**Stonks go brrr**

**Daniel Masarsky & Tommy Thongsythavong**

**12/15/2020**

**CSC521: Dr.Kaur**

Table of Contents

* Student Objectives
* Problem Specification
* Solution Processes and Design
  + Requirement Collection and Analysis
  + General Design Concepts
    - Use Case Diagram
    - Prototype of the final product
* Benchmark Specifications
* Tool List
* Time Schedule
* Grading Scheme
* List of Deliverables

Student Objectives

**Daniel Masarsky-** Personally Stocks have always seemed interesting to me, and I figured what better way to learn about stocks then to build an application that predicts them. Also, in CSC 475 I was tasked with creating a project involving AWS and I remember enjoying AWS and wanting to use the tools provided with AWS in my capstone in some capacity. As such my primary objective is to create a Stock Market Predictor that is hosted on AWS for ease of access.

**Tommy Thongsythavong-** After taking the machine learning course at Salem State I wanted to find a project that would require using machine learning techniques I learned. Also, I could use this stock market predictor to make better predictions than my guessing and improve upon that. This project is not a project to create something new and amazing but as a learning experience and to understand the process behind how a Stock Market Predictor is created.

Problem Specification

With the stock market as volatile as it is, wouldn’t it be nice to have an application that could predict (within a modicum of success) said market? So we will make a Machine learning stock market predictor based on various algorithms. The application will display the predicted closing prices of the next year on various stocks. This application instead of being one you need to download and run off your own machine will be hosted on Cloud servers to allow for ease of access.

Solution Processes and Design

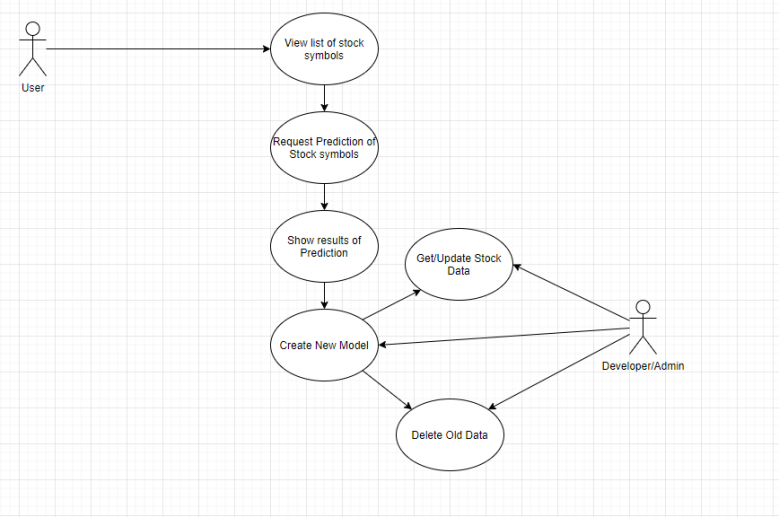
Requirement Collection and Analysis

**Requirement List:**

* If purchased what functions will the software provide?
* An online stock market predictor
* How useful is the software for the User?
* It's useful to make predictions on the stock market for users who do not have any other knowledge of the stock and do not have time to learn about it
* What benefit does the software possess?
* Allows Users to make more informed decisions when deciding on which Stocks to invest in

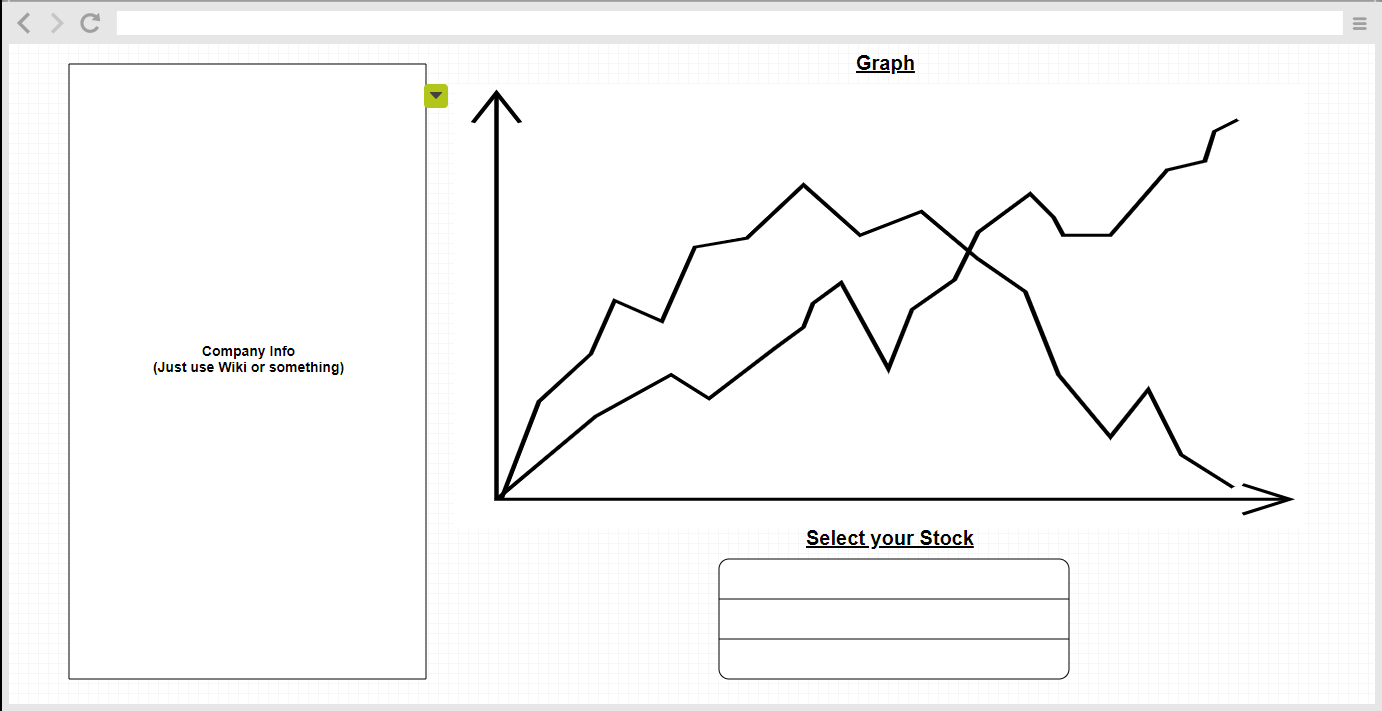
General Design Concepts

Use Case Diagram

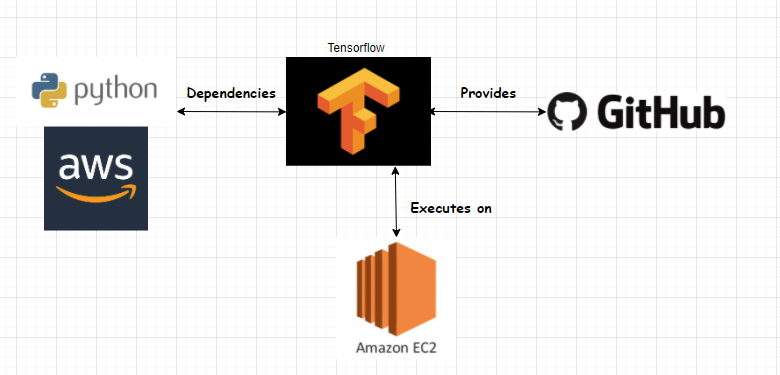


The User will be able to see the stocks from which they can choose from. From there the User requests predictions of Stocks, specifically the closing price of the day, and shows the User the results of said predictions creating a New Model that Gets/Updates the Stock Data while also deleting old data. Whereas the Developer/Admin can also view the deletion of old data as well as get and update stock data. The Admin/Developer can also access the creation of a new model.

Prototype of the Final Project

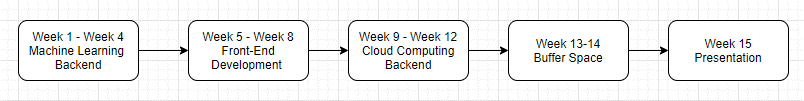


Our application will take the shape of a browser-based application. This application will display a graph with the closing price of the selected company. Selecting the company, which you want to see information about will be done with the search feature known as ‘Select your Stock’. Finally, this application will also display some rudimentary information about said company. Information that can easily be found on websites such as Wikipedia.



Benchmark Specifications

For the Benchmark Specifications we intend to split the project up into 3 major parts. These 3 parts will be; the machine learning portion of the backend. Followed by the frontend user interface development, finishing off with the cloud computing portion of the backend development. As such the 3 major Benchmarks will be completing those 3 portions of development.



Tool List

* CPU – Ryzen 7 2700x – CPU used to do computations on the model.
* VSCode (Standard editor) – Text editor
* Atom (Text Editor) – Text editor
* <https://www.invisionapp.com/> - Prototyping tool used to create diagrams.
* Python 3 – Main language used to create the backend
* HTML – Standard markup language for documents designed to be displayed in a web browser.
* CSS – Cascading Styling Sheet (Design for HTML) – Style Sheet language used for describing the presentation of a document written in a markup language.
* TensorFlow – Software library for machine learning.
* Keras – Open-source library that provides a Python interface for artificial neural networks. This acts as an interface for the TensorFlow library.

Time Schedule

For the Time Schedule pertaining to this project, We know that once a week every week starting the Spring 2021 semester we will meet with our advisor Kaur over Zoom to discuss the project, this may include; any problems we may run into, or any progress we made, or it could even just be as simple as checking in to let her know everything is going according the schedule we gave ourselves. Assuming that Spring 2021 will be a 15 week long semester what we project to be our Time Schedule would breaking the first 12 weeks into 4 week parts; the first 4 weeks being dedicated to the Machine Learning portion of the Backend, ideally finishing with it by the end of the first 4 weeks and from there dedicate the next 4 weeks to the Front-End development, and using any extra time between the two as buffer space to make sure there is no problems with the application. So, come the end of the 8th week of the semester, roughly half way through the semester we move on to the Cloud Computing portion of the backend. Dedicating another 4 weeks to that portion, and also any extra time gained from potentially finishing early will be dedicated to buffer time making sure there are no problems and improving upon whatever we can. This then would leave us with approximately 3 extra weeks of the semester, 2 if you do not count the last week being finals week, as we would like to be done before the 15th week of the semester. These 2 weeks are to be used to complete our presentation as well as to give us the extra time needed in case something we cannot predict occurs and give us the extra breathing room to slow down if we need to.

Grading Scheme

For the actual Grading Scheme of our project there are multiple different parts of the project to consider. There's the Front End which is the User Interface, and there is also the Back end consisting of the Machine Learning portion of the project, of which Tommy is in charge of development with Daniel providing a secondary development role and there is also the Cloud Computing portion of the project, of which Daniel is in charge of while Tommy provides a secondary role for that. Not to mention the actual presentation of the project itself is also important, a role shared by the two us. Thus, what we are proposing for what we believe seems like a fair Grading Scheme would be the following:

Front-End: 25%

Back-End: 50%

Presentation: 25%

The Front-End being what both of us believe to be the least intensive part of the project believe the grading scheme should reflect that. As such we believe the Front-End should only be 20% of the grade. The Back-End however is what we believe to be the bulk portion of the project, as it will consist of two primary portions; The Machine Learning and The Cloud Computing portions. Two reasonably time-consuming parts for what the project is going to be, as such we believe it should be the bulk of the grade, accounting for 50% of the actual grade, since without the Backend we have no project. Finally, we come to the last 25% of the grade, which we believe should account for the actual presentation of the project. As the presentation is still a key part of the project its grade should account for it, putting the presentation at the same weight as the Front-End portion of the project as the two possess similar end goals of enticing the viewer to like our project.

List of Deliverables

Deliverables:

* + - Original Proposal and Presentation File
    - Amendments to the Proposal
    - Appropriately Commented Source Code
    - Documentation of Project Functionality
    - Sample Output (Screenshot and/or report)
    - Executables and/or projects
    - Presentation Documents, including any presentation file
    - Project Journal: A narrative of the progress of the project, in clear, concise English, including any problems encountered and how said problems were addressed
    - Project Post Mortem: a summary of what was learned from the project and a discussion of how various aspects of the project might have been approached differently
    - A list of what areas of the proposal (if any) were not completed, including a reason why
    - Presentation of the completed project, including screenshots of the functioning project
    - Cloud Based Stock Market Predictor
    - The ability to see the Predicted Values in selected days future (To be determined days)