Bogotá DC 01/12/2024

**Codificación de módulos del software según requerimientos del proyecto GA7-220501096-AA2-EV01**

Análisis y desarrollo de software

Ficha: 2834899

Alvaro alexander meza gamboa

CC 1127061790

**Introducción**

En el desarrollo de software, la fase de codificación de módulos es fundamental para transformar los requisitos funcionales y no funcionales en un producto concreto. Esta etapa implica la creación de componentes individuales que, al combinarse, conforman el sistema de software completo. En este documento, se aborda la implementación de módulos para un proyecto desarrollado bajo la metodología estructurada, cumpliendo con los requisitos previamente definidos.

Para el desarrollo de este proyecto, se utilizó el entorno de desarrollo NetBeans, un entorno (IDE) que facilita el proceso de codificación, depuración y pruebas del software. En particular, se empleó el conector JDBC (Java Database Connectivity) para gestionar la interacción entre la aplicación y la base de datos MySQL, lo que permitió realizar operaciones de consulta, inserción, actualización y eliminación de datos.

**Objetivo**

Este trabajo tiene como objetivo principal la codificación de los módulos del software conforme a los requerimientos definidos en la fase de análisis del proyecto. Para ello, se empleará el entorno de desarrollo integrado NetBeans y se implementará la conectividad a bases de datos utilizando JDBC (Java Database Connectivity). Con esta implementación, se pretende garantizar que los módulos del sistema sean funcionales, eficientes y estén plenamente integrados con el sistema de gestión de bases de datos MySQL, asegurando así la calidad del software resultante.

**Desarrollo de actividad**

Interfaz de usuario gráfica, Texto, Aplicación

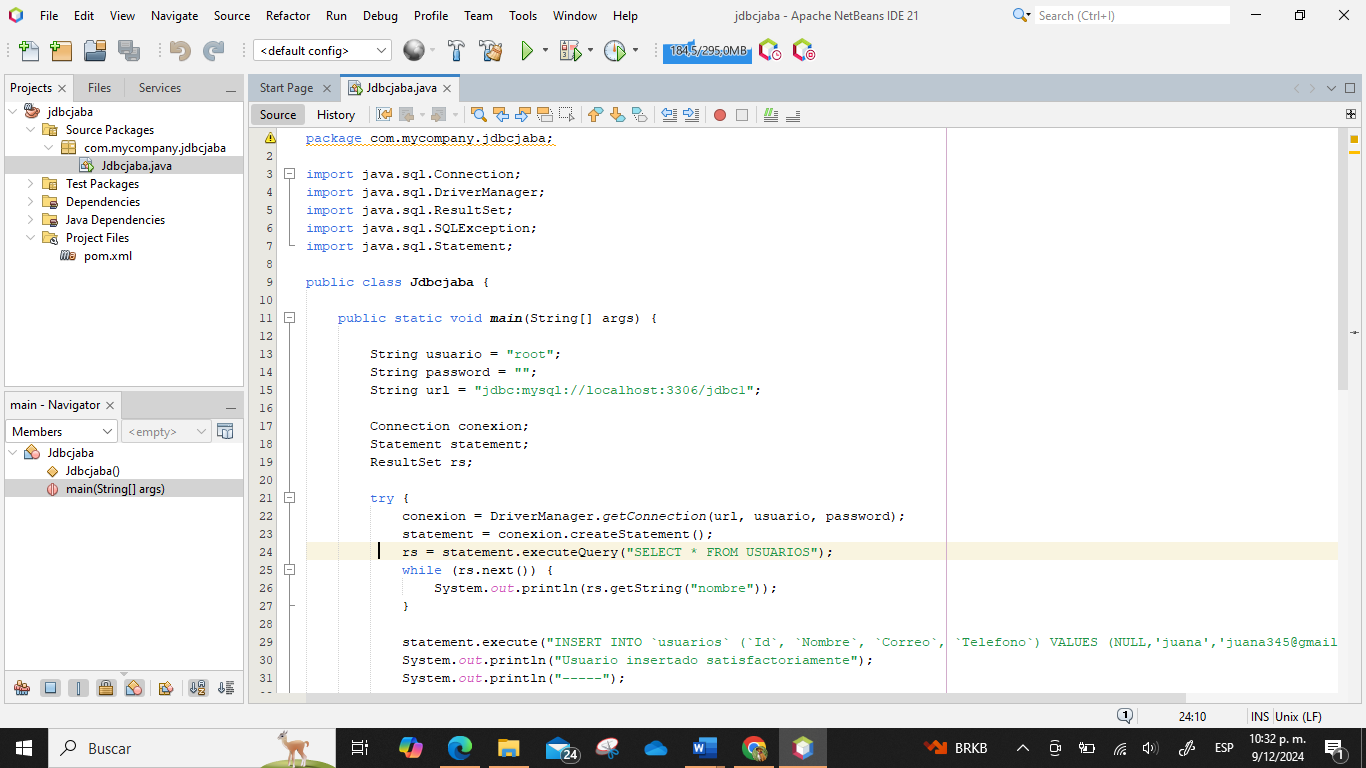
Descripción generada automáticamente

Se inicia creando el proyecto Maven llamado jdbcjaba Dentro de los Source Packages, se define un paquete llamado com.mycompany.jdbcjaba.

En la sección de Dependencies, se añade dos archivos JAR:

* mysql-connector-j-8.0.30.jar: que es el conector para poder trabajar con una base de datos MySQL desde mi proyecto.
* protobuf-java-3.25.1.jar: que contiene las dependencias necesarias para utilizar Google Protocol Buffers en Java.

Se configura la dependencia de JDK 21 en el apartado de Java Dependencies, lo cual indica que estoy utilizando esta versión de java.



Estas líneas importan clases que se necesitan para trabajar con bases de datos, manejar excepciones y registrar errores.

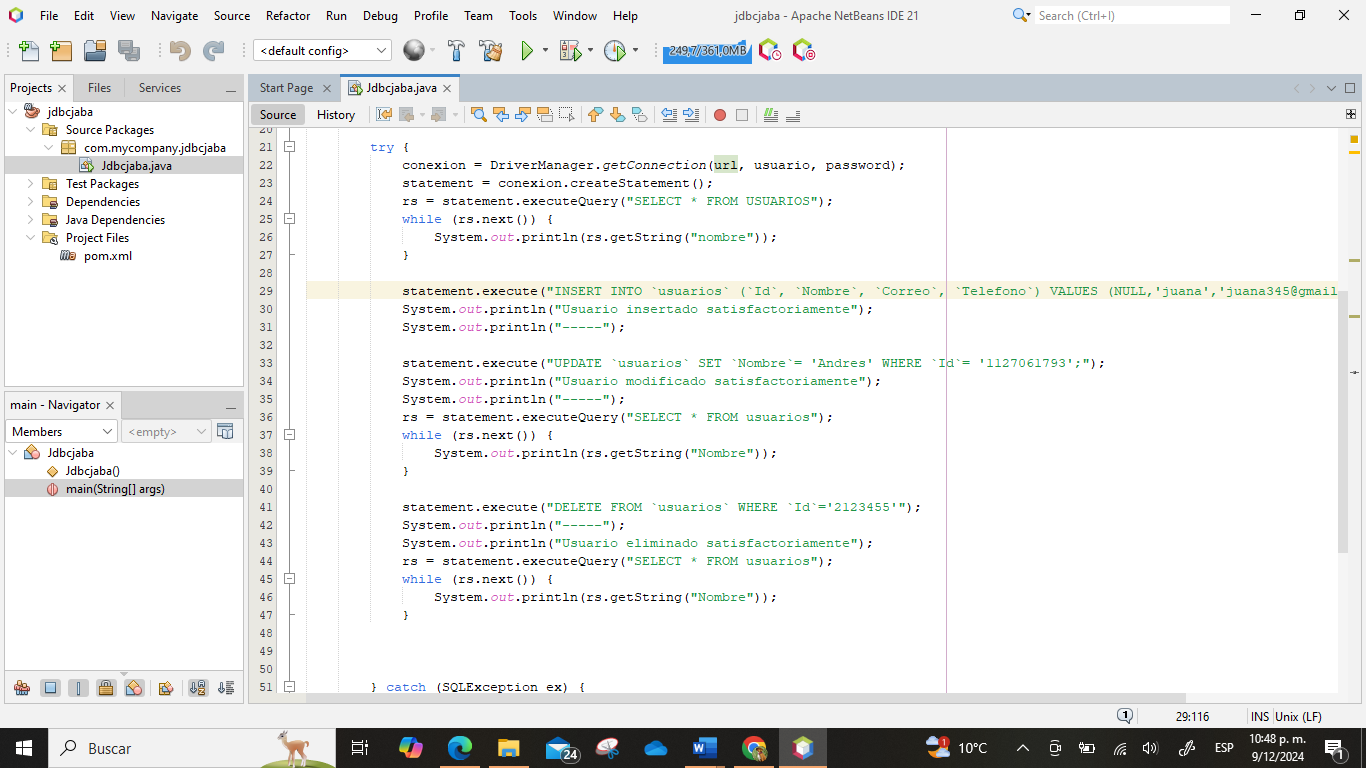
* Connection: Se usa para representar la conexión a la base de datos.
* DriverManager: Gestiona la creación de conexiones a bases de datos.
* ResultSet: Representa el conjunto de resultados devuelto por una consulta SQL.
* SQLException: Maneja los errores que ocurren durante las operaciones de la base de datos.
* Statement: Se utiliza para ejecutar consultas SQL.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

En este fragmento de código fue definir una clase llamada jdbcjaba que incluye el método principal main. En este método, se configuran los elementos necesarios para establecer una conexión a una base de datos MySQL usando JDBC (Java Database Connectivity).

1. URL de la base de datos: Definí la URL de la base de datos como "jdbc:mysql://localhost:3306/jdbc1"; de modulos. Este es el formato estándar de una conexión JDBC, donde:
   * jdbc:mysql:// especifica que se trata de una conexión JDBC a una base de datos MySQL.
   * localhost indica que la base de datos está alojada en mi máquina local.
   * 3306 es el puerto predeterminado de MySQL.
   * jdbc1 es el nombre de la base de datos que quiero conectar.
2. Usuario y contraseña:
   * El usuario está definido como root, que es el usuario predeterminado en muchas instalaciones de MySQL.
   * La contraseña está vacía (""), lo que indica que no se ha asignado una contraseña para este usuario en mi configuración local.
3. Objetos JDBC:
   * Connection (coneccion): se declara este objeto para establecer la conexión entre la aplicación Java y la base de datos.
   * Statement (statement): Este objeto permitirá ejecutar consultas SQL una vez que la conexión esté establecida.
   * ResultSet (rs): Este objeto contendrá los resultados de las consultas que se ejecuten a la base de datos, como, por ejemplo, una consulta SELECT que devuelva datos.



* Se carga el driver de MySQL.
* Se establece una conexión a la base de datos usando el DriverManager.
* Después, creé un objeto Statement para ejecutar una consulta SQL (SELECT \* FROM usuarios).
* Finalmente, recorrí los resultados de la consulta, mostrando los nombres de los usuarios en la consola.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Se realiza una inserción de datos en la tabla usuarios y luego ejecutar una consulta para verificar los resultados.

* Se utiliza la instrucción INSERT INTO para agregar un nuevo registro a la tabla usuarios. En este caso, se inserta el nombre "Juana" correo “juana345@gmail.com telefono “31465635” dejando el campo id como NULL para que se autogenere, asumiendo que la columna id es un campo autoincremental.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

En este código, se realiza una actualización de un registro en la tabla usuarios y luego se hace una consulta para verificar el cambio.

* + Se utiliza UPDATE para modificar el nombre del usuario cuyo id es 1127061793. En este caso, se cambia el valor de la columna nombre a "Andres" solo para el registro que tiene el id igual a 1127061793.
  + Después de realizar la actualización, ejecuté nuevamente una consulta SELECT \* FROM usuarios para obtener todos los registros actuales de la tabla usuarios, incluyendo el registro actualizado.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Se elimina un registro de la tabla usuarios y luego consultar los datos restantes para verificar la eliminación. A continuación, se describe el proceso:

* + Se utiliza la instrucción DELETE para eliminar el registro cuyo id es igual a 2123455. Esto elimina permanentemente dicho registro de la tabla usuarios.
  + Después de realizar la eliminación, ejecuté una consulta SELECT \* FROM usuarios para obtener todos los registros restantes en la tabla usuarios, exceptuando el que fue eliminado.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente



Construcción del proyecto Maven:

* El proyecto jdbcjaba se está construyendo con Maven, lo que incluye la fase de compilación (compile) y empaquetado (jar).
* No hay necesidad de recompilar clases, ya que todos los archivos están actualizados.

Ejecución del programa:

* El programa está ejecutando consultas SQL, como un SELECT \* FROM usuarios, y devuelve los nombres de los usuarios en la base de datos. En este caso, los nombres impresos en consola incluyen:
  + **Andres**
  + **ALVARO**
  + **Juana**
* Estos nombres se repiten varias veces en el resultado, lo que sugiere que se están realizando múltiples consultas o la tabla tiene registros duplicados de estos nombres.

Éxito en la construcción y ejecución:

* La construcción del proyecto fue exitosa, como lo indica el mensaje **"BUILD SUCCESS"**.
* También se muestra el **tiempo total** de ejecución (4.557 segundos) y la hora de finalización de la ejecución.

Una captura de pantalla de una computadora

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Se muestra la estructura de la base de datos creada mediante phpMuAdmin la cual opera por medio de Mysq utilizando JDBC desde la consola de NetBeans.

**Conclusión**

La codificación de los módulos del software según los requerimientos del proyecto constituye un paso crucial para garantizar sistemas funcionales y eficientes. En este trabajo se ha evidenciado cómo herramientas como NetBeans simplifican tanto la implementación como la depuración del código, mientras que el uso de JDBC garantiza una conexión sólida y confiable con la base de datos MySQL. Al concluir el proceso, los módulos desarrollados satisfacen los requisitos establecidos, demostrando una adecuada interpretación de las especificaciones y la eficacia de las tecnologías empleadas. Este proyecto no solo responde a las expectativas del cliente, sino que también establece una base robusta para futuras ampliaciones y mantenimientos del sistema.