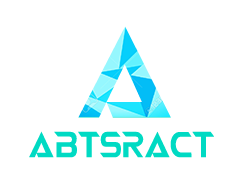
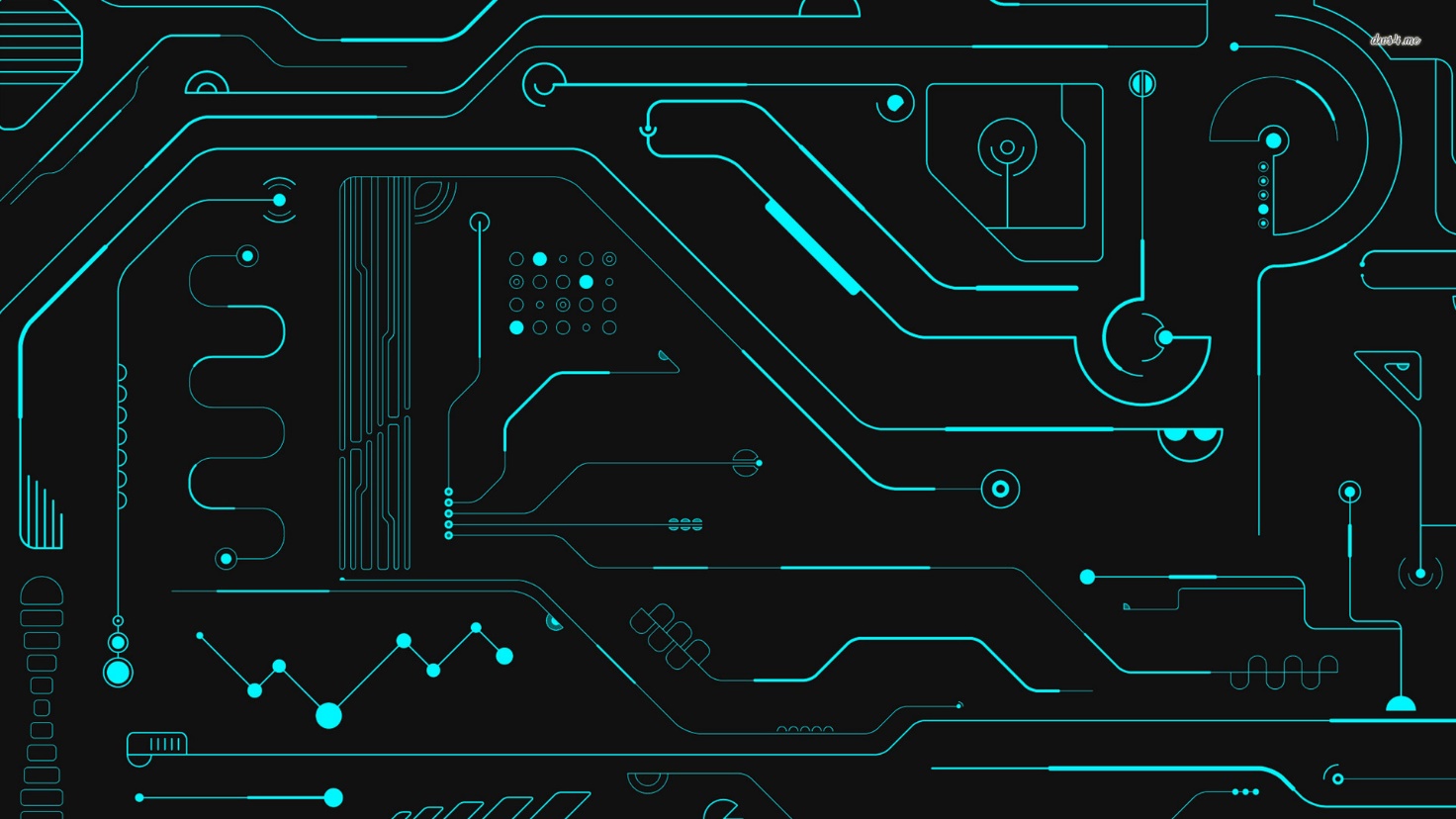
2017/08/01



324 ITRW

Progress Report:

Blockchain

Group 9:

1. Brendan Amato 26426358
2. Keagan du Toit 26117967
3. Theunnis Janse van Rensburg 25911864
4. Dahee (Joy) Kim 25920057
5. Vorster Naudé 26192349
6. Zander van Vuuren 24678317
7. Philip Venter 24674672

Table of Content

[Table of Content 1](#_Toc492638899)

[Table of Figure 2](#_Toc492638900)

[1. Business Introduction 1](#_Toc492638901)

[2. Website and Mobile Application Implementation 1](#_Toc492638902)

[2.1 Mobile Application 1](#_Toc492638903)

[2.2 Web Application 1](#_Toc492638904)

[3. Cloud Computing in the Solution 2](#_Toc492638905)

[4. Blockchain Implementation 3](#_Toc492638906)

[5. Description of Prototype 3](#_Toc492638907)

[5.1 Website. (Heavily incomplete) 3](#_Toc492638908)

[ Login: 3](#_Toc492638909)

[ Registration page: 4](#_Toc492638910)

[ Homepage (incomplete): 5](#_Toc492638911)

[5.2 Mobile App (Incomplete): 6](#_Toc492638912)

[ Login: 6](#_Toc492638913)

[ Register: 7](#_Toc492638914)

[6. Current Progress to Date 8](#_Toc492638915)

[6.1 Features that need to be completed 8](#_Toc492638916)

[ Feature Success Values 8](#_Toc492638917)

[ Features that need to be completed 9](#_Toc492638918)

[ Prioritization Values 9](#_Toc492638919)

[ Feature Dependencies 10](#_Toc492638920)

[ Participating Group Members 11](#_Toc492638921)

[7. Description of progress 12](#_Toc492638922)

[7.1 On-track in terms of scope and time 12](#_Toc492638923)

[7.2 Plans to stay on track 12](#_Toc492638924)

[8. Conclusion 12](#_Toc492638925)

Table of Figure

[Figure 1: Website - Login 3](#_Toc492638926)

[Figure 2: Website - Registration part 1 4](#_Toc492638927)

[Figure 3: Website - Registration part 2 4](#_Toc492638928)

[Figure 4: Website - Homepage 5](#_Toc492638929)

[Figure 5: Mobile - Login 6](#_Toc492638930)

[Figure 6: Mobile - Registration part 2 7](file:///C:\Users\삼성\Desktop\ITRW%20324%20-%20Progress%20Report.docx#_Toc492638931)

[Figure 7: Mobile - Registration part 1 7](file:///C:\Users\삼성\Desktop\ITRW%20324%20-%20Progress%20Report.docx#_Toc492638932)

[Figure 8: Burndown Chart 10](#_Toc492638933)

[Figure 9: Feature - Dependencies 10](#_Toc492638934)

[Figure 10: Feature - Group Participations 11](#_Toc492638935)

1. Business Introduction

One of the problems that occurs often in the digital media industry is copyright infringements. To extend this point, a few problems will be discussed. Many artists, whether it be: music, Digi-art, videos, etc. often have problems regarding this issue - especially for the amateur artists (those who have not many experience in this industry). Artists often find their work “stolen” - in a sense that they either find their work illegally distributed which leads to loss the possible profits; or find them being used by other people without their consent/permissions (e.g. remaking a piece a music without paying the copyright fees towards the owner). Other problems also include transaction issues, such as having scammers plotting schemes to get an income in various methods.

To combat these issues, Abstract has come up with a blockchain implemented technology. The solution is to allow artists (targeting on amateurs, hobbyists, and non-professional individuals) to sell their art (music or images) online, at the same time keep track of their product. In this manner, not only will the artists have control in who is illegally distributing their product, but also calculate the number of potential clients by using RIP (Registered Intellectual Property). RIP aims to assign each individual digital asset with a unique crypto-code. In doing so, Abstract can track the distribution of the digital media and alert the registered authors thereof. By using blockchain technology, authors can also have a rested assurance in the fact that they cannot be scammed financially as blockchain technologies are known to be safe and secure due to distributed processing.

1. Website and Mobile Application Implementation

The website and mobile application will be used to interface with the data contained on the EC2 AWS Windows Server 2016 server. These two components of the system will however not have the exact same functionality due to the nature of the platforms they run on.

# Mobile Application

The mobile application will be targeted solely for devices running Android API level 21 and later. The reason for this is that we want to use Google’s material design guidelines to make our application look and feel like a native Android application.

1. The functions that will be included in the mobile application are as follow:
2. The user will be able to register a new account when using the application for the first time.
3. The user will be able to login to his/her account when an account already exists.
4. The user will be able to stream music that is already owned.
5. The user will be able to view his/her profile and update certain personal information fields.

The user will not be able to buy new music because storing credit card information on a mobile phone is considered as a security flaw and also adds an immense number of programming to the application. At this stage it is not deemed feasible or necessary to include a payment option.

# Web Application

The website aims to provide a slightly more functionality than the native mobile application. It will also provide desktop users with an application that can run in all browsers, preferably not in IE, and connect seamlessly with our AWS instances. The website will also automatically scale to fit all screen sizes that connect to it using bootstrap.

The functionality provided in the website will include all the functionality of the mobile application as well as:

1. Viewing music that is not yet owned by the current user.
2. Previewing unowned music.
3. “Purchasing” new music. (Acquiring might be a better description due to the lack of currency exchange)
4. Downloading owned music. (The transaction hash will be included in the file).
5. Uploading new content to the website where other users can listen and buy it.

The website is developed using a framework called Angular 4. This intends that the website will consist of a single index.html page that will be updated by changing the components on this page. This will give the user an exceptionally fast web application and will minimize the amount of data transferred between client and server.

1. Cloud Computing in the Solution

The group plans to use 2 AWS servers, one to host the website, and another to host the web service and our database. AWS (Amazon Web Service) provides a free account with 750 hours of server time per month, seeing as months with 31 days are equal to 744 hours, this is more than enough, and if the group uses 2 AWS accounts we can host the website from one and the web service/MySQL database from the other. It has been decided on placing the database on the same server as the web service, to reduce latency - in order to avoid having another server to get access to the database. The web service can then be pinged by any device, no matter where they are. The major disadvantages of this, is the response time and size of the server. By using www.cloudping.info, we can get a rough estimate of the ping (latency) to each AWS server, this gives a better idea of which server to use, with the purpose to get the latency at a minimum.

The free servers have only 1GB of extremely high speed RAM, 1 processing core and 30GB of storage (when using windows server this already accounts to about ~14GB, so that leaves 16GB usable per server). The recommended operating systems available on AWS are as follows: Amazon Linux AMI, SUSE Linux Enterprise Server, Red hat Enterprise 7.4, Ubuntu Server 16.04, and Windows Server 2016. The group is using Windows server 2016, because the majority of the group members are keener to Windows than to Linux. Thus we have decided to will use the Windows platform, and connect by using a remote desktop connection. The region we decided to use is Ireland (EU West), as the average ping we could detect from www.cloudping.info is in the range of 190-220 MS because America and Southeast Asia are much higher, and London and Frankfurt (other EU West servers) are about the same.

Currently we have already implemented the first part that is hosting the web service and MySQL database on the server. The way we did this is by installing a server version of MySQL and connecting to the server with our development machines (You first have to open the correct ports on AWS’s EC2 workbench for this, as well as allowing the port through the server’s firewall), and installing node (to run our node.js web service), then finally downloading the web service files from GitHub, opening the correct ports on AWS and the server running the firewall.

Thus in conclusion, we will make use of a Windows server 2016 AWS instance located in the Ireland server. We believe this is the best combination for our project, and will suit us best within the scope of our project and be the best for our group to implement.

1. Blockchain Implementation

The group will be making use of a framework called Truffle, this was suggested to us by our Demi as it has an easy to follow tutorial and user-friendliness. This is a framework for “Ethereum”, which not only allows us to develop decentralized applications, but also deploy them.

Our service will make use of blockchain technology to store and keep track of the user's’ authorised songs. Our initial plan was to store the actual songs on a blockchain but realised that server capacity will quickly become an issue. By keeping track of who own the specific ‘copy’ of the song,  we can track who owns the song and if a song is released on websites (such as pirate bay and other torrent websites), it will be able to track to where and who the owner of the song is. In this manner, the group can see who the original leak of the song was. By using blockchain, the group is able to assure that each copy is original and linked to a specific user, making it easier to restore lost data, as it ultimately assists the artist from losing income due to piracy issues. Our current implementation only includes music but it is scalable to all type of Intellectual property or digital assets.

1. Description of Prototype

# Website. (Heavily incomplete)

## Login:

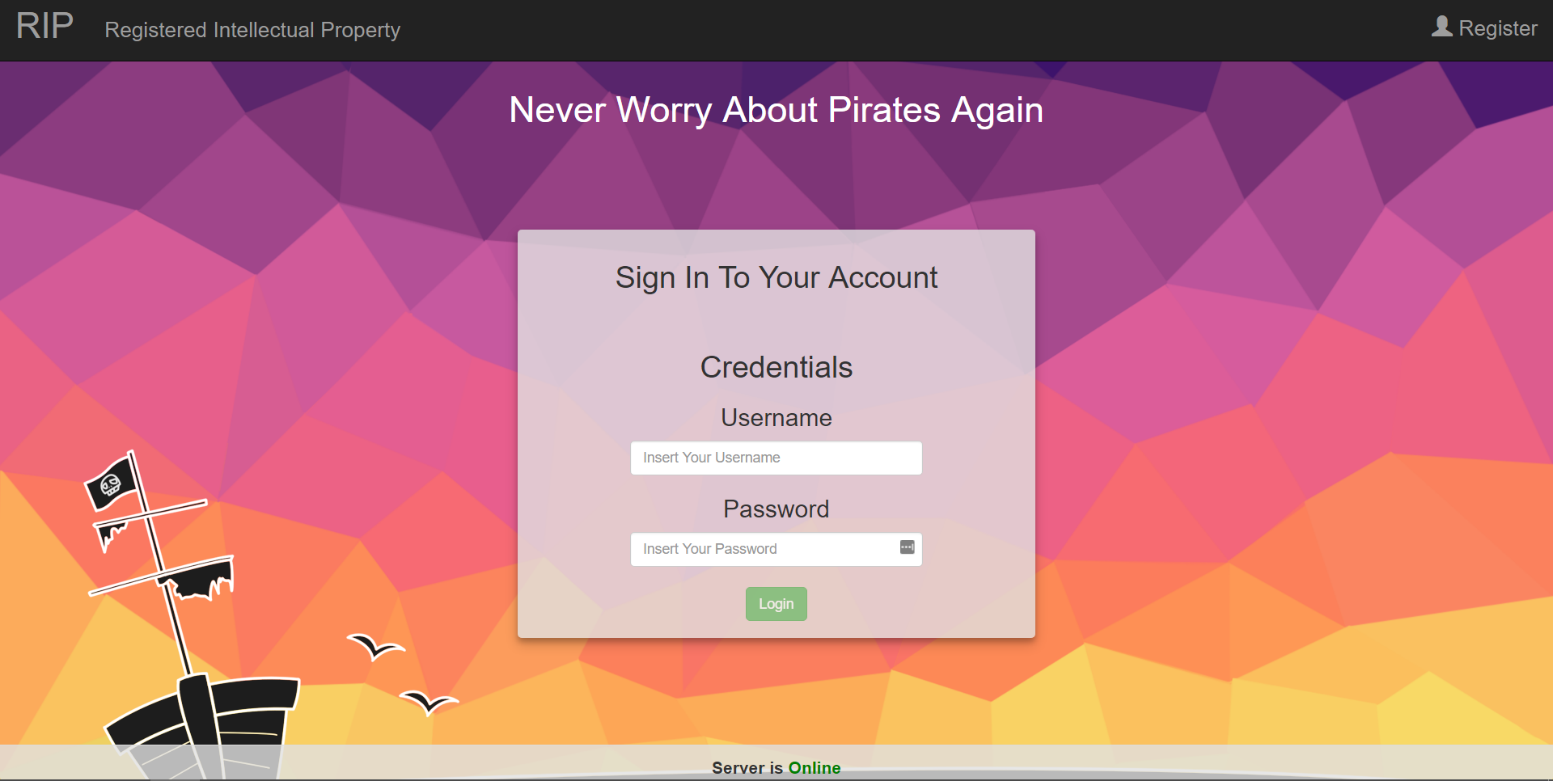
****

Figure 1: Website - Login

* Used for signing into your user account.
* Navigation to password recovery.
* Navigation to create an account page/registration.

## Registration page:

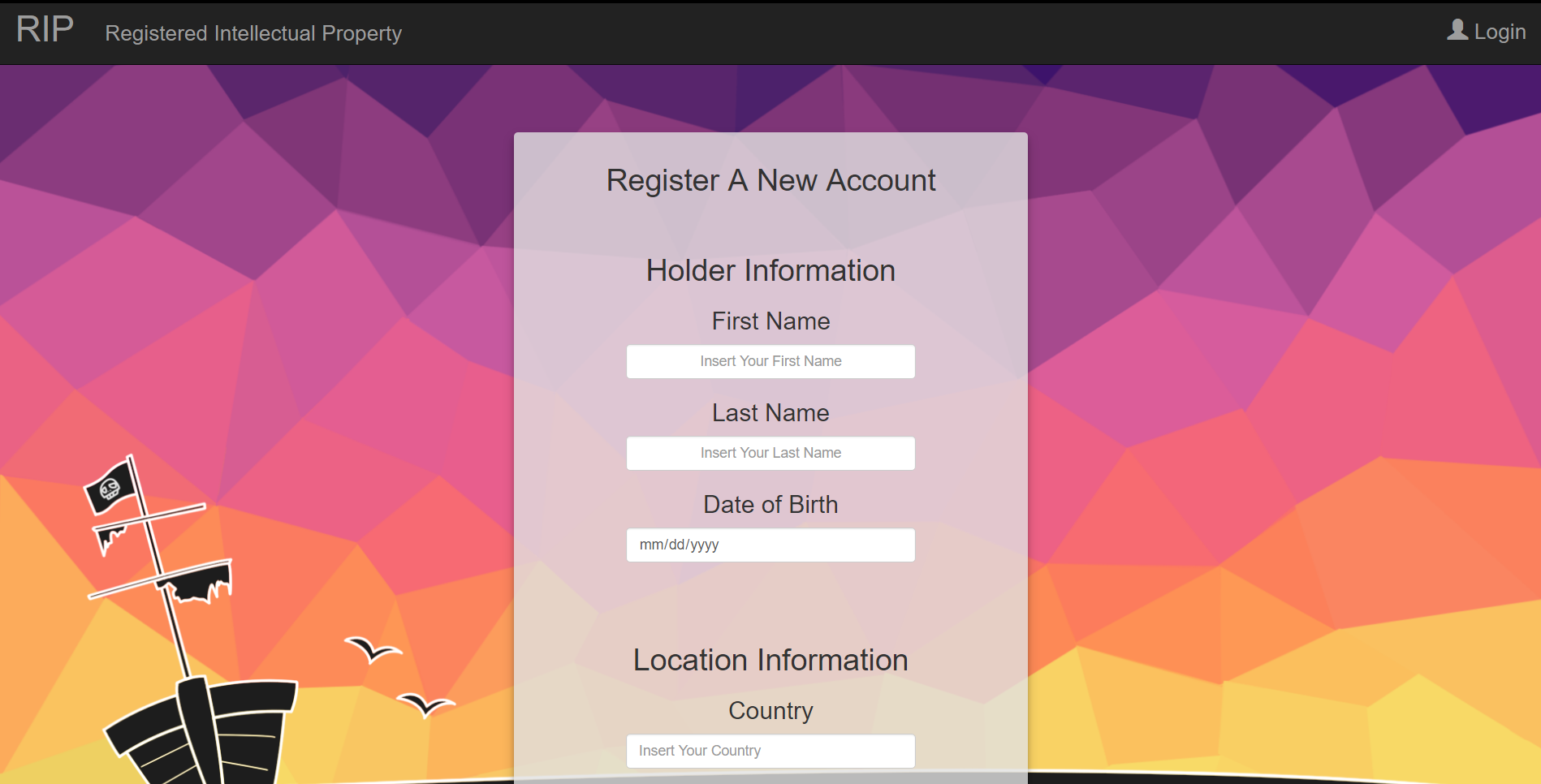


Figure 2: Website - Registration part 1

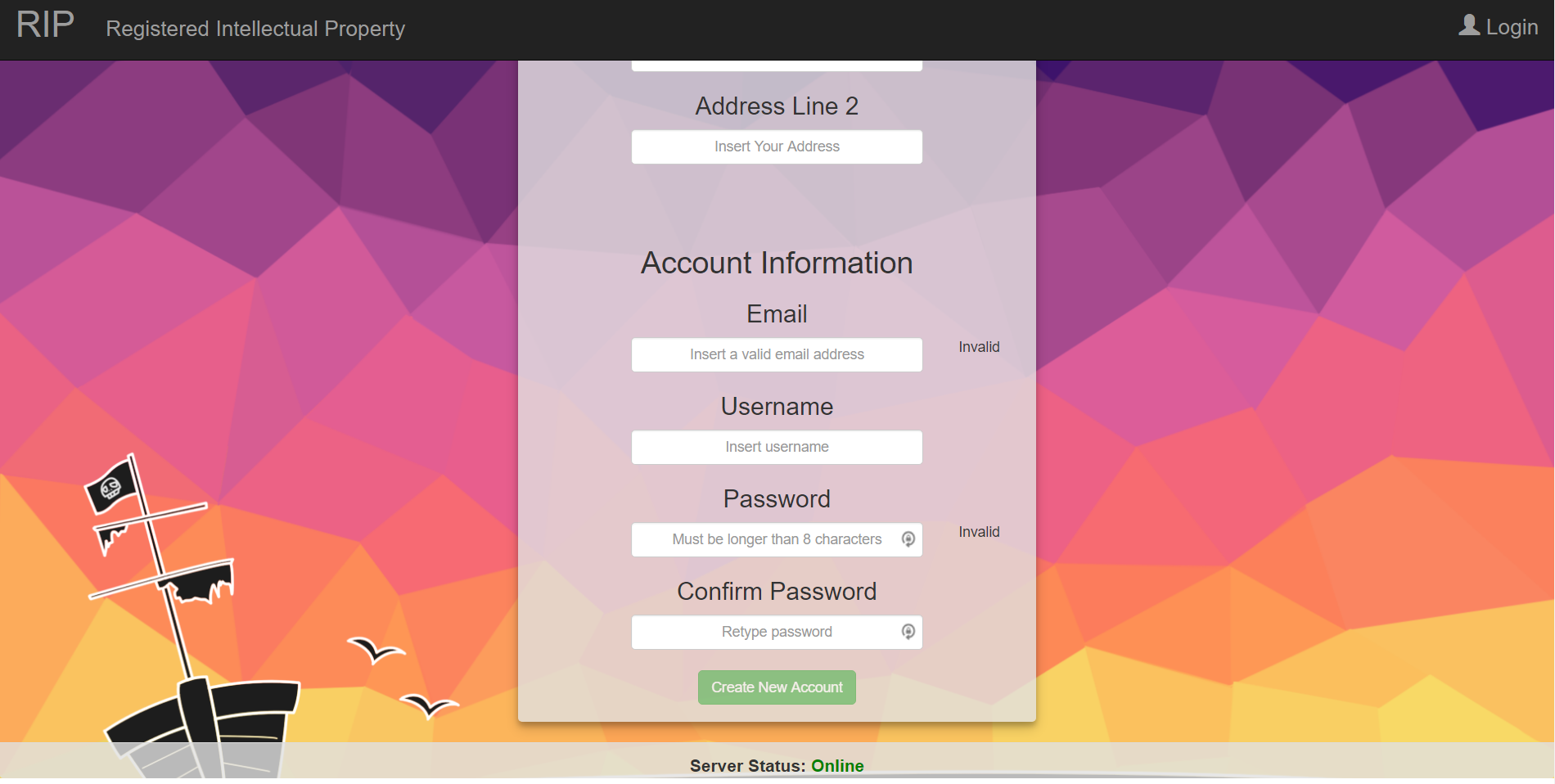


Figure 3: Website - Registration part 2

* Used to register new user accounts.
* Full data validation and verification.

## Homepage (incomplete):

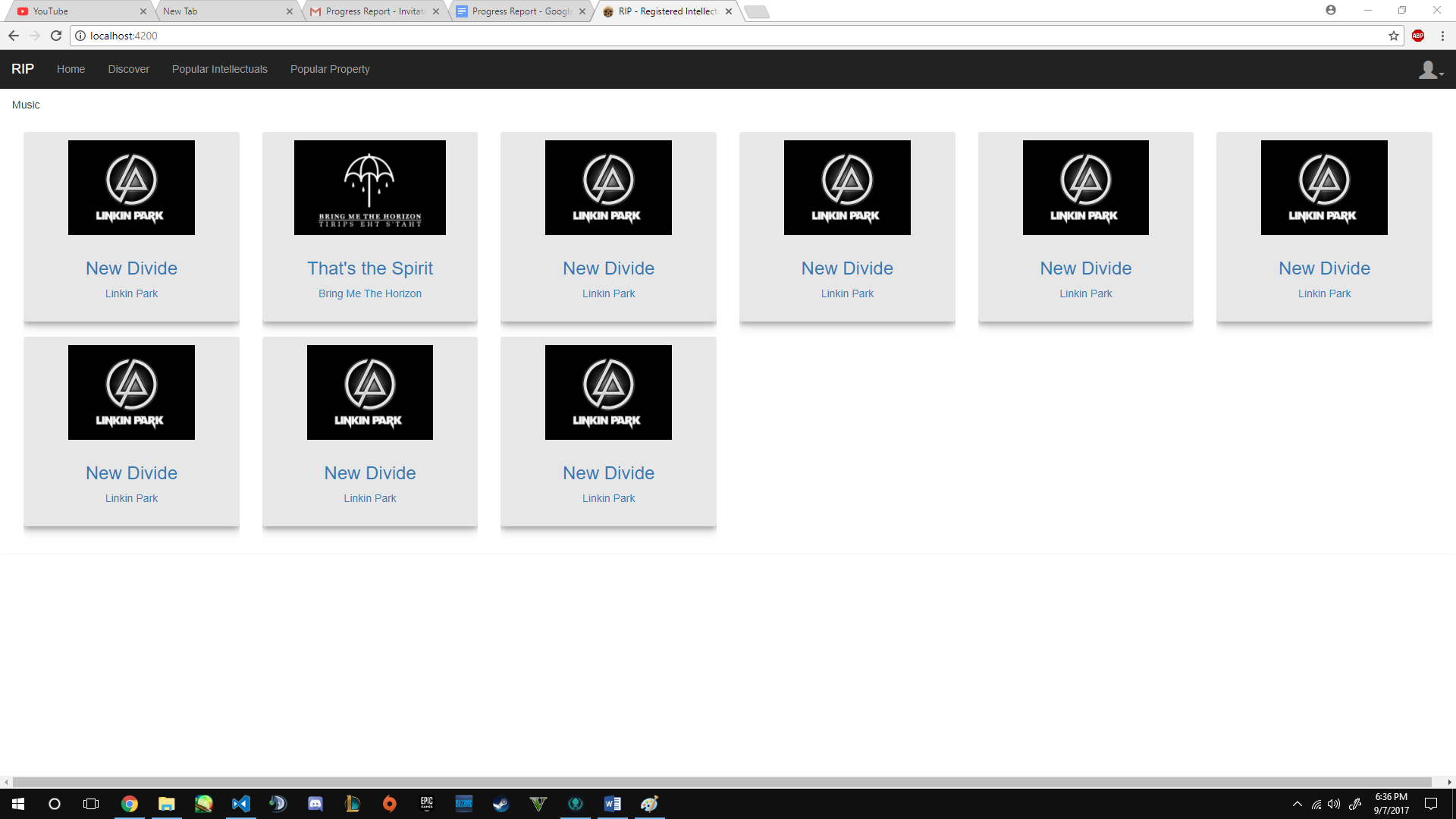


Figure 4: Website - Homepage

* User’s welcoming page it will support a library of all the users owned intellectual property.
* Will display all of the authors own intellectual property.
* Allows for easy navigation to all other components: discover, popular, search bar for easily finding what you are interested in, easy access uploads for the author, navigation to your user profile, login and log out, etc.

# Mobile App (Incomplete):

## Login:

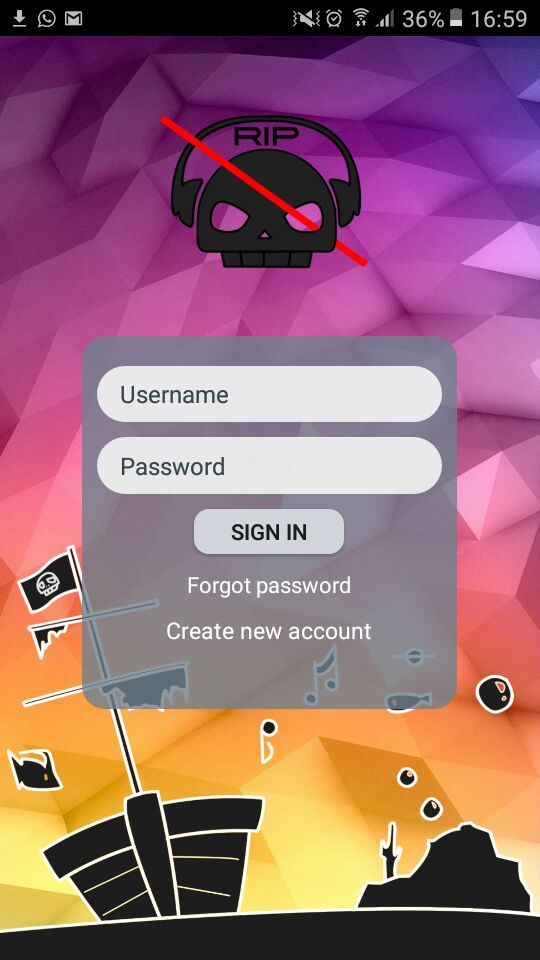
****

Figure 5: Mobile - Login

* Used for signing into your user account.
* Navigation to password recovery.
* Navigation to create an account page/registration.

## Register:

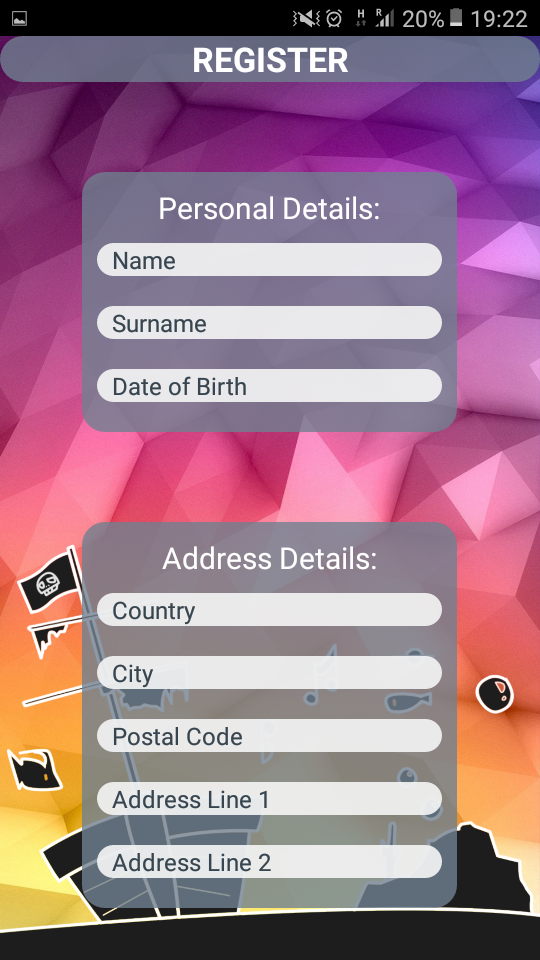
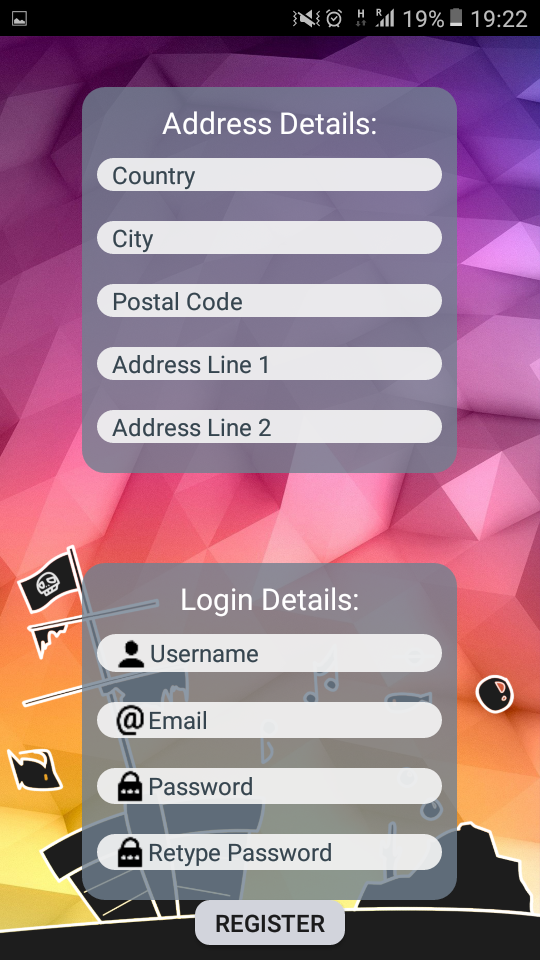
****

Figure 6: Mobile - Registration part 2

Figure 7: Mobile - Registration part 1

* Used to register new user accounts.
* Full data validation and verification.

The groups will be working up on polishing the backend and technologies needed for supporting the system. Once everything there is in order we can work on bringing the web and mobile apps to life. Currently the app is not yet working as a whole but there are components working and we will work on completing them soon.

1. Current Progress to Date

# Features that need to be completed

* Streaming owned music from the website by communicating with the web service.
* Registering the user on the database.
* Being able to purchase music from the website.
* Modify and view profile information.
* Adding a user friendly Graphical User Interface.
* Creating playlists of all your owned music.
* Implementing blockchain technologies with the online music.

## Feature Success Values

|  |  |
| --- | --- |
| **Feature** | **Success Value** |
| Streaming owned music from the mobile application and website by communicating with the web service. | **4** |
| Registering the user on the database. | **5** |
| Being able to purchase music from the website. | **5** |
| Modify and view profile information. | **3** |
| Creating playlists of all your owned music. | **1** |
| Adding a user friendly Graphical User Interface. | **3** |
| Implementing blockchain with the online music. | **5** |

Table 1: Feature - Success Value

## Features that need to be completed

|  |  |
| --- | --- |
| **Feature** | **Time/Difficulty Value** |
| Streaming owned music from the mobile application and website by communicating with the web service. | **5** |
| Registering the user on the database. | **8** |
| Being able to purchase music from the mobile application and website. | **7** |
| Modify and view profile information. | **6** |
| Creating playlists of all your owned music. | **5** |
| Adding a user friendly Graphical User Interface | **3** |
| Implementing blockchain with the online music. | **10** |

Table 2: Feature - Need to be completed

## Prioritization Values

|  |  |
| --- | --- |
| **Feature** | **Prioritization Value** |
| Implementing blockchain with the online music. | **50** |
| Registering the user on the database. | **40** |
| Being able to purchase music from the website. | **35** |
| Streaming owned music from the mobile application and website by communicating with the web service. | **20** |
| Modify and view profile information. | **18** |
| Adding a user friendly Graphical User Interface. | **9** |
| Creating playlists of all your owned music. | **5** |

Table 3: Feature - Prioritization Values

A burndown chart provided bellow shows the current state of our actual progress (blue) in comparison to our scheduled progress (red).

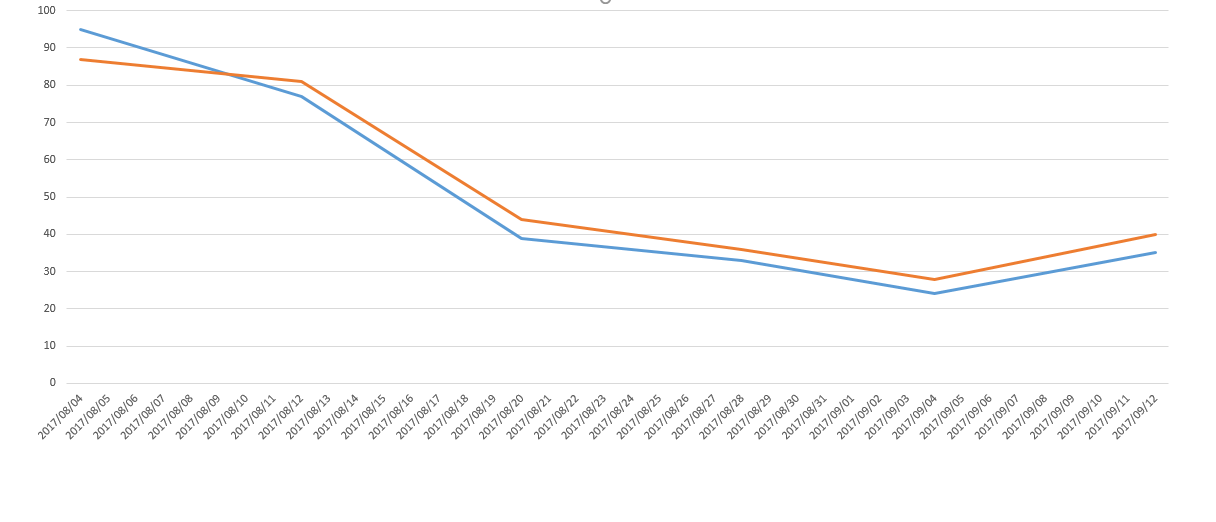
****

Figure 8: Burndown Chart

## Feature Dependencies

|  |  |
| --- | --- |
| **Feature 1** | **Feature 2** |
| Registering the user on the database. | Modify and view profile information. |
| Registering the user on the database. | Streaming owned music from the mobile application and website by communicating with the web service. |
| Registering the user on the database. | Being able to purchase music from the mobile application and website. |
| Registering the user on the database. | Creating playlists of all your owned music. |

Figure 9: Feature - Dependencies

## Participating Group Members

|  |  |
| --- | --- |
| **Feature** | **Group Member** |
| Implementing block chaining with the online music. | Keagan, Joy and Philip |
| Registering the user on the database. | Philip and Vorster |
| Being able to purchase music from the mobile application and website. | Brendan, Philip, Vorster and Zander |
| Streaming owned music from the mobile application and website by communicating with the web service. | Brendan, Keagan, Joy, Philip, Theunnis, Vorster and Zander |
| Modify and view profile information. | Brendan, Theunnis Philip and Vorster |
| Adding a user friendly Graphical User Interface. | Brendan, Joy, Theunnis, Vorster and Zander |
| Creating playlists of all your owned music. | Keagan, Philip and Vorster |

Figure 10: Feature - Group Participations

1. Description of progress

# On-track in terms of scope and time

Currently, the group project is not on track on scope and time. We have not completed streaming music features, purchase controls, profile information modification, creation of playlists, and blockchain implementation. However, the group has made significant progress throughout the past few days as group members are completing their AWS courses, the members are communicating and assisting each other to make efficient progress, and all “backbone” elements have been/ is being worked on till this very moment.

# Plans to stay on track

Nonetheless, the group has already implemented a solution to combat this issue. The group has decided to have consistent “Discord” meetings on a regular basis (at least twice a week) to ensure that the group checks on each other’s progresses as well as assist each other in “stuck” places. The use of “Discord” allows us to work at home with all software preinstalled computers saving us the time and frustration from reinstalling all of them on campus computers. By doing so, the group has already improved on time management, and have a more efficient work progress. We have also adjusted our time schedule taking tests and assignments from other modules into consideration and distribute the work amongst more members so that we can ensure a faster work progress.

1. Conclusion

In conclusion, the features needed has been discussed which includes: streaming owned music from the mobile application and website by communicating with the web service, registering the user on the database, being able to purchase music from the website, modify and view profile information, creating playlists of all your owned music, adding a user friendly Graphical User Interface, and implementing blockchain with the online music. In addition, the group has decided on all decisions such as the server of choice, or how to implement the database, etc. The group is currently working heavily (focusing on the project with high priority) on the project as a whole, and is making significant progress. Although the group may be slightly off schedule, we have adjusted out schedules taking tests and assignments from other modules into consideration, and is having consistent discord meetings to not only assist each other, but also to motivate each other to work faster and more efficiently.