IT-PLUS 24

GeoLocator Application with Email Functionality and Database Integration.

Description

Create a basic full-stack application where users can enter an address in a React.js frontend. The Node.js backend should first check if the address's geolocation data is in the database. If not, fetch it from a third-party API, store it in the database, and then return it to the frontend. Include an option to email the geolocation results in the frontend.

Requirements:

Frontend (React.js):

- User Interface: Simple interface for inputting an address and an email address.
- Results Display: Display the geolocation results on the webpage.
- Email Option: Include a button to send the geolocation results to the provided email.
- Communicate with Backend: Send address and email details to the Node.js backend.
- Error Handling: Basic user feedback for errors.

Backend (Node.js):

- API Endpoint: Endpoint to handle requests from the frontend.
- Database Check: Look up the address in the database; if not found, use a third-party API to get geolocation data.
- Email Functionality: Implement a feature to email the results using a service like Nodemailer.
- Error Handling: Basic error communication to the frontend.

Database:

- Data Storage: Store addresses with their geolocation data.
- Efficient Searching: Check if an address is already in the database before using the API.

Deployment:

- Live Deployment: Deploy the application to a cloud service.
- Accessible URL: Provide a live URL for the application.

Repository and Documentation:

- GitHub Repository: Add the evaluator as a collaborator.
- README: Instructions for setup and deployment.

Evaluation Criteria:

- Functionality: Complete and functioning as described.
- Code Quality: Organized and readable code.
- UI/UX: Basic but user-friendly interface.
- Database and Email Integration: Efficient database usage and working email feature.
- Deployment: Successful deployment and accessibility.

Submission Guidelines:

- The GitHub repository link and the live application URL.
- The evaluator ('spaesey') should be added as a collaborator on GitHub.

Time Frame:

Complete within 1 day of receipt.