SOFTWARE REQUIREMENTS SPECIFICATION

A Purposeful Walk Down Wallstreet

Nabeel Asghar | Michael Shields | Shojib Miah | Michael Chen
Wayne State University – February 11th, 2020

Senior Capstone Project

Revision History

Date	Description	Author	Comments
2/11/2020	Version 1	Nabeel Asghar, Michael Shields, Shojib Miah, Michael Chen	Original document that we submitted to canvas
2/12/2020	Version 1.5	Nabeel Asghar	Fixed the functional and nonfunction requirements dependencies. Also put one of the NFR into FR.
3/18/2020	Version 2.0	Nabeel Asghar	Fixed spacing issues and FR wording.
4/13/2020	Version 3.0	Nabeel Asghar	Fixed the functional and nonfunctional requirements. Deleted a few that made no sense.

Table of Contents

1. Introduction

This document will provide an overview of the Software Requirements Specification agreed upon between the client and the developers.

The General Motors FinTech (financial technology) team is responsible for improving an existing application that makes predictions about the future behavior of various financial indexes. The application indicates whether or not one should buy, sell, or hold a group of assets that are currently under management.

General Motors has been working with Wayne State University for 11 semesters on applications designed by students that utilize machine learning techniques in order to predict movements in various financial markets. Continuing with this past work, this team aims to enhance the application so it can conduct a thorough analysis of past financial data and make better decisions about managing a hypothetical portfolio of assets with increased accuracy and more robust and modular code. We will be implementing a prediction system based off of macroeconomic variables such as inflation and unemployment rate.

1.1 Purpose

The purpose of this document is to provide a description of the requirements associated with the GM FinTech Tableau application. This document will provide the guidelines to implement all the required features. It will show the design of the application.

The purpose of this entire project from General Motors is to find a way to predict long-term stock market trends. Although 11 semesters of work have gone into this project, the model still has not breached a very accurate prediction rate. We hope to predict long-term, not short term, trends of the entire stock market as whole, not just to predict stock prices, but incoming recessions as well.

1.2 Scope

The current scope of this project is that it takes data from a free data exchange server, servers such as YAHOO Finance and Quandl. It then analyses this data to predict long term trends onto our Tableau interface.

1.3 Definitions, Acronyms, and Abbreviations

MySQL: an open-source relational database management system we will use.

YAHOO Finance: a source of financial data from which we grab stock prices.

Quandl: an API (Application Programming Interface) that makes it easier for us to grab financial data specific to macroeconomic variables.

Tableau: a visualization tool that allows use to interact with our data in our database and visual in graphs and tables.

XGBoost: this is an open-source tool for machine learning in Python.

S&P 500: S&P (Standard and Poor) measures the value of the 500 most valuable stocks listed on the stock market in the United States.

NYSE: the NYSE (New York Stock Exchange) is a stock exchange in New York and is the largest stock exchange in the world.

ETF: an ETF (Exchange-Traded Fund) is a basket of securities which means a collection of stocks that you can trade in the NYSE.

CARZ: this is an index that tracks global auto manufacturing companies such as Ford, GM, Tesla, etc.

CMA: CMA (Cross-Moving Average) is an algorithm that detects when a shorter period moving averages intersects a longer period moving average.

EMA: EMA (Exponential-Moving Average) is a moving average that places a heavy emphasis on newer price points rather than older price points.

ARIMA: ARIMA (Auto-Regressive Integrated Moving Averages) is data-science model that allows you to predict future values by basing it off of momentum.

FRL: FRL (Fibonacci Retracement Lines) is a technical analysis tool for determining resistance and support levels.

MACD: MACD (Moving Average Convergence Divergence) is a model that tracks the relationship between two EMA lines, one shorter period and one longer period.

Random Forest Regression: this is a form of regression that is calculated using machine learning and decision trees.

SVM: SVM (Support Vector Machine) is a machine learning classifier.

GDP: Gross Domestic Product (GDP) is measure of all goods and services made by a country.

Misery Index: this is an economic indicator of how the average U.S. citizen is doing.

Adj. Close: this is the adjusted close price of stock factoring in dividends and splits.

1.4 References

MySQL documentation

YAHOO Finance

Quandl documentation

1.5 Overview

The rest of this document will go further in depth about how the program works, how it receives its data, analyzes it, and then displays it in a visually appealing yet understanding way. There will be more information on how the product can be used and what limits there are to it and what factors affect the product as a whole. It will then transition to information about the specific requirements. This will include external interface, functional, and non-function requirements that will be used in the project's design, implementation, and testing.

2. General Description

Below we will provide a general description of the application and its layers in its entirety. We will discuss the application's functionality, user interface, user interaction, as well as the backend code that will allow the applications' functions to execute. This will provide a picture of how the forecasting, directional accuracy, and buy/sell strategies come together to create a functioning application. We will finalize this section by discussing the constraints, dependencies, and assumptions which will lead into the next section covering the actual requirements of the application.

2.1 Product Perspective

The system currently consists of a user interface that collects data from our MySQL database and displays it in a Tableau worksheet. The data obtained from the MySQL database is fetched from YAHOO Finance as well as Quandl. YAHOO Finance is a financial data exchange that focuses on stock and index prices, while Quandl includes not only financial data but also economic and alternative data as well.

The application allows for enhanced user interaction to retrieve desired information. For example, data that is graphed and displayed in Tableau will allow the user to hover over specific data points on the graph to retrieve specific and expanded information pertaining to the specific data point. Tableau's interaction with the database is limited to solely the user interface display, meaning Tableau only has permission to read from the database.

The data fetched from YAHOO and Quandl is stored in a local MySQL database which is then leveraged by algorithms written in Python in order to forecast future market movements. Currently the application creates signals that indicate whether it is time to buy or sell using the algorithms written in Python, and these algorithms make their predictions based off of trends observed from different financial instruments such as the S&P 500, GM, CARZ, and others.

We plan on expanding these algorithms to instead of just generating buy and sell signals to also leverage the use of Macro Economic variables to create longer term price forecasts. The database that will store these variables will be able to be hosted on any system as long as the settings have been adjusted and access to the database has been granted.

2.2 Product Functions

Using the data fetched from Quandl as well as YAHOO finance the application will show predictions of long-term market movements for specific trading instruments such as the S&P 500, different automotive stocks and indexes, as well as pharmaceutical markets. Our plan is to use a mix of Macro Economic Variables as well as algorithms built around Cross Moving Averages (CMA), Exponential Moving Averages (EMA), as well as Auto-Regressive Integrated Moving Averages (ARIMA) to attempt to accurately forecast long term price trends and hopefully signal possible upcoming market corrections.

Currently the application uses Algorithms such as Fibonacci Retracement Lines (FRL), Moving Average Convergence Divergence (MACD), EMA, CMA, and other trading strategies to produce buy/sell signals as well as ARIMA, Random Forest Regression, XGBoost, and other algorithms to forecast shorter term price movements.

2.3 User Characteristics

The application is designed for data analysts with sufficient knowledge of financial markets who are able to work with a business intelligence application like Tableau. The user will be able to access the Tableau Dashboard to view the different forecast charts and portfolio trackers but will also be able to look at the raw data depending on permission level. The user will then want to be able to read the forecast predictions and raw data and determine if there are any discrepancies in the reporting of data or the predictions of the algorithms prior to them taking any action based on the results given by the application. Therefore, this application requires someone who is able to interpret data and act accordingly while knowing the limitations of the application's functionality.

2.4 General Constraints

An internet connection will be required in order for the data contained in the database to remain up to date. This is due to the fact that the data must be pulled from an online data source that requires an internet connection to access, and also because the system does not actively monitor the data in any dynamic way. The capacity of the MySQL database poses another restraint in the capabilities of this application as if the historical data exceeds the limit then the newer data pulled from YAHOO Finance will need to replace the old data.

Another constraint relating to obtaining the data pertains to the way the data is accessed. Any change in the data source or the methods used to obtain the data can cause errors when extracting the data, which results in inconsistent and improper computations. The application uses predetermined data sources to remedy this problem, but any additional data sources will require an update to the backend technology of the application.

2.5 Assumptions and Dependencies

The two primary dependencies required for this application are YAHOO Finance's financial exchange data as well as Quandl's economic data. Any new data sources added will require work by a developer to ensure the application remains stable and consistent.

The overall setup of the system requires multiple different dependencies and any error made in the installation of these dependencies can result in errors preventing the application from running.

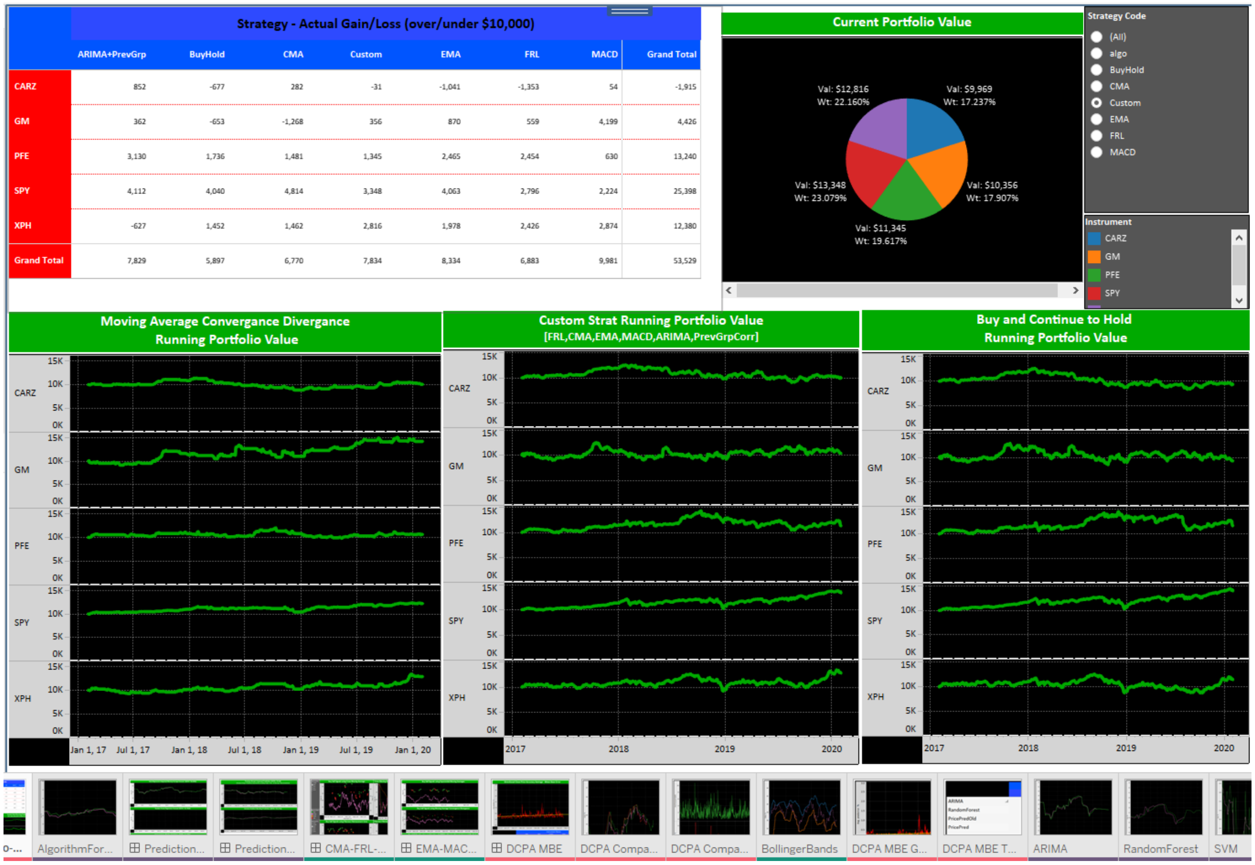
The sensitive nature of this application means extra care must be taken when following the installation guide, and every instruction must be followed in a strict manner.

It is assumed that the database and backend code will be stored and hosted locally. If there is any issue hosting the application and any relevant technologies, then the application will not function properly. Furthermore, it is assumed that all data sources retrieved from Yahoo Finance are configured with the data columns Date and Adjusted Close Price at minimum, and all data sources retrieved from Quandl are configured with the data columns Date and Statistic.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interface

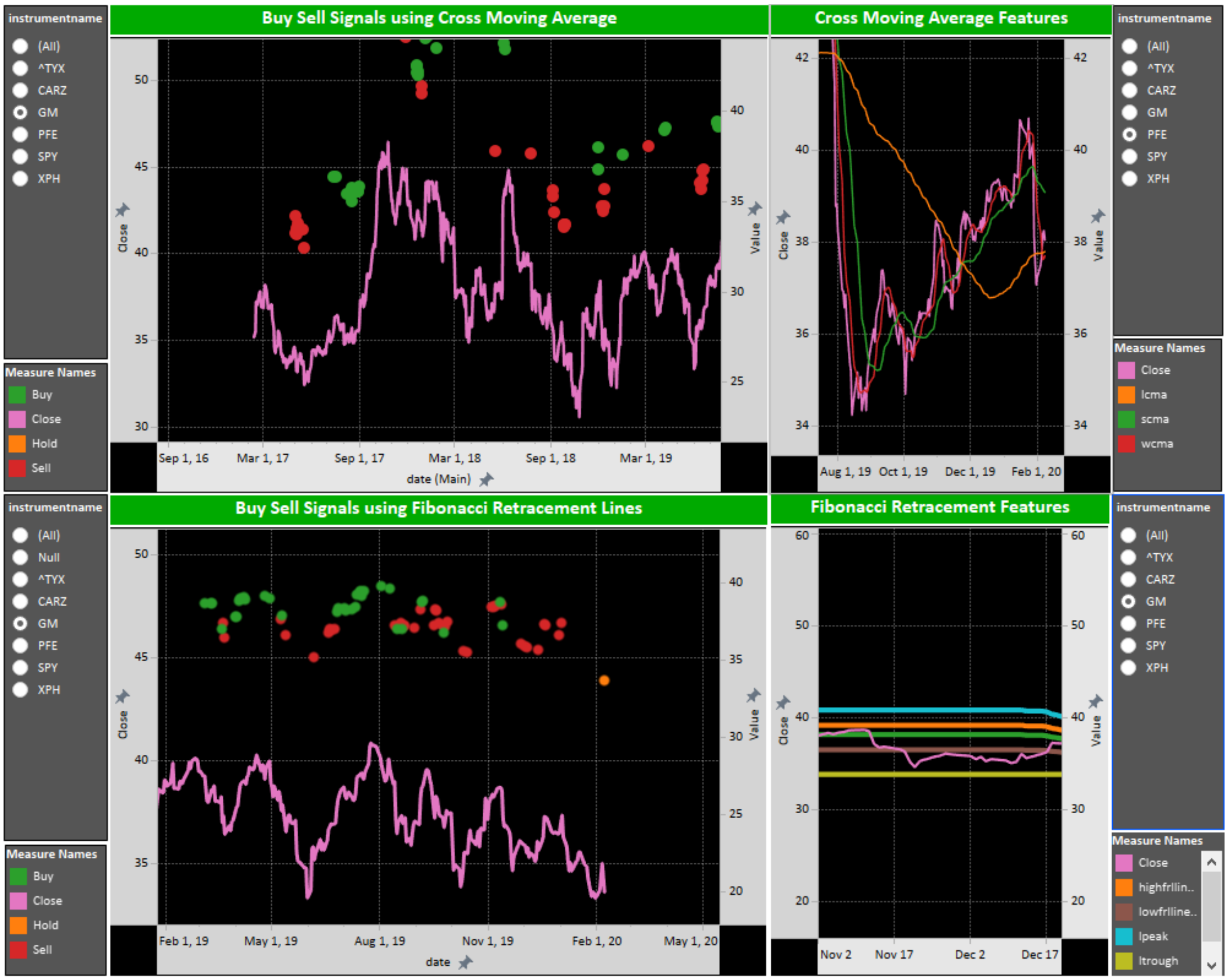


Strategy - Actual Gain/Loss (over/under $10,000)

	ARIMA+PrevGrp	BuyHold	CMA	Custom	EMA	FRL	MACD	Grand Total
CARZ	852	-677	282	-31	-1,041	-1,353	54	-1,915
GM	362	-653	-1,268	356	870	559	4,199	4,426
PFE	3,130	1,736	1,481	1,345	2,465	2,454	630	13,240
SPY	4,112	4,040	4,814	3,348	4,063	2,796	2,224	25,398
XPH	-627	1,452	1,462	2,816	1,978	2,426	2,874	12,380
Grand Total	7,829	5,897	6,770	7,834	8,334	6,883	9,981	53,529

Strategy Code
○ (All)
○ algo
○ BuyHold
○ CMA
● Custom
○ EMA
○ FRL
○ MACD

Layout of Tableau dashboard

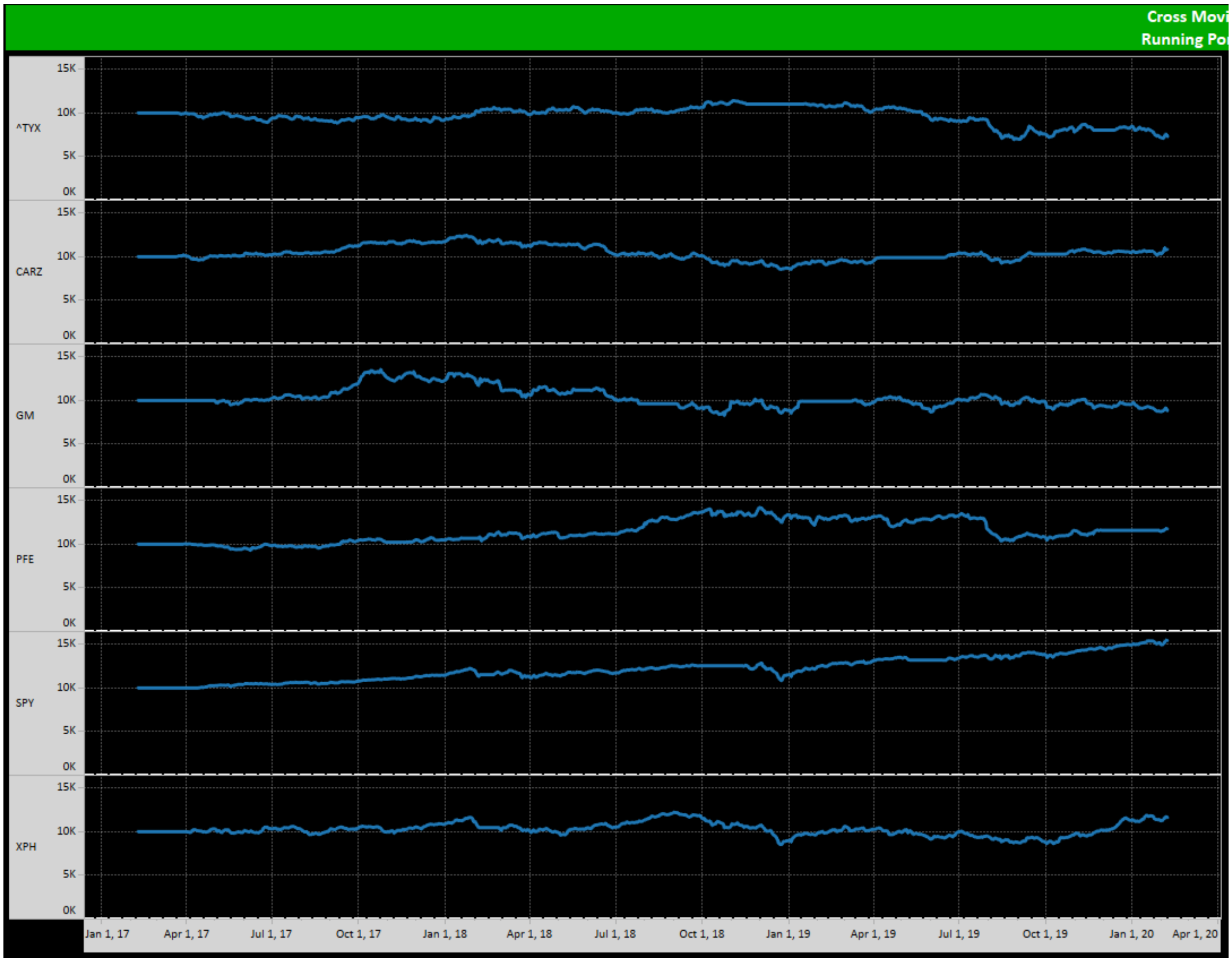


Layout of the buy and sell interface for CMA ad FRL

As you can see, the layout of Tableau interface is complicated but easy to read once the user figures out what to look for.

The user can select any instrument, the stock, at the side of the tool bar. They may select any other dashboards on the bottom of the interface where they can see how, for instance, CMA or FRL are doing.

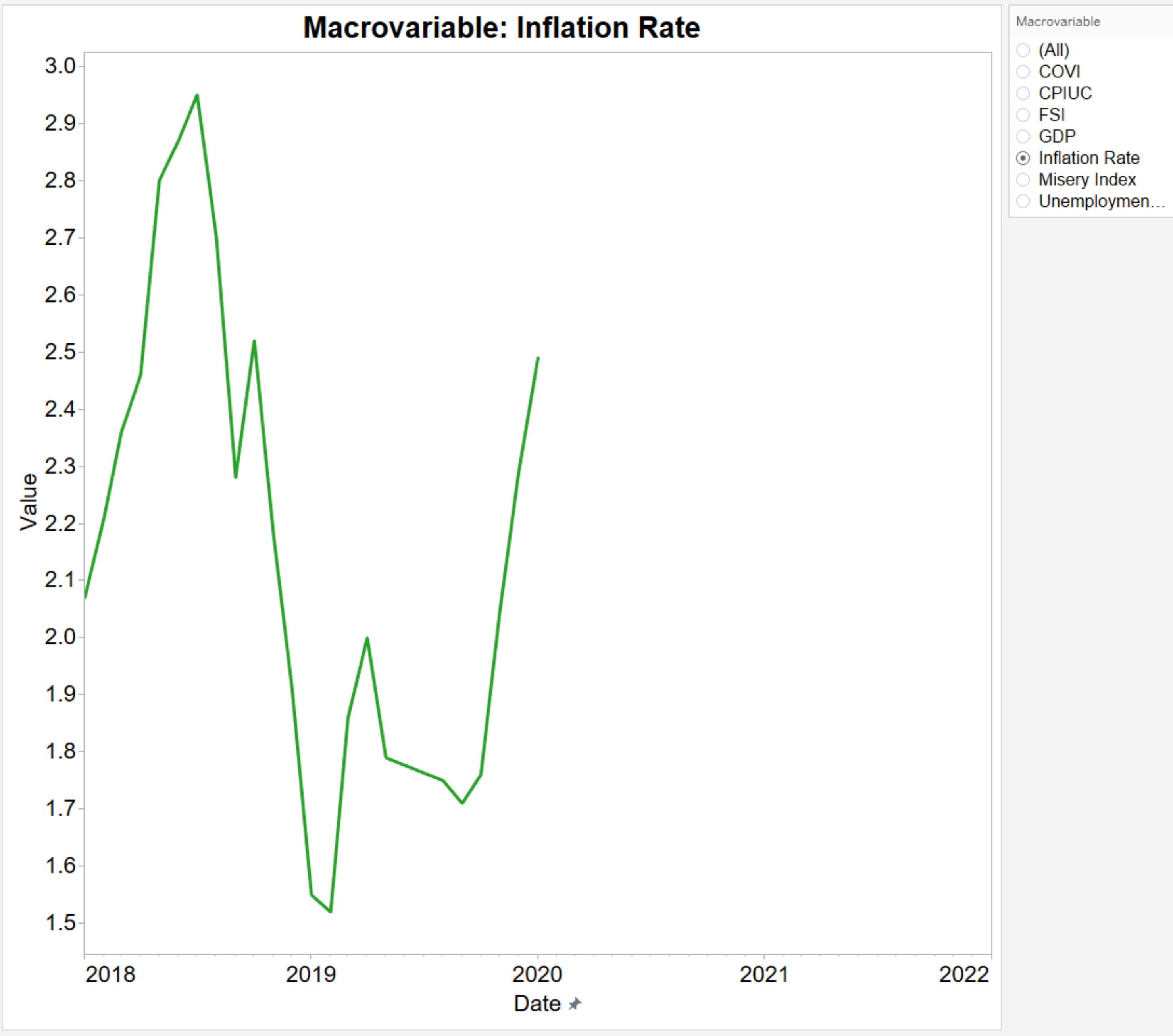
Although intimidating at first, the interface is cleverly done, however, it may need a few improvements for readability and simplicity.



Layout of different instruments or stocks

Here we have how different stocks and instruments did over the past few years. The information is up-to-date and queries it from YAHOO Finance and Quandl.

The next few pictures will be of the macroeconomic variables.



Macrovariable

○ (All)
○ COVI
○ CPIUC
○ FSI
○ GDP
● Inflation Rate
○ Misery Index
○ Unemploymen…

Dashboard of the macroeconomic variables

This is the dashboard which consist of the GDP, Misery Index, Unemployment Rate, Inflation Rate, etc. We hope to find a correlation between these to predict future economy states.

3.1.2 Hardware Interface

GM FinTech application is designed in Tableau which has flexible capabilities that can be easily integrated into any web browser for mass access. Tableau is mostly used in corporate environments within large organizations. No components of this application are intended for IoT.

GM FinTech is developed specifically for Windows machine (but also has compatibility with Mac OS) with current generation processors or newer with a steady network connection in order to safely retrieve and manipulate stock data. The device will require around 5 gigabytes of free disk space and 4 GB of RAM.

The Windows and Mac platform provides access to all the services need to run the application, most importantly to retrieve and process stock information from the YAHOO finance and Quandl.

3.1.3 Software Interface

The GM FinTech application requires the following Python libraries: Pandas API, MySQL server, YAHOO trader/Quandl, and more.

Trading data

- Real-Time Trading data is fetched from Yahoo trader and Quandl datasets using pyodbc API and Quandl python libraries, and then processed in a Python program and then sent directly to the database server where it is stored
- Incoming trading data consists of date and adjusted closing price for traded funds, SPY, XPH, PFE, GM, CARZ, ^TYX and more.

Macroeconomic/Algorithmic Manipulation and Decision

- Incoming trade data will be processed using various strategies and machine learning algorithms such as Arima, RandomForest, and buy and hold to name a few. Additionally, this data will be intertwined with data for inflation rate, unemployment rate, consumer price index, GDP, and various macroeconomic variables. Results of each calculation will be stored in a relational database.
- A few calculations may be done using stored SQL procedures, but most will be handled in Python’s middle processing layer.

Data View

- The data from the server will be linked to a Tableau custom dashboard application. This data includes inflation rate, unemployment, GDP, NPGDP, misery index, U.S. stock price, consumer price index, stock market confidence index, Adj. Close, Date, and the algorithm’s decision to buy, sell, or trade.
- Additional graphs will be shown on whether to buy and hold, SVM, SMA, CMA, as well as various other arithmetic calculating graphs.

3.1.4 Communications Interface

The GM FinTech application will have a network server that is running on the user’s machine and has access to the database. The server exists to retrieve information collected through Yahoo trader and Quandl to store it into the database. The product will then call the database system that stores the trade data and manipulate the trade data using many algorithms/macroeconomic variables.

The user’s machine will run the Python code which will update and process the latest trade information and update that information and sync it to the database for Tableau to display the data for the user.

3.2 Functional Requirements

ID	FR_0.1		
Name	Login information		
Description	Login username and password to use the application		
Rationale	Provides the ability to use the system		

Input	Username = root
	Password = password		

Output	Connection to the application		
Forward Dependency	Entire application		
Backward Dependency	None		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-1**		
Error Handling	User will have to contact the system administrator to request the username and password as there is no functionality to request a new username/password or reset it. Username and password are hardcoded as this is an internal application.		

ID	FR_1.0		
Name	Store macro variables into the database		
Description	Store inflation rate, unemployment rate, misery index, GDP, NPGDP, Stock market confidence index, consumer price index, and US stock price data into the database.		
Rationale	Stores essential variables needed for the project into database		
Input	Data received from Quandl datasets using pandas dataframe data structure to store and send to the database		
Output	Stored macroeconomic variables into the database		
Forward Dependency	Entire application		
Backward Dependency	FR 0.1		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-2**		
Error Handling	User will have to contact the system administrator to request the username and password as there is no functionality to request a new username/password or reset it. Username and password are hardcoded as this is an internal application.		

ID	FR_2.0		
Name	Analyze the results of the data and put them into our database		
Description	Using data stored in our MySQL database that has data from YAHOO Finance and Quandl, we analyze this data and calculate buy and sell points and predict prices.		
Rationale	Analyze data and store them into our MySQL database		
Input	Data received from Quandl and YAHOO Finance datasets using pandas dataframe data structure to store and send to the database		
Output	Stored data into the database		
Forward Dependency	FR_6.0, FR_ 7.0		
Backward Dependency	FR_1.0		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-2**		
Error Handling	User will have to analyze code to determine if there are any errors in it and analyze the MySQL Database to see if any issues are there.		

ID	FR_3.0		
Name	Introduce a new stock symbol, ^TYX (index 30-year bond)		
Description	Expand the applications in terms of reviewing more stocks. Data loads for any instrument must be seamless, fully automated and controlled in the Python environment		
Rationale	To increase the depth of the application and increase the market share by employing buying and selling opportunities to anther stock		
Input	YAHOO exchange data/Quandl Dataset		
Output	Data values to algorithms and other scripts as needed		
Forward Dependency	FR_6.0, FR_7.0		
Backward Dependency	FR_1.0		
Priority	**High**	**Priority Reason**	**Low Cost**
Priority Sequence	**H-3**		
Error Handling	Python code would need to be updated. If the error is due to exchange connectivity then another free exchange would need to be configured, field names will be impacted. User will not be able to address any error.		

ID	FR_4.0		
Name	Functionality of existing prediction techniques must be changed and improved using macroeconomic variables		
Description	Output of previously implemented strategies/algorithms must be recoded or reconfigured using macroeconomic variables. Delete certain scripts, as well as re-writing the existing scripts and show the results in a functional state on the front-end.		
Rationale	To improve the quality of existing predictive techniques (60-70% goal)		
Input	Algorithms and calculations in an existing data structure		
Output	Improved output of forecasts and predictions, visualized in a Tableau application		
Forward Dependency	None		
Backward Dependency	FR_1.0		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-4**		
Error Handling	Python code would need to be corrected, possibly some parts in the stored procedures as well. Users will not be able to address the error on their own.		

ID	FR_5.0		
Name	Introduce new algorithms/variables for predictions using adjusted close price data label		
Description	Application must provide the user with options to analyze a stock using different advanced algorithms/macroeconomic variables so they can confidently support business decisions		
Rationale	To increase the depth of the application		
Input	YAHOO exchange data, Quandl dataset		
Output	Predictions, engineered features, graphable data		
Forward Dependency	None		
Backward Dependency	FR_1.0		
Priority	**Medium**	**Priority Reason**	**Medium Scale**
Priority Sequence	**M-1**		
Error Handling	Python code would need to be updated. User will not be able to address any error. User can spot irregularities and report them to the system admin.		

ID	FR_6.0		
Name	Tableau shows various graphs: MSF1, MSF2, and MSF3 using macroeconomic variables and machine learning algorithms		
Description	Along the bottom of the Tableau dashboard, the graphs titles for the pricing data collected for various graphs will be displayed. The sheet labeled Macroecon Forecast should display our algorithms.		
Rationale	The user will be able to view the predictions of the algorithm of their choice by selecting the corresponding tab along the bottom of the Tableau Dashboard.		
Input	Data from database tables		
Output	Front End Tableau user interface		
Forward Dependency	None		
Backward Dependency	FR_2.0		
Priority	**Medium**	**Priority Reason**	**Low Scale**
Priority Sequence	**M-2**		
Error Handling	Data table or calculations will need to be reviewed. Tableau visual could also have an issue. User can manipulate the front-end by changing visual settings or values. If error is still not resolved a developer will need to review.		

ID	FR_7.0		
Name	Display the variables in the graph calculations and the trade signal		
Description	Hovering over the different graph's (using macro-variables) market indication lines show trade date, market value on that date, and trade indicator.		
Rationale	The user can hover over any point on the close price projected by the graph and be able to understand what the calculations of the graphs are occurring as well as including the trading signal at any given point.		
Input	Data from database tables		
Output	Front End Tableau user interface		
Forward Dependency	None		
Backward Dependency	FR_2.0		
Priority	**Low**	**Priority Reason**	**Low Scale**
Priority Sequence	**L-1**		
Error Handling	Any calculation error would need to be addressed by the developers. If there is any front-end error user can seek assistance from the system administrator.		

ID	FR_8.0		
Name	Refresh data		
Description	The data that is taken from YAHOO and Quandl should be up to the latest date.		
Rationale	To get current and most accurate up-to-date information, the application must do this every day.		
Input	None		
Output	None		
Forward Dependency	Entire application.		
Backward Dependency	FR_1.0		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-5**		

3.3 Non-Functional Requirements

3.3.1 Performance

ID	NFR_1.0		
Name	Graphs & Visualizations		
Description	Although the graphs are already easier to interpret, the color must be switched from black to white for a cleaner look as per the client. The graphs must clearly show a distinction between those that are using macroeconomic variables and those that are not. The user should be able to view individual graphs and measurements. Information related to each graph i.e. legend, info box, sheets, strategy codes should be displayed clearly.		
Rationale	For the user to have full comprehension of the data being displayed it is necessary that there is nothing ambiguous on the user interface		
Input	Data from the data-sources, Tableau canvas and tabs		
Output	Graphs display		
Forward Dependency	None		
Backward Dependency	None		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-3**		

ID	NFR_2.0		
Name	Dual Graphs		
Description	The two graphs on the data source trade charts will show how the previous algorithm is progressing and the other one will show how the new algorithm with macroeconomic variables will be progressing over a 100-day period.		
Rationale	To visually represent how the two algorithms, buy and sell and to visualize how that volumetric trading is outperforming the "all in" strategy.		
Input	Data from the data-sources configured in Tableau and Quandl		
Output	Graphs with 2 or more lines and measures in the user interface of Tableau		
Forward Dependency	None		
Backward Dependency	None		
Priority	**Medium**	**Priority Reason**	**Medium Scale**
Priority Sequence	**M-1**		

3.3.2 Reliability

ID	NFR_3.0		
Name	System Reliability		
Description	Tableau reliable pulls data from our MySQL server and is able to store and export data to be grabbed without querying our server.		
Input	None		
Output	None		
Forward Dependency	None		
Backward Dependency	None		
Ideal	100%		
Success	N/A		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-1**		

3.3.3 Availability

ID	NFR_4.0		
Name	System Availability		
Description	The availability of the system when it is used		
Input	None		
Output	None		
Forward Dependency	None		
Backward Dependency	None		
Ideal	100% fully functioning		
Success	N/A		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-1**		

ID	NFR_5.0		
Name	Database connection		
Description	The computer system must have an established connection with the database for the application to store the data obtained from YAHOO and Quandl Local Host: Local machine has the database connected locally on the hard drive Server Host: The database is on the server and the machine is connected properly		
Input	Existence of database in a DBMS with required structure		
Output	Entire User Interface in Tableau		
Forward Dependency	Entire Application		
Backward Dependency	Entire Application		
Ideal	100% connectivity		
Success	N/A		
Priority	**High**	**Priority Reason**	**High Cost**
Priority Sequence	**H-2**		

ID	NFR_6.0		
Name	Internet connection		
Description	For the application to obtain the current data there must be an internet connection		
Input	None		
Output	Connection to YAHOO's financial data exchange, Quandl datasets		
Forward Dependency	Application updates, database updates with new data		
Backward Dependency	None		
Ideal	100% fully functioning		
Success	N/A		
Priority	**Medium**	**Priority Reason**	**Medium Scale**
Priority Sequence	**M-1**		

3.3.4 Security

ID	NFR_7.0		
Name	System Security		
Description	The security of the system comes from the architecture and all the individual components of the system		
Rationale	Single User: This case makes the user the main security administrator. This implies that the user will have full access to the application when fully installed. Multiple Users: The user interface will be installed on all machines to view the data, however, only a few will have administrative rights to ensure security is handled appropriately.		
Input	None		
Output	None		
Forward Dependency	Security of transformed data		
Backward Dependency	None		
Ideal	100% secure connectivity		
Success	N/A		
Priority	**Medium**	**Priority Reason**	**Medium Scale**
Priority Sequence	**M-1**		

3.3.5 Maintainability

ID	NFR_8.0		
Name	System Maintainability		
Description	Platform design will be centered around the financial instruments		
Rationale	To allow the code to add more functionality and the ability to change the algorithms if needed		
Input	None		
Output	None		
Forward Dependency	Correct algorithm calculations		
Dependency	None		
Priority	**Medium**	**Priority Reason**	**Medium Cost**
Priority Sequence	**M-1**		

ID	NFR_9.0		
Name	Code Refactoring		
Description	Refactoring the code to make it more modular		
Rationale	Although the code has been heavily refactored, there are still multiple code redundancies within the scripts. By refactoring, some of the redundancies can be eliminated and can create better structure		
Input	Existing code		
Output	Modular code		
Forward Dependency	Entire Application		
Backward Dependency	All functional requirements		
Priority	**Medium**	**Priority Reason**	**Medium Scale**
Priority Sequence	**M-2**		

3.3.6 Portability

ID	NFR_10.0		
Name	Application Portability		
Description	The application should be able to function on any Personal Computer with Windows 10/Mac OS snow leopard and all applicable web browsers that the dashboard is embedded into. With the correct versions of required software installed, the application can be reproduced without any issue.		
Rationale	These are trusted and established platforms and APIs		
Input	None		
Output	None		
Forward Dependency	Must work on all PCs and Macs when all the required software is installed		
Backward Dependency	None		
Priority	**High**	**Priority Reason**	**Large Scale**
Priority Sequence	**H-1**		

3.4 Design Constraints

- Database structure is setup per client's requirements
- The database is currently set up to only run on MySQL DBMS
- Python 3.7 or later must be used for the back-end code execution
- Dependent libraries must be installed as required before any code execution
- GitLab/Github must be used for the latest version of the code, documentation and readme
- Tableau Software and services must be used to provide a user interface
- Steady internet connection is required to update YAHOO's financial exchange data, and Quandl's datasets.
- If source trading data is no longer available, data exchange switch may be required, current exchange has no cost associated with data pulls
- Size of data can impose a constraint on the database, physical storage limits cannot be exceeded
- Deciding on buy, sell or hold criteria, human input required

3.5 Logical Database Requirements

The system must store data retrieved from YAHOO daily trader and Quandl datasets into the database in 6 float values and one date value for each exchange traded fund and macroeconomic variable. The data values include the Date, Open, High, Low, Close, Adj. Close and Volume for each trade date.

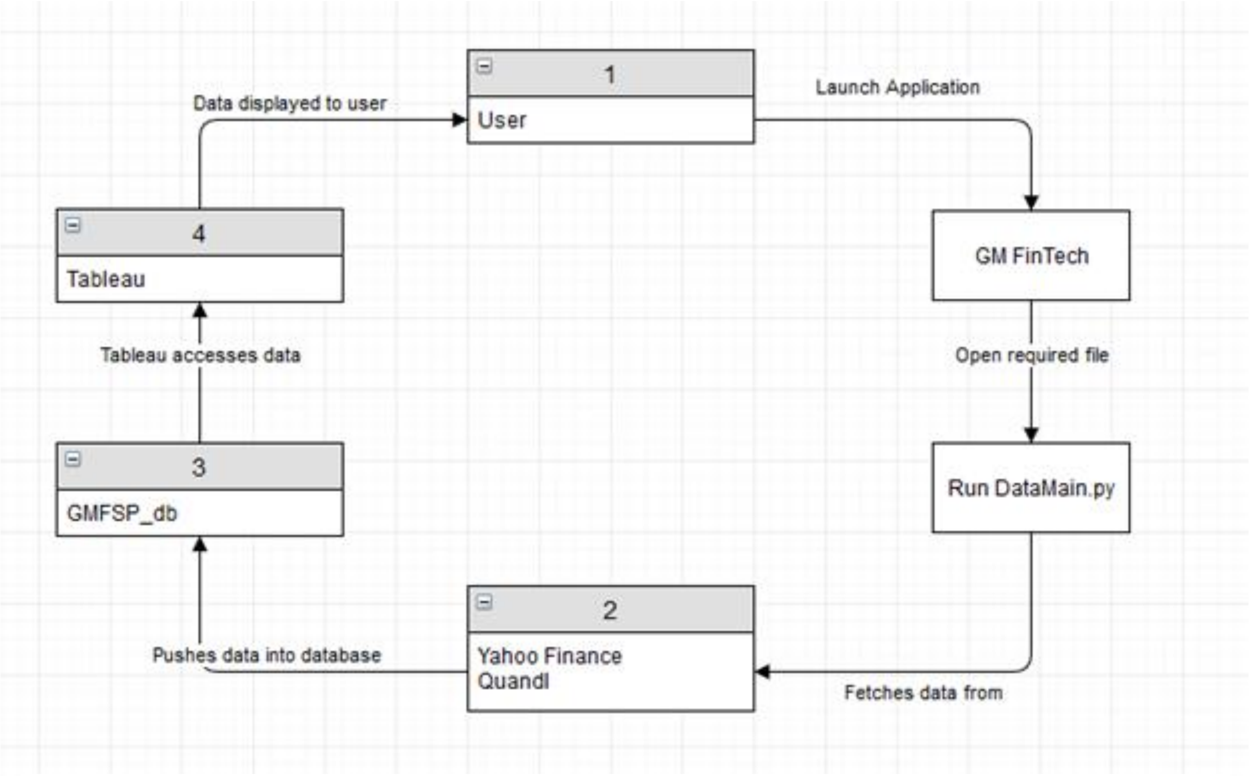
3.6 Other Requirements

This program requires knowledge of economics. Knowing how the stock market works and what impacts it in the short run and the long run is very important for this application. One must also know how indexes and ETFs work; with what stocks they are composed of. They also must know market trend lines such as a Simple Moving Average and an Exponential Moving Average.

Another requirement is knowing how to research. A simple google search will not help you out, but, research articles and journals are a must. We must delve in deep to study formulas and what economic variables have impact and how they have an impact.

4. Analysis Models

4.1 Data Flow Diagrams



Data Flow Diagram

Once the user runs our application, DataMain.py, the data is taken from YAHOO Finance and Quandl and put in the MySQL database.

It is then manipulated by Python and calculated and stored into different tables back into our database.

Tableau accesses the data by connecting to the database and using SQL queries to display the data visually into graphs.

A. Appendices

Project Proposal

A Purposeful Walk Down Walk-Street – Exploring Advanced Data Analytics in Financial Markets

WSU COURSE:

Senior Project Design and Ethics. Winter-Spring 2020

DESCRIPTION:

General Motors, a fortune 10 company, is currently one the leaders in Advanced Analytics implementations. Today, at GM, we leverage Advanced Analytics technologies to run our core Global Data, AI and Analytics Services Platform. Looking out over the next several years, the United States will need to continue to produce highly talented computer engineers and data scientists who are well versed in data analytics and data science. GM continues to partner with WSU to create real-world projects that will enable senior year students to learn these critical skills and technologies. The Use-Case chosen for this project is as follows:

Today's global markets and pricing of financial instruments are impacted week-by-week and day-by-day as a result of real-world events. In this project we will leverage fintech (financial technologies) to attempt to track and predict the positive or negative pricing movements within global financial markets. The students will design and implement key enhancements to our system that measures correlations with regards to financial instruments, trading strategies, and pricing. Ultimately, the WSU students will put their team efforts into improving our framework for descriptive and predictive analytics. They will elevate our platform to the next level and enable execution of highly accurate 'buy' and 'sell' opportunities for financial portfolios.

From GM's perspective, the design and implementation of data systems that are able to correlate and group data to measure and predict outcomes has very high value. The results and learnings from this Capstone project will be leveraged in GM IT projects in the future.

PROJECT DELIVERABLES:

1. Work with Financial Instruments and greatly enhance the accuracy of our platform.
2. Investigate and build on our existing Trading Strategies and Models. Integrate these trading strategies into a platform the gives us the ability to 'buy' and 'sell' financial instruments with the potential for maximizing return on investment.

3. Leverage the tool to create and track ‘Assets Under Management’ and prove out the accuracy and viability of the platform.
4. Leverage the front-end client and appropriate delivery mechanisms to allow the end-user to perform tasks within the system and analyze results.
5. Persistence layer migration: Leveraging open-source technologies, convert the current DB Layer into a performant, cutting-edge and world class Database implementation.

STUDENT TEAM:

Each team member may have the opportunity to work on 1-to-many of the 5 deliverables above.

TECHNOLOGIES:

- Big Data-HDFS (Hadoop) and/or Traditional RDBMS (Relational Database Management Systems)
- Open source ETL tool(s) or other extraction and loading software.
- Financial Trading Strategies, Trading algorithms, and FinTech (financial Technologies).
- Financial Trading Strategies based on organic AI/ML Models and algorithms.
- Linux, Windows and MacOS environments.
- Programming and/or scripting language – Primary/Preferred: Python. Secondary options (approval required): R, Java, MATLAB’s proprietary language.
- Front-end and user-interface will be Tableau or similar visualization tool.

GENERAL MOTORS CONTACT INFORMATION:

Joshua Feinstein

of Global Data, AI & Analytics Services

Financial Analytics Lead - General Motors

Joshua.feinstein@gm.com

1-313-319-5881

Brett Vermette

Director - Big Data Platform Infrastructure & Engineering

Wayne State University GM-IT Executive Sponsor

General Motors

Brett.Vermette@gm.com

1-313-910-6279

WAYNE STATE UNIVERSITY CONTACT INFORMATION:

Professor Seyed Mousavi

Wayne State University

Department of Computer Science

5057 Woodward Avenue, Suite 200.44, Floor 14

Phone: 313-577-0722

WSU STUDENT TEAM:

To be filled out once teams are created.