This document is meant for developers in the project.  
Please follow below steps to get started with the project and this document will be updated as we progress on the project.

1. **Cloning GIT repo ->**   
   *git clone https://github.com/madhunica-balasubramanian/EmployeePerfMonitoringSystem.git /Users/madhunica/Desktop/CMPE272/PROJECT/development --progress*

*cd /Users/madhunica/Desktop/CMPE272/PROJECT/development*

*git checkout development*

*Please create a feature branch for whatever feature you are working on and push to dev branch only if your feature is working end-end.*

1. **Create Structure in VS Code for frontend using react.js , backend using python, database setup using postgres.**

***BackEnd 🡪***

*cd /Users/madhunica/Desktop/CMPE272/PROJECT/EmployeePerfMonitoringSystem*

*mkdir -p backend/app/{models,routes,services,utils}*

*mkdir -p backend/tests*

*touch backend/app/\_\_init\_\_.py backend/app/config.py backend/app/main.py backend/requirements.txt*

*Structure*

*backend/*

*├── app/*

*│ ├── \_\_init\_\_.py*

*│ ├── models/*

*│ ├── routes/*

*│ ├── services/*

*│ └── utils/*

*├── tests/*

*├── config.py*

*├── requirements.txt*

*└── main.py*

***FrontEnd 🡪***

*mkdir -p frontend/src/{components,pages,services,utils}*

*touch frontend/src/{App.js,index.js}*

*mkdir frontend/public*

*touch frontend/package.json frontend/README.md*

*frontend/*

*├── public/*

*├── src/*

*│ ├── components/*

*│ ├── pages/*

*│ ├── services/*

*│ ├── utils/*

*│ ├── App.js*

*│ └── index.js*

*├── package.json*

*└── README.md*

***Database 🡪***

*mkdir -p database/migrations*

*touch database/init.sql*

*database/*

*├── migrations/*

*└── init.sql*

***Docker Setup 🡪***

*mkdir -p docker*

*touch docker/docker-compose.yml docker/Dockerfile.backend docker/Dockerfile.frontend*

*docker/*

*├── docker-compose.yml*

*├── Dockerfile.backend*

*└── Dockerfile.frontend*

touch .gitignore README.md , touch requirements.txt , cd backend , touch requirements.txt

1. **Setting up virtual environment and install dependencies**

*# Create and activate virtual environment*

python -m venv venv

source venv/bin/activate

pip install fastapi uvicorn psycopg2-binary sqlalchemy pydantic python-dotenv pytest

pip freeze > backend/requirements.txt

1. Running backend

Ensure you have the following lines in main.py (As its not abe to resolve app/ path)

***import sys***

***import os***

***sys.path.append(os.path.dirname(os.path.dirname(os.path.abspath(\_\_file\_\_))))***

**Run main.py from the VS CODE.**

You should see something like this.

*/Users/madhunica/Desktop/CMPE272/PROJECT/EmployeePerfMonitoringSystem/venv/lib/python3.11/site-packages/pydantic/\_internal/\_config.py:373: UserWarning: Valid config keys have changed in V2:*

*\* 'orm\_mode' has been renamed to 'from\_attributes'*

*warnings.warn(message, UserWarning)*

*INFO: Will watch for changes in these directories: ['/Users/madhunica/Desktop/CMPE272/PROJECT/EmployeePerfMonitoringSystem/backend/app']*

*INFO: Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)*

*INFO: Started reloader process [19667] using StatReload*

*/Users/madhunica/Desktop/CMPE272/PROJECT/EmployeePerfMonitoringSystem/venv/lib/python3.11/site-packages/pydantic/\_internal/\_config.py:373: UserWarning: Valid config keys have changed in V2:*

*\* 'orm\_mode' has been renamed to 'from\_attributes'*

*warnings.warn(message, UserWarning)*

*INFO: Started server process [19669]*

*INFO: Waiting for application startup.*

*INFO: Application startup complete.*

*/Users/madhunica/Desktop/CMPE272/PROJECT/EmployeePerfMonitoringSyste*

*m/venv/bin/python /Users/madhunica/Desktop/CMPE272/PROJECT/EmployeePerfMonitoringSystem/backend/app/main.py*

1. Go to frontend and install required dependencies

# Navigate to frontend directory

cd frontend

Install npm install react-router-dom axios

# Initialize React app with Create React App

npx create-react-app .

# Install necessary dependencies

npm install axios react-router-dom @mui/material @mui/icons-material @emotion/react @emotion/styled recharts jwt-decode

**Navigate to FrontEnd and run the react App**

*cd frontend/*

*npm install*

*npm start*

*You can now view frontend in the browser*

*Local: http://localhost:3000*

*On Your Network: http://10.0.0.100:3000*

Note: if you encounter node.js issue.  It typically occurs when you're using Node.js version 17 or later, which enforces stricter OpenSSL requirements.

If you don’t hve nvm installed,

curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.3/install.sh | bash

Always , do installations inside venv

rm -rf node\_modules package-lock.json

To remove and reinstall dependencies,

*rm -rf node\_modules package-lock.json*

*npm install*

Please follow the warnings and resolve any dependencies you have in VS code. The working set of package versions are as follows.

Once you are all set never delete package-lock.json

Let’s say you cloned or updated your repo.

*To reinstall****exact same versions****, run:*

*npm ci*

Always ensure, you commit these files.

*git add package.json package-lock.json*

*git commit -m "msg"*

To view current package version,

*npm list --depth=0*

*Sample snapshot,*

*(venv) madhunica@Madhunicas-MacBook-Air frontend % npm list --depth=0*

*frontend@0.1.0 /Users/madhunica/Desktop/CMPE272/PROJECT/EmployeePerfMonitoringSystem/frontend*

*├── @emotion/react@11.14.0*

*├── @emotion/styled@11.14.0*

*├── @mui/icons-material@7.0.2*

*├── @mui/material@7.0.2*

*├── @testing-library/dom@10.4.0*

*├── @testing-library/jest-dom@6.6.3*

*├── @testing-library/react@16.3.0*

*├── @testing-library/user-event@13.5.0*

*├── axios@1.8.4*

*├── core-js@3.41.0*

*├── jwt-decode@4.0.0*

*├── react-dom@19.1.0*

*├── react-router-dom@7.5.0*

*├── react-scripts@5.0.1*

*├── react@19.1.0*

*├── recharts@2.15.2*

*├── web-vitals@2.1.4*

*├── webpack-cli@6.0.1*

*└──* [*webpack@5.99.5*](mailto:webpack@5.99.5)

*(venv) madhunica@Madhunicas-MacBook-Air frontend % node -v*

*v22.14.0*

*(venv) madhunica@Madhunicas-MacBook-Air frontend % npm -v*

*10.9.2*

*(venv) madhunica@Madhunicas-MacBook-Air frontend %*

*This enforces node.js version.*

*echo "22.14.0" > .nvmrc*

*nvm use*

1. Please follow below steps to work with postgres sql

In this file config.py, you will find the database name

*DATABASE\_URL = os.getenv("DATABASE\_URL", "postgresql://postgres:postgres@localhost:5432/wellness\_db")*

*Breaking this down,*

*postgresql:// -> DB type (PostgreSQL)*

*postgres:postgres -> username:password*

*localhost:5432 -> host:port*

*wellness\_db -> database name*

*Note that, All API endpoints will be prefixed with this (e.g., /api/v1/users, /api/v1/login, etc.)*

**Now, How do we create this database ?**

1. Install postgres if not installed already.
2. in mac, brew services start postgresql@15

*(venv) madhunica@Madhunicas-MacBook-Air EmployeePerfMonitoringSystem %*

*(venv) madhunica@Madhunicas-MacBook-Air EmployeePerfMonitoringSystem % brew services restart postgresql@15*

*Stopping `postgresql@15`... (might take a while)*

*==> Successfully stopped `postgresql@15` (label: homebrew.mxcl.postgresql@15)*

*==> Successfully started `postgresql@15` (label:* [*homebrew.mxcl.postgresql@15*](mailto:homebrew.mxcl.postgresql@15)*)*

1. In terminal, access via psql postgres

*(venv) madhunica@Madhunicas-MacBook-Air EmployeePerfMonitoringSystem % psql postgres*

*psql (15.12 (Homebrew))*

*Type "help" for help.*

*postgres=# CREATE DATABASE wellness\_db;*

*CREATE DATABASE*

1. Confirm whether the URL works,

*psql -U postgres -h localhost -p 5432 wellness\_db*

*If prompted for a password, enter postgres (the one in your config.py)*

1. Add all this logic in docker-compose.yaml , Please refer that file for better understanding.