



Frugal - The Simple Budgeting App

Problem Statement & Market Research

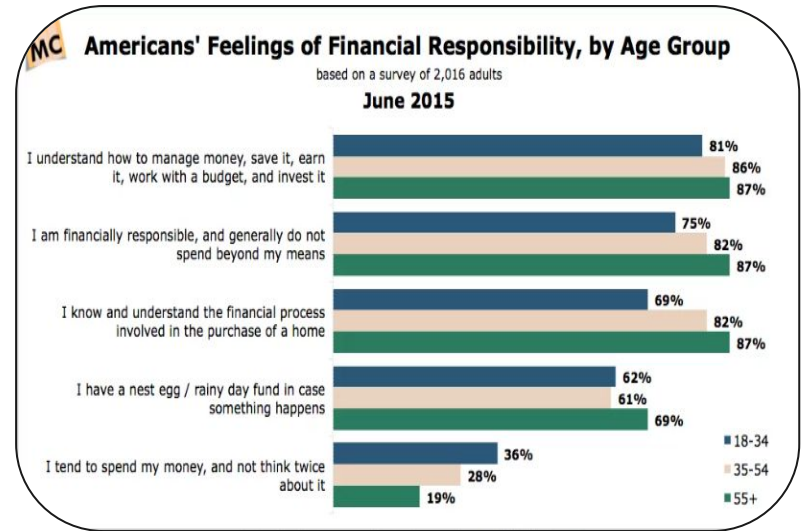
Why a budgeting app?

- Personal finance management apps are a real remedy for inefficient budgeting and impulsive spending
 - They increase financial literacy and money management by teaching users how to understand income and expenses.
- [Credit Karma's survey](#) shows that 56% of millennials feel their financial goals are out of reach
 - 85% of millennials say they've felt too burned out to deal with their personal finances
- Fits the constraint of the local-only CRUD pattern app required by the course (explored later)



Target Audience/User Stories

- Young adults ages 18-35
 - College Students, recently married couples, young adults living on their own
- Take less than a minute to complete
- Emphasize a problem give the solution
 - less than 25% of American households are debt-free



User Persona



Name: Alex Triton

Age: 21 years old

Occupation: College student

Location: San Diego, CA

Bio

- Alex is an aspiring software engineer in college. He lives alone in his apartment with his dog, Daisy. In his free time, he likes to hike.

Goals / Interests

- Create and monitor budget.
- Pay off student loans.
- Balance student life with his hobbies.

Frustrations

- Has a hard time categorizing and tracking his expenses.
- Can't stop overspending on unneeded goods.
- Always exceeds his target budget each month.

Our competitors (i.e. inspiration)



Acorns

Highlight: Education section with the basics of investing, FAQ, and a glossary



Qapital

Highlight: Emphasis on personal and shared savings goals toward a desired expense



Mint

Highlight: A simple expense tracker with categorization for expenses and detailed budget views



YNAB (You Need a Budget)

Highlight: Ability to send reports/notifications about weekly budget habits and overspending

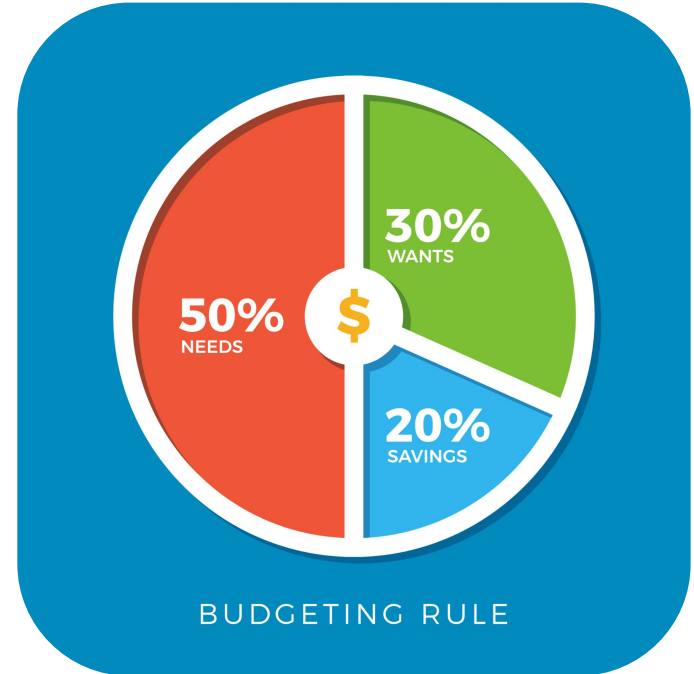
The **CRUD Pattern** & other constraints

Create

The user will be granted a page (+ icon) where they can simply add their expenses in a card format to a running list.

Formatted as: expense, category, description, date (default the day of the created expense)

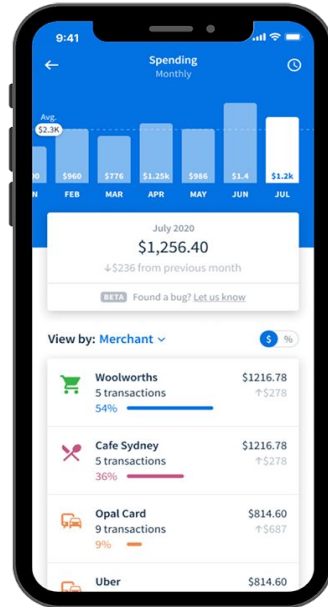
- Categories will be split into three groups.
 - Needs
 - House, Food, Transportation etc.
 - Wants
 - Hobbies, Entertainment, etc.
 - Savings
 - Investments, Retirement etc.
- This will essentially mimic a spreadsheet with all the expense made throughout the month



Read

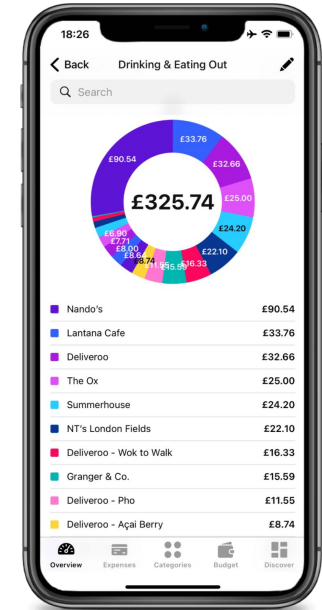
Home Page

- The user can select a dropdown to sort their expenses based on categories or months. (Single query to database)
- Spending by months over the years will be stored on the local database.



Breakdown Page (static)

- Updates when user clicks the tab for the breakdown page
- Shows spending percentages based on category in a graph (query per category to database)
- Shows total spending for that month



Updating an expense

- **Motivation**
 - Users might want to adjust their overall budget plan based on current events, or adjust specific details that were mistakenly entered. They will be able to do this on the homepage of the app by clicking the expense card.
- **Adjustable:**
 - Price
 - Date
 - Category
 - Description

Housing ^			Planned	Spent v
🏠 Mortgage/Rent			\$1,300.00	\$0.00
💧 Water			\$30.00	\$0.00
💡 Electricity			\$75.00	\$0.00
Add Item				

Transportation ^			Planned	Spent v
🚗 Gas			\$175.00	\$0.00
🛢️ Oil Change			\$40.00	\$0.00
Add Item				

Food ^			Planned	Spent v
🛒 Groceries			\$250.00	\$0.00
🍽️ Restaurants			\$75.00	\$0.00
Add Item				

Competitor example



Delete

- Deleting an expense can be accomplished in the home page of the app. One button symbolized with a trash bin will be next to each expense.
- At the click of the button it will give the user an “Are you sure you want to delete this expense?” to which the user will have two options... ‘yes’ or ‘no’.
- Deleting the expense will remove the expense from the list.



Motivation for Features/Priorities

- Creating expense cards into the three categories
 - It's more simple to have cards rather than the user having to over categorize everything (avoiding micromanaging)
- Reading the expenses into breakdown of the 3 categories
 - A simple breakdown like this would be easier to digest rather than throwing too many numbers and categories at the once(avoiding complicated charts)
- Updating the information about the expense
 - It's important to have this because mistakes happen all the time, or in case the user would wants update or add any additional notes about a purchase
- Deleting an expense is also a priority
 - Things change all the time, refunds, etc.

To Summarize the CRUD App

- User will start with a front static page describing the app. Will then be able to login/sign up.
- Then head to the create page after logging in where the user will instantly be able to create expenses with the requirements given.
- If the user wants to get a summary of the month so far they will be taken to a static page breaking down the three categories with the total expense of each.
- Our mission is simplicity and time efficiency. Most budget apps fail to keep customers because of their overly complicated menus and/or have visual distractions. One minute a week is all you need to input your expenses.



System Diagram

Find it here:

https://drive.google.com/file/d/1As7GAdV-frapL3KUh0UwPrEhKD_U7dyX/view?usp=sharing

Roadmap

Find it here:

https://docs.google.com/document/d/1uLT-Xrr5edWe7TOnyhSivqsz4LDtGLloL_biNyScb5Q/edit?usp=sharing

Frugal



Expenses



Breakdown



Logout

Expenses

Username



Categories



	Name	Category	Amount	Date	Action
	Hollister	Wants	\$76.97	10/31/22	<button>Edit</button>
	Costco	Needs	\$165.24	10/25/22	<button>Edit</button>
	Rent	Needs	\$600.00	10/15/22	<button>Edit</button>
	\$AMZN	Savings	\$450.00	10/04/22	<button>Edit</button>
	Movies	Wants	\$24.45	09/25/22	<button>Edit</button>

Frugal



Expenses



Breakdown



Logout

Breakdown

✓ Month

Total Spent: \$1500

Username



Needs: **50%**

Wants: **35%**

Savings: **50%**

Breakdown Wireframe



Tech Stack



Front-end

Electron.js
(HTML5, CSS, Javascript)

runtime framework that allows the user to create desktop-suite applications without internet connection



Back-end



Node.js
(JavaScript)

backend runtime environment that allows for creation of API to interface with frontend



Database



Local First Storage

storage of data in an accessible, offline, and privacy-friendly manner with **electron-store**

Front-End

Sign in



LOGIN

Username

Password

Login

- Username
- Password
- Security validation

Home Page



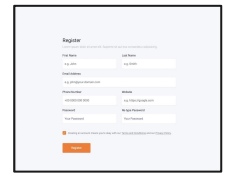
Create your Food Budget. Check tips in table header below.
You can create report in Report worksheet.

Date	Category	What	How many	Price per item	Price (sum)	Where
	Sweets	Candies	0.25	\$5.00	\$1.50	Grocery
	Drinks	Coke	1	\$1.00	\$1.00	Fast food
	Ready Meals	Milk	1	\$8.00	\$8.00	Fast food
	Meat	Beef	12	\$5.00	\$60.00	Grocery
	Alcohol	Whisky	1	\$25.00	\$25.00	Grocery
	Drinks	Soda	6	\$1.00	\$6.00	Home Delivery
	Ready Meals	Milk	1	\$8.00	\$8.00	Home Delivery
	Fruits & Vegetables	Apples	2	\$5.00	\$10.00	Home Delivery
	Drinks	Water	1	\$1.00	\$1.00	Grocery
	Drinks	Coke	1	\$1.00	\$1.00	Fast food
	Sweets	Chocolate	2	\$3.00	\$6.00	Grocery
	Sweets	Candies	5	\$1.00	\$5.00	Grocery
	Drinks	Water	2	\$0.75	\$1.50	Coffee shops
	Ready Meals	Rice	1	\$3.50	\$3.50	Restaurant
	Meat	Beef	0.6	\$2.00	\$1.20	Grocery
	Alcohol	Beer	1	\$2.00	\$2.00	Restaurant
	Drinks	Soda	1	\$1.00	\$1.00	Grocery
	Ready Meals	Milk	1	\$10.00	\$10.00	Fast food
	Fruits & Vegetables	Bananas	2	\$5.00	\$10.00	Home Delivery
	Drinks	Coke	1	\$0.75	\$0.75	Grocery
	Alcohol	Beer	1	\$1.00	\$1.00	Coffee shops
	Bread	Buns	1	\$0.00	\$1.00	Grocery

- All expenses stored as cards
- Expense Info
 - date, item name, item cost, needs/wants/savings
- View/update card by clicking card
- Delete expense icon (next to item)
- Generate Breakdown Report button
- Add Expense button

View/Update existing Expense

- Shows filled form
- Includes
 - Save changes button
 - Delete expense button



Register

First Name

Last Name

Address

City

State

Zip Code

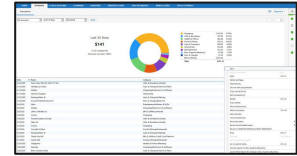
Phone

Email

Register

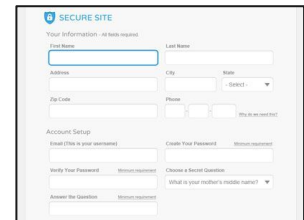
Generate Report

- Pie Chart
 - Needs/wants/savings
- NWS listings
 - Ordered chronologically?
 - Ordered greatest to least?



Add Expense

- Blank form
- Requires
 - Date
 - Item name
 - Item cost
 - Needs/wants/savings
- Add expense Button



SECURE SITE

Your information - no data report

First Name

Last Name

Address

City

State

Zip Code

Phone

Email

Account Setup

Email (this is your username)

Create Your Password

Verify Your Password

Answer the Question

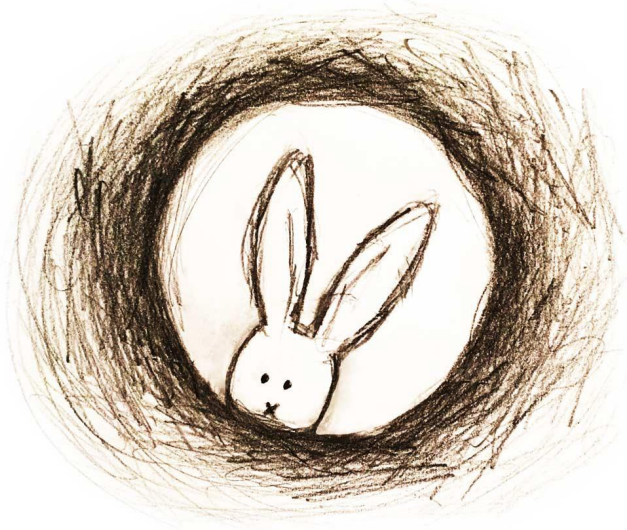
What is your mother's maiden name?



Local First Storage

- Want to have persisting data for things like user preference or app data
 - Do not want to use cloud storage, makes app inaccessible offline
 - Data ownership - user's data does not belong to third-party cloud providers
- electron-store
 - <https://www.npmjs.com/package/electron-store>
 - Saves all data in local JSON file

Risks and Rabbit holes



- Expense Report (change to static)
 - updates when user adds and deletes expenses
 - User's control increases unpredictability
- Data Storage
 - Individual Account (login, reset)
 - Security
- Technical difficulties
 - Quickly learn new languages
 - Github organization
 - Code review and merge
- Time Constraint