

GUIA #4

CONECTIVIDAD A BASES DE DATOS

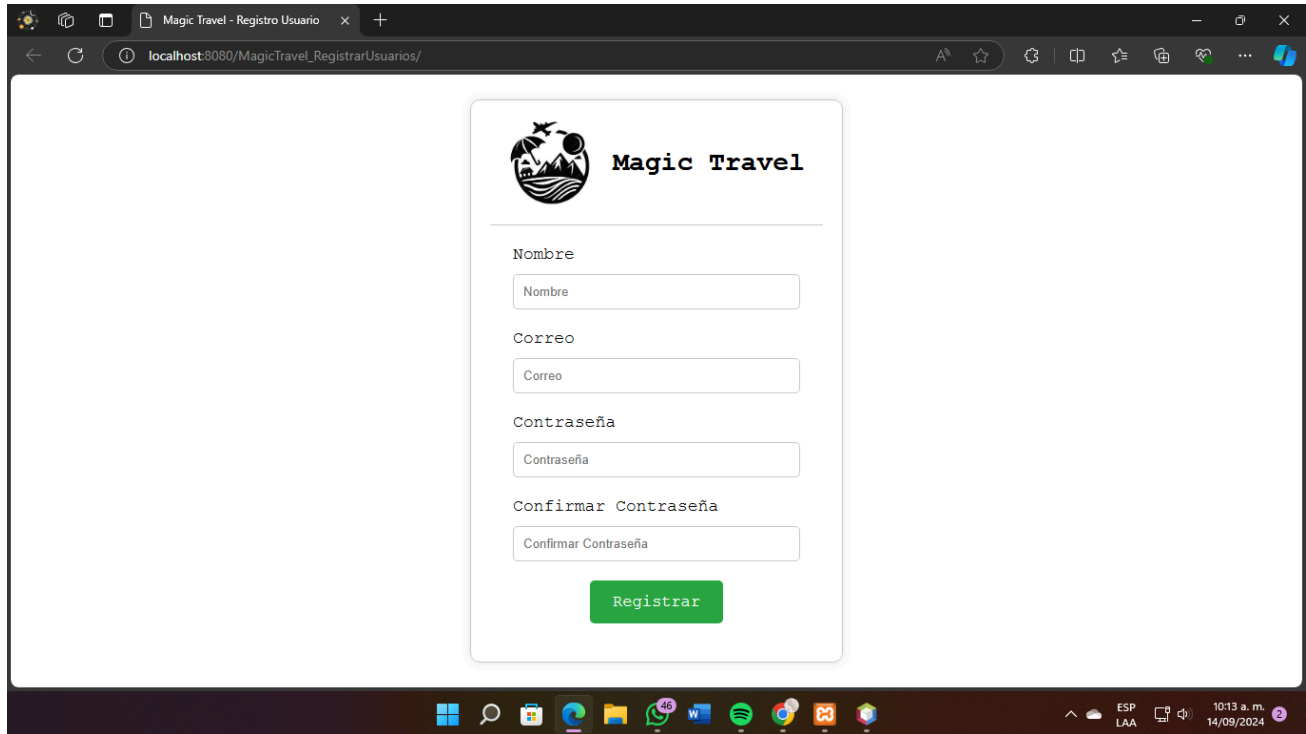
TALLER DE PROGRAMACIÓN
OLGA LUCÍA ROA BOHÓRQUEZ

KAROL STEFANY ORDOÑEZ PEÑALOZA
YEFERSON PIÑEROS PEÑA

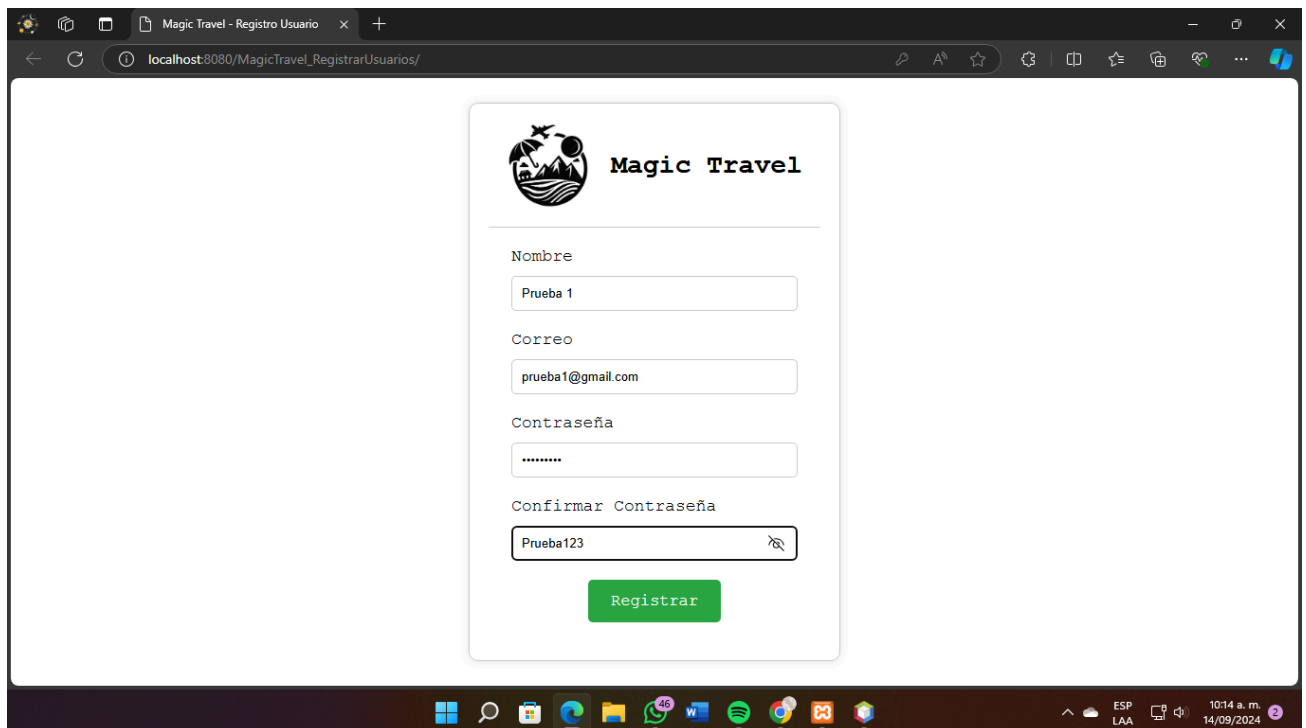
14/09/2024

INTRODUCCIÓN

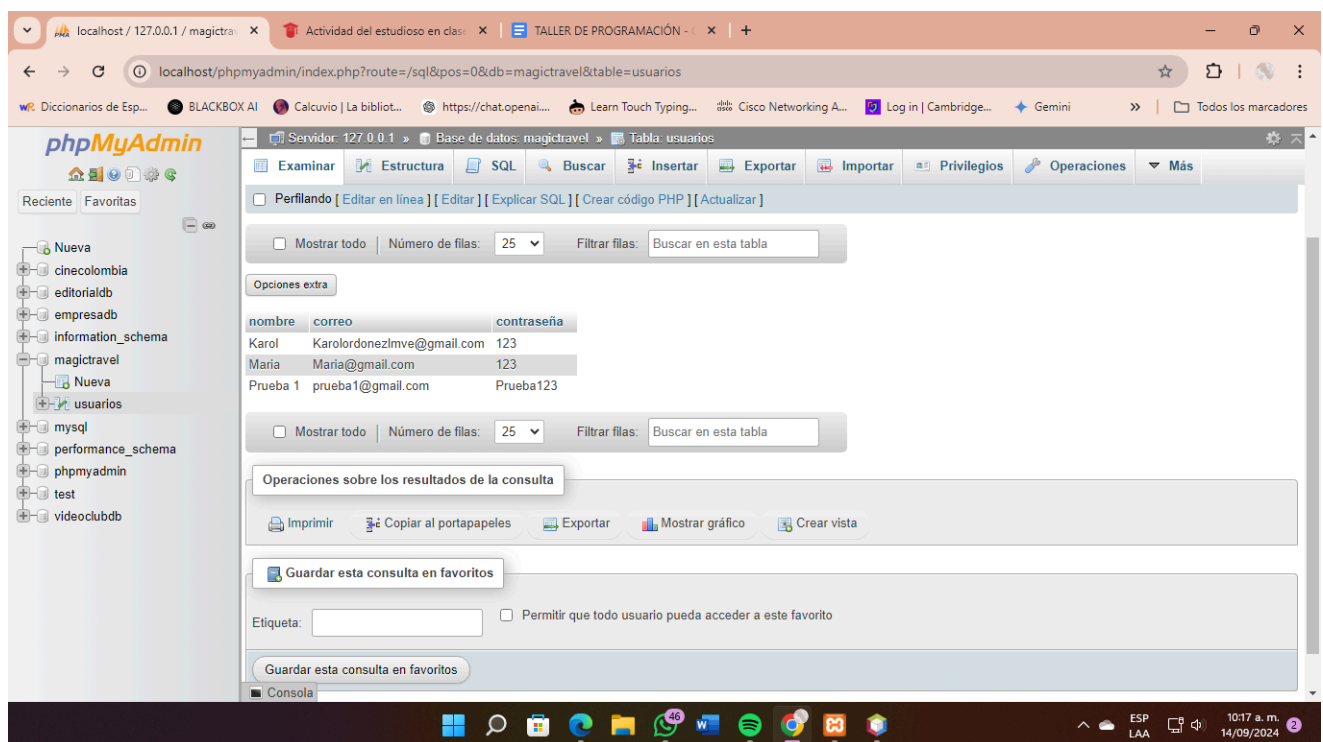
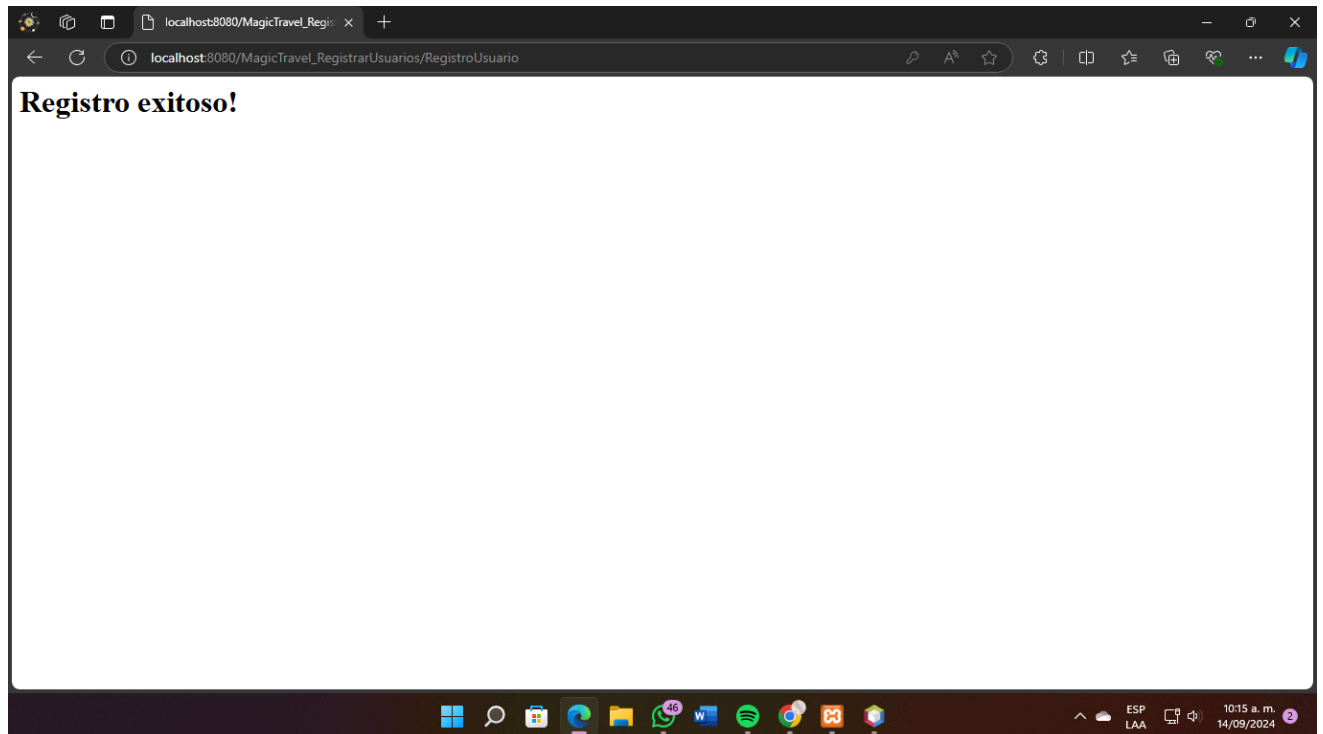
Diseñe e implemente una interfaz gráfica de usuario utilizando la librería Swing de Java o tecnología web (html, css y javascript o librerías de js) para realizar el Front-End del CRUD de la tabla. Recuerde validar campos de entrada.



The screenshot shows a web browser window with the address bar displaying "localhost:8080/MagicTravel_RegistrarUsuarios/". The page features a registration form for "Magic Travel". The form includes a logo with a mountain and sun icon, followed by the text "Magic Travel". Below this, there are four input fields labeled "Nombre", "Correo", "Contraseña", and "Confirmar Contraseña". Each field contains a placeholder text matching its label. At the bottom of the form is a green button labeled "Registrar". The browser's taskbar at the bottom shows various application icons and the system clock indicating 10:13 a.m. on 14/09/2024.



This screenshot shows the same registration form as above, but with sample data entered into the fields. The "Nombre" field contains "Prueba 1", the "Correo" field contains "prueba1@gmail.com", the "Contraseña" field contains "*****", and the "Confirmar Contraseña" field contains "Prueba123". The green "Registrar" button remains at the bottom. The browser window and taskbar are identical to the previous screenshot, showing the same URL and system time of 10:14 a.m. on 14/09/2024.



PREGUNTAS ORIENTADORAS

¿Cuáles fueron los aprendizajes obtenidos al realizar esta guía?, liste como mínimo 3 aprendizajes y relaciónelos con su futuro quehacer profesional.

- Diferencias entre BD Relacionales (SQL) y No Relacionales (NoSQL) que pueden ser útil para su debida aplicación a diferentes tipos de proyectos según lo necesitemos en ese momento
- La naturaleza de los patrones de diseño MVC y DTO que resulta útil para el manejo de datos en cada proyecto así como la transferencia de los mismos respectivamente a cada aplicación de servicio web (DTO) o cada controlador (MVC)
-

¿Dónde presentó mayor dificultad resolviendo la guía? y ¿cómo lo resolvieron? ¿Cuáles fueron las estrategias de solución?

- Obtener información sobre el patrón DAO fue complicado, ya que la mayor parte de información general estaba en wikipedia la cual no es una fuente confiable. Esto pudo solucionarse mediante la búsqueda minuciosa en diferentes fuentes y recatando información de cada una
- Se presentó dificultad en la elaboración del video en inglés ya que el uso de un lenguaje técnico en otro idioma es más complejo que el tradicional. La solución a este problema fue reescribir varias veces el diálogo, escucharlo y repetirlo para posteriormente grabarlo en fragmentos.

ACTIVIDAD DE TRABAJO AUTÓNOMO

1. Diferencia entre ODBC y JDBC

- ODBC es una interfaz estándar de Windows que facilita la comunicación entre sistemas de gestión de bases de datos y aplicaciones, generalmente desarrolladas en C o C++.
- JDBC es una interfaz estándar para conectar bases de datos con aplicaciones escritas en Java de Oracle.

“Database Toolbox ofrece una biblioteca en C++ que se conecta nativamente a un controlador ODBC y una biblioteca Java que se comunica directamente con un controlador JDBC puro. Según el entorno y los objetivos, se debe elegir entre ODBC o JDBC.”¹

El controlador ODBC nativo es recomendable cuando se busca:

- Mayor velocidad en la importación y exportación de datos.
- Manejo eficiente de grandes cantidades de memoria en operaciones de datos.
- Acceso a todas las funcionalidades excepto la función `runstoredprocedure`.

El controlador JDBC es ideal para:

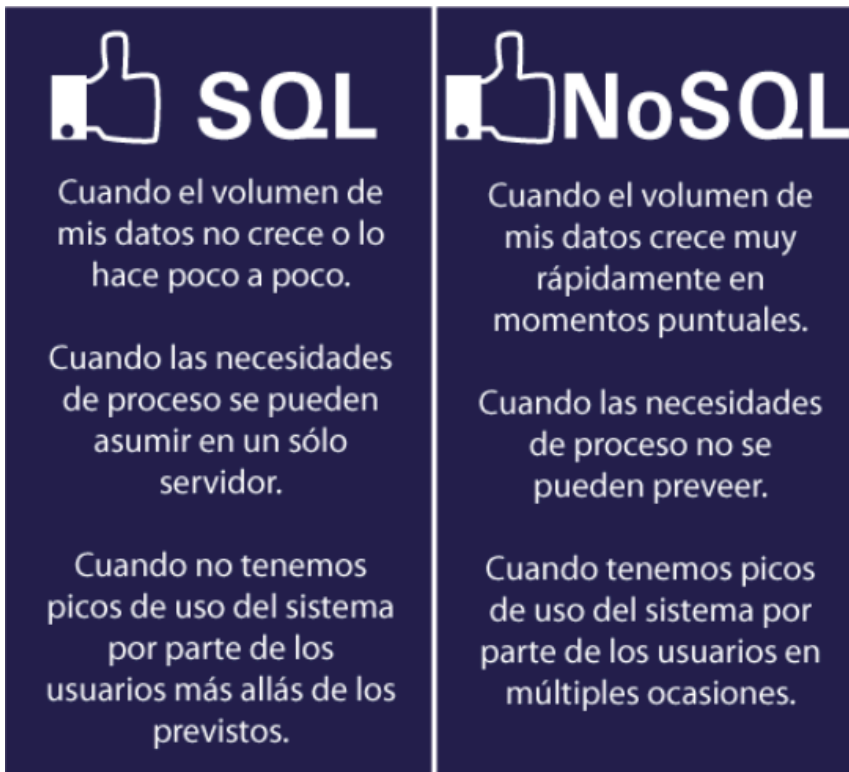
- Garantizar independencia de la plataforma, permitiendo trabajar en distintos sistemas operativos (incluidos Mac y Linux) y versiones de controladores.
- Acceder a todas las funciones del Database Toolbox.

Ambos controladores tienen una limitación relacionada al rendimiento de la memoria. En el caso de ODBC nativo, está restringido por la memoria de MATLAB, mientras que en JDBC influye tanto la memoria de MATLAB como la memoria de JVM.

2. Diferencias y Similitudes entre base de datos relacional y bases de datos NoSQL

- BD Relacionales: Son colecciones de datos organizados en conjuntos de tablas descritas de manera formal, desde las cuales se permite acceder a los datos o montarlos de maneras diferentes acorde a las necesidades o preferencias sin tener que reorganizar las tablas de las bases.
- BD No Relacionales: Son diseñadas con la labor específica de modelar datos específicos con esquemas que sean flexibles para crear aplicaciones más modernas. Son realmente fáciles de utilizar lo que las caracteriza por una funcionalidad buena en el rendimiento a escala

De manera más práctica, aquí los casos en los que podemos usar una u otra:



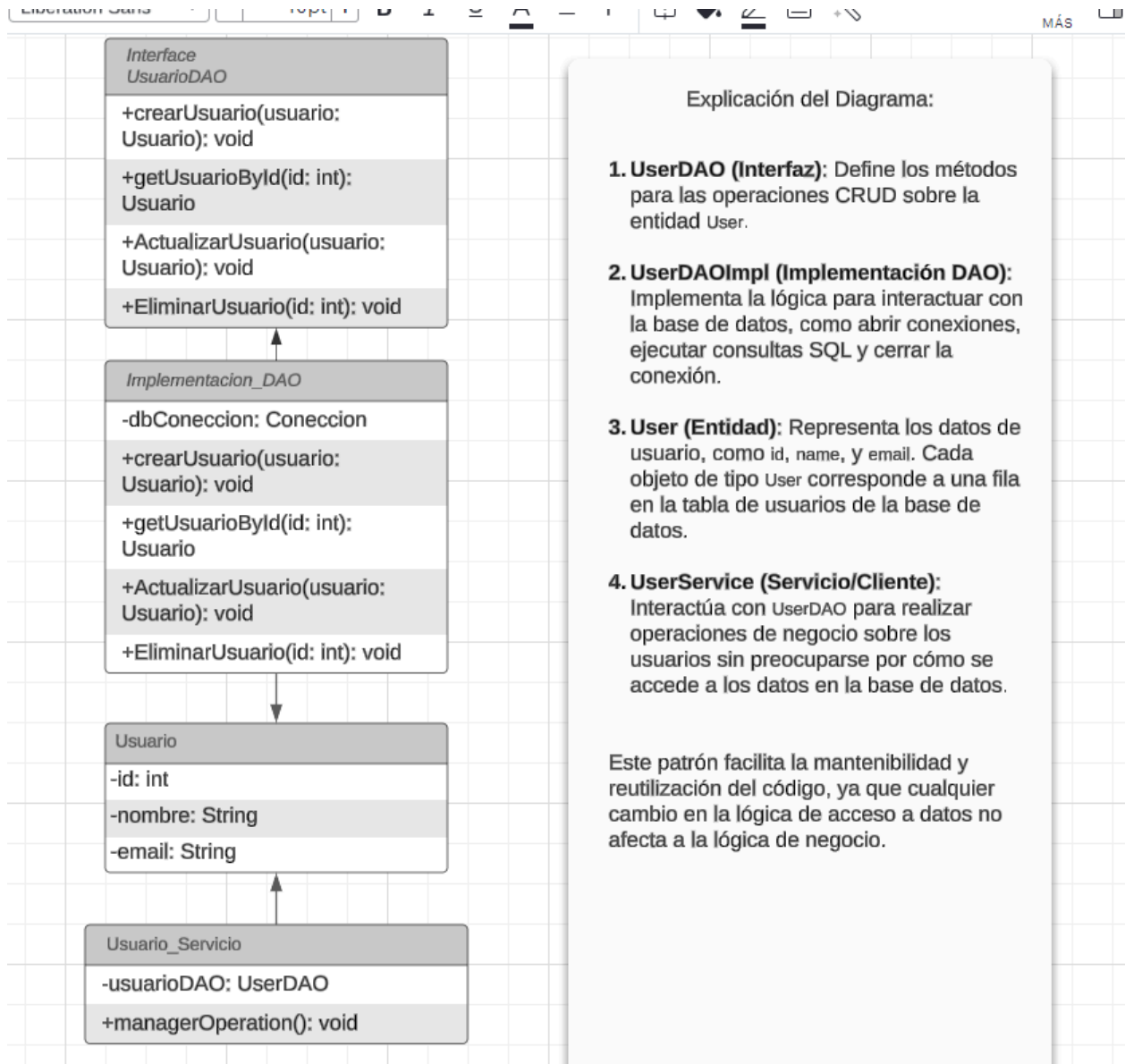
(Figura 1. Por ACADEMIA PRAGMA)²

- Diferencias entre SQL y NoSQL:
 1. Estructuras de los datos:
 - SQL: Organizan los datos en tablas con filas y columnas (esquema fijo), siguiendo un formato estructurado y normalizado.
 - NoSQL: Usa estructuras flexibles como documentos, grafos, pares clave-valor, y columnas anchas. No requieren un esquema fijo
 2. Modelo de Datos:
 - SQL: Está basada en el modelo de datos relacional, donde las relaciones entre los datos son clave.
 - NoSQL: Los datos no están necesariamente relacionados mediante claves foráneas. Se enfocan más en la estructura interna del dato.
 3. Escalabilidad:
 - SQL: Generalmente escalan de forma vertical (mejorando el hardware del servidor).

- NoSQL: Suelen escalar de forma horizontal (añadiendo más servidores o nodos).
4. Consultas:
- SQL: Usan SQL para realizar búsquedas
 - NoSQL: No utilizan SQL necesariamente; en su lugar, usan lenguajes de consulta específicos para su tipo de datos.
- Similitudes entre SQL y NoSQL:
1. **Almacenamiento de Datos:** Ambas permiten almacenar grandes volúmenes de datos.
 2. **Consulta de Datos:** Pueden proporcionar mecanismos eficientes para recuperar y consultar los datos, aunque utilizan diferentes lenguajes o métodos.
 3. **Disponibilidad de Indexación:** Tanto en bases de datos relacionales como no relacionales se pueden crear índices para mejorar la velocidad de las consultas.
 4. **Soporte de Transacciones:** Aunque manejan las transacciones de diferentes maneras (ACID vs BASE), ambas pueden gestionar transacciones según el nivel de consistencia requerido.

3. Definir Patrón Data Access Object (DAO), explique mediante un diagrama de UML.

DAO es un patrón arquitectónico el cual nos permite separar la lógica de acceso a datos de los objetos de negocio, de tal forma que encapsula toda la lógica de acceso de datos respecto a la aplicación.



4. Definir patrones Modelo Vista Controlador (MVC) y Data Transfer Object (DTO)

Modelo vista controlador (MVC): Es un patrón de diseño arquitectónico que separa una aplicación en tres componentes principales:

1. **Modelo:** Representa los datos de la aplicación y la lógica de negocio. Es responsable de acceder a la base de datos, gestionar el estado y aplicar reglas de negocio. El Modelo notifica a la Vista cuando hay cambios en los datos.

2. **Vista:** Es la interfaz de usuario que presenta los datos que provienen del Modelo. La Vista recibe actualizaciones del Modelo y presenta los datos de forma adecuada al usuario. No contiene lógica de negocio, solo se encarga de la presentación.
3. **Controlador:** Actúa como intermediario entre la vista y el modelo. Recibe las entradas del usuario desde la Vista, procesa esa información, por lo general interactuando con el modelo, y determina qué Vista se debe mostrar o qué acciones realizar.

Data Transfer Object (DTO): Es un patrón de diseño que se utiliza para transferir datos entre diferentes capas de una aplicación, especialmente en el contexto de aplicaciones distribuidas o servicios web.

1. Un DTO es un objeto plano que solo contiene propiedades o atributos para almacenar datos, sin lógica de negocio o comportamientos. Se utiliza principalmente para agrupar múltiples atributos en una única estructura para transportarlos entre sistemas.
2. Su propósito es mejorar el rendimiento y la eficiencia al reducir el número de llamadas necesarias para pasar datos entre las capas de la aplicación o entre diferentes sistemas.
3. Los DTO se usan comúnmente cuando hay que transferir datos a través de una red o entre procesos, ya que encapsulan los datos en un solo objeto, lo que minimiza el tráfico de datos innecesarios.

5. Defina qué es JSON y para que se utiliza.

JSON es un formato de texto que forma parte del sistema de JavaScript y que se deriva de su sintaxis, pero no tiene como objetivo la creación de programas, sino el acceso, almacenamiento e intercambio de datos. Usualmente es conocido como una alternativa al lenguaje XML.”³

Al ser un documento digital, JSON almacena información de manera organizada con el fin de poder hacer una búsqueda más fácil y de manera organizada sobre esta. Se utiliza para obtener información clara y de manera simple con valores fijos como “nombre” “id” “correo” que funcionan como indicadores de la información que contienen.


ACTIVIDAD DE COMPROBACIÓN DE TRABAJO AUTÓNOMO

The student makes a video IN ENGLISH of a minimum of 1 and a maximum of 3 minutes exposing a summary of the main idea of the theoretical framework, the consultation of the independent work activity and the code developed in the practical example proposed in the introductory process session 1 and 2. The scholar publishes the video in the activity FORUM.

DIALOGO DEL VIDEO:

The document discusses the differences between ODBC and JDBC, where ODBC offers faster and more efficient data and memory handling in C/C++ applications, while JDBC provides platform independence and is better suited for Java applications. It also highlights the characteristics of relational databases (SQL) and NoSQL databases, where SQL uses fixed structures and scales vertically, while NoSQL is more flexible and scales horizontally. Design patterns such as DAO, which separates data access logic, MVC, which divides the application into model, view, and controller, and DTO, which facilitates data transfer between layers, are explained. Lastly, JSON is described as a text format derived from JavaScript for storing and exchanging data in a simple and organized manner.

Link Video:

 WhatsApp Video 2024-09-14 at 2.58.01 PM.mp4

“<https://drive.google.com/file/d/1wzVtkw5HkWPewI5UupXyn4-0bTBjOOIm/view?usp=sharing>”

BIBLIOGRAFÍA

1. <https://la.mathworks.com/help/database/ug/choose-between-odbc-and-jdbc-drivers.html> - Mathworks, 09/2024.
2. <https://www.pragma.com.co/academia/lecciones/bases-de-datos-relacionales-vs.-no-relacionales> - Figura 1. ACADEMIA PRAGMA, 09/2024.
3. <https://blog.hubspot.es/website/que-es-json> - HubSpot, 09/2024