ITS66404 Software Engineering
A S S I G N M E N T   1
Project Based Learning (PBL) with Purpose Learning

HAND OUT DATE: 9th September 2022, Friday (Week 2)
HAND IN DATE: **7th October 2022 (Friday by 11:55pm Malaysia time) (Week 6)**

WEIGHTAGE: 20%

Instructions to students:
- The assignment should be attempted in groups of 5 students.
- Complete this cover sheet and attach it to your assignment – this should be your first page.

Student declaration:
I declare that: ● ***I understand what is meant by plagiarism***   ● ***The implication of plagiarism have been explained to us by our lecturer***   *This project is all our work and I have acknowledged any use of the published or unpublished works of other people.*

Names of Group Members

Name	Student ID
Ishihara Satoaki	0354208
Aman Mahmud Bin Amer Mahmud	0348725
Mohamed Fahad Farhan	0354487
Mohammad Sameed Khan	0353846
Thua Sin Wei	0354566
Jinan Nisar	0354690

Avoid copy and paste job in your report and it is considered as plagiarism. Plagiarism in all forms is forbidden. Students who submit plagiarised assignment will deserved a 0 marks.

Table of contents

Introduction

Our group is led by Ishihara Satoaki -leader of this group-, and other members are Mohammad Fahad Farhan, Mohammad Sameed Khan, Jinan Nisar, Aman Mahmud Bin Amer Mahmud, and Thua Sin Wei.

Contributions are as followed: Introduction to Solution by Fahad, Discuss impact of solution & business viability application by Jinan. Requirements Gathering' Elicitation, Scope Definition by Aman and Jinan, Functional requirements by Jinan & Aman & Fahad, and non-functional requirements by Sin Wei, Sameed, and Satoaki. Architecture overview parts are done by Jinan and Sameed and Sin Wei, UML diagrams are designed by Ishihara Satoaki. Conclusion and Future Enhancement by Fahad, Sameed and Jinan, and other remainings by Ishihara Satoaki.

Our team's vision is to "**Create a sustainable ecosystem**" where we aim to minimize the impact of waste on the environment by integrating technological pursuits in order to make a difference. The idea stems from being conservative with the resources that are provided to us and making use of an important mission statement **'Reduce and Reuse'**. Moreover, this venture that has been set out is part of creating a global system where organizations and individuals around the world can be a part of igniting change. The app 'ConserveR' allows users to leverage on the concept of **Re-using** as it encourages its users to reduce purchase of products from retailers and rather rely on existing products that other users are looking to **give away** or offer on a **lending** scheme. The app also supports reducing waste at the same time as less users will rely on the purchase of brand new products which reduces the disposal of waste. This vision is of extreme importance because avoiding the creation of waste is the most efficient strategy to minimize it. The extraction of raw materials from the earth, the manufacture of the product, and its transportation to the location where it will be sold all result in greenhouse gas emissions that contribute to climate change. Therefore, the best strategies to conserve natural resources, save the environment, and make savings are through reduction and reuse.

One person's trash is another person's treasure. Selling or giving any unneeded furniture, tools, or clothing instead of throwing it away will benefit others in addition to **eliminating trash**. Used books, functional gadgets, and unused furniture may all be donated to local churches, community centers, thrift shops, schools, and charity groups.

As a nation, we're generating further scrap and we do not know what to do with it. Ineffective or reckless disposal of this waste can contaminate the ecosystem and pose a public health threat. Citizens are discovering that there's no easy way to get relief of the scrap they formerly assumed could be buried or burned and forgotten.

Current disposal styles hang our health, safety, and eco-system, and pose fresh circular costs to society. Waste treated in this manner may pollute groundwater, gutters, and aqueducts. When waste is burned, it releases dangerous feasts into the air and leaves poisonous remains in the form of ash. These dangerous waste derivations find their way into humans and creatures in one form or another.

Therefore, we as a team have proposed a solution to overcome this problem.

Problem Statement

We are in a time where unprecedented disasters are occurring around the world due to climate change and heavy environmental disaster, which results in mass pollution, global warming and other catastrophes. In order to curb this, we need to come up with innovative solutions to tackle these challenging scenarios in a creative manner. Hence, we developed an app named ConserveR which allows users to give-away their existing products or offer it on a borrowing basis. This aligns with the SDG 12 (Responsible consumption and Production) and SDG 13 (Climate Action).

In recent times, food waste has entered growing interest from original, public and European policymakers, transnational organizations, NGOs as well as academics, adding enterprises about food security and environmental impacts. While food waste occurs in all stages of the food force chain, private homes have been linked as crucial actors in food waste generation. Besides food wastage, the era of fast fashion has also raised the deeply concerning problem of excessive dumping of unwanted clothes in open landfills. It has been estimated that the generation of textiles in 2018 was 17 million tons (EPA, 2018) which raises a point of concern for us all.

Furthermore, increased disposal of plastic waste has caused major issues with the environment by causing plastic pollution in the oceans which has negatively had an impact on marine life due to the non-biodegradable nature of plastic. Plastic pollution has become one of the most burning environmental issues, as fleetly adding products of disposable plastic products overwhelms the world's capability to deal with them. Plastic pollution is most visible in developing Asian and African nations, where scrap collection systems are frequently absent. This has raised concerns to the United Nations.

Lastly, the circulation of new production leads to resource depletion which can have a negative impact on the future of essential product generation. This is because nonrenewable resources like hydro-carbons are used to produce items such as plastic which then has a negative cyclical effect on the environment. This leads to the buildup in landfills which has a potential socio-economic cost.

Solution

To address the aforementioned problem statement, we have developed the ConserveR app which allows users to give away or lend their goods for a specific time-frame. The app helps improve responsible consumption and disposal as well as with the protection of the environment. This is because when an organization and the ordinary consumer uses the ConserveR application they are encouraging the use of existing products as they will request for products that are given away by others. This reduces consumption of single wrap plastic used to package products and reduces the need for manufacturing products which then leads to lower carbon emissions made by factories. It encourages reuse instead of excessive consumerism which helps not only reduce waste in a particular locality, but also helps preserve earth's natural resources which are depleting at a record high rate.

The app in particular helps to reduce food wastage as in our current age of heavy consumerism, people often overstock groceries or other food items which only end up expiring or rotting in the pantries or kitchen shelves. Since the application provides a platform for giving away food items as well, the end users are given the leverage to give the excess or unwanted food items away instead of dumping them straight into trash, which would only contribute to the waste management crisis and further pollute the surroundings. Besides this, it also prevents the need to discard clothes which results in landfill over congestion. Instead of being thrown away, the clothes can be passed down to another end user who is in need of a similar clothing item.

To add on, our application takes on a unique approach of helping meet the goal of sustainability, reduction of waste and the promotion of re-use. This is achieved by the platform's donation service where users are allowed to make a donation value of their choice to support various causes that are hosted by the application such as beach clean up efforts, donations that go towards environmental organizations like green-peace and the global environment center that support environmental efforts. This donation is partially also contributed towards the app developmental costs and maintenance costs.

Discuss impact of solution

The project aims to develop a social enterprise that impacts its demographic in a significantly positive manner by aligning itself with the UN's Sustainable development Goals, in particular Waste Management & Recycling. The application proposed also helps in building sustainable cities and communities as well as reducing inequalities, thereby strengthening its association with the SDGs and providing a wide impact solution. Our application extends its facilities to the entire nation, helping them connect with people who want to give away stuff permanently or for borrowing in their area.

Due to the flexibility and wide categories of items that can be listed, our application benefits people of all ages, gender and professions in various ways. It provides availability of useful stuff and helps reduce waste by giving an opportunity of reuse while benefiting all communities alike as less toxic waste is generated and pollution levels are kept in check. Our application has the potential of benefitting various categories of end users ranging from small businesses who want to promote their products to independent individuals who want to give their belongings away instead of discarding it, thereby limiting trash and building an effective giveaway network. Items that would otherwise get dumped in trash and oversaturated landfills all across the country, would be handed down to someone who needs them instead, whether as raw material for something new or as a usable item on its own. This would help manage the waste disposal systems in various cities as less waste would be produced and would also keep the cities clean as fewer people would dump unneeded items out in the open.

Our application provides an effective way not only for people to get rid of items they no longer need in an efficient and sustainable way, but also for people to find items they need by reusing instead of purchasing fresh from the stores, helping them save money and also reduce excessive consumerism which is detrimental to a sustainable city or community. In this way, the application can help decrease excess consumption in general which provides the earth with more time to replenish its resources and decrease the chances of running out of them.

Since the foundations of the project relies on the concept of reuse, it helps in both resource management on a broader time scale and waste management in a more short term perspective. Both these characteristics of our application make it an effective aid in devising a more sustainable way of life with less waste production and more efficient use of natural

resources. Our application effectively tackles the problem of excessive waste generation in cities by providing the masses with a platform to hand over their stuff for reuse and also provides them the opportunity to utilize second hand goods instead of buying items first hand, avoiding resource overuse.

Competitors:

NeighborGoods
Craigslist
eBay
Etsy

NeighborGoods

Strengths	Weaknesses
● Creates well-knit communities through a form of social network in addition to the donating, borrowing and selling services ● People are more likely to trust those in a similar neighborhood	● Focuses almost exclusively on local neighborhoods for trade
Opportunities	**Threats**
● Allows local communities to form at a rapid pace and build trust ● Whole neighborhoods are able to benefit from generous individuals	● Entire networks may have poor experiences based on the actions of a few or an individual. ● May lack growth if local communities do not have at least a few individuals who have the capability to share goods.

Craigslist

Strengths	Weaknesses
- Extremely popular, especially in the United States, but also is available in 70+ countries. - Established itself as dominant in the market - Simple business model - takes small fees for posting ads.	- Outdated platform not making use of recent advancements in web technology - Has poor reputation due to not properly enforcing illegal trade on the platform - Few protections against spam - Mostly attracts individuals looking for profit instead of charity - people are unlikely to donate items if they are charged a fee.
Opportunities	**Threats**
- Current popularity may allow it to grow in other countries it has not expanded to and may eventually become a global platform. - Large profits may allow the company to expand its services in the future.	- Many new companies are entering the market. Craigslist is broad in its product categories and may struggle to keep up with numerous platforms that are specialized for specific categories of trade - Fraud is widespread on the platform.

eBay

Strengths	Weaknesses
- Supporting international transactions and trade - Continually increasing amount of listings and sales being made	- Scams are frequent and eBay does little to get involved to prevent them. - Large amount of competition, both from large scale sites like Amazon and Alibaba and local selling options through social network features such as Facebook Marketplace.
Opportunities	**Threats**
More focus on the resale of used items may allow eBay to lock down a niche marketplace that is currently untouched by large marketplaces like Amazon and Alibaba	Can face legal trouble as the platform might facilitate fraud and illegal market manipulation for certain items.

Etsy

Strengths	Weaknesses
● Supports international transactions and trade ● High customer satisfaction ● High rate of reliable sellers ● Established as a central platform for small businesses	● Web technology is outdated
Opportunities	**Threats**
● On top of being able to support current markets, new markets may pop up in Etsy as certain categories become popular or "trendy". ● Increased progress towards globalization	● Prices are heavily influenced by inflation rates and may cause previously affordable items to become expensive ● Supply of items are usually low as they are made by small businesses or individuals.

Discuss business viability application

Despite being a non profit social enterprise, our application asserts its viability in the market through its significant social impact on its demographic. Viability simply means the ability to survive and in the case of non profits, it's measured in terms of how effectively it delivers the social value it's centered around i.e. its social viability. This measure is crucial in the aspect of gaining adequate funding and support from charity organizations and other volunteers. By providing significant assistance in the process of waste reduction and efficient resource management, our application shows a strong potential of social viability as it dedicates itself to an urgent environmental issue. It caters to a broad target audience which increases its potential of reaching a higher social impact and thereby strengthening its status as an efficient social enterprise.

The primary source of funds for our application are the users who download and use the app by making donations in the donation pool provided in the app if they're willing to support the social causes promoted. Besides direct donations, our application manages resource spendings through the support of volunteer programs, corporate partners and public or private NGOs who wish to join the social welfare cause. Our application gains support of corporations and other small businesses by providing them with a convenient platform to promote their new products or clear stock for old products with its give away and lending features.

In terms of market scope, our application would have a significant monopoly over the recycling and reuse market as it provides an efficient and user friendly approach to waste reduction that hasn't been frequently adopted. Although it would face competition from certain websites that strive to achieve similar goals or follow similar methods, its free of charge services and the opportunities to hand down items that would not have much market value on e-commerce platforms would help our application stand out. While certain applications do allow item listing and goods exchange services, they are mostly focused on making quick profit rather than donating items to reduce waste, which positively highlights our application as a donation app for greater social well being as opposed to just a buying and selling platform dedicated to profit making. All these characteristics increase the chances of it being widely adopted by environment conscious and utility seeking customers alike, thereby broadening the demographic of the platform and boosting popularity.

Since the application sustains itself by maintaining trust and working on greater social wellbeing, it focuses greatly on maintaining a decent brand image and enthusiastic political and corporate support. Therefore it significantly relies on frequent advertisement of its achievements and opportunities which are made possible by sufficient fundings from its end users and supporting organizations or corporations. The resource utilization from the system's side is very low as compared to general e-commerce sites as the application simply provides a platform for posting and accessing listings and incorporates no delivery services or order management processes. This makes it more likely for the application to continue delivering its service over a long period of time without requiring over the top funding.

System Requirements

1. **Requirements Gathering' Elicitation, Scope Definition**

For software engineering, requirements gathering which is also known as requirement elicitation is a practice to gather and define the software requirement. It involves acquiring business requirements, coding the data in the form of user stories, and building a timeline for the software development. Therefore, it is mandatory for software development in order to achieve success for both the user and the organization for the development of mobile applications. Without it, it can be detrimental for the software project causing it to jeopardize the mobile application leaving room for error. Thus, we have ensured gathering of requirements as well as defining of the scope before developing our application.

Requirements Gathering' Elicitation
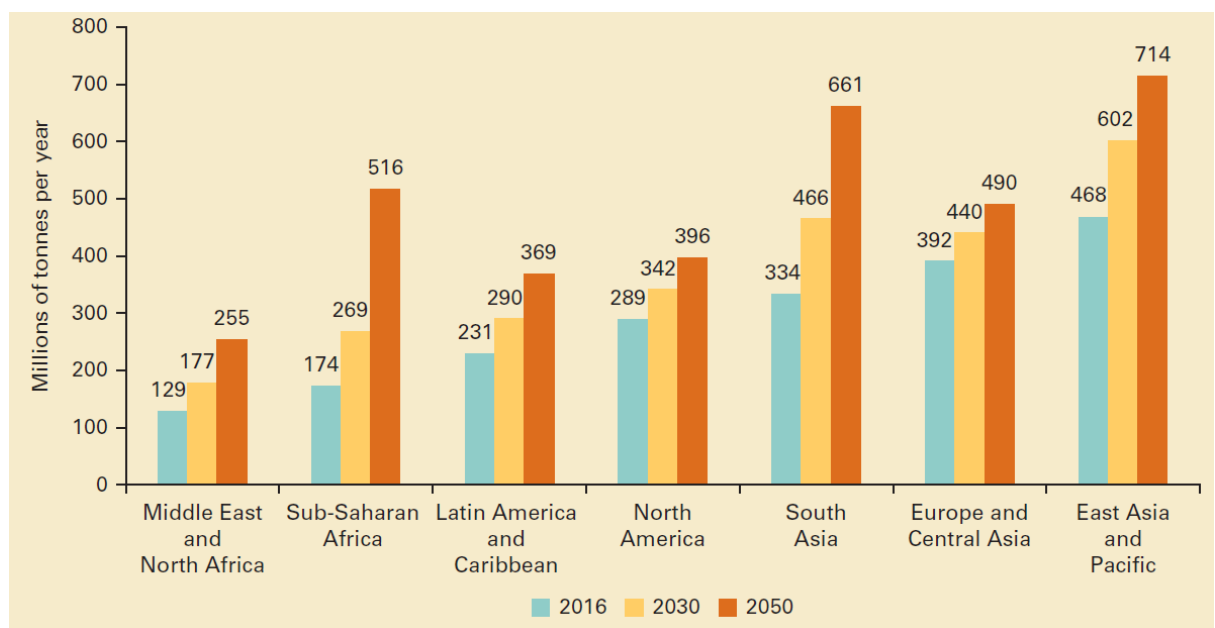
Based on the requirement elicitation, our mobile application was developed as a way to encourage people to reuse and recycle items by borrowing or sharing items that had been discarded or items that haven't been used in order to prevent waste from getting into the landfill or into the incendiary which could affect the environment. According to the World Bank, Every year, the world produces 2.01 billion tonnes of municipal solid waste, at least 33% of which is not managed in an environmentally safe manner. It is estimated that the global waste will increase around 3.40 billion tonnes which is double the amount of population growth over the same period and it will continue to

increase around 2050. To develop this mobile application, we have adopted a Software Development Life Cycle (SDLC) of waterfall model and an agile model as a logical progression of software development as well as providing details for gathering requirements. SDLC or known as Software Development Life Cycle is a diagrammatic representation of a software model that is built within the structure of steps or methods in order to develop the software with the instructed methods. Without implementing SDLC in software development, a software product would not be created in a methodical and disciplined way. Therefore, we have two types of SDLC models such as the waterfall model and the agile model.
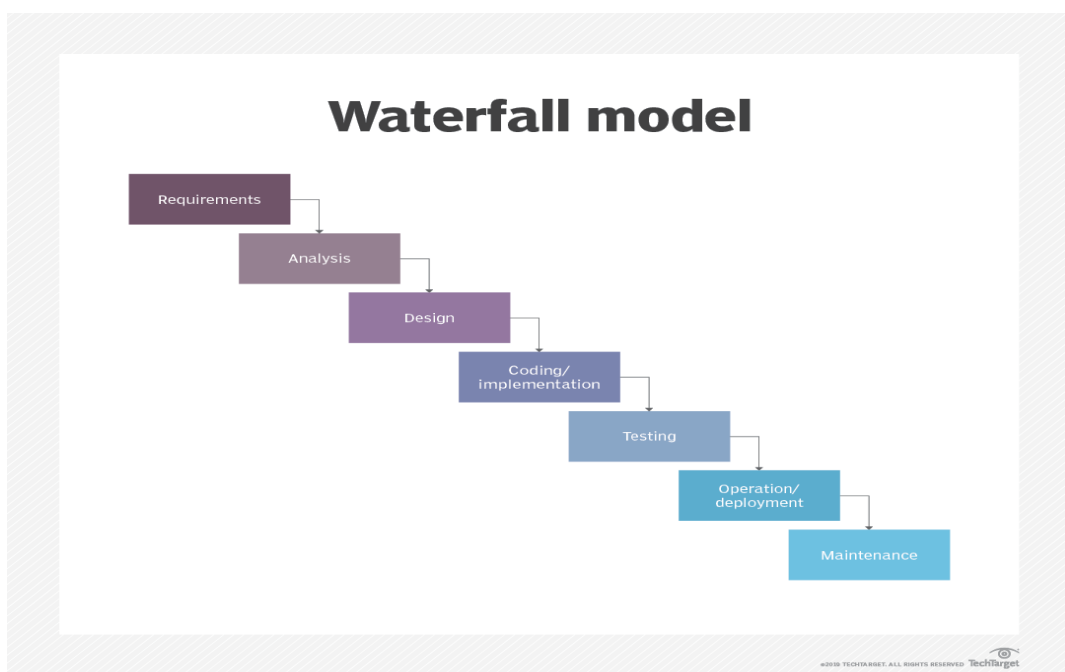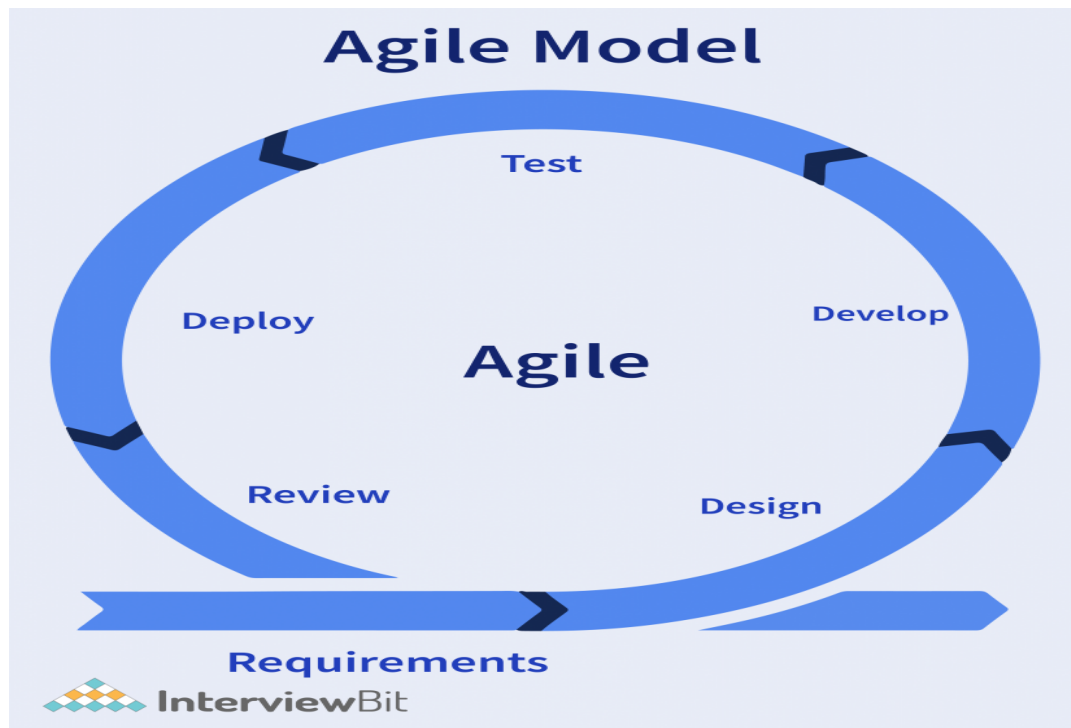
The waterfall model is a linear sequential life cycle which means each phase must move on with the next phase without overlapping it. Based on our own waterfall model, it has two types of steps for the software development namely planning step and analysis step. For the planning step, we have to come up with our own requirements and categorize them into two differences for the software development such as functional and non-functional requirements. We also need to provide the description and sub requirements in order to understand the functions and implement these features into a mobile application. For the analysis step, we need to analyze the business model and receive opinions and feedback from the users. In order to implement the business model, the mobile application needs to be viable as it competes with other mobile applications in the market such as ebay and etsy by reaching its own target audience and requires its own fundings by receiving donations from the stakeholders which is mainly from end users despite being a non profit social enterprise. Speaking of target audience, we need to set the requirements according to the needs of the end users, who are the main stakeholders in our application, by conducting interviews or surveys in order to analyze their opinion and their feedback on our mobile application. Since we complete the phase of the waterfall, we move on to an agile model.

Agile model is an existing phase that is cycling around for continuous improvement. Based on our own agile model, there are 5 steps to implement it for the software development such as design step, develop step, test step, review step and maintenance step. For the design step, we have decided to used react native as a main baseline for developing an mobile application and we have planned and developed our own architecture overview such as designing our own user interface (UI) and user experience (UX) to make more of emotional design which makes the UI system much easier to understand. We also developed our own system analysis and design such as use case diagrams and case diagrams in order to understand the functionality of the system software. After completing the design steps, we have managed to implement the develop phase where we input the code in react native from the previous design phase and test the output of the code to see the functionality. For the test step, we decided to test our beta mobile application as well as its functions to prevent encountering bugs which could bricked our software development and improve the aspect of it to enhance our mobile application. If the test step is completed, we conduct a review step where we review our own mobile application and review every aspect of it from the developers and users perspective. Lastly for the agile step, we have finally moved on to the maintenance step where we provide maintenance for mobile applications by updating the software and security feature once a year to prevent hacking and bugs from bricking the mobile application. Based on what we're gathered the requirement, we have managed to brainstorm our ideas applying each part of ideas into a single requirement. We then conducted a survey and provided some interviews with other people in order to get ideas as well as feedback and opinion.

Scope Definition

For the scope definition of this project, the main goal of our project is to encourage people to reuse and recycle items by sharing or borrowing which is aligned with SDG 12 and SDG 13 which is Responsible Consumption and Production as well as Climate Action. The application is governed by our development team which consists of separate sub teams contributing to each step in the SDLC. The main stakeholders of the app are the organizations and end users using it, whose data is managed by the system's database and stored over cloud. Since our application doesn't involve sale of products and only stores a user's personal information and financial details in case of

donations, the scope in terms of amount of data being stored is relatively small as compared to regular ecommerce sites. The relative low amount of data being processed also reduces the development and maintenance costs, which increases cost effectiveness. The cost of failure is also reduced with the correct planning of activities, more features meeting the requirements at first pass, on time delivery of services offered by the system and low rework costs. The simple but effective features of the application coupled with the efficient planning process are the key contributors in achieving this threshold.

2. Functional Requirements

Type	Requirements	Sub Requirements / Requirement details	Description
User level facility	1. Users shall be able to register and login.	• The system must allow new users to register themselves using their email, nickname, age, phone number and password. • The system must allow existing users to login by using their email or phone number and password. • The system should allow users to login through their google and facebook accounts.	The system allows duplication of nicknames but the email address used should be unique for each user. Only Malaysian phone numbers are accepted and users can log in through it as well. The nickname for people logging in through their google or facebook accounts is added automatically.
	2. Users should be able to post, edit and manage lists.	• Users can post a public listing falling under two categories: a) Giving away. b) Borrowing. • Users can edit a listing once posted to make changes. • If any listing is posted, users must add at least three pictures of the item being given away in their listing along with a description of it. • Users must add a pickup location and preferred pickup time range for the users wishing to request for the item in their listing. • The user who puts up a listing should be able to edit the status of a listing to "gone" once it's been given away. • A user can keep track of all their current listings or	Users can give away an item or offer it for borrowing by posting lists, which fall under two categories namely: Giving away and Borrowing. These lists can be edited after posting to make any further changes. A pick up location is added in the listing so that the user accessing the listing knows where to get the item from. The time for pick up of the item is sorted between the user giving away the item and the user requesting it. Since the system has no method of automatically detecting if an item is no longer available or already been given, the user who posts a listing has to update its status to "gone" after it has been given away or lended. Users can also keep track of all the listings they've posted and all their liked listings.

		past listings.	
		● The user can like a listing and view all the liked listings in one place.	
	3. Users should be able to access and request listings posted.	● Users can search for listings within a 40 km radius of their current location and view them. ● Users can choose between the list view and tile view to browse the listings. ● Users can filter the listings by adjusting the location of the listings to within 1 km, within 5 km, within 10km and within 20 km. They can also sort the filtered listings from newest to oldest to vice versa. ● The system should display a map showing all listings nearby (within 40 km radius).	Users can search through the listings posted on the app and filter them based on how near they want them to be. No user is shown a listing that's outside the 40km radius of their location. Users have the option of organizing the listings shown when searched in either the tile view or the list view. The list view simply lists the listings down without the thumbnail image while the tile view shows the thumbnail image. The app also has a map through which the user can navigate through the various listings.
	4. Users should be able to give away and take items.	● Users can give away an item under categories provided by the system.	The user can give away items that belong to various categories namely: Food, Clothing, Digital Appliance, Other. While the system grants the user the option to give away other things, it places a restriction on giving away certain items like prescription meds, weapons and any kind of living things.
	5. Users should be able to lend and borrow items.	● Users can post a public listing for a well usable item they wish to let other users borrow. ● Users can let other users borrow an item and restrict the borrowing to users with a profile rating	The borrowing lists have the same categories as giving away except with the exclusion of food as the application doesn't allow borrowing and lending of food. In addition, users can restrict the kind of users that can request to borrow their item according to their convenience. The

		of only 4 and above and location within 1 km. ● The time limit for listings posted for borrowing is limited and specified by the person who posts the listing.	user gives away an item for borrowing for a specific period which is stated beforehand but can be negotiated later.
	6. Users should be able to edit personal profiles and information.	● A user can edit their profile picture from the default icon to any picture in their phone or cloud storage. ● A user must add a nickname on their profile and can also add a short (150 characters) bio.	The application gives the users the ability to customize their profile by adding a nickname and description, as well as adding a profile picture of their own. The system only supports JPEG and PNG type images.
	7. Users should be able to message other users and request for listed items.	● Users can request to get a listed item by chatting with the user who posted the listing and settling on a pickup time and date. ● A user can access their old messages and archive them when they want.	The application allows users to chat with each other and the user who wants to request for a listed item can do it through the chat feature. Other information like exact pick up place, date and time can be discussed there as well.
	8. Users should be able to manage notifications.	● A user should be able to turn the notifications on or off for messages and ratings received.	The application sends a user notifications for every time they receive a message or a rating, or even when a new listing is posted in their area. Users have the option to turn these on or off besides being able to adjust the distance for which they'd like to receive the notification for the new listings posted.
	9. Users should be able to rate other users.	● The user who posted the listing can be rated out of five stars by any user who places a request for the listed item through chat. ● The user leaving the rating can also attach a review with it.	Users can be rated but only by the users who place a request to get the listed item. The user leaving a rating can also add a review of maximum 150 character length. The review and rating given will be visible on the lister's profile in the ratings and reviews of the user section.

	10. Users should be able to share listings and profiles.	• The users should be able to share a listing once viewed to social media platforms. • The users should be able to share the profile of other users to social media platforms.	Users can share a listing or the profile of another user on other platforms like meta platforms, gmail, twitter, discord, etc.
	11. Users should be able to change their location from the detected location.	• A user should be able to change their location from their current GPS detected location to any location they want on the map embedded in the app.	The application features a map that can be used to see all the listings nearby as well as give the users an option to change their location by dragging a location pin and dropping it on the new location they want to set as their current location instead of the GPS detected location.
	12. The users should be able to report harmful, inappropriate and offensive listings.	• The users should be able to report a listing under a particular category which states the reason for reporting.	The application allows the users to report a listing under various reasons namely: Profiting, Expired Item, Offensive to a particular demographic, Giving away living things and Other. If a user chooses the Other option as a reason when reporting, they must detail down the issue in a few lines.
	13. The users should be able to get help from the platform.	• The user can get help for their queries and technical issues by viewing the FAQ page in the system and asking other users in the comments section. • Users can get help by submitting a complaint or query to system support through the app's official mail.	The FAQ page consists of commonly asked questions followed by their answers and also new questions asked by the users which can be answered by other users. Besides this, the application provides the users support for any technical issue through the help center which can be reached through the app's official email.
	14. The users should be able to contribute to the app's development in the form of paid	• The user should be able to support the platform by subscribing to the app. • A donation system will be implemented whereby users will be presented	The application offers the users with two ways to monetarily support the environmental causes it stands for. For one, it gives them the option to directly support the platform and its prime cause by

	subscriptions.	with 2 or 3 donation causes such as an underwater cleanup, nature cleanup and other causes to further support sustainability of modern societies.	making paid subscriptions and then it also offers a donation system for external charity events that work towards other or more broad environmental causes.
System properties	1. The system shall confirm the email addresses of users and verify them.	● The system must send a confirmation email to the new user's mail after registration is initiated to complete the process and verify the user.	The user should be redirected back to the app after clicking on the verification link sent to their mail from where they can log in normally using the details put in the registration. The verification link is sent within the next 10 seconds of initiating registration and expires after 10 minutes.
	2. The system shall display date and time.	● The system must display the date and time of when a listing was first posted.	The system must sync itself with the date and time of the network provided date and time of the user and display it in every listing posted so users can gauge how old a listing is.
	3. The system shall keep track of and display views for listings.	● The system must keep track of and display the number of times a listing has been viewed by each unique user.	For the users to know how frequently a listing has been viewed and how popular it is, the system must show a view count on each of the listings which determines the number of times it has been viewed by a unique user.
	4. The system shall notify users of new listings.	● The system must notify the user about any new listings posted within a 2 km radius of them.	To keep the users updated with the listings near them, the system automatically notifies the user when a new listing is posted within a 2 km radius of them, if they choose to turn the option on in the notifications settings.
	5. The system shall be compatible with different platforms.	● The system should support cross platform functioning i.e. it should work on both iOS and android	The system is integrated with both iOS and android making the application a cross platform application. Users should be able to get the application from both the

		platforms.	Apple Appstore and Google Playstore.
	6. Identify new users.	● The system would identify new users with a new user sign next to their name.	If a user is new to the platform, the system would differentiate them with a 'new' tag displayed next to their name on their profile. This is done to distinguish between the users that are well acquainted with the platform and those who are still trying out its features for the first time.
Specific algorithms	1. Users cannot send unverified links through the application's chat feature.	● Users should be prompted with an error message in case they try to share any kind of unverified links in the chat.	In order to avoid phishing links and make the application more secure, the system does not allow sharing of unverified links through the app's chat feature. This includes any link with an unverified address or a description.
	2. The chat box should not hold more than 150 characters at once.	● The chat box can only hold 150 characters in one chat bubble. Any message longer would not be sent.	To keep the conversations precise and the communication to the point, the chat bubbles cannot exceed a character limit of 150.
	3. The help center must reply to the user query sent through mail within 24 hours.	● The help center must respond to a user's query through their email. ● The help center must respond to the query within 24 hrs after receiving the query.	The application provides the users with the convenience of a help center which works to ensure that all the queries of the users sent to them through mail are responded within 24 hours of sending. This makes the application efficient in dealing with technical problems.
Specific constraints	1. If a user's account is reported more than three times, the account is suspended.	● If a user is reported more than three times by three unique users, then the account is automatically suspended and all their data, including their lists, are removed from the app.	To ensure that the platform is safe for all, the system takes care of suspending accounts that are reported often to decrease potential threats and scams. One an account is suspended all their uploads are removed from the app as well.
	2. The system reminds the user to make use	● After 5 days of inactivity, the system reminds the	To bring in engagement and more active users, the system reminds

	of the app after 5 consecutive days of inactivity.	user to make use of the app again with a notification automatically sent.	inactive users to use the app again through automatic notifications that can be turned off in case the user does not wish to receive them.
	3. Once a user subscribes to the app, the user will be awarded a subscriber badge.	● A user is awarded a subscriber badge after purchasing the application subscription. ● The badge awarded is visible on the user's profile along with their other details.	To distinguish between the subscribers of the application with non subscribers, every subscriber is awarded with a badge that is visible on their profile. This is also done to show appreciation to the users who show direct monetary support to the platform.
Legal constraints	1. The system shall keep the chat information confidential.	● The application protects the user's data by keeping their chat information confidential. ● Chats are encrypted to ensure deeper confidentiality of information exchanged between two users.	Since users are likely to share some sensitive information in chat like personal addresses, phone numbers and even payment information, the application ensures that the chats are kept confidential and the messages are encrypted. This is done to follow the Personal Information Protection Act.
	2. The system must ensure the security of the user is protected by requiring the user to answer a security question of their choice every time they request to change the password.	● The system allows users to change their password only if they correctly answer a security question of their choice. ● The answer to the security question is stored at the time of registration.	When a user requests to change their password, the system prompts the user to answer a security question of their choice, the answer to which they would have already set during the registration process. If the answer entered during the change of password request shows to be the same as the one entered before, the system allows the user to change the password.
	3. The user should be of the age of 18 or above.	● The system should derive the user's age with the birthdate entered at the time of registration. ● This age should be of 18 or above for the user to successfully complete the registration process.	The application further ensures the safety of users by restricting its use to only people of the age of 18 or above. Even though the application doesn't involve sale, it works on similar grounds as trading and therefore follows the minimum legal age to trade.
	4. The user is required	● The system must detail	Just like any other sustainable

	to sign a license agreement which mentions the terms and conditions of the application when registering in the app.	down the terms and conditions of the app when a license agreement is made with the end users.	application, our app contains a list of its terms and conditions that a user must agree to when signing the license agreement at the time of registration.
	5. The system must not allow giving away and lending of any kind of prescription medications or drugs.	● The system prohibits the users from giving away or selling any kind of prescription medications. ● The system also prohibits the users from giving away or selling any kind of recreational drugs.	To go in accordance with the act on sale of drugs, the system does not allow listings of any kinds of drugs, whether recreational or medicinal.
	6. The system must take confirmation oaths from the users when needed.	● The system takes a confirmation oath from the users every time they post a listing to remind them that the system is not responsible for their items and any loss or damage is in the user's name. ● It also takes a confirmation oath during the signing up process to make sure the users agree with the terms and service of the application.	The application does not hold responsibility for any damage done to a user's listed item once it's given away. Since all items are given away on an individual basis, it's not feasible for the application to take responsibility for their state after giving away. Hence it makes the user sign a confirmation oath for agreeing that they are responsible for the item listed and not the application.

3. Non-Functional Requirements

Capability	1. The system should be able to process the request of 5,000 people's performance in a second without any decrement/crashes.
User Flexibiltiy	1. The design of a mobile app should be simple : minimum numbers of buttons, options, steps so that users can comfortably use the mobile app. Users should be able to post listings just in one step from the main page users frequently reach at first. 2. Users can download this mobile app and run on either iOS or Android, via mobile phones and tablets. Minimum requirements of versions are iOS 9.3.6 or Android 10. 3. This mobile app is only available in cities/towns in Selangor State, Malaysia.
Performance	1. The mobile app shall minimize the back-end complexities as much as possible for engineers in charge of the app to make and apply further updates smoothly to the system in the future. 2. The mobile app needs to save backup so that users' information is able to be restored with the backup files in an emergency.
Maintainability	1. The mobile app system shall load in less than 5 seconds of time for 98% of queries on both iOS and Android. 2. Users shall be able to send messages on chatting modes within 1 second of time on both iOS and Android.
Security	1. Only admin can manage and change system data, change permission for others : no other people should access the system without permission from admin. 2. The mobile app must be resilient to any kind of cyber attacks, such as DDoS or XSS attacks. 3. Users' information such as chat histories, personal information such as card information or phone numbers, etc. should be encrypted with AES encryption, end-to-end encryption, and so on for higher security and prevention of user information leakage.

Architectural overview

1. **Technical Architecture**

   For the application, our team has decided to use cross-platform development to propose our system. It can be used in either Android or an IOS phone, this can be easier for users to find our app from playstore and apple app store. Our team is using cross-platform to reduce the costs to develop applications, because creating native apps for different platforms will be expensive. Cross-platform will bring a better look and feel for user experience.

   Our application incorporates a microservice architecture rather than traditional monolithic and n-tier architectures to ensure fault isolation, faster deployment and lower use of resources. This means our application software is broken into several standalone components that work independently. All these components are then integrated into the overall system by the development team.
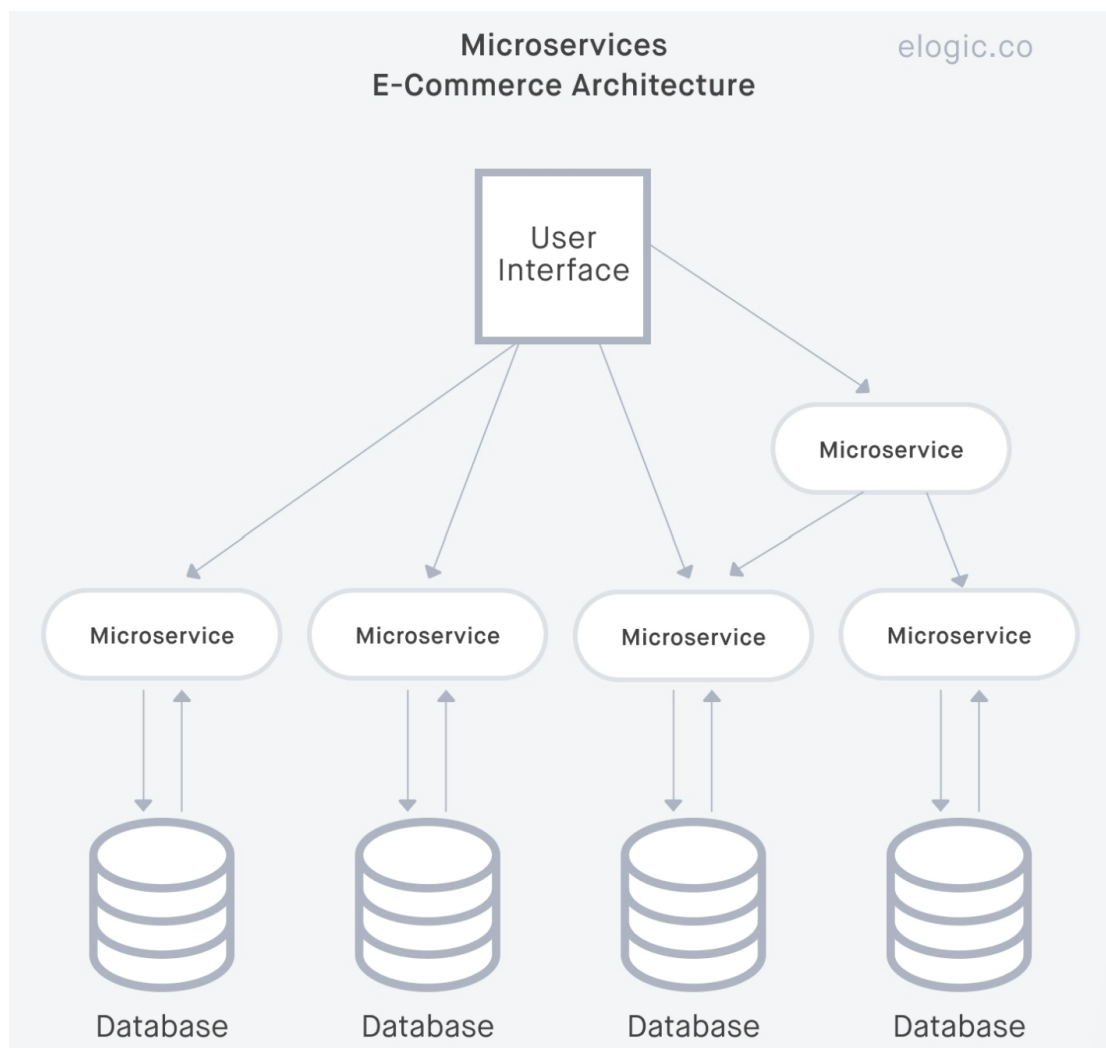
Due to its cross platform architecture, our application will be written in Java and Swift and the native code would be integrated with React Native to ensure its functionality on all platforms. React Native is a software framework that easily interfaces itself with the native code and ensures code reusability.

Our application will make use of Platform as a Service (PaaS) to host its data on cloud through cloud service providers like AWS. Since it works on a microservice architecture, it'll also make use of several services hosted on the cloud as Software as a Service (SaaS). In particular, our application makes use of AWS Amplify which is a full stack framework of a comprehensive set of SDKs, libraries, tools, and documentation for developing apps written in React Native.

We have made use of several useful technologies in our application like AI/ML to integrate visual and voice search features in our applications, enhancing the user experience and making the platform more user friendly.

2. User Interface (UI)

The technique that designers use to create user interfaces in software or electronic devices with an emphasis on aesthetics or style is known as user interface (UI) design. Designers strive to produce user-friendly and enjoyable interfaces. Graphical user interfaces and other types, such as voice-controlled interfaces, are referred to as "UI design."

When building an UI design, you should keep the following in mind:
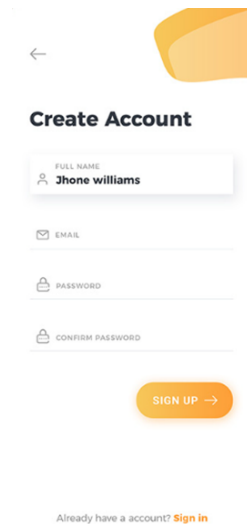
- Users judge designs quickly and care about usability and likeability and as a result, they are less concerned with your design and more concerned with doing their tasks quickly and efficiently.

- User interfaces should express brand values while also reinforcing user trust. Emotional design is good design. Users connect positive emotions with companies that communicate to them on several levels.

UI design for application

- When a user initially launches the application, he or she is greeted with the login screen with an option to sign up below which redirects them to the signup page, if they don't have an account associated with the app already.

Sign up –
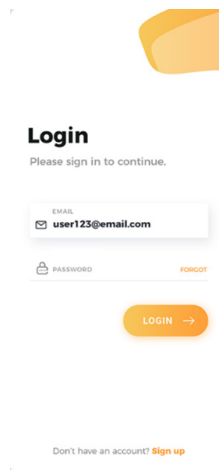
The user will be prompted to enter their name, email address, and a password for future logins.

Log In-

After successfully creating their account, the user will be taken to the login page, where they must enter the email address and password that they gave during the signup process.

App Permissions-

As soon as the user logs in, a window will appear requesting them to allow the app to find them by granting it permission to monitor the user using GPS.
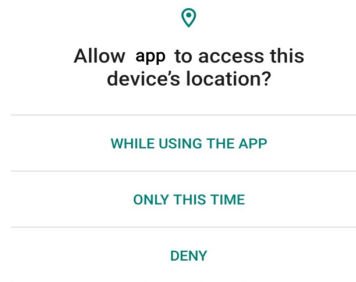
Setting up location-

When the user successfully provides the needed permission, it leads them to the app's integrated map to pinpoint where they reside and a search bar on top to make it easier to navigate using landmarks and a button at the bottom of the screen to finalize their address.
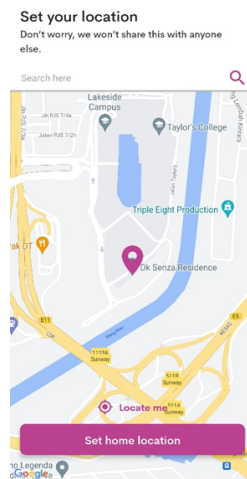
Supporting developers

After the user has successfully selected their location, the app displays a popup asking the user whether they want to support the cause of our app by giving a monthly/yearly subscription to help the developers maintain the app. The user can opt to support or continue to use the app.

Become a Supporter

✓ Unlock a special Supporter profile

Cheaper than a cup of coffee ☕ a month

| Yearly - save 50% | Best Value |
| RM 68.99 | |

Monthly
RM 10.99

Automatically renewed, cancel anytime. By selecting the
subscription, you agree to our terms & conditions and privacy
policy.

Continue
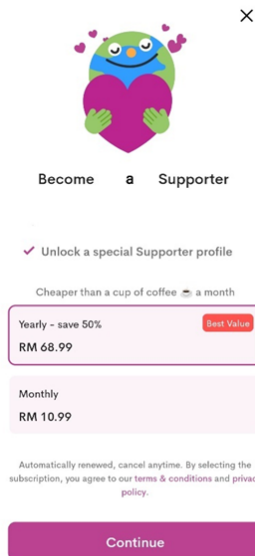
Home page –

After the user has completed all the formalities the user will be greeted with the homepage.

From the home page you can access various options such as:

● Listings uploaded by different users
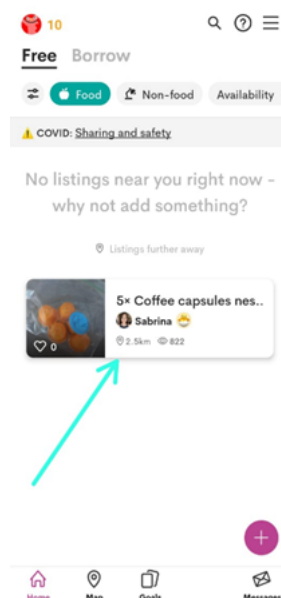
● Filters- to choose from such as – food item , non-food item, distance and sort by closest or newest listing
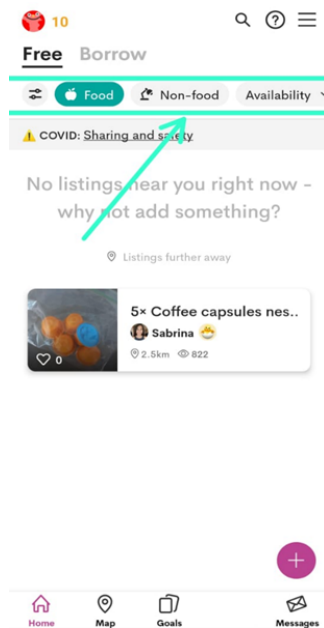
- **Hamburger menu-**

  In the hamburger menu there are many options like- profile tab, meals saved, my listings account, notification etc and you can further change your profile from the profile tab.
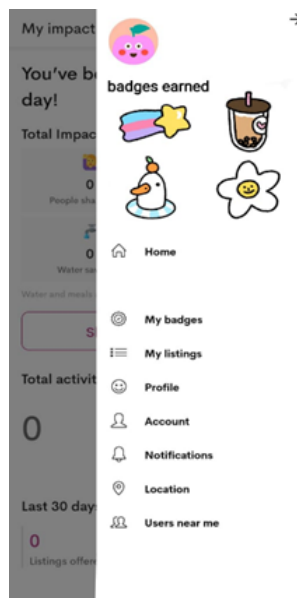
- **Borrow -**

  To borrow items there is a tab at the top which allows user to borrow items for a certain period of time
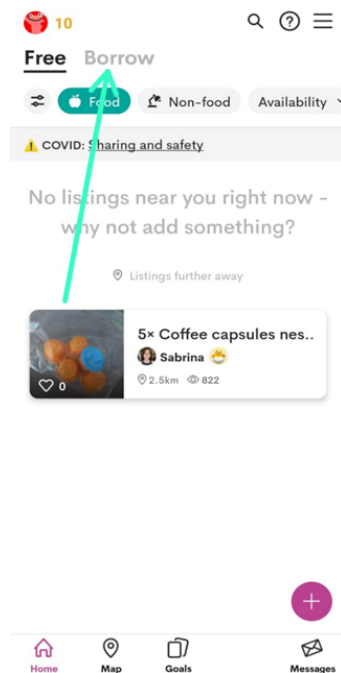
- Search –

  To search for a particular item with the filters given to get what you are looking for
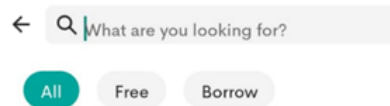
- Give/lend-

  This allows the user to give/lend his or her belongings to other users.
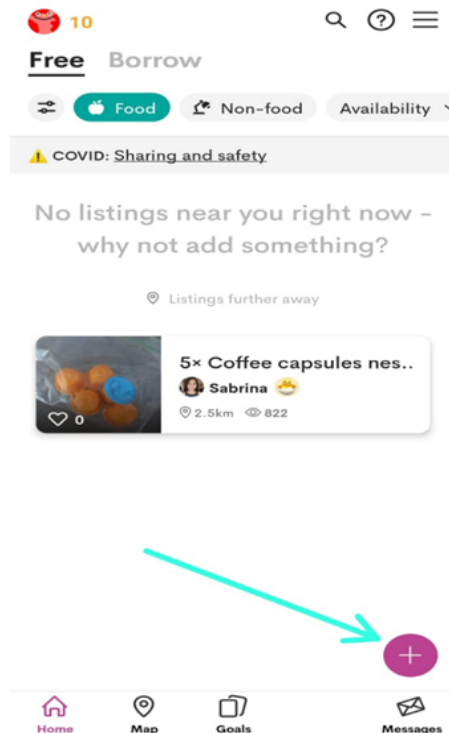
- Map

  This allows the user to see the listing posted nearby him/her. The maps tab can be found at the bottom of the screen next to home tab
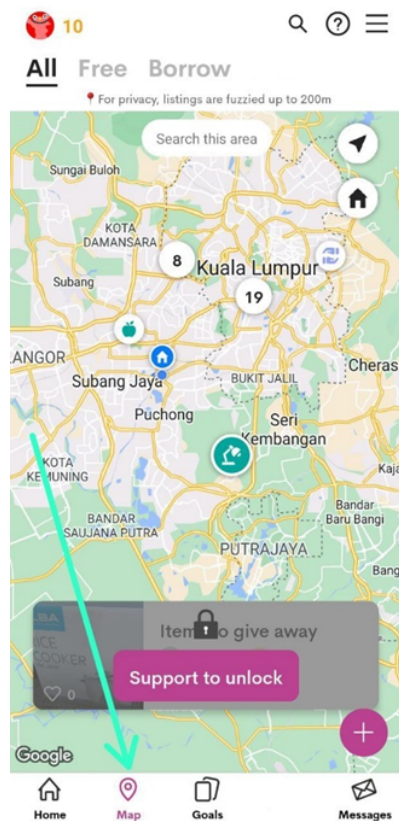
- buying page-

  on the buying/borrowing page you get various kinds of options like

  - sharing the listing with others
  - item picture and description
  - listing time
  - report tab(top right hand corner)
  - lenders approximate location
  - request tab
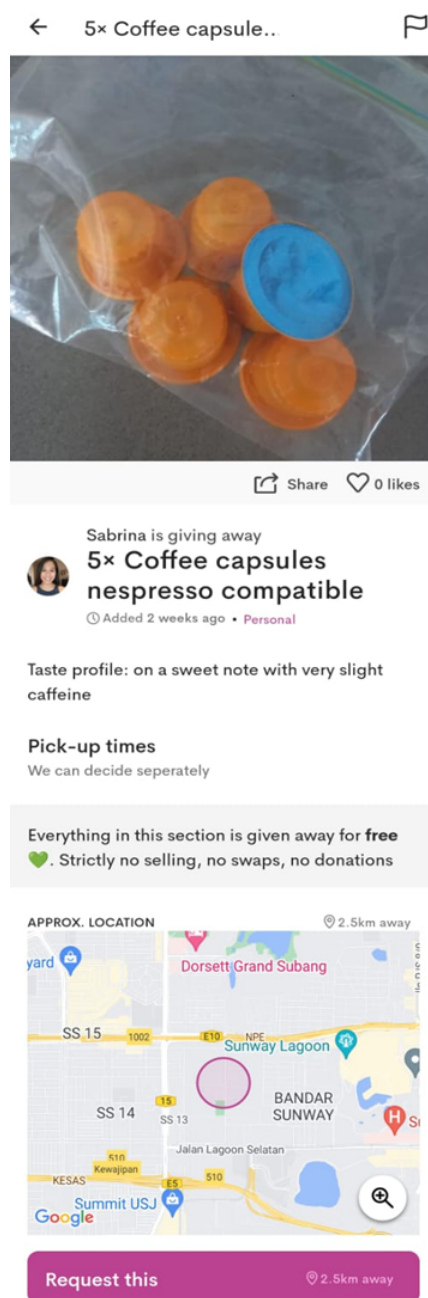
- Chat room-

  Once the user has found the item they need they can click on the request tab to enter a chat room with the donor/lender where the user will be able to text, view the profile of the lender/donor, listing popularity and to set a date, time and a place to pick up the listed items and the user can also send pictures in the chat as well.

  ←

  5× Coffee capsules nespresso compatible
  ⏱ Pick up: We can decide seperately

  Listing popularity 👍👍👍

  Sabrina
  📍 2.4km away
  Typically replies in 5 hours

  Send Sabrina a message...
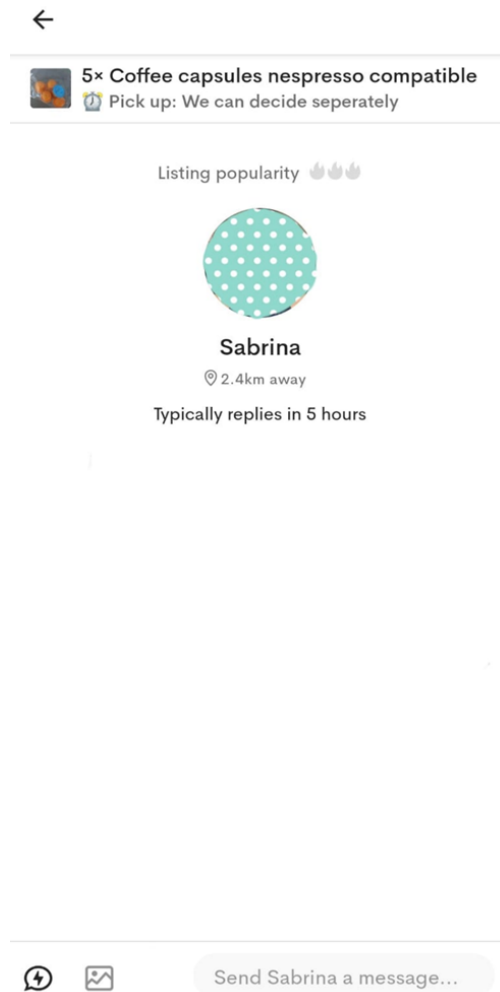
3. **User Experience (UX)**

   User experience is essential for every designer in order to provide users with a better feel and experience while interacting with our application. If an application is valuable but requires a great deal of time and effort, users will not be able to figure out how to utilize it, and they will abandon our application. Our team has used many ideas to our application in order to provide the optimal experience for every user or customer. The use of several UX concepts may make app usage simpler and more pleasant for users, as well as help them understand what they are doing.

   Our team has included intuitive navigation into our application. Providing users with navigational instructions is a crucial step for any application. Users may feel more at ease if the navigation is straightforward and easy to recognise. The navigation is clear, allowing the user to get acquainted with each navigation button and the destination that each button goes to. The navigation is also constant, allowing visitors to remember that it is always situated in the same location on every page.

   Next, we've included finger-friendly touch targets for each button in our application. Small buttons make the user experience more difficult, since they may be pushed incorrectly or accidentally. It is preferable to make the touch target larger and more apparent for consumers. Our team has designed each button to be the same size, between 8 and 10 mm, so that consumers can simply touch them with their fingers.

   In addition, the text inside our application is readable and intelligible. In comparison to laptops and desktop computers, mobile phones have smaller displays on which designers cannot fit the same amount of text on a page. So, our team has set the font size and line height to 11pt for tiny text and 14pt for big text, with a line spacing of 1.5 for all text. We will set the colour of the words and content to match the application's theme colour, which will be a colour that is visible and readable.

   Steven Hoober's study indicates that 49% of respondents often use just one thumb to execute activities on their smartphones. We will construct our application such that the menu, apply button, and delete/remove button are accessible to the majority of users. We positioned the navigation bar and several often used buttons for applying things where users can quickly reach them with their thumbs. On the other hand, we placed the delete or remove items button at the top, where it is difficult to reach, to avoid inadvertent presses while scrolling.

Our team has created the form to reduce the amount of text input. We will make the forms brief and straightforward to minimise the amount of time consumers spend inputting their information throughout the donation and borrowing processes. It is preferable to limit the amount of unneeded information on forms. Since typing on a smartphone is sluggish and prone to errors, our team included an auto-complete option to part of the user's information. Users just need to provide the very minimum of information, and the system will auto-complete and fill out the forms for them.

System Analysis and Design

1. Use Case Diagram
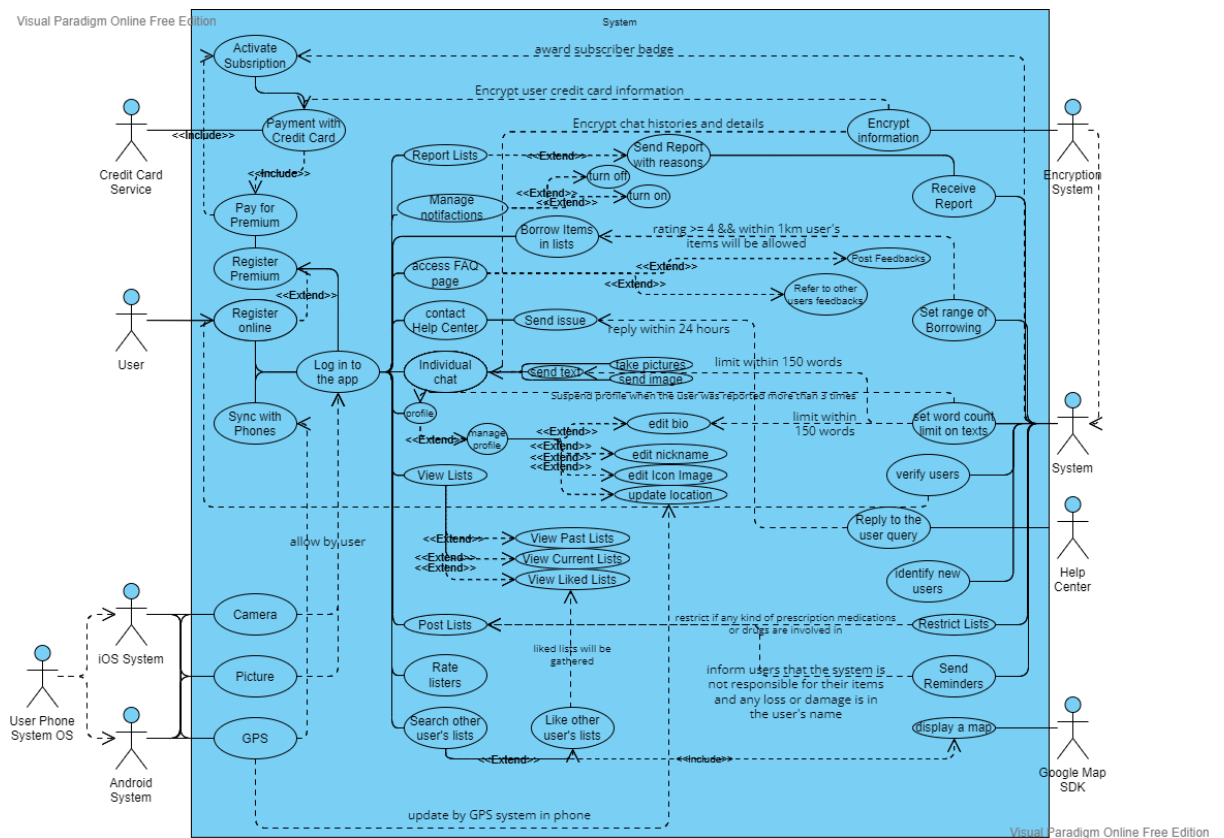
We have seven actors: *User*, *Credit Card Service*, *iOS System*, *Android System*, *Encryption System*, *System*, and *Help Center*. *iOS System* and *Android System* are divided from Actor: User Phone System OS.

First, the user needs to register online with the application to use. During the registration process, users will be verified by the system. After online registration, customers can either register for a premium subscription or not. When the user registers for a premium, they need to pay for a premium with a credit card, and Credit Card Service will complete the payment. After payment is made, System will allow the user to subscribe premium, award subscription badge that could be displayed on profile to the user. For the user would not register for a premium, they just log in to the mobile app after online registration. Premium subscription can also be made after the log in.

After logging in to the mobile application, the user are allowed to proceed to a couple of functions:

1. the user is allowed to go to their profile, edit their personal information such as bio, nickname, icon image, and update location. Location will be updated due to the data from GPS, which the user will sync in the registration.

2. the user is allowed to view lists. The lists the user can refer to are distributed into three categories: Past lists, Current lists, and liked lists. Lists that items are already passed to the borrowers/getters will be displayed on Past lists, and lists that items are still not passed to the borrowers/getters will be displayed on Current lists.

3. the user is allowed to post their own lists. However, if any kind of prescription medications or drugs are involved in the lists as items, those lists will be strictly proscribed by the system.

4. the user is allowed to alter their notification. If they turn on notifications, they will receive a wide variety of notifications including reminders to the user to make use of the app after 5 consecutive days of inactivity.

5. the user is allowed to communicate with others in individual chat rooms. In this system, chat is unified to only one-by-one chat, so there is no group chat. the user can send text and images -either take instantly with camera and send or select from gallery-. The text length is limited within 150 word counts.

6. the user is allowed to go to a help center to deal with issues they underwent or to ask what they are not aware of. The help center will send a reply to the user query within 24 hours.

7. the users are allowed to go to a FAQ page. In this page, the user can refer to other users' feedback, or post their feedback so others can refer to it.

8. If there are any irrelevant lists and when the user witnesses it, the user can report those lists. They could send a report about other users' lists with reasons, and the system will receive the report. When the user is reported more than three times, the profile will be suspended.

9. the user is allowed to search and look for other user's lists. they can display a map for search, but if the the user want to display a map, they must proceed to the page of searching other's lists. They can "like" other lists, and these "Liked lists" could be confirmed on "View Lists -> View Liked Lists".

10. the users are allowed to rate other users. The average rate above 4.0 and within 1 km user's items will be available to borrow or get.

This application is available on both iOS and Android operating systems. The users are required by the application system to sync their location, camera and gallery. Camera and gallery are required to sync in the stage of Log in, and GPS is required to sync in the stage of registration.

System is another important actor in this UML Use Case Diagram. System also has a couple of operations:

a. System receives reports of malicious or inappropriate lists. If the lists from the same user are reported a total of three times, then the system suspends the user who uploaded those lists.

b. System restricts borrowing for the user due to two conditions:

   i. the rating of borrower is above 4.0 out of 5.0
   ii. the item is located within 1 km from the borrower

c. System sets word count limits on texts. Chatting in individual chat rooms and description in the list are limited within 150 word counts.

d. System could restrict malicious lists. For example, if any kind of prescription medications or drugs are involved in the user' lists, the system will immediately ban them.

e. System always sends reminders to the user to inform them that the application is not responsible for their items and any loss or damage is in the user's name. If the user does not want to display this reminder anymore, the user can change the setting not to display this warning message.

f. When the user subscribes premium, the system awards, subscriber badge with activating subscription for the user.

Encryption system also depends on System. This actor always catches each message sent from the individual user to another, encrypting details to preserve information from leakage. Furthermore, when the user subscribes premium, this system also encrypts the user's card information.

Help Center receives any issues from the user, and replies to those user queries within 24 hours.

Google Map SDK is used to display a map when user search lists.

Credit Card Service is involved when the user wants to subscribe premium. The service implements the process of payment with credit card when the user inputs their card information, to activate subscription.

2. Class Diagram

User
- -userName: string
- -userAge: int
- -phoneNumber: long
- -emailAddress: string

Registration
- -nickname: string
- -age: int
- -phoneNumber: long
- -emailAddress: string
- -password: string
- -TermsNAgreements: longtext
- +setNickname(): void
- +setPassword(): void
- +setAge(): void
- +setPhoneNumber(): void
- +setEmailAddress(): void
- +syncWithGPS(): void
- +syncWithGoogleAc(): void
- +syncWithFacebookAc(): void
- +verifyEmailAddress(): boolean
- +verifyAge(): boolean
- +agreeTermsNAgreements(): boolean
- +subscribePremium(): void

Verification
- -emailAddress: string
- +sendVerificationEmail(): void

Premium
- -emailAddress string
- -phoneNumber long
- -paymentMethod string
- -cardNumber long
- +syncWithEWallet(): void
- +verifyCreditCard(): void

Credit Card Company
- -cardNumber long
- +receiveUserOTPInput(): void
- +verifyInputWithOTP(): void

Google Map SDK
- +displayMap(): void

Change Password
- -securityQuestion: text
- -securityAnswer: string
- +setSQ(): void
- +setSA(): void
- +answerSQ(): boolean

Log In
- -emailAddress: string
- -password: string
- +confirmEmailAddress(): boolean
- +confirmPassword(): boolean
- +syncWithCamera(): void
- +syncWithGallary(): void
- +changePassword(): void
- +subscribePremium(): void
- +getEmailAddress(): void
- +getPassword(): void

Search Items
- -userSearchKey: string
- +setLocationRange(): int
- +changeListView(): string
- +sortByDistance(): void
- +sortFromNewToOld(): void
- +filterResults(): void
- +likeOtherUsersLists(): void
- +reportOtherUsersList(): void
- +searchByMap(): void

Get Item
- -itemName: string
- -itemCategory: string
- -listCategory: string
- -quantity: int
- -images: binary
- -description: string[150]
- -posted: datetime
- -pickUp: datetime
- -tradeLocation: string

Profile
- -nickname: string
- -iconImage: binary
- -location: string
- -bio: text
- -rating: double
- +editNickname(): string
- +editBio(): binary
- +editIconImage(): string
- +updateLocation(): text
- +updateRating(): double

Home Page
- +goToChatroom(): void
- +switchNotification(): void
- +goToProfile(): void
- +goToSearchItems(): void
- +goToUserLists(): void
- +goToPostLists(): void
- +goToFAQPage(): void
- +goToHelpCenter(): void
- +goToReportUsers(): void
- +goToMap(): void

User Lists
- -itemName: string
- -itemCategory: string
- -listCategory: string
- -quantity: int
- -images: binary
- -description: string[150]
- -posted: datetime
- -pickUp: datetime
- -tradeLocation: string
- +shareOnSNS(): void

Back-end System
- +receiveReportOfUsers(): void
- +restrictPostedLists(): void
- +setWordCountOnTexts(): void
- +displayMapForSearch(): void
- +sendReminderToUsers(): void

FAQ Page
- -Feedback: text
- +postFeedbacks(): void
- +referFeedbacks(): void

Help Centre
- -issue: text
- +getUsersIssue(): void
- +sendReplyToUser(): text

Post Lists
- -itemName: string
- -itemCategory: string
- -listCategory: string
- -quantity: int
- -images: binary
- -description: string[150]
- -posted: datetime
- +shareOnSNS(): void

Individual Chat Room
- -ID: int
- -user1: string
- -user2: string
- -messages: Message
- +sendMessage(): Message
- +receiveMessage(): Message

Notification
- -on: boolean
- -off: boolean
- +turnOn(): void
- +turnOff(): void

Current Lists
- +shareOnSNS(): void
- +editList(): void
- +editListStatus(): void

Lists that items are still not given to somebody

Liked Lists
- +removeFromLikedLists(): void

Map
- -userSearchKey: string
- +syncWithGoogleMap(): void

Message
- -text: text
- -image: binary
- -sentDate: datetime
- -from: User
- -to: User
- -wasSent: boolean
- -wasRead: boolean

Past Lists
- -givenPerson: string

Lists that items are already given to somebody

Report users
- -reason: text
- +sendReport(): void

encrypt
- -text: text
- -image: binary
- +takeText()
- +takeImage()
- +doEncrypt()

decrypt
- -text: text
- -image: binary
- +takeText()
- +takeImage()
- +doDecrypt()
- +showText()
- +showImage()

Parent
child
child
child

This class diagram begins from the User class. User requires string type userName, int type userAge, long type phoneNumber, and string type emailAddress. the user could register for a couple of times with different email addresses or not make any accounts, so for each single user, Registration could be made 0..* times.

Registration class involves string type nickname, int type age, long type phoneNumber, string type emailAddress, string type password, and longtext type TermsNAgreements. The user needs to set a nickname, password, age, phone number, email address in each setter which are all void function types. User needs to sync their GPS to use this application. If a user wants to sync their Google or Facebook Account, they can sync. After all information is input, the user will receive a verification email to verify the one's email address. verifyAge() is the function to determine whether the user's age is below 18 or above 18, and if the user is below 18, the user cannot proceed with registration.

- Verification class receives string type emailAddress from the Registration class, and sends verification mail to the email address. Verification is possibly conducted by the user a couple of times.

Finally, the user is required to agree terms and agreements. If the user wants to subscribe premium, then the user can subscribe to it.

Premium requires string type emailAddress, long type phoneNumber, string type paymentMethod, and long type cardNumber.

- After the card number is input by the user, the Credit Card Company will receive the number and require the user to key in the OTP. When the OTP user key is correct, then the Credit Card Company allows the user to use the card.

String type emailAddress and string type password are required to Log In. User needs to confirm whether the emailAddress and password are correct, and if the user forgets the password, the person can change the password with a security question and answer. This change might not be caused, or might be caused multiple times. If the user wants to subscribe premium at the Log In stage, the user can subscribe to it.

Finally, the user is required to sync the Camera and Gallery of their phone. These Log In operations might not be caused, or caused a couple of times after registration.

Home Page class has nine operations, each operation is used to access each function in this application, and all of these functions are possibly implemented for multiple times or zero times:

1. Void type goToChatroom() is the function that lets the user proceed to an Individual Chat Room class. Individual Chat Room has int type ID, string type user1 and user2, and Message type messages. the user can send or receive texts and images as messages, and the sent date time and read date time remain so the user can confirm those date time information. Each message is encrypted to prevent leakage, only decrypted to the user in the chat room.

2. Void type SwitchNotification() is the function that lets the user switch notifications. turnOn() and turnOff() are used to switch.

3. Void type goToProfile() is the function that lets the user view and edit the profile. Profile involves string type nickname, binary type iconImage, string type location, text type bio, and double type rating. Location will be updated with the city/town name, prefecture name and country with the information from the user's mobile phone system. Rating will be dynamically altered by the average rating from other the user. for nickname, iconImage and bio, the user can change by oneself.

4. Void type goToSearchItems() is the function that lets the user search other the user' items in lists. Google Map SDK will display a map for the user searching items, so the user can use a map displayed, and the system receives the string type the userSearchKey. The user is allowed to filter results by the range of distance, sort results by distance or new-to-old, change view of lists, and place like on other users' lists.

   - Get Item class is linked with Search Items. After the user finishes searching, the user can proceed to negotiate with owners and to get items. String type itemName, itemCategory, listCategory, int type quantity, binary type images, string[150] type description, datetime type posted, datetime pickUp, and string type tradeLocation. Pick up

date time and location for trade will be determined with the owner of items on the individual chat room. Even though the user searches for items, the user either gets multiple items or does not get any items.

5. Void type goToUserLists() is the function that lets the user view some kinds of lists. The user is allowed to view current lists -lists that are posted but items are not passed to opponents yet-, past lists -lists that are posted and items are also already passed to opponents-, and liked lists -other users' lists that the user has placed like on-. These three types or lists are all child classes of User Lists, so each list has string type itemName, itemCategory, listCategory, int type quantity, binary type images, string[150] type description, datetime type posted, datetime pickUp, string type tradeLocation, and other independent attributes or operations (Current Lists include void type shareOnSNS(), editList() and editListStatus(). Past Lists include string type givenPerson, and Liked Lists include void type removeFromLikedLists().).

6. Void type goToPostLists() is the function that lets the user post lists with items. To post lists, the user must include string type itemName, itemCategory, listCategory, int type quantity, binary type images, string[150] type description, and datetime type posted. When posting the lists, the user can share lists on social media.

7. Void type goToFAQPage() is the function that lets the user proceed to the FAQ Page. The user can post text type feedback, or refer to other users' feedback.

8. Void type goToHelpCenter() is the function that lets the user proceed to the Help Center. The user can post text type issues, and Help Center will receive those issues and will send replies to each users' queries in text type.

9. Void type goToReportUsers() is the function that lets the user report other users. The user is required to send a report with text type reason, and the back-end system will receive the report.

10. Void type goToMap is the function that lets the user search on a map. Map receives string type userSearchKey, displaying a map by syncing with Google Map for the user.

Back-end System has some powerful and crucial operations:

a. System can receive reports of users. When the individual user receives a total of three times of reports, then the system will suspend the user.

b. If any kind of prescription medications or drugs are involved in the user' lists, the system will immediately ban them.

c. System always sends reminders to the user to inform them that the application is not responsible for their items and any loss or damage is in the user's name. If the user does not want to display this reminder anymore, the user can change the setting not to display this warning message.

d. System sets word count limits on texts. Chatting in individual chat rooms and description in the list are limited within 150 word counts.

e. System restricts borrowing for the user due to two conditions:

   i.   the rating of borrower is above 4.0 out of 5.0
   ii. the item is located within 1 km from the borrower

3. Object Diagram

Case 1

```
                              Hakeem: User
                              ┌──────────┐
                              │          │
                              └──────────┘
                                   │
   Hakeem: Verification      R: Registration       Credit Card
   ┌──────────────┐          ┌──────────┐          Company
   │              │──────────│          │──────────┌──────────┐
   └──────────────┘          └──────────┘          │          │
                                   │                └──────────┘
                              L: Log in
                              ┌──────────┐
                              │          │
                              └──────────┘
                                   │
                              H: Home Page
                              ┌──────────────┐
                              │+goToUserLists()│
                              └──────────────┘
                                   │
```

UL1: User Lists

itemName = bag
itemCategory = personal item
listCategory = giving away
quantity = 1
image
description: "This is the bag I have purchased in ZARA.
             I have used this for three years.
             No dusts, and no cracks or scratches."
posted = '2022-11-25 11:15:15.115'
pickUp
tradeLocation

+shareOnSNS()

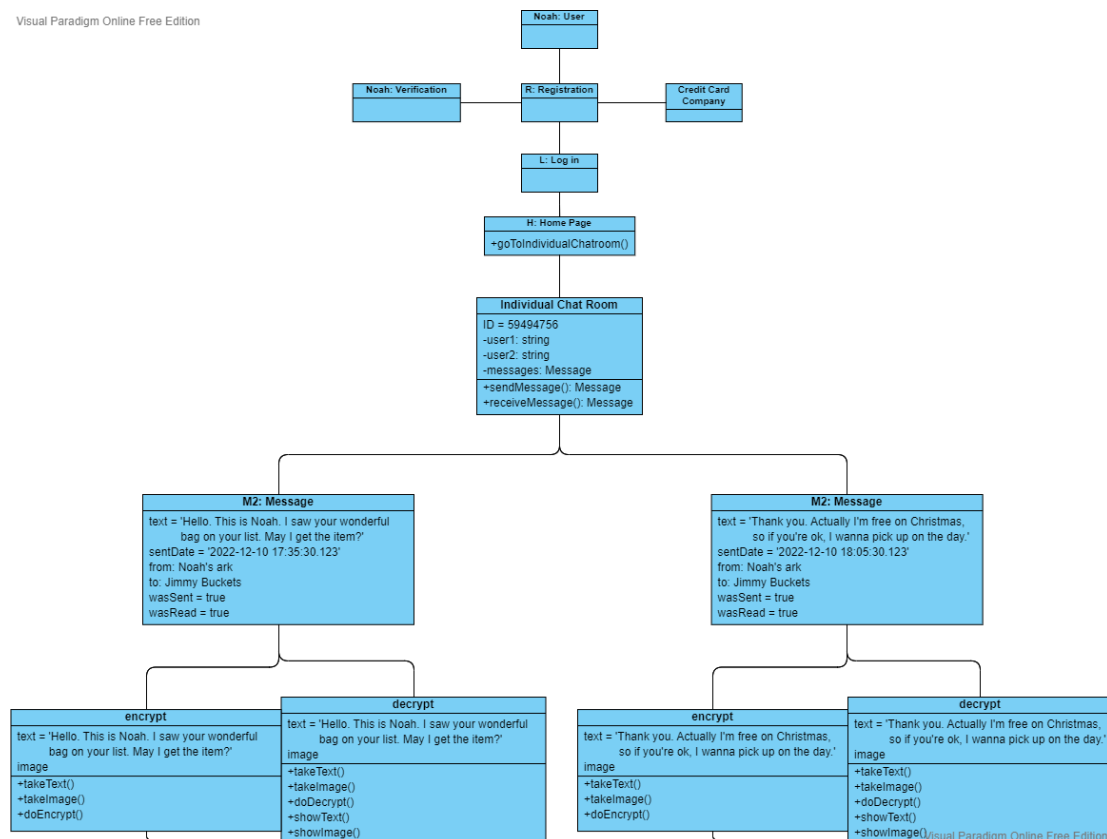**UL1: Current Lists**

+editList()
+editListStatus()

In this case, Hakeem Olajuwon will be the user. He is 34 years old, email address is dream34@gmail.com, and phone number is 60130349495.

Hakeem registers with the information above + nickname as "Dream", password as "dream34". Email address verification is successfully done, and he proceeds to the Home Page. Hakeem decided to go to User Lists.

He has one Current List. Name of item is bag, category is personal item, and category of list is "giving away". Quantity is only 1, and no image is posted with. Description "This is the bag I have purchased in ZARA. I have used this for three years. Nodusts, and no cracks or scratches." is also added, and the posted date time is 11:15 on 25th November, 2022. Pick up date time and location for trade are not determined.

Case 2

Noah: User

Noah: Verification — **R: Registration** — **Credit Card Company**

L: Log in

H: Home Page
+goToIndividualChatroom()

Individual Chat Room
ID = 59494756
-user1: string
-user2: string
-messages: Message
+sendMessage(): Message
+receiveMessage(): Message

M2: Message
text = 'Hello. This is Noah. I saw your wonderful bag on your list. May I get the item?'
sentDate = '2022-12-10 17:35:30.123'
from: Noah's ark
to: Jimmy Buckets
wasSent = true
wasRead = true

M2: Message
text = 'Thank you. Actually I'm free on Christmas, so if you're ok, I wanna pick up on the day.'
sentDate = '2022-12-10 18:05:30.123'
from: Noah's ark
to: Jimmy Buckets
wasSent = true
wasRead = true

encrypt
text = 'Hello. This is Noah. I saw your wonderful bag on your list. May I get the item?'
image
+takeText()
+takeImage()
+doEncrypt()

decrypt
text = 'Hello. This is Noah. I saw your wonderful bag on your list. May I get the item?'
image
+takeText()
+takeImage()
+doDecrypt()
+showText()
+showImage()

encrypt
text = 'Thank you. Actually I'm free on Christmas, so if you're ok, I wanna pick up on the day.'
image
+takeText()
+takeImage()
+doEncrypt()

decrypt
text = 'Thank you. Actually I'm free on Christmas, so if you're ok, I wanna pick up on the day.'
image
+takeText()
+takeImage()
+doDecrypt()
+showText()
+showImage()

In this case, Noah Smith will be the user. He is 18 years old, email address is noahsark950@gmail.com, and phone number is 60131220950.

Noah registers with the information above + nickname as "Noah's ark", password as "noahsark950". Email address verification is successfully done, and he proceeds to the Home Page. Hakeem decided to go to the Individual Chat Room that he is communicating with Jimmy Butler.

He has sent two messages. The first message is "Hello. This is Noah. I saw your wonderful bag on your list. May I get the item?" The second message is "Thank you. Actually I'm free on Christmas, so if you're ok, " Each message does not include any images, encrypted and decryption is only conducted for him and the receiver, Jimmy.

Case 3
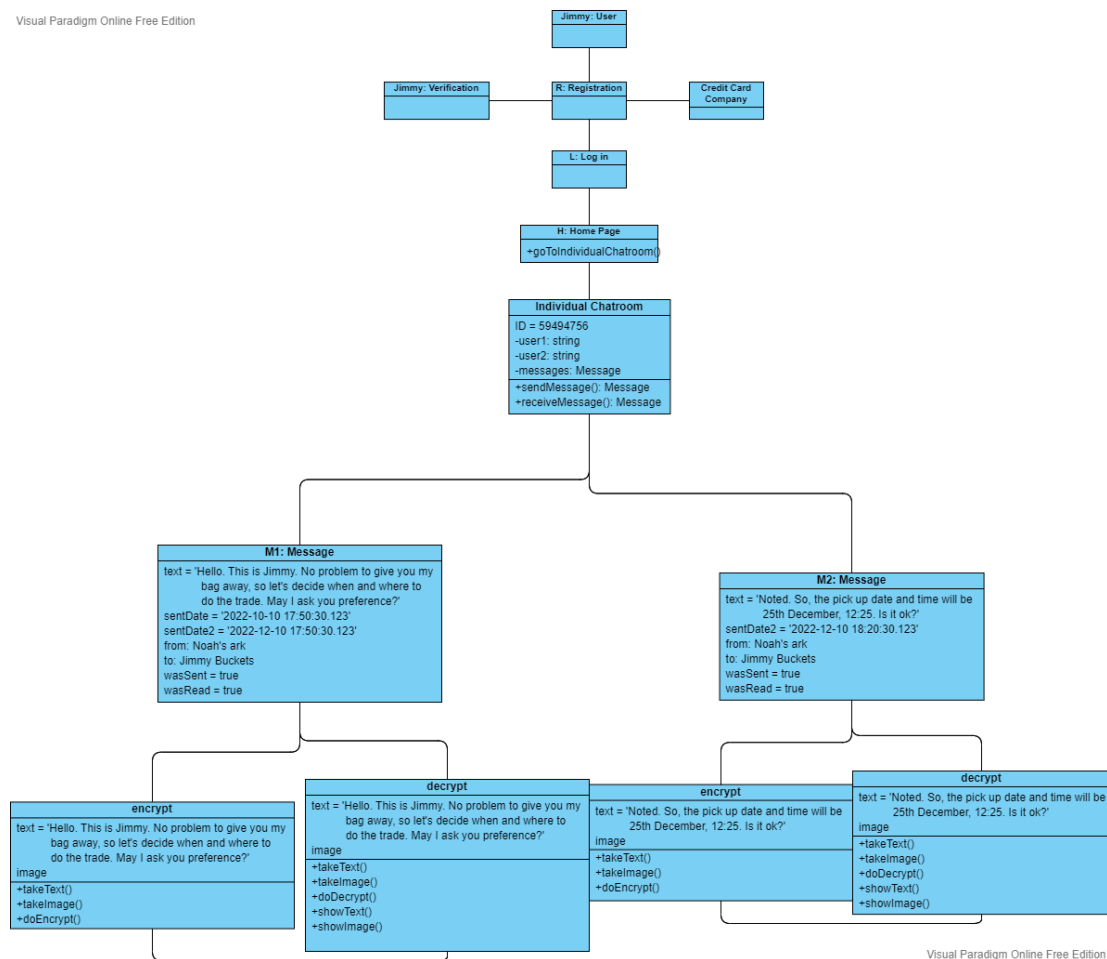
In this case, Jimmy Butler will be the user. He is 22 years old, email address is jimmybuckets22@gmail.com, and phone number is 601321221989.

Jimmy registers with the information above + nickname as "Jimmy Buckets", password as "jbuckets22". Email address verification is successfully done, and he proceeds to the Home Page. Jimmy decided to go to the Individual Chat Room that he is communicating with Noah Smith.

He has sent two messages. The first message is "Hello. This is Jimmy. No problem to give you my bag away, so let's decide when and where to do the trade. May I ask your preference?" The second message is "Noted. So, the pick up date and time will be 25th December, 12:25. Is it ok?" Each message does not include any images, encrypted and decryption is only conducted for him and the receiver, Noah.

Case 4
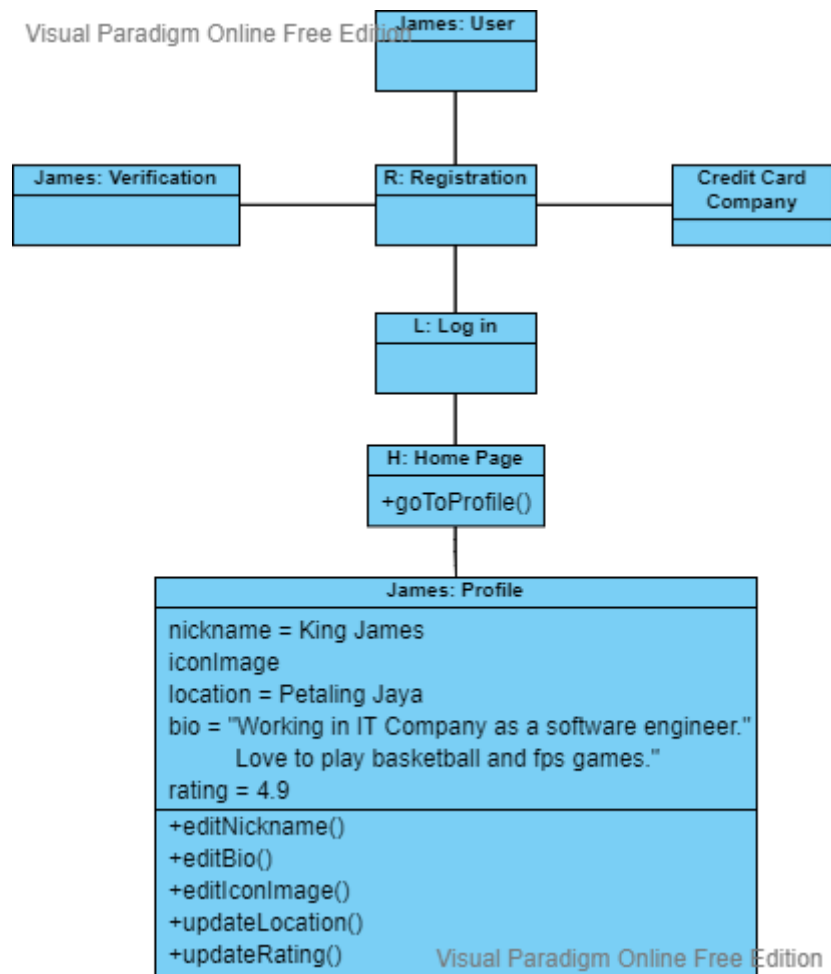
The last case is for James Corden. He is 23 years old, email address is kingjames23@gmail.com, and phone number is 60130231982.

James registers with the information above + nickname as "King James", password as "kingjames23". Email address verification is successfully done, and he proceeds to the Home Page. James decided to go to the Profile.

He could view his nickname as "King James", but he did not set any icon image. His location is Petaling Jaya, bio description is "Working in IT Company as a software engineer. Love to play basketball and fps games." His average rating is 4.9.

Conclusion and Future Enhancement

To conclude our ConserveR app aims to create a visible impact in society and see a reduction in the wastage of disposal and create social awareness through a promotional media plan to create publicity and make the app a worldwide movement which aligns with the sustainable development goals set by the united nations such as climate change and responsible consumption and disposal to make a physical difference and impact on the community we live in. This is because our app is supported by sponsors globally who are our patrons of this app and other well established organizations who wish to make an aimful impact. This app was curated carefully to establish what the user would expect to perform their requests like making requests from people who wish to give away and giving users of the app the ability to give away the items that they wish to dispose of. This app has to comply with government legislations set by food safety standards as our platform also aims to reduce food wastage to help with the problem of food shortage in the world. Therefore this app helps solve a multitude of problems parallelly to improve the world and make it a better place.

As we all know, nothing in this world is perfect, and neither is our software, but we work every day to make it a bit better and easier for our customers so they can have a better experience.

In the future, We want to address the majority of the issues in our application and to host an online forum for our app where we can solicit feedback on how we can improve our app. and the development team plans to include an in-app voice call system for increased privacy and an in-app navigation system to help users to meeting locations. Once the app gets a sufficient number of subscribers, we want to give 20 - 30% of the app's revenues to a charity of the users choosing . We are also attempting to minimize the size of our app by removing redundant elements in order to lessen our carbon footprint.

Rubric

7.0 Marking Scheme for Assignment 1

Criteria	Marks (10 marks for each criterion below)			
	Excellent (10-9)	Good (8-7)	Average (6-4)	Poor (3-1)
1.0 Introduction				
2.0 Problem, Solution & Innovation				
3.0 Solution & Innovation				
4.0 Social impact of your project & business viability				

5.0 Systems Requirements
  I.   Requirements Gathering & Elicitation, Scope Definition
  II. Functional Requirements
  III. Non-functional Requirements				

6.0 Architectural Overview
  I.   Technical Architecture
  II. User Interface (UI)
  III. User Experience (UX)				

7.0 Systems Analysis & Design
  I.   Use Case Diagram
  II. Class Diagram
  III. Object Diagram				

8.0 Conclusion & Future Enhancement				
9.0 References & Document Flow and Presentation				
10.0 Individual Participation				
SUB-TOTAL (100 marks)				
Total Marks (20 marks)				
Lecturer's Feedbacks:				

References

Shvetsova, Y. (2022) *Functional and Non-Functional Requirements of Online Shopping System: List & Examples*. [online]. Available at: https://elogic.co/blog/functional-and-non-functional-requirements-for-ecommerce-websites/ (Accessed: 15 September 2022)

UN Department of Economic and Social Affairs. () *SDG 12: Ensure sustainable consumption and production patterns*. [online]. Available at: https://sdgs.un.org/goals/goal12 (Accessed:)

Nishtha, M.() *Requirements Gathering – How We Solve The Biggest Problems With Consulting.* [online]. Available at: https://nmgtechnologies.com/blog/requirement-gathering-solve-biggest-problems-consulting.html (Accessed: 9 October 2022)

Emily, N. (2021) *Architectures for Modern eCommerce Applications.* [online]. Available at: https://hygraph.com/blog/architectures-for-modern-ecommerce-applications ()

By the Admin of RBN/. (2015) *Why SaaS and Microservices are Critical to Developing in the Cloud.* [online]. Available at: https://www.rightbrainnetworks.com/2015/01/29/why-saas-and-microservices-are-critical-to-developing-in-the-cloud/#:~:text=Microservices%20are%20the%20resulting%20standalone,part%20of%20a%20larger%20system ()

Understanding eCommerce. (2021) *Top 11 eCommerce Mobile App Technologies to Use in Your Business*. [online] Available at: https://understandingecommerce.com/top-11-ecommerce-mobile-app-technologies-to-use-in-your-business/ ()

Prashant, R. (2018) *The Microservices Approach to Mobile Application Development (Part I).* [online]. Available at: https://medium.com/hackernoon/microservices-architecture-for-mobile-application-development-part-i-20b4f4089a24 ()

Yuliia, S. (2022) Ecommerce Website Architecture: Why You Should Care About It. [online]. Available at: https://elogic.co/blog/ecommerce-architecture/ ()

By Enago Academy. (2021) *Writing an Impact Statement: Four Things You Need to Know*. [online]. Available at: https://www.enago.com/academy/writing-an-impact-statement-four-things-you-need-to-know/ ()

Beatriz, E. () *Chapter 2 How To Determine Market Viability and Conduct Product Research (2022 Guide).* [online]. Available at: https://www.bigcommerce.com/blog/evaluate-product-market-viability/#undefined ()

Bradie, S. (2011) Helping Your Nonprofit Organization Stay Viable During Tough Economic Times. *Grand Valley State University ScholarWorks@GVSU*, 25(), 4-9.

Tutorialspoint. () *SDLC - Agile Model*. [online]. Available at: https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm ()

Tutorialspoint. () *SDLC - Waterfall Model*. [online]. Available at: https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm ()

Geeksforgeeks. (2021) *Software Engineering | Requirements Elicitation*. [online]. Available at: https://www.geeksforgeeks.org/software-engineering-requirements-elicitation/ ()

The World Bank. () *WHAT A WASTE 2.0 - A Global Snapshot of Solid Waste Management to 2050*. [online]. Available at: https://datatopics.worldbank.org/what-a-waste/trends_in_solid_waste_management.html ()

javaTpoint. () *Software Development Life Cycle (SDLC)*. [online]. Available at: https://www.javatpoint.com/software-engineering-software-development-life-cycle ()

Sarah, L. (2019) *waterfall model*. [online]. Available at: https://www.techtarget.com/searchsoftwarequality/definition/waterfall-model#:~:text=The%20waterfall%20model%20is%20a,a%20logical%20progression%20of%20steps ()

Lindsay, M. (2019) *The Role of Color in Product Design: UX of Color Palettes*. [online]. Available at: https://xd.adobe.com/ideas/principles/web-design/ux-of-color-palettes/ ()

Sandeep, K. (2022) *Mobile App Login Signup UI*. [online]. Available at: https://www.uplabs.com/posts/mobile-app-login-signup-ui ()

Cheryhappygirl. (2018) *How To Use Color In UI Design Wisely to Create A Perfect UI Interface?* [online]. Available at: https://blog.prototypr.io/how-to-use-color-in-ui-design-wisely-to-create-a-perfect-ui-interface-6e524bd023bc ()

Interaction Design Foundation. () *User Interface Design*. [online]. Available at: https://www.interaction-design.org/literature/topics/ui-design ()

Individual Participation

Ishihara Satoaki	0354208	- System requirements - Non-functional requirements - System requirements - Requirements Gathering' Elicitation, Scope Definition - System Analysis and Design - UML Use Case diagram - UML Class diagram - UML Object diagram
Aman Mahmud Bin Amer Mahmud	0348725	- System requirements - Requirements Gathering' Elicitation, Scope Definition - System requirements - Functional requirements
Mohammad Fahad Farhan	0354487	- Introduction - Problem Statement - Solution - System requirements - Functional requirements - Conclusion and Future Enhancement
Mohammad Sameed Khan	0353846	- System requirements - Non-functional requirements - Architecture Overview- User Interface - Conclusion and future enhancements
Thua Sin Wei	0354566	- System requirements - Non-functional requirements - Architecture Overview- Technical Architecture - Architecture Overview - User Experience
Jinan Nisar	0354690	- Discuss impact of solution - Discuss business viability application - System requirements - Functional requirements - Architectural Overview - Technical Architecture