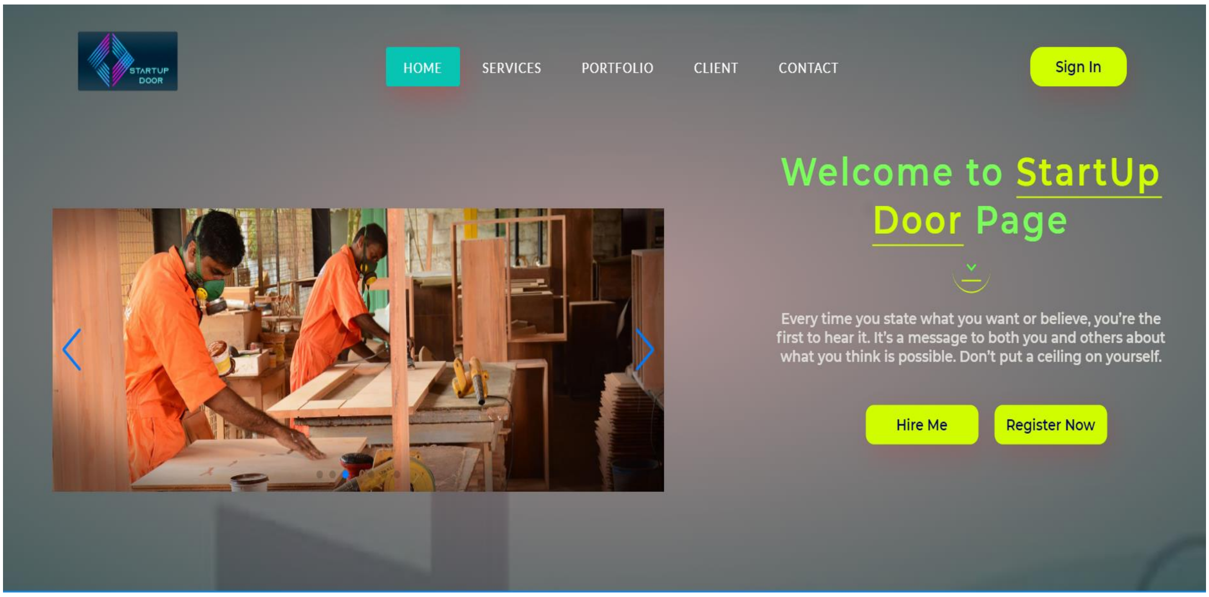


Welcome to StartUp Door Page

Every time you state what you want or believe, you're the first to hear it. It's a message to both you and others about what you think is possible. Don't put a ceiling on yourself.

Startup Door
Group 54 – Third Year Group Project

Details of Project Supervisor, Co-supervisor, Advisors, and Clients

Project Supervisor (Academic Staff of UCSC):

Name of the supervisor: Dr. S.S.P Mathara Archchi

Signature of the supervisor:

Date:

Project Co-Supervisor (Assigned by Course Coordinator):

Name of the co-supervisor: Ms. Tharushika Perera

Signature of the co-supervisor:

Date:

Project Advisors: (External industry advisors, if any)

(Please provide, Name, Organization, email address, and institute)

1. …………………………………………………………………………………………….
2. …………………………………………………………………………………………….
3. …………………………………………………………………………………………….

CS 3214 / IS 3113: Group Project II - 2021

Interim Report

Project Title:

Project Group Details

1. Group number: **G54**
2. Group members: List of Members Below

Name	*Reg. Number*	*Index Number*	*Email address*	*Mobile Phone*
W.P Pallewatta	2018/CS/114	18001149	pandula.pallewatta@aiesec.net	+94715979070
R.P.P.B.Somarathna	2018/CS/163	18001637	pasanbasitha9733@gmail.com	+94 771222541
D.T.Athukorala	2018/CS/017	18000177	tharakaathukorala21@gmail.co m	+94717963873
M.N.J Rathnayaka	2018/CS/133	18001335	mnjrathnayaka97@gmail.com	+94713559368
D. M. Samarasinghe	2018/IS/067	18020674	darshanamsamarasinghe@gmail .com	+94774678712
W.P Pallewatta	2018/CS/114	18001149	pandula.pallewatta@aiesec.net	+94715979070

The client of the Project (If applicable, otherwise supervisor will be considered as the client)

Name of the client: Dr. S.S.P Mathara Archchi

Address of the client:
University of Colombo School of Computing
UCSC Building Complex,
35, Reid Avenue, Colombo 7
SRI LANKA

Contact person at client: Dr. S.S.P Mathara Arachchi

The contact number of the contact person:+94773515017

E-mail address of the contact person: ssp@ucsc.cmb.ac.lk

Project Details:

1. Project Title:

Startup Door

2. The Goal and Objectives:

Goal: - Creating a Passive income mechanism for the middle-class community to uplift their lifestyle, provide opportunities to the users in different levels of services, and a platform to create new opportunities to build entrepreneurs for the society

Objectives: -

- Introduce new online social entrepreneurs to the community.
  - Providing new business opportunities and jobs.
  - Uplifting people's lifestyle providing necessities.
  - Opportunity to expand & build inter-network between client and service provider.
  - Convenient for the people who needed Social-basis services in their endeavours.
  - Obtaining an extra income without condemning huge capital from the service provider's end.

3. Problem Definition

- People don't have an online enterprise approach that combines the desired social cause with the picture of creativity, ingenuity, and commitment.
- Many middle-class people can't make a large investment in a business; However, the community is eager to take opportunities with minimal capital risks.
- People like to earn extra income without risking their capital and time. In such a case, there's is a lack of a legit approach to get these privileges
- People wanted to find temporary-based services to do their chores for their end.
- People seeking part-time employment in their spare time but don't have a good means to obtain a job that they want.

4. A Brief Introduction to the Project

A platform to create a bridge between the Service provider and a service consumer. Our consideration is to give the middle-class community a passive income opportunity (As a Service approach) with social entrepreneurship-based discipline.

Priority is to meet the necessary parties who need service according to the location he/she is and necessary service via **Startup Door**. From this approach, **Startup Door** provides social entrepreneurship discipline to all the users of our system and gives them an adequate opportunity to build their startups and businesses.
Startup door mainly provides services under these three categories which are,

3

- Technician Services
- Plant & Crop Services
- Food & Cuisine Services

Also, if a customer requests a service category besides these, can put a request to system-admin. **Startup Door** will be upgraded to provide a new service considering adequate service availability of the client's recommendations.

5. The Scope of the Project

IN-SCOPE:
The following features are expected to be developed in the system:

Users (possible actors) of the system:

- Service provider (Technicians, Plants & Crops, Food & Cuisine)
- Client
- Admin

Main functionalities of the system:

Login

- Two-step verification Process
- Email validation

Service provider

Service providers have to pay a monthly subscription fee according to the basic plan that **Startup Door** offers. If they provide more than one service, they should have a selection procedure and need to pay a relevant amount according to their plan.

- Divide service providers into main subcategories such as technicians, cuisine, and plants & Crops.

- Location-based tracking is provided to maximize the client’s reach.
- Create an account and customize account preferences
- Share their current GPS locations and availability
- Manage Cancellation Process
- Price comparison process with the highest price (bidding process) and notify the vendor.
- Profit tracking Mechanism

For technician

- Service providers can provide contract basis offers to clients.
- A service provider can supervise workers via a tracker and discard if some client does not provide the necessary work.

For cuisine

- They can add a menu (breakfast, lunch, dinner) and type of meal (vegetarian,chicken, etc.), with clients’ preferences

For plants & Crops

- Plants - Plant catalogue with availability details
- Crops - categories along with management mechanism

Client

- Search for relevant services based on their requirement and preferences
- Search and view the Service providers profiles
- Give feedback and rating value to service providers.
- Profile customization
- After the payment, upload the bank slip.
- Cancel the ongoing business process (But consisting of limitations & regulations)
- Rating basis promotions according to clients recommendations

Admin

- Add a new service component.
- Report generation management.
- Manage project managers
- Manage archives.

Product Manager

- Manage service providers and clients.

- Service management (update service/ view service feedback)
- Sending Information to the customers.
- Service providers will be ranked under their rating.

6. Feasibility Study

Feasibility is an important phase in the software development process, it enables developers to have an assessment of the product being developed. It refers to the feasibility study of the product in terms of outcomes of the product, operations required for implementing it. A feasibility study should be performed based on various criteria and parameters.

In this section, we have identified and categorized different resources required to carry out the project. The various feasibility studies are:

- Technical Feasibility
- Economic Feasibility
- Operational Feasibility
- Ethical and Legal Feasibility
- Schedule Feasibility

7.1 Technical Feasibility

This study is focused on gaining an understanding of the present technical resources of the organization and their applicability to the expected needs of the proposed system. The project is considered technically feasible and the internal technical capability is sufficient to support the project requirements.

Technical feasibility involves whether the necessary technology exists, technical guarantees of accuracy, reliability, ease of access, data security, and aspects of future expansion/modification.

- The system mainly consists of a web application for both clients and service providers. Users
- In this system, users don’t need a lot of technical knowledge regarding the operating system.

- Uses modern technology such as Postgresql, Express.js, React.js, and Node.js) to develop the product. Therefore, the end-users can easily use the system and get a better experience.

- React.js and Material UI are the best, fastest and easiest frameworks capable of creating the UIs creative and responsible for user attraction.

- Postgresql is a Relational Database that is similar to MySQL and has features like storing JSON data in table columns etc. We use PostgreSQL because having a relational database is very important to build a complex application (entrepreneurship portal) since it has high data-intensive tasks (bidding process) and maintains complex relationships.

- Express.js is a web application framework for Node.js. It provides various features that make web application development fast and easy which otherwise takes more time using only Node.js.
- Node.js open source cross-platform JavaScript that helps in the development of real-time network applications. It can also interpret JavaScript code via Google’s V8 JavaScript engine.

- As the system is web-based, it can be backed up with less effort.
- The system is expected to have adequate responses regardless of the number of users.
- It is available within given resource constraints. All these current technical resources are sufficient for the system. It can be easily applied to current problems and can handle this solution. We currently possess the necessary technologies.

In addition to those,

Git

● We use git for version control because we can get previous versions easily through Git.

GitHub

● We use GitHub for team collaboration. In GitHub, we can merge the codes of team members and continue the project easily.

Jest and Puppeteer

● Jest is a testing tool for testing React apps. It is designed with simplicity in mind. It offers a powerful API to build isolated tests, snapshot comparisons.

● Puppeteer is a Node.js library that provides a high-level API.

Trello

● Trello is used for managing and collaborating with team members on work projects and tasks.

7.2 Economical Feasibility

It refers to the benefits or outcomes we are deriving from the product as compared to the total cost we are spending for developing the benefits are more or less the same as the common system then it is not feasible to develop the product.

There are several types of hosting to choose from, and many different variables at play. Therefore, we have selected a free domain. We are using a free host server and free web development tools.

The system is economically feasible. The system provides the following benefits to the customer. the costs and benefits under the **STARTUP DOOR** project are as follows.

Cost

There are many types of costs related to this area, such as cost related to purchasing hardware or software, any training cost (staff training about the new system), or any cost when the new system is implemented to the company or organization, and other operational costs related to maintaining and upgrading the system throughout its life cycle.

Development Cost

These are the costs only once and associated with the development phase of the system.

● Software cost - Uses free and open-source software to develop and implement the system.

● Hardware cost - As the system is web-based, it's unnecessary to purchase the hardware.

● Implementation cost - Doesn't require.

● Domain and Web hosting services - Uses the free domain name and hosting service.

● Payment handling - Use the sandbox payment method in **PayPal / Pay Here**.

Operating Cost

These are the recurring costs throughout the project execution and operation time.

● Maintenance and Upgrading cost - will depend on the software bugs, security issues, software updates.

● Training Cost will not happen because only one person (Admin) operates the system from inside the business.

● Other costs - assume that will not happen. because the STARTUP DOOR is built on using free and open-source software.

Benefits

The benefits of the system that add value to the organization or the users can be tangible or intangible.

Intangible Benefits

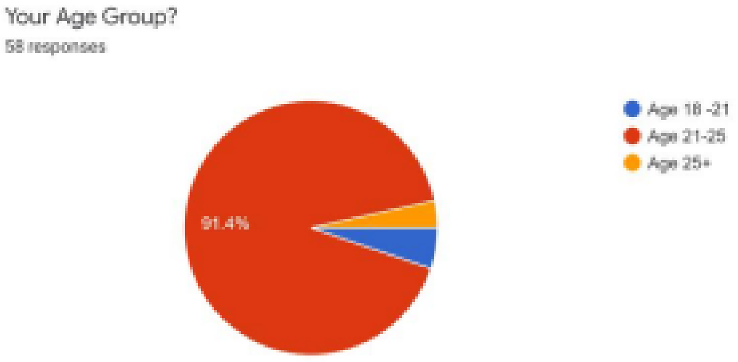
- Reduces the workload.
- Reduces the processing time
- Supports building a new customer base without effort.
- Easiness of starting a small business.
- Save the time that they spent on a business.

Tangible Benefits

- An opportunity to start a business with less investment.
- No need to meet each other to settle the payments.
- Earn extra income from a third-party source without a great commitment.
- Reduce the cost of running the business location.
- Increase the productivity of operations by being physically activity.

7.3 Operational Feasibility

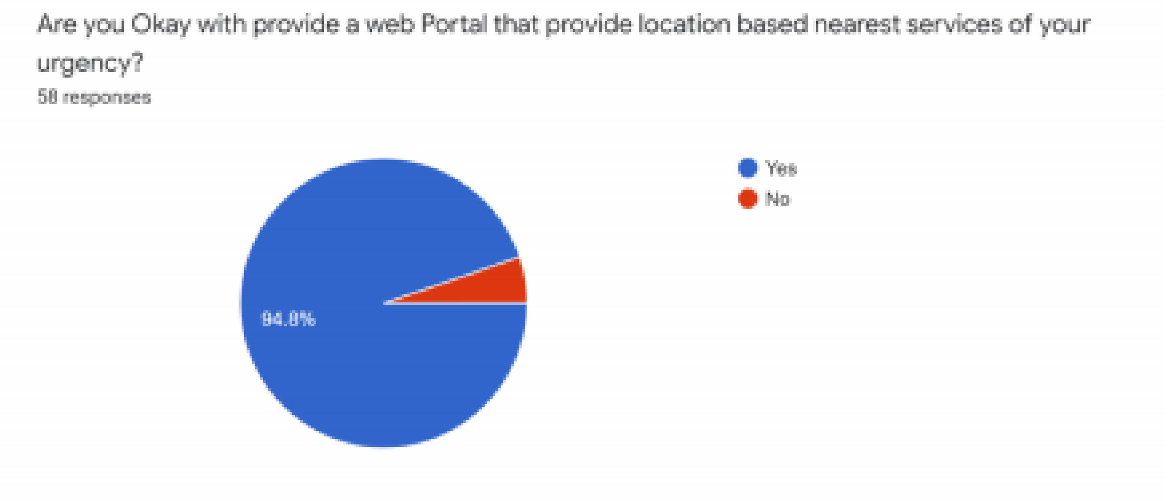
We have surveyed over 60 people, mainly those who live around Colombo. Over 90% of them are between the ages of 21 and 25. We let them know about our idea of our online service system and asked about their opinion on this system.

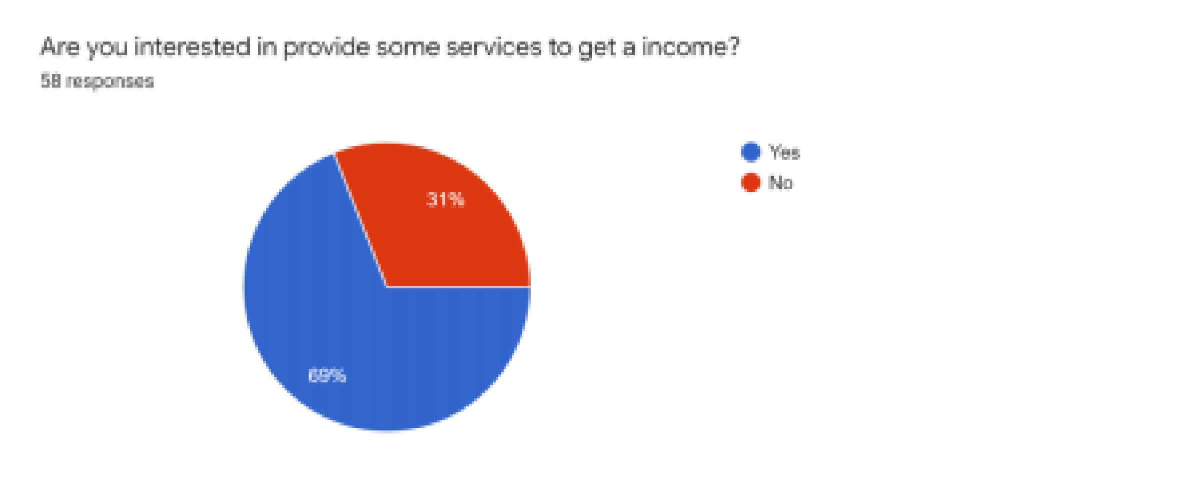


Operational feasibility studies are generally utilized to answer the following questions: Process: How do the end-users feel about a new process that may be implemented?

As this system is web-based, it is very easy to be operated by someone who has an internet connection and a smart device. As this system has an authentication process, the login process will be protected by doubtful logins.

As per our survey, nearly 95% of responses agreed with the portal that
provides location-based services.

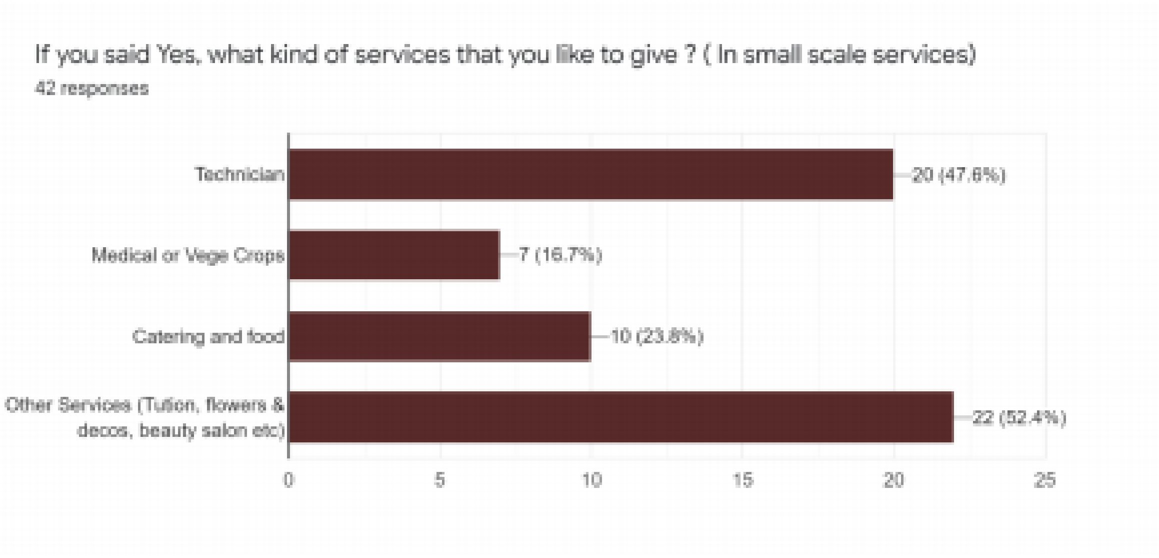


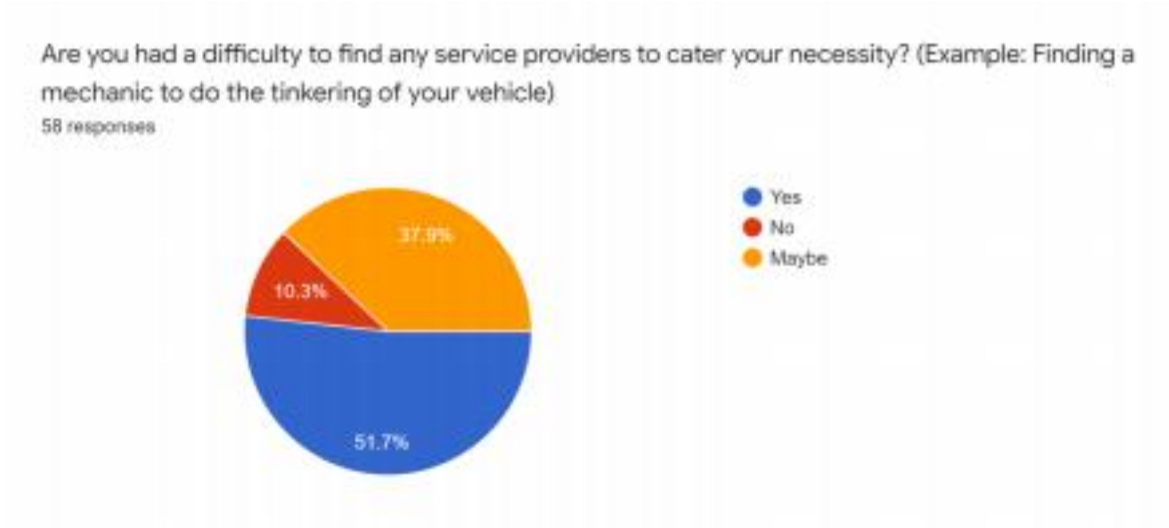


Evaluation – Not only is the process active in the organization but also considered whether it can be functional.

Through our survey, we found over 60% of people are facing problems with finding jobs during this lockdown period. In their area, they need urgent services like technical, food and cuisine, vegetables and medicine plants, and other services as well.

And results in nearly 70% of responses were agreed to provide service to get income. As the startup door launched, being able to aspect this website can create the path to businesses and new ideas.





On-Demand

Implementation – Stakeholder, manager, and end-user tasks.

In the STARTUP DOOR, all the end-users/ stakeholders are required to have a mobile phone, network facility, and valid email to communicate with the system and send notifications to them. When in a business society they also have those facilities. Then it’s very easy to be a part of the community of STARTUP DOOR.

In-House Strategies – How will the work environment be affected? How much will it change? Considering the current situation of society there isn’t any place or system to motivate their startups without any great commitment. So, that’s why the startup door addresses and motivates the people who want to gain more money/ who are willing to earn more money, and who are willing to start their own business as small as possible.

7.4 Ethical and Legal Feasibility

The STARTUP DOOR does not break any rules and regulations under the laws in our country. As responsible web developers, responsible to our customers to inform them of practices that are questionable or illegal with their projects

When the customer pays money for ordered things, the customer can pay, using online cards (credit/master/visa) and cash on delivery. While using the online cards all the details of the cards and users are highly protected.

This system includes GDPR (General Data Protection Regulation) & CCPA (California Consumer Privacy Act). GDPR means looking at how a website records visitor information, and what it does with it. This is European Union and our website does not comply with their data laws, we could face prosecution. CCPA is designed to protect the data of customers interacting with business in California in much the same way as the European legislation. CCPA specifically focuses on companies of a

certain size or business that makes money from selling customer details. So, we can concern those laws and regulations and therefore we can safely assume that this is a legally and ethically feasible project.

7.5 Schedule Feasibility

During the lack of time or the time become mandatory, we must finish the project within a given time

★ Number of work hours as a team per week = 4

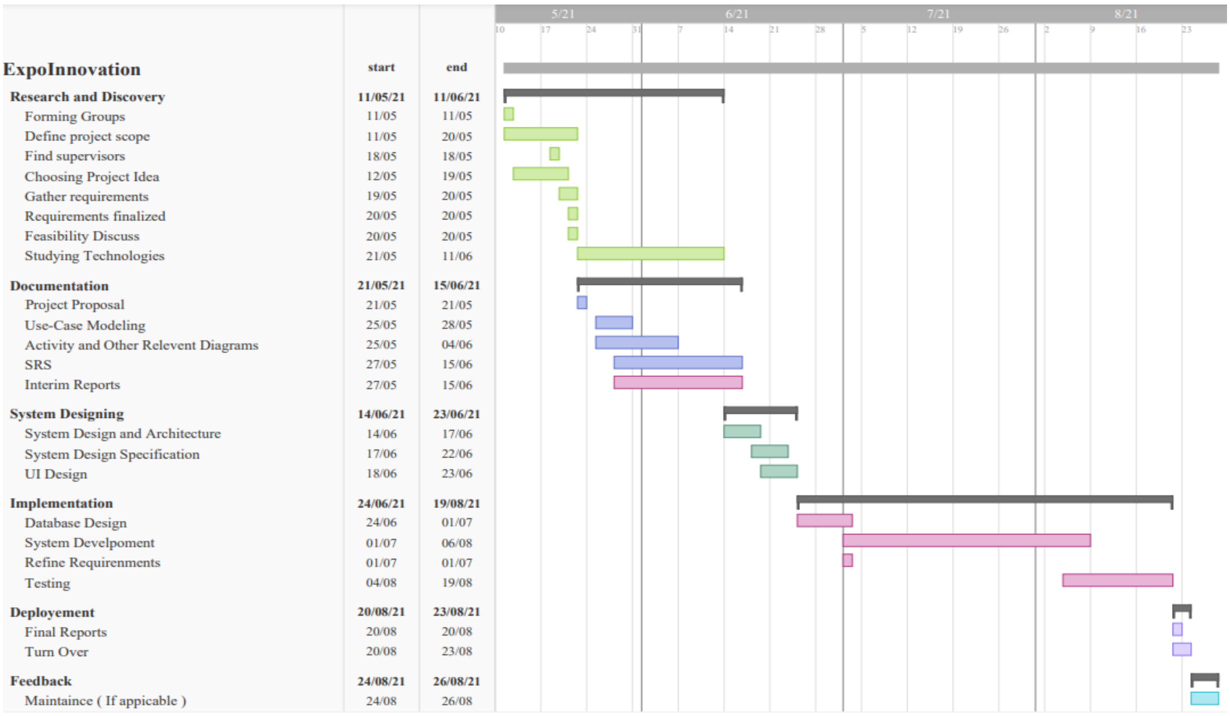
★ Number of individual work hours per week = 6

★ Man-hours per week = (4 + 6)*5 = 50

★ Estimated number of weeks = 14

★ Estimated total of man-hours = 50*14= 700

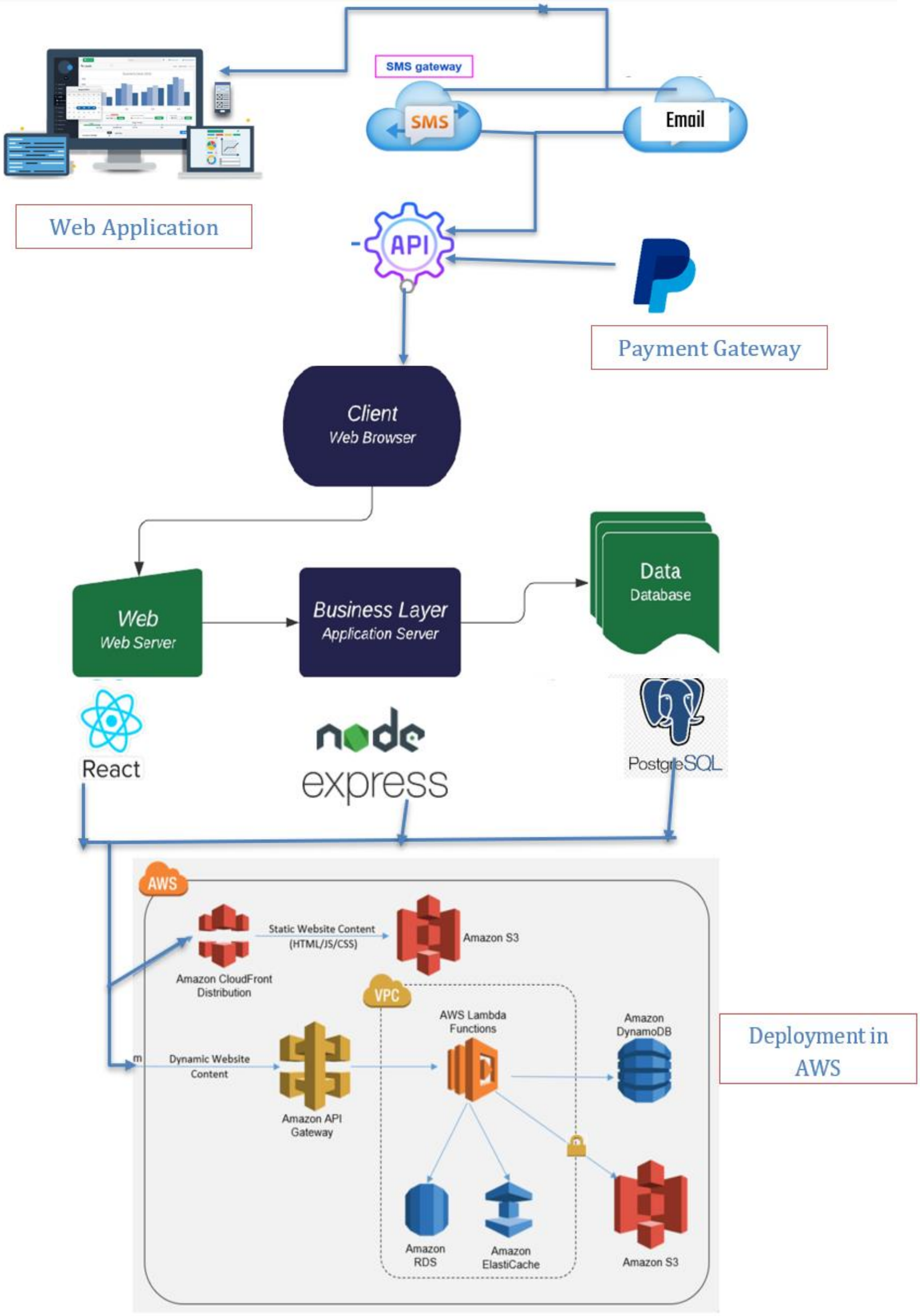
Since all of the required resources are readily available and with the study we conducted we can conclude that our project is feasible.



ExpoInnovation	start	end
Research and Discovery	**11/05/21**	**11/06/21**
Forming Groups	11/05	11/05
Define project scope	11/05	20/05
Find supervisors	18/05	18/05
Choosing Project Idea	12/05	19/05
Gather requirements	19/05	20/05
Requirements finalized	20/05	20/05
Feasibility Discuss	20/05	20/05
Studying Technologies	21/05	11/06
Documentation	**21/05/21**	**15/06/21**
Project Proposal	21/05	21/05
Use-Case Modeling	25/05	28/05
Activity and Other Relevent Diagrams	25/05	04/06
SRS	27/05	15/06
Interim Reports	27/05	15/06
System Designing	**14/06/21**	**23/06/21**
System Design and Architecture	14/06	17/06
System Design Specification	17/06	22/06
UI Design	18/06	23/06
Implementation	**24/06/21**	**19/08/21**
Database Design	24/06	01/07
System Develpoment	01/07	06/08
Refine Requirenments	01/07	01/07
Testing	04/08	19/08
Deployement	**20/08/21**	**23/08/21**
Final Reports	20/08	20/08
Turn Over	20/08	23/08
Feedback	**24/08/21**	**26/08/21**
Maintaince (If appicable)	24/08	26/08

7. Systems Architecture

High-Level Architecture

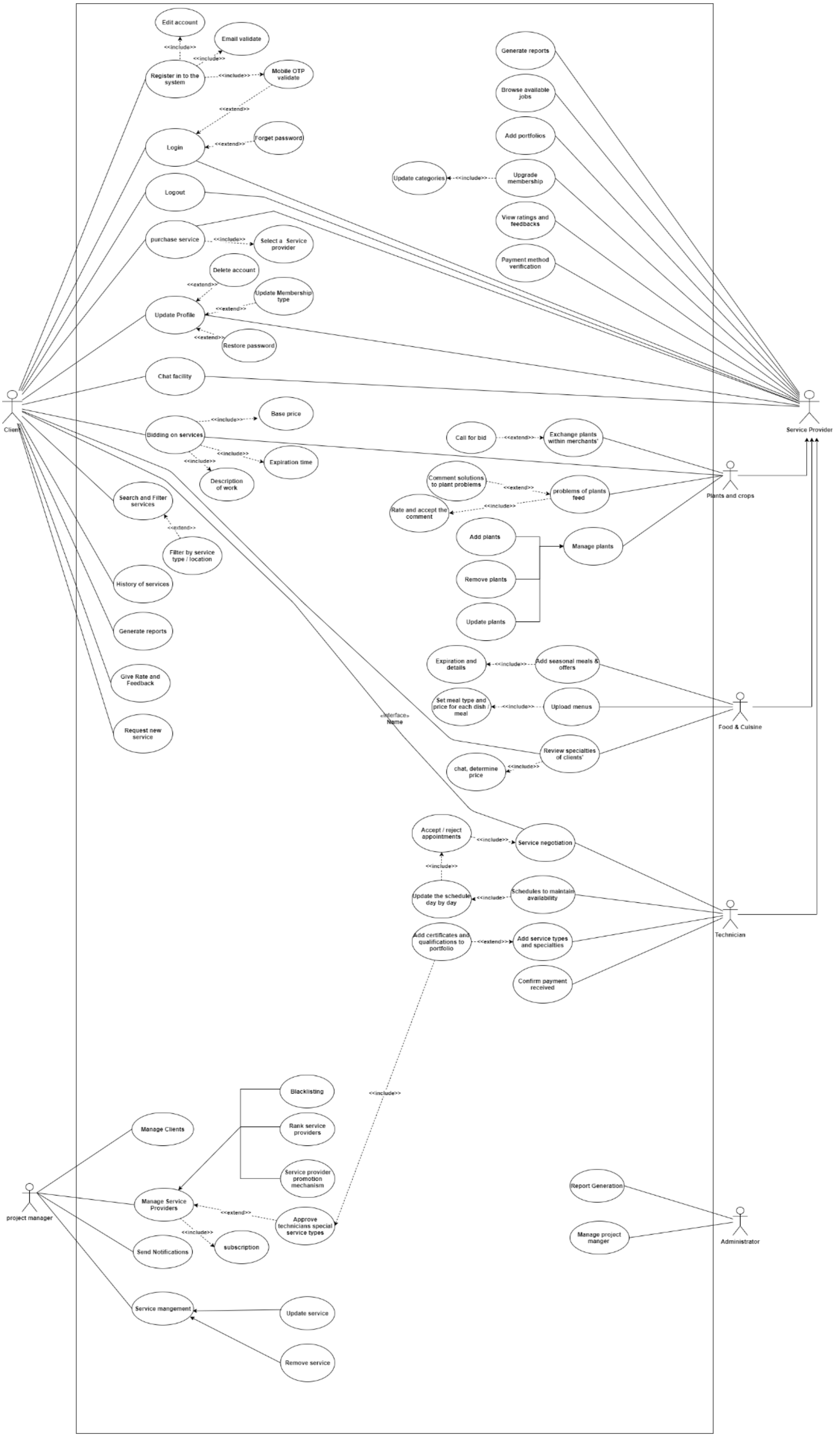

SMS gateway
SMS
Email
Web Application
API
Payment Gateway
Client
Web Browser
Web
Web Server
Business Layer
Application Server
Data
Database
React
node
express
PostgreSQL
AWS
Amazon CloudFront
Distribution
Static Website Content
(HTML/JS/CSS)
Amazon S3
VPC
AWS Lambda
Functions
Amazon
DynamoDB
Dynamic Website
Content
Amazon API
Gateway
Amazon
RDS
Amazon
ElastiCache
Amazon S3
Deployment in
AWS

8. Requirements Specification

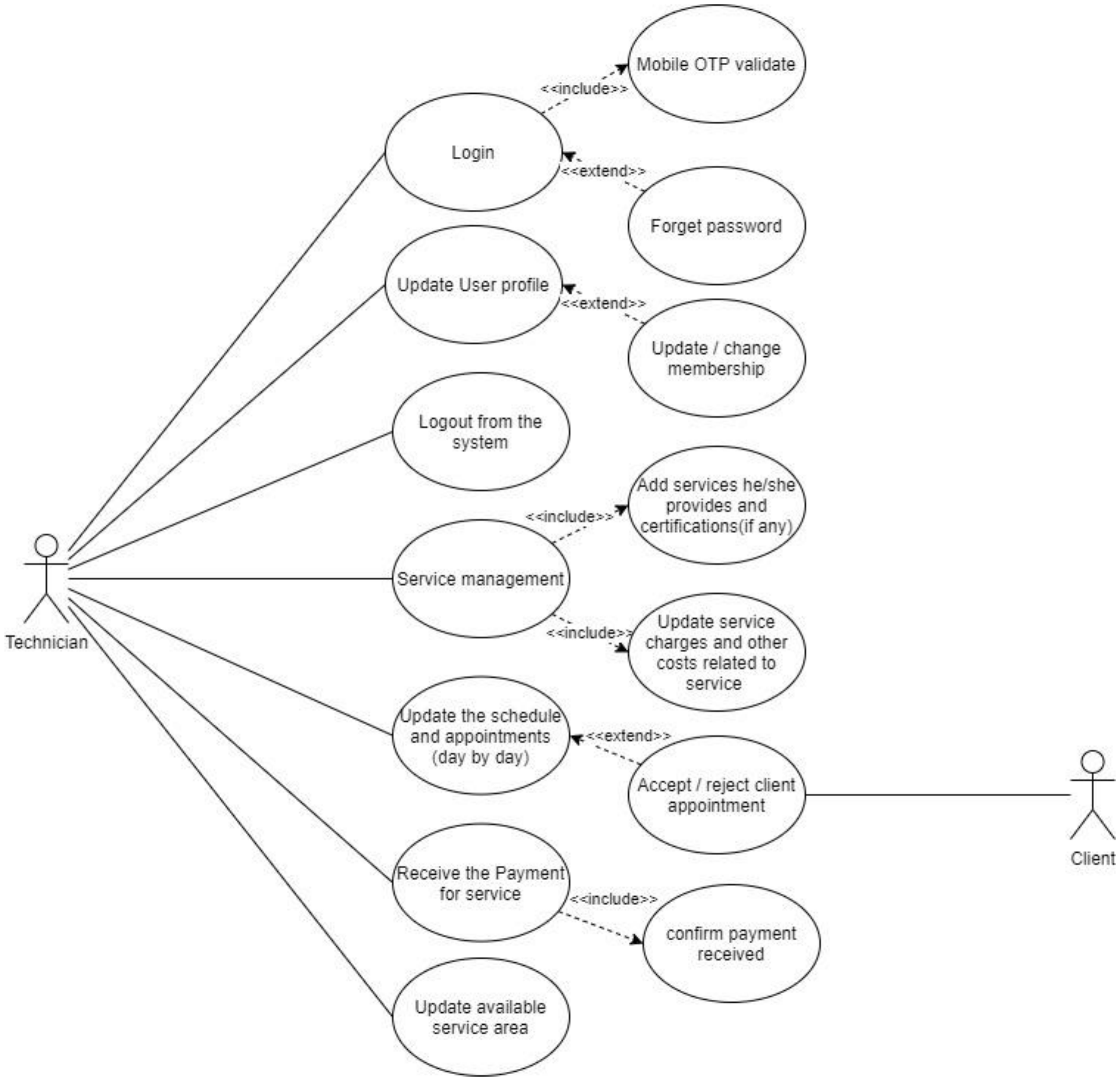
A brief introduction to requirement specification and diagrams (Use-case diagram, Class diagram, Component diagram, etc) depicting your systems requirements.

8.1 Use-case diagram

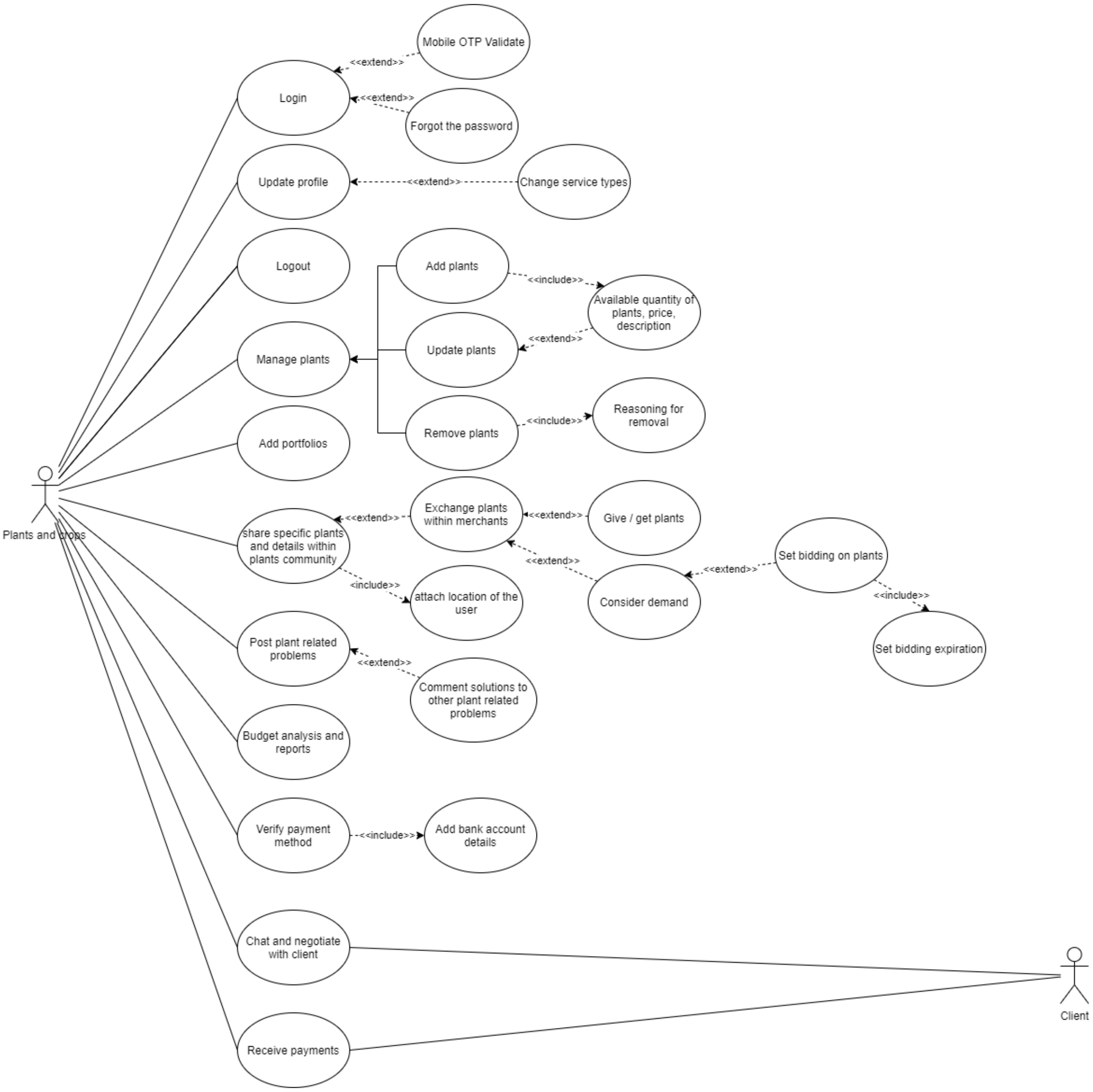
Link: https://app.diagrams.net/#G1xbMr5FhTcKrJWGxedcG3D3rda9eXhNxM


Edit account
Email validate
Register in to the system
Mobile OTP validate
<<include>>
<<include>>
<<include>>
<<extend>>
Login
Forget password
<<extend>>
Logout
purchase service
<<include>>
Select a Service provider
Delete account
<<extend>>
Update Membership type
<<extend>>
Update Profile
<<extend>>
Restore password
Chat facility
Bidding on services
<<include>>
Base price
<<include>>
Expiration time
<<include>>
Description of work
Search and Filter services
<<extend>>
Filter by service type / location
History of services
Generate reports
Give Rate and Feedback
Request new service
Client
Generate reports
Browse available jobs
Add portfolios
Update categories
<<include>>
Upgrade membership
View ratings and feedbacks
Payment method verification
Service Provider
Call for bid
<<extend>>
Exchange plants within merchants'
Comment solutions to plant problems
<<extend>>
problems of plants feed
<<include>>
Rate and accept the comment
Add plants
Remove plants
Update plants
Manage plants
Plants and crops
Expiration and details
<<include>>
Add seasonal meals & offers
Set meal type and price for each dish / meal
<<include>>
Upload menus
Food & Cuisine
«interface»
Name
Review specialties of clients'
<<include>>
chat, determine price
Accept / reject appointments
<<include>>
Service negotiation
<<include>>
Update the schedule day by day
<<include>>
Schedules to maintain availability
Add certificates and qualifications to portfolio
<<extend>>
Add service types and specialties
Confirm payment received
Technician
<<include>>
Blacklisting
Rank service providers
Service provider promotion mechanism
Manage Clients
project manager
Manage Service Providers
<<extend>>
Approve technicians special service types
<<include>>
subscription
Send Notifications
Service mangement
Update service
Remove service
Report Generation
Manage project manger
Administrator

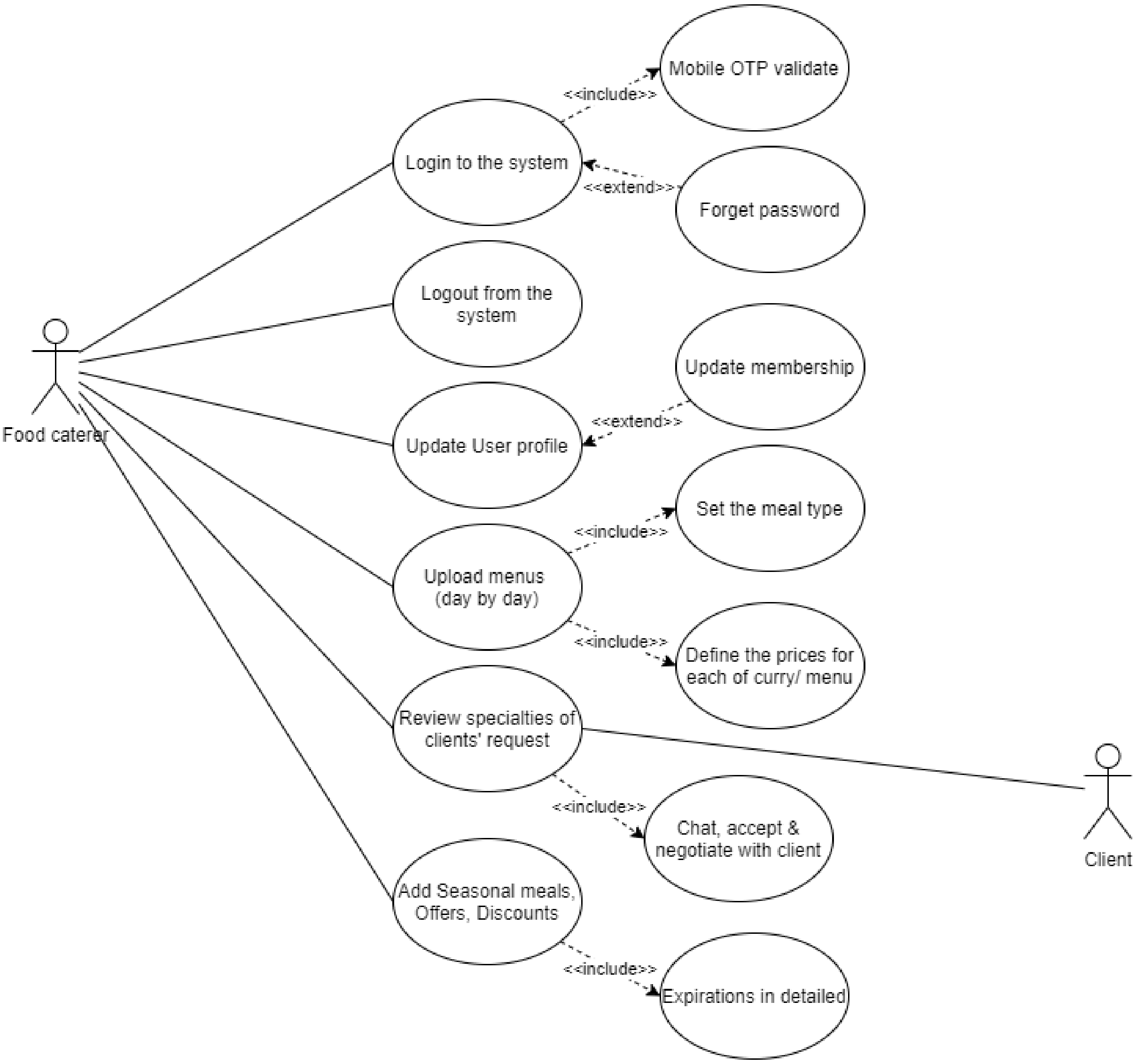
Technician use case


Mobile OTP validate
<<include>>
Login
<<extend>>
Forget password
Update User profile
<<extend>>
Update / change membership
Logout from the system
Add services he/she provides and certifications(if any)
<<include>>
Technician
Service management
<<include>>
Update service charges and other costs related to service
Update the schedule and appointments (day by day)
<<extend>>
Accept / reject client appointment
Client
Receive the Payment for service
<<include>>
confirm payment received
Update available service area

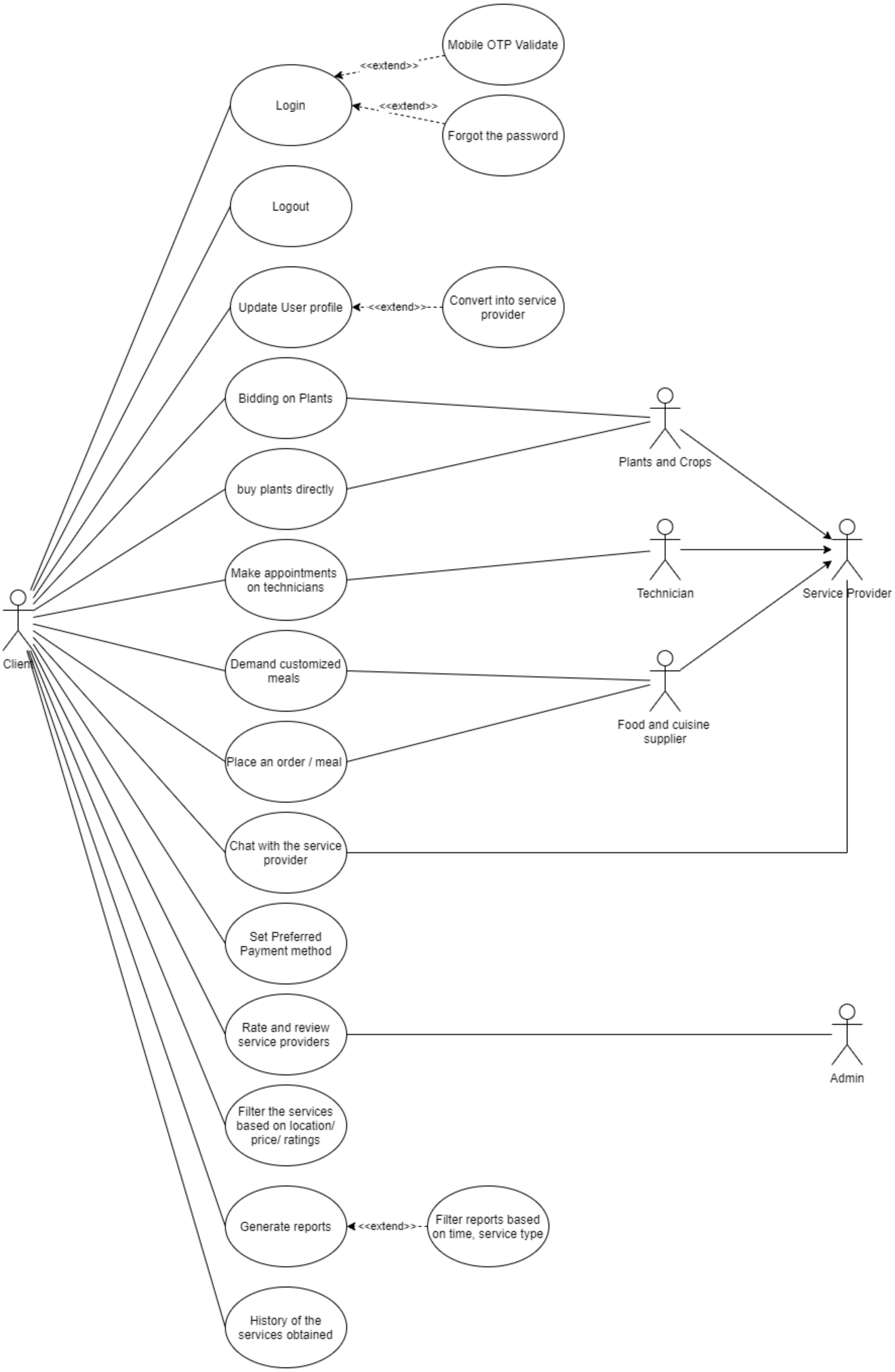
Plants and crops use case



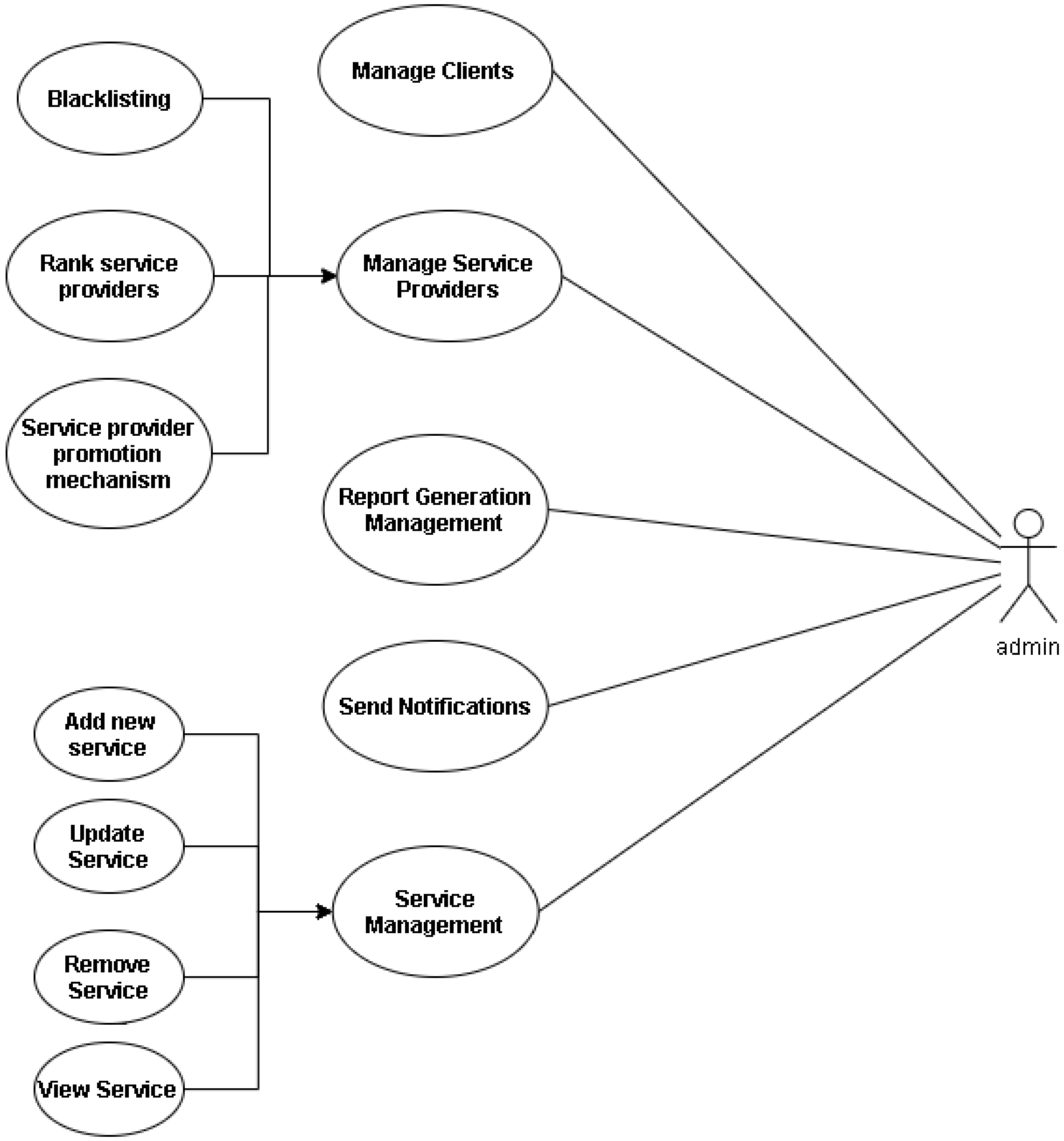
Food and cuisine use case



Client use case


Mobile OTP Validate
<<extend>>
Login
<<extend>>
Forgot the password
Logout
Update User profile
<<extend>>
Convert into service provider
Bidding on Plants
Plants and Crops
buy plants directly
Technician
Service Provider
Make appointments on technicians
Client
Demand customized meals
Food and cuisine supplier
Place an order / meal
Chat with the service provider
Set Preferred Payment method
Rate and review service providers
Admin
Filter the services based on location/ price/ ratings
Generate reports
<<extend>>
Filter reports based on time, service type
History of the services obtained

Admin use case



8.2Use case narratives

Narratives for all the users

Name of the use case :	Register in the system
Description :	The user who visits our website can be a registered customer or trader in our system.
Primary actor :	All the visiting users
Secondary actors :	-
Pre-conditions :	● Users must have a clear internet connection to operate on the website. ● Users have to enter our system landing page
Main flow	1. The user enters the website. 2. Click on the register button. 3. Provide necessary details in the registration form. 4. Accept the terms and conditions of the system. 5. Click on the register button to accept the details.
Postcondition :	Users will be registered in the system and have the ability to operate as a client and service provider as selected in the registration form.
Alternative Flows :	-

Name of the use case :	Log into the system
Description :	Registered users log into their own account in the system
Primary actor :	Registered users
Secondary actors :	System
Pre-conditions :	● User must have registered in the system before. ● Clear network infrastructure to operate in the system.
Main flow	1. User step into the system 2. Click on the login button 3. Give login credentials and accept. 4. System checks the last login date and time to the system and validates the right user by OTP code verifications. 5. Users give the OTP code verification. 6. Log into the system.
Postcondition :	Users can operate in the system and users have logged in. Last login will be updated in the system.
Alternative Flows :	-

Name of the use case :	Updating User Profile
Description :	After the authentication happens, The user's can go to his profile to edit their profile credentials or account picture or password.
Primary actor :	All the visiting users
Secondary actors :	-None
Pre-conditions :	● Users must have a Registered account ● User have to enter to our system landing page
Main flow	1. Log into the system. 2. In the Navigation panel, select update user account. 3. Update the details. 4. Save the details
Post condition :	Users will be registered in the system and have the ability to operate as a client and service provider as selected in the registration form.
Alternative Flows :	Select the provided icon from the dashboard.

Name of the use case :	Log out from the system
Description :	Logged in user can log out from the system
Primary actor :	Logged User
Secondary actors :	-
Pre-conditions :	● User must be logged into the system
Main flow	1. Click on the avatar icon in the top right corner 2. Select the logout option. 3. Confirm logout.
Postcondition :	Logged-in user will be logged out from the system.
Alternative Flows :	-

Narratives for Client

Name of the use case :	Bid on services
Description :	Client can bid on the services
Primary actor :	Client
Secondary actors :	Service provider
Pre-conditions :	● Client have to be registered in the system ● Logged in to the system
Main flow	1. Log into the system 2. Navigate to the services section. 3. Select a service
Post condition :	
Alternative Flows :	

Name of the use case :	Search and filter services and types
Description :	Client can search a particular service provider or service category
Primary actor :	Client
Secondary actors :	System
Pre-conditions :	● Registered and logged into the system
Main flow	1. Click on the search bar 2. Select a targeted tag (Service or service provider) 3. Type in the search bar 4. System will show the results based on the query. 5. Select the targeted result.
Postcondition :	Client can find results on a service category or service provider.
Alternative Flows :	-

Name of the use case :	Give rate and feedback on services
Description :	Client have to rate the service provider and give some reviews on the service.
Primary actor :	Client
Secondary actors :	System
Pre-conditions :	● Client have to obtain a service from a service provider. ● Settle the payments on service.
Main flow	1. Make a service request. 2. Obtain the service from a provider.

	3. Settle the payments. 4. Rate the provider and give some comments on service. 5. Finish the job.
Post condition :	Client gave some feedback based on the service they obtained.
Alternative Flows :	-

Name of the use case :	Maintain history of services obtained
Description :	System will maintain a history of services obtained by the client.
Primary actor :	System
Secondary actors :	Client
Pre-conditions :	● Client have to obtain some services on the system
Main flow	1. Client obtain a service from a service provider 2. Finish the job and rate the service provider 3. System will store the obtained services with a complete description.
Post condition :	● Clients can find service providers by the history. ● Client can refer to the history of the services obtained.
Alternative Flows :	-

Name of the use case :	Request a new service category
Description :	Client can request for a new service category from the administrator.
Primary actor :	Client
Secondary actors :	System
Pre-conditions :	● Client have to logged into the system
Main flow	1. Log into the system. 2. Navigate to the request services section from the dashboard. 3. Add the details of the service and category. 4. Click on the submit button. 5. System will store the request in the database. 6. Send directly to the administrator by the system.
Post condition :	Client request for a new service category.
Alternative Flows :	-

Narratives for service providers

Name of the use case :	Generate reports
Description :	Service provider generate the reports on their preferences.
Primary actor :	Service provider
Secondary actors :	System
Pre-conditions :	● User must log into the system
Main flow	1. User log into the system. 2. Navigate to generate reports section. 3. Select the report type you want to generate. . 4. Use filters to get more descriptive reports. 5. System will analyse the data and create into a report.
Post condition :	Service provider gets the reports based on their preference.
Alternative Flows :	-

Name of the use case :	Browse for available jobs
Description :	Service providers can search the range of services available in the system.
Primary actor :	Service provider
Secondary actors :	-
Pre-conditions :	● Provider have to be logged into the system
Main flow	1. Log into the system. 2. Navigate to the services section. 3. Search for the service type in the search bar or scroll the page. 4. Refer the section.
Post condition :	Service providers get a clear idea of the service types available in the system.
Alternative Flows :	Maintain the service types by the update profile section

Name of the use case :	Upgrade the membership
Description :	Service providers have to be a member of a particular service category to serve the customers. So in case a provider can expand the service types that serve and upgrade the membership.
Primary actor :	Service provider
Secondary actors :	System
Pre-conditions :	● Providers have to be logged into the system. ● Serve to the customers in the free membership time period.
Main flow	1. Service provider supply services to the customers in a free trial period.

	2. System will maintain a history of the services he/ she provided to the customers. 3. System will calculate the upgrading value of the membership according to services provided by the user. 4. After completion of the free trial period system will block the user profile temporarily and tend to move with a membership. 5. Service provider pay the membership fee and renew the account. 6. If the payment is not settled, the user account will be banned.
Post condition :	User will become a permanent service provider of the system.
Alternative Flows :	If the service provider is registered, ● Navigate to the update profile section and upgrade the membership. If the user is already a member, ● System will pop up the alerts on expiration of the membership. ● Click on the message and renew the membership or navigate to the update user profile section and update the value.

Name of the use case :	View ratings and feedbacks
Description :	Service provider can view the comments and ratings from the clients' that served.
Primary actor :	Service provider
Secondary actors :	System
Pre-conditions :	● Log into the system as a service provider.
Main flow	1. Log into the system 2. Navigate to the review comments section. 3. View the comments and ratings in detail.
Post condition :	Service providers get the will of the client on provided service.
Alternative Flows :	-

Name of the use case :	Add portfolios
Description :	Service provider can add the projects or jobs they have done before.
Primary actor :	Service provider
Secondary actors :	System
Pre-conditions :	● User must log into the system ● User must be registered as a service provider.
Main flow	1. Log into the system. 2. Navigate to add portfolio section from the dashboard. 3. Provide the details that are required. 4. Provide some images if you have. 5. Upload the portfolio.
Post condition :	User can maintain a portfolio to show previously done jobs to clients.

Alternative Flows :	User can add portfolios from the update profile section.

Name of the use case :	Payment method verification
Description :	Service provider must verify the payment method to get paid in online payments.
Primary actor :	Service provider
Secondary actors :	System
Pre-conditions :	● User must be registered as a service provider.
Main flow	1. Log into the system 2. Navigate to verify payments sections. 3. Give respective details on the form. 4. Submit the form. 5. System will update the details in the database.
Post condition :	Users can receive online payments through the system.
Alternative Flows :	-

Narratives for Plants and crops Service provider

Name of the use case :	Exchange plants within plants merchants
Description :	Users can exchange any variety of plants within plants community
Primary actor :	Plants provider
Secondary actors :	Plants provider
Pre-conditions :	● Users must be logged in and registered as plants and crops providers.
Main flow	1. Log into the system. 2. Navigate to the plant's feed. 3. Add new plant details and submit a. The system will generate it as a feed and display it in the plants feed. 4. Demand plants from other plant providers. a. Users can chat with the plant's provider also.
Post condition :	User demand on plants or user provide and exchange within the community.
Alternative Flows :	-

Name of the use case :	Problems on plants feed
Description :	User can add problems related to the plants as a post, and others can add solutions as comments.
Primary actor :	Plants provider
Secondary actors :	Plants provider

Pre-conditions :	● Users must register as plant providers.
Main flow	1. Log into the system. 2. Navigate to Problems feed. 3. Add a problem of plants. a. Add images to get a more descriptive manner. b. Add some description and submit. 4. Comment on the problems in the feed. a. Select the feed post. b. Click on the comment button. c. Type the comments. d. Select the comment button.
Post condition :	Users can communicate the problems of plants' within community.
Alternative Flows :	-

Name of the use case :	Manage Plants
Description :	User can manage the stock of available plants in the system.
Primary actor :	Plants provider
Secondary actors :	System
Pre-conditions :	● Users must be registered as plant providers.
Main flow	1. Log into the system. 2. Navigate to manage plants section 3. Add plants to the system a. Describe the type and specialties of the plants b. Price of the plant c. Quantity of available. 4. Update plant details a. Add more description. b. Update the quantity of plants available. c. Update the price of plants. 5. Remove a plant a. Reasoning for removal 6. Save the data 7. System will update the data in the database.
Post condition :	User can manage the plants quantity and prices in the system.
Alternative Flows :	-

Narratives for Food and cuisine provider

Name of the use case :	Upload menus
Description :	Providers can upload the menus which are available to serve.
Primary actor :	Food and cuisine provider
Secondary actors :	System

Pre-conditions :	● User must be registered as a food and cuisine provider.
Main flow	1. Log into the system 2. Navigate to menus section. 3. Click on the upload menu icon. a. Upload photos. b. Price of meal. c. Available quantity. d. Location 4. Click on uploaded menu a. Remove the menu. b. Update price. c. Available quantity. d. Location of the service. 5. Submit the details. 6. Database will be updated and creates a feed to newly updated menus by the system.
Post condition :	User can handle the available menus and remove menus as well.
Alternative Flows :	-

Name of the use case :	Add seasonal meals and offers
Description :	Providers can add some seasonal offers and meals during seasons.
Primary actor :	Food and cuisine provider
Secondary actors :	System
Pre-conditions :	● User must be registered as a food caterer.
Main flow	1. Log into the system. 2. If it is a seasonal meal a. Proceed as an add menu. b. Mark as a seasonal meal. 3. If it is a seasonal offer a. Click on the add offer button. b. Add necessary details. c. Add expiration. d. Click on submit. 4. Save the data in the database. 5. System will create a feed post and add it into the client's feed.
Post condition :	Food caterers add seasonal meals and offers.
Alternative Flows :	-

Name of the use case :	Custom orders
Description :	Food caterers can deal with direct customer meals.
Primary actor :	Client
Secondary actors :	Food caterer

Pre-conditions :	● Food caterers must log into the system. ● Client must request a meal.
Main flow	1. Log into the system 2. View the requests of the meals on the landing page. 3. If want to negotiate, chat with the client. 4. Confirm or reject the order.
Post condition :	Food caterers can deal with direct customer orders.
Alternative Flows :	-

Narratives for Technician

Name of the use case :	Schedules the work
Description :	Technicians can schedule the work to let know availability of the service to the clients.
Primary actor :	Technicians
Secondary actors :	System
Pre-conditions :	● User have to register as a technician.
Main flow	1. Log into the system. 2. Navigate to schedule work section 3. Add the details of the work 4. Add time slot and date. 5. Save the details. 6. System will update the calendar with respect to the time slot and date.
Post condition :	Technicians can schedule their work in the system.
Alternative Flows :	-

Name of the use case :	Add services types and specialties
Description :	Technicians can expand their job specialties.
Primary actor :	Technician
Secondary actors :	System
Pre-conditions :	● User must log into the system. ● User must be registered as a technician.
Main flow	1. Log into the system. 2. Navigate to the “add new services” section. 3. Type necessary details. 4. Add photos, if available. 5. Submit the document.
Post condition :	Technician expands his work range.

Alternative Flows :	-

Name of the use case :	Confirm payment received
Description :	After completion of the service technician is required to supply the details regarding the payment of the service
Primary actor :	Technician
Secondary actors :	-
Pre-conditions :	● Technician must log into the system ● Technician have to work with a job.
Main flow	1. Mark the job as completed. 2. Type the service charge gained from the client. 3. Mark as payment received. 4. Finish the job.
Post condition :	Job will be mark as done and with received payments.
Alternative Flows :	-

Name of the use case :	Service negotiation
Description :	Technician can negotiate with the client about the service.
Primary actor :	Technician
Secondary actors :	Client
Pre-conditions :	● Service provider have to be registers as technician
Main flow	1. Client make an appointment for service. 2. Send details to the technician about the service and appointment details. 3. Service providers use chat to negotiate service agreements, etc.
Post condition :	Technician and client agree to the service agreement and make it an actual service.
Alternative Flows :	-

Narratives for Project manager

Name of the use case :	Manage Clients
Description :	Project managers can manage the clients by referring the feedbacks from the service providers.
Primary actor :	Project manager
Secondary actors :	-

Pre-conditions :	● Logged in as project manager
Main flow	1. Log into the system 2. Navigate to Manage clients. 3. View the feedback by filtering the lowest ratings. 4. View the feedback given by service provider 5. Send notification upto 3 lowest ratings and feedback. 6. Ban or temporarily remove access to the account until pay the duties.
Post condition :	Clients can't deal as their preferences. They will be under control by project manager.
Alternative Flows :	-

Name of the use case :	Manage service providers
Description :	Project manager can take action based on the client's feedback and ratings on particular service provider.
Primary actor :	Project manager
Secondary actors :	-
Pre-conditions :	● Logged in as project manager.
Main flow	1. Log into the system. 2. Click on the service providers review on dashboard 3. View the feedback by filtering the lowest ratings. 4. View the feedback given by client. 5. Send notification upto 3 lowest ratings and feedback. 6. Ban or temporarily remove access to the account until pay the duties.
Post condition :	Service providers also under controlled by the project manager
Alternative Flows :	-

Name of the use case :	Send notifications
Description :	Project manager can send notification to service providers and clients both.
Primary actor :	Project manager
Secondary actors :	-
Pre-conditions :	● Logged in as project manager ● Check the detailed feedbacks given by clients and service providers.
Main flow	1. Log into the system. 2. Check the negative and positive feedback from clients/ service providers. 3. If negative a. Send notification on the negative feedback and inform them "You are under control.". 4. If positive a. Send notification on awarding the successful badges in the system.
Post condition :	Project manager sends notifications on their service through the system.

Alternative Flows :	-

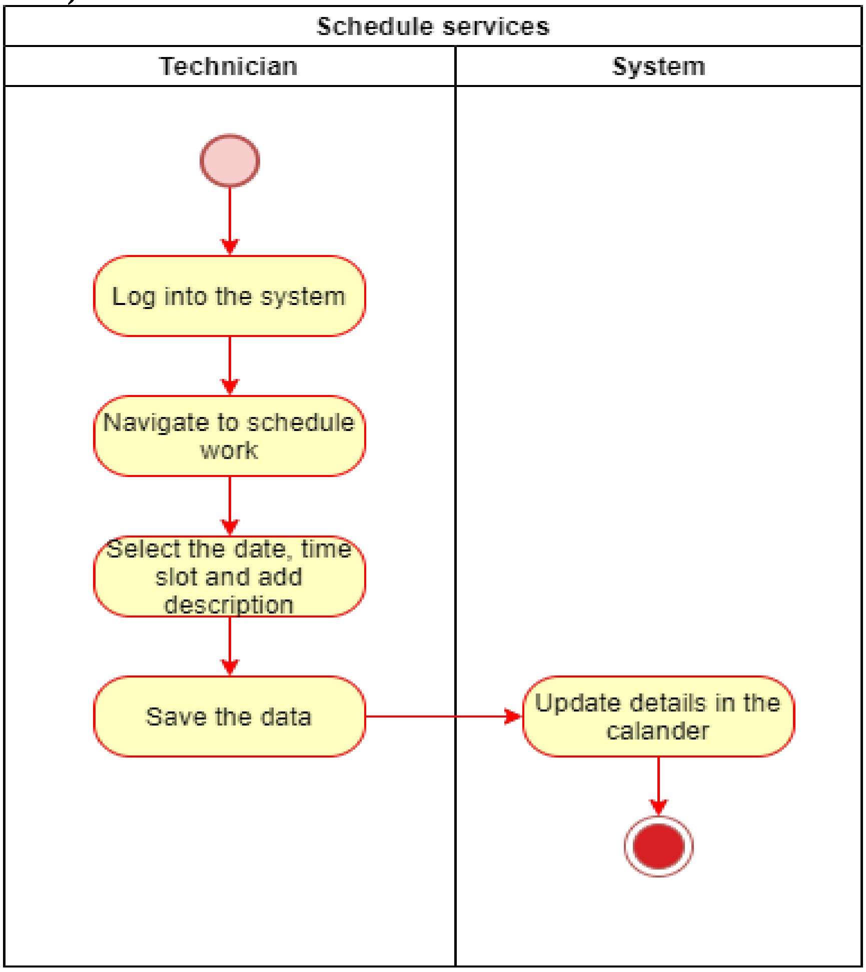
Name of the use case :	Service management
Description :	Project managers can add, remove and update service categories
Primary actor :	Project manager
Secondary actors :	System
Pre-conditions :	● Logged in as project manager
Main flow	1. Log into the system. 2. Navigate to the service management section 3. If want to add a service a. Select the add service category 4. If want to remove a service a. Select remove service category 5. If want to update a. Select update service category 6. Submit the details 7. System will update the services.
Post condition :	Project manager can maintain the services in this section.
Alternative Flows :	-

Narratives for admin

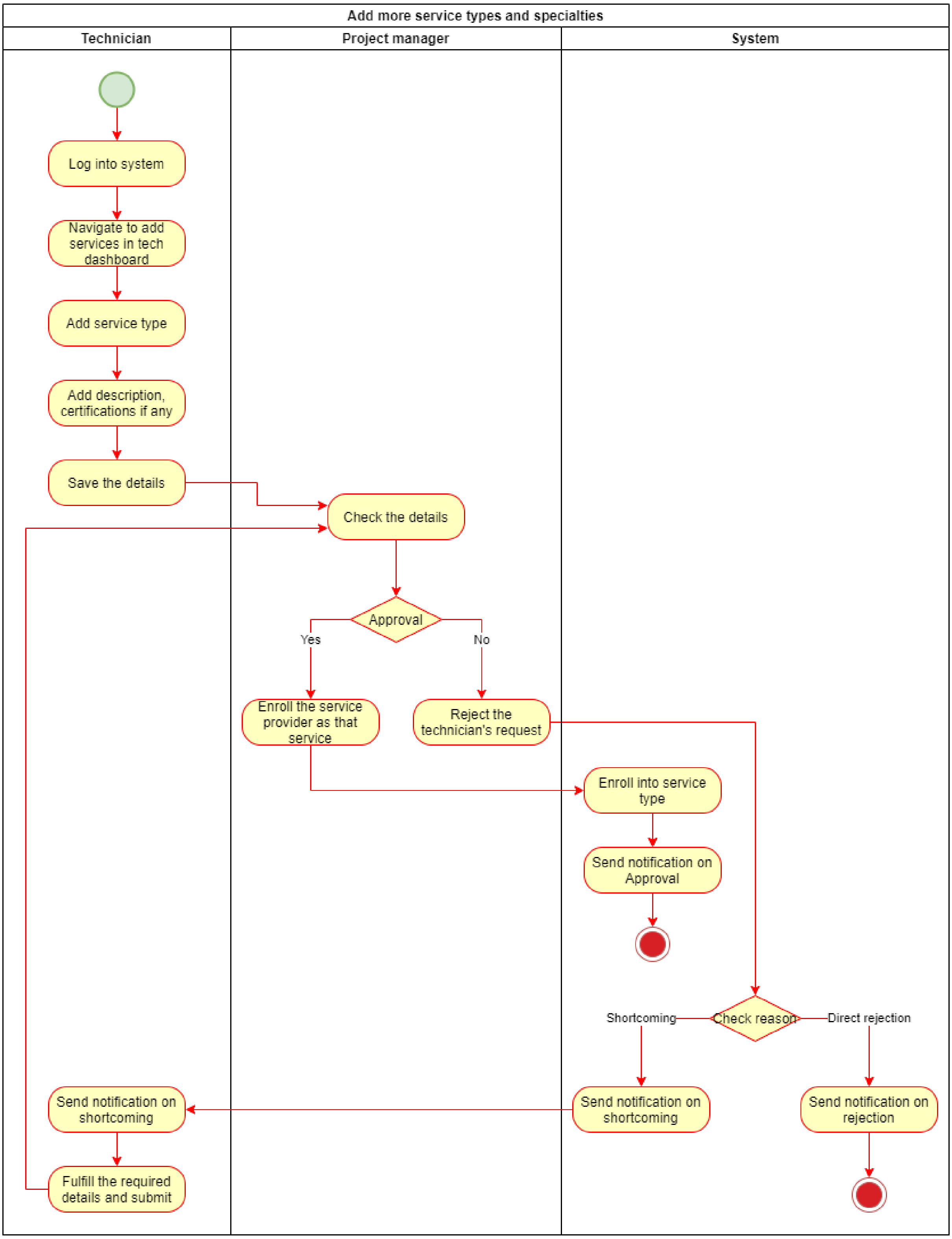
Name of the use case :	Report generation
Description :	Admin can generate the reports on his preference.
Primary actor :	Admin
Secondary actors :	System
Pre-conditions :	● Admin must logged in
Main flow	1. Log into the system 2. Navigate to the reports section. 3. Filter the reports by using filter option 4. Click the submit button to create reports.
Post condition :	Admin can generate the reports
Alternative Flows :	-

8.3 Activity diagrams

1) Technician

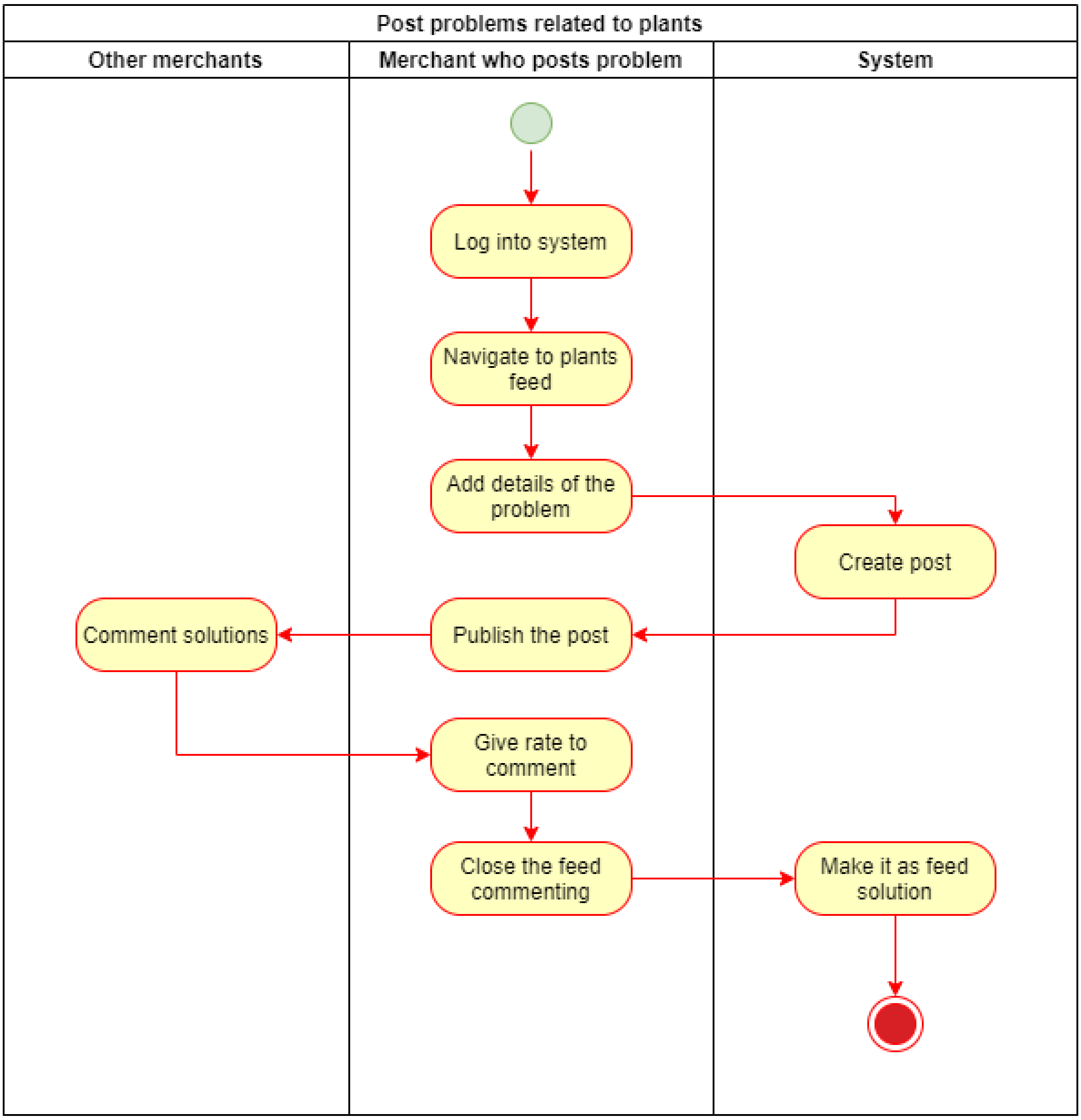

Schedule services
Technician
System
Log into the system
Navigate to schedule work
Select the date, time slot and add description
Save the data
Update details in the calander

8.3.1 Activity diagram for Schedule Services

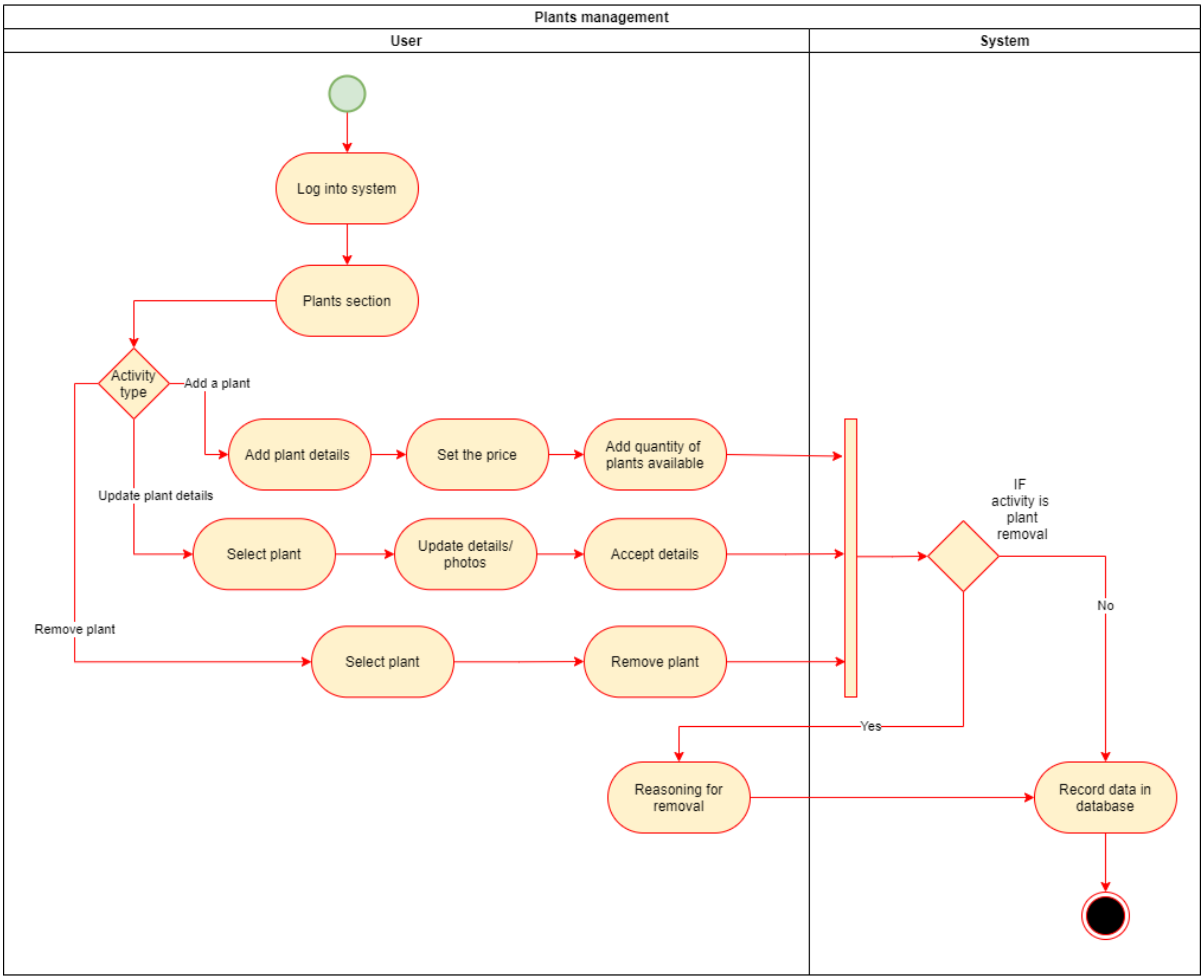

Add more service types and specialties
Technician
Project manager
System
Log into system
Navigate to add services in tech dashboard
Add service type
Add description, certifications if any
Save the details
Check the details
Approval
Yes
No
Enroll the service provider as that service
Reject the technician's request
Enroll into service type
Send notification on Approval
Check reason
Shortcoming
Direct rejection
Send notification on shortcoming
Send notification on rejection
Send notification on shortcoming
Fulfill the required details and submit

8.3.2 Activity diagram for add services types & Specialities

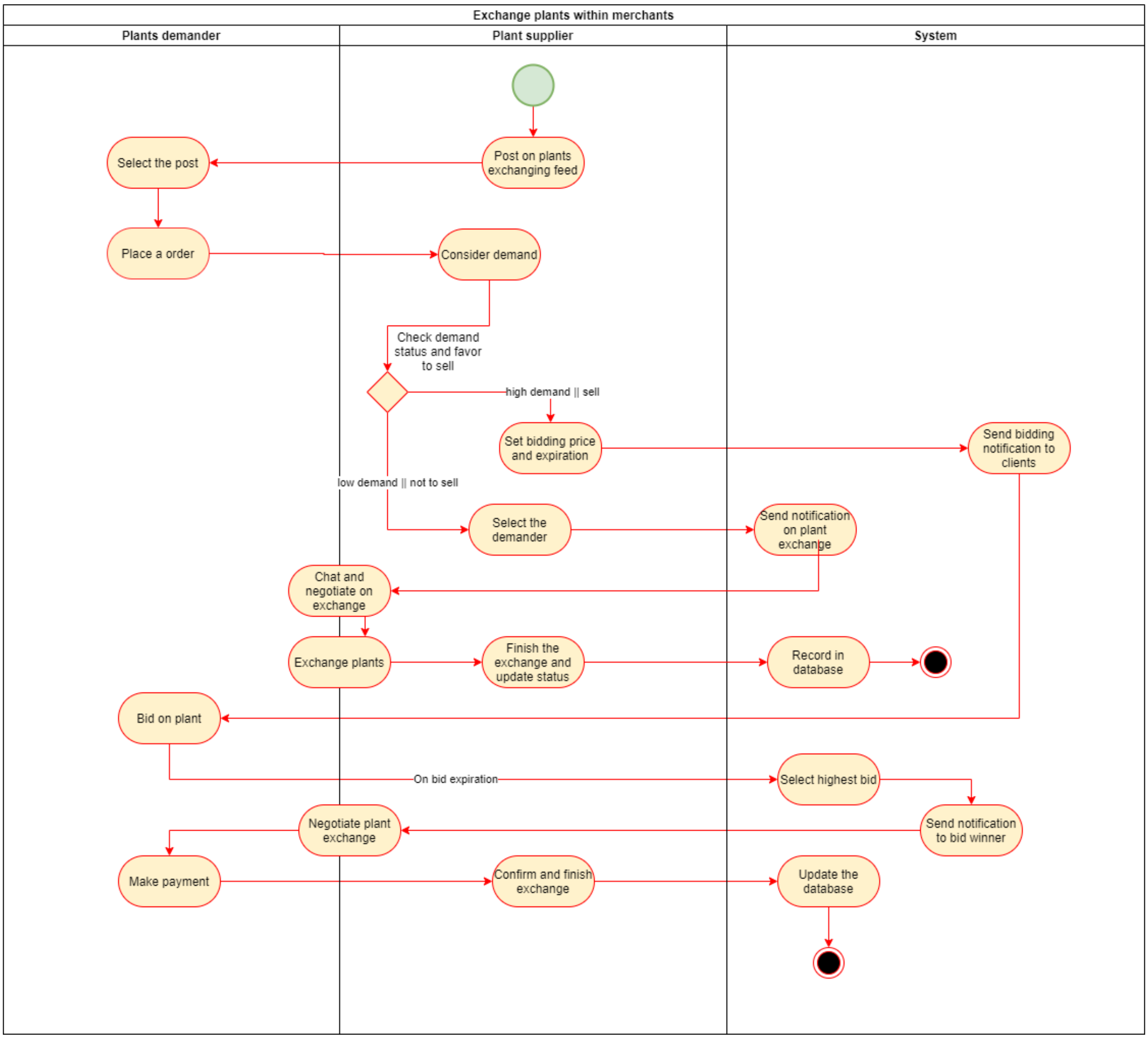
2) Plants and crops



8.3.3 Activity Diagram For Post Problems related to the Plants

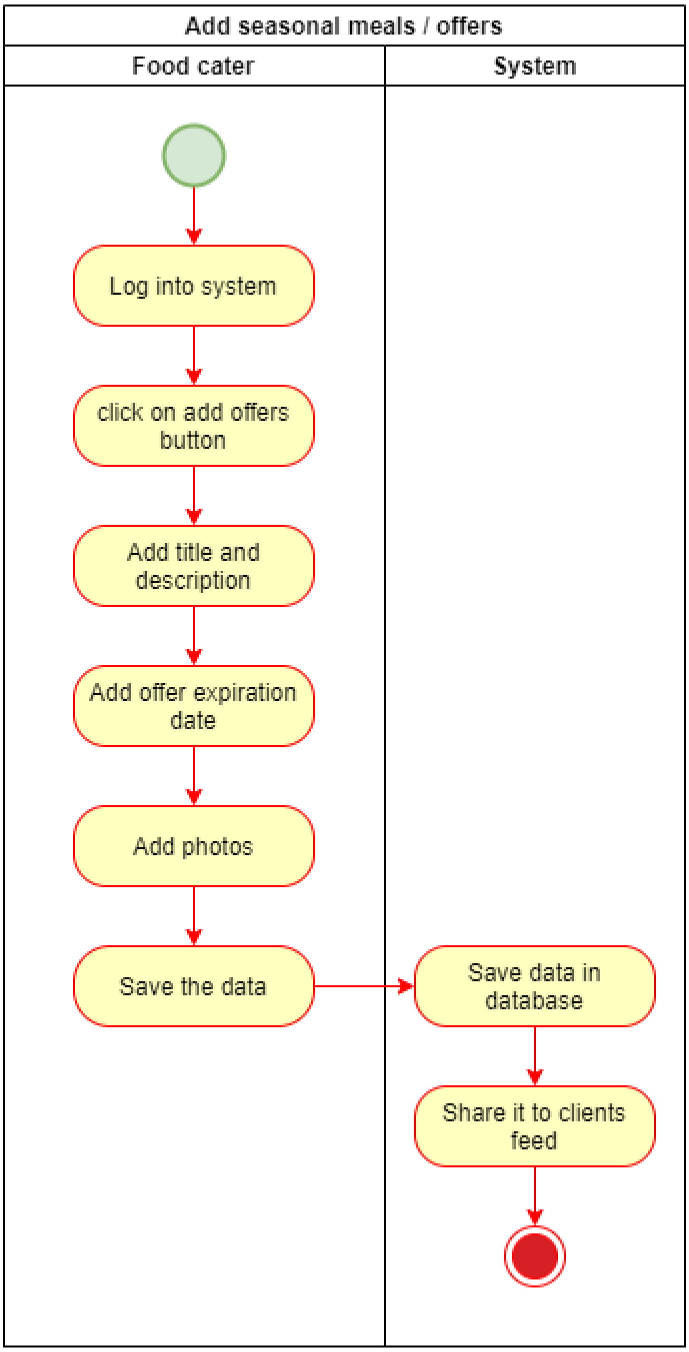


8.3.4 Activity diagram for Plants Management

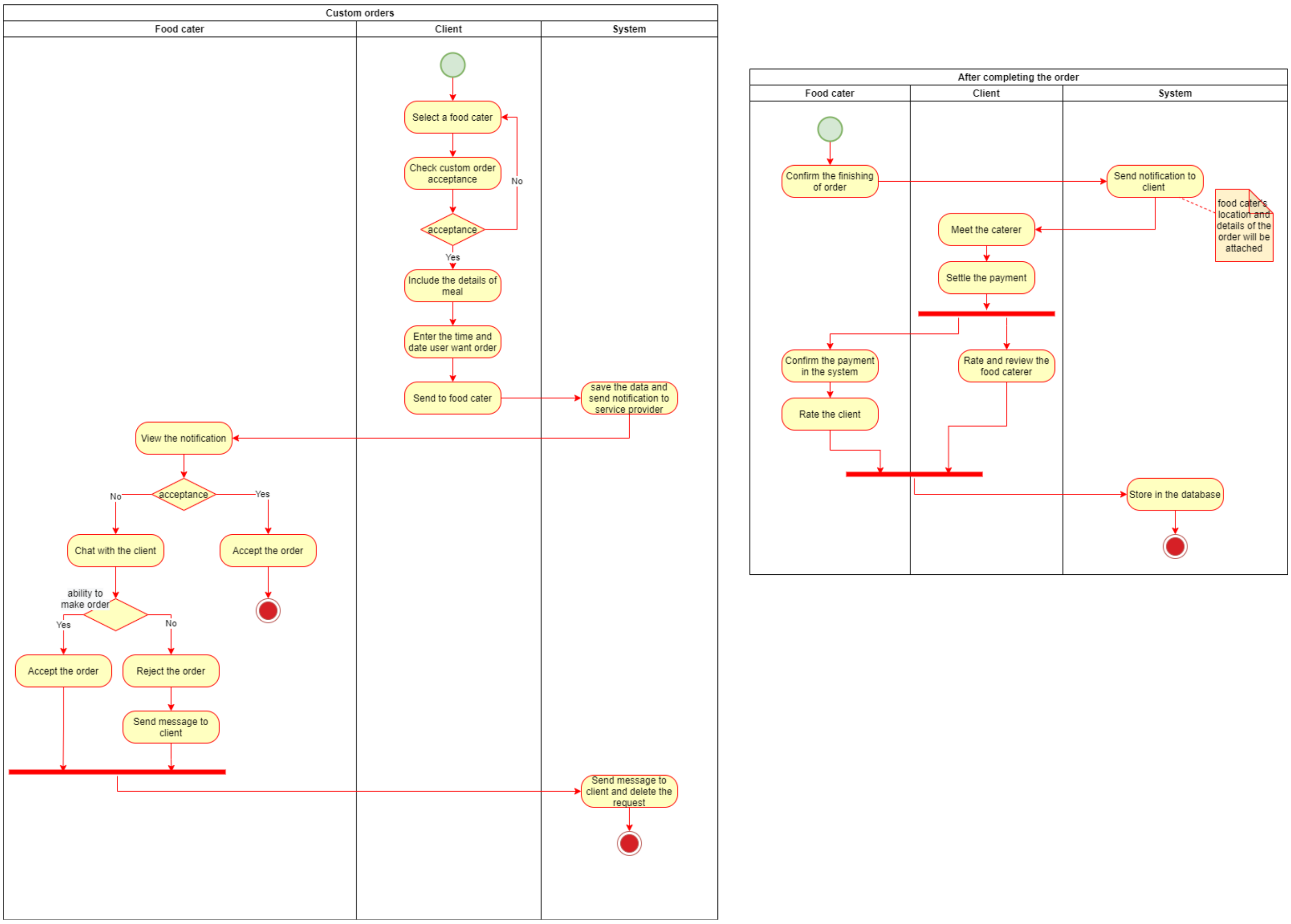

Exchange plants within merchants
Plants demander
Plant supplier
System
Post on plants exchanging feed
Select the post
Place a order
Consider demand
Check demand status and favor to sell
high demand || sell
Set bidding price and expiration
Send bidding notification to clients
low demand || not to sell
Select the demander
Send notification on plant exchange
Chat and negotiate on exchange
Exchange plants
Finish the exchange and update status
Record in database
Bid on plant
On bid expiration
Select highest bid
Send notification to bid winner
Negotiate plant exchange
Make payment
Confirm and finish exchange
Update the database

8.3.5 Activity Diagram for Exchange Plants with Merchants

3) Food and cuisine

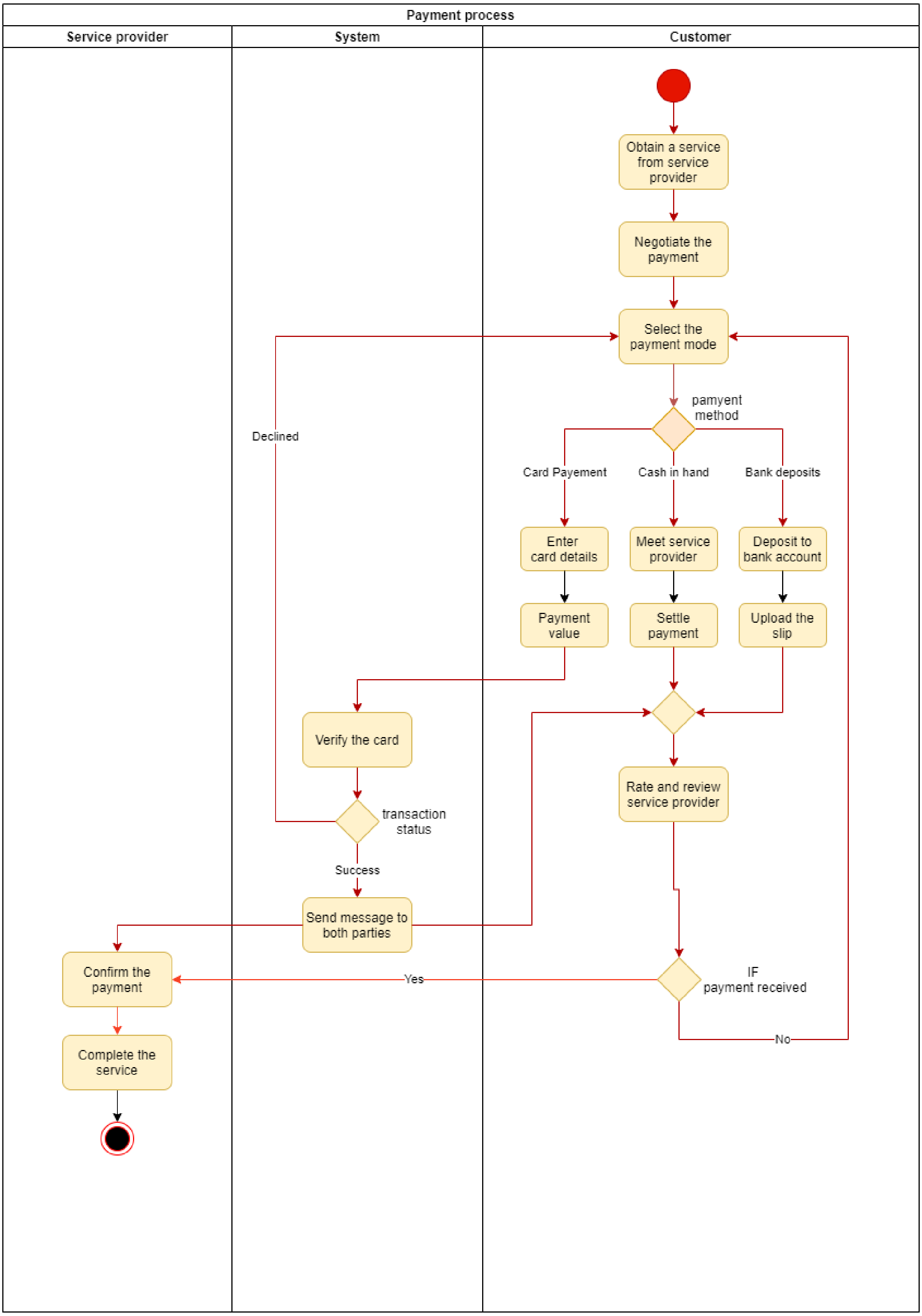

Add seasonal meals / offers
Food cater
System
Log into system
click on add offers button
Add title and description
Add offer expiration date
Add photos
Save the data
Save data in database
Share it to clients feed

8.3.6 Activity diagram for Seasonal Meals

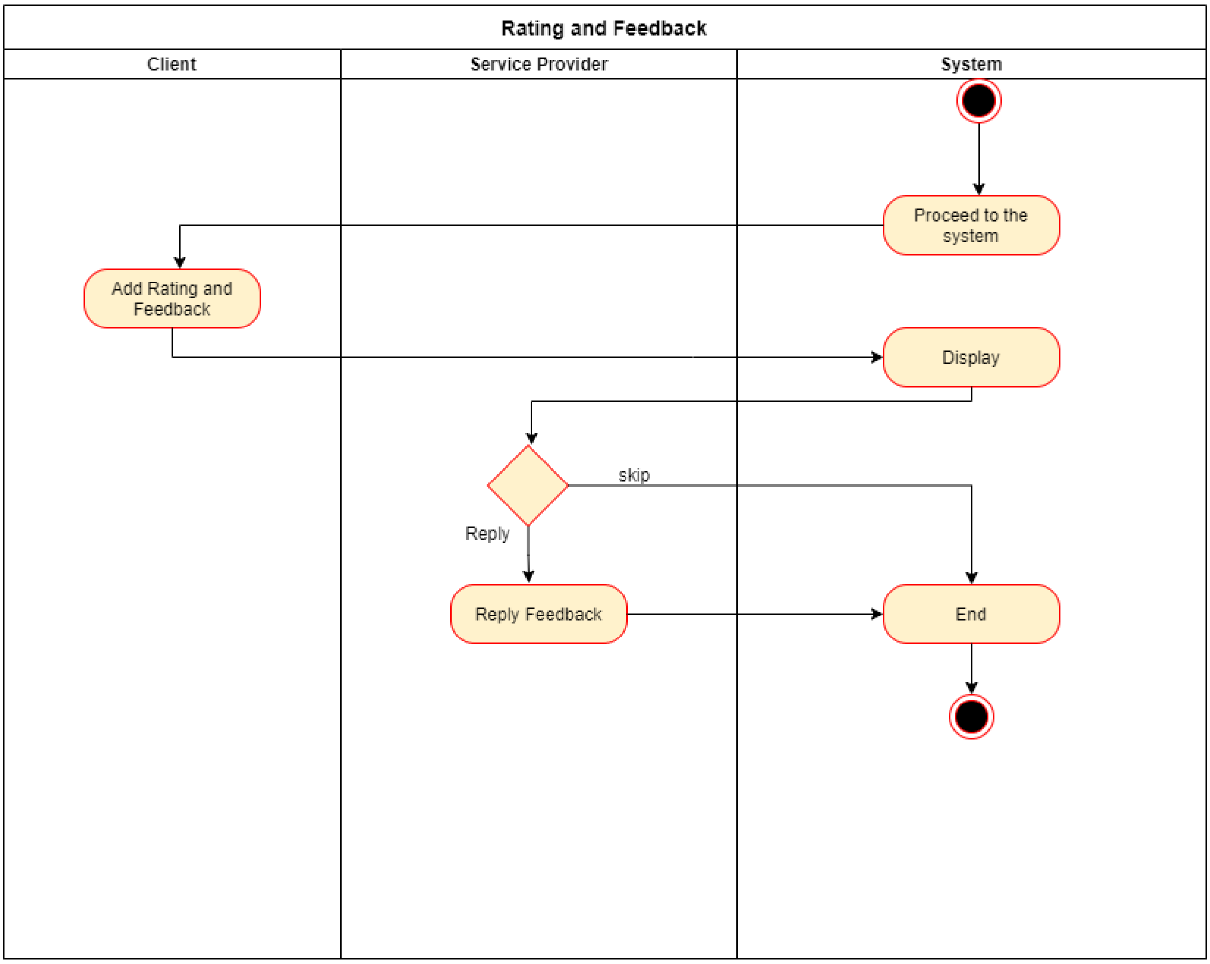

Custom orders
Food cater
Client
System
Select a food cater
Check custom order acceptance
acceptance
No
Yes
Include the details of meal
Enter the time and date user want order
Send to food cater
save the data and send notification to service provider
View the notification
acceptance
No
Yes
Chat with the client
Accept the order
ability to make order
Yes
No
Accept the order
Reject the order
Send message to client
Send message to client and delete the request
After completing the order
Food cater
Client
System
Confirm the finishing of order
Send notification to client
food cater's location and details of the order will be attached
Meet the caterer
Settle the payment
Confirm the payment in the system
Rate and review the food caterer
Rate the client
Store in the database

8.3.7 Activity diagrams Orders

4) Client

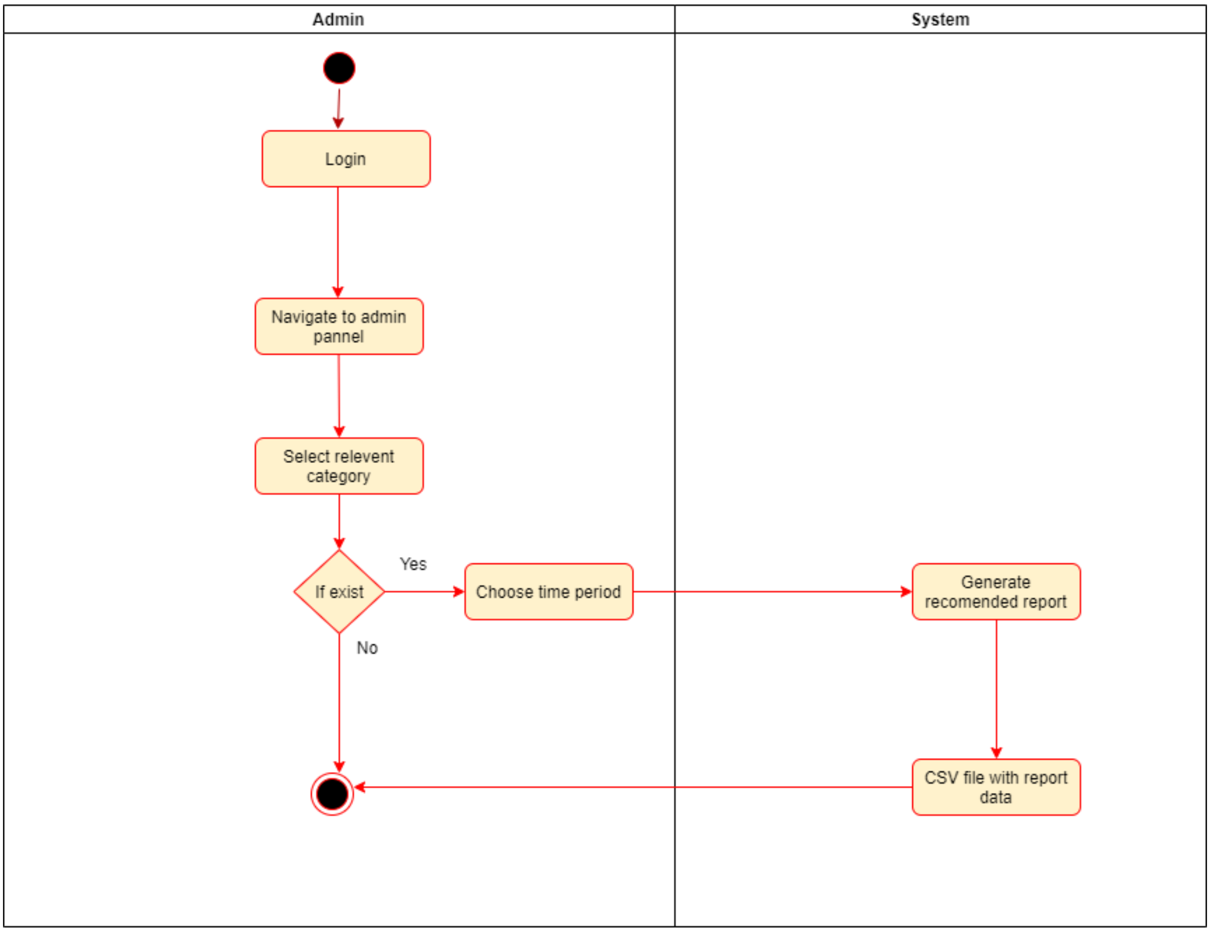


8.3.8 Activity Diagram for Payment Process

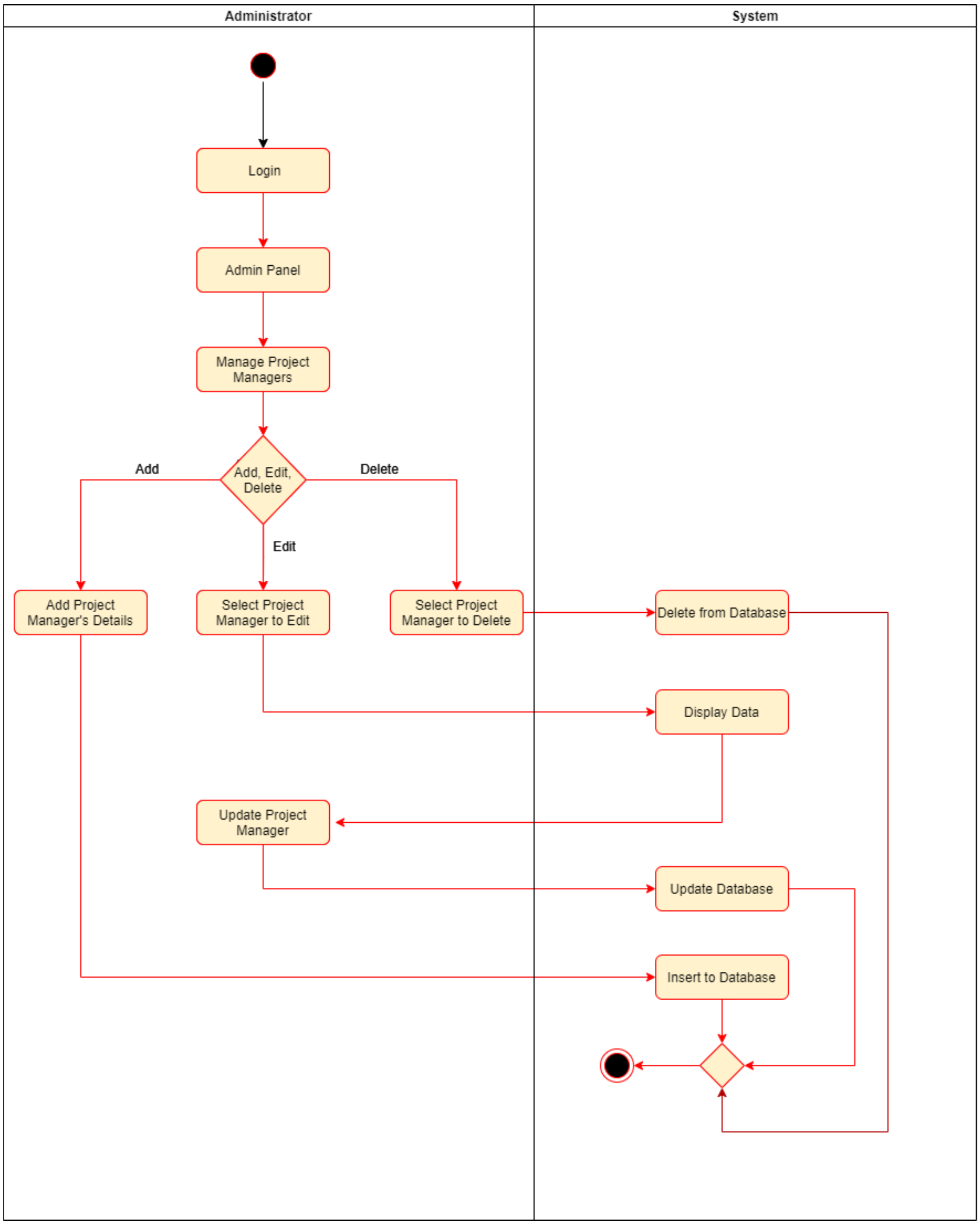


8.3.9 Activity diagram for Rating & Feedback

5) Administrator



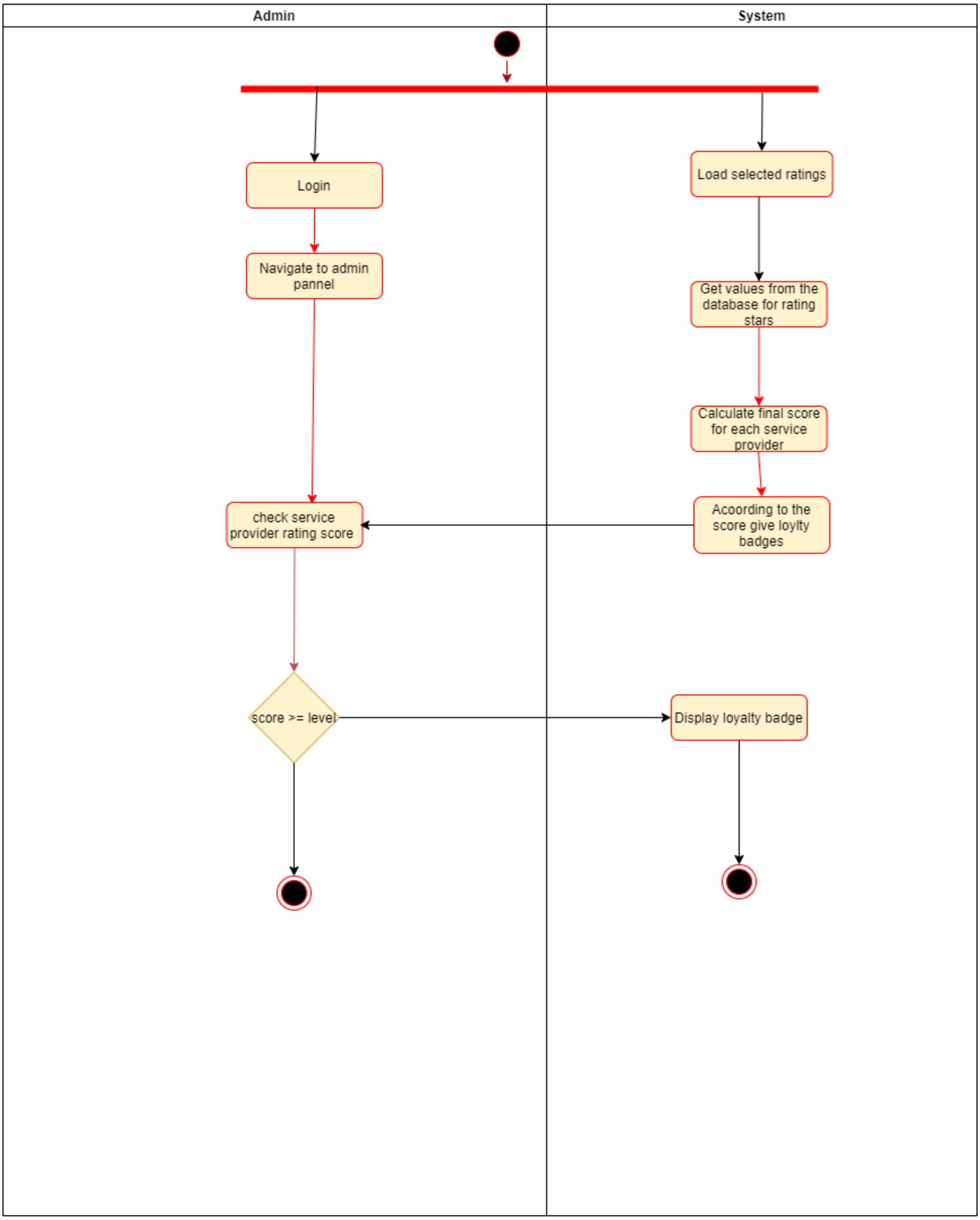
8.3.10 Activity diagram for Report Generation (For Admin)

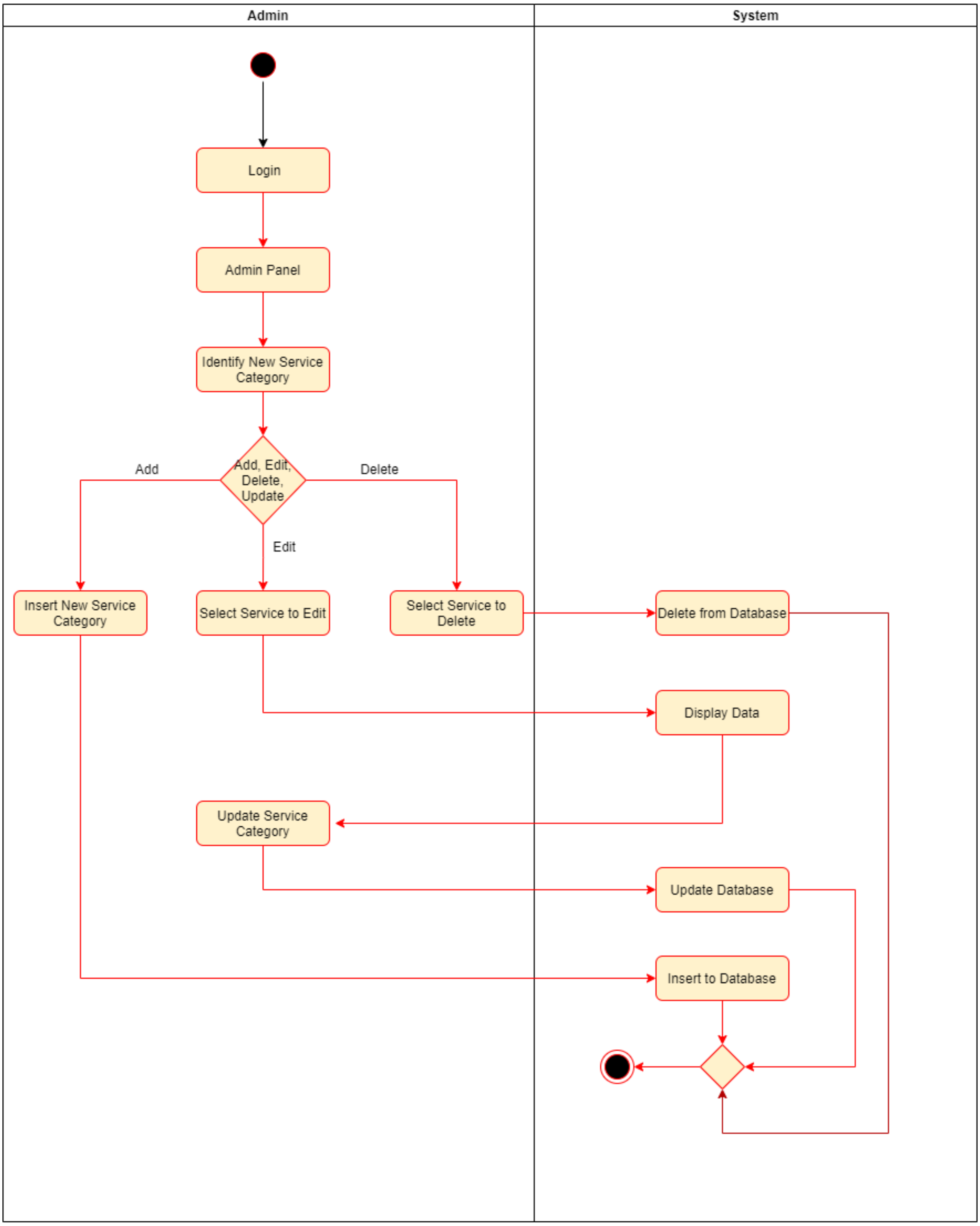


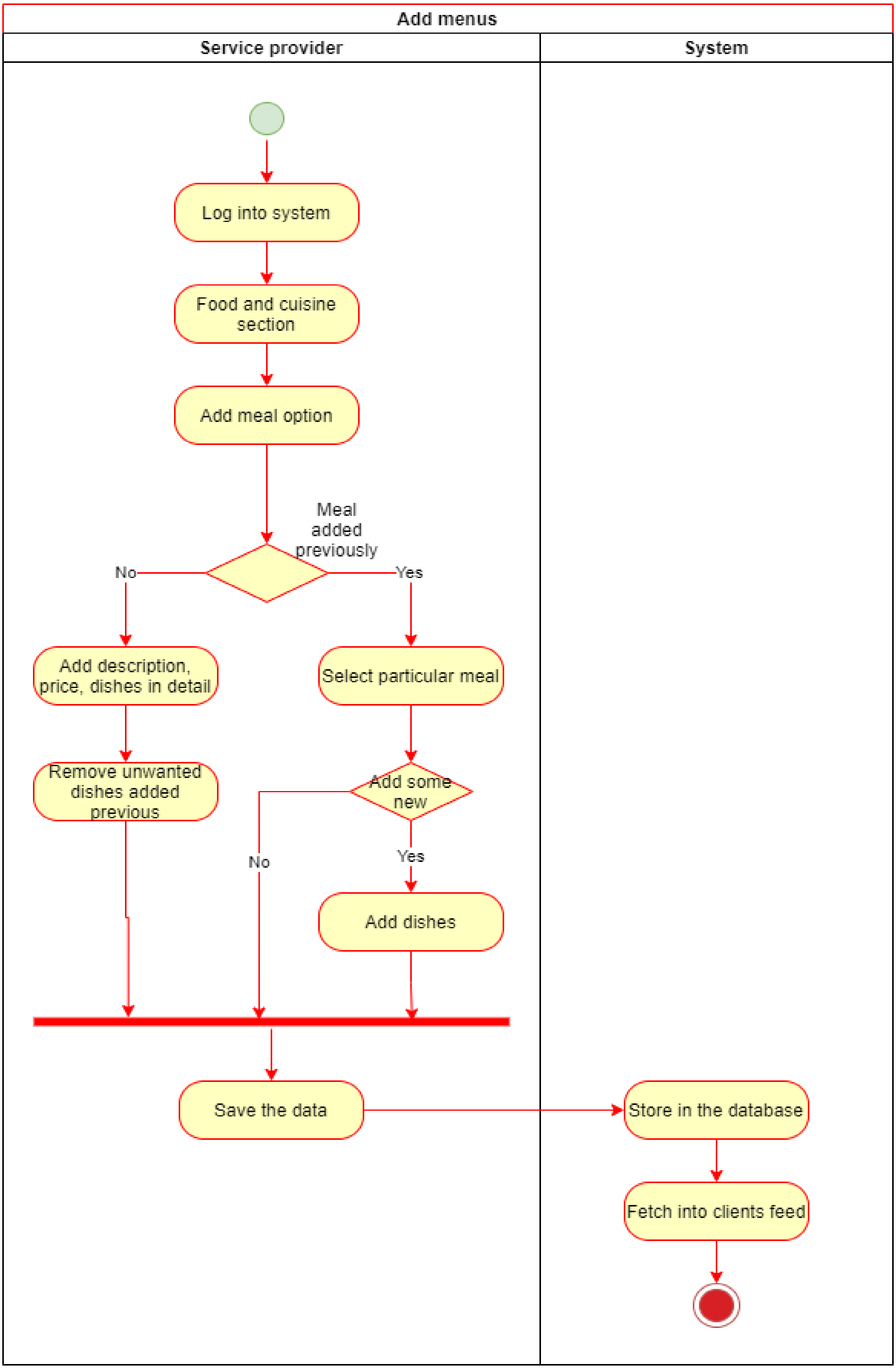
//active and deactive

8.3.11 Activity diagram for Admin

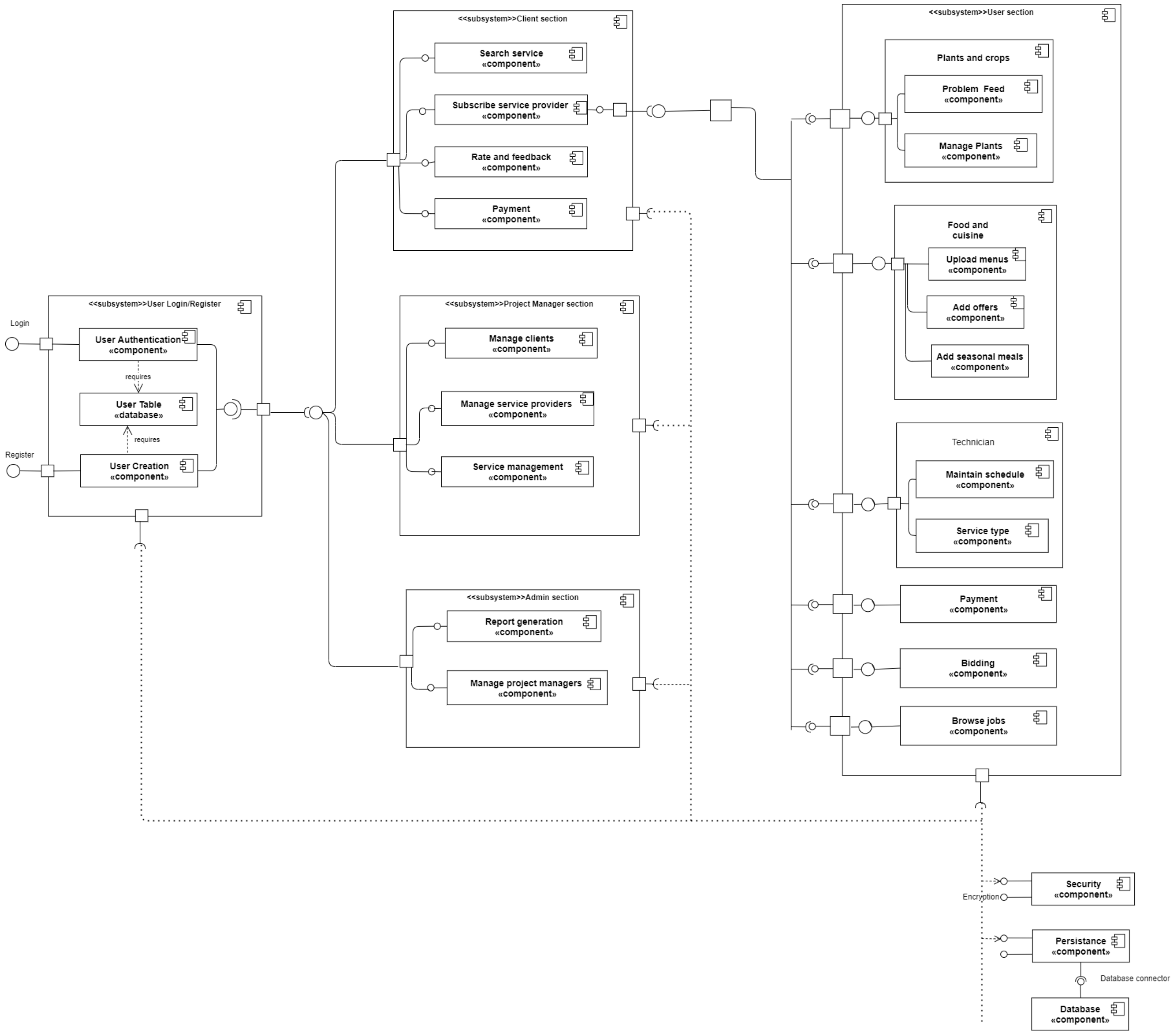
6) Project manager


Admin
System
Login
Navigate to admin pannel
check service provider rating score
score >= level
Load selected ratings
Get values from the database for rating stars
Calculate final score for each service provider
Acoording to the score give loylty badges
Display loyalty badge


Admin
System
Login
Admin Panel
Identify New Service Category
Add, Edit, Delete, Update
Add
Delete
Edit
Insert New Service Category
Select Service to Edit
Select Service to Delete
Delete from Database
Display Data
Update Service Category
Update Database
Insert to Database


Add menus
Service provider
System
Log into system
Food and cuisine section
Add meal option
Meal added previously
No
Yes
Add description, price, dishes in detail
Select particular meal
Remove unwanted dishes added previous
Add some new
No
Yes
Add dishes
Save the data
Store in the database
Fetch into clients feed

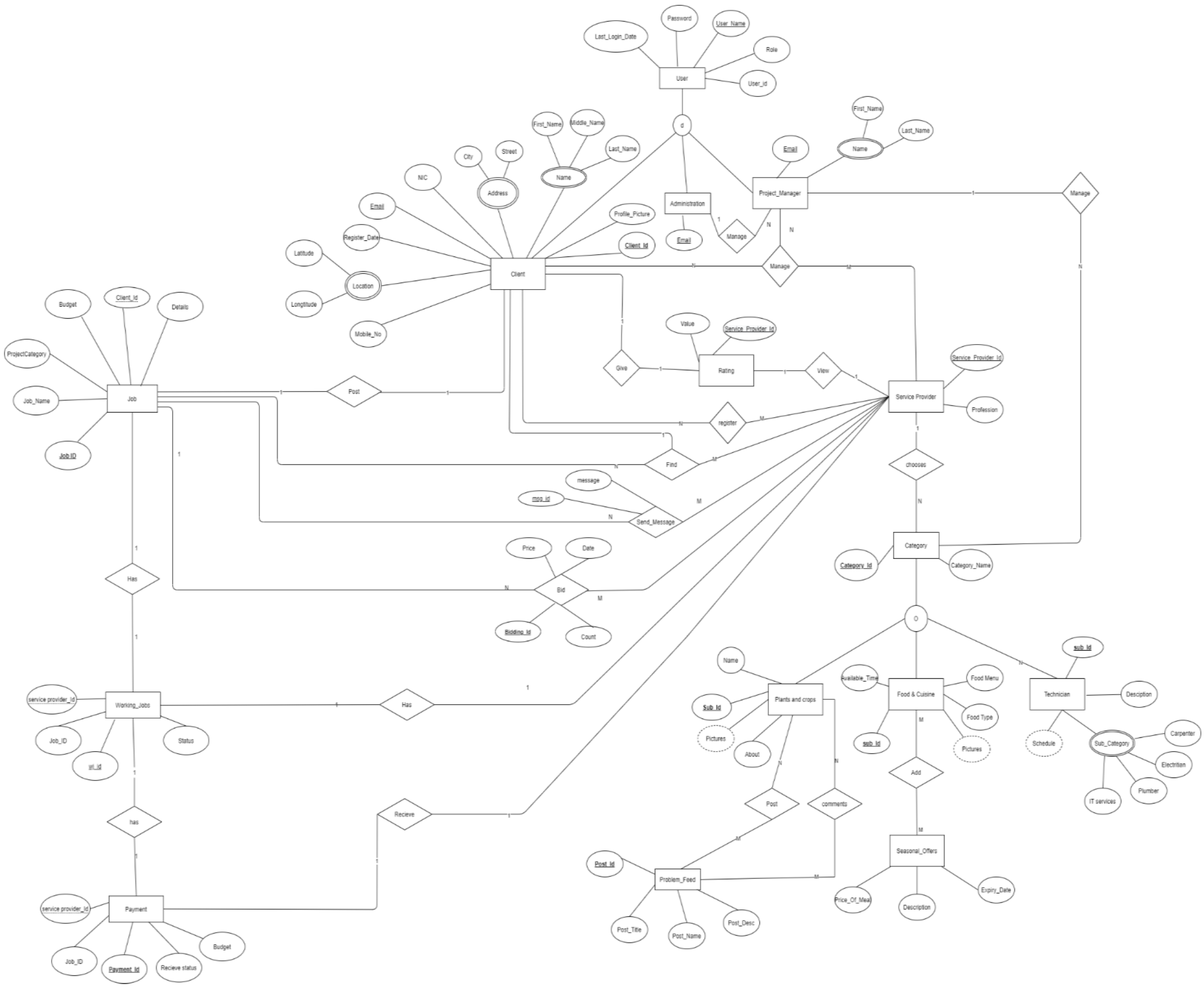
8.4 Component diagram



8.4 Component Diagram

link:- https://app.diagrams.net/#G1DhmhctVtJhpLuiOJTQH5vV82O8h-89yV

8.5 ER diagram

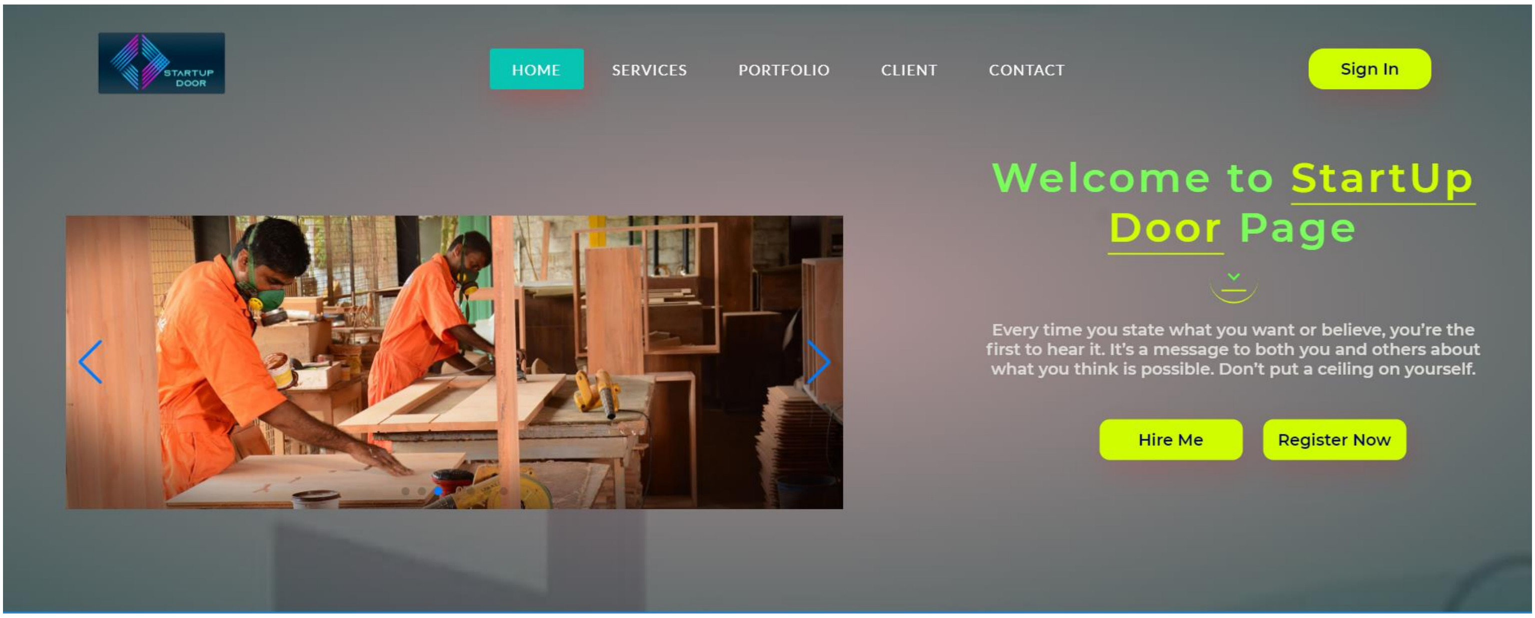


8.5 ER Diagram for Startup Door

link:-https://app.diagrams.net/#G1zqn1v2I0xSvIMWuJ2y6SZoKgdm5Rrg3x

9. User Interfaces

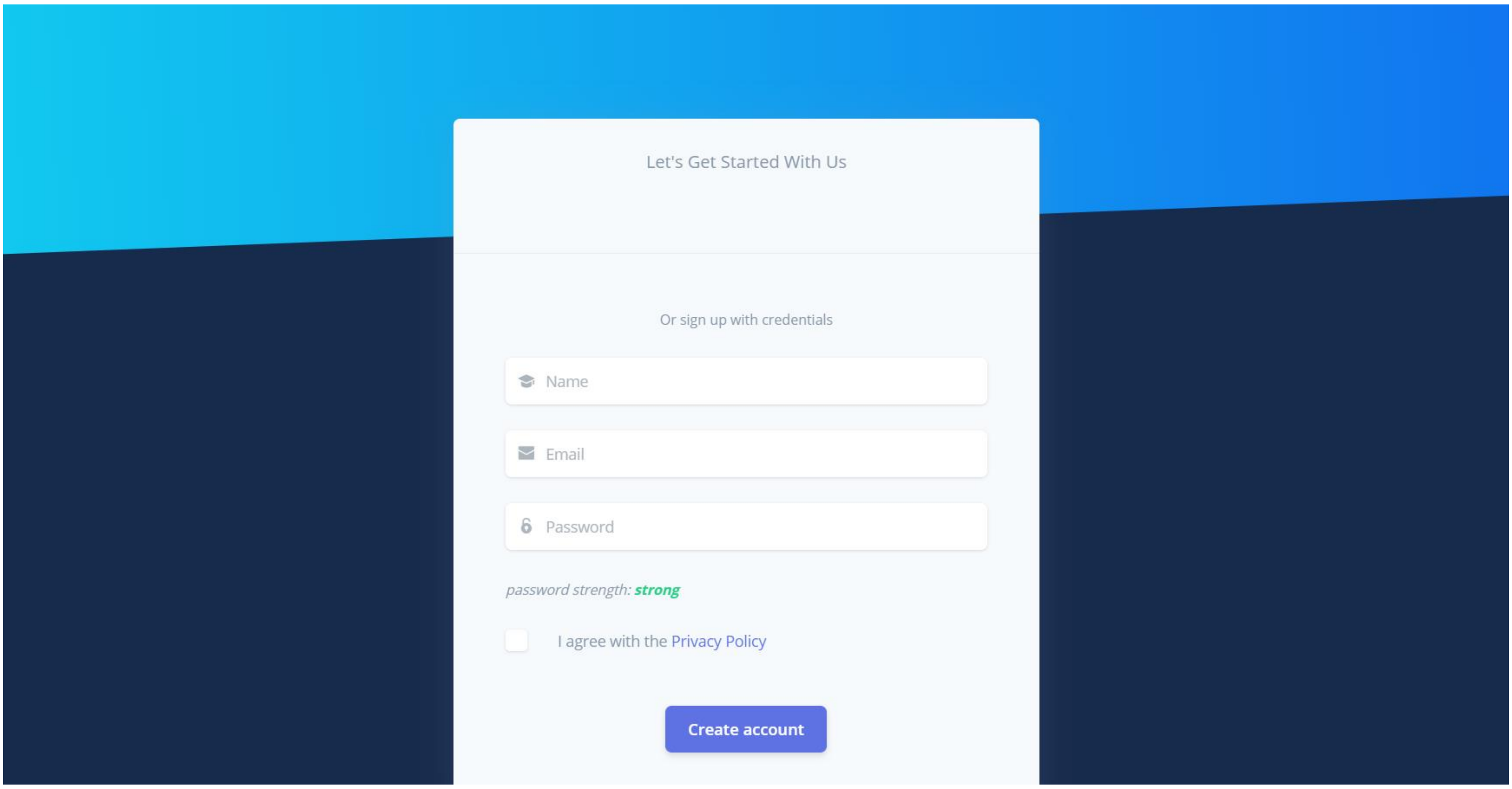
A brief introduction to the user interface design of your system and design decisions. Provide sample user interfaces (colour) to explain your decisions and justifications.



Welcome to StartUp Door Page

Every time you state what you want or believe, you're the first to hear it. It's a message to both you and others about what you think is possible. Don't put a ceiling on yourself.

9.1 Landing Page UI



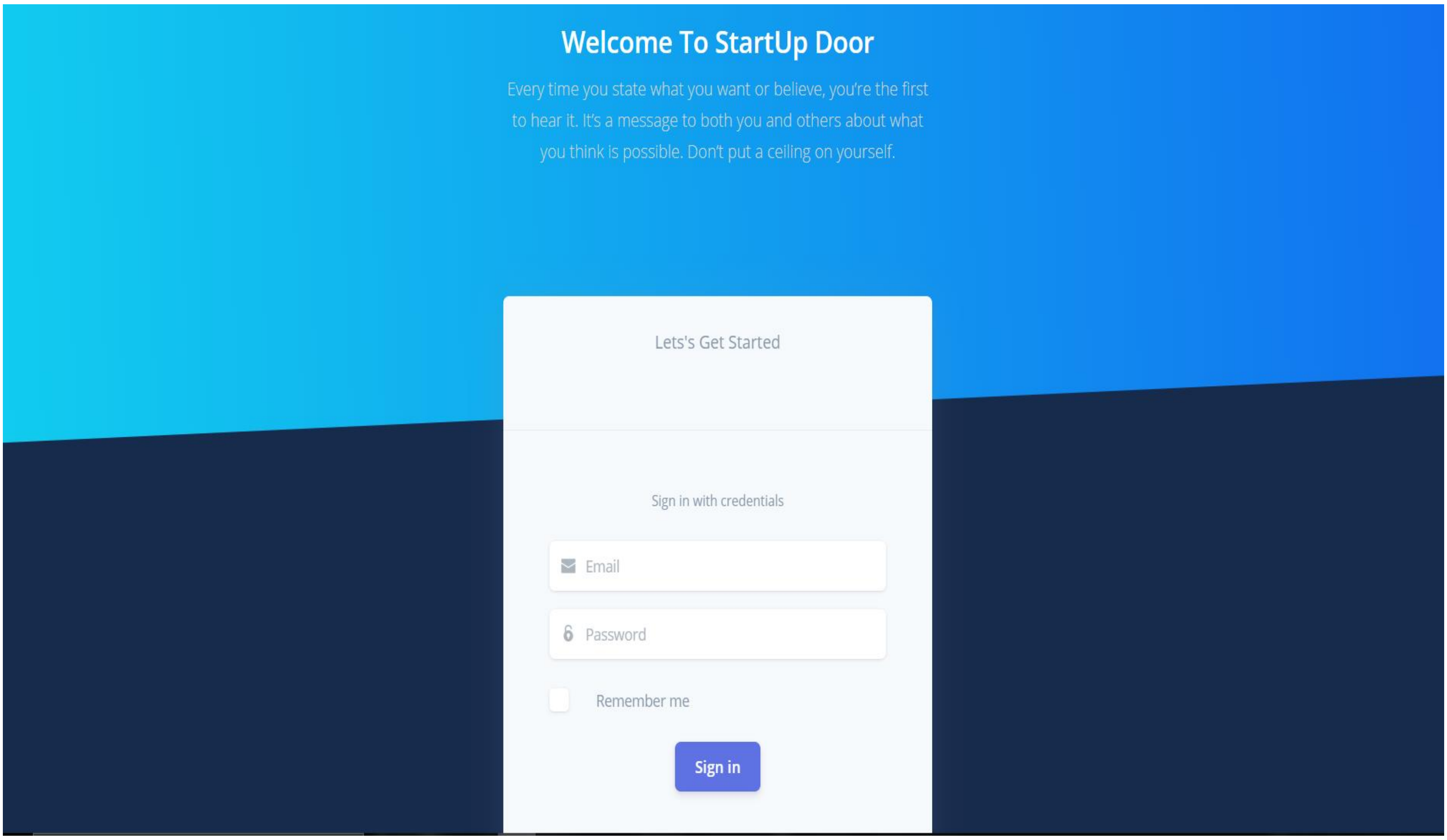
Let's Get Started With Us

Or sign up with credentials

password strength: ***strong***

□ I agree with the Privacy Policy

9.2 Register Page UI



Welcome To StartUp Door

Every time you state what you want or believe, you're the first to hear it. It's a message to both you and others about what you think is possible. Don't put a ceiling on yourself.

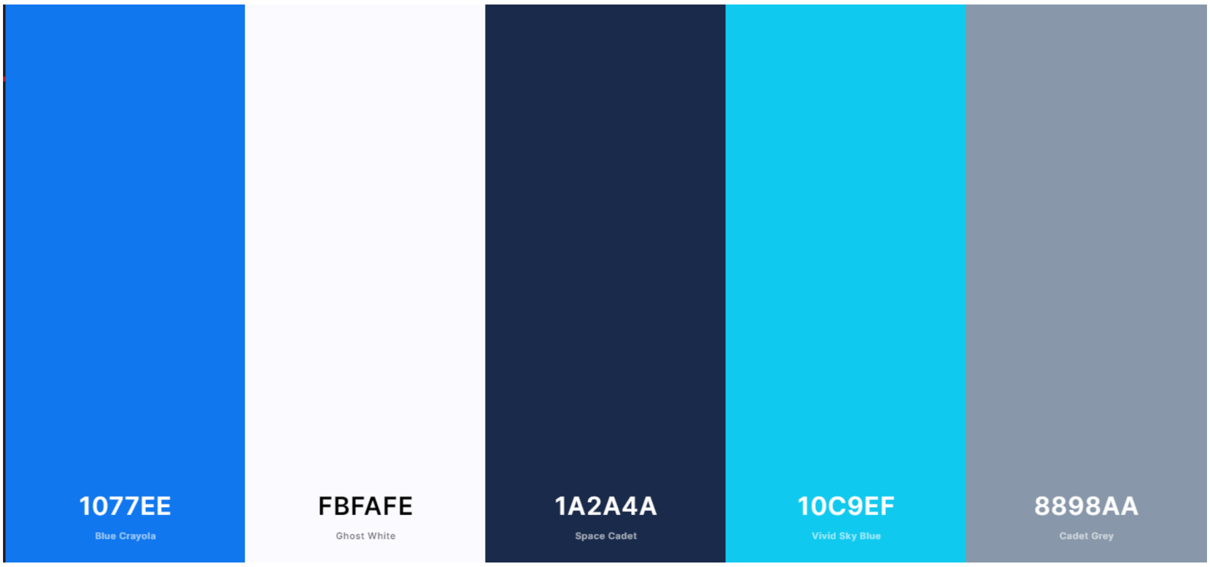
Lets's Get Started

Sign in with credentials

9.3 Login Page

Colour Palette

For the startup door, selecting a light color blue variant will enlighten the users to work easily, energetically & potentially. Blue represents people’s potentiality & enthusiasm for a task. For these reasons, I concluded to select a color pallet in the section below.



- Blue

We use blue-based colours evoking the natural world are relatively subdued, and colours aimed at older people are deeper. Also, We aimed at young people who tend to be pure and bright. So blue provide a like Positive Controlled, intellectual, harmonious, faithful, rational, sensible. This is an Entrepreneurship portal, So we need to provide a sensible, intellectual vibe to the clients and make it convenient to work with the system.

Using a light blue background is easy to read, it would be better to tone it down close to white cause our clients work with the system 24/7 hours. Pale grey, cream, or magnolia will be included in the necessary places.

10.Main deliverables of the system

1. Complete working software and source code (if the client doesn't want you to share the source code you have to produce the clients' consent and non-disclosure agreement signed with the client)
2. Complete Software Requirement Specification
3. User manual
4. Administrators manual together with deployment instructions
5. The license of the software (GPL, LGPL, BSD, Proprietary, etc.)

11.The Project Plan

This is a group project where all members are supposed to contribute equally as well as in the best possible way to make it a successful project. The project has the start time and end time within which it will be completed. There will be several milestones to be achieved to complete the work task within the time frame.

Develop the work breakdown structure with the timeframe and milestones. If possible, develop the Gantt Chart too, covering the system development, system documentation, and project evaluation.

Make sure you address all the issues raised and feedback received at proposal presentations to the project plan. You must give a self-reflection on your progress to your time plan.

Deployment Plan

When Deploying **StartUp Door** React Web Application, Selected **AWS** as the Cloud Service provider. AWS is a cloud computing platform that supports PostgreSQL through a fully managed database service with Amazon Relational Database Service (RDS) which is ideal to deploy our StartUp Door. Additionally use the free 12 months trial that was given. That is another major reason to Select AWS as a Cloud service platform to deploy the StartUp Door.

- Using Amazon EC2 instance to create a Bucket in the console in order to store the Application source codes for the deployment.

This virtual server is ideal for our web portal to be hosted in Aws infrastructure.

1) **Multiple Storage Options**

Amazon EC2 allows users to choose between multiple storage options based on our Startup Door Web portal requirements. Amazon EBS is a durable, block-level storage volume that you can attach to a single, running Amazon EC2 instance. After hosting our system on Aws , Amazon EBS can be used as a primary storage device for data that requires frequent and granular updates.

2) EBS-optimized Instances

Amazon EC2 provides instance types as EBS-optimized instances. (**For M6g, M5, M4, C6g, C5, C4, R6g, P3, P2, G3, and D2 instances**) This feature with our low budget does not provide any additional cost.

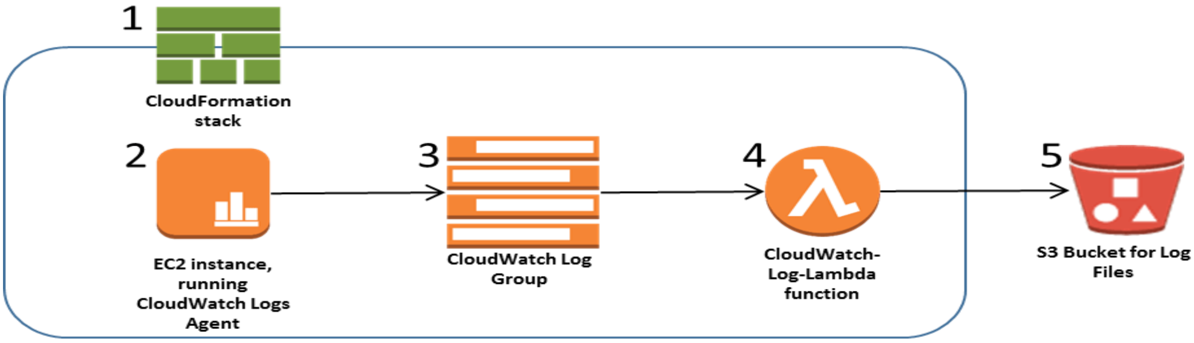
EBS-optimized instances enable EC2 instances to fully use the IOPS provisioned on an EBS volume. EBS-optimized instances deliver dedicated throughput between Amazon EC2 and Amazon EBS, with options between 500 and 4,000 Megabits per second (Mbps) depending on the instance type used which is really needed in our system to manage the clients and Service providers without any downtime of the System.

3) Compute Optimized

Also, the privilege to select computed optimized instances is ideal for the **Startup Door** that benefits from high-performance processors, suited for batch processing workloads, media transcoding, and high-performance web servers. In our System clients privilege with a **bidding system** that needs a high-performing web server to mitigate the latency.

4) Ease of Using

After creating an account with AWS, they provide a well-structured Cpanel to work with our web. Provide a lot of options & control mechanisms to utilize our web in Aws Cloud infrastructure.


1
CloudFormation stack
2
EC2 instance, running CloudWatch Logs Agent
3
CloudWatch Log Group
4
CloudWatch-Log-Lambda function
5
S3 Bucket for Log Files

Basic High-level architecture in EC2 instance, when we deploy our StartUp Door

12.References

[1] **GlassDoor Web Portal**
"Browse Jobs by Location." Glassdoor. https://www.glassdoor.com/sitedirectory/city-jobs.htm
(accessed: July 8, 2021).

[2] **Material UI Documentation**
https://material-ui.com/
(accessed: July 2, 2021).

[3] **Feasibility Study**
"Feasibility Study - Definition, How to Conduct, Contents." Corporate Finance Institute.
https://corporatefinanceinstitute.com/resources/knowledge/other/feasibility-study
(accessed: June 24, 2021).

[4] **React Js Tutorial**
https://reactjs.org/tutorial/tutorial.html
(accessed: July 2, 2021).

13.Declaration

A project proposal is a contract between students who will undertake the project and teachers who will supervise and coordinate this course module. Hence, all members of the project team should declare their willingness and readiness to carry out the project in their best within the rules, regulations, and code of ethics for this course.

We as members of the project titled Startup Door, certify that we will carry out this project according to guidelines provided by the coordinators and supervisors of the course as well as we will not incorporate, without acknowledgment, any material previously submitted for a degree or diploma in any university. To the best of our knowledge and belief, the project work will not contain any material previously published or written by another person or ourselves except where due reference is made in the text of appropriate places.

Name	***Signature***
(i) W.P Pallewatta	
(ii) R.P.P.B.Somarathna	
(iii) D.T.Athukorala	
(iv) M.N.J Rathnayaka	Janith
(v) D. M. Samarasinghe	