SCHOOL OF COMPUTER SCIENCES

CSE 441- SOFTWARE PROCESS AND QUALITY ASSURANCE 2024/2025

ASSIGNMENT 1

NAME	MATRIC NUMBER
THANEES A/L SEHKAR	158861

LECTURER'S NAME:

Mohd Heikal Husin, Ts. Dr.

SUBMISSION DATE: 13 December 2024

TABLE OF CONTENTS

Contents

1. Project Background

1.1 Problem background

In today's rapidly urbanizing world, the demand for reliable home maintenance services has grown significantly, encompassing various needs from plumbing and electrical repairs to landscaping and pest control. However, the home service market suffers from several critical challenges that create barriers for both customers and service providers.

A key issue is the fragmentation of information, where customers struggle to access reliable and comprehensive details about service providers. This fragmentation occurs because information is scattered across multiple platforms and sources, leaving customers without a centralized, trustworthy repository to make informed decisions. Additionally, customers face difficulties in verifying service provider credentials and expertise, raising serious concerns about service quality and safety. The issue is further exacerbated by the dispersion of customer reviews and ratings, making it difficult to determine a service provider's reliability based on inconsistent feedback spread across various platforms.

Moreover, the absence of a real-time booking system complicates the scheduling process, often leading to conflicts, long wait times, and communication breakdowns between customers and service providers. The lack of a standardized system for price transparency frequently results in unexpected costs for customers, eroding trust and satisfaction in service delivery. Traditional methods of payment and transaction handling also add complexity to the process, creating potential trust issues during financial transactions.

Previous attempts to address these challenges have typically focused on traditional directory listings or basic online platforms that offer limited functionality and verification processes. While these solutions provide some level of information aggregation, they often need to improve in offering real-time availability, secure payment systems, and comprehensive service provider verification which are the essential elements for building trust in the home service marketplace.

Through careful analysis of these challenges, there is a clear need for a comprehensive and modern solution that centralizes information, establishes verification mechanisms, facilitates real-time booking, enhances pricing transparency, and integrates secure payment systems. This would not only improve the efficiency and reliability of home maintenance

services but also promote economic growth in the service sector by fostering trust and enhancing user experiences.

1.2 System overview

1.2.1 System objectives

The objectives of HandyHub are:

- To connect customers with verified and skilled service providers for high-quality home maintenance services.
- To enable customers to book and schedule maintenance services based on their preferences and availability.
- To ensure transparency by providing detailed service descriptions, customer reviews, and ratings to help customers make informed decisions.
- To enable secure and seamless payment processing options, including online payments and cashless transactions.

1.2.2 System module breakdown

HandyHub is decomposed into 6 modules: Admin and User Management, Service Provider Management, Booking, Payment, Review and Rating, and Communication.

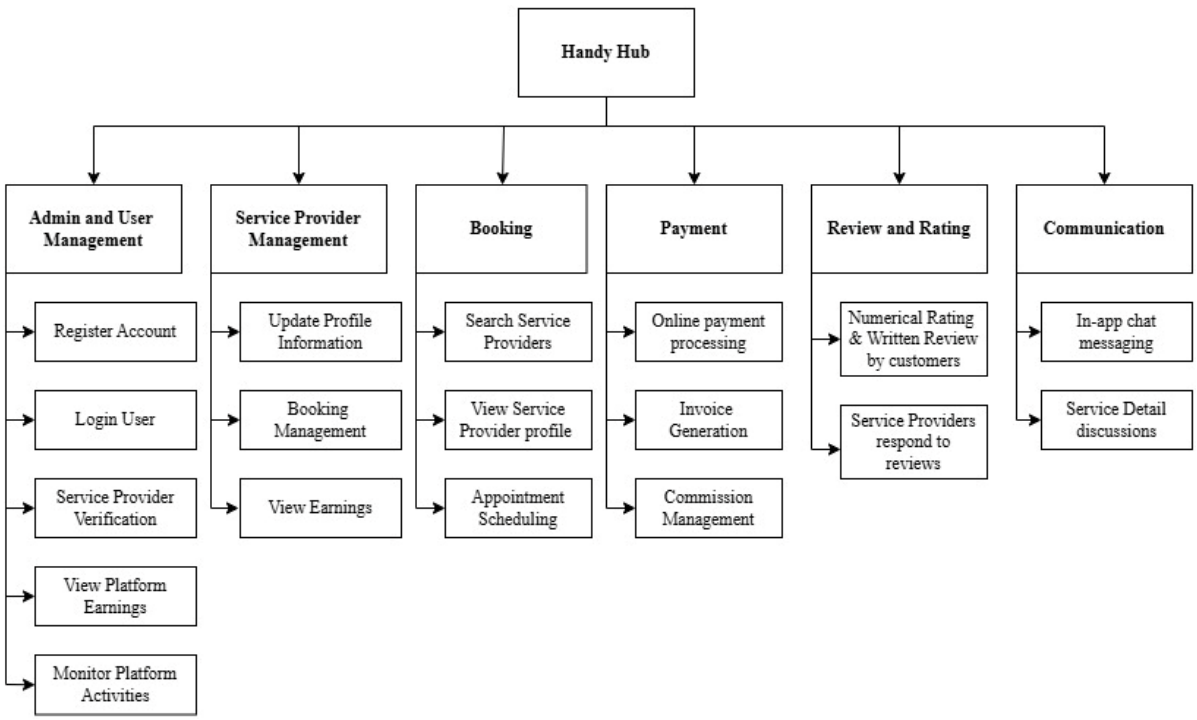

Figure 1: System Decomposition Diagram

1.2.3 System description and function

Admin and User Management Module:

The Admin and User Account Management module serves as the foundation for access control and identity management within HandyHub. It manages three distinct types of user registrations: service providers, who must submit professional credentials and verification documents for their specific service categories; customers, who provide basic registration information like name, email, and contact details; and administrators, who oversee platform operations. This module implements a secure login system with role-based access control, ensuring users can only access features relevant to their role. Customers and service providers can manage their profiles and update their information, while administrators have comprehensive oversight capabilities including the crucial role of verifying service provider credentials to ensure only qualified professionals are listed on the platform. Moreover, the admin also will be able to track their platform earnings.

Service Provider Module:

The Service Provider Module provides service professionals with a dedicated dashboard to effectively manage their activities on HandyHub. Service providers can update and maintain their profile information to ensure that their details remain accurate and up to date. Moreover, they can manage bookings by viewing detailed booking information like customer details, service requirements, and scheduled time. The dashboard also provides an earnings overview, enabling service providers to track their income from completed services.

Booking Module:

The Booking Module streamlines the service scheduling process with an intuitive interface for customers. It allows customers to search for specific service categories, view provider profiles and availability, and schedule appointments based on their preferences. A distinctive feature is the requirement for customers to upload images of the problem or task during booking, providing service providers with a clear visual understanding before the appointment. The module also handles booking confirmations, and scheduling notifications and maintains a comprehensive booking history for both customers and service providers.

Payment Module:

The Payment Module delivers a secure and transparent financial transaction system within HandyHub. It integrates with an established payment gateway to enable seamless online payments for services. The module handles both initial service payments and any additional costs that may arise during service delivery. It automatically generates detailed invoices for completed services and maintains a comprehensive transaction history. The module also manages platform commissions and ensures proper distribution of payments between service providers and the platform.

Review & Rating Module:

The Review & Rating Module allows customers to evaluate service providers' performance. Customers can provide both numerical ratings and written reviews detailing their experience. The module supports photo uploads with reviews, allowing customers to showcase the completed work. Service providers can respond to reviews, creating a dynamic feedback system that helps build trust and accountability. The ratings contribute to service providers' overall reputation on the platform, helping future customers make informed decisions while incentivizing service providers to maintain high service standards.

Communication Module:

The Communication Module enables direct interaction between customers and service providers within HandyHub. The integrated in-app chat system allows customers and service providers to discuss service details, and prices, clarify requirements, and coordinate appointments efficiently.

2. Software Process Model

For the development of HandyHub, the **Agile Software Development** model is the most suitable choice. Agile is an iterative and flexible framework that ensures the project is broken down into manageable segments, called sprints, allowing for incremental development of functional modules. This approach is especially effective on a 10-week timeline, as it encourages continuous progress while responding to changing demands. The adaptability of agile makes it easy to incorporate changes into the project without disrupting overall progress.

As a developer, agile allows me to focus on one module at a time, ensuring steady progress and preventing me from being overwhelmed. Working in sprints allows me to deliver core functionalities like user authentication, booking interfaces, and payment integration early in the process, while still leaving room for improvements based on real-world testing and feedback from my supervisor and stakeholders. The flexibility of agile allows us to pivot and adapt without significant delay if issues or better approaches arise during development.

A significant advantage of Agile is its focus on practical implementation over exhaustive documentation. While I will maintain essential documentation for my final year project report, Agile allows me to prioritize the development of working features. This ensures HandyHub's critical components, such as the admin panel, service provider module, and booking system, are ready to go quickly, leaving ample time for development.

Another key feature of Agile is continuous testing and iterative development. By integrating testing into each sprint, I can identify and resolve potential issues early, ensuring a stable and reliable system. The time-boxed structure of sprints helps me set clear goals and deadlines, enabling efficient resource and time management to stay on track with my project.

Execution Plan for HandyHub in 10 weeks:

Table 1: Execution Plan for HandyHub

Sprint	Duration	Tasks	Deliverables
Sprint 1	Week 1-2	Finalize requirements, set up the development environment, design system architecture, and implement admin module and user management.	Basic system architecture and the admin module.
Sprint 2	Week 3-4	Develop the service provider and booking modules, including profile management and booking system interface.	Functional service provider and booking modules.
Sprint 3	Week 5-6	Build backend APIs for the booking system and integrate a secure payment gateway.	Working backend with booking and payment features.
Sprint 4	Week 7-8	Develop the communication module and rating and review system.	Messaging and review features.
Sprint 5	Week 9-10	Perform system integration, testing, bug fixes, and deployment, and prepare for project submission.	Fully functional HandyHub system ready for demonstration.

3. Software Requirements Specification

3.1 System requirements

3.1.1 Functional requirements

Table 2: Functional requirements of HandyHub

#	Requirement	Description
1	Login user	Users should be able to log in to their existing account (customer, service provider, or admin).
2	Register User	The user should be able to register a new account with the appropriate role selection.
3	Verify service provider	Admin should be able to verify service provider credentials and documentation.
4	Update profile info	The service provider should be able to update their profile information and service details.
5	Logout user	Users should be able to sign out of their accounts securely.
6	Manage service listings	Service providers should be able to create and modify service listings.
7	Set pricing	Service providers should be able to set and update service pricing.
8	Track earnings	Admin and service providers should be able to view and track their earnings/payments.
9	Search service providers	Customers should be able to search service providers for specific service categories.
10	View provider profiles	Customers should be able to view service provider profiles and their reviews and ratings.

11	Schedule appointments	Customers should be able to book appointments with service providers.
12	Upload images	Customers should be able to upload images of problems during booking.
13	Process payments	The system should be able to process secure online payments.
14	Generate invoices	The system should be able to generate automated invoices for services upon payment completion.
15	Submit reviews	Customers should be able to submit ratings and written reviews upon service completion.
16	Respond to reviews	Service providers should be able to respond to customer reviews.
17	Send messages	Customers and Service Providers should be able to communicate through an in-app chat system.
18	View booking history	Users should be able to view their booking history.
19	Handle additional costs	Service providers should be able to add extra charges when needed.
20	Handle commission calculation	The system should be able to automatically calculate platform commission from customer payments, deduct it from the total payment, and distribute the remaining balance to the service provider.

3.1.2 Non-functional requirements

Table 3: Non-functional requirements of HandyHub

Quality Attribute	Requirement Description
Portability	• The system should be accessible on multiple platforms (Windows, macOS, iOS, Android). • The system should be compatible with major web browsers (Chrome, Firefox, Safari, Edge). • The system should maintain consistent functionality across different devices and screen sizes.
Security	• The system should encrypt all user data and sensitive information using industry-standard protocols. • The system should implement secure payment gateway integration for financial transactions. • The system should enforce strong password policies and user authentication
Reliability	• The system should achieve 99.9% uptime during operational hours. • The system should handle concurrent user sessions without performance degradation.
Scalability	• The system should support up to 10,000 concurrent users.
Performance	• The system should load pages within 3 seconds. • The system should process payments within 5 seconds. • The system should handle image uploads efficiently.
Usability	• The system should provide intuitive navigation and clear menu structures. • The system should have a consistent layout and design across all pages.

3.2 Use Case Modelling

3.2.1 Use case diagram

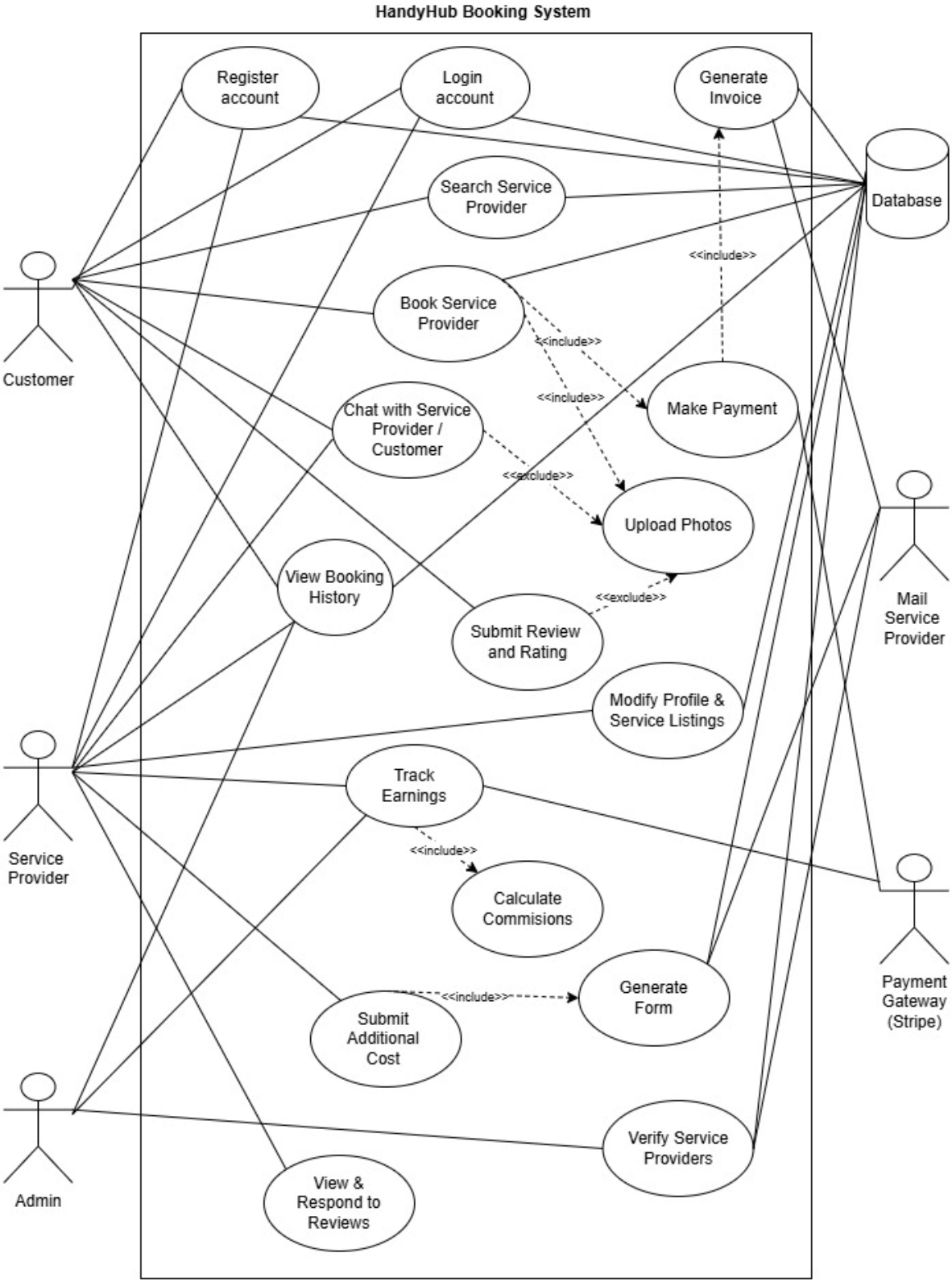

Figure 1: Use Case Diagram of HandyHub

3.2.2 Use case overview

Table 4: Use Case Overview

Use Case ID	Use Case Name	Use Case Description
U001	Register Account	New users, whether customers or service providers, can create an account on HandyHub by providing their personal information, email, password, and contact details. Service providers need to provide additional information for verifications.
U002	Login Account	Registered users can securely log in to their HandyHub accounts using their credentials (email and password).
U003	Search Service Provider	Customers can search for service providers based on the specific service category they need.
U004	Book Service Provider	Customers can schedule appointments by selecting a service provider, specifying their requirements, and choosing an available time slot in the booking form. Customers are required to upload a picture of the problem or task to provide service providers with a clearer understanding of the work required before the appointment.
U005	Make Payment	Customers can pay for the services directly through the system by clicking on the pay button upon filling up the booking form.
U006	Generate Invoice	The system will automatically generate a booking invoice upon payment completion. The generated invoice will then be sent directly to the customer's email as a booking confirmation.
U007	Chat with the Service Provider/Customer	A built-in messaging system allows direct communication between customers and service providers to discuss service details. The chat box enables both customers and service providers to upload photos, making it easier to share visual information about service requirements for better understanding and clarity.
U008	View Booking History	Customers, service providers and admin can access a detailed log of their previous and ongoing bookings, including service dates, provider details and transaction information.

U009	Submit Review & Rating	After a service is completed, customers can provide feedback by rating the service provider and writing reviews.
U010	Modify Profile & Service Listings	Service providers can update their profile details or modify their service offerings at any time upon logging into the system.
U011	Track Earnings	Service providers and admin can monitor the income generated from completed services through a dedicated earnings tracking feature. This functionality includes a detailed breakdown of total earnings, with calculated commissions displayed separately for each completed booking.
U012	Submit Additional Cost	Service Providers can add any extra charges incurred during a job, such as material costs and extended service hours or days.
U013	Generate Form	Upon the service provider submitting additional cost information, the system will automatically generate a form detailing the updated costs. This form will then be sent directly to the client's email.
U014	View & Respond to Reviews	Service Providers can review customer feedback and respond appropriately, either to thank customers or address concerns.
U015	Verify Service Providers	Admin reviews and verify the credentials submitted by service providers to ensure only qualified and trustworthy providers are listed on HandyHub.

3.2.3 Use Case Description

Table 5: Use case description for Register Account

No		Section	Description
ID	**1.1**	**Identifier**	U001
	1.2	**Name**	Register Account
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Customers, Service Providers
Use Case Definition	**4.1**	**Short Description**	New users, whether customers or service providers, can create an account on HandyHub by providing their personal information, email, password, and contact details. Service providers need to provide additional information for verifications.
	4.2	**Primary Actors**	Customers, Service Providers
	4.3	**Other Actors**	Database
	4.4	**Pre-Conditions**	• The user does not have an existing account linked to the email. • The device of the user has an internet connection.
	4.5	**Trigger**	The user taps the “Register” button.
	4.6	**Post-Conditions**	The user creates a new account successfully.
	4.7	**Results**	The login page is displayed.
	4.8	**Main Scenario**	**Customers:** 1. The customer enters their name. 2. The customer enters their email. 3. The customer enters the password and password confirmation. 4. Customers enter their contact details. 5. Customers click on the “Register” button. 6. The service authenticates the credentials. 7. The login page is displayed.

			Service Provider: 1. The service provider selects "Register as a Service Provider." 2. The system displays a registration form requiring the following: a. Name or company name. b. Email address and phone number. c. Password and password confirmation. d. Service category e. Pricing f. List of services offered with details g. Identification proof h. Business Registration Proof 3. The service provider fills out all required fields and attaches the required documents. 4. The service provider clicks on the “Register” button. 5. The service provider needs to wait till the admin approves their registration. 6. The verification status will be sent to the service provider’s email.
	4.9	**Exception Scenario**	Invalid Email Format: • The user enters an email that is not in the correct format. • Display an error message: "Please enter a valid email address." Password Mismatch: • The user enters a password and confirmation that does not match. • Display an error message: "Passwords do not match. Please try again." Missing Required Fields: • The user leaves one or more required fields (e.g., name, email) empty. • Display an error message: "All fields are required. Please complete the form." Duplicate Email: • The user attempts to register with an email that is already in use. • Display an error message: "This email is already registered. Please log in or use a different email." Admin Rejection: • The admin does not approve the service provider due to incomplete information.

			• Notify the service provider via email: "Your registration could not be approved. Please review and resubmit the required information."

Table 6: Use case description for Login Account

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U002
	1.2	**Name**	Login Account
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Customers, Service Providers
Use Case Definition	**4.1**	**Short Description**	Registered users can securely log in to their HandyHub accounts using their credentials (email and password).
	4.2	**Primary Actors**	Customers, Service Providers
	4.3	**Other Actors**	Database
	4.4	**Pre-Conditions**	• The user has an existing account. • The device of the user has internet connection.
	4.5	**Trigger**	The user taps the “Login” button.
	4.6	**Post-Conditions**	The user logs into their own account successfully.
	4.7	**Results**	The home page is displayed.
	4.8	**Main Scenario**	1. The user enters their email. 2. The user enters their password. 3. The user taps the “Login” button. 4. The system authenticates the credentials. 5. The system allows the user to access the application.
	4.9	**Exception Scenario**	Invalid Email: • The user enters an invalid email.

			• Display an error message: “Account does not exist.” Invalid Password: • The user inputs an invalid password. • Display an error message: “Wrong password. Try again or click Forgot password to reset it.”

Table 7: Use case description for Search Service Provider

No		Section	Description
ID	**1.1**	**Identifier**	U003
	1.2	**Name**	Search Service Provider
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Customers
Use Case Definition	**4.1**	**Short Description**	Customers can search for service providers based on the specific service category they need.
	4.2	**Primary Actors**	Customers
	4.3	**Other Actors**	Database
	4.4	**Pre-Conditions**	• The customer is registered and logged into the HandyHub system. • Internet connectivity is active to perform the search.
	4.5	**Trigger**	The customer initiates the search by selecting a service category
	4.6	**Post-Conditions**	The system displays a list of service providers matching the search criteria.
	4.7	**Results**	Customers can view and compare multiple service providers of the selected category.
	4.8	**Main Scenario**	1. The customer navigates to the " Service Category" section.

			2. The customer selects a service category. 3. Service providers that match the service category displayed. 4. The customer clicks on a specific service provider to view their profile.
	4.9	**Exception Scenario**	No Matching Service Providers Found: • The system cannot find any service providers that match the search criteria. • Display a message: "No service providers found for your search. Please try a different category or broaden your criteria." Network Connectivity Issue: • The customer's internet connection is disrupted during the search process. • Display a message: "Network error. Please check your internet connection and try again."

Table 8: Use case description for Book Service Provider

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U004
	1.2	**Name**	Book Service Provider
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Customers
Use Case Definition	**4.1**	**Short Description**	Customers can schedule appointments by selecting a service provider, specifying their requirements, and choosing an available time slot in the booking form. Customers are required to upload a picture of the problem or task to provide service providers with a clearer understanding of the work required before the appointment.
	4.2	**Primary Actors**	Customers
	4.3	**Other Actors**	Database

	4.4	**Pre-Conditions**	• The customer is registered and logged into the HandyHub system. • The customer has selected a service provider from the provider list. • The customer has access to a device with a camera or image upload capability.
	4.5	**Trigger**	The customer clicks on the "Book Now" button on a service provider's profile page.
	4.6	**Post-Conditions**	• Customers will be able to submit the booking form and proceed to the payment page.
	4.7	**Results**	The payment page will be displayed.
	4.8	**Main Scenario**	1. On the service provider's profile page, the customer clicks on "Book Now." 2. The system displays the booking form, where the customer: a. Describes the issue b. Enters their address c. Selects an available time and date slot from the provider’s schedule. d. Uploads a picture of the problem or task to be addressed. 3. The customer reviews the entered details and clicks the "Submit & Pay" button. 4. The payment page will be displayed.
	4.9	**Exception Scenario**	Incomplete Booking Form: • The customer submits the form without specifying requirements, selecting a time slot, or uploading an image. • Display an error message: "Please complete all required fields, including uploading a picture, to proceed." No Available Time Slots: • The customer selects a service provider with no available time slots. • Display a message: "No available time slots. Please choose a different date or service provider." Invalid Image Upload: • The customer uploads an unsupported file type or exceeds the image size limit (<5MB). • Display an error message: "Please upload a valid image file (e.g., JPEG, PNG) within the size limit of 5MB."

Table 9: Use case description for Make Payment

No		Section	Description
ID	**1.1**	**Identifier**	U005
	1.2	**Name**	Make Payment
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Customers
Use Case Definition	**4.1**	**Short Description**	Customers can pay for the services directly through the system by clicking on the pay button upon filling up the booking form.
	4.2	**Primary Actors**	Customers
	4.3	**Other Actors**	Payment Gateway
	4.4	**Pre-Conditions**	• The customer has completed the booking form with all the required details. • The customer has a valid payment method (e.g., credit/debit card, online banking).
	4.5	**Trigger**	The customer clicks the "Submit & Pay" button upon filling up the booking form.
	4.6	**Post-Conditions**	• The payment transaction is successfully processed and recorded in the system. • Payment receipt and booking confirmation are sent to the customer's email.
	4.7	**Results**	The customer has completed the payment for the booking.
	4.8	**Main Scenario**	1. After reviewing and submitting the booking form, the system redirects the customer to a secure payment gateway. 2. The customer enters their payment details. 3. The system validates the payment details and processes the transaction. 4. Upon successful payment, the customer is redirected to a page displaying the booking details and receipt.

	4.9	**Exception Scenario**	Incomplete Payment Details: • The customer submits the payment form without completing all fields. • Display an error message: "Please enter all required payment details to proceed." Payment Declined: • The payment is declined due to insufficient funds, expired cards, or other issues. • Display a message: "Payment declined. Please verify your payment method or use a different one." Duplicate Payment Attempt: • The customer accidentally initiates the payment process twice for the same booking. • Display a message: "Payment for this booking has already been completed."

Table 10: Use case description for Generate Invoice

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U006
	1.2	**Name**	Generate Invoice
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Customers
Use Case Definition	**4.1**	**Short Description**	Upon a customer completing payment for the services, the system will automatically generate a booking invoice. The generated invoice will then be sent directly to the customer's email as a booking confirmation.
	4.2	**Primary Actors**	Customers
	4.3	**Other Actors**	Mail Service Provider
	4.4	**Pre-Conditions**	• The customer has completed the payment for a booked service.

	4.5	**Trigger**	The payment transaction is completed successfully.
	4.6	**Post-Conditions**	• A detailed invoice is generated and displayed to the customer as a reference. • The invoice is sent to the customer's email.
	4.7	**Results**	The customer receives a booking invoice and confirmation via email.
	4.8	**Main Scenario**	1. Upon successful payment transaction, the system generates an invoice containing: a. Booking ID b. Customer and service provider details c. Date, time, and service category d. Total payment amount 2. The invoice is sent to the customer's registered email address. 3. The customer is redirected to a confirmation page showing the booking details.
	4.9	**Exception Scenario**	Failed Payment: • The payment fails, and no invoice is generated. • Display Message: "Payment was unsuccessful. Please retry to complete your booking."

Table 11: Use case description for Chat with the Service Provider/Customer

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U007
	1.2	**Name**	Chat with the Service Provider/Customer
Management	**2.1**	**Priority**	Moderate
	2.2	**Criticality**	Moderate
Context	**3.1**	**Sources**	Customers, Service Providers
Use Case Definition	**4.1**	**Short Description**	A built-in messaging system allows direct communication between customers and service providers to discuss service details.
	4.2	**Primary Actors**	Customers, Service Providers

	4.3	**Other Actors**	-
	4.4	**Pre-Conditions**	• Both the customer and service provider have active accounts on the platform.
	4.5	**Trigger**	A customer initiates a conversation to the service provider by clicking on the message icon.
	4.6	**Post-Conditions**	• Messages and photos are exchanged between the customer and the service provider via the chat box.
	4.7	**Results**	A clear and documented exchange of information between the customer and service provider.
	4.8	**Main Scenario**	1. The customer opens the chat box by clicking on the message icon and sends an initial message to the service provider. 2. The service provider receives a notification and replies to the message. 3. Both parties engage in a conversation to discuss service details, such as specific requirements, timelines, and pricing. 4. The chat system securely stores all messages and photos exchanged.
	4.9	**Exception Scenario**	Message Delivery Failure: • A message fails to send due to a network issue. • Display an error: "Message failed to send. Please check your internet connection and try again." Unsupported File Format or Size: • A user tries to upload a file that exceeds the maximum size limit or is in an unsupported format. • Display an error: "File upload failed. Please ensure the file is in a supported format and within the size limit (e.g., JPEG, PNG, max 5 MB)." Photo Upload Error: • The photo fails to upload due to a server issue. • Notify the user: "Photo upload failed due to a server issue. Please try again later."

Table 12: Use case description for View Booking History

No		Section	Description
ID	**1.1**	**Identifier**	U008
	1.2	**Name**	View Booking History
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Customers, Service Providers, Admin
Use Case Definition	**4.1**	**Short Description**	Customers, Service Providers, and admin can access a detailed log of their previous and ongoing bookings, including service dates, provider details, and transaction information.
	4.2	**Primary Actors**	Customers, Service Providers, Admin
	4.3	**Other Actors**	Database
	4.4	**Pre-Conditions**	• Users must have an active account on HandyHub. • Users must have completed at least one booking or have ongoing bookings in the system.
	4.5	**Trigger**	The user selects the "Booking" option from their dashboard.
	4.6	**Post-Conditions**	• The system displays the user's booking history in a clear and organized format. • Users can review details of previous and ongoing bookings, including service dates, provider or customer information, and payment records.
	4.7	**Results**	Users have a transparent view of their booking history.
	4.8	**Main Scenario**	1. The user navigates to their dashboard and clicks on the "Booking" option. 2. The system retrieves all past and ongoing bookings. 3. The booking history is displayed, showing: a. Service date and time.

			b. Service provider or customer details (name, contact information). c. Service description. d. Payment details, including amount and status. e. Booking status (completed, ongoing, or cancelled).
	4.9	**Exception Scenario**	No Booking History Available: • A new user without any past or ongoing bookings accesses the "Booking History" section. • Display a message: "No bookings found. Start by scheduling your first service!"

Table 13: Use case description for Submit Review & Rating

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U009
	1.2	**Name**	Submit Review & Rating
Management	**2.1**	**Priority**	Moderate
	2.2	**Criticality**	Moderate
Context	**3.1**	**Sources**	Customers
Use Case Definition	**4.1**	**Short Description**	After a service is completed, customers can provide feedback by rating the service provider and writing reviews.
	4.2	**Primary Actors**	Customers
	4.3	**Other Actors**	-
	4.4	**Pre-Conditions**	• The service must be marked as completed in the system. • The customer must have an active account on HandyHub.
	4.5	**Trigger**	The customer clicks the "Submit Review & Rating" button on their booking history page for completed services.

	4.6	**Post-Conditions**	• The review and rating are saved in the system and displayed on the service provider's profile.
	4.7	**Results**	The review and rating are displayed on the service provider’s profile.
	4.8	**Main Scenario**	1. The customer navigates to their booking history page. 2. The customer clicks the "Submit Review & Rating" button for the completed service. 3. The system displays a form where the customer: a. Selects a rating (e.g., 1–5 stars). b. Writes a review detailing their experience. 4. The customer submits the form by clicking the "Submit" button.
	4.9	**Exception Scenario**	Empty Feedback Form: • The customer submits the form without providing a rating or review. • Display an error: "Please provide both a rating and a review before submitting."

Table 14: Use case description for Modify Profile & Service Listings

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U010
	1.2	**Name**	Modify Profile & Service Listings
Management	**2.1**	**Priority**	Moderate
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Service Providers
Use Case Definition	**4.1**	**Short Description**	Service providers can update their profile details or modify their service offerings at any time upon logging into the system.
	4.2	**Primary Actors**	Service Providers
	4.3	**Other Actors**	Database

	4.4	**Pre-Conditions**	• The service provider must have an active account on HandyHub. • The service provider must successfully log in to their account.
	4.5	**Trigger**	The service provider navigates to the "Profile" section after logging into their account and selects the "Edit" option.
	4.6	**Post-Conditions**	• Updated profile details and service offerings are saved and reflected on the platform.
	4.7	**Results**	The login page is displayed.
	4.8	**Main Scenario**	1. The service provider navigates to the "Profile" section. 2. The service provider selects the “Edit” button. 3. The service provider makes the necessary changes, such as: a. Updating name, email, or phone number. b. Adding or removing services offered. c. Change pricing. 4. The service provider clicks the "Save" button to confirm the updates. 5. The system validates the changes and saves the updated information. 6. A confirmation message is displayed: "Your profile and services have been updated successfully."
	4.9	**Exception Scenario**	Incomplete Information: • The service provider leaves required fields blank while editing. • Display an error: "Please complete all required fields before saving changes."

Table 15: Use case description for Track Earnings

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U011
	1.2	**Name**	Track Earnings

Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Service Providers, Admin
Use Case Definition	**4.1**	**Short Description**	Service providers and admin can monitor the income generated from completed services through a dedicated earnings tracking feature.
	4.2	**Primary Actors**	Service Providers, Admin
	4.3	**Other Actors**	Payment Gateway
	4.4	**Pre-Conditions**	• Service providers and admins must be logged into the system. • Completed bookings with payment transactions must exist in the system.
	4.5	**Trigger**	The service provider or administrator navigates to the "Earnings" section in the system dashboard.
	4.6	**Post-Conditions**	• Service providers and admin can view a detailed breakdown of earnings, including total revenue and platform commissions.
	4.7	**Results**	• Service providers have clear visibility into their earnings and platform fees. • Admin can monitor platform revenue from commissions.
	4.8	**Main Scenario**	1. Service providers and admin click on the "Earnings" section in their respective dashboard menu. 2. The system retrieves and displays a detailed list of individual booking transactions, including: a. Service fee charged to the customer. b. Platform commission amount. c. Net earnings credited to the service provider.
	4.9	**Exception Scenario**	No Completed Transactions: • The user views the earnings section, but no completed bookings exist. • Display a message: "No earnings data available at this time."

Table 16: Use case description for Submit Additional Cost

No		Section	Description
ID	**1.1**	**Identifier**	U012
	1.2	**Name**	Submit Additional Cost
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Service Providers
Use Case Definition	**4.1**	**Short Description**	Service Providers can add any extra charges incurred during a job, such as material costs and extended service hours or days.
	4.2	**Primary Actors**	Service Providers
	4.3	**Other Actors**	-
	4.4	**Pre-Conditions**	• The service provider must be logged into the system. • A valid service booking must exist. • The service booking must be in progress.
	4.5	**Trigger**	The service provider navigates to the “Booking” section and selects a specific booking that is in progress. The provider clicks on "Add Additional Cost" in the booking details view.
	4.6	**Post-Conditions**	A form summarizing the additional charges is generated, including the entered details and the total cost.
	4.7	**Results**	The customer receives a clear breakdown of the additional costs via email.
	4.8	**Main Scenario**	1. The provider navigates to the "Bookings" section and selects a specific booking. 2. The provider clicks on "Add Additional Cost" in the booking details view. 3. The system displays a form where the provider enters details such as:

			a. Type of additional cost (e.g., materials, extra hours). b. Cost Description with the total amount c. Bank Details 4. The provider reviews the entered information and submits the form.
	4.9	**Exception Scenario**	Missing Details: • The provider omits mandatory fields in the additional cost form. • Display an error message: "Please fill in all required fields before submitting."

Table 17: Use case description for Generate Form

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U013
	1.2	**Name**	Generate Form
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High
Context	**3.1**	**Sources**	Service Providers
Use Case Definition	**4.1**	**Short Description**	Upon the service provider submitting additional cost information, the system will automatically generate a form detailing the updated costs. This form will then be sent directly to the client's email.
	4.2	**Primary Actors**	Service Providers
	4.3	**Other Actors**	Mail Service Provider
	4.4	**Pre-Conditions**	• The service provider has completed the service. • Additional cost information for a booking has been provided by the service provider.
	4.5	**Trigger**	The service provider submits the additional cost information via the booking details interface.

	4.6	**Post-Conditions**	• A detailed form consisting of the additional cost is generated by the system. • The form is sent to the customer emails who booked the service.
	4.7	**Results**	The customer receives an email of the additional cost.
	4.8	**Main Scenario**	1. The service provider submits the additional cost form. 2. The system processes the data and generates a detailed form with: a. Type of additional cost (e.g., materials, extra hours). b. Cost Description with total amount c. Bank Details 3. The system sends the generated form to the customer's email.
	4.9	**Exception Scenario**	Missing Details: • The provider omits mandatory fields in the additional cost form. • Display an error message: "Please fill in all required fields before submitting."

Table 18: Use case description for View & Respond to Reviews

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U014
	1.2	**Name**	View & Respond to Reviews
Management	**2.1**	**Priority**	Low
	2.2	**Criticality**	Moderate
Context	**3.1**	**Sources**	Service Providers
Use Case Definition	**4.1**	**Short Description**	Service Providers can review customer feedback and respond appropriately, either to thank customers or address concerns.
	4.2	**Primary Actors**	Service Providers
	4.3	**Other Actors**	-

	4.4	**Pre-Conditions**	• There are completed bookings with associated reviews provided by customers.
	4.5	**Trigger**	The service provider navigates to the "Reviews" section to check the feedback for a specific booking or service.
	4.6	**Post-Conditions**	• Service providers have visibility into the reviews and ratings given by customers. • Service providers will be able to respond to the reviews and ratings given by the customers.
	4.7	**Results**	Service providers can reply directly to customer reviews, expressing gratitude or addressing concerns.
	4.8	**Main Scenario**	1. The service provider navigates to the "Reviews" section. 2. The provider selects a specific booking to check the reviews submitted by customers. 3. The system displays relevant customer feedback. 4. The service provider clicks the "Respond" button. 5. A text input field is displayed, where the provider types their response. 6. The provider submits their response.
	4.9	**Exception Scenario**	Empty Feedback Form: • The service provider submits the form without providing the review. • Display an error: "Please provide a review before submitting."

Table 19: Use case description for Verify Service Providers

No		**Section**	**Description**
ID	**1.1**	**Identifier**	U015
	1.2	**Name**	Verify Service Providers
Management	**2.1**	**Priority**	High
	2.2	**Criticality**	High

Context	**3.1**	**Sources**	Admin
Use Case Definition	**4.1**	**Short Description**	Admin reviews and validates the credentials submitted by service providers to ensure only qualified and trustworthy providers are listed on HandyHub.
	4.2	**Primary Actors**	Admin
	4.3	**Other Actors**	Database, Mail Service Provider
	4.4	**Pre-Conditions**	• The service provider has submitted an account registration request, including personal information and necessary verification documents.
	4.5	**Trigger**	The admin accesses the "Service Providers" section in the admin panel.
	4.6	**Post-Conditions**	• The system updates the service provider's verification status as either **Approved** or **Rejected**.
	4.7	**Results**	The admin successfully reviews and validates service provider profiles and information.
	4.8	**Main Scenario**	1. The admin navigates to the "Service Provider" section. 2. The system displays a list of pending service provider registration requests along with their submitted profiles and documents. 3. The administrator selects a service provider to review their registration details and supporting documents: a. Identification proof b. Business Registration Proof 4. The admin evaluates the submitted information. 5. The administrator clicks on the "Approve" button to approve the registration request or selects "Reject" if the information is insufficient or incorrect. 6. The system updates the provider's verification status and sends a notification email to the service provider.
	4.9	**Exception Scenario**	Incomplete Documentation: • The service provider's submission contains missing documents.

			• Requests the service provider to upload the necessary documents and resubmit their application.

4. Requirement Traceability Matrix

Table 20: Requirement Traceability Matrix

Requirement ID	Requirement	Module	Priority	Development Status	Test Case ID	Test Case Description	Test Case Status	Test Case Result
U001	Register Account	Admin and User Account Management	High	Completed	TC001	Verify successful user registration flow	Completed	Pass
U002	Login Account	Admin and User Account Management	High	Completed	TC002	Verify secure login functionality	Completed	Pass
U003	Search Service Provider	Booking	High	Completed	TC003	Validate service provider search by category	Completed	Pass
U004	Book Service Provider	Booking	High	Completed	TC004	Verify booking form and appointment scheduling	Completed	Pass
U005	Make Payment	Payment	High	Completed	TC005	Test integrated payment processing	Completed	Pass
U006	Generate Invoice	Payment	High	Completed	TC006	Validate automatic invoice	Completed	Pass

						generation and email delivery		
U007	Chat with the Service Provider/Customer	Communication	Moderate	Completed	TC007	Ensure seamless in-app messaging and image uploading	Completed	Pass
U008	View Booking History	Booking	High	Completed	TC008	Verify booking history visibility for all users	Completed	Pass
U009	Submit Review & Rating	Review & Rating	Moderate	Completed	TC009	Test customer review and rating submission flow	Completed	Pass
U010	Modify Profile & Service Listings	Service Provider	Moderate	Completed	TC010	Validate service provider profile and service updates	Completed	Pass
U011	Track Earnings	Service Provider, Admin and User Management	High	Completed	TC011	Ensure accurate earnings tracking and reporting	Completed	Pass
U012	Submit Additional Cost	Service Provider	High	Completed	TC012	Verify additional cost submission and notification	Completed	Pass
U013	Generate Form	Service Provider	High	Completed	TC013	Test automatic form generation and email delivery	Completed	Pass

U014	View & Respond to Reviews	Service Provider	Low	Completed	TC014	Validate service provider review response functionality	Completed	Pass
U015	Verify Service Providers	Admin and User Management	High	Completed	TC015	Ensure admin can effectively review and approve providers	Completed	Pass

5. Software Metrics

5.1 Product metrics

Software product metrics focus on measuring the features and characteristics from various perspectives, focusing mainly on the source code of the product.

Table 21: Product Metrics

Metrics	**Description**
Size and design	• Lines of code across six core modules (Admin, Service Provider, Booking, Payment, Review, Communication). • The number of reusable Bootstrap CSS framework components used in UI elements (service cards, forms, review displays). • Count of API endpoints and database tables for core functionalities.
Complexity involved	• Integration of multiple technology layers (Bootstrap CSS frontend, Express.js backend, PostgreSQL database). • External service integrations (Stripe payment, email notifications, real-time chat).
Performance	• Loading time for service provider listings and profiles. • Response time for booking confirmations. • Payment processing speed via Stripe. • Image upload and retrieval speed for problem documentation. • Chat system response time.
Usability	• Measuring success rates for completing key tasks like service provider registration, booking creation, and payment processing.

5.2 Project metrics

Software project metrics are normally used by project managers to estimate the resources used in the development such as time effort, development costs, and so on to minimize unforeseen circumstances and mitigate risks.

Table 22: Project Metrics

Metrics	**Description**
Development cost and resources	• Track expenditure on technical infrastructure: Heroku cloud platform for application deployment, PostgreSQL database hosting, Stripe payment gateway integration costs. • HandyHub is deployed on Heroku, a cloud platform that hosts application deployments. The cost is derived from the Dyno hours used. Dynos are virtualized Linux containers used by Heroku to execute the deployed code.
Time effort	• Total person-hours spent on requirements analysis, software design, coding and implementation, testing, deployment, and maintenance phases. • The development of HandyHub development runs in 2-week sprint cycles for a total of 34 weeks.
Deliverables	• The number of completed tasks per iteration compared to the list of prioritized tasks that should be completed for the iteration to calculate the progress percentage. • This is measured by comparing the sprint backlog to the number of tasks completed from the backlog for every sprint.

5.3 Process metrics

Software process metrics measure the development process that takes place to create the software. This is measured for continuous process improvement which would lead to the improvement of product quality in the long run.

Table 23: Process Metrics

Metrics	Description
Time to locate the defect	• Measures time taken from when a software defect (bug) is first reported to the point when the root cause is fully analysed and understood. Lengthy identification delays fixing. • HandyHub development primarily uses white box testing to locate bugs by examining the source code.
Defect resolving time	• Tracks total elapsed time from the root cause of the defect being identified to successful fixing and re-testing of the affected feature. This includes time taken for prioritization, specification, and development fixes. • HandyHub bug fixes are documented in a Google Doc for future reference in case similar issues arise again to ensure maintainability.
Response time	• Measures the efficiency and speed of the acknowledgment and successful fixing of the defect by the developer.

6. System Screen Shots

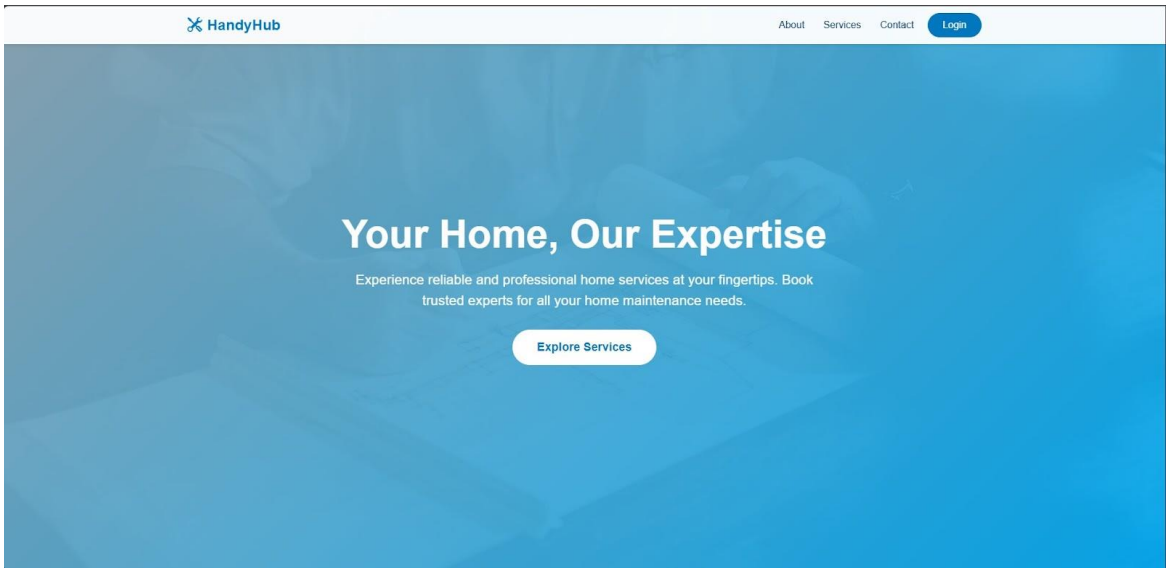

Figure 3: Landing Page

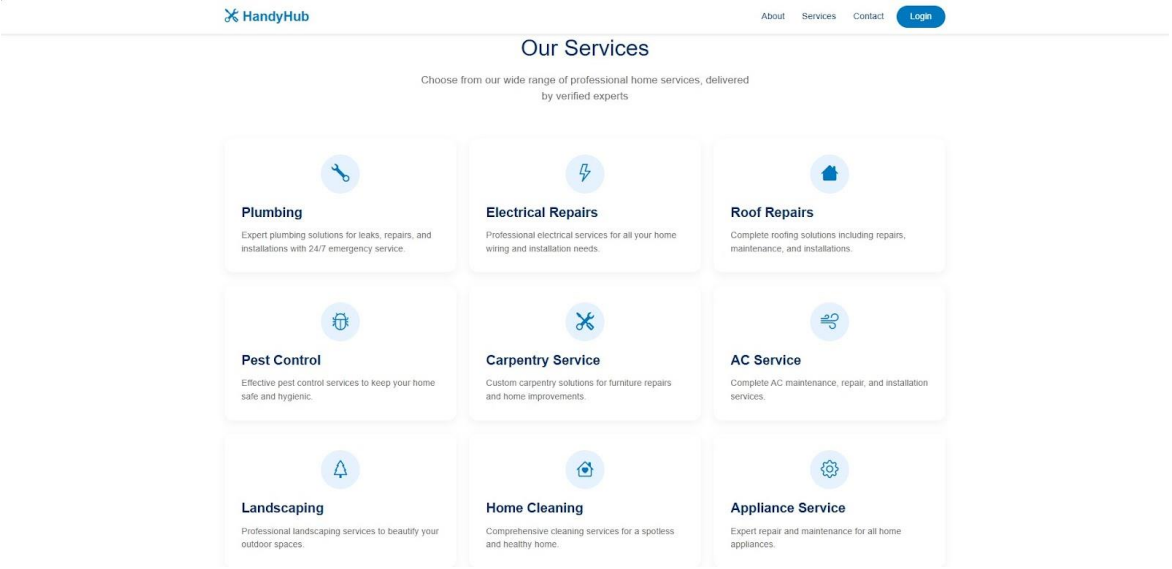

Figure 4: Service Category

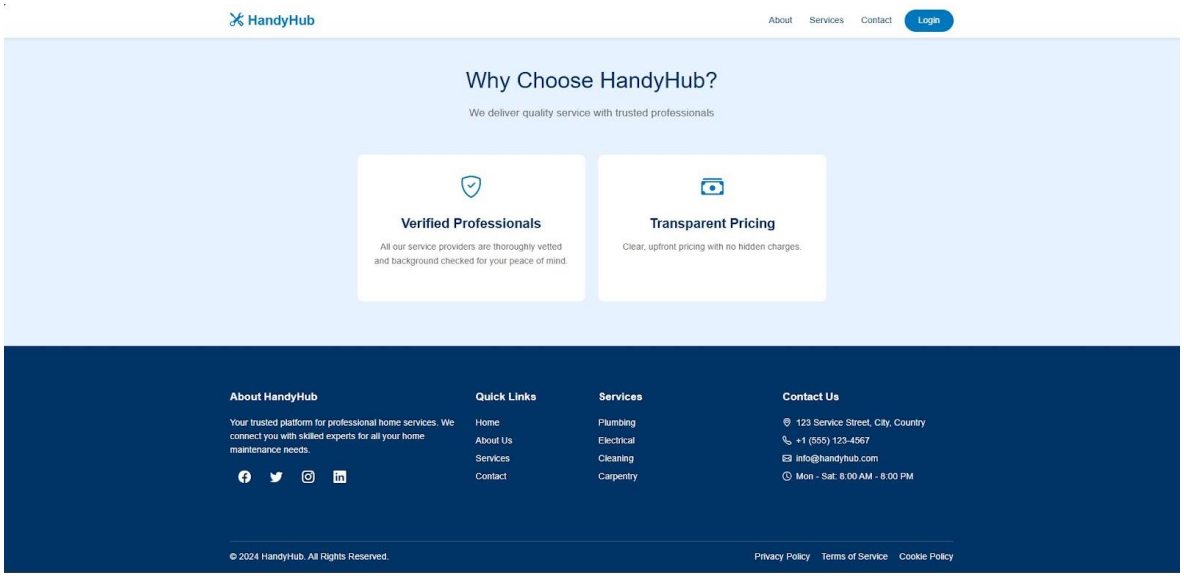

Figure 5: Footer

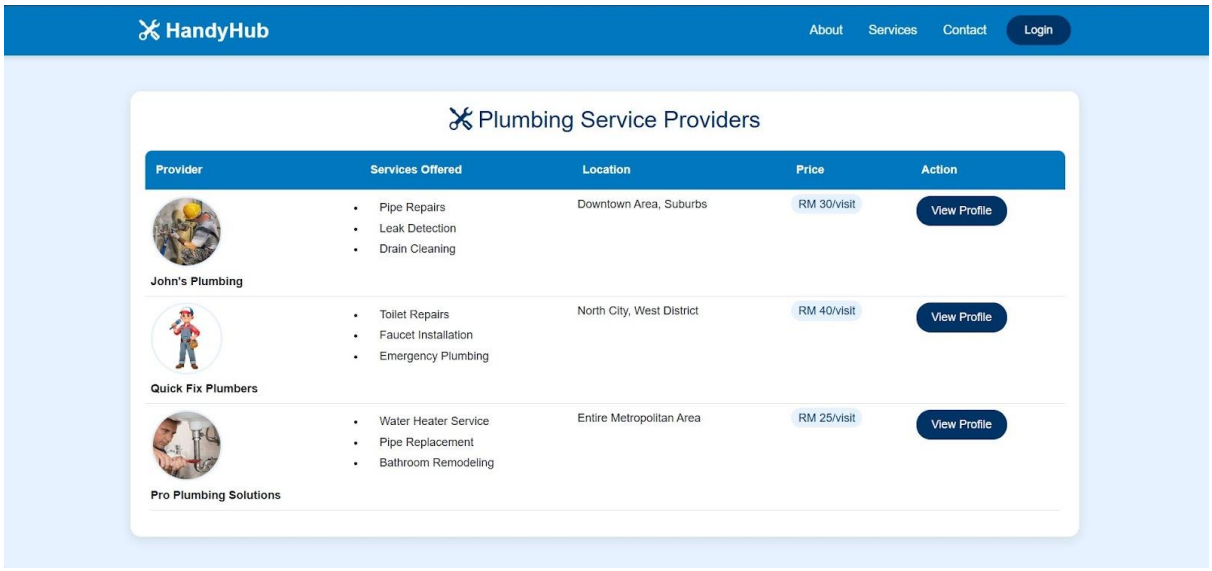

Figure 6: Service Provider List

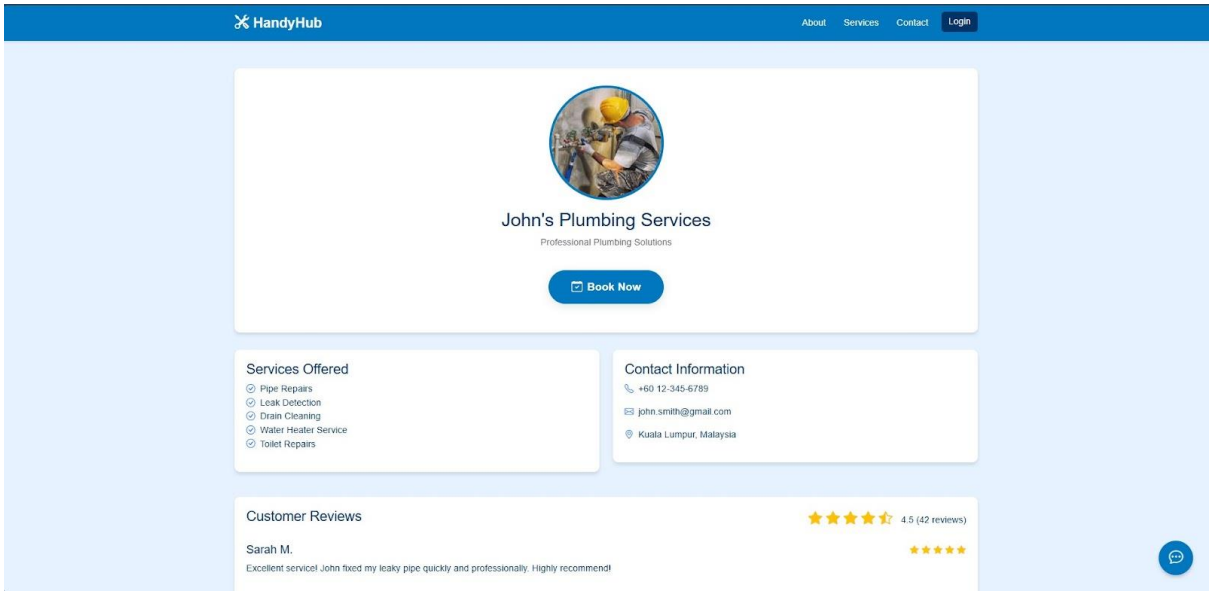

Figure 7: Service Provider Profile

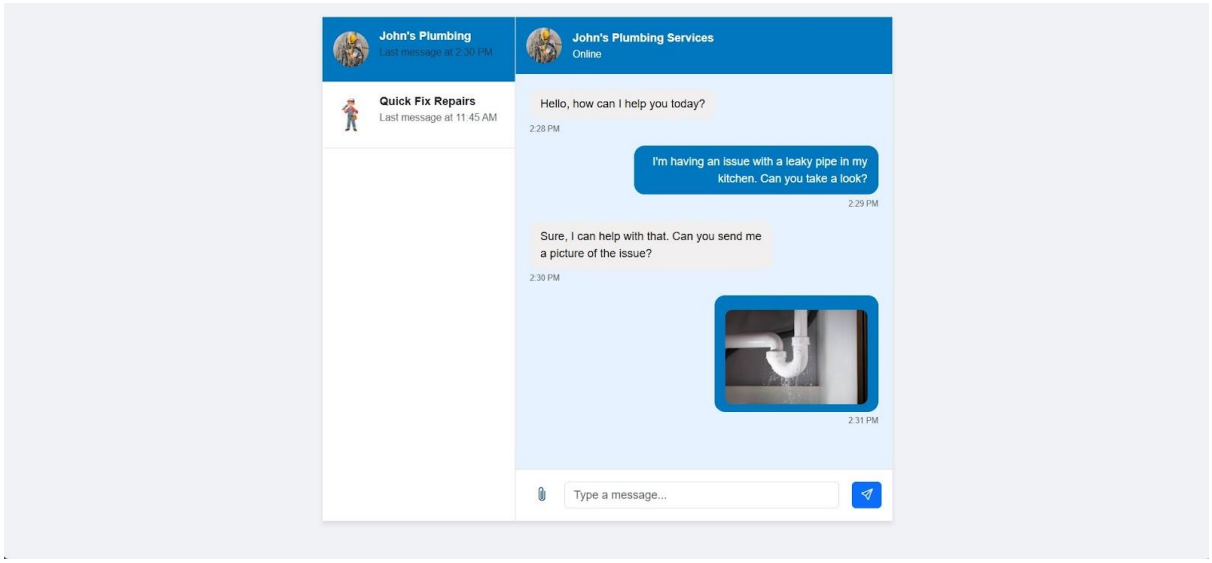

Figure 8: Chat box

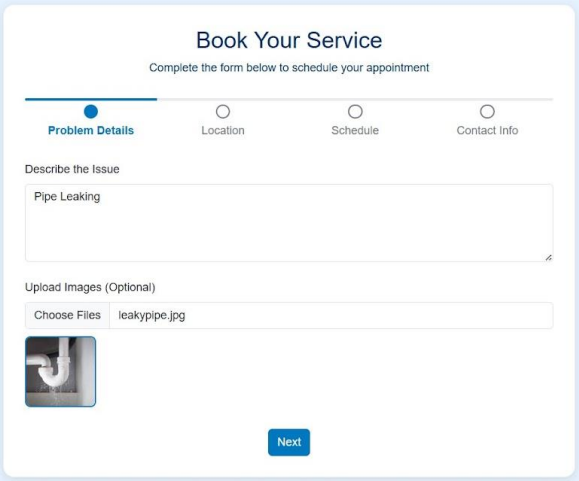

Figure 9: Booking Form (Problem Details)

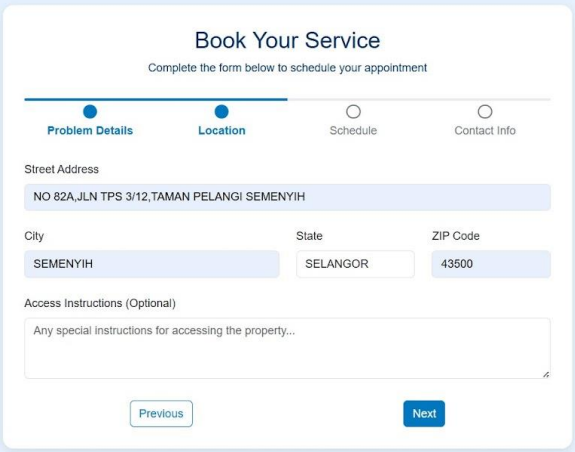

Figure 10: Booking Form (Location)

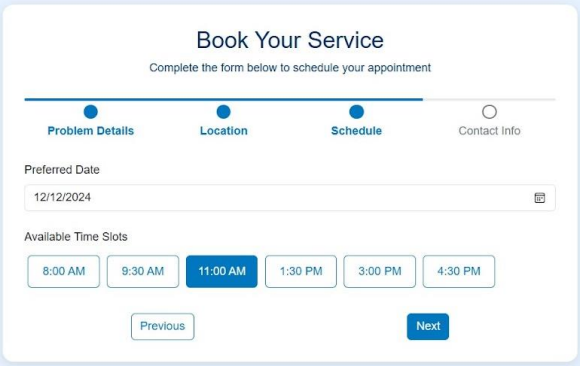

Figure 11: Booking Form (Schedule)

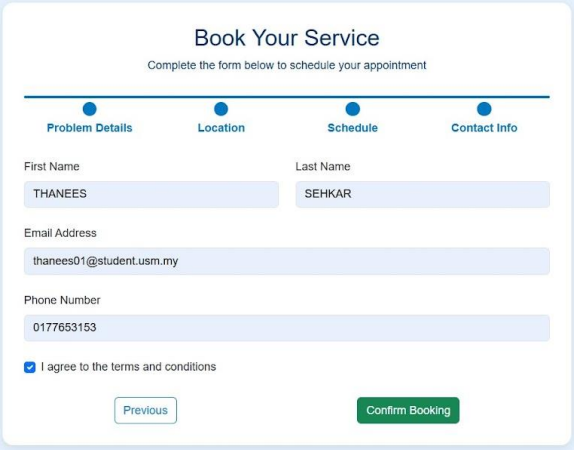

Figure 12: Booking Form (Contact Info)

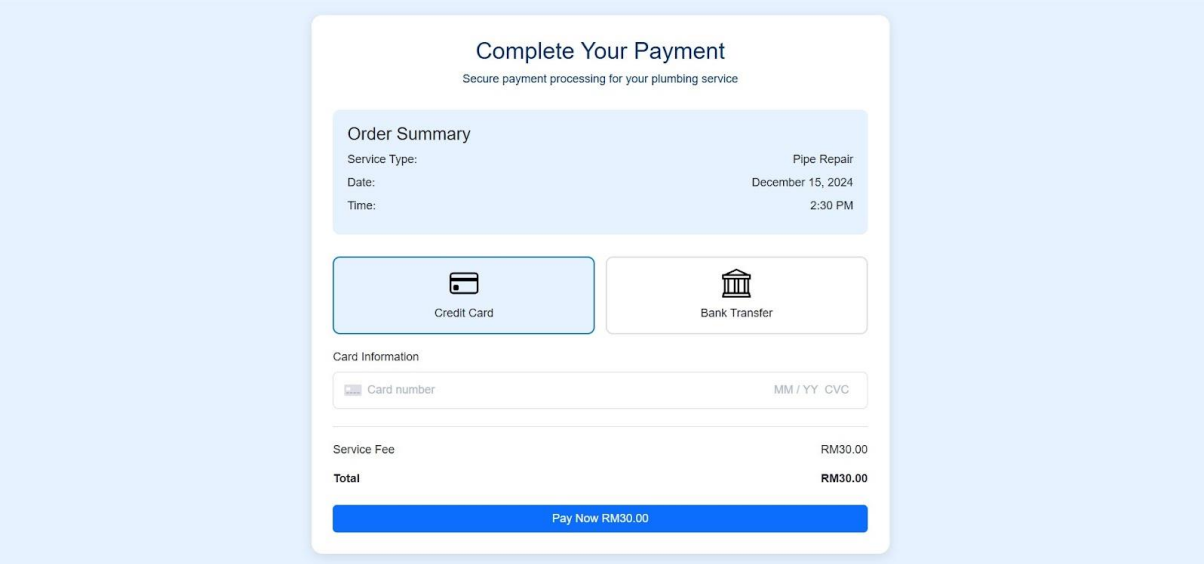

Figure 13: Payment Page

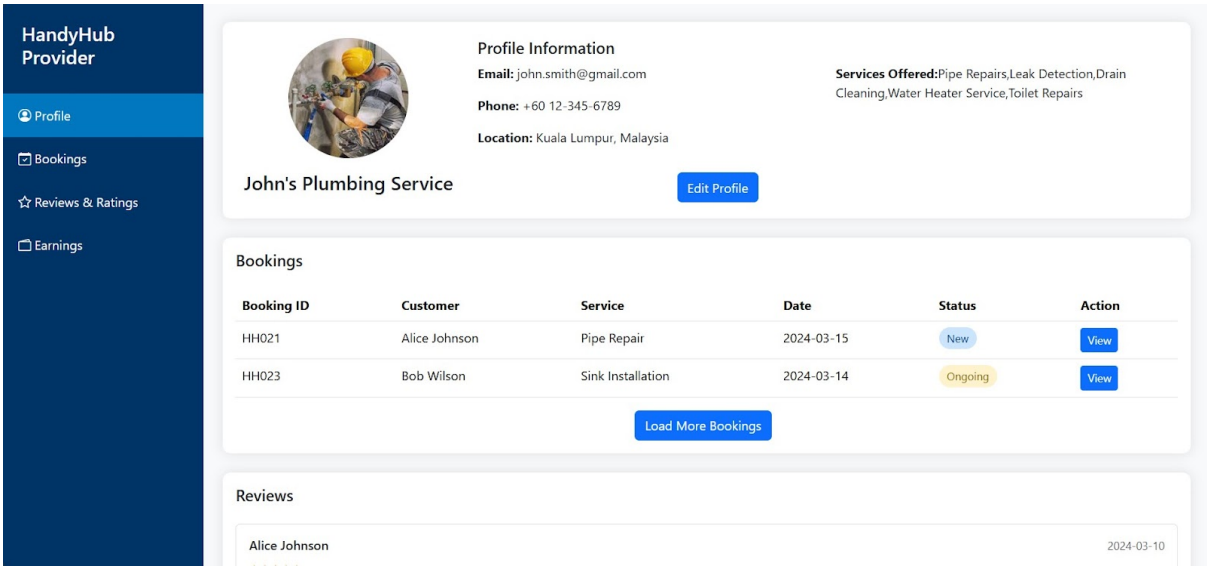

Figure 14: Service Provider Dashboard

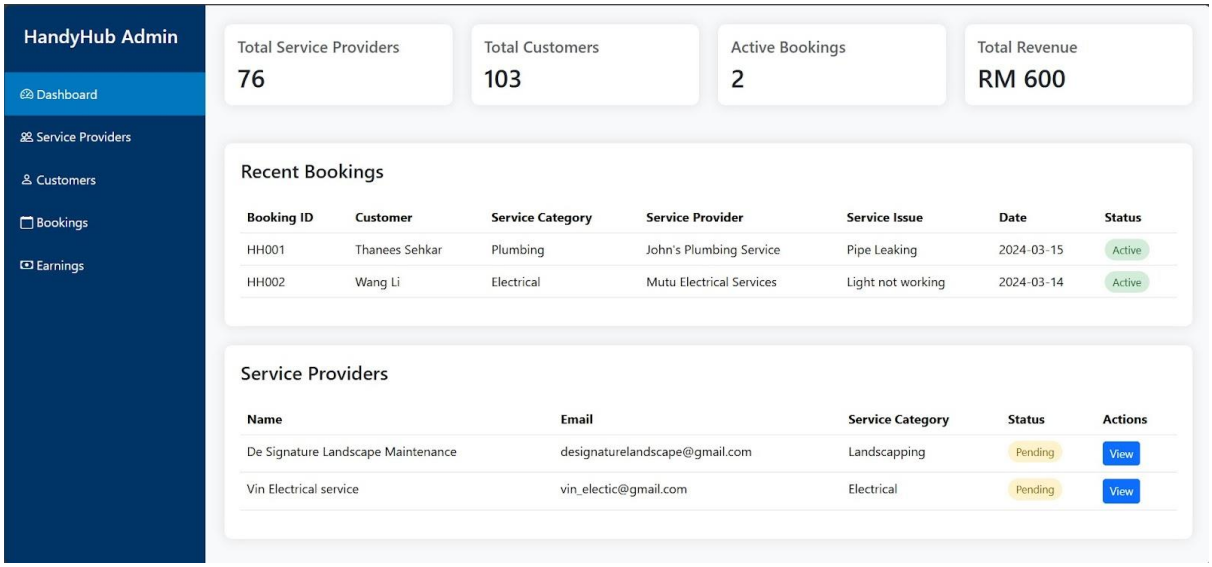

Figure 15: Admin Dashboard

7. Conclusion and future work

The HandyHub project has successfully addressed the key challenges faced by the home service industry, including fragmentation of information, lack of service provider verification, absence of real-time booking systems, pricing transparency issues, and complex payment handling. HandyHub has created a dependable and user-friendly platform for both customers and service providers by creating an integrated web application that centralises information about service providers, integrates secure payment processing, streamlines booking and scheduling, provides transparent pricing, and implements thorough verification procedures.

Through the deployment of HandyHub, customers can now easily find, book, and pay for home maintenance services with confidence, while service providers can streamline their operations, expand their client base, and maintain a positive reputation on the platform. The system's emphasis on transparency, security, and user-centric design has the potential to transform the home service industry, fostering economic growth and enhancing the overall quality of life for communities.

For future work, developing a mobile app for both customers and service providers could greatly improve accessibility and convenience, incorporating features like push notifications, location-based services, and mobile payment integration. Moreover, the HandyHub system can be integrated with smart home devices, which could enable automatic detection of maintenance needs and allow customers to monitor their home systems through the HandyHub platform. By pursuing these future development opportunities, HandyHub can continue evolving to better serve the home services industry and contribute to the well-being of communities and the environment.

8. References

Thinksys. (2024, September 5). *Agile Software Development Life cycle*. ThinkSys Inc. https://thinksys.com/development/agile-software-development-life-cycle/#:~:text=It%20uses%20an%20iterative%20approach,developing%2C%20and%20testing%2Fdeployment.

Sulemani, M. (n.d.). *What is a software process model? Top 7 models explained*. Educative. https://www.educative.io/blog/software-process-model-types

What is Use Case Diagram? (n.d.). https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/

GeeksforGeeks. (2024, October 22). *Functional vs. Non Functional Requirements*. GeeksforGeeks. https://www.geeksforgeeks.org/functional-vs-non-functional-requirements/

Good, L. (2024, June 5). *What is a Requirements Traceability Matrix (RTM)?* project-management.com. https://project-management.com/requirements-traceability-matrix-rtm/

Das, S. (2023, September 9). *What are Software Metrics in Software Engineering? How to Measure it ? | BrowserStack*. BrowserStack. https://www.browserstack.com/guide/what-is-software-metrics

Jessy, & Jessy. (2024, March 13). Guide to Home Maintenance: Resolving 10 Common Issues - UncleDefect.SG. *UncleDefect.SG - Singapore Number 1 HDB Condo Defects Specialist*. https://uncledefect.sg/guide-to-home-maintenance-resolving-10-common-issues/