

Philadelphia University

Department Of Information Technology

Home services application

Submitted By:

Majd Al-Deen Murad Al-Bawwab

Ibrahim Hassan Baroudi

Ahmad Bader Assa'd Abbas

UNI-NUMBER: 202010694

UNI-NUMBER: 202010603

UNI-NUMBER: 202010249

Supervised By:

Dr. Mohammad Taye

Table of Contents

CHAPTER I	4
Introduction	4
1.1 Introduction	4
1.2 Background:	4
1.3 Problem Statement:	4
1.4 Limitations:	4
1.5 Objectives:	5
1.6 Solution:	5
1.7 Scope:	5
1.8 Feasibility:	5
CHAPTER II	6
LITERATURE REVIEW	6
2.1 Introduction:	6
2.2 Review of Existing System:	6
2.3 Comparison between Available:	10
CHAPTER III	11
METHODOLOGY AND WORK PLAN	11
3.1 Introduction	11
3.2 System development methodologies (Waterfall development model):	11
3.3 Requirements gathering techniques:	11
3.4 Project Plan Gantt chart:	12
3.5 Development Tools:	13
CHAPTER IV	14
PROJECT DESIGN and SPECIFICATION	14
4.1 Introduction	14
4.2 System Specification	14
4.2.1 Functional requirements	14
4.2.2 Non-Functional requirements	15
4.3 Class Diagram	16
4.4 Database Design	17
4.4.1 Entity-Relationship Diagram (ERD)	18

4.4.2 Schema	19
4.5 User Interface	20
CHAPTER 5	21
SYSTEM IMPLEMENTATION	21
Introduction.....	21
5.1 Implementation of the Home Service Application:.....	21
CHAPTER 6	22
RESULT AND DISCUSSION.....	22
6.1 Introduction.....	22
6.2 Data Collection and Research	22
6.3 System Testing and User Feedback	22
CHAPTER 7	24
RECOMMENDATIONS AND CONCLUSION.....	24
7.1 Introduction.....	24
7.2 Conclusion	24
7.3 Limitations and Recommendations for future work	24
APPENDICES.....	25

CHAPTER I

Introduction

1.1 Introduction

The project we are working on, “Intelligent platform for scheduling skilled professionals for household maintenance,” is a mobile application designed to streamline the process of booking skilled professionals for household maintenance tasks. Whether fixing electrical issues, repairing plumbing, or refreshing the paint, it connects users with experienced service providers in their area. This project aims to revolutionize the way people find and hire skilled professionals for home maintenance tasks, providing a convenient platform for seamless booking, reliable services, and customer satisfaction.

1.2 Background:

The spark of this idea came from one of our members while trying to find a suitable man to help with the electricity of his house, frustrated by the inefficiencies of the whole process, he envisioned an easier way to connect homeowners with skilled experts. Once he shared his idea with us, we had a brainstorming session where we decided to apply our knowledge in the tech field to have an app that achieves our desired vision.

1.3 Problem Statement:

The traditional method of acquiring a skilled professional is slow, unreliable, and untrustworthy. With so many undecided factors, it leaves both the hiring and the hired in an ambiguous state where time, money, and availability are always fluctuating. We aim to solve this problem by introducing a modern platform that saves time, and effort and offers transparency for both parties.

1.4 Limitations:

As is the nature of this project, the sheer scale of it would be a problem, especially since we are a three-man team, so manpower would be our real struggle, Geographical coverage would be the obvious continuation of our limitation as the app will be bound to certain small regions, restricting access for users in areas where service providers aren't available. Another one would be limited scope; our project primarily focuses on home maintenance tasks; therefore, it may not address other aspects of household management or broader property-related issues.

1.5 Objectives:

1. Create a user-friendly mobile application for users to find the right man for the job.
2. Develop and launch a functional app within three months, focusing on essential features and core functionality.
3. Lay the groundwork for sustainable growth and scalability
4. Provide detailed reports for all parties to the agreed arrangement.
5. Enhance transparency and convenience for a customer that otherwise would have used conventional means.

1.6 Solution:

Our project introduces a comprehensive solution by developing a user-friendly mobile application that utilizes the newest technologies in the field to develop our platform. It encompasses three main interfaces for customers, professionals, and administrative personnel. Features include booking a service, automated notifications, the ability to cancel said service if it meets our criteria, and user feedback. This solution aims to improve accuracy, efficiency, and transparency on all fronts.

1.7 Scope:

We aim to develop a mobile application that facilitates the booking of home maintenance services. The project will focus on creating a Minimum Viable Product (MVP) with essential features such as user registration, service provider onboarding, booking scheduling, and basic communication functionalities. The scope also includes limited geographic coverage, targeting a campus area to manage resources effectively. Emphasis will be placed on user feedback, iterative improvement, and cost-efficient operations while adhering to academic timelines and budget constraints.

1.8 Feasibility:

The project is economically feasible due to its nature, which doesn't require much from the hardware side, aligning well with budget constraints. The university's operational support for projects on paper enhances their operational feasibility. A one-semester timeline allows for a well-considered and thorough prototype, with associated low technical and financial risks.

CHAPTER II LITERATURE REVIEW

2.1 Introduction:

In recent years, there has been a growing demand for convenient and efficient home services. However, there is a noticeable lack of dedicated applications that cater to this need. This gap in the market has inspired researchers and developers to explore innovative solutions to help individuals easily and reliably order home services.

It is incredibly important to review pre-existing solutions that aim to solve the same problem as ours, as to not only avoid blatant copying and plagiarism, but also to expand and improve said existing solutions and learn from their mistakes.

2.2 Review of Existing System:

There has been several systems that aim to provide a suitable application to what we are looking for, the first of these projects is (Maharah)

- (1) **Maharah:** Is an application to facilitate access to home maintenance services and construction work in less than a minute? it links service seekers from homeowners or companies to professional service providers.

Advantages:

- Possibility to request your service now to make necessary repairs, or later for future dates.
- It takes less than a minute to get the service, to receive information from the coordinator about customer appointment in recognition of customer location.
- Get a free cleaning service after you finish working.
- The customer gets a one-day work warranty, not including spare parts.
- The customer can submit complaints and suggestions to the management through instant conversations with the application or through the contact us page on the site or by sending you a Scoop account to serve customers in X application.

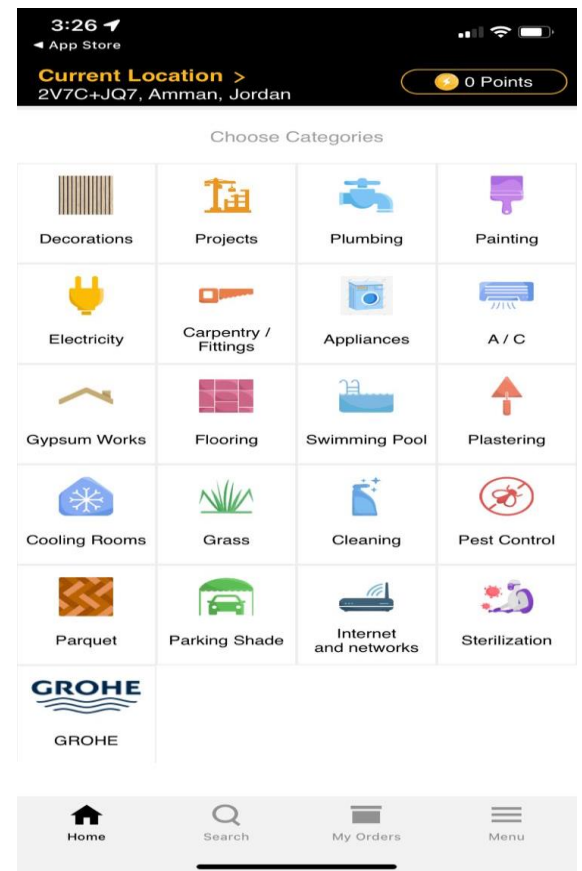


Figure (2-1): Maharah user interface

Disadvantages:

- Registry does not respond as it should be.
- Delay in response, lack of service, few service providers, few options.
- Provider takes more than it deserves on the services.

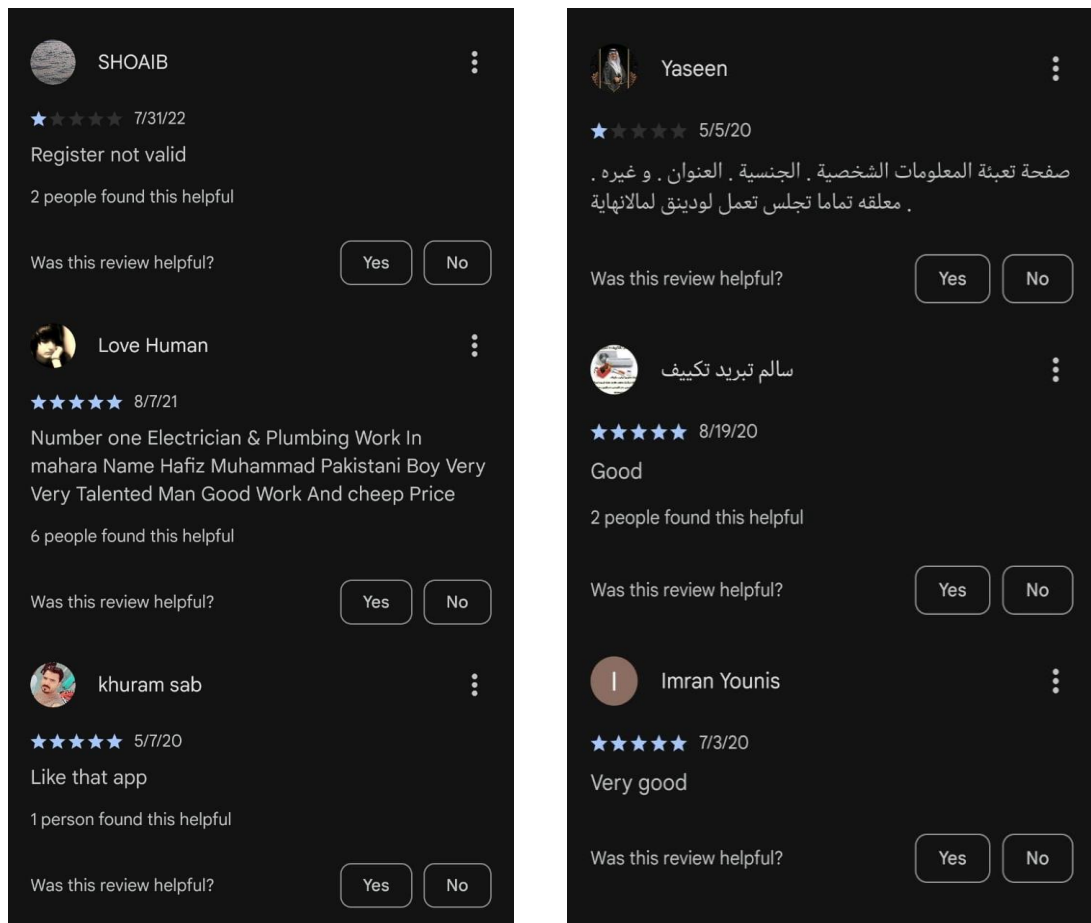


Figure (2-2): Maharah user reviews

- (2) **Fani:** An application that makes it easy for you to get a home maintenance technician easily, by connecting the user to the service provider required directly and in a simple way. Including: electricity, plumbing, air conditioning, electrical appliances, satellite sets, paints, furniture transfer, pest control, cleaning, carpentry.

Advantages:

- Getting a technician requires two steps only.
- Allows the customer to choose the appropriate technician according to a set of options.
- It works with both platforms, (IOS) and (Android), Each platform support Arabic and English languages.

Disadvantages:

- Sometimes app does not work properly.
- There is no response.
- Sometimes you can't log in.

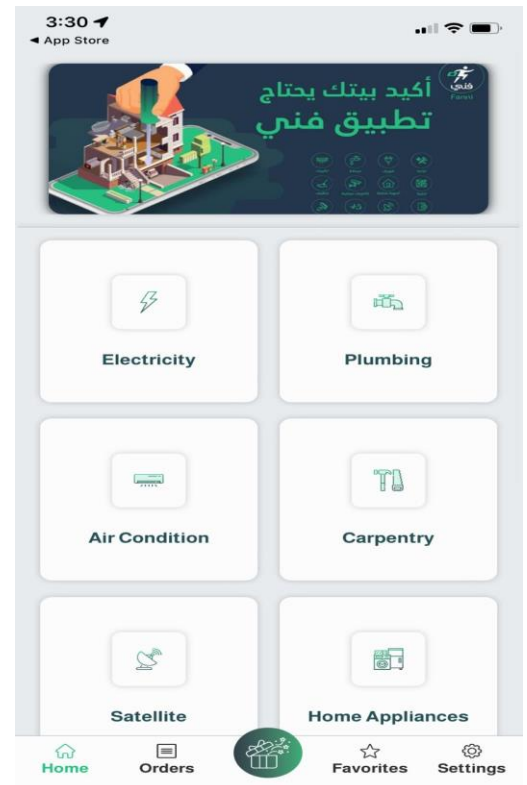


Figure (2-3): Fani user interface

- (3) **Ajeer:** It is an app that help customers through a mobile screen without having to look for maintenance services the traditional way. It offers a wide range of services such as plumbing, electrical work, carpentry, painting, and general repairs.

Advantages:

- Customers can find out about the approximate prices of the services.
- The possibility to request your service at a later date to make necessary repairs.

Disadvantages:

- This app is only available on the App Store for IOS platforms.
- Sometimes the app does not work properly.
- Some customers complain about some services not provided and some of them are not satisfied about the providers performance.



Figure (2-4): Ajeer user interface

2.3 Comparison between Available:

Table (2-1): Comparison between our system and pre-existing ones.

	Maharah	Fani	Ajeer	Our proposed System
Fraud Mitigation	<ul style="list-style-type: none"> -Transaction Monitoring. -Background Checks. 	<ul style="list-style-type: none"> -User verification. 	<ul style="list-style-type: none"> -Secure Payment Gateway. 	<ul style="list-style-type: none"> -Terms and Conditions -Rating and Review System
Functionality	<ul style="list-style-type: none"> -Geolocation Tracking 	<ul style="list-style-type: none"> -Feedback Loop 	<ul style="list-style-type: none"> -Two-Factor Authentication 	<ul style="list-style-type: none"> -Display Ratings and Reviews -Sorting and Filtering
Advantages	<ul style="list-style-type: none"> -Gets you free cleaning service after you finish working. 	<ul style="list-style-type: none"> -Getting a technician requires two steps only 	<ul style="list-style-type: none"> -The possibility to request your service at a later date to make necessary repairs 	<ul style="list-style-type: none"> -It offers multiple payment options -Its based in Jordan which none of the others are.
Disadvantages	<ul style="list-style-type: none"> -Difficulties in navigation inside the app (GUI) 	<ul style="list-style-type: none"> -Stability of the app 	<ul style="list-style-type: none"> -Unreasonable loading time 	<ul style="list-style-type: none"> -Privacy concerns -Scope

CHAPTER III

METHODOLOGY AND WORK PLAN

3.1 Introduction

Throughout this section we are going to discuss the methodology guiding the project, emphasizing a structured and sequential approach tailored to our project's characteristics and scope.

This lays the groundwork for more in-depth discussion of requirement gathering methods, a Gantt chart-based visual project plan, and necessary tools and technology in later parts.

Together, these components form a cohesive roadmap for the project's successful execution.

3.2 System development methodologies (Waterfall development model):

We opted for the Waterfall development model as the methodology for our project. This approach, known for its simplicity and clarity, ensures a systematic progression through all project phases. It involves distinct sequential stages like requirements gathering, design, implementation, testing, deployment, and maintenance.

Moreover, the Waterfall model is well-suited for our project's small-scale and less complex nature. Its well-defined roadmap helps minimize iteration risks, ensures predictable timelines and milestones, and promotes clear communication and collaboration. This, in turn, facilitates efficient development and successful implementation.

3.3 Requirements gathering techniques:

For gathering project requirements, We have chosen surveys as the primary method. They provide anonymity, ensuring honest feedback, and efficiently reach a broad audience. By thoughtfully crafting survey questions, We aim to gather both qualitative insights and quantitative data. Upon collecting responses, I'll meticulously analyze them to identify recurring themes and essential requirements.(look for attached file named survey) This comprehensive understanding will effectively guide the project's direction, ensuring alignment with our needs and preferences.

3.4 Project Plan Gantt chart:

Using a Gantt chart in a software graduation project offers a structured approach to project management, allowing us to effectively plan, track, and execute tasks within defined timelines.

Home services application

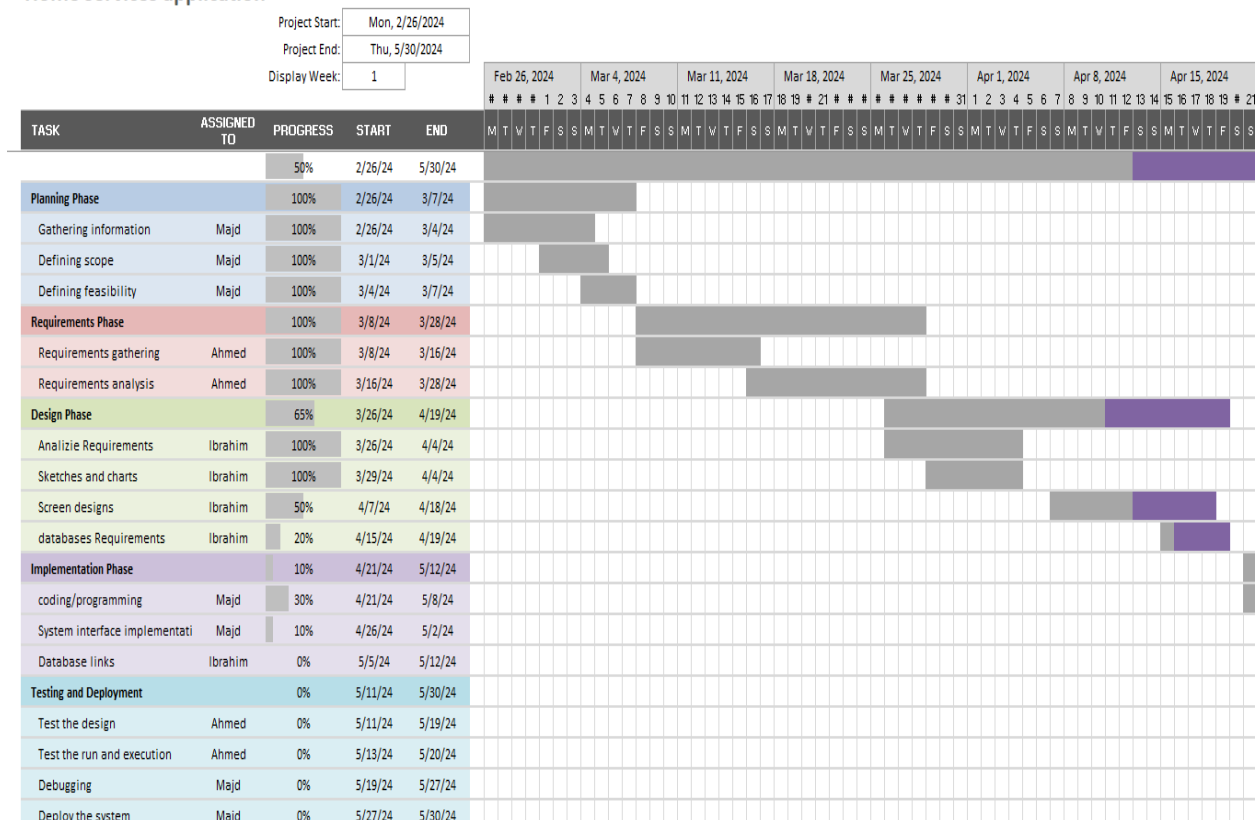


Figure (3-1): A Gantt chart that displays the phases of our project with respect to the timeline week(1-8)

Home services application

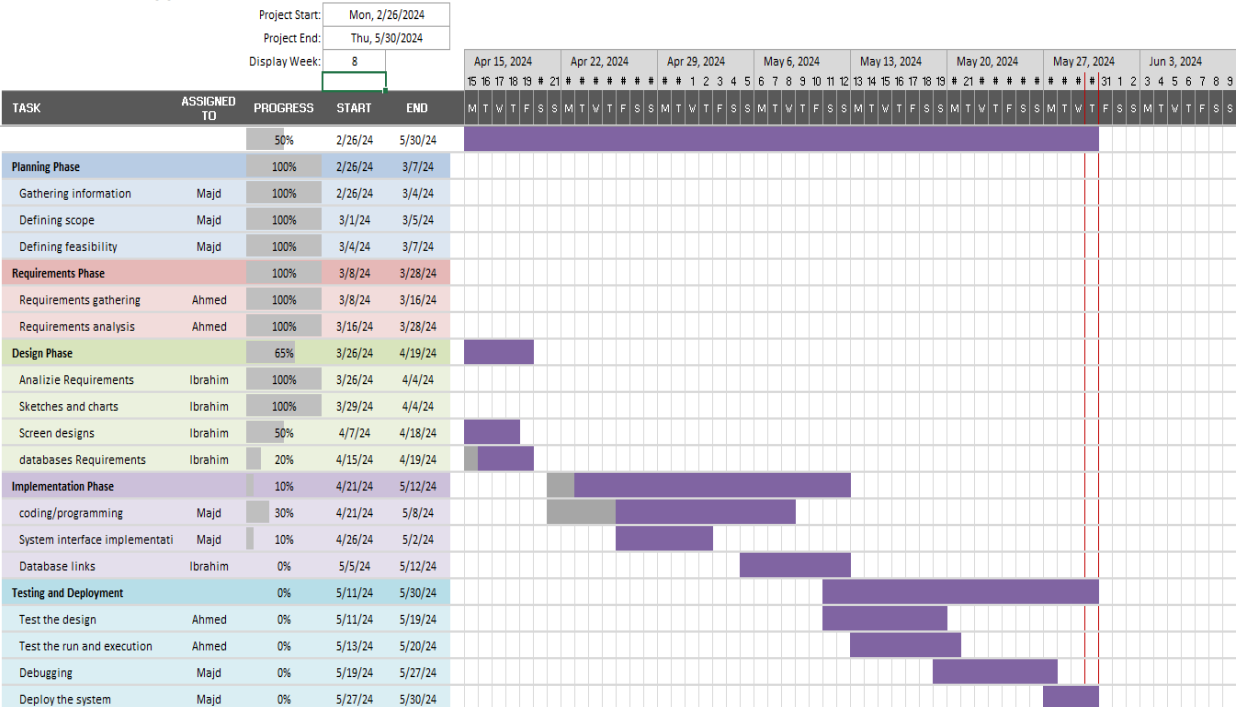


Figure (3-2): A Gantt chart that displays the phases of our project with respect to the timeline. week(8-14)

3.5 Development Tools:

Table (3-1): Tools and Languages Software Frameworks that will be used.

Language	Framework	Reason
HTML	Visual Studio Code	Front-end design for web applications.
CSS	Bootstrap	provides a collection of CSS components for creating responsive and mobile-first websites.
JavaScript	Bootstrap	easy to use and has a large community.
PHP	Symfony	Incredible Flexibility.
Python	Visual Studio Code	utilization of existing libraries.
SQL	Mango db using atlas	Its cloud based , it can handle my whole backend for me.

CHAPTER IV

PROJECT DESIGN and SPECIFICATION

4.1 Introduction

The success of any project hinges upon a well-structured design and meticulous specifications. Chapter 4 delves into the pivotal aspects of project design and specification, offering an intricate understanding of their significance in shaping the project's trajectory. This section furnishes a concise preview of the chapter's contents, elucidating why design and specification hold paramount importance in steering the project toward fruition.

4.2 System Specification

4.2.1 Functional requirements

It can be described as statements that outline the expected behaviour of a system. They specify what actions the system should perform to fulfil the user's requirements or expectations. These requirements are essentially the identifiable features or functionalities that the user can observe or interact with.

1-User Registration and Authentication:

- Users should be able to register with the application using email or social media accounts.
- Users should be able to log in securely with their credentials.

2-Search and Browse Services:

- Users should be able to search for service providers based on location, service category, ratings, and availability.
- Users should be able to view detailed profiles of service providers, including their services offered and availability.

3-Booking and Scheduling:

- Users should be able to book services from selected providers for specific dates and times.
- Service providers should receive notifications of new bookings and be able to accept or reject them.

4-Messaging and Communication:

- Users and service providers should be able to communicate directly through the application to discuss service details.
- Notifications should be sent to users and providers for important updates (e.g., booking confirmation, messages).

5-Review and Rating System:

- Users should be able to rate and leave reviews for service providers after completing a service.
- Service providers should have access to their ratings and reviews.
- Payment Processing:
- Users should be able to make secure payments through the application for booked services.
- Service providers should receive payments after services are rendered, minus any platform fees.

4.2.2 Non-Functional requirements

Non-functional requirements describe the characteristics or qualities that the system should have, such as performance, usability, security, and scalability. Examples of non-functional requirements:

1-Performance:

- The application should respond quickly to user interactions and searches, with minimal latency.
- The system should be able to handle a certain number of concurrent users without performance degradation.

2-Usability:

- The user interface should be intuitive and easy to navigate, catering to users of varying technical abilities.
- The application should be accessible on different devices (e.g., mobile phones, tablets, desktops).

3-Security:

- User data (e.g., personal information, payment details) should be securely stored and transmitted using encryption.
- The application should implement proper authentication and authorization mechanisms to prevent unauthorized access.

4-Reliability:

- The system should be available and reliable, minimizing downtime and disruptions to users.
- Data integrity and consistency should be maintained throughout the application.

5-Scalability:

- The application should be designed to handle increasing numbers of users and service providers as the platform grows.
- Scalability should be achieved through efficient database design, caching mechanisms, and scalable infrastructure.

4.3 Class Diagram

Creating a class diagram for our home services application can help visualize the structure of the system and the relationships between different classes or entities.

Starting with user class that can play role of different user type.

And the user searches for the service that he/she needs.

After the user finds the required service, he/she will reserve the Service Provider and select.

Then select the perfect day and time that the user wants.

The Service Providers in the same area as the User, will receive a notification that someone asking,

For the service he provides.

After booking the user chooses the pay method by cash or visa.

After booking the user chooses the pay method by cash or visa.

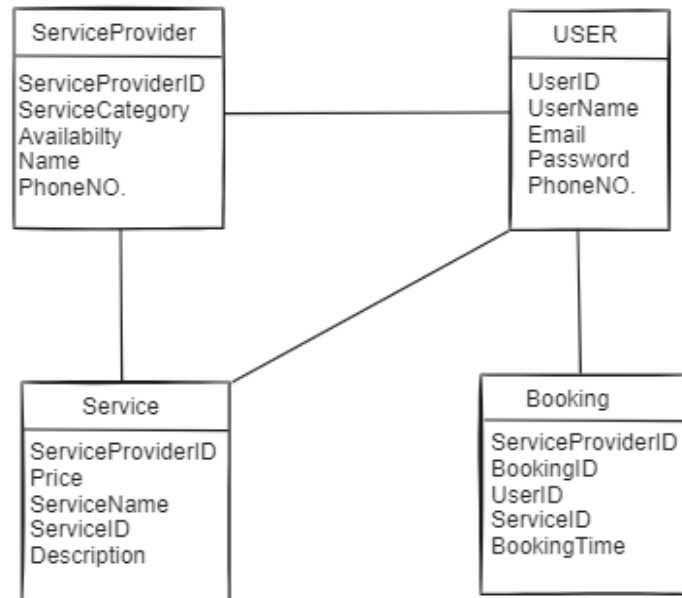


Figure (4-1): class diagram figure

4.4 Database Design

In this section, we will provide you with the database designs we are trying to implement. In our platform, users rely on databases to find and connect with service providers for their home-related needs. A well-designed database will serve as the central repository for all relevant information, ensuring that data is organized, accessible, and up-to-date. Whether it's a user searching for a plumber or a service provider managing their bookings, the database will play a pivotal role in facilitating seamless interactions and transactions. The database must also be able to handle increasing volumes of data and user traffic without compromising performance.

4.4.1 Entity-Relationship Diagram (ERD)

we utilized Entity-Relationship (ER) diagrams as a foundational tool for designing the database structure. ER diagrams provided a visual representation of how different components of our application's data interacted, ensuring efficient.

Designing an effective database schema was critical for our application's success. ER diagrams served as a blueprint for database design, guiding us in identifying entities, defining attributes, and establishing relationships.

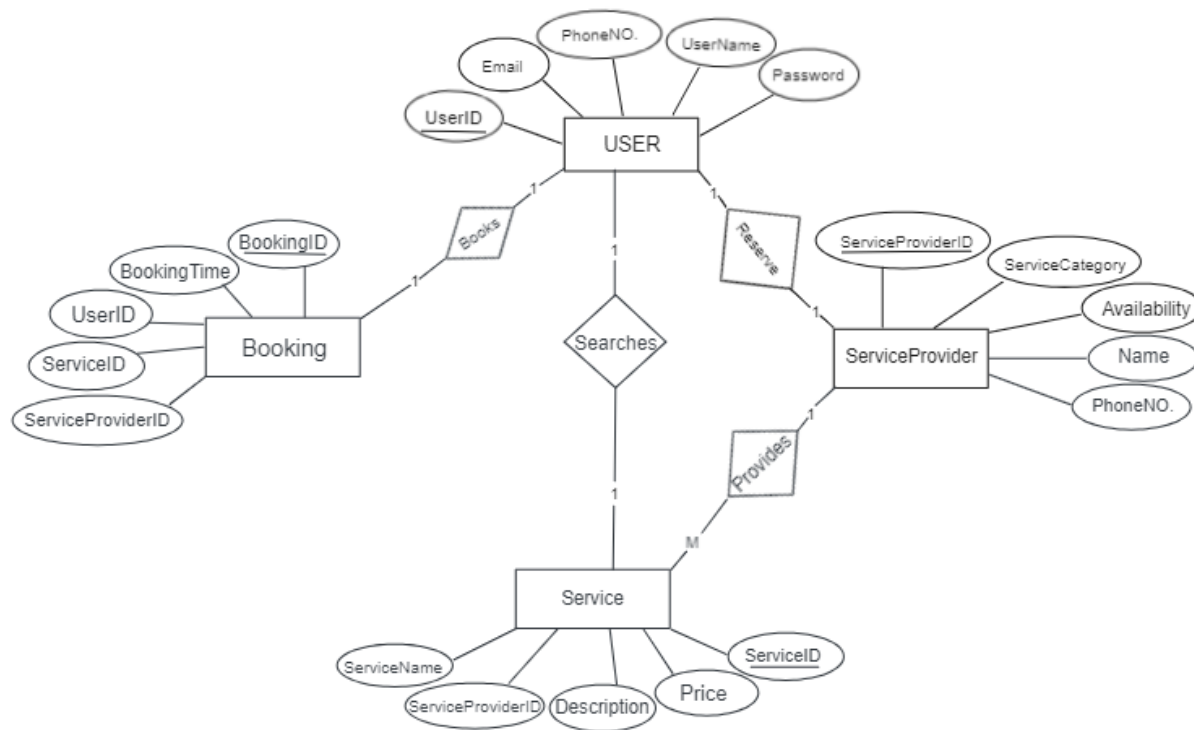


Figure (4-2): Entity-Relationship Diagram

4.4.2 Schema

In our home service application project for graduation, one of the key things we focused on was setting up a solid database schema. This basically means we worked on creating a well-organized structure for storing all the important info our app needed to handle.

the schema helped us avoid chaos by setting up rules and constraints for how data should be stored. It made sure that our data was tidy, consistent, and reliable, which is crucial for keeping our app running smoothly.

User

<u>UserID</u>	UserName	PhoneNO.	Email	Password
---------------	----------	----------	-------	----------

Service

<u>ServiceID</u>	ServiceName	Price	ServiceProviderID	Description
------------------	-------------	-------	-------------------	-------------

Booking

<u>BookingID</u>	BookingTime	UserID	ServiceProviderID	ServiceID
------------------	-------------	--------	-------------------	-----------

ServiceProvider

<u>ServiceProviderID</u>	ServiceCategory	Name	PhoneNO.	Availability
--------------------------	-----------------	------	----------	--------------

Figure (4-3): Schema Figure

4.5 User Interface

A well-designed user interface is essential for the success of our home services application. By focusing on responsiveness, usability, and visual appeal, we can create an engaging and efficient platform that meets the needs of both homeowners and service providers. Iterate on our design based on user feedback and continuously refine the UI to deliver a delightful experience for our users.

Designing a user interface (UI) for our home services application is crucial for providing a seamless and intuitive user experience. The UI should be user-friendly, visually appealing, and responsive across different devices.

1- User Authentication:

- Design user-friendly login and registration forms with options for social media login (e.g., Google, Facebook) to streamline the onboarding process.

2- Homepage:

- Feature a search bar, service categories, popular services, and a call-to-action button to encourage users to browse services.

3- User Profile:

- Enable users to view and edit their profiles, update contact information, manage bookings, and view transaction history.

4- Service Listings:

- Display service providers in a grid or list view, showcasing key information such as ratings, availability, and pricing.

5- Service Details:

- Provide detailed service descriptions, pricing options, availability calendars, and customer reviews on individual service pages.

6- Booking Interface:

- Include a booking form with date/time selectors, service options, and additional notes for users to request appointments.

7- Feedback and Reviews:

- Allow users to leave reviews and ratings for services they've used, with a section to display testimonials and recommendations.

CHAPTER 5

SYSTEM IMPLEMENTATION

Introduction

This chapter introduces the system implementation phase of the Home Service Application, providing an overview of the development process and detailing specific examples of how various parts of the system were built.

5.1 Implementation of the Home Service Application:

Database Integration:

- **Data Retrieval and Storage:** MongoDB is used as the database for storing user information, service details, and booking records. Mongoose, an ODM library for MongoDB, is employed to interact with the database. Various schemas are defined for different collections, such as Users, Services, and Bookings, ensuring data integrity and consistency.
- **Database Operations:** CRUD (Create, Read, Update, Delete) operations are implemented for managing data. For instance, when a user books a service, a new booking record is created in the database, and when a user updates their profile, the corresponding user record is modified.

User Interface Elements:

- **HTML and CSS:** The user interface is built using HTML5 for structure and CSS3 for styling. The design follows a modern, responsive approach to ensure usability across different devices.
- **JavaScript and Frontend Frameworks:** JavaScript, its used to handle dynamic elements on the frontend

Functionality Modules:

- **Service Booking:** The service booking functionality allows users to select a service, choose a preferred date and time, and provide additional details. The booking form collects this information and sends it to the server, where it is processed and stored in the database.
- **Payment Processing:** Users can choose between credit card and cash payments. For credit card payments, a secure form collects card details including card number, expiry date, and CVV.

CHAPTER 6

RESULT AND DISCUSSION

6.1 Introduction

This chapter presents the results and discussions related to the implementation of the Home Service Application. It provides context for the findings, highlighting the effectiveness and efficiency of the system based on the conducted tests and user feedback.

6.2 Data Collection and Research

During the implementation phase, data collection and research were critical in shaping the development process and ensuring the system met user needs and requirements. Various methods were employed:

- **User Surveys and Interviews:** Surveys and interviews were conducted with potential users to gather insights into their needs, preferences, and expectations for a home service application. This data was instrumental in designing features and functionalities.
- **Market Analysis:** Research was conducted on existing home service applications to identify best practices and common pitfalls. This analysis helped in benchmarking the system against industry standards and incorporating competitive features.

6.3 System Testing and User Feedback

Evaluating the implemented system involved comprehensive testing to ensure it meets the desired requirements and provides a seamless user experience.

Functional Testing:

- **Service Booking:** The functionality for booking various services, including selecting service types, dates, and payment methods, was rigorously tested

Usability Testing: User feedback was collected through usability testing sessions. Participants were asked to perform typical tasks, such as booking a service and making a payment, while their interactions with the system were observed. Feedback on the user interface, ease of navigation, and overall user experience was gathered and analyzed.

- Positive Feedback: Users appreciated the intuitive design, ease of use, and clear navigation. The booking process was found to be straightforward, and the payment options were convenient.

CHAPTER 7

RECOMMENDATIONS AND CONCLUSION

7.1 Introduction

This chapter provides a comprehensive wrap-up of the Home Service Application project. It includes a summary of key findings and outcomes, a discussion of the limitations encountered, and recommendations for future enhancements. The purpose of this chapter is to reflect on the project's success in meeting its objectives and to suggest directions for future work.

7.2 Conclusion

The Home Service Application project aimed to develop a user-friendly platform for booking various home services. The primary objectives were to provide seamless user authentication, integrate with a database for efficient data retrieval, design an intuitive user interface, and implement core functionalities such as service booking and payment processing.

Key findings from the project include:

- Database integration allowed for efficient storage and retrieval of user and booking data.
- The user interface received positive feedback for its intuitive design and ease of use.
- Core functionalities, including service booking and payment processing, were thoroughly tested and functioned as intended.

However, performance testing revealed that the system struggles under high traffic conditions, indicating a need for further optimization.

Overall, the project met its primary objectives, providing a functional and user-friendly application for home service bookings. The positive feedback from usability testing validates the effectiveness of the design and implementation.

7.3 Limitations and Recommendations for future work

During the project, several limitations were encountered:

- Performance Issues: The system showed decreased performance under heavy load, leading to slower response times.

- Service Descriptions: Users indicated a desire for more detailed service descriptions to better understand the offerings.

Based on these limitations, the following recommendations are proposed:

- Performance Optimization: Conduct a thorough review of the system architecture to identify bottlenecks. Consider optimizing the code and possibly upgrading server resources to handle higher loads more efficiently.
- Enhance Service Descriptions: Include more detailed descriptions for each service, possibly with images or videos, to give users a better understanding of what to expect.

APPENDICES

Questionnaires: we did a survey of when collocating the requirements needed for our project (you can find the file in the project folder) , the survey has helped us greatly improve our application for the better and made us focus on what the users deemed as important and directed us to the right path.