

Matthew Greenberg	Request For Proposal	August 30th, 2018
Moolahlah - Financial Planner Application		

Elevator Pitch

Who likes to save money? I believe everyone does! One of the easiest steps you can take to saving money is to create a budget. So why doesn't everyone have a budget? Because it takes time and most don't know where to start.

Moolahlah is an personal budget manager that aims to help it's contistuants save money by leverging sound financial planning.

Project Goals

The goal of this project is to create / finish a financial planning application that helps users be piscally responsible.

I will consider this project a success if... 1. The project is completed by the time of my graduation. 2. The code is immaculate, well commented, and documented. 3. The university is proud of my work.

The metric I am chasing is monthly recurring users. New user sign ups will be a great metric to ascertain interest, but recurring users will show the products maturity.

The application will be free for signup and general use. Revenue will be made from a premium version & targeted marketing via click through adds.

The target market of this application is the American population, ages 16 to 30, who want to take control of their finances. We would like to target a younger audience.

Scope Of Work & Information Architecture

Domain Purchased : moolahlah.com

A refresh of the initial design work is expected.

1) Platform Bootstrap - Setting up the initial backend server: Express, and MongoDB (Using Mongoose ORM) -

Integrating React & Redux via webpack into Express - Mailgun integration for emails and reminders 2) Front End Design - Sign Up, Login, Logout, Password Help - Budget Planning Dashboard - See below - Landing Page - Account Management Page - CMS / Blog - Long term not part of the MVP 3) Backend Development - REST API - Authorization System - Email / Password / Salt / Hash - Serve static content - Email Notification System

*Budget Planning Dashboard Quick month progress overview Analytic Graph : Shows monthly budget progress and pain points Budget Setup Walkthrough / Guide & Organization Easy transaction adding

Dashboard Storyboard and Wireframing are necessary artifacts that will be created.

Technical Requirements

This is the MERN stack -> Mongo, Express, React & Node.js For security purposes the app will sit behind NGINX web server via a reverse web proxy. Eslint will help maintain coding standards. JEST & Enzyme are great testing frameworks I would like to utilize, but the implication of TDD would increase time to delivery. I am not sure that I can afford it. I will use Redux for state management. Following the middleware design pattern outlined in Nir Kaufman's book 'Thinking In Redux'. Webpack is to be utilized as a task runner to compile javascript via babel. The webpack bundles will be served through Express.

E-commerce

For MVP, I don't need to process credit cards. When I introduce the Primum model, I will need to process credit cards. Merchant API and processing will dictate how payment processing is implemented.

Content Management

Outside of normal web app assets, I will implement a CMS / Blog that has information regarding creating financial wealth, tips, and other finance related material. All CMS will be custom.

Programming Language

Web Languages => Javascript / HTML / CSS Machine JS => Node.js

Webhosting

I need a host for the mongo database and node / nginx server. I am not dead set on any host at the moment. A

few that come to mind are AWS, Heroku, and Digital Ocean. I will determine the best host based on ease of deployment and CI / CD integration.

Browser Support

I would like to support IE11 and all Major Browsers (Chrome, Firefox, Safari, Edge); however, MVP only has Chrome and Firefox being supported.

Integrations

Mailgun integration - Will allow me to send notifications to the users by their email. Primum version (not mvp) has automatic banking transaction integration and credit monitoring / integration that will require multiple APIS.

User Accounts

All users will be required to signup for an account in order to use the service. In order to track a budget, the user needs be able to return at a later time. Single sign on options with google will be available, just not part of MVP.

Accessibility

Normal a11y compliance will be fallowed. Google lighthouse metrics will grade the performance. Perfect compliance is not part of MVP.

Responsive Desgin

Mobile first is going to be the goal. Wireframes will be created for the budget dashboard that include Mobile, Tablet and Desktop.

More Information

I believe I have covered all aspects of the design proposal. This is a lot of work and I am excited to get started. I actually began the idea for this product back in Portfolio 5 but never had the time to come anywhere close to finishing it due to the fact that the focus of the class was learning API integration. I spent a lot of my time in that class learning Firebase and Redux. I am more prepared now to complete this activity.

You can view the initial effort here: <https://mattdevio.github.io/Portfolio5/#/>

The initial effort only had firebase authentication support some super basic data entry fields on the 'dashboard'.