

Leave Management System

Technologies:

- Express JS for Backend
- Vite + React JS for Frontend
- MySQL for Database

Backend:

I have used **Express JS** to handle Backend functionalities.

Write **“npm start”** to run the Backend Server.

Dependencies:

```
"body-parser": "^1.20.2",  
"express": "^4.19.2",  
"mysql2": "^3.11.0",  
"nodemon": "^3.1.4"
```

I have maintained Routes for routing and controllers to handle the functionalities of the routes so that the code is cleaner.

Database: (import db to use)

I have used MySQL database. There are 4 tables in the DB:

	Table ▼
<input type="checkbox"/>	admin
<input type="checkbox"/>	employee
<input type="checkbox"/>	history
<input type="checkbox"/>	request

“admin” table is used for Admin Login,

“employee” table is used for Employee Login,

“request” table is used for submitting and handling leave requests and

“history” table is used to store leave history for every employee.

Key feature:

- *One employee cannot make another leave request before resolving the current request.
- *Every leave request is given an unique ID to monitor leave days for one employee.

Frontend:

I have used Vite + React JS for Frontend.

Write “**npm run dev**” to run the UI.

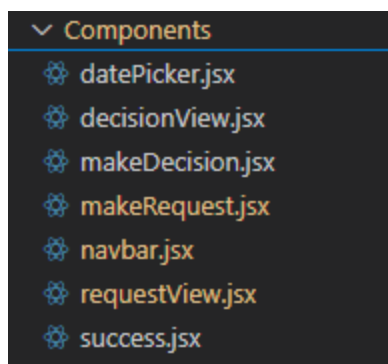
Dependencies:

```
"bootstrap": "^5.3.3",  
"react": "^18.3.1",  
"react-bootstrap": "^2.10.4",  
"react-datepicker": "^7.3.0",  
"react-dom": "^18.3.1",  
"react-hot-toast": "^2.4.1",  
"react-router-dom": "^6.26.1"
```

I have used **Bootstrap React** for styling. Also used “**react-router-dom**” for routing.

Key features:

- *I have used several “Components” to make the code clean and reusable-



- *I have used the “react-datepicker” to take input for the dates in a modern fashion. Also, used hooks to maintain functionality like; past dates cannot be used for more efficient input handling.

Issues:

I could not connect the frontend with the backend.

Future Updates:

- Integrate frontend with backend to enable all the functionalities.
- View and update profile for users.
- Delete employee from Admin dash board.
- More efficient and stable UI.

The End