



FACULTAD DE INGENIERIA Y CIENCIAS AGROPECUARIAS

DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO BASADO EN LA
TECNOLOGÍA R.F.I.D PARA LA COMPRA DE PRODUCTOS EN
SUPERMERCADOS

Trabajo de Titulación presentado en conformidad a los requisitos establecidos
para optar por el título de Ingeniero en Electrónica y Redes de Información

Profesor Guía

Héctor Fernando Chinchero Villacís

Autor

Paúl Eduardo Molina Lara

AÑO

2015

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo a través de reuniones periódicas con el estudiante, orientando sus conocimientos para un adecuado desarrollo del tema escogido, y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.”

Héctor Fernando Chinchero Villacís

Máster en Domótica

CI: 1715451330

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

Paúl Eduardo Molina Lara

CI: 1715847206

AGRADECIMIENTOS

Agradezco primero a Dios por haberme dado la fuerza, energía y perseverancia para culminar uno de mis tantos objetivos académicos que cumpliré a lo largo de mi vida, segundo agradezco a mi papá, mamá y hermano por haber sido un gran apoyo incondicional tanto en mi trabajo como en esas largas noches de estudio que se transformaban en madrugadas y por último gracias a la universidad y a mis profesores por otorgarme los conocimientos que ahora se transformaran en mi profesión.

DEDICATORIA

Dedico este trabajo de tesis a una persona muy especial en mi vida, mi madre aquella mujer fuerte inteligente e invencible quien me apoyo y lucho a diario para ver a su hijo culminar sus estudios universitarios. Gracias infinitas querida madre.

RESUMEN

Actualmente surge la necesidad por parte del sector empresarial de automatizar todos sus procesos, es así como surge la iniciativa de crear este proyecto en el cual se va a implementar un prototipo que incorpore la tecnología RFID al momento de automatizar los procesos que se cumplen a la hora de pagar por los productos adquiridos en el supermercado.

Para cumplir todos los objetivos de automatización, es necesario desarrollar estos procesos en la plataforma de programación NetBeans IDE 7.4 y la plataforma de base de datos MySQL Workbench 5.2, por sus altos niveles de versatilidad y operatividad.

Todo esto con el fin de mejorar la experiencia del cliente en el supermercado, de tal manera que se eliminen las largas filas en los cajeros y se minimice su tiempo de pago por los productos adquiridos.

ABSTRACT

Currently there arises the need for the business sector to automate all processes, so as the initiative to create this project which will arise in implementing a prototype that incorporates RFID technology when automating processes that meet the when paying for products purchased in the supermarket.

To meet all the goals of automation, these processes need to be developed in NetBeans IDE 7.4 platform programming and database platform MySQL Workbench 5.2, for high levels of versatility and functionality.

All this in order to better the customer experience in the supermarket, so that long lines are removed from ATMs and minimize its time payment for the products purchased.

ÍNDICE

Introducción.....	1
1.1 Antecedentes.....	1
1.2 Marco referencial	3
1.3 Alcance.....	4
1.4 Justificación	4
1.5 Objetivos generales	4
1.6 Objetivos específicos	5
2 Funcionamiento de la tecnología RFID	6
2.1 Infraestructura de la tecnología RFID	6
2.1.1 Etiquetas RFID.....	7
2.1.2 Arquitectura de una etiqueta RFID.....	7
2.2.1 Etiquetas pasivas.....	9
2.2.2 Etiquetas semi-pasivas	10
2.2.3 Tipo de comunicación de las etiquetas pasivas	10
2.2.4 Etiquetas activas.....	11
2.3 Criterios para la selección de las tarjetas RFID	12
2.4 Lectores RFID.....	14
2.5 Tipos de frecuencia óptimos para la identificación de tags electrónicos	16
2.6 Lector ID-20 usado en el prototipo	17
2.7 Ventajas de la tecnología RFID sobre el código de barras.....	18
2.7.1 Características generales de la tecnología RFID.....	20
2.7.2 Características generales de los códigos de barras	22
3. Planteamiento del problema	24
3.1 Necesidades	24
3.2 Análisis de requerimientos.....	25
3.3 Definición de funcionalidad o servicio	25

4. Desarrollo del prototipo	26
4.1 Hardware	26
4.2 Software	26
4.2.1 MySQL Workbench 5.2.....	26
4.2.1.1 Requerimientos de Hardware	27
4.2.1.2 Requerimientos de Software.....	27
4.2.1.3 Creación de la base de datos y tablas	28
4.2.2 NetBeans IDE 7.4	30
4.2.2.1 Requerimientos de Hardware	31
4.2.2.2 Requerimientos de Software.....	32
4.2.2.3 Diagramas de flujo	33
5. Manejo del puerto serie.....	34
5.1 CommPort	35
5.2 CommPortIdentifier	36
5.3 SerialPort.....	36
5.4 ParalellPort	37
6. Funcionamiento del prototipo del proyecto	38
6.1 Opciones del Menú Usuarios	41
6.2 Opciones del Menú Facturación	42
7. Conclusiones.....	53
8. Recomendaciones	55
REFERENCIAS.....	56
ANEXOS	57

Introducción

1.1 Antecedentes

Es complicado establecer un punto de partida claro para la tecnología RFID. La historia de la RFID aparece entrelazada con la del desarrollo de otras tecnologías de comunicaciones a la largo del siglo XX: ordenadores, tecnologías de la información, teléfonos móviles, redes inalámbricas, comunicaciones por satélite, GPS, etc. La existencia actual de aplicaciones viables basadas en RFID se debe al desarrollo progresivo de tres áreas tecnológicas principales (Miles, Sarma y Williams, 2008, pp. 12-15).

- Electrónica de radiofrecuencia. Necesaria para el desarrollo de las antenas y los sistemas de radiofrecuencia presentes en las etiquetas e interrogadores RFID.
- Tecnologías de la información. En su vertiente de computación (en el lector, en la propia etiqueta y en el sistema de información asociado) , en su vertiente de comunicaciones (para el envío de información entre etiqueta y lector, entre lector y sistema de información asociado).
- Tecnología de materiales. Necesaria para el abaratamiento de las etiquetas.

RFID no es una tecnología nueva, sino que lleva existiendo desde 1940. Durante la Segunda Guerra Mundial, los militares estadounidenses utilizaban un sistema de identificación por radiofrecuencia para el reconocimiento e identificación a distancia de los aviones: “Friend or Foe” (amigo o enemigo). Acabada la guerra, los científicos e ingenieros continuaron sus investigaciones sobre estos temas. En octubre de 1948, Harry Stockman publicó un artículo en los Proceedings of the IRE titulado “Communications by Means of Reflected Power”, que se puede considerar como la investigación más cercana al nacimiento de la RFID (Miles, Sarma y Williams, 2008, pp. 12).

A partir de ese momento, el desarrollo de la tecnología RFID ha sido lento pero constante. Durante la década de los 50 se realizaron multitud de estudios relacionados con la tecnología, principalmente orientados a crear sistemas seguros para su aplicación en minas de carbón, explotaciones petrolíferas, instalaciones nucleares, controles de acceso o sistemas antirrobo.

En los años 80 aparecieron nuevas aplicaciones. Fue la década de la completa implementación de la tecnología RFID. Los principales intereses en Estados Unidos estuvieron orientados al transporte, al acceso de personal y, más débilmente, a la identificación de animales. En Europa sí cobró un especial interés el seguimiento de ganado con receptores de identificación por radiofrecuencia como alternativa al marcado. Más tarde también aparecieron los primeros peajes electrónicos. La primera aplicación para aduanas se realizó en 1987, en Noruega, y en 1989 en Dallas. Todos los sistemas eran propietarios, y no existía la interoperabilidad.

Y a partir del año 2000, quedó claro que el objetivo de desarrollo de etiquetas a 0,05 dólares podría alcanzarse, con lo que la RFID podía convertirse en una tecnología candidata a sustituir a los códigos de barras existentes. El año 2003 marcó un hito en el desarrollo de la tecnología RFID: Walmart y el Departamento de Defensa (DoD) estadounidense decidieron adherirse a la tecnología RFID. Les siguieron otros fabricantes, como Target, Procter & Gamble y Gillette. En 2003 el centro AutoID se convirtió en EPCglobal, creadora de estándares adoptados por Walmart y el DoD (Miles, Sarma y Williams, 2008, pp. 13).

Hoy en día las tendencias empresariales tanto como industriales es llegar a la automatización de todos sus procesos tomando estas consideraciones es oportuno diseñar y desarrollar procesos basados en la tecnología RFID que nos permitan una adecuada automatización. El cual solvente las interrogantes que van a tener tanto los operadores y usuarios de las líneas de supermercados del Ecuador como:

- La cantidad de productos que pueden ser ingresados al sistema automáticamente con el uso de esta tecnología.
- No existirán largas filas en los cajeros.
- De que manera el precio total de los productos se me debitara automáticamente de la tarjeta de crédito o débito

En definitiva es una tecnología muy rentable para el uso empresarial ya que realmente incrementara la productividad, la calidad, disminuirá tiempos y costo. En el Ecuador ha ido creciendo poco a poco pero se tiene excelentes indicios de que el uso de esta tecnología crecerá anualmente por lo que incluso ya tenemos empresas ecuatorianas dedicadas a la tecnología RFID estadísticamente todo esto nos demuestra que los riesgos son menores al momento de implementar proyectos basados en la Tecnología RFID (Miles, Sarma y Williams, 2008, pp.14).

1.2 Marco referencial

El prototipo va a ser desarrollado dentro de un ambiente simulado de compra de productos en un supermercado, el cual va a constar de cuatro procesos operacionales:

- El primero donde se va a realizar la lectura de los tags electrónicos mediante un lector de tags o antena receptora de radio frecuencia. Los tags electrónicos llevan un microchip incorporado que almacena el código único identificativo del producto al que están adheridos, el lector envía una serie de ondas de radiofrecuencia al tag, que éste capta a través de una pequeña antena, estas ondas activan el microchip que mediante la micro antena y la radiofrecuencia, transmite al lector cual es el código único del producto.
- En el segundo proceso una vez que el lector haya recibido el código único del producto, lo transmite a una base de datos, donde se han almacenado previamente las características del producto en cuestión: peso, dimensiones, lugar de elaboración, fecha de elaboración, fecha de caducidad, etc.

- El tercer proceso realiza todo lo referente a la comunicación entre la PC y el lector de los tags electrónicos lo cual se va a lograr mediante la interfaz de Java NetBeans enlazada a una base de datos.
- El cuarto proceso, consiste en el envío de la factura al cliente a su correo electrónico mediante la aplicación desarrollada en NetBeans, detallando cada producto que adquirió con su respectiva información y precio. El valor total de su factura se debitará automáticamente de su tarjeta de débito o crédito.

1.3 Alcance

Este prototipo se va a diseñar para un solo cliente con un máximo de 20 productos cada uno con su correspondiente tag electrónico, donde el costo total de los productos sea debitado automáticamente de su tarjeta de débito o crédito.

1.4 Justificación

Las empresas ecuatorianas se han dado cuenta de la importancia de adherirse a esta tecnología para incrementar su productividad, calidad y disminuir tiempos y costos lo que nos demuestra que hay una gran oportunidad comercial para este tipo de implementaciones lo cual nos ha llevado a enfocarnos a las líneas de supermercados las cuales no han sido explotadas en su totalidad por parte de la tecnología RFID.

Para este prototipo se va a implementar una línea de solo 20 productos por que la adquisición de más tags electrónicos aumentaría el costo del proyecto tanto como en las etiquetas y en la compra de una antena más potente.

1.5 Objetivos generales

- Implementar un prototipo de identificación de datos de productos basado en la tecnología RFID para la compra en supermercados del Ecuador, que debite automáticamente de las tarjetas de crédito de los usuarios, el valor total de la compra.

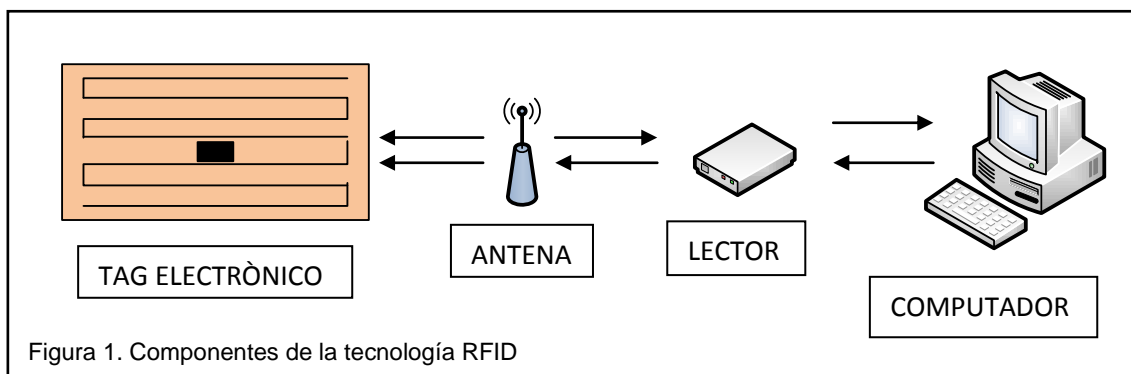
1.6 Objetivos específicos

- Implementar un algoritmo de programación basado en la interface de java NETBEANS para una correcta comunicación entre la tecnología RFID y el computador
- Enlazar la interface de java NETBEANS a una base de datos donde se almacene la información de cada producto correspondiente a cada número único de los tags electrónicos.
- Investigar los valores específicos de frecuencia con los cuales trabajará la antena para una correcta identificación de los tags electrónicos en cada producto.
- Demostrar la optimización y calidad de servicio que ganarían las principales líneas de supermercados del Ecuador disminuyendo el tiempo que el cliente permanece en caja.
- Implementar la identificación por radio frecuencia (RFID) para la compra de productos.

2 Funcionamiento de la tecnología RFID

Para que la tecnología RFID funcione, son necesarios tres elementos básicos: una **etiqueta electrónica** o tag, un **lector de tags** y una **base de datos**. Las etiquetas electrónicas llevan un microchip incorporado que almacena el código único identificativo del producto al que están adheridas. El lector envía una serie de ondas de radiofrecuencia al tag, que éste capta a través de una pequeña antena. Estas ondas activan el microchip, que, mediante la microantena y la radiofrecuencia, transmite al lector cual es el código único del artículo. En definitiva, un equipo lector envía una señal de interrogación a un conjunto de productos y estos responden enviando cada uno su número único de identificación. Por este motivo, se dice que la tecnología RFID es una tecnología de auto-identificación. Una vez que el lector ha recibido el código único del producto, lo transmite a una base de datos, donde se han almacenado previamente las características del artículo en cuestión: fecha de caducidad, material, peso, dimensiones, localización, etc. De este modo se hace posible consultar la identidad de algo o alguien en cualquier momento, ya sea el caso de una aplicación a un producto o a una persona (Hansen y Gillert, Apr 2008, pp. 20-25).

2.1 Infraestructura de la tecnología RFID



El sistema RFID se divide en cuatro procesos diferentes (Figura No.1):

- La aplicación en la PC maneja el lector y acciona los comandos.
- El lector se comunica con la etiqueta por medio de señales de RF.
 - El lector por órdenes de la aplicación genera una señal portadora.

- La señal portadora es enviada a través de la antena del lector.
- La señal portadora llega a las etiquetas.
- La etiqueta recibe y modifica la señal portadora para enviar de vuelta una señal modulada.
- La antena recibe la señal modulada y la envía al Lector.
- El lector decodifica la señal analógica para poder convertirla en datos
- Los datos digitales son enviados a la aplicación en el computador.

2.1.1 Etiquetas RFID

Las Etiquetas RFID constan de un dispositivo compuesto de un circuito electrónico y una antena integrados en uno solo. El circuito electrónico generalmente tiene una memoria para almacenar datos y alguna lógica de procesamiento. Parte de la memoria puede ser “de solo lectura”, utilizada para números de serie únicos escritos durante la etapa de producción, mientras que la otra parte puede ser tanto escrita como leída repetidamente.

La memoria puede ser también programada en fábrica por su fabricante o en el campo por el usuario final. Parte de la memoria puede ser bloqueada o desbloqueada con una clave (password) y parte puede ser bloqueada permanentemente (Ahson y Ilyas, Mar 2008, pp. 30-35).

2.1.2 Arquitectura de una etiqueta RFID

En la Figura No.2 se muestran los componentes de la arquitectura RFID los cuales son:

- Antena
 - Recibe y refleja las ondas de radiofrecuencia.
 - Es la parte más grande de la etiqueta.
 - Tiene su propia frecuencia de operación.
- Strap
 - Está conformado de un circuito integrado y dos almohadillas conductoras que pueden estar conectadas a una antena para habilitar la conversión de altas velocidades en empastes, etiquetas

o materiales de empaque y facilitar la manufactura de empastes o etiquetas.

- Circuito Integrado
 - Transforma las ondas de radio en energía eléctrica.
 - Almacenar y recuperar información.
 - Modula la señal a emitir
- Sustrato
 - Sostiene el circuito integrado y a la antena juntos sobre la etiqueta y el empaste.
 - El sustrato contiene un material adhesivo.
- Etiqueta
 - Es un ensamblaje de circuito integrado, con adhesivo a un lado.
 - Directamente pegado en los elementos.
- Empaste(inlay/insert)
 - Un ensamblaje de circuito electrónico, sin adhesivo.
 - Usado para incrustar los elementos electrónicos en la etiqueta.
- Adhesivo
 - Es aplicado a un lado del sustrato de una etiqueta.
 - Es usado para pegar etiquetas en los objetos.

(Ahson y Ilyas, Mar 2008)

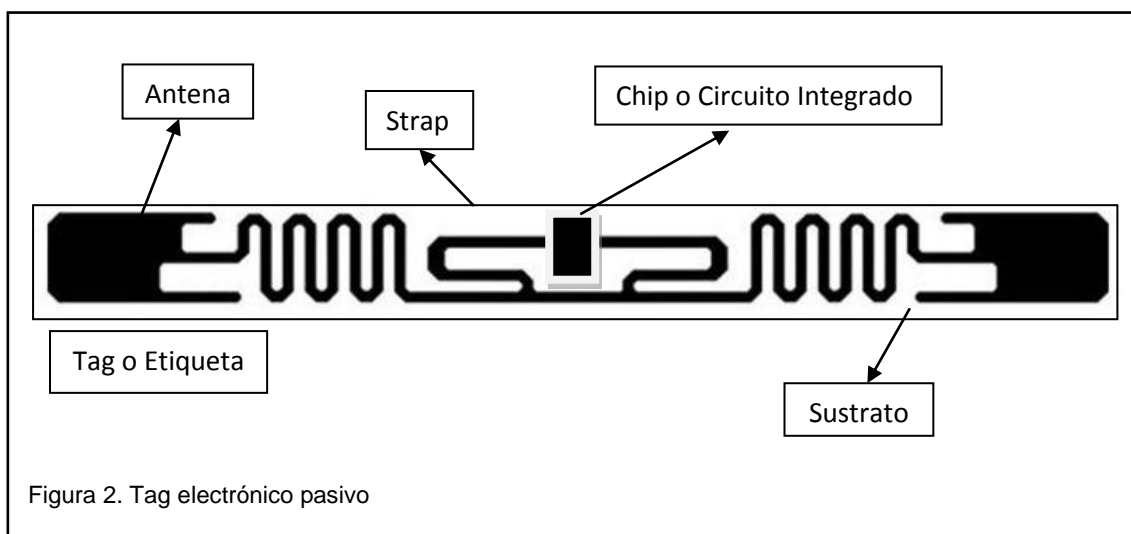


Figura 2. Tag electrónico pasivo

2.2 Clasificación de las etiquetas RFID

Las etiquetas RFID se clasifican en 4 grupos:

- Energía
 - Tags pasivos.
 - Tags semi-pasivos.
 - Tags activos.
- Frecuencia: Existen etiquetas electrónica que trabajan bajo los siguientes rangos de frecuencias:
 - 125KHz - 134KHz
 - 5MHz - 7MHz
 - 13.56MHz
 - 433MHz
 - 860MHz – 960MHz
 - 2.45GHz
- Protocolo.
 - Estandarizados.
 - Propietarios.
- Funcionalidad.
 - Clases de etiquetas EPC.
 - Capacidad de lectura.
 - Capacidad de escritura (Ahson y Ilyas, Mar 2008).
(Ahson y Ilyas, Mar 2008).

2.2.1 Etiquetas pasivas

Son aquellas que no tienen suministros de energía interno, la señal que le llega del lector es suficiente para poder integrar el circuito integrado CMOS de la etiqueta de forma que puede generar y transmitir una respuesta. Los tags pasivos suelen tener distancias de usos prácticos como de 10 cm (ISO 14443) y llegando hasta unos pocos metros (EPC e ISO 18000-6) según la frecuencia de funcionamiento, el tamaño y el diseño de la antena. Estas etiquetas operan a bajas frecuencias (Rantring, 2008).

Ventajas:

- Su tamaño es pequeño.
- Su peso es liviano.
- Son los más baratos en el mercado.
- Su vida útil es de más de 20 años.

Desventajas:

- Su rango de lectura es muy corto.
- Requieren lectoras de más potencia.
- Para su almacenamiento tiene una cantidad limitada de datos.

2.2.2 Etiquetas semi-pasivas

Son también llamadas pasivas con asistencia de batería (PAB); por que poseen una batería que le permite al circuito integrado de la etiqueta estar siempre alimentado, el lector mediante su señal de radio frecuencia activa la etiqueta. No contiene transistores y mantienen un rango de lectura de hasta 30 metros.

Ventajas con respecto a las etiquetas pasivas:

- Tiene un mayor rango de lectura.
- Fuente de alimentación propia.
- Requiere menos potencia por parte del lector.
- Tiene mayor capacidad de almacenamiento de datos.
- Son más funcionales como sensores de temperatura y GPS.

Desventajas con respecto a las etiquetas pasivas:

- Dimensiones mayores debido a la batería.
- Peso mucho mayor.
- Su tiempo de vida es limitado debido a la batería (5 - 10 años).

(Rantring, 2008)

2.2.3 Tipo de comunicación de las etiquetas pasivas

Las etiquetas pasivas y semi-pasivas emplean un sistema único de comunicación para evitar la necesidad de que una etiqueta de bajo costo necesite de un transistor.

En lugar de transmitir su propia señal, la etiqueta se acopla a la señal transmitida por el lector y según la variación del acoplamiento la etiqueta se comunica sin transmisor.

Existen dos tipos de comunicación:

- De campo cercano o acoplamiento inductivo:
Este es usado por las etiquetas de low frequency (LF) y high frequency (HF), como las ondas de radio frecuencia están hechas de componentes electromagnéticos el acoplamiento inductivo utiliza el campo magnético. La antena del lector tiene una bobina y la antena de la etiqueta también, la corriente en la antena del lector crea un campo magnético que a su vez crea corriente en la bobina de la antena de la etiqueta, variando la carga de la corriente el campo magnético pulsa y puede transferir datos.
- De campo lejano o acoplamiento reflejado pasivo:
Utilizado por las etiquetas de ultra high frequency y microondas, este utiliza el campo eléctrico, la energía del lector energiza la etiqueta y a través de la antena de la etiqueta se transfieren los datos hacia el lector.

2.2.4 Etiquetas activas

Son aquellas que contienen un transmisor e incorporan su propia fuente de alimentación en forma de baterías, pueden transmitir su señal en forma continua o en un momento definido.

Estas etiquetas son muchos más fiables que las pasivas debido a su capacidad de establecer sesiones con el lector, gracias a su fuente de energía son capaces de transmitir señales más potentes que las pasivas adquiriendo así rangos de lectura entre los 100 – 200 metros dependiendo de la potencia de la batería, estas características las lleva a ser más eficientes en entornos

difíciles para la radio frecuencia como el agua, el metal. Son mucho mayores en cuanto a peso, tamaño y almacenamiento que las etiquetas pasivas (Ranring, 2008).

Ventajas con respecto a las etiquetas pasivas:

- Rango de lectura mucho mayor.
- Menor potencia exigida al lector.
- Más funcionalidad.
- Más memoria para almacenamiento de datos.

Desventajas con respecto a las etiquetas pasivas:

- Precios mucho más alto.
- Su tamaño y peso son mucho mayor.
- Vida limitada de la batería (2 – 7 años).

2.3 Criterios para la selección de las tarjetas RFID

Sensibilidad

Es la habilidad del microchip para ser energizado y para potenciar la señal que va a ser enviada con su identificación hacia el lector, es decir, a mayor sensibilidad del microchip mayor rango de lectura.

Tamaño de la etiqueta

Su criterio se basa en la relación de a mayor tamaño mayor rango de lectura.

Forma de la etiqueta

Su criterio se basa en las diferentes formas de antenas que proveen diferentes niveles de desempeño.

Velocidad

Es uno de los criterios más importantes por que incrementa la confiabilidad de las lecturas y mejora los procesos del negocio; las etiquetas RFID disponibles hoy en día tienen velocidades de lectura que van desde 20 etiquetas/seg a 1000 etiquetas/seg.

Apilamiento apretado de etiquetas

Este criterio se refiere a un ambiente en donde tenemos un apilamiento de etiquetas estrechamente colocadas, existe una amplia variación de desempeño en este tipo de ambientes, ahora en la actualidad tenemos etiquetas que trabajan con efectividad aun situadas a 1 centímetro una de la otra.

Interferencia

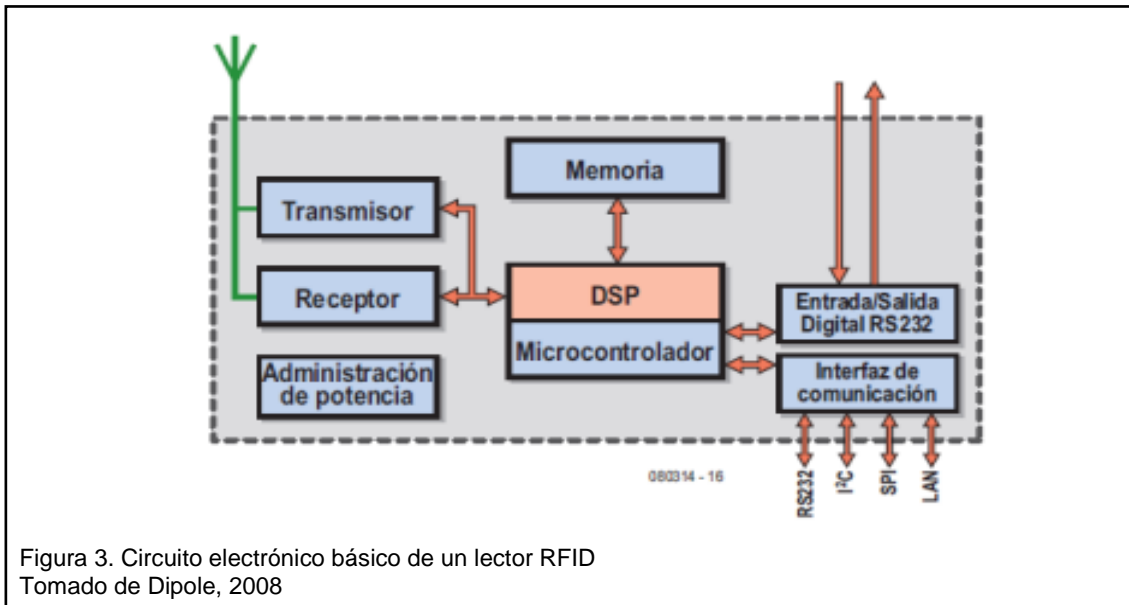
Los lectores y etiquetas bien diseñados se desempeñan bien en ambientes de radio frecuencia ruidosa.

Materiales a los que va a ir pegado la etiqueta

Materiales basados en metal y agua son generalmente hostiles para la tecnología RFID afectando de una manera negativa al rango de lectura los materiales más amigables para las etiquetas de RFID son el papel, cartón y plástico.

(Rantring, 2008)

2.4 Lectores RFID



Es transmitida una señal portadora al transmisor a través de la antena mientras la señal de referencia es receptada por el receptor, el microcontrolador en conjunto con el DSP (procesamiento de señal digital) procesan todos los datos transmitidos y recibidos, recibe comandos, se guardan procesos en su memoria y controla todos los componentes del lector. Los lectores RFID son aquellos dispositivos encargados de enviar una señal de radio frecuencia para detectar las posibles etiquetas dentro de un rango determinado, en esencia son transcriptores de señales de radio frecuencia para los tags electrónicos.

Se agrupan en 4 grandes grupos:

- Lectores fijos
- Lectores portátiles o manuales
- Lectores de sobremesa USB
- Lectores de carretilla

Estos 4 grandes grupos de lectores RFID cubren casi el 100% de todas las necesidades de los usuarios y dan respuesta a la captura inteligente de datos,

a todos ellos excepto a los manuales, es posible utilizar un amplio abanico de antenas que multiplican las posibilidades de adaptabilidad a casi todos los sectores del mercado (Dipole, 2008).

Lectores fijos.- Son aquellos encargados de generar las ondas que emiten las antenas hacia los tags, al mismo tiempo que reciben y decodifican la onda de radio frecuencia de los mismos que llega através de las antenas hacia el lector.



Lectores portátiles o manuales.- Son aquellos que facilitan la captura de datos manuales de forma masiva y con mayor rapidez que con código de barras, los modelos integrados en la propia PDA (Personal Digital Assistant) son los más fiables obteniendo el máximo rendimiento.



Lectores de sobremesa USB.- Es una gama de hardware que sirve para aplicaciones simples, sencillas, donde no requieren grandes prestaciones

tecnológicas de las lecturas. Su utilización básica es de corto alcance o Near-Field.



Figura 6. Lectores RFID de sobremesa USB
Tomado de Dipole, 2008

Lectores de carretilla.- Son aquellos que pueden capturar los datos, de los productos de su interior, de las ubicaciones de estanterías o en el suelo, del muelle de carga y de cualquier punto de referencia.



Figura 7. Lectores RFID de carretilla
Tomado de Dipole, 2008

2.5 Tipos de frecuencia óptimos para la identificación de tags electrónicos

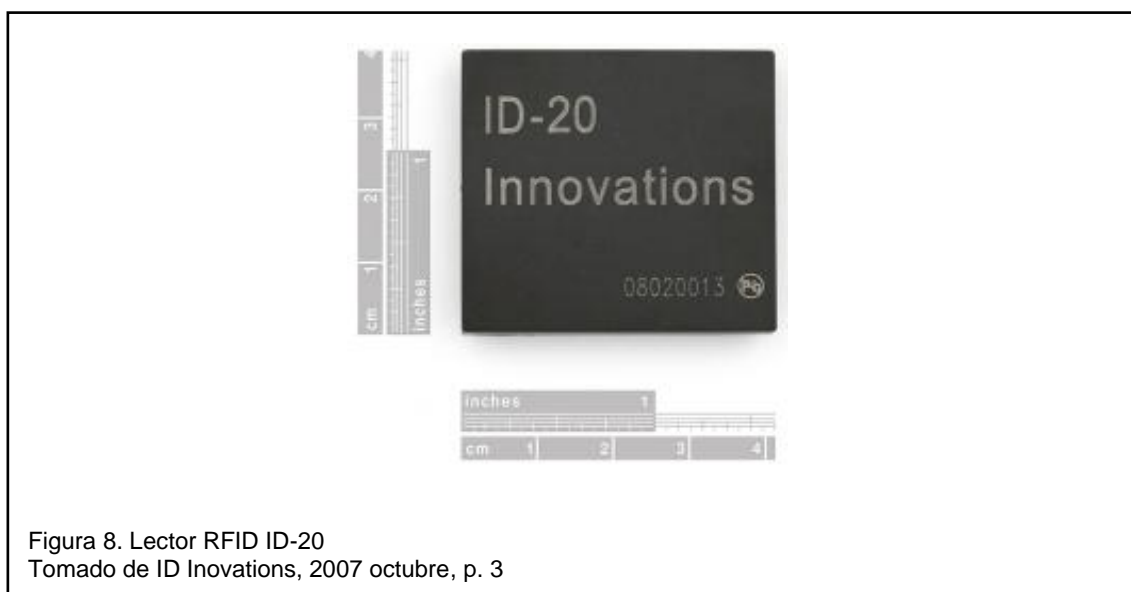
Tabla 1. Tipos de frecuencias para las identificaciones de tags electrónicos.

Frecuencias	Rango de Operación	Rango de lectura	Consumo de Energía	Rango de Transferencia de Datos	Tamaño de Longitud de Onda
Baja Frecuencia (LF)	30-300 KHz	10 cm	Bajo	Alto	10-1 Km
Media Frecuencia(MF)	300 KHz-3 MHz	20 cm	Bajo	Alto	100-100 mts

Alta Frecuencia (HF)	3-30 MHz	1 m	Moderado	Alto	100-10 mts
Ultra Alta Frecuencia (UHF)	300 MHz-3GHz	hasta 3 mts	Moderado	Rápido	1 m-1 dm
Súper Alta Frecuencia (SHF)	3-30 GHz	hasta 10 mts	Moderado	Súper Rápido	1 dm-1 cm

Tomado de Ahson y Ilyas, Mar 2008, p. 238

2.6 Lector ID-20 usado en el prototipo

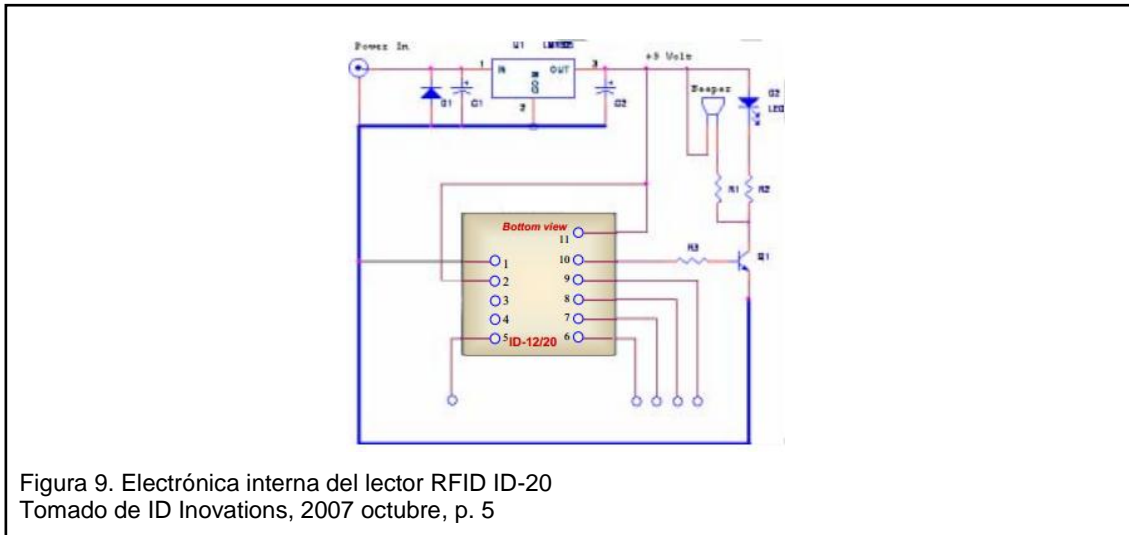


Es un lector RFID muy práctico ya que posee una antena interna y con la posibilidad de agregarle una antena externa para obtener un alcance de lectura más largo.

Cuando el sensor está conectado, con solo acercarle un tag RFID devuelve el código de identificación por su puerto serial, de tal manera que el código se podrá llevar a un microcontrolador o un ordenador utilizando un conversor USB-Serial; el lector posee las siguientes características:

- Voltaje de alimentación 5 V.
- Frecuencia de lectura: 125 KHz
- Compatible con tarjetas RFID de 64 bits (EM4001)

- Comunicación TTL serial a 9600 bps y RS232
- Emulación con tarjetas magnéticas
- Rango de lectura: 200 mm
- Dimensiones: 38mm x 40mm x 7mm



2.7 Ventajas de la tecnología RFID sobre el código de barras

La tecnología RFID supera muchas de las limitaciones del código de barras. A continuación se mencionan las ventajas de las etiquetas electrónicas:

- A diferencia del código de barras, las etiquetas electrónicas no necesitan contacto visual con el módulo lector para que éste pueda leerlas. La lectura se puede hacer a una distancia de hasta 10 metros.
- Mientras el código de barras identifica un tipo de producto, las etiquetas electrónicas identifican cada producto individual.
- La tecnología RFID permite leer múltiples etiquetas electrónicas simultáneamente. Los códigos de barras, por lo contrario, tienen que ser leídos secuencialmente.
- Las etiquetas electrónicas pueden almacenar mucha más información sobre un producto que el código de barras, que solo puede contener un código y, en algunos casos, un precio o cantidad.

- Mientras que sobre el código de barras se puede escribir solo una vez, sobre las etiquetas electrónicas se puede escribir todas las veces que haga falta.
- La tecnología RFID evita falsificaciones. Con una simple fotocopia se puede reproducir un código de barras. Las etiquetas electrónicas, en cambio, no se pueden copiar. Un tag sobre un artículo de marca garantiza su autenticidad.
- Un código de barras se estropea o se rompe fácilmente, mientras que una etiqueta electrónica es más resistente porque, normalmente, forma parte del producto o se coloca bajo una superficie protectora y soporta mejor la humedad y la temperatura (Hansen y Gillert, Apr 2008).

Si bien esta tecnología se ha diseñado principalmente para facilitar el trabajo de comercialización de productos de consumo, en diferentes áreas, se han encontrado usos a la posibilidad de seguir la vida de un chip durante un determinado periodo de tiempo. Por poner algunos ejemplos; en las entradas a espectáculos, para el control de acceso a las carreteras de peaje. En los hospitales, un brazalete puesto a un paciente, le identifica y asocia en cualquier lugar y momento con la medicación y régimen alimenticio que deba seguir. La comunidad europea piensa introducir esta técnica en la próxima generación de billetes para entre otras cosas de facilitar el recuento de billetes. En bibliotecas y centros de documentación se pueden localizar los libros y documentos solicitados, etc.

Es importante mencionar que uno de los problemas más importantes para su implementación a corto plazo a nivel de unidad, además del costo por etiqueta frente al precio del actual código de barras, corresponde a ciertas limitaciones técnicas, tales como la dificultad de leer RFID a través de líquidos o metales; por otro lado también existen limitaciones de cantidad de información a almacenar (96 bits).

En algunas industrias, la decisión de adoptar la tecnología RFID será acelerada debido a regulaciones gubernamentales.

La tecnología actual de las etiquetas de RFID ofrece numerosas limitaciones cuando se la compara con la tecnología de etiquetado ideal, la cual sería económica, reduciría la imposibilidad de faltas de lectura y ofrecería flexibilidad para el diseño de la antena. También proporcionaría pericia en el proceso con grandes rendimientos, permitiría las pruebas durante el proceso y el control de calidad y cumpliría normas neutrales al vendedor.

Por último, la etiqueta ideal integraría de forma completa el proceso de etiquetado y demarcación, proporcionaría un proceso integrado y flexible de fabricación de etiquetas, ofrecería una función de lectura/registro de alta velocidad y sería amigable para el medio ambiente.

La tecnología RFID existente, por el contrario, presenta numerosos desafíos. Los precios de las etiquetas son relativamente elevados, los problemas de lectura son importantes y el tamaño y la forma de las antenas sigue siendo muy inflexible. La tecnología de etiquetado existente tiene además un rendimiento limitado de la información del proceso, limitaciones de derecho de propiedad y otras no relacionadas con la normalización, además de problemas de confiabilidad y de logística, incluyendo los daños a las etiquetas durante el proceso de conversión de las mismas, la velocidad de lectura/registro, los problemas en la integración del etiquetado y demarcación. (Hansen y Gillert, Apr 2008)

2.7.1 Características generales de la tecnología RFID

- Fuente de alimentación propia mediante baterías de larga duración.
- Distancias de lectura escritura mayor de 10m a 100m generalmente.
- Diversas tecnologías y frecuencias.
 - Hasta 868 MHz (UHF) o según estándares aplicados.
 - 2,4 GHz frecuencia muy utilizada (banda ISM, *Industrial Scientific and Medical*), la misma que se aplica para dispositivos wireless LAN 802.11b.
- Memoria generalmente entre 4 y 32 kbytes.

- Batería de larga duración (generalmente baterías de litio / dióxido de manganeso)
- Fabricantes: TagMaster, Identec Solutions, Siemens, WhereNet, Bluesoft, Syris RFID.

La principal ventaja de los tags RFID activos respecto a los pasivos es el elevado rango de lectura, del orden de decenas de metros. Como desventajas, cabe destacar el precio, que es muy superior que los tags pasivos y la dependencia de alimentación por baterías. El tiempo de vida de las baterías depende de cada modelo de tag y también de la actividad de este, normalmente es del orden de años. Para facilitar la gestión de las baterías, es habitual que los tags RFID activos envíen al lector información del nivel de batería, lo que permite sustituir con antelación aquellas que están a punto de agotarse.

En el mercado se tiene una variada gama como por ejemplo el Tag RFID Activo SYTAG245-2C que tiene las características mencionadas:

- Frecuencia de comunicación: 2,45 GHz
- Rango de frecuencia: entre 2,40 y 2,48 GHz (opción de 125 kHz ó 13,56 MHz)
- Canal: 255
- Dirección: 65536 direcciones
- Wake on radio: ON / OFF
- RSSI: 0-255
- ID: 64 bits.
- Programación: configurable a partir de comandos.
- Led: acción o estatus.
- Conmutación: configurada como tag activo o tag ON/OFF.
- Memoria: de 4 a 32 kbytes (opcional).
- Batería: 3 VDC CR2032 × 1 ó × 2
- Duración de la batería: de 1 a 10 años
- Consumo en reposo: 3 uA @ 3 VDC.

- Consumo en funcionamiento: 24 mA @ 3 VDC
- Temperatura de funcionamiento: entre -10 y 55 °C
- Temperatura de almacenaje: entre -20 y 65 °C
- Humedad relativa de funcionamiento y almacenaje: de 5% a 95%

Ejemplos de otros modelos de tags RFID activos son: SYTAG245-2K, SYTAG245-2S, SYTAG245-TM.

2.7.2 Características generales de los códigos de barras

- La información se procesa y almacena con base en un sistema digital binario donde todo se resume a sucesiones de unos y ceros. La memoria y central de decisiones lógicas es un computador electrónico del tipo estándar, disponible ya en muchas empresas comerciales y generalmente compatible con las distintas marcas y modelos de preferencia en cada país. Estos equipos permiten también interconectar entre sí distintas sucursales o distribuidores centralizando toda la información. Ahora el distribuidor puede conocer mejor los parámetros dinámicos de sus circuitos comerciales, permitiéndole mejorar el rendimiento y las tomas de decisiones, ya que conocerá con exactitud y al instante toda la información proveniente de las bocas de venta estén o no en su casa central. Conoce los tiempos de permanencia de depósito de cada producto y los días y horas en que los consumidores realizan sus rutinas de compras, pudiendo entonces decidir en qué momento debe presentar ofertas, de qué productos y a qué precios.
- Los códigos de barras se imprimen en los envases, embalajes o etiquetas de los productos. Entre sus requisitos básicos se encuentran la visibilidad y fácil legibilidad por lo que es imprescindible un adecuado contraste de colores.
- El negro sobre fondo blanco es el más habitual encontrando también azul sobre blanco o negro sobre marrón en las cajas de cartón ondulado. El código de barras lo imprimen los fabricantes (o, más habitualmente, los fabricantes de envases y etiquetas por encargo de los primeros) y, en algunas ocasiones, los distribuidores.

- Codifica solamente números
- Usa poco espacio
- Por sus características, puede leerse parcialmente de modo accidental, por lo que se debe configurar el lector de código de barras para que lea cierta longitud de caracteres y evitar que se pierda información
- Solo codifica número de caracteres pares, por lo que añade ceros al inicio cuando se requiere

(Hansen y Gillert, Apr 2008, pp. 40-43).

3. Planteamiento del problema

3.1 Necesidades

El desarrollo del proyecto surge por la necesidad de otorgar un crecimiento tecnológico a las cadenas de supermercados, brindando así una automatización en la compra de sus productos ya que casi el 95% de sus productos cuentan con la tecnología de código de barras y bandas transportadoras controladas por el usuario.

La problemática para el cual se requiere el desarrollo del proyecto se ha enfocado esencialmente en las cajas de los supermercados donde los cajeros tienen que hacer un proceso rutinario y mecanizado, el cual consta en pasar cada producto por el lector de códigos de barras lo que provoca incomodidad y pérdida de tiempo para los clientes, en una estimación de 12 a 18 minutos en caja dependiendo la cantidad de productos que se tenga en el carrito de compras, estos tiempos se van duplicando según la cantidad de clientes que tenga el cajero dando así lugar para generarse las largas filas que se suelen ver en los supermercados al momento de pagar.

Los procesos que se realizan para que el cliente pueda pagar por sus productos son:

- a. Al llegar a la caja debe colocar todos sus productos en la banda transportadora.
- b. El cliente sigue colocando sus productos en la banda transportadora mientras el cajero debe ir tomando uno a la vez y pasarlo por el lector de código de barras.
- c. Después de haberse registrado todos los códigos de barras correspondientes a cada producto adquirido por el cliente, el cajero debe procesar la información en el ordenador.
- d. Se procede a pedir el modo de pago del cliente ya sea con tarjetas de débito, crédito o en efectivo.

- e. Una vez establecido el modo de pago del cliente, el cajero procede a emitir la factura con los precios desglosados de cada producto y el precio total.
- f. Mientras los procesos c, d, y e son ejecutados, debe haber otra persona tomando los productos para empacarlos en funda y así puedan ser llevados a casa por el cliente, de no ser así este proceso deberá ser realizado por el mismo cliente.

3.2 Análisis de requerimientos

Lo que necesita el cliente es un proceso de automatización que le de la libertad de evitarse las largas filas al momento de pagar en los supermercados, para lograr esto también se necesita brindar al cliente la opción de un pago totalmente automatizado, donde al momento de hacer la compra, la factura le llegue automáticamente al correo de su ordenador o teléfono móvil.

3.3 Definición de funcionalidad o servicio

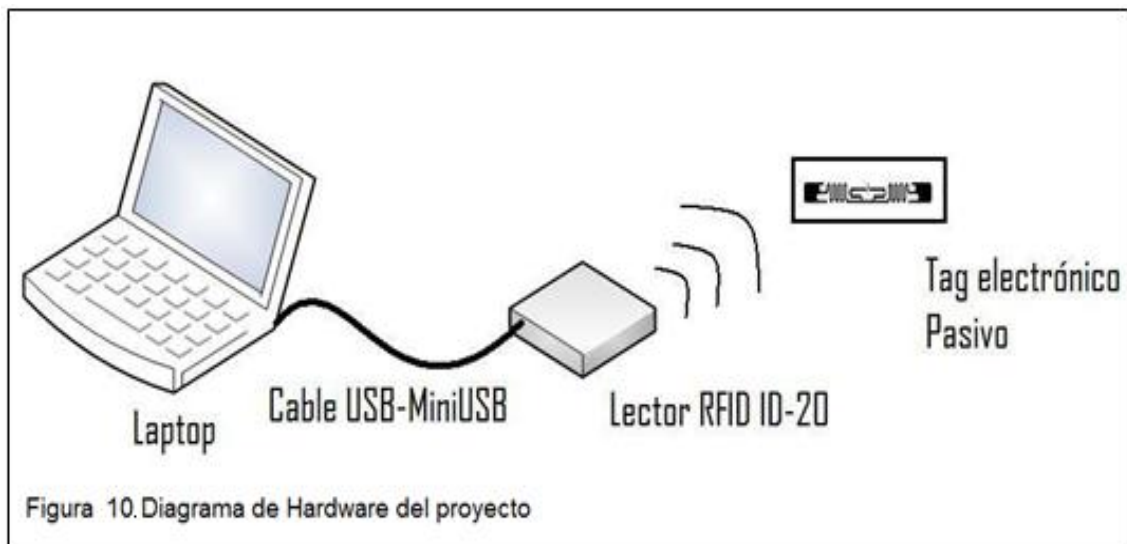
Esto se puede realizar con la tecnología RFID, ya que su funcionalidad principal es la automatización de procesos mediante la identificación por radio frecuencia de etiquetas electrónicas o tags electrónicos, los cuales van a ir adheridos a los productos que van a ser identificados por un lector RFID de manera masiva y automática en el caso de equipos ideales, pero para la realización del prototipo de este proyecto los productos van a ser identificados uno por uno ya que no se cuenta con equipos ideales por motivos de costos.

4. Desarrollo del prototipo

4.1 Hardware

Los componentes electrónicos que integran el prototipo son:

- Laptop DELL Inspiron 1318 (3 GB ram, S.O Win. Vista 32 bits, 1.87Ghz, 288 GB).
- Tags electrónicos pasivos.
- Lector RFID ID-20
- Cable de conexión USB – MiniUSB



4.2 Software

Las herramientas que utiliza el prototipo son:

- MySQL Workbench 5.2
- NetBeans IDE 7.4

4.2.1 MySQL Workbench 5.2

Es una herramienta visual unificada para los arquitectos de bases de datos, desarrolladores y administradores de bases. Este programa proporciona:

- Modelado de datos

- Desarrollo de SQL (Structured Query Language o Lenguaje Estructurado de Consulta)
- Administración de servidores

Modelado de datos

Permite crear modelos de un esquema de bases de datos gráfica y editar todos sus aspectos con el editor de tablas integral. El editor de tablas integral proporciona instalaciones fáciles de usar para los cuadros de edición, columnas, índices, activadores, partición, opciones, insertos, privilegios, rutinas y vistas.

Desarrollo SQL

Permite crear y administrar las conexiones a los servidores de bases de datos, además de permitir la conexión y los parámetros. MySQL Workbench proporciona la capacidad de ejecutar consultas SQL en conexiones de bases de datos utilizando el built-in que es un editor SQL.

Esta funcionalidad sustituye al previamente establecido por la aplicación independiente Query Browser.

Administración del Servidor

Permite crear y administrar las instancias del servidor.

4.2.1.1 Requerimientos de Hardware

- CPU: Intel Core o XEON (1.87Ghz – 3Ghz) o su igual procesador AMD
- Core: Single, Dual, Quad (son recomendables)
- RAM: 3Gb – 6Gb (son recomendables)
- Tarjeta de Aceleración Grafica: Nvidia o ATI de 2Gb - 4Gb
- Resolución de Display: 1280x1024 es recomendado y mínimo 1024x768

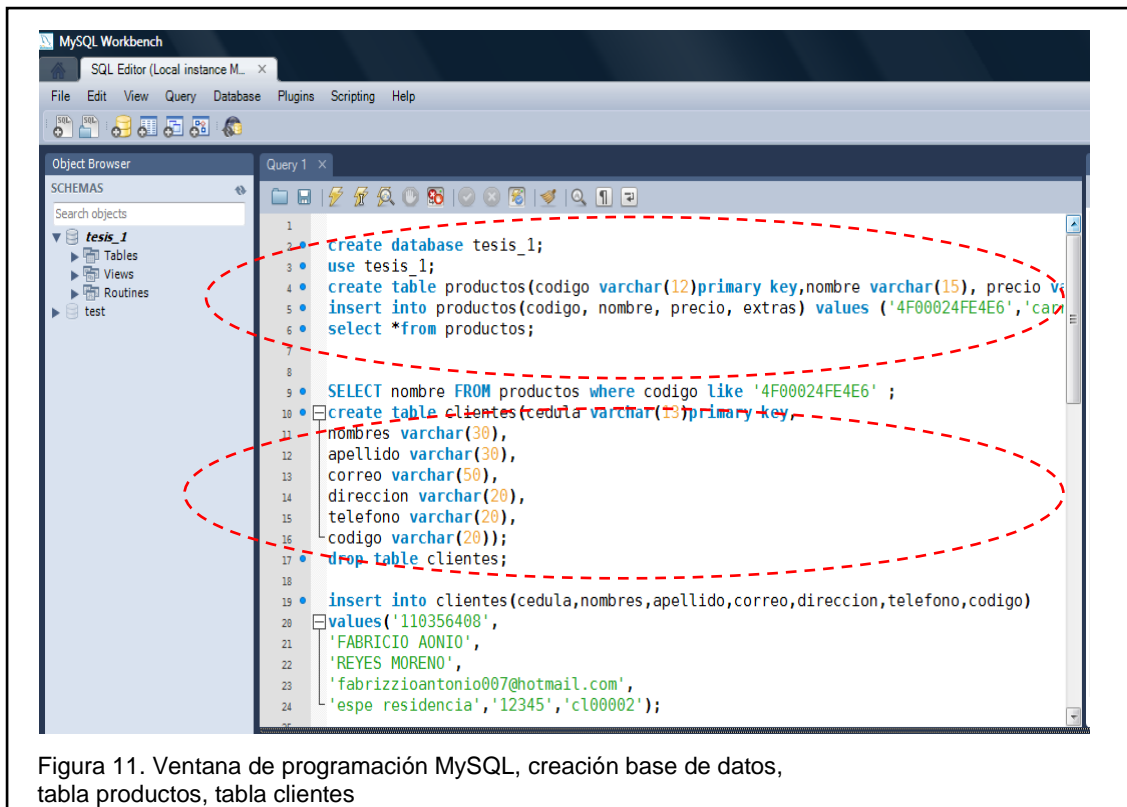
4.2.1.2 Requerimientos de Software

- Windows 7 Professional/Vista (32bits o 64bits)
- Mac OS X 10.6.1

- Ubuntu 9.10 (64 bits)
- Ubuntu 8.34 (32 bits/ 64 bits)
- Microsoft NET Framework 3.

4.2.1.3 Creación de la base de datos y tablas

Base de Datos tesis_1, es creada como se muestra en la siguiente figura:



La creación de la base de datos (tesis_1) y las tablas es una de las partes más importantes ya que contiene toda la información sobre los clientes, los productos y la **tarjeta** de crédito.

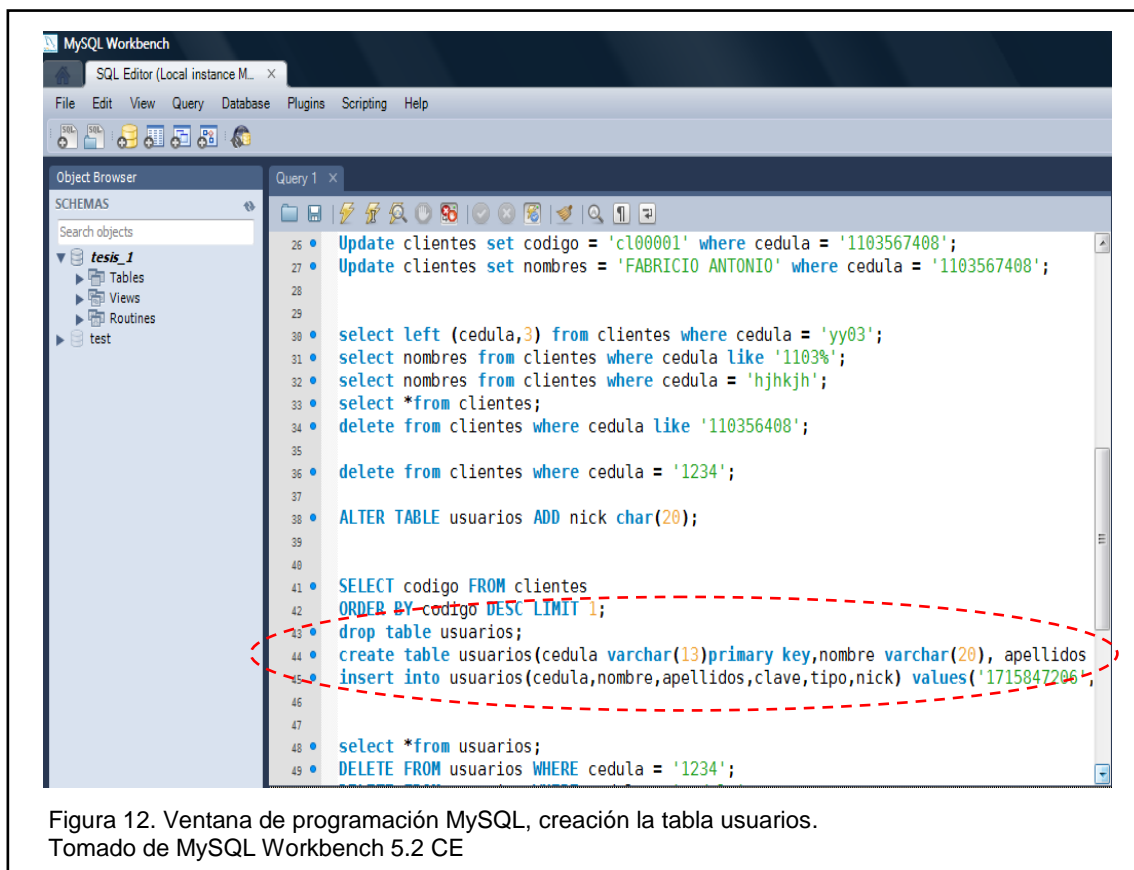
En primera instancia dentro de la base de datos tesis_1 se procedió a crear la tabla **productos** la cual contiene los campos:

- **código:** de tipo varchar (12), el cual es una clave primaria (primary key: es aquella que ha sido diseñada para identificar de manera única a los registros de una tabla a través de toda la estructura de la base de datos).

- **nombre:** de tipo varchar (15).
- **precio:** de tipo varchar (5).
- **extras:** de tipo varchar (20).

Luego se procede a crear la tabla **clientes** cuyos campos son:

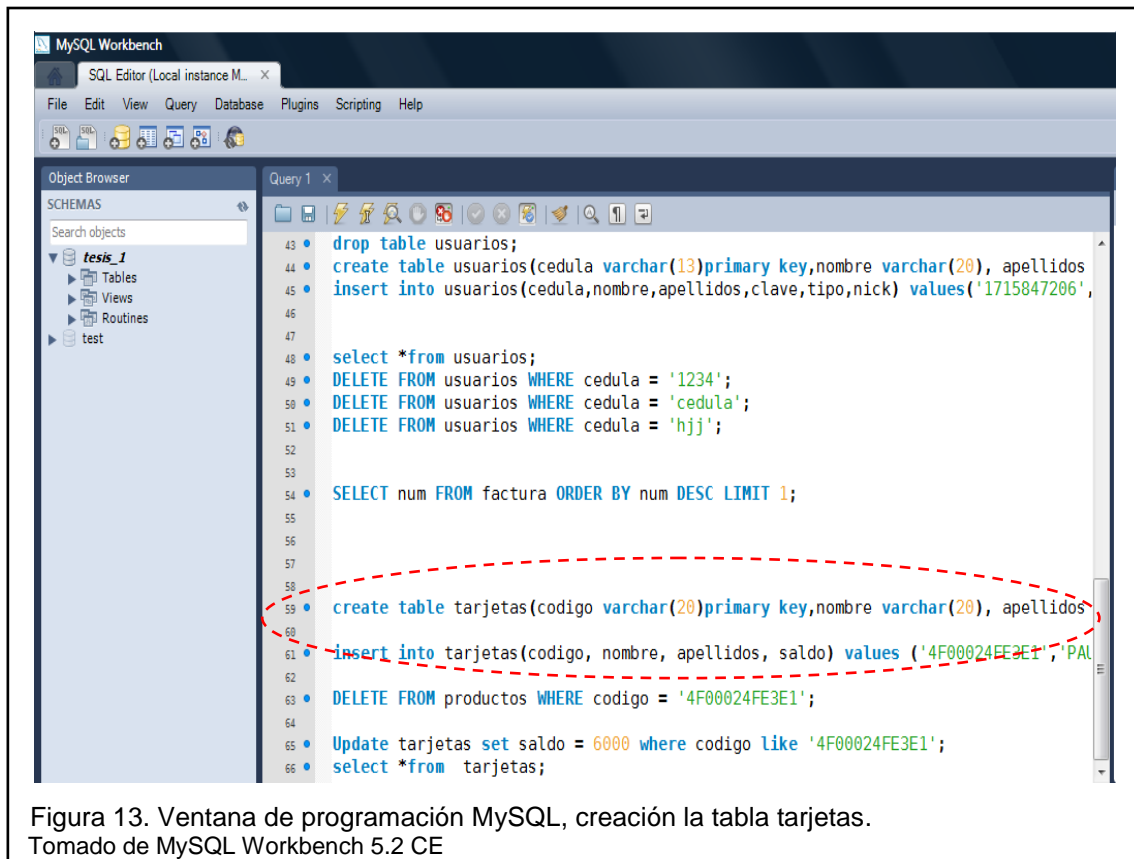
- **cedula:** de tipo varchar y es una clave primaria.
- **nombres:** de tipo varchar (30).
- **apellido:** de tipo varchar (30).
- **correo:** de tipo varchar (50).
- **dirección:** de tipo varchar (20).
- **teléfono:** de tipo varchar (20).
- **código:** de tipo varchar (20).



La tercera tabla en crear es la de los **usuarios**, la cual está formada por los siguientes campos:

- **cedula:** de tipo varchar (13) y es una clave primaria.

- **nombre:** de tipo varchar (20).
- **apellidos:** de tipo varchar (20).
- **clave:** de tipo varchar (10).
- **tipo:** de tipo varchar (2).
- **nick:** de tipo char (20)



La última tabla en crear es la tabla de **tarjetas** y es aquella que contiene los siguientes campos:

- **código:** de tipo varchar (20) y es una clave primaria.
- **nombre:** de tipo varchar (20).
- **apellidos:** de tipo varchar (20).
- **saldo:** de tipo int (10).

4.2.2 NetBeans IDE 7.4

Es un entorno de desarrollo gratuito y de código abierto, permite el uso de un amplio rango de tecnologías de desarrollo tanto para escritorio, como

aplicaciones Web o para dispositivos móviles además puede instalarse en varios sistemas operativos: Windows, Linux, Mac OS.

Características Principales

- Soporta todas las novedades en el lenguaje JAVA.
- Asistentes para la creación y configuración de distintos proyectos, incluida la elección de algunos frameworks.
- Buen editor de código multilinguaje.
- Simplifica la gestión de grandes proyectos con el uso de diferentes vistas, asistentes de ayuda y estructurando la visualización de manera ordenada
- Herramientas para depurado de errores, el debugger el cual es bastante útil para encontrar errores en las líneas de programación, podemos definir puntos de ruptura en las líneas del código que queramos analizar, monitorear en tiempo real los valores de propiedades y variables. Se puede utilizar el debugger en caliente, conectándonos a él cuando ya tenemos un proceso ejecutándose.
- Optimización del código por parte del Profiler que es el que nos ayuda a optimizar nuestras aplicaciones e intentar hacer que se ejecuten más rápido y con el mínimo uso de memoria.
- Acceso a base de datos, desde el propio NetBeans podemos conectarnos a distintos sistemas gestores de base de datos como pueden ser Oracle, MySQL, etc, de tal manera que podemos realizar consultas y modificaciones integrando todo esto en el propio IDE (Integrated Development Environment).

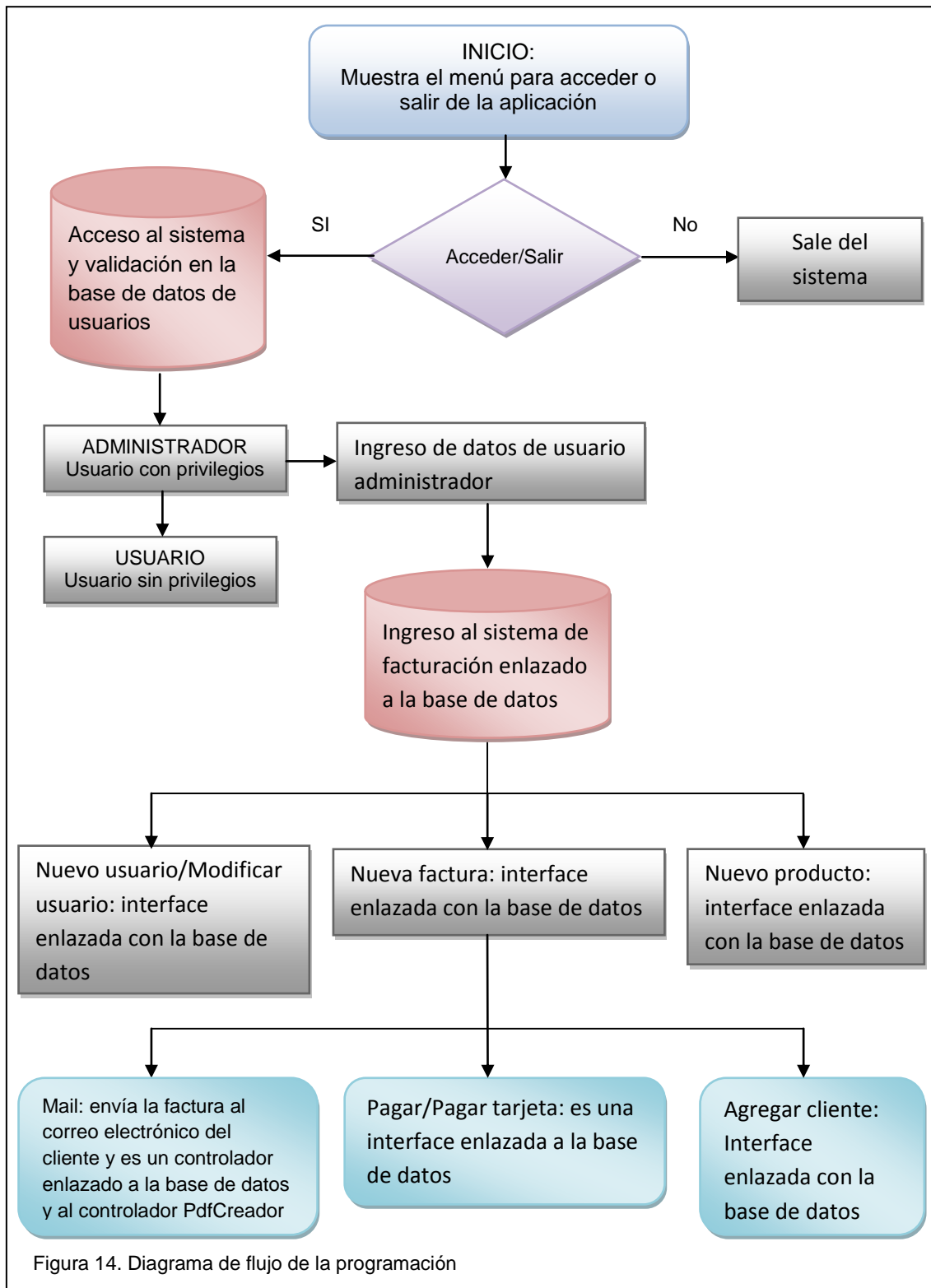
4.2.2.1 Requerimientos de Hardware

- Procesador: Intel Quad Core – Intel Core i5.
- Ram: 2 GB (32 bits) – 4GB (64 bits).
- Espacio en el disco: mínimo 2 GB.

4.2.2.2 Requerimientos de Software

- Sistemas Operativo Windows: Vista SP1 - Windows 7 Profesional.
- Ubuntu 12.04.
- Solaris OS versión 11 Express .
- Solaris OS versión 11 Express (x86/x64 plataform edition).

4.2.2.3 Diagramas de flujo



5. Manejo del puerto serie

El API Java Communications (COMM) es un paquete adicional para la plataforma Java2. Proporciona soporte para comunicación con dispositivos periféricos a través de los puertos serie y paralelo.

Este es API muy especial en el sentido de que aunque esté bien definido como multi-plataforma, debe descargarse una versión específica de las librerías COMM para utilizarlo e implementa los estándares RS-232 para puerto serie e IEEE-1284 para puerto paralelo.

La instalación del API de comunicaciones de java para Windows, se la realiza copiando los siguientes archivos en los paths indicados.

- comm.jar → C:\Program Files\Java\jdk1.7.0_51\jre\lib\ext
- win32com.dll → C:\Program Files\Java\jdk1.7.0_51\bin
- Javax.comm.properties → C:\Program Files\Java\jdk1.7.0_51\jre\lib

Además si existiera algún error, se debe copiar:

- win32com.dll → C:\Windows\System32

En el paquete de comunicaciones javax.comm existe una serie de clases que permiten tratar varios niveles de programación:

- **Nivel alto:** En este nivel tiene las clases **CommPortIdentifier** y **CommPort** que permiten el acceso a los puertos de comunicación.
- **Nivel Medio:** Con las **SerialPort** y **ParallelPort** cubre las interfaces físicas RS-232 para el puerto serie, e IEEE 1284 para el puerto paralelo.
- **Nivel Bajo:** Este nivel se enlaza con el sistema operativo y en el se encuentra el desarrollo de drivers.

Los servicios que proporciona este paquete son:

- Poder obtener puertos disponibles así como sus características.

- Abrir y mantener una comunicación en los puertos. Esto permite tener varias aplicaciones Java funcionando a la vez.
- Resolver colisiones entre aplicaciones.

Las clases disponibles para el manejo de puertos son:

- CommPort
- CommPortIdentifier
- ParallelPort
- SerialPort

5.1 CommPort

Esta es una clase abstracta que describe los métodos comunes de comunicación y serán las clases que heredan de ellas (SerialPort y ParallelPort) las que añaden métodos y variables propias del tipo del puerto.

- **getPortIdentifiers().-** Este método entrega un enumerado de tantos objetos CommPortIdentifier como puertos se disponga.
- **getPortType().-** Devuelve un entero que informa el tipo del puerto (serie o paralelo), se dispone de las constantes PORT_PARALLEL y PORT_SERIAL.
- **isCurrentOwned().-** Informa si esta libre o no el puerto, en el caso de que este ocupado se puede saber quien lo está utilizando mediante el método getCurrentOwner().
- **Open(String, int).-** Abre y por lo tanto reserva un puerto. Los parámetros son un String con el nombre de la aplicación que reserva el puerto y un int que indica el tiempo de espera para abrir el puerto. En el caso de que se intente abrir un puerto que este siendo utilizado saltara la excepción PortUseException.
- **Close().-** Permite liberar el puerto que se reservo con open, este notificara el cambio de dueño a las clases que se hubiesen registrado con el método addPortOwnershipListener.

- **getOutputStream().-** Permite enlazar la salida del puerto al OutputStream que devuelve para poder escribir en el puerto de la misma forma que si escribiera en un fichero.
- **getInputStream().-** Permite enlazar la entrada del puerto al InputStream para leer del puerto.

5.2 CommPortIdentifier

Administra la comunicación con los puertos. Controla el acceso a los puertos, determina los puertos disponibles y ocupados por los dispositivos finalmente maneja eventos de cambio sobre el estado.

5.3 SerialPort

Corresponde a la interface de bajo nivel del puerto serie que cumple con el estándar RS-232. La clase SerialPort hereda la clase abstracta CommPort y por lo tanto cuenta con sus métodos pero además de estos dispone de otros métodos y variables específicas para el tratamiento de los puertos series.

- **setSerialPortParam(int, int, int, int).** - Permite configurar los parámetros del Puerto serie. Este método utiliza la excepción `UnsupportedCommOperationException` en el caso de que los valores no sean soportados. Los parámetros son:
 - La velocidad.
 - Bits de datos, para indicar el valor se utiliza las constantes de la clase (`DATA_5`, `DATA_6`, `DATA_7`, `DATA_8`).
 - Bit o bits de stop, que puede ser 1,2 o 15. Las constantes que definen estas configuraciones son: `STOPBITS_1`, `STOPBITS_2`, `STOPBITS_1_5`.
 - Paridad, que pueden ser `PARITY_NONE` en el caso de no utilizar parida, `PARITY_ODD` para paridad impar, `PARITY_EVEN` para paridad par, `PARITY_MARK` para paridad por marca y `PARITY_SPACE` para paridad por espacio.

5.4 ParalellPort

Es el puerto paralelo implementado en la PC, el cual consiste en un conector de 17 líneas de señal y 8 líneas de tierra. Las líneas de señal se dividen en tres grupos:

- Bus de Control (4 Pines)
- Bus de Estado (5 Pines)
- Bus de Datos (8 Pines)

Para redireccionar los buses que integran el puerto paralelo se requiere del conocimiento de la ubicación de la dirección que se le asignó a este en la PC. Comúnmente se le asigna la dirección 378, que corresponde al bus de datos y que a su vez identifica al puerto paralelo.

El redireccionamiento permite seleccionar que puerto se requiere utilizar; el puerto paralelo va desde 0x378 hasta 0x37F. Estas direcciones establecen el uso que va a tener el puerto.

6. Funcionamiento del prototipo del proyecto

En primera instancia se procede a acceder al proyecto **SISTEMALECTURATAGS** dentro de NetBeans IDE 7.4 una vez ya realizado lo mencionado se procederá a correr el proyecto.

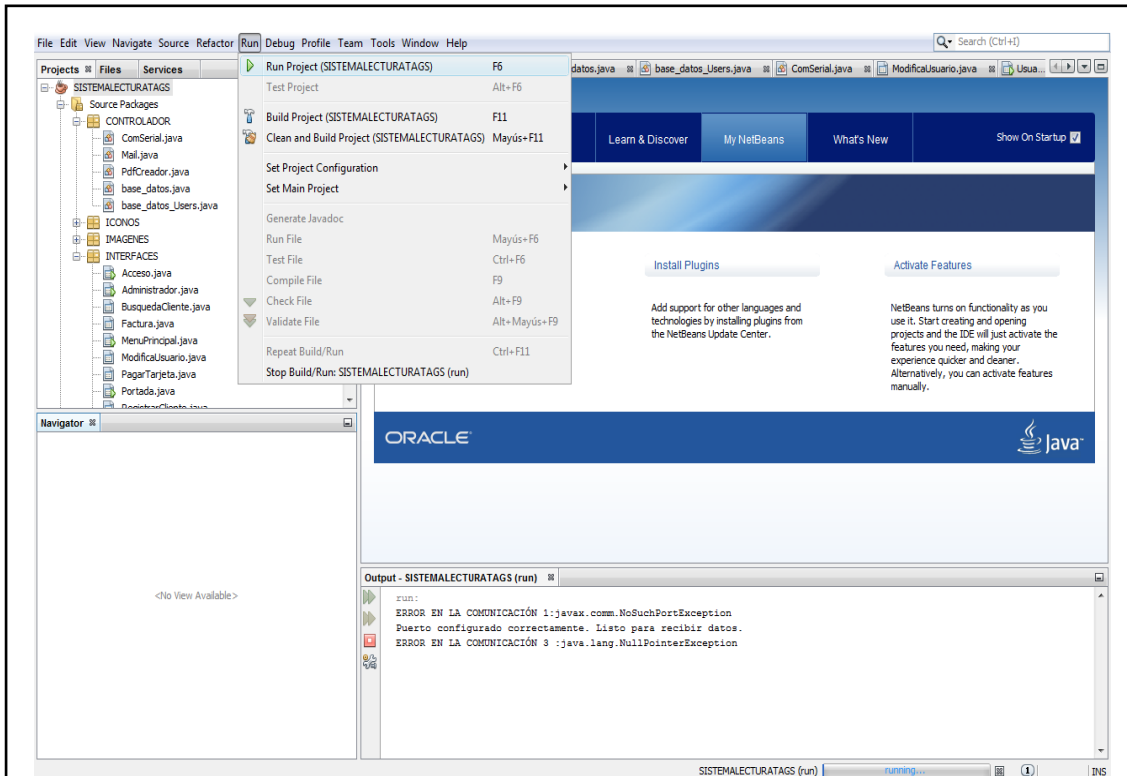


Figura 15. Entorno de la plataforma de programación NetBeans IDE 7.4.
Tomado de MySQL Workbench 5.2 CE



Figura 16. Ventana de Acceso del Sistema de Lectura de Tags

Se procederá a darle click en el botón ACCEDER, y se nos desplegara la ventana de acceso en la cual tendremos que ingresar nuestro usuario y contraseña y darle click en **ACEPTAR** para poder tener acceso al sistema.

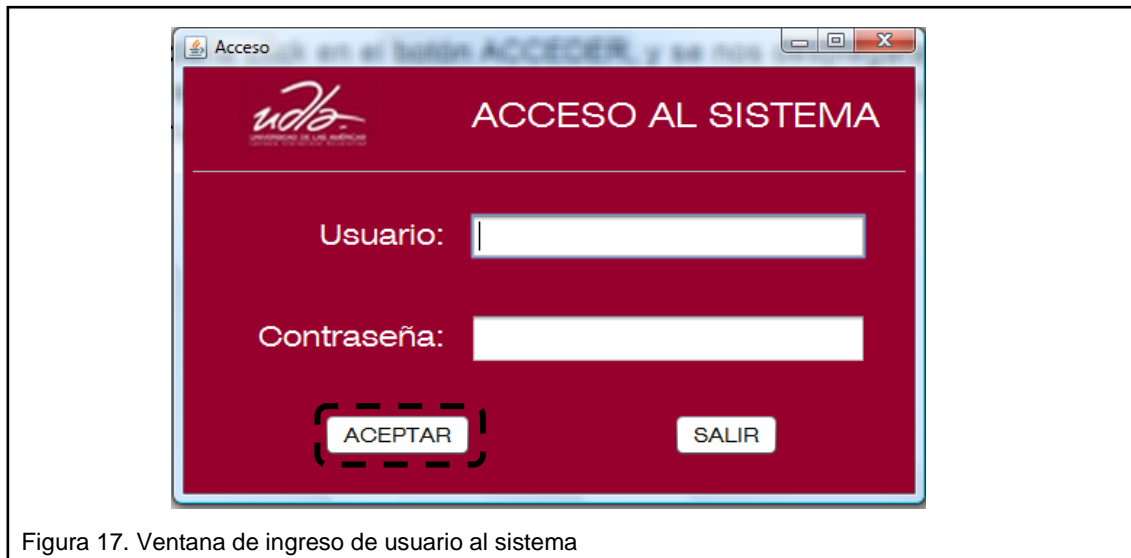


Figura 17. Ventana de ingreso de usuario al sistema

Después de ingresar nuestro usuario y contraseña correctos se nos desplegara una ventana dándonos la bienvenida al sistema y señalándonos qué tipo de usuario somos si estándar o administrador. Para ingresar al sistema le damos click en **MENÚ**

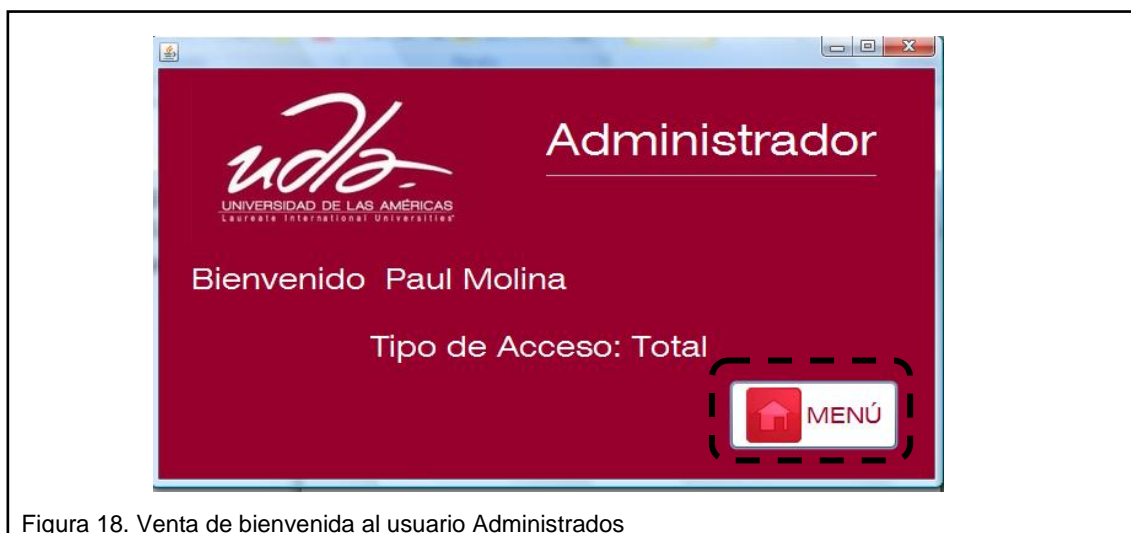


Figura 18. Venta de bienvenida al usuario Administrados

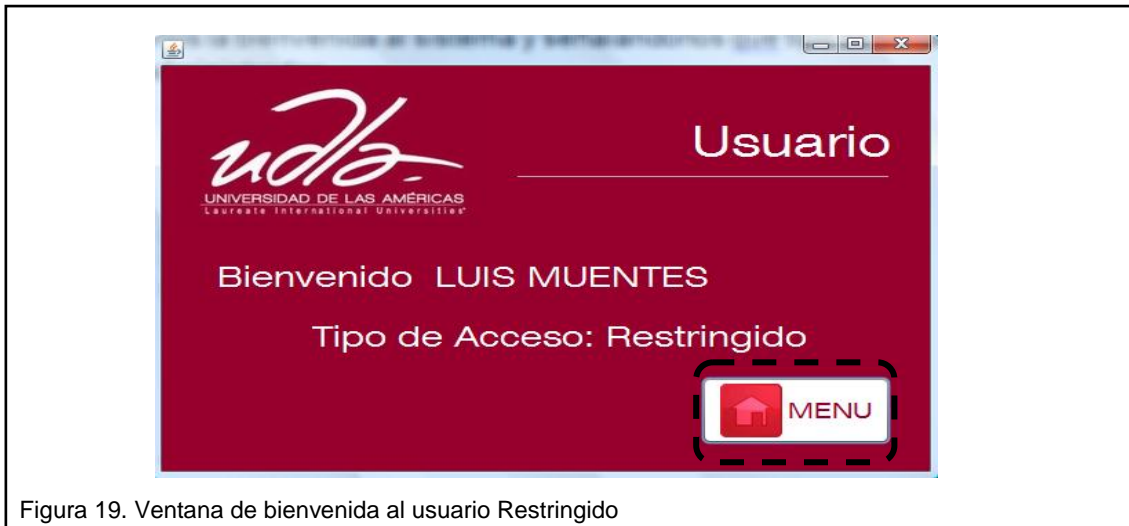


Figura 19. Ventana de bienvenida al usuario Restringido

Procedemos a dar click en botón menú y nos mostrara una ventana en la que tenemos tres pestañas las cuales son: Usuario, Facturación, Salir.

- **Usuario.-** Dentro de esta pestaña tenemos las opciones de: Reiniciar Sesión, Nuevo Usuario y Modificar Usuario.
- **Facturación.-** Dentro de esta pestaña tenemos las opciones de: Factura Actual, Nueva Factura, Nuevo Producto.
- **Salir.-** Esta pestaña permite salir del sistema.

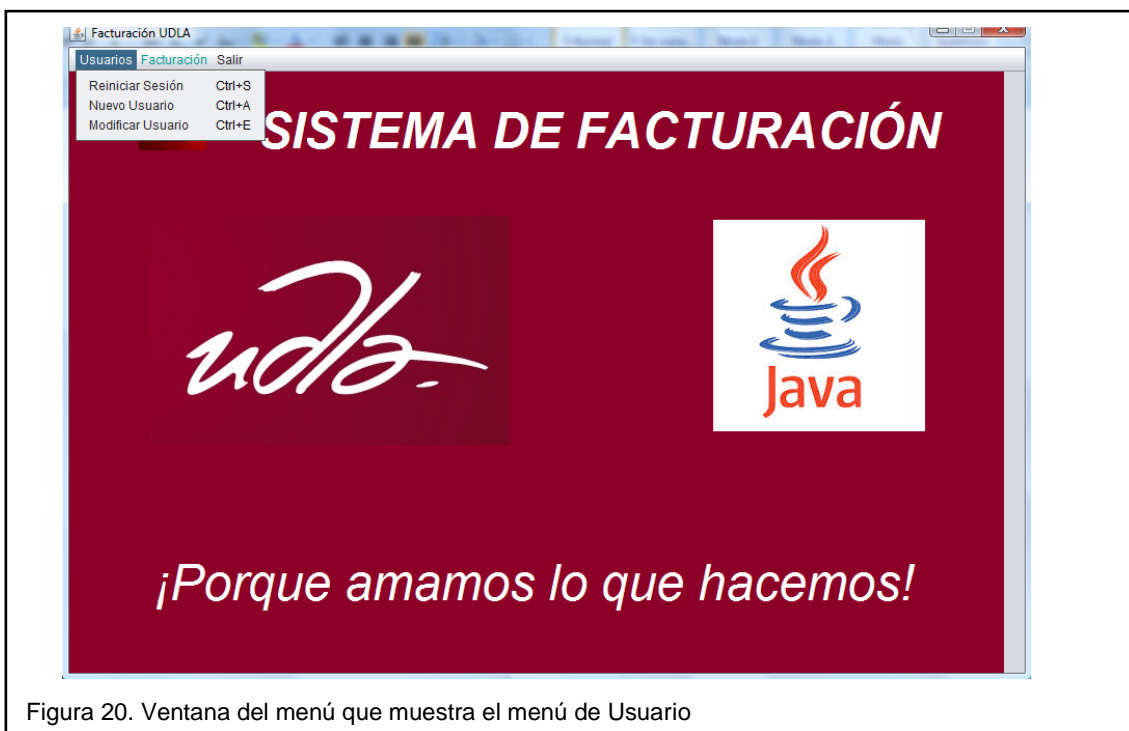


Figura 20. Ventana del menú que muestra el menú de Usuario

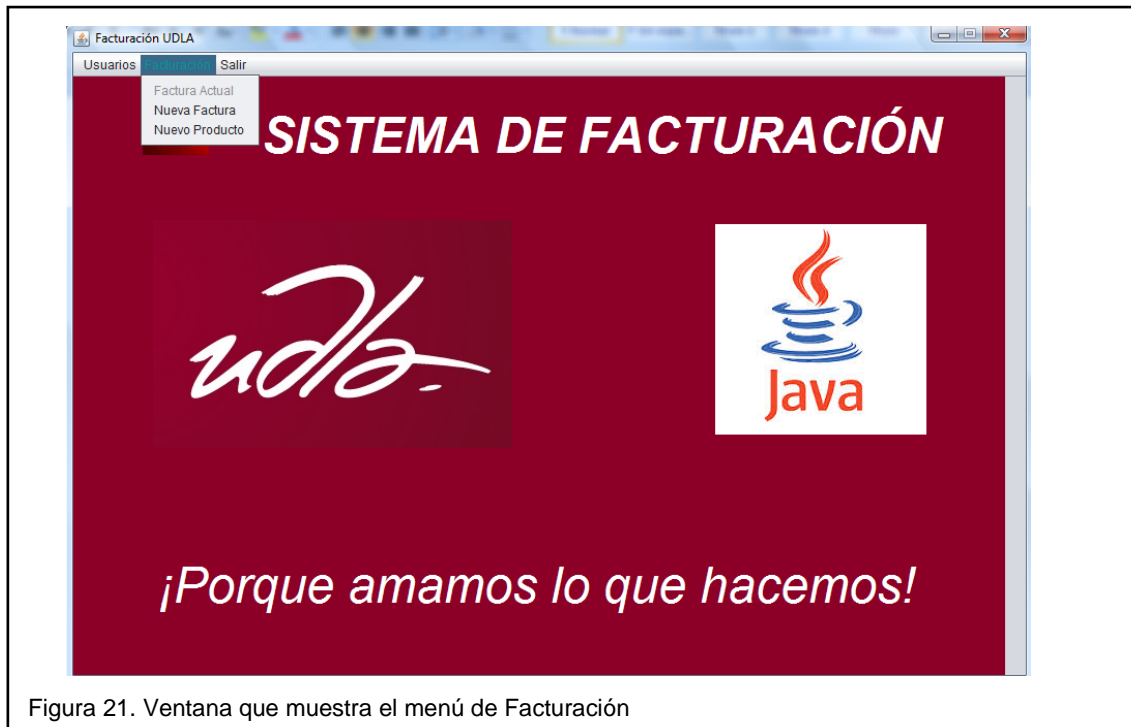


Figura 21. Ventana que muestra el menú de Facturación

6.1 Opciones del Menú Usuarios

Reiniciar Sesión.- Esta opción sirve para cerrar la sesión del usuario al darle click nos vuelve a desplegar la ventana de acceso donde nos pide ingresar nuevamente nuestro usuario y contraseña.

Nuevo Usuario.- Esta Opción nos permite crear un nuevo usuario, donde se nos mostrara la ventana de creación de usuario en la cual debemos ingresar todos los datos que se solicita y especificar si va a ser un usuario administrador o estándar, al finalizar se debe dar click en **AGRE**.

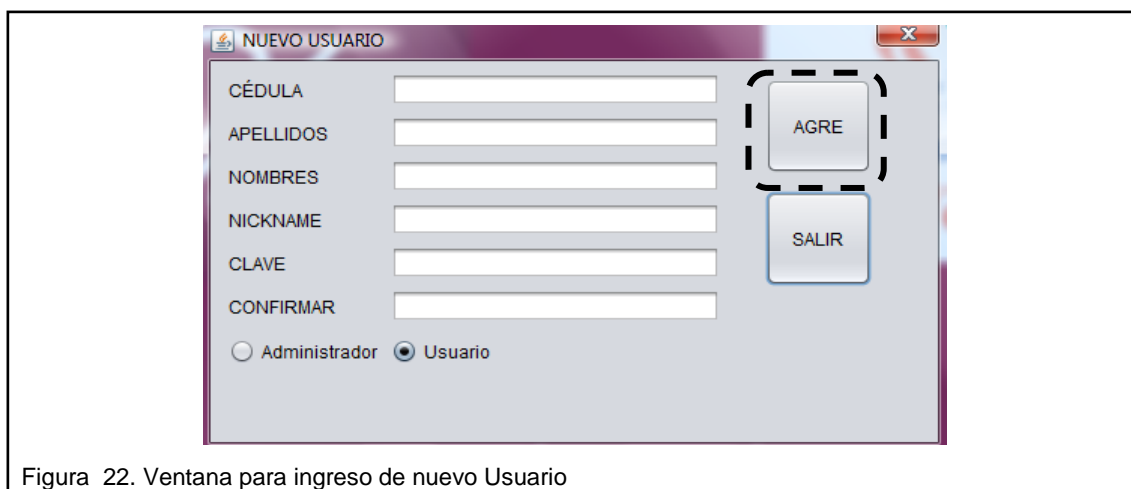


Figura 22. Ventana para ingreso de nuevo Usuario

Modificar Usuario.- Esta opción nos permite modificar usuarios que ya han sido creados en donde podemos buscar a esos usuarios por su número de cédula y una vez ya encontrado podemos modificar o editar los campos de la cédula, apellidos, nombres, nickname, clave y tipo, los cambios efectuados serán guardados al darle click en **ACTUALIZAR**.



Figura 23. Ventana de edición o modificación de usuario

6.2 Opciones del Menú Facturación

Factura Actual.- Esta opción nos muestra la factura actual que estamos realizando.

Nueva Factura.- Al darle click en esta opción procedemos a crear la factura para el cliente.

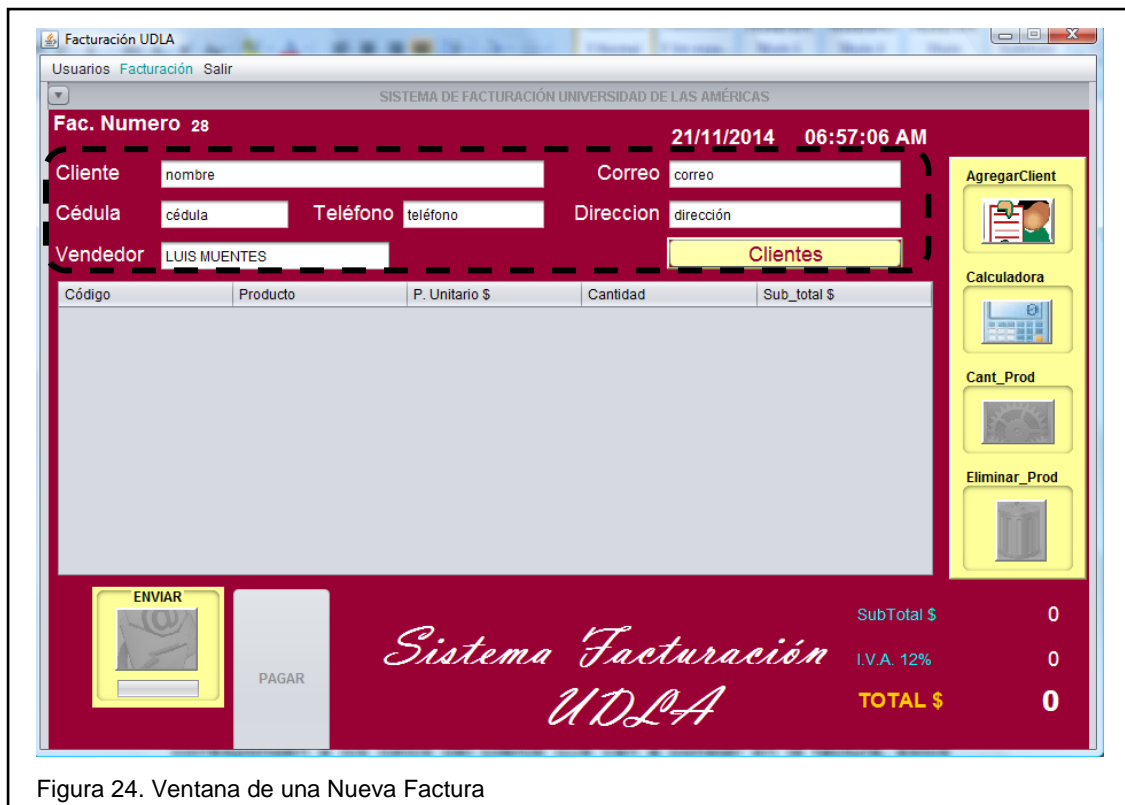


Figura 24. Ventana de una Nueva Factura

Como podemos observar en la imagen, esta es la ventana que corresponde a la creación de la factura para el cliente, donde los campos en la parte superior corresponden a los datos del cliente que van a constar en la factura, estos datos van a ser insertados de manera automática al darle click el botón **Cientes** e instantáneamente se mostrara la siguiente ventana.

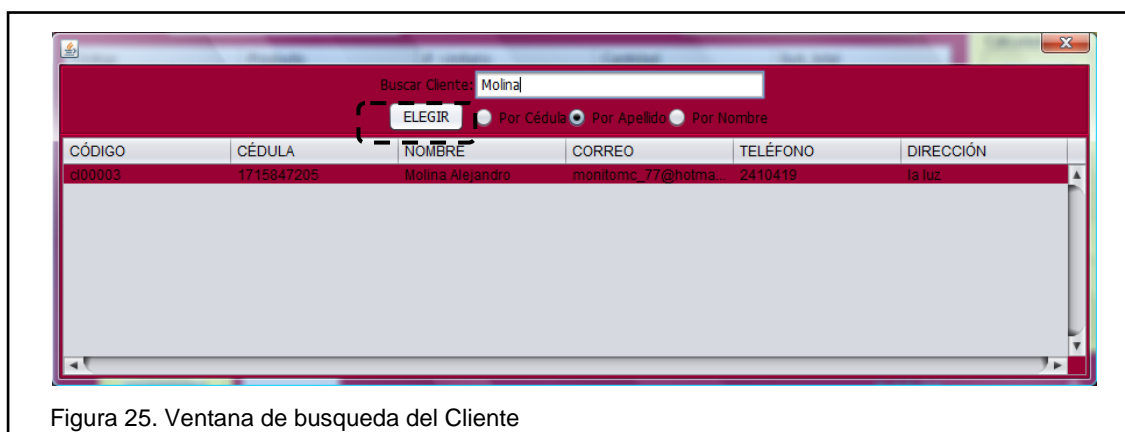


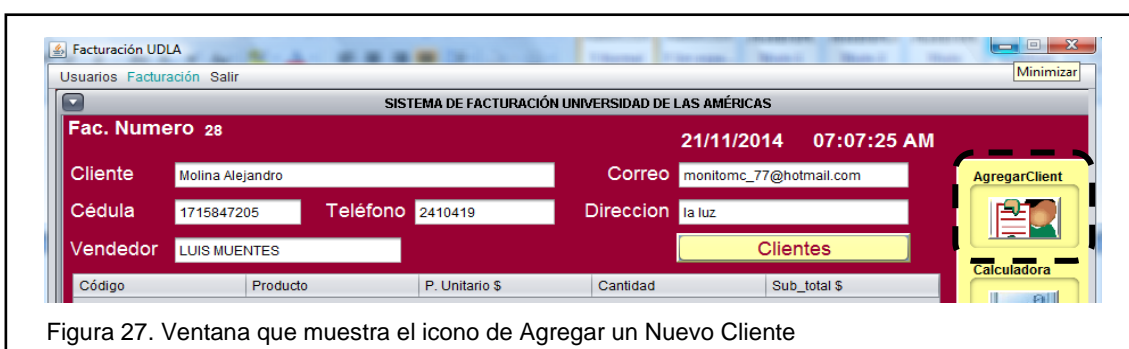
Figura 25. Ventana de búsqueda del Cliente

En la cual debemos primero buscar al cliente ya sea por su cedula, apellido o nombre una vez ya encontrado procedemos a seleccionarlo para luego darle click en el botón **ELEGIR** y así automáticamente nos aparecerán llenos los

campos correspondientes a la información del cliente como se observa en la imagen siguiente.



Si no se encuentra al cliente en el proceso de búsqueda que ejecuta el botón **Clientes**, quiere decir que es un cliente nuevo y no consta en la base de datos por lo tanto tendremos que agregar al nuevo cliente, para ello tendremos que ejecutar el botón **AgregarClient** dándole un click.



Donde se nos mostrara la siguiente ventana.



Figura 28. Ventana que muestra el registro de un Nuevo Cliente

Aquí tenemos la libertad de agregar a un nuevo cliente, para lo cual primero procedemos a dar un click en el botón **nuevo** aquel que tiene color verde, al realizar esto en primera instancia en el campo **CÓDIGO** nos aparecerá una serie que empieza con cl00001 si fuese el primer cliente que agregamos y cl00010 si fuese el décimo cliente agregado así sucesivamente, esta serie corresponde al código de identificación de cada cliente.

Lo segundo que se procederá a realizar es llenar todos los campos de información correspondientes al cliente como son: cédula, apellidos, nombres, correo, dirección y teléfono. Por último para guardar en la base de datos la información del nuevo cliente le damos click en el botón **Agregar**.

También podemos actualizar o editar a los clientes ya anteriormente agregados, para realizar esto vamos a depender del botón **buscar** el cual busca a los clientes por su cédula, es decir procedemos primero a escribir en el campo que corresponde, la cédula del cliente que queremos actualizar ya hecho esto procedemos a dar click en el botón **buscar**, ejecutado este proceso se ha encontrado el cliente deseado y se empezara a editar o actualizar cualquiera de los campos correspondientes a su información para finalizar se deben guardar los cambios realizados ejecutando con un click el botón **Actualizar**.



Figura 29. Ventana que muestra la búsqueda de un Cliente para ser editado o modificado



Figura 30. Ventana que muestra la edición o modificación de un cliente ya ingresado

El siguiente proceso a realizar es la lectura de los tags electrónicos que constan en cada producto que está comprando nuestro cliente, para llevar a cabo lo mencionado el cajero debe pasar los productos por el lector RFID e instantáneamente saltara la siguiente ventana.

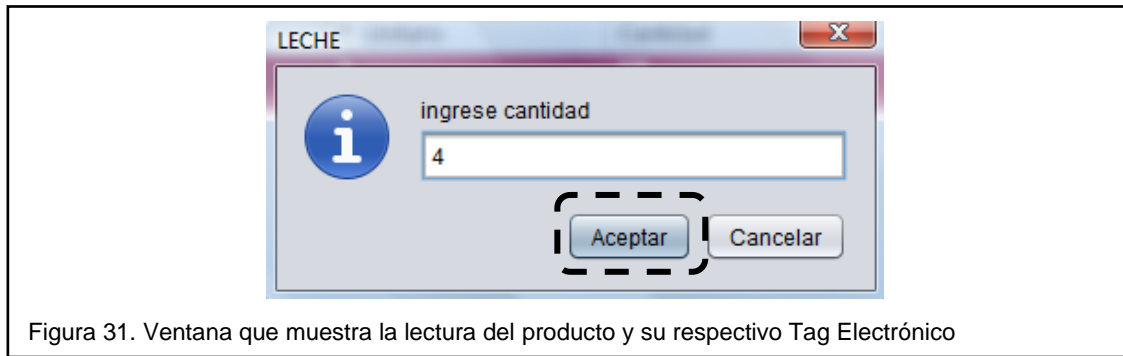


Figura 31. Ventana que muestra la lectura del producto y su respectivo Tag Electrónico

La cual nos especifica que producto es el que está leyendo el lector RFID y la cantidad que vamos a comprar. Para finalizar se le da click en **Aceptar** para que el producto y su respectiva cantidad a comprar sean ingresados al sistema de facturación que posteriormente va a ser entregado al cliente vía e-mail. Una vez ya que el lector RFID haya identificado todos los productos, la interfaz de nuestro sistema de facturación se verá como se muestra en la siguiente imagen.



Figura 32. Ventana que muestra el registro de los productos por su: código, tipo, precio unitario y precio total

El siguiente proceso a realizarse es la simulación del pago de los productos adquiridos por parte del cliente, donde el cajero ejecutara el botón **PAGAR**

dándole un click y se mostrara una ventana en la cual se realizara la simulación del pago con la tarjeta de crédito o debito del cliente.

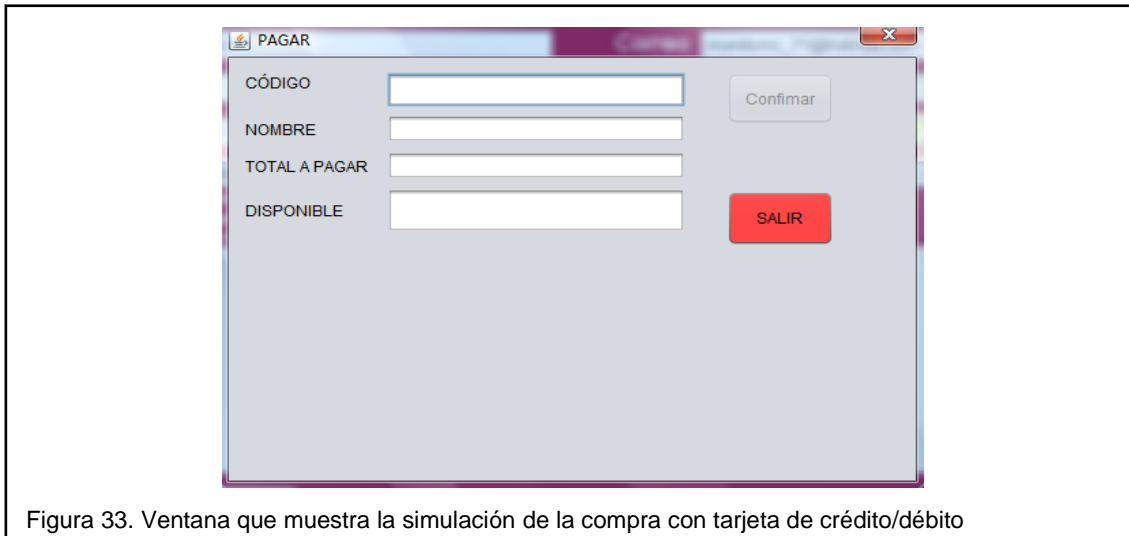


Figura 33. Ventana que muestra la simulación de la compra con tarjeta de crédito/débito

Para la ejecución de este proceso se ha programado un Tag RRFID que cumpla con la función de una tarjeta de crédito o debito y así poder realizar el pago. Para ello el cajero deberá pasar la tarjeta o Tag RFID por el lector y automáticamente se llenaran los campos respectivos a la información de la tarjeta y el total a pagar.

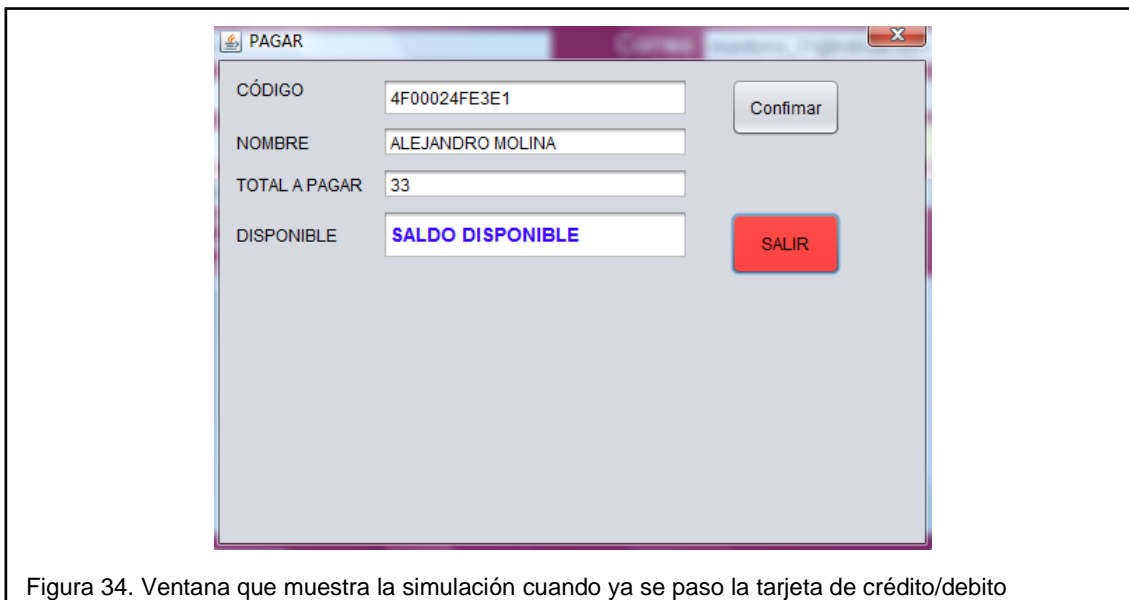


Figura 34. Ventana que muestra la simulación cuando ya se paso la tarjeta de crédito/debito

El cajero procederá a dar click en **Confirmar** para aceptar el pago donde se mostrara la siguiente ventana de confirmación en la cual se indica que el pago se ha realizado con éxito.

Se procede a dar click en Aceptar y se da por finalizado el proceso del pago por los productos que adquirió el cliente.

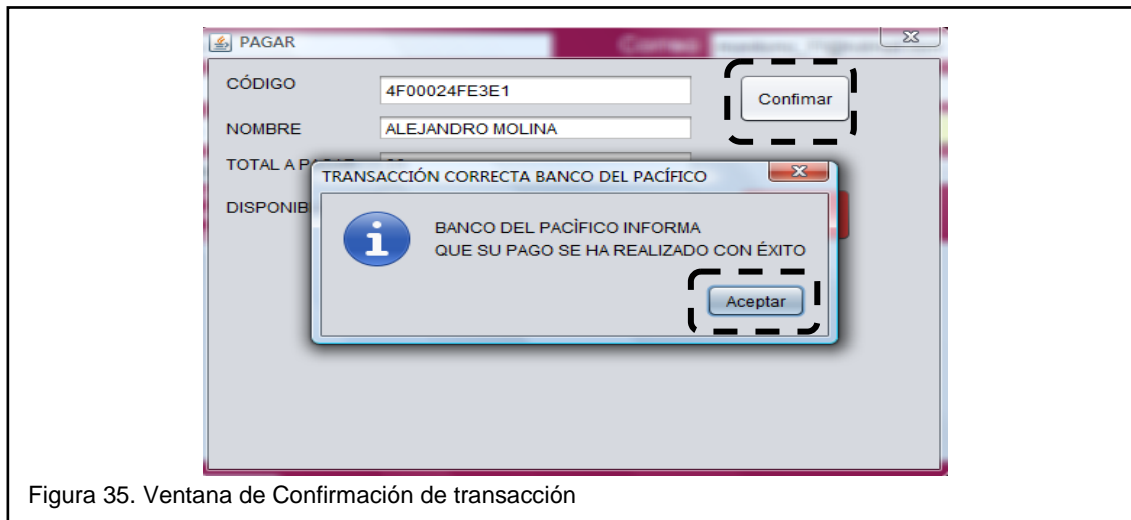


Figura 35. Ventana de Confirmación de transacción

El último proceso a realizarse es enviarle la factura al mail de nuestro cliente para lo cual el cajero debe hacer click en el botón **ENVIAR**.

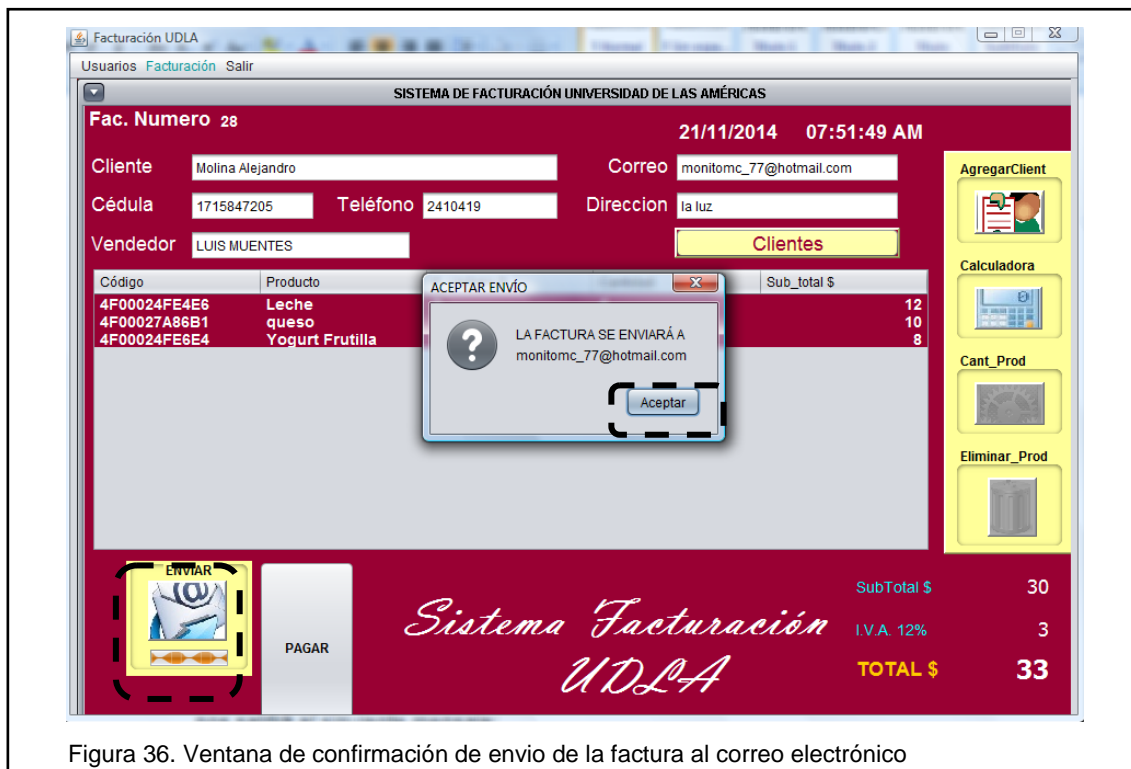
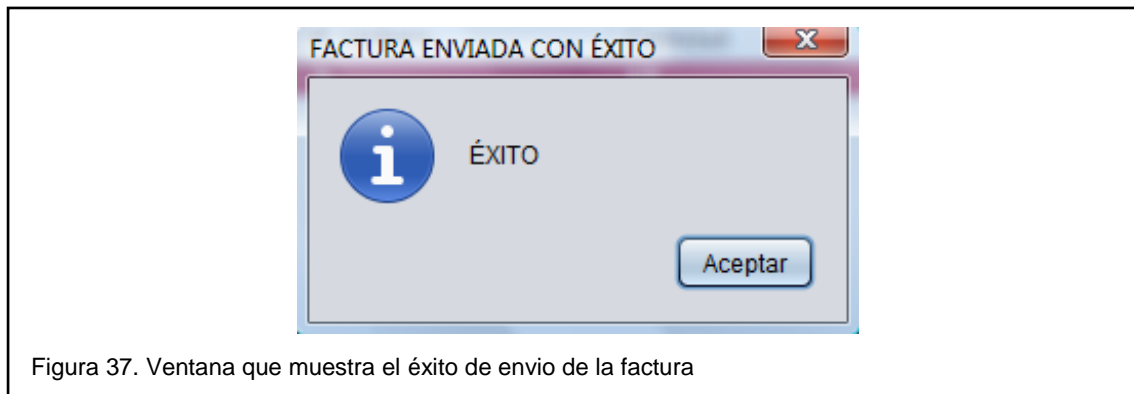


Figura 36. Ventana de confirmación de envío de la factura al correo electrónico

El cajero o usuario procederá a darle click en aceptar confirmando que a ese mail el cliente desea que le llegue su factura. Si la factura fue enviada con éxito nos saldrá el siguiente mensaje:



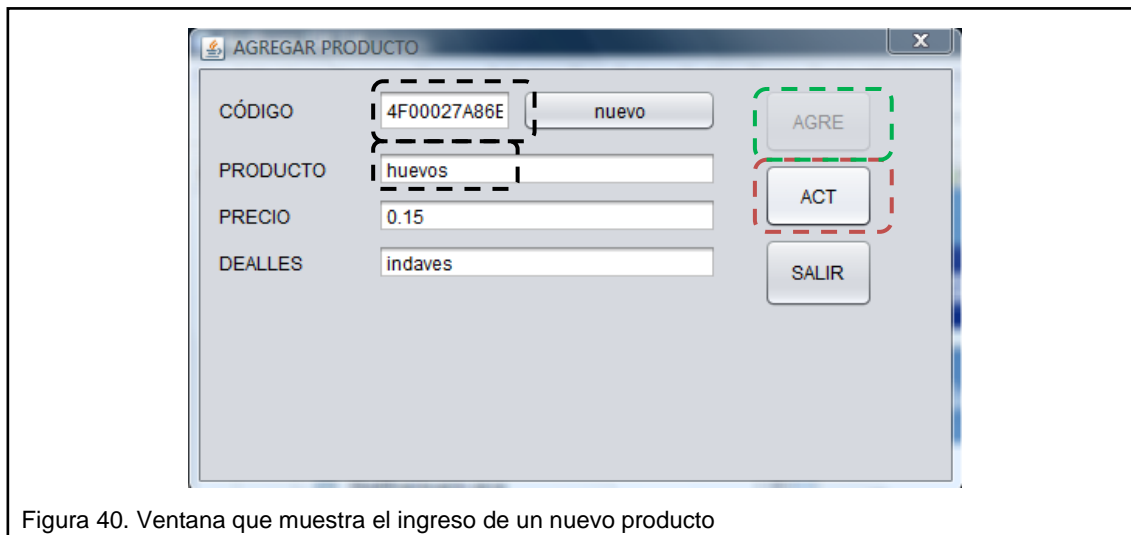
Posteriormente le se le dará click en Aceptar para terminar el proceso de facturación. A continuación veremos como se ve el mail de la factura cuando llega al correo electrónico de nuestro cliente.



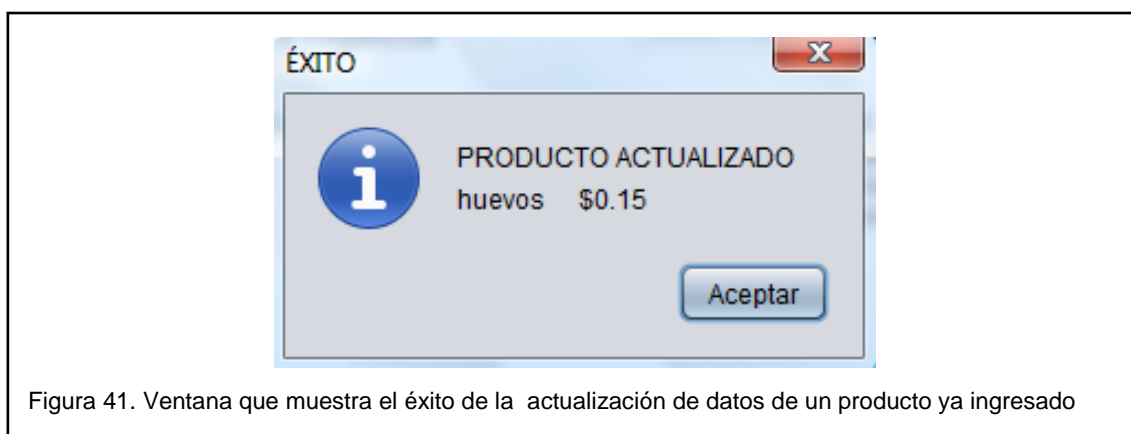
Si el cliente desea ver la factura tendrá que abrir el archivo adjunto directamente desde el e-mail o guardarlo en su computador o Smartphone, este tendrá una extensión .pdf y se mostrara como se ve en la siguiente imagen:



Nuevo Producto.- En esta opción solo puede ser utilizada por el usuario administrador el cual tiene un tipo de acceso total al sistema de facturación, de tal manera que se le permite al usuario ingresar un nuevo producto y esto se logra asociando el código de identificación de un nuevo tag con el producto que se quiera ingresar y a su vez también nos permite editar o actualizar productos ya anteriormente ingresados. La imagen a continuación muestra el código del tag en relación con el producto que se desea ingresar.



Donde se procederá a llenar los campos correspondientes al producto, en el campo precio el usuario deberá ingresar el precio unitario del producto, una vez ya llenos todos los campos se procederá a dar click en el botón **ACT**, si esta todo correcto se mostrara la siguiente ventana:



Esto sería para el caso en el que el usuario desee actualizar los campos de información de un producto ya anteriormente ingresado. Sin embargo puede que sea un nuevo código de tag al igual que el producto en ese momento se deberán llenar los campos correspondientes para así poderle dar click en el botón **AGRE** de tal manera que queden los datos guardados en la base de datos.

7. Conclusiones

La tecnología RFID tiene mucho potencial y definitivamente está revolucionando el mercado actual y en su futuro seguirá expuesto a cambios de automatización aplicados a diversas áreas dentro de las industrias alimenticia, textil, automotriz, logística, manufacturas, seguridad, medicina, sanidad, etc.

Lo que se obtiene con la implementación de la RFID es una identificación desatendida, automatizada y eficaz de los productos o suministros, este conjunto de características son las que buscan y necesitan las industrias para mantenerse de pie en un mercado que día a día se vuelve mucho más exigente.

A lo largo del proyecto se observó los diferentes resultados obtenidos los cuales se dieron en cuatro procesos diferentes, el primer resultado excelente se obtuvo en la lectura de los tags electrónicos adheridos a los productos, logrando así la apertura para el siguiente proceso en el cual se transmitía el número único de identificación del tag o etiqueta a una base de datos en MYSQL, de igual manera se obtuvieron resultados excelentes.

En el tercer proceso los resultados óptimos se lograron entre la comunicación de la PC y el lector RFID a través de las plataformas de programación de NETBEANS y MYSQL, en el cuarto proceso se obtuvo un resultado final de óptimas condiciones ya que se logró generar y enviar la factura de la compra de los productos al mail del cliente ya sea que este lo tenga en su smartphone u ordenador.

Al terminar este proyecto de tesis se determinó que la integración de las plataformas de programación MySQL y NetBeans con la tecnología RFID fue exitosa de tal manera que automatizó el proceso donde se ven implicadas las partes del hardware correspondientes al proyecto, es decir, todo lo referente a la autoidentificación de tags electrónicos que llevan nuestros productos del supermercado, lo cual se logra mediante nuestro lector RFID, cabe recalcar que la transmisión de datos entre el ordenador y los dispositivos RFID son excelentes a pesar de ser elementos ideales por motivos de costos, dado que

el presupuesto para equipos reales es demasiado elevado para ser implementados en un prototipo sin embargo el proyecto cumplió a cabalidad con todos sus objetivos y funcionalidades de identificación de radiofrecuencia.

8. Recomendaciones

Es importante recalcar que la plataforma de NetBeans no trae por defecto incorporada la librería win32com.dll y los archivos comm.jar, Javax.comm.properties, que son aquellos que nos ayudan con el manejo del puerto serie lo cual es la parte esencial del proyecto ya que nos permite la comunicación entre la tecnología RFID y el ordenador, es decir, al paso de información o datos entre el lector RFID y la Pc ya que el computador se comunica con el lector a través del puerto serie, motivo por el cual es vital copiar los archivos en los paths a los que corresponden tal como se lo hizo en el capítulo 5.

Sería muy importante que las nuevas versiones de NetBeans ya traigan incorporadas las librerías ya anteriormente mencionadas las cuales son necesarias para el manejo del puerto serie, esto sería de gran utilidad ya que evitaría hacer el proceso que se menciona en el capítulo 5 y adicional a esto cabe recalcar que no es fácil encontrar estas librerías en la internet.

REFERENCIAS

Miles, S., Sarma, S. y Williams, J. (2008). *RFID Technology and Applications*. Estados Unidos, Cambridge University Press.

Hansen, R. y Gillert, F. (2008). *RFID for the Optimization of Business Processes*, London England:Wiley.

Ahson, S. y Ilyas, M. (2008). *RFID Handbook Applications Technology Security and Privacy*, United State: CRC Press Inc.

Gerard, A. (2005). RFID: normativas y estándares. DATAcollection. Recuperado el 1 de Octubre del 2014 de <http://www.datacollection.eu/files/artFile/rfidspacial.pdf>

ID Inovations. (2007). Classic RFID module products. ID Series Datasheet. Recuperado el 4 de Julio del 2014 de http://www.id-innovations.com/httpdocs/EM%20moudule%20SERIES%202007-10-9_wfinal%20v22.pdf

Rantring, S. (2008). Soluciones tecnológicas integrales. Control de acceso-RFID. Recuperado el 12 de Febrero del 2014 de <http://www.rantring.com/rfid.htm>

Dipole. (2008). Lectores RFID. Recuperado el 15 de Marzo del 2014 de <http://www.dipolerfid.es/Productos/Lectores-RFID/Default>.

ANEXOS

Diagrama del proyecto con elementos ideales

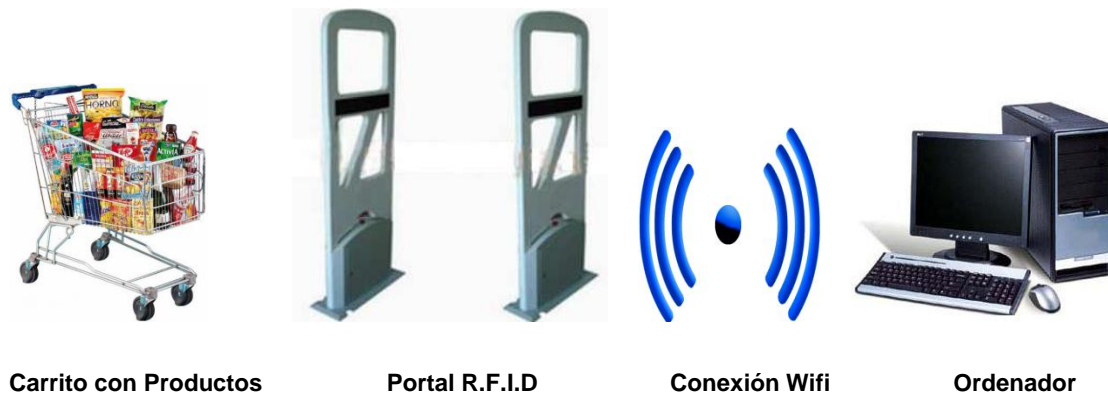


Figura No. 41 Diagrama del proyecto con elementos ideales

En este diagrama del proyecto el cual está implementado con elementos ideales se puede observar que en él participan procesos iguales que en nuestro prototipo, pero con ciertos cambios que son aquellos que marcan la diferencia en cuanto a su costo, los cuales son:

- Para que la antena del lector R.F.I.D. pueda leer el código único de identificación que lleva cada producto en su etiqueta electrónica, ya no será necesario que el cliente o cajero pase los productos uno por uno. Todos los códigos electrónicos de los productos podrán ser leídos al mismo tiempo, es decir, que el cliente podrá pasar todos los productos a la vez por el portal R.F.I.D.
- El portal R.F.I.D. sustituirá al lector implementado en nuestro prototipo debido a que tiene un mayor rango de recepción y transmisión de datos.
- La comunicación entre el portal R.F.I.D. y el ordenador se da mediante Wifi, esto ayuda a evitar el cableado y brinda versatilidad en la recepción de datos por parte del ordenador.

Características técnicas del portal R.F.I.D

- Protocolo de interfaz aerea: EPCglobal UHF Clase 1 Gen2 (18000-6C)
- Frecuencia de operación: UHF 860 – 960 MHz
- Potencia de transmisión: FCC 10-28.5 dBm, ETSI 10-27.5 dBm(nivel de potencia en decibelios en relación a un nivel de referencia de 1 mW)
- Max sensibilidad de recepción: -82 dBm
- Interfaz de datos: EPCglobal protocolo lector de bajo nivel (LLRP) v1.0.1
- Fuente de alimentación: En DC +24 V +/- 5% a traves de fuente de alimentación universal externa con conector de bloqueo.

Características físicas del portal R.F.I.D

- Dimensiones:1.70 m x 1.80 m
- Peso: 20 libras
- Temperatura de funcionamiento: -20 ° a 50 ° C

Caraterísticas de la antena:

- Elevación de ancho de haz: 80° +/- 3 °
- Azimut Ancho de haz: 60° +/- 3 °
- Polarización: Dual-Lineal de elementos en fase (DLPA)

A continuación se adjuntara las cotizaciones del portal RFID y los componentes que necesita para su funcionamiento:



Cotización No.
CAY10173
CAY10173

Cotizado a: Persona Natural	Fecha: 05/11/2013
Atención: Paul Molina	Dirección:
Validez de la Oferta: 7 días	Plazo de Entrega: 30 días laborables a partir del anticipo o de la orden de compra*
Condiciones de Pago: 60% anticipo, 40% contra entrega	Vendedor: Agustín Carrasco

ARTICULO(S)Y/SERVICIO(S)	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
Lector RFID MC3002Z-LC48BA Q E1 MOTOROLA	1	\$2.824,99	\$2.824,99
kit MC 3090 Z-LC48BA Q E1 MOTOROLA CABLE KIT	1	\$268,99	\$268,99
		SUBTOTAL	\$3.093,98
Observaciones: *El plazo de entrega puede variar según el stock en bodegas. El soporte técnico no incluye nuevos desarrollos ya que este deberá ser cobrado como adicional dependiendo al tiempo y complejidad que involucre.		Descuento Estudia ntill	\$50,00
		IVA 12%	\$365,28
		TOTAL	\$3409,26

Valor en letras: TRES MIL CUATROCIENTOS NUEVE DOLARES CON N26100 CENTAVOS

Cotizado por:
Firma y Sello

Aceptada por:
Firma y Sello



Amazonas N26-179 y Orellana, Ed Torrealka, Piso 8, Of. B
 Telf.: 02 2521 308 / 094 778 572 | info@caymansystems.com
 www.cayman systems.com

Gracias por confiar en nuestros productos y servicios



Empresa:	Fecha: 05/11/2013
Atención: Paul Molina	Teléfono:
E-mail:	Ext:
Dirección:	Celular: 09 9477 1673
Proyecto:	

Número de Parte	Descripción	Cantidad	Precio Unitario	Precio Total
FX7400-42315A30-WR	Easy to use and deploy, Gen2, POE, 4-port RFID reader for World Wide Use	1	1.228,31	\$ 1.228,31
PWRS-14000-260R	Power Supply (RFID Reader): 90-264VAC, 24VDC, 3.25A, 80 Watts	1	72,57	\$ 72,57
Total				\$ 1.300,88

Observaciones:

Precios no incluyen IVA

El tiempo de entrega: 45 días

Validez de la oferta: 15 días

Atentamente

Gino Zolezzi

Gerente Comercial Nacional

Protocolos y estándares de la tecnología RFID

Tabla No. 2 Estándares R.F.I.D

Banda de frecuencias	Nombre del estándar	Detalles
Menos de 135 KHz	ISO 18000-2	Uso aplicaciones menores
6,765 - 6,795 MHz	ISO 18000-3	Uso común
7,4 - 8,8 MHz	ISO 18000-3	Uso común
13,55 - 13,57 MHz	ISO 18000-3	Uso común
26,96 - 27,78 MHz	ISO 18000-3	Aplicaciones especiales
433MHz	ISO 18000-7	Tags activos en Asia
868 - 870 MHz	ISO 18000-6 A/B	Europa
902 - 928 MHz	AutoID clase 0/1	Norteamérica
860 - 960 MHz	EPCGlobal Gen2	Mundial
2400 - 2483 MHz	ISO 18000-4	Usado raramente en RFID
8725 - 5785 MHz	ISO 18000-5	Usado raramente en RFID

Tomado de Gerard, 2005 diciembre.

La ISO (The International Organization for Standardization) ha creado una serie de estándares R.F.I.D :

- La norma ISO 11784: define como se estructuran los datos en el tag.
- La norma ISO 11785: define el protocolo en la interfaz aire para los tags utilizados en sistemas de pago y tarjetas sin contactos (ISO 14443).
- La norma ISO 18047: ampliada por la norma ISO 18046, que regulaba las distintas evaluaciones de rendimiento que se puede hacer a tags y lectores.
- La norma ISO 7816: Tarjetas de indentificación, circuitos integrados.
- La norma ISO 9798: Técnicas de seguridad y autoindentificación.
- La norma ISO 11784: indentificación de animales.
- La norma ISO 15459: Identificación de unidades de transporte.

- Normas ISO 17458,17363,17364,17365,17366,17367: Aplicadas para cadenas de suministros.
- La norma ISO 18000-1 a 18000-7: RFID para gestion de objetos

El estándar EPC Global Gen2

(Código Electrónico de Producto Universal)

Otorga una regulación que permite una compatibilidad mundial de este protocolo en banda UHF. Este protocolo fue presentado al EPCGlobal el 1º de Octubre del 2014 y se aprobo en diciembre del mismo año, su producción comenzó en el segundo trimestre del año 2005.

Este permite leer 1500 tags/s en norteamérica y 600 tags/s en europa, la diferencia se debe principalmente al ancho de banda asignada a cada una de las regiones para el estándar.

Un punto muy importante es la selectividad de los tags. Se pueden seleccionar tags como si estuviéramos accediendo a una base de datos. Esto implica que si se tiene una producción completa de millones de objetos ya empaquetados, y se necesita localizar una parte de la producción pequeña, por ejemplo, se podra ordenar al lector que solo seleccione dichos tags. Luego un tag no será más que un objeto en una base de datos que se podrá leer, escribir, modificar, etc (Gerard, 2005 diciembre).

Algoritmos de Programación

Controladores ComSerial

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

package CONTROLADOR;

import INTERFACES.Factura;
import INTERFACES.PagarTarjeta;
import INTERFACES.RegistrarProducto;
import java.io.InputStream;
import javax.comm.CommPortIdentifier;
import javax.comm.SerialPort;
import javax.swing.JOptionPane;

/**
 * Clase que contiene métodos de carácter serial
 * @author Paúl Molina
 * @version 1.0
 */

/*
 *CommPortIdentifier = clase central para carácter con puertos
 *SerialPort = instancia de tipo clase (abstracta), permite la carácter
serial con protocolo rs-232
```

```

*OutputStream = JAVA API para enviar dato
*InputStream = JAVA API para recibir dato;
*byte[] buffer = se almacena los datos que llegan al pin de recepción del puerto
serie
*/

```

```

public class ComSerial extends Thread {

    nforma CommPortIdentifier IdPuerto;//clase propia de la librería nformació
para verificar puertos COM

    nforma SerialPort PuertoSerie;//clase propia de la librería nformació
comunicación serial

    public InputStream Entrada; //Esta clase abstracta es una superclase de
todas las clases que representan entrada de datos

    public byte[] buffer = new byte[1024];

    int flag = -1;//permite saber el momento en el que llega un dato al
InputStream(dato que llega por el pin de recepción)

    public static String codigotag = "";//se almacena ID de nforma

    public static String aux = "";//acumula byte a byte datos de un ID, se limpia
nforma de cada lectura

    //método de configuración de puerto serie
    public void ConfigSerial(String COM) {
        try {
            IdPuerto = CommPortIdentifier.getPortIdentifier(COM);
            //Identifica si puerto COM está disponible

            System.out.println("Puerto Configurado: " + COM );
            //Debugger

```

```

        if (IdPuerto.isCurrentlyOwned()) {
            JoptionPane.showMessageDialog(null, "PUERTO OCUPADO POR
OTRA APLICACIÓN...");
            //mensaje de error en caso de puerto ocupado
        } else {
            PuertoSerie = (SerialPort) IdPuerto.open("Comunic1", 2000);
            //genera una exigencia de procesamiento, se crea una APP llamada
Comunic en el administrdor de tareas de Windows

            PuertoSerie.setSerialPortParams(9600, SerialPort.DATABITS_8,
SerialPort.STOPBITS_1, SerialPort.PARITY_NONE);
            //configura parámetros de puerto serie

            Entrada = PuertoSerie.getInputStream();
            //define variable de almacenamoeto de datos entrantes
        }
    } catch (Exception e) {
        System.out.println("ERROR EN LA COMUNICACIÓN 1:" + e);
        //Mensaje de error en caso de existir
    }
}

//activa el Hilo de recepción, eel software siempre está "escuchando" el
puerto serie
public void RecepcionSerial() {

    try {
        this.start();//activa el hilo de recepción (siempre activo)
    } catch (Exception e) {
        System.out.println("ERROR EN LA COMUNICACIÓN 2 : " + e);
    }
}

```

```

}

//es necesario manipular los datos provnientes del puerto serie
//en 2 interfaces: Factura y Agregar Producto
//se utilizan banderas para indicar con qué jPanel se está trabajando
public static int bandera_factura;//es 1 cuando el usuario está facturando
public static int bandera_agregar;//es 1 cuando el usuario está agregando
nuevos productos
public static int bandera_pago;//es 1 cuando el usuario está pagando

//hilo de recepción de datos
@Override
public void run() {
    try {
        System.out.println("Puerto configurado correctamente. Listo para recibir
datos.");
        //Debugger
        while (true) {
            if ((flag = Entrada.read(buffer)) > -1) {
                codigotag = new String(buffer, 0, flag);//se recibe un carácter por
iteración
                aux = aux + codigotag;//acumulador de caracteres
                //LeerTag.txtRecibir.setText(aux);
                if(bandera_factura==1){//Actualiza Tabla en Factura
                    Factura.RF_ID(aux);
                }
                if(bandera_agregar==1){//Actualiza jTextField en Agregar
Productos
                    RegistrarProducto.inserta_producto(aux);
                }
                if(bandera_pago==1){//Actualiza jTextField en PagarTarjeta

```

```
        PagarTarjeta.inserta_informacion(aux);
    }
    System.out.println(bandera_factura+ " "+aux);
    System.out.println(bandera_agregar+ " "+aux);
    //Debugger para verificar banderas de comunicación
    aux=""; //limpieza de acumulador
    }
}

} catch (Exception e) {
    System.out.println("ERROR EN LA COMUNICACIÓN 3 :"+ e);
    //Debugger
}
}

}
```

Controlador Mail.java

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package CONTROLADOR;

import INTERFACES.Factura;
import java.util.Properties;
import java.util.logging.Level;
import java.util.logging.Logger;
```



```

import javax.activation.DataHandler;
import javax.activation.FileDataSource;
import javax.mail.BodyPart;
import javax.mail.Message;
import javax.mail.MessagingException;
import javax.mail.Session;
import javax.mail.Transport;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeBodyPart;
import javax.mail.internet.MimeMessage;
import javax.mail.internet.MimeMultipart;
import javax.swing.JOptionPane;

/**
 *
 * @author Paul Molina
 */
public class Mail {
    private final Properties properties = new Properties();
    private Session session;//clase abstracta que se utiliza para hacer LOGIN en
    cualquier actividad que lo requiera
        //en este caso en Gmail.
    public void init() {
        properties.put("mail.smtp.host", "smtp.gmail.com");//servodr de correo
    electronico
        properties.put("mail.smtp.starttls.enable", "true");//STARTTLS para IMAP y
    POP3 se usa para configurar puertos de conexion a internet
        properties.put("mail.smtp.port", 587);//se usa puerto para standar smtp
        properties.put("mail.smtp.mail.sender",
    "udla.facturacion@gmail.com");//mail del usuario que envia
        properties.put("mail.smtp.password", "udla1234");//clave de acceso

```

```

        properties.put("mail.smtp.user", "udla.facturacion@gmail.com");
        properties.put("mail.smtp.auth", "true");//para autenticacion de usuario y
contraseña
        session = Session.getDefaultInstance(properties);
    }

```

```

    public void adjunto(String destino,String asunto, String mensaje, String
nombre, String directorio) {

        try {
            Properties props = new Properties();
            props.put("mail.smtp.host", "smtp.gmail.com");//host y servidor
            props.setProperty("mail.smtp.starttls.enable", "true");//empieza
comunicacion con gmail
            props.setProperty("mail.smtp.port", "587");//puerto para standar SMTP
            props.setProperty("mail.smtp.user",
"udla.facturacion@gmail.com");//usuario
            props.setProperty("mail.smtp.auth", "true");//autenticacion

            Session session2= Session.getDefaultInstance(props, null);//verifica
LOGIN en gmail
            session2.setDebug(true);
            BodyPart texto = new MimeBodyPart();//se declara un tipo de dato de la
Clase BodyPart que permite estructurar un MAIL
            texto.setText(mensaje);//Se añade el texto proveniente del método
desde el cual se invocó e presente método (envia_mail ubicado en Clase
Factura)
            BodyPart adjunto = new MimeBodyPart();//se prepara para añadir dato
adjunto

```

```
    adjunto.setDataHandler(new DataHandler(new
FileDataSource(directorio+".pdf")));//se encuentra el PDF
    adjunto.setFileName(nombre+".pdf");//nombre del archivo con el que se
enviará el dato adjunto en el correo
    MimeMultipart multiParte = new MimeMultipart();//para estructurar varias
partes del mensaje

    multiParte.addBodyPart(texto);//añade el texto
    multiParte.addBodyPart(adjunto);//añade el documento PDF
    MimeMessage message = new MimeMessage(session2);//se enlaza
con Gmail
    // Se rellena el From
    message.setFrom(new
InternetAddress("udla.facturacion@gmail.com"));

    // Se rellenan los destinatarios
    message.addRecipient(Message.RecipientType.TO, new
InternetAddress(destino));

    // Se rellena el subject
    message.setSubject(asunto);

    // Se introduce el texto y el documento adjunto.
    message.setContent(multiParte);
    Transport t = session2.getTransport("smtp");
    t.connect("udla.facturacion@gmail.com","udla1234");//conexion a Gmail
    t.sendMessage(message,message.getAllRecipients());//envío de
mensaje
    t.close();
```

```

        JOptionPane.showMessageDialog(null, "ÉXITO","FACTURA ENVIADA
CON ÉXITO",JOptionPane.INFORMATION_MESSAGE);
        Factura.jProgressBar.setIndeterminate(false);
        Factura.jProgressBar.setValue(0);//se detiene barra de progreso en
Factura
    } catch (MessagingException ex) {
        JOptionPane.showMessageDialog(null, "VERIFIQUE DESTINO Y
CONEXIÓN A INTERNET\nMENSAJE NO ENVIADO","ERROR DE
ENVÍO",JOptionPane.ERROR_MESSAGE);//mensaje de error en caso de Fallo
de destino o no conexión Internet
        Factura.jProgressBar.setIndeterminate(false);
        Factura.jProgressBar.setValue(0);//se detiene barra de progreso en
Factura
        Logger.getLogger(Mail.class.getName()).log(Level.SEVERE, null, ex);
    }

}

}

```

Controlador PdfCreador.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package CONTROLADOR;

```

```

/***** START CODE
*****/

import com.itextpdf.text.BadElementException;
import com.itextpdf.text.BaseColor;
import com.itextpdf.text.Document;
import com.itextpdf.text.DocumentException;
import com.itextpdf.text.FontFactory;
import com.itextpdf.text.Image;
import com.itextpdf.text.Paragraph;
import com.itextpdf.text.pdf.PdfPTable;
import com.itextpdf.text.pdf.PdfWriter;
import java.awt.Font;
import java.io.*;
import java.net.MalformedURLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JTable;

/**
 *
 * @author Paúl Molina
 */
public class PdfCreador {

    static String Directorio= "C:\\Users\\ING. EDUARDO
MOLINA\\Desktop\\Tesis PMolina\\TESIS PAÚL
MOLINA\\SISTEMALECTURATAGS\\src\\IMAGENES";//directorio donde se
encuentran las imágenes que serán parte del documento adjunto

    public static void crea(String nombre,JTable tabla_productos, String subtotal,
String iva,

```

```

        String total, String numero, String comprador, String cedula, String
direccion,
        String telefono,String vendedor, String fecha, String hora) {

    try {
        int count = tabla_productos.getRowCount();//se recibe el número de
filas del jTable proveniente de Factura
        Document document=new Document(); //se crea el documento
        PdfWriter.getInstance(document,new
FileOutputStream(nombre+".pdf"));//se setea el nombre del documento
        document.open(); //se prepara para llenar el documento

        Image imagen = null;//instancia de tipo Image para imprimir image en
documento
        try {
            imagen =
Image.getInstance(Directorio+"//logo_deal.png");//encuentra imagen en
directorio

            imagen.setAlignment(Image.ALIGN_CENTER);//alineación al centro
            imagen.scalePercent(75);//escalamiento 75%
        } catch (BadElementException ex) {
            Logger.getLogger(PdfCreador.class.getName()).log(Level.SEVERE,
null, ex);
        } catch (MalformedURLException ex) {
            Logger.getLogger(PdfCreador.class.getName()).log(Level.SEVERE,
null, ex);
        } catch (IOException ex) {
            Logger.getLogger(PdfCreador.class.getName()).log(Level.SEVERE,
null, ex);
        }
        document.add(imagen);//se añade imangen
    }
}

```

```

document.add(new Paragraph("")); //se añade salto de linea
Paragraph parrafo0 = new Paragraph(); //crear nuevo párrafo
parrafo0.setAlignment(Paragraph.ALIGN_LEFT); //alinear al centro
parrafo0.setFont(FontFactory.getFont("Sans", 10, Font.BOLD,
BaseColor.BLACK)); //formato a párrafo
parrafo0.add("CLIENTE:           "+comprador+"\n"); //se añade
texto a párrafo
parrafo0.add("CÉDULA:           "+cedula+"\n"); //se añade
texto a párrafo
parrafo0.add("TELÉFONO:        "+telefono+"\n"); //se añade
texto a párrafo
parrafo0.add("DIRECCIÓN:       "+direccion+"\n"); //se añade
texto a párrafo
parrafo0.add("FECHA DE COMPRA:  "+fecha+"\n"); //se añade
texto a párrafo
parrafo0.add("HORA DE COMPRA:   "+hora+"\n"); //se añade texto
a párrafo
document.add(parrafo0); //se agrega párrafo a documento

Paragraph parrafoV = new Paragraph(); //crear nuevo párrafo
parrafoV.setAlignment(Paragraph.ALIGN_RIGHT); //alinear al centro
parrafoV.setFont(FontFactory.getFont("Sans", 10, Font.BOLD,
BaseColor.BLACK)); //formato a párrafo
parrafoV.add("VENDEDOR:      "+vendedor+"\n"); //se añade texto a
párrafo
document.add(parrafoV); //se agrega párrafo a documento

```

```

Paragraph parrafo1 = new Paragraph("\n");//crear nuevo párrafo
parrafo1.setAlignment(Paragraph.ALIGN_CENTER);//alinear al centro
parrafo1.setFont(FontFactory.getFont("Sans",20,Font.BOLD,
BaseColor.BLACK));//formato a párrafo
parrafo1.add("DETALLE DE COMPRA");//se añade texto a párrafo
document.add(parrafo1);//se agrega párrafo a documento

```

```

Paragraph parrafo2 = new Paragraph("\n");//crear nuevo párrafo
parrafo2.setAlignment(Paragraph.ALIGN_LEFT);//alinear al izquierda
parrafo2.setFont(FontFactory.getFont("Sans",10,Font.BOLD,
BaseColor.BLACK));//formato a párrafo
parrafo2.add("Factura número "+numero);//se añade texto a párrafo
document.add(parrafo2);//se agrega párrafo a documento

```

```

PdfPTable tab=new PdfPTable(5);//crear nueva tabla
tab.addCell("CODIGO");
tab.addCell("PRODUCTO");
tab.addCell("PRECIO
$");//*****
tab.addCell("CANTIDAD");
tab.addCell("SUBTOTAL $");//encabezado de tabla
//se generan filas cada 5 celdas
for(int i=0;i<count;i++){
    Object obj1 = GetData(tabla_productos, i, 0);
    Object obj2 = GetData(tabla_productos, i, 1);
    Object obj3 = GetData(tabla_productos, i, 2);
    Object obj4 = GetData(tabla_productos, i, 3);
    Object obj5 = GetData(tabla_productos, i, 4);//capturar celdas de la
tabla de Factura

    String value1=obj1.toString();

```



```

String value2=obj2.toString();
String value3=obj3.toString();
String value4=obj4.toString();
String value5=obj5.toString();//casting a String

tab.addCell(value1);
tab.addCell(value2);
tab.addCell(value3);
tab.addCell(value4);
tab.addCell(value5);//se añaden los valores a la table creada
anteriormente
    }
document.add(new Paragraph("")); //se añade salto de linea
document.add(tab); //se añade la tabla al documento adjunto

document.add(new Paragraph("Subtotal  $" +subtotal));
document.add(new Paragraph("Iva  $" +iva));
document.add(new Paragraph("TOTAL  $" +total+"\n")); //se añaden
parámetros de dinero a pagar al documento adjunto
//provenientes de Factura

Image imagen_amar = null; //instancia de tipo Image para imprimir imagen
en documento adjunto
    try {
        imagen_amar = Image.getInstance(Directorio+"//amar.png"); //se
encuentra imagen en directorio
        imagen_amar.setAlignment(Image.ALIGN_CENTER); //alinear imagen
al centro
        imagen_amar.scalePercent(50); //escala de reducción 50%
    } catch (BadElementException ex) {

```

```

        Logger.getLogger(PdfCreador.class.getName()).log(Level.SEVERE,
null, ex);
    } catch (MalformedURLException ex) {
        Logger.getLogger(PdfCreador.class.getName()).log(Level.SEVERE,
null, ex);
    } catch (IOException ex) {
        Logger.getLogger(PdfCreador.class.getName()).log(Level.SEVERE,
null, ex);
    }
    document.add(new Paragraph("\n\n ")); //salto de línea
    document.add(imagen_amar); //se añade imagen a documento

    document.close();//fin de documento adjunto

} catch (FileNotFoundException ex) {
    Logger.getLogger(PdfCreador.class.getName()).log(Level.SEVERE,
null, ex);
} catch (DocumentException ex) {
    Logger.getLogger(PdfCreador.class.getName()).log(Level.SEVERE,
null, ex);
}
}

    public static Object GetData(JTable table, int row_index, int col_index){
return table.getModel().getValueAt(row_index, col_index); //se utiliza para
trabajar con el Jtable proveniente de Factura
//y obtener la posición de una celda
específica
}
}

```

```

/***** END
CODE*****/

```

Controlador base_datos.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package CONTROLADOR;

import java.sql.*;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import javax.swing.JOptionPane;
/**
 *
 * @author Paúl Molina
 */
public class base_datos {
    public Statement s ;

    //Método para configuración de conexión con servidor SQL
    //tiene que ecutarse siempre antes de cualquier otro método de esta clase
    public void inicio() {
        try {
            String url="jdbc:mysql://localhost/tesis_1";//se levanta el servidor jdbc
            para hacer conexión con base de datos
            String usuario="root";

```

String clave ="root";//usuario y clave de instalación del servidor sql en la PC

```

    DriverManager.registerDriver(new org.gjt.mm.mysql.Driver());//se
instancia a la clase que comanda la interacción con base de datos
    System.out.println("driver Mysql cargando...");
    Connection conexion =
DriverManager.getConnection(url,usuario,clave);//se verifica usuario y clave de
servidor sql
    System.out.println("conexion abierta con "+url+"...");
    s= conexion.createStatement();//esta línea de código crea un puente o
traductor que permite escribir Querys desde java
    System.out.println("sentencia creada...\n");

    } catch (SQLException ex) {
        System.out.println("error1");
    }
}

//*****
****
//*****
//*****
//lógica de comportamiento para PRODUCTOS en base de datos

//Verifica si existe un producto al intentar
//guardar uno nuevo
//aumenta en 1 un contador por cada coincidencia
public int verificar_existencia_productos(String parametro){

```

```

int cont=0;
try {
    ResultSet rs = s.executeQuery("SELECT codigo FROM productos
where codigo like '"+parametro+"%';");
    while (rs.next()) {
        cont=cont+1;
    }
} catch (SQLException ex) {
    Logger.getLogger(base_datos.class.getName()).log(Level.SEVERE,
null, ex);
}
return cont;
}

```

```

//Altera el valor de las celdas de la tabla productos en Base de datos
//utiliza la respectiva Query con el comand UPDATE
public void actualizar_productos(String codigo, String nombre, String precio,
String extras){
    try {
        s.executeUpdate("UPDATE productos set nombre = " +nombre+"
where codigo = '"+codigo+"";");
        s.executeUpdate("UPDATE productos set precio = " +precio+" where
codigo = '"+codigo+"";");
        s.executeUpdate("UPDATE productos set extras = " +extras+" where
codigo = '"+codigo+"";");

    } catch (Exception ex) {

```

```

        System.out.println("error metodo insertar...");
    }
}

//Crea un nuevo producto
//utiliza la respectiva Query con el comand INSERT
public void insertar_producto(String codigo, String nombre, String precio,
String extras){

    try {
        s.executeUpdate("insert into productos(codigo,nombre,precio,extras)"
            + "values("
            + ""+codigo+","
            + ""+nombre+","
            + ""+precio+","
            + ""+extras+");");
    } catch (Exception ex) {
        System.out.println("error metodo insertar...");
    }
}

//Obtiene los valores de las celdas de la tabla PRODUCTOS
//a través de la generación de una matriz donde se almacenarían los datos
//para cada columna se crea un vector unidimensional
//para cada coincidencia se aumenta en 1 el número de filas de la matriz tipo
Object
//utiliza la respectiva Query con el comando SELECT ----- FROM
public Object[][] imprime_productos(String codigos){
    Object retorno_productos[][];
    String[] nombre=new String[40];
    String[] precio=new String[10];

```

```

int cont=0;
String codigo="";
if(codigos.length()>12){
    codigo=codigos.substring(1, 13);
    System.out.println(codigos);
    System.out.println(codigo);
}else {
    return null;
}

try {
    ResultSet rs = s.executeQuery("SELECT nombre FROM productos
where codigo like '"+codigo+"");
    while (rs.next()) {
        nombre[cont] = rs.getString("nombre");
        System.out.println(nombre[cont] + "\n");
        cont=cont+1;
    }

    rs = s.executeQuery("SELECT precio FROM productos where codigo
like '"+codigo+"");
    cont=0;
    while (rs.next()) {
        precio[cont] = rs.getString("precio");
        System.out.println(precio[cont] + "\n");
        cont=cont+1;
    }

    System.out.println("conexion cerrada");
} catch (SQLException ex) {

```

```

        JOptionPane.showMessageDialog(null, "No existe en base de datos",
"Error", JOptionPane.ERROR_MESSAGE);
        System.out.println("error2");
    }
    Object productos[][] = new Object[cont][5]; //SE DECLARA LA MATRIZ
    if(cont>0){
        for(int i=0;i<cont;i++){ //SE ALMACENAN DATOS EN EL CASO DE QUE
EXISTIENSEN(CONT>0)
            productos[i][0] = codigo;
            productos[i][1] = nombre[i];
            productos[i][2] = precio[i];
            int bandera_numeros=0;
            String cantidad = "";
            Pattern p = Pattern.compile("[0-9]"); //VALIDACIÓN SÓLO NUMEROS,
DECLARACIÓN DE PATRÓN

            while(bandera_numeros==0){
                cantidad = JOptionPane.showInputDialog(null,"ingrese
cantidad", nombre[i].toUpperCase(), JOptionPane.INFORMATION_MESSAGE);
                if(cantidad==null){
                    cantidad="0";
                    bandera_numeros=0;
                }
                for(int j = 0;j<cantidad.length();j++){
                    int caracter=cantidad.charAt(j);
                    if(caracter<48 || caracter>57 || j>3){
                        bandera_numeros=0;
                        break;
                    }else{
                        bandera_numeros=1;
                    }
                }
            }
        }
    }

```



```

        System.out.println("*****CHAR"+caracter);
    }
    if(cantidad==null){
        cantidad="0";
        bandera_numeros=0;
    }
    System.out.println("cantidad    "+cantidad);//SE INGRESA
CANTIDAD

    Matcher m = p.matcher(cantidad);//SE VERIFICA PATRÓN
    System.out.println("match    "+m);
    //DEBUG PARA VERIFICAR SI EXISTE COINCIDENCIA

    //PROCESO DE VALIDACIÓN
    if (m.find() && bandera_numeros==1) {
        if (Integer.parseInt(cantidad)>100 || "".equals(cantidad)){
            System.out.println("*****ENTRO A VALIDACION > 100,
VACIO");
            JOptionPane.showMessageDialog(null, "Ingrese valores
menores a 100");
            bandera_numeros=0;
        }
        if (Integer.parseInt(cantidad)<100 && !"".equals(cantidad)){
            System.out.println("*****cumple");
            bandera_numeros=1;
        }
    }

    }else{
        System.out.println("*****no hay coincidencia");
        JOptionPane.showMessageDialog(null, "Por favor, sólo
números");
    }
}

```

```

        bandera_numeros=0;
    }
}
    productos[i][3] = cantidad;//SE ASIGNA LA CANTIDAD INGRESADA
    productos[i][4] =
Integer.parseInt(cantidad)*Integer.parseInt(precio[i]);//PRECIO*CANTIDAD
}
    retorno_productos=productos;

}else {
    retorno_productos=null;
    JOptionPane.showMessageDialog(null, "PRODUCTO NO
REGISTRADO","ERROR",JOptionPane.ERROR_MESSAGE);
}
    System.out.println("*****"+cont);
    //DEBUG PARA VERIFICAR COINCIDENCIAS
    return retorno_productos;
}

//Obtiene los valores de las celdas de la tabla PRODUCTOS
//a través de la generación de un vector donde se almacenarían los datos
//utiliza la respectiva Query con el comando SELECT ----- FROM
public Object[][] imprime_productos_actualizar(String codigo){
    System.out.println("EN LA BASE");
    Object retorno_productos[][];
    String[] nombre=new String[40];
    String[] precio=new String[40];
    String[] extra=new String[40];
    int cont=0;

    try {

```

```

//se recalca que el contador siempre será 1
ResultSet rs = s.executeQuery("SELECT nombre FROM productos
where codigo like '"+codigo+"'");
while (rs.next()) {
    nombre[cont] = rs.getString("nombre");
    System.out.println(nombre[cont] + "\n");
    cont=cont+1;
}

rs = s.executeQuery("SELECT precio FROM productos where codigo
like '"+codigo+"'");
cont=0;
while (rs.next()) {
    precio[cont] = rs.getString("precio");//
    System.out.println(precio[cont] + "\n");
    cont=cont+1;
}

rs = s.executeQuery("SELECT extras FROM productos where codigo
like '"+codigo+"'");
cont=0;
while (rs.next()) {
    extra[cont] = rs.getString("extras");
    System.out.println(extra[cont] + "\n");
    cont=cont+1;
}

System.out.println("conexion cerrada");
} catch (SQLException ex) {
    JOptionPane.showMessageDialog(null, "No existe en base de datos",
>Error", JOptionPane.ERROR_MESSAGE);

```

```

        System.out.println("error2");
    }
    Object productos[][] = new Object[cont][4]; //declaración de matriz 1*4
    if(cont>0){
        for(int i=0;i<cont;i++){
            productos[i][0] = codigo;
            productos[i][1] = nombre[i];
            productos[i][2] = precio[i];
            productos[i][3] = extra[i]; //se asignan valores
        }
        retorno_productos=productos;

    }else {
        retorno_productos=null;
        JOptionPane.showMessageDialog(null, "repita"); //en el caso de falla de
lectura desde puerto serie
    }
    System.out.println("*****"+cont);
    return retorno_productos;
}

//*****
****

//*****
//*****

//FIN DE LÓGICA DE PRODUCTOS EN BASE DE DATOS

```

```

//*****
****

//*****
//*****

//lógica de comportamiento para TARJETAS en base de datos

//Verifica si existe un CLIENTE CON ESTA TARJETA al intentar
//guardar uno nuevo
//aumenta en 1 un contdor por cada coincidencia
public int verificar_existencia_tarjeta(String parametro){
    int cont=0;
    try {
        ResultSet rs = s.executeQuery("SELECT codigo FROM tarjetas where
codigo like '"+parametro+"%';");
        while (rs.next()) {
            cont=cont+1;
        }
    } catch (SQLException ex) {
        Logger.getLogger(base_datos.class.getName()).log(Level.SEVERE,
null, ex);
    }
    return cont;
}

//Verifica si existe un SALDO disponible al intentar

```

```

//pagar
public int obtener_saldo_tarjeta(String codigo){
    String saldo=null;
    try {
        ResultSet rs = s.executeQuery("SELECT saldo FROM tarjetas where
codigo like '"+codigo+"");
        while (rs.next()) {
            saldo = rs.getString("saldo");
        }
    } catch (SQLException ex) {
        Logger.getLogger(base_datos.class.getName()).log(Level.SEVERE,
null, ex);
    }
    int saldo_num = Integer.parseInt(saldo);
    return saldo_num;
}

```

```

//Obtiene NOMBRE
//al pagar
public String obtener_nombre_tarjeta(String codigo){
    String nombre=null;
    try {
        ResultSet rs = s.executeQuery("SELECT nombre FROM tarjetas where
codigo like '"+codigo+"");
        while (rs.next()) {
            nombre = rs.getString("nombre");
        }
    } catch (SQLException ex) {

```

```

        Logger.getLogger(base_datos.class.getName()).log(Level.SEVERE,
null, ex);
    }
    return nombre;

}

```

```

//Obtiene APELLIDO
//al pagar
public String obtener_apellido_tarjeta(String codigo){
    String apellido=null;
    try {
        ResultSet rs = s.executeQuery("SELECT apellidos FROM tarjetas
where codigo like '"+codigo+"");
        while (rs.next()) {
            apellido = rs.getString("apellidos");
        }
    } catch (SQLException ex) {
        Logger.getLogger(base_datos.class.getName()).log(Level.SEVERE,
null, ex);
    }
    return apellido;

}

```

```

//CAACTUALIZA TARJETAS
//utiliza la respectiva Query con el comand UPDATE
public void insertar_nuevosaldo(String codigo, int saldo){

```

```

try {
    s.executeUpdate("Update tarjetas set saldo = "+saldo+" where codigo
like '"+codigo+"");
    JOptionPane.showMessageDialog(null, "BANCO DEL PACÍFICO
INFORMA\nQUE SU PAGO SE HA REALIZADO CON
ÉXITO","TRANSACCIÓN CORRECTA BANCO DEL
PACÍFICO",JOptionPane.INFORMATION_MESSAGE);
} catch (Exception ex) {
    System.out.println("error metodo insertar...");
}
}

```

```
//*****
```

```
****
```

```
//*****
```

```
//*****
```

```
//FIN DE LÓGICA DE TARJETAS EN BASE DE DATOS
```

```
//*****
```

```
****
```

```
//*****
```

```
//*****
```

```
//lógica de comportamiento para CLIENTES en base de datos
```



```

//Verifica si existe un producto al intentar
//guardar uno nuevo desde REGISTRO CLIENTE
//aumenta en 1 un contdor por cada coincidencia
public int verificar_existencia_clientes(String parametro){
    int cont=0;
    try {
        ResultSet rs = s.executeQuery("SELECT cedula FROM clientes where
cedula like '"+parametro+"%");
        while (rs.next()) {
            cont=cont+1;
        }
    } catch (SQLException ex) {
        Logger.getLogger(base_datos.class.getName()).log(Level.SEVERE,
null, ex);
    }
    return cont;
}

```

```

//Ordena la tabla de clientes según el código y devuelve el último,
//se utiliza para que no existan cleintes con igual código
public int obtener_ultimo_codigo(){
    String codigo=null;
    try {
        ResultSet rs = s.executeQuery("SELECT codigo FROM clientes
ORDER BY codigo DESC LIMIT 1;");
        while (rs.next()) {
            codigo = rs.getString("codigo");

```

```

    }
} catch (SQLException ex) {
    Logger.getLogger(base_datos.class.getName()).log(Level.SEVERE,
null, ex);
}
codigo = codigo.substring(2, 7);
int code = Integer.parseInt(codigo);
return code;
}

```

//INSERTA UN CLIENTE CON LOS RESPECTIVO PARÁMETROS EN LA
TABLA CLIENTES

```

//Utiliza la QUERY INSERT INTO
public void insertar_cliente(String cedula, String nombres, String apellido,
String correo, String direccion, String telefono, String codigo){
    try {
        s.executeUpdate("insert into
clientes(cedula,nombres,apellido,correo,direccion,telefono,codigo)"
+ "values("
+ ""+cedula+","
+ ""+nombres+","
+ ""+apellido+","
+ ""+correo+","
+ ""+direccion+","
+ ""+telefono+","
+ ""+codigo+");");
    } catch (Exception ex) {

```

```

        System.out.println("error metodo insertar...");
    }
}

//Actualiza los datos de la fila respectiva de la tabla clientes
//Utiliza la Query UPDATE
public void actualizar_cliente(String cedula, String nombres, String apellido,
String correo, String direccion, String telefono, String codigo){
    try {
        s.executeUpdate("Update clientes set nombres = " +nombres+" where
cedula = "+cedula+";");
        s.executeUpdate("Update clientes set apellido = " +apellido+" where
cedula = "+cedula+";");
        s.executeUpdate("Update clientes set correo = " +correo+" where
cedula = "+cedula+";");
        s.executeUpdate("Update clientes set direccion = " +direccion+" where
cedula = "+cedula+";");
        s.executeUpdate("Update clientes set telefono = " +telefono+" where
cedula = "+cedula+";");

    } catch (Exception ex) {
        System.out.println("error metodo insertar...");
    }
}

//Devuelve una MATRIZ donde constan todos los clientes que coincidan con
el parámetro de búsqueda
//se utiliza una matriz porque la interfaz que controla este método tiene una
tabla de visualización de clientes

```

//el método que invoca este método se encuentra en el botón CLIENTES de FACTURA

```

public Object[][] imprime_clientes(String tipo, String parametro){
    String[] cedula= new String[150];
    String[] nombre=new String[150];
    String[] codigo=new String[150];
    String[] apellido=new String[150];
    String[] correo=new String[150];
    String[] direccion=new String[150];
    String[] telefono=new String[150];
    int cont=0;

    try {

        ResultSet rs = s.executeQuery("SELECT cedula FROM clientes where
"+ tipo + " like '"+parametro+"%'");
        while (rs.next()) {
            cedula[cont] = rs.getString("cedula");
            //System.out.println(cedula[cont] + "\n");
            cont=cont+1;
        }

        rs = s.executeQuery("SELECT nombres FROM clientes where "+ tipo +
" like '"+parametro+"%'");
        cont=0;
        while (rs.next()) {
            nombre[cont] = rs.getString("nombres");
            //System.out.println(nombre[cont] + "\n");
            cont=cont+1;
        }
    }
}

```

```
rs = s.executeQuery("SELECT apellido FROM clientes where "+ tipo + "
like '"+parametro+"%';");
cont=0;
while (rs.next()) {
    apellido[cont] = rs.getString("apellido");
    //System.out.println(apellido[cont] + "\n");
    cont=cont+1;
}
```

```
rs = s.executeQuery("SELECT correo FROM clientes where "+ tipo + "
like '"+parametro+"%';");
cont=0;
while (rs.next()) {
    correo[cont] = rs.getString("correo");
    //System.out.println(correo[cont] + "\n");
    cont=cont+1;
}
```

```
rs = s.executeQuery("SELECT direccion FROM clientes where "+ tipo + "
" like '"+parametro+"%';");
cont=0;
while (rs.next()) {
    direccion[cont] = rs.getString("direccion");
    //System.out.println(direccion[cont] + "\n");
    cont=cont+1;
}
```

```
rs = s.executeQuery("SELECT telefono FROM clientes where "+ tipo + "
like '"+parametro+"%';");
cont=0;
while (rs.next()) {
```

```

        telefono[cont] = rs.getString("telefono");
        //System.out.println(telefono[cont] + "\n");
        cont=cont+1;
    }

    rs = s.executeQuery("SELECT codigo FROM clientes where "+ tipo + "
like '"+parametro+"%'");
    cont=0;
    while (rs.next()) {
        codigo[cont] = rs.getString("codigo");
        //System.out.println(codigo[cont] + "\n");
        cont=cont+1;
    }
    System.out.println("conexion cerrada");
} catch (SQLException ex) {
    JOptionPane.showMessageDialog(null, "No existe en base de datos",
>Error", JOptionPane.ERROR_MESSAGE);
    System.out.println("error2");
}
Object clientes[][] = new Object[cont][6];//declaración de matriz
for(int i=0;i<cont;i++){
    clientes[i][0] = codigo[i];
    clientes[i][1] = cedula[i];
    clientes[i][2] = apellido[i]+" "+nombre[i];//e una celda se guarda Nombre
y Apellido
    clientes[i][3] = correo[i];
    clientes[i][4] = telefono[i];
    clientes[i][5] = direccion[i];
}
return clientes;
}

```

```

//Tiene la misma función que el método anterior
//en esta vez se cambia la estructura de la matriz que retorna este método
//porque se trata de actualizar jTextFields y no jTableles
public Object[][] imprime_clientes_registrar(String tipo, String parametro){
    String[] cedula= new String[150];
    String[] nombre=new String[150];
    String[] codigo=new String[150];
    String[] apellido=new String[150];
    String[] correo=new String[150];
    String[] direccion=new String[150];
    String[] telefono=new String[150];
    int cont=0;

    try {

        ResultSet rs = s.executeQuery("SELECT cedula FROM clientes where
"+ tipo + " = '"+parametro+"");
        while (rs.next()) {
            cedula[cont] = rs.getString("cedula");
            //System.out.println(cedula[cont] + "\n");
            cont=cont+1;
        }

        rs = s.executeQuery("SELECT nombres FROM clientes where "+ tipo +
" = '"+parametro+"");
        cont=0;
        while (rs.next()) {
            nombre[cont] = rs.getString("nombres");
            //System.out.println(nombre[cont] + "\n");
            cont=cont+1;
        }
    }
}

```

```
}

rs = s.executeQuery("SELECT apellido FROM clientes where "+ tipo + "
= "+parametro+"";");
cont=0;
while (rs.next()) {
    apellido[cont] = rs.getString("apellido");
    //System.out.println(apellido[cont] + "\n");
    cont=cont+1;
}

rs = s.executeQuery("SELECT correo FROM clientes where "+ tipo + "
"+parametro+"";");
cont=0;
while (rs.next()) {
    correo[cont] = rs.getString("correo");
    //System.out.println(correo[cont] + "\n");
    cont=cont+1;
}

rs = s.executeQuery("SELECT direccion FROM clientes where "+ tipo +
" = "+parametro+"";");
cont=0;
while (rs.next()) {
    direccion[cont] = rs.getString("direccion");
    //System.out.println(direccion[cont] + "\n");
    cont=cont+1;
}

rs = s.executeQuery("SELECT telefono FROM clientes where "+ tipo + "
= "+parametro+"";");
```



```

cont=0;
while (rs.next()) {
    telefono[cont] = rs.getString("telefono");
    //System.out.println(telefono[cont] + "\n");
    cont=cont+1;
}

rs = s.executeQuery("SELECT codigo FROM clientes where "+ tipo + "
= "+parametro+";");
cont=0;
while (rs.next()) {
    codigo[cont] = rs.getString("codigo");
    //System.out.println(codigo[cont] + "\n");
    cont=cont+1;
}
System.out.println("conexion cerrada");
} catch (SQLException ex) {
    JOptionPane.showMessageDialog(null, "No existe en base de datos",
"Error", JOptionPane.ERROR_MESSAGE);
    System.out.println("error2");
}
Object clientes[][] = new Object[cont][7];
for(int i=0;i<cont;i++){
    clientes[i][0] = codigo[i];
    clientes[i][1] = cedula[i];
    clientes[i][2] = apellido[i];
    clientes[i][3] = nombre[i];
    clientes[i][4] = correo[i];
    clientes[i][5] = telefono[i];
    clientes[i][6] = direccion[i];
}

```

```
return clientes;
```

```
}
```

```
//*****
```

```
***
```

```
//*****
```

```
//FIN DE COMPORTAMIENTO DE CLIENTES EN BASE DE DATOS
```

```
//*****
```

```
***
```

```
//*****
```

```
//INICIO DE COMPORTAMIENTO DE FACTURAS EN BASE DE DATOS
```

```
//Ordena la tabla de FACTURAS según el número y devuelve el último,
```

```
//se utiliza para que no existan facturas con igual número
```

```
public int obtener_ultima_factura(){
```

```
    String num=null;
```

```
    try {
```

```
        ResultSet rs = s.executeQuery("SELECT num FROM factura ORDER  
BY num DESC LIMIT 1;");
```

```
        while (rs.next()) {
```

```
            num = rs.getString("num");
```

```
        }
```

```

        } catch (SQLException ex) {
            Logger.getLogger(base_datos.class.getName()).log(Level.SEVERE,
null, ex);
        }
        int nume = Integer.parseInt(num);
        return nume;
    }

    //INSERTA UN NUEVO REGISTRO DE COMPRA AL ENVIAL EL MAIL AL
    CLIENTE
    public void insertar_proxima_factura(int numero){
        try {
            s.executeUpdate("insert into factura(num) values("+numero+"");
        } catch (Exception ex) {
            System.out.println("error metodo insertar...");
        }
    }
}

```

Controlador base_datos_Users.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package CONTROLADOR;

import java.sql.*;
import java.util.logging.Level;
import java.util.logging.Logger;

```

```
import javax.swing.JOptionPane;
/**
 *
 * @author Paúl Molina
 */
public class base_datos_Users {
    public Statement s ;

    public void inicio() {
        try {
            String url="jdbc:mysql://localhost/tesis_1";
            String usuario="root";
            String clave ="root";

            DriverManager.registerDriver(new org.gjt.mm.mysql.Driver());
            System.out.println("driver Mysql cargando...");
            Connection conexion =
DriverManager.getConnection(url,usuario,clave);
            System.out.println("conexion abierta con "+url+"...");
            s= conexion.createStatement();
            System.out.println("sentencia creada...\n");

        } catch (SQLException ex) {
            System.out.println("error1");

        }
    }
}
```

```
public void insertar_usuario(String cedula, String nombre, String apellidos,
String clave, String tipo, String nick){

    try {
        s.executeUpdate("insert into
usuarios(cedula,nombre,apellidos,clave,tipo,nick)"
            + "values("
            + ""+cedula+","
            + ""+nombre+","
            + ""+apellidos+","
            + ""+clave+","
            + ""+tipo+","
            + ""+nick+");");
    } catch (Exception ex) {
        System.out.println("error metodo insertar...");
    }

}
```

```
public String encuentra_usuario(String usuario){
    String usuario_encontrado = null;
    try {
        ResultSet rs = s.executeQuery("SELECT nick FROM usuarios where
nick = "+usuario+");");
        while (rs.next()) {
            usuario_encontrado = rs.getString("nick");
        }
    }
}
```

```

        //System.out.println(cedula[cont] + "\n");
    }
    System.out.println("conexion cerrada");
} catch (SQLException ex) {
    JOptionPane.showMessageDialog(null, "No existe en base de datos",
>Error", JOptionPane.ERROR_MESSAGE);
    System.out.println("error2");
}

return usuario_encontrado;
}

```

```

public String encuentra_nombre(String nick){
    String usuario_encontrado = null;
    try {
        ResultSet rs = s.executeQuery("SELECT nombre FROM usuarios
where nick = '"+nick+"'");
        while (rs.next()) {
            usuario_encontrado = rs.getString("nombre");
            //System.out.println(cedula[cont] + "\n");
        }
        System.out.println("conexion cerrada");
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(null, "No existe en base de datos",
>Error", JOptionPane.ERROR_MESSAGE);
        System.out.println("error2");
    }
    return usuario_encontrado;
}

```

```

public String encuentra_apellido(String nick){
    String usuario_encontrado = null;
    try {
        ResultSet rs = s.executeQuery("SELECT apellidos FROM usuarios
where nick = '"+nick+"';");
        while (rs.next()) {
            usuario_encontrado = rs.getString("apellidos");
            //System.out.println(cedula[cont] + "\n");
        }
        System.out.println("conexion cerrada");
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(null, "No existe en base de datos",
"Error", JOptionPane.ERROR_MESSAGE);
        System.out.println("error2");
    }
    return usuario_encontrado;
}

```

```

public String verifica_clave(String usuario, String clave){
    String clave_verificada = null;
    try {

        ResultSet rs = s.executeQuery("SELECT clave FROM usuarios
where clave = '"+clave+"';");
        while (rs.next()) {
            clave_verificada = rs.getString("clave");
            //System.out.println(cedula[cont] + "\n");
        }
        System.out.println("conexion cerrada");
    } catch (SQLException ex) {

```

```
        JOptionPane.showMessageDialog(null, "No existe en base de datos",  
"Error", JOptionPane.ERROR_MESSAGE);
```

```
        System.out.println("error2");
```

```
    }
```

```
    return clave_verificada;
```

```
}
```

```
public String tipo_usuario(String usuario){
```

```
    String tipo = null;
```

```
    try {
```

```
        ResultSet rs = s.executeQuery("SELECT tipo FROM usuarios where  
nick = '"+usuario+"'");
```

```
        while (rs.next()) {
```

```
            tipo = rs.getString("tipo");
```

```
            //System.out.println(cedula[cont] + "\n");
```

```
        }
```

```
        System.out.println("conexion cerrada");
```

```
    } catch (SQLException ex) {
```

```
        System.out.println("error2");
```

```
    }
```

```
    return tipo;
```

```
}
```

```
public void elimina_usuario(String cedula){
```

```
    int filas_mod=0;
```

```
    try {
```



```

        ResultSet rs = s.executeQuery("SELECT cedula FROM usuarios
WHERE cedula = '"+cedula+"'");
        while (rs.next()){
            filas_mod=filas_mod+1;
        }
        s.executeUpdate("DELETE FROM usuarios WHERE cedula =
 '"+cedula+"'");
        System.out.println("conexion cerrada_EXITO");
        System.out.println(filas_mod);
        if(filas_mod>0){
            JOptionPane.showMessageDialog(null,"Usuario
ELIMINADO","ÉXITO",JOptionPane.INFORMATION_MESSAGE);
        }else{
            JOptionPane.showMessageDialog(null,"No en base de
datos\nREVISE CÉDULA","ERROR",JOptionPane.ERROR_MESSAGE);
        }
    } catch (SQLException ex) {
        System.out.println("error2");

JOptionPane.showMessageDialog(null,"ERROR","ERROR",JOptionPane.ERR
OR_MESSAGE);
    }
}

```

```

public Object[][] imprime_usuarios(String tipo, String parametro){
    String[] nombre=new String[150];
    String[] apellido=new String[150];
    String[] clave=new String[150];
    String[] tipo_U=new String[15];
    String[] nick=new String[150];
    int cont=0;

```

```
try {  
  
    ResultSet rs = s.executeQuery("SELECT nombre FROM usuarios  
where "+ tipo + " = '"+parametro+"");  
    cont=0;  
    while (rs.next()) {  
        nombre[cont] = rs.getString("nombre");  
        //System.out.println(nombre[cont] + "\n");  
        cont=cont+1;  
    }  
  
    rs = s.executeQuery("SELECT apellidos FROM usuarios where "+  
tipo + " = '"+parametro+"");  
    cont=0;  
    while (rs.next()) {  
        apellido[cont] = rs.getString("apellidos");  
        //System.out.println(apellido[cont] + "\n");  
        cont=cont+1;  
    }  
  
    rs = s.executeQuery("SELECT clave FROM usuarios where "+ tipo +  
" = '"+parametro+"");  
    cont=0;  
    while (rs.next()) {  
        clave[cont] = rs.getString("clave");  
        //System.out.println(correo[cont] + "\n");  
        cont=cont+1;  
    }  
}
```

```

        rs = s.executeQuery("SELECT tipo FROM usuarios where "+ tipo + "
= "+parametro+"");
        cont=0;
        while (rs.next()) {
            tipo_U[cont] = rs.getString("tipo");
            //System.out.println(direccion[cont] + "\n");
            cont=cont+1;
        }

```

```

        rs = s.executeQuery("SELECT nick FROM usuarios where "+ tipo + "
= "+parametro+"");
        cont=0;
        while (rs.next()) {
            nick[cont] = rs.getString("nick");
            //System.out.println(direccion[cont] + "\n");
            cont=cont+1;
        }

```

```

        System.out.println("conexion cerrada");
    } catch (SQLException ex) {
        JOptionPane.showMessageDialog(null, "No existe en base de datos",
"Error", JOptionPane.ERROR_MESSAGE);
        System.out.println("error2");
    }
    Object usuarios[][] = new Object[cont][5];
    for(int i=0;i<cont;i++){
        usuarios[i][0] = apellido[i];
        usuarios[i][1] = nombre[i];
        usuarios[i][2] = clave[i];
        usuarios[i][3] = tipo_U[i];

```

```
        usuarios[i][4] = nick[i];
    }
    return usuarios;

}

public void actualizar_usuarios(String cedula, String nombres, String
apellido, String clave, String tipo, String nick){
    try {
        s.executeUpdate("Update usuarios set nombre = " +nombres+" where
cedula = "+cedula+";");
        s.executeUpdate("Update usuarios set apellidos = " +apellido+" where
cedula = "+cedula+";");
        s.executeUpdate("Update usuarios set clave = " +clave+" where cedula
= "+cedula+";");
        s.executeUpdate("Update usuarios set tipo = " +tipo+" where cedula =
"+cedula+";");
        s.executeUpdate("Update usuarios set nick = " +nick+" where cedula =
"+cedula+";");
        s.executeUpdate("Update usuarios set cedula = " +cedula+" where
cedula = "+cedula+";");

    } catch (Exception ex) {
        System.out.println("error metodo insertar...");
    }

}

}
```

Algoritmos de las interfaces

Interfaces Acceso.java

```
*  
* Acceso.java  
* Creada el 12 de noviembre de 2013  
*/  
  
package INTERFACES;  
  
import CONTROLADOR.ComSerial;  
import CONTROLADOR.base_datos_Users;  
import javax.swing.JOptionPane;  
  
/**  
*  
* @author Paul Molina  
*/  
  
public class Acceso extends javax.swing.JFrame {  
  
    /** Creates new form Acceso */  
    public Acceso() {  
        initComponents();  
        this.setLocationRelativeTo(null);//centrar frame en pantalla  
        txtUsuario.setText("");  
        txtcontraseña.setText("");//inicializo jTextField  
  
    }
```

```

@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel2 = new javax.swing.JPanel();
    jPanel1 = new javax.swing.JPanel();
    lblaccesosis = new javax.swing.JLabel();
    jSeparator1 = new javax.swing.JSeparator();
    lblusuario = new javax.swing.JLabel();
    txtUsuario = new javax.swing.JTextField();
    lblcontraseña = new javax.swing.JLabel();
    txtcontraseña = new javax.swing.JPasswordField();
    btnAceptar = new javax.swing.JButton();
    btnSalir = new javax.swing.JButton();
    jLabel1 = new javax.swing.JLabel();

    javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
    jPanel2.setLayout(jPanel2Layout);
    jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGap(0, 100, Short.MAX_VALUE)
);
    jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGap(0, 100, Short.MAX_VALUE)

```

```
);
```

```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);  
setTitle("Acceso");  
setBackground(new java.awt.Color(151, 0, 45));
```

```
jPanel1.setBackground(new java.awt.Color(151, 0, 45));
```

```
lblaccesosis.setBackground(new java.awt.Color(0, 0, 0));  
lblaccesosis.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); //
```

```
NOI18N
```

```
lblaccesosis.setForeground(new java.awt.Color(255, 255, 255));  
lblaccesosis.setText("ACCESO AL SISTEMA");
```

```
jSeparator1.setForeground(new java.awt.Color(255, 255, 255));
```

```
lblusuario.setFont(new java.awt.Font("Swis721 Ex BT", 0, 20)); // NOI18N  
lblusuario.setForeground(new java.awt.Color(255, 255, 255));  
lblusuario.setText("Usuario:");
```

```
txtUsuario.setFont(new java.awt.Font("Swis721 Ex BT", 0, 16)); // NOI18N
```

```
lblcontraseña.setFont(new java.awt.Font("Swis721 Ex BT", 0, 20)); //
```

```
NOI18N
```

```
lblcontraseña.setForeground(new java.awt.Color(255, 255, 255));  
lblcontraseña.setText("Contraseña:");
```

```
txtcontraseña.setFont(new java.awt.Font("Swis721 Ex BT", 0, 16)); //
```

```
NOI18N
```

```

btnAceptar.setBackground(new java.awt.Color(255, 255, 255));
btnAceptar.setFont(new java.awt.Font("Swis721 Ex BT", 0, 14)); // NOI18N
btnAceptar.setText("ACEPTAR");
btnAceptar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnAceptarActionPerformed(evt);
    }
});

```

```

btnSalir.setBackground(new java.awt.Color(255, 255, 255));
btnSalir.setFont(new java.awt.Font("Swis721 Ex BT", 0, 14)); // NOI18N
btnSalir.setText("SALIR");
btnSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnSalirActionPerformed(evt);
    }
});

```

```

jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/logo_udla_peq.jp
g"))); // NOI18N

```

```

javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(96, 96, 96)

```



```
.addComponent(btnAceptar)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(btnSalir)
    .addGap(93, 93, 93))
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup())
    .addContainerGap(35, Short.MAX_VALUE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup())

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.TRAILING)
    .addComponent(lblcontraseña)
    .addComponent(lblusuario))
    .addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.TRAILING, false)
    .addComponent(txtUsuario,
javax.swing.GroupLayout.DEFAULT_SIZE, 273, Short.MAX_VALUE)
    .addComponent(txtcontraseña))
    .addGap(33, 33, 33))
.addGroup(jPanel1Layout.createSequentialGroup())
    .addComponent(jLabel1)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
63, Short.MAX_VALUE)
```

```
    .addComponent(lblaccessosis)
```

```
    .addGap(24, 24, 24)))
```

```
.addGroup(jPanel1Layout.createSequentialGroup()
```

```
    .addContainerGap()
```

```
    .addComponent(jSeparator1)
```

```
    .addContainerGap())
```

```
);
```

```
jPanel1Layout.setVerticalGroup(
```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
NG)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup()
```

```
        .addContainerGap()
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
```

```
    .addComponent(jLabel1,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE, 52,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(lblaccessosis,
```

```
        javax.swing.GroupLayout.DEFAULT_SIZE,
```

```
        javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
D)
```

```
    .addComponent(jSeparator1,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE, 12,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
    .addComponent(txtUsuario,
javax.swing.GroupLayout.PREFERRED_SIZE, 32,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(lblusuario,
javax.swing.GroupLayout.PREFERRED_SIZE, 29,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(35, 35, 35)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
    .addComponent(txtcontraseña,
javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(lblcontraseña,
javax.swing.GroupLayout.PREFERRED_SIZE, 35,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
34, Short.MAX_VALUE)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
    .addComponent(btnAceptar)
    .addComponent(btnSalir)
    .addGap(25, 25, 25))
);
```

```

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );

        pack();
    } // </editor-fold>

    private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
        System.exit(0);
    }

    private void btnAceptarActionPerformed(java.awt.event.ActionEvent evt) {

        base_datos_Users bdu = new base_datos_Users();//declarar bjeto del
gestor de base de dats de usuario
        bdu.inicio();//iniciaiza conexion con base de datos
        String nombre= bdu.encuentra_usuario(txtUsuario.getText());//devuelve
null si no hay coincidencia

```

```

String clave=
bdu.verifica_clave(txtUsuario.getText(),txtcontraseña.getText());//devuelve null
si no hay coincidencia
    System.out.println("+++++++++++++++++++++++++clave"+nombre+"
"+clave);
    if (nombre!=null && clave!=null){
        this.setVisible(false);
        //MenuPrincipal menu=new MenuPrincipal();
        //menu.setVisible(true);
        String tipo_usuario= bdu.tipo_usuario(txtUsuario.getText());//obtiene tipo
de usuario
        String Vendedor=bdu.encuentra_nombre(txtUsuario.getText()+
"+bdu.encuentra_apellido(txtUsuario.getText());//verifica clave

        //menu.bandera_tipo_usuario(tipo_usuario);
        if("A".equals(tipo_usuario)){//despliega ventana para administrador
            Administrador admin = new Administrador();
            admin.setVisible(true);
            admin.setear(tipo_usuario, Vendedor);
        }
        if("U".equals(tipo_usuario)){//despliega ventana para usuario
            Usuario user = new Usuario();
            user.setVisible(true);
            user.setear(tipo_usuario, Vendedor);
        }

    }else{
        JOptionPane.showMessageDialog(null, "No existe en base de datos",
"Error", JOptionPane.ERROR_MESSAGE);
        txtUsuario.setText("");
        txtcontraseña.setText("");
    }

```

```

    }

}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Acceso.class.getName()).log(java.util.logging
.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

```

```
java.util.logging.Logger.getLogger(Acceso.class.getName()).log(java.util.logging
.Level.SEVERE, null, ex);
```

```
    } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(Acceso.class.getName()).log(java.util.logging
.Level.SEVERE, null, ex);
```

```
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(Acceso.class.getName()).log(java.util.logging
.Level.SEVERE, null, ex);
```

```
    }
```

```
//</editor-fold>
```

```
/* Create and display the form */
```

```
java.awt.EventQueue.invokeLater(new Runnable() {
```

```
    public void run() {
```

```
        new Acceso().setVisible(true);
```

```
    }
```

```
});
```

```
}
```

```
// Variables declaration - do not modify
```

```
private javax.swing.JButton btnAceptar;
```

```
private javax.swing.JButton btnSalir;
```

```
private javax.swing.JLabel jLabel1;
```

```
private javax.swing.JPanel jPanel1;
```

```
private javax.swing.JPanel jPanel2;
```

```
private javax.swing.JSeparator jSeparator1;
```

```
private javax.swing.JLabel lblaccesosis;
```

```

private javax.swing.JLabel lblcontraseña;
private javax.swing.JLabel lblusuario;
private javax.swing.JTextField txtUsuario;
private javax.swing.JPasswordField txtcontraseña;
// End of variables declaration
}

```

Interface Administrador.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * Administrador.java
 *
 * Creada el 12 de noviembre de 2013
 */
package INTERFACES;

/**
 *
 * @author Paul Molina
 */

public class Administrador extends javax.swing.JFrame {
    MenuPrincipal menu=new MenuPrincipal();
    /** Creates new form Administrador */
    public Administrador() {
        initComponents();
    }

```



```
        this.setLocationRelativeTo(null);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        lblsam = new javax.swing.JLabel();
        jSeparator4 = new javax.swing.JSeparator();
        btnMenu = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        lblBienvenido = new javax.swing.JLabel();
        lblsam2 = new javax.swing.JLabel();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jPanel1.setBackground(new java.awt.Color(151, 0, 45));
        jPanel1.setForeground(new java.awt.Color(255, 255, 255));

        lblsam.setFont(new java.awt.Font("Swis721 Ex BT", 0, 36)); // NOI18N
        lblsam.setForeground(new java.awt.Color(255, 255, 255));
        lblsam.setText("Administrador");

        btnMenu.setBackground(new java.awt.Color(255, 255, 255));
```

```

        btnMenu.setFont(new java.awt.Font("Swis721 Ex BT", 0, 18)); // NOI18N
        btnMenu.setForeground(new java.awt.Color(151, 0, 45));
        btnMenu.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/menu_principal.jpg")
)); // NOI18N
        btnMenu.setText("MENÚ");
        btnMenu.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnMenuActionPerformed(evt);
            }
        });

        jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/logo_udla.jpg")));
// NOI18N
        jLabel1.setText("jLabel1");

        lblBienvenido.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); //
NOI18N
        lblBienvenido.setForeground(new java.awt.Color(255, 255, 255));
        lblBienvenido.setText("Bienvenido...");

        lblsam2.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); // NOI18N
        lblsam2.setForeground(new java.awt.Color(255, 255, 255));
        lblsam2.setText("Tipo de Acceso: Total");

        javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(

```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
```

```
jPanel1Layout.createSequentialGroup())
```

```
    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
```

```
    .addComponent(btnMenu)
```

```
    .addGap(29, 29, 29))
```

```
    .addGroup(jPanel1Layout.createSequentialGroup())
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup())
```

```
        .addGap(172, 172, 172)
```

```
        .addComponent(lblsam2))
```

```
    .addGroup(jPanel1Layout.createSequentialGroup())
```

```
        .addGap(26, 26, 26)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(lblBienvenido,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 530,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup())
```

```
        .addComponent(jLabel1,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 234,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
        .addGap(55, 55, 55)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(lblsam)
```

```
    .addComponent(jSeparator4,
javax.swing.GroupLayout.PREFERRED_SIZE, 269,
javax.swing.GroupLayout.PREFERRED_SIZE))))))
```

```
    .addContainerGap(47, Short.MAX_VALUE))
```

```
);
```

```
jPanel1Layout.setVerticalGroup(
```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup())
```

```
    .addContainerGap()
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup())
```

```
    .addGap(29, 29, 29)
```

```
    .addComponent(lblsam,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 44,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
    .addComponent(jSeparator4,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
    .addComponent(jLabel1,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 136,
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(lblBienvenido,
javax.swing.GroupLayout.PREFERRED_SIZE, 44,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(11, 11, 11)
    .addComponent(lblsam2,
javax.swing.GroupLayout.PREFERRED_SIZE, 44,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(btnMenu)
    .addContainerGap(23, Short.MAX_VALUE)
);

```

```

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

```

```

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);
    layout.setVerticalGroup(

```

```

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,

```

```

javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(0, 0, Short.MAX_VALUE))
    );

```

```

    pack();
} // </editor-fold>

```

```

private void btnMenuActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    menu.setVisible(true);
    this.setVisible(false);
}

```

```

public void setear(String bandera_tipo, String vendedor){
    menu.bandera_tipo_usuario(bandera_tipo);//configura interfaz para uso de
administrador

```

```

    Factura.SetearVendedor(vendedor);//imprime vendedor en la factura
    lblBienvenido.setText("Bienvenido "+vendedor);
}

```

```

/**

```

```

 * @param args the command line arguments

```

```

 */

```

```

public static void main(String args[]) {

```

```

    /* Set the Nimbus look and feel */

```

```

    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">

```

```

    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.

```

```

    * For details see

```

```

http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

```

```

    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Administrador.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Administrador.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Administrador.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Administrador.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
    }
}
//</editor-fold>

/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {

```

```

        public void run() {
            new Administrador().setVisible(true);
        }
    });
}
// Variables declaration - do not modify
private javax.swing.JButton btnMenu;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
private javax.swing.JSeparator jSeparator4;
private javax.swing.JLabel lblBienvenido;
private javax.swing.JLabel lblsam;
private javax.swing.JLabel lblsam2;
// End of variables declaration
}

```

Interface BusquedaCliente.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package INTERFACES;

import CONTROLADOR.base_datos;
import java.awt.Dimension;
import java.awt.Toolkit;
import java.awt.event.KeyEvent;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

```



```

public class BusquedaCliente extends javax.swing.JPanel {
DefaultTableModel modelo_tabla;
    /**
     * Creates new form PanelVistaCliente
     */
    base_datos base_de_datos = new base_datos();

    public BusquedaCliente() {
        initComponents();
        //Tomo el tamaño de la pantalla
        Dimension pantallaTamano = Toolkit.getDefaultToolkit().getScreenSize();
        //Al ancho de la pantalla lo divido en 2 y le resto la mitad del ancho de mi
        ventana, con eso queda centrada en el eje X, para el eje Y es lo mismo pero
        con el alto:
        this.setLocation((pantallaTamano.width/2)-(this.getWidth()/2),
        (pantallaTamano.height/2)-(this.getHeight()/2));
        base_de_datos.inicio();//inicializa conexon con base de datos
        bgGrupo.add(rbCedula);
        bgGrupo.add(rbApellido);
        bgGrupo.add(rbNombre);//para poder elegir solo una opción

        String
        cabecera[]={ "CÓDIGO", "CÉDULA", "NOMBRE", "CORREO", "TELÉFONO", "DIR
        ECCIÓN"};
        String datos[][]={}; //matrices que conforman tabla de búsqueda
        modelo_tabla = new DefaultTableModel(datos,cabecera){
            @Override
            public Class<?> getColumnClass(int column) {
                return getValueAt(0, column).getClass();
            }
        }
    }
}

```

```

@Override
public boolean isCellEditable(int row, int column) {
    return false;
}
};
jtClientes.setModel(modelo_tabla);
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {
    java.awt.GridBagConstraints gridBagConstraints;

    bgGrupo = new javax.swing.ButtonGroup();
    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    txtDatoCliente = new javax.swing.JTextField();
    rbCedula = new javax.swing.JRadioButton();
    rbApellido = new javax.swing.JRadioButton();
    rbNombre = new javax.swing.JRadioButton();
    btnElegir = new javax.swing.JButton();
    jPanel2 = new javax.swing.JPanel();
    jScrollPane1 = new javax.swing.JScrollPane();
    jtClientes = new javax.swing.JTable();

```

```
setLayout(new java.awt.BorderLayout());

jPanel1.setBackground(new java.awt.Color(153, 0, 51));
jPanel1.setBorder(javax.swing.BorderFactory.createEtchedBorder());
jPanel1.setLayout(new java.awt.GridBagLayout());

jLabel1.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
jLabel1.setText("Buscar Cliente:");
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.ipady = 4;
gridBagConstraints.insets = new java.awt.Insets(2, 0, 1, 0);
jPanel1.add(jLabel1, gridBagConstraints);

txtDatoCliente.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
txtDatoCliente.setPreferredSize(new java.awt.Dimension(250, 21));
txtDatoCliente.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyPressed(java.awt.event.KeyEvent evt) {
        txtDatoClienteKeyPressed(evt);
    }
});
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.gridwidth = 3;
gridBagConstraints.ipady = 4;
gridBagConstraints.insets = new java.awt.Insets(2, 0, 1, 0);
jPanel1.add(txtDatoCliente, gridBagConstraints);

rbCedula.setBackground(new java.awt.Color(153, 0, 51));
rbCedula.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
rbCedula.setSelected(true);
rbCedula.setText("Por Cédula");
gridBagConstraints = new java.awt.GridBagConstraints();
```

```
gridBagConstraints.gridx = 1;
gridBagConstraints.gridy = 1;
gridBagConstraints.ipady = 4;
gridBagConstraints.insets = new java.awt.Insets(2, 0, 1, 0);
jPanel1.add(rbCedula, gridBagConstraints);

rbApellido.setBackground(new java.awt.Color(153, 0, 51));
rbApellido.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
rbApellido.setText("Por Apellido");
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.gridx = 2;
gridBagConstraints.gridy = 1;
gridBagConstraints.ipady = 4;
gridBagConstraints.insets = new java.awt.Insets(2, 0, 1, 0);
jPanel1.add(rbApellido, gridBagConstraints);

rbNombre.setBackground(new java.awt.Color(153, 0, 51));
rbNombre.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
rbNombre.setText("Por Nombre");
gridBagConstraints = new java.awt.GridBagConstraints();
gridBagConstraints.gridx = 3;
gridBagConstraints.gridy = 1;
gridBagConstraints.ipady = 4;
gridBagConstraints.insets = new java.awt.Insets(2, 0, 1, 0);
jPanel1.add(rbNombre, gridBagConstraints);

btnElegir.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
btnElegir.setText("ELEGIR");
btnElegir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnElegirActionPerformed(evt);
    }
});
```

```
    }  
});  
gridBagConstraints = new java.awt.GridBagConstraints();  
gridBagConstraints.gridx = 0;  
gridBagConstraints.gridy = 1;  
jPanel1.add(btnElegir, gridBagConstraints);  
  
add(jPanel1, java.awt.BorderLayout.PAGE_START);  
  
jPanel2.setBackground(new java.awt.Color(153, 0, 51));  
jPanel2.setLayout(new java.awt.BorderLayout());  
  
jScrollPane1.setBackground(new java.awt.Color(153, 0, 51));  
  
jScrollPane1.setHorizontalScrollBarPolicy(javax.swing.ScrollPaneConstants.HO  
RIZONTAL_SCROLLBAR_ALWAYS);  
  
jScrollPane1.setVerticalScrollBarPolicy(javax.swing.ScrollPaneConstants.VERT  
ICAL_SCROLLBAR_ALWAYS);  
jScrollPane1.setPreferredSize(new java.awt.Dimension(452, 120));  
  
jtClientes.setBackground(new java.awt.Color(153, 0, 51));  
jtClientes.setModel(new javax.swing.table.DefaultTableModel(  
    new Object [][] {  
        {null, null, null, null},  
        {null, null, null, null},  
        {null, null, null, null},  
        {null, null, null, null}  
    },  
    new String [] {  
        "Título 1", "Título 2", "Título 3", "Título 4"
```

```

    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});
jScrollPane1.setViewportView(jtClientes);

jPanel2.add(jScrollPane1, java.awt.BorderLayout.CENTER);

add(jPanel2, java.awt.BorderLayout.CENTER);
} // </editor-fold>

//el boton elegir actualiza la interfaz factura segun la fila seleccionada en la
tabla de búsqueda
private void btnElegirActionPerformed(java.awt.event.ActionEvent evt) {
    int i =jtClientes.getSelectedRow();
    if(jtClientes.getRowCount()>0 && i>=0){
        String code=(String)jtClientes.getValueAt(i,0);
        String cedula=(String)jtClientes.getValueAt(i,1);
        String cliente=(String)jtClientes.getValueAt(i,2);
        String correo=(String)jtClientes.getValueAt(i,3);
        String telefono=(String)jtClientes.getValueAt(i,4);
        String direccion=(String)jtClientes.getValueAt(i,5);
        Factura.escribe_datos(cedula, cliente, correo, telefono, direccion);
        Factura.esconde_dlg();
    }
}

```

```

        Factura.btnPagar.setEnabled(true);
    }else{
        JOptionPane.showMessageDialog(null, "NO HAY COINCIDENCIAS",
"NO COINCIDENCIAS",JOptionPane.ERROR_MESSAGE);
    }
}

```

```

private void txtDatoClienteKeyPressed(java.awt.event.KeyEvent evt) {
    if (evt.getKeyCode() == KeyEvent.VK_ENTER) {
        this.busqueda_cliente();//envia texto del campo de texto txtDatoCliente
        hacia un metodo de búsqueda en la clase Base de datos
    }
    this.busqueda_cliente();
}

```

//metodo que se comunica con el metodo buscador llamado imprimirClientes en la base de datos

```

public void busqueda_cliente(){
    String tipo=null;
    if(rbCedula.isSelected()){
        tipo="cedula";
    }else if(rbNombre.isSelected()){
        tipo="nombres";
    }else if(rbApellido.isSelected()){
        tipo="apellido";
    }

    this.limpiarTabla();
}

```

```

    Object cliente[][] =
base_de_datos.imprime_clientes(tipo,txtDatoCliente.getText());//imprime_client
es es un metodo en la clase base de datos que de
        //devuelve una matriz con todas las coincidencias de
búsqueda
    for(int i=0;i<cliente.length;i++){
        modelo_tabla.addRow(cliente[i]);//se graga una fila por cada
coincidencia que devuelve el metodo imprime_clientes
    }

}

public void limpiarTabla(){
    int filas=jtClientes.getRowCount();
    for (int i = 0;filas>i; i++) {
        modelo_tabla.removeRow(0);
    }

}

// Variables declaration - do not modify
private javax.swing.ButtonGroup bgGrupo;
private javax.swing.JButton btnElegir;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;

```



```
private javax.swing.JPanel jPanel2;  
private javax.swing.JScrollPane jScrollPane1;  
private javax.swing.JTable jtClientes;  
private javax.swing.JRadioButton rbApellido;  
private javax.swing.JRadioButton rbCedula;  
private javax.swing.JRadioButton rbNombre;  
private javax.swing.JTextField txtDatoCliente;  
// End of variables declaration  
}
```

Interface Factura.java

```
/*  
 * To change this template, choose Tools | Templates  
 * and open the template in the editor.  
 */  
package INTERFACES;  
  
import CONTROLADOR.ComSerial;  
import CONTROLADOR.Mail;  
import CONTROLADOR.PdfCreador;  
import CONTROLADOR.base_datos;  
import java.awt.Color;  
import java.io.IOException;  
import java.util.Calendar;  
import java.util.Date;  
import java.util.GregorianCalendar;  
import java.util.logging.Level;  
import java.util.logging.Logger;  
import javax.swing.JDialog;  
import javax.swing.JOptionPane;
```

```

import javax.swing.table.DefaultTableModel;

/**
 *
 * @author Paúl Molina
 */
public class Factura extends javax.swing.JInternalFrame implements Runnable{
public static DefaultTableModel modelo_tabla;//tipo de dato para definir
estructura de tabla
Mail mail = new Mail();
PdfCreador pdf = new PdfCreador();
String horas, minutos, segundos, ampm, dias, meses, anios;
Calendar calendario;
Thread h1;

@Override
public void run() {

    Thread ct = Thread.currentThread();
    while (ct == h1) {
        calcula();
        lblhora.setText(horas + ":" + minutos + ":" + segundos + " " + ampm);
        lblFecha.setText(dias + "/" + meses + "/" + anios);

        try {
            Thread.sleep(1000);
        } catch (InterruptedException e) {
        }
    }
}

```

```

}

public void calcula() {
    Calendar calendario = new GregorianCalendar();
    Date fechaHoraActual = new Date();

    calendario.setTime(fechaHoraActual);
    ampm = calendario.get(Calendar.AM_PM) == Calendar.AM ? "AM" : "PM";
    if (ampm.equals("PM")) {
        int h = calendario.get(Calendar.HOUR_OF_DAY) - 12;
        horas = h > 9 ? "" + h : "0" + h;
    } else {
        horas = calendario.get(Calendar.HOUR_OF_DAY) > 9 ? "" +
calendario.get(Calendar.HOUR_OF_DAY) : "0" +
calendario.get(Calendar.HOUR_OF_DAY);
    }
    minutos = calendario.get(Calendar.MINUTE) > 9 ? "" +
calendario.get(Calendar.MINUTE) : "0" + calendario.get(Calendar.MINUTE);
    segundos = calendario.get(Calendar.SECOND) > 9 ? "" +
calendario.get(Calendar.SECOND) : "0" + calendario.get(Calendar.SECOND);

    dias = Integer.toString(calendario.get(Calendar.DAY_OF_MONTH));

    meses = Integer.toString(calendario.get(Calendar.MONTH) + 1);
    if (meses.length() == 1) {
        meses = "0" + meses;
    } else {
        meses = meses;
    }
    anios = Integer.toString(calendario.get(Calendar.YEAR));
}

```

```

/**
 * Creates new form Factura
 */
public Factura() {
    initComponents();
    String cabecera[]={"Código","Producto","P. Unitario
$, "Cantidad", "Sub_total $"}; //define caecera de tabla de productos en
venta//*****
    String datos[][]={}; //matriz donde se almacenarn los productos que se
detecten en el disp RF. ID al momento de la venta
    modelo_tabla = new DefaultTableModel(datos,cabecera) {
        @Override
        public Class<?> getColumnClass(int column) {
            return getValueAt(0, column).getClass();
        }

        @Override
        public boolean isCellEditable(int row, int column) {
            return false;
        }
    }; //se define la tabla en la interfaz gráfica
    jTableProductos.setModel(modelo_tabla); //se crea la tabla en el espacio
designado por el jTable llamado TaleProductos en el diseño gráfico
    ComSerial.bandera_agregar=0;
    ComSerial.bandera_factura=1;
    btnModificar.setEnabled(false);
    this.setTitle("SISTEMA DE FACTURACIÓN UNIVERSIDAD DE LAS
AMÉRICAS");
    btnMail.setEnabled(false);
    btnPagar.setEnabled(false);

```

```

    btnEliminar.setEnabled(false);
    Color fondo= new Color(153,0,51);
    this.setBackground(fondo);
    h1 = new Thread(this);
    h1.start();
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    pnlCabecera = new javax.swing.JPanel();
    lblNum = new javax.swing.JLabel();
    txtCode = new javax.swing.JTextField();
    lblFecha = new javax.swing.JLabel();
    lblhora = new javax.swing.JLabel();
    pnlClienteTabla = new javax.swing.JPanel();
    lblCliente = new javax.swing.JLabel();
    lblCorreo = new javax.swing.JLabel();
    txtNombreCliente = new javax.swing.JTextField();
    lblVendedor = new javax.swing.JLabel();
    lblTelefono = new javax.swing.JLabel();
    lblDireccion = new javax.swing.JLabel();
    txtCedula = new javax.swing.JTextField();
    txtCorreo = new javax.swing.JTextField();
    lblCedula = new javax.swing.JLabel();

```

```
txtDireccion = new javax.swing.JTextField();
txtTelefono = new javax.swing.JTextField();
txtVendedor = new javax.swing.JTextField();
btnClientes = new javax.swing.JButton();
pnlSTabla = new javax.swing.JScrollPane();
jTableProductos = new javax.swing.JTable();
pnlMail = new javax.swing.JPanel();
btnMail = new javax.swing.JButton();
jProgressBar = new javax.swing.JProgressBar();
pnlPago = new javax.swing.JPanel();
txtTotal = new javax.swing.JTextField();
jLabel3 = new javax.swing.JLabel();
jLabel2 = new javax.swing.JLabel();
txtSubtotal = new javax.swing.JTextField();
lblSubtotal = new javax.swing.JLabel();
txtIva = new javax.swing.JTextField();
pnlSistFac = new javax.swing.JPanel();
jLabel1 = new javax.swing.JLabel();
jLabel4 = new javax.swing.JLabel();
pnlBotones = new javax.swing.JPanel();
jPanel1 = new javax.swing.JPanel();
btnAgregar = new javax.swing.JButton();
jPanel2 = new javax.swing.JPanel();
btnCalculadora = new javax.swing.JButton();
jPanel3 = new javax.swing.JPanel();
btnModificar = new javax.swing.JButton();
jPanel4 = new javax.swing.JPanel();
btnEliminar = new javax.swing.JButton();
btnPagar = new javax.swing.JButton();

setBackground(new java.awt.Color(153, 0, 51));
```

```
setBorder(null);
setPreferredSize(new java.awt.Dimension(920, 650));

pnlCabecera.setBackground(new java.awt.Color(153, 0, 51));

lblNum.setFont(new java.awt.Font("TI-Nspire", 1, 18)); // NOI18N
lblNum.setForeground(new java.awt.Color(255, 255, 255));
lblNum.setText("Fac. Numero");

txtCode.setEditable(false);
txtCode.setBackground(new java.awt.Color(153, 0, 51));
txtCode.setFont(new java.awt.Font("TI-Nspire", 1, 14)); // NOI18N
txtCode.setForeground(new java.awt.Color(255, 255, 255));
txtCode.setText("0000");
txtCode.setBorder(null);

lblFecha.setBackground(new java.awt.Color(153, 0, 51));
lblFecha.setFont(new java.awt.Font("TI-Nspire", 1, 18)); // NOI18N
lblFecha.setForeground(new java.awt.Color(255, 255, 255));
lblFecha.setText("fecha");

lblhora.setBackground(new java.awt.Color(153, 0, 51));
lblhora.setFont(new java.awt.Font("TI-Nspire", 1, 18)); // NOI18N
lblhora.setForeground(new java.awt.Color(255, 255, 255));
lblhora.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);
lblhora.setText("hora");

javax.swing.GroupLayout pnlCabeceraLayout = new
javax.swing.GroupLayout(pnlCabecera);
pnlCabecera.setLayout(pnlCabeceraLayout);
pnlCabeceraLayout.setHorizontalGroup(
```

```
pnlCabeceraLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(pnlCabeceraLayout.createSequentialGroup())
```

```
        .addGap(4, 4, 4)
```

```
        .addComponent(lblNum,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 114,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
        .addComponent(txtCode,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 91,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
```

```
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```
        .addComponent(lblFecha,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
        .addComponent(lblhora,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE, 117,
```

```
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
        .addGap(149, 149, 149))
```

```
    );
```

```
    pnlCabeceraLayout.setVerticalGroup(
```

```
pnlCabeceraLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(pnlCabeceraLayout.createSequentialGroup())
```



```

.addGroup(pnlCabeceraLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING)
        .addComponent(lblNum,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(pnlCabeceraLayout.createSequentialGroup())
        .addGap(4, 4, 4)

.addGroup(pnlCabeceraLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.LEADING, false)
        .addGroup(pnlCabeceraLayout.createSequentialGroup())
        .addGap(7, 7, 7)

.addGroup(pnlCabeceraLayout.createParallelGroup(javax.swing.GroupLayout.A
lignment.BASELINE)
        .addComponent(lblFecha,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(lblhora,
javax.swing.GroupLayout.PREFERRED_SIZE, 21,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addComponent(txtCode,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))))
        .addGap(0, 0, Short.MAX_VALUE))
);

pnlClienteTabla.setBackground(new java.awt.Color(153, 0, 51));

lblCliente.setBackground(new java.awt.Color(153, 0, 51));

```

```
lblCliente.setFont(new java.awt.Font("TI-Nspire", 0, 18)); // NOI18N  
lblCliente.setForeground(new java.awt.Color(255, 255, 255));  
lblCliente.setText("Cliente");
```

```
lblCorreo.setBackground(new java.awt.Color(153, 0, 51));  
lblCorreo.setFont(new java.awt.Font("TI-Nspire", 0, 18)); // NOI18N  
lblCorreo.setForeground(new java.awt.Color(255, 255, 255));  
lblCorreo.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);  
lblCorreo.setText("Correo");
```

```
txtNombreCliente.setEditable(false);  
txtNombreCliente.setText("cliente");
```

```
lblVendedor.setBackground(new java.awt.Color(153, 0, 51));  
lblVendedor.setFont(new java.awt.Font("TI-Nspire", 0, 18)); // NOI18N  
lblVendedor.setForeground(new java.awt.Color(255, 255, 255));  
lblVendedor.setText("Vendedor");
```

```
lblTelefono.setFont(new java.awt.Font("TI-Nspire", 0, 18)); // NOI18N  
lblTelefono.setForeground(new java.awt.Color(255, 255, 255));  
lblTelefono.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);  
lblTelefono.setText("Teléfono");
```

```
lblDireccion.setFont(new java.awt.Font("TI-Nspire", 0, 18)); // NOI18N  
lblDireccion.setForeground(new java.awt.Color(255, 255, 255));  
lblDireccion.setHorizontalAlignment(javax.swing.SwingConstants.RIGHT);  
lblDireccion.setText("Direccion");
```

```
txtCedula.setEditable(false);  
txtCedula.setText("cédula");
```

```
txtCorreo.setEditable(false);  
txtCorreo.setText("correo");
```

```
lblCedula.setFont(new java.awt.Font("TI-Nspire", 0, 18)); // NOI18N  
lblCedula.setForeground(new java.awt.Color(255, 255, 255));  
lblCedula.setText("Cédula");
```

```
txtDireccion.setEditable(false);  
txtDireccion.setText("dirección");
```

```
txtTelefono.setEditable(false);  
txtTelefono.setText("teléfono");
```

```
txtVendedor.setEditable(false);  
txtVendedor.setText("vendedor");
```

```
btnClientes.setBackground(new java.awt.Color(255, 255, 153));  
btnClientes.setFont(new java.awt.Font("TI-Nspire", 0, 18)); // NOI18N  
btnClientes.setForeground(new java.awt.Color(153, 0, 51));  
btnClientes.setText("Clientes");
```

```
btnClientes.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.  
border.BevelBorder.RAISED));
```

```
    btnClientes.addActionListener(new java.awt.event.ActionListener() {  
        public void actionPerformed(java.awt.event.ActionEvent evt) {  
            btnClientesActionPerformed(evt);  
        }  
    });
```

```
pnlSTabla.setBackground(new java.awt.Color(153, 0, 51));
```

```

jTableProductos.setBackground(new java.awt.Color(153, 0, 51));
jTableProductos.setFont(new java.awt.Font("TI-Nspire", 1, 14)); // NOI18N
jTableProductos.setForeground(new java.awt.Color(255, 255, 255));
jTableProductos.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null, null},
        {null, null, null, null, null},
        {null, "", null, null, null},
        {null, null, null, null, null}
    },
    new String [] {
        "Código", "Producto", "Precio C/U", "Cantidad", "SubTotal"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});
jTableProductos.setCursor(new
java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
jTableProductos.setGridColor(new java.awt.Color(255, 255, 255));
jTableProductos.addMouseListener(new java.awt.event.MouseAdapter() {
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        jTableProductosMouseClicked(evt);
    }
});
pnlSTabla.setViewportView(jTableProductos);

```

```

        pnlMail.setBackground(new java.awt.Color(255, 255, 153));
        pnlMail.setBorder(javax.swing.BorderFactory.createTitledBorder(null,
"ENVIAR", javax.swing.border.TitledBorder.CENTER,
javax.swing.border.TitledBorder.TOP));

        btnMail.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/mail.jpg"))); //
NOI18N
        btnMail.setToolTipText("ENVIAR AL MAIL CUANDO SE HA ELEGIDO UN
CLIENTE");

btnMail.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.bo
rder.BevelBorder.RAISED));

        btnMail.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
        btnMail.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnMailActionPerformed(evt);
            }
        });

        jProgressBar.setBackground(new java.awt.Color(51, 0, 255));

        javax.swing.GroupLayout pnlMailLayout = new
javax.swing.GroupLayout(pnlMail);
        pnlMail.setLayout(pnlMailLayout);
        pnlMailLayout.setHorizontalGroup(

pnlMailLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(pnlMailLayout.createSequentialGroup()

```

```

        .addContainerGap()
        .addComponent(btnMail,
javax.swing.GroupLayout.PREFERRED_SIZE, 72,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
pnlMailLayout.createSequentialGroup())
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jProgressBar,
javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap())
    );
    pnlMailLayout.setVerticalGroup(

pnlMailLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(pnlMailLayout.createSequentialGroup())
        .addComponent(btnMail,
javax.swing.GroupLayout.PREFERRED_SIZE, 55,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jProgressBar,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap())
    );

```

```
pnlPago.setBackground(new java.awt.Color(153, 0, 51));
```

```
txtTotal.setBackground(new java.awt.Color(153, 0, 51));
```

```
txtTotal.setFont(new java.awt.Font("Tahoma", 1, 24)); // NOI18N
```

```
txtTotal.setForeground(new java.awt.Color(255, 255, 255));
```

```
txtTotal.setHorizontalAlignment(javax.swing.JTextField.RIGHT);
```

```
txtTotal.setText("0.00");
```

```
txtTotal.setBorder(null);
```

```
jLabel3.setFont(new java.awt.Font("TI-Nspire", 1, 18)); // NOI18N
```

```
jLabel3.setForeground(new java.awt.Color(255, 204, 0));
```

```
jLabel3.setText("TOTAL $");
```

```
jLabel2.setFont(new java.awt.Font("TI-Nspire", 0, 14)); // NOI18N
```

```
jLabel2.setForeground(new java.awt.Color(0, 255, 255));
```

```
jLabel2.setText("I.V.A. 12%");
```

```
txtSubtotal.setBackground(new java.awt.Color(153, 0, 51));
```

```
txtSubtotal.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
txtSubtotal.setForeground(new java.awt.Color(255, 255, 255));
```

```
txtSubtotal.setHorizontalAlignment(javax.swing.JTextField.RIGHT);
```

```
txtSubtotal.setText("0.00");
```

```
txtSubtotal.setBorder(null);
```

```
lblSubtotal.setFont(new java.awt.Font("TI-Nspire", 0, 14)); // NOI18N
```

```
lblSubtotal.setForeground(new java.awt.Color(0, 255, 255));
```

```
lblSubtotal.setText("SubTotal $");
```

```
txtIva.setBackground(new java.awt.Color(153, 0, 51));
```

```
txtIva.setFont(new java.awt.Font("Tahoma", 0, 18)); // NOI18N
```

```
txtIva.setForeground(new java.awt.Color(255, 255, 255));
```

```

txtlva.setHorizontalAlignment(javax.swing.JTextField.RIGHT);
txtlva.setText("0.00");
txtlva.setBorder(null);

```

```

    javax.swing.GroupLayout pnlPagoLayout = new
javax.swing.GroupLayout(pnlPago);
    pnlPago.setLayout(pnlPagoLayout);
    pnlPagoLayout.setHorizontalGroup(

pnlPagoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(pnlPagoLayout.createSequentialGroup()
        .addGap(
            .addGroup(pnlPagoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                .addComponent(lblSubtotal,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel2,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(jLabel3,
javax.swing.GroupLayout.PREFERRED_SIZE, 76,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addGroup(pnlPagoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
                .addComponent(txtSubtotal)

```



```

        .addComponent(txtIva)
        .addComponent(txtTotal,
javax.swing.GroupLayout.PREFERRED_SIZE, 95,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(23, Short.MAX_VALUE)
    );
    pnlPagoLayout.setVerticalGroup(

pnlPagoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(pnlPagoLayout.createSequentialGroup()
        .addContainerGap()

.addGroup(pnlPagoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
        .addComponent(txtSubtotal,
javax.swing.GroupLayout.DEFAULT_SIZE, 34, Short.MAX_VALUE)
        .addComponent(lblSubtotal,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(pnlPagoLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
        .addComponent(txtIva, javax.swing.GroupLayout.DEFAULT_SIZE,
32, Short.MAX_VALUE)
        .addComponent(jLabel2,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

.addGroup(pnlPagoLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
        .addComponent(txtTotal)
        .addComponent(jLabel3,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
);

```

```

pnlSistFac.setBackground(new java.awt.Color(153, 0, 51));
pnlSistFac.setForeground(new java.awt.Color(153, 0, 102));

```

```

jLabel1.setFont(new java.awt.Font("Vladimir Script", 1, 48)); // NOI18N
jLabel1.setForeground(new java.awt.Color(255, 255, 255));
jLabel1.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel1.setText("Sistema Facturación");

```

```

jLabel4.setFont(new java.awt.Font("Vladimir Script", 1, 48)); // NOI18N
jLabel4.setForeground(new java.awt.Color(255, 255, 255));
jLabel4.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
jLabel4.setText("UDLA");

```

```

javax.swing.GroupLayout pnlSistFacLayout = new
javax.swing.GroupLayout(pnlSistFac);
pnlSistFac.setLayout(pnlSistFacLayout);
pnlSistFacLayout.setHorizontalGroup(

```

```

pnlSistFacLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)

```

```

        .addGroup(pnlSistFacLayout.createSequentialGroup()

.pnlSistFacLayout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
        .addGroup(pnlSistFacLayout.createSequentialGroup()
            .addGap(41, 41, 41)
            .addComponent(jLabel4,
javax.swing.GroupLayout.PREFERRED_SIZE, 396,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(pnlSistFacLayout.createSequentialGroup()
            .addGap(23, 23, 23)
            .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 396,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    pnlSistFacLayout.setVerticalGroup(

pnlSistFacLayout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(pnlSistFacLayout.createSequentialGroup()
            .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 47,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jLabel4,
javax.swing.GroupLayout.PREFERRED_SIZE, 47,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(29, Short.MAX_VALUE))

```

```
);
```

```
pnlBotones.setBackground(new java.awt.Color(255, 255, 153));
```

```
pnlBotones.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));
```

```
jPanel1.setBackground(new java.awt.Color(255, 255, 153));
```

```
jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder("AgregarClient  
"));
```

```
    btnAgregar.setIcon(new  
    javax.swing.ImageIcon(getClass().getResource("/ICONOS/agregar.jpg"))); //  
    NOI18N
```

```
    btnAgregar.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));
```

```
    btnAgregar.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
```

```
    btnAgregar.addActionListener(new java.awt.event.ActionListener() {  
        public void actionPerformed(java.awt.event.ActionEvent evt) {  
            btnAgregarActionPerformed(evt);  
        }  
    });
```

```
    javax.swing.GroupLayout jPanel1Layout = new  
    javax.swing.GroupLayout(jPanel1);  
    jPanel1.setLayout(jPanel1Layout);  
    jPanel1Layout.setHorizontalGroup(
```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(btnAgregar,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addContainerGap())
    );
jPanel1Layout.setVerticalGroup(
```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(btnAgregar,
javax.swing.GroupLayout.PREFERRED_SIZE, 40, Short.MAX_VALUE)
    );
```

```
jPanel2.setBackground(new java.awt.Color(255, 255, 153));
```

```
jPanel2.setBorder(javax.swing.BorderFactory.createTitledBorder("Calculadora")
);
```

```
    btnCalculadora.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/calcu.jpg"))); //
NOI18N
```

```
btnCalculadora.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.
swing.border.BevelBorder.RAISED));
```

```
btnCalculadora.setHorizontalTextPosition(javax.swing.SwingConstants.CENTE
R);
```

```

btnCalculadora.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnCalculadoraActionPerformed(evt);
    }
});

```

```

javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addGap(10, 10, 10)
        .addComponent(btnCalculadora)
        .addGap(10, 10, 10)
        .addComponent(jPanel3, javax.swing.GroupLayout.PREFERRED_SIZE,
Short.MAX_VALUE)
    );
jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addComponent(btnCalculadora,
javax.swing.GroupLayout.PREFERRED_SIZE, 39, Short.MAX_VALUE)
    );

jPanel3.setBackground(new java.awt.Color(255, 255, 153));

jPanel3.setBorder(javax.swing.BorderFactory.createTitledBorder("Cant_Prod"));

```

```

        btnModificar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/modificar.jpg"))); //
NOI18N

        btnModificar.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swi
ng.border.BevelBorder.RAISED));

        btnModificar.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
        btnModificar.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnModificarActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
        jPanel3.setLayout(jPanel3Layout);
        jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel3Layout.createSequentialGroup()
                .addGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                .addComponent(btnModificar)
                .addGap())
        );
        jPanel3Layout.setVerticalGroup(

```

```
jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(btnModificar,
javax.swing.GroupLayout.PREFERRED_SIZE, 39, Short.MAX_VALUE)
    );
```

```
jPanel4.setBackground(new java.awt.Color(255, 255, 153));
```

```
jPanel4.setBorder(javax.swing.BorderFactory.createTitledBorder("Eliminar_Prod"));
```

```
    btnEliminar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/eliminar.jpg"))); //
NOI18N
```

```
btnEliminar.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
border.BevelBorder.RAISED));
```

```
btnEliminar.setHorizontalTextPosition(javax.swing.SwingConstants.CENTER);
    btnEliminar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnEliminarActionPerformed(evt);
        }
    });
```

```
    javax.swing.GroupLayout jPanel4Layout = new
javax.swing.GroupLayout(jPanel4);
    jPanel4.setLayout(jPanel4Layout);
    jPanel4Layout.setHorizontalGroup(
```



```
jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel4Layout.createSequentialGroup()
        .addGap(18, 18, 18)
        .addComponent(btnEliminar)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    jPanel4Layout.setVerticalGroup(
```

```
jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel4Layout.createSequentialGroup()
        .addGap(0, 0, Short.MAX_VALUE)
        .addComponent(btnEliminar))
    );
```

```
    javax.swing.GroupLayout pnlBotonesLayout = new
javax.swing.GroupLayout(pnlBotones);
    pnlBotones.setLayout(pnlBotonesLayout);
    pnlBotonesLayout.setHorizontalGroup(
```

```
pnlBotonesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(pnlBotonesLayout.createSequentialGroup()
        .addContainerGap()
```

```
    .addGroup(pnlBotonesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
```

```

        .addComponent(jPanel3,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jPanel2,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jPanel1,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jPanel4,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    pnlBotonesLayout.setVerticalGroup(

pnlBotonesLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LE
ADING)
        .addGroup(pnlBotonesLayout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jPanel3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jPanel4, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
);

```

```

btnPagar.setFont(new java.awt.Font("TI-Nspire", 1, 12)); // NOI18N
btnPagar.setText("PAGAR");
btnPagar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnPagarActionPerformed(evt);
    }
});

```

```

javax.swing.GroupLayout pnlClienteTablaLayout = new
javax.swing.GroupLayout(pnlClienteTabla);
pnlClienteTabla.setLayout(pnlClienteTablaLayout);
pnlClienteTablaLayout.setHorizontalGroup(

```

```

pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment
.LEADING)
    .addGroup(pnlClienteTablaLayout.createSequentialGroup()
        .addContainerGap()

```

```

.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(pnlClienteTablaLayout.createSequentialGroup()
        .addGap(32, 32, 32)
        .addComponent(pnlMail,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(btnPagar,
javax.swing.GroupLayout.PREFERRED_SIZE, 88,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)
        .addComponent(pnlSistFac,
javax.swing.GroupLayout.PREFERRED_SIZE, 428,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(pnlPago,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE))
    .addGroup(pnlClienteTablaLayout.createSequentialGroup())

.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(pnlClienteTablaLayout.createSequentialGroup())

```

```
.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
.addGroup(pnlClienteTablaLayout.createSequentialGroup()  
    .addComponent(lblVendedor,  
javax.swing.GroupLayout.PREFERRED_SIZE, 84,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
    .addComponent(txtVendedor,  
javax.swing.GroupLayout.PREFERRED_SIZE, 203,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addGroup(pnlClienteTablaLayout.createSequentialGroup())
```

```
.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
```

```
.addGroup(javax.swing.GroupLayout.Alignment.LEADING,  
pnlClienteTablaLayout.createSequentialGroup()  
    .addComponent(lblCedula,  
javax.swing.GroupLayout.PREFERRED_SIZE, 84,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
    .addComponent(txtCedula,  
javax.swing.GroupLayout.PREFERRED_SIZE, 115,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addComponent(lblTelefono,
javax.swing.GroupLayout.PREFERRED_SIZE, 84,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addComponent(txtTelefono))

.addGroup(pnlClienteTablaLayout.createSequentialGroup()
.addComponent(lblCliente,
javax.swing.GroupLayout.PREFERRED_SIZE, 84,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addComponent(txtNombreCliente,
javax.swing.GroupLayout.PREFERRED_SIZE, 338,
javax.swing.GroupLayout.PREFERRED_SIZE)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(pnlClienteTablaLayout.createSequentialGroup()
.addComponent(lblCorreo,
javax.swing.GroupLayout.PREFERRED_SIZE, 93,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addComponent(txtCorreo,
javax.swing.GroupLayout.PREFERRED_SIZE, 206,
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```

.addGroup(pnlClienteTablaLayout.createSequentialGroup()
            .addComponent(lblDireccion,
                javax.swing.GroupLayout.PREFERRED_SIZE, 93,
                javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(txtDireccion)
            .addComponent(btnClientes,
                javax.swing.GroupLayout.DEFAULT_SIZE, 206, Short.MAX_VALUE))))))
            .addGap(0, 0, Short.MAX_VALUE))
            .addComponent(pnlSTabla)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

            .addComponent(pnlBotones,
                javax.swing.GroupLayout.PREFERRED_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.PREFERRED_SIZE)))
            .addGap(10, 10, 10))
        );
        pnlClienteTablaLayout.setVerticalGroup(

pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(pnlClienteTablaLayout.createSequentialGroup()
                .addContainerGap()

```

```
.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(pnlBotones,  
        javax.swing.GroupLayout.PREFERRED_SIZE,  
        javax.swing.GroupLayout.DEFAULT_SIZE,  
        javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addGroup(pnlClienteTablaLayout.createSequentialGroup())
```

```
.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(lblCliente,  
        javax.swing.GroupLayout.PREFERRED_SIZE, 26,  
        javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtNombreCliente,  
        javax.swing.GroupLayout.PREFERRED_SIZE,  
        javax.swing.GroupLayout.DEFAULT_SIZE,  
        javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtCorreo,  
        javax.swing.GroupLayout.PREFERRED_SIZE,  
        javax.swing.GroupLayout.DEFAULT_SIZE,  
        javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(lblCorreo,  
        javax.swing.GroupLayout.PREFERRED_SIZE, 26,  
        javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
```



```
        .addComponent(lblCedula,
javax.swing.GroupLayout.PREFERRED_SIZE, 26,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtCedula,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(lblTelefono,
javax.swing.GroupLayout.PREFERRED_SIZE, 26,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtTelefono,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(lblDireccion,
javax.swing.GroupLayout.PREFERRED_SIZE, 26,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtDireccion,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(lblVendedor,
javax.swing.GroupLayout.PREFERRED_SIZE, 26,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtVendedor,
javax.swing.GroupLayout.PREFERRED_SIZE,
```

```

javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(btnClientes))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(pnlSTabla,
javax.swing.GroupLayout.PREFERRED_SIZE, 260,
javax.swing.GroupLayout.PREFERRED_SIZE)))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(pnlClienteTablaLayout.createParallelGroup(javax.swing.GroupLayout.
Alignment.LEADING, false)
            .addGroup(pnlClienteTablaLayout.createSequentialGroup()
                .addGap(29, 29, 29)
                .addComponent(pnlSistFac,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addComponent(pnlPago,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(btnPagar,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(pnlMail,
javax.swing.GroupLayout.PREFERRED_SIZE, 105,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addContainerGap(13, Short.MAX_VALUE))
    );

```

```

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(1, 1, 1)

            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                .addComponent(pnlClienteTabla,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(pnlCabecera,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addGap(0, 0, 0)
            );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(1, 1, 1)
                .addComponent(pnlCabecera,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(0, 0, 0)
                .addComponent(pnlClienteTabla,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addContainerGap(44, Short.MAX_VALUE)
    );

    pack();
} // </editor-fold>

static long subtotal=0;
static Object productos[][] ;
static base_datos base_de_datos = new base_datos();
static JDialog dlg_busqueda_cliente = new JDialog();//espacio en la interfaz
destinado al JPanel para buscar cliente

private void btnClientesActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    BusquedaCliente clientes = new BusquedaCliente();//definición de objeto
de la clase BusquedaCliente (es un JPanel de visualización y búsqueda)
    dlg_busqueda_cliente.repaint();//regrafica en el caso de sobrescribir el
jDialog dgl_busqueda_cliente
    dlg_busqueda_cliente.add(clientes);//se agrega el JPanel de Buscar
Cliente
    dlg_busqueda_cliente.setSize(900, 300);//se define tamaño de panel
dentro del Frame Factura
    dlg_busqueda_cliente.setVisible(true);//se hace visible el JPanel

}

static JDialog dlg_agregar_cliente = new JDialog();//espacio en la interfaz
destinado al JPanel para agregar cliente nuevo

```

```

private void btnAgregarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    RegistrarCliente registrar = new RegistrarCliente();
    dlg_agregar_cliente.repaint();//regrafica en el caso de sobrescribir el
jDialog dgl_busqueda_cliente
    dlg_agregar_cliente.add(registrar);//se agrega el JPanel de Buscar Cliente
    dlg_agregar_cliente.setSize(520, 380);//se define tamaño de panel dentro
del Frame Factura
    dlg_agregar_cliente.setVisible(true);//se hace visible el JPanel
}

```

```

private void btnModificarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    System.out.println(fila);
    String cantidad_actual=String.valueOf(jTableProductos.getValueAt(fila,
3));
    long restale =
Long.parseLong(cantidad_actual)*Long.parseLong(String.valueOf(jTableProduc
tos.getValueAt(fila, 2)));
    String cantidad_nueva=null;
    int bandera_numeros=0;

    while(bandera_numeros==0){
        cantidad_nueva=JOptionPane.showInputDialog(null, "INGRESE
NUEVA CANTIDAD", "INGRESE CANTIDAD",
JOptionPane.OK_CANCEL_OPTION);
        if(cantidad_nueva==null){
            bandera_numeros=0;
            break;
        }
    }
}

```

```

        System.out.println("*****VALOR DE
CANTIDAD"+cantidad_nueva);
        for(int j = 0;j<cantidad_nueva.length();j++){
            int caracter=cantidad_nueva.charAt(j);
            if(caracter<48 || caracter>57 || j>3){
                bandera_numeros=0;
                JOptionPane.showMessageDialog(null, "DATO NO
VÁLIDO","ERROR DE CANTIDAD",JOptionPane.ERROR_MESSAGE);
                break;
            }else{
                bandera_numeros=1;
            }
            System.out.println("*****CHAR"+caracter);
        }
    }

    if(bandera_numeros==1){
        jTableProductos.setValueAt(cantidad_nueva,filas,3);
        long sumale =
Long.parseLong(cantidad_nueva)*Long.parseLong(String.valueOf(jTableProdu
ctos.getValueAt(filas, 2)));
        subtotal=subtotal-restale;
        jTableProductos.setValueAt(sumale,filas,4);
        subtotal=subtotal+sumale;
        long iva=(long) (0.12*subtotal);
        long total = iva +subtotal;
        txtSubtotal.setText(String.valueOf(subtotal));
        txtIva.setText(String.valueOf(iva));
        txtTotal.setText(String.valueOf(total));
    }
}

```

```

}

private void btnCalculadoraActionPerformed(java.awt.event.ActionEvent evt)
{
    try {
        // TODO add your handling code here:
        Runtime rt = Runtime.getRuntime();
        Process p = rt.exec("calc");
        try {
            p.waitFor();
        } catch (InterruptedException ex) {
            Logger.getLogger(Factura.class.getName()).log(Level.SEVERE, null,
ex);
        }
    } catch (IOException ex) {
        Logger.getLogger(Factura.class.getName()).log(Level.SEVERE, null,
ex);
    }
}
}

```

```

private void btnMailActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    if(jTableProductos.getRowCount(>0)){
        this.jProgressBar.setIndeterminate(true);
        JOptionPane.showMessageDialog(null,"LA FACTURA SE ENVIARÁ
A\n"+txtCorreo.getText(),"ACEPTAR
ENVÍO",JOptionPane.QUESTION_MESSAGE);
        base_de_datos.inicio();
    }
}

```

```

        int numero_factura=base_de_datos.obtener_ultima_factura()+1;
        base_de_datos.insertar_proxima_factura(numero_factura);
        this.envia_mail();
        this.limpiarTabla();
        btnMail.setEnabled(false);
        btnClientes.setEnabled(false);
    }else{
        JOptionPane.showMessageDialog(null, "NINGÚN PRODUCTO
        COMPRADO","FACTURA VACÍA",JOptionPane.ERROR_MESSAGE);
    }
}

private void btnEliminarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String cantidad_actual=String.valueOf(jTableProductos.getValueAt(fila,
3));
    long restale =
    Long.parseLong(cantidad_actual)*Long.parseLong(String.valueOf(jTableProduc
tos.getValueAt(fila, 2)));
    subtotal=subtotal-restale;
    long iva=(long) (0.12*subtotal);
    long total = iva +subtotal;
    txtSubtotal.setText(String.valueOf(subtotal));
    txtIva.setText(String.valueOf(iva));
    txtTotal.setText(String.valueOf(total));
    modelo_tabla.removeRow(fila);
    btnModificar.setEnabled(false);
    btnEliminar.setEnabled(false);
}

```



```
int fila ;
```

```
private void jTableProductosMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    fila = jTableProductos.getSelectedRow();
    System.out.println(fila);
    if(jTableProductos.getSelectedRow()>-1){
        btnModificar.setEnabled(true);
        btnEliminar.setEnabled(true);
    }
}
}
```

```
public static JDialog dlg_pagar = new JDialog();//espacio en la interfaz
destinado al JPanel para agregar cliente nuevo
```

```
private void btnPagarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    // TODO add your handling code here:
    PagarTarjeta pagar = new PagarTarjeta();
    dlg_pagar.repaint();//regrafica en el caso de sobrecribir el JDialog
dgl_busqueda_cliente
    dlg_pagar.add(pagar);//se agrega el JPanel de Buscar Cliente
    dlg_pagar.setSize(520, 380);//se define tamaño de panel dentro del Frame
Factura
    dlg_pagar.setVisible(true);//se hace visible el JPanel
    dlg_pagar.setDefaultCloseOperation(DO_NOTHING_ON_CLOSE);
    dlg_pagar.setLocationRelativeTo(null);
}
}
```

//método para actualizar campos de texto de frame Factura, se define static porque los datos provienen de otra clase, los campo de textos también han sido definidos como static

```
public static void escribe_datos(String cedula, String cliente, String correo,
String telefono, String direccion){
    txtNombreCliente.setText(cliente);
    txtCedula.setText(cedula);
    txtCorreo.setText(correo);
    txtDireccion.setText(direccion);
    txtTelefono.setText(telefono);
}
```

//método para ocultar los jPanels de búsqueda y de aregar una vez que han cumplido con su función

```
public static void esconde_dlg(){
    dlg_agregar_cliente.hide();
    dlg_busqueda_cliente.hide();
    dlg_pagar.hide();
}
```

```
// static int subtotal=0;
```

```
// static Object productos[][] ;
```

```
// static base_datos base_de_datos = new base_datos();
```

```
public static void RF_ID(String RF_ID){
    String ID = RF_ID;
    base_de_datos.inicio();
    productos = base_de_datos.imprime_productos(ID);
    System.out.println("veamosssssssssssssssssssss");
    if(productos!=null){
        System.out.println(productos[0][0]);
    }
}
```

```

System.out.println(productos[0][1]);
System.out.println(productos[0][2]);
System.out.println(productos[0][3]);
System.out.println(productos[0][4]);
if( !"0".equals(productos[0][3].toString())){
    modelo_tabla.addRow(productos[0]);
}
subtotal=subtotal+Integer.parseInt(productos[0][4].toString());
long iva=(long) (0.12*subtotal);
long total = iva +subtotal;
txtSubtotal.setText(String.valueOf(subtotal));
txtIva.setText(String.valueOf(iva));
txtTotal.setText(String.valueOf(total));
}
}

```

```

public void limpiarTabla(){
    int filas=jTableProductos.getRowCount();
    for (int i = 0;filas>i; i++) {
        modelo_tabla.removeRow(0);
        System.out.println("entro limpieza");
    }
}

```

```

public void envia_mail(){
    String nombre0 =
lblFecha.getText()+lblhora.getText()+txtNombreCliente.getText();
    String nombre = nombre0.replaceAll(" ", "");
    nombre = nombre.replaceAll("/", "");
}

```

```

    nombre = nombre.replaceAll(":", "");
    System.out.println(nombre);
    String directorio = "C:\\Users\\ING. EDUARDO MOLINA\\Desktop\\Tesis
PMolina\\TESIS PAÚL
MOLINA\\SISTEMALECTURATAGS\\FACTURAS\\"+nombre;

PdfCreador.crea(directorio,jTableProductos,txtSubtotal.getText(),txtIva.getText(
),txtTotal.getText(),

txtCode.getText(),txtNombreCliente.getText(),txtCedula.getText(),txtDireccion.g
etText(),

txtTelefono.getText(),txtVendedor.getText(),lblFecha.getText(),lblhora.getText())
;
    mail.adjunto(txtCorreo.getText(),"FACTURA","GRACIAS POR
PREFERIRNOS",nombre,directorio);

}

public static void SetearVendedor(String vendedor){
    txtVendedor.setText(vendedor);
}

// Variables declaration - do not modify
private javax.swing.JButton btnAgregar;
private javax.swing.JButton btnCalculadora;
public static javax.swing.JButton btnClientes;
private javax.swing.JButton btnEliminar;
public static javax.swing.JButton btnMail;
private javax.swing.JButton btnModificar;

```

```
public static javax.swing.JButton btnPagar;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JPanel jPanel4;
public static javax.swing.JProgressBar jProgressBar;
public static javax.swing.JTable jTableProductos;
private javax.swing.JLabel lblCedula;
private javax.swing.JLabel lblCliente;
private javax.swing.JLabel lblCorreo;
private javax.swing.JLabel lblDireccion;
private javax.swing.JLabel lblFecha;
private javax.swing.JLabel lblNum;
private javax.swing.JLabel lblSubtotal;
private javax.swing.JLabel lblTelefono;
private javax.swing.JLabel lblVendedor;
private javax.swing.JLabel lblhora;
private javax.swing.JPanel pnlBotones;
private javax.swing.JPanel pnlCabecera;
private javax.swing.JPanel pnlClienteTabla;
private javax.swing.JPanel pnlMail;
private javax.swing.JPanel pnlPago;
private javax.swing.JScrollPane pnlSTabla;
private javax.swing.JPanel pnlSistFac;
public static javax.swing.JTextField txtCedula;
public static javax.swing.JTextField txtCode;
public static javax.swing.JTextField txtCorreo;
```

```

    public static javax.swing.JTextField txtDireccion;
    public static javax.swing.JTextField txtIva;
    public static javax.swing.JTextField txtNombreCliente;
    public static javax.swing.JTextField txtSubtotal;
    public static javax.swing.JTextField txtTelefono;
    public static javax.swing.JTextField txtTotal;
    public static javax.swing.JTextField txtVendedor;
    // End of variables declaration
}

```

Interface MenuPrincipal.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * MenuPrincipal.java
 *
 * Creada el 12 de noviembre de 2013
 */
package INTERFACES;

import CONTROLADOR.ComSerial;
import CONTROLADOR.base_datos;
import java.awt.Color;
import javax.swing.JDialog;
import javax.swing.JOptionPane;
import javax.swing.UIManager;

```

```

/**
 *
 * @author Paul Molina
 */

public class MenuPrincipal extends javax.swing.JFrame {

    base_datos base_de_datos = new base_datos();

    /** Creates new form MenuPrincipal */
    public MenuPrincipal() {
        initComponents();
        this.setLocationRelativeTo(this);
        pnlFiguras.setVisible(true);//se visualiza panel de bienvenida
        pnlInicio.setVisible(false);//se esconde panel inicio
        miFacturaActual.setEnabled(false);//se esconde panel contenedor de
internalFrame factura
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">
    private void initComponents() {

        filemen = new javax.swing.JMenuBar();
        jMenu3 = new javax.swing.JMenu();

```

```
jMenu4 = new javax.swing.JMenu();
pnlFiguras = new javax.swing.JPanel();
lblJava = new javax.swing.JLabel();
lblUdla = new javax.swing.JLabel();
lblLema = new javax.swing.JLabel();
lblTitulo = new javax.swing.JLabel();
lblflecha = new javax.swing.JLabel();
pnlInicio = new javax.swing.JPanel();
jmenuMenu = new javax.swing.JMenuBar();
fmUsuarios = new javax.swing.JMenu();
miSesion = new javax.swing.JMenuItem();
miNuevo_Usuario = new javax.swing.JMenuItem();
miModificar = new javax.swing.JMenuItem();
jmFacturas = new javax.swing.JMenu();
miFacturaActual = new javax.swing.JMenuItem();
miNuevaFactura = new javax.swing.JMenuItem();
miNuevoPro = new javax.swing.JMenuItem();
fmSalir = new javax.swing.JMenu();
```

```
jMenu3.setText("File");
filemen.add(jMenu3);
```

```
jMenu4.setText("Edit");
filemen.add(jMenu4);
```

```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
setTitle("Facturación UDLA");
setBackground(new java.awt.Color(140, 0, 39));
setResizable(false);
```



```
pnlFiguras.setBackground(new java.awt.Color(140, 0, 39));  
pnlFiguras.setDoubleBuffered(false);  
pnlFiguras.setPreferredSize(new java.awt.Dimension(900, 530));
```

```
    lblJava.setIcon(new  
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/java.jpg"))); //  
NOI18N
```

```
    lblUdla.setIcon(new  
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/udla.jpg"))); //  
NOI18N
```

```
    lblLema.setFont(new java.awt.Font("TI-Nspire Sans", 2, 48)); // NOI18N  
    lblLema.setForeground(new java.awt.Color(255, 255, 255));  
    lblLema.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
    lblLema.setText("¡Porque amamos lo que hacemos!");
```

```
    lblTitulo.setBackground(new java.awt.Color(255, 255, 255));  
    lblTitulo.setFont(new java.awt.Font("TI-Nspire", 3, 48)); // NOI18N  
    lblTitulo.setForeground(new java.awt.Color(255, 255, 255));  
    lblTitulo.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
    lblTitulo.setText("SISTEMA DE FACTURACIÓN");
```

```
    lblflecha.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);  
    lblflecha.setIcon(new  
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/flecha.jpg"))); //  
NOI18N
```

```
    javax.swing.GroupLayout pnlFigurasLayout = new  
javax.swing.GroupLayout(pnlFiguras);  
    pnlFiguras.setLayout(pnlFigurasLayout);
```

```

    pnlFigurasLayout.setHorizontalGroup(

pnlFigurasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(pnlFigurasLayout.createSequentialGroup()
        .addComponent(lblLema, javax.swing.GroupLayout.DEFAULT_SIZE,
890, Short.MAX_VALUE)
        .addContainerGap())
    .addGroup(pnlFigurasLayout.createSequentialGroup()
        .addGap(77, 77, 77)
        .addComponent(lblUdla,
javax.swing.GroupLayout.PREFERRED_SIZE, 347,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(lblJava)
        .addGap(76, 76, 76))
    .addGroup(pnlFigurasLayout.createSequentialGroup()
        .addGap(67, 67, 67)
        .addComponent(lblflecha,
javax.swing.GroupLayout.PREFERRED_SIZE, 64,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATE
D)
        .addComponent(lblTitulo, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addGap(22, 22, 22))
    );
    pnlFigurasLayout.setVerticalGroup(

```

```

pnlFigurasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(pnlFigurasLayout.createSequentialGroup()
        .addContainerGap()

.addGroup(pnlFigurasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(lblTitulo,
javax.swing.GroupLayout.PREFERRED_SIZE, 100,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(lblflecha,
javax.swing.GroupLayout.PREFERRED_SIZE, 70,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addGap(31, 31, 31)

.addGroup(pnlFigurasLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(lblUdla,
javax.swing.GroupLayout.PREFERRED_SIZE, 225,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(lblJava,
javax.swing.GroupLayout.PREFERRED_SIZE, 217,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
91, Short.MAX_VALUE)
    .addComponent(lblLema)
    .addGap(57, 57, 57))
);

pnlInicio.setBackground(new java.awt.Color(140, 0, 39));

```

```

        javax.swing.GroupLayout pnlInicioLayout = new
javax.swing.GroupLayout(pnlInicio);
        pnlInicio.setLayout(pnlInicioLayout);
        pnlInicioLayout.setHorizontalGroup(

pnlInicioLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 900, Short.MAX_VALUE)
        );
        pnlInicioLayout.setVerticalGroup(

pnlInicioLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 580, Short.MAX_VALUE)
        );

jmenuMenu.setBackground(new java.awt.Color(255, 255, 153));

fmUsuarios.setBackground(new java.awt.Color(51, 204, 255));
fmUsuarios.setText("Usuarios");

miSesion.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event.KeyEvent.VK_S, java.awt.event.InputEvent.CTRL_MASK));
        miSesion.setText("Reiniciar Sesión");
        miSesion.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                miSesionActionPerformed(evt);
            }
        });

```

```
fmUsuarios.add(miSesion);
```

```
miNuevo_Usuario.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt
.event.KeyEvent.VK_A, java.awt.event.InputEvent.CTRL_MASK));
```

```
miNuevo_Usuario.setText("Nuevo Usuario");
```

```
miNuevo_Usuario.addActionListener(new java.awt.event.ActionListener() {
```

```
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
        miNuevo_UsuarioActionPerformed(evt);
```

```
    }
```

```
});
```

```
fmUsuarios.add(miNuevo_Usuario);
```

```
miModificar.setAccelerator(javax.swing.KeyStroke.getKeyStroke(java.awt.event
.KeyEvent.VK_E, java.awt.event.InputEvent.CTRL_MASK));
```

```
miModificar.setText("Modificar Usuario");
```

```
miModificar.addActionListener(new java.awt.event.ActionListener() {
```

```
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
        miModificarActionPerformed(evt);
```

```
    }
```

```
});
```

```
fmUsuarios.add(miModificar);
```

```
jmenuMenu.add(fmUsuarios);
```

```
jmFacturas.setBackground(new java.awt.Color(255, 255, 0));
```

```
jmFacturas.setBorder(null);
```

```
jmFacturas.setForeground(new java.awt.Color(0, 153, 153));
```

```
jmFacturas.setText("Facturación");
```

```
miFacturaActual.setText("Factura Actual");
miFacturaActual.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        miFacturaActualActionPerformed(evt);
    }
});
jmFacturas.add(miFacturaActual);
```

```
miNuevaFactura.setBackground(new java.awt.Color(255, 255, 0));
miNuevaFactura.setText("Nueva Factura");
miNuevaFactura.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        miNuevaFacturaActionPerformed(evt);
    }
});
jmFacturas.add(miNuevaFactura);
```

```
miNuevoPro.setText("Nuevo Producto");
miNuevoPro.setToolTipText("Sólo Administradores");
miNuevoPro.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        miNuevoProActionPerformed(evt);
    }
});
jmFacturas.add(miNuevoPro);
```

```
jmenuMenu.add(jmFacturas);
```

```
fmSalir.setBackground(new java.awt.Color(0, 255, 153));
fmSalir.setText("Salir");
fmSalir.addMouseListener(new java.awt.event.MouseAdapter() {
```

```
    public void mouseClicked(java.awt.event.MouseEvent evt) {
        fmSalirMouseClicked(evt);
    }
});
jmenuMenu.add(fmSalir);

setJMenuBar(jmenuMenu);

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .add(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(pnlInicio, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addContainerGap())
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
                .addGroup(layout.createSequentialGroup()
                    .addComponent(pnlFiguras,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(0, 20, Short.MAX_VALUE)))
        )
    );
layout.setVerticalGroup(
```

```

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(pnlInicio, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
    .addComponent(pnlFiguras,
javax.swing.GroupLayout.Alignment.TRAILING,
javax.swing.GroupLayout.DEFAULT_SIZE, 580, Short.MAX_VALUE))
);

pack();
} // </editor-fold>

```

//este metodo es invocado desde el inicio de sesión y es el encargado de
//configurar la interfaz según el tipo de usuario que inicio sesión

```

public void bandera_tipo_usuario(String desde_base){
    String tipo_usuario=desde_base;
    if("A".equals(tipo_usuario)){
        miNuevoPro.setEnabled(true);
        miModificar.setEnabled(true);
        miNuevo_Usuario.setEnabled(true);
    }else{
        miNuevoPro.setEnabled(false);
        miModificar.setEnabled(false);
        miNuevo_Usuario.setEnabled(false);
    }
}
}

```



```
RegistrarUsuario usuario = new RegistrarUsuario();//objeto de la clase
registrarUsuario
static JDialog dlg_agregar_usuario = new JDialog();//espacio en la interfaz
destinado al JPanel para agregar cliente nuevo

private void miNuevo_UsuarioActionPerformed(java.awt.event.ActionEvent evt)
{
    dlg_agregar_usuario.repaint();//regrafica en el caso de sobrescribir el
jDialog dgl_busqueda_cliente
    dlg_agregar_usuario.add(usuario);//se agrega el JPanel de Buscar Cliente
    dlg_agregar_usuario.setSize(500, 300);//se define tamaño de panel dentro
del Frame Factura
    dlg_agregar_usuario.setVisible(true);//se hace visible el JPanel
    dlg_agregar_usuario.setTitle("NUEVO USUARIO");
}

//esconde diferentes jPnels de comunicacion al hacer clic en cerrar en alguno
de ellos
public static void esconde_dlg(){
    dlg_agregar_producto.hide();
    dlg_agregar_usuario.hide();
    dlg_modificar_usuario.hide();
}

private void miSesionActionPerformed(java.awt.event.ActionEvent evt) {
this.setVisible(false);
Acceso N=new Acceso();
N.setVisible(true);
}
```

```
//objetos de factura y de comSerial para interaccion con RFid cards
```

```
Factura factura = new Factura();
```

```
ComSerial com = new ComSerial();
```

```
private void miFacturaActualActionPerformed(java.awt.event.ActionEvent
evt) {
```

```
    // TODO add your handling code here:
```

```
    com.bandera_factura=1;
```

```
    com.bandera_agregar=0;//estado de banderas al estar en inter4faz factura
```

```
    pnlFiguras.setVisible(false);//se esconde panel de bienvenida
```

```
    pnlInicio.setVisible(true);//se visualiza panel donde se coloca interfaz
```

```
(internal frame)factura
```

```
    factura.show();
```

```
    pnlInicio.add(factura);
```

```
    this.repaint();//em caso de error grafico, éste se corrige
```

```
    System.out.println("-----"+com.bandera_factura);//debug para
```

```
verificar estado de bandera
```

```
    Color fondo= new Color(153,0,1);
```

```
    pnlInicio.setBackground(fondo);
```

```
}
```

```
RegistrarProducto producto = new RegistrarProducto();
```

```
static JDialog dlg_agregar_producto = new JDialog();//espacio en la interfaz
destinado al jPanelel para agregar cliente nuevo
```

```
private void miNuevoProActionPerformed(java.awt.event.ActionEvent evt) {
```

```
    // TODO add your handling code here:
```

```
    com.bandera_factura=0;
```

```
    com.bandera_agregar=1;
```

```

        dlg_agregar_producto.repaint();//regrafica en el caso de sobrescribir el
jDialog dgl_busqueda_cliente
        dlg_agregar_producto.add(producto);//se agrega el JPanel de Buscar
Cliente
        dlg_agregar_producto.setSize(500, 300);//se define tamaño de panel
dentro del Frame Factura
        dlg_agregar_producto.setVisible(true);//se hace visible el JPanel

dlg_agregar_producto.setDefaultCloseOperation(DO_NOTHING_ON_CLOSE);
        System.out.println("-----"+com.bandera_factura);

}

```

```

ModificaUsuario mod_usuario = new ModificaUsuario();
static JDialog dlg_modificar_usuario = new JDialog();//espacio en la interfaz
destinado al JPanel para agregar cliente nuevo

private void miModificarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    dlg_modificar_usuario.repaint();//regrafica en el caso de sobrescribir el
jDialog dgl_busqueda_cliente
    dlg_modificar_usuario.add(mod_usuario);//se agrega el JPanel de Buscar
Cliente
    dlg_modificar_usuario.setSize(500, 300);//se define tamaño de panel
dentro del Frame Factura
    dlg_modificar_usuario.setVisible(true);//se hace visible el JPanel
    Color fondo= new Color(153,0,1);
    dlg_modificar_usuario.setBackground(fondo);
}

```

```

private void miNuevaFacturaActionPerformed(java.awt.event.ActionEvent
evt) {
    // TODO add your handling code here:
    //verificacion de inicio de nueva Factura
    if (JOptionPane.showConfirmDialog(null, "Está seguro?\nse iniciará un
nuevo proceso de facturación", "PRECAUCIÓN",
        JOptionPane.YES_NO_OPTION) == JOptionPane.YES_OPTION) {
        base_de_datos.inicio();//se inicia conexion con base de datos
        int numero_factura=base_de_datos.obtener_ultima_factura()+1;//para
generar numero de factura
        miFacturaActual.setEnabled(true);//se activa boton de factura actual
        com.bandera_factura=1;
        com.bandera_agregar=0;//estado de banderas cuando se encuantra en
factura
        pnlFiguras.setVisible(false);
        pnlInicio.setVisible(true);
        factura.show();
        pnlInicio.add(factura);
        factura.btnMail.setEnabled(false);
        factura.txtNombreCliente.setText("nombre");
        factura.txtTelefono.setText("teléfono");
        factura.txtCedula.setText("cédula");
        factura.txtCorreo.setText("correo");
        factura.txtDireccion.setText("dirección");
        factura.txtIva.setText("0");
        factura.txtSubtotal.setText("0");
        factura.txtTotal.setText("0");
        factura.txtCode.setText(String.valueOf(numero_factura+1));
        factura.limpiarTabla();//estado inicial de campos de texto y tabla de
productos
        Color fondo= new Color(153,0,1);

```

```

        pnInicio.setBackground(fondo);
        this.repaint();
        System.out.println("-----"+com.bandera_factura);
        factura.btnClientes.setEnabled(true);
    }else{

    }

}

private void fmSalirMouseClicked(java.awt.event.MouseEvent evt) {
    // TODO add your handling code here:
    //confirmacion al hacer clic en salir
    if (JOptionPane.showConfirmDialog(null, "Está seguro?\nSaldrá del
sistema de Facturación", "PRECAUCIÓN",
    JOptionPane.YES_NO_OPTION) == JOptionPane.YES_OPTION) {
        System.exit(0);
    }else{

    }
}

/**
 * @param args the command line arguments
 */
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">

```

/* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

* For details see

<http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html>

*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info :

javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(MenuPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(MenuPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(MenuPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(MenuPrincipal.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

```
/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {

    public void run() {
        new MenuPrincipal().setVisible(true);
    }
});
}

// Variables declaration - do not modify
private javax.swing.JMenuBar filemen;
private javax.swing.JMenu fmSalir;
private javax.swing.JMenu fmUsuarios;
private javax.swing.JMenu jMenuItem3;
private javax.swing.JMenu jMenuItem4;
private javax.swing.JMenu jmFacturas;
private javax.swing.JMenuBar jmenuMenu;
private javax.swing.JLabel lblJava;
private javax.swing.JLabel lblLema;
private javax.swing.JLabel lblTitulo;
private javax.swing.JLabel lblUdla;
private javax.swing.JLabel lblflecha;
private javax.swing.JMenuItem miFacturaActual;
private javax.swing.JMenuItem miModificar;
private javax.swing.JMenuItem miNuevaFactura;
private javax.swing.JMenuItem miNuevoPro;
private javax.swing.JMenuItem miNuevo_Usuario;
private javax.swing.JMenuItem miSesion;
public static javax.swing.JPanel pnlFiguras;
public static javax.swing.JPanel pnlInicio;
// End of variables declaration
```

```
}
```

Interface ModificaUsuario.java

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package INTERFACES;

import CONTROLADOR.base_datos;
import CONTROLADOR.base_datos_Users;
import java.awt.Color;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;

public class ModificaUsuario extends javax.swing.JPanel {

    base_datos_Users base_de_datos_U = new base_datos_Users();

    /**
     * Creates new form RegistrarCliente
     */
    public ModificaUsuario() {
        initComponents();
        base_de_datos_U.inicio();
        btnActualizar.setEnabled(false);
        MenuPrincipal.dlg_modificar_usuario.setTitle("MODIFICAR USUARIO");
        Color fondo= new Color(153,0,51);
    }
}
```



```
        this.setBackground(fondo);
    }

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    buttonGroup1 = new javax.swing.ButtonGroup();
    jPanel1 = new javax.swing.JPanel();
    txtClave = new javax.swing.JTextField();
    jLabel7 = new javax.swing.JLabel();
    txtApellidos = new javax.swing.JTextField();
    jLabel6 = new javax.swing.JLabel();
    txtCedula = new javax.swing.JTextField();
    txtNombre = new javax.swing.JTextField();
    txtTipoU = new javax.swing.JTextField();
    jLabel4 = new javax.swing.JLabel();
    txtNick = new javax.swing.JTextField();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel5 = new javax.swing.JLabel();
    jPanel2 = new javax.swing.JPanel();
    btnBuscar = new javax.swing.JButton();
    btnSalir = new javax.swing.JButton();
    btnEliminar = new javax.swing.JButton();
    btnActualizar = new javax.swing.JButton();
}
```

```
setBackground(new java.awt.Color(153, 0, 51));

jPanel1.setBackground(new java.awt.Color(153, 0, 51));

jLabel7.setText("TIPO");

jLabel6.setText("CLAVE");

txtCedula.addKeyListener(new java.awt.event.KeyAdapter() {
    public void keyTyped(java.awt.event.KeyEvent evt) {
        txtCedulaKeyTyped(evt);
    }
});

jLabel4.setText("NOMBRES");

jLabel2.setText("CÉDULA");

jLabel3.setText("APELLIDOS");

jLabel5.setText("NICKNAME");

javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
```

```
.addContainerGap()
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.LEADING)
```

```
    .addComponent(jLabel2,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(jLabel3,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(jLabel4,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(jLabel5,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(jLabel6,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(jLabel7,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addGap(23, 23, 23)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.LEADING)
```

```
    .addComponent(txtTipoU,  
javax.swing.GroupLayout.PREFERRED_SIZE, 219,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtClave,  
javax.swing.GroupLayout.PREFERRED_SIZE, 219,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```

        .addComponent(txtNick,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtNombre,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtApellidos,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtCedula,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap()
    );
    jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(jPanel1Layout.createSequentialGroup()

            .addGap(2, 2, 2)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(jLabel2,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)

                        .addComponent(txtCedula,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(jLabel3,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtApellidos,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(jLabel4,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtNombre,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(jLabel5,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtNick,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel6,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtClave,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(jLabel7,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtTipoU,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
);

jPanel2.setBackground(new java.awt.Color(153, 0, 51));
jPanel2.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));

btnBuscar.setText("BUSCAR");
btnBuscar.addActionListener(new java.awt.event.ActionListener() {

```

```
        public void actionPerformed(java.awt.event.ActionEvent evt) {  
            btnBuscarActionPerformed(evt);  
        }  
    });
```

```
    btnSalir.setText("SALIR");  
    btnSalir.addActionListener(new java.awt.event.ActionListener() {  
        public void actionPerformed(java.awt.event.ActionEvent evt) {  
            btnSalirActionPerformed(evt);  
        }  
    });
```

```
    btnEliminar.setText("ELIMINAR");  
    btnEliminar.addActionListener(new java.awt.event.ActionListener() {  
        public void actionPerformed(java.awt.event.ActionEvent evt) {  
            btnEliminarActionPerformed(evt);  
        }  
    });
```

```
    btnActualizar.setText("ACTUALIZAR");  
    btnActualizar.addActionListener(new java.awt.event.ActionListener() {  
        public void actionPerformed(java.awt.event.ActionEvent evt) {  
            btnActualizarActionPerformed(evt);  
        }  
    });
```

```
    javax.swing.GroupLayout jPanel2Layout = new  
    javax.swing.GroupLayout(jPanel2);  
    jPanel2.setLayout(jPanel2Layout);  
    jPanel2Layout.setHorizontalGroup(
```

```
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(btnBuscar,
javax.swing.GroupLayout.PREFERRED_SIZE, 81,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(btnEliminar,
javax.swing.GroupLayout.PREFERRED_SIZE, 81,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(btnActualizar,
javax.swing.GroupLayout.PREFERRED_SIZE, 81,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(btnSalir,
javax.swing.GroupLayout.PREFERRED_SIZE, 81,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
);
```

```
jPanel2Layout.setVerticalGroup(
```

```
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel2Layout.createSequentialGroup())
```

```
        .addContainerGap()
```

```
        .addComponent(btnBuscar,
javax.swing.GroupLayout.PREFERRED_SIZE, 38,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
        .addComponent(btnEliminar,
javax.swing.GroupLayout.PREFERRED_SIZE, 38,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```



```

        .addComponent(btnActualizar,
javax.swing.GroupLayout.PREFERRED_SIZE, 38,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(btnSalir,
javax.swing.GroupLayout.PREFERRED_SIZE, 38,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE)
    );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(this);
    this.setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addGroup(layout.createParallelGroup()
                .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(28, 28, 28)
                .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
            .addContainerGap())
    );
    layout.setVerticalGroup(

```

```

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
    .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
    .addContainerGap(13, Short.MAX_VALUE)
);
} // </editor-fold>

```

```

private void btnBuscarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Object usuario[][] =
base_de_datos_U.imprime_usuarios("cedula",txtCedula.getText());
    if(usuario.length>0){
        txtApellidos.setText(usuario[0][0].toString());
        txtNombre.setText(usuario[0][1].toString());
        txtClave.setText(usuario[0][2].toString());
        txtTipoU.setText(usuario[0][3].toString());
        txtNick.setText(usuario[0][4].toString());
        btnActualizar.setEnabled(true);
    }else{

```

```
JOptionPane.showMessageDialog(null,"No en base de datos","NO SE  
ENCUENTRA CLIENTE",JOptionPane.ERROR_MESSAGE);
```

```
}
```

```
}
```

```
private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
```

```
// TODO add your handling code here:
```

```
MenuPrincipal.esconde_dlg();
```

```
}
```

```
private void btnEliminarActionPerformed(java.awt.event.ActionEvent evt) {
```

```
base_de_datos_U.elimina_usuario(txtCedula.getText());
```

```
this.limpiar_parametros();
```

```
}
```

```
private void btnActualizarActionPerformed(java.awt.event.ActionEvent evt) {
```

```
// TODO add your handling code here:
```

```
base_de_datos_U.actualizar_usuarios(txtCedula.getText(),  
txtNombre.getText(), txtApellidos.getText(), txtClave.getText(),  
txtTipoU.getText(), txtNick.getText());
```

```
btnActualizar.setEnabled(false);
```

```
JOptionPane.showMessageDialog(null, "ACTUALIZACIÓN  
EXITOSA","ACTUALIZACIÓN  
EXITOSA",JOptionPane.INFORMATION_MESSAGE);
```

```
this.limpiar_parametros();
```

```
}
```

```
private void txtCedulaKeyTyped(java.awt.event.KeyEvent evt) {
```

```
// TODO add your handling code here:
```

```
char car = evt.getKeyChar();
```

```
        if((car<'0' || car>'9')) {
            evt.consume();
        }
    }

    public void limpiar_parametros(){
        txtApellidos.setText("");
        txtCedula.setText("");
        txtClave.setText("");
        txtNick.setText("");
        txtNombre.setText("");
        txtTipoU.setText("");
    }

    // Variables declaration - do not modify
    private javax.swing.JButton btnActualizar;
    private javax.swing.JButton btnBuscar;
    private javax.swing.JButton btnEliminar;
    private javax.swing.JButton btnSalir;
    private javax.swing.ButtonGroup buttonGroup1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
    private javax.swing.JTextField txtApellidos;
    private javax.swing.JTextField txtCedula;
    private javax.swing.JTextField txtClave;
```

```

private javax.swing.JTextField txtNick;
private javax.swing.JTextField txtNombre;
private javax.swing.JTextField txtTipoU;
// End of variables declaration
}

```

Interface PagarTarjeta.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package INTERFACES;

import CONTROLADOR.ComSerial;
import CONTROLADOR.base_datos;
import java.awt.Color;
import javax.swing.JOptionPane;

/**
 *
 * @author Paúl Molina
 */
public class PagarTarjeta extends javax.swing.JPanel {

    static base_datos base_de_datos = new base_datos();

    /**
     * Creates new form RegistrarCliente
     */
    public PagarTarjeta() {

```

```
initComponents();
base_de_datos.inicio();
btnConfirmar.setEnabled(false);
ComSerial.bandera_agregar=0;
ComSerial.bandera_factura=0;
ComSerial.bandera_pago=1;
Factura.dlg_pagar.setTitle("PAGAR");

}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    txtPagar = new javax.swing.JTextField();
    txtNombre = new javax.swing.JTextField();
    txtDisponible = new javax.swing.JTextField();
    lblDisponible = new javax.swing.JLabel();
    lblNombre = new javax.swing.JLabel();
    lblAPagar = new javax.swing.JLabel();
    txtCodigo = new javax.swing.JTextField();
    jPanel2 = new javax.swing.JPanel();
    btnConfirmar = new javax.swing.JButton();
    btnSalir = new javax.swing.JButton();
```



```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(jLabel1)
```

```
    .addComponent(lblNombre,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE, 77,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(lblDisponible,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE, 77,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(lblAPagar,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE, 92,
```

```
        javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
```

```
    .addComponent(txtDisponible,
```

```
        javax.swing.GroupLayout.DEFAULT_SIZE, 219, Short.MAX_VALUE)
```

```
    .addComponent(txtPagar,
```

```
        javax.swing.GroupLayout.DEFAULT_SIZE, 219, Short.MAX_VALUE)
```

```
    .addComponent(txtNombre,
```

```
        javax.swing.GroupLayout.DEFAULT_SIZE, 219, Short.MAX_VALUE)
```

```
    .addComponent(txtCodigo))
```

```
    .addContainerGap())
```

```
);
```

```
jPanel1Layout.setVerticalGroup(
```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```



```
.addGroup(jPanel1Layout.createSequentialGroup()  
    .addContainerGap()  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.LEADING)  
    .addComponent(jLabel1)  
    .addComponent(txtCodigo,  
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)  
    .addComponent(lblNombre,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
    .addComponent(txtNombre,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)  
    .addComponent(lblAPagar,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
    .addComponent(txtPagar,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(lblDisponible,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtDisponible,
javax.swing.GroupLayout.PREFERRED_SIZE, 36,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(32, 32, 32))
);

btnConfirmar.setText("Confirmar");
btnConfirmar.setToolTipText("Clic en NUEVO para habilitar opciones");
btnConfirmar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnConfirmarActionPerformed(evt);
    }
});

btnSalir.setBackground(new java.awt.Color(255, 51, 51));
btnSalir.setText("SALIR");
btnSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnSalirActionPerformed(evt);
    }
});

javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);

```

```

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(jPanel2Layout.createSequentialGroup()

        .addContainerGap()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

            .addComponent(btnConfirmar,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

            .addComponent(btnSalir,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))

    );

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(jPanel2Layout.createSequentialGroup()

        .addContainerGap()

        .addComponent(btnConfirmar,
javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addGap(54, 54, 54)

        .addComponent(btnSalir,
javax.swing.GroupLayout.PREFERRED_SIZE, 45,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))

    );

```

```

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(this);
    this.setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(16, Short.MAX_VALUE))
        );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
            .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
} // </editor-fold>

private void btnConfirmarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int saldo_actual =
base_de_datos.obtener_saldo_tarjeta(txtCodigo.getText()); //obtiene saldo
actual
    int pagar= Integer.parseInt(txtPagar.getText()); //obtiene total a pagar
    int saldo = saldo_actual-pagar; //calcula nuevo saldo
    base_de_datos.insertar_nuevosaldo(txtCodigo.getText(), saldo); //actualizo
base de datos
    MenuPrincipal.esconde_dlg(); //escondemos interfaz
    Factura.esconde_dlg(); //escondemos interfaz
    Factura.btnMail.setEnabled(true);
    com.bandera_factura=1;
    com.bandera_pago=0;
    com.bandera_agregar=0; //estado de banderas cuando se encuentra en
factura

}

ComSerial com = new ComSerial(); //objeto para acceder a comunicacion
serial y contrlar banderas
private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {

```

```

// TODO add your handling code here:
MenuPrincipal.esconde_dlg();
Factura.esconde_dlg();
com.bandera_factura=1;
com.bandera_pago=0;
com.bandera_agregar=0;//estado de banderas cuando se encuentra en
factura
}

static int informacion;
static String nombre;
static String apellido;

public static void inserta_informacion(String codigos){
    String codigo = null;
    if(codigos.length()>12){//cada tag tiene 12 caracteres mas dos espacios en
blanco
        codigo=codigos.substring(1, 13);//se ignora espacios en blanco
        System.out.println("EN EL FRAME DE PAGO");
        System.out.println(codigos);
        System.out.println(codigo);//debug para saber en donde nos
encontramos
        txtCodigo.setText(codigo);//PONE CODIGO EN LA INTERFAZ

        int existe =
base_de_datos.verificar_existencia_tarjeta(codigo);//VERIFICA EXISTENCIA
EN BASE DE DATOS
        if(existe>0){
            informacion = base_de_datos.obtener_saldo_tarjeta(codigo);
            System.out.println("SALDO "+informacion);
            nombre = base_de_datos.obtener_nombre_tarjeta(codigo);

```

```

        System.out.println("NOMBRE "+nombre);
        apellido = base_de_datos.obtener_apellido_tarjeta(codigo);
        System.out.println("APELLIDO "+apellido);
    }else{
        nombre=null;
        JOptionPane.showMessageDialog(null, "NO EXISTE
TARJETA","ERROR",JOptionPane.ERROR_MESSAGE);
    }

    if(nombre!=null){
        txtNombre.setText(nombre+" "+apellido);//ESCRIBO NONBRE EN
INTERFAZ
        String pagar = Factura.txtTotal.getText();//TOMO TOTAL A PAGAR
DE FACTURA
        int pagar_num = Integer.parseInt(pagar);//TRASFORMO DE String a
int
        txtPagar.setText(pagar);//colo total a pagar en interfaz
        if(informacion>pagar_num){//verifico disponibilidad de saldo
            txtDisponible.setText("SALDO DISPONIBLE");
            btnConfirmar.setEnabled(true);//si hay sldo disponible, activo boton
confirmar
        }else {
            txtDisponible.setText("NO SALDO DISPONIBLE");
        }
    }
}
}else{
    txtCodigo.setText("");
    JOptionPane.showMessageDialog(null, "ERROR DE
LECTURA","INTENTE NUEVAMENTE",JOptionPane.ERROR_MESSAGE);
}

```

```
}

public void limpiar(){
    txtCodigo.setText("");
    txtDisponible.setText("");
    txtPagar.setText("");
    txtNombre.setText("");
}

// Variables declaration - do not modify
private static javax.swing.JButton btnConfirmar;
private javax.swing.JButton btnSalir;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JLabel lblAPagar;
private javax.swing.JLabel lblDisponible;
private javax.swing.JLabel lblNombre;
public static javax.swing.JTextField txtCodigo;
private static javax.swing.JTextField txtDisponible;
private static javax.swing.JTextField txtNombre;
private static javax.swing.JTextField txtPagar;
// End of variables declaration
}
```

Interface Portada.java

```
/*
 * Portada.java
 *
```



```
* Creada el 12 de noviembre de 2013
*/

package INTERFACES;

import CONTROLADOR.ComSerial;
import java.awt.event.ActionEvent;

/**
 * @author Paul Molina
 */

public class Portada extends javax.swing.JFrame {

    public Portada() {
        initComponents();
        this.setLocationRelativeTo(this);
        ComSerial Comunicacion = new ComSerial();
        Comunicacion.ConfigSerial("COM3");//se inicia serial solo con este puerto
        Comunicacion.RecepcionSerial();//se activa hilo recepcion
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")

    /**
     // <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```

private void initComponents() {

    jPasswordField1 = new javax.swing.JPasswordField();
    JPanel1 = new javax.swing.JPanel();
    btnsalir = new javax.swing.JButton();
    btnAcceso = new javax.swing.JButton();
    jblespe1 = new javax.swing.JLabel();
    lbllogoespe = new javax.swing.JLabel();
    lbllogoespe1 = new javax.swing.JLabel();
    jblespe2 = new javax.swing.JLabel();

    jPasswordField1.setText("jPasswordField1");

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setTitle("LUCES AUTOMATICAS");

    JPanel1.setBackground(new java.awt.Color(151, 0, 45));
    JPanel1.setToolTipText("ELIJA ALGUNA OPCIÓN EN MENÚ
SUPERIOR");

    btnsalir.setBackground(new java.awt.Color(255, 255, 255));
    btnsalir.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); // NOI18N
    btnsalir.setForeground(new java.awt.Color(151, 0, 45));
    btnsalir.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/salir.jpg"))); //
NOI18N
    btnsalir.setText("SALIR");
    btnsalir.setToolTipText("");
    btnsalir.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {

```

```

        btnsalirActionPerformed(evt);
    }
});

btnAcceso.setBackground(new java.awt.Color(255, 255, 255));
btnAcceso.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); // NOI18N
btnAcceso.setForeground(new java.awt.Color(151, 0, 45));
btnAcceso.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/acceso_inicial.jpg")))
; // NOI18N
btnAcceso.setText("ACCEDER");
btnAcceso.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnAccesoActionPerformed(evt);
    }
});

jblespe1.setFont(new java.awt.Font("Swis721 Ex BT", 0, 28)); // NOI18N
jblespe1.setForeground(new java.awt.Color(255, 255, 255));
jblespe1.setText("FACTURACIÓN RF_ID");

lbllogoespe.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/logo_java.jpg")));
// NOI18N

lbllogoespe1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/logo_udla.jpg")));
// NOI18N

jblespe2.setFont(new java.awt.Font("Swis721 Ex BT", 0, 14)); // NOI18N
jblespe2.setForeground(new java.awt.Color(255, 255, 255));

```

```

jblespe2.setText("Desarrollado por: PAÚL MOLINA LARA");

javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addGap(25, 25, 25)
                    .addComponent(lbllogoespe)
                )
            )
        )
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
    )
    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(lbllogoespe1)
        .addComponent(jblespe1)))
    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1Layout.createSequentialGroup()
            .addGap(78, 78, 78)

```

```

        .addComponent(btnAcceso)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
        javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(btnsalir,
        javax.swing.GroupLayout.PREFERRED_SIZE, 224,
        javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(73, 73, 73))
            .addGap(0, 110, Short.MAX_VALUE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
        jPanel1Layout.createSequentialGroup()
            .addGap(0, 0, Short.MAX_VALUE)
            .addComponent(jblespe2)))
        .addContainerGap()
    );
    jPanel1Layout.setVerticalGroup(

    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
        jPanel1Layout.createSequentialGroup()

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(22, 22, 22)
                .addComponent(lbllogoespe,
        javax.swing.GroupLayout.PREFERRED_SIZE, 132,
        javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(73, 73, 73)

```

```

        .addComponent(jblespe1)))
        .addGap(30, 30, 30)
        .addComponent(lbllogoespe1,
javax.swing.GroupLayout.PREFERRED_SIZE, 132,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(18, 18, 18)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(btnsalir,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(btnAcceso,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addGap(18, 27, Short.MAX_VALUE)
        .addComponent(jblespe2)
        .addContainerGap()
);

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
);
layout.setVerticalGroup(

```

```
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );

    pack();
} // </editor-fold>

*/

private void initComponents() {

    jPasswordField1 = new javax.swing.JPasswordField();
    jPanel1 = new javax.swing.JPanel();
    btnsalir = new javax.swing.JButton();
    btnAcceso = new javax.swing.JButton();
    jblespe1 = new javax.swing.JLabel();
    lbllogoespe = new javax.swing.JLabel();
    lbllogoespe1 = new javax.swing.JLabel();
    jblespe2 = new javax.swing.JLabel();

    jPasswordField1.setText("jPasswordField1");

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setTitle("SISTEMA DE RECONOCIMIENTO DE TAGS");

    jPanel1.setBackground(new java.awt.Color(151, 0, 45));

    btnsalir.setBackground(new java.awt.Color(255, 255, 255));
```

```

    btnsalir.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); // NOI18N
    btnsalir.setForeground(new java.awt.Color(151, 0, 45));
    btnsalir.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/salir.jpg"))); //
NOI18N
    btnsalir.setText("SALIR");
    btnsalir.setToolTipText("");
    btnsalir.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnsalirActionPerformed(evt);
        }
    });

    btnAcceso.setBackground(new java.awt.Color(255, 255, 255));
    btnAcceso.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); // NOI18N
    btnAcceso.setForeground(new java.awt.Color(151, 0, 45));
    btnAcceso.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/acceso_inicial.jpg")))
; // NOI18N
    btnAcceso.setText("ACCEDER");
    btnAcceso.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnAccesoActionPerformed(evt);
        }
    });

    jblespe1.setFont(new java.awt.Font("Swis721 Ex BT", 0, 28)); // NOI18N
    jblespe1.setForeground(new java.awt.Color(255, 255, 255));
    jblespe1.setText("SISTEMA DE LECTURA DE TAGS");

```



```
        lbllogoespe.setIcon(new  
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/logo_java.jpg")));  
// NOI18N
```

```
        lbllogoespe1.setIcon(new  
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/logo_udla.jpg")));  
// NOI18N
```

```
        jblespe2.setFont(new java.awt.Font("Swis721 Ex BT", 0, 14)); // NOI18N  
        jblespe2.setForeground(new java.awt.Color(255, 255, 255));  
        jblespe2.setText("Desarrollado por: PAÚL MOLINA");
```

```
        javax.swing.GroupLayout jPanel1Layout = new  
javax.swing.GroupLayout(jPanel1);  
        jPanel1.setLayout(jPanel1Layout);  
        jPanel1Layout.setHorizontalGroup(  
  
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)  
        .addGroup(jPanel1Layout.createSequentialGroup()  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.TRAILING, false)  
        .addGroup(jPanel1Layout.createSequentialGroup()  
            .addGap(25, 25, 25)  
            .addComponent(lbllogoespe)  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATE  
D)
```

```

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING)
    .addComponent(lbllogoespe1)
    .addComponent(jblespe1)))
.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()
    .addGap(78, 78, 78)
    .addComponent(btnAcceso)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(btnsalir,
javax.swing.GroupLayout.PREFERRED_SIZE, 224,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(73, 73, 73)))
.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
    .addComponent(jblespe2)
    .addGap(20, 20, 20))
);
jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()

```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.LEADING)  
    .addGroup(jPanel1Layout.createSequentialGroup()  
        .addGap(22, 22, 22)  
        .addComponent(lbllogoespe,  
javax.swing.GroupLayout.PREFERRED_SIZE, 132,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
    .addGroup(jPanel1Layout.createSequentialGroup()  
        .addGap(73, 73, 73)  
        .addComponent(jblespe1)))  
.addGap(30, 30, 30)  
.addComponent(lbllogoespe1,  
javax.swing.GroupLayout.PREFERRED_SIZE, 132,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
.addGap(18, 18, 18)  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)  
    .addComponent(btnsalir,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
    .addComponent(btnAcceso,  
javax.swing.GroupLayout.DEFAULT_SIZE,  
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))  
.addGap(18, 27, Short.MAX_VALUE)  
.addComponent(jblespe2)  
.addContainerGap()  
  
);
```

```

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        );

        pack();
    } // </editor-fold>

private void btnsalirActionPerformed(java.awt.event.ActionEvent evt) {
    System.exit(0);
}

private void btnAccesoActionPerformed(java.awt.event.ActionEvent evt) {
    this.setVisible(false);
    Acceso B = new Acceso();
    B.setVisible(true);
    //activa nterfaz para ingresar a programa de facturaciòn
}

```

```

/**
 * @param args the command line arguments
 */

public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Portada.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Portada.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

```

```
java.util.logging.Logger.getLogger(Portada.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(Portada.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
```

```
    }
```

```
//</editor-fold>
```

```
/* Create and display the form */
```

```
java.awt.EventQueue.invokeLater(new Runnable() {
```

```
    public void run() {
```

```
        new Portada().setVisible(true);
```

```
    }
```

```
});
```

```
}
```

```
// Variables declaration - do not modify
```

```
private javax.swing.JButton btnAcceso;
```

```
private javax.swing.JButton btnsalir;
```

```
private javax.swing.JPanel jPanel1;
```

```
private javax.swing.JPasswordField jPasswordField1;
```

```
private javax.swing.JLabel jblespe1;
```

```
private javax.swing.JLabel jblespe2;
```

```
private javax.swing.JLabel lbllogoespe;
```

```
private javax.swing.JLabel lbllogoespe1;
```

```
// End of variables declaration
```

```
}
```

Interface RegistrarCliente.java

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package INTERFACES;

import CONTROLADOR.base_datos;
import java.awt.Color;
import javax.swing.JOptionPane;

/**
 *
 * @author Paúl Molina
 */
public class RegistrarCliente extends javax.swing.JPanel {

    base_datos base_de_datos = new base_datos();

    /**
     * Creates new form RegistrarCliente
     */
    public RegistrarCliente() {
        initComponents();
        base_de_datos.inicio();
        btnAgregar.setEnabled(false);
        btnActualizar.setEnabled(false);
        Factura.dlg_agregar_cliente.setTitle("REGISTRAR CLIENTE");
        txtApellidos.setEnabled(false);
        txtCedula.setEnabled(true);
    }
}
```

```
txtCorreo.setEnabled(false);
txtDireccion.setEnabled(false);
txtNombre.setEnabled(false);
txtTelefono.setEnabled(false);
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    txtDireccion = new javax.swing.JTextField();
    jLabel7 = new javax.swing.JLabel();
    jLabel1 = new javax.swing.JLabel();
    txtApellidos = new javax.swing.JTextField();
    jLabel6 = new javax.swing.JLabel();
    txtCedula = new javax.swing.JTextField();
    txtNombre = new javax.swing.JTextField();
    txtTelefono = new javax.swing.JTextField();
    jLabel4 = new javax.swing.JLabel();
    txtCorreo = new javax.swing.JTextField();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    txtCodigo = new javax.swing.JTextField();
    jLabel5 = new javax.swing.JLabel();
    btnNuevo = new javax.swing.JButton();
```



```
jPanel3 = new javax.swing.JPanel();
btnBuscar = new javax.swing.JButton();
jPanel2 = new javax.swing.JPanel();
jPanel4 = new javax.swing.JPanel();
btnAgregar = new javax.swing.JButton();
jPanel5 = new javax.swing.JPanel();
btnActualizar = new javax.swing.JButton();
jPanel6 = new javax.swing.JPanel();
btnSalir = new javax.swing.JButton();

setBackground(new java.awt.Color(153, 0, 51));

jPanel1.setBackground(new java.awt.Color(153, 0, 51));
jPanel1.setToolTipText("Clic en NUEVO para agregar cliente");

txtDireccion.setToolTipText("Clic en NUEVO para habilitar opciones");

jLabel7.setFont(new java.awt.Font("TI-Nspire", 0, 11)); // NOI18N
jLabel7.setForeground(new java.awt.Color(255, 255, 255));
jLabel7.setText("TELÉFONO");

jLabel1.setFont(new java.awt.Font("TI-Nspire", 0, 11)); // NOI18N
jLabel1.setForeground(new java.awt.Color(255, 255, 255));
jLabel1.setText("CÓDIGO");

txtApellidos.setToolTipText("Clic en NUEVO para habilitar opciones");

jLabel6.setFont(new java.awt.Font("TI-Nspire", 0, 11)); // NOI18N
jLabel6.setForeground(new java.awt.Color(255, 255, 255));
jLabel6.setText("DIRECCIÓN");
```

```
txtCedula.setText("Ingrese Cédula y clic en buscar");
txtCedula.setToolTipText("Clic en NUEVO para habilitar opciones");

txtNombre.setToolTipText("Clic en NUEVO para habilitar opciones");

txtTelefono.setToolTipText("Clic en NUEVO para habilitar opciones");

jLabel4.setFont(new java.awt.Font("TI-Nspire", 0, 11)); // NOI18N
jLabel4.setForeground(new java.awt.Color(255, 255, 255));
jLabel4.setText("NOMBRES");

txtCorreo.setToolTipText("Clic en NUEVO para habilitar opciones");

jLabel2.setFont(new java.awt.Font("TI-Nspire", 0, 11)); // NOI18N
jLabel2.setForeground(new java.awt.Color(255, 255, 255));
jLabel2.setText("CÉDULA");

jLabel3.setFont(new java.awt.Font("TI-Nspire", 0, 11)); // NOI18N
jLabel3.setForeground(new java.awt.Color(255, 255, 255));
jLabel3.setText("APELLIDOS");

txtCodigo.setEditable(false);

jLabel5.setFont(new java.awt.Font("TI-Nspire", 0, 11)); // NOI18N
jLabel5.setForeground(new java.awt.Color(255, 255, 255));
jLabel5.setText("CORREO");

btnNuevo.setBackground(new java.awt.Color(51, 255, 0));
btnNuevo.setText("nuevo");
btnNuevo.setToolTipText("Clic para habilitar opciones");
btnNuevo.addActionListener(new java.awt.event.ActionListener() {
```

```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnNuevoActionPerformed(evt);
        }
    });

    jPanel3.setBackground(new java.awt.Color(153, 0, 51));
    jPanel3.setBorder(javax.swing.BorderFactory.createTitledBorder(null,
"Buscar", javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
javax.swing.border.TitledBorder.DEFAULT_POSITION, new
java.awt.Font("Tahoma", 0, 11), new java.awt.Color(255, 255, 255))); // NOI18N

    btnBuscar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/lupa.jpg"))); //
NOI18N
    btnBuscar.setToolTipText("Búsqueda según Cédula");

    btnBuscar.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing
.border.BevelBorder.RAISED));

    btnBuscar.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
    btnBuscar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnBuscarActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
    jPanel3.setLayout(jPanel3Layout);
    jPanel3Layout.setHorizontalGroup(

```

```
jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel3Layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(btnBuscar,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addContainerGap())
    );
jPanel3Layout.setVerticalGroup(
```

```
jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(btnBuscar,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
    );
```

```
jPanel2.setBackground(new java.awt.Color(255, 255, 153));
```

```
jPanel4.setBackground(new java.awt.Color(255, 255, 153));
```

```
jPanel4.setBorder(javax.swing.BorderFactory.createTitledBorder(null,
"Agregar", javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
javax.swing.border.TitledBorder.DEFAULT_POSITION, new
java.awt.Font("Tahoma", 0, 11), new java.awt.Color(204, 0, 51))); // NOI18N
```

```
    btnAgregar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/agregar_p.jpg"))); //
NOI18N
```

```
btnAgregar.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.RAISED));
```

```
btnAgregar.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);
btnAgregar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnAregarActionPerformed(evt);
    }
});
```

```
javax.swing.GroupLayout jPanel4Layout = new
javax.swing.GroupLayout(jPanel4);
jPanel4.setLayout(jPanel4Layout);
jPanel4Layout.setHorizontalGroup(
```

```
jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel4Layout.createSequentialGroup()
```

```
        .addComponent(btnAregar)
        .addGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
```

```
jPanel4Layout.setVerticalGroup(
```

```
jPanel4Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel4Layout.createSequentialGroup()
```

```
        .addComponent(btnAregar,
javax.swing.GroupLayout.PREFERRED_SIZE, 45, Short.MAX_VALUE)
    );
```

```

jPanel5.setBackground(new java.awt.Color(255, 255, 153));
jPanel5.setBorder(javax.swing.BorderFactory.createTitledBorder(null,
"Actualizar", javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
javax.swing.border.TitledBorder.DEFAULT_POSITION, new
java.awt.Font("Tahoma", 0, 11), new java.awt.Color(204, 51, 0))); // NOI18N

    btnActualizar.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/actualizar.jpg"))); //
NOI18N

btnActualizar.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.sw
ing.border.BevelBorder.RAISED));
    btnActualizar.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            btnActualizarActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout jPanel5Layout = new
javax.swing.GroupLayout(jPanel5);
    jPanel5.setLayout(jPanel5Layout);
    jPanel5Layout.setHorizontalGroup(

jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel5Layout.createSequentialGroup()
        .addComponent(btnActualizar)
        .addGap(0, 0, Short.MAX_VALUE))
    );
jPanel5Layout.setVerticalGroup(

```

```
jPanel5Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(btnActualizar,
javax.swing.GroupLayout.PREFERRED_SIZE, 49, Short.MAX_VALUE)
    );
```

```
jPanel6.setBackground(new java.awt.Color(255, 255, 153));
```

```
jPanel6.setBorder(javax.swing.BorderFactory.createTitledBorder(null,
"Salir", javax.swing.border.TitledBorder.DEFAULT_JUSTIFICATION,
javax.swing.border.TitledBorder.DEFAULT_POSITION, new
java.awt.Font("Tahoma", 0, 11), new java.awt.Color(204, 51, 0))); // NOI18N
```

```
btnSalir.setBackground(new java.awt.Color(255, 255, 153));
```

```
btnSalir.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/salir_2.jpg"))); //
NOI18N
```

```
btnSalir.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.b
order.BevelBorder.RAISED));
```

```
btnSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnSalirActionPerformed(evt);
    }
});
```

```
javax.swing.GroupLayout jPanel6Layout = new
javax.swing.GroupLayout(jPanel6);
jPanel6.setLayout(jPanel6Layout);
jPanel6Layout.setHorizontalGroup(
```

```
jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel6Layout.createSequentialGroup()
        .addComponent(btnSalir, javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addContainerGap())
    );
jPanel6Layout.setVerticalGroup(
```

```
jPanel6Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addComponent(btnSalir,
        javax.swing.GroupLayout.PREFERRED_SIZE, 43,
        javax.swing.GroupLayout.PREFERRED_SIZE)
    );
```

```
    javax.swing.GroupLayout jPanel2Layout = new
        javax.swing.GroupLayout(jPanel2);
    jPanel2.setLayout(jPanel2Layout);
    jPanel2Layout.setHorizontalGroup(
```

```
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addContainerGap()

        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```



```

        .addComponent(jPanel4,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jPanel5,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(jPanel6,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
    );
    jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(jPanel2Layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanel4,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(4, 4, 4)
                .addComponent(jPanel5,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(jPanel6,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .add(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .add(jPanel1Layout.createSequentialGroup()
                    .addComponent(jLabel1)
                    .addComponent(jLabel2,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel3,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel4,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel5,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel6,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```
.addComponent(jLabel7,  
javax.swing.GroupLayout.PREFERRED_SIZE, 77,  
javax.swing.GroupLayout.PREFERRED_SIZE))  
.addGap(25, 25, 25)  
  
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.LEADING)  
.addGroup(jPanel1Layout.createSequentialGroup()  
.addComponent(txtCodigo,  
javax.swing.GroupLayout.PREFERRED_SIZE, 75,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
.addComponent(btnNuevo,  
javax.swing.GroupLayout.PREFERRED_SIZE, 71,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)  
.addComponent(jPanel3,  
javax.swing.GroupLayout.DEFAULT_SIZE,  
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))  
.addComponent(txtCedula)  
.addComponent(txtApellidos)  
.addComponent(txtNombre)  
.addComponent(txtCorreo)  
.addComponent(txtDireccion)  
.addComponent(txtTelefono)  
  
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATE  
D)  
.addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE,  
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```



```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
        .addComponent(txtCodigo,  
javax.swing.GroupLayout.PREFERRED_SIZE, 45,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
        .addComponent(jLabel1))))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
        .addComponent(jLabel2,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
        .addComponent(txtCedula,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
        .addComponent(jLabel3,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)  
        .addComponent(txtApellidos,  
javax.swing.GroupLayout.PREFERRED_SIZE, 33,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(jLabel4,  
        GroupLayout.PREFERRED_SIZE, 23,  
        GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtNombre,  
        GroupLayout.PREFERRED_SIZE, 31,  
        GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(jLabel5,  
        GroupLayout.PREFERRED_SIZE, 23,  
        GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtCorreo,  
        GroupLayout.PREFERRED_SIZE, 31,  
        GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(GroupLayout.Alignment.BASELINE)
```

```
    .addComponent(jLabel6,  
        GroupLayout.PREFERRED_SIZE, 23,  
        GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtDireccion,  
        GroupLayout.PREFERRED_SIZE, 31,  
        GroupLayout.PREFERRED_SIZE))
```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jLabel7,
        javax.swing.GroupLayout.PREFERRED_SIZE, 23,
        javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(txtTelefono,
        javax.swing.GroupLayout.PREFERRED_SIZE, 23,
        javax.swing.GroupLayout.PREFERRED_SIZE))))
    .addContainerGap()
);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(this);
this.setLayout(layout);
layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jPanel1,
            javax.swing.GroupLayout.PREFERRED_SIZE,
            javax.swing.GroupLayout.DEFAULT_SIZE,
            javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(12, Short.MAX_VALUE)
    );
layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()

```

```

        .addContainerGap()
        .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
} // </editor-fold>

```

```

private void btnAgregarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int bandera_insertar;
    int codigo = base_de_datos.obtener_ultimo_codigo();
    codigo=codigo+1;
    String codigo_string = Integer.toString(codigo);
    int longitud = codigo_string.length();
    for(int i=0;i<5-longitud;i++){
        codigo_string="0"+codigo_string;
    }
    System.out.println(codigo_string);

    if("").equals(txtApellidos.getText()){
        txtApellidos.setBackground(Color.RED);
    }else {
        txtApellidos.setBackground(Color.WHITE);
    }
    if("").equals(txtCedula.getText()){
        txtCedula.setBackground(Color.RED);
    }else {
        txtCedula.setBackground(Color.WHITE);
    }
}

```



```

}
if("").equals(txtCorreo.getText()){
    txtCorreo.setBackground(Color.RED);
}else {
    txtCorreo.setBackground(Color.WHITE);
}
if("").equals(txtDireccion.getText()){
    txtDireccion.setBackground(Color.RED);
}else {
    txtDireccion.setBackground(Color.WHITE);
}
if("").equals(txtNombre.getText()){
    txtNombre.setBackground(Color.RED);
}else {
    txtNombre.setBackground(Color.WHITE);
}
if("").equals(txtTelefono.getText()){
    txtTelefono.setBackground(Color.RED);
}else {
    txtTelefono.setBackground(Color.WHITE);
}

if("").equals(txtApellidos.getText()) || "".equals(txtCedula.getText()) ||
    "".equals(txtCorreo.getText()) || "".equals(txtDireccion.getText()) ||
    "".equals(txtNombre.getText()) || "".equals(txtTelefono.getText()){
    bandera_insertar=0;
    JOptionPane.showMessageDialog(this, "Completa parámetros en
rojo","ERROR",JOptionPane.ERROR_MESSAGE);
}else {
    bandera_insertar=1;
}

```

```

        int num_de_coincidencias =
base_de_datos.verificar_existencia_clientes(txtCedula.getText());
        //System.out.println("*****NUM_COINCIDENCIAS
"+num_de_coincidencias);
        if(num_de_coincidencias==0){
            bandera_insertar=1;
        }else{
            bandera_insertar=0;
            JOptionPane.showMessageDialog(this, "CLIENTE
EXISTENTE","ERROR",JOptionPane.ERROR_MESSAGE);
            this.limpiar();
        }

        if(flag==1){
            base_de_datos.insertar_cliente(txtCedula.getText(),
txtNombre.getText(), txtApellidos.getText(), txtCorreo.getText(),
txtDireccion.getText(), txtTelefono.getText(), txtCodigo.getText());
            JOptionPane.showMessageDialog(null, "CLIENTE
AGREGADO","ÉXITO",JOptionPane.INFORMATION_MESSAGE);
            this.limpiar();
            btnAgregar.setEnabled(false);
            btnNuevo.setEnabled(true);
        }
    }

private void btnNuevoActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    btnAgregar.setEnabled(true);
    int codigo = base_de_datos.obtener_ultimo_codigo();

```

```

codigo=codigo+1;
String codigo_string = Integer.toString(codigo);
int longitud = codigo_string.length();
for(int i=0;i<5-longitud;i++){
    codigo_string="0"+codigo_string;
}
System.out.println(codigo_string);
txtCodigo.setText("cl"+codigo_string);
this.limpiar();

```

```

txtApellidos.setEnabled(true);
txtCedula.setEnabled(true);
txtCorreo.setEnabled(true);
txtDireccion.setEnabled(true);
txtNombre.setEnabled(true);
txtTelefono.setEnabled(true);

```

```

}

```

```

private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Factura.esconde_dlg();
}

```

```

private void btnBuscarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Object cliente[][] =
base_de_datos.imprime_clientes_registrar("cedula",txtCedula.getText());
    if(cliente.length>0){
        txtCodigo.setText(cliente[0][0].toString());
        txtApellidos.setText(cliente[0][2].toString());
    }
}

```

```
txtNombre.setText(cliente[0][3].toString());
txtCorreo.setText(cliente[0][4].toString());
txtTelefono.setText(cliente[0][5].toString());
txtDireccion.setText(cliente[0][6].toString());
btnActualizar.setEnabled(true);
```

```
txtApellidos.setBackground(Color.WHITE);
txtCedula.setBackground(Color.WHITE);
txtCodigo.setBackground(Color.WHITE);
txtCorreo.setBackground(Color.WHITE);
txtDireccion.setBackground(Color.WHITE);
txtNombre.setBackground(Color.WHITE);
txtTelefono.setBackground(Color.WHITE);
```

```
txtApellidos.setEnabled(true);
txtCedula.setEnabled(true);
txtCorreo.setEnabled(true);
txtDireccion.setEnabled(true);
txtNombre.setEnabled(true);
txtTelefono.setEnabled(true);
```

```
}else{
    JOptionPane.showMessageDialog(null,"No en base de datos","NO SE
ENCUENTRA CLIENTE",JOptionPane.ERROR_MESSAGE);
}
}
```

```
private void btnActualizarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    base_de_datos.actualizar_cliente(txtCedula.getText(),
txtNombre.getText(), txtApellidos.getText(), txtCorreo.getText(),
txtDireccion.getText(), txtTelefono.getText(), txtCodigo.getText());
```

```
        btnActualizar.setEnabled(false);
        JOptionPane.showMessageDialog(null, "ACTUALIZACIÓN
EXITOSA", "ACTUALIZACIÓN
EXITOSA", JOptionPane.INFORMATION_MESSAGE);
    }
```

```
public void limpiar(){
    txtCedula.setText("");
    txtNombre.setText("");
    txtApellidos.setText("");
    txtDireccion.setText("");
    txtCorreo.setText("");
    txtTelefono.setText("");
}
```

```
// Variables declaration - do not modify
private javax.swing.JButton btnActualizar;
private javax.swing.JButton btnAgregar;
private javax.swing.JButton btnBuscar;
private javax.swing.JButton btnNuevo;
private javax.swing.JButton btnSalir;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JLabel jLabel4;
private javax.swing.JLabel jLabel5;
private javax.swing.JLabel jLabel6;
private javax.swing.JLabel jLabel7;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
```

```

private javax.swing.JPanel jPanel4;
private javax.swing.JPanel jPanel5;
private javax.swing.JPanel jPanel6;
private javax.swing.JTextField txtApellidos;
private javax.swing.JTextField txtCedula;
private javax.swing.JTextField txtCodigo;
private javax.swing.JTextField txtCorreo;
private javax.swing.JTextField txtDireccion;
private javax.swing.JTextField txtNombre;
private javax.swing.JTextField txtTelefono;
// End of variables declaration
}

```

Interface RegistrarProducto.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package INTERFACES;

import CONTROLADOR.ComSerial;
import CONTROLADOR.base_datos;
import java.awt.Color;
import javax.swing.JOptionPane;

/**
 *
 * @author Paúl Molina
 */
public class RegistrarProducto extends javax.swing.JPanel {

```

```
static base_datos base_de_datos = new base_datos();

/**
 * Creates new form RegistrarCliente
 */
public RegistrarProducto() {
    initComponents();
    base_de_datos.inicio();
    btnAgregar.setEnabled(false);
    ComSerial.bandera_agregar=1;
    ComSerial.bandera_factura=0;
    MenuPrincipal.dlg_agregar_producto.setTitle("AGREGAR PRODUCTO");
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    txtPrecio = new javax.swing.JTextField();
    txtProducto = new javax.swing.JTextField();
    txtDetalles = new javax.swing.JTextField();
    lblDetalles = new javax.swing.JLabel();
```

```
lblProducto = new javax.swing.JLabel();
lblPrecio = new javax.swing.JLabel();
txtCodigo = new javax.swing.JTextField();
btnNuevo = new javax.swing.JButton();
jPanel2 = new javax.swing.JPanel();
btnActualizar = new javax.swing.JButton();
btnAgregar = new javax.swing.JButton();
btnSalir = new javax.swing.JButton();

jLabel1.setText("CÓDIGO");

txtPrecio.setToolTipText("Clic en NUEVO para habilitar opciones");

txtProducto.setToolTipText("Clic en NUEVO para habilitar opciones");

txtDetalles.setToolTipText("Clic en NUEVO para habilitar opciones");

lblDetalles.setText("DEALLES");

lblProducto.setText("PRODUCTO");

lblPrecio.setText("PRECIO");

btnNuevo.setText("nuevo");
btnNuevo.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnNuevoActionPerformed(evt);
    }
});
```



```

        javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1)
                .addComponent(lblProducto,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(lblPrecio,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(lblDetalles,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(25, 25, 25)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(txtDetalles,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(txtPrecio,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addComponent(txtProducto,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addComponent(txtCodigo,
javax.swing.GroupLayout.PREFERRED_SIZE, 87,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(btnNuevo,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
        .addContainerGap()
    );
    jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(jPanel1Layout.createSequentialGroup()
            .addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
            .addComponent(jLabel1)
            .addComponent(txtCodigo,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(btnNuevo))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(lblProducto,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtProducto,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(lblPrecio,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtPrecio,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(lblDetalles,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtDetalles,  
javax.swing.GroupLayout.PREFERRED_SIZE, 23,  
javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
    .addContainerGap(32, Short.MAX_VALUE))
```

```
);
```

```
btnActualizar.setText("ACT");
```

```
btnActualizar.addActionListener(new java.awt.event.ActionListener() {
```

```
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
        btnActualizarActionPerformed(evt);
```

```
    }
```

```
});
```

```
btnAgregar.setText("AGRE");
```

```
btnAgregar.setToolTipText("Clic en NUEVO para habilitar opciones");
```

```
btnAgregar.addActionListener(new java.awt.event.ActionListener() {
```

```
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
        btnAgregarActionPerformed(evt);
```

```
    }
```

```
});
```

```
btnSalir.setText("SALIR");
```

```
btnSalir.addActionListener(new java.awt.event.ActionListener() {
```

```
    public void actionPerformed(java.awt.event.ActionEvent evt) {
```

```
        btnSalirActionPerformed(evt);
```

```
    }
```

```
});
```

```
javax.swing.GroupLayout jPanel2Layout = new
```

```
javax.swing.GroupLayout(jPanel2);
```

```
jPanel2.setLayout(jPanel2Layout);
```

```
jPanel2Layout.setHorizontalGroup(
```

```
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .add(jPanel2, javax.swing.GroupLayout.Alignment.LEADING, true)
```

```

        .addGroup(jPanel2Layout.createSequentialGroup())
        .addContainerGap()

        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
            .addComponent(btnAgregar,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(btnActualizar,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(btnSalir,
javax.swing.GroupLayout.DEFAULT_SIZE, 71, Short.MAX_VALUE)))
    );
    jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(btnAgregar,
javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(btnActualizar,
javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

```

```

        .addComponent(btnSalir,
javax.swing.GroupLayout.PREFERRED_SIZE, 45,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(this);
    this.setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(18, 18, 18)
            .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(20, Short.MAX_VALUE))
        );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(20, Short.MAX_VALUE))
        );
    layout.setVerticalGroup(

```

```

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel1,
        javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(jPanel2,
        javax.swing.GroupLayout.PREFERRED_SIZE,
        javax.swing.GroupLayout.DEFAULT_SIZE,
        javax.swing.GroupLayout.PREFERRED_SIZE))
.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
);
} // </editor-fold>

```

```

private void btnAgregarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

    int bandera_insertar;

    if("").equals(txtPrecio.getText()){
        txtPrecio.setBackground(Color.RED);
    }else {
        txtPrecio.setBackground(Color.WHITE);
    }
    if("").equals(txtProducto.getText()){
        txtProducto.setBackground(Color.RED);
    }else {
        txtProducto.setBackground(Color.WHITE);
    }
}

```

```

if("").equals(txtDetalles.getText()){
    txtDetalles.setBackground(Color.RED);
}else {
    txtDetalles.setBackground(Color.WHITE);
}

if("").equals(txtPrecio.getText()) || "").equals(txtProducto.getText()) ||
    "").equals(txtDetalles.getText()) || "").equals(txtCodigo.getText())){
    bandera_insertar=0;
    JOptionPane.showMessageDialog(this, "ERROR","Completa
parámetros en rojo",JOptionPane.ERROR_MESSAGE);
}else {
    bandera_insertar=1;
}

int num_de_coincidencias =
base_de_datos.verificar_existencia_productos(txtCodigo.getText());
//System.out.println("*****NUM_COINCIDENCIAS
"+num_de_coincidencias);
if(num_de_coincidencias==0){
    bandera_insertar=1;
}else{
    bandera_insertar=0;
    JOptionPane.showMessageDialog(this, "PRODUCTO
EXISTENTE","ERROR",JOptionPane.ERROR_MESSAGE);
    this.limpiar();
}

if(bandera_insertar==1){
    base_de_datos.insertar_producto(txtCodigo.getText(),
txtProducto.getText(), txtPrecio.getText(), txtDetalles.getText());

```



```

        btnAgregar.setEnabled(false);
        btnNuevo.setEnabled(true);
    }

}

private void btnNuevoActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    btnAgregar.setEnabled(true);
    //txtCodigo.setText(codigo_base);
    txtProducto.setText("");
    txtDetalles.setText("");
    txtPrecio.setText("");

}

```

ComSerial com = new ComSerial();//objeto para acceder a comunicacion serial y contrlar banderas

```

private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    MenuPrincipal.esconde_dlg();
    com.bandera_factura=1;
    com.bandera_agregar=0;//estado de banderas cuando se encuantra en
factura
}

```

```

private void btnActualizarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

base_de_datos.actualizar_productos(txtCodigo.getText(),txtProducto.getText(),t
xtPrecio.getText(),txtDetalles.getText());

```

```

        JOptionPane.showMessageDialog(null,"PRODUCTO ACTUALIZADO
\n"+txtProducto.getText()+
"$"+txtPrecio.getText(),"ÉXITO",JOptionPane.INFORMATION_MESSAGE);
        this.limpiar();
    }

```

```

static Object productos[][];
public static void inserta_producto(String codigos){
    String codigo = null;
    if(codigos.length()>12){
        codigo=codigos.substring(1, 13);
        System.out.println("EN EL FRAME");
        System.out.println(codigos);
        System.out.println(codigo);
        txtCodigo.setText(codigo);
        productos = base_de_datos.imprime_productos_actualizar(codigo);

        if(productos!=null){
            txtProducto.setText(productos[0][1].toString());
            txtPrecio.setText(productos[0][2].toString());
            txtDetalles.setText(productos[0][3].toString());
        }
    }else{
        txtCodigo.setText("");
        JOptionPane.showMessageDialog(null, "ERROR DE
LECTURA","INTENTE NUEVAMENTE",JOptionPane.ERROR_MESSAGE);
    }

}

public void limpiar(){

```

```

        txtCodigo.setText("");
        txtDetalles.setText("");
        txtPrecio.setText("");
        txtProducto.setText("");
    }

    // Variables declaration - do not modify
    private javax.swing.JButton btnActualizar;
    private javax.swing.JButton btnAgregar;
    private javax.swing.JButton btnNuevo;
    private javax.swing.JButton btnSalir;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
    private javax.swing.JLabel lblDetalles;
    private javax.swing.JLabel lblPrecio;
    private javax.swing.JLabel lblProducto;
    public static javax.swing.JTextField txtCodigo;
    private static javax.swing.JTextField txtDetalles;
    private static javax.swing.JTextField txtPrecio;
    private static javax.swing.JTextField txtProducto;
    // End of variables declaration
}

```

Interface RegistrarUsuario.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package INTERFACES;

```

```
import CONTROLADOR.ComSerial;
import CONTROLADOR.base_datos;
import java.awt.Color;
import javax.swing.JOptionPane;

/**
 *
 * @author Paúl Molina
 */
public class RegistrarProducto extends javax.swing.JPanel {

    static base_datos base_de_datos = new base_datos();

    /**
     * Creates new form RegistrarCliente
     */
    public RegistrarProducto() {
        initComponents();
        base_de_datos.inicio();
        btnAgregar.setEnabled(false);
        ComSerial.bandera_agregar=1;
        ComSerial.bandera_factura=0;
        MenuPrincipal.dlg_agregar_producto.setTitle("AGREGAR PRODUCTO");
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
}
```

```
*/  
@SuppressWarnings("unchecked")  
// <editor-fold defaultstate="collapsed" desc="Generated Code">  
private void initComponents() {  
  
    jPanel1 = new javax.swing.JPanel();  
    jLabel1 = new javax.swing.JLabel();  
    txtPrecio = new javax.swing.JTextField();  
    txtProducto = new javax.swing.JTextField();  
    txtDetalles = new javax.swing.JTextField();  
    lblDetalles = new javax.swing.JLabel();  
    lblProducto = new javax.swing.JLabel();  
    lblPrecio = new javax.swing.JLabel();  
    txtCodigo = new javax.swing.JTextField();  
    btnNuevo = new javax.swing.JButton();  
    jPanel2 = new javax.swing.JPanel();  
    btnActualizar = new javax.swing.JButton();  
    btnAgregar = new javax.swing.JButton();  
    btnSalir = new javax.swing.JButton();  
  
    jLabel1.setText("CÓDIGO");  
  
    txtPrecio.setToolTipText("Clic en NUEVO para habilitar opciones");  
  
    txtProducto.setToolTipText("Clic en NUEVO para habilitar opciones");  
  
    txtDetalles.setToolTipText("Clic en NUEVO para habilitar opciones");  
  
    lblDetalles.setText("DEALLES");  
  
    lblProducto.setText("PRODUCTO");
```

```
lblPrecio.setText("PRECIO");
```

```
btnNuevo.setText("nuevo");
```

```
btnNuevo.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnNuevoActionPerformed(evt);
    }
});
```

```
    javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(15, 15, 15)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jLabel1)
            .addComponent(lblProducto,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(lblPrecio,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE)
```

```

        .addComponent(lblDetalles,
javax.swing.GroupLayout.PREFERRED_SIZE, 77,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(25, 25, 25)

        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.LEADING, false)
            .addComponent(txtDetalles,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(txtPrecio,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(txtProducto,
javax.swing.GroupLayout.PREFERRED_SIZE, 219,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addComponent(txtCodigo,
javax.swing.GroupLayout.PREFERRED_SIZE, 87,
javax.swing.GroupLayout.PREFERRED_SIZE)

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(btnNuevo,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)))
            .addContainerGap())
    );
    jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(jPanel1Layout.createSequentialGroup()

```

```
.addContainerGap()
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(jLabel1)
```

```
    .addComponent(txtCodigo,
```

```
    javax.swing.GroupLayout.PREFERRED_SIZE,
```

```
    javax.swing.GroupLayout.DEFAULT_SIZE,
```

```
    javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(btnNuevo))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATE  
D)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(lblProducto,
```

```
    javax.swing.GroupLayout.PREFERRED_SIZE, 23,
```

```
    javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addComponent(txtProducto,
```

```
    javax.swing.GroupLayout.PREFERRED_SIZE, 23,
```

```
    javax.swing.GroupLayout.PREFERRED_SIZE))
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
```

```
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align  
ment.BASELINE)
```

```
    .addComponent(lblPrecio,
```

```
    javax.swing.GroupLayout.PREFERRED_SIZE, 23,
```

```
    javax.swing.GroupLayout.PREFERRED_SIZE)
```



```

        .addComponent(txtPrecio,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Align
ment.BASELINE)
        .addComponent(lblDetalles,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(txtDetalles,
javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(32, Short.MAX_VALUE)
);

btnActualizar.setText("ACT");
btnActualizar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnActualizarActionPerformed(evt);
    }
});

btnAgregar.setText("AGRE");
btnAgregar.setToolTipText("Clic en NUEVO para habilitar opciones");
btnAgregar.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnAgregarActionPerformed(evt);
    }
});

```

```

btnSalir.setText("SALIR");
btnSalir.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        btnSalirActionPerformed(evt);
    }
});

javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
jPanel2.setLayout(jPanel2Layout);
jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addGap(15, 15, 15)
        .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
            .addComponent(btnAgregar,
                javax.swing.GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(btnActualizar,
                javax.swing.GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(btnSalir,
                javax.swing.GroupLayout.DEFAULT_SIZE, 71, Short.MAX_VALUE))
        .addContainerGap(15, true));
jPanel2Layout.setVerticalGroup(

```

```
jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(btnAgregar,
javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(btnActualizar,
javax.swing.GroupLayout.PREFERRED_SIZE, 42,
javax.swing.GroupLayout.PREFERRED_SIZE)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(btnSalir,
javax.swing.GroupLayout.PREFERRED_SIZE, 45,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
```

```
    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(this);
    this.setLayout(layout);
    layout.setHorizontalGroup(
```

```
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
```

```

javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGap(18, 18, 18)
    .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(20, Short.MAX_VALUE))
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEA
DING)
        .addComponent(jPanel1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
} // </editor-fold>

private void btnAgregarActionPerformed(java.awt.event.ActionEvent evt) {

```

```

// TODO add your handling code here:

int bandera_insertar;

if("").equals(txtPrecio.getText()){
    txtPrecio.setBackground(Color.RED);
}else {
    txtPrecio.setBackground(Color.WHITE);
}
if("").equals(txtProducto.getText()){
    txtProducto.setBackground(Color.RED);
}else {
    txtProducto.setBackground(Color.WHITE);
}
if("").equals(txtDetalles.getText()){
    txtDetalles.setBackground(Color.RED);
}else {
    txtDetalles.setBackground(Color.WHITE);
}

if("").equals(txtPrecio.getText()) || "".equals(txtProducto.getText()) ||
"".equals(txtDetalles.getText()) || "".equals(txtCodigo.getText())){
    bandera_insertar=0;
    JOptionPane.showMessageDialog(this, "ERROR","Completa
parámetros en rojo",JOptionPane.ERROR_MESSAGE);
}else {
    bandera_insertar=1;
}

int num_de_coincidencias =
base_de_datos.verificar_existencia_productos(txtCodigo.getText());

```

```

        //System.out.println("*****NUM_COINCIDENCIAS
"+num_de_coincidencias);
        if(num_de_coincidencias==0){
            bandera_insertar=1;
        }else{
            bandera_insertar=0;
            JOptionPane.showMessageDialog(this, "PRODUCTO
EXISTENTE","ERROR",JOptionPane.ERROR_MESSAGE);
            this.limpiar();
        }

        if(bandera_insertar==1){
            base_de_datos.insertar_producto(txtCodigo.getText(),
txtProducto.getText(), txtPrecio.getText(), txtDetalles.getText());
            btnAgregar.setEnabled(false);
            btnNuevo.setEnabled(true);
        }

    }

private void btnNuevoActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    btnAgregar.setEnabled(true);
    //txtCodigo.setText(codigo_base);
    txtProducto.setText("");
    txtDetalles.setText("");
    txtPrecio.setText("");
}

```

ComSerial com = new ComSerial();//objeto para acceder a comunicacion serial y contrlar banderas

```
private void btnSalirActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    MenuPrincipal.esconde_dlg();
    com.bandera_factura=1;
    com.bandera_agregar=0;//estado de banderas cuando se encuantra en
factura
}
```

```
private void btnActualizarActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:

base_de_datos.actualizar_productos(txtCodigo.getText(),txtProducto.getText(),t
xtPrecio.getText(),txtDetalles.getText());
    JOptionPane.showMessageDialog(null,"PRODUCTO ACTUALIZADO
\n"+txtProducto.getText()+
"$"+txtPrecio.getText(),"ÉXITO",JOptionPane.INFORMATION_MESSAGE);
    this.limpiar();
}
```

```
static Object productos[][];
public static void inserta_producto(String codigos){
    String codigo = null;
    if(codigos.length()>12){
        codigo=codigos.substring(1, 13);
        System.out.println("EN EL FRAME");
        System.out.println(codigos);
        System.out.println(codigo);
        txtCodigo.setText(codigo);
        productos = base_de_datos.imprime_productos_actualizar(codigo);
    }
}
```

```
        if(productos!=null){
            txtProducto.setText(productos[0][1].toString());
            txtPrecio.setText(productos[0][2].toString());
            txtDetalles.setText(productos[0][3].toString());
        }
    }else{
        txtCodigo.setText("");
        JOptionPane.showMessageDialog(null, "ERROR DE
LECTURA", "INTENTE NUEVAMENTE", JOptionPane.ERROR_MESSAGE);
    }
}

public void limpiar(){
    txtCodigo.setText("");
    txtDetalles.setText("");
    txtPrecio.setText("");
    txtProducto.setText("");
}

// Variables declaration - do not modify
private javax.swing.JButton btnActualizar;
private javax.swing.JButton btnAgregar;
private javax.swing.JButton btnNuevo;
private javax.swing.JButton btnSalir;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JLabel lblDetalles;
private javax.swing.JLabel lblPrecio;
```



```

private javax.swing.JLabel lblProducto;
public static javax.swing.JTextField txtCodigo;
private static javax.swing.JTextField txtDetalles;
private static javax.swing.JTextField txtPrecio;
private static javax.swing.JTextField txtProducto;
// End of variables declaration
}

```

Interface Usuario.java

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

/*
 * Usuario.java
 *
 * Creada el 12 de noviembre de 2013
 */
package INTERFACES;

/**
 *
 * @author Paul Molina
 */

public class Usuario extends javax.swing.JFrame {
    MenuPrincipal menu=new MenuPrincipal();
    /** Creates new form Usuario */
    public Usuario() {

```

```
    initComponents();
    this.setLocationRelativeTo(null);
}

/** This method is called from within the constructor to
 * initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is
 * always regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    btnMenu = new javax.swing.JButton();
    jLabel1 = new javax.swing.JLabel();
    lblsam = new javax.swing.JLabel();
    lblBienvenido = new javax.swing.JLabel();
    lblsam2 = new javax.swing.JLabel();
    jSeparator4 = new javax.swing.JSeparator();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    jPanel1.setBackground(new java.awt.Color(151, 0, 45));

    btnMenu.setBackground(new java.awt.Color(255, 255, 255));
    btnMenu.setFont(new java.awt.Font("Swis721 Ex BT", 0, 18)); // NOI18N
    btnMenu.setForeground(new java.awt.Color(151, 0, 45));
```

```

        btnMenu.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/ICONOS/menu_principal.jpg")
)); // NOI18N
        btnMenu.setText("MENU");
        btnMenu.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                btnMenuActionPerformed(evt);
            }
        });

```

```

        jLabel1.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/IMAGENES/logo_udla.jpg")));
// NOI18N

```

```

        lblsam.setFont(new java.awt.Font("Swis721 Ex BT", 0, 36)); // NOI18N
        lblsam.setForeground(new java.awt.Color(255, 255, 255));
        lblsam.setText("Usuario");

```

```

        lblBienvenido.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); //
NOI18N
        lblBienvenido.setForeground(new java.awt.Color(255, 255, 255));
        lblBienvenido.setText("Bienvenido...");

```

```

        lblsam2.setFont(new java.awt.Font("Swis721 Ex BT", 0, 24)); // NOI18N
        lblsam2.setForeground(new java.awt.Color(255, 255, 255));
        lblsam2.setText("Tipo de Acceso: Restringido");

```

```

        javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(

```

```
jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addContainerGap()
```

```
    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addComponent(jLabel1,
                javax.swing.GroupLayout.PREFERRED_SIZE, 234,
                javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
        .addGroup(jPanel1Layout.createSequentialGroup()
```

```
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
        javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
```

```
    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
```

```
        .addComponent(btnMenu)
        .addComponent(lblsam)
        .addGap(28, 28, 28))
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(18, 18, 18)
        .addComponent(jSeparator4,
            javax.swing.GroupLayout.PREFERRED_SIZE, 269,
            javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(30, Short.MAX_VALUE))))
```

```

        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
        .addGap(0, 0, Short.MAX_VALUE)
        .addComponent(lblsam2)
        .addGap(89, 89, 89)))
    .addGroup(jPanel1Layout.createSequentialGroup()
        .addGap(40, 40, 40)
        .addComponent(lblBienvenido,
javax.swing.GroupLayout.PREFERRED_SIZE, 474,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE))
    );
jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

    .addGroup(jPanel1Layout.createSequentialGroup()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(41, 41, 41)
            .addComponent(lblsam,
javax.swing.GroupLayout.PREFERRED_SIZE, 44,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jSeparator4,
javax.swing.GroupLayout.PREFERRED_SIZE, 10,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(jPanel1Layout.createSequentialGroup()

```

```

        .addContainerGap()
        .addComponent(jLabel1,
javax.swing.GroupLayout.PREFERRED_SIZE, 136,
javax.swing.GroupLayout.PREFERRED_SIZE)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATE
D)

        .addComponent(lblBienvenido,
javax.swing.GroupLayout.PREFERRED_SIZE, 44,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(lblsam2,
javax.swing.GroupLayout.PREFERRED_SIZE, 44,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATE
D)

        .addComponent(btnMenu)
        .addContainerGap(22, Short.MAX_VALUE)
    );

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    );

```

```

        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addComponent(jPanel1,
                javax.swing.GroupLayout.PREFERRED_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE,
                javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(0, 0, Short.MAX_VALUE))
        );

        pack();
    }// </editor-fold>

private void btnMenuActionPerformed(java.awt.event.ActionEvent evt) {
// TODO add your handling code here:
    menu.setVisible(true);
    this.setVisible(false);
}

public void setear(String bandera_tipo, String vendedor){
    menu.bandera_tipo_usuario(bandera_tipo);//configura interfaz para usu de
    Usuario
    Factura.SetearVendedor(vendedor);//coloca nombre de vendedor en onterfaz
    lblBienvenido.setText("Bienvenido "+vendedor);//bienvenido mas nombre de
    vendedor en interfaz de inicio
}

/**
 * @param args the command line arguments

```

```

*/
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code
(optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default
look and feel.
    * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
    */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Usuario.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Usuario.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Usuario.class.getName()).log(java.util.loggin
g.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {

```



```
java.util.logging.Logger.getLogger(Usuario.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {
            new Usuario().setVisible(true);
        }
    });
}
// Variables declaration - do not modify
private javax.swing.JButton btnMenu;
private javax.swing.JLabel jLabel1;
private javax.swing.JPanel jPanel1;
private javax.swing.JSeparator jSeparator4;
private javax.swing.JLabel lblBienvenido;
private javax.swing.JLabel lblsam;
private javax.swing.JLabel lblsam2;
// End of variables declaration
}
```