Froogal: Finance Tracking Application

Chris Adamson, Dylan Commean, Johnathan Dickson

Dept. Of Computer Science and Information Technology

Abstract

Financial literacy is the cornerstone of wealth building. Budgeting is a powerful tool in one's financial journey; however, it alone is not sufficient. An analysis on one's spending habits is a key metric that reflects who the consumer is at their core. The process of analyzing spending can be a daunting task; where does one even start? Froogal aims to streamline this process by providing rich financial reports about one's spending, over various periods (day, weeks, months, and years), so our users can have a deeper understanding of their spending habits. This gives our users the upper hand in controlling their personal finances, thus combatting overspending.

Purpose

Froogal is the all-in-one solution to reviving your personal finances. Budgeting, albeit a powerful tool, is not sufficient by itself for controlling finances. The reason being is that it merely suggests approximated spending targets but does not address the root problem of finances: poor spending habits. It does not matter how much money one makes if they spend more than what is coming in. Froogal aims to highlight a user's spending habits by providing rich, data-driven models which pinpoint a user's spending based on certain categories. Additionally, Froogal attempts to consolidate a user's expenses such as bills and subscriptions in one place. Keeping all this data in a centralized dashboard further gives users a high-level overview of upcoming expenses without switching between multiple apps, thus allowing them to monitor more than just spending habits.

Technology

Being a reactive based web-application that needs to store and query user data, our tech stack includes various technologies. React handles the frontend by providing the ability to create a reactive based Single Page Application (SPA). ChakraUI is responsible for styling. It was chosen to leverage best practices in design knowledge and accessibility. The library provides predefined components which can be fully customized or used out of the box based on design requirements.

Froogal's backend and database is built on top of the Firebase platform. The platform offers several key services that are integral to the app. These services include Authentication and Firestore. Firestore is a No Structured Query Language (NoSQL)-like cloud database allows us to store our user records and financial information. Additionally, the database is protected by Security Rules which dictates the access level and privileges a user has for the database. Authentication enables secure authentication and various login methods (e.g., Google, email/password, and more). Firebase Hosting handles our need of serving the initial SPA as well as other assistance pages (e.g., 404, user login). See **Figure 1** to see the system from a high-level view. See **Figure 2** to see the data we capture.

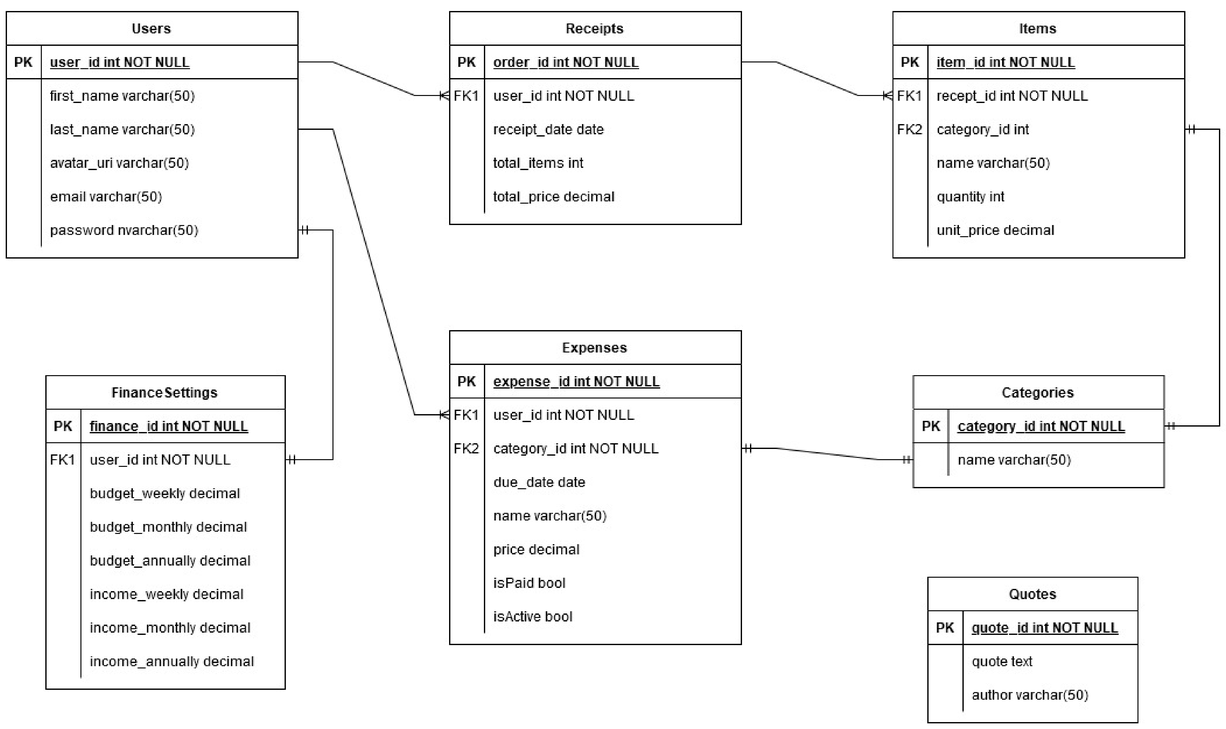

Figure 1: Entity Relational Diagram (ERD) of our system. Although our system uses noSQL, this assisted in data relationships

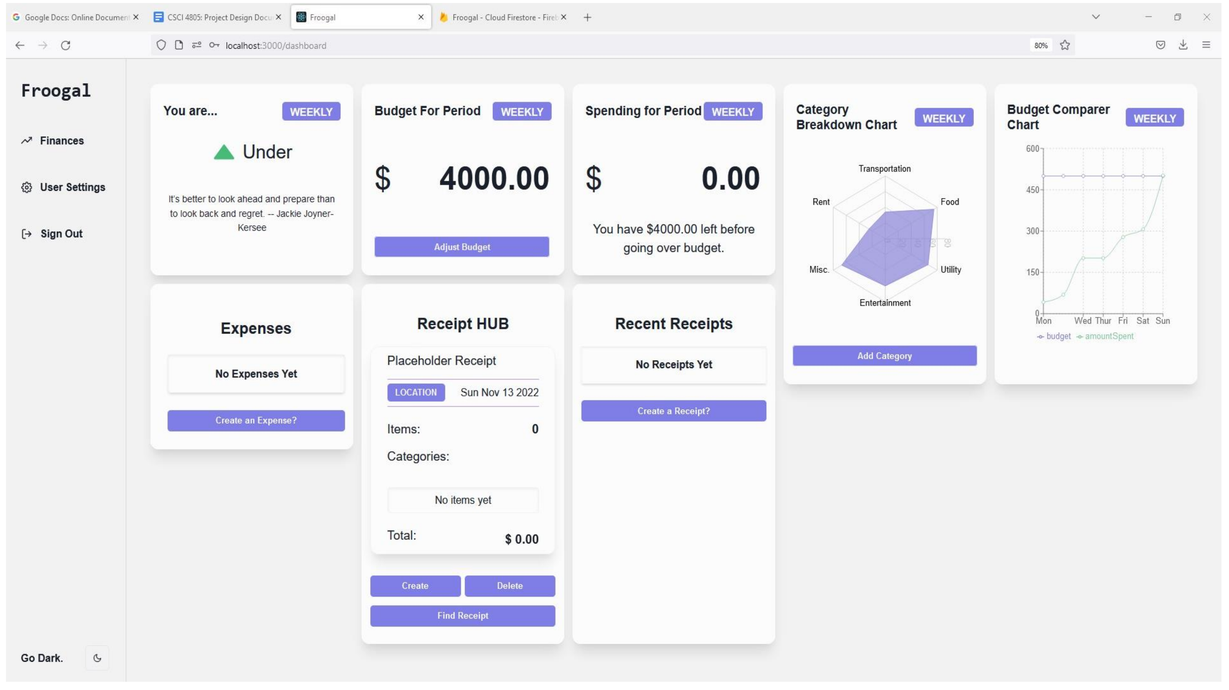

Figure 4: Main dashboard after login/signup. Data will be rendered based on user financial information.

Design

Froogal's UI aims to provide a modern component-based dashboard. Being a reactive application, the dashboard will render information according to the user's profile. The data is pulled from a collection of receipts, thus enabling the user to analyze their spending habits accurately over time. The dashboard also provides data visualization to help turn the abstract concept of spending into a concrete visual, highlighting the areas where a user allocates their money most. Additionally, a user can view their spending over various timeframes such as weekly, monthly, and annually. These views can be toggled in the Finance Settings drawer (see **Figure 3**).

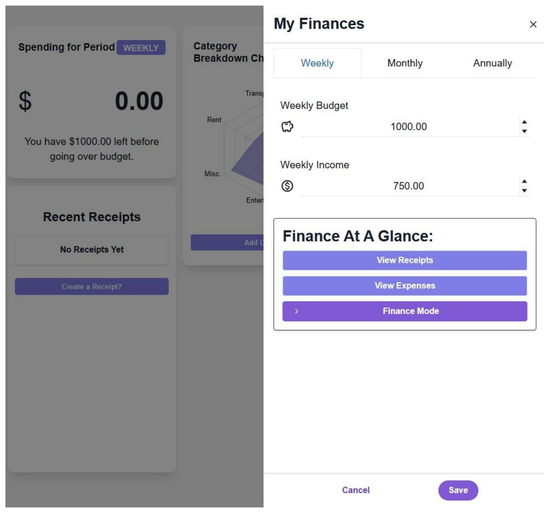

Figure 3: Snapshot of user Financial Settings drawer which controls displaying the dashboard in various modes and handles setting budgets and income.

Future Work

Closing in on phase one of Froogal's development, the team's goal has been to release a functional Minimum Viable Product (MVP). The main focuses of the MVP have been architecting the app, creating documentation, structuring the database, establishing correct database calls via Firebase API , and creating a concrete UI that displays all the correct user information and financial insights.

The next development phase for Froogal is heavily focused on UI and UX. This includes making sure the app is responsive for smaller devices (e.g., mobile and tablets), accessibility standards are adhered to based on Web Content Accessibility Guidelines (WCAG) and collecting feedback about the UI and considering implementing more functionality.

Acknowledgements

We would like to thank Dr. Nicholson for his support of students in the College of Science, Technology, Engineering & Mathematics, and Dr. Leong Lee for his support of students in the Department of Computer Science and Information Technology.

References

1. ReactJS - https://reactjs.org/
2. Chakra UI - https://chakra-ui.com/
3. Recharts - https://recharts.org/en-US/
4. Firebase - https://firebase.google.com/