### **ABSTRACT**

The Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors is a project that aims to provide a platform for doctors at the Covenant University Medical Center to advise students and provide prescriptions. The system will be accessible via the web, allowing doctors to easily access and update patient information, schedule appointments, and review medical history. The system will have a user-friendly interface, with separate modules for doctors, students, and administrators. Doctors will be able to view and update patient information, including medical history, diagnoses, and prescribed medications. They will also be able to schedule appointments and communicate with patients through the platform. Students will be able to access their own medical records and request appointments with doctors. They will also be able to view their prescribed medications and receive notifications when it is time to refill their prescriptions. The administrator module will allow the medical center staff to manage the system, including adding and updating user accounts, setting up appointments, and generating reports. The system will be implemented using a combination of web technologies, including ReactJs and CSS for the front-end, and NodeJs and MongoDb for the back-end. The project will also involve the design and development of a secure database to store patient and user information.

### **CHAPTER ONE**

### **BACKGROUND OF STUDY**

The background of the study for the Design and Implementation of a Web-based Medical Center Management System is centered around the need to improve the efficiency and effectiveness of the medical center at Covenant University. One of the main challenges facing the medical center is the lack of a centralized platform for managing student information and scheduling appointments. Additionally, there is no easy way for students to access their medical records or request appointments with doctors. The use of a web-based medical center management system has the potential to address these challenges by providing a centralized platform for doctors to view and update patient information, schedule appointments, and communicate with patients. It would also allow students to access their medical records and request appointments with doctors, improving the overall experience of seeking medical care at the medical center. There are several similar systems in use at other medical centers, but most of these systems are designed for use by large hospitals or healthcare organizations. The design and implementation of a web-based medical center management system specifically for Covenant University students and medical center doctors is intended to meet the unique needs of this particular population. Overall, the background of study for this project is driven by the need to improve the efficiency and effectiveness of the medical center at Covenant University, as well as the desire to provide a convenient and centralized platform for students and doctors to manage medical care.

### **STATEMENT OF PROBLEM**

The lack of a centralized platform for managing medical care results in delays and difficulties in accessing medical care, which can have a negative impact on the health and well-being of students. In addition, there is currently no easy way for students to access their own medical records or request appointments with doctors. This can be frustrating for students and can lead to missed or delayed medical appointments. The lack of a web-based medical center management system at Covenant University also limits the ability of doctors to effectively communicate with patients and track medical histories, which can impact the quality of care provided. Overall, the lack of a centralized platform for managing medical care at the medical center at Covenant University is a significant problem that needs to be addressed in order to improve the efficiency and effectiveness of the medical center and enhance the experience of seeking medical care for students.

### **AIMS AND OBJECTIVES**

1. To design and develop a web-based medical center management system that is user-friendly and easy to use for both doctors and students.
2. To create a centralized platform for managing patient information, including medical history, diagnoses, and prescribed medications.
3. To provide a convenient way for doctors to schedule appointments and communicate with patients.
4. To allow students to access their medical records and request appointments with doctors.
5. To design and develop a secure database to store patient and user information.
6. To improve the efficiency and effectiveness of the medical center by providing a centralized platform for managing medical care.
7. To enhance the experience of seeking medical care for students by providing a convenient and centralized platform for accessing medical services.
8. To evaluate the effectiveness of the system in meeting the needs of both doctors and students.

### **RESEARCH METHODOLOGY**

1. To design and develop a web-based medical center management system that is user-friendly and easy to use for both doctors and students: The system will be designed and developed using a combination of web technologies, including ReactJs, Redux, and CSS for the front-end, and NodeJs and MongoDb for the back-end. User experience testing will be conducted to ensure that the system is easy to use and intuitive for both doctors and students.
2. To create a centralized platform for managing patient information, including medical history, diagnoses, and prescribed medications: A secure database will be designed and developed to store patient information. The system will allow doctors to view and update patient information, including medical history, diagnoses, and prescribed medications.
3. To provide a convenient way for doctors to schedule appointments and communicate with patients: The system will include a module for doctors to schedule appointments and communicate with patients through the platform.
4. To allow students to access their medical records and request appointments with doctors: The system will include a module for students to access their own medical records and request appointments with doctors.
5. To design and develop a secure database to store patient and user information: The database will be designed and developed using industry-standard security measures to ensure the confidentiality and integrity of patient and user information.
6. To improve the efficiency and effectiveness of the medical center by providing a centralized platform for managing medical care: The system will be evaluated to determine its impact on the efficiency and effectiveness of the medical center, including the time saved by using a centralized platform for managing patient information and scheduling appointments.
7. To enhance the experience of seeking medical care for students by providing a convenient and centralized platform for accessing medical services: User experience testing will be conducted to assess the impact of the system on the experience of seeking medical care for students.
8. To evaluate the effectiveness of the system in meeting the needs of both doctors and students: The system will be evaluated to determine its effectiveness in meeting the needs of both doctors and students, including its impact on the efficiency and effectiveness of the medical center and the convenience and accessibility of medical services for students. This evaluation will involve collecting feedback from both doctors and students and analyzing usage data.

### **SIGNIFICANCE OF THE STUDY**

1. Improved efficiency and effectiveness of the medical center: By providing a centralized platform for managing patient information and scheduling appointments, the system has the potential to improve the efficiency and effectiveness of the medical center. This can result in reduced waiting times and improved patient care.
2. Enhanced convenience and accessibility of medical services for students: By allowing students to access their medical records and request appointments with doctors through the system, the project has the potential to enhance the convenience and accessibility of medical services for students. This can improve the overall experience of seeking medical care for students.
3. Improved communication between doctors and patients: The system will allow doctors to communicate with patients through the platform, which can improve the quality of care provided by enabling doctors to track medical histories and provide timely follow-up care.
4. Enhanced data security and confidentiality: The secure database designed and developed as part of this project will ensure the confidentiality and integrity of patient and user information.

### **LIMITATIONS OF THE STUDY**

1. Technical challenges: The project involves the design and development of a complex system, which may present technical challenges that may require additional resources and time to overcome.
2. User adoption: The success of the system will depend on the willingness of doctors and students to adopt the platform and use it regularly. There may be resistance to using a new system, which could limit the effectiveness of the project.
3. Limited applicability: The system is designed specifically for use at Covenant University and may not be applicable to other medical centers or universities.

### **CHAPTER 2**

### 

### **LITERATURE REVIEW**

One of the key benefits of web-based medical center management systems is their ability to provide a centralized platform for managing patient information and scheduling appointments. This can improve the efficiency and effectiveness of the medical center, as well as the experience of seeking medical care for patients. For example, a study by Kim et al. (2017) found that the implementation of a web-based medical center management system at a large hospital in South Korea resulted in significant improvements in the efficiency and effectiveness of the medical center, including reduced waiting times and improved patient satisfaction.

However, the adoption and use of web-based medical center management systems can also present challenges. One of the main challenges is the cost and resources required to design and implement such systems, which can be a significant barrier for small medical centers or universities (Gao et al., 2019). In addition, user adoption of the system can be a challenge, as there may be resistance to using a new system, particularly among doctors who are used to traditional paper-based systems (Zhou et al., 2018).

To address these challenges, it is important to consider the user experience when designing and implementing a web-based medical center management system. This includes ensuring that the system is easy to use and intuitive, as well as providing training and support to ensure that users are able to effectively utilize the system (Rhee et al., 2016). In addition, it is important to consider the specific needs of the target user group, in this case, Covenant University students and medical center doctors, in order to ensure that the system meets their needs and is well-suited to their workflow.

Overall, the literature review for the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors should include a comprehensive analysis of the research on the use of web-based medical center management systems, the benefits and challenges of such systems, and the factors that contribute to their success or failure. This review should also consider the specific needs and challenges of the target user group in order to inform the design and implementation of the system.

### **RELEVANT TERMS USED IN THIS STUDY**

1. Web-based medical center management system: A platform that is accessed via the web and used to manage patient information, schedule appointments, and communicates with patients.
2. User interface: The part of the system that is visible to users and allows them to interact with the system.
3. Database: A collection of data that is stored in a structured manner and can be accessed and updated through a computer system.
4. Front-end: The part of the system that is visible to users and handles the user interface and user experience.
5. Back-end: The part of the system that handles data storage and management, including the database and server-side logic.
6. ReactJs: it is a JavaScript library for building user interfaces, primarily used for building single-page applications and mobile applications.
7. CSS: A style sheet language used to control the appearance of web content.
8. Node.js : is an open-source, cross-platform runtime environment that executes JavaScript code outside of a web browser, allowing developers to build scalable, high-performance server side applications.
9. MongoDB: is a cross-platform, document-oriented database program that uses JSON-like documents with optional schemas, allowing developers to store and query large amounts of data efficiently.

### **REVIEW OF EXISTING SYSTEMS**

There are several existing web-based medical center management systems that are in use at hospitals and healthcare organizations around the world. These systems typically provide a centralized platform for managing patient information, scheduling appointments, and communicating with patients.

1. One example of an existing system is the Epic electronic health record (EHR) system, which is used by hospitals and healthcare organizations in the United States. The Epic EHR system includes a range of features, including the ability to manage patient information, schedule appointments, and communicate with patients through secure messaging. The system also includes tools for managing population health and generating reports.
2. Another example is the Cerner electronic health record (EHR) system, which is used by hospitals and healthcare organizations globally. The Cerner EHR system includes similar features to the Epic EHR system, including the ability to manage patient information, schedule appointments, and communicate with patients. It also includes tools for managing population health and generating reports.
3. Meditech: A web-based EHR system used by hospitals and healthcare organizations globally. The Meditech system includes features for managing patient information, scheduling appointments, and communicating with patients. It also includes tools for managing population health and generating reports.
4. AthenaHealth: A cloud-based EHR and practice management system used by medical practices and hospitals. The AthenaHealth system includes features for managing patient information, scheduling appointments, and communicating with patients. It also includes tools for managing billing and generating reports.
5. Practice Fusion: A cloud-based EHR system used by medical practices. The Practice Fusion system includes features for managing patient information, scheduling appointments, and communicating with patients. It also includes tools for managing billing and generating reports.
6. Greenway Health: A web-based EHR and practice management system used by medical practices and hospitals. The Greenway Health system includes features for managing patient information, scheduling appointments, and communicating with patients. It also includes tools for managing to bill and generating reports.

### **PRINCIPLES AND METHODS FOR THE SYSTEM**

1. User-centered design: This approach focuses on designing the system to meet the needs and preferences of the target user group, in this case, Covenant University students and medical center doctors. This may involve conducting user experience testing and gathering feedback to inform the design of the system.
2. Agile development: This method involves iterative development of the system, with a focus on rapid prototyping and continuous improvement. This allows the system to be developed and refined in a flexible and responsive manner.
3. Data security and confidentiality: Ensuring the confidentiality and integrity of patient and user information is a key concern when designing and implementing a web-based medical center management system. This may involve using industry-standard security measures and encryption techniques to protect sensitive data.
4. Interoperability: Ensuring that the system can easily exchange data with other systems, such as EHR systems or billing systems, is important for the efficient and effective operation of the medical center.
5. Scalability: The system should be designed to accommodate the potential growth in the number of users and the volume of data over time.

### **CHAPTER 3**

### **SYSTEM ANALYSIS AND DESIGN**

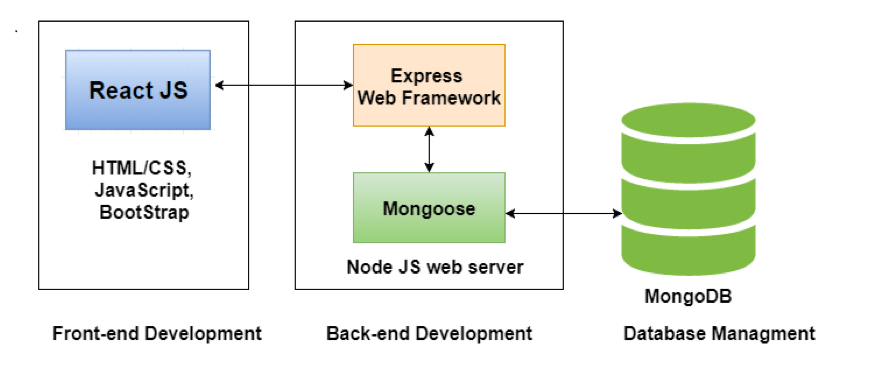
The system analysis and design for the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors would involve several key steps:

1. Define the scope and requirements of the system: This involves identifying the specific needs and requirements of the target user group, in this case, Covenant University students and medical center doctors, and defining the scope of the system to ensure that it meets these needs. This may involve conducting user research, such as interviews and focus groups, to gather feedback and input from the target user group.
2. Design the system architecture: This involves designing the overall structure and organization of the system, including the database, user interface, and server-side logic. This step may involve creating mockups or prototypes of the system to visualize how it will function and look.
3. Develop the system: This involves writing the code and implementing the system based on the design developed in the previous step. This may involve using a range of web technologies.
4. Test and debug the system: This involves testing the system to ensure that it is functioning correctly and fixing any issues that are identified.
5. Deploy the system: This involves launching the system and making it available for use by the target user group.
6. Evaluate the system: This involves collecting feedback from the target user group and analyzing usage data to determine the effectiveness of the system in meeting the needs of the medical center and the experience of seeking medical care for students.

### **SYSTEM ARCHITECTURE**

The system architecture for the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors would likely include the following components:

1. User interface: The user interface (UI) is the part of the system that is visible to users and allows them to interact with the system. The UI for this project would likely include a range of features, such as a login page, a dashboard for doctors to view and update patient information, a module for students to access their medical records and request appointments, and an administrator module for medical center staff to manage the system.
2. Database: The database is a collection of data that is stored in a structured manner and can be accessed and updated through the system. The database for this project would likely include tables for storing patient information, user accounts, and appointment schedules.
3. Server-side logic: The server-side logic handles the processing of data and the generation of dynamic content for the UI. This may include functions for managing user accounts, updating patient information, and generating reports.
4. Web server: The web server is responsible for hosting the system and serving the UI and other content to users.



System architecture

### **BUSINESS LOGIC**

The business logic for the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors would include the rules and processes that govern the operation of the system. This would likely include the following:

1. User accounts: The system would likely include rules and processes for managing user accounts, including the creation, modification, and deletion of user accounts. This may involve verifying the identity of users, assigning them appropriate permissions, and allowing them to reset their passwords if necessary.
2. Patient information management: The system would likely include rules and processes for managing patient information, including the creation, modification, and deletion of patient records. This may involve verifying the identity of patients, recording their medical history, and updating their records as needed.
3. Appointment scheduling: The system would likely include rules and processes for scheduling appointments, including the creation, modification, and cancellation of appointments. This may involve verifying the availability of doctors, ensuring that appointments are spaced appropriately, and allowing patients to request specific appointment times.
4. Communication with patients: The system would likely include rules and processes for allowing doctors to communicate with patients through the platform, including the ability to send and receive messages, track medical histories, and provide follow-up care.

### **DATA ACCESS LAYER**

The data access layer for the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors would be responsible for managing the storage and retrieval of data from the database. This layer would typically include the following components:

1. Data access objects (DAOs): The DAOs are responsible for handling the interactions between the system and the database, including the creation, modification, and deletion of records. They would typically include methods for querying the database and returning the results to the system.
2. Connection pool: A connection pool is a group of database connections that can be reused, rather than creating a new connection each time the database is accessed. This can improve the performance of the system by reducing the overhead of creating new connections.
3. Data transfer objects (DTOs): The DTOs are used to transfer data between the system and the database. They may include fields for storing data such as patient information, user accounts, and appointment schedules.

### **SYSTEM REQUIREMENTS**

The system requirements for the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors would likely include both functional and non-functional requirements.

### **FUNCTIONAL REQUIREMENTS**

Functional requirements are specific features or capabilities that the system must have in order to meet the needs of the target user group. Some examples of functional requirements for this project might include:

1. A login page for doctors to access the system
2. A dashboard for doctors to view and update patient information
3. A module for students to access their medical records and request appointments
4. An administrator module for medical center staff to manage the system
5. A secure messaging system for doctors to communicate with patients

### **NON-FUNCTIONAL REQUIREMENTS**

Non-functional requirements are broader requirements that relate to the overall performance, reliability, and usability of the system. Some examples of non-functional requirements for this project might include:

1. The system must be accessible from any device with a web browser
2. The system must be able to handle a high volume of users and data
3. The system must be secure, with measures in place to protect patient and user data
4. The system must be user-friendly and easy to use
5. The system must be scalable, with the ability to accommodate growth in the number of users and data over time

### **SYSTEM DESIGN**

The system design for the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors would involve the development of a detailed plan for the structure and organization of the system. This would likely include the following components:

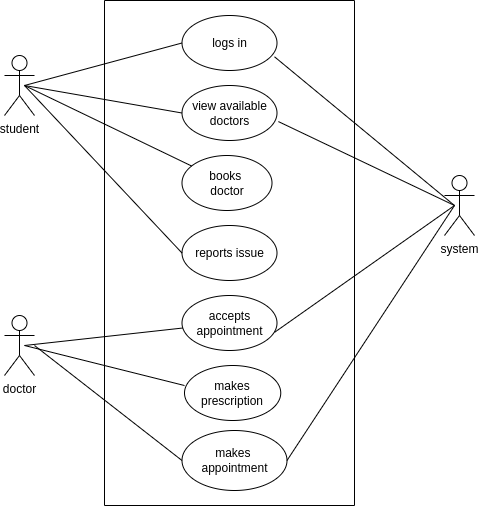
1. User interface design: The user interface (UI) is the part of the system that is visible to users and allows them to interact with the system. The UI design for this project would involve creating mockups or prototypes of the system to visualize how it will function and look. This may involve using tools such as wireframes or user flows to design the layout and navigation of the system.
2. Database design: The database is a collection of data that is stored in a structured manner and can be accessed and updated through the system. The database design for this project would involve creating tables and relationships to store and organize the data that the system will need to access. This may involve using a database management system such as MySQL to create and manage the database.
3. Server-side logic design: The server-side logic handles the processing of data and the generation of dynamic content for the UI. The design of the server-side logic for this project would involve creating functions and algorithms to handle tasks such as managing user accounts, updating patient information, and generating reports. This may involve using a programming language such as PHP to implement the server-side logic.

### **USE CASE DIAGRAM**

A use case diagram is a type of diagram that is used to represent the interactions between the system and the users in a specific context. It is typically used to model the functional requirements of the system and to identify the relationships between the different actors (i.e., the users of the system) and the use cases (i.e., the specific actions or processes that the system performs).

In the context of the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors, a use case diagram might include the following elements:

1. Actors: The actors in this system would likely include doctors, students, and medical center staff.
2. Use cases: The use cases for this system might include managing patient information, scheduling appointments, communicating with patients, and managing user accounts.
3. Relationships: The relationships between the actors and the use cases might include the doctors accessing and updating patient information, students requesting appointments, and medical center staff managing the system.



Use case diagram of proposed system

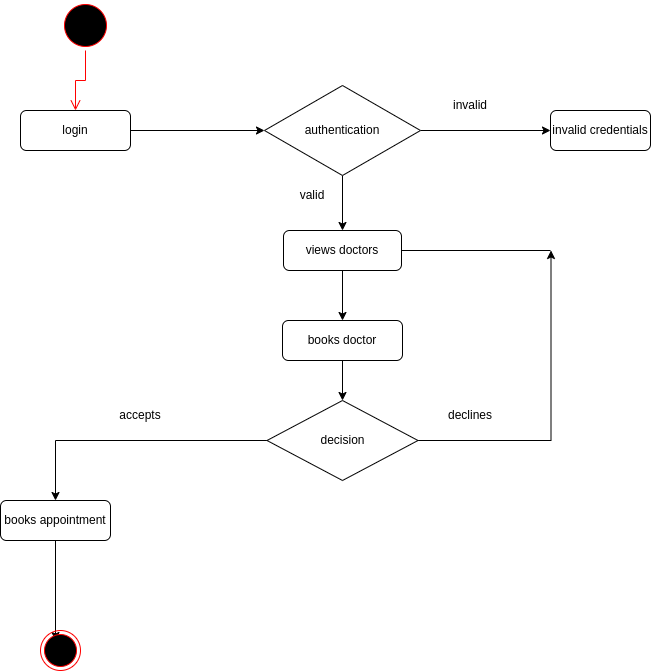
### **ACTIVITY DIAGRAM**

An activity diagram is a type of diagram that is used to model the flow of activities or processes within the system. It is typically used to represent the steps involved in a specific process or task, and to show the relationships between the different activities.

In the context of the Design and Implementation of a Web-based Medical Center Management System for Covenant University Students and Medical Center Doctors, an activity diagram might be used to model the flow of activities or processes within the system, such as the steps involved in a doctor updating a patient's medical record or a student requesting an appointment.

An activity diagram for this study might include the following elements:

1. Activities: The activities in this system might include tasks such as logging in, accessing patient information, scheduling an appointment, and sending a message to a doctor.
2. Flow of control: The flow of control in this system would show the order in which the activities are performed and the conditions under which certain activities are triggered. For example, a student might need to log in before they can access their medical record or request an appointment.
3. Decision points: The activity diagram might include decision points where the system must make a choice based on certain conditions. For example, the system might need to determine whether a doctor is available to schedule an appointment before allowing a student to request one.



ACTIVITY DIAGRAM OF SYSTEM