DEVELOPMENT OF A WEBSITE-BASED HOSPITAL MANAGEMENT SYSTEM

B.Sc. (Engineering) Project

A project Submitted to the Department of Information and Communication Engineering. Faculty of Engineering, University of Rajshahi in partial fulfillment of the degree of Bachelor of Science in Information and Communication Engineering.

SUBMITTED BY

Md. Jahidul Islam

Exam Roll: 1810277130
Session: 2017-2018
Course ID: ICE4252
B.Sc. (Engg.) Part -IV (Even) Exam 2021

SUPERVISED BY

Dr. Riaz Uddin Mondol

Department of Information and Communication Engineering
University of Rajshahi

DEPARTMENT OF INFORMATION AND COMMUNICATION ENGINEERING

UNIVERSITY OF RAJSHAHI

ACKNOWLEDGEMENT

I would like to express my sincere gratitude and appreciation to all the individuals and organizations who have contributed to the successful completion of this project. Their support, guidance, and encouragement have been invaluable throughout this journey.

First and foremost, I would like to extend my deepest gratitude to my project supervisor, *Dr. Riaz Uddin Mondol*, for his unwavering support, expertise, and guidance. His invaluable insights and constructive feedback have been instrumental in shaping the direction and quality of this project. I am truly grateful for his mentorship and encouragement.

I would also like to thank the faculty members and staff of University of Rajshahi, for providing me with the necessary resources and facilities to carry out this project. Their dedication to academic excellence and commitment to student success have been integral to my learning experience.

I am deeply thankful to my friends and classmates who have provided me with support and motivation throughout this project. Their encouragement, discussions, and willingness to lend a helping hand have been crucial in overcoming challenges and maintaining a positive outlook.

Thank you all for being an integral part of this journey and for your invaluable contributions.

ABSTRACT

The Hospital Management System (HMS) project developed is a comprehensive web-based solution designed to automate and streamline the operations of a hospital or healthcare facility. The system provides functionalities for patients, doctors, and administrators to efficiently manage appointments, patient profiles, medical history, and overall database access.

The patient module enables individuals to create accounts, login, and book appointments with doctors. Through a user-friendly interface, patients can browse available doctors, view their profiles, and select a suitable time slot for an appointment. The system validates and records the appointment details, sending confirmation notifications to both the patient and the doctor.

Doctors have their dedicated login portal where they can review and approve appointment requests made by patients. Upon approval, doctors gain access to the patient's profile and medical history, allowing them to provide personalized care and make informed decisions during consultations. They can update medical records, prescribe medication, and schedule follow-up appointments.

The administrator module offers comprehensive control over the system's databases and user management. Administrators can access and manage patient profiles, doctor information, appointment records, and other relevant data. They have the authority to add, modify, or remove user accounts, ensuring the system's integrity and security.

The system utilizes PHP as the server-side scripting language to handle user requests, process data, and interact with the MySQL database. HTML, CSS, and Bootstrap are employed to create a responsive and visually appealing user interface. JavaScript is used for client-side validation and enhancing user interactions, providing a smooth and interactive experience. By implementing the Hospital Management System developed using HTML, CSS, Bootstrap, JS, PHP, and MySQL, healthcare facilities can streamline appointment management, enhance patient-doctor interactions, and improve overall administrative efficiency.

The system empowers patients to conveniently schedule appointments, enables doctors to access patient information and medical history, and grants administrators centralized control over the system's databases. This integrated approach enhances patient care, optimizes resource utilization, and contributes to the efficient management of the hospital. We test the system by providing the data in all the ways and found the system works well.

CONTENTS

Chapters Titles Page

Chapter 1

Introduction

1.1 Background

The development of a Hospital Management System (HMS) stems from the need to address the challenges and limitations faced by healthcare facilities in managing their operations effectively. In traditional hospital settings, manual processes, paper-based systems, and fragmented data management often lead to inefficiencies, errors, and delays in patient care and administrative tasks.

These challenges include difficulties in managing patient records, scheduling appointments, coordinating between different departments, tracking inventory, and generating accurate reports. Paper-based records are susceptible to loss, damage, or misplacement, leading to challenges in accessing critical patient information when needed. Communication between doctors, nurses, and other staff members can be cumbersome, causing delays and miscommunication.

To overcome these challenges, a web-based Hospital Management System (HMS) is developed. This system leverages the power of web technologies to provide a centralized and automated solution that streamlines various processes, improves communication and coordination among hospital staff, enhances patient care, and enables remote accessibility to information and services.

1.2 Problem Statement

In an increasingly digital era, patients expect convenient access to healthcare services and information. The system should provide a centralized and automated solution that streamlines various processes, improves communication and coordination among hospital staff, patients and doctors and enables remote accessibility to information and services.

The development of a Hospital Management System will be the solution of the the pressing need for a centralized, automated, and web-based solution in healthcare facilities.

1.3 Objectives

Automation of Appointment Management: Enable patients to conveniently login, browse available doctors, and book appointments online. Streamline the appointment scheduling process, reducing waiting times and administrative burden.

Patient Convenience and Accessibility: Provide a user-friendly interface for patients to access the system, allowing them to easily book appointments from anywhere at any time. Enhance patient experience and satisfaction by offering a convenient online platform.

Efficient Doctor Appointment Approval: Enable doctors to login, review appointment requests, and approve or reject them based on availability. Empower doctors to manage their schedules effectively and ensure optimal utilization of their time. Access to Patient Profiles and Medical History: Allow doctors to access patient profiles and medical history upon approval of appointments. Provide doctors with comprehensive information to deliver personalized care, make informed decisions, and facilitate better treatment outcomes.

Centralized Database Management: Develop an administrative module for authorized personnel to login and access the overall databases of users. Enable administrators to manage patient profiles, doctor information, appointment records, and other relevant data efficiently.

Enhanced Data Security and Integrity: Implement robust security measures to protect patient data and ensure data integrity throughout the system. Employ secure login mechanisms, role-based access controls, and encryption techniques to safeguard sensitive information.

Streamlined Administrative Processes: Simplify administrative tasks through automation, reducing manual effort and minimizing errors. Enhance administrative efficiency by providing a centralized system for managing appointments, patient records, and overall database access.

Reporting and Analytics: Incorporate reporting and analytics functionalities to generate insightful reports, performance metrics, and data analysis. Assist administrators in making informed decisions, identifying trends, and optimizing hospital operations.

Scalability and Flexibility: Design the system to be scalable and flexible, allowing for future expansion and integration with other healthcare systems or modules. Accommodate the growing needs of the hospital and adapt to changing technological requirements.

Improved Hospital Management and Performance: By achieving the above objectives, the project aims to enhance the overall management and performance of the hospital. Increase operational efficiency, optimize resource utilization, improve patient care and satisfaction, and contribute to the success of the healthcare facility.

1.4 Scope

The scope of the project encompasses the development and implementation of a comprehensive Hospital Management System, covering patient management, doctor management, appointment management, medical records, database management, user interface, security, and scalability. The system aims to enhance the efficiency, accuracy, and patient care within the hospital while providing a user-friendly and secure platform for patients, doctors, and administrators.

1.5 Significance

The significance of the Hospital Management Database System project lies in its ability to transform hospital operations, enhance patient care, improve communication and coordination, enable data-driven decision making, and ensure compliance with regulatory standards. It plays a vital role in modernizing healthcare facilities, optimizing resources, and providing efficient and effective healthcare services to patients.

Moreover, the Hospital Management Database System project offers a wide range of significances, including cost reduction, improved resource utilization, streamlined financial management, remote accessibility, enhanced data accuracy and availability, support for research and analytics, integration with external systems, and opportunities for continuous improvement and scalability.

Chapter 2
System Overview

2.1 System Architecture

The system architecture for the Hospital Management System (HMS) project developed by following a 3-layered architecture pattern. This pattern separates the different components and functionalities of the system into distinct layers.

1. Presentation Layer:
    - This layer focuses on the user interface and user interaction with the system.
    - Technologies used: HTML, CSS, Bootstrap, JS.
    - Responsibilities:
        1. Handles the visual representation of the system, including web pages, forms, and user input validation.
        2. Implements responsive design to ensure optimal user experience across different devices.
        3. Facilitates user interaction and provides feedback through interactive elements and UI components.
        4. Communicates with the backend layer to retrieve and display data to the user.
2. Application layer:
    - This layer contains the core application logic and acts as an intermediary between the presentation and data layers.
    - Technologies used: PHP, JavaScript (Server-side).
    - Responsibilities:
        1. Processes user requests and coordinates the flow of data between the presentation and data layers.
        2. Implements business rules and algorithms for appointment booking, approval, and other functionalities.
        3. Validates user input and enforces data integrity before interacting with the data layer.
        4. Interacts with the data layer to retrieve, update, and store data.
3. Data Layer:
    - This layer manages the storage and retrieval of data in the system.
    - Technologies used: MySQL (Relational Database Management System).

- Responsibilities:
  1. Creates and maintains the database schema to store user profiles, appointment records, and medical history.
  2. Provides mechanisms for secure and efficient data access and retrieval.
  3. Implements data validation and integrity constraints to ensure the accuracy and consistency of data.
  4. Supports complex queries and transactions to handle data retrieval and updates.

Illustration of the system architecture:

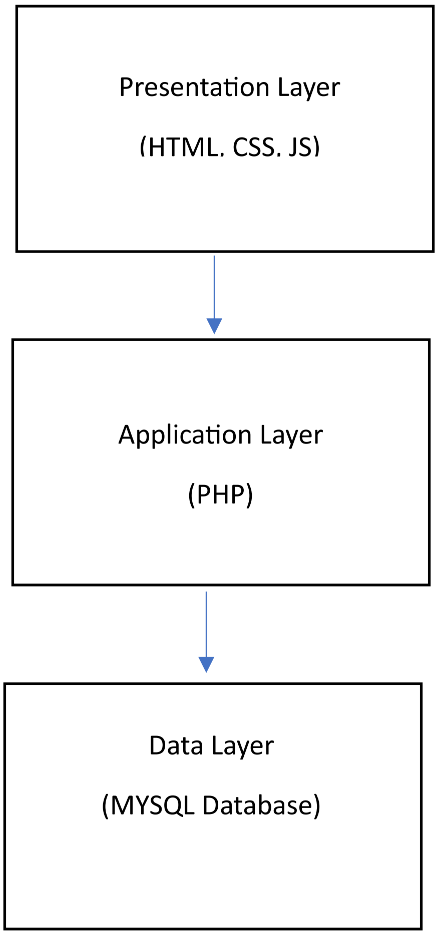


Fig: Simplified diagram of the of the HMS system architecture

2.2 Technology Stack

2.2.1 Front End Site:

- HTML (Hypertext Markup Language)
- CSS (Cascading Style Sheets)
- JS (JavaScript)

2.2.2 Back End Side:

- PHP (Hypertext Preprocessor)

2.2.3 Server

- Apache Server

2.2.4 Database Management System

- MySQL

Chapter 3
System Development Tools

3.1 WINDOWS 10

Windows 10 is a series of operating systems developed by Microsoft and released as part of its Windows NT family of operating systems. It is the successor to Windows 8.1, released nearly two years earlier, and was released to manufacturing on July 15,2015, and broadly released for the general public on July 29, 2015. Here Windows 10 operating system is used.

3.2 HTML

HTML, short for Hypertext Markup Language, is the standard markup language used for creating the structure and content of web pages. It provides a set of tags and elements that define the structure, layout, and formatting of a web document. HTML is the backbone of the World Wide Web and is essential for building static web pages and web applications.

With HTML, developers can structure the content of a web page by using various tags such as headings, paragraphs, lists, images, links, tables, and forms. These tags define the different elements and their relationships within the document, allowing browsers to interpret and display the content correctly.

HTML is designed to be platform-independent and can be rendered by any web browser. It provides a standard set of tags and elements that ensure consistent display across different devices and platforms. However, the visual appearance of HTML elements can be further enhanced and styled using Cascading Style Sheets (CSS).

In summary, HTML is the foundation of web development, providing a markup language for creating the structure and content of web pages. It allows developers to define the elements and their relationships within a document, providing a consistent and organized representation of information. HTML, in conjunction with CSS and JavaScript, forms the building blocks of the modern web, enabling the creation of interactive and visually appealing websites and web applications.

3.3 CSS

CSS (Cascading Style Sheets) is a stylesheet language used for describing the presentation and styling of HTML and XML documents. It is a fundamental component of web development, allowing developers to control the layout, colors, fonts, and other visual aspects of web pages. CSS separates the content and structure of a web page from its presentation, making it easier to maintain and update the design across multiple pages or an entire website. With CSS, developers can create attractive and consistent user interfaces, enhance accessibility, and adapt the layout for different devices and screen sizes. It offers a wide range of selectors, properties, and values to target specific elements and apply styling rules, providing flexibility and customization options for web design.

3.4 JavaScript

JavaScript is a high-level, interpreted programming language primarily used for creating dynamic and interactive web content. It runs on the client-side, meaning it is executed by the user's web browser. JavaScript allows developers to add interactivity, perform calculations, manipulate web page elements, handle events, and make HTTP requests, among other functionalities. It is a versatile language that can be used for both front-end and back-end development. With JavaScript, developers can create engaging user experiences, validate form inputs, create animations, implement real-time updates, and build web applications with rich functionality.

3.6 PHP

PHP (Hypertext Preprocessor) is a server-side scripting language used primarily for web development. It is widely used for creating dynamic and interactive web applications, handling form submissions, generating dynamic web content, and interacting with databases. PHP is embedded within HTML code and executed on the server before being sent to the client's browser, allowing for dynamic content generation.

Key points about PHP:

1. **Server-Side Scripting**: PHP is executed on the server-side, meaning the server processes the PHP code and generates the resulting HTML or other output, which is then sent to the client's browser. This enables the creation of dynamic web pages that can change based on user input, database queries, or other conditions.
2. **Easy Integration:** PHP can be seamlessly integrated with HTML, CSS, and JavaScript. It can be embedded directly within HTML code using special tags, making it straightforward to mix PHP logic with HTML markup.
3. **Wide Range of Applications**: PHP is versatile and can be used to build various types of web applications, including e-commerce websites, content management systems (CMS), social networking platforms, forums, and more. It has extensive libraries and frameworks that simplify common tasks and accelerate development.
4. **Database Interaction:** PHP provides native support for connecting and interacting with databases, including MySQL, PostgreSQL, and SQLite. This allows developers to perform database operations, such as querying, inserting, updating, and deleting data, making PHP suitable for building data-driven web applications.
5. **Community and Ecosystem**: PHP has a large and active community of developers, offering abundant resources, documentation, and community support. There is a wide range of open-source libraries, frameworks (such as Laravel, Symfony, and

CodeIgniter), and tools available that extend the functionality of PHP and facilitate rapid application development.

6. **Cross-Platform Compatibility:** PHP is a cross-platform language, compatible with various operating systems such as Windows, macOS, Linux, and Unix. This flexibility enables developers to deploy PHP applications on different web servers, including Apache, Nginx, and Microsoft IIS.
7. **Performance**: PHP has evolved over the years and has made significant improvements in terms of performance and speed. With the introduction of PHP 7 and subsequent versions, the language has become more efficient, reducing the execution time of scripts and enhancing overall performance.

In summary, PHP is a server-side scripting language widely used for web development. It enables the creation of dynamic web pages, interacts with databases, integrates with HTML and other front-end technologies, and offers a vast ecosystem of libraries and frameworks. PHP's ease of use, versatility, and extensive community support make it a popular choice for building dynamic and data-driven web applications.

3.7 MySQL

MySQL is a popular open-source relational database management system (RDBMS) that is widely used for storing, managing, and retrieving structured data. It provides a powerful and scalable solution for handling data in various applications, ranging from small websites to large-scale enterprise systems.

One of its notable features is its ease of use. MySQL has a user-friendly interface and a simple yet powerful query language, known as Structured Query Language (SQL). This allows developers to efficiently interact with the database, perform complex queries, and manipulate data with ease.

MySQL is highly scalable, capable of handling large amounts of data and serving multiple users simultaneously. It supports efficient indexing and advanced caching mechanisms, ensuring fast and optimized data retrieval even in high-demand environments.

Another advantage of MySQL is its robust security features. It provides user authentication and access control mechanisms, allowing administrators to define user privileges and restrict unauthorized access to data. MySQL also offers encryption capabilities to protect sensitive information stored in the database.

Furthermore, MySQL supports ACID (Atomicity, Consistency, Isolation, Durability) properties, which ensure data integrity and reliability. Transactions can be used to group multiple database operations into a single atomic unit, ensuring that either all operations are completed successfully or none are applied. This helps maintain the consistency and integrity of the data.

MySQL has excellent cross-platform compatibility, making it compatible with various operating systems such as Windows, macOS, Linux, and more. It can be seamlessly integrated with popular web development technologies like PHP, allowing developers to build dynamic web applications that interact with the database efficiently.

In summary, MySQL is a powerful and versatile RDBMS that offers a range of features to efficiently manage and retrieve structured data. Its ease of use, scalability, security, and robustness make it an ideal choice for various applications, from small websites to enterprise-level systems. The strong community support further enhances its appeal and ensures that developers have the resources and assistance they need to leverage the full potential of MySQL in their projects.

3.8 VS CODE

Visual Studio Code, commonly referred to as VS Code, is a popular source code editor developed by Microsoft. It is a lightweight and versatile tool that provides developers with a productive and customizable environment for writing, editing, and debugging code. With its extensive features and broad language support, Visual Studio Code has gained widespread adoption among developers across different platforms and programming languages.

The editor supports a broad range of programming languages, including but not limited to JavaScript, Python, C++, Java, and HTML. It offers syntax highlighting, code formatting, and other language-specific features, enhancing code readability and improving the overall development experience.

The editor is available for multiple operating systems, including Windows, macOS, and Linux, ensuring cross-platform compatibility and enabling developers to work in their preferred environment.

3.9 Apache Server

Apache Server, commonly known as Apache HTTP Server or simply Apache, is the world's most widely used web server software. It is an open-source, cross-platform server that enables the hosting and delivery of web content over the internet. Apache offers a robust and flexible platform for serving static and dynamic web pages, handling HTTP requests, and managing various web-related tasks.

Apache supports a range of operating systems, including Windows, macOS, Linux, and Unix-like systems, making it highly versatile and compatible with various server environments. It also integrates seamlessly with popular scripting languages such as PHP, Perl, and Python, allowing for the execution of dynamic web applications.

Apache Server is a highly scalable, reliable, and customizable web server software. Its versatility, cross-platform compatibility, extensive configuration options, modular architecture, and robust security features have made it a popular choice for hosting websites and serving web content across the internet.

Chapter 4

Requirement Analysis

4.1 Functional Requirements

- **Patient Management:** Allow patients to registration and login, update profile, book appointment, check appointment history and medical history.
- **Doctor Management:** Allow doctors to registration and login, update profile, view appointment history, add and manage patients and search functionality.
- **Appointment Management:** Allow patients to book appointments with doctors online. Enable doctors to view manage appointment requests.
- **Medical Records and History:** Allow to store and retrieve patients medical records and history securely. Allow doctors to view the medical history of patients, including past treatments and surgeries.
- **Reporting and Analytics**: Provide analysis and insights on patient visits, treatments, and outcome.
- **System Administration:** Admins only have login options and allow to manage patients, users, doctors and check queries and check session logs, appointments and reports. Admin can add and remove patients and doctors.

4.2 Non-functional requirements

- **Performance:** The system should have fast response times, ensuring efficient handling of user requests. It should be able to handle a high volume of concurrent users without significant degradation in performance
- **Reliability:** The system should be highly reliable, ensuring minimal downtime and interruptions.
- **Security:** The system should enforce strict access control to ensure that only authorized personnel can access sensitive patient information.
- **Scalability:** The system should be scalable, capable of handling increasing data volumes, users, and transactions without significant performance degradation.
- **Usability:** The system should have a user-friendly and intuitive interface, ensuring ease of use for doctors, staff, and patients.
- **Compatibility:** The system should be compatible with various web browsers, operating systems, and devices to ensure broad accessibility.
- **Data Integrity and Privacy**: The system should enforce data integrity rules to ensure accurate and reliable information storage and retrieval.

4.3 Constraints

- **Budget:** The project should be developed within a specified budget.
- **Time:** The project should be developed within the specified timeline.
- **Technology:** The project should be developed using specific technologies that are compatible with the existing IT infrastructure.
- **Legal and Regulatory Compliance:** The website should comply with legal and regulatory requirements related to data privacy, intellectual property rights, and accessibility.
- **Skill:** The hospital management website should be developed within the available skills of the students who are tasked with its development.

4.4 Use Case

Use Case: Hospital Management System (HMS)

Actors:

- Patient

- Doctor

- Administrator

Use Cases:

1. Register Patient: The Patient creates an account in the system.

2. Login: The Patient, Doctor, or Administrator logs into the system.

3. Book Appointment: The Patient requests an appointment with a Doctor.

4. View Appointments: The Patient can view their scheduled appointments.

5. Cancel Appointment: The Patient cancels a previously booked appointment.

6. Provide Medical Services: The Doctor provides medical services to the Patients.

7. Update Patient Record: The Doctor updates the Patient's medical records.

8. Manage Doctor Schedule: The Administrator manages the schedule of Doctors.

9. Manage Patient: The Administrator manages Patient information.

10. Manage Doctor: The Administrator manages Doctor information.

11. Generate Reports: The Administrator generates various reports related to appointments, Patients, or Doctors

Relationships:

Patient and Doctor: Association relationship, as a Patient can have appointments with a Doctor.

Administrator and Doctor: Association relationship, as the Administrator manages Doctor information.

Administrator and Patient: Association relationship, as the Administrator manages Patient information.

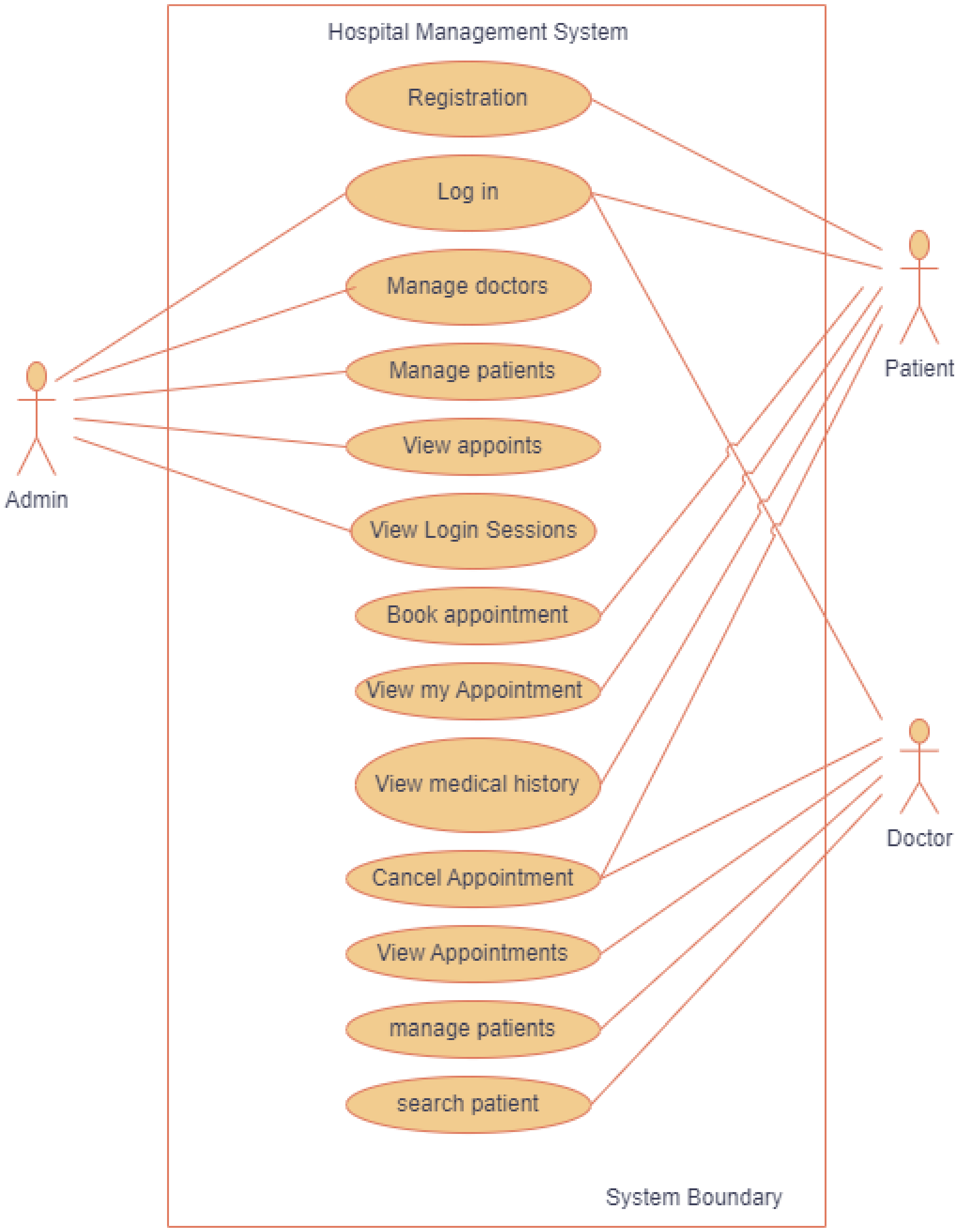


Fig: Use case diagram of Hospital Management System

Chapter 5

Design and Implementation

5.1 Home Page

We have tried to make website interface user-friendly and easy to navigate so that users can easily find the services. Below we will discuss about our project named Hospital Management System.

5.1.1 Home

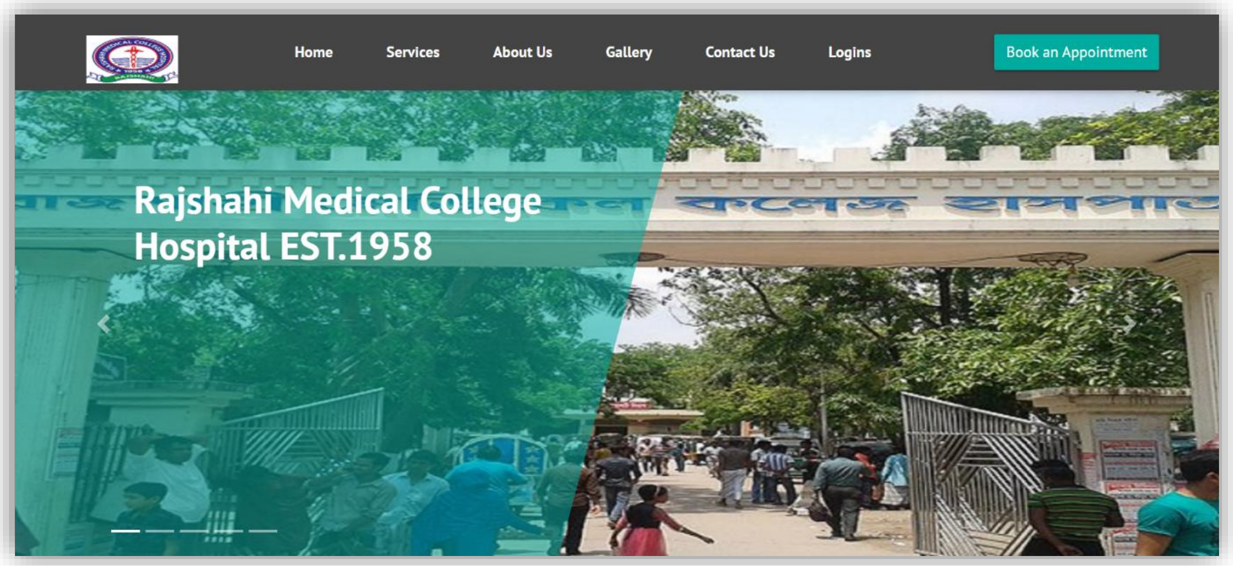


Fig 1: Front Page

This is opening page of our website. Anyone who will visit our website will see this page as a home page of this site. This page is covered the featured photos and events photos as slider. We have added 5 sliders with Zoom In feature. The picture will zoom in and change correspondingly.

5.1.2 Services

Services section represents a list of all services provide by the hospital. Services option added in the menu so that users can easily access the services of the hospital. It will be helpful for the users to know about their seeking department whether exists or not. There have lots of thing we can add in services section. But there we have just list the services and departments of the hospital. Users just can see the departments name whether the user seeking department exists in the hospital or not.

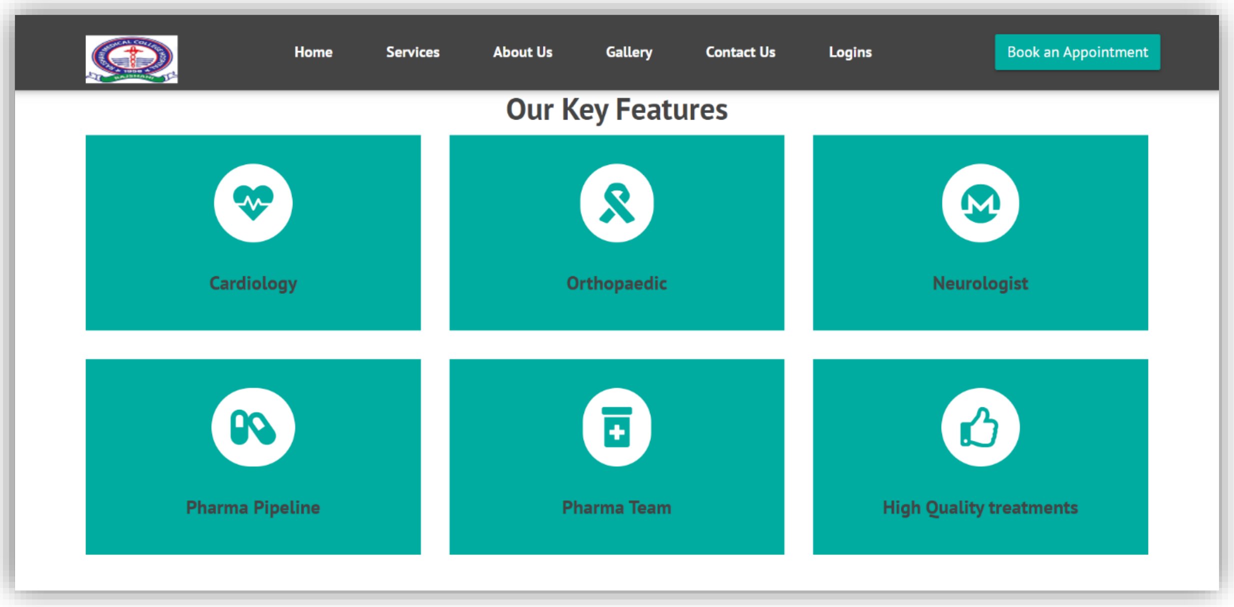
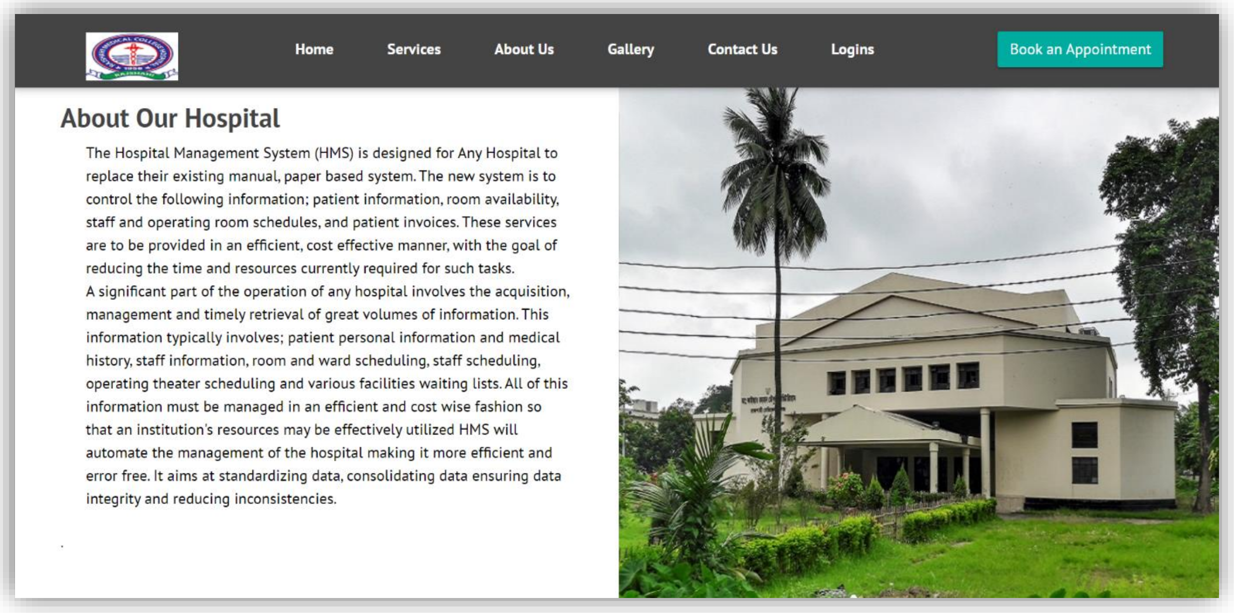


Fig 2: Services

5.1.3 About Us

About us section has been filled with the organization’s motives, good work and their services and medals and the effectiveness and about the quality of their services. It actually presents the hospital overview in short. Users will see the about us section like the followings below.



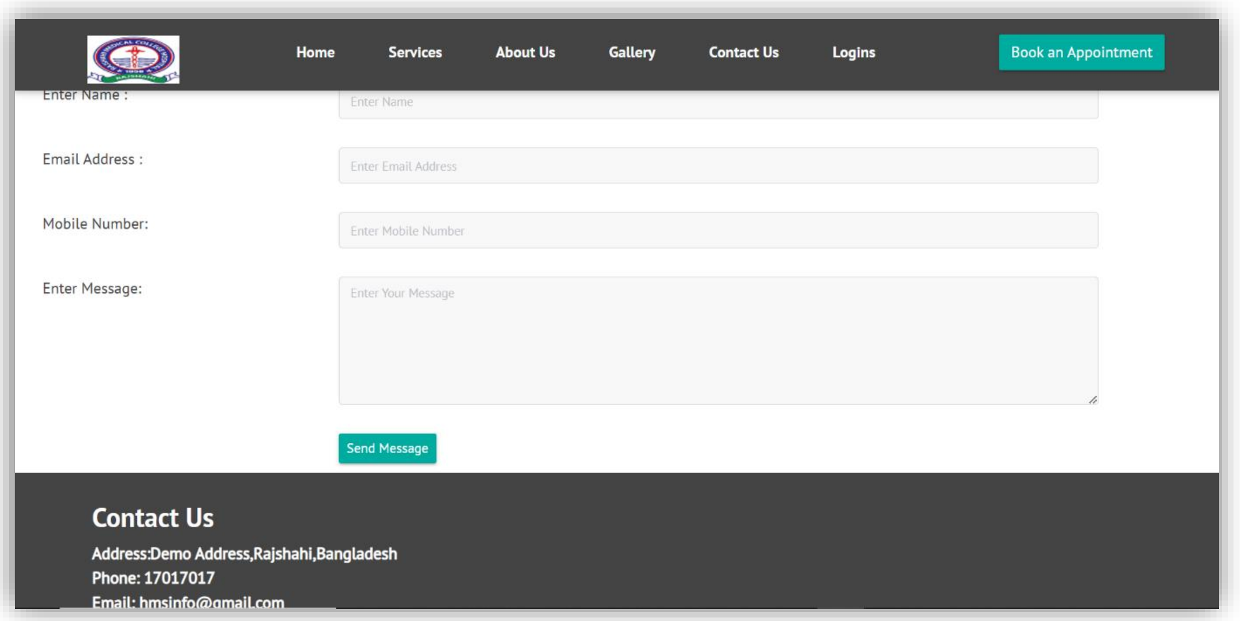
About Our Hospital

The Hospital Management System (HMS) is designed for Any Hospital to replace their existing manual, paper based system. The new system is to control the following information; patient information, room availability, staff and operating room schedules, and patient invoices. These services are to be provided in an efficient, cost effective manner, with the goal of reducing the time and resources currently required for such tasks.
A significant part of the operation of any hospital involves the acquisition, management and timely retrieval of great volumes of information. This information typically involves; patient personal information and medical history, staff information, room and ward scheduling, staff scheduling, operating theater scheduling and various facilities waiting lists. All of this information must be managed in an efficient and cost wise fashion so that an institution's resources may be effectively utilized HMS will automate the management of the hospital making it more efficient and error free. It aims at standardizing data, consolidating data ensuring data integrity and reducing inconsistencies.

Fig 3: About us

5.1.4 Contact Us

Anybody can contact with the hospital admins by sending messages for any needs. Contact section added in the home page so that anybody can contact with hospital admin emergency without any login hassle. There has a contact form so anybody could send any message. Besides there has the address, email and contact no. of the hospital so that users can easily find and communicate with the hospital authority.



Enter Name : Enter Name

Email Address : Enter Email Address

Mobile Number: Enter Mobile Number

Enter Message: Enter Your Message

Contact Us

Address:Demo Address,Rajshahi,Bangladesh
Phone: 17017017
Email: hmsinfo@gmail.com

Fig: Contact Us

5.1.5 Logins

Visitors will see the Logins section here. Patients, Doctors, Admins have to login for the respective functions such as patients have to register and login for booking appointments and doctors have to login to view the appointment history and the patients list and medical history, admins have to login for their respective work. Logins section has the Login interface for patients, doctors, admins.

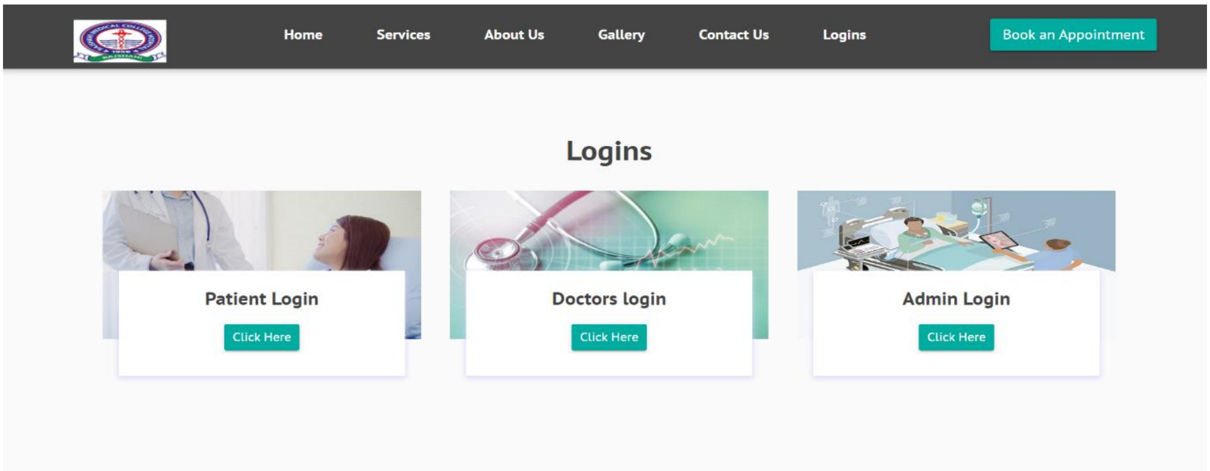
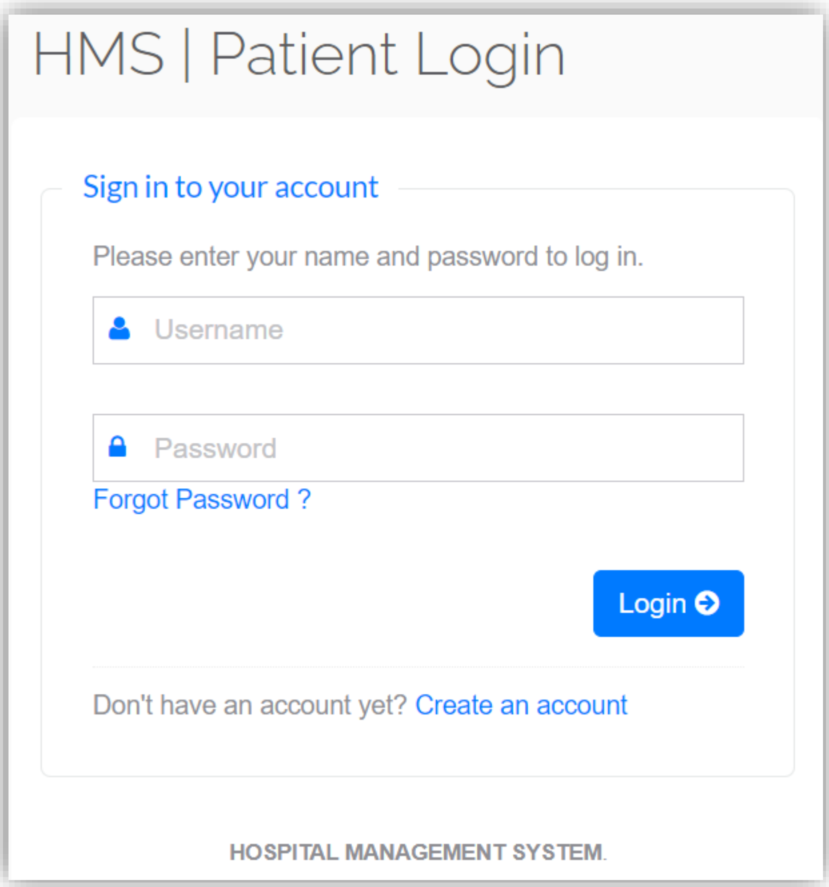


Fig: Logins

5.2 Patient Login

This is patient login section. Users can be loggedin who have already registered. There have username and password which is needed for login.



HMS | Patient Login

Sign in to your account

Please enter your name and password to log in.

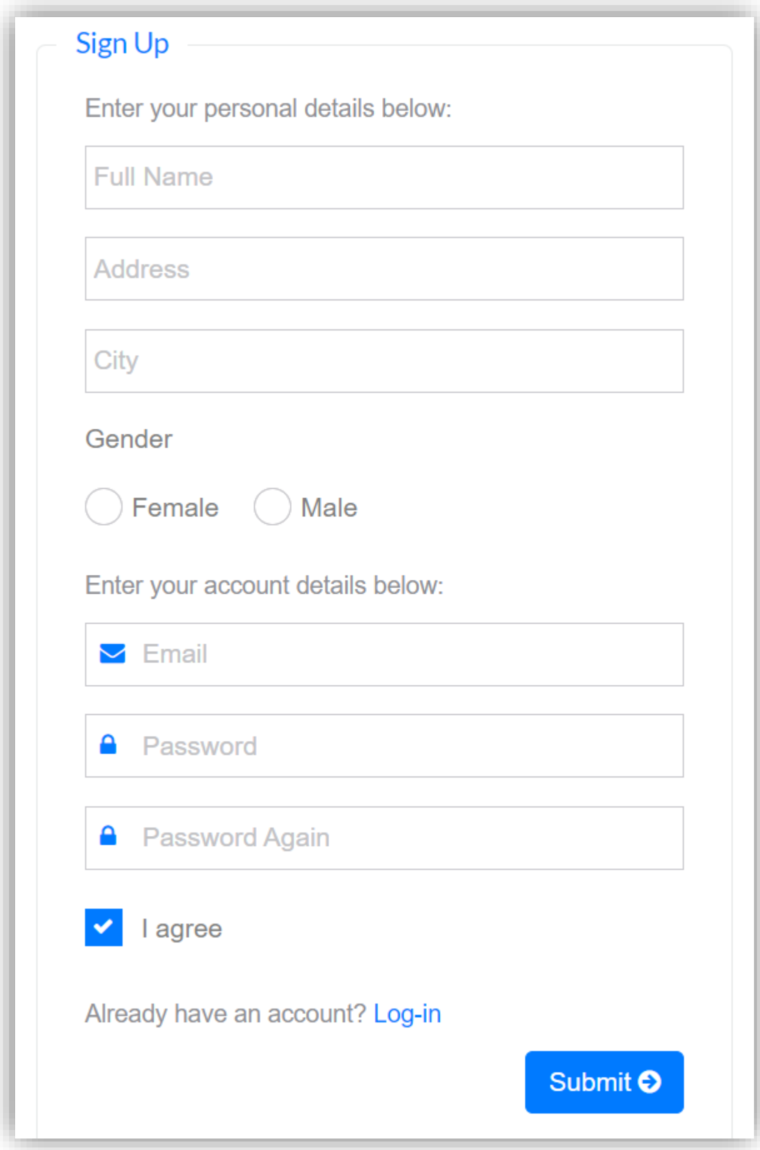
Don't have an account yet? Create an account

HOSPITAL MANAGEMENT SYSTEM.

Fig 6: Patient Login

5.3 Patient Registration

For registration users must to fill the options in the following Sign Up. This is compulsory for the users to login. In the login page, there has an Create an Account option, users will be redirected to the registration page by clicking it. If the users don't tick the agree option form won't be submitted hence fail in sign up. We have already ticked the agree option by program so that users don't have any hassle.



Sign Up

Enter your personal details below:

Full Name

Address

City

Gender

○ Female ○ Male

Enter your account details below:

Email

Password

Password Again

☑ I agree

Already have an account? Log-in

Fig 7: Users Registration page

5.4 Patient Dashboard

After login a user, he/she will get this page. There have My profile Option, my appointments option and book my appointment. User can update his/her profile by clicking My Profile and he/she can book appointment by clicking Book My Appointment and can view the appointment history by clicking My Appointments. Further there has Medical History of the patients.

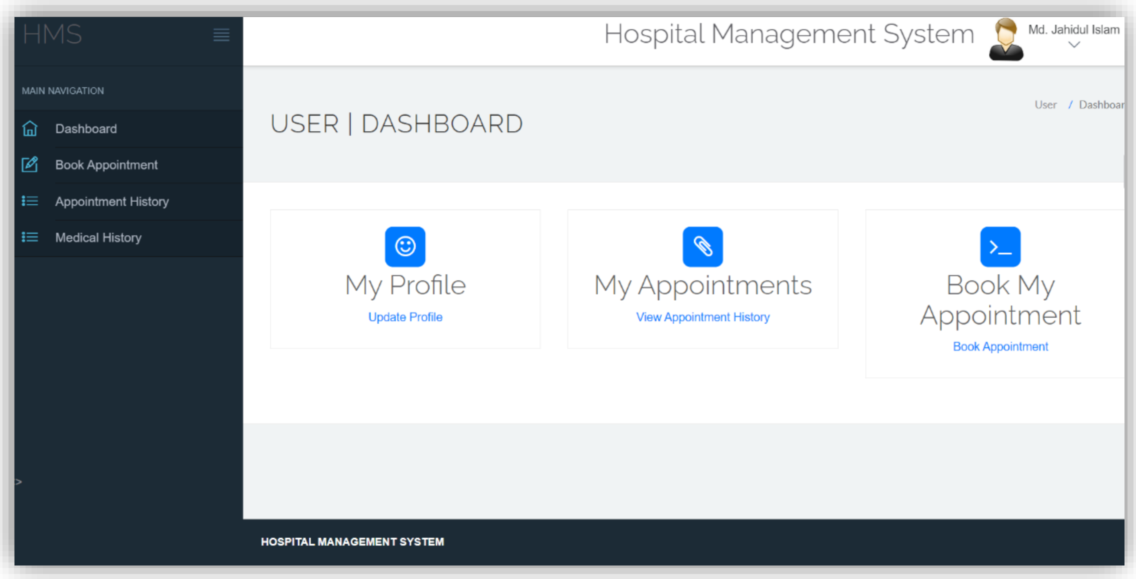
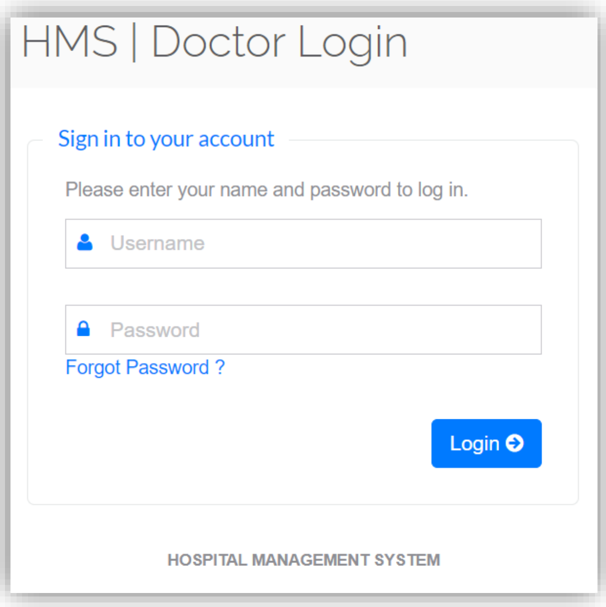


Fig: Patient | Dashboard

5.5 Doctor Login

This login page provides the facility for doctors to login and view the appointment status and his/her patients. Doctors don’t have registration option because hospital authority sets the doctors and distribute the username and password for the doctors. There has forgot password option if any doctor forgets his/her password he/she can reset the password by clicking forgot password option.



Please enter your name and password to log in.

Fig: Doctor login

5.6 Doctor Dashboard

After logged in a doctor he/she will get this page DOCTOR DASHBOARD. Doctors can update his profile by clicking My Profile. He/she can view the his/her appointment history and there has Patients option by clicking on this he/she can add and manage his/her patients.

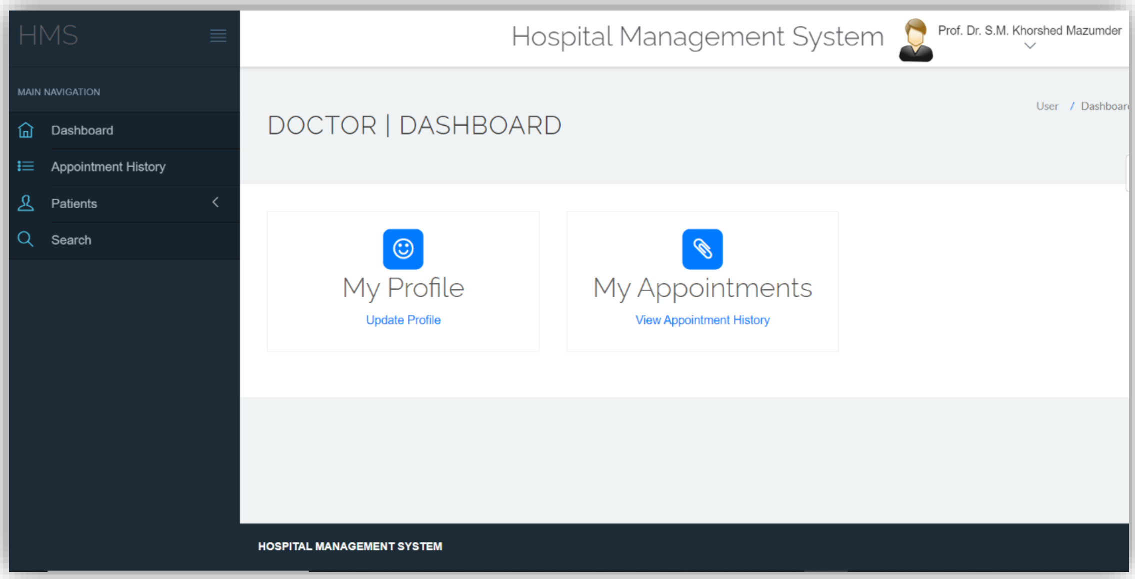


Fig: Doctor | Dashboard

5.7 Admin Login

This is admin login page. This page provides the facilities to log into the system.

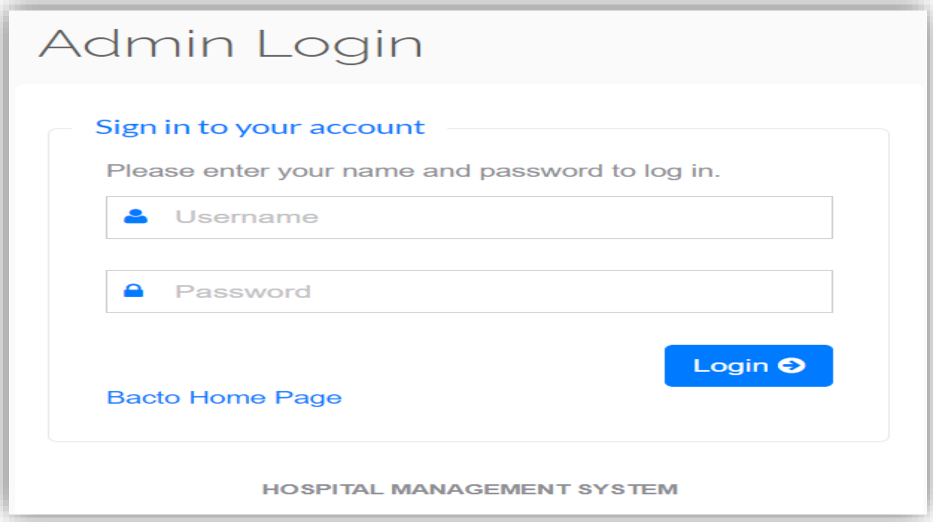

Admin Login
Sign in to your account
Please enter your name and password to log in.
Username
Password
Login
Bacto Home Page
HOSPITAL MANAGEMENT SYSTEM

Fig: Admin Login

5.8 Admin dashboard

Using this page admin can manage users, doctors, appointments, patients and admin can give feedback for queries. Admin can search the patients for checking his/her medical history. Furthermore, admin can view the session logs of the users as well as doctors. There has reports option. Admin can find the reports between dates and can analyze the patients reports for better outcome. System authority give the access to admins to do the following functions by providing him/her a specified admin username and password.

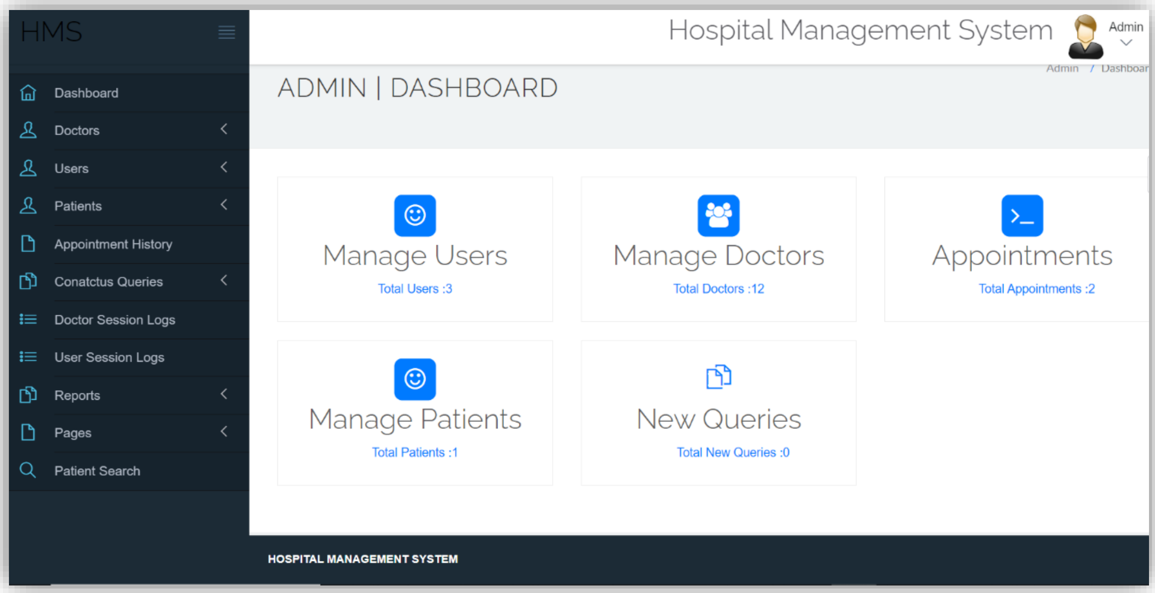


Fig: Admin | Dashboard

Chapter 6

Conclusion and Future Goals

In conclusion, the development of the Hospital Management System (HMS) project has been a significant step towards improving the efficiency, accuracy, and overall management of healthcare facilities. The HMS project has successfully addressed key challenges faced by traditional hospital settings, such as manual appointment scheduling, paper-based record-keeping, and inefficient data access. Through the implementation of the HMS, patients can now conveniently book appointments online, doctors can easily manage their schedules, and administrators can efficiently handle patient records and overall database management.

The project has demonstrated several important benefits. First, it has improved the accessibility and convenience for patients by allowing them to book appointments from anywhere at any time. This has led to reduced waiting times and increased patient satisfaction. Second, doctors have gained quick access to patient profiles and medical history, enabling them to provide personalized care and make informed decisions during consultations. Finally, administrators have experienced enhanced efficiency in managing patient records, doctor information, and appointment records, leading to improved overall hospital management.

there are several future goals that can further enhance the system and contribute to the continuous improvement of healthcare management. These goals encompass various aspects of the HMS, including functionality, scalability, usability, and integration with emerging technologies.

One future goal is to expand the functionality of the HMS by incorporating advanced features and modules. For example, integrating electronic health records (EHR) can provide a comprehensive view of a patient's medical history, enabling healthcare providers to make more informed decisions. The system can also include modules for inventory management, pharmacy management, and billing to streamline the overall healthcare process. By expanding the functionality, the HMS will become a comprehensive platform that covers various aspects of hospital management.

Another important goal is to enhance the scalability of the system. As healthcare facilities grow and patient volumes increase, the HMS should be able to handle the additional load without compromising performance. This can be achieved by optimizing the database structure, implementing caching mechanisms, and adopting cloud-based infrastructure to ensure scalability and high availability.

In conclusion, the future goals of the Hospital Management System project revolve around expanding functionality, enhancing scalability, improving usability, integrating emerging technologies, and ensuring compliance with regulations. By pursuing these goals, the HMS will continue to evolve as a comprehensive and efficient solution for hospital management, contributing to improved patient care, streamlined processes, and enhanced healthcare outcomes.

REFERENCES

1. https://www.w3schools.com/
2. https://developer.mozilla.org/
3. https://www.php.net/
4. https://dev.mysql.com/doc/
5. https://getbootstrap.com/docs/
6. https://stackoverflow.com/
7. https://github.com/
8. https://phptherightway.com/
9. https://www.phpjabbers.com/
10. https://www.codecourse.com/
11. https://www.codecourse.com/
12. https://laracasts.com/
13. https://www.sitepoint.com/
14. https://www.reddit.com/r/webdev/
15. https://getbootstrap.com/docs/
16. https://api.jquery.com/
17. https://html.com/
18. https://www.w3schools.com/css/
19. https://code.visualstudio.com/