#### **Project Elaboration**

* **Domain Model Diagram**

Create a diagram that identifies conceptual classes, associations, and attributes within your project domain. The diagram must use proper UML notation for a domain model diagram.

* **Supplemental Specification**

Include a complete set of requirements for your project. Use the FURPS+ model to find any and all requirements for your project. The functional requirement list must include requirements drawn out of the use cases. All requirements must be **numbered** and have an **assigned priority** using a ranking system such as *MoSCoW*.

* **Deployment Diagram**

Provide a diagram that is representative of the hardware and software platform(s) used to deploy your solution. Please include descriptions of all required hardware and software and justification for your choices. Proper UML notation is required.

* **Use Cases** (80% complete)

80% of all known use cases must be written out in detail using the fully dressed format, as discussed in class. Update the complete list of features.

* **Work Breakdown Structure (WBS)**

A detailed list of the tasks associated with your project based upon the vision, scope, and user scenarios. The list of tasks should be as granular as possible (work package level) so that the team can clearly see everything that needs to be accomplished in order to complete the project.

* **Updated Project Schedule**

An updated schedule that includes all tasks found in the WBS with specific duration and start and completion dates for each task. Make sure to include all milestones as well as phase and iteration dates and a list of the features that will be delivered for the two Construction sprints.

//copy and pastable starter

Use Case:

Description:

Actors:

Preconditions:

Flow of Events:

Postconditions:

**YGGDRASIL**

#######################################################

Use Case: Current stock information query

Description: the system shows information on a stock that the user specifies

Actors: User

Preconditions: Software is installed and configured

Flow of Events:

Use Case starts when customer selects Check Stock Price.

1. user enters the ticker symbol that they want info on
2. user selects which fields they are interested in
3. user submits request
4. system displays “fetching data”
5. system displays data

Postconditions: System displays data

############################################################

Use Case: Add symbol to subscription list

Description: the system adds a symbol that the user specifies to the subscription list

Actors: User

Preconditions: Software is installed and configured

Flow of Events:

Use Case starts when customer selects Add Subscription.

1. User enters symbol they want to add
2. System displays subscription added

Postconditions: none

############################################################

Use Case: Remove symbol to subscription list

Description: the system removes a symbol that the user specifies from the subscription list

Actors: User

Preconditions: Software is installed and configured

Flow of Events:

Use Case starts when customer selects Remove Subscription.

1. System displays current subscription list
2. User enters symbol they want to remove from the list
3. System displays subscription removed

Postconditions: none

############################################################

Use Case: Retreive subscription list

Description: the system displays all current subscriptions

Actors: User

Preconditions: Software is installed and configured

Flow of Events:

Use Case starts when customer selects Display Subscription.

1. System displays subscription list

Postconditions: none

############################################################

Use Case: Open Plugin

Description: System opens interface for specific plugin.

Actors: User

Preconditions: Plugins are installed for the user.

Flow of Events:

Use Case starts when customer selects Plugins in the Menu.

1. System displays list of currently implemented plugins
2. User selects one of the plug ins

Postconditions: System opens plug in menu

############################################################

Use Case: Install Plugin

Description: System installs plugin specified by the user.

Actors: User

Preconditions: User has the location of a Plugin that can be utilized by the system

Flow of Events:

Use Case starts when customer selects Install Plugin in the Menu.

1. System asks the user for the location of the plug in they want to install
2. User inputs the location of the plug in file
3. System displays “Checking file”
4. System displays “verifying that that the file can be installed”
5. System installs the plug in

Postconditions: System adds the plug in to the list of plug ins

########################################################

Use Case: Select Portfolio

Description: System loads portfolio of ticker symbols

Actors: User

Preconditions: User previously saved a portfolio of ticker symbols

Flow of Events:

Use case starts when the user selects “load portfolio”

1. System displays the available portfolios
2. User selects one
3. User selects load
4. System displays ticker symbols
5. System displays “updating symbols…”
6. System displays ticker updates

Postconditions: System displays updated ticker information

#######################################################

Use Case: Save Portfolio

Description: System saves portfolio of ticker symbols

Actors: User

Preconditions: User previously has open a portfolio of ticker symbols

Flow of Events:

Use case starts when the user selects “save portfolio”

1. System prompts user for a name for the portfolio if none chosen already
2. User enters a name if needed
3. User selects save
4. System displays “saving...”

Postconditions: Portfolio is saved

#######################################################

Use Case: Open Portfolio

Description: System opens a previously portfolio of ticker symbols

Actors: Users

Preconditions: User previously saved a portfolio of ticker symbols

Flow of Events:

Use case starts when the user selects “open portfolio”

1. System displays all previously saved portfolios

2. User selects one

3. User selects open

4. System displays “opening…”

5. System displays ticker symbols in new window

7. System displays ticker updates in new window

Postconditions: Portfolio is open for viewing/editing

#######################################################

Use Case: Export Portfolio

Description: System exports portfolio to a text file

Actors: Users

Preconditions: User previously saved a portfolio of ticker symbols

Flow of Events:

Use case starts when the user selects “export portfolio”

1. System prompts user for a file name to export to

2. User enters file name

3. User selects export

4. System displays “exporting...”

Postconditions: The portfolio is exported

#######################################################

Use Case: Import Portfolio

Description: System imports a portfolio from a text file

Actors: Users

Preconditions: User has a text file in the proper format

Flow of Events:

Use case starts when the user selects “import portfolio”

1. System prompts user for file path

2. User enters file path

3. User selects import

4. System displays “importing..”

5. System displays imported portfolio

Postconditions: Portfolio is imported for viewing/editing

#######################################################

Use Case: Add Ticker to Portfolio

Description: System opens a previously portfolio of ticker symbols

Actors: Users

Preconditions: User previously saved a portfolio of ticker symbols

Flow of Events:

Use case starts when the user selects “open portfolio”

1. System displays all previously saved portfolios

2. User selects one

3. User selects open

4. System displays “opening…”

5. System displays ticker symbols in new window

7. System displays ticker updates in new window

Postconditions: Portfolio is open for viewing/editing

#######################################################

Use Case: Remove Ticker from Portfolio

Description: System removes a ticker from the portfolio

Actors: Users

Preconditions: User has open a portfolio

Flow of Events:

Use case starts when the user selects a ticker

1. User opens ticker menu

2. User selects delete

3. System displays “deleting…”

Postconditions: Ticker is deleted

**GRAPHICAL**

#######################################################

Use Case: Load Graph

Description: System outputs graphical representation of stock data.

Actors: User

Preconditions: User has Graphical Plugin installed

Flow of Events:

Use case starts when the user selects “Graph Stock”

1. System prompts name of stock to display
2. User enters specific stock ticker

Postcondition: System Displays “General graph” or “Personal graph”

#######################################################

Use Case: Load General Graph

Description: System outputs general graphical representation of stock data.

Actors: User

Preconditions:

1. User has Graphical Plugin installed
2. User has selected “General Graph”

Flow of Events:

Use case starts when the user selects “General Graph”

1. System displays graph of desired stock price over the last 8 months.

Postcondition: System displays graph of stock information.

#######################################################

Use Case: Load Personal Graph

Description: System outputs graphical representation of stock data contained within the currently opened portfolio

Actors: User

Preconditions:

1. User has Graphical Plugin installed

2. User has opened at least one portfolio

3. User has selected “Personal Graph”

Flow of Events:

Use case starts when the user selects “Personal Graph”

1. System displays graph of desired stock price since adding the stock to their portfolio

Postconditions: System displays graph of stock information.

#######################################################

Use Case: Refresh Graph

Description: System re-outputs graphical representation of stock data including recently added data

Actors: User

Preconditions:

1. User has Graphical Plugin installed

2. User has either a general or personal graph displayed

Flow of Events:

Use case starts when the user selects “Refresh”

1. System displays graph of desired stock price with extended time line up to most recently available data

Postconditions: System displays updated graph of stock information.

#######################################################

Use Case: Export Graph

Description: System saves a picture of the desired stock graph to a specified location

Actors: User

Preconditions:

1. User has Graphical Plugin installed

2. User has either a general or personal graph displayed

Flow of Events:

Use case starts when the user selects “Export”

1. System displays “Export file to location”
2. User inputs file directory
3. System displays: “Saved graph in png format in desired directory”.

Postconditions: System displays graph is saved.

################################################################

Use Case: Query database for ticker history

Description: System retreives ticker history and displays it to the user

Actors: User

Preconditions:

################################################################

Use Case: Add a new database file (import database)

Description: System loads a database file

Actors: User

Preconditions:

################################################################

Use Case: Save database file (export database)

Description: System saves a database file

Actors: User

Preconditions:

Use Case List:

Yggdrasil

[X]Current stock information query

[X]Add symbol to subscription list

[X]Remove symbol from subscription list

[X]Retreive subscription list

[X]Open plug in

[X]Install new plugin

PLUGINS

Portfolio Display Interface

[X]Select portfolio

[X]Save portfolio

[X]Open portfolio

[X]Export Portfolio

[X]Import Portfolio

[X]Add Ticker to Portfolio

[X]Delete Ticker from Portfolio

Graphical

[X]Load Graph

[X]Load General Graph

[X]Load Personal Graph

[X]Refresh Graph

[X]Export Graph

Database

[]Query database for ticker history

[]Add a new database file (import database)

[]Save database file (export database)

VWAP Calculator

http://en.wikipedia.org/wiki/Volume-weighted\_average\_price