

**DESIGN AND DEVELOPMENT OF WEB-BASED HOSPITAL MANAGEMENT
SYSTEM IN BANGLADESH**

BY

**Abu Junaeid Shoaib
ID: 183-15-2316**

AND

**Istak Ahmed
ID: 183-15-12037**

This Report Presented in Partial Fulfillment of the Requirements for the Degree of
Bachelor of Science in Computer Science and Engineering

Supervised By

Professor Dr. Md. Fokhray Hossain
Professor
Department of CSE
Daffodil International University



DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH
SEPTEMBER 2022

APPROVAL

This Project titled “**Design and Development of Web-based Hospital Management System in Bangladesh**”, submitted by Abu Junaid Shoaib (183-15-2316) and Istak Ahmed (183-15-12037) to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 13 September, 2022.

BOARD OF EXAMINERS



13.09.2022

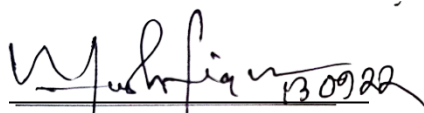
Dr. Sheak Rashed Haider Noori
Associate Professor and Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman



Sazzadur Ahmed (SZ)
Assistant Professor
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



13.09.22

Mushfiqur Rahman (MUR)
Senior Lecturer
Department of CSE
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



13.9.22

Dr. Md Sazzadur Rahman
Associate Professor
Institute of Information Technology
Jahangirnagar University

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Professor Dr. Md. Fokhray Hossain, Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Professor Dr. Md. Fokhray Hossain
Professor
Department of CSE
Daffodil International University

Submitted by:



(Abu Junaeid Shoaib)
ID: 183-15-2316
Department of CSE
Daffodil International University



(Istak Ahmed)
ID: 183-15-2316
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Professor Dr. Md. Fokhray Hossain, Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*FSIT*” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to Professor Dr Touhid Bhuiyan, Professor and Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

This project is intended to make a **Hospital Management System** that will improve hospital experiences for both patients and the hospital authorities. This system needs an internet connection to run. This system allows the user to Register and logs in to create an account from any place with an internet connection. After that, they will be able to do various kinds of tasks that are designed for them. This system also provides all the users with a unique user Id. Using tools such as PHP, Laravel, Java Script, HTML and CSS this system was built. Users can use this system to do things more efficiently and productively. The system can be entered using email and password. Two key modules in this system are (i) user (i. e. patients or doctors); and (ii) Administrator. The system maintains authentications in order to access the system. Administrator handles tasks like managing doctor information, patient information etc. The primary aim is to focus on every user who uses our service gets benefitted and make living easier. The major problem nowadays is the patient gets a report after consultation, many hospitals manage reports in their system but it's not available to the patient when he/she is outside. In this project, we going to provide the facility to store the report in the database and make it available from anywhere in the world.

TABLE OF CONTENTS

CONTENTS	PAGE
Board of examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
CHAPTER	
CHAPTER 1: Introduction	01-06
1.1 Background	01
1.2 Motivation	02
1.3 Problem Statement	03
1.4 Aim of the Project	03
1.5 Research Methodology	04
1.6 Proposed Solution	05
1.7 Report Layout	05
1.8 Conclusion	06
CHAPTER 2: Requirement Specification	07-27
2.1 Introduction	07
2.2 Business Process Modelling	07
2.3 Requirement Collection and Analysis	10

2.4 Use Case Modelling and Description	12
2.5 Logical Data Model	19
2.6 Design Requirement	21
2.6.1 Functional Requirement	22
2.6.2 Non-Functional Requirement	23
2.6.3 System Architecture	25
2.7 System Requirements	26
2.8 Conclusion	28
CHAPTER 3: Design Specification	29-36
3.1 Introduction	29
3.2 Front-End Design	29
3.3 Back-End Design	31
3.4 Interaction Design and User Experience (UX)	32
3.5 Implementation Requirement	35
3.5.1 VS Code	34
3.5.2 XAMPP	35
3.6 Conclusion	36
CHAPTER 4: System Development	37-40
4.1 Introduction	37
4.2 Core Module Coding Samples	37
4.3 Possible Problem Breakdowns	39
4.4 System Development life cycle	40
4.5 Conclusion	40

CHAPTER 5: Implementation and Testing	40-64
5.1 Introduction	41
5.2 Implementation of Database	41
5.2.1 Data-Table Name	43
5.2.2 Datatype Architecture	43
5.2.3 Table Architecture	44
5.2.4 MySQL Database View	46
5.2.5 Data-Table Structure	47
5.2.6 Routes	47
5.3 Implementation of Front-End Design	51
5.4 Testing Implementation	59
5.4.1 Methodology of Testing	60
5.4.2 Testing for Functionality	60
5.4.3 System Responsiveness Testing	61
5.5 Test Results and Reports	62
5.5.1 Results	63
5.5.2 Testing Scope	63
5.6 Conclusion	64
CHAPTER 6: Importance of the Hospital Management System In Bangladesh	65-67
6.1 Introduction	65
6.2 Necessity of Web-based Hospital Management System	65

6.3 Compliance and quality	66
6.4 Processing Time and Outcome	66
6.5 Opportunities and Obstacle	67
6.6 Conclusion	67
CHAPTER 7: Impact on Society, Environment and Sustainability	68-75
7.1 Introduction	68
7.2 Impact on Society	68
7.3 Impact on environment	70
7.4 Ethical Aspects	72
7.5 Sustainability plan	74
7.6 Conclusion	75
CHAPTER 8: Conclusion	76-77
8.1 Conclusion	76
8.2 Further Suggested Work	77
REFERENCE	78-80
APPENDICES	
APPENDIX 'A'	A1-A3
APPENDIX 'B'	B1-B7
APPENDIX 'C'	C1-C2

LIST OF FIGURES	PAGE NO
Figure 2.2.1: Business Process life cycle	08
Figure 2.2.2: Outpatient Business Process Model	09
Figure 2.2.3: Inpatient Business Process Model	10
Figure 2.4.1: Use case Diagram of Hospital Management System	13
Figure 2.5.1: Logical Data Model of Hospital Management System	20
Figure 2.5.2: E-R diagram of Hospital Management System	21
Figure 2.6.1: System Architecture	25
Figure 3.2.1: Front-End Architecture	30
Figure 3.3.1: Back-End Development	31
Figure 3.3.2: Back-End Design	32
Figure 3.4.1: UX Site Mapping	34
Figure 5.2.1: Database Table Name	43
Figure 5.2.2: Database Table Architecture	45
Figure 5.2.3: My SQL Database View	46
Figure 5.2.4: Database Table (users)	47
Figure 5.3.1: Hospital Management Home Page	54
Figure 5.3.2: User Registration Page	55
Figure 5.3.3: User validation Page	55
Figure 5.3.4: Login & After login page and feature	57
Figure 5.3.5: Admin panel	58
Figure 5.3.6: Doctor panel	59
Figure 5.4.1: Responsiveness Testing	62

LIST OF TABLES

TABLES	PAGE NO
Table 2.1 Use case description	14
Table 2.2: System Requirements Version 3x	26
Table 2.3: System Requirements Version 2x	27
Table 2.4: System Requirements Version 3x	27
Table 5.1: Hardware Requirements of the system	42
Table 5.2: Hardware Requirements of the system	42
Table 5.3: Testing Results of the system	63

Chapter 1

Introduction

1.1 Background

Rapidly rising information technology utilization in patient care increased documentation, coding and billing, and management are characteristics of healthcare, particularly in industrialized nations. Globalization of health information technology is raising the delivery of health services with more efficiency, fewer medical errors, and higher standards of care, by supplying physicians and patients with greater information (Pollak and Lorch, 2007:4) The acquisition, management, and utilization of information is the function's overarching purpose. Information to enhance medical services, effectiveness, governance, and processes for management and assistance.

The importance of healthcare to both individuals and governments, as well as its rising economic expenses, has made healthcare a significant topic of study for academics in business and other fields. Information systems (IS) have a lot to offer in terms of controlling healthcare expenses and enhancing services (Kolodner et al. 2008: 394). Additionally, according to Piontek et al. (2010: 618) healthcare affects how well people live their lives and contribute to society. Serious effects from healthcare errors can impair one's capacity for engaging in social and productive activities. Recent publications emphasize the seriousness of adverse hospital incidents and the risks they bring to people and the general public.

Health information systems acknowledge that new technological advancements, such as the use of the internet and other communication methods, provide enormous promise in the flow of information between the providers and recipients on the supply and management of healthcare services.

1.2 Motivation

We don't frequently take appointments to the hospitals unless we would like to although this visit doesn't consistently provide us with a pleasant experience. We encounter numerous problems there. There is a programmed and automated hospital management system but they're not providing us with any functionality.

You may find hospital management systems in a lot of different hospitals and they have many great features and all but they don't usually allow the patient.

Parallel to the patients, the doctors do not have access to their patient's information while they're not at the hospital. If there were any applications that would keep the patients, near their information like test reports, prescriptions and other essential materials that would be calming. If patients were able to go through or browse all the doctors available and ask for an appointment with his or her desired doctor. A doctor will also have similar experiences and benefits. This will reduce a lot of manual work like paper works and make things comfortable and easy for everyone associated with it.

I have got some personal experience visiting the hospital. In some hospitals, you have to go there physically or have someone bring you the reports. This is quite a hassle if you ask me and you have to wait in the queue for the report for quite a long time. When someone is really sick or it is an emergency every second counts.

I have gathered some experience from others by asking and questioning them and observing them for some time. People experience their worst nightmares when they need to travel to the hospital. Everyone avoids going to the hospital unless it is really necessary.

People become angry because they have few other options. This provided me the inspiration I needed to construct something for them and to provide assistance when they most needed it. I have no doubt that they will benefit from this product.

1.3 Problem Statement

In Bangladesh, there are a lot of clinics and hospitals that are following the file-based system, where people have to go to the hospital and to make an appointment. Every hospital tends to gather patients' information like name, age, medical information, transactions between hospital and patient etc. All these goes into files. These files are stored into lockers and shelves and get neglected for years and incase of fire they can be easily destroyed.

The traditional file-based system is very lengthy and time consuming. If a file gets lost it would take hours to find that file. Finding doctors is a big hustle. Normally a pain to find a specific doctor and seek treatment from them unlike WHMS, where doctors and labelled with their specialty and with their details.

The traditional hospital system takes a lot of time in queues for payment or to make an appointment. There are better ways to do this in this modern world.

1.4 Aim of the Project

Due to reasons including a rising population and higher demand for a quality of life, medical services are in constant demand in Bangladesh, as they are in many other nations. Our primary goal is to create a Web-based hospital management system that incorporates all of the key operational components of today's multi-speciality hospitals. Specific objectives of this project Hospital Management System are:

1. Design a better system for hospitals, Doctors and Patients.
2. Improved patient care.
3. Better patient management (Registration, Scheduling and long-term service).
4. Better communication and coordination among the different departments.
5. Give the senior management one point of responsibility.
6. Online payment system for less hassle.

7. Easy to use User Interface (UI).
8. Deliver reliable and quick service on a real time basis
9. Manage medical and physical check-ups services.
10. Provide the standard coverage of records that transacts.
11. Eliminate the chances of any kind of pilferage.
12. Enable the growth of the Hospital.
13. Data security and correct data retrieval.
14. Reduce scope of errors.
15. Reduce the documentation work.
16. To save time and money.
17. Reduce Hospital Operating costs and easily manageable.

1.5 Research Methodology

This essay looks at the creation of web-based hospital management systems. Investigating the insights of a web development in depth is necessary, therefore interpretative research is employed and for qualitative methods interviewing is necessary in chronological order.

In order to get a variety of viewpoints on this research field, various consulting firms and developing web-based information systems for other organizations were chosen as a case study, utilizing social interaction with participants in light of the study's background.

For the creation of the application, we followed the traditional process. The software or system development life cycle has long been used in software engineering (Lee, Sheridan-Smith, O'Neill, Leaney Sandrasegaran, and Markovits, 2003). A detailed, step-by-step process for developing an application system that satisfies business demands or organizational criteria is the primary definition of the SDLC, according to various research (McManus, 2003; Rob, 2004). It's interesting

to note that other studies suggest that the SDLC is not only a process model but also "a conceptual model," particularly for the management team to plan and develop information systems like transaction information systems, management information systems, and decision support systems (Hoffer, George, and Valacich, 2011).

In essence, the planning stage is where the proposed system must first be identified, prioritized, and arranged in accordance with the user requirements. The Analysis stage will next thoroughly review and arrange these requirements to present solutions. During the design phase, the requirements will be transformed into logical and physical system specifications that include in-depth descriptions of the technology's user features and distinct functions. At the implementation stage, the system will then be fully built and tested to ensure that the requirements are satisfied. Not to mention, the system will soon undergo enhancements and improvements throughout the Maintenance stage.

1.6 Proposed Solution

In Web-based Hospital Management System everything is computerized and digitized. People doesn't have to go outside and spend money to go to the hospital and make the appointment, they can do it with smartphone, laptop or iPad or any device that can be used to access the internet.

And most of the work that the hospital has to do manually will be done automatically. The doctors and admins can see the list of patients and doctors with just a simple click and not go for the hustle to go through every file.

Both patients and the hospital will benefit from the web-based hospital management system. Because automating a lot of the work that has been done manually for years saves a lot of time and money.

1.7 Report Layout

The report's layout provides an overview of each chapter. Below is a quick overview of each chapter:

- **Chapter 1:** Describes an introduction of the Web-Based Hospital Management System, Background, problem statement, aim of project, research methodology and the Report layout.
- **Chapter 2:** Describes the Requirement Specification, Introduction Business process modelling, Requirement Collection and Analysis, Use Case Modelling and Description, Logical Data Model, Design Requirement, System Requirements, Conclusion
- **Chapter 3:** Describes the Design Specification, Introduction, Front-End Design, Back-End Design, Interaction Design and User Experience (UX), Implementation Requirement, Conclusion.
- **Chapter 4:** Describes the Implementation and Testing, Introduction, Implementation of Database, Implementation of Front-End Design, Testing Implementation, Test Results and Reports, Conclusion.
- **Chapter 5:** Describes the Importance of the Hospital Management System
- In Bangladesh, Introduction, Necessity of Web-based Hospital Management System, Compliance and quality, Processing Time and Outcome, Opportunities and Obstacle and conclusion.
- **Chapter 6:** Describes Impact on Society, Environment and Sustainability, Introduction, Impact on Society, Impact on environment, Ethical Aspects, Sustainability plan, Conclusion
- **Chapter 7:** Describes the Conclusion, Further Suggested Work

1.8 Conclusion

The whole concept of a hospital management system is for the betterment of people's health.

Web-based hospital management systems will be very convenient for all the doctors, patients and the hospital itself. Hospitals can adapt this web-based method for managing the staff and finance very easily. Since the internet is very cheap and almost everyone has internet access. This web-based hospital management system will be useful and convenient to everyone who uses it.

CHAPTER 2: Requirement Specification

2.1 Introduction

This section explains the different hardware and software requirements and the tools used by the developer while working on the project. Additionally, it contains the specifications needed by the client to develop and use the system. This decision also pertains to the chosen operating system. The alternatives are subjected to the mandatory screening criteria, which combines the explicit criteria with certain implied criteria that the alternative.

Requirement specification is very important for building a project. This will have all the criteria and resources that will need to build the software or the system. It is very important to know the requirements if a developer wants to finish the project.

By watching the requirement specification of a project a developer can find out the cost and the software, hardware and skills needed to get started.

2.2 Business Process Modeling

A business process make reference to collection of linked activities that produce any kind of service or product for a client. Normally, a business is represented as a flowchart that depicts the life cycle of the process.

The procedure' significance is not limited to a single industry. Business owners often carry out business process modelling to optimize business performance, management and operations. As a result, the testing and development team can analyze the workflow represented by business process that can lead to greater efficiency in the associated activities.

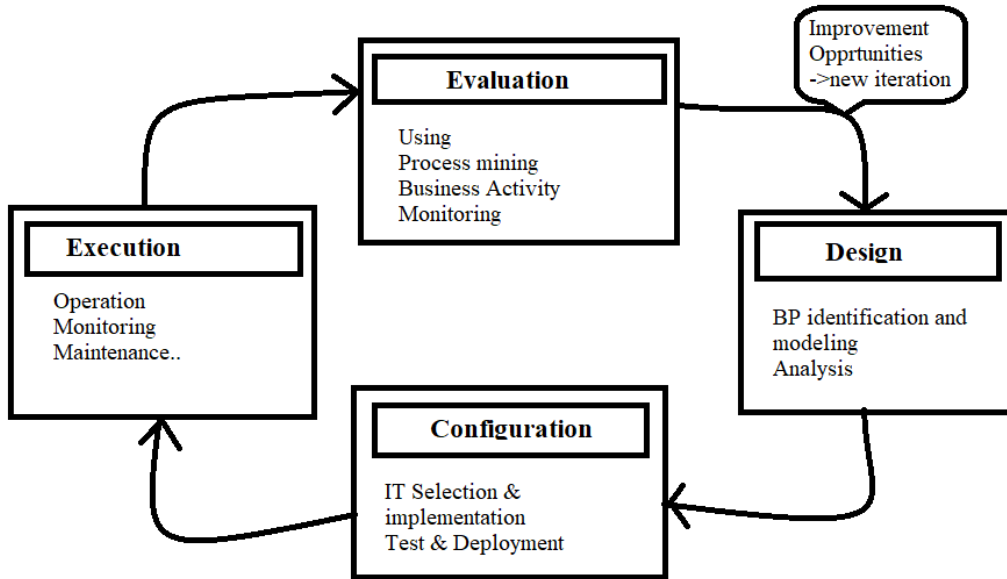


Figure 2.2.1: Business Process life cycle

Due to the nature of the services provided, health institutions base their decision-making procedures in this area on accomplishing their objective of offering high-quality medical care. The definition of healthcare processes, the multidisciplinary nature of healthcare, the variety and flexibility of the activities involved in healthcare processes, the requirement for connectivity between various information systems, and the concurrent updating of scientific knowledge in healthcare systems are the main challenges facing healthcare business process modelling.

Outpatient BPM:

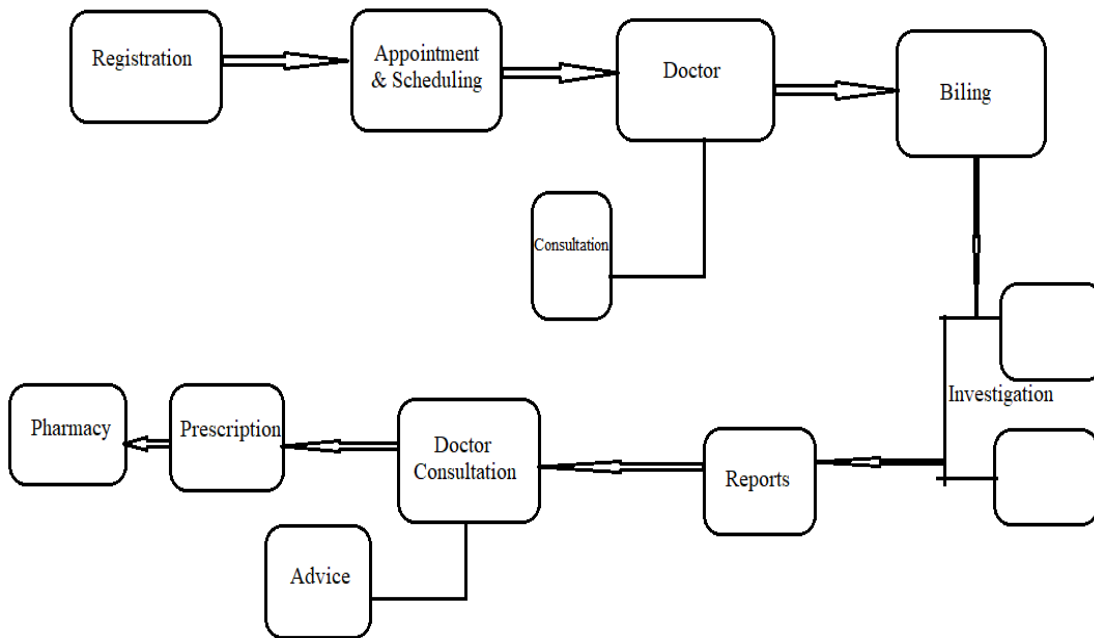


Figure 2.2.2: Outpatient Business Process Model

Outpatient means who doesn't need to stay in the hospital at all. They visit a doctor takes advice and leave. On the other hand, Inpatient requires stay in the hospital for quite a time until they get well, usually they are the severe one.

Inpatient BPM:

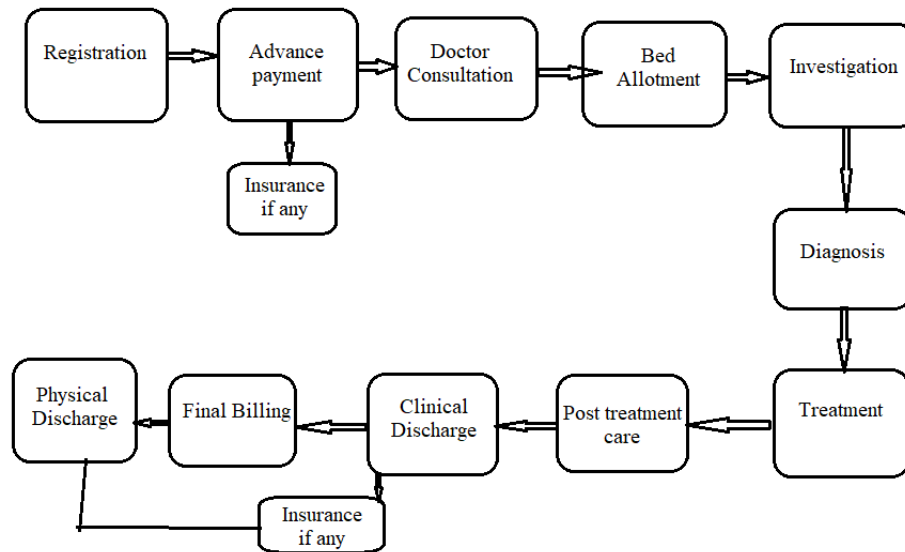


Figure 2.2.3: Inpatient Business Process Model

2.3 Requirement Collection and Analysis

The manual hospital administration system includes patient registration, keeping a record of their information in a file, and keeping track of the patient's hospital bills. The manual hospital system includes patient registration, documenting patient data as an entry or record in a file, and monitoring the patients' medical bills

Managing the entire hospital manually is tremendously difficult. Finding a single user record takes way too long, and managing several records sequentially is very difficult. These systems have been plagued by a number of problems. The following list of concerns is just a sample:

Lack of immediate retrieval: The user must search through numerous record books and this process takes time and effort in order to get specific information, such as the patient's information or history.

Lack of immediate information storage: It requires time and effort to store the information that is created by numerous transactions in the proper location.

Manual computation error-prone: Manual calculations are time-consuming, prone to error, and may produce inaccurate information.

Preparation of accurate and prompt reports: Gathering data from numerous record books is a challenging undertaking.

Records deterioration: Over time, the files used to store information start to wear out and become less reliable.

By creating a computerized system, the above-mentioned issues may be reduced, but doing so will also impede the computerization of the hospital management process. We anticipate that the computerized hospital management system will be useful and that it will improve administration and control, streamline operations, improve patient care while strictly controlling costs, and improve facilities. The system may also be strong, adaptable, and simple to operate.

The proposed Web-Based Hospital Management System is accustomed to take the data from the patients and then store it for further use. The main objective of the Hospital Management System is to accurately treat as well as decrease overtime payment. There are various features included in the HMS. Some of the system functions include Registration, Report generation, Patient profile out and more. In this project patient agent, doctor agent, admin agent and environment agent are designed to collect respective user requirements for hospital management system.

It is expected that a web-based Hospital Management System will provide feasible solution to the issues that arise both during and after the deployment of the Hospital Management

System. The reason is that a system like this may adapt to new requirements as they are recognized. Additionally, it will develop over time and gather knowledge to better grasp the emergent requirements that will unavoidably arise throughout system operation. In-depth research is being done on the application of suitable learning and sensing techniques.

2.4 Use Case Modeling and Description

A **use-case model** depicts how different user groups engage with the system to address a challenge. As a result, it explains the users' objectives, how they interact with the system, and the actions the system must take to achieve these programs or policies.

Various model components make up a use-case model. Use cases, actors, and their interactions are the three key components of a model. The use-case model must have the following fundamental model components.

Actor

A model component for every actor. The actor's name and a brief bio are listed under properties. For further details, refer to Concept: Actor.

Use Case

Each use case is represented by a model element. The use case name and use case specification are properties. For more details, see Artifact: Use Case and Concept: Use Case.

Associations

The ties between the actors are specified by association, and the use cases they participate in are characterized by associations. This kind of connection is referred to as a "communicates-association."

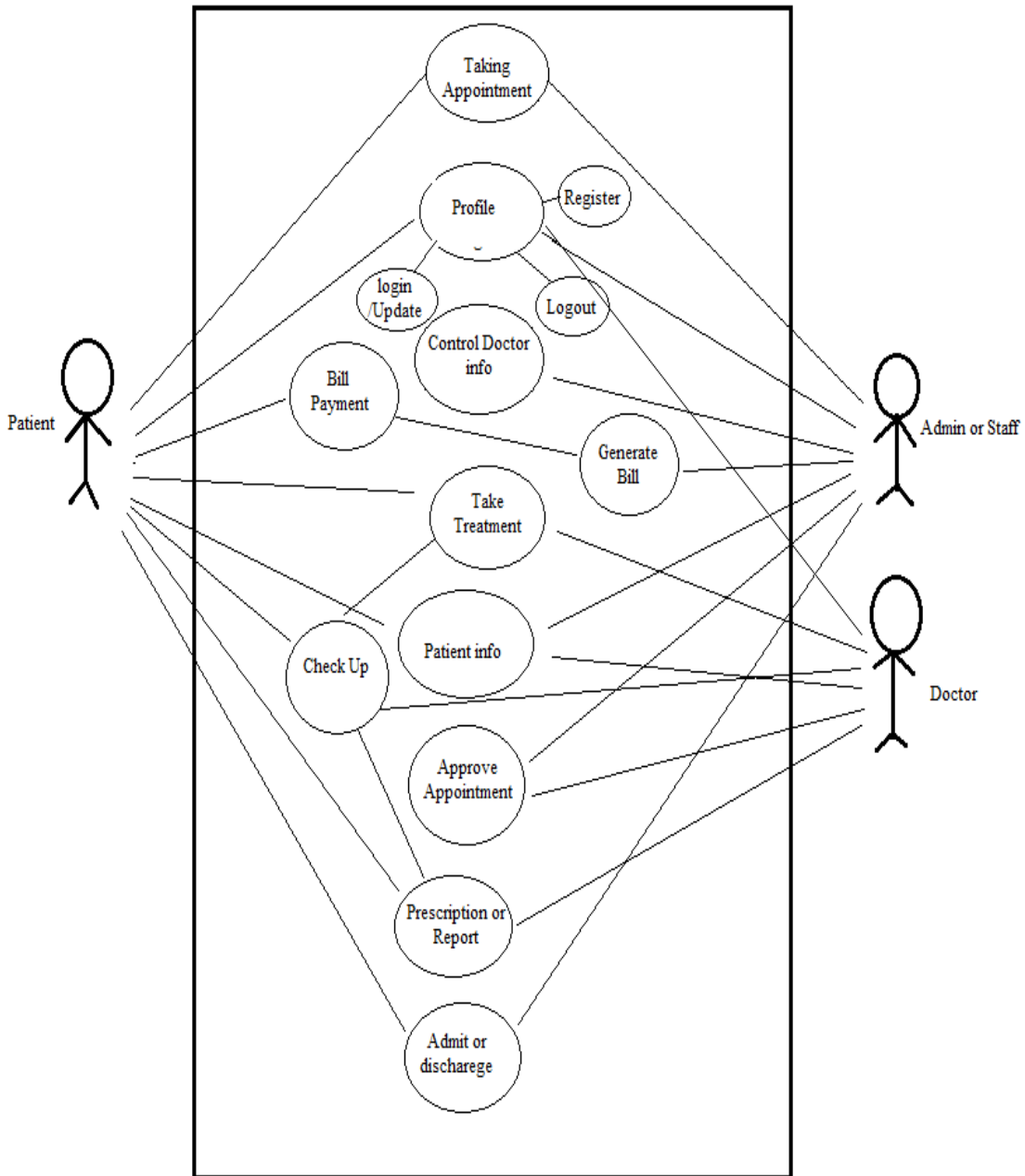


Figure 2.4.1: Use case Diagram of Hospital Management System

Table 2.1 Use case description

Use Case: Register

Actor: Doctor, Admin or Hospital Staff and Patient

Type: Primary

Description: If the user wants to access this application, they must provide pertinent information to Register.

Uses: To login

Extended by: None

Extends: None

Use Case: Log In

Actor: Doctor, Admin or Hospital Staff and Patient

Type: Primary

Description: If the user wants to access this application, they must provide valid Email, Password and profession to login.

Uses: Profile

Extended by: None

Extends: None

Use Case: Profile

Actor: Doctor, Admin or Hospital Staff and Patient

Type: Primary

Description: All users have their own profile. They can update and delete their profile.

Uses: Doctor, Admin or Hospital Staff and Patient

Extended by: None

Extends: Update, Delete and Logout

Use Case: Taking Appointment

Actor: Anyone who uses the system

Type: Primary

Description: Anyone with an account can take Appointment

Uses: Applies for doctor appointment

Extended by: None

Extends: Update Appointment

Use Case: Control Doctor Info

Actor: Admin or Staff

Type: Primary

Description: Admin can Update, Add or delete doctor information

Type: Primary

Description: Admin can Update, Add or delete doctor information

Uses: Changes or Adds up Doctor information

Extended by: none

Extends: Update, Add or delete

Use Case: Generate Bill

Actor: Admin or Staff

Type: Primary

Description: Admin can generate bill for the payment

Uses: Payment

Extended by: none

Extends: None

Use Case: Bill Payment

Actor: User or Patient

Type: Primary

Description: User can pay bill for the payment through with various payment option

Uses: Payment

Extended by: none

Extends: None

Use Case: Patient info

Actor: User or Patient, admin and Doctor

Type: Primary

Description: User can view their own info as well as the doctor and admin.

Uses: View Patient info

Extended by: none

Extends: None

Use Case: Appointment

Actor: User or Patient, admin and doctor

Type: Primary

Description: User can view their appointment progress as well as the doctor and the admin can control the appointment status.

Uses: Appointment

Extended by: none

Extends: Approve, Cancel

Use Case: Prescription or report

Actor: User or Patient, doctor

Type: Primary

Description: User can view the prescription or report and doctor can prescribe or

View report

Uses: Report and Prescription

Extended by: None

Extends: None

Use Case: Admit or Discharge

Actor: User or Patient, Admin

Type: Primary

Description: User can ask admin for admission or the admin can control the admission and discharge process.

Uses: Admit or discharge

Extended by: None

Extends: None

2.5 Logical Data Model

The structure of the data and its relationships are referred to as a logical data model. The details of how the data will be used are kept apart from the actual database. The used data is designed using the logical data model as a reference. The logical data model builds upon the core ideas of conceptual data modeling by including more specifics.

A logical data model is composed of three fundamental parts:

Entities: Each entity is a collection of elements, individuals, or ideas important to a firm.

Relationship: Every relationship is an association between two of the aforementioned objects.

Attributes: Each attribute is a defining feature, trait, or any other detail that helps to further describe an object.

These logical data model elements are each given a name and a written definition. These serve to continuously explain the information requirements as well as the business regulations. However, the aforementioned elements are simply intended to describe business requirements. They don't care how those business requirements are handled, put into practice, or stored.

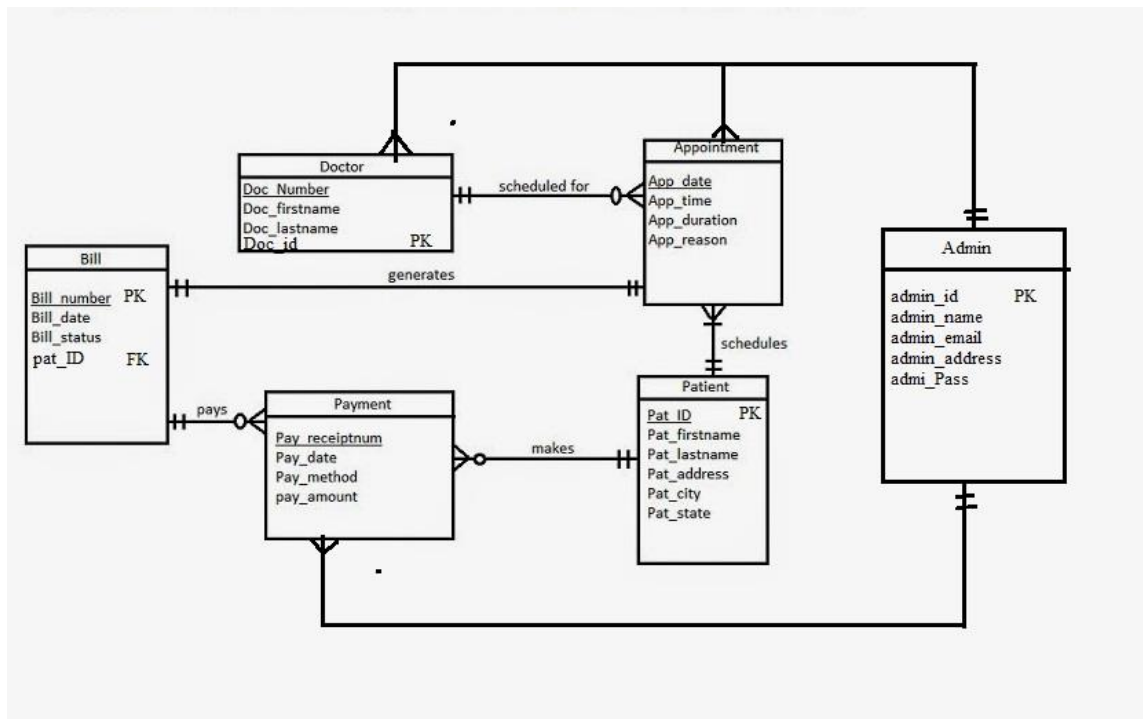


Figure 2.5.1: Logical Data Model of Hospital Management System

2.5.1 E-R Diagram

The Hospital Management database has some couple of number of tables. There are Login, Roles, User, Permission, Patient, Hospitals, medicine, Doctors etc.

In the Login there are #login_id, login_role_id, login_username, user_password.

In the User has #user_id, user_type, user_name, user_mobile, user_email, user_address.

In the Doctor there is #doctor_id, doc_name, doc_mobile, doc_address, doc_email, pass.

In the patient there is #patient_name, pat_id, pat_mobile, pat_email, pat_address etc.

On the other hand there is hospital which contains the admin and staff who controls over system who also carries some id, name, email, password.

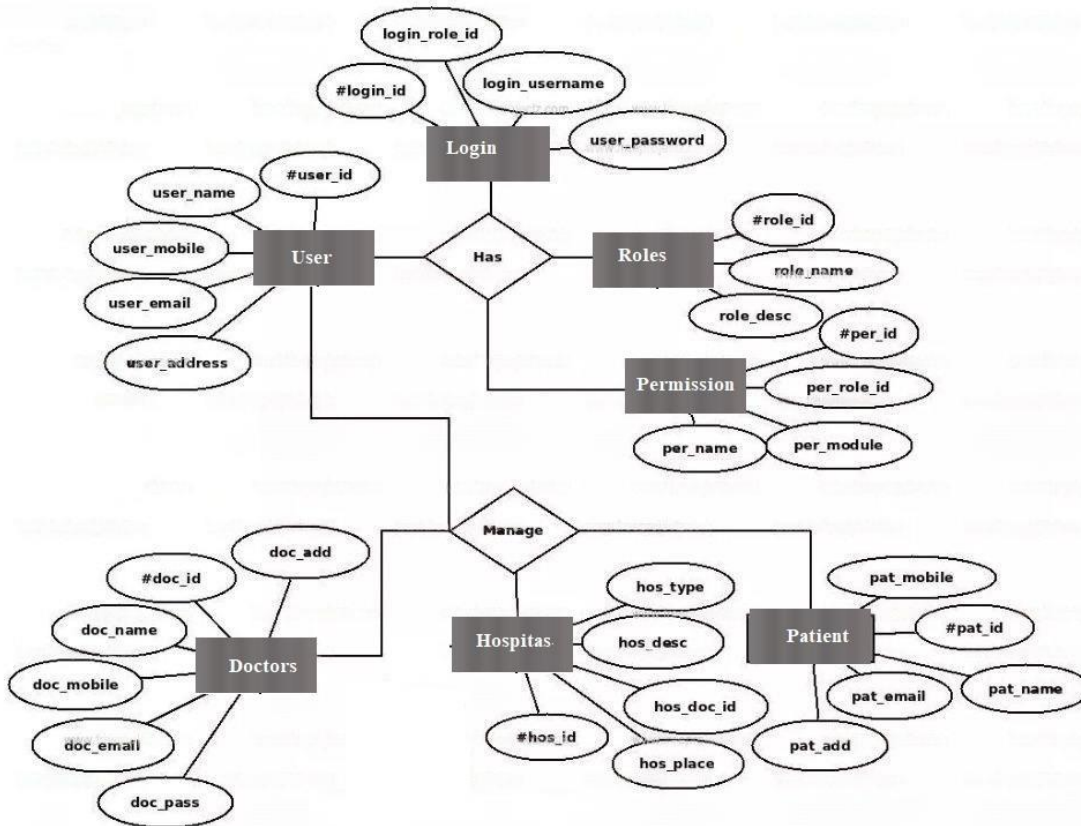


Figure 2.5.2: E-R diagram of Hospital Management System

2.6 Design Requirement

Design requirements are based on the want and needs of the project. Once the wants and needs have been identified it can be converted to engineering requirements. These engineering requirements become the part of the design requirements.

Basically, design requirements are the functional attributes that can be converted into design features. It explains the goals, purpose and scope of a project. They also provide guidelines for designing projects.

Design requirements can be divided into two categories: functional requirements and non-functional needs.

2.6.1 Functional Requirements

The functional requirements of the hospital management system, which includes registration, report generation, check-out, and database processes, contain numerous software needs specifications.

I. Registration Process of SRS (Software Requirements Specification)

- Adding Patients: Users may register in the hospital management system as patients.
- Giving each patient a unique ID: The HMS enables administrators to give each patient a unique ID before adding them to the patient's record sheet. The patients can use the ID both inside and outside of the facility..

II. Check Out of SRS:

- When a patient checks out of the hospital, the staff in the ward's administration area can remove their patient ID from the database.
- The staff in the ward's administrative section can add the vacant bed to the list of beds that are available.

III. Report Generation of SRS:

- Patient Information: The hospital management system creates a report on each patient with information such as the patient's name, phone number, address, the name of the doctor, bed number, who is assigned to them, and more.
- Bed Availability: The hospital management system also assists in producing reports on data such as bed number vacant or occupied, bed availability, ward name, and more.

IV. Database of SRS:

- Patient Data Required: Each patient must provide certain information, such as their first and last names, phone number, personal health number, address, postal code, city, country, and patient ID number.

- Updating patient information: The hospital management system enables users to update patient information according to the required information contained.

2.6.2 Non-Functional Requirements

There are a lot of software requirements specifications involved in the non-functional requirements of the Hospital Management System, which comprehend various process, namely Security, Maintainability, Performance and Reliability.

I. Security:

- Patient identification is required by the system, which the patient must do on the phone.
- Users who wish to access the system must have a Logon ID and password.
- The only person who can synchronize and carry out database adjustments, such as inserts, updates, and deletions, is the administrator.
- Front Desk Staff Rights: Front desk staff members have the ability to access any data in the hospital management system and add new patient records, but they do not have the authority to change any data.
- Administrator privileges: The administrator has the ability to see and edit any data in the hospital management system.

II. Performance:

- Response Time: After verifying the patient's details, the system sends an acknowledgement in just one second.
- User-Interface: Within five seconds, the user interface acknowledges.

III. Maintainability:

- Backup: The technology effectively provides data backup.

- Errors: The system will keep a record of each error and track them all.

IV. **Reliability:**

- System accessibility: The system is always accessible.

2.6.3 System Architecture

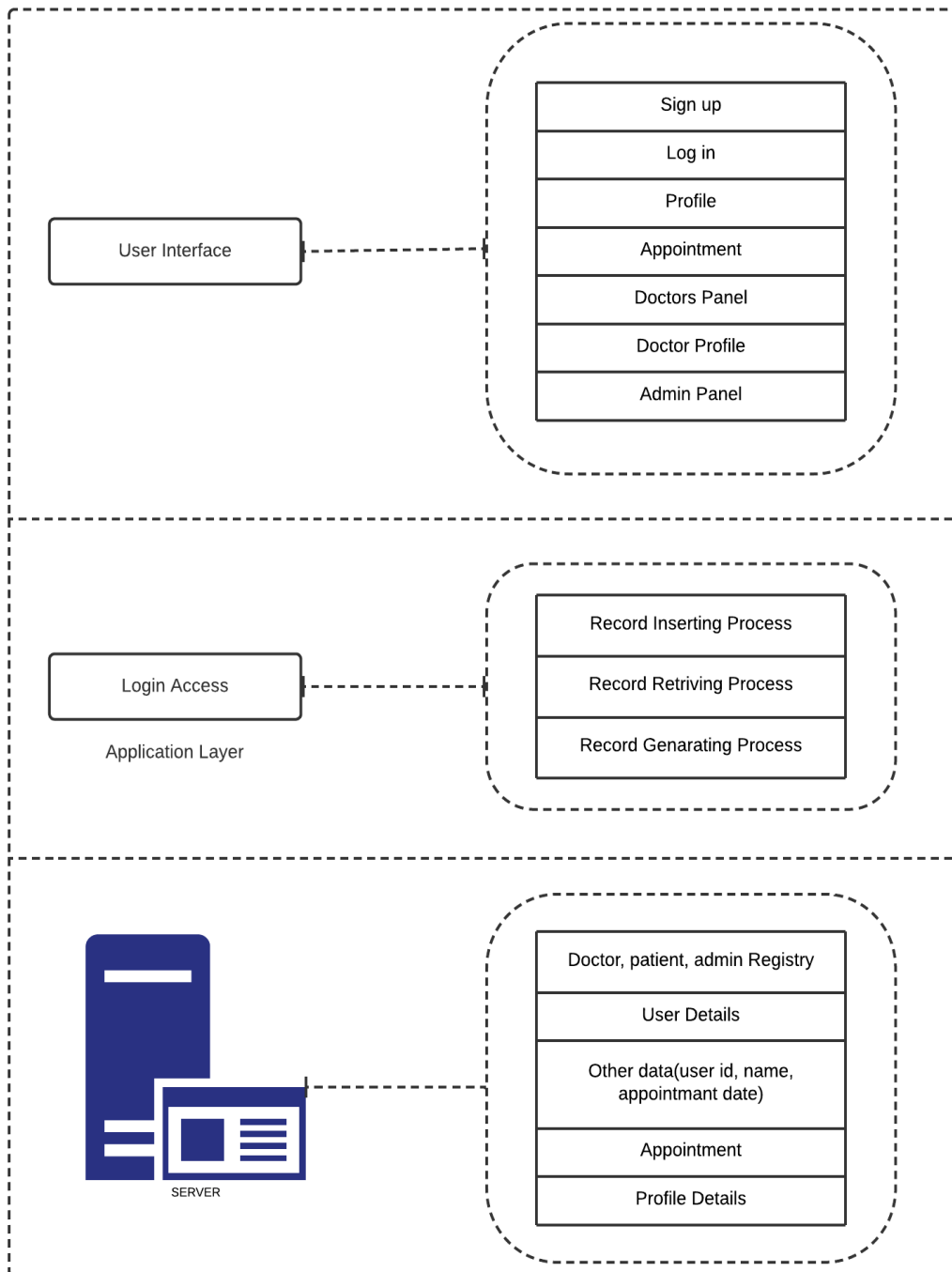


Figure 2.6.1: System Architecture

2.7 System Requirements

Table 2.2: System Requirements Version 3x

Criterion	Description
OS version	Microsoft® Windows® 7/8/10 (32-bit or 64-bit) Mac® OS X® 10.10 (Yosemite) or higher, up to 10.13 (macOS High Sierra) GNOME or KDE desktop Linux (64 bit capable of running 32-bit applications)
RAM	3 GB RAM minimum, 8 GB RAM recommended; plus
Disk space	2 GB of available disk space minimum, 4 GB recommended
PHP	Version: 8.0.8 (Recommended)
Screen resolution	1280×800 minimum screen resolution

Table 2.3: System Requirements Version 2x

Criterion	Description
OS version	Windows 7 or later Mac OS X 10.9.5 or later GNOME or KDE desktop Linux
RAM	8 GB RAM Recommended
Disk space	1.5 GB disk space Required
PHP	Version: 8.0.8 (Recommended)
Screen resolution	1280×800 minimum screen resolution

Table 2.4: System Requirements Version 3x

Criterion	Description
OS version	Mac OS X 10.8.5 or later GNOME, KDE or Unity desktop on Ubuntu or Fedora or GNU/Linux Debian
RAM	3GB RAM minimum, 4 GB recommended
Disk space	500 MB Disk Space
PHP	Version: 8.0.8 (Recommended)
Screen resolution	1280×800 minimum screen resolution

2.8 Conclusion

System requirements are one of the pillars for constructing a successful project, to sum up. System requirements are broad and specific statements that the developer makes in order to be able to meet their needs. The developer should have a list of the precise specifications they make. The price increases with the rigor and number of the requirements. The consumer must constantly strike a balance between having enough needs and budget-friendly requirements. Products may be affected by requirements that flow from a top organizational unit all the way down to the consumer and then to the producer. The system requirements' thoroughness and completeness are crucial because of this.

Without system requirements, both the project and the business will fail. A company will have a higher chance of success if the customer has system specifications that meet their needs. They will not only succeed, but also have a better possibility of making time and money savings. System requirements should be the foundation of any project and one of the first things to be established.

CHAPTER 3:

Design Specification

3.1 Introduction

A design specification is a thorough document that outlines every aspect of a process or product. It is, generally speaking, a combination of front-end and back-end user interface.. As the name implies, a design specification is a document that outlines the specifications for a certain project's design. The paper, which the client created after thorough investigation, has incredibly specific specifications for the design that must be strictly followed through to completion.

Design specifications are a crucial component of the construction documentation because trenchless building is done underground. To create a finished product that satisfies the client's needs, a contractor must follow the design guidelines and accepted best practices.

In the event that there are any design-related problems during the construction process, the document also protects the contractor from any liability.

3.2 Front-end Design

A computer or other device's front end is its graphical user interface (UI), which makes it easier to operate. The process of frontend design entails writing the presentational JavaScript, HTML, and CSS code that make up a user interface. A front-end designer, then, is a generalist with a solid command of UX concepts, a keen sense of aesthetics, and knowledge of HTML, CSS, and some JavaScript.

There are four areas that needed to give adequate consideration. These are:

- Fluid, grid-based layout.
- Consistency in design approach.
- The use of reusable components.
- Considering states

Front-End Architecture:

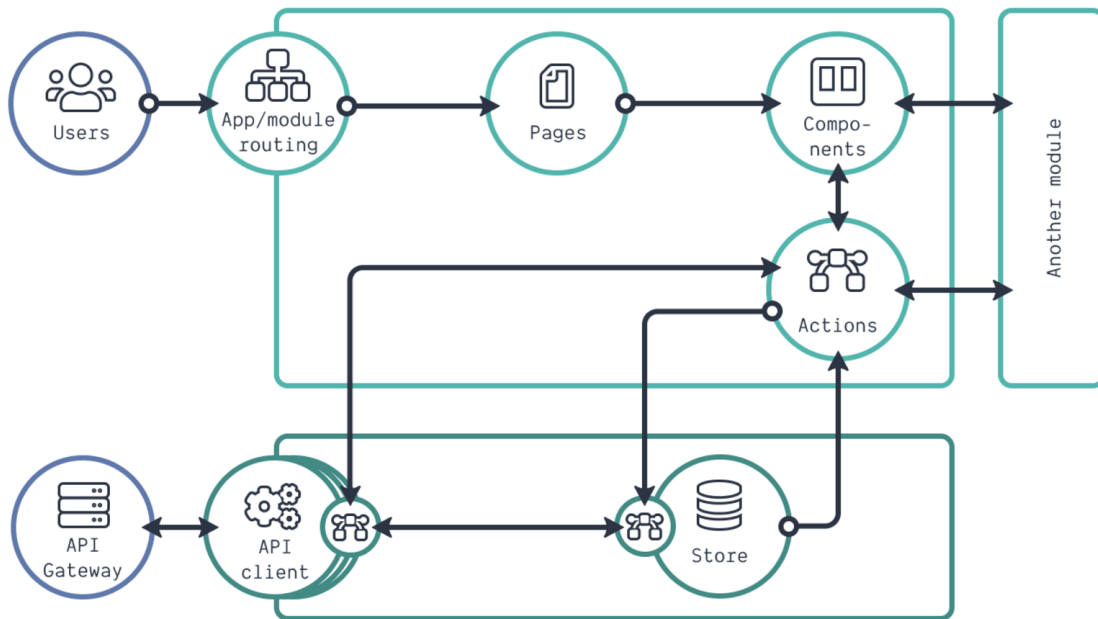


Figure 3.2.1: Front-End Architecture

Modern front-end applications are expected to handle a growing amount of the labor-intensive tasks. Bugs are arising more often as a result of this increased complexity. Since users interact with the front-end, we require an architecture that is dependable, scalable, and both. I currently favor a modular, domain-driven architecture. Although my vision may change, for the time being this is my chosen course of action.

The app routing ensures that users that interact with our application are directed to the appropriate module. Each module is included entirely. However, there will be some coupling because users want to utilize a single application, not a number of small ones. These characteristics or business logic are coupled together. Several characteristics can be shared between modules. This reasoning can be added to the application layer. This indicates that there is an interaction option for each module with the application layer. A configuration that necessitates using the client-side API to connect to our back-end, or API gateway, is an excellent example.

3.3 Back-End Design

The part of the website we don't see is called the backend (or "server-side"). Data storage, data organization, and client-side functionality are all its responsibilities. When the frontend and backend are in communication, data is sent and received for a web page to be displayed. The server-side takes request from the browser side whenever you complete a contact form, enter a website address or conduct a transaction (any user input on the client-side), and The frontend code that the browser can comprehend and display is returned by the server as data.

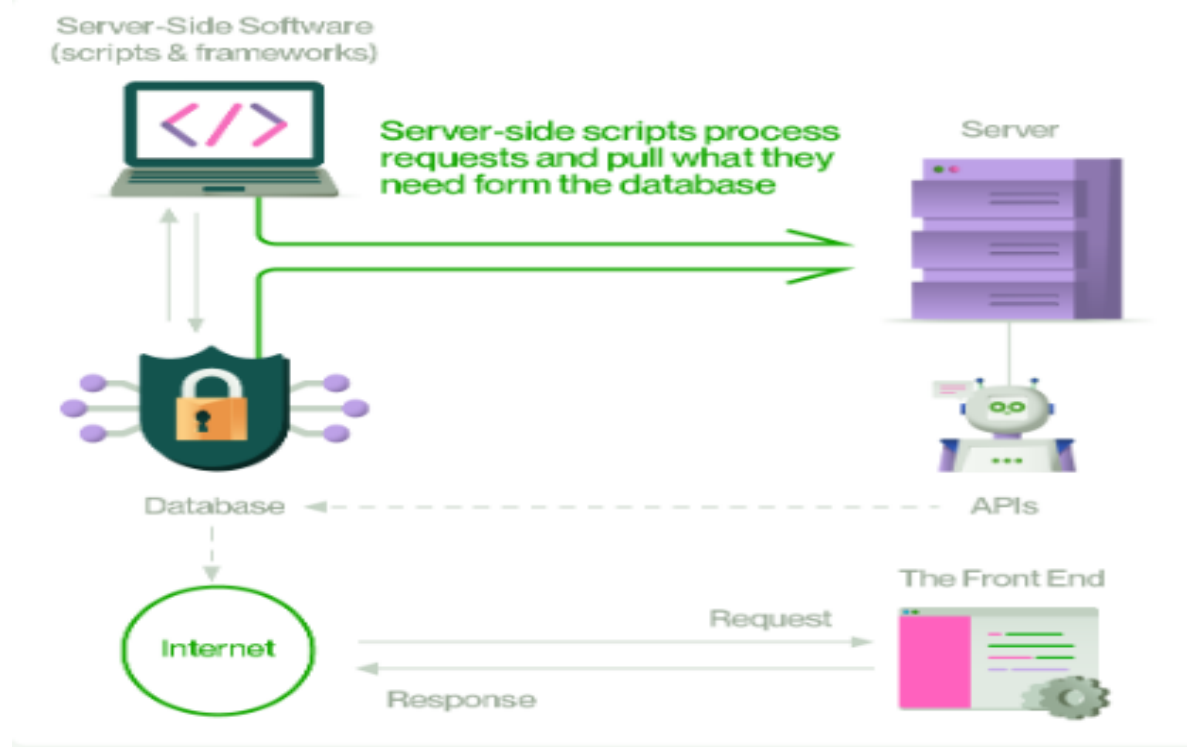


Figure 3.3.1: Back-End Development

Frameworks are collections of server-side programming languages that help build a website's back end. In it's project we have used **Laravel PHP Framework and My SQL** to implement the database.

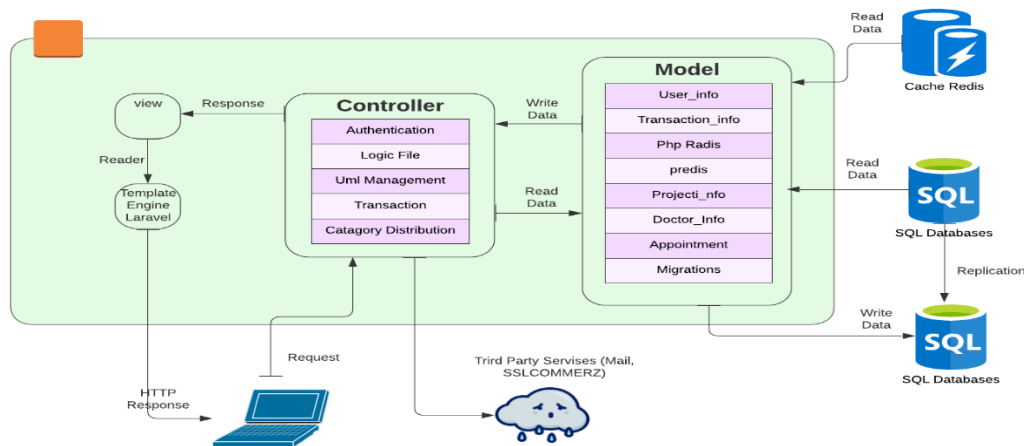


Figure 3.3.2: Back-End Design

Laravel is a reliable and simple to use open-source PHP framework. Laravel employs pre-existing components from various frameworks and adheres to the model-view-controller architectural pattern to assist in the development of online applications. The resulting web application is more organized and practical. The database, server-side framework, server, and operating system make up the "stack."

3.4 Interaction Design and User Experience (UX)

Sometimes, the terms "interaction design" and "user experience design" are used synonymously. Given the substantial overlap between interaction design and user experience design, that makes sense. The connection between a user and a product is crucial since UX design is all about impacting the user experience of a product. However, user experience and interaction design are not the same thing.

The way we approach user interactions is where UX and interaction design diverge most. The objective of interaction designers is to enhance the interactive experience for users at the point where they interact with a product. The user's path when interacting with a product is what UX designers focus on, not only the time of engagement.

To comprehend what interaction design entails, the five aspects of interaction design serve as a useful model.

1D: The website's words are easy to grasp and can convey the appropriate quantity of information to the user.

2D: To supplement words in communicating information to the user, visual representations like as images, icons, and typography are used.

3D: Our website is completely responsive to whatever hardware the user is using, and Physical Items or Space refers to the actual hardware and objects that a user interacts with.

4D: Time, users can gauge their progress through the use of sound and animation as well as through how long they spend interacting with the first three dimensions.

5D: Kevin Silver added the fifth dimension, "Behavior," which describes how the preceding dimensions define how a product interacts with its users. It also covers consumer and product responses.

This website's digital environment is shaped by user experience design, which leads visitors across its breadth and provides them with an emotional experience. A memorable experience is created when the visuals, content, structure, and navigation all work together. The UX is focused on the end user. If a user can quickly discover the information they need, they have succeeded in their task and may come back.

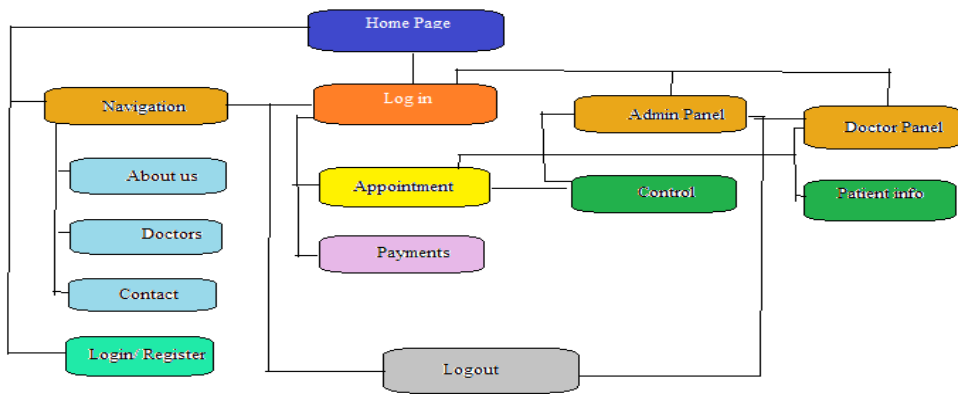


Figure 3.4.1: UX Site Mapping

3.5 Implementation Requirements

The actual process of converting requirements into website architectural designs, detailed designs, code, and test cases is known as requirements implementation. If needs are as precise and quantitative as feasible, the implementation of these work products is less difficult. The work results from website testing and architectural designing are the most in line with specifications. However, it is not simple to translate a set of functional and non-functional requirements into an architecture and reliable test cases. There isn't much information on how various architectural choices effect needs. Furthermore, even with adequate project resources, it is challenging to create test cases that accurately reflect requirements.

To implement the project, we have used some tools like VS code as editor, Xampp for the database environment.

3.5.1 VS Code

Source-code editors for Windows, Linux, and macOS were made by Microsoft and are frequently referred to as VS Code or Visual Studio Code. Among the features are debugging assistance, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git.

Users can alter the theme, keyboard shortcuts, preferences, and extensions that provide additional functionality.

According to the 82,000 participants in the Stack Overflow 2021 Developer Survey, 70% of them use Visual Studio Code as their primary development environment tool.

Feature:

- Develop. Write, navigate, and fix your code quickly.
- Debug. Diagnose, profile, and debug with ease.
- Test. Utilize thorough testing techniques to produce code of the highest caliber.
- Collaborate.
- Use version control, exercise agility, and effectively collaborate.
- Extend.

3.5.2 XAMPP

The open-source web server package called XAMPP deals with several stages. It is actually an acronym, with X denoting the "cross" stage, A denoting the Apache HTTP server, M denoting MySQL, P denoting PHP, and P denoting Perl. XAMPP was created with the intention of assisting web designers, programmers, software engineers, and planners in checking and auditing their work on their PCs even when they are not connected to the internet. In this way, XAMPP can essentially be used to maintain web pages even when not connected to the internet. Additionally, it may be used to create and design databases that are written in SQLite and MySQL. Furthermore, because XAMPP is designed to be a cross-platform server bundle, it may run on a variety of platforms and operating systems, including Microsoft Windows.

Formally, the developers of XAMPP intended it to be used only as a development tool, allowing web designers and software engineers to test their work independently on PCs without access to the Internet. Numerous crucial security features are automatically disabled in order to make this as simple as is reasonably possible given the conditions. XAMPP has the ability to serve web pages. The most important components of the package are watchword secured using an exceptional device.

3.6 Conclusion

System Design Specifications are the technical guidelines for the Software created to make it possible for any developer, programmer, or other individual with a reasonable level of expertise in software design, analysis, or programming to maintain and advance the Software, including all derivatives.

To avoid errors later in the system life cycle, it is crucial to set accurate design specifications early in the development phase. We still have to accept that requirements do change over time, though. Making sound requirements is difficult for a number of reasons. Because embedded systems must interface with the outside world in addition to other software components, their requirements are considerably more complicated than those for other systems. People with both technical and communication abilities are needed to establish the proper requirements.

CHAPTER 4: System Development

4.1 Introduction

The system development process organizes the target system implementation process and serves as the foundation for the system testing, verification, and conformance verification processes. The system development goal realization and system verification processes are constructed around the system development practice process and system implementation target results.

When creating an unique model, there isn't a previous system that can be observed in action and there isn't much or any knowledge available to describe what it might accomplish or how it might be made. To get from a situation where the system is available to a user when there was not a prior physical reality at hand, we must proceed through what is frequently a period of several years. We start this approach by building a number of models of the system we are designing.

4.2 Core Module Coding Samples

This is home module which authenticates user as User, Doctor or Admin.

```
public function addview()
{
    if(Auth::id())
    {
        if(Auth::user()->usertype==1)
        {
            return view('admin.add_doctor');
        }
        else{
            return redirect()->back();
        }
    }
    else
    {
        return redirect('login');
    }
}
```


This is add doctor function. This function allows us to add a new doctor to the database. The admin has the power to do this. The admin can add the doctors name, address, room no, photo etc.

```
public function upload(Request $request)
{
    $doctor=new doctor;

    $image = $request->file;

    $imagenname = time().'.'.$image->getClientOriginalExtension();

    $request->file->move('doctorimage',$imagenname);

    $doctor->image = $imagenname;

    $doctor->name = $request->name;

    $doctor->phone = $request->number;

    $doctor->room = $request->room;

    $doctor->speciality = $request->speciality;

    $doctor->save();

    return redirect()->back()->with('message','Doctor Added Successfully');
}
```

This Function shows appointments to the admin navigation

```
public function showappointment()
{
    $data=appointment::all();

    $data2=order::all();

    return view('admin.showappointment', compact('data','data2'));
}
```

This function shows all the doctors working in the hospital by fetching the data from the database.

```
public function showdoctor()
{
    $data=doctor::all();

    $data2=user::where('usertype','=',2)->get();

    return view('admin.showdoctor',compact('data','data2'));
}
```

4.3 Possible Problem Breakdowns

Every software has a common issue, thus I broke my project into several smaller tasks. The following are a few typical issues:

1. Ineffective scheduling.
2. Inadequate Requirements
3. Insufficient software testing
4. Redundant data.
5. Poor User Interface
6. Improper management of files and data.

I'll outline my steps for reducing this widespread issue below:

1. Real-time analysis and client input on the current web-based application
2. Gather precise requirements from a few hospitals and customer reviews.
3. I created a database with the proper relation after analyzing and planning the entire project.
4. Test each task thoroughly and seek user input for my project.

4.4 System Development Life Cycle

The majority of businesses find it advantageous to create and maintain their information system using a set of procedures known as a system development methodology. The evolution of information systems frequently follows a life cycle, like many other processes. The Systems Development Life Cycle (SDLC), which includes multiple phases that indicate the advancement of the system analysis and design work, is a typical technique for system development in many businesses.

Any life cycle may initially seem to have a series of phases that are sequentially organized, however this is not the case. The specific steps and their order are supposed to be modified in accordance with the management approach as needed for a project. For instance, the project can go back to a previous phase during any given SDLC phase if necessary. A commercial product may be briefly taken off the market and upgraded before being reintroduced if it does not perform well soon after its launch. It is also conceivable to finish some tasks from one phase of the system development life cycle while simultaneously finishing others from a different phase. The life cycle can iterate at times;

4.4 Conclusion

The process of developing man-made systems in order to meet preset needs is known as systems development. In order to produce new information that is specifically relevant to the problem and, as a result, identify a solution, we use the necessary components of humankind's knowledge base. It is a process for addressing problems in which we apply the proper knowledge from the body of human knowledge to produce new knowledge that is unique to the problem and, as a consequence, identify a solution to the problem.

CHAPTER 5:

Implementation and Testing

5.1 Introduction

In general, the practice of testing technological specification implementations is referred to as implementation and testing. This procedure checks both the practical applicability of the specification and the conformance of implementations to the specification. The effectiveness and compatibility of implementations are enhanced by this technique.

We have implemented databases, routes, tables, frameworks, libraries, bootstrap etc. also we have the implementation of front-end log-in forms, registration forms etc. After implementing all these it is imperative to test these features. Testing is a crucial part of any project. Before releasing any product it is imperative to check if the features are working or not. Testing helps improve the system or project and find bugs and broken links. It helps save a lot of money and time.

This phase of the project helps in maintaining and repairing the system and automation testing is very helpful in long-term maintenance.

In this section, we have shown the implementation of the database, data table names, datatype architectures, MySQL data view etc.

5.2 Implementation of Database

The WBHMS (Web-Based Hospital Management System) database is selected for system application with the following sketch.

Database design refers to the process of developing a data model for a database. In order to produce a data defining language design that may be used to create a database, This reasonable data structure must include all of the required fair and bodily implementation options and actual store characteristics. Each entity in a properly attributed data model has certain attributes. The phrase "database design" can be used to refer to a wide range of

components that make up a database system's overall graph. It is preferable to think as the rational layout of the key information structures that are utilized to store the data. These are the tables found in the relational model.

Table 5.1: Hardware Requirements of the system

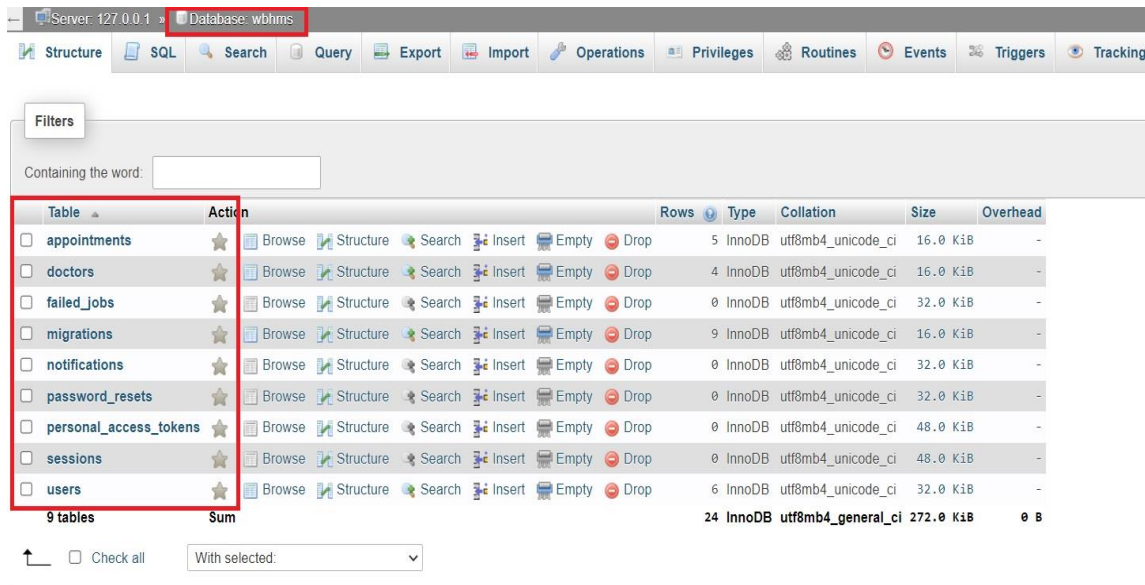
Hardware	Minimum System requirements
Processor	1.2 GHz Processor Speed
Memory	128 MB RAM (256 Preferable)
Disk Space	60 GIGABYTES
Display	800X600 Colors (1024x768 High color 16- bit recommended)

Table 5.2: Hardware Requirements of the system

Software	Minimum System Requirements
Operating System	Windows 7 or later
Runtime Environment	XAMPP Server
Database Management System	MySQL

5.2.1 Data-Table Name

The name of this Hospital Management System’s database is “wbhms” which has a couple of data tables.



The screenshot shows a database management interface for the 'wbhms' database. A red box highlights the table names in the 'Table' column. The tables listed are: appointments, doctors, failed_jobs, migrations, notifications, password_resets, personal_access_tokens, sessions, and users. The interface also shows a 'Filters' section, a 'Check all' checkbox, and a 'With selected:' dropdown menu.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> appointments	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> doctors	★ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> migrations	★ Browse Structure Search Insert Empty Drop	9	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> notifications	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> password_resets	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
<input type="checkbox"/> sessions	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
<input type="checkbox"/> users	★ Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
9 tables	Sum	24	InnoDB	utf8mb4_general_ci	272.0 KiB	0 B

Figure 5.2.1: Database Table Name

5.2.2 Data Type Architecture

Data architecture should create data standards for all of its data systems as a vision or model of their ultimate relationships. Because data integration necessitates data interactions between two or more data systems, data architectural standards should be used.

For its column and parameter declarations, the following data types are used.

- CHARACTER [CHAR [(length)]]
- VARCHAR (length)
- BOOLEAN
- SMALLINT
- INTEGER or INT

- DECIMAL
- NUMERIC
- REAL
- FLOAT(p)
- DOUBLE PRECISION
- DATE
- TIME
- TIMESTAMP
- CLOB [CHARACTER LARGE OBJECT [(length)] or CHAR LARGE OBJECT [(length)]
- BLOB [BINARY LARGE OBJECT [(length)]

5.2.3 Data Table Architecture

wbhms users	
id	: bigint(20) unsigned
name	: varchar(255)
email	: varchar(255)
phone	: varchar(255)
address	: varchar(255)
usertype	: varchar(255)
email_verified_at	: timestamp
password	: varchar(255)
two_factor_secret	: text
two_factor_recovery_codes	: text
two_factor_confirmed_at	: timestamp
remember_token	: varchar(100)
current_team_id	: bigint(20) unsigned
profile_photo_path	: varchar(2048)
created_at	: timestamp
updated_at	: timestamp

wbhms personal_access_tokens	
id	: bigint(20) unsigned
tokenable_type	: varchar(255)
tokenable_id	: bigint(20) unsigned
name	: varchar(255)
token	: varchar(64)
abilities	: text
last_used_at	: timestamp
created_at	: timestamp
updated_at	: timestamp

wbhms password_resets	
email	: varchar(255)
token	: varchar(255)
created_at	: timestamp

wbhms doctors	
id	: bigint(20) unsigned
name	: varchar(255)
phone	: varchar(255)
speciality	: varchar(255)
room	: varchar(255)
image	: varchar(255)
created_at	: timestamp
updated_at	: timestamp

wbhms migrations	
id	: int(10) unsigned
migration	: varchar(255)
batch	: int(11)

wbhms appointments	
id	bigint(20) unsigned
name	varchar(255)
email	varchar(255)
phone	varchar(255)
doctor	varchar(255)
date	varchar(255)
message	varchar(255)
status	varchar(255)
user_id	varchar(255)
created_at	timestamp
updated_at	timestamp

wbhms notifications	
id	char(36)
type	varchar(255)
notifiable_type	varchar(255)
notifiable_id	bigint(20) unsigned
data	text
read_at	timestamp
created_at	timestamp
updated_at	timestamp

wbhms sessions	
id	varchar(255)
user_id	bigint(20) unsigned
ip_address	varchar(45)
user_agent	text
payload	longtext
last_activity	int(11)

wbhms failed_jobs	
id	bigint(20) unsigned
uuid	varchar(255)
connection	text
queue	text
payload	longtext
exception	longtext
failed_at	timestamp

Figure 5.2.2: Database Table Architecture

5.2.4 My SQL Database View

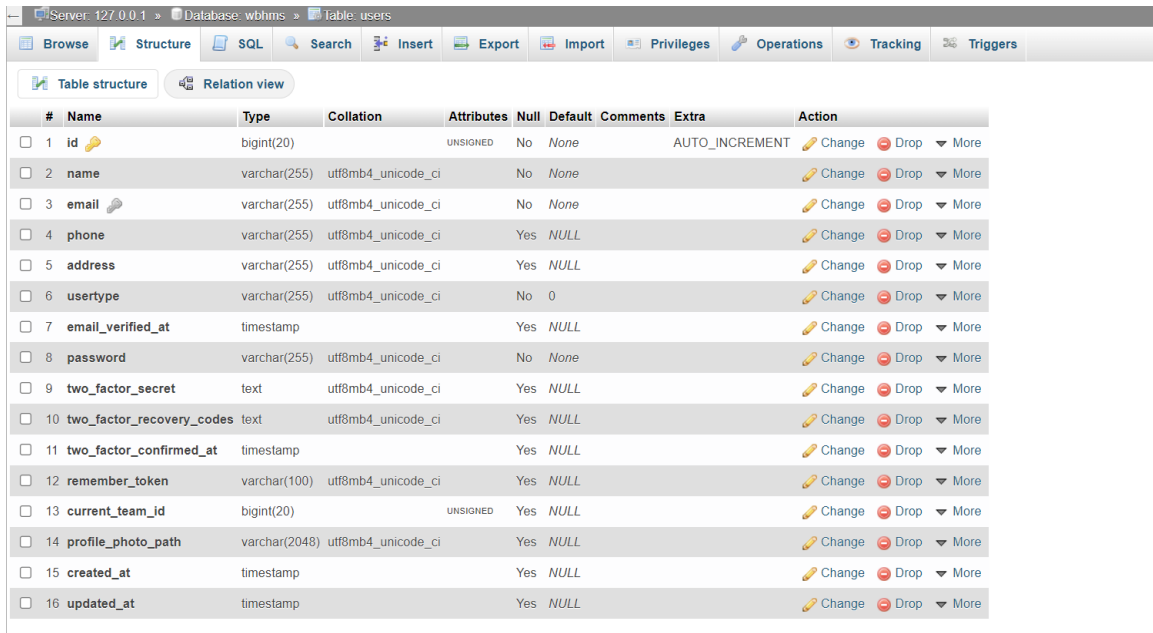
The screenshot displays the phpMyAdmin interface for a MySQL database named 'wbhms'. The left sidebar shows a tree view of databases and tables. The main area shows a table listing the tables in the database. Below the table is a 'Create table' dialog with fields for 'Name' and 'Number of columns'.

Table	Action	Rows	Type	Collation	Size	Overhead
<input type="checkbox"/> appointments	★ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> doctors	★ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> migrations	★ Browse Structure Search Insert Empty Drop	9	InnoDB	utf8mb4_unicode_ci	16.0 KiB	-
<input type="checkbox"/> notifications	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> password_resets	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
<input type="checkbox"/> sessions	★ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 KiB	-
<input type="checkbox"/> users	★ Browse Structure Search Insert Empty Drop	6	InnoDB	utf8mb4_unicode_ci	32.0 KiB	-
9 tables	Sum	24	InnoDB	utf8mb4_general_ci	272.0 KiB	0 B

Figure 5.2.3: My SQL Database View

5.2.5 Database Table Structure

Here table (**users**) structure shown below:



#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	bigint(20)		UNSIGNED	No	None		AUTO_INCREMENT	Change Drop More
2	name	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
3	email	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
4	phone	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
5	address	varchar(255)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
6	usertype	varchar(255)	utf8mb4_unicode_ci		No	0			Change Drop More
7	email_verified_at	timestamp			Yes	NULL			Change Drop More
8	password	varchar(255)	utf8mb4_unicode_ci		No	None			Change Drop More
9	two_factor_secret	text	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
10	two_factor_recovery_codes	text	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
11	two_factor_confirmed_at	timestamp			Yes	NULL			Change Drop More
12	remember_token	varchar(100)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
13	current_team_id	bigint(20)		UNSIGNED	Yes	NULL			Change Drop More
14	profile_photo_path	varchar(2048)	utf8mb4_unicode_ci		Yes	NULL			Change Drop More
15	created_at	timestamp			Yes	NULL			Change Drop More
16	updated_at	timestamp			Yes	NULL			Change Drop More

Figure 5.2.4: Database Table (users)

5.2.6 Routes

Api.php

```
<?php

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Route;

/*
|-----
| API Routes
|-----
|
|
```

Here is where you can register API routes for your application. These routes are loaded by the RouteServiceProvider within a group which is assigned the "api" middleware group. Enjoy building your API!

```
*/
```

```
Route::middleware('auth:sanctum')->get('/user', function (Request $request) {  
    return $request->user();  
});
```

Channel.php

```
<?php
```

```
use Illuminate\Support\Facades\Broadcast;
```

```
/*
```

```
|-----  
| Broadcast Channels  
|-----
```

```
|  
| Here you may register all of the event broadcasting channels that your  
| application supports. The given channel authorization callbacks are  
| used to check if an authenticated user can listen to the channel.
```

```
*/
```

```
Broadcast::channel('App.Models.User.{id}', function ($user, $id) {  
    return (int) $user->id === (int) $id;  
});
```

Console.php

```
<?php

use Illuminate\Foundation\Inspiring;
use Illuminate\Support\Facades\Artisan;

/*
|-----
| Console Routes
|-----
|
| This file is where you may define all of your Closure based console
| commands. Each Closure is bound to a command instance allowing a
| simple approach to interacting with each command's IO methods.
|
*/

Artisan::command('inspire', function () {
    $this->comment(Inspiring::quote());
})->purpose('Display an inspiring quote');
```

Web.php

```
<?php

use Illuminate\Support\Facades\Route;
use App\Http\Controllers\HomeController;

use App\Http\Controllers\AdminController;

/*
|-----
| Web Routes
|-----
|
| Here is where you can register web routes for your application. These
```

```

| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

Route::get('/',[HomeController::class,'index']);

Route::get('/home',[HomeController::class,'redirect']->middleware('auth','verified'));

Route::middleware([
    'auth:sanctum',
    config('jetstream.auth_session'),
    'verified'
])->group(function () {
    Route::get('/dashboard', function () {
        return view('dashboard');
    })->name('dashboard');
});

Route::get('/add_doctor_view',[AdminController::class,'addview']);

Route::post('/upload_doctor',[AdminController::class,'upload']);

Route::post('/appointment',[HomeController::class,'appointment']);

Route::get('/myappointment',[HomeController::class,'myappointment']);

Route::get('/cancel_appointment/{id}',[HomeController::class,'cancel_appointment']);

Route::get('/showappointment',[AdminController::class,'showappointment']);

Route::get('/approved/{id}',[AdminController::class,'approved']);

Route::get('/canceled/{id}',[AdminController::class,'canceled']);

```

```
Route::get('/showdoctor',[AdminController::class,'showdoctor']);

Route::get('/deletedoctor/{id}',[AdminController::class,'deletedoctor']);

Route::get('/updatedoctor/{id}',[AdminController::class,'updatedoctor']);

Route::post('/editdoctor/{id}',[AdminController::class,'editdoctor']);

Route::get('/emailview/{id}',[AdminController::class,'emailview']);

Route::post('/sendemail/{id}',[AdminController::class,'sendemail']);

Route::get('/doctorlist',[HomeController::class,'doctorlist']);

Route::get('/aboutus',[HomeController::class,'aboutus']);

Route::get('/newsdetails',[HomeController::class,'newsdetails']);

Route::get('/contact',[HomeController::class,'contact']);
```

5.3 Implementation of Front-End Design

The process of developing HTML, CSS, Bootstrap, and JavaScript for a website or Web application so that a user can view and interact with them directly is known as front-end web development, often referred to as client-side development. The challenge with front-end development is that the methods and technology used to design the front end of a website are always evolving, demanding regular monitoring of industry improvements.

The goal of developing this site is to guarantee that when people access it, they see the material in an easy-to-read and relevant style. This is exacerbated further by the fact that visitors today utilize a wide range of devices with varied screen sizes and resolutions, requiring the designer to consider these factors while building the site. They must guarantee that their site displays appropriately on multiple browsers (cross-browser), operating systems (cross-platform), and devices (cross-device), which necessitates significant planning on the developer's part.

The screenshot below shows the main view of the project. It's a captured image of what users will be able to see on screen.

Home Page: On the home page there are Eleven options available

The screenshot shows a healthcare website home page with the following elements:

- Header:** Contact information (+00 123 4455 6666, onehealth1000@example.com) and social media icons (Facebook, Twitter, Instagram). Navigation links include Home, About Us, Doctors, News, Contact, Login, and Register.
- Hero Section:** A large image of hands in white gloves with the text "LET'S MAKE YOUR LIFE HAPPIER" and "Healthy Living". A "Let's Consult" button is centered below the text.
- Service Cards:** Three cards below the hero section: "Chat with a doctors" (with a chat icon), "One-Health Protection" (with a shield icon), and "One-Health Pharmacy" (with a pharmacy icon).
- Welcome Section:** A section titled "Welcome to Your Health Center" featuring a photo of a female doctor in a white coat and mask. Below the photo is a "Learn More" button. The text below the title is placeholder text: "Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Accusantium aperiam earum ipsa eius, inventore nemo labore eaque porro consequatur ex aspernatur. Explicabo, excepturi accusantium! Placeat voluptates esse ut optio facilis!"
- Our Doctors Section:** A section titled "Our Doctors" displaying three doctor profiles with placeholder images and names. The first profile is labeled "Dialy", the second "Surgery", and the third "Heart". A navigation arrow is located below the profiles.

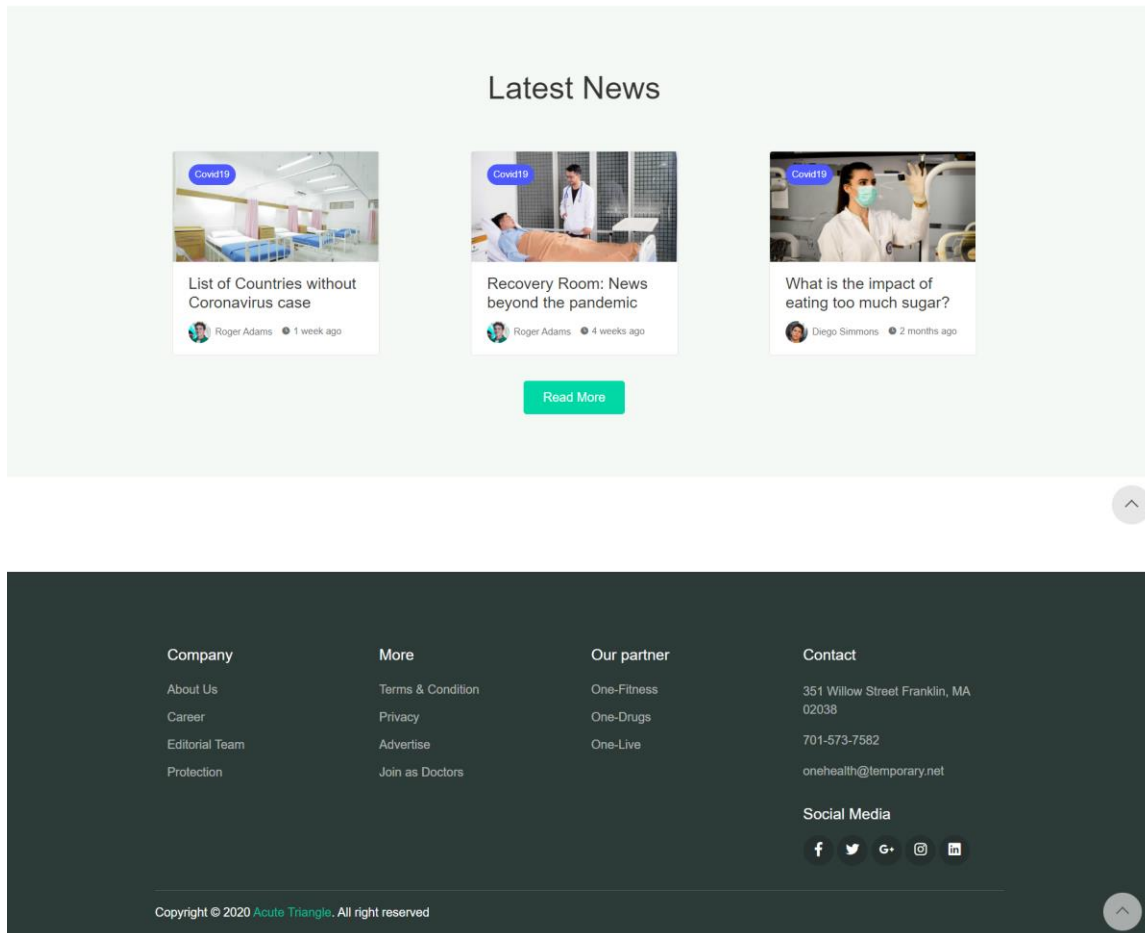
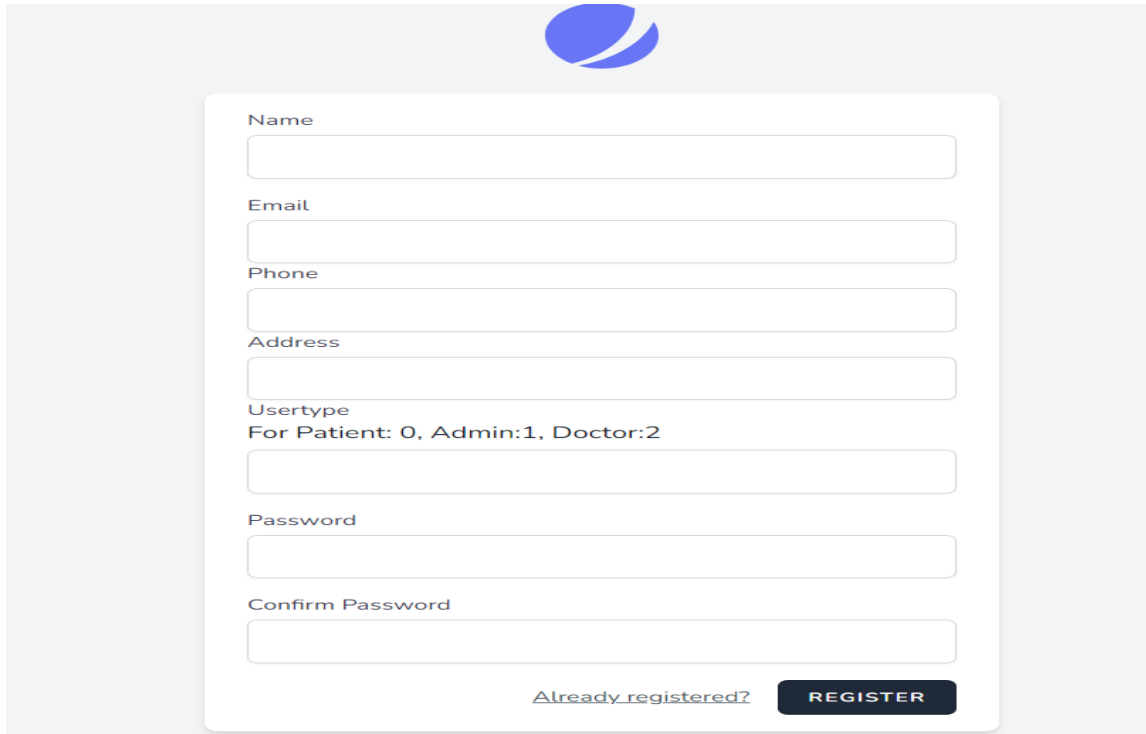


Figure 5.3.1: Hospital Management Home Page

Registration: Users need to fill up the registration form to create an account and to login into the system.

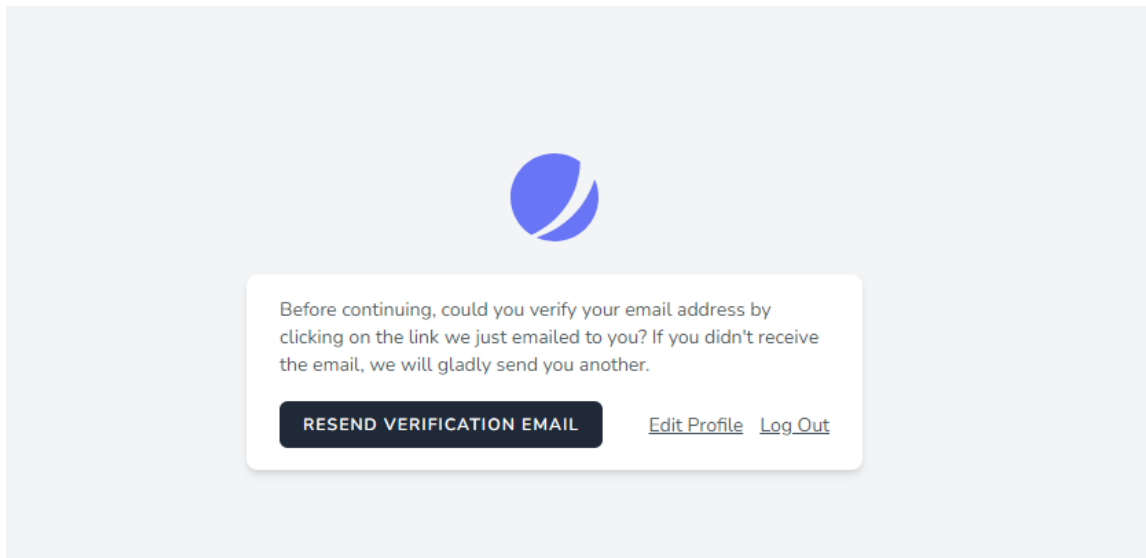
User needs to give Name, Email, Phone, Address, User type (How he/she wants to register themselves like as a doctor, patient or admin etc.), and Password (at least 8 characters). If the user already registered or have an account, they click on the already registered? And go to the login page to access their account.



The image shows a user registration form on a light gray background. At the top center is a blue circular logo with a white swoosh. The form is a white rounded rectangle containing the following fields from top to bottom: Name, Email, Phone, Address, Usertype (with a sub-label 'For Patient: 0, Admin:1, Doctor:2'), Password, and Confirm Password. Each field is a white rectangular input box. At the bottom right of the form is a dark blue button labeled 'REGISTER'. To the left of the button is a link that says 'Already registered?'.

Figure 5.3.2: User Registration Page

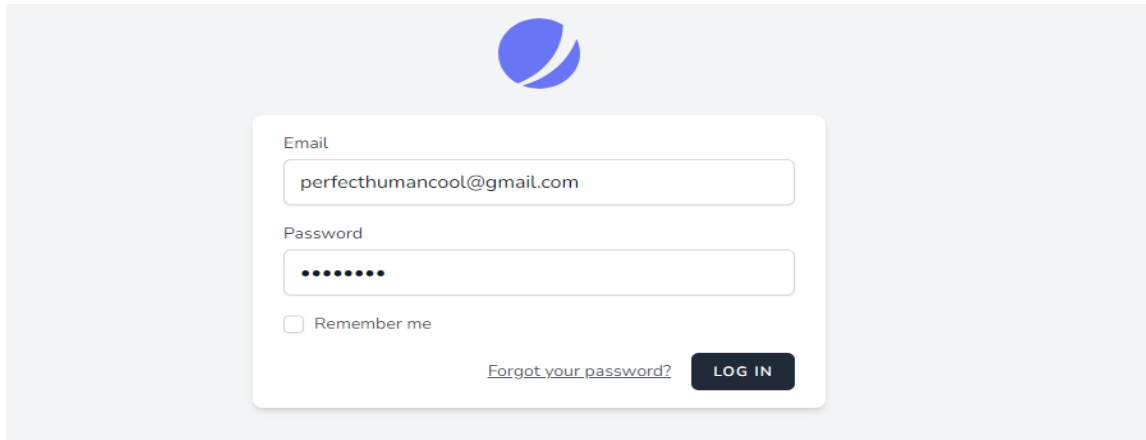
Validation: An Email with a validation link will be sent from the hospital to validate the user email.




The image shows a user validation page on a light gray background. At the top center is the same blue circular logo with a white swoosh. Below the logo is a white rounded rectangle containing the following text: 'Before continuing, could you verify your email address by clicking on the link we just emailed to you? If you didn't receive the email, we will gladly send you another.' Below this text is a dark blue button labeled 'RESEND VERIFICATION EMAIL'. To the right of the button are two links: 'Edit Profile' and 'Log Out'.

Figure 5.3.3: User validation Page

Login & After Login:





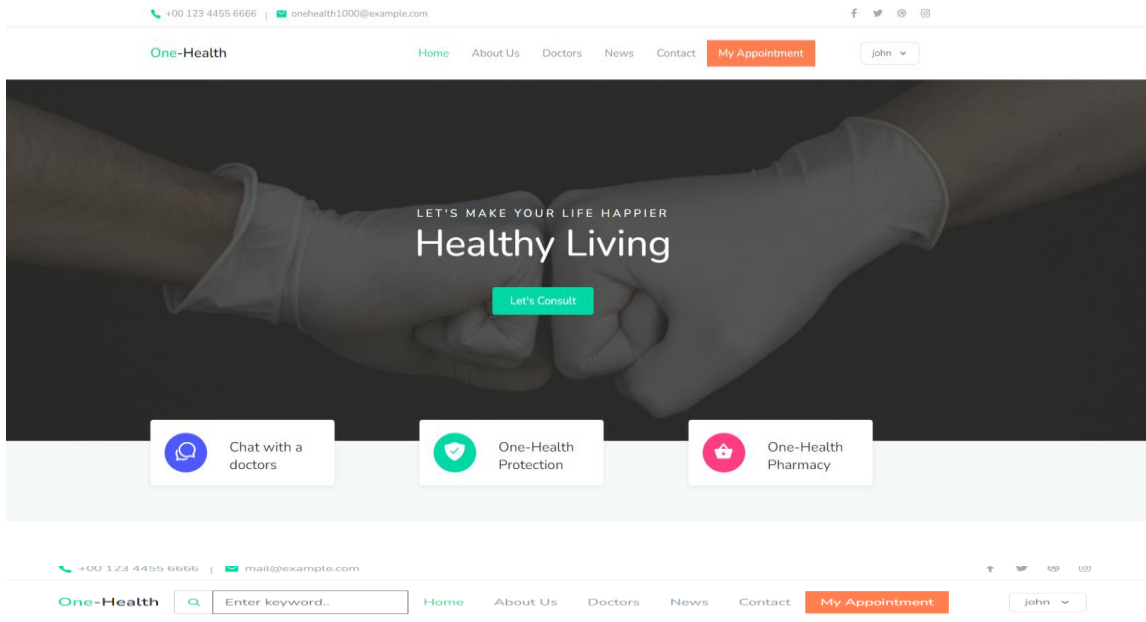
Email

Password

Remember me

[Forgot your password?](#)

After login available option will appear along with my appointment, where the patient can check the progress of the appointment.



+00 123 4455 6666 | onehealth1000@example.com

One-Health Home About Us Doctors News Contact My Appointment john

LET'S MAKE YOUR LIFE HAPPIER

Healthy Living

Let's Consult

Chat with a doctors One-Health Protection One-Health Pharmacy

+00 123 4455 6666 | mail@example.com

One-Health Enter keyword.. Home About Us Doctors News Contact My Appointment john

Doctor Name	Date	Message	Status	Cancel Appointment
Dr.Nabil	2022-08-11	drh	approved	Cancel

Make an Appointment

Full name	Email address..
mm/dd/yyyy	--Select Doctor--
Number..	
Enter message..	
<input type="submit" value="Submit Request"/>	

Figure 5.3.4: Login & After login page and feature

Admin panel and Features:

The screenshot displays the CORONA admin panel. At the top left is the logo 'CORONA' and a hamburger menu icon. At the top right is a user profile dropdown labeled 'admin'. On the left side, there is a navigation menu with three items: 'Add Doctors' (highlighted with a green background), 'Appointments', and 'All Doctors'. The main content area contains a form for adding a doctor with the following fields: 'Doctor Name' (text input with placeholder 'Write the name'), 'Phone' (text input with placeholder 'Phone Number'), 'Speciality' (dropdown menu with '--Select--'), 'Doctor Room' (text input with placeholder 'Write the Room Number'), and 'Doctor Image' (file upload button labeled 'Choose File' and 'No file chosen'). A 'Submit' button is located at the bottom of the form.

CORONA admin ▾

Navigation

- Add Doctors
- Appointments
- All Doctors

Customer Name	Email	Phone	Doctor Name	Date	Message	Status	Approved	Canceled	Send Mail
Junaeid Shoab	junaeidshoab@gmail.com	01711144445	Dr. Istak Ahmed	01711144445	nice	approved	Approved	Canceled	Send Mail
Junaeid Shoab	abu15-2316@diu.edu.bd	01711144445	Dr.Nabil	2022-07-01	need help	approved	Approved	Canceled	Send Mail
Junaeid Shoab	istak15-12037@diu.edu.bd	01711144445	Dr.Nabil	2022-07-01	jij	approved	Approved	Canceled	Send Mail
john	perfecthumancool@gmail.com	01735112233	Dr. Junaeid Shoab	2022-08-10	m	Canceled	Approved	Canceled	Send Mail
Junaeid Shoab	junaeidshoab@gmail.com	01711144445	Dr.Nabil	2022-08-11	djh	approved	Approved	Canceled	Send Mail

CORONA admin ▾

Navigation

- Add Doctors
- Appointments
- All Doctors





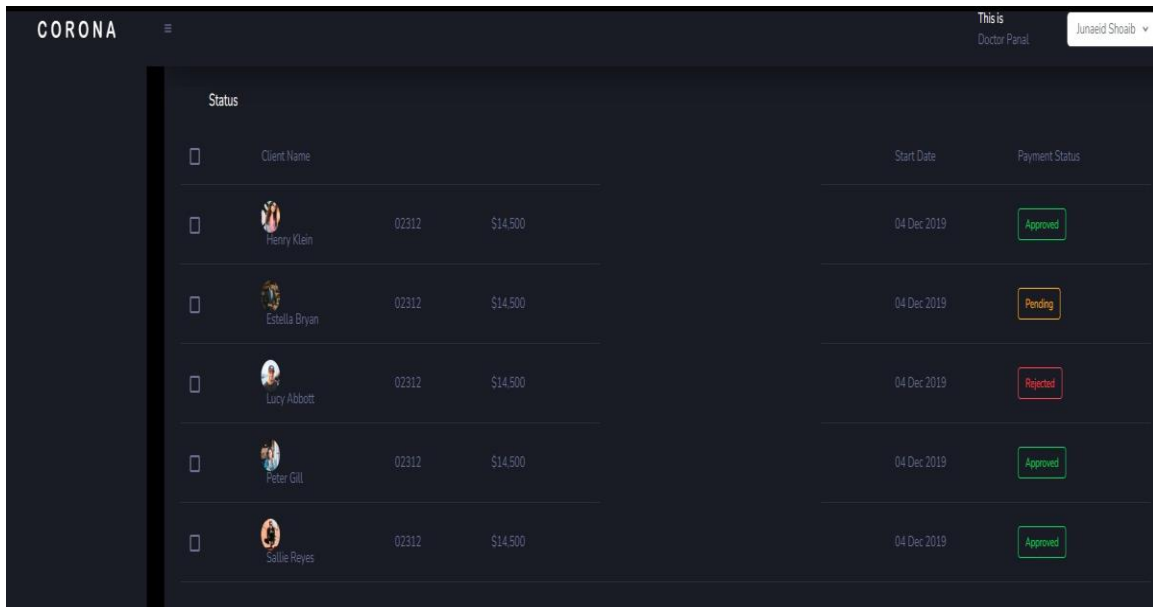
Doctor Name	Phone	Specility	Room No	Image	Dalete	Update
Ali Baba	01711144445	Brain	037		Delete	Update
Dr. Junaeid Shoab	01735112233	Surgery	102		Delete	Update
Dr.Nabil	01711144445	Heart	101		Delete	Update
Istak Ahmed	01622337656	Neurology	69		Delete	Update

Figure 5.3.5: Admin panel

Doctor panel:



The screenshot shows a web application interface for a doctor's panel. The header includes the logo 'CORONA' and the user's name 'Junaid Shoaib'. The main content area is titled 'Status' and displays a table of client information. Each row represents a client with a checkbox, a profile picture, the client's name, an ID, an amount, a start date, and a payment status button.






<input type="checkbox"/>	Client Name	ID	Amount	Start Date	Payment Status
<input type="checkbox"/>	 Henry Klein	02312	\$14,500	04 Dec 2019	Approved
<input type="checkbox"/>	 Estella Bryan	02312	\$14,500	04 Dec 2019	Pending
<input type="checkbox"/>	 Lucy Abbott	02312	\$14,500	04 Dec 2019	Rejected
<input type="checkbox"/>	 Peter Gill	02312	\$14,500	04 Dec 2019	Approved
<input type="checkbox"/>	 Sallie Reyes	02312	\$14,500	04 Dec 2019	Approved

Figure 5.3.6: Doctor panel

5.4 Testing Implementation

The following criteria were used to evaluate this project:

Fulfilling requirement specifications: If a project fulfills all requirements, including functional and non-functional needs, it is considered to be successful. It should be able to guarantee the requirements specifications, in other words.

Correctness: Correctness is one of the essential criteria for software development. For service-oriented software, perfection is the fundamental need. The program should function precisely and correctly in every single component.

Compatibility and Integrity: Integrity and compatibility are two key factors that determine whether a project will succeed. Any domain can use the web-based hospital administration system, which was designed for this purpose. It was also created in a way that it may influence virtualization, which is seen as a crucial element. Additionally, whether or not the application was integrated into the entire system affected how the system was evaluated.

Management of real-time: Real-time management: The program is used to manage health sectors. As a result, the real-time scenario must be preserved. Users of this system should be able to keep it running.

Reliability and security management: Security is a key component of every service-oriented system, and reliability and security management are two areas where they must coexist. The security characteristics were taken into consideration when the system was designed, and this is why those criteria were used to evaluate them.

User-friendliness: User-friendliness: Another unique factor used to evaluate the systems is the friendliness of any application. For instance, when utilizing the system, consumers of this solution ought to feel satisfied. In essence, a system should possess quality metrics including effectiveness, portability, reuse, adaptability, coherence, and loose coupling between various software-designed components.

5.4.1 Methodology of Testing

Testing System problems are typically discovered through software testing. A software experiment can be carried out to examining codes, design and acting to the whole system. Trying is inevitable to raise the quality of the system. Reviewing and Another fundamental concept is code testing of software engineering that is often overlooked in project development. Testing is an integral part of the system development process.

5.4.2 Testing for Functionality

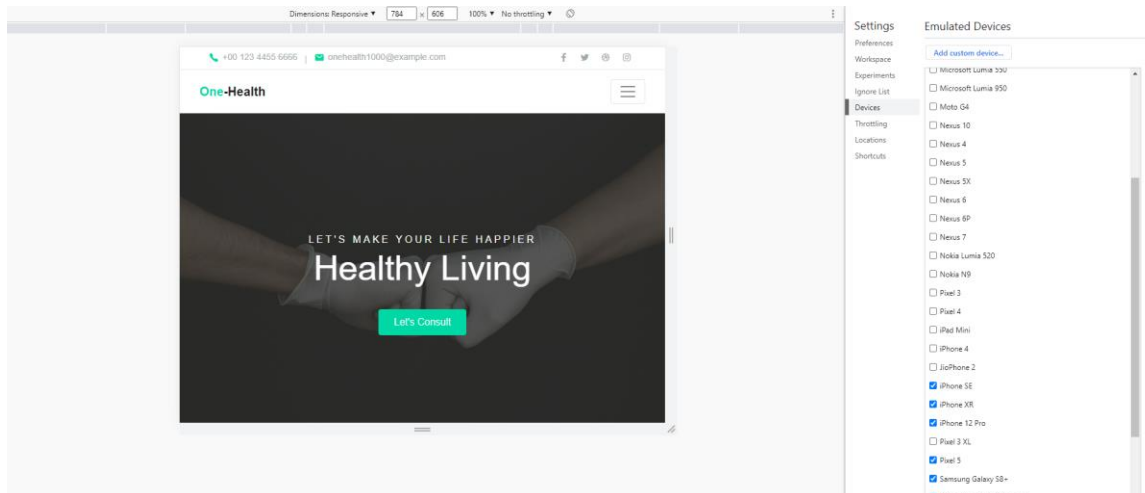
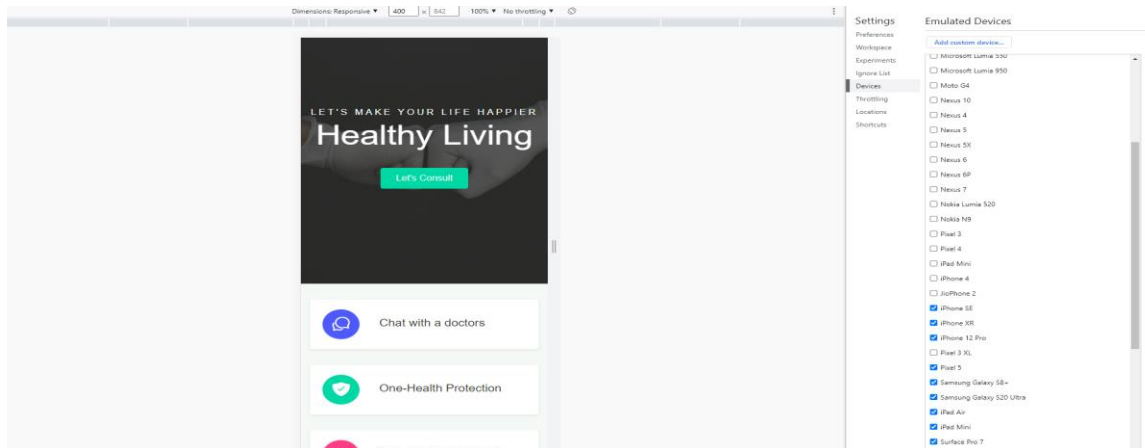
In functional testing, the tester must check the application to ensure that all of the user's defined needs, as stated in the supplemental constraint system, have been integrated or not.

There are two categories of functional testing:

Functional testing that is positive: Valid input is used to test the application's functionalities, as well as to ensure that the outputs are correct.

Functional testing that is negative: IT entails testing application functioning with a combination of incorrect inputs, unexpected operating situations, and other "out-of-bounds" events.

5.4.3 System Responsiveness Testing



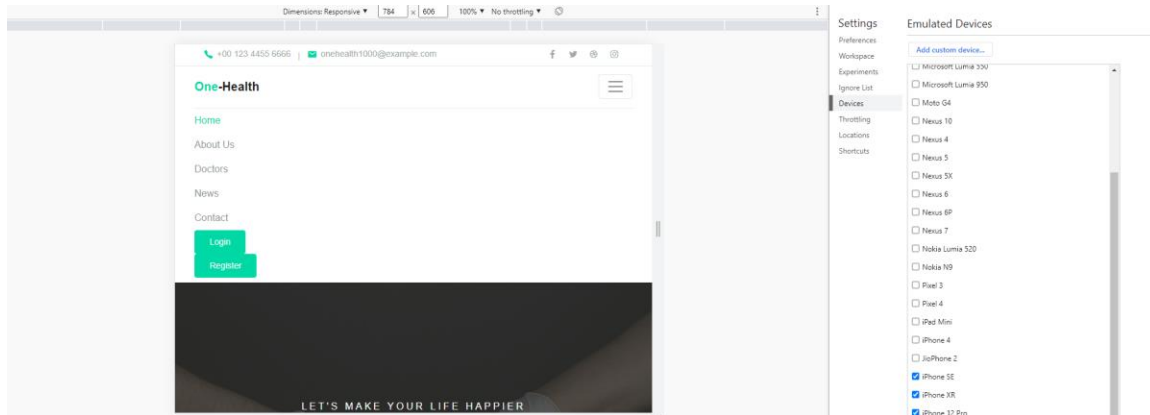


Figure 5.4.1: Responsiveness Testing

5.5 Test Results and Reports

a) Testing for System Integration

This is the testing done on the application that was tested, and it ensures that the entire program works as expected. Critical business scenarios were evaluated to ensure that critical application functionality functions as planned and without mistakes.

b) Regression Analysis

Each time a fresh build with defect repairs and new improvements is deployed for testing, regression testing is conducted. Regression testing is performed on the entire program, not just the new features and bug fixes. This testing guarantees that current functionality continues to function properly following a defect correction and the addition of new upgrades to the existing application. Test cases for new functionality are added to and performed alongside existing test cases.

5.5.1 Results

Table 5.3: Testing Results of the system

Test	User requirement addressed	Expected Results	Actual Results	Pass/fail	Date	Corrective Action
Logo Displayed	The website uses the company's house style	The logo is displayed and functional as the home button	As expected	Pass	12/08/2022	None
The navigation bar drop-down works	The website is easy to navigate	When clicked the bar drops down	As expected	Pass	12/08/2022	None
Navigation bar appears	The website is easy to navigate	Navigation bar appears	As expected	Pass	12/08/2022	None
Navigation bar hyperlink works	The website is easy to navigate	When clicked the Hyperlink works	As expected	Pass	12/08/2022	None

5.5.2 Testing Scope

a) In Scope

Functional Testing for the following modules is in the Scope of Testing

- Registration and login
- Payment
- Appointment

b) Out of Scope

Performance Testing was not done for this application.

c) Items not tested

Verification of connectivity with the third-party system Central repository system was not tested, as the connectivity could not be established due to some technical limitations. This can be verified during UAT (User Acceptance Testing) where the connectivity is available or can be established.

5.6 Conclusion

The process of creating test data, establishing and prioritizing test procedures, and, if necessary, setting up test harnesses and implementing automated test scripts is known as test implementation. When test designs are put into practice as test cases, test methods, and test data, tests are arranged and prioritized.

Selecting the appropriate tests and running them in the proper sequence is crucial. When we prioritize based on the likelihood of risk and difficulties, the significance of this even increases exponentially in risk-based tactics.

CHAPTER 6

Importance of Hospital Management System in Bangladesh

6.1 Introduction

Computers, cellphones, and iPads are all used in our modern world system, and they are all connected to one another through an efficient network for implementation. Healthcare is the most important component of our society, and many healthcare workers struggle to provide patients with useful, effective, and active services.

Now Think of a multispecialty hospital where numerous patients come in and leave out each day, making it difficult to secure and keep their information. A web-based hospital management system was created to reduce this kind of issue and manage the clinical, financial, and hospital administration aspects.

In addition to the difficulty of correctly automating and executing hospital operations, we require a reliable web-based hospital management system to monitor and control everything that occurs in labs and hospitals.

6.2 Necessity of Web-based Hospital Management System

In this modern world, everything is going online. Starting from NID cards, Businesses, land registrations, hospitals and many more. There are many ways to go online. Through Laptops, PC, smartphones like android phones, IOS phones, Ipads etc. Each of them is integrated differently to access the vast knowledge, information and data on the internet. Every application or app can't support all the devices. That's why web-based websites play a very important role in that.

With a web-based hospital management system, the site can be easily accessed online. Regardless of the device. Anyone can make an appointment in the safety of their time and money at home. Even the hospital administrators can access and know what is happening around the hospital just

by going online. Since it is online many security measures will keep the data and information of the patients and transactions between hospitals and people will be safe, secure and stored on a cloud. This is way more secure than a file-based system.

6.3 Compliance and Quality

Every hospital is required to submit a monthly report to the NABH accreditation detailing births, deaths, and the underlying causes and solutions. Selecting the best HMS lets you to send the reports fast and on time because organizing them manually is difficult.

The Hospital Management System carefully and effectively manages each report to ensure accurate findings.

For coordinated and quick care, lower costs, shorter wait times and readmissions, improved patient safety, and clinical care, everyone favours HMS for their hospitals.

6.4 Processing Time and Outcomes

Because all effective HMS systems follow the normal operating protocols, there are no chances for deviation. With the implementation of HMS in your labs and hospitals, you will be able to treat patients more successfully and quickly access their real-time reports, extra information about them, their prior clinical data, and more. The best patient results are anticipated as a result of this. The hospital management system makes work more accessible to staff and expedites process completion for better results.

6.5 Opportunities and Obstacles

The web-based hospital management system can create any opportunities and also with real-life problems there will be many obstacles with it.

The whole system can do many things like store and save transactions, user IDs, names, numbers, and pictures. It can also take appointments, and see how many people have made appointments. How many doctors there are, update their information and terminate them if needed. This will create job opportunities for admin positions, doctors can manage their time and work with a lot of patients. Patients can communicate with the hospitals directly rather than going around looking for people to find specific doctors.

With a real-life implemented application or system there will be obstacles too. Like storing the information and data needs a database with enough space. Now they can either have a server room or pay for the cloud. Although having server rooms can be beneficial for private uses it will be costly. But with monthly payments, clouds can do that same thing. If there is a need for a time when many HTTP requests start flooding the cloud can be helpful. Now, this is not a big obstacle. The main concern is with the internet connection. If the hospital has a weak internet connection the system will not work properly. So, to use the web-based hospital management system, the hospital will need to have a very strong internet connection.

6.6 Conclusion

The management of the health system must be logical and efficient because it is a crucial socioeconomic activity. A technology that enables sufficient control of the information generated in healthcare institutions is required for this.

As the primary actors in the health system, hospitals produce a significant amount of information, but it is frequently scattered or is not accessible in the required way or at the required time.

Without an effective system for managing healthcare facilities, a healthcare system would struggle to retain skilled employees, provide quality patient care, and generate a healthy profit

CHAPTER 7

Impact on Society, Environment and Sustainability

7.1 Introduction

A hospital management system, usually referred to as a hospital management information system, is a component of a medical informatics solution that primarily focuses on hospital administration needs. The HMS (Hospital Management System), a web-based or computer application, manages all hospital operations. All hospital operations are overseen by the customized integrated system, which includes patient data, appointment scheduling, billing, drug management, administration, electronic medical records, patient medical histories, inventory management, bed management, and revenue management.

Nursing homes, rehabilitation centers, clinics, hospitals, health clinics, dispensaries, and other healthcare institutions must all have a hospital management system. Some of the most significant benefits of implementing an HMS are role-based access control, data accuracy, revenue management, appointment scheduling, total cost savings, and data security.

7.2 Impact on Society

A hospital management system, often known as a hospital management information system, is a component of a medical informatics solution that focuses on the demands of hospital administration. The entire hospital functionalities are handled by the HMS (Hospital Management System), a web-based or computer application. The integrated system, which may be modified, manages all hospital activities such as patient data, appointment scheduling, billing, drug management, administration, electronic medical records, patient medical histories, inventory management, bed management, and revenue management.

A hospital management system is required for healthcare institutions such as nursing homes, rehabilitation centers, clinics, hospitals, health clinics, dispensaries, and others. The main benefits of using an HMS are role-based access control, data accuracy, revenue management, appointment scheduling, total cost savings, and data security.

Revenue administration:

A hospital or medical facility benefits humanity. In addition, since it is also regarded as a business, profit is more crucial. As it takes money to run a hospital, revenue management is one of the crucial components. Additionally, the outdated manual system cannot be used to track the same thing. An automated HMS that is tailored to the needs of the organization can assist in effectively resolving the issue. It provides precise and timely management and transactional data that gives you a complete view of how your company is operating.

Increased data security:

Hospitals, medical centers, and clinics rely on a manual system, which specialists have stressed and reinforced. It may result in increased data loss and theft when compared to automated methods. Installing a comprehensive and automatic HMS meant that your data was protected from illegal sources and accesses. We use a cutting-edge system with centralized controls, which is significant.

Make certain that we are not using a stand-alone home-grown one. Everything is managed via an access-controlled system, there is no room for error because information availability is determined by user rights. It is also one of the key reasons why high-quality, automated Hospital Management Systems are used in hospitals, rehabilitation centers, clinics, trauma centers, and nursing homes.

Errors are being reduced:

The hospital management system will help in the elimination of numerous sorts of mistakes caused by interventions such as missed billing, operational failure, clinical errors, cost leakages, missed appointments, and much more. Every procedure in the hospital administration system is automated, and several duties have been allocated to the software to complete without human interaction and precisely, considerably decreasing mistake.

If your hospital has a hospital management system, for example, an IPD patient's final bill amount can be easily generated because his reports and other sample bills are already billed and safe under the Patient unique Hospital ID, and the billing executive only needs to generate from the system and provide the statement to the patients.

Better Patient Care:

Improved labor productivity and patient data access lead to quicker and more accurate healthcare judgments. When the clinician requests the solution to be executed after obtaining the diagnostic report, faster support is necessary. All departments in hospitals are networked and integrated as a result of this automation, which increases both patient care quality and hospital turnover rates.

Quality and conformity:

Everyone favors HMS for their hospitals because it delivers coordinated and quick treatment, saves costs, decreases waiting time and readmissions, and enhances patient safety and clinical care.

The Hospital Management System promotes more collaboration among doctors, patients, hospital personnel, groups, and workflow managers. Portal solutions can unite legal items and applications from other portals, as well as manage a huge number of user requests, resulting in a chaotic information flow between patients, physicians, and staff.

Self-service for patients

HMS is assisting in making their system account for patients. Patients can make requests and reservations online. Receive the test results, consult with the medical specialist, and much more.

Consuming Time

Everything is planned with greater precision as services and collaborations improve in all possible ways. It saves all system clients time and provides them with up-to-date data.

7.3 Impact on Environment

By adopting an environmental management system, healthcare institutions may avoid pollution and assess and potentially resolve the lifecycle implications of their goods and services.

This will allow them to more effectively comply with applicable legislation, establish excellent community relations, deliver better healthcare services, and maintain industry competitiveness. A continual improvement framework that covers the following phases is what an environmental management system is. putting together an environmental policy

1. Assessing the organization's environmental impact.
2. Putting standards, programs, and procedures in place.
3. Changing behaviours and raising awareness.
4. Results measurement and auditing.
5. Evaluate progress and, if required, revise the environmental management system

An environmental management system, in essence, enables a facility to make its environmental goals a reality.

The advantages of implementing an environmental management system for the healthcare business and the general public are significant. Waste disposal, energy usage, and wastewater are major hospital expenses with severe environmental consequences. Many healthcare institutions are indifferent about their potential environmental effect. They may lack the necessary infrastructure or organization to handle environmental initiatives, or they may lack the necessary funding to undertake successful environmental management programs. Particularly now that man-made pollution has been connected to a rise in some types of human ailments, such as cancer, neurological, reproductive, and developmental consequences, and allergies, hospitals must take the lead in environmental awareness and preservation. Environmental conditions in hospitals should be carefully addressed and monitored, as shown by the recent development of environmental medicine as a medical specialty.

Environmental management systems can come from both internal and external sources, such as:

1. Management.
2. Employees whose jobs involve either security, money, or the environment.
3. Clients and guests.
4. Local government officials.
5. Lawmaker.
6. Services for ambulances and emergency response.

7.4 Ethical Aspects

The ethical facets of hospital administration unquestionably need greater consideration and examination than they have thus far gotten. Cloning, stem cell research on humans, and preimplantation diagnosis have far greater moral resonance right away than a field defined by words like "cost efficiency," "workflow management," "clinical pathways," and "process orientation," not to mention "total quality management," which seem dry and technical. However, the duties that hospital managers must do are rife with moral conundrums. All of these problems deserve careful ethical evaluation and analysis, including how to distribute available monies across departments, what clinical service portfolio to offer, and how to guarantee consistently high-quality services.

The institution receives support from the local community. Responsible decision-making on the part of hospital management is unquestionably an essential prerequisite if hospital staff are to be able to integrate ethics into their daily work, even though it is less likely that management decisions will directly harm individual patients than a mistake by medical staff. The patients being treated in the hospital, the personnel, and the community that supports the institution are at least three groups to which the hospital management must answer.

Regarding the first, healthcare institutions have a specific responsibility to their patients because of the special nature of the connection between patients and healthcare personnel

in the hospital context, which is often very uneven in terms of knowledge and authority. The use of advance directives, confidentiality of patient information, and respect for social, spiritual, and cultural beliefs are just a few of the mechanisms mentioned in the AHA's code of conduct as ways to protect patients' individual rights. Informed consent for diagnosis and therapy is also mentioned. Additionally, hospital managers are responsible for ensuring that all patients get consistent medical treatment.

Hospital management must make sure institutional mission statements and policies do not clash with, for example, the professional ethical codes of, doctors and nurses, with regard to the second category, organizational and personnel difficulties. In addition to this need, hospital managers ought to work hard to support efforts made by the medical personnel to integrate ethical principles into their everyday work. However, as anybody acquainted with the effects of medical rationing knows, a shortage of resources for healthcare raises ethical concerns, either in terms of resource allocation or a lack of working hours, which may limit staff participation in tasks that are crucial from an ethical perspective. The management of the hospital can utilize its authority to resolve these problems and promote moral principles. Like any other company, employees and medical personnel should have fair wages, benefits, and the chance to partake in ongoing education in order to consistently raise the standard of healthcare.

In terms of the third category, hospitals' community role necessitates structuring their health care provider to appropriately satisfy the demands of society in a given local environment in order to fulfill their public duties. Given that there are no apparent upper limits to medical products demand and supply, as evidenced by the extensively discussed World Health Organization (WHO) definition of health, some sensible criteria for resource distribution are necessary.

Health-care facilities must also devote special attention to the requirements and medical issues of vulnerable populations, regardless of whether their vulnerability originates from a specific lack of autonomy (as in the case of the very young, the old, or psychiatric patients) or from social marginalization (as in the case of the poor or inadequately insured). Finally, because health care institutions are supported by insurance costs or other types of government money, they are under a specific duty to make "fair and effective use of available health care delivery

resources." 1 When the public is responsible for the creation and operation of hospitals, hospital administration is accountable to the public. To summarize, hospital managers may affect a wide variety of organizational characteristics in order to promote ethical health care.

Although medical ethics largely focuses with disputes between patients and their health care personnel, it is obvious that the factors that shape the institutional framework in which such conflicts arise, such as the hospital, are equally important for finding suitable 67 solutions. The introduction and development of common and consensual codes of conduct at many hospitals today can be seen as evidence of a rising understanding of such a larger perception of variables influencing moral conflicts in medicine.

7.5 Sustainability Plan

Terms like sustainability, transparent management, environmental stewardship, and social responsibility are gaining traction in the business world, which is becoming more than just about money. Healthcare organizations' operations have a significant impact on society in three ways: economically, environmentally, and socially. Sustainability and these three characteristics must be specified in the organization's policy and strategy planning; top management should be intimately involved in the processes. These methods should be utilized in everyday operations rather than only for marketing goals. The organization must explicitly specify how this policy will govern procedures, rules, and activities.

Today's buzzwords include sustainability and resiliency. However, in order for healthcare institutions to achieve success in these areas, a practical, practicable approach that is compatible with budgetary constraints is essential. Incorporating sustainability strategically can help hospitals save money and become more efficient.

Sustainability involves system examination, including physical infrastructure and organizational processes, as well as the execution of policies based on effective budget planning. A healthcare institution must have dependable and robust engineering systems to ensure the safety of its patients.

While other sectors may be more eager to try out new technology and techniques to promote sustainability, healthcare facilities must look for solutions that have been proved to work in the

unique healthcare context. A strategic approach to health care sustainability might include variables such as restricted resources, demanding facility certification processes, and a low risk tolerance.

7.6 Conclusion

Hospitals are constantly being upgraded, modified, changed, or expanded. Because the condition of health care is continually evolving, hospitals regularly repurpose space and add equipment without fully addressing the cumulative consequences of minor changes on building infrastructure and operations. Frequently, projects are finished by one division of facility employees but maintained by another. Installed technology, as well as patient safety and satisfaction enhancements, may be detrimental to the building's core plant if not correctly conveyed — or, worse, may be entirely unknown to the operations and maintenance team. Healthcare facility workers are ideally positioned to explore, develop, and execute initiatives and practices that improve sustainability. This, however, demands both leadership and investment.

Cleaning and calibrating sensors, reprogramming the controls workstation, and replacing failing components are all low-cost options that can result in instant energy savings and comfort benefits. When combined, these fast improvements have the ability to drastically lower a facility's yearly energy expenses. Hospitals may reach their goals by considering sustainability strategically and investing in energy efficiency. Lowering operating costs, boosting patient happiness, and promoting community health can all be aided by optimizing facility operations.

CHAPTER 8

Conclusion

8.1 Conclusion

This hospital Management system is a web-based management system that allows users to consult with doctors with the help of appointments and hospitals can manage their staff like nurses, doctors etc. Both patients and hospitals can benefit from using this application. The web-based hospital management system is user-friendly to navigate easily and to find the resources they need for better health. Hope this will be a better solution than the traditional file systems and every hospital and clinic will adopt this web-based system for better time and less cost.

In today's modern medical environment, the web-based hospital management system is an essential component of the lifecycle. Numerous daily tasks are automated, and it makes user communication simple. The creation of a web-based hospital administration system is a fantastic opportunity to develop a clear, effective, and flexible healthcare delivery paradigm. The implementation of a web-based hospital management system project helps to manage financial and human resources, improve everyday operations, apply hospital rules, coordinate and communicate with users, and advertise hospital services. It also helps to save all kinds of data and records. The hospital administration, personnel, and patients' needs are all met by this positive ruling, which also makes it easier for them to communicate. It has evolved into the standard method of running the hospital. Numerous clinics are still creating new hospital management system project components despite having already experienced its advantages.

The major objective is to develop a web-based hospital management system that will allow patients and hospital administration to receive the reaction times they require. Our web-based system has a number of essential objectives, one of which is to foster a sense of community and positive interactions between patients and physicians.

8.2 Further Suggested Work

- Interface design will be updated
- More features and functionalities will be added
- Reliability of the web-based application should be increased

The system will have other features in the future like more modules like Nurse module, receptionist module, room module, pharmacist module etc. The interface can be changed. More animation, user friendly and simple and interactive features can be added. Managing the doctors, patient, nurses, receptionist, pharmacist can be controlled and monitored by the admin panel. The monthly reports like sales and managing shipments and ordering medicine online can also be added to the project for further development.

It is completely depended on the consumers and their needs. The more features and panels are needed to satisfy and made easy for the consumers can be added.

Reference

Conference/Journal Papers:

- [1] R. S. Adb-Ali, S. M. Al-Qaraawi², M. S. Croock, “Web Based E-Hospital Management System”, Iraqi Journal of Computers, Communications, Control & Systems Engineering (IJCCCE), Vol. 18, pp. 12-13, April 2018.
- [2] Ilo S.F, I. Abraham, A. Joyce .C “Designing A Web Based Hospital Management System for MOUAU Clinic”, International Journal of Trend in Research and Development, Volume 2(6), pp. 287-290, Nov-Dec 2015.
- [3] O.A Adebisi, D.A Oladosu, O.A Busari and Y.V Oyewola, “Design and Implementation of Hospital Management System”, International Journal of Engineering and Innovative Technology (IJEIT), Vol. 5, pp. 31-34, July 2015.
- [4] E. O. Boadu, G. k. Armah, “Role-Based Access Control (Rbac) Based in Hospital Management”, International Refereed Journal of Engineering and Science (IRJES), Vol. 3, pp. 53-65, July 2015.
- [5] J. K. Arthur, M. S. Jang, “The Analysis and Design of an Integrated Hospital Management System,” The Case of Motherlove Hospital, International Journal of Computer Science Issues, Vol 12, 2015.
- [6] Z. Chaczko, C. Chiu, A. K. Singh, “Smart Hospital Management System: An Integration of Enterprise Level Solutions Utilising Open Group Architecture Framework (TOGAF),” ISBN: 978-1-4244-5539-3/10/, 2010.
- [7] F. K. S. Gihan, A. M. Haneef, “Database and Flat File Validation Tool,” International Journal of Engineering Science Invention, Vol 2 , pp.38-42, 2013.
- [8] J. Hasan, “Integrated Hospital Information System (HIS) Special focus on BIRDEM Hospital (600 Beds),” E-Health Telecommunication Systems and Networks, Vol 2, pp 29-35, 2013.
- [9] H. D. Lieu, V. T. Dung, “Health Management,” Vietnam: Ministry of Health Medical Publisher, 2008.
- [10] H. N. Phuong, M. N. Anh, H. H. Anh, “Electronic Medical Record. Vietnam,” Ministry of Health Medical Publisher, 2008.
- [11] B. Premkumar, K. Kalpana, “E –Hospital Management & Hospital Information Systems – Changing Trends,” I.J. Information Engineering and Electronic Business, 1, pp 50-58, 2013.
- [12] Brown, M, and B P McCool. “Health care systems: predictions for the future.” © 1990 Aspen Publishers, Inc. Vol. 15: pp 87-94, 1990.
- [13] T. Garrido, B. Raymond, L. Jamieson, L. Liang, A. Wiesenthal,. “Making the business case for hospital information systems.” Journal of Healthcare Finance, Vol (2), pp 21–22, 2004
- [14] J. Ovretveit, T. Scott, et al. (2007a). “Implementation of electronic medical records in hospitals: two case studies.” Health Policy 84.

Books:

[15] M. Stauffer, Laravel: Up & Runnikng, 2nd Edition, O'Reilly, 2019, pp. 100-119.

Websites:

[16] Hospital Management System, available at << https://en.wikipedia.org/wiki/Hospital_information_System>>, last accessed on 15-08-2022 at 12:00 PM.

[17] Importance of the Hospital Management System, available at << <https://electronichealthreporter.com/importance-of-the-hospital-management-system/>>>, last accessed on 13-08-2022 at 01:00 AM.

[18] Challenges while implementing the Hospital Management System, available at <<<https://mocdoc.in/blog/challenges-while-implementing-the-hospital-management-system>>>last accessed on 13-08-2022 at 02:00 AM.

[19] How to Develop a Hospital Management System, available at <<<https://cprimestudios.com/blog/how-develop-hospital-management-system>>>last accessed on 05-06-2022 at 03:00 PM.

[20] J. Bisbal, D. Berry, “An Analysis framework for Electronic Health Record System,” available at<<http://www.tecn.upf.es/~jbisbal/publications/me09-01-0002_Bisbal.pdf>>last accessed on: 06-06-2022 at 08:00 AM.

[21] R. Ranjan, How to secure your cloud database in an insecure world, available: <<<https://www.ibm.com/blogs/cloud-computing/2014/10/secure-cloud-database-insecure-world/>>> last accessed on: 06-06-2022 at 11:00 AM.

[22] Alice Kok Thailand: successful e-health system lauded. FutureGov. available at: <<<http://www.futuregov.asia/articles/2009/may/14/thai-and-successful-e-health-system-increase-health/>>> last accessed on: 05-06-2022 at 06:00 PM.

[23] Belgium Federal Public Service – FPS report (2002). Recommendations and quality criteria for hospital information systems. Available at <<www.health.belgium.be/filestore/8054405/his_v1s_e_n_8054405_en.pdf>>last accessed on: 21-04- 2022 at 04:45 PM.

[24] Björn Schreiweis. Modelling the Hospital Information System of the Karolinska University Hospital in Stockholm. University of Heidelberg, Heilbronn University and Karolinska Institutet. Available at <<http://ki.se/content/1/c6/10/46/20/Diplomarbeit_Bjo_ern_Schreiweis.pdf>>last accessed on: 21-04- 2022 at 06:45 PM.

[25] Bradley Malin. Guidance on De-identification of Protected Health Information. Office for Civil Rights., U.S.Department of Health & Human Services. Available at <<http://www.hhs.gov/ocr/privacy/hipaa/understanding/coveridentities/Deidentification/hhs_deid_guidance.pdf>>last accessed on: 20-05- 2022 at 06:45 PM.

[26] Fujisoft (2012). Fujisoft Hospital Solutions (Japan). available at <http://www.fsi.co.jp/e/solutions/hospital_solutions/index.html>>last accessed on: 15-06- 2022 at 04:30 PM.

APPENDICES

APPENDIX 'A'

CODE

MODEL

Appointment.php

```
Hospital > app > Models > Appointment.php
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Notifications\Notifiable;
8
9  class Appointment extends Model
10 {
11     use HasFactory;
12
13     use Notifiable;
14 }
15
```

Doctor.php

```
Hospital > app > Models > Doctor.php
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Doctor extends Model
9  {
10     use HasFactory;
11 }
12
```

User.php

```
Hospital > app > Models > User.php
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Contracts\Auth\MustVerifyEmail;
6  use Illuminate\Database\Eloquent\Factories\HasFactory;
7  use Illuminate\Foundation\Auth\User as Authenticatable;
8  use Illuminate\Notifications\Notifiable;
9  use Laravel\Fortify\TwoFactorAuthenticatable;
10 use Laravel\Jetstream\HasProfilePhoto;
11 use Laravel\Sanctum\HasApiTokens;
12
13 class User extends Authenticatable implements MustVerifyEmail
14 {
15     use HasApiTokens;
16     use HasFactory;
17     use HasProfilePhoto;
18     use Notifiable;
19     use TwoFactorAuthenticatable;
20
21     /**
22      * The attributes that are mass assignable.
23      *
24      * @var string[]
25      */
26     protected $fillable = [
27         'name',
28         'email',
29         'phone',
30         'usertype',
31         'address',
32         'password',
33     ];
```

Hospital > app > Models > User.php

```
34
35     /**
36     * The attributes that should be hidden for serialization.
37     *
38     * @var array
39     */
40     protected $hidden = [
41         'password',
42         'remember_token',
43         'two_factor_recovery_codes',
44         'two_factor_secret',
45     ];
46
47     /**
48     * The attributes that should be cast.
49     *
50     * @var array
51     */
52     protected $casts = [
53         'email_verified_at' => 'datetime',
54     ];
55
56     /**
57     * The accessors to append to the model's array form.
58     *
59     * @var array
60     */
61     protected $appends = [
62         'profile_photo_url',
63     ];
64 }
65
```

APPENDIX 'B'

Controllers

AdminController.php

```
Hospital > app > Http > Controllers > AdminController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  use App\Models\Doctor;
8
9  use App\Models\Appointment;
10
11 use Illuminate\Support\Facades\Auth;
12
13 use Notification;
14
15 use App\Notifications\SendEmailNotification;
16
17 class AdminController extends Controller
18 {
19     public function addview()
20     {
21         if(Auth::id())
22         {
23             if(Auth::user()->usertype==1)
24             {
25                 return view('admin.add_doctor');
26             }
27             else{
28                 return redirect()->back();
29             }
30         }
31         else
32         {
33             return redirect('login');
34         }
35     }
36 }
37
```

Hospital > app > Http > Controllers > AdminController.php

```
39 public function upload(Request $request)
40 {
41     $doctor=new doctor;
42
43     $image = $request->file;
44
45     $imagename = time().'.'.$image->getClientOriginalExtension();
46
47     $request->file->move('doctorimage',$imagename);
48
49     $doctor->image = $imagename;
50
51     $doctor->name = $request->name;
52
53     $doctor->phone = $request->number;
54
55     $doctor->room = $request->room;
56
57     $doctor->speciality = $request->speciality;
58
59     $doctor->save();
60
61     return redirect()->back()->with('message','Doctor Added Successfully');
62 }
63
64 public function showappointment()
65 {
66     $data=appointment::all();
67
68     return view('admin.showappointment', compact('data'));
69 }
70
```

```
72 public function approved($id)
73 {
74     $data=appointment::find($id);
75
76     $data->status='approved';
77
78     $data->save();
79
80     return redirect()->back();
81
82 }
83
```



```

85     public function canceled($id)
86     {
87         $data=appointment::find($id);
88
89         $data->status='Canceled';
90
91         $data->save();
92
93         return redirect()->back();
94     }
95
96
97
98     public function showdoctor()
99     {
100         $data=doctor::all();
101
102         return view('admin.showdoctor',compact('data'));
103     }
104
105
106
107     public function deletedoctor($id)
108     {
109         $data=doctor::find($id);
110
111         $data->delete();
112
113         return redirect()->back();
114     }
115
116

```

```

118     public function updatedoctor($id)
119     {
120         $data=doctor::find($id);
121
122         return view('admin.update_doctor',compact('data'));
123     }
124
125
126     public function editdoctor(Request $request, $id)
127     {
128         $doctor=doctor::find($id);
129
130         $doctor->name=$request->name;
131
132         $doctor->phone=$request->phone;
133
134         $doctor->speciality=$request->speciality;
135
136         $doctor->room=$request->room;
137
138         $image=$request->file;
139
140         if($image){
141             $imagenname=time().'.'.$image->getClientOriginalExtension();
142
143             $request->file->move('doctorimage',$imagenname);
144
145             $doctor->image=$imagenname;
146
147         }
148

```

```

156 public function emailview($id)
157 {
158     $data=appointment::find($id);
159     return view('admin.email_view',compact('data'));
160 }
161
162 public function sendemail(Request $request,$id)
163
164 {
165     $data = appointment::find($id);
166     $details=[
167         'greeting' => $request->greeting,
168         'body' => $request->body,
169         'actiontext' => $request->actiontext,
170         'actionurl' => $request->actionurl,
171         'ending' => $request->ending
172
173
174     ];
175
176     Notification::send($data,new SendEmailNotification($details));
177
178     return redirect()->back()->with('message','Sent Successfully');
179 }
180
181 public function showappoint()
182 {
183     $data=appointment::all();
184
185     return view('doctorhome.showappointment', compact('data'));
186 }
187

```

```

188 public function showdoctor1()
189 {
190     $data=doctor::all();
191
192     return view('doctorhome.showdoc',compact('data'));
193 }
194
195 }
196

```

Controller.php

```
Hospital > app > Http > Controllers > Controller.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Foundation\Auth\Access\AuthorizesRequests;
6  use Illuminate\Foundation\Bus\DispatchesJobs;
7  use Illuminate\Foundation\Validation\ValidatesRequests;
8  use Illuminate\Routing\Controller as BaseController;
9
10 class Controller extends BaseController
11 {
12     use AuthorizesRequests, DispatchesJobs, ValidatesRequests;
13 }
14
```

HomeController.php

```
Hospital > app > Http > Controllers > HomeController.php
1  <?php
2  namespace App\Http\Controllers;
3
4  use Illuminate\Http\Request;
5
6  use Illuminate\Support\Facades\Auth;
7
8  use App\Models\User;
9
10 use App\Models\Doctor;
11
12 use App\Models\Appointment;
13
14 class HomeController extends Controller
15 {
16     public function redirect()
17     {
18         if(Auth::id())
19         {
20             if(Auth::user()->usertype=='0')
21             {
22                 $doctor = doctor::all();
23                 return view('user.home', compact('doctor'));
24             }
25             elseif(Auth::user()->usertype=='2'){
26                 return view('doctorhome.home');
27             }
28             else{
29                 return view('admin.home');
30             }
31         }
32         else
33         {
34             return redirect()->back();
35         }
36     }
37 }
```

```
Hospital > app > Http > Controllers > HomeController.php
38
39 public function index()
40 {
41     if(Auth::id())
42     {
43         return redirect('home');
44     }
45     else{
46         $doctor = doctor::all();
47         return view('user.home', compact('doctor'));
48     }
49 }
50
51
52 public function appointment(Request $request)
53 {
54     $data = new appointment;
55     $data->name=$request->name;
56     $data->email=$request->email;
57     $data->date=$request->date ;
58     $data->phone=$request->number;
59     $data->message=$request->message;
60     $data->doctor=$request->doctor;
61     $data->status='In Progress';
62
63     if(Auth::id())
64     {
65         $data->user_id=Auth::user()->id;
66     }
67
68     $data->save();
69
70     return redirect()->back()->with('message','Appointment Request Succesful . We will contact soon');
71
72 }
```

```
74 public function myappointment(){
75     if(Auth::id())
76     {
77         $userid=Auth::user()->id;
78
79         $appoint=appointment::where('user_id',$userid)->get();
80         return view('user.my_appointment',compact('appoint'));
81     }
82     else{
83         return redirect()->back();
84     }
85 }
86
87 public function cancel_appointment($id)
88 {
89     $data=appointment::find($id);
90
91     $data->delete();
92
93     return redirect()->back();
94 }
95
96
97
98 public function doctorlist(){
99
100     $doctor = doctor::all();
101
102     return view('user.doctorlist', compact('doctor'));
103 }
104
105 public function aboutus(){
106
107     return view('user.about_us');
108 }
109 }
```

```
110     public function newsdetails(){
111     }
112         return view('user.news_details');
113     }
114
115     public function contact(){
116     }
117         return view('user.contact');
118     }
119
120     public function payment(){
121     }
122         return view('user.payment');
123     }
124     public function showappoint(){
125     }
126         $data=appointment::all();
127
128         return view('doctorhome.showappoint',compact('data'));
129     }
130
131     // public function showdoc(){
132     }
133         //     $doctor = doctor::all();
134
135         //     return view('doctorhome.showdoc', compact('doctor'));
136         // }
137
138
139 }
140
141
```

APPENDIX 'C'

web.php

```
Hospital > routes > web.php
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\HomeController;
5  use App\Http\Controllers\SslCommerzPaymentController;
6  use App\Http\Controllers\AdminController;
7  /*
8  |-----
9  | Web Routes
10 |-----
11 |
12 | Here is where you can register web routes for your application. These
13 | routes are loaded by the RouteServiceProvider within a group which
14 | contains the "web" middleware group. Now create something great!
15 |
16 */
17
18
19 Route::get('/',[HomeController::class,'index']);
20
21 Route::get('/home',[HomeController::class,'redirect']->middleware('auth','verified'));
22
23 Route::middleware([
24     'auth:sanctum',
25     config('jetstream.auth_session'),
26     'verified'
27 ]->group(function () {
28     Route::get('/dashboard', function () {
29         return view('dashboard');
30     }->name('dashboard');
31 });
32
33 Route::get('/add_doctor_view',[AdminController::class,'addview']);
34
35 Route::post('/upload_doctor',[AdminController::class,'upload']);
```

Hospital > routes > web.php

```
35 Route::post('/upload_doctor',[AdminController::class,'upload']);
36
37 Route::post('/appointment',[HomeController::class,'appointment']);
38
39 Route::get('/myappointment',[HomeController::class,'myappointment']);
40
41 Route::get('/cancel_appointment/{id}',[HomeController::class,'cancel_appointment']);
42
43 Route::get('/showappointment',[AdminController::class,'showappointment']);
44
45 Route::get('/approved/{id}',[AdminController::class,'approved']);
46
47 Route::get('/canceled/{id}',[AdminController::class,'canceled']);
48
49 Route::get('/showdoctor',[AdminController::class,'showdoctor']);
50
51 Route::get('/showdoctor1',[AdminController::class,'showdoctor1']);
52
53 Route::get('/deleteddoctor/{id}',[AdminController::class,'deleteddoctor']);
54
55 Route::get('/updateddoctor/{id}',[AdminController::class,'updateddoctor']);
56
57 Route::post('/editdoctor/{id}',[AdminController::class,'editdoctor']);
58
59 Route::get('/emailview/{id}',[AdminController::class,'emailview']);
60
61 Route::post('/sendemail/{id}',[AdminController::class,'sendemail']);
62
63 Route::get('/doctorlist',[HomeController::class,'doctorlist']);
64
65 Route::get('/aboutus',[HomeController::class,'aboutus']);
66
67 Route::get('/newsdetails',[HomeController::class,'newsdetails']);
68
69 Route::get('/contact',[HomeController::class,'contact']);
70
```

```
73 Route::get('/showappoint',[HomeController::class,'showappoint']);
74
75 //Route::get('/showdoc',[HomeController::class,'showdoc']);
76
77 Route::get('/example1',[SslCommerzPaymentController::class,'exampleEasyCheckout']);
78 Route::get('/example2',[SslCommerzPaymentController::class,'exampleHostedCheckout']);
79
80 Route::post('/pay',[SslCommerzPaymentController::class,'index']);
81 Route::post('/pay-via-ajax',[SslCommerzPaymentController::class,'payViaAjax']);
82
83 Route::post('/success',[SslCommerzPaymentController::class,'success']);
84 Route::post('/fail',[SslCommerzPaymentController::class,'fail']);
85 Route::post('/cancel',[SslCommerzPaymentController::class,'cancel']);
86
87 Route::post('/ipn',[SslCommerzPaymentController::class,'ipn']);
88 //SSLCOMMERZ END
```

Turnitin Originality Report

Processed on: 08-Sep-2022 12:34 +06
 ID: 1894962510
 Word Count: 12809
 Submitted: 1

Similarity Index
28%

Similarity by Source

Internet Sources: 20%
 Publications: 4%
 Student Papers: 23%

183-15-12037 By Istak Ahmed

2% match (Internet from 03-Dec-2020)

<https://mocdoc.in/blog/what-are-the-benefits-of-the-hospital-management-system>

1% match (Internet from 09-Jan-2022)

<http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/6655/172-15-9705%20%2822%25%29.pdf?isAllowed=y&sequence=1>

1% match (Internet from 14-Dec-2021)

http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/6472/181-15-1740%2018_.pdf?isAllowed=y&sequence=1

1% match (Internet from 14-Feb-2022)

<http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/6783/153-15-6621%20%2c%2020%25.pdf?isAllowed=y&sequence=1>

1% match (Internet from 05-Jan-2022)

http://dspace.daffodilvarsity.edu.bd:8080/bitstream/handle/123456789/5604/141-15-3097%20%286_.%29.pdf?isAllowed=y&sequence=1

1% match ()

[Biller-Andorno, Nikola, Lenk, Christian, Leititis, J., "Ethics, FRM, and hospital management", "BMJ", 2004](#)

1% match (student papers from 22-May-2022)

[Submitted to University of Peloponnese on 2022-05-22](#)

1% match ()

[Abdul-Aziz, Azlihan, Koronios, Andy, Gap, Jing, Suhaizan Sulong, Muhammad, "A Methodology for the Development of Web-based Information Systems: Web Development Team Perspective", AIS Electronic Library \(AISeL\), 2012](#)

1% match (student papers from 10-Oct-2021)

[Submitted to The University of the South Pacific on 2021-10-10](#)

1% match (student papers from 05-Sep-2022)

[Submitted to University of Sunderland on 2022-09-05](#)

1% match (student papers from 24-Apr-2022)

[Submitted to Middle East College of Information Technology on 2022-04-24](#)

1% match (student papers from 15-Apr-2018)