# **CHAPTER THREE**

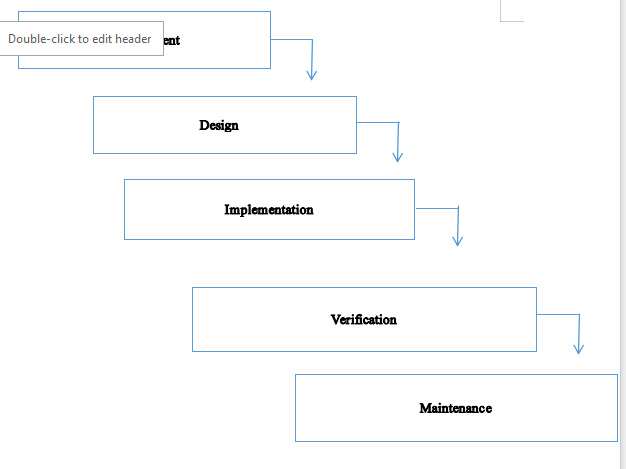
# **METHODOLOGY**

## 3.1 Introduction

In recent years, advancements in technology have revolutionized healthcare administration, leading to the development of integrated digital solutions that streamline the appointment process. This chapter delves into the design, implementation, and functionalities of the application, highlighting its role in optimizing resource allocation, minimizing wait times, and improving overall patient care. By exploring the features and benefits of this innovative platform, this chapter aims to underscore its significance in fostering a patient-centric approach while empowering healthcare providers to deliver timely and effective medical services.

**3.1.1 System Analysis**

The architecture of the whole project is analyzed. System analysis is the process of defining the architecture, components, and data of a system to satisfy specified requirements. Design is a method of studying a system by examining its component parts and their interactions before implementation began. The system was analyzed and designed. In this section, use cases, requirement analysis, and other parts are described in detail.



**Figure 3.1: Waterfall Methodology for** Electronic Doctor-Patient Appointment Management Application

**3.1.2 Requirements Analysis**

The requirements are to be collected before starting project development lifecycle. To design and develop the system, functional as well as non-functional requirements of the system have been studied.

**i. Functional Requirements**

Different functional requirements of the system have been identified and are listed as below:

1. **User Registration and Authentication:**

- Users should be able to create accounts.

- Users should be able to log in securely.

1. **Appointment Scheduling:**

- Users should be able to view available time slots for appointments.

-Users should be able to select a preferred time slot.

- The system should prevent double booking of time slots.

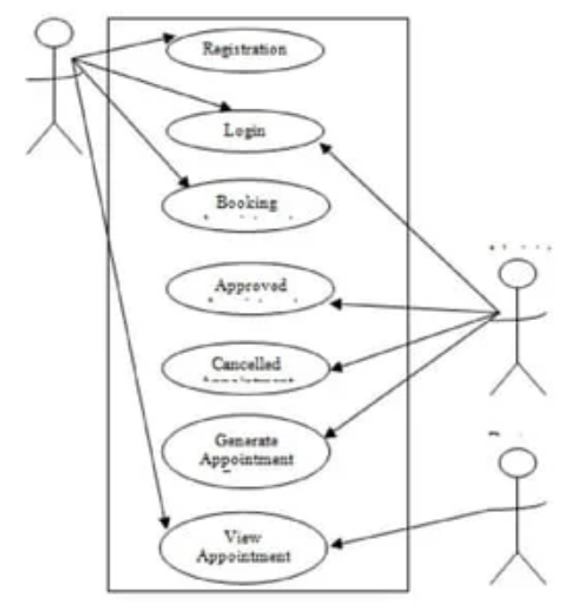
1. **Doctor Profiles:**

- Display detailed profiles of doctors, including their specialties, qualifications, and availability.

1. **Cancellation and Rescheduling:**

- Users should be able to cancel or reschedule appointments within a reasonable time frame

**USECASE DIAGRAM**

****

ii. Non Functional Requirement  
Different non-functional requirement have been studied and identified and are listed as below:

Security  
The system is secure from outside attacks as authorized user and admin are allowed to access the data. Admin representative on duty can log into the system and have access to the Doctor Appointment System but access to have various subsystems is protected by the user login screen that requires a username and password. This system uses at least 8-character passwords for security. Different validation process is used.

Performance  
The performance of the system is fast and accurate as in this system database is normalized so it provide fast operations.

3.1.2 Feasibility Analysis  
i. Technical feasibility study  
The system is technically feasible as the requirement for the development of the system is easily accessible. The necessary hardware and software required for the development and implementation of the system is available. The basic programming language which is suitable for project is available and the libraries required for project is capable of achieving the result that we are aiming for. All the existing resources can be used for the development and maintenance system.

ii. Operational feasibility study  
The system is easy to operate with the basic knowledge of computer and internet and well trained manpower is not necessary. User can also easily access the system as it is user friendly in many aspects with good User Interface (UI). This system include all the requirements used for Doctor Appointment System and this system is completely operational and can be successfully implemented and administration feel easy to use this system as it is userfriendly.

2.   
iii. Economic feasibility study  
The system is economically feasible and cost effective. As all the tools and resources required are either open sources or free. After the completion of the system organization didn’t need to deploy any new hardware and software as the required software and hardware. The existing resource of the system can be used.

iv. Schedule feasibility study  
The system is completed within scheduled time and do not exceed the scheduled time.

Table 3.1: Gantt chart Table for AGRO system

| **Task Name** | **Duration** |
| --- | --- |
| Getting Started | 2 weeks |
| System Design & Architecture | 2 weeks |
| Implementation | 7 weeks |
| Deployment | 4 weeks |
| Documentation | 12 weeks |

Figure 3.3: Gantt Chart of Electronic Doctor-Patient Appointment Management Application

**3.2 Data Modelling (ER Diagram)**  
In the Doctor Appointment System's Entity-Relationship (ER) Diagram, essential entities include "Patient," "Doctor," and "Appointment." Patient and Doctor entities have attributes like ID, name, and contact information. The Appointment entity links patients and doctors, with attributes such as appointment ID, date, and time. Relationships depict associations, such as "Booked By" connecting Patient and Appointment, and "Manages" connecting Doctor and Appointment. Additional entities like "TimeSlot" and "Feedback" may be included. Cardinality notations show the relationships between entities. The ER Diagram serves as a visual representation of how data is structured and related in the Doctor Appointment System, facilitating effective database design.

### 3.2.1 Development Methodology

The development methodology chosen for this project is waterfall model. This methodology was selected due to its suitability for iterative development and its ability to accommodate changing requirements based on user feedback and system testing.

### 3.2.2 Technologies Used

The technologies employed in the development of the application include:

* **PHP**: Used for server-side scripting to handle backend logic and database interactions.
* **JavaScript**: Implemented for client-side scripting to enhance user interactivity and validate form inputs.
* **Bootstrap**: Utilized for front-end framework to ensure responsive design and consistent UI elements.
* **HTML**: Used for structuring the content and defining the elements of the web pages.
* **CSS**: Applied for styling the HTML elements to enhance the visual presentation and user experience.

### 3.2.3 Data Management

Data management in the application involves the use of Database MYSQL, which provides a robust platform for storing and retrieving patient information, appointment schedules, and user credentials securely. The database schema was designed to optimize queries and ensure data integrity throughout the application.

### 3.2.4 User Authentication and Authorization

1. Security measures include implementing user authentication and authorization mechanisms to protect sensitive patient data and ensure that only authorized personnel can access or modify specific information within the system.

* **User Registration and Authentication:**

- Users should be able to create accounts.

- Users should be able to log in securely.

* **Appointment Scheduling:**

- Users should be able to view available time slots for appointments.

-Users should be able to select a preferred time slot.

- The system should prevent double booking of time slots.

* **Doctor Profiles:**

- Display detailed profiles of doctors, including their specialties, qualifications, and availability.

* **Cancellation and Rescheduling:**

- Users should be able to cancel or reschedule appointments within a reasonable time frame

**3.3 System Design**  
To realize the different functional requirement of the system in graphical form, different design diagram of the system has been prepared which are as follows:

**3.3.1 Architectural Design**  
For this system, three tier architecture is used which includes user interface, web server and database. In architectural design, basic structure of the system is shown.

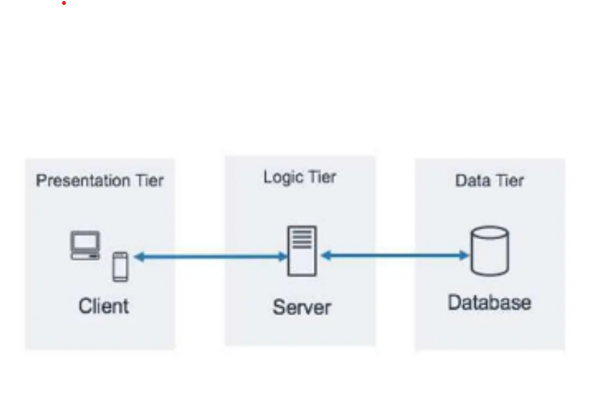


Figure 3.7: Architectural Design of Electronic Doctor-Patient Appointment Management Application

**3.3.2 System Flowchart**  
The system flowchart for the Doctor Appointment System illustrates the sequential steps and interactions within the application. Beginning with the patient's initiation of the appointment request, the flowchart outlines the process of selecting a preferred date and time, checking the availability of the chosen healthcare provider, and confirming the appointment. Simultaneously, the system verifies and updates the appointment schedule. Once confirmed, the system generates automated reminders for both patients and healthcare providers. Additionally, the flowchart includes a feedback loop to capture patient experiences, contributing to continuous system improvement. This visual representation ensures a clear understanding of the end-to-end workflow, fostering efficient and transparent communication in the appointment management process.

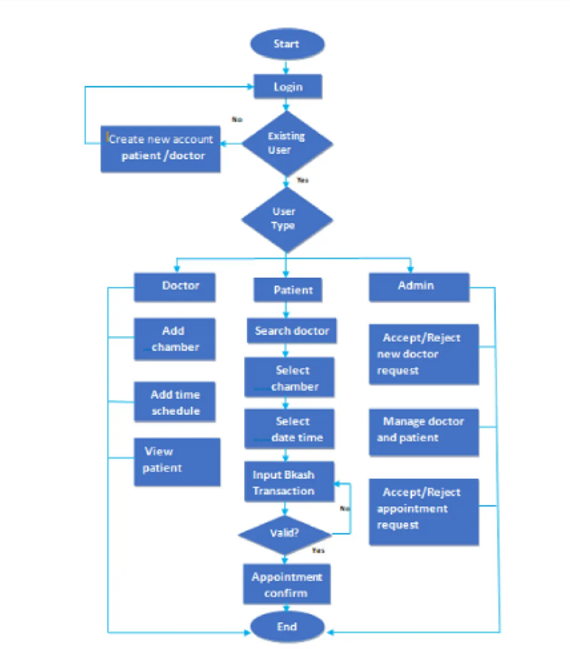


Figure 3.8: Flow chart of Electronic Doctor-Patient Appointment Management Application

### 3.3.3 User Interface Design

#### Front-End Design

The user interface (UI) was designed using Bootstrap for responsive design and cross-browser compatibility. The UI design focuses on intuitive navigation, clear information presentation, and accessibility for users of all technical proficiency levels.

#### Back-End Design

PHP scripts handle server-side logic, including user authentication, appointment scheduling, and database interactions. JavaScript is used to enhance UI interactivity, validate form inputs, and provide real-time feedback to users during the appointment booking process.

**3.4 Summary**

The development process of the Electronic Doctor-Patient Appointment Management Application is detailed. This includes the methodologies employed, the technologies utilized (PHP, JavaScript, Bootstrap, HTML, and CSS), and the design principles adhered to in creating a functional and user-friendly system.