|  |
| --- |
| Software Engineering CSC 4350 |
| Tickr |
| Chalie Neopane, Austin Tyler, Justin Little, Haroon Qahtan, Natsai Ndebele |

**Section 1:**

**Team**

**Section 1:**

Group Name: Tickr

|  |  |
| --- | --- |
| **Team Members** | **Role** |
| Haroon Qahtan | Back End developer / Project Designer |
| Natsai Ndebele | Team Coordinator /Front End developer /Technical Documenter |
| Justin Little | Front End developer / Project Designer |
| Charlie Neopane | Back End developer / Technical Documenter |
| Austin Tyler | Full Stack developer |

Roles Definitions:

-Back End Developer: in charge of working on the proposed solution’s Back-end

-Front End Developer: in charge of working on the proposed solution’s Front-end

-Project Designer: designs the system and how data will flow through the system

-Technical Documenter: in charge of making the proposed solution’s technical documentation.

-Full Stack Developer: in charge of working on the system’s front-end and back-end

**Section 2:**

**Team Member Resumes**

**HAROON QAHTAN**

404-988-4543 | <https://linkedin.com/in/HaroonQahtan>| [haroonqahtan@gmail.com](mailto:haroonqahtan@gmail.com)

<https://github.com/eychcue>

**EDUCATION**

**Georgia State University –College of Arts & Sciences -** Atlanta, Georgia **May 2019**

*Bachelor of Science: Computer Science*

**SKILLS**

**Languages:** Java**,** Python, Bash, C, JavaScript, HTML/CSS, Swift, Assembly x86, Lua

**Technologies:** Android, Bootstrap, Flask, AWS, GIT

**Other skills:** Fluent in Arabic, Analytical thinker, Superior verbal and written communication skills

**WORK EXPERIENCE**

**Georgia State University,** Atlanta, Georgia **April 2017 – Present**

Workstation Technician

* Provide technical assistance and support related to computer systems, hardware, and software.
* Respond to tickets from numerous Georgia State University departments which require me to run diagnostic programs, isolate problems, and determine the best implementation for a solution.

**TECHNICAL EXPERIENCE/AWARDS**

**Won First Place at MartaHackaton Fall 2017**

* Contributed in building an IOS app with my team called MartaGoNoGo that advises Marta riders when to ride the bus with the least delays.
* Developed a Machine Learning Model in Python using the Scikit Learn library that took in information about weather, traffic, and historical bus arrivals to provide an accurate prediction of whether today will be a good day to ride Marta bus to get to the destination on time.

**Won Overall Award Anthem Prize at hackGSU Spring 2017**

* Received the best overall Anthem prize for our project Vizitr that solves the problem of allowing people to visit companies without the hassle of being time-consuming.
* Worked on the Alexa Echo app that communicated with the visitor to get their information and pass it on to a WebApp that handled the users’ information as well as keeping a record of their visit.

**PROJECTS**

**StockBot**

* StockBot is a project started at a hackathon (HackEmory) which I decided to continue to work on. It is an Amazon Alexa application that allows users to get more detailed information about stocks.
* Built the application using Python, Amazon Web Services, and the Yahoo Finance API.

**Junya -** App Game Development

*Founder*

* Developed and published two casual mobile games using Corona SDK and Buildbox to the Play Store.
* Designed special user interface for apps using design software such as Photoshop and Illustrator.
* Focused on revenue generation by managing all aspects of our income using Google Analytics and AdMob.

**Buildr –** Home Depot Code-athlon

* Buildr is a web app that helps users plan for a DIY project by walking them through the tools needed for the project and an easy portal to calculate and buy all the items from the Home Depot website.
* Built the backend API using Python which scrapped the Home Depot website for products, prices and exact locations of where items are located.

**LEADERSHIP AND ACTIVITIES**

**PantherHackers,** Georgia State University **August 2016 – Present**

*Finance Committee Chair*

* Make financial decisions on how to plan PantherHackers’ budget and best way to spend it
* Seek and maintain sponsorships from companies for PantherHackers
* Work with other committee’s in the club to manage and maintain their committee budget

Natsai Ndebele

**Computer Science Education:**

* Georgia State University, Bachelor of Science in Computer Science, Concentration Graphics and Human Computer Interaction
* Classification: Junior

**Useful Courses for this project:**

* Windowing systems (a course on graphical user interfaces)

**Relevant Projects:**

**ExeBeatz**, *Team Lead and Developer*

* Created and organized Technical Documentation
* Designed and created a Java based app for users to exercise listening to music
* Designed and created quality assurance checks to ensure the prototype matched the designs specifications and requirements

**Bethel Campus Fellowship Web Team:**

* Currently working with a web team based in the US to build a conference app for Bethel Campus Fellowship’s annual conferences.

**Machabeng Gate Registration System:**

* Designed and created a Gate Registration System for Machabeng College International School of Lesotho.
* Created and organized Technical Documentation

**Work Experience:**

* Back Stage Capital Database Manager
* Activvely, Social Media and Marketing intern
* iD Tech, Mobile App Development and Video Production for Youtube Instructor

**Proficient in:**

* Java
* Html
* CSS
* SQL
* Java FX

**Knowledge of:**

* PHP
* Python (not too proficient)
* Javascript
* Git Hub

Justin Little:

**Education - Georgia State University**

Bachelors of Science - Computer Science

Concentration in Software Systems

Junior - Expected Graduation May 2019

**Relevant Coursework**

* Principles of Computer Science I & II
* Discrete Mathematics
* Computer Organization
* System-Level Programming
* Data Structures
* Computer Architecture

**Current Coursework**

* Design & Analysis of Algorithms
* Web Programming
* Software Engineering

**Career Experience**

* Apple - iOS Technical Support Advisor
* Freelance Web Development

**Proficient In**

* Java
* C
* JavaScript
* HTML
* CSS

**Working Knowledge In**

* Python
* PHP
* SQL
* Git & Github
* APIs

**Tools Used:**

* IntelliJ IDEA
* Sublime Text
* Terminal
* FileZilla

Charlie Neopane:

**Computer Science Education:**

**Associate of Science, Computer Science** ->Georgia Perimeter College Graduated May 2016

Relevant Project:

Temperature Conversion from Fahrenheit to Celsius or vice-versa using GUI in Java programming.

**Bachelor of Science, Computer Science**

Relevant Projects:

**Individual Projects:**

* Weight Conversion GUI (2016): If user input his/her weight in pounds then it will be converted to grams or vice-versa.
* Bank Account Transaction with deposit, interest rate and withdrawn functions in Java Program.

**Group Projects:**

* Build a website with PHP and CSS. Web sites is based on the mad libs games.
* Build a website called E-Commerce with PHP –MVC. It was based on the booking flight, prepaid parking and renting the cars.
* Develop an android app called FormAgent where a person working for certain company can write what is the problem with customer and they can send those notes to theirs headquarter.

**Technology Skills**

Programming Languages (2016): Java, C++, Python, PHP, HTML, CSS, Android App Developer (Beginner)

Austin Tyler:

**Education - Georgia State University**

Bachelors of Computer Science

Junior - Expected graduation Spring 2019

**Career experience**

Office Depot Customer Tech Support

**Proficient In:**

Java

**Working Knowledge**

C

Python

**Topic:**

Business/Finance

As a group for our Software Engineering Project, we would like to work on a financial solution, which tracks user finances by monitoring spending habits and helps the user make monthly cryptocurrency investments by monitors cryptocurrency markets.

**Section 3:**

**Planning and Scheduling:**

**Section 3.1**

**Challenges and Risks :**

·         What is the single most serious challenge you see in developing the product on schedule?

o    One of the biggest challenges we see for this deliverable is having to create the various diagrams and components. None of us have that much experience with the parts we have been assigned and there is an elevated risk of getting it wrong. To minimize this we will talk to the professor for clarification and use available online sources for guidance.

·         What are 2+ risks you can foresee in completing this project?

o    Properly understanding the problem statement will play a huge role on the completion of this part of the project. If we do not completely understand the problem statement we may not be able to correctly complete the following models. To avoid this we will do extensive research on our problem and have several meetings to ensure that we are on the same page.

o    One risk we see for this is the failure to complete the schedule on time which will affect the submission of the app on time. To avoid this we will use monitoring tools such as Git-hub and Slack to ensure that everyone is on track.

**Tasks**

* Identify principal tasks
* Identify risks & plan for them
* divide tasks among group and make schedule
* Create a GitHub & Slack and invite everyone to them
* Create a Problem Statement
* Identify the System Requirements & model them in UML
* Develop use cases for the project & list their requirements
* Create a video detailing the problem statement for the project
* Create Video

**Schedule:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Task** | **Effort (person-days)** | **Duration (hours/or days/ etc.)** | | **Dependencies** | **Allocated to** | **Monitoring** |
| **Min** | **Max** |
| Outline Plan   * Hardware Software Requirements * Challenge and Risks | 2 days | 2 days | 3 days |  | Austin Tyler | Git Hub |
| Scheduling | 1 day | 1 day | 2 days |  | Natsai Ndebele | Git Hub |
| Version Control set-up | 1 day | 30 min | 1 hour |  | Justin Little | Git Hub |
| Update Git-hub Project Tracker | 1 day | 30 min | 1 hour |  | Justin Little | Git Hub |
| Problem Statement | 1 day | 1 day | 2 days |  | Haroon Qahtan | Git Hub |
| System requirements   * Read chapter 5 * UML Activities diagram | 2 days | 2 days | 3 days |  | Charlie Neopane |  |
| Use case diagrams   * Read chapter 4 | 2 days | 2 days | 3 days | System Requirements | Justin Little |  |
| Virtual Check-in | 30 min | 30 min | 1 hour | All tasks above | All | Slack |
| Video Script/ Plan | 30 min | 30 min | 1 hour | Problem Statement | Natsai Ndebele |  |
| Video Recording | 30 min | 30 min | 1 hour | Video Script | All |  |
| Video Upload | 30 min | 30 min | 1 hour | Video Recording | Natsai Ndebele |  |

**Method to calculate effort used: Experience Method**

Most of the tasks were given up to a day, because we had to take time to read and understand how to correctly do the tasks as we had 0 up to a little experience with some of the tasks we had to do.

Factors that were also taken into consideration were the time it would take to attend to some tasks, because of school interfering. Time taken to clarify issues by asking the team lead who would then speak to the professor.

**Monitoring and Reporting Mechanisms:**

To monitor our progress, we are using Git Hub progress tracker Kanban boards. With these what we will do is when a task is in progress its on the progress board, and when complete its on the complete. Another method to monitor and report to each other is slack, which we will also use to communicate any progress or issues experienced.

Before recording the video, we will also have a team meeting for all members to give an update on all their various tasks and what needed to be done.

**Section 3.2**

**Challenges and Risks:**

* What is the single most serious challenge you see in developing the product on schedule?

One of the biggest challenges we for see for developing this product is the fact that we do not have as much experience with Class Diagrams and Behavioral Modelling. This may lead to a delay in the development of the diagrams as a lot of research and reading on the diagrams will have to be done and this may take some time. In addition, there is a risk in getting the diagrams not entirely correct. To avoid this, we will communicate with the professor and use the resources provided to us.

* What are 2+ risks you can foresee in completing this project?

This project relies on various APIs to get info on the user’s finances, stock prices, and crypto-currency rates. One significant risk we see at this point of the project is correctly representing the various API’s in the Class Diagrams and their interactions in the Behavioral Modelling.

In addition, another risk we foresee is with our Behavioral Modelling diagrams, we have had issues with out Use Case Diagrams which the Behavioral Modelling relies on. This poses the risk of getting the Behavioral modelling wrong. To minimize this risk we will consult the professor to ensure that the diagrams are correct to avoid the domino effect of errors.

* Ways to minimize risks

Ensuring that tasks are started as soon as possible. This will help the group monitor if tasks are being done correctly. It also gives us a safety period to get feedback from the professor and correct any changes that need to be made. The Project Manager will also monitor the group’s progress via slack and git hub to ensure that all tasks are being completed correctly.

**Tasks:**

* Identify principal tasks
* Identify risks & plan for them divide tasks among group and make schedule
* Revise and Complete Requirement Doc
* Revise Use Case Diagrams
* Work on Class Diagrams
* Work on Behavioral Modelling
* Plane video Script
* Record video

**Schedule:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Task** | **Effort (person-days)** | **Duration (hours/or days/ etc.)** | | **Dependencies** | **Allocated to** | **Monitoring** |
| **Min** | **Max** |
| Outline Plan   * Challenge and Risks * How to minimize challenges | 1 day | 1 day | 1 day |  | Natsai Ndebele | Git Hub |
| Scheduling | 1 day | 1 day | 1 day |  | Natsai Ndebele | Git Hub |
| * Revise and Complete Requirement Doc | 2 days | 1 day | 2 days |  | Charlie Neopane | Git Hub |
| * Revise Use Case Diagrams | 1 day | 1 day | 2 days |  | Justin Little | Git Hub |
| Class Diagrams Part 1:   * Identify Objects and associations between them | 2 days | 1 day | 2 days | Revised Use case diagrams | Justin Little | Slack |
| Class Diagrams Part 2:   * Identify multiplicity and attributes of objects | 1 day | 1 day | 1 day | Class Diagrams Part 1 | Haroon Qahtan | Slack |
| Class Diagrams Part 3 | 2 days | 2 days | 2 days | Class Diagrams Part 2 | Haroon Qahtan | Slack |
| Behavioral Modelling | 3 days | 2 days | 3 days |  | Austin Tyler | Git Hub |
| Plan video script | 30 min | 30 min | 1 hour | All of the above | Natsai Ndebele | Slack |
| Record video | 1 hour | 30 min | 1 hour | Video Script | All |  |

**Method to calculate effort used: Experience Method**

This time around, a lot of revision had to be done to previously completed tasks. The feed back that we had helped give us more confidence and somewhat experience with this task. The only tasks we really were unsure of and had little to zero experience with were the Class Diagrams and Behavioral Models. So a lot more time was taken reading and studying on those.

Factors that were also taken into consideration were the time it would take to attend to some tasks, because of school interfering. Time taken to clarify issues by asking the team lead who would then speak to the professor.

**Monitoring and Reporting Mechanisms:**

To monitor our progress, we are using Git Hub progress tracker Kanban boards. With these what we will do is when a task is in progress its on the progress board, and when complete its on the complete. Another method to monitor and report to each other is slack, which we will also use to communicate any progress or issues experienced.

Before recording the video, we will also have a team meeting for all members to give an update on all their various tasks and what needed to be done.

**Section 3.3**

**Challenges and Risks:**

* What is the single most serious challenge you see in developing the product on schedule?

When going over our class diagrams we came to the realization that one of the API’s on our diagrams has been discontinued by Google, so we need to change the API to another one which operates differently and may affect the structure of our class diagrams, design pattern and use case diagrams. The challenge lies in identifying how it affects our diagrams. Once we have an idea of how we will then revise the diagrams and then seek advice from the professor to ensure that our diagrams are correct.

* What are 2+ risks you can foresee in completing this project?

The first part of the project to be implemented relies on linking an API to our app to get the stocks. Since we need to switch our API, one risk we foresee concerns the new API and whether or not we will be able to correctly parse the data into our database.

One other risk we for see for developing this product is the fact that we still unsure as a group on what to do for the Design Pattern and Implementing the solution. We are not entirely sure on how to produce an easy to implement product. This will take a while and a lot of research, and checking with the professor.

* Ways to minimize risks

To avoid a major time delay, the Project Manager will have several both physical and virtual (via phone and Google Hangouts) meetings with the various team members to let them know what the need to do for this part of the project, and give them feedback from the meeting. In addition, the Git-hub Kanban board will have all the necessary tasks uploaded for all the team members before the meeting. All documents containing feedback from the professor will be uploaded to the slack channel as soon as possible for a reference. Ensuring that tasks are started as soon as possible. This will help the group monitor if tasks are being done correctly. It also gives us a safety period to get feedback from the professor and correct any changes that need to be made. The Project Manager will also monitor the group’s progress via slack and git hub to ensure that all tasks are being completed correctly. A virtual Google Hangouts meeting will be held to track progress during the weekend to ensure that everyone knows what they are doing. For this tasks we will increase the frequency of meetings to monitor progress.

**Tasks:**

* Revise Use Case Diagram
* Revise Class Diagrams
* Revise Test Cases
* Revise Sequence Diagrams
* Revise other parts of Documents 1, 2, 3
* Create Design Pattern
* Implement the System Design

**Schedule:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Task** | **Effort (person-days)** | **Duration (hours/or days/ etc.)** | | **Dependencies** | **Allocated to** | **Monitoring** |
| **Min** | **Max** |
| Outline Plan   * Challenge and Risks * How to minimize challenges | 1 day | 1 day | 1 day |  | Natsai Ndebele | Git Hub |
| Scheduling | 1 day | 1 day | 1 day |  | Natsai Ndebele | Git Hub |
| * Revise Use Case Diagram | 2 days | 1 day | 2 days |  | Justin Little | Slack |
| * Revise Class Diagrams Add   data type, operations return types, of relationships in terms of: association, aggregation, generalization, multiplicity | 1 day | 1 day | 2 days |  | Haroon | Git Hub |
| Revise Test Cases   * Add details and steps to test cases | 2 days | 1 day | 2 days |  | Natsai Ndebele | Git Hub |
| Refine sequence diagrams. | 1 day | 1 day | 1 day | Use case diagrams | Austin Tyler | Slack |
| Create Design Pattern | 2 days | 2 days | 2 days | Class Diagrams | Haroon Qahtan | Slack |
| Implement the System Design | 8 days | 6 days | 9days |  | Charlie Neopane, Haroon Qahtan | Slack |
| * Revise other parts of Documents 1, 2, 3 | 3 days | 1 day | 3 days |  | Natsai Ndebele | Github |
| Plan video script | 30 min | 30 min | 1 hour | All of the above | Natsai Ndebele | Slack |
| Record video | 1 hour | 30 min | 1 hour | Video Script | All |  |
| Edit video | 2 hours | 1.5 hours | 2 hours | All of the above | Natsai Ndebele | Slack |

**Method to calculate effort used: Experience Method**

This time around, a lot of revision had to be done to previously completed tasks. The feed back that we had helped give us more confidence and somewhat experience with this task. The only tasks we really were unsure of is the implementation part which will take us a while because we must do research on a replacement for the API we wanted. So a lot more time was taken doing the research.

Factors that were also taken into consideration were the time it would take to attend to some tasks, because of school interfering. Time taken to clarify issues by asking the team lead who would then speak to the professor.

**Monitoring and Reporting Mechanisms:**

To monitor our progress, we are using Git Hub progress tracker Kanban boards. With these what we will do is when a task is in progress its on the progress board, and when complete its on the complete. Another method to monitor and report to each other is slack, which we will also use to communicate any progress or issues experienced.

Before recording the video, we will also have a team meeting for all members to give an update on all their various tasks and what needed to be done.

Section 4

Problem Statement

**Problem Statement:**

What is your product, on a high level?

* Our product is an app that allows users to track their finances. Whether that’s bank balance, stock portfolios and even cryptocurrency portfolios. The users can create an account, link their bank, and then begin to add multiple investments that they have made. Therefore, our application will keep track of all those investments and then give an overall worth of the portfolio based on those multiple investments.

Whom is it for?

* Our product is for anyone who needs a way to keep track of their finances and investments. In this context, investments being stocks and cryptocurrencies.

What problem does it solve?

* Keeping up with all your finances might be time consuming especially if you invest in a personal portfolio. Therefore, our app solves the problem of keeping track of the finances happening in your bank as well as the investments that the user makes on either stocks or cryptocurrencies.

What alternatives are available?

* There are multiple applications out there that track personal finances such as **Personal Capital Finance, SigFig Wealth Management, Ticker: Stocks Portfolio Manager, and Mint.**

Why is this project compelling and worth developing?

* Our project does what the other apps do, and that is to combine multiple assets into one application. The apps mentioned above focus on one type of tracking. Either it is stocks or personal bank finance. What we are trying to do with Tickr is to create a platform where the average user can include multiple types of finances that they own.
* This project is worth developing because it will need the use of multiple API’s and we believe that learning to work with API’s is very crucial when working in the real world. Therefore, it is a good way for us to practice and learn new methods.

Describe the top-level objectives, differentiators, target customers, and scope of your product.

* Top level objectives is to research more on how to retrieve financial information from our users. Also create a flow chart and basic design to see how our app is going to look like and how it will users  be able to interact with it.
* Our main differentiator is that a user can add multiple sources of finance activity. This is very unique to our app since there isn’t an application in the market that allows users to include bank activity, stock investments as well as cryptocurrency investments. Also our application will be on mobile since we believe it will reach a greater audience based on the rise of mobile use as opposed to web clients.
* Our target customers are people who have a bank account and who also invest or want to invest in either stocks or cryptocurrencies to create a portfolio. Our target audience also needs a way to keep track of all movements in their portfolios as well as keep track of their overall worth.
* As of right now, the scope of our product is targeting multiple sources of income/wealth. As well as creating the application on a mobile device that being on the Android platform.

What are the competitors and what is novel in your approach?

* As mentioned early on, some of the competitors out there are **Personal Capital Finance, SigFig Wealth Management, Ticker: Stocks Portfolio Manager, and Mint.**
* However, our approach to this application is novel because we are not only focusing on one source of income/wealth. Whereas the other applications are focused on either bank tracking or portfolio tracking. Our goal is to combine the two to give our users an overall value of their wealth.

Make it clear that the system can be built, making good use of the available resources and

technology.

* The system will be built on the Android platform, and we already have some team members who are familiar with this technology.
* Also, we understand what goes into investing in stocks and cryptocurrencies, therefore, we are working with a familiar domain.
* In order to retrieve data to input into our application, we have already researched some API’s that allow for stocks and cryptocurrency price retrieval such as Google Finance API and Coinbase API. Also we have been researching to find a suitable bank API and we believe that Plaid’s API works best with our application in order to retrieve bank info.

What is interesting about this project from a technical point of view?

* The heavy use of API’s is very interesting in this project. Learning to work with them and implanting them is really game changing. The API’s we are working with are all finance based however one is just for banking while the others are to retrieve stock and cryptocurrency information in real time.
* Also, this project will be developed on a mobile platform (Android) which means that it will be available just a few taps away.

Section 5

System Requirements

**System Requirement Description**

**Brief Description of Tickr:** The system which helps to keep track of individual’s finances like bank balance, stock portfolios along with cryptocurrency portfolios.

**User inputs:** Sign up/Login credentials, Option to select Bank, Coinbase, Stock Market.

**Requirement Description:** The user will be able to keep track of all the transaction of bank, and Coinbase along with monitor the stock market of selected stock.

>>  User will Sign up/Login in to Tickr

>>  User will have three options

**I. Bank**

·      Uses Plaid API to declare Bank name

·      Using the credentials, verify the information and Tickr will pull bank account transaction

**II. Coinbase**

·      User inputs login information for Coinbase

·      Tickr will pull the report of cryptocurrency balances

**III. Stock Market**

·      User will pick one or more stocks

·      After seeing the user selection, Tickr will use the Google Finance API to show the current rates for those selected stocks.

**Outputs:** Full report of spending graph, Coinbase balances, stock prices, current cryptocurrency rates.

**Test Cases:**

1. **Verify whether user logged in with Google Account.**

**Description:** Used to verify if the user is logged into the system using their Google Account

**Test Inputs:** Google Account username and password

**Expected:** A successful login output to inform the user that they have successfully logged in using their Google account

**Results:** Google e-mail or name displayed in the app for user details, and successful login to the app home page

**Dependencies:** none

**Initialization:** The user is on the log-in page of the system

**Test Steps:**

1. Ensure that you are on app’s login page
2. Click the login with Google button
3. Enter Google e-mail and password or if already logged into google on device, allow app to access Google account
4. Press login via Google button
5. **When user selects the Bank, verify the login credentials for the bank**

**Description:** When the user has successfully logged into our app in order to pull their personal financial data they will need to login to their bank so that we can retrieve their personal financial data. The details they provide to login will need to be verified

**Test Inputs:** Bank username and password

**Expected:** A successful login into bank which will retrieve the user’s bank details

**Results:** Username displayed on app screen and their bank balance displayed

**Dependencies:** Depends on the user successfully logging into their account

**Initialization:** Click the Bank login button

**Test Steps:**

1. Select Bank
2. Enter username and password
3. If details are correct the user is logged into their bank and their bank details are retrieved.
4. If details are incorrect an error will be outputted informing the user that their login details are incorrect.
5. **When user selects Stocks, they can select either Coinbase and verify the login credentials and allow to login if requirement is met.**

**Description:** Once logged into the app the user can Click the Stock button and then choose to get their cryptocurrency information from Coinbase

**Test Inputs:** correct username and password

**Expected:** successful authentication and Coinbase cryptocurrency values returned

**Results:** Coinbase username displayed on app screen and their bank balance displayed

**Dependencies:** Successful login into their Coinbase

**Initialization:** Click the login to Coinbase account

**Test Steps:**

1. Click the Coinbase login button
2. Enter username and password
3. If username and password are not correct an error message prompting the user to enter their correct credentials will be outputted
4. If the username and password are correct the user’s cryptocurrency information from Coinbase will be outputted.
5. **Verify whether the Coinbase is showing the current rates and balances of cryptocurrency**

**Description:** To verify whether or not the Coinbase API is showing the current value of the stock pulled from the API, we will have the tester act as the system admin and compare the values of the pulled stock to the current value of the market

**Test Inputs:** Value of stock from API compared to value of stock from market

**Expected:** Value pulled from the API and displayed in the app should match

**Results:** Matching values of the stocks

**Dependencies:** Depends on user being logged into their Coinbase account.

**Initialization:** Log into the user Coinbase account

**Test Steps:**

1. Once successfully logged into Coinbase
2. Check that the value displayed for the user’s cryptocurrency stock values matches the market value.
3. **Verify the price of stock by comparing with known stock tracker.**

**Description:** To ensure that the correct value of the stocks is being pulled by the API into our app, we will verify this by comparing it to the current value of the stock from a known stock trader

**Test Inputs:** Value of the Stock in the app pulled from the API

**Expected:** Values of Stocks pulled from the API should match the current market value

**Results:** Matching prices of Stocks in app and from trusted stock trader

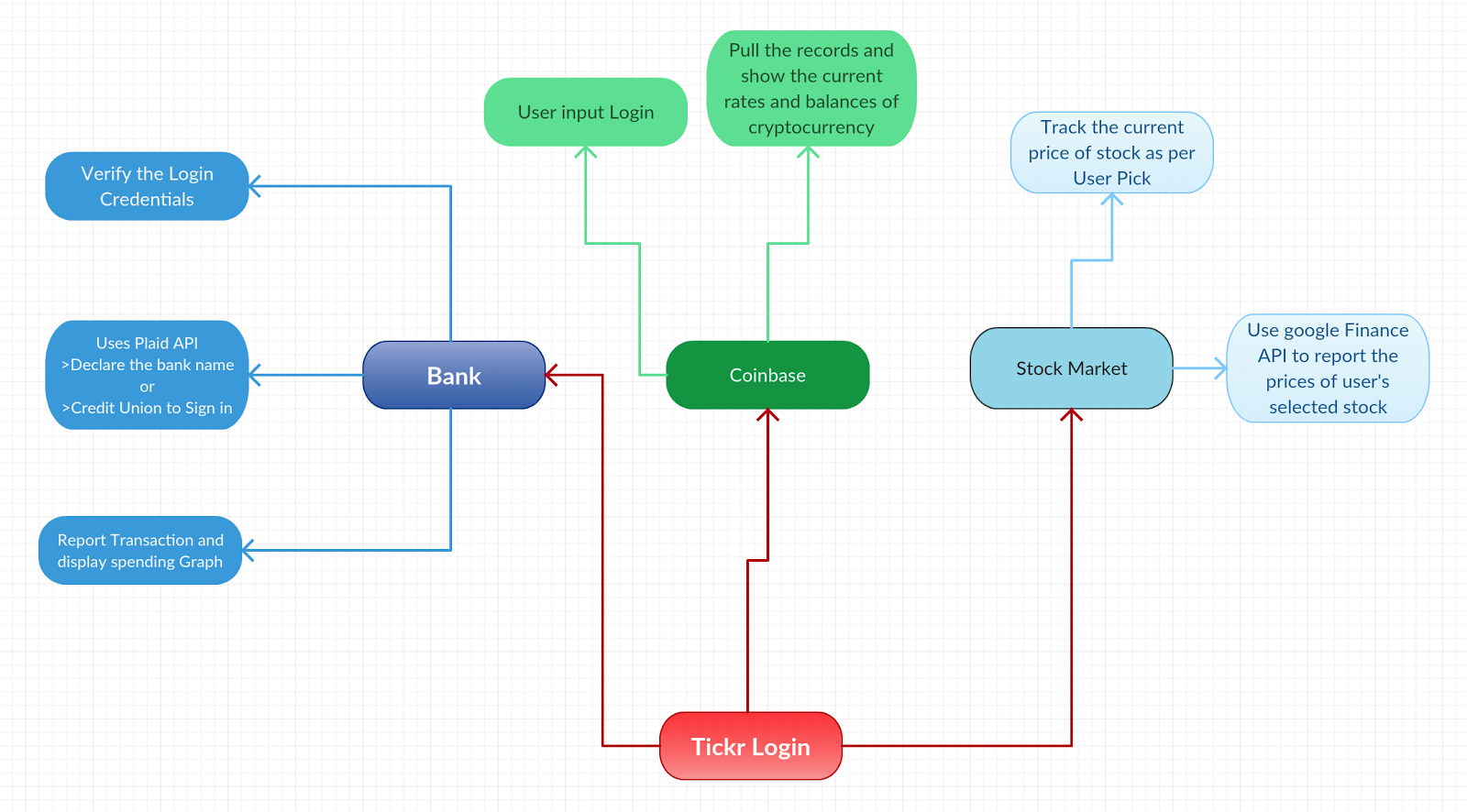
**Dependencies:** The user must have successfully logged into the Ticker app

**Initialization:** User clicks the Stock button to display stock prizes

**Test Steps:**

1. Once successfully logged into the Tickr app
2. Click the stock button to display stock prices
3. Check that the prices of the stocks match the current market value of the stock from a trusted stock broker’s site.

***Based on the above system requirement, the context model is drawn below:***



Section 6

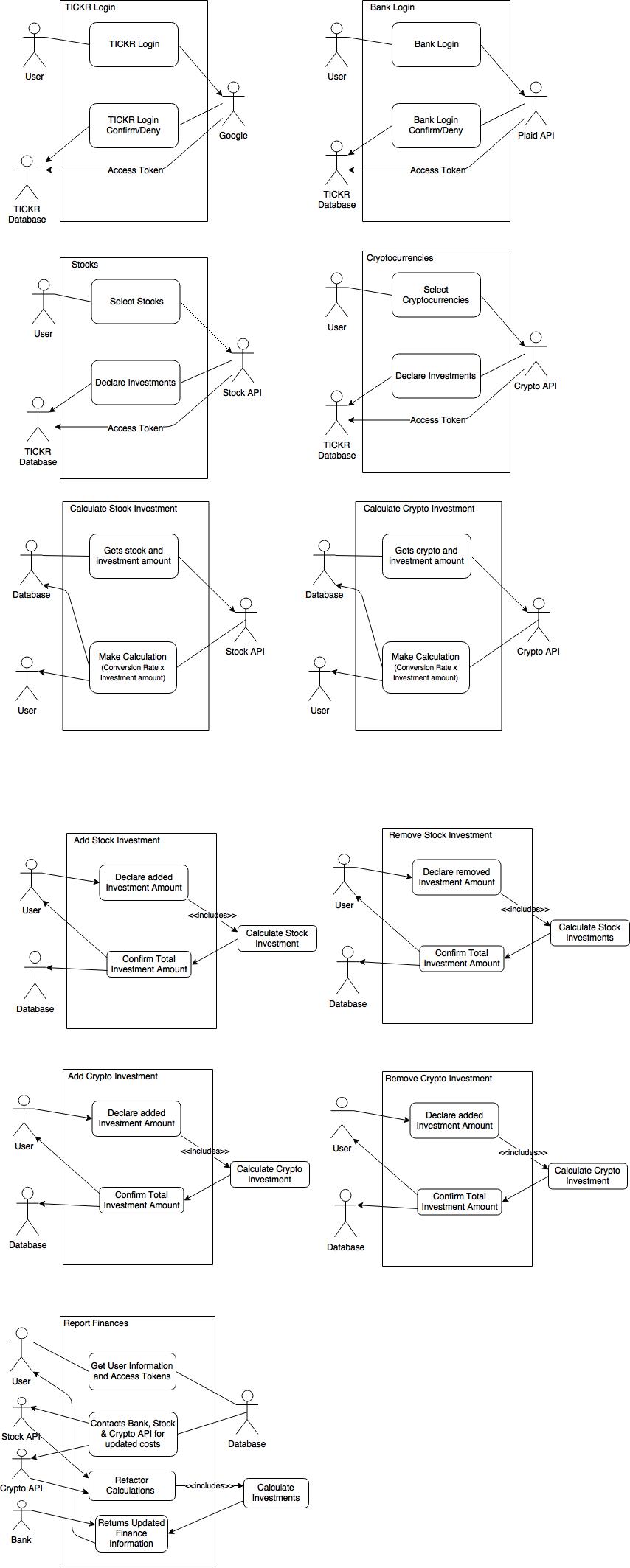
Use Cases:

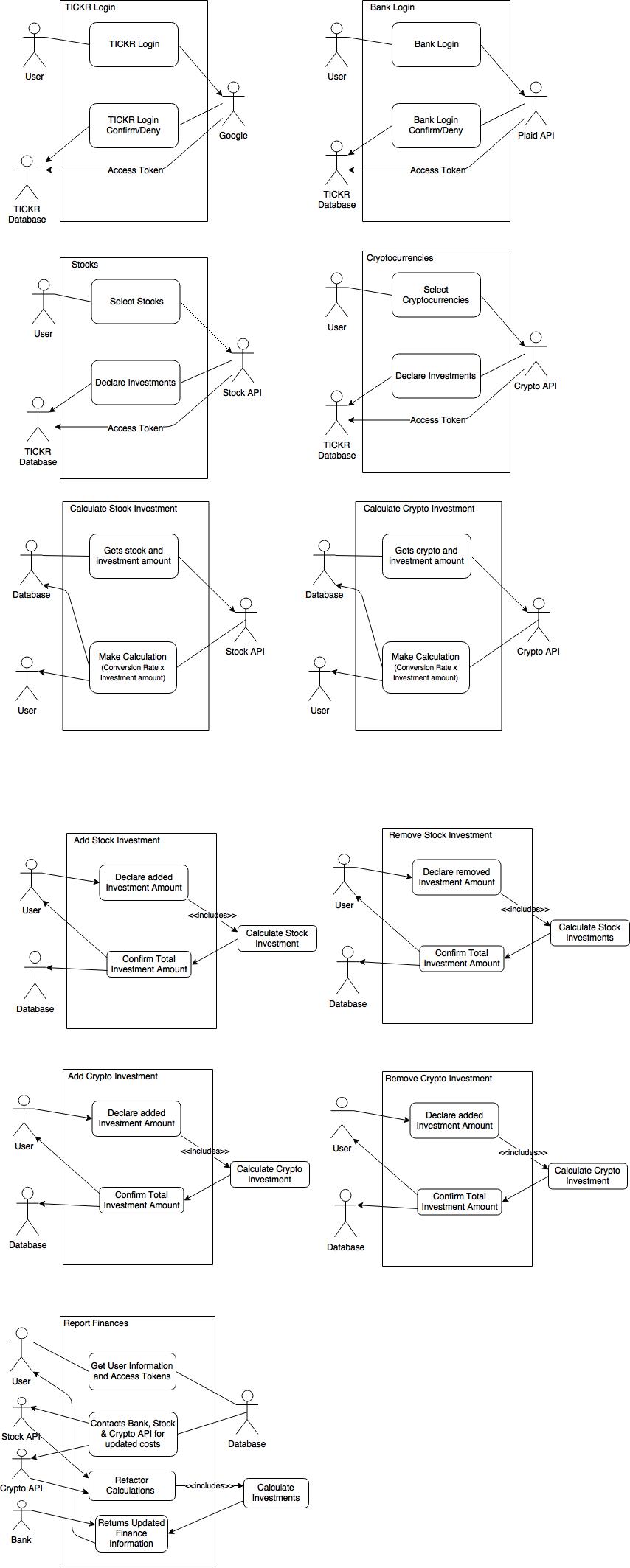
|  |  |
| --- | --- |
| Use Case #1 - TICKR Login | |
| Description | User will login to TICKR application via Google |
| Actors | User, Google, TICKR Database |
| Preconditions | * User is signing into TICKR for the first time * User is returning to login page after being logged out * User has already attempted login, and returning after failing to login |
| Inputs | The user will input Google Account Credentials (Email address and password). If credentials are validated, and access token is stored into the database |
| Outputs | Google will output to the system if the credentials were validated or not. If they are validated, Google will output an access token to the system, which is stored in the database. If they are not validated, then Google will output to the system that credentials were invalid, and the system will output this information to the user. |
| Process | 1. The user is presented with the Login page, where they will select the option to Login with Google. 2. The user will input Google Account Credentials, and Google will output to the system if the credentials are validated or not. 3. If they are validated, Google will output an access token to the system, which is stored in the database. 4. If they are not validated, then Google will output to the system that credentials were invalid, and the system will output this information to the user. |
| Postconditions | Successful:   * The credentials are validated with Google * The user receives confirmation that the credentials were validated * An access token is stored in the database for the user’s later access to the system.   Unsuccessful:   * The credentials are not validated with Google * The user receives a notification that the credentials were denied * No access token is stored in the database * Customer is re-prompted to sign in |

|  |  |
| --- | --- |
| **Tickr Use Case #2 - Personal Finance Only** | |
| Actors | Finance-tracking user, User Log-In (Facebook or Google), Bank or Credit Union, |
| Description | The user may log onto Tickr to check their personal finances. To access the user’s personal information that is encrypted and secured, a login via Facebook or Google is required. When setting up the Tickr account for the first time, you’ll enter your Bank name, account number, and routing number. You can also set up another account later! The Personal Finance tab will contain balance(s), recent transactions, and a spending graph. The investment tab will give the user the option to link their Coinbase account or track stocks later. |
| Data | Tickr Login (Facebook or Google), Bank Information. |
| Stimulus | User follows prompts to create/log into Tickr account and inputs bank information. |
| Response | Confirmation at every step that credentials were accepted; Full report of personal financial information (balances, transactions, spending graph) |
| Comments | The user must input at least one bank account to have content in the Personal Finance tab. |

|  |  |
| --- | --- |
| **Tickr Use Case #3 - Personal Finance & Crypto Tracker** | |
| Actors | Finance-tracking user, User Log-In (Facebook or Google), Bank or Credit Union, Coinbase Account |
| Description | The user may log onto Tickr to check their personal finances, and cryptocurrency rates for investing. To access the user’s personal information that is encrypted and secured, a login via Facebook or Google is required. When setting up the Tickr account for the first time, you’ll enter your Bank name, account number, and routing number. You can also set up another account later! Optionally, you can set up your Coinbase account, view your balances & the current rates! The Personal Finance tab will contain balance(s), recent transactions, and a spending graph. Also, the following page will contain current cryptocurrency rates and Coinbase balances. |
| Data | Tickr Login (Facebook or Google), Bank Information, Coinbase login |
| Stimulus | User follows prompts to create/log into Tickr account and inputs bank & Coinbase information and submits information to view all data |
| Response | Confirmation at every step that credentials were accepted; Full report of personal financial information (balances, transactions, spending graph); Coinbase balance(s) & current cryptocurrency rates. |
| Comments | The user must input at least one bank account to have content in the Personal Finance tab and sign into their Coinbase account. |

|  |  |
| --- | --- |
| **Tickr Use Case #4 - Personal Finance & Stock Tracker** | |
| Actors | Finance-tracking user, User Log-In (Facebook or Google), Bank or Credit Union, NASDAQ & NYSE stocks |
| Description | The user may log onto Tickr to check their personal finances, and current stock prices for investing. To access the user’s personal information that is encrypted and secured, a login via Facebook or Google is required. When setting up the Tickr account for the first time, you’ll enter your Bank name, account number, and routing number. The Personal Finance tab will contain balance(s), recent transactions, and a spending graph. Also, the following page will contain stock prices to make the best investment decisions! |
| Data | Tickr Login (Facebook or Google), Bank Information, and stocks |
| Stimulus | User follows prompts to create/log into Tickr account and inputs bank and stock information, and submits information to view all data |
| Response | Confirmation at every step that credentials were accepted; Full report of personal financial information (balances, transactions, spending graph) and current stock prices |
| Comments | The user must input at least one bank account to have content in the Personal Finance tab and select stocks to view content in the Investment tab. |





Section 7

Class Diagrams

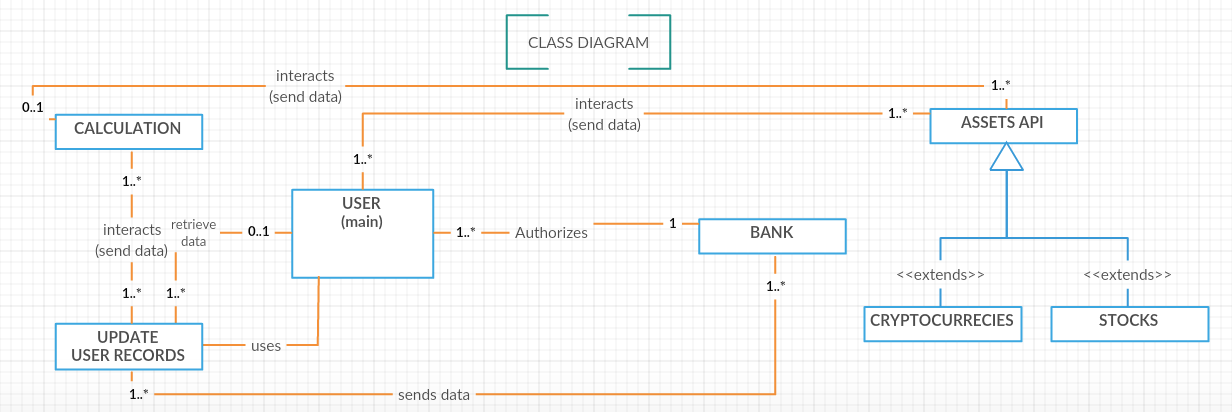
Class Diagram:

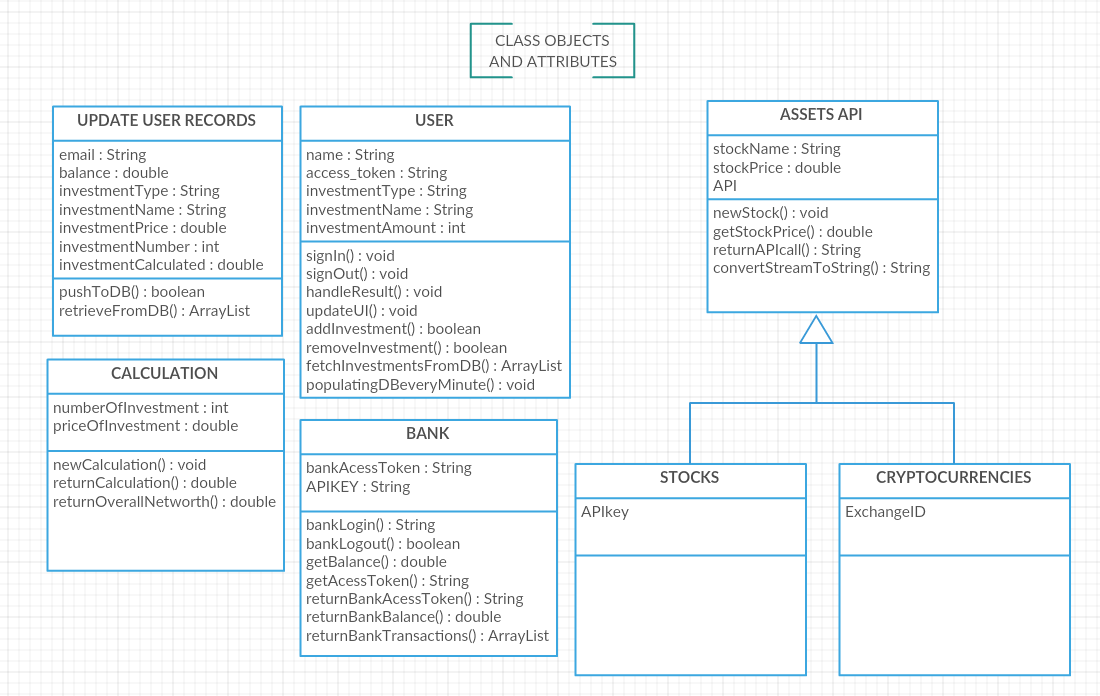
* Identify objects: The objects that are identified in Tickr are as follows:
* A User object which allows the user to login/logout as well as allow the user to enter types of investments and their amount.
* A Stocks object which takes in the type of investment from user which is defined as a stock as well as the stock name so that it can access a stock API (Google Finance) and return the price of that selected stock.
* A Bank object which allows the user to enter their credentials to login to their bank. It uses a third party API (Plaid) in order to access the bank safely.
* A Cryptocurrencies object which takes in the type of investment from user which is defined as a crypto as well as the cryptocurrency name so that it can access a cryptocurrency API (Coinbase) and return the price of that selected Cryptocurrency.
* A Calculation object which communicated back and forth with the database that stores all stock, bank, crypto holdings and then calculates the worth of a certain stock or crypto.
* What are the associations between them?
* The User interacts (sends data) with the Stocks, Bank, and the Cryptocurrency objects. The data sent form the User object to the bank is a request from the Plaid PI that we are using in order to authorize a user’s bank safely. The data sent from the User Object to the Stock’s Object is the name of the stock that the user entered. Also the data sent from the User object to the Cryptocurrencies object is the name of the crypto that the user entered. Since the Stock, Cryptocurrencies, and Bank object send their data to the database, the Calculation class pulls from the database and does the calculations of the worth of the portfolio/investment and then pushes it back to the database. It is then that the User Object pulls from the database to retrieve the new calculated data and display it to the user screen.
* What is their multiplicity?
* User to Stock – 1..\* → 1..\*
* User to Bank – 1..\* → 1
* User to Cryptocurrencies – 1..\* → 1..\*

With database included:

* Calculation to DB - 1..\* → 1..\*
* User to DB – 1..\* → 1..\*

* What are the attributes of the object?
* For the User object, the attributes include:
  + Name – Users name pulled from the Google login API
  + Access\_token – where we store the access token from login from Google Login API
  + investmentType – type of investment provided by the user so that we know what object to send that data to.
  + investmentName – Name of the investment can be a ticker or actual name of stock or cryptocurrency in order to send to the Stock or Cryptocurrencies object and pull the correct price information from the API’s.
  + investmentAmount – amount of shares/coins that is then stored in the database and then sent to the calculations class to get an accurate representation of value.
* For the Stocks Object, the attributes include:
  + StockName – name of the stock to request data from the API
  + StockPrice – price returned from the API
* For the Cryptocurrency Object, the attributes include:
  + cryptocurrencyName – name of the cryptocurrency to request data from the API
  + cryptocurrencyPrice – price returned from the API
* For the Bank Object, the attributes include:
  + bankAcessToken – stores the access token provided by the Plaid API in order to access the users bank for balance and authentication.
* For the Calculation Object, the attributes include:
  + numerOfInvestment – stores the number of shares/coins that the user has entered which it receives from the Database
  + priceOfInvestment – stores the current price of that particular investment.
* What operations are defined on the object?
* For the User object, the operations include:
  + Login()
  + Logout()
  + addInvestment() – adds an investment into the database by first contacting the stocks or cryptocurrencies object
  + removeInvestment() - removes an investment from the database
  + fetchInvestmentFromDB() – gets investment information and calculation from the database.
* For the Stocks object, the operations include:
  + newStock() – creates a new stock object
  + getStockPrice() – uses the name provided and access the API to get the current price of the stock.
  + pushResultsToDB() – send the results it got from the API (which is price) to the database.
* For the Stocks object, the operations include:
  + newCoin() – creates a new Cryptocurrency object
  + getCryptoPrice() – uses the name provided and access the API to get the current price of the cryptocurrency.
  + pushResultsToDB() – send the results it got from the API (which is price) to the database.
* For the Caclulation object, the operations include:
  + newCalculation() – creates a new Calculation object where it takes the attributes and does the specified calculations based on the price and amount of investment.
  + returnCalculationToDB() – pushes the newly created data (worth of investment) to the database
  + returnOverallNetworthToDB() – returns an overall portfolio worth based on past and current investments back to the database.

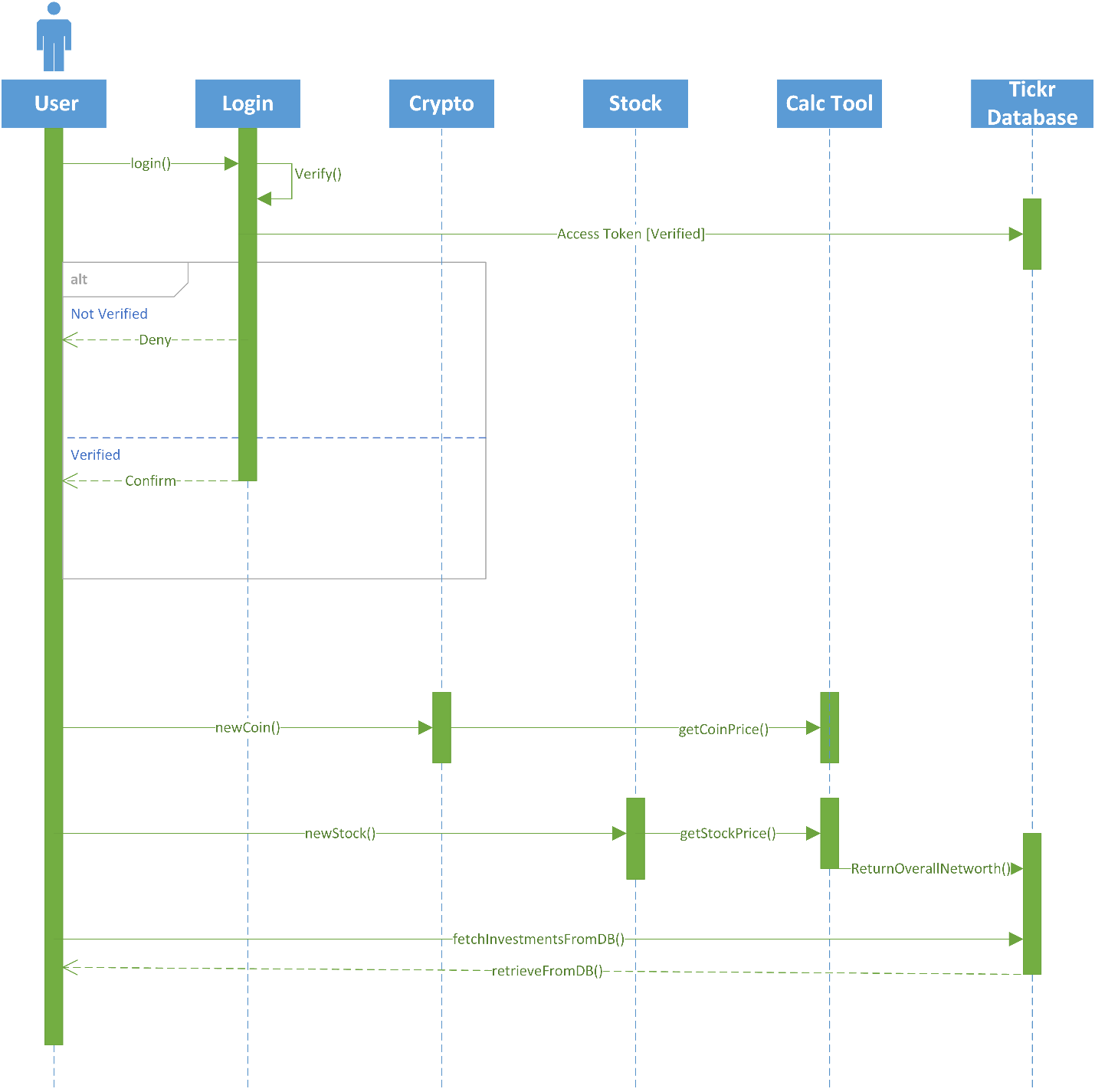




Section 8

Behavioral Modeling

Sequence Diagram:

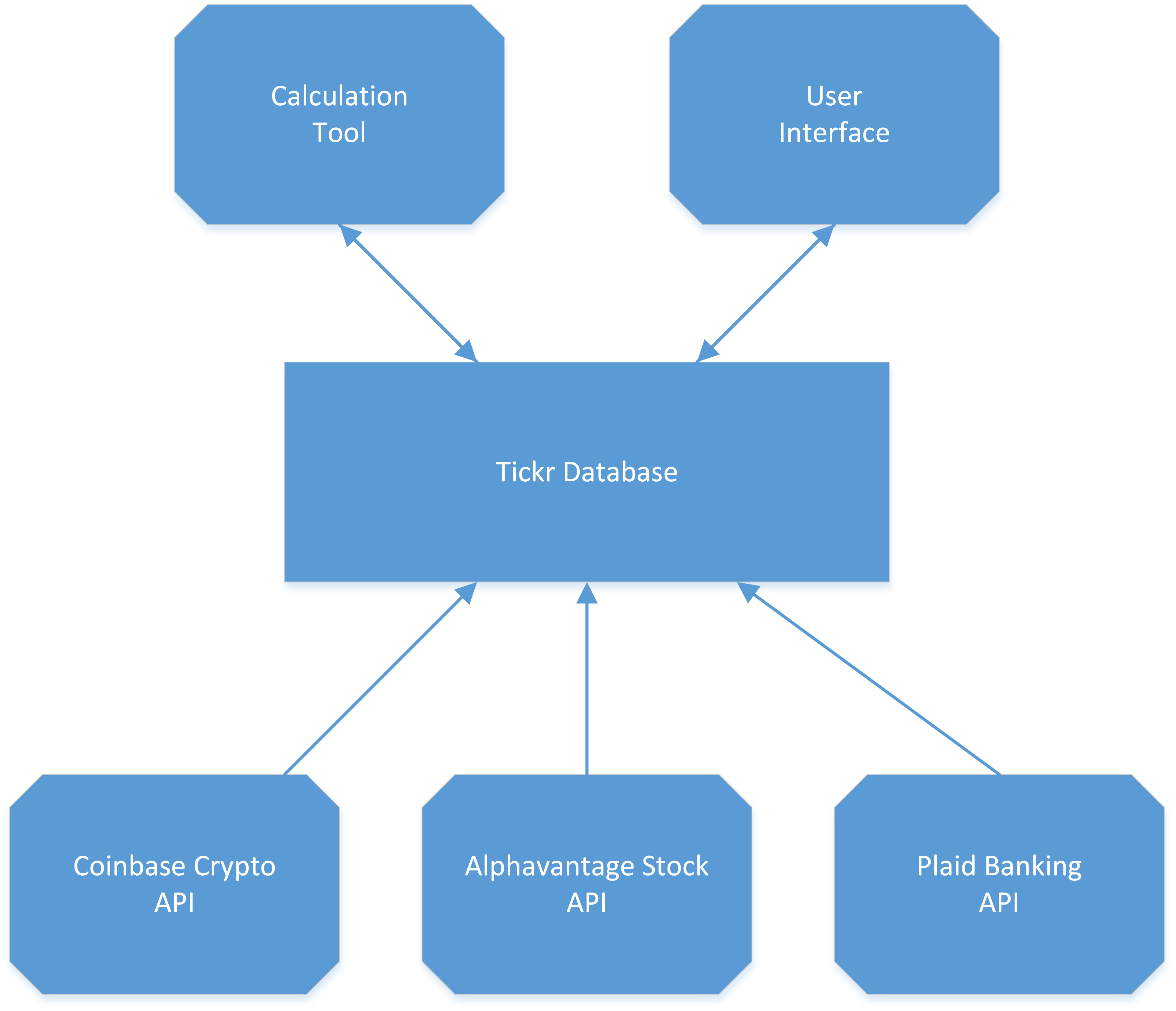


Section 9

System Architecture

**Architectural Diagram:**

After analyzing the requirements and specifications, we have determined that a shared repository architecture is the best fit for Tickr. A shared repository is a database that multiple tools connect to and use to share information. This is perfect for Tickr as we are pulling data from multiple APIs and then manipulating that data elsewhere. We pull up to date prices for stocks and crypto currencies through our APIs, as well as the user's personal banking info, and store these in the shared Tickr database. Then our calculations tool pulls this information, as well as data on how many shares or coins a user has, to calculate the users net worth. After this we can display the relevant information to the user as a trend line of their wealth over time. Since most of the functions of our system rely on the same data, the shared repository architecture makes the most sense.

****

Section 10

Implementation

**Implement the system design**

<https://appetize.io/embed/z8u9chhpxv7b66wfnt005q3bam>?

How to test implementation of the system?

I. Open the URL given above

2. Once open the URL, Click TAP TO PLAY on the Emulator.

3. After clicking TAP TO PLAY, it will launch the Tickr App

4. When app is launched, you can clicked the sign in Button.

5. When sign in Button is clicked, it will ask you to provide the gmail login credentials like  
     username and password.

6. Google will verify the account and let you sign in if provided information is correct.

7. After login, User can select any three options from the Tickr home page.

Section 11

Appendix

**Appendix A:**

**Youtube video links:**

Project Video Links:

Video 1

<https://youtu.be/5r5EhSzsBiA>

Video 2

<https://youtu.be/YmF81g4QNUs>

Video 3

<https://youtu.be/THC7Oxu25DY>

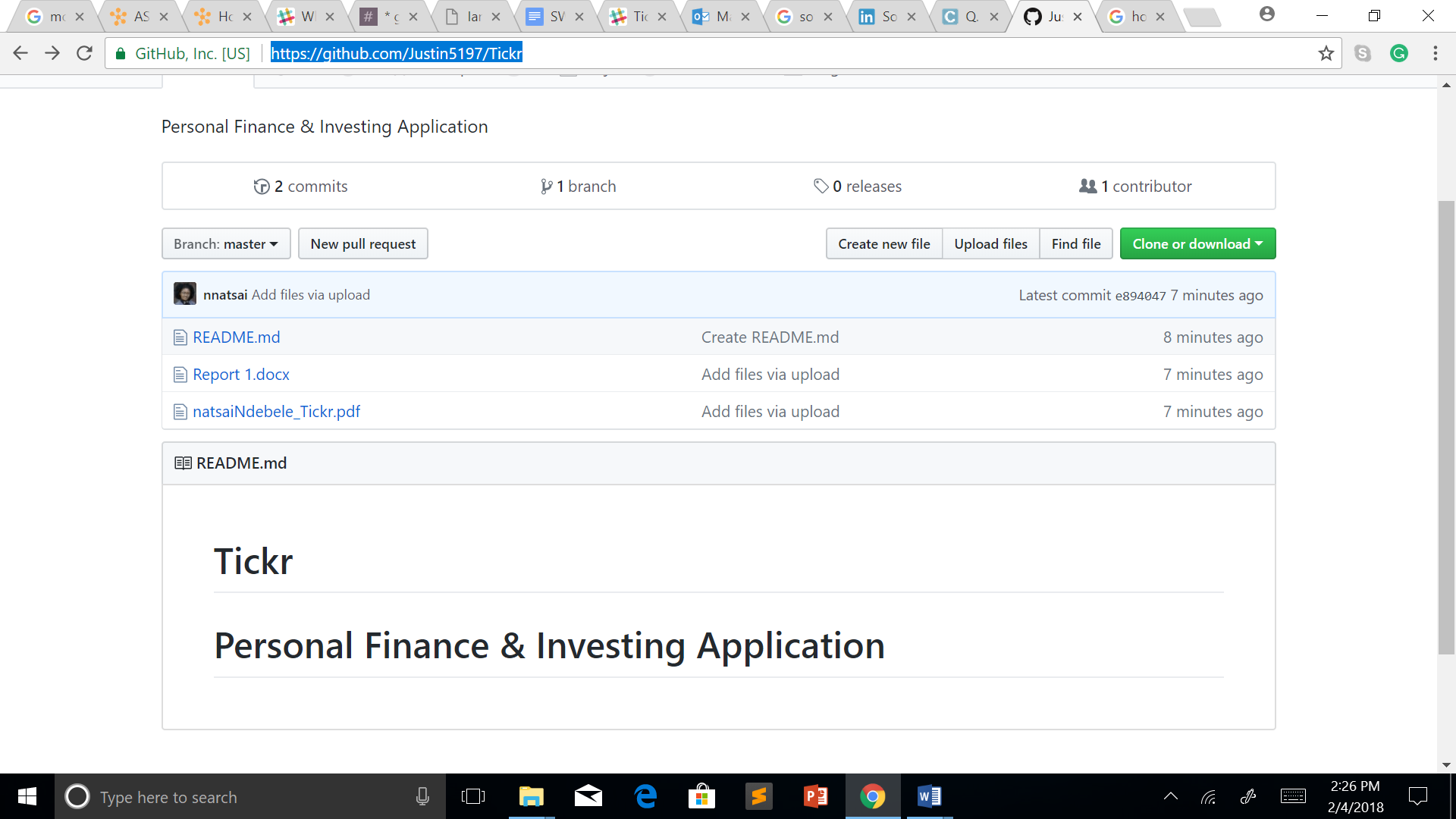
**Appendix B:**

**Git Hub Logs:**

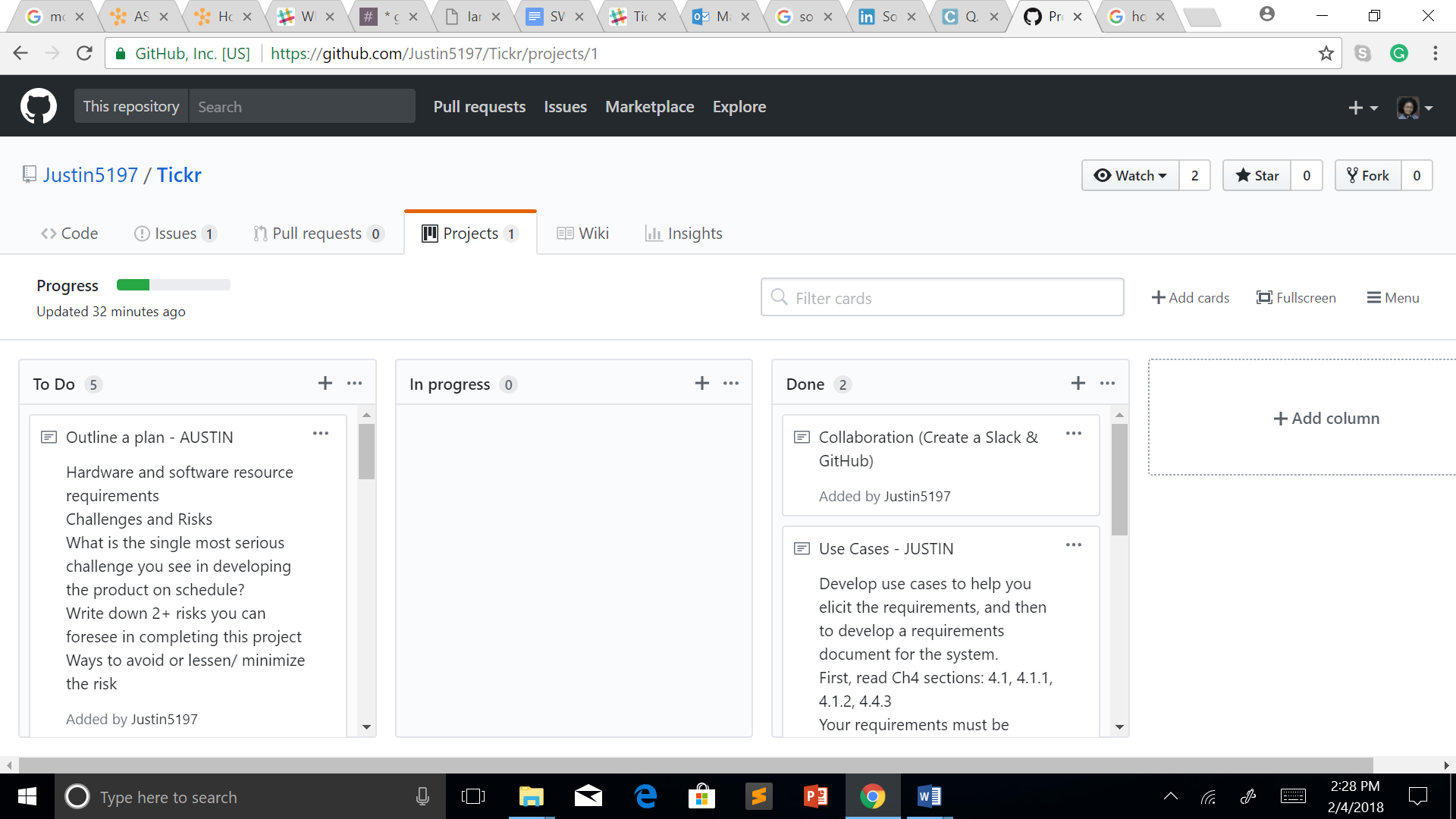
**Assignment 2:**

Git hub link: <https://github.com/Justin5197/Tickr>

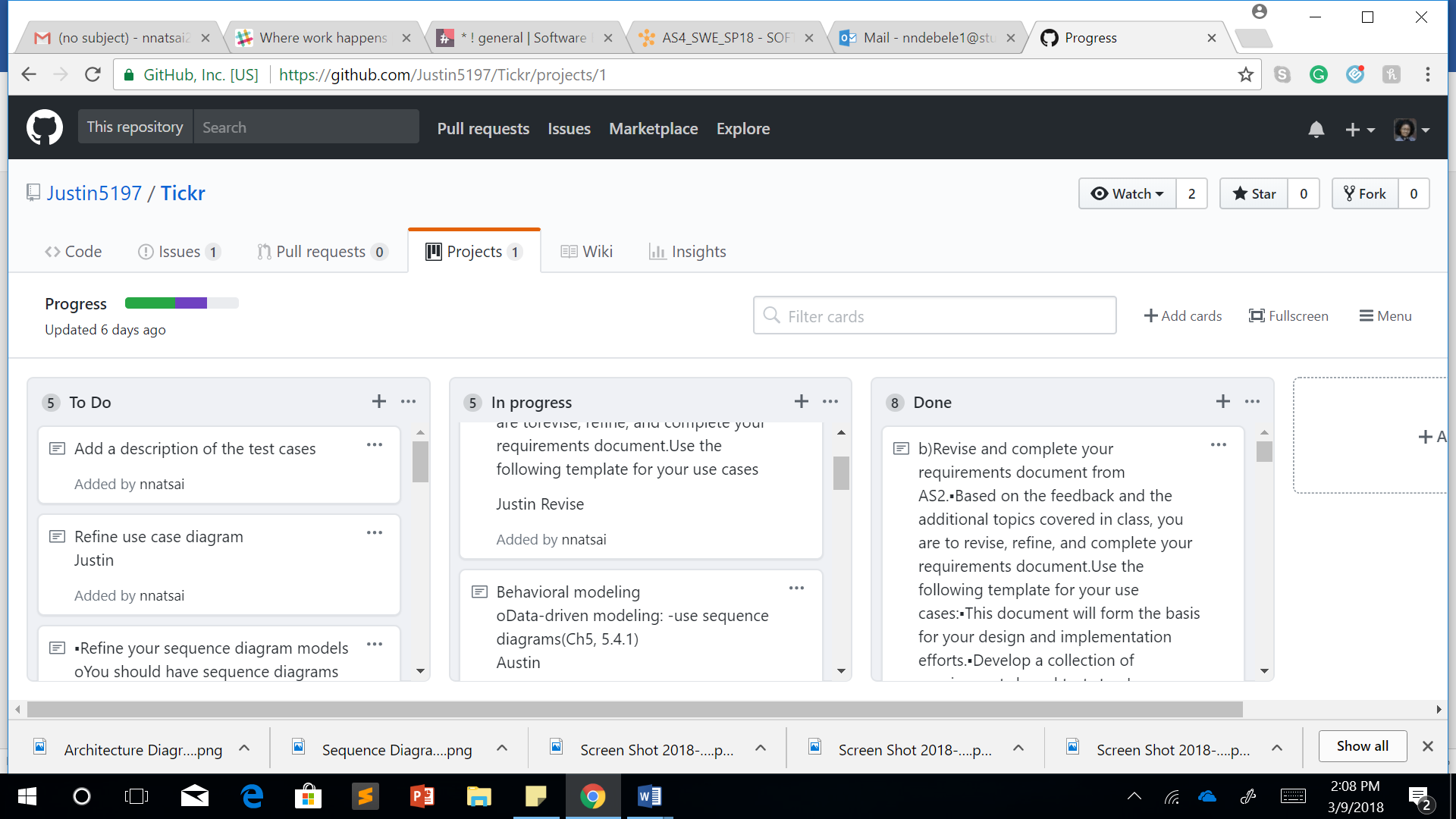
Screen shots:

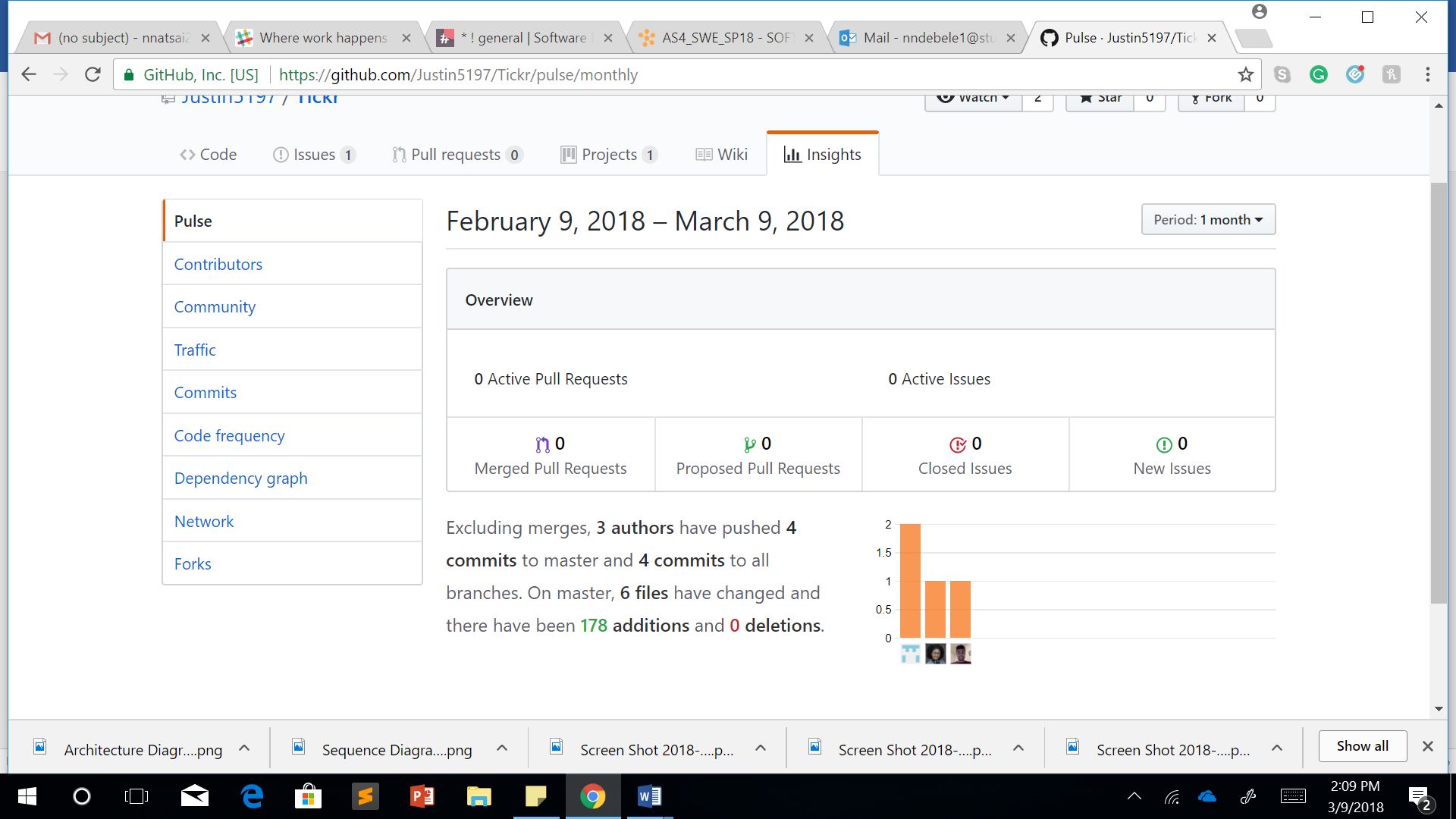


Git hub progress tracker:

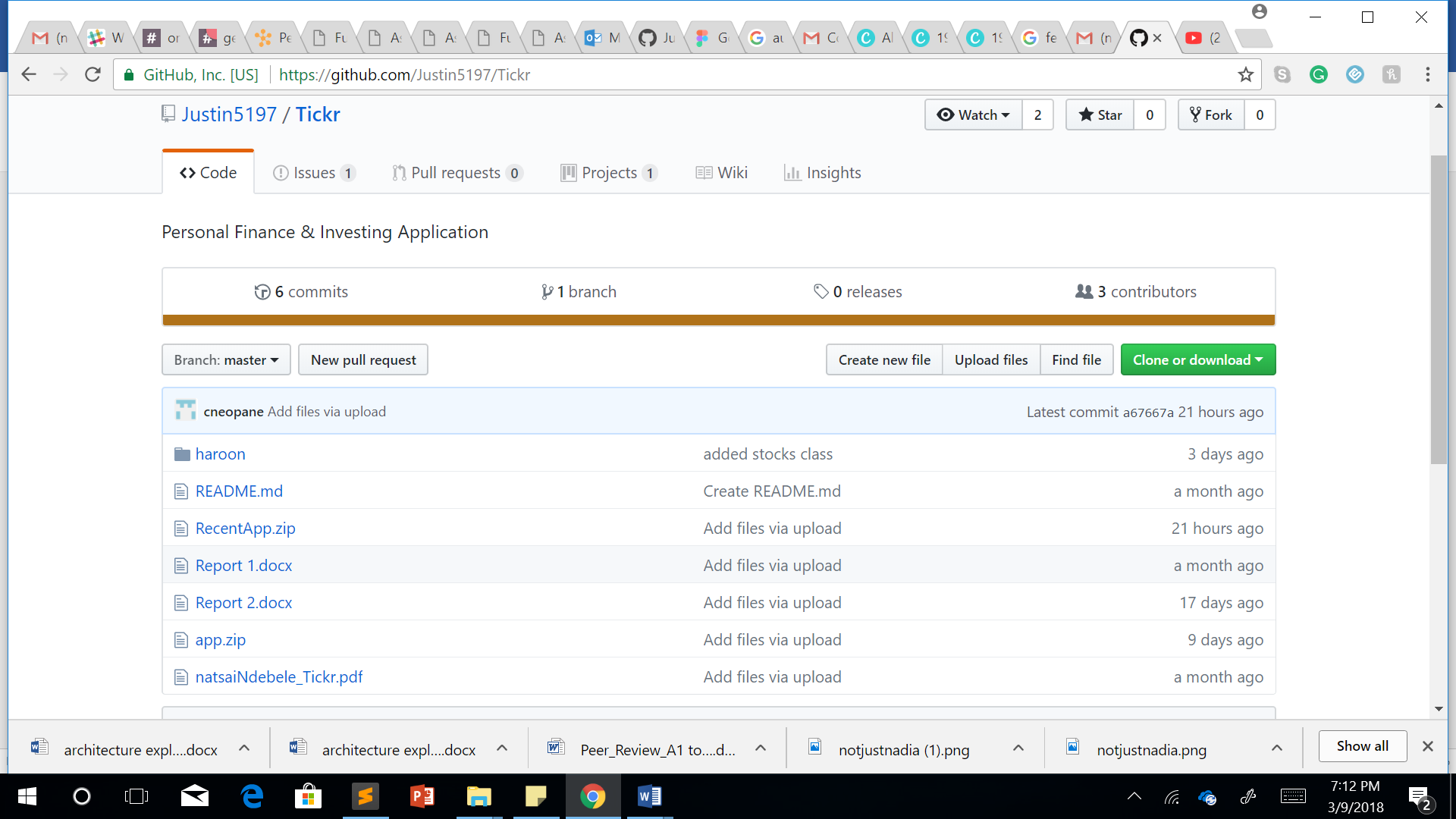


**Assignment 3:**



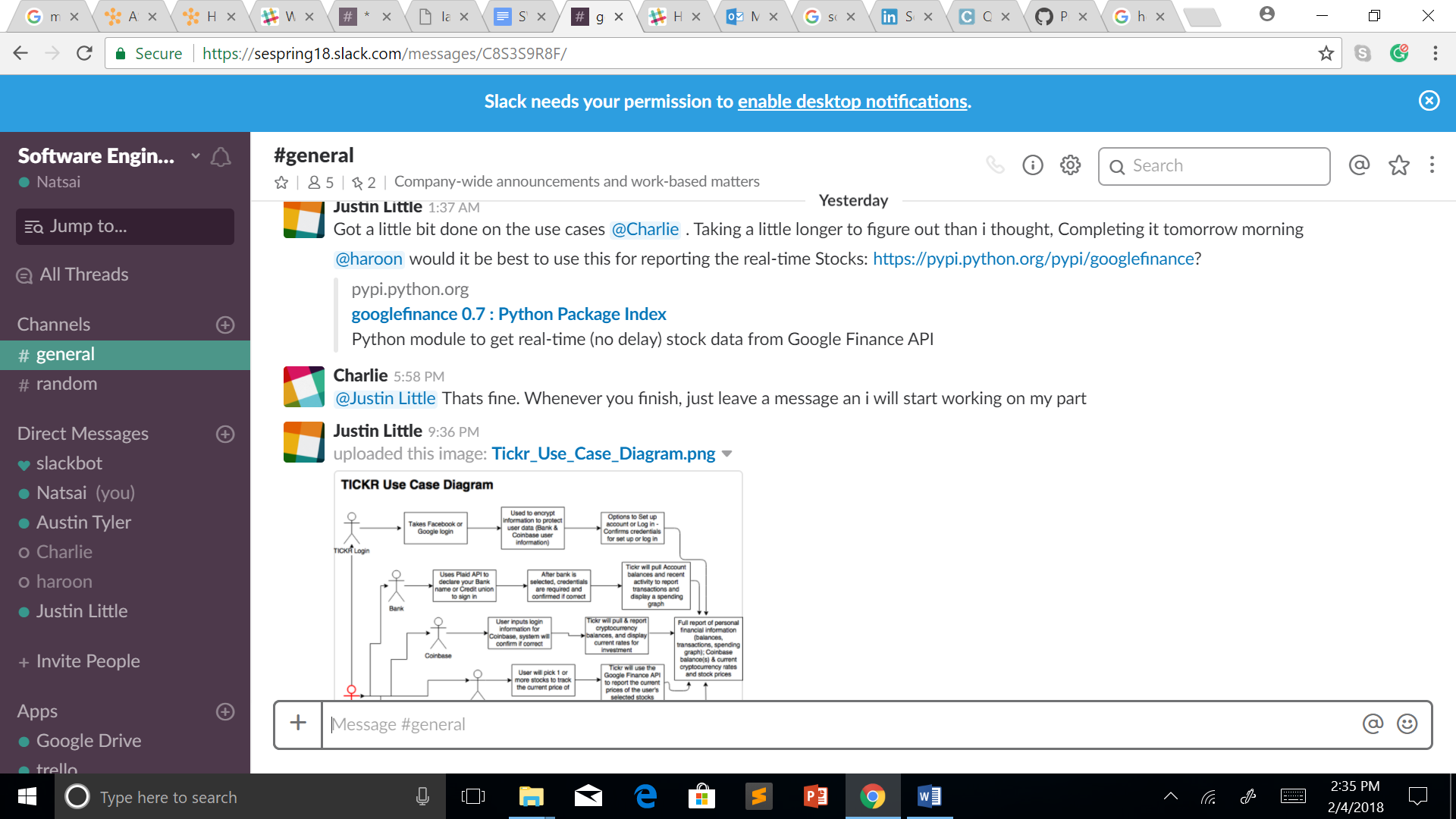


**Assignment 4:**

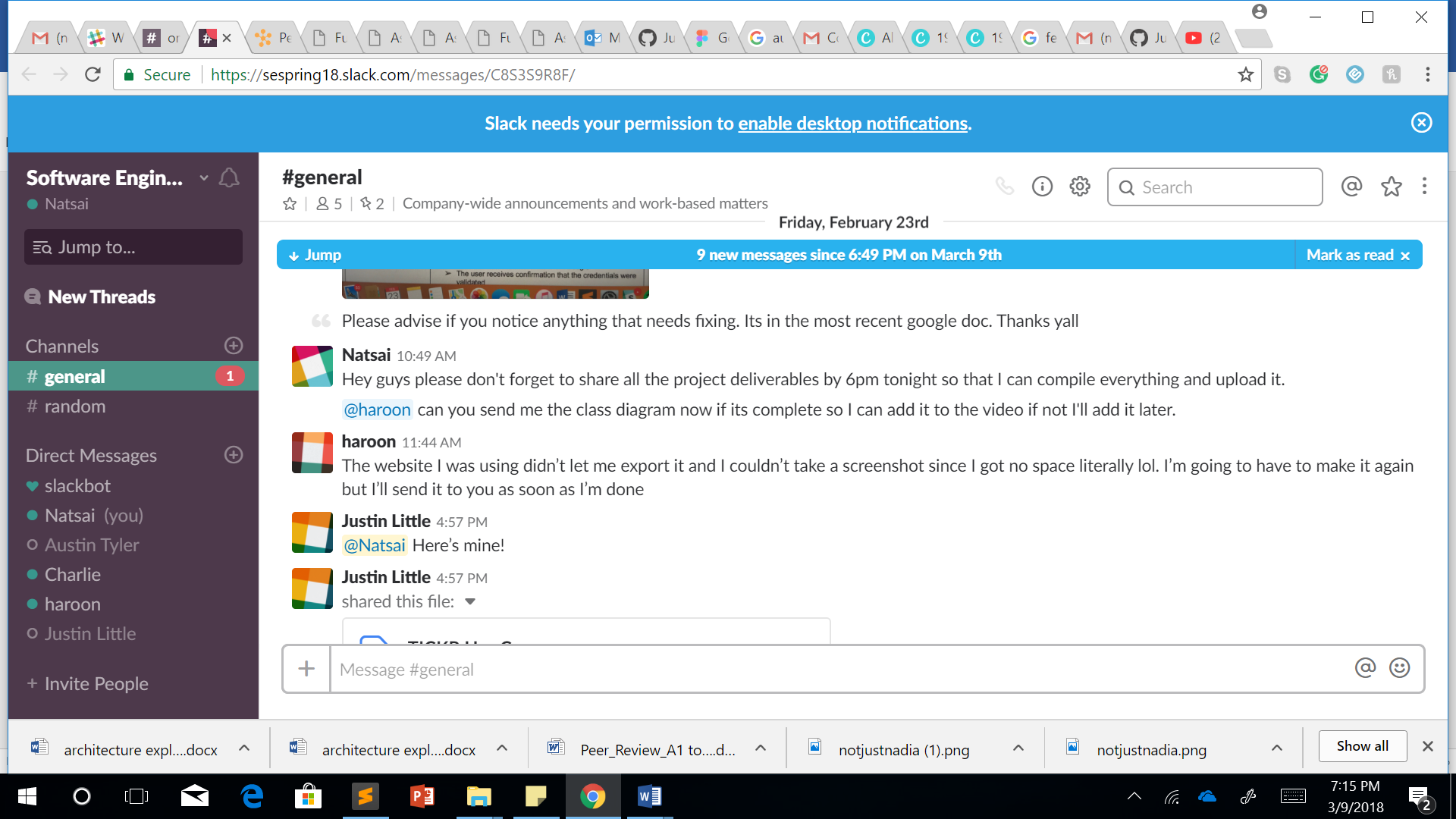


**Appendix C:**

**Assignment 2:**



**Assignment 3:**



**Assignment 4:**

