HYGIEIAHUB – BOOKING AND CLEANING HUB

NUR SHAHIRA ATILIA BINTI ZAINUDDIN

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

**BORANG PENGESAHAN STATUS LAPORAN**

JUDUL: HYGIEAHUB – BOOKING AND CLEANING HUB

SESI PENGAJIAN: 2024 / 2025

Saya: NUR SHAHIRA ATILIA BINTI ZAINUDDIN

mengaku membenarkan tesis Projek Sarjana Muda ini disimpan di Perpustakaan Universiti Teknikal Malaysia Melaka dengan syarat-syarat kegunaan seperti berikut:

1. Tesis dan projekadalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan unituk tujuan pengajian sahaja.
3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. \* Sila tandakan (✓)

|  |  |  |
| --- | --- | --- |
| \_\_\_\_\_\_\_\_\_\_ | SULIT | (Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972) |
| \_\_\_\_\_\_\_\_\_\_ | TERHAD | (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi / badan di mana penyelidikan dijalankan) |
| \_\_\_\_\_\_\_\_\_\_ | TIDAK TERHAD |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| (TANDATANGAN PELAJAR) |  | (TANDATANGAN PENYELIA) |
| Alamat tetap: 99, Jalan Duranta 3, |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Taman Desa Duranta, 70450, Seremban, Negeri Sembilan |  | Nama Penyelia |
| Tarikh: 21 Jun 2025 |  | Tarikh: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

CATATAN: \* Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

1. karan antara institusi pengajian tinggi.
2. \* Sila tandakan (✓)

|  |  |  |
| --- | --- | --- |
| \_\_\_\_\_\_\_\_\_\_ | SULIT | (Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972) |
| \_\_\_\_\_\_\_\_\_\_ | TERHAD | (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi / badan di mana penyelidikan dijalankan) |
| \_\_\_\_\_\_\_\_\_\_ | TIDAK TERHAD |  |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| (TANDATANGAN PELAJAR) |  | (TANDATANGAN PENYELIA) |
| Alamat tetap: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | Nama Penyelia |
| Tarikh: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  | Tarikh: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

CATATAN: \* Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

HYGIEAHUB – BOOKING AND CLEANING HUB

NUR SHAHIRA ATILIA BINTI ZAINUDDIN

This report is submitted in partial fulfillment of the requirements for the

Bachelor of [Computer Science (Database Management)] with Honours.

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2025

# DECLARATION

I hereby declare that this project report entitled

[HYGIEAHUB – BOOKING AND CLEANING HUB]

is written by me and is my own effort and that no part has been plagiarized

without citations.

STUDENT :  Date : \_\_\_\_\_\_\_\_

(NUR SHAHIRA ATILIA BINTI ZAINUDDIN)

I hereby declare that I have read this project report and found

this project report is sufficient in term of the scope and quality for the award of

Bachelor of [Computer Science (Database Management)] with Honours.

SUPERVISOR : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date : \_\_\_\_\_\_\_\_

([NAME OF THE SUPERVISOR])

# DECLARATION

I hereby declare that this project report entitled

[HYGIEAHUB – BOOKING AND CLEANING HUB]

is written by me and is my own effort and that no part has been plagiarized

without citations.

STUDENT : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date : \_\_\_\_\_\_\_\_

(NUR SHAHIRA ATILIA BINTI ZAINUDDIN)

I hereby declare that I have read this project report and found

this project report is sufficient in term of the scope and quality for the award of

Bachelor of [Computer Science (Database Management)] with Honours.

SUPERVISOR : \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date : \_\_\_\_\_\_\_\_

([NAME OF THE SUPERVISOR])

# DEDICATION

[To my beloved parents...]

# ACKNOWLEDGEMENTS

[I would like to thank for giving assistant to complete this project successfully…...

I would also like to thank my beloved parents who have been giving me support and motivation throughout my project…]

# ABSTRACT

[Abstract must start from here. Abstracts should be concise, WRITTEN IN ONE SPACING and justification and **NOT MORE THAN 300 WORDS** IN A SINGLE PAGE. Abstract is not the same as synopsis or report summary. Abstracts are WRITTEN WITH **ONE PARAGRAPH**. It should briefly state **the field of report studies, problems to be solved, solutions, research processes; and the results obtained.**]

# ABSTRAK

[Abstrak mesti bermula disini. Abstrak mestilah ringkas, DITULIS DALAM SATU LANGKAU dan justifikasi **TIDAK LEBIH DARIPADA 300 PERKATAAN** DALAM SATU MUKASURAT sahaja. Abstrak tidak sama dengan sinopsis atau ringkasan tesis. Abstrak DITULIS DALAM **SATU PERENGGAN**. Ia hendaklah menyatakan dengan ringkas **bidang kajian tesis, masalah yang hendak diselesaikan, cara penyelesaian, proses penyelidikan; dan keputusan yang diperolehi**.]

# table of contents

|  |  |
| --- | --- |
|  | **PAGE** |

DECLARATION ii

DECLARATION ii

DEDICATION iii

ACKNOWLEDGEMENTS iv

ABSTRACT v

ABSTRAK vi

table of contents vii

list of tables xi

list of figures xii

List of Abbreviations xiv

List of ATTACHMENTS xv

Chapter 1: INTRODUCTION 1

1.1 Project Background 1

1.2 Problem Statement 2

1.3 Objective 2

1.4 Scope 3

1.4.1 Potential Users 3

1.4.2 Scope 3

1.4.2.1 Registration and Login 3

1.4.2.2 Creating and Updating House Types and Services 3

1.4.2.3 Booking Cleaning Services 4

1.4.2.4 Managing Bookings 4

1.4.2.5 Payment Processing 4

1.4.2.6 Customer Feedback & Ratings 5

1.4.2.7 Reporting & Analytics 5

1.5 Expected Output 5

1.6 Conclusion 6

Chapter 2: literature review AND PROJECT METHODOLOGY 7

2.1 Introduction 7

2.2 Facts And Findings 7

2.2.1 Existing Cleaning Service Platforms 7

2.2.2 Importance of Simplified Systems for Local Businesses 8

2.2.3 Trends in Web-Based Booking Systems 8

2.3 Project Methodology 9

2.4 Project requirements 10

2.4.1 Software Requirements 10

2.4.2 Hardware Requirements 10

2.4.3 Functional Requirements 11

2.4.4 Non-Functional Requirements 11

2.5 Project Schedule and Milestones 12

2.5.1 Gantt Chart 12

2.6 Conclusion 12

Chapter 3: ANALYSIS 13

3.1 Introduction 13

3.2 Problem Analysis 13

3.3 Requirement Analysis 14

3.3.1 Functional Requirements 14

3.3.2 Non-Functional Requirements 16

3.3.3 Technical Requirements 16

3.4 Conclusion 16

Chapter 4: DESIGN 18

4.1 Introduction 18

4.2 High-Level Design 18

4.2.1 System Flowchart (Traditional Process) 19

4.2.2 Data Flow Diagram (DFD) 20

4.2.2.1 Context Diagram 21

4.2.2.2 Level 0 22

4.2.2.3 Level 1 25

4.3 Detailed Design 29

4.3.1 Entity Relationship Diagram (ERD) 29

4.3.1.1 Business Rules 29

4.3.1.2 ERD 31

4.3.2 Data Dictionary 31

4.3.3 Interface Design 33

4.3.3.1 Interface Structure 34

4.3.3.2 Interface Design – Input 35

4.3.3.3 Interface Design – Output 39

4.4 Conclusion 43

Chapter 5: IMPLEMENTATION 44

Chapter 6: TESTING 45

Chapter 7: PROJECT CONCLUSION 46

references 47

# list of tables

|  |  |
| --- | --- |
|  | **PAGE** |

[Table 2.1: Software Requirements 8](#_Toc195646998)

[Table 2.2: Hardware Requirements 8](#_Toc195646999)

[Table 3.1: Functional Requirements 12](#_Toc195647000)

[Table 3.2: Non-Functional Requirements 13](#_Toc195647001)

[Table 3.3: Technical Requirements 14](#_Toc195647002)

# list of figures

|  |  |
| --- | --- |
|  | **PAGE** |

Figure 2.1: Gantt Chart 12

Figure 4.1: System Flowchart (Traditional System) 20

Figure 4.2: Context Diagram 21

Figure 4.3: Level 0 of DFD for Process 1, 2, and 3 22

Figure 4.4: Level 0 of DFD for Process 5 and 6 23

Figure 4.5: Level 0 of DFD for Process 6 and 7 24

Figure 4.6: Level 1 of DFD for Process 1 25

Figure 4.7: Level 1 of DFD for Process 2 and 3 27

Figure 4.8: Level 1 of DFD for Process 4 27

Figure 4.9: Level 1 of DFD for Process 5 and 6 28

Figure 4.10: ERD 31

Figure 4.11: Data Dictionary of Table house\_type 32

Figure 4.12: Data Dictionary of Table customer 32

Figure 4.13: Data Dictionary of Table staff 32

Figure 4.14: Data Dictionary of Table additional\_service 32

Figure 4.15: Data Dictionary of Table booking 33

Figure 4.16: Data Dictionary of Table booking\_service 33

Figure 4.17: Data Dictionary of Table booking\_cleaner 33

Figure 4.18: Data Dictionary of Table payment 33

Figure 4.19: Data Dictionary of Table feedback 33

Figure 4.20: Interface Structure for User Customer 34

Figure 4.21: Interface Structure for User Staff 35

Figure 4.22: Customer - Sign Up Page 36

Figure 4.23: Customer & Staff - Sign In Page 36

Figure 4.24: Customer - Book Now Page 37

Figure 4.25: Staff - Register Staff Page 37

Figure 4.26: Staff - Add Service Page 38

Figure 4.27: Staff - Edit Service Page 38

Figure 4.28: Staff - Manage Booking Page for editing 39

Figure 4.29: Customer - Bookings Page 40

Figure 4.30: Customer - Community Page 40

Figure 4.31: Staff - View Service Page 41

Figure 4.32: Staff - Manage Staff Page 41

Figure 4.33: Staff - Manage Booking Page 42

Figure 4.34: Staff - Sales Report Page 42

Figure 4.35: Staff - Feedback Report Page 43

# List of Abbreviations

|  |  |  |
| --- | --- | --- |
| **FYP** | **-** | **Final Year Project** |
| **DFD** | **-** | **Data Flow Diagram** |
| **ERD** | **-** | **Entity Relationship Diagram** |

# List of ATTACHMENTS

|  |  |  |  |
| --- | --- | --- | --- |
|  | | | **PAGE** |
|  | | |  |
| …….. |  | ………… |  |
| …….. |  | ………… |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# INTRODUCTION

## Project Background

HygieiaHub is a dedicated cleaning service booking platform developed to simplify and streamline the process of scheduling professional home cleaning services. In many traditional settings, booking such services involves manual communication, inconsistent pricing, and unclear service expectations, often leading to inconvenience and customer dissatisfaction. HygieiaHub addresses these challenges by offering a structured and efficient digital solution tailored for both customers and service providers.

The platform provides a set of predefined cleaning service packages based on house types, allowing users to quickly select the service that best suits their needs. By standardizing the offerings and linking them to specific house categories, HygieiaHub ensures clarity in pricing and expectations, reducing the likelihood of misunderstandings or hidden charges. A key feature of the platform is its exclusive use of cash-on-delivery (COD) as the sole payment method. This approach caters to users who prefer traditional payment practices, offering them greater control and confidence by allowing payment only after the service has been completed to their satisfaction.

From an operational standpoint, HygieiaHub also facilitates smoother management of service requests. It enables cleaning service providers to receive and track bookings efficiently, allocate staff, and ensure timely delivery of services. This dual focus on customer convenience and operational oversight makes HygieiaHub a practical and reliable tool in the growing on-demand service industry.

In summary, HygieiaHub is designed to modernize the home cleaning service experience by combining standardized service packages, transparent pricing, and COD-based transactions within a user-friendly digital platform.

## Problem Statement

Many customers experience inconvenience when attempting to book cleaning services due to unstructured and outdated scheduling systems. The absence of a standardized booking process often results in confusion, miscommunication, and difficulty securing preferred time slots. This not only affects customer satisfaction but also impacts the efficiency of service providers in managing daily operations.

In addition to booking issues, there is often no reliable system in place for tracking service status. Customers are left uncertain about when or if a service will be delivered, which diminishes trust in the provider. On the service side, staff members face challenges in monitoring bookings, assigning tasks, and responding promptly to customer requests, leading to inefficiencies and potential service delays.

Furthermore, the lack of data analytics and reporting tools severely limits the ability of service providers to evaluate performance and anticipate demand. Without insights into customer behavior, peak booking times, or service preferences, it becomes difficult to optimize operations, plan resource allocation, or improve service quality. This data gap hinders growth and responsiveness in a competitive service market.

## Objective

The first objective is to implement a structured and user-friendly booking system that simplifies the process of scheduling cleaning services. By providing predefined service options and clear availability based on house type and time slots, the system aims to reduce booking conflicts and ensure customers can efficiently secure appointments that meet their needs.

The second objective is to establish a reliable service tracking mechanism that enhances transparency and communication between customers and service providers. This feature will allow customers to monitor the status of their bookings in real time, while enabling staff to manage appointments, allocate resources, and oversee operations more effectively.

The third objective is to develop an integrated reporting and analytics system that captures key data points such as service demand, peak booking times, and customer preferences. These insights will support data-driven decision-making, allowing service managers to optimize operations, improve service quality, and better align offerings with customer expectations.

## Scope

### Potential Users

1. Customer – Individuals seeking to book professional cleaning services.
2. Staff – Personnel responsible for managing services, bookings, and system administration.

### Scope

#### ****Registration and Login****

* Users can create an account and log in using a valid email address and password.
* Only registered users are granted access to the system.
* Access by: Customers, Staff

#### Creating and Updating House Types and Services

* Staff can define various house types (e.g., apartment, terrace, bungalow) and create associated cleaning services.
* Staff can also update house type details, edit service information, or delete services when necessary.
* Access by: Staffs

#### ****Booking Cleaning Services****

* Customers can browse a list of available cleaning services, each with a description and corresponding price.
* Booking options include selecting optional services, choosing base booking hours, specifying the number of cleaners, entering custom requests, and selecting a preferred date and time.
* The system automatically assigns a cleaner based on real-time availability once a booking is confirmed.
* **Access by:** Customers

#### ****Managing Bookings****

* Staff can view, update, and manage all customer bookings, including changing booking statuses (e.g., Pending → Completed / Cancelled).
* Customers can view the current status of their bookings and are allowed to cancel if needed.
* **Access by:** Customers (view only), Staffs (manage bookings)

#### ****Payment Processing****

* The system automatically calculates the total cost of services based on selected booking details, including service type and applicable taxes.
* Customers are notified to prepare cash-on-delivery (COD) payment upon service completion.
* Staff are responsible for updating the payment status after service completion.
* **Access by:** Customers (view payment details), Staffs (update payment status)

#### ****Customer Feedback & Ratings****

* Customers can provide feedback by rating completed services and leaving written comments.
* Staff can access and review customer feedback to assess service quality and identify areas for improvement.
* **Access by:** Customers (submit and view feedback), Staffs (view feedback)

#### ****Reporting & Analytics****

* Staff can generate performance reports based on daily, monthly, or yearly service sales, as well as monthly customer feedback summaries.
* These insights help in monitoring performance and improving operational decisions.
* **Access by:** Staffs

## Expected Output

One of the primary expected outcomes of this project is a fully functional cleaning service booking platform that enables customers to schedule and manage cleaning services with ease. By providing a selection of predefined services tied to specific house types and prices, the platform eliminates ambiguity in service selection and cost estimation. The user-friendly interface will allow customers to make bookings, set preferences, and receive service confirmations efficiently, ensuring a smooth and convenient experience from start to finish.

Another significant expected result is the implementation of an efficient booking and service status tracking system. This feature will allow customers to stay informed about their service progress in real time, increasing transparency and trust in the platform. At the same time, staff members and managers will benefit from streamlined operational control, enabling them to update booking statuses, reassign resources, and respond to service changes promptly. This dual-view system enhances both user satisfaction and backend efficiency.

Additionally, the platform is expected to include a reporting and analytics feature that provides valuable insights for service managers. This tool will offer data on key metrics such as booking frequency, peak service hours, customer feedback trends, and overall service performance. With access to this information, managers can make informed decisions to optimize staffing, improve service offerings, and align business strategies with customer demand patterns, ultimately supporting continuous improvement and sustainable growth.

## Conclusion

In conclusion, this chapter has introduced HygieiaHub, a web-based cleaning service booking platform designed to address limitations found in current service platforms, particularly in terms of complexity and regional relevance. By focusing on simplicity and essential features, the platform aims to provide a user-friendly experience for customers and efficient management tools for admins and managers. The problem statement, objectives, scope, and project significance lay the foundation for the development and implementation of a practical, database-driven solution tailored to a single service provider.

# literature review AND PROJECT METHODOLOGY

## Introduction

This chapter presents the literature review and methodology adopted for the development of HygieiaHub, a simplified cleaning service booking platform designed to enhance user convenience and operational efficiency. The aim is to explore existing research, studies, and systems related to online service booking platforms, particularly those involving manual coordination challenges, booking automation, and service tracking. By analyzing the limitations of current solutions and identifying best practices, this chapter establishes the foundation for the system's functional design.

In addition, this chapter details the methodology selected for system development, including the development model, tools, technologies, and system requirements used throughout the project. The development process was guided by the need to create a structured, user-focused solution that supports real-time booking management, service status tracking, and administrative reporting. The approach ensures the system is both technically sound and aligned with user expectations.

## Facts And Findings

### Existing Cleaning Service Platforms

Several existing platforms, such as *Kaodim, Maideasy,* and *ServisHero*, provide on-demand home cleaning services to urban consumers. While these platforms offer a broad range of features, they often introduce complexity in navigation and require users to go through multi-step booking processes. Moreover, many of these services lack transparent pricing structures or promote recurring subscription models, which may not appeal to users seeking one-time services. Additionally, due to their operation across various regions, these platforms tend to adopt a generalized design approach, which may compromise user experience in specific local contexts.

This observation aligns with findings from usability studies. For example, Lee et al. (2015) highlighted that simplicity and interactivity in mobile application usability are positively correlated with user satisfaction, trust, and brand loyalty. This underlines the importance of designing service platforms that are easy to navigate and tailored to user expectations especially in a localized context.

### Importance of Simplified Systems for Local Businesses

Small and medium-sized enterprises (SMEs), particularly in the service sector, often prioritize practical and easily manageable solutions over complex, feature-heavy platforms. These businesses typically require clear booking workflows, intuitive interfaces, and basic analytics, rather than advanced enterprise-level systems. As Brown and Lockett (2004) observed, SMEs tend to favor low-complexity applications and shy away from tools with steep learning curves or unnecessary functionalities.

In this regard, HygieiaHub adopts a minimalist design approach that emphasizes functionality over feature-bloat. It focuses specifically on one-time cleaning service bookings, enabling local businesses to manage operations efficiently without requiring advanced technical knowledge or dedicated IT support.

### Trends in Web-Based Booking Systems

Web-based platforms remain the most accessible and widely adopted solution for service-based businesses, primarily due to their cross-platform compatibility, ease of deployment, and scalability. Technologies such as PHP and MySQL continue to be prominent choices for academic and small-scale commercial projects. These technologies integrate well with development tools like XAMPP and Visual Studio Code, providing a lightweight and flexible development environment (Halim, Abdullah, & Mohd Yasin, 2020).

For the development of HygieiaHub, a web-based architecture ensures that users can access the system through any modern browser without the need for dedicated mobile or desktop applications. This not only simplifies development and maintenance but also increases the platform’s accessibility for both customers and staff.

## Project Methodology

For the development of HygieiaHub, the **Waterfall Model** was selected as the project methodology due to its structured and linear approach, which is well-suited for projects with clearly defined requirements and scope. Given that the system objectives and functional boundaries were established early in the planning phase, the sequential nature of the Waterfall Model allowed for disciplined progression through each development stage.

The first stage, Requirement Analysis, involved examining existing platforms such as Maideasy and Kaodim, alongside gathering user expectations through informal observations and case studies. From this process, key features were identified: user registration and login, service booking with optional add-ons, administrative management for staff, basic reporting and analytics, and a feedback system for completed services.

In the System Design phase, diagrams were created using tools like *draw.io* to visualize system architecture and data flow. This included flowcharts, context diagrams, Data Flow Diagrams (DFDs), Entity-Relationship Diagrams (ERDs), and interface structure models. The database design was centered around core entities such as *Users*, *Bookings*, *Additional\_Service*, and *Staff*, ensuring that the system could support essential functionalities with logical data relationships.

During the Implementation stage, development was carried out using *Visual Studio Code* as the primary code editor. The backend was developed using *PHP*, with *HTML* and *CSS* used for structuring and styling the frontend. *XAMPP* served as the local server environment, and *MySQL* was utilized for database management. These technologies were chosen for their accessibility, integration simplicity, and suitability for small to medium-scale web applications.

The Testing phase involved both unit testing and integration testing. Individual modules such as booking, login, and feedback were tested to ensure functional correctness. Integration testing was performed to verify that different components of the system (e.g., booking flow, status updates, and feedback submission) interacted seamlessly without errors.

Finally, in the Deployment and Maintenance stage, the platform was deployed in a local environment for demonstration and testing purposes. While the system is currently hosted locally, the codebase is documented and structured to support future deployment to a live web server if necessary. Regular maintenance activities, such as code refactoring and bug fixing, are also planned to ensure long-term functionality and scalability.

## Project requirements

### Software Requirements

Table 2.1: Software Requirements

|  |  |
| --- | --- |
| **Software** | **Purpose** |
| Visual Studio Code | Code development environment |
| XAMPP (MySQL, Apache) | Web server and database |
| Draw.io | System design diagrams |
| Google Chrome | System testing (browser) |

### Hardware Requirements

Table 2.2: Hardware Requirements

|  |  |
| --- | --- |
| **Component** | **Minimum Specification** |
| Processor | Intel i3 or higher |
| RAM | 4 GB minimum |
| Storage | 50 GB available space |

### Functional Requirements

* User Registration and Login
* Create and Update House Types
* Create, Update, and View Additional Services
* Book and View Cleaning Services
* Process Payment
* Create and Update Staff Information
* Manage Bookings’ Status and Payments’ Status
* Create and View Feedbacks
* Analytics and Reporting

### Non-Functional Requirements

* User-friendly Interface
* COD Payment Support
* Secure Authentication
* Fast Response Time

## Project Schedule and Milestones

### Gantt Chart



Figure 2.1: Gantt Chart

## Conclusion

In conclusion, this chapter has reviewed key literature related to existing service platforms and emphasized the benefits of designing a simplified, localized system such as HygieiaHub. The selected Waterfall methodology provides a structured and sequential approach suitable for the project's well-defined scope. Tools like Visual Studio Code, XAMPP’s MySQL, and draw.io were chosen for system development and design purposes. The project’s functional and non-functional requirements were outlined clearly to guide implementation. Additionally, a project timeline was carefully planned and managed by the developer to ensure that each development phase is completed systematically and within the allocated timeframe. These elements collectively form a strong foundation for the system's successful development.

# ANALYSIS

## Introduction

This chapter presents a comprehensive analysis of the core problems addressed by HygieiaHub and identifies the system requirements necessary for its development. The analysis process involves translating the findings from the literature review and project objectives into detailed system specifications. It defines what the system must do (functional requirements) and how it should perform (non-functional requirements). This phase acts as a critical bridge between theoretical understanding and the practical implementation of the system, ensuring that the final solution aligns with both user needs and technical feasibility.

## Problem Analysis

Many existing cleaning service platforms are designed with scalability in mind, often serving users across multiple regions. However, this broad focus results in systems that are unnecessarily complex for small, local service providers. One of the main issues is complex navigation which platforms often include numerous features and layered interfaces that can confuse users, especially first-time visitors or elderly customers who are less familiar with digital environments.

Another problem is the inclusion of unnecessary features such as AI-driven scheduling or support for multiple companies. While these may be useful for large-scale operations, they add to system overhead and may complicate usage for small businesses that only require basic tools to manage bookings and staff.

The third issue involves payment processing. Many existing platforms prioritize online or card-based payments, which may not be suitable for users in areas where Cash on Delivery (COD) is still the preferred method. For customers who value simplicity and direct payment upon service completion, these digital-only models create a barrier to adoption.

Lastly, booking management is often inefficient in platforms that lack proper administrative tools. Without a straightforward system for assigning cleaners to appointments or tracking service progress, staff can struggle to keep operations organized. HygieiaHub is specifically designed to address these challenges by focusing only on the essential functions needed for a smooth, locally-optimized service experience.

## Requirement Analysis

The requirements for the HygieiaHub platform were identified through a combination of user expectation analysis, observation of existing service platforms, and alignment with the project's specific goals. The focus was to design a system that is both user-friendly and efficient for localized cleaning service providers, emphasizing simplicity, reliability, and ease of use. The identified requirements are divided into two categories: functional requirements and non-functional requirements, ensuring that both system capabilities and quality attributes are clearly addressed.

### Functional Requirements

These define the core actions the system must support:

Table 3.1: Functional Requirements

|  |  |
| --- | --- |
| **Functionality** | **Description** |
| User Registration and Login | Customers should be able to register and log in securely. Admin adds staff accounts manually. |
| Create and Update House Types | Staffs may create house types. Staffs can also update service. |
| Create, Update, and View Additional Services | Staffs may create and update or delete additional services while also can view defined services with price and descriptions. |
| Book and View Cleaning Services | Customers should be able to select services, time, date, booking hours, number of cleaners, and include optional instructions. Cleaners should be auto-assigned when a booking is confirmed. Customers may review their booking history |
| Process Payment (COD) | System should calculates price based on (booking details + selected additional services) × number of cleaners + service tax, then create an invoice for customer. Customer pays on-site and receipt will be generated. |
| Create and Update Staff Information | Staffs can register a staff as cleaner or admin for their branch. Staffs should be able to update cleaners’ information. |
| Manage Bookings’ Status and Payments’ Status | Staffs can view bookings, and update booking or payment statuses. |
| Create and View Feedbacks | After service completion, customers can leave ratings and optionally comment. Customers should be able to view other feedbacks by other customers. |
| Analytics and Reporting | Staffs can generate reports (daily, monthly, and yearly sales, and feedback reports). |

### Non-Functional Requirements

These define quality standards the system must meet:

Table 3.2: Non-Functional Requirements

|  |  |
| --- | --- |
| **Requirement** | **Description** |
| Usability | Easy navigation and user-friendly interface for all users. |
| Accessibility | System designed for desktop browser access. |
| Performance | Fast loading times and responsive design. |
| Security | Secure login with password protection and input validation. |
| Simplicity | No unnecessary features; only core functions to reduce user confusion. |

### Technical Requirements

Table 3.3: Technical Requirements

|  |  |
| --- | --- |
| **Type** | **Specification** |
| Development Tool | Microsoft Visual Studio Code |
| Backend & Database | PHP and MySQL (via XAMPP) |
| Design Tool | draw.io for diagrams and planning |
| Testing Tool | Localhost testing via XAMPP (browser-based) |

## Conclusion

In summary, this chapter has identified the core problems present in many existing service platforms and clarified how HygieiaHub addresses those problems with a simpler, localized, and user-friendly approach. The system’s requirements were thoroughly analyzed and defined in terms of functionality, usability, performance, and security. These clearly defined requirements will guide the system design and development process.

# DESIGN

## Introduction

This chapter outlines the system design phase of HygieiaHub, detailing how the system’s architecture is structured and how each component interacts to fulfill user and business requirements. The design phase plays a vital role in bridging the gap between requirement analysis and actual system implementation, providing a clear technical blueprint for developers. A well-defined design ensures that the system will operate efficiently, be maintainable, and meet user expectations in terms of usability and performance.

To achieve this, both high-level and low-level design elements are presented. The high-level design captures the overall system structure, showing how major components communicate, while the low-level design dives into specific processes and data interactions. This chapter utilizes various visual modeling tools including flowcharts, context diagrams, Data Flow Diagrams (DFDs), Entity-Relationship Diagrams (ERDs), and interface structure diagrams. These diagrams help clarify how data flows through the system, how users interact with it, and how internal components such as the database and backend logic are organized.

## High-Level Design

High-level design focuses on the overall architecture of HygieiaHub, describing how system components interact and how users engage with the platform. This stage defines the backbone of the system, including how data is processed and how responsibilities are distributed across modules. It provides a clear conceptual understanding of system behavior without diving into implementation details, serving as a guide for developers and stakeholders alike.

### System Flowchart (Traditional Process)

The system flowchart provided in this section illustrates the traditional process flow followed by cleaning service businesses before adopting a digital platform like HygieiaHub. In many local companies, operations such as booking requests, staff assignments, and payment confirmations are handled manually through phone calls, text messages, notebooks, or simple chat applications. These methods are often disorganized, time-consuming, and susceptible to human error.

This traditional process typically involves customers calling in to request services, followed by manual recording by staff, informal staff assignment based on availability, and lack of systematic payment tracking especially when dealing with Cash on Delivery (COD). There is limited visibility for both staff and customers regarding booking statuses or service records, which reduces overall service reliability and customer satisfaction.

By analyzing this flow, HygieiaHub was designed to directly resolve these inefficiencies by automating the booking process, integrating cleaner assignment, and standardizing payment reminders and service tracking. The new system transforms these disconnected manual steps into a seamless digital workflow, improving accuracy, traceability, and user convenience.

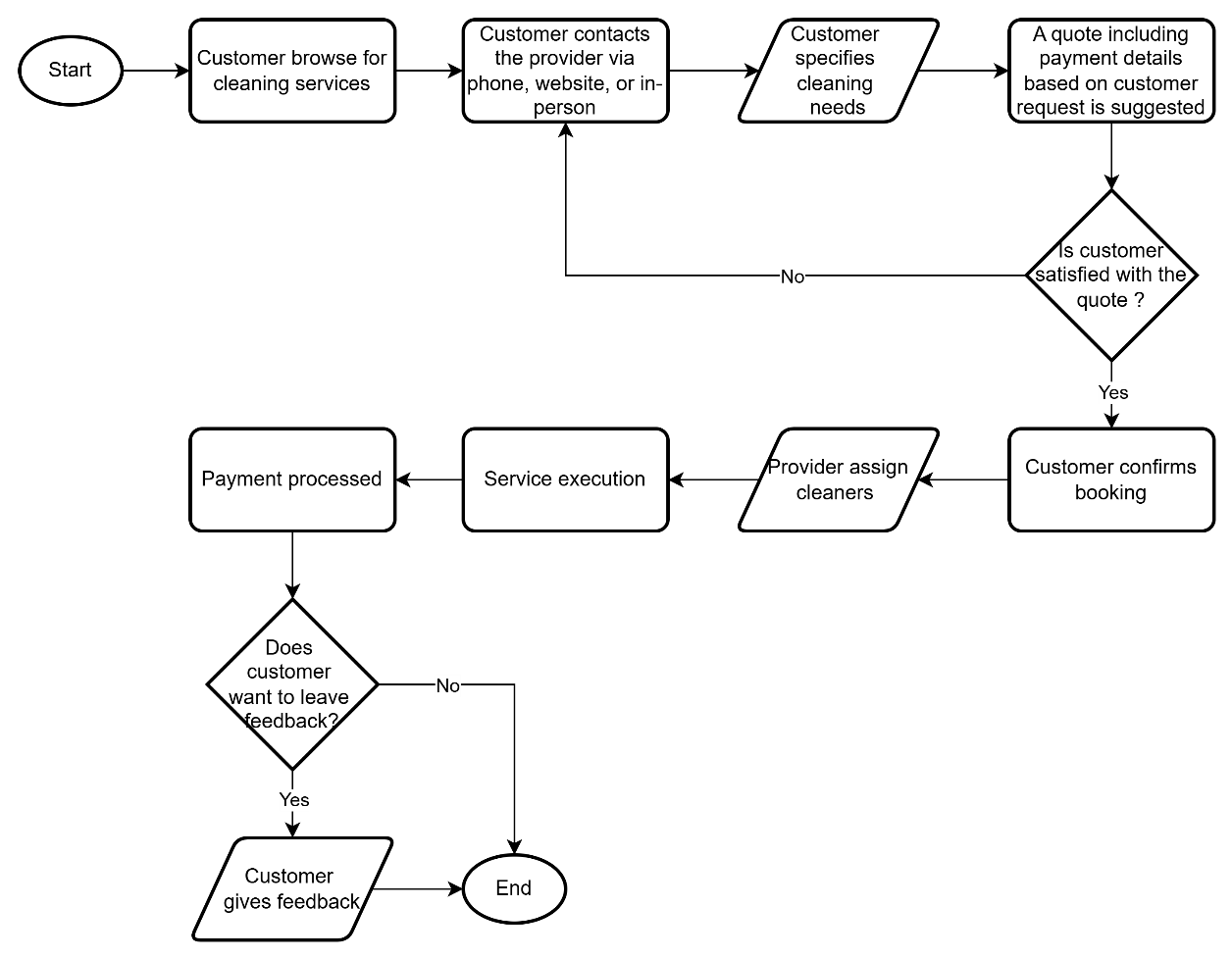


Figure 4.1: System Flowchart (Traditional System)

### Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) is a fundamental design tool that illustrates how data moves through the HygieiaHub system. It highlights the interaction between external users (such as customers and staff) and the internal processes that manage bookings, services, payments, and feedback. By visualizing these data flows, the DFD helps ensure that each component of the system is logically connected and that all functional requirements are addressed efficiently.

The DFD is presented across three levels to reflect increasing detail. The Context Diagram provides a high-level view, treating the system as a single process that exchanges data with external entities. The Level 0 DFD breaks this down into major internal processes such as booking handling, user management, and payment calculation. Finally, the Level 1 DFD expands these processes further into specific actions and data interactions, offering a more granular look at how each function operates within the system.

#### Context Diagram

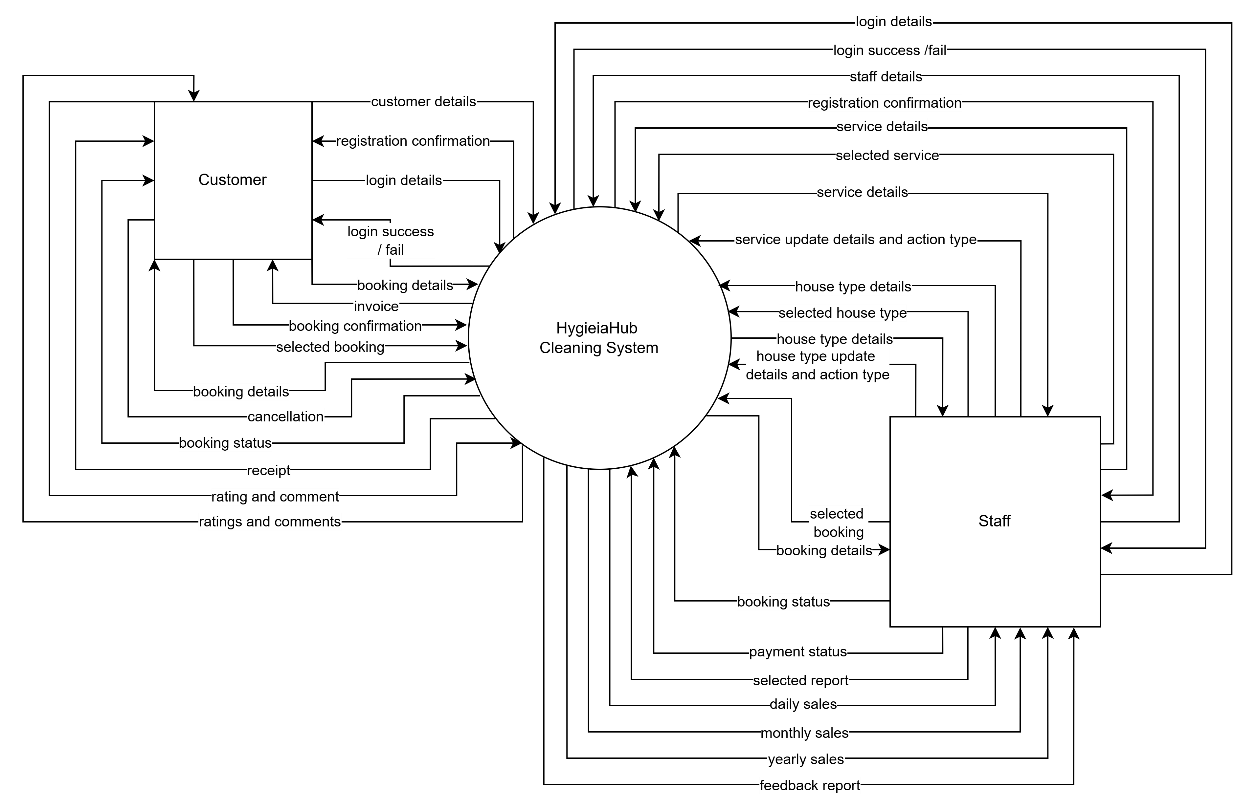


Figure 4.2: Context Diagram

#### Level 0

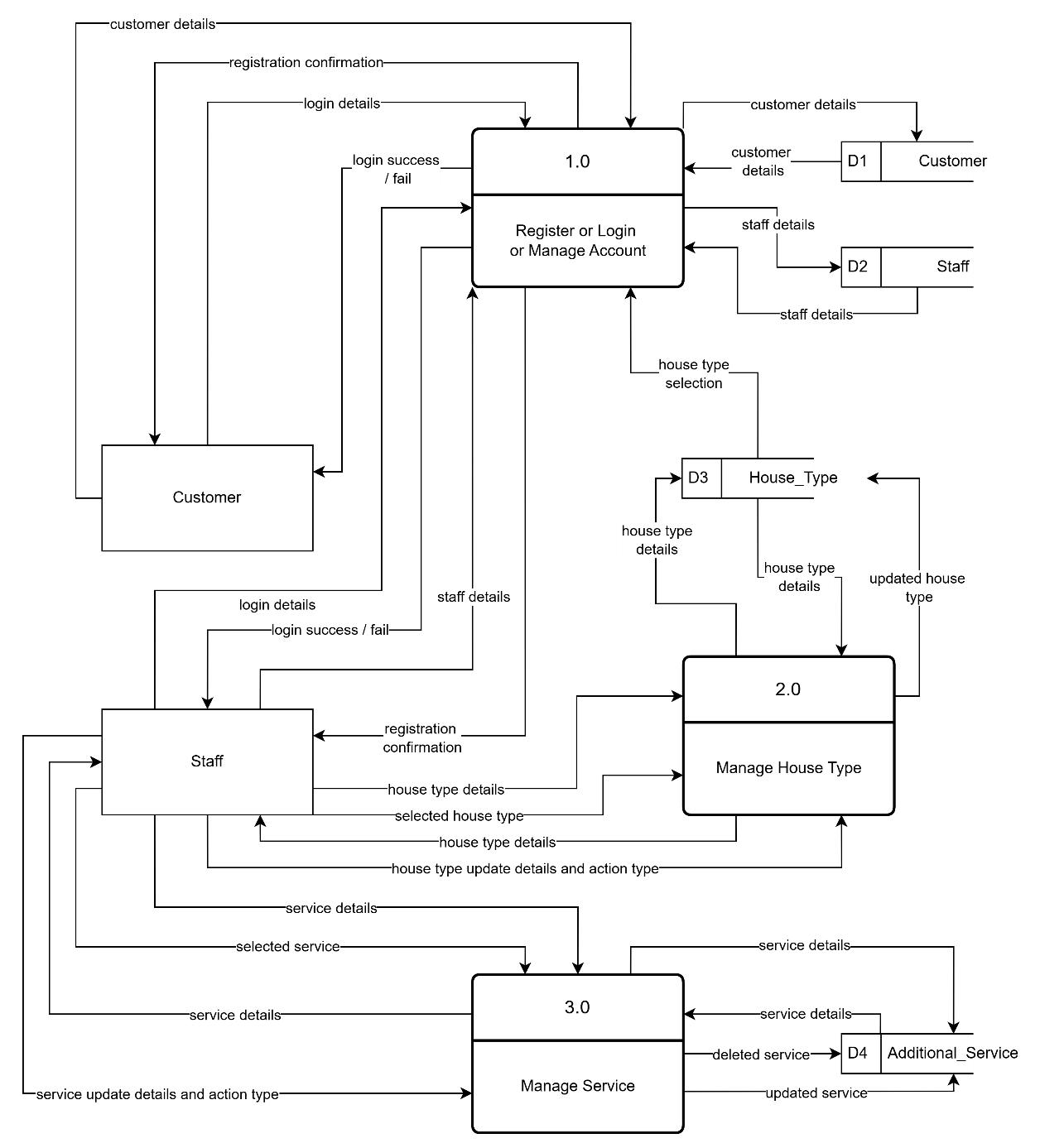


Figure 4.3: Level 0 of DFD for Process 1, 2, and 3

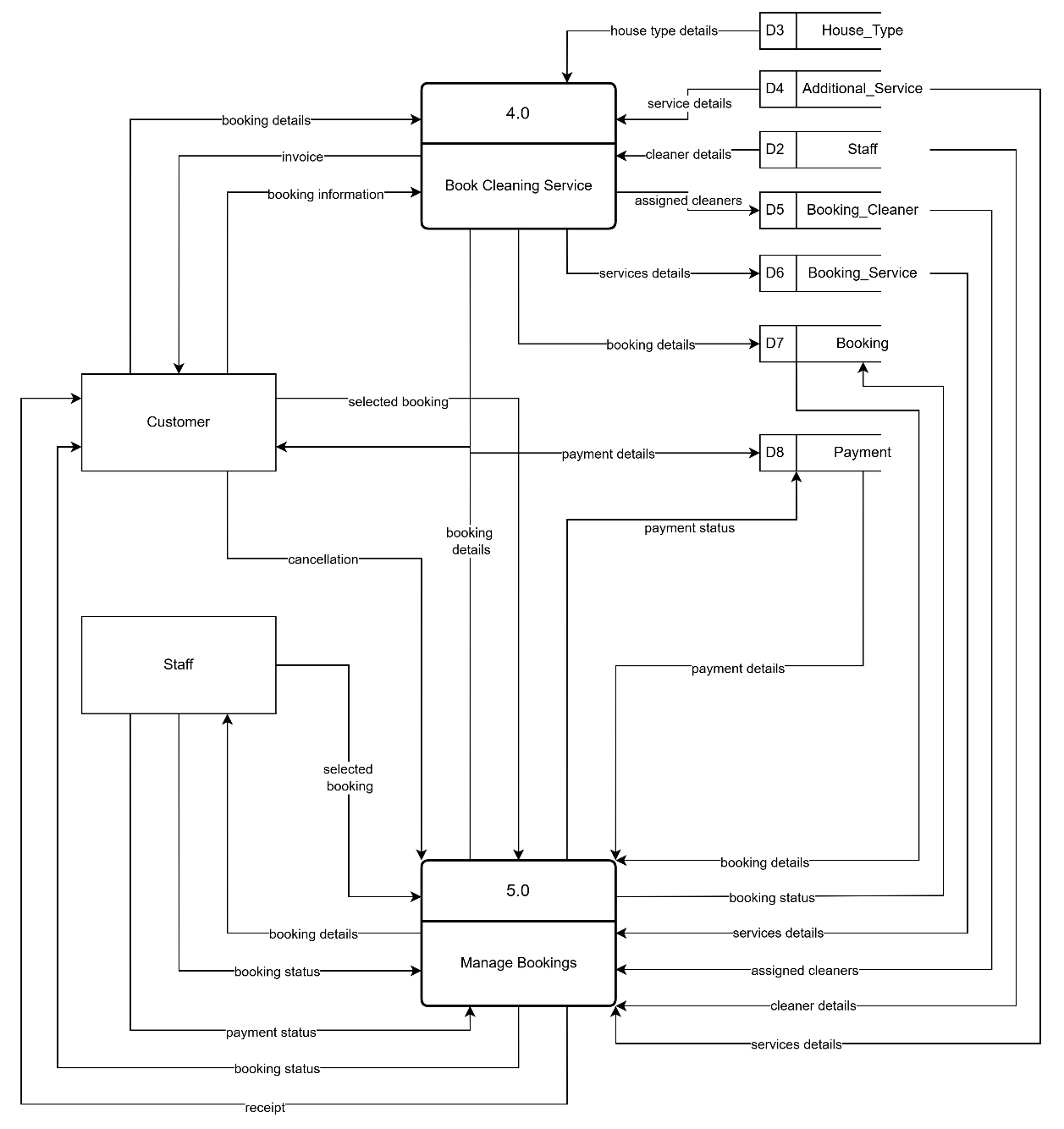


Figure 4.4: Level 0 of DFD for Process 5 and 6

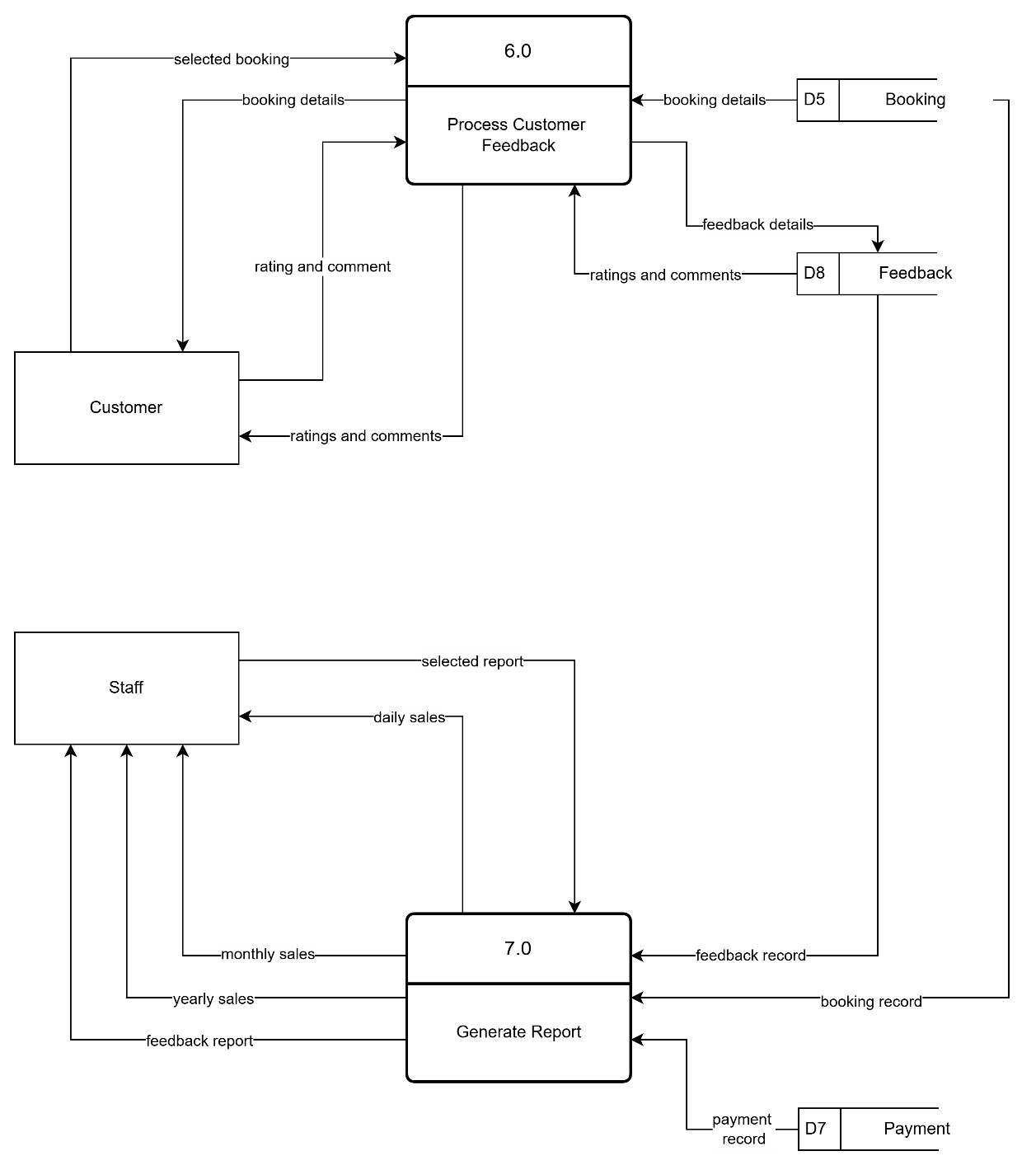


Figure 4.5: Level 0 of DFD for Process 6 and 7

#### Level 1

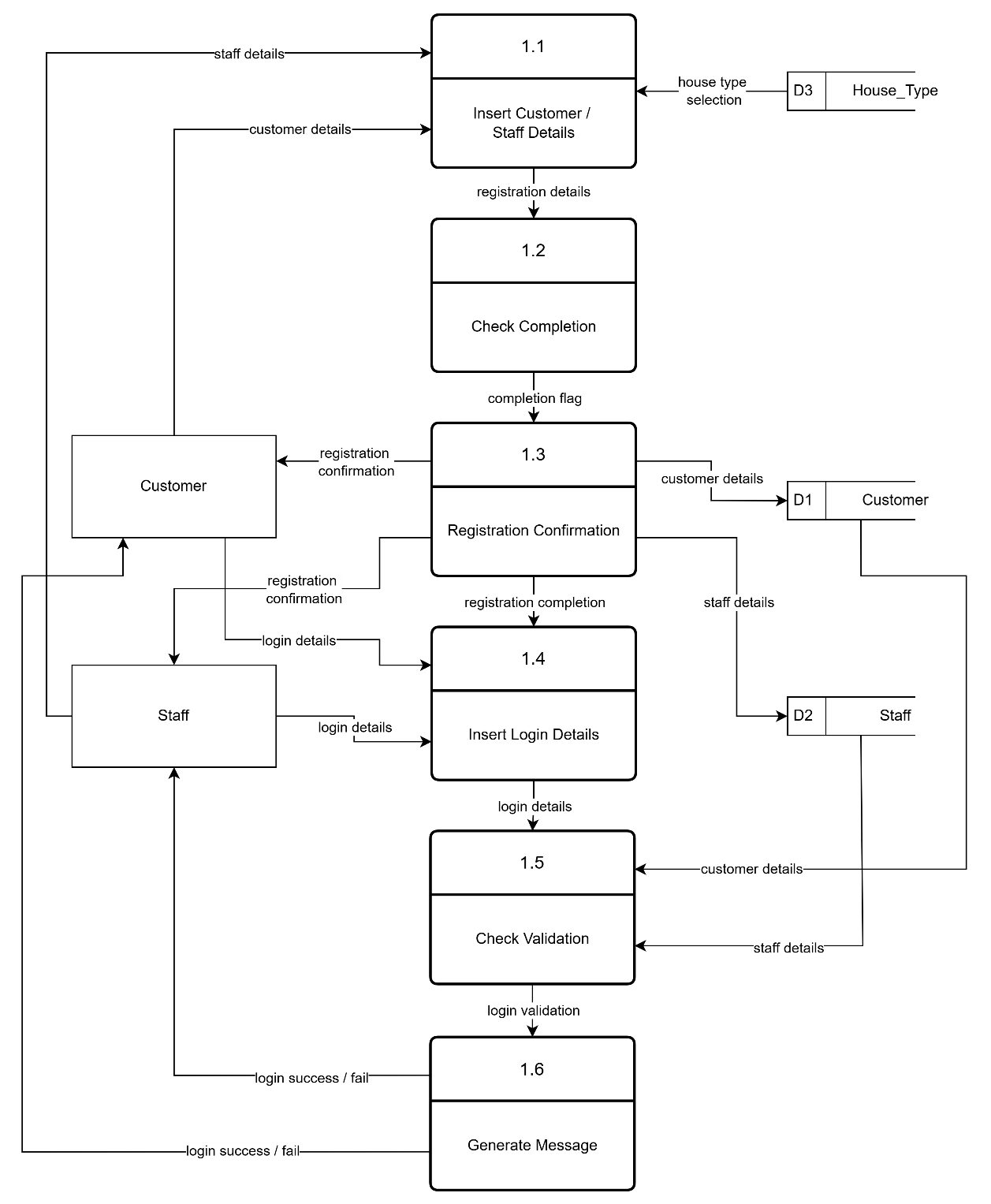


Figure 4.6: Level 1 of DFD for Process 1

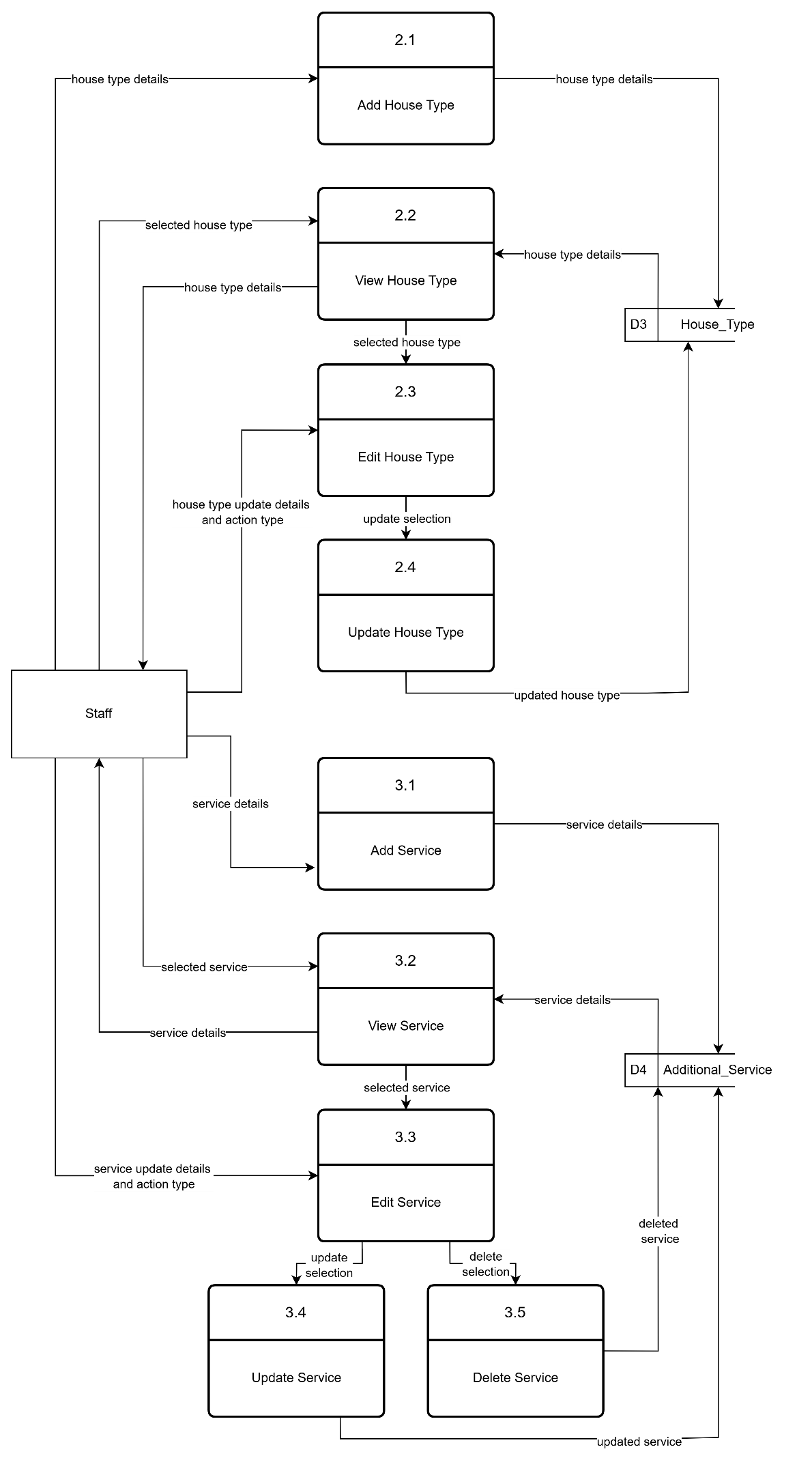


Figure 4.7: Level 1 of DFD for Process 2 and 3

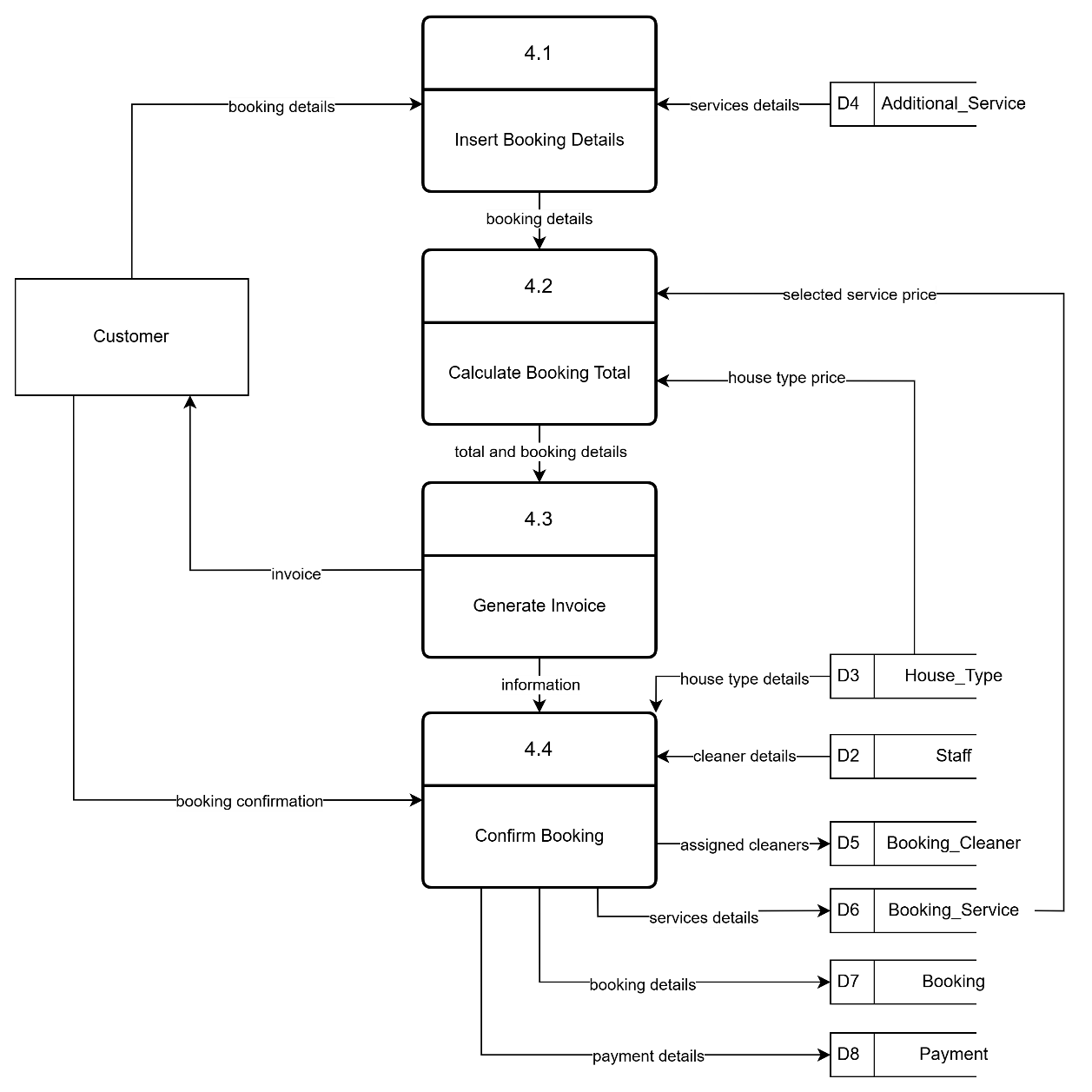


Figure 4.8: Level 1 of DFD for Process 4

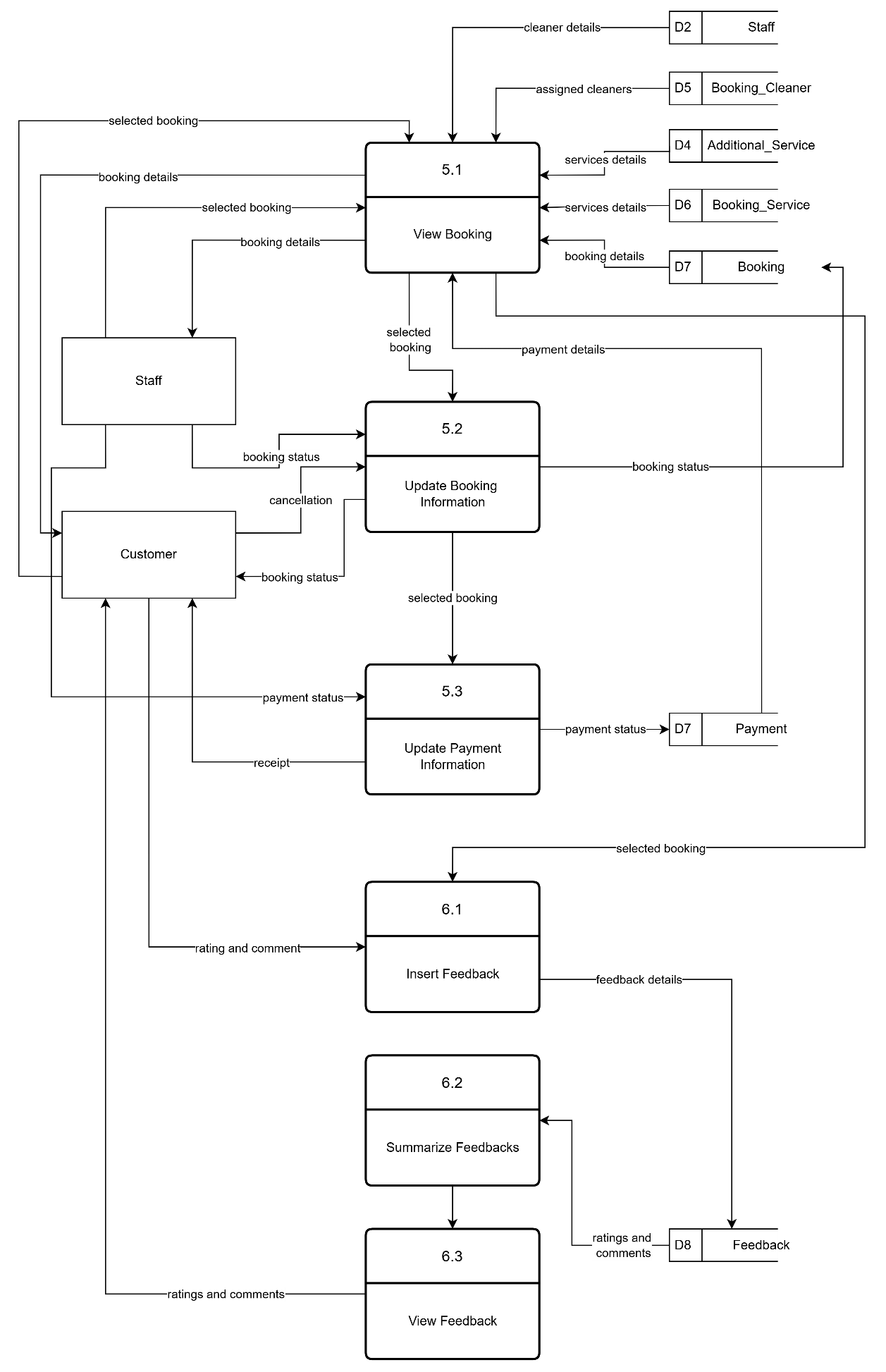


Figure 4.9: Level 1 of DFD for Process 5 and 6

## Detailed Design

Detailed design focuses on the technical implementation of the system’s components. It transforms the high-level architecture into a concrete blueprint by defining the internal logic, data structures, and interactions among system modules. This phase ensures that each functional requirement is supported by an efficient and scalable technical structure.

### Entity Relationship Diagram (ERD)

The Entity Relationship Diagram (ERD) defines the structure of the HygieiaHub database and illustrates the relationships between its core entities. Key entities include Customers, Bookings, House\_Types, Services, Staff, and Feedback. The ERD ensures that data is stored in a normalized format, reducing redundancy and supporting efficient data retrieval. Relationships such as one-to-many (e.g., one user can make many bookings) and many-to-one (e.g., multiple services linked to one booking) are clearly represented. By visualizing how data is connected, the ERD plays a crucial role in guiding database implementation and maintaining data integrity throughout the system.

#### Business Rules

1. A registered customer can make none or many bookings. While a booking can be made by only one customer.
2. A customer can have one type of house while a type of house can be owned by none or many customer.
3. A booking may be for only one type of house while a type of house may have none or many type of house.
4. A booking can have zero or many additional service while an additional service may be in zero or many bookings.
5. A booking may have one or many cleaners while a cleaner may have one or many bookings.
6. A booking can have only one payment and a payment can have only one booking.
7. A feedback may be referred to only one booking and a booking may have none or only one feedback.

#### ERD

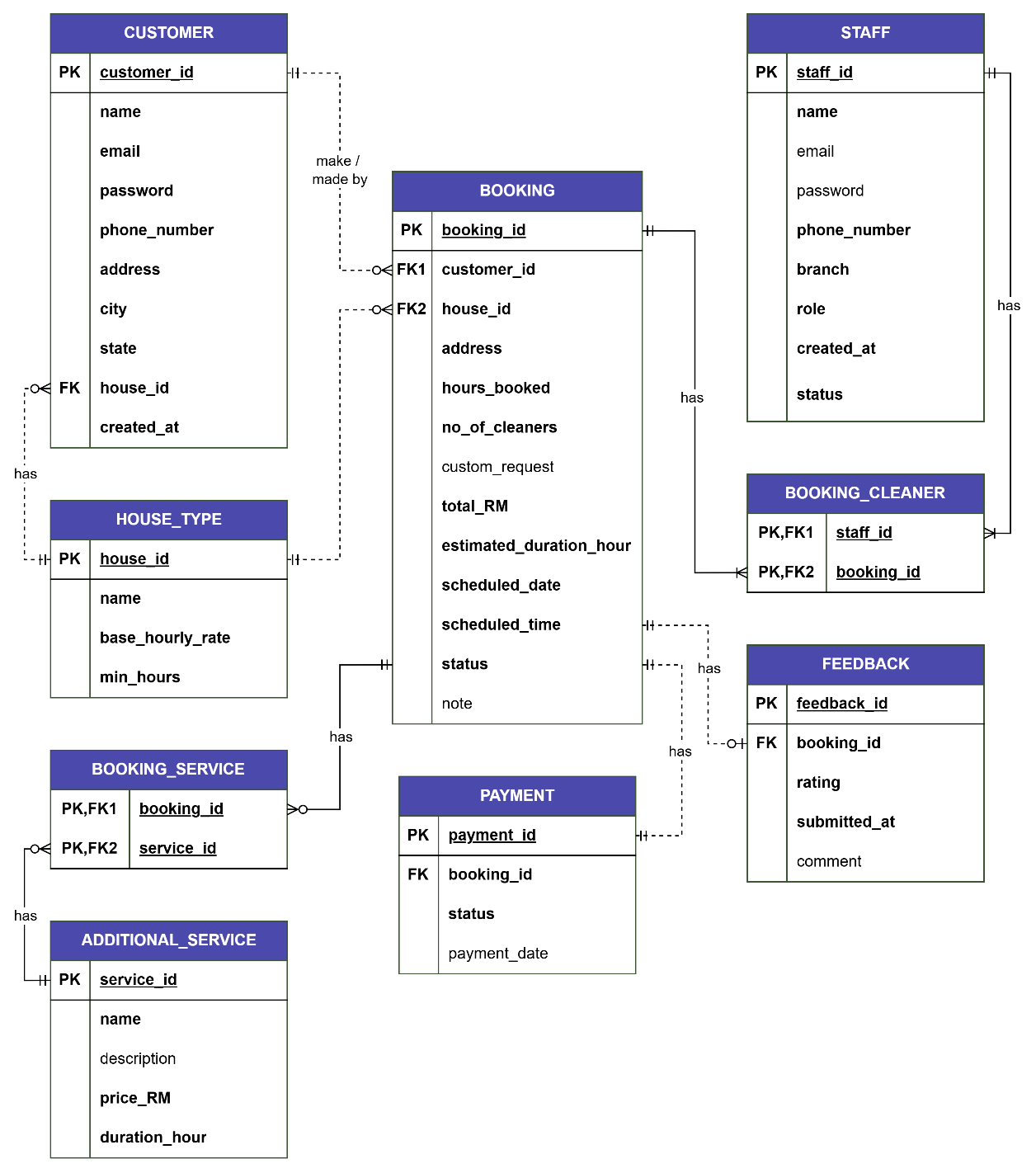


Figure 4.10: ERD

### Data Dictionary

The data dictionary provides a comprehensive definition of all database fields used within the HygieiaHub system. It details each table's attributes, including field names, data types, non-empty values, constraints (such as primary keys and foreign keys), and default values. This documentation ensures consistency and clarity in database design and development. By referencing the data dictionary, developers and maintainers can understand how data is stored, validated, and related across tables minimizing errors and supporting reliable system performance. It also serves as an essential guide for future updates or integration efforts involving the system’s backend.

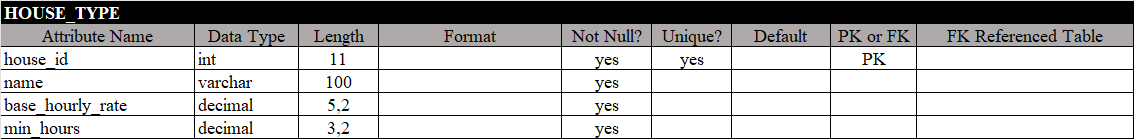


Figure 4.11: Data Dictionary of Table house\_type

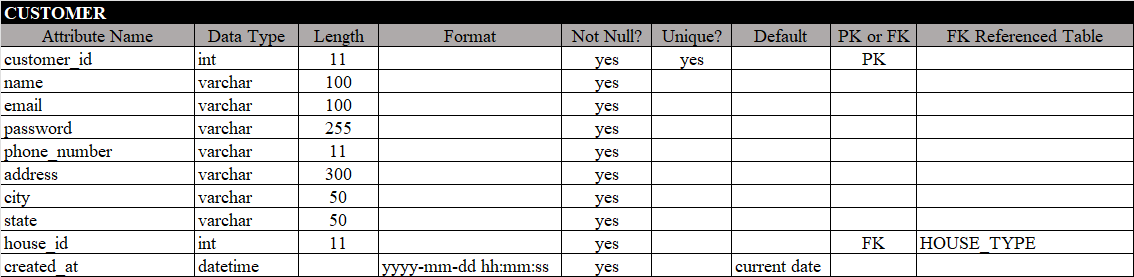


Figure 4.12: Data Dictionary of Table customer

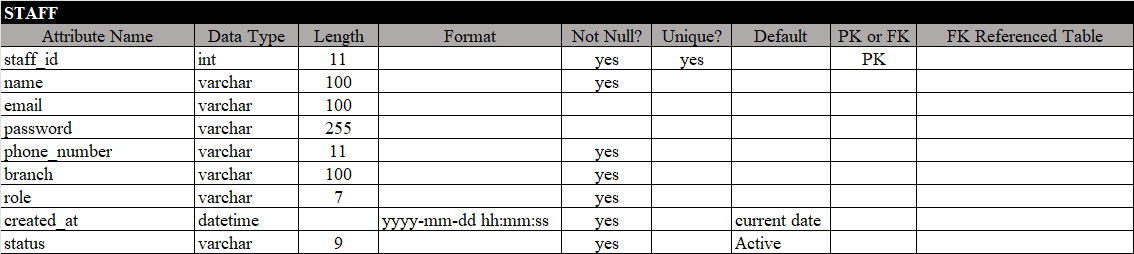


Figure 4.13: Data Dictionary of Table staff

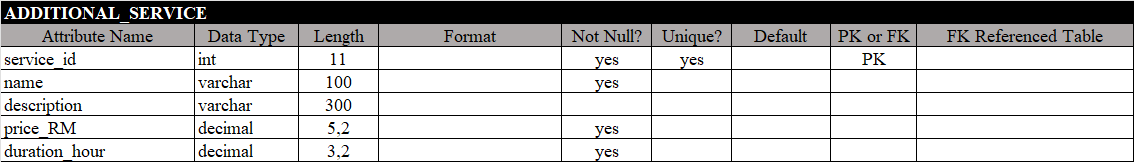


Figure 4.14: Data Dictionary of Table additional\_service

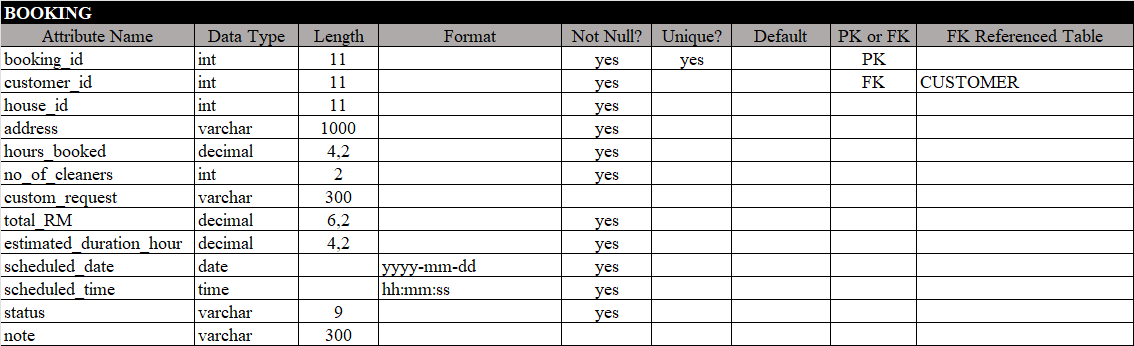


Figure 4.15: Data Dictionary of Table booking

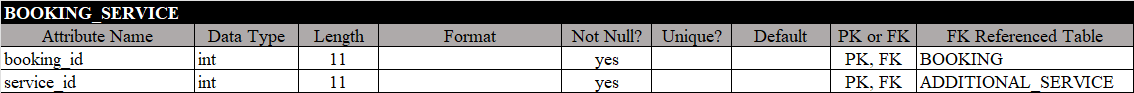


Figure 4.16: Data Dictionary of Table booking\_service

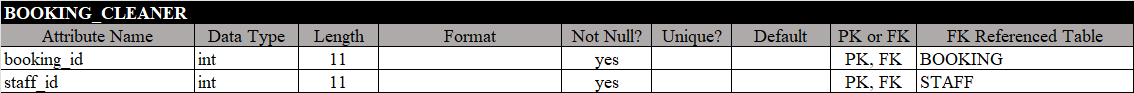


Figure 4.17: Data Dictionary of Table booking\_cleaner

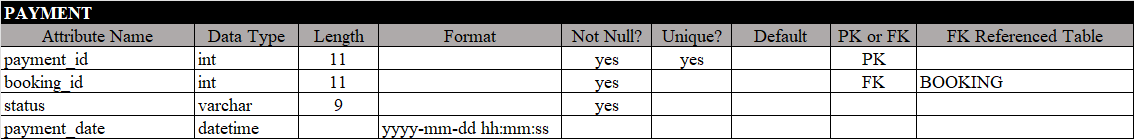


Figure 4.18: Data Dictionary of Table payment

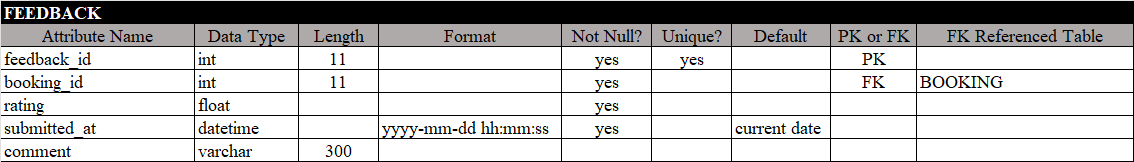


Figure 4.19: Data Dictionary of Table feedback

### Interface Design

Interface design focuses on how users interact with the system visually and functionally. It plays a vital role in ensuring usability, accessibility, and a smooth user experience. In HygieiaHub, the interface is designed to be simple and intuitive, aligning with the system’s goal of providing a streamlined booking process. Key interfaces include pages for registration, login, service selection, booking details, and feedback submission. Each screen is structured to minimize user confusion and guide actions clearly through form inputs, buttons, and navigation elements.

#### Interface Structure

The interface structure maps out the layout and hierarchy of the system’s user interfaces, illustrating how screens are organized and how users navigate between them. It shows the main entry points for both customer and staff roles, such as login, dashboard, booking form, and management panels. This structure acts like a sitemap, ensuring a logical flow between pages starting from login or registration, moving to service selection, and proceeding to booking confirmation and feedback. Designing this flow in advance helps maintain consistency across the system and ensures that users have a coherent and guided experience throughout their interaction with the platform.

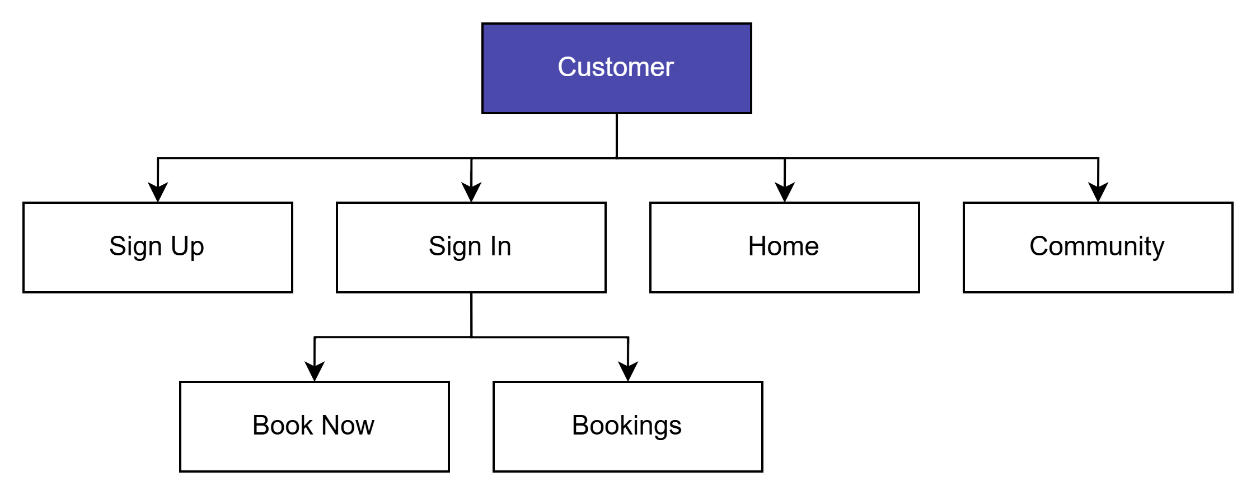


Figure 4.20: Interface Structure for User Customer

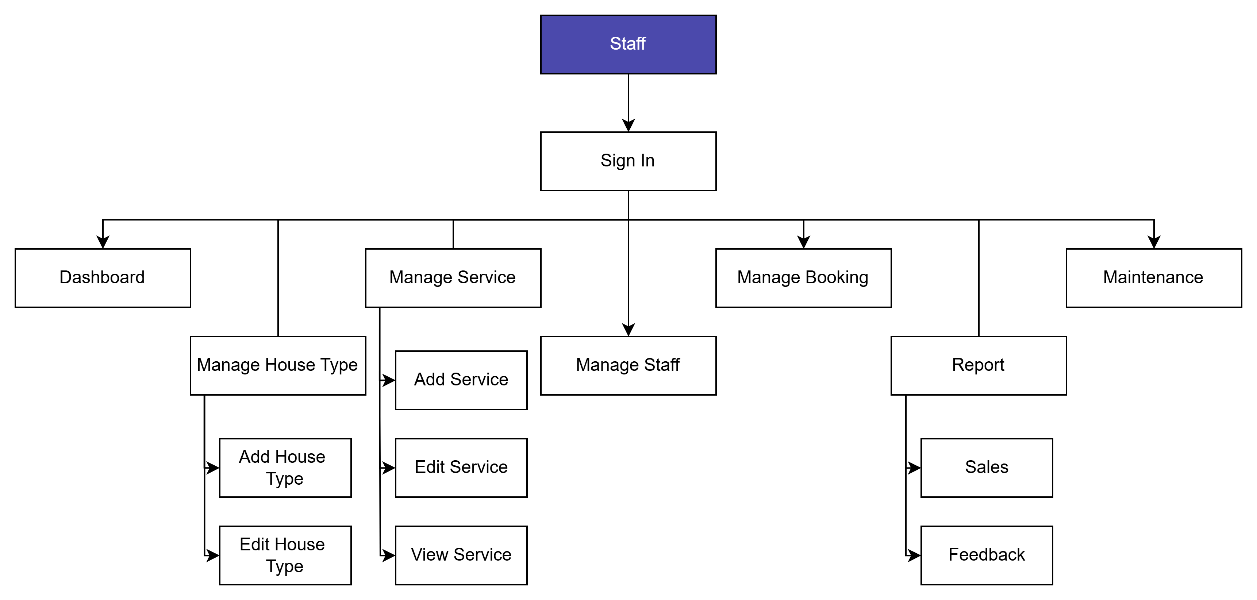


Figure 4.21: Interface Structure for User Staff

#### Interface Design – Input

Input interface design refers to screens where users enter data into the system. In HygieiaHub, these include registration and login forms, service booking forms, and feedback submission pages. These interfaces are designed to be user-friendly, with clear labels, appropriate input fields, and helpful validation messages to guide the user and reduce input errors. For example, the booking form allows customers to select service types, booking dates, house types, and add optional requests which are all structured in a step-by-step layout to simplify the process.

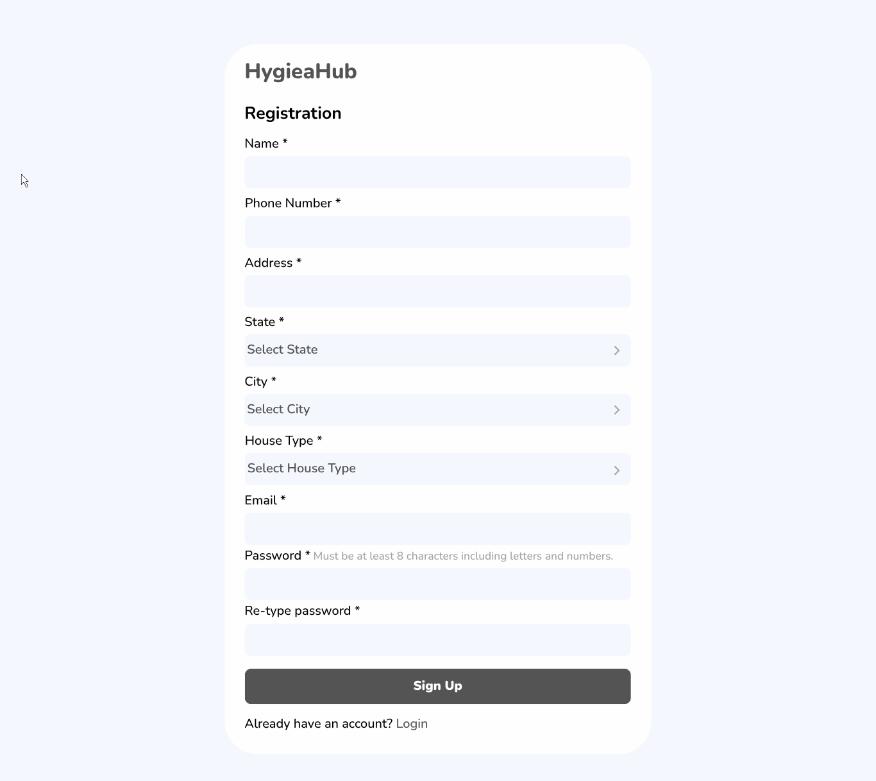


Figure 4.22: Customer - Sign Up Page

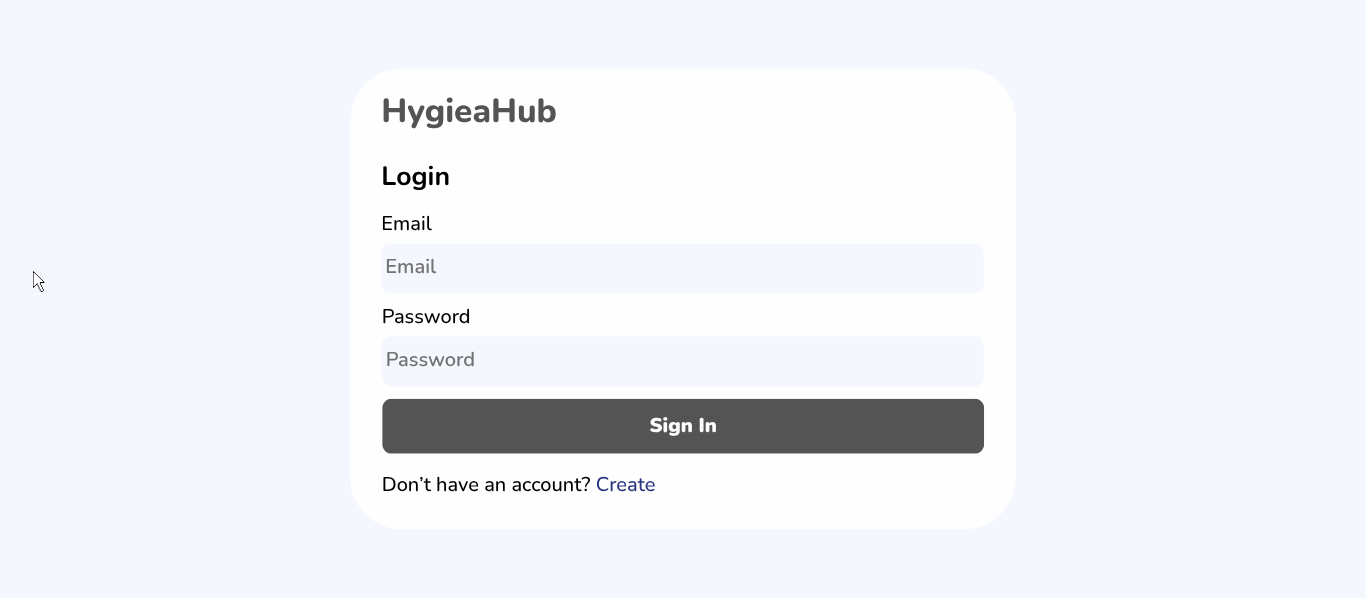


Figure 4.23: Customer & Staff - Sign In Page

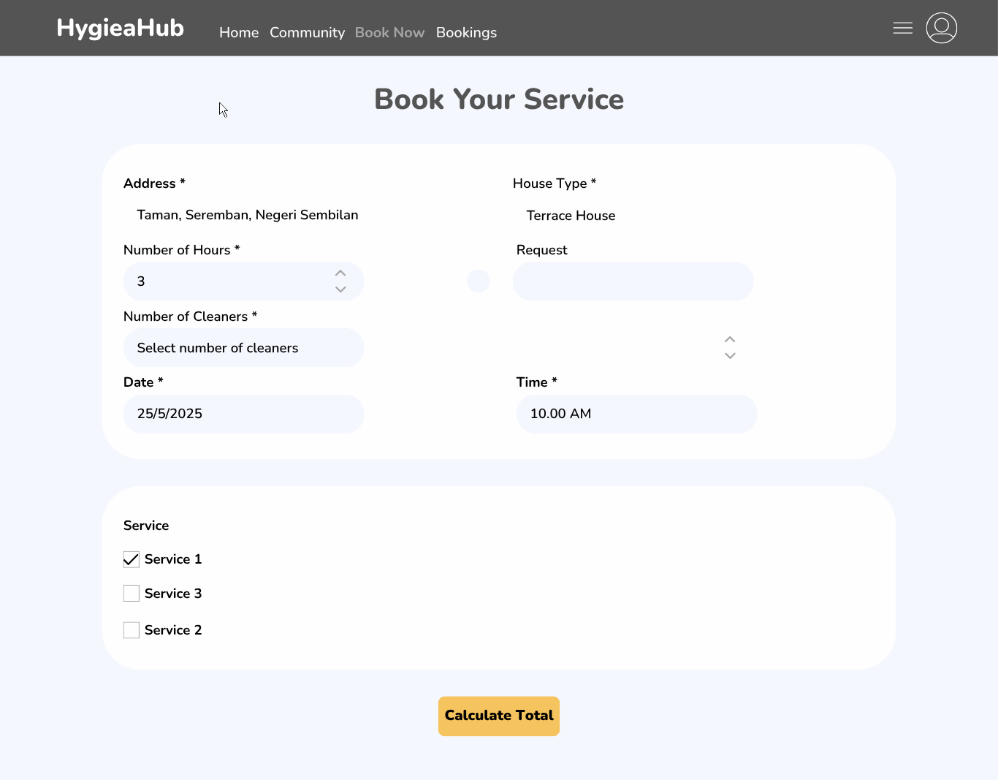


Figure 4.24: Customer - Book Now Page

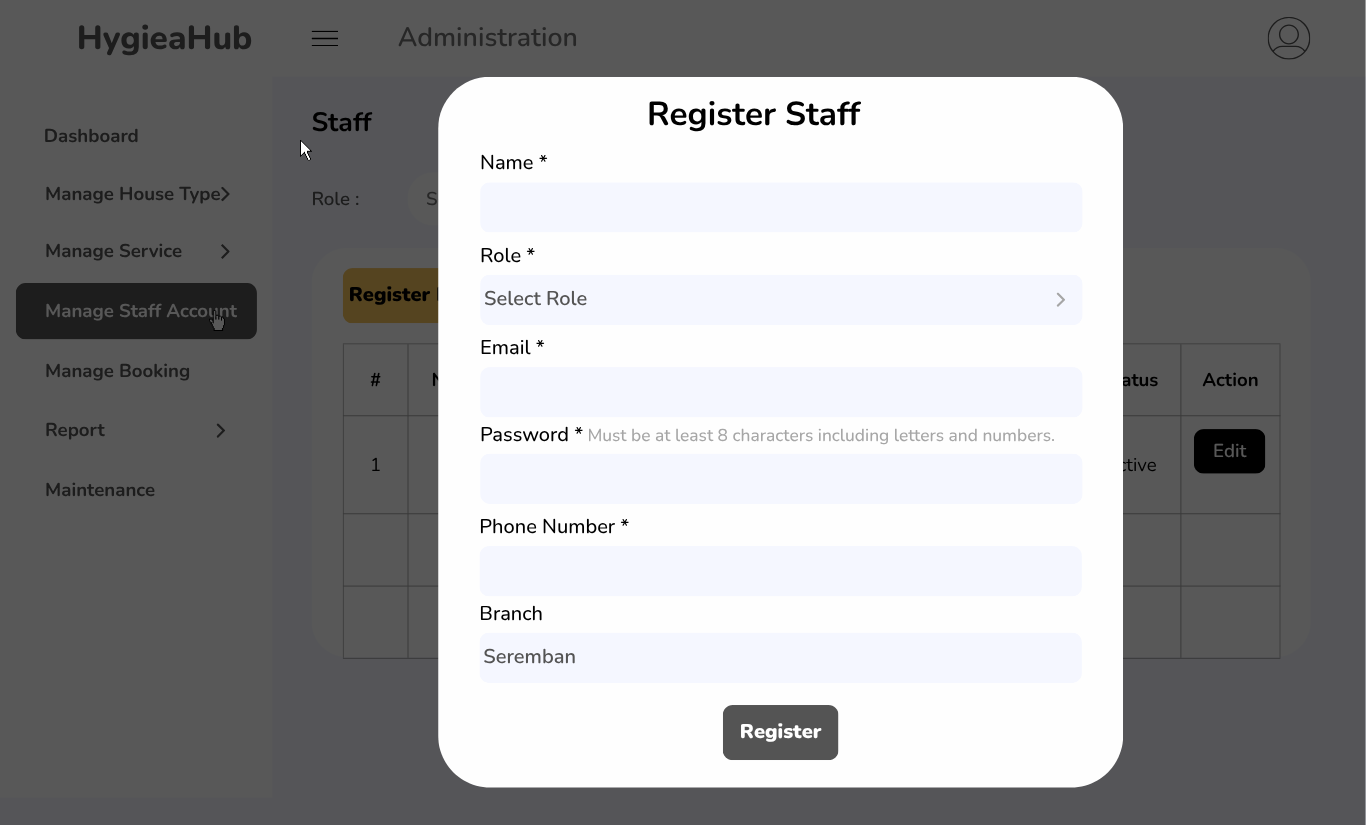


Figure 4.25: Staff - Register Staff Page

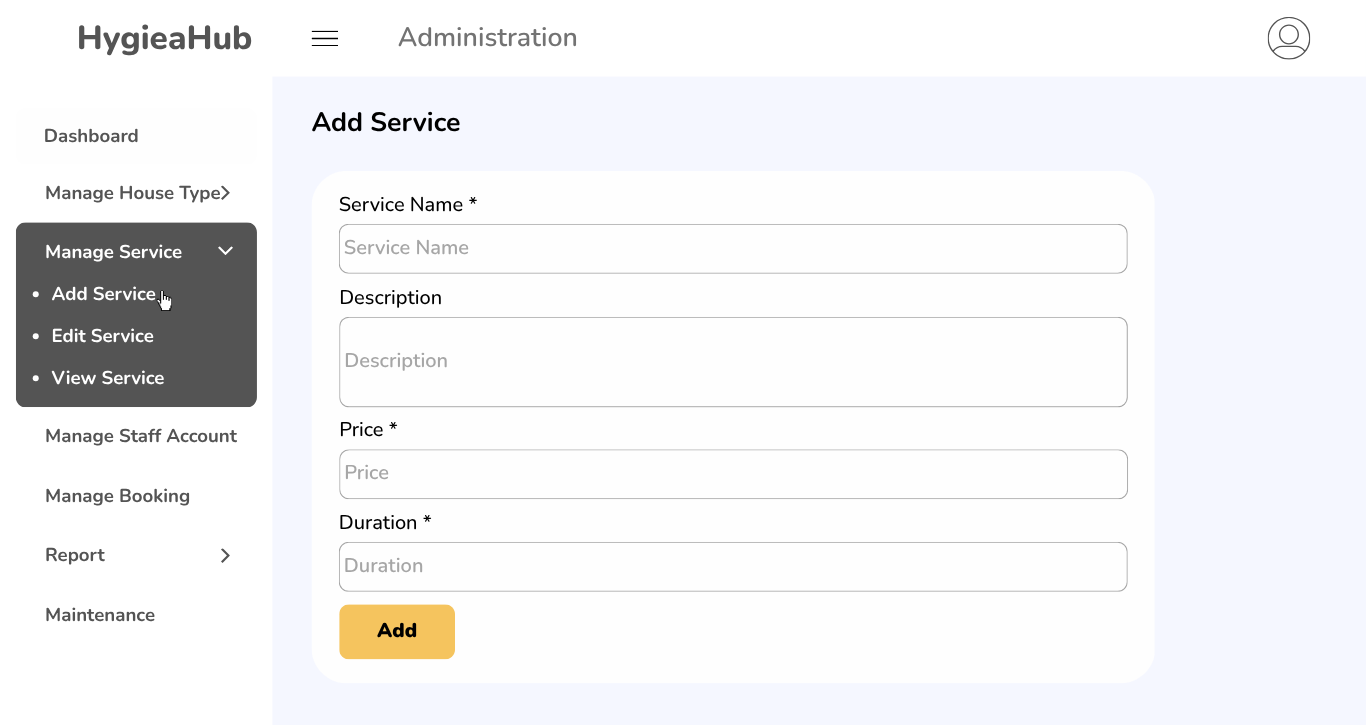


Figure 4.26: Staff - Add Service Page

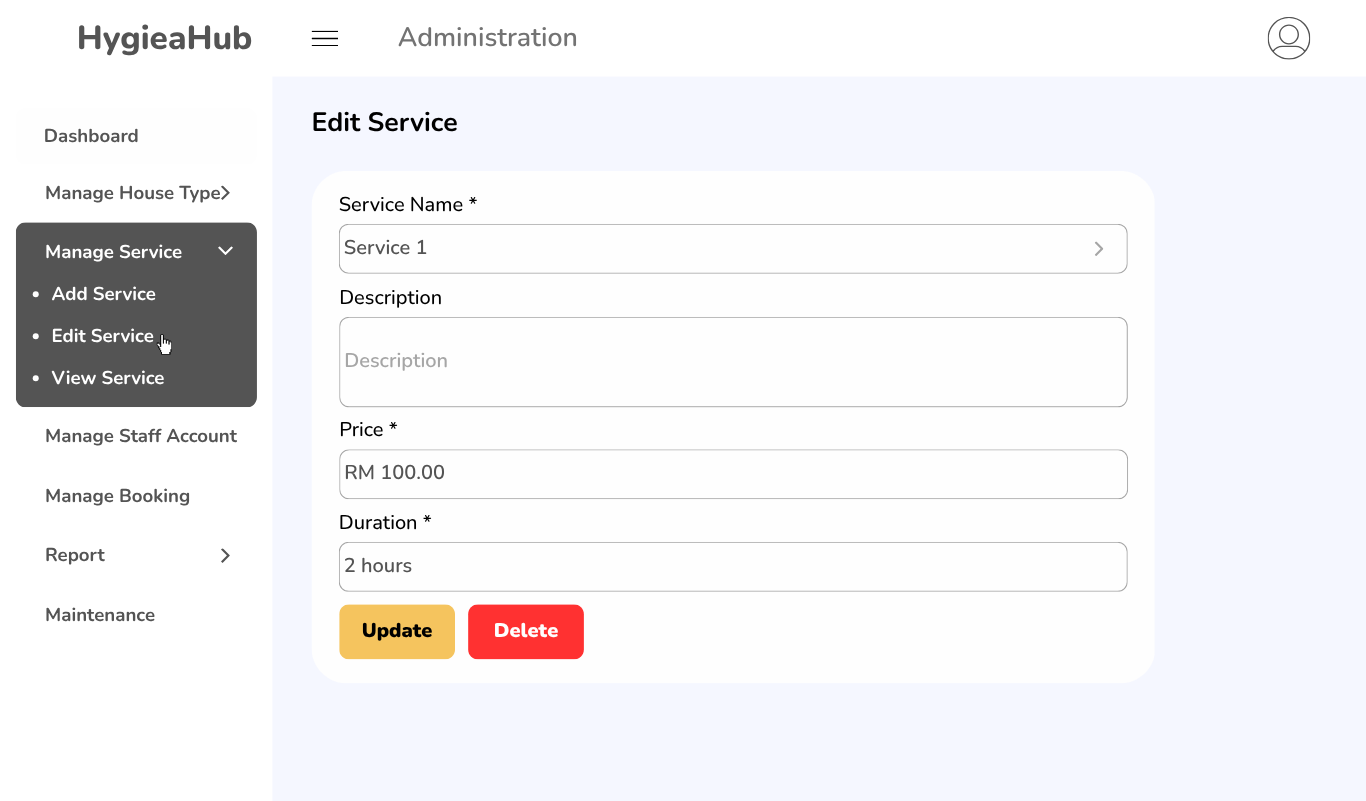


Figure 4.27: Staff - Edit Service Page

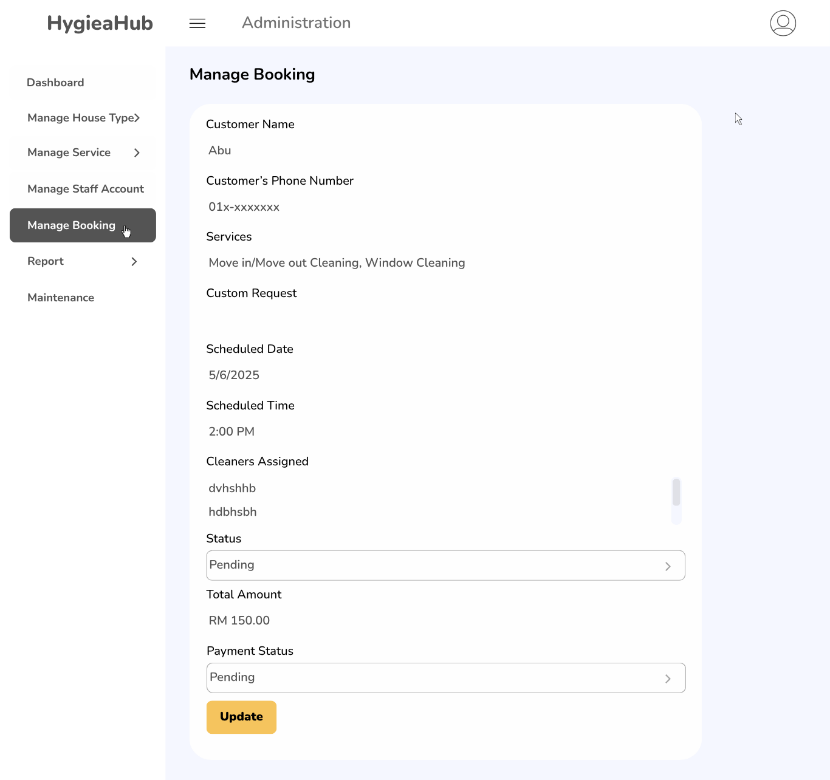


Figure 4.28: Staff - Manage Booking Page for editing

#### Interface Design – Output

Output interfaces present users with information processed by the system in a clear and organized format. In HygieiaHub, these include booking summaries, status tracking pages, service details, and feedback overviews. For staff, output screens include booking management panels and sales or feedback reports. These interfaces are structured to ensure clarity, using tables, status indicators, and summarized views to help users quickly understand the information presented. The goal is to support informed decision-making and improve overall transparency of service operations.

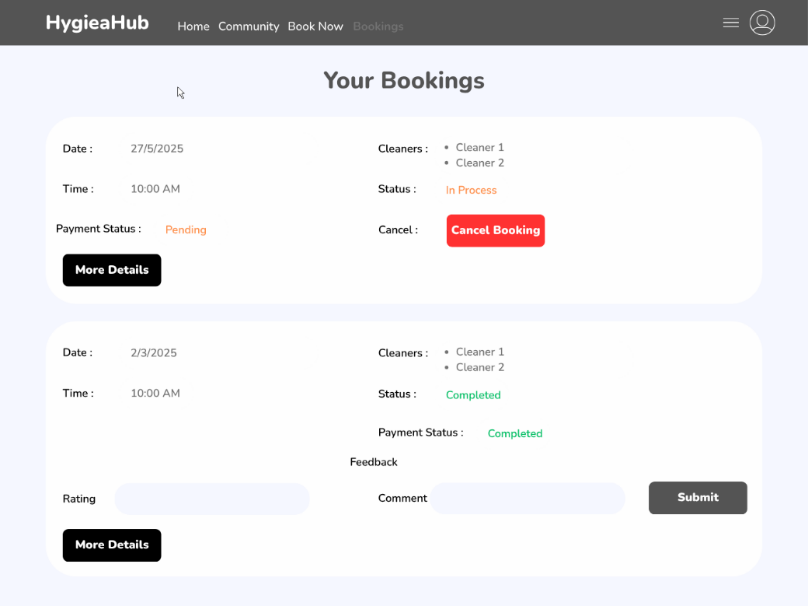


Figure 4.29: Customer - Bookings Page

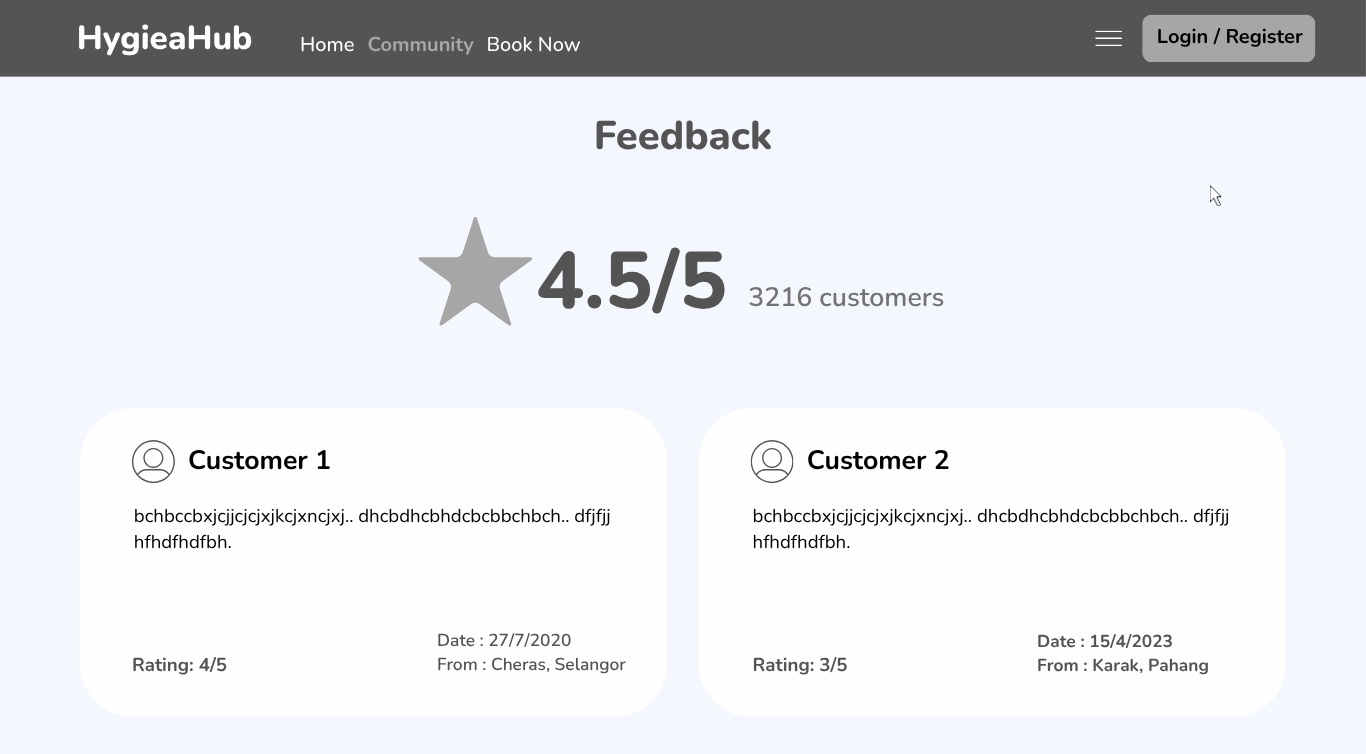


Figure 4.30: Customer - Community Page

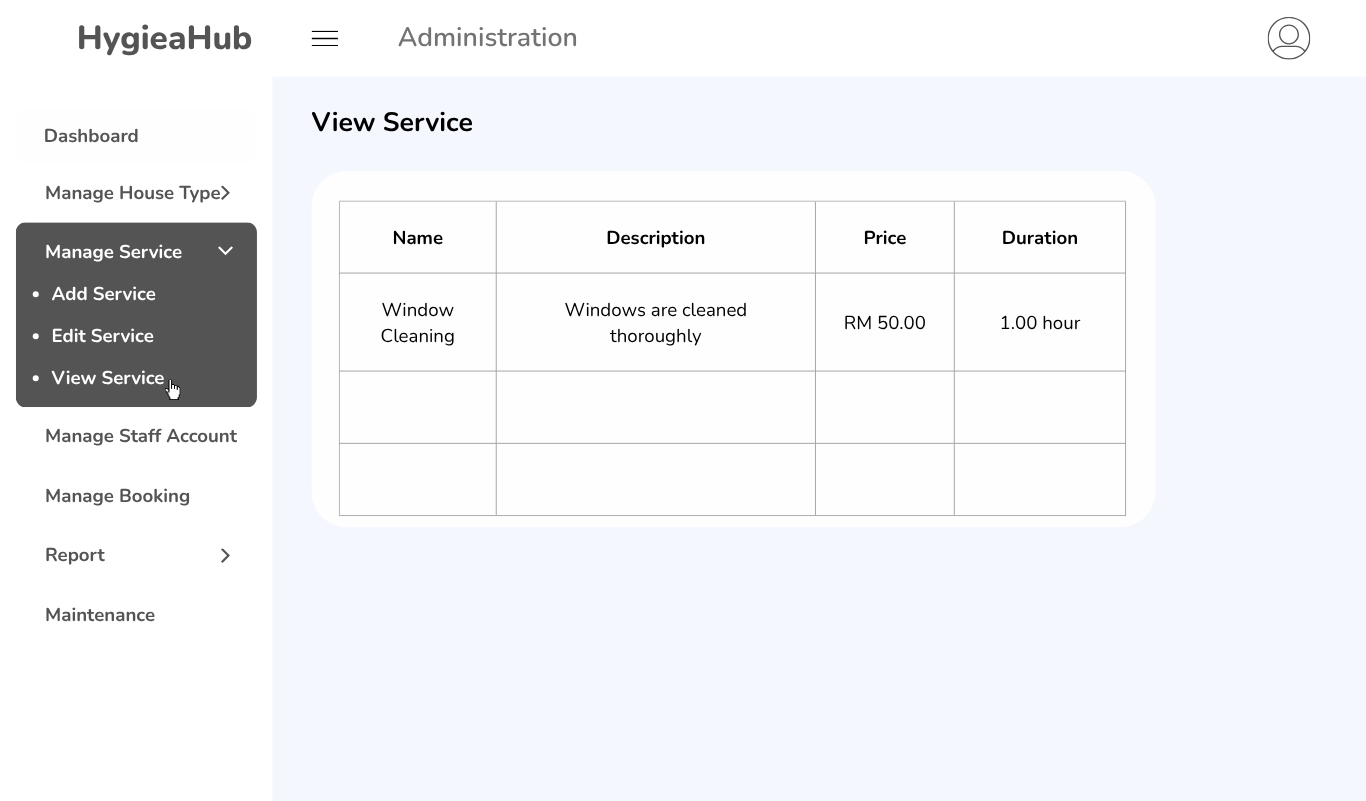


Figure 4.31: Staff - View Service Page

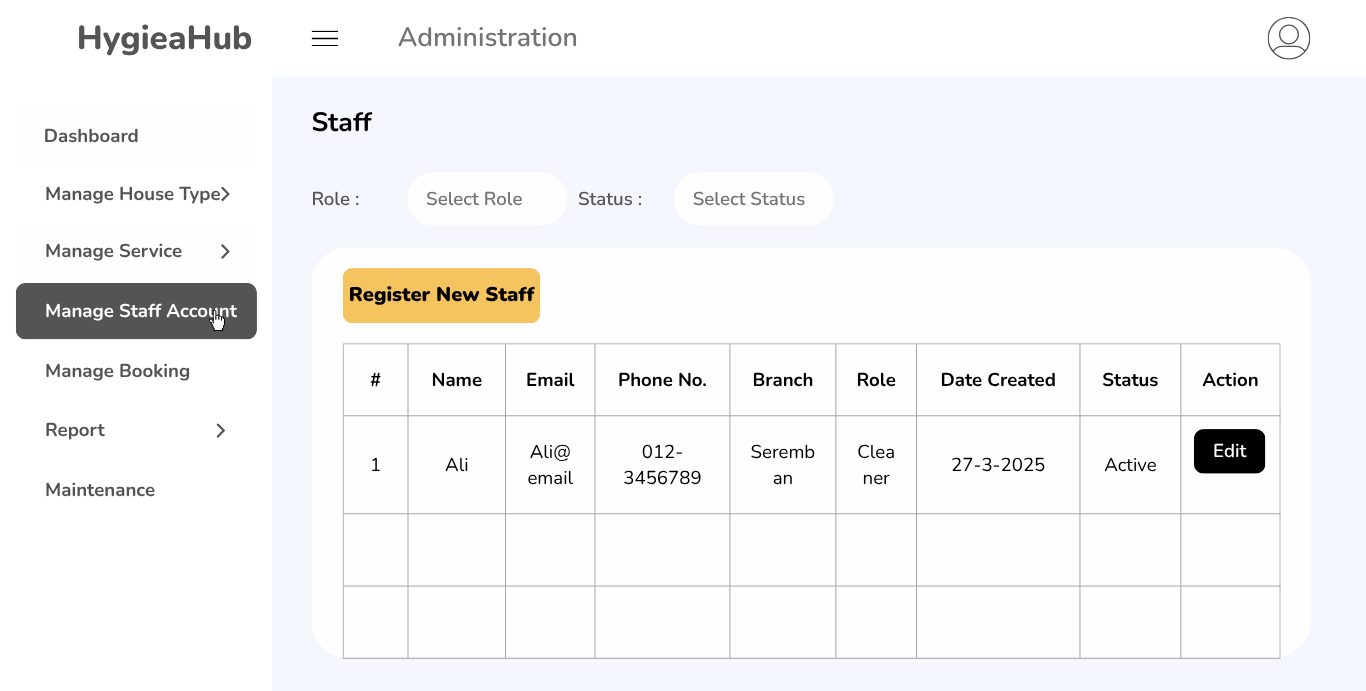


Figure 4.32: Staff - Manage Staff Page

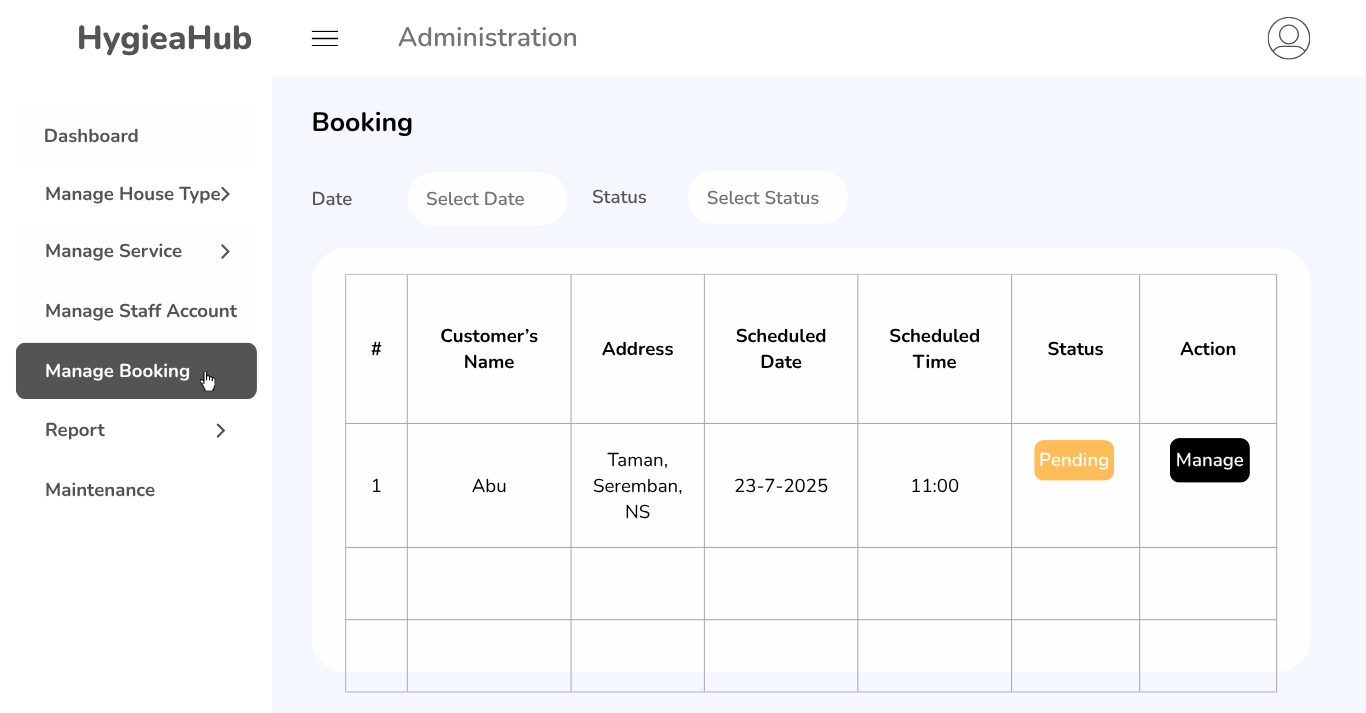


Figure 4.33: Staff - Manage Booking Page

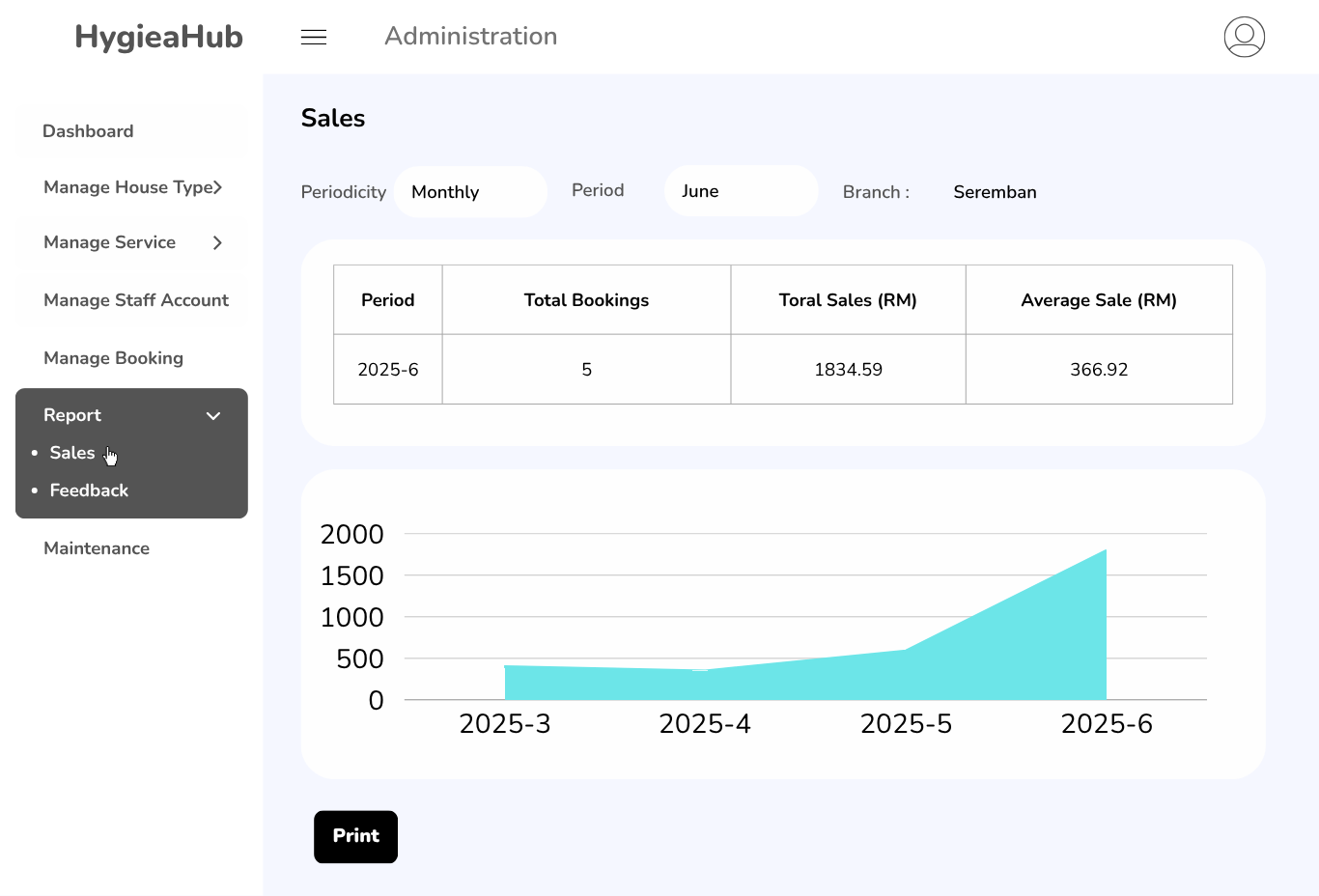


Figure 4.34: Staff - Sales Report Page

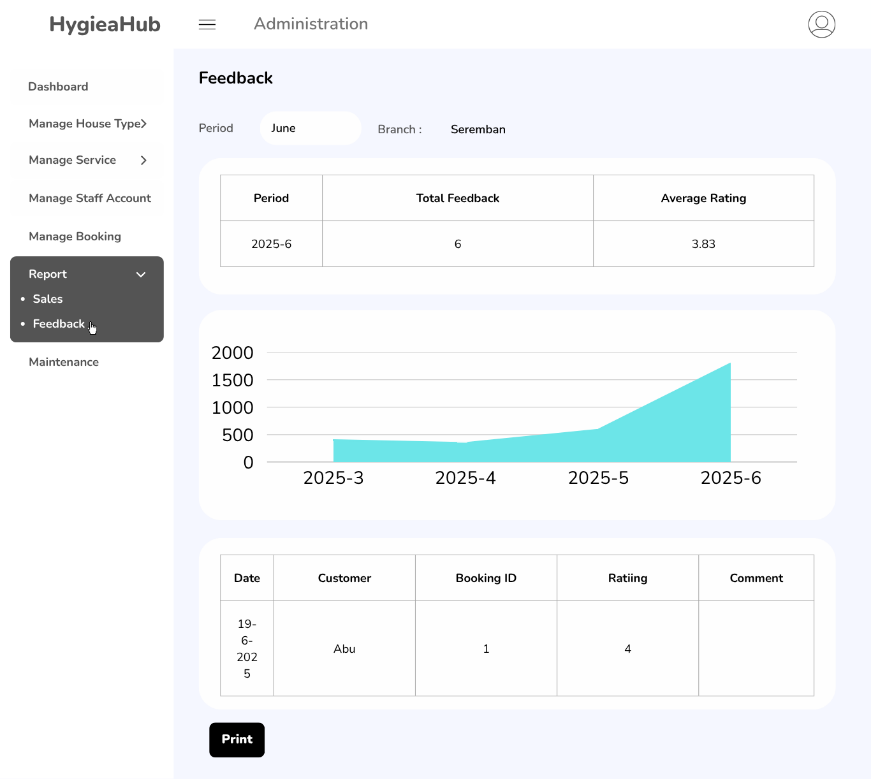


Figure 4.35: Staff - Feedback Report Page

## Conclusion

In conclusion, this chapter provided both a high-level and detailed view of the HygieiaHub system design. Through system flowcharts, DFDs, ERD, and wireframe-based UI designs, the structure and interactions of system components have been clearly outlined. The data dictionary further supports the implementation process by offering a well-documented reference for database fields. This comprehensive design phase lays the groundwork for effective system development and implementation in the next chapter.

# IMPLEMENTATION

# TESTING

# PROJECT CONCLUSION

# references

Lee, S., Moon, M., Kim, D., & Yi, M. Y. (2015). *Antecedents and consequences of mobile phone usability: Linking simplicity and interactivity to satisfaction, trust, and brand loyalty*. Information & Management, *52*(3), 295–304.

Brown, D. H., & Lockett, N. (2004). *Potential of critical e-applications for engaging SMEs in e-business: A provider perspective*. European Journal of Information Systems, *13*(1), 21–34.

Halim, Y., Abdullah, A., & Tuan Mohd Yasin, S. N. (2020). *Creation of final year project administration system using PHP script and MySQL*. Journal of Counseling and Educational Technology, 5(1), Article 157.