**Project Documentation: React Frontend for 430LProject**

This project is a React-based frontend for the 430LProject. It is designed to provide a user-friendly interface for the backend services and data provided by the project. This documentation aims to provide a comprehensive overview of the project, including its structure, installation, and usage.

**Table of Contents**

1. Introduction
2. Project Structure
3. Prerequisites
4. Installation
5. Usage
6. Component Overview
7. Conclusion

**1. Introduction**

The 430LProject is a web application that aims to demonstrate the functionality of the backend services through a React-based frontend. This frontend provides a user interface to interact with the backend services, allowing users to access data and perform various actions.

**2. Project Structure**

Key directories and files in the project:

* **public/**: Contains static files, such as **index.html**, which serves as the template for the React application.
* **src/**: Contains the source code for the React application, including components, styles, and the main entry point (**index.js**).
* **components/**: Holds individual React components that make up the application.
* **App.js**: The main application component that ties together all other components.
* **index.js**: The main entry point for the React application.
* **package.json**: Contains project metadata, dependencies, and scripts.

**3. Prerequisites**

Before installing and running this project, ensure that you have the following installed on your machine:

* Node.js (>= 14.x.x)
* npm (>= 6.x.x)

**4. Installation**

To install the dependencies for this project, follow these steps:

1. Clone the repository:

bashCopy code

git clone https://github.com/michael-ferzli/430LProject.git

1. Change to the **react** directory:

bashCopy code

cd 430LProject/react

1. Install the dependencies:

bashCopy code

npm install

**5. Usage**

To run the project in development mode, execute the following command in the **react** directory:

bashCopy code

npm start

This will start a development server at **http://localhost:3000**. Any changes made to the source code will automatically reload the application.

To build the project for production, run:

bashCopy code

npm run build

This will create an optimized build in the **build/** directory.

**6. Component Overview**

This section provides a brief overview of the main components in the **src/components/** directory:

* **Header.js**: A reusable header component that can be placed at the top of the application.
* **Footer.js**: A reusable footer component that can be placed at the bottom of the application.
* **DataView.js**: A component responsible for fetching and displaying data from the backend services.
* **ActionButton.js**: A component that represents a button used to perform actions, such as creating, updating, or deleting data.
* New implementation

In additional to the previous implementation, this React application now allows users to perform exchange transactions, view exchange rates, statistics, and participate in a forum. The application consists of the following components:

* ExchangeRateGraph:

Displays a line chart of the average exchange rate over time. Users can choose to view data for the last day, month, year, or all-time.

* Forum:

Allows users to post messages and view the last 5 messages in the forum.

* ExchangeRateStats:

Displays exchange rate statistics such as maximum, minimum, and average exchange rates for different time periods (last day, week, month, and year).

* UserTransactions:

Allows users to perform exchange transactions by posting offers and accepting available offers.

* MiniGame:

A small game where the user is asked to compute some calculation related to the exchange rate and amount indicated with the type of exchange also stated.

**7. Conclusion**

The 430LProject's backend services can be accessed through this React frontend, which offers a user-friendly interface. Simple maintenance and the addition of new features are made possible by the modular component structure. The project can be rapidly set up and used by adhering to the installation and usage guidelines provided in this documentation. Understanding the structure of the program and how the many components work together to provide a smooth user experience may be done by starting with the component overview.

It's crucial to keep the documentation current as the project develops in order to guarantee easy onboarding for new contributors and preserve clarity for current team members. The functionality and usefulness of the program can be improved overall by adding new elements, features, or enhancements in the future.

In conclusion, the 430LProject's React frontend is a crucial tool for connecting with the backend services and gives users an aesthetically pleasing and useful interface. This documentation ought to act as an exhaustive how-to manual for beginning the project and comprehending its components.