|  |  |
| --- | --- |
| A picture of a winding road and trees  **MEDICARE SYSTEM**  **STREAMLINING HEALTHCARE OPERATIONS** | SUbMITTED BY GROUP 5 |

**Abstract**

Hospital management systems (HMS) are comprehensive software solutions designed to streamline and automate various administrative, operational, and clinical processes in healthcare facilities. This abstract provides an overview of hospital management systems, highlighting their key features and benefits. It discusses the modules typically included in an HMS, such as patient registration, appointment scheduling, electronic health records (EHR), billing and invoicing, inventory management, and reporting. The abstract also emphasizes the importance of a user-friendly interface, data security, interoperability, and scalability in a hospital management system. Furthermore, it emphasizes the role of HMS in enhancing patient care, optimizing resource utilization, improving operational efficiency, and facilitating decision-making. Overall, hospital management systems serve as essential tools for modern healthcare organizations, enabling them to deliver high-quality care while ensuring effective management of hospital operations.

Contents

[**Introduction** 7](#_Toc138102531)

[**Project description** 7](#_Toc138102532)

[**Problem Definition** 8](#_Toc138102533)

[Objectives 9](#_Toc138102534)

[Scope of the project 9](#_Toc138102535)

[Key advantages of hospital management software 10](#_Toc138102536)

[**Limitation of the system** 11](#_Toc138102537)

[**System Designing** 12](#_Toc138102538)

[**Introduction** 12](#_Toc138102539)

[**Design Process** 12](#_Toc138102540)

[**The System Design Process** 16](#_Toc138102541)

[**Tools for the designing of the Hospital management** 18](#_Toc138102542)

[25](#_Toc138102543)

[26](#_Toc138102544)

[**Existing Software Development Tools.** 27](#_Toc138102545)

[**Application Model** 28](#_Toc138102546)

[**Importance of system design** 29](#_Toc138102547)

[**IMPLEMENTATION** 30](#_Toc138102548)

[**Introduction** 30](#_Toc138102549)

[**System implementation strategy** 30](#_Toc138102550)

[**Development specification** 31](#_Toc138102551)

[**Code details and code efficiencies** 32](#_Toc138102552)

[**User Interface Implementation** 35](#_Toc138102553)

[**Doctors dashboard** 37](#_Toc138102554)

[**Admin dashboard** 37](#_Toc138102555)

[**Testing** 39](#_Toc138102556)

[**Introduction** 39](#_Toc138102557)

[**Results** 40](#_Toc138102558)

[**Evaluation of Software Solutions and Quality** 47](#_Toc138102559)

[**SUMMARY, CONCLUSION AND RECOMMENDATIONS** 51](#_Toc138102560)

[**Introduction** 51](#_Toc138102561)

[**Conclusion** 51](#_Toc138102562)

[**Future works** 51](#_Toc138102564)

[**Conclusion:** 53](#_Toc138102565)

[**QUESTIONARES** 54](#_Toc138102566)

**TABLES OF FIGURES**

[Figure 1: Scrum Methodology of the MediPro Web System 10](#_Toc138098459)

[Figure 2 : Daily Schedule of the Scrum and their activities 12](#_Toc138098460)

[Figure 3: diagram showing the Use case 16](#_Toc138098461)

[Figure 4: Diplays the Sequence diagram for the application 17](#_Toc138098462)

[Figure 5: Display deployment application 18](#_Toc138098463)

[Figure 6: Diagram showing Flowchart 20](#_Toc138098464)

[Figure 7 : Context level DFD Hospital Management System 21](#_Toc138098465)

[Figure 8 : Dfd showing Administrator 22](#_Toc138098466)

[Figure 9: Diagram Showing Level 1 DFD 23](#_Toc138098467)

[Figure 10: Diagram Showing Level 2 DFD 23](#_Toc138098468)

[Figure 11: Showing Landing page 32](#_Toc138098469)

[Figure 12: Showing loging page 33](#_Toc138098470)

[Figure 13: Showing Doctors Portal 34](#_Toc138098471)

[Figure 14: Showing the Administrators Portal 34](#_Toc138098472)

[Figure 15: doctor portal after enterring correct credential 37](#_Toc138098473)

[Figure 16: Wrong credential alert 38](#_Toc138098474)

[Figure 17: Registering new user 38](#_Toc138098475)

[Figure 18: Alert showing patient data added 39](#_Toc138098476)

[Figure 19: generate reports on patient visits and treatment 39](#_Toc138098477)

[Figure 20: Adding test result 40](#_Toc138098478)

[Figure 21: alert showing Added successfully 40](#_Toc138098479)

[Figure 22: The results after the test has been performed 41](#_Toc138098480)

[Figure 23: Updated patients details 41](#_Toc138098481)

**ABBREVIATIONS & ACRONYMS**

ADMIN - Administrator

FAQ - Frequently Asked Questions

GUI - Graphical User Interface

HTML - Hyper Text Markup Language

ICT - Information and Communication Technology

IS - Information System

Lab - Laboratory

LAN - Local Area Network

MIS - Management Information System

PHP - Hypertext Pre-Processor

RAM - Random Access Memory

RM - Records Management

RMS - Records Management System

SQL - Structured Query Language

XP - Extreme Programming

HMS - Hospital Management System

## **Introduction**

### **Project description**

Our software development group embarked on a challenging project to develop a comprehensive Hospital Management System (HMS) aimed at improving the efficiency and effectiveness of hospital operations. The HMS was developed to streamline administrative tasks, enhance patient care, and facilitate smooth communication among different departments within the hospital.

With a focus on creating, a user-friendly interface and incorporating essential features, our group set out to create a robust and scalable system that would meet the specific needs of hospitals. Throughout the development process, we employed agile methodologies to ensure continuous collaboration, iterative improvements, and responsiveness to feedback from stakeholders, including hospital staff and administrators.

The primary objective of our HMS is to centralize and automate various hospital processes, such as patient registration, appointment scheduling, billing, and reporting. By implementing this system, we aim to reduce paperwork, minimize errors, and enhance the overall efficiency of hospital management.

Our group consisted of five dedicated members, each with unique skills and areas of expertise. John led the architectural design and database development, Sarah focused on gathering requirements and designing the user interface, Lisa was responsible for testing and quality assurance, and Michael contributed to system implementation, user acceptance testing, and deployment. Lisa was the project lead that guided, made sure that project timelines were met on time and system was developed as per the SRS document.

Over the course of the project, we collaborated closely with hospital staff, conducting interviews and incorporating their feedback to ensure that the HMS met their specific needs. This collaborative approach enabled us to create a customized system that aligns with the workflow and requirements of the hospital, ultimately enhancing productivity and patient care.

In this report, we will delve into the individual contributions of each group member, highlighting their roles and responsibilities throughout the development process. By highlighting the collective efforts of our team, we aim to demonstrate how each member's expertise contributed to the successful creation of the Hospital Management System.

### **Problem Definition**

The absence of a well-established information system to support patients and staff has caused inconveniences due to the limitations of the current system, particularly its heavy reliance on paperwork. Paper-based files occupy a significant amount of office space and hinder the recording, processing, and retrieval of patient details. Storing physical documents poses challenges as different document types require varying sizes of folders or storage spaces. Medical records typically exist in physical forms like X-ray films, paper documents, and photographs, often with different sizes and shapes. Managing physical records necessitates substantial storage space.

Additionally, handling paper records involves time-consuming and costly tasks such as copying, faxing, collecting, and transporting them to a single or multiple locations for review by healthcare providers. The use of file cabinets, where individual patient cards are enclosed in files, further complicates the process of locating and processing records.

Moreover, paper records are prone to being lost, misplaced, or becoming illegible. Implementing electronic health record technology would address these issues and bring significant improvements to patient care, enhancing health and safety. This project will execute the design and implementation of a web-based health information system to achieve these objectives. While there are existing health information systems limited to local area networks and accessible only by selected staff, they are not integrated with other hospital information systems or health information systems. Furthermore, patient information cannot be accessed outside the institution unless a Virtual Private Network (VPN) is used. The goal of this project is to develop a patient and health workers database system that can be deployed over a network.

### Objectives

The hospital management system is developed to the help in the daily running operation of the the hospital, dispensary, and any healthcare facilities. The activities such as storing doctors list, generation of the hospital/doctors/patient reports and tracking the activities in the hosipital and etc. It is designed to meet the following objectives:

* To computerize all details stored in the system regarding patient details & hospital details.
* The system should store the history of the patient visits to the hospital for easy decision making and tracking of patients ailments.
* The system should work as one integrated whole having different modules that interact with each other.
* Scheduling the appointment of patient with doctors to make it convenient for both.
* Scheduling the services of specialized doctors and emergency properly so that facilities provided by hospital are utilized fully in effective and efficient manner.
* If the medical store issues medicines to patients, it should reduce the stock status of the medical store and vice-versa.
* It should be able to handle the test reports of patients conducted in the pathology lab of the hospital.
* The inventory should be updated automatically whenever a transaction is made.
* The information of the patients should be kept up to date and there record should be kept in the system for historical purposes.

### Scope of the project

The developed hospital management system is available to be in use by any clinic, dispensary and healthcare centers to get, store and process information about the patients, for future reference and for accounting purposes. Most clinics and dispensary find it hard to purchase and use the available software in the market because the software licenses are quite expensive and are barley customized to meet their requirements. Most of these healthcare facilities resort to the manual way of using paperwork, which does not efficient and effective and is does guarantee for the security of the sensitive information. The system is to reduce time of used by the patients when moving up and down in search for the lab results and reduce room for errors since information is recorded to the patient’s unique number.

### Key advantages of hospital management software

* **Easy access to data**. Hospital management software stores all patient data in an electronic format: medical history, methods of treatment, tests results. The system ensures that these records are available at any time so doctors don’t have to make special requests or wait long for answers.
* **Cost-efficiency**. With a patient management system, hospitals don’t need to hire more staff, allowing them to keep down labor costs.
* **Reduced scope for error**. What is the cost of errors in medical records? It can be the difference between dying and making a full recovery. Yearly, such errors contribute to [200,000 deaths](https://www.techjockey.com/blog/why-hospitals-need-invest-hospital-management-system). An HMS reduces the number of errors caused by illegible handwriting and missing or duplicate records.
* **Better patient experience**. Hospital management software works not only for you. An HMS is patient-oriented, making interactions with your medical center/hospital less stressful for patients.
* **Better staff interactions**. Hospital software improves cooperation and teamwork. For example, laboratory staff in charge of medical tests can use this software to share test results with a doctor in minutes.
* **Improved revenue management**. Even though hospitals serve people, profitability is important. It requires a fortune to run a hospital. An HMS brings revenue management to a new level. With hospital management software, managers can receive fast and accurate financial reports. This gives them a feeling of control over how the business is running.

### **Limitation of the system**

1. Its accessibility is very dependent on the availability of network. Lack of internet connection will hinder one from accessing the application on the computer. Therefore, it will not be used in areas that have poor network connection.
2. The users are required to fill in the correct information to avoid errors i.e. if the lab technician input wrong figures, the doctor may will not make the right decision regarding the ailment of the patient and sometimes result to repetition of same test hence wastage of resources.
3. The fear of data breaches.
4. The lack of IT-friendly medical staff members.
5. Difficulties associated with the transition from paper-based to digital processes
6. Compliance with legal regulations.

## **System Designing**

### **Introduction**

This section will explain in detail the architecture of the SCRUM methodology to optimize and streamline the process of administrative control of medical records. The Web System is developed with the PHP language accompanied by the JavaScript language using the base manager of MYSQL data, likewise it is designed in the Figma and Microsoft Visio tools in the designing of the system for the administrative control of the medical records in a Health care simulate the necessary interfaces.

### **Design Process**

Design is a crucial step in the software development process. It aids in the transformation of the user requirements into a software system that meets those requirements of the users. Design involves choosing suitable data structures, algorithms, and interfaces to implement the system. It also involves trade-offs among performance, memory usage, and other factors. Good design can make a significant difference in the quality and usability of a software system. (Simplilearn, 2023).

#### **Scrum process**

The development of this Medipro Hospital management System, we used agile technique: Scrum Methodology. Scrum is an iterative and incremental framework that helps teams deliver high-quality products in a timely manner. It is based on the principles of transparency, inspection, and adaptation, and provides a flexible and collaborative approach to project management. Scrum is a simple framework for effective team collaboration on complex software projects.  The Scrum Framework poster provides a graphical view of how Scrum is implemented at a team level within an organization as shown in the *fig 1* below.

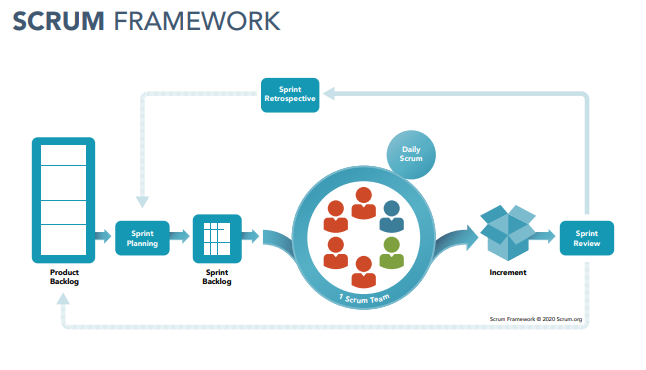


Figure 1: Scrum Methodology of the MediPro Web System

The system was guided by the scrum phases:

1. **Phase I. Product Backlog**

The product backlog is a prioritized list of user stories, features, and requirements that need to be implemented in the project. In this phase the requirements of the most relevant processes of the project will be gathered together with the client to form the design of the web system. Likewise, the administrators or users who will use this system and the use cases that will be made will also be identified. We used this as a base of the primary source of work for the development team.

The following is the list of the system user cases that we used to capture the requirement of the users.s

|  |  |
| --- | --- |
| **No.** | **User stories** |
| 1 | As an administrator, I need to register all patients in the database to have better control of patient treatments and thus be able to make a good diagnosis. |
| 2 | As a doctor, I need to be able to make a complete record of all the treatments and medications of each of the patients registered. |
| 3 | As an administrator, I need the medical consultations that were carried out in order to have better control and to have daily reports of the medical consultations carried out |
| 4 | As an administrator, I need to have all the medical records and personal data of each one of them to have better control of the history to finally have a report of the diseases that are frequented in certain districts |
| 5 | As aa administrator, I need the medication history to have better control of the quantities sold during the day, week, etc. |
| 6 | As a doctor, I need access to evaluate the medical appointments to make daily reports of the amounts of daily consultations that are performed. |
| 7 | As a pharmacist, I want to be able to view all the drugs present in the system and the remaining quantity |
| 8 | As a lab technician I was to be able to view the doctors prescribed test on patient and be able to send the test results to the doctor. |

Table 1: Table showing the User stories

1. **Phase II. Sprint Backlog/Planning**

During sprint planning, the product owner and the development team collaborate to determine which items from the product backlog will be worked on in the upcoming sprint. The team defines the sprint goal and selects the user stories they commit to completing during the sprint.

We came up on with the following sprint in development process:

1. **Sprint 1: Patient records control**

This module will be carried out by modeling the design and construction, in which doctors will be able to enter through a login, being located in the patient records section, in order for them to record the patient data. The module should be able to register new users and keep track of their history visits.

1. **Sprint 2:** **Control of medical treatments and equipment.** Once the patient is selected, the type of treatment according to the patient needs is chosen. Tracking of the equipment in the hospital for the accountability purposes
2. **Sprint 3.** **Medical consultation control**. Once the treatment has been registered, the next appointment will be scheduled. The information should easily flow from the reception /nurse who records the details of the patient, to the doctor to the lab technician and the pharmacist.
3. **Sprint 4: Control of the medical history of patients**. This module will record the details of the treatments of each of the patients, as well as personal information every time the patients visits the hosipital.
4. **Phase III. Daily Scrum/Stand-up**

They are short meeting held by development team every day or certain defined period. The meetings that were held weekly to supervise that the project was carried out as the as per the requirement. The team members shared on their progress, plans for the days and any challenges they faced. This promoted collaboration, transparency and quick problem solving.

We designed the schedule and assigned priority to members(scrums), the image below shows the:

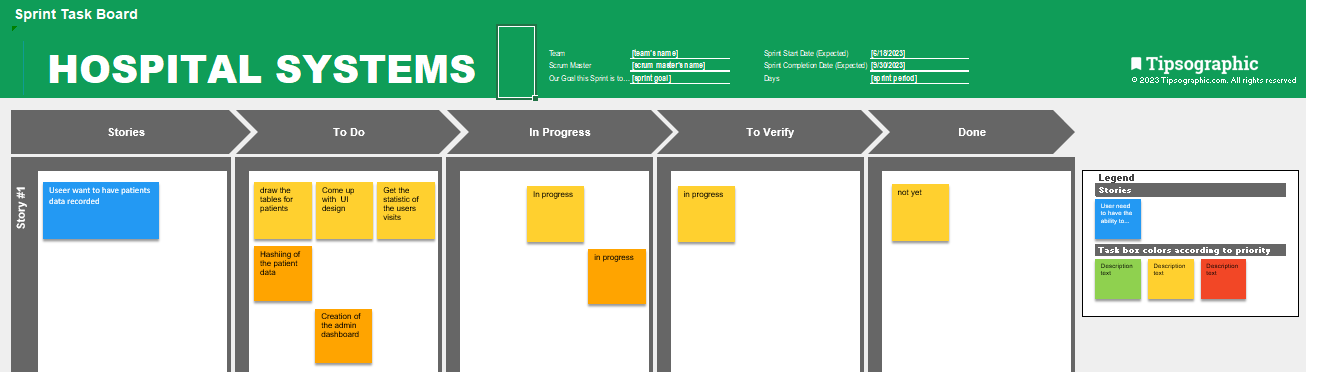


Figure 2 : Daily Schedule of the Scrum and their activities

1. **Phase IV. Sprint Review**

In this phase the new user stories or new requirements are presented to improve the system, it is completed by making the latest revisions to the system.

1. **Phase V. Sprint Retrospective**

Lastly, new improvements for the administrative control web system for a hospital are identified. Figure 1. It shows the steps performed in the SCRUM methodology to expedite the administrative control process.

In addition, in this phase, new interfaces were designed for the interaction between the user and the system, so that they can view their information or make inquiries if needed.

Throughout the project, I undertook the role of a dedicated product owner to represents the stakeholders and ensures that their requirements are effectively communicated to the development team.

By applying the Scrum framework to the design and analysis of a hospital management system, it promoted agility, collaboration, and iterative development, resulting in a system that met the specific needs of the healthcare environment.

### **The System Design Process**

The purpose of the Design phase is to plan a solution for the problem specified by the requirements. System design aims to identify the modules that should be in the system, the specification of those modules and how they interact with each other to produce the results. The goal of the design process is to produce a model that can be used later to build that system. The produced model is called design of the system. System design is the process of defining the architecture, components, modules, interfaces and data for a system to satisfy specified requirements.

System design develops the architectural details required to build a system or product. The process used in the design encompasses the following activities:

• Partition the analysis model into subsystems.

• Identify concurrency that is dictated by the problem.

• Allocate subsystems to processors and tasks.

• Develop a design for the user interface.

• Choose a basic strategy for implementing data management.

• Identify global resources and the control mechanisms required to access them.

• Design an appropriate control mechanism for the system, including task management.

• Consider how boundary conditions should be handled.

• Review and consider trade-offs.

1. **Design of the Input Details**

Input design is a part of overall system design, requires the very careful analysis of the input data items. The goal of the input design is to make the data entry easier, logical and free from errors. The user controls input data. The commonly used input, output devices are mouse, keyboard and the visual display unit (Monitor). The well designed, well organized screen formats are used to acquire the inputs. The data accepted is stored on database file. Our system is classified into subsystem such as

* + 1. Admission
    2. Staff Details
    3. Billing
    4. Consultation Details
    5. Equipment management Details
    6. Pharmaceutical Management
    7. Data statistics Report

1. **Design of the Output**

Details Output is the most important and direct source of information to the user. Efficient & intelligent output design improves the system relationships with the users and helps in decision-making. The output is collected in order to help the user to make a wise decision. It is normally presented in oral and paper form.

### **Tools for the designing of the Hospital management**

Software designing tools are applications or platforms that aid in the design and creation of software systems. These tools facilitate various aspects of the software development life cycle, including requirements gathering, system modeling, architecture design, user interface design, and code generation. The following are tools used for designing:

**UML (Unified Modeling Language) Tools**

UML tools such as Visual Paradigm, Enterprise Architect, Lucidchart, and PlantUML allow developers to create visual diagrams and models to represent the structure, behavior, and relationships of the software components. These diagrams include class diagrams, sequence diagrams, state diagrams, and more.

The primary goals in the design of the UML were:

* 1. Provide users with a ready-to-use, expressive visual modelling language so they can develop and exchange meaningful models.
  2. Provide extensibility and specialization mechanisms to extend the core concepts.
  3. Be independent of particular programming languages and development processes.

Provide a formal basis for understanding the modelling language

#### **User Cases**

Use Cases are text stories, widely used to discover and record requirements. Use cases need to be more detailed or structured and emphasize the user goals and perspective. A use case diagram in the Unified Modelling Language (UML) is a type of behavioral diagram defined by and created from a use case analysis. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted. The Figure below illustrates exactly what kind of possibilities or options are for user and administrator of the system



Figure 3: diagram showing the Use case

#### **Sequence Diagram**

Sequence diagrams are a type of UML (Unified Modeling Language) diagram that visually represents the interactions and order of events between objects or components within a software system. They illustrate the flow of messages or method calls between different entities, such as objects, classes, or components, over a specific period of time. The flow of messages, events and actions between the objects or components of a system have been easily used to represent or modelled by UML sequence diagrams. Sequence diagrams are mainly used for finding the logic of the system. Document, model the design and displaying the architecture of the system, by describing the actions that need to be performed for completing a task. To conclude, sequence diagrams are useful in system architecture, as really good engineering tools to design appropriate system, they have been used also in describing object-oriented systems. In other systems, we use the tool for showing the system architecture with flow diagrams and protocol stack design with analysis.



Figure 4: Diplays the Sequence diagram for the application

#### Deployment application

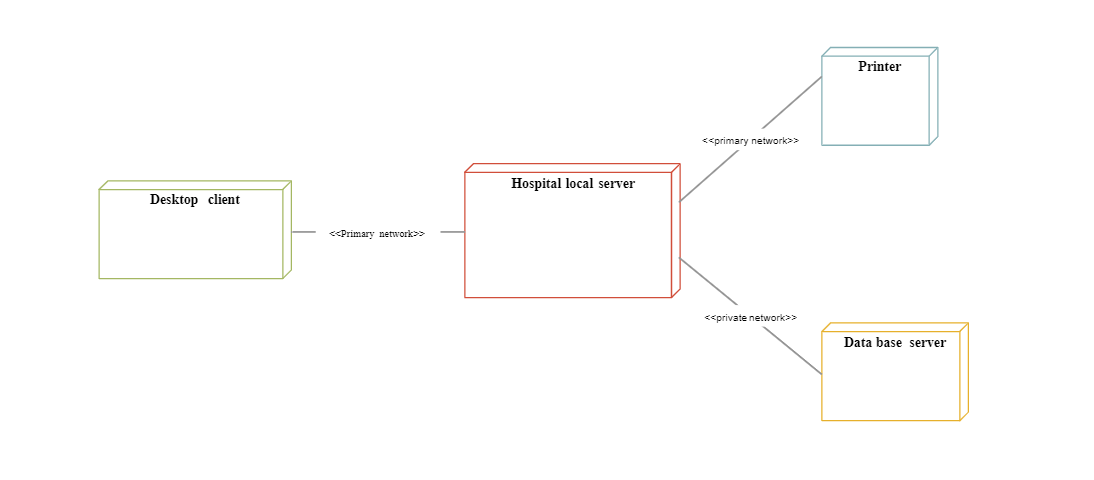
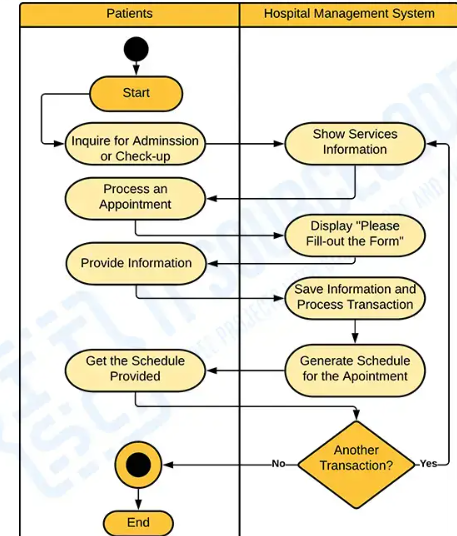
A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them. Deployment diagrams are typically used to visualize the physical hardware and software of a system.

Figure 5: Display deployment application

#### **Activity diagram**

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram. Activity diagrams are often used in business process modeling. They can also describe the steps in a use case diagram. We came up with the following diagram to illustrate the activities:



#### **Flow chart**

#### The system flow diagram is a visual representation of all processed in sequential order. The System flow chart diagram is a graphical representation of the relation between all the major parts or step of the system. Flow chart diagram can not include minor parts of the system

#### .

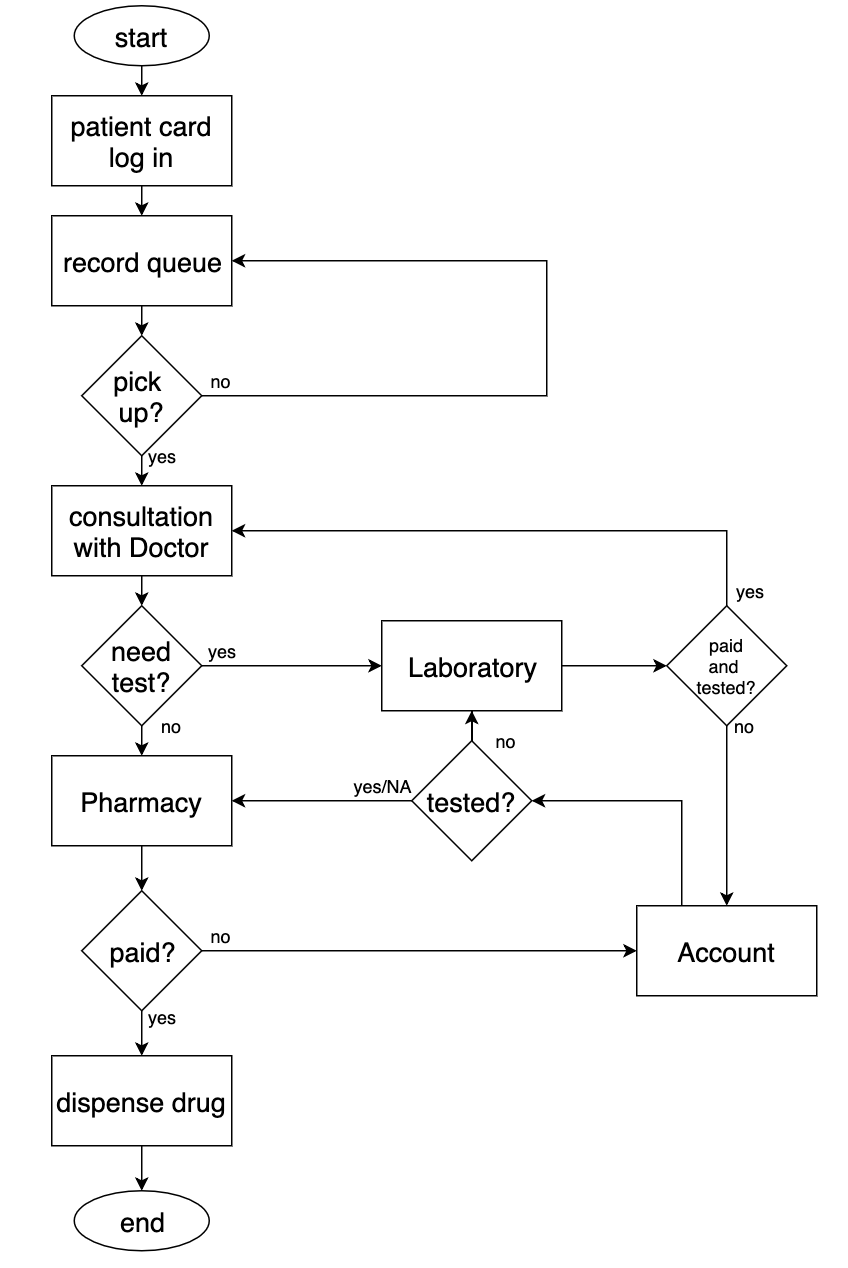


Figure 6: Diagram showing Flowchart

#### **DFD diagram**

This Data flow diagram for hospital management contains a large set of color samples and libraries with predesigned vector pictograms and symbols of health, healthcare equipment, medical instruments, pharmaceutical tools, transport, medication, departments of healthcare organizations, medical icons of people, and human anatomy, as well as predesigned flowchart objects, connectors, and arrows, making it the best for designing clear and comprehensive Medical Wor This data flow diagram for hospital management can also be used for healthcare workflow analysis and management. This hospital management data flow diagram is used to create an overview of hospital management without too much detail.

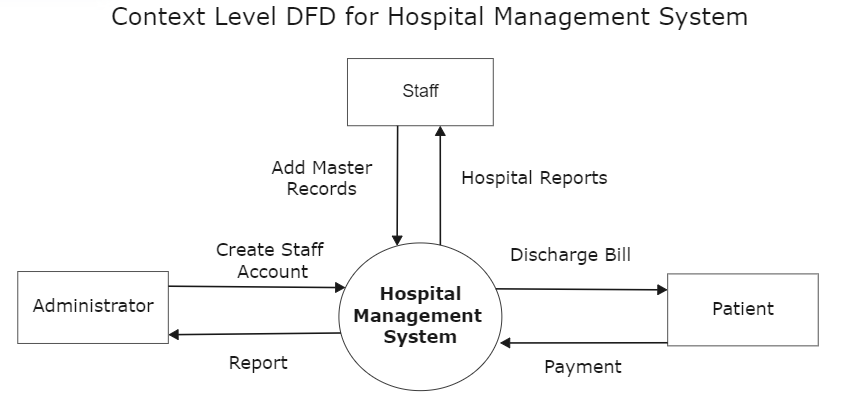


Figure 7 : Context level DFD Hospital Management System

**Staf**

1. The Administrator controls the Hospital Management System, overseeing its functioning.
2. The Hospital Management System interacts with various data sources, such as Staff Records and Patients Records.
3. The Staff Records store information about hospital staff, including their roles, schedules, and contact details.
4. The Patients Records store data related to patients, including their medical history, personal information, and assigned healthcare providers.

### 

Figure 8 : Dfd showing Administrator

### 

Figure 9: Diagram Showing Level 1 DFD

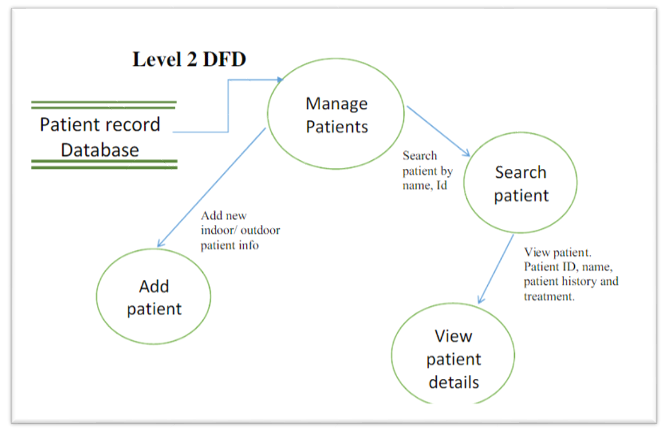


Figure 10: Diagram Showing Level 2 DFD

### **Existing Software Development Tools.**

**Visual Studio Code:**

A lightweight and versatile code editor with extensive PHP language support and a wide range of extensions for enhanced development.

**Git:**

A distributed version control system that allows for efficient collaboration and tracking of code changes. Platforms like GitHub or GitLab can be used for hosting and managing repositories.

**MySQL:**

A widely-used open-source relational database management system that integrates well with PHP and offers excellent performance and scalability.

**HTML/CSS:**

Standard markup and styling languages used for creating web interfaces. Frameworks like Bootstrap or Tailwind CSS can be employed to expedite UI development.

**JavaScript**

A versatile scripting language that enables dynamic and interactive elements on web pages. Libraries like jQuery or frameworks like Vue.js or React.js can enhance UI functionality.

**PHP**

(Hypertext Preprocessor) is a widely-used scripting language for web development for backend.

These tools can significantly streamline the design and development process for a web-based PHP hospital management system, providing efficiency, collaboration, and code quality assurance.

### **Application Model**

An incremental model was used for the application, which is based on the collection of the descriptive data through the people related to the process itself, i.e. those who work in the hospital. For this purpose, it was necessary to identify the processes, with the aim of accelerating these with the system to be implemented. The processes to be optimized should be plotted on a graph [5], [24], as shown in Figure 2. Figure 2: Incremental model of administrative control of medical records

1. **First Increment.**

Patient records control In this first increase, all the information of the people who undergo treatments will be collected, all this information will be stored in a MySQL database. This information will include the records of the patient's health problems, the disease they suffer from, the treatment they are undergoing with their respective schedules, among others; this way the doctor will closely follow the evolution of the treatment.

1. **Second Increment.**

Control of medical treatments In this increase, all patients will be registered not only with their patient ID and full names, but also with their photos, as well as the information of the doctor and nurse who attended them at the health center.

1. **Third Increment.**

Medical consultation control In the control of the consultations, the administrator will register the consultations of the patients. With these records, the administrator will have better control of the attendance of the patient at the health center for proper treatment.

1. **Fourth Increment.**

Control of the medical history of patients, once the previous steps have been carried out correctly, the fourth increment will deal with the control of the data entered to avoid possible errors that may have arisen throughout the process.

After defining the increments, the steps of the Scrum methodology will be carried out.

### **Importance of system design**

Having a system design closes the gap between the requirements and the actual implementation. It gives a blueprint of how the system should work and what modules should be present to achieve the objectives. It also helps to keep the implementation consistent with the architectural modules and expected outcome.

## **IMPLEMENTATION**

### **Introduction**

The implementation stage of software development is the process of developing an executable system for delivery to the customer.  The implementation of software design starts in terms of writing program code in the suitable programming language and developing error-free executable programs efficiently

### **System implementation strategy**

A project implementation strategy depicts the precise steps to take to turn a concept into reality. For successful project, implementation strategies attempt to provide step-by-step, reproducible processes that coordinate project execution until the requirements and objectives of the projects are achieved.

In implementation of the HMS, we used html and bootstrap for the frontend. Besides, that we had prototyped the frontend application on figma which helped us to see how responsive the system will be on different screen sizes. We used the java script framework jQuery for animation and validation of data from the frontend since it JQuery simplifies HTML document traversing, event handling, animating, and Ajax interactions for rapid web development.

Backend/Server side was implemented with PHP, since it is scalable, PHP allows web developers to create dynamic content and interact with databases. PHP is known for its simplicity, speed, and flexibility — features that have made it a cornerstone in the web development world.

The system runs on XAMPP server with PHP version 8.0. We used the Maria DB database, which is a relation Database Management System (RDMS) since its biggest advantage is in speed and performance. When it comes to performing queries or replication, Maria DB is faster than MySQL. Therefore, if you need a high-performance relational database solution, Maria DB is a good choice.

The optimal implementation strategy for the given model is the Phased implementation approach. It helps an organization to verify its approach for full application deployment through phased implementation. The piloting implementation strategy can identify operability issues associated with production-like situations and give you a chance to fix them before the whole application gets up and running. Pilot implementations appear to offer a mechanism to move prototype from the lab to the field, allowing users to interact with a system design in real-world settings while the design is still changeable, and developers to gather input from real-world use.

### **Development specification**

We used these specification for the development of the hospital management system

**Hardware Specifications**

Processor: Core i 3 and above

RAM: 2GB or Higher

Disk Space: 500 Mb

LAN Ethernet 10/100Mbps card/bus.

**Software Specifications**

Operating System: Windows 10/11

Web Browser: Chrome / Brave/ Mozilla browser

Database: Apache 2.0 as web server

PHP : Version 8

MySQL version 5.0.1 or higher as database

### **Code details and code efficiencies**

Code efficiency is a broad term used to depict the reliability, speed and programming methodology used in developing codes for an application.

The system was designed with Object oriented Programming PHP rather than the usual procedural programming style. Adopting PHP OOP promotes code organization, reusability, maintainability, and scalability. It enhances code readability, testability, and development efficiency, leading to more robust and flexible applications.

We used comment to enhance readability of the system code. The system development used a modular approach which helped to reduce the use of using redundant code throughout the system.

We designed and built the HMS system prototypes during the development. Building the system involved a series of steps that involved data gathering, data preparation, exploratory analysis, model training and testing, model evaluation and saving.

We are going to share the code snippet for the system:

#### **Frontend login with html**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Login</title>

<link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.7.1/css/all.css">

</head>

<body>

<div class="login">

<h1>Login</h1>

<form action="authenticate.php" method="post">

<label for="username">

<i class="fas fa-user"></i>

</label>

<input type="text" name="username" placeholder="Username" id="username" required>

<label for="password">

<i class="fas fa-lock"></i>

</label>

<input type="password" name="password" placeholder="Password" id="password" required>

<input type="submit" value="Login">

</form>

</div>

</body>

</html>

#### **Login Logic from the Server side**

**<?php** session\_start(); if (isset($\_POST['login'])) {

// Connect to the database

$mysqli = new mysqli("localhost", "username", "password", "login\_system");

// Check for errors

if ($mysqli->connect\_error) { die("Connection failed: " . $mysqli->connect\_error); }

// Prepare and bind the SQL statement

$stmt = $mysqli->prepare("SELECT id, password FROM users WHERE username = ?"); $stmt->bind\_param("s", $username);

// Get the form data

$username = $\_POST['username']; $password = $\_POST['password'];

// Execute the SQL statement

$stmt->execute(); $stmt->store\_result();

// Check if the user exists

if ($stmt->num\_rows > 0) {

// Bind the result to variables

$stmt->bind\_result($id, $hashed\_password);

// Fetch the result

$stmt->fetch();

// Verify the password

if (password\_verify($password, $hashed\_password)) {

// Set the session variables

$\_SESSION['loggedin'] = true; $\_SESSION['id'] = $id; $\_SESSION['username'] = $username;

// Redirect to the user's dashboard

header("Location: dashboard.php"); exit; } else { echo "Incorrect password!"; } } else { echo "User not found!"; }

// Close the connection

$stmt->close(); $mysqli->close();

}

#### **Encryption**

$hash = password\_hash($password, PASSWORD\_BCRYPT, array("cost" => 10));

#### **Decision making process**

if (password\_verify($password, $hash)) {

if (password\_needs\_rehash($hash, $algorithm, $options)) {

$hash = password\_hash($password, $algorithm, $options);

}

}

### **User Interface Implementation**

This section we are going to share the screenshots of various module in the system.

#### **The landing page**

Is the first page that the user interact with when accessing the websites

It contains the link to the the doctors page and the admins portal

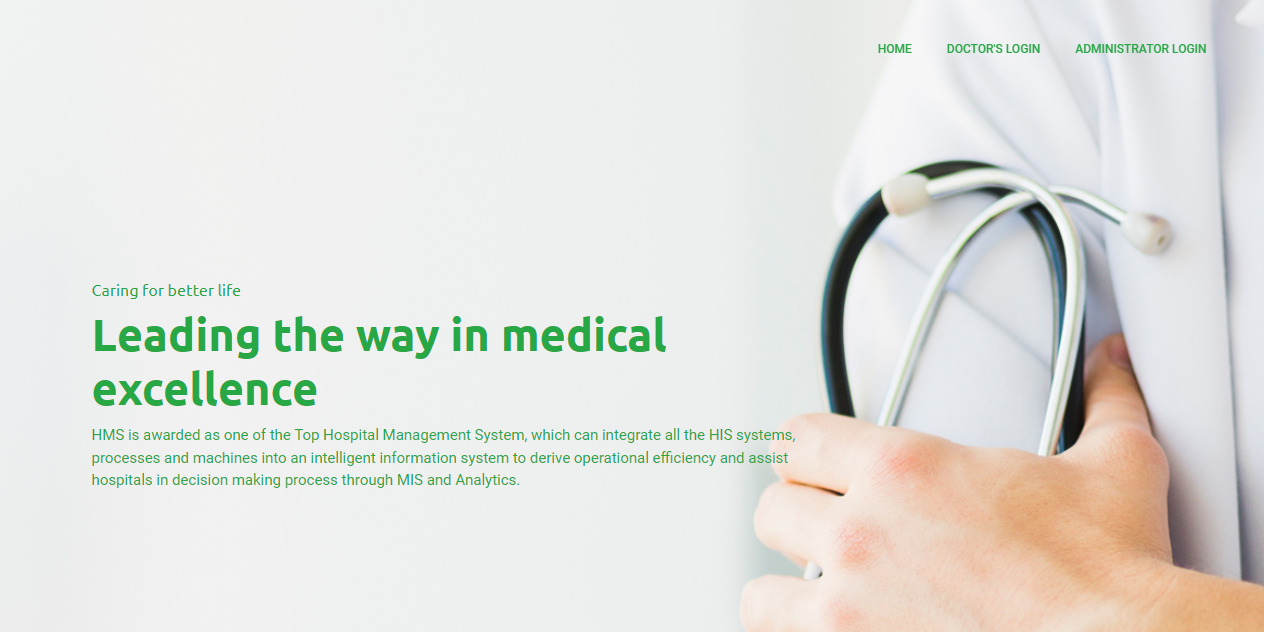


Figure 11: Showing Landing page

#### **Login page**

The page where the user inputs the credential before login(username/email and password)

It contains link to forgetting the password incase the user forgot

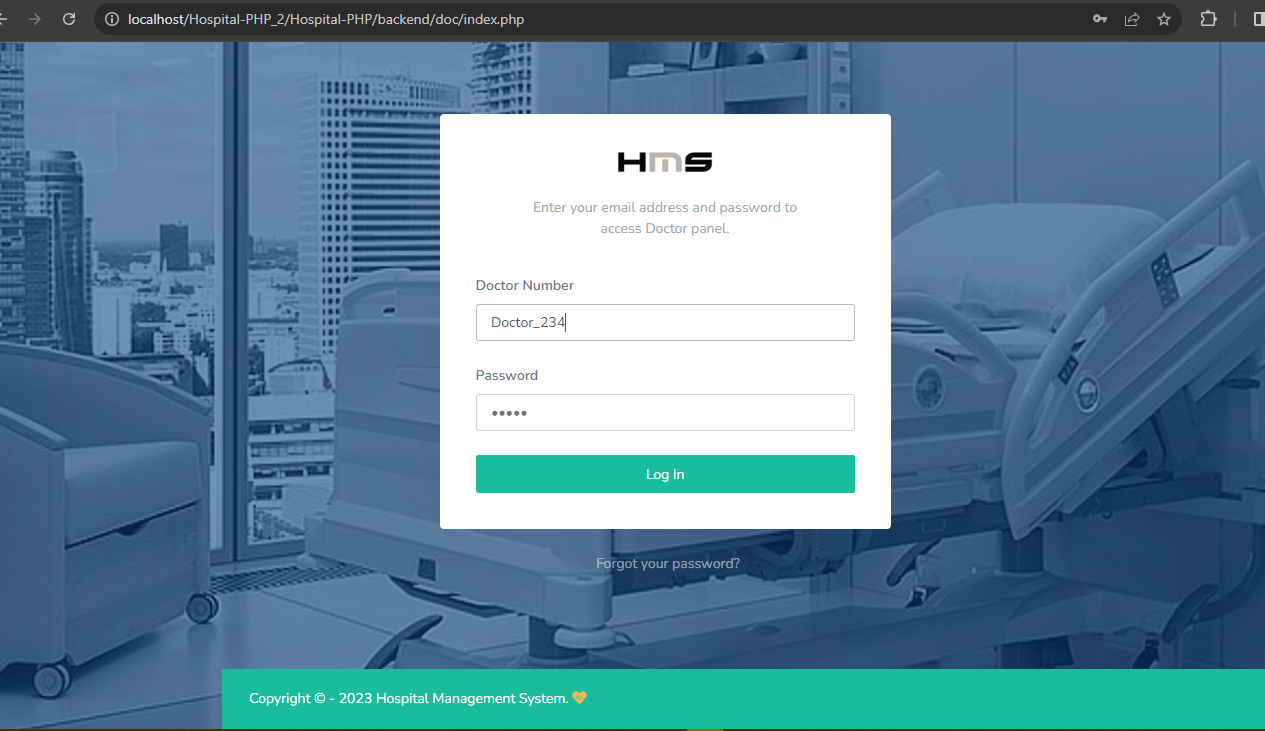


Figure 12: Showing loging page

### **Doctors dashboard**

Is where the doctor is able to manage and view the patient activities and data in the system

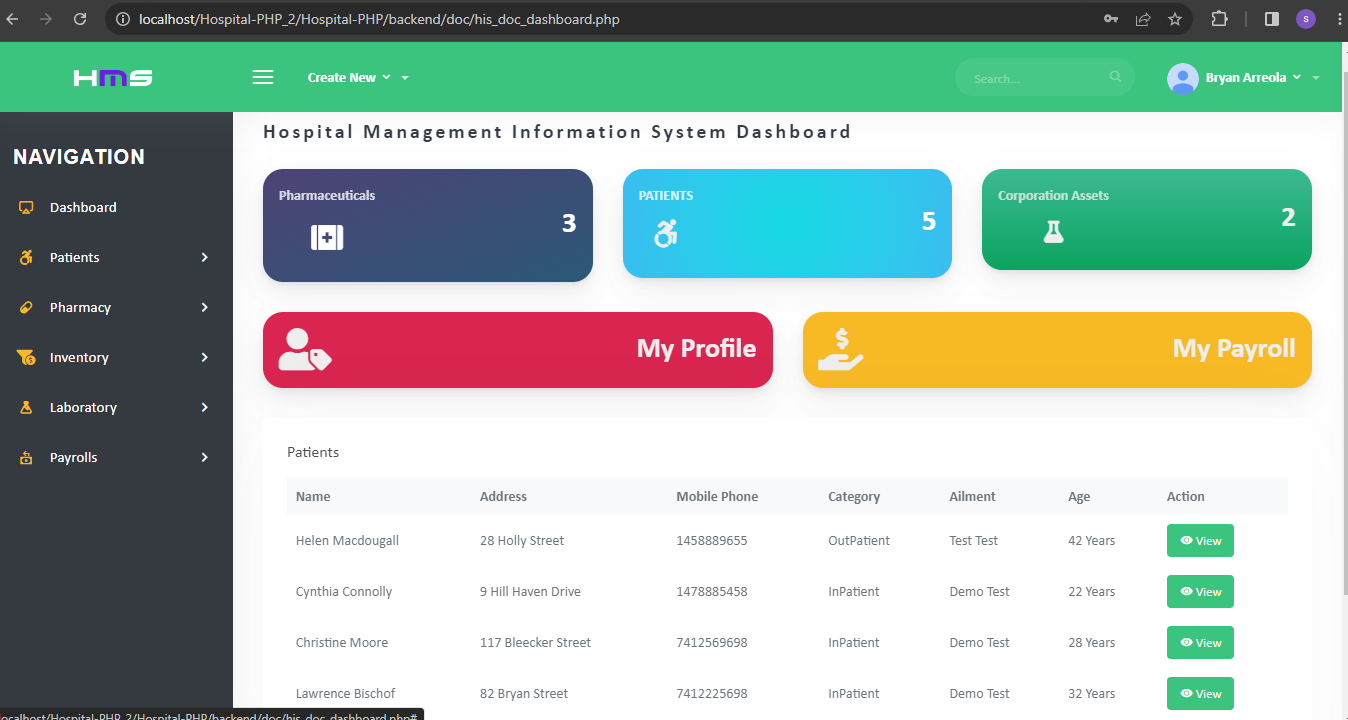


Figure 13: Showing Doctors Portal

### **Admin dashboard**

Is where the administrator is able to manage and view the patient activities and data in the system. The admin is also able to track and manage the equipment, activities, employees, employee’s payroll etc.

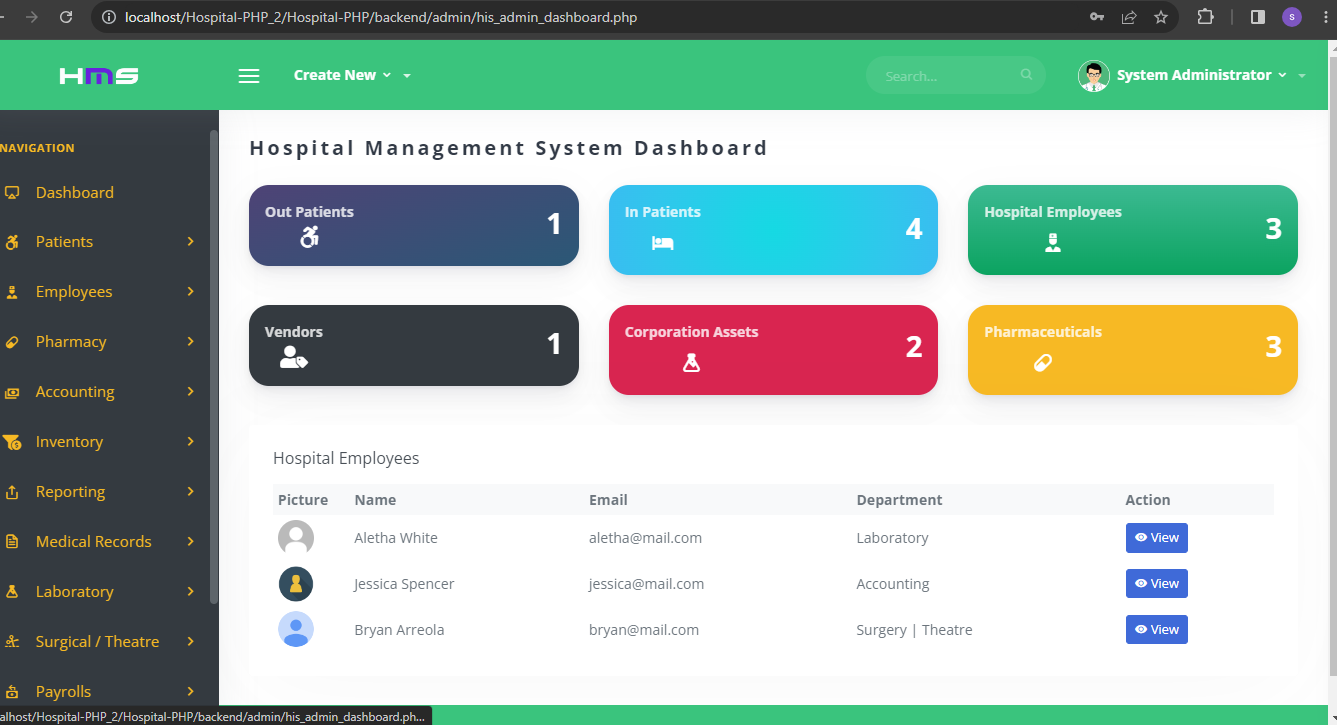


Figure 14: Showing the Administrators Portal

## **Testing**

### **Introduction**

Software testing for a hospital management system is an essential process that ensures the system functions effectively and reliably in a critical healthcare environment. Software testing has the power to point out all the defects and flaws during development. The testing phase helps identify and rectify any errors, vulnerabilities, or usability issues that may arise in the software, thus enhancing the overall quality and performance of the system.

System testing verifies and validates the behavior and characteristics of complete and fully integrated product based on the software requirements specification documentation. Testing is always done from the onset of development process considering the fact that the development was based on prototyping.

There are two types of testing basis, which include black box testing and white box testing. White box testing usually includes the internal working of the system/software involved that is the whole essence of the code part of the system. Black box testing does not need the tester to have any knowledge of the system or the internal structure of the system. On the other hand, system testing done through testing which includes; unit testing, integrating testing, usability testing and acceptance testing.

#### **Unit Testing**

Unit testing is the type of testing that deals with testing of individual sub system and componentsof the system. In unit testing, the modules or subsystems of a web portal for mental health careand counseling services system were tested the results are summarized in the table 2 below.

We were looking at the following component:

1. Verify that doctors can access patient medical records.
2. Verify that the system can login with correct credential
3. Test that the system can generate reports on patient visits and treatment.
4. Test that the system can handle laboratory test results.
5. Test that lab technician can view the result send by doctor
6. Update the details of the patients

### **Results**

##### **Verify that the system can login with correct credential**

When the doctor enter correct credential he is able to login and access tha portal

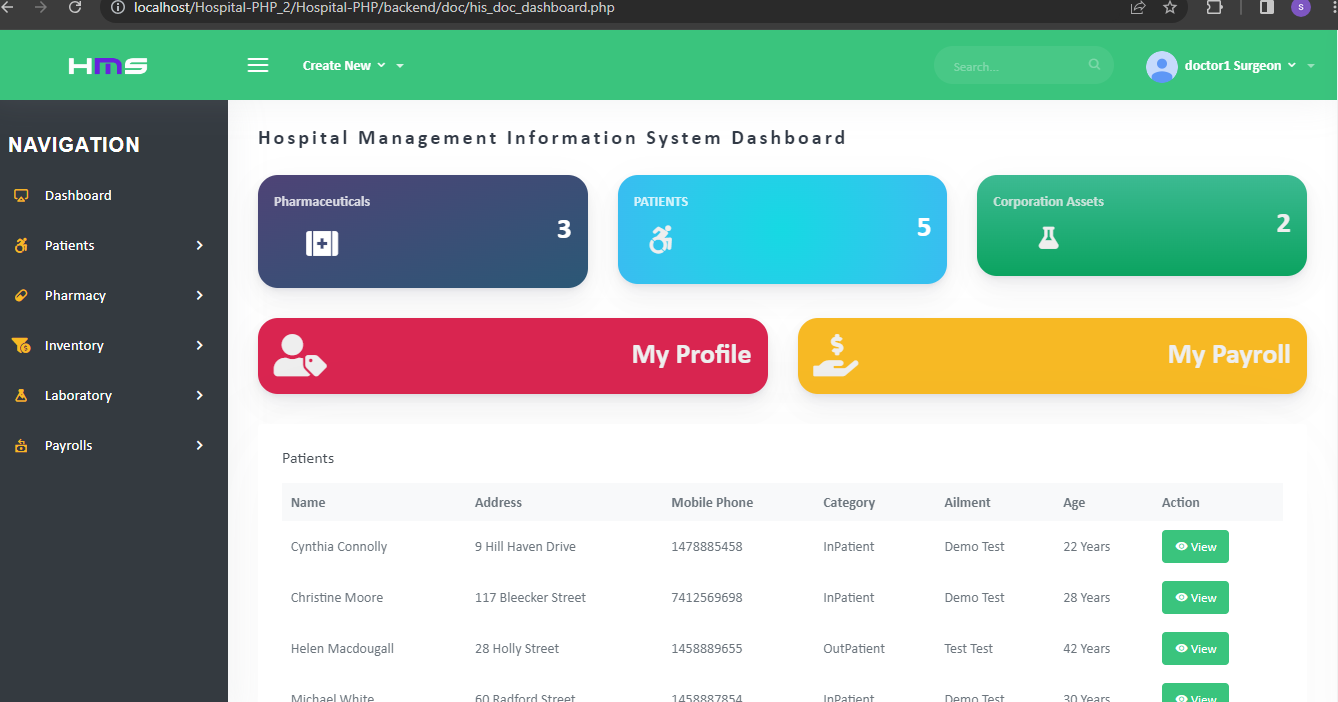


Figure 15: doctor portal after enterring correct credential

Or else when he uses the login credential he get the alert of failed login

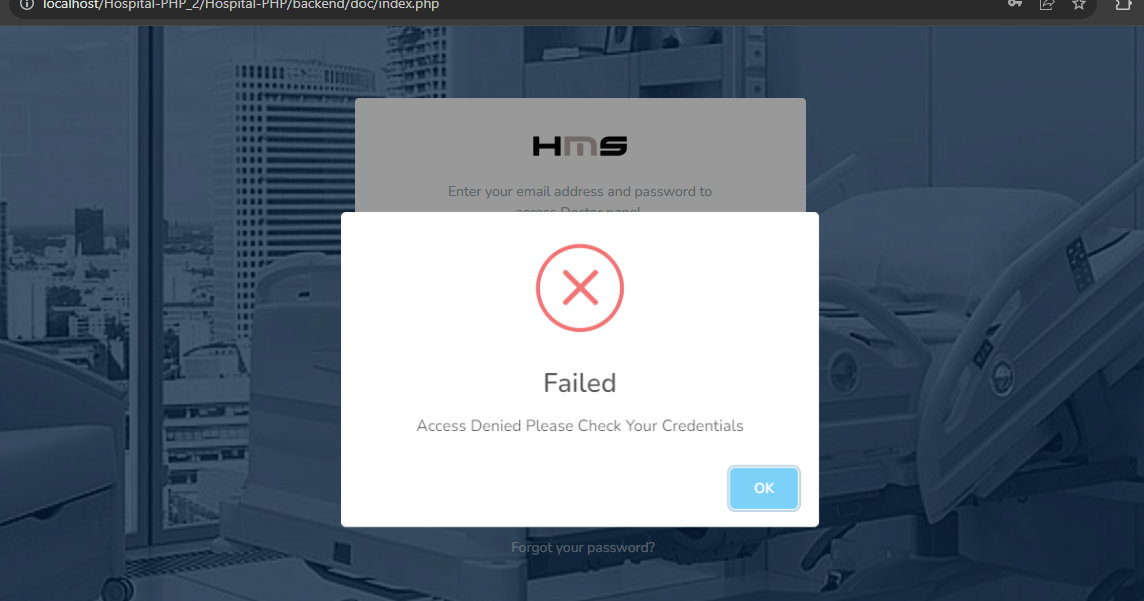


Figure 16: Wrong credential alert

##### **Test that the register new patients**

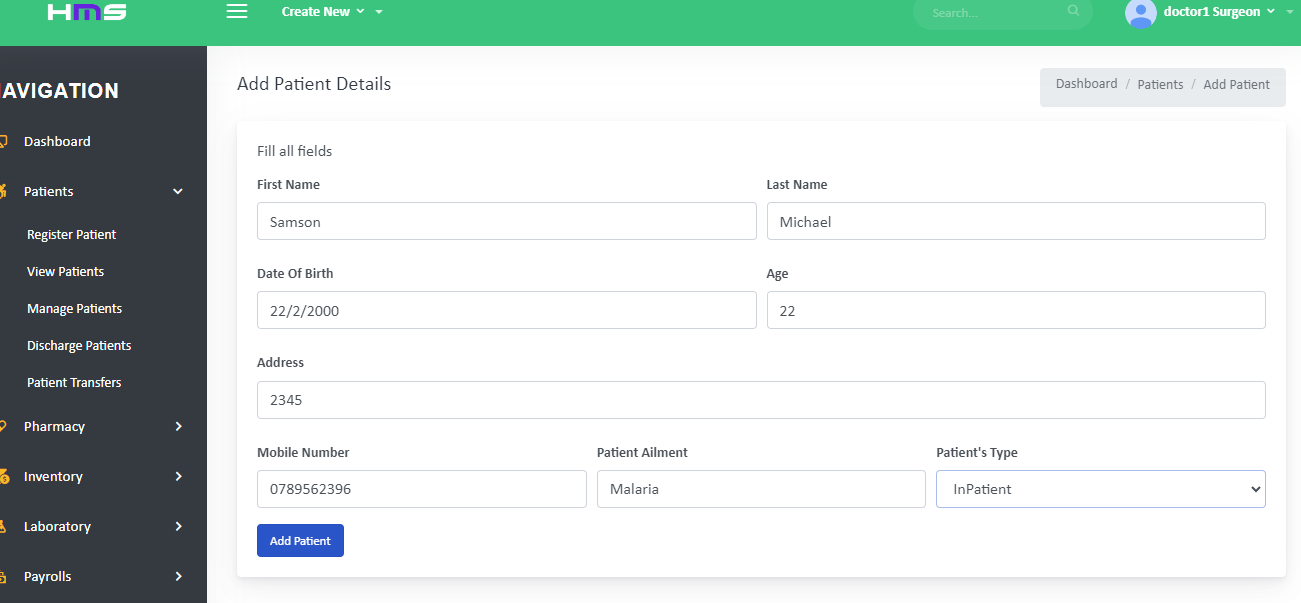


Figure 17: Registering new user

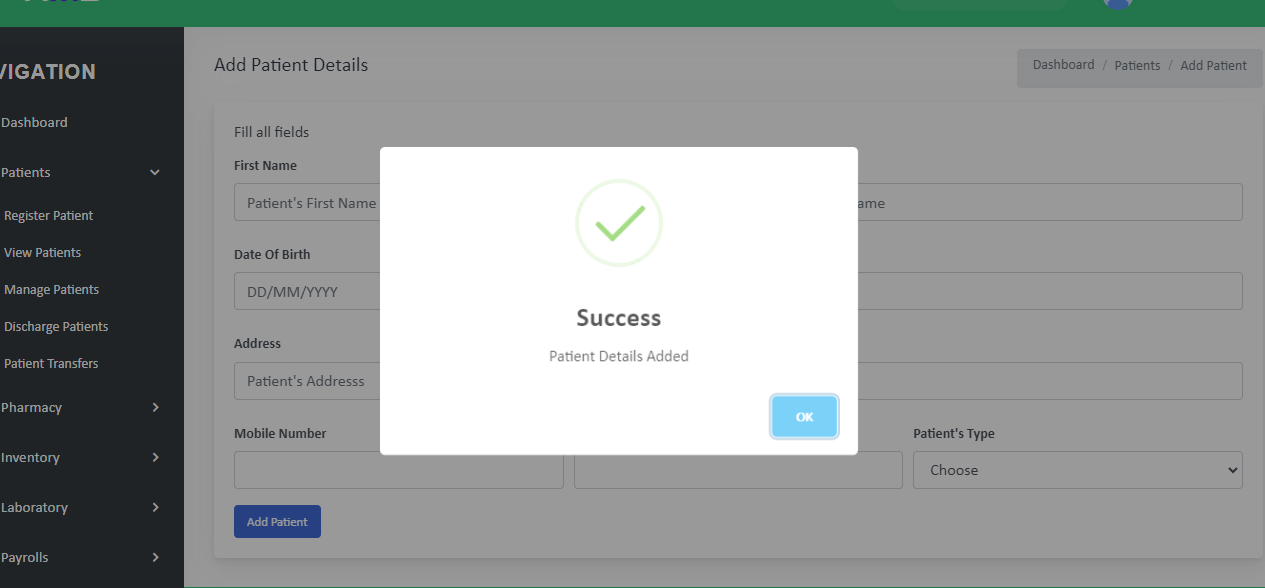


Figure 18: Alert showing patient data added

#### **.**

##### **Test that the system can generate reports on patient visits and treatment.**

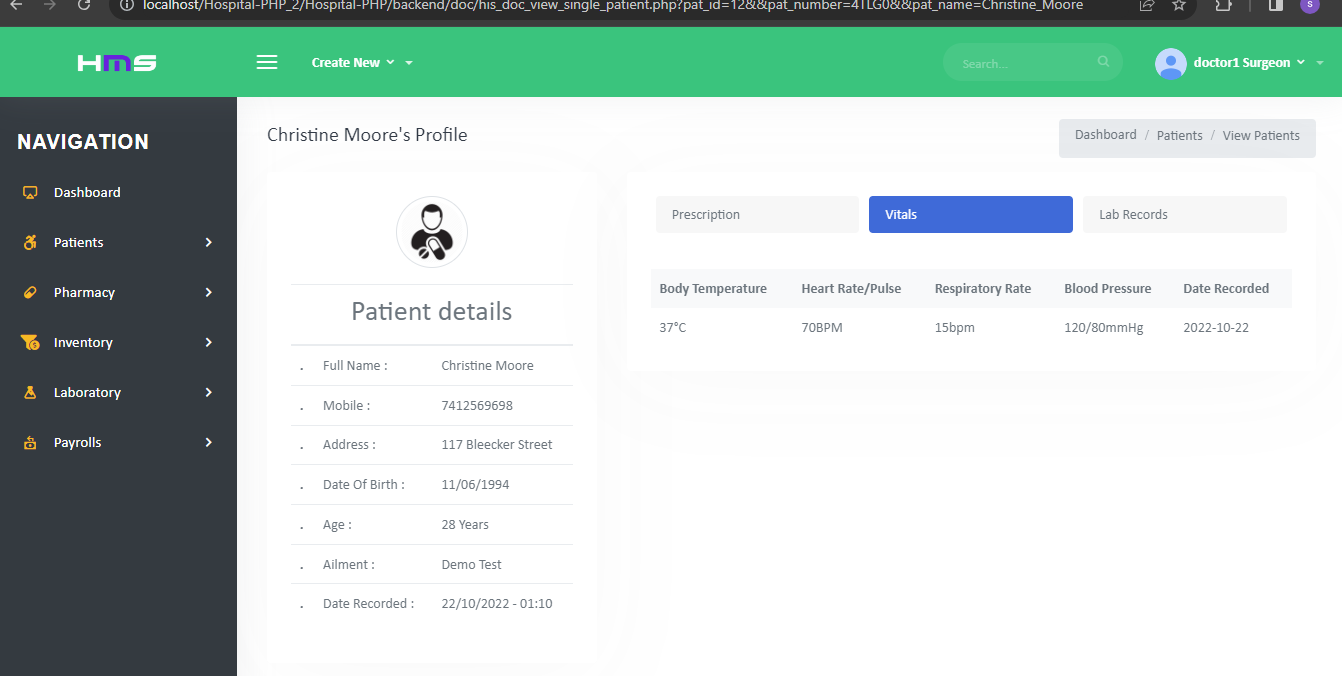


Figure 19: generate reports on patient visits and treatment

##### **Test that the system can handle laboratory test results.**

Adding of test result

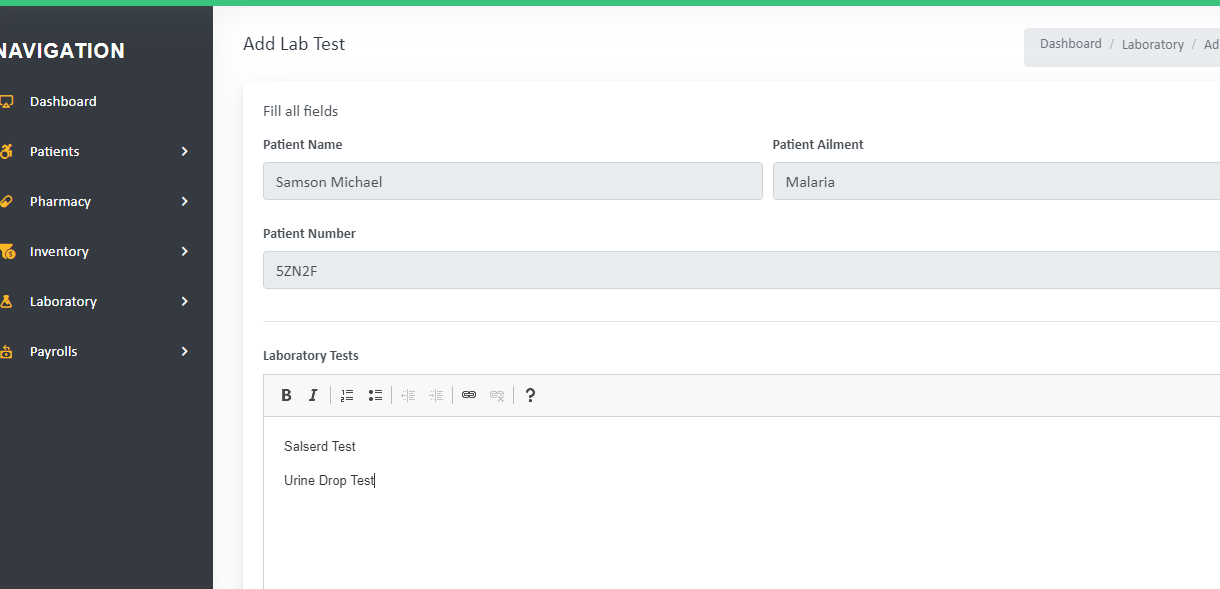


Figure 20: Adding test result

Added successfully

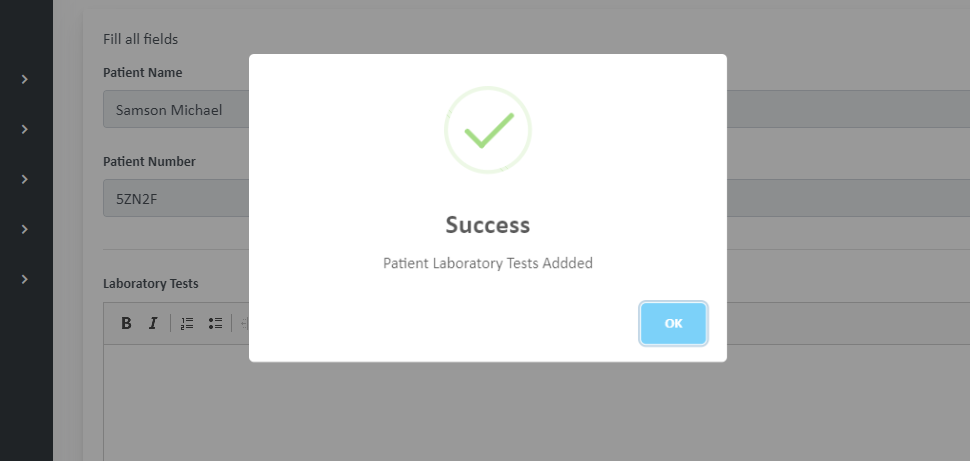


Figure 21: alert showing Added successfully

##### **Test that lab technician can view the result send by doctor**

The results after the test has been performed

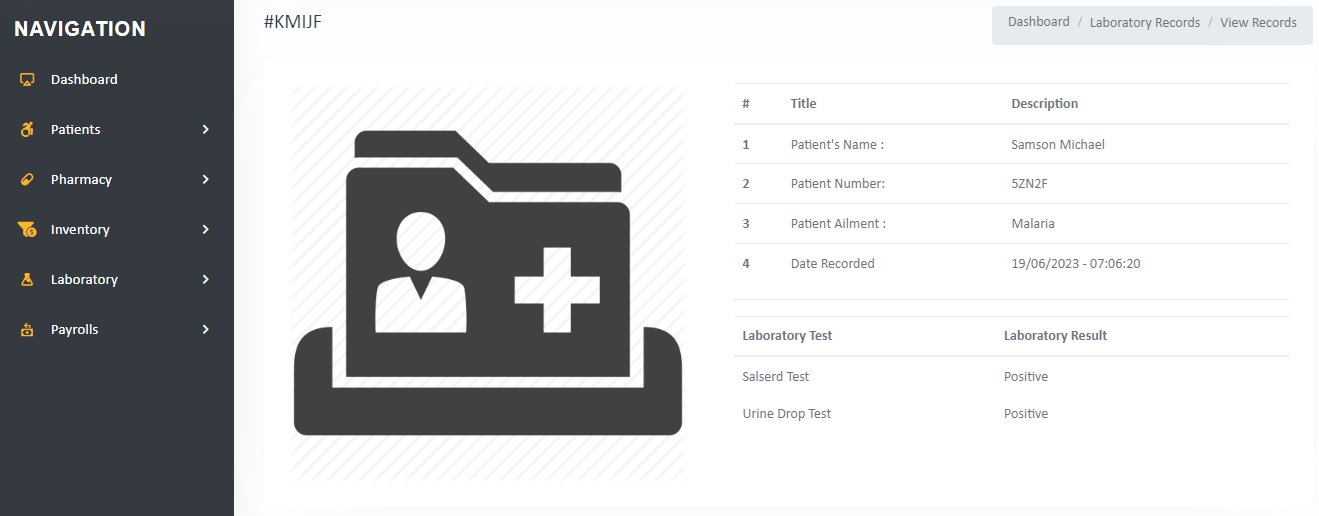


Figure 22: The results after the test has been performed

##### **Update details of the patient**

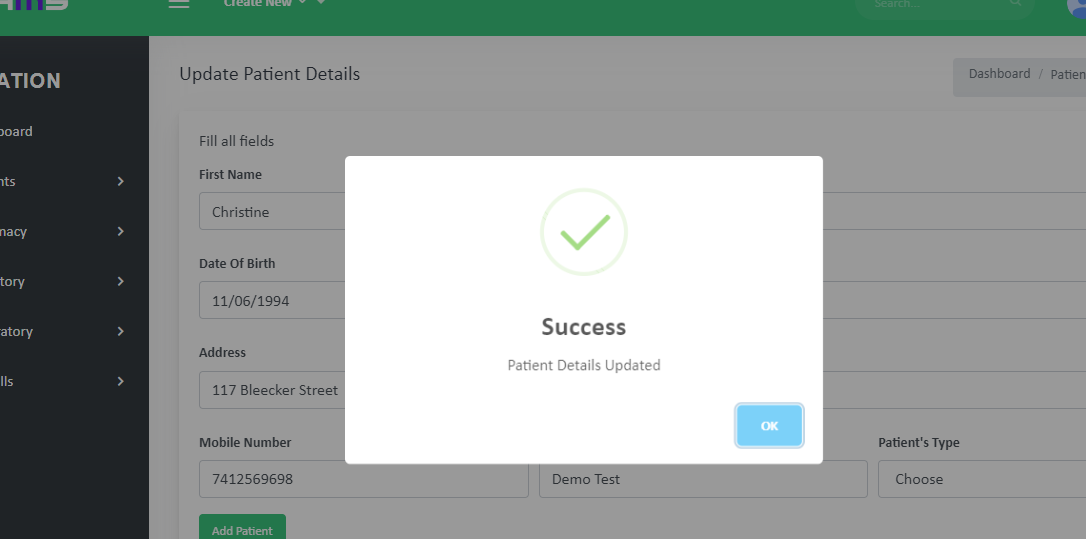


Figure 23: Updated patients details

#### **TEST CASES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **no** | **Test objective** | **Test input** | **expectation** | **Result** |
| 1 | Verify that doctors can access patient medical records. | Clicking view patient record | See correct patient record | PASSED |
| 2 | Verify that the system can login with correct credential | Right credential | Logging into admin/Doctors portal. | PASSED |
| 3 | Verify that the system can login with wrong credential | Wrong credential | Does not logging and prompts an alert | PASSED |
| 4 | Test that the system can generate reports on patient visits and treatment. | Click patient data | Able to see patient visit history and treatments | PASSED |
| 5 | Test that the system can handle laboratory test results. | Stores the data of the patient test and result | View result to the doctor and see the test | PASSED |
| 6 | Test that lab technician can view the result send by doctor | Enter result of test and click submit | Send result to the doctor | PASSED |
| 7 | Update the details of the patients | Change patient details and click submit button | Changes details of the patient | PASSED |

Table 2: Table showing the unit tests performed

#### **Integration Testing**

This is a type of testing in which the integrated subsystems or modules of the system are merged together and performed their functional and no function requirements. It deals mostly with theclient/server relation and helps to discover errors that may have occurred in the system.

#### **System Testing**

In this type of testing the system is tested in accordance to the user requirements. According to the requirements a web portal for mental health care and counselling system, an administrator had role of managing users by activating, deactivating and deleting them, on the other hand a patient could search a particular counselor/ doctor, view their profiles and were connected. In this kind of a testing, no knowledge on inner designs or coding was required; hence, it’s called black box testing.

#### **Acceptance Testing**

*Acceptance testing* is the last phase of software testing before making the system available for use.

*User Acceptance Testing* is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements. We carried out the tests above and our system passed all the stated tests.

**Conclusion of the Testing**

According to the data that were analyzed from user acceptance test. It was found that the system had meet optimal user requirements and it was proven to be helpful in providing online health care and hence it can be able to reduce the number of chances of to which many had suffered since they couldn’t get the quick help and led to problems. The additional features and functionalities such as video conferencing and payments added to the future work to enhance system performance and usability.

## **Evaluation of Software Solutions and Quality**

#### **Introduction:**

Evaluating software solutions and ensuring their quality is particularly crucial in the context of a hospital management system. A hospital management system plays a critical role in managing patient data, streamlining healthcare operations, and providing efficient medical services. The evaluation process helps hospitals and healthcare organizations select the most suitable software solution that meets their unique requirements and maintains a high standard of quality. This section will discuss the importance of evaluating software solutions for a hospital management system and ensuring their quality, outlining key considerations and best practices for effective evaluation and quality assurance.

#### **Importance of Evaluating Software Solutions:**

The evaluation of software solutions for a hospital management system holds significant importance due to the following reasons:

1. **Fit-for-Purpose/ Alignment with Specific Requirements:**

An effective evaluation process ensures that the chosen software solution aligns with the unique requirements of a hospital. For instance, evaluating the appointment scheduling module involves verifying whether it supports multiple specialties, integrates with the existing electronic health record (EHR) system, and offers flexibility in managing patient appointments. This evaluation ensures that the software can effectively support and streamline their operations.

1. **Performance and Scalability:**

Hospitals operate under demanding conditions, with numerous concurrent users and large amounts of data. When evaluating a software solution, it is vital to assess its performance and scalability. For instance, load testing the system can simulate peak usage scenarios, ensuring that it can handle a high volume of transactions without compromising response times or system stability. . This evaluation ensures that the system can efficiently manage the demands of a busy healthcare environment.

1. **Security and Compliance:**

Hospitals handle sensitive patient data, making security a top priority. Evaluating software solutions includes assessing their security measures and compliance with healthcare regulations like HIPAA (Health Insurance Portability and Accountability Act). For example, a robust hospital management system should encrypt patient data, enforce access controls, and provide audit logs for compliance audits. Evaluating software solutions ensures that the chosen system adheres to these standards, protecting patient privacy and maintaining data security.

1. **Usability and User Experience:**

The usability of a hospital management system significantly impacts its adoption and efficiency. Evaluating the software's user interface and user experience involves conducting user feedback sessions and usability tests. For instance, ensuring that the medication administration module has a streamlined interface with intuitive workflows can reduce medication errors and enhance user satisfaction. This evaluation ensures that the system is easy to use, minimizes user errors, and enhances productivity.

1. **Integration and Interoperability:**

Hospital management systems need to integrate with various other healthcare systems, such as laboratory information systems, radiology systems, and billing systems. Evaluating a software solution includes assessing its interoperability capabilities and verifying smooth data exchange between systems. For example, testing the integration of the laboratory module with the laboratory information system ensures accurate and timely retrieval of test results. . Checking thoroughly the software solutions helps determine their compatibility and integration capabilities with these existing systems. Seamless integration minimizes data discrepancies and ensures a cohesive healthcare ecosystem.

#### **Ensuring Quality in Hospital Management Systems:**

To ensure the quality of a hospital management system, the following considerations and practices are important:

1. **Comprehensive Testing:**

Conduct functional testing to validate each module's functionality, such as patient registration, electronic health records, billing, and inventory management. Test scenarios can include verifying accurate data entry, appropriate system responses, and error handling. Implementing a robust testing strategy that covers functional testing, performance testing, security testing, and interoperability testing. Thoroughly test the system to identify and resolve defects, validate data accuracy, and ensure compliance with healthcare standards.

1. **Performance Testing:**

Evaluate the system's performance by stress testing it with a high number of concurrent users and a large dataset. Measure response times, resource utilization, and system stability to ensure optimal performance under peak loads.

1. **Security Testing:**

Perform penetration testing and vulnerability assessments to identify and address security loopholes. Test access controls, data encryption, and vulnerability patches to safeguard patient data and prevent unauthorized access. Pay close attention to data security and privacy aspects. Evaluate the system's data encryption, access controls, audit logs, and compliance with privacy regulations. Regularly update security measures to protect patient information from unauthorized access and data breaches.

1. **Usability Testing:**

Engage users, including doctors, nurses, and administrative staff, to gather feedback on the system's usability. Conduct usability tests to identify areas for improvement and refine the user interface and workflows. Collect their feedback, evaluate their experience, and address any usability issues or concerns. UAT helps ensure that the system meets user expectations and aligns with their workflow requirements

1. **Integration Testing:**

Test the integration of the hospital management system with other systems, ensuring seamless data exchange and smooth interoperability. Validate accurate data transfer, synchronization, and proper functionality of interfaces and APIs.

1. **Continuous Quality Improvement:**

Regularly collect user feedback, analyze system usage data, and prioritize enhancements based on user needs. Implement an iterative development process that incorporates user feedback and ensures ongoing quality improvements. Regularly engage with healthcare professionals and stakeholders to understand evolving needs and incorporate necessary changes into the system.

#### **Conclusion:**

Evaluating software solutions and ensuring the quality of a hospital management system are critical steps to enhance patient care, streamline operations, and improve overall healthcare efficiency. By meticulously evaluating software solutions for compatibility, performance, security, usability, and interoperability, hospitals can select and implement a robust system that meets

## **SUMMARY, CONCLUSION AND RECOMMENDATIONS**

### **Introduction**

This section had an aim of implementing hospital management system, the summary, conclusion and recommendations are going to be discussed below, this section provides report to get the overall picture of what the system and this report was about.

### **Conclusion**

## In Conclusion, from a proper analysis and assessment of the designed system, it can be safely conclude that the system is an efficient, usable and reliable records management system. It works properly and adequately meets the minimum expectations that were set per the requirements. The new system is expected to give benefits increased overall productivity, performance and efficient records management.

### **Future works**

Future Works in Hospital Management Systems:

1. **Integration of Artificial Intelligence (AI) and Machine Learning (ML):**

The integration of AI and ML technologies holds immense potential for hospital management systems. These technologies can be leveraged to automate tasks, analyze large volumes of data for predictive analytics, improve clinical decision support, and enhance patient outcomes. Future works could focus on developing AI-driven modules, such as intelligent patient triaging, predictive maintenance of medical equipment, and personalized treatment recommendations.

1. **Internet of Things (IoT) Integration:**

The IoT can play a significant role in hospital management systems by connecting various medical devices, wearables, and sensors. This integration can enable real-time monitoring of patient vitals, asset tracking, automated inventory management, and efficient utilization of hospital resources. Future developments could involve leveraging IoT to enhance patient safety, optimize workflows, and improve overall operational efficiency.

1. **Telemedicine and Remote Patient Monitoring:**

The COVID-19 pandemic has accelerated the adoption of telemedicine and remote patient monitoring solutions. Future works in hospital management systems could focus on expanding and enhancing these capabilities. This includes developing integrated telemedicine modules that seamlessly connect patients and healthcare providers, enabling remote consultations, virtual follow-ups, and remote monitoring of patients' health conditions.

1. **Blockchain for Data Security and Interoperability:**

Blockchain technology offers enhanced data security, integrity, and interoperability, which are critical in healthcare systems. Future works could explore the integration of blockchain into hospital management systems to ensure secure sharing of patient data across different healthcare providers, enable efficient management of consent and authorization, and enhance data privacy.

1. **Mobile Applications and Patient Engagement:**

Mobile applications play a crucial role in patient engagement and empowerment. Future developments in hospital management systems could involve the development of patient-centric mobile applications that provide personalized health information, appointment management, medication reminders, and access to medical records. These applications can foster active patient participation, improve patient satisfaction, and facilitate better health outcomes.

1. **Data Analytics and Business Intelligence:**

Hospital management systems generate vast amounts of data that can be harnessed for meaningful insights. Future works could focus on advanced data analytics and business intelligence capabilities within the system. This includes leveraging data mining, visualization techniques, and predictive analytics to identify trends, optimize resource allocation, improve operational efficiency, and support data-driven decision-making.

1. **Enhanced Interoperability and Health Information Exchange:**

Interoperability remains a challenge in healthcare systems. Future works should aim to improve interoperability by adopting standardized data formats, implementing Health Information Exchange (HIE) frameworks, and fostering seamless integration with external systems. This would enable efficient sharing of patient information, facilitate coordinated care, and improve care transitions.

### **Conclusion:**

Future developments in hospital management systems hold immense promise for revolutionizing healthcare delivery. By embracing emerging technologies, such as AI, ML, IoT, blockchain, and mobile applications, hospital management systems can transform patient care, enhance operational efficiency, improve data security, and enable better decision-making. Additionally, advanced analytics, enhanced interoperability, and telemedicine capabilities will play a crucial role in shaping the future of hospital management systems, ultimately leading to improved healthcare outcomes and patient experiences.

## 

## **QUESTIONARES**

#### Questionnaires rolled out to the users

