Design Document

App Architecture

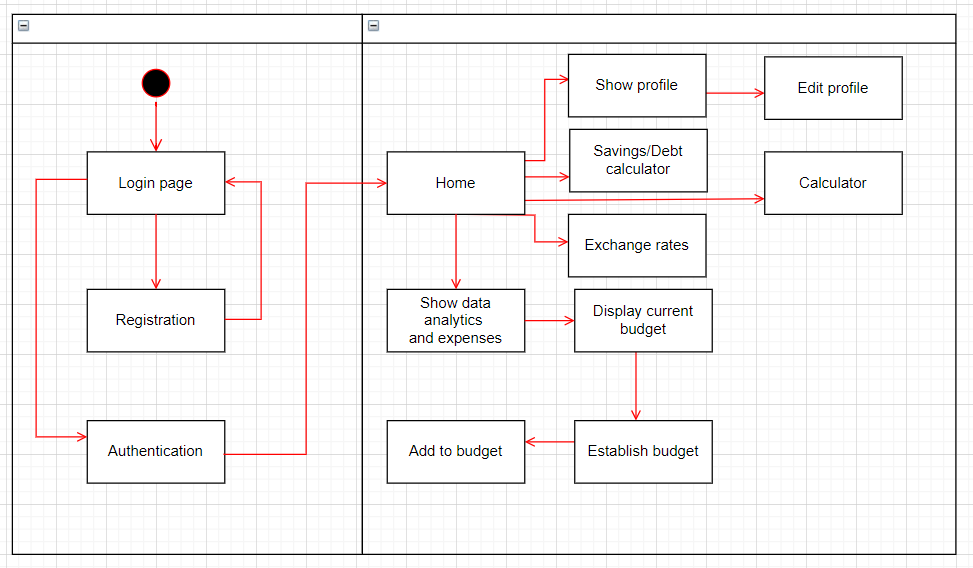
Web-based application using React and Firebase.

Data Model

User accounts will be stored in a database attached to their information such as their profile, desired budget, and expenses.

UML Chart

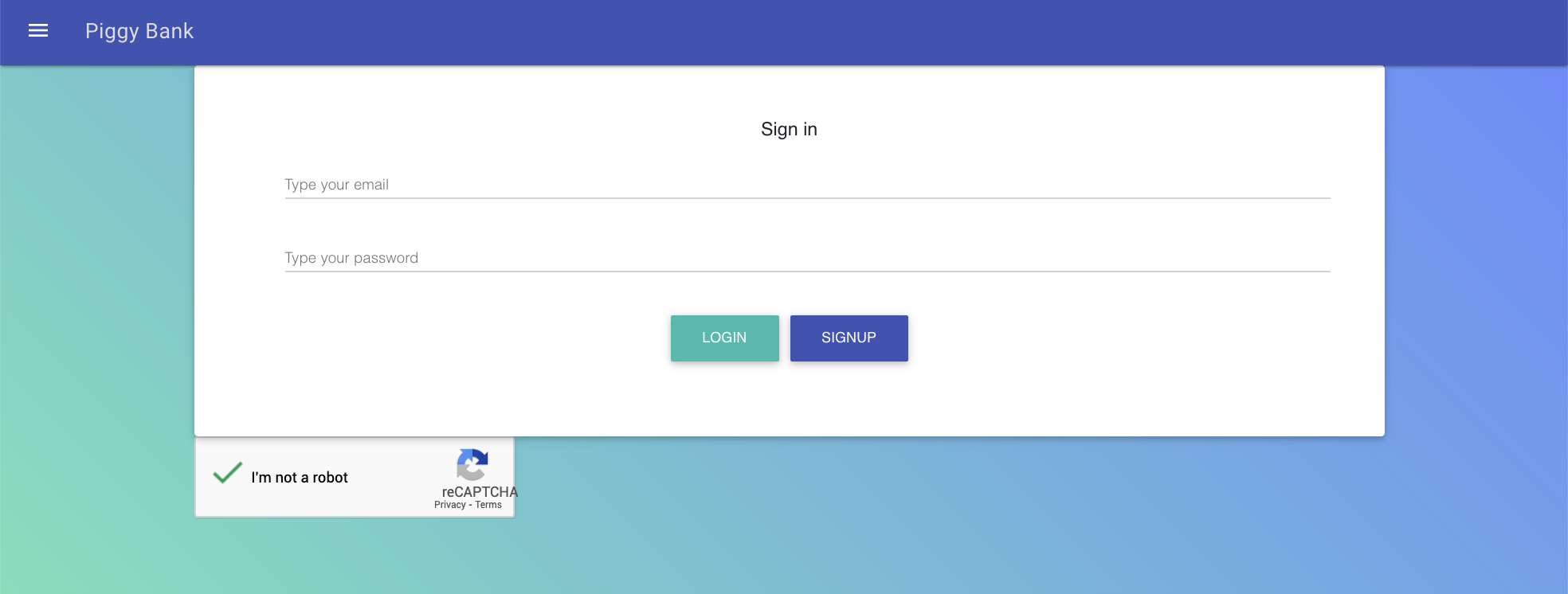
User first encounters login page before going into the home page



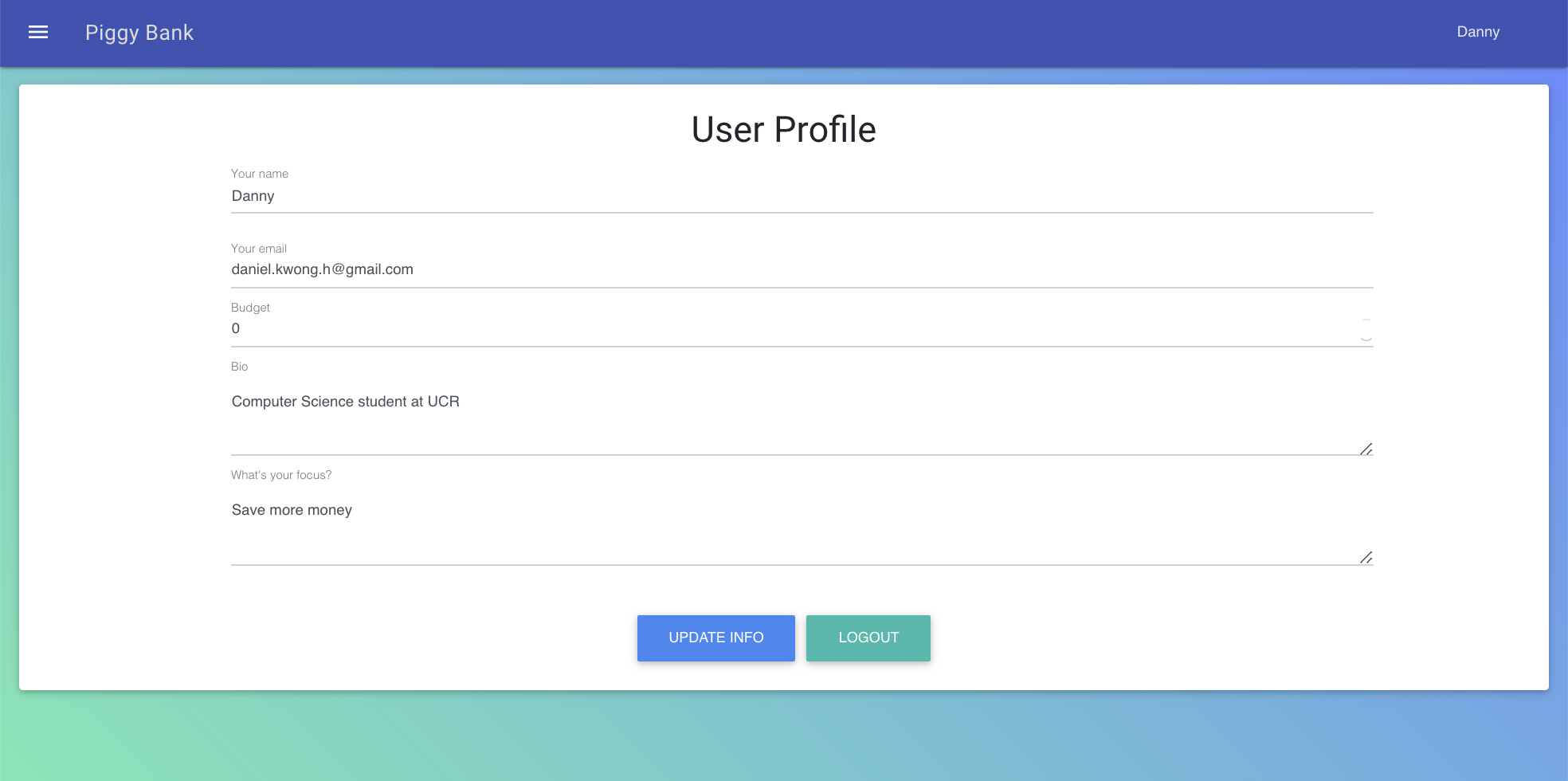
UI:

**UI** is designed in such a way that related things are combined or close to each other and unrelated things are separated. In the case of a mistake, the system displays an error message. Through a system of messages, the design keeps users informed of actions, errors, or other feedback. We use the same names to perform the same operations with different objects in order to reduce ambiguity and redundancy.

**Login/Sign-up Page:** The user is prompted to enter login credentials or sign-up for a new account

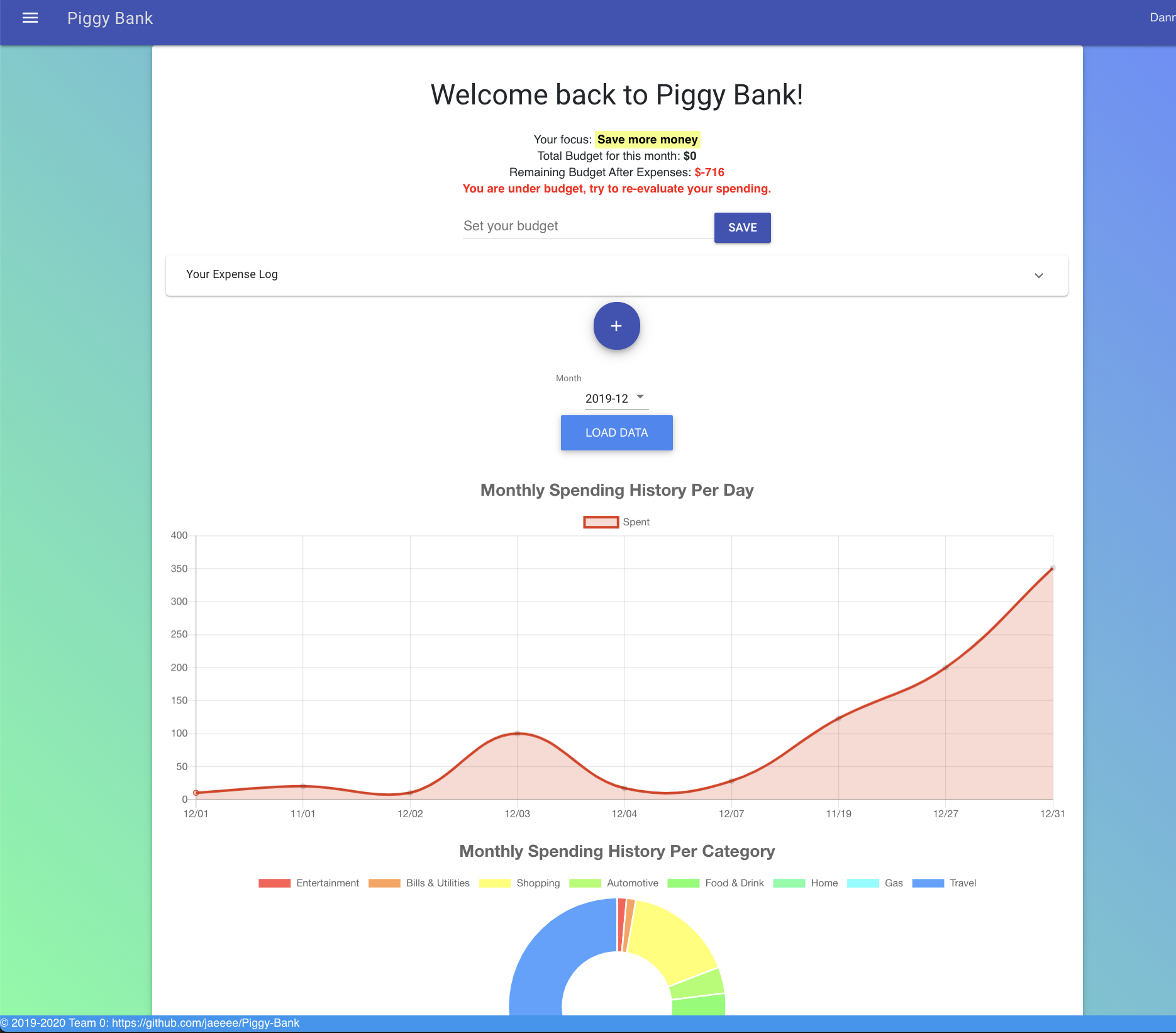


**Profile Page:** A page dedicated to the user profile settings, allowing the user to change their bio, display name, and focus (goals). Also serves as a pipeline for logging out of the website.

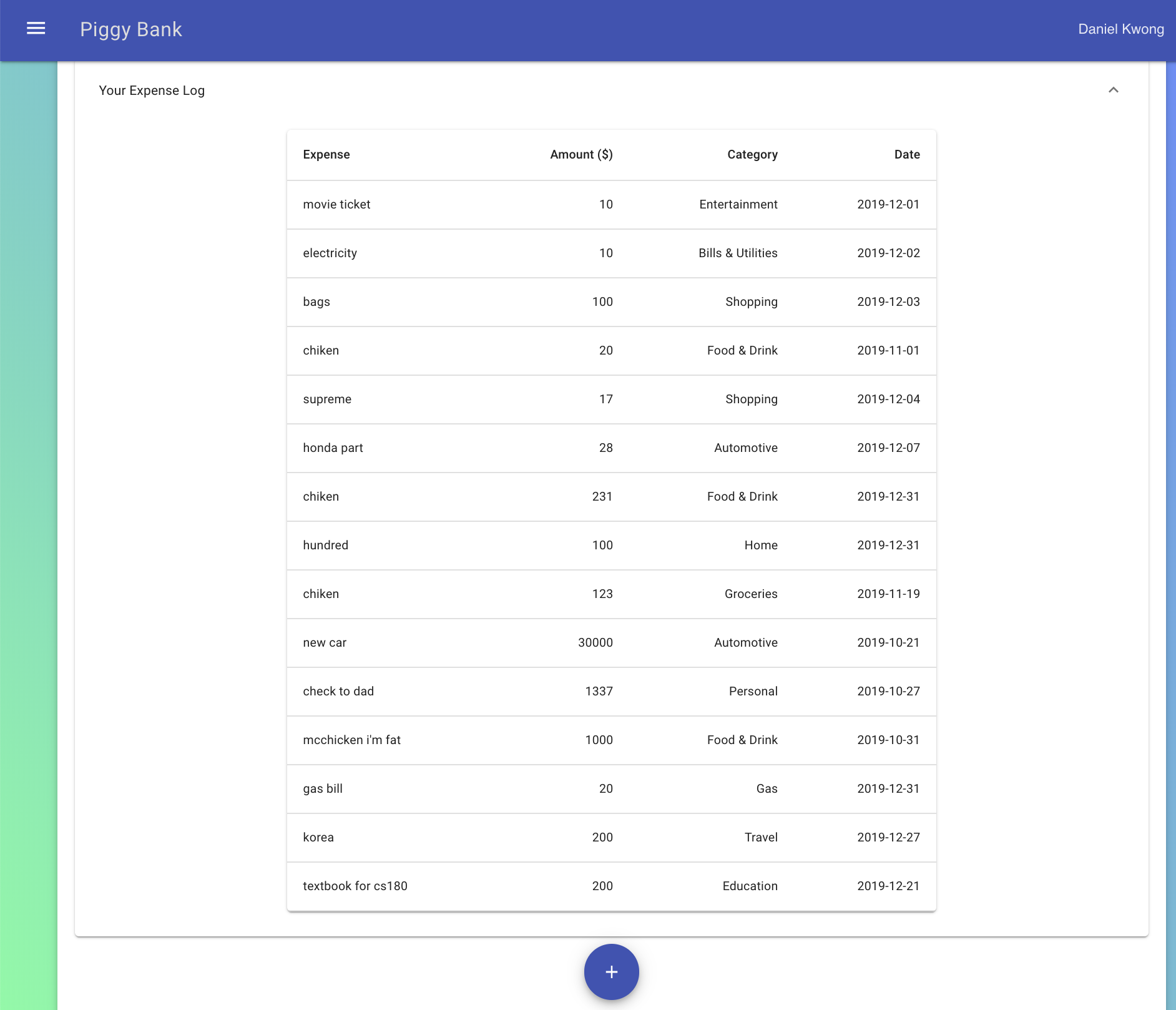


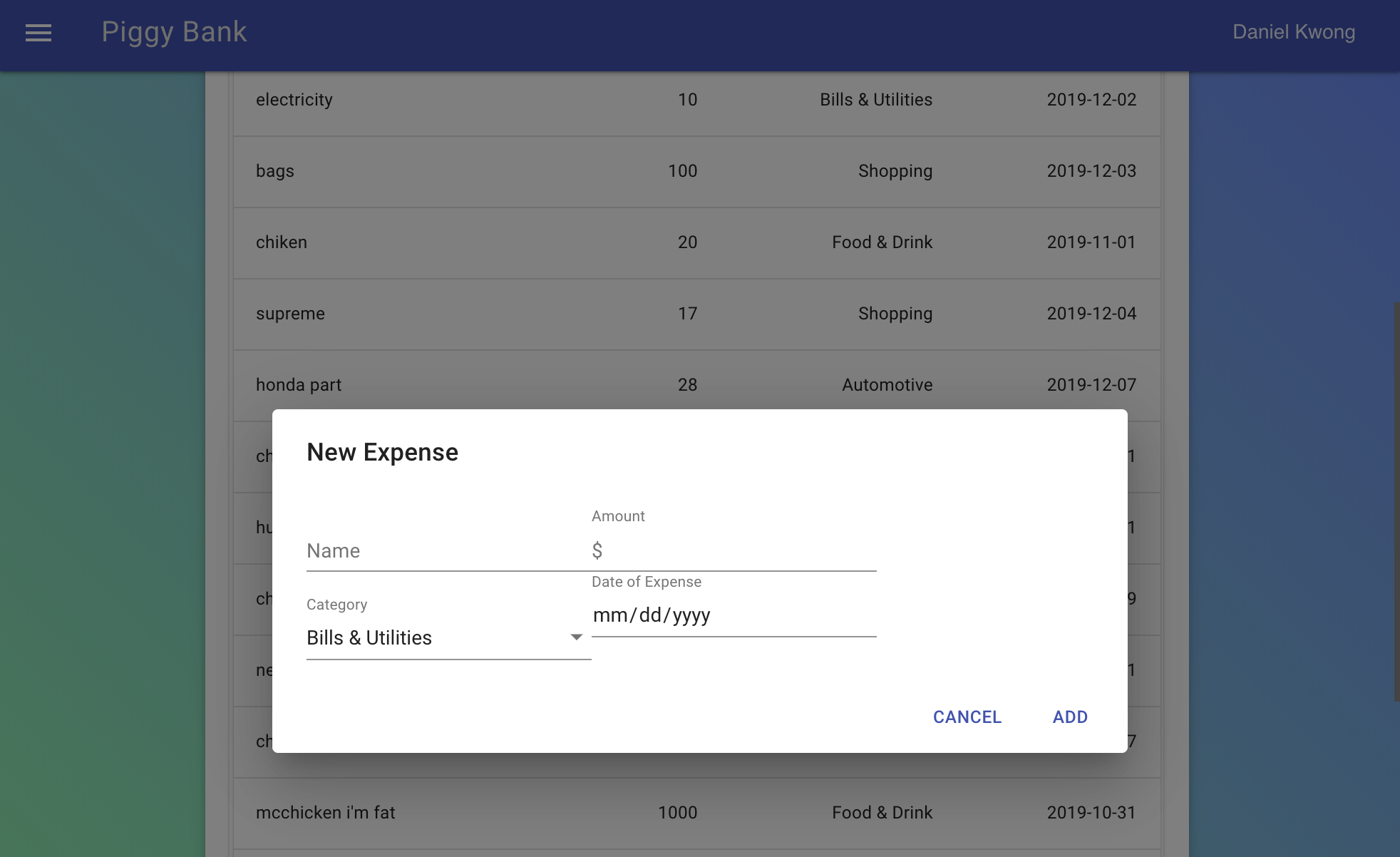
**Home Page:** Displays user goal above the current set budget. Also has a table of expenses the user has input and a button to add more expenses.

**Charts of Monthly Spending (now part of home page):** Displays a pie chart of the different categories of spending for the specified month. Also includes a line graph of money spent over the specified month.

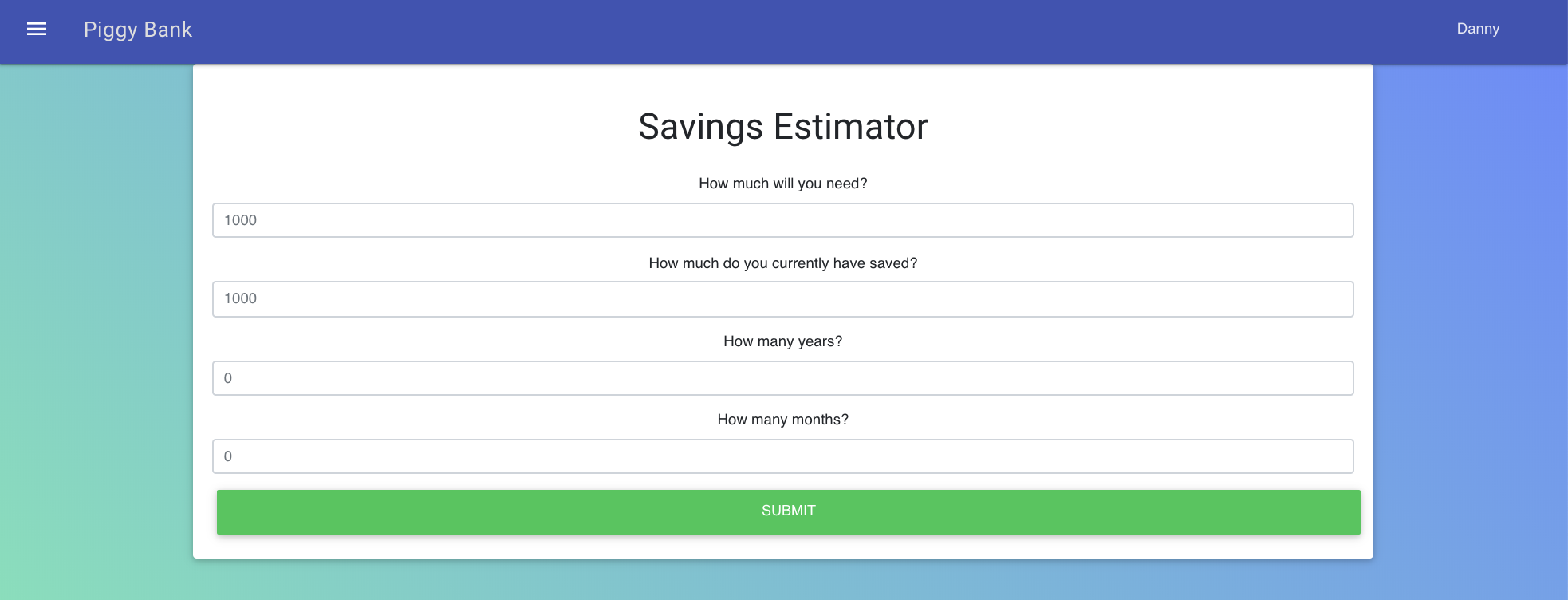


**Expenses Log and Form:** Allows users to see a complete log of all expenses and insert a new expense into the database.

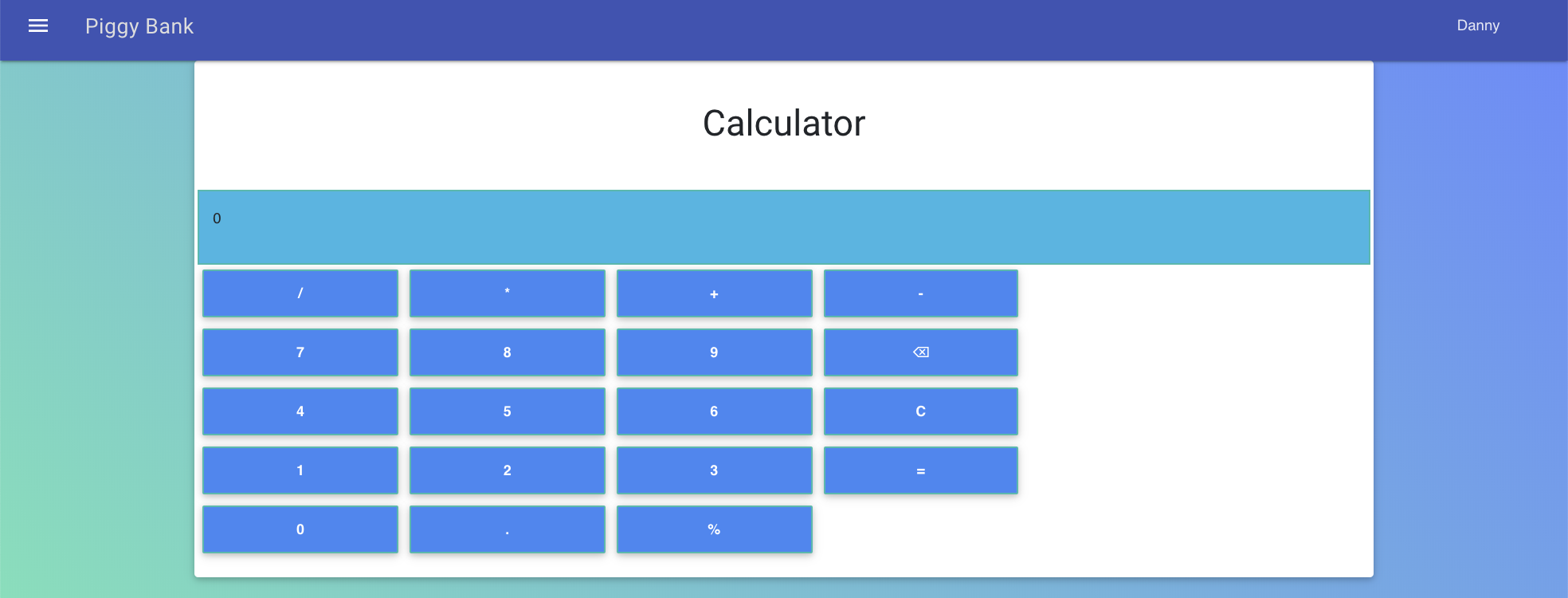




**Savings/Debt calculator:** Calculates how much you can spend or need to save to reach a certain goal. Used [this](https://www.ally.com/education/financial-life/savings-goal-calculator/) for calculation reference



**In-App Calculator:** functions like a normal calculator, taken from [here](https://github.com/kml1990/react-calculator)



**Exchange Rates:** Shows a list of exchange rates for different currencies from a Google API

