

**Universidad Tecnológica del Centro de Veracruz**

**Tecnologías de la información y la comunicación**

**“DISEÑO DE UNA APLICACIÓN DESKTOP”**

**PERZAN**

**Presenta:**

**González Blanco María de los Ángeles**

**Ramírez García Alan Hernán**

**Sosa Rincón Daniel Isaac**

Cuitláhuac, Ver a 22 de septiembre del 2015

Contenido

[Índice de figuras 3](#_Toc436400747)

[Resumen 4](#_Toc436400748)

[Abstract 5](#_Toc436400749)

[Capítulo 1 Descripción del proyecto. 6](#_Toc436400750)

[Capítulo 2 Herramientas 11](#_Toc436400751)

[Capítulo 3 Marco teórico 14](#_Toc436400752)

[Capítulo 4 Metodología de Desarrollo 18](#_Toc436400753)

[Capítulo 5 Conclusiones 22](#_Toc436400754)

[Bibliografía 23](#_Toc436400755)

# Índice de figuras

[Imagen 1 9](file:///C:\Users\disr__000\Documents\utcv7\Perzan-master\1_DEFINICION\DESCRIPCION_DEL_PROYECTO.docx#_Toc436401509)

[Imagen 2 11](#_Toc436401510)

[Imagen 3 11](file:///C:\Users\disr__000\Documents\utcv7\Perzan-master\1_DEFINICION\DESCRIPCION_DEL_PROYECTO.docx#_Toc436401511)

[Imagen 4 11](file:///C:\Users\disr__000\Documents\utcv7\Perzan-master\1_DEFINICION\DESCRIPCION_DEL_PROYECTO.docx#_Toc436401512)

[Imagen 5 12](file:///C:\Users\disr__000\Documents\utcv7\Perzan-master\1_DEFINICION\DESCRIPCION_DEL_PROYECTO.docx#_Toc436401513)

[Imagen 6 12](file:///C:\Users\disr__000\Documents\utcv7\Perzan-master\1_DEFINICION\DESCRIPCION_DEL_PROYECTO.docx#_Toc436401514)

[Imagen 7 13](file:///C:\Users\disr__000\Documents\utcv7\Perzan-master\1_DEFINICION\DESCRIPCION_DEL_PROYECTO.docx#_Toc436401515)

[Imagen 8 13](#_Toc436401516)

[Imagen 9 13](#_Toc436401517)

[Imagen 10 20](#_Toc436401518)

# Resumen

Durante el cuatrimestre se le dio mantenimiento a la aplicación de escritorio ya que hacía falta implementar ciertas funcionalidades, aún faltan detalles, pero son muy mínimos. La interfaz de la aplicación es aún más atractiva que al principio y la manipulación de datos es más fácil para quien sea el usuario.

Para Realizar la aplicación se utilizaron algunos programas que permitieron su realización como fue Eclipse esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado *Java Development Toolkit* (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). Otro es Scene Builder un programa que permite crear la interfaz de la aplicación como son los botones, las tablas etc. Navicat es un programa que permite administrar las bases de datos que son las que contendrán la información que se ira generando al correr la aplicación.

A grandes rasgos, esto fue lo que se utilizó y lo que hace la aplicación es registrar empleados, registrar productos, modificar datos y eliminar.

Abstract  
  
During the quarter a desktop application to a company that sells decorative materials of interior and exterior that allows keeping track of existing products in store and sales on the day was done.

To Perform some application programs that allowed its realization as Eclipse platform was used, typically has been used to develop integrated development environments (IDE) such as Java IDE called Java Development Toolkit (JDT) and compiler ( CJD) shipped as part of Eclipse (which are also used to develop the same Eclipse). Another Scene Builder is a program that lets you create the application interface such as buttons, tables etc. Navicat is a program that manages databases are those that contain the information that anger generated by running the application.

Broadly speaking, this was what was used and what the application is to register employees, register products, modify and delete data.

# Capítulo 1 Descripción del proyecto.

1.1 Antecedentes

Materiales “PERZAN” fundada en 2008 ubicada en ese entonces sobre la carretera federal Fortín-Huatusco con avenida Amado Nervo en Chocamán, empezó como un pequeño negocio no contando con propiedad, al término de dos años se contaba con dos establecimientos, se decidió unir ambos para tener una obtener una propiedad y así seguir creciendo, se ubicó sobre la misma carretera pero con diferente avenida, ahora con avenida 12, al tener nueva dirección resulto más atrayente hacia clientes que nunca habían adquirido productos en esta empresa, al igual que se ha convertido en proveedora de diferentes ferreteras en la zona, como negocio de giro comercial y estando en el mismo por más de 6 años, nunca había tenido una aplicación la cual pudiera administrar y agilizar procesos dentro de la misma, contando con un jefe, un encargado y dos empleados, esta aplicación favorecerá en mucho ya que se podrá tener un registro de clientes, productos y ventas, lo cual para la empresa es muy importante dentro de su contabilidad llevar todos estos registros, al tener clientes registrados será de gran facilidad saber la dirección y teléfono del cliente así como la existencia de los productos que se requieren y así una mejor atención y entrega de producto solicitado, en la actualidad las tecnologías son de gran ayuda por lo tanto esta empresa debía mantenerse a la vanguardia para mayor eficacia y a su vez seguir a la altura de otras más en la región, con este proyecto se obtendrán resultados positivos en cuanto a mayor atención a los clientes.

1.2 Descripción de Problemática.

La problemática que se genera con respecto a la aplicación de escritorio, es la falta de información actualizada, que los usuarios necesitan para realizar sus diferentes procesos de registros en tiempo real, para el ingreso de las ventas realizadas, actualización de productos e información de proveedores.

Se presentaron algunas inconsistencias en el diseño de la aplicación, así como también entre la funcionalidad y el manual de usuario.

Algunos de los problemas presentados a nivel código y base de datos son la clase login, no hacia el acceso a la clase administrador lo que se hizo fue cambiar los métodos y especificar la lógica de una mejor manera.

El botón cancelar de la clase login al ser seleccionado mandaba la vista de administrador lo que hizo que se agregara un parámetro al método en el main en el cual se lleva a la vista administrador.

Las tablas de la vista administrador no se llenaban completas, para esto se cambiaron los nombres de cada columna referenciada al objeto que contienen.

Los botones se activaban y desactivaban en desorden, entonces lo que se hizo fue revisar los métodos que activan y desactivan los botones y establecer un orden para que se el proceso sea mejor.

Los textfield no se vaciaban cuando se agregaba un registro entonces se creó un método que vacíe los text field y además los desactive.

Cuando se agregaba un registro la tabla en la vista no se actualizaba lo que se hizo fue cambiar el modo de llenado de la tabla con otro método.

Las funciones de base de datos retornaban un mensaje de redundancia de datos: solución crear las funciones en base de datos con parámetros que se llamen diferente a los campos de las tablas para así evitar redundancias.

Las funciones de base de datos no retornaba algún valor como solución se revisaron las funciones erróneas y se encontró que los datos que se retornaban no eran iguales a los que solicitaba la consulta.

La tabla detalle\_venta y detalle\_compra no se encontraban en la base de datos, aun cuando ahí se observaba la segunda palabra del nombre de cada una de las tablas estaba escrita en mayúscula y Postgresql no reconocía el nombre de las mismas, así que se cambió el nombre a detalle\_venta y detalle\_compra.

1.3 Objetivo General

Mejorar e implementar en la aplicación de escritorio ciertas funcionalidades que permitan al usuario manipular los datos con más facilidad y eficacia, de acuerdo a las actividades que realizan cada uno de los empleados, asi como la creación de reportes y crear respaldos de la base de datos.

1.4 Objetivos Específicos

Utilizar Postgresql como gestor de base de datos.

Añadir a cada ventana las actividades que hace cada empleado.

Modificar la interfaz gráfica mediante el uso de JavaFX.

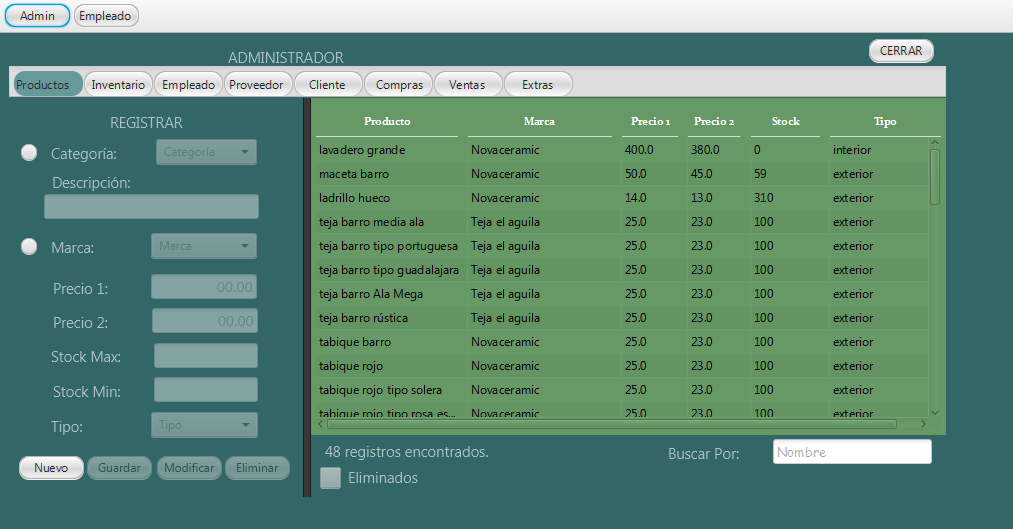
Agilizar el servicio de venta.

Manipular la información dentro de la base de datos y utilizar el código correctamente para que el tiempo de respuesta no sea demasiado alto en una búsqueda de información.

1.5 Justificación

La solución que nosotros presentamos es haciendo unos cambios a la interfaz para que el usuario pueda manipular toda la información que él podría necesitar en su momento de acuerdo a la jerarquía que presente cada empleado en la empresa, tal es el caso de agregar la ventana de proveedores donde el administrador, puede introducir datos, modificar, eliminar y consultar información. Así mismo sucede con la ventana de productos donde se manejan los productos por marcar, categoría, precio proveedor, precio público, y un stock máximo y mínimo. Se agregó una ventana llamada características donde se enlistan los productos y desde las mismas tablas se puede manipular la información como el usuario desee.

La aplicación desarrollada permite al usuario que hace uso de ella realizar de una manera intuitiva y sencilla, el manejo de la información de los productos, entradas y salidas y clientes frecuentes. De esta forma el personal, mediante la herramienta tecnológica, tiene total capacidad para introducir y consultar la información, siendo el mismo el principal conductor de las ideas que desea implementar.

Se busca que sea una aplicación completa que beneficie al administrador como a sus empleados con el manejo de información, y por qué no al cliente, así ya no tendría que esperar tanto tiempo en que le busquen si hay en existencia algún material.

Imagen

1.6 Alcances

La aplicación permitirá manipular información de productos, empleados, proveedores, compras y ventas que se manejan dentro de la empresa.

Mediante una base de datos con conexión a Postgresql se realiza la autentificación del usuario con respecto al cargo que ocupa, ya sea empleado o administrador tendrá acceso a la información que solicite.

El tiempo de respuesta al buscar información solicitada debe ser ágil, para que no sea lenta la aplicación.

**1.7 Limitaciones**

La aplicación solo puede registrar datos en la misma, no tendrá la función de imprimir ticket de compra.

La aplicación no maneja registros para generar facturas.

No muestra un detalle venta, por una venta específica, igual una compra.

# Capítulo 2 Herramientas

Para la creación y modificación de la aplicación se utilizaron los siguientes programas:

* Eclipse: Es una potente y completa plataforma de Programación, desarrollo y compilación de elementos tan variados como sitios web, programas en C++ o aplicaciones Java.

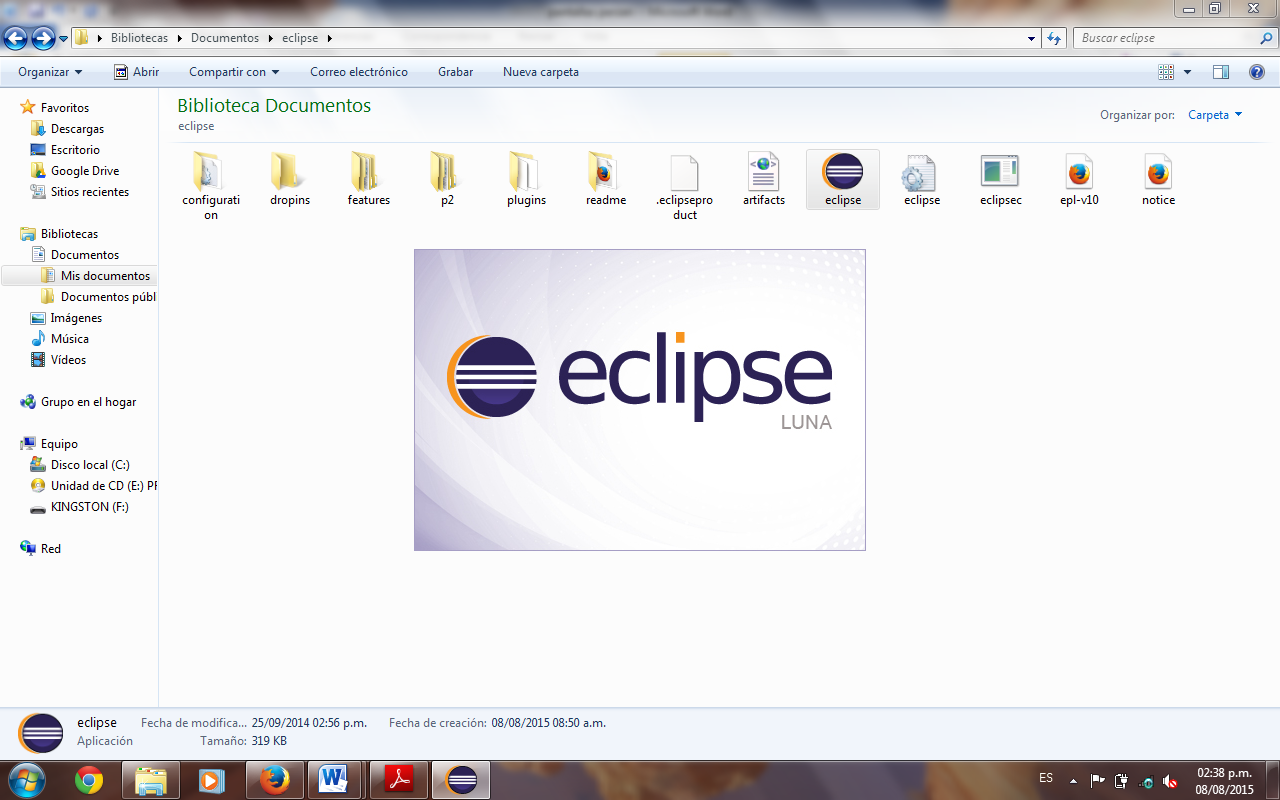


Imagen 2

* JavaFX: es una plataforma de software que permite crear y ejecutar aplicaciones web que pueden ejecutarse en una gran variedad de dispositivos.



Imagen

* PostgreSQL: Es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD y con su código fuente disponible libremente. Es el sistema de gestión de bases de datos de código abierto más potente del mercado y en sus últimas versiones no tiene nada que envidiarle a otras bases de datos comerciales.

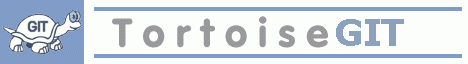
Imagen

* Github: es una plataforma de desarrollo colaborativo de software para alojar proyectos utilizando el sistema de control de versiones Git. El código se almacena de forma pública, aunque también se puede hacer de forma privada, creando una cuenta de pago.



Imagen

* TortoiseGit es un cliente de software de control de versiones Git, implementado como una extensión de shell de Windows.



Imagen

* Computadora portátil: Es un ordenador personal móvil o transportable, que pesa normalmente entre 1 y 3 kg. Los ordenadores portátiles son capaces de realizar la mayor parte de las tareas que realizan los ordenadores de escritorio, también llamados "de torre", con similar capacidad y con la ventaja de su peso y tamaño reducidos; ello sumado también a que tienen la capacidad de operar por un período determinado sin estar conectadas a una red eléctrica.



Imagen

* JasperReports: Es una librería de creación de [informes](https://es.wikipedia.org/wiki/Informe) que tiene la habilidad de entregar contenido enriquecido al [monitor](https://es.wikipedia.org/wiki/Monitor_de_computadora), a la [impresora](https://es.wikipedia.org/wiki/Impresora) o a ficheros [PDF](https://es.wikipedia.org/wiki/PDF), [HTML](https://es.wikipedia.org/wiki/HTML), [XLS](https://es.wikipedia.org/wiki/XLS), [CSV](https://es.wikipedia.org/wiki/CSV) y [XML](https://es.wikipedia.org/wiki/XML).



Imagen 8

* IReport: es un constructor / diseñador de informes visual, poderoso, intuitivo y fácil de usar para JasperReports escrito en Java. Este instrumento permite que los usuarios corrijan visualmente informes complejos con cartas, imágenes, subinformes, etc.



Imagen 9

# Capítulo 3 Marco teórico

3.1 JavaFX

Amplía la potencia de Java permitiendo a los desarrolladores utilizar cualquier biblioteca de Java en aplicaciones JavaFX. De esta forma, los desarrolladores pueden ampliar sus capacidades en Java y utilizar la tecnología de presentación que JavaFX proporciona para crear atractivo visual. Como usuario, podrá ejecutar aplicaciones JavaFX en un explorador o arrastrarlos y soltarlos en el escritorio.

Características principales de JavaFX:

* Permite a los desarrolladores integrar gráficos vectoriales, animación, sonido y activos web de vídeo en una aplicación interactiva, completa y atractiva
* Amplía la tecnología Java permitiendo el uso de cualquier biblioteca de Java en una aplicación JavaFX
* Permite mantener un eficaz flujo de trabajo entre diseñador y desarrollador en el que los diseñadores pueden trabajar en las herramientas que deseen mientras colaboran con los desarrolladores

3.2 Scene Builder

Escena JavaFX Builder está liberado bajo la licencia BSD Oracle. Se proporciona el código fuente y se desarrolló a través del Proyecto OpenJFX en la Comunidad OpenJDK. Por favor consulte la OpenJFX Wiki para obtener instrucciones detalladas sobre cómo descargar y construir su código fuente. Tenga en cuenta que el código fuente de esta aplicación se puede encontrar en el "apps / scenebuilder" directorio.

3.3 Eclipse

Eclipse es un programa informático compuesto por un conjunto de herramientas de programación de código abierto multiplataforma para desarrollar lo que el proyecto llama "Aplicaciones de Cliente Enriquecido", opuesto a las aplicaciones "Cliente-liviano" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado *Java Development Toolkit* (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). Sin embargo, también se puede usar para otros tipos de aplicaciones cliente, como BitTorrent o Azureus.

3.4 Navicat for MySQL

Es una solución ideal para la administración y el desarrollo de MySQL / MariaDB. Permite la conexión a bases de datos MySQL y MariaDB simultáneamente desde una sola aplicación. Este frontal con todo incluido ofrece una intuitiva y potente interfaz gráfica de gran alcance para la gestión, el desarrollo y el mantenimiento de las bases de datos.

3.5 Patrón MVC

Es un patrón de arquitectura de las aplicaciones software

* Separa la lógica de negocio de la interfaz de usuario
* Facilita la evolución por separado de ambos aspectos
* Incrementa reutilización y flexibilidad.

3.6 Patrón Singleton

El patrón Singleton garantiza que una clase sólo tenga una instancia y proporciona un punto de acceso global a ésta instancia.

El funcionamiento de este patrón es muy sencillo y podría reducirse a los siguientes conceptos:

* Ocultar el constructor de la clase Singleton, para que los clientes no puedan crear instancias.
* Declarar en la clase Singleton una variable miembro privada que contenga la referencia a la instancia única que queremos gestionar.
* Proveer en la clase Singleton una función o propiedad que brinde acceso a la única instancia gestionada por el Singleton. Los clientes acceden a la instancia a través de esta función o propiedad.
* Estas reglas se cumplen en todas las implementaciones del Singleton, independientemente de los recaudos que deban tomarse para soportar la correcta ejecución en entornos multihilo.

3.7 JasperReports

Es software libre que nos permite crear informes, donde los ficheros se definen en un xml para posteriormente ser compilado por las librerías JasperReports (.jasper), es competencia de Crystal reports en los usuarios Windows.

JasperReports está compuesto por conjuntos de librería java, una de las ventajas que tiene JasperReports es [la comunidad](http://community.jaspersoft.com/project/jasperreports-library). Podemos usarla en aplicaciones de Java como lo son J2EE y también en aplicaciones web. Dentro de este tema nos encontramos con iReportes una herramienta de diseño visual para JasperReports, te ayudara a diseñar reportes visualmente en poco tiempo.

3.8 IReport

Es un constructor / diseñador de informes visual, poderoso, intuitivo y fácil de usar para JasperReports escrito en Java. Este instrumento permite que los usuarios corrijan visualmente informes complejos con cartas, imágenes, subinformes, etc. iReport está además integrado [con JFreeChart](http://www.jfree.org/jfreechart/), una de las bibliotecas gráficas OpenSource más difundida para Java. Los datos para imprimir pueden ser recuperados por varios caminos incluso múltiples uniones JDBC, TableModels, JavaBeans, XML, etc.

La lista siguiente describe algunas de las características importantes de iReport:

* 100% escrito en JAVA y además OPENSOURCE y gratuito.
* Maneja el 98% de las etiquetas de JasperReports
* Permite diseñar con sus propias herramientas: rectángulos, líneas, elipses, campos de los textfields, cartas, subreports (subreportes).
* Soporta internacionalización nativamente.
* Browser de la estructura del documento.
* Recopilador y exportador integrados.
* Soporta JDBC.
* Soporta JavaBeans como orígenes de datos (éstos deben implementar la interface JRDataSource).
* Incluye Wizard’s (asistentes) para crear automáticamente informes.
* Tiene asistentes para generar los subreportes
* Tiene asistentes para las plantillas.
* Facilidad de instalación.

3.9 Definición de roles

Líder del Proyecto: Asigna los recursos, gestiona las prioridades, coordina las interacciones con el cliente y mantiene al equipo del proyecto enfocado en los objetivos. Establece un conjunto de prácticas que aseguran la integridad y calidad del proyecto. Se encarga de supervisar la planificación y control de proyecto.

Analista-Diseñador: Captura, especificación y validación de requisitos interactuando con el cliente mediante entrevistas. Elabora el modelo de análisis y diseño. Colaboración en la elaboración de pruebas funcionales y el modelo de datos.

Programador: Construcción de prototipos. Colaboración en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario.

3.10 Descripción del producto

La aplicación PERZAN es creada para satisfacer la necesidades de la empresa PERZAN, la cual es llevar a cabo el control de compras y ventas, se dispone de dos tipos de usuarios, un administrador, el cual se encarga del registro de productos y empleados que ingresan a la empresa, y un usuario empleado, el cual se encargara de registrar ventas y compras de la empresa.

Para cada usuario hay una interface distinta, cada una con permisos específicos para cada tipo de usuario. Se mostrara un inicio de sesión en él se validara el tipo de usuario que utilizara el sistema.

Se tiene conexión a base de datos en PostgreSQL, la aplicación permite guardar, modificar, buscar y eliminar datos, se establece un límite mínimo y un límite máximo (stock) para los productos con los que trabaja la empresa.

Con la aplicación PERZAN se le da solución al problema de no tener un sistema más rápido y que permita llevar un registro más accesible de la empresa.

# Capítulo 4 Metodología de Desarrollo

4.1 MoProSoft

MoProSoft es un modelo de procesos para la industria de software nacional, que fomenta la estandarización de su operación a través de la incorporación de las mejores prácticas en gestión e ingeniería de software. La adopción del modelo permite elevar la capacidad de las organizaciones que desarrollan o mantienen software para ofrecer servicios con calidad y alcanzar niveles internacionales de competitividad. Es también aplicable en áreas internas de desarrollo de software de las empresas de diversos giros.

4.2 Características

* Es específico para el desarrollo y mantenimiento de software.
* Es sencillo de entender y adoptar.
* Facilita el cumplimiento de los requisitos de otros modelos como ISO 9000:2000, CMM y CMMI.
* Se enfoca a procesos.
* Se le considera práctico en su aplicación, principalmente en organizaciones pequeñas, con bajos niveles de madurez.
* Comprende un documento de menos de 200 páginas que, al compararlo con otros modelos y estándares, lo hace bastante práctico.
* Resulta acorde con la estructura de las organizaciones mexicanas de la industria de software.
* Está orientado a mejorar los procesos, para contribuir a los objetivos de negocio, y no simplemente ser un marco de referencia o certificación.
* Tiene un bajo costo, tanto para su adopción como para su evaluación.

4.2 ¿Para qué sirve?

* Mejora la calidad del software producido por la empresa que adopta el modelo.
* Eleva la capacidad de las organizaciones para ofrecer servicios con calidad y alcanzar niveles internacionales de competitividad.
* Integra todos los procesos de la organización y mantiene la alineación con los objetivos estratégicos.
* Inicia el camino a la adopción de los modelos ISO 9000 o CMMI.
* Sirve para implantar un programa de mejora continua.
* Permite reconocer a las organizaciones mexicanas por su nivel de madurez de procesos.
* Facilita la selección de proveedores.
* Permite obtener acceso a las prácticas de ingeniería de software de clase mundial.

4.3 Estructura de MoProSoft

El modelo pretende apoyar a las organizaciones en la estandarización de sus prácticas, en la evaluación de su efectividad y en la integración de la mejora continua. Sintetiza las mejores prácticas en un conjunto pequeño de procesos que abarcan las responsabilidades asociadas a la estructura de una organización que son: la Alta Dirección, Gestión y Operación.

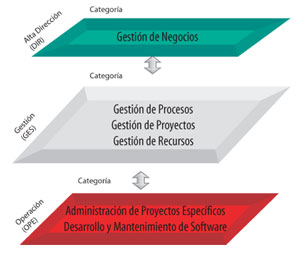


Imagen 10

4.4 Beneficios

* Al tener prácticas integradas, que abarcan desde la gestión de negocio hasta el desarrollo y mantenimiento de software, las empresas logran un mayor control sobre su desempeño en el mercado.
* El costo de incorporación del nuevo personal podría disminuir si se enfocan a la educación y a la capacitación sobre un modelo único.
* Las empresas pequeñas, al seguir procesos similares, podrían asociarse con mayor facilidad para afrontar proyectos de mayor envergadura.
* La exportación de servicios de software de las empresas mexicanas será más factible, incluso se podría disminuir la necesidad de la intermediación de las empresas trasnacionales, gracias a que MoProSoft considera las prácticas reconocidas en el ámbito internacional.

La implantación de MoProSoft no demanda la incorporación de personal especializado en las empresas, únicamente requiere de una adecuada capacitación del personal existente.

# Capítulo 5 Conclusiones

Conclusión

Se realizaron cambios a la aplicación de escritorio, ya que en un principio era sencilla en sus funciones.

Actualmente la aplicación cuenta con dos usuarios, que es el administrador y el empleado, el panel de administrador tiene varias ventanas como la de productos, compras, ventas,

Se realizó una aplicación que permite entrar en un panel de acuerdo al cargo que se tenga (Administrador o empleado). En el panel de administrador se pueden registrar a los empleados y los productos que existan, también se pueden ver las ventas realizadas por cada uno de los empleados. Los datos de los empleados pueden ser modificados cada que se requiera o eliminarlos.

En el panel de empleado se muestra una lista de los productos existentes en venta y también cuenta con la parte para realizar ventas, aún faltan algunos detalles que en el siguiente cuatrimestre se podrán mejorar.

Se ocuparon programas como Eclipse, Scene Builder 2.0, y Navicat.

Para realizar la aplicación nos basamos en la metodología XP ya que el proyecto es pequeño.

# Bibliografía

Eckel, B. (2007). *Piensa Java.* Prentice Hall.

Elmasri, R. N. ( 2008). *Fundamentos de Sistemas de Bases de Datos.* Pearson Education.

Silberschatz, A., Korth, H.F., Sudarshan, S., Fun*d*amentos *d*e Bases *d*e *D*atos, 5ª edición, Madrid, 2006

<http://www.oracle.com/technetwork/es/java/javafx/overview/index.html>

<http://www.coderanch.com/how-to/java/Java-FAQ>