

Crypto currency tracker

Not Fast, Just Furious



October 24, 2020

Members: kyle supple, parth patel, hager albadri

Stakeholders: Not Fast, Just Furious, Ike Quigley, CSC-340, UNCG

We have abided by the UNCG Academic Integrity Policy on this assignment

Table of content

1. Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   1. Purpose . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   2. Document Conventions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   3. Intended Audience . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   4. Definitions/ Jargon . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   5. Project Scope. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   6. Technical Challenges. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   7. References. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
2. Overall Descriptions. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   1. Product Features. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   2. User Characteristics . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   3. Operating environment . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   4. Design & Implementation Constraints . . . . . . . . . . . . . . . . . . . . . . . .
   5. Assumptions & Dependencies . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
3. Functional Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   1. Primary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
   2. Secondary . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
4. Technical Requirements
   1. Operating Systems & Compatibility
   2. Interface Requirements
      1. User Interface
      2. Hardware Interface
      3. Software Requirements
      4. Communications Interface
5. Non-Functional Requirements
   1. Performance Requirements
   2. Safety & Recovery Requirements
   3. Security Requirements
   4. Policy Requirements
   5. Software Quality Attributes
      1. Availability
      2. Correctness
      3. Maintainability
      4. Portability
   6. Process Requirements
      1. Development Process
      2. Time Constraints
      3. Cost & Delivery Date

**1 Introduction**

**1.1 Purpose:**

Cryptocurrency is a digital asset designed to work as a medium of exchange wherein individual coin ownership records are stored in a ledger existing in a form of computerized database using strong cryptography to secure transaction records, to control the creation of additional coins, and to verify the transfer of coin ownership. Because of how cryptocurrency has gained attention, we want to make a program that helps users to check the stock of every crypto currency and make decision based on the market. The program

**1.2 Document Conventions:**

API: Application Programming Interface.

SQL: Structured Query Language.

**1.3 Intended Audience:**

We want our program to be available for everyone whether they have experience with cryptocurrency or not. Our main intended audience is Ike Quigley, CSC-340 students and UNCG.

**1.4 Definitions:**

JDK: Java Development Kit.

SQL Database: A relational database that stores information across multiple tables.

**1.5 project Scope:**

This program is intended to help users who are interested in cryptocurrency to make decision regarding buying certain stocks. Each user will have their own account with an email and a password of their own, said account will be stored in our database for next time log in and validation. If the user happens to forget their password or username, they should be able to reset the password and retrieve their username using their email address stored in our database.

Our program will make connections to three different APIs which are AlphaVantage, CoinRank, CoinHistory. Our program depends mainly on those APIs as they are our main source of data. After retrieving the data from the APIs, the user should be able to display all the cryptocurrencies (if “all” is selected) and choose their favorite to save and compare for best outcome. Our program provides four different types of charts for the user to choose, bar chart, pie chart, line chart, and candle chart. The user should be able to add more than one cryptocurrency to the line chart and compare accordingly.

All online users will be visible to one another; thus, users will have the ability to add each other and become friends. In the future, we want to give the user the ability to send a message and to appear offline for friends and other users.

**1.6 Technical Challenges:**

Our main technical challenge is connecting to an API as we’ve never done that before.

**1.7 References:**

<https://blockgeeks.com/guides/what-is-cryptocurrency/>

**2 Overall Description**

**2.1 Product Features:**

* Login page.
* Ability to reset password .
* Ability to retrieve username.
* Ability to create a new user.
* Ability to use graphs to show data.
* Ability to search coins by name.
* Ability to save and remove coins.
* Ability to compare coins.
* Ability to add and delete friends.
* Ability to switch between dark and light mode .
* Properly closing and exiting the program by clicking “x” or “log out”

**2.2 User Characteristics:**

As of now, there is only one type of user who has access to our main features of the program. The user is simply any user who is interested in cryptocurrency.

**2.3 Operating Environment:**

Our program can be used by anyone who has access to a laptop or PC with internet connection.

2.4 Design and Implementation Constraints:

**2.5 Assumptions and Dependencies:**

The program mainly depends on the AlphaVantage, CoinRank, CoinHistory APIs from which we get the data from. Our main mission is to keep checking if the used APIs are still in a good shape and are not affecting our users’ interactions.

**3 Functional Requirements:**

**3.1 Primary:**

The program’s primary functions are: connecting to the 3 different APIs we are using for getting cryptocurrency data from in this program, ability to view all cryptocurrencies in a table and saving the ones that the user is interested in, using charts to display the saved coins or any coin by searching its symbol, adding online users to the friend list, and viewing the prices in different currencies.

**3.2 Secondary:**

The program can not be accessed without having a user, so our main window features are: creating a new user, resetting the password, retrieving the username and of course logging in.

**4 Technical Requirements**:

**4.1 Operating Systems and Compatibility:**

The program should work any operating system that has Java installed and internet connection

**4.2 Interface Requirements**

**4.2.1 User Interface:**

The user will need a keyboard and a mouse to be able to login and use the program as intended.

**4.2.2 Hardware Interface:**

The program doesn’t require hardware that is intended for gaming or heavy use, but rather basic hardware which includes a running PC/ laptop with function parts to be able to connect to the internet and interact with the program.

**4.2.3 Software Interface:**

The only requirement this program has is to have Java Development Kit 8 installed on the laptop or PC that the user wishes to use.

**4.2.4 Communications Interface:**

The very first connection our program will make is with the database to verify the user’s log in information or to store new user’s information. The second connection is with the three APIs used to retrieve data and process the returned information using JSON.

**5 Non-Functional Requirements**

5.1 Performance Requirements:

**5.2 Safety and Recovery Requirements:**

The user shouldn’t be affected when the program crashes because we’ve added the ability to save coins to the “Saved coins” list which can be accessed at any time.

**5.3 Security Requirements:**

The program will require the user to create an account which consists of an email, username, and password that must be longer than 6 characters and includes a number and a special character. The information will be stored in the database.

**5.4 Policy Requirements:**

As of now we don’t have any policy requirements, however, we might have the user agree to our terms and conditions after signing up.

**5.5 Software Quality Attributes**

**5.5.1 Availability:**

The program should always be available to be used at any given time except for when there’s no internet access, account issues, or during updates/ maintenance.

5.5.2 Correctness:

**5.5.3 Maintainability:**

The developers should make sure that the APIs used are still running well with no delays or whatsoever.

**5.5.4 Portability:**

The program should be runnable on any machine that meets the Performance Requirements (5.5.1) specified in this document.

**5.6 Process Requirements**

5.6.1 Development Process:

**5.6.2 Time Constraints:**

The program will be developed and modified starting from August 22nd, 2020 to December 1st, 2020.

**5.6.3 Cost and Delivery Date:**

This product will be free of charge to customers. The product should be ready to our stakeholders and the public on November 27th, 2020.