**Team 3**

**Software Engineering Final Project**

**Nicole A. Cill**

**Yuwei Zhou**

**Keonta S. Hafley**

**Keith Estrella**

**Brian Yao**

**11th December 2019**

**Written by Nicole A. Cill**

**Version: (1) Created: December 2nd, 2019**

**Table of Contents**

1. **Introduction**

**1.1 Purpose**

* 1. **Scope**

**1.3 Definitions, Acronyms, & Abbreviations**

1. **Overall Description**

**2.1 Product Perspective**

**2.2**

**2.3**

1. **Requirement Engineering**

**3.1**

**3.2**

1. **System Modeling**

**4.1**

1. **Architectural Design**
2. **Software Design and Development**

**6.1**

1. **Database Design**
2. **Project Management**

**8.1 Progress Log**

1. **Introduction**

Initially, Team 3 struggled to finalize an idea for our final project. We regrouped and reprogrammed multiple times until we had an ideal and fully functional system. The system interface will retrieve stock information from a database. After collecting the data, we will use a variety of features to display and interpret the information.

* 1. **Purpose**

This project allowed us all to work in a collaborative environment in a software engineering team. We were also able to have a better understanding of the IEX Cloud API and how to utilize it. Learning how to work with the command calls of IEX Cloud API, made us more comfortable with using RESTful API’s as well. Our team was even able to neatly and clearly present the data of our [hypothetical] clients.

* 1. **Scope**

The software we developed will use IEX Cloud API to obtain the stock information. This information will be recollected/ updated on a daily basis. We will be receiving data from three stock symbols: (1) rfem, (2) googl, (3) appl. Our database will collect about twenty parameters, such as: basic company info, a description of the company, which stock exchange the company uses, etc. The software will also report the maximum price, minimum price, average price, and median price based on the data collected over a 30-day time span.

* 1. **Definitions, Acronyms, & Abbreviations**

All of the definitions, acronyms, and abbreviations that will be seen throughout the document will be listed here.

* IEX Cloud API
* RESTful API
* JavaServer Pages (JSPs)
* MVC

1. **Overall Description**

Overall, we wanted to create a software that would make analyzing the stock information of a 30-day time period as easy and clear as possible. We wanted the user to be able to select any day [or range of days] to display and compute several operations to the data. The user will be able to: (1) simply list the data, (2) view the changes in data overtime, (3) compute the mean/min./max./median. With the use of graphs and tables, all of the information will be plainly displayed for users to view.

**2.1 Product Perspective**

Our system is not one of a kind. There are several software’s that have already been developed and can do the same functions ours can do. Similarly, to how we produce this system ourselves, many people can do it too. However, after carefully constructing the best layout and program to be able to do the basic functions we wanted, we finally created one that works. Stock trading websites/apps such as: Robinhood, Acorn, Stash, etc. all can do the things our software can do. Therefore, we like to think of ours as the simplest form of stocking information.

**5. Architectural Design**

1. The user makes a request to the web server and will receive a response using JavaServer Pages (JSPs).
2. The web server will host the applications various layers which will conform with MVC:
   1. Presentation layer will allow users to interact with the application via HTTP requests and responses rendered in a browser.
   2. Application layer manages the flow of the application, implements business logic and liaises with the data layer to process requests from users and their responses. It is open-source, and third-party products reside here.
   3. Data layer handles domain data and provides persistence and retrieval services for the database.
3. The database is where the data is persisted and retrieved.
4. The web services allow for interaction with other applications.

**6. Software Design and Development**

**6.1 Daily Update Routine**

By using Linux crontab, we are able to schedule the daily update routing “iextry.py” to kick off daily at 7:00 a.m.

**7. Database Design**

Our database is simple but effective. There are two indexes: (1) company and (2) master. With the master index pointing/ collecting information from the company index, we are able to list out the basic but necessary information for each of the companies. The information the database lists out, but is not limited to,

1. The CEO’s name
2. The company’s symbol/sigil
3. The company’s address
   1. The street
   2. The city
   3. The zip codes
   4. The state
   5. The country
4. The company’s name
5. 3-4 sentence description of the company
6. The amount of employee’s the company has
7. The stock exchange used for the company
8. The company’s industry
9. The issue types
10. The company’s phone number
11. The sector
12. The securities name
13. The company’s website link

**8. Project Management**

Our group experiences several setbacks. Technically, we did not officially begin to make progress until November 17th. After adding Keonta to our group and receiving some guidance from Dr. Wong we were able to start.

**8.1 Progress Log**

|  |  |
| --- | --- |
| September 25th | Teams were assigned.  Team leader: Yuwei  Team members: Nicole C., Keith E., Austin S. |
| October 4th | Austin S. was removed from Team 3 and replaced with Brian Yao. |
| October 5th | All email addresses were obtained, and Nicole C. sent out a group email. |
| October 8th | Yuwei, Keith, and Nicole had brief (5 minute) meeting to discuss the start of the project. |
| October 15th | Yuwei, Keith, and Nicole had meeting to discuss the project and the approach the team will take. |
| October 23rd | Team meeting with Yuwei, Brian, and Nicole.  We further discussed the approach we would  take. Clarified the purpose and extent of the  project. |
| November 4th | Keonta H. was added to Team 3. |
| November 17th | Dr. Wong provided examples, models, and  templates [of System diagrams,  Architecture Layer diagrams, and  Software Stack diagrams] to help us get  started. As well as, scheduled a group phone  call with us. |
| November 18th | Team 3 had a Skype phone call with Dr. Wai-Tak Wong. We were given job roles, and task. We uploaded all the initial information on Trello. |
| November 25th | Team 3 had a follow up phone call with Dr.  Wong to make sure we were staying on track  and making progress.  Brian created the architectural design for the program.  Keonta created the database design for the program. |
| November 27th | Dr. Wong put a cron schedule to update  Keonta's python program to update the  daily data. |
| December 3rd | Brief team meeting to discuss the remainder of  the project. Also, gathered more information to  add to the document. |
| December 9th | Team 3 had a final follow up call with Dr.  Wong to discuss the presentation. |
| December 11th | Team 3 had a Skype call to discuss added  features and to schedule a team meeting for to  finalize the presentation. |