

CENTRO SUPERIOR TECNICO DE LA UNIVERSIDAD NACIONAL DE TRUJILLO - CESTUNT



PROYECTO:

MINIMARKET “*BOMBELBUR*”

PROFESOR:

ARTURO DIAZ PULIDO

INTEGRANTES:

- SEBASTIAN MARQUINA MARYCIELO
- MORENO CASTAÑEDA LUZ ANGIE
- LOYOLA RODRIGUEZ BRENDA EMPERATRIZ
- ÑIQUE JIMENEZ SERGIO JEAN PIERE

TRUJILLO – PERU
2014

INDICE

Capítulo 1 MARCO TEORICO

- 1.1 ¿Qué es JAVA?
- 1.2 ¿Qué es JDVC?
- 1.3 ¿Qué es NETBEANS?
- 1.4 ¿Qué es una base de datos?
- 1.5 ¿Qué es Workbench?
- 1.6 ¿Qué es Entidad – Relación?
- 1.7 ¿Qué es Modelo Relacional?

Capítulo 2 INGENIERIA DEL PROYECTO

- 2.1 Ciclo de Vida del Software
 - 2.1.1 Análisis
 - 2.1.2 División
 - 2.1.3 Implementación

Capítulo 3 CONCLUSION

Capítulo 4 LINKOGRAFIA

Capítulo 1 MARCO TEORICO

1.1 ¿Qué es JAVA?

Java es un lenguaje de programación y la primera plataforma informática creada por Sun Microsystems en 1995. Es la tecnología subyacente que permite el uso de programas punteros, como herramientas, juegos y aplicaciones de negocios.

Java es una tecnología que se usa para el desarrollo de aplicaciones que convierten a la Web en un elemento más interesante y útil. Java no es lo mismo que javascript, que se trata de una tecnología sencilla que se usa para crear páginas web y solamente se ejecuta en el explorador.

1.2 ¿Qué es JDBC?

JDBC es usado para enviar comandos SQL hacia una base de datos relacional, que puede ser Oracle, Infomix, SyBase, etc.

JDBC Establece una conexión con una BD para envía sentencias SQL y este mismo procesa los resultados.

1.3 ¿Qué es NETBEANS?

NetBeans es un entorno de desarrollo integrado libre, hecho principalmente para el lenguaje de programación Java.

La plataforma NetBeans permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos. Un módulo es un archivo Java que contiene clases de java escritas para interactuar con las APIs de NetBeans y un archivo especial (manifest file) que lo identifica como módulo. Las aplicaciones construidas a partir de módulos pueden ser extendidas agregándole nuevos módulos. Debido a que los módulos pueden ser desarrollados independientemente, las aplicaciones basadas en la plataforma NetBeans pueden ser extendidas fácilmente por otros desarrolladores de software.

1.4 ¿Qué es una Base de Datos?

Una base de datos es un "almacén" que permite guardar grandes cantidades de información de forma organizada, para luego poder usarlo fácilmente. Y estas bases de datos se pueden crear y diseñar usando diferentes sistemas de Gestor de Bases de Datos, conocido con las siglas SGBD, que es un software que actúa como interfaz, entre los datos almacenados y el usuario que desea manejar tales datos.

1.5 ¿Qué es WORKBENCH?

Es una herramienta visual de diseño de bases de datos que integra desarrollo de software, administración de bases de datos, diseño de bases de datos, creación y mantenimiento para el sistema de base de datos MySQL.

Con esta herramienta se puede elaborar una representación visual de las tablas, vistas, procedimientos almacenados y claves foráneas de la base de datos. Además, es capaz de sincronizar el modelo en desarrollo con la base de datos real. Se puede realizar una ingeniería directa e ingeniería inversa para exportar e importar el esquema de una base de datos ya existente el cual haya sido guardado o hecho copia de seguridad con MySQL Administrador.

1.6 ¿Qué es Entidad – Relación?

En esencia, el modelo entidad-relación (en adelante E-R), consiste en buscar las entidades que describan los objetos que intervienen en el problema y las relaciones entre esas entidades.

Todo esto se plasma en un esquema gráfico que tiene por objeto, por una parte, ayudar al programador durante la codificación y por otra, al usuario a comprender el problema y el funcionamiento del programa.

1.7 ¿Qué es Modelo Relacional?

El modelo relacional para la gestión de una base de datos es un modelo de datos basado en la lógica de predicados y en la teoría de conjuntos. Es el modelo más utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente.

En este modelo todos los datos son almacenados en relaciones, y como cada relación es un conjunto de datos, el orden en el que éstos se almacenen no tiene relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar por un usuario no experto. La información puede ser recuperada o almacenada por medio de consultas que ofrecen una amplia flexibilidad y poder para administrar la información.

Capítulo 2 INGENIERIA DEL PROYECTO

2.1 Ciclo de Vida del Software

2.1.1 Análisis

En este modelo todos los datos son almacenados en relaciones, y como cada relación es un conjunto de datos, el orden en el que éstos se almacenen no tiene relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar por un usuario no experto. La información puede ser recuperada o almacenada por medio de consultas que ofrecen una amplia flexibilidad y poder para administrar la información.

En el caso de nuestro proyecto de análisis comenzó con una primera entrevista con el gerente y dueño del minimarket. En esta primera entrevista se platicó acerca de todas las actividades que se realizan en el negocio. Llegando a concluir con estas entrevistas a definir cuáles serán los requerimientos que se implementaran en el software a desarrollar.

Los principales requerimientos funcionales son los siguientes:

- El proyecto debe girar en torno a la posibilidad de poder modelar Registro de clientes, productos, almacenes, proveedores, ventas y compras. Este software debe tener la posibilidad de reaccionar a las acciones de los usuarios que se encuentre utilizando el sistema, dado que se debe tener una cierta interacción y comunicación entre los usuarios y software en sí mismo. Estas acciones y eventos deben de quedar capturados y procesados por el software, para poder producir respuestas al usuario.
- Esta herramienta de software debe tener la facultad de desplegar y presentar diferentes actividades. Las actividades tendrán que ser desplegadas bajo petición de un usuario, y tendrán que tener comportamientos específicos y detallados con respecto a las acciones que el usuario tenga que realizar para poder resolver el problema que se presente a través de ella.
- La herramienta a implementar no tiene como objetivo el funcionar como un asistente sino como un alcance que se limita a presentar dentro del software las actividades que el

usuario utilizaran para el correcto funcionamiento del minimarket.

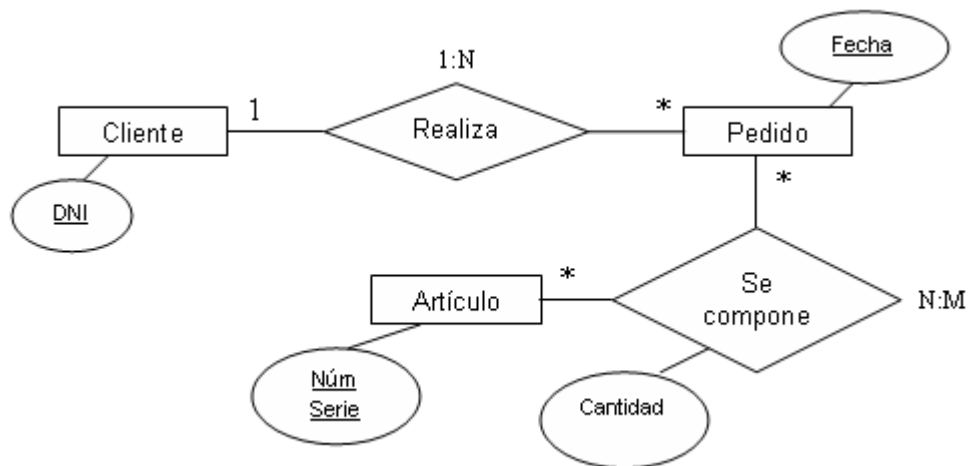
Los principales requerimientos no funcionales son los siguientes:

- Características de los usuarios: Se refiere a cuestiones como la edad, características físicas, experiencias previas, actitudes o intereses. Las características físicas son complementos irrelevantes.
- Problema o necesidad a atender: En el caso nuestro es modelar el mantenimiento y funcionalidad del negocio.

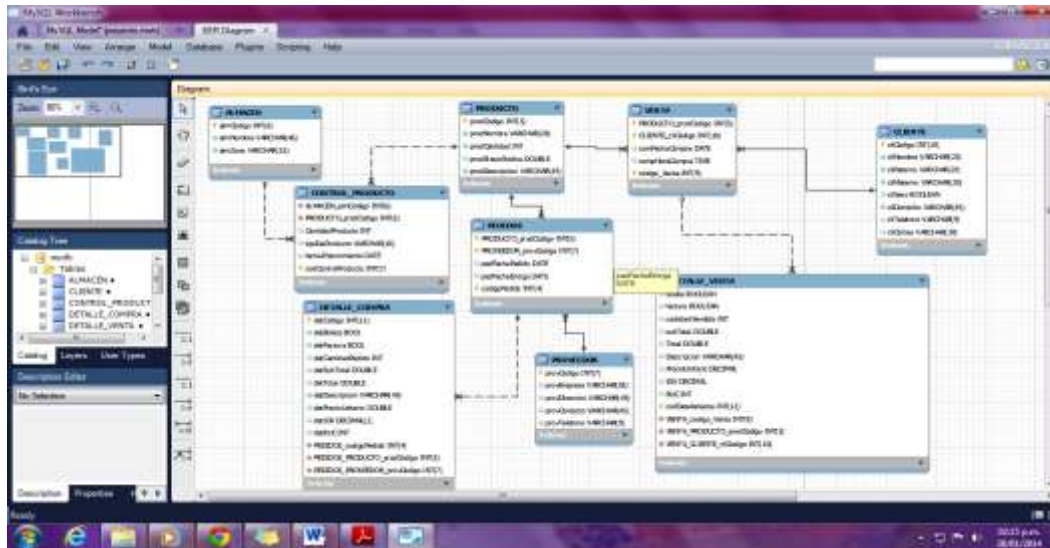
2.1.2 Diseño

El diseño del software desarrolla un modelo de instrumentación o implantación basado en los modelos conceptuales desarrollados durante el análisis del sistema. Implica diseñar la decisión sobre la distribución de datos y procesos. Generalmente aquí se produce el diseño de datos, un diseño de interfaz y un diseño procedimental.

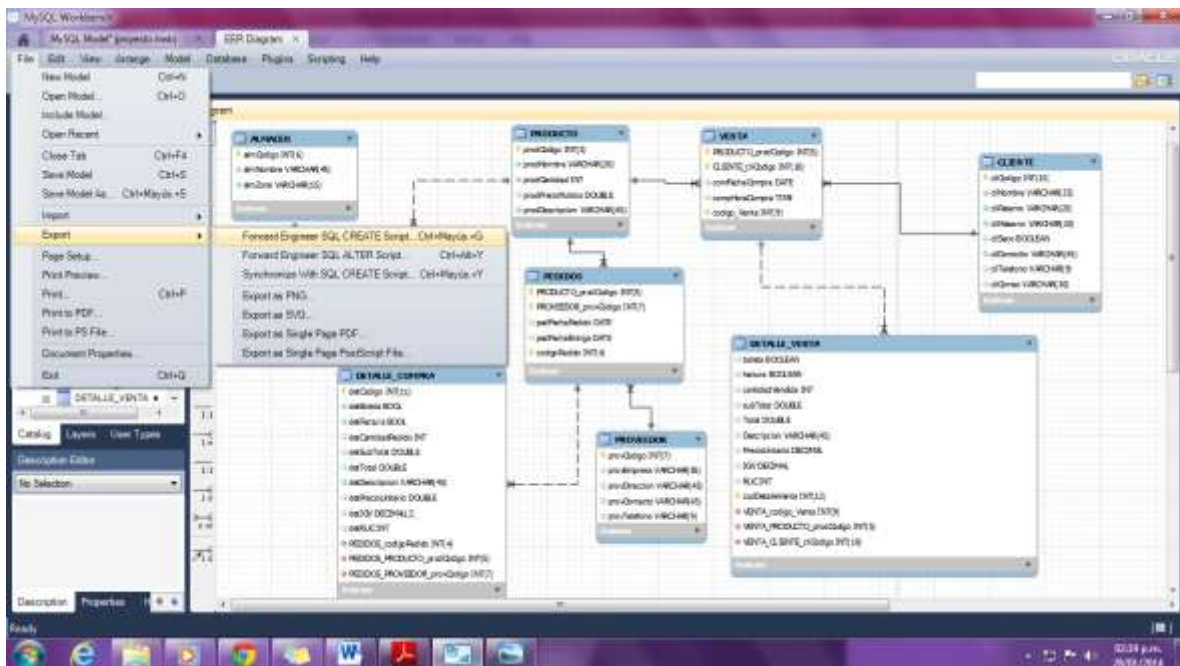
- Elaboración del Modelo Entidad-Relación: Identificación de entidades y atributos.

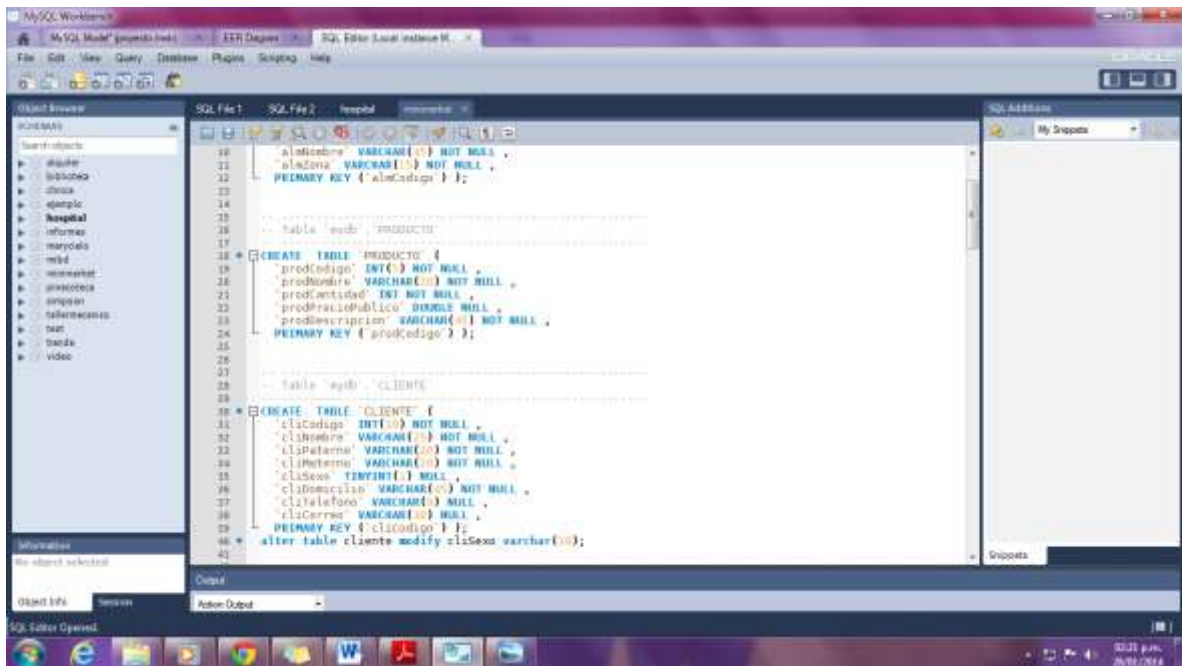


- Elaboración del Modelo Relacional: Los entidades se conviertan en tablas.

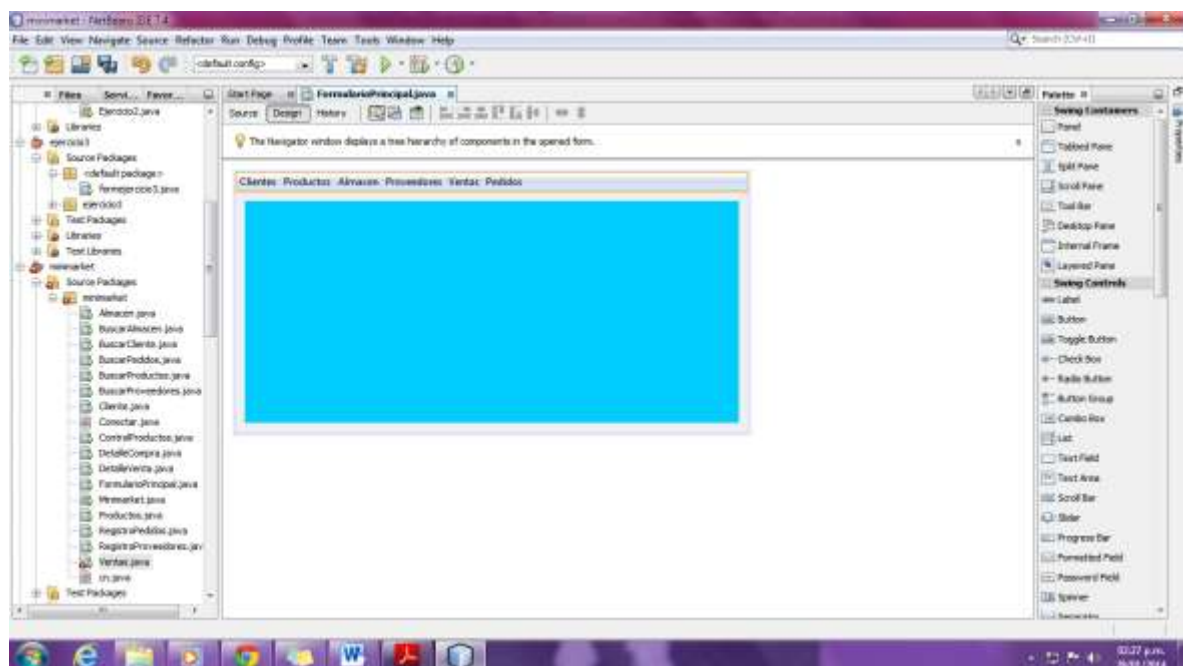


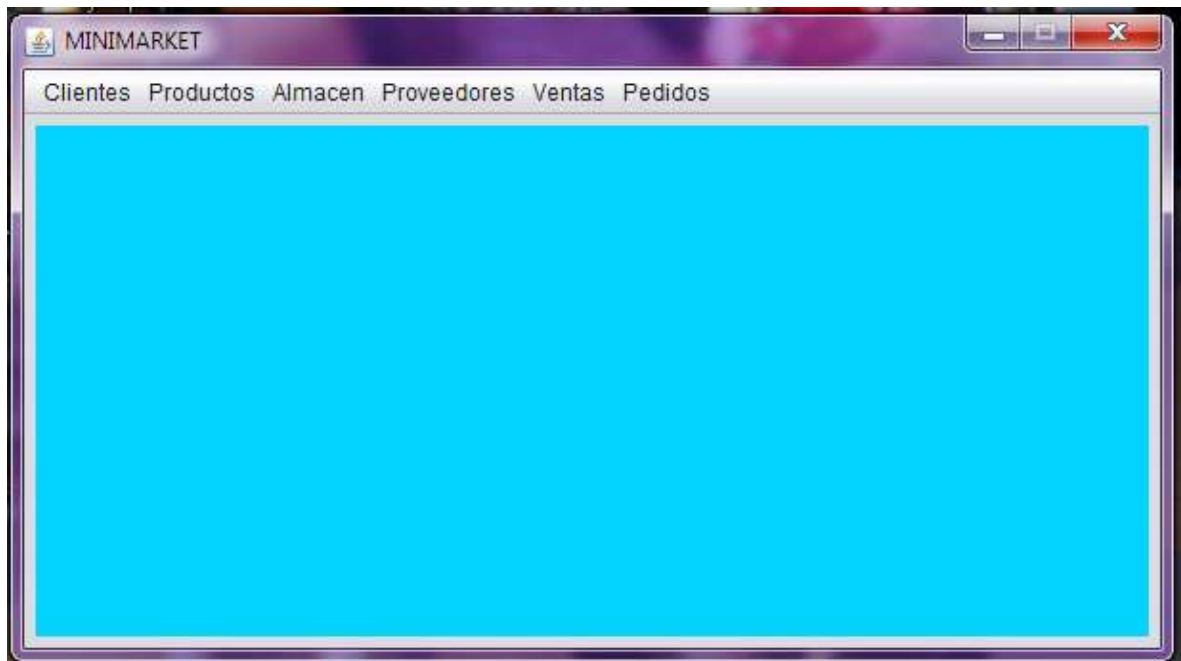
- Exportamos la base de datos y obtenemos el script del modelo relacional.



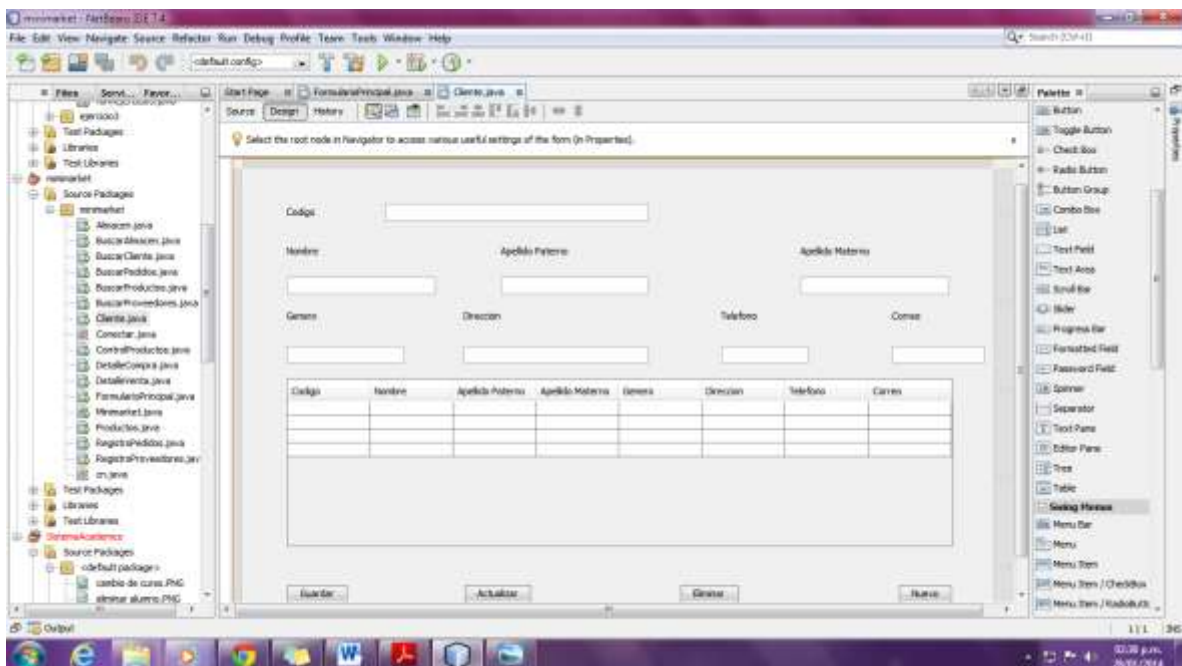


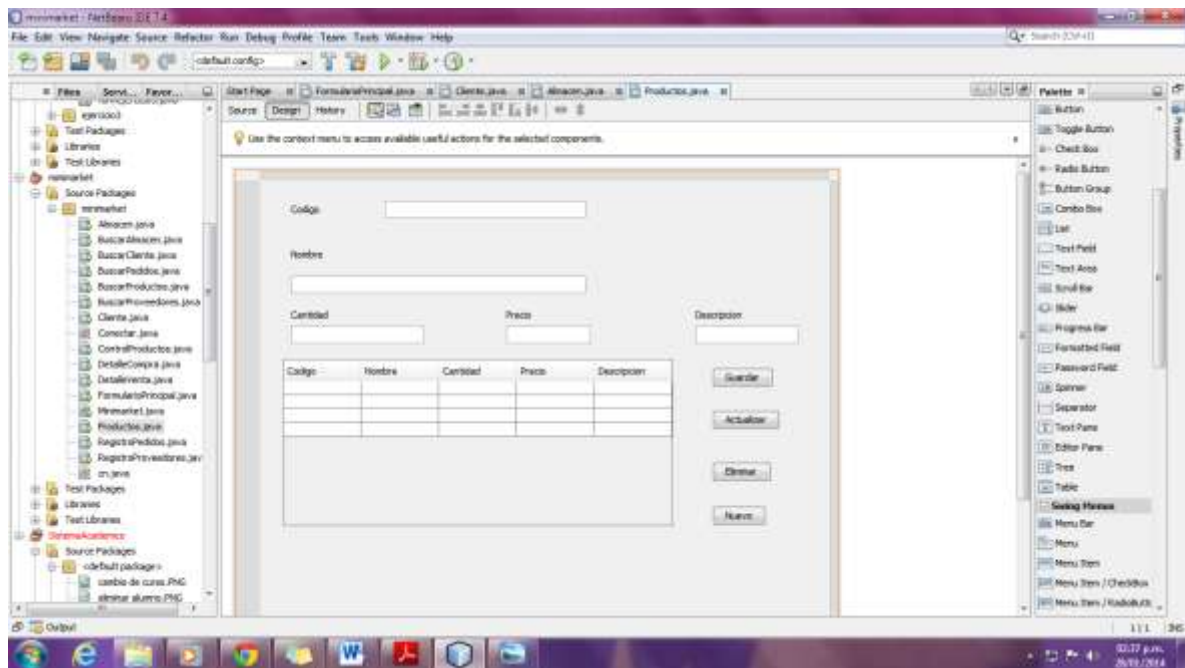
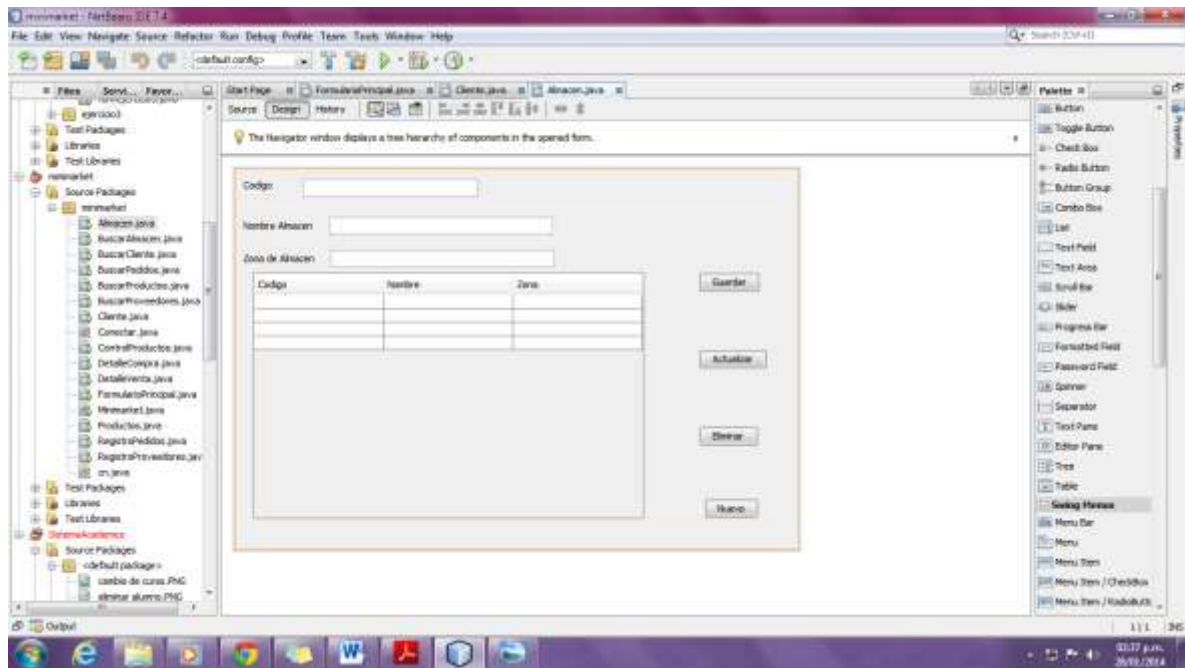
- Creamos la interfaz principal en NetBeans utilizando un Jpanel, Menu Bar y MenuItem:

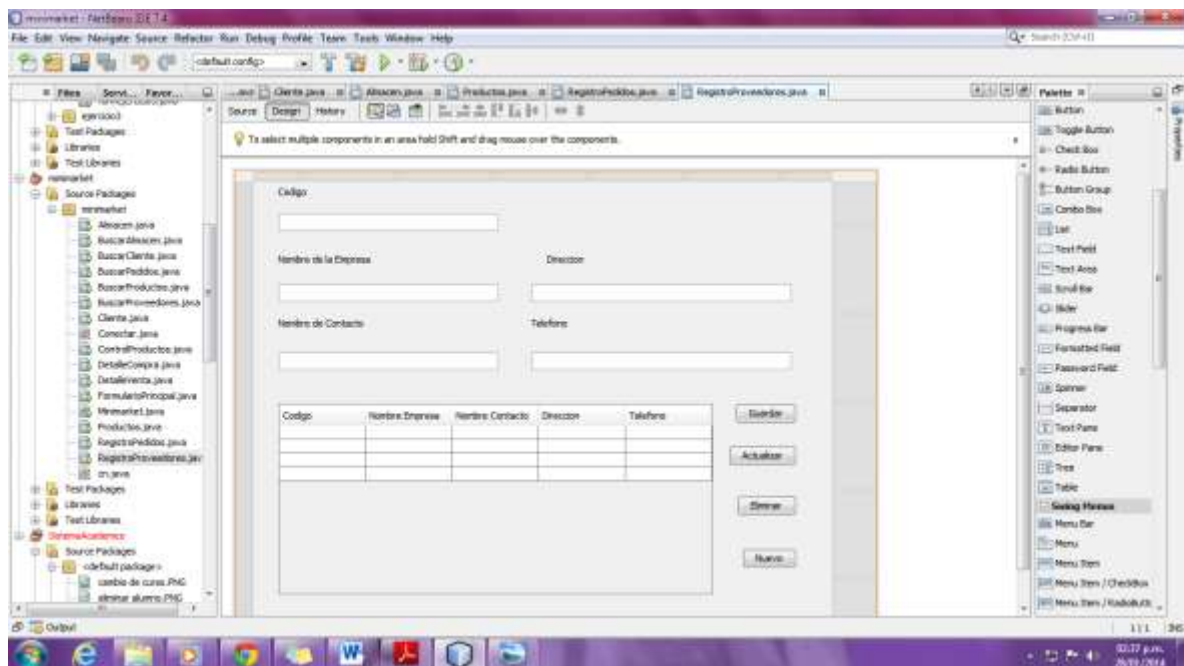
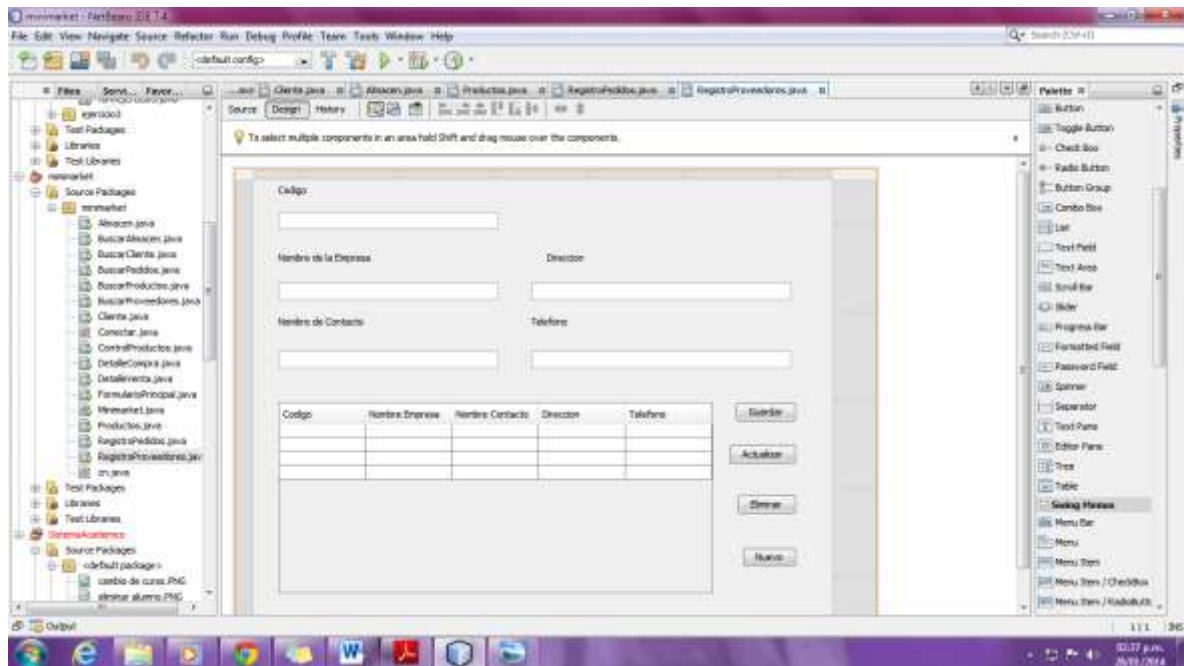


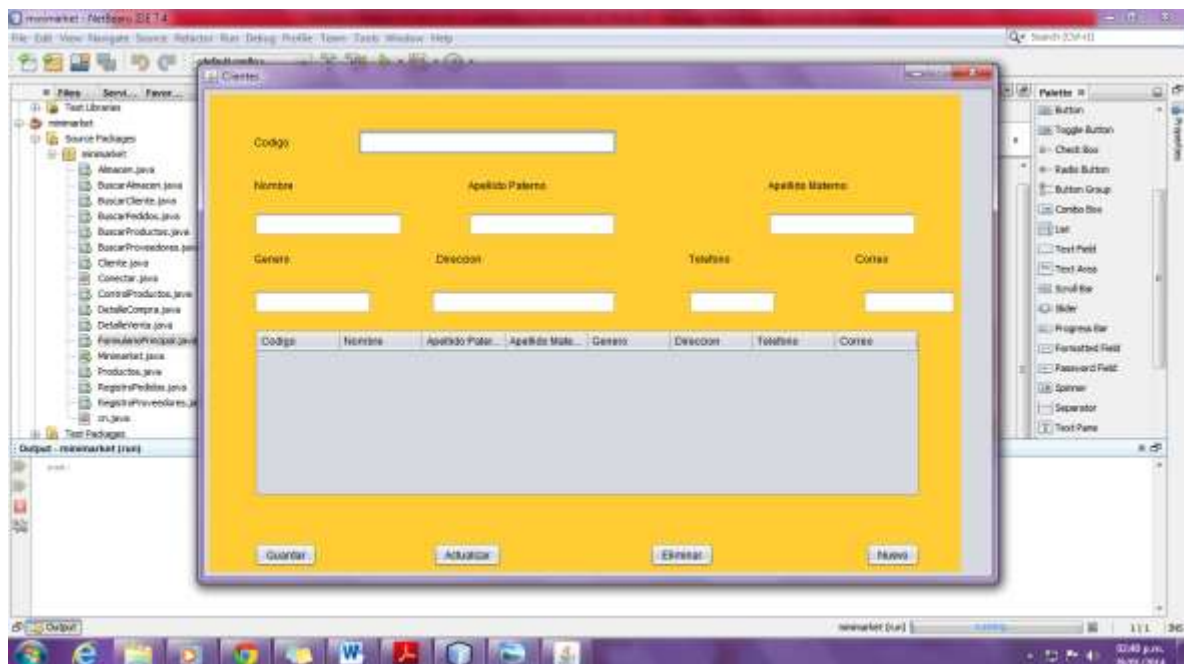
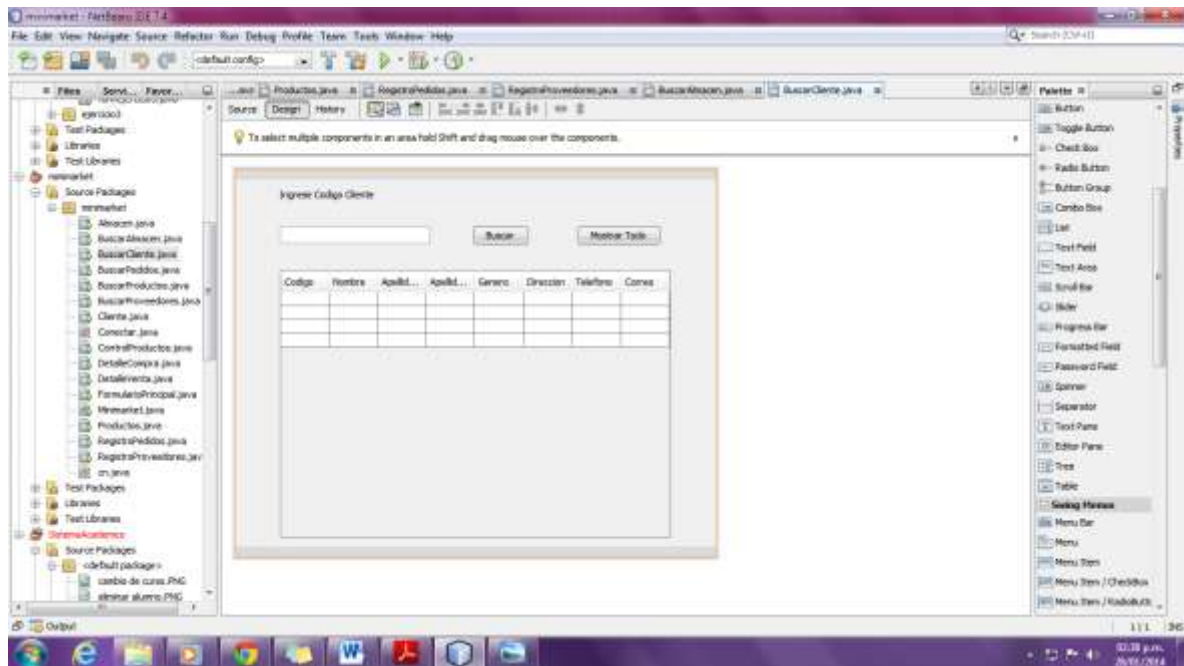


- Creación de las Interfaz de cada tabla empleando JPanel, Label, TextField, Table y Button.









Clientes

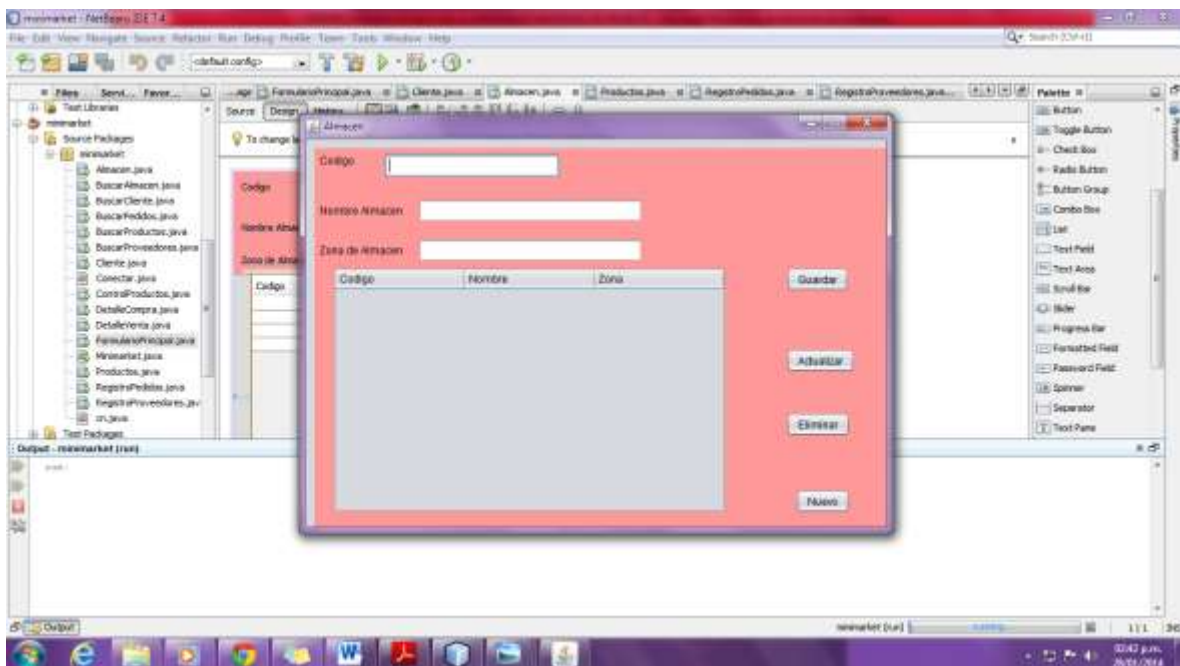
Codigo

Nombre Apellido Paterno Apellido Materno

Genero Direccion Telefono Correo

Codigo	Nombre	Apellido Pater...	Apellido Mate...	Genero	Direccion	Telefono	Correo

Guardar Actualizar Eliminar Nuevo



Almacen

Codigo

Nombre Almacen

Zona de Almacen

Codigo	Nombre	Zona
--------	--------	------

Guardar

Actualizar

Eliminar

Nuevo

Alumno

Codigo:

Nombre:

Cantidad: Precio: Descripcion:

Codigo	Nombre	Cantidad	Precio	Descripcion
--------	--------	----------	--------	-------------

Guardar

Actualizar

Eliminar

Nuevo

Buscar Cliente

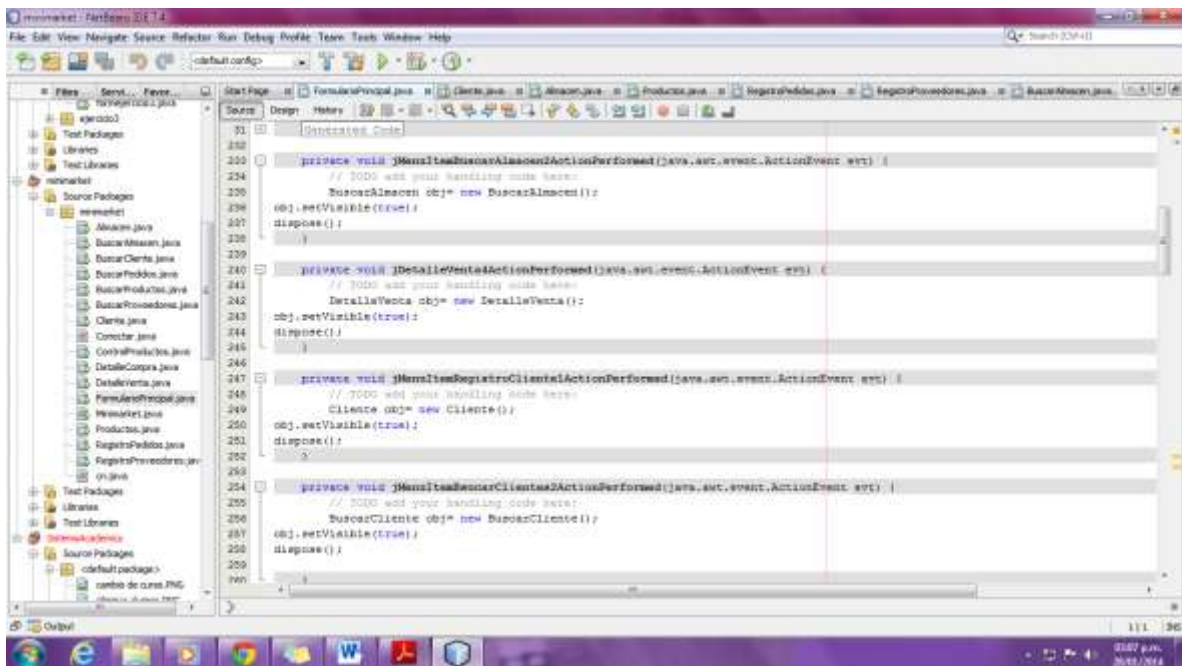
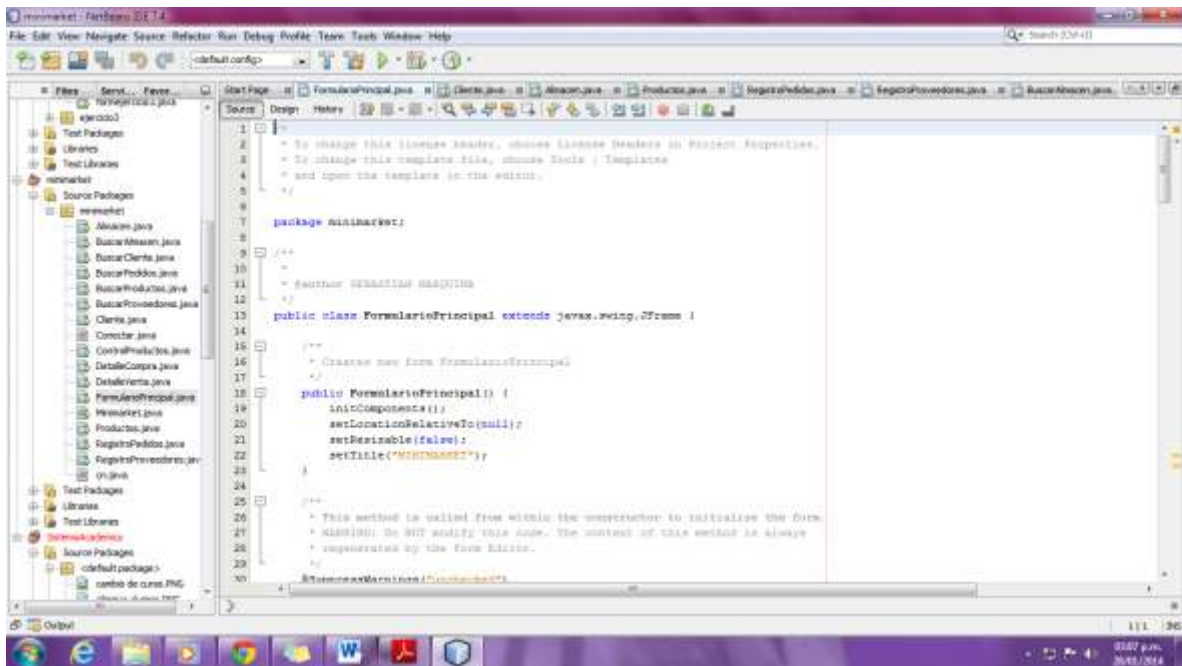
IngreseCodigo Cliente:

Buscar Mostrar Todo

Codigo	Nombre	Apellid...	Apellid...	Genero	Domic...	Telefo...	Correo
--------	--------	------------	------------	--------	----------	-----------	--------

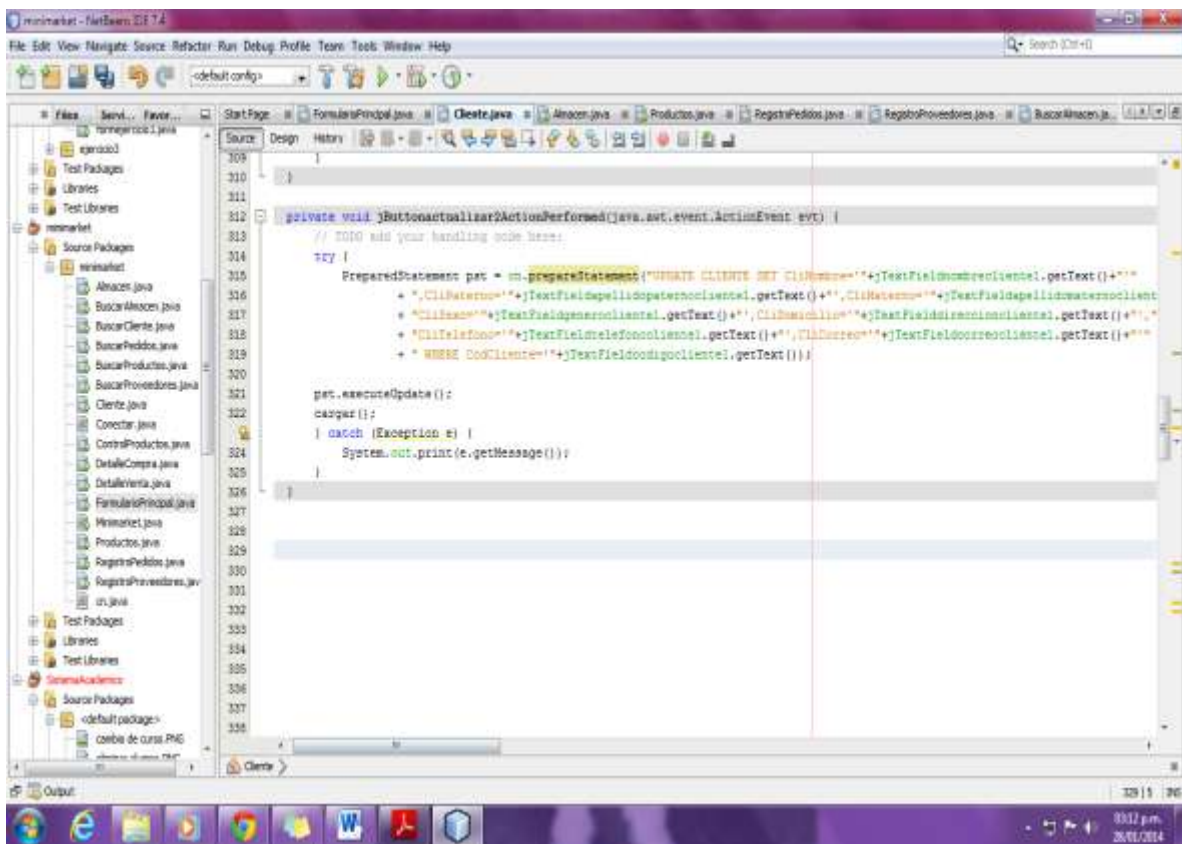
2.1.3 Implementación

Se procede a programar o implementar los diseños especificados en el modelo de diseño. A Cada interfaz creada se le implementa su código para dar respuesta a lo que el usuario le está pidiendo al utilizar el software. Aquí se codificara el formulario principal, los 9 formularios que se han establecido de acuerdo a nuestra BD.





- El código para el Botón Actualizar será el siguiente:

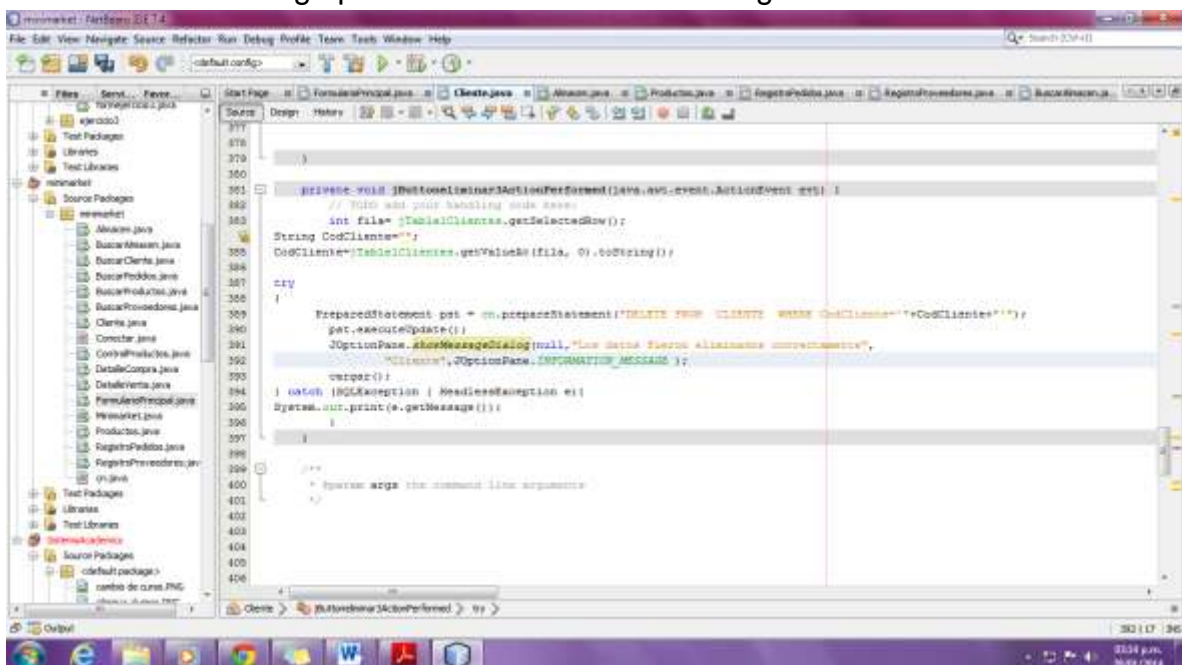


```

309
310
311
312 private void jButtonActualizarActionPerformed(java.awt.event.ActionEvent evt) {
313     // TODO add your handling code here:
314     try {
315         PreparedStatement pst = cn.prepareStatement("UPDATE CLIENTE SET CodigoCliente='"+TextFieldCodigoCliente.getText()+"'"+
316             " , CodigoPatron='"+TextFieldCodigoPatronCliente.getText()+"' , CodigoMateria='"+TextFieldCodigoMateriaCliente.getText()+"'"+
317             " , CodigoAsesor='"+TextFieldCodigoAsesorCliente.getText()+"' , CodigoDireccion='"+TextFieldCodigoDireccionCliente.getText()+"'"+
318             " , CodigoTelefono='"+TextFieldCodigoTelefonoCliente.getText()+"' , CodigoDomicilio='"+TextFieldCodigoDomicilioCliente.getText()+"'"+
319             " WHERE CodCliente='"+TextFieldCodigoCliente.getText()+"");
320
321         pst.executeUpdate();
322         cn.close();
323     } catch (Exception e) {
324         System.out.print(e.getMessage());
325     }
326 }
327
328
329
330
331
332
333
334

```

- El código para Botón Eliminar será el siguiente:

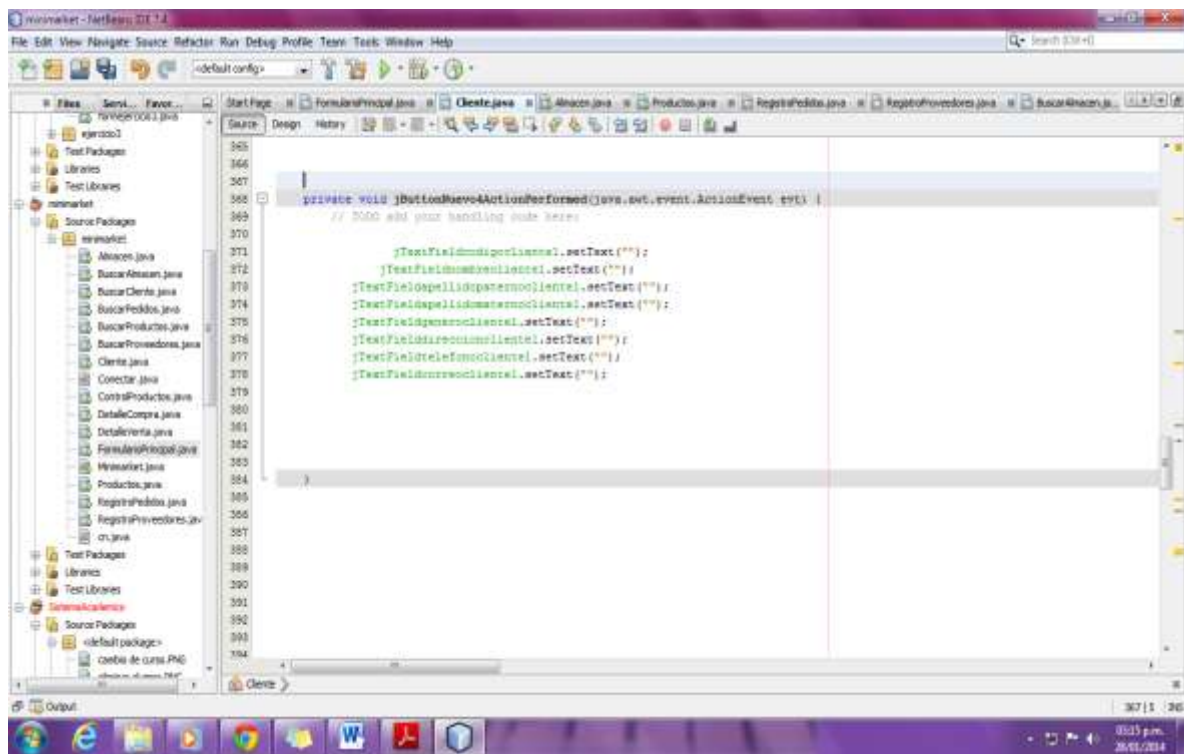


```

377
378
379
380
381 private void jButtonEliminarActionPerformed(java.awt.event.ActionEvent evt) {
382     // TODO add your handling code here:
383     int fila= jTableClientes.getSelectedRow();
384     String CodCliente="";
385     CodCliente=jTableClientes.getValueAt(fila, 0).toString();
386     try {
387         PreparedStatement pst = cn.prepareStatement("DELETE FROM CLIENTE WHERE CodCliente='"+CodCliente+"'");
388         pst.executeUpdate();
389         JOptionPane.showMessageDialog(null, "Los datos fueron eliminados correctamente",
390             "Clientes", JOptionPane.INFORMATION_MESSAGE );
391     } catch (SQLException | HeadlessException e) {
392         System.out.print(e.getMessage());
393     }
394 }
395
396
397
398
399
400
401
402
403
404
405
406

```

- El código para el Botón Nuevo será el siguiente:



Capítulo 3 CONCLUSION

De una manera general, podemos constatar al final de este informe que la creación de un software para una empresa, negocio, centro de estudios, etc. se orienta hoy día hacia la búsqueda de soluciones comunes para poder satisfacer las necesidades o requerimientos que el cliente necesita. Hoy en día los diversos programas que existen nos brindan diversas funciones y características para poder crear software que cumpla con las exigencias de los clientes.

Nosotros hemos optado por emplear MySQL como nuestro SGBD, ya que desde nuestro punto de vista es un gestor que está a la altura de nuestro proyecto y NetBeans IDE para la creación de nuestros formularios ya que es un entorno de desarrollo hecho por el lenguaje de programación de JAVA. Por último concluimos que tanto MySql, NetBeans y Java son tres motores entendibles y fáciles de emplear al momento de emplearlos para la creación de un software.

Capítulo 4 LINKOGRAFIA

- https://www.google.com.pe/?qws_rd=cr&ei=bNXnUtSuGYaekAfNzYGqBw#g=que+es+un+modelo+entidad+relacion
- http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/fuentes_k_jf/capitulo2.pdf
- <http://www.monografias.com/trabajos88/mysql-worckbench/mysql-worckbench.shtml>
- http://es.wikipedia.org/wiki/Modelo_relacional
- http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/fuentes_k_jf/capitulo2.pdf
- http://www.java.com/es/download/whatis_java.jsp