****

**ESCUELA SUPERIOR POLITECNICA DEL LITORAL**

**PROYECTO DE APLICACIONES DISTRIBUIDAS**

**Tema: MODULO DE FACTURACIÓN**

**Integrantes:**

* **JOSELYN MINDIOLA**

Índice

[1. Reseña de la Empresa 1](#_Toc422092152)

[2. Solución Propuesta 1](#_Toc422092153)

[3. Alcances 1](#_Toc422092154)

[4. Plataforma de Desarrollo 2](#_Toc422092155)

[5. Servicios por capa 2](#_Toc422092156)

[6. Servicios con cada aplicación 4](#_Toc422092157)

[7. Estructuras 4](#_Toc422092158)

[8. Diagrama de clases de seguridad 5](#_Toc422092159)

[9. Diagrama de clases de Facturación 6](#_Toc422092160)

# Reseña de la Empresa



La empresa “Market Sport” se dedica a la venta de diferentes producto al por menor, ya que ofrece bienes de consumo entre los que se encuentran zapatos, pelotas, camisetas y gorras. De acuerdo a las respectivas investigaciones y levantamiento de información de la empresa, hemos encontrado que la problemática se encuentra en la parte de facturación, esto se debe a que la empresa no cuenta con un sistema de información y esto causa robo de productos, faltante de dinero, es decir, mala organización en las actividades de cierre de caja.

# Solución Propuesta

Se propone el desarrollo de un sistema de facturación para automatizar y simplificar los procesos de facturación, mejorar las actividades de cierre y apertura de caja  y a su vez facilitar la toma de decisiones en el negocio

# Alcances

* Actualizar el stock al incurrir una venta
* Generar factura
* Anulación de Factura
* Apertura de caja
* Cierre de caja
* Asientos contables respectivos
* Reporte en base a ventas:
* Mejor vendedor (Mensual)
* La mejor venta (Mensual)
* Producto con menor salida (Mensual)
* Anulaciones de factura (Mensual)

# Plataforma de Desarrollo

El sistema será desarrollado con el lenguaje de programación:

* PHP editor de texto Notepad ++
* XAMPP  servidor independiente de plataforma, software libre, el servidor web Apache y los intérpretes para lenguajes de script: PHP

Base de datos:

* MySQL (Workbench): Versión 6.3.3 Software Libre

# Servicios por capa

|  |
| --- |
| SEGURIDAD |

|  |
| --- |
| PRESENTACIÓN |

|  |
| --- |
| MENSAJERÍA DE ERRORES |

|  |
| --- |
| LOG OPERATIVO |

|  |
| --- |
| REGLAS DEL NEGOCIO |

|  |
| --- |
| DATOS |

|  |
| --- |
| AUDITORIA |

**Presentación:**

* Se encargara de proveer el entorno grafico
* Validaciones con respecto a las pantallas

**Mensajería de Errores:**

* Dara a conocer de los posibles mensajes que se presenten en la aplicación tanto de información, advertencia o error que pueda generar la aplicación.

**Log Operativo:**

* Registro de los procesos del sistema

**Auditoria:**

* Registro de procesos propensos a fraude que servirán de ayudada para el auditor

**Reglas del Negocio:**

* Se utilizara el Método de línea recta en una depreciación.
* Se revalorizara los Activos Fijos por un perito para venderlos después de darlos de baja en una venta de garaje.
* En la empresa habrá descuentos por liquidación.
* Solo se aceptaran dos usuarios por caja al día.
* Se pagara en efectivo y cheque
* Se utilizara el plan de cuenta con el que se manejaba la empresa
* La revisión de los productos recién comprados la realizara el departamento de inventario para saber si existirá alguna devolución.
* El sistema de compra se encargara de las compras de los productos y de los activos fijos de la empresa
* El sistema de Inventario será el único que se encargara de colocar código de los productos

**Datos:**

* Controlará el acceso a la base de datos.
* Realizara operaciones CRUD

# Servicios con cada aplicación

|  |  |  |  |
| --- | --- | --- | --- |
| SISTEMA | SERVICIO | PARAMETRO DE SALIDA | PARAMETRO DE ENTRADA |
| Seguridad | validación de usuario y contraseña | usuario y contraseña | 1: usuario correcto  0: usuario incorrecto  2: usuario no existe |
| Seguridad | asignación de menú dinámico | roles | menú dinámico |
| Inventario | Listado de productos a vender | Código de producto | 1: Descripción, precio  0: no existe producto |
| Inventario | Validación de stock | Código producto  Cantidad a vender | 1: no hay stock  0: si existe stock |
| Inventario | Actualización de stock | Código producto  Stock nuevo | 1: actualizado  0: no se actualizo |
| Contabilidad | Ajuste de cierre caja | Categoría producto  Código cuenta  Tipo cuenta | 1:ajuste correcto  0: ajuste invalido |

# Estructuras

|  |
| --- |
| Log\_Operativo |
| Id\_log\_operativo |
| Aplicación |
| Transacción |
| Proceso |
| Fecha\_proceso |
| Hora\_inicio |
| Hora\_fin |
| usuario |

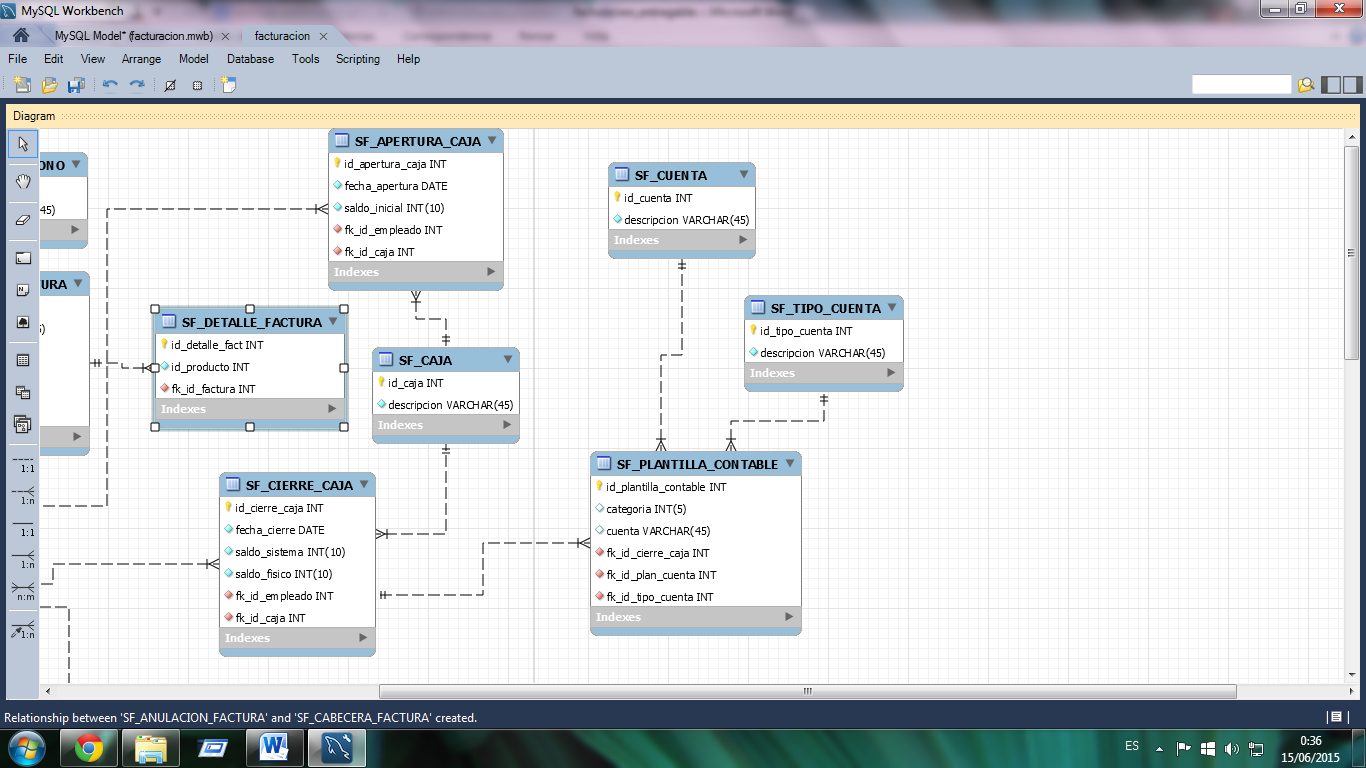
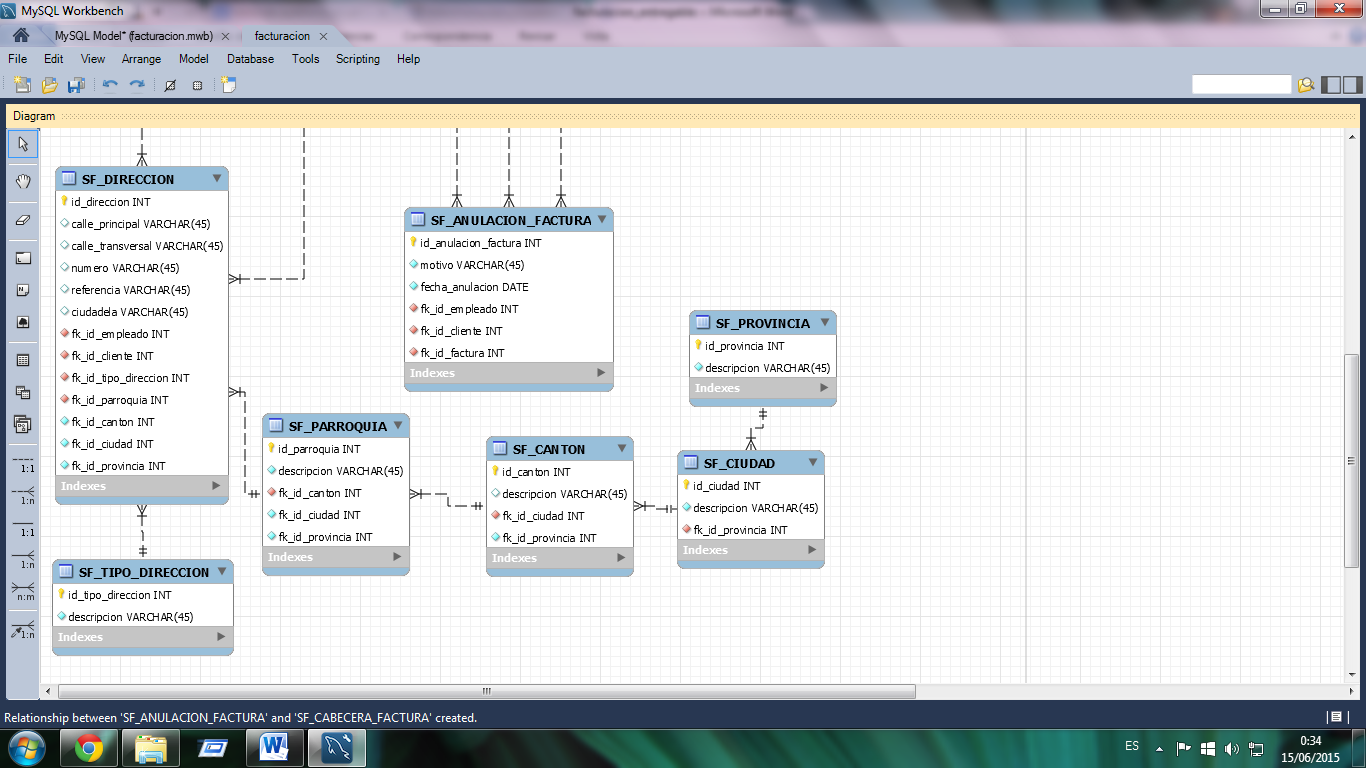
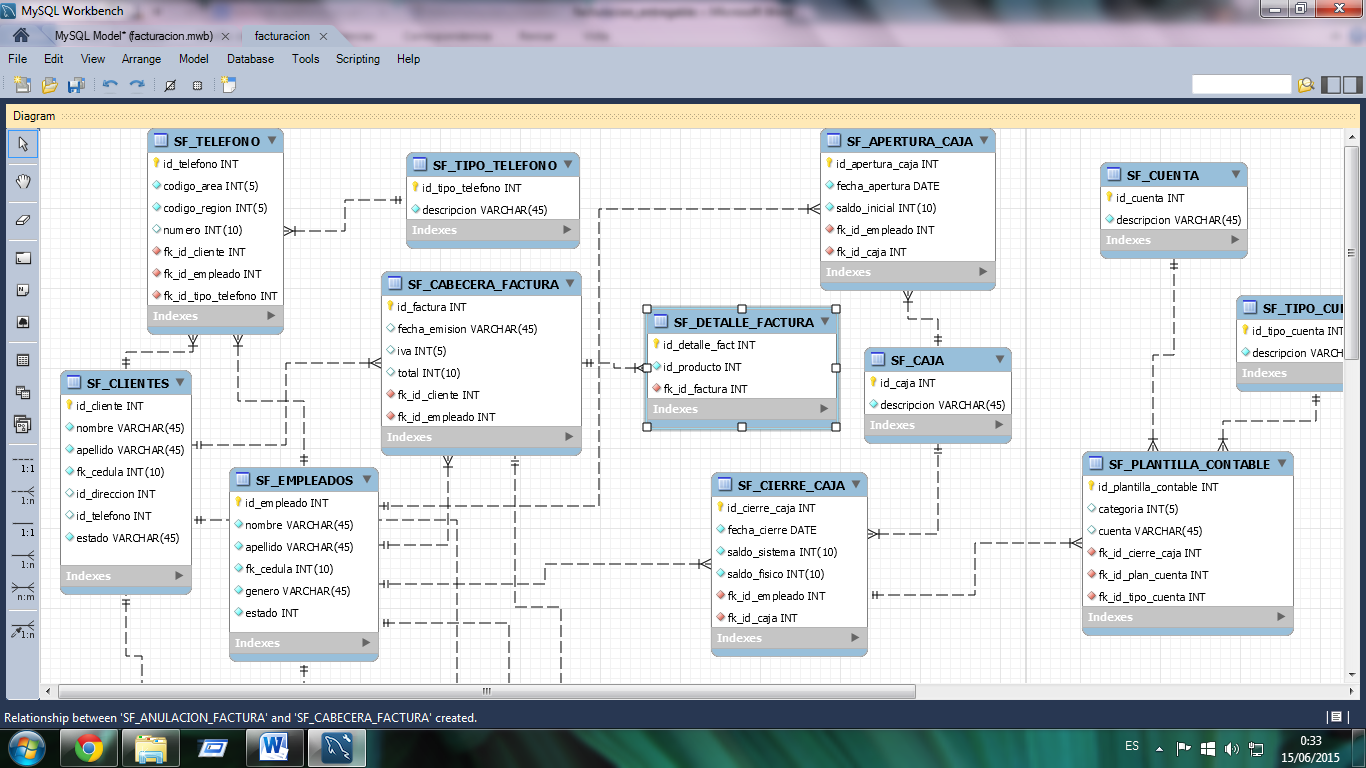
|  |
| --- |
| Auditoria |
| Id\_auditoria |
| Aplicación |
| Transacción |
| Proceso |
| Fecha\_proceso |
| Hora\_inicio |
| Hora\_fin |
| usuario |
| Anulación\_factura |
| Cierre\_caja |

# Diagrama de clases de seguridad

## Métodos Seguridad

* Usuario:
* Crear usuario();
* Consultar usuario();
* Eliminar usuario();
* Modificar usuario();
* Cambiar contraseña();
* Bloquear usuario();
* Rol:
* Crear rol();
* Eliminar rol();
* Modificar rol();
* Consultar rol();

# Diagrama de clases de Facturación



## Métodos Facturación

* Cabecera\_factura:
* Crear factura();
* Anular factura();
* Consultar factura();
* Calcular total();
* Imprimir factura();
* Cliente:
* Crear cliente();
* Eliminar cliente();
* Consultar cliente();
* Modificar cliente();
* Detalle\_factura:
* Calcular subtotal por línea();
* Cuenta:
* Agregar cuenta();
* Eliminar cuenta();