

Actividad 2. Programa Banco Mexicano (parte 1)

Lenguajes de Programación IV

Ingeniería en Desarrollo de Software

Tutor: Aarón Iván Salazar Macías

Alumno: José Luis Pacheco González

Fecha: 24 de marzo 2024

Índice

Introducción	3
Descripción	4
Justificación	5
Desarrollo	6
• Interfaz	6
• Codificación	10
Conclusión	24
Referencias	25

Introducción

En esta etapa del proyecto "Programa Banco Mexicano" se desarrolla la primera parte, la cual implica la creación de diseños de interfaces y funciones del menú, junto con el establecimiento de la base de datos en MySQL Workbench. Esta etapa inicial es fundamental, ya que sienta las bases para el desarrollo posterior del sistema bancario.

En primer lugar, la elaboración de los diseños de las interfaces y las funciones del menú requiere un enfoque meticuloso y centrado en el usuario. Se deben tener en cuenta las necesidades y expectativas, así como la eficiencia operativa del sistema. Cada elemento del diseño debe ser intuitivo y fácil de navegar, proporcionando una experiencia fluida y satisfactoria para todos los usuarios.

Por otro lado, la creación de la base de datos en MySQL Workbench es un paso crítico para garantizar la integridad y la seguridad de los datos del banco. Se deben diseñar tablas adecuadas para almacenar información. Además, se deben implementar relaciones y restricciones para mantener la coherencia y la precisión de los datos.

Descripción

Se requiere desarrollar un programa en Java 8 para Banco Mexicano que permita a sus clientes realizar operaciones básicas como depósitos, retiros y consultas de saldo. La pantalla principal del programa debe presentar un menú con cuatro opciones: Depósito, Retiro, Saldo y Salir.

Si el usuario elige la opción de Depósito, el programa debe capturar la cantidad a depositar ingresada por el usuario. Luego, preguntará si desea realizar otro depósito. Si la respuesta es "Sí", volverá a la pantalla de Depósito; si es "No", regresará al menú principal.

En caso de elegir la opción de Retiro, el programa solicitará al usuario la cantidad a retirar. Después de realizar el retiro, volverá automáticamente al menú principal.

La base de datos aún no se menciona específicamente en la instrucción, pero se sugiere que el programa estará conectado a una base de datos que gestionará las transacciones realizadas por los clientes. El objetivo es crear una interfaz interactiva que facilite las operaciones bancarias básicas para los clientes del Banco Mexicano.

Justificación

La creación de diseños de interfaces y funciones de menú en NetBeans, junto con el desarrollo de la base de datos en MySQL Workbench, es fundamental para el éxito de cualquier proyecto de desarrollo de software. Estas actividades son vitales porque establecen la estructura y la experiencia de usuario de la aplicación.

En primer lugar, los diseños de interfaces bien pensados y las funciones de menú intuitivas mejoran significativamente la usabilidad de la aplicación. Una interfaz limpia y bien diseñada facilita la navegación y la comprensión de las características de la aplicación, lo que conduce a una mayor satisfacción del usuario y una menor tasa de abandono.

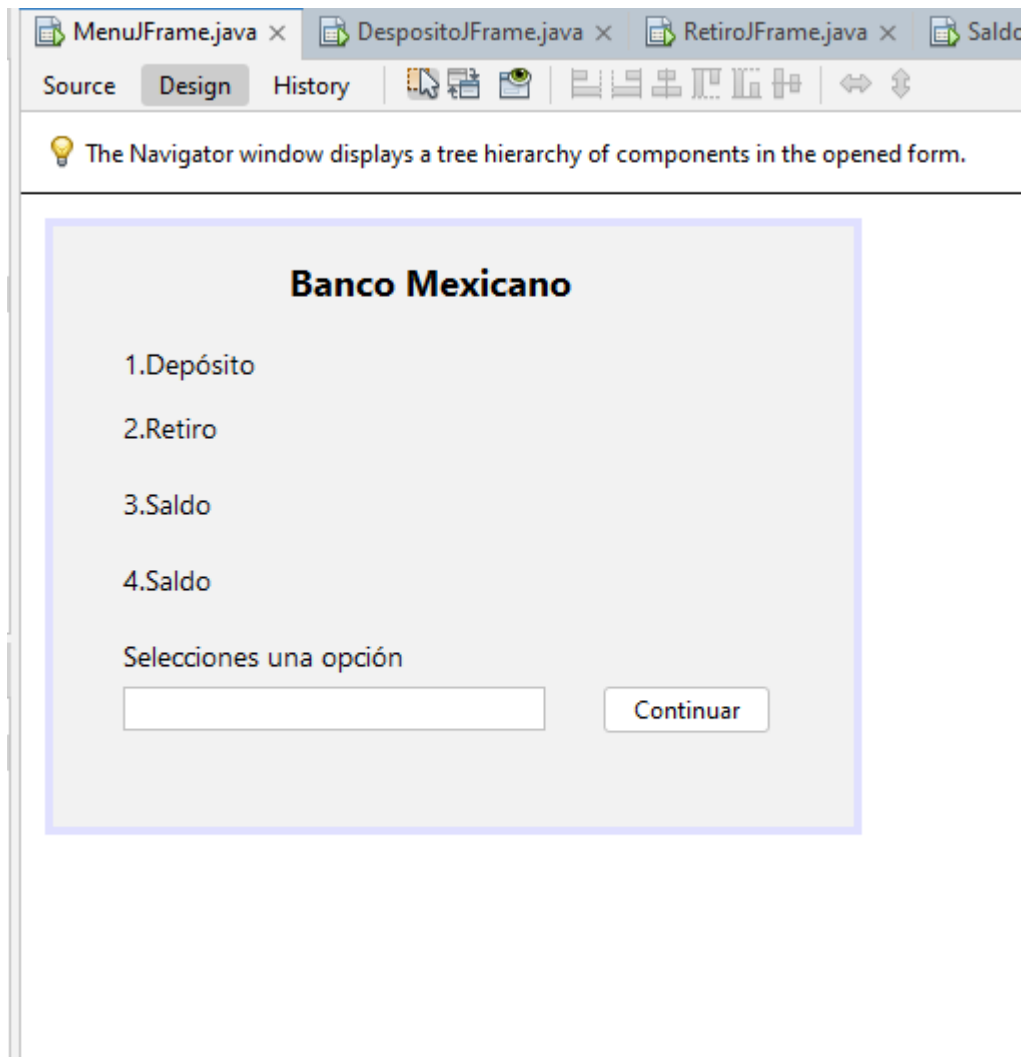
Además, la creación de la base de datos en MySQL Workbench es esencial para gestionar eficientemente los datos de la aplicación. Una base de datos bien diseñada garantiza la integridad, la seguridad y el rendimiento de los datos, lo que se traduce en una aplicación más robusta y confiable.

En resumen, la importancia de crear los diseños de interfaces y funciones del menú en NetBeans, así como la creación de la base de datos en MySQL Workbench, radica en su impacto directo en la experiencia del usuario y en la gestión eficiente de los datos. Estas actividades sientan las bases para el desarrollo exitoso de aplicaciones de software de calidad.

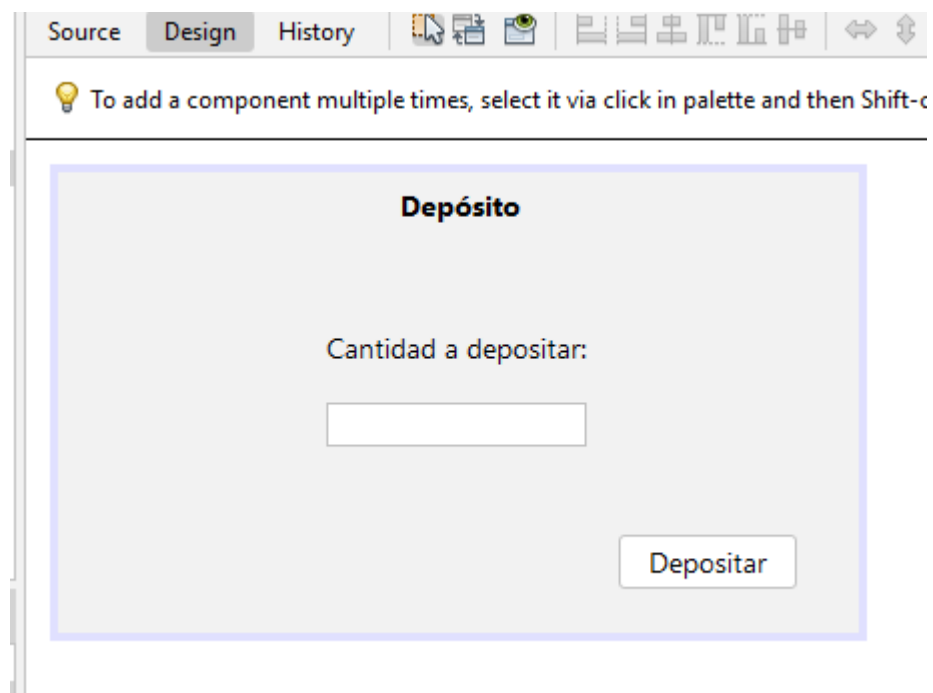
Desarrollo

- **Interfaz**

Como primer paso se requiere diseñar las formas requeridas, empezando por el menú.



A continuación, se muestra el form correspondiente al depósito.



The image shows a screenshot of a web form titled "Depósito" (Deposit) within a design tool interface. The interface includes tabs for "Source", "Design", and "History", with "Design" currently selected. A toolbar with various icons is visible above the form. A tooltip message reads: "To add a component multiple times, select it via click in palette and then Shift-c". The form itself is a light gray rectangle with a purple border. It contains the title "Depósito" at the top, followed by the label "Cantidad a depositar:" (Amount to deposit:). Below the label is a white text input field. At the bottom right of the form is a button labeled "Depositar" (Deposit).

Source Design History

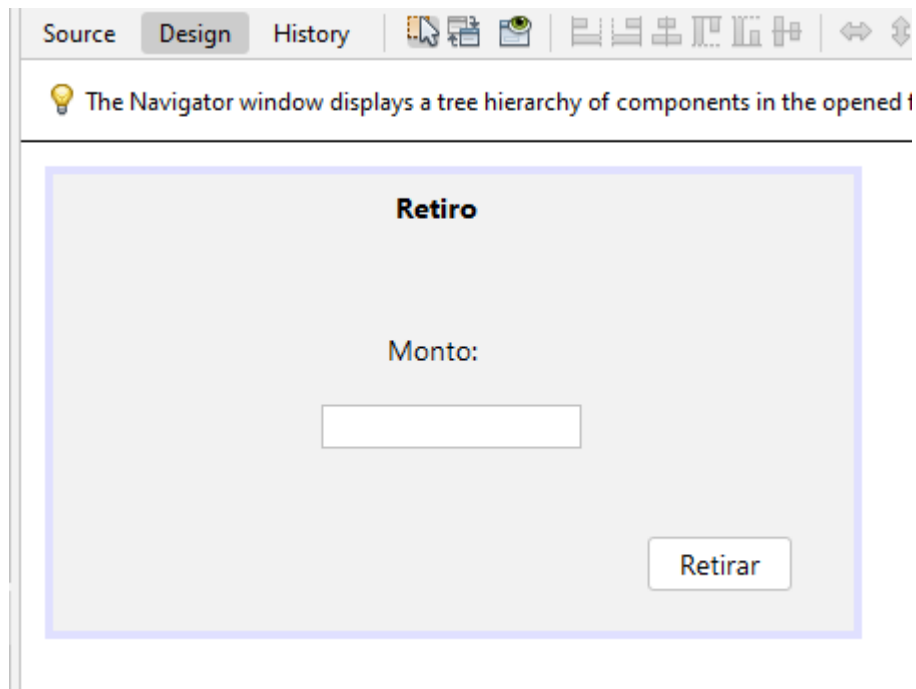
To add a component multiple times, select it via click in palette and then Shift-c

Depósito

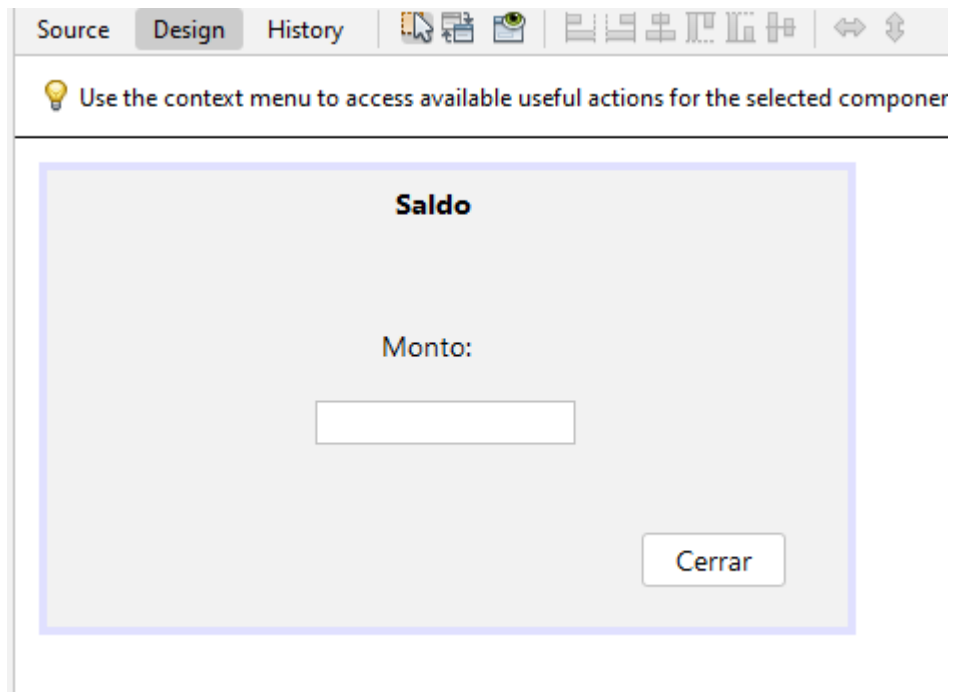
Cantidad a depositar:

Depositar

Se agregó también el form para realiza retiros.



Por último, se agrega el form de saldo.



The image shows a screenshot of a software development environment, likely an IDE, with a 'Design' tab selected. The interface includes a toolbar with various icons for design and development. Below the toolbar, a message reads: "Use the context menu to access available useful actions for the selected componer". The main area displays a form titled "Saldo" with a label "Monto:" and a text input field. A "Cerrar" button is located at the bottom right of the form.

Source Design History

Use the context menu to access available useful actions for the selected componer

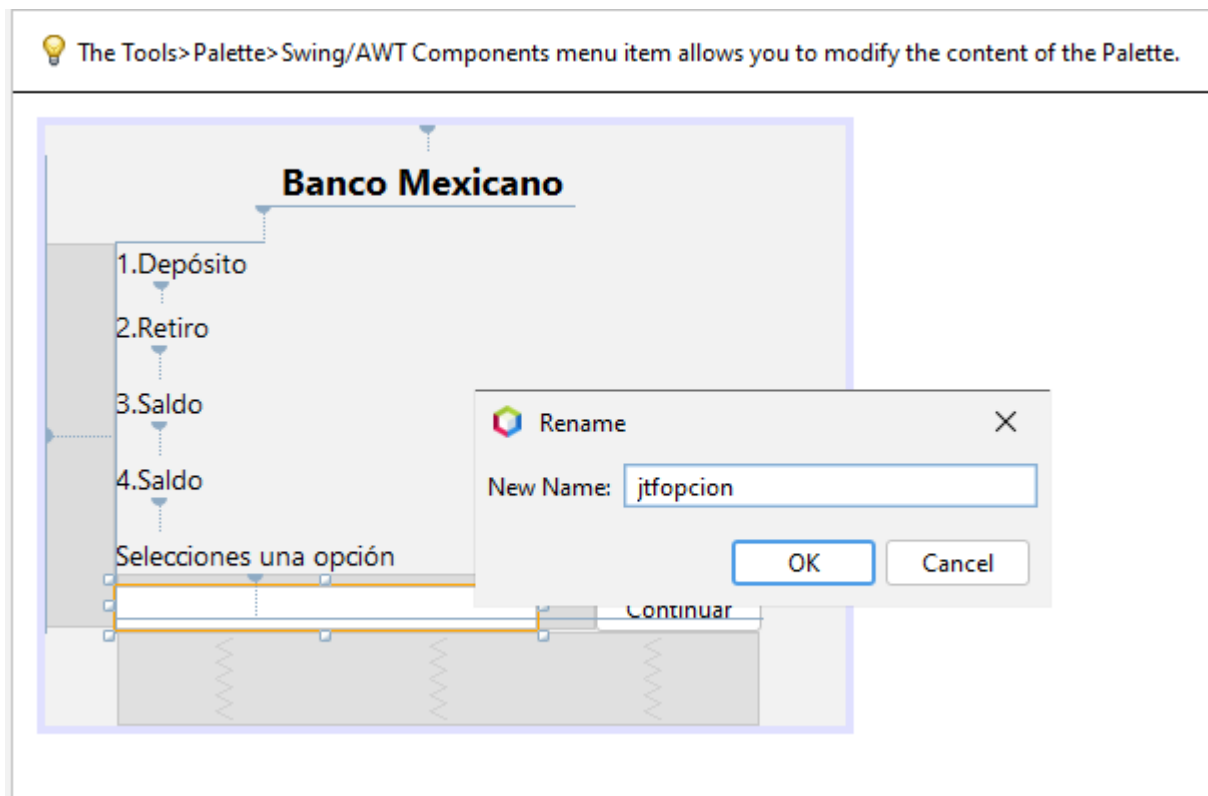
Saldo


Monto:

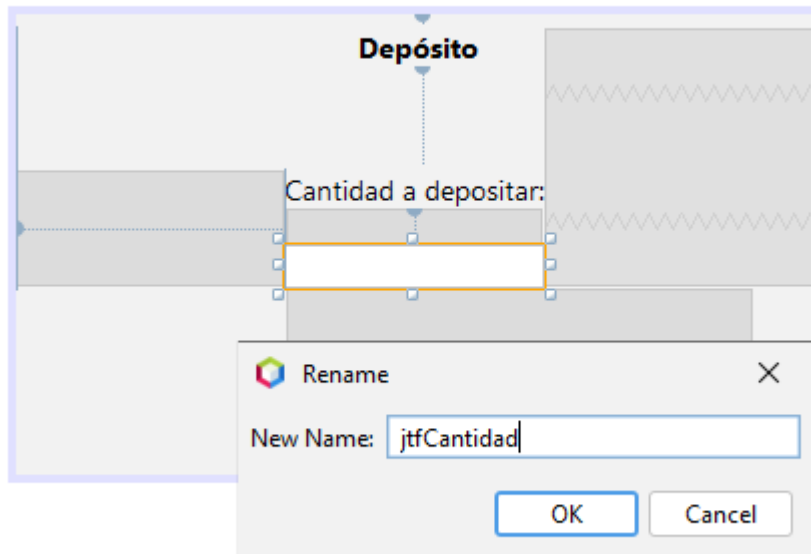
Cerrar

- **Codificación**

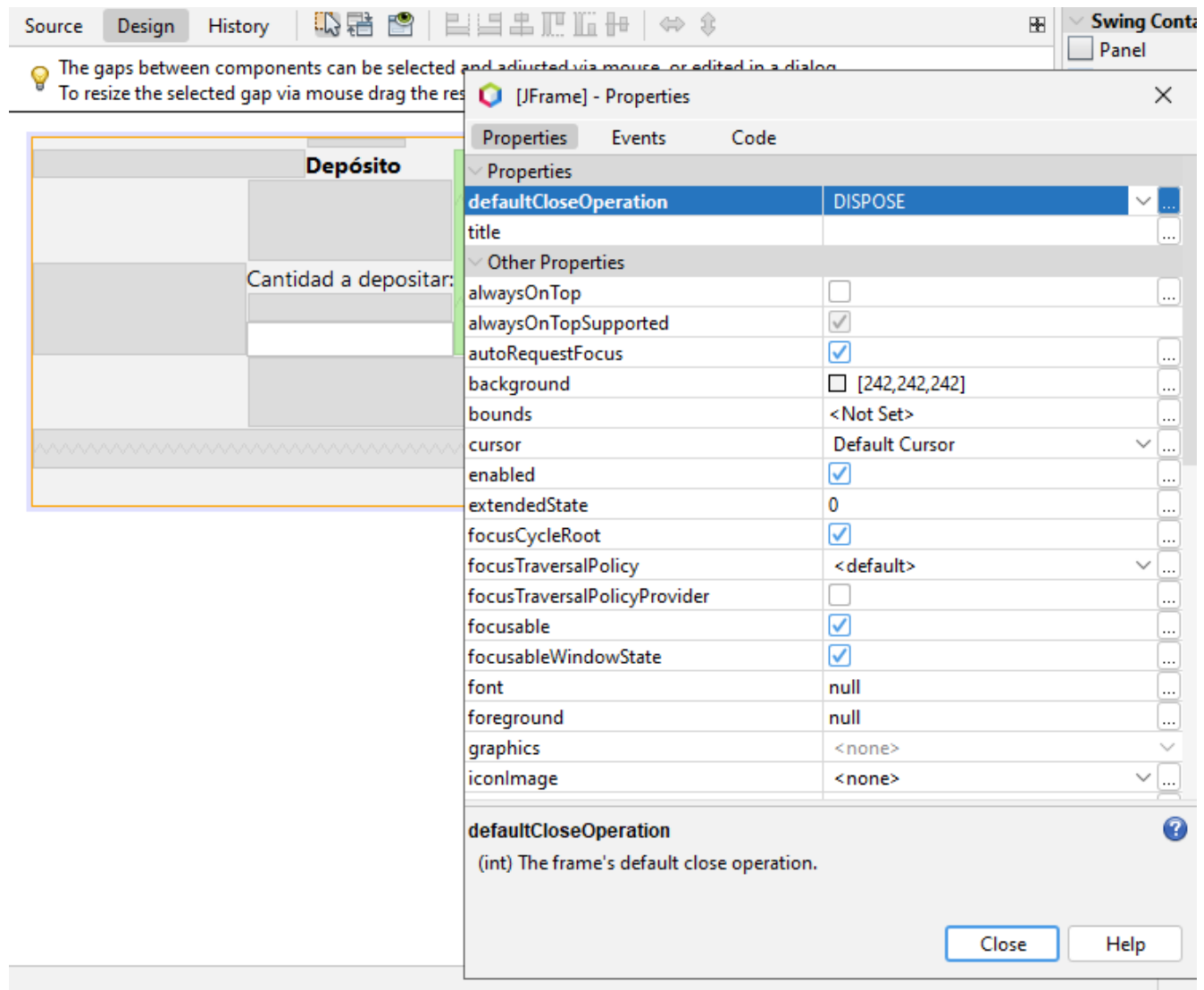
Antes de iniciar a realizar la codificación, es necesario cambiar el nombre de todos los elementos integrados en los forms, ya sean etiquetas, botones o cajas de texto.



 The Navigator window displays a tree hierarchy of components in the opened form.



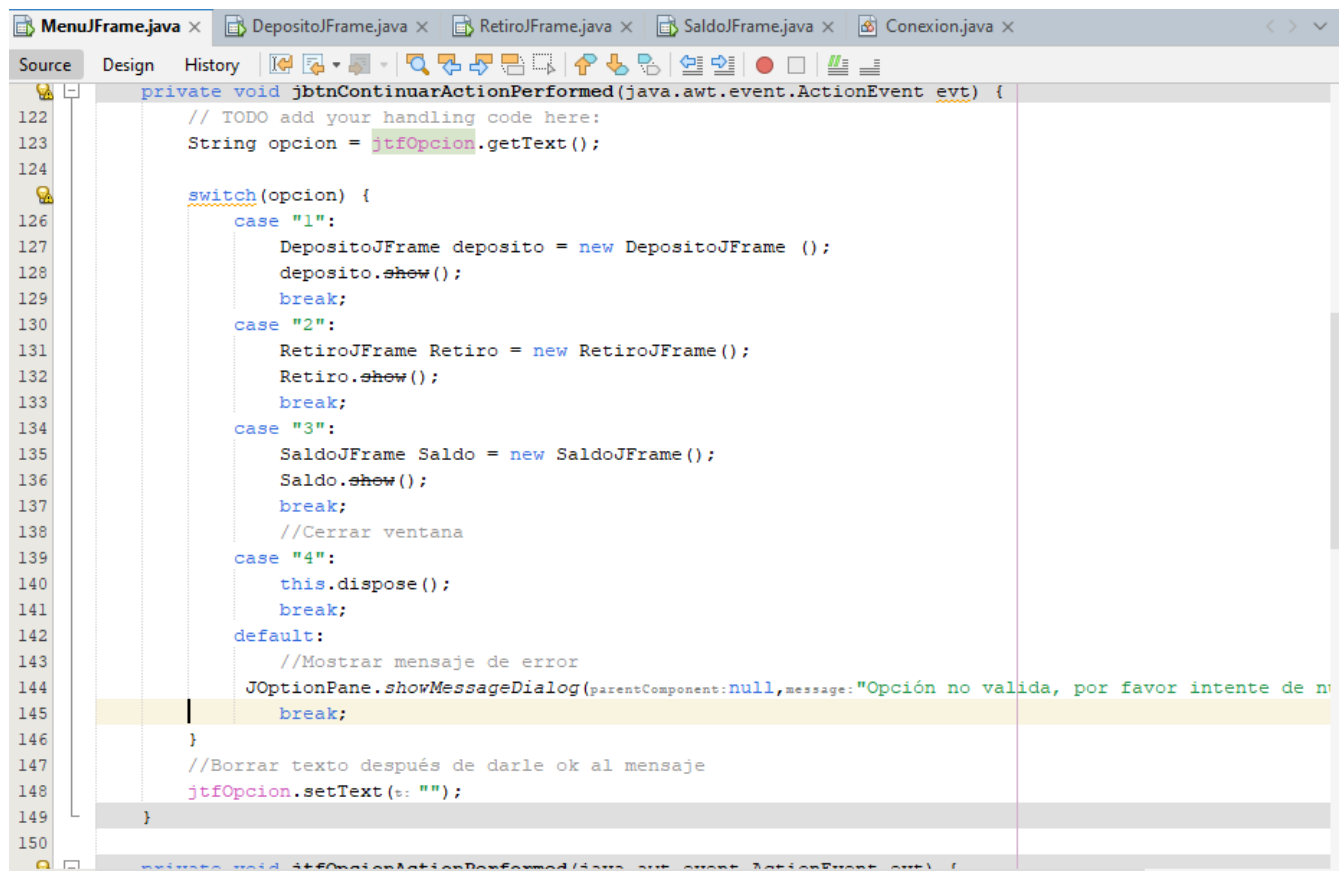
Al seleccionar esta opción Dispose se evita que al ejecutarse el programa las ventanas se cierren, esto se debe hacer con los forms de depósito, retiro y saldo.



En esta captura se muestra el código necesario para poder ejecutar las funciones necesarias desde el menú, dichas funciones son las siguientes:

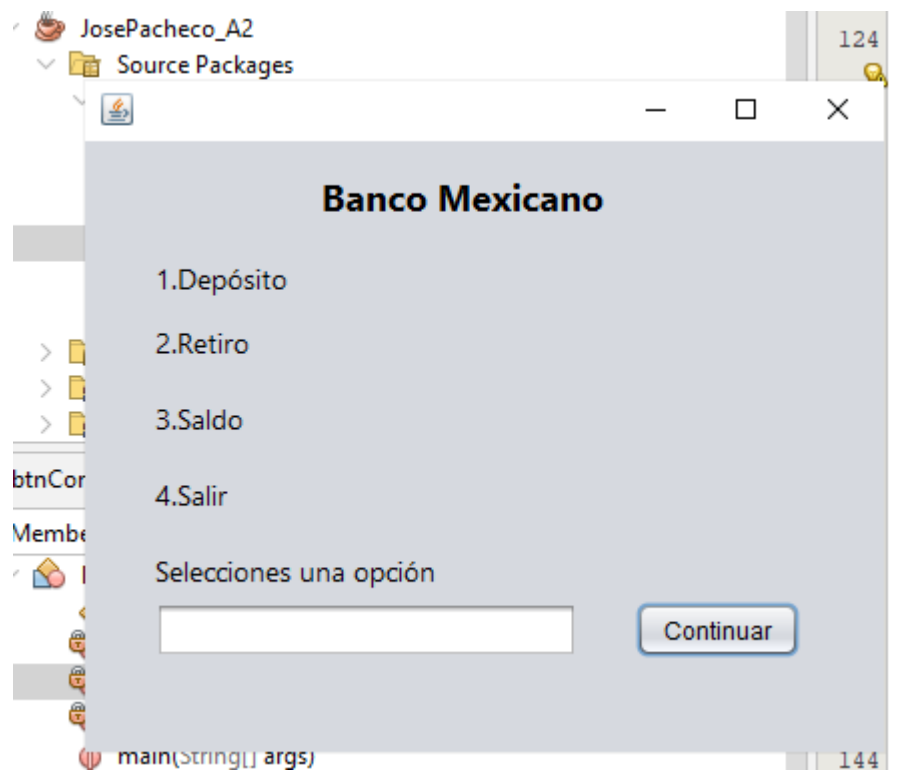
1. Depósito
2. Retiro
3. Saldo
4. Salir

En la captura se muestra el código para ejecutar cada función en el orden solicitado, así como un mensaje error que se muestra en caso de ingresar un número o carácter erróneo.

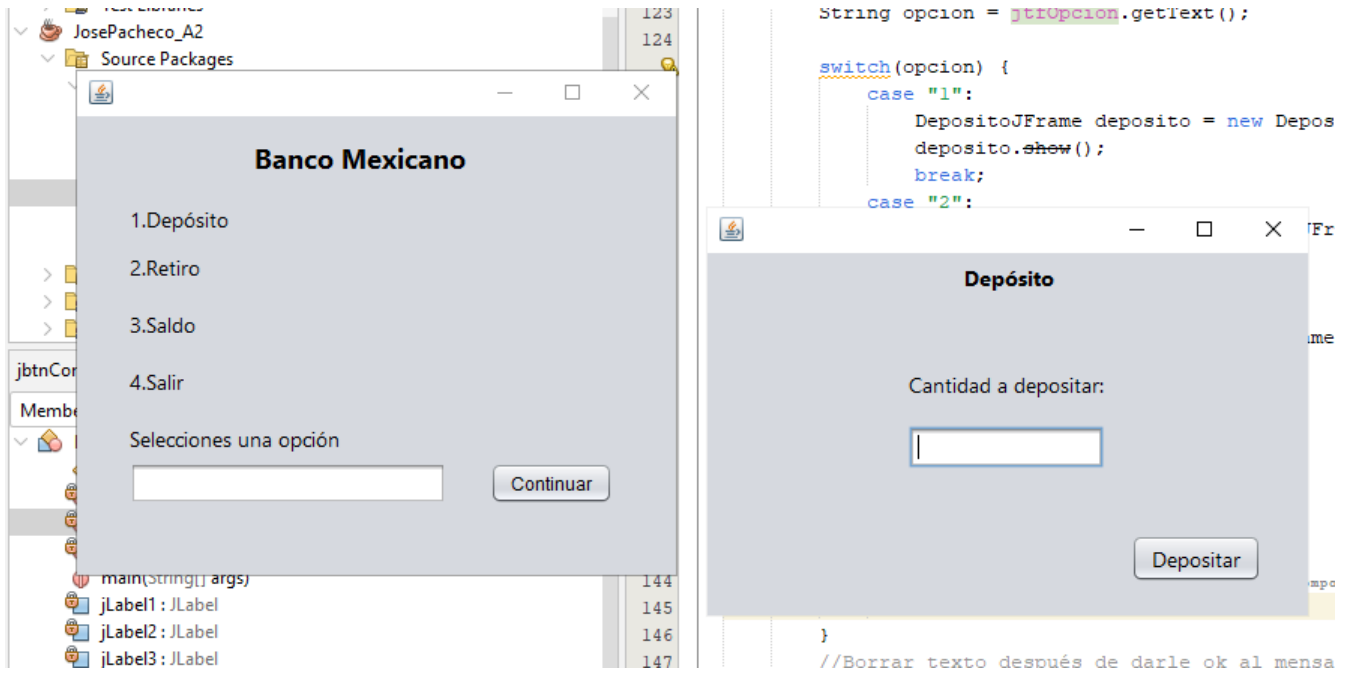


```
122 private void jButtonContinuarActionPerformed(java.awt.event.ActionEvent evt) {  
123     // TODO add your handling code here:  
124     String opcion = jTextFieldOpcion.getText();  
  
125     switch(opcion) {  
126         case "1":  
127             DepositoJFrame deposito = new DepositoJFrame ();  
128             deposito.show();  
129             break;  
130         case "2":  
131             RetiroJFrame Retiro = new RetiroJFrame();  
132             Retiro.show();  
133             break;  
134         case "3":  
135             SaldoJFrame Saldo = new SaldoJFrame();  
136             Saldo.show();  
137             break;  
138             //Cerrar ventana  
139         case "4":  
140             this.dispose();  
141             break;  
142         default:  
143             //Mostrar mensaje de error  
144             JOptionPane.showMessageDialog(parentComponent:null,message:"Opción no valida, por favor intente de n  
145             break;  
146     }  
147     //Borrar texto después de darle ok al mensaje  
148     jTextFieldOpcion.setText("");  
149 }  
150 private void jTextFieldOpcionActionPerformed(java.awt.event.ActionEvent evt) {
```

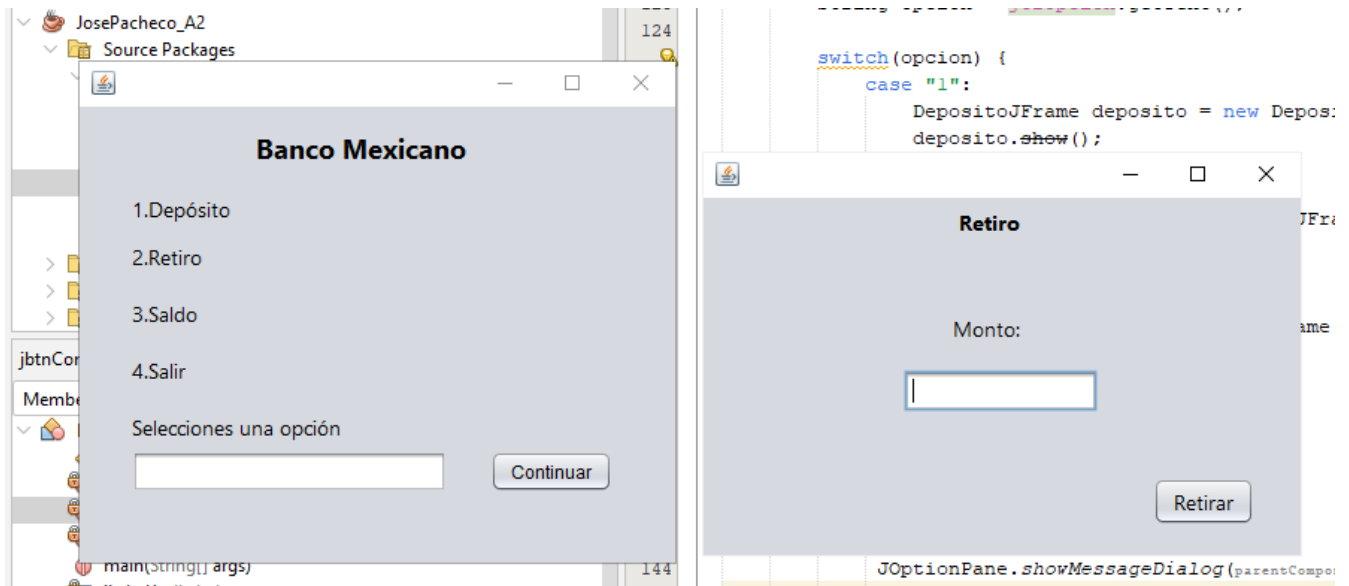
Al ejecutar el form de menú se muestra la interfaz inicial del programa.



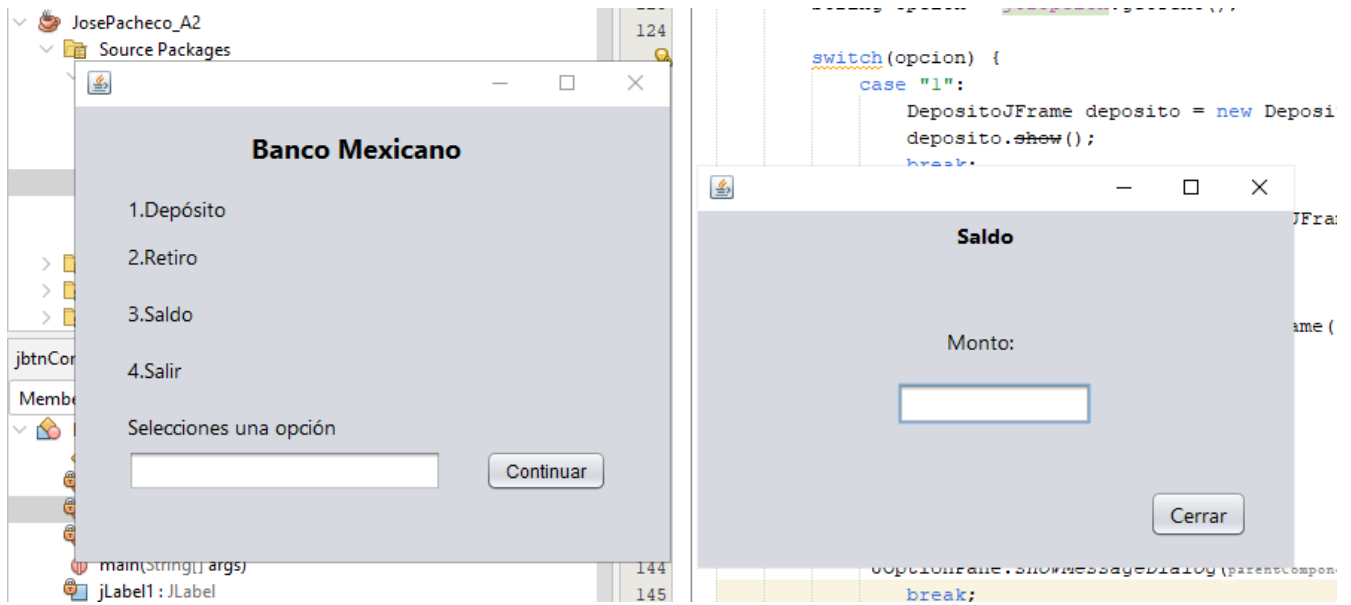
Al ingresar el número 1 y oprimir continuar se muestra la ventana de depósito.



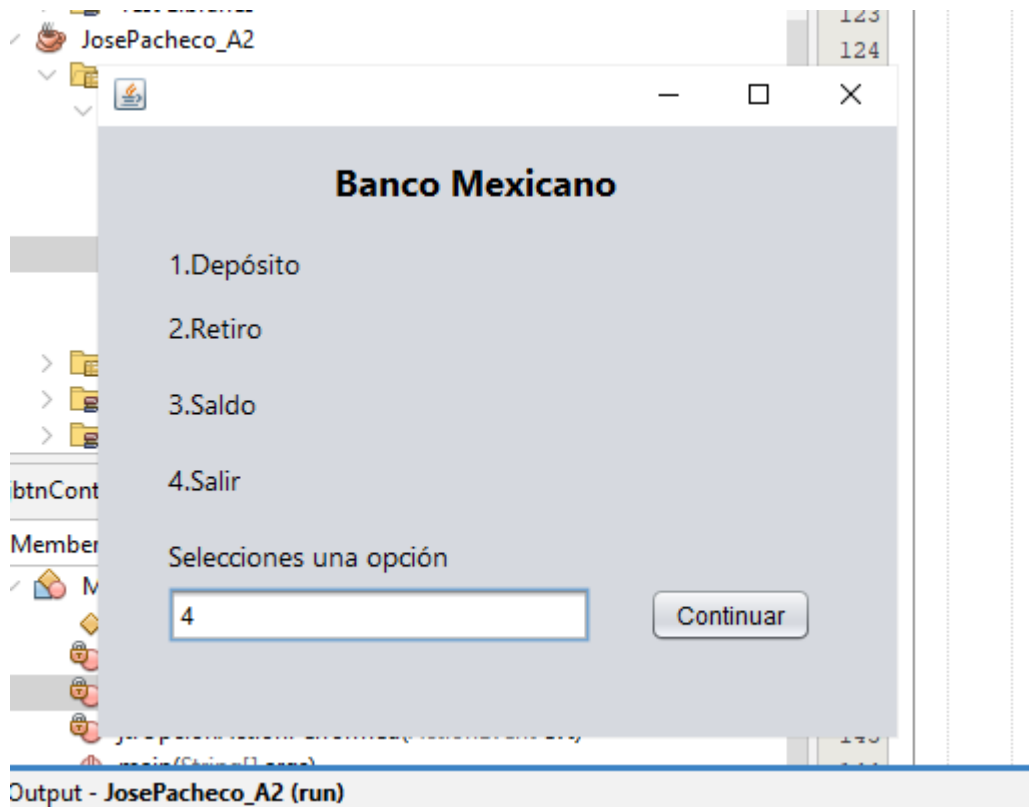
Si se ingresa el número 2 se muestra la ventana de retiro.



Al ingresar el número 3 se muestra la ventana de saldo.



Y por último al ingresar el 4 se cierra la aplicación.




Continuando con el proceso de creación del sistema se debe crear la conexión por medio de MySQL Workbench.

[Browse Documentation >](#)

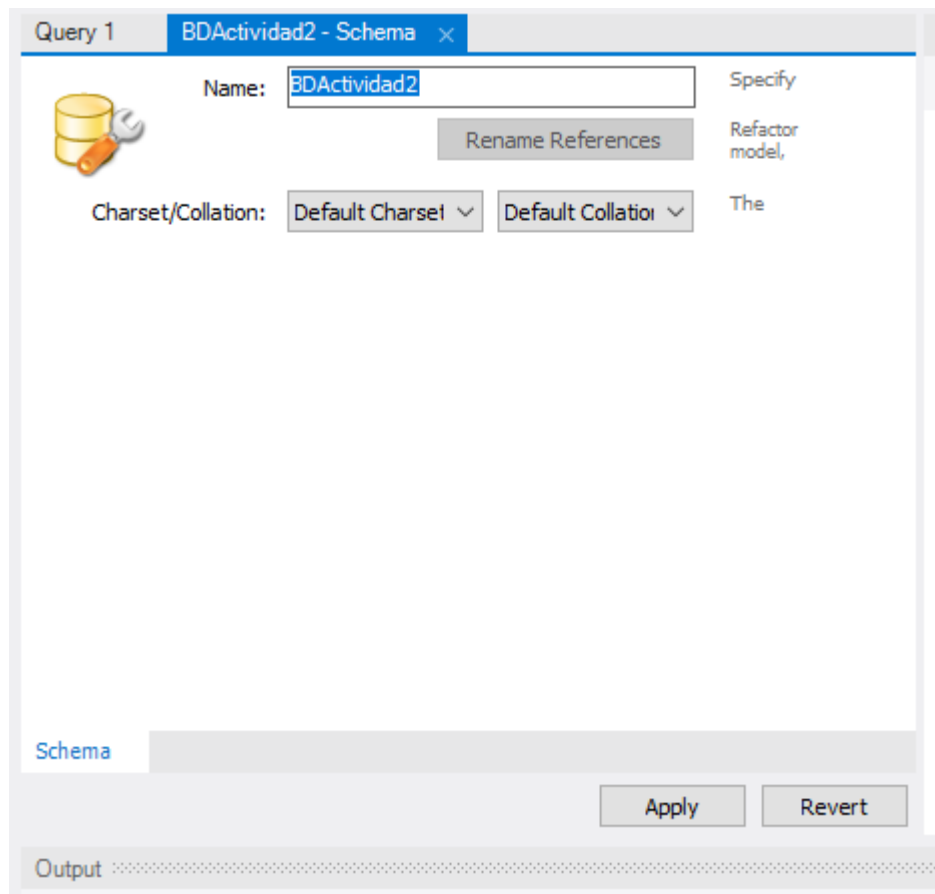
MySQL Connections

conexionlocal

 root

 127.0.0.1:3306

Así como la creación la Base de datos.



Navigator

SCHEMAS

Filter objects

bdactividad2

Tables

Views

Stored Procedures

Functions

phpmyadmin

test


Administration

Schemas

Information

Schema: bdactividad2

bdactividad2 - Schema

Name: bdactividad2

Rename References

Charset/Collation: Default Charset Default Collation

Specify Refactor model, The

Schema

Apply

Revert

En esta captura se muestra la creación de las tablas.

Query 1

bdactividad2 - Schema

cuenta - 1 table

Conexion

Table Name:

Schema: **bdactividad2**

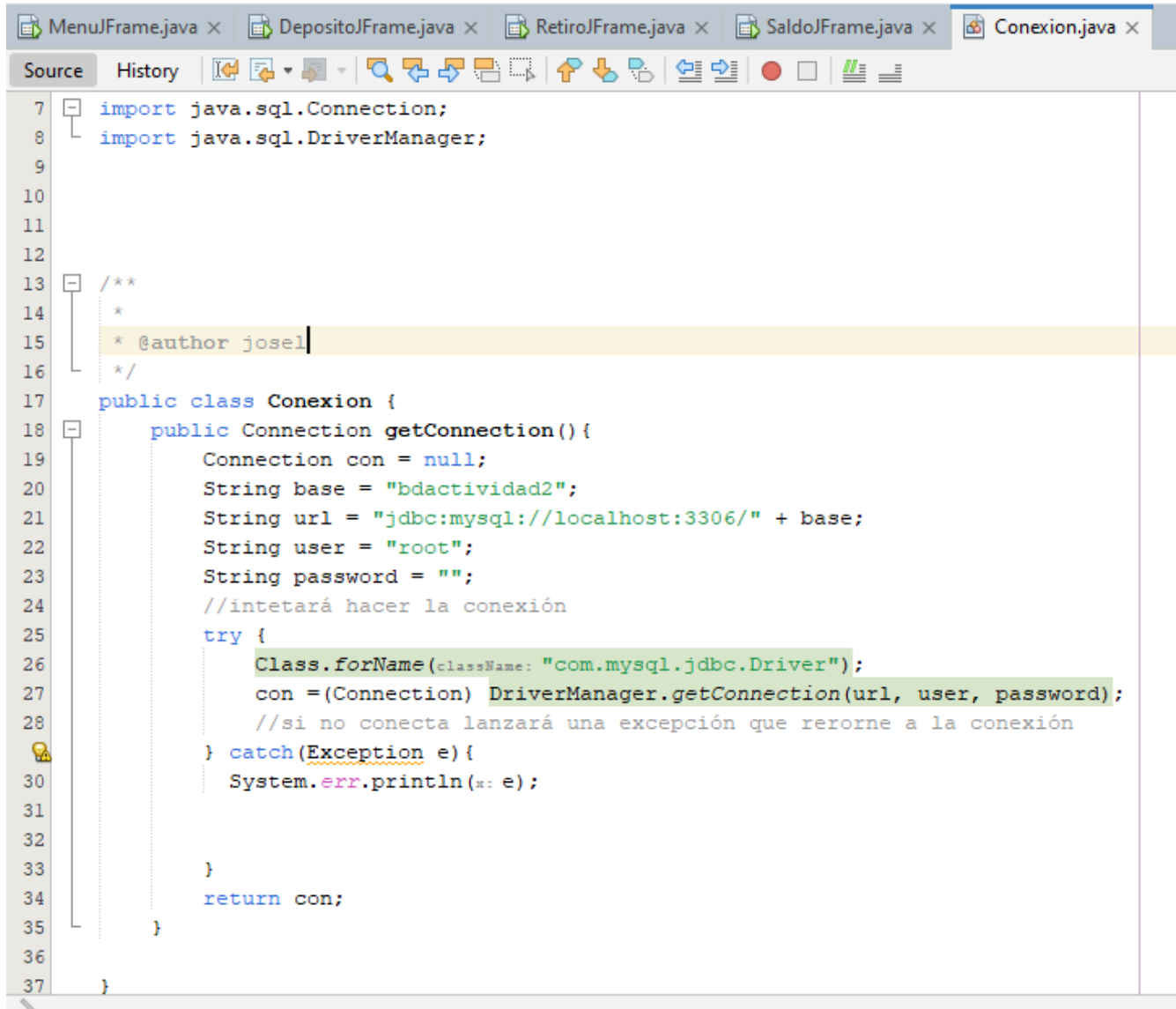
Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	INT(11)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
saldo	DECIMAL(10,0)	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	NULL
		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Column Name:

Data Type:

Como parte final de la actividad se debe crear una nueva clase en Net beans para realizar la conexión entre la aplicación y la base de datos.

Se declaran las variables para la conexión, tomando en cuenta el nombre que se le asignó a la BD en MySQL Workbench y la URL que se generó al crear la conexión local.



```
7  import java.sql.Connection;
8  import java.sql.DriverManager;
9
10
11
12
13  /**
14   *
15   * @author josel
16   */
17  public class Conexion {
18      public Connection getConnection() {
19          Connection con = null;
20          String base = "bdactividad2";
21          String url = "jdbc:mysql://localhost:3306/" + base;
22          String user = "root";
23          String password = "";
24          //intetará hacer la conexión
25          try {
26              Class.forName(className: "com.mysql.jdbc.Driver");
27              con =(Connection) DriverManager.getConnection(url, user, password);
28              //si no conecta lanzará una excepción que rerorne a la conexión
29          } catch (Exception e) {
30              System.err.println(x: e);
31          }
32      }
33      return con;
34  }
35  }
```

Conclusión

La creación de diseños de interfaces y funciones de menú en NetBeans, combinada con el desarrollo de bases de datos en MySQL Workbench, ofrece beneficios sustanciales tanto en entornos laborales como en situaciones cotidianas. En el ámbito laboral, estas habilidades son cruciales para los desarrolladores de software, ya que les permiten construir aplicaciones intuitivas y eficientes que satisfacen las necesidades específicas de los usuarios. La capacidad de diseñar interfaces atractivas y funcionales en NetBeans garantiza una experiencia de usuario positiva, lo que aumenta la productividad y la satisfacción del cliente. Por otro lado, la creación de bases de datos sólidas en MySQL Workbench facilita la gestión de datos empresariales, mejorando la eficiencia operativa y la toma de decisiones informadas. Además, estas habilidades también son valiosas en entornos cotidianos, ya que permiten a los individuos crear aplicaciones personalizadas y sitios web que mejoran su vida diaria. Desde aplicaciones móviles hasta herramientas de gestión del hogar, el conocimiento de NetBeans y MySQL Workbench capacita a las personas para diseñar y desarrollar soluciones innovadoras que optimizan sus actividades diarias.

Referencias

Conectar Aplicación JAVA a una Base de Datos MySQL. (n.d.). Anw.es. Retrieved March 25, 2024, from <https://blog.anw.es/como-conectar-aplicacion-java-a-una-base-de-datos-mysql/>

Desarrollo de aplicaciones con Java: ¿cómo funciona? (2022, July 12). Blog | NextU LATAM. <https://www.nextu.com/blog/desarrollo-aplicaciones-con-java-rc22/>



Enlace Github