

Title of the Project

Care Pulse Web System

Submitted By

Antor Daury

ID: 193-16-462

Department of Computing & Information System

Daffodil International University

Supervised By

Md Mehedi Hassan

Lecturer

Department of Computing & Information System

Daffodil International University

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in
Computing and Information System



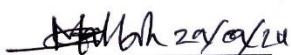
Department of Computing and Information System
Daffodil International University.
Dhaka, Bangladesh

Submission Date: 29/09/2024

APPROVAL

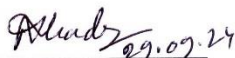
This Project titled “**Care Pulse Web System**”, Submitted by **Antor Daury**, ID No: **193-16-462** to the Department of Computing & Information System, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computing & Information System and approved as to its style and contents. The presentation has been held on 29-09-2024.

BOARD OF EXAMINERS


29/09/24

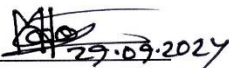
Md Sarwar Hossain Mollah
Associate Professor and Head
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Chairman


29.09.24

Md. Nasimul Kader
Assistant Professor
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner


29.09.2024

Md. Mehedi Hassan
Lecturer
Department of Computing & Information System
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Md. Miftah Uddin
Head of SBU, Brain Station 23

External Examiner

Declaration

I hereby declare that; this project has been done by me under supervision of **Md. Mehedi Hassan, Lecturer**, department of Computing and Information System (CIS) of Daffodil International University. I am also declaring that this project or any part of there has never been submitted anywhere else for the award of any educational degree like, B.Sc., M.Sc., Diploma or other qualifications.

Supervised By


29.09.2024

Md. Mehedi Hassan
Lecturer
Department of CIS
Daffodil International University

Submitted By



Name: Antor Daury
ID: 193-16-462
Department of CIS
Daffodil International University

Acknowledgement

Completing the 'Care Pulse ' project through research has been a fulfilling endeavor. I am deeply grateful for divine guidance throughout my academic journey.

I extend my sincerest appreciation to my supervisor, Mr. Md. Mehedi Hassan, Lecturer, Department of CIS, Daffodil International University. His invaluable time, guidance, and encouragement were instrumental in the successful completion of this project. His expertise, positive outlook, and understanding significantly contributed to my learning and growth. I am truly indebted to his support.

Executive Summary

Care Pulse is an advanced healthcare management system designed to optimize communication and streamline appointment processes between patients, doctors, and administrators. Utilizing state-of-the-art technologies including NodeJS, Express, Prisma, WebRTC (via Agora.io), PostgreSQL, and Next.js, the platform ensures secure and efficient data management. Key features include real-time communication, secure user authentication, comprehensive appointment scheduling, and detailed patient profiles. Admins can manage doctor accounts and appointments, doctors can access patient histories and manage prescriptions, and patients can book appointments, manage medical records, and provide feedback. The system also incorporates AI for disease identification from prescribed medicines. Payment integration and automated notifications enhance user experience, making healthcare interactions seamless and efficient. Care Pulse aims to revolutionize healthcare management by providing a robust, user-friendly platform that caters to the needs of all stakeholders.

Table of Contents

Acknowledgement.....	iii
Executive Summary.....	iv
Chapter 1 - Introduction.....	1
Chapter 2 - Initial Study.....	2
2.1 Project proposal:.....	2
2.2 Background of the Project:.....	9
2.3 Problem areas:	9
2.4 Possible solution:	10
Chapter 3 - Literature Review.....	11
3.1 Discussion on problem and solution domain based published articles:	11
3.2 Recommended approach:	11
3.2 Why to Use:	15
3.4 Implementation Plan:.....	15
Chapter 4 – Planning.....	16
4.1 Project Plan:.....	16
4.1.1 Work Backdown Structure (WBS):.....	16
Table 4.1: WBS for Care Pulse.....	16
4.1.2 Time Duration/Time-Boxing:.....	17
4.1.3 Gantt Chart.....	18
4.2 Test plan:	18
4.2.2 Required Test:.....	19
Chapter-5 Feasibility.....	20
5.1 Economic Feasibility	20
5.2 Technological Feasibility	21
5.3 Operational Feasibility.....	21
5.4 Cost Benefit Analysis	21
Chapter 6 – Foundation.....	23
6.1 Over All Requirement List:.....	23
6.1.1 List of functional requirements:.....	23
6.1.2 List of Not Functional Requirement:	23
6.2 What Technology Implemented:	24
6.3 Recommendations and Justifications:.....	25

Chapter 7-Explorations	26
7.1 Old System Use Cases:	26
7.2 Old system activity diagram:	27
7.3 Prioritization Requirement List (PRL):	29
7.4 Prototype of the System:	29
Chapter 8 - Engineering	32
8.1 New System Modules:	32
8.2 New System Use Case diagram	32
8.3 New system ERD Diagram:.....	33
8.4 System Interface Design/Prototype:.....	33
Chapter 9 - Deployment/Development	35
9.1 Core Modules Coding Simples:.....	35
9.2 Possible problem breakdown:	38
Chapter 10 - Testing.....	39
10.1 Unit Testing:	39
10.2 Module testing:	40
10.3 Integration Testing	41
Chapter 11-Implementation.....	42
11.1 Training:	42
11.2 Big bang:	43
11.3 Recommended implementation process:	43
Chapter 12 - Critical Appraise and Evaluation	44
12.1 Objective Cloud be Met:	44
12.1.2 How much better could have been done:	44
12.1.3 How much better could be the features of Solution ?	45
12.1.4 Which Features Could Not be Touch.....	45
Chapter 13- Conclusion	46
13.1 Summery of the Project	46
13.2 Goal of the Project	46
13.3 My experience	47
References	48

List of Figures

Figure 2.1 Demo of Care Pulse	2
Figure 3.1 Waterfall model diagram.....	12
Figure 3.2 Agile model diagram.....	13
Figure 3.3 DSDM Process Methodology	14
Figure 4.1 Gantt-Chart.....	18
Figure 7.1 Usecase diagram of Care Pulse	26
Figure 7.2 Admin-activity diagram of Care Pulse.....	27
Figure 7.3 User-activity diagram of Care Pulse	28
Figure 7.4 Doctor selection prototype	30
Figure 7.5 Feedback prototype	30
Figure 7.6 Homepage Care Pulse	31
Figure 8.1 Newsystem Usecase diagram	32
Figure 8.2 ERD-Diagram	33
Figure 8.3 Monitoring part	33
Figure 8.4 Time schedule	34
Figure 8.5 User profile.....	34
Figure 8.6 Test selection.....	34
Figure 9.1 Banner Code.....	35
Figure 9.2 ImageApi code	36
Figure 9.3 Transaction Code.....	36
Figure 9.4 Patient-account Code.....	37
Figure 9.5 Authentication Code.....	37
Figure 10.1 Login System-testing	40
Figure 10.2 Module-testing	41
Figure 10.3 Integretion-testing	41

List of Tables

Table 2.1 Time boxing table for Care Pulse	7
Table 4.1 WBS for Care Pulse.....	16
Table 4.2 Care Pulse System Execution.....	17
Table 5.1 Economic Feasibility	20
Table 5.2 Cost Benefit Analysis	22
Table 5.3 Total revenue	22
Table 7.1 Prioritized-Requirement List of Care Pulse	29
Table 10.1 Testing-unit.....	39
Table 10.2 Module-Testing	40
Table 11.1 Training table of Care Pulse	42

Chapter 1 - Introduction

Project Description

Care Pulse is a healthcare management system designed to enhance communication and streamline appointment processes between patients, doctors, and administrators. Care Pulse offers robust functionalities for various user roles like administrators can manage doctor accounts and appointment schedules, doctors can view and set appointments, access patient medical histories, and generate detailed prescriptions, patients can book appointments, manage their medical records, and securely pay consultation fees. Real time communication, automated email notifications, and AI driven disease identification from prescriptions further elevate the user experience. This comprehensive system not only simplifies administrative tasks but also fosters better patient doctor interactions, ultimately aiming to improve the overall quality of healthcare services.

Chapter 2 - Initial Study

2.1 Project proposal:

1. Initial Conception

A. A Brief description of the project

Care Pulse is an innovative healthcare management system designed to streamline communication and appointment processes between patients, doctors, and administrators. Utilizing cutting-edge technologies like NodeJS, Express, Prisma, WebRTC (via Agora.io), PostgreSQL, and Next.js, the platform ensures secure data management and real-time interactions. Key features include seamless appointment scheduling, comprehensive patient profiles, secure payment processing, and automated email notifications. Doctors can generate detailed prescriptions and attach medical notes, while patients can easily manage their medical history and book appointments. The system also incorporates AI for disease identification based on prescribed medicines, enhancing the overall healthcare experience..

B. Proof of concept:

i. Prototype:

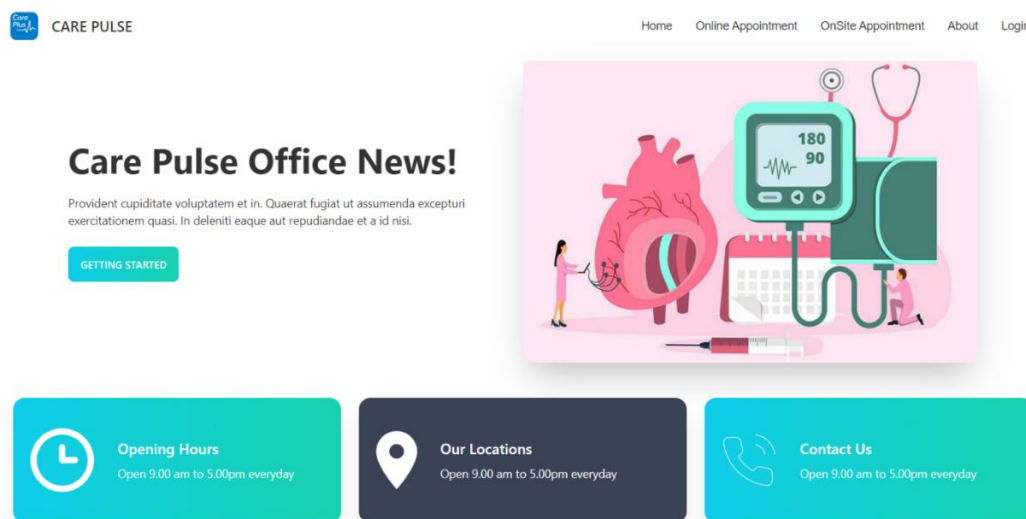


Figure 2.1: Demo of a care pulse

ii. Initial research:

Market Viability:

The market viability of Care Pulse is a digital healthcare solutions and the widespread adoption of telemedicine. As the healthcare industry transitions towards more efficient and patient-centric models, the need for comprehensive platforms that streamline communication, appointment management, and medical record keeping has become paramount. Care Pulse addresses these needs by leveraging cutting-edge technologies such as NodeJS, WebRTC, and PostgreSQL to provide a robust and secure system for doctors, patients, and administrators. The integration of real-time communication and AI powered disease identification enhances the platform's appeal, offering advanced features that set it apart from competitors. Additionally, the system's ability to facilitate seamless payment processing and automated notifications adds significant value, ensuring a smooth user experience. With a growing emphasis on remote healthcare services, Care Pulse is well-positioned to capture a substantial share of the digital healthcare market.[1]

2. Feasibility Study

A. Operational Feasibility:

The use of NodeJS, Express, Prisma, and PostgreSQL ensures a robust backend capable of handling high volumes of data and user interactions efficiently. WebRTC via Agora.io enables seamless real-time communication between doctors and patients, crucial for remote consultations. The single-page application architecture using Next.js provides a smooth and responsive user experience, minimizing load times and enhancing usability. Additionally, secure user authentication, payment processing, and automated email notifications streamline administrative tasks, reducing manual effort and errors. The platform's design addresses the core needs of administrators, doctors, and patients, ensuring that each role has access to essential functionalities that enhance operational workflows. With detailed prescription management, comprehensive patient profiles, and AI-powered disease identification, the portal is well-equipped to meet the dynamic demands of modern healthcare environments, thus ensuring its operational success and sustainability.

B. Technical Feasibility:

The Care Pulse leverages proven technologies like NodeJS, Express, Prisma, and PostgreSQL, ensuring robust server-side development and efficient data management. WebRTC (Agora.io) enables reliable real time communication. Next.js offers a performant, scalable framework for the front end. The integration of secure authentication, payment processing, and automated notifications is well-supported by these technologies. The project's success is bolstered by the widespread adoption and community support of these tools, ensuring maintainability and scalability.

Hardware

- Computer.
- 8 GB RAM.
- 1TB SSD.
- Internet connectivity.

Software

- NodeJS.
- Express.
- Prisma.
- PostgreSQL.
- WebRTC.
- Next.js.
- Visual studio code.
- Browser.

C. Economic feasibility:

The Care Pulse potential to streamline healthcare services, reduce administrative costs, and generate revenue through subscription fees from doctors and payment processing fees. Initial development and ongoing maintenance costs are offset by improved efficiency, increased patient engagement, and scalable opportunities in telemedicine and AI integration.[2]

Costs are divided into three parts:

- Software cost.
- Hosting cost.
- Storage cost.

C. Comparative analysis:

In Bangladesh, healthcare systems are often burdened by inefficiencies and limited access to technology. Comparing Care Pulse with traditional practices, the platform offers significant advancements. It enables seamless communication, efficient appointment scheduling, and comprehensive patient record management, reducing administrative burdens. Real time communication via WebRTC addresses the challenge of in-person visits in rural areas. Secure payment integration and automated notifications streamline processes for patients and doctors. Compared to existing systems, Care Pulse leverages modern technologies to enhance accessibility, improve healthcare delivery, and ensure better data management, aligning with Bangladesh's growing need for digital health solutions.

3. Foundation

A. Define goal and object:

Goal: The Care Pulse project is to create a seamless, efficient, and secure healthcare management system that enhances communication and appointment processes among patients, doctors, and administrators. By leveraging modern technologies, the portal aims to provide a user-friendly interface for managing medical appointments, patient records, prescriptions, and real-time consultations, ultimately improving the overall patient care experience.

Objectives:

- Enhance Communication.
- Streamline Appointment Management.
- Improve Patient Experience.
- Efficient Prescription Management.
- Implement AI for Disease Identification.
- Ensure Scalability and Performance.
- Admin Functionality.
- User Feedback and Reviews.

Hight level features/requirement to achieve goals/objectives:

- User Authentication and Authorization.
- Appointment Management.
- Real Time Communication.
- Medical Record Management.
- Payment Integration.
- Notification System.
- User Reviews and Ratings.
- Admin Dashboard.
- Data Security and Privacy.
- AI Integration.
- User Interface and Experience
- Reporting and Analytics.

Nonfunctional requirements:

- Performance.
- Availability.
- Security.
- Usability.
- Maintainability.
- Interoperability.
- Reliability.
- Support and Documentation.

4. Exploring and Engineering

A. Iterative development:

Time-Boxing

Table 2.1: Time boxing table for care pulse

No	Description of task	Time	Resource
I	Initial Study	20 days	User & Developer
2	Requirement Gathering and Analysis	15 days	Developer User & Admin
3	Design	15 days	Developer User & Admin
4	Implementation	20 days	Developer & User
5	Testing	22 days	Developer & User
6	Documentation	30 days	Developer & Analyst

Deployment:

Deploying the Care Pulse involves several critical steps to ensure a smooth and secure rollout. The backend, developed with NodeJS and Express, is containerized using Docker for consistent environments across development, testing, and production. PostgreSQL, the chosen database, is hosted on a managed service such as Amazon RDS to ensure reliability and scalability. Prisma, the ORM, manages database schema and migrations efficiently.

The real-time communication feature, powered by WebRTC via Agora.io, requires proper configuration of Agora's SDKs and API keys in the production environment. Frontend create with Node.js,Vercel, providing server-side rendering (SSR) for fast load times and improved SEO. These pipelines automate testing, building, and deployment processes, ensuring that any changes pass through rigorous testing before going live. Environment variables, secrets, and API keys are securely managed using services like AWS Secrets Manager or GitHub Secrets. For security, HTTPS is enforced across the application using SSL certificates, and JWT-based authentication ensures secure user sessions. Monitoring and logging are implemented using services like AWS CloudWatch or Loggly to track performance and identify issues. This comprehensive deployment strategy ensures that Care Pulse operates smoothly and securely, providing an optimal experience for all users..

2.2 Background of the Project:

With the increasing demand for streamlined healthcare services, this project aims to revolutionize the way patients, doctors, and administrators interact. Leveraging cutting edge technologies like NodeJS, Express, Prisma, WebRTC (Agora.io), and PostgreSQL, the system ensures secure and efficient data management. The platform offers real-time communication, seamless appointment scheduling, comprehensive patient profiles, and secure payment processing. It also integrates AI to aid in disease identification from prescribed medicines. By providing a single page application with server-side rendering using Next.js, the Care Pulse delivers an intuitive user experience. This project is poised to enhance healthcare delivery, improve patient outcomes, and simplify administrative tasks, making quality healthcare more accessible and efficient for everyone involved.

2.3 Problem areas:

Implementing the Care Pulse project can present several challenges across different aspects of the system. Identifying these potential problem areas early can help in devising strategies to mitigate them effectively.

1. Real-time Communication.
2. Data Security and Privacy.
3. System Integration.
4. User Experience.
5. AI Implementation.
6. Performance and Scalability.
7. Automated Notifications.
8. Regulatory Compliance.
9. Technical Debt and Maintenance.

2.4 Possible solution:

To address potential problem areas in the Care Pulse project, several solutions can be implemented.

1. **Scalability and Performance:** Use cloud-based infrastructure to ensure the system can handle increasing numbers of users and data efficiently.
2. **Security:** Implement robust authentication and authorization protocols to safeguard sensitive medical information.
3. **Real time Communication:** Leverage WebRTC with Agora.io for reliable and high-quality video and audio communication, while incorporating fallbacks and optimizations to ensure minimal latency and high availability.
4. **Appointment Scheduling:** Use an intuitive and responsive scheduling system with real-time updates and automated reminders to prevent conflicts and no-shows. **Data Management:** Employ Prisma with PostgreSQL for efficient data handling and backup solutions to prevent data loss.
5. **Payment Processing:** Integrate secure payment gateways and ensure compliance with payment security standards, such as PCI-DSS, to handle transactions smoothly.
6. **AI Integration:** Develop and test AI algorithms thoroughly for accuracy in disease identification, and ensure they are validated against real-world data.
7. **User Experience:** Design a user-friendly interface with accessibility features to cater to all users, including those with disabilities. By addressing these areas, the Care Pulse can deliver a reliable, secure, and efficient healthcare management solution.

Chapter 3 - Literature Review

A literature review involve offering a new interpretation of existing material, integrating new insights with established ones, or tracing the intellectual development of the field, including significant debates.[3]

3.1 Discussion on problem and solution domain based published articles:

The development of a healthcare management system like Care Pulse addresses several critical problems identified in the healthcare domain. One of the main issues is the inefficiency in communication and coordination between patients, doctors, and healthcare administrators. According to a study by Health Affairs, poor communication in healthcare settings can lead to medical errors, delayed treatment, and reduced patient satisfaction.[4] The traditional methods of managing appointments and medical records are often errors, as noted by a report from the National Center for Biotechnology Information (NCBI), which highlights that digitization in healthcare can significantly enhance accuracy and efficiency.[5]

The solution domain for these problems involves leveraging modern web technologies to create a integrated system for managing healthcare interactions. The use of NodeJS and Express for server-side development ensures a scalable and responsive backend. WebRTC via Agora.io enables real-time video consultations, which are essential for telemedicine area gaining significant especially post-COVID-19 pandemic as highlighted by the American Medical Association (AMA).[6] Prisma and PostgreSQL offer robust and secure database management, critical for handling sensitive medical data. Furthermore, Next.js enhances user experience through efficient rendering and interactivity, crucial for both patients and healthcare providers.

By integrating these technologies, Care Pulse aims to streamline appointment scheduling, medical record management, and real time communication. This comprehensive approach not only addresses current inefficiencies but also aligns with the ongoing digital transformation in healthcare, which is increasingly being seen as a necessity rather than an option.

3.2 Recommended approach:

In the Care Pulse system different methods are used. This discussion will examine three methods, describing their benefits and drawbacks helps to make informed decisions for the project.

Waterfall Model:

In the Waterfall Model, each phase must be fully completed before the next phase can start. This type of model is mainly used for small projects with clear and stable requirements.[7]

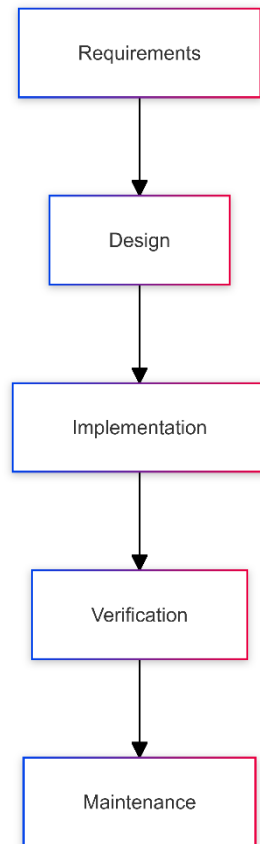


Figure 3.1: Waterfall model diagra

Agile Methodology:

Agile methodology is a set of principles through an iterative and incremental process. In software development, Agile emphasizes the importance of team collaboration and delivering a functional product quickly to meet customer needs and expectations.[8]

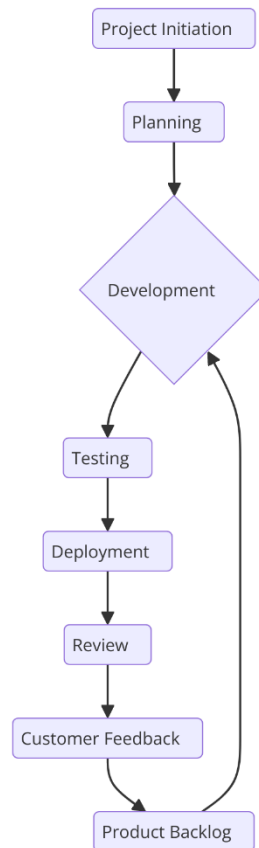


Figure 3.2: Agile model diagram

Dynamic System Development Methodology

It aimed to enhance the Rapid Application Development approach, which focused on quick prototyping and iterative changes based on user feedback.[9]

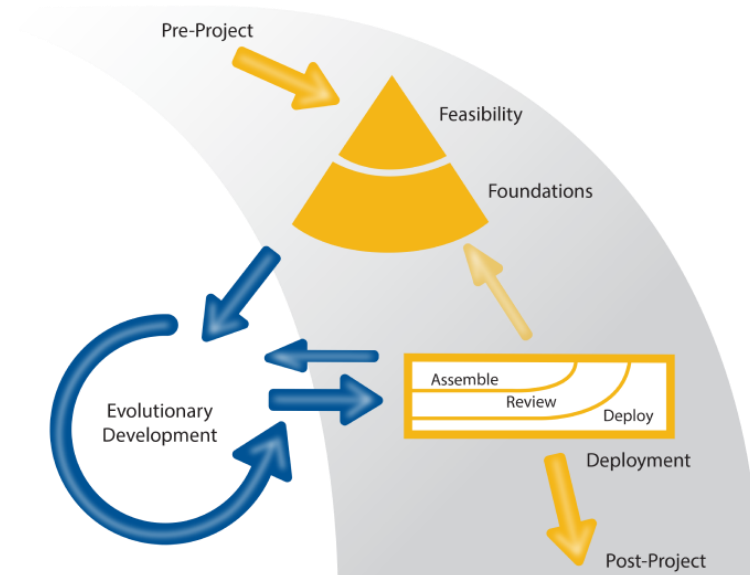


Figure 3.3: DSDM Process Methodology

The development of the Care Pulse system is part of my research project and faces challenges due to the need for a unified approach. Since not all aspects of the study were clearly defined from the beginning, my main tasks involve collecting data, analyzing it, and prioritizing development tasks. Working as the sole developer without a dedicated team, I find that combining elements from both the waterfall and DSDM models is effective. The waterfall model helps with detailed requirement gathering, while the DSDM model improves the efficiency of ongoing development. This combined approach allows for flexibility and responsiveness throughout the project, accommodating iterative changes and improvements as needed.

3.2 Why to Use:

The Care Pulse project addresses key challenges in healthcare management. The development process is designed to meet the system's needs effectively, supporting its overall goals and success. The chosen approach aligns well with these requirements, although specific implementation details are not yet available. Several factors influence this decision:

- The system facilitates smooth interaction between patients, doctors, and administrators.
- It includes features that promote user engagement.
- The system is capable of delivering the required functionalities.
- It provides timely support when needed.
- The project is designed to be completed within the planned timeframe.

3.4 Implementation Plan:

The implementation plan for the Care Pulse involves a phased approach starting with setting up the foundational technology stack, including NodeJS, Express, Prisma, PostgreSQL, and Next.js. The initial phase focuses on developing core functionalities for user authentication, appointment management, and real-time communication using WebRTC. Subsequently, the system will integrate comprehensive patient and doctor profiles, prescription management, and secure payment processing. AI capabilities for disease identification will be incorporated last. Continuous testing and user feedback will guide iterative improvements, ensuring seamless functionality and security. The final phase includes deployment, monitoring, and maintenance to ensure long-term performance and reliability.

Chapter 4 – Planning

4.1 Project Plan:

The Care Pulse project focuses important problems in healthcare by improving communication and coordination among patients, doctors, and administrators. Research shows that poor communication in healthcare can lead to medical errors, treatment delays, and lower patient satisfaction. Traditional methods of managing appointments and medical records are often inefficient and prone to mistakes. Moving to digital systems, as supported by various studies, improves both accuracy and efficiency.[10]

4.1.1 Work Breakdown Structure (WBS):

The work breakdown structure serves as a detailed guide for designing complex multi tiered emergency areas as part of their development process. It helps identify areas for improvement and outlines the steps needed to enhance team operations. Each level of work is carefully analyzed, moving from the main tasks to more detailed stages. By using the work breakdown structure, the team ensures the best possible outcomes. This framework integrates smoothly with the carefully planned breakdown structures.

Table 4.1: WBS for Care Pulse

No.	Task-description	Time/Day
01	Registration & login system.	8
02	UI Design for web	8
04	Customer able to search doctor based on their disease.	10
05	Online payments.	25
06	Booking and online receipts generated by the system.	20
07	Real time tracking for the doctors availability.	15
08	AI integretion	22

4.1.2 Time Duration/Time-Boxing:

It is a technique that involves allocating a fixed maximum unit of time to an activity in advance, and then completing the activity within that time frame.[11]

Table 4.2: Care Pulse System Execution

No	Description of task	Time/Days	Resource
1	Initial Study	20	User & Developer
2	Requirement Gathering and Analysis	15	Developer User & Admin
3	Design	15	Developer User & Admin
4	Implementation	20	Developer & User
5	Testing	22	Developer & User
6	Documentation	30	Developer & Analyst

4.1.3 Gantt Chart

By planning and showing all tasks in one place, the chart helps teams stay organized and meet deadlines.[12]

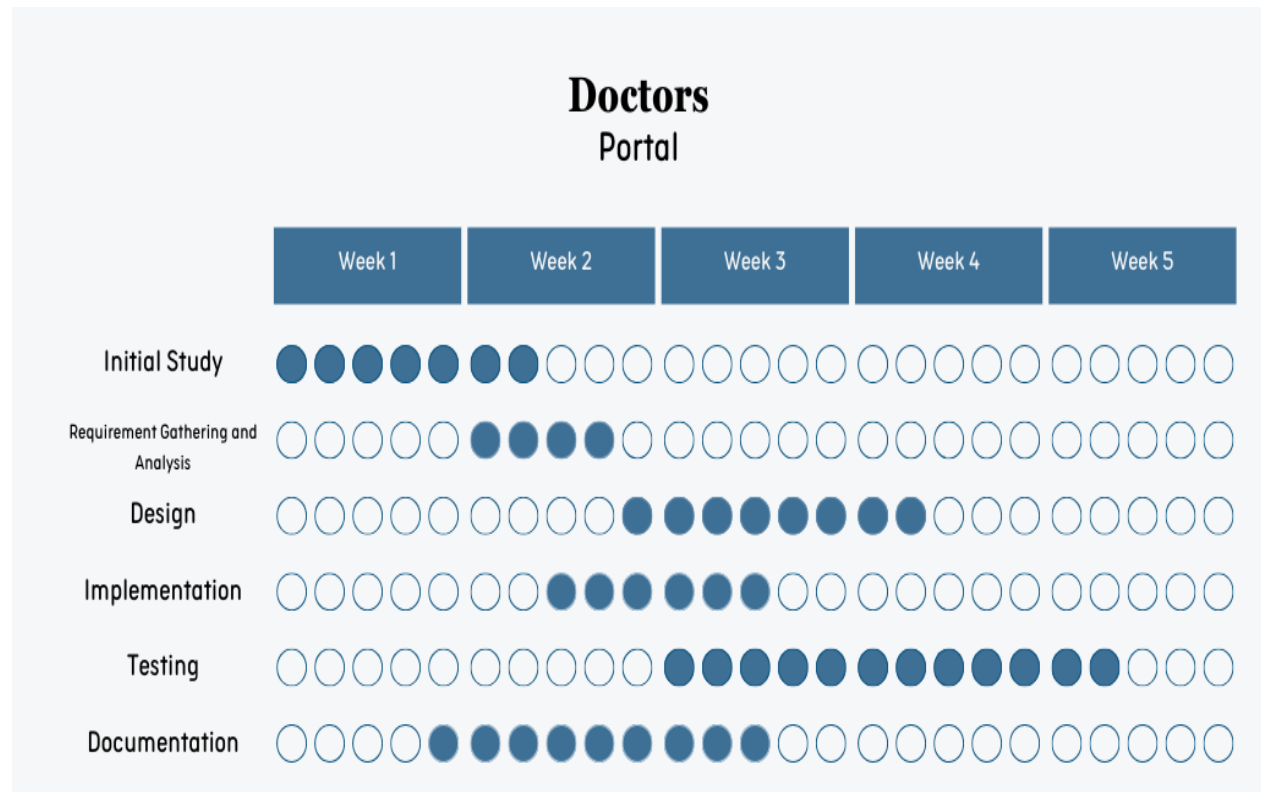


Fig 4.1: Gant chart-Care Pulse

4.2 Test plan:

Care Pulse project will ensure that all system components function correctly and meet user requirements. Functional tests will cover user roles admin, doctor, and patient—to verify account management, appointment scheduling, prescription generation, and payment processing. Security tests will focus on authentication, authorization, and data protection. Performance testing will assess system responsiveness and scalability under varying loads. Real-time communication capabilities via WebRTC will be tested for reliability and latency.

4.2.2 Required Test:

Functional Testing:

- **Integration Testing:** This phase is essential in software systems, especially for an organization with official software certification. Integration testing involves testing multiple components together, incorporating necessary features or groups to ensure smooth functionality. Verification processes are used to confirm that all software modules operate seamlessly before proceeding with final test executions.
- **Unit Testing:** This phase involves detailed testing of individual segments and elements of the project to ensure their proper functioning and accuracy. Unit testing is crucial for verifying regulatory compliance and supporting development activities.
- **System Testing:** Following integration testing, system testing focuses on a comprehensive evaluation of the system's capabilities, features, and requirements. This phase highlights the importance of effective workload management.
- **Acceptance Testing:** The final phase of testing, acceptance testing, involves direct user participation. Its goal is to evaluate user satisfaction and ensure that the system meets essential business requirements.

Nonfunctional Testing:

- **Security Testing:** This phase focuses on evaluating the security strength of the Care Pulse, both from an internal and external perspective. It includes identifying vulnerabilities, such as SQL injection, to fortify the system against potential threats. Thorough assessments of both internal and external security measures are performed to mitigate risks and safeguard sensitive data.
- **Usability Testing:** This phase involves assessing the usability of the Care Pulse to ensure it is user-friendly and easy to navigate. The goal is to enhance the overall user experience by focusing on convenience and effectiveness, making sure that users can interact with the system efficiently.
- **Reliability Testing:** This phase involves testing the software's durability and resilience. It aims to evaluate the Doctors Care Pulse's ability to perform reliably and consistently under various conditions, ensuring the system's robustness and dependability.

Chapter-5 Feasibility

A feasibility study is an evaluation that assesses the potential success of a proposed project, such as launching a new product line or implementing a technical system. It examines key aspects like technical, economic, and legal factors to determine whether the project is a sound investment. Additionally, the study helps identify possible challenges and issues that could emerge during the project's execution.

5.1 Economic Feasibility

The Creative Crafting web application system incorporates AI-based image generation, an authentication system, admin panel, buyer and seller account types, transaction management, payment processing, profile management, online receipt and payment summaries, and features like cart management and wishlists. The system's deployment is straightforward, with additional costs like installation dependent on specific needs. Furthermore, I provide free system maintenance for six months post-implementation, subject to any changes or new requirements that may affect service continuity. To use the service, acquiring a server and domain name is essential.

Table 5.1: Economic Feasibility

Components	Price
Software	30,000/=
Hardware	80,000/=
Hosting	15,000/=
Others	9,000/=
Total	134,000/=

5.2 Technological Feasibility

The technical feasibility of the Care Pulse project is given the robust and modern technologies proposed for its development. The use of NodeJS and Express ensures efficient server-side performance, while Next.js, a React framework, enables a responsive and dynamic user interface. Prisma, as the ORM tool, provides seamless interaction with PostgreSQL, a reliable and scalable database system that can handle complex queries and large volumes of medical records. Integration with WebRTC through Agora.io facilitates secure and real-time communication between doctors and patients, which is critical for remote consultations. The chosen technologies are well-supported, widely adopted, and have strong community backing, reducing the risk of technical challenges. Additionally, the implementation of AI for disease identification from prescribed medicines leverages cutting-edge innovation to enhance diagnostic accuracy, further solidifying the technical foundation of the Care Pulse. Overall, the technical infrastructure is well-suited to meet the project's objectives effectively and securely.

5.3 Operational Feasibility

The operational feasibility of the Care Pulse centers on its ability to seamlessly integrate into the existing healthcare workflows while enhancing efficiency and user experience. The platform is designed to support the day-to-day operations of administrators, doctors, and patients by streamlining appointment scheduling, medical record management, and real-time communication. Given the intuitive interface, role-specific functionalities, and secure data handling, the portal is well-suited to meet the operational needs of healthcare providers. The use of cutting-edge technologies like WebRTC for real-time communication and Prisma for database management ensures smooth operation with minimal downtime. The system's ability to handle critical tasks such as prescription management and secure payment processing highlights its readiness for real-world application. Overall, the Care Pulse is operationally feasible, with the capacity to significantly improve healthcare service delivery without disrupting current practices.

5.4 Cost Benefit Analysis

This data highlights the inherent instability in the current industrial landscape. At present, each online task takes an entire day to complete. Implementing this system is expected to offer substantial advantages to users.

Table 5.2: Cost Benefit Analysis

No.	Components	Price 01 year	Price 02 year	Price 03 year	Price 04 year	Total
1	Software	1500/=	1500/=	1500/=	-	4500/=
2	Hardware	30000/=	-	30000/=	-	60,000/=
3	Hosting	6000/=	6000/=	6000/=	6000/=	24,000/=
4	Office	80000/=	-	-	-	80000/=
5	Staff	7000/=	7000/=	7000/=	70000/=	280000/=
6	Development	8000/=	8000/=	8000/=	8000/=	32000/=
	Total	132,500/=	52,500/=	22,500/=	21,000	228,500/=

Total revenue:

Table 5.3: Total revenue

Total Earn	132,500/=	52,500/=	22,500/=	21,000/=	228,500/=
Total Expense	30,400/=	30,400/=	30,400/=	30,400/=	121,600/=
Total Revenu	162,900	82,900/=	52900/=	51,400/=	350,100/=

After completing a detailed cost analysis, the revenue is projected to increase to 350100 /=. This shows, I am confident that this program has significant profitability

Chapter 6 – Foundation

6.1 Over All Requirement List:

There is two main types of requirements, one functional requirements and two non functional requirements.

6.1.1 List of functional requirements:

- User Registration and Authentication
- Appointment Scheduling
- Patient Profile Management
- Prescription Management
- Payment Processing:
- Real-time Communication
- Review and Rating System
- Notification System
- Admin Management

6.1.2 List of Not Functional Requirement:

- Performance
- Security
- Scalability
- Reliability
- Usability
- Maintainability
- Data Integrity
- Compliance
- Auditability

6.2 What Technology Implemented:

The Care Pulse utilizes a range of modern technologies to create a comprehensive healthcare management system. At its core, NodeJS and Express are used for server-side development, providing a robust and scalable backend. Prisma, an Object-Relational Mapping (ORM) tool, simplifies database management with PostgreSQL, ensuring efficient data handling and retrieval. For real-time communication between doctors and patients, the system integrates WebRTC through Agora.io, facilitating seamless video consultations. The front-end is built with Next.js, a React framework, enabling a dynamic and responsive user interface. Together, these technologies ensure the Care Pulse is secure, user-friendly, and capable of handling various healthcare management tasks, from scheduling and managing appointments to generating and delivering prescriptions. This tech stack supports a streamlined, efficient, and interactive experience for users across different roles in the healthcare ecosystem.

Key Features:

- Real Time Communication
- User Authentication
- Appointment Scheduling
- Patient Profiles
- Prescription Management
- Payment Integration
- Automated Notifications
- Doctor Profiles
- Medical Record Management
- Review System
- Admin Controls
- Single-Page Application (SSR)

6.3 Recommendations and Justifications:

The Care Pulse web system is highly recommended for its ability to streamline healthcare management through its comprehensive suite of features. It effectively integrates technologies like NodeJS, Express, Prisma, WebRTC (via Agora.io), and PostgreSQL, ensuring robust performance and real-time communication capabilities. The system supports multiple user roles, including admins, doctors, and patients, each with tailored functionalities that enhance the overall efficiency of healthcare operations. For instance, real-time video consultations through WebRTC improve accessibility and convenience, while Prisma and PostgreSQL facilitate secure and efficient data management. The system's focus on secure user authentication, automated appointment scheduling, and seamless payment processing addresses critical needs in healthcare management. Additionally, features like prescription management and automated email notifications enhance user experience by providing timely and accurate information. These functionalities not only ensure the security and reliability of the system but also contribute to a smoother and more effective healthcare experience for all users involved. Overall, the Care Pulse offers a well-rounded solution that addresses key challenges in healthcare management, justifying its implementation as a robust and efficient system.

Chapter 7-Explorations

Creative Crafting involves making handmade items using diverse materials and techniques. This practice nurtures creativity, relaxation, and personal expression. It encompasses activities like knitting, sewing, painting, and more, offering a fulfilling way to produce unique, personalized items, often pursued as a hobby or a small business venture.

7.1 Old System Use Cases:

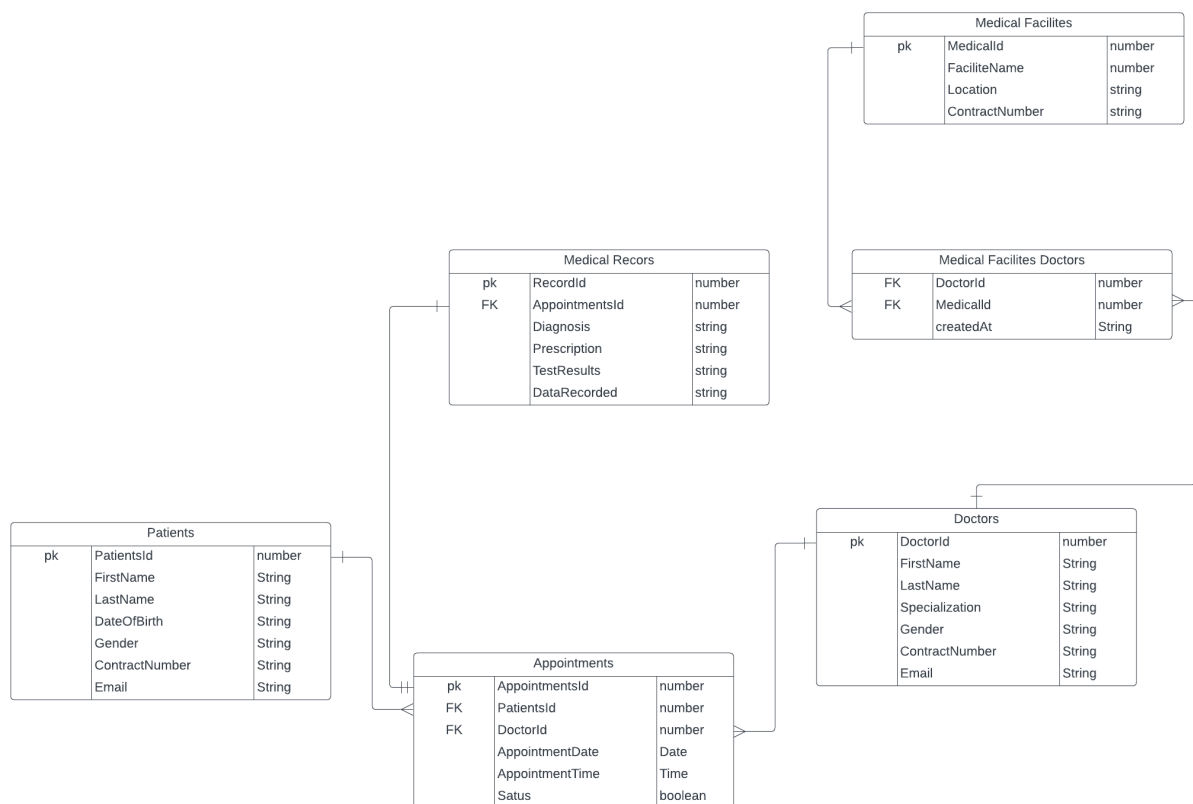


Fig 7.1: Usecase diagram of Care Pulse

7.2 Old system activity diagram:

Admin activity diagram:

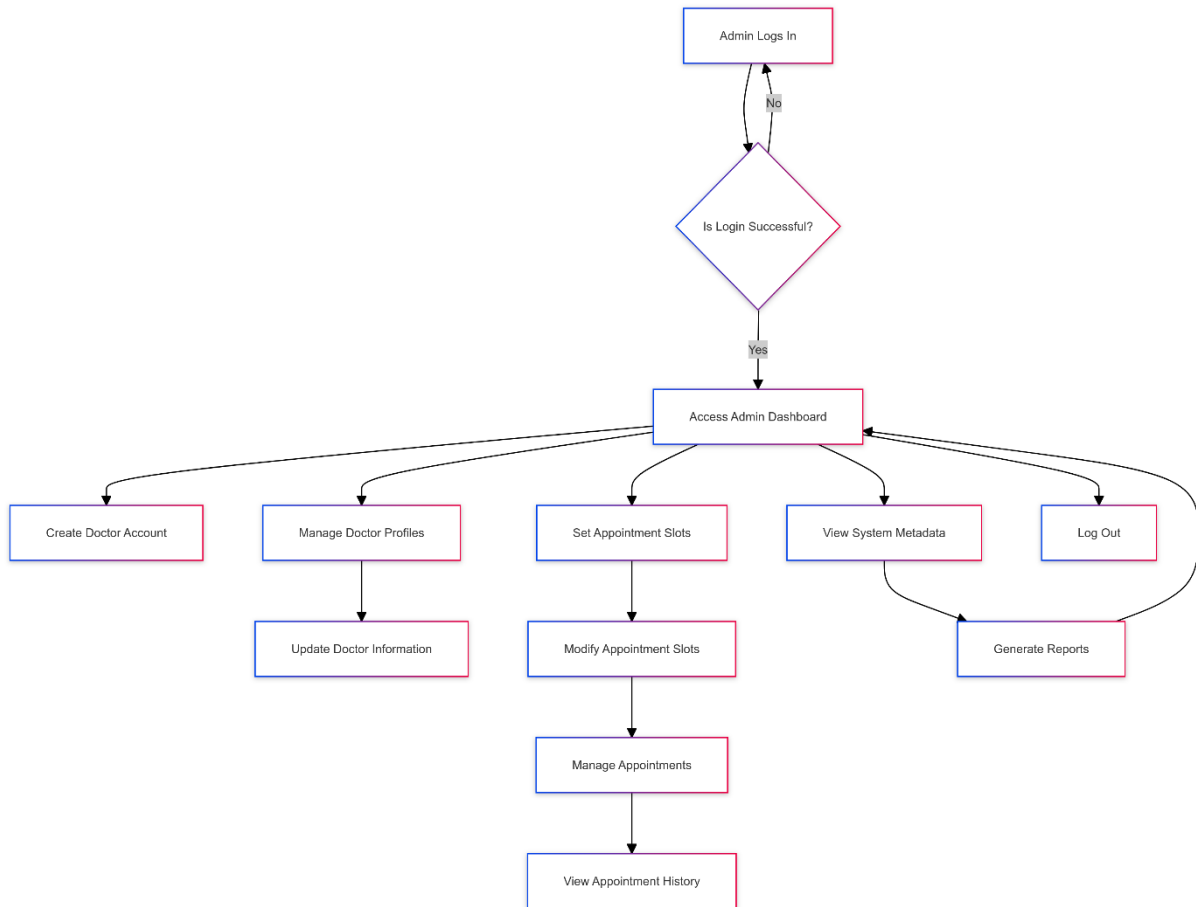


Fig 7.2: Admin-activity diagram of care pulse

User activity diagram:

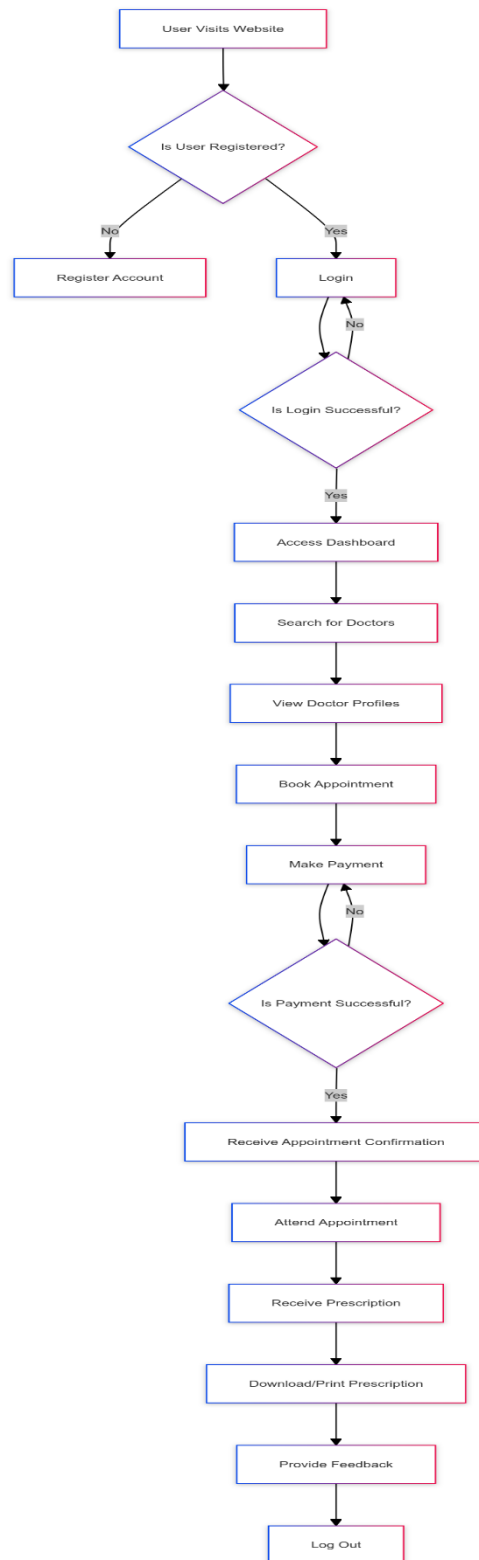


Fig 7.3: User-activity diagram of care pulse

7.3 Prioritization Requirement List (PRL):

Applying Moscow:

The Moscow rule is fundamental to the requirements list, and. This approach helps in prioritizing functionality and efficiently understanding requirements. Application Process

Moscow stands for Mandatory, Recommended, Optional and Optional features. Below are the characteristics of Moscow.[13]

Table 7.1: Prioritized-Requirement List of Care Pulse

Requirement No	Requirement Name	Priority
01	Implement WebRTC for video/voice calls between doctors and patients.	High
02	Develop a robust system for secure login and account management.	High
03	Create a system for booking, rescheduling, and managing appointments.	High
04	Build detailed patient profiles with medical history, diagnostic reports, and prescriptions	High
05	Enable doctors to generate, send, and manage electronic prescriptions.	High
06	Implement a secure system for processing consultation fees and transactions.	High
07	Provide tools for admins to manage doctor accounts, appointments, and view system metadata.	Medium
08	Show detailed information about doctors, including specializations and availability.	Low
09	Allow patients to leave reviews and ratings for doctors.	Low

7.4 Prototype of the System:

Prototyping is a strategic approach designed to optimize website development by focusing on essential marketing elements.[13] Below, you will find the prototype for the new system:

Doctors Portal

Home
Appointment
OnSite Appointment
About
Dashboard
Sign out

All users

Doctor Varify

Add A Doctor

Manage Doctors

Create Appointment

All Appointment

Health Checked

Review

My Profile

Activity

Chnage Password

Reset Password

Health Report

Add To Doctor

Name

Medical College

Specialty

Teeth Orthodontics

Email

Registered Doctor License

Gender

Male

Appointment Fee

Designation

District Name

Comilla | কুমিল্লা

Qualification

Photo

Choose File No file chosen

Password

Doctor Phone Number

Experience

Current Working Place

Chamber Address

Fig 7.4: Doctor selection prototype

Doctors Portal

Home
Appointment
OnSite Appointment
About
Dashboard
Sign out

Online Appointment

Onsite Appointment

My Health

Video Call

My Profile

Medical History

Activity

Health Checked

Review

Chnage Password

Reset Password

My Review Section

Search by comment, Patients mail, or doctor's email

good experiences

Rating: 5

Doctor: kaspia@gmail.com

Average Rating: No rating available

Patient: ali@gmail.com

Appointment Id: 66710dadb20bc9a588cac1e0

Date: 7/16/2024

DELETE

the quick

Rating: 4

Doctor: kaspia@gmail.com

Average Rating: No rating available

Patient: ali@gmail.com

Appointment Id: 66710dbcb20bc9a588cac1e1

Date: 7/16/2024

DELETE

Fig 7.5: Feedback prototype

© Daffodil International University

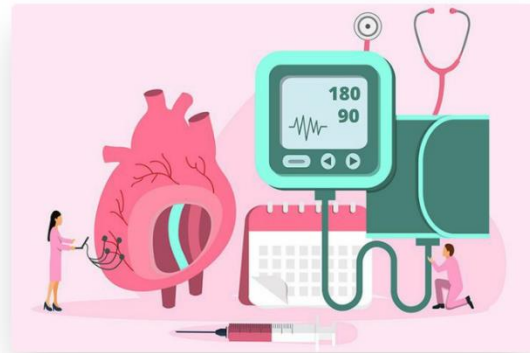
30



Care Pulse Office News!

Provident cupiditate voluptatem et in. Quaerat fugiat ut assumenda excepturi exercitationem quasi. In deleniti eaque aut repudiandae et a id nisi.

GETTING STARTED



Opening Hours

Open 9.00 am to 5.00pm everyday



Our Locations

Open 9.00 am to 5.00pm everyday



Contact Us

Open 9.00 am to 5.00pm everyday

Fig 7.6: Homepage care pulse

Chapter 8 - Engineering

8.1 New System Modules:

The proposed creative crafting web application system offers a comprehensive suite meticulously crafted to address user needs effectively. These modules are designed to cater to a wide range of requirements, ensuring versatility and flexibility in usage. Below, I outline the primary trends and models of the proposed system, showcasing its standout features and unique capabilities.

8.2 New System Use Case diagram

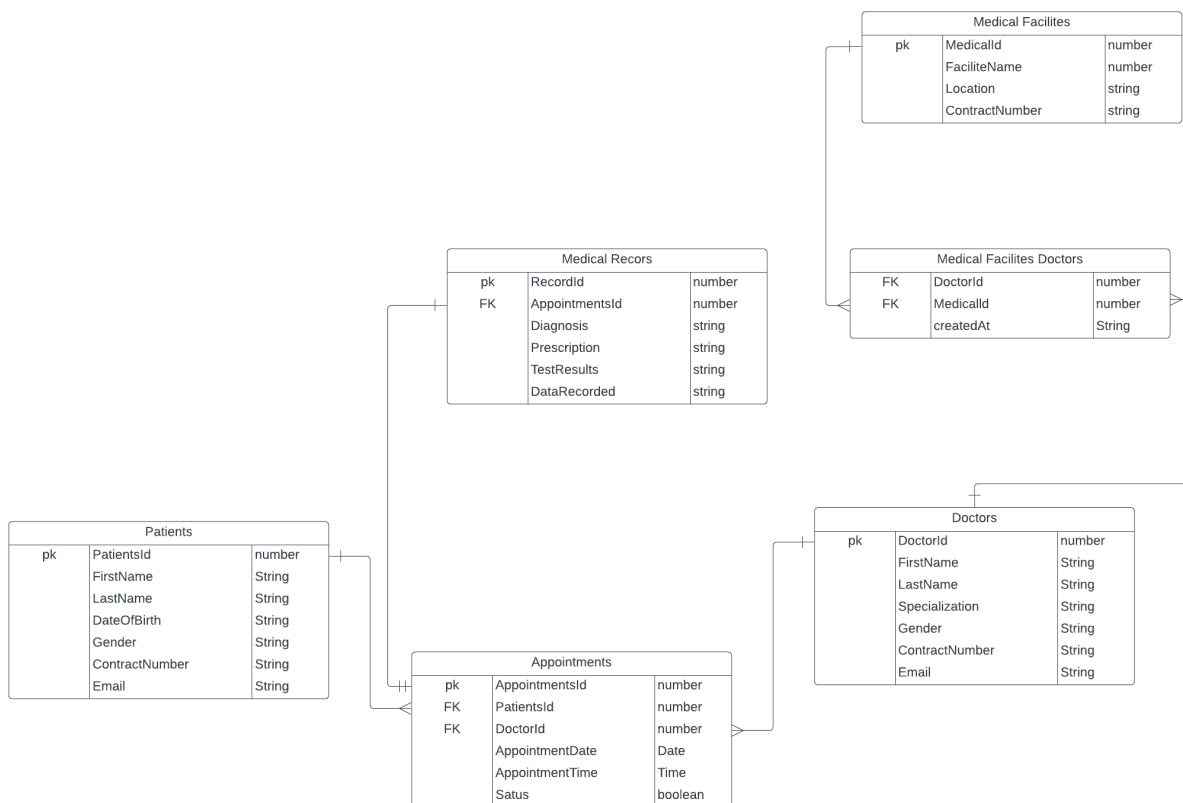


Fig 8.1: Newsystem Usecase diagram

8.3 New system ERD Diagram:

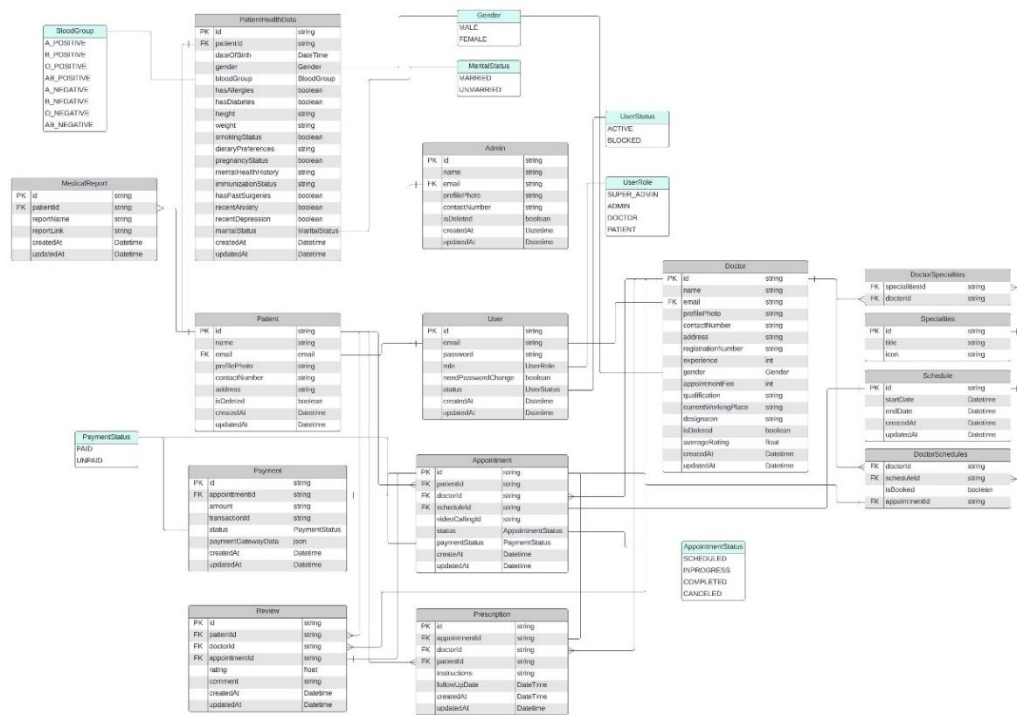


Fig 8.2: ERD-Diagram

8.4 System Interface Design/Prototype:

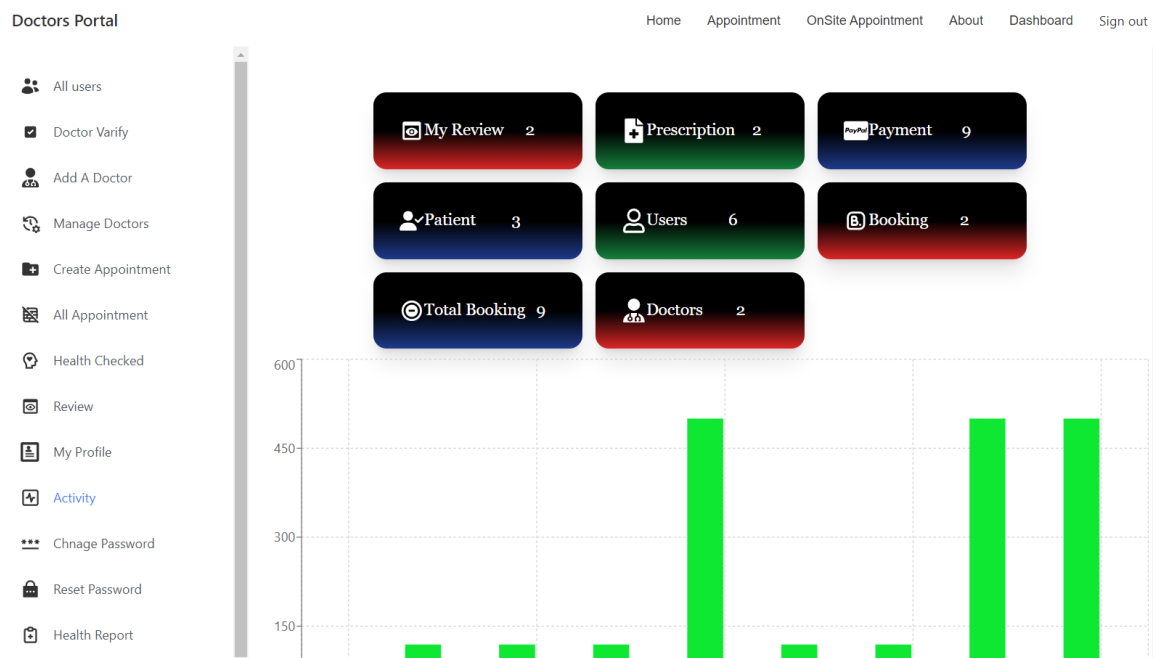


Fig 8.3: Monitoring part



Fig 8.4: Time schedule

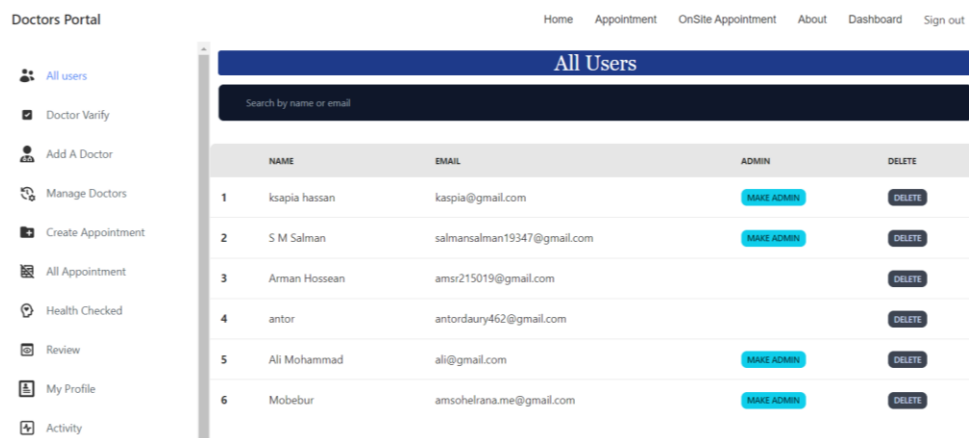


Fig 8.5: User profile

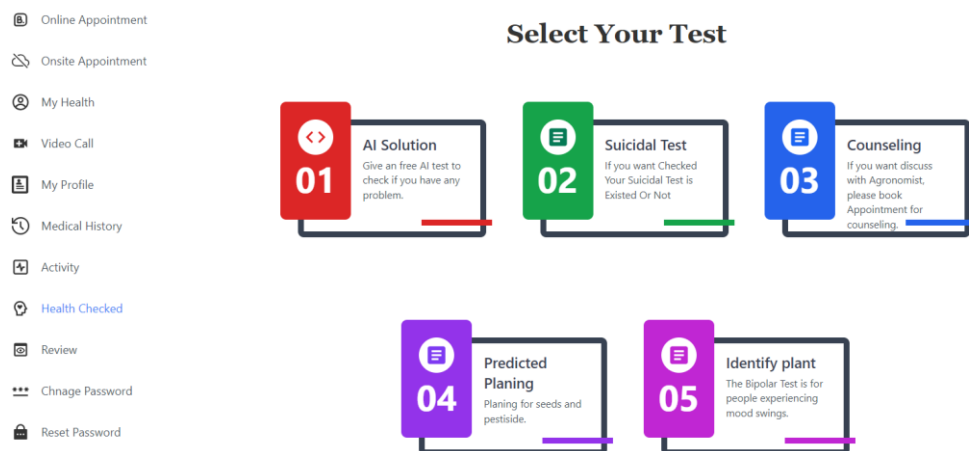


Fig 8.6: Test selection

Chapter 9 - Deployment/Development

9.1 Core Modules Coding Simples:

At this stage of the Care Pulse project, implementation and construction began after completing all the necessary analyses. Breaking down tasks into smaller, manageable segments made the development process much smoother. I utilized a variety of programming languages and tools to build the recommended framework, including Node.js, Express.js, Prisma, Next.js, Tailwind CSS, Agora.io for WebRTC, PostgreSQL, and JavaScript. Herer are some examples of the code:

```
~ import React from "react";

import chair from "../../assets/images/bannerImage.jpeg";
import { DatePicker } from "react-day-picker";

~ const AppointmentBanner = ({ selectedDate, setSelectedDate }) => {
~   return (
~     <header className="my-6">
~       <div className="hero">
~         <div className="hero-content flex-col lg:flex-row-reverse">
~           <img
~             src={chair}
~             alt="dentist chair"
~             className="max-w-sm rounded-lg shadow-2xl"
~           />
~           <div className="mr-6 sm:w-full">
~             <DatePicker
~               mode="single"
~               selected={selectedDate}
~               onSelect={setSelectedDate}
~             />
~           </div>
~         </div>
~       </div>
~     </header>
~   )
~ }
```

Fig 9.1: Banner Code

```

const handleAddDoctor = (data) => {
  const image = data.image[0];
  const formData = new FormData();
  formData.append("image", image);
  const url = `https://api.imgbb.com/1/upload?key=${imageHostKey}`;
  fetch(url, {
    method: "POST",
    body: formData,
  })
    .then((res) => res.json())
    .then((imgData) => {
      if (imgData.success) {
        data.image = imgData.data.url;
        data.experience = Number(data.experience);
        data.appointmentfee = Number(data.appointmentfee);
        setSignUPError("");
        createUser(data.email, data.password)
          .then((result) => {
            const user = result.user;
            const userInfo = {
              displayName: data.name.

```

Fig 9.2: ImageApi code

```

app.delete(`/api/v1/deleteAccount`, verifyJWT, async (req, res) => {
  const email = req.decoded.email;
  // start transaction rollback
  const session = client.startSession();
  try {
    session.startTransaction();
    const result = await usersCollection.deleteOne({ email }, { session });
    if (!result) {
      throw new Error("User Collection Session Failed");
    }
    const patientResult = await patientCollection.deleteOne(
      { email },
      { session }
    );
    if (!patientResult) {
      throw new Error("Patient Collection Session Failed");
    }

    await session.commitTransaction();
    await session.endSession();
  }

```

Fig 9.3: Transaction Code

```

async function run() {
  app.put("/api/v1/isOnsiteBookingSloat/:id", verifyJWT, async (req, res) => {
    const email = req.decoded.email;
    const { id } = req.params;
    const data = req.body;

    // checked is it patients Account
    const isPatient = await usersCollection.findOne(
      { email },
      { projection: { role: 1 } }
    );

    if (
      isPatient.role === process.env.USER_ROLE &&
      isPatient.role === process.env.USER_ROLE_ADMIN
    ) {
      return res.send({
        status: httpStatus.UNAUTHORIZED,
        message: "Only Patient Can Be Booking",
      });
    }
  });
}

```

Fig 9.4: Patient-account Code

```

import toast from "react-hot-toast";
const PutAction = (url, selectedSpecialties, refetch) => {
  fetch(url, {
    method: "PUT",
    headers: {
      "content-type": "application/json",
      authorization: `bearer ${localStorage.getItem("accessToken")}`,
    },
    body: JSON.stringify(selectedSpecialties),
  })
  .then((res) => {
    if (!res.ok) {
      throw new Error("API ERROR");
    }
    return res.json();
  })
  .then((data) => {
    toast.success(data?.message);
    refetch();
  })
  .catch((error) => {
    toast.error(error?.message);
  });
}

```

Fig 9.5: Authentication Code

9.2 Possible problem breakdown:

The system I propose boasts a number of notable features. Decomposing the system is essential for smooth development. Failure to do so will significantly hamper the progress of the entire system.

The Creative Crafting web application system integrates AI-based image generation, enhancing user creativity by providing unique and personalized design suggestions. The system has a robust authentication system, ensuring secure access for all users. An intuitive admin panel is provided to efficiently manage content, users, and transactions. The platform supports both buyer and seller account types, facilitating a seamless marketplace experience. The transaction system is designed for safe and efficient payments, supported by a comprehensive payment system that accommodates various methods. Users can update their profiles, view online receipts or payment summaries, and manage their activities easily. Additionally, the system features 'Add to Cart' and 'Wish List' functionalities to enhance the shopping experience, allowing users to save items for future purchases and conveniently track their desired products.

Chapter 10 - Testing

10.1 Unit Testing:

The unit testing process is essential. Individual assessments ensure the project code functions correctly in this context.[14]

Table 10.1: Testing-unit

Integration testing	Test Class	Designed by Antor Daury		
Data source:	Functionality Test	Tested By Antor Daury		
No.	Test name	Task name	Exxpected Output	Final Out Put
1.1	Product Testing	Expect Data: From fill	Successfull	Successfull

Login

Email

amshr215019@gmail.com

Password

.....

Forget Password?

LOGIN

New to Doctors Portal [Create new Account](#)

OR

CONTINUE WITH GOOGLE

Fig 10.1: Login System-testing

10.2 Module testing:

Module testing refers to protocols, standards, classes, or software types used to rigorously examine systems and applications. It is advisable to concentrate on testing the internal components of the system on a smaller scale, rather than attempting to test all applications concurrently.[15]

Table 10.2: Module-Testing

Integration testing	Test Class	Designed by Antor Daury		
Data source:	Object Test Functionality	Tested By Antor Daury		
No.	Test name	Task name	Output	Final Out Put

1.1	Registration testing	Expect Data: From filling	Successfully creates Account	Successfully creates Account
-----	----------------------	------------------------------------	------------------------------------	------------------------------------

Login

Email

amsr215019@gmail.com

Password

.....

Forget Password?

LOGIN

Fig 10.2:Module-testing

10.3 Integration Testing

Integrated testing checks the functionality of individual software units in a system. Its purpose of identifying design-related issues emphasizes the relevance of the archive integration test.[16]

Personal Details
Please fill out all the fields.

Full Name

Email Address

Address / Street City

Country / region State / province Account Types

Bangladesh Dhaka Seller Account

Browse... pro.jpg

Password

Confirm Password

☒ My billing address is different than above.

SUBMIT

Fig 10.3: Integretion-testing

Chapter 11-Implementation

11.1 Training:

Training is an important aspect of development. Training system users is essential for staying up to date on the newest system updates and user instructions. After training, users will find it simple to navigate and comprehend the system.[17]

Table 11.1: Training table of Care Pulse

NO	User	Training Sect.	Time Duration	Note
01	Admin	Account	3 hr	User Application
02		User Activity	45 min	Activity of user
03		User Dashboard	1 hr	Dashboard Information
06	Doctor/Paitient	Create prescription with AI	30 min	Ai prescribe generation system
		Video calling	10 min	Doctor patient communication
		Used with UI	1.3 hours	Adaptability

11.2 Big bang:

Its primary role is the implementation of a new system while retiring an existing one. The main course of action is to complete testing and then install the new system. However, moving data from the old system to the new platform may suffer delays and inherent dangers such as potential data loss or corruption, particularly in terms of page management.[18]

11.3 Recommended implementation process:

The technology is now being deployed across different organizational departments as part of a pilot project. I believe it is possible to construct a successful pilot phase in this scenario. Finally, the system will be fully operational after all features work as expected.

Chapter 12 - Critical Appraise and Evaluation

It highlights important requirements that weren't met and explains why. It also notes that more work and possibly some guidance are needed to meet these requirements.[19]

12.1 Objective Cloud be Met:

Here are some additional objectives that could be met in the Care Pulse web system to enhance its functionality and user experience:

- **Advanced analytics and reporting:** Provide detailed analytics and reporting tools for doctors and administrators to track appointment trends, patient outcomes, and system performance.
- **Customizable notifications:** Allow users to customize notification preferences for appointment reminders, prescription updates, and other important alerts.
- **Multilingual support:** Include support for multiple languages to cater to a diverse user base and improve accessibility.
- **Patient education resources:** Provide access to educational materials and resources related to health and wellness for patients.
- **Advanced search and filtering:** Enhance search and filtering options for patients to find doctors based on specialty, location, availability, and other criteria.
- **Patient and doctor forums:** Create forums or discussion boards for patients and doctors to share information, ask questions, and discuss health-related topics.

12.1.2 How much better could have been done:

- Enhanced User Experience
- Performance Optimization
- Security Enhancements
- Comprehensive Reporting
- Feedback and Support

12.1.3 How much better could be the features of Solution ?

Care Pulse excels with its advanced features, offering seamless real-time communication through WebRTC for consultations.[20] It ensures secure and efficient appointment management and patient data handling. The platform integrates a robust authentication system, automated email notifications for appointments and prescriptions, and a secure payment gateway. Doctors can manage detailed patient profiles and generate prescriptions easily, while patients benefit from streamlined appointment booking, medical record management, and access to prescriptions. The system's comprehensive admin panel allows efficient management of doctor accounts and appointment slots, ensuring a smooth and effective healthcare management experience for all users.

12.1.4 Which Features Could Not be Touch

- Real time communication
- Secure user authentication:
- Efficient appointment management
- Comprehensive patient profiles
- Effective prescription management
- Secure payment system
- Automated notifications

Chapter 13- Conclusion

13.1 Summary of the Project

The Care Pulse project is a comprehensive web-based healthcare management system designed to enhance communication and streamline processes among patients, doctors, and administrators. Utilizing advanced technologies such as NodeJS, Express, Prisma, WebRTC (via Agora.io), and PostgreSQL, the system offers a robust platform for managing appointments, medical records, and prescriptions. It supports three primary user roles: admins, who manage doctor accounts and appointment schedules; doctors, who oversee patient profiles and generate prescriptions; and patients, who can book appointments, access medical records, and receive prescriptions. Key features include real-time video consultations, secure user authentication, automated payment processing, and email notifications. The system ensures user-friendly interactions and reliable performance through detailed usability and reliability testing. By integrating these elements, the Care Pulse aims to improve healthcare management efficiency, enhance patient care for all users and this project addresses the need for secure, efficient, and accessible healthcare services, making it a valuable tool in modern healthcare systems.

13.2 Goal of the Project

Care Pulse project main goal is, create an comprehensive, user-friendly healthcare management system that streamlines interactions between patients, doctors, and administrators. By integrating advanced technologies such as NodeJS, Express, Prisma, WebRTC, and PostgreSQL, the project aims to enhance the efficiency of appointment scheduling, real-time consultations, and medical record management. The system seeks to improve the accessibility and convenience of healthcare services through secure communication, seamless appointment management, and automated processes, ultimately providing a more effective and organized approach to healthcare delivery.

13.3 My experience

The completion of the Care Pulse web application project has been a highly enriching experience, offering me valuable insights into information gathering, problem-solving, and overcoming technical challenges. Initially, the project's complex technical requirements were demanding, but they provided an excellent platform for honing my skills. The Care Pulse features a range of advanced functionalities, including real-time communication via WebRTC, a secure authentication system, a comprehensive admin panel, appointment and prescription management, detailed patient profiles, and a streamlined payment system. Effective time and cost management were crucial to delivering a successful solution.

I am confident that the expertise and knowledge gained from this project will greatly enhance my future endeavors.

References

- [1]. Research and Market Viability - The Office of Learning Technologies. (n.d.).
<https://www.uab.edu/elearning/services/market-viability>
- [2]. Economic feasibility. (n.d.). CEOPedia | Management Online.
https://ceopedia.org/index.php/Economic_feasibility
- [3]. Academic writing: What is a literature review? | SFU Library. (n.d.).
<https://www.lib.sfu.ca/about/branches-depts/slc/writing/assignments/lit-review>
- [4]. Howick, J., Bennett-Weston, A., Solomon, J., Nockels, K., Bostock, J., & Keshtkar, L. (2024). How does communication affect patient safety? Protocol for a systematic review and logic model. *BMJ Open*, 14(5), e085312. <https://doi.org/10.1136/bmjopen-2024-085312>
- [5]. Abernethy, A., Adams, L., Barrett, M., Bechtel, C., Brennan, P., Butte, A., Faulkner, J., Fontaine, E., Friedhoff, S., Halamka, J., Howell, M., Johnson, K., Long, P., McGraw, D., Miller, R., Lee, P., Perlin, J., Rucker, D., Sandy, L., . . . Valdes, K. (2022). The Promise of Digital Health: Then, Now, and the Future. *NAM Perspectives*, 6(22). <https://doi.org/10.31478/202206e>
- [6]. Shaver, J. (2022). The State of Telehealth Before and After the COVID-19 Pandemic. *Primary Care Clinics in Office Practice*, 49(4), 517–530. <https://doi.org/10.1016/j.pop.2022.04.002>
- [7] Team, A. (2024, July 22). Waterfall Model: What Is It, When and How To Use It? actiTIME - Time Tracking Software. <https://www.actitime.com/project-management/what-is-waterfall-model>
- [8] Understand Agile methodology | Definition and overview | Sumo Logic. (2024, March 13). Sumo Logic. <https://www.sumologic.com/glossary/agile-methodology/>
- [9] Dynamic Systems Development Method (DSDM). (2022, February 11). ProductPlan. <https://www.productplan.com/glossary/dynamic-systems-development-method/>
- [10] Rudder, A. (2024, June 12). What Is A Project Management Plan And How To Create One. *Forbes Advisor*. <https://www.forbes.com/advisor/business/project-management-plan/>
- [11] Liew, B. (2024, July 15). The Timeboxing Technique: a Comprehensive Guide to Get Started Right Away! <https://www.usemotion.com/blog/timeboxing-technique/>
- [12] Martins, J. (2024, January 23). What is a Gantt Chart? Guide to Project Timelines [2024] • Asana. Asana. <https://asana.com/resources/gantt-chart-basics>

- [13] What is MoSCoW Prioritisation & How to Use it? | Zeda.io. (n.d.). <https://zedai.io/blog/what-is-moscow-prioritisation-how-to-use-it>
- [14] Team, C. O. (2024, August 1). Unit Testing. Codefresh. <https://codefresh.io/learn/unit-testing/>
- [15] Module Tests. (2023, July 13). IONOS Digital Guide. <https://www.ionos.com/digitalguide/websites/web-development/module-tests-in-software-development/?srltid=AfmBOoo7YwiLtKN6hGhvaRRgJ3IyoYxEcDeopx5x4TjDi9aIdftphgf0>
- [16] Sufi, A. G. (2024, March 21). Understanding Integration Tests: Scoping Strategies, and Incremental Testing Techniques. Medium. <https://ahmadgsufi.medium.com/understanding-integration-tests-scoping-strategies-and-incremental-testing-techniques-6038d9fddacc>
- [17] Multiple Release Hardware. (n.d.). In Phase 8: Implementation (pp. 1–12). <https://doit.maryland.gov/SDLC/Documents/SDLC%20Phase%2008%20Implementation%20Phase%20Multiple%20Hardware.pdf>
- [18] Pylypenko, R. (2024b, February 12). 7 Phases of the System Development Life Cycle | Intellectsoft. Intellectsoft Blog. [https://www.intellectsoft.net/blog/what-is-system-development-life-cycle/#:~:text=The%20system%20development%20life%20cycle%20\(SDLC\)%20is%20a%20complex%20project,testing%2C%20implementation%2C%20and%20maintenance.](https://www.intellectsoft.net/blog/what-is-system-development-life-cycle/#:~:text=The%20system%20development%20life%20cycle%20(SDLC)%20is%20a%20complex%20project,testing%2C%20implementation%2C%20and%20maintenance.)
- [19] Admodisastro, N. (2021). Critical Evaluation on Software Development Process Models with Respect to Mobile Software Development. Upm. https://www.academia.edu/50894933/Critical_Evaluation_on_Software_Development_Process_Models_with_Respect_to_Mobile_Software_Development
- [20] Get started with WebRTC. (2012, July 23). web.dev. <https://web.dev/articles/webrtc-basics>

193-16-462

ORIGINALITY REPORT

12%

SIMILARITY INDEX

9%

INTERNET SOURCES

1%

PUBLICATIONS

11%

STUDENT PAPERS

PRIMARY SOURCES

1

Submitted to Daffodil International University

Student Paper

3%

2

Submitted to American Public University System

Student Paper

1%

3

[dspace.daffodilvarsity.edu.bd:8080](https://dspace.daffodilvarsity.edu.bd/8080)

Internet Source

1%

4

Submitted to University of Maryland, Global Campus

Student Paper

1%

5

Submitted to Southern New Hampshire University - Continuing Education

Student Paper

1%

6

Submitted to Northern Kentucky University

Student Paper

1%

7

Mansoor, Masab Ahmed. "Telehealth Transformation: A Mixed-Methods Study on Organizational Change Processes and Outcomes in Private Medical Practices", South College, 2024

Publication

<1%