

Team members:

Jon Gordon
Sage Long
Jack Blackburn
Mirna Masri
Hampton Walker
Nick Spooner

What features were completed? What worked?

- **Login page**
 - **What worked? What was completed?**
 - HTML form to input username and password fills correctly, cancel button takes you back to home page, Remember me button works, and you cannot submit until all fields have been completed. Styling is also there and consistent with the rest of the site.
 - **Future implementations?**
 - need to implement a link to register page if they don't have an account, also need to implement forgot password page. Submit button needs to take the user to the right page and check to make sure the user exists. Also need to push up the user image to github. Connecting the database of users to this page to check if a user exists will be the next step, once I complete the above features.
- **Register Page**
 - Format and Styling complete, still need to implement password verification, and also a disabled submit button when not all required fields are filled out.
- **Login Home:**
 - This will be the first page that a user sees when navigating to mockstock. It is just simply a Home page with two options, Login or Register, where the user can navigate to either one easily. Only thing missing is a little bit more styling perhaps, as the page is mostly empty.
- **News page:**
 - **What worked? What was completed?**
 - onLoad, news page will load 9 articles that are the most recent. User clicks on the image of the associated article to be brought to the article. Users can change the news page at the bottom of the page to display 9 different, older articles. Only 9 articles are displayed per page in a 3x3 matrix of cards.

- **Future implementations?**
 - Filter by category to show articles related to a specific category, integrate with API to pull real time articles from database instead of locally stored information.
- **Homepage:**
 - **What worked? What was completed?**
 - The homepage created so far Contains our vision statement, also. It contains a slideshow animation that will automatically change, and on each slide, there's a little bit of information about our website.
 - **Future implementations?**
 - On the side of the homework page, there will be graphs showing the user's stock gains and losses. also, the top three stocks and bottom three stocks on the market
 - The team's page will contain a card containing some interesting information about our team and our TA. Also, group accounts on gitHub.
- **Trade Page**
 - **What worked?**
 - Trade page created where users can input values for ticker company symbol and quantity. These values will be retrieved with API (below) to find actual value of stock based on current pricing. Drop down selection for buy/sell which will determine which input fields will get displayed/hidden. Trade page created where users can input values for ticker company symbol and quantity.
 - **Future implementations?**
 - These values will be used with API (below) to find actual value of stock based on current pricing. Drop down selection for buy/sell which will determine which input fields will get displayed/hidden. User validation with their wallet amount is not available yet as database is still in development. Hiding displaying input fields based on drop-down selection for buy/sell on trade page is still in development.
- **API Stuff**
 - **What worked?**
 - Javascript file to pull individual quotes when API called
 - Export to JSON with correct formatting
 - Javascript file to take symbol as command-line argument and pass into API function call
 - **Future implementations?**

- Basis for javascript file to pull symbol information for S&P 500 from API complete, isn't functioning but it might be changed to fit recommendations given by TA
- Anything that needs to be sent to the database must be done with API stuff, so this is where we will be spending most of our time in the near future.
- **Profile page:**
 - **What worked?**
 - Page where users can view stock positions. Table is currently static and does not show actual users' positions. This will need to be updated with information from the database.

What issues did we face during development?

Table on profile.html is static and we need to find a way to update this table information with our API using node.js. Javascript will be utilized to update displayed tables. File to pull S&P 500 data doesn't function since API call doesn't work with for loops, needs to change according to TA recommendation. For the Login page, the submit button is not working the way I want it to, the link it's supposed to go to after clicking it isn't working right so I have to fix that.

Advice from TA:

Make sure we know our limit on the number of requests per second available for our APIs. If we are running the API in a for loop, it's probably sending hundreds of requests per second, so this can cause many unforeseen errors. Also, treat our APIs as ways to pull specific information instead of using them to constant check lots of unneeded information specific to the user. I.e. treat our APIs as a demand basis. Finally, we should hone in on a specific feature that we want to be implemented. We have lots of features, and integrating more features at once will cause more problems. As we continue development, we will begin to realize how complex some parts of our project are going to be, and thus we should only focus on what is essential to our project before its deadline.

Individual Contributions:

- Jon Gordon:
 - profile.html, trade.html
- Jack Blackburn:
 - Created login.html
- Sage Long:
 - Created register.html, loginHome.html
- Hampton Walker:

- Created news.html, animated buttons, merged everybody's individual css to styles.css and changes styling across pages as needed, began database design
- Mirna Masri:
 - Created homepage.html, team.html, animation between pictures/text on homepage
- Nick Spooner:
 - Created quotes.js, quotesInput.js, working on quotes500.js and any other API-related Javascript files (big node guy)

TOTAL DEMO TIME (including interview): 12 minutes