Desarrollo Guía 4

Jullians Mauricio Amado Gutierrez

Tomas Alejandro Santiago Reyes

Juan Esteban Cabal Bautista

Universidad Manuela Beltrán

Taller de programación

Olga Lucia Roa Bohórquez

17 de marzo de 2025

Preguntas orientadoras

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

Comprensión a la hora de tener un código y saber para que sirve cada elemento que esta dispuesto a desarrollarse en la guía, ya que esto impulsara a la comprensión de códigos mas extensión y de un peso mayor al trabajar con una compañía.

Adaptación con un código nuevo, el no tener que empezar de 0 a desarrollar todas las ideas que se requieren, sino que, empezar a reutilizar uno ya creado para así desarrollar habilidades sobre el manejo de códigos ajenos.

Lectura del código, no solo es adaptarse y comprender, es saber que hace cada variable, cada elemento que hace que el código funcione, saber que es lo que hace la ejecución, y saber sustentar con buenos argumentos el código que nos dejaron y que en un futuro nos dejaran para sustentarlo y explicarlo a las demás personas

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

No tuvimos ninguna complicación a la hora de resolver la guía, desde el inicio hubo un poco de confusión sobre que tocaba hacer exactamente, pero todo después fluyó de una manera en la que pudimos desarrollar esta guía de laboratorio.

1. Diferencia entre ODBC y JDBC

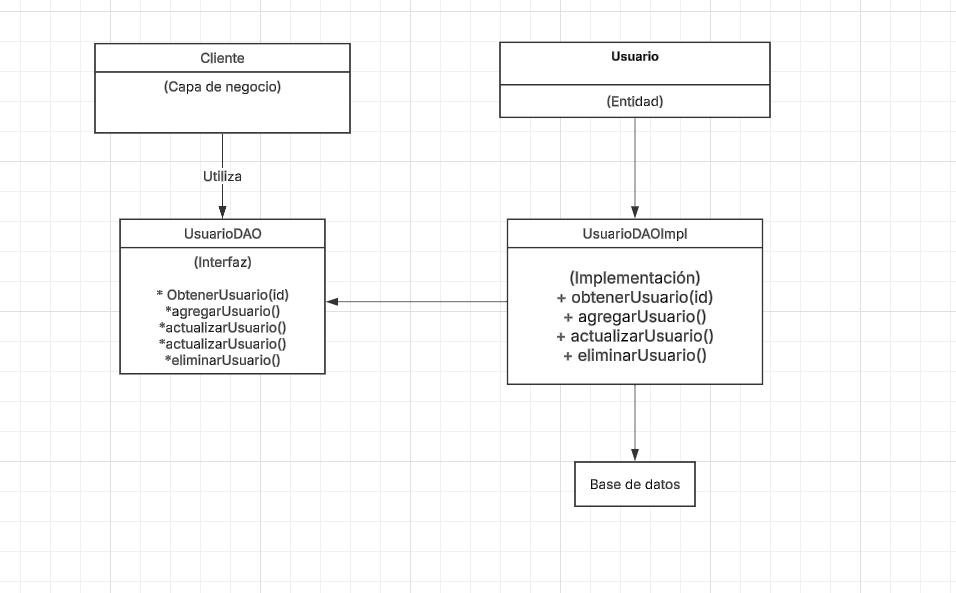
|  |  |
| --- | --- |
| ODBC | JDBC |
| Api estándar de Microsoft que permite a las aplicaciones acceder a bases de datos por medio de un controlador. | Api específica para java que permite la conexión a bases de datos. Proporciona interfaz estándar para interactuar con diferentes bases de datos. |
| Multiplataforma (C, C++, etc) | Solo Java |

1. Diferencias y similitudes entre bases de datos relacional y bases de datos NoSQL.

|  |  |
| --- | --- |
| Relacional | NoSQL |
| Se basa en tablas y filas | Basado en documentos, clave-valor, grafos o columnas. |
| Usa lenguaje SQL | Lenguajes específicos MongoDB etc. |

* 1. Similitudes:
     1. ambas permiten almacenar y consultar datos.
     2. Se pueden usar en sistemas de alto rendimiento
     3. Admiten esquemas y relaciones entre datos (de forma diferente)

1. Patron Data Access Object (DAO)
   1. El patrón DAO se usa para separar la lógica de acceso a la base de datos de la lógica de negocio, es decir, funciona con una interfaz abstracta para realizar operaciones CRUD sin exponer los detalles de la persistencia.



1. Patrones MVC y DTO
   1. MVC(Modelo Vista Controlador)
      1. Es un patrón arquitectónico que separa la aplicación en tres componentes.
         1. Modelo: maneja la lógica del negocio y los datos de entrada.
         2. Vista: encargado de la presentación.
         3. Controlador: gestiona la interacción entre el usuario y el modelo.
   2. DTO(Data Transfer Object)
      1. Es un patrón usado para transferir datos entre las diferentes capas de una aplicación sin exponer directamente la base de datos.
2. ¿Qué es JSON y para que se utiliza?
   1. JSON o JavaScript Object Notation, es un formato ligero para intercambio de datos basado en texto, resulta fácil de leer tanto a las personas como a los sistemas.
      1. Se usa para la transferencia de datos en APIs y servicios web.
      2. Almacenamiento de configuración en aplicaciones.
      3. Intercambio de datos entre sistemas heterogéneos.

|  |  |  |
| --- | --- | --- |
| Funcionalidad | Bases de datos SQL | Bases de datos NoSQL (MongoDB) |
| C- Create | INSERT INTO tabla | db.coleccion.insertOne({ campo1: "valor1", campo2: "valor2" }); |
| R- Read | SELECT \* FROM tabla WHERE columna = valor; | db.coleccion.find({ campo: "valor" }); |
| U-Update | UPDATE tabla SET columna = nuevo\_valor WHERE condición; | db.coleccion.updateOne({ campo: "valor" }, { $set: { campo: "nuevo\_valor" } }); |
| D-Delete | DELETE FROM tabla WHERE condición; | db.coleccion.deleteOne({ campo: "valor" }); |
| Concepto aplicados | ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad), SQL estructurado, relaciones entre tablas con claves foráneas. | BASE (Basic Availability, Soft-state, Eventual Consistency), JSON/BSON, documentos no estructurados o semiestructurados |