# **Title: Bro-Online**

# *Table of Contents*

1. Title Page
2. [Table of contents](#_Table_of_Contents)
3. [Abstract/summary](#_Abstract/summary)
4. [Main text](#_Main_Text)
   1. [Context](#_Context)
   2. [The Product](#_The_Product)
   3. Considerations to:
      1. [Sustainability](#_Sustainability)
      2. [Legal/Ethical](#_Legal/Ethical)
      3. [Intellectual Property](#_Intellectual_Property)
5. [Project Planning](#_Project_Planning)
6. [Conclusion](#_Conclusion)
7. [Acknowledgements](#_Acknowledgements)
8. [Reference](#_Reference)

# **Abstract/summary**

Web Application which is used to form and maintain the groups. Built via ReactJS and Firebase. As a user, you can register and login to the site. Then, you can create/delete groups. You can add/remove members. Furthermore, you can search for people and the other groups from the search page, and group chat via messaging.

This application purpose is to give people a tool to form groups, which they can use for their goal, e.g. self-improvement.

The concept of brotherhood has influence on this product.

# **Main Text**

## **Context**

**Introduction**

In beginning, I set out creating a web app which focus was to influence brotherhood formation in our society and it should only be used by men. However, now this idea has evolved.

Now I think that brotherhood is not the focus; the focus should be placed on the individual's wants; "what is the person's goal?"; be that male or female.

Once the person has establish their desire, then they should look into creating/joining groups as an option if that helps them to accelerate towards their goal. Creating/joining group is what my product could help with.

In conclusion, brotherhood is the by-product of a goal (e.g. survival). For example: brotherhood is more apparent during hard times such as poverty, or war. In these circumstances, if people chose to be lone-wolf, then they would have lower chance of survival. 'Advantages of brotherhood' shows the power of brotherhood, and why we humans naturally form it during tough times.

Please note that, since I have already done fair amount of background researching during summer 2020, I will not repeating it on here. If you would like to view this document then, please follow the link below:

<https://cseegit.essex.ac.uk/ce301_2020/ce301_rai_ajaya/-/blob/master/Challenge%20Week/Background%20Research.docx>

On here, I will be researching on slightly different topics to find if the app supports the new idea.

**What is sisterhood?**

Like the brotherhood, sisterhood is the relationship between sisters [1].

In the documentary, "Becoming"[4], although this documentary is about Michelle Obama. Several parts in the movie, there has been sisterhood moments.

E.g.

13mins "The African Amercian Museum in Philadelpha"

22mins

29 mins “Chicago Illinois”

49 mins

I would say these are sisterhood moments because from my own experience of training with other men (e.g. Martial arts), and being part of a gang; the clips above resonates similar energy to that of a brotherhood.

The difference between the brotherhood and sisterhood is that brotherhood is much more about doing and sisterhood is much more about talking.

Is there a difference between men and women? If a brotherhood can consist of men from different race why can it not have women too?

On the discussesion, "Whether men and women can ever be equal".

JP says that men and women are mostly alike. But, if we look at the population level at the extreme parts, the small difference can be extreme. E.g. He states that most aggressive people 1/100; they are all male and that is why there are mostly male inmates. 3/4 in humanity and social science are females.

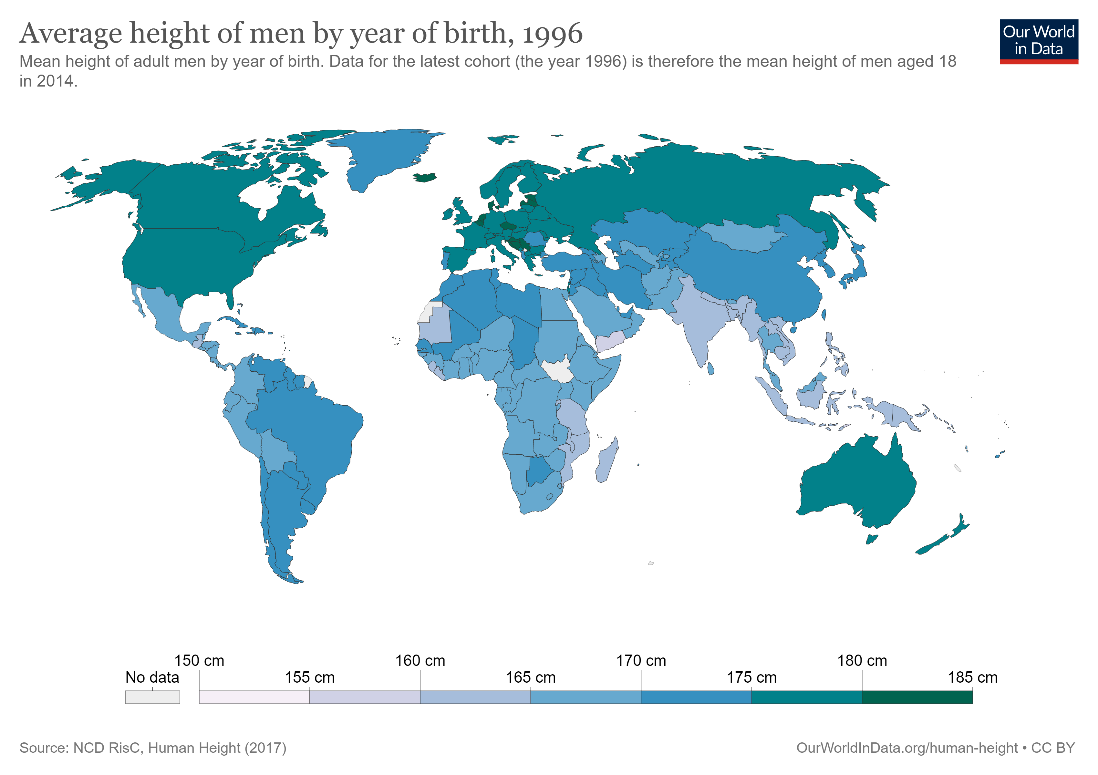
He also states that the Scandinavian countries are the countries that is pushing the equality the hardest, but its having the opposite effect regards to gender.

Sophie Walker, leader of the Women's Equality Party also makes good point where she states that

* The scandinavian countries haven't got there, but JP comes back and says that it is heading towards the equality but equality between men and women is not decreasing instead its increasing. So given this, the assumes that the end goal === huge difference rather than equality between men and women
* Men outnumber women by 2/1 in parliament, in local government and children grow with the idea of pink and blue, and the media also pushes this idea. Its fair to say that men have always been in the higher power than women, and kids grow of watching this from an early age so this could have an effect.

Furthermore, from my own experience, I have also found that small difference does make difference. For example: I came to the UK when I was 13 years old, in the beginning I couldn't mix with the white group there was "us and them" mentality. However, now I realise that we are mostly the same because I am now able to speak English and be part of the group in the martial arts were most of the people are white, and when we train, its the similar feeling that I gotten when I was playing with my race back in Nepal.

However, there is difference for sure. For example, I am originally from Nepal which have the shortest people on the planet, as shown by the image below:



[3]

This height difference does play a major part in martial arts grappling and basketball.

Overall, I have concluded that human are mostly similar, however, the small difference can be significate on the extreme parts. And this differs from gender and race; and I would assume that there are more differences. The reason why I research these difference was to find out if women would be interested in the brotherhood, and the conclusion that I have reach is that yes most women would not be interested in the brotherhood. But, along with this finding I also come to realise that even men will choose to be with specific men's group (e.g. Height, race) because it might provide that small difference. Overall, I choose to make the app accessible to everyone because in the end, its the people's choice and there are many things that I and we do not know. So, giving power to people and letting them experiment acts as a learning in itself.

Ref:

[1] <https://dictionary.cambridge.org/dictionary/english/sisterhood>

[2] [Jordan Peterson discusses whether men and women can ever be equal](https://www.youtube.com/watch?v=Iy4vq8RdPGU)

[3] <https://ourworldindata.org/human-height>

[4] [Becoming | Official Trailer | Netflix](https://www.youtube.com/watch?v=wePNJGL7nDU)

## **The Product**

**Overview**

Please view the [Abstract/summary](#_Abstract/summary) section for this part.

**Sign Up**

If you go to the site's homepage without logging in, you will be redirected to the 'Sign Up' Page.

Fill in the details and click submit. The details will be sent to the backend in JSON format which uploads the data to the Firestore.

Then, the firebase server generates and sends a token to the Frontend (user’s browser). After the successful submission, you will be directed to the homepage.

Please note that the validation of the form has not been implemented.

**Login/Log Out**

To login, you have to start from Sign Up page, which have text “Already registered click here to login”. Please click on the “click here”. And this will lead you to the Login Page.

On the Login Page, simply type the “User Name” and the “Password” that you registered with, and click the “Login” button. When you click the “Login” button, the “User Name” and “Password” are sent to the backend via POST request. The backend checks if these datas are valid and responds with a token. The token is then used via browser to indicate that the user has been logged in.

After the login process, you will be directed to the Home Page.

In terms Log Out, please click on the Log Out icon on the top right, which removes the token from the browser.

**Adding a Group**

The user must be logged in to add groups, and it can be done from the home page.

Simply, type the group name and click submit. When you do this, the name of the group is sent to the server with a POST request.

Then, the backend adds the group to the user's document and a new group is created on the group document which will have the user ID in the Firebase. Two changes need to take place so that we can keep track of the group member's individaul groups, and the group members on a given group.

**Deleting a group**

The user must be logged in to delete a group, and it can be done on the home page.

Simply click on the “Delete” button next to the Group that you want to delete. Once the button is clicked, the DELETE request is sent to the Backend, which deletes all the data of the group and the group is automatically deleted.

**Adding Group Members**

On the app, please create a group and click on it, which will lead you to the Group Page. Then, click a button which says “Add member” which will lead you to a search page from there, you can choose people that you want to your group. After you have chosen a person, it will redirect you back to the Group Page and the chosen person would be display.

People that you see on the Search Page are shown by first retrieving the registered users from the Firestore and rendering it on the browser.

Once you have chosen a person for you group, POST request is sent to the backend which:

* 1. Adds the user to the group. Purpose: keep track of the group members on a give group
  2. Adds the group to the user’s collection. Purpose: keep track of the group member’s individual groups.

**Removing Group Members**

To remove a group member from the web app, you have to go to the group which has members and click on the “Remove” button. Once the “Remove” button has been clicked, DELETE request is sent to the backend, which deletes the two actions done on the “Adding Group Members” section above.

**Bio on the Group Page**

The user can Edit the Bio on the Group Page. To add something to the Bio, simply type and click submit. The click of the submit button sends the POST request to the backend with the text that was written, which gets added to the Firestore.

**Chat System**

The purpose of the message on this app is to plan meet ups, and also come to agreement of the regular sessions.

This has been achieved with the help of Firestore and the ReactJS. Firebase has a function where it listens to any changes on the database and once any changes happen, it detects it and send the data to the front end which updates the messages without needing to refresh the page which is thanks to the ReactJS, which updates and renders just the right component when there is a change in data without needing to refresh the page.

**Search Page**

You can navigate to the Search Page by clicking the search icon from the navigation bar on the top.

The Search Page displays the users registered to the web application, as well as the groups that are live on the site. This is done by sending two separate GET request (e.g. separate for users and groups) to the backend and displaying the result on the browser.

search\_grp\_bro.js also has a component JoinAgrp which purpose is to display a "Join" button that the user can click so that they can join a group.

Alongside the each group, you should see “Join” button. If you click the “Join” button, a POST request is sent to the backend with data:

* UserName
* GroupID

We need the data above to first detect the group and then add the user to the group.

**Testing is covered in the Technical Documentation below**

**Technical Documentation in GitLab**

Please follow the link below to view the TD which goes into detail of each functionalities described above:

<https://cseegit.essex.ac.uk/ce301_2020/ce301_rai_ajaya>

## **Sustainability**

Since the project is web application, it requires server if it were to be deployed live. Depending on how the server is run, it can cause sustainability issues.

Google firebase allows live deployment, so their servers would be used to host the application. The "Google Environmental Report 2019" [1], page 3 shows that they are very committed to make their services sustainable. For example: it states that, they have been carbon neutral from 2006-2018. Additionally, it states that Google have been able to get their energy 100% from the renewable resources for two consecutive years.

Reference:

[1] <https://services.google.com/fh/files/misc/google_2019-environmental-report.pdf>

## **Legal/Ethical**

The web application draws its inspiration from the brotherhood which is amoral. E.g. Brotherhood is apparent in the military, but it is also apparent in a criminal gangs. [2] shows the criminal activities from the gang called Bloods in America. So, there is a risk that this application could draw attention from the criminal gangs or encourage people to form their own criminal gangs. This is especially true when the kids today are influenced by the rap culture that advocates gang, drugs and violence.

One way to prevent negative groups formation in the application is via design of the website which indicates positive and heroic. Another step could be monitoring the groups for illegal activities. But, this is a volition to the individual's privacy. So, there has to be a balance between privacy and protecting the community, which means only allowing monitoring to the people who have that responsibility, and ensuring a good standard security.

Could talk about the security that I learnt from NS to gain more marks.?

Consideration has to be made in terms of meeting the GDPR (General Data Protection Regulation) [3], if the web application were to be live released. Main things include:

1. Allowing user to have control of their data, e.g. At any point if they want to delete all their data, then it should be allowed
2. Pseudonymisation of the personal data
3. If the data has been breached, it requires company to let the users know immediatly
4. Encourages the company to do everything that they can in order to ensure that the personal data has been protected well

How to meeting these regulations:

1. New feature should be created for this, where user is given access to a button which they can click in order to delete all their data in the database
2. Pseudonymisation could be achieved by following the relational database theory, to the point where the data is highly segmented. This should enable sensitive information to have its own table (collection in terms of Firestore). This means that if an attacked hacked to the database, and obtained a table, it should not make sense because the table would not have a lot of information to make sense of it.
3. If the breach were to occur, letting the users know should be straight forward by sending email which they signed up with
4. This part can be achieved by following the standard, also looking at how others have done this

Ref:

[2] <https://search.justice.gov/search?query=bloods&op=Search&affiliate=justice>

[3] <https://gdpr-info.eu/>

## **Intellectual Property**

Intellectual Property is applied differently in various parts of the data. These include:

* Person's profile, e.g. Name, age, location, picture
* Groups that are created
* Chat messages

Person's profile should be owned by the individaul who created it.

Groups should be owned by the leader, once the leader functionality is added.

Chat messages should be owned by participates.

For the Group owner and the Chat messages, the Intellectual Property changes from individual to individual, e.g. Some members might leave the group which means that they should no longer have the rights to the Messages. This transition of the Intellectual Property could be an issue because due to bug random user might get added to the group chat; giving him/her rights to the chat. Or user might accidently add wrong person to the group.

To prevent the bug from adding random people to the group, tests and peer review of the code can be used. And to combat user from adding wrong person to the group, a confirmation alert box could be used.

# **Project Planning**

## **Introduction**

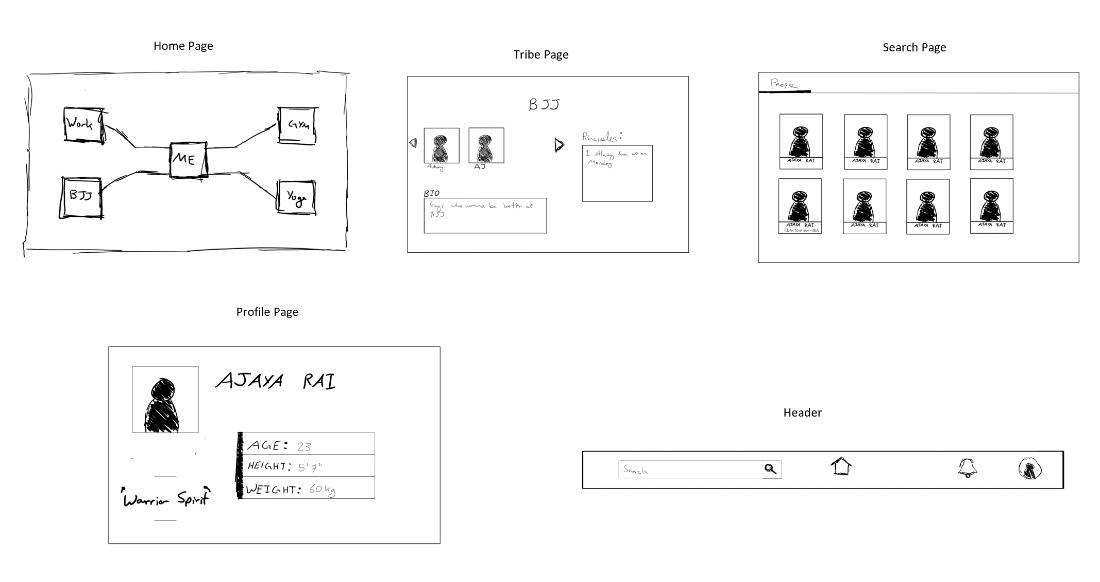
On the main text, there are two different coloured paragraph; Green explains the adaptation that needed to be made halfway through to the project, Purple touches on what was learnt form the planning. Furthermore the Risk Management section has another Purple coloured paragraph.

## **Main Text**

During the Challenge Week, I had set out to complete all the [logic](https://cseegit.essex.ac.uk/ce301_2020/ce301_rai_ajaya/-/blob/master/Challenge%20Week/Logic%202.drawio) for the MVP. To open the file, follow the steps below:

1. Download the .io file
2. Go to [draw.io](https://app.diagrams.net/)
3. Click on "Open Existing Diagram"
4. Select the downloaded file
5. In the end, you should see the *flow chart*

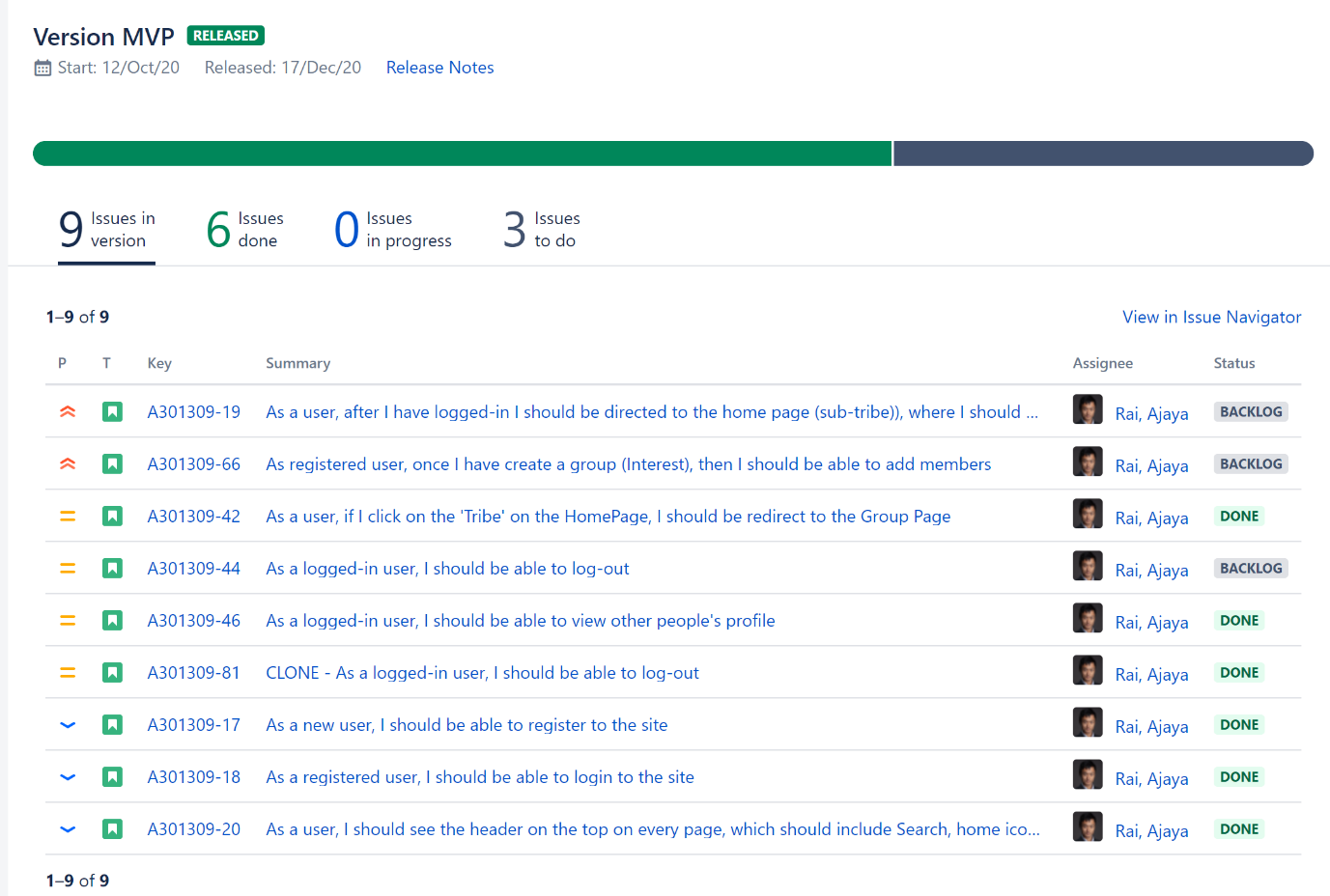
After MVP, I wanted to focus on the GUI, as shown below:



Main features that I planned on completing were:

* User being able to register
* User being able to login
* User being able to add/remove groups
* User being able to add/remove members to the group
* Having a profile page where user can have their bio and change picture
* Having a leader functionality where only the lead can add members to the group
* Search page where the user can see the people close to him + see other group
  + User can click on the people to see their profile
  + User can click on the group to see its details
* User can logout

Image below shows what I actually got done:



Regarding the above features, the features that completed include:

* User being able to register [DONE]
* User being able to login [DONE]
* User being able to add/remove groups [half DONE]
* User being able to add/remove members to the group [half DONE]
* Having a profile page where user can have their bio and change picture
* Having a leader functionality where only the lead can add members to the group
* Search page where the user can see the people close to him + see other group
  + User can click on the people to see their profile
  + User can click on the group to see its details
* User can logout

When I say [half DONE] for the

* Add/remove groups
* Add/remove members

The Add/remove members was done, but only the add part was done for the groups not the remove. Furthermore, the process required refreshing the page. For example, lets say I wanted to add a group, I would type the group's name and hit submit to render this to the website, the user had to refresh the page.

Although from this MVP it was clear that I have overestimated my ability and need to change my game plan, I did not do it because, I though that I would have complete more features during the Christmas.

During the Christmas holiday, I did do some progress, e.g.

* Completed the Add/remove groups/members features without the need to refresh the page.

However, most of my time was spent on the [emulator](https://cseegit.essex.ac.uk/ce301_2020/ce301_rai_ajaya/-/blob/master/final_product/technical_documentation/testing.md) for the Firebase Cloud Function, which in the end wasn't even worthwhile because, the Cloud Function wasn't used much as I had anticipated. Furthermore, during the Christmas Holiday, I had to look for jobs and prepare for the interviews.

I had yet again overestimated my ability because I couldn't finish all the features during the Christmas. Because of this you would assume that I would come up with new game plan, but I didn't. WHY?

Only during the [Week 18 meeting](https://cseejira.essex.ac.uk/browse/A301309-73), I realised that I needed to change my game plan because I wasn't making much progress, and this is all thanks to Dr David (my supervisor). From here on, I decided to put my focus on the functionality and not the design because of my slow progression.

And from here on, the features that I prioritised included:

* Chat system
* Search page:
  + Search People
  + Search group -> user able to join the groups shown on the search page
* Leader:
  + Whoever created the group === leader
  + Leader is able to change the bio
  + Leader is able to add/remove members

Till Week 18, I have been focusing on my original plan; not realising that the progression was very slow. So, the changes had to be made in this stage. Changes included:

* Having new features that serve the purpose of my idea which was to giving people tool to create groups (similar to the brotherhood) for their specific needs, e.g. Self-improvement. These features are mention above.
* Focusing on the functionality rather than the design
* Focusing on the prototype rather than trying to complete the task to a company standard

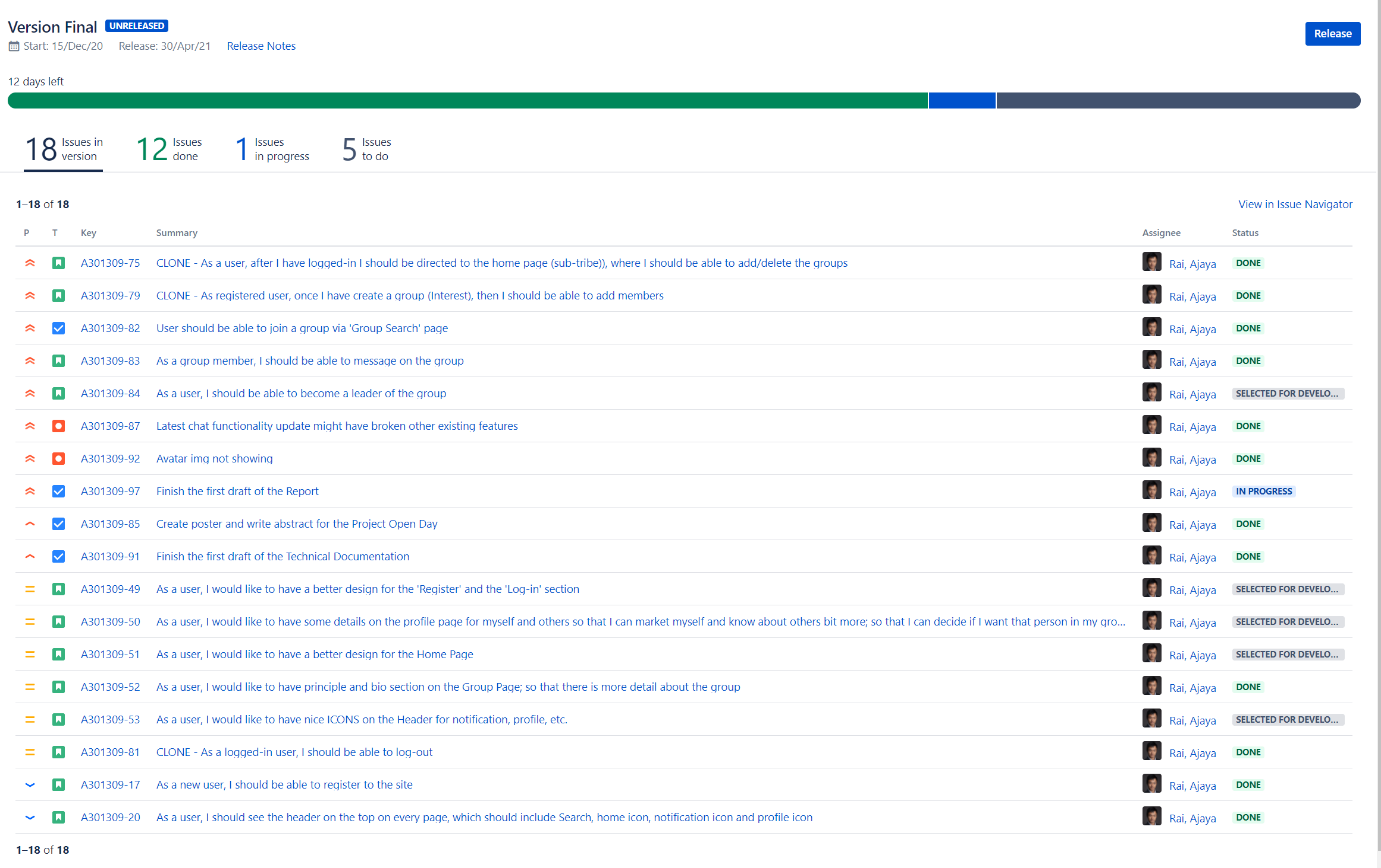
Overall, I was really good at adapting to these changes, because I knew that I had to carry out the changes immediately, otherwise risk of have the Product that is not up to the standard.

From the Week 18 meeting, I learnt that it is essential to be aware of the project's progression, and constantly review the goal set in the beginning (e.g. Why do you want to create this app). This is because without it, appropriate changes would not be made in order to deliver the software that is useful for the client in a given timeline.

On Week 28, I have finished:

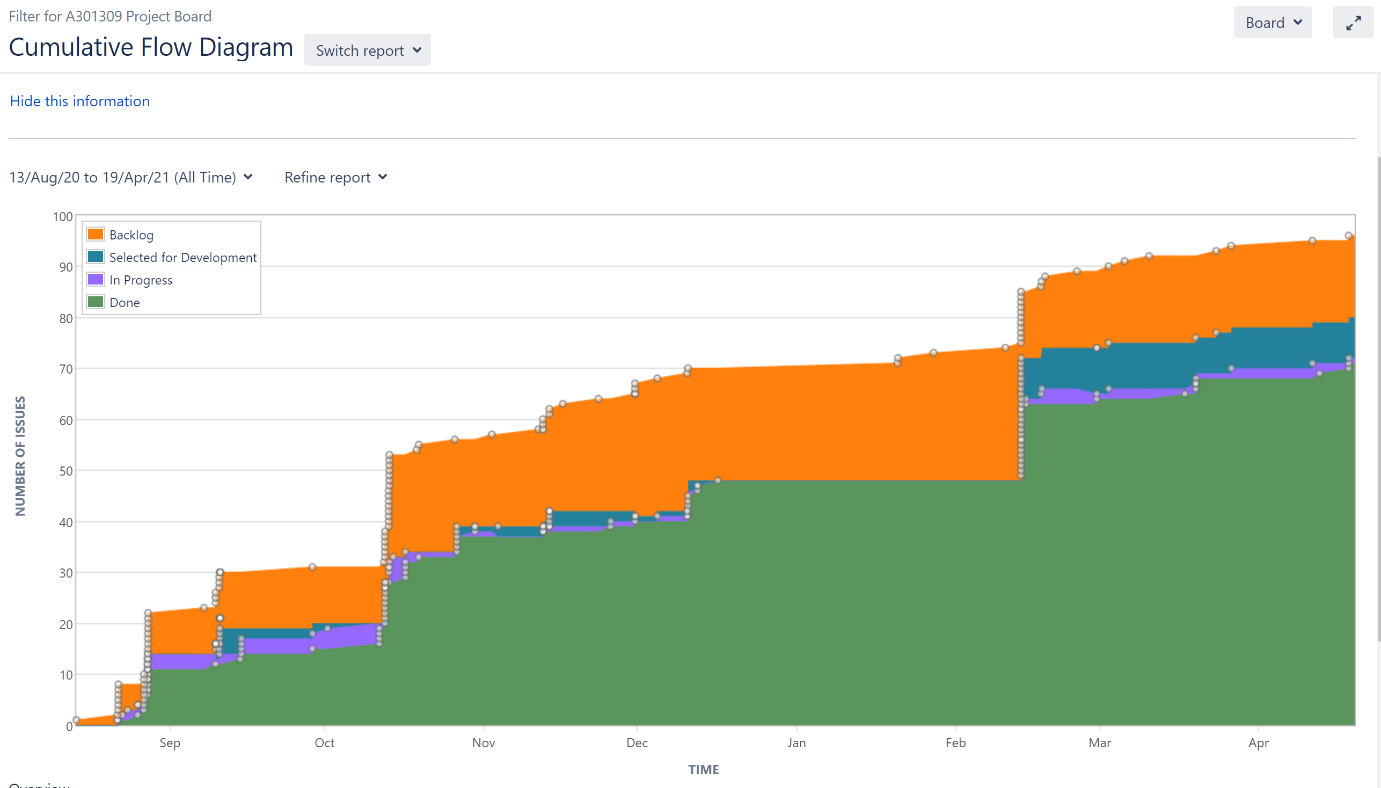
* Chat system [DONE]
* Search page: [DONE]
  + Search People
  + Search group -> user able to join the groups shown on the search page
* Leader:
  + Whoever created the group === leader
  + Leader is able to change the bio
  + Leader is able to add/remove members

For the evidence in Jira please see the image below:

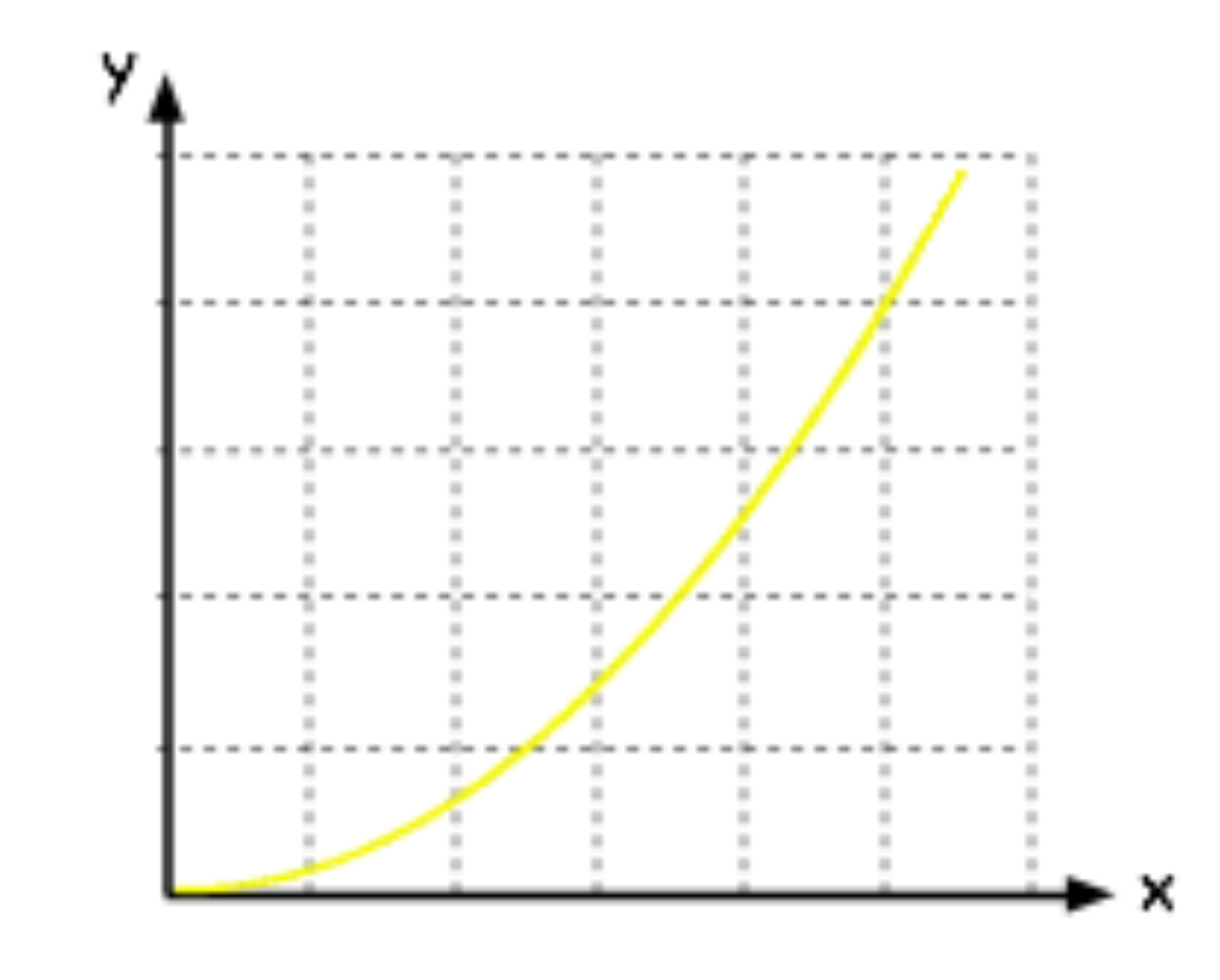


## **Assessing the momentum**

The image below summaries my momentum throughout the academic year:



As you can see above, the momentum has been fairly consistent. I would have assumed that the progress would be something like the diagram below:



Where the progression is slow in the beginning, however, progression is accelerated in at later stage. I would assume that this should have been the case because in the beginning, I am not familiar with the technology and how to solve the problem, but as I start doing it, it should be clearer and clearer; thus making the implementation quicker in the end.

I believe this is the case in terms of coding, however since the Capstone Project required other documentations like Posters, Reports, etc. These lead to the slow progress.

## **Risk management**

3 risks (ordered by their impacts):

1. Choose ReactJS and firebase without prior experience
2. Planning what I will be completing
3. Firebase Function Emulator

Firstly, going with the tools ReactJS and Firebase in the beginning was a risk, because I was not confident with the ReactJS and I had never used Firebase before.

So, I had to research and think that going with these tools was a good idea. To see the research that I have done please look at the "What language to use?" section on the [Background Research](https://cseegit.essex.ac.uk/ce301_2020/ce301_rai_ajaya/-/blob/master/Challenge%20Week/Background%20Research.docx).

There I concluded that most web app similar to mine used ReactJS for the frontend and PHP for the backend.

And I did further research after this on the topic which development stack had the most tutorials and help especially for the project similar to mine. I concluded that ReactJS and the Google Firebase was a popular option. Some of the projects that I found and tried include:

* [Basic Facebook Clone](https://cseegit.essex.ac.uk/ce301_2020/ce301_rai_ajaya/-/tree/master/Summer%20preparation/Projects/fb-clone)
* [Basic Slack Clone](https://cseegit.essex.ac.uk/ce301_2020/ce301_rai_ajaya/-/tree/master/Summer%20preparation/Projects/slack-clone)
* [Social-ape](https://cseegit.essex.ac.uk/ce301_2020/ce301_rai_ajaya/-/tree/master/Challenge%20Week/Project) (basic twitter clone)

Overall, the risk was minimised by researching and testing the technology, before making a final decision.

And this method has minimised the risk of the tools not being able to build the project. I say this because, on 19th April 2021, I confirm that the project that I have made is solely via ReactJS and the Firebase. Furthermore, if there were more time, I believe that the whole idea that I have envisioned for the future can be built by these two technologies.

The second risk was from the planning of what I wanted to complete.

This is a risk because once the plan is completed of what will be delivered, this had to be reported to the client (supervisor). And not meeting what was said to be done will affect the payment from the client in the real scenario, but on here, it is about marks.

How did I tried to minimise the risk of failing to provide the feature that were planned?

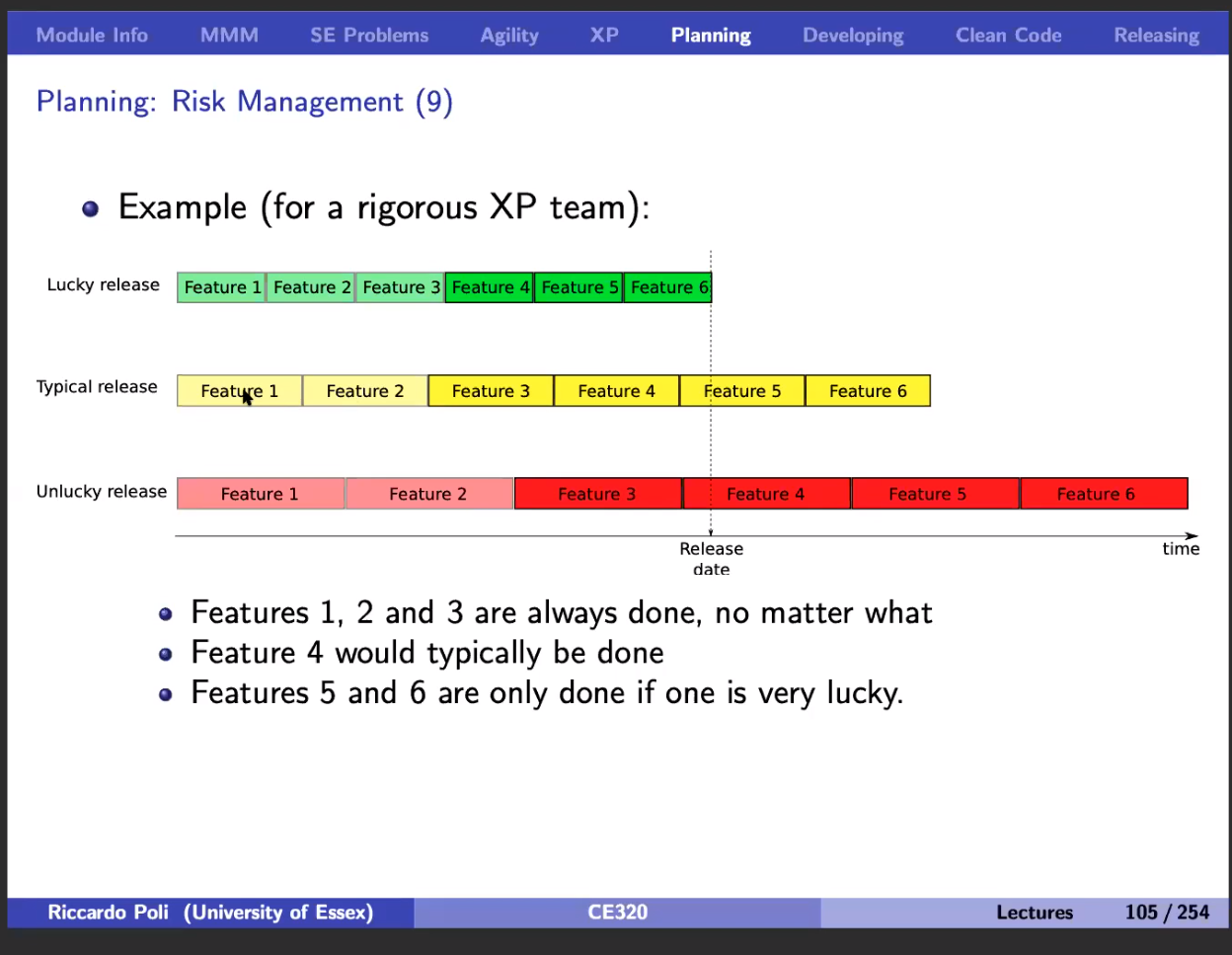
This has been my weakness. I had a great plan in the beginning, I knew exactly what I needed to do and which was a lot of work, and because of this, my initial thought was that I will give my everything, if this lead to me completing my planned project then it's great, otherwise I will just show what I finished.

Since this project required me to have specific requirements and talk to client (supervisor) to conclude what I will be delivering, my prior thought had to be discarded. However, it still influenced the planning of what will be delivered. This is because I was already consumed by the prior idea.

Furthermore, there was no research on how to plan what to be delivered. The way it was done is simply going with the intuition rather than the methodology. And this was a serious mistake, because half way through to my project, it was clear that I won't be able to complete fair amount of what I had planned during the Challenge Week.

How should I have tackled this?

In the beginning, I should have researched on how the respectable companies plan their delivers. One way they do this via method shown below:



[1]

This is a method learnt from "Large Scale Software Systems and Extreme Programming", where estimation is taken into account in the beginning which we call, the Lucky Release and the chance of this happening is only 10%. So, in order to get the 50% chance we have to stretch the feature so we can only guarentee Feature 1, 2, 3, and 4. And if we want to be 90% right on our planning then Unlucky release should be used where Feature 1, 2 and 3 are the ones that are guarenteed.

In conclousion, to make the devlivery plan robust, I should have used the method described above.

Ref:

[1] <https://www1.essex.ac.uk/modules/Default.aspx?coursecode=CE320&level=6&period=AU&campus=CO&year=20>

## **Reflection on overall achievements**

* Personally, disappointed on myself; had high expectation of myself.
* But, did everything that I could and gave all I could. So, this is my ability
* All I could do is give everything, which I did; so no point in getting disappointed
* But, thanks to this I know my ability and able to plan appropriatly regarding it

Personally, I have been disappointed on myself, because I thought I could have done much more; had high expectation. Nonetheless, I have given all the effort that I could give. So, being disappointed is not waste because regardless of the disappointment, I would have achieved the same amount of features completed because I gave everything that I could.

What I could take away from this is that, now I am much more conscious and accurate about my ability and estimate regarding how much I can complete a project.

Furthermore, now I am convinced that ReactJS and Firebase can be used to complete the project for the live release which I am planning to do in the future.

## **Reflection on overall performance.**

* Gave everything,
* What would u have done to improve it?
  + Nothing, cuz I serious did everthing that I could
* All the mistakes had to made which I learnt from -> could have planned, but that wont been efficient?

Overall, I gave everything that I could, so I am satisfied with my performance.

Although, there have been mistakes:

* Planning not being accurate
* Spending a lot of time setting up the Firebase Function Emulator that I did not use much in the end

I would argue that making mistakes was a good path because if I were to plan thoroughly in the beginning before I carried out these tasks, this would have been inefficient because the planning would have taken long; especially when I am trying to plan something that I have not done before.

So, it was better to "Just to it" and learn from the mistakes.

## **Reflection on what learnt**

Consistantly overetimated the work done -> so use the method learn in XP in the future

* reactJS and firebase
* More about the product, start with just for men and the reason was to bring the brotherhood in our society
* Getting side tracked; so always review what u are doing and have someone who checks on me if I am side tracked

For more about what I learnt, please look at the "Context" section in "Main Text", there I mentioned about changing my product's aim sightly from only focusing on men to allowing everyone to access the software, and from having the software to bring the brotherhood to the society; to giving tools to the society where individuals could use for their advantage, e.g. Self-improvement.

## **Did the Agile methodology prove suitable?**

* Yes, gave me chance to try out instantly without finishing the plan where I learnt the most
* Evolution felt natural, e.g. product changing focus from only men and reason to bring bhood in society

Agile methodology was appropriate for this project, because as you can see above, there have been many evolution of the project, including:

* Update of the features after Week 18
* Change of focus from targeting the web app to only men to all the people who are interested + Rather than using the software to build the brotherhood, focus is placed on giving people tool for building groups where they can utilise for individual needs, e.g. Self-improvement.

Without the Agile's focus on client interactions (meeting with supervisor), I would not have able to detect/evolve on the two points mentioned above. Furthermore, due to the Agile's flexibility, I was able to take actions on the new findings.

Whereas regarding waterfall methodology, it would have encouraged me to plan first, then implement. The waterfall is more rigid in terms of not being able to review the original idea and evolve.

Furthermore, Agile have many short iterations of planning and implementation, and this is a huge advantage because its impossible to cover everything in the planning which you can quickly identify while implementing. For example, since I was not familiar with the ReactJS and Firebase, implementing early helped me to see the potential of the tools and how I could use it for my project.

# **Conclusion**

In conclusion, please see below for what I planned in the beginning and what I was able to deliver, along with what I am planning to do in the future.

Planned:

* User being able to register
* User being able to login
* User being able to add/remove groups
* User being able to add/remove members to the group
* Having a leader functionality where only the leader can add members to the group
* Search page where the user can see the people close to him/her + see other group
  + User can click on the people to see their profile
  + User can click on the group to see its details
* User can logout

Delivered:

* User being able to register
* User being able to login
* User being able to add/remove groups [buggy]
* User being able to add/remove members to the group
* Search page where the user can see the people close to him/her + see other group [partially done]
  + User can click on the people to see their profile
  + User can click on the group to see its details
* User can logout
* Chat System

Future plan:

* Design, it should highly the brotherhood concept
* Search page where the user can see the people close to him/her + see other group
  + User can click on the people to see their profile
  + User can click on the group to see its details
* Change the name of the app
* Group leader functionality, where the leader has power to add/remove people from the group. And the leader can be chosen via voting system
* Calendar functionality where group leader can plan the activities
* Users will have like and dislike value
* Members can trigger notification to all of the group members if they suddenly wanted to invite people. E.g. Lets say suddenly Sam felt like playing a basketball with the group, he can just use the app to notify others

# **Acknowledgements**

# **Reference**