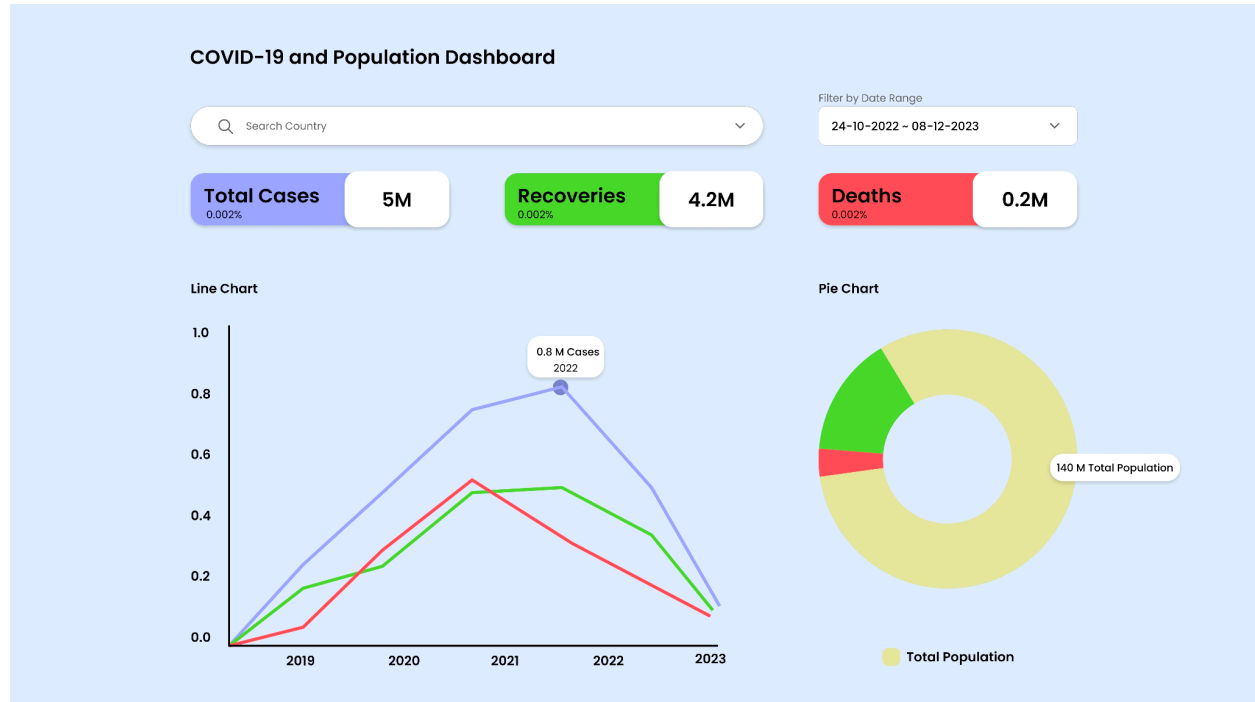


Frontend React JS Developer Assignment`



Objective

Create a COVID-19 dashboard using React.js, which visualizes statistical data through interactive charts and conveys pandemic statistics, including total cases, recoveries, and deaths as per the provided Mockup.

Requirements

API Integration

You will be integrating two RESTful APIs to fetch necessary data for the dashboard.

COVID-19 Historical Data:

- Fetch historical COVID-19 data for a default country (USA) and allow the user to select different countries from a dropdown menu.
- Use the following cURL request to understand the API endpoint:

- Method: `GET`
- Endpoint:
`https://disease.sh/v3/covid-19/historical/{country}?lastdays=1500`
- Replace `{country}` with the chosen country from the dropdown. Make sure to send the ISO code of the country. Example: `usa`
- The `lastdays=1500` parameter is hardcoded to fetch data for the last 1500 days.
- The response will be in JSON format containing the historical case data.

Country Options for Dropdown:

- To populate the dropdown with country options, integrate a REST API that provides a list of countries.
- Suggested API endpoint:
 - Endpoint: `GET https://restcountries.com/v3.1/all`
 - This endpoint does not require any parameters.
 - It will return a list of countries along with various details, from which you can extract the country names and codes for the dropdown.

Other Requirements

- Implement React components such as statistical cards, line chart, and pie chart.
- Ensure application state is managed effectively using React's state management features.
- The dashboard must be responsive and work on a variety of screen sizes.
- Apply error handling for all API requests.
- Write clean, modular, and well-commented code.
- Style the application to match the provided mockup as closely as possible.
- Use Git for version control and include a README.md with project setup instructions.

Deliverables

- A public GitHub repository with all source code.
- A live demo is optional but encouraged.

Evaluation Criteria

- Adherence to the provided mockup and functionality.
- Code quality and organization.
- Responsiveness and usability across devices.
- Performance and efficiency of the application.

Submission

Submit your completed assignment by **15th March** via one of the following methods:

- Provide a link to your public GitHub repository.
- Send a ZIP file containing the full project source code.