

Florida International University
Knight Foundation School of Computing and Information Sciences

Software Engineering Focus

Final Deliverable

Project Title: Stock Picker

Team Members:

Mark Charles Oliver
Sebastian Cubillos
Gino Surace

Product Owner(s):

Mark Charles Oliver

Instructor: Masoud Sadjadi

The MIT License (MIT)
Copyright (c) 2016 Florida International University

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Abstract

This document introduces the information necessary to gain a good understanding of the Stock Picker program, a bot that helps the user pick out stock options. The system leverages learning algorithms and financial data analysis to provide personalized and accurate stock option recommendations. The main focus was optimizing the back-end to ensure the systems adaptability, accuracy, and transparency to make it a viable tool for investors.

Introduction

Current System

This is the first iteration of the Stock Picker program, as such the scope for the project was rather small. Currently it consists of a small bot that scrapes the historical stock data from Nasdaq and current day stock data Yahoo! Finance. The reason both sites are scraped is that Nasdaq does not post the current day whereas Yahoo! Finance does. Provided the saved data is less than one month out of date, this allows for the program to remain current when visiting only one site after the initial set up.

User Stories

The following section provides the detailed user stories that were implemented in this iteration of the Stock Picker project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories

User Story #1 - NextCloud Communications

Title: As developer, I want to learn NextCloud group meetings, NextCloud groups and file sharing so we can communicate for this project

Acceptance Criteria

Acceptance Criteria 1: All members join the NextCloud group setup for this class

Acceptance Criteria 2: Make a NextCloud meeting

Acceptance Criteria 3: Store and access files in NextCloud's virtual drive

Estimate: 10 points

User Story #2 - Coding Environment Set up

Title: As a software developer, I would like to have the necessary development environment on my personal computer to create the program

Acceptance Criteria

Acceptance Criteria 1: A python environment with the required libraries and the ability to install more as needed.

Acceptance Criteria 2: The ability to run Jupyter Notebooks

Estimate: 10 points

User Story #3 - Nasdaq Scraping

Title: As a user I want historical stock data so any predictions can be accurate

Acceptance Criteria

Acceptance Criteria 1: Program visits Nasdaq's site for the stock

Acceptance Criteria 2: Program navigates to the 5 year historical data

Acceptance Criteria 3: Program downloads the CSV to the correct folder

Estimate: 20 points

User Story #4 - Yahoo! Finance Scraping

Title: As a user I want current day stock data so any predictions can be accurate

Acceptance Criteria

Acceptance Criteria 1: Program visits Yahoo's site for the stock

Acceptance Criteria 2: Program downloads the CSV to the correct folder

Estimate: 10 points

User Story #5 - CSV Appending

Title: As a developer I want to use stock data from a single CSV file to simplify training the machine learning algorithm

Acceptance Criteria

Acceptance Criteria 1: Program correctly appends the data from Yahoo! Finance to Nasdaq's resulting in a single CSV file

Estimate: 5 points

User Story #6 - Stock Data Automation

Title: As a user I want the program to be able to make predictions on multiple stocks so I can make informed choices about the predictions

Acceptance Criteria

Acceptance Criteria 1: The program navigates to Nasdaq and downloads the historical data for all the listed stocks

Acceptance Criteria 2: The program repeats this with the current data for Yahoo! Finance

Acceptance Criteria 3: The program appends all the CSV files correctly

Estimate: 10 points

User Story #7 - Model Design and Initial implementation

Title: As a user I want the program to make predictions using the data collected so I can make good choices when buying stock options

Acceptance Criteria

Acceptance Criteria 1: The model reads the input data,

Acceptance Criteria 2: The price prediction of the model is comparable to a known price from

Estimate: 20 points

User Story #8 - Model Verification and Refining

Title: As a user I want the predictions the program makes to be as accurate as possible

Acceptance Criteria

Acceptance Criteria 1: The model reads the input data

Acceptance Criteria 2: The price prediction of the model is comparable to the

Estimate: 20 points

Pending User Stories

NextCloud Integration

Title: As a user I want to be able to access and run the program via NextCloud

Acceptance Criteria

Acceptance Criteria 1: The program saves data to the networked drive

Acceptance Criteria 2: The program can run as a NextCloud App

Project Plan

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components chosen for this project.

Hardware

- Computers running Linux, Mac OS or windows.
- Internet Connection

Software (Backend)

- Jupyter Notebooks
- Pandas, Numpy, Tensorflow, Sklearn, Keras, Selenium, Glob
- Aioshutil, Regex for the scraper getClosingStocks
- Logging, Collections, Math, and PDB for Data Science

Sprint Planning

Sprint 1

Attendance: Mark Charles Oliver, Sebastian Cubillos, Gino Surace

Sprints pulled into the story:

User Story #1: NextCloud Communications

User Story #2: Coding Environment Set up

Estimated Velocity: 20

All members agreed to work on both stories

Sprint 2

Attendance: Mark Charles Oliver, Sebastian Cubillos, Gino Surace

Sprints pulled into the story:

User Story #3: Nasdaq Scraping

Estimated Velocity: 20

All members agreed to work on both stories

Sprint 3

Attendance: Mark Charles Oliver, Sebastian Cubillos, Gino Surace

Sprints pulled into the story:

User Story #3: Nasdaq Scraping

User Story #4: Yahoo! Finance Scraping

Estimated Velocity: 30

All members agreed to work on both stories

Sprint 4

Attendance: Mark Charles Oliver, Sebastian Cubillos, Gino Surace

Sprints pulled into the story:

User Story #4: Yahoo! Finance Scraping

User Story #5: CSV Appending

User Story #6: Stock Data Automation

Estimated Velocity: 25

All members agreed to work on all three stories

Sprint 5

Attendance: Mark Charles Oliver, Sebastian Cubillos, Gino Surace

Sprints pulled into the story:

User Story #7: Model Design and Initial implementation

Estimated Velocity: 20

All members agreed to work on all three stories

Sprint 6 - Needs work

Attendance: Mark Charles Oliver, Sebastian Cubillos, Gino Surace

Sprints pulled into the story:

User Story #8: Model Verification and Refining

Estimated Velocity: 20

All members agreed to work on all three stories

Sprint 7

Attendance: Mark Charles Oliver, Sebastian Cubillos, Gino Surace

Sprints pulled into the story:

User Story #Final - Prepare for showcase

System Design

Currently the code consists of two main portions: the first part is a web scraper that downloads the CSV files of stock price data from Nasdaq and Yahoo! Finance. It then appends the two files together so that only a single file for each stock exists

The second portion analyzes the data with a linear regression algorithm to determine which out of the stocks would be the best to purchase options on. It is currently best run on Thursdays specifically, or using CSV files that stop on Thursday, as that was determined to be the day with the best increases. The algorithm is trained with the historical data and compared to the known results of price increase or decreases.

System Validation

Test Case 1: Web_Scraper_01

Test steps:

1. Boot up the Notebook “getClosingStocks”

2. Run setup code cell
3. Run web scraping cells for Nasdaq page

Expected results: CSV of chosen stock saved in Data file

Actual results: As expected

Status(Pass/Fail): Pass

Test Case 2: Web_Scraper_02

Test steps:

1. Boot up the Notebook “getClosingStocks”
2. Run setup code cell
3. Run web scraping cells for Nasdaq and Yahoo pages
4. Run the CSV appending cells

Expected results: A single CSV of chosen stock saved in Data file containing both historical and current day data

Actual results: As expected

Status(Pass/Fail): Pass

Test Case 3: Web_Scraper_03

Test steps:

1. Boot up the Notebook “getClosingStocks”
2. Run setup code cell
3. Run web scraping cells for multiple Nasdaq and Yahoo pages
4. Run the CSV appending cells

Expected results: 32 CSVs of chosen stocks saved in Data file containing both historical and current day data

Actual results: As expected

Status(Pass/Fail): Pass

Appendix

Sprint 1 Review Minutes

Product Owner: Mark Oliver

Scrum master: Gino Surace

Stories Completed in Sprint:

Story ID	Story Title	Assigned To	Status
2	As a software developer, I would like to have the necessary development environment on my personal computer to create the program	Mark, Gino, Sebastian	Completed

Stories NOT Completed in Sprint

Story ID	Story Title	Assigned To	Status
1	As developer, I want to learn NextCloud group meetings, NextCloud groups and file sharing so we can communicate for this project	Mark, Gino, Sebastian	In progress

Total Velocity: 15

Sprint 2 Review Minutes

Product Owner: Mark Oliver

Scrum master: Gino Surace

Stories Completed in Sprint:

Story ID	Story Title	Assigned To	Status
2	As developer, I want to learn NextCloud group meetings, NextCloud groups and file sharing so we can communicate for this project	Mark, Gino, Sebastian	Completed

Stories NOT Completed in Sprint:

3	As a user I want historical stock data so any predictions can be accurate	Mark, Gino, Sebastian	Completed
---	---	-----------------------	-----------

Total Velocity: 15

Sprint 3 Review Minutes

Product Owner: Mark Oliver

Scrum master: Gino Surace

Stories Completed in Sprint:

Story ID	Story Title	Assigned To	Status
3	As a user I want historical stock data so any predictions can be accurate	Mark, Gino, Sebastian	Completed

Stories NOT Completed in Sprint:

Story ID	Story Title	Assigned To	Status
4	As a user I want current day stock data so any predictions can be accurate	Mark, Gino, Sebastian	Completed

Total Velocity: 15

Sprint 4 Review Minutes

Product Owner: Mark Oliver

Scrum master: Gino Surace

Stories Completed in Sprint:

Story ID	Story Title	Assigned To	Status
4	As a user I want current day stock data so any predictions can be accurate	Mark, Gino, Sebastian	Completed
5	As a developer I want to use stock data from a single CSV file to simplify training the machine learning algorithm	Mark, Gino, Sebastian	Completed
6	As a user I want the program to be able to make predictions on multiple stocks so I can make informed choices about the predictions	Mark, Gino, Sebastian	Completed

Stories NOT Completed in Sprint:

None all stories pulled in completed

Total Velocity: 20

Sprint 5 Review Minutes

Product Owner: Mark Oliver

Scrum master: Sebastian Cubillos

Stories Completed in Sprint:

Story ID	Story Title	Assigned To	Status
7	Model Design and Initial implementation	Mark, Gino, Sebastian	Completed

Stories NOT Completed in Sprint:

None all stories pulled in completed

Total Velocity: 20

Sprint 6 Review Minutes

Product Owner: Mark Oliver

Scrum master: Sebastian Cubillos

Stories Completed in Sprint:

Story ID	Story Title	Assigned To	Status
8	Model Verification and Refining	Mark, Gino, Sebastian	Completed

Stories NOT Completed in Sprint

None all stories pulled in completed

Total Velocity: 20