**11/11 Monday**

*4:21pm*

Received the API project

*6:17pm*

Research about options, API and azure

Resource:

What is an API ? <https://www.youtube.com/watch?v=s7wmiS2mSXY>

What is a REST API? <https://www.youtube.com/watch?v=7YcW25PHnAA>

What is azure? Cloud service (probably use for deploying [use AWS?Azure])

JS/HTML use to integrate API to web application

*9:00pm*

Do Scrum artifacts on Trello <https://trello.com/b/SP1wNgGq/vetaapi>

**11/12 Tuesday**

*7:22am*

Research about vanilla call option

Resource: <https://www.investopedia.com/terms/v/vanillaoption.asp>

*4:14pm*

Discussion with Trey about the information needed to price ATM(AT THE MONEY) call options

* Black Scholes and Heston give real value, but we are looking for estimation
* Research about Black Scholes

*7:09pm*

Decided that the formula for the estimation of vanilla call options is

Price = ( 0.4 \* Volatility \* sqrt(Time Ratio) \* Base Price)

*7:13pm*

The price of the ATM option (call and puyt) has 3 month until expiration. Volatility is 23%. Stock Price is $45.

Answer: 0.4 \* 0.23 \* sqrt(3/12) \* 45 = 2.07

*9:28pm*

Learn more about ATM,OTM and ITM.

Resource:

<https://www.youtube.com/watch?v=ipf0Yg2Z4Gs&t=479s>

<https://www.youtube.com/watch?v=MiybniIIvx0>

**11/15 Friday**

*1:50am*

Finish setting up environment and git commit (First commit)

2:34am

Found out about CORS (Cross Origin Requests) that are common in APIs. Errors appears when JS is not on the same server.

Example: API server is hosted by HeroKU and frontend code for the API is at the nullified. Will produce error because JS cannot make request to API server without this header,CORS.

3:13am

Decided to erase all the code regarding code factory because it is hard to implement code factory with PyMongo(MongoDB)

3:55am

Learned that Postman will be needed to test Rest API

12:02pm

Make a SECRET\_KEY and store it at the .env and gitignore the .env file

12:27pm

Make the get route and check with mongodb sample dataset

1:13pm

Checking the get route is still not successfully but found possible ERROR: no modules found for dnspython to use mongo+srv

1:19pm

Still Error, possible error: retryWrites is currently true . Probably change to False

1:30pm

Possible Error:

.env file

MONGO\_URI=mongodb+srv://karenws:ravekun@[cluster0-9whhr.mongodb.net/test?retryWrites=true&w=majority](http://cluster0-9whhr.mongodb.net/test?retryWrites=true&w=majority)

1:33pm

TEST: Get route with postman success.

Error is 1:30pm 11/15 Friday

So they are targeting a databasename Test but database name is vanilla

1:53pm

Finish writing post route and testing post route with POSTMAN

1:58pm

Error found: JSON file naming

2:00pm

Post route is working and pushed to github

6:23pm

Research about How to connect Flask API to React js

**11/17 Sunday**

2:12pm

Found out missing section (can be popup or alert) for the output

6:30pm

Finally finish the output section (took a long time because of pushing the wrong JSON object to the vanilla array)

6:40pm

Tested the code, refactoring some code to make it prettier, send the pictures to Trey

6:42pm

Still researching about how to upload Flask and React JS to AWS

7:30pm

Editing the environmental environment to make sure eb --version is working. Still did not work

7:38pm

Use virtual machine to download awsebcli . Failed at first because Eb version was python 2. Solved this with pip3 instead of pip.

8:09pm

Used AWS EB gui to upload code, ERROR

8:17pm

Solved the problelm by pip3 freeze the requirements, another error WGSPI refer to a file that does not exists

11:04pm

Start searching to deploy flask API to heroku

11:13pm

Realise mistake for flask API

11:28pm

Successfully deploy the flask API with AWS EB

11:35pm

Successfully deploy frontend React with S3

11:45pm

Fetch link error, and tests are all success

11:59pm

Create a README.md

**11/18 Monday**

12:10am

Finish documentation

Future TODO:

* Make sure that form inputs are numbers
* Make sure that form inputs have value
* Have a progress bar
* Fetch data from yahoofinance to make better API