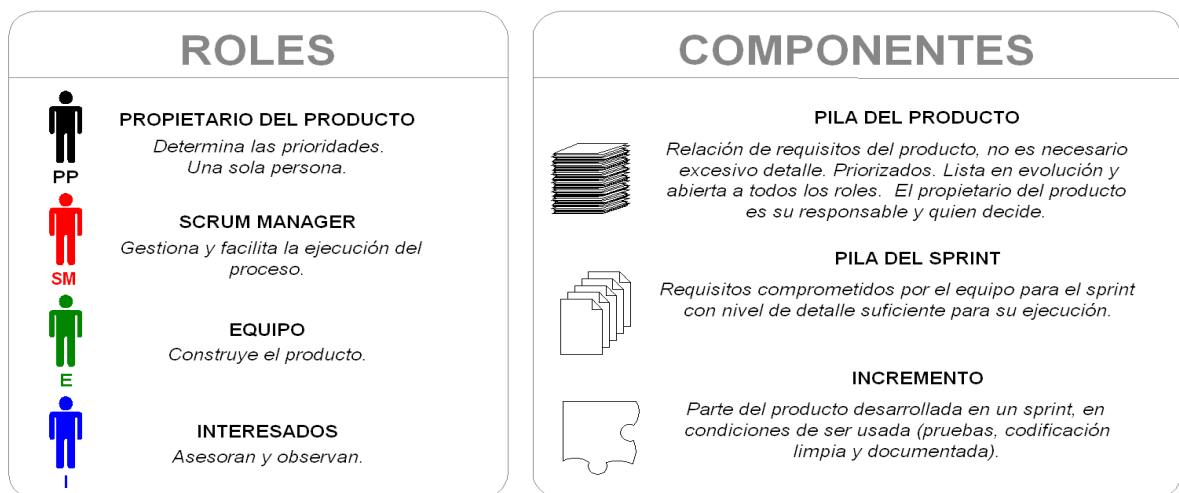


Figura 1.1. Diagrama de roles y componentes de SCRUM

Una vez definidos los roles podemos proceder a establecer las interacciones y las reglas mediante las cuales se rige esta metodología.

Componentes

1. Pila del Producto o Product Backlog es la lista de tareas, funcionalidades o requerimientos a realizar. Y es el propietario del producto es el encargado de establecer las prioridades, ya que al fin y al cabo, es la persona que mantiene y actualiza, la lista de tareas. Cuenta con varias columnas que nos indican el estatus de cada tarea, las columnas para el *Product BackLog* de SiFaDig serán:
 - a. Estatus: indica si la tarea ya fue planificada o aun si es que aun no ha sido planificada
 - b. Tamaño: indica la cantidad de trabajo que se requiere invertir en esta tarea, el trabajo puede ser el tiempo que toma o la complejidad de realizar la tarea.
 - c. Sprint: indica en que ciclo fue realizada esta tarea.
 - d. Prioridad: indica que tan importante es llevar a cabo dicha tarea.

2. Pila del Sprint o Sprint Backlog corresponde a una pequeña fracción del Product Backlog. Es decir, del Product Backlog, son las tareas que tienen que ser realizadas en un sprint y que van a formar parte del Sprint Backlog.
3. Incremento resultado del sprint, este incremento debe haber pasado las pruebas correspondientes además de estar en condiciones de integrarse con el resto de los incrementos previamente
4. Sprint o ciclo las tareas del Sprint Backlog se deben hacer (recomendado) en unas 2 semanas ó 4 semanas. Hay Sprint de 2 semanas y hay Sprint de 4 semanas. Eso debe de ser marcado antes de iniciar. Una norma fundamental es que mientras un Sprint se inicia, éste no puede ser alterado o modificado.
5. Inicio de Ciclo al inicio de cada ciclo se mostrara una tabla que está compuesta de los objetivos generales del ciclo y de los puntos más importantes que se resolverán en el mismo. A su vez se muestra una pequeña tabla con las tareas a realizar, cada tarea estará clasificada por tamaño y prioridad.
 - a. **Tamaño:** es la cantidad de trabajo que se estima tomara realizar dicha tarea, este tamaño se toma del ProductBackLog.
 - b. **Prioridad:** determina que tan importante es realizar dicha tarea. La escala de prioridades va del 1 al 4, siendo la prioridad 1 la más alta y la 4 la más baja. Esta prioridad también se toma del ProductBackLog.
6. Gráfica de Quemado los requerimientos o tareas estarán clasificados de acuerdo a su prioridad y al trabajo que estas demandaran. Al inicio de cada sprint se mostrara una grafica en la cual se muestra como estarán distribuidas las tareas a lo largo del ciclo y la cantidad de trabajo que estas demandan. Esta grafica estará compuesta de los siguientes elementos:
 - a. Tareas Completadas: Indica la cantidad trabajo que ha sido realizado y en qué día se completo.
 - b. Desempeño Ideal: Indica cual sería la forma ideal de llevar a cabo las tareas, esta indicador siempre corresponde a la diagonal de la grafica ya que la diagonal supone una distribución equilibrada de las tareas entre los días que comprende el sprint.
 - c. Esfuerzo Remanente: Muestra la cantidad de trabajo pendiente conforme van pasando los días y tomando en cuenta que algunas de las tareas ya fueron completadas
7. Gráfica de Avance esta gráfica estará dividida de acuerdo a las tareas principales y en ella se mostrara el avance general que se lleva, tanto esta grafica como la grafica de quemado se basan en el Product BackLog.
8. El Daily Scrum Meeting es una reunión que se realiza todos los días que dure el Sprint con el equipo de desarrollo o de trabajo. Se trata de una reunión operativa, informal y ágil, de un máximo de 30 minutos, en la que se le hace 3 preguntas a cada integrante del equipo.
 - 1.¿Como vas con tus tareas.
 - 2.¿Te hace falta algo para continuar con tus tareas?.
 - 3.¿Tienes algún punto que pueda ser utilizado para un análisis en retrospectiva?.Está destinada a informar y a resolver las dificultades que hayan surgido en el transcurso del Sprint . Es una reunión informal y se recomienda llevar a cabo de pie, para no alargar demasiado la reunión.

Ventajas de Scrum

Como ya mencione con anterioridad, SCRUM no es la única metodología sin embargo es la que mejor se adapta a las necesidades actuales en lo que a desarrollo de software concierne, algunas de sus principales ventajas son las siguientes:

1. Es una metodología fácil de aprender y de seguir.
2. Es muy flexible, ya que siempre se está atento a las necesidades del usuario.
3. A diferencia de otras metodologías, el objetivo principal es la interacción con el usuario, el compromiso con los productos y no tanto la documentación.
4. No es necesaria documentación excesiva y exhaustiva como con otras metodologías.
5. Fácil de comenzar a utilizar.

Capítulo 2: Open Enterprise Resource Planning (OpenERP)

2.1 Definición

Los sistemas informáticos han probado ser más que una herramienta para automatizar procesos con los que se pueden reducir costos, aumentar ganancias y mantener una mejor organización de la información y de las múltiples áreas que conforman la empresa en cuestión.

Para lograr este objetivo es necesario fijar algunas ideas y manejar estándares para lo cual un sistema ERP (Enterprise Resource Planning) es la solución ideal.

Se tienen muchas barreras al tratar de implementar herramientas de este tipo (ERP) como son los costos en cuanto a riesgos e integración a tal grado que son pocas las medianas-empresas que han adoptado este tipo sistemas dentro de sus procesos internos. Además de que las grandes empresas desarrolladoras de ERPs como lo son SAP, Microsoft y Oracle no han tenido la capacidad para conciliar la potencia de un ERP con la flexibilidad que requiere el usuario final. Sin embargo en este punto es donde el proceso de desarrollo de software libre y los nuevos modelos adoptados por sus desarrolladores plantean una nueva forma de resolver el problema de costos e integración que este tipo de software empresarial genera.

Para que un ERP sea completamente accesible para una mediana-empresa el principal objetivo es la reducción de costos. El software libre ofrece una excelente alternativa a este respecto, ya que elimina factores como:

- ³⁵₁₇ Vendedores.
- ³⁵₁₇ Intermediarios.
- ³⁵₁₇ Impuestos relacionados.

Y esto se logra mediante la reutilización de código y la publicación de actualizaciones de manera gratuita

2.1.1 Ventajas de OpenERP

La principal ventaja al hacer uso de una herramienta del tipo OpenERP es el hecho de que es una herramienta libre y por lo tanto no se tiene que pagar por la licencia al hacer uso de ella, además de ser una herramienta en constante cambio y siempre muy dinámica, adaptándose a las necesidades sumamente cambiantes del software de hoy en día.

Otra de las principales ventajas que se tiene al hacer uso de OpenERP es el hecho de que las labores básicas y bien conocidas de la mayoría de los sistemas, tales como:

- ³⁵₁₇ Administración de usuarios.
- ³⁵₁₇ Administración de productos.
- ³⁵₁₇ Persistencia de datos.
- ³⁵₁₇ Reportes.

Ya están más que desarrolladas y probadas, por lo que permite en nuestro caso dedicarnos únicamente al desarrollo del módulo en cuestión o sea, el de la Facturación Electrónica.

Por otro lado OpenERP cuenta con una infinidad de módulos que ya han sido desarrollados y se encuentran disponibles para su uso, mismos que cuentan con soporte por parte de la comunidad encargada del desarrollo de OpenERP.

Y por último pero no menos importante, OpenERP está diseñado de manera tal que cualquier programador con un poco de experiencia y con el conocimiento adecuado, puede modificar y adaptar esta herramienta a las necesidades del usuario, lo que la vuelve una herramienta muy poderosa y accesible para cualquier empresa.

Otra de las grandes ventajas de un Open ERP es el hecho de que miles de personas (trabajadores, desarrolladores, empresarios, etc.) utilizan herramientas como estas y a su vez participan, de acuerdo a su competencia laboral en la mejora y desarrollo de las mismas lo que trae como resultado una herramienta viva, siempre cambiante y sin límite de posibles aplicaciones.

Pero la prueba final en el desarrollo e integración de un Open ERP es lograr que sea simple de utilizar y lo suficientemente flexible para que se adapte a las necesidades del usuario. En esta tesis se tratará de mostrar un camino para el desarrollo del módulo destinado a la generación de facturas digitales, que puede ser incrustado en un OpenERP.

2.1.1.1 Modularidad de OpenERP

Gracias a su alto nivel de modularidad⁷, el desarrollo colaborativo dentro de Open ERP ha generado un gran número de aplicaciones que pueden ser incluidas al modelo que será adoptado por la empresa que lo requiera, proporcionando un gran abanico de posibilidades para el usuario final y para la empresa.

De tal manera que, el usuario y el desarrollador pueden trabajar en conjunto agrupando los miles de módulos y funcionalidades disponibles de acuerdo a las necesidades de la empresa. De forma tal que al final el sistema generado sea el modelo mejor adaptado a los requerimientos del usuario.

Open ERP (una vez llamado Tiny ERP) es un software administrativo que es descargado más que cualquier otro en el mundo, con más de 600 descargas por día. Está disponible en 18 lenguajes y cuenta con una red global de socios y contribuyentes. Como dato adicional Open ERP es utilizado por una gran cantidad de empresas que van desde empresas dedicadas al ramo de la agricultura, textiles, pasando por editoriales y hasta agencias de viajes.

Esto lo que nos indica es que por el momento Open ERP es la opción ideal para desarrollar e integrar el módulo de facturación digital que se trata en esta tesis. Aunado a esto, Open ERP se puede instalar fácilmente bajo ambientes Linux y Windows, lo que lo hace aun más llamativo para cualquier usuario.

2.1.2 Desventajas de OpenERP

Por experiencia propia he podido constatar que la principal desventaja que enfrenta OpenERP, es la resistencia por parte de los usuarios al cambio de sistemas y de procesos. Esta resistencia es una tendencia natural por parte del ser humano, ya que muchos de los usuarios prefieren seguir utilizando sus viejos sistemas con tal de no aprender las nuevas utilidades y aplicaciones que puede llegar a ofrecer OpenERP.

2.2 Diseño y Arquitectura OpenERP

La primera decisión importante que se debe tomar al instalar Open ERP, es decidir qué ambiente se desea utilizar.

³⁵₁₇ Utilizar una arquitectura, cliente-servidor.

³⁵₁₇ Una aplicación tipo cliente GTK⁸ instalada en cada computadora que se vaya a utilizar.

Si lo que se busca es una aplicación que será utilizada por muchos usuarios y que, para facilidad del cliente es preferible evitarle la instalación, entonces la decisión no es muy complicada y lo ideal sería tener Open ERP instalado en un servidor web y que los usuarios tengan acceso a este sistema vía web.

Por otro lado si el número de usuarios es mínimo y no hay inconveniente en cuanto a la instalación, la mejor decisión sería instalar una aplicación tipo GTK

⁷ Es la capacidad que tiene un sistema de ser estudiado, visto o entendido como la unión de varias partes que interactúan entre sí y que trabajan para alcanzar un objetivo común, realizando cada una de ellas una tarea necesaria para la consecución de dicho objetivo [11].

⁸ GTK "GIMP Tool Kit" es una biblioteca del equipo GTK+, la cual contiene los objetos y funciones para crear la interfaz gráfica de usuario. Maneja elementos como ventanas, botones, menús, etiquetas, deslizadores, pestañas, etc [12].

Nota: Hay una pequeña diferencia entre los dos clientes (web y GTK). La principal diferencia entre ambos es el caching⁹, principalmente se utiliza para acelerar el acceso a los elementos que componen el sistema. Sin embargo, esto provoca que al realizar cambios en la aplicación, éstos no se reflejen de manera inmediata causando algunas fallas al momento de realizar las pruebas. Se recomienda utilizar el cliente GTK durante la etapa de desarrollo e incluso para la implementación, cuando sea posible.

2.3 Diagrama de un sistema tipo OpenERP

Un sistema del tipo Open ERP, se compone principalmente de 3 elementos:

- ³⁵₁₇ Un servidor para la base de datos con manejador tipo PostgreSQL, en la cual se almacena la mayor parte de la información necesaria para que el sistema funcione de manera correcta, así como los datos de configuración del mismo.
- ³⁵₁₇ Un servidor para la aplicación en sí misma, es decir, un servidor que se encargue de almacenar el código y la lógica para que aplicación funcione correctamente.
- ³⁵₁₇ Y de manera opcional un servidor web que se encargue de realizar la conexión a la aplicación desde cualquier navegador estándar. En caso de que la instalación se lleve a cabo mediante un cliente tipo GTK, este servidor no es necesario.

A continuación se muestra una pequeña imagen de cómo se vería el diagrama.

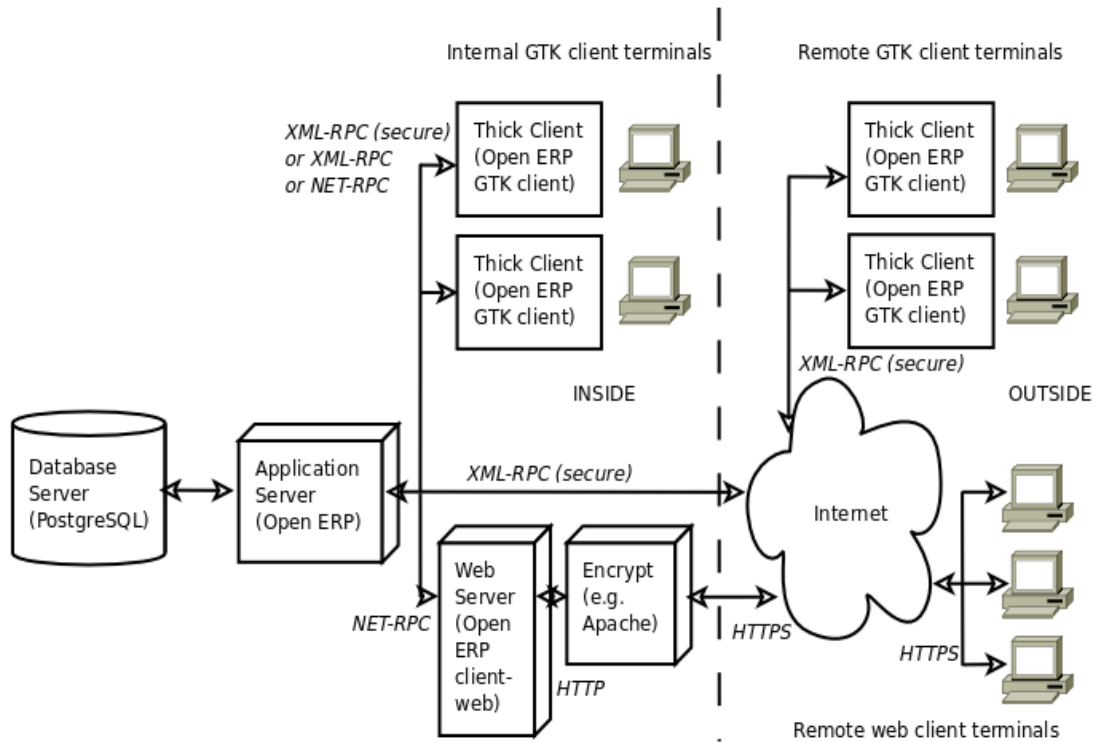


Figura 2.1. Arquitectura de Open ERP (Documentación OpenERP)

Vale la pena aclarar que se tomó la decisión de utilizar PostgreSQL para el manejo de la base de datos, debido a que es un sistema de código abierto y de alto desempeño que puede ser comparado con muchos otros manejadores de bases de datos como MySQL o FirebirdSQL. Además de que puede ser utilizado en múltiples sistemas tipo Unix/Linux y en la mayoría de las versiones de Windows, Mac, Solaris entre algunos otros.

⁹ Almacenar temporalmente los datos frecuentemente accedidos más cerca del solicitante de los mismos.

2.4 Componentes

2.4.1 Python

El lenguaje utilizado para desarrollar OpenERP es Python, es un lenguaje multiparadigma que brinda soporte para la orientación a objetos. Es un lenguaje interpretado, con tipado dinámico y explícitamente tipado, es un lenguaje que hace énfasis en una sintaxis sencilla y un código limpio y muy sencillo de entender. Gracias a que fue desarrollado en a finales de los años 80' le permitió a su creador Guido Van Rossum incorporar las mejores características de sus predecesores y tratar de minimizar al máximo las desventajas o puntos débiles de los mismos.

En la documentación de OpenERP se menciona que se decidió utilizar Python para desarrollar OpenERP debido a la sencillez del lenguaje y que se buscaba atraer al máximo número de desarrolladores posibles que ayudaran a generar una gran comunidad entorno a OpenERP.

Gracias a la simplicidad del lenguaje se logra que la curva de aprendizaje para cualquier desarrollador en el tema OpenERP se mantenga muy baja y que el desarrollo sea muy claro, intuitivo y sencillo. Podría dedicar capítulos enteros para mencionar las ventajas, desventajas o para justificar el por qué se uso Python para el desarrollo de OpenERP, sin embargo es más que suficiente para los objetivos de esta tesis dejar en claro que la principal ventaja de Python, así como el por qué se utilizó este lenguaje para desarrollar OpenERP es por eso que es un lenguaje completo, que ofrece múltiples ventajas, pero sobre todo resulta sencillo de aprender.

2.4.2 Arquitectura

Dentro del desarrollo de software una de las arquitecturas más frecuentemente implementada gracias a sus muchas ventajas y flexibilidad, así como sencillez, es la arquitectura tipo **Cliente-Servidor**. La principal característica de esta arquitectura es el hecho de que la aplicación así como el SDB se encuentran instaladas en una o más computadoras denominadas *servidores* y son las encargadas de ejecutar los cálculos e instrucciones solicitadas por otra computadora denominada *cliente* que se encarga de mostrar los resultados de las instrucciones realizadas. Este tipo de arquitectura puede ser implementada sin importar si el sistema se ejecutará en una o más computadoras o si se ejecutará en una red local o a través de internet y es por eso que este tipo de arquitectura es muy flexible y adaptable, además de que la idea principal es muy sencilla, un cliente que genera instrucciones y uno o más servidores que las ejecutan.

Un pequeño esquema de cómo funciona la arquitectura cliente- servidor se muestra a continuación.

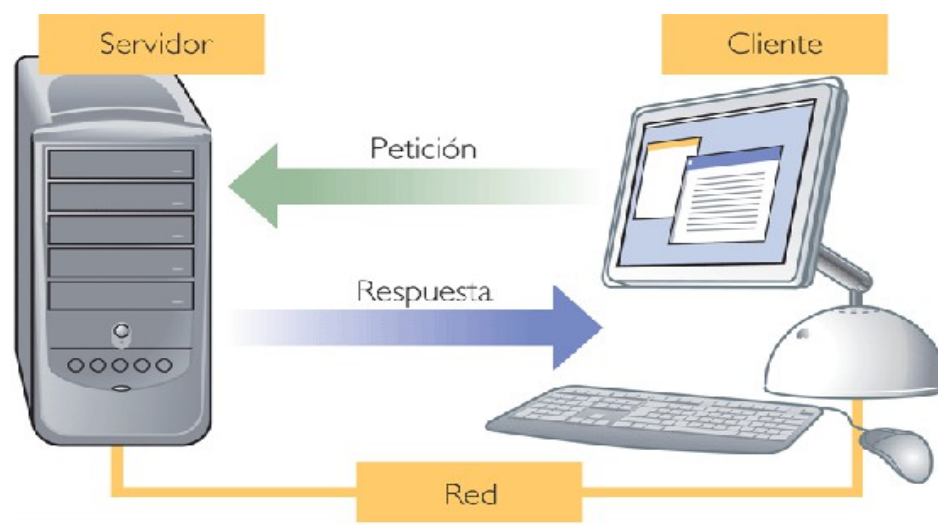


Figura 2.2 Arquitectura Cliente-Servidor

2.4.3 Sistema Manejador de Bases de Datos (SMBD)

Con respecto al manejador de la base de datos, OpenERP utiliza PostgreSQL o como es comúnmente conocido PostgreSQL¹⁰. En la documentación de OpenERP se menciona el por qué se utiliza PostgreSQL como SMBD, principalmente esto se debe a la naturaleza libre de este manejador, así como sus múltiples ventajas. En esta sección me limitaré a mencionar algunas de ellas, pero lo más importante es dejar en claro que la elección del SMBD para cualquier sistema es una decisión muy importante ya que es sobre la que recae gran parte del desempeño del sistema. Es más que bien sabido que PSQL es una opción altamente recomendada y probada, así que sin más preámbulos, éstas son sus principales ventajas y el por qué se utiliza de base para OpenERP.

2.4.3.1 Ventajas PostgreSQL

³⁵/₁₇ Alta concurrencia

Mediante un sistema denominado Multiversion Concurrency Control (MVCC Acceso concurrente multiversión, por sus siglas en inglés) PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo *commit*¹¹. Esta estrategia es superior al uso de bloqueos por tabla o por filas que es común en otras bases, eliminando la necesidad del uso de bloqueos explícitos.

³⁵/₁₇ Amplia variedad de tipos nativos

PostgreSQL provee soporte nativo para:

- ³⁵/₁₇ Números de precisión arbitraria.
- ³⁵/₁₇ Texto de largo ilimitado.
- ³⁵/₁₇ Figuras geométricas (con una variedad de funciones asociadas).
- ³⁵/₁₇ Direcciones IP (IPv4¹² e IPv6¹³).
 - Bloques de direcciones estilo CIDR.
 - Direcciones MAC.
 - Arreglos.

³⁵/₁₇ Claves Ajenas

También denominadas Llaves ajenas o Claves Foráneas (*foreign keys*), son parte del modelo relacional y son muy útiles para generar relaciones entre tablas dentro de una base de datos.

³⁵/₁₇ Disparadores(*triggers*)

Un disparador o *trigger* se define como una acción específica que se realiza de acuerdo a un evento, cuando éste ocurra dentro de la base de datos. En PostgreSQL esto significa la ejecución de un procedimiento almacenado basado en una determinada acción sobre una tabla específica.

2.5 Instalación de OpenERP

La decisión más importante antes de instalar el servidor de OpenERP es elegir el sistema operativo adecuado para llevar a cabo dicha instalación. Por experiencia he aprendido que un sistema operativo de licencia libre es la elección ideal para instalar OpenERP y esto es debido a que OpenERP como su

¹⁰ Es un sistema de gestión de base de datos relacional orientada a objetos y libre, publicado bajo la licencia BSD. Como muchos otros proyectos de código abierto, el desarrollo de PostgreSQL no es manejado por una empresa y/o persona, sino que es dirigido por una comunidad de desarrolladores que trabajan de forma desinteresada, altruista, libre y/o apoyados por organizaciones comerciales [13].

¹¹ Commit: sentencia que hace referencia al momento de enviar una solicitud a la base de datos, ya sea para consulta o modificación de la misma [13].

¹² Es la cuarta versión del protocolo Internet Protocol (IP) y es la primera en ser implementada a gran escala, además de utilizar direcciones de 32 bits [14].

¹³ Sexta versión de protocolo IP, destinado a substituir al IPV4 ya que permite 2¹²⁸ direcciones de red [15].

nombre lo indica fue creado pensando en que sería utilizado dentro de un entorno con la misma filosofía, es decir un sistema operativo de licencia libre. En este caso en particular se tomo la decisión de utilizar Ubuntu debido a que es un sistema operativo basado en Linux y cuya distribución es ampliamente conocida, la cual además de contar con una amplia documentación que facilita su uso cuenta con un gran soporte de parte de la comunidad, por otro lado es un sistema operativo sencillo de instalar y mantener actualizado características que lo hacen ideal para instalar OpenERP.

2.5.1 Cliente-Servidor

En este momento vale la pena aclarar, aunque sea un poco redundante, que la arquitectura en la cual está basada OpenERP es una arquitectura del tipo cliente-servidor y que al momento de realizar la instalación hay una versión del servidor sin embargo se cuenta con 2 opciones para el cliente. Una de ellas es la versión GTK y la otra es la versión web. La principal ventaja del modelo cliente-servidor radica en su versatilidad y fácil comprensión.

³⁵₁₇ **Versatilidad:** Este modelo es muy versátil ya que nos indica que sin importar el tipo de interfaz gráfica que se desee utilizar, el servidor debe cumplir con las peticiones del cliente y esto permite elegir entre una o varias versiones del tipo de cliente que se vaya a utilizar.

³⁵₁₇ **Fácil Compresión:** Ya que este modelo únicamente está integrado por dos elementos, el cliente y el servidor, y las tareas para ambos están claramente definidas, esto permite que pueda ser comprendido cualquier programador.

2.5.2 Sobre OpenERP-Ubuntu

El sistema operativo del servidor sobre el que se instala OpenERP es recomendable que sea un sistema operativo tipo UNIX o de preferencia una distribución de licencia libre. Esto no quiere decir que OpenERP no pueda ser instalado en un servidor cuyo sistema operativo no sea de licencia libre, como Windows, sin embargo se han detectado múltiples problemas de compatibilidad que pueden ser resueltos sin mayor problema, pero dada la naturaleza de esta tesis las ventajas que conlleva utilizar un sistema operativo de licencia libre son suficientes como para decidir instalar OpenERP sobre Ubuntu¹⁴ Server 10.10. Algunas de las principales ventajas, como ya se había mencionado con anterioridad son:

³⁵₁₇ **Fácil Instalación:** Para adquirir una copia de esta distribución basta con solicitarla vía correo electrónico o descargarla desde la página oficial de Ubuntu. El proceso de instalación no requiere conocimientos avanzados sobre sistemas operativos y si permite definir de manera sencilla los componentes que se desean instalar.

³⁵₁₇ **Actualizaciones:** Las actualizaciones de esta distribución se realizan de manera constante y cuentan con amplia documentación y soporte de parte de la comunidad. Además dichas actualizaciones son realmente sencillas de llevar a cabo sin mayores dificultades.

³⁵₁₇ **Módulos y Bibliotecas:** Al momento de llevar a cabo la instalación, se puede decidir cuales módulos y bibliotecas se desea instalar, lo cual es muy útil ya que para el correcto funcionamiento y desarrollo de OpenERP son necesarias algunos módulos y bibliotecas en particular.

La combinación entre OpenERP y Ubuntu fue la decisión obvia después de haber investigado un poco al respecto, existe basta documentación sobre la interacción entre OpenERP y otras distribuciones de licencia libre como Fedora o Mandriva, por mencionar algunas. Sin embargo dado lo amigable de Ubuntu y la gran comunidad de usuarios y desarrolladores de Ubuntu, así como de documentación y manuales decidí utilizar *Ubuntu Server 10.10* para instalar ambos servidores de OpenERP. Se tratara de manera más detalla este tema al momento de describir el proceso de instalación

¹⁴ *Es una distribución GNU/Linux basada en Debian GNU/Linux que proporciona un sistema operativo actualizado y estable para el usuario medio, con un fuerte enfoque en la facilidad de uso e instalación del sistema. Al igual que otras distribuciones se compone de múltiples paquetes de software normalmente distribuidos bajo una licencia libre o de código abierto. Estadísticas web sugieren que el porcentaje de mercado de Ubuntu dentro de las distribuciones Linux es de aproximadamente 50%, y con una tendencia a subir como servidor web [16].*

Capítulo 3: Planteamiento del Problema.

Uno de los objetivos de esta tesis es mostrar paso a paso cuales procedimientos formaron parte del desarrollo del sistema de facturación digital que sea funcional y estable. A fin de tener un mejor control de dichos procedimientos fue necesario el uso de técnicas de ingeniería de software para lo cual es necesario primero hacer un planteamiento del problema y así mismo plantear una solución.

3.1 Contexto General

El concepto de facturación digital en México es un tema relativamente reciente y surge para acabar con viejos vicios y desventajas de la factura tradicional. La facturación digital busca agilizar la expedición y procesos asociados a la generación de una factura, así como reforzar los elementos de seguridad que hacen de una factura un elemento valido para fines fiscales. Lo escrito en el apartado 1.1. de esta tesis así como lo antes mencionado justifican el por qué es necesario implementar un sistema de facturación digital, además es complementado por el hecho de que al implementar un sistema de facturación digital, se obliga al contribuyente a emitir facturas en tiempo real y así acabar con viejas costumbres y malos hábitos, además de que mediante este sistema, el SAT de manera conjunta con el contribuyente pueden llevar una mejor organización de las facturas y a su vez de las declaraciones que este tenga que realizar, de manera tal que al final es un proceso benéfico para ambas partes.

¿Quiénes pueden emitir factura electrónica?

Podrán optar por emitir factura electrónica las personas físicas y las personas morales que deseen optimizar, agilizar y automatizar sus procesos para emitir facturas. Las personas físicas y las personas morales que cuenten con su firma electrónica avanzada vigente y que lleven su contabilidad en sistemas electrónicos. (Fundamento legal: Artículo 29 noveno párrafo del CFF).

¿Qué requisitos necesito para implementar la factura electrónica?

³⁵~~17~~ Contar con un Certificado de Firma Electrónica Avanzada vigente.

³⁵~~17~~ Contar con uno o más certificados de sello digital.

³⁵~~17~~ Contar con un determinado rango de folios asignados por el SAT.

³⁵~~17~~ Tener una contabilidad simultanea.

³⁵~~17~~ Envío de reporte mensual.

³⁵~~17~~ (Fundamento legal: Artículo 29 noveno párrafo del Código Fiscal de la Federación)

³⁵~~17~~ Conocer modelo de negocio

¿Qué es una contabilidad simultánea?

Que al asignarse el folio, y en su caso serie, el sistema electrónico en que se lleve la contabilidad efectúe simultáneamente el registro contable en cuentas y subcuentas afectadas por cada operación.

¿Qué es un certificado digital?

Es un archivo con extensión *.cer que asocian una llave pública con la identidad de su propietario. Existen dos tipos de certificados digitales: los certificados de *Firma Electrónica Avanzada* y el certificado de sello digital.

¿Qué es un certificado de Firma Electrónica Avanzada?

Tipo de certificado digital que identifica a personas físicas o morales y da poder para obtener *Certificados de Sello Digital, Solicitud de Folios* y para firmar el *Reporte Mensual*, entre otros servicios.

¿Qué es un certificado de sello digital?

Tipo de certificado digital para uso exclusivo de los comprobantes fiscales digitales.

¿Cuál es la relación de la firma electrónica avanzada con la emisión de facturas electrónicas?

La firma electrónica avanzada juega un papel crucial en la facturación electrónica, ya que sin ella las facturas electrónicas no pueden ser generadas

¿En qué consiste la solicitud de folios?

La solicitud de folios consiste en proporcionar al contribuyente folios válidos para usarlos en su proceso de facturación electrónica. Para solicitar dichos folios, deberá ingresar a la página web del Sistema Integral de Comprobantes Fiscales (SICOFI). Una vez dentro del portal, deberá realizar lo siguiente:

³⁵₁₇ Ingrese su RFC, Seleccione su serie y rango de folios.

³⁵₁₇ Indique la ruta o localización del certificado de Tu Firm@ (Firma Electrónica Avanzada).

³⁵₁₇ Indique la ruta o localización de su llave privada (*.key) asociada a dicho certificado de Tu Firm@.

Al terminar dicho proceso, usted contará con uno o varios rangos de folios asignados por el SAT.

¿Cómo se obtienen los folios para facturas electrónicas?

Todos los contribuyentes que opten por la emisión de facturas electrónicas deberán tramitar un certificado de sello digital, así como una solicitud de rango de folios. La solicitud de folios consiste en proporcionar al contribuyente folios válidos (iniciando con el folio 1) para usarlos en su proceso de facturación. - Para solicitar dichos folios, deberá ingresar a la página web del Sistema Integral de Comprobantes Fiscales (SICOFI). Una vez dentro del portal, deberá realizar lo siguiente:

³⁵₁₇ Ingrese su RFC.

³⁵₁₇ Seleccione su serie y rango de folios.

³⁵₁₇ Indique la ruta o localización del certificado de Tu Firm@ (Firma Electrónica Avanzada).

³⁵₁₇ Indique la ruta o localización de su llave privada (*.key) asociada a dicho certificado de Tu Firm@.

Al terminar dicho proceso, usted contará con uno o varios rangos de folios asignados por el SAT.

¿Cuándo y a través de qué medio se debe proporcionar al SAT la información de las facturas electrónicas que se hayan expedido?

Se deberá proporcionar mensualmente al *Servicio de Administración Tributaria*, a través de medios electrónicos, la información correspondiente a las facturas electrónicas que se hayan expedido con los folios asignados utilizados en el mes inmediato anterior a aquél en que se proporcione la información. Consulte la sección de reporte de movimientos al SAT para obtener más información. (Fundamento legal: Artículo 29 fracción III inciso c. del Código Fiscal de la Federación).

¿Cómo se genera la FEA o sello digital?

Para la generación de sellos digitales se utiliza criptografía de clave pública aplicada a una cadena original. Los pasos para generarlos son:

³⁵₁₇ Generar cadena Original en formato UTF-8.

³⁵₁₇ Generar la función hash (resumen ó digestión) a partir de la cadena original utilizando el algoritmo MD5 (función de un solo sentido tal que para cualquier entrada produce una salida compleja de 16 bytes denominada "digestión").




³⁵₁₇ Encriptar la función hash utilizando el algoritmo RSAPrivateEncrypt y la llave privada del emisor. Para desencriptar se utiliza el algoritmo RSAPublicDecrypt y la llave publica del emisor.

³⁵₁₇ Convertir La encriptación a base 64. Para mayor especificación dirigirse al anexo 20 del DOF 1 septiembre, 2004. en su rubro D.

¿Qué es la cadena original?

Se entiende como cadena original, a la secuencia de datos formada con la información contenida dentro del comprobante fiscal digital, establecida en el Rubro C “Estándar de comprobante fiscal digital extensible” de este anexo. Siguiendo para ello las reglas y la secuencia especificadas en el Anexo 20 del DOF 1 septiembre, 2004 en su rubro D.

A partir de la reforma del Código Fiscal de la Federación el 28 de junio de 2006, se establecieron las bases de regulación para la prestación de servicios de emisión y envío de comprobantes fiscales digitales. Con esa reforma y con la publicación de las reglas específicas en meses posteriores en la Resolución Miscelánea Fiscal, el SAT anuncia tres formas de facturar electrónicamente en México, a saber:

-  **Facturación por medios propios:** consiste en la generación de facturas en las instalaciones de la empresa emisora. Esto puede hacerse utilizando un software desarrollado internamente o una aplicación desarrollada por un tercero, pero operada por personal de la empresa emisora.
-  **Facturación por medio de un proveedor autorizado por el SAT para proveer el servicio de emisión y entrega de Comprobantes Fiscales Digitales:** consiste en la emisión y entrega de comprobantes fiscales digitales por parte de una entidad fuera del domicilio fiscal de la empresa, por medios electrónicos y de manera completamente digital, sin que por ello se considere que se lleva la contabilidad fuera del domicilio mencionado. La entidad debe contar con la autorización y certificación de procesos por parte del Servicio de Administración Tributaria (SAT) para generar y procesar facturas. Con esta modalidad los emisores en poco tiempo utilizan las funcionalidades del servicio ofrecido que se ajusten a sus procesos o necesidades, sin invertir en el costo total de un producto y con la certeza del apego a la normativa fiscal en todo momento. Además del proceso de emisión, la certificación que brinda el SAT a proveedores especializados incluye también procesos de entrega, lo que facilita integrar comunidades de colaboración electrónica entre clientes y proveedores.
-  **Facturación por medio de la aplicación gratuita del SAT: Micro-E:** Diseñado para personas físicas y morales dedicadas a actividades empresariales, prestación de servicios profesionales o arrendamiento de bienes inmuebles cuyos ingresos anuales no son mayores de cuatro millones de pesos. Este servicio no tiene costo. Es posible además, llegar el control de las operaciones y las obligaciones fiscales

. En el paquete de reformas al Código Fiscal de la Federación (CFF) 2010, aprobado por la Cámara de Diputados y publicado por el Diario Oficial de la Federación el 7 de diciembre de 2009, incluye las modificaciones en materia de comprobantes fiscales que están en vigor.

Obligatoriedad de la Factura Electrónica.

La cámara de Senadores Aprobó en todos sus términos las modificaciones al Código Fiscal de la Federación para hacer obligatoria la Factura Electrónica para todas las facturas superiores a 2 mil pesos a partir de Enero del 2011. Obligatoriedad de la Factura Electrónica.

En esta dirección electrónica se podrá encontrar la última versión del documento conocido como anexo 20 en el cual se especifican los estándares requeridos por el sat para la generación de facturas electrónicas.

ftp://ftp2.sat.gob.mx/asistencia_servicio_ftp/publicaciones/cfd/Anexo20RMF2010.pdf

Lista de modificaciones y fechas al Anexo 20

- ³⁵/₁₇ Anexo 20 de la Miscelánea Fiscal 2006
- ³⁵/₁₇ Anexo 20 de la Miscelánea Fiscal 2009
- ³⁵/₁₇ Anexo 20 de la Miscelánea Fiscal 2010

2.2 Definición del problema

Como ya es bien conocido el desarrollo de un sistema es un proceso arduo, complejo y que demanda una gran capacidad de abstracción y de organización, elementos sin los cuales el sistema resultante no podrá más que ser un fracaso y una pérdida de tiempo. Es por esto que en esta tesis se mostrará una forma de desarrollar un sistema de calidad que sea capaz de emitir facturas digitales y algunos otros documentos que tendrán validez fiscal ante cualquier autoridad que así lo requiera.

De manera general el objetivo de esta tesis es generar un manual de desarrollo que sirva como base o como guía para todas aquellas empresas que cuenten con un equipo de desarrollo informático y que encuentren en la facturación digital y en los sistemas tipo ERP, en particular OpenERP una solución coherente a las desventajas antes mencionadas que conlleva seguir haciendo uso de la manera tradicional de facturar. Además de ofrecerles una mejor manera de administrar y manejar su empresa utilizando software libre

Así que en pocas palabras y a manera de listado las funcionalidades del sistema que se pretende describir y desarrollar en esta tesis son las siguientes:

- 1 Ingresar al sistema.
 - 1.1 Registrarse en el sistema.
 - 1.2 Ingresar en el sistema.
 - 1.2.1 Iniciar sesión.
 - 1.2.2 Cerrar sesión.
- 2 Administración de Usuarios.
 - 2.1 Alta de usuarios.
 - 2.1.1 Asignación de Grupo.
 - 2.1.2 Creación de Grupo.
 - 2.1.3 Asignación de Permisos.
 - 2.2 Modificación de usuarios.
 - 2.2.1 Modificación de Permisos.
 - 2.3 Baja de usuarios
- 3 Administración de Productos.
 - 3.1 Registrar productos.
 - 3.2 Cambiar características de los productos.
 - 3.3 Dar de baja productos.
 - 3.4 Inventario de productos.
 - 3.4.1 Eliminar de la base la cantidad de productos que hayan sido vendidos.
 - 3.4.2 Al momento de dar de alta productos, actualizar el inventario.
- 4 Generar Facturas.
 - 4.1 Capturar los datos asociados a una factura con múltiples productos.
 - 4.2 Realizar cálculos necesarios para generar la factura.
 - 4.3 Generar elementos de seguridad asociados a la factura en cuestión.
 - 4.3.1 Sello digital.
 - 4.3.2 Numero de autorización.
 - 4.4 Crear elementos adicionales de una factura digital.
 - 4.4.1 Cadena original (resumen de los elementos de la factura en formato .txt).
 - 4.4.2 Archivo .xml (mismos elementos de la cadena original pero en formato .xml).
 - 4.4.3 Archivo .pdf (producto final y factura lista para ser entregada en formato .pdf).
 - 4.5 Sellar de manera digital la factura.
 - 4.6 Generar Factura Digital.
 - 4.7 Almacenar Facturas Generadas.
 - 4.8 Imprimir Factura Digital.
 - 4.9 Enviar Factura Digital.

Para tales fines será necesario dividir las tareas en ciclos y cada ciclo estará compuesto de tareas especificadas en el Product Backlog, ya que como SCRUM así lo marca, es necesario tomar dichas tareas del Product Backlog para poder llevar un control de dichas tareas y a su vez poder actualizar de manera dinámica dicho documento. Será necesario generar un par de ciclos para instalar el ambiente y el resto para la implementación del modulo de facturación.

Capítulo 4: Instalación del ambiente de desarrollo

Es tema relevante de este capítulo ahondar en el tema de los alcances y limitaciones del sistema que se desarrollara a lo largo de esta tesis, al cual a partir de este capítulo y en adelante llamaremos **SiFaDig** (Sistema de Facturación Digital). Como ya se ha mencionado con anterioridad, el eje focal de esta tesis es el desarrollo de un sistema de facturación que sea capaz principalmente de generar facturas digitales con todos los elementos necesarios para que dichas facturas sean válidas para cualquier efecto fiscal y administrativo, principalmente ante el SAT (Sistema de Administración Tributaria), ya que éste es el encargado de regular el proceso de facturación en México, y es con el SAT con el que SiFaDig tendrá una gran interacción tanto para validar las facturas generadas como para enviar reportes y recibir los folios necesarios para la generación de facturas electrónicas.

Al desarrollar cualquier sistema informático, siempre es necesario partir de funcionalidades básicas, como: ingresar al sistema, administración de usuarios, inventarios, etc. y con base en estas funcionalidades básicas comenzar el desarrollo, cabe mencionar que este proceso se acentúa más aun debido a la metodología elegida para el desarrollo de SiFaDig, que en este caso es SCRUM, ya las reglas de SCRUM marcan que las tareas se dividen en pequeños requerimientos y a partir de ellos se incrementa el desarrollo. Sin embargo debido a que estas funcionalidades básicas y algunas otras ya han sido ampliamente examinadas y optimizadas a lo largo del desarrollo de software se decidió dejar de lado el análisis y el desarrollo de las mismas al sistema conocido como OpenERP, para dedicarnos de fondo al tema relevante de esta tesis como generar una factura digital.

4.1 Especificación de requerimientos

Una vez terminado SiFaDig, contará con todas las características de la lista descrita en el capítulo 3, es decir, contará con todos los elementos necesarios para poder generar una factura digital, así como para poder llevar a cabo todas las labores administrativas que así lo requieran. Sin embargo, y dado que la unidad básica de SCRUM es el ciclo, para el primer ciclo de este desarrollo únicamente bastará con describir los pasos para la instalación de OpenERP y como generar usuarios.

4.2 Análisis del problema

Al realizar el análisis necesario para el desarrollo de SiFaDig, se requirió realizar una investigación completa y detallada de los elementos que involucran la generación de una factura digital. Fue necesario recabar información por diversos medios como: cursos en las oficinas del SAT, asesoría telefónica y vía chat, para poder entender e identificar los diversos elementos que entran en juego¹⁵.

4.2.1 Definición de Actores y Casos de Uso.

El primer paso es identificar a los actores que formaran parte de este sistema. Del capítulo anterior, podemos darnos cuenta que surgen 3 actores importantes: el administrador, el usuario y el SAT así como algunas funcionalidades obvias del sistema, lo que da como resultado un diagrama donde se modela de manera gráfica la interacción entre estos actores y las funcionalidades que realizan con el sistema.

¹⁵ Se dará por sentado que el lector de esta tesis cuenta con algunos de estos elementos fundamentales para generar una factura electrónica como la FEA, ya que inicialmente es competencia de esta tesis guiar al lector en el proceso para generar un sistema capaz de generar facturas electrónicas y no el cómo realizar todos los procesos necesarios para darse de alta ante el SAT, sin embargo vale la pena mencionar que, sin elementos como la FEA o el CFD, no es posible generar facturas electrónicas.

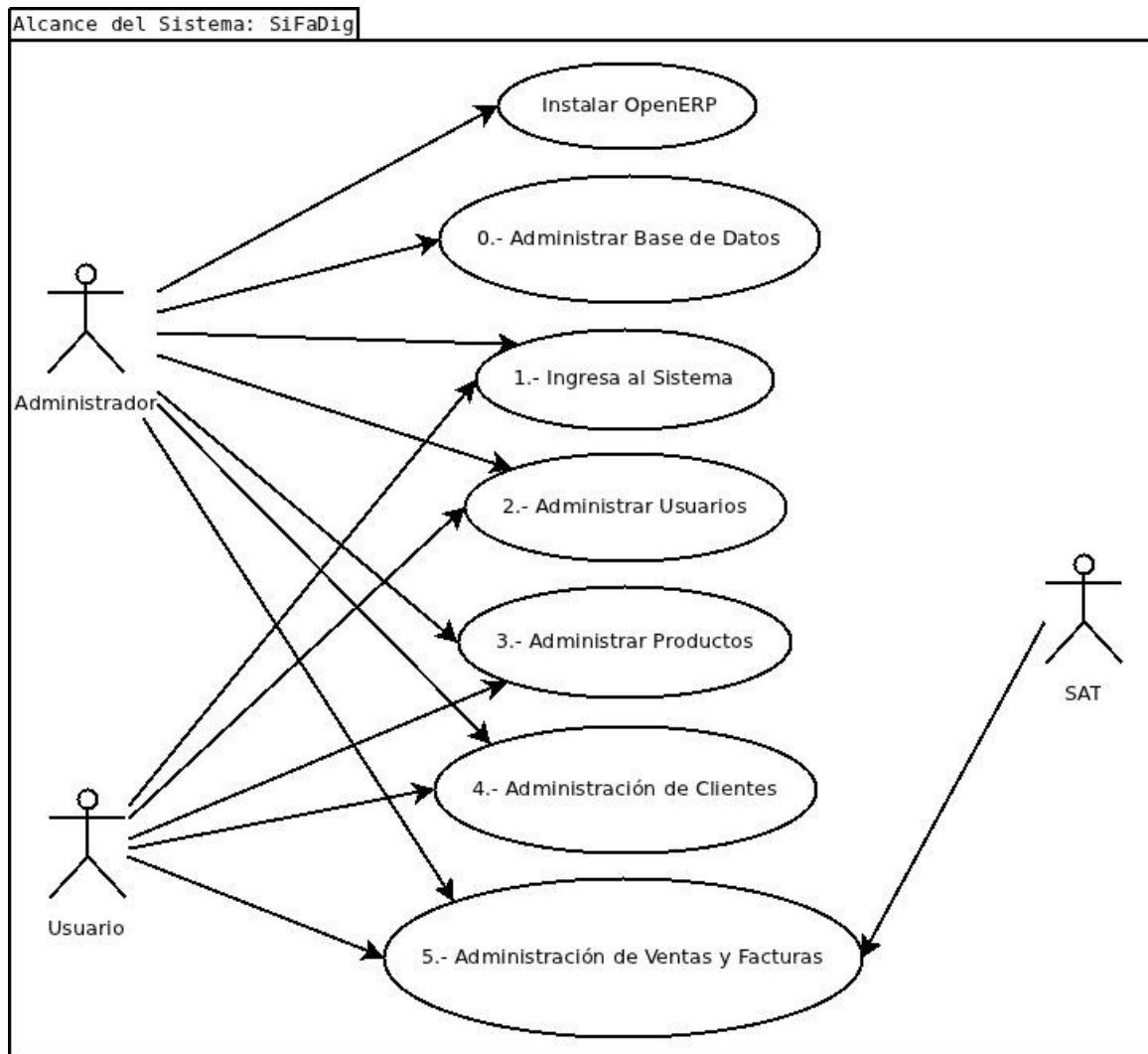


Figura 4.1. Diagrama de casos de uso General

Como se puede apreciar en la Figura 4.1 se muestra la totalidad de las funciones que debe llevar a cabo SiFaDig, sin embargo como ya se mencionó con anterioridad lo ideal es descomponer este diagrama en módulos más pequeños que puedan ser analizados e implementados de manera sencilla.

Aquí es importante recalcar que el ciclo 0 debe ser llevado a cabo por el administrador del sistema, ya que él será el encargado de administrar la base de datos y será el único con los privilegios adecuados para poder realizar modificaciones a la base de datos. En el ciclo 0 se analizarán todas las tareas relacionadas con la administración de la base de datos, desde la creación de la misma hasta la creación de usuarios.

Se hará uso de UML¹⁶ para los diagramas de casos de uso, así como para los diagramas de clases y diagramas de flujo.

De la Figura 4.1 se puede apreciar que la funcionalidad básica y por lo mismo la primera que deberá implementarse primero es el acceso al sistema, es decir, lo correspondiente al siguiente subdiagrama.

¹⁶ **Lenguaje Unificado de Modelado** (LUM o UML, por sus siglas en inglés, Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad [17].

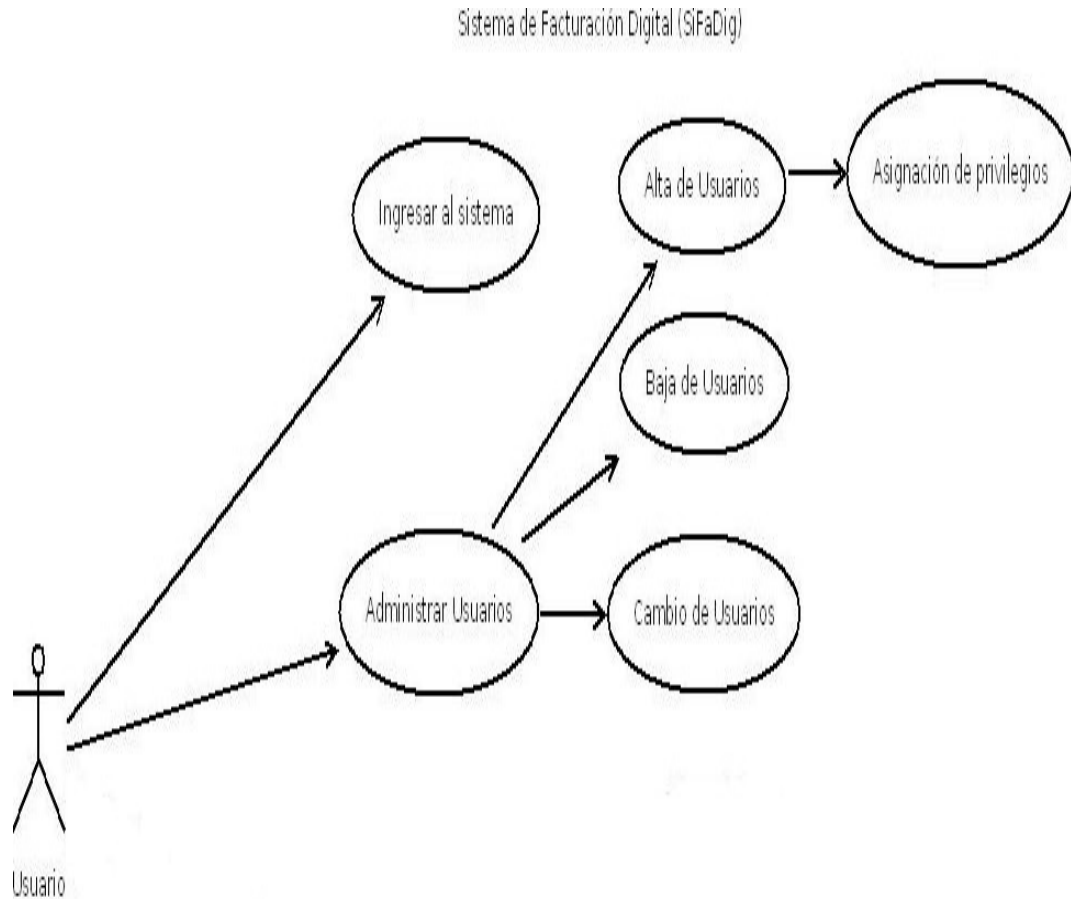


Figura 4.2 Casos de uso: Ingresar al Sistema y Administrar Usuarios

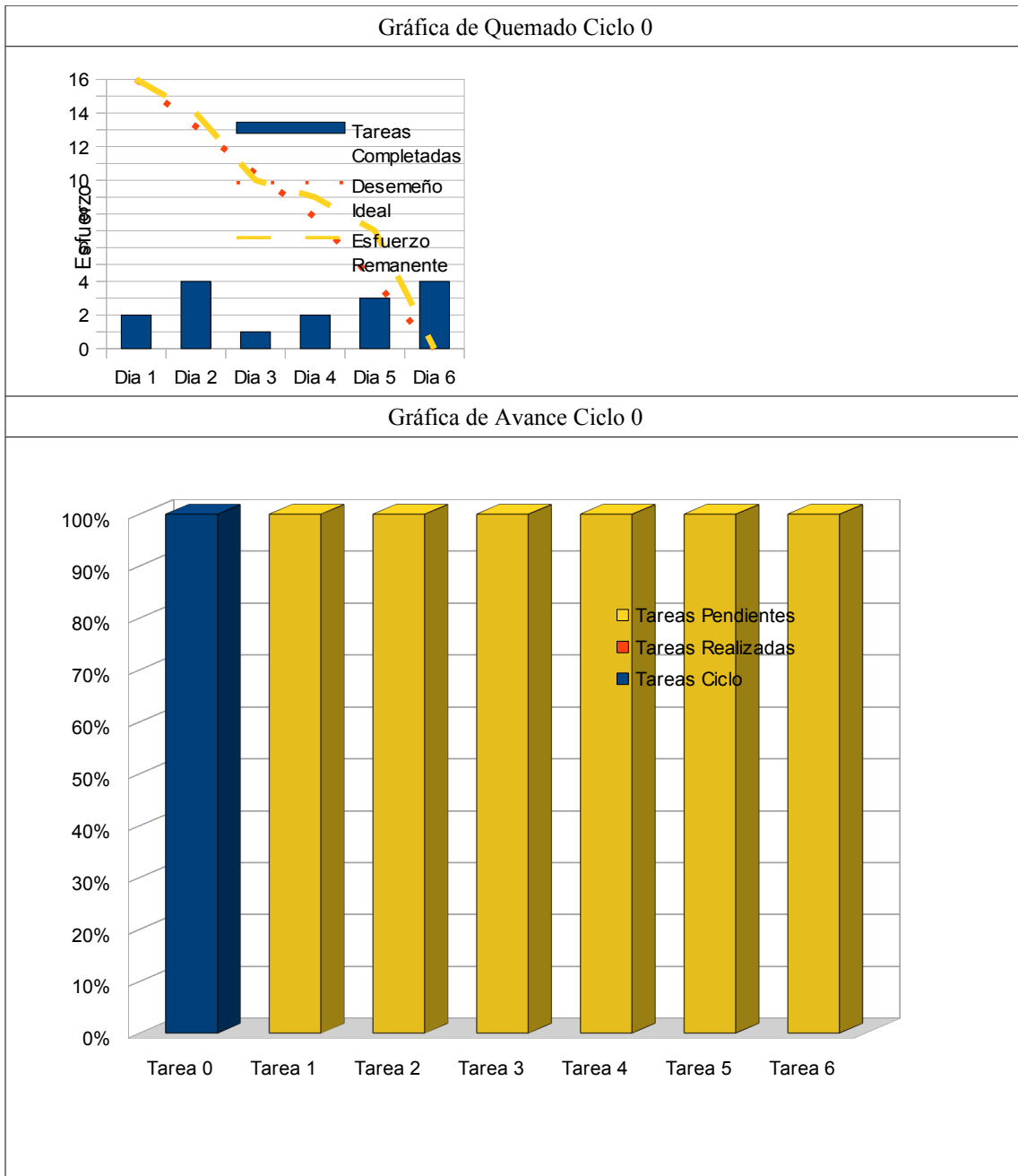
Como se puede ver en la Figura 4.2, la funcionalidad de Ingresar al sistema, obliga a generar un módulo de administración de usuarios en el cual se pueda llevar a cabo altas, bajas y cambios, así como asignación de privilegios a los usuarios.

Es bien conocido en el ámbito informático que siempre es de gran utilidad tener diferentes tipos de usuarios, así como privilegios en el sistema ya que esto conforma la base para que cada usuario tenga acceso únicamente a lo que le corresponde y así se puede tener una relación directa con las responsabilidades que tiene cada usuario tanto dentro del sistema como fuera de él. Corresponde a este capítulo generar un primer ciclo de SCRUM en el cual se definan clases, métodos y demás objetos que tendrán un papel importante dentro del caso de uso Ingresar al Sistema y Administrar usuarios.

4.3 Ciclo 0: Instalación de OpenERP

Este ciclo será utilizado para mostrar los pasos que hay que seguir para poder instalar OpenERP, desde la descarga de los paquetes necesarios hasta la instalación de las dependencias y/o bibliotecas necesarias para que tanto el servidor como el cliente de OpenERP puedan desempeñarse de manera adecuada.

Sprint Backlog: SiFaDig			
Ciclo 0: Instalación OpenERP	<p>Objetivos Generales: En este ciclo inicial, se busca generar el ambiente adecuado para que el administrador pueda iniciar con la gestión del sistema. Además se busca que este ciclo sienta las bases para poder desarrollar el módulo correspondiente a la facturación digital, ya que es necesario conocer la forma en la que se trabaja dentro de OpenERP. El tiempo estimado de duración de las tareas marcadas a continuación es de una semana y se espera que al término de estas tareas se pueda comenzar con el desarrollo necesario. Además es prioridad de este ciclo crear el ambiente de desarrollo apoyándose en un IDE, que para este caso será Eclipse.</p> <p>Puntos Importantes:</p> <ul style="list-style-type: none"> ³⁵/₁₇ Instalar OpenERP. ³⁵/₁₇ Configurar Ambiente de desarrollo 		
Fecha de Inicio: 14/03/2011			
Fecha de Término: 20/03/2011			
Tareas			
Id	Descripción de la Tarea	Tamaño	Prioridad
0	Crear Ambiente de desarrollo	3	1
0.1	Instalación OpenERP	2	1
0.1.1	Configuración PostgreSQL	1	1
0.1.2	Instalación de Bibliotecas	2	1
0.2	Configurar Ambiente de desarrollo	4	1
0.2.1	Descargar Eclipse	1	1
0.2.2	Configurar Eclipse	3	1



Tarea 0.1: instalación OpenERP

En esta sección de la tesis se describirá paso a paso el procedimiento que hay que seguir para poder dar comienzo al desarrollo del módulo de *Facturación Electrónica* que será utilizado en conjunto con OpenERP. Primero se mostrará la forma en la cual se puede comenzar con el desarrollo desde un block de notas¹⁷ y posteriormente cómo desarrollar módulos para OpenERP desde un IDE, que en este caso será Eclipse.

La versión con la cual se trabajará a lo largo de esta tesis es la versión más actual de OpenERP disponible hasta el momento que es la versión 6.0.1 tanto del cliente como del servidor. Es importante recalcar que OpenERP maneja 2 tipos de clientes:

- ✦ **Cliente GTK:** Este cliente es el cliente comúnmente conocido en las aplicaciones de escritorio y que se encarga de realizar las peticiones al servidor de OpenERP. Este cliente cuenta con algunas ventajas y desventajas sobre el cliente web.

¹⁷ Editor de texto básico, que únicamente tiene como finalidad la escritura de caracteres sin las ventajas que ofrece un editor de textos más especializado.

- ✧ **Cliente Web:** Este cliente es el encargado de realizar las peticiones del usuario desde una interfaz web, es decir vía algún navegador. De igual manera que el cliente *GTK*, cuenta con algunas ventajas y desventajas sobre este, siendo una de las más importantes, el hecho de que no es necesario instalar el cliente, únicamente se ejecuta y se accede a este mediante el navegador.

Uno de los puntos más fuertes de OpenERP es su versatilidad y dinamismo, además la interfaz gráfica es una de sus grandes ventajas, ya que sin importar cual de los 2 tipos de clientes que se elija, la funcionalidad y eficiencia de OpenERP se mantienen sin sufrir cambios y la experiencia es prácticamente la misma sin importar el cliente. En realidad la elección del tipo de cliente tiende a depender más de las necesidades del usuario y de con cual de ellos se sienta más cómodo. En esta tesis únicamente se mostrará el procedimiento para la implementación del módulo de facturación electrónica destinado al cliente web. Ya que OpenERP está regido por la arquitectura MVC, la teoría nos dicta que al final de cuentas el módulo desarrollado en esta tesis debe funcionar perfectamente para el cliente *GTK*, lo único que haría falta sería el desarrollo de la interfaz gráfica, sin embargo ese es un tema que no será tratado en esta tesis principalmente por el alcance de la misma.

Descargando el código fuente y creado el espacio de trabajo

Para descargar el código fuente de OpenERP es necesario ingresar al siguiente sitio web <http://www.openerp.com/downloads> y seleccionar de la sección **Sources & Packages** el código fuente tanto del servidor como del cliente. En esta sección de descargas se puede encontrar la versión web como la versión *GTK* del cliente, sin embargo para fines de esta tesis nos enfocaremos en el desarrollo para la versión web de OpenERP.

Una vez descargados ambos paquetes fuente procedemos a crear una carpeta a la que se puede asignar el nombre que queramos, se le asignó el nombre de “**OperERP**”, dentro de esta carpeta procedemos a descomprimir ambos paquetes fuentes, de tal manera que tengamos dos carpetas, una llamada “**openerp-client-6.0.1**” y la otra “**openerp-web-6.0.1**” ambas dentro de la carpeta recién creada. Una vez realizados los pasos antes descritos, procedemos con la Instalación del SMBD.

Ubuntu Server 10.10

Al momento de la Instalación de *Ubuntu Server 10.10* podemos elegir 3 paquetes que son esenciales para OpenERP y esto son:

³⁵₁₇ **LAMP server** (es necesario ya que aquí viene Apache).

³⁵₁₇ **OpenSSH server** (para administración del servidor de manera remota).

³⁵₁₇ **PostgreSQL database** (es el manejador de bases de datos que se utiliza en OpenERP).

Sin embargo en caso de no haber seleccionado estos paquetes al momento de la instalación, éstos pueden ser instalados de manera muy sencilla mediante línea de comandos o utilizando el **gestor de Instalación de Ubuntu**. Dado que el SMBD requiere crear un usuario para poder administrar las bases de datos asignadas a OpenERP, procedo con la descripción de los pasos a seguir suponiendo que no se eligió el paquete *PostgreSQL database* al momento de la instalación de *Ubuntu Server 10.10*.

Instalado PostgreSQL

El primer componente necesario para poder ejecutar y posteriormente desarrollar módulos para OpenERP es el SMBD, que como ya se había mencionado con anterioridad en este caso es PostgreSQL. Así que primero tenemos que instalar dicho SMBD, se teclea el siguiente comando:

```
sudo apt-get install postgresql
```

Este comando instalará los paquetes necesarios para poder hacer uso de PostgreSQL como manejador de las bases de datos. Al ejecutar este comando el gestor de Instalación de *Ubuntu* busca e instala la versión más reciente que se encuentre en sus repositorios, que en este caso es la versión 8.4

Para la versión 6.0.1 de OpenERP no se permite lanzar el servidor con el usuario *postgres*¹⁸, por lo que lo mejor es crear un usuario mediante el cual podamos lanzar el servidor, esto lo hacemos mediante los siguientes comandos.

```
usuario@maquina$ sudo useradd openerp
usuario@maquina$ sudo passwd openerp
```

Con el primero comando agregamos un nuevo usuario a nuestro sistema y con el segundo comando le asignamos una contraseña, se nos pedirá ingresar la contraseña y confirmar la misma. Una vez hecho lo anterior procedemos a crear un usuario en *postgresql* con el cual podamos crear la base necesaria para poder levantar el servidor de OpenERP. Para crear dicho usuario y asignarle los permisos necesario tecleamos la siguiente combinación de comandos.

```
usuario@maquina$ sudo su - postgres -c "createuser --createdb --no-createrole
--pwprompt openerp"
```

Mediante esta combinación de comandos creamos el usuario *openerp* dentro de *postgresql*, con el modificador *--createdb* le damos permiso para crear bases de datos, mediante el segundo modificador del comando *--no-createrole* indicamos que este usuario no puede crear roles y mediante el último modificador *--pwprompt* le asignamos de una vez la contraseña. Al teclear *enter*, se pide ingresar la contraseña para el nuevo rol, también se pide confirmar la misma y por último se nos pregunta si el nuevo rol será un superusuario, a lo que respondemos con la tecla "s" y presionamos "enter". Ya con el SMBD instalado y conFigurado el siguiente paso es instalar los paquetes necesarios para poder desarrollar módulos para OpenERP.

Paquetes y Bibliotecas

Como ya se ha mencionado con anterioridad OpenERP está escrito en *Python*, así que para poder realizar modificaciones a los módulos existentes y desarrollar los módulos que deseemos es necesario llevar a cabo la instalación de las siguientes bibliotecas y paquetes. Es importante mencionar que en esta tesis únicamente se mencionarán los componentes necesarios para el desarrollo, para mayor detalle de las bibliotecas y de los paquetes que son requeridos, se puede encontrar basta documentación en la red.

Las bibliotecas necesarias para la ejecución de OpenERP se listan a continuación:

```
35 17 python (Viene con Ubuntu Server 10.10).
35 17 postgresql (Lo instalé antes, con el instalador de Ubuntu Server).
35 17 psycopg2 (Módulo de python).
35 17 reportlab (Librería de python para generar PDFs).
35 17 lxml (Módulo de python).
35 17 pytz (Módulo de python).
35 17 pyYaml (Módulo de Python).
35 17 vobject (Módulo de Python).
```

Además es necesario instalar los siguientes paquetes para compilar y para satisfacer algunas de las dependencias tanto en el servidor y en el cliente web de OpenERP:

```
35 17 build-essential.
35 17 Python-setuptools.
35 17 Python-profiler.
35 17 python-pastedeploy.
```

```
$ sudo apt-get install -y postgresql graphviz python-psycopg2 python-lxml python-tz
python-imaging python-pychart python-egenix-mxdatettime build-essential python-setuptools
python-profiler python-pastedeploy python-vobject
```

¹⁸ Es el usuario por default después de instalar *postgresql*.

```
$ sudo easy_install PyYaml
```

Con todo lo necesario ya instalado, ahora únicamente resta probar que todo funcione como se espera, es decir podamos ejecutar tanto el servidor como el cliente de OpenERP.

Levantando el servidor

Lo primero es cambiar al usuario “**openerp**” mediante el siguiente comando.

```
$sudo su openerp
```

Posteriormente en la carpeta en la cual se haya descomprimido el código fuente del servidor openerp y se busca la carpeta *bin*.

```
$cd/home/usuario/openerp/openerp-server/bin
```

Por último únicamente se ejecuta el archivo “**openerp-server.py**”.

```
$. /openerp-server.py
```

Mediante esta secuencia de comandos se puso en funcionamiento el servidor de **openerp** y tendremos acceso a la base de datos mediante el usuario “**openerp**”. Después de haber ejecutado estos comandos la consola debe lucir muy similar a la *Figura 0.1*.

```

Archivo Editar Ver Buscar Terminal Ayuda
mike@mike-laptop:~$ sudo su openerp
$ cd Documentos/Tesis/OpenERP/Desarrollo/openerp-server-6.0.1/bin
$ ./openerp-server.py
[2011-02-21 10:28:02,829][?] INFO:server:OpenERP version - 6.0.1
[2011-02-21 10:28:02,829][?] INFO:server:addons_path - /home/mike/Documentos/Tesis/OpenERP/Desarrollo/openerp-server-6.0.1/bin/addons
[2011-02-21 10:28:02,829][?] INFO:server:database hostname - localhost
[2011-02-21 10:28:02,830][?] INFO:server:database port - 5432
[2011-02-21 10:28:02,830][?] INFO:server:database user - openerp
[2011-02-21 10:28:02,830][?] INFO:server:initialising distributed objects services
[2011-02-21 10:28:04,250][?] INFO:web-services:starting HTTP service at 0.0.0.0 port 8069
[2011-02-21 10:28:04,251][?] INFO:web-services:starting HTTPS service at 0.0.0.0 port 8071
[2011-02-21 10:28:04,251][?] INFO:web-services:Registered XML-RPC over HTTP
[2011-02-21 10:28:04,287][?] INFO:web-services:starting NET-RPC service at 0.0.0.0 port 8070
[2011-02-21 10:28:04,288][?] INFO:server:Starting 3 services
[2011-02-21 10:28:04,290][?] INFO:server:OpenERP server is running, waiting for connections...

```

Figura 4.2 Consola que muestra que el servidor de OpenERP ha iniciado de manera exitosa

En este momento el servidor openerp está en funcionamiento y esperando solicitudes del cliente openerp.

Iniciando el cliente de OpenERP

Para levantar el cliente únicamente es necesario ubicarse en la carpeta en la cual se haya descomprimido el código fuente del servidor, que en este caso es: “/Documentos/Tesis/OpenERP/Desarrollo/openerp-client-6.0.1”. De la carpeta “bin” se ejecuta el archivo “openerp-client.py”.

```
$cd /home/usuario/openerp/openerp-client/bin  
$. /openerp-client.py
```

Ya que se ejecutaron ambos comandos, tanto el cliente como el servidor esta funcionando y en espera de ser utilizados. Ahora basta con abrir algún navegador y teclear la siguiente URL:

<http://localhost:8080>

Y de inmediato se debe mostrar en pantalla la página inicial de OpenERP, que debe ser la que se muestra en la Figura 4 -3.

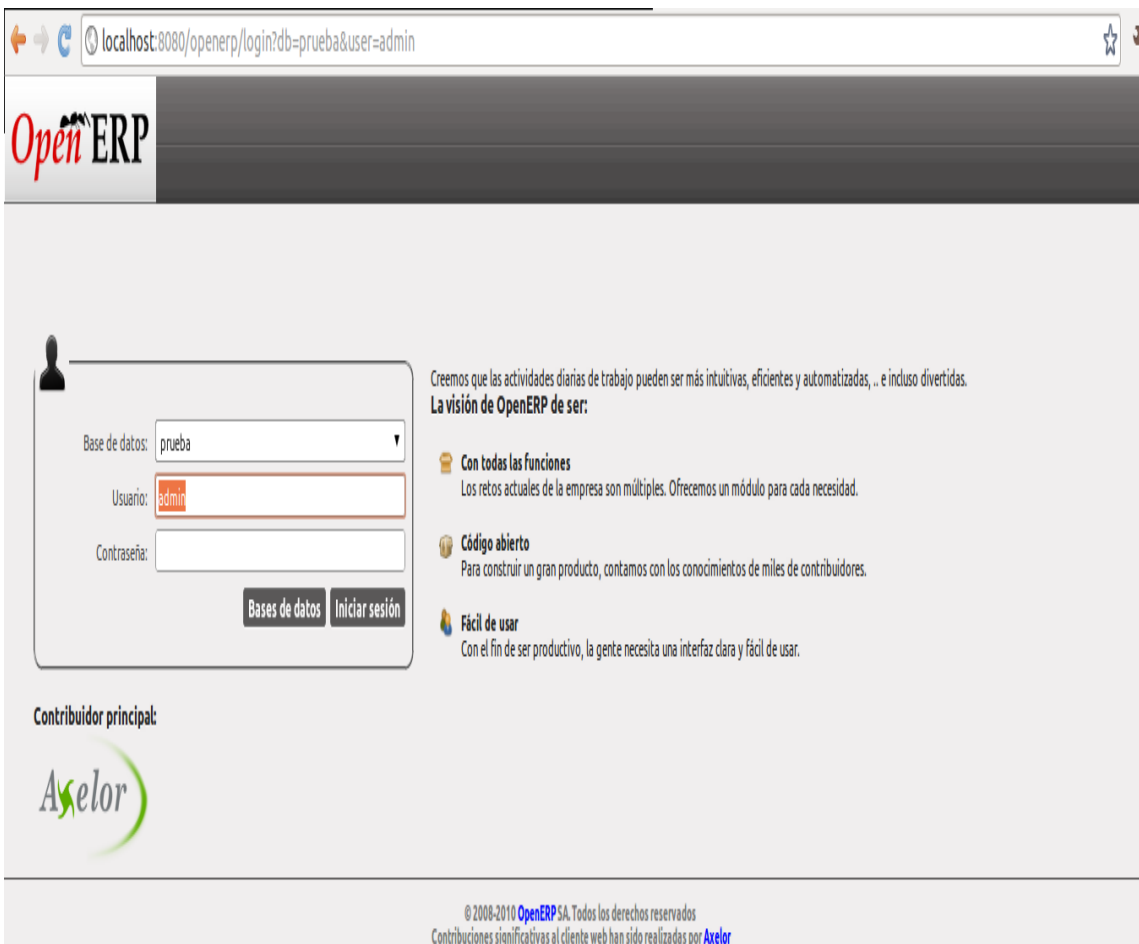


Figura 4.3 Pantalla de logeo para OpernERP

Hasta este punto hemos cumplido con la tarea 0.1 y sus derivados, ahora es necesario continuar con las tareas restantes para este ciclo.

Tarea 0.2: Configurar Ambiente de Desarrollo

Modelo Vista Controlador

OpenERP está implementado haciendo uso del Modelo Vista Controlador o MVC¹⁹, como su nombre lo indica es un modelo que divide en 3 componentes principales la estructura que conforma OpenERP, estos son el modelo, la vista y el controlador.

Todo modulo dentro de OpenERP está conformado principalmente por 3 elementos básicos:

- ³⁵₁₇ El **modelo**: un conjunto de relaciones (tablas) dentro de la base de datos.
- ³⁵₁₇ La **vista**: uno o más archivos XML que definen la interfaz de usuario.
- ³⁵₁₇ El **controlador**: son todas aquellas clases creadas en python y que en conjunto conforman la lógica que determina el funcionamiento de cada modulo.

Para crear un nuevo módulo, basta con generar una carpeta con el nombre del módulo que deseemos crear y colocarla dentro de la siguiente ruta:

```
/home/usuario/openerp/openerp-server/bin/addons
```

Y dentro de esta carpeta debe haber 4 archivos básicos:

- ³⁵₁₇ **__init__.py**: en este archivo se pueden incluir métodos y atributos del paquete.
- ³⁵₁₇ **__terp__.py**: este archivo contiene la descripción del módulo
- ³⁵₁₇ **coche.py**: aquí se define la lógica del módulo y la relación con el modelo.
- ³⁵₁₇ **coche_view.xml**: la interfaz de usuario, es definida mediante este archivo.

¹⁹ Es un patrón de programación de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos. La entre estos componentes se lleva a cabo mediante el controlador, es decir, la vista se comunica con el controlador y a su vez este se comunica con el modelo y viceversa, sin embargo la vista y el modelo nunca puede interactuar directamente [18].

Tomando en cuenta lo anterior ahora parece mucho más sencillo comenzar a desarrollar con no más que el block de notas el módulo de facturación que necesitamos, sin embargo las cosas aún pueden ser mucho mas sencillas, como se explica a continuación.

Interacción Eclipse-OpenERP

Para fines prácticos en esta sección únicamente se describe la manera en la cual se debe instalar Eclipse y cómo configurarlo dentro de Ubuntu 10.10 para poder generar código de manera aún más sencilla. Para ello es necesario agregar los complementos necesarios para que *Eclipse* y *Python* puedan interactuar de manera adecuada.

Lo primero que necesitamos es asegurarnos que nuestra distribución de Ubuntu tenga instalada la última versión del jdk²⁰ de Java y en caso de no contar con ella, basta con teclear desde una consola:

```
sudo apt-get install sun-java6-jdk
```

También podemos utilizar *synaptic*²¹ para dicha instalación, una vez que estemos seguros de contar con el jdk de java, procedemos a descargar la versión clásica de Eclipse, la cual podemos encontrar en el sitio web <http://eclipse.org/downloads/>²².

Ahora lo primero que hay que hacer es instalar el plugin *PyDev* para poder desarrollar en Eclipse usando Python y esto lo hacemos seleccionando desde el la barra de navegación de Eclipse *Help->Install New Software*. Tal como se muestra en la Figura 4.3

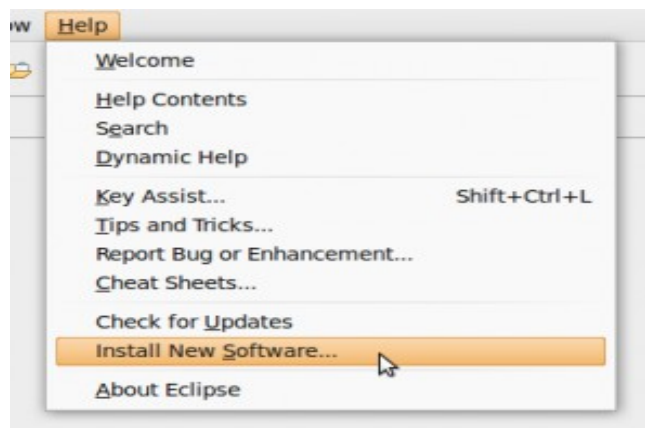


Figura 4.3 Pestaña Help Eclipse

En la ventana que se abre dentro del campo *Work With* introducimos la dirección en la cual se encuentra este plugin <http://pydev.org/updates> y presionamos el botón *Add*.

En la nueva venta que se abre tecleamos *PyDev* en el campo *Name:* y presionamos *Ok* para continuar con el proceso.

²⁰ **Java Development Kit** de sus siglas en ingles, es el conjunto de herramientas necesarias para poder desarrollar aplicaciones en lenguaje Java.

²¹ Gestor de actualizaciones y descargas de software para distribuciones de código libre en particular para Ubuntu [16].

²² Se descomprime dicha carpeta y se localizar el ejecutable de Eclipse e iniciar el editor, no hace falta instalación.

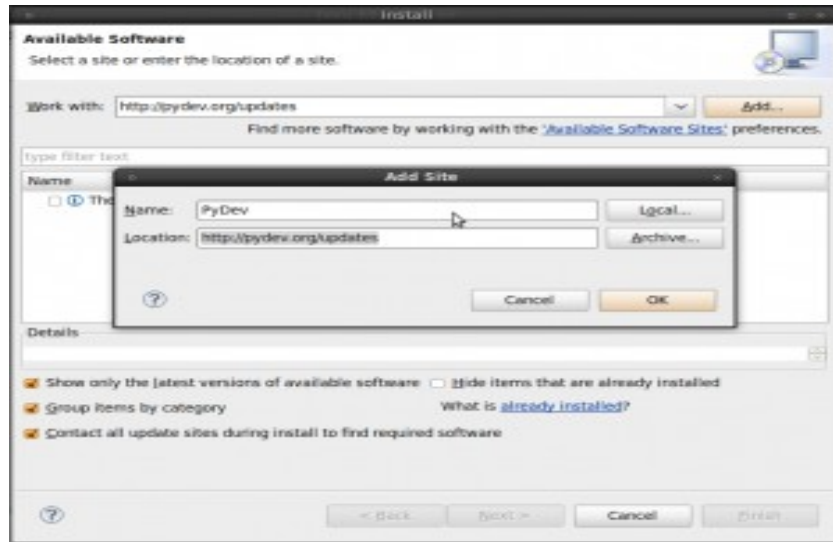


Figura 4.4 Ventana de Selección de Servidores

Se mostrará una lista con varios checkbox seleccionamos *PyDev* y presionamos *Next*.

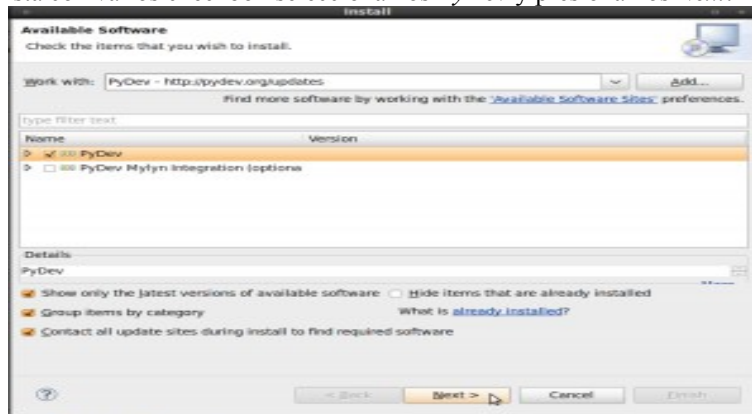


Figura 4.5 Lista de Selección de Bibliotecas

Después seleccionamos *PyDev for Eclipse* y presionamos *Next* para indicar que ya elegimos el complemento.

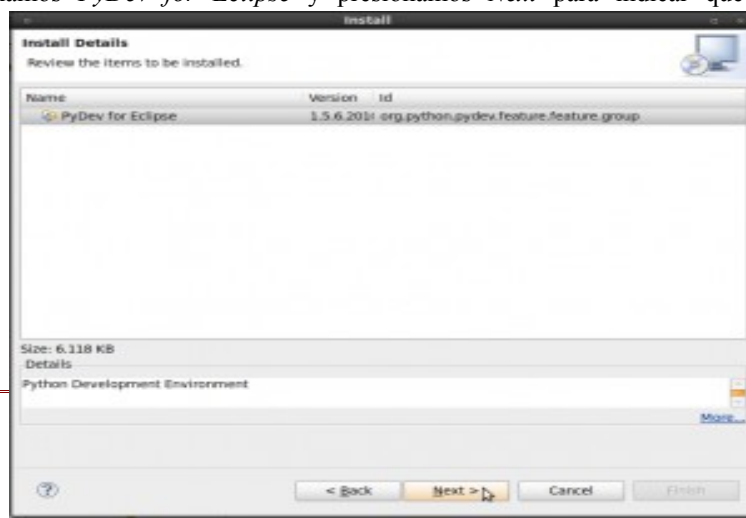


Figura 4.6 Ventana de Confirmación de Software Seleccionado

Aceptamos la licencia y presionamos *Finish*, durante la instalación se muestra una advertencia, sin embargo no es algo de importancia así que podemos presionar *Ok*, de tal manera que al final Eclipse nos preguntará si deseamos reiniciar la aplicación para completar la instalación, seleccionamos *yes*.

Ahora el paso restante es configurar *Eclipse* para poder trabajar, lo primero que hay que hacer después de que *Eclipse* haya reiniciado es configurar *python* para poder ocuparlo como lenguaje de programación. Buscamos el menú *Window* y seleccionamos la opción *Preferences*, en la nueva ventana que se despliega expandamos el menú *PyDev* que está en el panel del lado izquierdo, seleccionamos *Interpreter-Python*, como se muestra en la Figura 4.7.

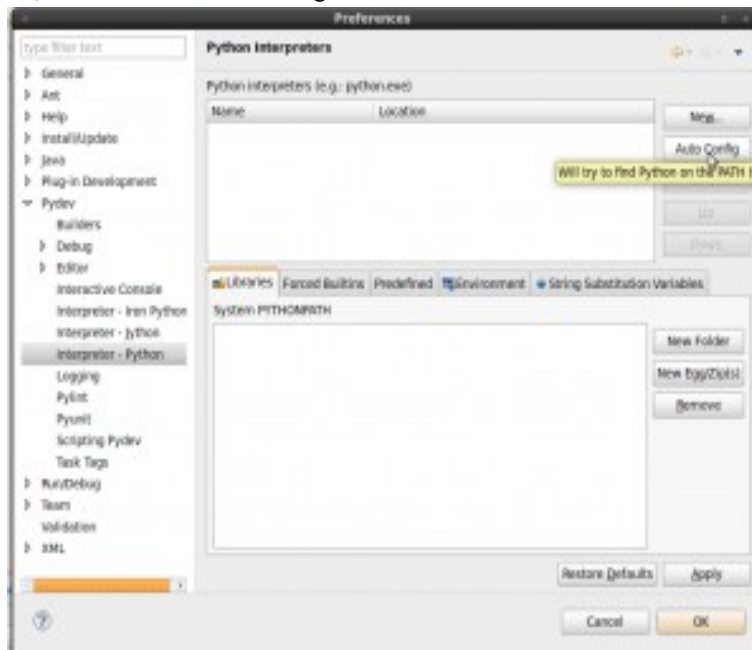


Figura 4.7 Ventana Preferences de Eclipse

Presionamos el botón *Auto Config* y después de haber realizado este paso se despliega una nueva ventana en la que todos los recuadros de selección estarán marcados menos el primero de ellos, presionamos *OK* para concluir con el proceso. Y por último presionamos *OK* en la ventana de preferencias y con esto ya queda configurado Eclipse para poder utilizar Python.

Extensión para editar documentos XML.

De igual manera que el modelo MVC, ahora es necesario agregar las herramientas para poder editar la vista del sistema, es decir los documentos XML. Repetimos los primeros paso anteriores, *Help -> Install New Software* pero ahora damos click en la lista expandible del campo *Work With* y seleccionamos "*Helios – <http://download.eclipse.org/releases/helios>*", una vez que ya se cargó la lista de opciones marcamos el checkbox correspondiente a *Eclipse XML Editors and Tools* y presionamos *Next*.

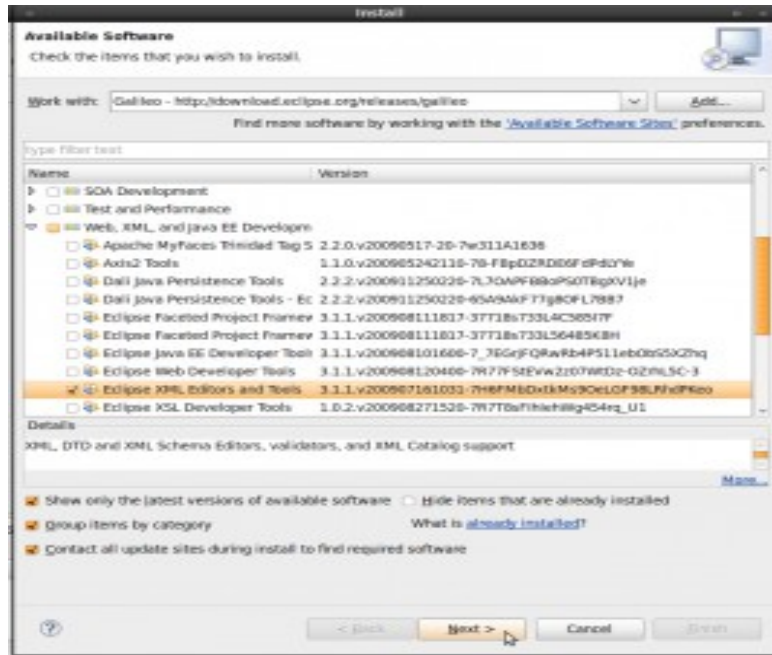


Figura 4.8 Lista de Software Disponible para Instalar

Después es necesario expandir la opción de *Eclipse XML Editors and Tools* y seleccionamos *WST XML Core* y presionamos *Next*.

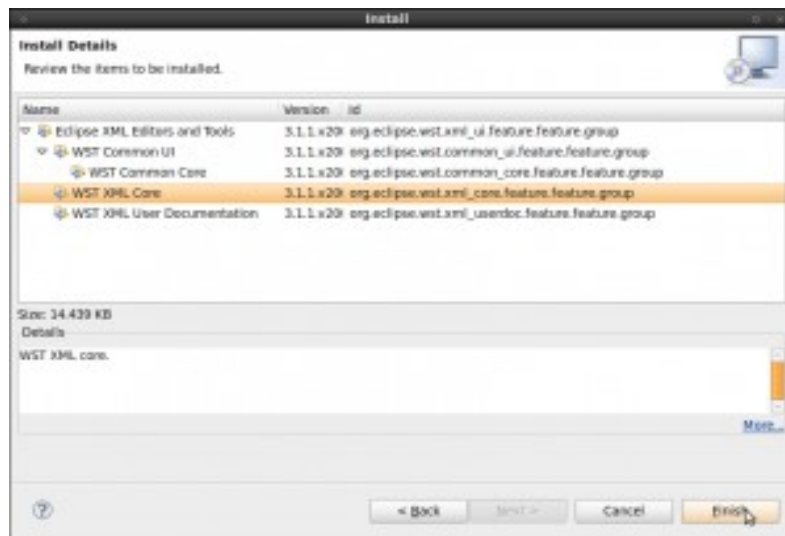


Figura 4.9 Lista de Software Seleccionado para Instalar

Aceptamos la licencia y presionamos *OK*, Eclipse preguntará una vez más si es necesario reiniciar, aceptamos y ahora ya se tienen instaladas las herramientas para editar documentos XML, por último aunque no es necesario, pero ayuda mucho, se pueden agregar 2 templates²³ o plantillas tanto para el controlador como para la vista.

Template Python y XML

La última parte de esta sección se enfocará a explicar cómo agregar dos *templates* tanto para generar código Python como para generar documentos XML. Es necesario descargar ambos templates:

³⁵₁₇ Python: <http://openerp-eclipse-template.googlecode.com/svn/trunk/templates-openerp.xml>

²³ Mejor conocido como plantilla, es un documento que facilita el desarrollo de código, ya que sirve como base para completar dicho documento y generar el código necesario.

35
17 XML:http://openerp-eclipse-template.googlecode.com/svn/trunk/Openerp-eclipse-xml-template.xml

Ambos templates hay que copiarlos en un archivo de texto plano y guardarlos con cualquier nombre, pero con la extensión .xml. Posteriormente vamos al menú *Window -> Preferences* dentro del menú *PyDev* buscamos la opción *Templates* y Después de haber seleccionado esta opción, en el panel del lado derecho de la pantalla, presionamos el botón *Import* y buscamos la ruta del archivo en el cual guardamos el template para Python.

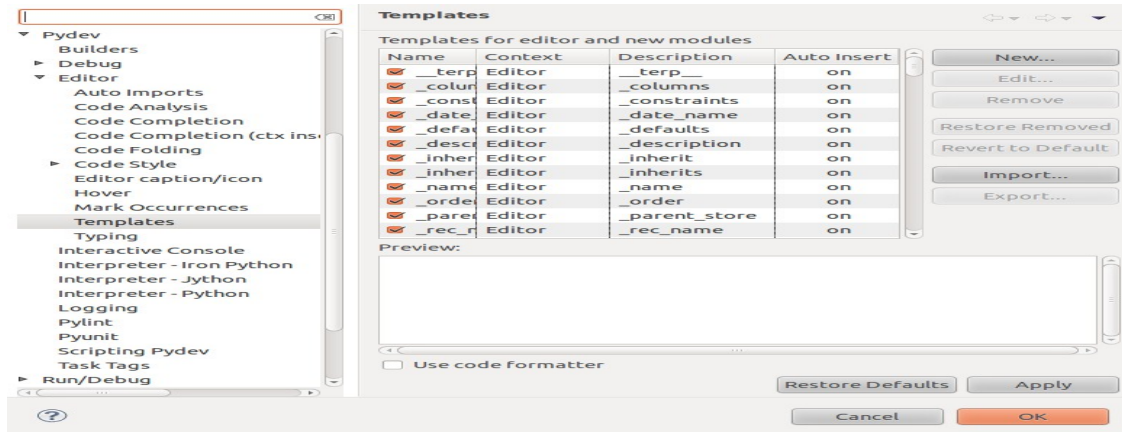


Figura 4.10 Templates Eclipse

Y el mismo procedimiento se repite pero ahora en el menú XML, como se muestra en la Figura 4.11.

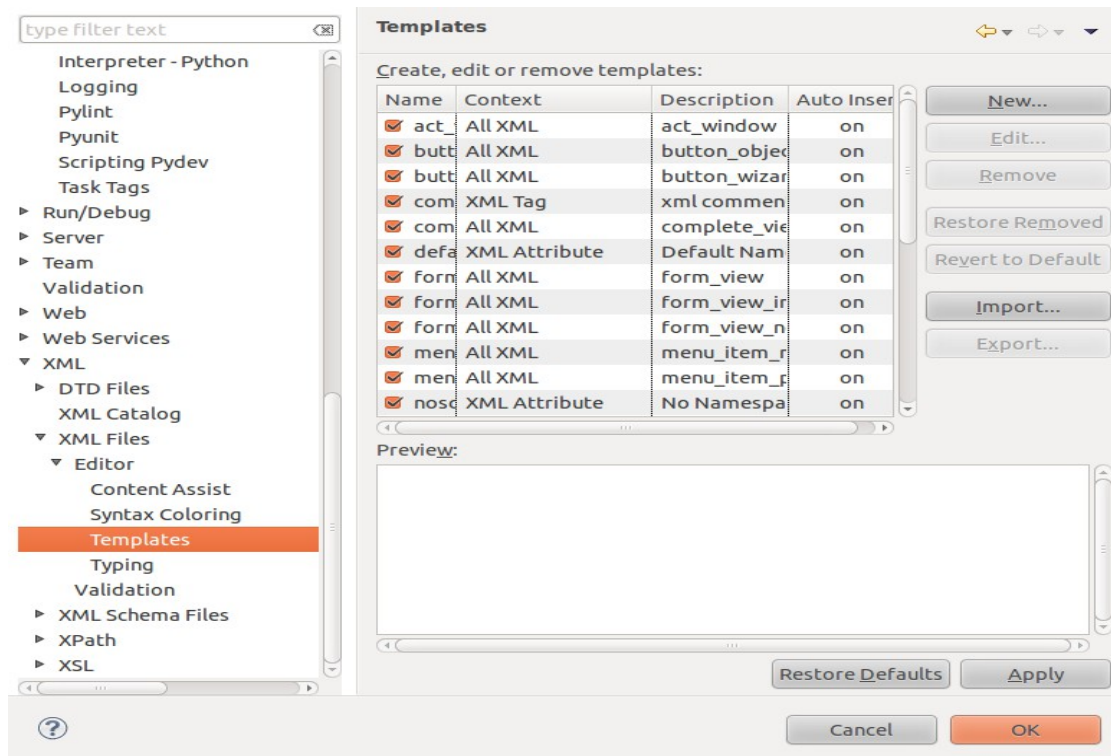


Figura 4.11 Template Seleccionado en Eclipse

Con estos últimos cambios ya quedan agregadas las plantillas para el desarrollo de módulos para OpenERP. Así concluyen las tareas del ciclo 0.

Retrospectiva del ciclo

En este primer ciclo fue evidente que es necesario distribuir de manera eficiente el tiempo destinado a cada tarea y que es más importante aun apegarse a dicha planeación ya que al llevar a cabo esto de manera adecuada estamos cumpliendo con los estándares que señala SCRUM. Además por ser el primer ciclo se dieron circunstancias que no estaban previstas, como el agregar tareas al *Product BackLog* o agregar secciones que no corresponden propiamente a una tarea dentro del ciclo pero que son importantes para los usuarios del sistema, como es el caso de automatizar los servicios mediante *Script's*. Estas lecciones son importantes ya que son la esencia de las metodologías ágiles en las que se considera que el software es un ente en constante cambio.

Por otro lado puede decirse que al término de este ciclo no queda pendiente ninguna tarea relacionada con la instalación de OpenERP ni con la instalación y configuración de Eclipse y por lo tanto podemos proceder a revisar algunas de las funciones básicas de OpenERP, antes de pasar de lleno al desarrollo.

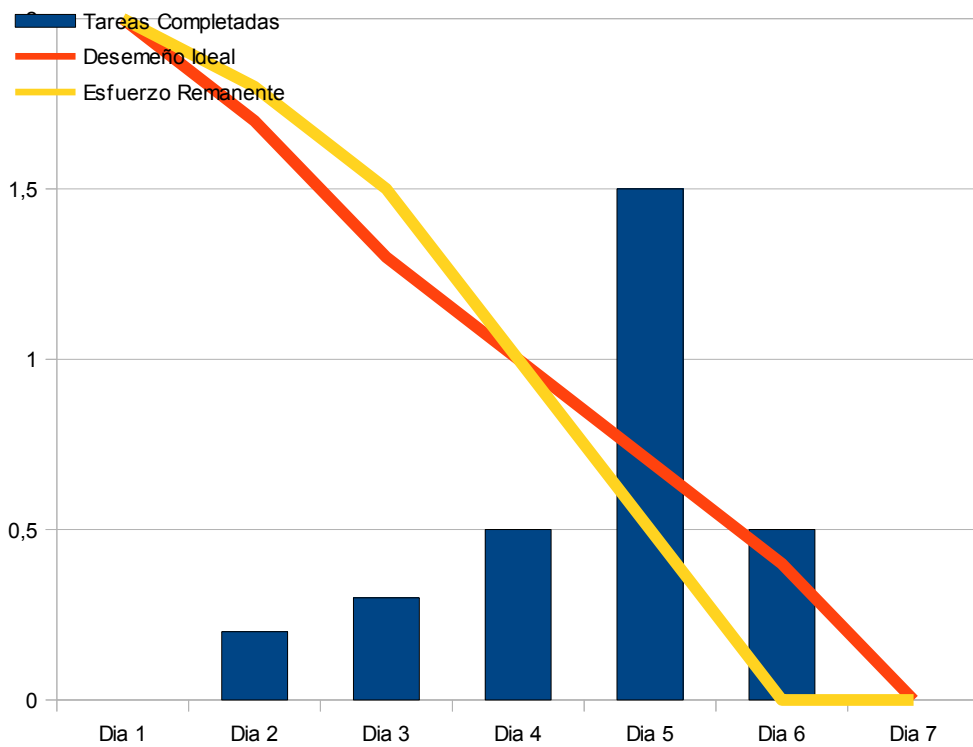
4.4 Ciclo 1: Administrar Base de Datos

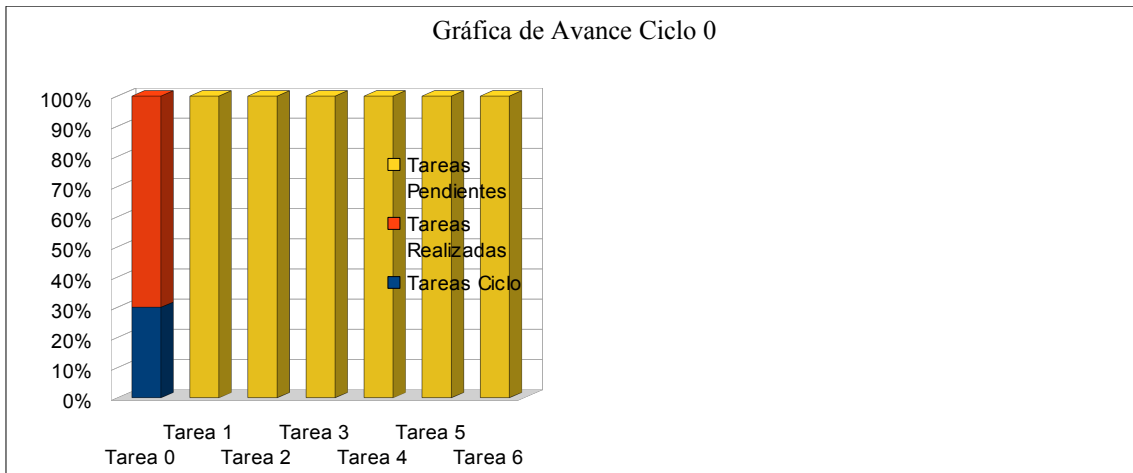
Este ciclo está destinado a crear la base de datos y mostrar como el administrador se encarga de gestionar las tareas que tienen que ver directamente con la base de datos, tales como crearla, crear usuarios y generar los respaldos de la base de datos.

Sprint Backlog: SiFaDig			
Ciclo: 1	<p>Objetivos Generales: El objetivo general de este ciclo es comenzar con la configuración inicial de OpenERP, el ciclo pasado sirvió para instalar paquetes, bibliotecas y crear el entorno necesario para el desarrollo del sistema. Pero es objetivo de este ciclo crear la base de datos necesaria para OpenERP, así como mostrar como ingresar al sistema después de haber generado al usuario administrador. Dado que a partir de este ciclo comenzaré a mostrar funciones del sistema, este ciclo iniciará con el caso de uso correspondiente a estas tareas.</p> <p>Puntos Importantes:</p> <ul style="list-style-type: none"> ³⁵/₁₇ Generar Base de Datos y Administrador. ³⁵/₁₇ Ingresar al Sistema. 		
Fecha de Inicio: 21/03/2011			
Fecha de Terminó: 27/03/2011			
Tareas			
Id	Descripción de la Tarea	Tamaño	Prioridad
0.3	Generar B.D. y Administrador	1	1

1	Ingresar al Sistema	1	1
---	---------------------	---	---

Gráfica de Quemado Ciclo 0





Descripción del caso de uso

Después de haber instalado OpenERP, es necesario generar el usuario administrador y la base de datos en la cual OpenERP almacena la información necesaria para poder funcionar. Para poder crear la base de datos es necesario desde la pantalla inicial presionar el botón correspondiente a la opción *Bases de Datos*, después proporcionar los datos correspondientes y presionar *Crear*.

Caso de Uso: Nueva Base de Datos	Identificador: 0.1
<pre> graph LR Admin[Administrador] --> U0((0.- Administrar Bases de Datos)) U0 --> U01((0.1.- Nueva Bases de Datos)) </pre>	
Descripción:	Después de haber instalado OpenERP, es necesario generar el usuario administrador y la base de datos en la cual OpenERP almacena la información necesaria para poder funcionar. Para poder crear la base de datos es necesario desde la pantalla inicial presionar el botón correspondiente a la opción <i>Bases de Datos</i> , después proporcionar los datos correspondientes y seleccionamos <i>Crear</i> .
Nivel:	El nivel de este caso de uso es básico, ya que sin haber creado la base de datos no se puede ingresar al sistema.
Actor Principal:	Administrador.
Parte Interesada:	Usuarios del sistema.
Iniciador del Caso de uso:	Este caso de uso se inicia justo después de haber instalado OpenERP y haber seleccionado desde el navegador y en la página inicial el botón <i>Bases de Datos</i> y después <i>Crear</i> .
Pre-condiciones:	La única precondición es haber instalado de manera correcta el OpenERP y haber ejecutado de manera correcta tanto el servidor como el cliente web de OpenERP.

Flujo de Desempeño Óptimo:		
Paso	Entrada	Respuesta del Sistema
1	El usuario selecciona el botón Bases de Datos .	Se muestra una lista de de opciones en la parte superior derecha del navegador , con las opciones correspondientes a la administración de las bases de datos.
2	El usuario selecciona la opción Crear .	Se muestra al usuario los datos que debe proporcionar para crear la base de datos como: ³⁵ ₁₇ Puerto de conexión. ³⁵ ₁₇ Contraseña del Administrador de las bases de datos. ³⁵ ₁₇ Nombre de la Base de Datos. ³⁵ ₁₇ Se pide indicar si es que se quiere datos de prueba. ³⁵ ₁₇ Lenguaje por defecto ³⁵ ₁₇ Asignar y confirmar contraseña del administrador que utilizara esta nueva base de datos.
3	El usuario ingresa de manera correcta y sin errores los datos solicitados y presiona el botón de Aceptar .	Se muestra una pequeña pantalla con la leyenda “Operación en Progreso....”. Una vez creada la base de datos se muestra una nueva ventana con el resumen de la información referentes a la nueva base de datos
4	El usuario presiona el botón en el que se lee Omitir Asistentes de Configuración .	El sistema muestra la pantalla principal de OpenERP.
Flujo Alternativo:		
Paso	Entrada	Respuesta del Sistema
4.1	El usuario presiona Empezar Configuración .	El sistema pide selecciona que tipo de interfaz desea utilizar el usuario y aconseja utilizar la interfaz simplificada.
4.2	Una vez seleccionada la interfaz, el usuario presiona Siguiente .	En una nueva pantalla se pide proporcionar los datos de la empresa, como: ³⁵ ₁₇ Nombre. ³⁵ ₁₇ Dirección. ³⁵ ₁₇ Correo Electrónico. ³⁵ ₁₇ Sitio Web. ³⁵ ₁₇ Moneda. ³⁵ ₁₇ Logo.
4.3	El usuario captura los datos correspondientes a su empresa y presiona Siguiente .	En la siguiente ventana el sistema pide seleccionar los módulos que serán incluidos en la nueva instancia. Se muestra una serie de recuadros de seleccion, con las opciones que se pueden seleccionar y se indica que de no estar seguro, estos módulos pueden ser agregados posteriormente, presionando el botón de Saltar .
4.4	Una vez seleccionados estos módulos el usuario presiona el botón Siguiente .	El sistema muestra la pantalla principal de OpenERP.
Post-condiciones:		Debe mostrarse en pantalla la leyenda, “La Base de Datos ha sido

	creada de manera Exitosa” y posteriormente será posible ingresar al sistema con el usuario y contraseña que se haya asignado al momento de crear la base de datos
Excepciones no controladas:	<p>³⁵₁₇ No se puede crear la base debido a que no hay conexión con el servidor de la base de datos.</p> <p>³⁵₁₇ El servidor de OpenERP se ejecuto con un usuario diferente al usuario “openerp”.</p> <p>³⁵₁₇ Se utilizan caracteres no permitidos para el nombre de la base de datos.</p> <p>³⁵₁₇ No hay espacio suficiente en memoria para poder crear la nueva base de datos.</p>
Consecuencias de Fallo:	³⁵ ₁₇ De existir un fallo al momento de crear la base de datos, esta no se genera y por lo tanto no se puede ingresar al sistema.
Garantías Minina:	<p>³⁵₁₇ Se garantiza que de ocurrir algún problema al momento de generar la base de datos, esto no afectara el desempeño del sistema por lo que basta con repetir el procedimiento para poder concluirlo.</p> <p>³⁵₁₇ De igual manera se garantiza que al momento de crear una nueva base de datos, no serán eliminadas ni modificadas las bases de datos existentes en el servidor en el que haya sido instalado el OpenERP.</p>
Frecuencia:	Este caso de uso deberá ser invocado cada vez que se desee crear una nueva instancia de la base de datos. En particular cuando se instale por primera vez el OpenERP.
Supuestos:	Se asume que el usuario domina alguno de los lenguajes en los que puede ser instalado el sistema.

Comentarios:

Desempeño:

³⁵₁₇ El puerto preseleccionado al momento de generar una nueva base de datos, es el puerto 8087, pero puede modificarse, aunque no se recomienda.

³⁵₁₇ Cuando se genera una nueva base de datos la contraseña por default para el administrador de las bases de datos es “admin”, pero puede ser modificada.

Interfaz de Usuario:

³⁵₁₇ El idioma del sistema puede ser modificado para que sea mas amigable con el usuario.

³⁵₁₇ Las contraseñas al momento de ser tecleadas no se muestran por seguridad del usuario.

Casos Siguientes:

³⁵₁₇ Una vez creada la base de datos, el usuario puede ingresar al sistema y hacer uso de cada uno de los módulos que haya instalado al momento de la configuración.

³⁵₁₇ El usuario puede eliminar, restaurar o respaldar la base de datos recién creada.

³⁵₁₇ En caso de no necesitar de ninguna otra funcionalidad, el usuario puede cerrar su sesión y salir del sistema.

Tarea 0.3: Crear Base de Datos y Administrador en OpenERP

Una vez que se logró levantar tanto el cliente como el servidor de OpenERP, tal como se mostró en el ciclo 0, ahora es necesario ingresar a la pagina inicial de OpenERP, esto lo hacemos tecleando en el navegador la siguiente URL: <http://localhost:8080>, de inmediato el navegador mostrará la siguiente pantalla.



Figura 4.12 Pantalla de Ingreso OpenERP

Una vez ubicados en esta pantalla, lo primero que hay que hacer es seleccionar la opción *Bases de Datos*. Y se mostrará la siguiente pantalla.

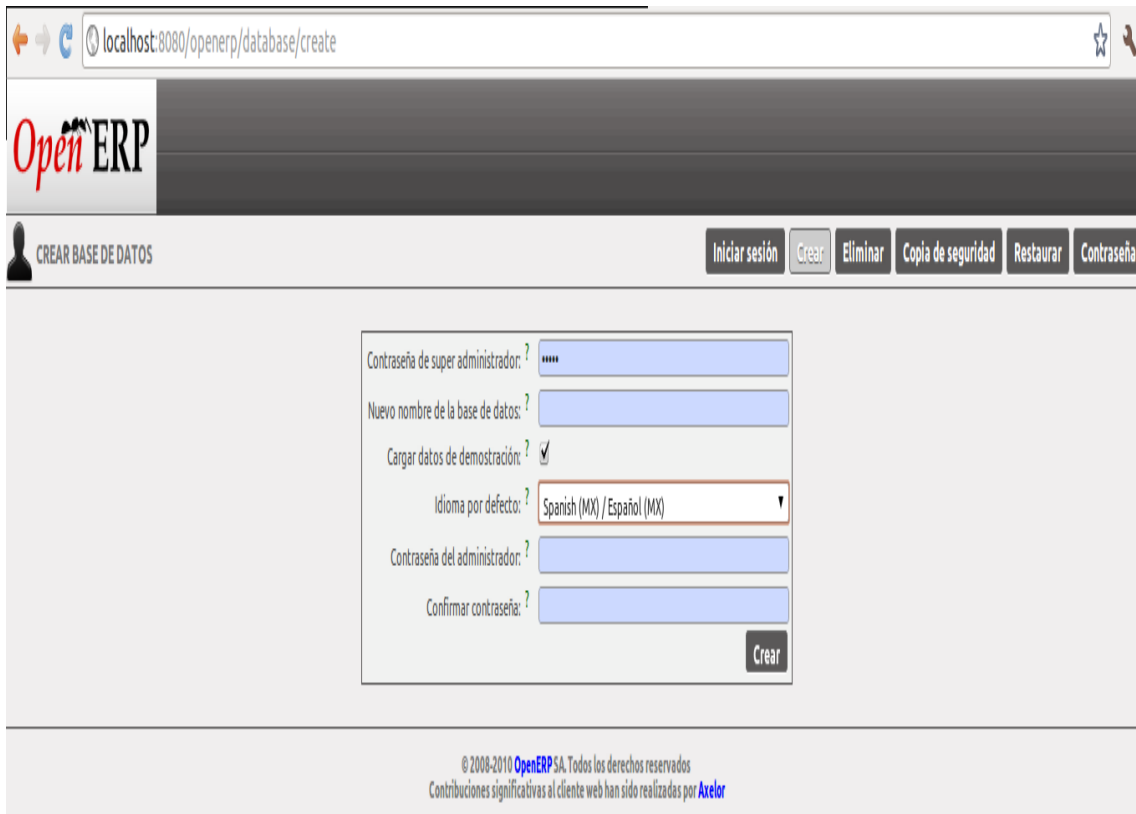
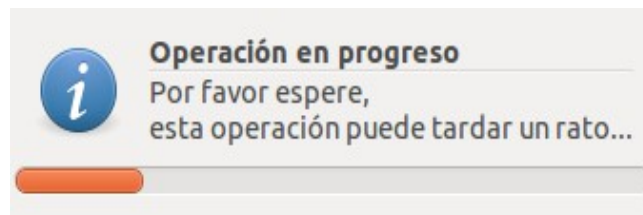


Figura 4.13 Pantalla Crear Base de Datos

En esta pantalla es necesario capturar los datos que se piden, se puede crear la base vacía o cargar datos por defecto y seleccionar el lenguaje que se utilizara en la base de datos. Por último es necesario presionar el botón *Crear*.

Vale la pena recalcar que si algún dato no es capturado el sistema no permite crear la base de datos. Una vez que se presiona el botón *Crear* se muestra una pequeña pantalla indicando que se esta creando la base de datos.



Después se despliega una ventana en la que se muestra el resumen de la base que ha sido creada y se pide elegir una de 2 opciones:

³⁵₁₇ **Omitir asistentes de configuración:** Si se elige esta opción de inmediato se muestra la pantalla principal de OpenERP correspondiente a la sesión del administrador y con los módulos básicos como ventas y administración.

³⁵₁₇ **Empezar configuración:** Al seleccionar esta opción se permite agregar capturar los datos de nuestra empresa, así como agregar módulos adicionales que permiten mayor funcionalidad del sistema.

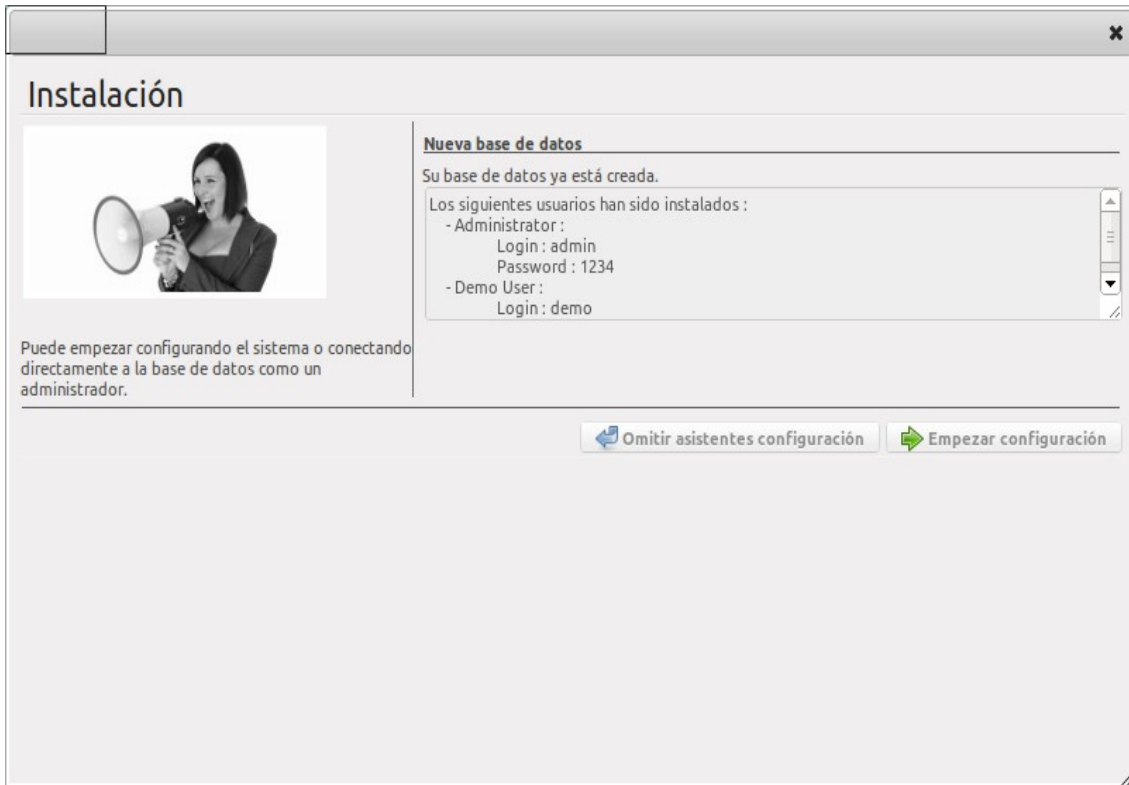


Figura 4.14 Pantalla de confirmación para crear base de datos

En caso de haber elegido la primera opción, el sistema muestra la siguiente pantalla, en la que el usuario puede comenzar a hacer uso del sistema. Esta es la pantalla principal del sistema y muestra las opciones disponibles.

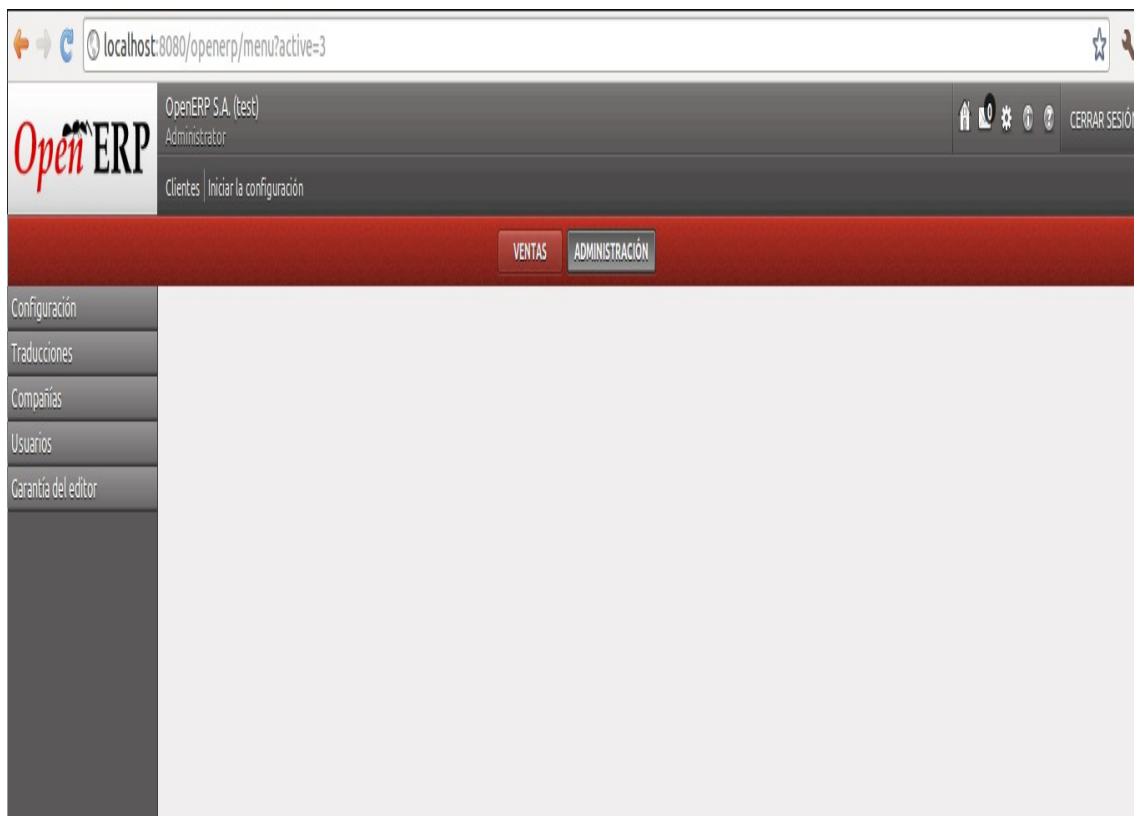


Figura 4.15 Panel Principal OpenERP

En caso de haber elegido la segunda opción se mostrará la siguiente pantalla, en la que se pide al usuario configurar la interfaz que utilizará, se sugiere hacer uso de la *Interfaz Simplificada*.



Figura 4.16 Ventana Configuración de Interfaz

Una vez seleccionada el tipo de interfaz se presiona *Siguiete* y en la siguiente pantalla es necesario capturar los datos de la compañía, así como su logo, no se permite continuar si los campos obligatorios (campos resaltados color obscuro) no son llenados. Cada uno de estos campos corresponden a los datos de la compañía que hará uso del sistema y estos datos serán empleados para los procesos del sistema, como por ejemplo al momento de generar una factura.

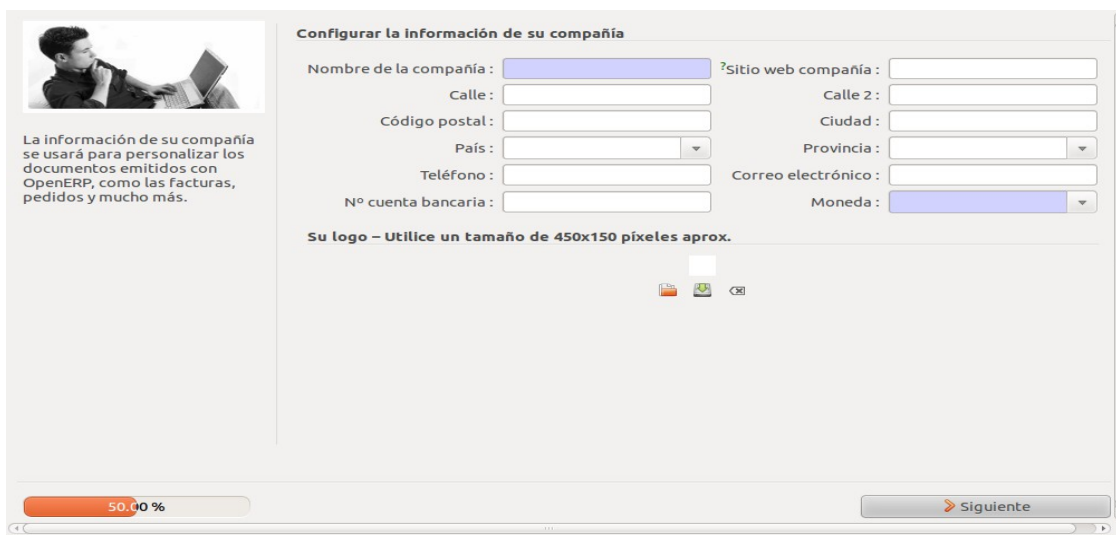


Figura 4.17 Configuración Datos de la Compañía

Ya con los datos necesarios en el formulario es necesario presionar *Siguiete* para avanzar a la siguiente ventana en la cual podemos seleccionar mediante recuadros de selección los módulos que podemos agregar a nuestra Instalación. Cada uno de éstos recuadros de selección corresponde a un modulo

destinado a realizar alguna función determinada en el sistema. Para mayor información de los módulos disponibles y de sus características se puede buscar en el sitio web <http://www.openerp.com/>.

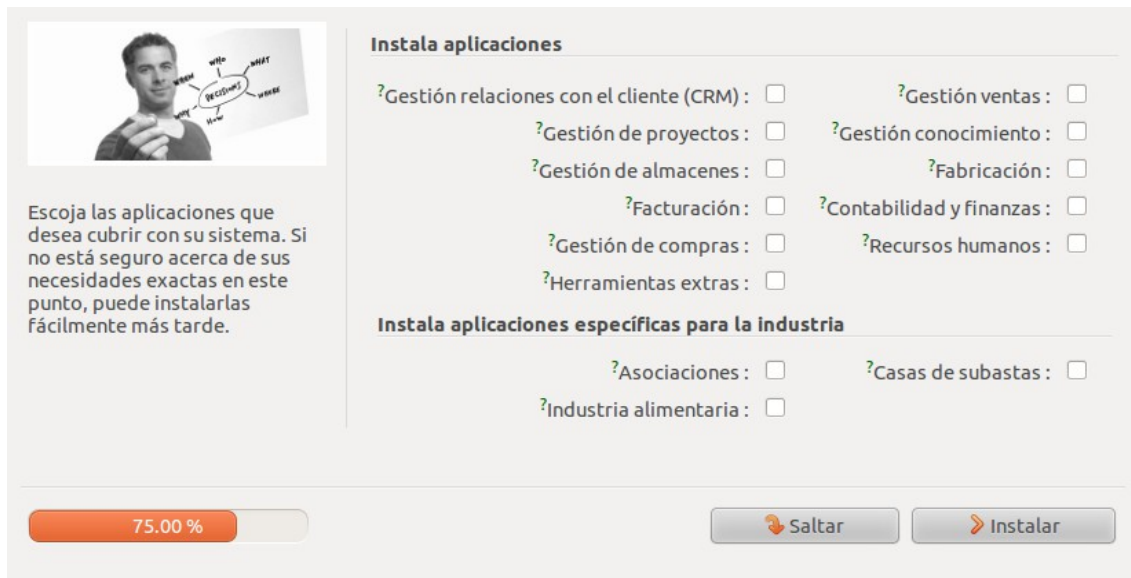


Figura 4.18

Seleccionamos los módulos que vayamos a utilizar y seleccionamos una de dos opciones:

- ³⁵₁₇ **Saltar:** Si elegimos esta opción no se instala modulo alguno y se manda al usuario a la ventana inicial.
- ³⁵₁₇ **Instalar:** Se instalan los módulos seleccionados (toma algunos minutos) y después de haber instalado estos módulos, se redirige al usuario a la ventana inicial de OpenERP.

En ambos casos al terminar la configuración, la sesión con la que se inicia es la sesión del administrador del sistema.

Ingresar al sistema como administrador.

Por último únicamente resta ingresar al sistema como administrador, para realizar esto únicamente se tiene que teclear la siguiente dirección en el navegador, <http://localhost:8080> y con esto se muestra la pagina principal de OpenERP ver Figura 4.19 la cual muestra la pantalla de ingreso al sistema en la que se puede seleccionar la base de datos y es necesario ingresar con el usuario y contraseña correcta para poder tener acceso al sistema



Figura 4.19 Pantalla de Ingreso OpenERP

Una vez ubicados en esta parte del sistema es necesario ingresar el usuario y contraseña correspondiente al administrador y con esto ingresamos al sistema, con privilegios de administración. De haber ingresado de manera correcta al sistema se mostrara la siguiente pantalla y en la esquina superior izquierda se puede ver que el usuario registrado es el *Administrador*.

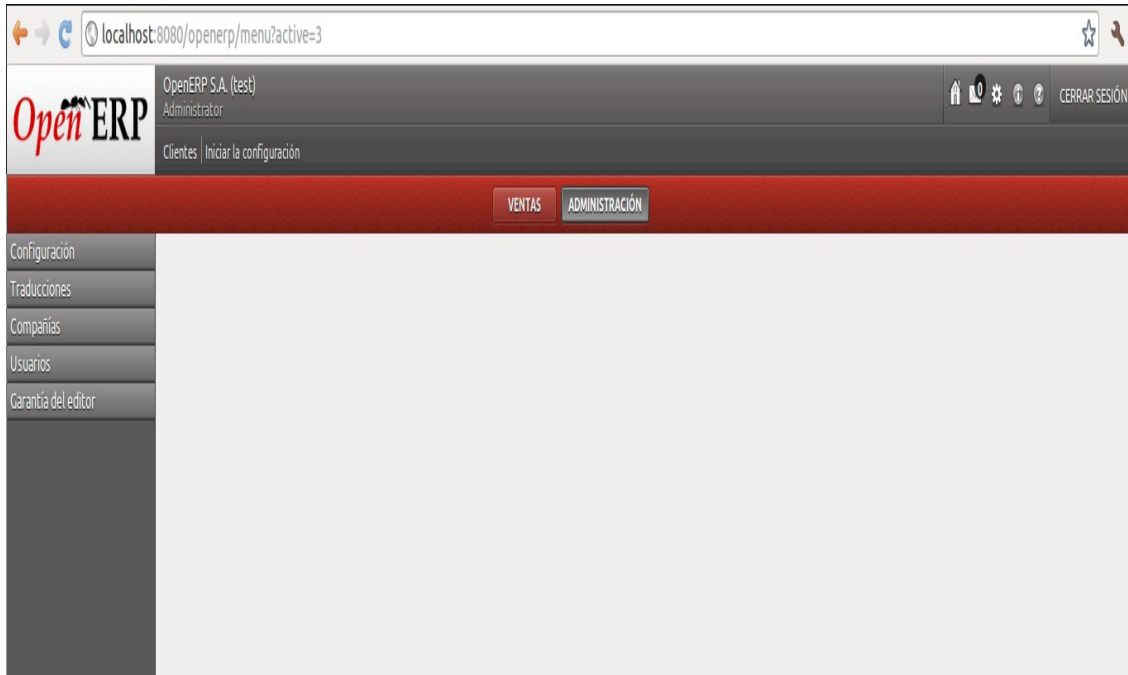


Figura 4.20 Panel Principal OpenERP

Y con estos sencillos pasos ya tenemos generado el ambiente para poder administrar nuestro sistema y posteriormente generar el desarrollo correspondiente.

Retrospectiva del ciclo:

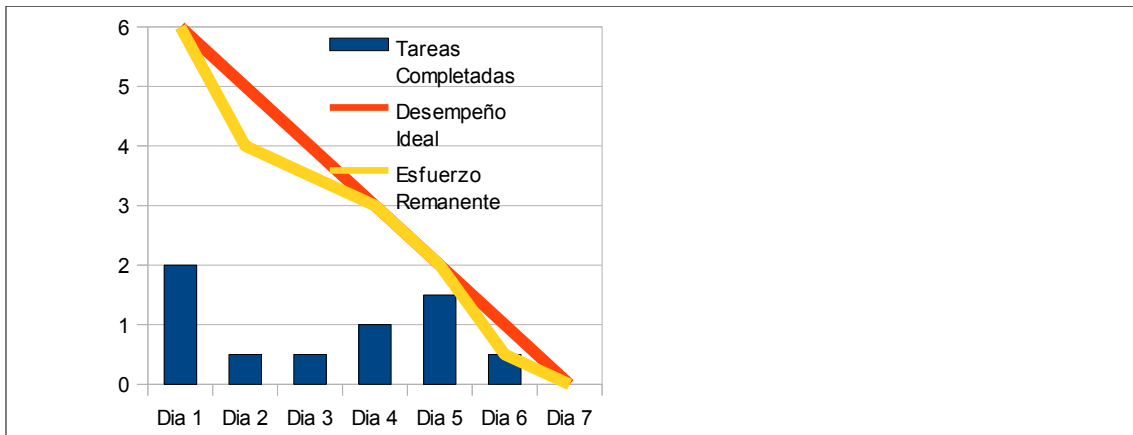
Como conclusión de este ciclo podemos observar que OpenERP es un sistema complejo y su curva de aprendizaje es amplia, así que es necesario dedicarle varios ciclos a los múltiples procesos que maneja a fin de poder entender su lógica y sus procedimientos, esto se puede justificar al leer la documentación de OpenERP así como al momento de hacer uso del mismo y ver la gran cantidad de funcionalidades que este ofrece al usuario. Además se puede asegurar que al momento del cierre de este ciclo, ya es posible realizar pruebas con la totalidad de los módulos precargados en OpenERP, ya que cada uno de ellos se encuentra disponible en el panel principal y son accesibles al usuario.

Capítulo 5. Implementación de las funciones del módulo SiFaDig

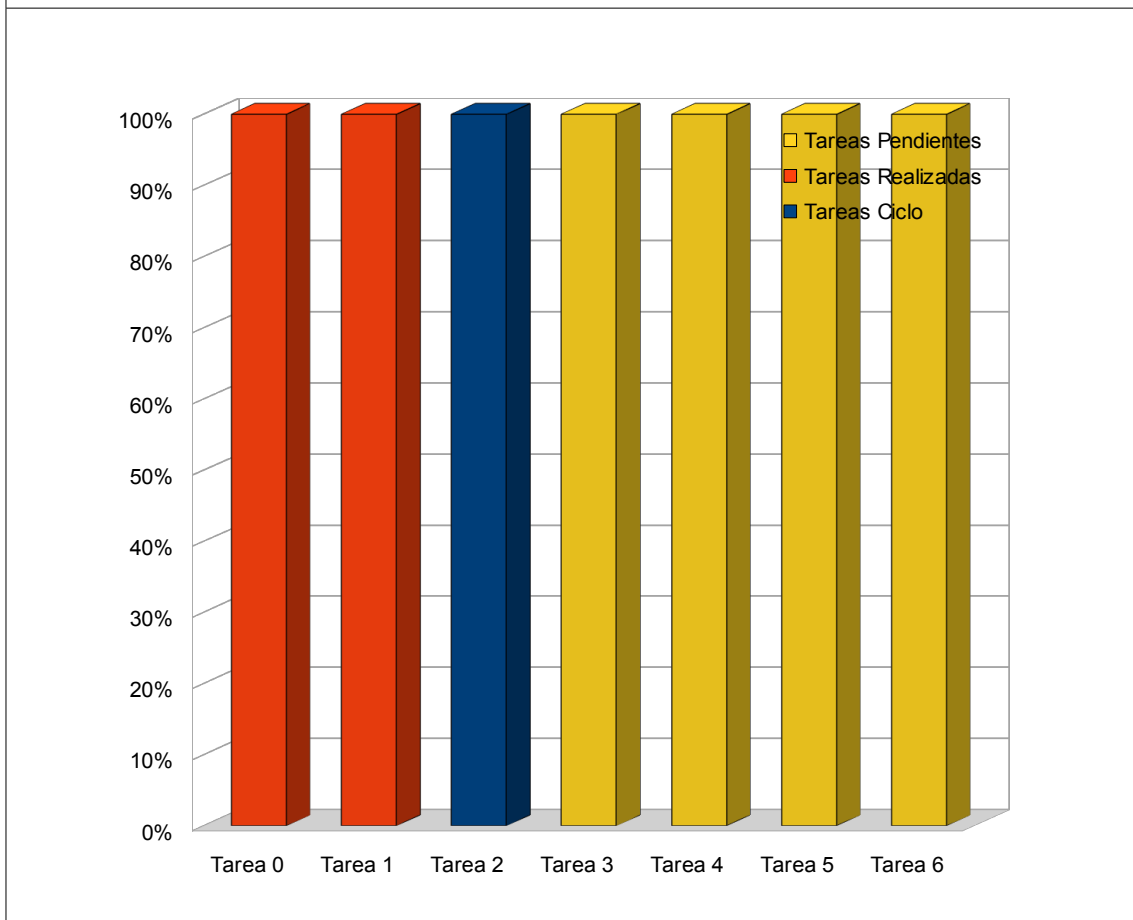
A partir de este capítulo se comenzará a dar forma a los elementos que integran el módulo de facturación electrónica. Tales como son los usuarios, los productos, las ventas etc. y en este ciclo en particular se mostrará cómo se dan de alta los usuarios en el sistema y como es que funciona el sistema de permisos de OpenERP. Es necesario mostrar cómo es que funcionan cada uno de los elementos que conforman SiFaDig de tal manera que se pueda ingresar al sistema con un usuario adecuado y que este pueda realizar los procesos necesarios para generar una factura electrónica y para poder llevar a cabo todos éstos procesos es necesario interactuar con todos y cada uno de los módulos que conforman el sistema mismos que serán analizados posteriormente.

Sprint Backlog: SiFaDig			
Ciclo: 2	<p>Objetivos Generales: Como objetivo principal de este ciclo, se tiene el crear usuarios que sean capaces de ingresar al sistema para poder hacer uso del mismo. Las reglas de OpenERP nos dictan que al momento de crear un usuario es requisito indispensable asignarle una serie de permisos que le conceden a este usuario el acceso a los diferentes módulos que componen OpenERP, además de poder asignar el acceso a cada modulo, también es posible personalizar los privilegios²⁴ que tenga el permiso, como lo son los privilegios de lectura y escritura. También es posible crear grupos que comprenden una serie de permisos y a cada usuario se le puede asignar uno o mas grupos, garantizándole a este usuario los permisos y privilegios de cada uno de los grupos a los que pertenezca. En este ciclo se creara un usuario, un grupo y se mostrara como relacionar a un usuario con sus permisos.</p> <p>Tareas para este Ciclo:</p> <ul style="list-style-type: none"> ⤴ Alta de Usuarios. ⤴ Asignación de Grupo. ⤴ Creación de Grupo. ⤴ Asignación de Permisos. 		
Fecha de Inicio: 18/04/2011			
Fecha de Terminó: 24/04/2011			
Tareas			
Id	Descripción de la Tarea	Tamaño	Prioridad
2	Administración de Usuarios	2	4
2.1	Alta de Usuarios	1	4
2.1.1	Asignación de Grupo	1	4
2.1.2	Creación de Grupo	1	4
2.1.3	Asignación de Permisos	1	4
Gráfica de Quemado Ciclo 2			

²⁴ Privilegios: Además de los permisos para acceder a los módulos de OpenERP, cada usuario tiene diferentes privilegios dentro de cada modulo, como puede ser privilegio de leer, escribir, crear o eliminar.



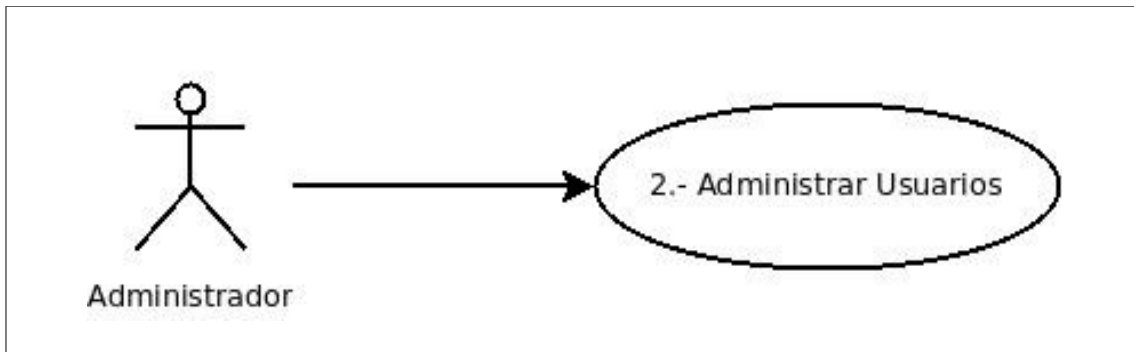
Gráfica de Avance Ciclo 2



Descripción del caso de uso

Una vez que ya fue instalado el sistema y que la base de datos ya fue creada, ahora es necesario crear y administrar los usuarios que harán uso sistema. Para poder llevar a cabo estas tareas, OpenERP provee una serie de mecanismos para garantizar el adecuado desempeño del sistema y de sus usuarios. Tales mecanismos están conformados por grupos, privilegios y permisos, que a lo largo de este ciclo serán utilizarlos.

Caso de Uso: Administrar Usuarios	Identificador: 2
--	-------------------------



Descripción:	Cualquier usuario con los privilegios adecuados puede crear usuarios, sin embargo en este caso en particular usaremos al “administrador” del sistema creado en el ciclo 0 para poder generar nuevos usuarios. Una vez que se haya ingresado en el sistema es necesario ubicar el botón para acceder a la administración del sistema, dentro del modulo de administración, en el menú del lado izquierdo se encuentra el botón de <i>usuarios</i> . La siguiente pantalla nos muestra los usuarios existentes y también aquí se ubica el botón “ <i>nuevo</i> ” para crear un nuevo usuario y asignar los datos y permisos correspondientes
Nivel:	El nivel de este caso de uso es medio, ya que puede trabajarse con el usuario administrador sin embargo, una de las piezas básicas de todo sistema usable es la administración de usuarios.
Actor Principal:	Administrador
Parte Interesada:	El o los administradores del sistema son los principales interesados en este caso de uso.
Iniciador del Caso de uso:	Se puede crear usuarios en cualquier momento, así que este caso de uso se inicia justamente cuando es necesario agregar un nuevo usuario al sistema.
Pre-condiciones:	Para poder crear usuarios, es necesario logearse en el sistema con un usuario que tenga los privilegios adecuados, como el administrador, de no ser así no se podrá crear usuarios. Otra precondicion es que el usuario que se quiera crear no exista en la base de datos.

Flujo de Desempeño Óptimo:

Paso	Entrada	Respuesta del Sistema
1	El usuario selecciona el botón <i>Administrar</i> .	Se muestra una lista de de opciones en la parte superior izquierda del navegador, una de estas opciones corresponde a <i>Usuarios</i> .
2	El usuario selecciona la opción <i>Usuarios</i> .	Se muestra una lista con los usuarios registrados en el sistema, la cual se puede consultar mediante múltiples parámetros. Además también se muestra un botón en el que se lee <i>Nuevo</i> , que sirve para crear un nuevos usuarios
3	El usuario presiona el botón <i>Nuevo</i> , para crear un nuevo usuario.	Se piden algunos datos del nuevo usuario como: ³⁵ ₁₇ Nombre de Usuario. ³⁵ ₁₇ Nombre. ³⁵ ₁₇ Contraseña. ³⁵ ₁₇ Datos de contacto. ³⁵ ₁₇ Idioma. Los campos sombreados de color azul son obligatorios, lo que significa que si no son proporcionados por el usuario, no puede crearse el

		nuevo usuario. Los grupos a los que pertenece por default un nuevo usuario son: ³⁵ ₁₇ Employee. ³⁵ ₁₇ Partner Manager En el flujo alternativo se mostrara como crear un nuevo grupo y asignarlo al usuario recién creado.
4	Se capturan los campos correspondientes y se presiona el botón <i>Guardar</i> .	El sistema muestra la pantalla con el resumen de los datos del nuevo usuario.

Flujo Alternativo:

Paso	Entrada	Respuesta del Sistema
3.1	El usuario presiona el botón de <i>Añadir</i> , dentro de la ventana de <i>Grupos</i> .	El sistema muestra una lista de grupos existentes, y el usuario puede seleccionar tantos como quiera o crear uno nuevo. En caso de haber seleccionado mas de un grupo con los mismos permisos pero diferentes privilegios, se le asignara al usuario el privilegio de mayor nivel.
3.2	El usuario selecciona el botón <i>Nuevo</i> , para crear un nuevo grupo.	Se muestra una nueva ventana donde se puede capturar el campo del nuevo grupo y también se muestra un panel con varias pestañas: ³⁵ ₁₇ Usuarios: Usuarios que pertenecen a este grupo. ³⁵ ₁₇ Menús: Menús a los que tiene acceso este grupo. ³⁵ ₁₇ Permisos de acceso: Lectura y/o Escritura. ³⁵ ₁₇ Reglas: Reglas de acceso a este grupo. ³⁵ ₁₇ Notas: Nota referente al grupo.
3.3	El usuario captura los datos necesarios y genera el nuevo grupo. Desde la pestaña de usuarios se agrega el usuario que está siendo creado y se presiona <i>Guardar</i> .	El sistema vuelve a mostrar la pantalla con los datos del usuario que ha sido creado, pero ahora además de los permisos que ya tenía este usuario, se puede ver el permiso recién asignado.
4.4	Finalmente en la pantalla del resumen del usuario, se presiona el botón <i>Guardar</i> y así el usuario ha sido creado y ya cuenta con los privilegios asignados.	El sistema bloquea los datos del usuario, indicando que la operación ha sido exitosa.

Post-condiciones:	Puede cerrarse la sesión del administrador e ingresar con el usuario recién creado para verificar los permisos asignados al mismo.
Excepciones no controladas:	³⁵ ₁₇ No se puede crear el usuario debido a que no hay conexión con el servidor de la base de datos. ³⁵ ₁₇ No hay espacio suficiente en memoria para poder crear al nuevo usuario.
Consecuencias de Fallo:	³⁵ ₁₇ De existir un fallo al momento de crear al nuevo usuario, este no se genera y por lo tanto no se puede ingresar al sistema.
Garantías Minina:	³⁵ ₁₇ Se garantiza que de ocurrir algún problema al momento de generar el nuevo usuario, esto no afectara el desempeño del sistema por lo que basta con repetir el procedimiento para poder concluirlo. ³⁵ ₁₇ De igual manera se garantiza que al momento de crear una nueva usuario, no serán alterados los datos almacenados en la base de datos.

	³⁵ ₁₇ Se garantiza la persistencia de los datos del nuevo usuario.
Frecuencia:	Este caso de uso tiene una frecuencia media, pero depende en gran medida del número de usuarios que sean generados en el mismo.
Supuestos:	Se asume que el usuario encargado de administrar usuarios tenga conocimientos de los procesos del sistema en general.

Comentarios:

Desempeño:

- ³⁵₁₇ Siempre que se crea un nuevo usuario, se le asignan los grupos básicos:
- ³⁵₁₇ Employee: grupo de empleados, este cuenta con privilegios básicos para los empleados del sistema.
- ³⁵₁₇ Partner Manager: grupo de asociados, cuando un usuario es creado en el sistema se considera un asociado y cuenta con privilegios de acceso a datos del usuario.

Interfaz de Usuario:

- ³⁵₁₇ El idioma del sistema puede ser seleccionado al momento de crear al usuario para que la interfaz sea más amigable con el usuario.

Casos Sigüientes:

- ³⁵₁₇ En el siguiente caso de uso mostraremos como realizar modificaciones tanto a los usuarios como a los grupos que les fueron asignados y a sus privilegios.

Tarea 2.1 Alta de Usuarios

Una vez que se haya ingresado como administrador el sistema mostrara la siguiente pantalla. Es el panel principal de OpenERP y es la pantalla que todo usuario puede ver cuando ingresa al sistema, muestra los modulos a los que se tiene acceso y los datos del usuario registrado.

Implementando la Factura Digital mediante SCRUM

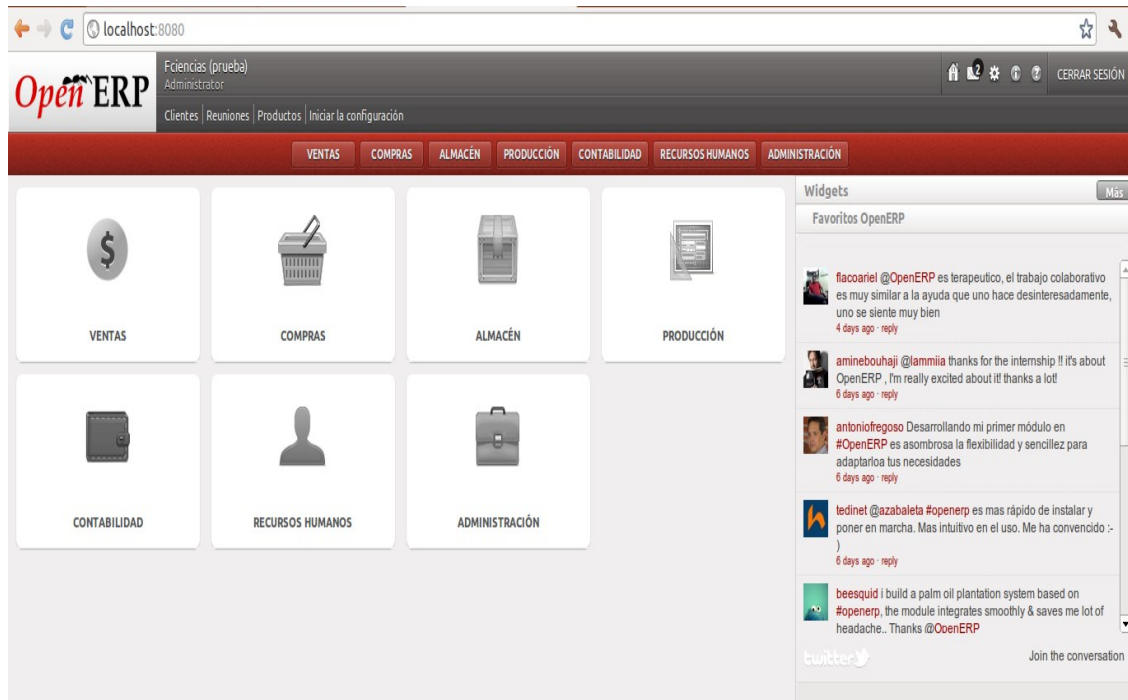
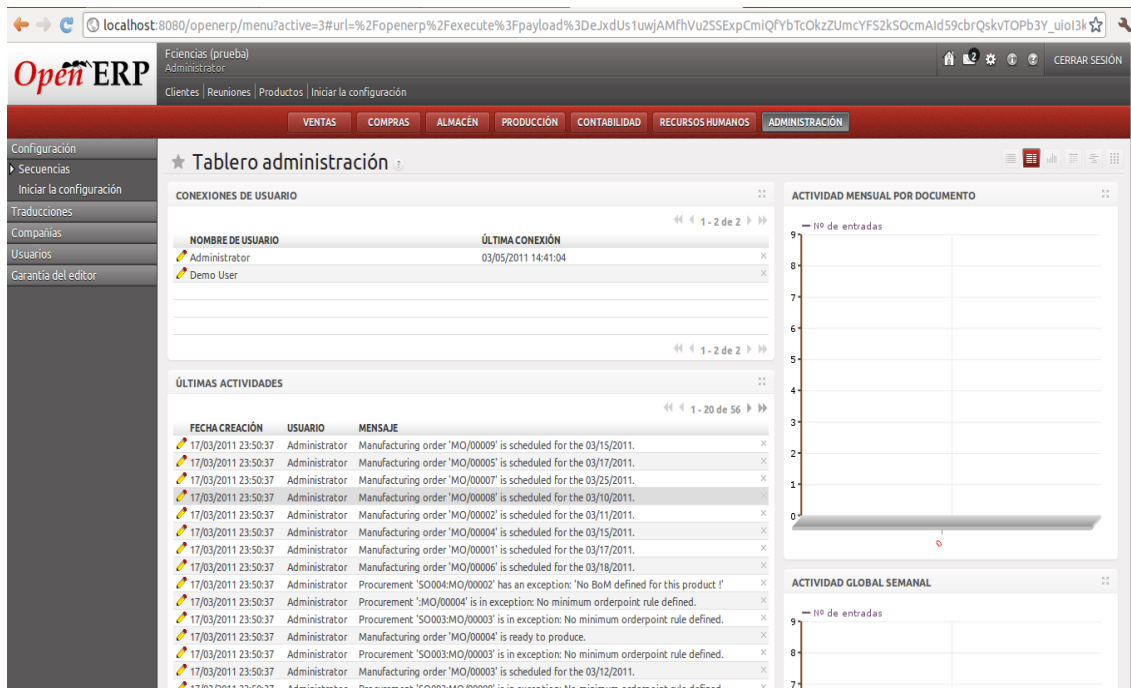


Figura 5.1 Panel principal OpenERP

Posteriormente el usuario debe seleccionar la opción de “ADMINISTRACION” que se puede identificar con el logo del maletín. La siguiente ventana que se muestra después de haber seleccionado esta opción es la correspondiente a la Figura 5.2, en esta ventana se podrán observar los accesos que se han tenido en el sistema.



NOMBRE DE USUARIO	ÚLTIMA CONEXIÓN
Administrator	03/05/2011 14:41:04
Demo User	

FECHA CREACIÓN	USUARIO	MENSAJE
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00009' is scheduled for the 03/15/2011.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00005' is scheduled for the 03/17/2011.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00007' is scheduled for the 03/25/2011.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00008' is scheduled for the 03/10/2011.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00002' is scheduled for the 03/11/2011.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00004' is scheduled for the 03/15/2011.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00001' is scheduled for the 03/17/2011.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00006' is scheduled for the 03/18/2011.
17/03/2011 23:50:37	Administrator	Procurement 'SO004:MO/00002' has an exception: 'No BoM defined for this product'
17/03/2011 23:50:37	Administrator	Procurement 'MO/00004' is in exception: No minimum orderpoint rule defined.
17/03/2011 23:50:37	Administrator	Procurement 'SO003:MO/00003' is in exception: No minimum orderpoint rule defined.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00004' is ready to produce.
17/03/2011 23:50:37	Administrator	Procurement 'SO003:MO/00003' is in exception: No minimum orderpoint rule defined.
17/03/2011 23:50:37	Administrator	Manufacturing order 'MO/00003' is scheduled for the 03/12/2011.
17/03/2011 23:50:37	Administrator	Procurement 'SO003:MO/00008' is in exception: No minimum orderpoint rule defined.

Figura 5.2 Tablero de administración

Implementando la Factura Digital mediante SCRUM

Dentro de esta ventana es necesario ubicar el menú de *Usuarios*, que se muestra en el menú del lado superior izquierdo. Una vez que se haya seleccionado esta opción se podrá ver la pantalla correspondiente a los usuarios registrados en el sistema y es en esta misma donde se puede crear un nuevo usuario al cual podremos asignarle los privilegios adecuados y con el cual podremos realizar las tareas adecuadas para generar una factura electrónica.

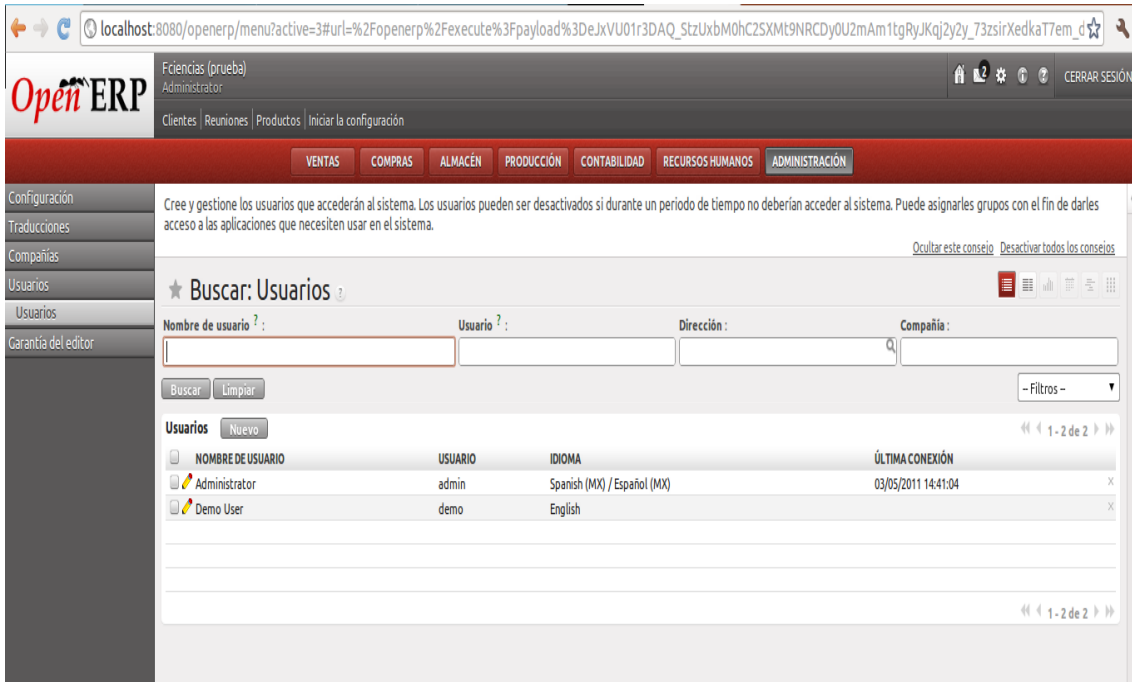


Figura 5.3 Panel de Alta y Búsqueda de usuarios

En la ventana de la Figura 5.3, podemos ver la lista de usuarios registrados y arriba de esta lista se ubica el botón *Nuevo*, que nos sirve para crear un nuevo usuario, mismo que al presionarlo se muestra una ventana de captura de datos para el nuevo usuario, como la que se muestra en la Figura 5.4.

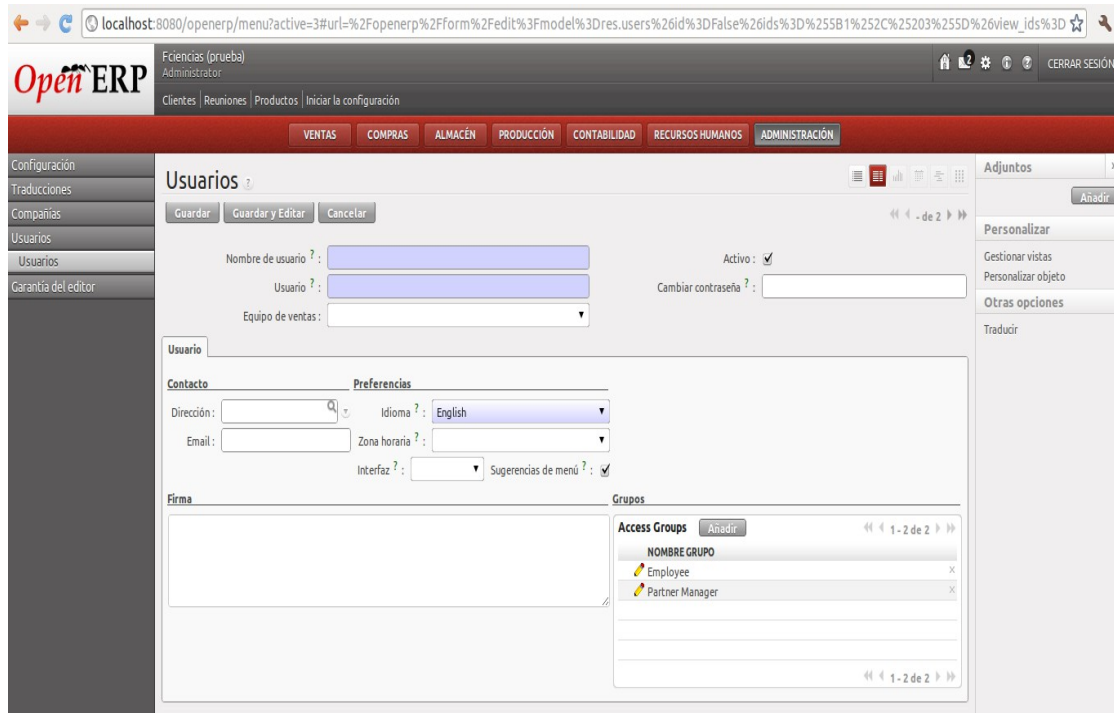


Figura 5.4 Alta de Usuarios

En esta ventana es necesario capturar los datos del usuario, los campos en color oscuro son campos obligatorios y de no ser capturados no se permite la creación del usuario. Al terminar la captura de datos, se puede presionar el botón de *Guardar* y con estos pasos el usuario ha sido creado, sin embargo de ser así, los grupos y por consiguiente permisos asignados a este usuario, son los correspondientes al grupo *Employee* y *Partner Manager*.

Tarea 2.1.1 Asignación de Grupos.

Para poder asignar permisos es necesario presionar el botón *Añadir* que se encuentra en la sección de *Grupos*, este botón lo podemos ver en la parte inferior derecha de la Figura 5.4. Una vez presionado este botón, se muestra la siguiente pantalla donde se puede seleccionar los grupos a los que pertenecerá el usuario que esta siendo creado.

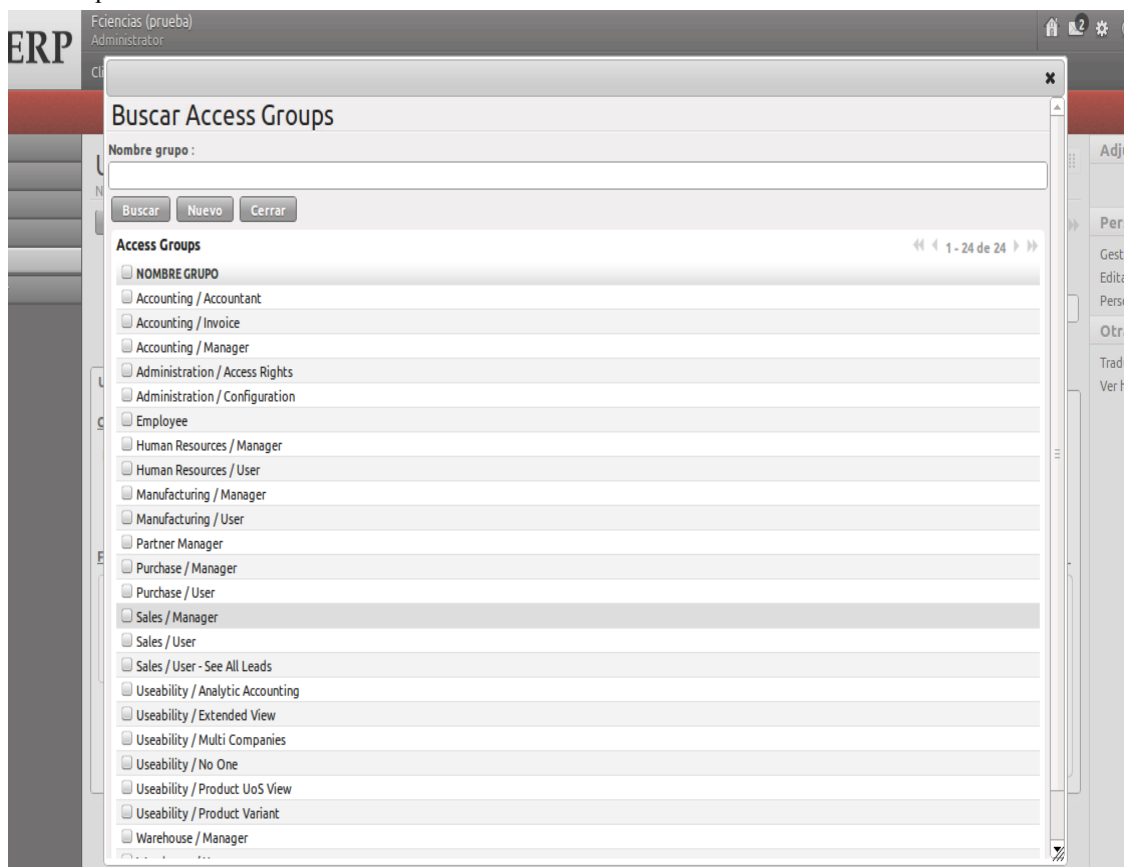


Figura 5.5 Búsqueda de grupos

En esta pantalla es posible seleccionar todos los grupos a los que pertenece el usuario en cuestión, esto se hace palomeando el cuadro de selección que aparece del lado izquierdo de la lista de grupos y posteriormente presionar el botón *Seleccionar*, este botón se muestra una vez que se ha seleccionado al menos un grupo. Ya que han sido seleccionados los grupos deseados y estos se muestran en el resumen de los datos del usuario, se puede presionar guardar y así el usuario es creado y pertenece a todos los grupos seleccionados,

Retrospectiva del ciclo:

Como se puede ver en este ciclo, una parte fundamental de todo sistema es la que corresponde a la parte de permisos y privilegios. En general este ciclo tiene una prioridad baja, ya que es posible trabajar con el usuario administrador para realizar las tareas correspondientes al sistema, sin embargo en esta tesis se busca generar una solución a aquellas empresas interesadas en la administración y organización de sus procedimientos.

A lo largo de este ciclo, notamos la necesidad de agregar la tarea al *Product Backlog* de instalación del modulo de facturación electrónica, ya que no existe y es necesario explicar cómo llevar a cabo este sencillo paso.

5.2 Ciclo 3: Modificación de Usuarios y Permisos

Como ya es una costumbre en este ciclo haremos uso del diagrama general de casos de uso y en particular analizaremos el caso de uso correspondiente a la modificación de usuarios y de sus permisos. Este caso de uso corresponde al caso de uso 2.- *Administrar Usuarios*. Dentro de los objetivos principales de este ciclo tenemos los siguientes:

- ³⁵₁₇ Modificación de Usuario.
- ³⁵₁₇ Modificación de Permisos.
- ³⁵₁₇ Baja de Usuario.
- ³⁵₁₇ Gestión de Permisos.

En general se manejan muchos tipos de grupos, como:

- ³⁵₁₇ Ventas: usuarios destinados únicamente a vender productos o servicios.
- ³⁵₁₇ Compras: usuarios que se encargaran del reabastecimiento.
- ³⁵₁₇ Punto de Venta: usuarios que se encargaran de realizar los cobros de las ventas.
- ³⁵₁₇ Recursos Humanos: usuarios que se encargan de administrar los recursos humanos

Sin embargo todos ellos se pueden clasificar en 3 tipos principales:

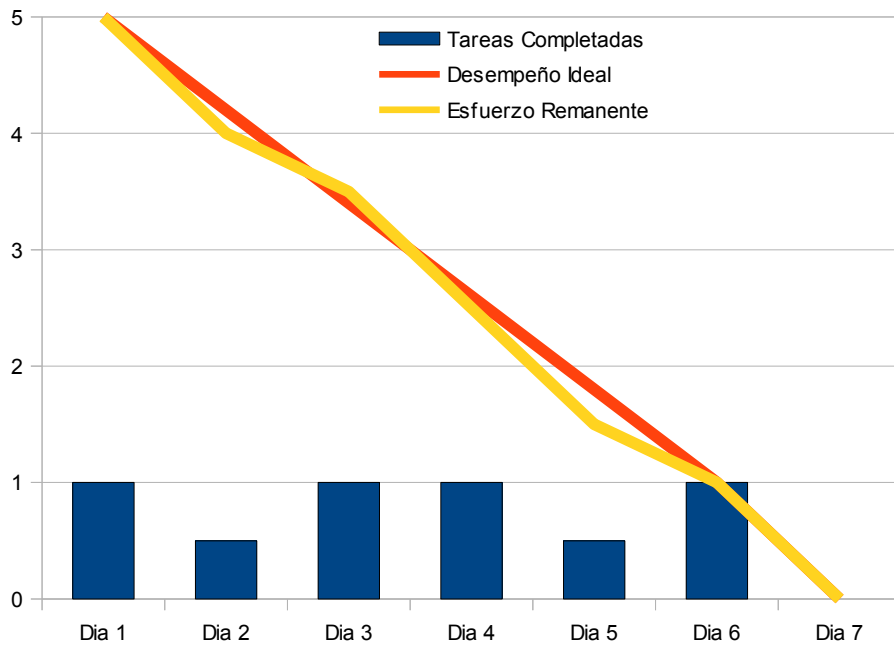
- ³⁵₁₇ Administrator: es el rango más alto de los privilegios que se puede tener en un grupo
- ³⁵₁₇ Manager: es un rango medio con el cual únicamente se tiene acceso a algunas opciones en el grupo
- ³⁵₁₇ User: rango básico destinado a usuarios que no tendrán ningún rol administrativo.

La manera más sencilla de entender el concepto de grupo, es comprendiendo que una empresa puede (o debe) estar dividida en muchas áreas, y cada una de estas áreas debe tener una jerarquía. Así puede ser que dentro del departamento de ventas existan varios usuarios que necesiten tener acceso a este modulo, así que lo mejor es asignarle el grupo *Ventas/Usuarios* o si se prefiere, se puede crear un nuevo grupo personalizado para este grupo de usuarios y así los podemos separar del grupo de supervisores del área de ventas que bien pueden ser asignados al grupo *Venta/Administrador*. Además debe existir un usuario que pertenezca a todos los grupos o bien que tenga todos los permisos de acceso que se pueden asignar en el sistema, este usuario es conocido como *Administrador*.

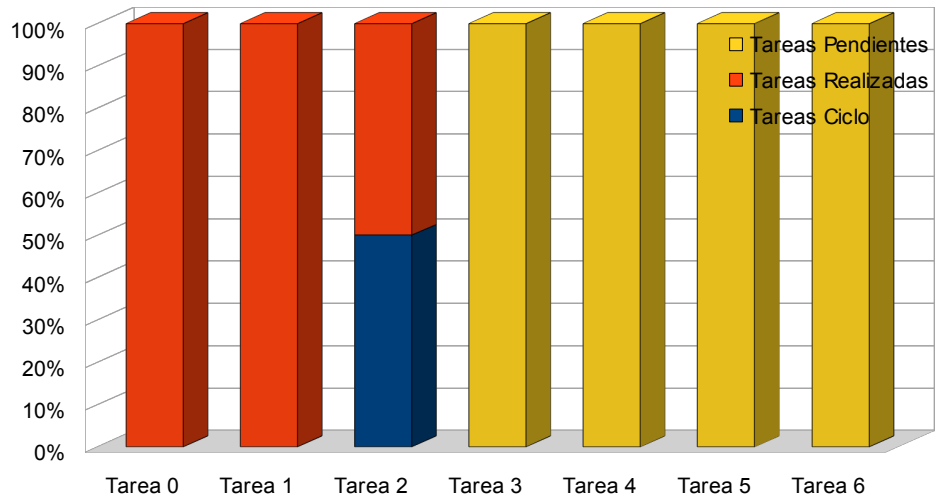
Sprint Backlog: SiFaDig	
Ciclo: 3	<p>Objetivos Generales: Este ciclo puede considerarse una continuación del ciclo anterior, esto principalmente debido a que después de haber creado un usuario y asignar los permisos correspondientes, es deseable realizar modificaciones sobre este usuario y sus permisos o incluso eliminar al usuario. Será objetivo fundamental de este ciclo mostrar cómo modificar los datos ingresados al momento de crear un usuario, cómo modificar los permisos que tenga asignados y cómo dar de baja un usuario. Algunas de estas tareas fueron parcialmente ejemplificadas en el ciclo pasado y es objetivo fundamental de este ciclo complementar las tareas analizadas en el ciclo pasado para así poder dar por cerrado el caso de uso Administrar Usuarios.</p> <p>Tareas para este Ciclo:</p> <ul style="list-style-type: none"> ³⁵₁₇ Modificación de Usuario. ³⁵₁₇ Modificación de Permisos. ³⁵₁₇ Baja de Usuario. ³⁵₁₇ Gestión de Permisos.
Fecha de Inicio: 02/05/2011	
Fecha de Terminó: 08/05/2011	

Tareas			
Id	Descripción de la Tarea	Tamaño	Prioridad
2.2	Modificación de Usuario	1	4
2.2.1	Modificación de Permisos	1	4
2.3	Baja de Usuario	1	4
2.4	Gestión de Permisos.	2	4

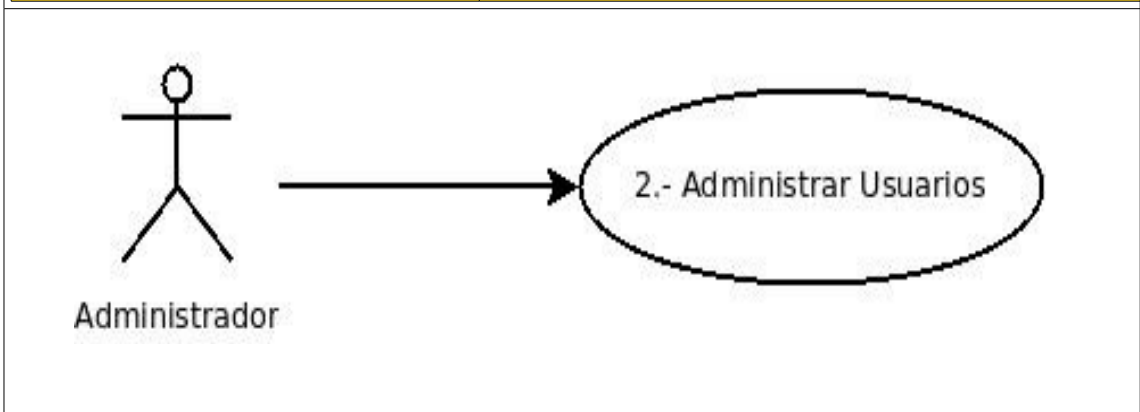
Gráfica de Quemado Ciclo 3



Gráfica de Avance Ciclo 3



Caso de Uso: Administrar Usuarios	Identificador: 2
--	-------------------------



Descripción:	Cualquier usuario con los privilegios adecuados puede modificar usuarios y los permisos asignados, incluso puede eliminar al usuario o crear una nueva serie de permisos y asignarlos al usuario que se desea modificar. Para no crear un nuevo usuario y para mantener el rol de administrador, como el rol con mayor rango, usaremos al usuario “administrador” para realizar las tareas correspondientes a este ciclo. Una vez que se haya ingresado en el sistema es necesario ubicar el botón para acceder a la administración del sistema, dentro del módulo de administración, en el menú del lado izquierdo se encuentra el botón de <i>usuarios</i> . La siguiente pantalla nos muestra los usuarios existentes y también aquí se ubica el botón <i>editar</i> para modificar los datos de un usuario ya existente, así como para editar los permisos asignados a éste, también en este menú podemos eliminar usuarios.
Nivel:	El nivel de este caso de uso es alto, debido a que una de las principales características de todo ERP es el que permita modificar elementos que interactúan con el sistema, en este caso usuarios. De no permitir la Modificación de usuarios, el sistema sería prácticamente inoperante ya que si se crea un usuario con datos incorrectos la única solución es crear uno nuevo, pero esto solo representa un desperdicio de valiosos recursos del sistema.
Actor Principal:	Administrador.
Parte Interesada:	El administrador del sistema es el principal interesado en este caso de uso.
Iniciador del Caso de uso:	Al momento de querer modificar algún usuario, ya sea por algún error al momento de su captura o porque es necesario modificar sus permisos y privilegios.
Pre-condiciones:	Para poder administrar usuarios y sus privilegios, es necesario ingresar al sistema con un usuario que tenga los privilegios adecuados, como el administrador, de no ser así no se podrá tener acceso al menú para administrar usuarios. Otra precondición es que el usuario que se quiera editar ya exista en la base de datos.

Flujo de Desempeño Óptimo:

Paso	Entrada	Respuesta del Sistema
1	El usuario selecciona el botón <i>Administrar</i> .	Se muestra una lista de de opciones en la parte superior izquierda del navegador, una de estas opciones corresponde a <i>Usuarios</i> .
2	El usuario selecciona la opción <i>Usuarios</i> .	Se muestra una lista con los usuarios registrados en el sistema, la cual se puede consultar mediante múltiples parámetros.
3	El usuario presiona <i>click</i> , sobre el usuario	Se muestra la pantalla con los datos

	que desea modificar/eliminar.	<p>correspondientes al usuario seleccionado:</p> <p>³⁵₁₇ Nombre de Usuario. ³⁵₁₇ Nombre. ³⁵₁₇ Contraseña. ³⁵₁₇ Datos de contacto. ³⁵₁₇ Idioma.</p> <p>También se muestra en la parte inferior derecha, los grupos a los que pertenece el usuario y por ende, los privilegios con los que cuenta dicho usuario. Si los grupos del usuario no han sido modificados después de haber sido creado, los grupos a los que pertenece son:</p> <p>³⁵₁₇ Employee. ³⁵₁₇ Partner Manager</p> <p>Además arriba del nombre de usuario, se muestran los botones <i>Editar</i> y <i>Eliminar</i>.</p> <p>En el flujo alternativo se mostrará como modificar los datos y grupos del usuario seleccionado.</p>
4	El usuario presiona el botón <i>Eliminar</i> .	El sistema muestra la pantalla en la que se pide confirmar que se desea eliminar al usuario.
5	El usuario selecciona <i>Aceptar</i> .	Se devuelve al usuario a la ventana de usuarios.

Flujo Alternativo:

Paso	Entrada	Respuesta del Sistema
4.1	El usuario presiona el botón de <i>Editar</i> , dentro de la ventana con el resumen de los datos y privilegios del usuario.	El sistema convierte editables los campos correspondientes a este usuario, para que el administrador pueda modificar los datos y grupos.
4.2	El usuario captura las modificaciones, puede eliminar algún grupo al que pertenezca este usuario seleccionando en la <i>x</i> de dicho grupo. Y también puede modificar los accesos y privilegios de dicho grupo. EL usuario selecciona el grupo al que pertenece el usuario y que desea modificar.	<p>El sistema muestra la pantalla correspondiente al grupo seleccionado. En la parte superior se muestra una serie de pestañas:</p> <p>³⁵₁₇ USUARIOS: Son los usuarios del sistema de OpenERP que acceden al aplicativo desde cualquiera de las opciones válidas de clientes actualmente disponibles en OpenERP (GTK, web, Koo, Mac). Los usuarios representan a las personas físicas. Estos se identifican con un nombre de usuario y una contraseña.</p> <p>³⁵₁₇ MENU: Los usuarios, en función del rol que tengan asignado, estarán facultados para acceder a determinadas secciones de Menú. Así por ejemplo al grupo de usuarios comerciales, se les puede establecer que no dispongan de acceso al menú de área financiera.</p> <p>³⁵₁₇ PERMISOS DE ACCESO: Corresponde a los permisos que se tendrán dentro de las aplicaciones: leer, escribir, crear y eliminar.</p> <p>³⁵₁₇ REGLAS: Es una herramienta que nos permite simplificar el proceso de otorgar determinados roles a grupos de usuarios. Especialmente útil cuando tenemos</p>

		múltiples usuarios. Así por ejemplo si deseamos que en una organización de 18 socios comerciales cada cual vea las cuentas que tiene asignada, es más sencillo aplicar sobre el grupo reglas que hacerlo usuario por usuario.
4.3	El usuario selecciona alguna de las opciones mencionadas en el paso anterior y presiona el botón <i>añadir</i> para agregar más objetos al elemento seleccionado o elimina algún elemento del grupo presionado la x.	Si el usuario desea agregar más objetos a alguna de las opciones mencionadas en el paso anterior, el sistema mostrará una nueva ventana donde el usuario puede buscar o crear usuarios, permisos, menús, reglas. Si elimina algún objeto del elemento seleccionado, el sistema simplemente lo quita de la lista.
4.4	Dentro de la pantalla del grupo que se estaba editando el usuario presiona <i>Guardar</i> .	El sistema bloquea los campos correspondientes a las pestañas usuarios, permisos, menús reglas.
4.5	Se presiona el botón “Cerrar” .	El sistema regresa a la pantalla inicial del usuario a modificar.
4.6	Por último el usuario presiona el botón de “Guardar” para registrar todos los cambios realizados al usuario.	Se inhabilitan para modificación los campos correspondientes al usuario seleccionado, hasta que se vuelva a presionar el botón <i>Editar</i>.
	Post-condiciones:	Los campos correspondientes al usuario que está siendo modificado, se inhabilitan y no pueden ser modificados. Además se puede volver a ingresar a la pantalla de dicho usuario y corroborar que los campos han sido modificados.
	Excepciones no controladas:	No se puede crear el usuario debido a que no hay conexión con el servidor de la base de datos.
	Consecuencias de Fallo:	³⁵ ₁₇ De existir un fallo al momento de modificar un usuario, los cambios realizados no se almacenan en la base de datos y por lo tanto la información del usuario se mantiene intacta.
	Garantías Minima:	³⁵ ₁₇ Se garantiza que de ocurrir algún problema al momento de modificar al usuario, esto no afectara el desempeño del sistema por lo que basta con repetir el procedimiento para poder concluirlo. ³⁵ ₁₇ De igual manera se garantiza que al momento de modificar un usuario, no serán alterados los datos almacenados en la base de datos. ³⁵ ₁₇ Se garantiza la persistencia de los datos del usuario modificado.
	Frecuencia:	Este caso de uso tiene una frecuencia media.
	Supuestos:	Se asume que el usuario encargado de administrar usuarios tenga conocimientos de los procesos del sistema en general.

Comentarios:

Desempeño:

- ³⁵₁₇ Siempre que se crea un nuevo usuario, se le asignan los grupos básicos:
- ³⁵₁₇ Employee.
- ³⁵₁₇ Partner Manager.

Interfaz de Usuario:

- ³⁵₁₇ El idioma del sistema puede ser seleccionado al momento de crear al usuario para que la interfaz sea mas amigable con el usuario.

³⁵₁₇ Para poder tener acceso a los menús necesarios para las modificaciones vistas en este ciclo, es necesario que el administrador tenga habilitada la *Interfaz Avanzada*, de otro modo no podrá realizar las modificaciones correspondientes. Para activar dicha interfaz, es necesario seguir la secuencia *Menú->Usuarios->(Administrador)Interfaz: Avanzada->Guardar*.

Casos Siguietes:

³⁵₁₇ En el siguiente caso de uso, el cual es administración de usuarios mostraremos como realizar modificaciones tanto a los usuarios como a los grupos que les fueron asignados y a sus privilegios.

Tarea 2.3: Baja de Usuario

Una vez que se haya ingresado como administrador el sistema mostrará la siguiente pantalla, la cual muestra los menús y opciones correspondientes al administrador.

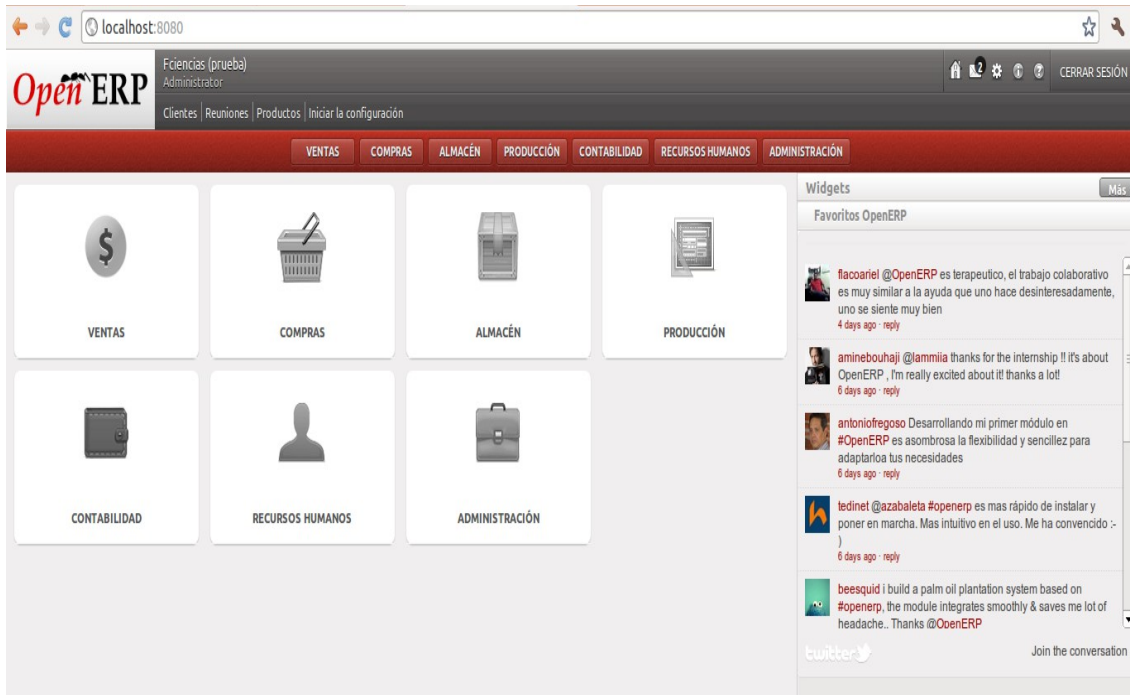


Figura 5.6 Panel Inicial del Administrador

Posteriormente el usuario debe seleccionar la opción de “ADMINISTRACION” que se puede identificar con el logo del maletín. La siguiente ventana que se muestra después de haber seleccionado esta opción es la correspondiente a la Figura 5.7 y es el panel en el cual se pueden muestran los accesos al sistema.

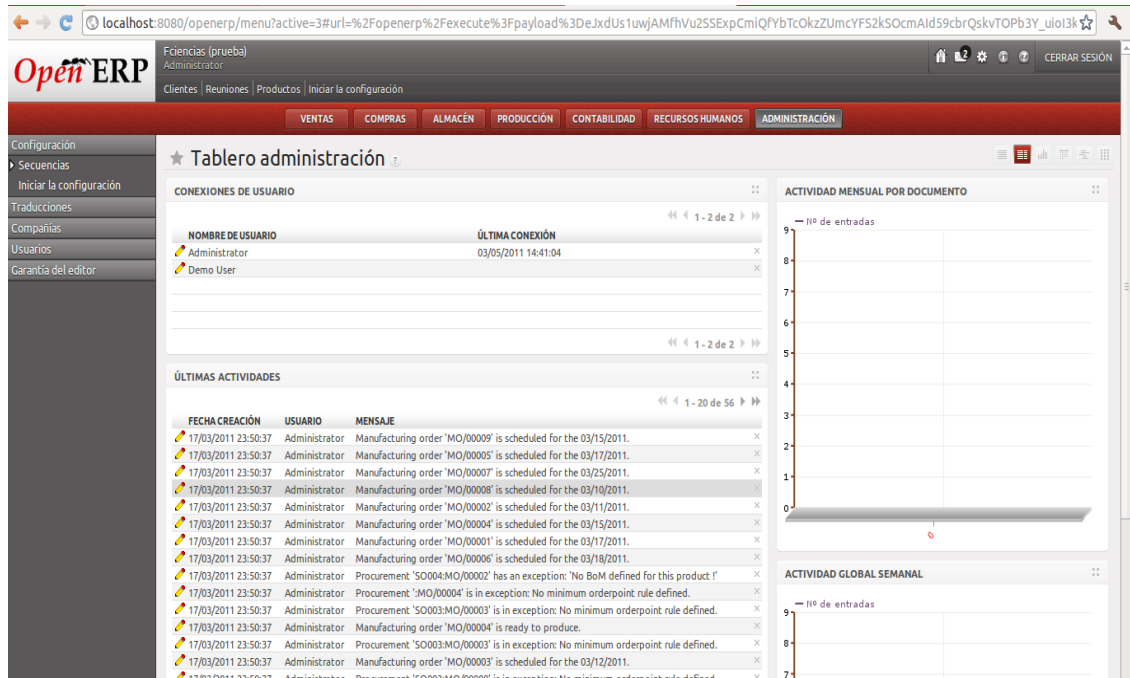


Figura 5.7 Tablero de administración

Dentro de esta ventana es necesario ubicar el menú de *Usuarios*, que se ubica en el menú del lado superior izquierdo. Una vez que se haya seleccionado esta opción se muestra la pantalla correspondiente a los usuarios registrados en el sistema y es en esta misma donde se puede crear un nuevo usuario.

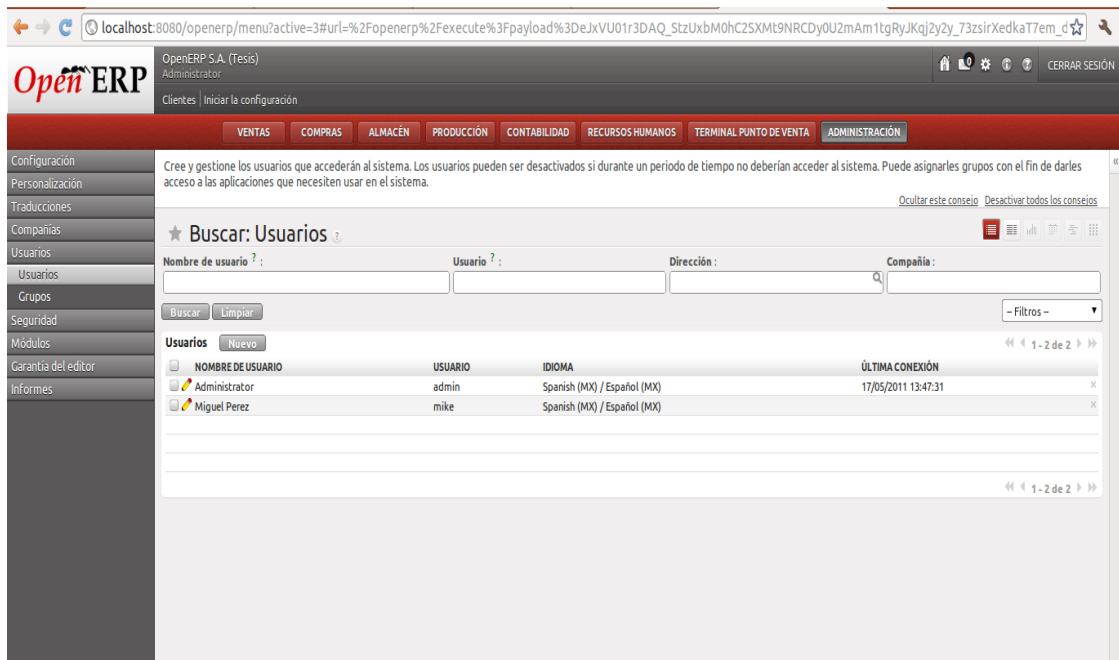


Figura 5.8 Panel de Alta y Búsqueda de usuarios

En la ventana de la Figura 5.8, podemos ver la lista de usuarios registrados (ya he creado un usuario de prueba “mike”). Dentro de esta lista, es posible seleccionar algún usuario y ya sea que modifiquemos sus datos o que lo eliminemos de forma definitiva presionando la *x* que se encuentra al final de la línea en la que se encuentre el usuario. También se puede eliminar un usuario seleccionándolo y posteriormente presionar el botón *eliminar*. En este caso vamos a eliminar al usuario *mike*.

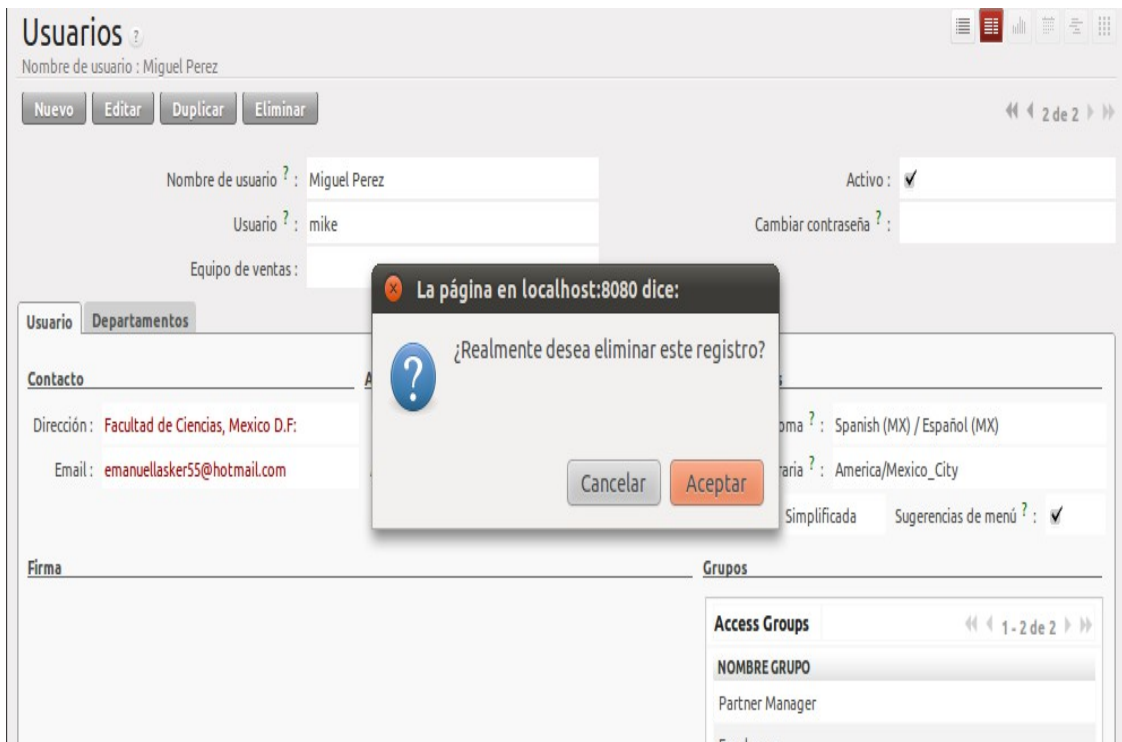


Figura 5.9 Advertencia de eliminado

En esta ventana es necesario confirmar que se desea eliminar al usuario seleccionado presionando el botón *aceptar*. De ser así el sistema elimina al usuario y regresa al usuario a la ventana en la que se muestra la lista de usuarios, como se aprecia en la Figura 5.10.

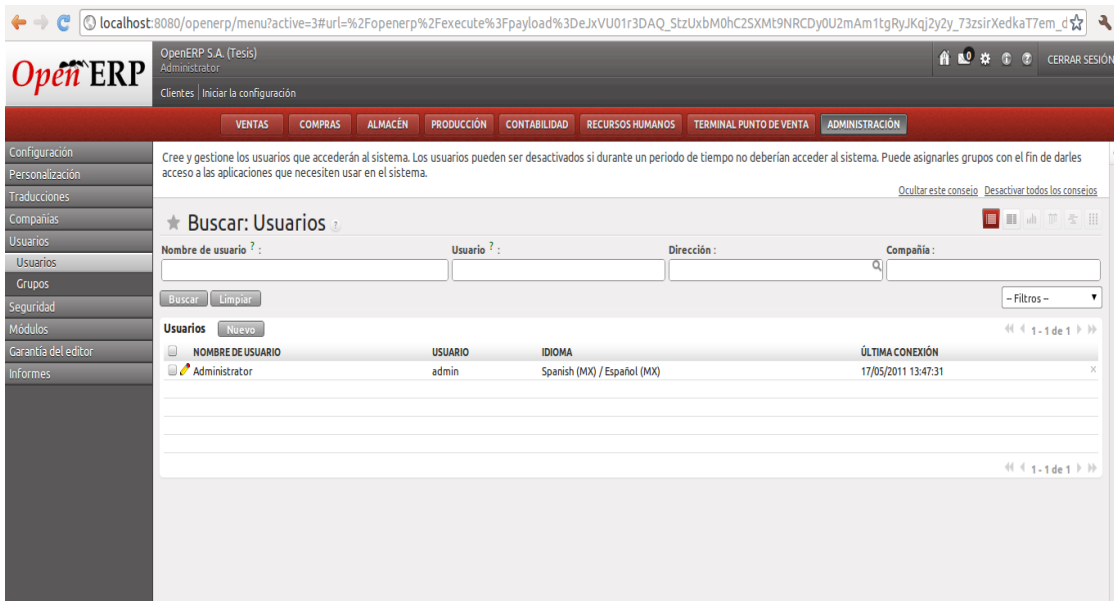


Figura 5.10 Panel de Alta y Búsqueda de usuarios

Tarea 2.2: Modificación de Usuarios.

Para poder modificar los datos, grupos, permisos de un usuario seleccionado, es necesario desplazarse hasta la pantalla que se muestra en la Figura 3.4. Para fines de este documento, volveremos a utilizar al usuario *mike* para modificar sus datos. Una vez seleccionado el usuario a modificar, es necesario presionar el botón *Editar*, que se encuentra del lado izquierdo del botón *Eliminar*. Una vez que se presiona este botón, los campos correspondientes al usuario, se habilitan para ser modificados, como se puede apreciar en la Figura 5.11.

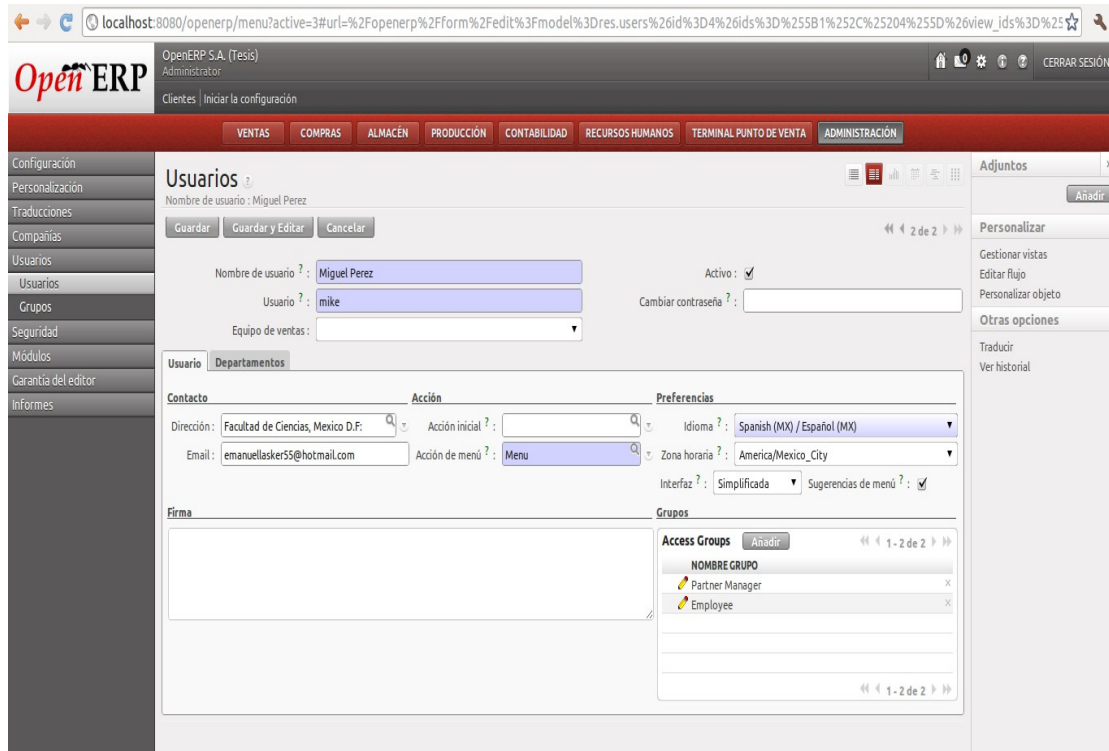


Figura 5.11 Panel de modificación de usuarios

Una vez que el administrador capturo las modificaciones necesarias para este usuario, es posible eliminar algún grupo al que pertenezca o agregar un grupo existente, seleccionando la *x* al final de la línea en la que se encuentra el grupo. También es posible crear un nuevo grupo como se vio en el ciclo pasado,. La Figura 5.12 muestra dentro del campo *Grupos*, el grupo *Tarea2.4*, mismo grupo que procederemos a modificar.

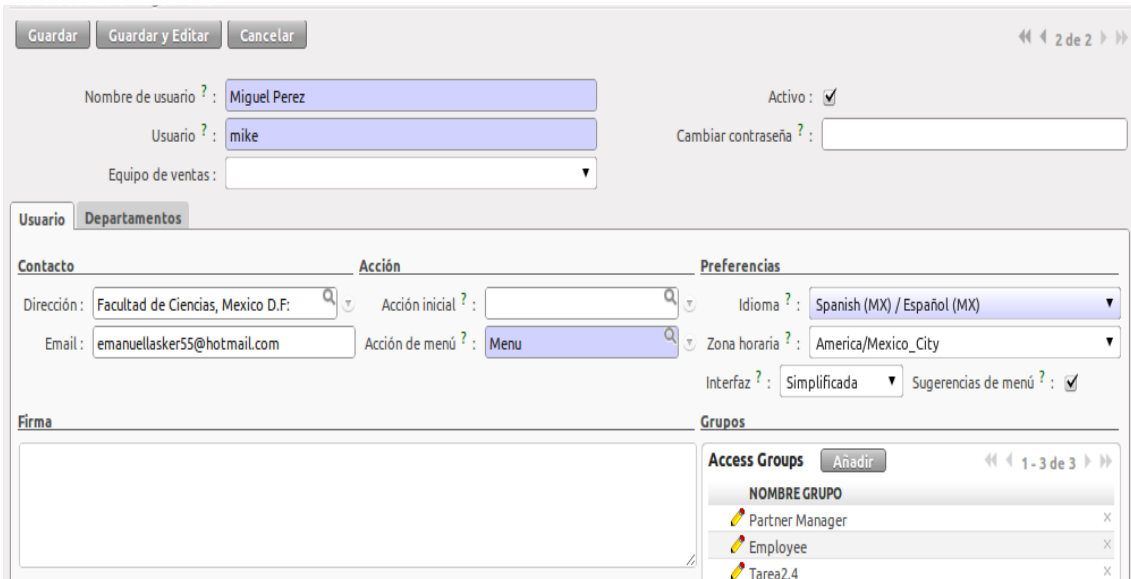


Figura 5.12 Panel de modificación de usuarios

Tarea 2.2.1: Modificar Permisos.

Como podemos ver en la Figura 5.13, se muestra la lista de grupos a los que pertenece el usuario *mike*, seleccionando la *x* de alguno de estos grupos, el usuario deja de pertenecer al grupo, sin embargo el grupo sigue existiendo dentro de *OpenERP*. Por otro lado si se selecciona alguno de estos grupos, este es susceptible a ser modificado, en este caso modificaremos el grupo *Tarea2.4*. Al seleccionar el grupo *Tarea2.4* se mostrara la siguiente pantalla.

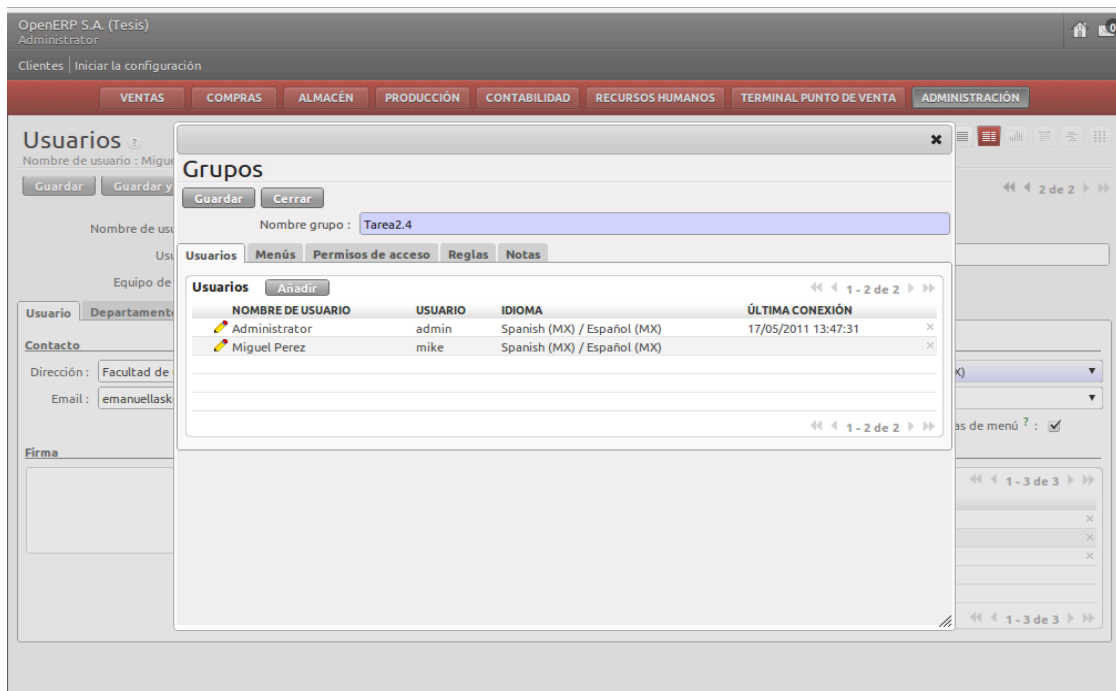


Figura 5.13 Panel de administración de grupos por usuario

Tarea 2.4: Gestión de Permisos.

Como podemos apreciar en la Figura 5.14, el campo del nombre del grupo puede ser modificado. En la parte superior y debajo del nombre del grupo podemos apreciar 5 pestañas y son 4 de ellas las que nos interesan en este ciclo, la sección de notas es muy útil pero en este caso no son de vital importancia:

- 35 17 **USUARIOS:** Son los usuarios del sistema de OpenERP que acceden al aplicativo desde cualquiera de las opciones válidas de clientes actualmente disponibles en OpenERP (GTK, web, Koo, Mac). Los usuarios representan a las personas físicas. Estos se identifican con un nombre de usuario y una contraseña.
- 35 17 **MENU:** Los usuarios, en función del rol que tengan asignado, estarán facultados para acceder a determinadas secciones del menú. Así por ejemplo al grupo de usuarios comerciales, se le puede establecer que no dispongan de acceso al menú de área financiera.
- 35 17 **PERMISOS DE ACCESO:** Corresponde a los permisos que se tendrán dentro de las aplicaciones: leer, escribir, crear y eliminar.
- 35 17 **REGLAS:** Es una herramienta que nos permite simplificar el proceso de otorgar a determinado roles a grupos de usuarios. Especialmente útil cuando tenemos múltiples usuarios. Así por ejemplo si deseamos que en una organización de 18 comerciales cada cual vea las cuentas que tiene asignada, es más sencillo aplicar sobre el grupo reglas que hacerlo usuario por usuario.
- 35 17 **NOTAS:** En esta sección se puede dejar un pequeño comentario respecto al grupo, sus usuarios, sus menús o permisos de accesos.

Un campo al estar sombreado, nos indica que este campo no puede quedar en blanco. Por otro lado también se puede apreciar tanto el botón *Añadir* como la *x* con la que eliminamos objetos que pertenecen a un determinado elemento.

Una vez que el grupo haya sido modificado, ya sea: agregando usuarios, eliminando menús, agregando o quitando permisos o cualquier movimiento que se quiera hacer sobre el grupo, basta con presionar el botón *Guardar* dentro de la ventana del grupo y los campos del grupo se deshabilitan para ser modificados, como se muestra en la Figura 5.14.

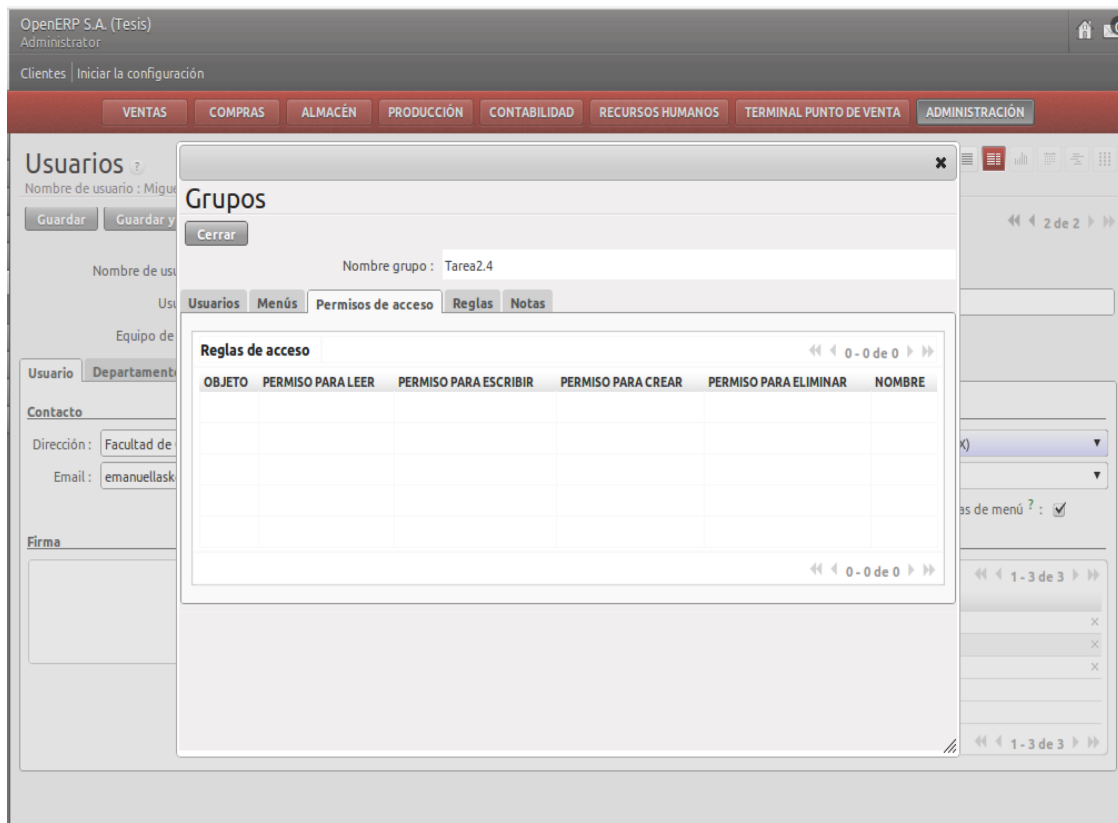


Figura 5.14 Permisos de acceso a grupos

El campo del nombre se deshabilita para modificaciones y se quita el color oscuro, indicando que ya no es posible realizar cambios a este grupo, hasta que se vuelva a seleccionar este grupo para ser modificado desde el menú del usuario. Ahora presionamos el botón *Cerrar* para volver a la pantalla de los datos del usuario y podamos guardar los cambios presionando el botón *Guardar*.

The screenshot displays the 'Panel modificación de usuarios' in OpenERP. At the top, there are buttons for 'Nuevo', 'Editar', 'Duplicar', and 'Eliminar'. The user's name 'Miguel Perez' is displayed in a disabled state. Other fields include 'Usuario' (mike), 'Equipo de ventas', 'Activo' (checked), and 'Cambiar contraseña'. Below the user information, there are tabs for 'Usuario' and 'Departamentos'. The main content area is divided into several sections: 'Contacto' (Dirección: Facultad de Ciencias, Mexico D.F.; Email: emanuellasker55@hotmail.com), 'Acción' (Acción inicial, Acción de menú: Menu), 'Preferencias' (Idioma: Spanish (MX) / Español (MX); Zona horaria: America/Mexico_City; Interfaz: Simplificada; Sugerencias de menú: checked), 'Firma', and 'Grupos'. The 'Grupos' section shows a list of 'Access Groups' with 'NOMBRE GRUPO' as a header, and items like 'Partner Manager', 'Employee', and 'Tarea2.4'.

Figura 5.15 Panel modificación de usuarios

De igual manera que con el grupo, al presionar el botón *Guardar* los campos dejan de ser susceptibles a modificaciones y los cambios realizados al usuario y a sus permisos, son guardados en la base de datos.

Observaciones: Existen menús que no han sido asignados a un grupo en específico y por lo tanto están disponibles para todos los usuarios, siempre y cuando cuenten con acceso al panel inicial (como el menú *Libreta de Direcciones*, dentro del módulo de *Ventas*). Para poder restringir el acceso a este menú y que no sea visible para todo usuario, es necesario asignarlo a algún grupo ya creado, al realizar esta acción, OpenERP determina que el menú no es de acceso a todo usuario y solo se le muestra a los usuarios que pertenezcan al grupo con acceso a este menú.

Para la versión 6.0.1 de OpenERP se han reportado errores al momento de tratar de asignar menús a los grupos. Algunos usuarios han mencionado que al buscar un determinado menú para ser asignado a un grupo, este menú no aparece en la lista de menús disponibles, aunque el menú si exista y se accesible para otros usuarios. Este y otros errores han sido corregidos para la versión 6.0.2, sin embargo en mi experiencia particular con la versión 6.0.1 no he tenido este tipo de problemas, además de que al revisar la documentación respecto a este tema se menciona que únicamente los usuarios de **windows** han reportado estos problemas, por lo que continuare utilizando la versión 6.0.1.

Además es importante enfatizar la diferencia entre **menú** y **permiso de acceso**. Existen algunos que únicamente son visibles si se cuenta con los permisos adecuados. Por ejemplo, aunque un grupo tenga asignado el menú *Ventas*, los usuarios que pertenezcan únicamente a este grupo no podrán visualizar el módulo de *Ventas*, debido a que no cuentan con el permiso para acceder al *panel inicial*²⁵ donde se muestra el módulo de *Ventas*. Es necesario editar el grupo en cuestión y agregar permisos de acceso al *panel inicial*.

²⁵Panel Inicial: Dentro de OpenERP, se conoce como panel inicial a la sección donde se muestran los módulos a los que el usuario tiene acceso. Este panel se muestra en la parte superior izquierda, debajo del logotipo de OpenERP.

Ejemplo: A continuación editare el grupo *Tarea2.4* para mostrar los puntos anteriormente mencionados, el objetivo de este ejemplo será editar el grupo *Tarea2.4* de tal forma que el ó los usuarios que pertenezcan únicamente a este grupo, solo tengan acceso al modulo de *Ventas*, dentro de este a los pedidos de ventas y acceso a los clientes.

El primer paso para poder lograr esto es eliminar todos los usuarios, menús y permisos que tenga asignado el grupo *Tarea2.4*, de tal forma que sea un cascaron para poder crear los accesos a los menús y los permisos correspondientes. La tarea de eliminar usuarios, menús y permisos ya se describo con anterioridad, así que únicamente procedo a mostrar la imagen del grupo *Tarea2.4*, sin usuarios, menús y permisos. En esta pantalla se muestran los menús asociados al grupo *Tarea2.4*.

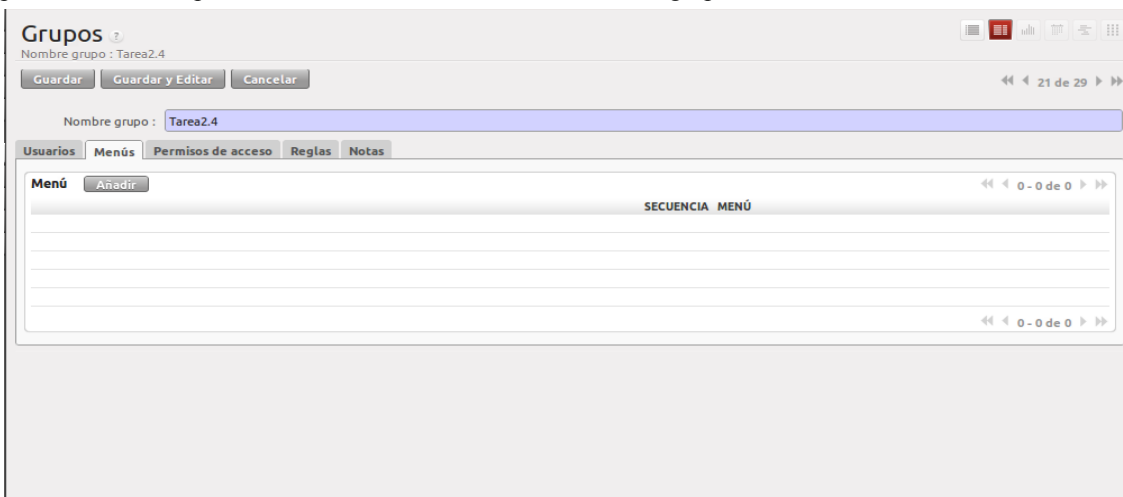


Figura 5.16 Panel administración de menús (vacío)

En esta imagen se puede notar que el grupo *Tarea2.4* ya no cuenta ningún usuario, menú o permiso de acceso. Ahora es necesario agregar los menús y permisos que se desean, en este caso el acceso al menú *Ventas* y sus respectivos submenús y el permiso para el *panel inicial*.

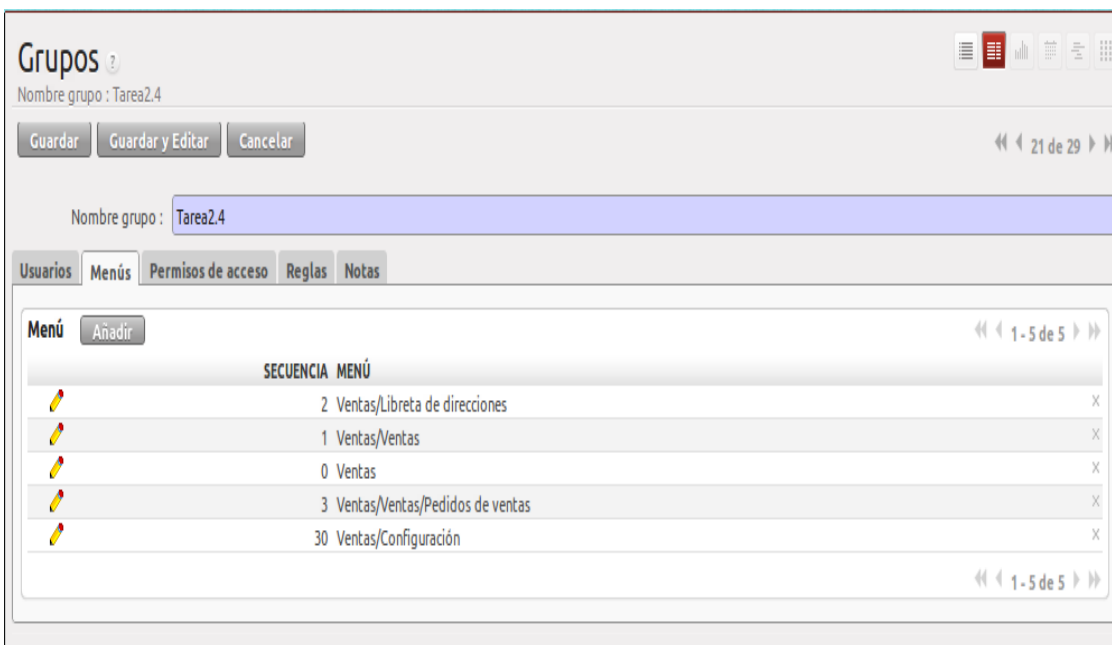


Figura 5.17 Panel administración de menús (vacío)

En la Figura 5.17 podemos apreciar que ya se agregaron los menús necesarios para poder generar una venta y que posteriormente nos permitirán generar una factura electrónica:

³⁵₁₇ Ventas: Modulo principal de ventas

- 35
17 Ventas/Ventas: Submenu de ventas, en el que se puede generar un pedido de ventas (una venta).
- 35
17 Ventas/Libreta de direcciones: Submenu de ventas en el que se puede consultar datos de los clientes.
- 35
17 Ventas/Ventas/Pedidos de ventas: Desde este menú se genera el pedido de venta.
- 35
17 Ventas/Configuración: Acceso a configuraciones básicas del modulo ventas.

Sin embargo si ingresamos con el usuario *mike* podemos notar que este usuario cuenta con acceso a algunos menús a los que no se le ha concedido acceso, como es el caso de *Recursos Humanos* y existen otros como *Pedidos de Ventas* dentro del modulo *Ventas* al cual no tenemos acceso y esto es debido a que los menús a los que se tienen acceso sin haberlo asignado, son menús que no han sido asignados a un grupo en particular. Y en el caso en particular de *Pedidos de Venta*, es necesario asignar permisos de acceso dentro del grupo.



Figura 5.18 Tablero de Ventas

Como se puede ver en la parte superior y del lado izquierdo de la Figura 5.18 el usuario *mike* tiene acceso a los menús y submenús:

- 35
17 Recursos humanos: todo lo relacionado con los empleados, como nomina, vacaciones etc.
- 35
17 Reuniones: se almacenan las posibles reuniones de trabajo con los clientes.
- 35
17 Productos: en este menú se encuentran todo lo relacionado con los productos.

Mismos que en ningún momento concedimos acceso, de igual manera no se puede generar una orden de venta por que no se tienen acceso a ese menú. Basta con asignar los permisos adecuados y asignarle un grupo a estos menús que no pertenecen a un grupo en particular para solucionar el problema.

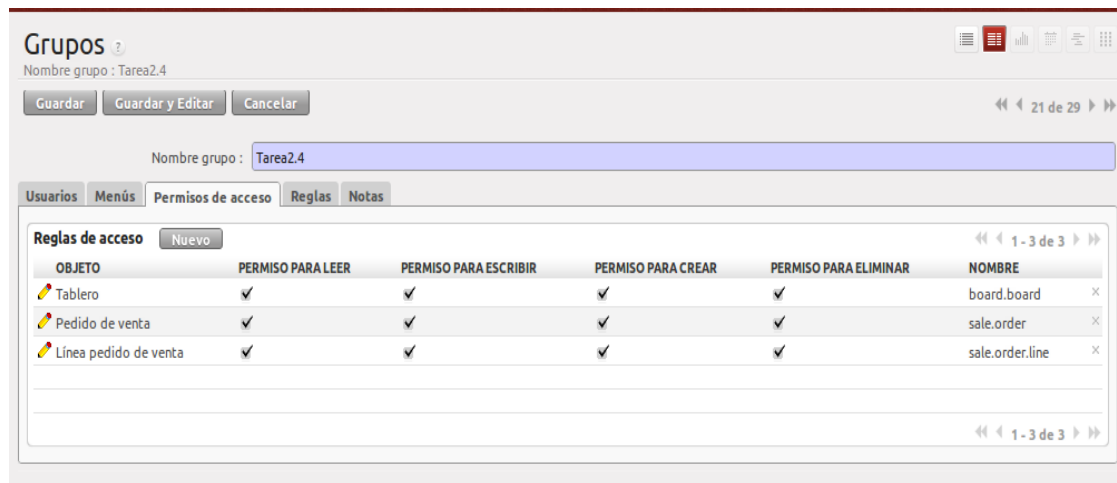


Figura 5.19 Permisos de acceso por grupo

Una vez que asignamos los *permisos de acceso* correctos y relacionamos los menús que están de sobra con algún grupo en particular, se puede apreciar que el usuario *mike* ya cuenta únicamente con los menús y permisos que se especificaron desde un inicio en este ejemplo.

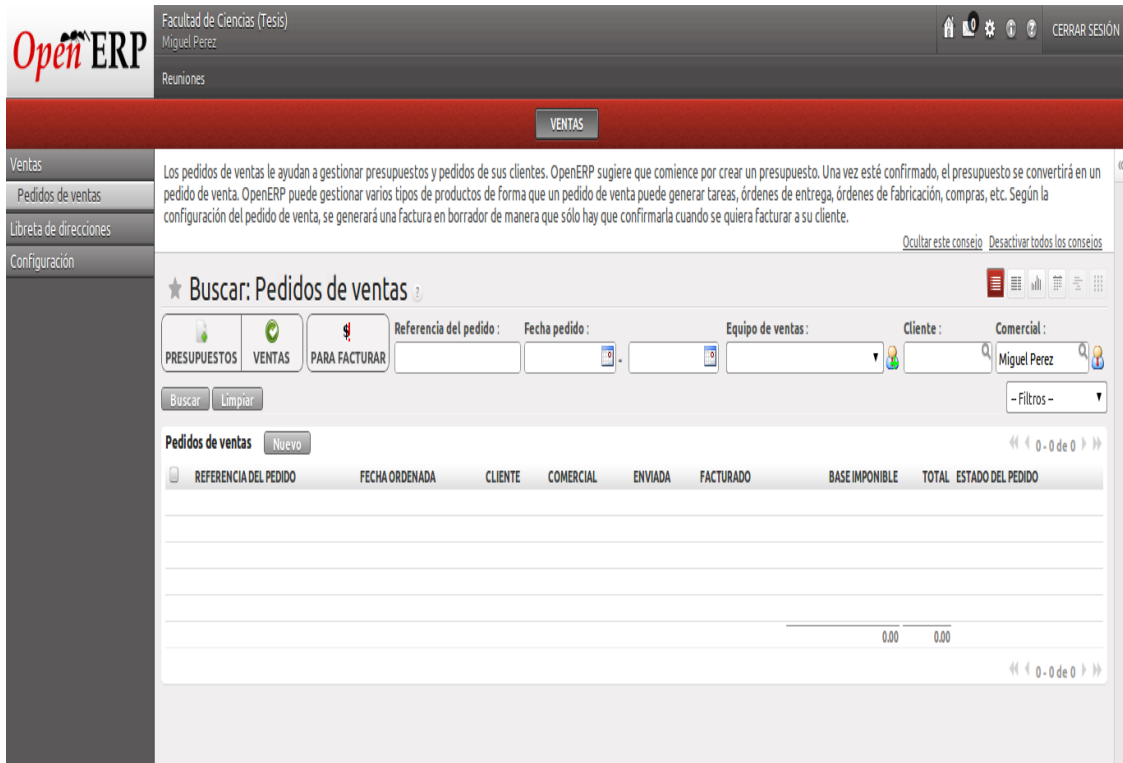


Figura 5.20 Panel de Ventas

Cierre del Ciclo:

Durante este ciclo, pudimos apreciar la complejidad de la estructura de permisos y privilegios, también nos dimos cuenta de la gran versatilidad para poder interactuar con esta, ya que existe más de una manera de realizar las acciones deseadas y aun así no se pierde la integridad y seguridad del sistema.

Además también nos percatamos de la necesidad de modificar algunos partes de los ciclos pasados para aclarar detalles indispensables, como lo es el hecho de habilitar la interfaz avanzada para las tareas correspondientes al ciclo 2 y 3. Además de mencionar que es necesario al momento de la instalación seleccionar los módulos de: clientes, productos, ventas y facturación.

Hasta este punto hemos concluido con la administración de usuarios, ahora es necesario continuar con la administración de clientes y de productos para poder incursionar de lleno en el modulo de facturación electrónica.

5.3 Revisión de Ciclos y Product BackLog

Antes de iniciar este ciclo como normalmente lo he hecho, es importante mencionar que según el *Product BackLog* de *SiFaDig* se ha alcanzado la mitad del número de ciclos del que consta este proyecto y quisiera hacer un pequeño resumen de las tareas realizadas y de las tareas pendientes, además de mostrar la última versión del *Product BackLog* en el cual podremos ver los cambios que ha sufrido desde su versión inicial.

Tareas realizadas:

- ³⁵₁₇ Crear Ambiente de Desarrollo.
- ³⁵₁₇ Ingresar al Sistema.
- ³⁵₁₇ Administración de Usuarios.

Tareas Pendientes:

- ³⁵₁₇ Administración de Productos.
- ³⁵₁₇ Administración de Clientes.
- ³⁵₁₇ Administración de Ventas y Facturas.
- ³⁵₁₇ Reportes.

Para estos momentos ya nos es familiar los conceptos que hemos tratado con las tareas realizadas y que se repetirán en las tareas pendientes. Por mencionar algunos de estos conceptos podemos recordar que en OpenERP se utilizan “reglas” para conceder acceso a determinados menús y de igual manera se hará uso de “reglas” para poder administrar de una manera más adecuada el inventario de productos. Otro de los elementos ya conocidos en OpenERP como lo son los campos “obligatorios”, esto hace referencia a que existen algunos campos dentro de los formularios (por ejemplo el formulario de alta de usuarios) que no pueden quedar vacíos al momento de guardar cambios. Estos campos están sombreados de color oscuro para indicar su importancia, mismos campos que cambian de color en caso de querer guardar cambios sin haber llenado de manera correcta estos campos. Por último algo importante en OpenERP y en la mayoría de los sistemas actuales es el que una misma función puede llevarse a cabo de diferentes maneras y en particular en OpenERP en muchas ocasiones puede llevarse a cabo en más de una forma. En este documento se mostrara la manera más fácil e intuitiva de realizar las tareas, pero se sin que esto signifique que es la única forma de llevar a cabo dicha tarea.

Es importante mencionar estos detalles, ya que a partir de este momento dejaré de hacer énfasis en estos, para dar paso a conceptos nuevos o conceptos que requieran de más detenimiento como lo son las ventas o los clientes y por lo tanto no será posible destinar demasiado tiempo a conceptos ya vistos en ciclos pasados y que se repiten constantemente como lo son los accesos a los menús o los privilegios de acceso.

Product BackLog Versión 2.0

El eje focal sobre el que gira SCRUM es el *Product BackLog* y a lo largo de estos ciclos probó su importancia además de ser un documento altamente dinámico.

Durante los ciclos pasados el *Product BackLog* ha sufrido muchos cambios, en la mayoría de los casos estos cambios se mencionan en el cierre de cada ciclo, y posteriormente se modificaba el *Product BackLog* lo que daba paso a una nueva versión de este documento.

En caso de que estos cambios no fueran mencionados en el cierre de algún ciclo, se fueron registrando para su futura documentación lo que también provoca cambios en el *Product BackLog* y genera una nueva versión. Es de esperarse que a lo largo de los ciclos restantes se tengan aún más cambios, esto únicamente confirma la importancia de este documento y el por qué desde un inicio se mencionó que este documento es uno de los más importantes dentro de esta metodología. De no existir este documento, las tareas realizadas en cada ciclo podrían perderse o en el peor de los casos podrían pasarse por alto y no ser detectadas hasta una fase avanzada del proyecto, lo que, en un ambiente empresarial daría como resultado pérdidas económicas y de recursos importantes.

Cambios en el Product Backlog:

- ³⁵/₁₇ **Ciclo 0:** Uno de los principales cambios sucedió en Ciclo 0, ya que saltó a la vista la necesidad de extender un poco mas este ciclo para poder explicar de manera más clara cómo poder generar un ambiente adecuado tanto para el desarrollo como para la ejecución.
- ³⁵/₁₇ **Ciclo 1:** En este ciclo no se registraron cambios importantes en el *Product BackLog*.
- ³⁵/₁₇ **Ciclo 2:** Otro de los cambios importantes pero que aún no han sido registrados, es la necesidad de agregar un par de tareas destinadas a la instalación de nuevos módulos, como en el caso de *SiFaDig* el módulo de facturación electrónica.
- ³⁵/₁₇ **Ciclo 3:** En este ciclo, notamos que hacía falta modificar algunos segmentos de ciclos anteriores, como lo es el que para poder asignar permisos y grupos, es necesario contar con la versión avanzada de la interfaz de OpenERP. Otro de los cambios fue el mencionar en el ciclo 0, la necesidad de agregar al momento de la instalación algunos módulos importantes, como el de clientes, productos, ventas y facturación.

Después de haber mencionado todos los cambios realizados en el *Product BackLog*, procedo a mostrar la versión más reciente de este.

Product Backlog: SiFaDig					+ alta = 1
					+ baja = 4
Id	Descripción de la Tarea	Estatus	Tamaño	Sprint	Prioridad
0	Crear Ambiente de desarrollo	Planificada	3	0	1
0.1	Instalación OpenERP	Planificada	2	0	1
0.1.1	Configuración PostgreSQL	Planificada	1	0	1
0.1.2	Instalación de Bibliotecas	Planificada	2	0	1
0.2	Configurar Ambiente de desarrollo	Planificada	4	0	1
0.2.1	Descargar Eclipse	Planificada	1	0	1
0.2.2	Configurar Eclipse	Planificada	3	0	1
0.3	Generar BD y Administrador	Planificada	1	1	1
1	Ingresar al Sistema	Planificada	1	1	1
2	Administración de Usuarios	Planificada	2	2	4
2.1	Alta de Usuario	Planificada	1	2	4
2.1.1	Asignación de Grupo	Planificada	1	2	4
2.1.2	Creación de Grupo	Planificada	1	2	4
2.1.3	Asignación de Permisos	Planificada	1	2	4
2.2	Modificación de Usuario	Planificada	1	3	4
2.2.1	Modificación de Permisos	Planificada	1	3	4
2.3	Baja de Usuario	Planificada	1	3	4
2.4	Gestión de Permisos	Planificada	2	3	4
3	Administración de Productos	Planificada	3	5	2
3.1	Alta de Productos	Planificada	1	5	3
3.2	Modificaciones de Productos	Planificada	1	5	3
3.3	Baja de Productos	Planificada	1	5	3
3.4	Inventario de Productos	Planificada	2	5	3
3.4.1	Actualizar Inventario al vender	Planificada	2	5	3
3.4.2	Renovar Inventario	Planificada	2	5	3
4	Administración de Clientes	Planificada	4	6	2
4.1	Alta de Clientes	Planificada	1	6	3
4.2	Modificación de Clientes	Planificada	1	6	3
4.3	Baja de Clientes	Planificada	1	6	3
4.4	Historial de Clientes	Planificada	1	6	4
5	Administración de Ventas y Facturas	Planificada	6	7	1
5.1.1	Generar preventa	Planificada	1	7	3
5.1.2	Modificar preventa	Planificada	1	7	3
5.1.3	Eliminar preventa	Planificada	1	7	3
5.2.1	Generar Venta	Planificada	1	7	3
5.2.2	Historial de Ventas	Planificada	1	7	4
6	Instalación módulo facturación electrónica	Planificada	1	8	3
6.1.1	Localizar código fuente	Planificada	1	8	3
6.1.2	Descargar código fuente (Bazaar)	Planificada	1	8	3
6.1.3	Instalación de módulos en OpenERP	Planificada	1	8	3
6.1.4	Generar Factura Electrónica	Planificada	1	8	3

Figura 5.30 Product BackLog v. 2.0

Cierre del Ciclo:

En este ciclo pudimos realizar una recopilación de los avances realizados y de los cambios que han tenido lugar en el desarrollo del proyecto *SiFaDig*, en particular dentro del *Product BackLog*. Sin embargo una de las lecciones que nos deja este ciclo es que el proceso de desarrollo de software hoy en día tiene que responder de una manera adecuada a las necesidades cambiantes y dinámicas de los productos que demandan las empresas y los usuarios.

Es por esto que el *Product BackLog* cumple una función importantísima dentro de SCRUM ya que sirve de esqueleto ordenado de las actividades que se realizan en el proyecto y es sumamente flexible debido a que lleva la cuenta del avance de la construcción e iteración de los ciclos del proyecto.

Ahora que ya se ha completado este ciclo, el siguiente paso es continuar con las tareas pendientes del *Product BackLog* y la siguiente tarea corresponde a la administración de productos. En el siguiente ciclo podremos ver algunas de las tareas y acciones que se pueden desempeñar en OpenERP concernientes a los clientes.

5.4 Ciclo 5: Administración de Productos

La administración de productos tiene que ver con todo lo relacionado con un producto, desde la cantidad existente en inventario y todos sus datos hasta el número de productos vendidos. Es por eso que es necesario contar con un modulo que se encargue de llevar a cabo estas tareas.

El punto más importante a tratar en este ciclo es el que respecta a cómo poder registrar productos dentro de OpenERP, para que posteriormente estos productos puedan ser vendidos a los clientes y se pueda generar una factura.

Existen una infinidad de tareas relacionadas con los productos y con su administración como lo son: el reabastecimiento automatizado, las reglas que nos indican cuando un producto está por agotarse, etc. Como ya es costumbre para este ciclo nos apoyaremos en el diagrama de casos de uso para poder mostrar cómo es que funciona la administración de productos dentro de OpenERP. Dentro del diagrama general de casos de uso, el caso de uso correspondiente es el caso de uso

3.- Administrar Productos.

- ³⁵₁₇ Alta de Productos.
- ³⁵₁₇ Modificación de de Productos.
- ³⁵₁₇ Baja de Productos
- ³⁵₁₇ Gestión de Permisos.
- ³⁵₁₇ Actualizar Inventario.
- ³⁵₁₇ Renovar Inventario.

En OpenERP la manera en la que se manejan tanto los productos como los servicios y los clientes comparten mucho en común, sin embargo una peculiaridad que distingue a los productos y servicios es la capacidad de poder generar categorías tanto de productos como de servicios, lo cual es muy útil para las empresas que manejan una gran cantidad de productos y/o servicios ya que le es muy complicado el poder organizarlos de manera adecuada. La administración de productos y servicios es muy similar, por lo que dentro de OpenERP la forma en la que podemos diferenciar un producto de un servicio es mediante el atributo “*Tipo de Producto*”, al dar de alta un producto podemos definir mediante el campo del mismo nombre si el producto en cuestión es un producto y pertenece al inventario o *stock*²⁶ de productos o si se trata de un servicio. En este ciclo únicamente mostraremos como administrar productos, pero se sobreentiende que la administración de servicios es similar.

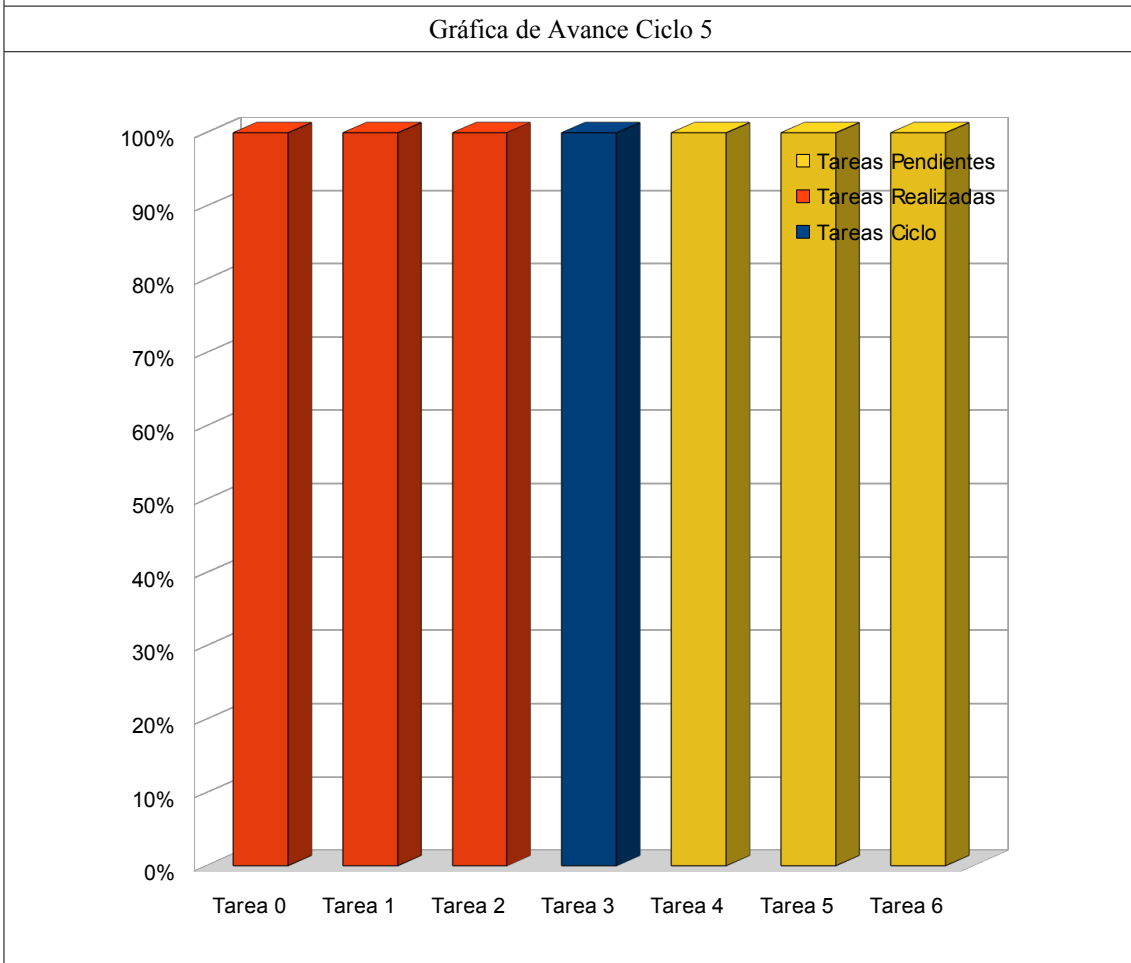
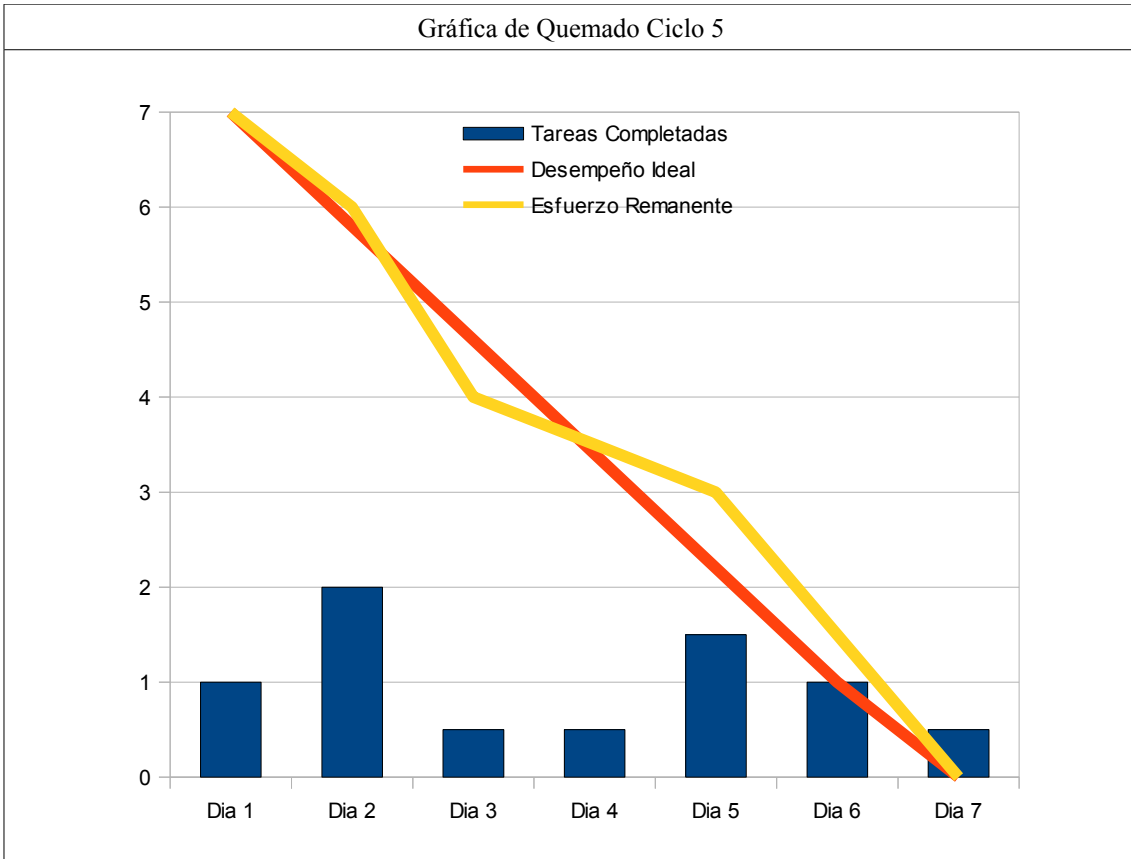
Por otro lado, OpenERP permite generar reglas que indiquen las cantidades de productos que han sido utilizadas al momento de ser vendidos y así generar una alerta para que ese producto en particular sea reabastecido en caso de ser necesario. OpenERP cuenta con un sin fin de funcionalidades relacionadas con la administración y organización de los productos, sin embargo para los fines de esta tesis únicamente nos interesa mostrar a los productos como un objeto necesario para generar una factura.

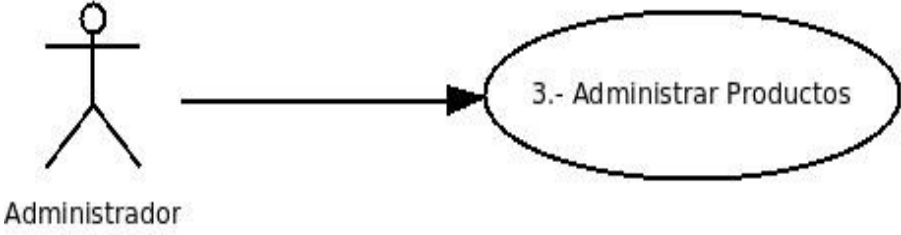
Como ya se ha mostrado en secciones anteriores, únicamente los usuarios con los privilegios adecuados pueden tener acceso al menú para la administración de productos. Como en ciclos pasados, para los ejemplos que se mostrarán en este ciclo se utilizará al usuario *Administrador*, pero como ya se mencionó anteriormente, cualquier usuario con los privilegios correctos (o el grupo adecuado) puede realizar las mismas operaciones que veremos en este ciclo. Es importante recalcar que dentro de OpenERP por lo general existe más de una forma de realizar una misma tarea, y en este ciclo únicamente analizaremos una de ellas. Además como ya hemos mencionado la finalidad de esta tesis es generar facturas electrónicas y para ello es más que suficiente con que podamos administrar los productos de una sola forma.

Durante este ciclo podremos ver algunos ejemplos que muestren como realizar las tareas que comprende este ciclo, pero para las funcionalidades avanzadas que ofrece OpenERP con respecto a la administración de productos se pide al lector consultar la documentación relacionada con este tema.

²⁶ *Todo lo referente a los bienes que una persona u organización tienen y que sirven para la realización de sus objetivos. Se puede traducir como inventario.*

Caso de Uso: Administrar Productos		Identificador: 3	
Ciclo: 5	<p>Objetivos Generales: Los productos son un elemento fundamental dentro de una empresa, sin embargo para los fines de este ciclo es más que suficiente con poder realizar las acciones básicas de la administración: alta, baja, cambio y algunas tareas adicionales que mostraremos dentro del desarrollo de este ciclo.</p> <p>3.- Administrar Productos.</p> <p>Tareas para este Ciclo:</p> <ul style="list-style-type: none"> ³⁵/₁₇ Alta de Productos. ³⁵/₁₇ Modificación de de Productos. ³⁵/₁₇ Baja de Productos ³⁵/₁₇ Actualizar Inventario al vender. ³⁵/₁₇ Renovar Inventario. 		
Fecha de Inicio: 09/05/2011			
Fecha de Terminó: 15/05/2011			
Tareas			
Id	Descripción de la Tarea	Tamaño	Prioridad
3.1	Alta de Productos	1	3
3.2	Modificación de Productos	1	3
3.3	Baja de Productos	1	3
3.4.1	Actualizar Inventario al vender	2	3
3.4.2	Renovar Inventario	2	3
Tareas			



Caso de Uso: Administrar Productos		Identificador: 3
 <p>Administrador</p>		
Descripción:	Este caso de uso nos sirve para mostrar cómo es que un usuario con los privilegios adecuados puede realizar las acciones básicas sobre la administración de los productos. Una vez que el usuario se encuentre en la pantalla inicial del sistema, puede ubicar el módulo de productos en el panel de ventas, ya que dentro del submenú de <i>Productos</i> , se encuentra las opciones <i>Productos por categoría</i> o <i>Productos</i> . Se selecciona la opción “Productos” para dar inicio a la administración de productos e iniciar con este caso de uso.	
Nivel:	Se puede decir que el nivel de este caso de uso es medio, ya que puede no es necesario administrar productos para poder generar facturas electrónicas. Sin embargo ya que se busca dar una solución completa a toda empresa que desee implementar OpenERP, es preferible generar la factura con los productos de la misma empresa.	
Actor Principal:	Administrador	
Parte Interesada:	Existen varias partes interesadas en este caso de uso, sin embargo al que más le debe importar es al administrador de ventas/almacén.	
Iniciador del Caso de uso:	Se inicia en el momento que es necesario dar de alta, baja o cambiar algún producto existen varias formas para acceder al control de los productos y con esto iniciar este caso de uso.	
Pre-condiciones:	Para poder administrar los productos y sus diversas acciones, es necesario logearse en el sistema con un usuario que tenga los privilegios adecuados, como el administrador, de no ser así no se podrá tener acceso al menú para administrar productos.	
Flujo de Desempeño Óptimo:		
Paso	Entrada	Respuesta del Sistema
1	El usuario selecciona el botón <i>Ventas</i> , que se encuentra en el menú inicial de OpenERP.	Dentro del panel principal, se muestran las opciones relacionadas con las ventas y una de ellas es <i>Productos</i> .
2	El usuario selecciona la opción <i>Productos</i> .	El botón correspondiente con los productos se expande y muestra las opciones: $\frac{35}{17}$ Productos por categoría. $\frac{35}{17}$ Productos.
3	El usuario selecciona la opción <i>Productos</i> dentro del menú que recién acaba de expandirse.	Se muestra una lista con los productos que ya han sido registrados dentro del sistema y algunas opciones para buscar dentro de estos. Además el sistema muestra el botón <i>Nuevo</i> , que nos permite registrar un nuevo producto.
4	El usuario presiona el botón <i>Nuevo</i> .	El sistema muestra una pantalla con múltiples pestañas, pero la más importante es la que tiene la

		<p>etiqueta <i>Información</i> y es la que contiene los campos necesarios que se tienen que capturar para dar registrar un producto. Los campos obligatorios son:</p> <ul style="list-style-type: none"> ³⁵/₁₇ Nombre del producto. ³⁵/₁₇ Precio del producto. ³⁵/₁₇ Categoría: seleccionar una categoría. ³⁵/₁₇ Unidad de medida por defecto: kg, mts, etc-. ³⁵/₁₇ Unidad de medida compra del producto: kg, mts, etc. <p>Las pestañas restantes contienen información relativa al producto y su Interacción con algunas otras áreas del sistema.</p>
5	El usuario captura los datos necesarios para dar de alta el producto y presiona <i>Guardar</i> .	En caso de haber llenado de manera satisfactoria el formulario correspondiente al producto, el sistema muestra la misma pantalla con los datos del producto pero los campos ya no son editables, lo que significa que el producto ya está almacenado en la base de datos. Además se habilita el botón <i>Eliminar</i> , con el cual podemos eliminar este producto del inventario.

Flujo Alternativo:

Paso	Entrada	Respuesta del Sistema
5.1	El usuario presiona el botón de <i>Editar</i> , dentro de la ventana con el resumen de los datos del producto.	El sistema permite editar los campos correspondientes a este producto para que el usuario pueda modificar los datos.
5.2	El usuario captura las modificaciones relacionadas al producto, pero siempre respetando los campos obligatorios y se presiona el botón <i>Guardar</i> .	Una vez más el sistema almacena los cambios correspondientes a este producto y vuelve a mostrar el formulario con los datos correspondientes al producto pero ahora estos campos ya no son editables, lo que significa que los cambios han sido almacenados de manera exitosa.
5.3	El usuario selecciona la opción correspondiente al <i>Almacén</i> , ya sea desde el menú superior o desde la pantalla inicial.	El sistema muestra del lado superior izquierdo el panel inicial correspondiente a la opción <i>Almacén</i> .
5.4	Dentro del panel inicial correspondiente al <i>Almacén</i> , el usuario expande la opción <i>Configuración</i> .	El sistema muestra las opciones: <ul style="list-style-type: none"> ³⁵/₁₇ Gestión de Almacenes. ³⁵/₁₇ Producto. ³⁵/₁₇ Envío ³⁵/₁₇ Abastecimientos automáticos.
5.5	El usuario expande la opción <i>Abastecimientos automáticos</i> .	El sistema muestra la opción <i>Reglas de stock mínimo</i> .
5.6	El usuario da <i>click</i> sobre la opción <i>Reglas de stock mínimo</i> .	Se muestra un formulario en el cual el usuario puede generar las reglas correspondientes a los <i>niveles mínimos</i> de los productos que se deseen, los campos obligatorios son: <ul style="list-style-type: none"> ³⁵/₁₇ Nombre: nombre de la regla. ³⁵/₁₇ Almacén: de cual almacén se trata. ³⁵/₁₇ Producto: productos que deseamos tener bajo monitoreo.

		³⁵ ₁₇ Ubicación: ubicación del inventario. ³⁵ ₁₇ UdM del producto: unidad de medida del producto. ³⁵ ₁₇ Cantidad min: cantidad mínima del producto. ³⁵ ₁₇ Cantidad max: cantidad máxima del producto. ³⁵ ₁₇ Cantidad múltiple: la cantidad abastecida será redondeada hacia arriba a éste múltiplo.
5.7	El usuario captura los campos necesarios para generar la regla y presiona <i>Guardar</i> .	El sistema muestra el formulario correspondiente a la regla recién generada, pero sus campos ya no son editables, lo que indica que la regla ya fue generada de manera exitosa. Además se muestran los botones <i>Editar</i> y <i>Eliminar</i>, con los que podemos editar y eliminar la regla que acabamos de generar.
	Post-condiciones:	Al verificar la lista de productos podemos constatar que el producto recién registrado ya se encuentra en esta lista, Además si se realizaron modificaciones al producto también las podemos comprobar seleccionando el producto en esta lista.
	Excepciones no controladas:	<ul style="list-style-type: none"> ⚠ No se puede registrar el producto debido a que no hay conexión con el servidor de la base de datos.
	<u>Consecuencias de Fallo:</u>	<ul style="list-style-type: none"> ⚠ De existir un fallo al momento registrar o modificar un producto, los cambios realizados no se almacenan en la base de datos y por lo tanto el inventario no sufre alteración.
	<u>Garantías Minina:</u>	<ul style="list-style-type: none"> ⚠ Se garantiza que de ocurrir algún problema al momento de registrar o modificar un producto, esto no afectara el desempeño del sistema por lo que basta con repetir el procedimiento para poder concluirlo. ⚠ De igual manera se garantiza que al momento de registrar o modificar un producto, no serán alterados los datos almacenados en la base de datos. ⚠ Se garantiza la persistencia de los datos del producto modificado.
	<u>Frecuencia:</u>	Este caso de uso tiene una frecuencia alta en un principio, y media con forme los productos son registrados en el sistema.
	<u>Supuestos:</u>	Se asume que el usuario encargado de administrar productos tenga conocimientos básicos de los procesos del inventario.

Comentarios:

Desempeño:

- ³⁵₁₇ Siempre que se registra un nuevo producto, se le asignan valores por defecto:
- ³⁵₁₇ Puede ser vendido: indica si este producto se mostrará en la lista de productos al momento de generar una venta
- ³⁵₁₇ Puede ser comprado: indica si este producto se mostrará en la lista de productos al momento de reabastecer el inventario.

Interfaz de Usuario:

- ³⁵₁₇ Es necesario haber habilitado el módulo de almacén para poder generar reglas y administrar los productos.

Casos Siguietes:

- ³⁵₁₇ En el siguiente caso de uso mostraremos como administrar los clientes, para que posteriormente podamos registrar una venta y generar una factura.

Tarea 3.1: Alta de Productos.

Como ya se mencionó en un inicio de este ciclo, los ejemplos que se muestra a continuación son realizados con el usuario *Administrador*, sin embargo cualquier usuario con los privilegios adecuados puede administrar los productos y las reglas relacionadas a estos. Una vez registrado como administrador el sistema mostrará la siguiente pantalla, la cual muestra todas las opciones disponibles para el *Administrador*.

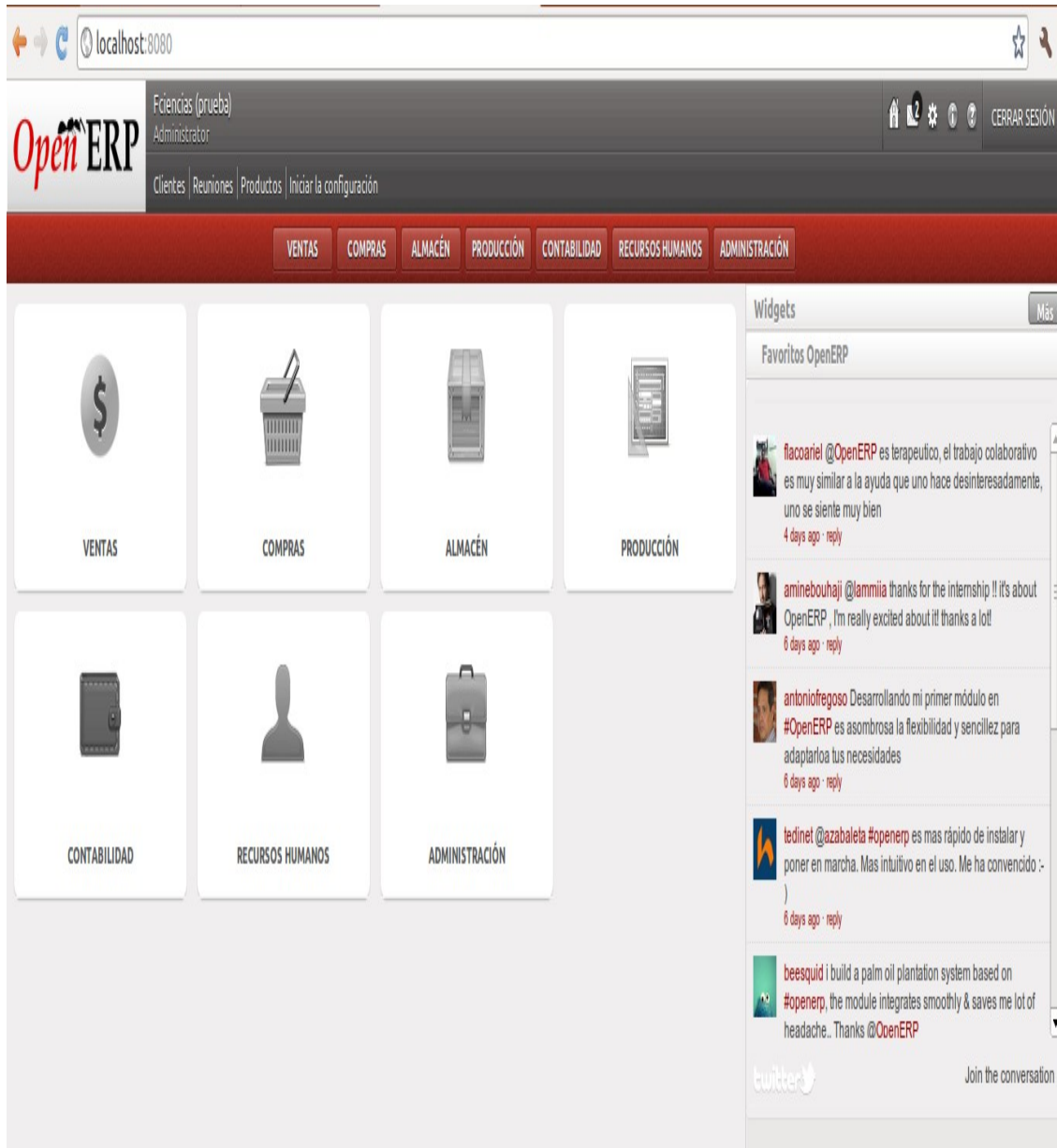


Figura 5.31 Panel principal Administrador

En la Figura 5.31 podemos ubicar el módulo correspondiente a las *Ventas* tanto en el menú superior como en el menú principal. Una vez en el menú de ventas podemos ubicar múltiples menús como: reuniones, llamadas telefónicas, etc. Sin embargo en este ciclo nos interesa registrar un nuevo producto, así que es necesario expandir la opción *Productos* y posteriormente seleccionar la opción *Productos*, como se muestra en la Figura 5.32

Implementando la Factura Digital mediante SCRUM

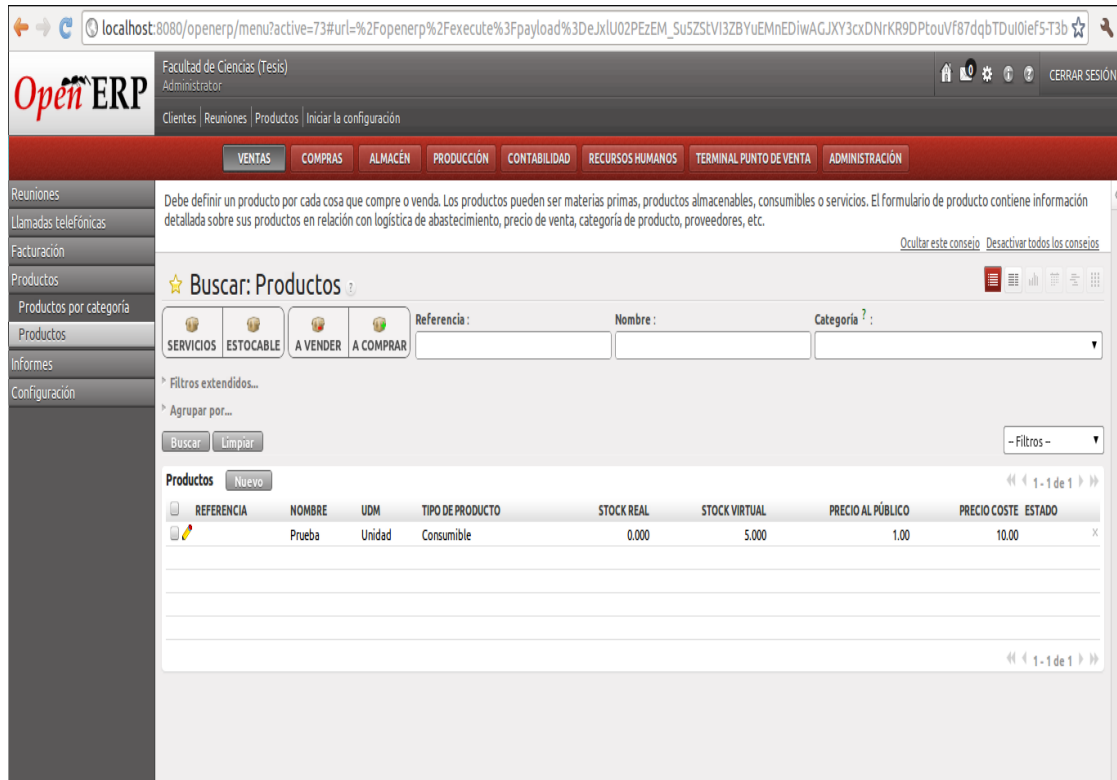


Figura 5.32 Panel de Productos

En esta pantalla podemos encontrar la lista de productos (y servicios) registrados en el sistema, también podemos buscar los productos mediante algunos campos como: referencia, nombre o categoría. Y lo más importante es que al seleccionar algún producto, lo podemos editar o eliminar, además de poder crear un nuevo producto presionando el botón *Nuevo*, lo que nos lleva a la siguiente pantalla la cual nos permite dar de alta o realizar modificaciones a los productos registrados en el sistema.

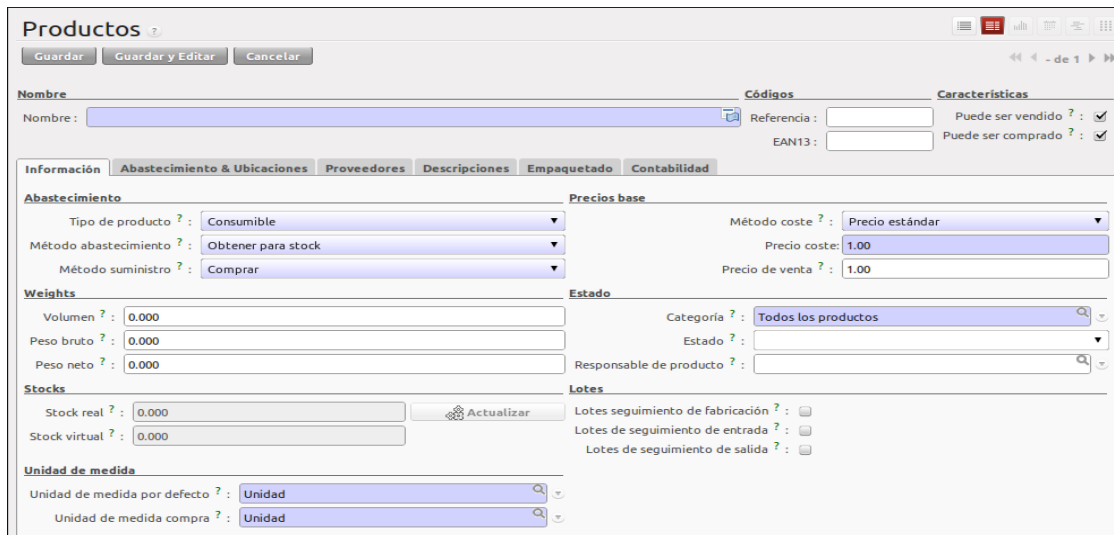


Figura 5.33

Como podemos apreciar en la Figura 5.33, debajo del campo *Nombre* se pueden ver una serie de pestañas que guardan información relacionada al producto que estamos dando de alta. Sin embargo, la pestaña que nos interesa es la que tiene la etiqueta “Información” y que es la que almacena los datos más importantes del producto. Por otro lado como ya es costumbre, los campos sombreados de color oscuro, son aquellos campos obligatorios, además cada campo contiene un pequeño símbolo de interrogación en el cual

podemos encontrar una pequeña descripción del campo. Uno de los campos más importantes al momento de dar de alta un producto es el campo *Tipo de Producto* y el cual puede tomar 3 valores:

- ³⁵/₁₇ Consumible: productos que podemos consumir de manera interna en nuestra empresa.
- ³⁵/₁₇ Almacenable: productos que podemos vender.
- ³⁵/₁₇ Servicio: servicios que ofrece la empresa.

En este caso en particular daremos de alta el producto como almacenable y antes de continuar vale la pena aclarar la diferencia entre stock real y stock virtual.

Stock Real: Es la cantidad real de productos que se tienen almacenados en el inventario.

Stock Virtual: Es la cantidad real de productos que se tienen almacenados en inventario, menos la cantidad de productos que aun no han sido vendidos pero que ya están registrados en algún pedido de ventas que aun no ha sido confirmado.

Procedemos a capturar los datos del producto recalcando que el campo *stock real*, debe tener algún valor mayor que cero para que podamos generar una *regla de stock mínimo* y pueda generarse una orden de compra en caso de que el producto caiga por debajo del mínimo.

Para asignar la cantidad de unidades de este producto, es necesario presionar el botón *Actualizar* e indicar la cantidad de unidades de este producto y el inventario en el que se encuentra almacenado. Al presionar el botón Actualizar se muestra la pantalla de la Figura 5.34 en la que registraremos una cantidad de 5.0 unidades de este producto, en el campo *Ubicación* seleccionaremos *Stock*, y el campo *Lote de producción* lo dejamos en blanco ya que no se manejarán lotes, de tal manera que se vea como se muestra en la siguiente imagen.

Figura 5.34 Pantalla cantidad de producto

Después de presionar el botón *Aplicar* en la pantalla anterior podremos ver una imagen similar a la que se muestra en la Figura 5.35. Es en la pantalla de la Figura 5.35 en la cual podemos dar de alta los productos que son necesarios para poder generar una factura electrónica, ya que de no tener productos registrados en el sistema, sería imposible poder realizar ventas que generen facturas electrónicas.

Figura 5.35 Panel de Alta de Productos

Ahora podemos presionar el botón *Guardar* para que los datos capturados sean registrados en el sistema y el producto quede registrado de manera correcta. Una vez que se presiona este botón el sistema deshabilita los campos y no permite que sean editados ya que los cambios fueron registrados en el sistema, como se muestra en la siguiente imagen.

Figura 5.36 Panel de Alta de Productos (solo lectura)

Tarea 3.2: Modificación de Productos

Dentro de la Figura 5.36 podemos encontrar el botón *Editar*, que nos permite regresar a la Figura 5.35 y editar los campos que deseemos del producto que hayamos registrado. Una vez realizados los cambios necesarios podemos volver a presionar el botón *Guardar* y los cambios serán almacenados posteriormente el sistema volverá a mostrar una imagen similar a la Figura 5.36.

Tarea 3.3: Baja de productos

Para dar de baja un producto, tenemos que ubicarnos en la pantalla de la Figura 5.32 y ya sea que eliminemos de manera unitaria un producto seleccionando la *x* que se muestra al final del resumen de cada producto o que se seleccionen los recuadros de selección de los productos que se quieran dar de baja y posteriormente se presione el botón *Eliminar*.

Una vez que se hayan eliminado algún producto por cualquiera de las 2 vías, se mostrara la siguiente advertencia, la cual nos permite confirmar o cancelar la decisión de eliminar un producto.

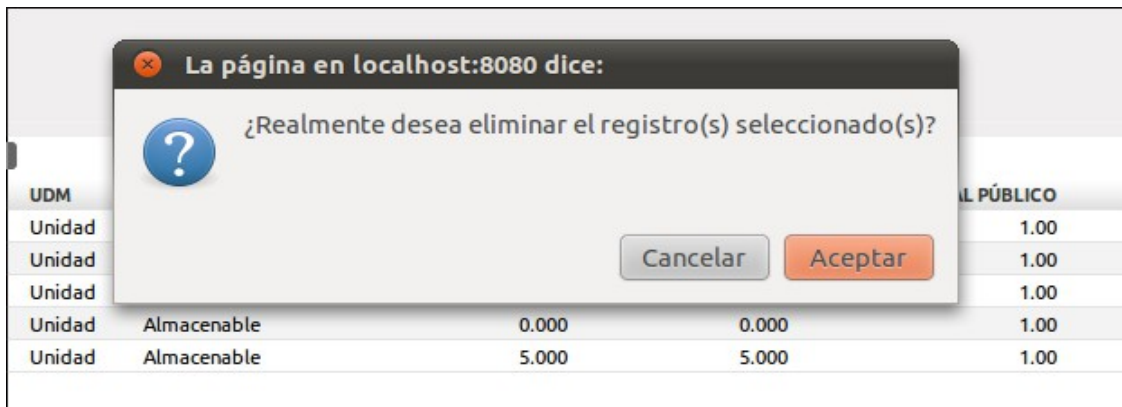


Figura 5.37 Ventana de confirmación de baja de productos

Y una vez que se haya presionado el botón *Aceptar*, el producto será eliminado²⁷ de la lista de productos y de la base de datos.

Tarea 3.4.1: Actualizar inventario al vender.

Aunque esta tarea está muy relacionada con los productos, es realizada de manera automática por el módulo de ventas, es decir que el único requisito para que esta tarea sea realizada es que el producto haya sido registrado como almacenable y que la cantidad de unidades de este producto sea mayor que cero, para que al momento de realizar una venta real (una venta que ya tenga una factura asociada) el/los productos facturados sean descontados del inventario. En el momento en el que lleguemos a las tareas correspondientes a las ventas podremos constatar que esta tarea sea llevada a cabo de manera exitosa.

Tarea 3.4.2: Renovar inventario

Esta tarea trata de generar una alerta para evitar que los productos almacenados en el Stock/Inventario rebasen un mínimo y así tratar de evitar que las ventas sean detenidas por la falta de productos. La tarea 3.4.2 está fuertemente relacionada con un objeto de OpenERP llamado **regla mínima de stock**.

Regla mínima de stock: es un objeto que nos permite asociar una cantidad mínima a un producto, de tal manera que sí al vender muchas unidades de este producto se alcanza un mínimo, el sistema genere una alerta para indicarnos que el producto está por agitarse y que es necesario reabastecerlo o en su defecto genera de manera automática la orden de compra para reabastecer el producto.

Para poder iniciar con esta tarea es necesario ubicarse en la pantalla de la Figura 5.31 y seleccionar el módulo correspondiente al *Almacén*. Una vez en este menú, es necesario expandir el menú *Configuración*, después hay que expandir el menú *Abastecimientos automáticos* hasta llegar a la opción *Reglas de stock mínimo*, como se muestra en la Figura 5.8

²⁷ Para poder eliminar algún producto es necesario verificar que este no esté relacionado con alguna tarea pendiente, ya que de ser así el producto no podrá ser eliminado.

Implementando la Factura Digital mediante SCRUM

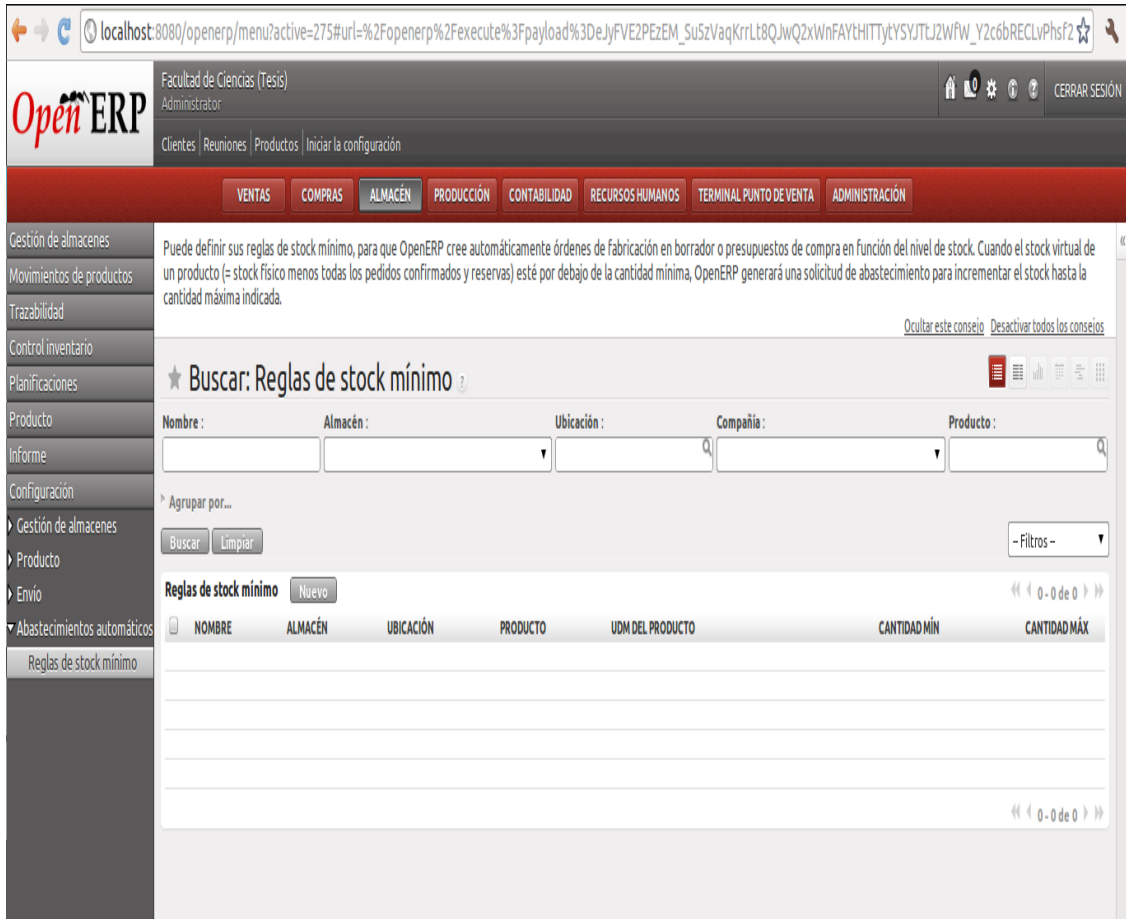


Figura 5.38 Panel de Almacén

Posteriormente presionamos el botón *Nuevo*, para poder generar una nueva regla y se mostrará la siguiente imagen, en la que podemos capturar los datos necesarios para asociar un mínimo de unidades a un producto almacenado.

Como ya es costumbre los campos sombreados de color azul son obligatorios, así que procedemos a capturar los siguientes datos:

- 35 17 Nombre: Regla3.4
- 35 17 Producto: Tarea3.1
- 35 17 Ubicación: Stock
- 35 17 Cantidad min: 3.0
- 35 17 Cantidad max: 10.0

Reglas de stock mínimo
Nombre: Regla3.4

Guardar Guardar y Editar Cancelar

Información general **Ubicaciones**

Nombre: Regla3.4 Almacén: Facultad de Ciencias

Producto: Tarea3.1 Ubicación: Stock

UdM del producto: Unidad

Reglas de cantidad **Misc.**

Cantidad mín: 3.00 Último abastecimiento:

Cantidad máx: 10.00 Activo:

Ctdad múltiple: 10

Órdenes de abastecimiento a procesar

Líneas de abastecimiento

FECHA PLANIFICADA	DOCUMENTO ORIGEN	PRODUCTO	CANTIDAD	UDM	ÚLTIMO ERROR

Figura 5.39 Panel Regla de stock mínimo

En el panel de la Figura 5.39 definimos el nombre de la regla, a que producto será asociada y cuál será el mínimo con el cual se lanza la alerta de stock mínimo. Ahora podemos presionar el botón *Guardar* y la regla ya ha sido registrada y almacenada, de tal manera que al llegar al mínimo marcado se genere una orden para el reabastecimiento del producto.

Reglas de stock mínimo
Nombre: Regla3.4

Nuevo Editar Duplicar Eliminar

Información general **Ubicaciones**

Nombre: Regla3.4 Almacén: Facultad de Ciencias

Producto: Tarea3.1 Ubicación: Stock

UdM del producto: Unidad

Reglas de cantidad **Misc.**

Cantidad mín: 3.00 Último abastecimiento:

Cantidad máx: 10.00 Activo:

Ctdad múltiple: 10

Órdenes de abastecimiento a procesar

Líneas de abastecimiento

FECHA PLANIFICADA	DOCUMENTO ORIGEN	PRODUCTO	CANTIDAD	UDM	ÚLTIMO ERROR

Figura 5.40 Panel Regla de stock mínimo (solo lectura)

En la figura 5.40 podemos apreciar la misma pantalla de la Figura 5.39 pero con los campos bloqueados para evitar su modificación.

Retrospectiva del Ciclo:

Con este ciclo nos pudimos familiarizar aun mas con los procesos y elementos de OpenERP, ya debe ser algo común para estas alturas los menús de búsqueda o las listas asociadas a los módulos. Además al concluir este ciclo nos dimos cuenta que las acciones y procesos asociados a los productos/servicios son muchas y muy útiles, es necesario con dominar las funcionalidades básicas para poder continuar con el desarrollo del proyecto.

En el próximo ciclo podremos analizar el módulo relacionado con los clientes y así poder finalmente, generar una venta y posteriormente una factura electrónica.

5.5 Ciclo 6: Administración de clientes

Ya que hemos revisado como administrar los productos, la mecánica bajo la cual se administran los clientes es similar, es decir, la forma en la que se dan de alta, baja y se realizan cambios a los clientes es casi idéntica a la que se maneja con los productos, salvo algunas pequeñas diferencias. Se tomó la decisión de iniciar con la administración de productos antes que la de los clientes debido a que la administración de productos es un tanto mas sencilla que la administración de clientes y al haber revisado anteriormente la administración de productos, esto podría servir como una base para iniciar el ciclo correspondiente a la gestión de clientes.

De igual manera que para el ciclo anterior, en este ciclo existen múltiples relaciones entre los usuarios, clientes, productos, servicios y demás elementos que integran *SiFaDig* y es por este motivo que continuaremos haciendo uso de la técnica de casos de uso para ejemplificar de manera más clara como es que se lleva a cabo el proceso de la administración de clientes, que corresponde al 6o ciclo de SCRUM y dentro del diagrama de casos de uso general es el caso **administrar clientes**.

- ³⁵₁₇ Alta de clientes
- ³⁵₁₇ Modificación de clientes.
- ³⁵₁₇ Baja de Productos

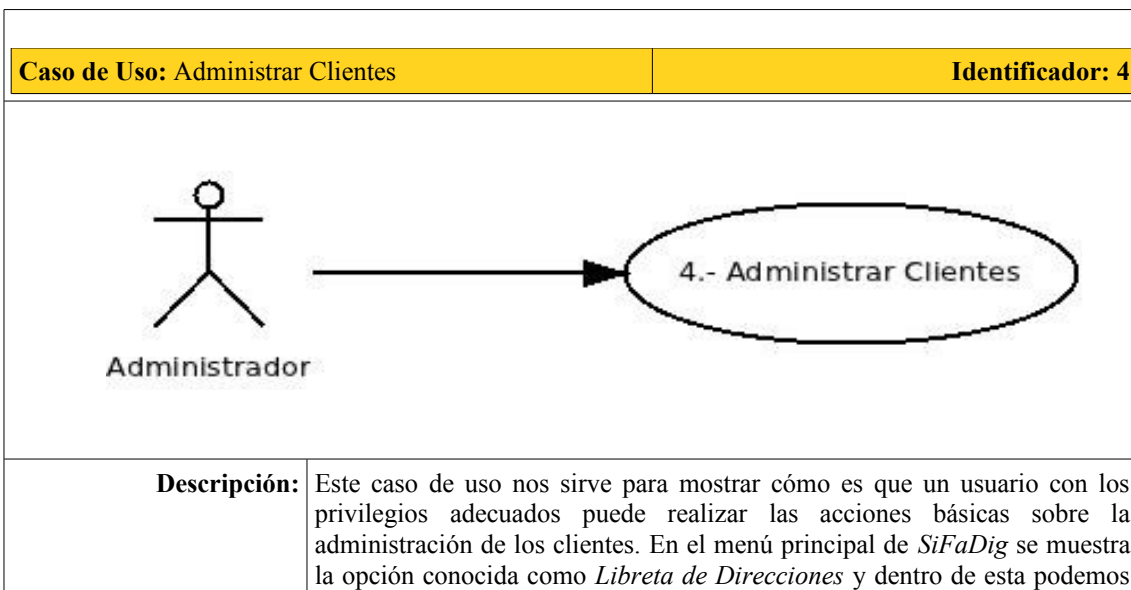
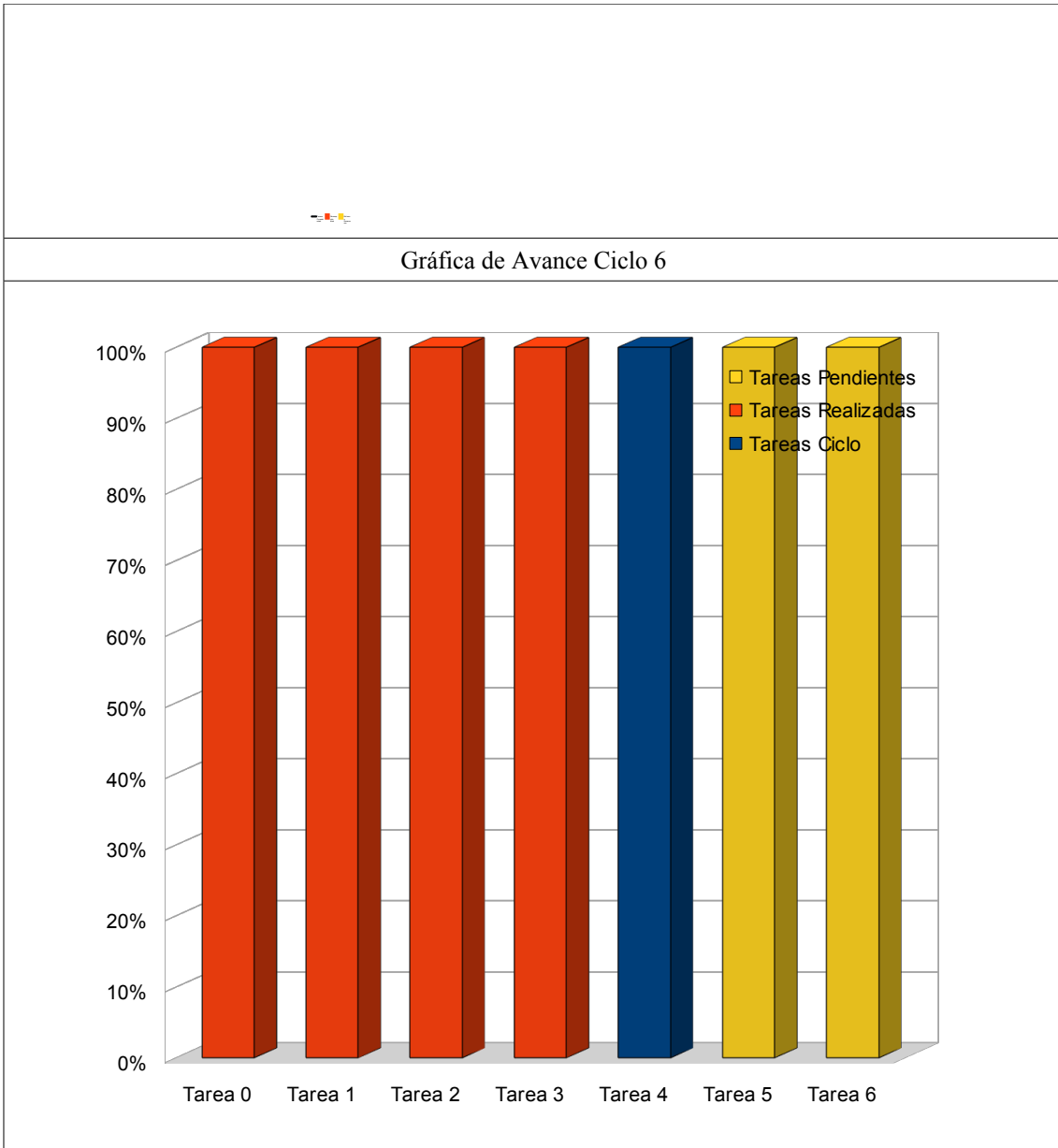
De manera similar que en el caso de los productos en el que podíamos distinguir entre productos y servicios, en el caso de los clientes también podremos generar 2 grupos:

- ³⁵₁₇ Clientes: clientes a los cuales se les puede vender productos y servicios.
- ³⁵₁₇ Proveedores: proveedores que se encargan de reabastecer el inventario.

Dependerá del grupo en que el que se asigne una categoría a cada uno de los *Clientes* para poder hacer uso del mismo, en distintos procesos dentro de la aplicación.

La administración de proveedores es muy similar a la administración de clientes y no entra en el alcance de esta tesis debido a que únicamente nos enfocaremos en el proceso que nos lleva a obtener una factura electrónica y para dicho proceso es suficiente con saber cómo administrar a los clientes.

<p>Ciclo: 6</p>	<p>Objetivos Generales: La administración de clientes es pieza fundamental en cualquier ERP, para este ciclo se pretende dar un recorrido a todo el proceso que involucra la alta, baja y modificación de clientes para que de esta manera la información almacenada para cada cliente, pueda ser utilizada al momento de generar una factura electrónica.</p> <p>4.- Administrar Clientes</p> <p>Tareas para este Ciclo:</p> <p>³⁵₁₇ Alta de Clientes. ³⁵₁₇ Modificación de de Clientes. ³⁵₁₇ Baja de Clientes.</p>		
<p>Fecha de Inicio: 09/05/2011</p>			
<p>Fecha de Terminó: 15/05/2011</p>			
<p>Tareas</p>			
<p>Id</p>	<p>Descripción de la Tarea</p>	<p>Tamaño</p>	<p>Prioridad</p>
<p>4.1</p>	<p>Alta de Clientes</p>	<p>1</p>	<p>3</p>
<p>4.2</p>	<p>Modificación de Clientes</p>	<p>2</p>	<p>3</p>
<p>4.3</p>	<p>Baja de Clientes</p>	<p>1</p>	<p>3</p>
<p>Gráfica de Quemado Ciclo 6</p>			



	ubicar el módulo de <i>Cientes</i> con el cual da inicio el análisis de este ciclo.
Nivel:	El nivel de este ciclo es medio, dado que no es posible generar una factura sin los datos del cliente que está facturando. A su vez requiere de un conocimiento avanzado del sistema para poder administrar de manera adecuada a los clientes y sus procesos
Actor Principal:	Administrador.
Parte Interesada:	Existen varias partes interesadas en este caso de uso, sin embargo al que más le debe importar es al administrador de ventas/almacén.
Iniciador del Caso de uso:	Se inicia en el momento que es necesario dar de alta, baja o cambiar algún cliente/proveedor, existen varias formas para acceder al control de los clientes y con esto iniciar este caso de uso.
Pre-condiciones:	Para poder administrar los clientes y sus diversas acciones, es necesario ingresar en el sistema con un usuario que tenga los privilegios adecuados, como el administrador, de no ser así no se podrá tener acceso al menú para administrar clientes.

Flujo de Desempeño Óptimo:

Paso	Entrada	Respuesta del Sistema
1	El usuario selecciona el botón <i>Libreta de Direcciones</i> , que se encuentra en el menú inicial de OpenERP.	Dentro del panel principal, se despliega una nueva opción en la que se puede leer <i>Cientes</i> .
2	El usuario selecciona la opción <i>Cientes</i> .	La vista ubicada del lado derecho del menú principal cambia y ahora muestra los clientes y proveedores registrados y las opciones relacionadas a los mismos, incluyendo la opción <i>Nuevo</i> .
3	El usuario presiona el botón <i>Nuevo (Cliente)</i> .	El sistema muestra una pantalla con múltiples pestañas, pero la más importante es la que tiene la etiqueta <i>General</i> y es la que contiene los campos necesarios que se tienen que capturar para registrar un nuevo cliente. El único campo obligatorio para poder registrar un nuevo cliente es el nombre del mismo. Dentro de la pestaña <i>General</i> se muestran 2 recuadros de selección mediante los cuales podemos indicar si estamos registrando a un cliente o a un proveedor. Las pestañas restantes contienen información relativa al producto y su Interacción con algunas otras áreas del sistema.
4	Una vez capturados los datos necesarios el usuario presiona el botón de <i>Guardar</i> .	En caso de haber llenado de manera satisfactoria el formulario correspondiente al cliente, el sistema muestra la misma pantalla con los datos del producto pero los campos ya no son editables, lo que significa que el cliente ya está almacenado en la base de datos. Además se habilita el botón <i>Eliminar</i> , con el cual podemos eliminar este cliente de la base de datos.

Flujo Alternativo:

Paso	Entrada	Respuesta del Sistema
------	---------	-----------------------

5.1	El usuario presiona el botón de <i>Editar</i> , dentro de la ventana con el resumen de los datos del cliente.	El sistema permite modificar los campos correspondientes a este cliente para que el usuario pueda modificar los datos.
5.2	El usuario captura las modificaciones relacionadas al cliente, pero siempre respetando los campos obligatorios y se presiona el botón <i>Guardar</i> .	Una vez más el sistema almacena los cambios correspondientes a este cliente y vuelve a mostrar el formulario con los datos correspondientes al cliente pero ahora estos campos ya no son editables, lo que significa que los cambios han sido almacenados de manera exitosa.
5.3	El usuario selecciona la opción correspondiente al <i>Historial</i> .	El sistema despliega una pequeña ventana en la que se puede ver el historial de modificaciones al cliente en cuestión, se muestran campos como: ³⁵ ₁₇ Creado por el usuario. ³⁵ ₁₇ Fecha de creación. ³⁵ ₁₇ Última modificación realizada por. ³⁵ ₁₇ Propietario. ³⁵ ₁₇ Grupo propietario módulo interno.
Post-condiciones:	Podemos revisar el listado de clientes para poder constatar que el cliente (proveedor) y la información relacionada con éste se encuentran almacenadas de manera correcta.	
Excepciones no controladas:	³⁵ ₁₇	No se puede registrar el cliente debido a que no hay conexión con el servidor de la base de datos.
Consecuencias de Fallo:	³⁵ ₁₇	De existir un fallo al momento registrar o modificar un cliente, los cambios realizados no se almacenan en la base de datos.
Garantías Minina:	³⁵ ₁₇	Se garantiza que de ocurrir algún problema al momento de registrar o modificar un cliente, esto no afectará el desempeño del sistema por lo que basta con repetir el procedimiento para poder concluirlo. ³⁵ ₁₇ De igual manera se garantiza que al momento de registrar o modificar un cliente, no serán alterados los datos almacenados en la base de datos. ³⁵ ₁₇ Se garantiza la persistencia de los datos del cliente modificado.
Frecuencia:	Este caso de uso tiene una frecuencia alta en un principio y media conforme los clientes son registrados en el sistema.	
Supuestos:	Se asume que el usuario encargado de administrar productos tenga conocimientos básicos de administración de clientes y proveedores.	

Comentarios:

Desempeño:

- ³⁵₁₇ Siempre que se registra un nuevo cliente, se puede dividir en 2 grupos:
 - Cliente: funciona como un cliente común y corriente y se le pueden agendar citas, ventas en automático, correos e incluso envío de mensajes vía celular.
 - Proveedor: se puede registrar a un cliente con la bandera de proveedor para que pueda ser parte de otros procesos como el del almacén, además de que un mismo cliente puede jugar ambos papeles, cliente y proveedor.

Interfaz de Usuario:

- ³⁵₁₇ Es necesario haber habilitado el módulo de proveedores y clientes para poder administrar de forma completa a los clientes.

Casos Siguietes:

- ³⁵₁₇ Una vez que ya revisamos tanto a los clientes como a los productos los casos de uso siguientes son:
- Ventas: es el modulo encargado de registrar las ventas y determinar si es posible generar una factura.
 - Facturación: en este modulo se generan las facturas electrónicas que puede emitir el sistema.

Tarea 4.1: Alta de Clientes.

Como ya se mencionó al inicio de este ciclo, los ejemplos que se muestra a continuación son realizados por el usuario administrador, sin embargo cualquier usuario con los privilegios adecuados puede administrar los clientes y sus características. La Figura 5.41 muestra el panel principal del administrador.

Implementando la Factura Digital mediante SCRUM

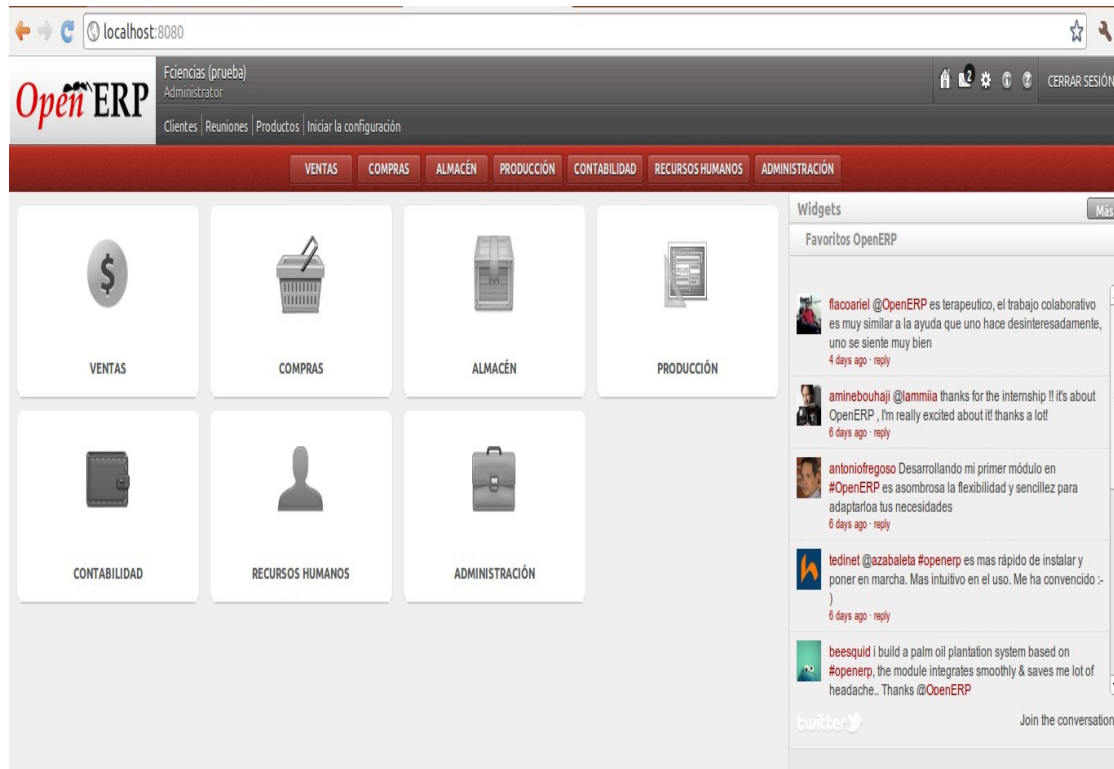


Figura 5.41 Panel principal Administrador

La Figura 5.41 es la pantalla principal que muestra el sistema siempre que el usuario ingresa y dependiendo de los módulos que tenga habilitados se mostrarán diferentes opciones. Para el caso de los clientes es necesario seleccionar cualquiera de las opciones que se muestran en pantalla para que se despliegue el menú principal del lado izquierdo de la pantalla, en el cual encontraremos la opción para la administración de clientes/proveedores.

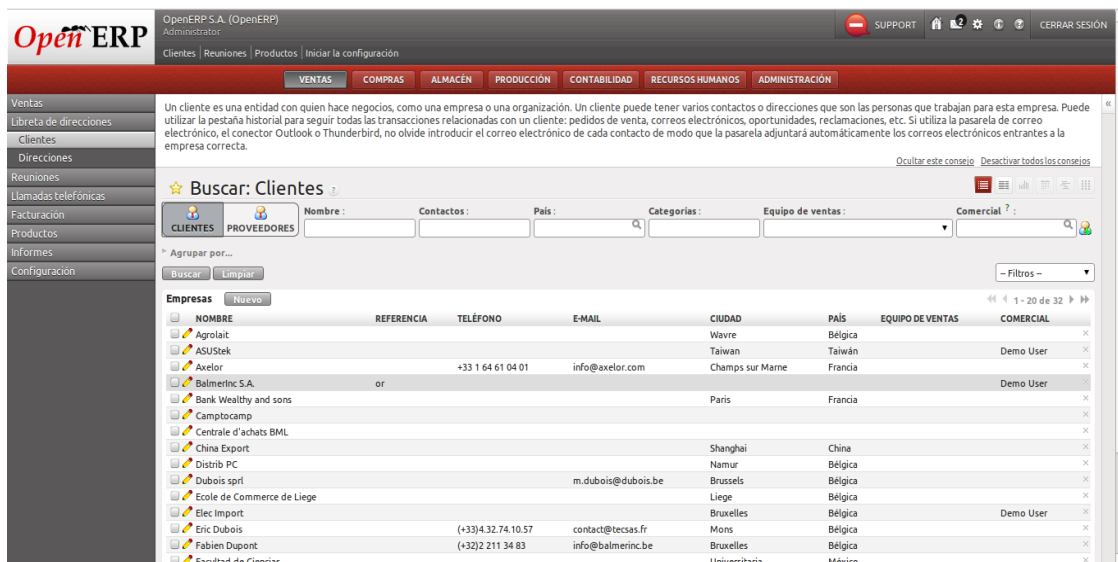


Figura 5.42 Panel Ventas

En esta Figura 5.42 podemos encontrar la lista de clientes (y proveedores) registrados en el sistema, también podemos buscar los productos mediante algunos campos como: nombre, contactos, país. Y lo más importante es que al seleccionar algún cliente, lo podemos editar o eliminar, además de poder crear

un nuevo cliente presionando el botón *Nuevo*, lo que nos lleva a la siguiente pantalla, la cual permite dar de alta un cliente, así como modificar sus datos en caso de ser necesario.

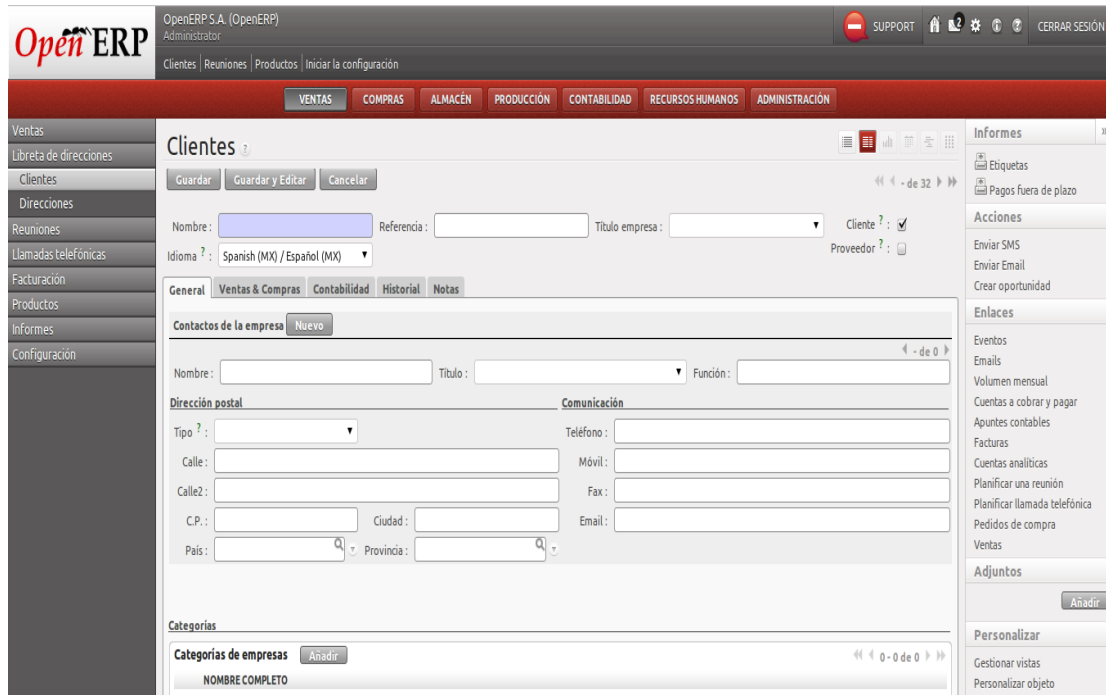


Figura 5.43 Panel de Clientes

De igual manera que en el caso de los productos, debajo del campo destinado para el *Nombre* podemos encontrar varias pestañas que despliegan mas campos para la gestión de nuestros clientes/proveedores, sin embargo las pestañas que en este momento nos deben importar más son las pestañas relacionadas con los datos *Generales* del cliente y también la pestaña relacionada con su *Historial*. Todas las pestañas son importantes y en determinado momento juegan un papel importante en los procesos que sigue el sistema, pero el único campo obligatorio para poder continuar con este caso de uso es el del nombre del cliente.

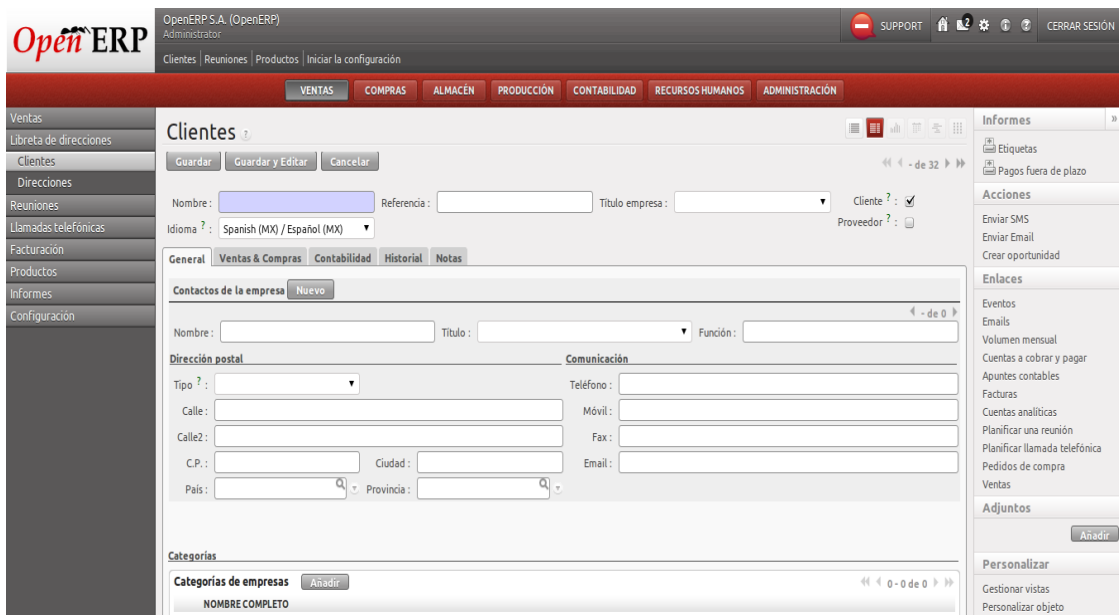


Figura 5.44 Panel de Proveedores

Así que procedemos a capturarlo y presionar el botón de *Guardar*. Una vez que el cliente ha sido registrado de manera exitosa podemos apreciar como en la gran mayoría de los casos que después de

haber presionado el botón de *Guardar*, los campos que antes podían ser editados dejan de serlo y eso nos indica que el cliente ha sido capturado de manera exitosa como se puede apreciar en la siguiente Figura.

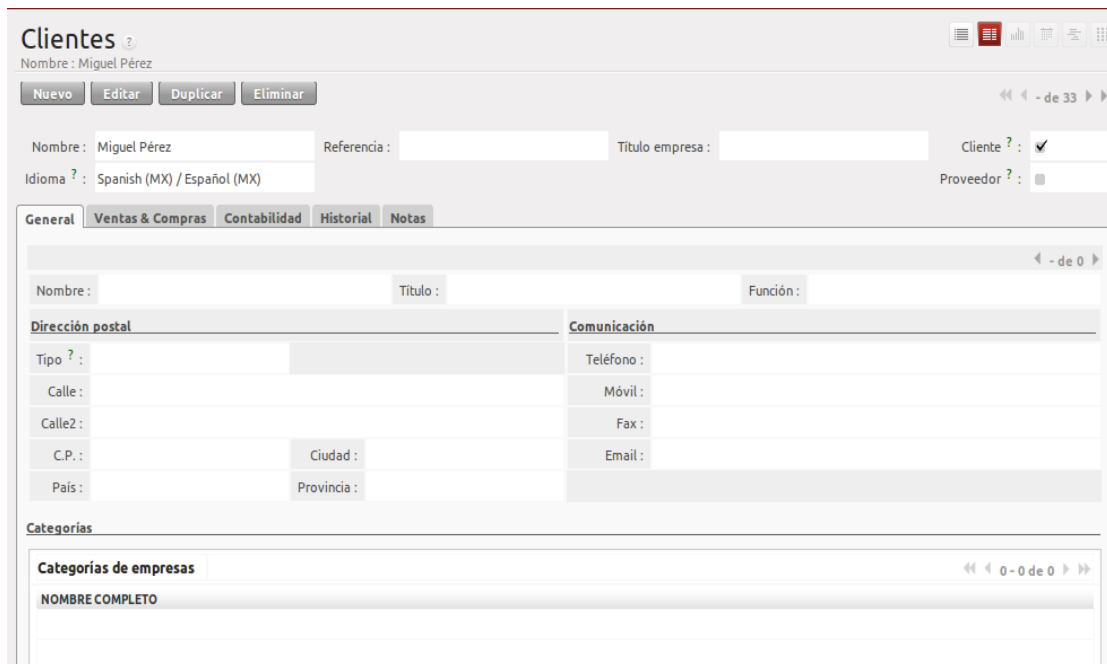


Figura 5.45 Panel de captura de datos del cliente

Tarea 4.2: Modificación de Clientes

En la Figura 5.45 se puede apreciar un botón en el que se lee *Editar*, mismo que al presionarlo vuelve a hacer que los campos retomen su modalidad de “*editables*”, lo que nos permite editar la información de cualquier cliente que se desee. Al terminar de editar el cliente en cuestión basta con repetir el proceso para finalizar la captura de un nuevo cliente y ya con eso la tarea 4.2 queda concluida, para corroborar lo anterior basta con revisar el listado de clientes y buscar el cliente que fue modificado de tal manera que podamos apreciar los cambios realizados al mismo.

Tarea 3.3: Baja de Clientes

La baja de un cliente es en realidad una tarea muy sencilla, basta con volver a la pantalla que se muestra en la Figura 5.42 y seleccionar un cliente del listado, de tal manera que al hacer esto se palomea el recuadro de selección correspondiente a este cliente y también se habilita el botón para eliminar al mismo, como se muestra en la siguiente Figura 5.46. En esta pantalla podemos buscar clientes por varios campos, así



Figura 5.46 Panel de búsqueda de clientes

Ya con el botón de *Eliminar* habilitado es suficiente con presionarlo para que se muestre una ventana emergente en la que se nos pide confirmar que se desea eliminar al cliente seleccionado.

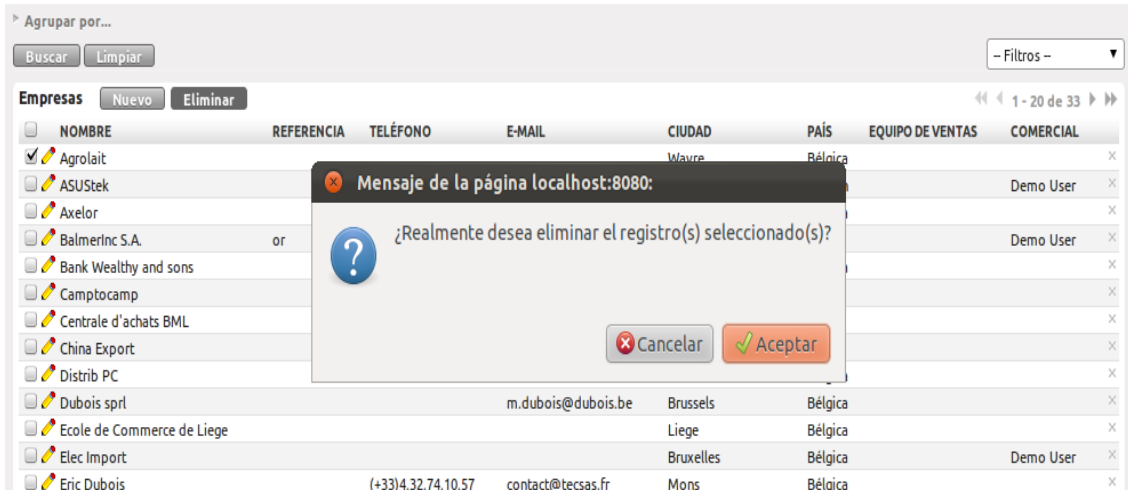


Figura 5.47 Ventana de confirmación de baja de productos

Si se presiona *Aceptar* el sistema tarda unos segundos en procesar la solicitud y validar que no haya procesos asociados a este cliente en particular. En caso de no tener procesos pendientes el cliente es eliminado y ya no es accesible desde el listado de clientes.

Retrospectiva del Ciclo:

La administración de clientes nos da un panorama más completo de cómo funcionan los procesos internos del sistema y de cómo interactúan los diferentes elementos que lo integran. Al mismo tiempo que nos recuerda en gran medida la administración de productos.

Para el próximo ciclo entraremos de lleno al proceso que es necesario seguir para poder generar una preventiva y posteriormente una venta que puede ser facturada.

5.6 Ciclo 7: Administración de Ventas

En este ciclo se tratará todo lo relacionado con las “ventas”, desde como capturar los datos de una venta que puede o no ser cancelada, hasta la modificación de los datos relacionados a una venta y como poder cancelar una venta.

Con todos los elementos que hemos analizado en ciclos pasados, tales como la administración de productos y de clientes, podemos introducir el concepto de ventas el cual sienta las bases para poder generar facturas electrónicas. En este ciclo trataremos el tema de las “preventas”, que son aquellos datos capturados por el usuario y que en un futuro se convertirán en una venta, pero que aun pueden cancelarse o modificarse sin que se haya emitido factura alguna. Este concepto es muy útil ya que en la mayoría de las veces las preventas son modificadas para agregar o quitar productos o incluso para ser canceladas y si no existiera este concepto la modificación o cancelación de una venta sería un proceso más complicado o incluso podría generar problemas a nivel contable o fiscal. Las preventas son un elemento inherente de OpenERP, de tal manera que una venta y a su vez una factura no puede ser generada si no ha pasado por el proceso de preventa.

Como en todos los ciclos pasados en este ciclo haremos referencia a ciclos anteriores, ya que para poder generar una venta (preventa) es necesario contar con un cliente, un producto o servicio y un procedimiento mediante el cual podamos asignar una cantidad de productos y/o servicios a un cliente en particular. Estas tareas pueden a su vez subdividirse en otras muchas y la lista de estas tareas puede verse de la siguiente manera:

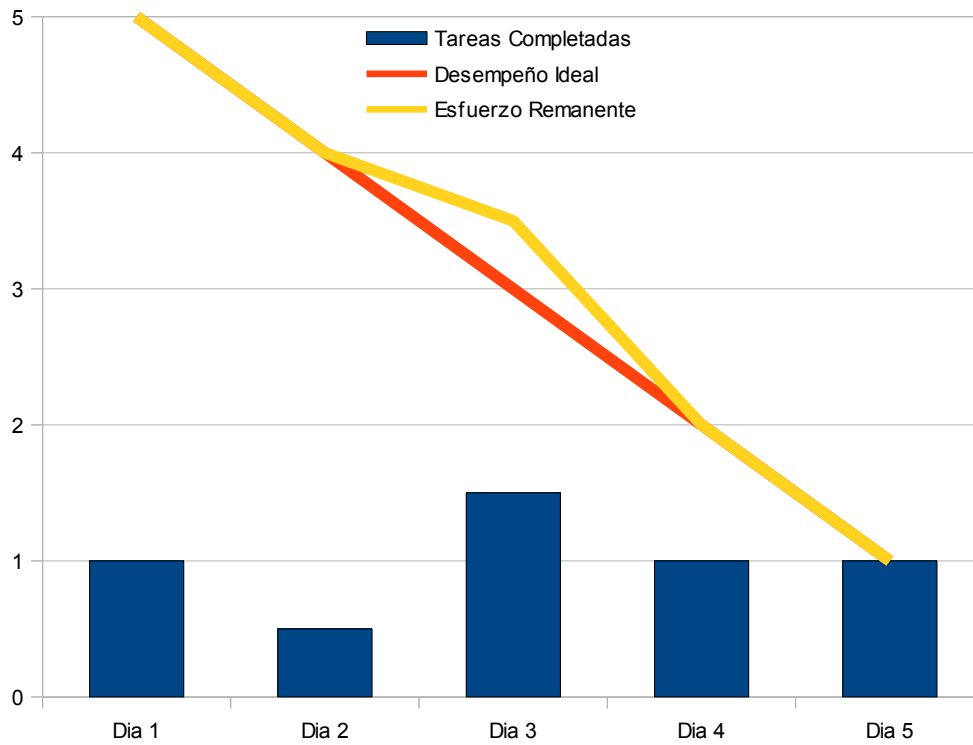
- ³⁵/₁₇ Generar preventa.
- ³⁵/₁₇ Modificar preventa.
- ³⁵/₁₇ Eliminar preventa
- ³⁵/₁₇ Generar venta.
- ³⁵/₁₇ Historial de ventas.

La manera en la que se genera una venta es muy similar a la forma en la que se maneja este recurso en cualquier ERP. Lo primero que se necesita para generar una venta es un borrador en el que se indique que se tiene la intención de adquirir un producto y/o servicio a este borrador se le conoce como **preventa**. En una preventa el usuario captura el/los artículos que desea adquirir y una vez que el cliente acepta el precio calculado en la preventa y los artículos que comprende, entonces el usuario puede tomar esta preventa y asignarle el carácter de una *venta*, con la cual posteriormente podremos generar una factura asociada a la misma y a su vez una factura electrónica.

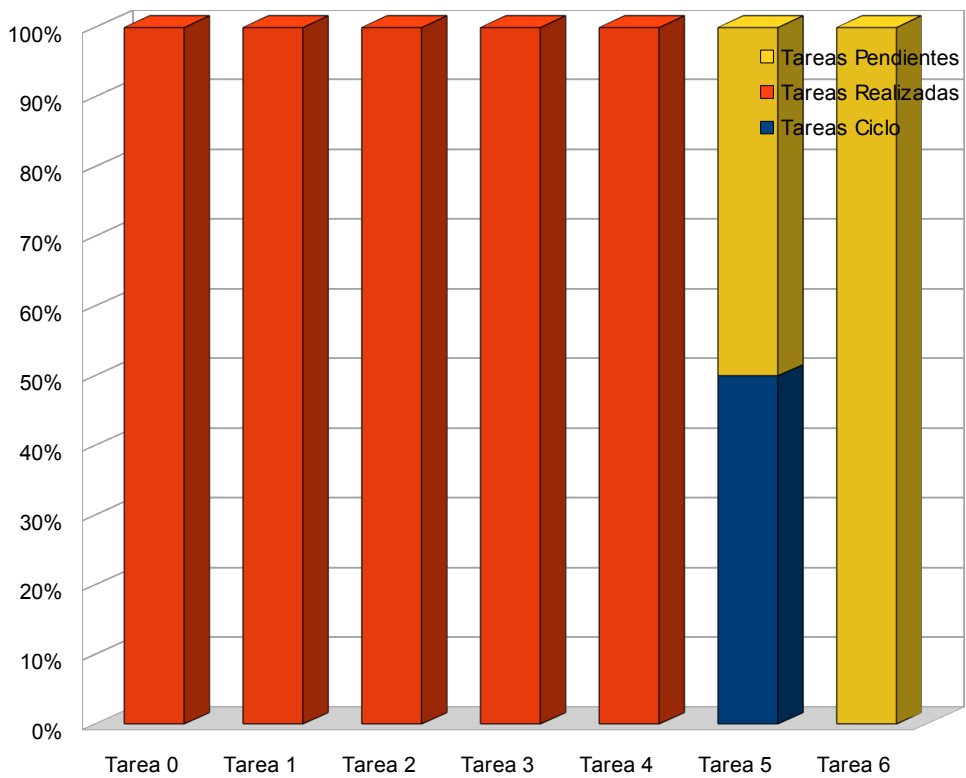
Para poder generar una preventa y posteriormente una factura es necesario que el usuario se encuentre en un grupo que tenga acceso a este módulo, o en su defecto que se cree un nuevo grupo y que sea asociado al usuario en cuestión. Este procedimiento ya fue analizado en ciclos pasados, por lo que en este ciclo se hará uso del usuario administrador para llevar a cabo las labores necesarias para poder generar una venta.

Sprint Backlog: SiFaDig		Identificador: 5	
Ciclo: 7	<p>Objetivos Generales: Para poder generar una factura electrónica es necesario que antes se haya generado una venta para tal fin se destinó este ciclo en el cual podremos ver de manera más detallada cual es el proceso necesario para capturar los datos de una venta, así como ver el historial de ventas.</p> <p>5.- Administrar Ventas</p> <p>Tareas para este Ciclo:</p> <ul style="list-style-type: none"> ³⁵/₁₇ Generar preventa. ³⁵/₁₇ Modificar preventa. ³⁵/₁₇ Eliminar preventa. ³⁵/₁₇ Generar venta. ³⁵/₁₇ Historial de ventas. 		
Fecha de Inicio: 16/05/2011			
Fecha de Terminó: 23/05/2011			
Tareas			
Id	Descripción de la Tarea	Tamaño	Prioridad
5.1.1	Generar preventa	1	3
5.1.2	Modificar preventa	1	3
5.1.3	Eliminar preventa	1	3
5.2.1	Generar venta	1	3
5.2.2	Historial de ventas	1	4

Gráfica de Quemado Ciclo 7



Gráfica de Avance Ciclo 7



Caso de Uso: Administrar Clientes		Identificador: 5
Descripción:	Este caso de uso se enfoca en atender las necesidades del usuario cuando desea generar una preventa que después se convertirá en una venta y posteriormente en una factura. A su vez se muestra la manera en la que se administran las ventas y de igual forma cómo es posible acceder al historial de ventas. Este caso inicia cuando el usuario ubica en la pantalla inicial del sistema la opción de <i>Ventas</i> y posteriormente selecciona la opción <i>Pedidos de Ventas</i> .	
Nivel:	El nivel de este ciclo es alto, dado que una preventa es un paso anterior para poder llegar a generar una factura electrónica y de no pasar por este procedimiento no se podrá generar una factura electrónica.	
Actor Principal:	Administrador	
Parte Interesada:	El grupo principal interesado en este caso es el correspondiente al grupo asociado a la gestión de ventas y de igual manera al cliente que forma parte de la venta.	
Iniciador del Caso de uso:	Se inicia en el momento que se tiene que generar una preventa. Existen varias formas para acceder al menú de ventas y con esto iniciar este caso de uso.	
Pre-condiciones:	Para poder administrar las ventas y sus diversas acciones, es necesario ingresar en el sistema con un usuario que tenga los privilegios adecuados, como el administrador, de no ser así no se podrá tener acceso al menú para administrar productos.	
Flujo de Desempeño Óptimo:		
Paso	Entrada	Respuesta del Sistema
1	El usuario selecciona el botón <i>Ventas</i> , que se encuentra en el menú inicial de OpenERP.	Dentro del panel principal, se despliega una nueva opción en la que se puede leer <i>Pedidos de Ventas</i> .
2	El usuario selecciona la opción <i>Pedidos de Ventas</i> .	La vista ubicada del lado derecho del menú principal cambia y ahora muestran las preventas que han sido generadas hasta ese momento. Así como también se muestra un botón en el que se lee <i>Nuevo</i> .
3	El usuario presiona el botón <i>Nuevo (Preventa)</i> .	El sistema muestra una pantalla con múltiples pestañas, pero la más importante es la que tiene la etiqueta <i>Pedido de Venta</i> y es la que contiene los campos necesarios que se tienen que capturar para una nueva preventa. A diferencia de otros módulos,

		<p>para poder generar una preventa, es obligatorio capturar los siguientes campos:</p> <ul style="list-style-type: none"> ³⁵/₁₇ Referencia del pedido (autogenerado). ³⁵/₁₇ Fecha (se puede seleccionar de un calendario). ³⁵/₁₇ Tienda (en caso de gestionar mas de una sucursal). ³⁵/₁₇ Cliente. ³⁵/₁₇ Dirección de Factura (dirección fiscal). ³⁵/₁₇ Tarifa (en caso de aplicar tarifa especial). ³⁵/₁₇ Contacto del pedido (datos de contacto del pedido). ³⁵/₁₇ Dirección de envío. <p>Las pestañas restantes contienen información relativa al pedido de venta y su interacción con algunas otras áreas del sistema.</p>
4	Una vez capturados los datos necesarios el usuario presiona el botón de <i>Guardar</i> .	En caso de haber llenado de manera satisfactoria el formulario correspondiente al nuevo pedido de venta, el sistema muestra la misma pantalla con los datos de la preventa pero los campos ya no son editables, lo que significa que el cliente ya esta almacenado en la base de datos. Además se habilita el botón <i>Eliminar</i> , con el cual podemos eliminar esta preventa del stock de preventas.

Flujo Alternativo:

Paso	Entrada	Respuesta del Sistema
5.1	El usuario presiona el botón de <i>Editar</i> , dentro de la ventana con el resumen de los datos de la preventa.	El sistema convierte editables los campos correspondientes a este pedido de venta para que el usuario pueda modificar los datos.
5.2	El usuario captura las modificaciones relacionadas al pedido de venta, pero siempre respetando los campos obligatorios y se presiona el botón <i>Guardar</i> .	Una vez mas el sistema almacena los cambios correspondientes a este pedido y vuelve a mostrar el formulario con los datos correspondientes al pedido pero ahora estos campos ya no son editables, lo que significa que los cambios han sido almacenados de manera exitosa.
5.3	El usuario selecciona la pestaña correspondiente al <i>Historial</i> ,	El sistema muestra la información correspondiente al historial relacionado a esta preventa y se pueden encontrar datos como: <ul style="list-style-type: none"> ³⁵/₁₇ Fecha de captura. ³⁵/₁₇ Numero. ³⁵/₁₇ Cliente. ³⁵/₁₇ Cliente. ³⁵/₁₇ Descripción.

Post-condiciones:	Al regresar al menú de los pedidos de ventas se puede constatar que la preventa antes capturada ya se encuentra en el listado de preventas.
Excepciones no controladas:	³⁵ / ₁₇ No se puede registrar la preventa debido a que no hay conexión con el servidor de la base de datos.
Consecuencias de Fallo:	³⁵ / ₁₇ De existir un fallo al momento registrar o modificar una preventa, los cambios realizados no se almacenan en la base de datos y por lo tanto el historial de preventas no genera cambio alguno.
Garantías Minina:	⚡ Se garantiza que de ocurrir algún problema al momento de

	<p>registrar o modificar una preventa, esto no afectará el desempeño del sistema por lo que basta con repetir el procedimiento para poder concluirlo.</p> <ul style="list-style-type: none"> ⤴ De igual manera se garantiza que al momento de registrar o modificar una preventa, no serán alterados los datos almacenados en la base de datos. ⤴ Se garantiza la persistencia de los datos de una preventa modificada.
Frecuencia:	Este caso de uso tiene una frecuencia alta debido a que es un caso de uso muy frecuente.
Supuestos:	Se asume que el usuario encargado de administrar las ventas tenga conocimientos básicos de administración de ventas.

Comentarios:

Desempeño:

- ³⁵₁₇ Siempre que se registra una nueva preventa.
- ³⁵₁₇ Esta puede permanecer tal cual y generar modificaciones hasta que sea necesario convertirla en venta.
- ³⁵₁₇ Eliminar la preventa y cancelarla por completo.

Interfaz de Usuario:

- ³⁵₁₇ Es necesario haber habilitado el modo de *interfaz avanzada* para poder administrar de manera completa el modulo de ventas de ventas para poder administrar de manera correcta.

Casos Siguietes:

- ³⁵₁₇ El caso que continuará es el caso relacionado con la generación de facturas.

Tarea 5.1: Generar preventa.

Para poder generar una preventa es necesario que el usuario con el que se este llevando a cabo la operación cuente con los permisos necesarios o que en su defecto pertenezca al grupo destinado a la administración de ventas. De no ser así no se tendrá acceso al menú relacionado con las ventas, en este caso como en casos pasados haremos uso del usuario administrador para poder generar una venta.

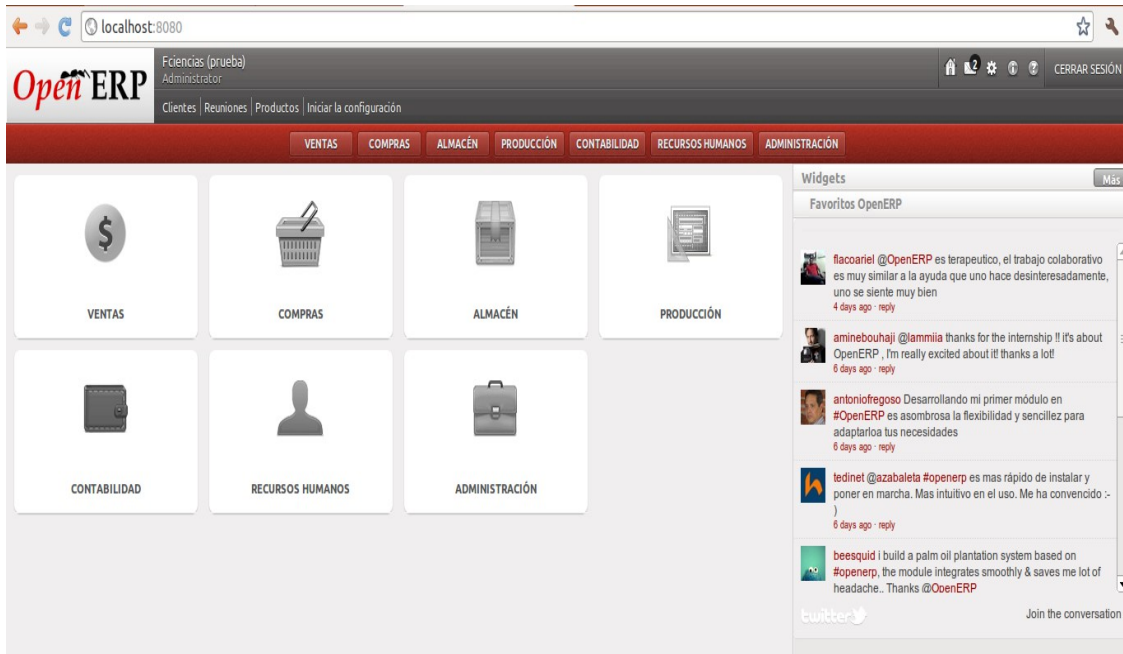


Figura 5.48 Panel principal Administrador

La Figura 5.48 es la pantalla principal que muestra el sistema siempre que el usuario ingresa y dependiendo de los módulos que tenga habilitados se mostrarán diferentes opciones. Para el caso de las ventas es necesario seleccionar la opción en la que se puede leer *Ventas*, para que podamos capturar los datos necesarios y así generar una venta.



Figura 5.49 Tablero de ventas

En esta Figura 5.49 podemos encontrar la lista de preventas registrados en el sistema, a diferencia de otras pantallas en este caso la única forma de buscar una preventa es recorriendo el listado de preventas. También es en esta pantalla en la que podemos encontrar en el menú del lado izquierdo la opción en la que se lee *Pedidos de Ventas* y en la cual podemos encontrar la opción para capturar una nueva preventa.

Los pedidos de ventas le ayudan a gestionar presupuestos y pedidos de sus clientes. OpenERP sugiere que comience por crear un presupuesto. Una vez esté confirmado, el presupuesto se convertirá en un pedido de venta. OpenERP puede gestionar varios tipos de productos de forma que un pedido de venta puede generar tareas, órdenes de entrega, órdenes de fabricación, compras, etc. Según la configuración del pedido de venta, se generará una factura en borrador de manera que sólo hay que confirmarla cuando se quiera facturar a su cliente.

[Ocultar este consejo](#) [Desactivar todos los consejos](#)

★ **Buscar: Pedidos de ventas** [Iconos de herramientas]

Referencia del pedido:
 Fecha pedido:
 Equipo de ventas:
 Cliente:
 Comercial:

Agrupar por... - Filtros -

Pedidos de ventas 1-3 de 3

REFERENCIA DEL PEDIDO	FECHA PEDIDO	REFERENCIA CLIENTE	CLIENTE	COMERCIAL	ENVIADA	FACTURADO	BASE IMPONIBLE	TOTAL	ESTADO DEL PEDIDO
SO008	14/02/2012		China Export	Administrator	0.0%	0.0%	0.00	0.00	Presupuesto
SO007	14/02/2012		Agrolait	Administrator	0.0%	0.0%	0.00	0.00	Presupuesto
SO001	02/02/2012		Agrolait	Administrator	0.0%	0.0%	3,263.00	3,263.00	Presupuesto
							3,263.00	3,263.00	

1-3 de 3

Figura 5.50 Panel de búsqueda de pedidos de ventas

En la pantalla de la Figura 5.50 podemos ver una vez más el listado de las preventas capturadas, así como también las opciones (campos de búsqueda: Referencia del pedido, Fecha del pedido, Cliente, etc.) para llevar una búsqueda especializada o incluso filtrar las ventas para ver las que ya pueden ser facturadas. La pantalla nos muestra un botón con el que podemos capturar una nueva preventa y al presionarlo se mostrará una ventana como la siguiente.

Pedidos de ventas [Iconos de herramientas]

- de 3

Referencia del pedido:
 Fecha pedido:
 Pagado ? :

Tienda:
 Referencia cliente:
 Entregado ? :

Cliente:
 Contacto del pedido ? :

Dirección de factura ? :
 Dirección de envío ? :

Tarifa ? :
 Cuenta analítica ? :

Líneas pedido de ventas 0-0 de 0

DESCRIPCIÓN	CTDAD	UDM	DESCUENTO (%)	PRECIO UNIDAD	SUBTOTAL

0-0 de 0

Base imponible ? :
 Impuestos ? :
 Total ? :

Estado del pedido ? :

Figura 5.51 Panel de alta y modificación de pedidos de ventas

Es necesario capturar los datos obligatorios para generar una nueva preventa, algo que facilita en gran medida el trabajo es la capacidad de poder buscar un cliente y al seleccionarlo el resto de los campos correspondientes se llenan de manera automática. Ya que han sido capturados todos los campos obligatorios procedemos a presionar el botón de *Guardar*, con los que los campos dejan de ser editables y se muestra una imagen como la de la Figura 5.52.

Pedidos de ventas
Referencia del pedido: SO009

Nuevo Editar Duplicar Eliminar

Referencia del pedido: SO009 Fecha pedido: 14/02/2012 Pagado ? :

Tienda: Facultad de Ciencias Referencia cliente: Entregado ? :

Pedido de venta Otra información Historial

Cliente: Maxtor Contacto del pedido ? : Wong, China Hong Kong 56 Beijing street

Dirección de factura ? : Wong, China Hong Kong 56 Beijing street Dirección de envío ? : Wong, China Hong Kong 56 Beijing street

Tarifa ? : Tarifa pública (MXN) Cuenta analítica ? :

DESCRIPCIÓN	CTDAD	UDM	DESCUENTO (%)	PRECIO UNIDAD	SUBTOTAL

Base imponible ? : 0.00 Impuestos ? : 0.00 Total ? : 0.00

Calcular Avanzar factura

Estado del pedido ? : Presupuesto

Cancelar pedido Confirmar pedido

Figura 5.52 Panel de alta y modificación de pedidos de ventas (solo lectura)

Tarea 5.1.2: Modificación de preventas

En la Figura anterior 5.52 se puede apreciar un botón en el que se lee *Editar*, que al presionarlo vuelve a hacer que los campos retomen su modalidad de “editables”, lo que nos permite editar la información de cualquier preventa que se desee. Al terminar de editar la venta en cuestión basta con repetir el proceso para finalizar la captura de una nueva venta y ya con eso la tarea 5.1.2 queda concluida, para corroborar lo anterior basta con revisar el listado de preventas y buscar la preventa que fue modificado de tal manera que podamos apreciar los cambios realizados a la misma.

Tarea 5.1.3: Eliminar preventa

Para eliminar una preventa basta con volver a la pantalla que se muestra en la Figura 5.50 y seleccionar la preventa que se quiera eliminar, de tal manera que al hacer esto se palomea el cuadro de selección correspondiente a esa preventa y también se habilita el botón para eliminarla, como se muestra en la siguiente Figura 5.53.

★ Buscar: Pedidos de ventas

PRESUPUESTOS VENTAS PARA FACTURAR

Referencia del pedido: Fecha pedido: Equipo de ventas: Cliente: Comercial: Administrator

Agrupar por... Buscar Limpiar

REFERENCIA DEL PEDIDO	FECHA PEDIDO	REFERENCIA CLIENTE	CLIENTE	COMERCIAL	ENVIADA	FACTURADO	BASE IMPONIBLE	TOTAL	ESTADO DEL PEDIDO
<input type="checkbox"/> SO007	14/02/2012	Agrolait	Administrator	0.0%	0.0%	0.00	0.00	Presupuesto	×
<input type="checkbox"/> SO001	02/02/2012	Agrolait	Administrator	0.0%	0.0%	3,263.00	3,263.00	Presupuesto	×
							3,263.00	3,263.00	

Figura 5.53 Panel de búsqueda de pedidos de ventas

Ya con el botón de *Eliminar* habilitado es suficiente con presionarlo para que se muestre una ventana emergente en la que se nos pide confirmar que se desea eliminar la preventa seleccionada.

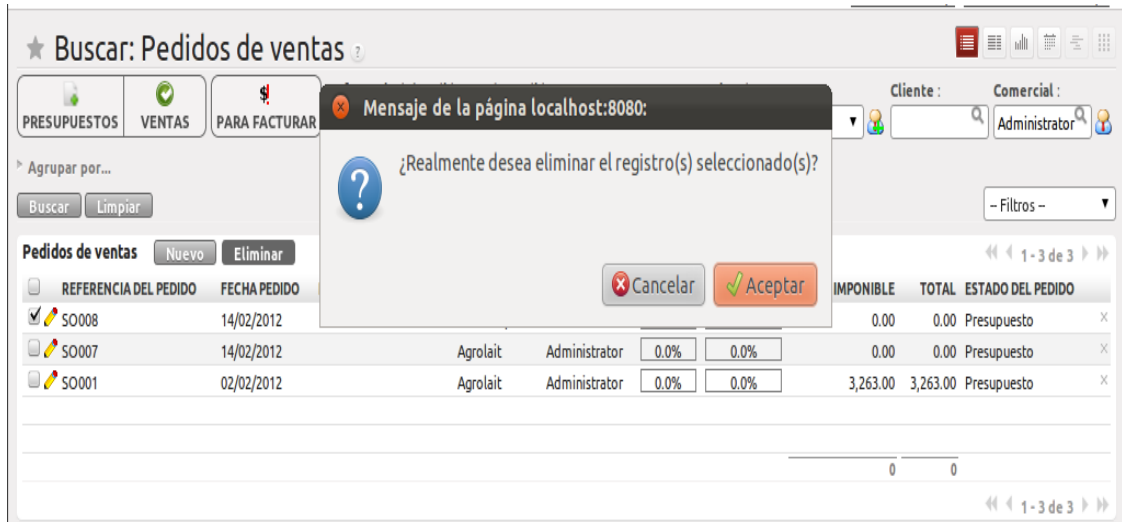


Figura 5.54 Ventana de confirmación de baja de ventas

Si se presiona *Aceptar* el sistema tarda unos segundos en procesar la solicitud y validar que no haya procesos asociados a esta preventa en particular. En caso de no tener procesos pendientes la preventa es eliminada y ya no es accesible desde el listado de ventas.

Tarea 5.2.1: Generar Venta

Una vez que la preventa ya fue generada y almacenada esta puede convertirse en una venta, la cual tiene la característica de que ya puede ser facturada. Una venta es un pedido de venta que ya fue confirmado por el cliente y que por lo tanto ya está en proceso de ser despachada y que a su vez tendrá que ser pagada por el cliente. En la Figura 5.55 podemos ver el botón con el que se confirma el pedido y la preventa se convierte en una venta que ya puede ser facturada. En este pedido aun no se capturan productos y/o servicios pero se puede identificar que el pedido es para el cliente *Maxtor*, que la referencia del pedido es la *SO009*, así como la fecha del pedido es del *14/02/2012* además de otros campos que corresponden tanto al cliente como al pedido.

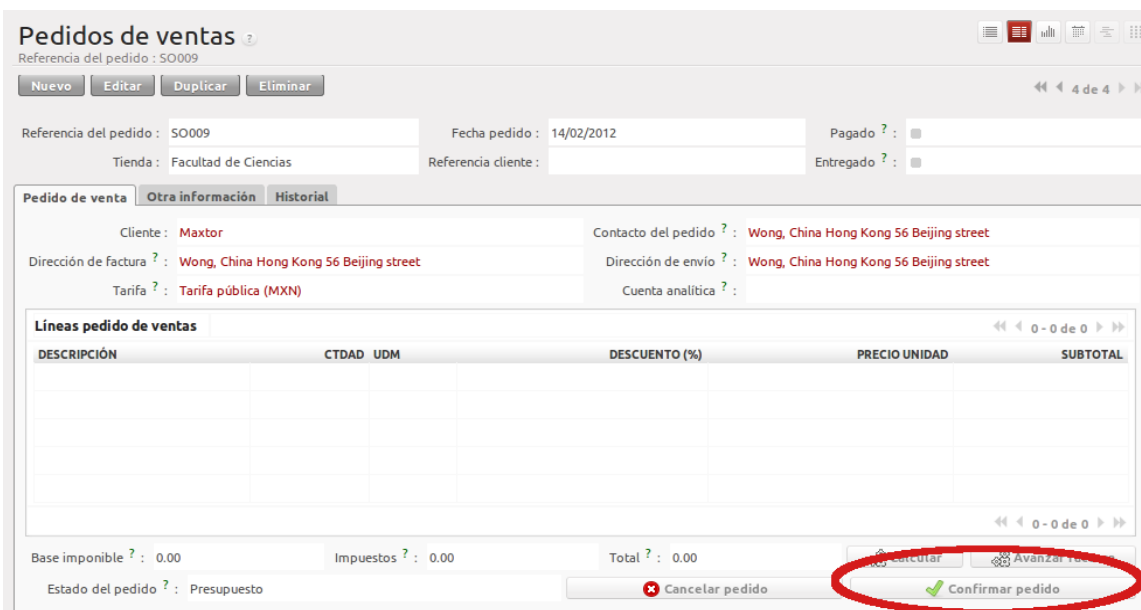


Figura 5.55 Panel de pedidos de ventas

Tarea 5.2.1: Historial de Ventas

Por último, el historial de ventas lo podemos consultar en el *Tablero de Ventas*, este tablero se localiza en el menú principal de ventas y es en este tablero en el cual podemos consultar el estatus de cada venta que ya ha sido generada. En este tablero se puede determinar si la venta ya fue facturada, enviada y todo lo relacionado con el historial de las ventas. En el recuadro de color negro se muestran los datos más importantes relacionados a los pedidos de ventas y al seleccionar alguno de ellos podemos ingresar al pedido de venta de manera más específica, de tal manera que podamos modificar estos datos o incluso cancelar la venta

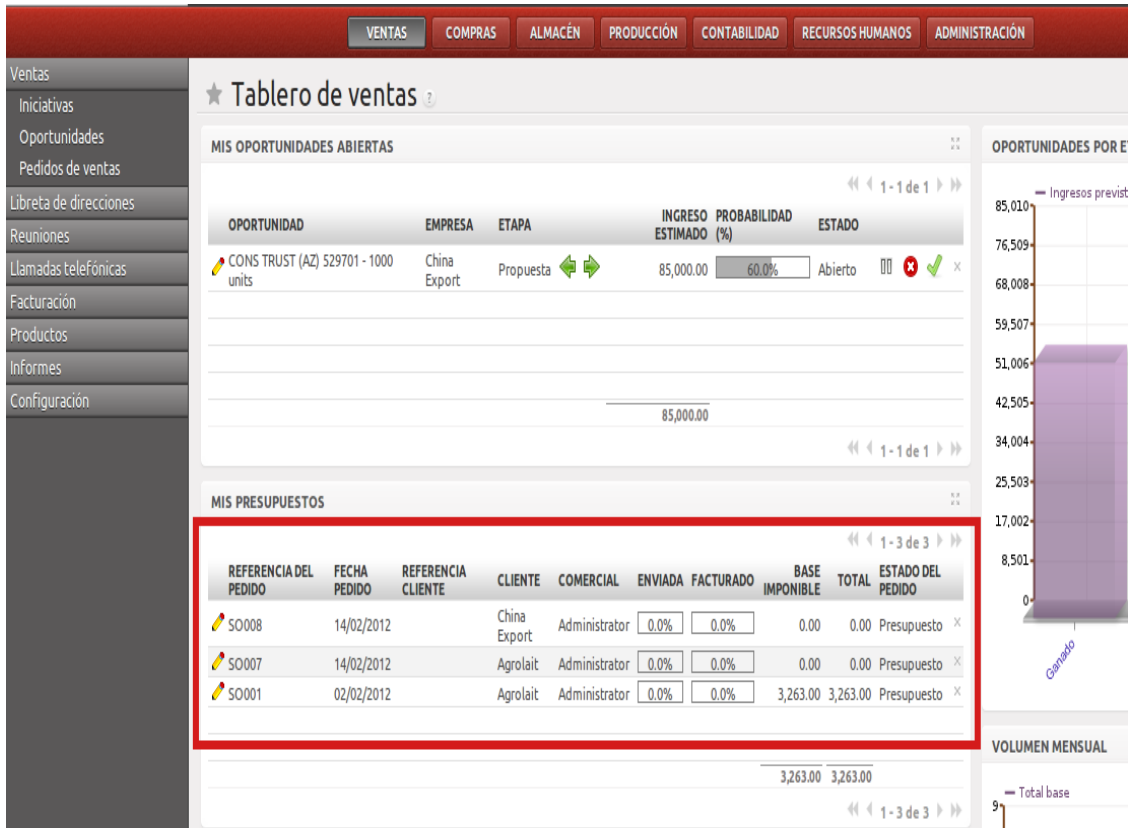


Figura 5.56 Tablero de ventas

Retrospectiva del Ciclo:

Este ciclo es uno de los más importantes debido a que es el ciclo en el cual se generan las ventas necesarias para poder facturar de manera electrónica. Es necesario seguir cada una de las tareas descritas en este ciclo para poder generar una factura, de no llevar a cabo cada una de las tareas planteadas en este ciclo no es posible generar ventas y por lo tanto no sería posible generar facturas.

En el siguiente ciclo podremos ver el proceso para instalar el módulo para poder facturar de manera electrónica.

5.7 Ciclo 8: Instalación del módulo de facturación electrónica

Para poder comenzar a generar facturas electrónicas es necesario instalar el módulo adecuado para el estándar de facturación mexicana. Es necesario mencionar que existe más de una versión que cumple con el propósito de generar facturas electrónicas. Sin embargo el módulo que instalaremos en este ciclo es uno de los que ya llevan más tiempo en desarrollo, aproximadamente desde Mayo del año 2011²⁸ y ha probado ser lo suficientemente maduro como para poder implementarse de manera estable en un ambiente de producción.

Es necesario cumplir con algunos prerrequisitos antes de instalar este módulo, como lo es localizar las fuentes del mismo descargar el código e instalarlo. Estas tareas se muestran a continuación.

- ³⁵₁₇ Localizar código fuente
- ³⁵₁₇ Descargar código fuente (Bazaar²⁹).
- ³⁵₁₇ Instalación de módulos necesarios.

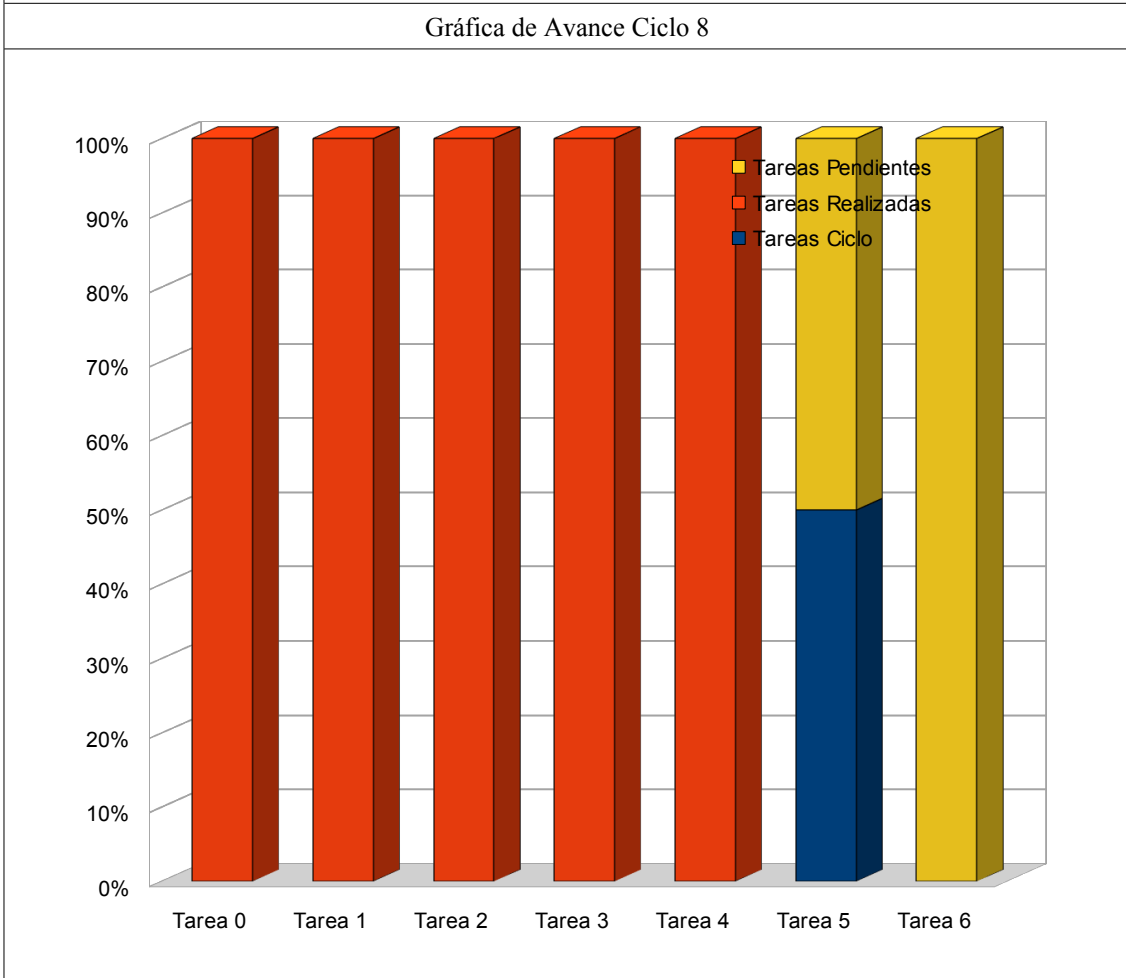
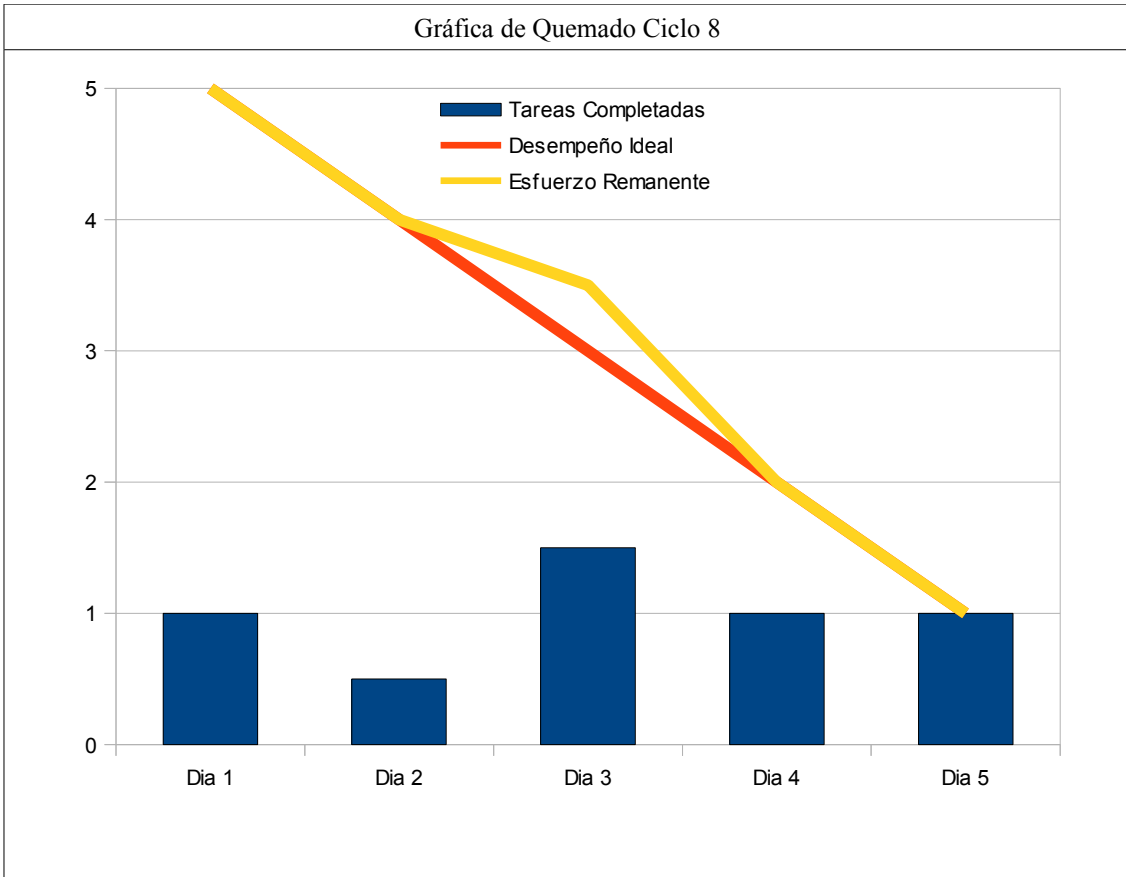
El conjunto de módulos que serán instalados han sido generados por la comunidad de desarrolladores especializados en OpenERP para México, por lo tanto el código es libre y cualquiera puede tener acceso al mismo. Es importante mencionar que en ningún momento el autor de esta tesis se adjudica la autoría del desarrollo del código de este módulo y por el contrario, agradezco el que el código sea accesible y fácil de obtener³⁰ para todo desarrollador.

²⁸ Es la fecha en la que se subió la primera versión del módulo de facturación electrónica mexicana en el sitio <https://launchpad.net/openerp-mexico-localization>

²⁹ Sistema de control de versiones que es utilizado por la comunidad de desarrollo de software para contribuir a la mejora de los proyectos de código libre.

³⁰ Para poder obtener el código fuente de los módulos requeridos fue necesario hacer uso de esta Bazaar, Con esta herramienta tanto se puede tener acceso al código fuente como también se puede realizar aportaciones para mejorar el desempeño de este módulo.

Sprint Backlog: SiFaDig			
Ciclo: 8	<p>Objetivos Generales: El modulo de facturación electrónica consta de vario componentes que deben ser instalados, además para poder tener acceso a estos módulos es necesario ubicar las fuentes y descargarlas para posteriormente proceder a instalarlo.</p> <p>6.- Instalación del módulo de facturación electrónica</p> <p>Tareas para este ciclo:</p> <p>³⁵₁₇ Localizar código fuente. ³⁵₁₇ Descargar código fuente (Bazaar). ³⁵₁₇ Instalación de módulos en OpenERP. ³⁵₁₇ Generar Factura Electrónica.</p>		
Fecha de Inicio: 24/05/2011			
Fecha de Terminó: 31/05/2011			
Tareas			
Id	Descripción de la Tarea	Tamaño	Prioridad
6.1.1	Localizar código fuente	1	3
6.1.2	Descargar código fuente (Bazaar)	1	3
6.1.3	Instalación de módulos en OpenERP	1	3
6.1.4	Generar Factura Electrónica	1	3



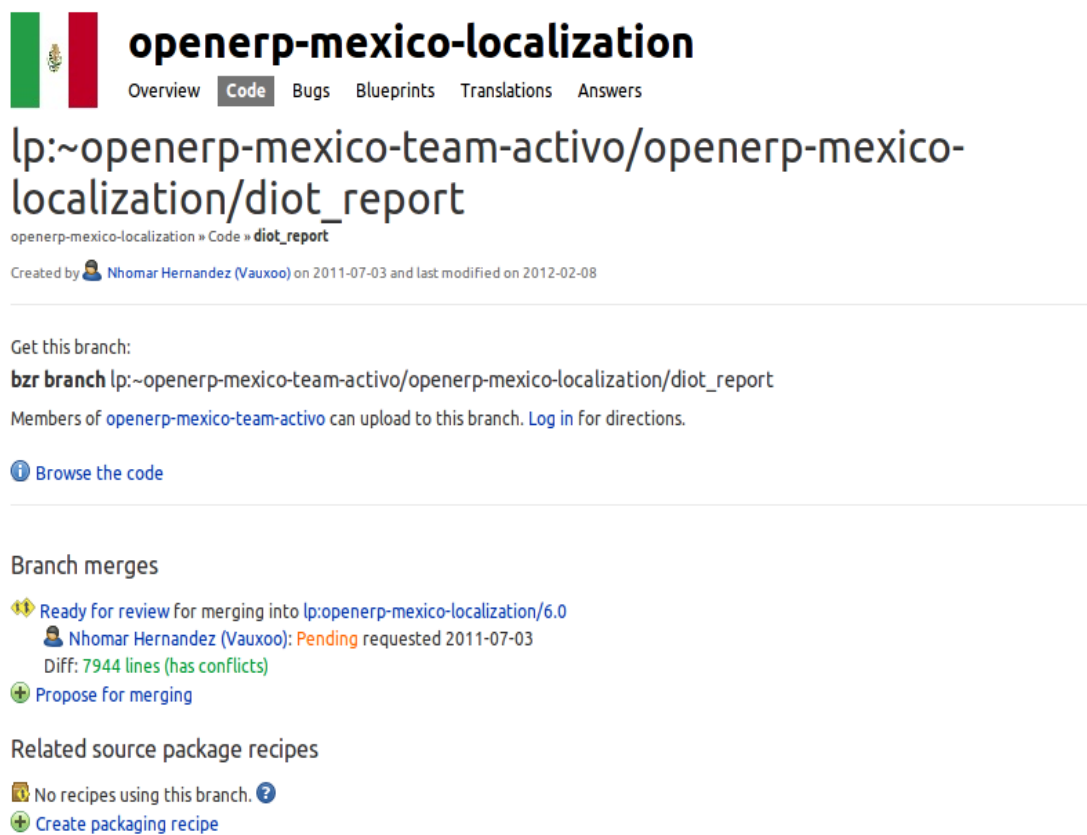
Tarea 6.1.1: Localizar código fuente.

Tras un arduo análisis y comparación entre los diferentes módulos de facturación electrónica que existen disponibles de manera libre, se encontró que la más estable es:

https://code.launchpad.net/~openerp-mexico-team-activo/openerp-mexico-localization/diot_report

Es el más estable debido a que es el modulo al que se le han realizado más pruebas y contribuciones de parte de la comunidad de desarrolladores mexicanos. Esto se puede constatar al revisar los repositorios de código dentro del sitio de descarga y verificar que constantemente se reportan errores y actualizaciones relacionadas con este modulo.

En este sitio se pueden encontrar varias versiones de este modulo, así como diferentes revisiones del mismo y reportes, lo que trae como ventaja que se pueden identificar e incluso resolver errores que se hayan detectado en alguna versión disponible.



The screenshot shows the Launchpad interface for the code branch 'diot_report' within the 'openerp-mexico-localization' project. At the top, there is a navigation bar with links for 'Overview', 'Code', 'Bugs', 'Blueprints', 'Translations', and 'Answers'. The main heading is 'lp:~openerp-mexico-team-activo/openerp-mexico-localization/diot_report'. Below this, it indicates the branch was created by 'Nhomar Hernandez (Vauxoo)' on 2011-07-03 and last modified on 2012-02-08. A section titled 'Get this branch:' provides the 'bzd branch' command: 'lp:~openerp-mexico-team-activo/openerp-mexico-localization/diot_report'. It also notes that members of the 'openerp-mexico-team-activo' can upload to this branch. There is a 'Browse the code' link. The 'Branch merges' section shows a merge request from 'Nhomar Hernandez (Vauxoo)' on 2011-07-03, with a diff of 7944 lines and conflicts. It includes 'Ready for review' and 'Propose for merging' options. The 'Related source package recipes' section shows 'No recipes using this branch.' and a 'Create packaging recipe' link.

Figura 6.1 Sitio de descarga del modulo de facturación electrónica mexicana

En Figura 6.1 podemos encontrar múltiples ligas con los que podemos navegar en el código fuente, así como reportar errores o realizar anotaciones que puedan servir para identificar fallos en el código o posibles actualizaciones a los estándares de facturación que puedan afectar el desempeño del modulo, sin embargo hay que poner atención en la parte que se menciona la instrucción, ya que esta instrucción será útil al momento de descargar el modulo de facturación electrónica.

```
bzd branch lp:~openerp-mexico-team-activo/openerp-mexico-localization/diot_report
```

Tarea 6.1.2: Descarga del código fuente (Bazaar).

Antes de proceder a descargar el código fuente, es necesario instalar **Bazaar**. Es un sistema de control de versiones patrocinado por Canonical³¹. **Bazaar**, al contrario que **Subversion**³² y **CVS**³³, que solo soportan un modelo de repositorio centralizado, permite un sistema distribuido de control de versiones, lo que da un mayor eficiencia en la colaboración. Particularmente, **Bazaar** está diseñado para aumentar el nivel de participación de la comunidad en proyectos de código abierto.

Para poder instalar **Bazaar** basta con teclear desde una terminal el comando:

```
$ sudo apt-get install bzzr
```

Y con este sencillo comando tenemos instalado **Bazaar** y lo necesario para acceder al repositorio en el cual se encuentra el código fuente del módulo de facturación electrónica. Ahora abrimos una terminal y tecleamos la ruta de la tarea anterior, con lo cual podemos proceder a descargar el código de los módulos que se quieren instalar.

```
bzzr branch lp:~openerp-mexico-team-activo/openerp-mexico-localization/diot_report
```

Después de haber tecleado este comando podemos ubicar la carpeta *diot_report* en la ruta *./home/Documentos*. Y con esto ya tenemos en nuestro equipo lo necesario para poder proceder con la siguiente y última tarea de este ciclo.

6.1.3: Instalación de módulos en OpenERP

La forma en la cual se instalan los módulos de la facturación electrónica es idéntica a la forma en la cual se instala cualquier módulo. Lo primero que hay que hacer es ubicar la ruta del directorio “*addons*”, en mi caso en particular se encuentra en “*./home/mike/Documentos/Tesis/openerp/server/bin/addons*”. Ya con esta carpeta localizada lo siguiente es copiar el contenido de la carpeta “*diot_report*” (vista en la tarea pasada) y pasarlo a la carpeta “*addons*”.

Ahora es necesario ubicar la sección de administración dentro de *OpenERP*, posteriormente seleccionar la opción *Módulos* y actualizar la lista de módulos que pueden ser instalados. En este panel podremos ver los módulos instalados, así como también podemos habilitar nuevos módulos que vengan precargados pero que no hayan sido activados o como en este caso instalar un módulo completamente ajeno a la instalación por defecto.

NOMBRE	CATEGORÍA	DESCRIPCIÓN BREVE	AUTOR	ÚLTIMA VERSIÓN	VERSIÓN INSTALADA	ESTADO
account	Accounting	Gestión contable y financiera	OpenERP SA	6.0.1.1	6.0.1.1	Instalado
account_accountant	Accounting	Contable	OpenERP SA	6.0.1.1	6.0.1.1	Instalado
account_analytic_analysis	Accounting	report_account_analytic	Campocamp	6.0.1.1		No instalado
account_analytic_default	Accounting	Account Analytic Default	OpenERP SA	6.0.1.0		No instalado
account_analytic_plans	Accounting	Multiple-plans management in Analytic Accounting	OpenERP SA	6.0.1.0		No instalado
account_anglo_saxon	Inventory Control	Stock Accounting For Anglo Saxon countries	OpenERP SA, Ventos	6.0.1.2		No instalado
account_budget	Accounting	Budget Management	OpenERP SA	6.0.1.0		No instalado
account_cancel	Accounting	Account Cancel	OpenERP SA	6.0.1.1		No instalado
account_chart	Accounting	Planes contables	OpenERP SA	6.0.1.1	6.0.1.1	Instalado
account_coda	Account CODA	Account CODA - import bank statements from coda file	OpenERP SA	6.0.1.0		No instalado

³¹ Empresa privada que brinda soporte a proyectos de software libre, como en este caso a **Bazaar** [20].

³² Sistema de control de versiones que permite llevar un control adecuado de las varias versiones que se pueden generar de un determinado software [21].

³³ Concurrent Version System de sus siglas en inglés y al igual que **Subversion** es un sistema destinado a mantener el control de las versiones de software [22].

Figura 6.2 Panel de Administración de Módulos

En el menú principal podemos ubicar la opción *Actualizar lista de módulos*, una vez que hayamos seleccionado esta opción se nos pedirá confirmar la actualización, lo cual tenemos que aceptar para que la lista sea actualizada, como se ve en la Figura 6.3







Figura 6.3 Panel de Actualización de módulos

Una vez que ya se ha actualizado la lista, se muestra otra pantalla con el resumen de la operación anterior, con la cantidad de módulos disponibles y el número de módulos agregados, como se ve en la Figura 6.4.



Figura 6.4 Panel de resumen de módulos actualizados

Ahora lo que resta es ubicar en la lista de módulos disponible los módulos:

-  **base_vat_mx:** modulo que realiza las validaciones necesarias para crear una factura electrónica.
-  **l10n_mx_facturae:** modulo encargado de generar las facturas electrónicas
-  **l10n_mx_facturae_cbb:** modulo encargado de generar el código de barras bidimensional.
-  **l10n_mx_facturae_cfdi:** modulo encargado de generar los comprobantes fiscales digitales.

Una vez ubicados estos módulos, es necesario seleccionar cada uno de ellos y en la siguiente pantalla presionar el botón *Programar para instalación*, como se muestra en la siguiente imagen.

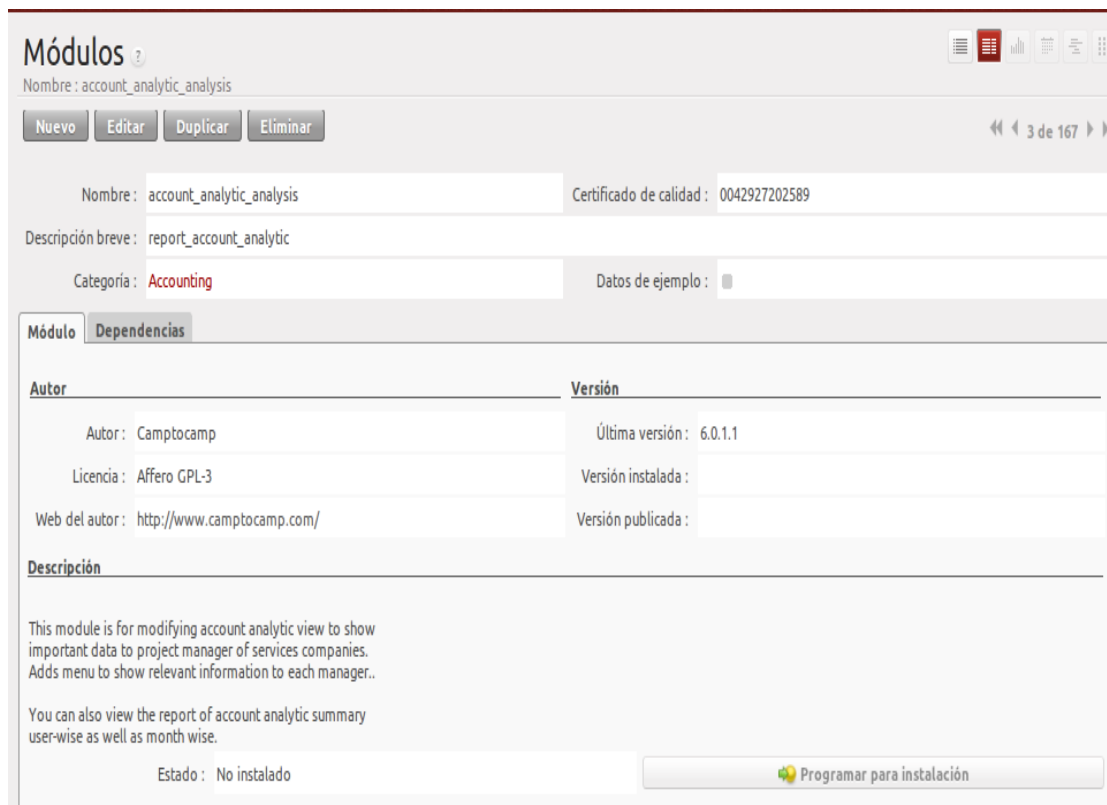


Figura 6.5 Panel de resumen del modulo

Una vez que todos los módulos han sido programados para ser instalados, es necesario presionar el botón *Aplicar actualizaciones programada*, el cual podemos ubicar en el menú de la Figura 6.5. De igual manera que en el procedimiento pasado se mostrara un pequeño resumen con la cantidad de módulos instalados y en caso de ser necesario se pedirá configurar cada uno de ellos.

Una vez que han sido terminadas estas tareas, ya tenemos instalado y listo para ser usado el módulo de facturación electrónica. En el siguiente ciclo se mostrara como generar de manera sencilla una factura electrónica.

Tarea 6.1.4: Generar Factura Electrónica.

Ya con todos los módulos necesarios instalados, podemos generar una factura desde cero como ya hemos visto en el ciclo 6 o podemos ir directamente al módulo de *Contabilidad* y seleccionar alguna de las facturas que ya estén generadas para que podamos ver del lado superior izquierdo, el menú correspondiente a la facturación electrónica, como se muestra en la siguiente Figura 6.6.

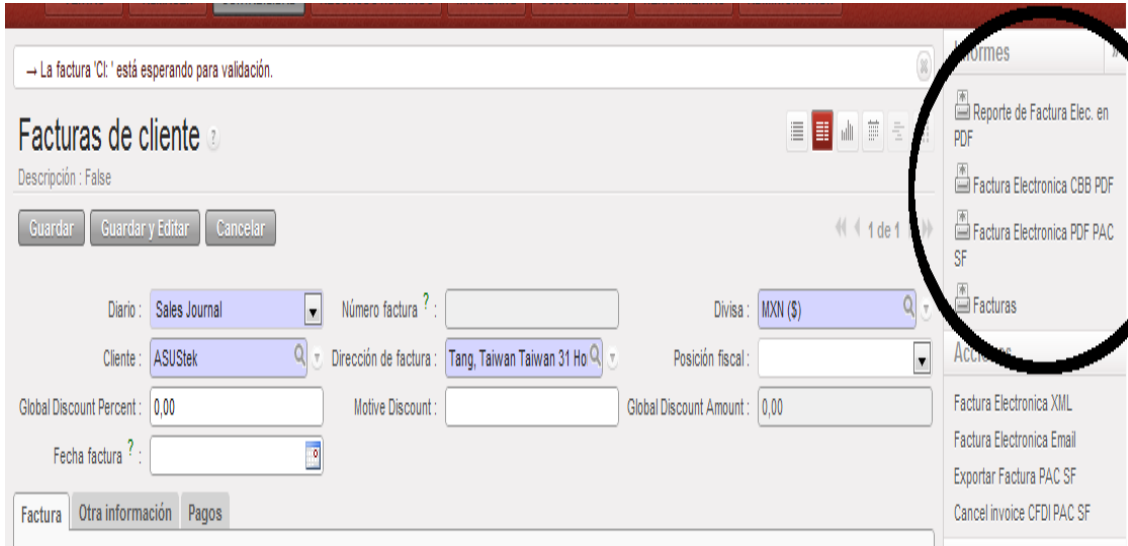


Figura 6.6 Panel de Facturas de Cliente

Y es en este menú en el cual podemos encontrar todas las herramientas necesarias para generar una factura electrónica, así como los elementos que la integran como:

- 35 17 Archivo XML.
- 35 17 Cadena Original.
- 35 17 Sello digital.
- 35 17 Factura electrónica en formato PDF.

De igual manera en el menú del lado superior izquierdo podemos encontrar un botón en el que se lee “*Factura Electrónica Reporte Mensual TXT*”. Mismo que al presionarlo podemos seleccionar el mes del cual queremos generar el reporte, como se muestra en la siguiente Figura.

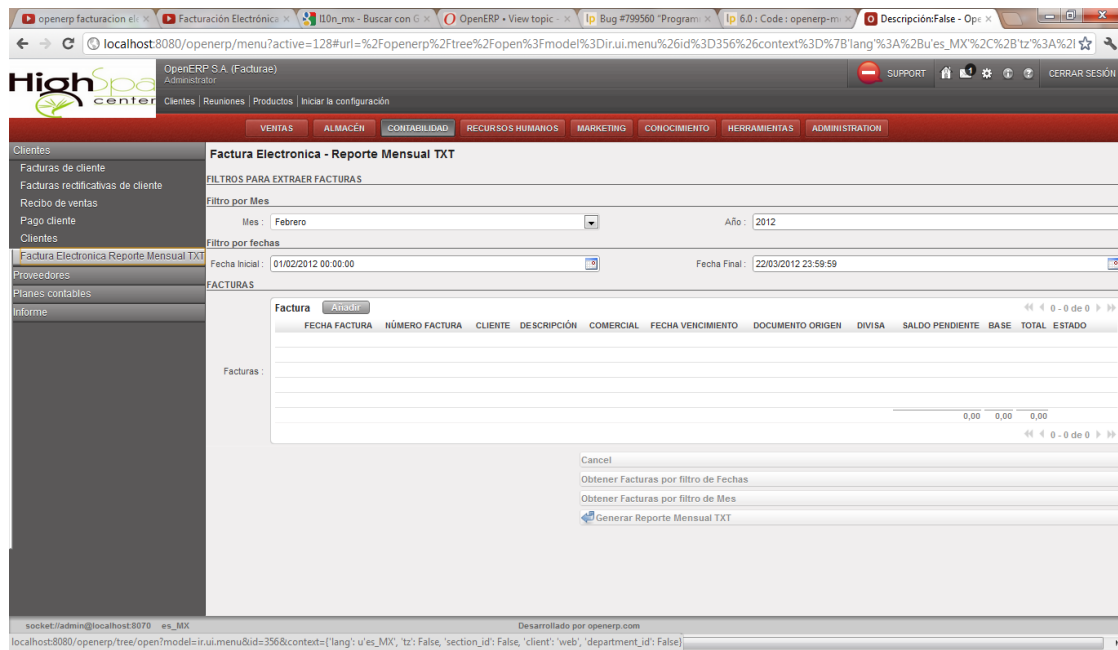


Figura 6.7 Panel de reporte mensual TXT

Y por último solo resta mostrar la imagen de una factura electrónica como es generada una vez que ha atravesado por todo el proceso mencionado en esta tesis.

	IVA (16.0%) \$	800.00
	Total \$	5800.00

IMPORTE CON LETRA:
CINCO MIL OCHOCIENTOS PESOS 00/100 M. N.

PAGO EN UNA SOLA EXHIBICIÓN

"Este documento es una representación impresa de un CFDI"
CFDI, Comprobante Fiscal Digital por Internet

Certificado del emisor	Certificado del SAT	Fecha de Timbrado	Folio Fiscal
00001000000104464083	30001000000100000801	2011-11-04 02:45:57	A464A301-0476-4335-9773-3BC01A78EE08



Sello Digital Emisor:
ajAGxtYEsVn2YAo5Q92tMPVodyNUpaglWaiQCd1xsw7qUzIB325V1oYmd7p5YDY3P+FzIqj90
/ntB0znmS4s85kvScrQ039Odxsm19bBdgt3erdm4Wjglx4mcPp3eqKdZRQzzkWt6MZv8vzxSg1
VuzADY8obTv6aod5FYgNDxE=

Sello Digital SAT:
Jakoeo2GsN5mYM94uxjenZmOcStAmdd3+TrXaTcck2ci5ZtlmMv8jqr2Xqa7YUXxOrfYCNK+Te
T6vWlOn//IS1rRML5000NkEIAKqyR4j815CDihuAnDhOi+vxFjMkIYKhTVMrL+Q/8ZbObPebD
lUzpzAIWRhtoPEEc+BacM6M=

Cadena Original del complemento de certificación digital del SAT:
||1.0|A464A301-0476-4335-9773-3BC01A78EE08|2011-11-03T20:45:57|gyG8zNuiatl
oj6R5cPOHy1BAA+KMuki2hfgR8MoOo5xnVcpEBOqj4kkoYlkBcWvkEN17AgNjy93t1j2YhivRu
WQeoCHDgeMsfOfLWMTkj3vKhAdALJomSUDKFURX2PHNN1YAS7YOOalKy+TpacTA2SFdeF1dqqM
p677BIKpKdCw=|30001000000100000801||



Figura 6.8 Factura electrónica emitida

En esta última imagen podemos ver todos los elementos que integran una factura electrónica conjuntos y con los cuales podemos generar una factura electrónica como lo mandan las normas del SAT.

Conclusiones

Esta tesis tuvo como objetivo principal implementar un sistema de facturación digital, que pudiera ofrecer una solución a todo aquel que requiera incluir los estándares de la facturación electrónica mexicana en sus procesos. Todo esto se logró siguiendo una metodología ágil conocida como SCRUM, la cual sirvió de marco teórico para poder concluir de manera adecuada con las tareas que se plantearon al inicio de esta tesis.

Los objetivos de este trabajo eran:

- ³⁵₁₇ Desarrollar el módulo de facturación electrónica, sin embargo a lo largo del trabajo y con la constante interacción con *OpenERP* nos encontramos con que el módulo ya había sido desarrollado y que era más sencillo mostrar los pasos a seguir para poder conseguirlo e instalarlo, los cuales son:
 1. Identificar el módulo adecuado.
 2. Descargar la versión correcta del modulo.
 3. Instalar el módulo de facturación así como los componentes necesarios para que el módulo de facturación se desempeñara de manera correcta.
- ³⁵₁₇ Mediante el uso de *OpenERP* mostrar todos los elementos que son necesarios para poder generar una factura electrónica y a su vez mostrar la interacción entre cada uno de ellos.
- ³⁵₁₇ Siguiendo SCRUM tenemos una metodología ágil y ordenada que nos permite organizar las tareas y llevar un control sobre la realización de cada una de ellas, así como de las tareas faltantes.

En esta tesis se puede apreciar cómo es posible implementar un sistema tipo *Enterprise Resource Planning* (ERP) que pueda generar facturas electrónicas de manera automatizada y que además cumpla con los estándares de la contabilidad mexicana regida por el SAT.

En el último ciclo de SCRUM se puede apreciar la factura electrónica generada por el sistema, misma que resume el trabajo realizado para poder llegar a ella.

Por otro lado SCRUM nos sirvió de apoyo para poder tener un orden y un control efectivo sobre las actividades y el tiempo destinado a cada una de ellas. Además de ser una herramienta útil y flexible para poder generar una bitácora en la cual se documentaran los cambios y riesgos que se pueden presentar a lo largo de un proyecto. Una de las características principales de SCRUM es su flexibilidad y esto se pudo constatar de manera tangible en cada uno de los ciclos reafirmando lo importante que es esta característica para la gestión de proyectos en la actualidad, ya que de no haber contando con esta flexibilidad se tendrían que haber realizado cambios en los primeros capítulos de estas tesis, debido a que en éstos se fijaron las tareas a realizar. El cierre de cada ciclo fue uno de los elementos más importantes a lo largo de los mismos, ya que nos sirvió para poder identificar de manera clara los riesgos en cada uno de ellos y prevenirlos, como lo fue el hecho de que existan actualizaciones y correcciones para *OpenERP* que puedan afectar de manera importante el desempeño de éste.

En esta tesis se mostró que es posible implementar un sistema tipo ERP que integre la facturación digital de manera sencilla y con esto ayudar a resolver el problema de las empresas que no cuentan con los recursos necesarios para poder desarrollar un sistema de facturación digital por su propia cuenta. Como se puede apreciar en la última sección de esta tesis, el sistema genera facturas electrónicas con todos los elementos que marca el SAT, sin embargo el sistema aun no es puesto en producción debido a que para llevar a cabo esto es necesario contar con una empresa dispuesta a dedicar tiempo para implementarlo en sus instalaciones y esto queda fuera del alcance de esta tesis.

Bibliografía:

- [1]SCRUM <http://scrum.org.mx/> consulta: 15/12/2010
- [2]OpenERP <http://doc.openerp.com/book/0/index.html> consulta: 15/12/2010
- [3]Angel Moya <http://angelmoya.es/?s=Crear+Modulos+OpenERP> consulta: 20/12/2010
- [4]SAT
http://www.sat.gob.mx/sitio_internet/asistencia_contribuyente/principiantes/comprobantes_fiscales/ consulta: 05/02/2011
- [5]OpenERP <http://doc.openerp.com/v6.1/index.html> consulta: 07/04/2011
- [6] Metalenguaje: <http://www.wordreference.com/definicion/metalenguaje> consulta: 09/05/2011
- [7]SAT
http://www.sat.gob.mx/sitio_internet/asistencia_contribuyente/principiantes/comprobantes_fiscales/66_19339.html consulta: 03/06/2011
- [8]XHTML http://www.w3schools.com/html/html_xhtml.asp consulta: 15/10/2011
- [9]XSLT <http://www.w3schools.com/xsl/> consulta: 21/07/2011
- [10]Manifiesto ágil http://es.wikipedia.org/wiki/Manifiesto_%C3%A1gil consulta: 15/10/2011
- [11]Modularidad <http://www.alegsa.com.ar/Dic/modularidad.php> consulta: 16/10/2011
- [12]GTK www.acronymfinder.com/GTK.html consulta: 30/11/2011
- [13]PostgreSQL http://www.postgresql.org/es/sobre_postgresql consulta: 06/12/2011
- [14]IPv4 www.ipv6.mx/index.php/informacion/fundamentos/ipv4 consulta: 20/12/2011
- [15]IPv6 www.ipv6.mx/index.php/informacion/fundamentos/ipv6 consulta: 20/12/2011
- [16]Ubuntu <http://www.ubuntu-es.org/> consulta: 10/01/2012
- [17]UML www.uml.org/ consulta: 21/02/2012
- [18]MVC http://es.wikipedia.org/wiki/Modelo_Vista_Controlador consulta: 30/04/2012
- [19]Python <http://pyspanishdoc.sourceforge.net/ref/introduction.html> consulta: 25/06/2012
- [20]Canonical <http://www.canonical.com/> consulta: 15/07/2012
- [21]Subversion <http://subversion.tigris.org/> consulta: 15/07/2012
- [22]CVS <http://cvs.nongnu.org/> consulta: 15/09/2012
- [23]Modulo Facturación Electrónica <https://launchpad.net/openerp-mexico-localization> consulta: 20/07/2012

E. M. JIMÉNEZ Ingeniería de Software Ágil – Editorial: Bubok Publishing, Número de Páginas: 300, Primera Edición: Julio 2010.