# LICET -  Application Documentation

## Table of Contents

1. Introduction
2. Backend
    - Technologies Usedtechnologies-used
    - Project Structureproject-structure
    - Database Setupdatabase-setup
    - API Endpointsapi-endpoints
3. Frontend
    - Technologies Usedtechnologies-used-frontend
    - Project Structureproject-structure-frontend
    - Componentscomponents
    - Routingrouting
4. Usage
    - Setting Up the Backendsetting-up-the-backend
    - Setting Up the Databasesetting-up-the-database
    - Setting Up the Frontendsetting-up-the-frontend
5. Future Improvements
6. Conclusion

## 1. Introduction

This documentation provides an overview of a web application consisting of a backend built with Express.js and a frontend built with React. The application serves as a basic login system with different user roles and includes functionality for retrieving attendance data, activity assignments, and more.

## 2. Backend

Technologies Used

- Node.js
- Express.js
- MySQL or any other relational database
- Axios for making HTTP requests

Project Structure

The backend project structure follows a modular pattern:

- models: Contains modules for different functionalities e.g., login, attendance, activity_assignment.
- routes: Defines Express routes that use the modules from the models.
- index.js: Entry point for the server, sets up Express, middleware, and listens on a specified port.

Database Setup

1. Set up a MySQL database.

2. Configure the database connection in `models/db.js`.

API Endpoints

1. POST /login
   - Endpoint for user authentication.
   - Accepts registration number and date of birth.
   - Returns a success message and user data if authentication is successful.

2. POST /attendance
   - Retrieves attendance data for a given registration number.

3. GET /activity-assignment
   - Retrieves activity assignment data.

4. POST /insertStudent
   - Inserts new student details into the database.

5. POST /updateAttendance
   - Updates attendance status based on the provided data.

6. POST /insertActivityAssignment
   - Inserts new activity assignment data into the database.

## 3. Frontend

Technologies Used Frontend

- React
- Axios for making HTTP requests
- React Router

Project Structure Frontend

- components: Contains React components for different pages e.g., LoginPage, HomePage, AdminHome.
- App.js: Entry point for the frontend, defines routes using React Router.

Components

1. LoginPage
   - Handles user authentication.
   - Sends a POST request to the backend with login data.
   - Redirects users to different pages based on their roles.

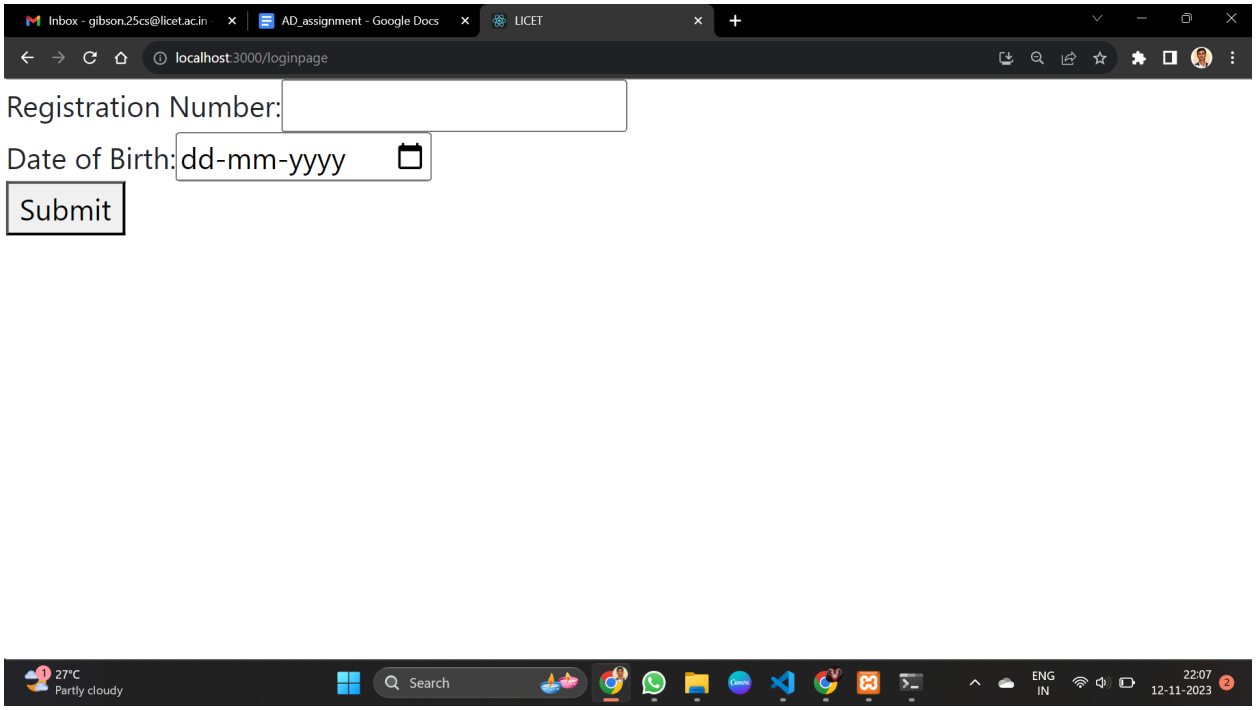
2. HomePage
   - Displays attendance data for the logged-in student.

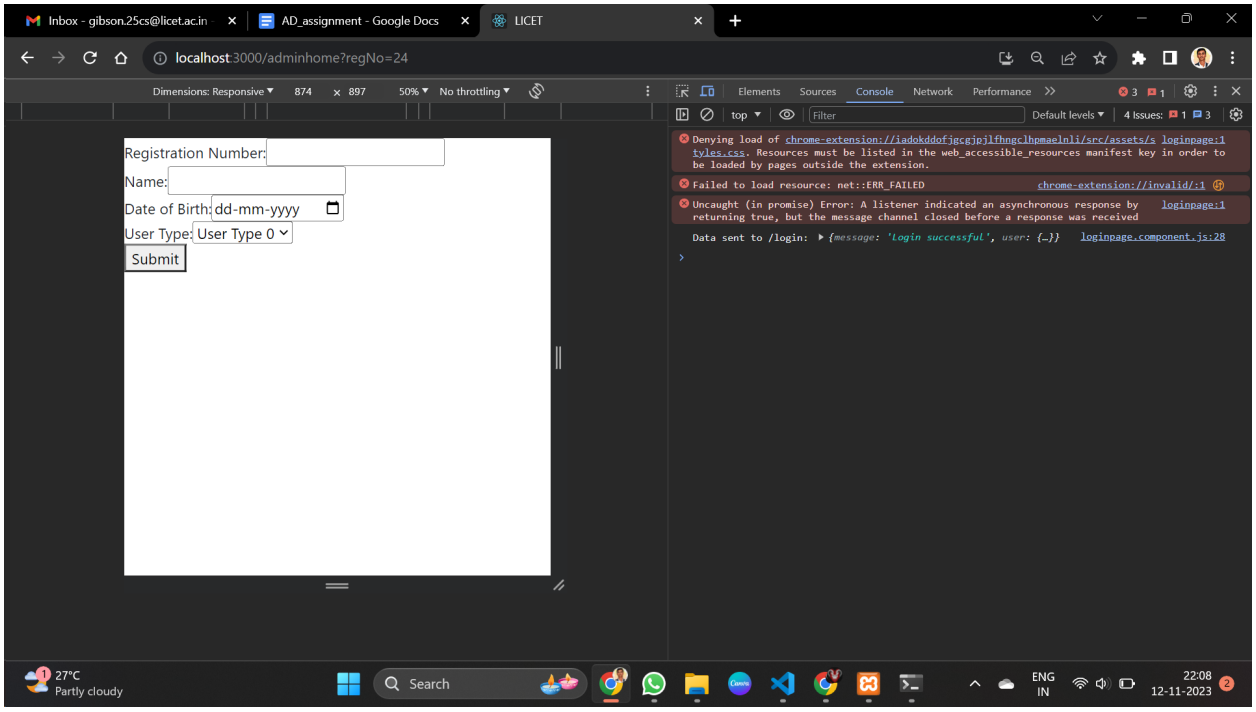
3. AdminHome
   - Displays activity assignment data for an admin.

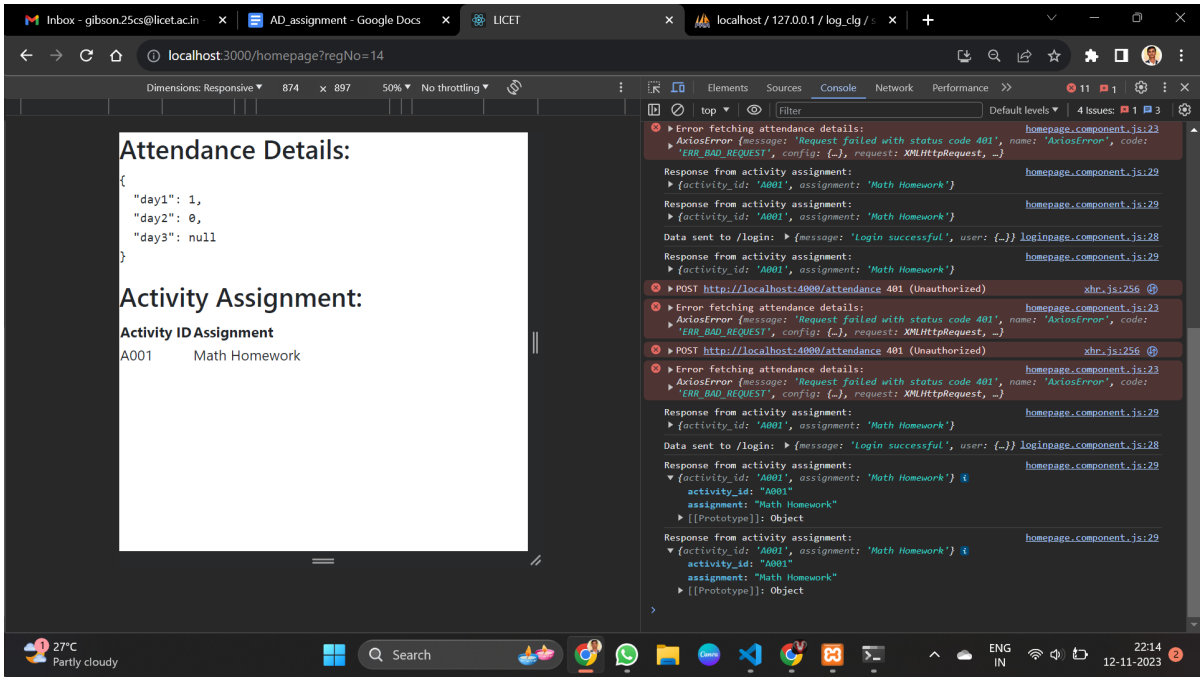
Routing

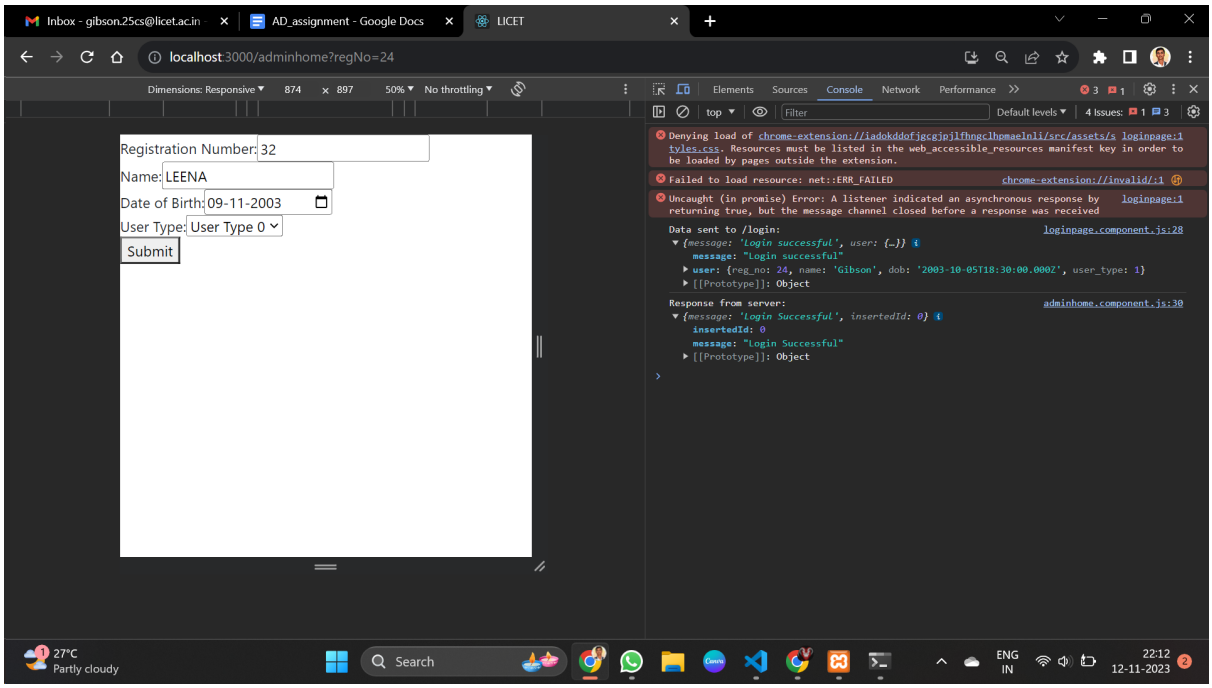
The frontend uses React Router for navigation:

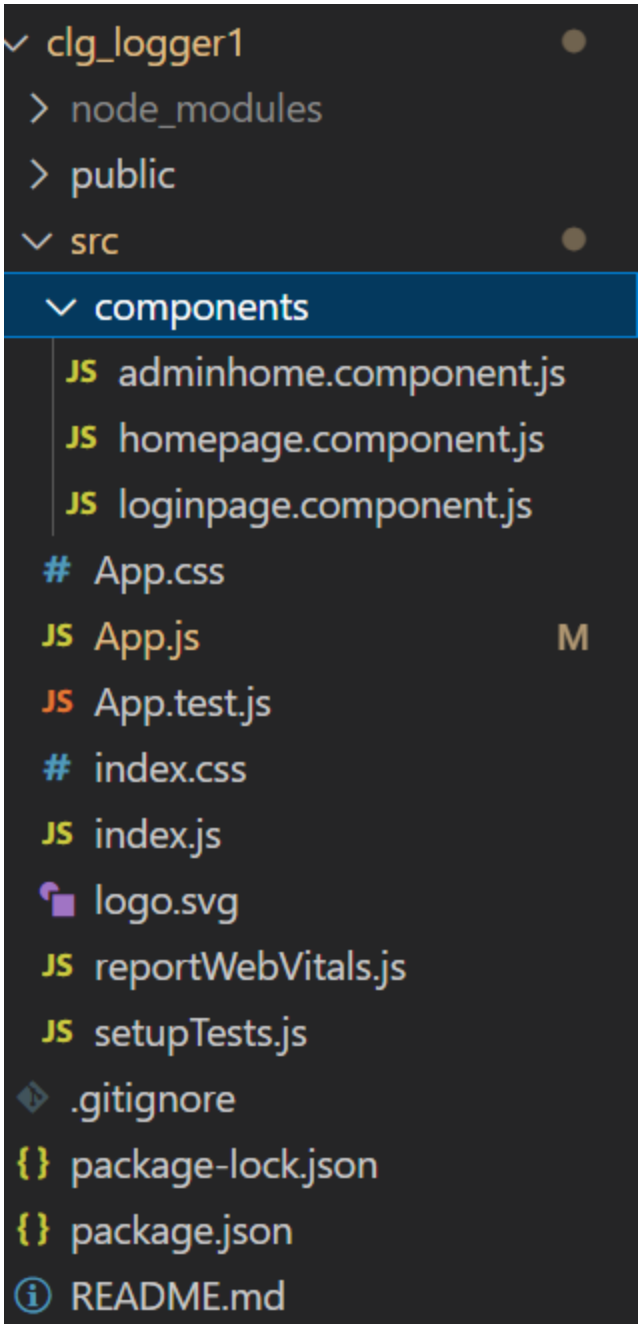
- `/loginpage`: Displays the login page.
- `/homepage`: Displays attendance data for a student.
- `/adminhome`: Displays activity assignment data for an admin.

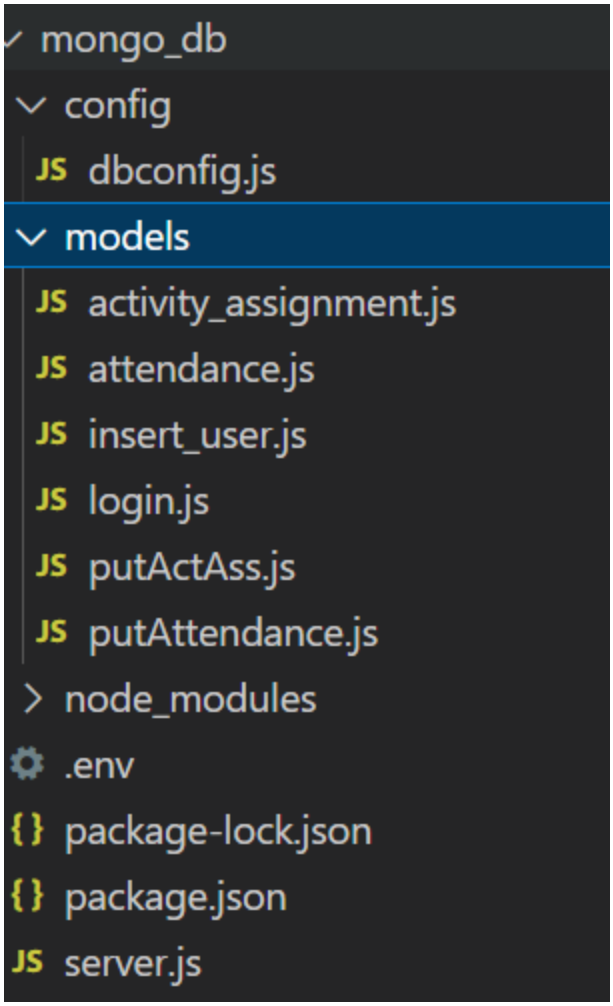

Inbox - gibson.25cs@licet.ac.in
AD_assignment - Google Docs
LICET
localhost:3000/loginpage
Registration Number:
Date of Birth: dd-mm-yyyy
Submit
27°C
Partly cloudy
Search
ENG
IN
22:07
12-11-2023


Inbox - gibson.25cs@licet.ac.in
AD_assignment - Google Docs
LICET
localhost:3000/adminhome?regNo=24
Dimensions: Responsive 874 x 897 50% No throttling
Registration Number:
Name:
Date of Birth: dd-mm-yyyy
User Type: User Type 0
Submit
Elements Sources Console Network Performance
Denying load of chrome-extension://iadokddofjgcgjpjlfhngclhpmaelnli/src/assets/styles.css. Resources must be listed in the web_accessible_resources manifest key in order to be loaded by pages outside the extension.
Failed to load resource: net::ERR_FAILED
Uncaught (in promise) Error: A listener indicated an asynchronous response by returning true, but the message channel closed before a response was received
Data sent to /login: {message: 'Login successful', user: {…}}
27°C
Partly cloudy
Search
ENG
IN
22:08
12-11-2023


Inbox - gibson.25cs@licet.ac.in
AD_assignment - Google Docs
LICET
localhost / 127.0.0.1 / log_clg /
localhost:3000/homepage?regNo=14
Attendance Details:
{
"day1": 1,
"day2": 0,
"day3": null
}
Activity Assignment:
Activity ID Assignment
A001 Math Homework
Elements Sources Console Network Performance
Error fetching attendance details:
AxiosError {message: 'Request failed with status code 401', name: 'AxiosError', code: 'ERR_BAD_REQUEST', config: {…}, request: XMLHttpRequest, …}
Response from activity assignment:
{activity_id: 'A001', assignment: 'Math Homework'}
Data sent to /login: {message: 'Login successful', user: {…}}
POST http://localhost:4000/attendance 401 (Unauthorized)
activity_id: "A001"
assignment: "Math Homework"
[[Prototype]]: Object
27°C
Partly cloudy
Search
ENG
IN
22:14
12-11-2023


Inbox - gibson.25cs@licet.ac.in
AD_assignment - Google Docs
LICET
localhost:3000/adminhome?regNo=24
Registration Number: 32
Name: LEENA
Date of Birth: 09-11-2003
User Type: User Type 0
Submit
Denying load of chrome-extension://iadokddofjgcgjpjlfhngclhpmaelnli/src/assets/styles.css. Resources must be listed in the web_accessible_resources manifest key in order to be loaded by pages outside the extension.
Failed to load resource: net::ERR_FAILED
Uncaught (in promise) Error: A listener indicated an asynchronous response by returning true, but the message channel closed before a response was received
Data sent to /login:
{message: 'Login successful', user: {…}}
message: "Login successful"
user: {reg_no: 24, name: 'Gibson', dob: '2003-10-05T18:30:00.000Z', user_type: 1}
[[Prototype]]: Object
Response from server:
{message: 'Login Successful', insertedId: 0}
insertedId: 0
message: "Login Successful"
[[Prototype]]: Object
27°C
Partly cloudy
Search
ENG
IN
22:12
12-11-2023


clg_logger1
node_modules
public
src
components
adminhome.component.js
homepage.component.js
loginpage.component.js
App.css
App.js
M
App.test.js
index.css
index.js
logo.svg
reportWebVitals.js
setupTests.js
.gitignore
package-lock.json
package.json
README.md


mongo_db
config
dbconfig.js
models
activity_assignment.js
attendance.js
insert_user.js
login.js
putActAss.js
putAttendance.js
node_modules
.env
package-lock.json
package.json
server.js