MintTrack

Language: TypeScript, HTML5, CSS, SQL

Project Summary: It should be a 1-2 paragraph description of what your project is.

Description:

The project aims to build a web application that allows users to track shared expenses, split bills among group members, and send notifications for payment reminders. Users can log transactions, divide costs based on custom rules, and notify members about pending payments. The system offers a detailed overview of users' spending habits. Our application will integrate secure user authentication and allow for easy exporting of financial reports.

The key challenge we aim to solve is managing complex group expense scenarios, especially custom proportions, and keeping users updated with their current balances and obligations. Our app will also offer detailed breakdowns of all spending activities, along with export features for users to generate reports of their expenses.

Creative Component:
Users generate data visualizations on their monthly expenditure, group spending, and debt repayments. The app will also feature smart notifications, triggered by conditions like overdue payments or high spending alerts.

Usefulness:
While there are other applications that perform expense tracking, ours will stand out with its enhanced data analysis features, greater flexibility in payment management, and personalized notifications.

Realness:
We store user-generated transaction data in a relational database (PostgreSQL) to support a multi-user environment.

Data sources:

1. **User Transaction Data**: Generated and stored as part of the application, stored in PostgreSQL for tracking user spending.
2. **External Currency Exchange Rates API:** Integrate a service that provides up-to-date exchange rates for various currencies. This would be useful if users are managing expenses in different currencies. By fetching real-time exchange rates, the app can convert expenses into a single currency for consistent tracking and reporting.
3. **Public Financial Data API:** Integrate a service that provides general financial data or economic indicators, such as inflation rates, average income statistics, or cost of living indices. This could help users contextualize their spending by comparing it to broader economic trends or benchmarks.

Functionality:

A list of entities that the database manages:

- User
- Group
- Spending
- Transaction
- Category

A user can form/join a group, and a group consists of one or more users.

Each group represents one tracking account, and it holds a list of spending transaction (can be categorized via category entity). For each payment split, there will be a record inside the database (Transaction) to track its complete status.

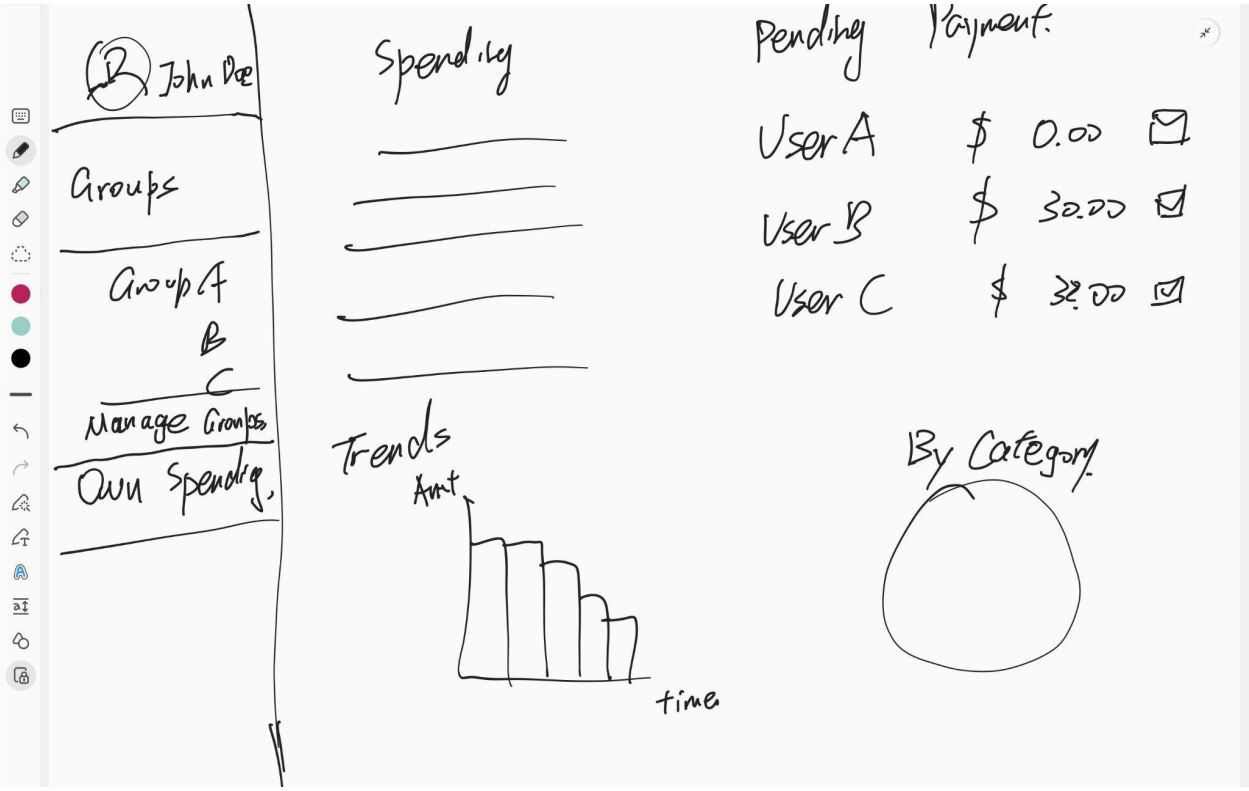
Our application will provide the following CRUD functionalities:

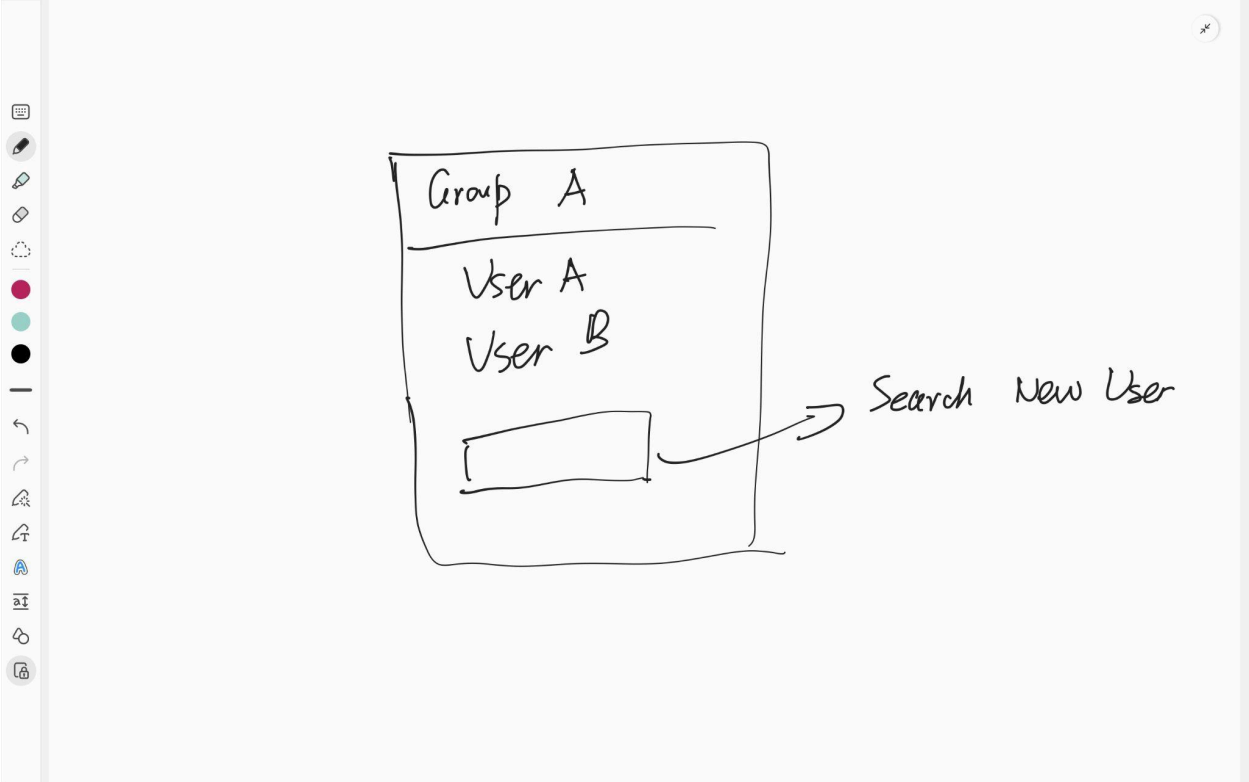
- **Create**: Users can create an account, set up groups, and input expenses.
- **Update**: Users can edit transactions, update their profile, and change bill-splitting preferences.
- **Search**: Users can search for transactions by date, group, or category.
- **Delete**: Users can delete expenses or leave groups.

Users will interact with the app primarily through a dashboard that displays an overview of spending and pending payments. They can filter and sort expenses, view a history of group transactions, and track payment status. Notifications will be sent via email and in-app alerts when payments are due or expenses are added.

Low-fidelity UI Mockup:
The app's UI will consist of a dashboard displaying:

1. Total spending
2. Group-wise transaction summary
3. Individual outstanding payments
4. A "Split Bill" button that allows users to quickly divide a new expense among group members
5. Notifications center with reminders and alerts





Project Work Distribution:

- **Backend Development (User Management, Expense Tracking)**: Qian, Lin, Cui, Huang
- **Frontend UI/UX (Dashboard, Forms, Notifications)**: Huang, Lin
- **Data Integration/ Database Management (APIs, Reporting)**: Qian, Lin, Cui, Huang
- **Testing, Documentation, and Deployment (CI/CD setup, Final integration)**: Huang, Qian, Cui

Backend systems will be distributed with each member responsible for different service components like authentication, transaction management, and notifications.