



DEPARTMENT OF COMPUTER SCIENCE

BTECH (HONS) COMPUTER SCIENCE

HIT 200

CLINIC APPOINTMENTS AND MANAGEMENT SYSTEM

THEMBINKOSI SIKOSANA H220250Q

TANATSWA MUGABE H220269E

JOSEPHINE SITHOLE H220760C

SUPERVISED BY MR MUTEVANI

**THIS RESEARCH AND DEVELOPMENT PROPOSAL WAS SUBMITTED TO
HARARE INSTITUTE OF TECHNOLOGY IN PARTIAL FULFILMENT OF THE
BACHELOR OF TECHNOLOGY (HONOURS) DEGREE IN COMPUTER SCIENCE
IN THE YEAR 2024**

COPYRIGHTS

All rights reserved. No part of this project may be reproduced, stored in any retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise from scholarly purpose, without the prior written permission of the author or of Harare Institute of Technology on behalf of the author.

DECLARATION

We do hereby declare that this work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

Student's Signature: Date

(Tanatswa Mugabe)

Student's Signature: Date

(Josephine Sithole)

Student's Signature: Date

(Thembinkosi Sikosana)

Supervisor's Signature: Date

(Mr Mutevani)

DEDICATION

We would like to dedicate this piece of work to our families for the love, care and support they have given to us throughout the project, and our lecturers for the guidance they provided throughout the project.

ACKNOWLEDGEMENTS

Firstly, we would like to thank the Lord God almighty for His mercies and His enabling grace that carried us for seeing us through this project.

Secondly, we would like to thank our supervisors Mr Fadaraliki and Mr Mutevani for their guidance through which we managed to come up with an innovative project.

Lastly our sincere gratitude goes to all lecturers in the School of Information Science and Technology, for without them it would have been impossible for us to acquire the knowledge and be able to undertake this project.

Contents

COPYRIGHTS	i
DECLARATION	ii
DEDICATION	iii
ACKNOWLEDGEMENTS	iv
Contents	v
CHAPTER 1.....	1
1.1 Introduction.....	1
1.2 Background.....	1
1.3 Problem Statement.....	2
1.4 Objectives	2
1.5 Significance of the Project	2
1.6 Scope.....	3
1.7 Expected Results.....	3
1.8 Project timeline	4
1.9 Conclusion	4
CHAPTER 2.....	5
2.1 Introduction.....	5
2.2 Synthesis of Literatures.....	5
2.3 Conclusion	6
CHAPTER 3: ANALYSIS	8
3.1 Information Gathering Tools	8
3.2 Description of the system.....	10
3.2.1 Description of Current System.....	10
3.2.2Problems with current system	10
3.2.3 Description of proposed solution	10
3.3 Functional Analysis of The Proposed System	12
3.3.1 Functional Requirements	12
3.3.2 Non-Functional Requirements	12
3.2.1 context level diagram	13
3.2.5.2 DFD Level 1 of proposed system.....	14

3.2.2 Process Flow chart of the proposed system	15
3.3.3 Use Case.....	16
3.4 Feasibility Study	17
3.4.1 Technical Feasibility.....	17
3.4.2 Economic Feasibility.....	17
3.4.3 Operational feasibility.....	18
3.5 Interface Requirements	18
3.6 Technical Requirements.....	19
3.7 Assumptions.....	19
3.8 Conclusion	19
Chapter 4	20
4.1 Introduction.....	20
4.2 Proposed Solution	20
4.3 Solution Architecture	20
4.4 Constraints	22
4.5 Security Design.....	22
4.6 System Design Models	23
4.6.1 UML Activity Diagram.....	23
4.6.2 UML Class Diagram	24
4.6.3 UML Deployment Diagram	24
4.7 Database Modelling	25
4.7.1 ER Diagram	25
4.7.2 Data Dictionary	26
4.7.3 Relational Schema.....	26
4.8 ALGORITHM DESIGN	26
4.9 INTERFACE DESIGN	29
4.10 CONCLUSION.....	33
CHAPTER 5: IMPLEMENTATION.....	34
5.1 Introduction.....	34
5.2 Code Conventions.....	34
5.3 Coding Strategy	38
5.4 Coding Review.....	39

5.5 Conclusion	40
CHAPTER 6: SYSTEMS TESTING	41
6.1 Introduction.....	41
6.2 Testing Categories	41
6.2.1 White box testing	41
6.2.2 Black box testing.....	42
6.3 TYPES OF TESING AND RESULTS	42
6.3.1 Functional Testing.....	42
6.3.2 Non-Functional Testing	44
6.4 Levels of Testing and Results	45
6.4.1 Unit Testing	45
6.4.2 Integration Testing	46
6.4.3 Systems Testing	46
6.4.4 Acceptance Testing	47
6.5 SYSTEM EVALUATION	47
6.6 CONCLUSION.....	47
CHAPTER 7: CONCLUSION.....	48
7.1 Introduction.....	48
7.2 Results.....	48
7.3 Scope of Future Work.....	49
7.4 APPENDIX.....	51
7.4.1 References.....	51

CHAPTER 1: INTRODUCTION

1.1 Introduction

In a period stamped by mechanical headways and a developing request for productive healthcare administrations, the Appointments Booking and Management System develops as a transformative arrangement. This framework leverages the control of computerized development to address the challenges confronted by common professionals, especially in creating nations like our own.

The Appointments Booking and Management System could be a comprehensive computer program application outlined to streamline healthcare organization, upgrade persistent care coordination, and cultivate superior engagement between patients and doctors. Through its user-friendly interface and highlights, this system points to improve the way healthcare administrations are overly managed.

1.2 Background

The urgent need for improvements in healthcare services in Zimbabwe is a driving force behind the development of a system that makes it easier for general practitioners to schedule appointments. Similar to many developing countries, Zimbabwe has several healthcare-related issues, such as uneven healthcare delivery, ineffective appointment scheduling systems, and restricted access to healthcare. With the development of a digital platform that facilitates effective appointment scheduling, we hope to tackle these problems head-on and greatly enhance the general healthcare experience for patients and providers. The goal of achieving fair access to healthcare is the driving force behind this system. In Zimbabwe, the dispersion of healthcare facilities geographically frequently ends in inadequate access to healthcare for rural and underprivileged areas.

Patients all around the country may now quickly schedule appointments with medical practitioners, no matter where they live, thanks to this digital solution. The system seeks to ensure that every Zimbabwean can immediately obtain medical assistance, thereby improving their overall health and well-being, by promoting accessibility and lowering geographical barriers. Moreover, effective appointment scheduling enables medical professionals to maximize their time and resources, guaranteeing that they can treat a larger patient population

promptly. This enhances the rapport between patients and providers and helps make healthcare services more efficiently delivered across Zimbabwe.

1.3 Problem Statement

- The current system takes up a lot of time as it is all done manually and leads to long waiting times for patients which is very inconvenient.
- The lack of a technologically-driven solution hinders efficient patient involvement and resource allocation efficiency for healthcare practitioners.

1.4 Objectives

The main objectives of the research are as follows:

- To provide 24-hour access to an appointment booking portal
- To allow management of all scheduled appointments by clinic administration
- To allow online payment for appointments via a payment gateway

1.5 Significance of the Project

The justification for building this appointment scheduling system for clinics with general practitioners in Zimbabwe is rooted in several critical reasons:

- **Easy Healthcare Access:** The system attempts to close the gap in healthcare access by making it simple for people to book appointments with general practitioners anywhere in Zimbabwe, including distant and underserved locations.
- **Improves Efficiency:** Setting up the system is crucial for improving clinic operations. It decreases scheduling conflicts, makes the most use of available resources, and eases the administrative strain on healthcare professionals. In addition to helping healthcare workers, this efficiency enhances patients' overall experiences.
- **Improves Patient Engagement:** The system gives patients more control by giving them access to an easy-to-use platform for making, managing, and following up on appointments. Patients have more influence over their healthcare, which can improve health outcomes and increase self-care participation.

- **Improved Care Quality:** The system helps to raise the standard of healthcare services offered in Zimbabwe. It improves the quality of care by lowering waiting times, minimizing mistakes, and promoting patient-provider communication.
- **Patient Satisfaction and Trust:** A well-designed, effective system for scheduling appointments promotes patient satisfaction and faith in the healthcare system. When patients can easily schedule appointments and receive care in a way that respects their time and convenience, they are more likely to seek timely treatment.

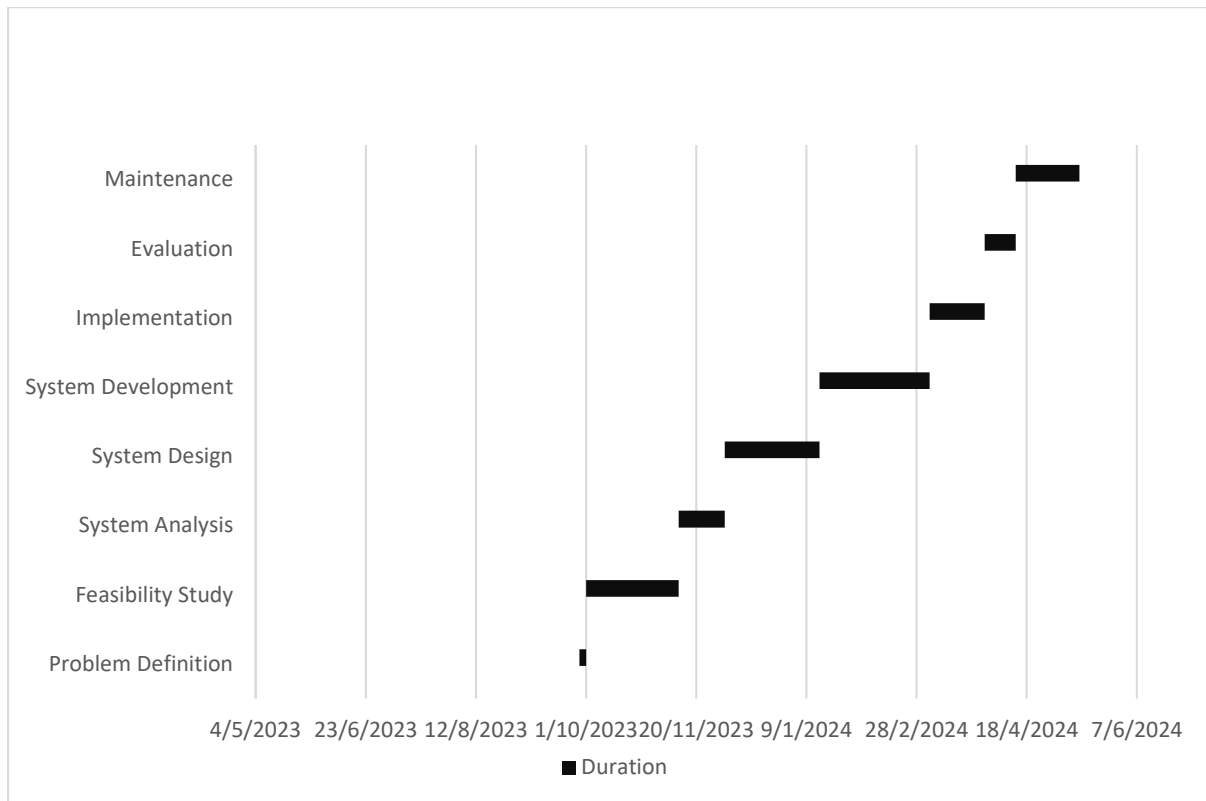
1.6 Scope

Our scope is Community Health Clinic, which are public clinic with general practitioners located in Zimbabwe

1.7 Expected Results

- Develop an algorithm that optimizes appointment scheduling in healthcare clinics, reducing patient wait times and enhancing resource allocation.
- Develop a user-friendly app that empowers patients to manage their health, access medical information, and improving overall healthcare experiences.
- Develop a system that allows healthcare providers to electronically manage appointments
- Produce a research paper presenting novel findings and insights into healthcare management practices, contributing to the academic knowledge base in the field.

1.8 Project timeline



1.9 Conclusion

Therefore, in this chapter we have highlighted the issues of the current system and proposed a solution to it. From the solution the process will become electronic compared to the paper based and the work done by clinic administrators will be greatly reduced.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

In the realm of healthcare management, the efficacy of appointment and scheduling systems for general surgeries stands as a critical component in ensuring optimal patient care and resource utilization. A comprehensive understanding of the existing literature surrounding this subject is paramount for informed decision-making and strategic enhancements within healthcare facilities. This synthesis delves into the wealth of papers and documentation available, shedding light on the intricacies, challenges, and advancements in the contemporary landscape of appointment and scheduling systems for general surgeries.

2.2 Synthesis of Literatures

Related Works

"Design of Intelligent and Secure Hospital Appointment Scheduling System," the authors propose a system aimed at enhancing healthcare delivery by automating appointment scheduling while ensuring security. By negotiating appointment schedules with medical teams automatically, the system optimizes resource utilization and satisfies patients' demands for timely access to medical services. However, potential challenges in implementation, security, user acceptance, scalability, and interoperability are identified, underscoring the need for robust solutions.

"Research on an Electronic Medical Record System Based on the Internet" introduces an electronic medical record system to improve healthcare management. While the system offers potential benefits such as improved efficiency and quality of care, gaps such as the lack of empirical evidence and inadequate addressing of security concerns are noted. Additionally, the paper emphasizes the importance of considering user needs and preferences for successful adoption.

"Digital Transformation in Healthcare" reveals various changes and challenges induced by digital technologies. While digital transformation offers numerous benefits, including convenience, quality improvement, and accessibility expansion, significant gaps such as the lack of empirical studies and interdisciplinary research are identified. These findings underscore the need for evidence-based approaches to advance digital transformation in healthcare effectively.

"Digital Transformation in Healthcare on Demand" emphasizes the inevitability of digital transformation in healthcare and its potential to enhance quality of care while reducing costs. However, challenges such as interoperability issues and security concerns highlight the importance of holistic solutions. The paper underscores the need for user-friendly digital solutions that address diverse stakeholder needs.

"The Internet of Things for Smart Healthcare": explores IoT technology's potential to revolutionize healthcare delivery. While IoT applications offer numerous benefits, challenges such as security and data quality issues must be addressed. The paper emphasizes the importance of robust cybersecurity measures and interoperable systems to realize the full potential of IoT in healthcare.

"Healthcare 5.0: A Paradigm Shift in Digital Healthcare System": introduces the concept of healthcare 5.0, leveraging AI, IoT, and 5G communication for personalized healthcare services. While the vision of healthcare 5.0 holds promise, challenges such as security, interoperability, and user acceptance need to be addressed. The paper advocates for collaborative efforts and evidence-based approaches to overcome these challenges.

"An Automated Model for Booking Appointment in Health Care Sector": addresses inefficiencies in appointment booking systems by proposing an automated model. The model offers features such as instant booking, unique patient identification, and online payment. However, gaps such as errors, delays, and lack of instant booking highlight the need for robust solutions that prioritize patient satisfaction and efficiency.

"The Privacy Preservation of Patients' Health Records": discusses the advantages and risks of digitizing Electronic Health Records (EHRs). While digital systems offer benefits such as improved record-keeping, privacy concerns related to data security and misuse must be addressed. The paper proposes privacy preservation models and machine learning algorithms to safeguard patient data effectively.

2.3 Conclusion

In synthesizing these papers, it becomes evident that while each contributes unique perspectives and insights, they collectively highlight the imperative need for a comprehensive digital patient record system accompanied by automated appointment management in healthcare. Across the papers, there is a recurring theme of leveraging digital technologies to improve efficiency, enhance patient care, and address longstanding challenges within the healthcare sector.

However, notable contrasts emerge among the papers, particularly in their focus areas and proposed solutions. For instance, while some papers emphasize the technical aspects of digital transformation, others delve into the organizational and human factors influencing its implementation. This contrast underscores the multifaceted nature of healthcare digitization, necessitating a holistic approach that considers technological, social, and regulatory dimensions.

Despite these differences, common patterns emerge, such as the emphasis on interoperability, security, and user acceptance. Many papers highlight the importance of seamless data exchange and integration across disparate systems to facilitate continuity of care and enable informed decision-making. Additionally, the significance of robust security measures to safeguard patient data from unauthorized access and misuse is consistently underscored across the literature.

Furthermore, the papers collectively recognize the pivotal role of user involvement and engagement in the successful implementation of digital healthcare solutions. Whether it be patients, healthcare providers, or other stakeholders, active participation and feedback are deemed essential for driving adoption and ensuring the usability and effectiveness of digital tools and systems.

In conclusion, while each paper offers unique insights and perspectives on the digitization of healthcare, they converge on the overarching need for a comprehensive and integrated approach to digital transformation. By addressing challenges such as interoperability, security, and user acceptance through collaborative efforts and evidence-based strategies, healthcare organizations can realize the full potential of digital technologies to enhance patient outcomes, streamline operations, and improve overall healthcare delivery. The synthesis of these papers serves as a call to action for stakeholders across the healthcare ecosystem to prioritize investments in digital infrastructure and innovation, ultimately advancing towards a more connected, efficient, and patient-centric healthcare system.

CHAPTER 3: ANALYSIS

3.1 Information Gathering Tools

Appointment Scheduling is a process that has been there even before technology. We need to have an understanding of the traditional way of scheduling appointments in a clinic before coming up with a new one in order to prevent any gaps by integrating the two systems as efficiently as possible. Some information-gathering tools were used to get a clear understanding of the scheduling process. These including questionnaires, interviews and case studies. These tools were used to obtain information on the following trivial points:

- Patient registration
- Scheduling and Scheduling conflicts
- Patient record storage
- Data privacy
- Assignment of Doctors

Strategic use of the above-mentioned instruments was made, with an emphasis on the goals, queries, and crucial details to record. Interviews were conducted with nurses and doctors from various clinics that oversee the appointment scheduling system, and prospective users of the system, such as patients and clinic receptionists were given the questionnaires.

Interview Guide for Nurses and Doctors

1. Please tell us a little about yourself and your background.
2. What is your role in the current scheduling system in the clinic?
3. Does the clinic use a manual scheduling system?
4. (If answer in question 3 is yes) What is your take on the implementation of a payment and automated scheduling system
5. (If answer to question 3 is no) Give a subjective analysis of the current system based on this experience. (Emphasising on any changes you would recommend)
6. For how long have you been employed at the clinic?
7. What are the strengths and weaknesses you have observed in the current scheduling system?
8. What do you think are the most important user needs for an appointment scheduling system and how can a new system address these needs?
9. Are there any additional comments or suggestions you would like to make about the scheduling system or its development process?

Thank you for your time and valuable insights.

THIS QUESTIONNAIRE WAS DESIGNED STRICTLY FOR EDUCATIONAL RESEARCH PURPOSES. THE INFORMATION PROVIDED SHALL NOT BE DISCLOSED TO EXTERNAL PARTIES. FEEL FREE TO FILL IN AS REQUIRED

1. State the clinic you are representing.

2. Does your clinic allow scheduling of appointments?

☐
YES☐
NO

3. If yes, please describe the scheduling process.

4. What are the strengths in the current scheduling process?

5. What are the weaknesses in it?

6. Given that a computerised system is to be implemented (in the case that it was not previously used) or corrective measures are being put to an existing one, what are the functionalities you would consider essential as a user?

7. Are there any additional comments or suggestions you would like to make about the scheduling system or its development process?

Thank you for your time and valuable insights

3.2 Description of the system

3.2.1 Description of Current System

The current system is a manual paper-based system that greatly relies on the filling of forms by patients and writing in diaries and on papers by doctors and receptionists. The only digital aspect at these clinics is the ability to use biometrics to verify the validity of a patient's medical aid service card.

When a patient walks in, they are asked to fill in a form that requires them to fill in their details and personal information. After this they tell the receptionist or nurse the day, he/she would like to see the doctor, the receptionist then manually checks in their diary to see what times on the specified day haven't been given to other patients. If a day is already fully booked, the patient is given a time slot on the nearest date from their date of preference.

3.2.2 Problems with current system

However, it is not without its challenges. The manual process of patients filling out forms and receptionists manually checking diaries for available time slots is time-consuming and inefficient. Receptionists may spend significant time flipping through pages and cross-referencing schedules, leading to delays and potential errors in booking appointments. Manual data entry and record-keeping increase the likelihood of data entry errors and inconsistencies.

Another major problem is limited access to information. Paper-based forms and diaries restrict access to patient information and appointment schedules to physical locations within the clinic. Receptionists and doctors may not have real-time access to patient records and appointment availability, making it difficult to respond promptly to patient inquiries or changes in scheduling.

Other problems include increased risk of errors because paper-based systems are prone to errors such as illegible handwriting, misplaced documents, and double-booked appointments. Poor patient experience due to long wait times for appointment booking and uncertainty about appointment availability and Difficulty in Tracking and Reporting because paper-based records make it challenging to track clinic activity, analyse appointment trends, and generate reports for performance evaluation or regulatory compliance.

The last problem I'd like to mention is the dependency on physical resources. The manual system relies on physical resources such as paper forms, diaries, and filing cabinets, leading to increased storage space requirements and potential loss or damage to documents.

3.2.3 Description of proposed solution

The proposed solution is to develop a digital appointment scheduling system to replace the existing manual paper-based process used in the clinic. This digital system aims to streamline and automate the appointment booking process, enhance patient experience, and improve overall clinic efficiency.

Through the system, patients will be able to prompt the system to schedule an appointment for him/her. The system will also allow the patient to make a payment for his/her appointment

through the app. Doctors will have access to their patients records and they will be able to provide feedback to their patients through the application and the patients will also be able to access their feedback through using a search of their patented and appointment.

The receptionist is in-charge of managing doctors' personal details, patient details and appointment details. Only the receptionist can reschedule or delete appointments. The system will also be able to distinguish between doctors, patients and the receptionist to give each user their unique and appropriate features and functionality

. Our system in conclusion is going to have the following algorithms:

- ✓ An algorithm that allows scheduling appointments without double-booking
- ✓ An algorithm to generate unique IDs for Doctors.
- ✓ The system will allow online payment of appointments.
- ✓ An algorithm that gives patients access to ONLY their feedback from the doctor
- ✓ A system that sends scheduling confirmation emails to patients

Constraints of proposed system

1. Budgetary Constraints:
 - Limited financial resources may restrict the implementation of advanced features or integration with existing systems.
 - Budget constraints may impact the choice of technology, development resources, and ongoing maintenance costs.
2. Technical Constraints:
 - Compatibility issues with existing hardware, software, or network infrastructure may pose challenges during system integration.
 - Technical limitations of the chosen development platform or programming language may restrict the implementation of certain features or functionalities.
3. Data Security and Privacy Concerns
 - Compliance with data protection regulations such as HIPAA (Health Insurance Portability and Accountability Act) imposes strict requirements on the handling and storage of patient information.
 - Ensuring data security and privacy while storing and transmitting sensitive patient data is crucial but may require additional resources and expertise.
4. User Adoption and Training
 - Resistance to change from clinic staff accustomed to manual processes may hinder the adoption of the new system.
 - Adequate training and support for clinic staff are essential to ensure smooth transition and effective use of the digital appointment scheduling system.
5. Integration with Existing Systems:
 - Integration with existing clinic management systems, electronic health records (EHR), or billing systems may pose compatibility challenges.

- Lack of standardized interfaces or APIs (Application Programming Interfaces) for data exchange between systems could hinder seamless integration.
6. Scalability and Performance:
 - The system must be designed to handle increasing patient volumes and growing clinic operations over time.
 - Scalability issues may arise if the system architecture or database design is not sufficiently robust to accommodate future growth.
 7. Reliability and Availability:
 - Downtime or system outages could disrupt clinic operations and impact patient care.
 - Implementing redundancy, backup, and disaster recovery measures is essential to ensure system reliability and availability.

3.3 Functional Analysis of The Proposed System

3.3.1 Functional Requirements

1. User registration and login
 - The system should allow new users to register themselves and existing users to log in using their credentials.
2. Appointment Booking:
 - Provide an interface for receptionists or nurses to book appointments for patients.
 - Allow patients to book their own appointments via the website.
 - Allow staff to select a date and view available time slots for that day.
 - Display real-time updates on appointment availability and suggest alternative dates or times if the selected slot is fully booked.
3. Appointment Management:
 - Enable staff to schedule, reschedule, or cancel appointments as needed.
 - Ensure that double-bookings are prevented and appointments are properly synchronized with doctor schedules.
4. User Roles and Permissions:
 - Define different user roles (e.g., receptionists, nurses, doctors) with varying levels of access and permissions.
 - Ensure that users can only access functionality and data relevant to their role.
5. Payment of appointments:
 - Allow patients to pay for the appointments they would've scheduled for and send an email confirming appointment time after payment.

3.3.2 Non-Functional Requirements

- 1. Security:** The system must be secure to prevent any unauthorized access, tampering, or manipulation of the scheduled appointments
- 2. Reliability:** The system must be reliable and accurate to ensure that each appointment is scheduled correctly

- 3. Availability:** The system must be available to all users at any given time
- 4. Usability:** The system must be user-friendly and easy to use, even for patients who may not be familiar with technology.
- 5. Scalability:** The system must be scalable to accommodate any increase in the number of patients to the clinic.
- 6. Performance:** The system must perform efficiently and effectively, with minimal delays or errors. (NB: System performance is affected by the specifications of one's machine)
- 8. Maintainability:** The system must be easy to maintain and update to ensure that it remains secure and reliable over time.
- 9. Compatibility:** The system must be compatible with existing infrastructure and systems used clinics around the country
- 10. Privacy:** The system must ensure the privacy of each patient's personal information.

3.2.1 context level diagram

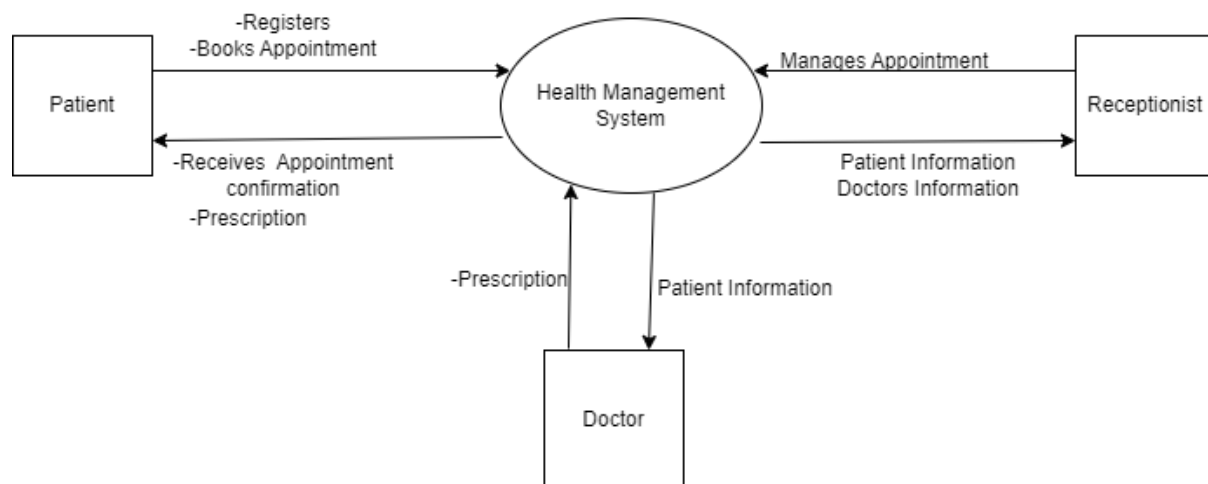


Fig 3.1: context level diagram of proposed system

3.2.5.2 DFD Level 1 of proposed system

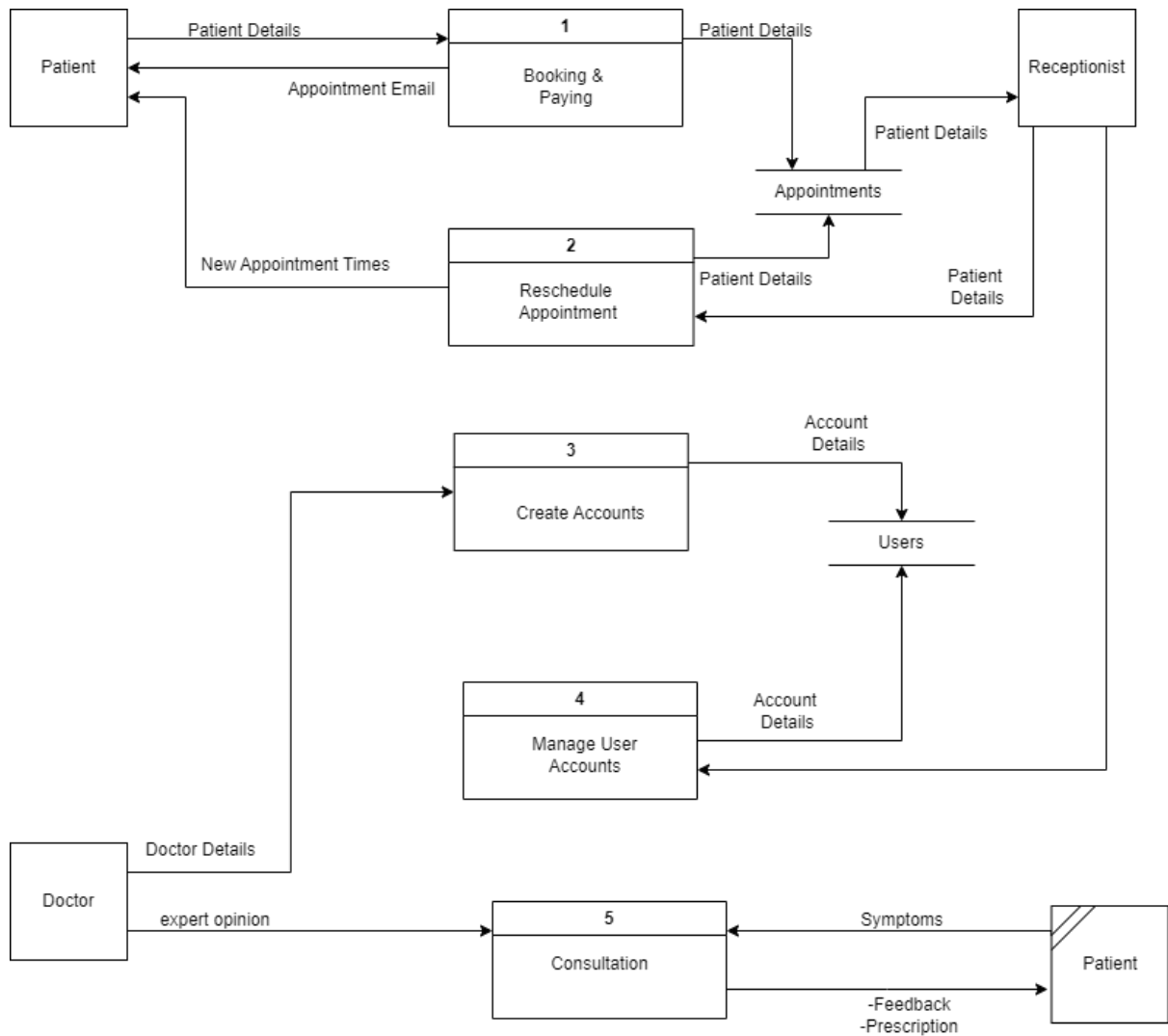


Fig 3.2: level 1 diagram of proposed system

3.2.2 Process Flow chart of the proposed system

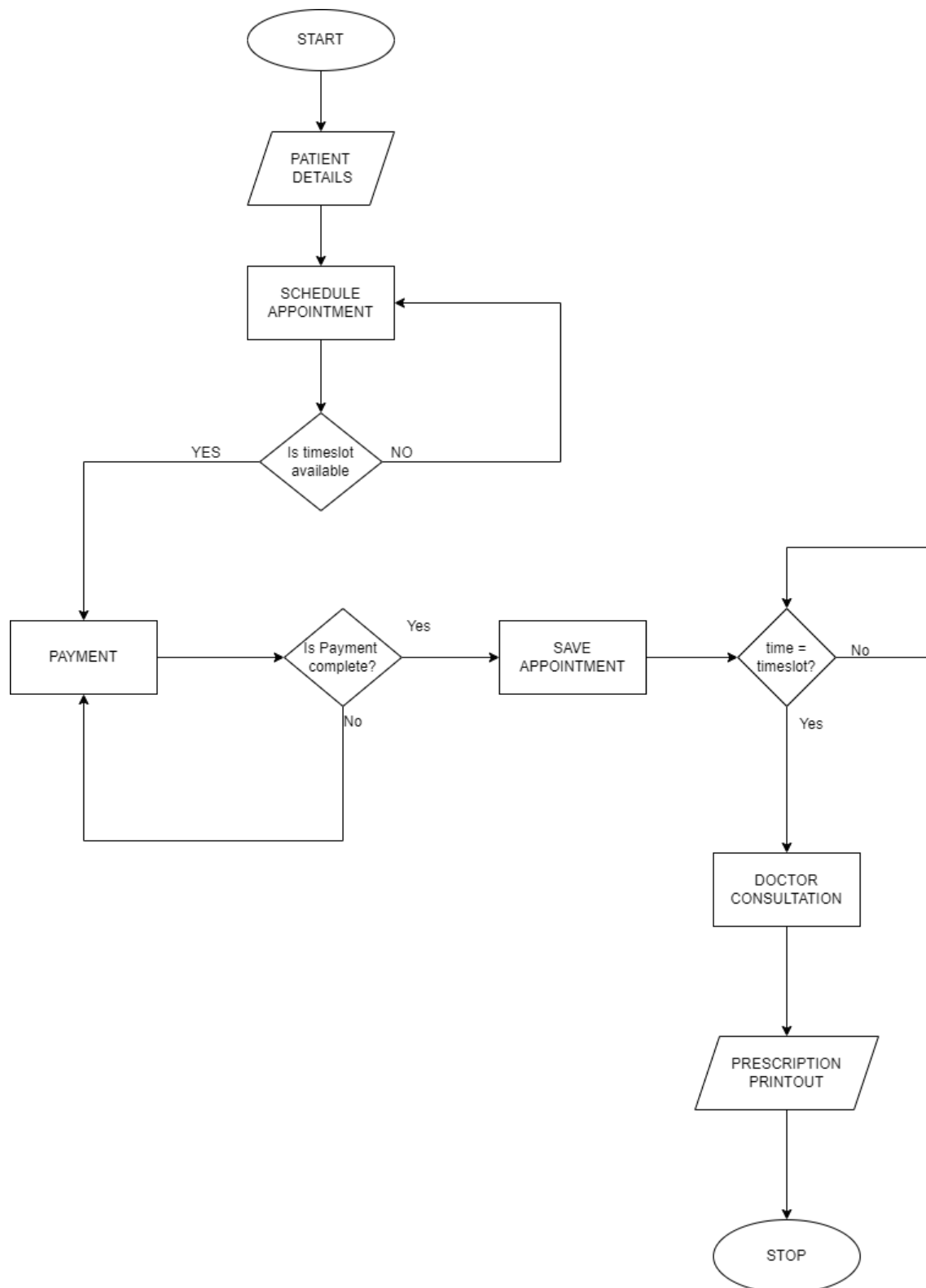


Fig 3.1: process flowchart of proposed system

3.3.3 Use Case

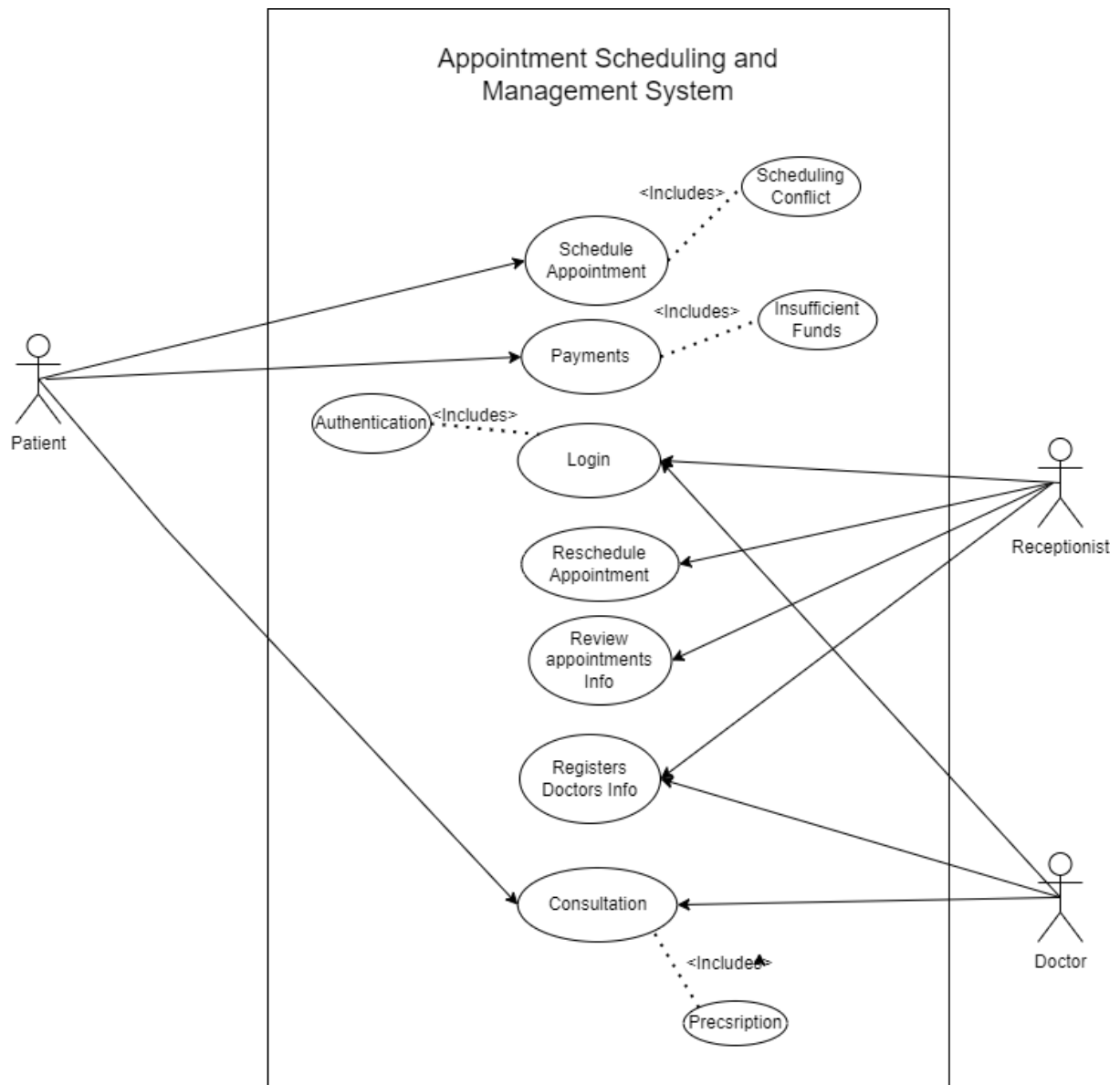


Fig 3.1: use case of proposed system

3.4 Feasibility Study

3.4.1 Technical Feasibility

The technical feasibility of our digital appointment scheduling system is crucial to ensure that the project can be successfully developed and implemented. This section outlines the technical skills and resources required for the project:

- **Database Management Skills:** Knowledge of database management systems is essential for handling data storage, retrieval, and manipulation. We will leverage Microsoft SQL, a robust and scalable relational database system, to store and manage patient records, appointment schedules, and other relevant data.
- **Backend Development Proficiency:** Our system's backend will be built using ASP.NET, a server-side C# runtime environment. This will enable us to implement business logic, handle data processing, and facilitate communication between the frontend interface and the database. Our system also includes a website that allows users to book appointments, so we will be using PHP for the server-side functions of booking appointments and payments
- **Frontend Development:** Our systems user interface will be designed using C#'s .NET framework to come up with easy-to-use user environments for the clinic staff. The websites interface is designed using HTML, JavaScript and CSS to provide patients with an easy-to-use interface

3.4.2 Economic Feasibility

Economic feasibility entails evaluating whether the benefits of the proposed system outweigh the costs associated with its development, implementation, and maintenance. The following factors are considered in the economic feasibility analysis:

- **Cost-Benefit Analysis:** A comprehensive cost-benefit analysis will be conducted to assess the financial implications of developing, deploying, and maintaining the digital appointment scheduling system. This analysis will quantify both tangible benefits, such as cost savings from improved efficiency and reduced administrative overhead, and intangible benefits, such as enhanced patient satisfaction and staff productivity. In this case, the benefits far outweigh the costs of developing the system.
- **Return on Investment (ROI):** The projected ROI of the system will be calculated based on the anticipated benefits and costs over a specified timeframe. This will help determine the feasibility of the investment and provide stakeholders with insight into the potential financial gains.
- **Risk Assessment:** Potential risks and uncertainties that may impact the economic feasibility of the project, such as budget overruns, technology obsolescence, and market changes, will be identified and mitigated through proactive risk management strategies.

3.4.3 Operational feasibility

Operational feasibility assesses the practicality and effectiveness of implementing the proposed system within the organization's existing operational environment. This section examines the operational considerations associated with the development and deployment of the digital appointment scheduling system:

- **Workflow Integration:** Evaluate the compatibility of the system with existing clinic workflows and procedures. Ensure that the system seamlessly integrates with established processes for patient registration, appointment scheduling, and staff coordination to minimize disruption and maximize efficiency.
- **User Acceptance and Training:** Assess the readiness of clinic staff to adopt and use the system effectively. Provide comprehensive training and support to ensure that users understand how to navigate the system, perform key tasks, and troubleshoot common issues. Solicit feedback from users to identify areas for improvement and address usability concerns.
- **Resource Availability:** Determine the availability of resources, including personnel, equipment, and infrastructure, required to support system implementation and operation. Allocate sufficient resources for tasks such as system development, testing, training, and ongoing maintenance to ensure successful deployment and sustained usability.
- **Scalability and Flexibility:** Assess the system's scalability and flexibility to accommodate future growth and evolving clinic requirements. We can ensure that the system architecture and design allow for easy expansion, customization, and integration with additional features or modules as needed to meet changing operational needs.

By conducting a thorough analysis of the technical requirements, economic considerations and operational considerations, we can determine the feasibility of developing and implementing the digital appointment scheduling system, ensuring that it delivers value to stakeholders while remaining financially viable in the long run.

3.5 Interface Requirements

Based on our research, some of the interface requirements for an appointment scheduling system include:

- The user interface should be intuitive and easy to navigate for both clinic staff and patients, with clearly labelled options and minimal complexity.
- Patients should be able to easily book appointments and make payments, while clinic staff should have access to a centralized dashboard for managing appointments and doctor ac
- The interface should provide real-time updates on appointment availability and accounts for quick rescheduling, or cancellation of appointments.
- Security features such as user authentication, and access levels should be integrated into the interface to protect patient privacy and comply with regulatory requirements.

3.6 Technical Requirements

Software

- Operating system: the system can be launched from a Windows Operating System (Windows 10 or better)
- Database: Microsoft SQL Server database will be used to store appointment data and doctor account information
- Frontend and backend frameworks: C#, HTML and CSS will be used to design the frontend of the system. ASP.NET framework, PHP and JavaScript will be use to integrate the backend of the system, as well as adding functionality

Hardware

- A device with processor speed of 2.5 GHz or higher may be required.
- RAM of at least 4GB
- 512GB external HDD to store data for backup

3.7 Assumptions

- Booking an appointment typically involves checking the availability of time slots for a specific date. If the appointment slots are stored in a data structure like a hash table or a balanced tree, the time complexity for checking availability could be constant ($O(1)$). However, if data is stored in JSON file and linear search or traversal of a list is required, the time complexity could be linear ($O(n)$). If the list needs to be sorted before being traversed, time complexity would be ($O(n \log n)$)
- The system complies with relevant healthcare regulations and standards, such as HIPAA, GDPR (General Data Protection Regulation), and local privacy laws.
- The system is available and accessible to users 24 hours a day, with minimal downtime for maintenance or updates.

3.8 Conclusion

The proposed digital appointment scheduling system is not only feasible but also highly relevant, addressing the limitations of the current manual system while offering enhanced functionality. With all requirements met, the system represents a significant step forward in modernizing clinic operations and improving efficiency. Developers can confidently proceed to the next stage of development.

Chapter 4: DESIGN

4.1 Introduction

Following a comprehensive analysis of the functional requirements of the proposed appointment system, the design phase delineates the physical blueprint of the system. Designing entails articulating the solution by delineating how the system fulfils the requirements established and articulated during the analysis phase. This chapter serves as the architectural plan detailing how the proposed system will be constructed and function. The primary output of this phase is the technical design, which holds paramount importance for the technical development team. While users may contribute to the high-level appearance of the system, the technical team is responsible for crafting the low-level specifics. The focal points of the design phase encompass Physical Design, System Architecture, Database Design, Interface Design, Program Design, and Test Design.

4.2 Proposed Solution

The proposed solution is an intuitive and user-friendly Appointment Management System accessible through web interfaces for patient, and a C# application post a successful registration process for clinic staff and doctors. To ensure universal accessibility and ease of navigation, the system will be available in English language. The system aims to address two core functionalities:

- Scheduling appointments seamlessly online
- Managing appointment slots efficiently

The Appointment Management System will facilitate appointment scheduling for users and streamline the management of appointment slots, ensuring optimal utilization of available time slots. Additionally, the system will provide features for administrators to oversee appointment schedules, allocate resources efficiently, and generate prescriptions for patients.

4.3 Solution Architecture

The appointment management system is a comprehensive software solution designed to streamline the scheduling process, enhance efficiency, and improve user experience. It functions as a centralized platform that accepts appointment requests and facilitates the management of appointment slots, ensuring seamless coordination between users and service providers. The system architecture comprises the following key components:

1. **User Interface:** The system features a user-friendly interface accessible via web browsers or mobile applications. Users, including both clients and service providers, interact with the system through intuitive interfaces to schedule, view, and manage appointments.
2. **Appointment Scheduler:** The core functionality of the system revolves around the appointment scheduler module, which processes appointment requests, checks availability, and allocates time slots based on predefined criteria and business rules. This module employs algorithms to optimize scheduling efficiency and minimize conflicts.
3. **Database Management System (DBMS):** A robust DBMS serves as the backend infrastructure for storing and managing appointment-related data, including user

profiles, appointment schedules, availability calendars, and historical records. The database ensures data integrity, scalability, and accessibility for authorized users.

4. Notification Engine: The system includes a notification engine responsible for sending appointment confirmations to both users and service providers.
5. Administrative Dashboard: An administrative dashboard provides authorized personnel with insights into system performance, appointment metrics, user activity, and other relevant analytics. Administrators can monitor appointment schedules, reschedule them and even delete them where necessary.
6. Security Measures: To safeguard sensitive user information and maintain data privacy, the system implements robust security measures, including encryption protocols on the database, access controls, and authentication mechanisms. Compliance with data protection regulations ensures confidentiality and integrity of user data.

Overall, the appointment management system offers a comprehensive solution to streamline appointment scheduling processes, optimize resource utilization, and enhance user satisfaction for both clients and service providers.

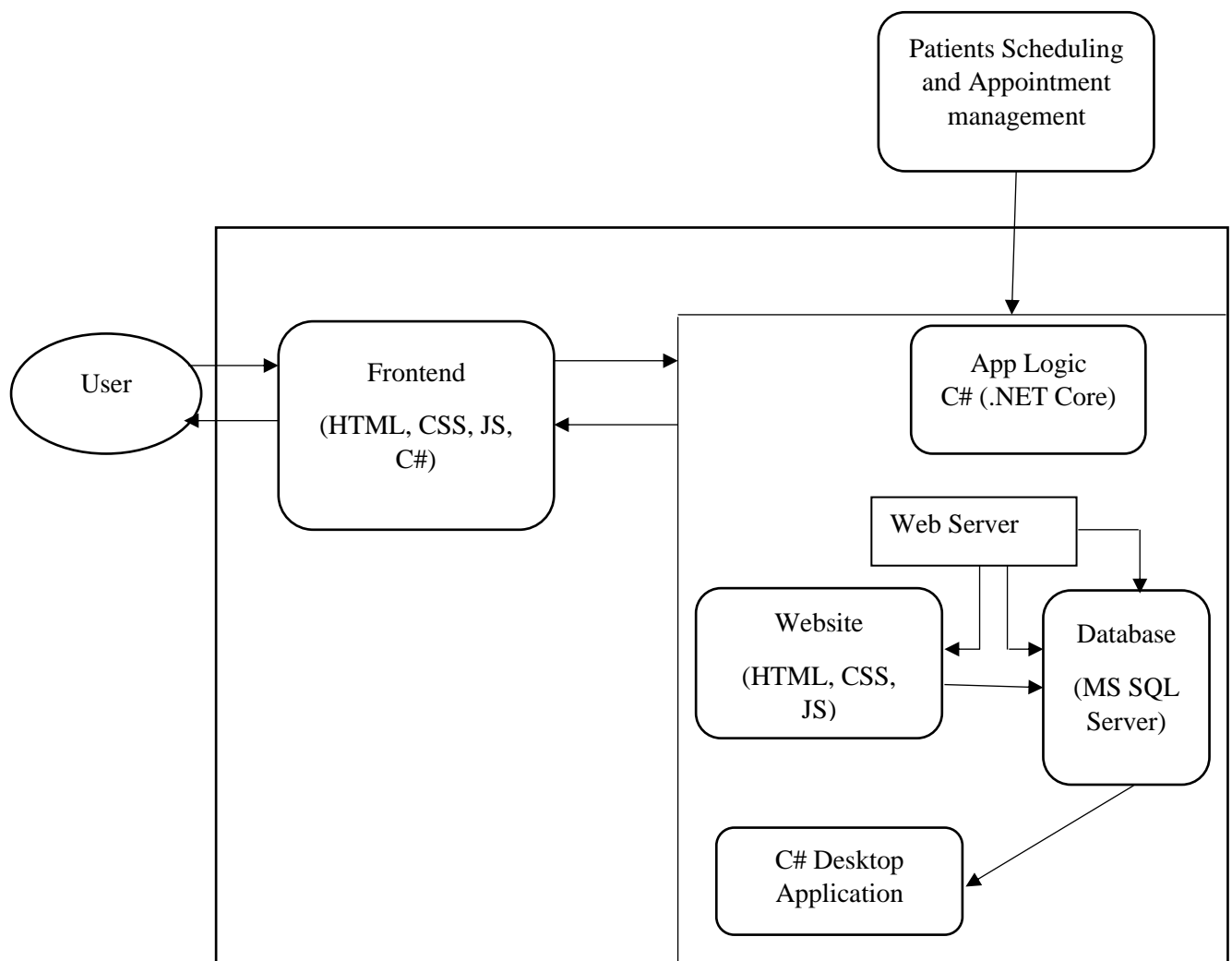


Fig 4.1: Solution Architecture

4.4 Constraints

The proposed appointments system operates within certain constraints to ensure its functionality and effectiveness:

- **Internet Connectivity:** The system requires a stable internet connection for both users and administrators to access its features. Without internet access, users will be unable to schedule appointments, and administrators won't be able to manage appointments or view system data.
- **Device Compatibility:** Users need devices such as computers, smartphones, or tablets with internet browsing capabilities to access the system. Additionally, administrators require devices compatible with the system's administrative interface for managing appointments and system settings.

4.5 Security Design

Security measures are implemented to protect the appointments system from unauthorized access, data breaches, and malicious activities:

- **User Authentication:** Users are required to authenticate themselves before accessing the system, typically through a username and password combination. This ensures that only authorized individuals can schedule or modify appointments.
- **Data Encryption:** Sensitive data, such as user information and appointment details, are encrypted on the database to prevent unauthorized access or tampering during transmission and storage.
- **Access Control:** Role-based access control mechanisms are employed to restrict user privileges based on their roles within the system. Administrators have elevated permissions to manage appointments and system settings, while regular users have limited access.

These security design principles aim to ensure the confidentiality, integrity, and availability of data within the appointments system, safeguarding it against potential security risks and vulnerabilities.

4.6 System Design Models

4.6.1 UML Activity Diagram

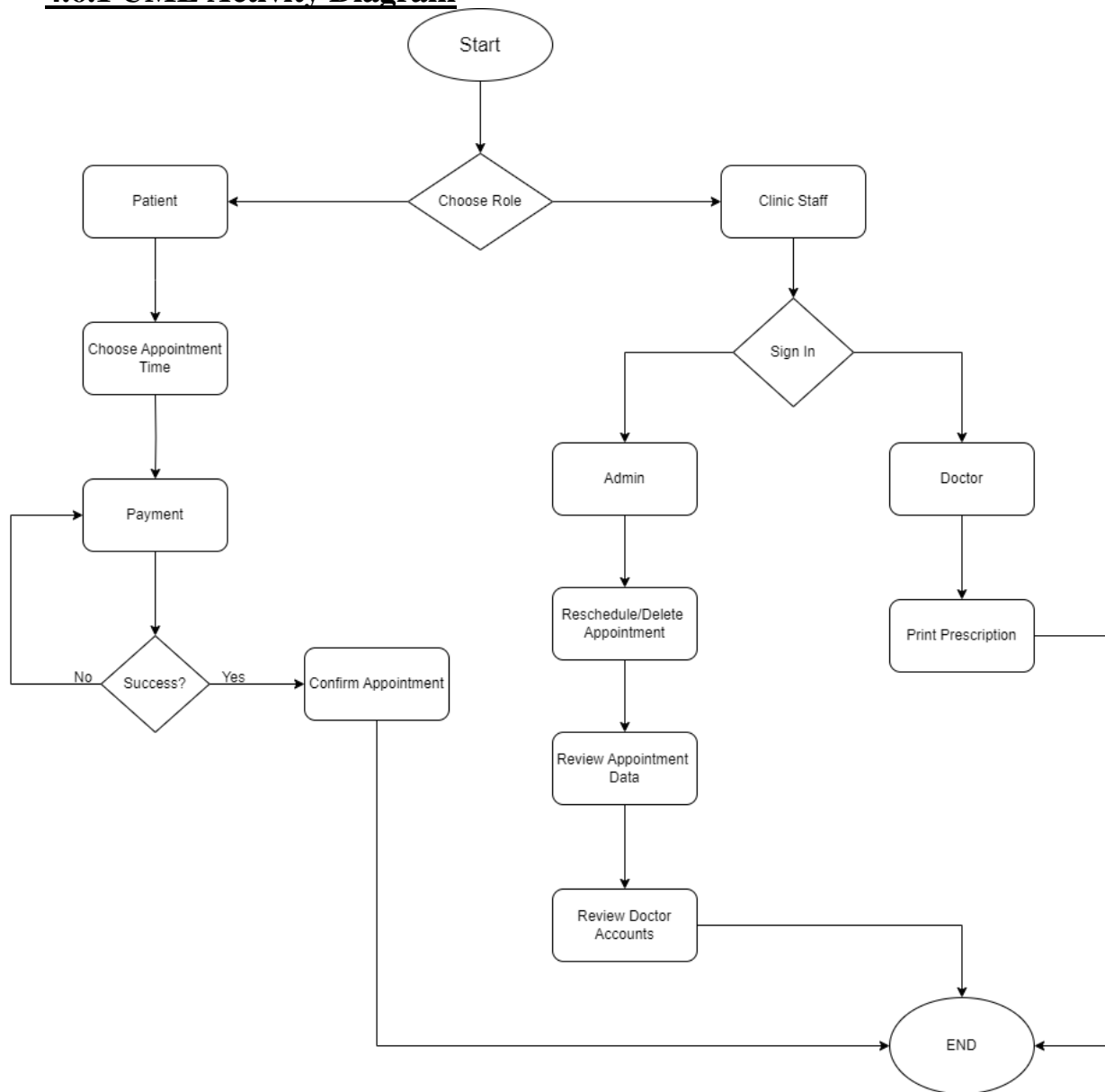


Fig 4.2: UML Activity Diagram

4.6.2 UML Class Diagram

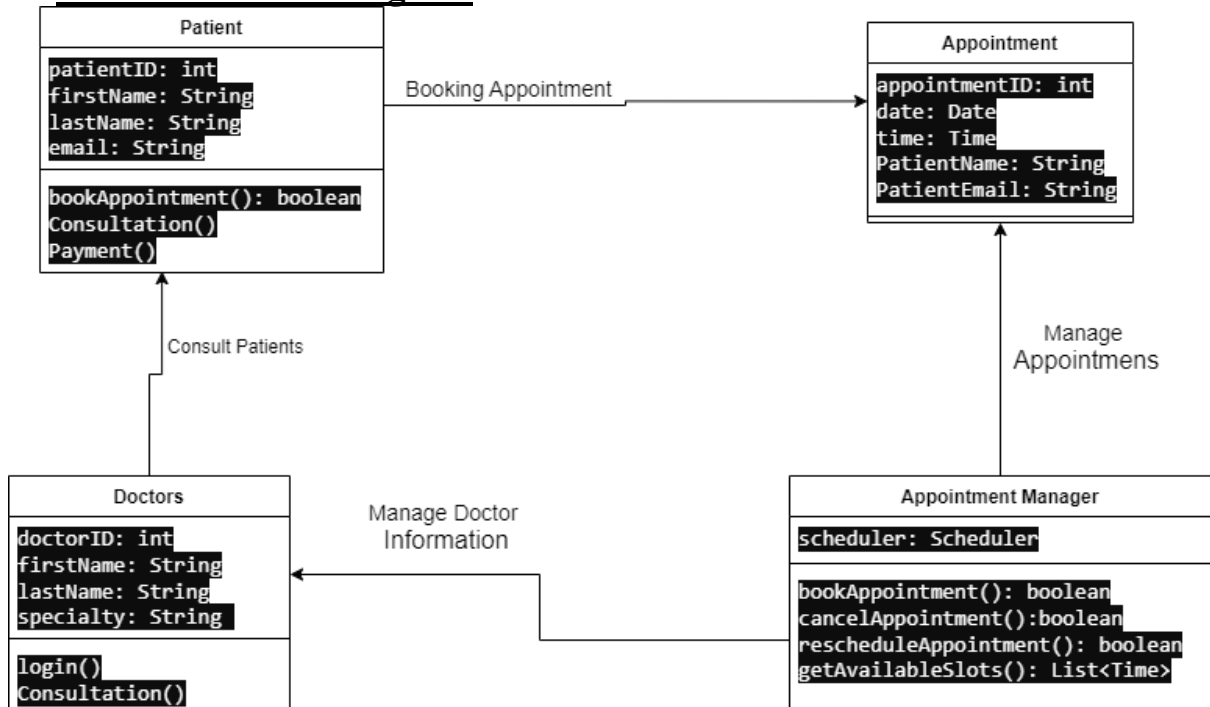


Fig 4.4: UML Class Diagram

4.6.3 UML Deployment Diagram

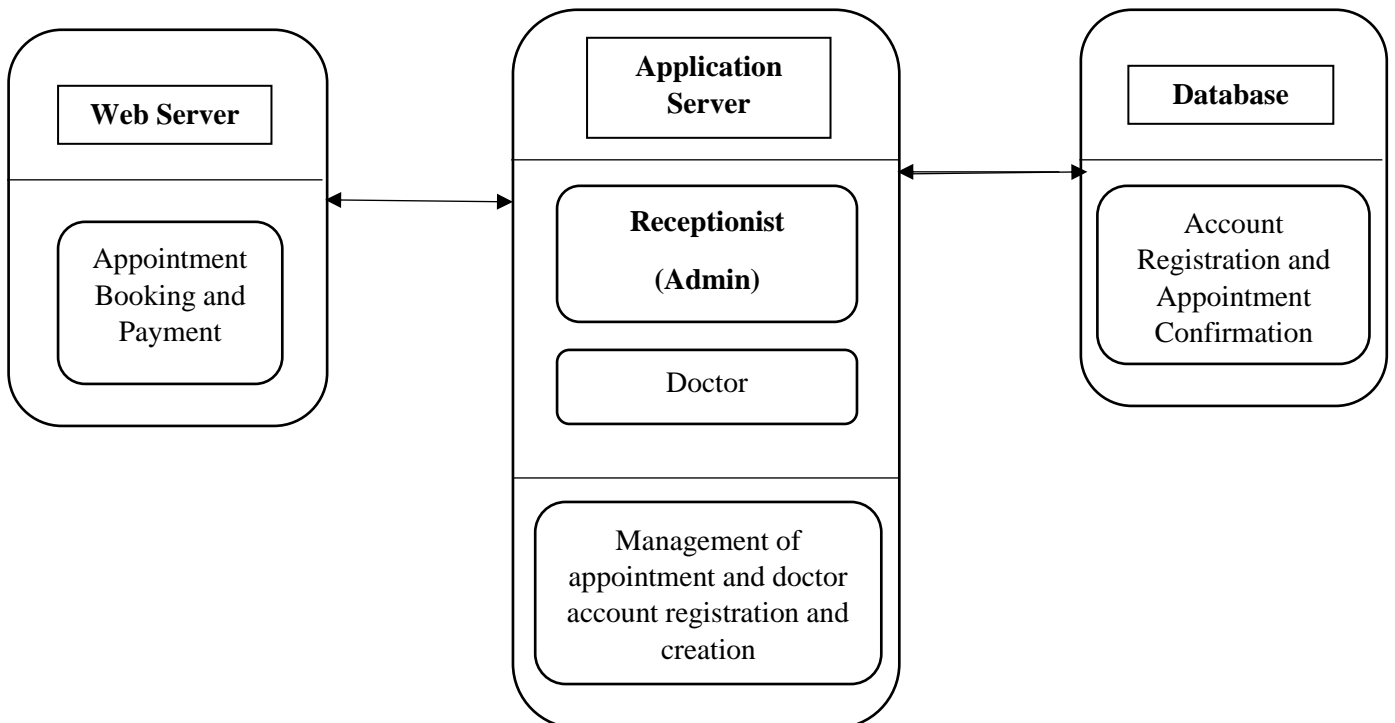


Fig 4.4: UML Deployment Diagram

4.7 Database Modelling

4.7.1 Entity Relationship Diagram

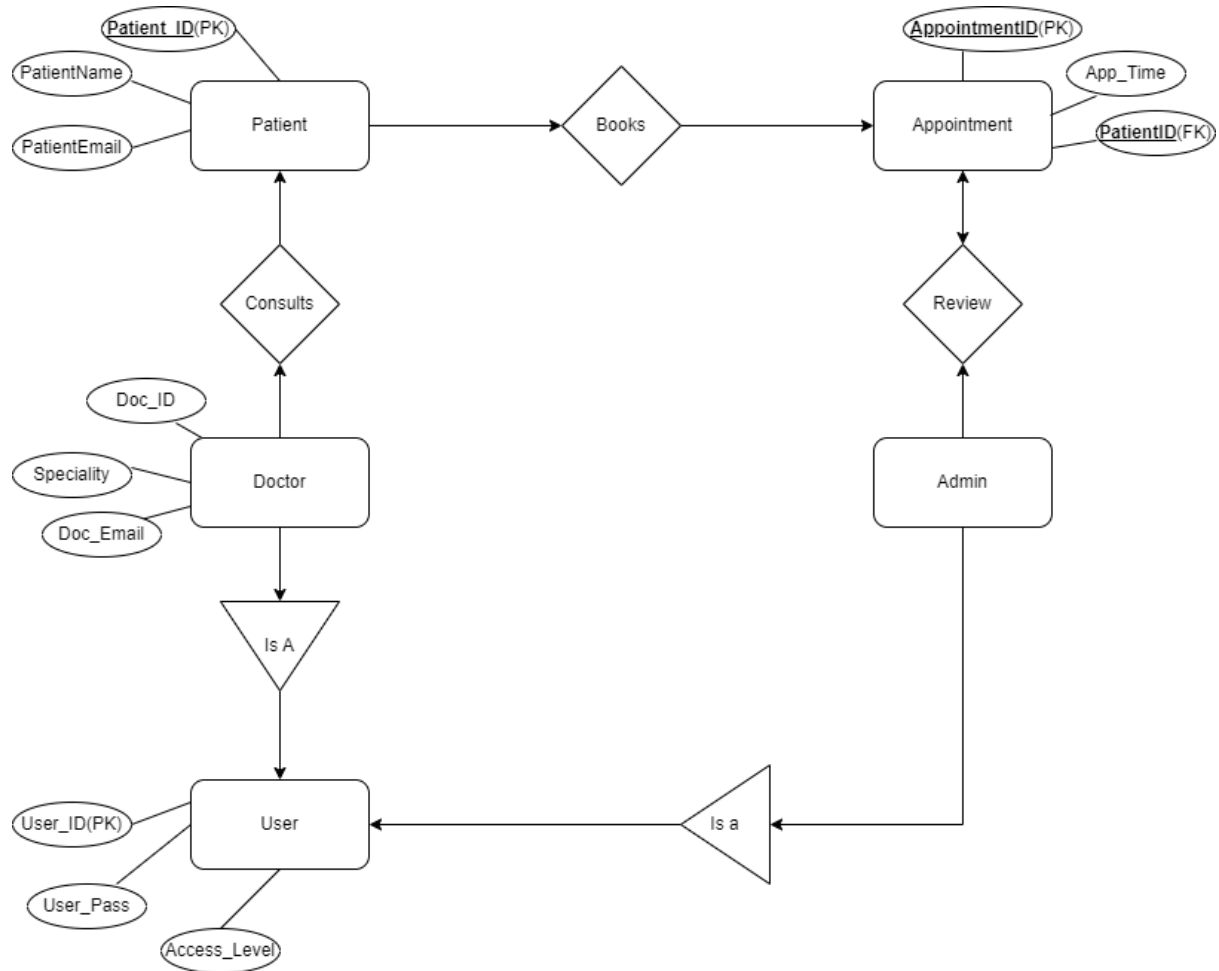


Fig 4.5: ER Diagram

4.7.2 Data Dictionary

The data dictionary for the appointment system provides detailed information about the database's metadata, including names, attributes, and data elements.

- **Doc_Id**: Unique identifier for each doctor in the database.
- **national_ID**: identifier for patients.
- **email**: The patients email address
- **appointment_datetimr**: Date the appointment is scheduled for
- **phone_number**: patients phone number
- **pmessage**: a message scripted by the patient when they book their appointment.
- **Department**: the department to which the appointment is booked

4.7.3 Relational Schema

The relational schema outlines the design and structure of the database tables containing appointment-related information.

This relational schema provides a structured representation of the database tables and their attributes, facilitating efficient data management and retrieval within the appointment system.

fullname	national_id	email	phone_number	department	Pmessage	appointment_datetime
Tapiwa Much	78-8783838L88	sally@gmail.com	0778567373	general		2024-04-25 11:20:00.000
Tendai Mbetsa	78_2344567K99	tendai@gmail.com	0772345678	general		2024-04-27 10:40:00.000
Takunda Masanga	55-7574738L99	salvatoremugabe@gmail.com	0772358268	general		2024-04-27 11:20:00.000

4.8 ALGORITHM DESIGN

Flowcharts play a crucial role in visualizing the logic and workflow of system modules before they are implemented as code. They provide developers with a clear understanding of the processes involved, reducing implementation time and facilitating easy documentation of the solution. Below are the flowcharts for the two modules associated with the appointments system.

Appointment Booking Flowchart

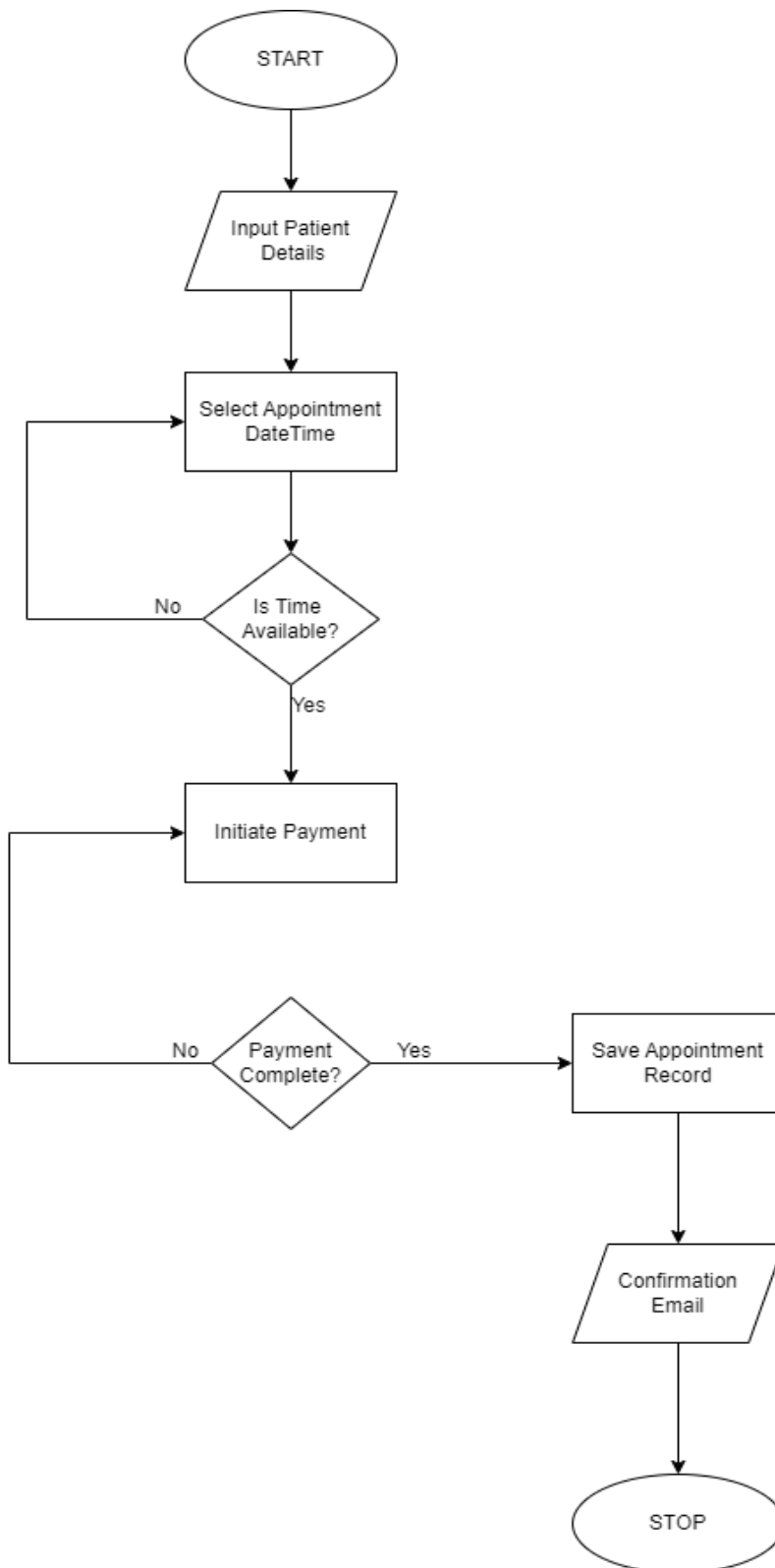


Fig 4.6: Booking flowchart

Appointment Management Module Flowchart

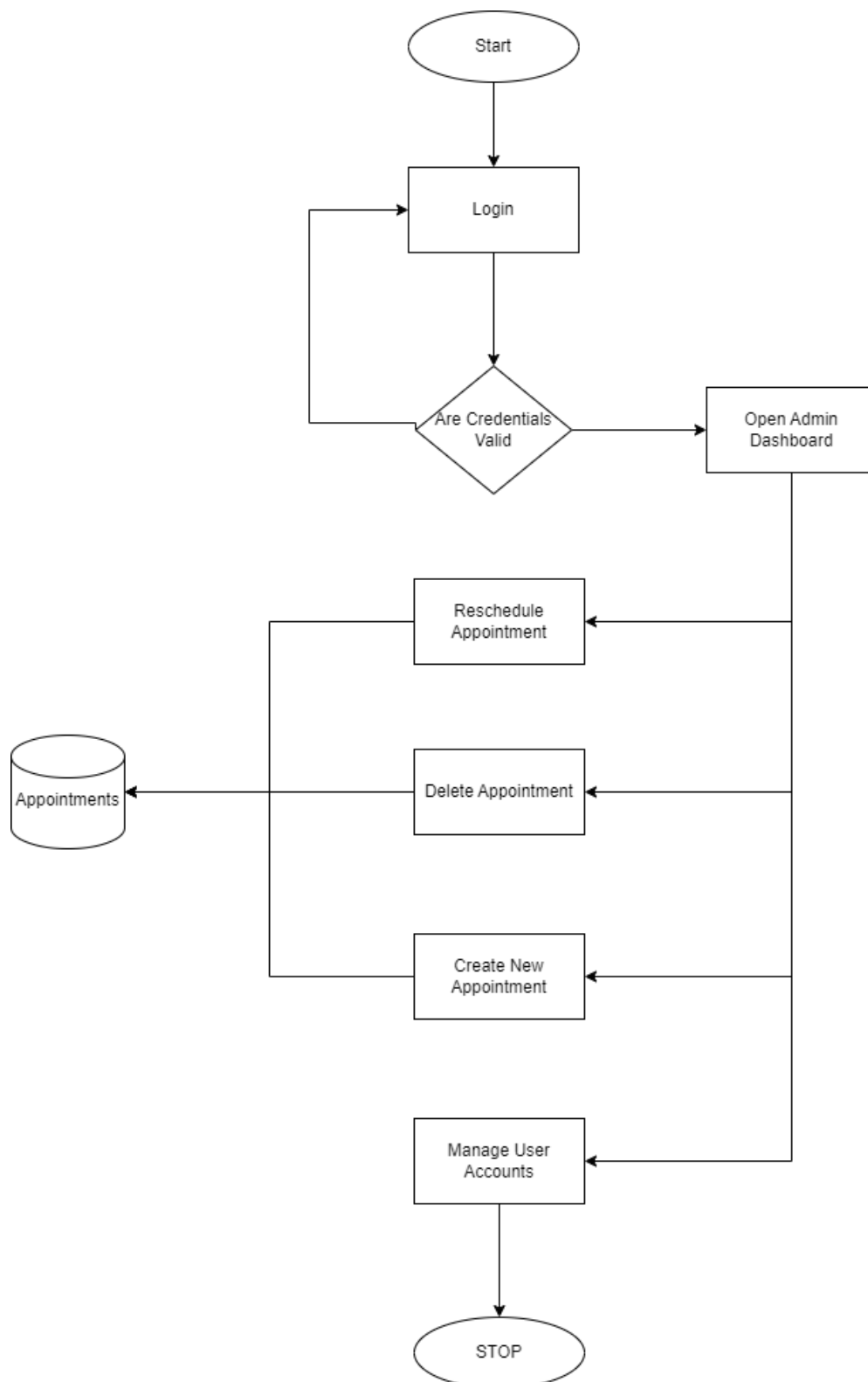


Fig 4.7: Appointment Management flowchart

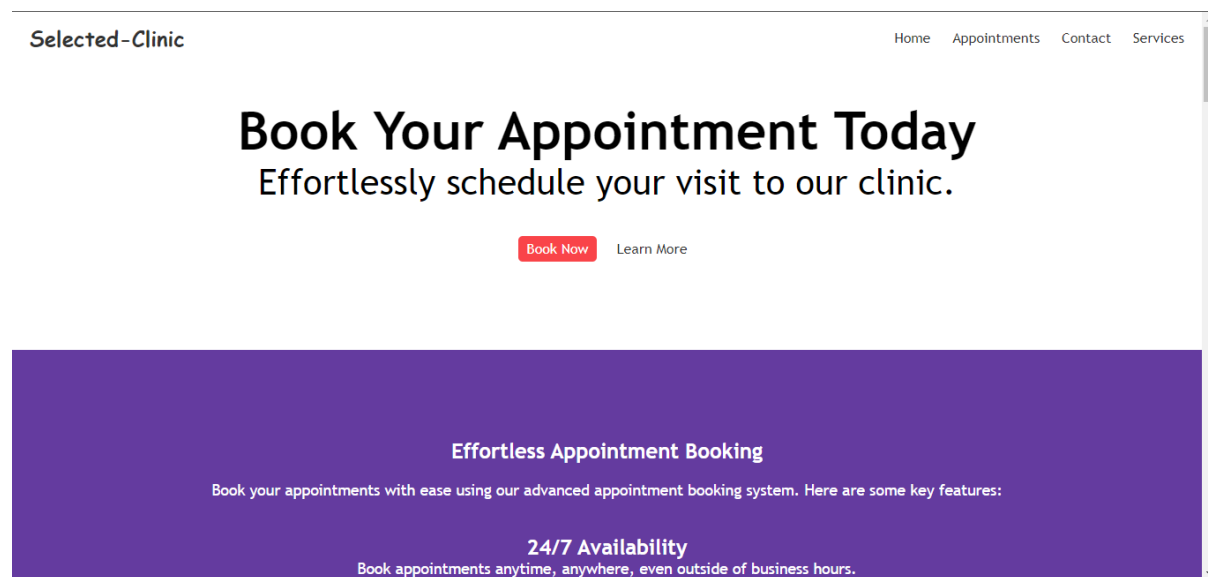
4.9 INTERFACE DESIGN

The appointments system is meticulously designed to adhere to the following key design principles, ensuring an intuitive and efficient user experience:

1. **Consistency:** Maintaining uniformity across the interface for easy navigation.
2. **User Familiarity:** Incorporating familiar terms and concepts for user convenience.
3. **Efficiency:** Optimizing performance for fast execution with minimal resources.
4. **User Guidance:** Providing assistance throughout the user journey for seamless interaction.
5. **Accessibility:** Ensuring the system is accessible to users with diverse needs and abilities.
6. **Scalability:** Designing the system to accommodate growth and increased usage over time.

These principles ensure an intuitive and efficient user experience.

Landing Page



Booking Page

Book Your Appointment

Ready to schedule your visit? Book your appointment now!

Select Department

dd/mm/yyyy

Select Appointment Time

Is Payment completed? ☐

Message (optional)

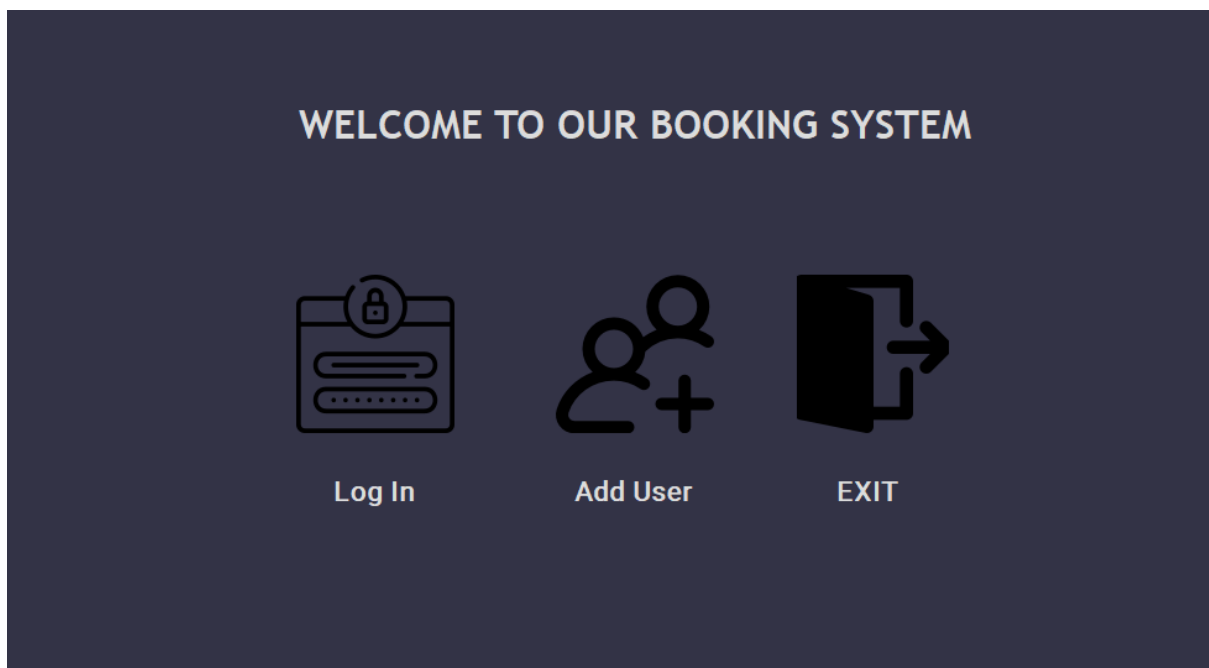
PayPal

☐ Debit or Credit Card

Powered by PayPal

Submit

Desktop App Landing Page



User Registration Page

Appointments System	User Registration	
<div>✕ Exit Application</div> <div>← Back To Login</div>	Staff ID/Username	<input type="text"/>
	First Name	<input type="text"/>
	Surname	<input type="text"/>
	Specialty	<input type="text"/>
	Phone Number	<input type="text"/>
	Email	<input type="text"/>
	Telephone Number	<input type="text"/>
<div>GenerateID</div>		
<div>Account Details</div> <div>Password <input type="password"/></div> <div>Re-enter Password <input type="password"/></div> <div>Show</div>		
<div>Save</div>		

Login Page

PLEASE ENTER USERNAME AND PASSWORD

LOGIN AS	<input type="text"/>
USERNAME	<input type="text"/>
PASSWORD	<input type="password"/>

LOGIN


CLEAR

EXIT

Admin Dashboard

Add New Appoinment

Review Accounts

 Exit Form

View Appointments

1 of 28


Name

	fullname	national_id	appointment_datetime
▶	Tapiwa Much	78-8783838L88	25/4/2024 11:20
	Tendai Mbetsa	78_2344567K99	27/4/2024 10:40
	Takunda Masanga	55-7574738L99	27/4/2024 11:20
	Tadiwa Masanga	55-7574738L99	27/4/2024 12:00
	Tapiwa Mbetsa	77-7372728K88	27/4/2024 13:20
	Tapiwa Mbetsa	77-7372728K88	27/4/2024 15:20
	Takunda Mbetsa	77-7372728K88	27/4/2024 14:00
	Tanatswa Mugabe	45-4747474L22	11/4/2024 12:40
	Tadiwa Mucha	66-6748392L22	27/4/2024 16:40
	Tanaka Mutumhe	77-4563829K44	27/4/2024 18:40
	Takunda Ching	58-8888577L55	26/4/2024 11:20
	Tanatswa Mugabe	38-5684848L88	4/4/2024 11:20
	Sal Muggy	75-8585858L22	23/4/2024 10:00
	Sal Muggy	75-8585858L22	23/4/2024 12:00

Appointment Management Page

Home

View Doctor Accounts

 Exit

Add Appointment

Full Name

National ID

Email

Phone Number

Department

Pmessage

Appointment Time Date

Save

PreviousAddNext

Delete

Review User Accounts Page

Home

Exit Form

View User Account Details

User_ID

	Users_ID	pass
*		

Doctors Profile Home

Patient Feedback

Exit Form

Doctors Profile

02/05/2024 23:13

Name

	fullname	email	appointment_datetime
▶	Tapiwa Much	sally@gmail.com	25/4/2024 11:20
	Tendai Mbetsa	tendai@gmail.com	27/4/2024 10:40
	Takunda Masanga	salvatoremugabe@gmail.com	27/4/2024 11:20
	Tadiwa Masanga	salvatoremugabe@gmail.com	27/4/2024 12:00
	Tapiwa Mbetsa	tanamuggy@gmail.com	27/4/2024 13:20
	Tapiwa Mbetsa	tanamuggy@gmail.com	27/4/2024 15:20
	Takunda Mbetsa	tanamuggy@gmail.com	27/4/2024 14:00
	Tanatswa Mugabe	sl@gmail.com	11/4/2024 12:40
	Tadiwa Mucha	tanamuggy21261@gmail.com	27/4/2024 16:40

4.10 CONCLUSION

This chapter delves into the design phase of the appointments system, outlining its architecture and key components. It defines the necessary requirements for the project's successful completion and evaluates its feasibility based on the established design and requirements. Given the anticipated benefits of the system, there's a clear imperative to develop a prototype to showcase its advantages. The project entails minimal requirements, necessitating a simple prototype with basic functionalities. Therefore, it is both feasible and practical to proceed with the development, testing, and presentation of this prototype.

CHAPTER 5: IMPLEMENTATION

5.1 Introduction

A software implementation strategy is a methodical approach to effectively integrating a software-based service into the workflow of an organization or an individual end user. This chapter focuses on the implementation process required to create our application to meet its stated requirements.

5.2 Code Conventions

The core of our system's code is developed using the .NET framework, leveraging its robustness and flexibility to ensure efficient performance and scalability. Additionally, the backend functionalities are implemented in PHP, harnessing its versatility and compatibility with various databases and web servers. This combination of technologies enables us to create a dynamic and responsive system that meets the diverse needs of our users while maintaining high standards of reliability and security.

Need for coding Conventions

Coding conventions are crucial in software development for both users and programmers for a variety of reasons, including the fact that unambiguous code allows developers to reuse code when necessary. This has the potential to significantly cut both costs and development efforts. Furthermore, applying coding standards would aid in the early detection of errors or perhaps their entire prevention. This improves efficiency throughout the software development process. As a result, locating and correcting flaws in software becomes much easier if the source code is written consistently.

1. Naming Conventions:

For our naming conventions, the code follows a combination of camel case and snake case naming conventions. Variables and functions are written in camel case, while constant values like database credentials are written in snake case.

2. Comments:

We also made use of comments to explain the purpose of certain sections of code and to provide additional insights. Comments are used to describe the establishment of the database connection, validation, and sanitization of form data, error handling, and the execution of SQL queries.

3. Indentation and formatting:

The use of consistent indentation with four spaces for each level of indentation was implemented. The opening and closing braces are placed on separate lines, and there are blank lines used to separate logical sections of code.

4. Conditional statement formatting:

We implemented consistent formatting for conditional statements (if and else) with the opening brace on the same line and the closing brace on a new line.

5. File organization:

The code includes the necessary require statements to import the required PHPMailer files. The code also follows a logical structure, with the database connection establishment, form validation and sanitization, SQL queries, and email sending code organized in separate sections.

6. Whitespace usage:

The use of whitespace in our code is to improve readability. Blank lines are used to separate logical sections of code, such as the establishment of the database connection, validation and sanitization of form data, execution of SQL queries, and email sending code.

To conclude, we made use of the stated coding conventions to allow for easier maintainability and readability for all group members when accessing the code.

Example code for the Booking Process

```
<?php
```

```
use PHPMailer\PHPMailer\Exception;
use PHPMailer\PHPMailer\PHPMailer;
require 'C:\xampp\htdocs\PHPMailer-master\PHPMailer-master\src\SMTP.php';
require 'C:\xampp\htdocs\PHPMailer-master\PHPMailer-master\src\PHPMailer.php';
require 'C:\xampp\htdocs\PHPMailer-master\PHPMailer-master\src\Exception.php';
```

```
$serverName = "SALVATOREM";
$databasename = "Health_Management_System";
$uid = ""; // Fill in your username
$password = ""; // Fill in your password
```

```
// Establish connection
$connectionInfo = array(
    "Database" => $databasename,
    "UID" => $uid,
    "PWD" => $password
);

$conn = sqlsrv_connect($serverName, $connectionInfo);
if ($conn === false) {
    die("Connection failed: " . print_r(sqlsrv_errors(), true));
} else {
    echo " ";
}
```

```
function sanitize_input($data)
{
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
```

```

// Check if the form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Validate and sanitize form data
    $fullname = sanitize_input($_POST['fullname']);
    $national_id = sanitize_input($_POST['national_id']);
    $email = sanitize_input($_POST['email']);
    $phone = sanitize_input($_POST['phone']);
    $department = sanitize_input($_POST['department']);
    $message = sanitize_input($_POST['message']);
    $appointment_date = sanitize_input($_POST['appointment_date']);
    $appointment_time = sanitize_input($_POST['appointment_time']);

    // Validate email address
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $errors[] = "Invalid email address";
    }

    // Validate phone number
    function validate_phone($phone)
    {
        // Remove non-numeric characters from phone number
        $phone = preg_replace('/^D/', "", $phone);
        // Check if phone number contains exactly 10 digits (assuming 10-digit format)
        return (strlen($phone) === 10 && is_numeric($phone));
    }

    // Validate national ID
    function validate_national_id($national_id)
    {
        // Regular expression pattern to match the Zimbabwean national ID format
        $pattern = '/^\d{2}-\d{7}[A-Z]\d{2}$/';

        // Check if the national ID matches the pattern
        if (preg_match($pattern, $national_id)) {
            return true; // National ID is valid
        } else {
            return false; // National ID is not valid
        }
    }

    // Check if there are any errors
    if (empty($errors)) {
        // No validation errors, proceed to save the booking data in the database
        // Combine appointment date and time
        $appointment_datetime = $appointment_date . ' ' . $appointment_time;

        // Prepare SQL query to check for existing appointments on the selected date and
        time
        $sql_check = "SELECT * FROM spnmt WHERE appointment_datetime = ?";
    }
}

```

```

$params_check = array($appointment_datetime);

// Execute SQL query
$stmt_check = sqlsrv_query($conn, $sql_check, $params_check);
if ($stmt_check === false) {
    die("Error executing query: " . print_r(sqlsrv_errors(), true));
}

// Check if there are any existing appointments at the selected date and time
if (sqlsrv_has_rows($stmt_check)) {
    echo "Sorry, the selected appointment time is already booked. Please choose a
different time.";
} else {
    // No existing appointments, proceed to insert the new appointment
    // Prepare SQL query to insert data into the "appointments" table
    $sql_insert = "INSERT INTO spnmt (fullname, national_id, email,
phone_number, department, Pmessage, appointment_datetime)
VALUES (?, ?, ?, ?, ?, ?, ?)";
    $params_insert = array($fullname, $national_id, $email, $phone, $department,
$message, $appointment_datetime);

    // Execute SQL query
    $stmt_insert = sqlsrv_query($conn, $sql_insert, $params_insert);
    if ($stmt_insert === false) {
        die("Error inserting data: " . print_r(sqlsrv_errors(), true));
    } else {
        // Show a pop-up message with the appointment details
        echo "<script>alert('Booking successful! Your appointment is on
$appointment_date at $appointment_time.');

```

```

        $mail->Body = "Dear $fullname,\n\nYour appointment has been
confirmed.\nDate: $appointment_date\nTime: $appointment_time\n\nThank you for
choosing Selected-Clinic.";

        $mail->send();
        echo 'Appointment confirmation email sent to ' . $email;
    } catch (Exception $e) {
        echo 'Failed to send appointment confirmation email. Error: ' . $mail-
>ErrorInfo;
    }
}
}
} else {
    // Validation errors occurred, send response with errors
    http_response_code(400);
    echo json_encode($errors);
}
} else {
    // Handle invalid request method
    http_response_code(400);
    echo "Invalid request";
}
?>

```

5.3 Coding Strategy

A few essential elements were included in the coding strategy for our appointment scheduling system including a desktop application for clinic administration and a web application for patient booking. First, patients would communicate with the web application by entering their information, choosing their appointment times, and receiving emails confirming their appointments. This includes back-end programming for server-side logic, database interactions, and email notifications in addition to front-end development for user interface design. The MVC (Model View Controller) architectural pattern was used to create this web application. This design pattern allows for the separation of application source code into loosely linked objects that aid in the execution of certain projects. Data security and integrity, input validation, and error management were considered and measures were put in place to ensure. However, the clinic management desktop program features were primarily scheduling staff, keeping track of patients, and scheduling appointments. To store and retrieve data, database connectivity was implemented.

Development Methodology

In this project, the Agile software development life cycle was used. It gathers information from multiple sources and user reviews. Customers were involved at every stage of the software development process thanks to this methodology. This suggests that every user's need will have been satisfied by the finished product when it launches.

Table 5.1: Coding Strategy Table

Frontend programming	Html, JavaScript, CSS, C#
Platform	Web platform, Desktop platform
Backend programming	PHP
Framework	.NET Framework
Database	Microsoft SQL Server
Architectural pattern	MVC architectural pattern
Methodology	Agile software development

5.4 Coding Review

The coding review refers to the systematic examination of the source code of the system. The primary objective is to identify and address any errors or issues that may have been missed during the initial stages of our system's development, ultimately enhancing the overall appearance and quality of the system. To review our code, we made use of a manual checklist. This checklist is to ensure that specific criteria, relevant to your appointment scheduling system, are carefully reviewed and evaluated during the process. By conducting this coding review, we were able to identify and resolve potential issues before they impacted the functionality and performance of your appointment scheduling system.

The following checklist was made to use to check whether the code conformed to its expectations.

Table 5.2: Manual Checklist Structure

General	ANSWER
▪ Is the user interface intuitive and easy to navigate?	YES
▪ Does the system provide sufficient feedback and validation to users during the appointment booking process?	YES
▪ Are appropriate error messages displayed to users when necessary?	YES
▪ Is the code well-structured and organized?	YES
▪ Are functions and classes logically grouped and named?	YES
▪ Are there any consistent indentation styles and spacing?	PARTIALLY
Performance	
▪ Are there any potential performance bottlenecks in the code?	NO
▪ Are database queries optimized and are indexes properly met?	YES
▪ Are there any unnecessary or redundant code segments that can be optimized?	NO
Testing and Documentation	
▪ Is the code adequately tested with unit tests or integration tests?	YES
▪ Is there sufficient documentation explaining the system's architecture, functionality, and usage?	YES
Security	

▪ Are sensitive data, such as passwords handled securely?	YES
▪ Are access controls and permissions implemented appropriately to restrict unauthorized access?	YES
Database Interaction	
▪ Are database connections established securely and effectively?	YES
▪ Are transactions properly handled where necessary?	YES
Data validation	
▪ Are appropriate validation checks in place for fields such as email addresses, phone numbers, and dates?	YES

5.5 Conclusion

To sum up, the project's development process was implemented to make sure the system and webpage were made in accordance with accepted standards and conventions as well as user needs

CHAPTER 6: SYSTEMS TESTING

6.1 Introduction

The purpose of this section is to validate the implemented software product's functionality and completeness. The system will be put to the test to find any vulnerabilities and see if the modules that are currently in place work as intended.

6.2 Testing Categories

To determine the overall effectiveness, the system will undergo two major testing activities, which are:

- ✓ White box testing
- ✓ Black box testing

6.2.1 White box testing

White box testing for the appointment scheduling system involves examining the internal structure, design, and code to ensure that they correctly implement the expected input-output flow. Testers have visibility into the system's code and perform tests to verify the anticipated outcomes.

Table 6.1: White box Test cases

Test case	Execution procedure	Expected result	Actual result	Test result (pass/fail)
Login Functionality	1. Provide valid username and password	Doctor is authenticated and redirected to their profile dashboard	Doctor directed to profile dashboard	Pass
Online Booking	1. Enter patient's details	Patient appointment is booked and get notified of successful booking	Patient appointment is booked and get notified of successful booking	Pass
Appointment Management	Add new appointment with valid details	Appointment is successfully added to the system	Appointment added	Pass
	View all appointments in the system	Appointment details are Visible to the administration	Details are viewed by administration	Pass
	Update existing appointment with new details	Appointment details are updated in the system	Appointment details updated	Pass
	Cancel existing appointment	Appointment is removed from the system	Appointment record is deleted	Pass
Error Handling	Provide invalid email or password	Error message is displayed indicating invalid credentials	Error message displayed	Pass
	Attempt to schedule appointment with invalid inputs	Error message is displayed indicating missing information	Error message displayed	Pass

	Attempt to schedule before payment	Error message indicating that payment needs to be complete before finalising appointment	Error message displayed	Pass
--	------------------------------------	--	-------------------------	------

6.2.2 Black box testing

This testing method entails assessing the system's functionality without prior knowledge of its internal mechanisms. Users interact with the system, and observations are made based on their inputs. The black box testing for our system includes the following aspects:

Test case	Execution procedure	Expected result	Actual result	Test result (pass/fail)
Login	1. Provide username, password and role 2. Click on the login button	User is redirected to his/her respective page	User's respective page is opened	Pass
Appointment Booking	1. Fill in the patient's details. 2. Select date and time from available options 3. Complete Payment 4. Click on the submit button	Patient must receive a message on screen notifying them that the appointment booking was successful and an email confirming the appointment and the time	Onscreen message and email are received	Pass
Appointment Management	Type in a patient's name into the text field	Filter all appointments and display only the required appointment	Records are filtered and the required appointment is displayed	Pass

Table 6.2 : Black box test case

6.3 TYPES OF TESTING AND RESULTS

After completing both black box and white box testing, the system will undergo more advanced forms of testing. These refined testing methods encompass:

- Functional testing
- Non-functional testing

6.3.1 Functional Testing

Functional testing involves evaluating whether software behaves according to predefined requirements using questions and answers. Testers, unaware of the internal system logic, focus on verifying if the system operates as intended. For the appointments system, functional testing was conducted post-integration of all modules to ensure alignment with the specified requirements outlined in Chapter 3.

PLEASE ENTER USERNAME AND PASSWORD

LOGIN AS

USERNAME

PASSWORD

Login Error X

Invalid username or password.

OK

LOGIN

CLEAR

EXIT

The Login and Booking phases were tested, clicking login using a wrong username or password resulted in an error message. Also, in the event of trying to book an appointment without completing the payment, the system would give an output that the user should first complete payment before successfully booking the appointment.

localhost:8080/AppointmentsWebsite/booking.html

07/21/2024

General

10/05/2024

12:00:00

Is Payment completed? ☐

Message (optional)

PayPal

Debit or Credit Card

Powered by PayPal

Submit

localhost:8080 says

Please complete the payment before submitting the form.

OK

localhost:8080 says

Please complete the payment before submitting the form.

OK

In the event of trying to submit the appointment booking without filling out all the required fields, an error message is produced asking the user to fill in the required field



The image shows a web form for appointment booking. It has five input fields: 'Full Name', '65-7573839L2', 'tana@gmail.com', 'Phone Number', and a dropdown menu labeled 'General'. An orange error message box with a white exclamation mark icon is positioned over the second field, displaying the text 'Please fill out this field.'.

6.3.2 Non-Functional Testing

This testing evaluates the software's non-functional characteristics, including compatibility and functionality, among others. It aims to assess the system's readiness based on these non-functional conditions, which are not covered in functional tests. Non-functional testing serves the following objectives:

- Enhancing the reliability, performance, manageability, and accessibility of the system.
- Improving the installation, configuration, execution, and monitoring processes of the system.
- Building and enhancing understanding of the system's functionalities and employed technologies.

Non-functional testing is based on the following characteristics and produced the results below:

Security: this defines how the system is protected from intentional and unintentional threats from both internal and external sources.

- Security testing has identified and addressed vulnerabilities in the appointment scheduling system, ensuring robust protection against unauthorized access, data breaches, and other security threats. Authentication, authorization, encryption, and data protection mechanisms are implemented effectively to safeguard user data.

Reliability: defines the degree to which the system executes the required tasks consistently and without error.

- The reliability testing confirms that the appointment scheduling system performs consistently and predictably over time. The system demonstrates resilience to failures, crashes, and errors, ensuring uninterrupted availability for users.

Availability: defines how much the user can rely on the system whilst it is running.

- Availability testing verifies that the appointment scheduling system maintains high uptime and accessibility, ensuring operational continuity for users. The system exhibits reliability, fault tolerance, and effective recovery mechanisms to minimize downtime and service interruptions.

Flexibility: outlines how quickly and easily it is to run the system in various hardware and software setups.

- Compatibility testing confirms that the appointment scheduling system works seamlessly across different devices, browsers, and operating systems. The system exhibits compatibility with a wide range of platforms, ensuring a consistent user experience for all users.

Scalability Testing: Scalability testing assesses a system's ability to handle increasing workload or data volume gracefully, without compromising performance or functionality.

- Scalability testing results indicate that the appointment scheduling system can effectively handle increasing workload and user demand over time. The system demonstrates scalability in terms of capacity, performance under high loads, and the ability to scale resources dynamically to accommodate growth.

6.4 Levels of Testing and Results

Upon the completion of the functional and non-functional testing were, the system will be subjected to more refined forms of testing. These forms of testing include:

- ✓ Unit Testing
- ✓ Integration Testing
- ✓ Systems Testing
- ✓ Acceptance Testing

6.4.1 Unit Testing

Unit testing is a form of testing where individual units of the system are checked for errors and vulnerabilities. A unit is the smallest testable part of a software. It normally is a single functional component that takes a single or multiple inputs. Through unit testing, each unit of the software code was tested to verify if it performs as expected. Unit testing of the system was performed and revealed the following:

Appointment Booking Unit:

- Unit testing has verified Appointments can be successfully booked by users after providing all necessary information and completing payments.

Login Unit:

- Users can login to the system only using their saved passwords, a wrong password or username will result in users being denied access to the system

Appointment Management Unit:

- Administration can successfully view, update, reschedule or delete appointment records as required.

6.4.2 Integration Testing

Integration Testing is a form of testing where software modules are integrated alogically and tested as a group. Integration testing was conducted to expose defects in the interaction between software modules when they are integrated. It focuses on checking data communication amongst modules.

Results of the Integration Test: The system forms perfectly blend into each other to work together where required. Login form leads to corresponding dashboards depending on the login information and data submitted on the website is visible to the administrator using the desktop application, who can then modify this data when necessary.

6.4.3 Systems Testing

This level of testing evaluates the complete and fully integrated appointments software to validate its end-to-end functionality. The system underwent the following system testing techniques:

- **Hardware/Software Testing:** This involved testing the interactions between the hardware components and the appointments software to ensure compatibility and smooth operation.
- **Recovery Testing:** This was conducted to assess the software's reliability and effectiveness in recovering from potential crashes or failures, ensuring that it can resume normal operation without data loss or corruption.

Results of Hardware Testing: The hardware testing phase revealed that the appointments software operated seamlessly with various hardware configurations, including different operating systems and device types. Compatibility issues were minimal, and the software demonstrated robust performance across diverse hardware environments.

Results of Recovery Testing: During recovery testing, the appointments software successfully recovered from simulated crashes and system failures. It demonstrated prompt recovery and data integrity preservation, ensuring that critical appointment information remained intact even after unexpected disruptions. This confirmed the software's reliability and resilience in real-world usage scenarios.

6.4.4 Acceptance Testing

Acceptance testing is a crucial technique to ensure that the appointments system meets the specified requirements and is ready for deployment to end users. It verifies compliance with business requirements and assesses if the system meets the necessary criteria for user acceptance. The forms of acceptance testing conducted include:

- **Alpha Testing:** This involves rigorous testing performed internally to identify and resolve any potential issues or bugs before the final software release to end users.
- **Beta Testing:** This stage involves testing the application in a real-world environment by actual clients before the official release. It allows for feedback gathering and validation of the system's functionality and usability in real-world scenarios.

Results of Alpha Testing: During alpha testing, internal testers meticulously evaluated the appointments system, identifying and addressing any software bugs or usability issues. This phase ensured that the system met internal quality standards and was ready for external testing.

Results of Beta Testing: In beta testing, real clients used the appointments system in their everyday workflows to provide feedback on its performance and usability. This feedback was valuable in refining the system, addressing user concerns, and ensuring that it met the needs and expectations of its intended users.

6.5 SYSTEM EVALUATION

System evaluation has involved running the software under known predefined conditions with defined inputs and documenting the outcomes. The appointment scheduling system has been evaluated based on predefined criteria, including performance, reliability, security, and user satisfaction. The outcomes of the evaluation process indicate that the system meets expectations and performs effectively under various conditions.

6.6 CONCLUSION

In conclusion, this chapter was focused on the testing of the system for efficient, effectiveness and also security provided by the system. Several tests were done using all possible inputs that can be entered by users and the systems has proved to be outstanding prior to all the tests.

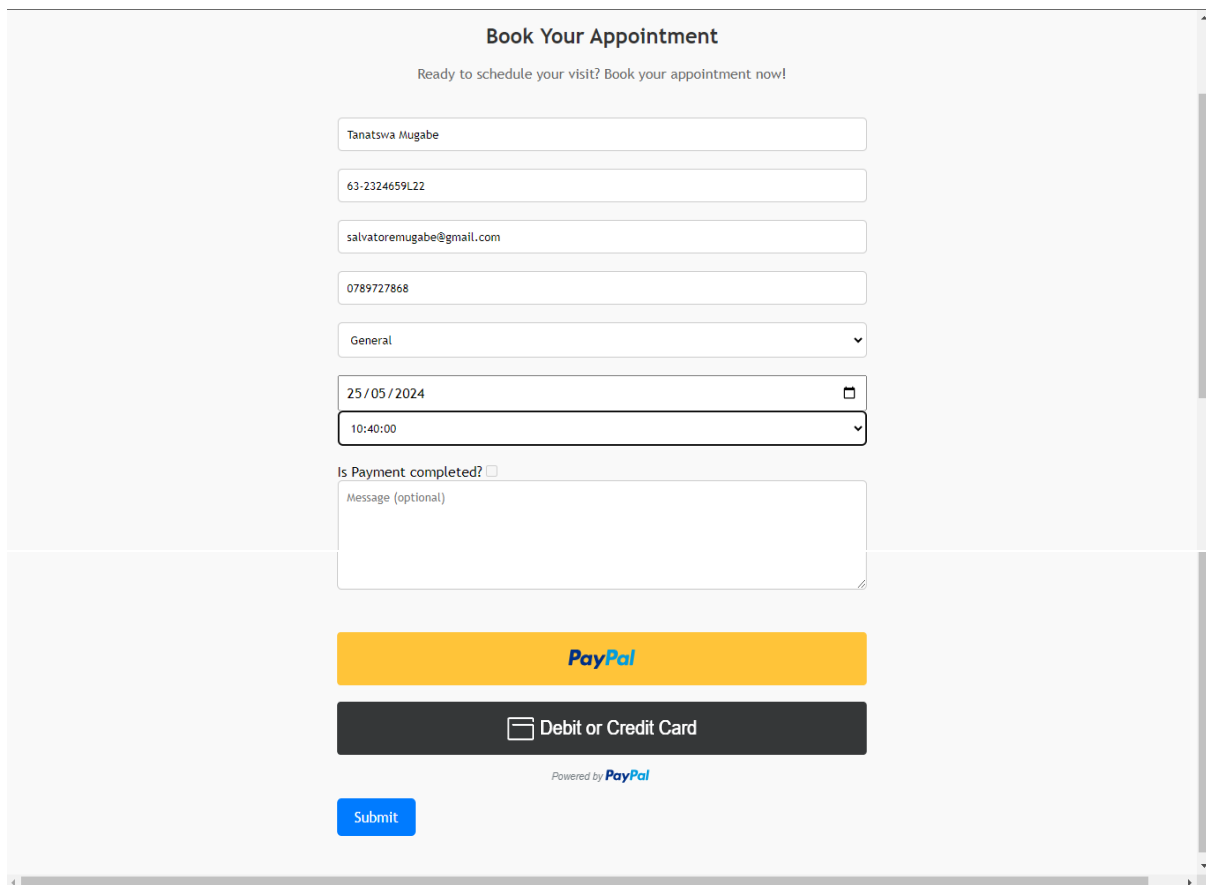
CHAPTER 7: CONCLUSION

7.1 Introduction

In terms of achieving the intended goals, our project has been successful. The few issues that arose throughout the project's development are evidence that the project was carefully designed. Although it worked well and is prepared for use. This chapter will examine the outcomes, the extent of upcoming work, and the system operation suggestions.

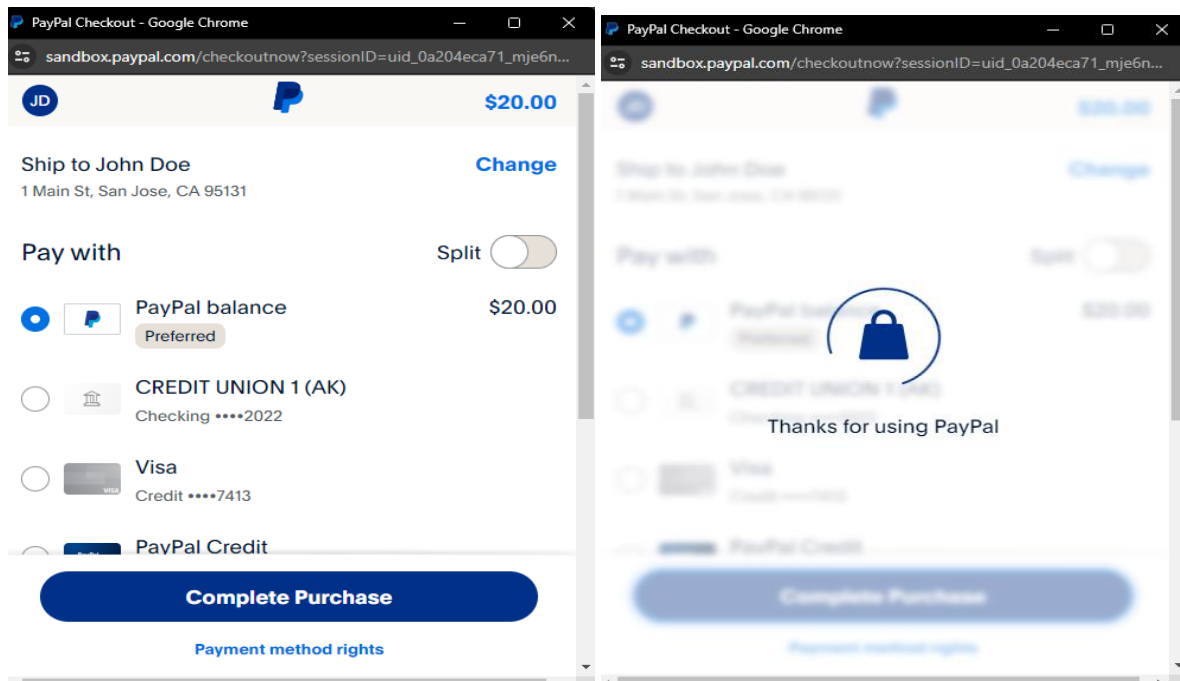
7.2 Results

Our system was able to satisfy all the requirements and expected functionalities. It was able to allow patients to book their appointments on the website as well as update the appointments on the system at the selected clinic.



The screenshot displays a web form titled "Book Your Appointment" with the subtitle "Ready to schedule your visit? Book your appointment now!". The form contains several input fields: a text field for the name "Tanatswa Mugabe", a text field for the ID number "63-2324659L22", a text field for the email address "salvatoremugabe@gmail.com", a text field for the phone number "0789727868", a dropdown menu for the appointment type set to "General", a date picker for "25/05/2024", and a time picker for "10:40:00". Below these fields is a checkbox for "Is Payment completed?" which is unchecked, followed by a text area for an optional message. At the bottom, there are two large buttons: a yellow "PayPal" button and a dark grey "Debit or Credit Card" button. A small "Powered by PayPal" logo is visible below the payment buttons. A blue "Submit" button is located at the very bottom of the form.

In addition to that, the system is successfully processing transactions via PayPal



Clinic Administrators are also able to manage reschedule, delete or manually add appointments via the desktop application designed for their administrative purposes. The screen for that function is shown below.

Home

View Doctor Accounts

Exit

Add Appointment

Full Name	Tapiwa Mbetsa
National ID	77-7372728K88
Email	tanamuggy@gmail.com
Phone Number	0737373737
Department	general
Pmessage	
Appointment Time Date	27/4/2024 13:20

Previous

Save

Next

Add

Delete

7.3 Scope of Future Work

In considering future enhancements for the appointment system, there is significant scope for integrating artificial intelligence (AI) and mobile technology to further optimize functionality and user experience. AI algorithms could be employed to analyse appointment data, patient preferences, and historical trends to offer personalized scheduling recommendations, reducing wait times and improving overall efficiency. Additionally, AI-powered chatbots could be implemented to provide real-time assistance and appointment booking support, enhancing accessibility and user engagement. Furthermore, leveraging mobile technology could enable seamless appointment management on-the-go, allowing patients to schedule, reschedule, or

cancel appointments conveniently from their smartphones. By incorporating AI and mobile capabilities, the appointment system can evolve into a more intelligent, intuitive, and user-centric platform, ultimately enhancing healthcare service delivery and patient satisfaction.

Corrective Maintenance:

Our approach to corrective maintenance involves regular monitoring and inspection of the system and website for errors and performance-related issues. We will promptly address any identified problems with the application's logic, coding, and design to ensure optimal functionality and user experience. This proactive stance ensures that any bugs or issues are swiftly resolved to maintain system reliability and minimize disruptions for users.

Adaptive Maintenance:

Given the evolving nature of the healthcare environment and scheduling trends, we will implement updates to the system on a quarterly basis. These updates will reflect changes in user requirements, industry standards, and technological advancements. By staying abreast of these developments, we can ensure that the system remains relevant, efficient, and aligned with the needs of users and stakeholders.

Perfective Maintenance:

As part of our commitment to continuous improvement, perfective maintenance focuses on enhancing the system's performance, usability, and functionality over time. This involves identifying opportunities for refinement and optimization based on user feedback, emerging technologies, and best practices. By proactively seeking ways to enhance the system's capabilities, we can ensure that it continues to meet the evolving needs and expectations of users while delivering maximum value and utility.

Maintenance Activities

1. User Support and Training:

The team provides ongoing support to address user inquiries, guide new users, and troubleshoot issues for users at the clinics.

2. System Upgrades:

Assessing and implementing upgrades to the current software versions, frameworks, and technologies to keep the system up-to-date.

3. User Interface Enhancements:

We make improvements to the user interface to enhance the user experience for the patients and clinic administration and streamline the booking process.

4. Backup and Disaster Recovery:

Implementing regular data backups and disaster recovery measures to minimize data loss in case of system failures or disasters that could occur at the clinics.

5. Security Updates:

Applying security patches and updates to protect against potential vulnerabilities and ensure patients' and doctors' details such as their emails and payment details are safe.

6. Regulatory Compliance:

We ensure compliance with relevant laws, regulations, and industry standards, such as data protection regulations and healthcare privacy requirements.

7.4 APPENDIX

7.4.1 References

1. P. J. Schulz et al., "Mobile Health Apps to Facilitate Self-Care: A Qualitative Study of User Experiences," PLoS ONE, vol. 11, no. 5, May 2016, Art. no. e0156164.
2. K. Singh et al., "The role of data science in healthcare advancements: applications, benefits, and future prospects," Health Information Science and Systems, vol. 9, no. 1, Dec. 2021, Art. no. 10.
3. M. T. Ortuño et al., "Operations research in global health: a scoping review with a focus on the themes of health equity," Global Health Research and Policy, vol. 2, no. 1, Dec. 2017, Art. no. 17.
4. S.-Y. Lee et al., "A systematic review on mHealth applications for patients with chronic obstructive pulmonary disease," BMC Medical Informatics and Decision Making, vol. 20, no. 1, Dec. 2020, Art. no. 12.
5. A.-M. Tully et al., "The impact of mobile phone-based interventions on maternal-child nutrition outcomes: a systematic review," Globalization and Health, vol. 16, no. 1, Dec. 2020, Art. no. 111.
6. "Design and Implementation of an Online Appointment Scheduling System" by H. T. Abdel-Baseer, S. F. Abdelkader, and M. F. Tolba: This study presents the design and implementation of an online appointment scheduling system, discussing the system architecture, user interface design, and security considerations.
7. "Appointment Scheduling Algorithms: A Comprehensive Survey" by Seyed Mohammad Mirjalili, Andrew Lim, and Sadaf R. Alam: This survey paper provides an overview of various appointment scheduling algorithms and techniques, discussing their advantages, limitations, and applicability to different scheduling scenarios.
8. "Appointment Scheduling Systems: A Review" by S. R. Nidhi and S. R. Sudha: This review paper explores different appointment scheduling systems, including their features, functionalities, and the benefits they offer to organizations and users.

Glossary

- .NET Framework: A software framework developed by Microsoft for building and running applications.
- Acceptance Testing: Testing conducted to determine if a system meets acceptance criteria and is ready for deployment.
- AppointEase: Refers to the appointment system discussed in the conversation.
- Black Box Testing: Testing the functionality of a system without knowledge of its internal workings.
- DFD (Data Flow Diagram): A diagram that represents the flow of data within a system.
- ERD: Entity-Relationship Diagram, a visual representation of the relationships between entities in a database.
- Functional Testing: Testing the functionality of a system to ensure it meets specified requirements.
- Functionality: The capability of a system or software to perform specific tasks or operations.
- Maintenance: Activities performed to ensure the continued functioning and improvement of a system or software.
- Non-functional Testing: Testing the performance, reliability, and other attributes of a system beyond its functional requirements.
- PHP: Hypertext Preprocessor, a server-side scripting language used for web development.
- Prototype: A preliminary version of a system or product used for testing and evaluation.
- System Overview: A description of the features, functionality, and purpose of the system.
- System Testing: Testing the entire system as a whole to ensure it functions correctly in different scenarios.
- UML: Unified Modeling Language, a standardized modeling language used in software engineering.
- Use Case: A description of how a user interacts with a system to achieve a particular goal.
- White Box Testing: Testing the internal logic and structure of a system.