



## CSCI321 – Project

Project Topic: Recommender System based on collaborative filtering

Topic Code: CSIT-23-S1-06

## Project Requirements

## Project Team

Group: FYP-23-S1-14		
Student ID:	Name:	Email:
7222828	Loo Joon Wee	<a href="mailto:jwloo002@mymail.sim.edu.sg">jwloo002@mymail.sim.edu.sg</a>
7372772	Low Rui Hao	<a href="mailto:rhlrlow001@mymail.sim.edu.sg">rhlrlow001@mymail.sim.edu.sg</a>
7436208	Muhamad Amir Akmal BMS	<a href="mailto:mohammad103@mymail.sim.edu.sg">mohammad103@mymail.sim.edu.sg</a>
7224527	Tham Jia Xuan Travis	<a href="mailto:jxttham001@mymail.sim.edu.sg">jxttham001@mymail.sim.edu.sg</a>
7222439	Wee Zee En Leonard	<a href="mailto:zelwee001@mymail.sim.edu.sg">zelwee001@mymail.sim.edu.sg</a>
Supervisor:	Mr Terence Chew	<a href="mailto:tchew@uow.edu.au">tchew@uow.edu.au</a>
Accessor:	Dr Prem	<a href="mailto:premrajan_p@yahoo.com">premrajan_p@yahoo.com</a>

## Document Control

### Title: Project Requirements

Owner	Current Version	Last Change on		Approved By
		Date	Time	
FYP-23-S1-14	1.1	30/1/2022	2.00pm	All
FYP-23-S1-14	1.2	5/2/2023	10.30pm	All
FYP-23-S1-14	1.3	10/02/23	10pm	All

Record of versions:

Revision Date	Description	Section Affected	Changes Made By	Version after Revision
30/1/2023	Starting of outline and inputting of content	Document Control, Introduction	Joon Wee	1.0
5/2/2023	Finishing of introduction, inputting content for glossary and some overall description. Adding functional, non functional and interface requirements	Introduction, Glossary, Overall Description, Functional, Non-functional, Interface Requirements	Joon Wee, Amir	1.1
10/02/23	Edit introduction purpose Adding Business model Edit and finish up functional, non-functional and interface requirements	Introduction, Business Model Functional, Non-functional, Interface Requirements	Joon Wee Amir	1.2
14/02/23	Refining business model Refining overall description	Business Model Overall Description	Amir	1.3
19/02/23	Start working on creating and updating researches and comparisons for	Researches and Comparisons	Joon Wee	1.4

## Project Requirements

	web crawl and collaborative filtering algorithm			
26/02/23	<p>Start working on adding Programming languages and database comparison</p> <p>Add and update meeting minutes</p>	<p>Researches and Comparisons</p> <p>Meeting Minutes</p>	Joon Wee	1.5
12/03/23	<p>Start working on adding individual pros and cons of choosing a specific language and database.</p> <p>Start working on adding researches and comparisons for hosting platforms</p> <p>Add and update meeting minutes</p>	<p>Researches and Comparisons</p> <p>Meeting Minutes</p>	Joon Wee	1.6
20/03/23	<p>Add and update wireframes</p> <p>Add and update meeting minutes</p>	<p>Wireframe</p> <p>Meeting Minutes</p>	Joon Wee	1.7
24/03/23	<p>Updating wireframes</p> <p>Update version control</p>	<p>Document Control</p> <p>Researches and Comparison</p>	Joon Wee	1.8

## Project Requirements

	<p>Refine whole report</p> <p>Add and update meeting minutes</p> <p>Added and updated Sequence diagrams</p>	<p>Requirements</p> <p>Wireframe</p> <p>Meeting Minutes</p> <p>Sequence Diagrams</p>	Amir	
25/03/23	<p>Updated Sequence diagrams</p> <p>Updated Use case</p> <p>Adding of gannt chart for WBS</p>	<p>Sequence diagrams</p> <p>Use Case (functional Requirement table)</p> <p>Gannt Chart</p>	Amir Joon Wee	1.9
8/04/23	<p>Updated feature mind map</p> <p>Added meeting minute 11</p>	<p>Requirements</p> <p>Meeting minutes</p>	Joon Wee	2.0

## Table of Contents

<b>Project Team</b>	<b>2</b>
<b>Document Control</b>	<b>3</b>
Record of versions:	4
<b>Table of Contents</b>	<b>7</b>
<b>Introduction</b>	<b>10</b>
Purpose	10
Project scope	10
Document target audience and objectives	10
<b>Glossary</b>	<b>11</b>
<b>Business Model</b>	<b>12</b>
Target Audience	12
Chosen Model	12
Proposed Price Plan	12
<b>Overall Description</b>	<b>13</b>
Product Direction	13
Product Features	13
Comparison between SaaS Recommender Systems	14
Types of users and features	20
Operating Environment	21
Design and Development constraints	21
Assumptions and dependencies	21
<b>Researches and Comparisons</b>	<b>22</b>
Web Crawling	22
References:	30
Collaborative Filtering Algorithms	31
SVD:	31
SVD++:	33
RandomForestRegressor:	35
Summary:	37
References:	38
Programming Languages	39
Analysis of Python vs R	39
Python	39

## Project Requirements

R	41
Summary	42
References:	42
Python vs PHP vs Java vs C#	43
Database comparison	44
Individual Pros and Cons of language and database chosen	45
Hosting Platforms	46
Hosting platform comparison	46
<b>Requirements</b>	<b>47</b>
Product Features Mind Map	47
Functional Requirements	48
Administrator Use Cases	53
Organization Use Cases	64
Individual Use Cases	81
Sequence Diagrams	92
Admin	92
Organization	94
Individual	96
Non-Functional Requirements	98
Performance	99
Accuracy	99
Reliability	99
Recommender system can perform normally even after prolonged usage (up to 1 hour).	99
Security	99
Usability	99
Maintainability	99
The mean time to restore the system (MTTR) in the event of a system failure will not be greater than an hour.	99
Availability	99
Compatibility	99
Portability	99
Scalability	100
Interface Requirements	101
User Interface	101

## Project Requirements

Software Interface	101
Communications Interface	101
Hardware Interface	101
<b>Wireframe</b>	<b>102</b>
Admin	102
Organisation	108
Individual	118
<b>Gannt Chart (Work Breakdown Structure)</b>	<b>125</b>
<b>Meeting Minutes</b>	<b>127</b>
Meeting minute 1	127
Meeting minute 2	128
Meeting minute 3	130
Meeting minute 4	132
Meeting minute 5	134
Meeting minute 6	136
Meeting minute 7	138
Meeting minute 8	139
Meeting minute 9	140
Meeting minute 10	141
Meeting minute 11	142
Meeting minute 12	143
Meeting minute 13	144

## Introduction

RECS is a website-based product where it offers our clients the ability to predict existing user's rating and preferences on different items as well as providing recommendations on similar or other products that the specific user might be interested in based on the data set input by the owner. Our client can add new products to the recommender system that they have not added before into their e-commerce site to generate a predicted rating for the products and make more informed decisions based on the result of the predictions returned. While for our customer, the system allows them to predict and get recommendations of similar items on a single item from the data we web crawl in other websites.

## Purpose

The purpose of this project is to research, study, understand and utilise the concept and technology used in recommender systems from different websites, so that we can develop a product to predict and recommend products for users on a smaller scale with slightly different features. We aim to develop it through researching and utilising collaborative filtering, web crawling, website development, python coding, and algorithms on predictive systems.

## Project scope

To design and develop a product to provide service to our clients in knowing their specific target audience preferences and ratings on different items as well as providing the client's recommendations on products that the specific target audience might be interested in, so that they can better provide their services and products to their customers. Individual customers can also use our service to check on relatable or similar products based on the product they enter in our service, and we will web crawl it on other websites and provide the answer to the individuals. The crawling component will be through websites like Lazada, Amazon, Shopee, Alibaba.

## Document target audience and objectives

The document is created for project stakeholders, supervisor, accessor, and our development team. So as to allow everyone to have a clearer view of the whole project from scratch to completed version, its purpose and its development processes as well as requirements.

## Glossary

**Python:** A dynamic, high-level, free open source, and interpreted programming language, often used to build websites and software, automate tasks and conduct data analysis.

**PHP:** Hypertext Preprocessor. A widely used open-source general purpose server-side scripting language used for dynamic web development and can be embedded into HTML codes.

**MYSQL:** My Structured Query Language. An open-source relational database management system. As with other relational databases, MYSQL stores data in tables made up of rows and columns. Users can define, manipulate, control and query data using Structured Query Language, more commonly known as SQL. The standard language for relational database management systems.

**Scrapy:** An open-source fast high-level web crawling and web scraping framework, used to crawl websites and extract structured data from their pages. It can be used for a wide range of purposes, from data mining to monitoring and automated testing.

**SVD++:** Singular Value Decomposition. It is a collaborative filtering algorithm where it does factorization of a specific matrix into three matrices. Mainly used to analyse user's preference for each factor and the extent to which it contains the various factors from the observed ratings and some implicit feedback from users and then to predict the missing score.

**Client:** Refers to E-commerce website owners

**Customer:** Refers to individual customer

## Business Model

### Target Audience

The target audience intended for our product service are E-commerce site owners and also any customers that frequent E-commerce sites to purchase products.

### Chosen Model

We have decided to employ a subscription based model since we intend to deliver our recommender system RECS as a Software as a Service (SaaS) to our clients and customers. We will be providing them with SaaS in the form of a website in which they will be able to interact with RECS. Clients will be able to enter their own data set from which they gathered from their own E-commerce site that they own. Clients and customers are also able to make use of RECS to crawl other E-commerce sites such as amazon for example, to retrieve predicted ratings for products or to retrieve recommended products.

### Proposed Price Plan

Since we are employing a subscription based model, clients and customers both can subscribe to our service to use RECS, at a proposed starting monthly price of \$9.90 for Individuals and \$14.90 for Organizations.

## Overall Description

### Product Direction

Focuses on providing clients with the ability to predict specific user's rating on specific items and recommend relatable or similar items that the user might be interested in, so as to allow the e-commerce website to better cater to their target audience and increase revenue. Also, providing individual customers with the ability to get the rating of specific items and recommend them relatable or similar items that they might be interested in also. All these through collaborative filtering algorithm SVD++ and python in producing the system product, PHP and MYSQL in producing the website for customers to navigate, as well as for the product to display the processed data and the management of our customer's databases, Scrapy to web crawl and mine the data in different websites to retrieve existing data sets for prediction and recommendations.

### Product Features

- Inputting of existing database sets
- Predicting of user's rating on specific item
- Retrieving rating of specific item
- Recommending relatable or similar items that users might be interested in based on a specific item inputted
- User Accounts Management
- Customers' confidential privacy
- Subscriptions

## Comparison between SaaS Recommender Systems

### Recombee

Recombee is a relatively popular recommender System that offers partnership with companies to improve their website's recommender system. Once partnered, a dedicated account manager will be introduced to their client and a lot of support on how to manage the data and recommendations.

Pricing is based on the number of page views per month, as it's an approximate indication of the number of active users and recommendation requests. For the upgraded plans, the starting price is 99 USD per month, which is around 130 Singapore Dollars. However, there's also a free plan that allows up to 100000 recommendation requests per month. New users also get to enjoy a 30 Days Free Trial period.

Recombee is also able to deliver over 10000 recommendations per second. Its API can return the recommendations almost immediately upon user's action on the partnered website and it takes as little as 200ms. 9gag is one of the organisations that use Recombee.

Pros	Cons
Easily Accessible	Not user-friendly
30 Days Free Trial	
Free Plan	
Fast	

### Yusp

Yusp, developed by Gravity R & D, is one of the more popular recommendation systems used by organisations like Cora, N11, etc. It provides highly scalable solutions to ecommerce websites.

## Project Requirements

For Yusp, it caters heavily towards clients, where the functionalities are only available to clients with organisations, and they have to book sessions first, after which they are informed of the meeting time, etc. It is unlike Recombee, which provides more leeway and freedom for self exploration.

Yusp favours heavily on client experience and results. The pricing model is on a case by case basis, and heavily depends on personalization modules and traffic size. It is still a monthly subscription, just that the pricing is not set in stone and generally not known to the public.

Only downside is that it's not the most accessible to users who just want to test out the program.

Pros	Cons
Focuses heavily on providing good client experience	Not easily accessible

### **Rumo**

Rumo, formed by Spideo, is a recommendation engine for entertainment content. It has a very user-friendly interface and it's easy to understand its features.

The learning curve is easy. The starting price is at around 100 pounds per month for 1000 catalogue items, which is around 160 Singapore Dollars, and this amount increases based on the number of catalogue items. It also provides a 30 days free trial for users. It is easy to sign up and new users can login immediately upon account creation.

Pros	Cons
Easily Accessible	Expensive
Soft Ergonomics (User Friendly)	
30 Days Free Trial	

### **RECS**

Our product caters towards both companies and individuals. For companies, they can upload their own data and get recommendations of items or prediction of ratings through our services.

For individuals who do not have the datasets, they can still utilise our services as we have service dedicated towards individuals in general. For them, it's really as simple as getting item recommendations just by typing in keywords. The interface is also minimalistic and easy to understand. Security is also emphasised with 2FA authentication. Its speed is also up to par with the other competitors in the industry.

Like Recombee and Rumo, the site's function can be accessed immediately upon subscription and selection of organisation or individual. We will also be offering a 30 Days free trial option, but there will be a limited number of recommendations, as well as limited database size uploadable. For our upgraded plans, the starting price will be at around SGD \$9.90 per month for individuals, and SGD\$19.90 for organizations. This price will increase depending on the number of recommendations.

Below is a table that shows our product in comparison with some of the other similar products in the industry:

## Project Requirements

Products	Recombee	Yusp	Rumo	RECS
<b>Catered towards organizations</b>	✓	✓	✓	✓
<b>Catered towards individuals</b>				✓
<b>Easily accessible. Once signed up or subscribed, service is available immediately</b>	✓		✓	✓
<b>2FA Security</b>				✓
<b>30 Days Free Trial</b>	✓		✓	✓
<b>Starting Price (Below SGD\$50/month)</b>				✓

**References:**

- Pricing [Online]. Recombee. Available at: <https://www.recombee.com/pricing.html> (Accessed: 6 February 2023).
- Recommendation Engine for Real-time Personalization [Online]. Recombee. Available at: <https://www.recombee.com/product.html> (Accessed: 6 February 2023).
- Features [Online]. Join Rumo. Available at: <https://www.rumo.co/features/> (Accessed: 6 February 2023).
- Pricing [Online]. Join Rumo. Available at: <https://www.rumo.co/pricing/> (Accessed: 7 February 2023).
- Product - Yusp Personalization Engine [Online]. Yusp. Available at: <https://www.yusp.com/product/> (Accessed: 7 February 2023).
- Yusp FAQ [Online]. Yusp. Available at: <https://www.yusp.com/knowledge-base/yusp-faq/> (Accessed: 9 February 2023).

## Types of users and features

Client (e-commerce website owner)

- Creating user account
- Management of account
- Inputting of database set into the system
- Retrieving ratings and recommendations based on the data set inputted
- Searching of specific item in the system
- Retrieving ratings and recommendation from our web crawling service
- View history

Customer (individual customers)

- Creating user account
- Management of account
- Searching of specific item in the system
- Retrieving ratings and recommendation from our web crawling service
- View history

Administrator

- Creating user account
- Management and adjusting of website content
- Management(suspend/unsuspend) of customer's accounts
- Deleting user accounts
- Editing of user accounts

## Operating Environment

Operating System: Windows 10

Programming Language: Python

IDE: Jupyter Notebook, Visual Studio Code, Notepad++,

Website: 000webhost, AWS

Database: MySQL

Web Crawling: Scrapy

Browsers supported: Google Chrome, Internet Explorer, FireFox

## Design and Development constraints

- Limited programming skills and knowledge on implementing more creative ideas.
- Limited privileges in free website hosting platforms, unable to maintain the website as desired.
- Limited manpower and resources to design UI as desired.
- Limited capabilities on doing real time web crawling data mining, therefore data sets have to be prepared beforehand time to time by ourselves.

## Assumptions and dependencies

- All clients and customers have internet access.
- All clients and customers have basic computer usage skills and knowledge.
- All clients and customers have to be able to read and understand english.
- All clients and customers know what they are looking for when using our service.

## Researches and Comparisons

### Web Crawling

Web crawlers are programs that's used to index data on web pages. To build a recommendation system, we would need to crawl over tens of thousands of data that spans numerous pages. We are also planning to do the data mining aspect of our product in Python, so it would be in our best interest to find a framework that runs on Python. It helps that Python is also regarded as the most commonly used programming language for web crawling or web scraping. Not to mention our team's familiarity with the Python language.

Taking that into consideration, we did some testing and upon further research, Our team researched and found three frameworks that stood out from the rest. They are three of the more well known frameworks or tools for web crawling, namely Scrapy, Selenium and Beautiful Soup. Upon testing and further research, our team decided that we will be using Scrapy for the product.

### **Scrapy**

Pros	Cons
Big Community	Doesn't support JavaScript by default
Fast	Steep Learning Curve
Consumes little memory and CPU space	
Extensibility	
Auto Throttle function which can adjust speed of requests	
Free	

Scrapy is a free and open-source Python framework that was designed by Mydeco and Insophia, and is currently maintained by Zyte, formerly known as ScrapingHub. It is a framework that is solely constructed for web scraping or web crawling. In fact, it is one of the most well known frameworks for web crawling, if not the most popular. Since Scrapy is built using Twisted, an event-driven networking framework, it is asynchronous, which means that requests aren't made one at a time, but in parallel. This results in fast and efficient crawling, which is very suitable for large scale web crawling, and that makes it a top choice for building scalable web crawlers. It is also extensible, meaning that one can develop and introduce new functions to the existing framework based on one's needs.

To add on, Scrapy also consumes little memory and CPU Space, and one can extract the data out in formats such CSV, JSON and XML. It also has the Auto Throttle setting that allows Scrapy to automatically adjust its speed based on the traffic of the website it is crawling. This will decrease the chance of detection, which increases the rate of success.

## Project Requirements

If there's a downside to using Scrapy, it would be its steep learning curve as well as the fact that it does not handle JavaScript by default. However, this weakness is mostly mitigated as the maintainer of Scrapy, Zyte, has created Scrapy Splash, an alternative that has the capability to handle JavaScript. Splash is essentially a web browser that can process multiple pages and can handle JavaScript. And Scrapy Splash is an integration of Splash and Scrapy.

### **Selenium**

One of the other tools or frameworks that were placed in consideration is Selenium. Developed by Jason Huggins in 2004, Selenium is primarily a free and open-source web browser automation tool, meaning that it renders web pages for test automation of web applications, but it can also be used for web crawling. It runs on Python, Java, C#, etc.

Pros	Cons
Support JavaScript	Slow
Easy Learning Curve	No extensibility
Free	Not memory efficient

In fact, it's precisely because it constantly deals with web browsers which is why Selenium is such a popular choice for web crawling, as it is able to handle JavaScript. This is great for web crawling as many websites use JavaScript to create dynamic content. This is the one thing that it has over Scrapy apart from its easy learning curve. As it handles JavaScript by default, it's easier to use compared to Scrapy Splash, which is way harder to set up.

Moreover, when you consider the fact that Selenium is slower than Scrapy and uses quite a bit of memory, there isn't much of a debate. The way Selenium works is that it renders the entire browser, which includes the HTML, CSS and JavaScript. Only after that then it begins the actual parsing process. Meanwhile, Scrapy simply visits the URL given, with minimum loading and work. As such, it is really not surprising that Selenium isn't as efficient as Scrapy, which is solely a dedicated web scraping tool.

## Project Requirements

Selenium is usually used for small data and also suffers from extensibility, meaning that it will be hard to extend the project when compared to Scrapy, where one can easily develop or add custom functions to the existing framework, allowing for flexibility.

### **Beautiful Soup**

Lastly, another option we considered is Beautiful Soup, which is a free Python package developed by Leonard Richardson, that can help pull data out from HTML or XML files.

Pros	Cons
Easy Learning Curve	Slow (Many dependencies)
Free	Minimal proxy support
	Not memory efficient

Beautiful Soup has an easy learning curve, which is friendly towards beginners. However, it is more of a parser rather than that of a scraping tool, as compared to Scrapy, which is built to be a scraping tool. As such, it requires dependencies to help carry out specific functions. For example, one will need to use the ‘request’ library to make requests to the target webpage. This will result in it being slower, as well as being harder to transfer its code around if required, in comparison to Scrapy.

Beautiful Soup is more suitable for smaller projects. For instance, if it is just about getting specific elements from a target webpage, then Beautiful Soup might be a better choice. It also has minimal proxy support unlike that of Scrapy, so it’s hard to extract a large amount of information without getting IP blocked. Its community also isn’t as big as Scrapy and thus has a lesser support system.

## Project Requirements

### Conclusion

Frameworks/Tools	Beautiful Soup	Selenium	Scrapy
<b>Speed</b>	✓✓	—	✓✓
<b>Memory and CPU Space Efficiency</b>	—	—	✓
<b>Suitable for Big Data</b>	—	—	✓
<b>Community Support</b>	✓	✓	✓✓
<b>Easy Learning Curve</b>	✓	✓	—
<b>Extensibility</b>	—	✓	✓
<b>JavaScript Support by Default</b>	—	✓	—
<b>Good Proxy Support</b>	—	—	✓
<b>Scalability</b>	—	—	✓
<b>Free</b>	✓	✓	✓

In summary, after comparing the pros and cons, as well as testing out the frameworks or tools, our team has come to the conclusion that Scrapy is the framework we will be utilising for this project.

## Project Requirements

As our project would need us to crawl tens of thousands of data, we would need a framework or tool that is fast, uses little memory and doesn't consume large CPU space. It should be scalable and be able to handle large amounts of data. Scrapy fits these requirements to a tee.

Not only is it fast and scalable, it has a huge community to reach out to if we need some sort of support. And it is also extensible. Its Auto Throttle function that adjusts the speed of requests based on traffic is also a great tool to have, to minimise IP blocking. Even though it has its flaws, like not being able to handle JavaScript by default, this can be mitigated with Scrapy Splash. Scrapy Splash may be harder to set up as compared to Selenium, which handles JavaScript by default, but the overall pros of Scrapy easily outweigh its cons. Once we consider the existence of Scrapy Splash, the other frameworks and tools don't really have anything over Scrapy anymore. Hence, there's no real debate why we decided on Scrapy.

## References:

- Ruggeri, L, (2022). *Scrapy Vs. Beautiful Soup: Which is the best choice for your business?* [Online]. Bright Data. Available at: <https://brightdata.com/blog/leadership/scrapy-vs-beautiful-soup> (Accessed: 9 February 2023).
- Siddiqi, (2021). *Scrapy vs Selenium / Web Scraping Frameworks* [Online]. CodersLegacy. Available at: <https://coderslegacy.com/scrapy-vs-selenium-python-frameworks/> (Accessed: 2 February 2023).
- Yin, M, (2021). *Scrapy Tutorial #1: Scrapy VS Beautiful Soup* [Online]. AccordBox. Available at: <https://www.accordbox.com/blog/scrapy-tutorial-1-scrapy-vs-beautiful-soup/> (Accessed 1 February 2023).
- Pycoach, T, (2020). *Web Scraping with Beautiful Soup, Selenium, or Scrapy?* [Online]. Towards Data Science. Available at: <https://towardsdatascience.com/web-scraping-with-beautiful-soup-selenium-or-scrapy-62c6f3545de7> (Accessed 31 Jan 2023).
- (2022). *BeautifulSoup vs. Scrapy: Which Is Better For Web Scraping?* [Online]. Scraping Robot. <https://scrapingrobot.com/blog/beautifulsoup-vs-scrapy/> (Accessed 1 February 2023).
- *Intro to Web Scraping with Scrapy.* [Online]. ScrapeOps. Available at: <https://scrapeops.io/python-scrapy-playbook/scrapy-web-scraping-intro/> (Accessed 1 February 2023).
- Yin, M. (2021). *Web Scraping Framework Review: Scrapy VS Selenium.* [Online]. AccordBox. Available at: <https://www.accordbox.com/blog/web-scraping-framework-review-scrapy-vs-selenium/> (Accessed 1 February 2023).

## Collaborative Filtering Algorithms

### Analysis and comparison of SVD vs SVD++ vs RandomForestRegressor

#### SVD:

The prediction  $\hat{r}_{ui}$  is set as:

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T p_u$$

If user  $u$  is unknown, then the bias  $b_u$  and the factors  $p_u$  are assumed to be zero. The same applies for item  $i$  with  $b_i$  and  $q_i$ .

For details, see equation (5) from [KBV09]. See also [RRSK10], section 5.3.1.

To estimate all the unknown, we minimize the following regularized squared error:

$$\sum_{r_{ui} \in R_{train}} (r_{ui} - \hat{r}_{ui})^2 + \lambda (b_i^2 + b_u^2 + ||q_i||^2 + ||p_u||^2)$$

The minimization is performed by a very straightforward stochastic gradient descent:

$$\begin{aligned} b_u &\leftarrow b_u + \gamma(e_{ui} - \lambda b_u) \\ b_i &\leftarrow b_i + \gamma(e_{ui} - \lambda b_i) \\ p_u &\leftarrow p_u + \gamma(e_{ui} \cdot q_i - \lambda p_u) \\ q_i &\leftarrow q_i + \gamma(e_{ui} \cdot p_u - \lambda q_i) \end{aligned}$$

where  $e_{ui} = r_{ui} - \hat{r}_{ui}$ . These steps are performed over all the ratings of the trainset and repeated `n_epochs` times. Baselines are initialized to `0`. User and item factors are randomly initialized according to a normal distribution, which can be tuned using the `init_mean` and `init_std_dev` parameters.

You also have control over the learning rate  $\gamma$  and the regularization term  $\lambda$ . Both can be different for each kind of parameter (see below). By default, learning rates are set to `0.005` and regularization terms are set to `0.02`.

Both screenshots are to explain SVD functions on how the bias and minimization of regularized squared error using stochastic gradient descent. SVD won a third prize for Netflix prize competition, popularized by Simon Funk.

## Project Requirements

### Pros:

SVD method has benefits such as reducing data complexity through extraction of latent factors, removing noise, improving algorithm efficiency, and being more reliable than other matrix factorization techniques. SVD also tackles data sparsity and scalability issues in recommendation systems.

### Cons:

SVD is sensitive to outliers and irrelevant data. SVD also has difficulty in handling implicit feedback data. Many users do not rate their purchased items upon which to base a prediction, therefore collaborative filtering cannot provide recommendations for new items. Accurate recommendations for an item may not be possible until enough ratings have been received, which may take some time even after users start rating the item.

### SVD:

Pros	Cons
Reducing data complexity through extraction of latent factors	Sensitive to outliers and irrelevant data
Remove noise and increase efficiency of the Algorithm	Difficulty in handling implicit feedback data
Tackles data sparsity and scalability issues in recommendation systems	Must have enough recommendations to have a decent accuracy

### SVD++:

The SVD++ algorithm, an extension of `svd` taking into account implicit ratings.

The prediction  $\hat{r}_{ui}$  is set as:

$$\hat{r}_{ui} = \mu + b_u + b_i + q_i^T \left( p_u + |I_u|^{-\frac{1}{2}} \sum_{j \in I_u} y_j \right)$$

Where the  $y_j$  terms are a new set of item factors that capture implicit ratings. Here, an implicit rating describes the fact that a user  $u$  rated an item  $j$ , regardless of the rating value.

If user  $u$  is unknown, then the bias  $b_u$  and the factors  $p_u$  are assumed to be zero. The same applies for item  $i$  with  $b_i$ ,  $q_i$  and  $y_i$ .

For details, see section 4 of [Kor08]. See also [RRSK10], section 5.3.1.

Just as for `svd`, the parameters are learned using a SGD on the regularized squared error objective.

Baselines are initialized to `0`. User and item factors are randomly initialized according to a normal distribution, which can be tuned using the `init_mean` and `init_std_dev` parameters.

You have control over the learning rate  $\gamma$  and the regularization term  $\lambda$ . Both can be different for each kind of parameter (see below). By default, learning rates are set to `0.005` and regularization terms are set to `0.02`.

Screenshot taken to explain SVD++ functions.

SVD(++) works by improving Matrix Factorization, matrix R into the product of three matrices:  $R = U * D * V^t$ .

#### Pros:

Like SVD, SVD++ can be scalable to fit in large datasets. Nevertheless, SVD++ gradient descent introduces backtracking, and takes in implicit account ratings.

#### Cons:

SVD++ faces similar cons as SVD. However, SVD++ takes in implicit account ratings, therefore eliminating SVD (difficulty in handling implicit feedback data) drawback.

## Project Requirements

Pros	Cons
Reducing data complexity through extraction of latent factors	Sensitive to outliers and irrelevant data
Remove noise and increase efficiency of the Algorithm	Must have enough recommendations to have a decent accuracy
Tackles data sparsity and scalability issues in recommendation systems	
Gradient descent introduces backtracking for increase accuracy	

### **RandomForestRegressor:**

#### **Pros:**

The RandomForestRegressor contains a bundle of decision trees to make a classification and it is also considered a saving technique when it comes to overfitting of a decision tree model. RandomForestRegressor are efficient to train, depending on the number of trees needed to create.

#### **Cons:**

A significant drawback of Random Forest is that an excessive number of trees can render the algorithm sluggish and unsuitable for predictions in real-time with sparse data. They tend to be slow when generating predictions after training. The accuracy of prediction can increase due to the greater number of trees. Random forest requires a higher time and space to train the model as a larger number of trees are involved. If the number of weak learners (decision trees) in the Random Forest, represented by  $n$ , is not sufficient, it can result in underfitting. On the other hand, if  $n$  is excessive, it leads to increased computational cost, and beyond a certain point, increasing  $n$  will have minimal impact on improving the model. Therefore, it is essential to find an optimal value of  $n$  that balances both the tree number and learning efficiency.

## Project Requirements

Pros	Cons:
Building a bundle of trees helps it to overcome overfitting of a decision tree model	Unsuitable for predictions with sparse data
Efficient to train, depending on the number of trees needed to create	Greater the number of trees, the higher time to train
	Sensitive to setting the number of learners (decision trees) which may lead to underfitting or increased computational cost
	Slow when generating predictions after training

**Summary:**

Machine Learning Algorithms	SVD	SVD++	RandomForestRegressor
Matrix Factorization (with SGD)	✓	✓✓	✗
Decision Trees	✗	✗	✓
Scalable for large datasets	✓	✓	✓
Implicit Account Ratings	✗	✓	✗
Stop overfitting of dataset	✓	✓	✓
Sparse data	✓	✓	✗
Accuracy	✓	✓✓	✓

**References:**

- <https://surprise.readthedocs.io/en> [Online]. Available at: [https://surprise.readthedocs.io/en/stable/matrix\\_factorization.html](https://surprise.readthedocs.io/en/stable/matrix_factorization.html) (Accessed: 30 January 2023).
- <https://sifter.org> [Online]. Available at: <https://sifter.org/~simon/journal/20061211.html> (Accessed: 31 January 2023).
- [www.mdpi.com](http://www.mdpi.com) [Online]. Available at: <https://www.mdpi.com/2504-2289/6/4/121> (Accessed: 31 January 2023).
- [www.analyticsvidhya.com](http://www.analyticsvidhya.com) [Online]. Available at: [https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-recommendation-engine-python/?utm\\_source=blog&utm\\_medium=5-applications-singular-value-decomposition-svd-data-science](https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-recommendation-engine-python/?utm_source=blog&utm_medium=5-applications-singular-value-decomposition-svd-data-science) (Accessed: 1 February 2023).
- <https://surprise.readthedocs.io> [Online]. Available at: [https://surprise.readthedocs.io/en/stable/matrix\\_factorization.html](https://surprise.readthedocs.io/en/stable/matrix_factorization.html) (Accessed: 1 February 2023).
- [www.mdpi.com](http://www.mdpi.com) [Online]. Available at: <https://www.mdpi.com/2078-2489/11/7/369> (Accessed: 1 February 2023).
- [www.section.io](http://www.section.io) [Online]. Available at: <https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/> (Accessed: 2 February 2023).
- [www.rebellionresearch.com](http://www.rebellionresearch.com) [Online]. Available at: <https://www.rebellionresearch.com/what-are-the-disadvantages-of-random-forest> (Accessed: 2 February 2023).
- <https://builtin.com> [Online]. Available at: <https://builtin.com/data-science/random-forest-algorithm> (Accessed: 3 February 2023).

## Programming Languages

### **Analysis of Python vs R**

#### **Python**

**Pros:** Python is a versatile programming language that can be used for a variety of tasks, including web development, data manipulation, and machine learning, due to its general-purpose nature. Even if the designers try to use a lower-level language like C, C++, or Java, it is often more effective to integrate different components using a Python wrapper. Free to download and has many IDEs to code on like Visual Studio, Notepad++ or Jupyter Notebook through Anaconda.

**Cons:** Includes a very few statistical model packages. The Global Interpreter Lock (GIL) makes threading in Python challenging and causes multi-threaded CPU-bound applications tend to perform slower than single-threaded ones. Therefore, for AI projects, it is more advantageous to use multiprocessing instead of multithreaded programming.

## Project Requirements

Pros	Cons
Versatile programming language	Includes a very few statistical model packages
Can be used for wide range of tasks such as data manipulation and machine learning	Multi-threaded workloads tend to not work as well and have to use multiprocessing to overcome multithreading
Effective to integrate different lower-level languages like C, C++, or Java	
Free to download and has many IDEs to code on like Visual Studio, Notepad++, etc.	

## Project Requirements

### R

**Pros:** R is a statistical programming language. R is primarily used for statistical analysis and data visualization. Free to download and have a good IDE like Rstudio.

**Cons:** R has a steep learning curve to new users. Learning every new package slows down development and the lack of comprehensive documentation for R exacerbates the problem. Not backwards compatible with older basic libraries versions e.g. R3.6.3 version, unable to use Hmsic v4.5.

Pros	Cons
Statistical programming language	Steep learning curve
Good for statistical analysis and data visualization	Lack of comprehensive documentation for packages slows down development speed
Free to download and have a good IDE like Rstudio	Not backwards compatible with older machine learning or basic libraries

## Project Requirements

### Summary

Programming Language	Python	R
Compatible with other languages	✓	✗
Machine Learning packages	✓	✓
Backwards compatible to older libraries	✓	✗
Learning Curve	✓	✗
Free	✓	✓

### References:

- [www.datacamp.com \[Online\]. Available at:  
https://www.datacamp.com/blog/python-vs-r-for-data-science-whats-the-difference](https://www.datacamp.com/blog/python-vs-r-for-data-science-whats-the-difference)  
(Accessed: 6 February 2023).
- <https://towardsdatascience.com/python-vs-r-whats-best-for-machine-learning-93432084b480> [Online]. Available at:  
https://towardsdatascience.com/python-vs-r-whats-best-for-machine-learning-93432084b480  
(Accessed: 7 February 2023).

## Python vs PHP vs Java vs C#

Language \ Tools	Python	PHP	Java	C#
Free	✓	✓	✓	✓
Object Oriented Programming	✓	✓	✓	✓
Wide range of library			✓	✓
Easy to learn	✓	✓		
Fast loading rate		✓		✓
Server-side programming	✓	✓	✓	✓
Fast loading time		✓		
security	✓		✓	✓

As shown from the table above, we have decided to use PHP as our main web programming language.

PHP is one of the best languages for web development. It is a scripting language mainly used for the server-side of simple, dynamic sites and web apps, which we think will be most suitable for our project.

Although the security for PHP is not the best, we can prevent those vulnerabilities ourselves with good coding practice. One of the examples will be SQL injection. Instead of normal SQL query with no validation or sanitization, we can use a prepared statement to prevent it.

## Database comparison

Database Tools	MySQL	Maria DB	PostgreSQL	MongoDB	firebase
Easy to learn	✓	✓		✓	✓
security	✓	✓	✓		
Speed	✓	✓	✓	✓	✓
Reliability	✓	✓	✓		
scalability				✓	✓

As shown from the table above, we decided to use MySQL as our database for the web application.

MySQL is one of the most popular relational database systems. It is also renowned for being the most secure and reliable database management system used in web applications. It features a different storage engine framework that facilitates system administrators to configure the MySQL database server for flawless performance.

## Individual Pros and Cons of language and database chosen

### **PHP:**

#### Pros of PHP:

- Faster rate of loading over slow internet speed.
- Offers embedded HTML programming. Good synergy between HTML and PHP
- Flexible and combines well with other web development technologies.
- Open-source web language, hence, is completely free (budget friendly)

#### Cons of PHP:

- Security is not at its best as the ACSII text files will be easily accessible.

### **MYSQL database:**

#### Pros of MySQL:

- Assurance of 24/7 uptime.
- offers host-based verification and password encryption.
- It has a unique storage engine that makes the administrator easier to configure.
- High performance. Ensure optimal speed with most demanding applications.
- It is open source and free.

#### Cons of MySQL:

- It is challenging to debug stored procedures.
- challenging to cope with the syntax of SQL in MySQL.
- limited compliance with SQL standard.

## Project Requirements

### References:

<https://www.altexsoft.com/blog/business/comparing-database-management-systems-mysql-postgresql-mssql-server-mongodb-elasticsearch-and-others/>

## Hosting Platforms

### Hosting platform comparison

	Github.io	firease	wordpress	webHost	AWS
free	✓	✓		✓	✓
Easy to use	✓	✓	✓	✓	
Supports PHP			✓	✓	✓
Supports MySQL	✓		✓	✓	✓
Limited storage (free ver.)	✓	✓	✓	✓	✓
Security	✓		✓		✓

Webhost is an online web hosting platform that is completely free. Although it is not one of the most secure providers as SSL certificates are not included, it has most of the basic features of a paid service.

For security such as password, it will be hashed before storing it in the database.

For now we are using a website called webhost (<https://www.000webhost.com>). But as we progress, we may not be able to use it as if our data set comes in, I think it will hit the limited space of the free version.

The link below is the in-production webpage. One sample data for username as password is also provided. You should be able to log in and out.

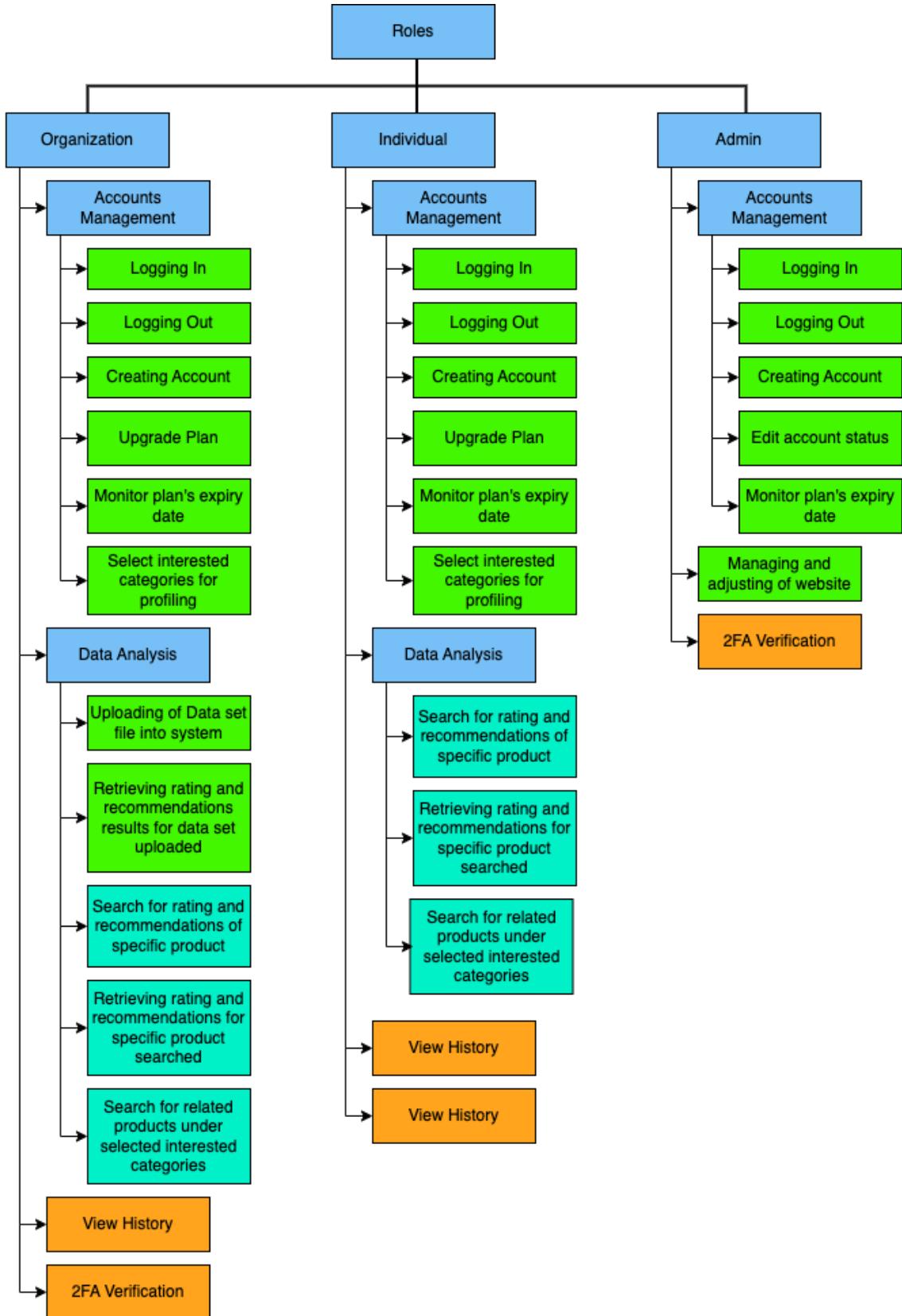
(Username: Tan, Password: Pass123) (<https://trailrun123.000webhostapp.com/index.php>)

WordPress needs to subscribe to and pay before you can use it, Github.io doesn't support PHP as we tried it and have to change to HTML before it can run.

For AWS, it is quite complicated. We have tried different ways to play around it and it is not what we are looking for. We followed the steps in YouTube and it was something like we have to have a virtual setup to publish it there.

# Requirements

## Product Features Mind Map



## Functional Requirements

#	Requirement Description	Priority	Use Case ID
1	As an Administrator I want to be able to log in to my account to perform my administrative duties	High	UC1
2	As an Administrator I want to be able to log out of my account so that I can end session for security reasons	High	UC2
3	As an Administrator I want to be able to create user accounts for organizations/individuals who subscribe to the service	High	UC3
4	As an Administrator I want to be able to edit user accounts information so that I can assist users in editing their information if needed	High	UC4
5	As an Administrator I want to delete user accounts to reduce the risk of inactive accounts	High	UC5
6	As an Administrator I want to manage user account status so that I can suspend users	Medium	UC6
7	As an Administrator I want to be able to monitor users subscription plans so that I can help remind/prompt them for renewal	High	UC7
8	As an Administrator I want there to be 2FA secure log in so that unauthorized access to my account is prevented	medium	UC8

## Project Requirements

9	As a Organization I want to be able to login to access the website dashboard and use the service	High	UC9
10	As a Organization I want to be able to log out so I can end the session and prevent unauthorized access/use of my account	High	UC10
11	As an Organization I want to be able to create my own user account to access and use the service	High	UC11
12	As an Organization I want to be able to edit my account information so I can edit my details if there are changes	High	UC12
13	As an Organization I want to be able to upload my own customer data set into the system to create the predictive model used for ratings prediction and recommendations	High	UC13
14	As an Organization I want to be able to delete my uploaded data set so that I remove old data set that is no longer relevant	High	UC14
15	As an Organization I want to be able to use RECS' data to generate a ratings prediction to predict how a user 'x' will rate a product 'x'	High	UC15
16	As an Organization I want to be able to use RECS' data generate product recommendations based on the entered product so that I can recommend them to my	High	UC16

## Project Requirements

	customers		
17	As an Organization I want to be able to add a list of URLs so that those ecommerce sites can be crawled and be used to create a new prediction model	High	UC17
18	As an Organization I want to be able to generate a ratings prediction based on the web crawling service to predict how a user 'x' will rate a product 'x'	High	UC18
19	As an Organization I want to be able to generate product recommendations of an entered product from the provided web crawling services, so that I can recommend them to my customers	High	UC19
20	As an Organization I want to be able to view the result history of ratings/recommendations I generated	Medium	UC20
21	As an Organization I want to be able to upgrade my subscription plan so that I can use the features freely	High	UC21
22	As an Organization I want to be able to monitor my plan's expiry so that I can know when to renew the plan	High	UC22
23	As an Organization I want to be able to choose what category of products I am interested in so that I can get better rating predictions and recommendations on products	High	UC23
24	As an Organization I want there to be 2FA secure log in so that unauthorized access to my account is prevented	medium	UC24

## Project Requirements

25	As an Individual I want to be able to login to access the web dashboard and service	High	UC25
26	As an Individual I want to be able to log out so that I can end my session to ensure no unauthorized access/use of my account	High	UC26
27	As an Individual I want to be able to create a user account to access and use the service	High	UC27
28	As an Individual I want to be able to edit my account information to edit my details if there are any changes	High	UC28
29	As an Individual I want to be able to use RECS' data to generate a ratings prediction to predict how a user 'x' will rate a product 'x'	High	UC29
30	As an Individual I want to be able to use RECS' data generate product recommendations based on a product i enter so i can get recommendations for similar products	High	UC30
31	As an Individual I want to be able to add a list of URLs so that those ecommerce sites can be crawled and be used to create a new prediction model	High	UC31
32	As an Individual I want to be able to generate a ratings prediction based on the web crawling service to predict how a user 'x' will rate a product 'x'	High	UC32

## Project Requirements

33	As an Individual I want to be able to retrieve recommendations for products from web crawling service provided based on the product I enter	High	UC33
34	As an Individual I want to be able view result history of ratings and recommendations I generated	Medium	UC34
35	As an Individual I want to be able to upgrade my subscription plan so that I can use of all the features available without limits	High	UC 35
36	As an Individual I want to be able to monitor my plan expiry plan so that I can know when to renew my subscription	High	UC36
37	As an Individual I want 2FA I want there to be 2FA secure log in so that unauthorized access to my account is prevented	Medium	UC37

## Administrator Use Cases

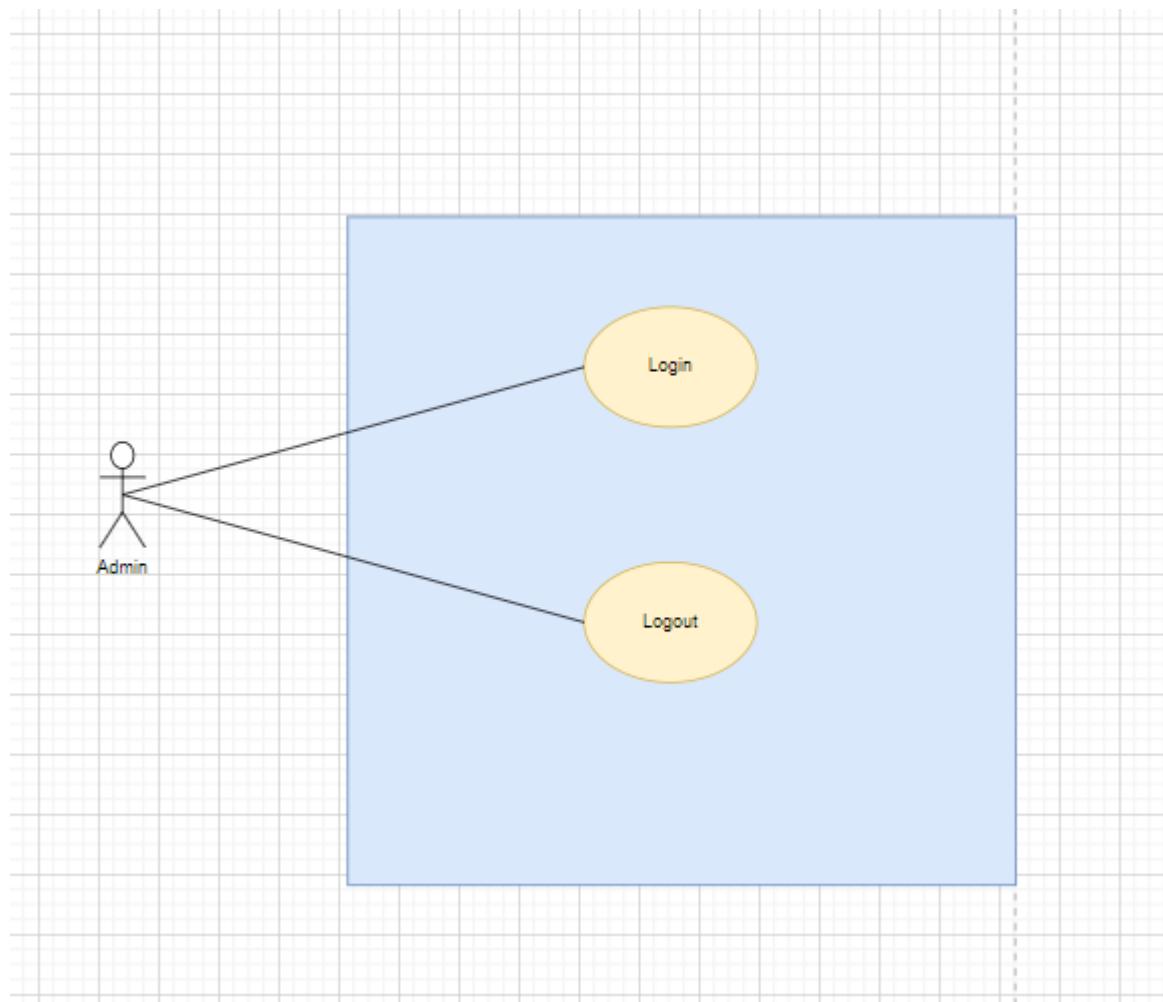


Figure 1: UC1 and UC2

## Project Requirements

### Use Case Description 1

Use Case ID	UC1
Use Case Name	Admin login
Actor(s)	Admin
Goal	Admin is able to login to their account
Pre-Condition	Admin already has an account created
Post-Condition	Admin successfully logged in
Trigger(s)	Admin wants to log in to their account
Main Flow	<ol style="list-style-type: none"><li>1. Admin sees website display form asking for login credentials</li><li>2. Admin enters credentials</li><li>3. Clicks the “login” button</li><li>4. Website verifies the credentials</li><li>5. Admin will land on the homepage</li><li>6. End of flow</li></ol>
Sub Flow	N.A
Exception	<p>Invalid credentials:</p> <ol style="list-style-type: none"><li>1. Admin see error message pop up “Invalid credentials”</li><li>2. Admin closes the pop up</li><li>3. Admin re-enters their credentials</li><li>4. End of flow</li></ol>

## Project Requirements

### Use Case Description 2

Use Case ID	UC2
Use Case Name	Admin log out
Actor(s)	Admin
Goal	Admin is able to log out of their account
Pre-Condition	Admin already logged in their account
Post-Condition	Admin successfully logs out of their account
Trigger(s)	Admin wants to log out of their account to end their session
Main Flow	<ol style="list-style-type: none"><li>1. Admin clicks the “logout” button</li><li>2. website logs admin out</li><li>3. Admin will be redirected back to the login page</li><li>4. Session ended</li><li>5. End of flow</li></ol>
Sub Flow	N.A
Exception	N.A

## Project Requirements

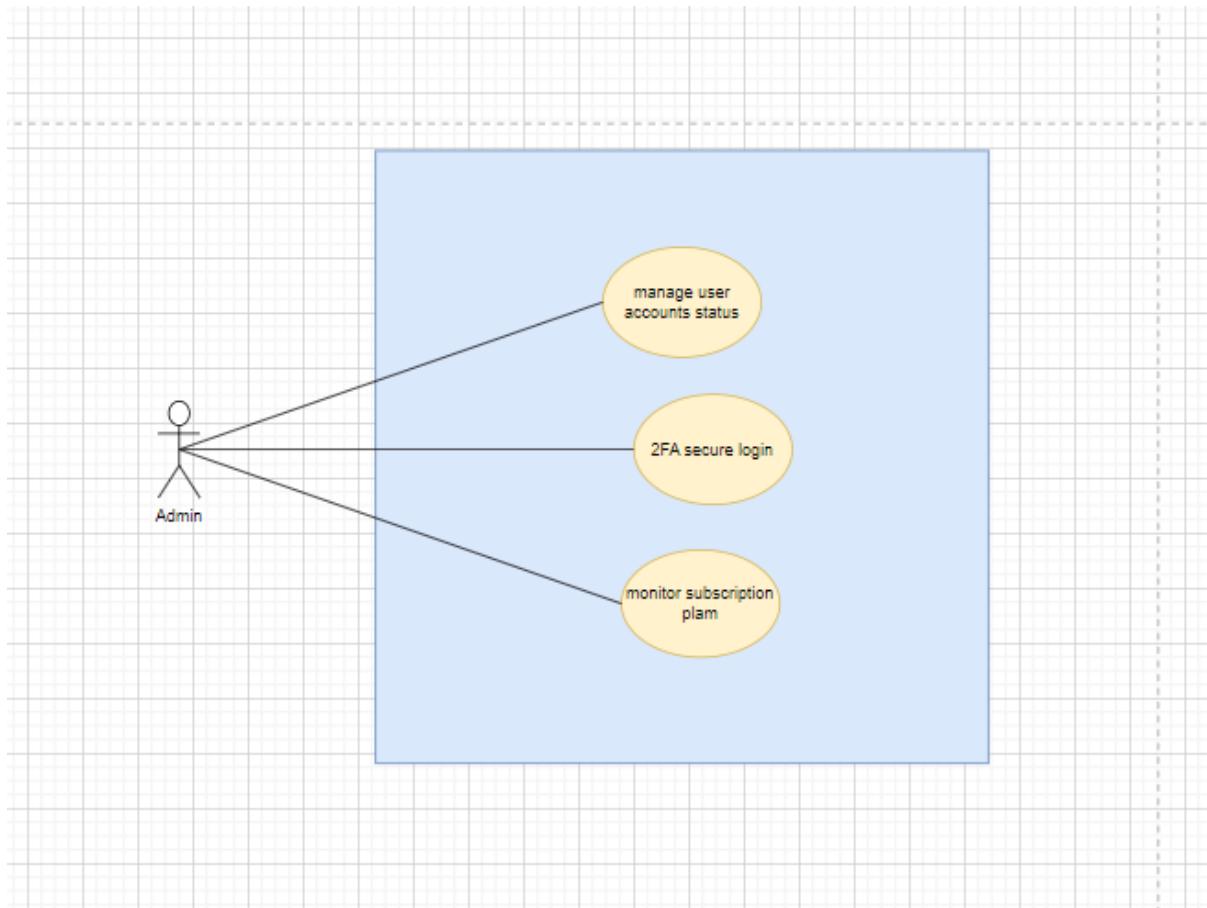


Figure 2: UC3, UC4 and UC5

## Project Requirements

### Use Case Description 3

Use Case ID	UC3
Use Case Name	Admin creates user account
Actor(s)	Admin
Goal	Admin will be able to create user accounts for clients/customers
Pre-Condition	Admin must be logged into their account
Post-Condition	A new user account has been created
Trigger(s)	Admin creating a user account for new clients/customers
Main Flow	<ol style="list-style-type: none"> <li>1. Admin clicks on the “create new account” button</li> <li>2. A new account form is displayed by website</li> <li>3. Admin fills in the informations of the new user</li> <li>4. Admin makes credentials for the new user to use for logging in</li> <li>5. Admin clicks “create” button to submit</li> <li>6. The website validates and stores the new account and its information</li> <li>7. New user account has been created</li> <li>8. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input:</p> <p>Website display error pop up “invalid input”</p> <p>Admin closes pop up</p> <p>Repeat main flow</p> <p>End of flow</p>

## Project Requirements

### Use Case Description 4

Use Case ID	UC4
Use Case Name	Admin edits user account
Actor(s)	Admin
Goal	Admin will be able to edit user accounts for clients/customers
Pre-Condition	<p>Admin must be logged into their account</p> <p>User account has to exists/already been created</p>
Post-Condition	User account information has been edited.
Trigger(s)	Client/Customer asks admin to help edit their user account information
Main Flow	<ol style="list-style-type: none"> <li>1. Admin clicks on the “edit account” button</li> <li>2. Website displays edit form</li> <li>3. Admin changes the informations of the client/customer as per requested</li> <li>4. Admin clicks “confirm” button to submit</li> <li>5. The website validates and stores changes made</li> <li>6. User account successfully edited</li> <li>7. User account reflect changes made</li> <li>8. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input:</p> <ol style="list-style-type: none"> <li>1. Website display error pop up “invalid input”</li> <li>2. Admin closes pop up</li> <li>3. Repeat main flow</li> <li>4. End of flow</li> </ol> <p>Admin cancels:</p> <ol style="list-style-type: none"> <li>1. Admin clicks “cancel” button</li> <li>2. confirmation message prompt appears</li> <li>3. Admin clicks “yes”</li> <li>4. Prompt and edit form closes</li> </ol>

## Project Requirements

	5. User account page displayed with no changes 6. End of flow
--	--

## Project Requirements

### Use Case Description 5

Use Case ID	UC5
Use Case Name	Admin deletes user account
Actor(s)	Admin
Goal	User account gets deleted
Pre-Condition	Admin logged into their account User account has to exist/already been created
Post-Condition	Selected user account is successfully deleted
Trigger(s)	Admin deletes user account of clients/customer that are no longer subscribed Client/customer requests for their account to be deleted
Main Flow	<ol style="list-style-type: none"> <li>1. Admin clicks “delete” button</li> <li>2. confirm delete prompt appears</li> <li>3. Admin clicks “yes”</li> <li>4. website deletes stored user account information</li> <li>5. Page refreshes and user account successfully deleted</li> <li>6. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Cancel:</p> <ol style="list-style-type: none"> <li>1. On confirm delete prompt admin clicks cancel</li> <li>2. prompt closes</li> <li>3. User account not deleted</li> <li>4. End of flow</li> </ol>

## Project Requirements

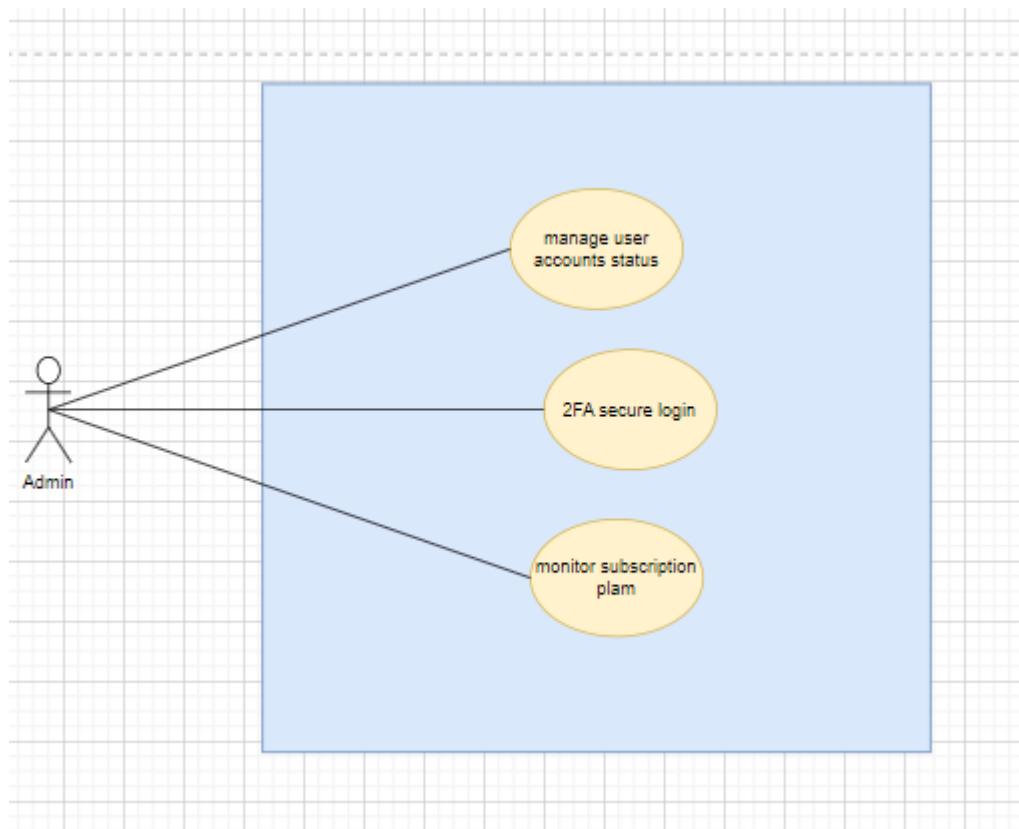


Figure 3: UC6, UC7 and UC8

## Project Requirements

### Use Case Description 6

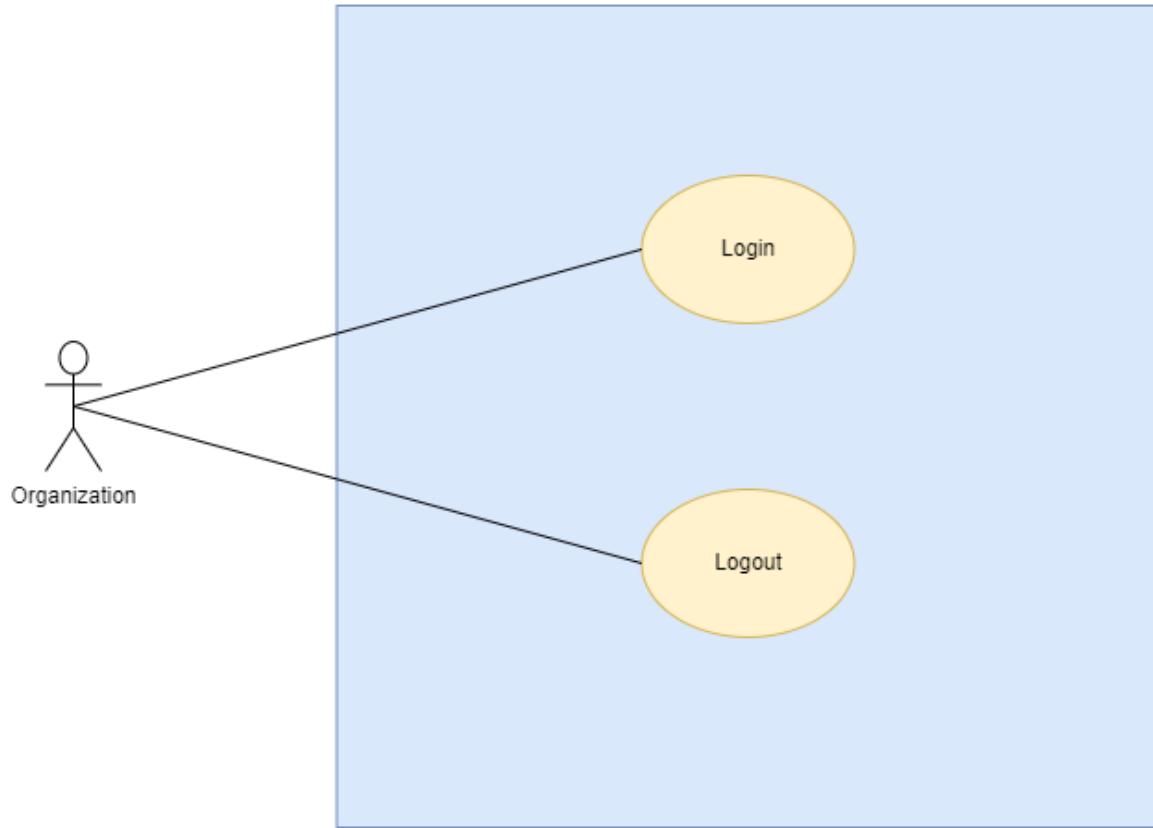
Use Case ID	UC6
Use Case Name	Admin manages user account
Actor(s)	Admin
Goal	User account gets suspended/unsuspended
Pre-Condition	Admin logged into their account User account has to exist/already been created
Post-Condition	Selected user account is suspended/unsuspended
Trigger(s)	Client/customer wants to pause their subscription so the admin needs to suspend their account (instead of delete) Client/customer wants to resume their subscription. Admin needs to unsuspend their account
Main Flow	Suspend account: <ol style="list-style-type: none"><li>1. Admin clicks “manage account” button</li><li>2. pop up appears prompting “suspend”/“unsuspend”</li><li>3. Admin clicks “suspend”</li><li>4. website changes the user account status</li><li>5. Page refreshes and user account successfully suspended</li><li>6. End of flow</li></ol>
Sub Flow	Unsuspend account: <ol style="list-style-type: none"><li>1. Admin clicks “manage account” button</li><li>2. pop up appears prompting “suspend”/“unsuspend”</li><li>3. Admin clicks “unsuspend”</li><li>4. website changes the user account status</li><li>5. Page refreshes and user account successfully unsuspended</li><li>6. End of flow</li></ol>
Exception	N.A

## Project Requirements

### Use Case Description 7

Use Case ID	UC7
Use Case Name	Admin manages website content
Actor(s)	Admin
Goal	Edit/change website display content
Pre-Condition	Admin logged into their account
Post-Condition	Website content changed
Trigger(s)	Admin wants to edit/change content on display on the website
Main Flow	<ol style="list-style-type: none"> <li>1. Admin clicks “manage website” button</li> <li>2. Webpage becomes editable</li> <li>3. Admin makes changes to necessary display content</li> <li>4. Admin clicks “confirm changes” at the top right</li> <li>5. Website processes and save changes</li> <li>6. Webpage refresh display changes</li> <li>7. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Cancel:</p> <ol style="list-style-type: none"> <li>1. On confirm delete prompt admin clicks cancel</li> <li>2. prompt closes</li> <li>3. Website display content not changed</li> <li>4. End of flow</li> </ol>

## Organization Use Cases



*Figure 4: UC8 and UC9*

## Project Requirements

### Use Case Description 8

Use Case ID	UC8
Use Case Name	Organization login
Actor(s)	Organization
Goal	Organization able to log in and access their account
Pre-Condition	Organization already created an account
Post-Condition	Organization successfully logged in to the account
Trigger	Organization wants to login and access the service
Main Flow	<ol style="list-style-type: none"> <li>1. Organization sees website display form asking for login credentials</li> <li>2. Organization enters credentials</li> <li>3. Clicks the “login” button</li> <li>4. Website verifies the credentials</li> <li>5. Organization will land on the homepage</li> <li>6. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid credentials:</p> <ol style="list-style-type: none"> <li>1. Organization see error message pop up “Invalid credentials”</li> <li>2. Organization closes the pop up</li> <li>3. Organization re-enters their credentials</li> <li>4. End of flow</li> </ol>

## Project Requirements

### Use Case Description 9

Use Case ID	UC9
Use Case Name	Organization log out
Actor(s)	Organization
Goal	Organization is able to log out of their account
Pre-Condition	Organization already logged in their account
Post-Condition	Organization successfully logs out of their account
Trigger(s)	Organization wants to log out to their account to end their session
Main Flow	<ol style="list-style-type: none"><li>1. Organization clicks the “logout” button</li><li>2. website logs admin out</li><li>3. Organization will be redirected back to the login page</li><li>4. Session ended</li><li>5. End of flow</li></ol>
Sub Flow	N.A
Exception	N.A

## Project Requirements

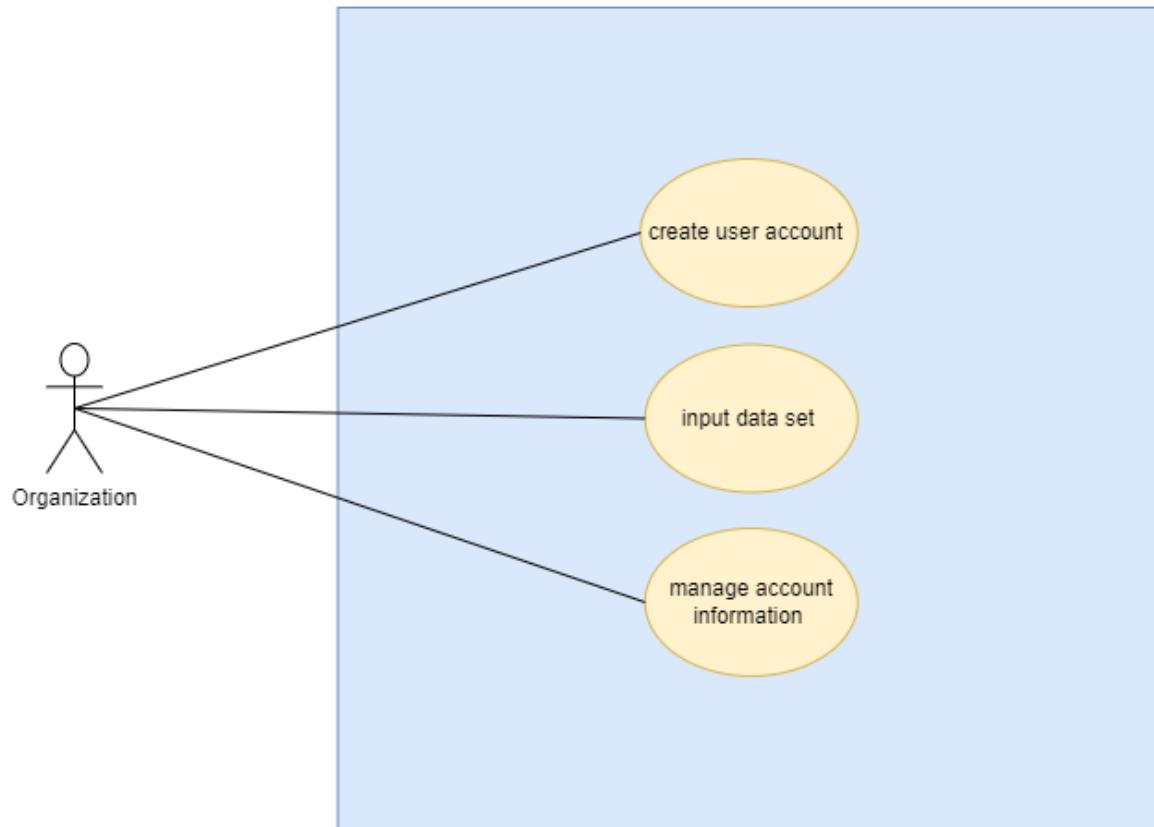


Figure 5: UC10, UC11 and UC12

## Project Requirements

### Use Case Description 10

Use Case ID	UC10
Use Case Name	Organization creates user account
Actor(s)	Organization
Goal	Organization will be able to create their own user account
Pre-Condition	Organization is at the main landing page (login page)
Post-Condition	A new user account has been created
Trigger(s)	Organization creating a new user account for themselves
Main Flow	<ol style="list-style-type: none"> <li>1. Organization clicks on the “create new account” button at the login page</li> <li>2. A new account form is displayed by website</li> <li>3. Organization fills in their necessary information</li> <li>4. Organization makes credentials for them to use for logging in</li> <li>5. Organization clicks “create” button to submit</li> <li>6. The website validates and stores the new account and its information</li> <li>7. New user account has been created</li> <li>8. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input:</p> <ol style="list-style-type: none"> <li>1. Website display error pop up “invalid input”</li> <li>2. Organization closes pop up</li> <li>3. Repeat main flow</li> <li>4. End of flow</li> </ol>

## Project Requirements

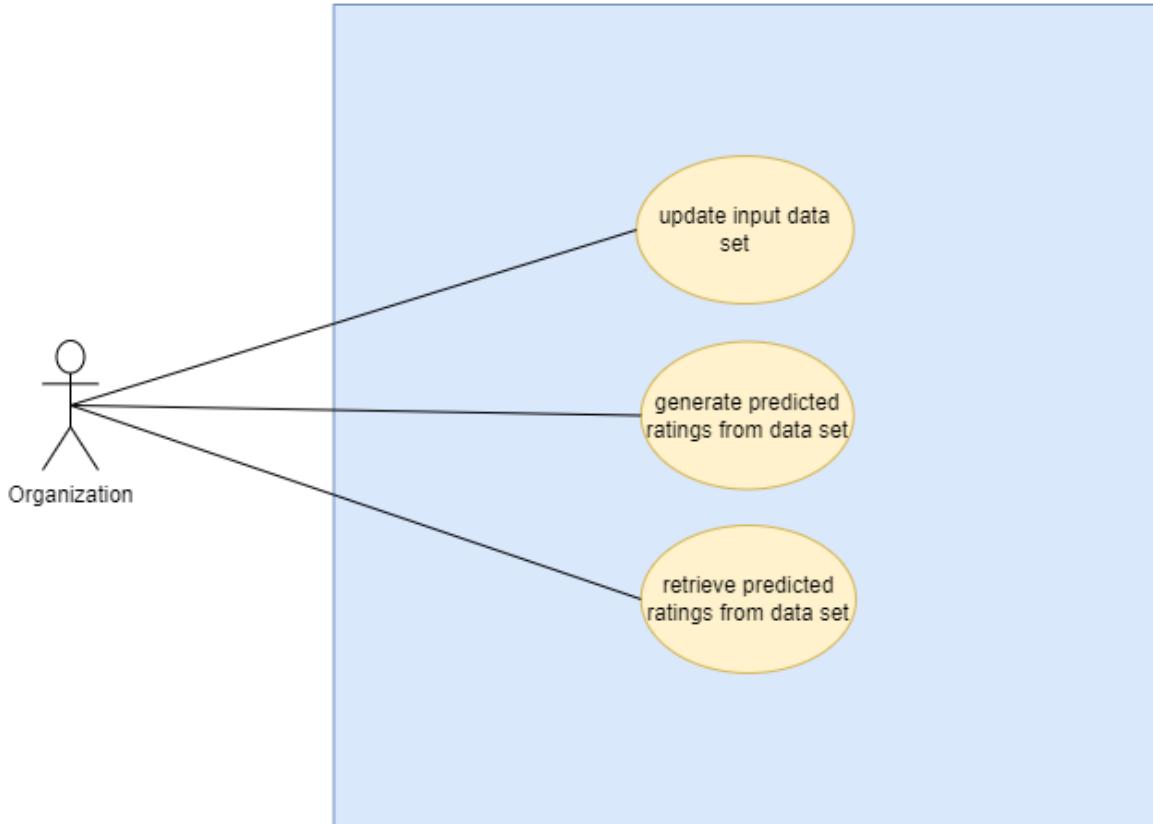
### Use Case Description 11

Use Case ID	UC11
Use Case Name	Organization inputs data set
Actor(s)	Organization
Goal	Organization will be able to input their own data set for RECS system to learn
Pre-Condition	Organization must have already been logged into their account
Post-Condition	Data set inputted into RECS system
Trigger(s)	Organization wants to input their own customer data set into the RECS system to train it
Main Flow	<ol style="list-style-type: none"> <li>1. Organization clicks on the “add data set” button</li> <li>2. Organization is popup appears prompting Organization to upload their data set</li> <li>3. Organization clicks “add” button at the bottom of popup</li> <li>4. Website takes the data set and sends it to the RECS system</li> <li>5. RECS system process and learn the data set</li> <li>6. Website redirects Organization back to their homepage with popup “successfully added data set”</li> <li>7. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input(no data set selected):</p> <ol style="list-style-type: none"> <li>1. Website display error message “invalid input”</li> <li>2. Repeat main flow</li> <li>3. End of flow</li> </ol> <p>Cancel:</p> <ol style="list-style-type: none"> <li>1. Organization clicks “cancel”</li> <li>2. Popup window close</li> <li>3. End of flow</li> </ol>

## Project Requirements

### Use Case Description 12

Use Case ID	UC12
Use Case Name	Organization manages user account information
Actor(s)	Organization
Goal	Organization edits their user account information
Pre-Condition	Organization has to be logged into their user account
Post-Condition	Organization user account information has been edited
Trigger(s)	Organization wants to change and update their user account information (e.g password)
Main Flow	<ol style="list-style-type: none"> <li>1. Organization clicks on the user account tab</li> <li>2. Organization clicks “manage info”</li> <li>3. Website allows text field to be editable</li> <li>4. Organization makes changes to their user info that they wanted to update</li> <li>5. Organization clicks “save” at the bottom right</li> <li>6. Website saves changes and update database</li> <li>7. Website refreshes and display newly updated info</li> <li>8. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Cancel:</p> <ol style="list-style-type: none"> <li>1. Organization clicks “cancel” at the top right</li> <li>2. Website disables text fields for editing</li> <li>3. End of flow</li> </ol> <p>Invalid Input:</p> <ol style="list-style-type: none"> <li>1. Organization sees error message next to text field saying “invalid input”</li> <li>2. Repeat main flow (from 4)</li> <li>3. End of flow</li> </ol>



*Figure 6: UC13, UC14, UC15*

## Project Requirements

### Use Case Description 13

Use Case ID	UC13
Use Case Name	Organization updates inputted data set
Actor(s)	Organization
Goal	Organization will be able to update and change the data set for RECS system to learn by uploading a new set
Pre-Condition	Organization must have already been logged into their account Organization must have already added a data set previously
Post-Condition	New Data set inputted into RECS system
Trigger(s)	Organization wants to change/update the previous data set
Main Flow	<ol style="list-style-type: none"> <li>1. Organization clicks on the “update” button</li> <li>2. Organization is popup appears prompting Organization to upload their new data set</li> <li>3. Organization clicks “add” button at the bottom of popup</li> <li>4. Website takes new data set and sends it to the RECS system</li> <li>5. RECS system will update itself by learning new data set</li> <li>6. Website redirects Organization back to their homepage with popup “successfully updated”</li> <li>7. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input(no data set selected):</p> <ol style="list-style-type: none"> <li>1. Website display error message “invalid input”</li> <li>2. Repeat main flow</li> <li>3. End of flow</li> </ol> <p>Cancel:</p> <ol style="list-style-type: none"> <li>1. Organization clicks “cancel”</li> <li>2. Popup window close</li> <li>3. End of flow</li> </ol>

## Project Requirements

### Use Case Description 14

Use Case ID	UC14
Use Case Name	Organization generates predicted rating
Actor(s)	Organization
Goal	Organization generates predicted rating for products based on the data set they input into RECS
Pre-Condition	Organization must have already been logged into their account Organization must have already added a data set into RECS
Post-Condition	RECS generates and return predicted rating for product requested by Organization
Trigger(s)	Organization wants generate and find predicted rating for products they input to RECS, based on their data set that they added
Main Flow	<ol style="list-style-type: none"> <li>1. Organization clicks on the “generate rating” tab</li> <li>2. Organization redirected to predicted ratings page</li> <li>3. Organization sees a text field prompting for Organization to search for a product</li> <li>4. Organization enters product they want to get predicted rating for</li> <li>5. Website sends request to RECS</li> <li>6. RECS processes request and returns prediction</li> <li>7. Website displays predicted rating for searched product</li> <li>8. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input:</p> <ol style="list-style-type: none"> <li>1. Organization sees error message next to text field “invalid input”</li> <li>2. Repeat main flow (from 4)</li> <li>3. End of flow</li> </ol>

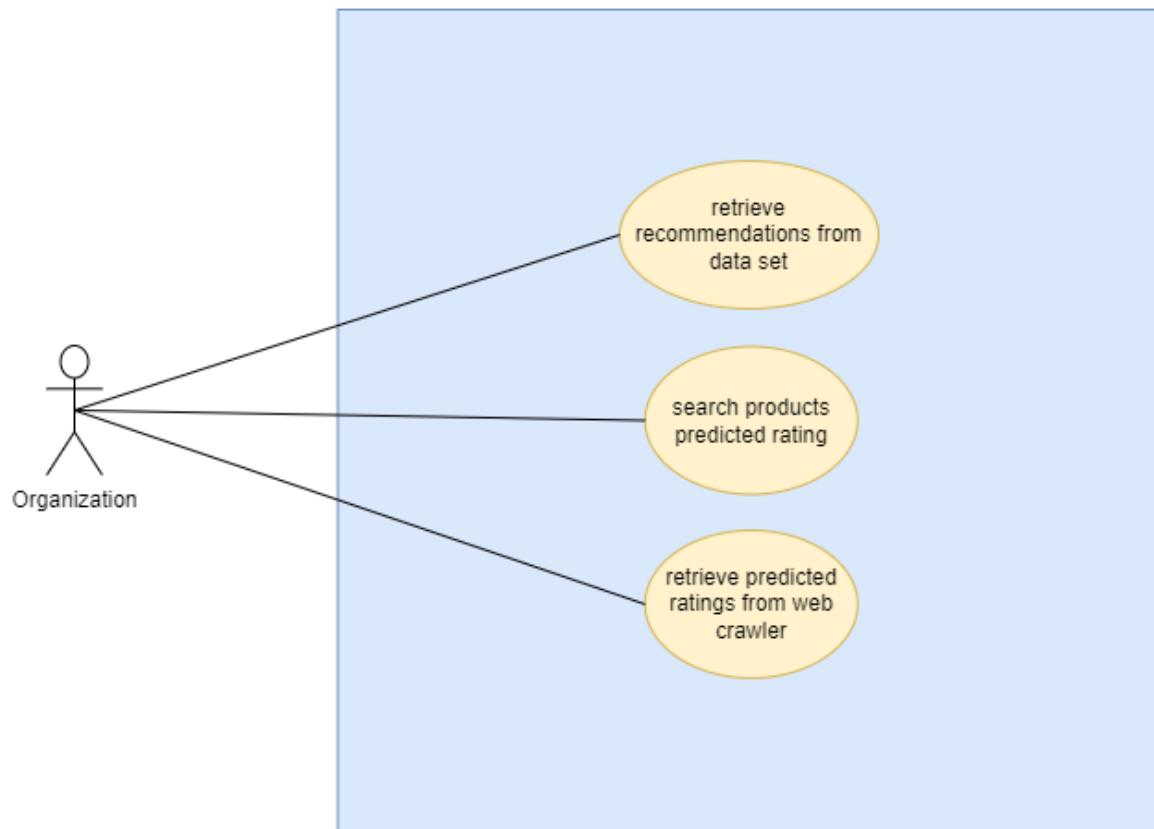


Figure 7: UC16, UC17 and UC18

## Project Requirements

### Use Case Description 15

Use Case ID	UC15
Use Case Name	Organization retrieve recommendations
Actor(s)	Organization
Goal	Organization retrieves recommendations for products that are similar to well-liked products in their data set to suggest to their customer RECS
Pre-Condition	Organization must have already been logged into their account Organization must have already added a data set into RECS
Post-Condition	RECS retrieves recommendations of products to suggest to their customer base
Trigger(s)	Organization wants get recommendations of products similar to the well-liked products in their given data set
Main Flow	<ol style="list-style-type: none"> <li>1. Organization clicks on the “recommendations” tab</li> <li>2. Organization redirected to predicted recommendations page</li> <li>3. Organization clicks “get recommendations” button</li> <li>4. Website sends request to RECS</li> <li>5. RECS processes request and returns recommendations</li> <li>6. Website displays recommendations</li> <li>7. End of flow</li> </ol>
Sub Flow	N.A
Exception	N.A

## Project Requirements

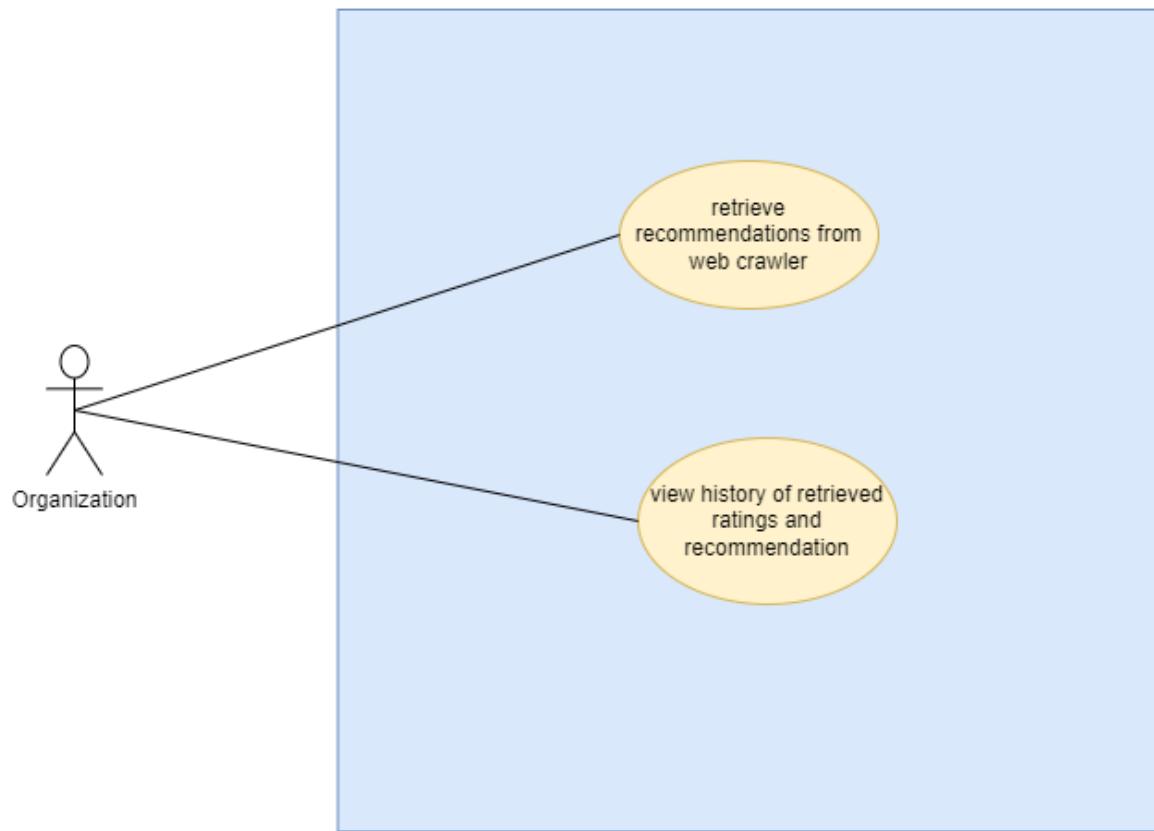
### Use Case Description 16

Use Case ID	UC16
Use Case Name	Organization search for product rating
Actor(s)	Organization
Goal	Organization should be able to input a search for predicted ratings of products
Pre-Condition	Organization is logged into their account
Post-Condition	Organization is able to find product they search for
Trigger	Organization wants to search for specific product and their predicted rating
Main Flow	<ol style="list-style-type: none"> <li>1. At their main homepage, Organization clicks tab “search”</li> <li>2. Website displays a search form</li> <li>3. Organization enters the product they want to search the predicted rating of</li> <li>4. Website processes and searches the database</li> <li>5. Organization sees the refreshed page displaying the searched product</li> <li>6. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>invalid input:</p> <ol style="list-style-type: none"> <li>1. Organization enters invalid input</li> <li>2. website displays error popup alert</li> <li>3. Organization closes the popup</li> <li>4. Organization repeats main flow</li> <li>5. End of flow</li> </ol>

## Project Requirements

### Use Case Description 17

Use Case ID	UC17
Use Case Name	Organization generates predicted rating from RECS web crawler
Actor(s)	Organization
Goal	Organization generates predicted rating for products based on RECS on web crawling service
Pre-Condition	Organization must have already been logged into their account
Post-Condition	RECS web crawler gathers data, process it and generate predicted rating for products the Organization requests
Trigger(s)	Organization wants generate and find predicted rating for products they input to RECS, based on their data set that they added
Main Flow	<ol style="list-style-type: none"> <li>1. Organization clicks on the “generate ratings” tab</li> <li>2. Organization redirected to predicted ratings page</li> <li>3. Organization clicks “crawl” button</li> <li>4. RECS crawls for data on various e-commerce sites</li> <li>5. Organization is notified “crawling done”</li> <li>6. Organization enters product they want to get predicted rating for in the text field</li> <li>7. Website sends request to RECS</li> <li>8. RECS processes request and returns prediction</li> <li>9. Website displays predicted rating for searched product</li> <li>10. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input:</p> <ol style="list-style-type: none"> <li>1. Organization sees error message next to text field “invalid input”</li> <li>2. Repeat main flow (from 4)</li> <li>3. End of flow</li> </ol>



*Figure 7: UC19 and UC20*

## Project Requirements

### Use Case Description 18

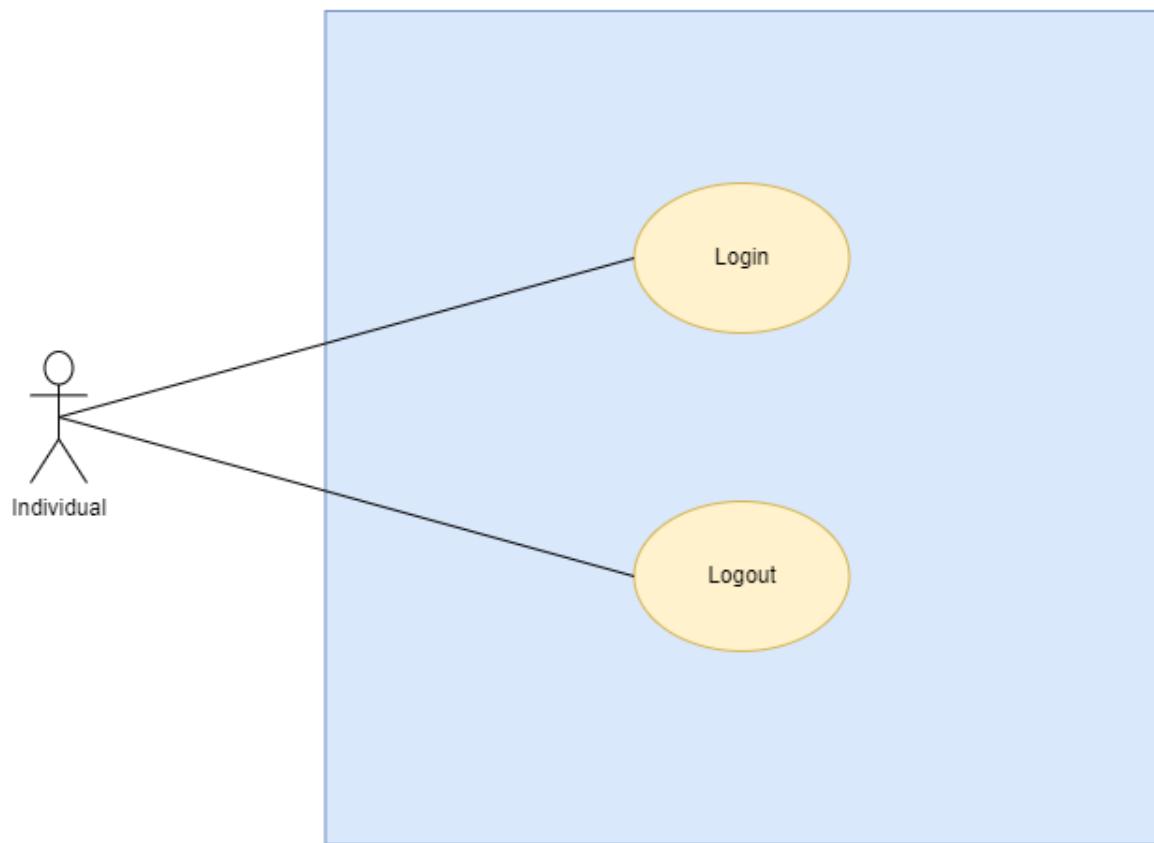
Use Case ID	UC18
Use Case Name	Organization retrieve recommendations from RECS web crawler
Actor(s)	Organization
Goal	Organization retrieves recommendations for products that are well-liked products found from RECS own web crawling
Pre-Condition	Organization must have already been logged into their account
Post-Condition	RECS retrieves recommendations of products to suggest to their customer base
Trigger(s)	Organization wants get recommendations of products that are well-liked products from RECS web crawling other e-commerce site
Main Flow	<ol style="list-style-type: none"> <li>1. Organization clicks on the “recommendations” tab</li> <li>2. Organization redirected to predicted recommendations page</li> <li>3. Organization clicks “crawl” button</li> <li>4. RECS crawls for data on various e-commerce sites</li> <li>5. Organization is notified “crawling done”</li> <li>6. Organization clicks “get recommendations” button</li> <li>7. Website sends request to RECS</li> <li>8. RECS processes request and returns recommendations</li> <li>9. End of flow</li> </ol>
Sub Flow	N.A
Exception	N.A

## Project Requirements

### Use Case Description 19

Use Case ID	UC19
Use Case Name	Organization view history
Actor(s)	Organization
Goal	View the history of the predicted ratings and/or recommendations of products that they retrieved previously
Pre-Condition	<p>Organization is logged into their account</p> <p>Organization has previously retrieved either a predicted rating of a product or retrieved recommended product</p>
Post-Condition	Organization all previously retrieved rating/recommendation
Trigger	Organization wants to check what product they had previously viewed the predicted ratings for and/or the recommended products they previously checked
Main Flow	<ol style="list-style-type: none"> <li>1. At their main homepage, Organization clicks tab "view history"</li> <li>2. Website process request and gets the necessary information from database</li> <li>3. Organization is brought to a different page</li> <li>4. History of all previous retrievals are displayed</li> <li>5. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>No previous retrieval:</p> <ol style="list-style-type: none"> <li>1. Organization clicks view history</li> <li>2. Organization is brought to view history page but its empty</li> <li>3. End of flow</li> </ol>

## Individual Use Cases



*Figure 8: UC21 and UC22*

## Project Requirements

### Use Case Description 20

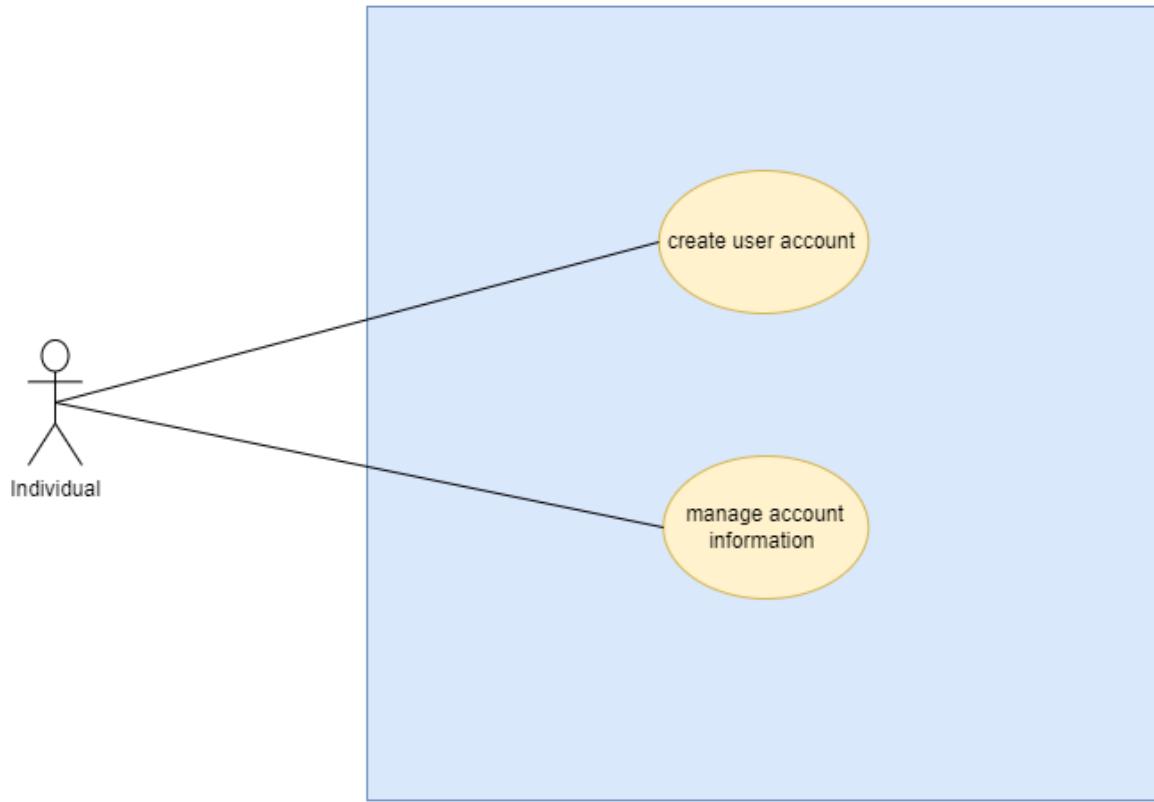
Use Case ID	UC20
Use Case Name	Individual login
Actor(s)	Individual
Goal	Individual able to log in and access their account
Pre-Condition	Individual already created an account
Post-Condition	Individual successfully logged in to the account
Trigger	Individual wants to login and access the service
Main Flow	<ol style="list-style-type: none"><li>1. Individual sees website display form asking for login credentials</li><li>2. Individual enters credentials</li><li>3. Clicks the “login” button</li><li>4. Website verifies the credentials</li><li>5. Individual will land on the homepage</li><li>6. End of flow</li></ol>
Sub Flow	N.A
Exception	<p>Invalid credentials:</p> <ol style="list-style-type: none"><li>1. Individual see error message pop up “Invalid credentials”</li><li>2. Individual closes the pop up</li><li>3. Individual re-enters their credentials</li><li>4. End of flow</li></ol>

## Project Requirements

### Use Case Description 21

Use Case ID	UC21
Use Case Name	Individual log out
Actor(s)	Individual
Goal	Individual is able to log out of their account
Pre-Condition	Individual already logged in their account
Post-Condition	Individual successfully logs out of their account
Trigger(s)	Individual wants to log out of their account to end their session
Main Flow	<ol style="list-style-type: none"><li>1. Individual clicks the “logout” button</li><li>2. website logs admin out</li><li>3. Individual will be redirected back to the login page</li><li>4. Session ended</li><li>5. End of flow</li></ol>
Sub Flow	N.A
Exception	N.A

## Project Requirements



*Figure 9: UC23 and UC24*

## Project Requirements

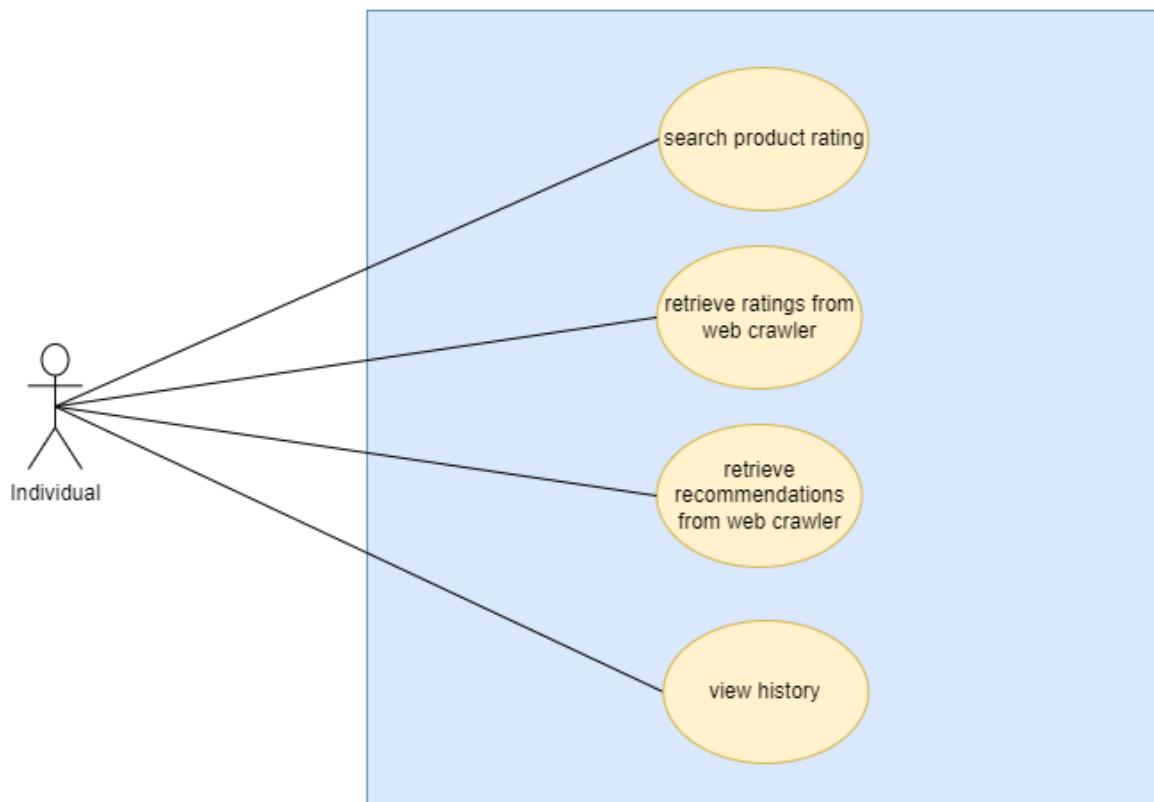
### Use Case Description 22

Use Case ID	UC22
Use Case Name	Individual creates user account
Actor(s)	Individual
Goal	Individual will be able to create their own user account
Pre-Condition	Individual is at the main landing page (login page)
Post-Condition	A new user Individual has been created
Trigger(s)	Individual creating a new user account for themselves
Main Flow	<ol style="list-style-type: none"> <li>1. Individual clicks on the “create new account” button at the login page</li> <li>2. A new account form is displayed by website</li> <li>3. Individual fills in their necessary information</li> <li>4. Individual makes credentials for them to use for logging in</li> <li>5. Individual clicks “create” button to submit</li> <li>6. The website validates and stores the new account and its information</li> <li>7. New user account has been created</li> <li>8. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input:</p> <ol style="list-style-type: none"> <li>1. Website display error pop up “invalid input”</li> <li>2. Individual closes pop up</li> <li>3. Repeat main flow</li> <li>4. End of flow</li> </ol>

## Project Requirements

### Use Case Description 23

Use Case ID	UC23
Use Case Name	Individual manages user account information
Actor(s)	Individual
Goal	Individual edits their user account information
Pre-Condition	Individual has to be logged into their user account
Post-Condition	Individual user account information has been edited
Trigger(s)	Individual wants to change and update their user account information (e.g password)
Main Flow	<ol style="list-style-type: none"> <li>1. Individual clicks on the user account tab</li> <li>2. Individual clicks “manage info”</li> <li>3. Website allows text field to be editable</li> <li>4. Individual makes changes to their user info that they wanted to update</li> <li>5. Individual clicks “save” at the bottom right</li> <li>6. Website saves changes and update database</li> <li>7. Website refreshes and display newly updated info</li> <li>8. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Cancel:</p> <ol style="list-style-type: none"> <li>1. Individual clicks “cancel” at the top right</li> <li>2. Website disables text fields for editing</li> <li>3. End of flow</li> </ol> <p>Invalid Input:</p> <ol style="list-style-type: none"> <li>1. Individual sees error message next to text field saying “invalid input”</li> <li>2. Repeat main flow (from 4)</li> <li>3. End of flow</li> </ol>



*Figure 9: UC25, UC26 and UC27*

## Project Requirements

### Use Case Description 24

Use Case ID	UC24
Use Case Name	Individual search for product rating
Actor(s)	Individual
Goal	Individual should be able to input a search for predicted ratings of products
Pre-Condition	Individual is logged into their account
Post-Condition	Individual is able to find product they search for
Trigger	Individual wants to search for specific product and their predicted rating
Main Flow	<ol style="list-style-type: none"> <li>1. At their main homepage, Individual clicks tab “search”</li> <li>2. Website displays a search form</li> <li>3. Individual enters the product they want to search the predicted rating of</li> <li>4. Website processes and searches the database</li> <li>5. Individual sees the refreshed page displaying the searched product</li> <li>6. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>invalid input:</p> <ol style="list-style-type: none"> <li>1. Individual enters invalid input</li> <li>2. website displays error popup alert</li> <li>3. Individual closes the popup</li> <li>4. Individual repeats main flow</li> <li>5. End of flow</li> </ol>

## Project Requirements

### Use Case Description 25

Use Case ID	UC25
Use Case Name	Individual generates predicted rating from RECS web crawler
Actor(s)	Individual
Goal	Individual generates predicted rating for products based on RECS on web crawling service
Pre-Condition	Individual must have already been logged into their account
Post-Condition	RECS web crawler gathers data, process it and generate predicted rating for products the Individual requests
Trigger(s)	Individual wants generate and find predicted rating for products they input to RECS, based on their data set that they added
Main Flow	<ol style="list-style-type: none"> <li>1. Individual clicks on the “generate ratings” tab</li> <li>2. Individual redirected to predicted ratings page</li> <li>3. Individual clicks “crawl” button</li> <li>4. RECS crawls for data on various e-commerce sites</li> <li>5. Individual is notified “crawling done”</li> <li>6. Individual enters product they want to get predicted rating for in the text field</li> <li>7. Website sends request to RECS</li> <li>8. RECS processes request and returns prediction</li> <li>9. Website displays predicted rating for searched product</li> <li>10. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>Invalid input:</p> <ol style="list-style-type: none"> <li>1. Individual sees error message next to text field “invalid input”</li> <li>2. Repeat main flow (from 4)</li> <li>3. End of flow</li> </ol>

## Project Requirements

### Use Case Description 26

Use Case ID	UC26
Use Case Name	Individual retrieve recommendations from RECS web crawler
Actor(s)	Individual
Goal	Individual retrieves recommendations for products that are well-liked products found from RECS own web crawling
Pre-Condition	Individual must have already been logged into their account
Post-Condition	RECS retrieves recommendations of products to suggest to their Individual base
Trigger(s)	Individual wants get recommendations of products that are well-liked products from RECS web crawling other e-commerce site
Main Flow	<ol style="list-style-type: none"> <li>1. Individual clicks on the “recommendations” tab</li> <li>2. Individual redirected to predicted recommendations page</li> <li>3. Individual clicks “crawl” button</li> <li>4. RECS crawls for data on various e-commerce sites</li> <li>5. Individual is notified “crawling done”</li> <li>6. Individual clicks “get recommendations” button</li> <li>7. Website sends request to RECS</li> <li>8. RECS processes request and returns recommendations</li> <li>9. End of flow</li> </ol>
Sub Flow	N.A
Exception	N.A

## Project Requirements

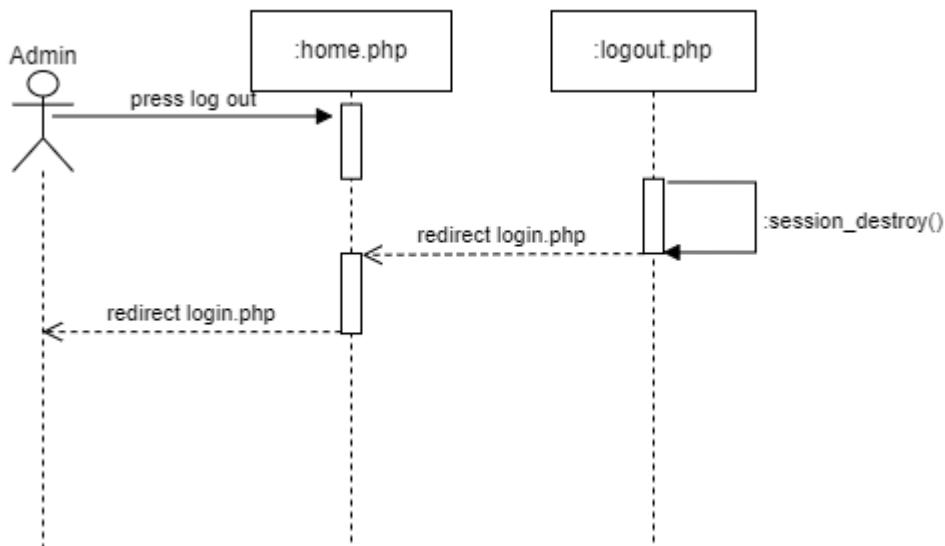
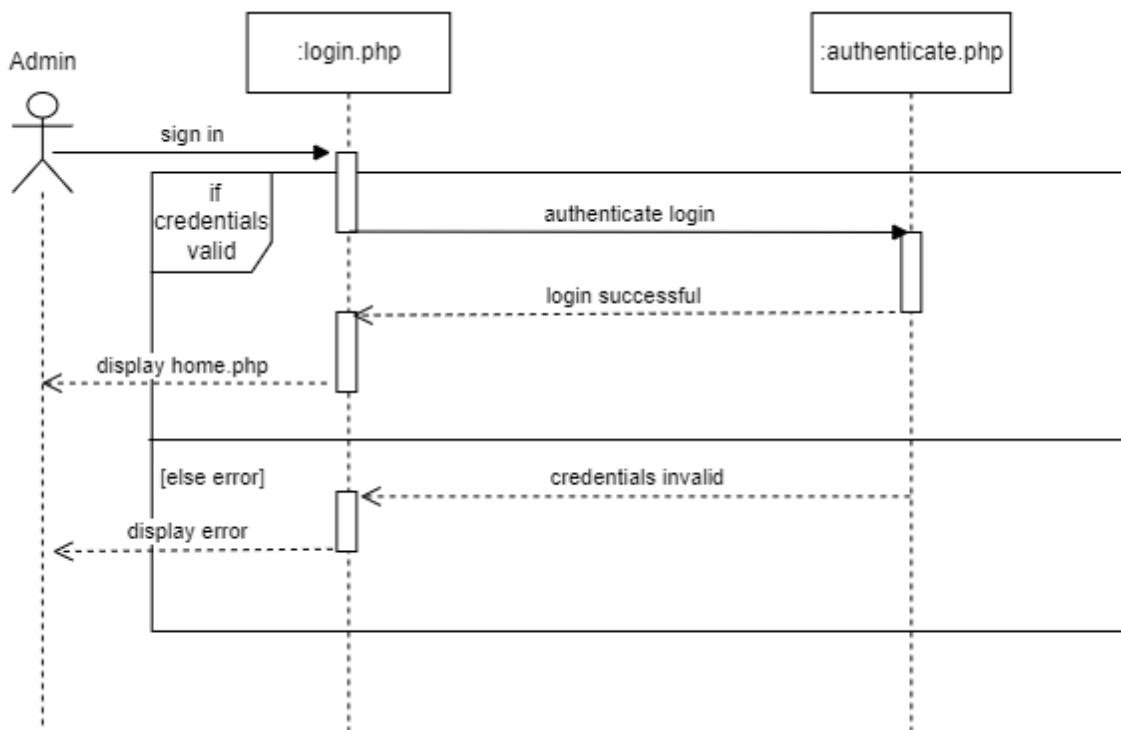
### Use Case Description 27

Use Case ID	UC27
Use Case Name	Individual view history
Actor(s)	Individual
Goal	View the history of the predicted ratings and/or recommendations of products that they retrieved previously
Pre-Condition	Individual is logged into their account  Individual has previously retrieved either a predicted rating of a product or retrieved recommended product
Post-Condition	Individual previously retrieved rating/recommendation
Trigger	Individual wants to check what product they had previously viewed the predicted ratings for and/or the recommended products they previously checked
Main Flow	<ol style="list-style-type: none"> <li>1. At their main homepage, Individual clicks tab "view history"</li> <li>2. Website process request and gets the necessary information from database</li> <li>3. Individual is brought to a different page</li> <li>4. History of all previous retrievals are displayed</li> <li>5. End of flow</li> </ol>
Sub Flow	N.A
Exception	<p>No previous retrieval:</p> <ol style="list-style-type: none"> <li>1. Individual clicks view history</li> <li>2. Individual is brought to view history page but its empty</li> <li>3. End of flow</li> </ol>

## Sequence Diagrams

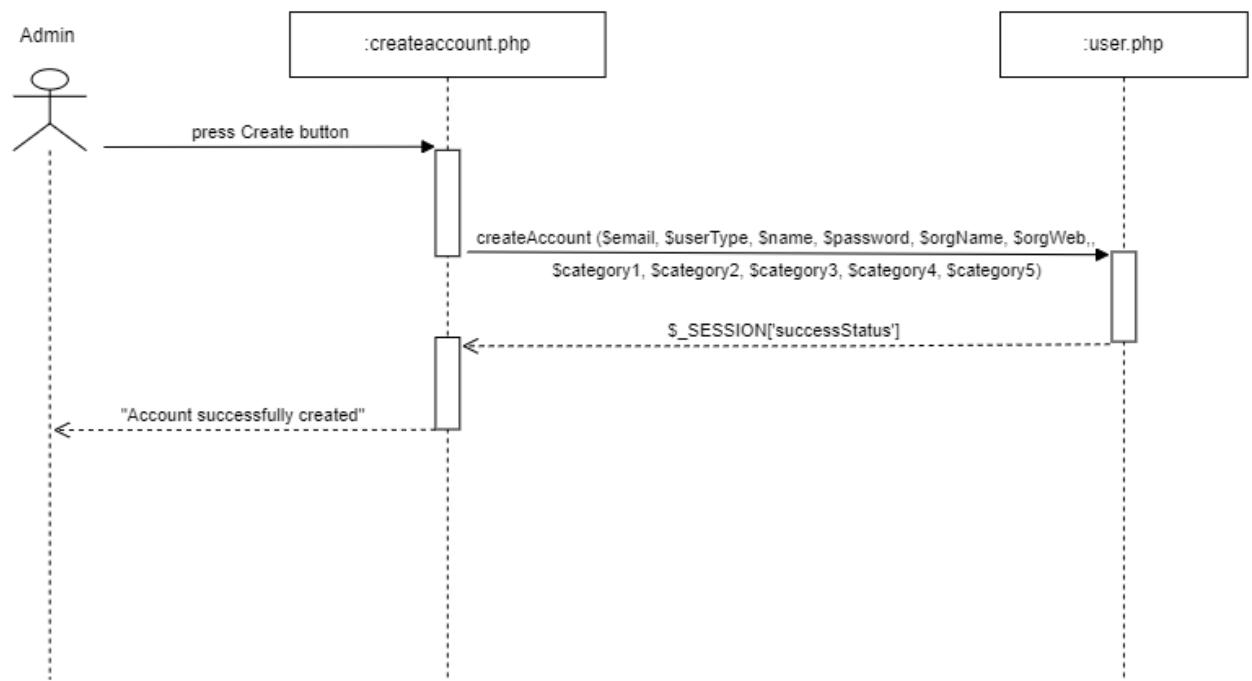
### Admin

UC 1 & 2 (login/logout)

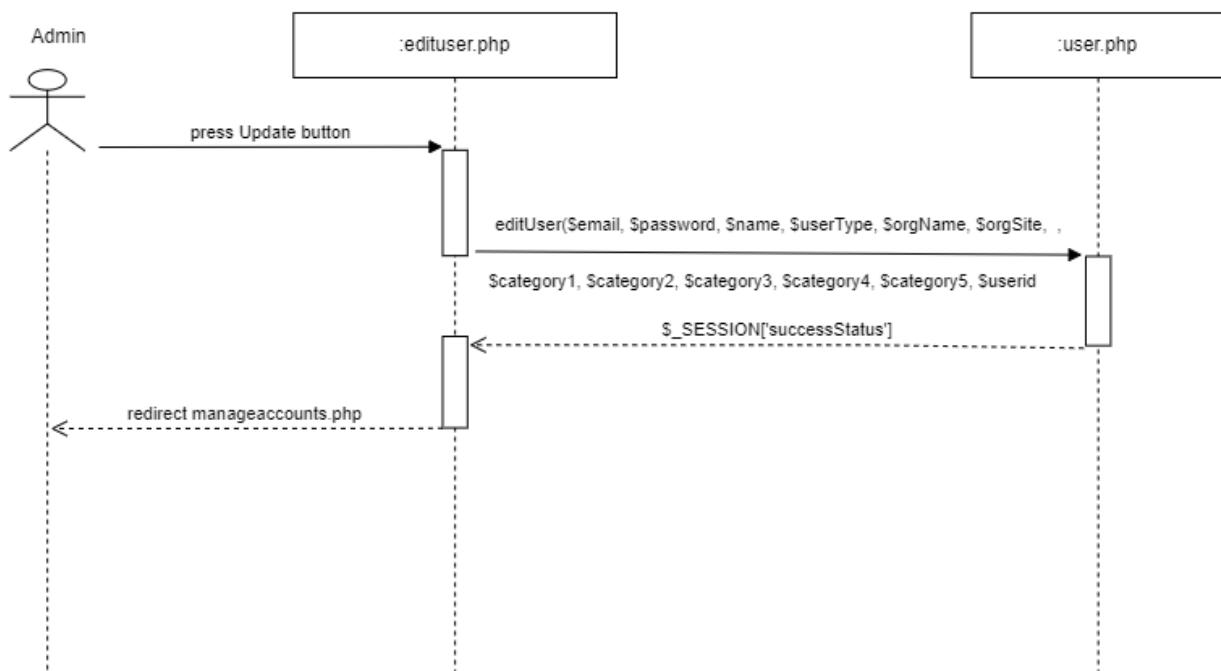


## Project Requirements

### UC 3 (create users)

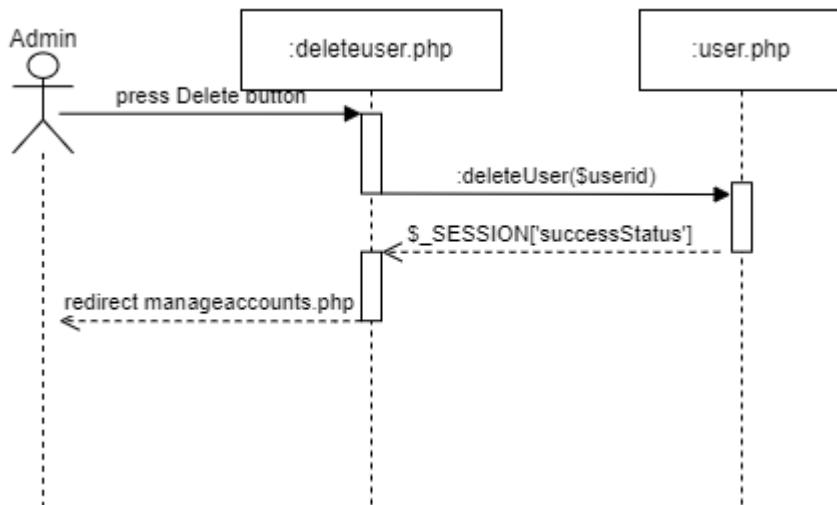


### UC 4 (edit user account)

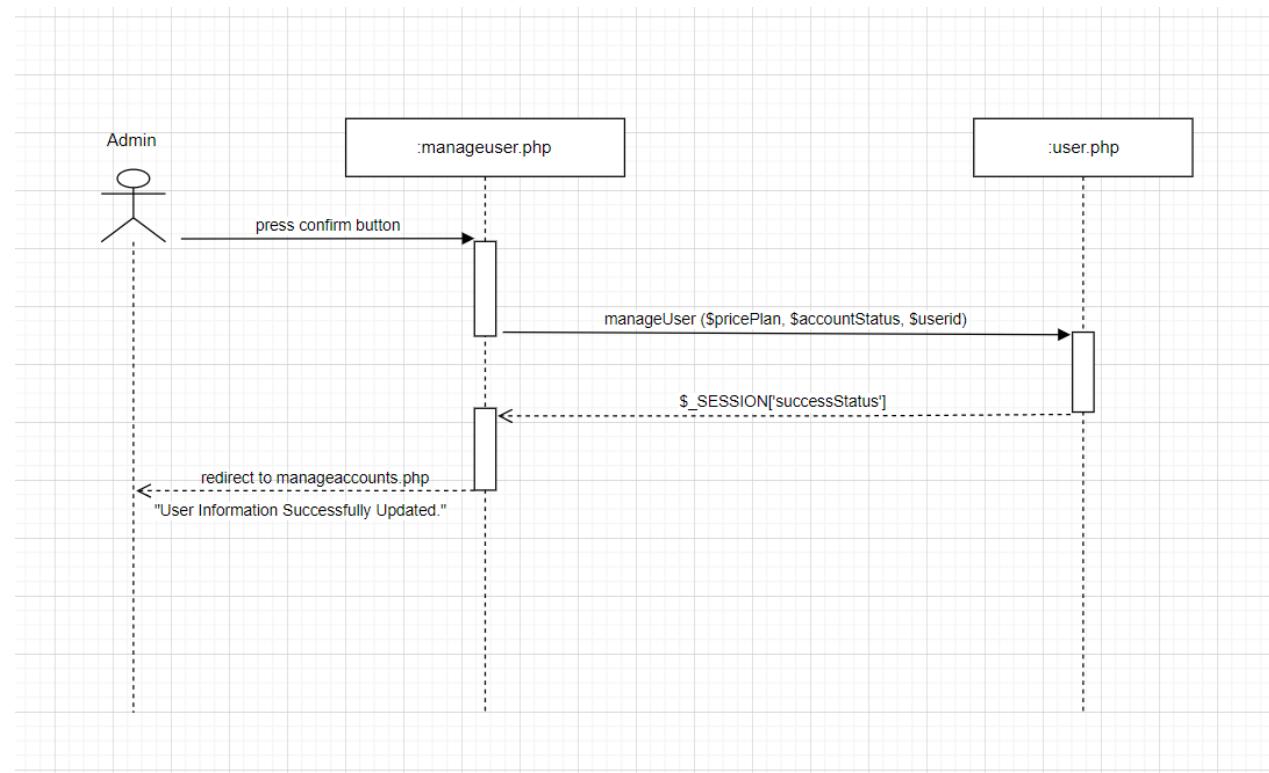


## Project Requirements

### UC 5 (delete users)

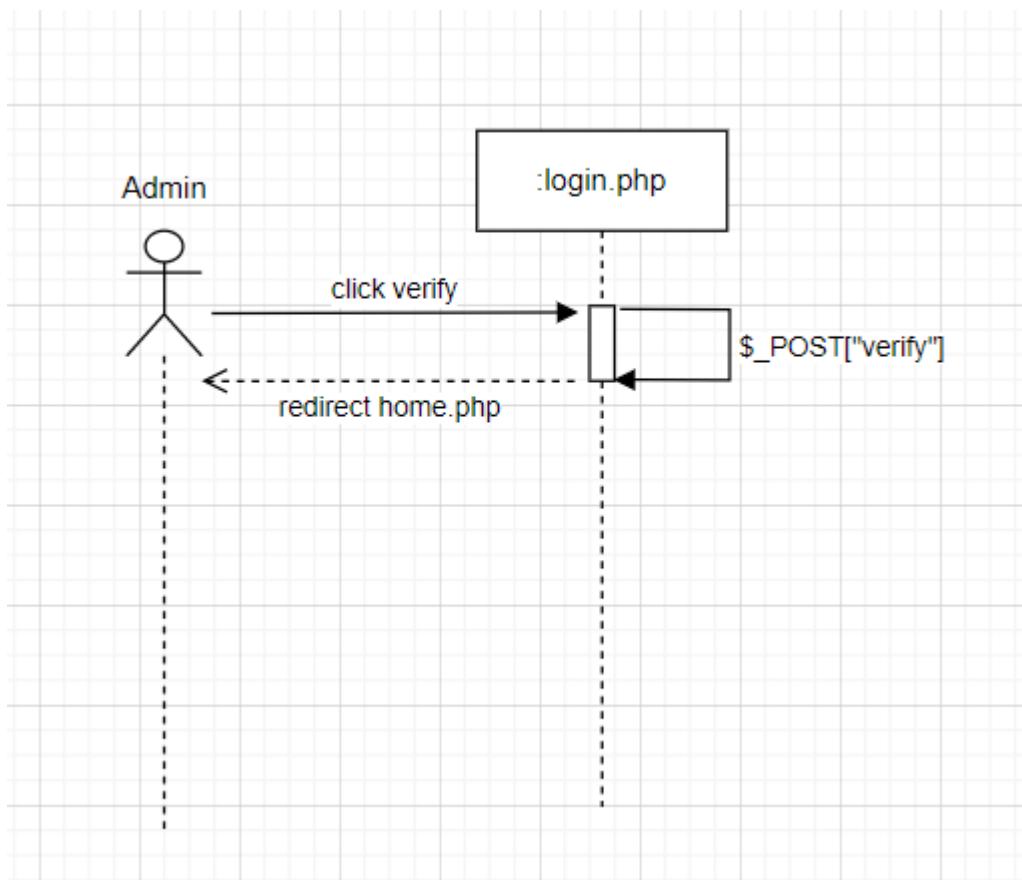


### UC 6 (Manage account status)



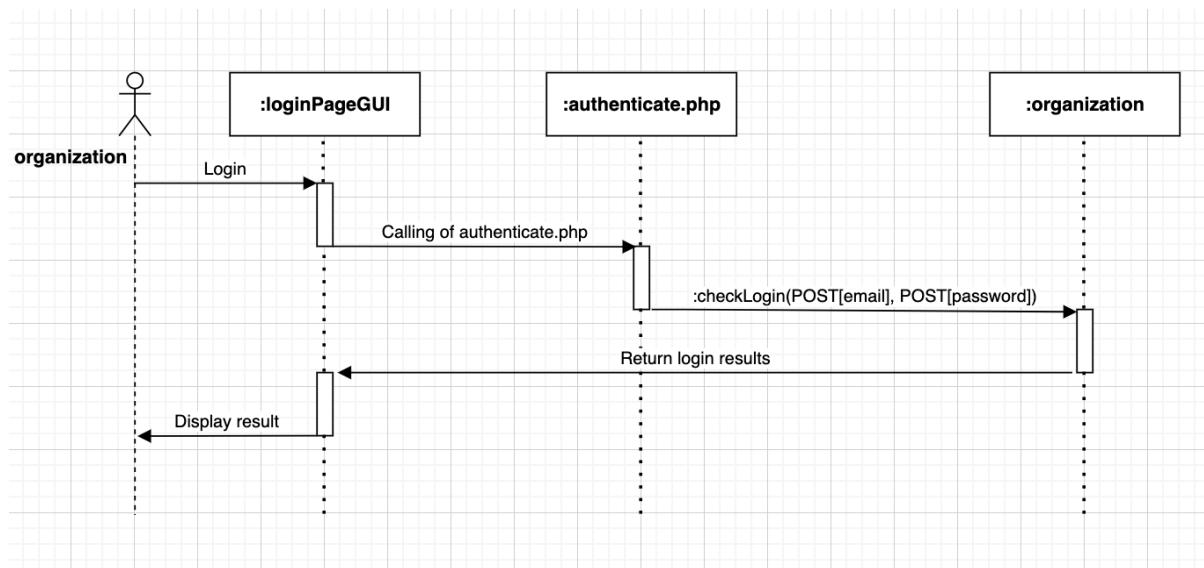
## Project Requirements

### UC 8 (2FA login)

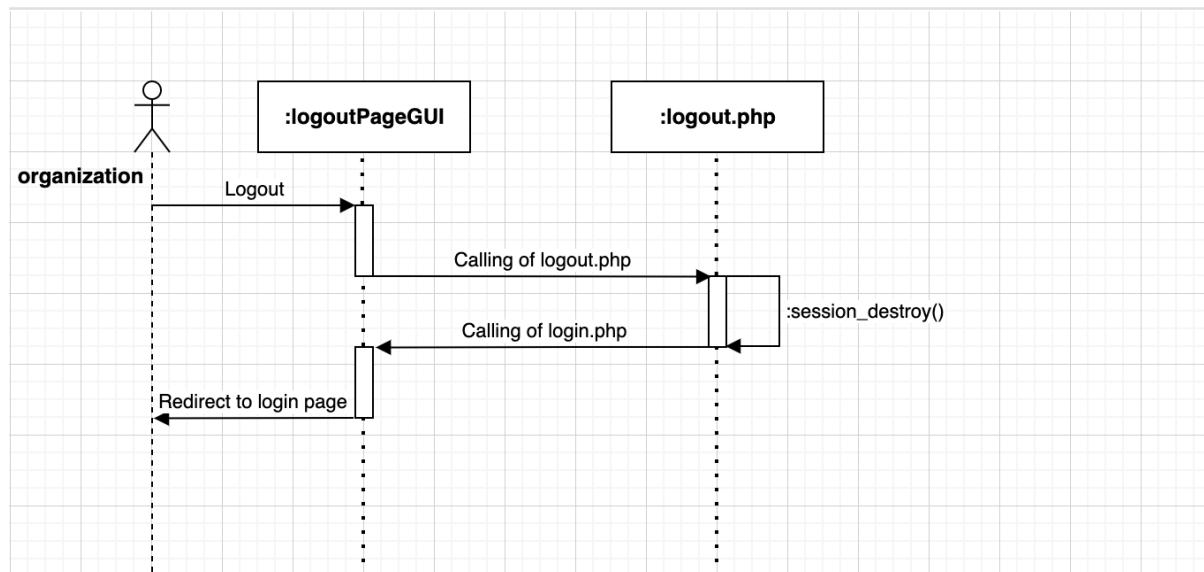


## Organization

### UC 9 (Login)

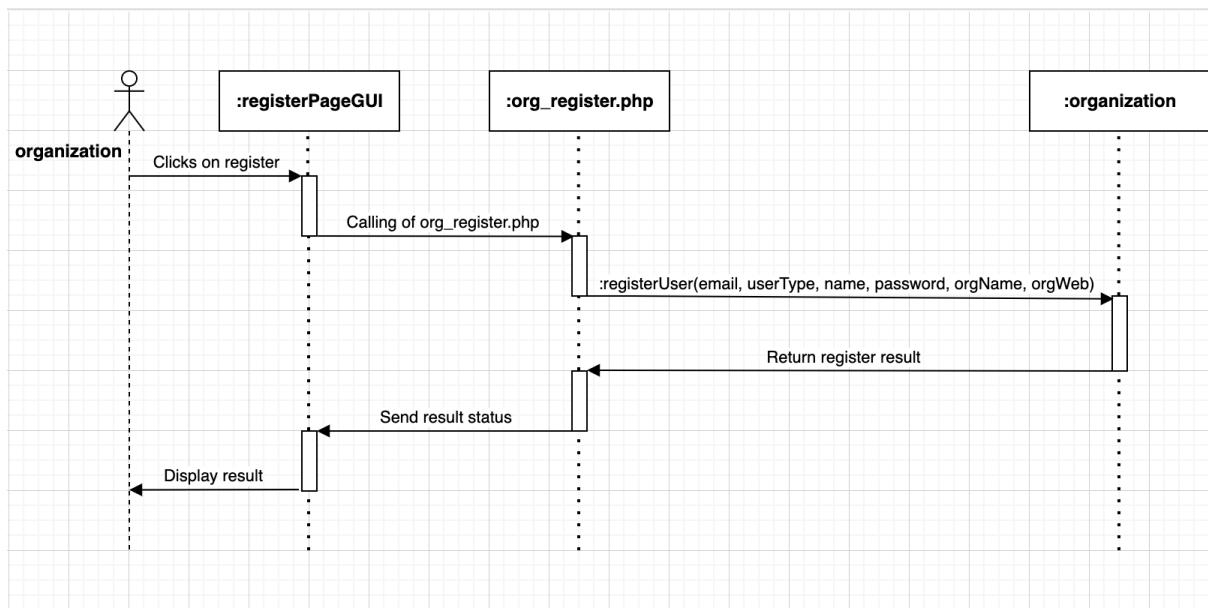


### UC 10 (Logout)

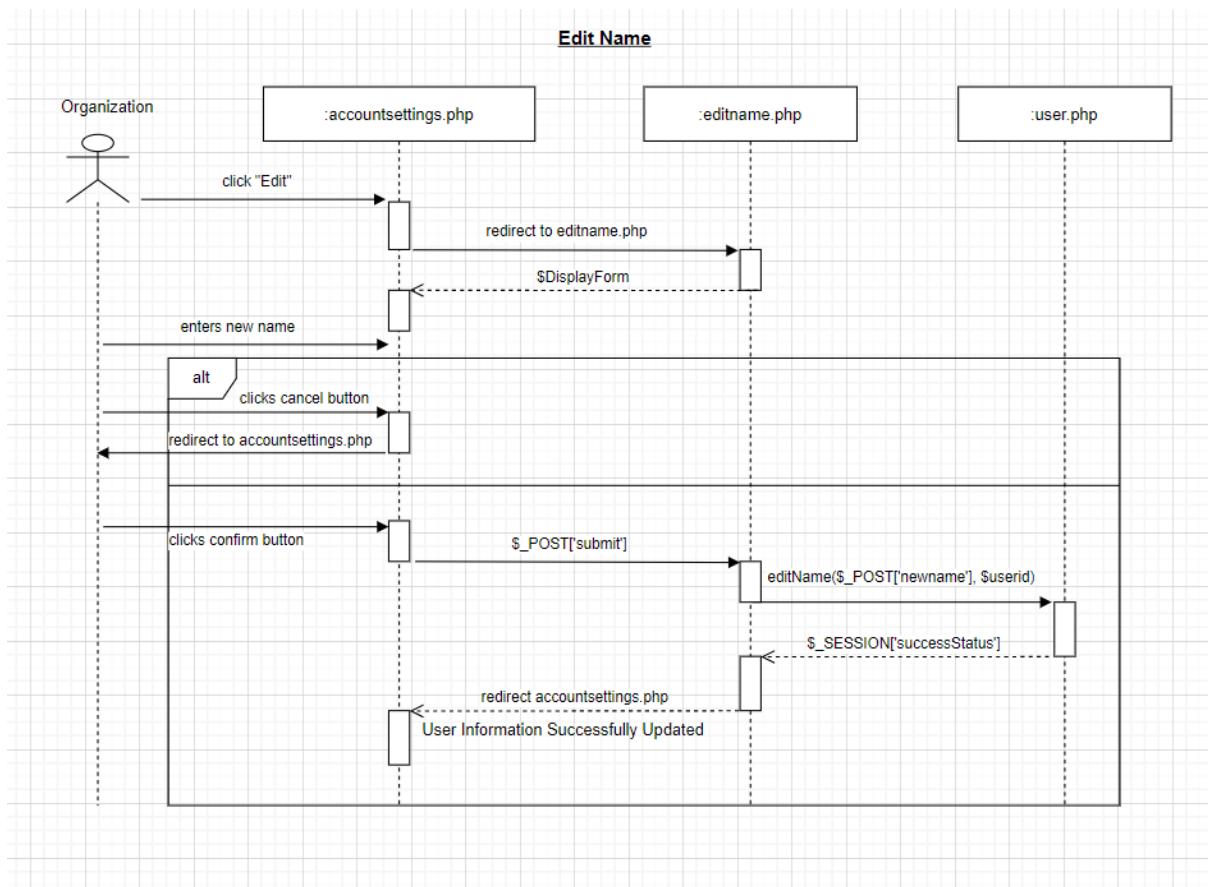


## Project Requirements

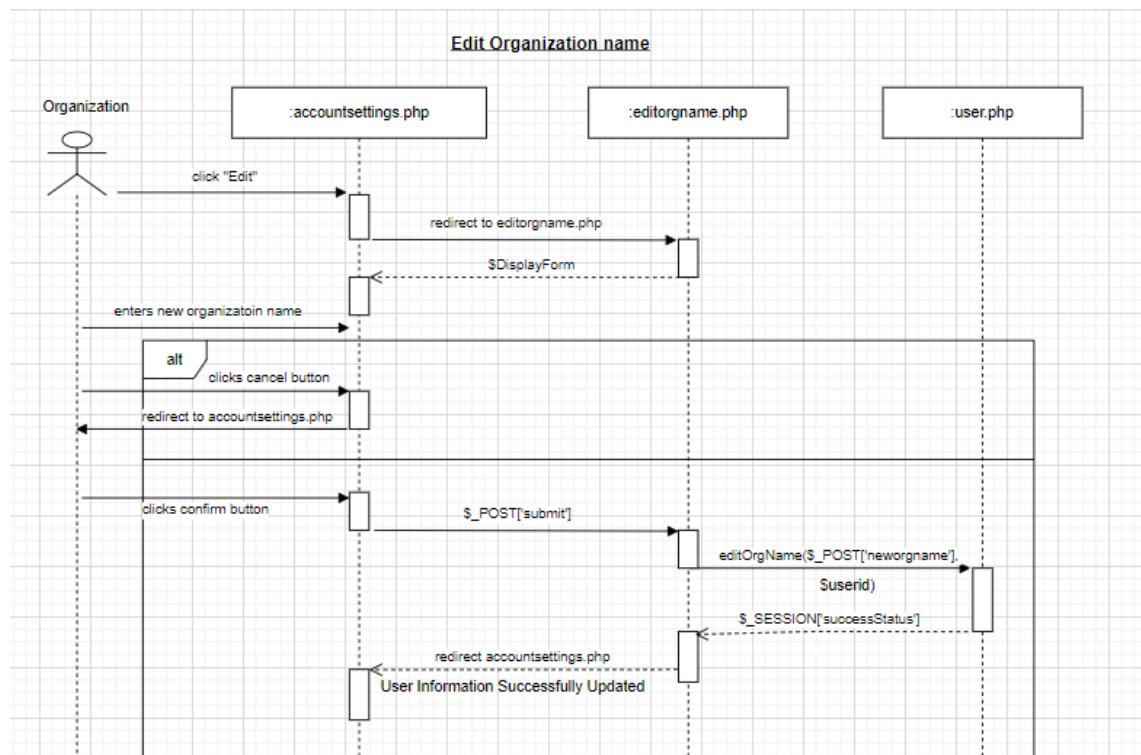
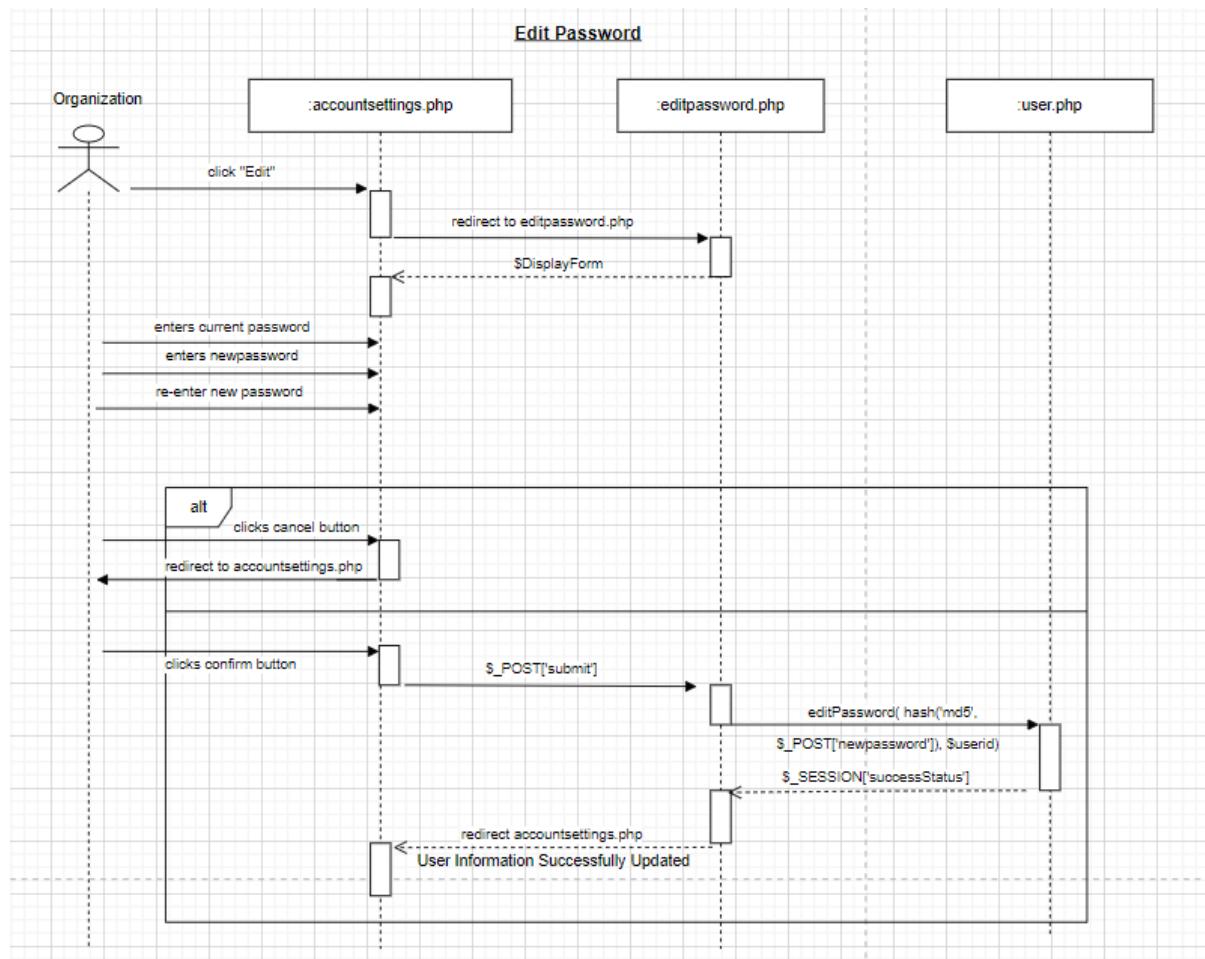
### UC 11 (create account)



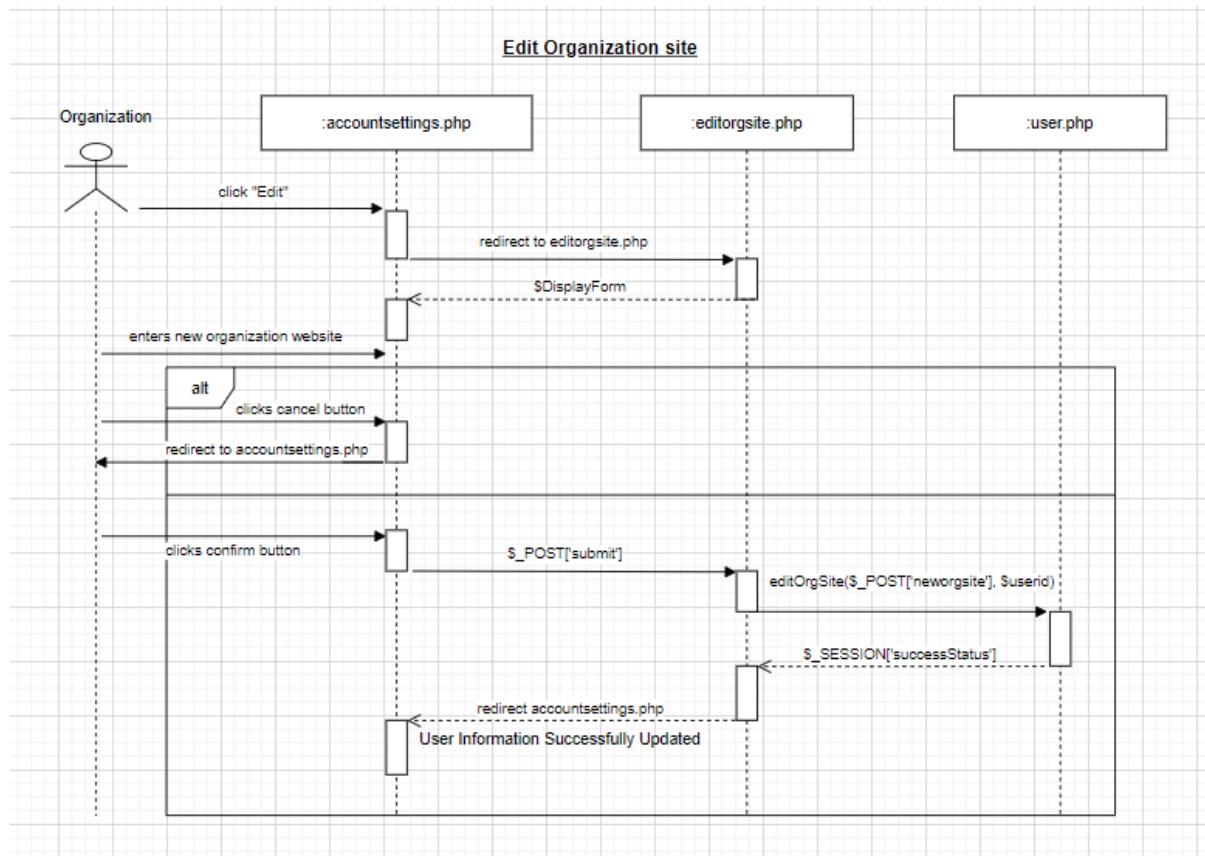
### UC 12 (edit account)



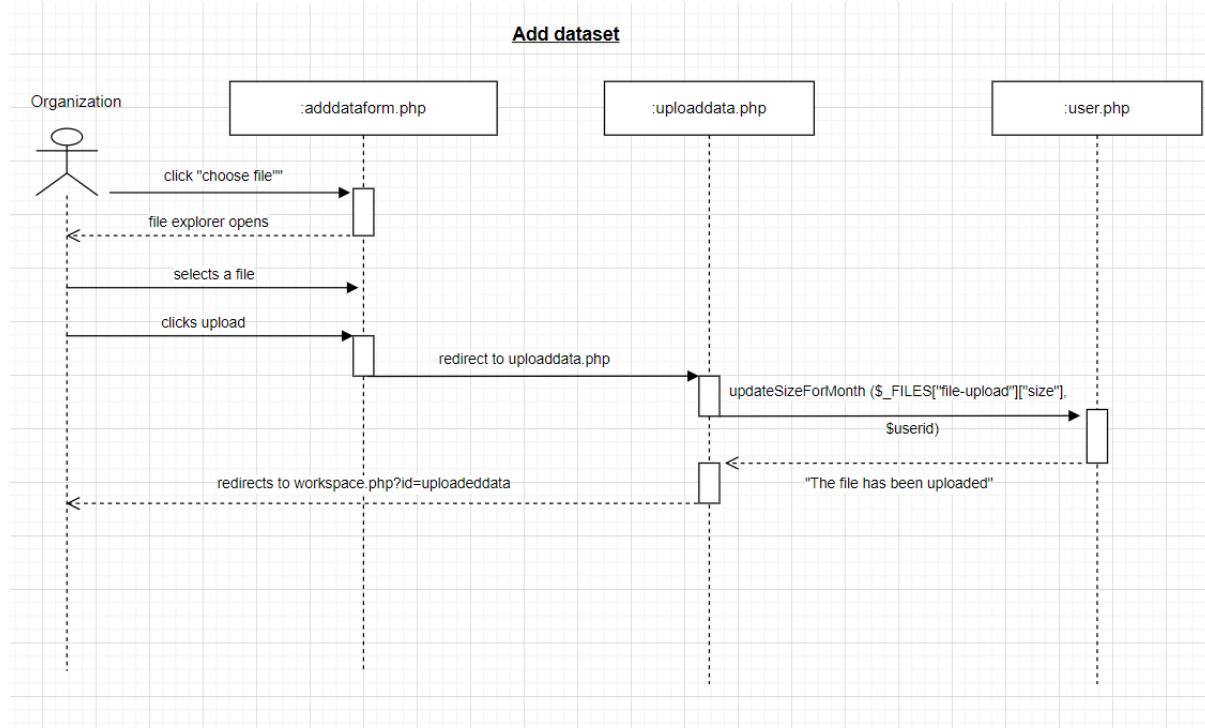
## Project Requirements



## Project Requirements

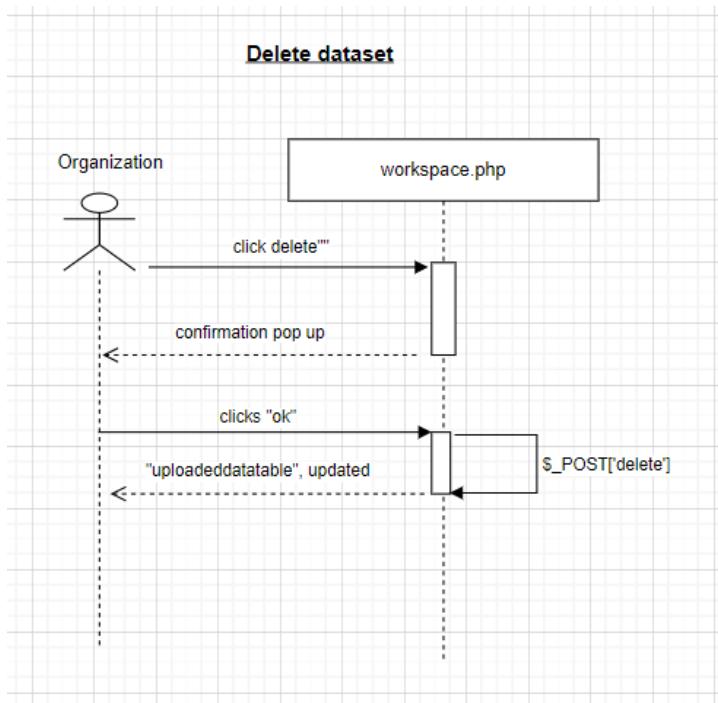


## UC 13



## Project Requirements

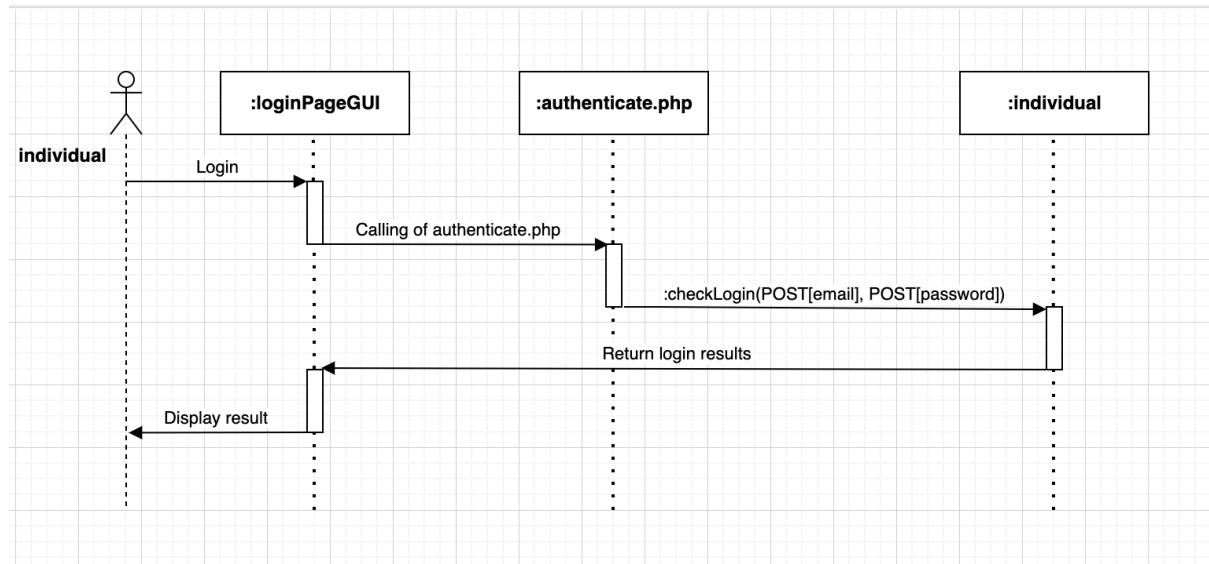
### UC 14 (delete dataset)



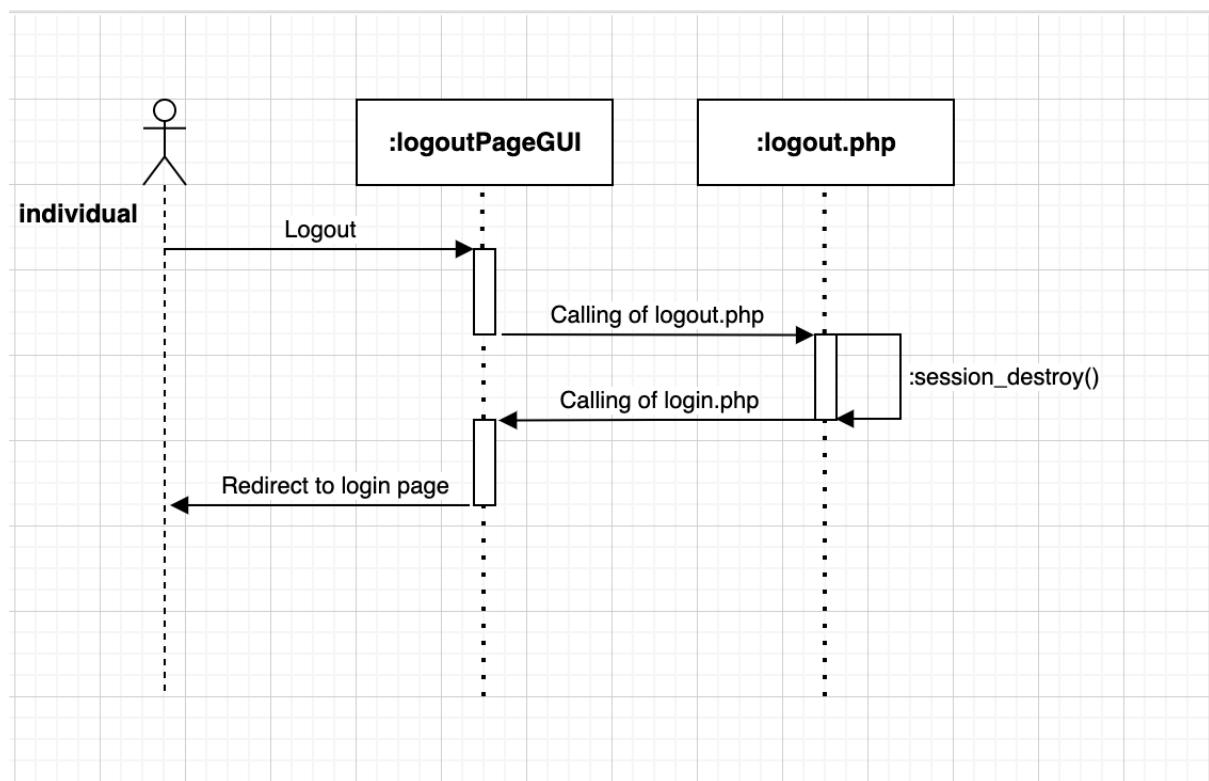
### UC 15 (Use RECS' data generate rating)

## Individual

### Login

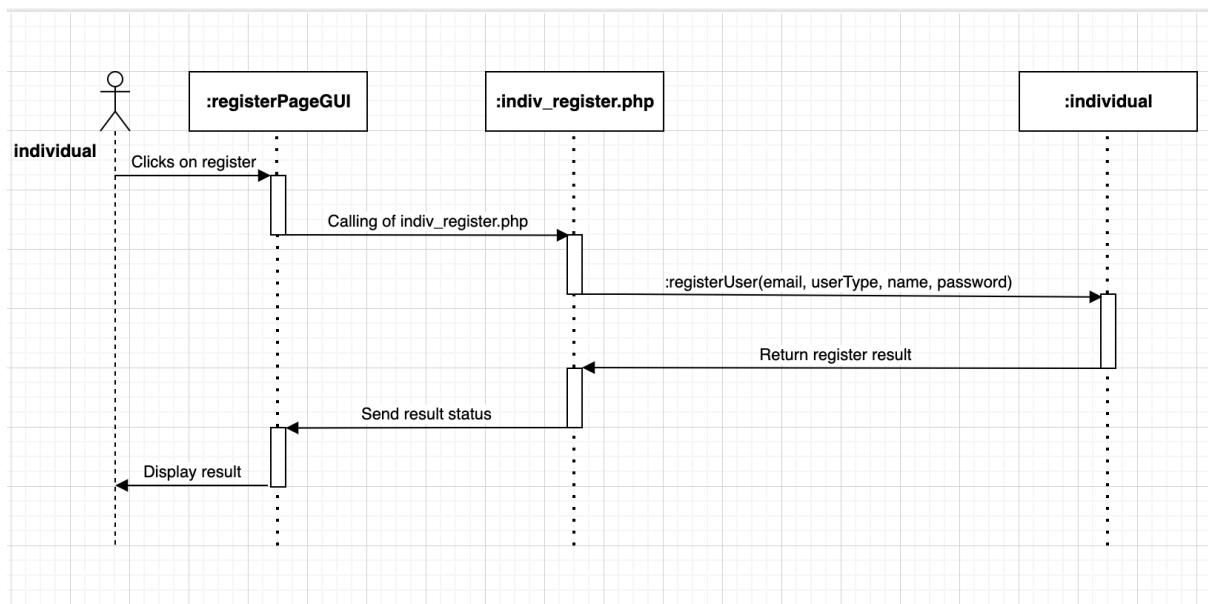


### Logout



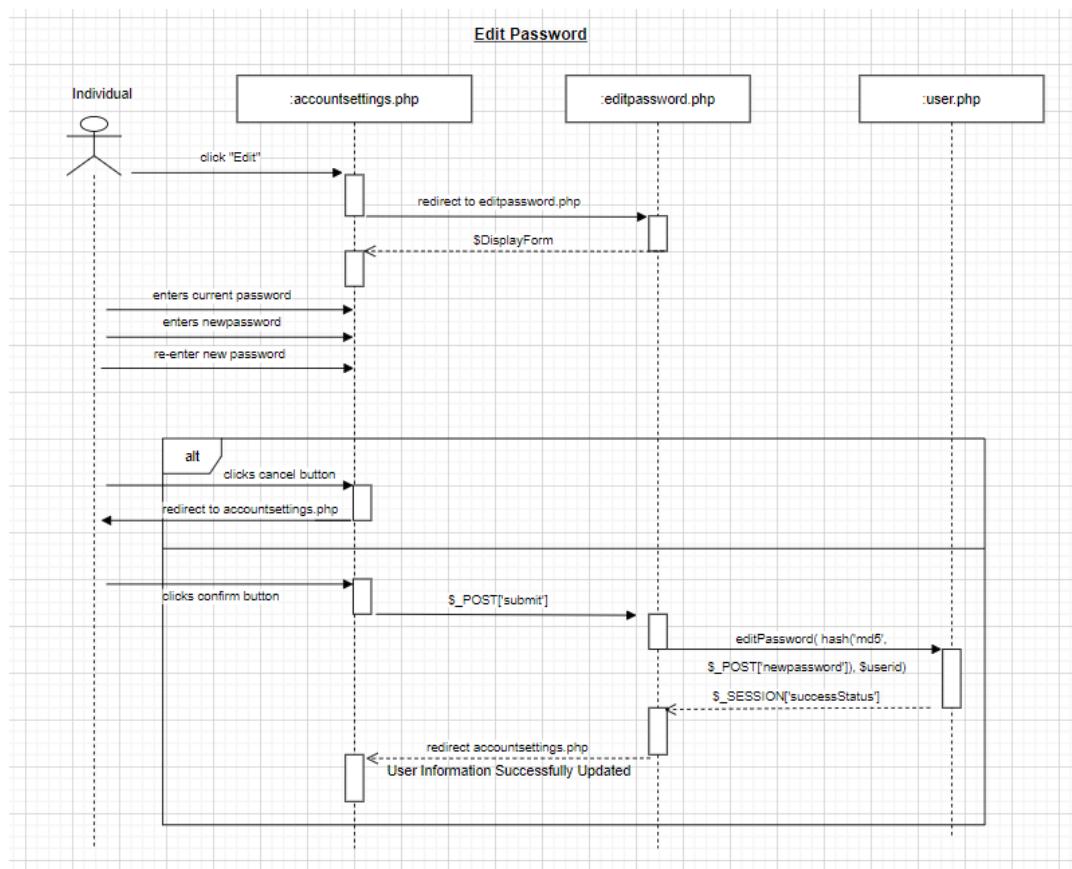
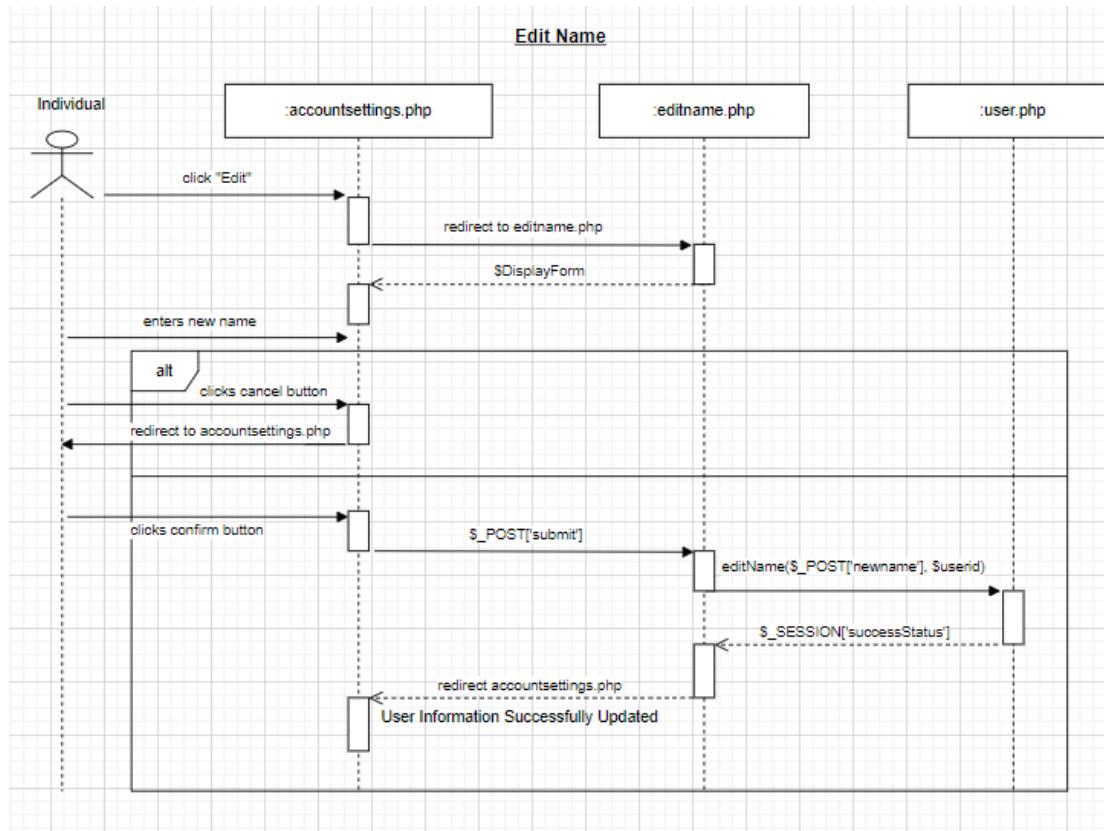
## Project Requirements

### UC (create account)



## Project Requirements

### UC (Edit account)



## Non-Functional Requirements

#	Requirement Description	Priority
1	Performance	High
2	Accuracy	High
3	Reliability	High
4	Security	High
5	Usability	High
6	Maintainability	High
7	Availability	Medium
8	Compatibility	Medium
9	Portability	Medium
10	Scalability	Low

## **Performance**

The Recommender System should be able to process a large load of 10, 000 data in less than a minute. Website dashboard should be able to handle requests in under 5 seconds.

## **Accuracy**

Recommender system should be able to predict user ratings of products at an accuracy of 70%.

## **Reliability**

Recommender system can perform normally even after prolonged usage (up to 1 hour).

## **Security**

Ensuring that all accounts and the account information are properly secured and only accessible by the respective owners of the account and administrator. 2 factor authentication to ensure the owner of the account is the one accessing. Database will be password encrypted to prevent any unauthorized access or leakage of data.

## **Usability**

Clients/users with an account should be able to easily access and view the website dashboard for rating predictions (of products) and recommendations on similar products to promote to their customers. The input of data set into the recommender system should also be simple, making it easy to use.

## **Maintainability**

The mean time to restore the system (MTTR) in the event of a system failure will not be greater than an hour.

## **Availability**

The recommender system and website dashboard will be available to clients/users 99% of the time outside of maintenance periods.

## **Compatibility**

System should be compatible across Mac OS and Linux. System is able to run across different browsers (chrome, safari, firefox).

## **Portability**

System is able to run on windows 10 and should be able to run on windows 11 without any effect on its performance.

## **Scalability**

The Recommender system should be scalable enough to support up to 10,000 visits at the same time while maintaining optimal performance.

## Interface Requirements

### **User Interface**

Our recommender system comes with a simple design and acts as a web dashboard for which the user can use to view the predicted ratings of products and get recommendations for products they can promote to their customers. When they first land on the webpage they will see two textboxes for them to enter username and password, or for new clients/users they can create a new account by clicking sign up in the top right corner of the landing page. The web dashboard they access after login will have a menu accessible at the top of the website to allow for ease of navigation to other pages. The web dashboard will also allow the client/user to easily add a new data set for the recommender system to use as training data with the aid of a simple add button easily located at the top of the web page on the right corner.

### **Software Interface**

Operating System: Windows 10, macOS, linux

Database: MySQL

Front End: PHP

WebCrawler: Scrapy / Python

Back End: Python

### **Communications Interface**

The website dashboard will be accessible to all browsers that support PHP, HTTP/HTTPS

### **Hardware Interface**

The website dashboard does not require any additional or specific hardware as it is able to run on computers as they are.

## Wireframe

### Admin

UC1:

<b>LOGIN</b>	
Email:	<input type="text"/>
Password:	<input type="password"/>
Forgot Password?	
<input type="button" value="Sign In"/>	

UC2:

User (Initials)
<input type="button" value="Log Out"/>

**UC3:**

	Create Accounts	
--	-----------------	--

**Create Account**

Individual

Email:

Password:

Confirm Password:

Name:

	Create Accounts	
--	-----------------	--

**Create Account**

Admin

Email:

Password:

Confirm Password:

Name:

**UC4:**

	Manage Accounts							
<b>Manage Accounts</b>								
<table border="1"><tr><td></td><td style="text-align: right;">Edit</td></tr><tr><td></td><td style="text-align: right;">Edit</td></tr><tr><td></td><td style="text-align: right;">Edit</td></tr></table>				Edit		Edit		Edit
	Edit							
	Edit							
	Edit							

	Manage Accounts	
<u>Edit User</u>		
User ID:	Organization Name:	
<input type="text"/>	<input type="text"/>	
Email:	Organization Website:	
<input type="text"/>	<input type="text"/>	
Password:		
<input type="text"/>		
Confirm Password:		
<input type="text"/>		
Name:		
<input type="text"/>		
User Type:	<input type="button" value="▼"/>	
	<input type="button" value="Cancel"/>	<input type="button" value="Update"/>

## Project Requirements

UC5:

	Manage Accounts	
<h2>Manage Accounts</h2>		
		Delete
		Delete
		Delete

	Manage Accounts	
<h2>Delete User</h2>		
User ID:	<input type="text"/>	
Email:	<input type="text"/>	
Name:	<input type="text"/>	
Organization Name:	<input type="text"/>	
Organization Website:	<input type="text"/>	
<input type="button" value="Cancel"/> <input type="button" value="Delete"/>		

## Project Requirements

### UC6:

	Manage Accounts	
--	-----------------	--

**Manage Accounts**

Manage
Manage
Manage

	Manage Accounts	
--	-----------------	--

**Manage User**

User ID:

Email:

Name:

User Type:

Price Plan:  
 ▽

Account Status:  
 ▽

## Project Requirements

### **UC7(Replace manage website contents for now, this is search UC):**

	Manage Accounts				
Manage Accounts					
	Search				
<table border="1"><tr><td></td></tr><tr><td></td></tr><tr><td></td></tr></table>					

## Organisation

**UC8:**

<b>LOGIN</b>	
Email:	<input type="text"/>
Password:	<input type="password"/>
Forgot Password?	
<input type="button" value="Sign In"/>	

**UC9:**

User (Initials)
<input type="text"/>
<input type="button" value="Log Out"/>

## Project Requirements

### UC10:

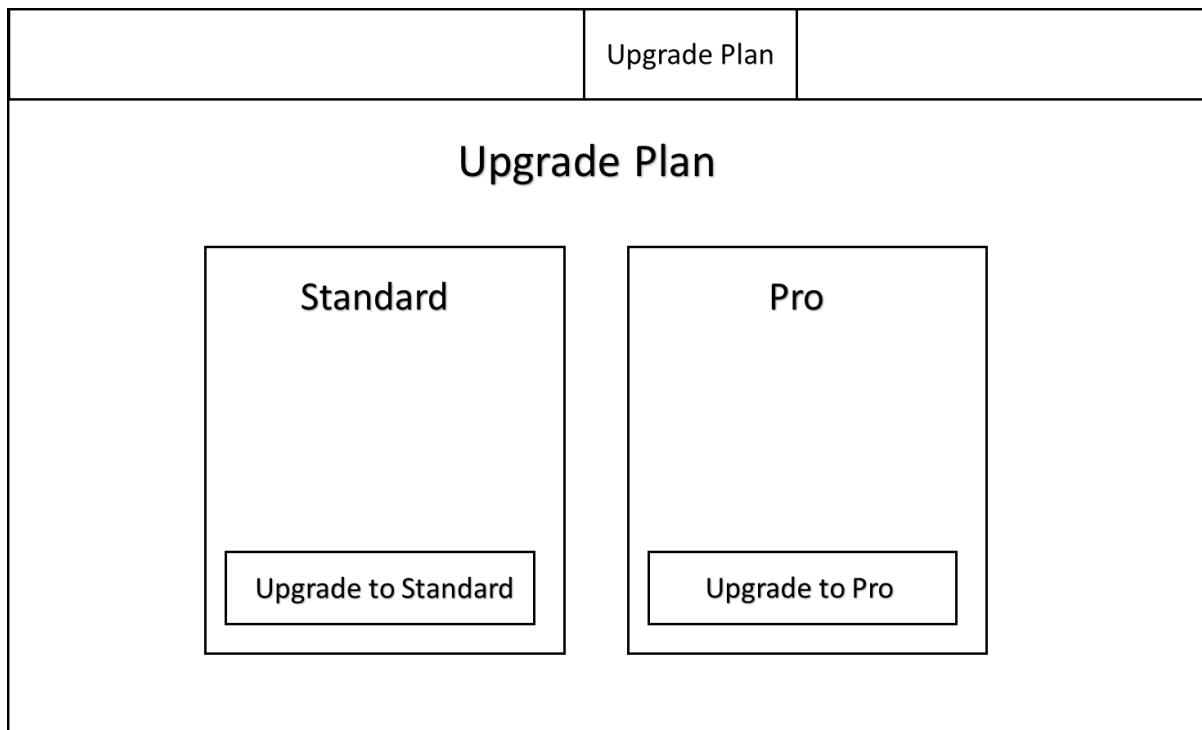
	<p style="text-align: center;"><b>SIGN UP</b></p> <p style="text-align: center;"><u>Organization</u></p> <p>Email: <input type="text"/></p> <p>Password: <input type="password"/></p> <p>Confirm Password: <input type="password"/></p> <p>Name: <input type="text"/></p> <p>Organization Name: <input type="text"/></p> <p>Organization Website: <input type="text"/></p> <p><input type="checkbox"/> I accept the terms and conditions</p> <p style="text-align: center;"><input type="button" value="Sign Up"/></p>	
--	--	--

### UC11:

	<p>Workspace</p>	
		<p style="text-align: center;"><b>Add Data Set</b></p>
	<p>Add Data Set</p>	<p style="text-align: center;"><input type="text"/> <input type="button" value="Upload"/></p>

## Project Requirements

### UPGRADE PLANS:



### UC12:



## Account Settings

Email:	<input type="text" value="*****"/>	UserType:	<input type="text" value="Organization"/>
Password:	<input type="text" value="*****"/> Edit	Name:	<input type="text" value="*****"/> Edit
Organization Name:	<input type="text" value="*****"/> Edit	Organization Website:	<input type="text" value="*****"/> Edit

## Change Password

Current Password:	<input type="text" value="*****"/>
New Password:	<input type="text" value="*****"/>
Confirm New Password:	<input type="text" value="*****"/>

**Change Name**

Current Name:  
Lorem Ipsum

New Name:  
\*\*\*\*\*

Cancel      Change Name

**Change Organization Name**

Current Organization Name:  
Lorem Ipsum

New Organization Name:  
\*\*\*\*\*

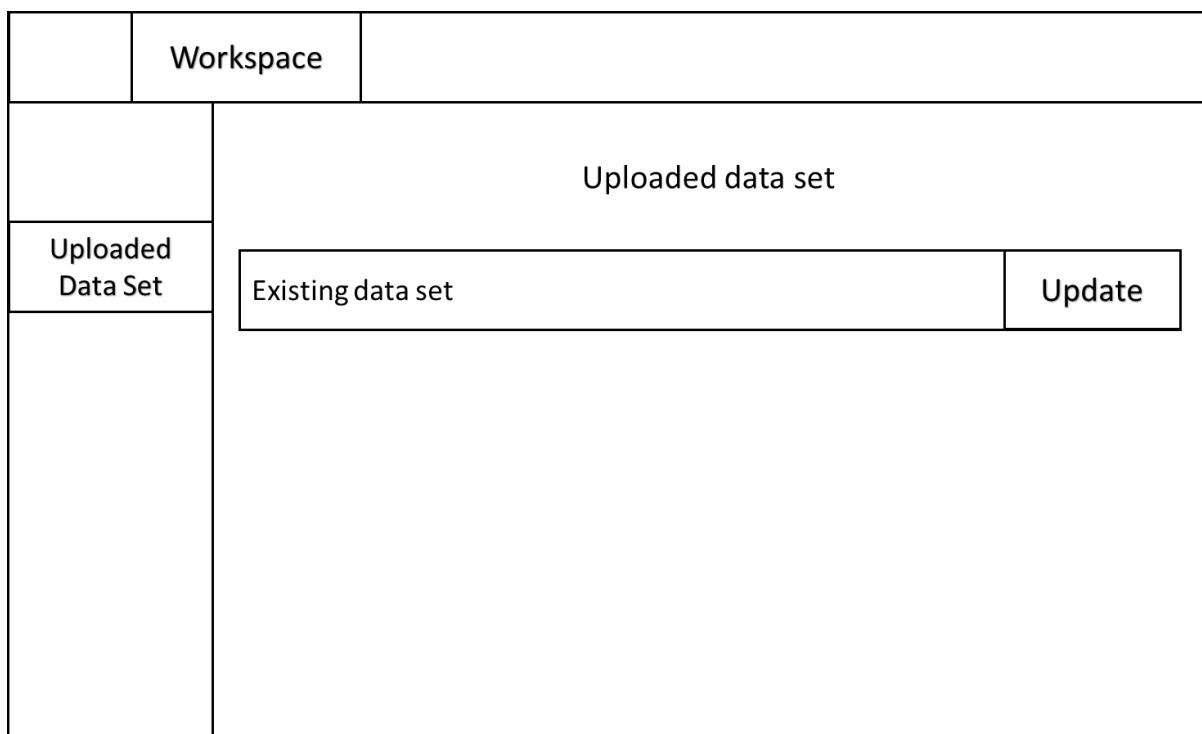
Cancel      Change Name

## Change Organization Website

Current Organization Website:  
Lorem Ipsum

New Organization Website:  
\*\*\*\*\*

**UC13:**



## Project Requirements

### UC14:

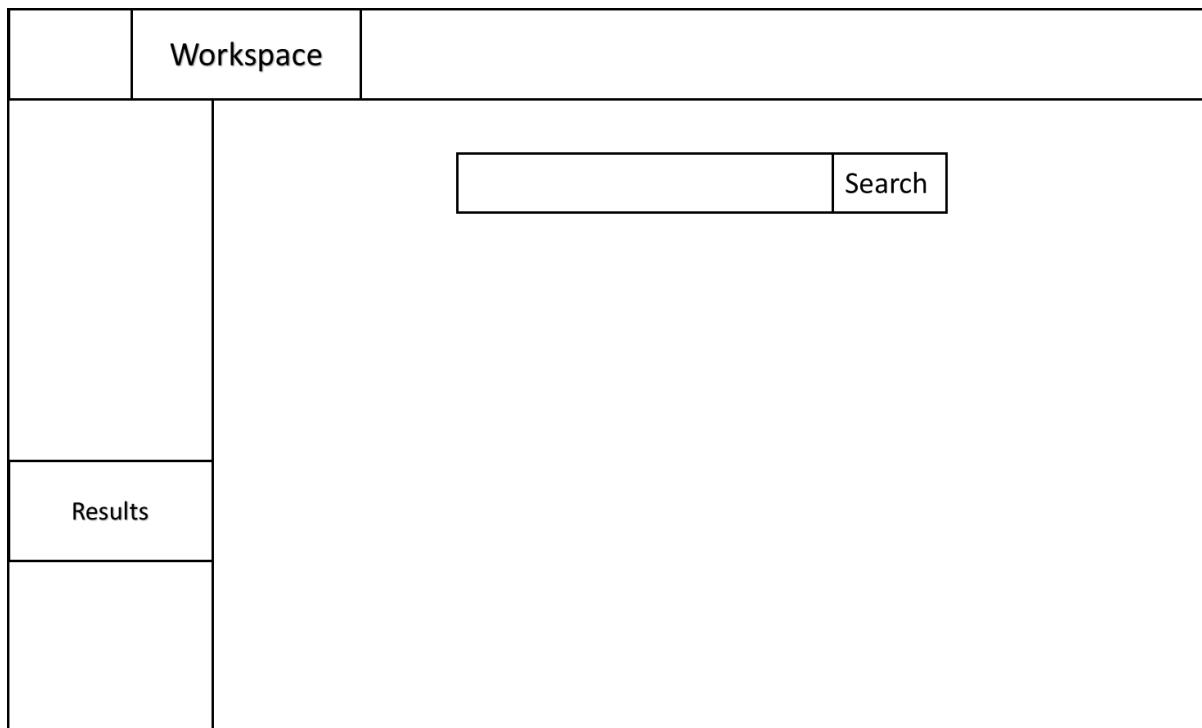
	Workspace	
	<h3>Ratings Predictions</h3>	
Generate Ratings/Recommendations	User: <input type="text"/>	Product: <input type="text"/>
	<input type="button" value="Generate"/>	
	Results: User A will give product a rating of X	

### UC15:

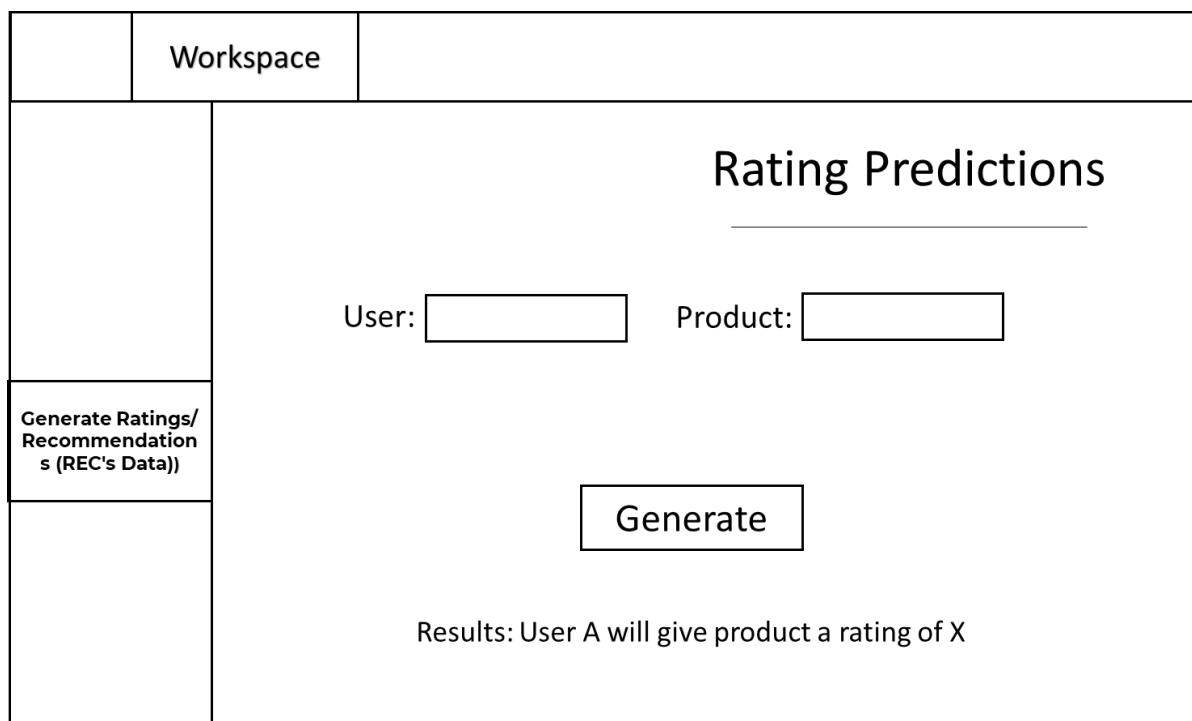
	Workspace	
	<h3>Recommendations</h3>	
Generate Ratings/Recommendations	Product: <input type="text"/>	
	<input type="button" value="Generate"/>	
	Recommendations: X, Y, Z	

## Project Requirements

### UC16 (replace current uc16?):

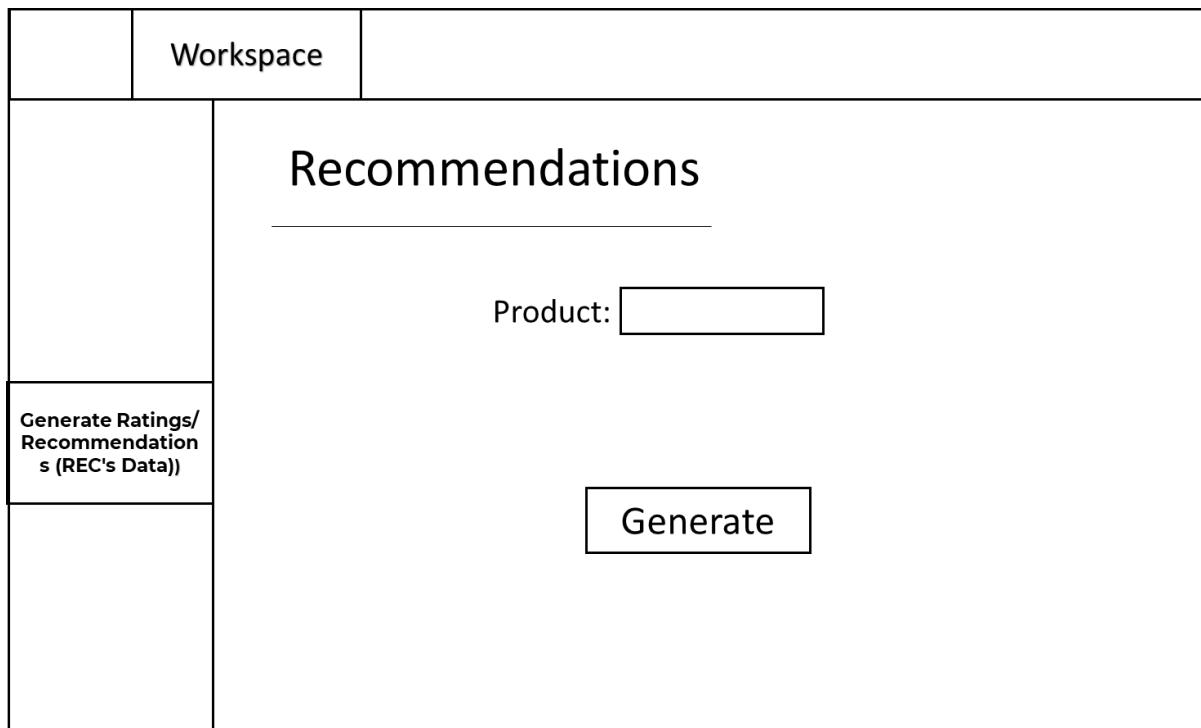


### UC17:

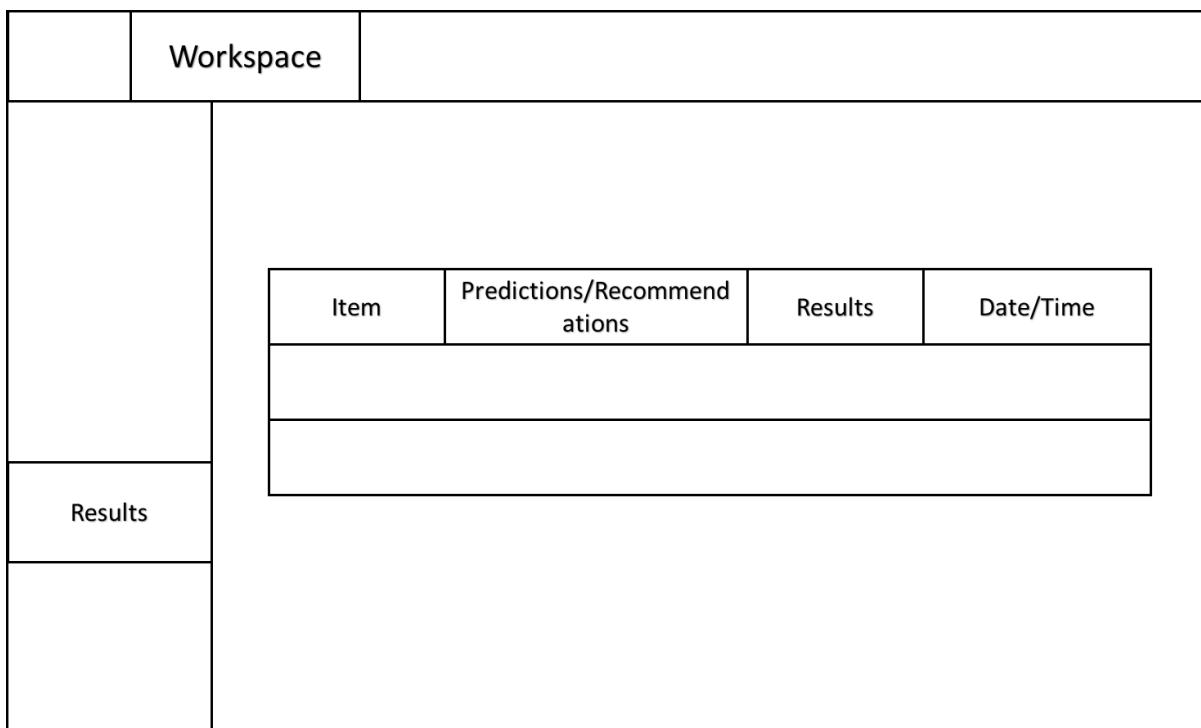


## Project Requirements

### UC18:



### UC19:



## Project Requirements

### NEW UCs (add list and uploaded list for web crawling)

	Workspace	
	Add List of URLs	
Add List of URLs	<input type="file"/> Upload	

	Workspace	
	Uploaded List of URLs	
Uploaded List of URLs	<input type="text"/>	

## Individual

**UC20:**

	<p style="text-align: center;"><b>LOGIN</b></p> <hr/> <p>Email:</p> <input type="text"/> <p>Password:</p> <input type="password"/> <p style="text-align: center;">Forgot Password?</p> <p style="text-align: center; background-color: #e0e0e0; padding: 5px;">Sign In</p>	
--	--	--

**U21:**

	User (Initials)
	<input type="button" value="Log Out"/>

**U22:**

	<p style="text-align: center;"><b>SIGN UP</b></p> <p style="text-align: center;"><u>Individual</u></p> <p>Email: <input type="text"/></p> <p>Password: <input type="password"/></p> <p>Confirm Password: <input type="password"/></p> <p>Name: <input type="text"/></p> <p><input type="checkbox"/> I accept the terms and conditions</p> <p style="text-align: center;"><b>Sign Up</b></p>	
--	---	--

**UPGRADE PLANS:**

	Upgrade Plan	
<b>Upgrade Plan</b>		
<b>Standard</b>	<b>Pro</b>	
<input type="button" value="Upgrade to Standard"/>	<input type="button" value="Upgrade to Pro"/>	

## Project Requirements

**U23:**

	User (Initials)
	Account Settings

### Account Settings

Email:	*****	UserType:	Organization
Password:	*****	Name:	*****
			Edit

## Change Password

Current Password:

New Password:

Confirm New Password:

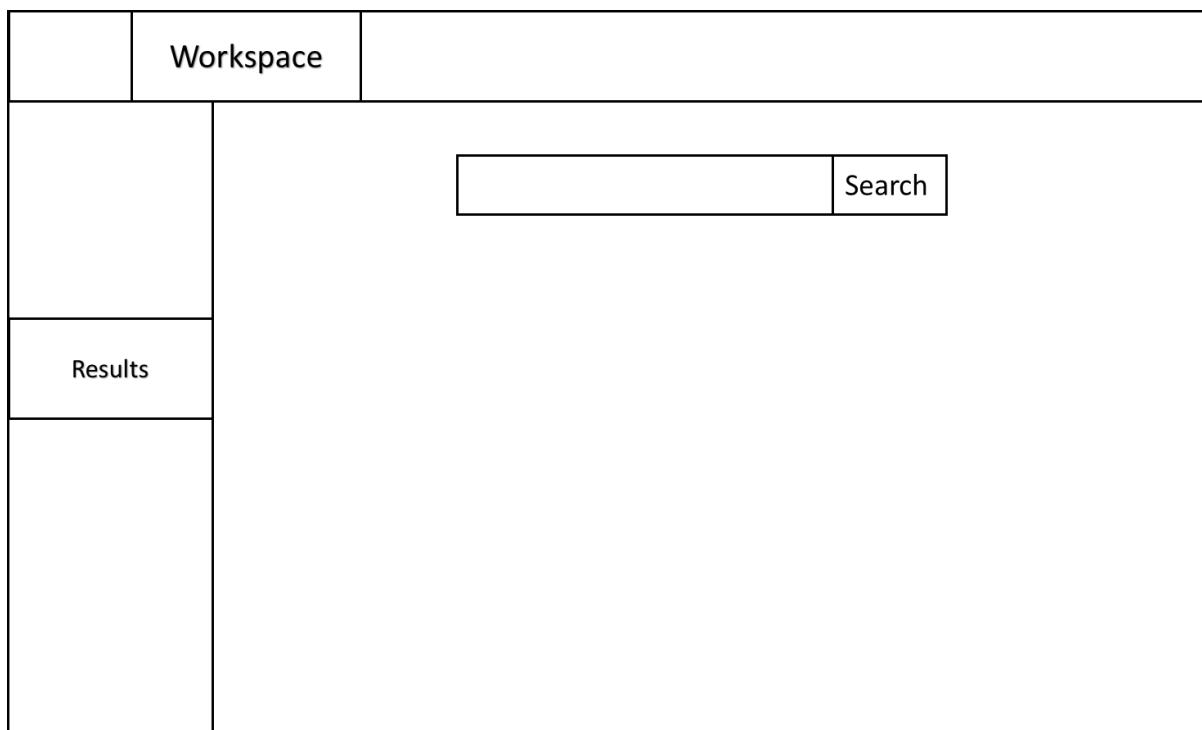
## Change Name

Current Name:

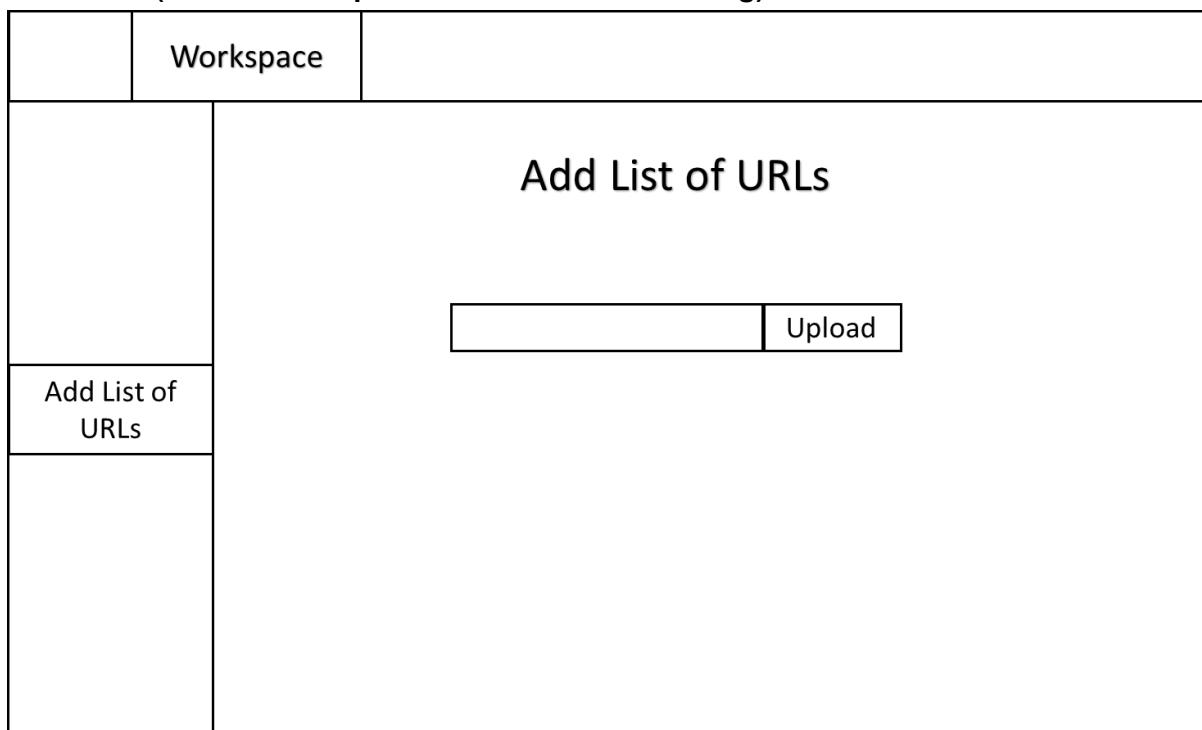
New Name:

## Project Requirements

**U24:**



**NEW UCs (add list and uploaded list for web crawling)**



## Project Requirements

	Workspace	
		<b>Uploaded List of URLs</b>  <input type="text"/>
Uploaded List of URLs		

### U25:

	Workspace	
		<b>Rating Predictions</b> _____
Generate Ratings/ Recommendations (REC's Data))	User: <input type="text"/> Product: <input type="text"/>	
	<input type="button" value="Generate"/>	
	Results: User A will give product a rating of X	

## Project Requirements

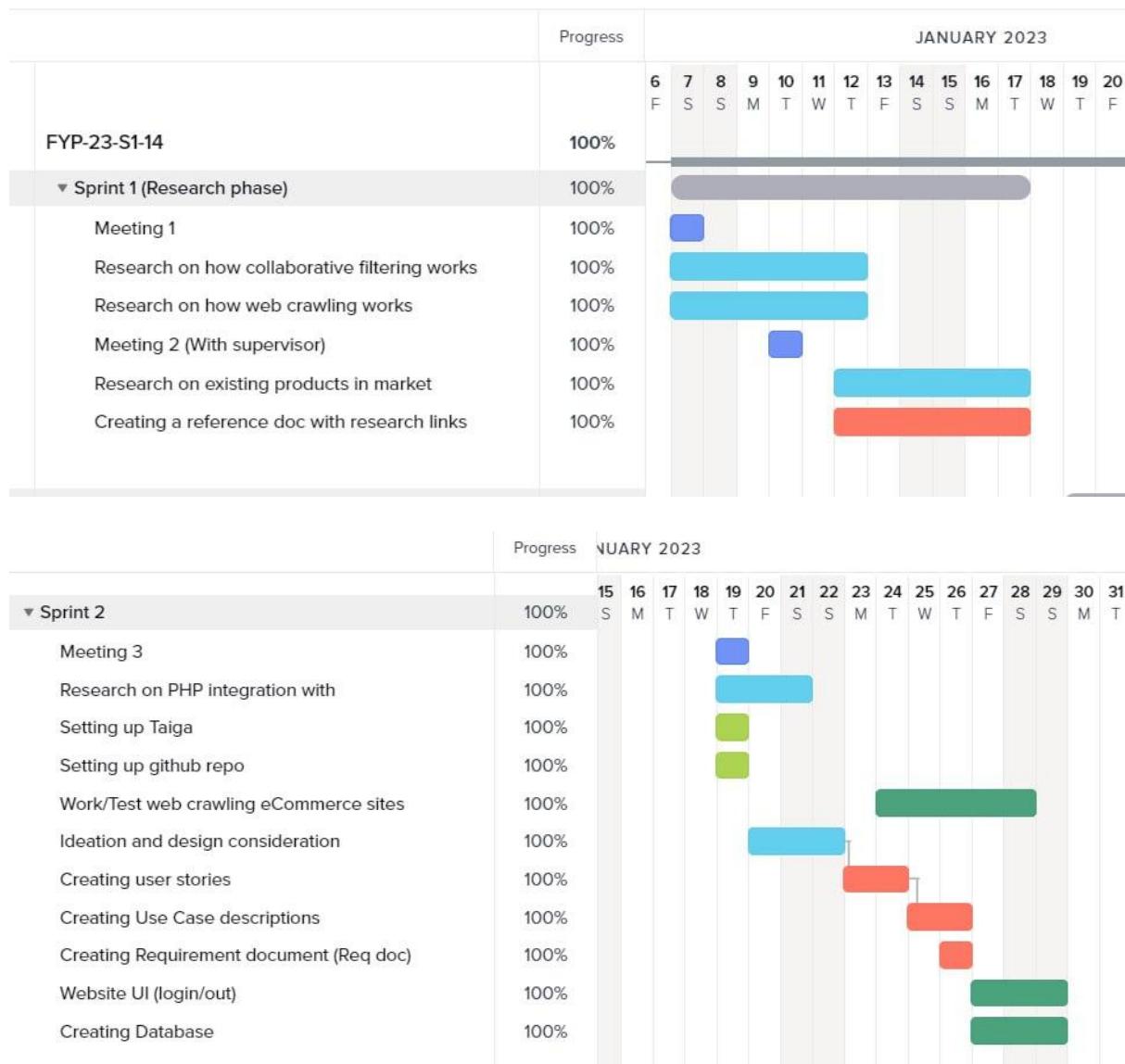
**U26:**

	Workspace	
	Recommendations	
Generate Ratings/ Recommendation s (REC's Data))	Product: <input type="text"/>	
	<input type="button" value="Generate"/>	

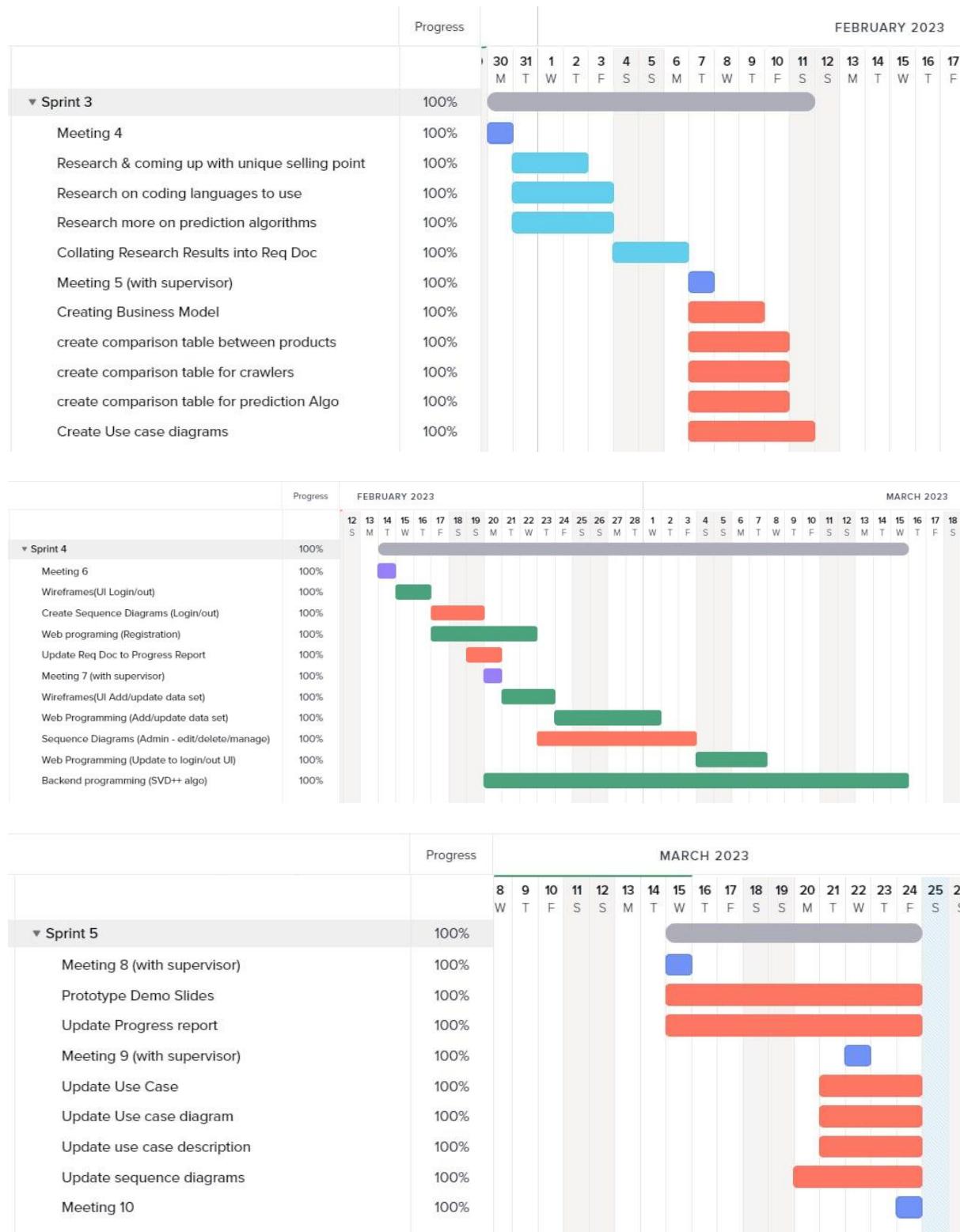
**U27:**

	Workspace													
Results	<table border="1"><thead><tr><th>Item</th><th>Predictions/Recommendations</th><th>Results</th><th>Date/Time</th></tr></thead><tbody><tr><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td></tr></tbody></table>		Item	Predictions/Recommendations	Results	Date/Time								
Item	Predictions/Recommendations	Results	Date/Time											

# Gantt Chart (Work Breakdown Structure)



## Project Requirements



# Meeting Minutes

## Meeting minute 1

<b>Meeting Title:</b> Sprint 1 Meeting 1 (Once per week)		
<b>Date:</b>	7/1/2023	
<b>Time:</b>	5.00PM	
<b>Location:</b>	Face to Face Meeting	
<b>Person In Charge:</b>	Loo Joon Wee	
<b>Project Title:</b>	A Recommender System based on Collaborative Filtering	
<b>Members involved: (Attendance)</b>		
<b>Name:</b>	<b>Roles:</b>	
Loo Joon Wee	Scrum Master, Software Developer, QA Tester	
Muhamad Amir Akmal BMS	Software Developer, QA Tester	
Wee Zee En Leonard	Programmer	
Low Rui Hao	Programmer	
Tham Jia Xuan Travis	Software Developer, Admin	
<b>Progress Status:</b>		
<b>Name:</b>	<b>Status Update:</b>	
Loo Joon Wee	Finished	
Muhamad Amir Akmal BMS	Finished	
Wee Zee En Leonard	Finished	
Low Rui Hao	Finished	
Tham Jia Xuan Travis	Finished	
<b>Meeting purpose and content:</b>		
<ul style="list-style-type: none"> <li>- Finalising team roles</li> <li>- Discussion regarding project outline</li> <li>- Discussion regarding uncertainties and coming up with questions to ask project supervisor</li> </ul>		
<b>Taskings:</b>		
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>
Discussion of roles and project outline.	Everyone	7/1/2023
Discussion of any uncertainties and coming up with questions to ask the project supervisor.	Everyone	10/1/2023

## Meeting minute 2

<b>Meeting Title:</b> Sprint 1 Meeting 2 (Once per week)		
<b>Date:</b>	10/1/2023	
<b>Time:</b>	4.00PM	
<b>Location:</b>	Virtual Meeting	
<b>Person In Charge:</b>	Loo Joon Wee	
<b>Project Title:</b>	A Recommender System based on Collaborative Filtering	
<b>Members involved: (Attendance)</b>		
<b>Name:</b>	<b>Roles:</b>	
Loo Joon Wee	Scrum Master, Software Developer, QA Tester	
Muhamad Amir Akmal BMS	Software Developer, QA Tester	
Wee Zee En Leonard	Programmer	
Low Rui Hao	Programmer	
Tham Jia Xuan Travis	Software Developer, Admin	
<b>Progress Status:</b>		
<b>Name:</b>	<b>Status Update:</b>	
Loo Joon Wee	Finished	
Muhamad Amir Akmal BMS	Finished	
Wee Zee En Leonard	Finished	
Low Rui Hao	Finished	
Tham Jia Xuan Travis	Finished	
<b>Meeting purpose and content:</b>		
<ul style="list-style-type: none"> <li>- Liaise with supervisor, familiarise and introduce each other</li> <li>- Understanding the project basic requirements and overall details/outline</li> <li>- Clarify with supervisor on our individual doubts</li> <li>- Discussion and plan on approach and plan for coming week</li> </ul>		
<b>Taskings:</b>		
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>
Read up, research and understand more on topic from all aspects. Includes: <ul style="list-style-type: none"> <li>- Existing Products</li> <li>- Data Collection</li> <li>- Crawling</li> <li>- Prediction</li> <li>System/Model</li> <li>- Collaborative Filtering</li> <li>- Scrape</li> </ul>	Everyone	17/1/2023

## Project Requirements

Update the research/resources word document with useful links found while researching, for the rest of the team to review	Everyone	<input checked="" type="checkbox"/> 17/1/2023
Finish and upload current Meeting minute 2 as well as project charter	Joon Wee	17/1/2023

## Meeting minute 3

<b>Meeting Title:</b> Sprint 1 Meeting 3 (Once per week)		
<b>Date:</b>	19/1/2023	
<b>Time:</b>	6.00PM	
<b>Location:</b>	Virtual Meeting	
<b>Person In Charge:</b>	Loo Joon Wee	
<b>Project Title:</b>	A Recommender System based on Collaborative Filtering	
<b>Members involved: (Attendance)</b>		
<b>Name:</b>	<b>Roles:</b>	
Loo Joon Wee	Scrum Master, Software Developer, QA Tester	
Muhamad Amir Akmal BMS	Software Developer, QA Tester	
Wee Zee En Leonard	Programmer	
Low Rui Hao	Programmer	
Tham Jia Xuan Travis	Software Developer, Admin	
<b>Progress Status:</b>		
<b>Name:</b>	<b>Status Update:</b>	
Loo Joon Wee	Finished	
Muhamad Amir Akmal BMS	Finished	
Wee Zee En Leonard	Finished	
Low Rui Hao	Finished	
Tham Jia Xuan Travis	Finished	
<b>Meeting purpose and content:</b>		
<ul style="list-style-type: none"> <li>- Discussion of research</li> <li>- Ideation</li> <li>- Fixing of project foundation and outline</li> <li>- Finalising of architecture</li> <li>- Giving of direction and allocating taskings to individual</li> </ul>		
<b>Taskings:</b>		
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>
Start to work on website layout for log in and log out, as well as simple database for users.	Rui Hao	29/1/2023
Start to work on website UI implementation and collaborate with Rui Hao on the website	Travis	29/1/2023
Research more in-depth on integration of PHP with	Leonard	29/1/2023

## Project Requirements

<b>Python, as well as start to work on website crawling</b>		
<b>Work on user stories, user stories descriptions.</b>	Amir and Joon Wee	29/1/2023
<b>Create and share GitHub</b>	Amir	29/1/2023
<b>Settle on meeting minutes 3, requirements document and setting up of Taiga</b>	Joon Wee	29/1/2023

## Meeting minute 4

<b>Meeting Title:</b> Sprint 1 Meeting 4		
<b>Date:</b>	30/1/2023	
<b>Time:</b>	4.00PM	
<b>Location:</b>	Virtual Meeting	
<b>Person In Charge:</b>	Loo Joon Wee	
<b>Project Title:</b>	A Recommender System based on Collaborative Filtering	
<b>Members involved: (Attendance)</b>		
<b>Name:</b>	<b>Roles:</b>	
Loo Joon Wee	Scrum Master, Software Developer, QA Tester	
Muhamad Amir Akmal BMS	Software Developer, QA Tester	
Wee Zee En Leonard	Programmer	
Low Rui Hao	Programmer	
Tham Jia Xuan Travis	Software Developer, Admin	
<b>Progress Status:</b>		
<b>Name:</b>	<b>Status Update:</b>	
Loo Joon Wee	Finished	
Muhamad Amir Akmal BMS	Finished	
Wee Zee En Leonard	Finished	
Low Rui Hao	Finished	
Tham Jia Xuan Travis	Finished	
<b>Meeting purpose and content:</b>		
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Meet up with supervisor and see the feedbacks and suggestions on our progression</li> <li>- Reorganising the project progression and backlogs</li> <li>- Giving of next direction and allocating taskings to individual</li> </ul>		
<b>Taskings:</b>		
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>
Research on PHP and other possible languages on creating and maintaining website, as well as their individual pros and cons to why choosing a specific language to use.	Rui Hao	6/2/2023
Look at other existing products, find out individual unique selling points and come		

## Project Requirements

up with ideas for our own unique selling points.		
Research and try out web crawling tools and their individual pros and cons, as well as why deciding to use a specific tool.	Travis	6/2/2023
Research and try out on different types of prediction algorithms and their individual pros and cons, as well as why deciding on a specific algorithm.	Leonard	6/2/2023
Work on use case diagrams, use case descriptions.  Collating the researches of other members and documenting down in the requirement document.	Amir and Joon Wee	6/2/2023
Settle on meeting minutes 4, organising and continuing the layout of the requirement document.	Joon Wee	6/2/2023

## Meeting minute 5

<b>Meeting Title:</b> Sprint 3 Meeting 5		
<b>Date:</b>	7/2/2023	
<b>Time:</b>	4.00PM	
<b>Location:</b>	Virtual Meeting	
<b>Person In Charge:</b>	Loo Joon Wee	
<b>Members involved: (Attendance)</b>		
<b>Name:</b>	<b>Roles:</b>	
Loo Joon Wee	Scrum Master, Software Developer, QA Tester	
Muhamad Amir Akmal BMS	Software Developer, QA Tester	
Wee Zee En Leonard	Programmer	
Low Rui Hao	Programmer	
Tham Jia Xuan Travis	Software Developer, Admin	
<b>Progress Status:</b>		
<b>Name:</b>	<b>Status Update:</b>	
Loo Joon Wee	Finished	
Muhamad Amir Akmal BMS	Finished	
Wee Zee En Leonard	Finished	
Low Rui Hao	Finished	
Tham Jia Xuan Travis	Finished	
<b>Meeting purpose and content:</b>		
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Meet up with supervisor and see the feedbacks and suggestions on our progression</li> <li>- Reorganising the project progression and backlogs</li> <li>- Giving of next direction and allocating taskings to individual</li> <li>- Compiling and checking of requirement document for first submission</li> </ul>		
<b>Taskings:</b>		
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>
Adding of diagram for visual purposes, for the document on the research on pros and cons of PHP as well as other languages used for the website.	Rui Hao	11/2/2023
Research and documenting down as well as doing the diagrams to show the		

## Project Requirements

differences and pros and cons for different hosting platforms for websites.		
<p>Adding diagrams for document on the research on comparison between web crawling tools as well as their pros and cons.</p> <p>Researching and documenting down as well as doing diagrams to show the pros and cons as well as differences between other existing recommender system.</p>	Travis	11/2/2023
Adding diagrams for document on the research on algorithms for predictive technology.	Leonard	11/2/2023
<p>Work on use case diagrams, use case descriptions.</p> <p>Collating the researches of other members and documenting down in a requirement document.</p>	Amir and Joon Wee	11/2/2023
Updating on Taiga and working on deciding and documenting down Business model for our product.	Amir	11/2/2023
Settle on meeting minutes 5, organising and continuing the layout of the requirement document.	Joon Wee	11/2/2023

## Meeting minute 6

<b>Meeting Title:</b> Sprint 4 Meeting 6		
<b>Date:</b>	14/2/2023	
<b>Time:</b>	9.00PM	
<b>Location:</b>	Virtual Meeting	
<b>Person In Charge:</b>	Loo Joon Wee	
<b>Members involved: (Attendance)</b>		
<b>Name:</b>	<b>Roles:</b>	
Loo Joon Wee	Scrum Master, Software Developer, QA Tester	
Muhamad Amir Akmal BMS	Software Developer, QA Tester	
Wee Zee En Leonard	Programmer	
Low Rui Hao	Programmer	
Tham Jia Xuan Travis	Software Developer, Admin	
<b>Progress Status:</b>		
<b>Name:</b>	<b>Status Update:</b>	
Loo Joon Wee	Finished	
Muhamad Amir Akmal BMS	Finished	
Wee Zee En Leonard	Finished	
Low Rui Hao	Finished	
Tham Jia Xuan Travis	Finished	
<b>Meeting purpose and content:</b>		
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Reorganising the project progression and backlogs</li> <li>- Giving of next direction and allocating taskings to individual</li> </ul>		
<b>Taskings:</b>		
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>
Carry on with website programming for registration page as well as functional page for inputting database set, as well as managing the database side.	Rui Hao	20/2/2023
Carry on the design and implementation of UI side for registration page and functional page for inputting database set, as well as refining the log in and out page.	Travis	20/2/2023
Continue working on SVD++ side	Leonard	20/2/2023

## Project Requirements

Work on use case descriptions. Updating of requirement document	Amir and Joon Wee	20/2/2023
Updating on Taiga.	Amir	20/2/2023
Settle on meeting minutes 6.	Joon Wee	20/2/2023

## Meeting minute 7

<b>Meeting Title:</b> Sprint 4 Meeting 7		
<b>Date:</b>	20/2/2023	
<b>Time:</b>	4.00PM	
<b>Location:</b>	Virtual Meeting	
<b>Person In Charge:</b>	Loo Joon Wee	
<b>Members involved: (Attendance)</b>		
<b>Name:</b>	<b>Roles:</b>	
Loo Joon Wee	Scrum Master, Software Developer, QA Tester	
Muhamad Amir Akmal BMS	Software Developer, QA Tester	
Wee Zee En Leonard	Programmer	
Low Rui Hao	Programmer	
Tham Jia Xuan Travis	Software Developer, Admin	
<b>Progress Status:</b>		
<b>Name:</b>	<b>Status Update:</b>	
Loo Joon Wee	Finished	
Muhamad Amir Akmal BMS	Finished	
Wee Zee En Leonard	Finished	
Low Rui Hao	Finished	
Tham Jia Xuan Travis	Finished	
<b>Meeting purpose and content:</b>		
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Reorganising the project progression and backlogs</li> <li>- Checking with supervisor on guide for further progression and updating of plan to him</li> <li>- Giving of next direction and allocating taskings to individual</li> </ul>		
<b>Taskings:</b>		
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>
Carry on progressing with last meeting's taskings	Rui Hao	15/3/2023
Carry on progressing with last meeting's taskings, and do up the wireframe for the UI	Travis	15/3/2023
Continue working on programming for SVD++ side	Leonard	15/3/2023
Work on use case descriptions. Updating of requirement document	Amir and Joon Wee	15/3/2023
Updating of Taiga	Joon Wee	15/3/2023
Settle on meeting minutes 7	Joon Wee	15/3/2023

## Meeting minute 8

<b>Meeting Title:</b> Sprint 5 Meeting 8		
<b>Date:</b>	15/3/2023	
<b>Time:</b>	11.00AM	
<b>Location:</b>	Virtual Meeting	
<b>Person In Charge:</b>	Muhamad Amir Akmal BMS	
<b>Members involved: (Attendance)</b>		
<b>Name:</b>	<b>Roles:</b>	
Muhamad Amir Akmal BMS	Software Developer, QA Tester	
Wee Zee En Leonard	Programmer	
Low Rui Hao	Programmer	
Tham Jia Xuan Travis	Software Developer, Admin	
<b>Progress Status:</b>		
<b>Name:</b>	<b>Status Update:</b>	
Muhamad Amir Akmal BMS	Finished	
Wee Zee En Leonard	Finished	
Low Rui Hao	Finished	
Tham Jia Xuan Travis	Finished	
<b>Meeting purpose and content:</b>		
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Checking with supervisor on guide for further progression and updating of plan to him</li> <li>- Giving of next direction and allocating taskings to individual</li> </ul>		
<b>Taskings:</b>		
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>
Carry on integrating the website with the prediction model (Backend)	Rui Hao	22/3/2023
Carry on adding more things to the UI based on what supervisor mentioned (Frontend)	Travis	22/3/2023
Continue working on touching up the SVD++ and web crawling side	Leonard	22/3/2023
Work on sequence diagrams. And work on the progress report documentation, alongside prototype presentation slides	Amir and Joon Wee	22/3/2023
Settle on meeting minutes 8	Travis	22/3/2023

## Meeting minute 9

<b>Meeting Title:</b> Sprint 5 Meeting 9				
<b>Date:</b>	22/3/2023			
<b>Time:</b>	11.00AM			
<b>Location:</b>	Virtual Meeting			
<b>Person In Charge:</b>	Loo Joon Wee			
<b>Members involved: (Attendance)</b>				
<b>Name:</b>	<b>Roles:</b>			
Loo Joon Wee	Scrum Master, Software Developer, QA Tester			
Muhamad Amir Akmal BMS	Software Developer, QA Tester			
Wee Zee En Leonard	Programmer			
Low Rui Hao	Programmer			
Tham Jia Xuan Travis	Software Developer, Admin			
<b>Progress Status:</b>				
<b>Name:</b>	<b>Status Update:</b>			
Loo Joon Wee	Finished			
Muhamad Amir Akmal BMS	Finished			
Wee Zee En Leonard	Finished			
Low Rui Hao	Finished			
Tham Jia Xuan Travis	Finished			
<b>Meeting purpose and content:</b>				
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Checking with supervisor on guide for further progression and updating of plan to him</li> <li>- Giving of next direction and allocating taskings to individual</li> <li>- Checking of demonstration slides and product with the team and supervisor</li> </ul>				
<b>Taskings:</b>				
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>		
Carry on integrating the website with the prediction model (Backend)	Rui Hao	25/3/2023		
Touching up and refining UI (Frontend)	Travis	25/3/2023		
Continue working on touching up the SVD++ and web crawling side	Leonard	25/3/2023		
Work on sequence diagrams.	Amir and Joon Wee	25/3/2023		
Work on demonstration slides.				
Update progress report document.				
Settle on meeting minutes 9	Joon Wee	25/3/2023		

## Meeting minute 10

<b>Meeting Title:</b> Sprint 5 Meeting 10				
<b>Date:</b>	24/3/2023			
<b>Time:</b>	12.00PM			
<b>Location:</b>	Virtual Meeting			
<b>Person In Charge:</b>	Loo Joon Wee			
<b>Members involved: (Attendance)</b>				
<b>Name:</b>	<b>Roles:</b>			
Loo Joon Wee	Scrum Master, Software Developer, QA Tester			
Muhamad Amir Akmal BMS	Software Developer, QA Tester			
Wee Zee En Leonard	Programmer			
Low Rui Hao	Programmer			
Tham Jia Xuan Travis	Software Developer, Admin			
<b>Progress Status:</b>				
<b>Name:</b>	<b>Status Update:</b>			
Loo Joon Wee	Finished			
Muhamad Amir Akmal BMS	Finished			
Wee Zee En Leonard	Finished			
Low Rui Hao	Finished			
Tham Jia Xuan Travis	Finished			
<b>Meeting purpose and content:</b>				
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Giving of next direction and allocating taskings to individual</li> <li>- Checking of demonstration slides and product with the team</li> <li>- Allocating parts of demonstration presentation portion for individual</li> </ul>				
<b>Taskings:</b>				
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>		
Carry on integrating the website with the prediction model (Backend)	Rui Hao	25/3/2023		
Touching up and refining UI (Frontend)	Travis	25/3/2023		
Continue working on touching up the SVD++ and web crawling side	Leonard	25/3/2023		
Work on sequence diagrams.	Amir and Joon Wee	25/3/2023		
Refine demonstration slides.				
Update progress report document.				
Settle on meeting minutes 10	Joon Wee	25/3/2023		
Prepare for demonstration presentation on individual assigned part	Everyone	25/3/2023		

## Meeting minute 11

<b>Meeting Title:</b> Sprint 6 Meeting 11				
<b>Date:</b>	4/4/2023			
<b>Time:</b>	9.30AM			
<b>Location:</b>	Virtual Meeting			
<b>Person In Charge:</b>	Loo Joon Wee			
<b>Members involved: (Attendance)</b>				
<b>Name:</b>	<b>Roles:</b>			
Loo Joon Wee	Scrum Master, Software Developer, QA Tester			
Muhamad Amir Akmal BMS	Software Developer, QA Tester			
Wee Zee En Leonard	Programmer			
Low Rui Hao	Programmer			
Tham Jia Xuan Travis	Software Developer, Admin			
<b>Progress Status:</b>				
<b>Name:</b>	<b>Status Update:</b>			
Loo Joon Wee	Finished			
Muhamad Amir Akmal BMS	Finished			
Wee Zee En Leonard	Finished			
Low Rui Hao	Finished			
Tham Jia Xuan Travis	Finished			
<b>Meeting purpose and content:</b>				
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Giving of next direction and allocating taskings to individual</li> </ul>				
<b>Taskings:</b>				
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>		
Carry on integrating the website with the prediction model as well as email verification and new implemented features	Rui Hao	18/4/2023		
Touching up and refining UI for home page for free trial for unregistered users.	Travis	18/4/2023		
Working on UI for new implemented features and category selection				
Work on scraping based on categories	Leonard	18/4/2023		
Work on sequence diagrams.	Amir and Joon Wee	18/4/2023		
Update progress report document.				
Settle on meeting minutes 11	Joon Wee	18/4/2023		
Work on test plan and test cases				

## Meeting minute 12

<b>Meeting Title:</b> Sprint 7 Meeting 12				
<b>Date:</b>	18/4/2023			
<b>Time:</b>	11.00AM			
<b>Location:</b>	Virtual Meeting			
<b>Person In Charge:</b>	Loo Joon Wee			
<b>Members involved: (Attendance)</b>				
<b>Name:</b>	<b>Roles:</b>			
Loo Joon Wee	Scrum Master, Software Developer, QA Tester			
Muhamad Amir Akmal BMS	Software Developer, QA Tester			
Wee Zee En Leonard	Programmer			
Low Rui Hao	Programmer			
Tham Jia Xuan Travis	Software Developer, Admin			
<b>Progress Status:</b>				
<b>Name:</b>	<b>Status Update:</b>			
Loo Joon Wee	Finished			
Muhamad Amir Akmal BMS	Finished			
Wee Zee En Leonard	Finished			
Low Rui Hao	Finished			
Tham Jia Xuan Travis	Finished			
<b>Meeting purpose and content:</b>				
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Giving of next direction and allocating taskings to individual</li> <li>- Updating supervisor on progression and work plan for following weeks as well as getting feedbacks on improvements</li> </ul>				
<b>Taskings:</b>				
Tasks:	Assigned To:	Due Date:		
Carry on integrating the website with the prediction model, trial and web scraping.	Rui Hao	2/5/2023		
Work on UI polish and assist Rui hao on integration	Travis	2/5/2023		
Work on scraping based on multiple categories together	Leonard	2/5/2023		
Work on sequence diagrams.	Amir and Joon Wee	2/5/2023		
Update progress report document.				
Work on sprint 7 part 1 test cases	Amir	2/5/2023		
Settle on meeting minutes 12	Joon Wee	2/5/2023		
Finish up test plan and sprint 6 test cases				

## Meeting minute 13

<b>Meeting Title:</b> Sprint 7 Meeting 13				
<b>Date:</b>	2/5/2023			
<b>Time:</b>	11.00AM			
<b>Location:</b>	Virtual Meeting			
<b>Person In Charge:</b>	Loo Joon Wee			
<b>Members involved: (Attendance)</b>				
<b>Name:</b>	<b>Roles:</b>			
Loo Joon Wee	Scrum Master, Software Developer, QA Tester			
Muhamad Amir Akmal BMS	Software Developer, QA Tester			
Wee Zee En Leonard	Programmer			
Low Rui Hao	Programmer			
Tham Jia Xuan Travis	Software Developer, Admin			
<b>Progress Status:</b>				
<b>Name:</b>	<b>Status Update:</b>			
Loo Joon Wee	Finished			
Muhamad Amir Akmal BMS	Finished			
Wee Zee En Leonard	Finished			
Low Rui Hao	Finished			
Tham Jia Xuan Travis	Finished			
<b>Meeting purpose and content:</b>				
<ul style="list-style-type: none"> <li>- Checking of individual progress</li> <li>- Giving of next direction and allocating taskings to individual</li> <li>- Showing supervisor current progression and work plan for following week</li> </ul>				
<b>Taskings:</b>				
<b>Tasks:</b>	<b>Assigned To:</b>	<b>Due Date:</b>		
Solve the bugs in website and work on UI aspect of the website for error situations on all functions	Rui Hao	9/5/2023		
Work on integration of web scraping into the website and the UI display of results from rating predictions as well as recommendations	Travis	9/5/2023		
Help travis to work on web scraping such that result can display in a user friendly way	Leonard	9/5/2023		
Work on sequence diagrams.	Amir and Joon Wee	9/5/2023		
<b>Update progress report document.</b>				
Finish up sprint 7 part 1 test cases	Amir	9/5/2023		
Work on testing for sprint 7 part 1 test cases				
Settle on meeting minutes 13	Joon Wee	9/5/2023		
Work on sprint 7 part 2 test cases				
Work on testing for sprint 6 test cases				