# MY HEALTH SAFETY ENVIRONMENT USING WEB RESPONSIVE (MyHSE)

DEVESHWARAN A/L ARUMUGAM

Bachelor of Computer Science
(Software Engineering) with Honours

UNIVERSITI MALAYSIA PAHANG

**UNIVERSITI MALAYSIA PAHANG**

**DECLARATION OF THESIS AND COPYRIGHT**

Author's Full Name : Deveshwaran A/L Arumugam

Date of Birth

Title : My Health Safety Environment using Web Responsive (MyHSE)

Academic Session : Semester 2 2022/2023

I declare that this thesis is classified as:

| | | |
|---|---|---|
| ☐ | CONFIDENTIAL | (Contains confidential information under the Official Secret Act 1997)* |
| ☐ | RESTRICTED | (Contains restricted information as specified by the organization where research was done)* |
| ☑ | OPEN ACCESS | I agree that my thesis to be published as online open access (Full Text) |

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

_____
(Student's Signature)

New IC/Passport Number
Date:20/06/2023

_____
(Supervisor's Signature)

AZLINA BINTI ZAINUDDIN
Name of Supervisor
Date: 28 July 2023

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

## SUPERVISOR'S DECLARATION

I/~~We~~* hereby declare that I/~~We~~* have checked this thesis/~~project~~* and in my/~~our~~* opinion, this thesis/~~project~~* is adequate in terms of scope and quality for the award of the degree of *~~Doctor of Philosophy/ Master of Engineering/ Master of Science in~~ Bachelor of Computer Science in Software Engineering.

_____

(Supervisor's Signature)

Full Name : AZLINA BINTI ZAINUDDIN
LECTURER
FACULTY OF COMPUTING
COLLEGE OF COMPUTING & APPLIED SCIENCES
UNIVERSITI MALAYSIA PAHANG
26600 PEKAN, PAHANG DARUL MAKMUR
TEL : 09-424 4838 FAX : 09-424 4866

Position :

Date : 28 July 2023

-

_____

(Co-supervisor's Signature)

Full Name :

Position :

Date :

## STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

_____

(Student's Signature)

Full Name : DEVESHWARAN A/L ARUMUGAM

ID Number : CB20173

Date : 20 JUNE 2023

# MY HEALTH SAFETY ENVIRONMENT USING WEB RESPONSIVE (MyHSE)

DEVESHWARAN A/L ARUMUGAM

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Computer Science (Software Engineering) with Honours

Faculty of Computing
UNIVERSITI MALAYSIA PAHANG

JULY 2023

## ACKNOWLEDGEMENTS

First and foremost, I'd like to thank my supervisor, Ts. Azlina binti Zainuddin, for providing me with invaluable guidance and support throughout the entire thesis writing process. Her unwavering encouragement and constructive feedback were critical in shaping the project's final outcome.

I would also like to thank the faculty members of the software engineering department for providing me with a solid foundation in the field as well as the opportunity to pursue my academic interests.

I'd like to thank my friends and family for their love, support, and understanding during this difficult time. Their motivation and encouragement were a constant source of inspiration for me.

Last but not least, I'd like to thank all of the respondents who took part in this study and provided me with the necessary information and feedback. This thesis would not be possible without their invaluable assistance.

# ABSTRAK

Pembangunan My Health Safety Environment (MyHSE), sistem responsif berasaskan web yang dicipta untuk menangani isu penting dalam sektor keselamatan dan kesihatan pekerjaan (OSH), diterangkan dalam tesis ini. Sistem yang dicadangkan menyediakan platform terpusat untuk mengakses perkhidmatan OSH dalam usaha menangani isu seperti penipuan, prosedur pengesahan yang berlarutan dan kekurangan penyedia perkhidmatan yang boleh dipercayai. Matlamat utama MyHSE adalah untuk menawarkan platform yang boleh dipercayai dan mesra pengguna supaya penyedia perkhidmatan OSH boleh mempamerkan tawaran mereka dan pelanggan boleh mengakses dan menempah perkhidmatan ini dengan cepat. Sistem ini bertujuan untuk menjamin kesahihan penyedia perkhidmatan dengan menyepadukan mekanisme pengesahan orang yang kompeten yang menggunakan sijil dan nombor pendaftaran. Metodologi Pembangunan Aplikasi Rapid (RAD), yang menekankan prototaip berulang dan penglibatan pengguna, telah digunakan dalam pembangunan MyHSE. Pendekatan RAD menjadikannya lebih mudah untuk membuat prototaip dengan cepat dan memperbaikinya secara berulang sebagai tindak balas kepada maklum balas pengguna. Metodologi membenarkan semakan dan penyesuaian biasa, menjamin bahawa kefungsian dan reka bentuk sistem dengan tepat memenuhi keperluan perubahan kedua-dua penyedia perkhidmatan dan pelanggan. MyHSE menawarkan beberapa ciri penting yang akan meningkatkan pengalaman pelanggan dalam sektor OSH. Pembekal perkhidmatan OSH boleh memuat naik dan mengiklankan perkhidmatan yang mereka sediakan dengan cepat, lengkap dengan maklumat hubungan yang berkaitan dan spesifikasi perkhidmatan. Pelanggan, seterusnya, mempunyai akses kepada pelbagai perkhidmatan yang boleh dipercayai, menjadikan tempahan dan permintaan perkhidmatan lebih mudah. MyHSE menggunakan mekanisme pengesahan orang kompeten yang kukuh untuk memupuk kepercayaan dan kebergantungan. Pembekal perkhidmatan mesti melalui pengesahan, di mana kesahihan lesen dan nombor pendaftaran mereka diperiksa. Pelanggan yakin dengan proses pengesahan yang ketat ini bahawa mereka bekerja dengan profesional yang berkelayakan dan cekap.

## ABSTRACT

The development of My Health Safety Environment (MyHSE), a web-based responsive system created to address important issues in the occupational safety and health (OSH) sector, is described in this thesis. The proposed system provides a centralised platform for accessing OSH services in an effort to address issues like scams, drawn-out verification procedures, and a lack of dependable service providers. MyHSE's main goal is to offer a dependable and user-friendly platform so that OSH service providers can showcase their offerings and subscribers can quickly access and reserve these services. The system aims to guarantee the legitimacy of service providers by integrating a competent person verification mechanism that makes use of certificates and registration numbers. The Rapid Application Development (RAD) methodology, which emphasises iterative prototyping and user involvement, was used in the development of MyHSE. The RAD approach made it easier to quickly create prototypes and iteratively improve them in response to user feedback. The methodology allowed for regular revisions and adaptations, guaranteeing that the system's functionality and design precisely met the changing needs of both service providers and subscribers. MyHSE offers a number of crucial features that will improve the customer experience in the OSH sector. Providers of OSH services can quickly upload and advertise the services they provide, complete with pertinent contact information and service specifics. Subscribers, in turn, have access to a wide range of trustworthy services, making booking and service requests easier. MyHSE uses a strong competent person verification mechanism to foster trust and dependability. Service providers must go through verification, during which the validity of their licences and registration numbers is examined. Subscribers are reassured by this rigorous verification process that they are working with qualified and competent professionals.

## TABLE OF CONTENT

## LIST OF TABLES

## LIST OF FIGURES

## LIST OF ABBREVIATIONS

| | |
|---|---|
| OSH | Occupational Safety and Health |
| OSHA | Occupational Safety and Health Administration |
| SQL | Structured Query Language |
| BOSH | Berkat OSH Services |
| COVID-19 | Coronavirus disease |
| OHSNET | OHS Network Sdn Bhd |
| OHD | Occupational Health Doctors |
| CHRA | Chemical Health Risk Assessment |
| HRA | Health Risk Assessment |
| HCI | Human-Computer Interaction |
| US | United States |
| MyHSE | My Health Safety Environment |
| RAD | Rapid Application Development |
| ERD | Entity Relationship Diagram |
| iOS | iPhone Operating System |
| UAT | User Acceptance Test |
| FTP | File Transfer Protocol |
| TCP | Transmission Control Protocol |
| DNS | Domain Name System |

# CHAPTER 1

# INTRODUCTION

## 1.1 Introduction

Occupational Safety and Health (OSH) is concerned with all aspects of workplace health and safety, with a strong emphasis on risk prevention. Its goal is to keep people safe from workplace accidents and harm (Phia, 2022). Occupational health issues are generally more difficult to confront. Thus, they receive less attention than occupational safety issues. The moment health is addressed, however, safety is addressed as well, because a healthy workplace is by definition also a safe workplace. However, the opposite may be true: a so-called safe workplace is not always a healthy workplace. The important point is that health and safety issues must be addressed in every workplace. In general, the above definition of occupational health and safety encompasses both health and safety in their broadest contexts.

Organisations have developed OSH services in accordance with global standards and Malaysian legislation to address this issue. These organisations provide services to other businesses in fields such as chemical, medicine, ergonomics, hygiene, and many others. Companies that provide OSH services now have their own website where they can display their service catalogues and company information. This has been a trend that many organisations have followed in terms of digitalization measures. However, the systems developed by the companies serve only two purposes: displaying their services and contact information.

Therefore, a responsive web-based system for centralised OSH services has been proposed. In this system, users with various roles will be able to upload their services, and subscribers will be able to book the service. In Chapter three, we will go over the system's functionality in greater detail.

## 1.2 Problem Statement

As a result of global digitalization, many scams have emerged in the occupational safety and health (OSH) industry. According to (Jane, 2022), small businesses lost $1.3 million as a result of an OSHA labour law poster scam. Moreover, the traditional method of obtaining OSH service is time-consuming due to numerous verification processes and the search for a legitimate service provider. There are countless OSH service providers available, but clients have no way of knowing if they are authorised. Users or clients must be assured that the service provider can be relied on. The implementation of competent person verification using the certificate and registration number ensures that the client is confident that this application is legitimate and that no suspicions arise.

Furthermore, while there are numerous websites and applications for OSH services, these applications only display the service they provide. Customers are also unable to contact the service provider directly for booking inquiries or price negotiations. According to (Team I. E., 2021), effective communication helps businesses build trust with their customers and articulate their needs, expectations, and challenges. Improve the client relationship and potentially add more leads to the business by communicating more effectively. Clients will be able to get a good idea of the services and pricing by displaying a pool of services with the competent person and their details. They will also be able to contact the service provider directly for clarification.

As a result, the proposed project with a centralised OSH service application is expected to solve some of the industry's current problems. From competent person verification to posting services, users will be able to select the service they require from a pool of options. They will also be able to contact the individual directly and book the service. Other users will be able to use the application with less suspicion.

## 1.3 Objective

The objectives of this project are:

i. To study existing OSH service applications and its functionality.

ii. To design a responsive web-based for centralized OSH service system as a solution from the study.

iii. To evaluate the functionality and usability of the proposed system.

## 1.4 Scope

User

i. Public user/guest

ii. Competent person/service provider with valid certification to provide services

iii. Subscriber of the application

iv. System administrator

Development

i. System will be developed using bootstrap for the front-end and Laravel for the back-end

ii. MySQL will be used for the database

iii. System can be used both in mobile and computer browser as it will be a responsive web-based system

System

i. Covers only OSH services in Malaysia

## 1.5 Thesis Organisation

This thesis is divided into five chapters. The first chapter discusses the introduction to the centralized OSH service system, which uses a responsive web-based framework, as well as the problem statement, objectives, and scope.

The second chapter depicts a review of the literature on three existing OSH service systems. In this chapter, the advantages and disadvantages of existing systems will be compared in order to provide better guidance for the development of the system.

The methodology used in this project will be discussed in Chapter three, which includes the proposed system design, software and hardware specifications, and a Gantt chart.

Chapter four describes the project's implementation, results, and discussion. It demonstrates how the results for this project were achieved successfully by adhering to the plan outlined in the previous chapter

This project's findings will be summarized in Chapter five. This chapter will also discuss the project's limitations, limits, and future work.

# CHAPTER 2

# LITERATURE REVIEW

## 2.1 Introduction

This chapter describes three (3) existing OSH service management systems. Section 2.2 discusses and demonstrates in detail how the three existing complaint systems work. Following that, the system's advantages and disadvantages will be discussed in depth. Section 2.3 will include a comparison of these three systems. Using the information from the comparison, a more efficient and effective process for developing this project can be created and designed.

## 2.2 Existing Systems

### 2.2.1 Berkat OSH Services Portal

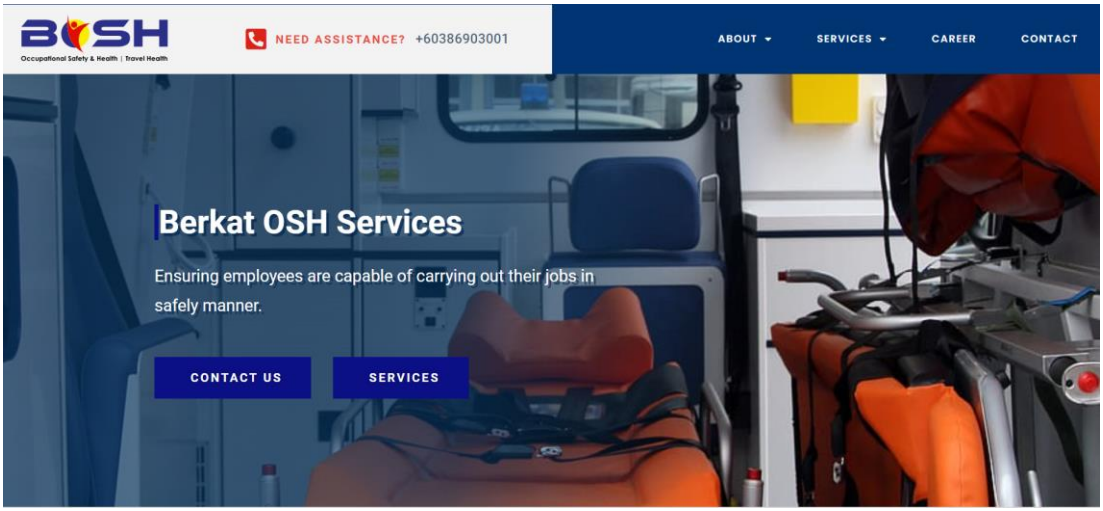



Figure 2.1 BOSH Main Page

Berkat OSH Services (BOSH) is a local company that offers occupational safety and health services to a wide range of industries. This business is registered with the Ministry of Finance and Petronas. They also have three occupational health clinics staffed by qualified medical examiners from Petronas, Shell, Oguk, Marine, and Petron. They offer remote medical service, occupational health service, employee wellness programme, COVID-19 management, and many other services (BOSH, 2022).


Figure 2.2 BOSH Services Page

The user interface of this portal is simple and understandable, which is one of its advantages. This makes the user feel at ease when using the application. This portal's primary function is to display their pool of services and officers. The next function is contact information for user inquiries, and users can also send a message using the form provided on the contact page. The system's final function is that users looking for job opportunities can apply for available positions by submitting their information along with their resume. This application is a responsive web-based system that users can access via their mobile browser.

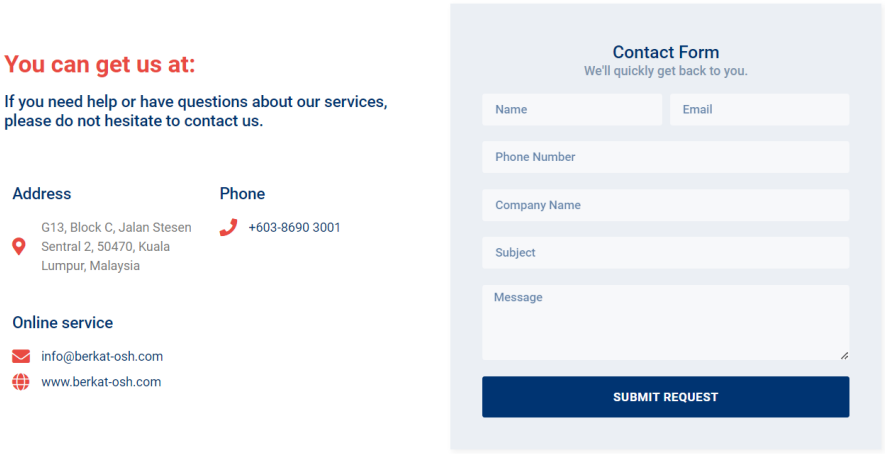



Figure 2.3 BOSH Contact Us Page

Although this system serves its purpose, there are a few disadvantages that must be addressed. First and foremost, aside from displaying information on the pool of services and the officers involved, there isn't much that users can view, such as previous client reviews. Furthermore, users are unable to contact specific officers directly for any inquiries because the contact information provided is the companies. This will make company-client communication time-consuming. Moreover, users cannot be certain that the officers involved have been verified by the government.

### 2.2.2 OSH Network Sdn Bhd Portal

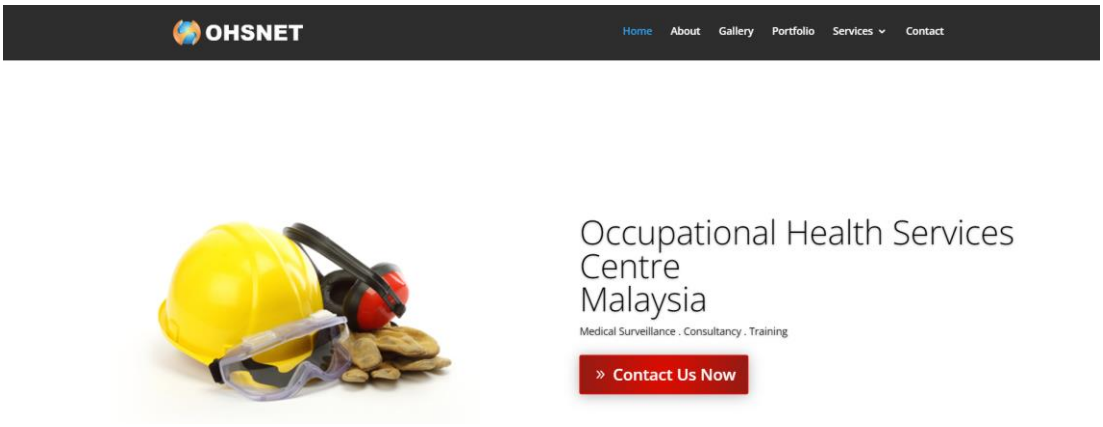



Figure 2.4 OSHNET Main Page

Dr. Azrul R. Abdullah founded OHSNET (OHS Network Sdn Bhd) in 2013. It functions as a network of certified and dedicated occupational health practitioners such as occupational health specialists, occupational health doctors (OHD), industrial

hygienists, competent individuals such as CHRA, ergonomists, HRAs, and other OSH practitioners. They provide services such as medical surveillance, occupational safety and health training, consultation, and a wellness programme (Syah, 2021).

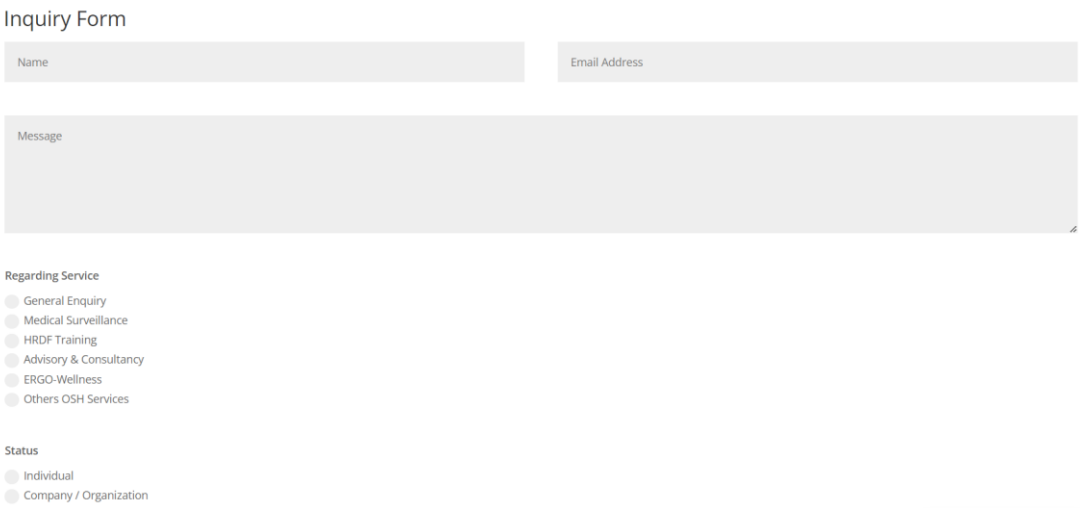



Figure 2.5 OSHNET Inquiry Form

One of the benefits of this portal is that it was created using a responsive web-based design, which allows users to access it from any device. This system also includes the basic function of displaying all of the services they provide, including the service gallery. In their portfolio, they also show all of the companies with which they collaborate. Users can contact the company using the information they provide or by completing the contact form on the website. This portal's user interface is consistent and simple for new users to grasp.

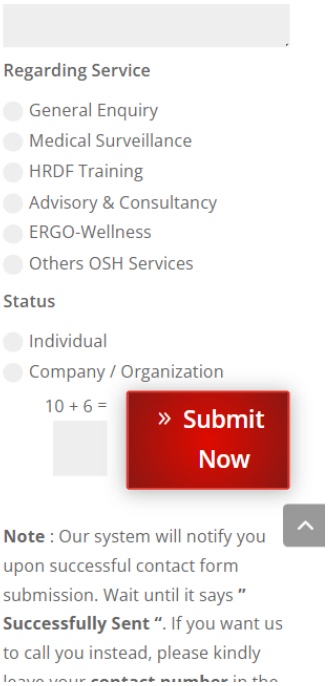



Figure 2.6 OSHNET Mobile View Flaw

The disadvantage of this system is that users can only view the pool of services without any further information on the process and protocols. Not only that, but the portal is designed to be accessible via mobile browser. Nevertheless, the design has flaws in a few areas, such as the contact page, as shown in the figure above. There is no rating or review system in place to help users determine the legitimacy of the system and its officers. Users are unable to book services online, so the portal merely serves as a display tool.

### 2.2.3 US Department of Labour: OSHA Portal

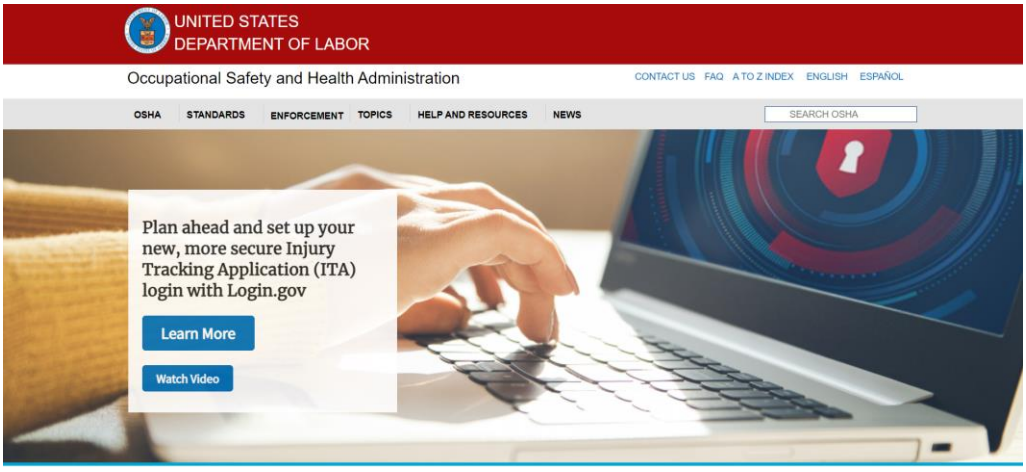



Figure 2.7 OSHA Portal Main Page

The Occupational Safety and Health Administration (OSHA) was established by the United States Congress to ensure safe and healthy working conditions for workers by establishing and enforcing standards as well as providing training, outreach, education, and assistance. The OSH Act applies to the majority of private sector employers and their employees, as well as some public sector employers and employees in the 50 states and certain territories and jurisdictions under federal authority (Parker, 2018). In accordance with the law and regulations, this portal serves as the foundation for all OSH services in the United States.

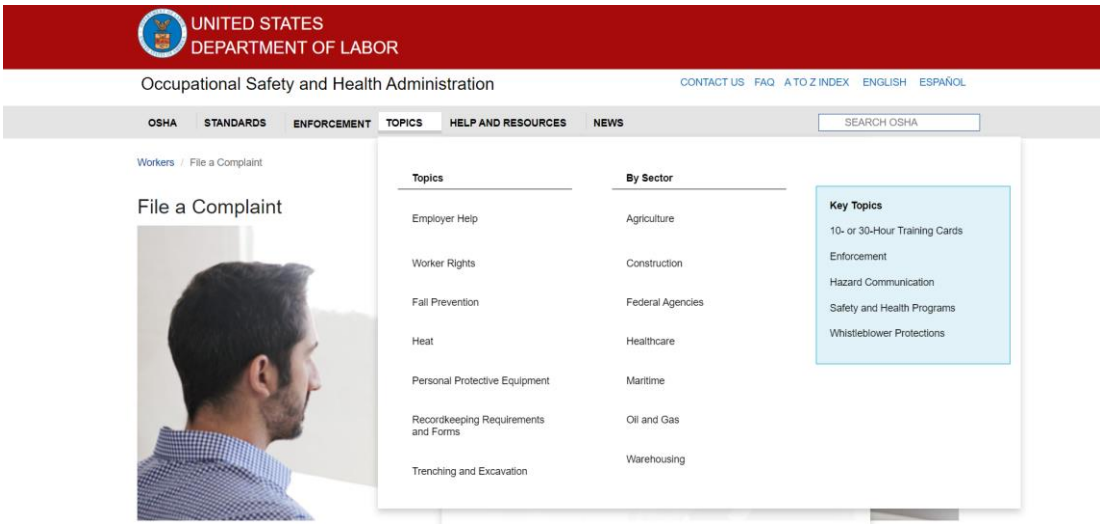



Figure 2.8 OSHA Portal Features

One of the benefits of this system is that it is rich in resources, allowing users to find anything related to the OSH field. Everything from data and statistics to reports and publications is saved in the portal and ready for the user to use. Users can also file a complaint using the numerous options available. The user interface is consistent throughout the system, and the website is responsive in design. This portal's sub sections have been properly divided according to sectors to make it easier for users to find the resource they require. As shown in the figure above, some examples include oil and gas, construction, and many others. This system also includes a wide range of language user can choose from.


Figure 2.9 OSHA Portal User Interface Design

Although this system contains all of the resources that the user requires, the user interface does not meet Human-Computer Interaction (HCI) standards. As illustrated in the figure above, there are far too many words, and the font size makes it difficult to read. This may cause user frustration, which has a direct impact on the user experience. Furthermore, there are no functions that allow users to book a service directly from the portal. Besides that, every time a user navigates through the system, the page takes at least seven seconds to load.

## 2.3 Analysis Comparison of Existing System

### 2.3.1 Comparison of Features on Existing System with Proposed Project

Table 2.1 Comparison of Features on Existing System with Proposed Project

| System Name | About Company | List of Services | Manage Profile | Add Booking | Contact Details | Translation | Indexed Database |
|---|---|---|---|---|---|---|---|
| Berkat OSH Services Portal | ✓ | ✓ | × | × | ✓ | × | × |
| OHS Network Sdn Bhd Portal | ✓ | ✓ | × | × | ✓ | × | × |
| US Department of Labour: OSHA Portal | ✓ | × | × | × | ✓ | ✓ | ✓ |
| MyHSE | ✓ | ✓ | ✓ | ✓ | ✓ | × | ✓ |

### 2.3.2 Advantages and Disadvantages

Table 2.2 Advantages and Disadvantages

| System Name | Scope | Type | Advantages | Disadvantages |
|---|---|---|---|---|
| Berkat OSH Services Portal | Focuses on OSH services in Malaysia | Responsive web-based | • Responsive web-based<br>• Good user interface<br>• Display pool of services<br>• Contains contact form<br>• Simple content<br>• Easy to navigate | • No direct booking<br>• No direct contact |
| OHS Network Sdn Bhd Portal | Focuses on OSH services in Malaysia | Responsive web-based | • Responsive web-based<br>• Good user interface<br>• Display pool of services<br>• Gallery of previous services<br>• Contains contact form<br>• Easy to navigate | • No direct booking<br>• No direct contact<br>• Mobile view has flaws in certain areas |
| US Department of Labour: OSHA Portal | Focuses on OSH services in US only | Responsive web-based | • Responsive web-based<br>• Display pool of services<br>• Rich with resources<br>• Easy navigation<br>• Indexed database | • Slow response time<br>• Bad interface design<br>• No direct contact<br>• No review on previous services |

### 2.3.3 Relevance of comparison with MyHSE

The findings on these three existing systems show that the companies approach their clients differently with their OSH services. This study provides a better understanding of the local and US portal features that will be implemented in this project. It also proposes new ideas that should be implemented to ensure that this new system does not become another typical OSH service portal. In a nutshell, this comparison and research on existing systems has had a significant impact on the project implementation thoughts.

## 2.4 Summary

In summary, the Occupational Safety and Health (OSH) industry is in the advanced digital globalisation era, where small businesses are being impacted and losing millions to scammers on the OSHA labour law poster. Furthermore, due to numerous verification processes and the search for a legitimate service provider, the current method of obtaining OSH service is time-consuming. Therefore, this project with a centralised OSH service application was proposed to solve some of the industry's current issues by allowing users to choose the service they need from a pool of options, contact the individual directly, and book the service.

Three existing OSH service portals, two of which were local systems and one of which was a US portal, were studied and compared based on features and user experience. Moreover, the advantages and disadvantages of these systems were examined in order to provide a fresh perspective on the proposed project. This concept will be used to properly direct this project in order to meet all client requirements.

# CHAPTER 3

# METHODOLOGY

## 3.1 Introduction

This chapter discusses the project's overall approach or framework. The methodology is a structured process that allows for the creation of high-quality, low-cost software in the shortest amount of time. The framework's goal is to produce superior software that meets customer expectations and demands since it clarifies the problem or goal because it is easy to get ahead of oneself when undertaking a large project.

## 3.2 Project Management Framework

Rapid Application Development (RAD) refers to an adaptive software development model that relies on prototyping and immediate feedback rather than detailed planning. In general, the RAD approach prioritises development and prototyping over planning. Rapid application development allows developers to make multiple iterations and updates to software without having to start from scratch. This helps to ensure that the final product is more quality-focused and meets the needs of the end users (Wood, 2022).

The benefits of the RAD model include the ability to change requirements at any time. Since prototyping and refinement will be done iteratively, the client will be able to request changes more easily. This promotes client feedback. Since the reviews will be quick, the development time will be significantly reduced. Furthermore, the time between prototypes and iterations is short because any changes in requirements are quickly amended. Finally, integration is not an issue because it is built in from the start of the project.

Although this model has advantages, it also has disadvantages. For example, highly skilled developers are required because the requirements will change frequently, and the developers must determine whether it is efficient and feasible. This model also considers user requirements throughout the product's life cycle (Simmons, 2022). Finally, rapid application development can only be used to create systems that can be modularized.

The four major stages in the RAD model are requirement planning, user design, construction, and cutover.

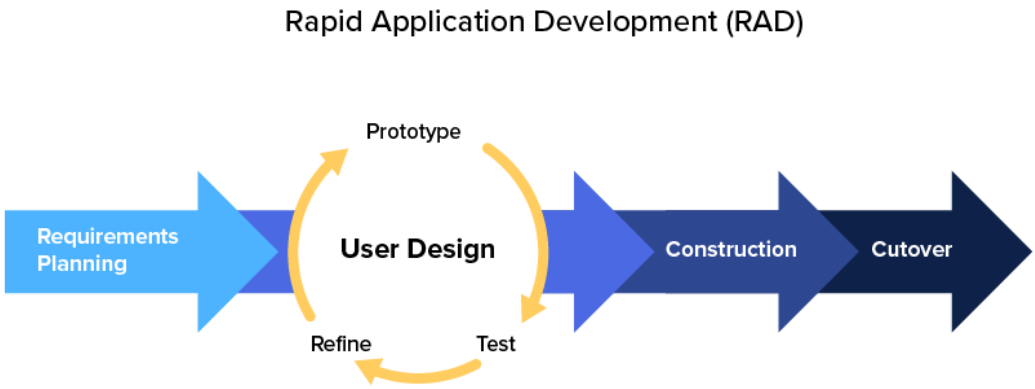



Figure 3.1 Rapid Application Development Methodology

### 3.2.1 Requirement Planning

During this stage, the developers, clients, and grant requirement team members communicate to establish project goals and expectations, as well as current and potential issues that will need to be addressed during construction. My supervisor had discussed the requirements with the client's company's principal consultant in order to build this system. This set of requirements was then analysed and revised in order to create a proper software requirement.

### 3.2.2 User Design

During this phase, clients collaborate with developers to ensure that their needs are met at every stage of the design process. This phase is similar to customizable software development in that users can test each product prototype at each stage to ensure it meets their expectations (Singh, 2022). The bootstrap template will be used to create an interactive interface for this system's prototype. Any changes to the requirements will be made only at the interface level, with no coding involved. Iteratively, all of the bugs and kinks are worked out. The developer creates a prototype, the client tests it, and then they collaborate to discuss what worked and what didn't.

### 3.2.3 Construction

The prototypes and beta systems from the design phase are converted into the working model in phase three. Since the majority of the issues and changes were addressed during the iterative design phase, developers can quickly build the final working model. This third phase is critical because the client can still provide feedback throughout the process. They can propose modifications, changes, or even new ideas to solve problems as they arise (Team L. C., 2021). In this project, the bootstrap prototype will be converted into a working system by using the Laravel framework as the system's back-end. The database will be built using the previously designed ERD, and all system modules will be tested.

### 3.2.4 Cutover

The finished product is launched during this phase of cutover. It includes data conversion, testing, and system transition, as well as user training. This project will be implemented in the client-approved domain, myhse.com. Testing will be performed prior to deployment to ensure that there are no bugs in the production server (Makadia, 2022). While the coders and clients continue to look for bugs in the system, all final changes are made.

## 3.3 Project Requirement

This section will go over the system's functional and non-functional requirements, constraints, and limitations. This project's user requirements are derived from interviews and feedback from the principal consultant.

### 3.3.1 Functional Requirement

1. Guest
    i. The system shall allow guest to view all available services
    ii. The system shall allow guest to subscribe and create a new account
2. Competent Person/Service Provider
    i. The system shall allow competent person to login into the system
    ii. The system shall allow competent person to manage their profile
    iii. The system shall allow competent person to upload their certificate and registration number
    iv. The system shall allow competent person to upload their services
    v. The system shall allow competent person to view any bookings on their services
3. Subscriber
    i. The system shall allow subscriber to login into the system
    ii. The system shall allow subscriber person to manage their profile
    iii. The system shall allow subscriber to view all available services and the contact details of the service provider
    iv. The system shall allow subscriber to search for services
    v. The system shall allow subscriber to book any service on the available date and time

4. Administrator

    i. The system shall allow admin to login into the system

    ii. The system shall allow admin person to manage their profile

    iii. The system shall allow admin verify the competent person

    iv. The system shall allow admin to approve subscriptions

    v. The system shall allow admin to manage all services

    vi. The system shall allow admin to manage all bookings

    vii. The system shall allow admin to manage all profiles

### 3.3.2 Non-Functional Requirement

Usability

i. This system's interface must be user-friendly and simple to use

ii. The interface must be efficient for all users to be able to use this system

### 3.3.3 Constraints and Limitations

Constraints

i. Users need internet connection to access the system

ii. The system supports Windows 7, Android 6.0 and iOS 11.0 or later

Limitations

i. The system will only be available in English language

ii. The system is limited to users in Malaysia

## 3.4 Proposed Design

### 3.4.1 Context Diagram

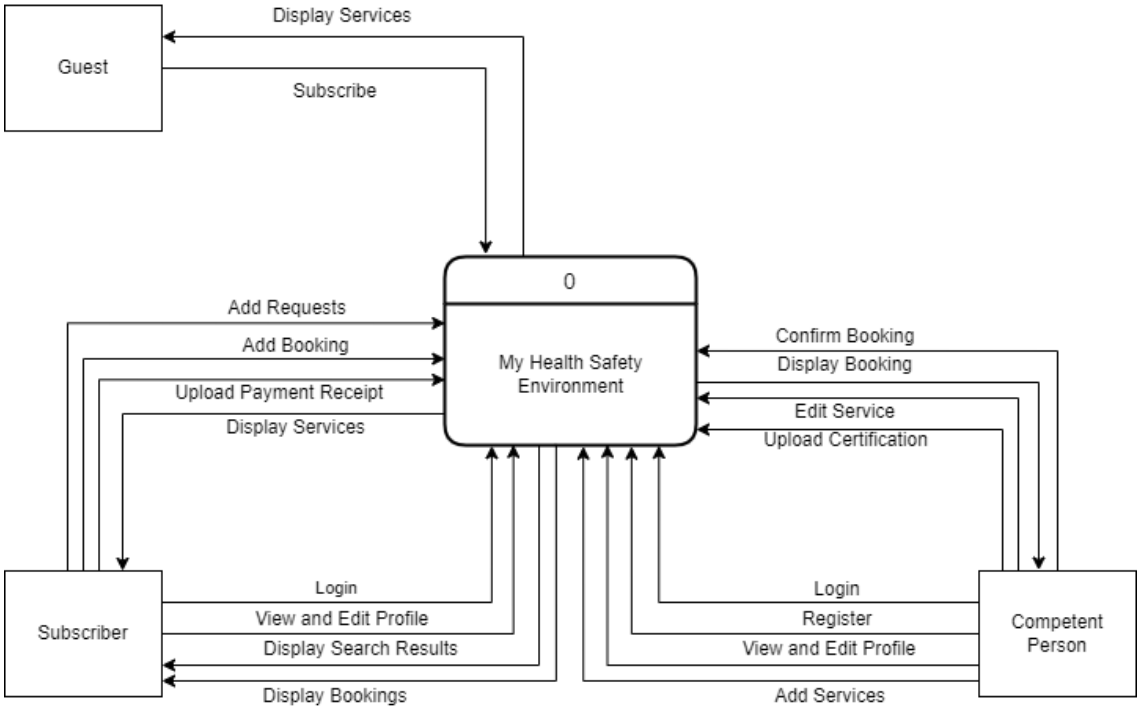



Figure 3.2 Context Diagram of myHSE

### 3.4.2 Use Case Diagram

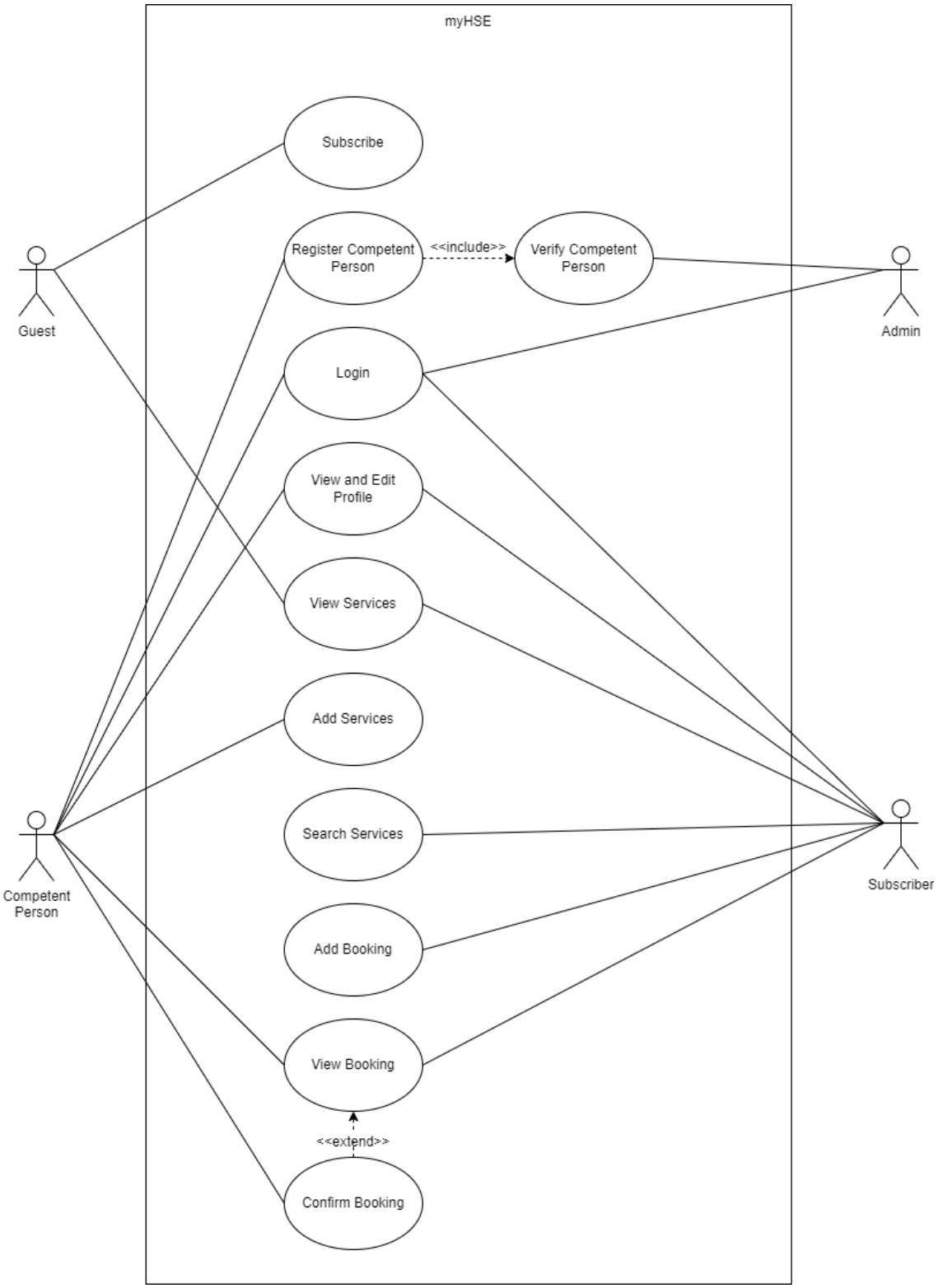



Figure 3.3 Use Case Diagram of myHSE

### 3.4.3 Use Case Description

#### 3.4.3.1 Login

Table 3.1 Login Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by subscriber, competent person and administrator. Users can enter their log in credentials and the system will authenticate the request to log them into the system. |
| **Actor** | Subscriber, Competent Person and Administrator |
| **Pre-Conditions** | All users must have a registered account |
| **Basic Flow** | 1. The use case begins when user clicks the login button in the landing page of the system.<br>2. The user must insert their username and password.<br>3. The user clicks the <<LOGIN>> button.<br>4. The system validates and authenticates the request. **[E1: Incorrect Credentials] [R1: Registered Account]**<br>5. The system recognizes the user type and display the home page accordingly.<br>6. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | **E1: Incorrect Credentials**<br>1. The system displays an error message that the user entered invalid credentials.<br>2. The use case continues with step 2 from basic flow. |
| **Post Conditions** | The system displays the main page according to the user type. |
| **Rules** | **R1: Registered Account**<br>1. The credentials entered must be available in the user database of the system. |
| **Constraints** | None |

#### 3.4.3.2 Subscribe

Table 3.2 Subscribe Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by guest. Guest can enter all required information to subscribe and create a new account. |
| **Actor** | Guest |
| **Pre-Conditions** | User access the system |
| **Basic Flow** | 1. The use case begins when user click on the <<SUBSCRIBE>> button.<br>2. The system displays the subscribe page.<br>3. The user must insert all the required information in this page and clicks the <<SUBSCRIBE>> button.<br>4. The system validates the data and saves in the database. **[E1: Username in use] [C1: Unique Username]**<br>5. The system redirects user to a restricted page until the admin approves this subscriber.<br>6. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | **E1: Username**<br>1. The system displays that the username is already in use by different account.<br>2. The use case continues with step 3 from basic flow. |
| **Post Conditions** | User logged in to the system |
| **Rules** | None |
| **Constraints** | **C1: Unique Username**<br>The username that is used during subscribing must be unique and not tied to any existing account |

#### 3.4.3.3 Register Competent Person

Table 3.3 Register Competent Person Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by competent person. Competent person can enter all required information to register and create a new account. |
| **Actor** | Competent Person |
| **Pre-Conditions** | User access the system |
| **Basic Flow** | 1. The use case begins when competent person clicks the <<REGISTER>> button.<br>2. The system displays the register competent person page.<br>3. The user must insert all the required page in this page and click the <<REGISTER>> button.<br>4. The system validates the request and saves the data in the database. **[E1: Username in use] [C1: Unique Username]**<br>5. The system redirects the user to a restricted page until the admin approves this competent person.<br>6. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | **E1: Username in use**<br>1. The system displays that the username is already in use by different account.<br>2. The use case continues with step 3 from basic flow. |
| **Post Conditions** | User logged in to the system |
| **Rules** | None |
| **Constraints** | **C1: Unique Username**<br>The username that is used during subscribing must be unique and not tied to any existing account |

#### 3.4.3.4 Verify Competent Person

Table 3.4 Verify Competent Person Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by competent person's registration request. The admin will verify the request and approve or reject the request. |
| **Actor** | Administrator |
| **Pre-Conditions** | Competent person submits the registration form |
| **Basic Flow** | 1. The use case begins when administrator access the verify competent person page.<br>2. The system displays all registration request.<br>3. The administrator selects on a particular request.<br>4. The system displays the details of the competent person.<br>5. The administrator checks for the registration number on a third-party system.<br>6. The administrator can then approve or reject the request.<br>7. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | None |
| **Post Conditions** | The competent person registration status is finalized |
| **Rules** | None |
| **Constraints** | None |

#### 3.4.3.5 View and Edit Profile

Table 3.5 View and Edit Profile Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by subscriber and competent person. |
| **Actor** | Subscriber and Competent Person |
| **Pre-Conditions** | Users are logged in the system |
| **Basic Flow** | 1. The use case begins when users click the <<PROFILE>> button.<br>2. The system displays all information of the user.<br>3. The user can then click the <<EDIT>> button to edit any details.<br>4. The system redirects user to the edit page.<br>5. The user can change their email, contact number, first name and last name.<br>6. The user clicks <<SAVE>> button.<br>7. The system validates the data and saves in the database.<br>8. The system displays data has been saved.<br>9. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | None |
| **Post Conditions** | User's new profile data is updated |
| **Rules** | None |
| **Constraints** | None |

### 3.4.3.6 View Services

Table 3.6 View Services Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by guest and subscriber. Users can view all the available services posted by the competent person |
| **Actor** | Guest and Subscriber |
| **Pre-Conditions** | Access the home page |
| **Basic Flow** | 1. This use case begins when users access the all services page of the system.<br>2. The system displays all the categories of services available.<br>3. Users can click on one particular category.<br>4. The system displays all the services available for this category. **[E1: No Services]**<br>5. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | **E1: No Services**<br>1. The system displays that there are no service available for this category.<br>2. The use case continues with step 5 from basic flow. |
| **Post Conditions** | User views all the available services |
| **Rules** | None |
| **Constraints** | None |

#### 3.4.3.7 Add Services

Table 3.7 Add Services Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by competent person. Competent person can post new services to be viewed by guest and subscriber. |
| **Actor** | Competent Person |
| **Pre-Conditions** | Competent person is verified |
| **Basic Flow** | 1. This use case begins when competent person accesses the services menu and clicks the <<ADD>> button.<br>2. The system checks if the user is verified. **[E1: User not Verified] [R1: Verified User]**<br>3. The system displays the add service form.<br>4. The user can enter all the required information and click the <<ADD>> button.<br>5. The system validates and saves the data.<br>6. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | **E1: User not Verified**<br>1. The system displays that the user is not verified.<br>2. The user clicks <<OK>> button.<br>3. The system redirects the user to the home page.<br>4. The use case continues with step 1 from the basic flow. |
| **Post Conditions** | The service is successfully added |
| **Rules** | **R1: Verified User**<br>1. The competent person must be verified in order to post a new service. |
| **Constraints** | None |

#### 3.4.3.8 Add Booking

Table 3.8 Add Booking Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by subscriber. Subscriber can add a booking after viewing the service posted by competent person. |
| **Actor** | Subscriber |
| **Pre-Conditions** | User logged into the system |
| **Basic Flow** | 1. This use case begins when user clicks the <<BOOK>> button in the view service page.<br>2. The system will display the booking page.<br>3. User must enter the date, time and location of the service booked. **[E1: Unavailable Date]**<br>4. The system will prompt user for confirmation.<br>5. User can click <<CONFIRM>> button to confirm the booking.<br>6. The system sends notification to the competent person.<br>7. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | **E1: Unavailable Date**<br>1. The system will display that the date booked is unavailable.<br>2. The use case continues with step 3 from basic flow. |
| **Post Conditions** | User successfully add a booking |
| **Rules** | None |
| **Constraints** | None |

#### 3.4.3.9 View Booking

Table 3.9 View Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by competent person and subscriber. Competent person and subscriber can view the booking that have been done. |
| **Actor** | Competent Person and Subscriber |
| **Pre-Conditions** | Subscriber added a booking |
| **Basic Flow** | 1. This use case begins when users access the booking menu.<br>2. Users can then click the <<VIEW>> button to view all the bookings that had been made.<br>3. The system displays all the bookings. **[E1: No Bookings]**<br>4. Users can click on one particular booking to view the details of the booking.<br>5. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | **E1: No Bookings**<br>1. The system displays that there are no bookings made.<br>2. The use case continues with step 2 from basic flow. |
| **Post Conditions** | Users able to view all bookings made |
| **Rules** | None |
| **Constraints** | None |

#### 3.4.3.10 Confirm Booking

Table 3.10 Confirm Booking Use Case Description

| | |
|---|---|
| **Brief Description** | This use case is initiated by competent person. Competent person can confirm bookings made by subscriber. |
| **Actor** | Competent Person |
| **Pre-Conditions** | Booking must be made prior to confirmation |
| **Basic Flow** | 1. This use case begins when competent person clicks the <<VIEW>> button in the bookings page.<br>2. Competent person can then <<CONFIRM>> or <<REJECT>> the bookings by clicking on the respective buttons.<br>3. After confirmation, the system will update the booking status in the database.<br>4. The system will then block the date and time so that other subscriber won't be able to book on that particular date and time.<br>5. The use case ends here. |
| **Alternative Flow** | None |
| **Exception Flow** | None |
| **Post Conditions** | Booking status is updated accordingly |
| **Rules** | None |
| **Constraints** | None |

### 3.4.4 Activity Diagram

#### 3.4.4.1 Login

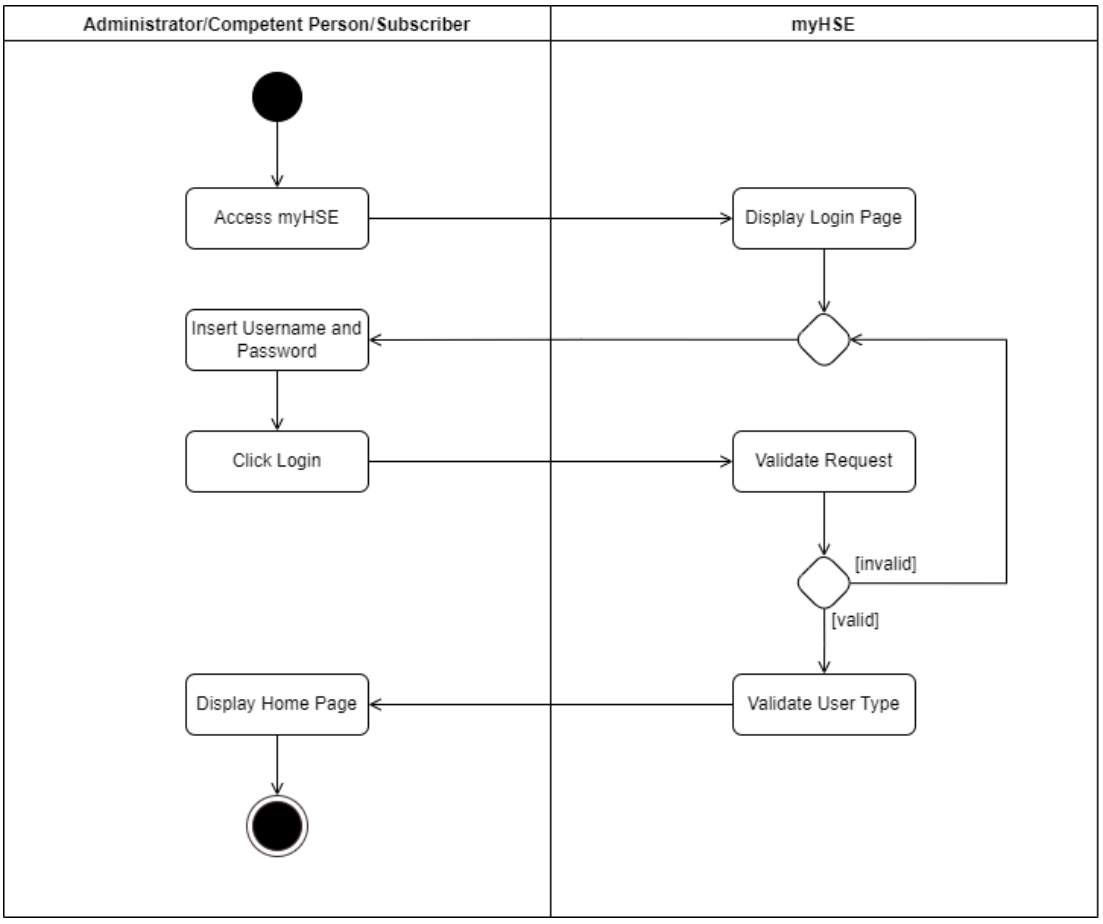



Figure 3.4 Login Activity Diagram

### 3.4.4.2 Subscribe

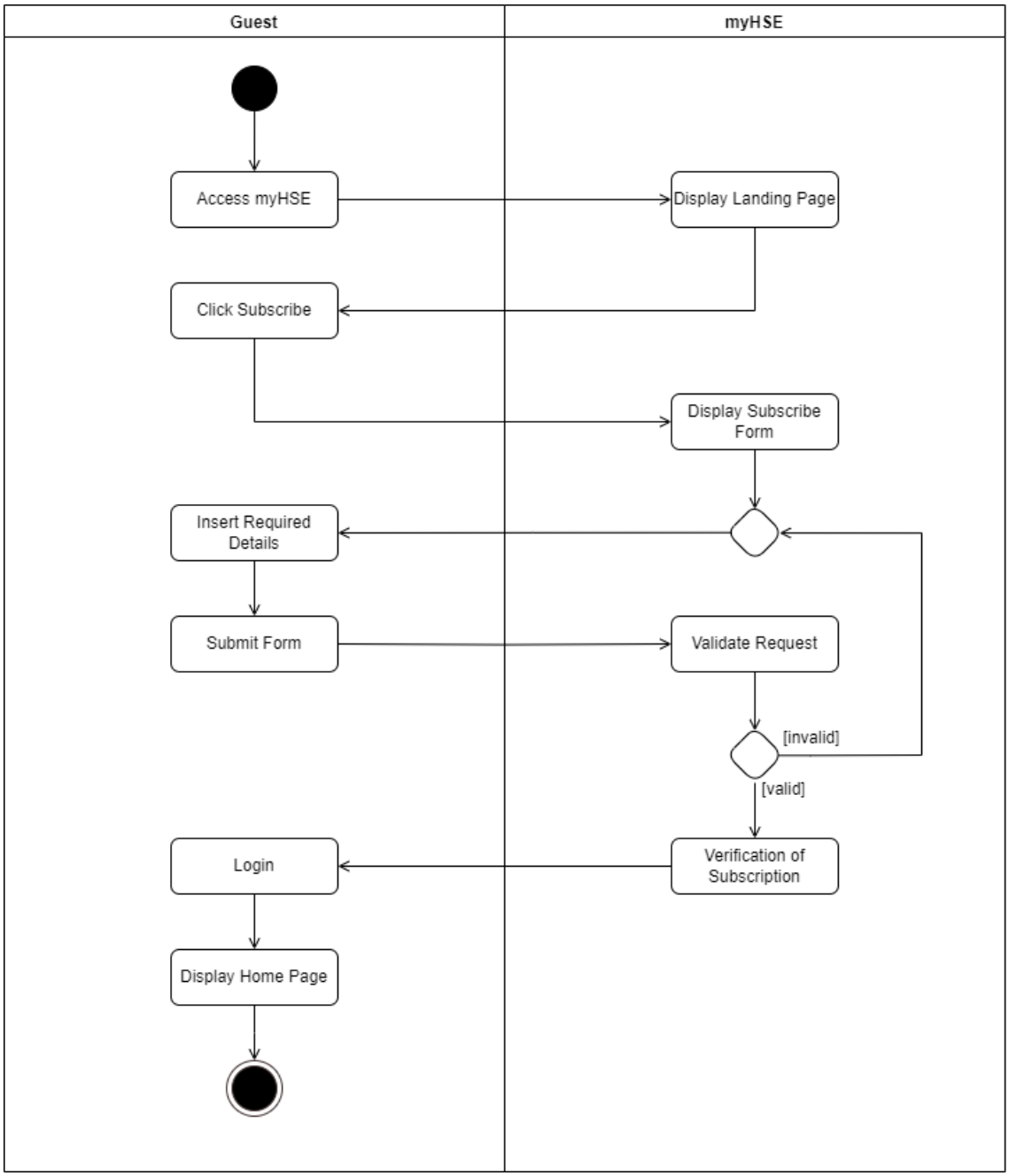



Figure 3.5 Subscribe Activity Diagram

### 3.4.4.3 Register Competent Person

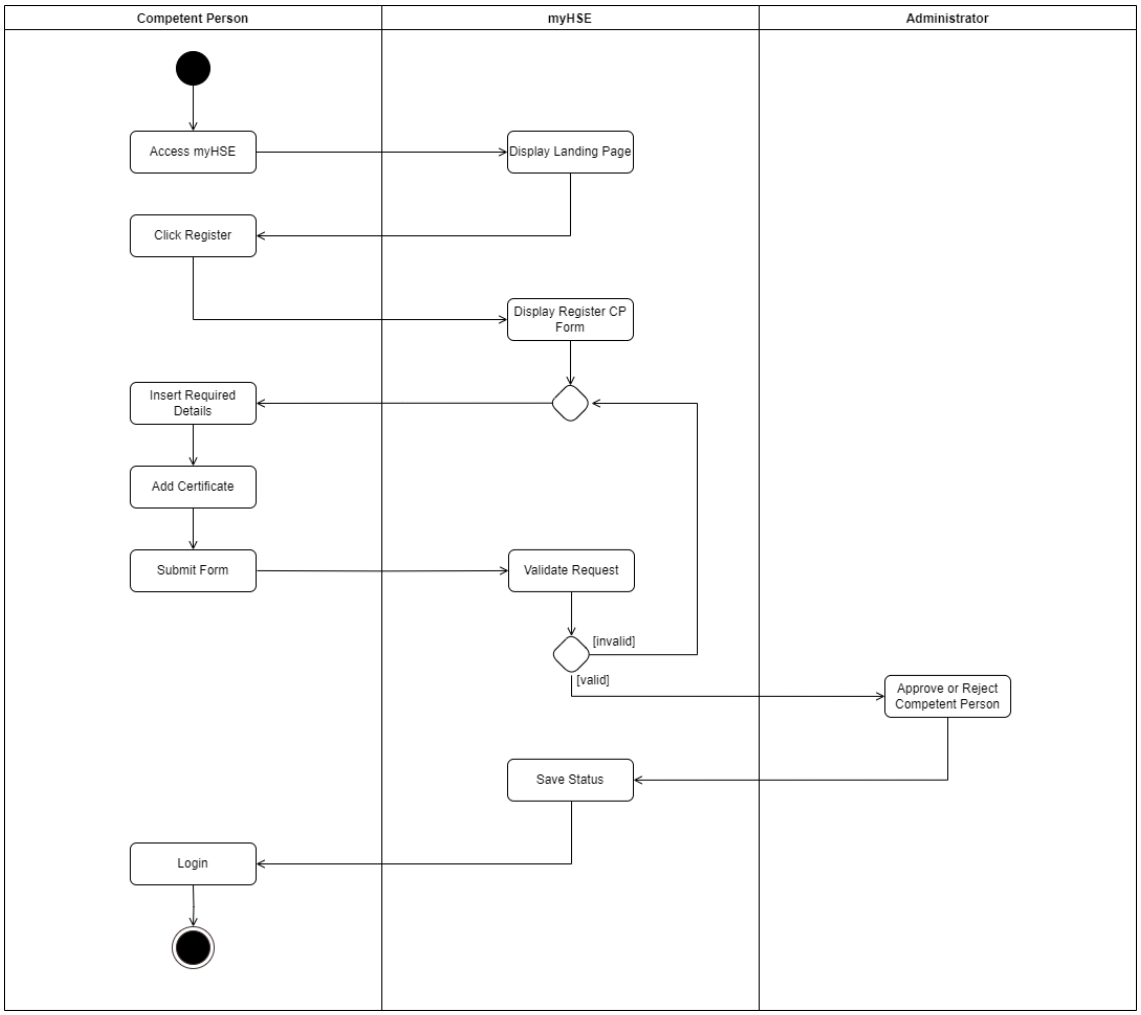



Figure 3.6 Register Competent Person Activity Diagram

### 3.4.4.4 View and Edit Profile

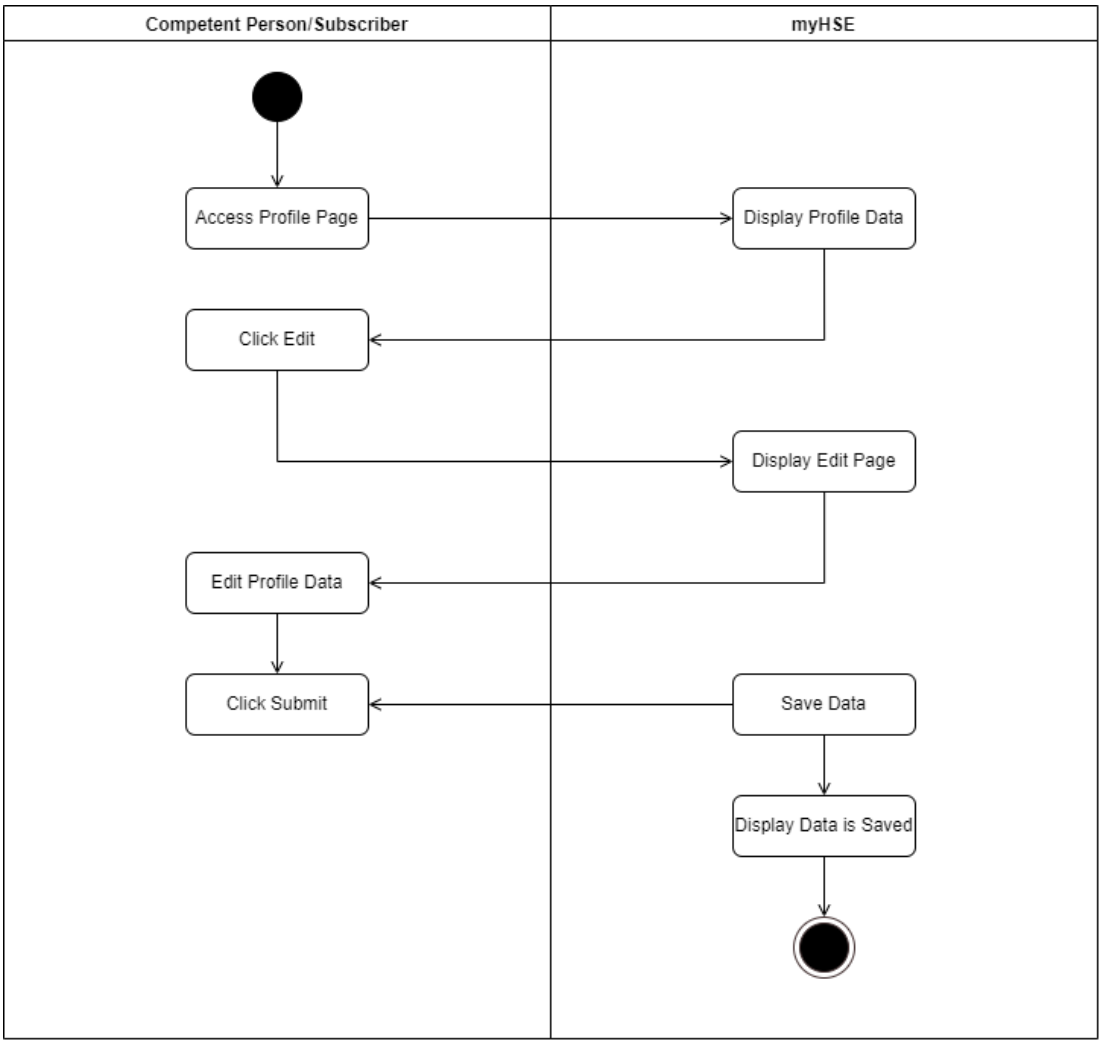



Figure 3.7 View and Edit Profile Activity Diagram

### 3.4.4.5 Manage Service

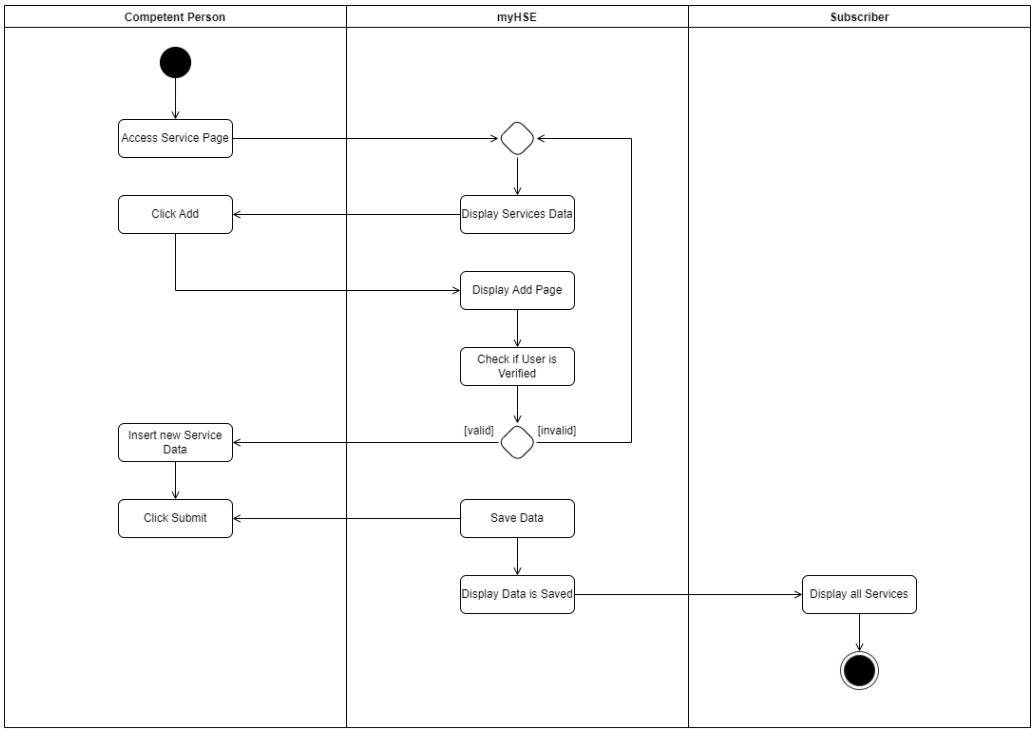



Figure 3.8 Manage Service Activity Diagram

### 3.4.4.6 Manage Booking

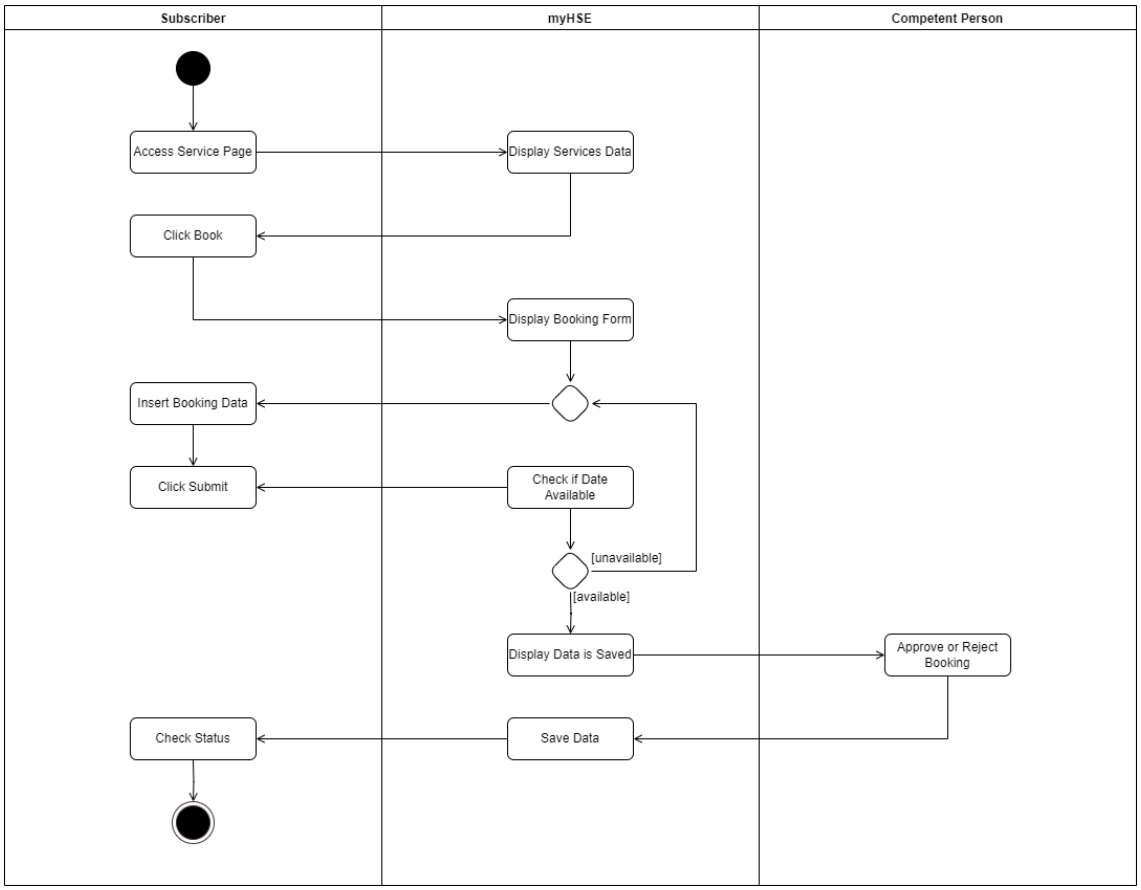



Figure 3.9 Manage Booking Activity Diagram

### 3.4.5 Flowchart

#### 3.4.5.1 Guest

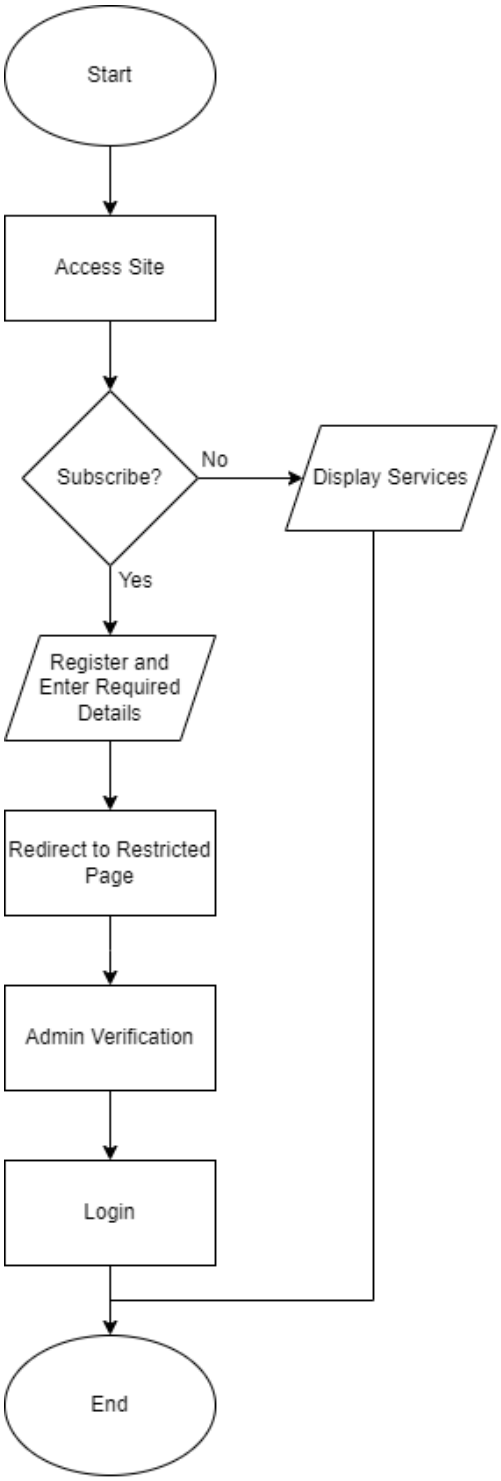



Figure 3.10 Guest Flowchart

### 3.4.5.2 Subscriber

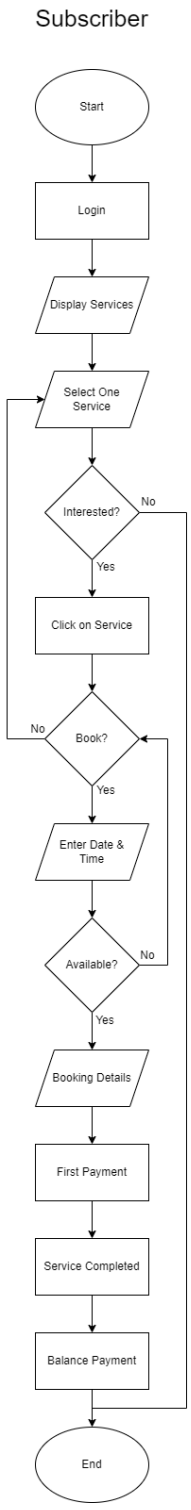



Figure 3.11 Subscriber Flowchart

### 3.4.5.3 Competent Person

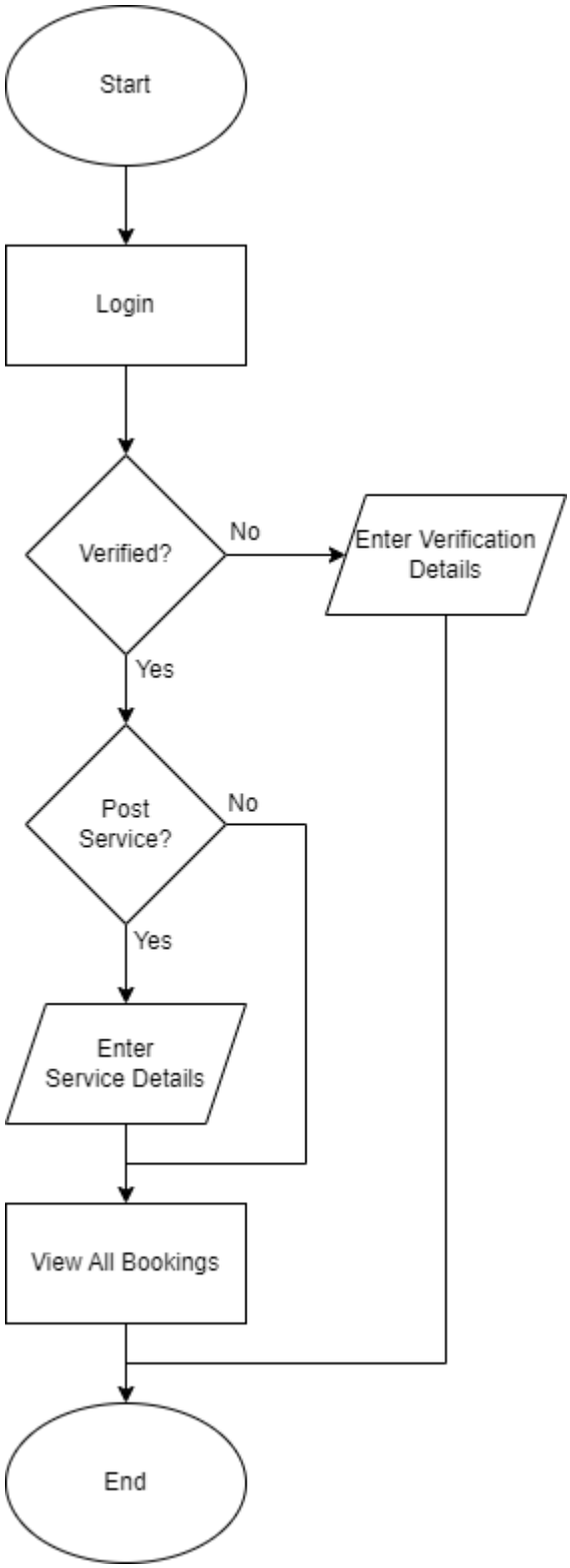



Figure 3.12 Competent Person Flowchart

#### 3.4.5.4 Administrator

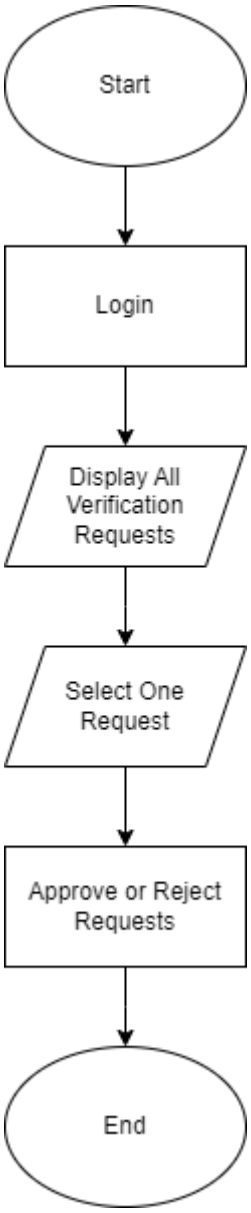



Figure 3.13 Administrator Flowchart

## 3.5 Data Design

### 3.5.1 Entity Relationship Diagram

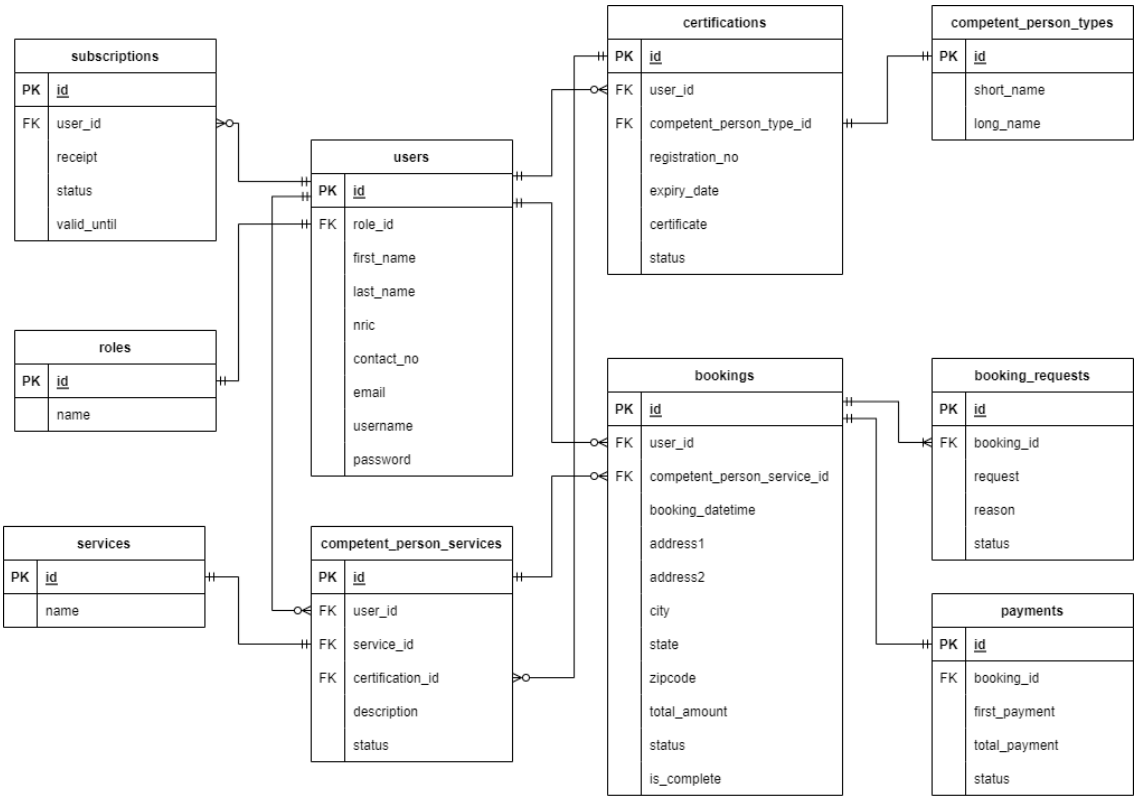



Figure 3.14 Entity Relationship Diagram for myHSE

### 3.5.2 Data Dictionary

#### 3.5.2.1 Roles

Table 3.11 Roles Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for role | INT | PK |
| name | User role type | STRING | |

#### 3.5.2.2 Users

Table 3.12 Users Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for users | INT | PK |
| role_id | Reference to role table | INT | FK |
| first_name | User first name | STRING | |
| last_name | User last name | STRING | |
| nric | User NRIC | STRING | |
| contact_no | User contact number | STRING | |
| email | User email | STRING | |
| username | Username | STRING | |
| password | Password | STRING | |

#### 3.5.2.3 Certifications

Table 3.13 Certifications Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for certification | INT | PK |
| user_id | Reference to user table | INT | FK |
| competent_person_type_id | Reference to competent_person_type table | INT | FK |
| registration_no | Certificate registration number | STRING | |
| expiry_date | Certificate expiry date | DATE | |
| certificate | Certificate file location | STRING | |
| status | Certificate verify status | STRING | |

#### 3.5.2.4 Competent_person_services

Table 3.14 Competent_person_services Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for CP service | INT | PK |
| user_id | Reference to user table for competent person user | INT | FK |
| service_id | Reference to service table | INT | FK |
| certification_id | Reference to certification table | INT | |
| status | Service visibility | BOOLEAN | |
| description | CP service description | TEXT | |

#### 3.5.2.5 Services

Table 3.15 Services Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for service | INT | PK |
| name | Service name | STRING | |

#### 3.5.2.6 Subscriptions

Table 3.16 Subscriptions Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for subscription | INT | PK |
| user_id | Reference to user table | INT | FK |
| receipt | Receipt stored location | STRING | |
| valid_until | Subscription valid until date | DATE | |
| status | Approval status | BOOLEAN | |

#### 3.5.2.7 Competent_person_types

Table 3.17 Competent_person_types Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for competent person types | INT | PK |
| short_name | CP type short name | STRING | |
| long_name | Cp type long name | STRING | |

#### 3.5.2.8 Bookings

Table 3.18 Bookings Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for bookings | INT | PK |
| user_id | Reference to user table for subscriber user | INT | FK |
| competent_person_service_id | Reference to CP Service table | INT | FK |
| booking_datetime | Booking datetime | DATETIME | |
| address 1 | Address line 1 | STRING | |
| address 2 | Address line 2 | STRING | |
| city | City | STRING | |
| state | State | STRING | |
| zipcode | Zip code | STRING | |
| total_amount | Total service charge | DECIMAL | |
| is_complete | Completion status | BOOLEAN | |
| status | approval status | BOOLEAN | |

#### 3.5.2.9 Booking_requests

Table 3.19 Booking_requests Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for booking requests | INT | PK |
| request | Request description | STRING | |
| reason | Reject reason | | |
| status | Request approval | STRING | |

#### 3.5.2.10 Payments

Table 3.20 Payments Data Dictionary

| Attribute | Description | Data Type | Constraint |
|---|---|---|---|
| id | Unique ID for payments | INT | PK |
| first_payment | Location for receipt of first payment | STRING | |
| balance_payment | Location for receipt of second payment | STRING | |

## 3.6 Proof of Initial Concept

This section will provide evidence of the preliminary work done on the proposed software using mock-up. Overall, the design mock-up serves as a proof of concept for the software and lays the groundwork for the development and testing phases.

### 3.6.1 Guest

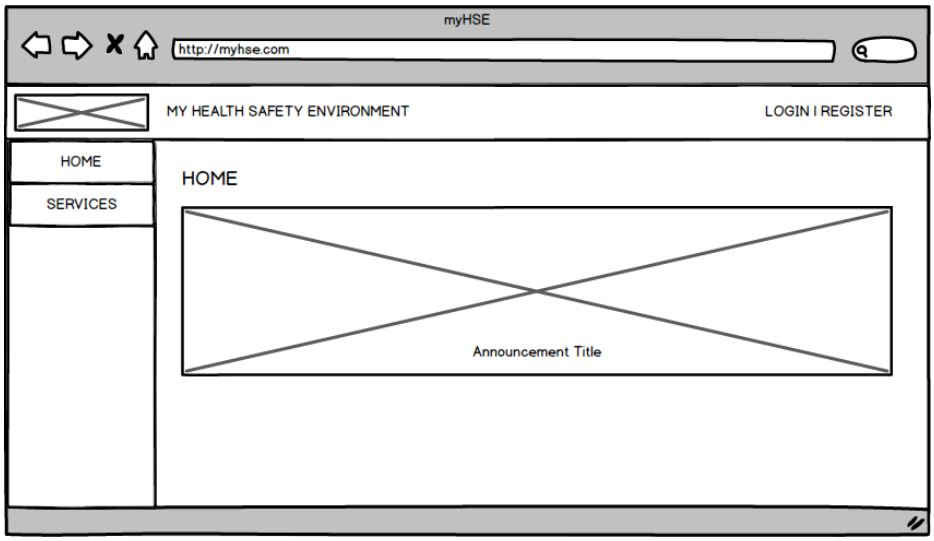



Figure 3.15 Guest Home Page

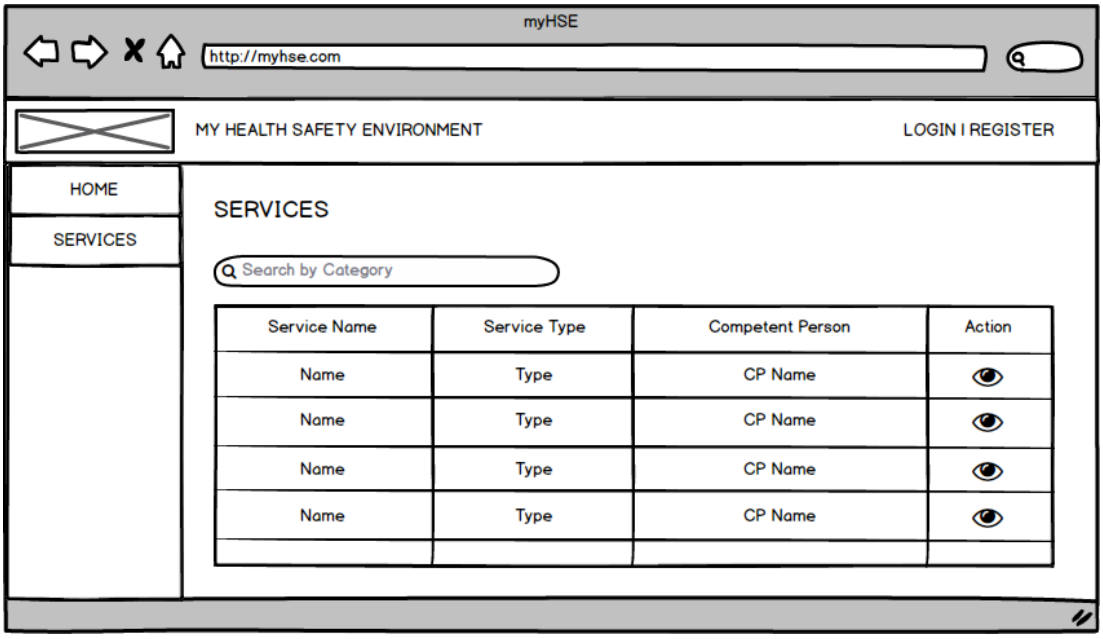



Figure 3.16 Guest Services Page

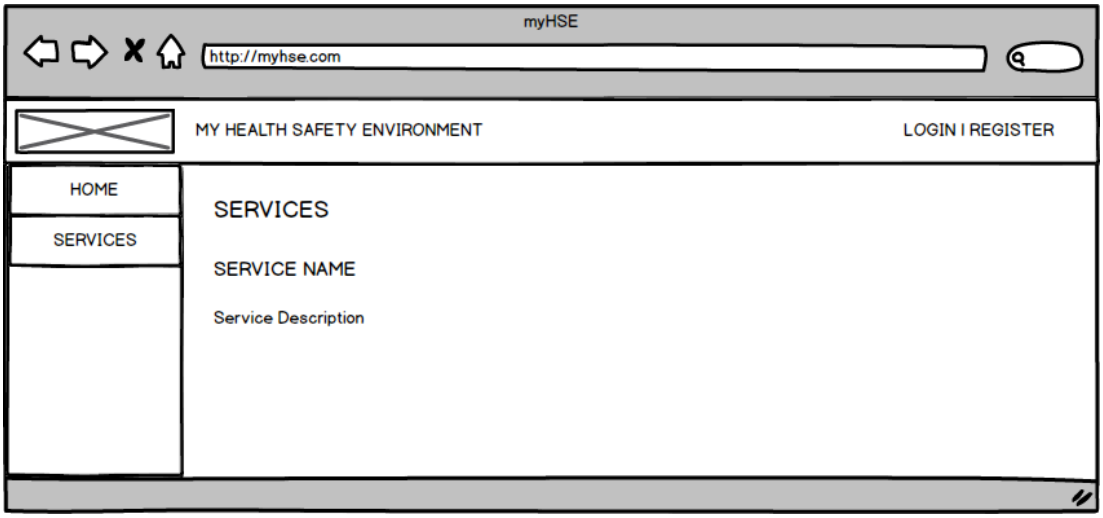



Figure 3.17 Guest Service Detail

### 3.6.2 Subscriber

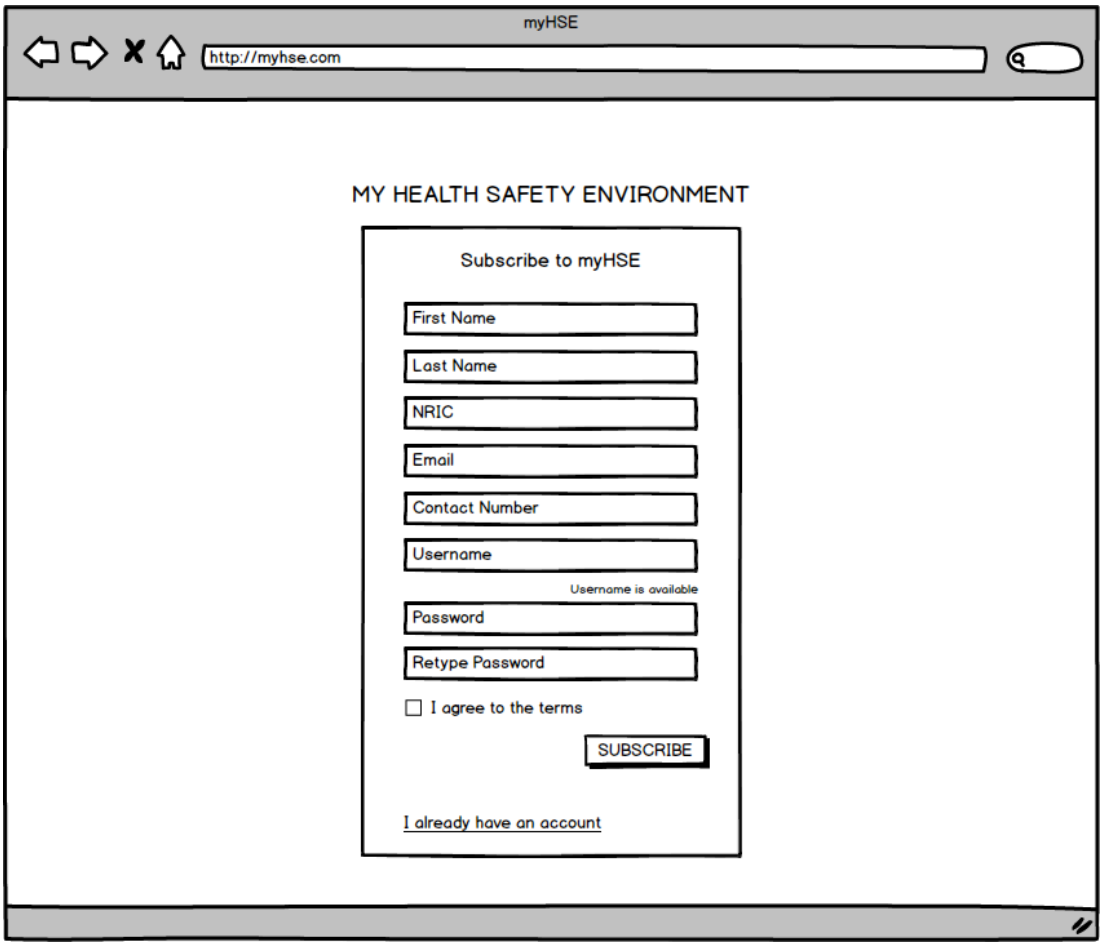



Figure 3.18 Subscriber Subscribe Page

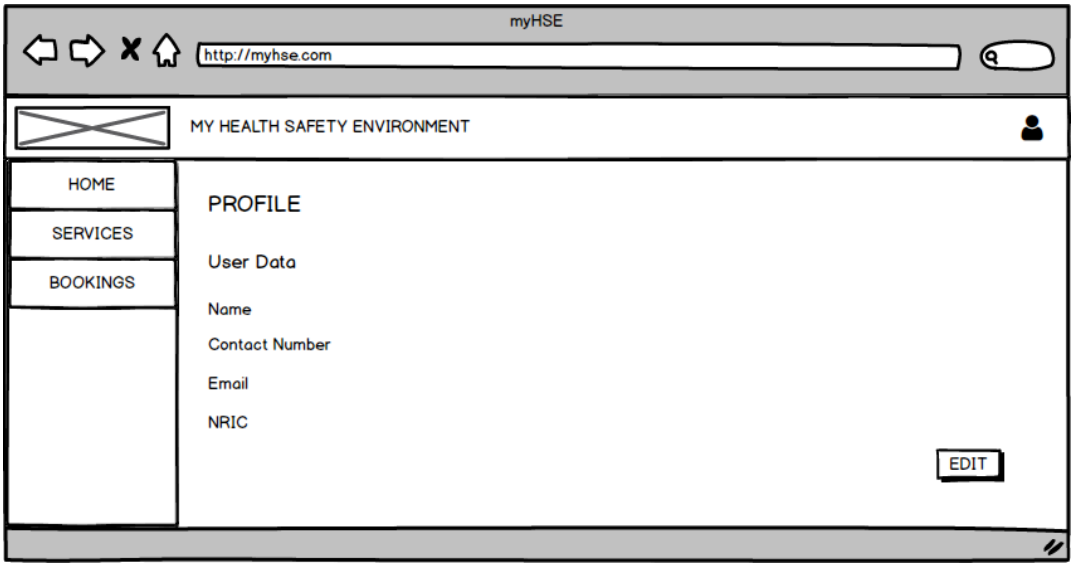



Figure 3.19 Subscriber Profile Page

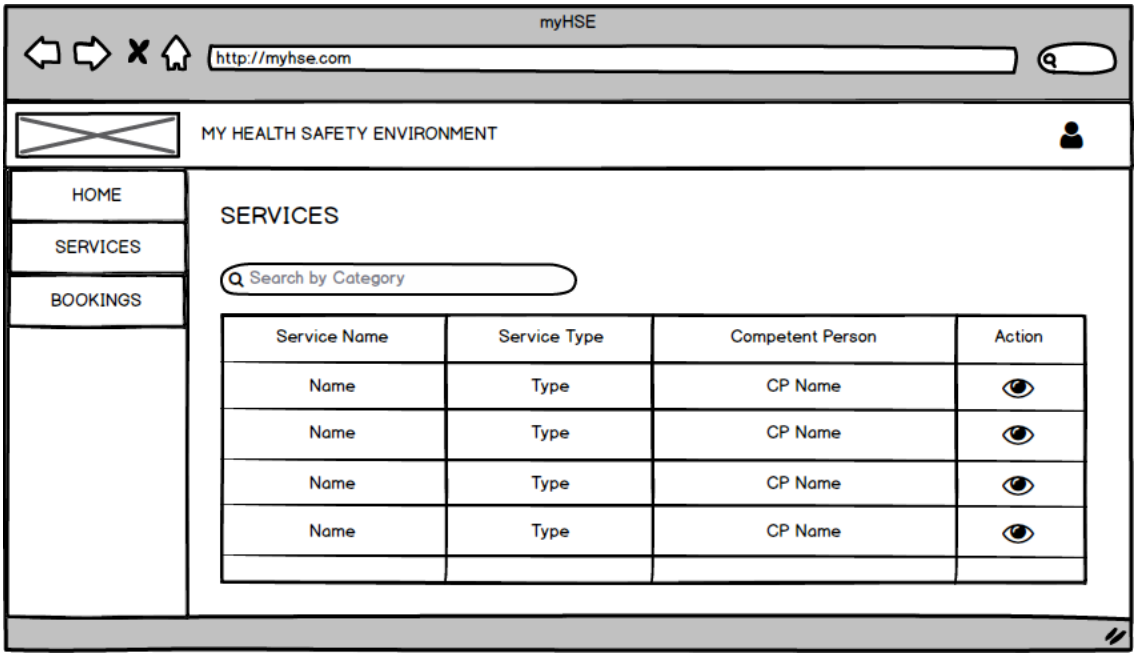



Figure 3.20 Subscriber View Services Page

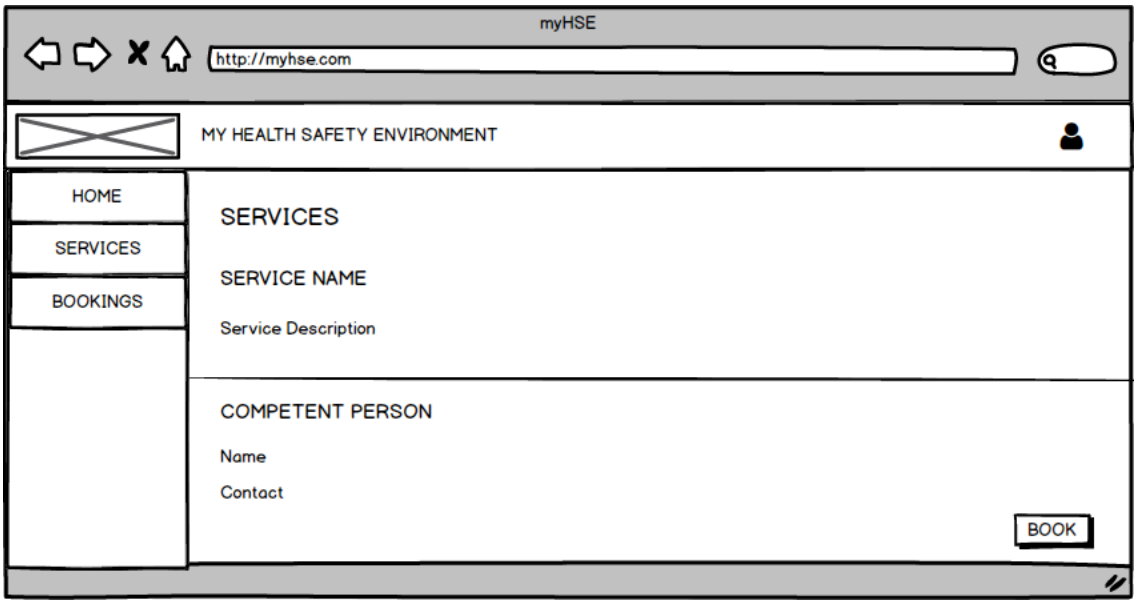



Figure 3.21 Subscriber View Service Detail Page

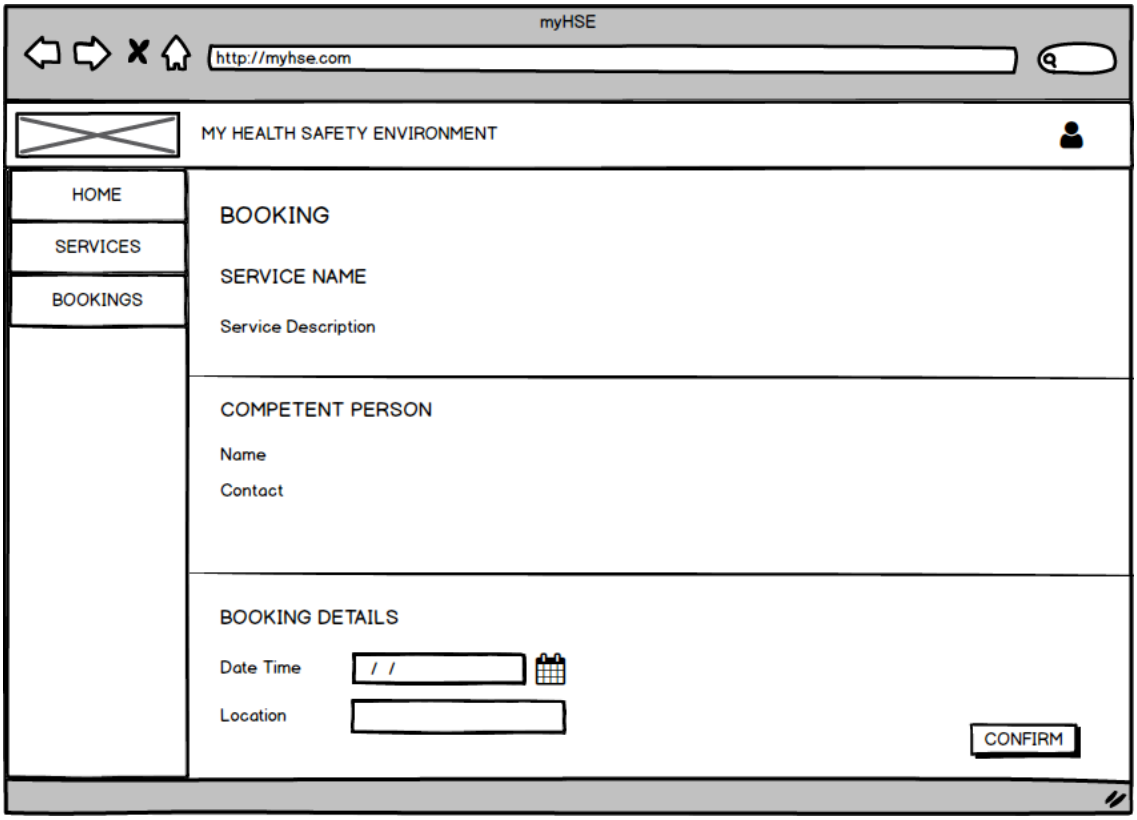



Figure 3.22 Subscriber Add Booking Page

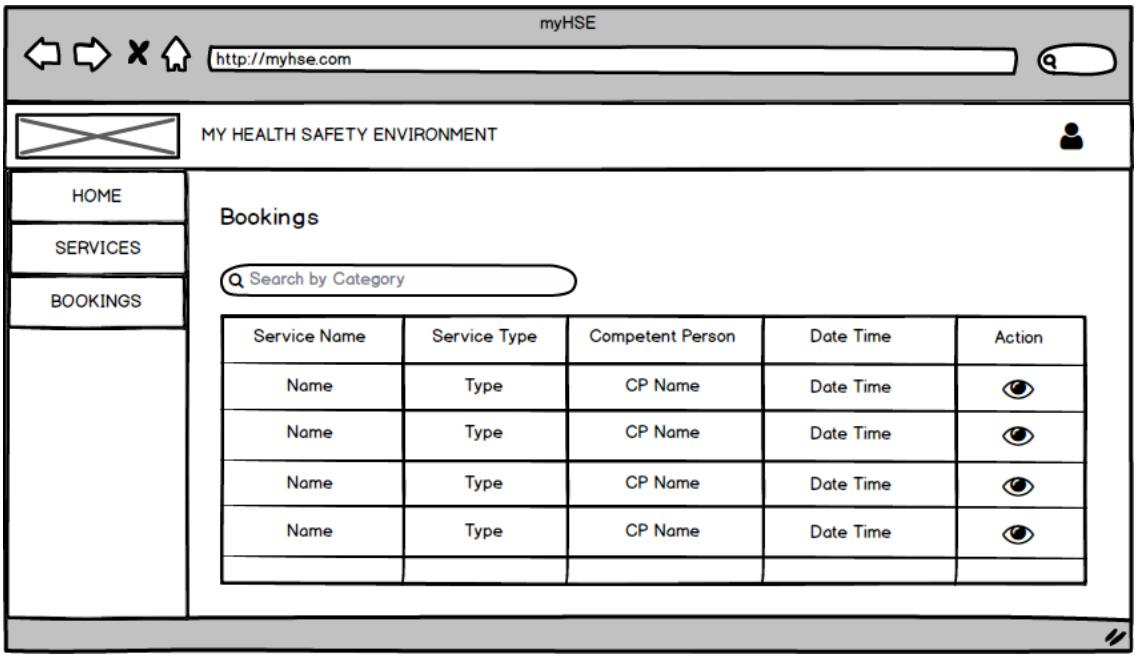



Figure 3.23 Subscriber View Bookings Page

Competent Person

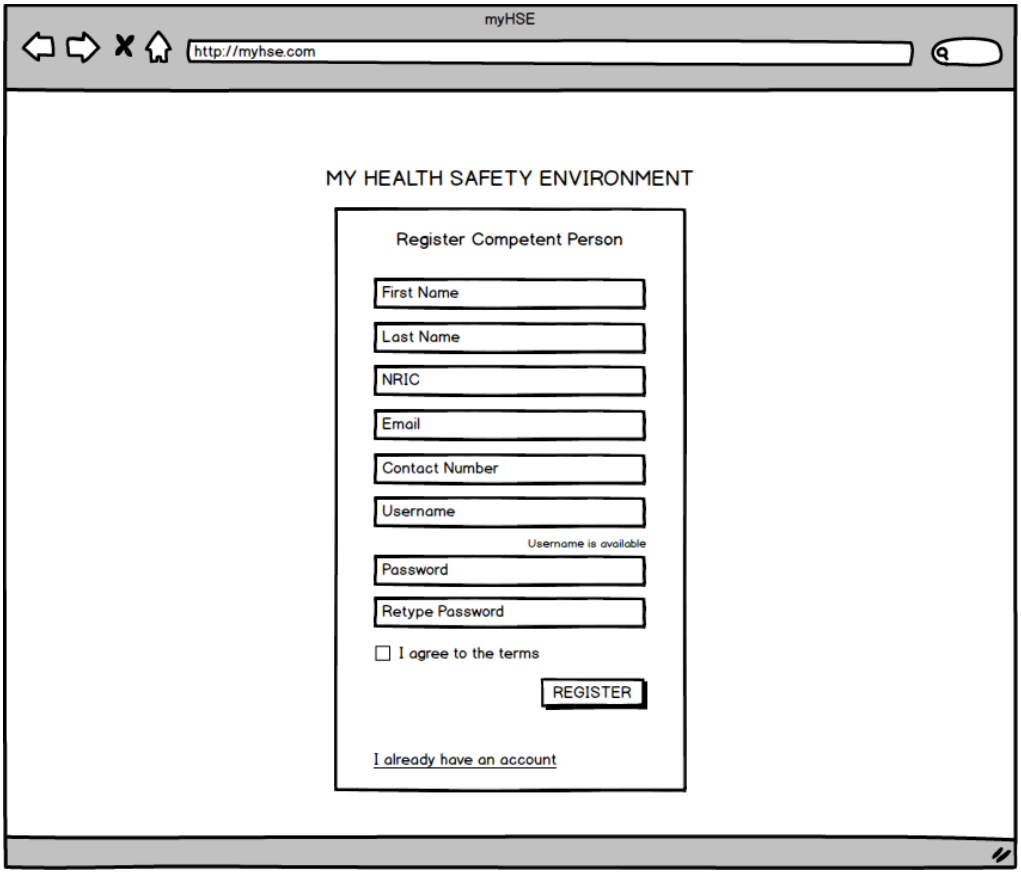



Figure 3.24 Competent Person Register Page


Figure 3.25 Competent Person Add Certificate Page

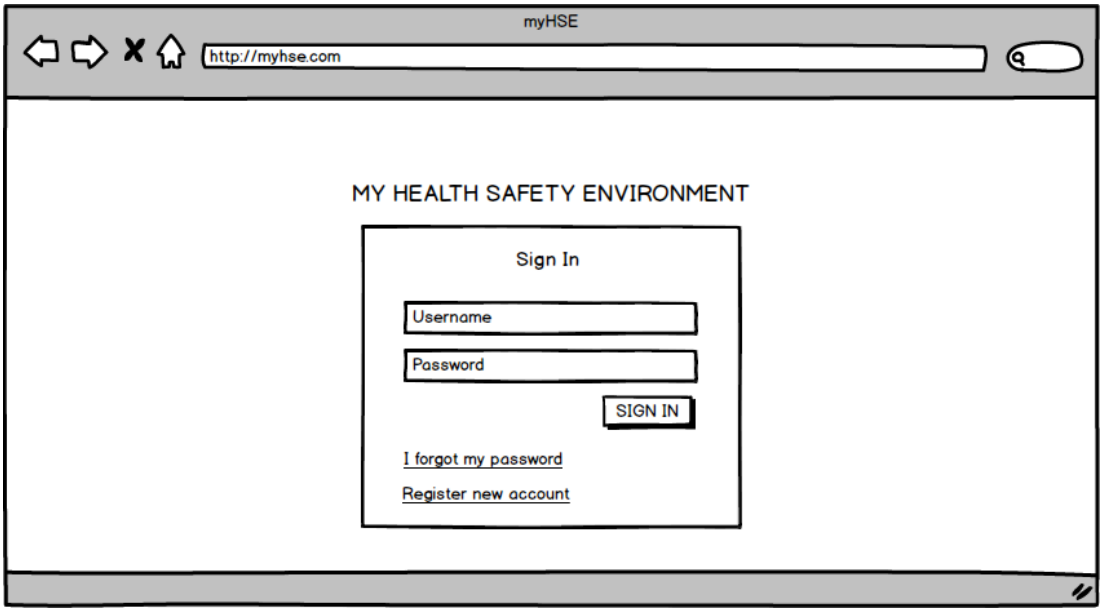



Figure 3.26 Competent Person Login Page

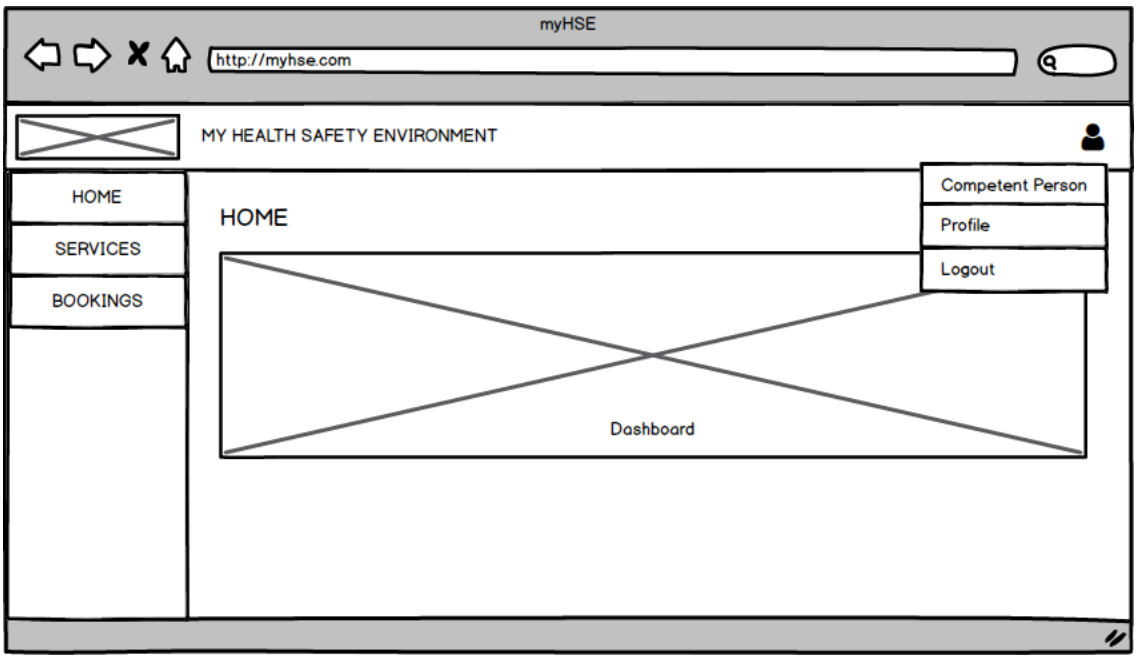



Figure 3.27 Competent Person Home Page

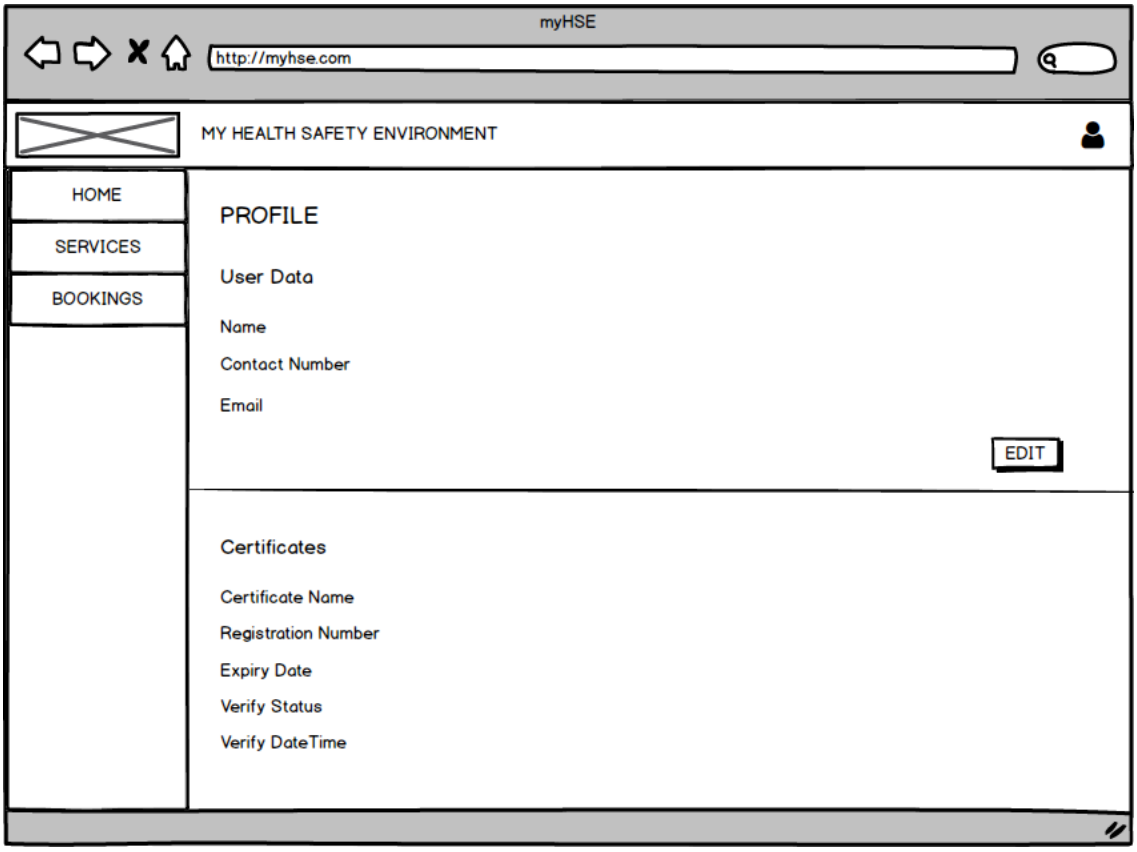



Figure 3.28 Competent Person Profile Page


Figure 3.29 Competent Person View Services

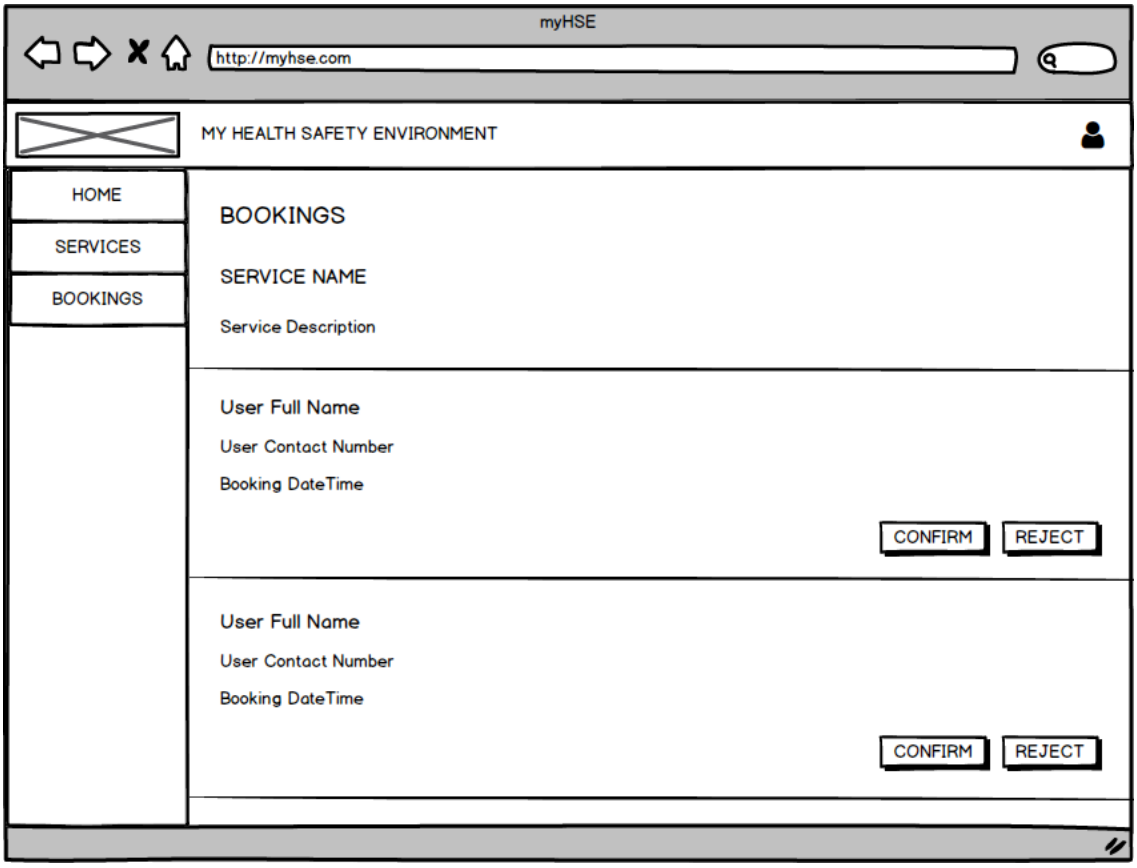



Figure 3.30 Competent Person View Bookings

### 3.6.3 Administrator

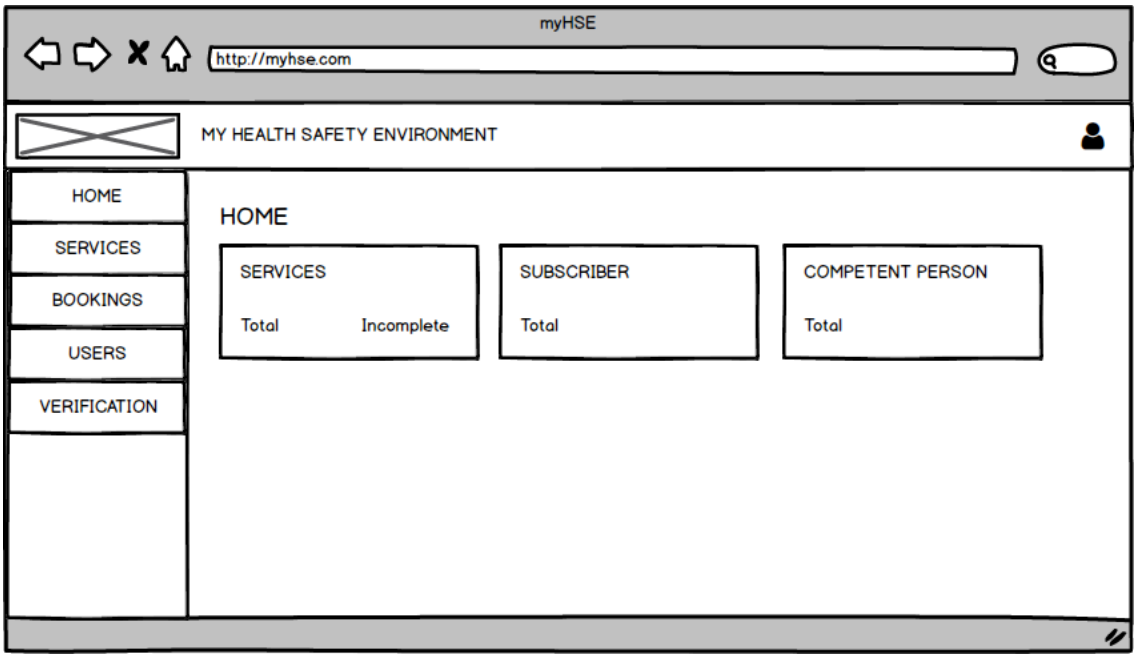



Figure 3.31 Administrator Home Page

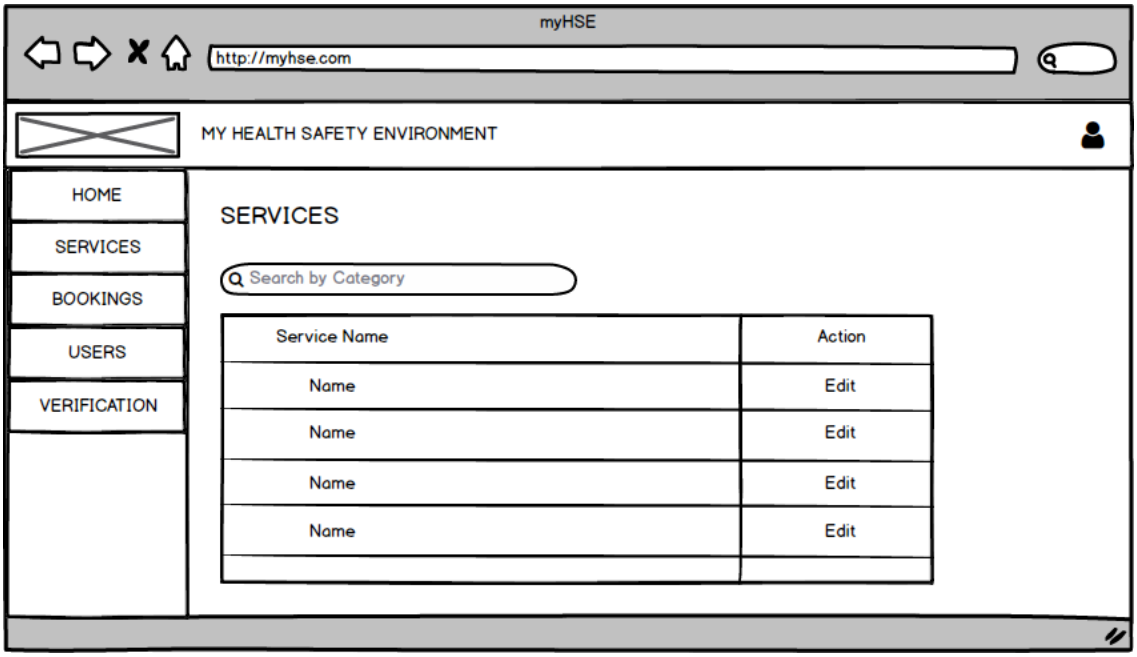



Figure 3.32 Administrator Services Page

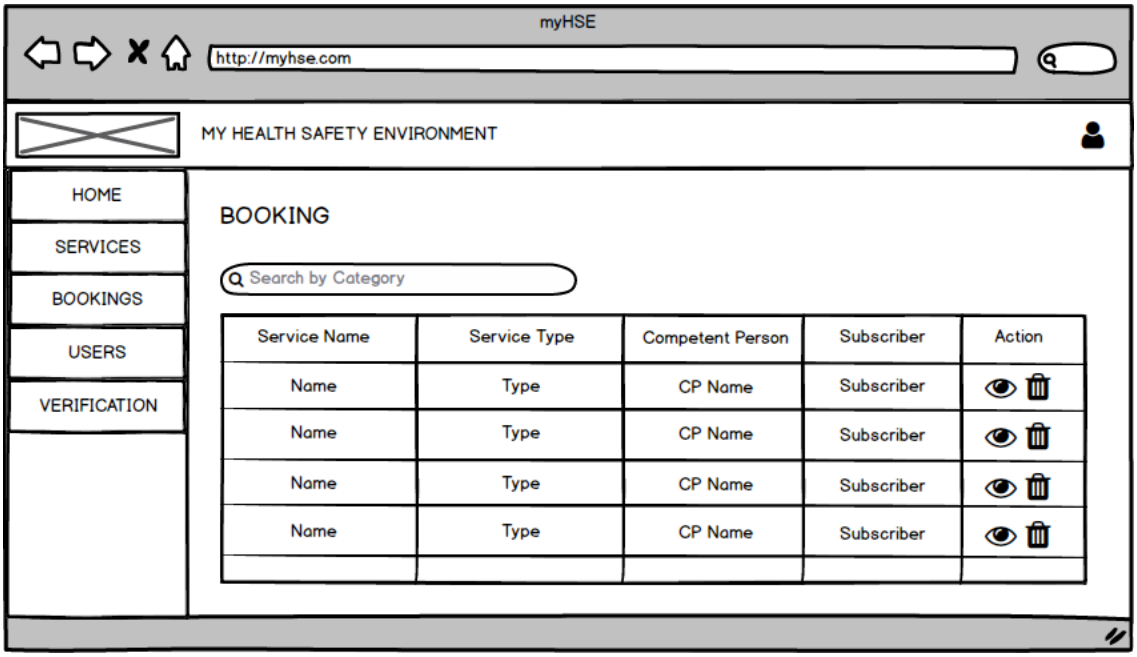



Figure 3.33 Administrator Booking Page

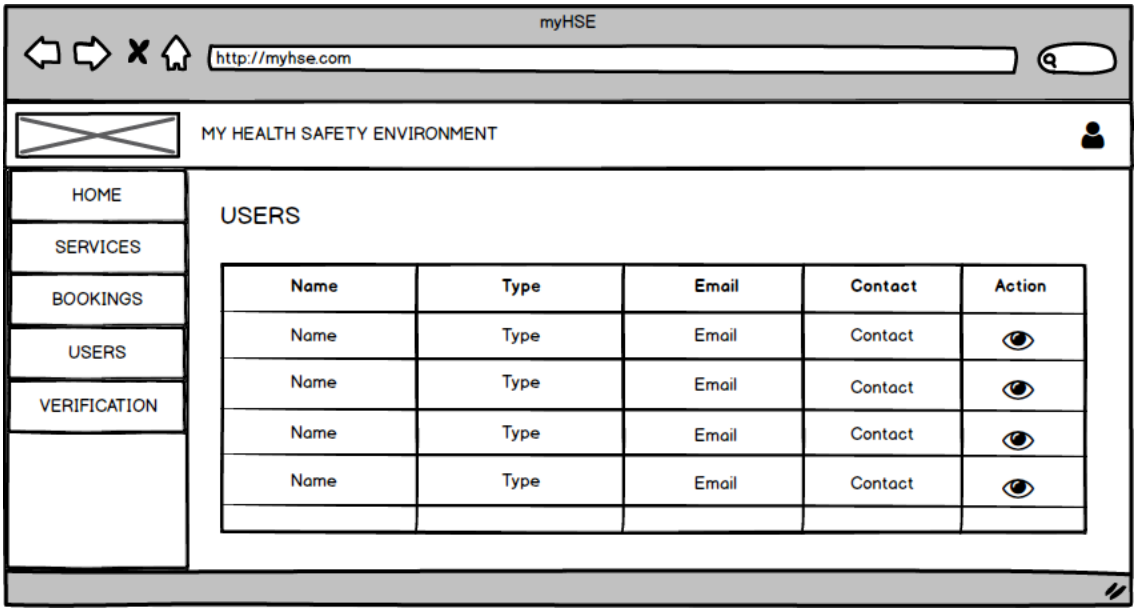



Figure 3.34 Administrator Users Page

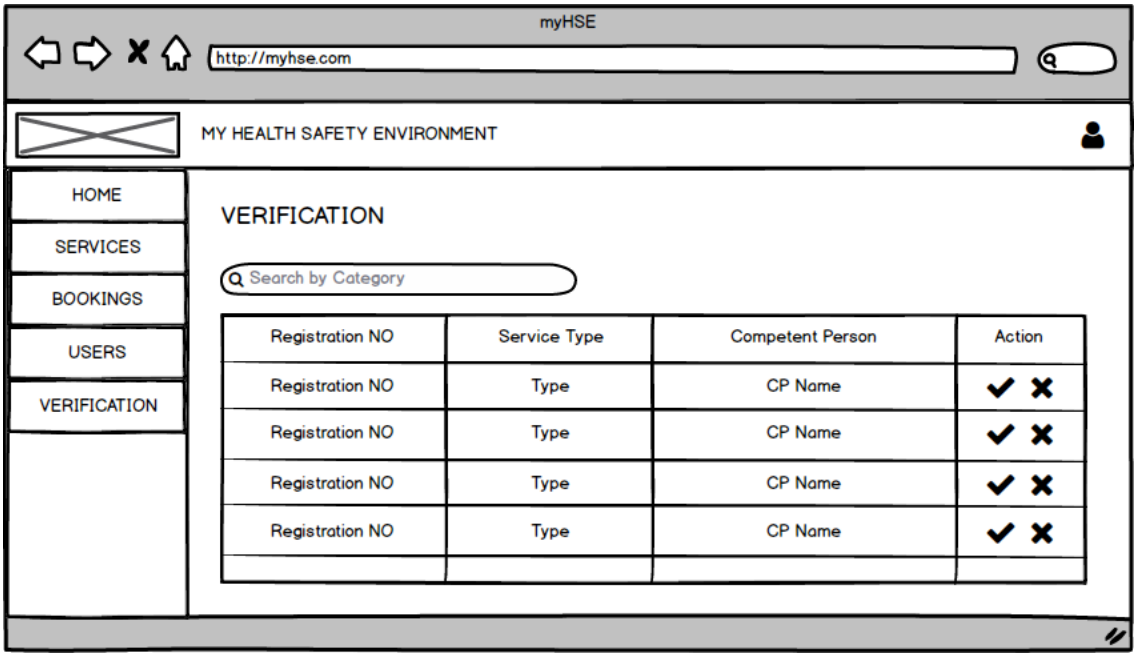



Figure 3.35 Administrator Verification Page

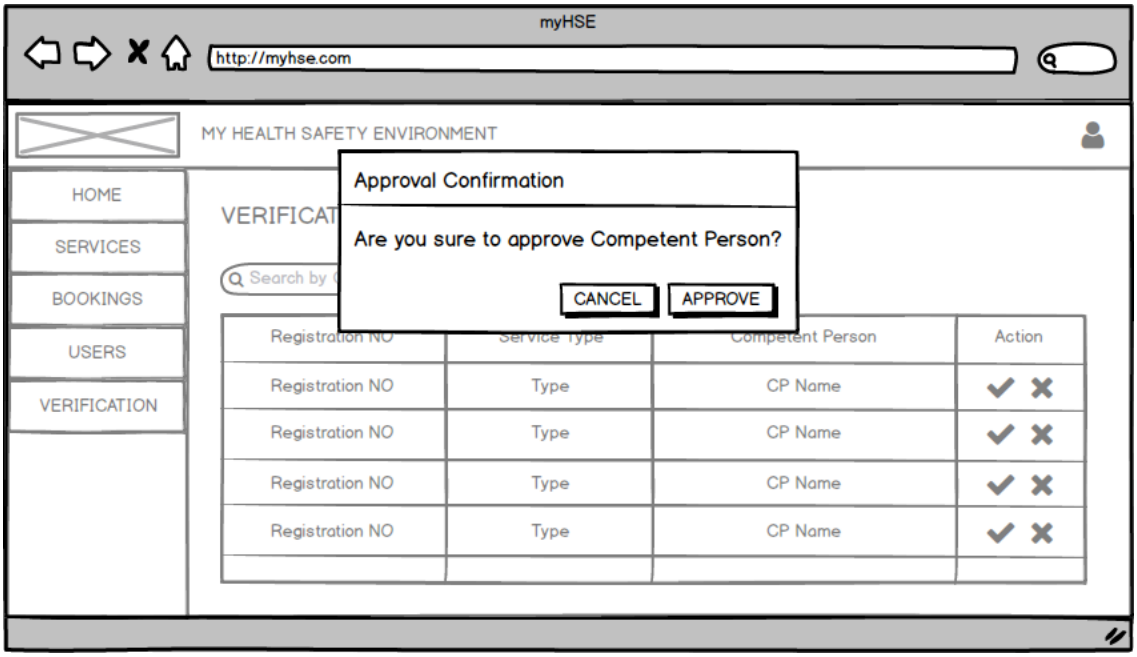



Figure 3.36 Administrator Verification Approval Page

## 3.7 Testing/Validation Plan

The following activities would be included in a testing or validation plan for the proposed software system using web responsive:

1. Data selection for testing

   To test the functionality of the proposed solution, a sample set of data will be selected from the system database. This data will include a variety of scenarios, such as different types of incidents, user roles, and data entry levels.

2. Testing implementation

   The proposed software will be tested using a combination of manual and automated testing. A team of users will perform manual testing on the system's functionality and usability by performing tasks such as data entry, reporting, and data retrieval.

3. User acceptance testing

   This test will be performed to assess the system's usability and user-friendliness. To evaluate the effectiveness of the proposed solution, comparison testing will be performed with other similar systems on the market. The sample of questions in the UAT form is shown in Appendix A.

## 3.8 Potential Use of Proposed Solution

MyHSE, the proposed solution, is a centralised OSH management system aimed at providing a one-stop platform for Occupational Safety and Health (OSH) services. The system's goal is to improve the efficiency and accessibility of OSH services for both providers and subscribers.

Companies and organisations can use MyHSE in real-time to manage their OSH compliance requirements. Guest users can view all available services and register for an account on the system. Competent person or service providers can use the system to manage their profiles, upload certificates and registration numbers, and upload services. They can also view service bookings and communicate with potential customers. The system allows subscribers to view all available services, book services and communicate with service providers. Administrators can also use the system to verify competent person, manage services, manage bookings and profiles.

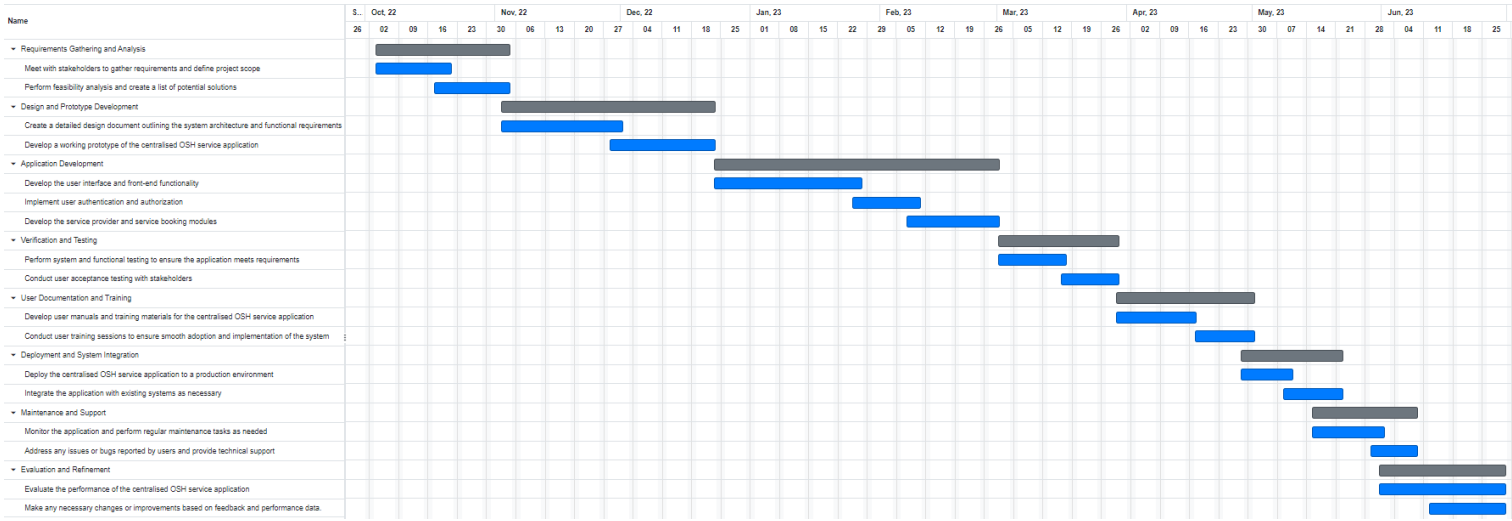
MyHSE provides the following advantages:

- Companies and organisations will benefit from increased efficiency in OSH compliance management
- Subscribers have easy access to OSH services
- Service providers' and their services' visibility has increased
- Communication between service providers and subscribers has improved

The proposed solution has the potential for commercialization because it can be offered as a service to companies and organisations to manage their OSH compliance requirements. Subscription and service fees can also be used to generate revenue.

Finally, MyHSE aims to provide a comprehensive and user-friendly platform for OSH services that can improve the efficiency and accessibility of OSH compliance management for businesses and organisations, as well as a platform for service providers to showcase and increase the visibility of their services

## 3.9 Gantt Chart



# CHAPTER 4

# IMPLEMENTATIONS, RESULTS AND DISCUSSION

## 4.1 Introduction

The implementations and outcomes of the MyHSE system will be covered in this chapter. With the help of illustrations and in-depth descriptions based on the system's previously mentioned functionalities, it is intended to provide a thorough explanation of the system's interfaces and how they operate. Functional testing will be used to confirm and validate all functionalities before the MyHSE system development phase is complete. Any system flaws and bugs will be fixed during this phase. Tests on user acceptability will be conducted to gauge user satisfaction and ascertain what users anticipate from the system in order to make sure it meets their needs. Future MyHSE system development will take into account user suggestions and feedback. The system will be launched to the general public and deployed to the server during the last stage of development.

## 4.2 Implementation Process

Laravel, Bootstrap, and MySQL were used in conjunction to implement the MyHSE system. The development of a practical and user-friendly centralised OSH service application required a number of steps during the implementation process. The following are the implementation's main components; framework selection, user interface design, database management, functionality implementation, testing and debugging.

### 4.2.1 Development of Environment

Due to its adoption of the Model-View-Controller (MVC) architectural pattern, Laravel was chosen as the framework for the MyHSE system. This method creates a more structured and manageable codebase by separating the application's logic, data management, and presentation layers. Developers can take advantage of pre-built features

like routing, session management, and authentication thanks to Laravel's robust ecosystem, greatly streamlining the development process. The framework's Eloquent Object-Relational Mapping (ORM) streamlines database interactions by offering a simple syntax for object-oriented principles-based database queries.


Figure 4.1 Laravel Framework

Due to its responsive design capabilities and pre-styled UI components, Bootstrap was chosen for the front-end design of the MyHSE system. The application adapts to different screen sizes with ease thanks to Bootstrap's responsive grid system, which improves user experience on various devices. Additionally, developers can create a consistent and aesthetically pleasing user interface using Bootstrap's extensive component library, which provides a wide range of pre-built and customizable UI elements. The front-end design of the MyHSE system can be made to conform to the particular design specifications by modifying Bootstrap's default styles and components.

Given its benefits in modularity, code reuse, and maintainability, object-oriented programming (OOP) was chosen over structured programming. Modular code structures are encouraged by OOP principles like encapsulation, inheritance, and polymorphism, which make it simpler to comprehend, maintain, and expand the application. Code readability and scalability are improved by OOP's emphasis on code organisation and the separation of concerns. Additionally, OOP encourages code reuse, enabling programmers to use preexisting code components in various application components or upcoming projects. As a result of OOP's modular design, testing is made easier, leading to higher-quality code and more dependable software in general.

The MyHSE system benefits from a well-structured and maintainable codebase, responsive design capabilities, and customizable user interface components by using Laravel as the framework and Bootstrap for front-end design. The system's modularity, reusability, and scalability are further improved by the adoption of OOP principles, encouraging effective development techniques and raising the level of the application's general quality.

### 4.2.2 Database Implementation

MySQL was selected as the database management system to be used for the MyHSE system's database implementation. For storing and retrieving data, MySQL provides a trustworthy and durable solution. It has a large user base and is renowned for its scalability, performance, and platform compatibility.

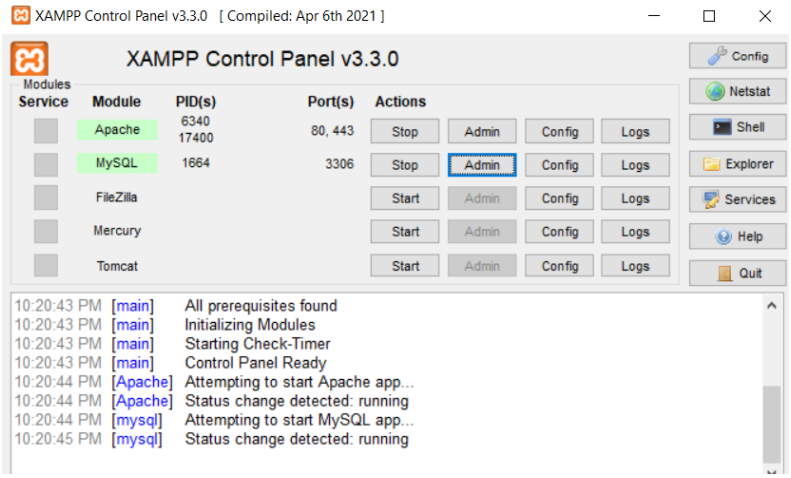



Figure 4.2 XAMPP Control Panel

XAMPP was used to set up a local server environment for the purposes of development. Popular software called XAMPP provides a full web development stack for local testing and development. It consists of Apache, MySQL, PHP, and Perl. Before being installed on a live server, the MyHSE system could be tested and improved locally using XAMPP.

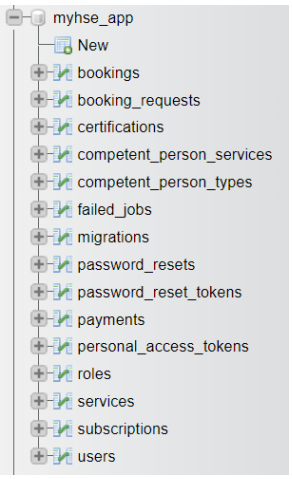



Figure 4.3 Database Tables of MyHSE System

The MyHSE system's data requirements were taken into consideration when designing and structuring the MySQL database. To store user information, service information, bookings and other pertinent data, tables were made. Primary and foreign keys were used to connect tables, ensuring data integrity and facilitating effective querying.

Eloquent ORM, which provides an expressive syntax for carrying out CRUD (Create, Read, Update, Delete) operations, was used to implement queries and database operations within the Laravel framework. To represent database tables and specify relationships between them, eloquent models were developed.

The MyHSE system was installed on a live server after the database design and implementation were finished, tested locally, and approved. The server environment was set up to support MySQL, ensuring that the application could store and retrieve data without any problems. This made it possible for users to access the system and communicate securely and effectively with the database.

During the implementation process, the database for the MyHSE system was successfully structured, tested, and deployed using MySQL and XAMPP. XAMPP provided a controlled environment for perfecting the system's functionality and guaranteeing data consistency during local server testing. The MyHSE system was prepared to handle actual data with the final deployment to a live server, giving users access to dependable and secure data storage.

## 4.3 System Interface

### 4.3.1 Guest

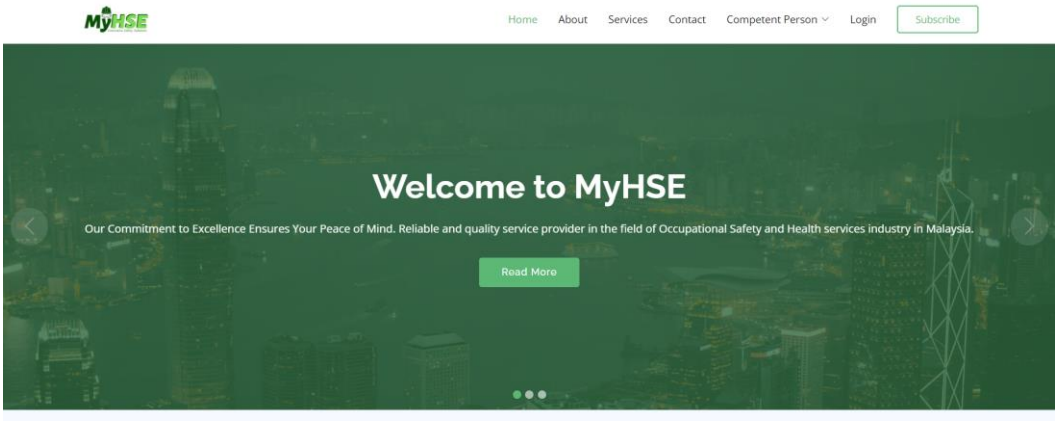



Figure 4.4 Guest Landing Page

The interface for guest users or public users who access the MyHSE system will be offered restricted access and focuses on providing company information and services. The first point of contact is the landing page, which presents pertinent information about the business and the variety of services offered. This background material aids visitors in comprehending the objectives and features of the MyHSE system.

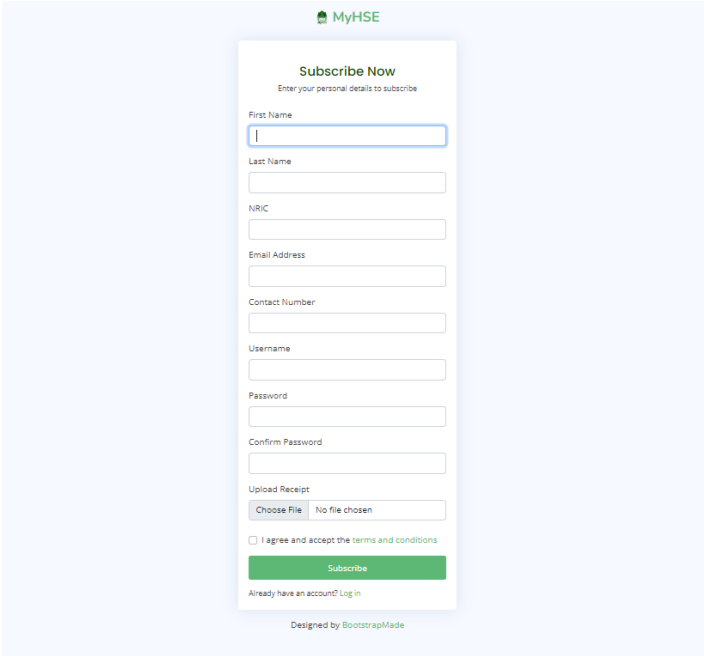



Figure 4.5 Subscription Form for Guest

The option to subscribe to the services offered is available to visitors who fill out a subscription form. The subscription form gathers important information like name, contact details, and preferred services. After the user submits the form, the system processes the data and directs them to a secure page.

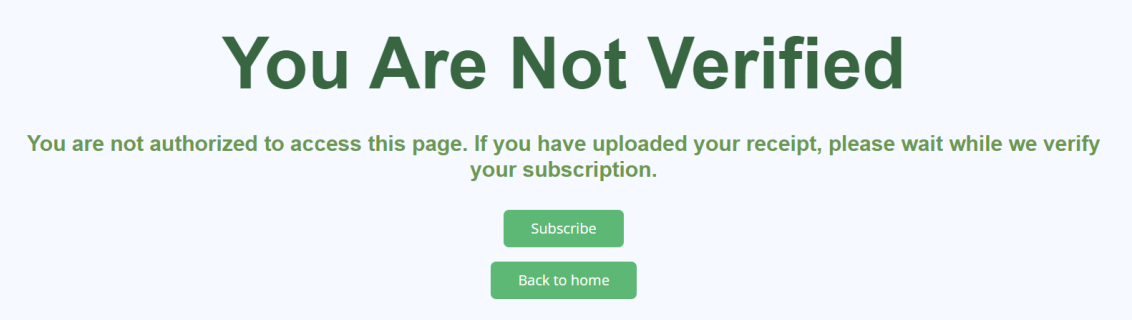



Figure 4.6 Restricted Page for Guest

Only subscribers have access to the restricted page, and they are unable to use the system's full capabilities until the administrator has verified their subscription. Only authorised users are granted access to the system's features thanks to this verification process. Guest users will get a confirmation message letting them know their subscription is being examined while they wait for verification.

### 4.3.2 Competent Person

The MyHSE system's Competent Person interface offers a platform for qualified users to register and gain access to particular features and functionalities. They are directed to a restricted page where they can upload the details of their certification after registering. Before granting access to the system, the admin will then check the submitted certifications.

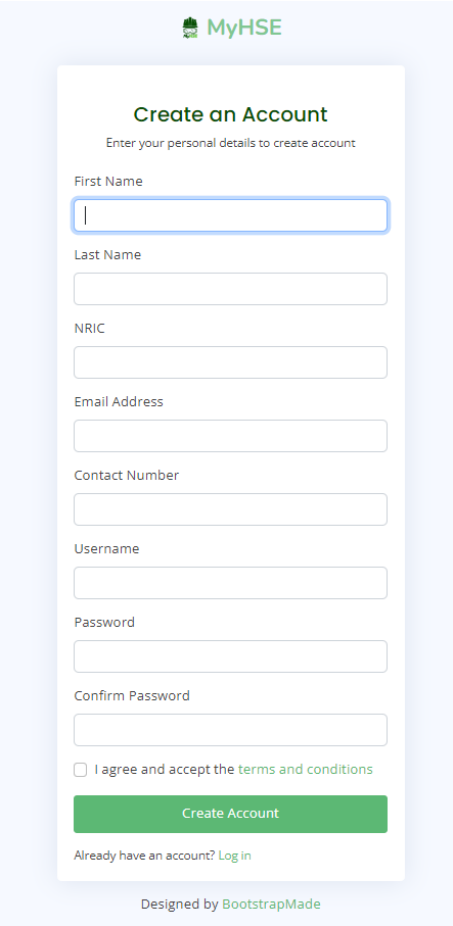



Figure 4.7 Competent Person Register Form

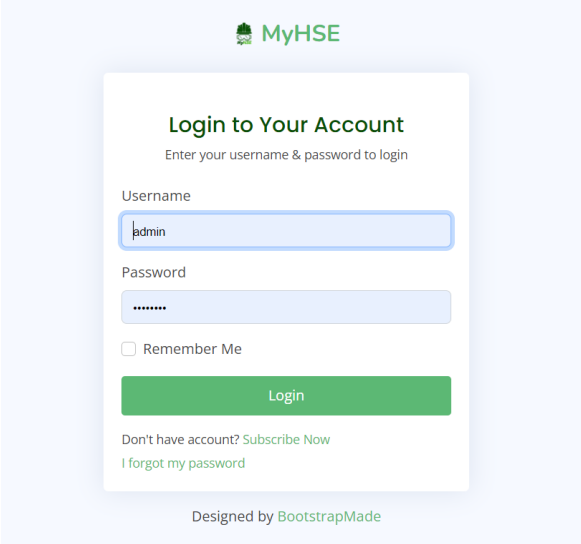



Figure 4.8 Login Page

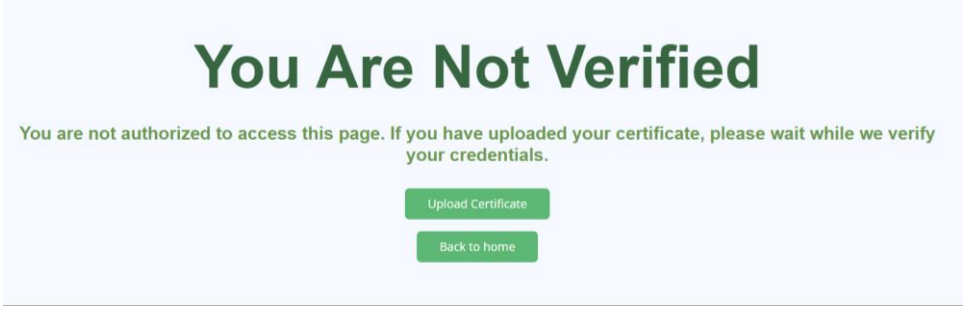



Figure 4.9 Restricted Page for Competent Person

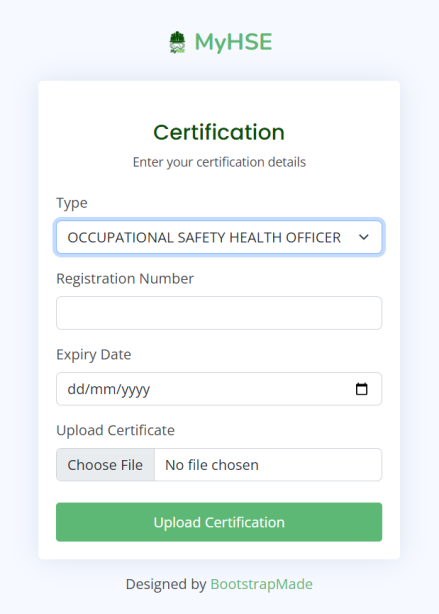



Figure 4.10 Upload Certification

Competent Persons can log in and are then taken to their unique dashboard after being verified. They can view information about their reservations and certifications on the dashboard, which acts as a central hub. They can keep track of their engagements and credentials thanks to this consolidated view.

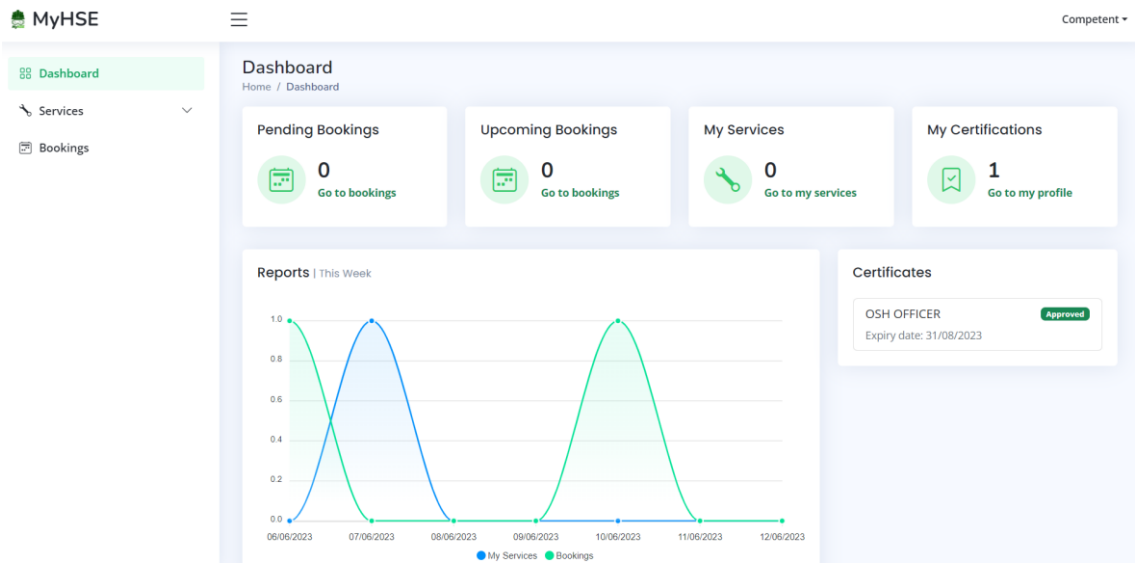



Figure 4.11 Competent Person Dashboard

Competent Persons can view and edit their personal information, including contact details and credentials, in the Profile section. Additionally, they can directly update or upload new certifications from the Profile page. In this section, they can also modify their password for even more security.

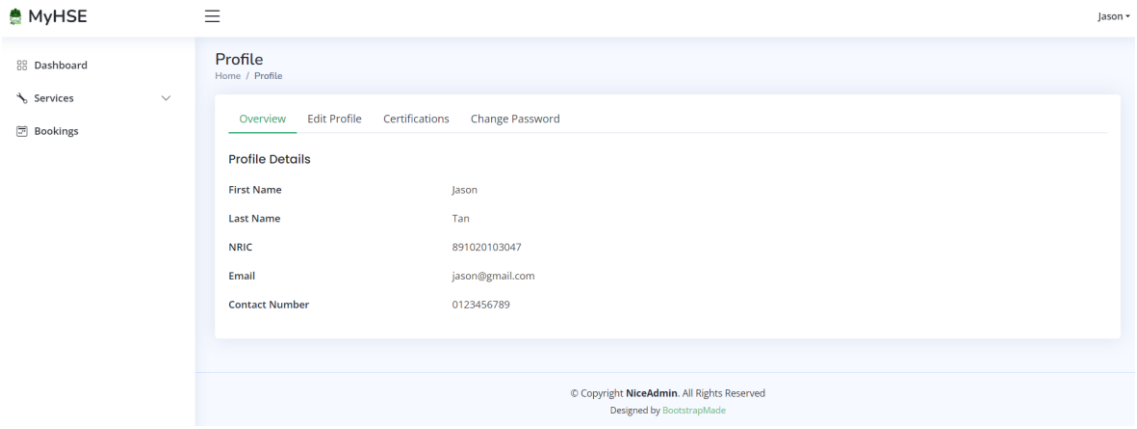



Figure 4.12 Profile Overview

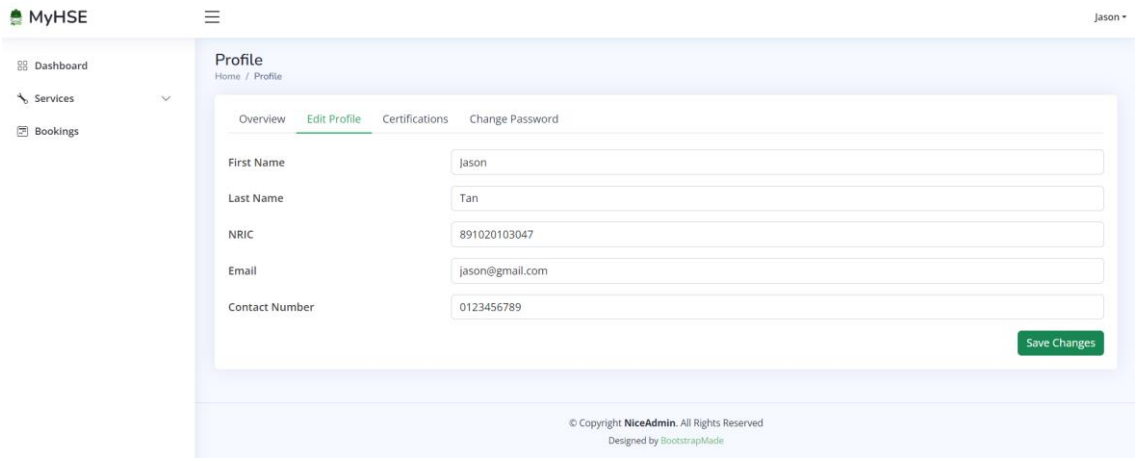



Figure 4.13 Edit Profile

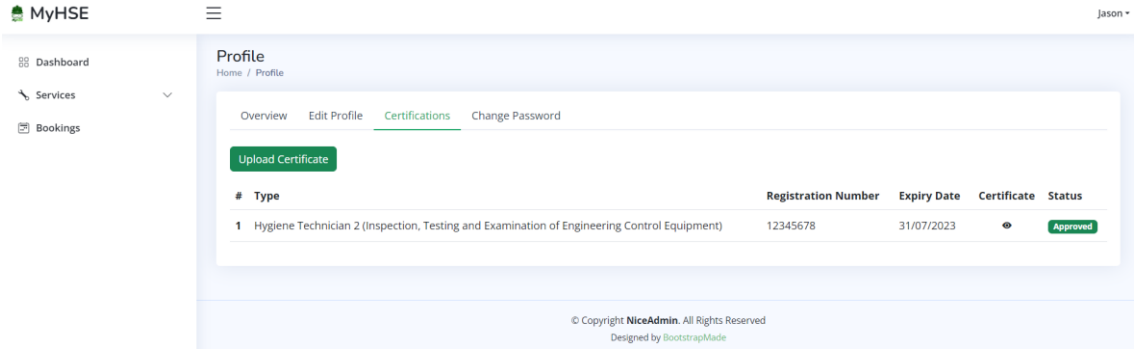



Figure 4.14 Certifications Page

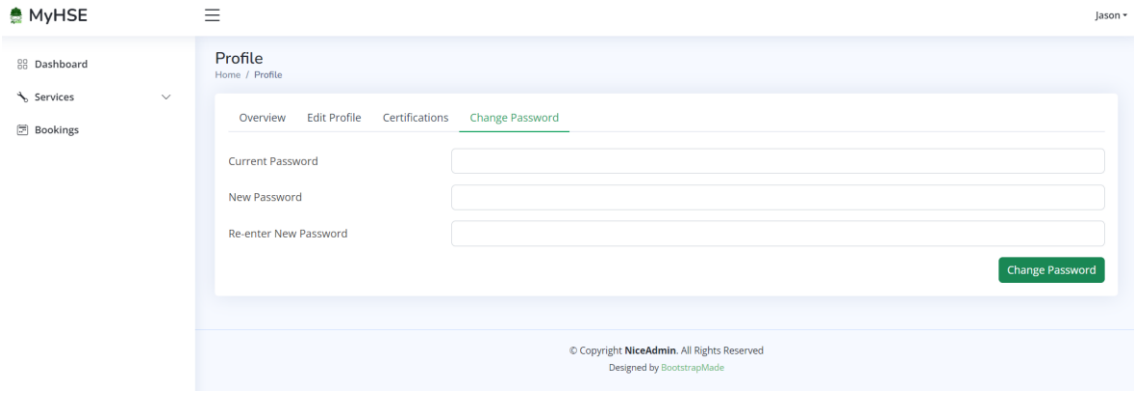



Figure 4.15 Change Password Page

Competent People are able to add their services to the system. They can enter information about their services, such as descriptions, specifications, and availability, in the Services section. Additionally, they have control over whether or not their services are visible to subscribers. Competent Persons can view a thorough summary of their services, including the precise information they offered, on the View page. This enables them to check and modify the information about their services as necessary.

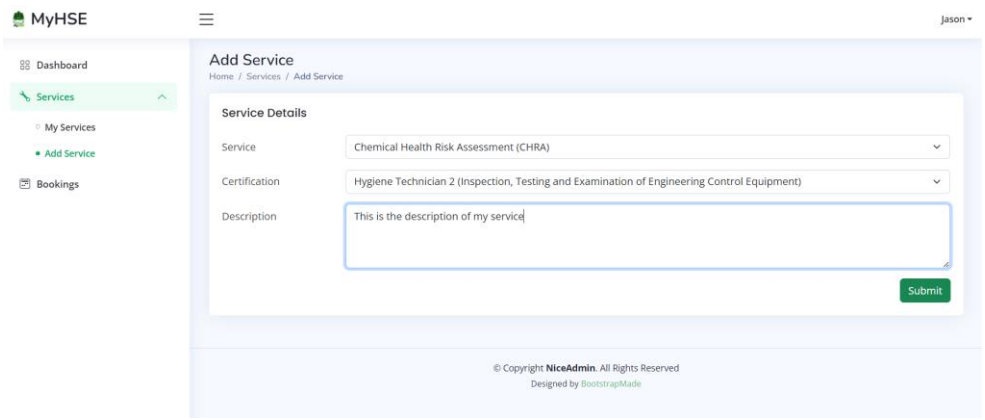



Figure 4.16 Add Service Page

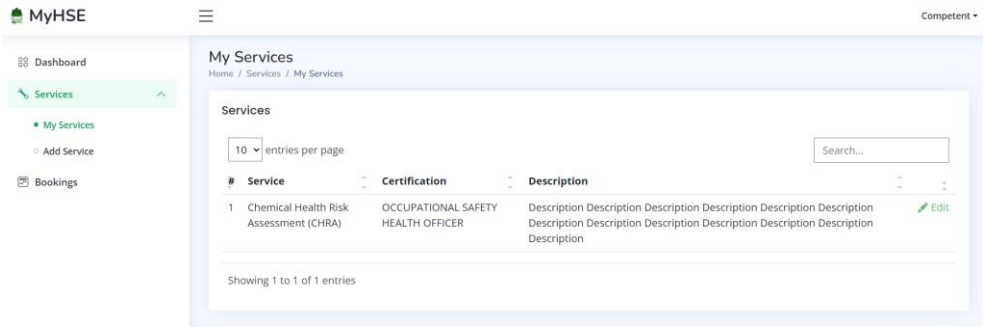



Figure 4.17 View Services Page

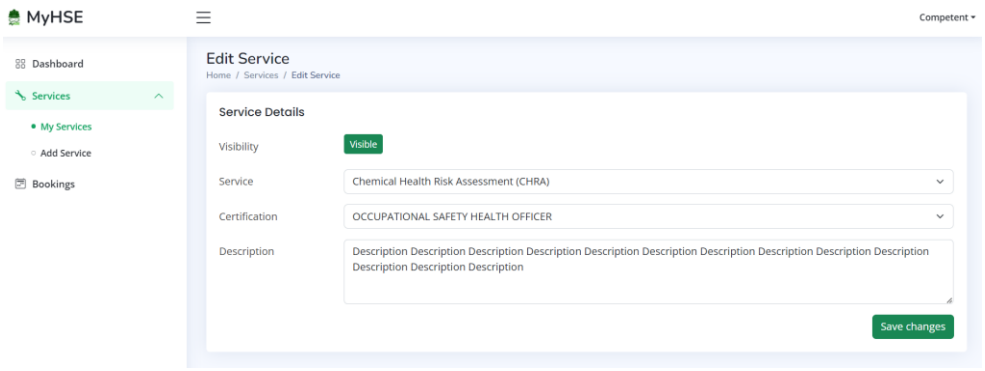



Figure 4.18 Edit Service Page

They can also keep track of subscribers' reservations for their services and take the appropriate action. They have the power to decide whether to accept or reject reservations and to decide how much each service will cost.

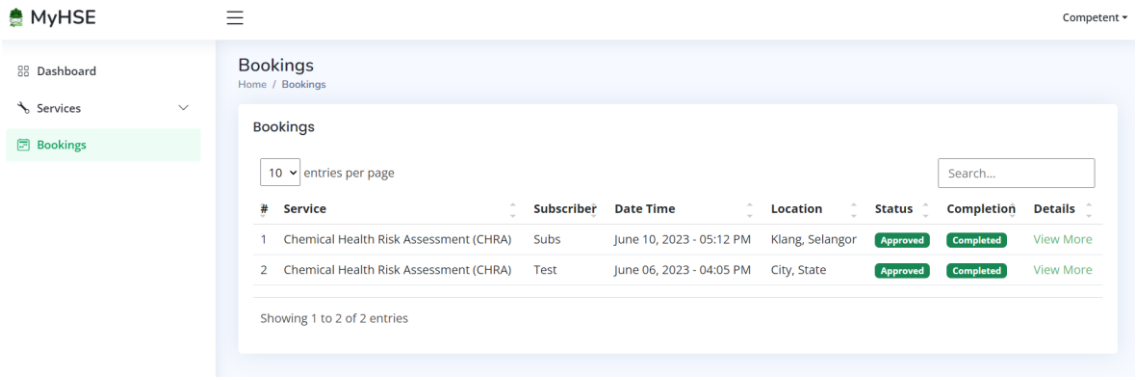



Figure 4.19 View Bookings Page

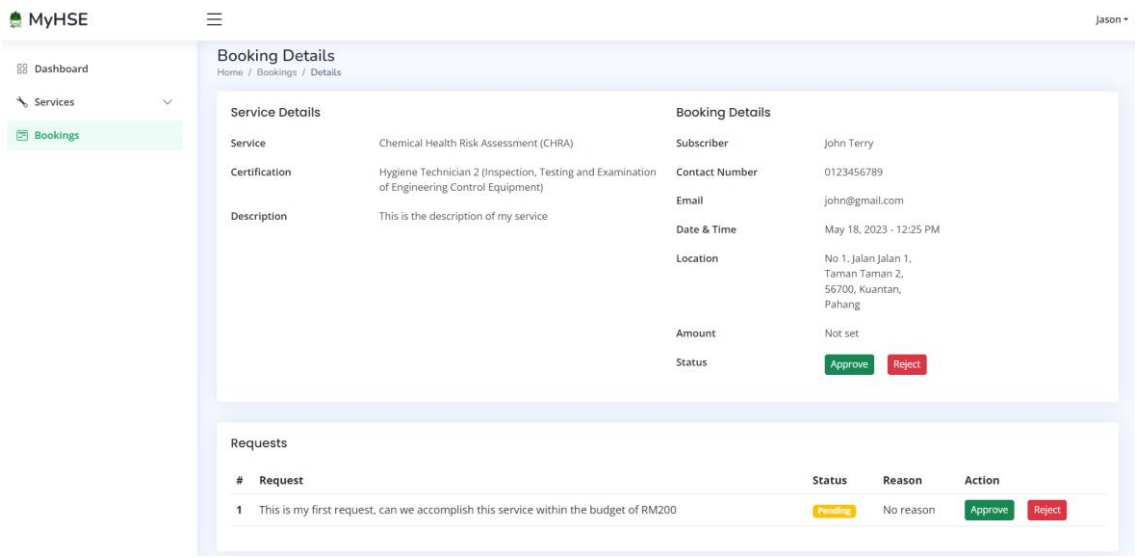



Figure 4.20 Booking Details Page

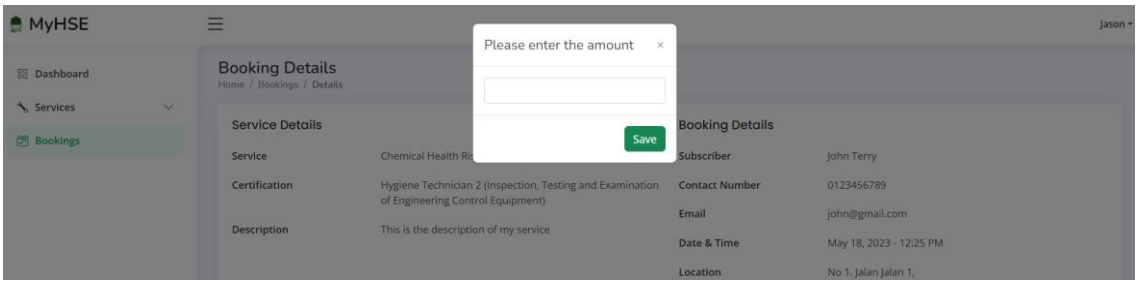



Figure 4.21 Set Amount Prompt Message

### 4.3.3 Subscriber

Registered users have a dedicated platform to access and manage their subscriptions and bookings through the Subscriber interface in the MyHSE system. Subscribers are directed to their individual dashboard after logging in using the same login page as the Competent Person. Subscribers can view their subscription's details, including its validity and any bookings they have made, in the dashboard. They can easily see their subscription status and upcoming engagements thanks to this.

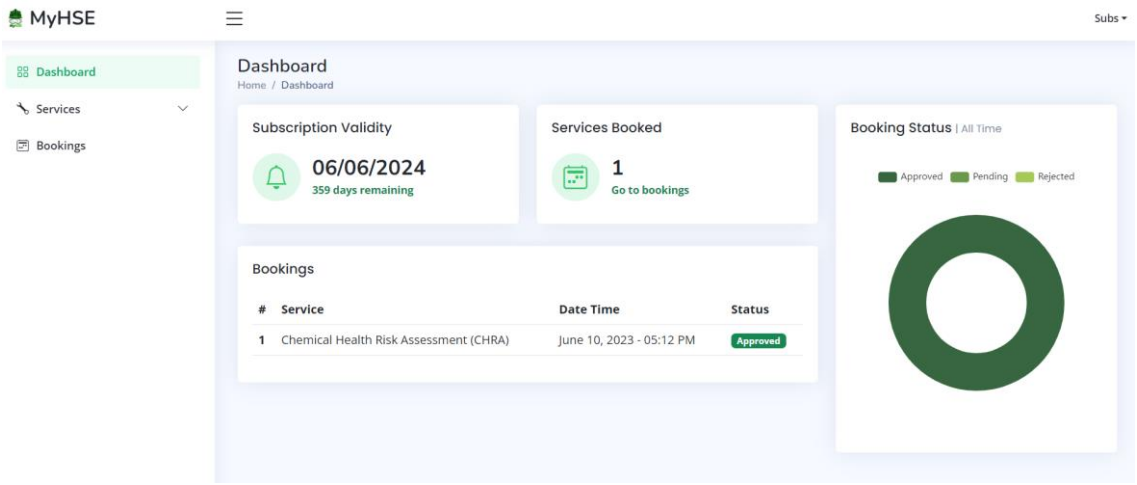



Figure 4.22 Subscriber Dashboard

Subscribers have access to view and edit their profile information in a manner akin to the Competent Person interface. They can change their contact information, preferences, and other pertinent data. Subscribers can access their subscription history rather than certifications. They can look over their past subscription information in this section to see how engaged they have been with the system.

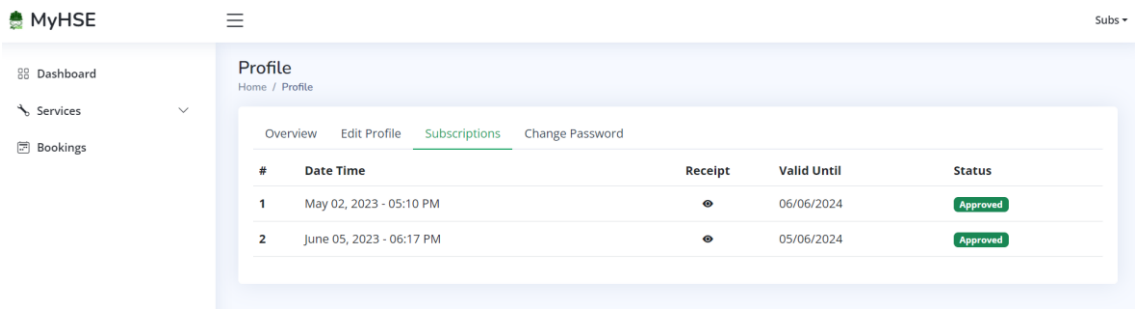



Figure 4.23 Subscriptions

Subscribers can browse and view all services offered by Competent Persons in the Services section. They can look over the availability, requirements, and descriptions of the services. Subscribers can book a service by clicking the "Book" button, which takes them to a booking form. The Competent Person assigned to the service will receive the booking request once the booking form has been submitted. After reviewing the request, the competent person will take the appropriate action.

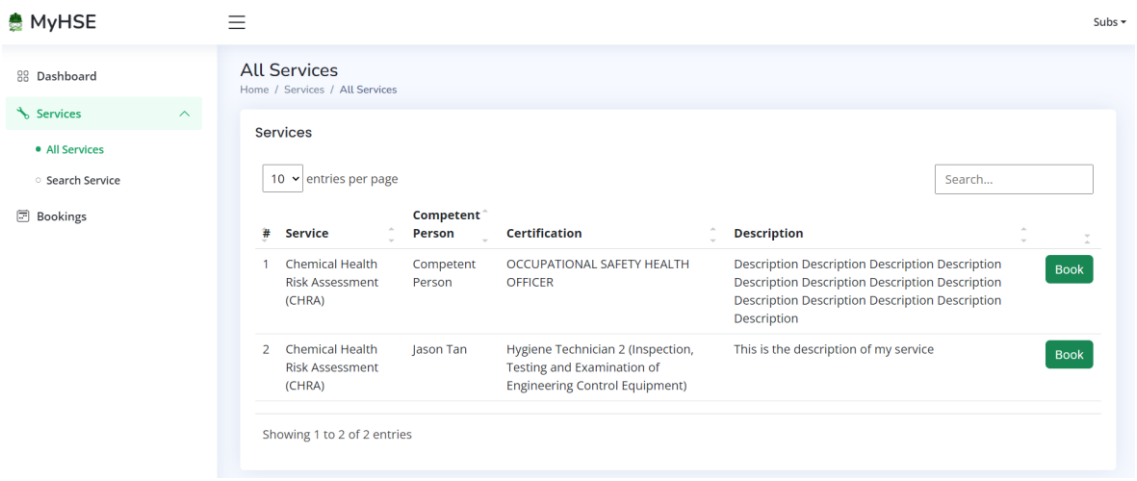



Figure 4.24 Display all Services Page

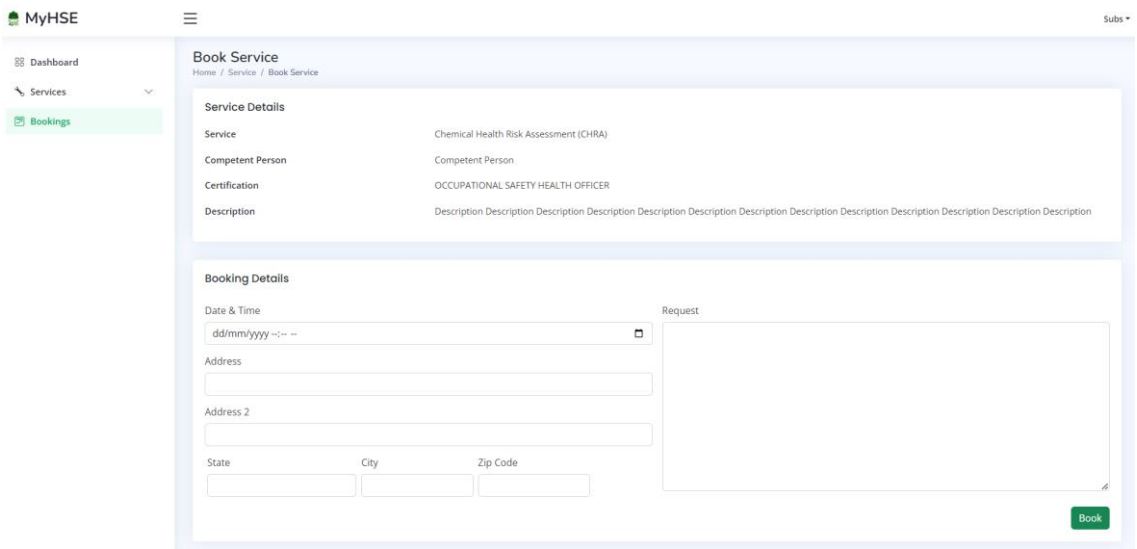



Figure 4.25 Book Service Form

Subscribers have the option to look for particular services in the Search tab based on their preferences. In order to assist Subscribers in locating services that meet their needs, the system will display pertinent services if they are available. Subscribers can add special requests or other relevant details when viewing the details of a specific booking. This enables them to express any particular needs or preferences they may have.


Figure 4.26 Search Service Page


Figure 4.27 Search Result Page

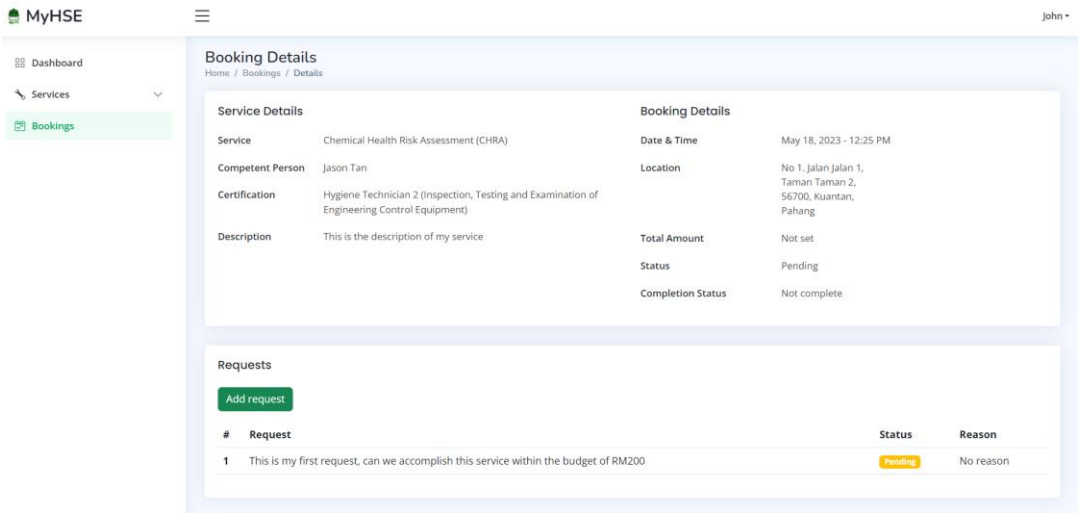



Figure 4.28 Booking Details Page

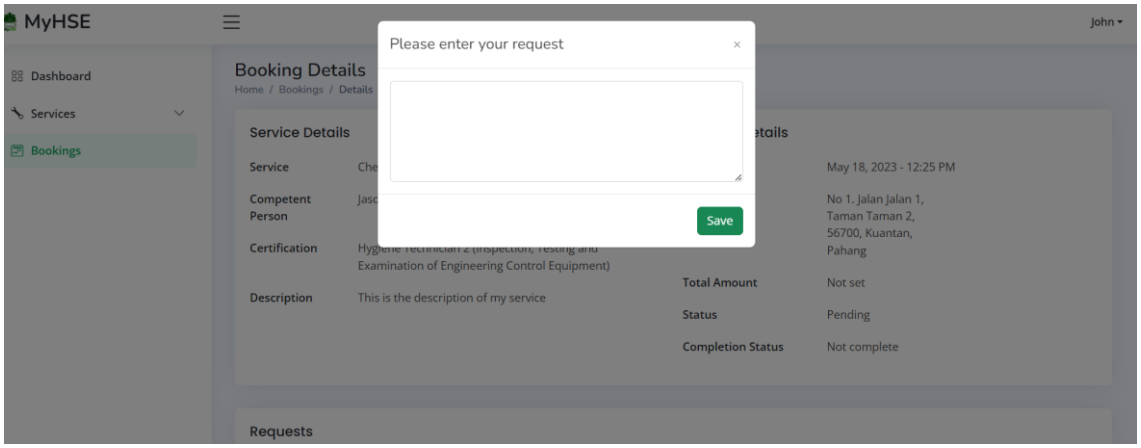



Figure 4.29 Add Request Prompt Message

Subscribers must make the first payment if the Competent Person accepts the booking request. Subscribers are required to make the last payment to cover the outstanding balance once the service is over.

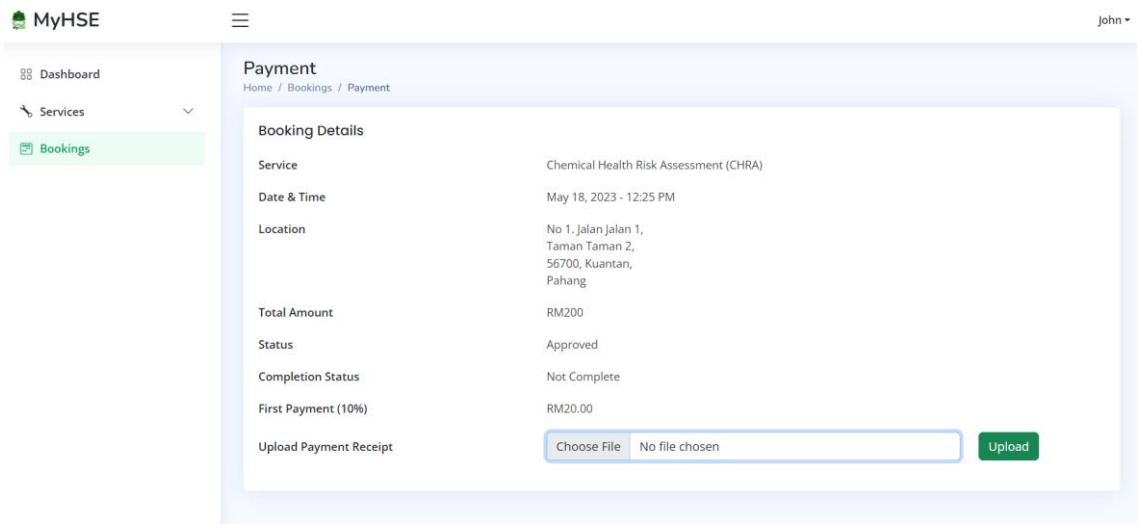



Figure 4.30 Add First Payement Page

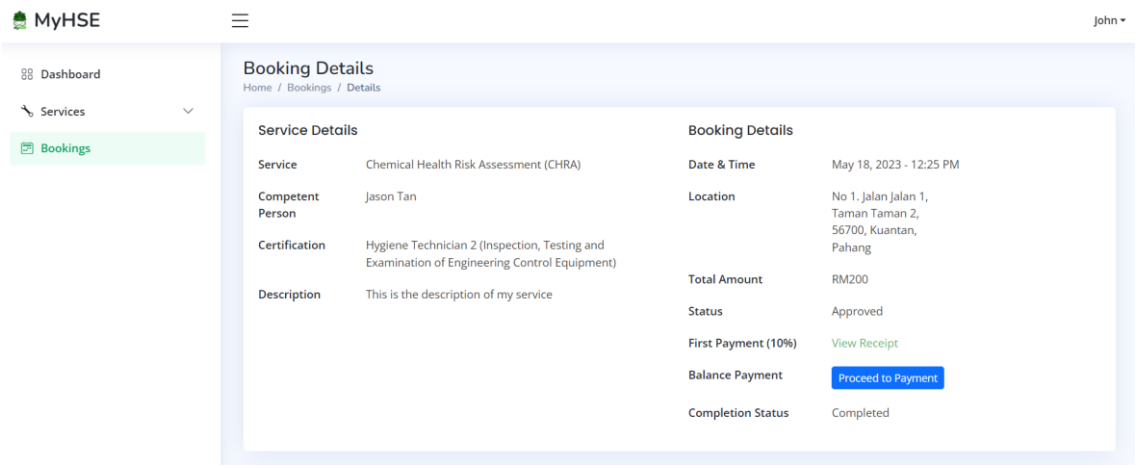



Figure 4.31 Proceed to Balance Payment

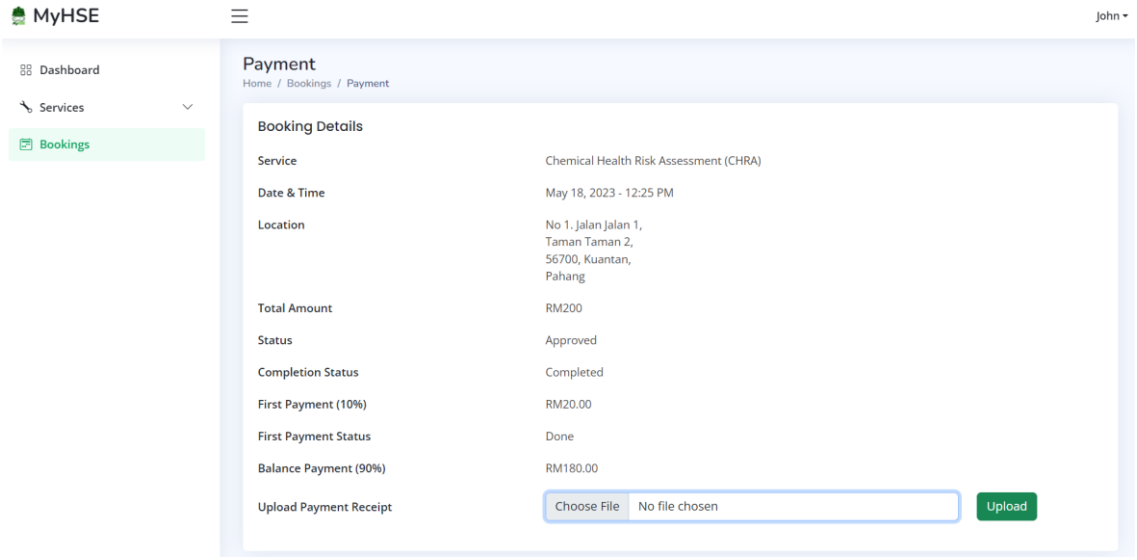



Figure 4.32 Add Balance Payment Page

### 4.3.4 Administrator

The MyHSE system's Administrator interface offers a complete set of tools and functionalities for controlling the system's operations. Similar to Competent Persons and Subscribers, Administrators can log in using the same login page, and after successfully doing so, they are taken to the admin dashboard. Administrators have access to a graph showing the bookings and services used over the previous week within the admin dashboard. This graphical display gives a brief summary of system activity and performance. Administrators have the same profile editing and viewing capabilities as other user types. They can change their contact information, preferences, and other pertinent data.

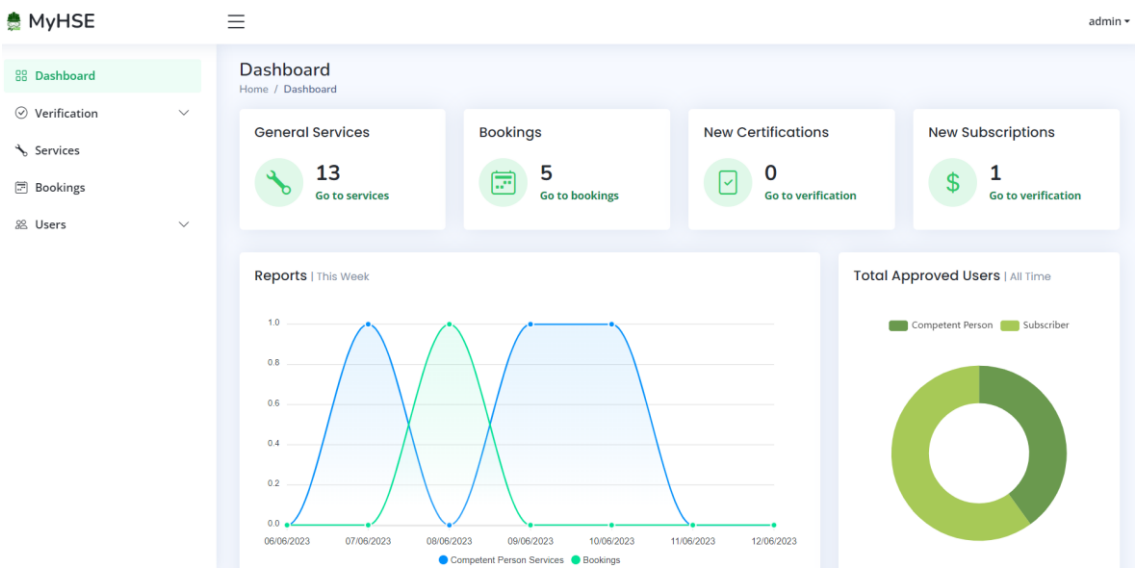



Figure 4.33 Administrator Dashboard

Administrators are essential in confirming new Competent Person certifications and Subscriber subscription requests. Administrators examine the submitted certifications and subscription information during the verification process. The certifications or subscriptions can then be approved or rejected, which causes a confirmation message to be sent to the appropriate users. This guarantees that the system's features and services are only available to verified and legitimate users.

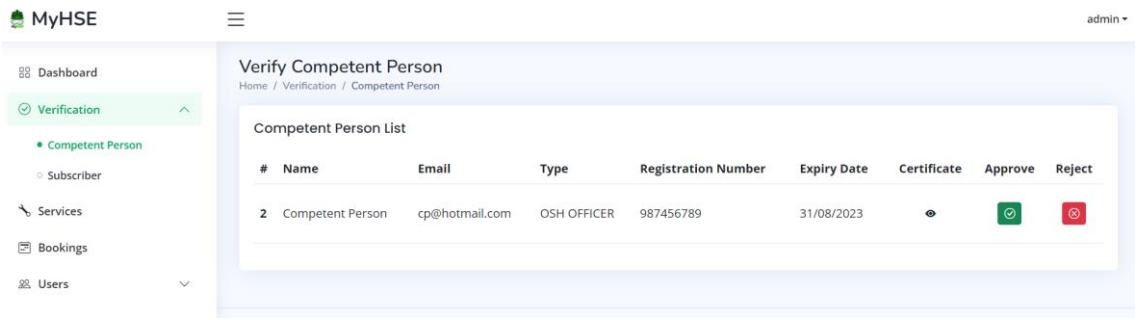



Figure 4.34 Verify Competent Person Page

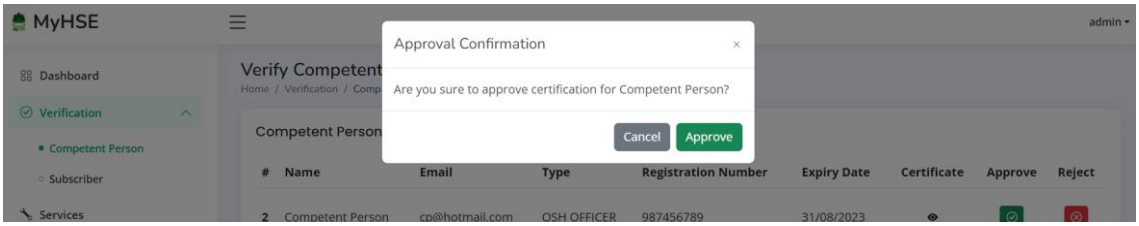



Figure 4.35 Approval Confirmation Message

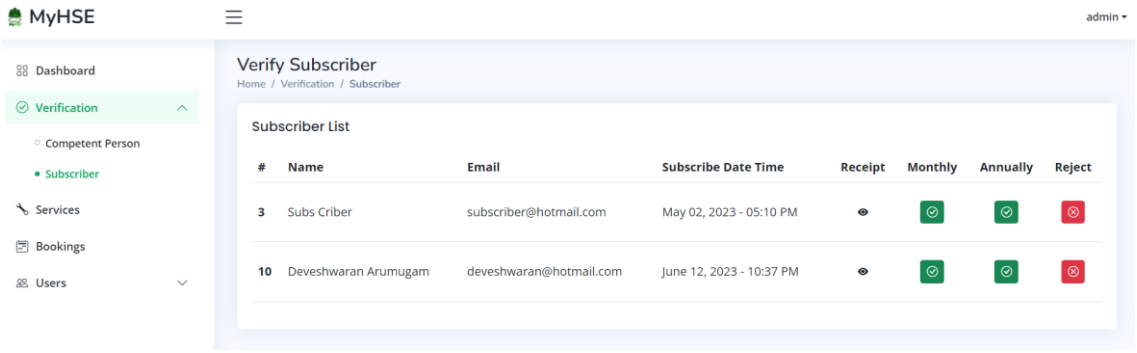



Figure 4.36 Verify Subscriber Page

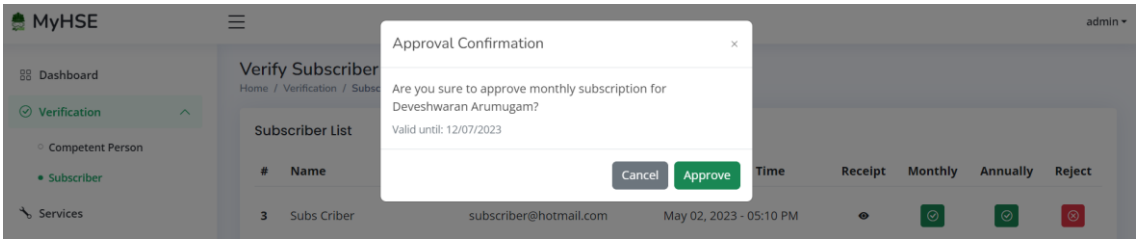



Figure 4.37 Approval Confirmation Message

Administrators have the ability to view and modify the general service details in addition to user verification. This includes the capability to alter service requirements, availability, and description. Administrators can guarantee that users have access to accurate and current information by maintaining control over the service specifics.

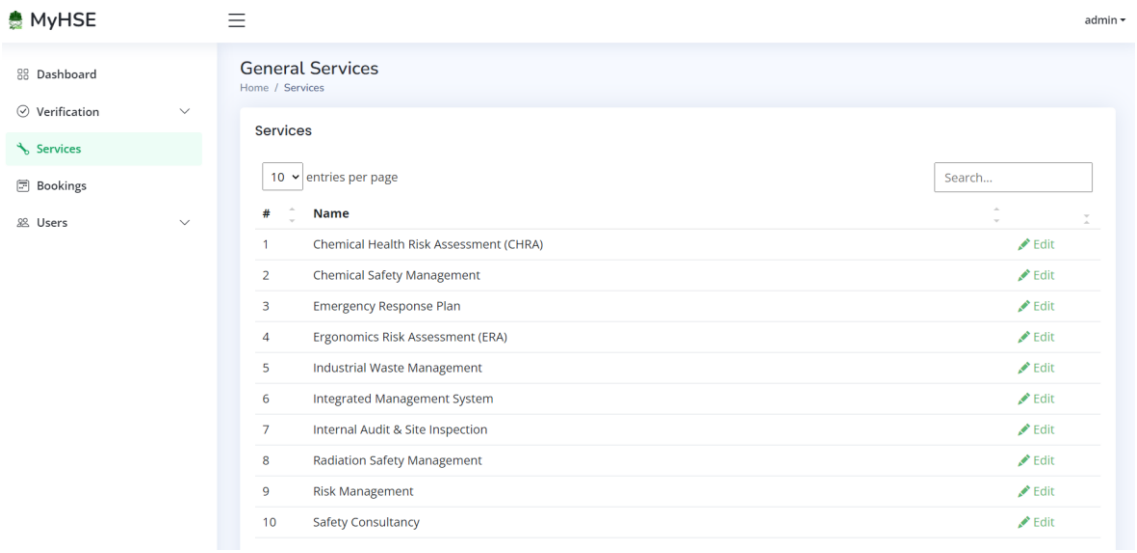



Figure 4.38 Display all Services Page

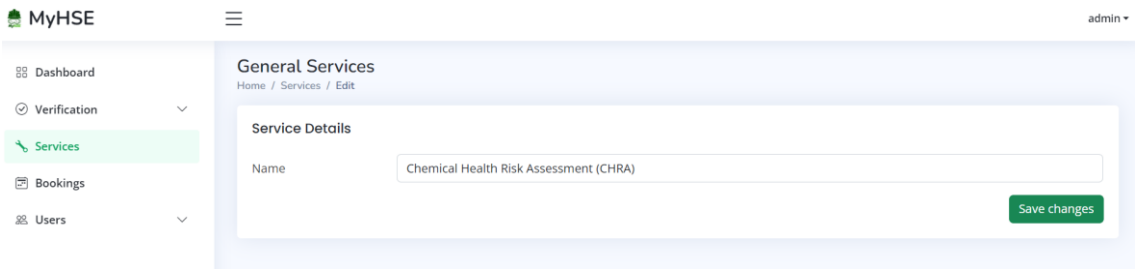



Figure 4.39 Edit Service Page

All bookings made through the system are accessible to administrators. They can examine each booking's specifics, such as the service, date, and subscriber details. Administrators can stay informed about the most recent bookings and make the necessary arrangements thanks to this comprehensive view.

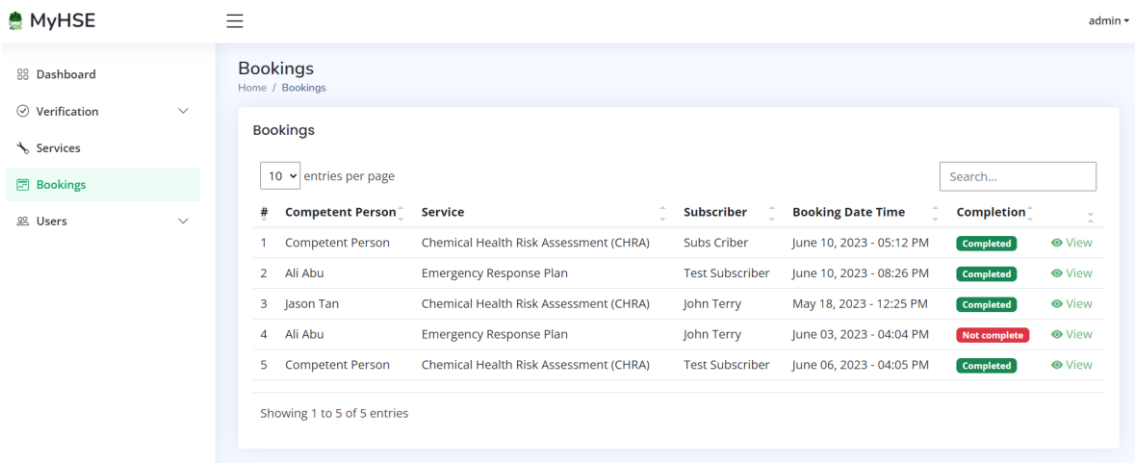



Figure 4.40 Display all Bookings Page

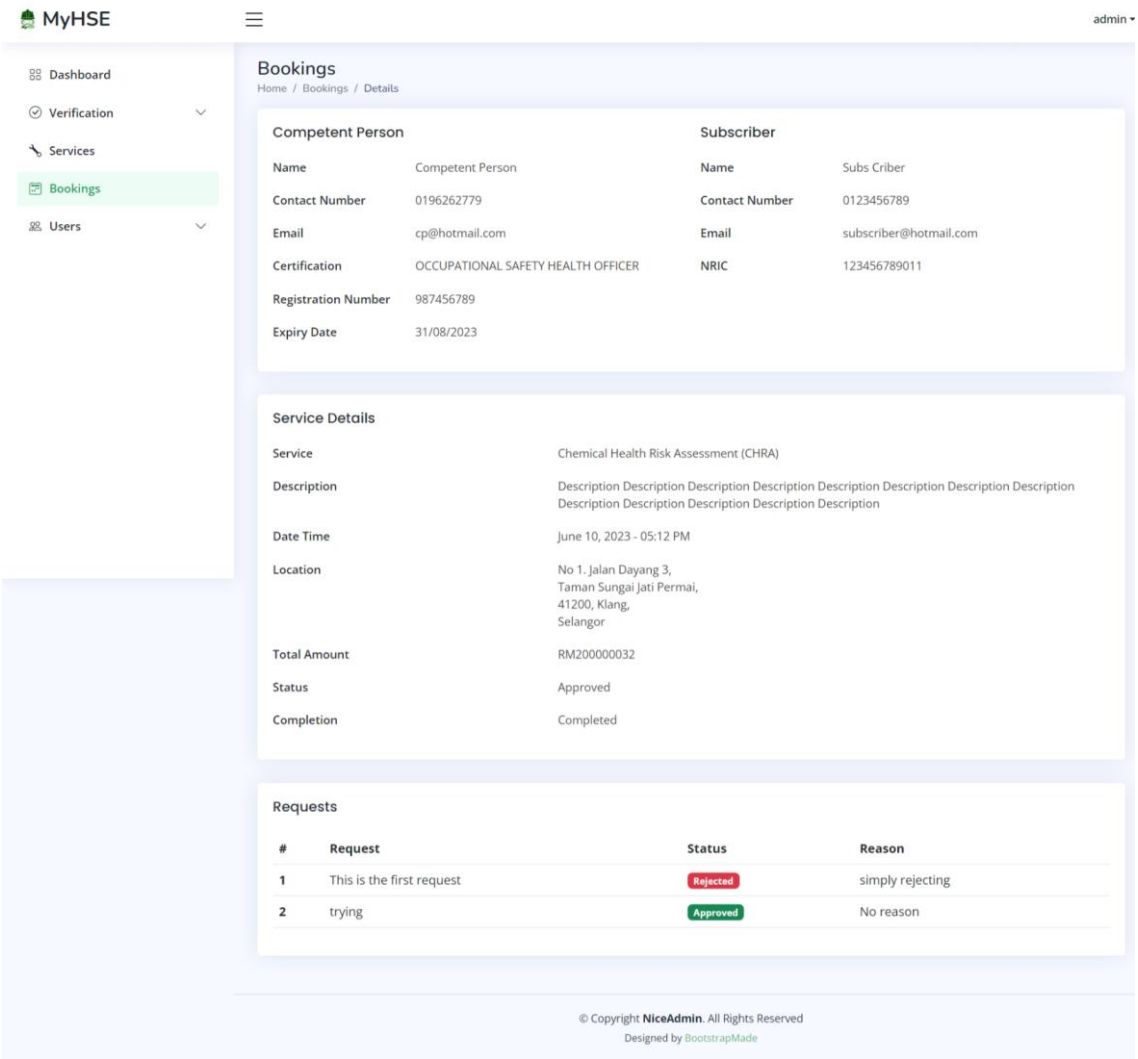



Figure 4.41 Booking Details Page

Administrators also have access to a list of all system users and their associated information. Since they have access to view and modify user data, they can make sure that the user database is accurate and current. Additionally, administrators have the ability to add new administrators to the system, giving them the rights required to run it efficiently.

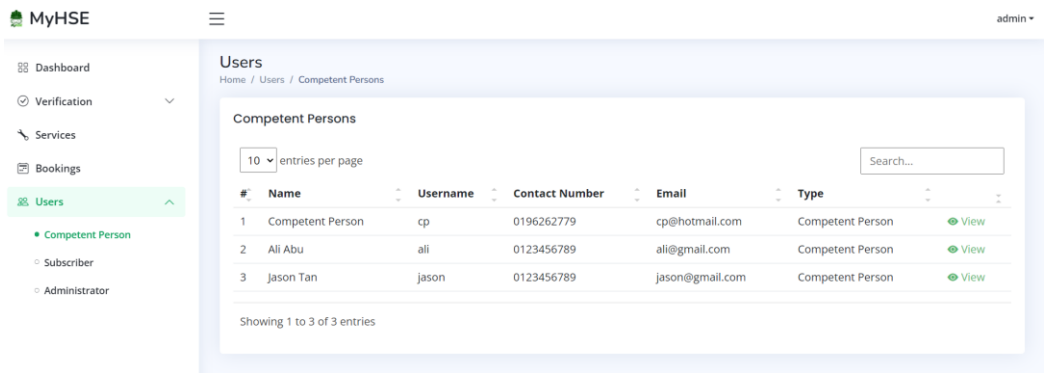



Figure 4.42 Display all Competent Persons

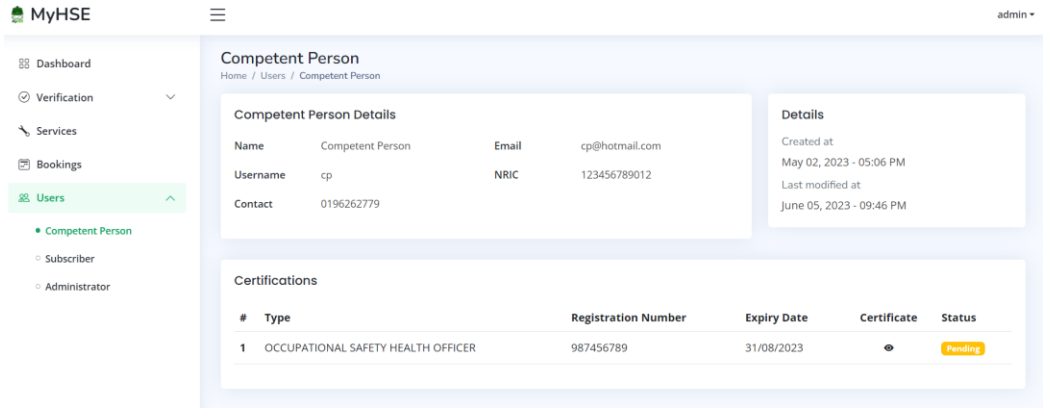



Figure 4.43 Display Competent Person Details Page

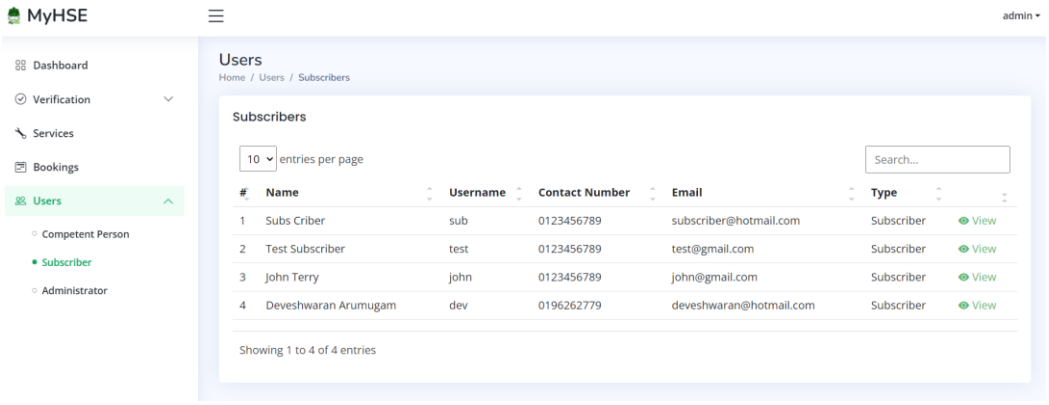



Figure 4.44 Display all Subscribers

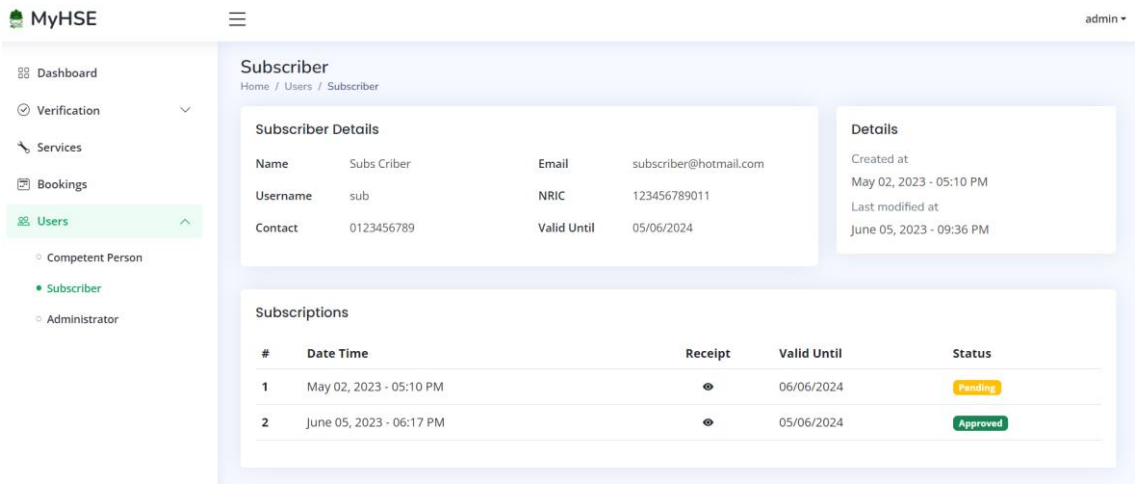



Figure 4.45 Display Subscriber Details Page

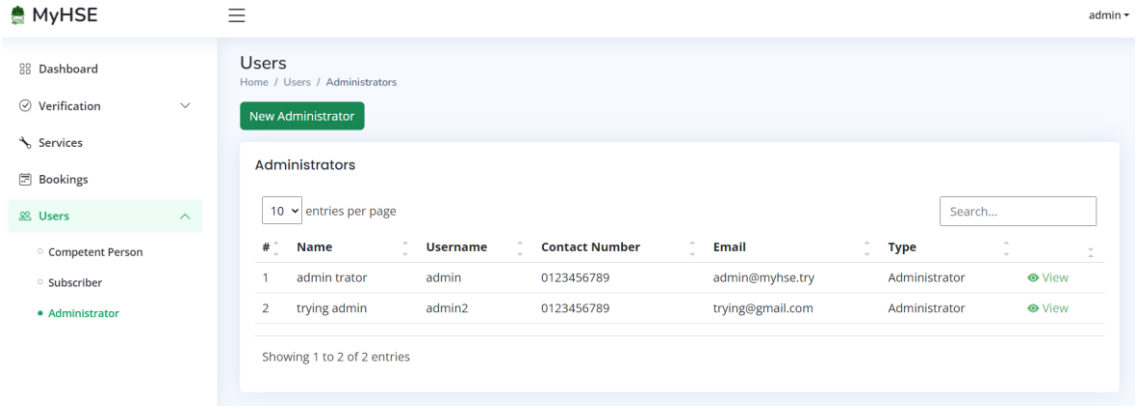



Figure 4.46 Display all Administrator

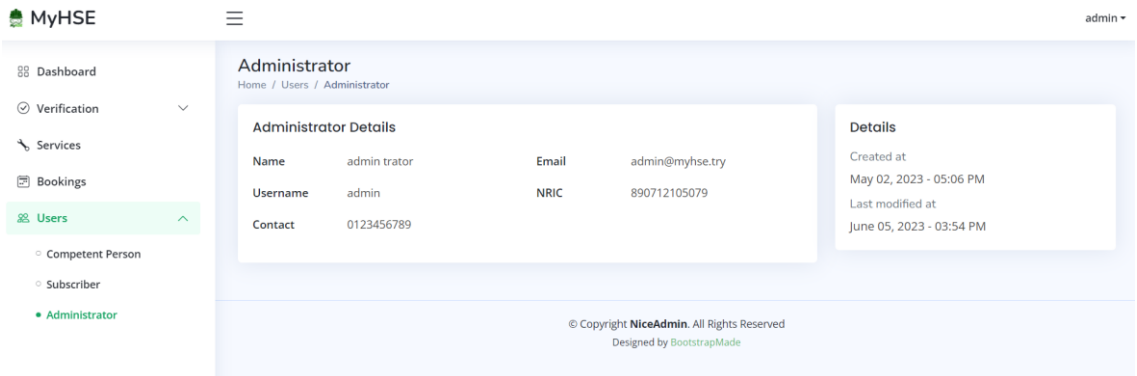



Figure 4.47 Display Administrator Details Page

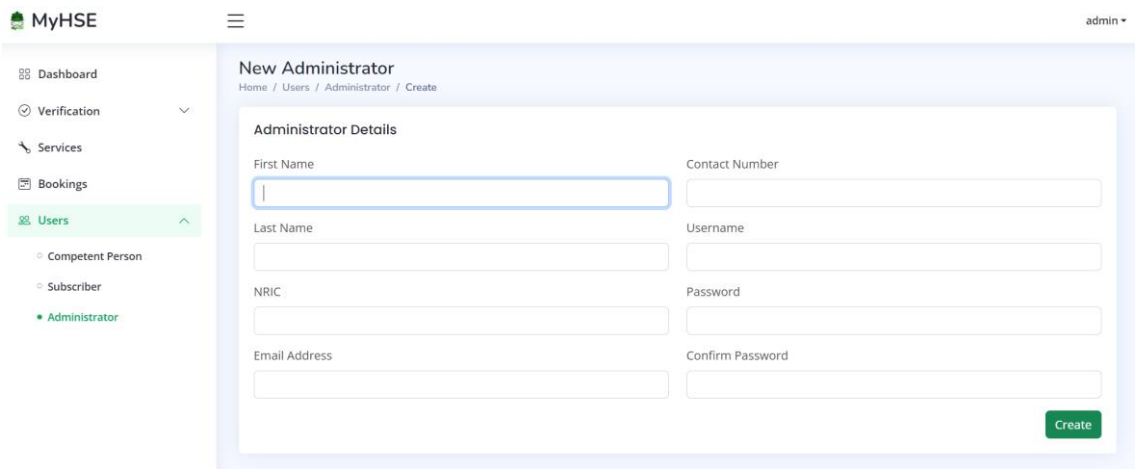



Figure 4.48 Add New Administrator Page

## 4.4 Deployment

The MyHSE Laravel system was deployed in a number of steps. CoreFTP, an FTP client that enables file transfers over a TCP-based network, was used to first transfer the files to the server. Following the successful upload of the files, the system needed to be set up to work with the server's configurations. The.env file, which contains crucial configuration settings for the Laravel application, including database credentials and application keys, was altered to achieve this. To ensure proper connectivity with the server and other environment-specific configurations, the .env file was updated with the necessary settings.

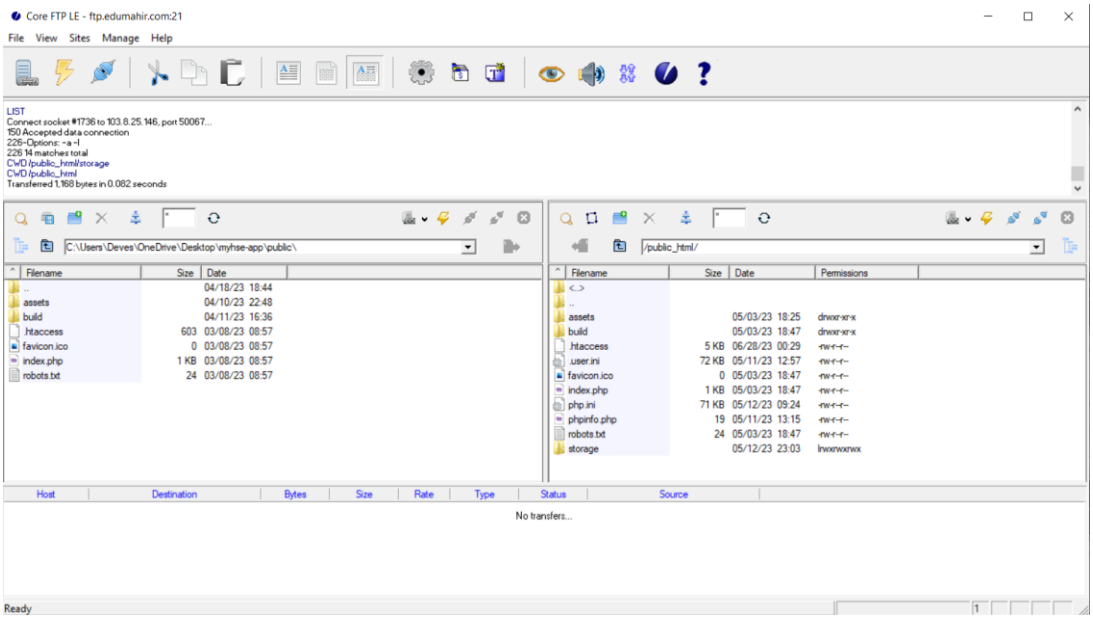



Figure 4.49 CoreFTP Connected to Server

Additionally, Laravel's storage system was set up to allow web access to files. By design, files kept in the storage/app directory are inaccessible directly. To fix this, the Laravel artisan command-line tool's storage:link command was used. This command established a symbolic link between the public/storage directory and the storage/app/public directory, enabling webserver access to files in the storage/app/public directory.

The domain configuration was set up appropriately to allow access to the system at hse.edumahir.com. The DNS records for hse.edumahir.com needed to be correctly set up to point to the server where the Laravel application was deployed in order to accomplish this. In this step, the domain's A (IPv4) or AAAA (IPv6) records are updated using a domain registrar or DNS management service.

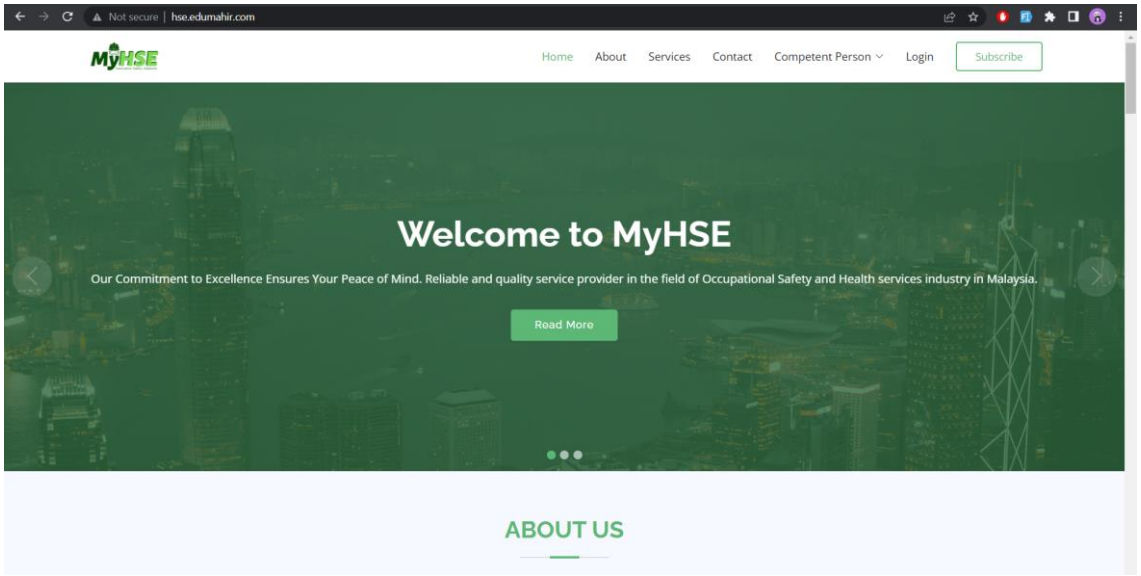



Figure 4.50 MyHSE Accessed from Internet

The Laravel application was then configured to run on the web server. This procedure involved setting up a virtual host or server block configuration that pointed requests towards the public directory of the Laravel project as the document root. Additionally, rewrite rules were set up to enable clean URLs and PHP support was enabled.

The MyHSE system was successfully deployed after these steps and made available at hse.edumahir.com. Now, users can use their web browsers to access the system by going to the designated URL.

## 4.5 Testing

User Acceptance Testing (UAT) was used for verification and validation in order to guarantee the MyHSE system's usability and functionality. UAT is a vital testing strategy that involves having end users assess the system's performance in a practical setting. A UAT test form was used to direct the evaluation procedure during the testing phase. This form, which is included in Appendix A, lists the various test scenarios and evaluation criteria. The UAT test form acts as a guide for carrying out the testing procedure and guarantees that all crucial components of the system are carefully examined.

The main goal of the UAT testing is to evaluate the system's functionality and usability from the viewpoint of the end users. Testers who represent the system's intended users interact with the MyHSE system in a set of predetermined ways to see how well it performs in comparison to the requirements. The test scenarios include user registration, subscription, service browsing, booking process, verification process, and administrative tasks. They also cover a wide range of system functionalities. By highlighting any problems, inconsistencies, or improvements they find during the testing process, testers provide feedback on their experience.

The MyHSE system underwent extensive testing using the UAT method to make sure it satisfies the needs and expectations of its intended users. The systematic evaluation made possible by the UAT test form allowed for the detection and correction of any potential problems or usability issues.

To guarantee a seamless user experience across various screen sizes and resolutions, it is essential to test the responsiveness of a system developed for both mobile and PC devices. The Google Chrome browser comes with a robust set of web development and debugging tools called Chrome DevTools. These tools allow testing and optimising the MyHSE system for different devices, including PCs and mobile devices. In the figure below, the flaw in the responsiveness of the table in the system was detected using the Chrome DevTool. This bug was then fixed for seamless user usage.

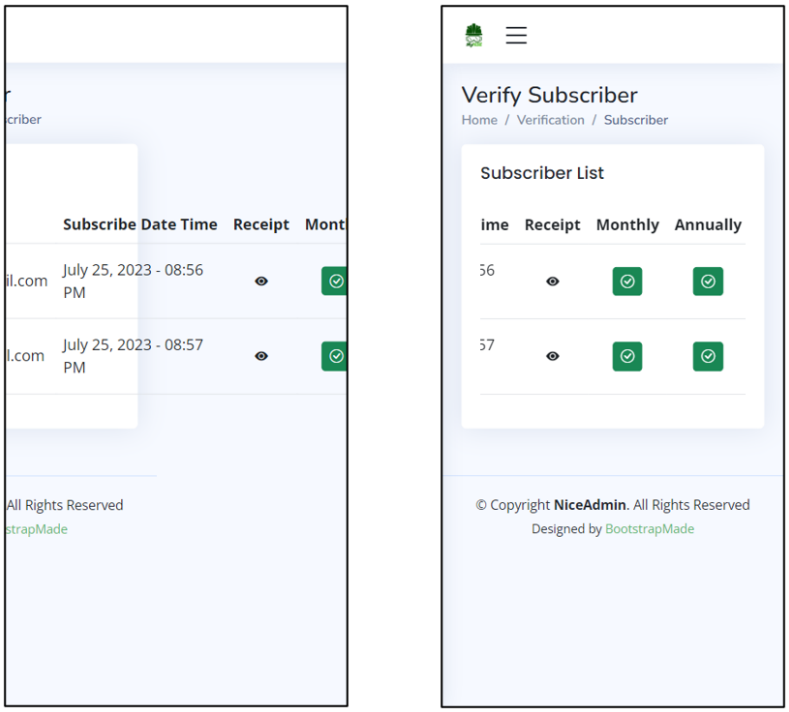



Figure 4.51 Before and After Testing Usability Using Chrome DevTools

# CHAPTER 5

# CONCLUSION

## 5.1 Introduction

In this chapter, the development of the MyHSE system will be summarized. The MyHSE system does more than just list services and phone numbers. It makes it possible for customers and service providers to ask questions, discuss prices, and build trust. This system aims to enhance industry efficiency, accessibility, and accountability by offering a centralised platform for OSH services. It addresses the current issues of fraud and lack of verification while providing a thorough solution that is advantageous to both customers and service providers. The system encourages client satisfaction, trust, and efficient teamwork in the area of occupational safety and health through its features and functionalities. The MyHSE system, which addresses industry challenges and improves the overall OSH service experience, represents a sizable advancement in the field. It is a useful tool for ensuring workplace security and advancing personal wellbeing, ultimately resulting in a safer and healthier working environment for everyone.

## 5.2 Constraints

### Requirement Control

The project's scope grew beyond its initial bounds as a result of the ongoing modifications and additions to the requirements, a process known as scope creep. The project's timeline was significantly impacted by the ongoing modifications and additions to the requirements. The documentation had to be updated, project plan was reassessed, and more resources was allocated as a result of each change. As a result, the delivery of the system by the original deadline was delayed and the schedule slipped.

As the project developed, the client not only changed already-existing requirements but also added entirely new ones. Sometimes unrelated to the original scope, these new additions required significant rework to properly accommodate them. This made it difficult to effectively plan and carry out the development as a result of the influx of new requirements, which produced a feeling of uncertainty.

**Budget Constraints**

Budgetary restrictions on the project had an impact on some system functionalities. The lack of a filesystem on the server that was used to host the application made it difficult to store uploaded files. All uploaded files were consequently directly stored within the server's files. Although this strategy was used to work within the budgetary constraints, there may be some scalability and file management issues.

The implementation of advanced payment features like online banking integration and FPX payment was also constrained by the budgetary constraints. Users must therefore upload their payment receipts for manual review and approval by the administrator. Depending on the admin's availability, this manual verification process could cause some delays. Nevertheless, despite these restrictions, efforts were made to speed up the payment verification process and guarantee users' prompt approval.

**Time constraints**

Aspects of the system testing, particularly security testing, could not be done as thoroughly as desired due to time constraints. Despite the efforts made to ensure system security, thorough security testing would have required more time for vulnerability assessments, penetration testing, and in-depth analysis of potential risks. Nevertheless, despite these drawbacks, the security measures put in place aim to offer a secure environment for user data and transactions.

## 5.3 Future Work

There are several improvements that could be done to the MyHSE system in the future:

- Upgrade Server Infrastructure: To improve file management capabilities and saleability for storing uploaded files, think about using a server with a filesystem. With this upgrade, the system's file storage would be effective and well-organized.

- Implement secure online payment gateways to make it easier for customers to pay for subscriptions and services. Users can conduct smooth and secure transactions by enabling online banking and payment options like FPX, improving the overall user experience.

- Enhance Security Measures: Perform thorough security testing, such as penetration testing and vulnerability assessments, to find and fix potential risks and vulnerabilities. To ensure the highest level of system security, implement additional security features like two-factor authentication, encryption, and secure data storage procedures.

## REFERENCES


BOSH, T. (2022, 04 18). *About Berkat OSH*. Retrieved from Berkat OSH Services: https://www.berkat-osh.com/about/

Jane. (2022, 07 02). *Avoiding OSHA Scams*. Retrieved from decisely: https://decisely.com/avoiding-osha-scams/

Makadia, H. (2022, 12 29). *Your Complete Guide To Rapid Application Development (RAD)*. Retrieved from maruti techlabs: https://marutitech.com/rapid-application-development/

Parker, D. L. (2018, 07 26). *Occupational Safety and Health Administration*. Retrieved from OSHA: https://www.osha.gov/aboutosha

Phia. (2022, 11 15). *Occupational safety*. Retrieved from peopleHum: https://www.peoplehum.com/glossary/occupational-safety

Simmons, T. (2022, 05 15). *Rapid Application Development: What It Is and What's Next*. Retrieved from plutora: https://www.plutora.com/blog/rapid-application-development-what-it-is-and-whats-next

Singh, V. (2022, 09 24). *What is Rapid Application Development Model (RAD)?* Retrieved from hacrio: https://hackr.io/blog/rapid-application-development-model#:~:text=User%20Description%2FUser%20Design,-The%20second%20phase&text=It%20involves%20taking%20user%20feedback,with%20distinct%20features%20and%20functions.

Syah, A. (2021, 04 12). *OHSNET*. Retrieved from OHSNET: https://www.ohsnet.com.my/services/

Team, I. E. (2021, 05 26). *Client Communication: Its Elements and 5 Ways To Improve It*. Retrieved from Indeed: https://www.indeed.com/career-advice/career-development/client-communication#:~:text=Good%20communication%20helps%20businesses%20develop,more%20leads%20to%20the%20business.O

Team, L. C. (2021, 09 12). *4 Phases of Rapid Application Development Methodology*. Retrieved from Lucidchart: https://www.lucidchart.com/blog/rapid-application-development-methodology

Wood, G. (2022, 10 31). *Rapid Application Development (RAD) Model: An Ultimate Guide For App Developers in 2022*. Retrieved from kissflow: https://kissflow.com/application-development/rad/rapid-application-development/

## APPENDIX A
## USER ACCEPTANCE TEST FORM

| No. | Module | Activity | Status | | Comment |
|---|---|---|---|---|---|
| | | | Pass | Fail | |
| 1 | Login | Login into the system | | | |
| | | Redirect to dashboard according to user role | | | |
| 2 | Subscribe | Submit required data | | | |
| | | Redirect to restricted page | | | |
| 3 | Register competent person | Submit required data | | | |
| | | Redirect to restricted page | | | |
| 4 | Verify competent person | Approve competent person | | | |
| | | Reject competent person | | | |
| 5 | Verify subscriber | Approve subscriber annually | | | |
| | | Approve subscriber monthly | | | |
| | | Reject subscriber person | | | |
| 6 | Manage profile | Display profile details | | | |
| | | Edit profile | | | |
| | | Display certifications for competent person | | | |
| | | Display subscriptions for subscriber | | | |
| | | Change password | | | |
| 7 | Manage users | Display all competent persons | | | |
| | | Display competent person details | | | |
| | | Display all subscribers | | | |
| | | Display subscriber details | | | |
| | | Display all administrators | | | |
| | | Display administrator details | | | |
| | | Add new administrator | | | |
| 8 | Manage service | System displays all services | | | |
| | | Add new service | | | |
| | | System displays search results | | | |
| | | System updates service data | | | |
| 9 | Manage booking | Approve booking | | | |
| | | Reject booking | | | |
| | | Add new booking | | | |
| | | System displays all existing booking | | | |
| | | Display booking details | | | |
| | | Add request | | | |
| | | Add reason | | | |
| | | Add completion | | | |
| | | Add first payment | | | |
| | | Add balance payment | | | |