Hospital Management System Requirements Specification

May 24, 2024

Prepared by **Group C2**:

Gerjan Haxhija

Kevin Rroga

Miki Belegu

Xhoni Dërvishi

Received by:

M.Sc. Igli Draçi

Table of Contents

1. Executive Summary

1.1 Project Overview

Managing hospital operations in today's healthcare environment can be a significantly difficult undertaking. Paper files and other ineffective record-keeping practices have become outdated and inefficient. This is where our Hospital Management System (HMS) can help modernize things in this situation.

With technology advancing significantly and rapidly, hospitals are finding that they must adopt digital solutions. However, a lack of infrastructure and resources makes the development of these digital systems difficult in many areas, particularly in developing nations, such as Albania.

A robust HMS might completely revolutionize the way hospitals run. Managing patient records, streamlining administrative work, and improving communication between healthcare providers and their patients are only some of the benefits, with others being in patient safety, privacy, and even general healthcare standards and hospital efficiency.

Given the effect of a HMS on people’s healthcare, great care must be given to a number of considerations in the development of our project, such as security, interoperability, and scalability. This will be a complex process that will involve collaboration between software engineers, healthcare professionals, and administrators.

Yet, despite the challenges, the advantages of a well-designed HMS are greatly important. By investing in the development and implementation of an advanced HMS, hospitals can position themselves at the forefront of healthcare innovation, enhancing healthcare standards and patient satisfaction.

1.2 Purpose and Scope of this Specification

1.2.1 Scope

1.2.1.1 Patient Management:

- Patient registration and demographic data capture.
- Electronic Health Records (EHRs) for storing and managing patient medical history.
- Appointment scheduling and management.
- Secure patient portal for accessing personal medical records.

1.2.1.2 Clinical Functionality:

- Doctor's dashboard for viewing patient EHRs, recording observations, and prescribing medications.
- Order management for requesting lab tests, imaging procedures, etc.
- Integration with basic medical equipment (optional).

1.2.1.3 Administrative Management:

- User access control with different levels of permission.
- Patient billing and insurance management (optional).
- Inventory management for medical supplies.
- Reporting and analytics to track key performance indicators (KPIs).

1.2.2 Out of Scope

- Features requiring high-bandwidth internet connectivity (e.g., real-time video conferencing).
- Integration with complex medical devices requiring specialized interfaces.
- Advanced features like telemedicine consultations (may be considered for future versions).
- Financial transactions and payment processing (can be integrated with a separate system).

1.2.3 Intended Audience

- **Software Developers:** This document provides a detailed description of the HMS functionalities and technical requirements for development.
- **Hospital receptionists:** This document outlines the benefits of the HMS for improving hospital operations and resource allocation.
- **Medical Professionals:** This document provides an overview of how the HMS will streamline patient care and medical record management.
- **End Users (Patients):** This document will be supplemented by user guides explaining how to access and utilize the patient portal features

2. Product/Service Description

Albania's public healthcare system faces significant challenges in providing its services because of inadequate funding and faulty or undeveloped systems for health departments. The various health facilities, which are the most crucial places for citizens to seek healthcare, face different and important obstacles, notably their dependence on paper-based documentation.

To address these challenges, there's a growing demand for digital solutions that streamline record-keeping processes and enhance overall healthcare management efficiency. Our Hospital Management System (HMS) offers a solution for all health facilities.

Our HMS revolutionizes traditional paper-based record-keeping by digitizing patient records, scheduling appointments, and tracking medical histories. By transitioning to our digital solution, clinics can overcome the restrictions of paper-based documentation, improve information accessibility, and streamline communication between healthcare providers and patients.

Designed to bridge the gap between obsolete practices and modern digital healthcare administration, our HMS optimizes workflows while ensuring data accuracy and security. By enhancing effectiveness and accessibility, our HMS aims to contribute to the delivery of proficient healthcare services within the context of our developing country’s healthcare systems.

2.1 Product Context

The proposed Hospital Management System (HMS) is designed as an independent and self-contained platform that primarily serves the needs of clinics, community health centers, and hospitals. It is not directly related to specific existing healthcare institutions but rather provides a comprehensive solution that can be implemented in various healthcare settings. The system is designed to interface primarily with its internal components, such as modules for patient records management, appointment scheduling, medical examinations, and administrative tasks.

While the HMS is designed as an independent system, it can offer the option for exporting patient data in standardized formats (e.g., ICD-10, HL7, FHIR) to facilitate data exchange with other healthcare institutions that deploy compatible systems. This allows for seamless transfer of patient records when patients transition between different healthcare providers or institutions, such as when they move from one clinic to another or require care at a hospital.

2.2 User Characteristics

There are three types of users that will interact with the system:

1. **Patient:**
    - View medical session information and download it
    - Contact doctor
    - Leave feedback
    - View main page
    - Log in and out
    - View personal profile
    - Change password
2. **Doctor:**
    - See a list of their patients
    - See a list of their patients waiting to be examined
    - Search in their patients' list
    - Fill visit session forms for each patient waiting to be examined (must first be assigned by receptionist)
    - View profiles of their patients
    - View examination records of their patients
    - Download examination record of a patient
    - View main page
    - Log in and out
    - View personal profile
    - Change password
3. **Receptionist:**
    - View the list of all patients
    - View the profiles of patients
    - View the examination records of all patients
    - Edit personal information of patients
    - Export the files of a patient (send to another hospital)
    - Add/delete a patient
    - Create a visit session and allow the doctor to fill the form for the current examination
    - Search a patient
    - View list of doctors and their information

- Add/delete a doctor
- Search a doctor
- Edit personal information of doctors
- View main page
- Log in and out
- View personal profile
- Change password

2.3 Assumptions

List of assumptions affecting the requirements:

- Availability of Necessary Documentation: It is assumed that the receptionist will have access to and verify all required documents when adding new patients or doctors to the system, ensuring compliance with legal and regulatory standards.
- Confidentiality of Patient Information: The system assumes that patient information is highly confidential and can only be accessed by authorized personnel, such as the receptionist or the patient's family doctor. Exporting patient files is restricted to specific scenarios, such as transferring to another healthcare facility or continuing treatment in a hospital.
- Limited User Permissions: The system assumes that user roles and permissions are predefined, with the receptionist responsible for creating and managing receptionist profiles. Only the receptionist has the authority to add, delete, or modify patient and doctor information, ensuring data integrity and security.
- User Training and Expertise: It is assumed that doctors and receptionists will receive adequate training to use the software effectively, minimizing errors and misconceptions during system operation. Users with read-only access, such as patients, do not require training as their interaction is limited to viewing their own information.
- Equipment and Connectivity: The system assumes that all doctors and receptionists within the polyclinic have access to a computer connected to the internet, as the web-based application relies on internet connectivity for seamless access and data management.
- Data Integrity and Security Measures: The system assumes that safeguards are in place to maintain data integrity and prevent unauthorized access. For example, doctors are responsible for the accuracy of medical examination records, and once finalized, these records cannot be altered to maintain a reliable audit trail.
- Compliance with Legal and Ethical Standards: The system assumes adherence to legal and ethical standards regarding patient confidentiality, data handling, and user responsibilities. Any changes to these standards may require corresponding updates to the system's requirements and functionalities.

2.4 Constraints

2.4.1 **Hardware Requirements:** For optimal use of the HMS, all staff (receptionists and doctors) will require access to dedicated workstations during working hours. This ensures they can utilize the web application effectively.

2.4.2 **User Training and Competency:** Successful system implementation hinges on user competency. Receptionists and doctors will undergo training to understand their specific functionalities and responsibilities within the HMS. This empowers them to work efficiently, avoid interfering with each other's tasks, and most importantly, safeguard patient privacy. Using the application correctly will maximize its performance and benefits for all users.

2.4.3 **Internet Connectivity:** As a web-based application, the HMS relies on a stable internet connection to function optimally. Internet access is crucial for various tasks, including:

- Retrieving data from the central database.
- Enabling patient access to the portal (if applicable).
- Facilitating secure data backups to cloud storage.
- Enabling data exchange with other institutions (exporting/importing patient records).
- Distributing software updates and bug fixes.

2.5 Dependencies

2.5.1 The new product will require the receptionist to be registered and available in the system for actions such as adding new patients or doctors.

2.5.2 The availability of the receptionist is essential for tasks like deleting existing patients or doctors from the system.

2.5.3 Updates to patient or doctor profiles are dependent on the presence and availability of the receptionist in the system.

2.5.4 The export and transfer of a patient's medical file to another polyclinic or hospital are contingent upon the receptionist being available to perform the action.

2.5.5 Doctors can only view the profiles and examination records of patients assigned to them within the polyclinic.

2.5.6 Patients can access their latest medical visit records only after the doctor has filled out and saved the examination form.

2.5.7 Patients cannot undergo examinations unless a doctor is assigned to them within the system.

3. Requirements

3.1 Functional Requirements

Req #	Functional Requirement	Comment	Priority	Date
FR_01	Multi-level user access system with distinct interfaces for Receptionists, Doctors, and Patients.	Ensures users only access functionalities relevant to their role.	1	25-03-2024
FR_02	Secure user accounts with strong passwords changeable by the user (usernames uneditable).	Protects patient data and ensures user accountability.	1	25-03-2024
FR_03	Receptionist can view a complete list of all patients.	Enables efficient patient management.	1	25-03-2024
FR_04	Receptionist can search for specific patients using various criteria.	Facilitates quick retrieval of patient information.	2	25-03-2024
FR_05	Receptionist can add new patients to the system.	Enables registration of new patients at the polyclinic.	1	25-03-2024
FR_06	Receptionist can edit existing patient information (excluding usernames).	Allows for updating patient details as needed.	1	25-03-2024
FR_07	Receptionist can delete patient accounts (with appropriate safeguards).	Enables removal of inactive or duplicate patient records.	1	25-03-2024
FR_08	Receptionist can view a patient's full profile, including medical history.	Provides comprehensive patient information for informed decision-making.	1	25-03-2024
FR_09	Receptionist can export patient medical records in a downloadable format (e.g., PDF).	Enables sharing of patient medical records with other institutions if needed.	1	25-03-2024
FR_10	Receptionist can view a list of all doctors associated with the polyclinic.	Supports efficient staff management.	2	25-03-2024
FR_11	Receptionist can search for specific doctors.	Simplifies locating specific doctors within the polyclinic.	1	25-03-2024
FR_12	Receptionist can add new doctors to the system.	Enables onboarding of new doctors at the polyclinic.	1	25-03-2024
FR_13	Receptionist can edit existing doctor information (excluding usernames).	Allows for updating doctor details as needed.	1	25-03-2024
FR_14	Receptionist can delete doctor accounts (with appropriate safeguards).	Enables removal of inactive doctor accounts.	1	25-03-2024
FR_15	Receptionist can view a doctor's full profile.	Provides comprehensive doctor information for reference.	1	25-03-2024
FR_16	Doctor can view a list of all patients assigned to the doctor.	Enables efficient management of assigned patients.	1	25-03-2024

FR_17	Doctor can view or download complete patient medical profiles.	Provides doctors with comprehensive patient information for diagnosis and treatment planning.	1	25-03-2024
FR_18	Doctor can view a list of patients waiting for examination.	Optimizes scheduling and workflow for patient examinations.	1	25-03-2024
FR_19	Doctor can electronically record patient examination findings and treatment plans.	Enables efficient and secure documentation of patient care.	1	25-03-2024
FR_20	Patient can view personal information and medical visit history.	Empowers patients to access and manage their health information.	1	25-03-2024
FR_21	Patient can download medical records as a PDF file.	Enables patients to share their medical records securely with other healthcare providers.	1	25-03-2024
FR_22	Patient can securely contact doctors through the HMS (optional).	Facilitates communication between patients and doctors (if internet connectivity allows).	2	25-03-2024
FR_23	Patient cannot see the profiles of other doctors of the polyclinic.	Protects privacy of other doctors and avoids confusion for patients.	2	25-03-2024
FR_24	Patient can leave feedback about the service.	Enables gathering patient feedback for quality improvement.	2	25-03-2024
FR_25	Receptionist can assign patients to doctors for examinations.	Enables secure management of assigned patients to doctors from receptionists	1	25-03-2024
FR_26	Each user can see the general information of the polyclinic (e.g., contact details, operating hours).	Provides basic information about the polyclinic to all users.	1	25-03-2024
FR_27	The web application has to be responsive.	Ensures optimal user experience across various devices (desktops, tablets, smartphones).	3	25-03-2024
FR_28	Receptionists and doctors can see a spreadsheet of records generated by the system.	Provides data and statistics for analysis of the polyclinic to receptionists and doctors.	3	25-03-2024

3.2 Non-Functional Requirements

3.2.1 Product Requirements

3.2.1.1 User Interface Requirements

Doctor Interface:

Dashboard: Upon logging in, the doctor should be greeted with a customizable dashboard displaying patient appointments, upcoming surgeries, and pending tasks.

Patient Management: Easy access to patient profiles with comprehensive medical histories, lab results, prescriptions, and treatment plans.

Appointment Management: Ability to schedule, reschedule, or cancel appointments with patients. Calendar view for better visualization of appointments.

Prescription Management: Intuitive interface for prescribing medications, including dosage, frequency, and instructions. Integration with drug databases for quick lookup.

Clinical Notes: Structured forms or templates for documenting patient visits, including examination findings, diagnoses, and treatment plans.

Communication: Secure messaging system for communication with patients, nurses, and other healthcare providers.

Reports: Access to patient reports, test results, and medical imaging. Ability to generate customized reports for research or administrative purposes.

Notifications: Instant notifications for new appointments, lab results, or urgent messages.

Patient Interface:

Dashboard: A simplified dashboard displaying upcoming appointments, recent lab results, and prescribed medications.

Appointment Booking: User-friendly interface for scheduling appointments with doctors. Ability to view available time slots and select preferred dates.

Medical History: Access to personal health records, including diagnoses, treatments, and allergies. Clear visualization of lab results and medical imaging.

Prescription Management: View prescribed medications with instructions and refill options. Request refills directly from the interface.

Communication: Secure messaging system to communicate with doctors, nurses, or administrative staff.

Billing and Payments: Access to billing information, invoices, and payment history. Ability to make payments securely online.

Feedback: Provide feedback on doctor visits and overall experience to improve services.

Receptionist Interface:

User Management: Ability to add, remove, or modify user accounts for doctors, patients, and other staff members.

Role-Based Access Control: Define permissions and access levels for different user roles. Customize access to sensitive information.

System Configuration: Manage system settings, including clinic hours, appointment types, and notification preferences.

Financial Management: Monitor revenue, expenses, and financial reports. Manage insurance claims and reimbursements.

Analytics: Access to analytics and insights on patient demographics, appointment trends, and service utilization.

Audit Trail: Track user activities and changes made to patient records for accountability and compliance purposes.

System Maintenance: Perform routine maintenance tasks such as backups, updates, and database management.

General Interface Characteristics:

User-Friendly: Intuitive design with easy navigation and minimal learning curve.

Responsive Design: Compatible with various devices and screen sizes, including desktops, tablets, and smartphones.

Customizable: Allow users to customize preferences, such as language, theme, and dashboard widgets.

Accessible: Ensure accessibility features for users with disabilities, including screen readers and keyboard navigation.

Error Handling: Clear error messages and prompts for incorrect inputs or system failures. Provide guidance on how to resolve issues.

Security: Implement robust authentication mechanisms, data encryption, and role-based access control to protect patient information and comply with privacy regulations.

3.2.1.2 Usability

Learnability:

The system should be intuitive and easy to learn for new users, with minimal training required.

Onboarding tutorials or walkthroughs should be available to guide users through basic functionalities upon their first login.

The interface should employ familiar design patterns and terminology consistent with industry standards to reduce cognitive load.

User Documentation and Help:

Comprehensive user documentation should be provided, covering all aspects of system functionality and usage.

The documentation should be easily accessible from within the system and available in multiple formats (e.g., online help, PDF manuals).

Context-sensitive help should be integrated into the interface, offering relevant guidance and explanations based on the user's current task or location within the system.

Help resources should include step-by-step instructions for common tasks, troubleshooting guides for potential issues, and FAQs to address common queries.

Navigation and Task Completion:

The navigation should be logical and intuitive, with clear menu structures and labels that reflect the user's mental model of the system.

Common tasks should be easily accessible from the main dashboard or through prominent shortcuts, reducing the need for excessive navigation.

The system should support efficient task completion, minimizing the number of steps required to perform routine activities such as scheduling appointments or updating patient records.

Feedback and Error Handling:

The system should provide immediate feedback to user actions, confirming successful operations and alerting users to any errors or issues encountered.

Error messages should be clear, concise, and actionable, guiding users on how to rectify the problem and preventing frustration or confusion.

Validation checks should be implemented to prevent users from entering invalid data or making unintended changes.

Customization and Personalization:

The system should support user preferences and customization options, allowing users to personalize their workspace layout, color schemes, and default settings.

Users should be able to create custom views or filters to streamline their workflow and focus on relevant information.

Accessibility:

The system should adhere to accessibility standards and guidelines, ensuring compatibility with assistive technologies such as screen readers and keyboard navigation.

Text should be legible and resizable, with sufficient color contrast to accommodate users with visual impairments.

Interactive elements should be keyboard accessible, providing alternative methods for users who cannot rely on mouse input.

Performance:

The system should be responsive and performant, with minimal loading times for key functionalities.

Long-running processes, such as generating reports or processing large datasets, should provide progress indicators to keep users informed of the system's status.

3.2.1.3 Efficiency

Performance Optimization:

The system should be optimized for speed and responsiveness, especially during peak usage times.
Database queries, data retrieval, and processing should be efficiently executed to minimize waiting times for users.
Utilize caching mechanisms to reduce redundant data retrieval and improve system performance.

Streamlined Workflows:

Design workflows that minimize unnecessary steps and streamline common tasks, such as patient registration, appointment scheduling, and billing processes.
Implement automation where feasible to reduce manual data entry and repetitive administrative tasks.
Provide shortcuts, templates, and presets to expedite routine activities for healthcare providers and administrative staff.

Integration Capabilities:

Integrate with other healthcare systems and external databases to streamline data exchange and eliminate duplicate data entry.
Support interoperability standards such as HL7 for seamless communication between different systems within the hospital ecosystem.
Enable integration with medical devices and equipment to automate data capture and improve accuracy.

Resource Optimization:
Efficiently allocate resources such as hospital beds, operating rooms, and medical equipment based on demand and utilization patterns.
Utilize predictive analytics to forecast patient admissions, optimize staffing levels, and allocate resources effectively.

Real-time Information Access:
Ensure that critical patient information, including medical records, test results, and medication history, is readily accessible in real-time to healthcare providers.
Implement role-based access control to restrict access to sensitive information while providing relevant data to authorized users promptly.

Reporting and Analytics:
Provide robust reporting and analytics capabilities to monitor key performance indicators (KPIs), track operational metrics, and identify areas for improvement.
Offer customizable dashboards and reports tailored to the needs of different stakeholders, including receptionists, clinicians, and financial analysts.

Mobile Access:
Support mobile access to the HMS through native mobile apps or responsive web interfaces, allowing healthcare providers to access patient information and perform tasks remotely.
Ensure that mobile interfaces are optimized for performance and usability, with essential features accessible on smaller screens and limited bandwidth.

Scalability and Flexibility:
Design the HMS architecture to scale seamlessly as the hospital grows and patient volumes increase.
Utilize cloud-based infrastructure or scalable on-premises solutions to accommodate fluctuating resource demands and accommodate future expansion.
Allow for customization and configuration to adapt to evolving clinical workflows and organizational requirements.

Training and Support:
Provide comprehensive training programs for staff members to familiarize them with the HMS functionalities and best practices for efficient use.
Offer ongoing support channels, including helpdesk support, user forums, and knowledge bases, to address user queries and troubleshoot issues promptly.

3.2.1.3.1 Performance Requirements

Concurrent Terminals Support:

The system must be able to support a minimum of 100 terminals simultaneously.

Concurrent User Support:

The system must be capable of handling at least 500 simultaneous users concurrently.

Patient Records Handling:

The system must be capable of handling a minimum of 100,000 patient records.

User Information Access Time:

Each user should be able to access their respective information within 2 seconds of initiating a request.

Dynamic Numerical Requirements (Normal Workload):

The system should process at least 1000 patient admissions per day.

Appointment scheduling should handle a minimum of 5000 appointments per week.

Prescription requests should be processed at a rate of at least 100 per hour.

Billing transactions should be processed within 3 seconds of submission.

Lab test results should be generated and available to users within 1 hour of test completion.

Dynamic Numerical Requirements (Peak Workload):

During peak hours, the system should handle a surge of up to 1000 simultaneous users without performance degradation.

Appointment scheduling should accommodate a peak load of 100 appointments per hour.

Prescription requests should be processed at a rate of at least 200 per hour during peak periods.

The system should maintain responsiveness, with 95% of transactions completing within 2 seconds, even under peak load conditions.

Lab test results should remain available within 2 hours of test completion, even during peak workload periods.

Response Time Requirements:

95% of user interactions, including appointment scheduling, prescription requests, and accessing patient records, should be completed within 2 seconds under normal workload conditions.

During peak workload conditions, 95% of user interactions should be completed within 3 seconds to maintain acceptable performance levels.

System-generated reports, such as financial reports or operational analytics, should be generated within 5 minutes of request submission.

Data Processing Requirements:

The system should process and store patient data securely, with a minimum throughput of 100 transactions per second.

Data backups should be completed daily, with a maximum downtime of 1 hour for backup processes.

3.2.1.3.2 Space Requirements

Physical Space:

Hardware Infrastructure: Depending on whether the hospital management system is hosted on-premises or in the cloud, physical space may be required for servers, networking equipment, and storage devices. This could range from a dedicated server room to rack space in a data center.
Workstations: Space may be needed for desktop computers or terminals used by hospital staff to access the management system. This could include office space within the hospital or workstations in clinical areas.

Digital Storage:

Database Storage: The hospital management system will require storage space for the database(s) that store patient records, administrative data, and other information. The amount of storage needed will depend on factors such as the size of the hospital, the volume of patient data, and retention policies.
File Storage: In addition to databases, space may be needed for storing files such as medical images, scanned documents, and reports generated by the system.
Backup Storage: Adequate space must be allocated for regular backups of the system data to ensure data integrity and disaster recovery preparedness. This may involve onsite or offsite backup solutions.
Software Installation: Space may be required for installing and storing the software components of the hospital management system, including the application itself, operating system dependencies, and any additional software modules or plugins.

Accessibility and Security Considerations:

Physical security measures should be in place to protect hardware infrastructure and sensitive data stored on-site.
Digital security measures, such as encryption and access controls, should be implemented to safeguard patient information and system integrity.

3.2.1.4 Dependability

Availability:

- The application will be available every day of the year, 24 hours a day.
- The application will be available to everyone owning a PC or mobile phone connected to the internet.
- In case of any update or maintenance process, the application shall not be down for more than 1 hour.

Reliability:

The system focuses on reliability, ensuring consistent performance and minimal downtime. It has backup strategies to safeguard against data loss and service interruptions. Regular monitoring and proactive maintenance further enhance the system’s reliability, providing users with a trustworthy platform.

3.2.1.5 Security

Sensitive information like password will be hashed to protect privacy.

- Login will have 2FA (2 Factor Authentication).
- The system will validate user inputs before entering them in the database.

3.2.2 Organizational Requirements

3.2.2.1 Environmental Requirements

The Hospital Management System will be deployed on existing personal computers (PCs) within the hospital. The system should be compatible with standard PC hardware configurations and should not require any specialized hardware.

The system should be robust and capable of operating efficiently in standard office environments. The system must integrate seamlessly with the hospital's current electronic health records (EHR) system.

3.2.2.2 Operational Requirements

The hospital management system will support the following primary functions:

- **Record Management:** Efficient creation, updating, and management of patient records.
- **Exporting Records:** Functionality to export patient records in PDF and XML formats.
- **User Assignment:** Assign patients to doctors for consultations and manage these assignments through an intuitive interface.
- **Communication:** Facilitate secure messaging between doctors, receptionists, and patients.
- **Form Handling:** Create, manage, and process electronic forms for patient visits and consultations.

The system will be utilized by:

Doctors: For accessing patient records, managing appointments, and communication.

Receptionists: For scheduling appointments, managing patient information, and performing administrative tasks.

Patients: For accessing their own medical information, scheduling appointments, and communication with healthcare providers.

The system must be available 24/7 to support the hospital's round-the-clock operations, ensuring high reliability and uptime even during off-peak hours and emergency situations.

3.2.2.3 Development Requirements

The development of the hospital management system will adhere to the Waterfall methodology. This structured approach includes sequential phases such as requirement analysis, system design, implementation, testing, deployment, and maintenance.

The system must implement robust security measures to protect patient data, including data encryption, user authentication, access control, and regular security audits.
The following programming languages, frameworks, and technologies are mandated for this project:

- **HTML and CSS:** For front-end user interface design.
- **Java:** For core application logic and server-side development.
- **PHP:** For server-side scripting and integration tasks.
- **MySQL:** For database management and data storage.
- **Bootstrap**: For responsive and modern front-end design.
- **Ajax**: For asynchronous web page updates without reloading the entire page.

The system can be developed and deployed in either a cloud-based or on-premises setup. If the hospital's IT infrastructure supports on-premises deployment, this may be considered. Alternatively, a cloud-based solution could offer better scalability and lower upfront costs, ensuring ease of maintenance and future upgrades.

3.2.3 External Requirements

3.2.3.1 Regulatory Requirements

3.2.3.1.1 **Data Protection Law (Law No. 9887)**: The system must comply with Albania's data protection laws to ensure the lawful processing and protection of personal data, including patient health information.

3.2.3.1.2 **Healthcare Regulations**: The system should adhere to regulations set forth by the Albanian Ministry of Health and Social Protection to maintain standards of healthcare delivery and patient care.

3.2.3.2 Ethical Requirements

3.2.3.2.1 **Patient Privacy and Confidentiality:** The system must ensure strict adherence to ethical principles regarding patient privacy and confidentiality. This includes securing sensitive medical information, limiting access to authorized personnel only, and implementing encryption protocols for data transmission and storage.

3.2.3.2.2 **Informed Consent:** The system should support the ethical requirement of obtaining informed consent from patients for any medical procedures, treatments, or sharing of their health information. This includes providing clear and comprehensive information to patients about the purpose, risks, and benefits of any proposed interventions.

3.2.3.2.3 **End-of-Life Care and Dignity:** The system should support ethical practices related to end-of-life care, respecting patients' wishes, preferences, and dignity. This includes facilitating advanced care planning, palliative care options, and ensuring a compassionate approach to terminal illnesses.

3.2.3.3 Legislative Requirements

3.2.3.3.1 Patient Rights Legislation: The system must adhere to legislation protecting patient rights, including the right to privacy, informed consent, and access to medical records.

3.2.3.3.2 Accessibility Standards: The system should comply with accessibility standards outlined in Albanian legislation to ensure that individuals with disabilities can access healthcare services and information provided by the system. Electronic Health Records (EHR) Legislation: If applicable, the system should meet requirements outlined in legislation related to electronic health records (EHR), to facilitate the adoption and interoperability of electronic health records in Albania.

3.2.3.3.3 Healthcare Facility Licensing Regulations: The system should support compliance with regulations governing the licensing and operation of healthcare facilities in Albania, ensuring that hospitals using the system meet legal requirements for operation and service provision.

3.2.3.3.4 Accounting Requirements

3.2.3.3.4.1 Compliance with Regulations from the Proper Government Institution:

The accounting system must adhere to regulations set forth by the proper government institution for reporting financial information, including balance sheets, income statements, and cash flow statements.

3.2.3.3.4.2 Data Naming Conventions:

All financial data, including accounts, transactions, and reports, must follow standardized naming conventions as specified by the proper government institution to ensure consistency and accuracy in record-keeping.

3.2.3.3.4.3 Accounting Procedures and Policies:

The system must support and enforce established accounting procedures and policies mandated by the proper government institution, such as accrual accounting, inventory valuation methods, depreciation schedules, and revenue recognition criteria.

3.2.3.3.4.4 Audit Tracing and Logging:

The accounting software must include robust audit tracing and logging capabilities to track and record all financial transactions, changes to financial data, and user activities as required by the proper government institution. This includes capturing before and after values for any modifications made to critical financial records.

3.2.3.3.4.5 Financial Statement Generation:

The system should facilitate the generation of standardized financial statements, including balance sheets, income statements, and cash flow statements, in compliance with regulations from the proper government institution. These statements should be easily accessible and printable for internal and external reporting purposes.

3.2.3.3.5 Security Requirements

The data stored in the system's database is classified as sensitive information, necessitating robust security measures. Compliance with Law No. 9887, dated 10.03.2008, as amended by Law No. 48/2012, "On the Protection of Personal Data," mandates that personal information must remain private and accessible only to authorized individuals. As a result, user information is safeguarded using hashed passwords to prevent unauthorized access. Additionally, doctors and receptionists are required to exercise caution when handling patients' personal information to ensure confidentiality and privacy.

3.3 Domain Requirements

The Web Application functions within the realm of Albania's Public Medical System, focusing on digitizing patient records to streamline data access effectively. Data security is of utmost importance, with strict access controls limiting systems entry to authorized users. The application is tailored for deployment within a specific clinic’s isolated network and does not necessitate integration with external systems.

The system shall support the appropriate medical terminologies and coding systems for accurate documentation and billing processes (in Albania, ICD-10).

4. User Scenarios/Use Cases

4.1 Requirements Analysis

4.1.1 User Scenarios

4.1.1.1 User Scenarios List

Nb.	Name	Description
US_01	User signs up	Users: patients, doctors and receptionists sign up by filling the fields with the required personal data.
US_02	User logs in	Users: patients, doctors and receptionists log in using email and password
US_03	View personal data	Users: patients, doctors and receptionists can view their personal information.
US_04	Update personal data	Users: patients, doctors and receptionists can update their personal information.
US_05	Search a doctor	Users: receptionists and patients can search for a specific doctor.
US_06	Search a patient	Users: receptionists and doctors can search for a specific patient.
US_07	Search a receptionist	Main receptionist is able to search for the other receptionists.
US_08	User logs out	Users: receptionists, patients and doctors can log out from their account.
US_09	Send message	Receptionists can send messages to doctors and patients. Doctors can send messages to their patients. Patients can send messages to their doctors.
US_10	Patients' list	Receptionists can view the list of the patients. Doctors can also check a list of all his/her registered patients.
US_11	Doctors’ list	Users: receptionists and patients can view the list of the

		doctors. Patients can also check a list of the doctors they have been allocated to.
US_12	Receive reports	Receptionists are able to receive reports from patients regarding any issue for doctors and decide to forward the report to the respective doctor or not.
US_13	Export User	Receptionist can export a patient information and examinations (xml file) to another hospital for a specific treatment.
US_14	Make an appointment	Receptionist allows doctor to make an appointment or a visit for a specific patient.
US_15	See the timetable	Patients can see the schedule or the working hours of a doctor in case they want to have a visit. Receptionist have access to the timetable in case they have to postpone a visit or allocate a new time slot for the visit.
US_16	View statistics	Doctor and receptionist can download a spreadsheet of monthly records generated by the system. (nr of patients, nr of surgeries, nr of emergencies).
US_17	Change password	Patients, receptionist and doctors change their password
US_18	Add a new patient user	Receptionist creates an account for a new patient
US_19	Add a new doctor user	Receptionist creates an account for a new doctor
US_20	Leave a feedback	The patients can leave a feedback about the service or give a rating about any doctor.
US_21	Download patient’s examination.	Receptionist can download the examinations of all patients. Doctor can download the examinations of the patient. Patient can download his own examinations.

4.1.1.2 User Scenarios Extended

- **User Scenario 1** - *User signs up*
  - User chooses the ‘Signup’ button and chooses their user type: patient/doctor/receptionist.
  - User is redirected to the signup page.
  - User enters their personal information (Full Name, Birthday, E-mail, Password).
  - A “Welcome” message will be displayed.

- **User Scenario 2** - *User logs in*
  - User chooses the ‘Login’ button.
  - User is redirected to the login page.
  - User enters their email address and password.
  - User presses the ‘Submit’ button.
  - If data is correct, the user is redirected to his home page.
  - If data is incorrect an error message will be shown and the patient is requested to re enter their credentials. Patient restarts from step 2.

- **User Scenario 3** - *View personal data*
  - Patient logs in using steps in **User Scenario 2**.
  - Patient chooses ‘My Profile’ on the navigation bar.
  - Patient can view their personal data.

- **User Scenario 4** - *Update personal data*
  - Patient opens their profile using steps in **User Scenario 3**.
  - Patient can update anything that they want to update.
  - Patient presses the button “Save Changes”.
  - If all data are entered correctly, the message “Successful” will be shown to the user.
  - If any data is not entered correctly, an error message will be shown below the specific incorrectly entered data.

- **User Scenario 5** - *Search a doctor*
  - Patients:
    - Patient gets a list of all doctors or their registered doctors using steps in **User Scenario 11**.

        - Above the list, an ‘Input’ field will be shown and next to it the ‘Search’ button. The patient can type the name of the doctor they want to search and press the button ‘Search’.
        - If the doctor exists, the doctor’s name and ‘View’ button next to it will appear.
        - If they do not exist, a message ‘Doctor does not exist’ will appear.
    - Receptionist:
        - Receptionist gets a list of all doctors using steps in **User Scenario 11**.
        - Above the list, an ‘Input’ field will be shown and next to it the ‘Search’ button. The receptionist can type the name of the doctor they want to search and press the button ‘Search’.
        - If the doctor exists, the doctor’s name and ‘View’ button next to it will appear.
        - If they do not exist, a message ‘Doctor does not exist’ will appear.

- **User Scenario 6** - *Search a patient*
    - Doctors:
        - Doctor gets the list of their patients using steps in **User Scenario 10**.
        - Above the list, an ‘Input’ field will be shown and next to it the ‘Search’ button. The doctor can type the name of the patient they want to search and press the button ‘Search’.
        - If the patient exists, the patient’s name and ‘View’ button next to it will appear.
        - If they do not exist, a pop up message ‘Patient does not exist’ will appear.
    - Receptionist:
        - Receptionist gets the list of doctors using steps in **User Scenario 10**.
        - Above the list, an ‘input’ field will be shown and next to it the ‘Search’ button. The receptionist can type the name of the patient they want to search and press the button ‘Search’.
        - If the patient exists, the patient’s name and ‘View’ button next to it will appear.
        - If they do not exist, a pop up message ‘Patient does not exist’ will appear.

- **User Scenario 7** - *Search a receptionist*
    - The main receptionist (id=1) logs in using steps in **User Scenario 2**.
    - Main receptionist chooses the ‘All Admins’ on their page.
    - Above the list, an ‘Input’ field will be shown and next to it the ‘Search’ button. The main receptionist can type the name of the receptionist they want to search and press the button ‘Search’.
    - If the receptionist exists, their name and ‘View’ button next to it will appear.
    - If they do not exist, a pop up message ‘receptionist does not exist’ will appear.

- **User Scenario 8** - *User logs out*

- User logs in using steps in **User Scenario 2.**
- User follows some of the scenarios mentioned above.
- User clicks the 'Log out' button on the navigation bar.
- User will be logged out of their account and will be redirected to the home page.

- **User Scenario 9** - *Send message*
  - Doctor:
    - Doctor logs in using steps in **User Scenario 2**.
    - Doctor can choose the 'Send Message' button on their page.
    - A dropdown menu will be shown to them where they can choose if they want to send a message to one of their registered patients. Below, a field is shown where the doctor can type the message out.
    - After finishing the message, they can press the 'Send' button next to the field so that the message can be sent to the destination.
  - Receptionist:
    - Receptionist logs in using steps in **User Scenario 2**.
    - Receptionist chooses 'Send Message' on their page.
    - A dropdown menu will be shown to them where they can choose if they want to send this message to all patients, all doctors, or to any specific receiver, so they type in the name of the patient or doctor they want to send this message to. Below, a field is shown where the receptionist can type the message out.
    - After finishing the message, they can press the 'Send' button next to the field so that the message can be sent to the destination.
  - Patient:
    - Patient logs in using steps in **User Scenario 2**.
    - Patient can choose the 'Send Message" button on their page.
    - A dropdown menu will be shown to them where they can choose if they want to send a message to one of their registered doctors. Below, a field is shown where the patient can type the message out.
    - After finishing the message, they can press the 'Send' button next to the field so that the message can be sent to the destination.

- **User Scenario 10** - *Patients' List*
  - *Receptionist*:
    - Receptionist logs in using steps in **User Scenario 2**.
    - Receptionist chooses 'All Patients' on their page.
    - A list of all patients will be shown to them.
  - Doctors:
    - Doctors logs in using steps in **User Scenario 2**.

- Doctor can choose the ‘Registered Patients’ button on their page.
- A list of all of their patients will be displayed.

- **User Scenario 11** - *Doctors’ List*
  - Receptionist:
    - Receptionist logs in using steps in **User Scenario 2**.
    - Receptionist chooses ‘All Doctors’ on his page.
    - A list of all doctors will be shown to them.
  - Patients:
    - Patient logs in using steps in **User Scenario 2**.
    - Patient can choose the ‘All Doctors’ button on their page.
    - A list of all doctors will be displayed.
    - Patient can also choose the ‘Registered Doctors’ button on their page.
    - A list of all the doctors they has been allocated to will be displayed.

- **User Scenario 12** - *Receive reports*
  - Receptionist logs in using steps in **User Scenario 2**.
  - Receptionist can choose the ‘Reports’ button on the navigation bar.
  - A list of all reports will be displayed with the reason why and the name and email of the doctor that has been reported.
  - Next to each report, there are two buttons: ‘Ignore’ and ‘Ban’.
  - If the receptionist chooses the ‘Ignore’ button, it means he is ignoring this report and that the doctor can continue to use his account.
  - If the receptionist chooses the ‘Ban’ button, it means that he has banned this doctor from continuing his work and the doctor can not enter with his credentials.

- **User Scenario 13** - *Export user*
  - Receptionist searches for a patient using steps in **User Scenario 6**.
  - Receptionist clicks the button “Export” for that specific patient.
  - A dialog box is displayed and the receptionist chooses whether they are sure to continue.
  - If the receptionist does not want to export, they remain at the same page.
  - If the receptionist exports, all the information and details for that patient is written in an XML file and downloaded in the computer.
  - The exported file is sent to another hospital or polyclinic for further examination.

- **User Scenario 14** - Make an appointment
  - Doctor views the list of patients waiting to be examined following the steps in User Scenario 10

- Doctor clicks the button “Create Visit” for a specific patient
- A form with all the needed fields to be filled during a medical visit will be shown
- Doctor fills all the empty fields
- Doctor double checks if the data entered is correct
- Doctor clicks the button “Finish Examination”
- Alert: Are you sure you want to continue? YES/ NO
- If NO, do nothing, we stay at the same page
- If YES, the data is saved in the database. An informative message “Visit created
- successfully!” will be shown and the doctor is redirected to his/her home page.
- Also a PDF file is created.

- **User Scenario 15** - See timetable
  - User logs in via steps in User Scenario 02
  - User clicks the “Timetable” button.
  - A new page loads with all the appointments scheduled for that patient.

- **User Scenario 16** - View statistics
  - User (doctor or receptionist) logs in via steps in User Scenario 02.
  - User clicks the “Statistics” button.
  - A new page loads with all the data recorded on a monthly basis regarding visits, surgeries, emergencies etc. It is downloadable.

- **User Scenario 17 –** Change password
  - User logs in following the steps in User Scenario 01
  - User chooses the menu “Change Password”
  - User types his old password and his new password (2 times)
  - User presses the button “Save Changes”
  - If the old password is correct and if the new password is the same in both fields the user is alerted: “Password was changed successfully!”
  - In case the old password is wrong or the new password is not the same in both fields then the user is alerted: “Password was not changed. Please try again.”

- **User Scenario 18** – Add a new patient user
  - Receptionist logs in following the steps User Scenario 02
  - Receptionist clicks on “Add a New Patient”
  - Receptionist fills all the empty fields
  - Receptionist double checks if the data entered is correct
  - Receptionist clicks the button “Add Patient”
  - Alert: Are you sure you want to continue? YES/ NO

- If NO, do nothing, we stay at the same page
- If YES, system validates data according to the specified requirements
- If validation is passed successfully, the data is saved in the database and the account is created. An informative message “Patient added successfully!” will be shown and the receptionist is redirected to their home page.
- If validation is not passed successfully, informative messages will show where the problem is, so the receptionist can fix it and continue again from clicking “Add Patient”.

- **User Scenario 19** – Add a new doctor user
  - Receptionist logs in following the steps in User Scenario 02.
  - Receptionist clicks on “Add a New Doctor”
  - Receptionist fills all the empty fields
  - Receptionist double checks if the data entered is correct
  - Receptionist clicks the button “Add Doctor”
  - Alert: Are you sure you want to continue? YES/ NO
  - If NO, do nothing, we stay at the same page
  - If YES, system validates data according to the specified requirements
  - If validation is passed successfully, the data is saved in the database and the account is created. An informative message “Doctor added successfully!” will be shown and the receptionist is redirected to his/her home page.
  - If validation is not passed successfully, informative messages will show where the problem is, so the receptionist can fix it and continue again from clicking “Add Doctor”.
- **User Scenario 20** - Leave feedback
  - Patient logs in via steps in User Scenario 02..
  - Then clicks on “Rating” button.
  - Rating and feedback page opens up.
  - Patients can give feedback about the service offered in the polyclinic or about any dissatisfaction they had or about any tip to help us improve our service.
  - Patients clicks “Send” and all the reviews and ratings are sent to our database. These can be seen on the statistics page.
- **User Scenario 21 -** Download patient’s examination
  - Receptionist or Doctor
    - User views examinations of a patient following steps in User Scenario 06.
    - User clicks on “Download” button.
    - A PDF file will be downloaded on the PC.
  - Patient
    - Patient views examinations of themself following steps in User Scenario 03.
    - Patient clicks “Download” button.
    - A PDF file will be downloaded on the PC.

4.1.2 Use Cases

UC Name	User signs up
Summary	User signs up and creates a new account by providing their personal information.
Dependency	None
Actors	Patient / Receptionist / Doctor
Preconditions	User must not have an existing account.
Description of the Main Sequence	• User selects the "Signup" button. • User specifies their user type as either a patient, doctor, or receptionist. • User is redirected to the signup page. • User enters their personal information, including Full Name, Birthday, E-mail, and Password. • User presses the ‘Submit’ button. • Upon successful submission, a "Welcome" message is displayed to acknowledge the successful signup.
Description of the Alternative Sequence	If data is not entered correctly, the user is requested to re-enter their credentials.
Non functional requirements	User must have their personal information.
Postconditions	User account is created, and the user is logged in.

UC_01 - US_01 : User signs up

UC Name	User logs in
Summary	User logs in to access their account.
Dependency	None
Actors	Patient / Receptionist / Doctor
Preconditions	User must have an active account.
Description of the Main Sequence	• User selects the "Login" button. • User is redirected to the login page. • User enters their email address and password. • User presses the ‘Submit’ button. • Upon successful submission, the user is redirected to their home page.
Description of the	• If data is incorrect, an error message is shown, and the patient is

Alternative Sequence	requested to re-enter their credentials. User restarts from step 2.
Non functional requirements	User must have their email address and password.
Postconditions	User is logged in and redirected to their home page.

UC_02 - US_02 : User logs in

UC Name	View personal data
Summary	The patient views their personal data on their profile page.
Dependency	UC_02 - US_02 : Patient logs in
Actors	Patient
Preconditions	Patient must be logged in.
Description of the Main Sequence	• Patient logs in using the steps outlined in User Scenario 2. • Patient selects ‘My Profile’ on the navigation bar. • Patient is directed to their profile page. • Patient views their personal data on the profile page.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	Patient successfully views their personal data.

UC_03 - US_03 : View personal data

UC Name	Update personal data
Summary	Patient updates their personal data on their profile page.
Dependency	UC_03 - US_03 : View Personal Data
Actors	Patient
Preconditions	Patient must be logged in and accessing their profile.
Description of the Main Sequence	• Patient opens their profile using the steps outlined in User Scenario 3. • Patient updates the desired personal data. • Patient presses the "Save Changes" button. • If all data are entered correctly, the message "Successful" is shown to the user.

Description of the Alternative Sequence	• If any data is not entered correctly, a message is shown below the specific incorrectly entered data.
Non functional requirements	None
Postconditions	Patient's personal data is updated successfully.

UC_04 - US_04 : Update personal data

UC Name	Search a doctor
Summary	Patients or Receptionists search for a doctor by name.
Dependency	UC_11 - US_11 : Doctors' List
Actors	Patient / Receptionist
Preconditions	The user (Patient or Receptionist) is logged in and has navigated to the page displaying a list of doctors (as per User Scenario 11).
Description of the Main Sequence	• User (Patient or Receptionist) navigates to the page displaying a list of doctors. • Above the list, an input field and a "Search" button are displayed. • User enters the name of the doctor they want to search for in the input field. • User presses the "Search" button. • The doctor's name and a "View" button appear next to it.
Description of the Alternative Sequence	• If the doctor does not exist, a message "Doctor does not exist" is displayed.
Non functional requirements	None
Postconditions	The user successfully searches for a doctor.

UC_05 - US_05 : Search a doctor

UC Name	Search a patient
Summary	Doctors or Receptionists search for a patient by name.
Dependency	UC_10 - US_10 : Patients' List
Actors	Receptionist / Doctor
Preconditions	The user (Receptionist or Doctor) is logged in and has navigated to the page displaying a list of patients (as per User Scenario 10).

Description of the Main Sequence	• User (Receptionist or Doctor) navigates to the page displaying a list of patients. • Above the list, an input field and a "Search" button are displayed. • User enters the name of the patient they want to search for in the input field. • User presses the "Search" button. • The patient's name and a "View" button appear next to it.
Description of the Alternative Sequence	• If the patient does not exist, a pop-up message "Patient does not exist" appears.
Non functional requirements	None
Postconditions	The user successfully searches for a patient.

UC_06 - US_06 : Search a patient

UC Name	Search a receptionist
Summary	Main receptionist searches for another receptionist by name.
Dependency	None
Actors	Receptionist
Preconditions	The main receptionist is logged in and has navigated to the page displaying a list of all admins.
Description of the Main Sequence	• The main receptionist logs in using the steps outlined in User Scenario 2. • The main receptionist chooses "All Admins" on their page. • Above the list of receptionists, an input field and a "Search" button are displayed. • The main receptionist enters the name of the receptionist they want to search for in the input field. • The main receptionist presses the "Search" button. • If the receptionist exists, their name and a "View" button appear next to it.
Description of the Alternative Sequence	• If the receptionist does not exist, a pop-up message "Receptionist does not exist" appears.
Non functional	None

requirements	
Postconditions	The main receptionist successfully searches for another receptionist.

UC_07 - US_07 : Search a receptionist

UC Name	User logs out
Summary	User logs out of their account.
Dependency	None
Actors	Patient / Receptionist / Doctor
Preconditions	User is logged in.
Description of the Main Sequence	• User logs in using the steps outlined in User Scenario 2. • User follows some of the scenarios mentioned above. • User clicks the "Log out" button on the navigation bar. • User is logged out of their account. • User is redirected to the home page.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	User is successfully logged out of their account and redirected to the home page.

UC_08 - US_08 : User logs out

UC Name	Send message
Summary	Users send messages to other users within the system.
Dependency	None
Actors	Patient / Receptionist / Doctor
Preconditions	The user is logged in.
Description of the Main Sequence	• User logs in using the steps outlined in User Scenario 2. • User chooses the "Send Message" button on their page. • A dropdown menu is revealed: • For patients: they can choose to send a message to one of their registered doctors; • For receptionists: they can choose to send the message to all patients, all doctors, or to any specific receiver by typing in the name of the patient or doctor.

	• For doctors: they can choose to send a message to one of their registered patients. • Below the dropdown menu, a field is shown where the user can type the message. • After finishing the message, the user presses the "Send" button next to the field to send the message to the destination.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	The message is successfully sent to the destination.

UC_09 - US_09 : Send message

UC Name	View patients' list
Summary	Receptionists and Doctors view a list of patients associated with their account.
Dependency	None
Actors	Receptionist / Doctor
Preconditions	The user (Receptionist or Doctor) is logged in.
Description of the Main Sequence	• Receptionist logs in using the steps outlined in User Scenario 2. • For receptionists: they choose "All Patients" on their page. • For doctors: they choose the "Registered Patients" button on their page. A list of all patients associated with the user’s account is displayed.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	The user successfully views the list of patients associated with their account.

UC_10 - US_10 : View patients' list

UC Name	View doctors' list
Summary	Receptionists and Patients view a list of doctors.

Dependency	None
Actors	Patient / Receptionist
Preconditions	The user (Patient or Receptionist) is logged in.
Description of the Main Sequence	For Receptionists: Receptionist logs in using the steps outlined in User Scenario 2. Receptionist chooses "All Doctors" on their page. A list of all doctors is shown to the receptionist. For Patients: Patient logs in using the steps outlined in User Scenario 2. Patient chooses the "All Doctors" button on their page. A list of all doctors is displayed. Patient can also choose the "Registered Doctors" button on their page. A list of all the doctors the patient has been allocated to is displayed.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	The user successfully views the list of doctors.

UC_11 - US_11: View doctors' list

UC Name	Receive reports
Summary	Receptionists are able to receive reports from patients regarding any issue for doctors and decide to ignore it or not.
Dependency	None
Actors	Receptionist
Preconditions	Receptionist must be logged in
Description of the Main Sequence	• Receptionist logs in using steps in User Scenario 2. • Receptionist can choose the ‘Reports’ button on the navigation bar. • A list of all reports will be displayed with the reason why and the name

	and email of the doctor that has been reported. • Next to each report, there are two buttons: 'Ignore' and 'Ban'.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	• If the receptionist chooses the 'Ignore' button, it means he is ignoring this report and that the doctor can continue to use his account. • If the receptionist chooses the 'Ban' button, it means that he has banned this doctor from continuing his work and the doctor can not enter with his credentials.

UC_12 - US_12 : Receive reports

UC Name	Export user
Summary	Receptionist can export a patient information and examinations (xml file) to another hospital for a specific treatment.
Dependency	UC_06 - US_06 : Search a patient
Actors	Receptionist
Preconditions	Receptionists has searched the patient
Description of the Main Sequence	• Receptionist searches for a patient using steps in User Scenario 6. • Receptionist clicks the button export for that specific patient. • A dialog box is displayed and the receptionist chooses whether they is sure to continue. • If the receptionist does not want to export, they remain at the same page. • If the receptionist exports, all the information and details for that patient is written in a XML file and downloaded in the computer.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	The exported file is sent to another hospital or polyclinic for further examination

UC_13 - US_13: Export user

UC Name	Make an appointment
Summary	Receptionist allows doctor to make an appointment or a visit for a specific

	patient.
Dependency	UC_10 - US_10 : View patients' list
Actors	Doctor
Preconditions	Doctor has patients in the patient's list
Description of the Main Sequence	• Doctor views the list of patients waiting to be examined following the steps in User Scenario 10 • Doctor clicks the button "Create Visit" for a specific patient • A form with all the needed fields to be filled during a medical visit will be shown • Doctor fills all the empty fields • Doctor double checks if the data entered is correct • Doctor clicks the button "Finish Examination" • Alert: Are you sure you want to continue? YES/ NO • If YES, the data is saved in the database. An informative message "Visit created • successfully!" will be shown and the doctor is redirected to his/her home page.
Description of the Alternative Sequence	• If NO, do nothing, we stay at the same page
Non functional requirements	None
Postconditions	• Also a PDF file is created.

UC_14 - US_14: Make an appointment

UC Name	See Timetable
Summary	Receptionist and patients can see the schedule or the working hours for a specific doctor they want to have a visit with.
Dependency	None
Actors	Doctor / Receptionist / Patient
Preconditions	User is logged in
Description of the Main Sequence	• User logs in via steps in User Scenario 02 • User clicks the "Timetable" button. • A new page loads with all the appointments scheduled for that patient.
Description of the Alternative Sequence	None
Non functional	None

requirements	
Postconditions	None

UC_15 - US_15 : See Timetable

UC Name	View Statistics
Summary	Doctor and receptionist can download a spreadsheet of monthly records generated by the system. (nr of patients, nr of surgeries, nr of emergencies).
Dependency	None
Actors	Receptionist / Doctor
Preconditions	Doctor or receptionist is logged in
Description of the Main Sequence	• Doctor or receptionist logs in via steps in User Scenario 02. • Doctor or receptionist clicks the “Statistics” button. • A new page loads with all the data recorded on a monthly basis. It is downloadable.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	None

UC_16 - US_16 : View Statistics

UC Name	Change password
Summary	Patients, receptionists and doctors change their password
Dependency	None
Actors	Doctor / Receptionist / Patient
Preconditions	User is logged in
Description of the Main Sequence	• User logs in following the steps in US_01. • User chooses the menu “Change Password” • User types his old password and his new password (2 times) • User presses the button “Save Changes” • If the old password is correct and if the new password is the same in both fields, the user is alerted: “Password was changed successfully!”
Description of the Alternative Sequence	• In case the old password is wrong, or the new password is not the same in both fields, then the user is alerted: “Password was not changed.

	Please try again.”
Non functional requirements	None
Postconditions	Password is changed

UC_17 - US_17 : Change password

UC Name	Add a new patient user
Summary	Receptionist creates an account for a new patient
Dependency	None
Actors	Receptionist
Preconditions	Receptionist is logged in
Description of the Main Sequence	• Receptionist logs in following the steps US_02 • Receptionist clicks on “Add a New Patient” • Receptionist fills all the empty fields • Receptionist double checks if the data entered is correct • Receptionist clicks the button “Add Patient” • Alert: Are you sure you want to continue? YES/ NO • If YES, system validates data according to the specified requirements • If validation is passed successfully, the data is saved in the database and the account is created. An informative message “Patient added successfully!” will be shown and the receptionist is redirected to their home page.
Description of the Alternative Sequence	• If NO, do nothing, we stay at the same page • If validation is not passed successfully, informative messages will show where the problem is, so the receptionist can fix it and continue again from from clicking “Add Patient”.
Non functional requirements	None
Postconditions	Patient is added

UC_18 - US_18 : Add a new patient user

UC Name	Add a new doctor user
Summary	Receptionist creates an account for a new doctor
Dependency	None
Actors	Receptionist
Preconditions	Receptionist is logged in

Description of the Main Sequence	• Receptionist logs in following the steps in US_02 • Receptionist clicks on “Add a New Doctor” • Receptionist fills all the empty fields • Receptionist double checks if the data entered is correct • Receptionist clicks the button “Add Doctor” • Alert: Are you sure you want to continue? YES/ NO • If YES, system validates data according to the specified requirements • If validation is passed successfully, the data is saved in the database and the account is created. An informative message “Doctor added successfully!” will be shown and the receptionist is redirected to his/her home page.
Description of the Alternative Sequence	• If NO, do nothing, we stay at the same page • If validation is not passed successfully, informative messages will show where the problem is, so the receptionist can fix it and continue again from from clicking “Add Doctor”.
Non functional requirements	None
Postconditions	Doctor is added

UC_19 - US_19 : Add a new doctor user

UC Name	Leave feedback
Summary	Patients provide feedback about a doctor or a surgery, which helps doctors improve their performance.
Dependency	None
Actors	Patient
Preconditions	Patients must be logged in and doctors must be logged in to see the feedback.
Description of the Main Sequence	Patient logs into their account and goes to the main menu. By clicking on a specific doctor, they can see details about that doctor and also leave feedback. By clicking the button “Leave Feedback”, a new tab opens where there is a text box to leave a comment and also a rating bar for the doctor.
Description of the Alternative Sequence	• Receptionist may review feedback data as well.
Non functional requirements	None
Postconditions	The feedback is considered.

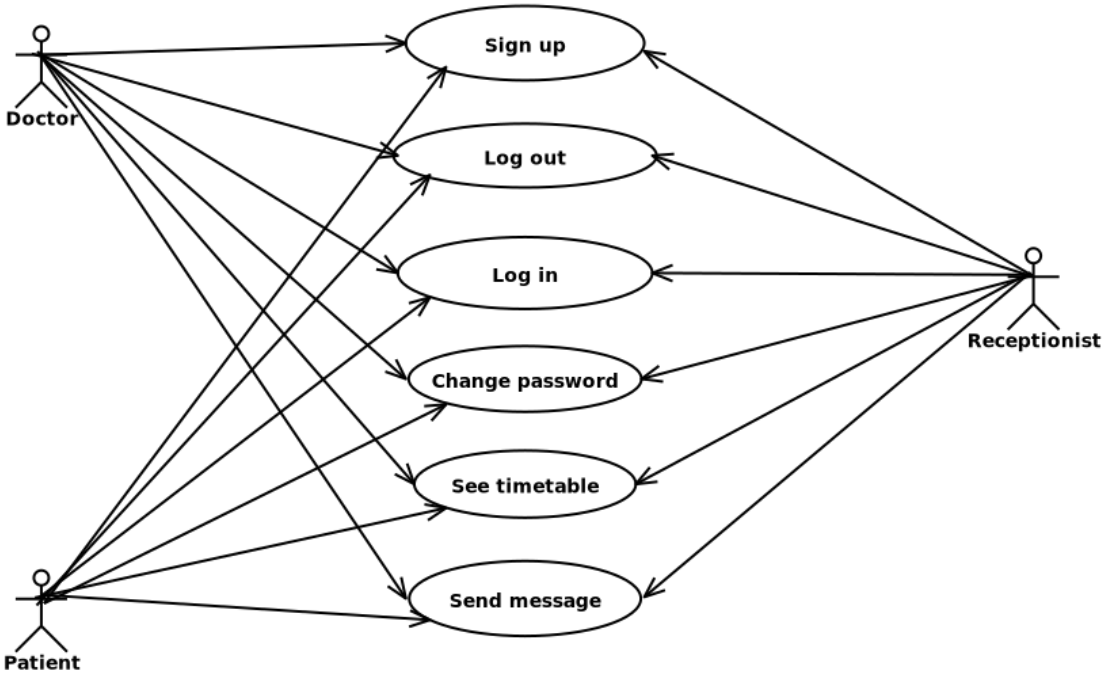
UC_20 - US_20 : Leave feedback

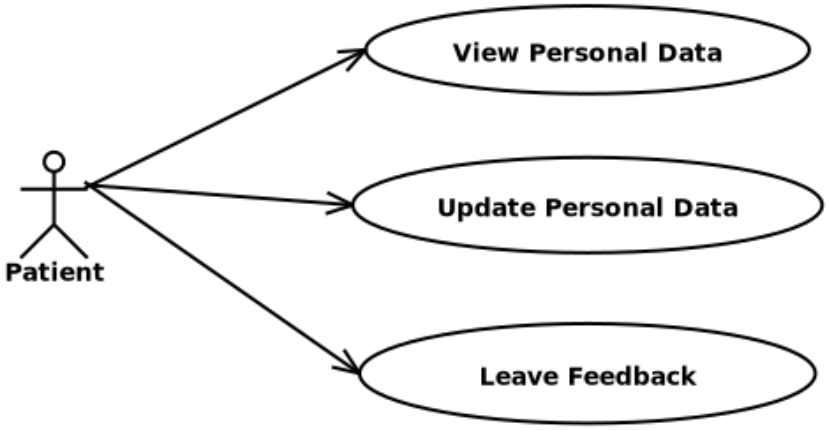
UC Name	**Download patient's examination**
Summary	Receptionists, Doctors, and Patients download a patient's examination report.
Dependency	UC_06-US_06/UC_03-US_03
Actors	Receptionist / Doctor / Patient
Preconditions	The user (Receptionist, Doctor, or Patient) is logged in and has viewed the examinations of the patient.
Description of the Main Sequence	For Receptionists/Doctors: 1. Receptionist or Doctor views the examinations of a patient following the steps outlined in User Scenario 06 2. Receptionist or Doctor clicks on the "Download" button. 3. A PDF file containing the patient's examination report is downloaded on the PC. For Patients: 1. Patient views their examinations following the steps outlined in User Scenario 03. 2. Patient clicks on the "Download" button. 3. A PDF file containing the patient's examination report is downloaded on the PC.
Description of the Alternative Sequence	None
Non functional requirements	None
Postconditions	The user successfully downloads the patient's examination report.

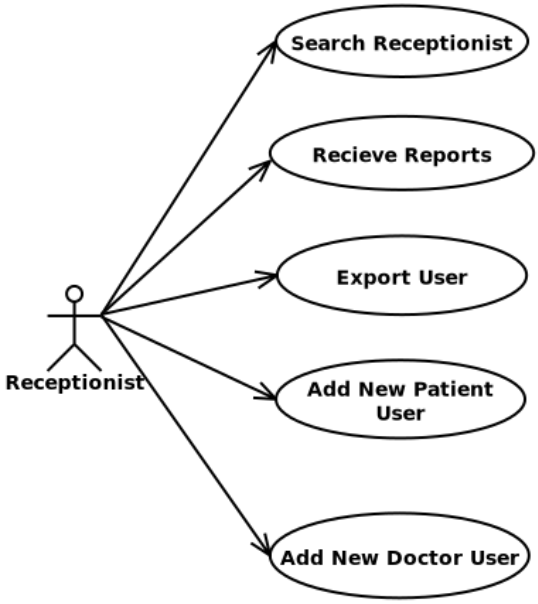
UC_21 - US_21 : Download patient's examination

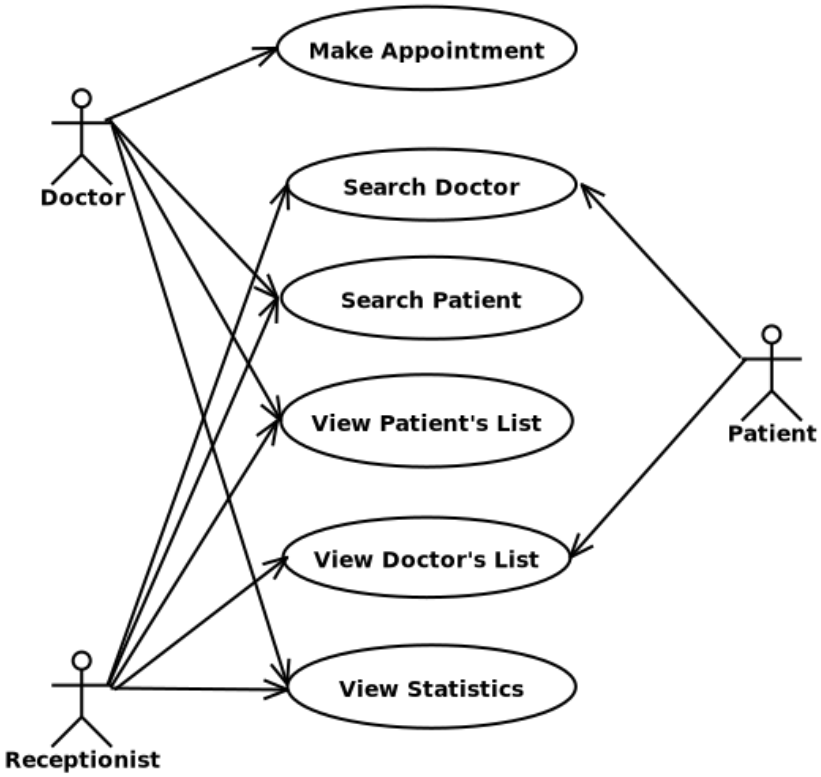
4.2 Behavioral Diagrams

4.2.1 Use Case Diagrams

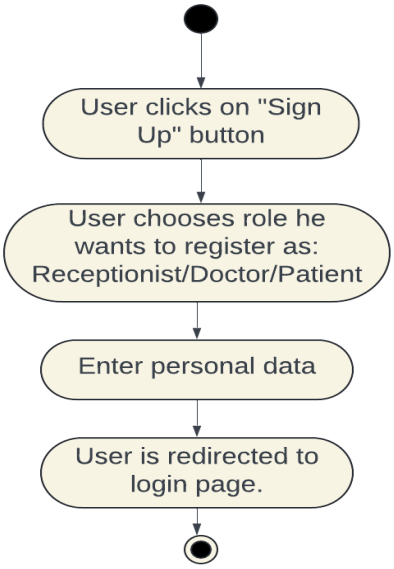




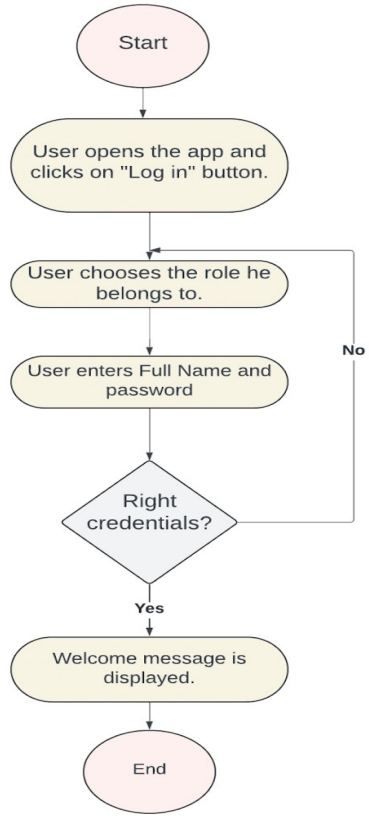

Search Receptionist
Recieve Reports
Export User
Add New Patient User
Add New Doctor User
Receptionist


Make Appointment
Search Doctor
Search Patient
View Patient's List
View Doctor's List
View Statistics
Doctor
Patient
Receptionist

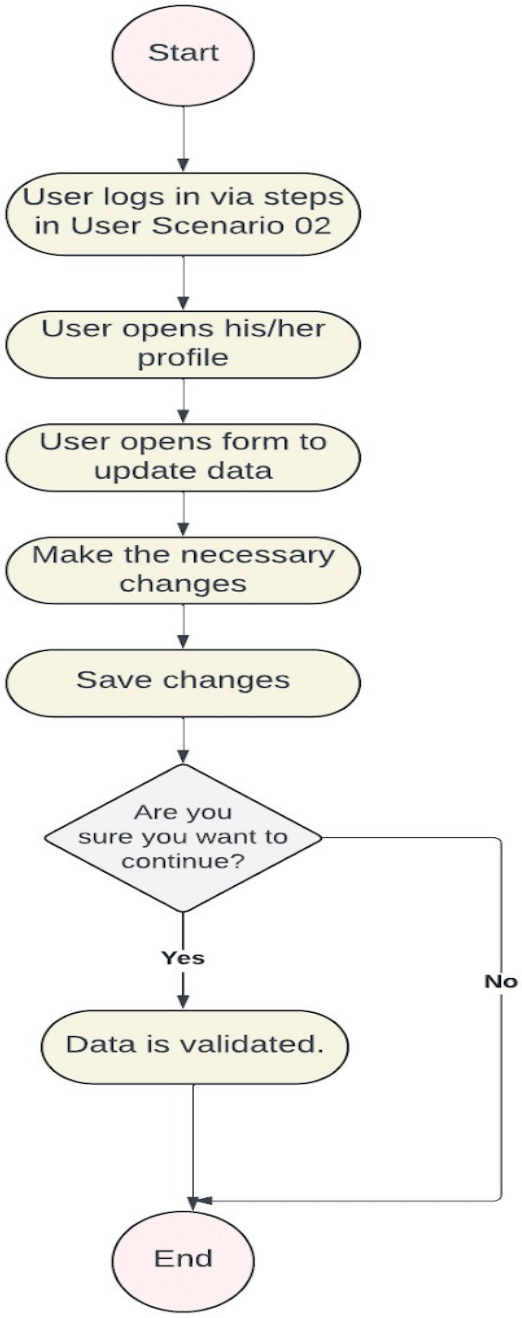
4.2.2 Activity Diagrams



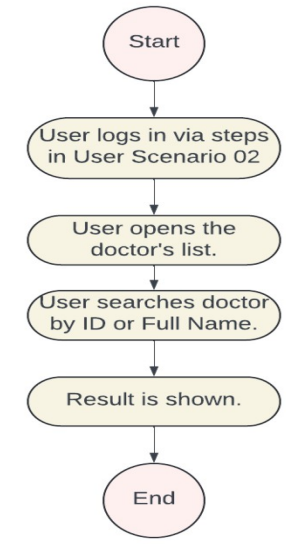
User Scenario 01 - Sign Up



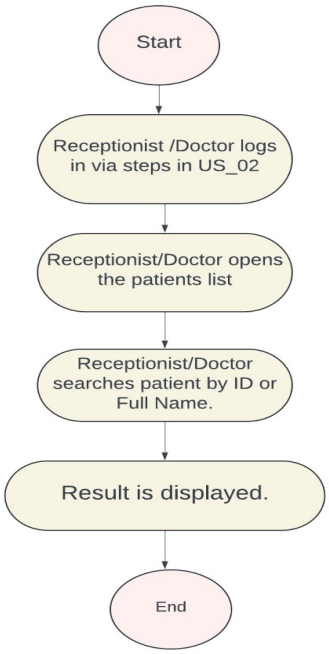
User Scenario 2 - Log in



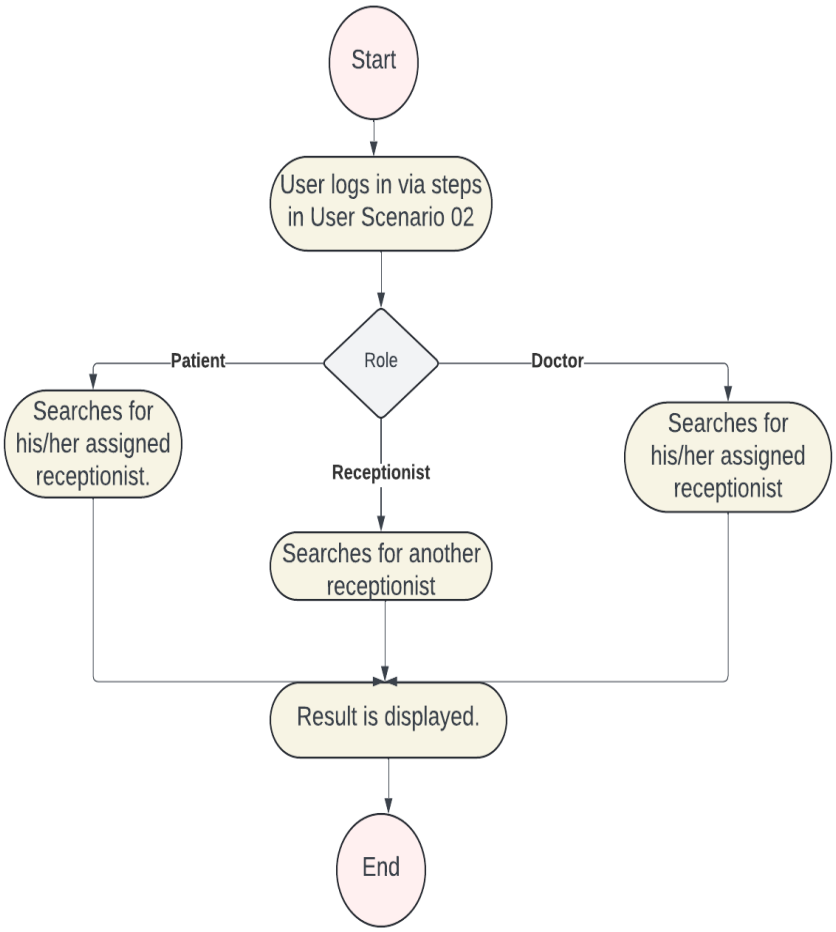
User Scenario 04 - Update Personal Data



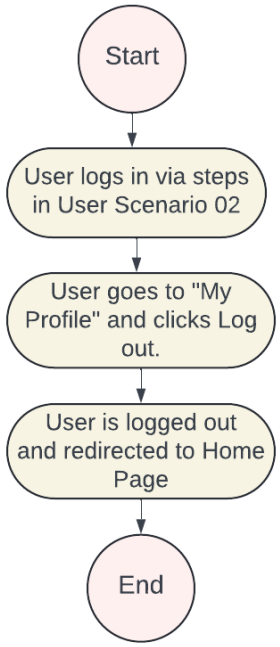
User Scenario 05 - Search a doctor



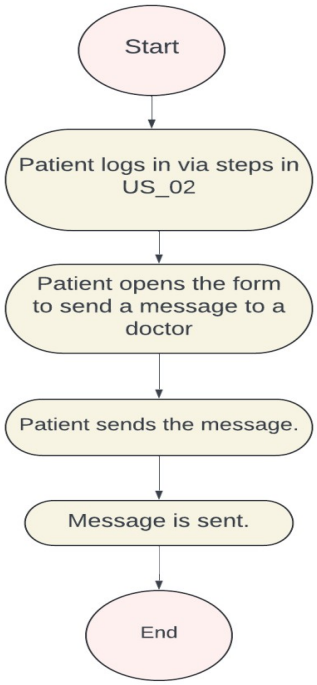
User Scenario 6 - Search a patient



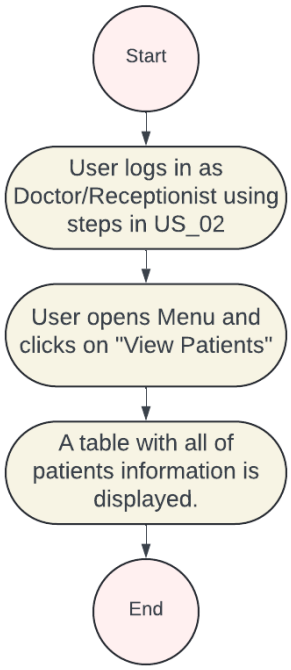
User Scenario 07 - Search a receptionist



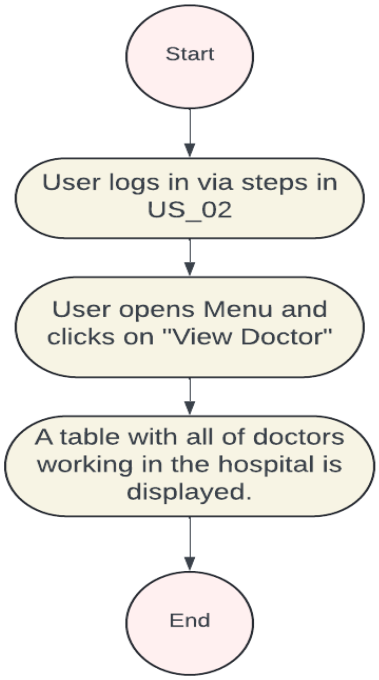
User Scenario 08 - User logs out



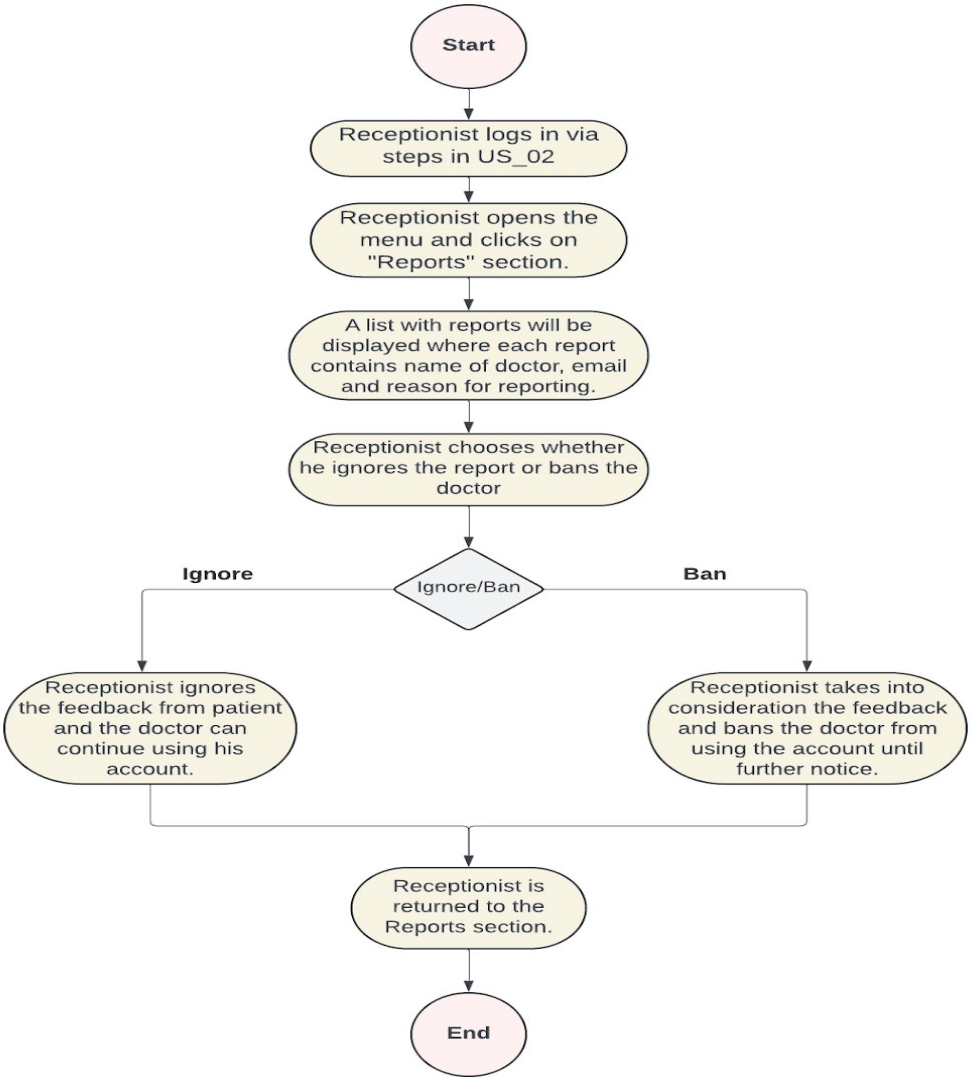
User Scenario 9 - Send a message



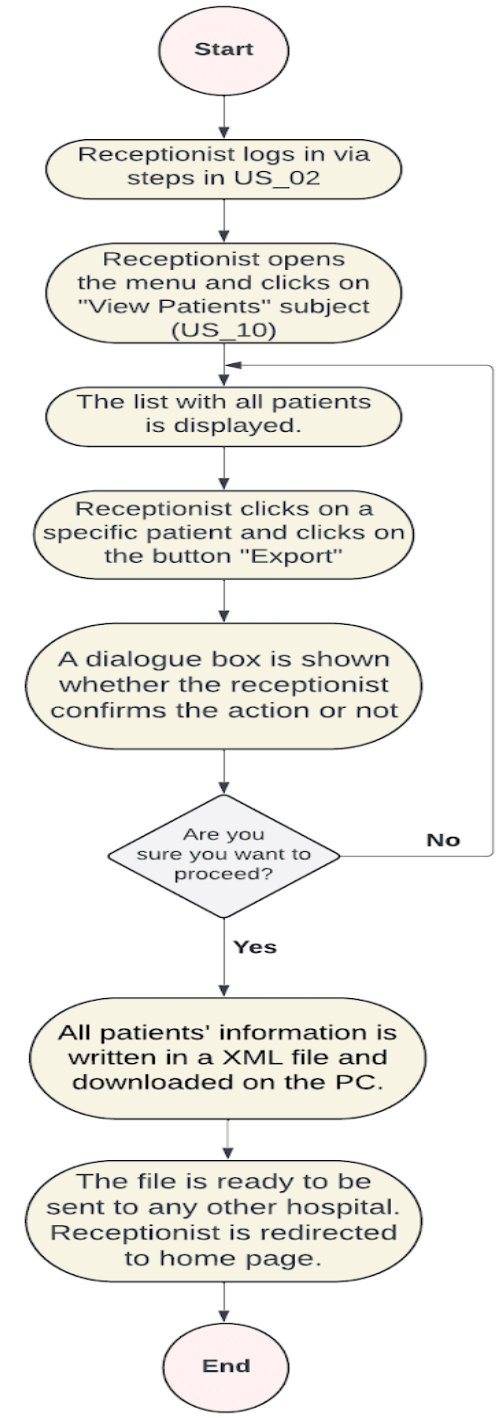
User Scenario 10 - View Patient List



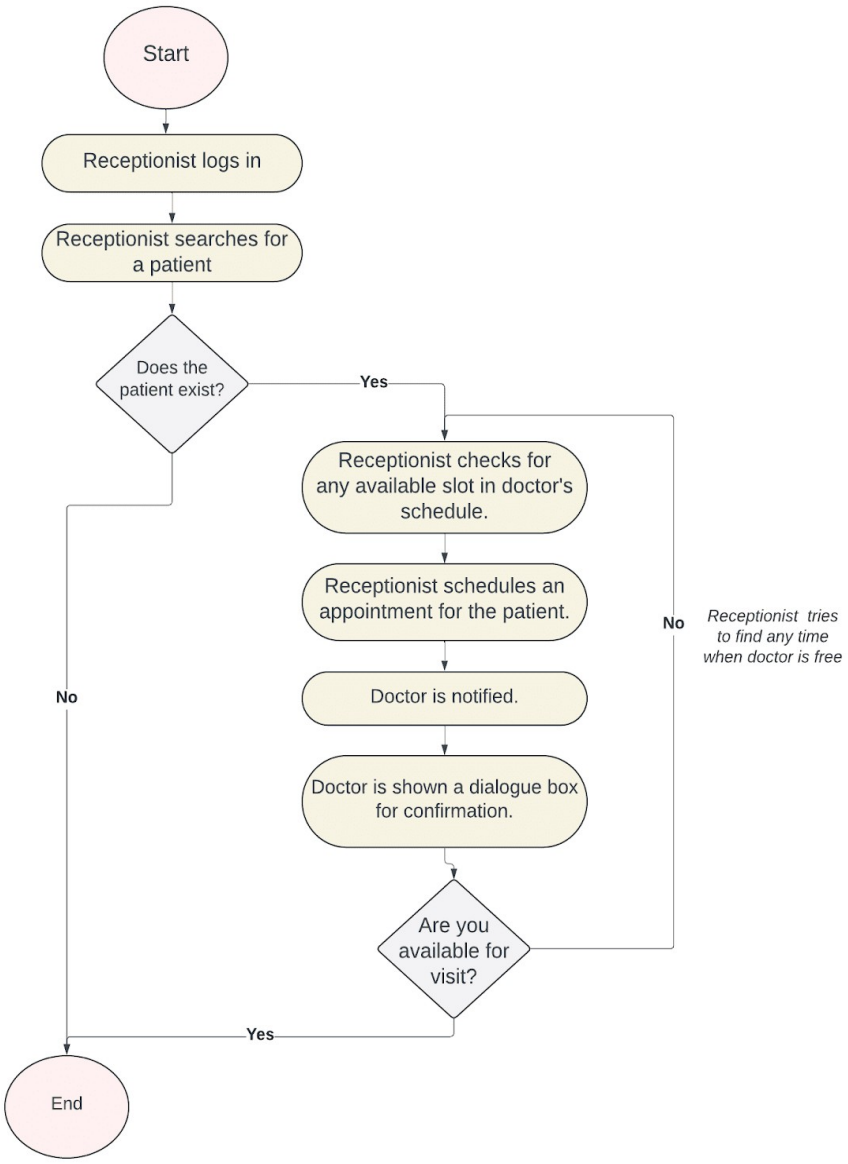
User Scenario 11 - View Doctor List



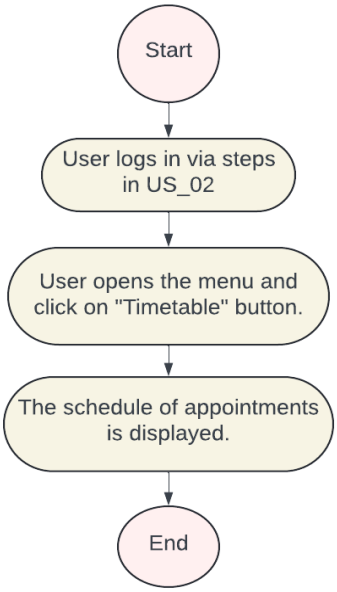
User Scenario 12 - Receive Reports



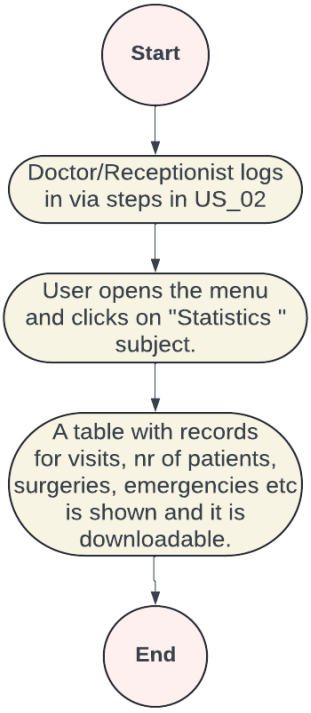
User Scenario 13 - Export User



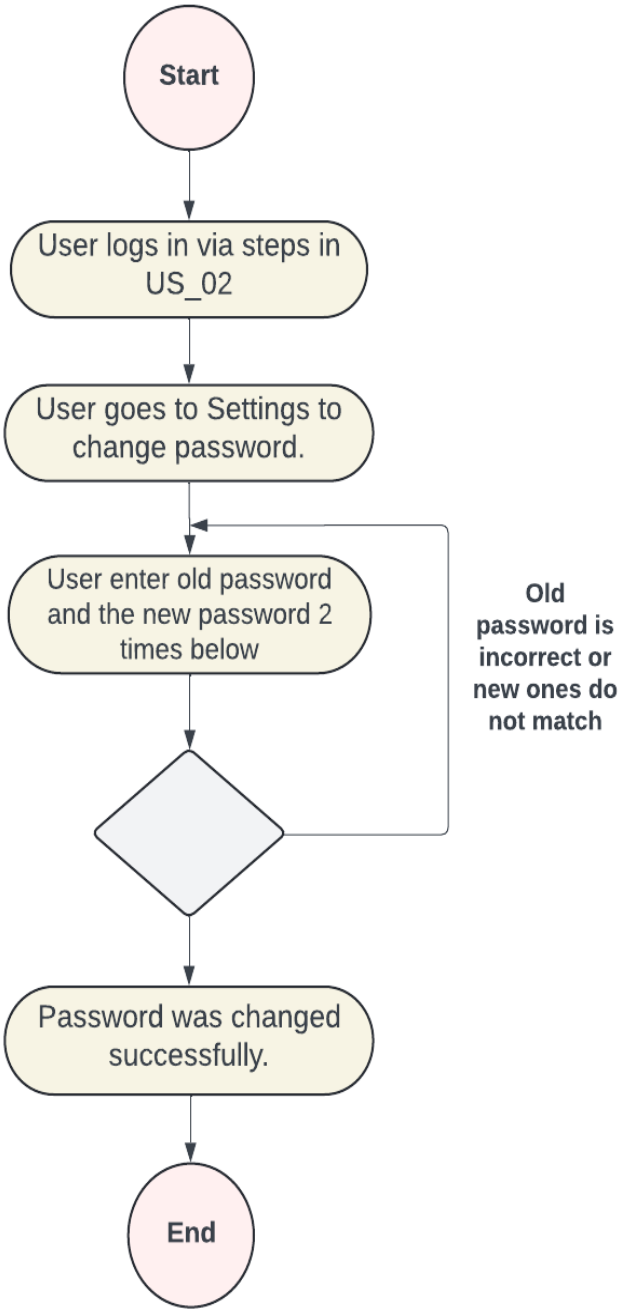
User Scenario 14 - Make an appointment



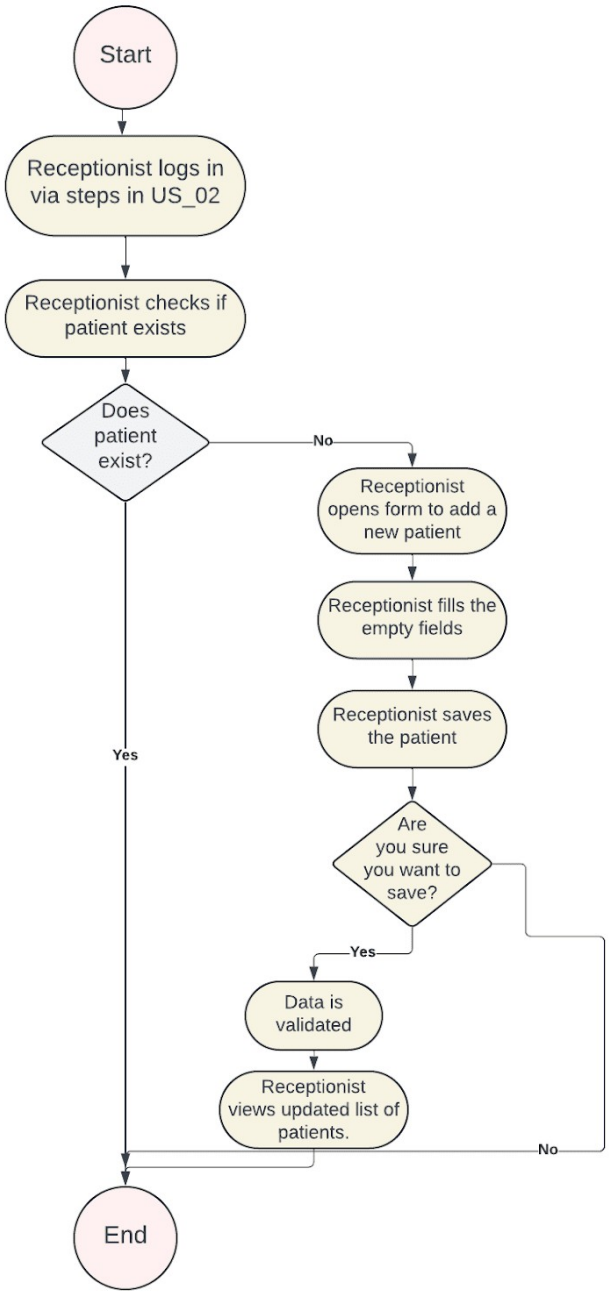
User Scenario 15 - See the timetable



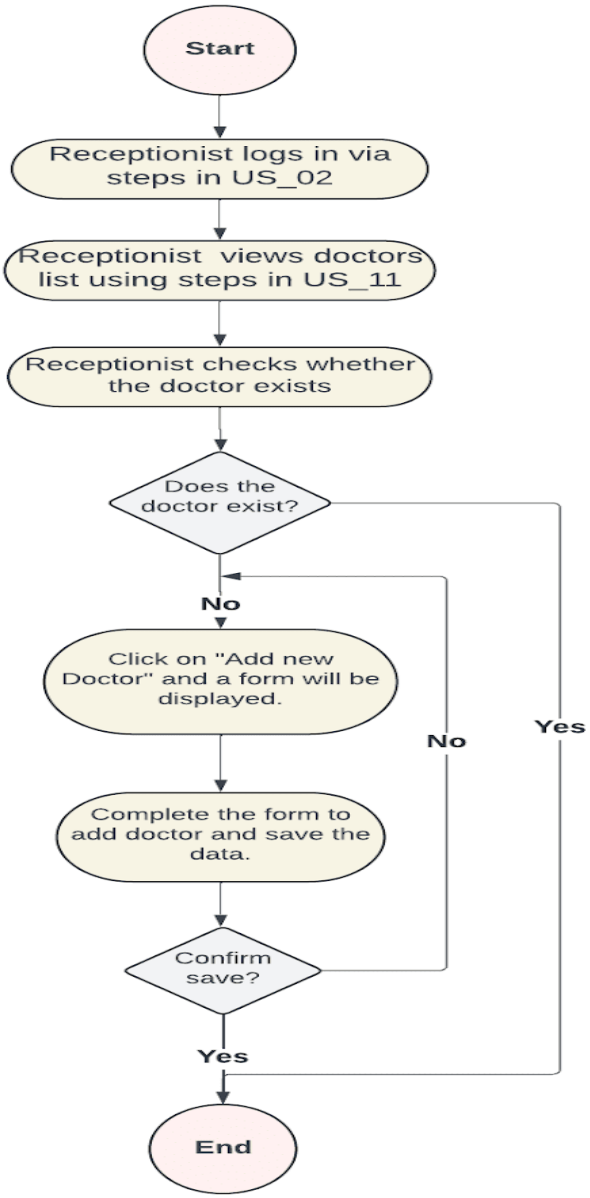
User Scenario 16 - View Statistics



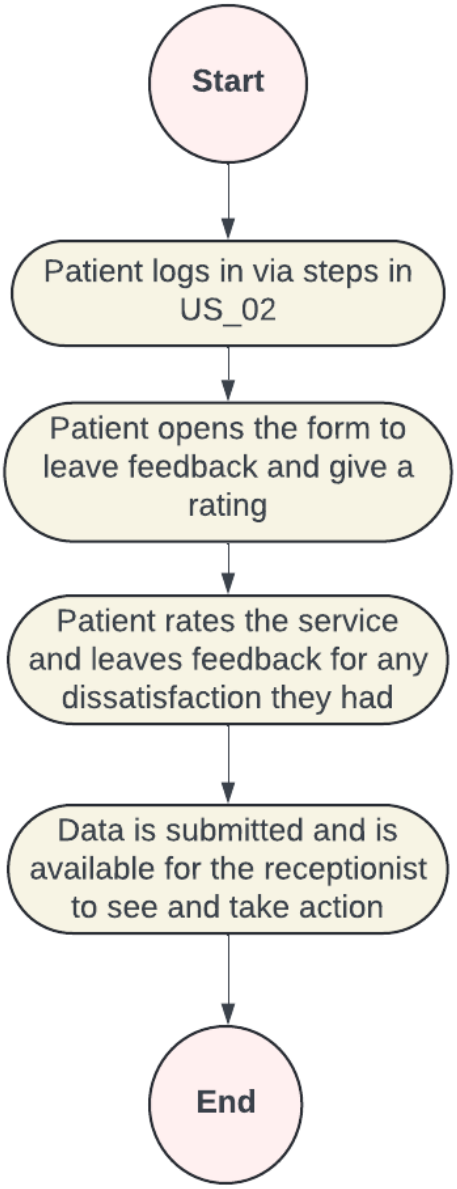
User Scenario 17 - Change Password



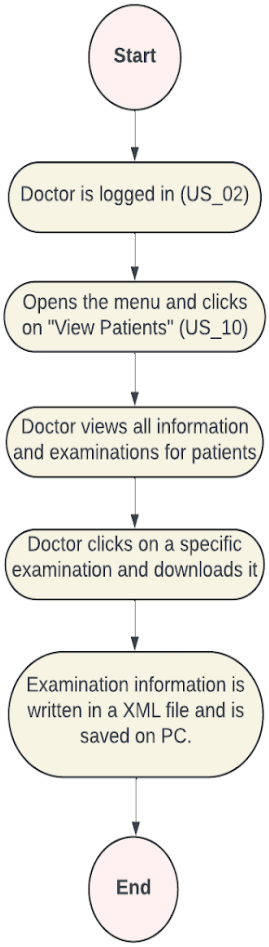
User Scenario 18 - Add a new patient user



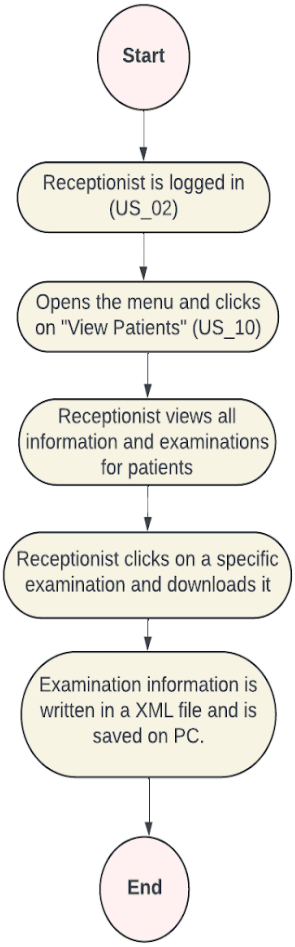
User Scenario 19 - Add a new doctor user



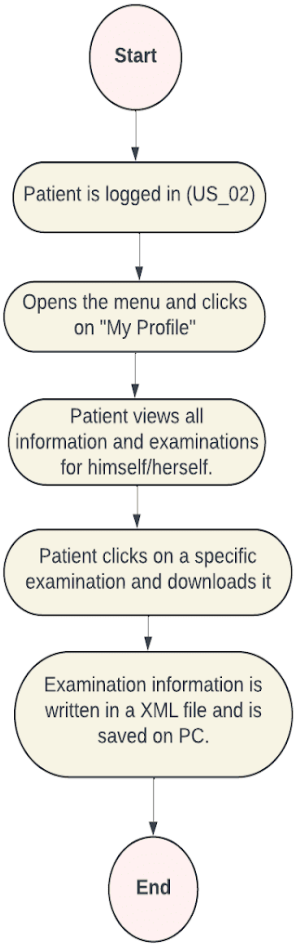
User Scenario 20 - Leave Feedback



User Scenario 21/a - Doctor downloads patient's examination

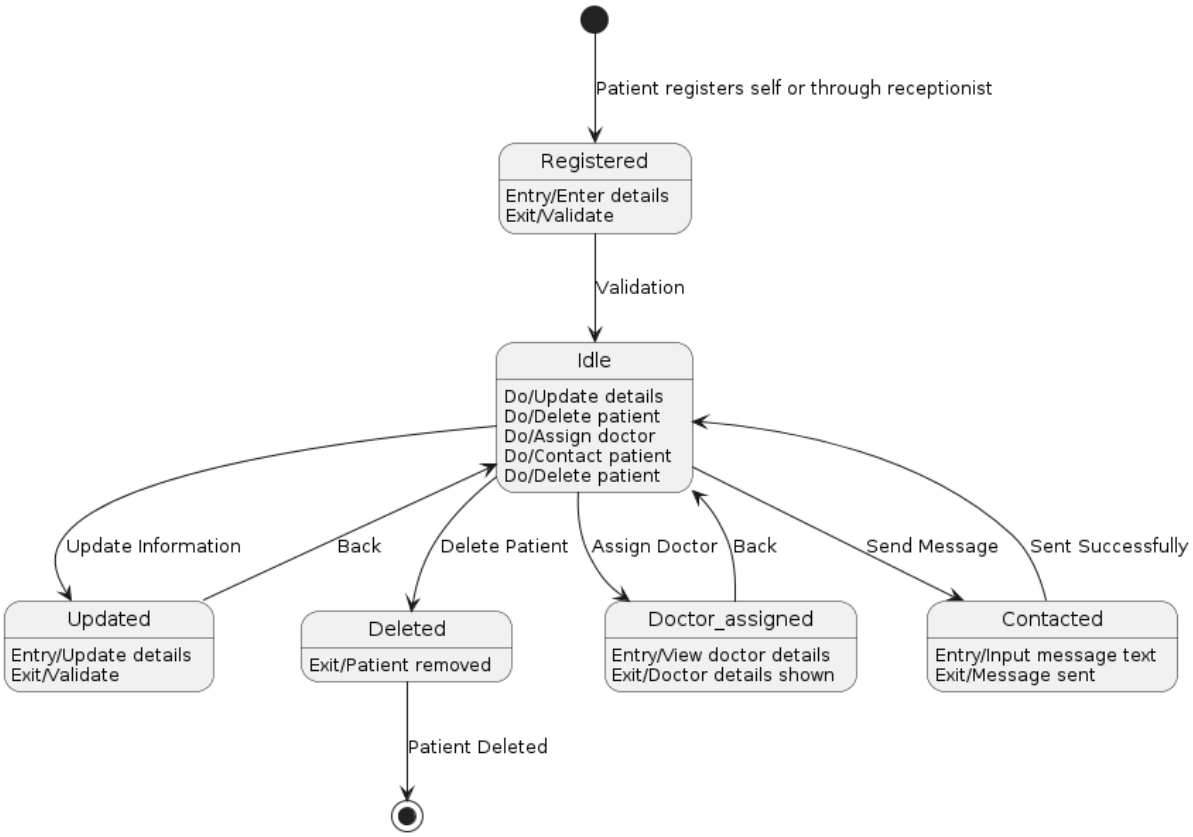


User Scenario 21/b - Receptionist downloads patient's examination

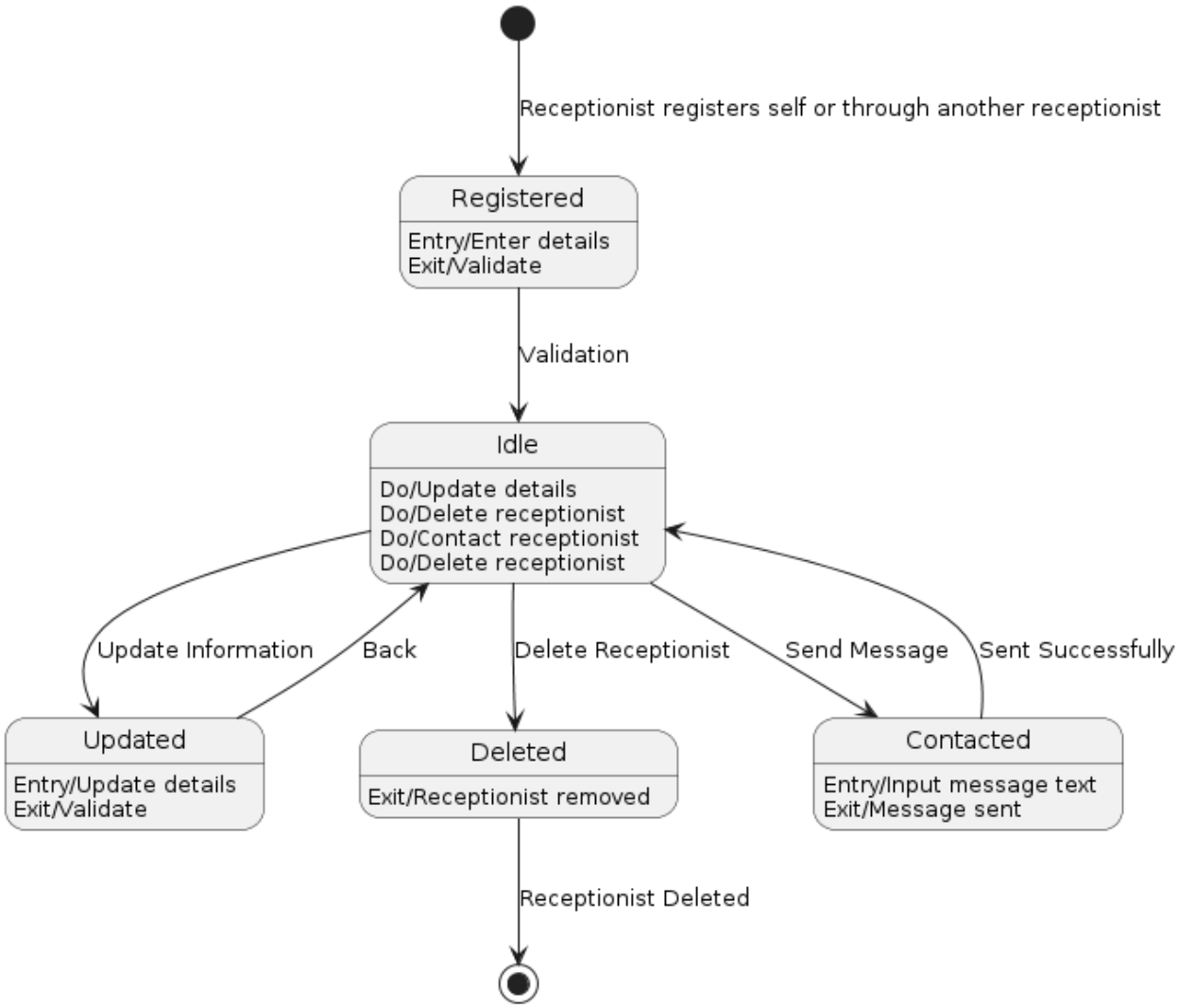


User Scenario 21/c - Patient downloads his/her examination

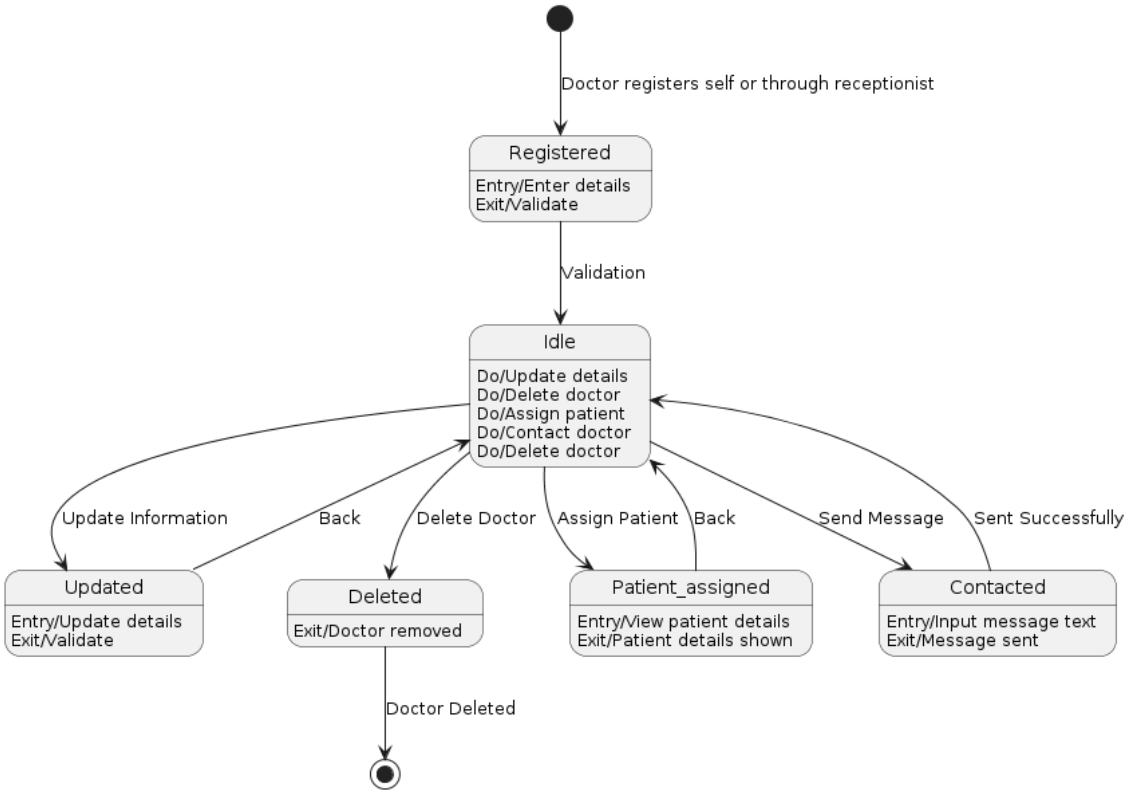
4.2.3 State Diagrams



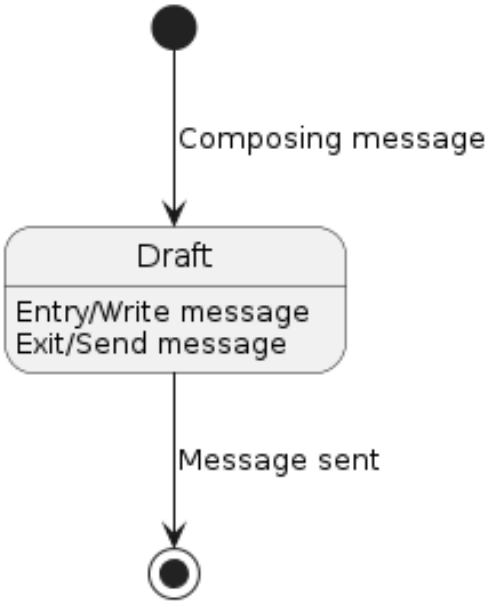
SD_01 - Patient



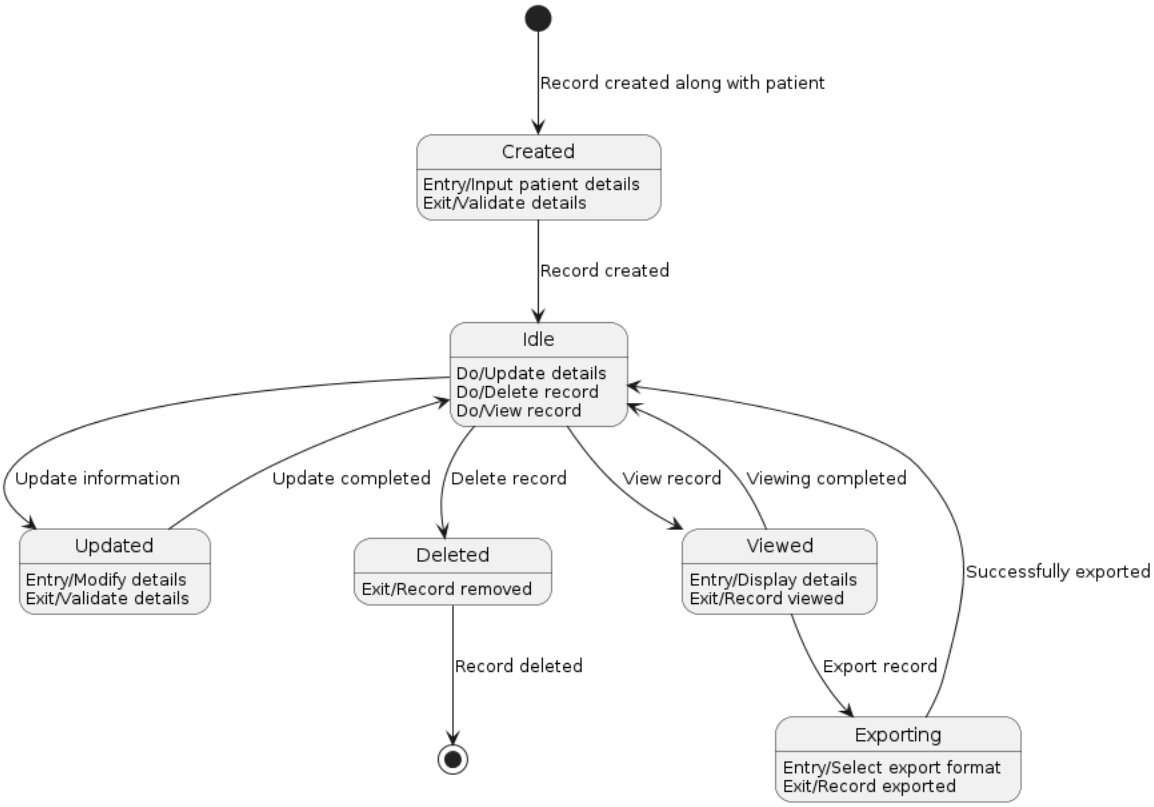
SD_02 - Receptionist



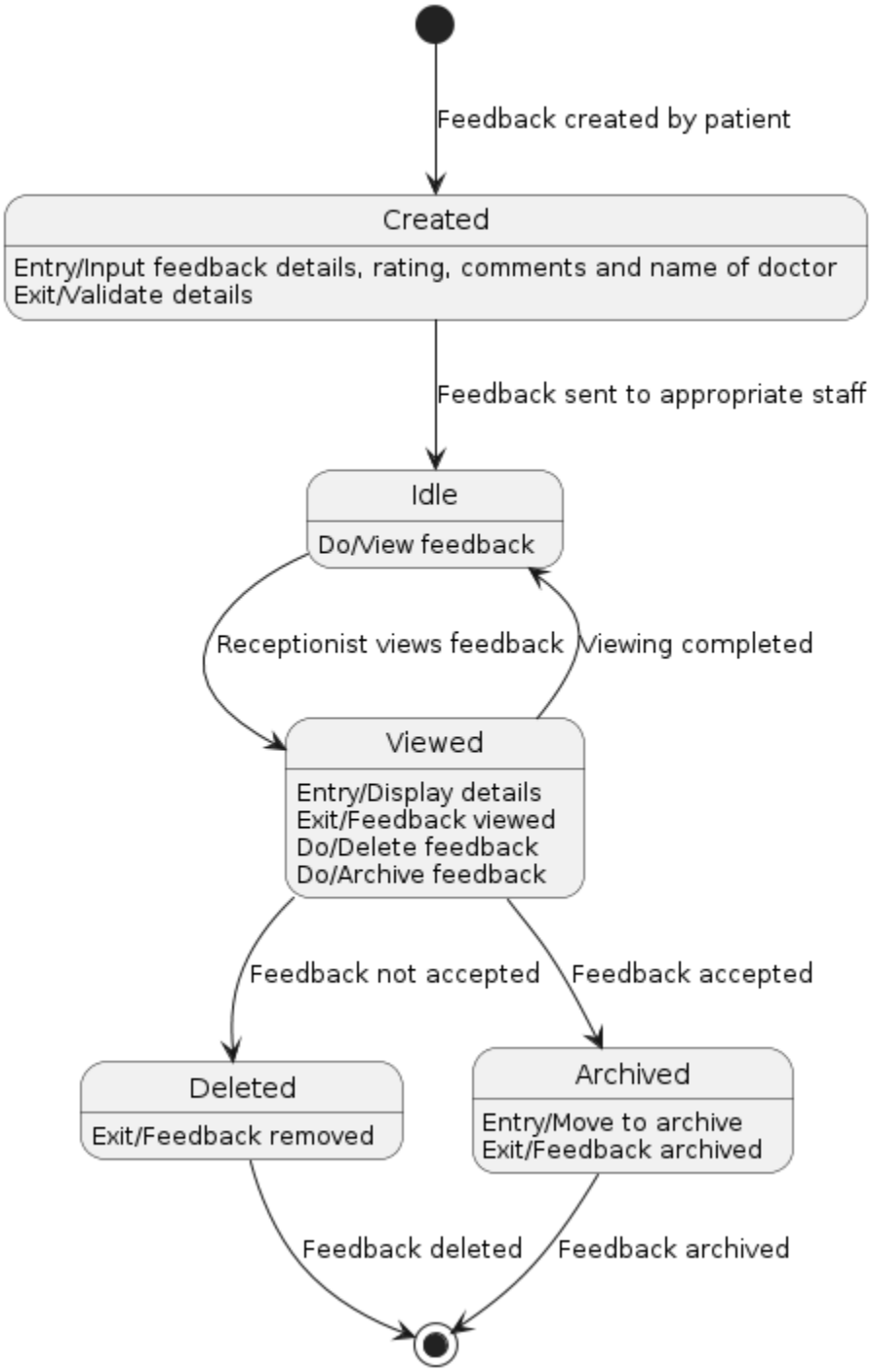
SD_03 - Doctor



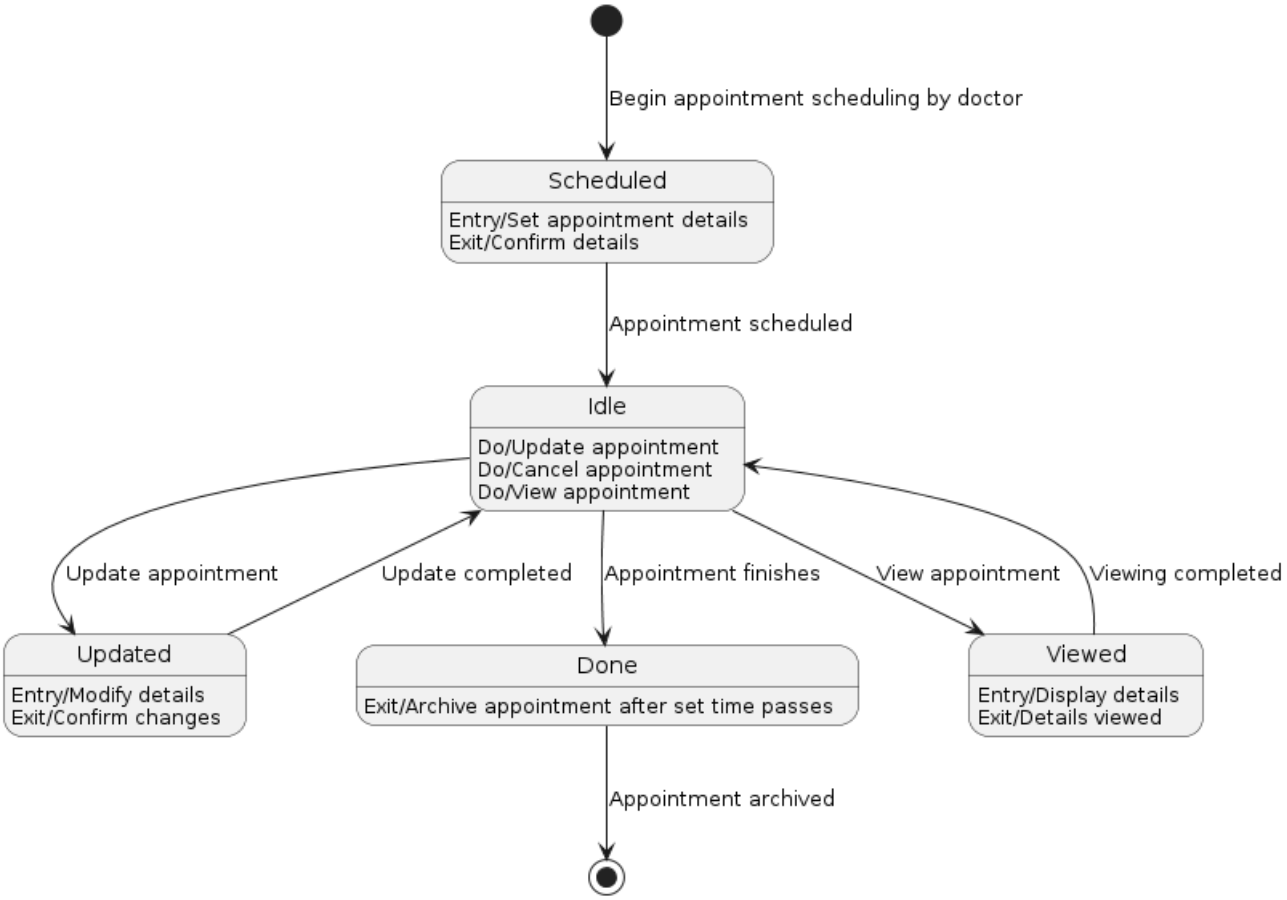
SD_04 - Message



SD_05 - Medical Record

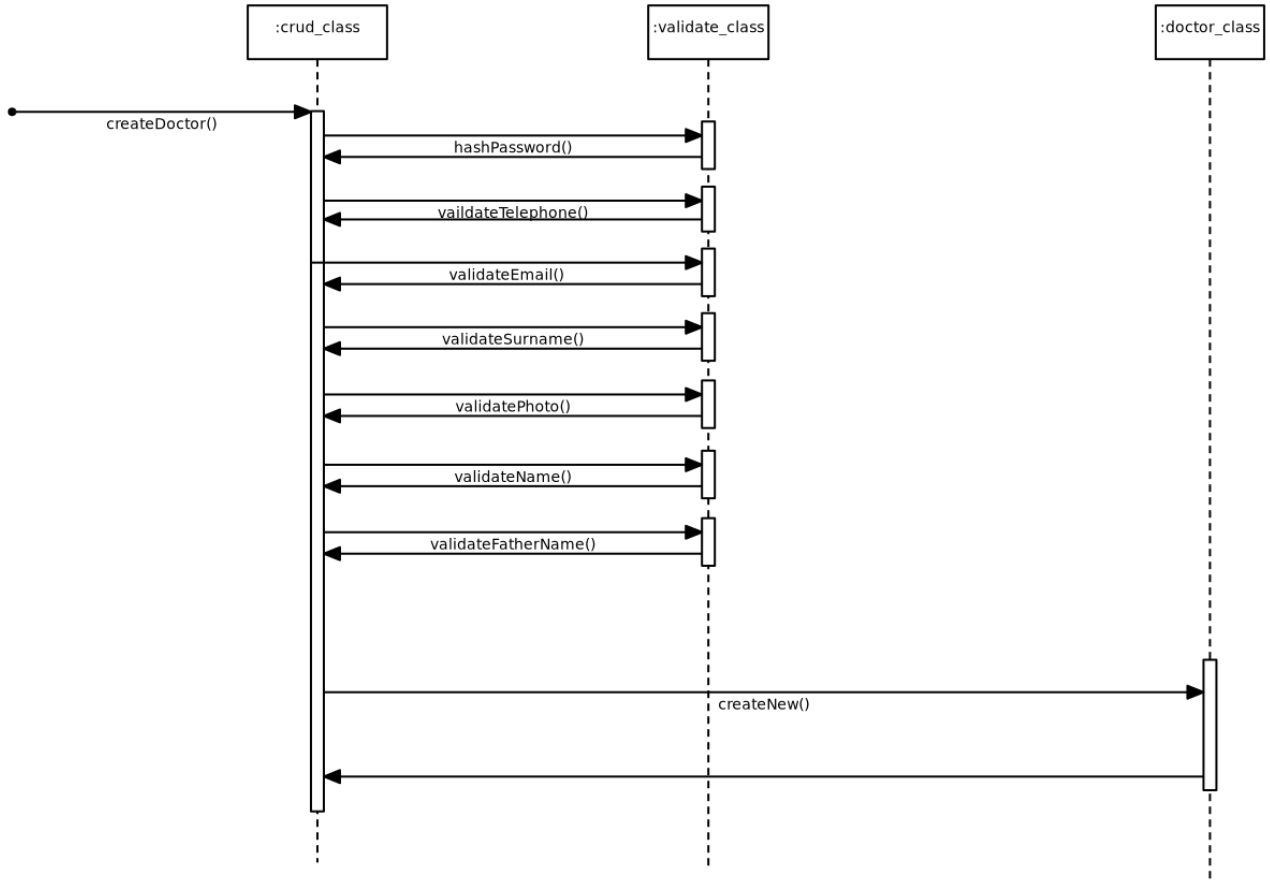


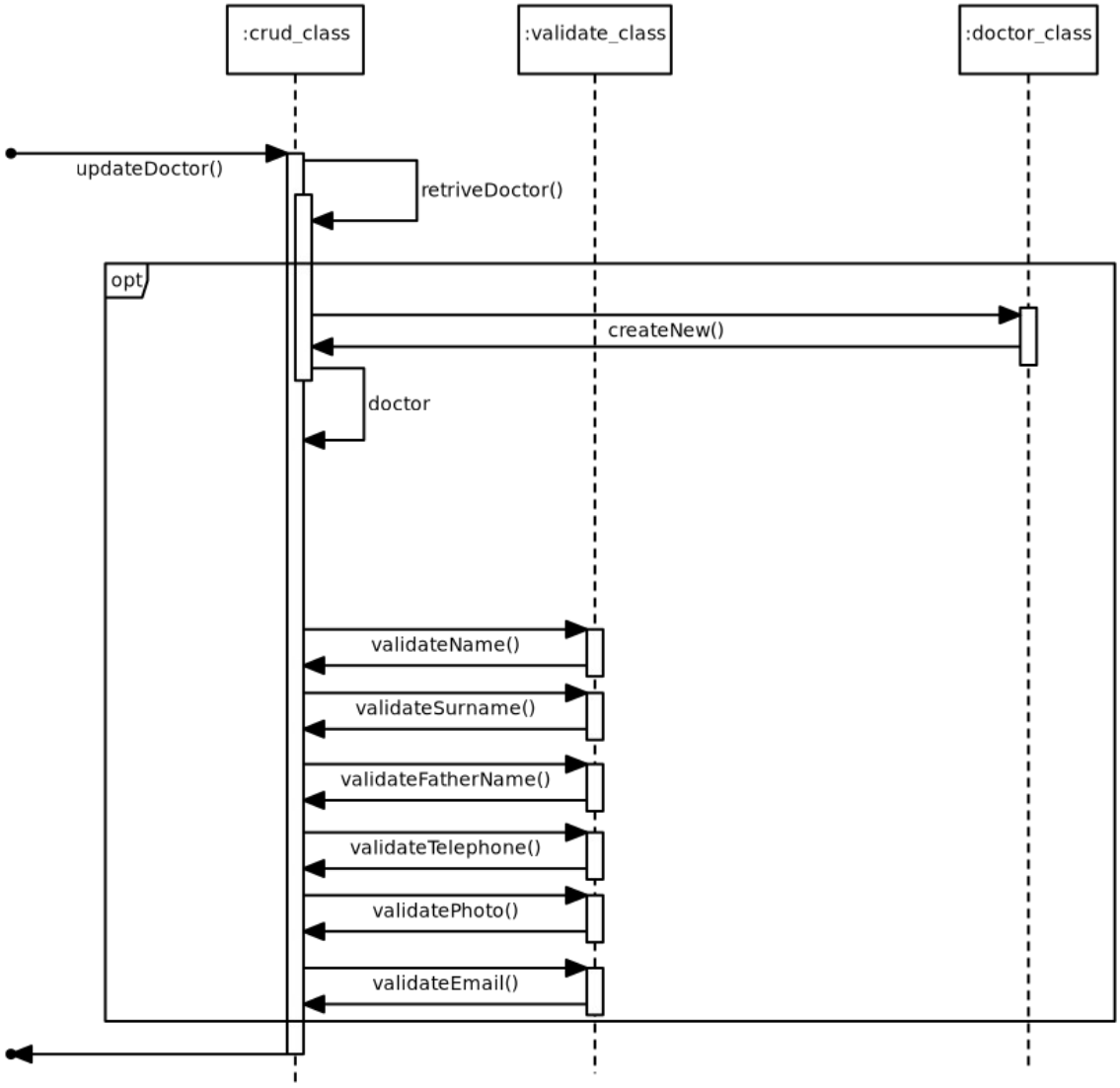
SD_06 - Feedback

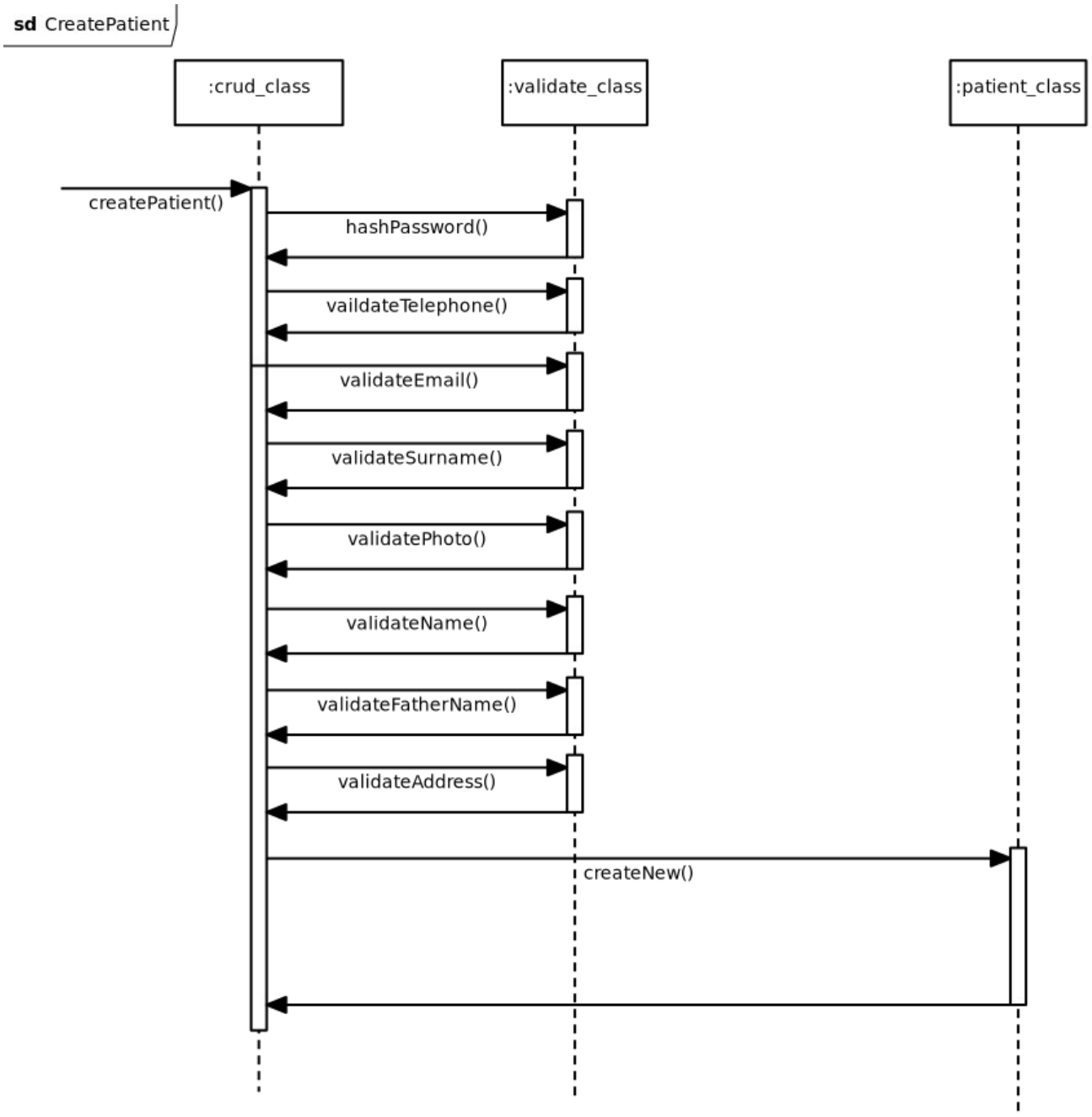


SD_07 - Appointment

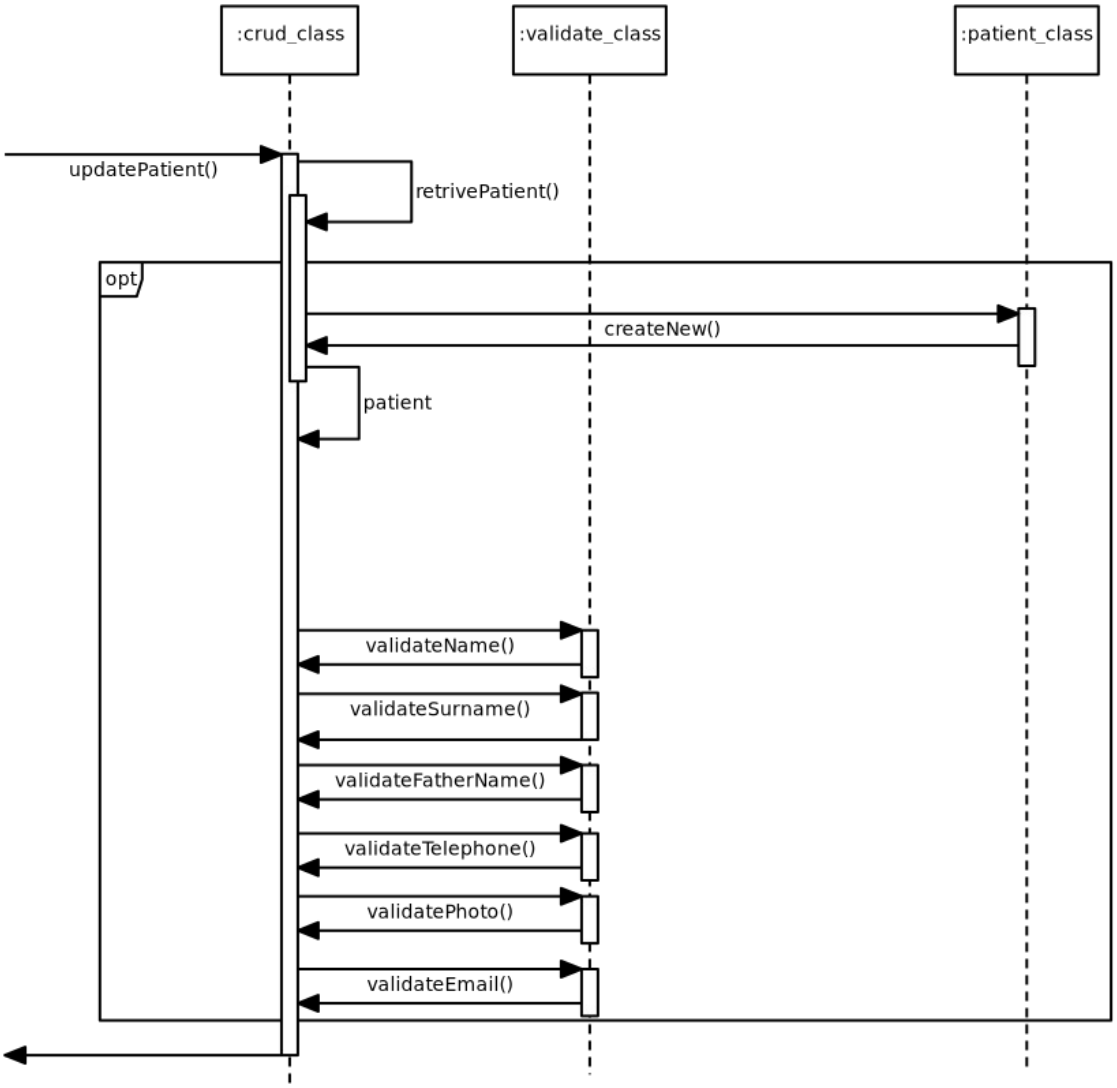
4.2.4 Sequence Diagrams

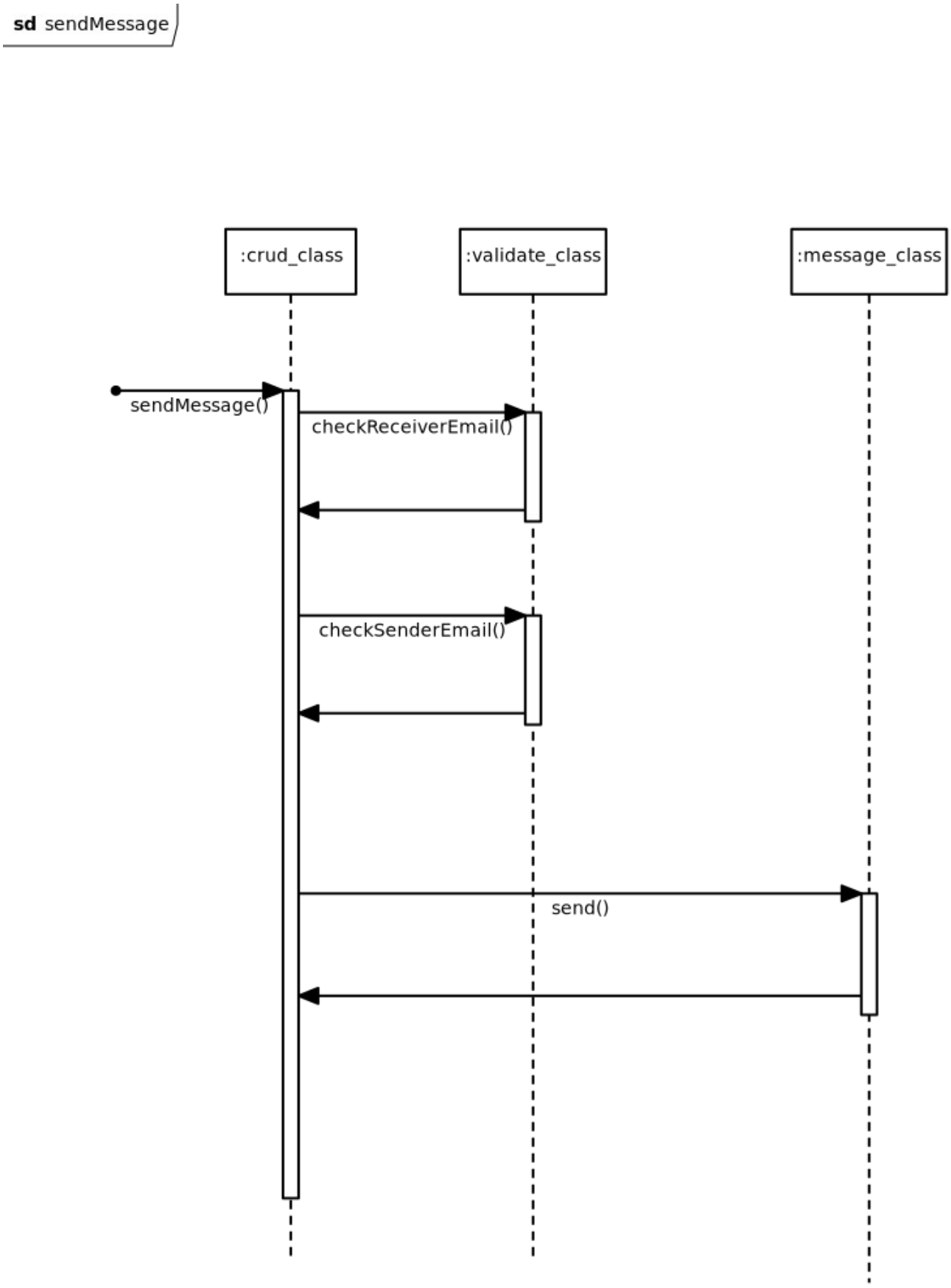


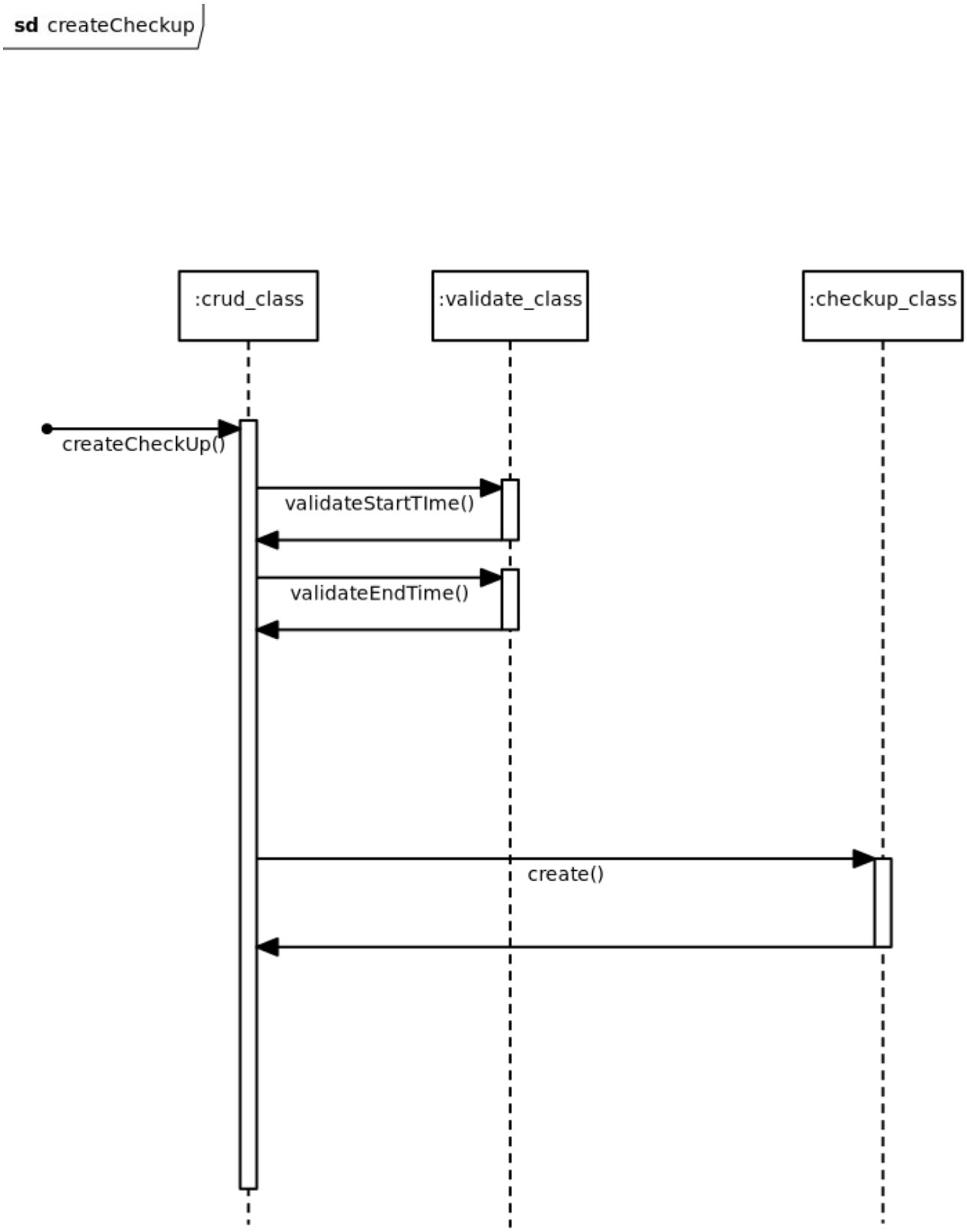

:crud_class
:validate_class
:doctor_class
updateDoctor()
retriveDoctor()
opt
createNew()
doctor
validateName()
validateSurname()
validateFatherName()
validateTelephone()
validatePhoto()
validateEmail()


sd CreatePatient
:crud_class
:validate_class
:patient_class
createPatient()
hashPassword()
vaildateTelephone()
validateEmail()
validateSurname()
validatePhoto()
validateName()
validateFatherName()
validateAddress()
createNew()

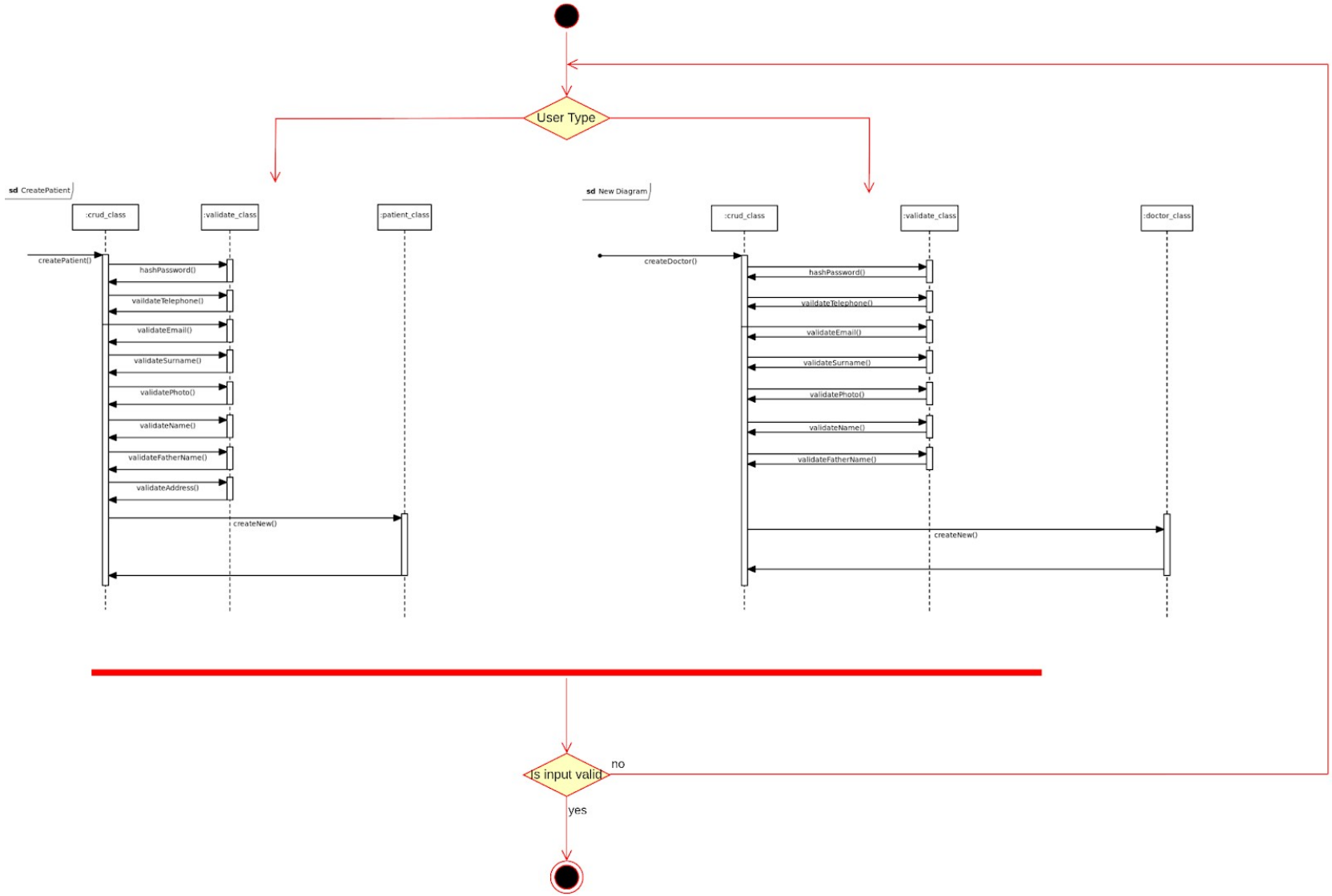
sd updatePatient

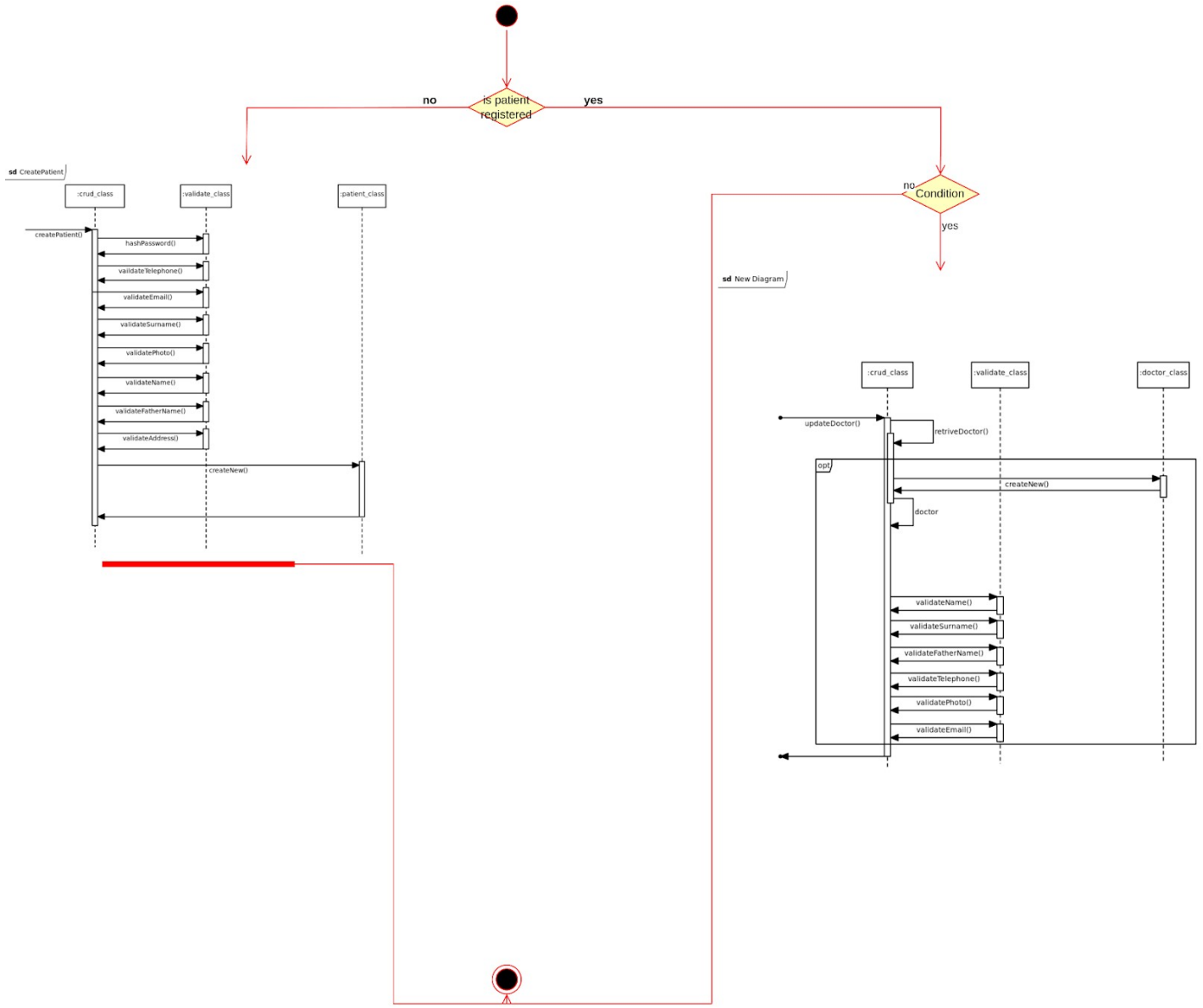



sd sendMessage
:crud_class
:validate_class
:message_class
sendMessage()
checkReceiverEmail()
checkSenderEmail()
send()

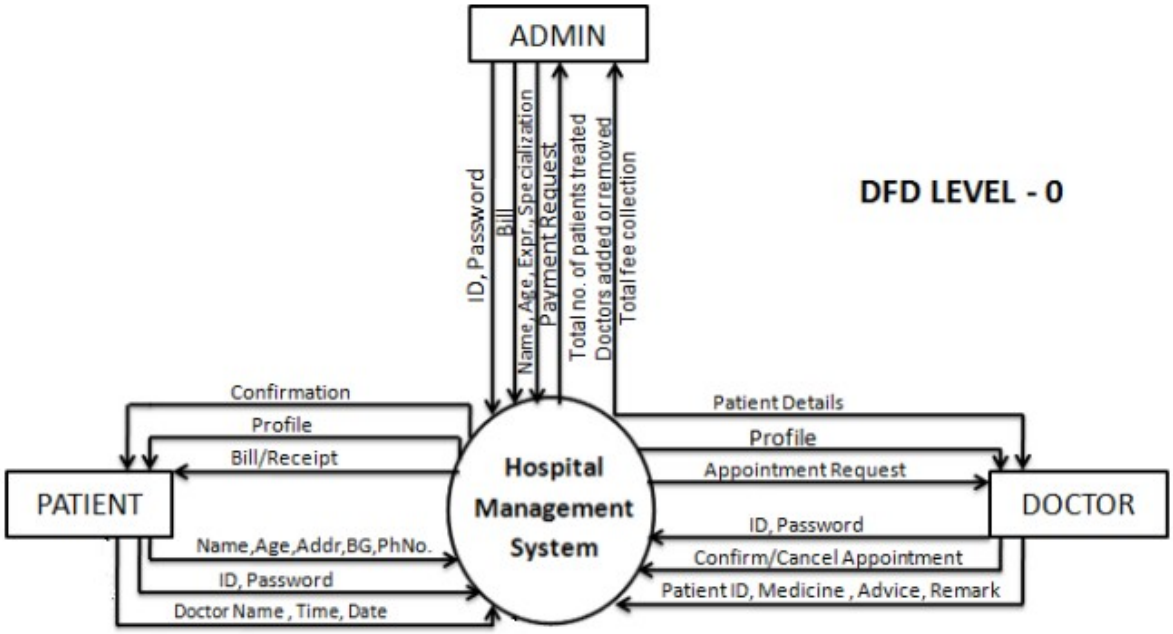

sd createCheckup
:crud_class
:validate_class
:checkup_class
createCheckUp()
validateStartTIme()
validateEndTime()
create()

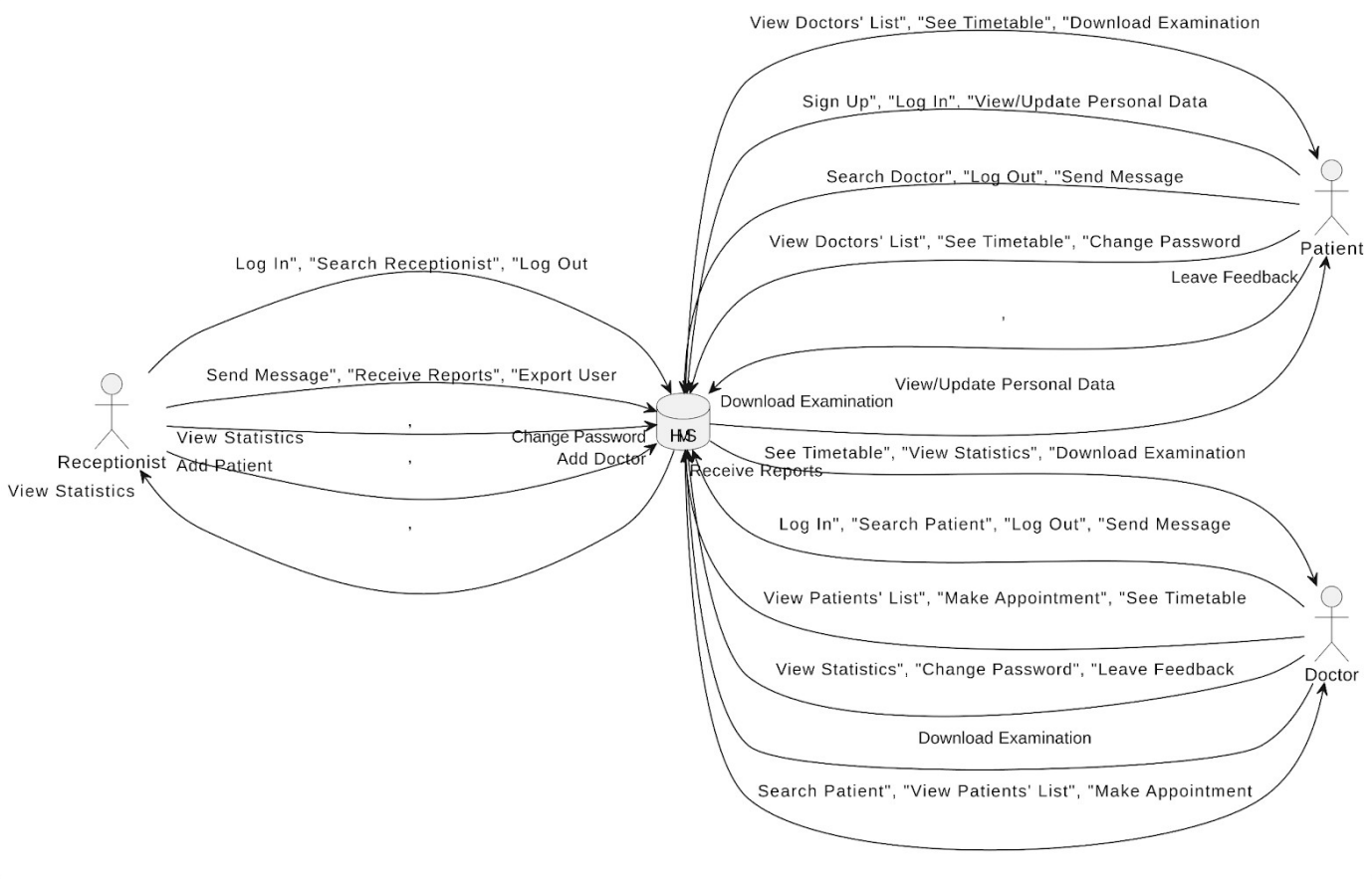
4.2.5 Interaction Diagrams



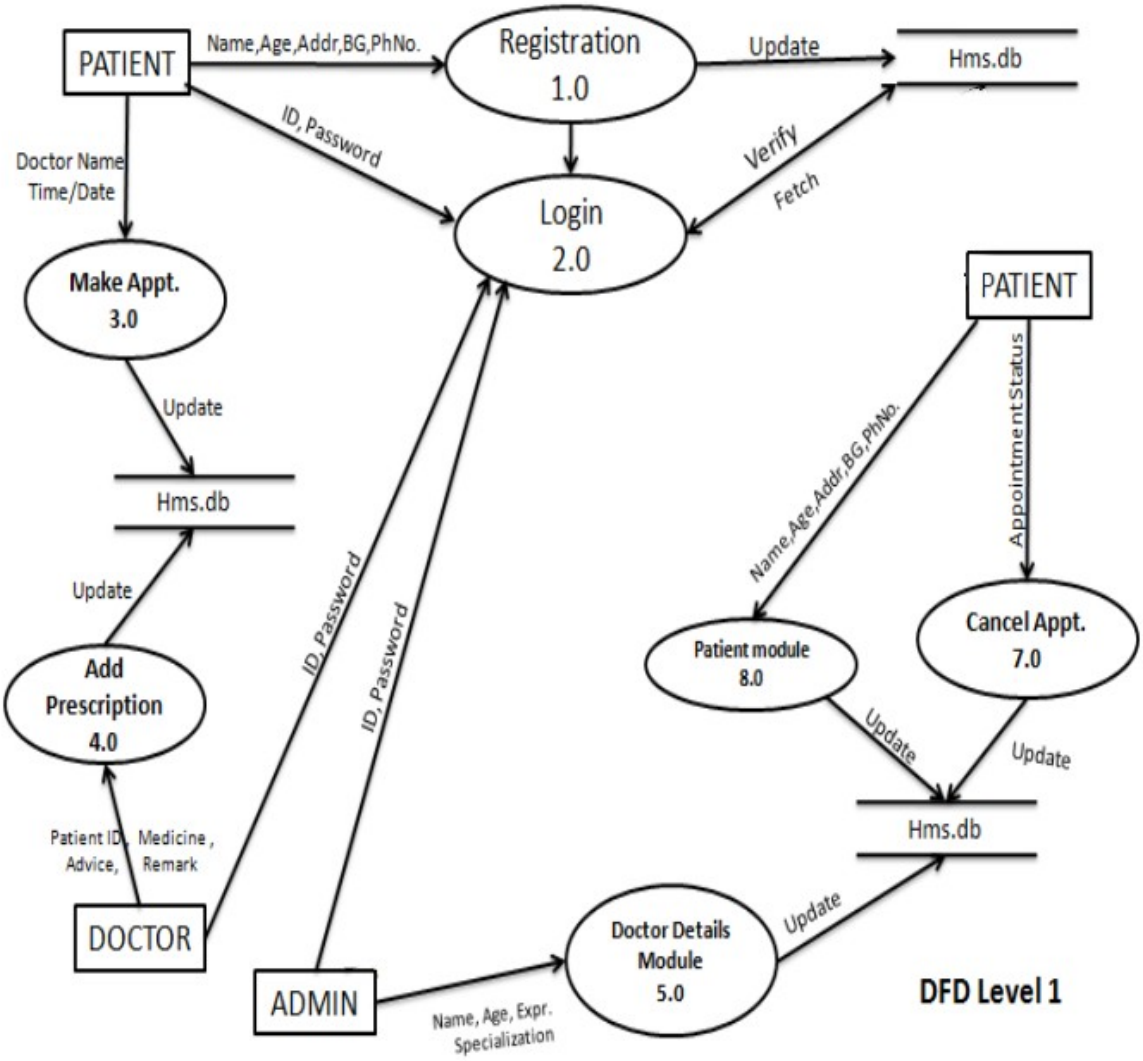

is patient registered
no
yes
sd CreatePatient
:crud_class
:validate_class
:patient_class
createPatient()
hashPassword()
validateTelephone()
validateEmail()
validateSurname()
validatePhoto()
validateName()
validateFatherName()
validateAddress()
createNew()
Condition
no
yes
sd New Diagram
:crud_class
:validate_class
:doctor_class
updateDoctor()
retriveDoctor()
opt
createNew()
doctor
validateName()
validateSurname()
validateFatherName()
validateTelephone()
validatePhoto()
validateEmail()

4.3 Data Flow Diagrams

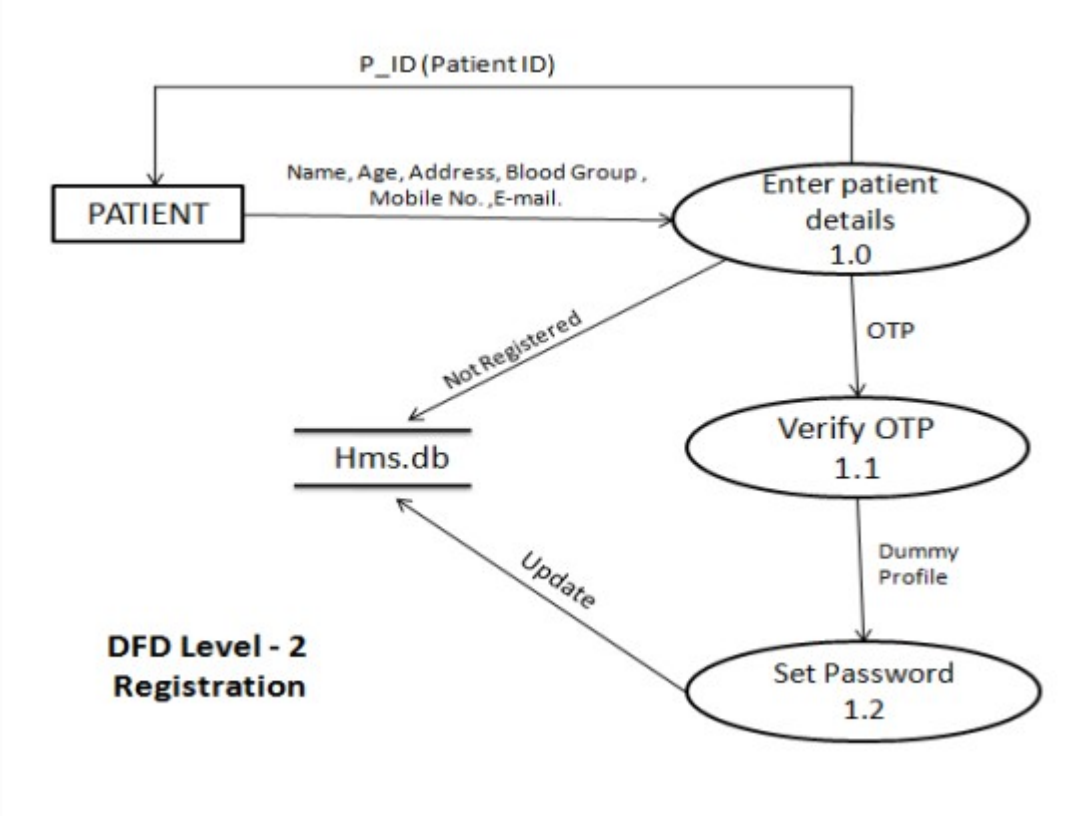




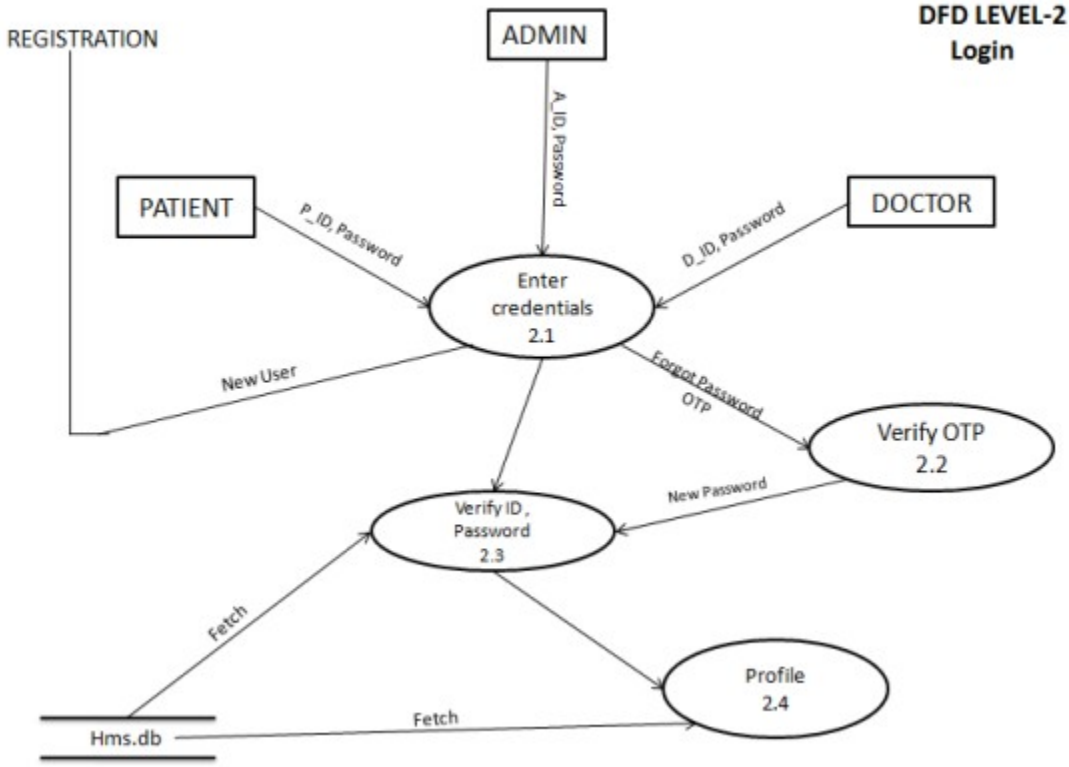
Context level DFD


PATIENT
Name,Age,Addr,BG,PhNo.
Registration
1.0
Update
Hms.db
ID, Password
Doctor Name
Time/Date
Verify
Fetch
Login
2.0
Make Appt.
3.0
PATIENT
Update
Hms.db
Name,Age,Addr,BG,PhNo.
AppointmentStatus
Update
ID, Password
ID, Password
Patient module
8.0
Cancel Appt.
7.0
Add
Prescription
4.0
Update
Update
Hms.db
Patient ID, Medicine ,
Advice, Remark
DOCTOR
ADMIN
Doctor Details
Module
5.0
Update
Name, Age, Expr.
Specialization
DFD Level 1

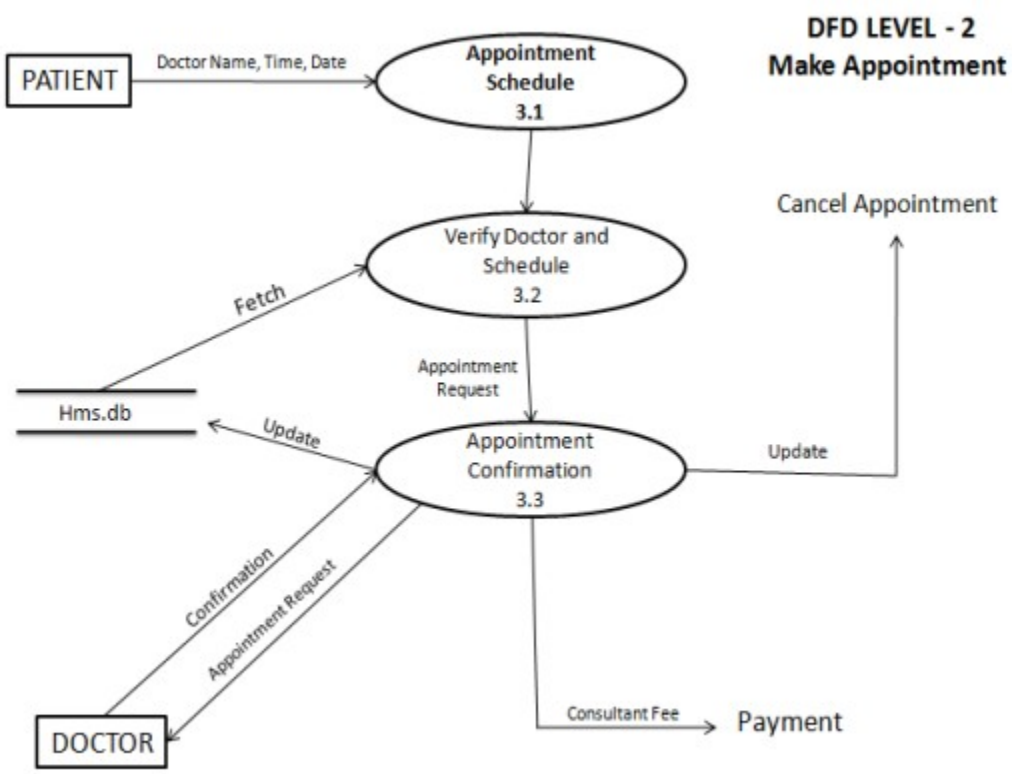
Level 1 DFD



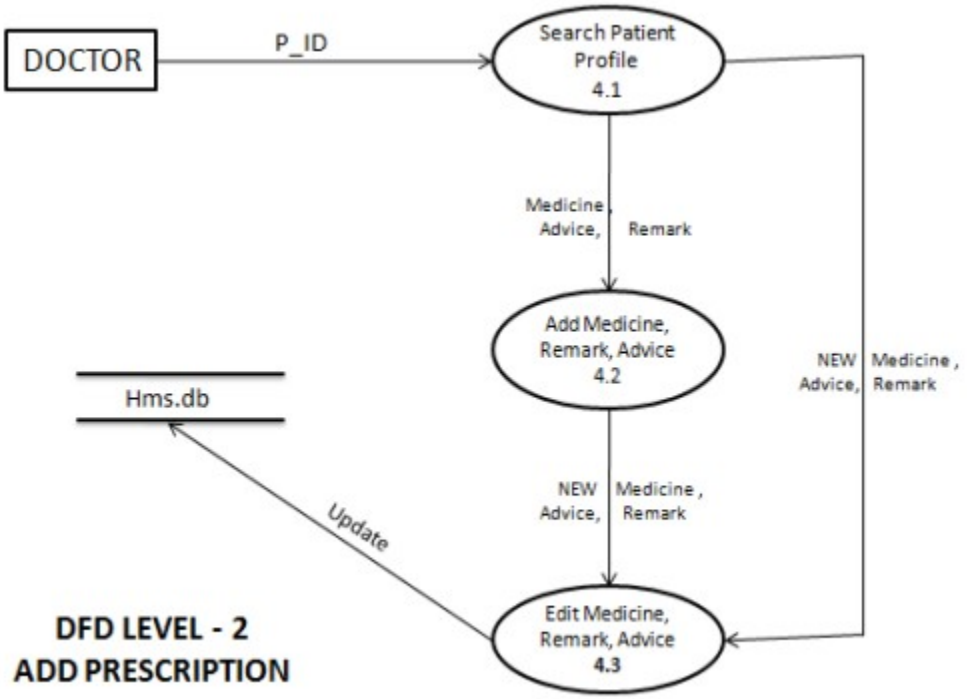
Level 2 Registration



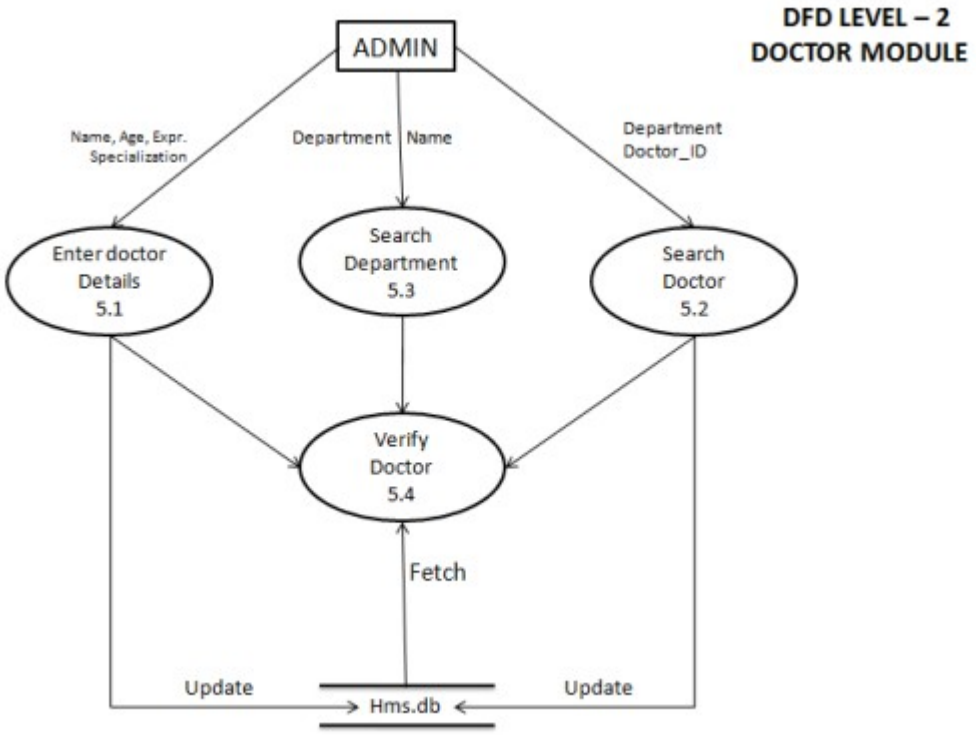
Level 2 Login



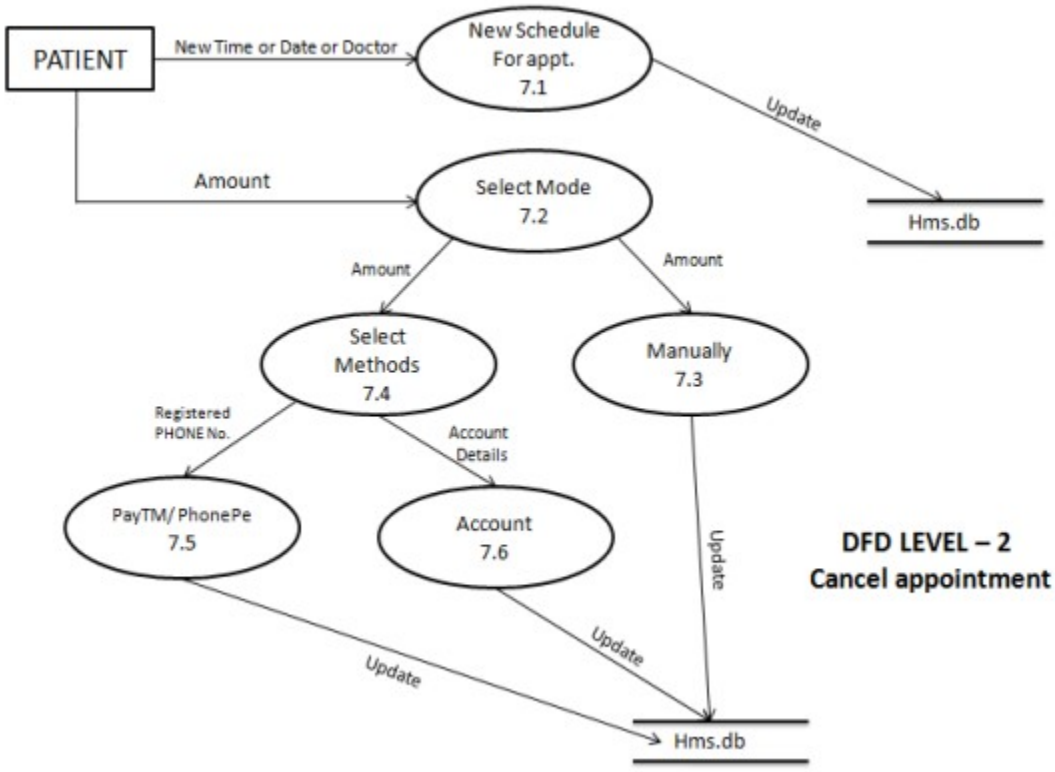
Level 2 Make Appointment



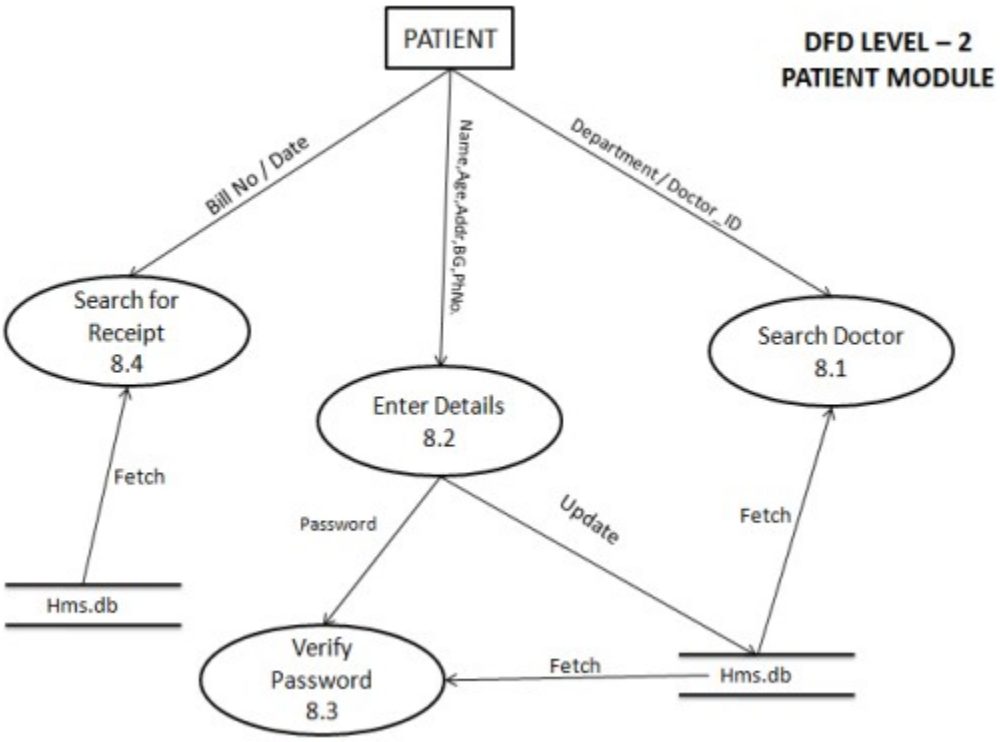
Level 2 Add Description



Level 2 Doctor Module



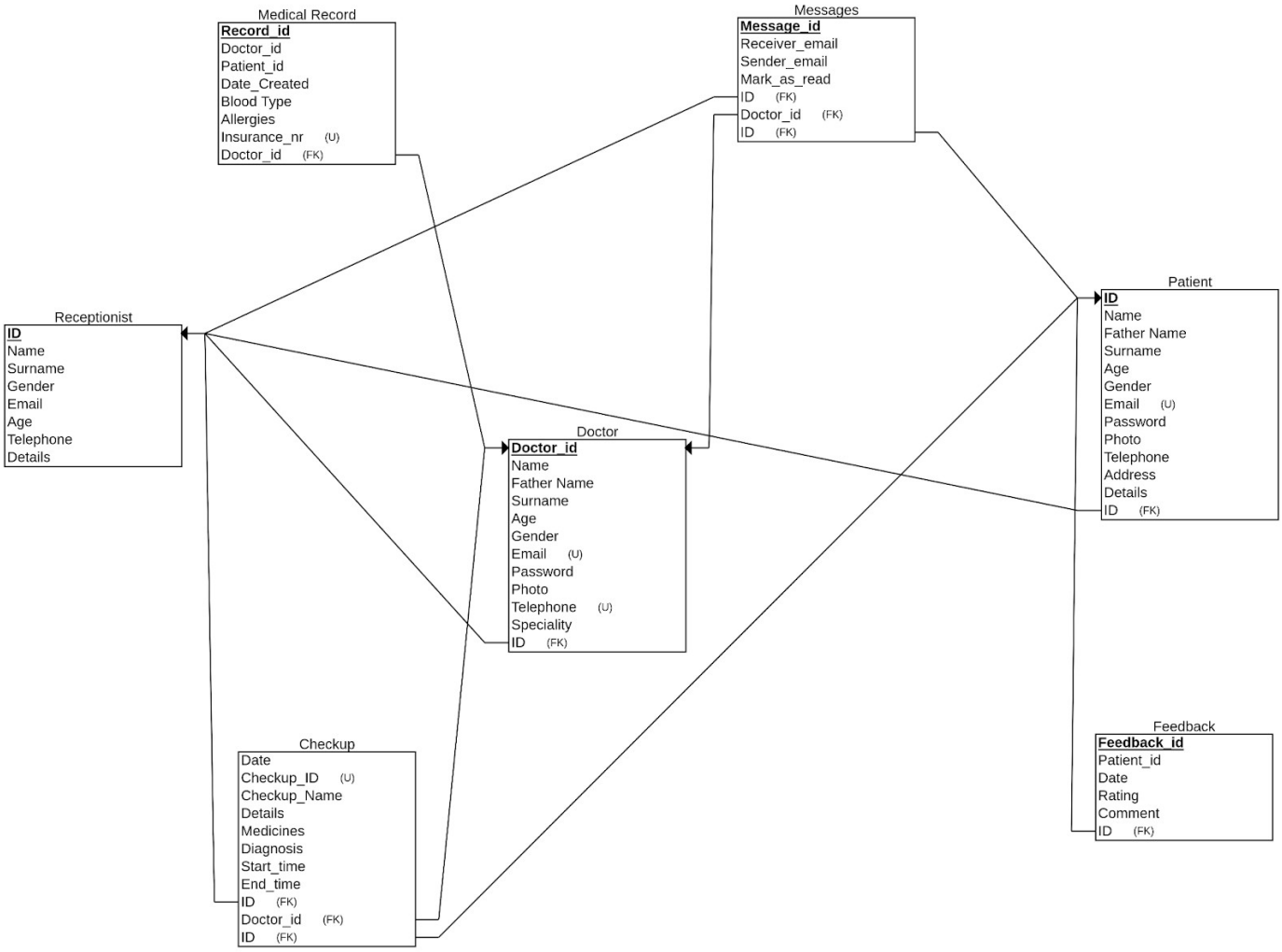
Level 2 Cancel Appointment



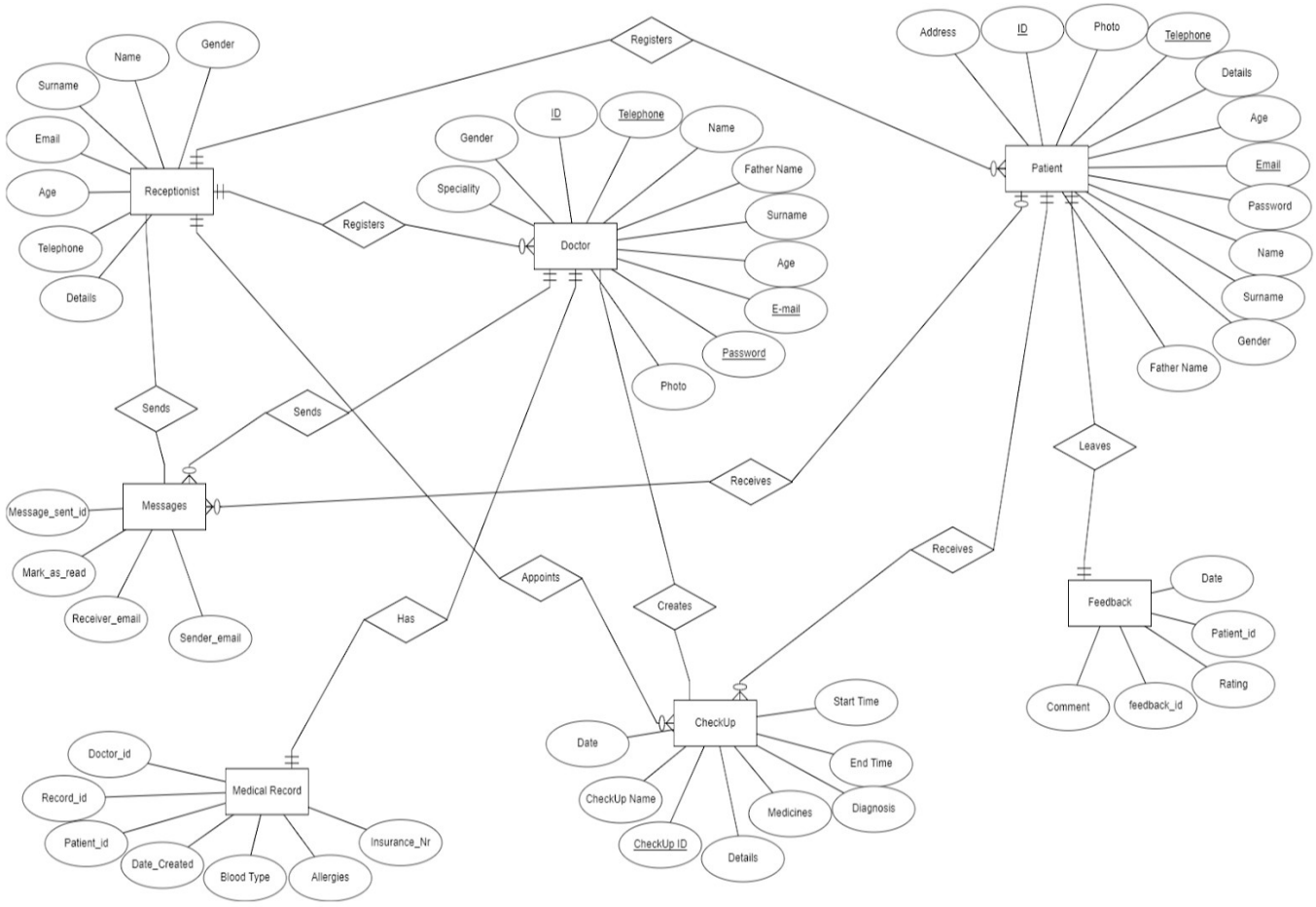
Level 2 Patient Module

4.4 Entity Relation

4.4.1 Database Schema Design

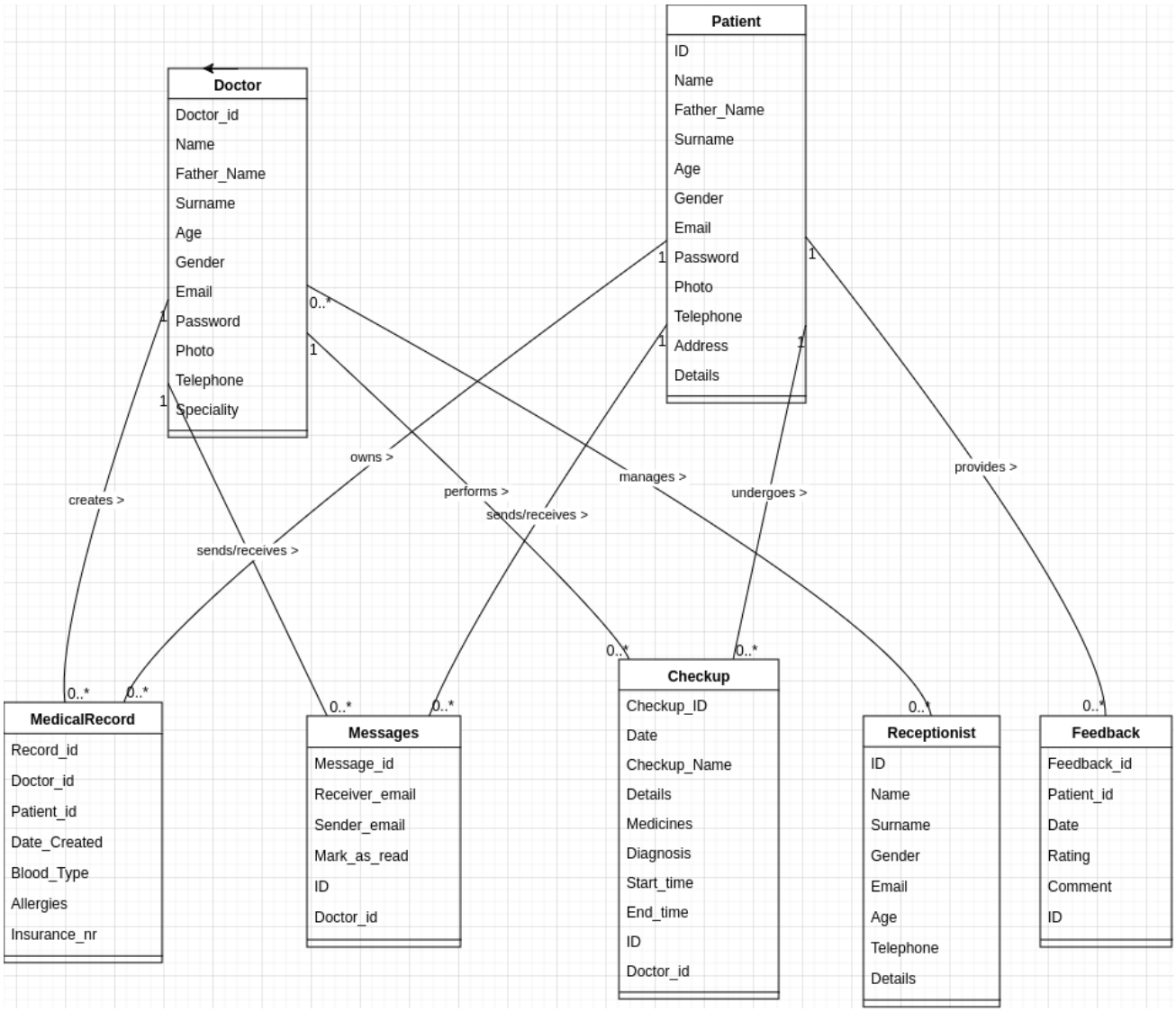


4.4.2 Entity Relation Diagram

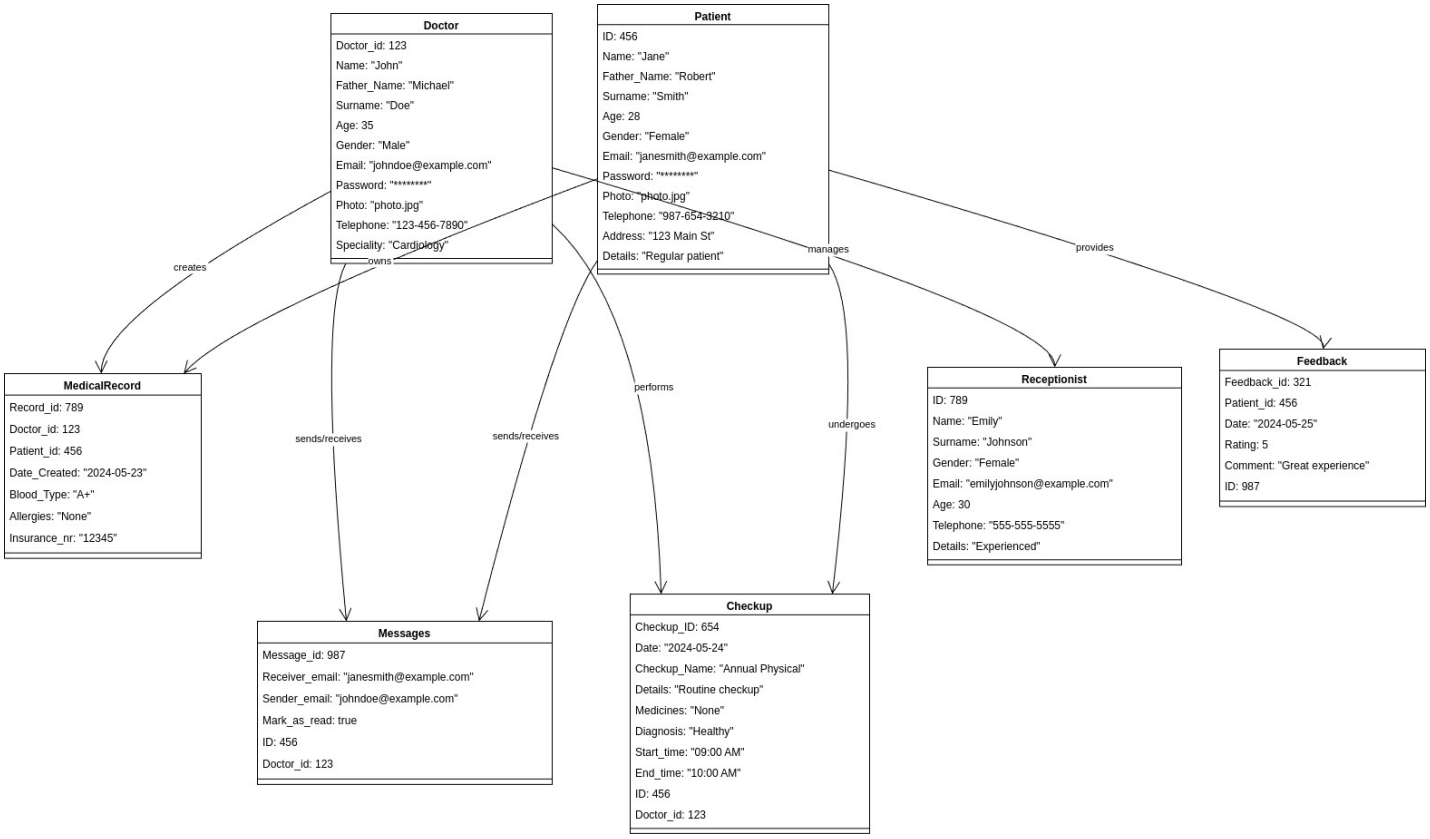


4.5 Structural Diagrams

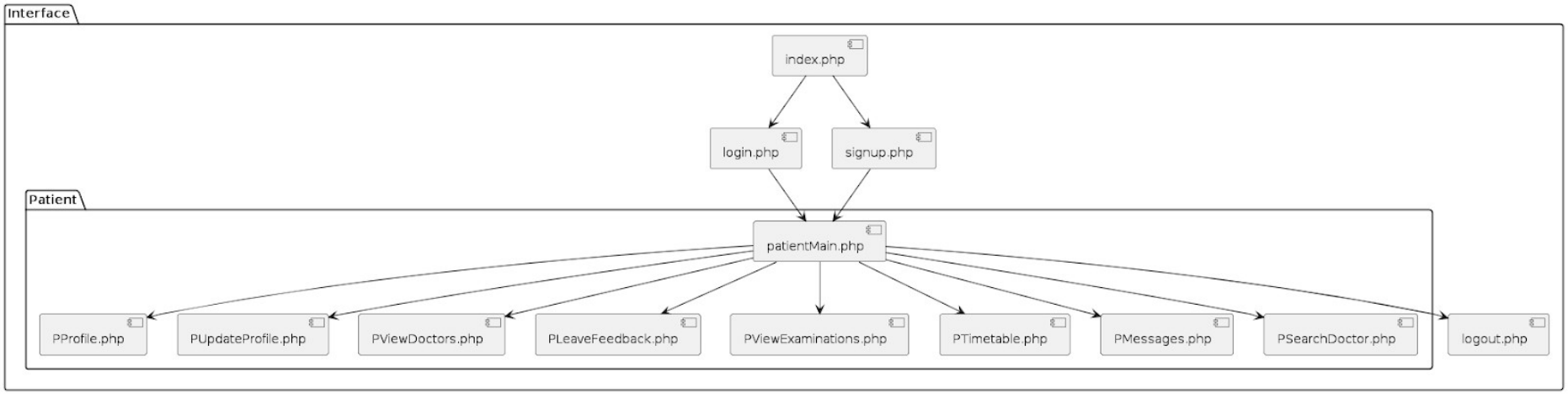
4.5.1 Class Diagram



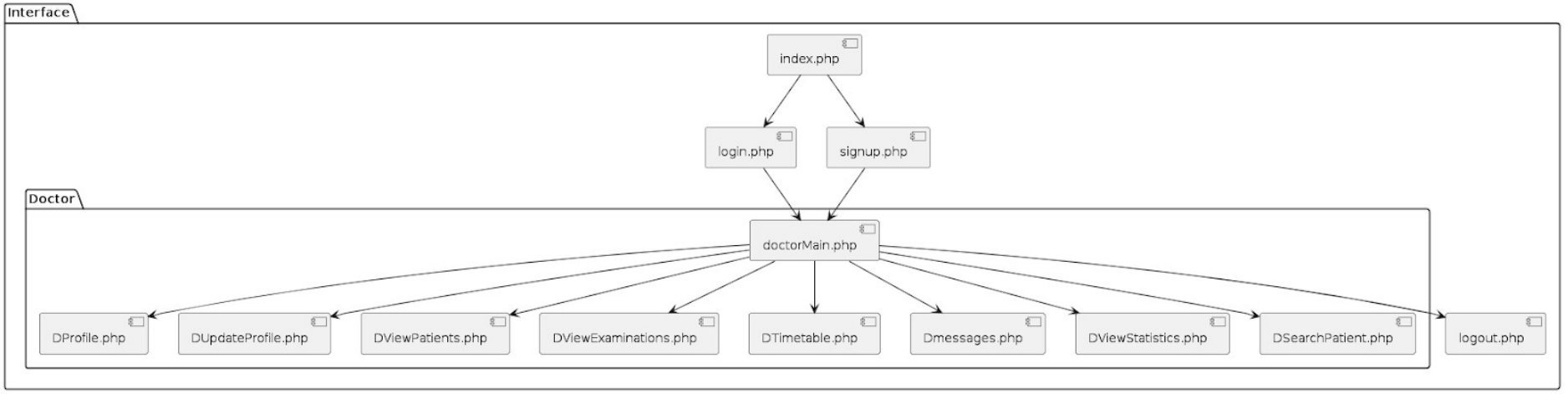
4.5.2 Object Diagram



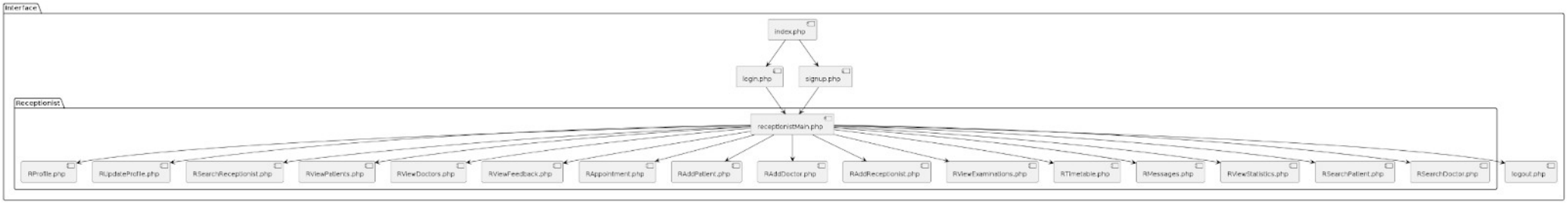
4.5.3 Component Diagrams



CoD_01 - Interface Package (Patient)



CoD_02 - Interface Package (Doctor)



CoD_03 - Interface Package (Receptionist)

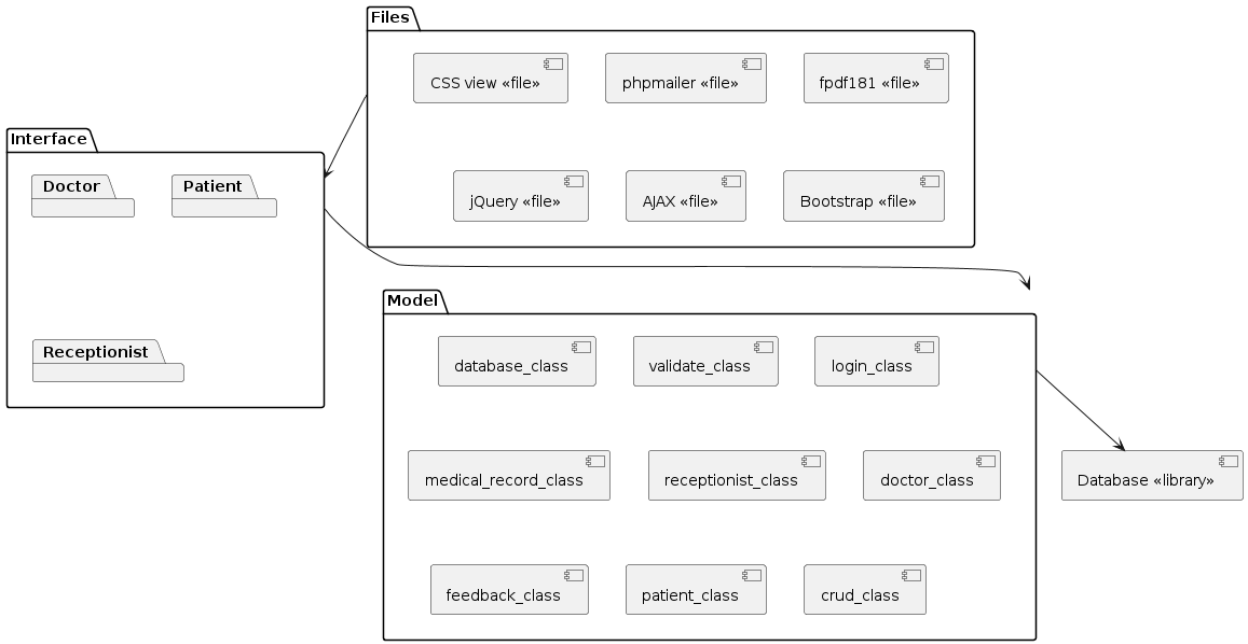
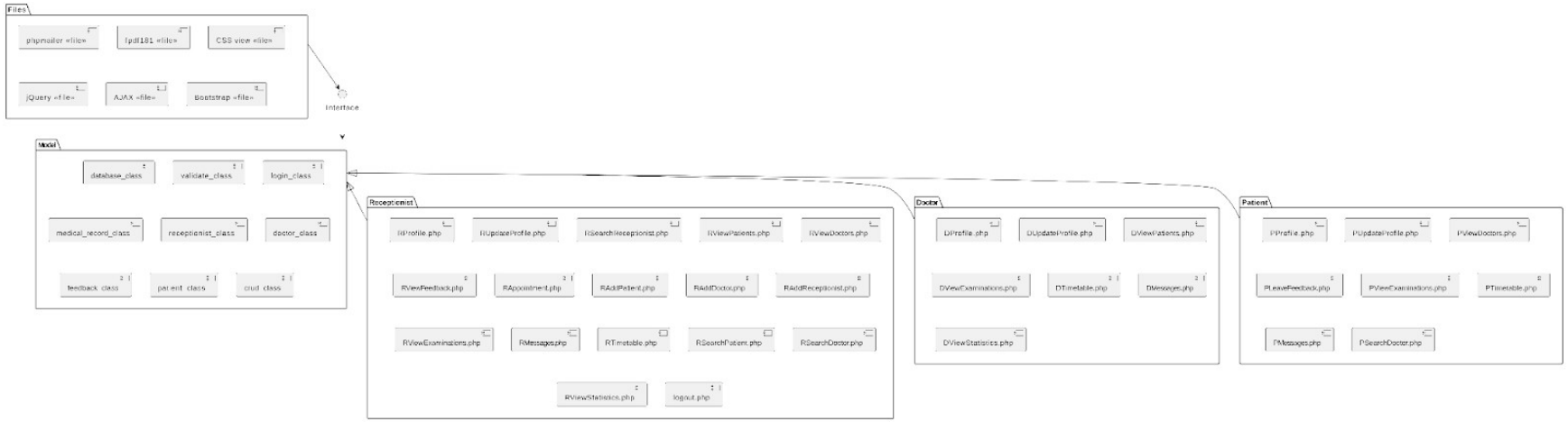
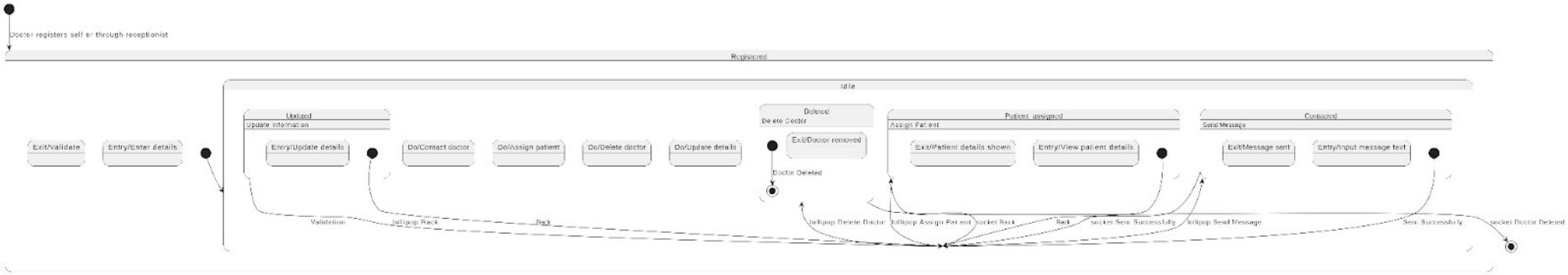
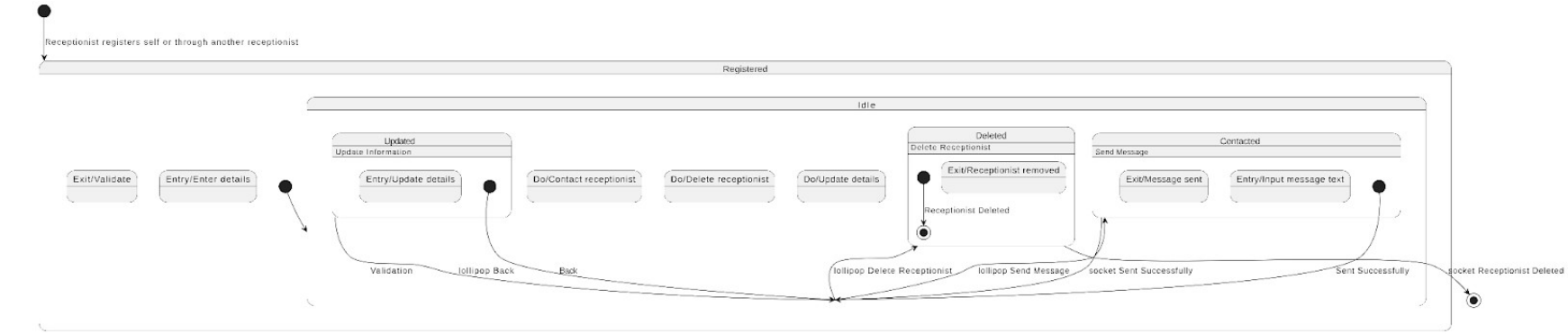


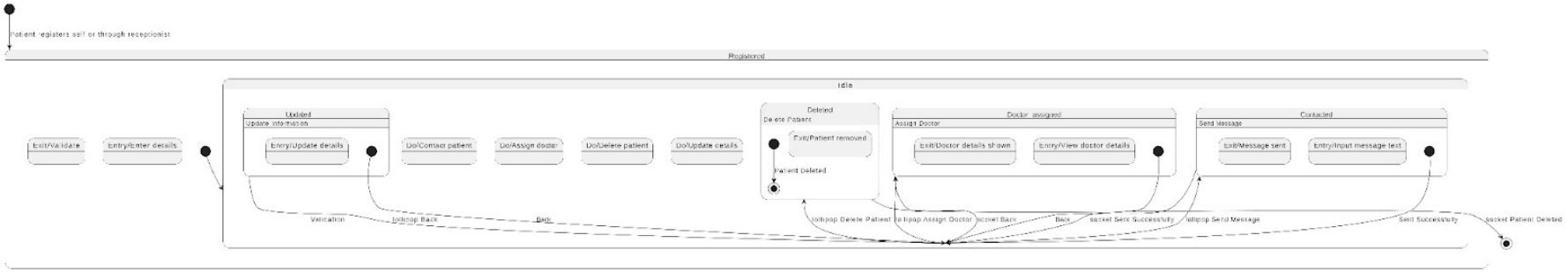
Figure 1: CoD_04 - Other packages

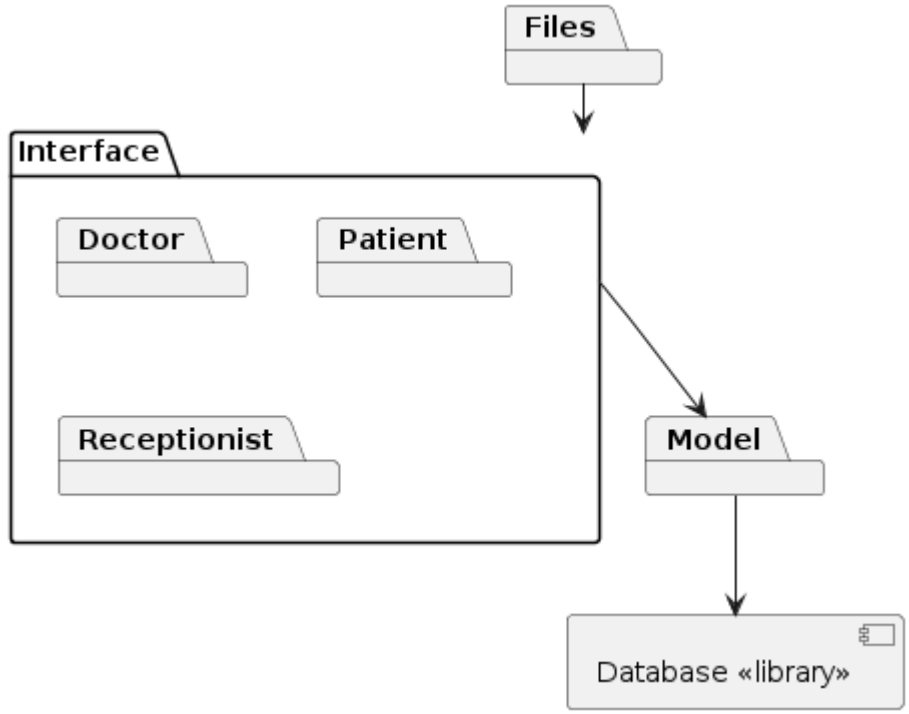
Alternate Component Diagrams using lollipops and sockets:





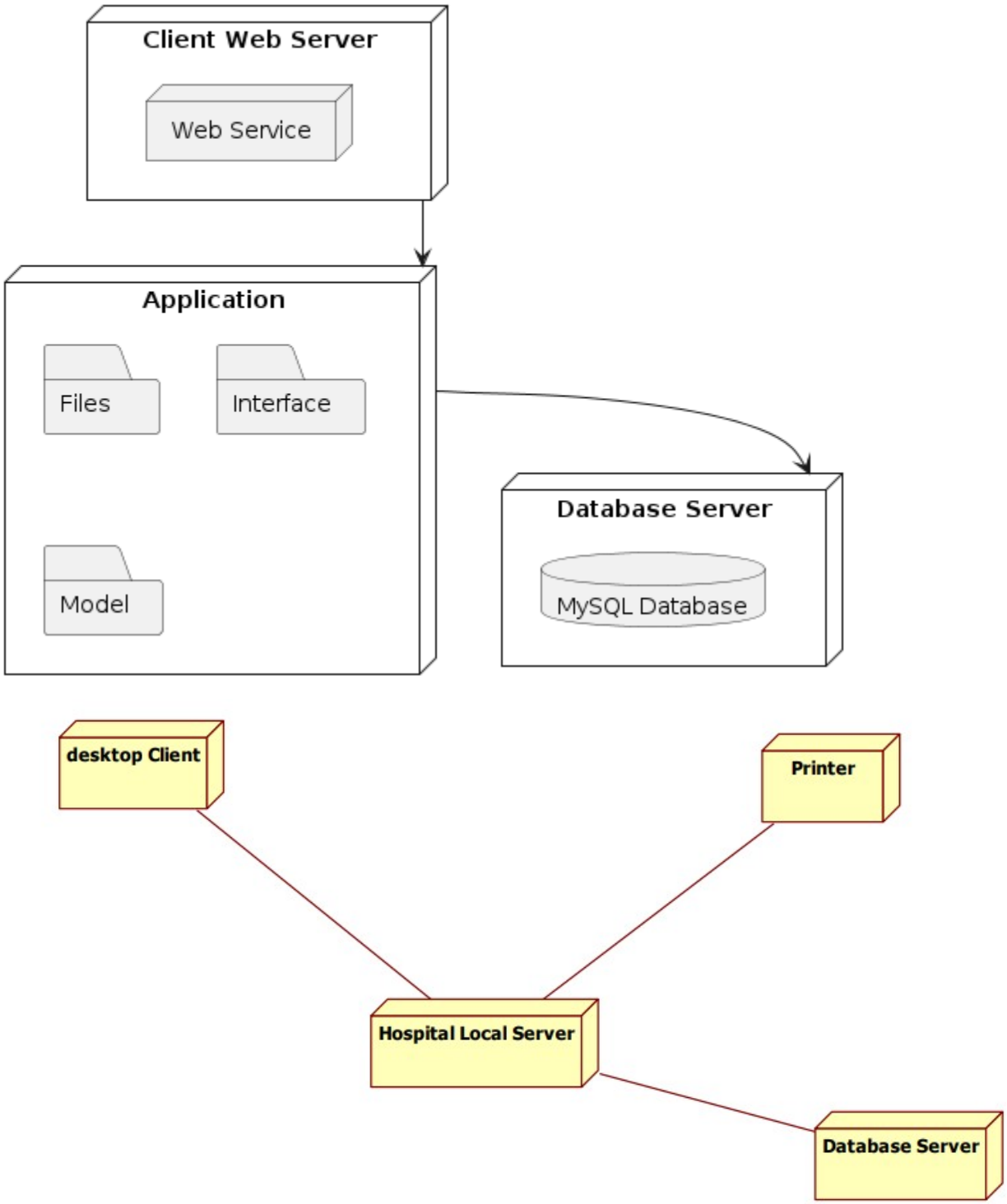

Receptionist registers self or through another receptionist
Registered
Idle
Updated
Update Information
Exit/Validate
Entry/Enter details
Entry/Update details
Do/Contact receptionist
Do/Delete receptionist
Do/Update details
Deleted
Delete Receptionist
Exit/Receptionist removed
Receptionist Deleted
Contacted
Send Message
Exit/Message sent
Entry/Input message text
Validation
lollipop Back
Back
lollipop Delete Receptionist
lollipop Send Message
socket Sent Successfully
Sent Successfully
socket Receptionist Deleted


Patient registers self or through receptionist
Registered
Idle
Updated
Update Information
Exit/Validate
Entry/Enter details
Entry/Update details
Do/Contact patient
Do/Assign doctor
Do/Delete patient
Do/Update details
Deleted
Delete Patient
Exit/Patient removed
Patient Deleted
Doctor assigned
Assign Doctor
Exit/Doctor details shown
Entry/View doctor details
Contacted
Send Message
Exit/Message sent
Entry/Input message text
Validation
lollipop Back
Back
lollipop Delete Patient
lollipop Assign Doctor
socket Back
Back
socket Sent Successfully
lollipop Send Message
Sent Successfully
socket Patient Deleted



Package Diagram

4.5.4 Deployment Diagrams



5. Implementation

The Hospital Management System (HMS) is a dynamic web application developed using a combination of client-side and server-side scripting technologies. The communication between the client and server occurs over the HTTP protocol.

For client-side scripting, we have utilized the following technologies:

- HTML5 (HyperText Markup Language): For structuring the web pages.
- CSS3 (Cascading Style Sheets): For styling and layout.
- JavaScript: For interactive elements and functionality.
- Ajax (Asynchronous JavaScript and XML): For asynchronous web page updates without reloading the entire page.
- jQuery: A JavaScript library to simplify DOM manipulation and event handling.
- Bootstrap: For responsive and modern front-end design.

For server-side scripting, we have employed PHP, enabling users to interact with the software and each other. The application is built using an Object-Oriented Programming (OOP) approach, with classes designed for each system user, such as Doctor, Receptionist, and Patient. Additionally, a CRUD class handles all operations to create, read, update, and delete patients, receptionists, doctors, medical records, medical visits, and feedback. A validation class ensures proper input validation for text, passwords, emails, and more.

APPENDIX

The appendixes are not to be considered part of the requirements.

1. Definitions, Acronyms, and Abbreviations

CoD – Component Diagram

DFD_## - Data Flow Diagram followed by a number

ERD – Entity Relation Diagram

FR_## - Functional Requirement followed by a number

HMS - Hospital Management System

PDF – Portable Document Format

SD_## - State Diagram followed by a number

UC_## - Use Case followed by a number

US_## - User Scenario followed by a number

XML – Extensible Markup Language

2. References

Constitution of the Republic of Albania
Law No.10107/2009 “On health care in the Republic of Albania”
Law no.93 / 2014 "*On the Inclusion and Accessibility of Persons with Disabilities*"
Law no. 8098/1996 "On the Status of the blind persons"
Law no. 8626/2000 "On the status of paraplegic and tetraplegic invalids”
Law no. 7889/1994 "On the Status of Invalids"
Law no. 121/2016 "On social services in the Republic of Albania"
https://www.afapdp.org/wp-content/uploads/2018/05/Albanie-Loi-n%C2%B0-9887-sur-la-protection-des-donnees-personnelles-2008.pdf
https://shendetesia.gov.al/wp-content/uploads/2022/03/Plani-Kombetar-2021-2025_ENG.pdf
https://health.ec.europa.eu/system/files/2018-04/ev_20161206_co12_en_0.pdf
https://egov.traceinternational.org/country_categorization/79/44
icd10.shendetesia.gov.al