

Board

**CS180 - Moni**



School Free



Team Visible



NC



Invite

## Backlog

**Priority - Low**

**Serverside Frontend**

**Serverside Backend**

Create ML (Predicts stocks)

**Priority - Low**

**Size - Small**

Create UML for Server (Julian)



Decide icon for application (Julian)



**Priority - Medium**

**Size - Large**

**Serverside Backend**

Store Updated Stocks into CSV ()

**Priority - Low**

**Size - Large**

**Serverside Backend**

Train ML Model to predict next stock

## Sprint Backlog

Look into Chart.js or other library that can do "Candlestick"charts

## In Progress

**Serverside Frontend**

**Size - Medium**

**Clientside Frontend**

Make Data, Analytics and News  
Picker which display different screens  
(Julian)

  1



Make 'Choose Asset Type' Button  
functional (Julian, Patrick)



## Testing

**Size - Medium**   **Priority - High**

**Clientside Backend**

Create functions to filter stock data  
by date range (Nick, Patrick)

 1

NC



## Sprint 4 - Completed

## Sprint 3 - Completed

**Size - Medium**   **Priority - High**

**Serverside Backend**

Send side bar labels in chunks  
(Mohamed, Julian)

 1 



**Priority - Medium    Size - Small**

**Clientside Frontend**

Change color of graph depending on  
previousDayClosed value (Julian)

👁️ 🔗 2



**Size - Medium    Priority - High**

**Serverside Backend**

Pull new data from the web and  
append to csv (Julian)

👁️



**Size - Small    Priority - High**

**Serverside Backend**

Make try catch block to prevent  
server from crashing (Julian)

👁️



**Size - Medium**

**Clientside Frontend**

Make Column chart in bottom of  
graph for volume (Julian)

👁️ 🔗 1



**Size - Small    Clientside Backend**

**Clientside Frontend**

Create slider to choose between  
stocks and etfs in HTML/CSS  
(Patrick)



**Priority - Medium    Size - Small**

**Clientside Frontend**

Make Sidebar percent change in colored box on previousDayClosed (Hayden)

 1



**Size - Medium    Priority - High**

**Clientside Frontend**

Create Button to choose stocks/etfs to view (Hayden)

 1



Updated UI to look like the mock up (Julian)





**Priority - Medium    Size - Small**

**Serverside Backend**

Added Delete functionality to Server (Julian)





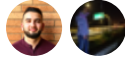
## Sprint 2 - Completed

---

**Priority - High**

**Serverside Backend**

Added GetLabels Function (Julian, Mohamed)



**Priority - Medium    Size - Medium**

**Serverside Backend**

Multi Thread the import of files (Julian)



**Size - Medium    Priority - High**

**Serverside Backend**

Server fetches name of stock before sending data to client (Julian)



**Size - Medium    Priority - High**

**Clientside Backend**

Load Pubnub JSON Request into Stock Object (Hayden, Julian, Mohamed, Nicholas)



NC

**Size - Medium    Priority - High**

**Clientside Backend**

Sort stocks by date(Hayden, Mohammed)



**Size - Medium    Priority - High**

**Clientside Frontend**

Create chart to view stock data (Nicholas)

NC

**Size - Small    Priority - High**

**Clientside Frontend**

Add previousDayClosed value to stock side bar (Nicholas)

NC

Priority - Medium    Size - Medium

**Clientside Frontend**

Create date range filters buttons for graph (Nicholas)

NC

## Sprint 1- Completed

Design Frontend Layout (Julian)

  2



Unit Test    Serverside Backend

Make Unit Tests for Database Class (Julian)





Unit Test    Serverside Backend

Make Unit Tests for Stock Class (Julian)

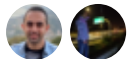




Size - Small    Priority - High

**Clientside Backend**

Create Stock class(Hayden, Mohammed)



Priority - Medium    Size - Small

**Clientside Backend**

Stock dummy generator(Hayden,Mohammed)



Import Stock/ETF CSVs as Dict  
(Mohamed, Julian, Patrick)



Push initial Electron project to GitHub  
(Mohamed)



**Size - Small    Priority - High**  
**Serverside Backend**

Design JSON request architecture  
(Julian)



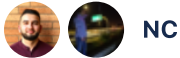
**Priority - High**  
**Clientside Backend**

Request PubNub data (Julian,  
Mohamed, Nick)



**Priority - High**  
**Clientside Backend**

Receive PubNub data (Julian,  
Mohamed, Nick)



Pubnub Communication Protocol  
(Julian)



**Size - Small    Priority - High**  
**Serverside Backend**

Set up PubNub (Julian)



**Clientside Backend**

Create Stock Object (Julian, Patrick)

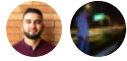


**Priority - High**

**Serverside Frontend**

**Serverside Backend**

Accept request from PubNub and  
return requested Data to PubNub  
(Julian, Mohamed)



**Sprint 5 - Completed**

**Sprint 6 - Completed**

**Sprint 7 - Completed**

**Sprint 8 - Completed**

**Sprint 9 - Completed**

**Sprint 10 - Completed**