

### 3.4 Project Design

#### Flowchart

Flowchart of the system

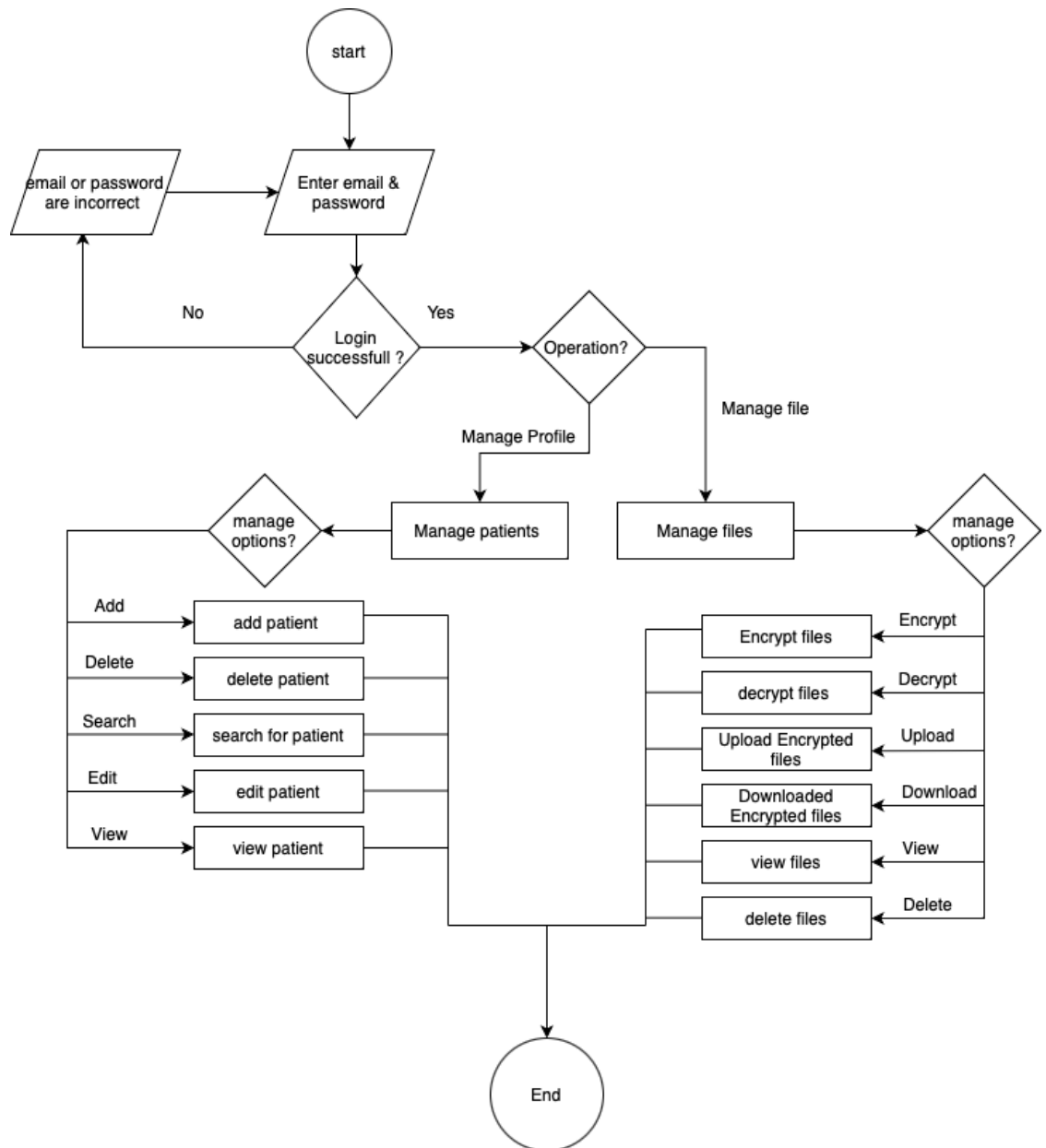


Figure 9 flowchart of the system

The system initiates with the login process, where the admin/doctor inputs their email and password. If the login attempt fails, the system prompts the admin/doctor to retry. Upon successful login, the admin/doctor gains access to two main functionalities. The first is the management of patients, offering options to add, delete, view, search, and edit patient information. The second involves the management of files, providing functionalities such as encrypting files, decrypting files, uploading encrypted files, downloading encrypted files, viewing files, and deleting files. After completing the desired tasks, the system concludes.

## Context Diagram

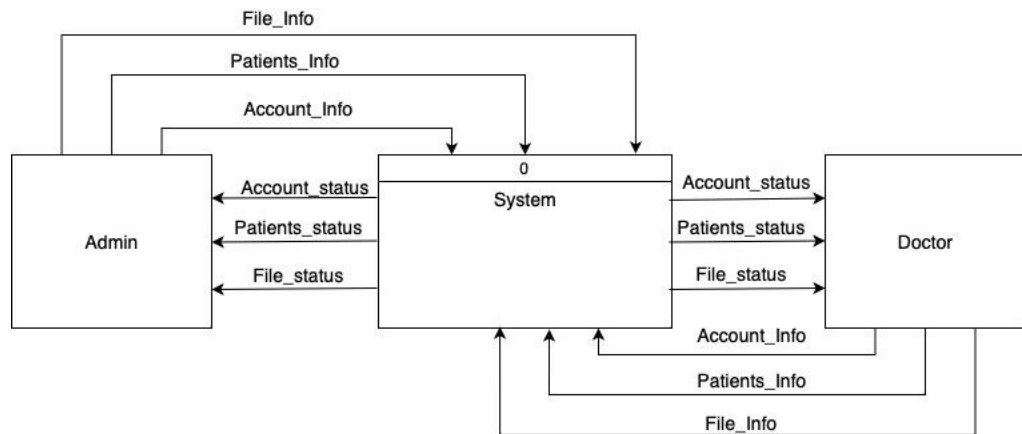


Figure 15 Context Diagram

In the context diagram, the administrator and doctor have distinct interactions with the system, both beginning with login procedures. The administrator sends login details and receives login status, with the ability to update patient and file information, to which the system responds with the corresponding statuses. Similarly, the doctor also logs in, with the system responding to their updates on patient information and files, reflecting a controlled access and data management process within the healthcare system.

## Use case diagram

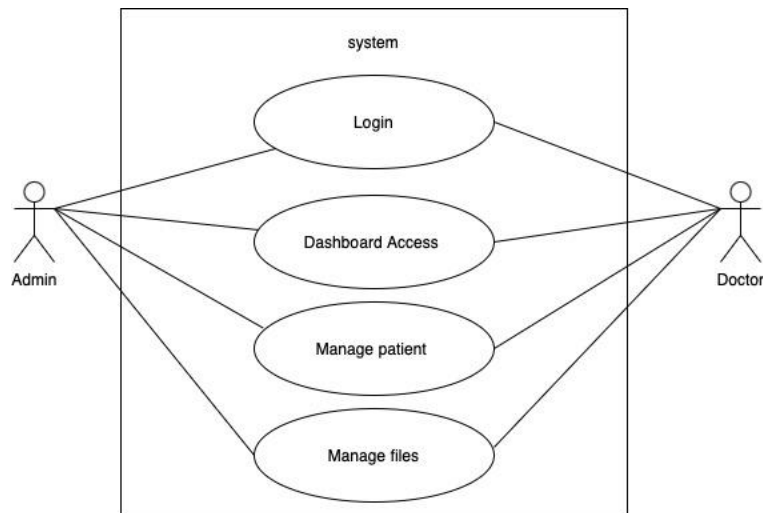


Figure 16 Use Case Diagram

In the system's use case, both the admin and doctor actors engage in parallel activities. Each begins by logging into the system, then gains dashboard access to navigate their respective functionalities. The admin has privileges to manage patient information and files, encompassing adding, editing, and deleting data. Similarly, the doctor can also manage patient details and interact with the file system, albeit possibly with different access levels or functionalities based on their role. The system tracks these activities, updating statuses accordingly to ensure secure data handling.

## Use Case Description

### Login

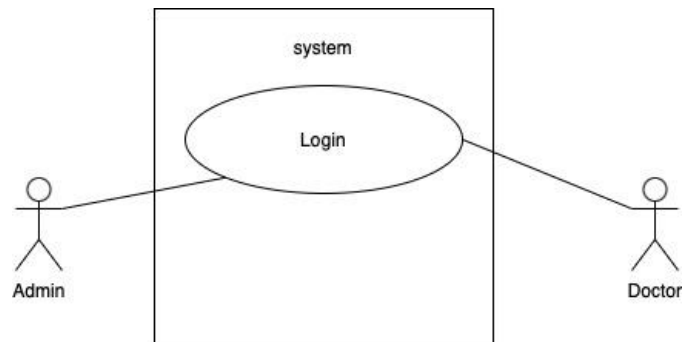


Figure 17 Use Case description\_Login

Table 4 Use Case Decription\_Login

Use Case ID	PKU-UC-01
Brief Description	To access the system and perform administrative tasks.
Actor	Admin, doctor
Pre-Condition	<ol style="list-style-type: none"><li>1. The system is operational.</li><li>2. The admin/doctor account exists in the system.</li></ol>
Basic Flow	<ol style="list-style-type: none"><li>1. Admin/Doctor enters valid credentials (username and password) on the login page.</li><li>2. The system verifies the credentials.</li><li>3. If the credentials are valid, the admin is successfully logged into the system.</li></ol>
Alternative Flow	If the credentials are invalid, the system displays an error message, and the admin/doctor is prompted to re-enter the correct credentials.
Exception Flow	If the provided credentials are invalid, the system displays an error message and prompts the admin/doctor to enter valid credentials.
Post-Condition	The admin/doctor gains access to the system and can proceed to the dashboard.

## Dashboard access

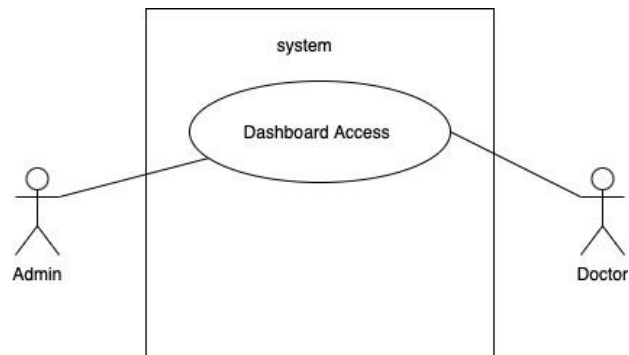


Figure 18 Use Case description\_Dashboard access

Table 5 Use Case Decription\_Dashboard access

Use Case ID	PKU-UC-02
Brief Description	To view the system dashboard for an overview of system status and activities.
Actor	Admin, doctor
Pre-Condition	The admin/doctor is logged into the system.
Basic Flow	After successful login, the admin/doctor is directed to the system dashboard.
Alternative Flow	
Exception Flow	
Post-Condition	The admin/doctor has access to real-time information and insights provided by the dashboard.

## Manage patient

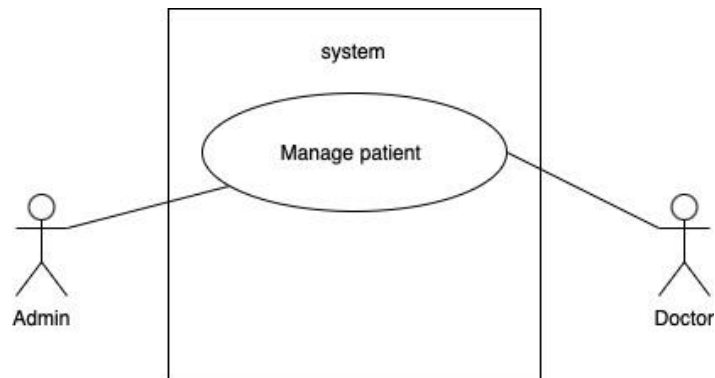


Figure 19 Use Case description\_Manage patient

Table 6 Use Case Description\_Manage patient

Use Case ID	PKU-UC-03
Brief Description	To manage patient information within the system.
Actor	Admin, doctor
Pre-Condition	<ol style="list-style-type: none"> <li>1. The admin/doctor is logged into the system.</li> <li>2. The admin/doctor has access to the dashboard.</li> </ol>
Basic Flow	<ol style="list-style-type: none"> <li>1. From the dashboard, the admin/dpctor navigates to the "Manage Patient" section.</li> <li>2. The system presents options for adding, updating, or deleting patient information.</li> <li>3. The admin/doctor performs the desired action, such as adding a new patient or updating existing patient details.</li> </ol>
Alternative Flow	The admin/doctor can search for specific patient records.
Exception Flow	If there is an issue with patient management (e.g., adding a duplicate patient), the system provides an error message.
Post-Condition	Patient information is updated or modified as per the admin's actions.

## Manage files

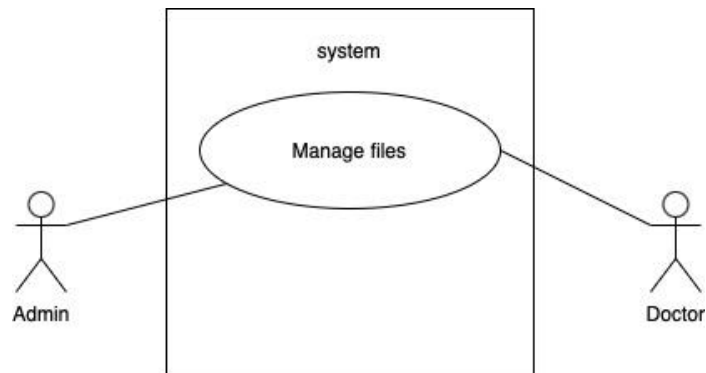


Figure 20 Use Case description\_Manage files

Table 7 Use Case Description\_Manage files

Use Case ID	PKU-UC-04
Brief Description	To manage files within the system, including encryption and decryption tasks.
Actor	Admin, doctor
Pre-Condition	1- The admin/doctor is logged into the system. 2- The admin/doctor has access to the dashboard.
Basic Flow	1. From the dashboard, the admin navigates to the "Manage Files" section. 2. The system provides options for uploading, encrypting, decrypting, downloading, view, and deleting files. 3. The admin/doctor selects the desired action, such as uploading a new file or decrypting an existing file.
Alternative Flow	The admin can configure encryption parameters before uploading files.
Exception Flow	If there is an issue with file management (e.g., file upload failure or download error), the system provides an error message.
Post-Condition	File management actions are executed as per the admin's selections.



## Activity Diagram

### Login

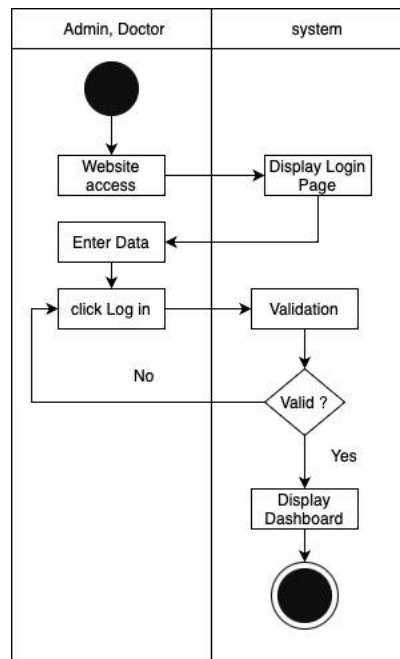


Figure 21 Activity Diagram\_Login

In the login activity, the admin/doctor gains access to the website, triggering the display of the login page by the system. The admin/doctor then enters their login credentials (email and password) and clicks the login button. The system validates the entered information, prompting the admin/doctor to retry if the validation fails. Upon successful validation, the system proceeds to display the dashboard, marking the completion of the login process.

## Manage patients

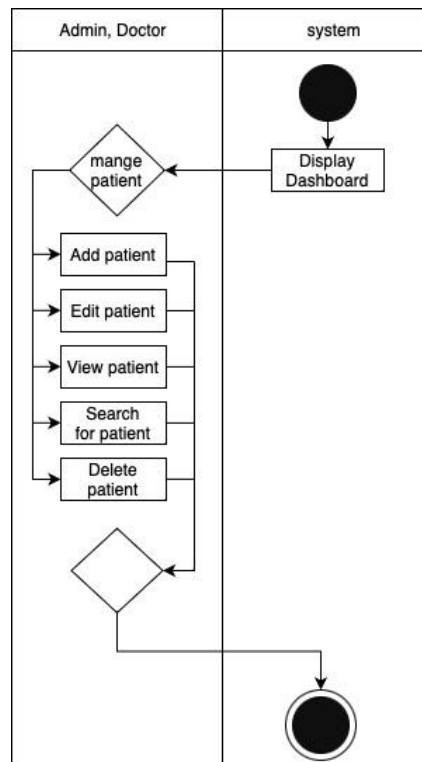


Figure 22 Activity Diagram\_Manage patient

In the "Manage Patient" activity, the system initially displays the dashboard for the admin and doctor. Admin and doctor, within the dashboard, engages in patient management tasks such as adding, editing, viewing, searching, and deleting patient records. These functionalities provide the admin with comprehensive control over patient data within the system.

## Manage files

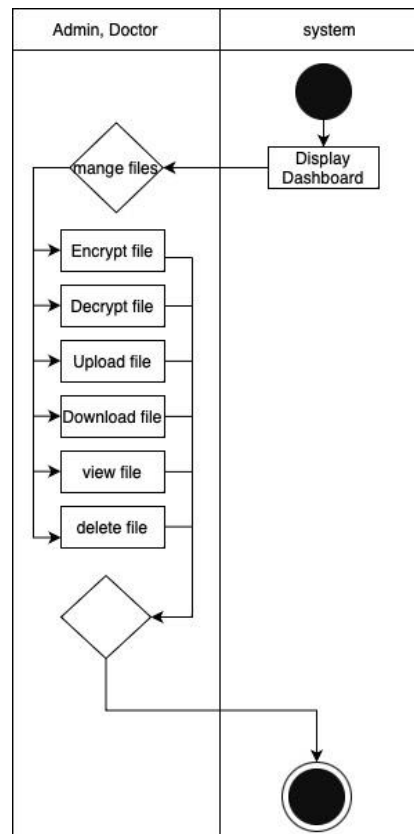


Figure 23 Activity Diagram\_Manage files

In the "Manage Files" activity, the system first presents the dashboard to the admin and doctor. Within the dashboard, the admin/doctor engages in file management tasks, encompassing actions such as encrypting files, decrypting files, uploading encrypted files, downloading encrypted files, viewing files, and deleting files. These actions give the admin flexible control over healthcare files, making sure file management in the system is secure and efficient.

### 3.5 Data Design

This subtopic is called data design. It is the process of creating a logical and physical representation of the data that will be used in a system. A key component of data design is the Entity-Relationship Diagram (ERD), which illustrates the relationships between different entities in the system. Another important component is the data dictionary, which describes the structure, organization, and attributes of the data.

#### ERD (Entity Relationship Diagram)

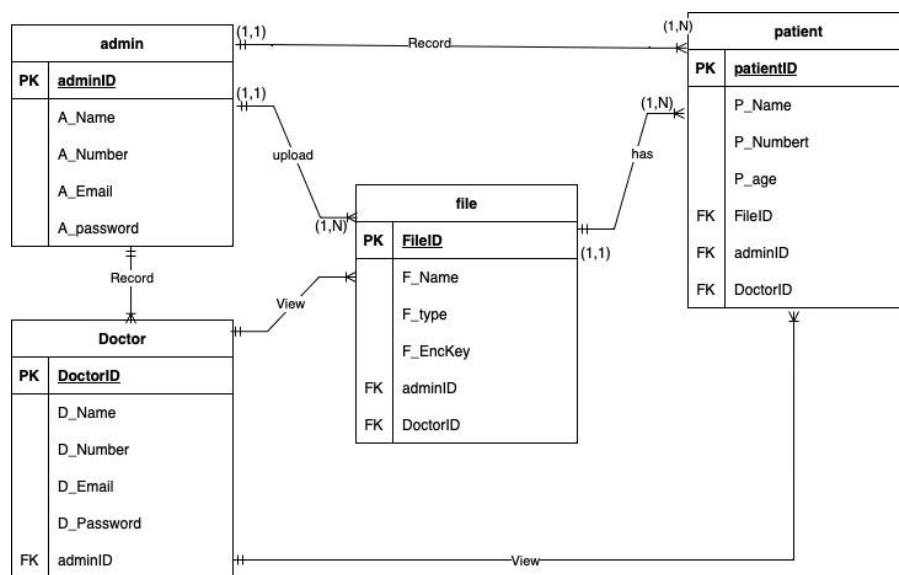


Figure 24 ERD

The Entity-Relationship Diagram (ERD) illustrates the database schema for a secure healthcare system, detailing the interactions among Admin, Doctor, Patient, and File entities. The Admin entity is empowered to upload multiple Files, each File being identifiable by unique attributes like ID, name, type, and encryption key. Patients are linked to Files in a one-to-many relationship, showing that a single patient can have multiple files. Additionally, Doctors are presented as a separate entity with their own details, who can view Files, implying a role-based access in the system. This ERD encapsulates the workflow of managing encrypted patient files, with distinct roles for Admins and Doctors, ensuring a secure and organized data management structure within the healthcare setting.

## Data Dictionary

This section presents the data dictionary, which is derived from the Entity Relationship Diagram (ERD) of the proposed system as illustrated in Figure 24.

### Data Dictionary for Admin

Table 8 Data Dictionary\_Admin

Field Name	Data Type	Description	Constraint
adminID	Int	Unique identifier for an administrator	PK
A_Name	varchar(255)	Full name of the administrator	
A_Number	varchar(15)	Contact number for the administrator	
A_Email	varchar(255)	Email address of the administrator	
A_Password	varchar(255)	password for admin login	

## Data Dictionary for Patient

Table 9 Data Dictionary\_Patient

Field Name	Data Type	Description	Constraint
patientID	Int	Unique identifier for an patient	PK
P_Name	varchar(255)	Full name of the patient	
P_Number	varchar(15)	Contact number for the patient	
P_age	Int	Age of patient	
adminID	Int	Identifier for the admin who responsible for patient	FK
FileID	Int	Unique identifier for a file	FK
doctorID	Int	Identifier for the Doctor who responsible for patient	FK

## Data Dictionary for File

Table 10 Data Dictionary\_File

Field Name	Data Type	Description	Constraint
FileID	Int	Unique identifier for a file	PK
F_Name	varchar(255)	name of the file	
F_type	varchar(50)	Type of the file	
F_EncKey	varchar(255)	Encryption key used for the file	
adminID	Int	Identifier for the admin who uploaded the file	FK
doctorID	Int	Identifier for the Doctor who responsible for patient	FK

## Data Dictionary for doctor

Table 11 Data dictionary\_Doctor

Field Name	Data Type	Description	Constraint
DoctorID	Int	Unique identifier for a doctor	PK
D_Name	varchar(255)	Full name of the doctor	
D_Number	varchar(15)	Contact number for the doctor	
D_Email	varchar(255)	Email address of the doctor	
D_Password	varchar(255)	password for doctor login	
adminID	Int	Identifier for the admin	FK



### 3.6 Proof of Initial Concept

This subsection presents the proposed system's interface mock-ups, depicting the fundamental layout, imagery, buttons, and icons. The system boasts a responsive web design, meaning that the layout adapts dynamically to fit the screen size of the device in use.

#### Login Page

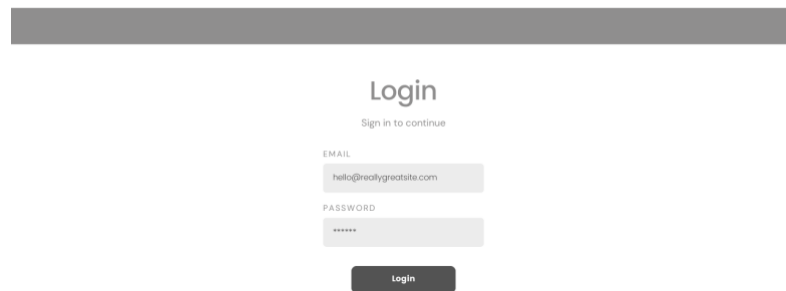


Figure 25 Home Page of the system

The mockup illustrates a streamlined login page with email and password fields, capped by a prominent "Login" button for system access.

## System Dashboard

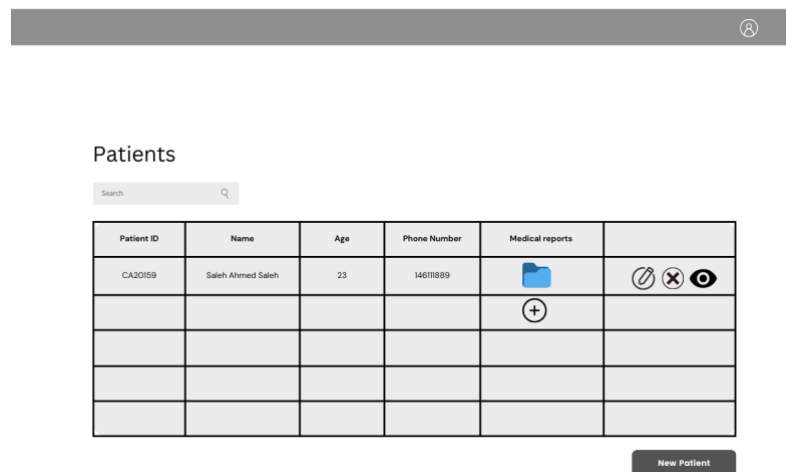


Figure 26 Dashboard of the system

The mockup displays the dashboard for patient management, featuring a table that lists patients by ID, name, age, and phone number, with an option to view medical reports. Icons for editing, deleting, and viewing details are included, and a 'New Patient' button is present to add new patient.

## Adding patient

The mockup shows a web interface for adding a new patient. At the top, there is a dark grey header bar with a small user icon on the right. Below the header, the title 'Add patient' is centered. The form consists of four input fields arranged in two columns. The left column contains 'PATIENT ID' with the value 'CE00395' and 'AGE' with the value '22'. The right column contains 'NAME' with the value 'Faisal Mohammed Abode' and 'PHONE NUMBER' with the value '0145657849'. Below the input fields is a dark grey button labeled 'ADD'.

PATIENT ID	NAME
CE00395	Faisal Mohammed Abode

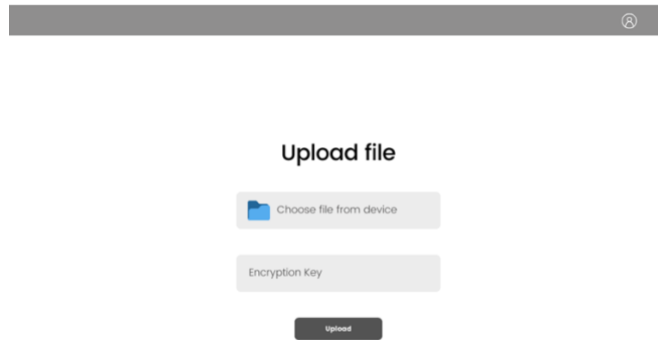
AGE	PHONE NUMBER
22	0145657849

ADD

Figure 27 adding patient

The mockup displays a form for adding a new patient to the system. It has fields for entering the patient ID, name, age, and phone number. A prominent "ADD" button is provided to submit the information and create the new patient record.

## Uploading encrypted file

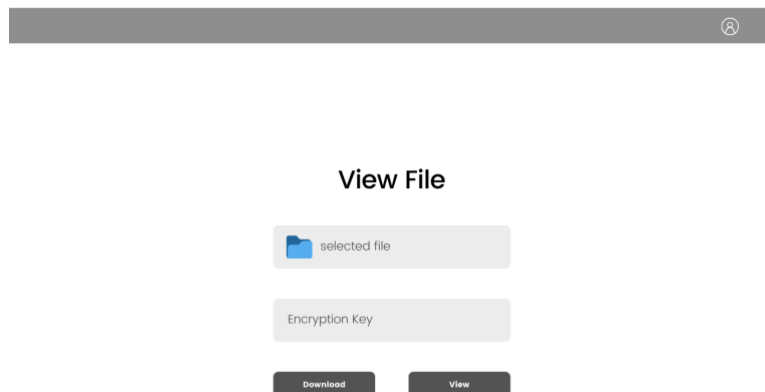


The mockup shows a file upload interface. At the top, there is a dark gray header bar with a user profile icon on the right. Below the header, the title "Upload file" is centered. Under the title, there is a button labeled "Choose file from device" with a folder icon. Below this button is a text input field labeled "Encryption Key". At the bottom, there is a dark gray button labeled "Upload".

Figure 28 uploading file

The mockup shows a file upload interface designed for administrators to securely upload and encrypt files. It provides a "Choose file from device" button to select files and a field for entering an "Encryption Key". Below these, an "Upload" button is ready to initiate the file encryption and upload process.

## Viewing file



The mockup shows a file viewing interface. At the top, there is a dark gray header bar with a user profile icon on the right. Below the header, the title "View File" is centered. Under the title, there is a button labeled "selected file" with a folder icon. Below this button is a text input field labeled "Encryption Key". At the bottom, there are two dark gray buttons: "Download" and "View".

Figure 29 viewing file

The mockup shows a page for viewing and accessing encrypted files. admin can select a file, input the corresponding encryption key, and then choose to either download or view the file.