REPUBLIC OF CAMEROON **REPUBLIQUE DU CAMEROUN**

Peace – Work – Fatherland **Paix – Travail – Paitrie**

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER ENGINEERING

SPECIALIZATION: SOFTWARE ENGINEERING

COURSE TITLE: SOFTWARE QUALITY TOOLS AND TESTING

ASSESSMENT AND FEATURE REQUIREMENT FOR TELEMEDCINE APP: MEDIK

	NAMES	MATRICULE NUMBERS
1	QUINUEL TABOT NDIP-AGBOR	FE21A300
2	INDAH RICOBELLE MBAH	FE21A204
3	SIRRI THERESIA ANYE	FE21A306
4	TSAPZE ZAMBOU ROSELINE	FE21A328
5	ENOW MYKE-AUSTINE ETA	FE21A183
6	MOKFEMBAM FABRICE KONGNYUY	FE21A240
7	MONDOA ROBERT	FE21A241
8	TAKEM JIM	FE21A309
9	KENFACK SAMEZA	FE21A213
10	FOLETIA AJONGAKUI SYNCLAIR	FE21A190

COURSE INSTRUCTOR: DR. TSAGUE ALINE

TABLE OF CONTENTS

I. INTRODUCTION

Telemedicine applications, also known as telehealth apps, have emerged as transformative tools in the healthcare landscape. These mobile applications enable individuals to access healthcare services remotely, bridging geographical distances and overcoming mobility challenges. By leveraging the power of smartphones and other digital devices, telemedicine applications are revolutionizing the way healthcare is delivered and consumed.

MEDIK, a telemedicine application, stands at the forefront of this transformative movement. MEDIK offers a comprehensive suite of services designed to provide convenient and accessible healthcare to individuals from all walks of life. The application's core functionalities include:

- **Real-time video and audio consultations:** MEDIK empowers patients to connect with licensed healthcare providers via secure video or audio calls, enabling face-to-face interactions for diagnosis, treatment planning, and follow-ups.

- **Secure messaging:** MEDIK facilitates seamless communication between patients and healthcare providers through a secure messaging platform, allowing for asynchronous exchanges of information, questions, and concerns.

- **Online prescription generation:** MEDIK streamlines the prescription process by allowing healthcare providers to electronically generate and send prescriptions directly to pharmacies, eliminating the need for paper prescriptions.

- **Pharmacy services:** MEDIK integrates with partner pharmacies, enabling patients to conveniently order and receive medications directly through the application.

- **Appointment scheduling:** MEDIK simplifies appointment scheduling, allowing patients to browse available time slots and book appointments with their preferred healthcare providers directly from the app.

- **Personalized health notifications:** MEDIK goes beyond consultations and prescriptions by providing personalized health notifications based on a patient's medical history, lab test results, and diagnoses. These notifications may include health tips, medication reminders, and follow-up appointment prompts.

  MEDIK's comprehensive suite of features caters to a wide range of healthcare needs, from routine checkups and prescription renewals to chronic disease management and mental health counseling. The application's user-friendly interface and secure communication protocols make it accessible to individuals of all ages and technical proficiency.

In a world where healthcare accessibility is often hindered by geographical barriers, mobility limitations, and time constraints, telemedicine applications like MEDIK are paving the way for a more equitable and inclusive healthcare system. By bridging the gap between patients and healthcare providers, these applications are empowering individuals to take control of their health and well-being.

II. PROJECT DETAILS, REQUIREMENT DESCRIPTION AND STANDARDS ANALYSIS

1) FUNCTIONAL REQUIREMENTS:

➢ User Authentication and Security of User Data:

- Secured user registration and login using multiple authentication factors, including username, password, and biometric fingerprint.
- Enforced password complexity and change requirements.
- Secure storage and encryption of user credentials and personal data.
- Implementation of role-based access control (RBAC) to restrict access to sensitive information.
- Regular security audits and vulnerability assessments.

➢ Patient-Healthcare Provider Communication and Consultation:

- Real-time video, audio, and messaging communication between patients and healthcare providers.
- Scheduling and management of virtual consultations.
- Secure file sharing for exchanging medical records, images, and other relevant documents.
- Integration with third-party communication platforms.
- Real-time translation services for multilingual communication.

➢ Payment Processing and Transactions

- Integration with secure payment gateways for online transactions.
- Support for multiple payment methods (credit cards, debit cards, mobile wallets).
- Generation of detailed transaction receipts for consultations, diagnosis, and medication purchases.
- Secure storage and processing of payment information.
- Compliance with relevant payment industry standards (PCI DSS).

➢ Appointment Scheduling and Management

- Real-time scheduling and management of appointments for both patients and healthcare providers.
- Availability of appointment slots based on healthcare providers' schedules.
- Synchronization with healthcare providers' calendars and scheduling systems.
- Automated appointment reminders and notifications.
- Waitlist management for oversubscribed appointments.
- Integration with patient calendars for seamless scheduling.

➢ **Medical Records Management**

- Secure and centralized storage of patient medical records, including test results, prescriptions, and other relevant health information.
- Patient-controlled access to their medical records.
- Sharing of medical records with healthcare providers with patient consent.
- Data encryption and access logging for audit purposes.
- Compliance with relevant data privacy regulations (HIPAA, GDPR).
- Integration with electronic health record (EHR) systems for comprehensive patient data management.

➢ **Prescription Services**

- Electronic prescription (eRx) generation and management.
- Direct transmission of prescriptions to pharmacies.
- Patient consent management for sharing prescription information with pharmacies.
- Integration with pharmacy management systems.
- Medication affordability and delivery options, including generic drug recommendations and pharmacy discounts.

2) <u>NON-FUNCTIONAL REQUIREMENTS:</u>

➢ **Robust Security:**

- MEDIK adheres to strict security standards (e.g., HIPAA compliance) to ensure the confidentiality, integrity, and privacy of patient data.
- MEDIK provides strongly authenticated, access controlled and sustainable database to store and backup data of patients for medical and research purposes.

- MEDIK is to provide a secured internet connection that is essential to enable video conferencing between patients and healthcare providers mainly by encryption.

➢ **Usability and User Experience:**

- MEDIK has an intuitive and user-friendly interface, making it easy for users to navigate, schedule appointments, communicate, and access their information.
- MEDIK incorporates an expert assistant to walk new users through the interface as well as provide directives to achieving patient's objective on the application.

➢ **Reliability and Availability:**

- MEDIK is reliable, ensuring consistent availability and minimal downtime to avoid.
- MEDIK incorporate error free payments and retrieval of user data at any time possible time.

➢ **Accessibility:**

- MEDIK will comply with accessibility standards to ensure that individuals with disabilities can access and use the app effectively.
- A text to speech and speech to text functionality to make it comfortable for disabled users.

➢ **Performance and Scalability:**

- MEDIK is capable of handling concurrent users and providing a smooth and responsive experience, even during peak usage times.
- MEDIK has fast response time between using action and effect hence reducing waiting time.

➢ **Compatibility and Integrations:**

- The app should be compatible with various devices, operating systems (e.g., iOS, Android), and browsers to cater to a wide range of users.
- It should also support integrations with other healthcare systems, such as electronic health records (EHR) or pharmacy systems.

➢ **Regulatory Compliance:**

- The app should adhere to relevant regulations and guidelines, such as HIPAA, GDPR, and local telemedicine practice guidelines.

- **Internationalization:**

  - MEDIK is multilingual and can be used across 50+ international languages.

3) ARCHITECTURAL REQUIREMENTS:

- **Stakeholders:**

  - Medical will basically involve three major stakeholders i.e., the patient, healthcare center (which will consist of health experts) and a pharmacy.

- **Layers:**

MEDIK consists of 3 main layers or tiers namely

- **The Presentation layer** also commonly known as the frontend or user interface using React native due to its light weight and cross platform nature.

- **The Application layer** also commonly known as the logic tier or middle tier which will be done using TypeScript due to its assured security and high-performance in integration with React native.

- **The Data layer** which will involve manipulation of data in the database which will be done using MongoDB due to its role-based access control features as well as security features like authentication, authorization, and encryption

- **User Interfaces (Clients)**
  - **Patient App**: A mobile or web app that allows patients to register, sign in, manage their profiles, schedule appointments, and communicate with healthcare providers in real-time.
  - **Healthcare Provider App**: A web app used by doctors and other healthcare providers to manage appointments, access patient records, and conduct virtual consultations.
  - **Admin Panel**: A web-based interface for system administrators to manage the platform, users, and data.

- **Communication Components**

- **Video/Audio Streaming**: Real-time communication channels for virtual consultations.
- **Chat Messaging**: Text-based communication for quick queries and follow-ups.

- **APIs and Logic**
  - **Authentication and Authorization**: Secure methods for user registration and access control.
  - **Appointment Management**: Functionality to schedule, modify, or cancel appointments.
  - **Medical Record Management**: Secure storage and retrieval of patient medical records.
  - **AI Model API**: for Expert System consultations and user guide through system

- **Storage and Infrastructure Services**
  - **Database**: A secure database to store user data, medical records, appointment details, and other relevant information.
  - **File Storage:** For storing and sharing documents, images, and other files.
  - **Payment Gateway**: To process financial transactions for services rendered.

- **Security Measures**
  - **Data Encryption**: To protect sensitive data during transmission and at rest.
  - **Compliance**: Adherence to healthcare regulations like HIPAA for patient data privacy.
  - **Access Logs and Monitoring**: To track usage and detect any unauthorized access.

- **Integration with External Systems**
  - **Pharmacy Systems**: For sending prescriptions and managing medication orders.

4) INTERNATIONAL QUALITY STANDARDS FOR TELEMEDICINE APPS:

Internationally socially used standards include:

- **Health Insurance Portability and Accountability Act (HIPAA):** HIPAA sets the standards for protecting sensitive patient health information in the United States and backed by the **Wealth Health Organisation (WHO).** If your telemedicine app serves users in the U.S., compliance with HIPAA regulations is crucial to safeguard patient privacy and security.

- **General Data Protection Regulation (GDPR):** GDPR is a comprehensive data protection regulation that applies to the European Union (EU) and European Economic Area (EEA). If your telemedicine app serves users in EU/EEA countries, compliance with GDPR is necessary to ensure the lawful and secure processing of personal data.

- **International Organization for Standardization (ISO) 27001:** ISO 27001 is an internationally recognized standard for information security management systems. Adhering to ISO 27001 guidelines helps ensure that your telemedicine app implements robust security controls, risk management processes, and data protection measures.

The professional descriptions of most commonly used international standards for telemedicine apps include:

- **Health Level 7 (HL7) :** HL7 is an international standard for the exchange of electronic health record (EHR) data. It is a comprehensive standard that covers a wide range of clinical and administrative data, including patient demographics, diagnoses, medications, allergies, and laboratory results. HL7 is widely used in telemedicine applications to exchange data between different healthcare systems.

- **Digital Imaging and Communications in Medicine (DICOM):** DICOM is an international standard for the exchange of medical images, such as X-rays, CT scans, and MRI scans. DICOM is widely used in telemedicine applications to transmit medical images from one location to another for consultation or diagnostic purposes.

- **Fast Healthcare Interoperability Resources (FHIR):** FHIR is a newer standard that is based on modern web technologies, such as JSON and HTTP. FHIR is designed to be more flexible and easier to implement than HL7, and it is gaining popularity in the telemedicine industry.

III. ANALYSIS AND RESEARCH ILLUSTRATION FOR MEDIK

1) OVERVIEW

Telemedicine is the use of telecommunications technology to provide healthcare services remotely. The telemedicine app, MEDIK, is out to provide patients with convenient and affordable access to healthcare services, especially for those in rural or underserved areas.

2) ANALYSIS

MEDIK is a telemedicine app that provides patients with a variety of services, including:

- Video, audio and messages consultations with healthcare providers
- Prescription management
- Medical records management
- Secure communication
- Appointment scheduling
- Patient education materials
- Pharmacy services

MEDIK is a user-friendly app that is easy to navigate. The app is also secure and compliant with all applicable privacy regulations.

3) RESEARCH

There is a growing body of research that supports the effectiveness of telemedicine apps. For example, a study published in the Journal of the American Medical Association found that telemedicine was as effective as in-person care for a variety of conditions, including chronic diseases and mental health conditions. And Fig 1. Below shows an illustration of MEDIK's details

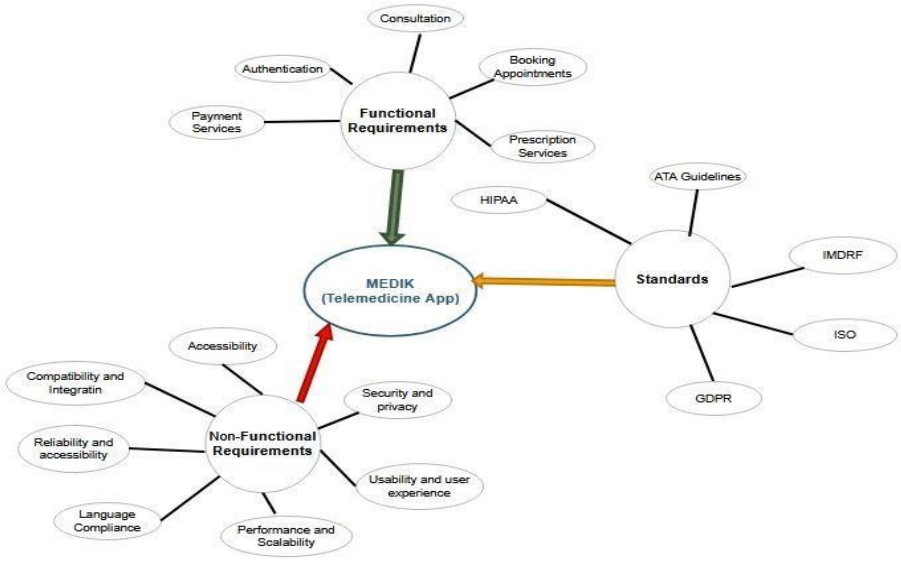

Figure 1: Illustration of requirement of MEDIK

4) ILLUSTRATION

Imagine a patient named Sarah who lives in a rural area. Sarah has a chronic disease that requires regular monitoring. In the past, Sarah had to drive for hours to see her doctor. However, now that she has MEDIK, she can schedule a video consultation with her doctor from the comfort of her own home.

IV. SYSTEM, USER INTERFACE (UI) AND DATABASE DESIGN OF MEDIK

Following the requirements, research and analysis done, the design was divided in two (2) major phases namely:

- System design
- User Interface (UI) design
- Database design

1. SYSTEM DESIGN

System Design refers to the process of defining the architecture, modules, components, interfaces, and data flow of a system to satisfy specific requirements. It involves creating a blueprint for the construction of a system that meets the requirements of the application. In achieving this design, mainly UML (Unified Modelling Language) was used which practically involved providing UML diagrams based on analysis and then describing them. The System design involved the use of four UML diagrams. That is:

- Use case diagram
- Class diagram
- Activity diagram
- Sequence diagram

❖ **Use case for MEDIK**

The use case diagram of MEDIK as part of the UML describes and graphically represents the actors, the different use cases (action or system services to fulfill

the needs of actors) and their relationships as well a clarify scope and system boundaries of MEDIK.
Figure 2 below shows the use case diagram of MEDIK.

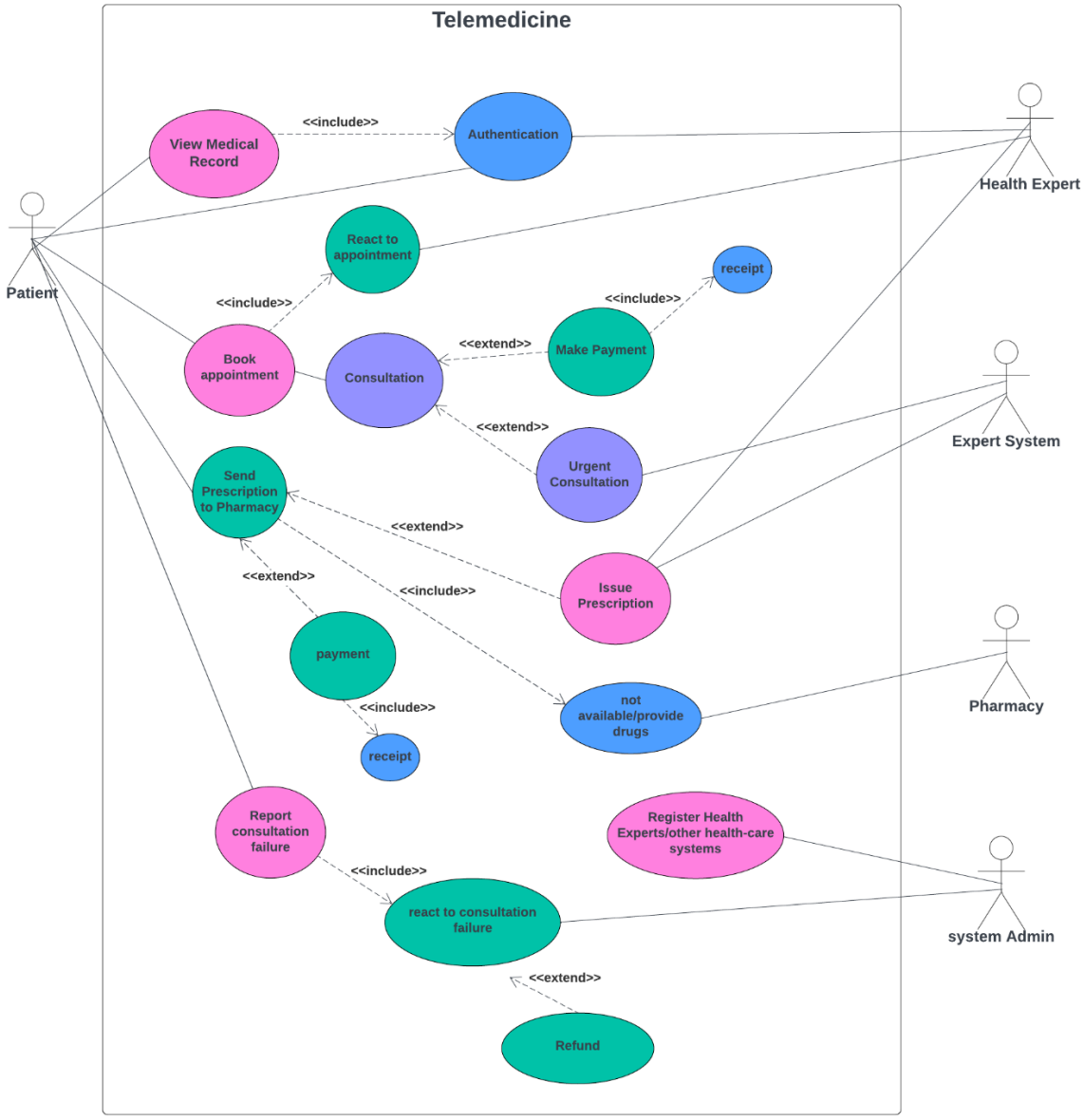

Figure 2: Use case diagram of telemedicine application MEDIK

Detailed Description of MEDIK's Use Cases

1. **View Medical Record**

- **Actor**: Patient

- **Goal:** To allow the patient to access and view their medical history and records.

- **Preconditions**: Patient must be logged into the system.

- **Postconditions**: Patient views the desired medical records.

- **Basic Flow**:

1. Patient requests to view medical record.

2. System verifies patient's identity and authorization.

3. System retrieves and displays the medical record.

2. **React to Appointment**

- **Actor**: Health Expert
- **Goal**: To respond to patient's request to access consultation and prescription services with health expert within the telemedicine system.
- **Preconditions**: Health expert is logged in and has reviewed the patient's request.
- **Postconditions**: Patient gains access to requested services.
- **Basic Flow**:

1. Health expert receives a request for confirmation.

2. Health expert reviews schedule and the patient's medical history.

3. Health expert grants or denies authorization for consultation.

3. **Consultation**

- **Actor:** Patient, Health Expert
- **Goal:** To facilitate a virtual consultation and prescription between the patient and the health expert.

- **Preconditions**: Patient has been approved on request of consultation via appointment booking.
- **Postconditions**: Consultation is completed, and advice is given.
- Basic Flow:

1. Patient requests a consultation and is approved.
2. System schedules the consultation session with the health expert.
3. Health expert and patient conduct the consultation.

4. **Make Payment**

- **Actor**: Patient
- **Goal**: To process payment for the telemedicine services rendered.
- **Preconditions**: Patient has billing information after request for consultation is authorized (approved).
- **Postconditions**: Payment is processed, and receipt is generated.
- **Basic Flow**:

1. Patient selects services for payment.
2. System calculates total cost and presents it to the patient.
3. Patient confirms payment, and the system processes the transaction.

5. **Send Prescription**

- **Actor**: Patient, Pharmacy
- Goal: To send a prescription from the patient to the pharmacy.
- Preconditions: Consultation has occurred, and a prescription is needed.
- Postconditions: Pharmacy receives the prescription and prepares medication.
- Basic Flow:

1. Health expert generates a prescription.

2. Prescription is sent electronically to the chosen pharmacy by patient.

3. Pharmacy confirms receipt and prepares medication for the patient.

6. **Register Health Experts/ Health-care Provider**

- **Actor:** System Admin

- **Goal:** To register new health-care providers into the telemedicine system.

- **Preconditions:** System admin has the necessary information to register a provider.

- **Postconditions:** New health-care provider is added to the system.

- **Basic Flow:**

1. System admin collects required information from the health-care provider.

2. System admin inputs the information into the system.

3. System confirms registration and notifies the health-care provider.

7. **Urgent Consultation**

- **Actor:** Expert System
- **Goal:** To get an instant consultation session with patient for prescription and recommendation.
- **Preconditions:** Patient has interacted with the expert system.
- **Postconditions:** Expert system provides prescription and recommendations.
- **Basic Flow:**

1. Patient makes aware symptoms and concerns to the expert system.

2. Expert system analyzes the information.

3. Expert system issues prescriptions and recommendations.

8. **Report Consultation Failure**

- **Actor:** Patient

- **Goal:** To resolve issues with regards to consultation failure.

- **Preconditions:** Consultation scheduled for a given time didn't push through.

- **Postconditions:** Issue is resolved and may involve a refund.

- **Basic Flow:**

1. Patients reports failure to System admin.

2. System Admin reviews report and checks if are in line with system logs.

3.System Admin gives feedback (Reacts to report) and may involve refund.

❖ **Class Diagram for MEDIK**

The class diagram of MEDIK telemedicine application visualises the classes and attribute, methods and relationship between classes. The class diagram of MEDIK is shown in figure 3 below.

Figure 3: Class Diagram of MEDIK

Description of Class Diagram

Classes and Their Attributes:

- Medical Record, Patient, Doctor, Health Expert, Pharmacy, Appointment, Payment, and Prescription are the main classes represented as boxes.

- Inside each box, the attributes (properties or data members) and methods (functions or services) of the class are listed as in line with the analysis and requirements.

Relationships:

- The lines connecting the boxes indicate the **associations** between classes. For example, the Medical Record class is associated with Patient, Doctor, and Health Expert classes.

- **Associations** can represent different types of relationships, such as aggregation, composition, or inheritance, which define how objects of these classes interact with each other.

Purpose of Class Diagrams:

- Class diagrams are part of the Unified Modeling Language (UML) and are used to model the static structure of systems.

- They help developers and stakeholders understand the system's architecture and the interconnections between different components.

- They serve as a blueprint for implementing the system in code.

- **Activity Diagram for MEDIK**

  The activity diagram of MEDIK telemedicine application shows the flow of action in the application. The activity diagrams of MEDIK and their description are shown below:

  - **Login / Registration:**

    This activity diagram effectively illustrates the steps a user must take to either log in or register for the healthcare system, as well as the possible outcomes of each action. It also highlights the different roles users can assume once they are part of the system

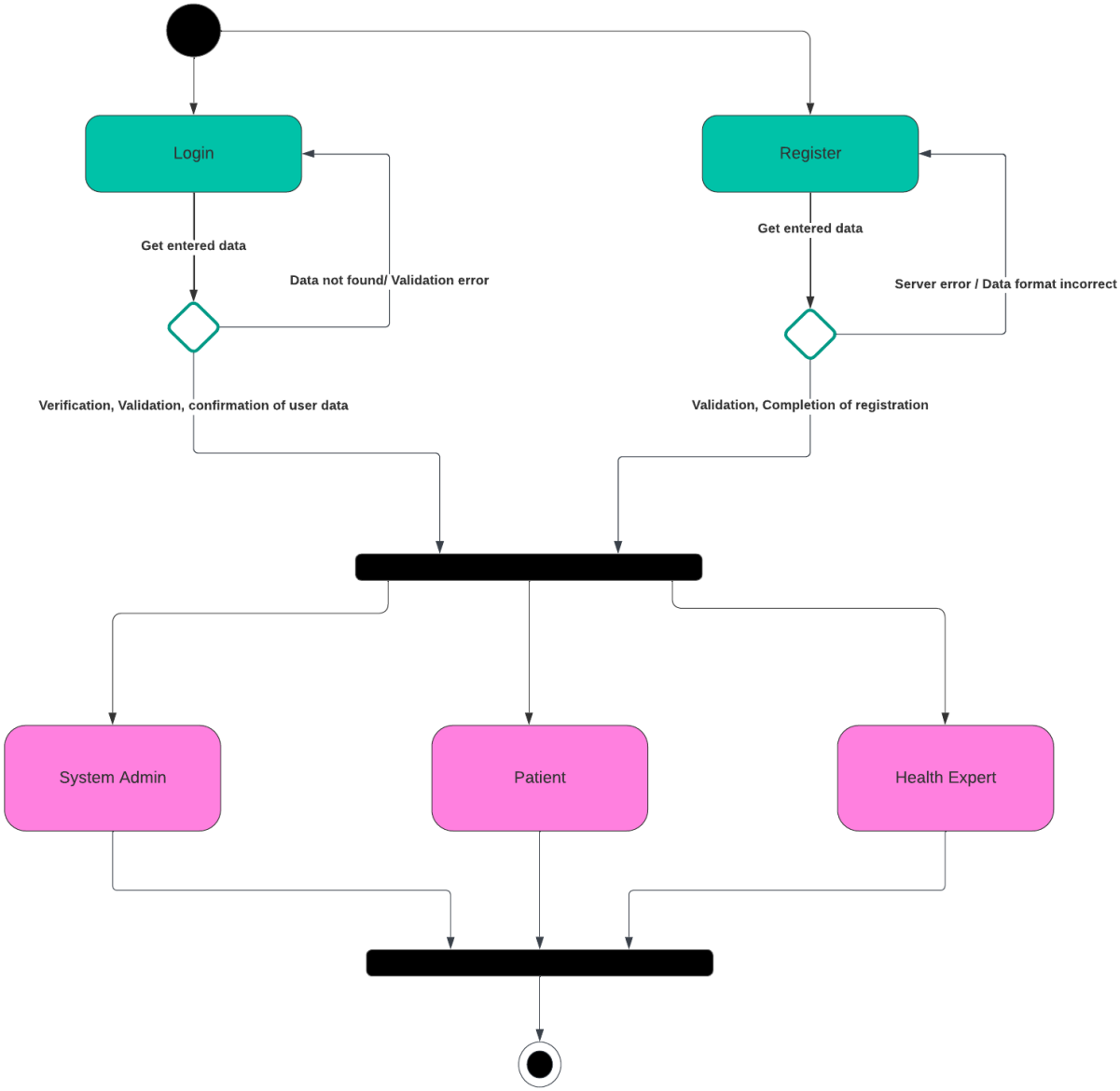

Figure 4: Activity for Login and Registration for MEDIK

Description of Diagram

Login Process:

1. **Start**: The process begins when a user attempts to log in.

2. **Enter Data**: The user enters their login credentials.

3. **Data Validation**: The system checks if the entered data is found and correct.

- If the data is not found or incorrect, the user receives an error message and is prompted to try again.

- If the data is correct, the user is verified and validated.

4. **Confirm User Data**: Upon successful validation, the user's data is confirmed, and they gain access to the system.

Registration Process:

1. **Start**: The process initiates when a user decides to register.

2. **Enter Data**: The user provides their registration details.

3. **Data Validation**: The system validates the entered data.

- If the data is incorrect, the user receives an error message and is asked to correct the details.

- If the data is correct, the user is validated.

4. **Complete Registration**: After successful validation, the user's registration is completed, and they can now log in to the system.

User Roles:

- The diagram also outlines the different roles a user can have within the system:

- **System Admin:** Responsible for managing the system's overall functionality and user accounts.

- **Patient:** Can access personal health records, schedule appointments, and communicate with health experts.

- **Health Expert:** Provides healthcare services and conducts consultations.

- **Appointment booking for consultation:**

  This activity diagram illustrates the booking of appointments by patients for consultation with health expert. Figure 5 and 6 below illustrates how appointment booking occurs in MEDIK on the patient and doctor's end respectively.

Figure 5: Patients activity for appointment booking

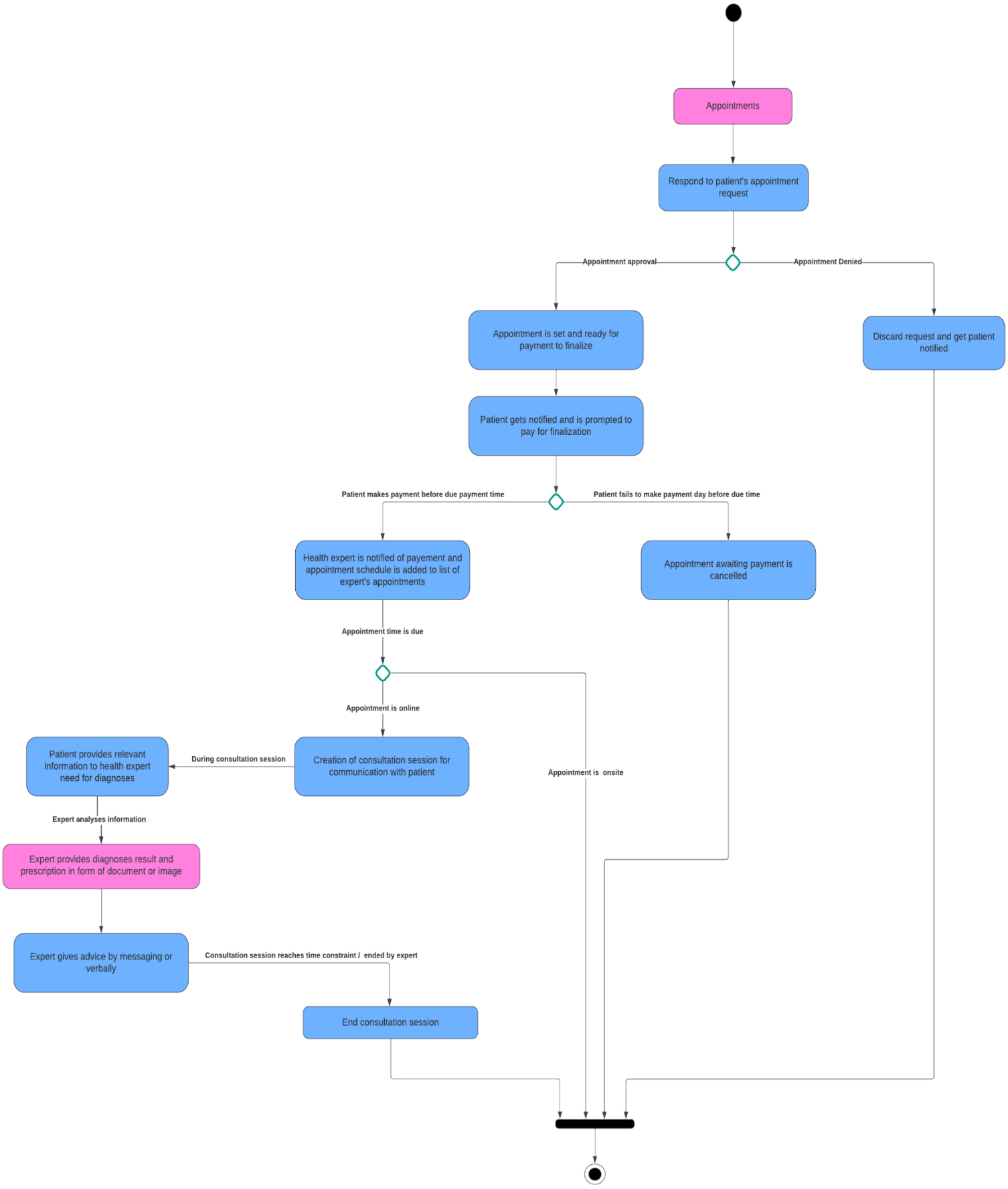

Figure 6: Health Experts activity

- **Patient Activity:**
  Beside booking appointments patients also have the ability to view their medical record, report consultation issues and send prescription they received from health expert to pharmacy. The figure below illustrates the flow of action for all these.

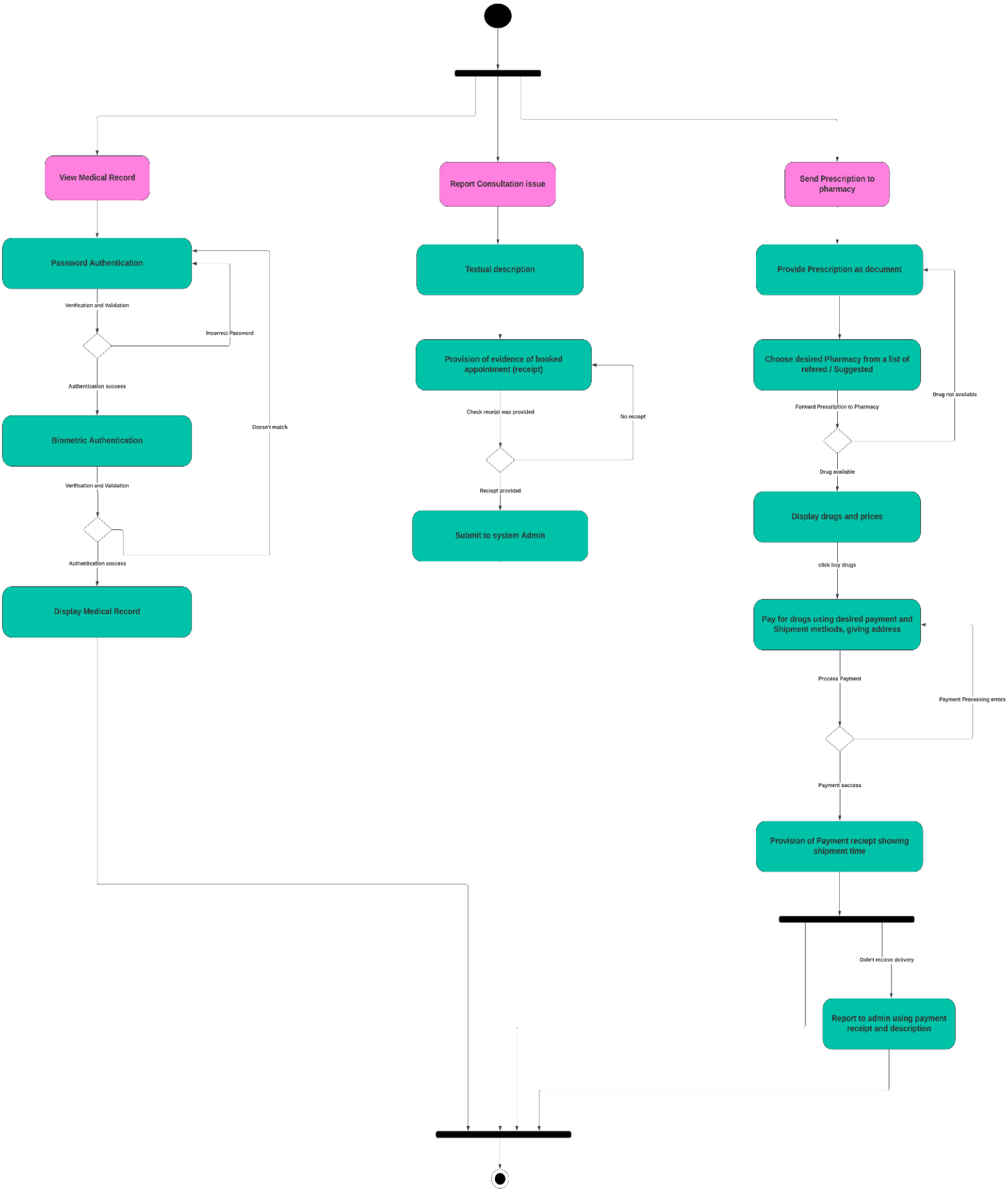

Figure 7: Patient's activity

❖ **Sequence Diagram for MEDIK**

The sequence diagram focuses on illustrating the dynamic behavior of the telemedicine system, showing the flow of messages and the sequence of method invocations. Illustrations are shown by figures below

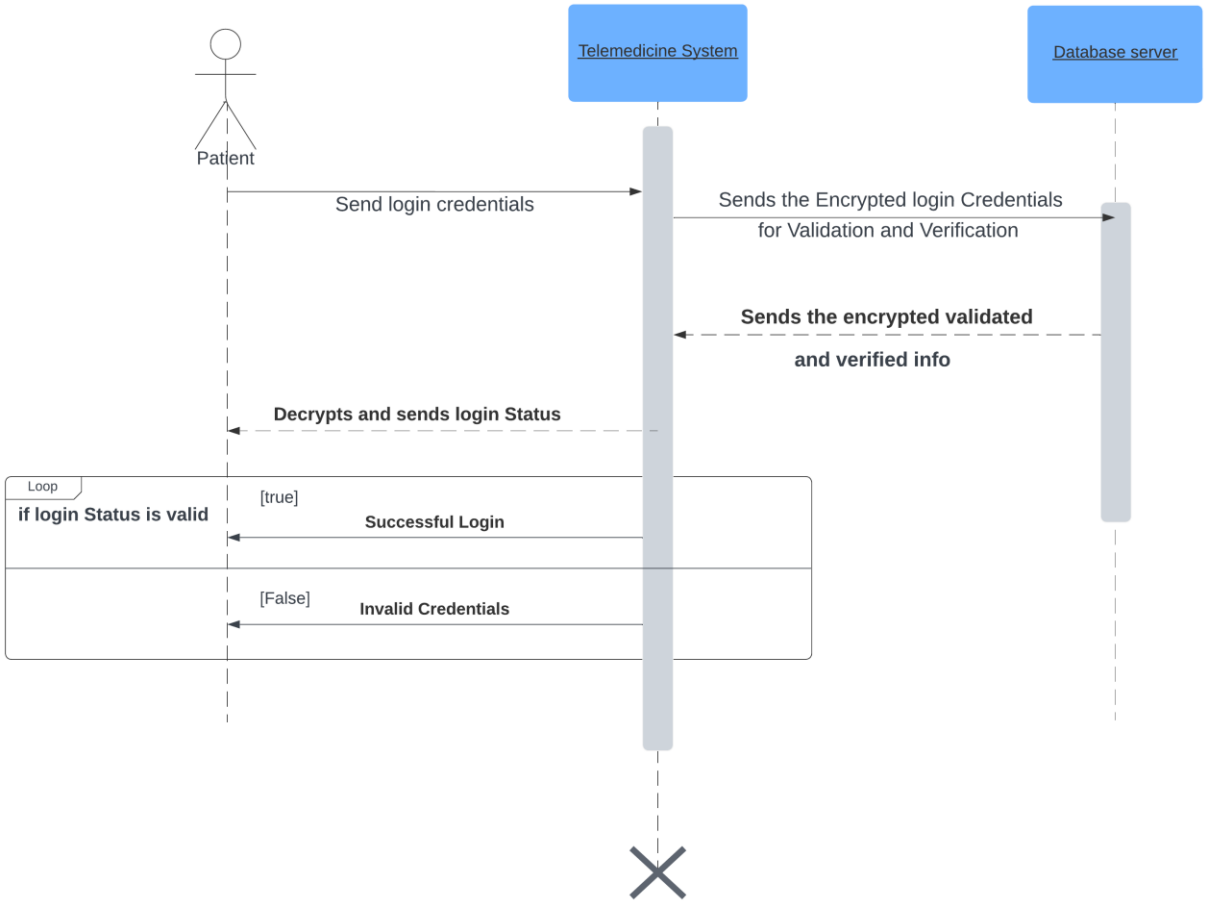

Figure 8: Sequence Diagram for authentication

Figure 9: Sequence Diagram for appointment booking for consultation.

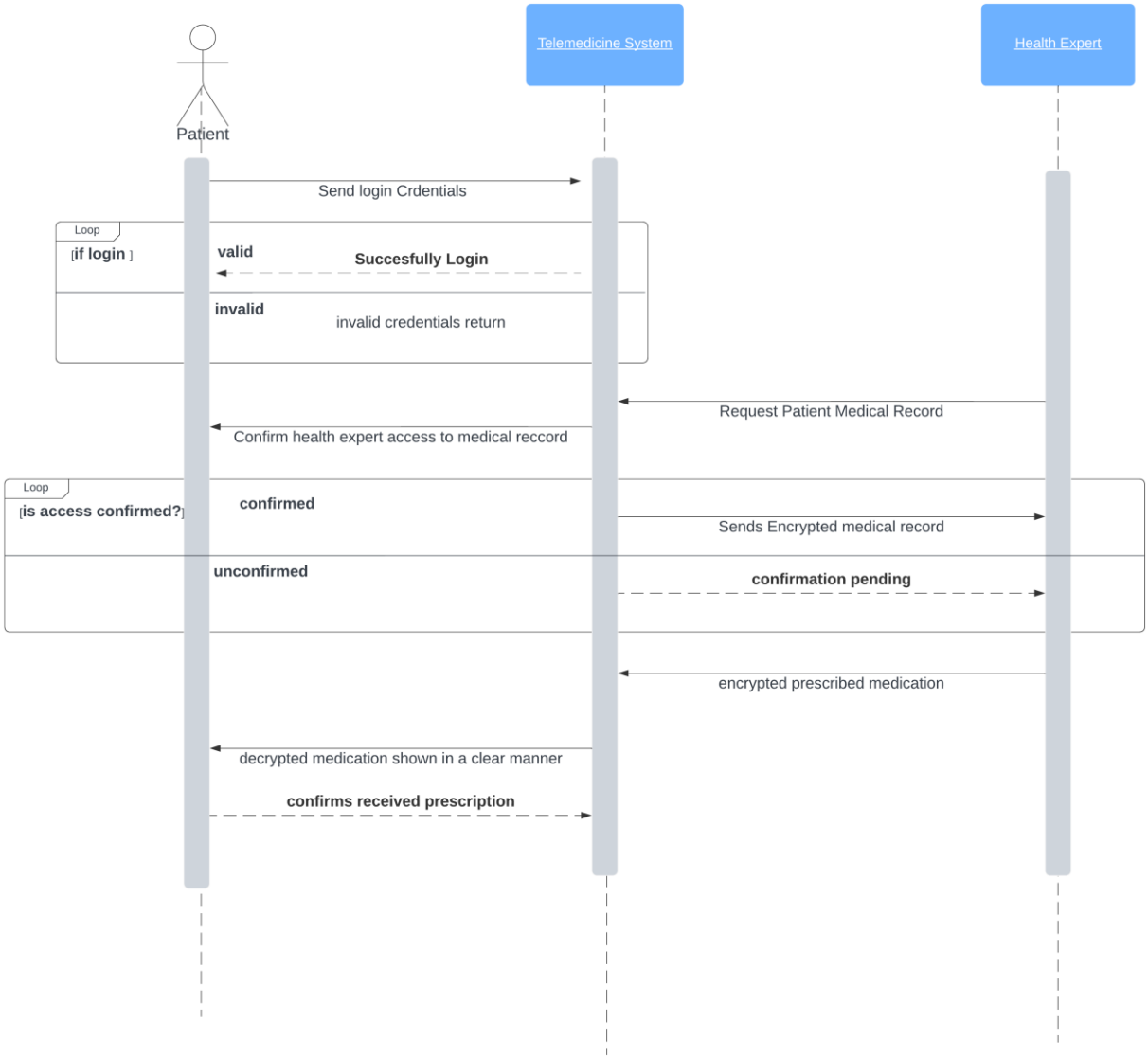

Figure 10: Sequence Diagram for prescription issuing.

Description of Sequence Diagram:

For Authentication (as on Figure 8)

- The diagram explains the login process for a patient using an intermediate system and a database server.

- The diagram consists of three columns: Patient, Intermediate System, and Database Server.
- The patient sends their login credentials to the intermediate system.
- The intermediate system sends the credentials to the database server for verification.
- If the credentials are valid, the database server sends the encrypted validated and verified info back to the intermediate system.
- If the credentials are invalid, the intermediate system sends an invalid credentials status to the patient.

HIPAA is a set of standards that regulate the use and disclosure of protected health information (PHI) [1]. The login process in the diagram is compliant with HIPAA standards, as it does not disclose any PHI.

The sequence diagram shows the process of sending a login credentials to a database server in a telemedicine system in accordance with the HIPAA standard.

Sequence:

1. The patient sends the encrypted login credentials to the database server for validation and verification.
2. The database server decrypts and validates the login credentials.
3. If the login credentials are valid, the database server sends the encrypted validated and verified info to the patient.
4. The patient decrypts and sends the login status.
5. If the login status is valid, the patient is successfully logged in. Otherwise, the patient is notified that the login credentials are invalid.

HIPAA compliance:

The sequence diagram complies with the HIPAA standard in the following ways:

- The login credentials are encrypted before they are sent over the network. This protects the confidentiality of the patient's information in case of a data breach.
- The login credentials are validated and verified by the database server before the patient is granted access to their account. This helps to ensure that only authorized individuals can access patient information.
- The patient's login status is encrypted before it is sent back to the patient. This protects the privacy of the patient's account status.

Overall, the sequence diagram shows a secure and compliant process for sending login credentials to a database server in a telemedicine system in accordance with the HIPAA standard.

For booking Appointments (as on figure 9)

The sequence diagram you provided shows the following process for a patient to book an appointment with a healthcare expert in a telemedicine system in accordance with the ISO 27001 standard:

1. The patient sends their login credentials to the telemedicine system.
2. The telemedicine system validates the patient's login credentials.
3. If the login credentials are valid, the telemedicine system displays the patient's schedule.
4. The patient selects an appointment type.
5. The telemedicine system displays a list of available healthcare experts for the selected appointment type.
6. The patient selects a healthcare expert and a schedule.
7. The telemedicine system sends a payment request to the patient.
8. The patient makes the payment.
9. The telemedicine system encrypts and secures the appointment details.
10. The telemedicine system confirms the appointment with the patient.

ISO 27001 compliance:

The sequence diagram complies with the ISO 27001 standard in the following ways:

- The patient's login credentials are validated before they are granted access to the telemedicine system. This helps to ensure that only authorized individuals can access the system.
- The appointment details are encrypted and secured to protect the confidentiality of the patient's information.
- The payment process is secure and compliant with industry standards.

Additional security considerations:

In addition to the security measures shown in the sequence diagram, the telemedicine system should also implement the following security controls to comply with the ISO 27001 standard:

- Access control: The telemedicine system should implement access control measures to restrict access to the system and its data to authorized individuals.
- Audit trails: The telemedicine system should log all activities performed in the system to create an audit trail. This audit trail can be used to investigate any suspicious activity or security incidents.
- Risk management: The telemedicine system should implement a risk management process to identify, assess, and mitigate security risks.
- Security awareness training: All employees who have access to the telemedicine system should receive security awareness training. This training should cover topics such as password security, phishing attacks, and social engineering.

By implementing these security measures, the telemedicine system can help to protect the confidentiality, integrity, and availability of patient information in accordance with the ISO 27001 standard.

For prescription (Figure 10)

Sure, here is an updated sequence diagram for the prescribing of medication by a doctor to a patient in a telemedicine system in accordance with the HIPAA and ISO 27001 standards:
Sequence:

1. The patient logs into the telemedicine system.
2. The doctor requests the patient's medical record to prescribe medication.
3. The patient confirms access.
4. The doctor prescribes the medication to the patient.
5. The telemedicine system encrypts the prescription.
6. The telemedicine system sends the encrypted prescription to the patient's pharmacy.
7. The patient's pharmacy decrypts the prescription and dispenses the medication to the patient.

HIPAA and ISO 27001 compliance:
The sequence diagram complies with the HIPAA and ISO 27001 standards in the following ways:

- The patient authenticates to the telemedicine system before requesting a prescription. This helps to ensure that only authorized individuals can request prescriptions.
- The telemedicine system encrypts the prescription before it is sent to the patient's pharmacy. This protects the confidentiality of the patient's prescription information in case of a data breach.
- The patient's pharmacy decrypts the prescription before it is dispensed. This protects the privacy of the patient's prescription information.
- The telemedicine system and the patient's pharmacy implement access control measures to restrict access to prescription information to authorized individuals.
- The telemedicine system and the patient's pharmacy log all activities performed to create audit trails. These audit trails can be used to investigate any suspicious activity or security incidents.
- The telemedicine system and the patient's pharmacy implement risk management processes to identify, assess, and mitigate security risks.
- The telemedicine system and the patient's pharmacy provide security awareness training to all employees who have access to prescription information.

Overall, the sequence diagram shows a secure and compliant process for the prescribing of medication by a doctor to a patient in a telemedicine system in accordance with the HIPAA and ISO 27001 standards.

2. UI DESIGN

Images to illustrate the user interface of MEDIK are shown in the figures below

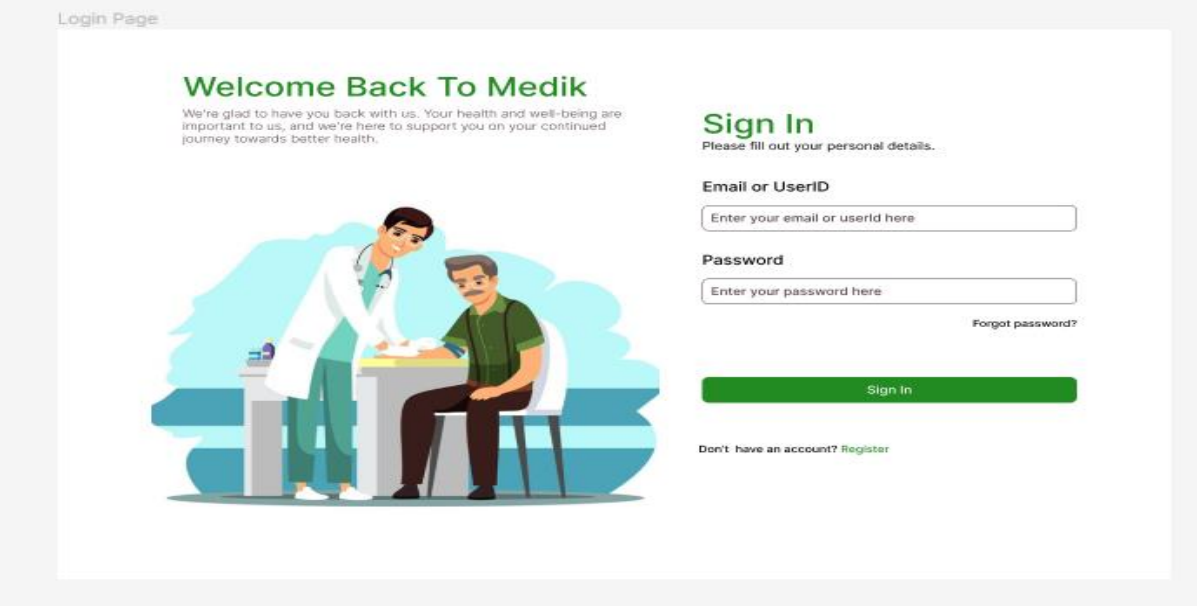

Figure 11: Desktop View Login page for MEDIK

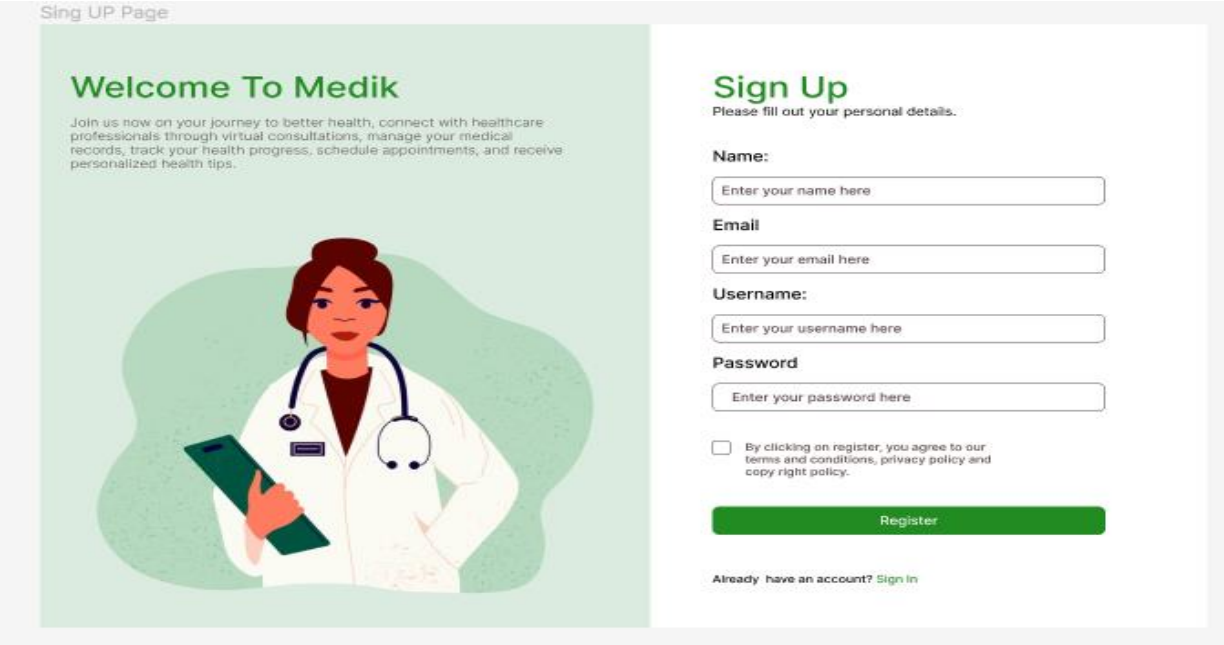

Figure 12: Desktop View Registration page

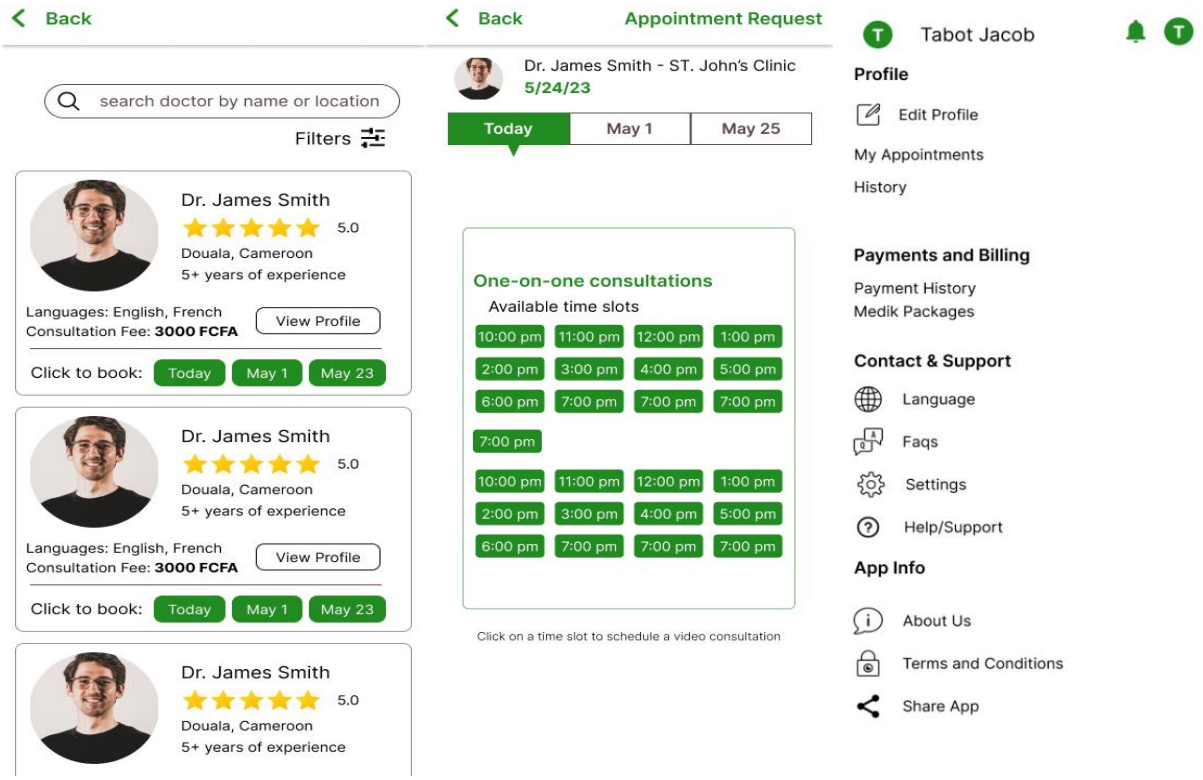

Figure 13: Mobile View of Book appointment with doctor page with Navigation at the end

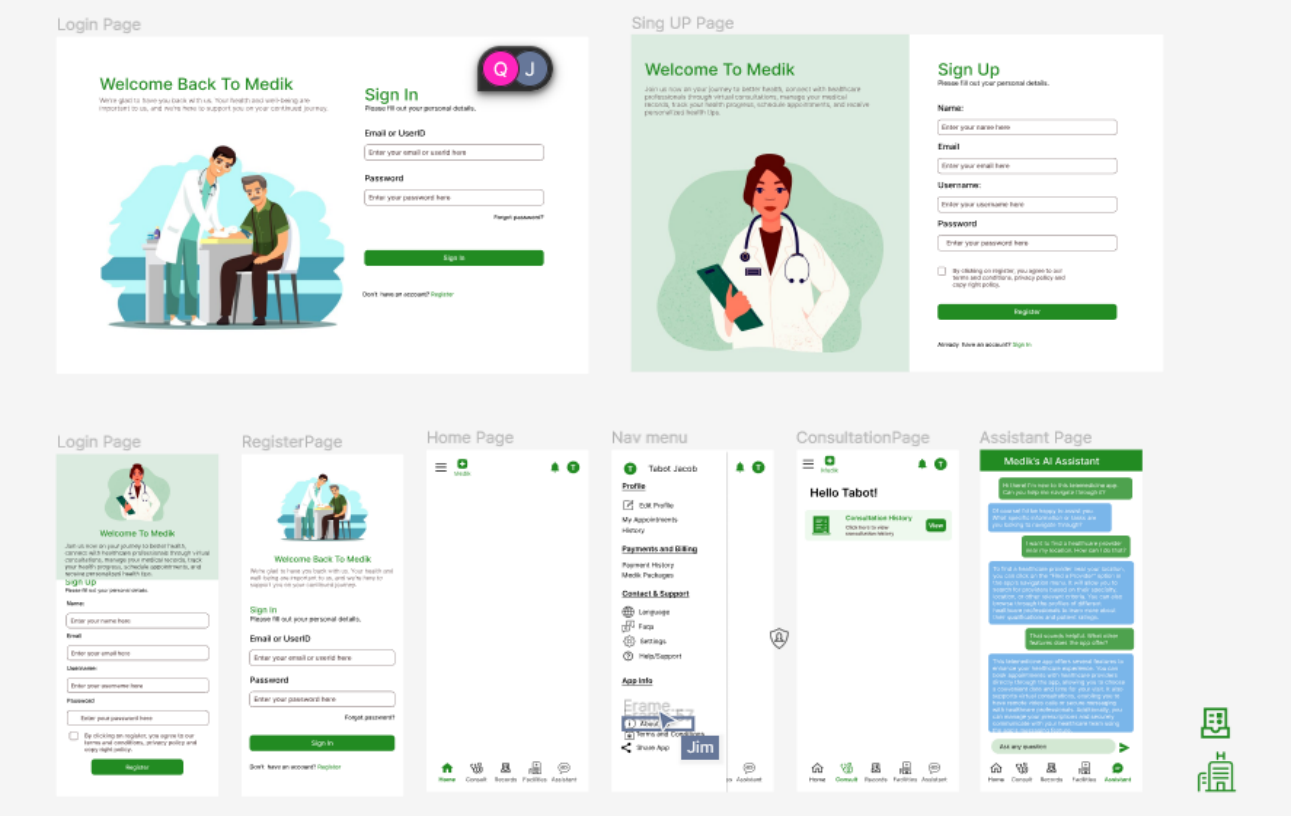

Figure 14: Multiple page view of Desktop and Mobile version

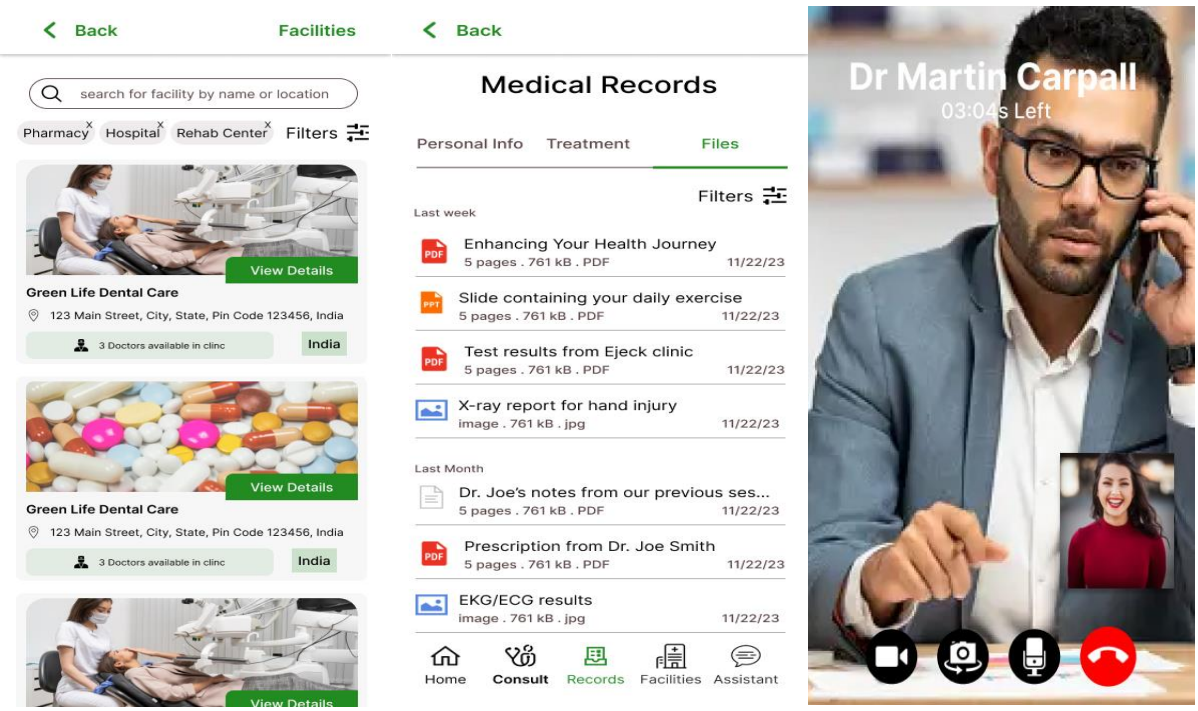

Figure 15: Mobile view of Search health facility, medical record and Consultation sessions via audio respectively

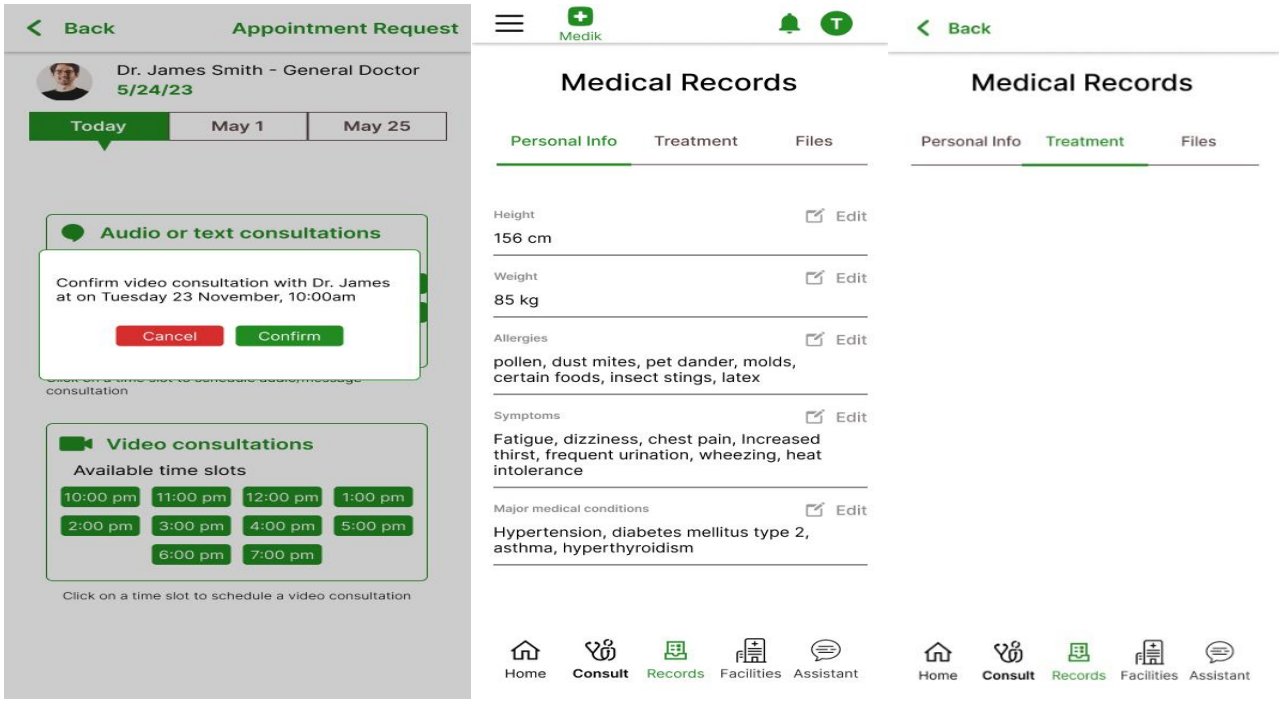

Figure 16: Mobile view of Appointment, medical record respectively

3. DATABASE DESIGN

The design phase involves producing a conceptual and a logical model which will define detailly the database structure.

- **Conceptual Model:**
  This phase of the design focused on high-level structure of data, entities and their relationship in the telemedicine application. An ER model was produced to illustrate this as shown by the figure below

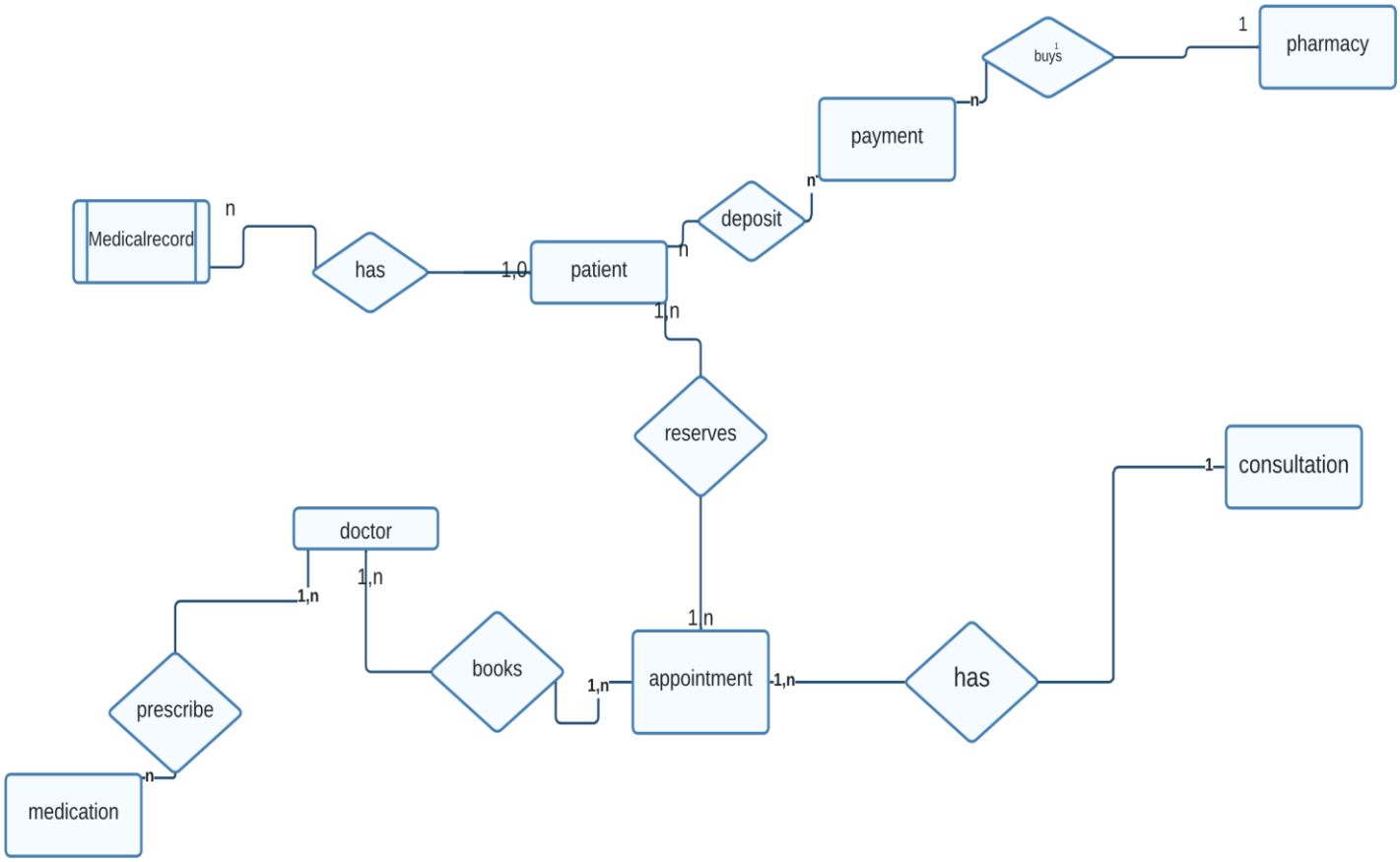

Figure 17: ER model for database

Entities:

- Patient: Represents individuals receiving medical care.

- **Doctor**: Represents medical professionals providing care.

- **Pharmacy**: Represents the entity where medications are dispensed.

- **Medication**: Represents the drugs prescribed to patients.

- **Appointment**: Represents scheduled consultations between patients and doctors.

- **Medical Record**: Represents the documentation of a patient's medical history and treatments.

Relationships:

- **Patient has Medical Record**: Indicates that each patient possesses a medical record.

- **Patient reserves Appointment**: Shows that patients can schedule appointments.

- **Patient buys Medication**: Reflects the transaction where patients purchase prescribed medication.

- **Doctor prescribes Medication**: Indicates that doctors are responsible for prescribing medication to patients.

- **Doctor books Appointment**: Shows that doctors can schedule appointments with patients.

- **Pharmacy has Medication**: Indicates that pharmacies stock various medications.

- **Appointment has Consultation**: Represents the actual consultation event within an appointment.

ER Model Components:

- **Entities**: Represented by rectangles, they are objects that exist independently and have a unique identity.

- **Relationships**: Represented by lines, they describe how entities are related to one another.

- **Attributes**: Not explicitly shown in your description, but typically represented by ovals, they are properties or characteristics of entities.

- **Logical Model:**
  This phase of the design focused on deep-level structure of data, database tables and their relationship in the telemedicine application. This is what to be implemented by the developers. It is illustrated using the Relational Schema as shown below

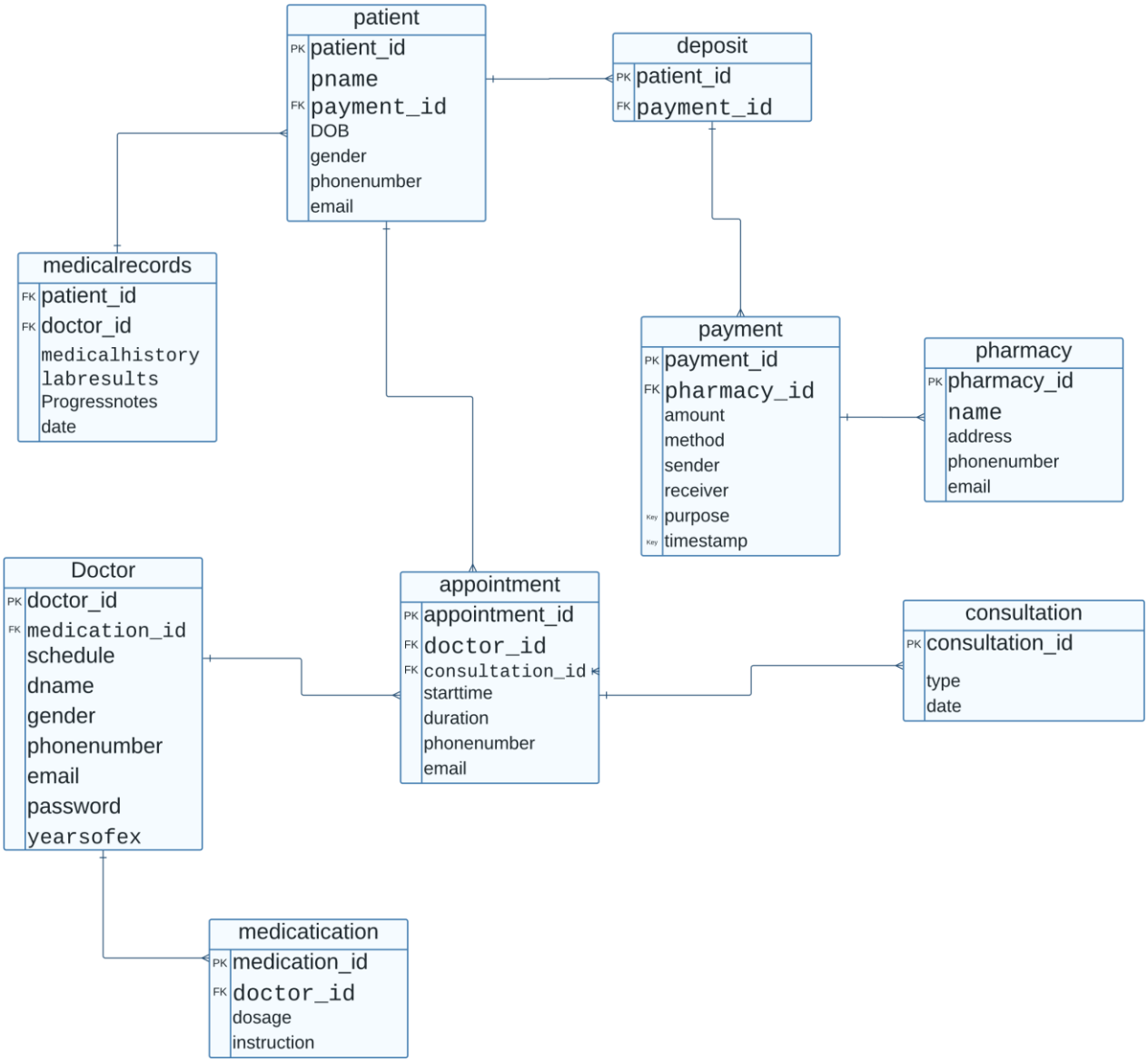

Figure 18: Logical Model for database

V. Software Quality Model for MEDIK Telemedicine App

The purpose of this software quality model is to define the quality attributes and requirements for the MEDIK telemedicine app. This model will be used to guide the development and testing of the app to ensure that it meets the needs of its users and stake holders. AdEQUATE and ISO/IEC 25010 where the models chosen for the quality development of our app.

The AdEQUATE Software Quality Evaluation Model is a quality in use model that focuses on assessing the quality of software from the user's perspective. It was developed in the 1990s and has been used to evaluate a wide range of software applications, including telemedicine apps. The model is based on the following four dimensions:

Adequacy: The software should meet the user's needs and expectations.

Usability: The software should be easy to learn, use, and understand.

Performance: The software should be responsive and efficient.

Reliability: The software should be free from defects and should be able to handle unexpected situations.

The AdEQUATE Model is a useful tool for evaluating the usability of telemedicine apps. It can help to identify areas where the app is difficult to use and can provide suggestions for improvement. The model is also relatively easy to apply, making it a good choice for organizations that are new to software quality assessment.

ISO/IEC 25010 Quality Models

The ISO/IEC 25010 Quality Models are a family of quality models that provide a comprehensive framework for assessing the quality of software systems. The models were developed by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC). The models are based on the following eight characteristics:

Functionality: The software should fulfill its specified requirements.

Reliability: The software should be free from defects and should be able to handle unexpected situations.

Usability: The software should be easy to learn, use, and understand.

Efficiency: The software should make optimal use of system resources.

Maintainability: The software should be easy to modify and maintain.

Portability: The software should be easy to transfer from one hardware or software environment to another.

Security: The software should protect user data from unauthorized access, use, disclosure, disruption, modification, or destruction.

Testability: The software should be designed in a way that makes it easy to test.

The ISO/IEC 25010 Quality Models are a valuable tool for assessing the overall quality of telemedicine apps. They can help to identify areas where the app is not meeting its requirements and can provide suggestions for improvement. The models are also a good choice for organizations that are looking to implement a quality management system for their software development process.

Comparison of the Two Models

The AdEQUATE Model and the ISO/IEC 25010 Quality Models are both useful tools for assessing the quality of telemedicine apps. However, they have different strengths and weaknesses. The AdEQUATE Model is particularly useful for evaluating usability, while the ISO/IEC 25010 Quality Models provide a more comprehensive framework for assessing software quality.

VI. SOFTWARE TESTING

Software testing is the process of checking the quality, functionality, and performance of a software product before launching. To do software testing, testers either interact with the software manually

or execute test scripts to find bugs and errors, ensuring that the software works as expected. Software testing is also done to see if business logic is fulfilled, or if there are any missing gaps in requirements that need immediate tackles. Testing is categorized into two as seen below:

Testing Categories

1) **Static Testing**

Static testing is an approach to testing the software application without executing the actual code. This is performed at the early stage of development to identify the issues in the project documents in multiple ways, namely reviews, walkthroughs, and inspections.
Static testing is analysing the project specifications at the initial stage of development. If defects are detected at the early stage cost of the testing is reduced. With static testing, we can identify ambiguities in project documentation, misunderstandings of requirements, or flaws in the requirement and design issues. Static testing is required to improve development productivity. Coding mistakes can be detected and rectified at the initial stage of development by static testing.

Types of Static Testing

Static testing can be done mainly by manual and automated methods as described below.

- **Design Testing:**

  Design testing is the process of checking designs in terms of requirements and content.

  Found below are the various types of design testing which will be implemented at the design stage of our development process.

- **UI testing:**

- **Find differences between requirements and designs** – sometimes after reviewing requirements it can happen that the reviews don't end up in the designs e.g. according to requirements, view X should contain the name and surname field, and the design includes the name and date of birth fields.

- **find typos in texts** – developers copy text from designs, and mistakes happen all the time. Detecting a typo at this stage means that you won't have to correct it directly in the (semi) finished application.

- **find deficiencies in the requirements** – e.g. according to the designs, the view should contain five fields (because of reasons), and the description specified only three (because it didn't take "the reasons" under consideration).

- **find deficiencies in designs** – it can happen the other way round too, when the design doesn't contain all the elements described in the requirements, e.g. missing buttons.

Manual Methods of Static Testing

Detailed evaluation of project-specific documents done manually performed by different project members like architects, designers, managers, moderators, and reviewers.

Types of manual methods:

1. Inspections
2. Walkthroughs
3. Informal Reviews
4. Technical Review

- **Inspections**: These are the most formal way of static testing, a dedicated moderator initiates the process and conducts a complete formal review by scheduling a meeting. Proper documentation has been prepared before and after the meeting, and peer reviews will happen to understand the product. observations and issues are documented as issue logs and the moderator will do a follow-up and schedule meetings accordingly till the closure. The Author is responsible for resolving the identified defects and improving the quality. Reviewers are responsible for checking the documents and reviewing the defects during the inspection process. The manager will take care of the planning for the inspection process and ensures whether the goal has been met or not.

- **Informal Reviews:** These reviews are performed mainly as peer reviews, management participation is not mandatory. There is no need for any report generation after the meeting. Feedback from the team will be captured and implemented wherever necessary.

- **Technical Review**: A formal way of review conducted by a trained moderator or by a technical expert. Potential defects can be identified by the document review by architects, designers, and users who focus on the content of the documentation. This is beneficial to understand the progress of the project. The major goal of this method is to focus on technical concept corrections and identifying alternatives if required, to improve the understanding of technical information among the team members.

- **Walkthrough:** This is not a formal process, usually the meeting is led by the author of the documentation, aiming for a common understanding of the proposed solutions and taking the feedback. It's like a knowledge transfer and examining the content of the documentation

with the participants. This is useful for high-level documents like requirement specifications.

Automation Method of Static Testing

The automation method of static testing is nothing but code analysis by some tools. Source code analysis or debugging is done by different tools, and by the developers. It is called static analysis.

- **Static Analysis:** Static analysis is the code analysis performed to understand the code structure and coding standards. It's a debugging performed to examine the source code without executing the program. This is helpful for developers to ensure identifying programming errors, syntaxx errors, security issues and performance issues, and coding standard violations. Different tools like Pycharm, Checkstyle, and SourceMeter are available to analyse data flows and control flows in the source code in the targeted programming language. Some examples of these tools are mentioned here.

Types of automation methods or static analysis

- **Control flow analysis :** This is a static code analysis techniques used to check the control flow of the programming language and it is defined in control flow graph and verifies the functions or modules and subroutines to check the process flows.

- **Data flow analysis:** **T**his method used to verify the data structures and definitions of variables and their usage without executing the code.

- **Failure analysis :** This method is used to identify the causes of design failures and condition failures**,** incorrect behaviour of the modules etc.

- **Interface analysis:** This method is used to analyze the different interfaces used the program model and to identify the integrations and simulations.

2. Dynamic Testing

Dynamic testing refers to analyzing code's dynamic behavior in MEDIK. In this type of testing, input is given and gotten as per expectation through executing a test case. The test

cases can be run manually or through an automation process, and the software code must be compiled and run for this.

The main purpose of dynamic testing is to validate the software and ensure it works properly without any faults after the installation.

Types of Dynamic Testing

1. Functional Testing

It checks the functionality of an application as per the requirement specifications. Each module needs to be tested by giving an input, assuming an output, verify the actual result with the expected one. Further, this testing divides into four types –

- **Unit testing:** It tests the code's accuracy and validates every software module component. It determines that every component or unit can work independently.
- **Integration testing:** It integrates or combines each component and tests the data flow between them. It ensures that the components work together and interact well.
- **System Testing:** It makes to test the entire system. So it's also known as end-to-end testing. Work through all the modules and check the features so that the product fits the business requirements.
- **User acceptance testing:** Customers perform this test just before releasing the software in the market to check the system meets the real user's conditions and business specifications.
-

2. Non-Functional Testing

It implies checking the quality of the software. That implies testing whether the software meets the end users' requirements. It expands the product's usability, maintainability, effectiveness, and performance. Hence it reduces the manufacturing risk for the non-functional components.

- Performance testing**:** In this testing, we would check how MEDIK can perform in different conditions. Which are;
  - **Speed Testing:** The time requires loading a web page with all components- texts, images, videos, etc.
  - **Load Testing:** Test MEDIK'S stability when users increase gradually. That means, by this test, you can check the system's performance under variable loads.
  - **Stress Testing:** It sets a limit on which the system breaks due to a sudden increase in users' number.
- **Security Testing:** Security testing reveals the vulnerabilities and threats of a system. Also, it ensures that the system is protected from unauthorized access, data leakages, attacks, and other issues. Then fix the issues before deployment.
- Usability testing**:** This test checks how easily an end user can handle a software/system/application. Additionally, it will check the app's flexibility and capability to reach the user's requirements.

VII. Benefits of MEDIK To The Cameroonian Society

MEDIK will offer a number of benefits including:

- Increased access to healthcare: MEDIK will provide patients with access to healthcare services that they might not otherwise have. This is especially beneficial for patients in rural or underserved areas.
- Convenience: MEDIK will allow patients to see their doctor from the comfort of their own home. This can save patients time and money.
- Affordability: MEDIK will be more affordable than traditional in-person care.

- Improved quality of care: MEDIK will improve the quality of care by providing patients with more timely and convenient access to healthcare services.

VIII. CONCLUSION

In conclusion, MEDIK MEDIK offers a user-friendly interface that enables patients to conveniently access virtual consultations with licensed healthcare professionals from the comfort of their homes. Through secure video calls and messaging systems, patients can receive medical advice, diagnoses, and even prescriptions, saving time and eliminating the barriers of distance and transportation.

With a strong focus on patient privacy and data security, our app complies with international professional standards such as HIPAA and GDPR. We have implemented robust encryption protocols and stringent access controls to ensure the confidentiality and integrity of patient information, instilling trust and confidence in our users.

presents a comprehensive solution that addresses the evolving needs of healthcare delivery in the digital age. By leveraging the power of technology, we aim to revolutionize the way patients and healthcare providers connect and interact, ultimately enhancing access to quality healthcare services.

Furthermore, MEDIK integrates seamlessly with existing healthcare systems through adherence to HL7 standards, enabling efficient sharing and exchange of electronic health information. This interoperability enhances care coordination and streamlines the healthcare workflow, leading to improved patient outcomes and operational efficiency.

Our team of experienced healthcare professionals, software developers, and user experience designers are collaborating for the production of MEDIK that prioritizes user satisfaction and clinical effectiveness. We have incorporated evidence-based guidelines from reputable organizations like the ATA and WHO to ensure that our app adheres to the highest industry standards.

In summary, MEDIK offers a transformative solution that empowers patients and healthcare providers alike. By embracing technology and adhering to international professional standards, we are redefining healthcare delivery, making it more accessible, efficient, and patient-centric. Join us on this journey as we shape the future of healthcare through our innovative telemedicine app.

XI. REFERENCE:

1) **Google Bard AI : https://bard.google.com/** visited lastly on the 07/011/2023

2) Health Insurance Portability and Accountability Act (HIPAA)

- Book: ***HIPAA: A Practical Guide for Covered Entities and Business Associates*** **by Laurie A. Morgan**

  Author: **Laurie A. Morgan**

  Publication Date: **2022**

3)International Organization for Standardization (ISO) 13485:2016

- Book: ***ISO 13485:2016 Quality Management Systems - Medical Devices - Application of ISO 9001:2015 for Regulatory Purposes*** **by John Woodcock**

  Author: **John Woodcock**

  Publication Date: **2018**

5) International Electrotechnical Commission (IEC) 60601-1

- Book: ***IEC 60601-1:2005 Medical electrical equipment - General requirements for safety*** **by International Electrotechnical Commission**

  Author: **International Electrotechnical Commission**

  Publication Date: **2005**

6) Medical Device Directive (MDD)

- Book: ***The Medical Device Directive - A Practical Guide for Medical Device Manufacturers*** **by Peter Chapman**

  Author: **Peter Chapman**

  Publication Date: **2013**

7) Medical Device Regulation (MDR)

- Book: ***The Medical Device Regulation - A Practical Guide for Medical Device Manufacturers*** **by Peter Chapman**

  Author: **Peter Chapman**

  Publication Date: **2020**

8) World Health Organization (WHO) Recommendations for Telemedicine

- Book**:** ***WHO Recommendations for Telemedicine: Digital technologies for health*** **by World Health Organization**

  Author: **World Health Organization**

  Publication Date: **2021**

8) International Telecommunication Union (ITU) Guidelines for Telemedicine

- Book: ***ITU Guidelines for Telemedicine: Use of telecommunication/information and communication technologies (ICT) for the delivery of health care services at a distance*** **by International Telecommunication Union**

  Author: **International Telecommunication Union**

  Publication Date: **2015**

9) System Design Resources

- STARUML Software
  - Draw IO

Author: **International Telecommunication Union**

10) Publication Date: **2015**

11) App and browser testing made easy (browserStack) available at: https://www.browserstack.com/guide Visited lastly 17/11/23

12) https://katalon.com/resources-center/blog/software-testing

13) Telemedicine Online Platform Database Design available at : https://itsourcecode.com/free-projects/database-design-projects/telemedicine-online-platform-database-design/#google_vignette

14) Sources: en.wikipedia.org/wiki/ISO/IEC_15288