Project Proposal
Mathieu Steele
WDD 330, Spring 2021

Purpose

The shared budget application is a progressive web app that is a home-screen installable mobile application developed in JavaScript and allows one or more people to collaborate on a shared budget to track monthly expenses in user-specified categories.

Audience

The intended audience is for couples looking to manage household expenses such as gasoline, groceries, home improvement, and miscellaneous. The project idea came from my own interactions with my wife to communicate about household expenses and track month-to-date spending in various categories. Couples like us would benefit from a simple distraction-free interface to track and manage this type of data and collaborate together.
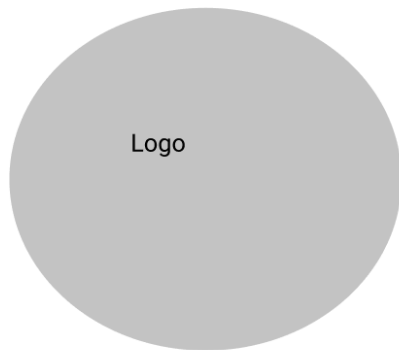
Data Sources

The primary data source will be a Postgres database stored on Heroku and made available for interaction using an API accessible by JavaScript. I will develop the backend with Python and deploy to Heroku.

This External API will use cookie-based authentication and communicate with the API using fetch to consume the JSON.

Initial Module List

authentication.js
budget-categories.js
expenses.js

Wireframes

Logo

Username

Password

Sign In

Create an Account

< **July 2021 Budget** >

Edit

Groceries	$1500
. . ..	$267
• -- -- - - -- - - - -	$154
Remaining	$1079

+ Add an item

Gasoline	$200
. . ..	$35
• -- -- - - -- - - - -	$27
Remaining	$138

+ Add an item

< Cancel

Edit Category

Category Name

Groceries

Monthly Budget

$1500.00

Save

< Cancel

Adding Expense

Category

Groceries ▼

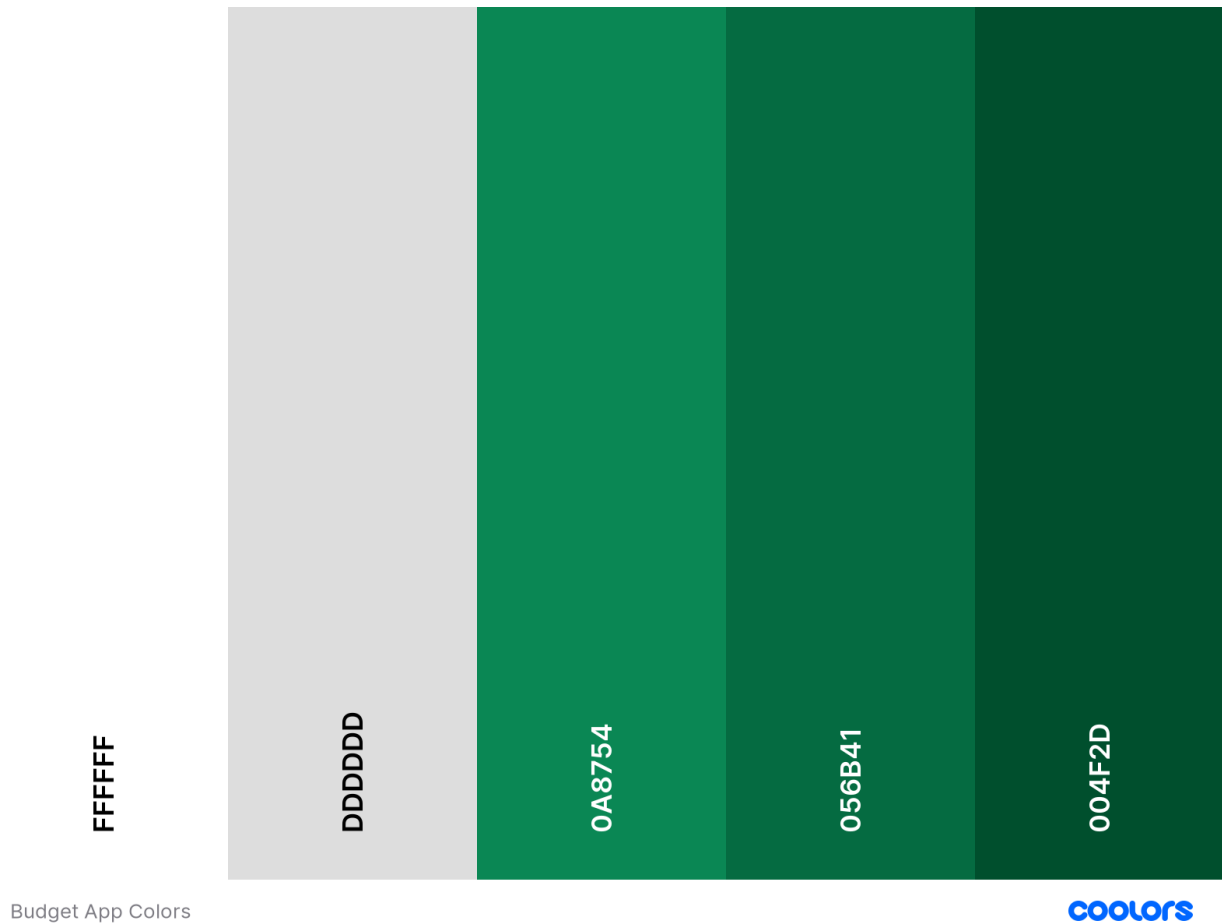Date

(defaults to today)

Amount

$25.00

Note

Produce

Save

Color Palette:

FFFFFF   DDDDDD   0A8754   056B41   004F2D

Budget App Colors   coolors

Typography:

Lato (a google Font) – Font of choice for application.

Development Schedule

Week 10

      Complete Budget Screen Display and Styling pulling static data from JSON File

      Create Login Form

Week 11

      Allow user to log in / Authentication with backend API.

Pull Dummy Data from External API instead of static file

Week 12

Wire up the forms to create, edit, and delete categories and expense entries

Week 13

Testing, Fine tuning of Lighthouse score and PWA requirements

Add additional features and stylistic improvements

Week 14

Catch up on any outstanding requirements.

Turn in Assignment, historically week 14 is only a two- or three-day week.