HalloDoc

BE-CE Semester- VIII

Prepared at

TatvaSoft House, Near Shivalik Business Center Sarkhej - Gandhinagar Highway, Rajpath Rangoli Rd, Ahmedabad, Gujarat 380054

Prepared By

Aakash Jitendrabhai Dave.

ID No. 200320107005

Guided By:

Prof. Stephy Patel

Department of CE

LJIET, Ahmedabad

External Guide:

Ms. Sweety Patel

Tatvasoft

Ahmedabad, Gujarat.

SUBMITTED TO

L.J. Institute of Engineering & Technology

Gujarat Technological University

L. J. Institute of Engineering & Technology

Ahmedabad

CERTIFICATE

This is to certify that the project report submitted along with the project entitled **HalloDoc** has been carried out by **Aakash Jitendrabhai Dave** under my guidance in partial fulfillment for the degree of Bachelor of Engineering in Computer Engineering, 8th Semester of Gujarat Technological University, Ahmadabad during the academic year 2023-24.

Prof. Stephy Patel — *Internal Guide*

Prof. Shruti Raval — *Head of the Department*

INTERNSHIP CERTIFICATE

L. J. Institute of Engineering & Technology
Ahmedabad

DECLARATION

We hereby declare that the Internship report submitted along with the Internship entitled **HalloDoc** submitted in partial fulfillment for the degree of Bachelor of Engineering in **Computer Engineering** to Gujarat Technological University, Ahmedabad, is a bonafide record of original project work carried out by me at Tatvasoft under the supervision of Ms. Sweety Patel and that no part of this report has been directly copied from any students' reports or taken from any other source, without providing due reference.

.

Name of Student **Signature of Student**

Aakash Dave

ACKNOWLEDGEMENT

I wish to express my sincere gratitude to my External guide Sweety Patel for continuously guiding me at the company and answering all my doubts with patience. I would also like to thank Prof. Shruti Raval (H.O.D. of IT Department) for motivating me every time whenever I get confused, I would also like to thank my Internal Guide Prof. Stephy Patel for helping me through my internship by giving me the necessary suggestions and advices along with their valuable co-ordination in completing this Internship.

I also thank my parents, friends and all the members of the family for their precious support and encouragement which they had provided in completion of my work. In addition to that, I would also like to mention the company personals who gave me the permission to use and experience the valuable resources required for the Internship.

Thus, in conclusion to the above said, I once again thank the staff members of TATVASOFT for their valuable support in completion of the Internship.

Name of Student **Date:**

AAKASH DAVE

200320107005 **Signature of Student**

TABLE OF CONTENTS

CHAPTERS	PAGE NO.
Title Page	**i**
Certificate Page	**ii**
Internship Certificate	**iii**
Declaration	**iv**
Acknowledgements	**v**
Table of Contents	**vi**
Abstract	**ix**
List of Figures	**x**
List of Tables	**xi**
CHAPTER 1 INTRODUCTION OF PROJECT & COMPANY PROFILE	**01**
1.1 Introduction	
1.1.1 Company Profile	
1.1.2 Company Products	
1.1.3Company Mission and Vision	
1.2 Introduction of the Project	
1.2.1 Purpose of the Project	
1.2.2 Function Requirements	
1.2.3 Problems in Existing System	
1.2.4 Main Modules	
CHAPTER 2 SYSTEM REQUIREMENTS	**05**
2.1 Hardware & Software Requirements	
2.1.1 Server-Side Requirements	
2.1.2 Developer Side Requirements	
2.1.3 User Side Requirements	
CHAPTER 3 WORK SHEET REPORT	**08**
3.1 WORK SHEET REPORT (15 DAYS)	
CHAPTER 4 FRONT END OF SYSTEM	**15**

4.1 About Front End	
4.1.1 about HTML	
4.1.2 about CSS	
4.1.3 about JavaScript	
4.2 Why Use PHP	
CHAPTER 5 BACK END OF SYSTEM	**18**
5.1 About PHP	
5.2 Why Use PHP	
5.3 About Back End	
5.3.1 about MySQL	
5.3.2 How MySQL Works	
5.3.3 MySQL Features	
5.4 Why use MySQL	
CHAPTER 6 SYSTEM DESIGN	**20**
6.1 Use Case Diagram	
6.2 Class Diagram	
CHAPTER 7 DATA DICTIONARY	**27**
7.1 **Introduction**	
7.2 **List of Tables**	
7.2.1: user_master	
7.2.2: bakery_info	
7.2.3: product_category_master	
7.2.4: product_master	
7.2.5: booking_master	
7.2. 6: delivery_boy	
7.2.7: social_media	
7.2.8: payment_method	
7.2.9: state_ master	

7.2.10: city_master	
7.2.11: area_master	
7.2.12: banner_master	
7.2.13: feedback_master	
7.2.14: contact_us_master	
7.2.15: offer_master	
CHAPTER 8 TESTING	**43**
8.1 Testing Plan	
8.2 Testing Strategies	
8.3 Testing Method	
8.4 Test Case	
CHAPTER 9 SNAPSHOT OF WEBSITE	**46**
9.1 Admin Site	
9.2 Users Site	
CHAPTER 10 ADVANTAGES	**53**
10.1 Advantages	
10.2 Limitations	
CHAPTER 11 CONCLUSION AND FUTURE ENHANCEMENT	**54**
11.1 Conclusion	
11.2 Future Enhancement	
CHAPTER 12 BIBLIOGRAPHY	**55**
12.1 Course Outcome	
12.2 Books	
12.2 Web Reference	

HALLODOC

Enrollment No.: 200320107005

Student Name: AAKASH DAVE

L. J. INSTITUTE OF ENGINEERING AND TECHNOLOGY
(College Code:132)

Semester: VIII, Computer Engineering Department

ABSTRACT

HalloDoc is a cutting-edge platform poised to revolutionize healthcare delivery by seamlessly integrating online doctor consultations and diagnostic services. With a user-friendly interface catering to both physicians and patients, HalloDoc streamlines medical processes, enhancing patient care and optimizing workflow for healthcare providers.

The platform offers a comprehensive suite of features, facilitating efficient access to patient records, appointment management, and secure communication channels between doctors and patients. Patients can conveniently request care for themselves or on behalf of others, fostering accessibility and convenience in healthcare delivery.

Admin functionalities empower comprehensive oversight, enabling the review of patient and physician records, case management, and the ability to manage requests effectively.

HalloDoc represents a significant advancement in healthcare technology, bridging the gap between healthcare providers and patients, thereby promoting accessibility, efficiency, and improved patient outcomes in the digital age.

LIST OF FIGURES

FIGURE NAME	PAGE NO.
Figure 6.2.1: Use case diagram	**36**

LIST OF TABLES

TABLE NAME	PAGE NO.
TABLE NAME 1(Table 1.1: Table name)	**10**
TABLE NAME 2	**20**

CHAPTER: 1

INTRODUCTION

1.1 Introduction of Project

1.2 Introduction of Company

1.2.1 Company Profile

1.2.2 Company Products

1.1 Introduction of Project

HalloDoc is a cutting-edge healthcare platform aimed at revolutionizing patient-doctor interactions. Our mission is to remove barriers to healthcare by offering virtual consultations and diagnostic services, eliminating the need for physical visits. Like effective marketing relies on understanding ideal customers, HalloDoc tailors healthcare solutions by considering individual needs and preferences. With a user-friendly interface, we empower patients to take control of their health conveniently. HalloDoc is committed to transforming healthcare delivery, ensuring accessible and personalized care anytime, anywhere.

Purpose

HalloDoc aims to break down barriers in healthcare by fostering trust and accessibility. Just as understanding the ideal customer is crucial in effective marketing, we believe in understanding patients deeply. Our platform facilitates virtual consultations and diagnostics, eliminating physical barriers to care. Through personalized interactions, we address individual needs, habits, and aspirations, ensuring tailored healthcare solutions. HalloDoc empowers patients to engage with their health, making informed decisions. By leveraging technology and understanding patients comprehensively, we strive to revolutionize healthcare delivery, providing accessible and trustworthy care to all.

Objective

HalloDoc's core goal is to enhance healthcare accessibility and trust, our platform empowers patients with virtual consultations and diagnostics. Through this innovative approach, HalloDoc aims to revolutionize healthcare, bridging the gap between patients and providers while ensuring accurate representation of patient needs.

Scope

The scope of the HalloDoc project includes facilitating virtual consultations and offering diagnostic services for remote monitoring and analysis. Efficient patient management tools will be developed, along with telemedicine features for remote healthcare delivery. A user-friendly interface and dedicated mobile application will enhance accessibility. Integration with existing healthcare systems will enable seamless data exchange, while features for patient feedback and ratings will promote transparency and accountability. Continuous improvement mechanisms will ensure HalloDoc remains innovative in healthcare technology.

1.2 Introduction of Company

1.2.1 Company Profile

Tatvasoft

TatvaSoft, founded in 2001 and headquartered in Ahmedabad, is a distinguished software product development company known for its expertise in designing and implementing mobile apps, advanced software solutions, and websites. Specializing in crafting comprehensive digital solutions, TatvaSoft assists businesses across various sectors in overcoming their challenges and achieving their goals effectively.

With a skilled team proficient in a multitude of technologies, TatvaSoft offers bespoke website development services tailored to the unique needs of each client. Whether it's creating dynamic e-commerce platforms, interactive corporate websites, or engaging web applications, TatvaSoft leverages its expertise to deliver high-quality, visually appealing, and user-friendly websites.

In addition to website development, TatvaSoft excels in mobile app development and software implementation. By conceptualizing and strategizing the entire development process, the company ensures that projects are executed with precision, reducing risks, managing costs, and delivering solutions that meet and exceed client expectations.

Through its commitment to excellence and innovation, TatvaSoft has earned a reputation as a trusted partner for businesses seeking to leverage technology to drive growth and success in today's digital landscape.

1.2.2 Company Products

- Custom Software Development
- Web Development
- Mobile App Development
- UI/UX Design
- Product Development & Maintenance
- Cloud Services
- Big Data & Analytics Solutions
- IT Services & IT Consulting

CHAPTER: 2
SYSTEM REQUIREMENTS

2.1 Hardware & Software Requirements

2.1.1 Server-Side Requirements

2.1.2 Developer Side Requirements

2.1.3 User Side Requirements

2.1.1 Server-Side Requirements

Web Server: For hosting the Halodoc web application, you can use Microsoft Internet Information Services (IIS) or deploy the application to a cloud platform like Azure App Service.

Application Framework:.NET provides a robust application framework for building web applications. You can use ASP.NET Core, which offers features for developing server-side logic, handling HTTP requests, and interacting with databases.

Database Management System (DBMS): PostgreSQL is the chosen DBMS. Ensure that you have PostgreSQL installed and configured to store and manage user data, medical records, and other information securely.

Programming Language: Server-side development is done using C# with ASP.NET Core, which is a popular choice for .NET web applications. C# integrates seamlessly with ASP.NET Core and provides strong typing, performance, and scalability.

Authentication and Authorization: Implement authentication and authorization using ASP.NET Core Identity, which provides features for managing user authentication, roles, and permissions.

2.1.2 Developer Side Requirements

Integrated Development Environment (IDE): Developers will primarily use Microsoft Visual Studio as the integrated development environment (IDE) for .NET development. Visual Studio offers a comprehensive set of tools for code editing, debugging, version control integration, and project management.

Database Management Tools: Developers will utilize database management tools compatible with PostgreSQL, such as pgAdmin or DBeaver, for tasks such as database schema design, querying, data manipulation, and debugging SQL queries directly within the IDE.

Version Control System: Git will be the version control system (VCS) of choice for managing source code. Developers can use Visual Studio's built-in Git integration or external platforms like GitHub, GitLab.

Dependency Management: .NET projects rely on NuGet as the package manager for managing dependencies. Visual Studio provides seamless integration with NuGet, allowing developers to easily install, update, and manage project dependencies directly from within the IDE.

2.1.3 User Side Requirements

Web and Mobile Accessibility: The Halodoc platform should be accessible via web browsers on desktops, laptops, tablets, and smartphones. Additionally, native mobile applications for iOS and Android devices should be available to provide a consistent user experience.

Cross-Browser Compatibility: The web version of Halodoc should be compatible with popular web browsers such as Google Chrome, Mozilla Firefox, Apple Safari, Microsoft Edge, and others. This ensures that users can access the platform regardless of their preferred browser.

Responsive Design: The user interface of Halodoc should be responsive and adapt to different screen sizes and resolutions. This ensures a consistent and optimal viewing experience across devices, whether users are accessing the platform from a desktop computer or a mobile device.

Ease of Navigation: The user interface should be intuitive and easy to navigate, allowing users to access various features and functionalities without confusion. Clear navigation menus, buttons, and links should guide users through the platform seamlessly.

User Authentication and Authorization: Users should be able to register, log in, and manage their accounts securely. The authentication process should be straightforward, and appropriate authorization mechanisms should be in place to control access to sensitive features and data.

Interactive Features: The platform should include interactive features such as forms, buttons, and menus to facilitate user interaction and engagement. Features like real-time chat, appointment scheduling, and medication reminders enhance the user experience and utility of the platform.

Multimedia Support: Users should be able to access multimedia content such as images, videos, and audio recordings within the platform. This may include educational materials, instructional videos, or visual aids to supplement healthcare information and consultations.

CHAPTER: 3

Daily Task

3.1 Work Sheet Report (15 DAYS)

3.1 Work Sheet Report (15 DAYS):

SUGGESTED 15 DAYS WORK SHEET REPORT	
Student Name:	AAKASH DAVE
Enrollment No:	200320107005
Internship/Project Title	Hallodoc
Tools and Technologies	Bootstrap , Dotnet and Postgres
Company/ Organization Name	Tatvasoft
Student's Activity Details:	

Week Number	**Start Date to End Date**	**Tasks to be assigned**	**Tasks to be completed**	**Remarks**
1	08-01-24 to 12-01-24	Introduction to SRS Watching videos on training portal about Github , coding standards, HTML, CSS,JS and Git. Assignment 1 Assignment 2 Assignment 3	Introduction to SRS Watching videos on training portal about Github , coding standards, HTML, CSS,JS and Git. Assignment 1 Assignment 2 Assignment 3	
2	16-01-24 to 19-01-24	Assignment 4 **Design** -hallodoc Patient platform login Forget Password Submit request Four request forms	Assignment 4 **Design** -hallodoc Patient platform login Forget Password Submit request Four request forms	

SUGGESTED 15 DAYS WORK SHEET REPORT

Student Name:	AAKASH DAVE
Enrollment No:	200320107005
Internship/Project Title	Hallodoc
Tools and Technologies	Bootstrap , Dotnet and Postgres
Company/ Organization Name	Tatvasoft

Student's Activity Details:

Week Number	Start Date to End Date	Tasks to be assigned	Tasks to be completed	Remarks
3	22-01-24 to 25-01-24	Assignment SQl -1 Assignment SQl -2 Assignment SQl -3 Assignment SQl -4	Assignment SQl -1 Assignment SQl -2 Assignment SQl -3 Assignment SQl -4	
4	29-01-24 to 02-02-24	Watching Postgres videos and understanding MVC architecture Create MVC UI of Patient platform login Forget Password Submit request Four request forms	Watching Postgres videos and understanding MVC architecture Create MVC UI of Patient platform login Forget Password Submit request Four request forms	

SUGGESTED 15 DAYS WORK SHEET REPORT

Student Name:	AAKASH DAVE
Enrollment No:	200320107005
Internship/Project Title	Hallodoc
Tools and Technologies	Bootstrap , Dotnet and Postgres
Company/ Organization Name	Tatvasoft

Student's Activity Details:

Week Number	Start Date to End Date	Tasks to be assigned	Tasks to be completed	Remarks
5	05-02-24 to 09-02-24	Create MVC UI of Four request forms Postgres integration of Forms Patient Dashboard designs	Create MVC UI of Four request forms Postgres integration of Forms Patient Dashboard designs	
6	12-02-24 to 16-02-24	Patient Dashboard integration View Documents- design and integration Review Agreement and Profile- design and integration	Patient Dashboard integration View Documents- design and integration Review Agreement and Profile- design and integration	

SUGGESTED 15 DAYS WORK SHEET REPORT

Student Name:	AAKASH DAVE
Enrollment No:	200320107005
Internship/Project Title	Hallodoc
Tools and Technologies	Bootstrap , Dotnet and Postgres
Company/ Organization Name	Tatvasoft

Student's Activity Details:

Week Number	Start Date to End Date	Tasks to be assigned	Tasks to be completed	Remarks
7	19-02-24 to 23-02-24	Admin Dashboard – UI and integration View Case UI and integration	Admin Dashboard – UI and integration View Case UI and integration	
8	26-02-24 to 01-03-24	View Notes , cancel case, block case and assign case – UI and integration	View Notes , cancel case, block case and assign case – UI and integration	

SUGGESTED 15 DAYS WORK SHEET REPORT

Student Name:	AAKASH DAVE
Enrollment No:	200320107005
Internship/Project Title	Hallodoc
Tools and Technologies	Bootstrap , Dotnet and Postgres
Company/ Organization Name	Tatvasoft

Student's Activity Details:

Week Number	Start Date to End Date	Tasks to be assigned	Tasks to be completed	Remarks
9	04-03-24 to 08-03-24	View uploads , orders and transfer case – UI and integration	View uploads , orders and transfer case – UI and integration	
10	11-03-24 to 15-03-24	Clear case, Send Agreement, Encounter Form– UI and integration	Clear case, Send Agreement, Encounter Form– UI and integration	

SUGGESTED 15 DAYS WORK SHEET REPORT

Student Name:	AAKASH DAVE
Enrollment No:	200320107005
Internship/Project Title	Hallodoc
Tools and Technologies	Bootstrap , Dotnet and Postgres
Company/ Organization Name	Tatvasoft

Student's Activity Details:

Week Number	Start Date to End Date	Tasks to be assigned	Tasks to be completed	Remarks

CHAPTER 4
FRONT END OF SYSTEM

4.1 About Front End

4.1.1 About Bootstrap
4.1.2 About AJAX and Jquery
4.1.3 About HTML
4.1.4 About CSS
4.1.5 About JavaScript

4.1.1 About Bootstrap

Bootstrap is a popular front-end framework for building responsive and mobile-first websites and web applications. Developed by Twitter, it offers pre-designed HTML, CSS, and JavaScript components for creating a consistent and visually appealing user interface.

One of Bootstrap's key features is its grid system, which allows developers to create responsive layouts easily. The grid system is based on a 12-column layout, which can be customized to suit different screen sizes, from large desktops to mobile devices. This flexibility ensures that websites built with Bootstrap look good on any device.

Bootstrap also provides a wide range of components, including buttons, forms, navigation bars, and carousels, which can be easily customized and integrated into projects. Additionally, it offers built-in support for responsive typography, ensuring that text scales appropriately across devices. Developers can take advantage of Bootstrap's extensive documentation and community support to quickly learn and implement its features. Furthermore, Bootstrap can be easily integrated with other front-end technologies and frameworks, such as jQuery and AngularJS, to enhance functionality

4.1.2 About AJAX and Jquery

AJAX (Asynchronous JavaScript and XML) is a technique used in web development to create dynamic and interactive web applications. It enables browsers to send and receive data asynchronously without requiring the entire page to reload. AJAX is typically implemented using JavaScript and XML (though JSON is more common nowadays), allowing for seamless communication between the client and server.

With AJAX, developers can update specific parts of a web page dynamically, enhancing user experience by providing faster and smoother interactions. Common use cases include form submission, live data updates, and auto-complete search features.

jQuery is a fast, small, and feature-rich JavaScript library that simplifies HTML document traversal and manipulation, event handling, animation, and AJAX interactions. It abstracts away many of the complexities of JavaScript, providing a simpler syntax for common tasks. jQuery's extensive library of plugins further extends its capabilities, allowing developers to add advanced functionality to their websites with minimal effort.

Combining AJAX with jQuery simplifies the process of making asynchronous requests and handling responses, as jQuery provides convenient methods for performing AJAX operations. Its AJAX functions allow developers to fetch data from a server, send data to a server, and handle server responses with ease**.**

4.1.3 About HTML

HTML, or HyperText Markup Language, is the foundation of web pages. It acts like a skeleton, providing structure and organization to the content displayed on a website. Think of it as a set of instructions that tells a web browser how to present text, images, videos, and other elements.
HTML uses tags, written within angle brackets (< and >), to define different parts of a web page. These tags create a hierarchy, allowing you to build headings, paragraphs, lists, links, and more. While HTML doesn't directly control the visual appearance, it lays the groundwork for applying styles using CSS. Overall, HTML is essential for creating the basic building blocks of any web page.

4.1.4 About CSS

CSS, or Cascading Style Sheets, is the language that controls the visual presentation of a web page. It complements HTML, which provides structure, by defining the look and feel of the content.
Imagine CSS like a fashion designer for your web page. Using CSS, you can style elements like text, buttons, and images with properties such as font size, color, background, and layout. You can create margins, padding, borders, and apply visual effects for a more engaging user experience. CSS allows you to define styles globally ortarget specific elements, ensuring a consistent and visually appealing website. In short, CSS breathes life into the structure of your web page, transforming it into a visually compelling experience.

4.1.5 About JavaScript

JavaScript is a versatile scripting language that adds interactivity and dynamism to web pages. Unlike HTML and CSS, which define structure and presentation, JavaScript brings web pages to life with dynamic behavior.
Imagine a website without JavaScript – it would be static and unresponsive. JavaScript allows you to create features like animations, form validation, interactive elements, and more. It can manipulate the content of a web page, respond to user actions (clicks, scrolls), and communicate with servers to retrieve or update data.
JavaScript code can be embedded directly in HTML pages or loaded from separate files. While it's primarily used for client-side scripting within web browsers, JavaScript can also be used for server-side scripting environments, expanding its reach beyond web development. In essence, JavaScript empowers web developers to create dynamic and engaging user experiences.

CHAPTER: 5
BACK END OF SYSTEM

5.1 About Back End
5.1.1 About.Net
5.1.2 About PostgresSQL Databases

5.1.1 About.Net
.NET is a versatile and powerful framework developed by Microsoft for building various types of applications, including web applications, desktop applications, mobile apps, and cloud-based services. It provides a robust, object-oriented programming environment and a rich set of libraries and tools for developers.

ASP.NET is a key component of the .NET framework, specifically designed for building dynamic web applications and services. It offers several frameworks, including ASP.NET Web Forms, ASP.NET MVC (Model-View-Controller), and ASP.NET Web API, each catering to different development needs.

ASP.NET Web Forms follows a traditional event-driven model, allowing developers to create web applications using familiar concepts such as server controls and view state. It abstracts away many complexities of web development, enabling rapid application development.

On the other hand, ASP.NET MVC provides a more modern and flexible approach to building web applications, separating concerns into models, views, and controllers. This architectural pattern promotes clean code, testability, and separation of concerns, making it well-suited for large-scale applications.

ASP.NET Web API is a framework for building RESTful APIs using .NET, enabling seamless integration with web and mobile applications. It simplifies the process of exposing data and services over HTTP, facilitating communication between different parts of an application or between different applications altogether.

Additionally, .NET Core, a cross-platform and open-source version of the .NET framework, further extends the capabilities of .NET, allowing developers to build and deploy applications on various platforms, including Windows, Linux, and macOS.

5.1.2 About PostgresSQL Databases

PostgreSQL, often referred to as "Postgres," is a powerful, open-source relational database management system (RDBMS) known for its reliability, robustness, and advanced features. It offers a wide range of capabilities for storing, managing, and querying structured data, making it suitable for a variety of applications, from small-scale projects to large enterprise systems.

One of PostgreSQL's key strengths is its adherence to SQL standards, providing a comprehensive set of SQL features along with support for advanced data types such as arrays, JSON, XML, and geometric data. This flexibility allows developers to model complex data structures and perform sophisticated queries efficiently.

PostgreSQL also boasts advanced transaction support, ensuring data integrity and consistency even in high-concurrency environments. It supports ACID (Atomicity, Consistency, Isolation, Durability) properties, making it suitable for applications that require strict transactional guarantees.

Another notable feature of PostgreSQL is its extensibility. Developers can extend PostgreSQL's functionality by creating custom data types, functions, and procedural languages using PL/pgSQL, PL/Python, PL/Perl, and other languages. Additionally, PostgreSQL supports a wide range of extensions developed by the community, providing additional features such as full-text search, geographic information systems (GIS), and more.

Furthermore, PostgreSQL offers built-in replication, high availability, and scalability features, allowing developers to design robust and scalable architectures for their applications. It supports various replication methods, including streaming replication, logical replication, and synchronous replication, enabling data distribution and fault tolerance.

CHAPTER: 6

SYSTEM DESIGN

6.1 Use Case Diagram

6.2 Class Diagram

6.1 Use Case Diagram

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships.

It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

- **Purpose of Use Case Diagrams**

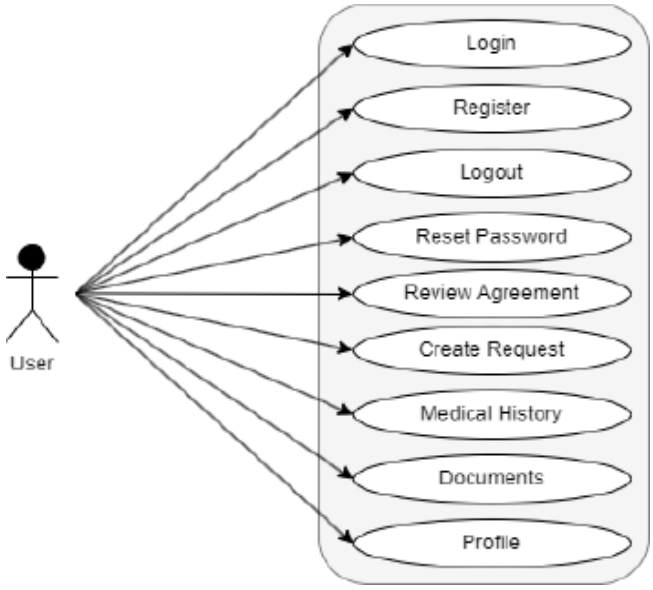
  The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirement, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams.

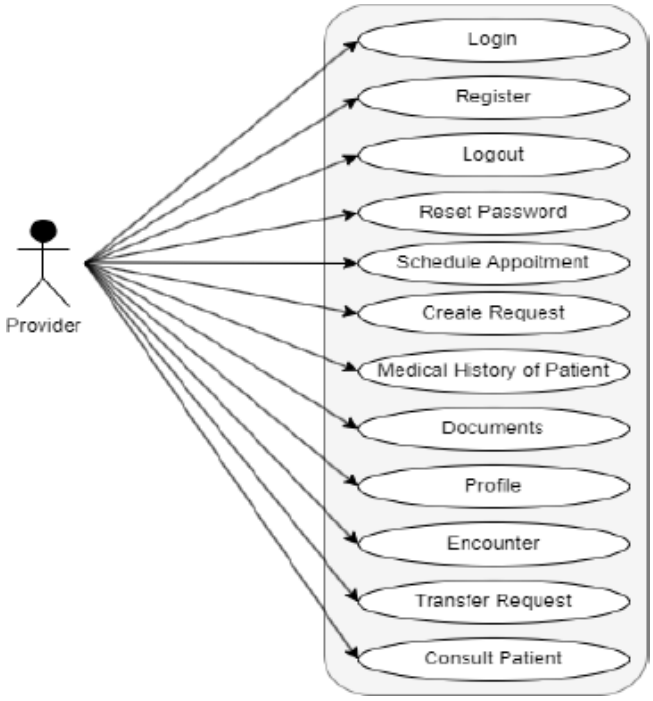
  - It gathers the system's needs.
  - It depicts the external view of the system.
  - It recognizes the internal as well as external factors that influence the system.
  - It represents the interaction between the actors.

Symbols used in Use case:

Symbol	Description
	Actor is an entity which interacts with the system. Actors carry out use case
	The use cases represent the behavior of the system. Typically various function are represented as use cases
	It identifies an interaction between actors and use cases. Each association represents a dialog.

6.1.1 Use case Diagram :-





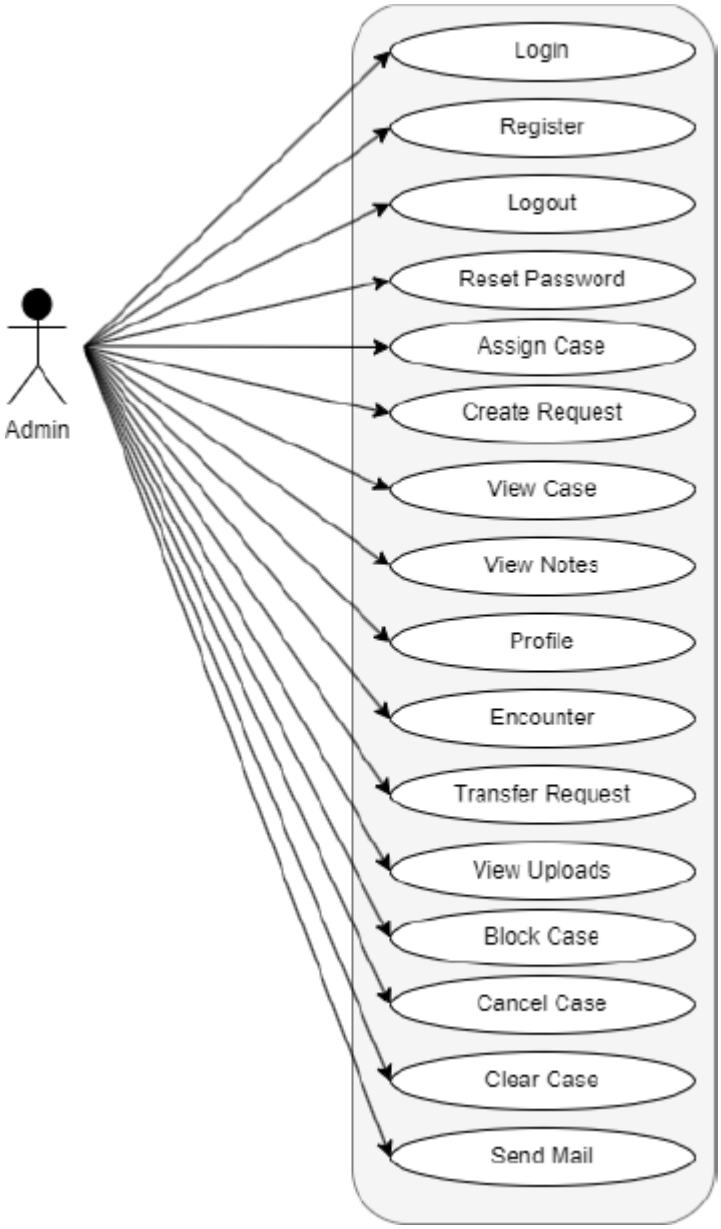


Figure 6.1.1 Use Case diagram

6.2 Data Flow Diagram

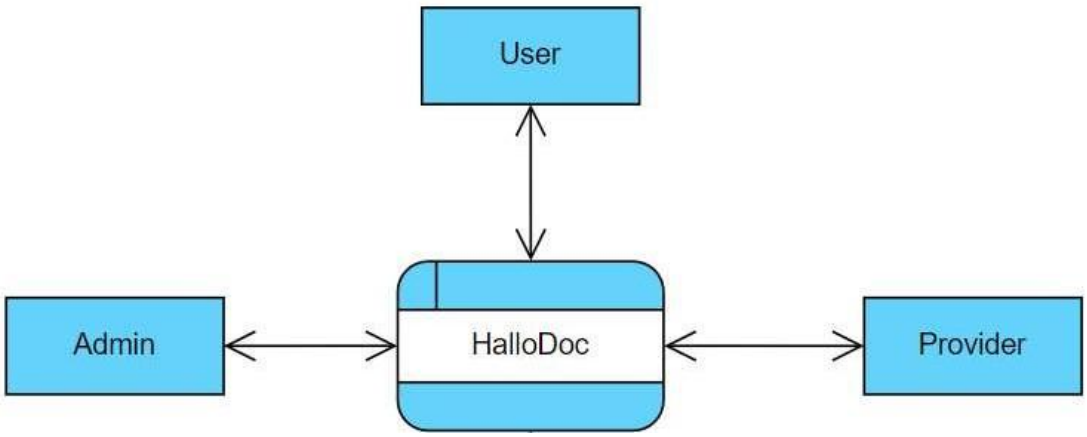
An activity diagram is a type of UML (Unified Modelling Language) diagram used to describe the flow of activities in a system or process. It shows the sequence of activities or tasks, their dependencies, and decision points in a graphical representation. Activity diagrams are commonly used in software development to model business processes, use cases, and workflows. They help to visualize the steps involved in a process and identify areas for improvement.
The purpose of an activity diagram is to visually represent the workflow or flow of control within a system. It focuses on the actions or activities performed by a system and how they relate to each other. Here's a breakdown of its key functionalities:

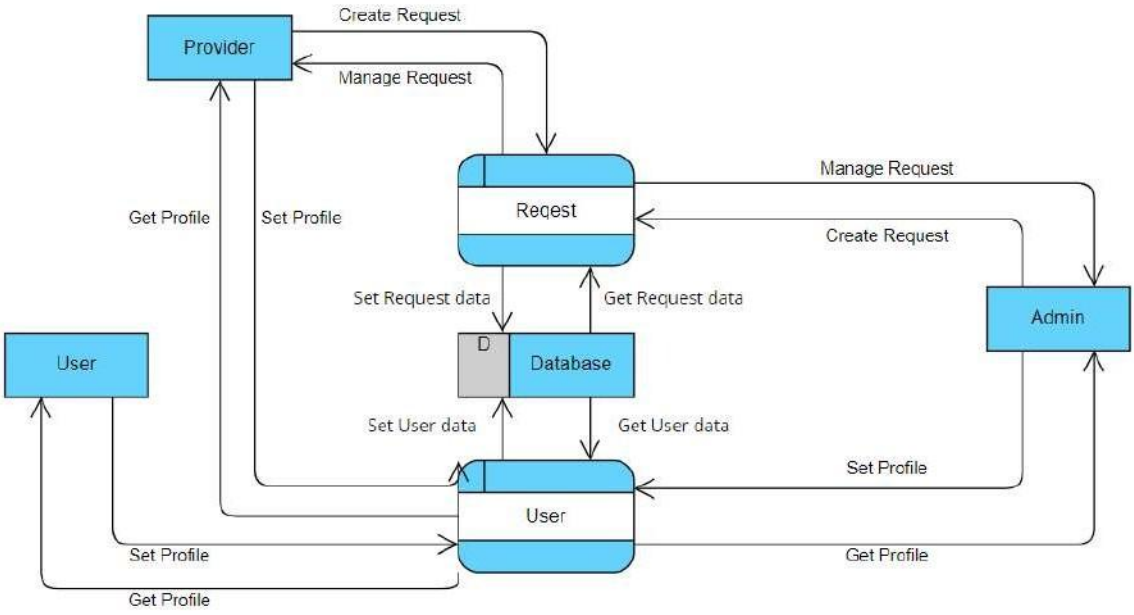
Modeling Dynamic Behavior: Activity diagrams depict the sequential or concurrent execution of activities within a system. This helps visualize the steps involved in a process, including decision points, branching paths, and alternative scenarios.

Understanding System Functionality: By portraying the flow of activities, activity diagrams provide a clear understanding of how a system works from start to finish. This is valuable for both developers and non-technical stakeholders.

Identifying System Interactions: They can illustrate interactions between different parts of a system, such as how a user interacts with the system or how different internal components work together.

Communication and Documentation: Activity diagrams serve as a communication tool to document system behavior and facilitate discussions between developers, designers, and other project members. They provide a shared understanding of the system's functionality.




Provider
Create Request
Manage Request
Get Profile
Set Profile
Reqest
Manage Request
Create Request
Admin
Set Request data
Get Request data
User
D
Database
Set User data
Get User data
Set Profile
User
Set Profile
Get Profile
Get Profile

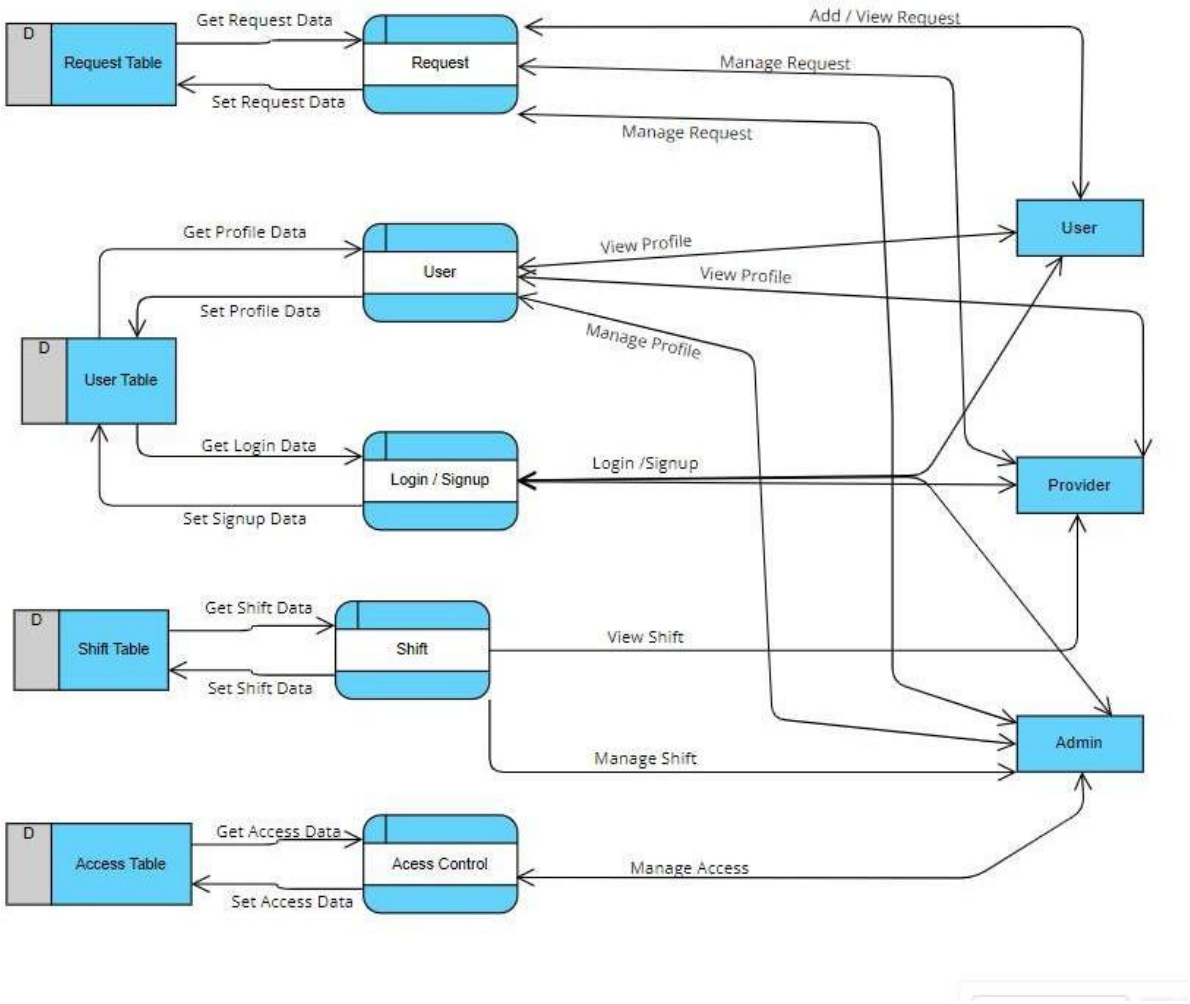


Figure 6.2.1 Data flow Diagrams

CHAPTER: 7

DATA DICTIONARY

7.1 Introduction

7.2 List of Tables

7.2.1 Admin

7.2.2 AdminRegion

7.2.3 AspNetRoles

7.2.4 AspNetUserRoles

7.2.5 AspNetUsers

7.2.6 BlockRequests

7.2.7 Business

7.2.8 CaseTag

7.2.9 Concierge

7.2.10 EmailLog

7.2.11 HealthProfessionals

7.2.12 OrderDetails

7.2.13 HealthProfessionalType

7.2.14 Menu

7.2.15 Physician

7.2.16 PhysicianLocation

7.2.17 PhysicianNotification

7.2.18 PhysicianRegion

7.2.19 Region

7.2.20 Request

7.2.21 RequestBusiness

7.2.22 RequestClient

7.2.23 RequestClosed

7.2.24 RequestConcierge

7.2.25 RequestNotes

7.2.26 RequestStatusLog

7.2.27 RequestType

7.2.28 RequestWiseFile

7.2.29 Role

7.2.30 RoleMenu

7.2.31 Shift

7.2.32 ShiftDetail

7.2.33 ShiftDetailRegion

7.2.34 SMSLog

7.2.35 User

7.1 Introduction

Data Dictionary is an important part of a project or system which contains all definition of elements in the system. In Data Dictionary you will find a list of all elements composing the data flowing through a system.

- The major elements of a system:
  - Data Flow
  - Data Store
  - Processes
- The data dictionary stores all details and description of these elements.
- The data dictionary provides additional information about the system.
- The data dictionary contains the data about data e.g. Student is a data and this student belongs to this course is a description of STUDENT data which is stored in data dictionary.
- Why is Data Dictionary important?
  - To manage the details in large system.
  - To communicate a common meaning for all system elements.
  - To document the features of system.
  - To determine where system changes should made.
  - To locate errors and omissions in the system.

7.2 List of Tables:

- **Table 7.2.1: Admin**

Table Name	Admin			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
AdminId	**int**		**No**	**Its Primary Key.**
AspNetUserId	**nvarchar**	**128**	**No**	**Its Foreign key of AspNetUsers Table.**
FirstName	**nvarchar**	**100**	**No**	
LastName	**nvarchar**	**100**	**Yes**	
Email	**nvarchar**	**50**	**No**	
Mobile	**nvarchar**	**20**	**Yes**	
Address1	**nvarchar**	**500**	**Yes**	
Address2		**500**	**Yes**	
City	**nvarchar**	**100**	**Yes**	
RegionId	**int**		**Yes**	

Zip	**nvarchar**	**10**	**Yes**	
AltPhone	**nvarchar**	**20**	**Yes**	
CreatedBy	**nvarchar**	**128**	**No**	
CreatedDate	**datetime**		**No**	
ModifiedBy	**nvarchar**	**128**	**Yes**	**Its Foreign key of AspNetUsers Table.**
ModifiedDate	**datetime**		**Yes**	
Status	**tinyint**		**Yes**	
IsDeleted	**bit**		**Yes**	
RoleId	**int**		**Yes**	

- **Table 7.2.2: AdminRegion**

Table Name	**AdminRegion**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
AdminRegionId	**int**		**No**	**Its Primary Key.**
AdminId	**int**		**No**	**Its Foreign key of Admin Table.**
RegionId	**int**		**No**	**Its Foreign key of Region Table.**

- **Table 7.2.3: AspNetRoles**

Table Name	**AspNetRoles**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
Id	**nvarchar**	**128**	**No**	**Its Primary Key.**
Name	**nvarchar**	**256**	**No**	

- **Table 7.2.4: AspNetUserRoles**

Table Name	**AspNetUserRoles**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
UserId	**nvarchar**	**128**	**No**	**Its Primary Key.Its Foreign key of AspNetUsers Table.**
RoleId	**nvarchar**	**128**	**No**	**Its Primary Key.**

- **Table 7.2.5: AspNetUsers**

Table Name	**AspNetUsers**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
Id	**nvarchar**	**128**	**No**	**Its Primary Key.**
UserName	**nvarchar**	**256**	**No**	

PasswordHash	**nvarchar**	**MAX**	**Yes**	
Email	**nvarchar**	**256**	**Yes**	
PhoneNumber	**nvarchar**	**20**	**Yes**	
IP	**nvarchar**	**20**	**Yes**	
CreatedDate	**datetime**		**No**	
CreatedDate	**datetime**		**Yes**	

- **Table 7.2.6: BlockRequests**

Table Name	**BlockRequests**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
BlockRequestId	**int**		**No**	**Its Primary Key.**
PhoneNumber	**nvarchar**	**50**	**Yes**	
Email	**nvarchar**	**50**	**Yes**	
IsActive	**bit**		**Yes**	
Reason	**nvarchar**	**MAX**	**Yes**	
RequestId	**nvarchar**	**50**	**No**	
IP	**nvarchar**	**20**	**Yes**	
CreatedDate	**datetime**		**Yes**	
ModifiedDate	**datetime**		**Yes**	

- **Table 7.2.7: Business**

Table Name	**Business**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
BusinessId	**int**	**100**	**No**	**Its Primary Key.**
Name	**nvarchar**		**No**	
Address1	**nvarchar**	**500**	**Yes**	
Address2	**nvarchar**	**500**	**Yes**	
City	**nvarchar**	**50**	**Yes**	
RegionId	**int**		**Yes**	**Its Foreign key of Region Table.**
ZipCode	**nvarchar**	**10**	**Yes**	
PhoneNumber	**nvarchar**	**20**	**Yes**	
FaxNumber	**nvarchar**	**20**	**Yes**	
IsRegistered	**bit**		**Yes**	
CreatedBy	**nvarchar**	**128**	**Yes**	**Its Foreign key of AspNetUsers Table.**
CreatedDate	**datetime**		**No**	
ModifiedBy	**nvarchar**	**128**	**Yes**	**Its Foreign key of AspNetUsers Table.**
ModifiedDate	**datetime**		**Yes**	
Status	**tinyint**		**Yes**	
IsDeleted	**bit**		**Yes**	

IP	**nvarchar**	**20**	**Yes**	

- **Table 7.2.8: CaseTag**

Table Name	**CaseTag**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
CaseTagId	**int**		**No**	
Name	**nvarchar**	**50**	**No**	

- **Table 7.2.8: CaseTag**

Table Name	**Concierge**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
ConciergeId	**int**		**No**	**Its Primary Key.**
ConciergeName	**nvarchar**	**100**	**No**	
Address	**nvarchar**	**150**	**Yes**	
Street	**nvarchar**	**50**	**No**	
City	**nvarchar**	**50**	**No**	
State	**nvarchar**	**50**	**No**	
ZipCode	**nvarchar**	**50**	**No**	
CreatedDate	**datetime**		**No**	
RegionId	**int**		**No**	**Its Foreign key of Region Table.**
RoleId	**nvarchar**	**20**	**Yes**	

- **Table 7.2.10: EmailLog**

Table Name	**EmailLog**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
EmailLogID	**decimal**	**9**	**No**	**Its Primary Key.**
EmailTemplate	**nvarchar**	**MAX**	**No**	
SubjectName	**nvarchar**	**200**	**No**	
EmailID	**nvarchar**	**200**	**No**	
ConfirmationNumber	**nvarchar**	**200**	**Yes**	
FilePath	**nvarchar**	**MAX**	**Yes**	
RoleId	**int**		**Yes**	
RequestId	**int**		**Yes**	
AdminId	**int**		**Yes**	
PhysicianId	**int**		**Yes**	
CreateDate	**datetime**		**No**	
SentDate	**datetime**		**Yes**	

IsEmailSent	**bit**		**Yes**	
SentTries	**int**		**Yes**	
Action	**int**		**Yes**	

- **Table 7.2.11: HealthProfessionals**

Table Name	**HealthProfessionals**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
VendorId	**int**		**No**	**Its Primary Key.**
VendorName	**nvarchar**	**100**	**No**	
Profession	**int**		**Yes**	**Its Foreign key of HealthProfessionalType Table.**
FaxNumber	**nvarchar**	**50**	**No**	
Address	**nvarchar**	**150**	**Yes**	
City	**nvarchar**	**100**	**Yes**	
State	**nvarchar**	**50**	**Yes**	
Zip	**nvarchar**	**50**	**Yes**	
RegionId	**int**		**Yes**	
CreatedDate	**datetime**		**No**	
ModifiedDate	**datetime**		**Yes**	
PhoneNumber	**nvarchar**	**100**	**Yes**	
IsDeleted	**bit**		**Yes**	
IP	**nvarchar**	**20**	**Yes**	
Email	**nvarchar**	**50**	**Yes**	
BusinessContact	**nvarchar**	**100**	**Yes**	

- **Table 7.2.12: OrderDetails**

Table Name	**OrderDetails**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
Id	**int**		**No**	**Its Primary Key.**
VendorId	**int**		**Yes**	
RequestId	**int**		**Yes**	
FaxNumber	**nvarchar**	50	**Yes**	
Email	**nvarchar**	50	**Yes**	
BusinessContact	**nvarchar**	100	**Yes**	
Prescription	**nvarchar**	max	**Yes**	
NoOfRefill	**int**		**Yes**	
CreatedDate	**datetime**		**Yes**	
CreatedBy	**nvarchar**	100	**Yes**	

- **Table 7.2.13: HealthProfessionalType**

Table Name	HealthProfessionalType			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
HealthProfessionalId	**int**		**No**	**Its Primary Key.**
ProfessionName	**nvarchar**	**50**	**No**	
CreatedDate	**datetime**		**No**	
IsActive	**bit**		**Yes**	
IsDeleted	**bit**		**Yes**	

- **Table 7.2.14: Menu**

Table Name	Menu			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
MenuId	**int**		**No**	**Its Primary Key.**
Name	**nvarchar**	**50**	**No**	
AccountType	**tinyint**		**No**	**1 - Admin, 2 - Physician**
SortOrder	**int**		**Yes**	

- **Table 7.2.15: Physician**

Table Name	Physician			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
PhysicianId	**int**		**No**	**Its Primary Key.**
AspNetUserId	**nvarchar**	**128**	**Yes**	**Its Foreign key of AspNetUsers Table.**
FirstName	**nvarchar**	**100**	**No**	
LastName	**nvarchar**	**100**	**Yes**	
Email	**nvarchar**	**50**	**No**	
Mobile	**nvarchar**	**20**	**Yes**	
MedicalLicense	**nvarchar**	**500**	**Yes**	
Photo	**nvarchar**	**100**	**Yes**	
AdminNotes	**nvarchar**	**500**	**Yes**	
IsAgreementDoc	**bit**		**Yes**	
IsBackgroundDoc	**bit**		**Yes**	
IsTrainingDoc	**bit**		**Yes**	
IsNonDisclosureDoc	**bit**		**Yes**	
Address1	**nvarchar**	500	**Yes**	
Address2	**nvarchar**	500	**Yes**	
City	**nvarchar**	100	**Yes**	
RegionId	**int**		**Yes**	
Zip	**nvarchar**	10	**Yes**	

AltPhone	**nvarchar**	20	**Yes**	
CreatedBy	**nvarchar**	128	**No**	Its Foreign key of AspNetUsers Table.
CreatedDate	**datetime**		**No**	
ModifiedBy	**nvarchar**	128	**Yes**	Its Foreign key of AspNetUsers Table.
ModifiedDate	**datetime**		**Yes**	
Status	**tinyint**		**Yes**	
BusinessName	**nvarchar**	100	**No**	
BusinessWebsite	**nvarchar**	200	**No**	
IsDeleted	**bit**		**Yes**	
RoleId	**int**		**Yes**	
NPINumber	**nvarchar**	500	**Yes**	
IsLicenseDoc	**bit**		**Yes**	
Signature	**nvarchar**	100	**Yes**	
IsCredentialDoc	**bit**		**Yes**	
IsTokenGenerate	**bit**		**Yes**	
SyncEmailAddress	**nvarchar**	50	**Yes**	

- **Table 7.2.16: PhysicianLocation**

Table Name	**PhysicianLocation**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
LocationId	**int**		**No**	
PhysicianId	**int**		**No**	**Its Foreign key of Physician Table.**
Latitude	**decimal**	**9**	**Yes**	
Longitude	**decimal**	**9**	**Yes**	
CreatedDate	**datetime**		**Yes**	
PhysicianName	**nvarchar**	**50**	**Yes**	
Address	**nvarchar**	**500**	**Yes**	

- **Table 7.2.17: PhysicianNotification**

Table Name	**PhysicianNotification**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
id	**int**		**No**	**Its Primary Key.**
PhysicianId	**int**		**No**	**Its Foreign key of Physician Table.**
IsNotificationStopped	**bit**		**No**	

- **Table 7.2.18: PhysicianRegion**

Table Name	PhysicianRegion			
Field Name	Data Type	Length	Nullable	Comments
PhysicianRegionId	int		No	Its Primary Key.
PhysicianId	int		No	Its Foreign key of Physician Table.
RegionId	int		No	Its Foreign key of Region Table.

- **Table 7.2.19: Region**

Table Name	Region			
Field Name	Data Type	Length	Nullable	Comments
RegionId	int		No	Its Primary Key.
Name	nvarchar	50	No	
Abbreviation	nvarchar	50	Yes	

- **Table 7.2.20: Request**

Table Name	Request			
Field Name	Data Type	Length	Nullable	Comments
RequestId	int		No	Its Primary Key.
RequestTypeId	int		No	1- Business, 2- Patient, 3- Family & 4- Concierge
UserId	int		Yes	Its Foreign key of User Table.
FirstName	nvarchar	100	Yes	
LastName	nvarchar	100	Yes	
PhoneNumber	nvarchar	23	Yes	
Email	nvarchar	50	Yes	
Status	tinyint		No	1-Unassigned,2-Accepted,3-Cancelled,4-Reserving,5-MDEnRoute,6-MDOnSite,7-FollowUp,8-Closed,9-Locked,10-Declined,11-Consult,12-Clear,13-CancelledByProvider,14-CCUploadedByClient,15-CCApprovedByAdmin
PhysicianId	int		Yes	Its Foreign key of Physician Table.
ConfirmationNumber	nvarchar	20	Yes	
CreatedDate	datetime		No	
IsDeleted	bit		Yes	
ModifiedDate	datetime		Yes	
DeclinedBy	varchar	250	Yes	
IsUrgentEmailSent	bit		No	
LastWellnessDate	datetime		Yes	
IsMobile	bit		Yes	
CallType	tinyint		Yes	

CompletedByPhysician	bit		Yes	
LastReservationDate	datetime		Yes	
AcceptedDate	datetime		Yes	
RelationName	nvarchar	100	Yes	
CaseNumber	nvarchar	50	Yes	
IP	nvarchar	20	Yes	
CaseTag	nvarchar	50	Yes	
CaseTagPhysician	nvarchar	50	Yes	
PatientAccountId	nvarchar	128	Yes	
CreatedUserId	int		Yes	

- **Table 7.2.21: RequestBusiness**

Table Name	RequestBusiness			
Field Name	Data Type	Length	Nullable	Comments
RequestBusinessId	int		No	Its Primary Key.
RequestId	int		No	Its Foreign key of Request Table.
BusinessId	int		No	Its Foreign key of Business Table.
IP	nvarchar	20	Yes	

- **Table 7.2.22: RequestClient**

Table Name	RequestClient			
Field Name	Data Type	Length	Nullable	Comments
RequestClientId	int		No	Its Primary Key.
RequestId	int		No	Its Foreign key of Request Table.
FirstName	nvarchar	100	No	
LastName	nvarchar	100	Yes	
PhoneNumber	nvarchar	23	Yes	
Location	nvarchar	100	Yes	
Address	nvarchar	500	Yes	
RegionId	int		Yes	Its Foreign key of Region Table.
NotiMobile	nvarchar	20	Yes	
NotiEmail	nvarchar	50	Yes	
Notes	nvarchar	500	Yes	
Email	nvarchar	50	Yes	
strMonth	nvarchar	20	Yes	
intYear	int		Yes	
intDate	int		Yes	
IsMobile	bit		Yes	
Street	nvarchar	100	Yes	

City	**nvarchar**	**100**	**Yes**	
State	**nvarchar**	**100**	**Yes**	
ZipCode	**nvarchar**	**10**	**Yes**	
CommunicationType	**tinyint**		**Yes**	
RemindReservationCount	**tinyint**		**Yes**	
RemindHouseCallCount	**tinyint**		**Yes**	
IsSetFollowupSent	**tinyint**		**Yes**	
IP	**nvarchar**	**20**	**Yes**	
IsReservationReminderSen	**tinyint**		**Yes**	
Latitude	**decimal**	9	**Yes**	
Longitude	**decimal**	9	**Yes**	

- **Table 7.2.23: RequestClosed**

Table Name	**RequestClosed**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
RequestClosedId	**int**		**No**	**Its Primary Key.**
RequestId	**int**		**No**	**Its Foreign key of Request Table.**
RequestStatusLogId	**int**		**No**	**Its Foreign key of RequestStatusLog Table.**
PhyNotes	**nvarchar**	**500**	**Yes**	
ClientNotes	**nvarchar**	**500**	**Yes**	
IP	**nvarchar**	**20**	**Yes**	

- **Table 7.2.24: RequestConcierge**

Table Name	**RequestConcierge**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
Id	**int**		**No**	**Its Primary Key.**
RequestId	**int**		**No**	**Its Foreign key of Request Table.**
ConciergeId	**int**		**No**	**Its Foreign key of Concierge Table.**
IP	**nvarchar**	**20**	**Yes**	

- **Table 7.2.25: RequestNotes**

Table Name	**RequestNotes**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
RequestNotesId	**Int**		**No**	**Its Primary Key.**
RequestId	**Int**		**No**	**Its Foreign key of Request Table.**
strMonth	**Nvarchar**	**20**	**Yes**	
intYear	**Int**		**Yes**	

intDate	**Int**		**Yes**	
PhysicianNotes	**Nvarchar**	**500**	**Yes**	
AdminNotes	**Nvarchar**	**500**	**Yes**	
CreatedBy	**Nvarchar**	**128**	**No**	
CreatedDate	**Datetime**		**No**	
ModifiedBy	**Nvarchar**	**128**	**Yes**	
ModifiedDate	**Datetime**		**Yes**	
IP	**Nvarchar**	**20**	**Yes**	
AdministrativeNotes	**Nvarchar**	**500**	**Yes**	

- **Table 7.2.26: RequestStatusLog**

Table Name	**RequestStatusLog**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
RequestStatusLogId	**int**		**No**	**Its Primary Key.**
RequestId	**int**		**No**	**Its Foreign key of Request Table.**
Status	**tinyint**		**No**	
PhysicianId	**int**		**Yes**	**Its Foreign key of Physician Table.**
AdminId	**int**		**Yes**	**Its Foreign key of Admin Table.**
TransToPhysicianId	**int**		**Yes**	**Its Foreign key of Physician Table.**
Notes	**nvarchar**	**500**	**Yes**	
CreatedDate	**datetime**		**No**	
IP	**nvarchar**	**20**	**Yes**	
TransToAdmin	**bit**		**Yes**	

- **Table 7.2.27: RequestType**

Table Name	**RequestType**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
RequestTypeId	**int**		**No**	**Its Primary Key.**
Name	**nvarchar**	**50**	**No**	

- **Table 7.2.28: RequestWiseFile**

Table Name	**RequestWiseFile**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
RequestWiseFileID	**int**		**No**	**Its Primary Key.**
RequestId	**int**		**No**	**Its Foreign key of Request Table.**
FileName	**nvarchar**	**500**	**No**	
CreatedDate	**datetime**		**No**	

PhysicianId	**int**		**Yes**	**Its Foreign key of Physician Table.**
AdminId	**int**		**Yes**	**Its Foreign key of Admin Table.**
DocType	**tinyint**		**Yes**	**1-TestOne,2-MedicalReport,3-CostReceipt**
IsFrontSide	**bit**		**Yes**	
IsCompensation	**bit**		**Yes**	
IP	**nvarchar**	**20**	**Yes**	
IsFinalize	**bit**		**Yes**	
IsDeleted	**bit**		**Yes**	
IsPatientRecords	**bit**		**Yes**	

- **Table 7.2.29: Role**

Table Name	**Role**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
RoleId	**int**		**No**	**Its Primary Key.**
Name	**nvarchar**	**50**	**No**	
AccountType	**tinyint**		**No**	1 - Admin, 2 - Physician
CreatedBy	**nvarchar**	**128**	**No**	
CreatedDate	**datetime**		**No**	
ModifiedBy	**nvarchar**	**128**	**Yes**	
ModifiedDate	**datetime**		**Yes**	
IsDeleted	**bit**		**No**	
IP	**nvarchar**	**20**	**Yes**	

- **Table 7.2.30: RoleMenu**

Table Name	**RoleMenu**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
RoleMenuId	**int**		**No**	**Its Primary Key.**
RoleId	**int**		**No**	**Its Foreign key of Role Table.**
MenuId	**int**		**No**	**Its Foreign key of Menu Table.**

- **Table 7.2.31: Shift**

Table Name	**Shift**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
ShiftId	**int**		**No**	**Its Primary Key.**
PhysicianId	**int**		**No**	**Its Foreign key of Physician Table.**
StartDate	**date**		**No**	
IsRepeat	**bit**		**No**	

WeekDays	**char**	**7**	**Yes**	
RepeatUpto	**int**		**Yes**	
CreatedBy	**nvarchar**	**128**	**No**	**Its Foreign key of AspNetUsers Table.**
CreatedDate	**datetime**		**No**	
IP	**nvarchar**	**20**	**Yes**	

- **Table 7.2.32: ShiftDetail**

Table Name	**ShiftDetail**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
ShiftDetailId	**int**		**No**	**Its Primary Key.**
ShiftId	**int**		**No**	**Its Foreign key of Shift Table.**
ShiftDate	**datetime**		**No**	
RegionId	**int**		**Yes**	
StartTime	**time**		**No**	
EndTime	**time**		**No**	
Status	**tinyint**		**No**	
IsDeleted	**bit**		**No**	
ModifiedBy	**nvarchar**	**128**	**Yes**	**Its Foreign key of AspNetUsers Table.**
ModifiedDate	**datetime**		**Yes**	
LastRunningDate	**datetime**		**Yes**	
EventId	**nvarchar**	**100**	**Yes**	

- **Table 7.2.33: ShiftDetailRegion**

Table Name	**ShiftDetailRegion**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
ShiftDetailRegionId	**int**		**No**	**Its Primary Key.**
ShiftDetailId	**int**		**No**	**Its Foreign key of ShiftDetail Table.**
RegionId	**int**		**No**	**Its Foreign key of Region Table.**
IsDeleted	**bit**		**Yes**	

- **Table 7.2.34: SMSLog**

Table Name	**SMSLog**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
SMSLogID	**decimal**	**9**	**No**	**Its Primary Key.**
SMSTemplate	**nvarchar**	**MAX**	**No**	
MobileNumber	**nvarchar**	**50**	**No**	
ConfirmationNumber	**nvarchar**	**200**	**Yes**	

RoleId	**int**		**Yes**	
AdminId	**int**		**Yes**	
RequestId	**int**		**Yes**	
PhysicianId	**int**		**Yes**	
CreateDate	**datetime**		**No**	
SentDate	**datetime**		**Yes**	
IsSMSSent	**bit**		**Yes**	
SentTries	**int**		**No**	
Action	**int**		**Yes**	

- **Table 7.2.35: User**

Table Name	**User**			
Field Name	**Data Type**	**Length**	**Nullable**	**Comments**
UserId	**int**		**No**	**Its Primary Key.**
AspNetUserId	**nvarchar**	**128**	**Yes**	**Its Foreign key of AspNetUsers Table.**
FirstName	**nvarchar**	**100**	**No**	
LastName	**nvarchar**	**100**	**Yes**	
Email	**nvarchar**	**50**	**No**	
Mobile	**nvarchar**	**20**	**Yes**	
IsMobile	**bit**		**Yes**	
Street	**nvarchar**	**100**	**Yes**	
City	**nvarchar**	**100**	**Yes**	
State	**nvarchar**	**100**	**Yes**	
RegionId	**int**		**Yes**	
ZipCode	**nvarchar**	**10**	**Yes**	
strMonth	**nvarchar**	**20**	**Yes**	
intYear	**int**		**Yes**	
intDate	**int**		**Yes**	
CreatedBy	**nvarchar**	128	**No**	
CreatedDate	**datetime**		**No**	
ModifiedBy	**nvarchar**	128	**Yes**	
ModifiedDate	**datetime**		**Yes**	
Status	**tinyint**		**Yes**	
IsDeleted	**bit**		**Yes**	
IP	**nvarchar**	20	**Yes**	
IsRequestWithEmail	**bit**			

Chapter 8: TESTING

Testing Plan

A test plan is the cornerstone of a successful testing implementation. The testing plan represents the overall approach to the test. In many ways, the test plan serves as a summary of the test activities that will be performed. It shows how the tests will be organized, and outlines all of the tester's needs that must be met in order to properly carry out the test.
The goal of test planning is to establish the list of tasks that, if performed, will identify all of the requirements that have not been met in the software. There are many standards that can be used for developing test plans. Early in the deployment planning phase, the testing effort, and identifies the methodology that your team will use to conduct tests. It also identifies the hardware, software, and tools required for testing and the features and functions that will be tested. A well- rounded test plan notes any risk factors that jeopardize testing and includes a testing schedule. So, I can say that Test Planning details the activities, dependencies and effort required to conducting the system test.

Test cases :

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)
1	Click Register Button	Blank Fields	Give the message of mandatory field	Give Validation message to enter mandatory field	PASS

2	All Data would be entered	Enter All Data Name : ABC, email : ABC@gmail.com , password : 123456	Give the message of Register Successfully	Register Successfully	PASS

Table 8.1 Test Cases 01 - Login

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)
1	Click Login Button	Blank Fields	Give the message of mandatory field	Give Validation message to enter mandatory field	PASS
2	All Data would be entered	Enter All Data email:ABC@gmail.com, password : 123456	Give the message of Login Successfully	Login Successfully	PASS

Table 8.2 Test Case 02 – Request Form

Step	Test Steps	Test Data	Expected Result	Actual Result	Status (Pass/Fail)
1	Add Address	Blank Fields	Give the message of mandatory field	Give Validation message to enter mandatory field	PASS

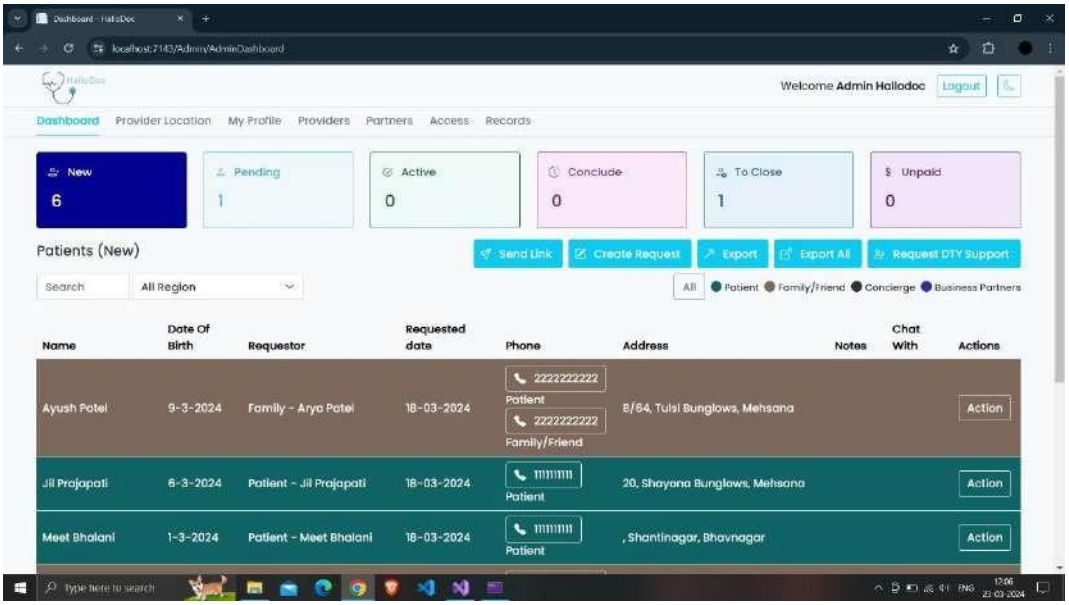
2	All Data would be entered	Enter All Data email:ABC@gmail.com, Name : Delvadiya Shyam, Address : “Shyam” , Krishna Park, Jamjodhpur zip code: 360530 , state : Gujarat , country : India , mobile no :9879879870	Give the message of Add Successfully	Add Successfully	PASS

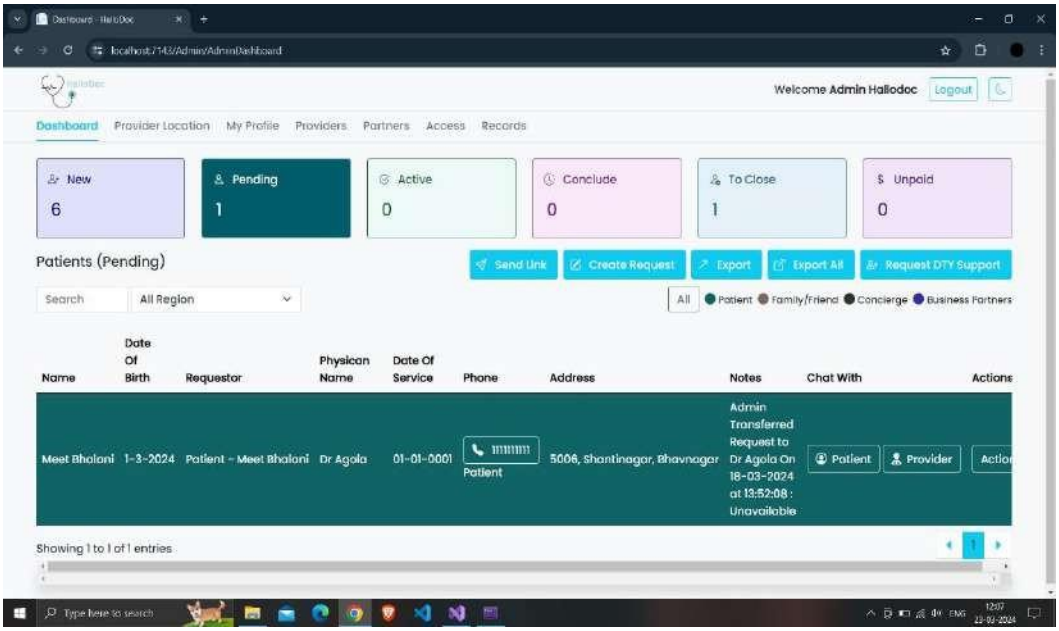
Table 8.3 Test Cases 03 – PopUp Form

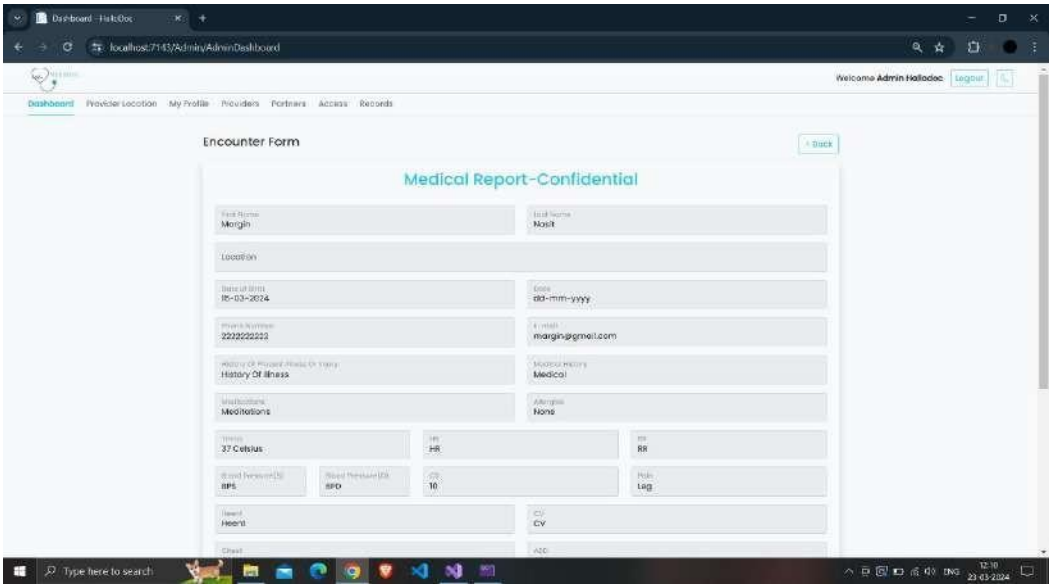
CHAPTER: 9

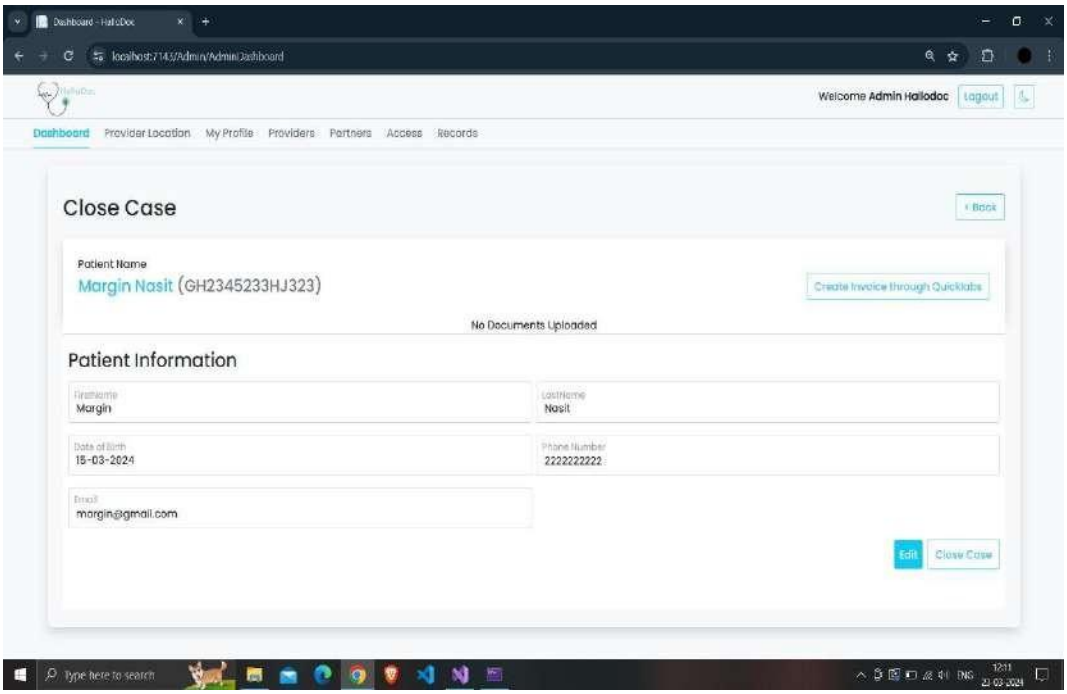
SNAPSHOT OF WEBSITE

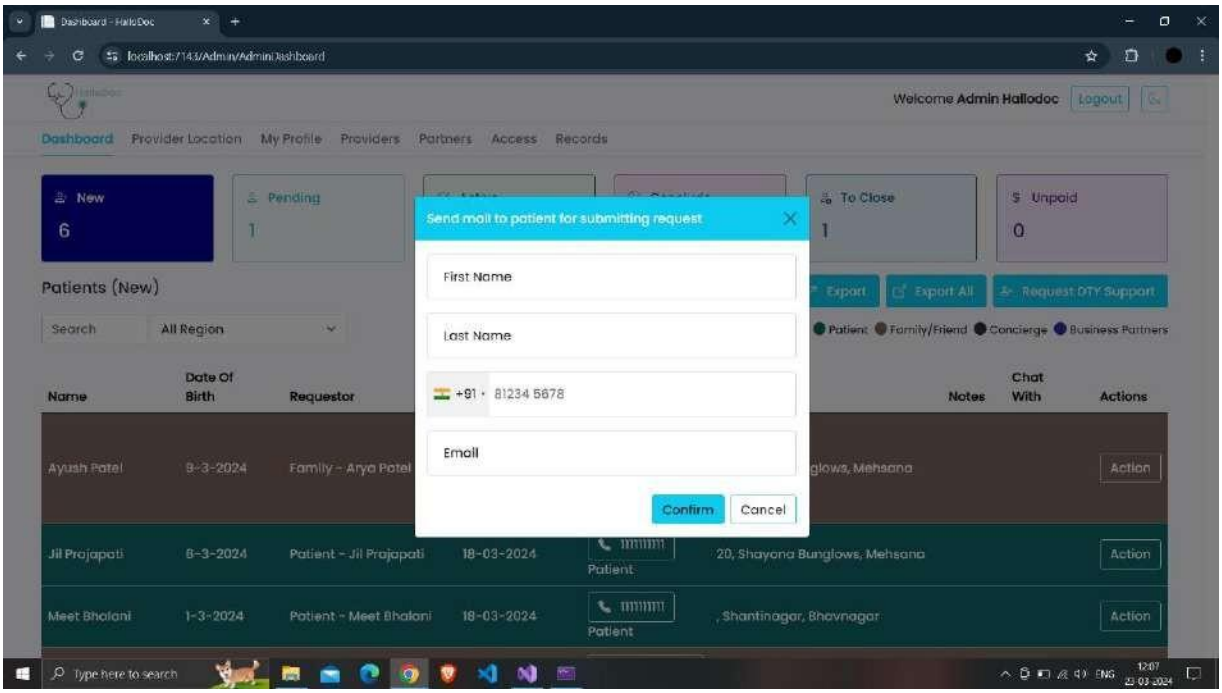
9.1 Admin Site

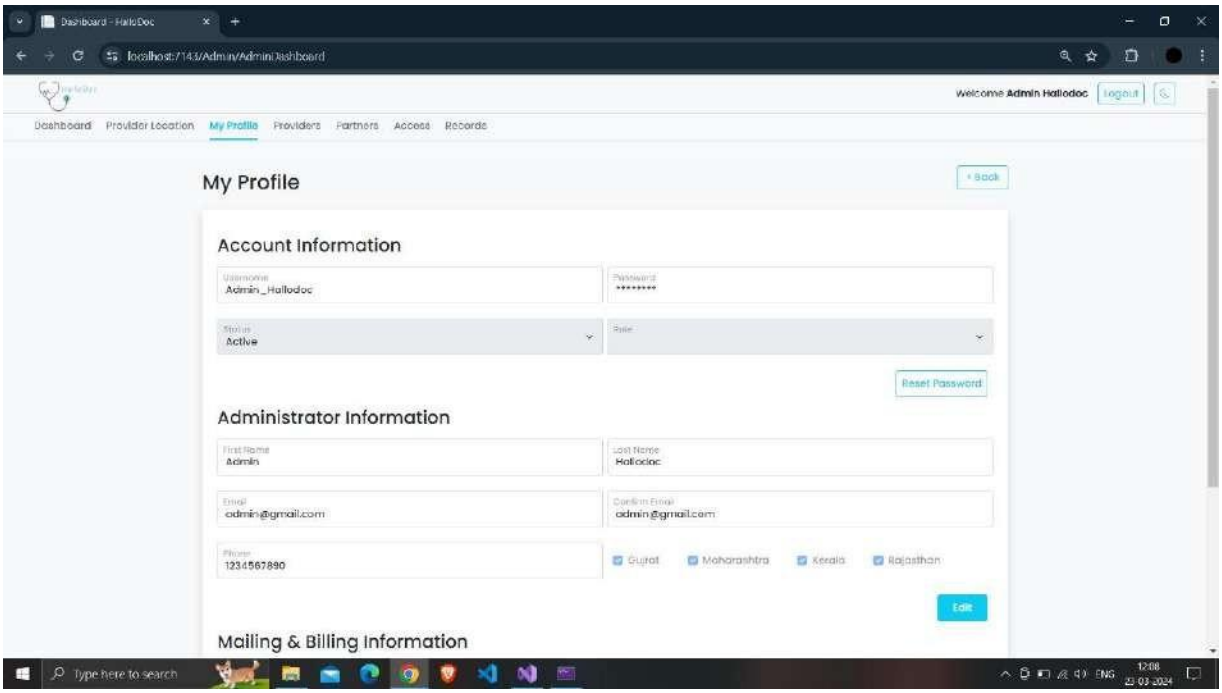


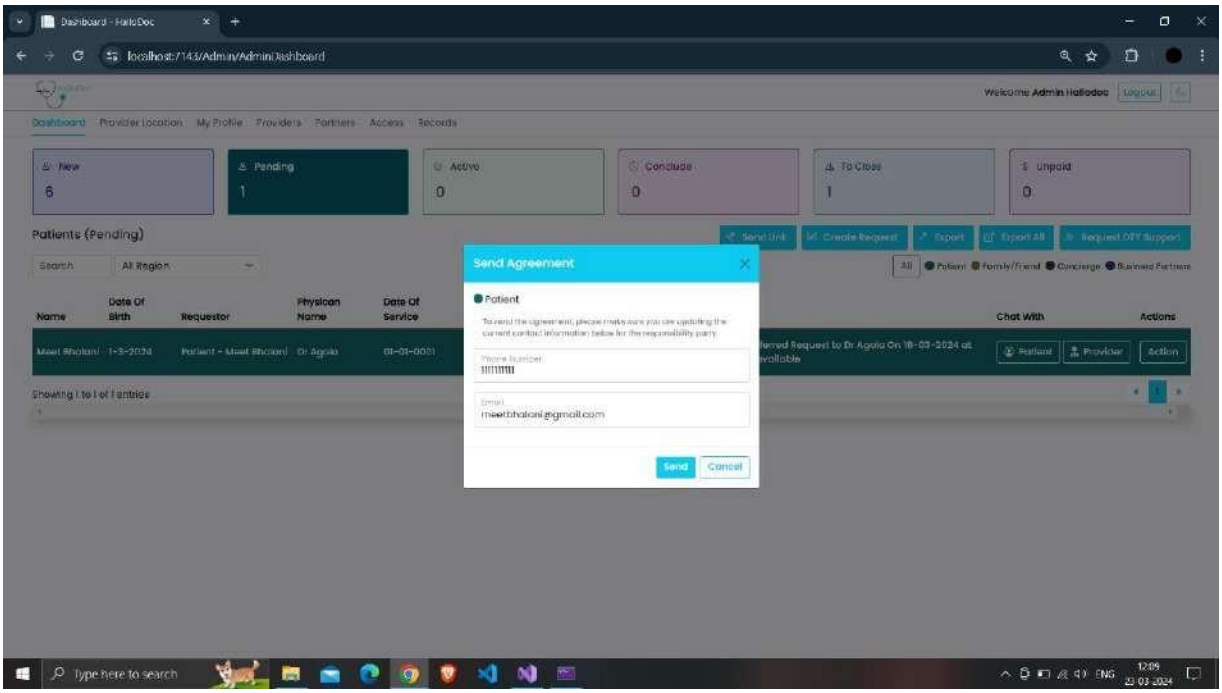



Dashboard - HalloDoc
localhost:7143/Admin/AdminDashboard
Welcome Admin Hallodoc
Logout
Dashboard
Provider Location
My Profile
Providers
Partners
Access
Records
Encounter Form
Back
Medical Report-Confidential
Margin
Nasit
Location
15-03-2024
dd-mm-yyyy
2222222222
margin@gmail.com
History Of Illness
Medical
Medications
None
37 Celsius
HR
RR
BPS
BPD
10
Pain
Lag
Heent
CV
Chest
ABD
Type here to search
ENG
12:10
23-03-2024

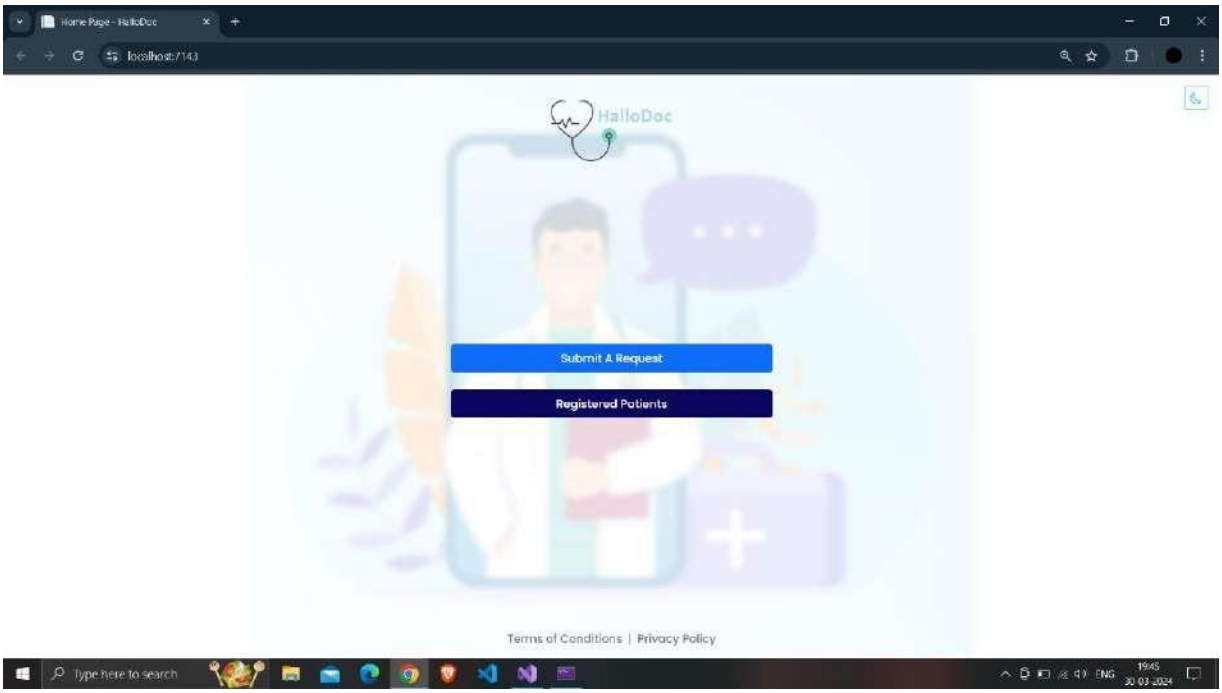

Dashboard - HalloDoc
localhost:7143/Admin/AdminDashboard
Welcome Admin Hallodoc
Logout
Dashboard
Provider Location
My Profile
Providers
Partners
Access
Records
Close Case
Back
Patient Name
Margin Nasit (GH2345233HJ323)
Create Invoice through Quickbooks
No Documents Uploaded
Patient Information
FirstName
Margin
LastName
Nasit
Date of Birth
15-03-2024
Phone Number
2222222222
Email
margin@gmail.com
Edit
Close Case
Type here to search
ENG
12:11
23-03-2024

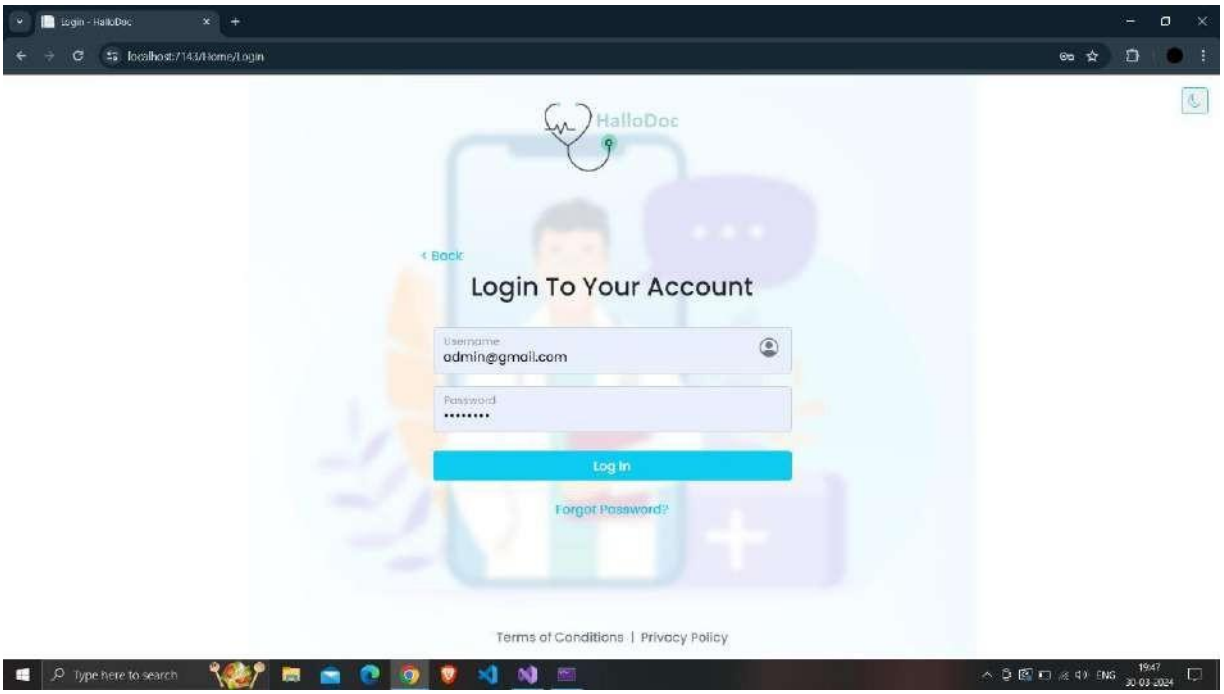

Dashboard - HalloDoc
localhost:7143/Admin/AdminDashboard
Welcome Admin Hallodoc
Logout
Dashboard
Provider Location
My Profile
Providers
Partners
Access
Records
New
6
Pending
1
To Close
1
Unpaid
0
Send mail to patient for submitting request
First Name
Last Name
+91
81234 5678
Email
Confirm
Cancel
Patients (New)
Search
All Region
Export
Export All
Request DTY Support
Patient
Family/Friend
Concierge
Business Partners
Name
Date Of Birth
Requestor
Notes
Chat With
Actions
Ayush Patel
9-3-2024
Family - Arya Patel
Action
Jil Prajapati
8-3-2024
Patient - Jil Prajapati
18-03-2024
Patient
20, Shayona Bunglows, Mehsana
Action
Meet Bhalani
1-3-2024
Patient - Meet Bhalani
18-03-2024
Patient
, Shantinagar, Bhavnagar
Action
Type here to search
ENG
12:07
23-03-2024

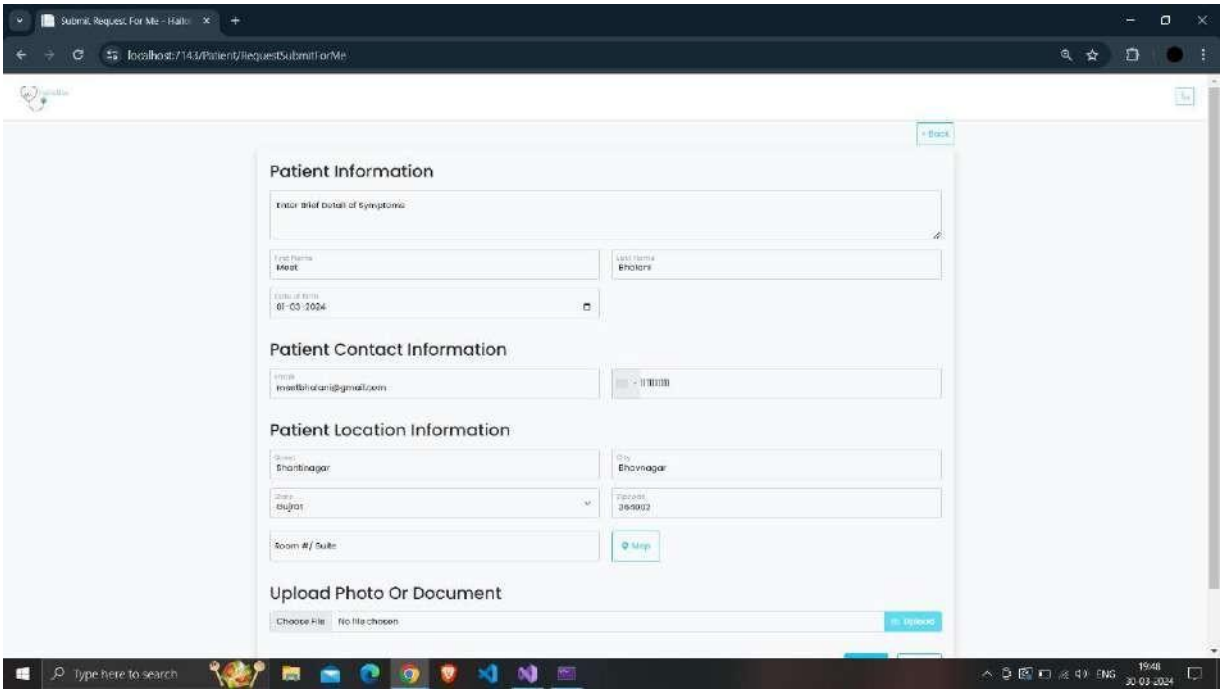

Dashboard - HalloDoc
localhost:7143/Admin/AdminDashboard
Welcome Admin Hallodoc
Logout
Dashboard
Provider Location
My Profile
Providers
Partners
Access
Records
My Profile
Back
Account Information
Username
Admin_Hallodoc
Password
Status
Active
Role
Reset Password
Administrator Information
First Name
Admin
Last Name
Hallodoc
Email
admin@gmail.com
Confirm Email
admin@gmail.com
Phone
1234567890
Gujrat
Maharashtra
Kerala
Rajasthan
Edit
Mailing & Billing Information
Type here to search
ENG
12:08
23-03-2024

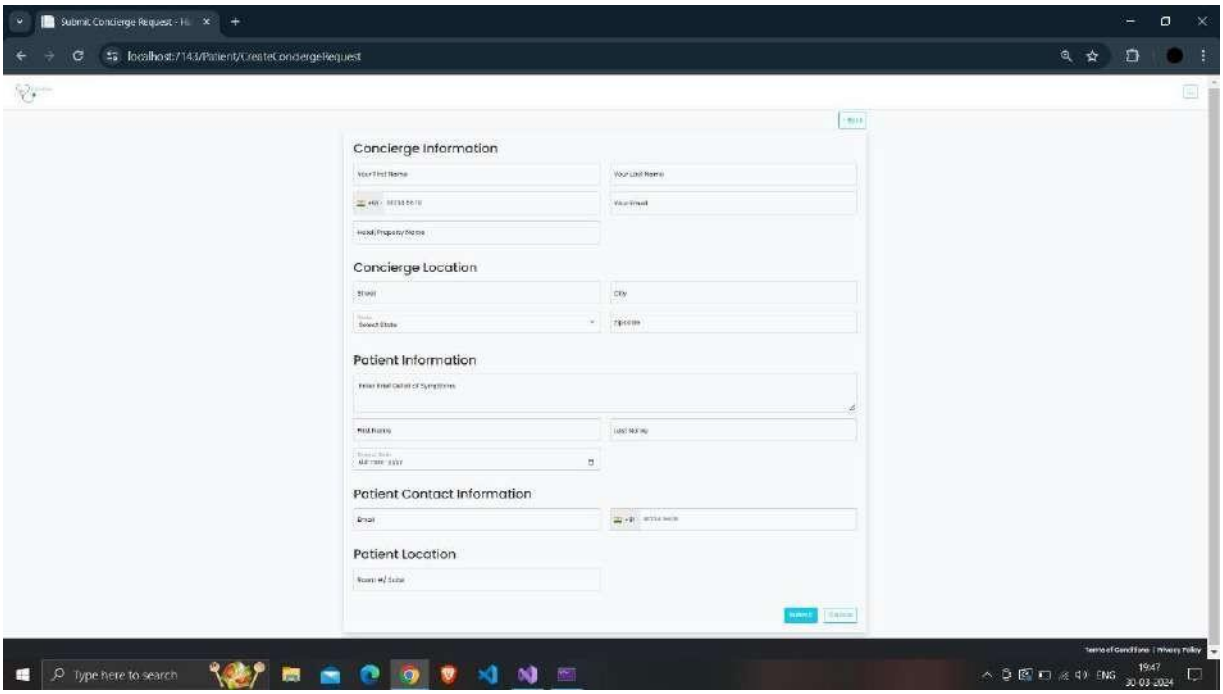


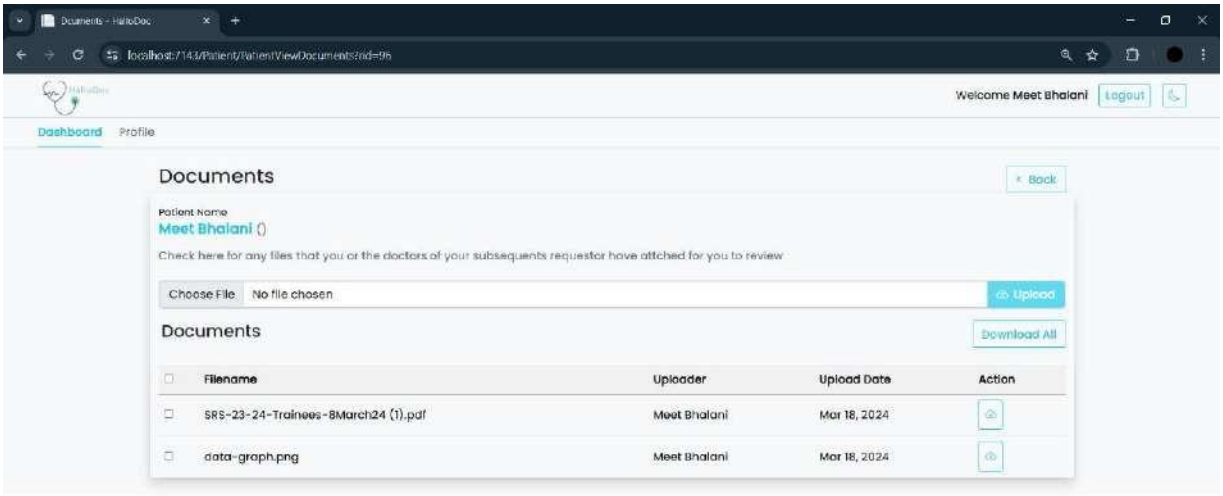
9.2 Users Site

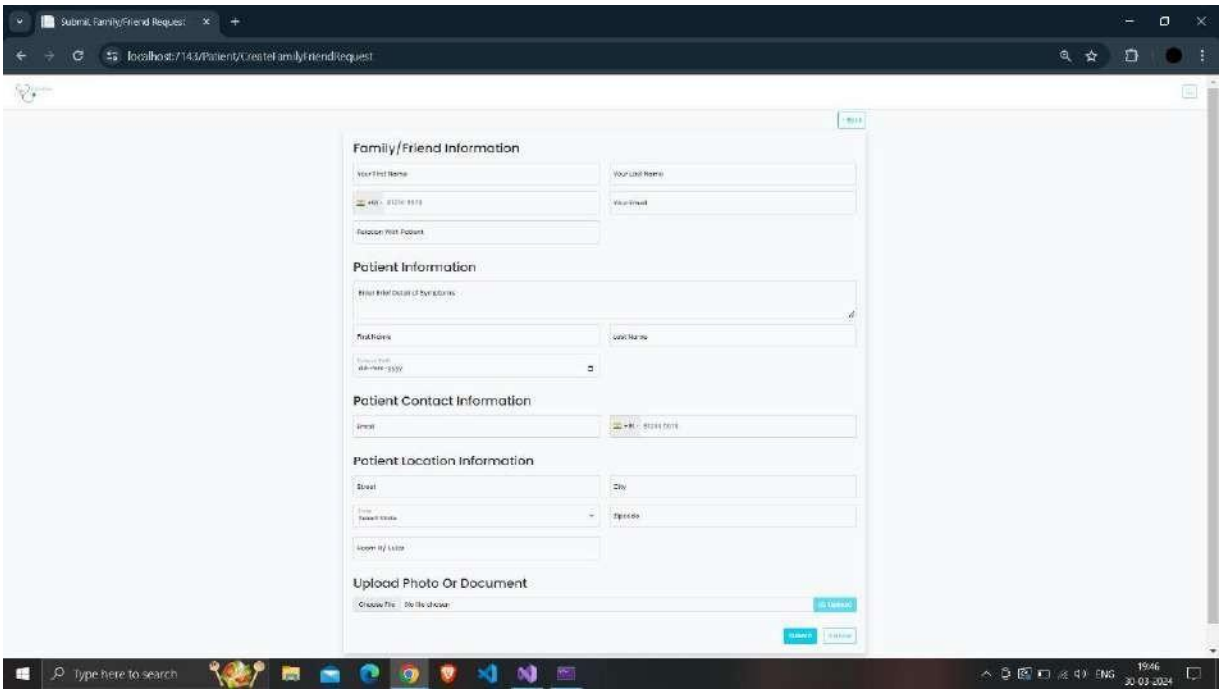


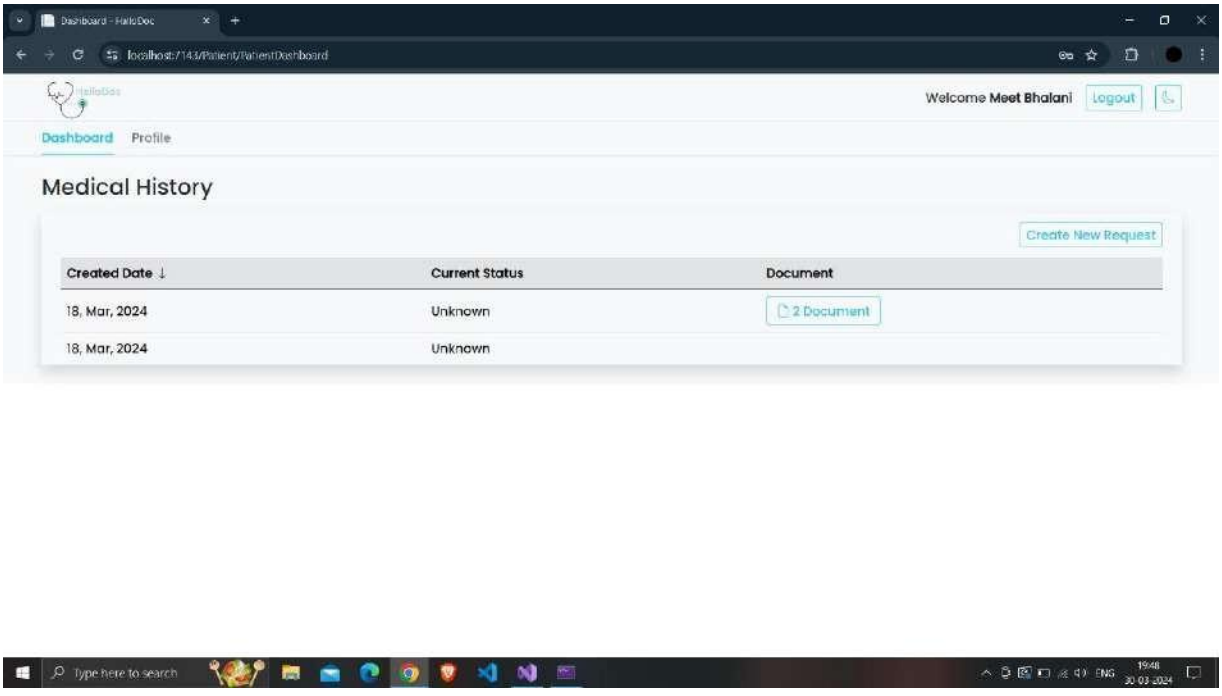

Login - HalloDoc
localhost:7143/Home/Login
HalloDoc
Back
Login To Your Account
Username
admin@gmail.com
Password
Log In
Forgot Password?
Terms of Conditions | Privacy Policy
Type here to search


Submit Request For Me - Hallo
localhost:7143/Patient/RequestSubmitForMe
Patient Information
Enter Brief Detail of Symptoms
Patient Contact Information
Patient Location Information
Bhavnagar
Gujrat
Upload Photo Or Document
Choose File
No file chosen
Type here to search


Concierge Information
Concierge Location
Patient Information
Patient Contact Information
Patient Location


Welcome Meet Bhalani
Logout
Dashboard
Profile
Documents
Back
Patient Name
Meet Bhalani ()
Check here for any files that you or the doctors of your subsequents requestor have attched for you to review
Choose File
No file chosen
Upload
Documents
Download All
Filename
Uploader
Upload Date
Action
SRS-23-24-Trainees-8March24 (1).pdf
Meet Bhalani
Mar 18, 2024
data-graph.png
Meet Bhalani
Mar 18, 2024


Family/Friend Information
Patient Information
Patient Contact Information
Patient Location Information
Upload Photo Or Document


Welcome Meet Bhalani
Logout
Dashboard
Profile
Medical History
Create New Request
Created Date
Current Status
Document
18, Mar, 2024
Unknown
2 Document
18, Mar, 2024
Unknown

CHAPTER 10:

ADVANTAGES

Convenience: Halodoc offers users convenient access to healthcare services from the comfort of their homes. With features such as online consultations and medicine delivery, users can receive medical advice, prescriptions, and medications without the need to travel to a physical clinic or pharmacy.

Accessibility: The platform improves access to healthcare services, especially for individuals living in remote areas or those with limited mobility. Users can connect with qualified doctors and access healthcare resources regardless of their location, helping to bridge the gap between urban and rural healthcare access.

Time-saving: Halodoc saves users time by eliminating the need for long waiting times at clinics or pharmacies. With online consultations and medicine delivery services, users can receive prompt medical attention and prescriptions without having to wait for appointments or queue at pharmacies.

Healthcare Education: Halodoc provides valuable health information and educational resources to users, empowering them to make informed decisions about their health and wellness. Users can access articles, tips, and guides on various health topics, promoting preventive care and healthy lifestyles

Cost-effective: By streamlining healthcare delivery and reducing overhead costs associated with traditional healthcare services, Halodoc offers cost-effective solutions for users. Online consultations may also be more affordable compared to in-person visits to healthcare providers.

Enhanced Patient Care: Halodoc facilitates continuity of care by enabling users to consult with the same doctors across multiple appointments. This fosters a more personalized and patient-centered approach to healthcare, leading to better health outcomes and patient satisfaction.

Efficient Healthcare Delivery: The project optimizes healthcare delivery by leveraging technology to connect users with healthcare providers and streamline administrative processes. This efficiency benefits both users and healthcare professionals, allowing for better resource allocation and utilization.

CHAPTER 11: Conclusion and Future Enhancement

Conclusion:

In conclusion, Halodoc represents a groundbreaking initiative in the healthcare sector, revolutionizing how individuals access and interact with healthcare services in Indonesia. By harnessing the power of technology, Halodoc offers a comprehensive platform that addresses the challenges of convenience, accessibility, and affordability in healthcare delivery.

Through features such as online consultations, medicine delivery, and health education resources, Halodoc empowers users to take control of their health and well-being, regardless of their location or circumstances. The project not only enhances convenience and efficiency but also fosters a patient-centric approach to healthcare, promoting continuity of care and personalized medical attention.

Moreover, Halodoc's impact extends beyond individual users, contributing to the optimization of healthcare delivery systems and resource utilization. By leveraging digital platforms and data-driven insights, Halodoc facilitates more efficient healthcare processes, ultimately leading to improved health outcomes and enhanced patient satisfaction.

Future Enhancement

1. Integration of AI for enhanced diagnostics.
2. Expansion of services to include specialist consultations.
3. Implementation of wearable technology for remote patient monitoring

CHAPTER: 12

BIBLIOGRAPHY

12.1 Course Outcome

Sr. No.	CO statement	Marks % weightage	Chapter No.
CO-1	Undertake problem identification, formulation and solution	20%	• **CHAPTER 1** INTRODUCTION
CO-2	Design engineering solutions to complex problems utilising a systematic approach and team work	30%	• **CHAPTER 2** SYSTEM REQUIREMENTS • **CHAPTER 3** DAILY TASK
CO-3	Communicate with engineers and the community at large in written and oral forms	20%	• **CHAPTER 6** SYSTEM DESIGN • **CHAPTER 10** ADVANTAGES • **CHAPTER 11** CONCLUSION AND FUTURE ENHANCEMENT
CO-4	Demonstrate the knowledge and understanding of engineering and management principle and apply it to assigned project	30%	• **CHAPTER 4** FRONT END OF SYSTEM • **CHAPTER 5** BACK END OF SYSTEM • **CHAPTER 6** SYSTEM DESIGN • **CHAPTER 7** DATA DICTIONARY • **CHAPTER 8** TESTING

12.2 Books

12.3 Web Reference

1. https://online.visual-paradigm.com
2. https://learn.microsoft.com/en-us/aspnet/mvc/
3. https://getbootstrap.com/docs/5.0/getting-started/introduction/
4. https://stackoverflow.com/
5. https://www.w3schools.com/
6. https://www.c-sharpcorner.com/