

CHAPTER-1

INTRODUCTION OF PROJECT

1.1 Introduction

The main purpose of the project, entitled "**Hospital Management System**" is to create a web-based application that is secure, user-friendly, quick, and cost-effective. HMS that allows patients, doctors, and administrators to interact with the hospital's information system through a web interface. The system is built using HTML5/CSS3, JavaScript, Bootstrap, XAMPP, PHP, MySQL, and TCP/IP technologies. HMS enables various hospital and medical processes to be performed online. It consists of registration, login of patients, and booking their appointments with doctors storing their details in the system. It provides a login page for patients, doctors, and admins each have their username and password. It consists of three modules. Those are the patient, doctor and admin. This Web Application maintains authentication to access the information. Administrators can see patient and doctor information, appointment schedules and add new doctors as part of administrative tasks. A database was created one for the patient and the other for the doctors so that admin can access it. The Patient module includes booking appointments and checking prescriptions. A patient can pay a doctor's Fee online. The doctor module allows doctors to view appointments, give prescriptions and search for patients. Web-based technology provides a wide range of online services in practically every industry. The majority of jobs may be completed online, which helps to minimize the workload, expense, and effort. The paper discusses the concept of a web-based platform that would enable various hospital and medical processes to be performed online utilizing Web networking technologies, which could be crucial for implementing the functionality of online medical administration. This will aid in the administration of patients, the management of doctor schedules, and the maintenance of patient data that are accessible throughout the hospital online patient data storage, management, communication, analysis, and updating. Therefore, by implementing this web-based application many tasks that would be time consuming and inconvenient can be accomplished.

1.2 Objectives

The objective of this project is to develop hospital management web-based application with a front-end with react and the back end with mongo database. This software will help to be more efficient in handling the booking doctors, booking lab test slots, pharmacy services, and getting health programs. This system consists of an admin handling part, which means admin can manage users, pharmacy systems, health program management, and manage booking of doctor's appointments and lab tests. of their patients. It also explains the user interface, different models that could be used to develop software such as this.

- The main objective of this project is to develop a hospital management system using PHP and MySQL (structured query language)
- It will help to provide convenience to patients to make appointment, improve the efficiency, cost-effectiveness, patient care outcomes.
- It will improve the data security of the patients as well as doctors and to store the patient data for future uses.
- It will also aim to provide low-cost maintenance to the health care departments.
- It will also provide separate login for doctors, patients and admin where they can check their details such as doctors can check time of their appointment with the patients, patients can check the availability of the doctor, empty beds and many more.
- It will reduce the chance of data loss by storing all hospital data records in computer and can have backup of this data also.
- It will improve efficiency by avoiding human errors, reduces the work of documentation and have better audit control.

1.3 Scope

The Hospital Management System (HMS) is a comprehensive application designed to automate and streamline the administrative, clinical, and financial activities of a hospital. It serves as a centralized platform to manage hospital operations efficiently, ensuring quality healthcare services. The system will be used as the application that serves hospitals, clinic, dispensaries or other health institutions. The intention of the system is to increase the number of patients that can be treated and managed properly. If the hospital management system is file based, management of the hospital has to put much effort on securing the files. They can be easily damaged by fire, insects and natural disasters. Also could be misplaced by losing data and information.

1.4 Literature Review

Hospital Management Systems (HMS) have evolved into integral tools that streamline and enhance the delivery of healthcare services. With the rapid advances in information technology, these systems are increasingly critical in managing vast amounts of patient data, streamlining administrative processes, and improving clinical workflows. This literature review examines the evolution of HMS, its core components, methodologies employed in its development, associated challenges, and future research directions. By synthesizing findings from diverse studies and industry reports, this review aims to provide a comprehensive understanding of how HMS can revolutionize healthcare delivery and improve operational efficiency in hospitals and other healthcare institutions.

Historical Evolution and Context

The history of HMS reflects the broader evolution of information systems within healthcare. Early hospital information systems (HIS) emerged in the 1960s and 1970s, primarily focusing on administrative tasks such as patient registration, billing, and basic record-keeping. Researchers like Shortliffe (1976) and Leavitt (1982) emphasized the need for computer-aided diagnosis and patient management. The limitations of early mainframe-based systems, such as high costs and limited accessibility, prompted a shift towards more flexible, microcomputer-based solutions in the 1980s and 1990s.

During this period, the advent of personal computers and the development of user-friendly software opened new avenues for the implementation of HMS. The introduction of relational database management systems (RDBMS) like Oracle and MySQL further enabled the handling of large datasets efficiently. The literature from the 1990s to early 2000s (e.g., works by Shortliffe et al. and subsequent reviews) highlighted the benefits of digitizing patient records and automating routine administrative processes, while also noting the challenges of data interoperability and integration with legacy systems.

1.5 Components and Functionalities

Modern HMS incorporate a broad range of functionalities that extend beyond administrative management to include clinical, financial, and operational dimensions. Researchers such as Bates et al. (2003) have identified several core modules that form the backbone of HMS:

1.5.1 Patient Information Management

A fundamental component of HMS is the management of patient data. The patient needs to be registered and log in after logging on he can search for the doctor by giving the location, the reason or problem. Basing on the doctor availability the admin will confirm the booking request and will send to mail that the booking is confirmed he can also view in the status and he can also give feedback basing the performance of the doctor.

1.5.2 Appointment and Scheduling Systems

Efficient management of appointments and scheduling is another critical function of HMS. These systems facilitate the scheduling of patient visits, consultations, and procedures, thereby reducing waiting times and optimizing the use of hospital resources. Literature by Gupta and Kohli (2006) illustrates how automated scheduling systems have improved the operational efficiency of hospitals by minimizing manual errors and enhancing coordination between departments.

1.5.3 Admin

Admin needs to login with username and password and in the admin home screen, he can see the basic functionalities of admin. Admin can view the registered doctors and patients. He can also view the patient's request and doctors requests and he will confirm the patients and doctors requests.

1.5.4 Doctor

Doctor need to be registered by giving the necessary details like experience, timing, fees etc. After registering he need to log in and in the home screen he can view the basic functionalities. He can view the patient request forwarded from admin and he can accept and he can also view the feedback given by patients.

CHAPTER-2

SYSTEM ANALYSIS AND DESIGN

2.1 System Analysis and Design

System Analysis and Design (SAD) is a broad term for describing methodologies for developing high quality Information System which combines Information Technology, people and Data to support business requirement. The SAD technique is not only limited to IT systems and can be used to create just about anything. It is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. It is a problem solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish their purpose. It is a process of planning a new business system or replacing an existing system by defining its components or modules to satisfy the specific requirements. Before planning, you need to understand the old system thoroughly and determine how computers can best be used in order to operate efficiently.

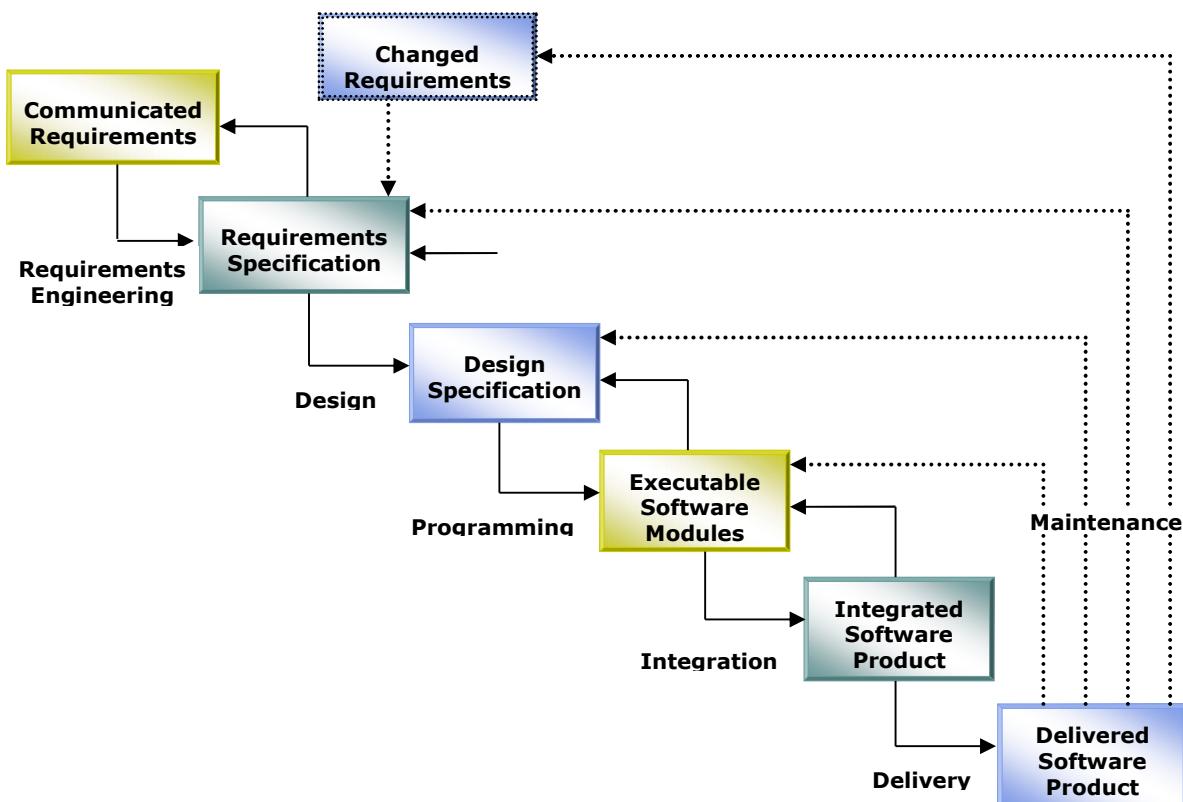


Fig-2.1 System Analysis

The administrative user interface concentrates on the consistent information that is practically, part of the organizational activities and which needs proper authentication for the data collection. The interfaces help the administrations with all the transactional states like Data insertion, Data deletion and Date updation along with the extensive data search capabilities.

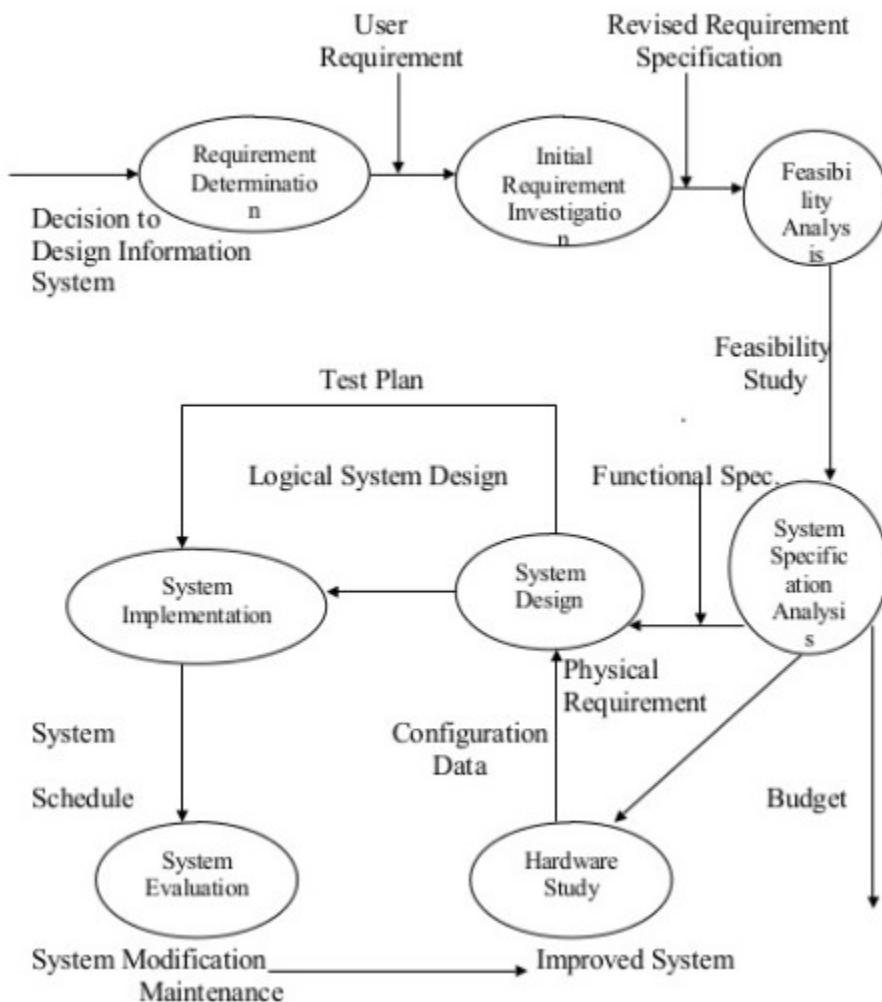


Fig-2.2 System Design Plan

The operational or generic user interface helps the users upon the system in transactions through the existing data and required services. The operational user interface also helps the ordinary users in managing their own information helps the ordinary users in managing their own information in a customized manner as per the assisted flexibilities.

2.2 Existing System

Organization's storage on a daily basis. All of these details are currently kept in the file system. There will accumulate several files because of this daily basis. The hospitals manage and handle every single task by hand. Therefore, it could require a lot of time and effort to complete each activity. A lot of labor is required. At the moment, manual processes are used only in Hospital to manage daily operations. Patients will schedule appointments with doctors and lab tests with the receptionist. Only at hospitals may patients purchase pharmacy items; there is no option for patient delivery. Only at the hospital are the healthcare options offered to patients. All the patient details, doctor details, lab test results are manually taken on papers and feuded on computer latterly. And the reports will be generated by hands with the help to experts.

2.3 Existing System Problem

The existing system requires a lot of time. Absence of security components Every task needs to be completed by hand. The majority of tasks and activities depend on specialists and human resources. No direct communication with the senior officers. The accuracy level is subjective. High expense is required for manual system management. Difficulty in getting backup data and transfer data. Difficulty in inserting IoT technology and robotic intelligent. Manual system in lack in user friendliness and it not that much reliable with current technological world.

2.4 Proposed System

All of a hospital's data and operations are compiled on a single platform by a hospital management system. The hospital information system includes all of the hospital's information processing and storage components. This means that it encompasses more than just the computer systems, networks, and computer-based application systems that are installed on them. Rather, it refers to the information contained within the hospital as a whole.

2.5 Feasibility Study

All projects are feasible when given unlimited resources and infinite time. It's both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. The efforts and resources spent in developing the system will be a waste if the end solution does not offer timely and satisfactory solution to its users. Feasibility study is a test of system proposed regarding workability, impact on the organization ability to meet user needs, and effective use of resources. Thus when a new application is proposed, it normally goes through a feasibility study before it is approved for development. Feasibility and risk analysis are related in many ways. If project risk is great, the possibility of producing quality software is reduced.

2.5.1 Operational Feasibility

People are inherently resistant to change, and computers have been known to facilitate change. An estimate should be made of how strong a reaction the user staff is likely to have toward the development of a computerized system. It is common knowledge that computer installations have something to do with turnover, transfers, retraining, and changes in employee job status. Therefore, it is understood that the introduction of a candidate system requires special effort to educate, sell and train the staff on new ways of conducting business.

2.5.2 Technical Feasibility

Technical feasibility centers around the existing computer system (hardware, software, etc.) and to what extend it can support the proposed addition. For example, if the current computer is operating at 80 percent capacity an arbitrary ceiling then running another application could overload the system or require additional hardware. This involves financial considerations to accommodate technical enhancements. If the budget is a serious constraint, then the project is judged not feasible.

2.5.3 Cost Benefit Analysis

Economic analysis is the most frequently used method for evaluating the effectiveness of a candidate system. More commonly known as cost benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. If benefits outweigh costs, then the decision is made to design and implement the system. Otherwise, further justification or alterations in the proposed system will have to be made if it is to have a

chance of being approved. This is an ongoing effort that improves in accuracy at each phase in the system life cycle.

2.6 Software Requirement Specification

This SRS document outlines a detailed framework for the development of a Hospital Management System that is robust, secure, and user-friendly. By addressing both functional and non-functional requirements, this specification aims to ensure that the final system will not only streamline hospital operations and improve patient care but also remain adaptable to future technological advancements and regulatory requirements.

The successful implementation of this HMS will contribute significantly to operational efficiency, data accuracy, and overall patient satisfaction. Stakeholders are encouraged to review and provide feedback on this document to ensure that all aspects of the hospital's requirements are addressed comprehensively.

Product Perspective

The HMS is envisioned as a standalone web-based application that integrates with existing hospital systems (e.g., laboratory information systems, radiology systems, and insurance claim processing) through secure APIs. It will interface with a relational database (e.g., MySQL) for persistent data storage. The system's design emphasizes modularity, scalability, and security, ensuring that it can evolve as the hospital's needs change.

Overview

This document is organized into sections detailing the overall system description, functional and non-functional requirements, system interfaces, design constraints, and other pertinent information that will guide the development of the HMS.

Purpose

The purpose of this SRS is to provide a detailed description of the Hospital Management System (HMS). This document specifies the requirements and functionalities that the HMS must provide to support the operational, clinical, and administrative processes in a hospital environment. It is

intended for use by project stakeholders, including developers, system architects, quality assurance teams, and hospital management.

Scope

The Hospital Management System is designed as a web-based application that will automate and streamline various hospital functions, including patient registration, appointment scheduling, clinical record management, billing, pharmacy, laboratory services, and human resource management. The system aims to improve operational efficiency, reduce administrative overhead, and enhance the quality of patient care through seamless data integration and real-time reporting.

Hardware Specification

Processor	:	i3 2.83 GHz
Memory	:	1GB
Hard Disk	:	80GB
Monitor	:	14" or above
Mouse	:	Standard Mouse
Key Board	:	104 keys

Software Specification

Operating System	:	WINDOWS 2008/2010
Programming Language	:	HTML/JS/CSS/PHP/BOOTSTRAP
Database	:	MYSQL

Chapter -3

Methodology and System Design

3.1 Methodology

The study adopts the use of object oriented analysis and design method. The underlying principle is that one models software systems as collections of cooperating objects, treating individual objects as instances of a class within a hierarchy of classes. Object-oriented analysis describes an information system by identifying things called objects. An object represents a real person, place, event, or transaction. For example, when a patient makes an appointment to see a doctor, the patient is an object, the doctor is an object, and the appointment itself is an object. Object-oriented analysis is a popular approach that sees a system from the viewpoint of the objects themselves as they function and interact. The end product of object-oriented analysis is an object model, which represents the information system in terms of objects and object-oriented concepts. The following procedures shall be followed in the execution of the work:

- i. Data Collection/Information Gathering: Information was gathered on flow of the manual method of medical appointment and consultation.
- ii. Modelling: Well-defined UML diagrams (Data Flow Diagram, Use Case Diagram, Sequence Diagram) were used for the modelling the proposed system.
- iii. Design and Implementation: Object-oriented design approach is adopted for the design of the proposed system, which is to be implemented as android-based.

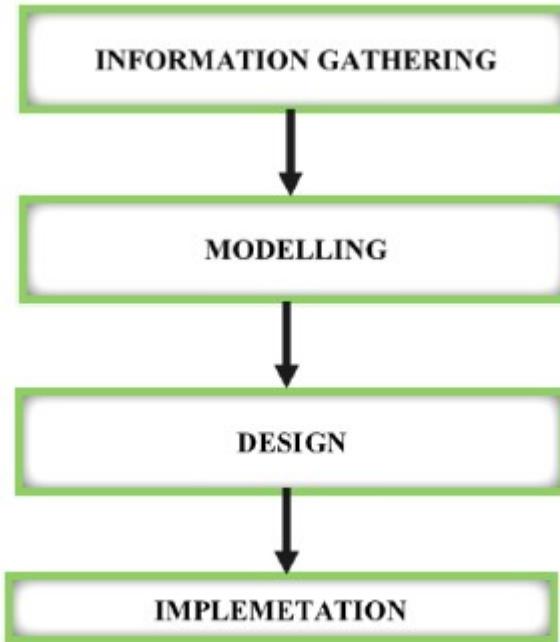
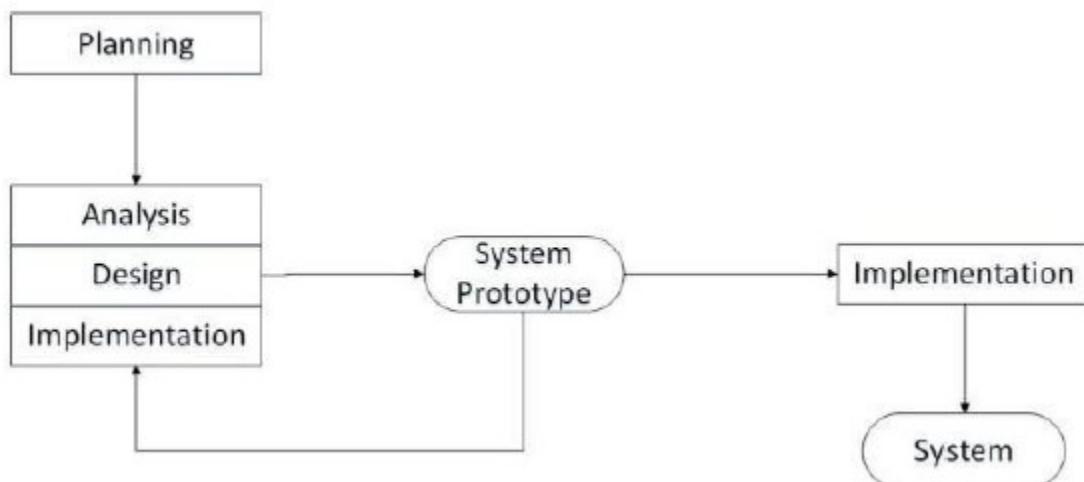


Fig-3.1 Methodology Plan

3.2 System Design

It is the most creative and challenging phase of the system life cycle. The analysis phase is used to design the logical model of the system whereas the design phase is used to design the physical model. Many things are to be done in this phase .we began the designing process by identifying forms, reports and the other outputs the system will produce. Then the specify data on each were pinpointed. we sketched the forms or say, the displays, as expected to appear, on paper, so it serves as model for the project to began finally we design the form on computer display, using one of the automated system design tool, that is PHP 7.0.



After the forms were designed, the next step was to specify the data to be inputted, calculated and stored individual data items and calculation procedure were written in detail. File structure such as paper files were selected the procedures were written so as how to process the data and procedures the output during the programming phase. The documents were design ion the form of charts. Output design means what should be the format for presenting the results. It should be in most convenient and attractive format for the user. The input design deals with what should be the input to the system and thus prepare the input format. File design deals with how the data has to be stored on physical devices. Process design includes the description of the procedure for carrying out operations on the given data.

3.3 Design Decisions

The system design process is a step-by- step adherence of clear procedures and guidelines. Though, certain clear procedures and guidelines have emerged in recent days. But still much of design work depends on knowledge and experience of the designer. When designer works working in system design, he will face different type of problems. Many of these will be due to constraints imposed by the user or limitations of the hardware and software available in the market. Sometimes it is difficult to enumerate the complexity of the problems and solutions there of since the variety of likely problems is so great and no solutions are exactly similar. However, following considerations should be kept in mind during the system designing phase:-design objectives, practicality, efficiency, cost etc. Major designs are:-

3.3.1 Database Design: This activity deals with the design of the physical database. Program design: In conjunction with database design is a decision on the programming language to be used and the flowcharting and coding, and debugging procedure prior to conversion. The operating system limits the programming languages that will run of the system.

System and program test preparation-Each aspect of the system has a separate test requirements. System testing is done after all programming and testing completed the test on the system and program test requirements become a part of design specifications a prerequisite to implementation.

3.3.2 Design Process: The computer system design process is an exercise of specifying how, the system will work. It is an iterative process, which is based on the system will be do as shown in the feasibility report. Mainly, following five parts have been included in the system design process.

3.3.3 Output Design: The starting point of the design process is the proper knowledge of system requirements which will normally be converted in terms of output.

3.3.4 Input Design: Once the output requirements have been finalized, the next step is to find out what data need to be made available to the system to produce the desired outputs. The basic documents in which these data are available need to be identified. If necessary, these documents may have to be revised or new documents may have to be introduced.

3.3.5 File Design: Once the input data is captured in the system, these may be preserved either for a short or long period. These data will generally be stored in files in a logical manner. The designer will have to devise the techniques of storing and retrieving data from these files.

3.3.7 Procedure Design: This step involves specifications of how processing will be performed. In this, there are two aspects:

3.3.6 Computer Procedure: The computer procedure will specify what functions will be carried out on computer, what will be different programs and in what sequence the program will be run.

3.3.7 Non-computer Procedure: The non-computer procedure will specify the manual procedures for feeding input data, receiving outputs etc.

3.4 Design Technique

Design is a multi steps process that focuses on data structure, software, software architecture, external details and interface between the modules. The design processes also translate the requirements into representation of software that can be accessed for quality before coding begins. Computer software designs changes continually as new methods, better analysis and broader understanding evolve. Software design is at a relatively early stage in its revolution. Therefore, software design methodology locks the depth, flexibility and quantitative nature that are normally associated with more classical engineering disciplines.

How ever techniques for software design do exist, criteria for design qualities are available and design notation can be applied. Once software requirements have been analysed and specified,

software design is the first of three activities- Design, code, test, that are required to build and verify software. Each activities transform information in a manner that ultimately results in a validation of computer software. The importance software design can be started with a single word quality. Design is the place where quality fostered in software development. Design provides us with the representations of the software that can be accessed for quality. Design the only way that we can accurately translate a customer's requirement into a finished software product or system. Without design, risk of building an unstable system exists-one that will fail when small changes are made one that may be difficult to test.

3.4.1 Internal Design

The input design is the link between the information system and the users. It comprises the directing specification and procedures for data preparations and those steps that are necessary to put transaction data into a usable form for processing data entry. The designs of inputs focuses on controlling the amount of inputs required, controlling errors, avoiding delay, avoiding extra steps and keeping the process simple. System analyst decides the following input designs details: Data correction (that is, correcting the errors that are found at any of the earlier stages).

3.4.2 External Design

Designing computer output should proceed in an organized, well thought out manner. The term output applies to any information produced by an information system whether printed or displayed. When analyst designs computer output, they identified the specific output is needed to meet the information requirements. Computer output is the most important and direct source of information to the user. Output design is a process that involves designing necessary outputs that have to be various users according to their requirements.

3.4.3 Architectural Design

Architectural design begins with recognition that the screen is composed of different areas. Layout tools assist the analyst in specifying the content of the single and multiple design formats. All screens have been provided with menus, push buttons facilities, icons and control buttons such as add/delete/edit/find/clear /exit etc. The main screen consists of main menu from which we can move to another forms or screens.

3.4.4 Procedural Design

The procedural design transforms structural component in to a procedural description of the software. Source is generated and testing is conducted to integrate and validate to software. The design of input and output screen comes under the procedural design input/output design is according to needs of the user. The input and output design are related to each other in sense that the accuracy data depends on the accuracy of the input data and processing of input data. Thus for this proposed system the input and output design are in the form of forms. In the forms based interface design the user give the input by filling the blanks of the screen.

3.4.5 Database Design

Database files are the key source of information into the system. It is the process of designing database files which are the key source of information to the system. The files should be properly designed and planned for collection, accumulation, editing the required information. The objectives of the file design are to provide effective auxiliary storage and to contribute to the overall the efficiency of the computer program component of the system. In concepts of database design, there are two types of data – physical data and logical data.

Database Name: hms

Table structure for table admin

```
CREATE TABLE IF NOT EXISTS `admin` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `username` varchar(255) NOT NULL,
  `password` varchar(255) NOT NULL,
  `updationDate` varchar(255) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;
```

Table structure for table appointment

```
CREATE TABLE IF NOT EXISTS `appointment` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `doctorSpecialization` varchar(255) DEFAULT NULL,
```

```
'doctorId` int(11) DEFAULT NULL,  
 `userId` int(11) DEFAULT NULL,  
 `consultancyFees` int(11) DEFAULT NULL,  
 `appointmentDate` varchar(255) DEFAULT NULL,  
 `appointmentTime` varchar(255) DEFAULT NULL,  
 `postingDate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
 CURRENT_TIMESTAMP,  
 `userStatus` int(11) DEFAULT NULL,  
 `doctorStatus` int(11) DEFAULT NULL,  
 `updationDate` timestamp NOT NULL DEFAULT,  
 PRIMARY KEY (`id`))
```

Table structure for table doctors

```
CREATE TABLE IF NOT EXISTS `doctors` (  
 `id` int(11) NOT NULL AUTO_INCREMENT,  
 `specilization` varchar(255) DEFAULT NULL,  
 `doctorName` varchar(255) DEFAULT NULL,  
 `address` longtext,  
 `docFees` varchar(255) DEFAULT NULL,  
 `contactno` bigint(11) DEFAULT NULL,  
 `docEmail` varchar(255) DEFAULT NULL,  
 `password` varchar(255) DEFAULT NULL,  
 `creationDate` timestamp NULL DEFAULT NULL ON UPDATE CURRENT_TIMESTAMP,  
 `updationDate` timestamp NOT NULL DEFAULT,  
 PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1  
 AUTO_INCREMENT=10 ;
```

Table structure for table doctorslog

```
CREATE TABLE IF NOT EXISTS `doctorslog` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `uid` int(11) DEFAULT NULL,
  `username` varchar(255) DEFAULT NULL,
  `userip` binary(16) DEFAULT NULL,
  `loginTime` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
  `logout` varchar(255) DEFAULT NULL,
  `status` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=29 ;
```

Table structure for table doctorspecialization

```
CREATE TABLE IF NOT EXISTS `doctorspecialization` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `specilization` varchar(255) DEFAULT NULL,
  `creationDate` timestamp NULL DEFAULT NULL ON UPDATE CURRENT_TIMESTAMP,
  `updationDate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=10 ;
```

Table structure for table contactus

```
CREATE TABLE IF NOT EXISTS `contactus` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `fullname` varchar(255) DEFAULT NULL,
  `email` varchar(255) DEFAULT NULL,
  `contactno` bigint(12) DEFAULT NULL,
  `message` mediumtext,
  `PostingDate` date DEFAULT NULL,
  `AdminRemark` mediumtext,
  `LastupdationDate` timestamp NULL DEFAULT NULL ON UPDATE CURRENT_TIMESTAMP,
```

```
'IsRead' int(11) DEFAULT NULL,  
PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

Table structure for table medicalhistory

```
CREATE TABLE IF NOT EXISTS `medicalhistory` (  
    `ID` int(10) NOT NULL AUTO_INCREMENT,  
    `PatientID` int(10) DEFAULT NULL,  
    `BloodPressure` varchar(200) DEFAULT NULL,  
    `BloodSugar` varchar(200) NOT NULL,  
    `Weight` varchar(100) DEFAULT NULL,  
    `Temperature` varchar(200) DEFAULT NULL,  
    `MedicalPres` mediumtext,  
    `CreationDate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE  
    CURRENT_TIMESTAMP,  
    PRIMARY KEY (`ID`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

Table structure for table patient

```
CREATE TABLE IF NOT EXISTS `patient` (  
    `ID` int(10) NOT NULL AUTO_INCREMENT,  
    `Docid` int(10) DEFAULT NULL,  
    `PatientName` varchar(200) DEFAULT NULL,  
    `PatientContno` bigint(10) DEFAULT NULL,  
    `PatientEmail` varchar(200) DEFAULT NULL,  
    `PatientGender` varchar(50) DEFAULT NULL,  
    `PatientAdd` mediumtext,  
    `PatientAge` int(10) DEFAULT NULL,  
    `PatientMedhis` mediumtext,  
    `CreationDate` date DEFAULT NULL,  
    `UpdationDate` timestamp NULL DEFAULT NULL ON UPDATE CURRENT_TIMESTAMP,
```

```
PRIMARY KEY (`ID`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=6 ;
```

Table structure for table userlog

```
CREATE TABLE IF NOT EXISTS `userlog` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `uid` int(11) DEFAULT NULL,
  `username` varchar(255) DEFAULT NULL,
  `userip` binary(16) DEFAULT NULL,
  `loginTime` timestamp NULL DEFAULT CURRENT_TIMESTAMP,
  `logout` varchar(255) DEFAULT NULL,
  `status` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=30 ;
```

Table structure for table users

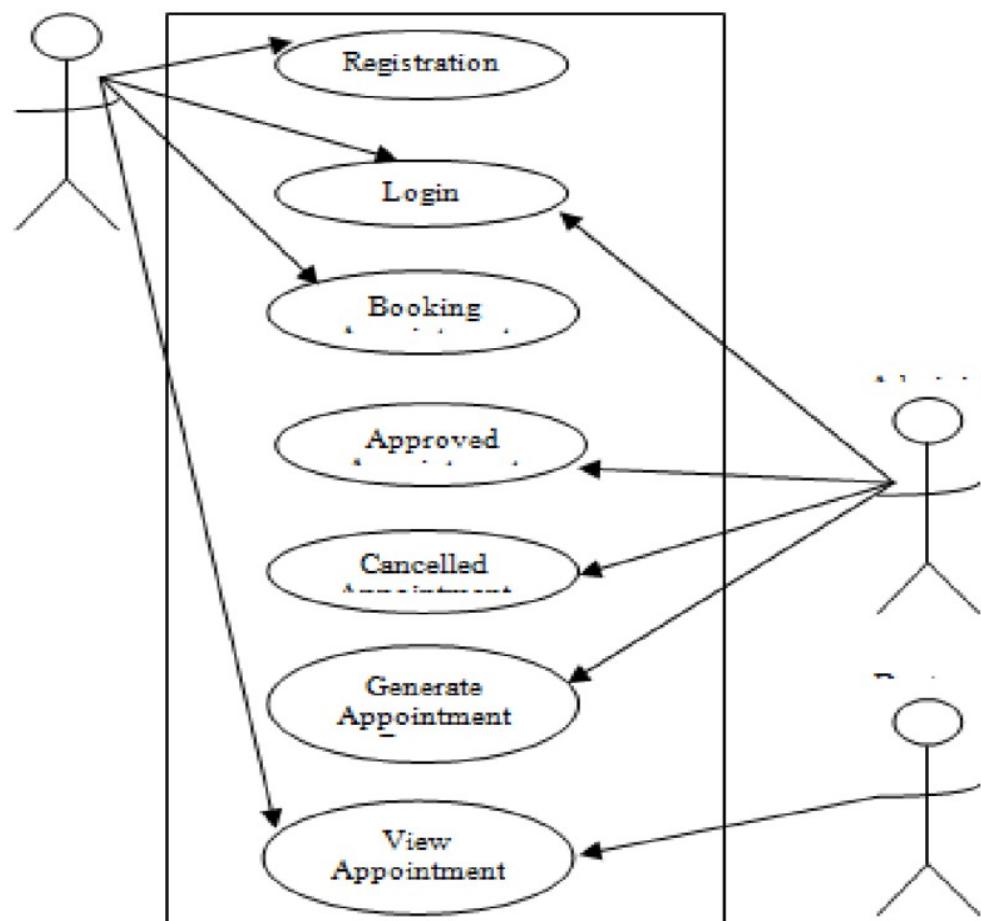
```
CREATE TABLE IF NOT EXISTS `users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `fullName` varchar(255) DEFAULT NULL,
  `address` longtext,
  `city` varchar(255) DEFAULT NULL,
  `gender` varchar(255) DEFAULT NULL,
  `email` varchar(255) DEFAULT NULL,
  `password` varchar(255) DEFAULT NULL,
  `regDate` timestamp NULL DEFAULT NULL ON UPDATE CURRENT_TIMESTAMP,
  `updationDate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `email` (`email`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

Chapter- 4

Data Flow Diagram

4.1 Data Flow Diagrams

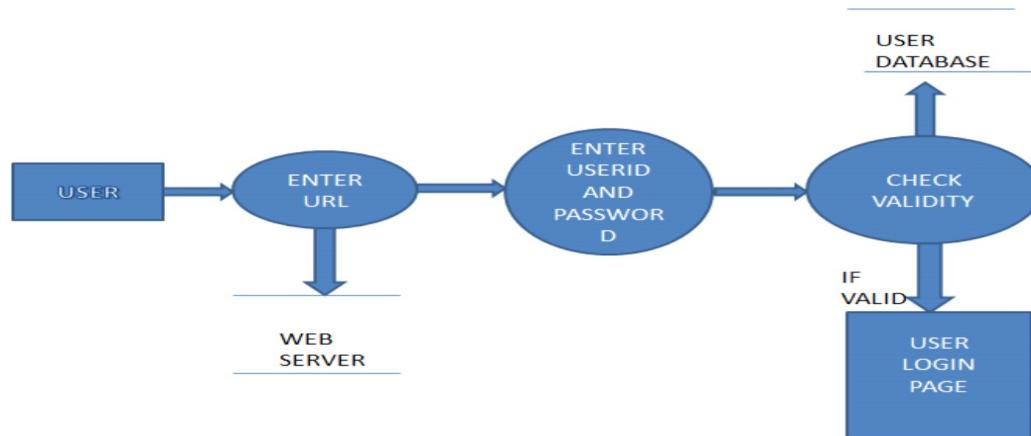
A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various subprocesses the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships. A DFD, in simple words, is a hierarchical graphical model of a system that shows the different processing activities or functions that the system performs and the data interchange among these functions. In the DFD terminology, it is useful to consider each function as a process that consumes some input data and produces some output data.



The DFD (also known as the bubble chart) is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data generated by the system. The main reason why the DFD technique is so popular is probably because of the fact that DFD is a very simple formalism- it is simple to understand and use. A DFD model uses a very limited number of primitive symbols to represent the functions performed by a system and the data flow among these functions. Starting with a set of high-level functions that a system performs, a DFD model hierarchically represents various sub functions. In fact, any hierarchical model is simple to understand. Human mind is such that it can easily understand any hierarchical model of a system-because in a hierarchical model, starting with a very simple and abstract model of a system; different details of the system can be slowly introduced through different hierarchies.

DFD for User Login

After entering to the home page of the website , student can choose the USER LOGIN option where they are asked to enter username & password , and if he/she is a valid user then a user login page will be displayed.

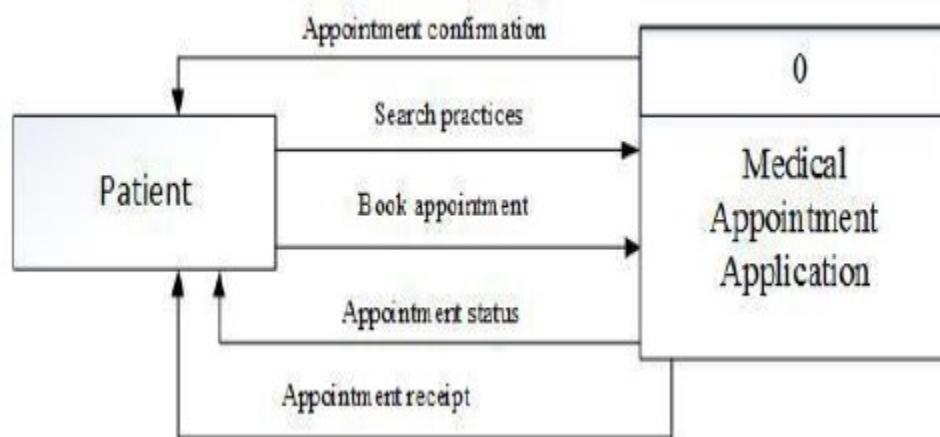


DFD for Account Creation

After the home page login there will be an option of CREATE AN ACCOUNT where after entering user detail, if all the fields are filled then a request will be sent to the librarian who will approve him as a registered member of the website.

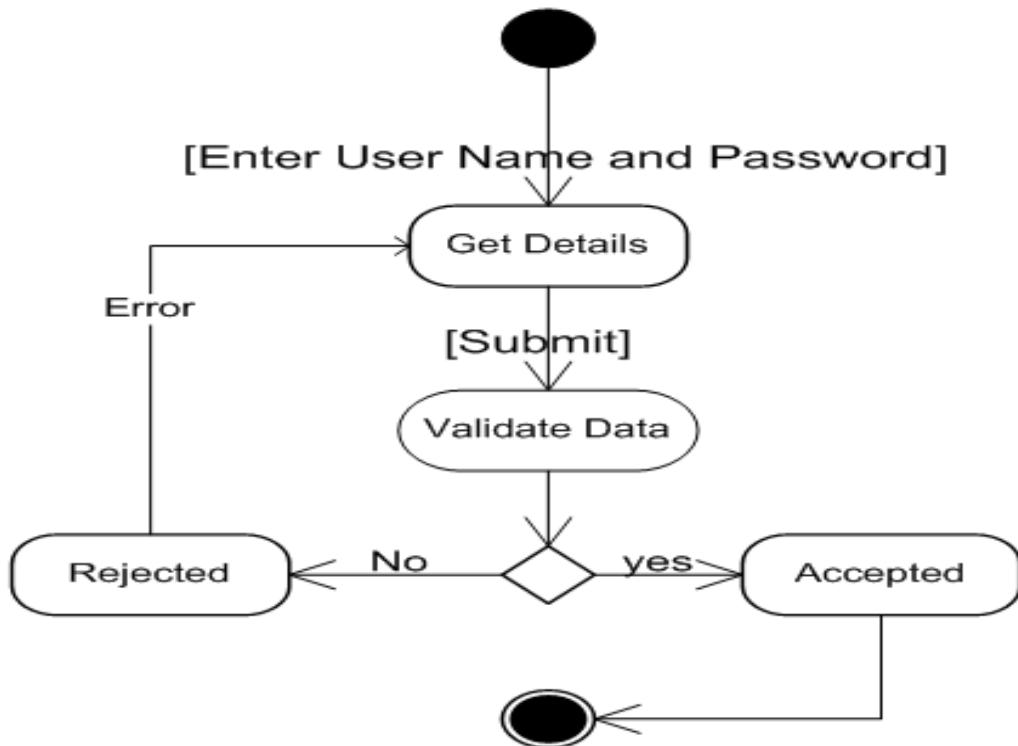
4.2 Context Diagram

The Context Diagram shows the system under consideration as a single high-level process and then shows the relationship that the system has with other external entities. Context Diagram is a Context-Level Data-Flow Diagram or a Level-0 Data Flow Diagram. Since a Context Diagram is a specialized version of Data-Flow Diagram, understanding a bit about Data-Flow Diagrams can be helpful. A Data-Flow Diagram (DFD) is a graphical visualization of the movement of data through an information system. DFDs are one of the three essential components of the structured-systems analysis and design method (SSADM).



A directed arc or an arrow is used as a data flow symbol. A data flow symbol represents the data flow occurring between two processes, or between an external entity and a process, in the direction of the data flow arrow. Data flow symbols are usually annotated with the corresponding data names.

The entity-relationship data model is based on a perception of a real world that consists of a collection of basic objects called entities and of relationships among these objects. An entity is an “object” in the real world that is distinguishable from other objects. For e.g. each customer is an entity and rooms can be considered to be entities. Entities are described by a set of attributes. The set of all entities of the same type and the set of all relationships of the same type are termed as an entity set and relationship set respectively.



A data store represents a logical file. It is represented using two parallel lines. A logical file can represent either a data store symbol which can represent either a data structure or a physical file on disk. Each data store is connected to a process by means of a data flow symbol. The direction of the data flow arrow shows whether data is being read from or written into a data store. An arrow flowing in or out of a data store implicitly represents the entire data of the data store and hence arrows connecting to a data store need not to be annotated with the name of the corresponding data items.

4.3 Use Case Diagram

A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform.

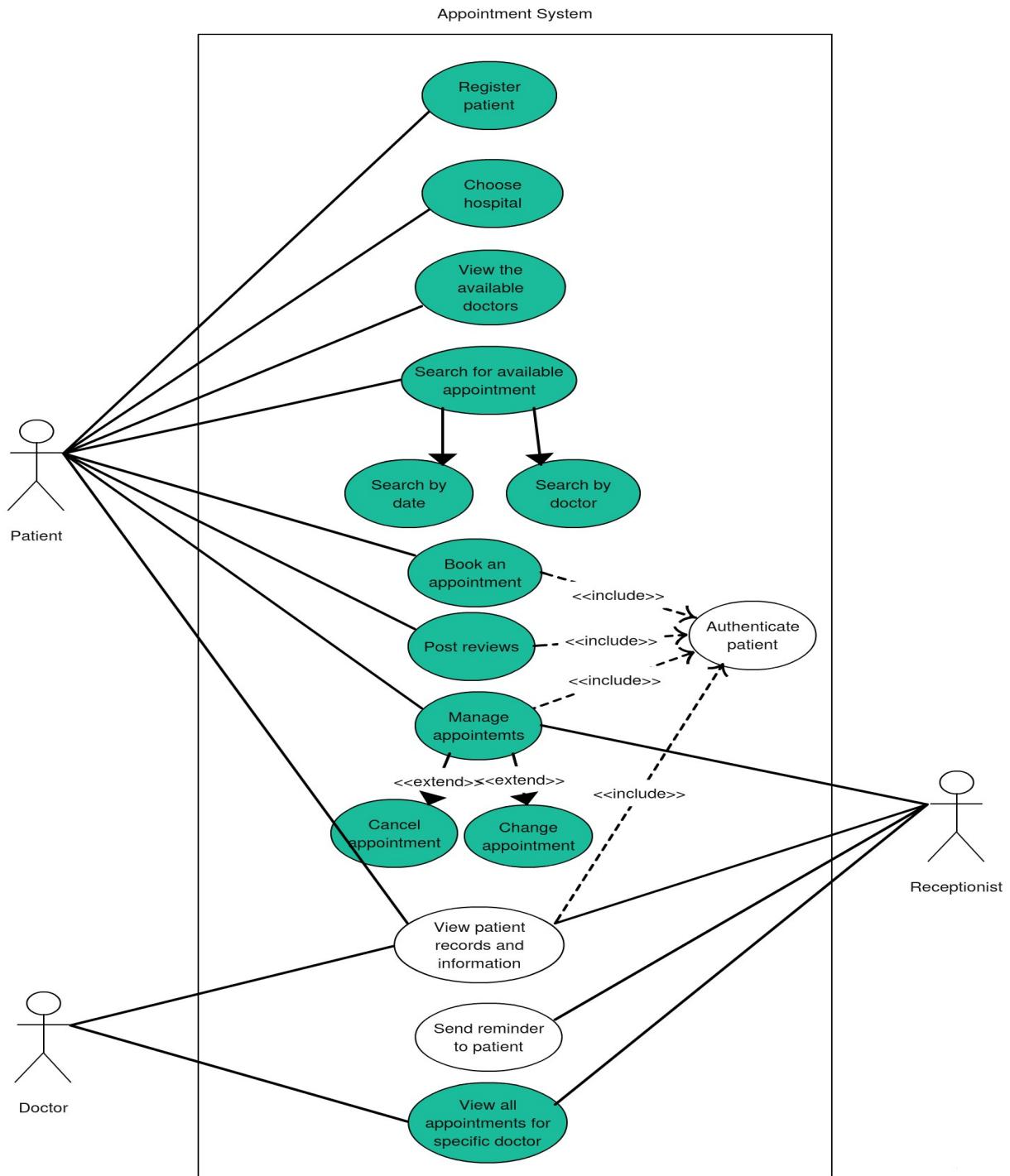


Fig-4.8 use case diagram

4.4 Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another

activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another.

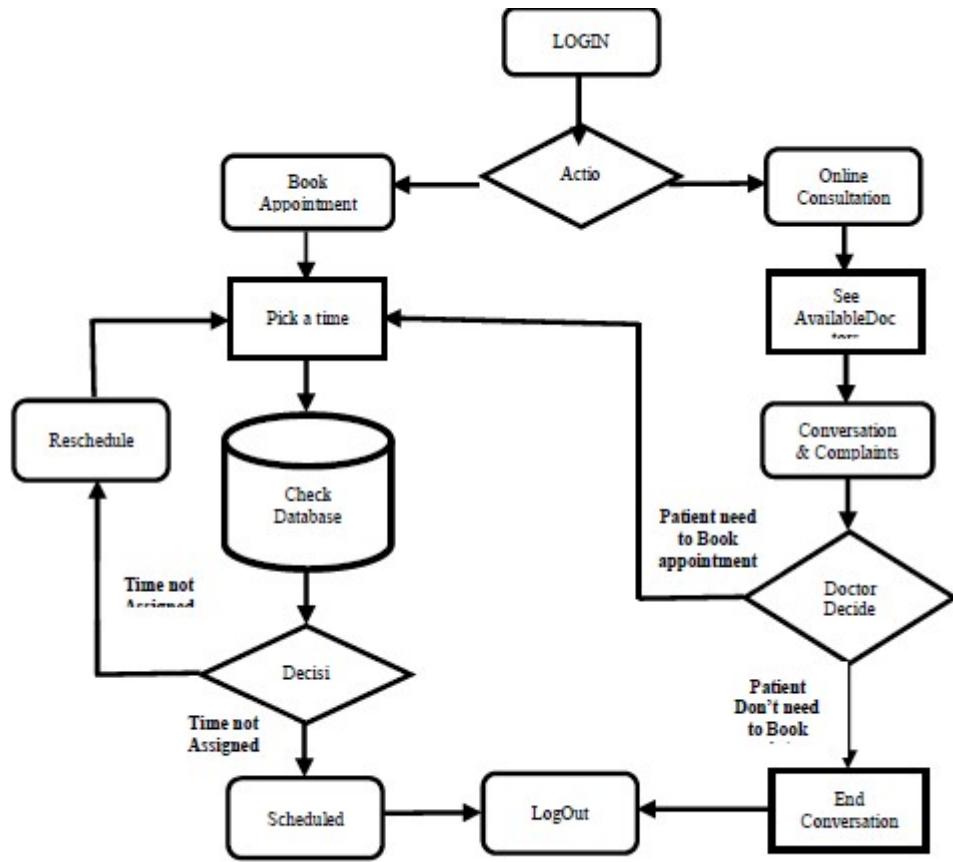


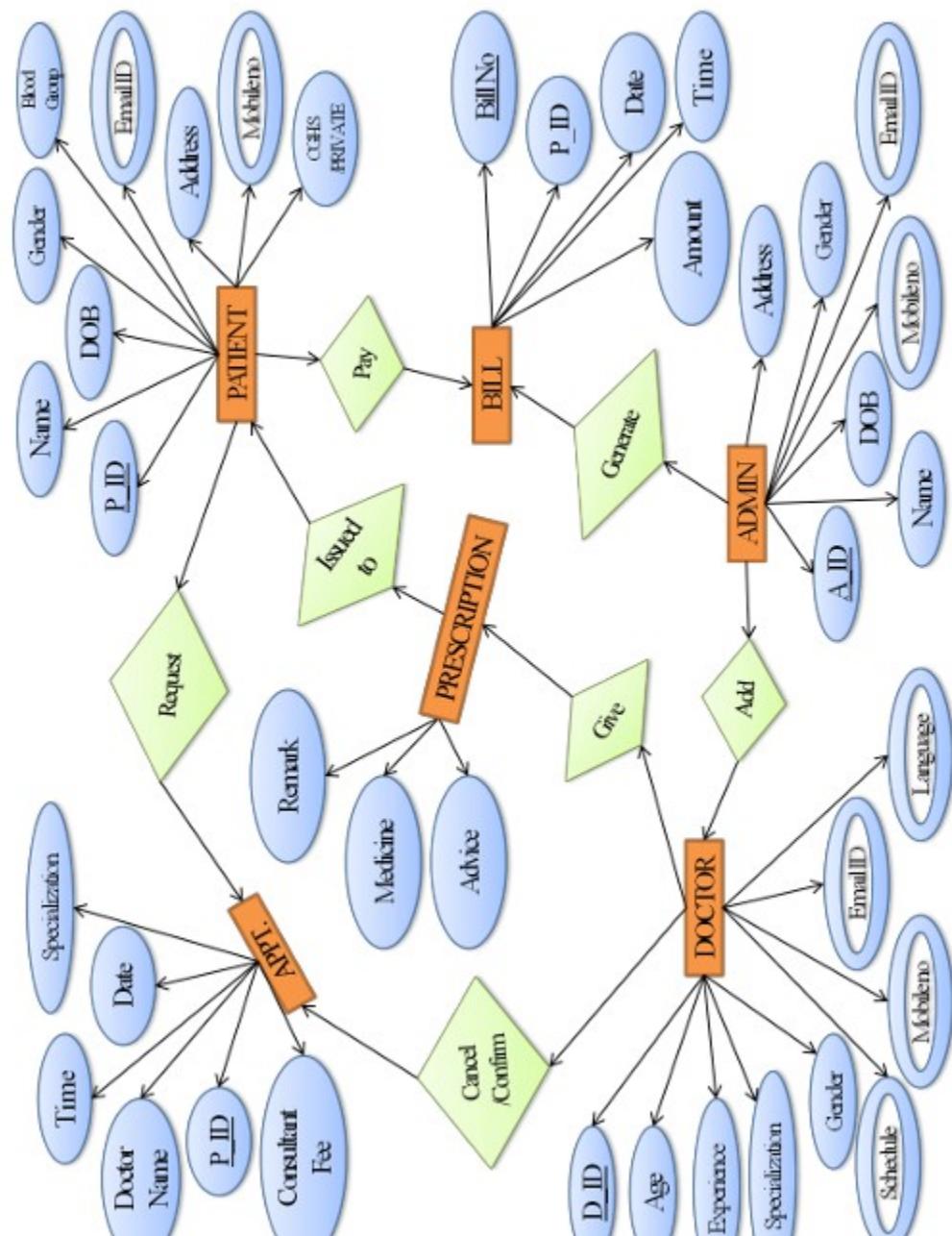
Fig-4.9 Activity diagram

4.5 Class Diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

4.6 E-R Diagrams

The Entity-Relationship (ER) model was originally proposed as a way to unify the network and relational database views. A basic component of the model is the Entity-Relationship diagram which is used to visually represents data objects. An Entity-Relationship (ER) model is a conceptual data model that defines the structure of a database. It represents data as entities, attributes, and relationships. For an HMS, the ER model helps in organizing various types of data (e.g., patients, doctors, appointments) in a way that reflects real-world interactions within a hospital.



Chapter-5

Testing Analysis and Evaluation

5.1 Testing Analysis and Evaluation

Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is Defect free. It involves execution of a software component or system component to evaluate one or more properties of interest. Software testing also helps to identify errors, gaps or missing requirements in contrary to the actual requirements. It can be either done manually or using automated tools.

Test Analysis is the process of looking into test artifacts to base your test conditions/test cases. Hence, it is also called Test Basis. Testers can create Test Conditions by looking into the Application Under Test or use their experience. But mostly, test cases are derived from test artifacts. Testing is a set of activities that can be planned in advanced and conducted systematically. A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements.

5.1.1 Unit Testing

Unit testing focuses verification efforts on the smallest unit of software design module. The unit test is always white box oriented. The tests that occur as part of unit testing are testing the module interface, examining the local data structures, testing the boundary conditions, execution all the independent paths and testing error-handling paths. Using the details design description as a guide, important control paths are tested to uncover errors within boundary of the module. The relative complexity of tests and the errors detected as a result is limited by the constrained scope established for unit testing. Unit testing is normally considered an adjacent to coding steps. After source level code has been developed, reviewed, and verified for correct syntax, unit test case design begins. A review of design information provides guidance for establishing test cases that are likely to uncover error in each case of the categories discussed above. Each test case should be coupled with a set of expected results.

5.1.2 Integration Testing

Integration testing is a systematic technique or construction the program structure while at the same time conducting tests to uncover errors associated with interfacing. Scope of testing summarizes the specific functional, performance, and internal design characteristics that are to be tested. It employs top-down testing and bottom-up testing methods for this case. The objective is to take unit tested modules and build a program structure that has been dictated by design. There is often a tendency to attempt no incremental integration; that is to construct the program using a “big bang” approach. The entire modules are combined in advance. The entire program is tested as whole and chaos usually result! A set of error is encountered.

5.1.3 Performance Testing

Performance testing is the process of determining the speed, responsiveness and stability of a computer, network, software program or device under a workload. Performance testing can involve quantitative tests done in a lab, or occur in the production environment in limited scenarios. Different testing techniques require a different set of skill by testers. While performing Black box testing, the tester does not require the knowledge of coding. A good knowledge of JavaScript and SQL commands comes handy to avert such software security threats. Timing for both read and update transactions should be gathered to determine whether system functions are being performed in an acceptable timeframe.

5.1.4 System Testing

Quality assurance is an important step in software engineering. This overlaps with all the phases of development right from the requirement analysis. This quality requirement of the software system must be clearly extracted during the requirement analysis and all the subsequent phases should be made biased to that, the final testing will become trivial and less expensive. There are number of quality parameters like correctness, accuracy, reliability, robustness, efficiency, effectiveness, reusability, maintainability etc..

5.1.5 Boundary Conditions Test

Boundary value analysis is a type of black box or specification based testing technique in which tests are performed using the boundary values. Boundary conditions as in case of generating

sequences were tested to ensure that the module operates properly at boundaries establish to limit or restrict processing.

5.1.6 Acceptance Testing

Acceptance testing, a testing technique performed to determine whether or not the software system has met the requirement specifications. The main purpose of this test is to evaluate the system's compliance with the business requirements and verify if it has met the required criteria for delivery to end users.

5.1.7 Accessibility Testing

Accessibility testing is a subset of usability testing where the users under consideration people with all abilities and disabilities are. The significance of this testing is to verify both usability and accessibility.

5.1.8 Integration Test

The objective of Integration Test is to take the until tested modules and build a program structure that has been defined in the design. We have done top down integration, which is constructing and testing small segments where errors are easier to isolate, and corrected

5.1.9 Block Box Testing

Black box testing, also known as Behavioral Testing, is a software testing method in which the internal structure/design/implementation of the item being tested is not known to the tester. These tests can be functional or non-functional, though usually functional. This method is named so because the software program, in the eyes of the tester, is like a black box.

5.1.10 Validation Testing

At the culmination of integration testing, software is completely assembled as a package, interfacing errors have been uncovered and corrected, and a final series of software tests namely validation tests are performed. Validation succeeds when the software functions in the manner that can be easily accepted by the customer. After validation test has been conducted, one of the possible condition is satisfied. The functions or performance characteristics confirmed to specifications are acceptable.

Chapter- 6

System Implementation and Maintenance

6.1 System Implementation

System implementation involves deploying the developed Organ Donation Management System (ODMS) into the live environment where it is accessible to end-users. The implementation phase is crucial to ensure that the system functions as intended.

Steps in Implementation:

1. Planning:

- Identify the implementation strategy (e.g., phased, parallel, direct cutover, or pilot).
- Define the timeline and resource allocation.

2. System Installation:

- Install the application on servers or cloud infrastructure.
- Set up databases and configure necessary connections.

3. Data Migration:

- Transfer data from existing systems to the new ODMS.
- Validate data accuracy and integrity.

4. User Training:

- Provide training to donors, hospitals, legal authorities, and administrators.
- Conduct hands-on workshops and provide user manuals.

5. Testing in Production:

- Perform final round testing in the live environment to ensure all components are functioning.

6. Go Live:

- Launch the system for real-world use.
- Provide initial support to monitor performance and troubleshoot issues.

6.2 System Maintenance

System maintenance involves the ongoing process of ensuring the ODMS runs smoothly by addressing bugs, improving performance, and implementing necessary updates. Maintenance ensures the system remains reliable and up-to-date.

Types of Maintenance:

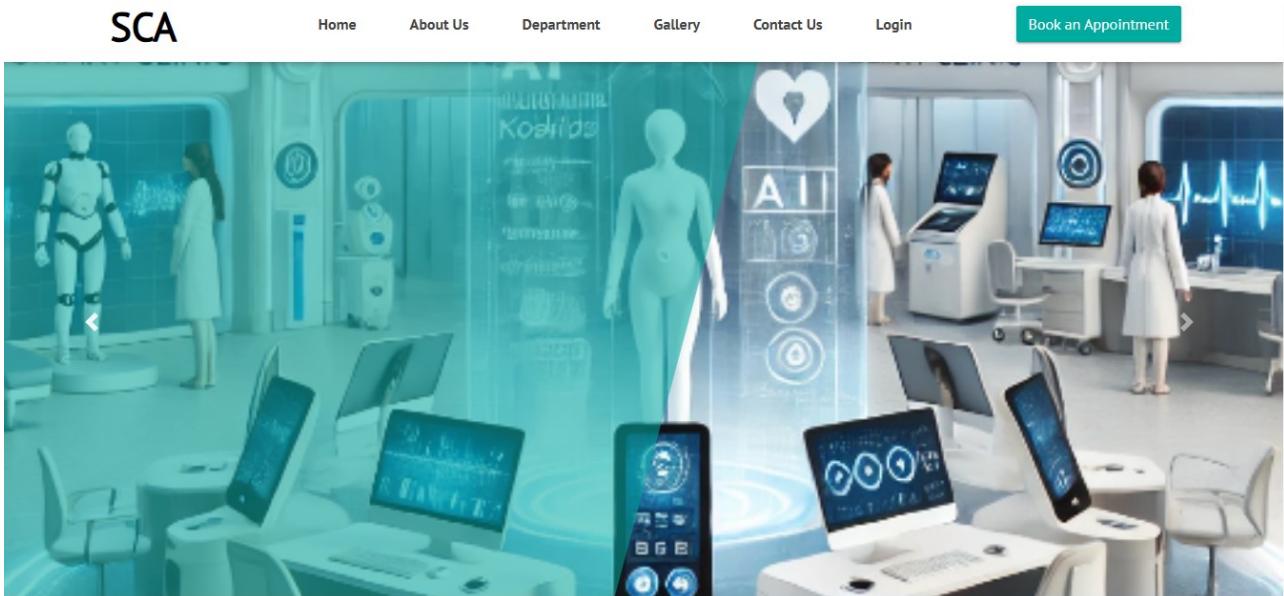
- 1. Corrective Maintenance:**
 - Fixing bugs and issues reported by users or identified through monitoring.
- 2. Adaptive Maintenance:**
 - Updating the system to remain compatible with changing legal, regulatory, or operational requirements.
- 3. Perfective Maintenance:**
 - Enhancing existing features, improving performance, and refining the user experience.
- 4. Preventive Maintenance:**
 - Proactively monitoring system performance to identify and resolve potential issues before they become critical.

6.3 Maintenance Activities:

- Monitoring:** Continuously monitor server performance, database health, and user activity.
- Backup and Recovery:** Schedule regular backups and ensure disaster recovery plans are in place.
- Security Management:** Apply software patches, monitor for cyber threats, and perform security audits.
- User Support:** Provide technical support, resolve user-reported issues, and manage user feedback.
- Performance Tuning:** Optimize database queries and improve application response times.

Chapter-7

Output Layout Screen



Index Page



About Our Smart Clinic Automation

Welcome to Smart Clinic Automation System is a sophisticated technological solution designed to streamline healthcare operations, enhance patient experiences, and optimize clinical workflows. Leveraging advanced technologies like Artificial Intelligence (AI), the Internet of Things (IoT), cloud computing, and data analytics, this system offers seamless automation of various administrative and medical processes.

The future of healthcare management. We are a cutting-edge technology-driven solution designed to revolutionize how clinics, hospitals, and healthcare facilities operate. Our goal is to bring automation, efficiency, and digital transformation to medical practices, ensuring better patient care and smoother workflows. At Smart Clinic Automation, we believe that technology can bridge the gap between healthcare providers and patients, creating a seamless, efficient, and patient-friendly experience. Our system integrates Artificial Intelligence (AI), Machine Learning (ML), the Internet of Things (IoT), Cloud Computing, and Big Data Analytics to streamline clinic operations, enhance diagnostics, and improve decision-making. Our Vision Our vision is to transform traditional healthcare systems into smart, digital, and automated environments where doctors, nurses, and administrative staff can focus more on patient care.

about Page

Departments

Take a look at some of our departments



Cardiology



Orthopaedic



Neurologist



General Medicine



Pediatrics



High Quality Treatments

Department page

[All](#) [Dental](#) [Cardiology](#) [Neurology](#) [Laboratory](#)



Gallery Page

Get In Touch

Name :

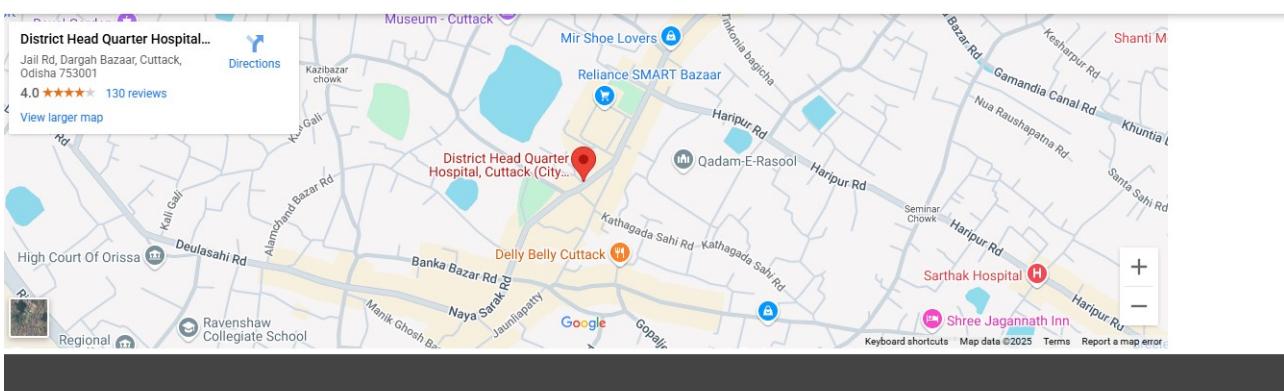
Email :

Mobile :

Message:

[Send Message](#)

Contact Page



Useful Links

[About us](#)[Services](#)

Get In Touch

Mission Road Buxibazar Cuttack-753001

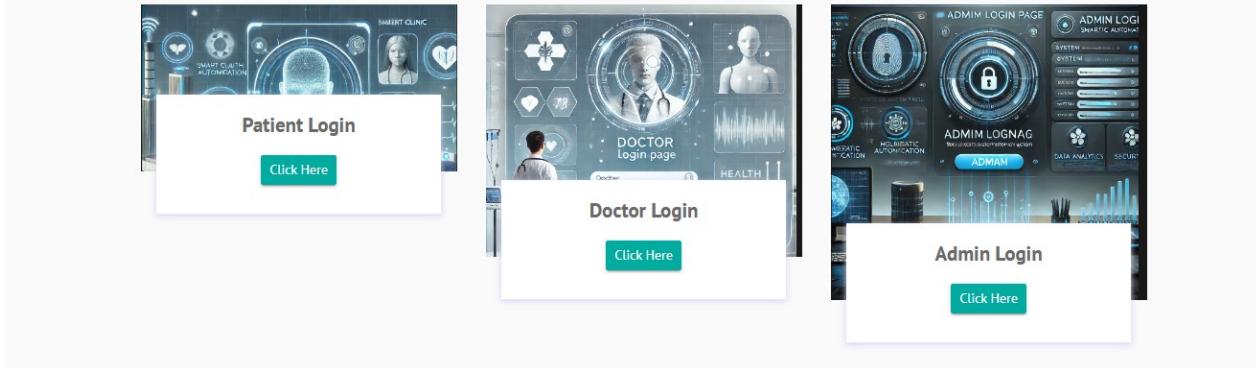
Phone: 9853274179

Email: care@gmail.com

Timing: 8-Am to 12-Pm and 3pm to 5pm

GoogleMap Page

All Login Panel Here



All Logins Page

Patient Registration

Sign Up

Mamono

Buxibaza

Cuttack

Gender

Female Male

mamono@gmail.com
Email available for Registration .

.....

I agree

Already have an account? [Log-in](#)

[Submit ➔](#)

Patient Registration Page

localhost says

Successfully Registered. You can login now

OK

Patient Registration confirm page

Goto Website

Patient Login Panel

 mamono@gmail.com

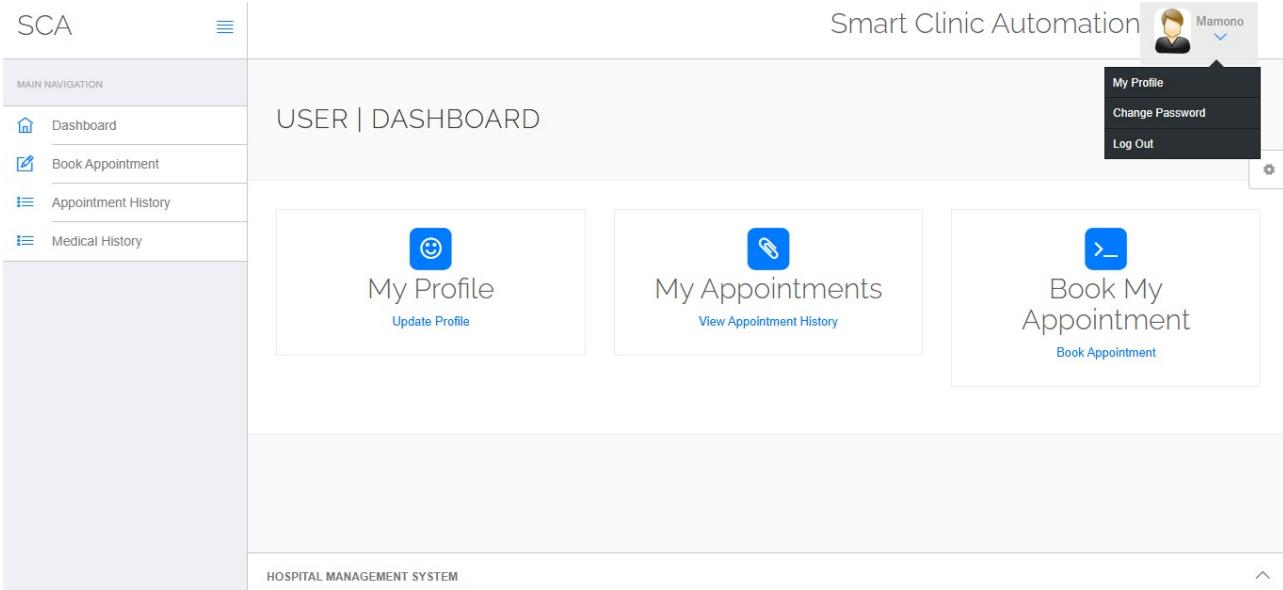


[Forgot Password ?](#)

Don't have an account yet? [Create an account](#)

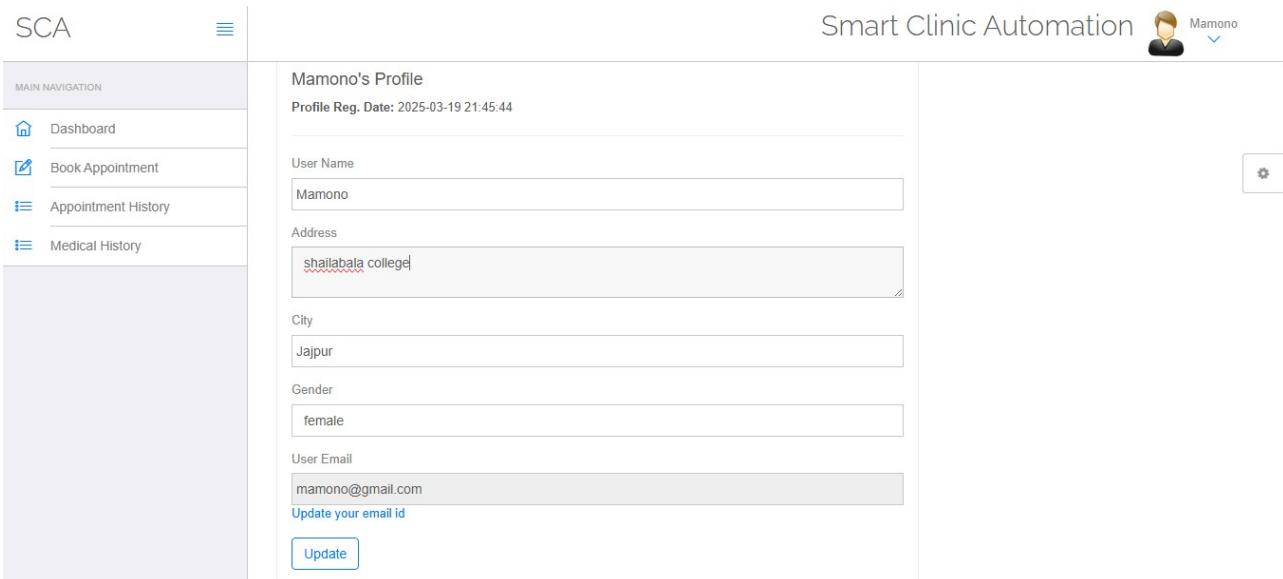
SMART CLINIC AUTOMATION.

Patient login page



The screenshot shows the Smart Clinic Automation patient dashboard. At the top right, there's a user profile icon for 'Mamono' with a dropdown menu containing 'My Profile', 'Change Password', and 'Log Out'. The main title 'USER | DASHBOARD' is centered above three cards: 'My Profile' (with a smiley face icon), 'My Appointments' (with a clipboard icon), and 'Book My Appointment' (with a right-pointing arrow icon). A sidebar on the left lists 'MAIN NAVIGATION' items: Dashboard, Book Appointment, Appointment History, and Medical History. The bottom of the page has a footer bar with the text 'HOSPITAL MANAGEMENT SYSTEM'.

Patient dashboard page



The screenshot shows the 'Patient profile modification page'. It features a sidebar with 'MAIN NAVIGATION' items: Dashboard, Book Appointment, Appointment History, and Medical History. The main content area displays 'Mamono's Profile' with a registration date of '2025-03-19 21:45:44'. It includes input fields for 'User Name' (Mamono), 'Address' (shailabala college), 'City' (Jajpur), 'Gender' (female), and 'User Email' (mamono@gmail.com). There is also a link 'Update your email id' and a blue 'Update' button. The top right corner shows the user profile 'Mamono' with a dropdown menu and a gear icon.

Patient profile modification page

Book Appointment

Doctor Specialization
Obstetrics and Gynecology

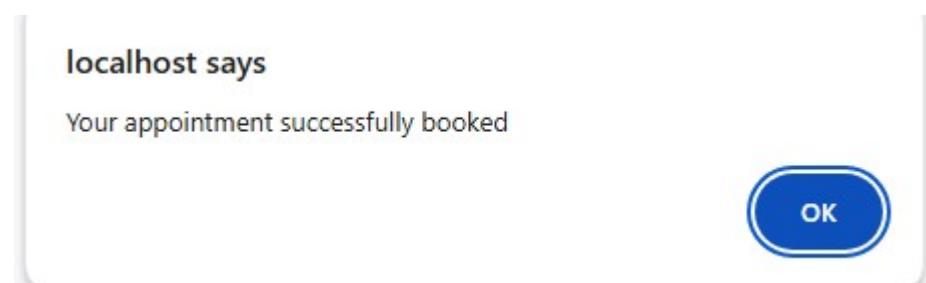
Doctors
Dr. Rashmita Ray

Consultancy Fees
400

Date
2025-03-23

Time
10:00 AM
eg : 10:00 PM

Patient book appointment page



Patient book appointment confirm page

SCA

MAIN NAVIGATION

- Dashboard
- Book Appointment
- Appointment History
- Medical History

Smart Clinic Automation Mamono

User / Appointment History

USER | APPOINTMENT HISTORY

#	Doctor Name	Specialization	Consultancy Fee	Appointment Date / Time	Appointment Creation Date	Current Status	Action
1.	Dr. Rashmita Ray	Obstetrics and Gynecology	400	2025-03-23 / 10:00 AM	2025-03-19 21:52:36	Active	<input type="button" value="Cancel"/>

Patient book appointment view page

[Goto Website](#)

Doctor Login Panel

[Forgot Password ?](#)

[Login](#)

Doctor login page

SCA

MAIN NAVIGATION

- Dashboard
- Appointment History
- Patients
- Search

Smart Clinic Automation Dr. Rashmita Ray

User / Dashboard

DOCTOR | DASHBOARD

[My Profile](#) [View Appointment History](#)

[My Appointments](#)

SCA

MAIN NAVIGATION

- Dashboard
- Appointment History
- Patients
- Search

Smart Clinic Automation Dr. Rashmita Ray

Doctor / Appointment History

DOCTOR | APPOINTMENT HISTORY

#	Patient Name	Specialization	Consultancy Fee	Appointment Date / Time	Appointment Creation Date	Current Status	Action
1.	Mamono	Obstetrics and Gynecology	400	2025-03-23 / 10:00 AM	2025-03-19 21:52:36	Active	Cancel

Doctor dashboard page

SCA

Smart Clinic Automation Dr. Rashmita Ray

Patient Name: Arpita

Patient Contact no: 7978403417

Patient Email: arpita@gmail.com

Email available for Registration.

Gender: Female Male

Patient Address: Mission Road, Cuttack

Patient Age: 20

Medical History: Normal fever

Doctor add patient page

SCA

Smart Clinic Automation Dr. Rashmita Ray

DOCTOR | MANAGE PATIENTS

Doctor / Manage Patients

Manage Patients

Patient Details			
Patient Name	Arpita	Patient Email	arpita@gmail.com
Patient Mobile Number	7978403417	Patient Address	Mission Road, Cuttack
Patient Gender	female	Patient Age	20
Patient Medical History(if any)	Normal fever	Patient Reg Date	2025-03-19 22:03:13

Medical History						
#	Blood Pressure	Weight	Blood Sugar	Body Temperature	Medical Prescription	Visit Date

Add Medical History

Doctor view patient page

Add Medical History

Blood Pressure :	120/79
Blood Sugar :	96/110
Weight :	45
Body Temperature :	94
Prescription :	Rx. 1. Paracetamol 2. Nimesulide 3. Diclofenac 4. Disprin

Doctor add medicine page

SCA

Smart Clinic Automation Dr. Rashmita Ray

MAIN NAVIGATION

- Dashboard
- Appointment History
- Patients
- Search

Manage Patients

Patient Details				
Patient Name	Arpita	Patient Email	arpita@gmail.com	
Patient Mobile Number	7978403417	Patient Address	Mission Road, Cuttack	
Patient Gender	female	Patient Age	20	
Patient Medical History(if any)	Normal fever	Patient Reg Date	2025-03-19 22:03:13	

Medical History						
#	Blood Pressure	Weight	Blood Sugar	Body Temperature	Medical Prescription	Visit Date
1	120/79	45	96/110	94	Rx. 1. Paracetamol 2. Nimesulide 3. Diclofenac 4. Disprin	2025-03-19 22:08:46

Doctor view history page

SCA

MAIN NAVIGATION

- Dashboard
- Appointment History
- Patients
- Search

Doctor Specialization
Obstetrics and Gynecology

Doctor Name
Dr. Rashmita Ray

Doctor Clinic Address
Mangalabag Cuttack

Doctor Consultancy Fees
400

Doctor Contact no
9853274176

Doctor Email
rashmi@gmail.com

[Update](#)



Dr. Rashmita Ray



Doctor profile modification page

Admin Login Panel

 admin



[Login !\[\]\(813ececbaa3655d7974515416839bd69_img.jpg\)](#)

[Goto Website](#)

Admin login page



Change Password

Log Out

ADMIN | DASHBOARD

MAIN NAVIGATION	
	Dashboard
	Doctors
	Users
	Patients
	Appointment History
	Contactus Queries
	Doctor Session Logs
	User Session Logs
	Reports
	Pages
	Patient Search

 Online Booking
Total : 2

 Doctors
Total : 7

 Appointments
Total : 4

 Patients
Total : 4

 Messages
Total : 1

HOSPITAL MANAGEMENT SYSTEM

Admin dashboard page



Manage Doctor Specialization

#	Specialization	Creation Date	Updation Date	Action
1.	Orthopedics	2025-03-09 23:39:46	2025-03-09 14:56:47	
2.	Medicine	2025-03-09 23:40:49	2025-03-09 14:57:59	
3.	Obstetrics and Gynecology	2025-03-09 23:40:09	0000-00-00 00:00:00	
4.	Dermatology	2025-03-09 23:40:11	0000-00-00 00:00:00	
5.	Pediatrics	2025-03-09 23:40:13	0000-00-00 00:00:00	
6.	Radiology	2025-03-09 23:40:16	0000-00-00 00:00:00	
7.	General Surgery	2025-03-09 23:40:19	0000-00-00 00:00:00	
8.	Ophthalmology	2025-03-09 23:40:23	0000-00-00 00:00:00	
9.	Anesthesia	2025-03-09 23:40:29	0000-00-00 00:00:00	
10.	Pathology	2025-03-09 23:40:35	0000-00-00 00:00:00	
11.	ENT	2025-03-09 23:40:40	0000-00-00 00:00:00	
12.	Dental Care	2025-03-09 23:40:55	0000-00-00 00:00:00	
13.	Dermatologists	2025-03-09 23:41:35	0000-00-00 00:00:00	

Admin view doctor specialization page

MAIN NAVIGATION	
	Dashboard
	Doctors
	Users
	Patients
	Appointment History
	Consultant Queries
	Doctor Session Logs
	User Session Logs
	Reports
	Pages
	Patient Search

Manage Doctors

#	Specialization	Doctor Name	Date of Joining	Action
1.	ENT	Dr.Anuradha Nayak	2025-03-10 23:46:52	 
2.	Endocrinologists	Dr. Anvi Anwesha Ray	2025-03-11 06:36:41	 
3.	Pediatrics	Dr. Soudamini Ray	2025-03-15 14:42:23	 
4.	Orthopedics	Dr. Manoj Kumar Ray	2025-03-15 14:43:11	 
5.	Medicine	Dr. Bimal Kumar Ray	2025-03-15 14:44:11	 
6.	Obstetrics and Gynecology	Dr. Rashmita Ray	2025-03-15 14:45:18	 
7.	General Surgery	Dr. Nirmal Ray	2025-03-15 17:47:01	 

Admin view all doctor page

MAIN NAVIGATION	
	Dashboard
	Doctors
	Users
	Patients
	Appointment History
	Consultant Queries
	Doctor Session Logs
	User Session Logs
	Reports
	Pages
	Patient Search

Doctor Details updated Successfully

Edit Doctor info

Dr.Anuradha Ray's Profile

Profile Reg. Date: 2025-03-10 23:46:52
Profile Last Updation Date: 2025-03-19 22:32:00

Doctor Specialization

Doctor Name

Doctor Clinic Address

Doctor Consultancy Fees

Admin update doctor profile page

MAIN NAVIGATION	
	Dashboard
	Doctors
	Users
	Patients
	Appointment History
	Contactus Queries
	Doctor Session Logs
	User Session Logs
	Reports
	Pages
	Patient Search

PATIENTS | APPOINTMENT HISTORY

Patients / Appointment History

#	Doctor Name	Patient Name	Specialization	Consultancy Fee	Appointment Date / Time	Appointment Creation Date	Current Status	Action
1.	Dr. Anvi Anwesha Ray	Bimal Kumar Ray	Endocrinologists	800	2025-03-17 / 5:00 PM	2025-03-15 16:57:53	Active	No Action yet
2.	Dr. Rashmita Ray	Mamono	Obstetrics and Gynecology	400	2025-03-23 / 10:00 AM	2025-03-19 21:52:36	Active	No Action yet

Admin view all appointment history

MAIN NAVIGATION	
	Dashboard
	Doctors
	Users
	Patients
	Appointment History
	Contactus Queries
	Doctor Session Logs
	User Session Logs
	Reports
	Pages
	Patient Search

Manage Patients

Patient Details

Patient Name	Arpita	Patient Email	arpita@gmail.com
Patient Mobile Number	7978403417	Patient Address	Mission Road, Cuttack
Patient Gender	female	Patient Age	20
Patient Medical History(if any)	Normal fever	Patient Reg Date	2025-03-19 22:03:13

Medical History

#	Blood Pressure	Weight	Blood Sugar	Body Temperature	Medical Prescription	Visit Date
1	120/79	45	96/110	94	Rx. 1. Paracetamol 2. Nimesulide 3. Diclofenac 4. Disprin	2025-03-19 22:08:46

Admin view patient personal information

HMS



Hospital Management System



Admin 

MAIN NAVIGATION

-  Dashboard
-  Doctors <
-  Users <
-  Patients <
-  Appointment History
-  Contact Us <
-  Doctor Session Logs
-  User Session Logs
-  Reports <
-  Pages <
-  Patient Search

ADMIN | VIEW PATIENTS

Admin / View Patients

Search by Name/Mobile No.

[Search](#)

Result against "7978403417" keyword

#	Patient Name	Patient Contact Number	Patient Gender	Creation Date	Updation Date	Action
1.	Arpita	7978403417	female	2025-03-19 22:03:13		View

Admin searching patient

Admin add about us page

BETWEEN DATES REPORTS

MAIN NAVIGATION

- [Dashboard](#)
- [Doctors](#)
- [Users](#)
- [Patients](#)
- [Appointment History](#)
- [Consultus Queries](#)
- [Doctor Session Logs](#)
- [User Session Logs](#)
- [Reports](#)
- [Pages](#)
- [Patient Search](#)

Between Dates Reports

From Date:

01-Mar-2025

To Date:

19-Mar-2025

[Submit](#)

Admin view report between date wise

ADMIN | VIEW PATIENTS

MAIN NAVIGATION

- [Dashboard](#)
- [Doctors](#)
- [Users](#)
- [Patients](#)
- [Appointment History](#)
- [Consultus Queries](#)
- [Doctor Session Logs](#)
- [User Session Logs](#)
- [Reports](#)
- [Pages](#)
- [Patient Search](#)

Between dates reports

Report from 2025-03-01 to 2025-03-19

#	Patient Name	Patient Contact Number	Patient Gender	Reg. Date-Time	Updation Date-Time	Action
1.	Rahul	9876543218	male	2025-03-14 10:53:35		View
2.	Amit	9999999999	male	2025-03-14 14:31:26		View
3.	bimal ray	23456789	Male	2025-03-15 17:00:54	2025-03-19 22:45:34	View
4.	Arpita	7978403417	female	2025-03-19 22:03:13		View

Admin view report between date wise

MAIN NAVIGATION	
	Dashboard
	Doctors
	Users
	Patients
	Appointment History
	Contactus Queries
	Doctor Session Logs
	User Session Logs
	Reports
	Pages
	Patient Search

ADMIN | VIEW PATIENTS

Admin / View Patients

Between dates reports

Report from 2025-03-19 to 2025-03-19

#	Patient Name	Patient Contact Number	Patient Gender	Reg. Date-Time	Updation Date-Time	Action
1.	Arpita	7978403417	female	2025-03-19 22:03:13		<button>View</button>

Admin view report between day wise

MAIN NAVIGATION	
	Dashboard
	Doctors
	Users
	Patients
	Appointment History
	Contactus Queries
	Doctor Session Logs
	User Session Logs
	Reports
	Pages
	Patient Search

Manage Query Details

Hospital Management System

Full Name	BIMALKUMAR RAY
Email Id	bimal@hotmail.com
Contact Numner	9853274179
Message	sample testing
Query Date	2025-03-15 17:25:30
Admin Remark	Thank you for your valuable comment!

Update

Admin Reply for Message

Chapter-8

System Security

8.1 System Security

The protection of computer based resources that includes hardware, software, data, procedures and people against unauthorized use or natural Disaster is known as System Security.

System Security can be divided into four related issues:

- **System Security** refers to the technical innovations and procedures applied to the hardware and operation systems to protect against deliberate or accidental damage from a defined threat.
- **Data Security** is the protection of data from loss, disclosure, modification and destruction.
- **System Integrity** refers to the proper functioning of hardware and programs, appropriate physical security and safety against external threats such as eavesdropping and wiretapping.
- **Privacy** defines the rights of the user or organizations to determine what information they are willing to share with or accept from others and how the organization can be protected against unwelcome, unfair or excessive dissemination of information about it.
- **Confidentiality** is a special status given to sensitive information in a database to minimize the possible invasion of privacy. It is an attribute of information that characterizes its need for protection.

8.2 Security in Software

System security refers to various validations on data in form of checks and controls to avoid the system from failing. It is always important to ensure that only valid data is entered and only valid operations are performed on the system. The system employs two types of checks and controls:

8.3 Client Side Validation

Various client side validations are used to ensure on the client side that only valid data is entered. Client side validation saves server time and load to handle invalid data. Some checks imposed are:

- VBScript is used to ensure those required fields are filled with suitable data only. Maximum lengths of the fields of the forms are appropriately defined.
- Forms cannot be submitted without filling up the mandatory data so that manual mistakes of submitting empty fields that are mandatory can be sorted out at the client side to save the server time and load.
- Tab-indexes are set according to the need and taking into account the ease of user while working with the system.

8.4 Server Side Validation

Some checks cannot be applied at client side. Server side checks are necessary to save the system from failing and intimating the user that some invalid operation has been performed or the performed operation is restricted. Some of the server side checks imposed is:

- Server side constraint has been imposed to check for the validity of primary key and foreign key. A primary key value cannot be duplicated. Any attempt to duplicate the primary value results into a message intimating the user about those values through the forms using foreign key can be updated only of the existing foreign key values.
- User is intimating through appropriate messages about the successful operations or exceptions occurring at server side.
- Various Access Control Mechanisms have been built so that one user may not agitate upon another. Access permissions to various types of users are controlled according to the organizational structure. Only permitted users can log on to the system and can have access according to their category. User- name, passwords and permissions are controlled o the server side.
- Using server side validation, constraints on several restricted operations are imposed.

8.5 Conclusion

The project is known as **Hospital management System**. Most of the work are done through computer system these days so, this application contains modules like patient accounts, doctors accounts, admin accounts as primary users. This will help patients to book an appointment without visiting to hospital physically through online mode. This application contains various records like doctor availability, medicine availability, bed availability, medical history of patients, billings and many more. It is true that Hospital management System have completely changed the health care system. It is flexible we can make any changes by adding more features according to the new needs in health care fields.

8.6 Future Improvement

This system will be updated with many more features in future. This system will have inbuilt AI to enhance patient care. In future feature like real time video calls will be added to interact with doctors without visiting them physically which enable remote consultations, monitoring of patients with chronic conditions. In future with increasing in digitalization of health care we should also have to address the evolving cybersecurity challenges. In future we will add more modules to this application for more benefits

- **Predictive Analytics:** Implement AI-driven models to predict patient outcomes, optimize resource allocation, and forecast patient admissions.
- **Integration with Wearable Devices:** Enable integration with IoT devices and wearable health monitors to capture real-time patient data for continuous monitoring.
- **Decision Support Systems:** Use machine learning to provide clinical decision support by analyzing patient data in real time.
- **Automation of Routine Tasks:** Automate administrative tasks such as appointment scheduling, billing, and inventory management using intelligent algorithms.
- **Personal Health Records:** Enhance patient portals to offer more personalized health records, educational resources, and health tracking tools.
- **Patient Engagement:** Integrate interactive features, such as appointment reminders, health tips
- **Automation of Routine Tasks:** Automate administrative tasks such as appointment scheduling, billing, and inventory management using intelligent algorithms.

8.7 Bibliography

Text Books Referred

1. **HTML and CSS: Design and Build Websites** by Jon Duckett
2. **JavaScript and JQuery: Interactive Front-End Web Development** by Jon Duckett
3. **HTML5 and CSS3 All-in-One For Dummies** by Andy Harris
4. **Database System Concepts** by Abraham Silberschatz, Henry F. Korth, and S. Sudarshan
5. **Fundamentals of Database Systems** by Ramez Elmasri and Shamkant B. Navathe
6. **Database Management Systems** by Raghu Ramakrishnan and Johannes Gehrke
7. **SQL: The Complete Reference** by James R. Groff and Paul N. Weinberg
8. **PHP and MySQL Web Development** by Luke Welling and Laura Thomson
9. **Learning PHP, MySQL & JavaScript** by Robin Nixon
10. **Modern PHP: New Features and Good Practices** by Josh Lockhart
11. **PHP Objects, Patterns, and Practice** by Matt Zandstra
12. **Programming PHP** by Kevin Tatroe, Peter MacIntyre, and Rasmus Lerdorf

References Articles

- R. Batra and A. S. Pall, “Barriers to adoption of hospital management systems: A study of Punjab healthcare industry,” Prabandhan: Indian Journal of Management, vol. 9, no. 11, 2016, doi: 10.17010/pijom/2016/v9i11/105320.
- R. G. Misal, “Advanced Hospital Management System,” Int J Res Appl Sci Eng Technol, vol. 10, no. 6, 2022, doi: 10.22214/ijraset.2022.43686.
- P. K. Yadav and R. Kumar, “Online Hospital Management System,” SSRN Electronic Journal, 2022, doi: 10.2139/ssrn.4104606
- D. S. O. O. E. C. M, “Interlinked Hospital Management System,” International Journal of Science and Research (IJSR), vol. 7, no. 2, 2018
- Web Based E-Hospital Management System,” Iraqi Journal of Computer, Communication, Control and System Engineering, 2018, doi: 10.33103/uot.ijccce.18.1.2.
- Singh, “IoT enabled smart hospital management system for Covid-19 patients,” Turkish Journal of Computer and Mathematics ..., 2021.

*****ALL THE BEST*****