



AUSTIN HOSPITAL MANAGEMENT SYSTEM

Presented by:

Bianca Carvalho - K221552
Lunshwa Shakya - K231451
Santiago Ortiz - K200370
Sulav Homagai - K220028
Willow Sienna - K191297



INTRODUCTION

Project Aim: Develop and Implement a new mobile application for AHMS stakeholders

- Improve efficiency scheduling and notifications.
- Enable user-friendly interactions
- Implement AES encryption to protect data/ prevent fraud.



CURRENT SYSTEM

- Ineffective and lacks efficiency for the stakeholders

Current process:

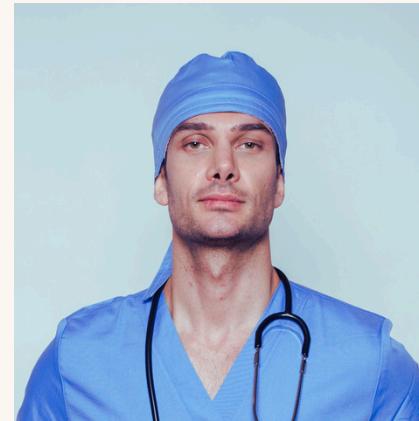
- patients is to call the clinic or attend in person to book an appointment
- does not allow for updating, notification and scheduling processes
- causes confusion, missed appointments, miscommunication and frustrations



AHMS PROJECT STAKEHOLDERS



Administration Staff



Doctor & Health-care Practitioners



IT & Support Team



Patients



Hospital



Investors & Executive Board

SCOPE AND DELIVERABLES

01.

User Registration:

- AHMS requires patients to be able to register and login using a secure and unique UserID and password.

02.

Doctor Filter:

- AHMS will need the functionality for patients to filter through doctors to choose, the filters the system requires is doctors gender, speciality, profiles etc.

03.

Patient History:

- Doctors will be able to search their patient records and history at the clinic using the patient UserID.

04.

Encryption & Security:

- The system requires AES Security Encryption to secure data using a public key cryptosystem.

05.

Notification System:

- Doctors and patients need the function to receive alerts and notifications on their respective devices to ensure they have automated reminders and communication.

06.

Appointment Management:

- Doctors require the ability to accept, reject or modify an appointment request, and patients require a confirmation message or notification with real-time updates on modifications, cancellations etc.

RESEARCH AND DEVELOPMENT

The methodology to ensure we build the correct system

- Requirements Analysis process
- System Design Analysis
- Implementation Methods Discussed
- Testing System Components
- Deployment of the new System
- Maintenance and Auditing
- Feasibility Studies



SYSTEM ANALYSIS & REQUIREMENTS: SOFTWARE & HARDWARE

HARDWARE

- **Processor:** A multi-core processor like intel core i5 for windows user and M1 chip for mac-OS user.
- **RAM:** A minimum of 4 GB RAM is recommended.
- **External storage:** An external hard drive can be used as a backup storage.
- **Other peripherals:** Keyboard, mouse, printer, and scanner.

SOFTWARE

- **Operating system:** Windows 10 or later for windows users and mac-OS Big Sur 11.0 or later is recommended.
- **XAMPP web server:** It is a free open source web server platform.
- **A compatible web browser:** Web browsers like chrome, safari can be used.

User Requirements

The use cases of the AHMS reflect the users or stakeholders that are involved in the hospital and their relative action.

The users such as admin, doctor, patient, clinic, and cashier are called actors later on in diagrams.



FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

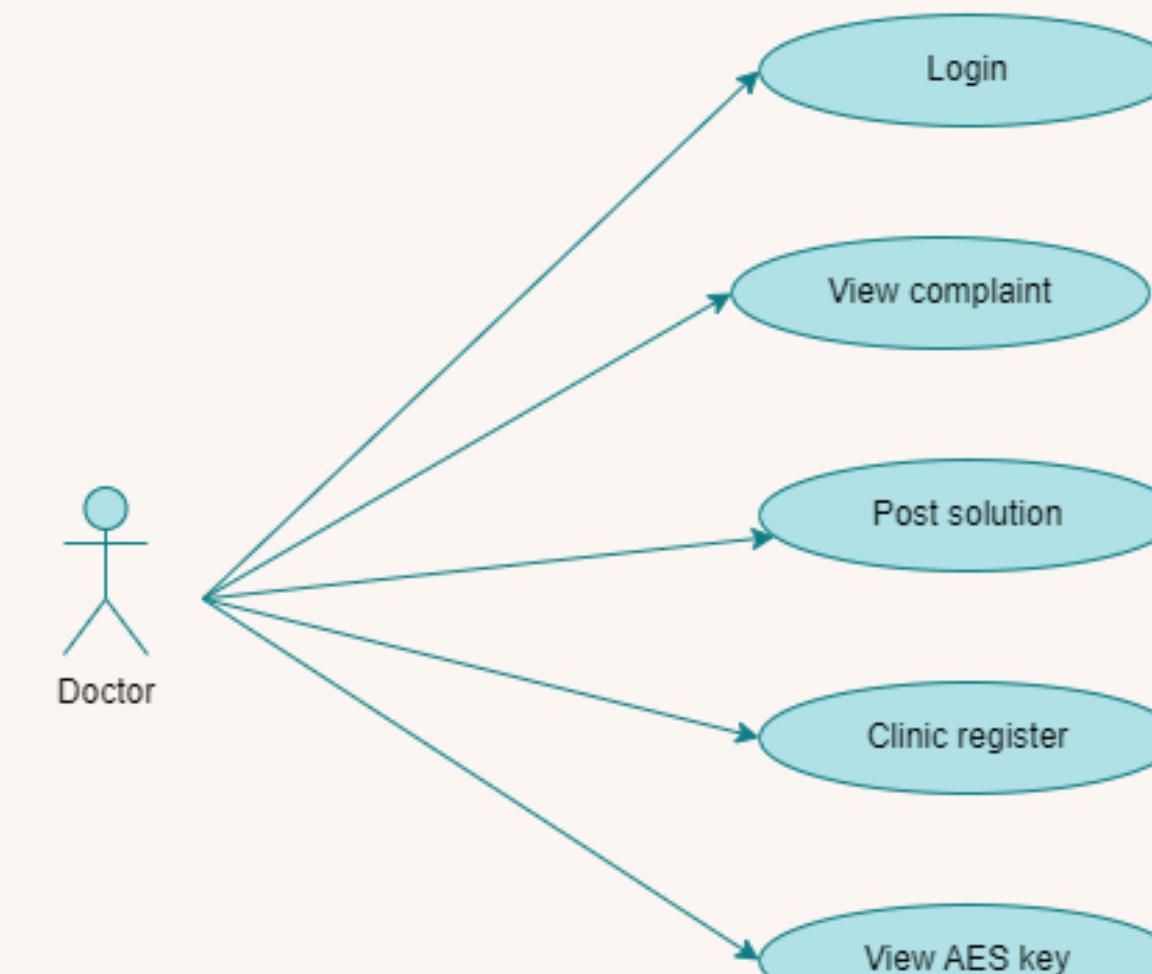
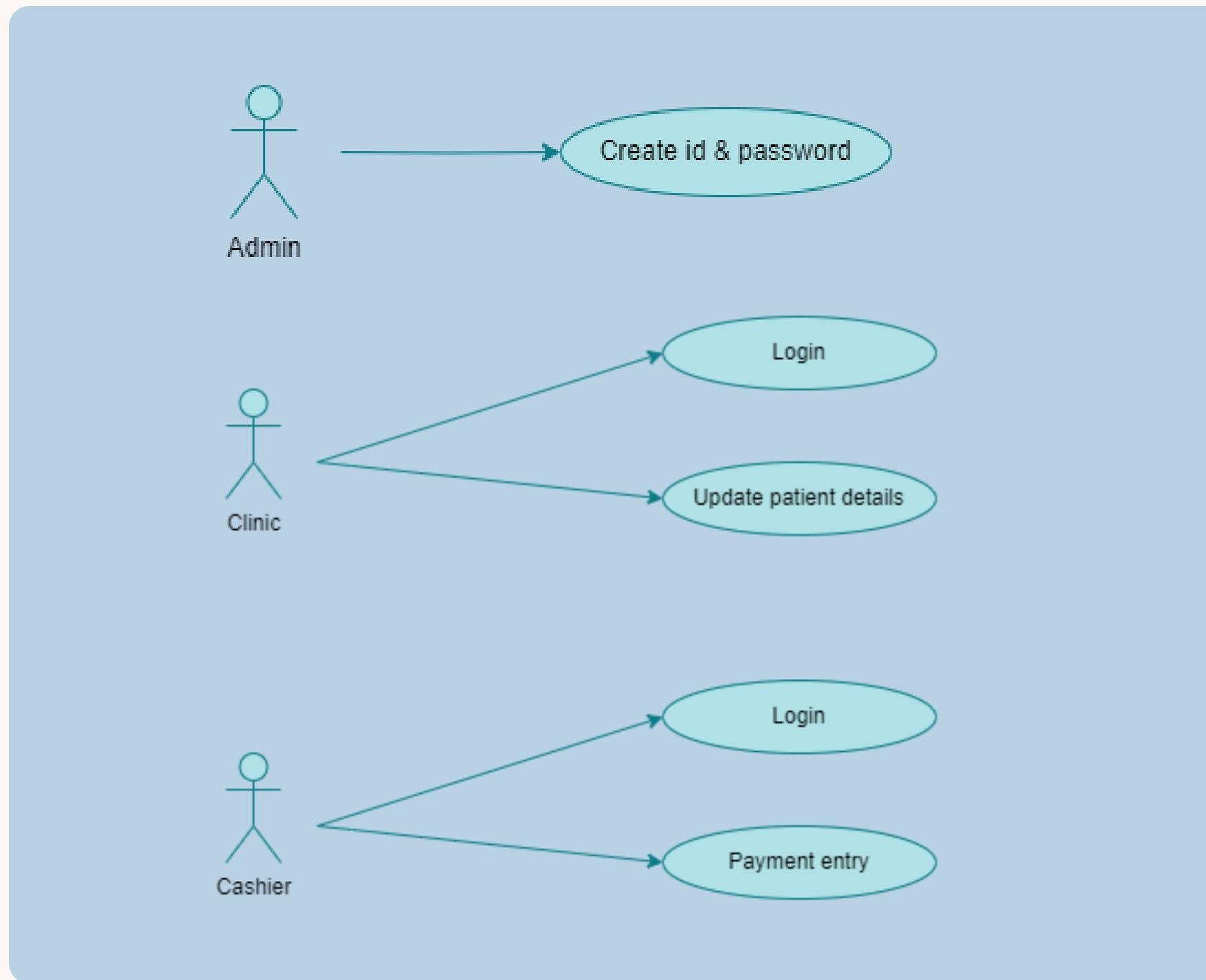
Functional Requirements:

- Admin can create the users
- Users can login
- Users can update their profile
- Doctor can view the complaints
- Doctors can register details
- Patients can post the AES key and the doctor can view them
- Patients can filter through doctors
- Cashier record payments

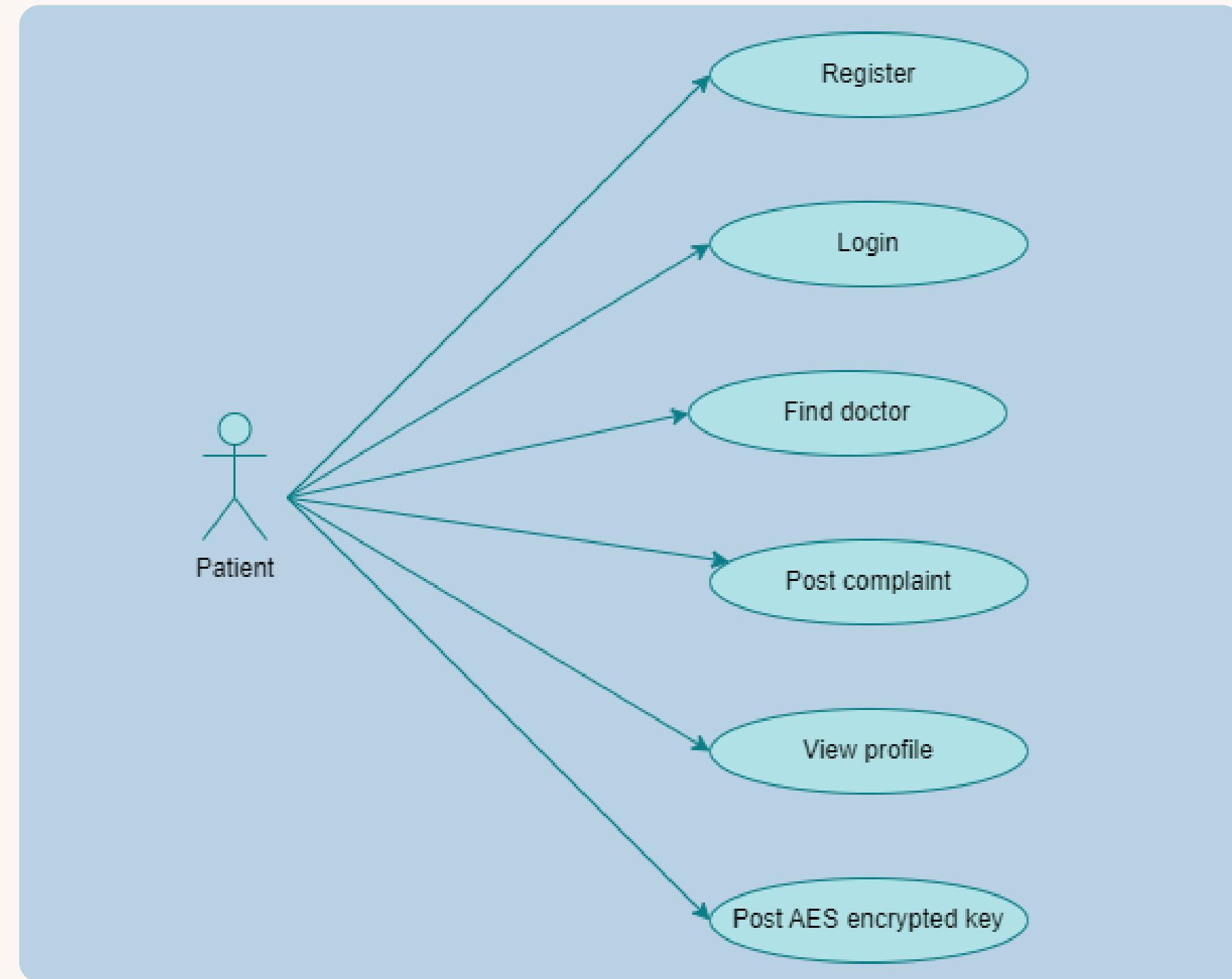
Non-Functional Requirements:

- Responsiveness
- Privacy
- Security
- Scalability
- Reliability
- Quality
- Response Time
- Performance
- Re-usability

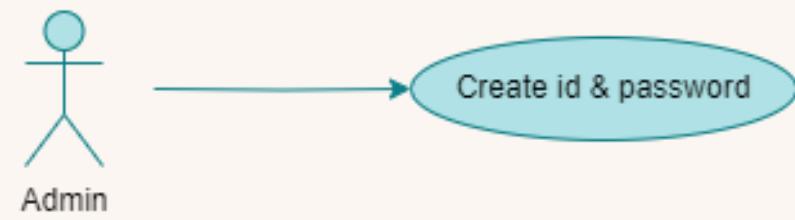
USE CASES - AHMS USERS



USE CASES - PATIENT

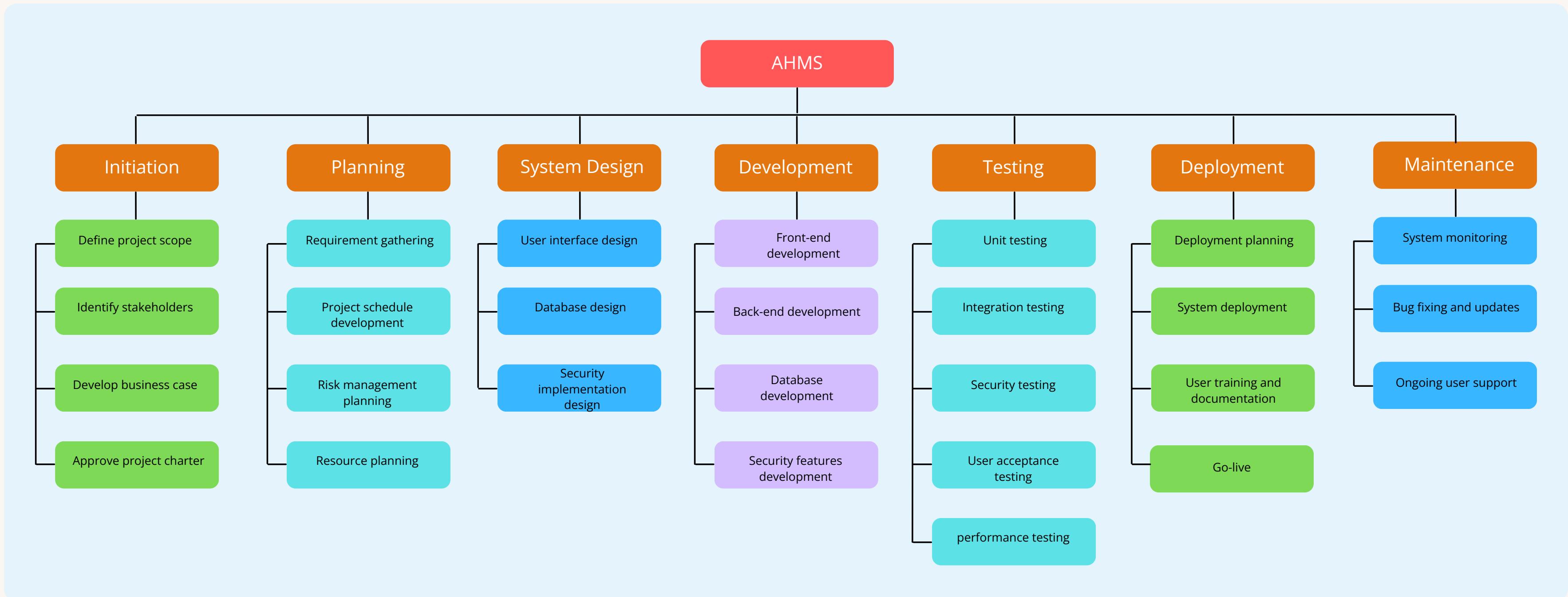


USER CASE STUDIES: ADMN



CREATE ID & PASSWORD USE CASE	
NAME:	CREATE ID & PASSWORD
ACTOR/ROLE:	ADMIN
DESCRIPTION:	DESCRIBES THE PROCESS TO CREATE USERS AS DOCTOR, CLINIC STAFF & CASHIERS.
SUCCESSFUL COMPLETION:	<ol style="list-style-type: none">1. ADMIN LOGS INTO THE SYSTEM.2. GOES TO USERS.3. CREATE A NEW USER BUTTON.4. ADD NEW USER DETAILS.5. CLICK CREATE.
ALTERNATIVE:	
PRECONDITION:	ADMIN SHOULD HAVE THE DETAILS OF THE USER
POSTCONDITION:	USERS AS DOCTOR, CLINIC STAFF, OR CASHIER ARE CREATED.
ASSUMPTIONS:	USER DOES NOT HAVE AN ACCOUNT.

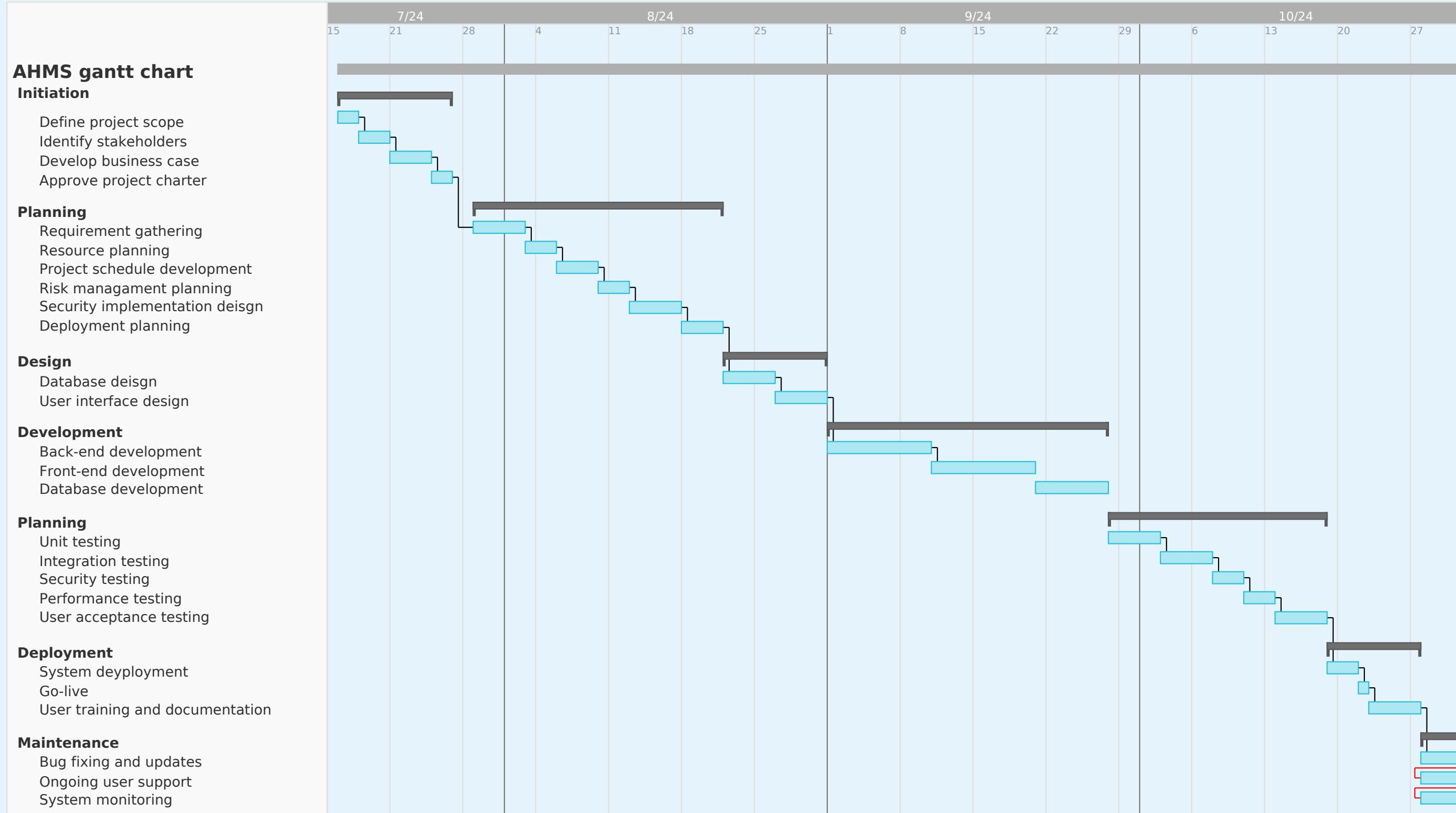
WORK BREAKDOWN STRUCTURE



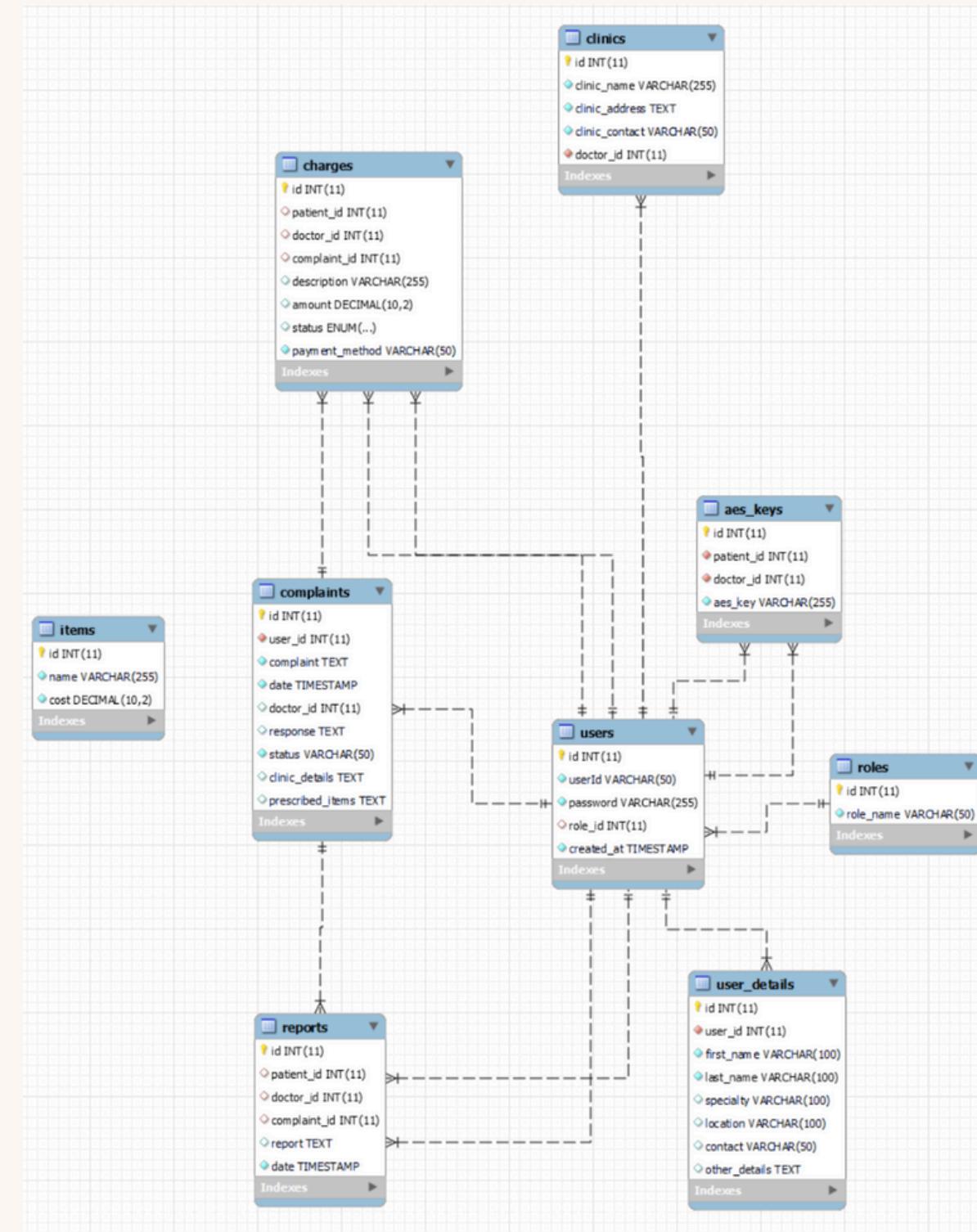
PROJECT SCHEDULE

Phase	Task	Duration (Days)	Start Date	End Date	Phase	Task	Duration (Days)	Start Date	End Date
Initiation	Define project scope	2	16/07/2024	17/07/2024	Testing	Unit testing	5	28/09/2024	02/10/2024
	Identify stakeholders	3	18/07/2024	20/07/2024		Integration testing	5	03/10/2024	07/10/2024
	Develop business case	4	21/07/2024	24/07/2024		Security testing	3	08/10/2024	10/10/2024
	Approve project charter	2	25/07/2024	26/07/2024		Performance testing	3	11/10/2024	13/10/2024
Planning	Requirement gathering	5	29/07/2024	02/08/2024		User acceptance testing	5	14/10/2024	18/10/2024
	Resource planning	3	03/08/2024	05/08/2024		System deployment	3	19/10/2024	21/10/2024
	Project schedule development	4	06/08/2024	09/08/2024		Go-live	1	22/10/2024	22/10/2024
	Risk management planning	3	10/08/2024	12/08/2024		User training and documentation	5	23/10/2024	27/10/2024
Design	Security implementation design	5	13/08/2024	17/08/2024	Maintenance	Bug fixing and updates	Ongoing	28/10/2024	Ongoing
	Deployment planning	4	18/08/2024	21/08/2024		Ongoing user support	Ongoing	28/10/2024	Ongoing
	Database design	5	22/08/2024	26/08/2024		System monitoring	Ongoing	28/10/2024	Ongoing
	User interface design	5	27/08/2024	31/08/2024					
Development	Back-end development	10	01/09/2024	10/09/2024					
	Front-end development	10	11/09/2024	20/09/2024					
	Database development	7	21/09/2024	27/09/2024					

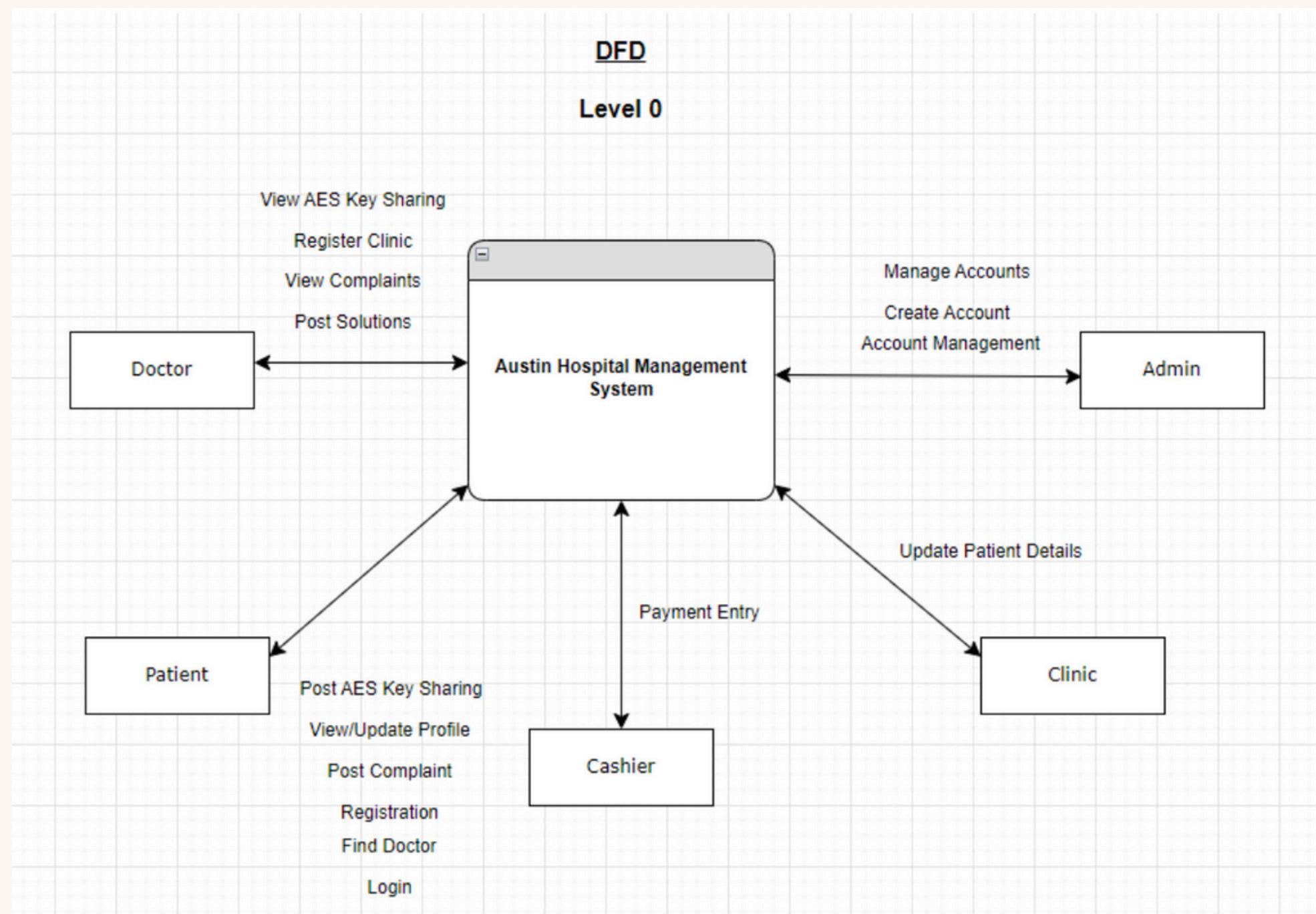
GANTT CHART



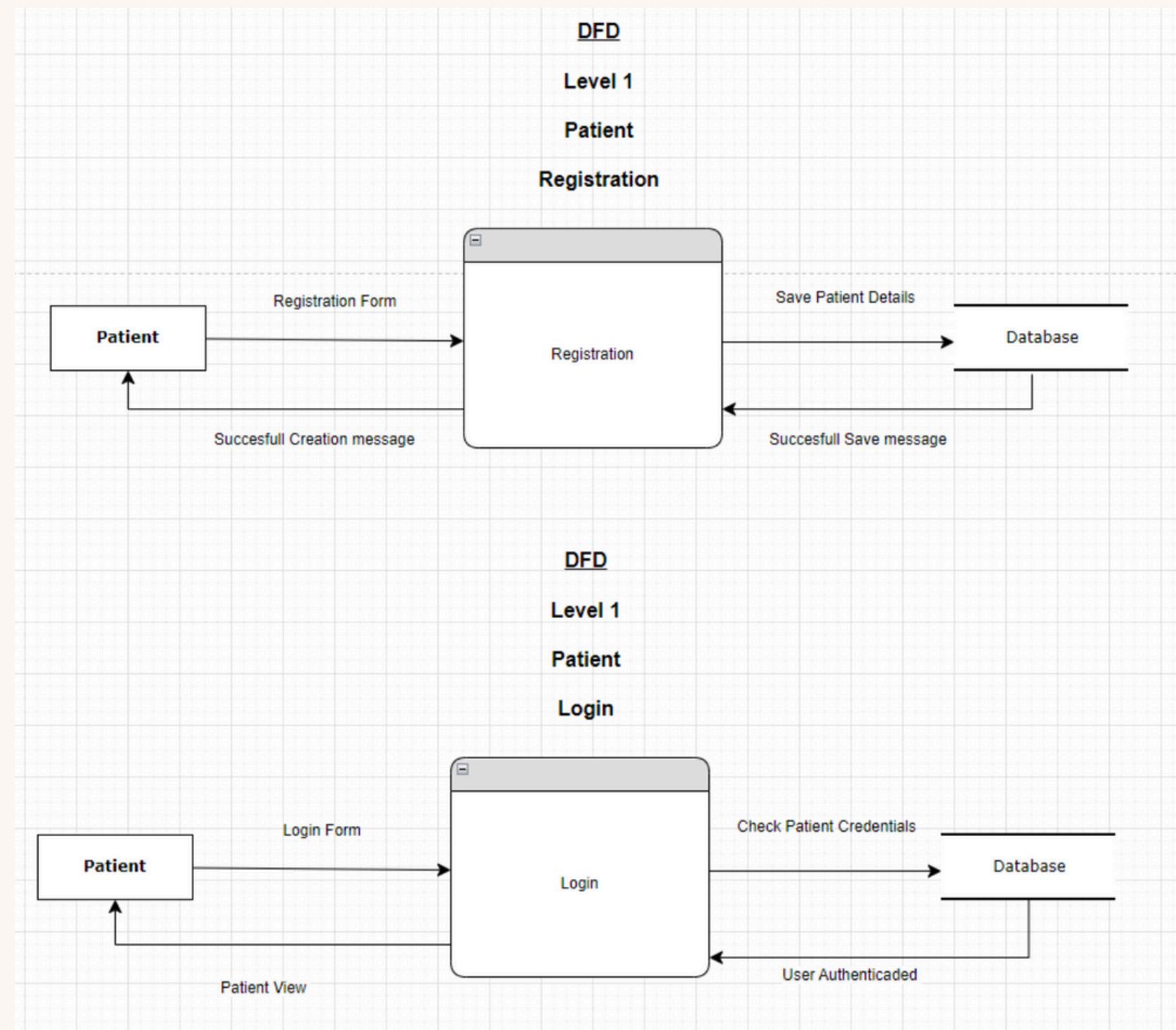
ENTITY RELATIONSHIP DIAGRAM (ERD)



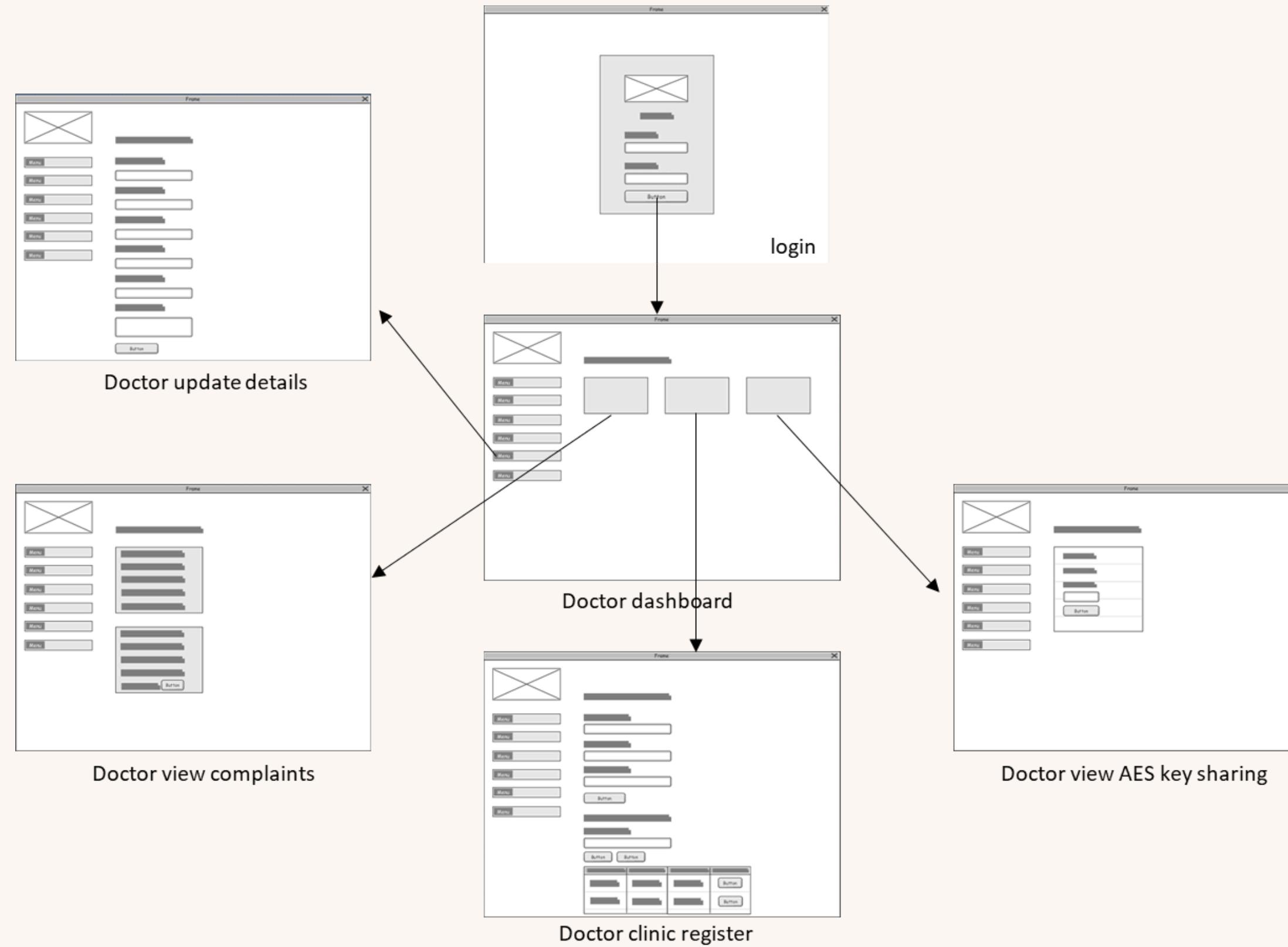
FLOWDIAGRAMS(DFD)



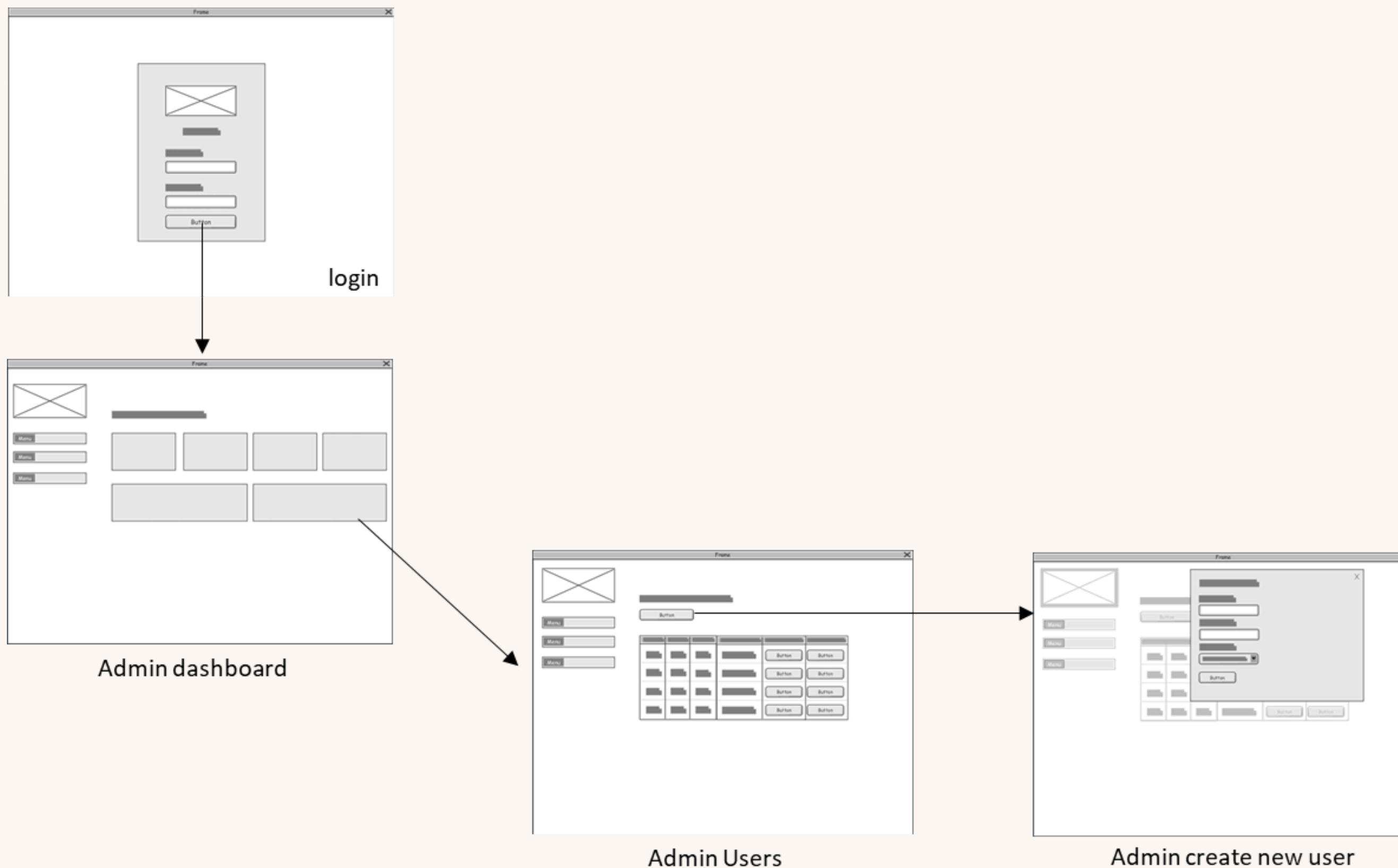
FLOWDIAGRAMS(DFD)



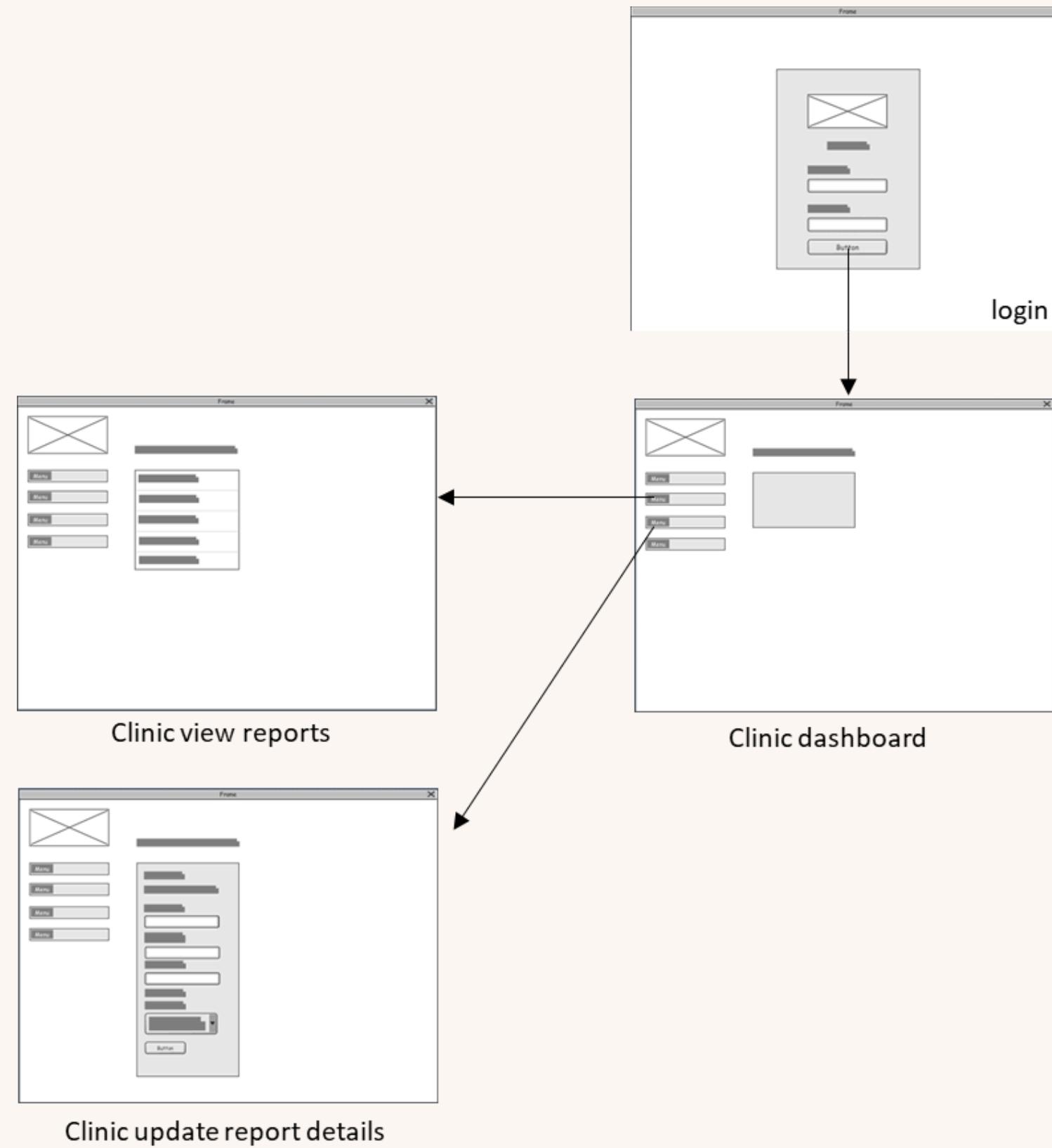
USER INTERFACE DESIGN



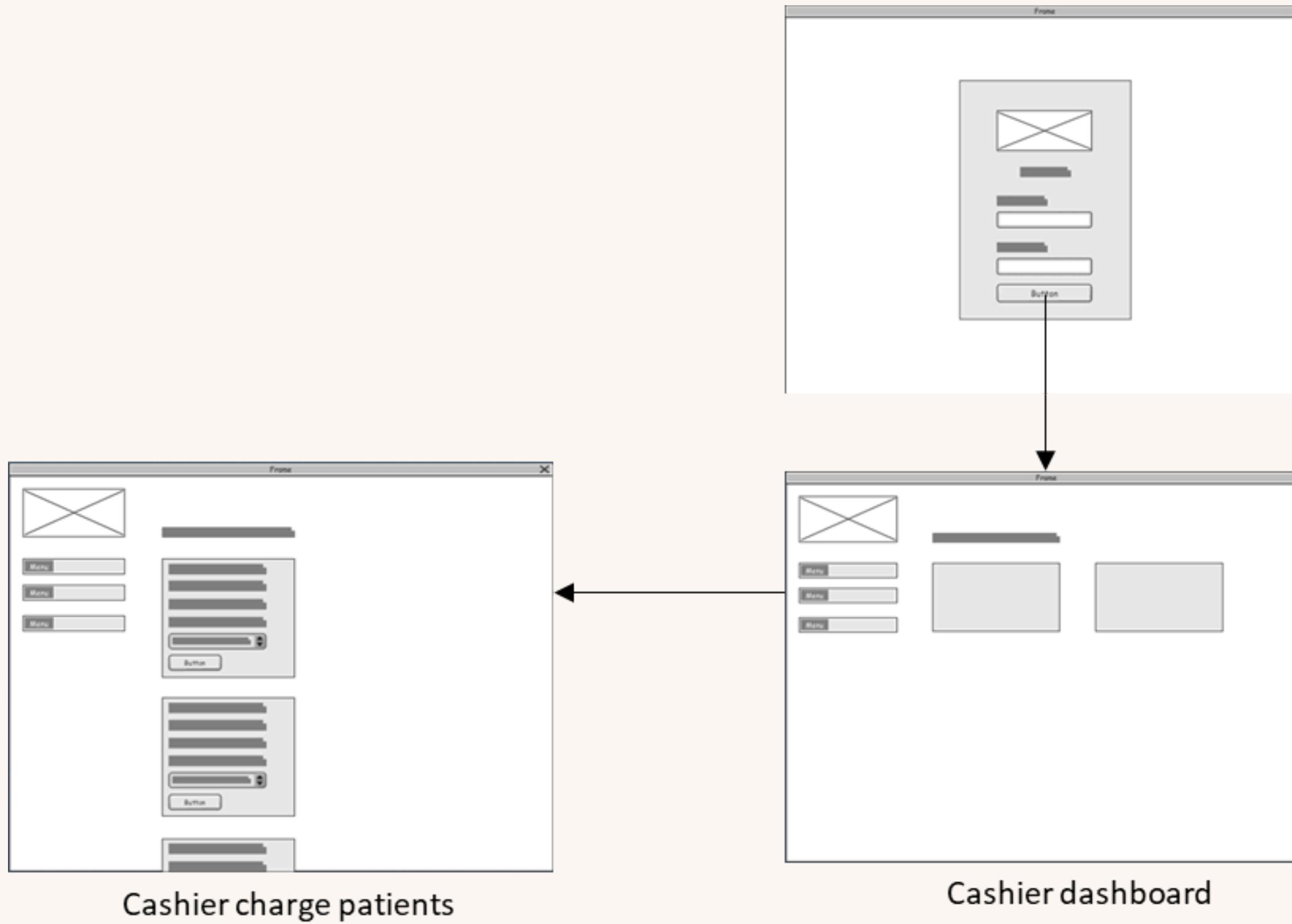
USER INTERFACE DESIGN



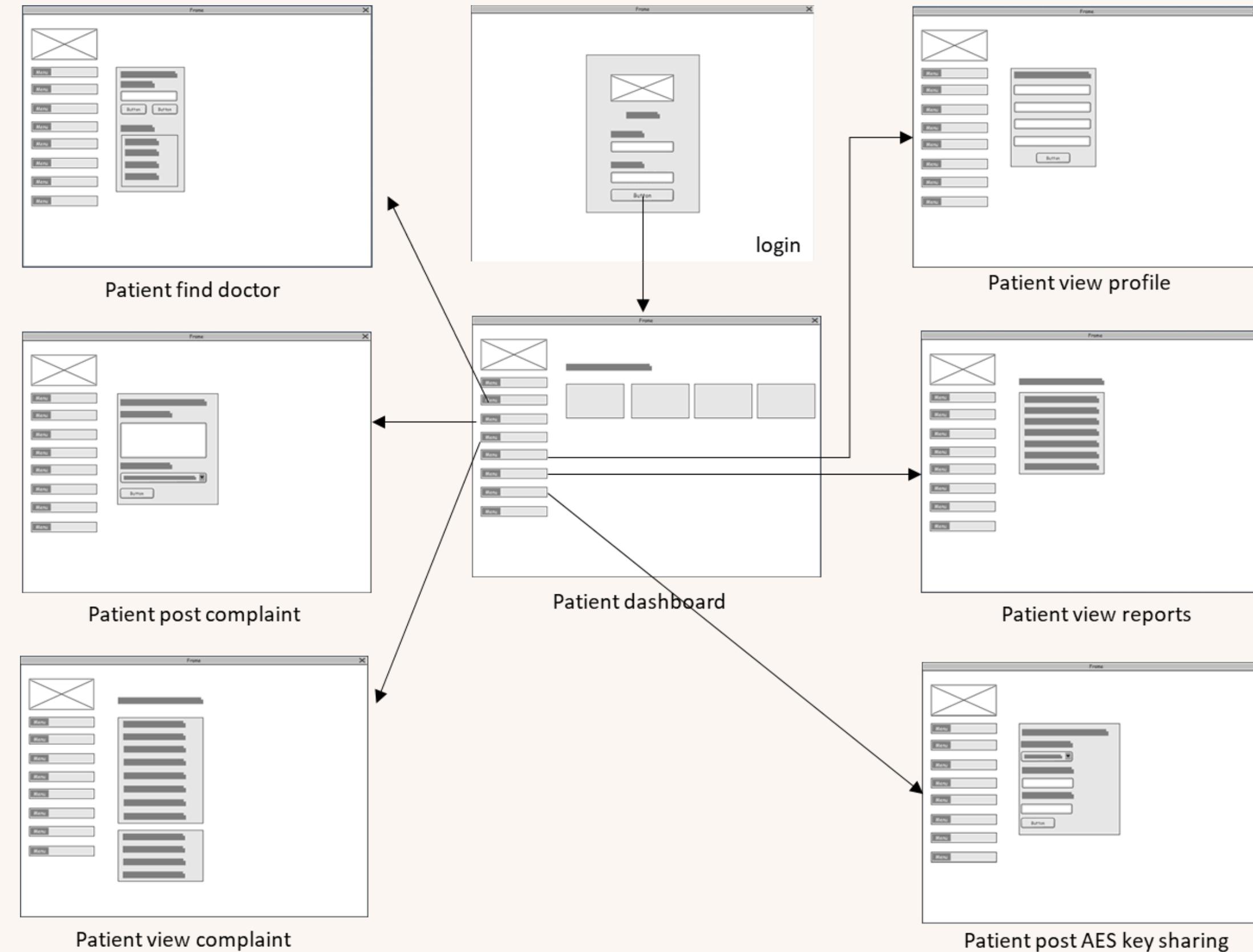
USER INTERFACE DESIGN



USER INTERFACE DESIGN



USER INTERFACE DESIGN





IMPLEMENTATION OF DATA ENCRYPTION AND ANONYMIZATION

AES Encryption:

- Encrypted patient data with AES using OpenSSL.
- Secure key management to restrict access.

RBAC Implementation:

- Restricted data access based on user roles.

Data Anonymization:

- Pseudonymization and data masking to protect patient identities

RISKS IN A DBMS

- Data Security
- Downtime of a System
- Data Integrity
- Compliance & Regulations
- Scalability Limitations
- Integration Constraints



ETHICAL RISKS IN A DBMS

- Privacy and Security
- Data Misuse
- Discrimination & Bias
- Knowledgeable Consent
- Ownership & Control of Data
- Data Security in Third-Party Services

TESTING PLAN OVERVIEW

A COMPREHENSIVE TESTING PLAN
ENSURES THAT ALL COMPONENTS OF
THE SYSTEM
ARE VERIFIED FOR FUNCTIONALITY,
PERFORMANCE, SECURITY, AND
RELIABILITY.



THE
PLAN INCLUDES VARIOUS TESTING PHASES
AND COVERS BOTH FUNCTIONAL AND NON-
FUNCTIONAL REQUIREMENTS.

TEST CASES



TEST CASES ARE DEVELOPED FOR EACH FEATURE AND FUNCTIONALITY OF THE SYSTEM. THEY INCLUDE:

- **UNIT TESTING: VERIFYING INDIVIDUAL COMPONENTS.**
- **INTEGRATION TESTING: ENSURING MODULES WORK TOGETHER.**
- **SYSTEM TESTING: END-TO-END TESTING OF THE COMPLETE SYSTEM.**
- **ACCEPTANCE TESTING: VALIDATING THE SYSTEM MEETS BUSINESS REQUIREMENTS.**



TEST PLAN EXECUTION

THE TEST PLAN EXECUTION INVOLVES RUNNING THE TEST CASES AS PER THE SCHEDULE. IT INCLUDES:

- **TEST ENVIRONMENT SETUP: PREPARING THE TESTING ENVIRONMENT.**
- **TEST DATA PREPARATION: CREATING DATA SETS FOR TESTING.**
- **TEST EXECUTION: RUNNING THE TESTS AND LOGGING DEFECTS.**
- **TEST REPORTING: DOCUMENTING TEST RESULTS AND ISSUES.**

**CONCLUSION: YOUR HEALTH SYSTEM,
ALWAYS A PRIORITY**



Austin Hospital

**THANK YOU
VERY
MUCH!**

