

Kenia Duque
Steve Yoo
Monish Ratanji
Andrew Sie

Documentation and Guide For Future Development

Index.html

With limited time, we decided to not focus on the aesthetics of the landing page so we kept the template design code as is. Two ideas we had were to embed a recorded video onto the iPhone and change the color of the iPhone. Currently, there is only one image being shown--the homepage. With a video however, we could inform and show the user the different features of the app quickly. Changing the iPhone color from Space Grey to Silver was also discussed. The Space Grey iPhone just blends in with the black background.

Sign Up/Log In

We made the Sign Up and Login pages with a similar design yet a clean execution. Firebase is our back-end service so a user's email/password were just a key/pair value for Firebase. Right now when a new user signs up, it takes them to the Log In page where they have to input their information again. In the future, we'll implement a successful user registration to go to the homepage immediately after the Sign Up page. We implemented error detection where the app will spit out validation on successful actions, as well as state error messages if the user inputs a bad value (ie. email already in use, invalid password, etc). In the future, we will also connect Facebook and Google+ to give the user more options and freedom.

Home.html

The market message is now in real time--it'll say that the market is open during the hours of 9AM-5PM PST or that it is closed before and after these hours. We also got the graph to plot normalized data for the last 30 days, separating the graphs for gold/platinum from the graph for silver because of scaling issues. In the future we plan to add graphing for the bullion that is in the collection. The biggest problem we had with this page was figuring out how to parse the data from an ajax server and work that into the graph. The chart.js library that we used doesn't make it easy to change the data on a graph, and doesn't allow data to be retrieved from it. Combined with missing days for when the market is closed for the market data fetched from quandl, it took some creativity to figure out how to normalize the data and collect it all at once to be added to the graph. The result is about 10 functions that daisy chain call each other passing more and more data along until all the data is collected.

Settings.html

We made settings.html to be the portal to user management. When a user clicks on the icon-cog in the right hand corner of the app, they will be led to a page that welcomes them into Settings with a small message that includes their username--this user experience is very personalized. The user has three options: change password, change email, and log out. We added small icons to the password and email buttons to visually differentiate with the logout button. Yet again we used Firebase and it's methods to do authentication and report success and failure messages to the user.

My Gold/Silver/Platinum

For these three pages, the table at the bottom updates correctly whenever a user adds or deletes a coin. While the search bar is still present in the UI, we did not implement its functionality because we were considering removing this feature. Yes--it would be helpful to have a search bar if a user has a lot of coins, but there is no clear indication of what to search for since these coins don't have unique names. For example, what would a user input if they were searching for a dollar with quantity 12? dollar or 12? In addition, the history for gold/silver/platinum values are plotted on the graph.

Item Detail

When a user clicks on a selected coin from the previous coin-homepage, that specific item detail appears. This is done through javascript in a way that mimics a GET request to a server. For future development, we will use an actual GET request instead of a 'fake' one. The request is sent with the type of metal and the key of the item from firebase. The key is then searched for on firebase in order to provide the item details. A delete button is available on the item detail page that allows the user to delete the selected item. For future development, an 'edit item' button may be implemented.

Add Item

For add.html we kept most of the fields as input text boxes; we believe it's cleaner than if we used a calendar widget. We also added three of the most popular coin types, but of course, in the future we'll find a cleaner way of importing that information. The UI for coin photo upload is still present but because of time we didn't focus our attention to it. There were two options we were thinking for the upload though: either the regular way of accessing a file/camera, or populating a stock image since most of the coins look the same.