“Trade Kings”

Technical Documentation

https://github.com/bsonani/TradeKings

Software Engineering (14:332:452)

GROUP 13:
Christopher Cena
Raymond Del Rosario
Karlo Fernandez
Diego Ordonez
Krutant Patel
Nakul Patel
Chris Salandra
Akarsh Sardana
Bhargav Sonani

April 1, 2018

Trade Kings Forum:

The start off this portion was coded using a combination of PHP, SQL,HTML and CSS. Originally, consisting of 11 scripts ,however, two scripts a registration and login were simply used for debugging and testing and eventually emitted. Still, the form takes part of the remaining 9 scripts. An account, connection, menu bar (header), index page(home page),member page, profile, topic, post and replay script. The account page was coded to display user information as well as future stock information from the user only accessible by the user pulled using php SQL commands from an Amazon database. The connection page is a small script in order to connect to the databases. The menu bar script was coded in order to display a menu bar accessible in almost every other script. The home page was encoded in order to store all the topics in a SQL database in order to allow all users a necessary starting point. The member page pulls users emails from each database and they are linked to the users profile page. The profile page was encoded similar to the account page except it is accessible to all and it is linked to the members page. The topic page was used to display user topics as well as replies and uses query commands to pull user posts from a database.The replay script was used to store replies from a database and the post script was used to store topics to a database. A forum database table was formulated consisting of three tables a user table for user information, a topic table to store topics and a replay table to store replies. Replies were linked to topics by storing the topics unique primary identifier key and only be displaying when their key matched the current topic.

Signing Up/ Email Verification:

Our webpage were all designed in bootstrap and html, where we wanted to give the user a simple and easy experience. We allow the user four different inputs to inform information regarding themselves to us. From there, the information is pass along to our email function. The Function was programmed in php and allows us to send a verification email to the user who signed up on our website. When the email is sent, a random six-digit code will be assigned to that user's account and included in their verification email. The email was created with google, and allows the developers to send emails through their localhost. While the random number was also programmed where the developers will not see the number assigned. This is to allow users to feel more comfortable and secure about their information being on our website's database. The final set is with our verification page setup. We give the user a box to input their code they were given in the email that specifically sent to them. Our program will check if the number is given matches the one the user was assigned. If so, the user is granted access. If the input number and assigned number don't match up, the user will be given a message and asked to repeat the process.

Researching Stocks:

In order to research and analyze a stock, a user must go to the Research Stock page where they will be prompted with a search bar to enter a stock ticker (symbol). The search bar is implemented using an HTML form element with an input element (search box) and button (submit button). The form element is styled using CSS. Once the desired ticker is submitted. The input is taken by Javascript and is used to access the stock API (IEX) and retrieve the necessary data. The data receives is in JSON format. Javascript is used to parse the JSON data and manipulate to come up with all the desired content, basic information, general company information, overview statistics, financials, similar equities, related news, and graphs. All the text displayed in the webpaged is styled in CSS based on a defined pattern. Furthemore, the plotly.js library in Javascript is used to take time series data and create a graph. The user interface of the page is done with a combination of CSS Grid and Bootstrap. The Bootstrap is used to format the inner contents of smaller elements to displayed in an organized and user-friendly manner. CSS Grid is utilized to organize the webpage on a larger scale by taking various information elements and organizing them on the page accordingly to create a main panel with a side panels containing other information.

Researching Cryptocurrencies:

In order to keep the website organized, user-friendly, and with a consistent theme, the Researching Cryptocurrencies page follows the same user interface format as the Researching Stocks page; thus, it involves most of the same technical concepts. First, the search bar implemented in the same exact way. Also there is a main panel with a series of side panels created using CSS Grid. Furthermore, the interior of many of these grid elements further use CSS grid or Bootstrap to organize the contents. Javascript is used to access the APIs (currently Alpha Vantage and CryptoCompare) and obtain the data, which is also in JSON format, to be displayed. In addition, plotly.js is used to create the current graph that is illustrated. Moreover, the text follows the same format defined in the Researching Stocks page styled using CSS.

Creating/Joining League:

When users wish to either create or join a league, they must first go to the league's main page. On the main page, a table illustrates to the user which leagues he/she is currently active in. Similar to all other functionalities, these features are implemented using php, javascript, sql, html, and css. So in order to display this table, a sql query is sent to the database through php. It identifies the user because a php session was created so that the user is uniquely identified regardless of which page he/she goes on. Through php, the database is queried to return a list of leagues that contains the user's id, which in this case is their email address. The returned data is then displayed in a table that is formatted using the Bootstrap library and personally customized with CSS.

Furthermore, the user has the ability to create a league. The form to enter in the required data for creating a new league is built through html. Once it is detected that all fields are completed the user is allowed to submit the request. This is confirmed through html, as it is set, that all input values in this form are set to required. Then html posts all the input values, so they can be processed by php. When php receives the input values, it simply compiles all of them together and sends an insert query to the league database table, adding a new league.

Similarly, when a user wishes to join a league, it operates in the same way. A table displays all the leagues that currently exists in the database. This is done by another sql query through php. Then using php, it parses the publicity of each league, and dynamically displays the joining league. If a specific league's publicity is set to public, then a "join" button is shown, however, if the league is private, then a "request invite" button is shown. Once, the user clicks the "join" button, it posts the name of the league, so that php can recognize this request. Php then sends a query to the database adding the user to that specific league. This is reflected on the table, as it updates to show the number of players in the current league. It does by querying the database and counting the number of unique email addresses under that league. It also reflected on the main page, as it updates the leagues that the user is in.

Educational Content Page:

The Educational Content is, currently, mostly HTML heading and paragraph elements with text containing Stock and Cryptocurrency definitions. The text on the page is styled using CSS. There is a table of contents developed using CSS and HTML to allow the user to jump to the content of any major heading available. Finally, the page is organized using CSS Grid.