**Caption**

InvestMe is a cross-platform mobile investing application designed to teach users how to invest better. The application is written in Dart, using the Flutter framework. InvestMe pulls up-to-date stock data from the Alpha Vantage API and compares the data to user inputs stored in Google’s Firebase. Below are just a few of the pages in the app. Almost all of the

**Home Screen**

The home screen acts as the first page you will see when launching InvestMe. For returning users, entering your username and password will unlock your account. New users should click the link at the bottom to start setting up a new account. The app allows users to create an InvestMe account using their Google and Facebook credentials. This is completed on the page utilized for creating a new account.

I created the design page to set the style for the entire app. For instance, the size and width of the input boxes, the font style, and the color scheme became a key part of the app. Users expect consistency and will see the same patterns throughout. As this was my first time writing in Dart and using Flutter, the learning curve took some time. Fortunately, Dart is a C-derivative and styling Flutter widgets is similar to using CSS. After completing the layout of the page, the team hooked the application to Google’s Firebase, effectively allowing users to login and store data.

**New Entry**

One of the key features is the ability for users to be able to log their expenditures. This page presents an easy to read, self-explanatory interface. Below the “Enter Amount” text, a calendar icon is available (defaulted to the present date), so that users can date their input. After, the user can input the amount and name of the expenditure, and any additional notes they find helpful. Clicking save then logs the entry into the Firebase database.

Users who are utilizing the app properly, will be returning to this page often. Each purchase, specifically purchases that may be viewed as unnecessary (coffee, snacks, etc), should be logged and dated. By doing so, InvestMe will have accurate user data to compare the stock market data against. This will allow the app to provide accurate suggestions.

**Tutorial Page**

Investing is complicated, and most people do not know where to start. This app was designed for users such as this. The tutorial page consists of a slider of three separate screens that explain exactly what InvestMe does. It describes how the user can log their data, view recent expenditures, and compare data with stock information. InvestMe was designed with simplicity in mind and I did not believe the tutorial needed to appear more than once. For this reason, the tutorial only appears the first time a user logs into the account.

**My Stocks Page**

This page acts as the user’s control center for checking stocks and preparing stocks to be compared with their expenditures. At the top of the page there is a search box that searches the Alpha Vantage API for the user’s input. If the input is found that stock will appear at the bottom section of the screen. From here, the user can mark the stock as a “favorite.” Now this stock will be tracked and used when comparing inputs to the Alpha Vantage API.

Although the point of InvestMe was not to be just another way to check stock data, that ability was something we recognized when developing the app. The My Stocks page could allow users to keep track of stocks they are interested in, in addition to being able to prepare them for comparison. When a user gets tired of a watching a particular stock, they can simply remove them from the list.

**Compare Stocks Page**

After designating stocks as favorites, the user can then properly utilize the compare stocks page. Here, the user’s favorite stocks will appear along with a “Compare” button. At this time, the lower half of the screen will be empty. The user can select which stocks they want compared from “My Favorites” and then press “Compare.” This will initiate a popup that will ask the user to select a certain timeframe. For example, if the user presses “Past week,” the selected stocks will be compared to the past week worth of inputted expenditures. Then, the lower half of the screen will be filled with a graph comparing the stocks to the user data.

I believed that using a visual representation, such as a graph, would drive the point of wasteful spending home. As consumers, we have become desensitized to the “numbers” in our bank accounts. We very rarely see physical money anymore and have lost the visual warning of what we are actually spending. Hopefully, seeing the numbers visualized will educate and cause users to slow down when making purchases.

**Menu Slider**

The menu slider is available from most pages on InvestMe. Clicking the menu icon causes the screen to slide out from the left slide. On the menu slider there are several options: Search, New Entry, Previous Expenditures, Compare Stock Data, My Stocks, Need Help, and Settings. All of these are self-explanatory and act as a quick way to navigate the application. In addition to text, each option has an icon to give users a hint on what the pressed page will do. For instance, the Compare Stock Data option has a graph icon, a reference to the fact stocks are compared using graphs. When the user is done, simply clicking outside of the menu will cause it to close.

**Welcome Page**

The Welcome Page is the first thing new users will see when entering the app. The page contains two fields, an email and password field, and two clickable links, a button and hyperlinked text. For returning users, they can enter their authentication credentials that were created when they first made their account, this will take them to the Character List page once submit it clicked. For new users, clicking the “Sign Up!” text will bring them to a page to create a new account.

This app was designed with simplicity in mind. It was intended to cut through the complexity of other character creation apps. I incorporated this idea into the style of the application as well. Each page from the Welcome Page, to the Character Creation page is meant to complete a single task.

**Character List Page**

The Character List page contains a list of all characters registered to this account. Each time a new character is created, the character will be appended to the list. Each name is a hypertext link that will take the user to the that specific character’s page. At the bottom of the page is a button that will bring the user to the Create New Character page.

The page was designed to be the “control room” of the app. From here, users are able to select all of their registered characters and add new ones. For players using multiple characters in a game, simply pressing back after selecting a character, will return you to this page so you can select a new character quickly. The two pages were designed for this interconnective behavior.

**Create Character Page**

**Edit Character Page**

From the Character List Page, to the Character Page, there is a button that will bring the user to the Edit character Page. This page allows the user to make any and all alterations to the currently selected character. The name field and all other fields will be editable. For instance, if a character has the stat of “Health” with a current value of “40,” you could easily change “Health” to “Life” and “40” to “30.” Furthermore, users can add additional fields to their characters with the click of a button. Ideally, users would have this page up during a game session to make quick alterations to any stats that are changing in-game. At the very bottom of the page there is a toggle that will stop the Confirmation Page from appearing when a user saves the changes. It could become tedious to constantly be saving changes and going through the Confirmation Page during a game.

**Card Text**

Built on the Flutter framework, this application allows users to log expenditures and compare them to up to date stock data. If a user spent 70 dollars on coffee this week, the app will show a better purchase would have been a 70 dollars purchase of their favor coffee company’s stock!

Built on the Angular framework, this application is a multipurpose character creation web app for any type of role-playing game. Users can quickly create a character and add as many or as little fields as necessary. In addition, the application offers templates for more common classes.