

**Nombre: Alan Navia**

**Codigo: 6780**

## **DESARROLLAR LA CAPA DE ACCESO A DATOS DEL PROYECTO DE CONCILIACIÓN BANCARIA.**

### **1. Introducción**

Este informe documenta el proceso de configuración e implementación del backend para el sistema de conciliación bancaria. Se utilizó **Node.js con Express** para la capa de acceso a datos y **MySQL en XAMPP** como gestor de base de datos.

### **2. Requisitos Previos**

- Tener instalado **Node.js** y **npm**.
- XAMPP en funcionamiento con el servicio de MySQL activado.
- Visual Studio Code como editor de código.
- Postman o navegador para pruebas de API.

### **3. Creación de la Base de Datos**

Se creó la base de datos en **phpMyAdmin** dentro de XAMPP ejecutando el siguiente script SQL:

```
CREATE DATABASE conciliacion_bancaria;
USE conciliacion_bancaria;

CREATE TABLE Usuarios (
    id_usuario INT AUTO_INCREMENT PRIMARY KEY,
    nombre VARCHAR(50),
    apellido VARCHAR(50),
    email VARCHAR(100) UNIQUE,
    contrasena VARCHAR(255),
    id_rol INT,
    FOREIGN KEY (id_rol) REFERENCES Roles(id_rol)
);

CREATE TABLE Roles (
    id_rol INT AUTO_INCREMENT PRIMARY KEY,
    nombre VARCHAR(50)
);
```

(Omitido el resto de las tablas por brevedad, pero están incluidas en la implementación final).

### **4. Creación del Backend en Node.js con Express**

- 1. Inicialización del Proyecto**
  - Se creó una carpeta llamada `backend`.
  - Se inicializó el proyecto con:

```
npm init -y
```

## 2. Instalación de Dependencias

```
npm install express mysql2 cors dotenv nodemon
```

## 3. Configuración del Servidor en `index.js`

Se creó el archivo `index.js` y se agregó el siguiente código:

```
4. import express from "express";
5. import cors from "cors";
6. import mysql from "mysql2";
7. import dotenv from "dotenv";
8.
9. dotenv.config();
10. const app = express();
11. app.use(cors());
12. app.use(express.json());
13.
14. const db = mysql.createConnection({
15.   host: "localhost",
16.   user: "root",
17.   password: "",
18.   database: "conciliacion_bancaria"
19. });
20.
21. db.connect((err) => {
22.   if (err) {
23.     console.error("Error conectando a la base de datos:", err);
24.     return;
25.   }
26.   console.log("Conexión exitosa a la base de datos");
27. });
28.
29. app.get("/", (req, res) => {
30.   res.send("Servidor funcionando correctamente");
31. });
32.
33. const PORT = process.env.PORT || 4000;
34. app.listen(PORT, () => {
35.   console.log(`Servidor corriendo en
    http://localhost:${PORT}`);
36. });
```

## 36. Configuración de `package.json`

Se agregaron los scripts:

```
37. "scripts": {
38.   "start": "node index.js",
39.   "dev": "nodemon index.js"
40. }
```

## 5. Ejecución del Servidor

Para ejecutar el servidor:

```
npm run dev
```

Si la configuración es correcta, la terminal mostrará:

Servidor corriendo en `http://localhost:4000`  
Conexión exitosa a la base de datos

## 6. Pruebas de Conectividad

Para verificar que el backend está funcionando:

1. Abrir un navegador y acceder a `http://localhost:4000/`.
2. Debería mostrar el mensaje `Servidor funcionando correctamente`.
3. También se puede probar con **Postman**.

## 7. Enlace al Proyecto

<https://github.com/Alan290989/conciliacion-bancaria>

---

Este informe detalla los pasos para la implementación del backend con Express, MySQL y XAMPP. Si hay errores o ajustes necesarios, se pueden revisar los logs de la terminal o los mensajes de error devueltos por el servidor.