**GITHUB INSTALLATION TO VISUAL STUDIO:**

Create A file where you will store this project on your pc

After you created a file or folder, open that folder on Visual Studio

**Type this in terminal**

**git clone** [**https://github.com/DanieruuDev/Metrobank-STRONG-Program-Management-System.git**](https://github.com/DanieruuDev/Metrobank-STRONG-Program-Management-System.git)  
  
Once installation is done move to the client directory by typing in the terminal

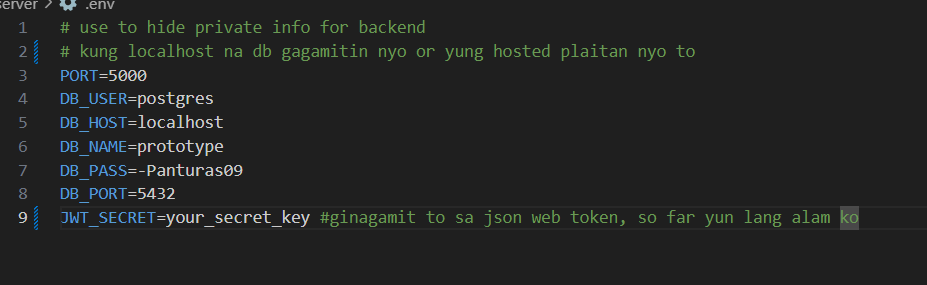
**cd client**

then type **npm i** to initialize the react then do it to the serve side too

**NOTE ON .env:**

**.env** is yung file na ilalagya natin ang sensitive info like api keys and other things. Upon cloning hindi sya masasama once na napush kase protected, so you need to create it again.

Currently anf merong .env ay ang server so add file then create .env as name.

Sample in this pic, kung hosted or local nyo ang gagamitin palitan nyo info nito  


**Client Side Initialization:**

**Installing React:**

npm create vite@latest .

npm i

**Installing Tailwind:**

npm install tailwindcss @tailwindcss/vite

**Installing Dependencies:**

npm i axios jwt-decode lucide-react react-router-dom xlsx

* **axios** – A promise-based HTTP client for making API requests in the browser and Node.js. It simplifies sending GET, POST, PUT, and DELETE requests.
* **jwt-decode** – A library for decoding JSON Web Tokens (JWTs) to extract user information without needing a backend request.
* **lucide-react** – A collection of modern and customizable icons for React applications.
* **react-router-dom** – A routing library for React that enables navigation and dynamic page rendering in single-page applications (SPAs).
* **xlsx** – A library for reading, writing, and manipulating Excel (XLSX) files in JavaScript, useful for exporting and importing spreadsheet data.

**Client File Structure:**

 **src/main.tsx**

* The entry point of your application
* This is where React is initialized and your App component is rendered to the DOM

 **src/App.tsx**

* The root component of your application
* Usually contains the main layout and routing setup

 **src/pages/**

* This directory contains your page components
* You'll frequently add new pages or modify existing ones here
* Each file typically represents a different route in your application

 **src/components/**

* Contains reusable UI components
* You'll create and edit components here frequently

 **src/context/**

* Contains React context providers for state management
* Important when working with application-wide state

 **src/hooks/**

* Custom React hooks that encapsulate reusable logic
* You'll refer to these when implementing component behavior

 **src/utils/**

* Helper functions that are used throughout the application
* Important for common functionality

 **src/index.css**

* Global styles for your application
* Important for maintaining consistent styling

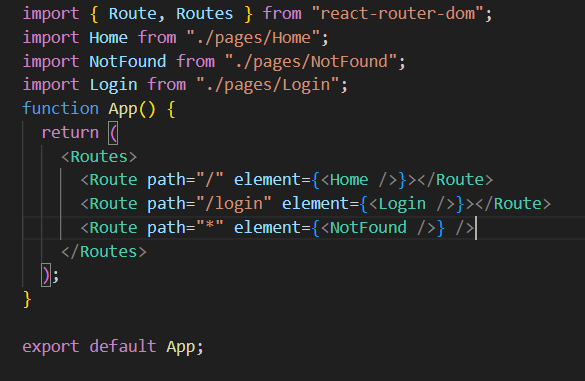
 **package.json**

* Defines your project dependencies and scripts
* You'll update this when adding new packages

 **vite.config.ts**

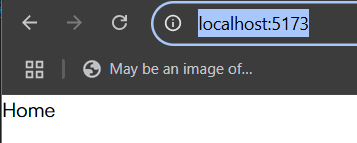
* Configuration for the Vite build tool
* Important when you need to adjust build settings or add plugins

**react-router-dom –** Stored in the src/App.tsx, use for routing page like anchor link

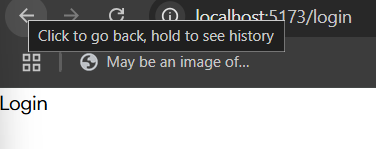


Usage of react-router-dom:

<http://localhost:5173/>



<http://localhost:5173/login>



**Server-Side Initialization:**

npm install express pg cors dotenv body-parser morgan bcrypt exceljs multer jsonwebtoken

npm install --save-dev nodemon

**npm init –** to initialize the node and express

**✅ Main Dependencies (Used in Production)**

| **Package** | **Description** |
| --- | --- |
| express | Web framework for building APIs. |
| pg | PostgreSQL client to interact with the database. |
| cors | Enables Cross-Origin Resource Sharing (for frontend-backend communication). |
| dotenv | Loads environment variables from a .env file. |
| body-parser | Parses incoming JSON and form data in requests. |
| bcrypt | Hashes and verifies passwords securely. |
| exceljs | Reads and writes Excel files (.xlsx). |
| multer | Handles file uploads (images, documents, etc.). |
| jsonwebtoken | Creates and verifies JSON Web Tokens (JWT) for authentication. |

**✅ Dev Dependencies (Used for Development)**

| **Package** | **Description** |
| --- | --- |
| nodemon | Automatically restarts the server when code changes (only for development). |

**How to run in server side:**

Punta muna kayo sa directory ng server side nyo:

**cd server** -> ito type sa terminal

**npm run dev** -> para mag run yung server mo

**cntrl c** para mag stop

**Server File Structure:**

 **📂 controllers/** → Handles business logic (e.g., userController.js).

 **📂 database/** → Manages database connections (db-connect.js).

 **📂 middlewares/** → Custom middlewares (e.g., authMiddleware.js).

 **📂 routes/** → Defines API routes (userRoutes.js).

 **📂 utils/** → Helper functions (hashPassword.js).

 **📄 .env** → Stores sensitive config (DB credentials, secrets).

 **📄 index.js** → Main entry file (starts Express, loads routes).

 **📄 package.json** → Project dependencies & scripts.