MediManager

"Navigating Health, Guiding Wellness:
Your Trusted System."

MediManager is a comprehensive software solution designed to streamline and enhance the operational efficiency of hospital management.

Yeshwantrao Chavan College of Engineering, Nagpur

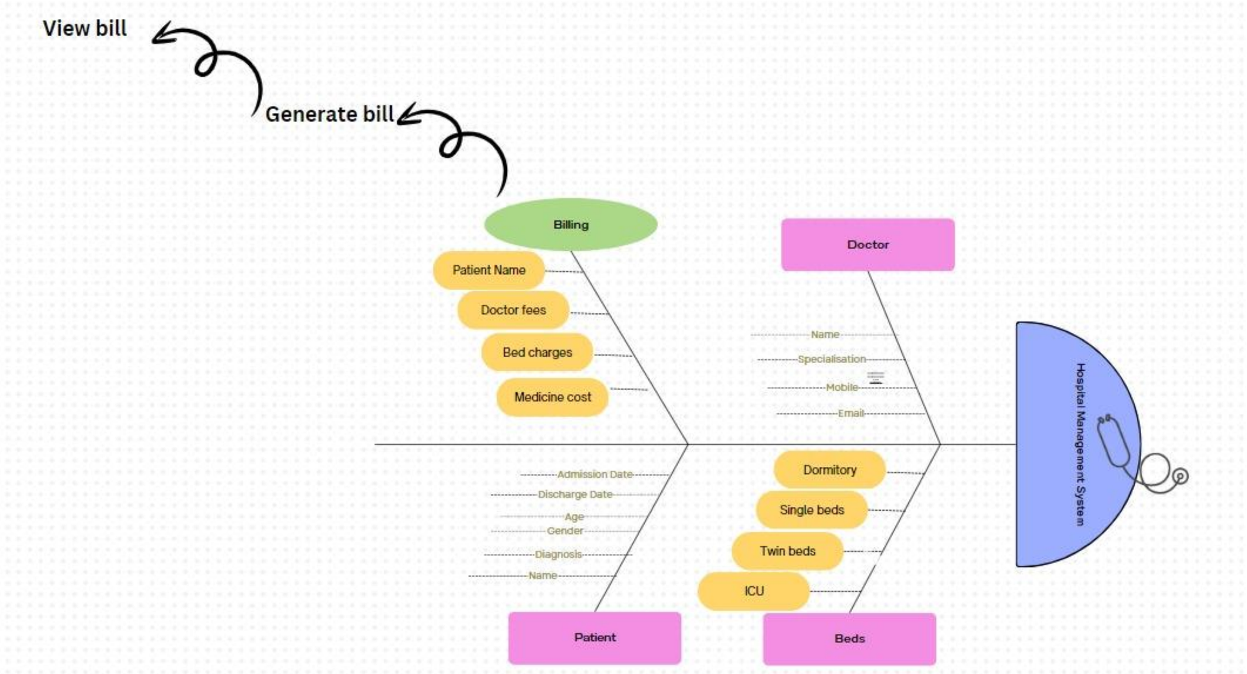
Eagles Batch 2022

Group: Tech Trio

Hospital Management System

S. No	Name	Reg. no	Roles
1	Debadrita Chattopadhyay	22070341	Team Lead
2	Debasrita Chattopadhyay	22070346	Developer
3	Mohika Jugele	22070274	Developer

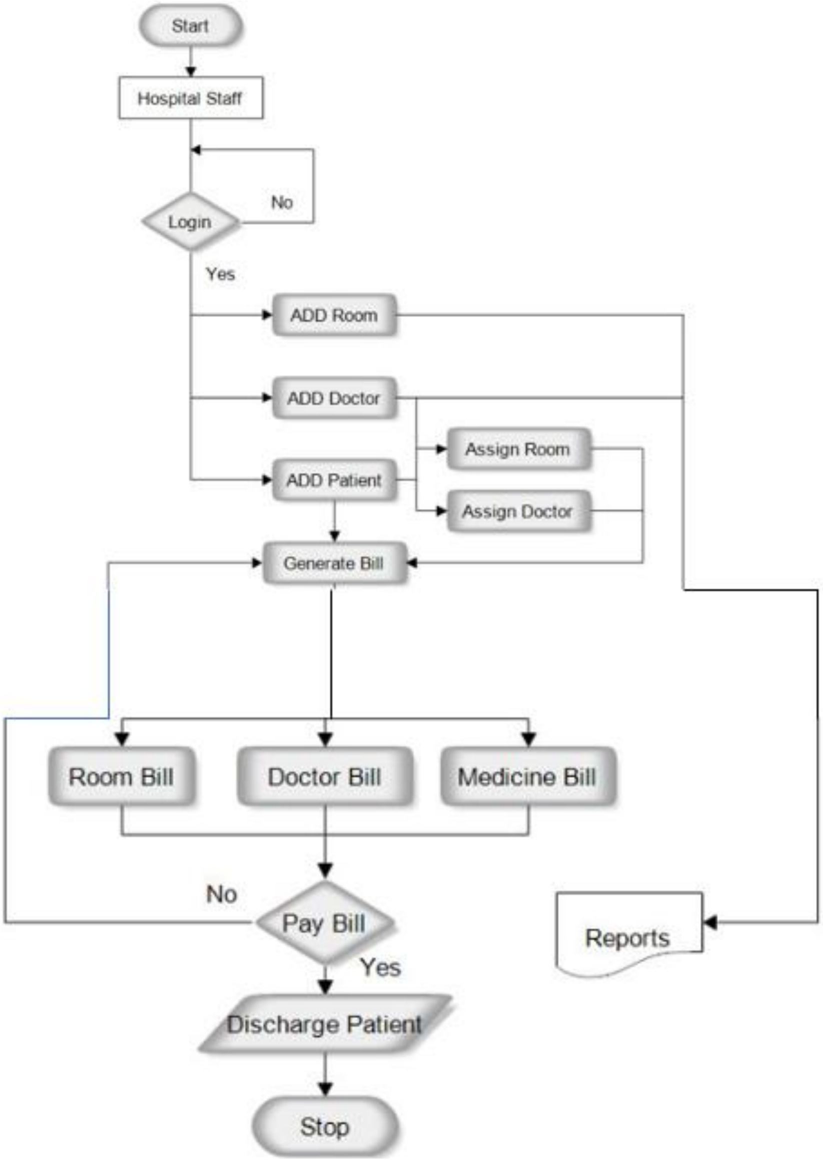
System Architecture and feature list



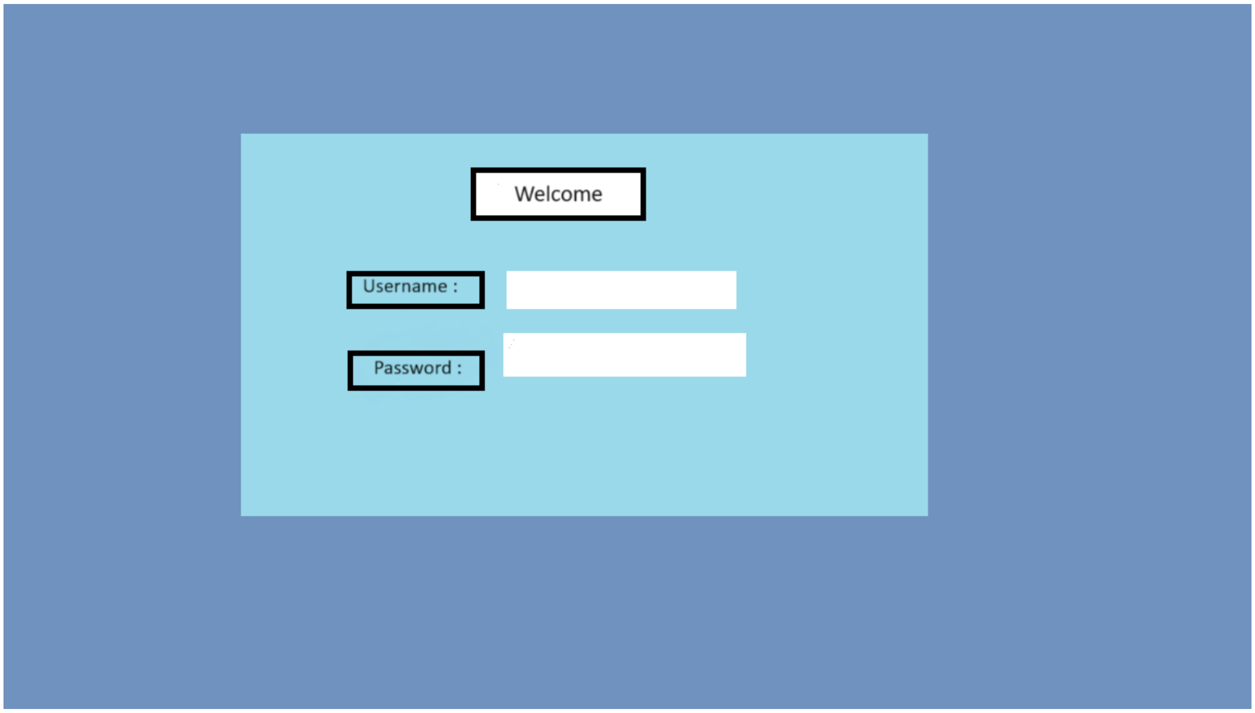
Feature list:

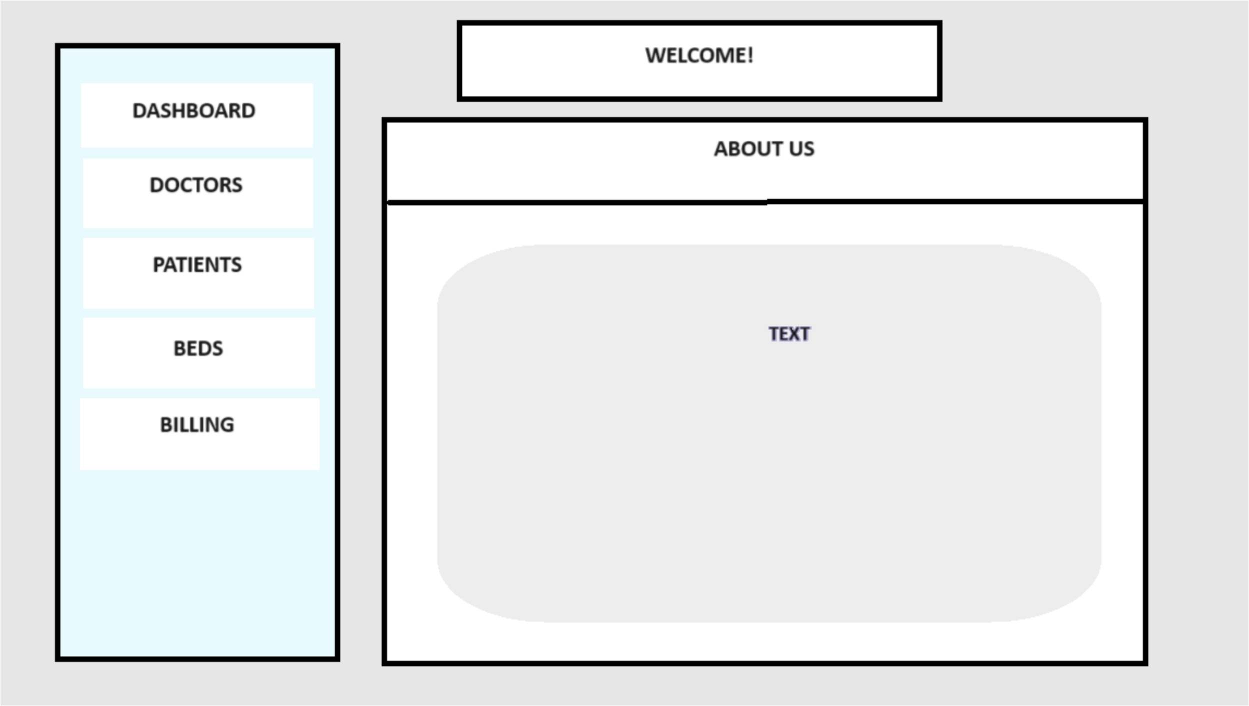
- Patient management (registration, admission, diagnosis)
- Appointment scheduling
- Billing
- Bed management

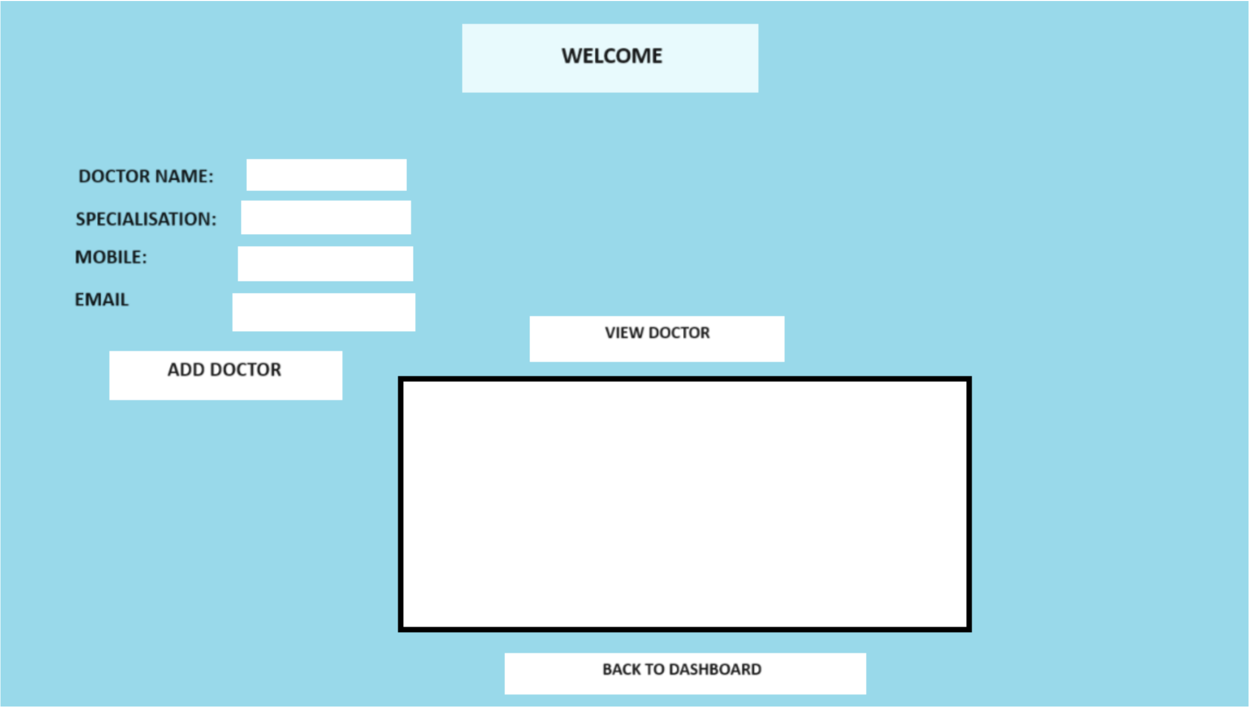
Logic Flow:

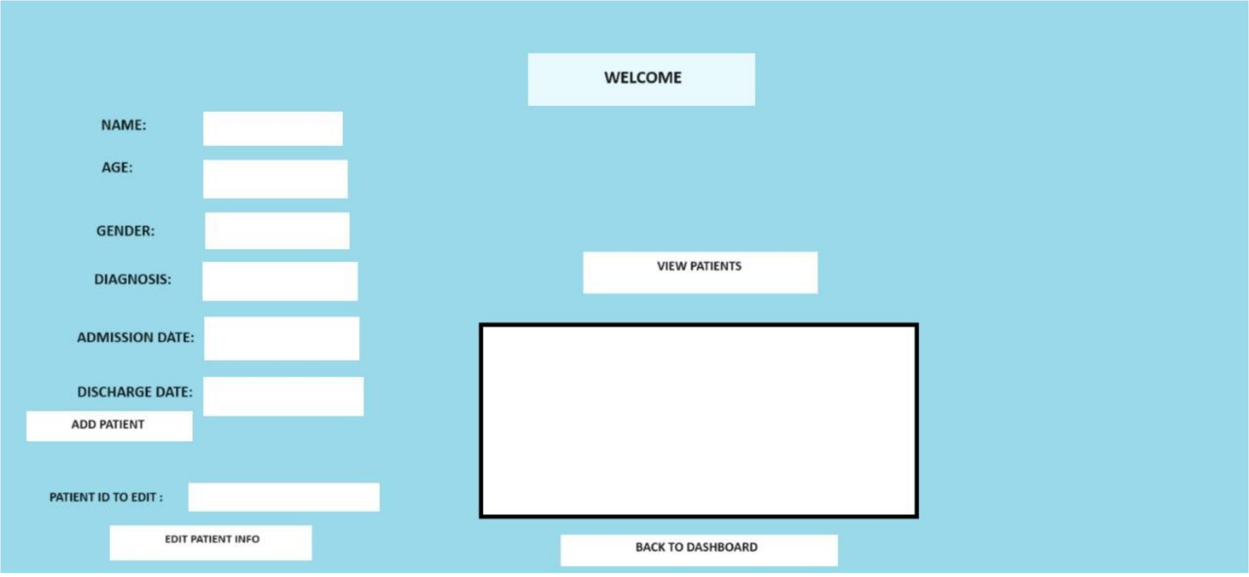


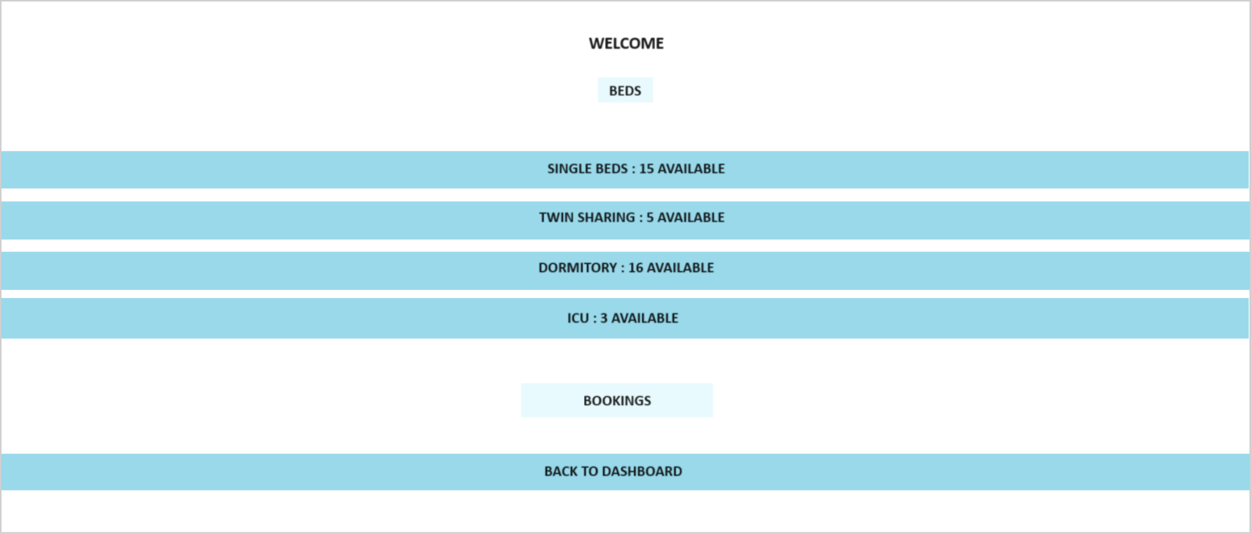
Rough low fidelity GUI design:






WELCOME
DOCTOR NAME:
SPECIALISATION:
MOBILE:
EMAIL
VIEW DOCTOR
ADD DOCTOR
BACK TO DASHBOARD


WELCOME
NAME:
AGE:
GENDER:
VIEW PATIENTS
DIAGNOSIS:
ADMISSION DATE:
DISCHARGE DATE:
ADD PATIENT
PATIENT ID TO EDIT :
EDIT PATIENT INFO
BACK TO DASHBOARD



WELCOME

BEDS

SINGLE BEDS : 15 AVAILABLE

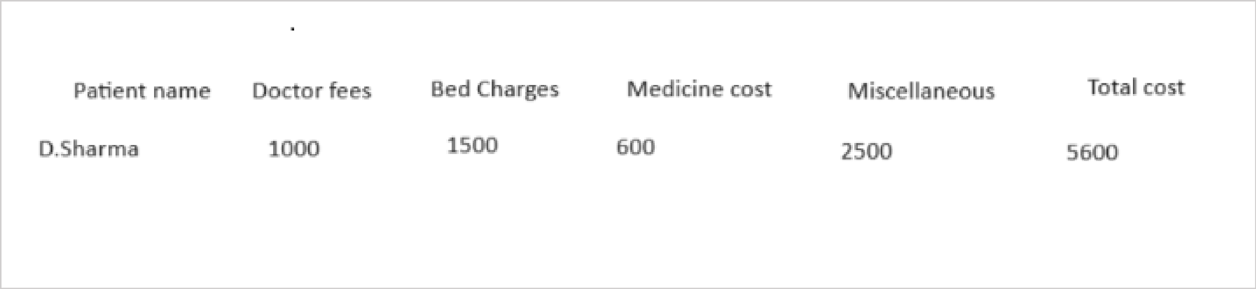
TWIN SHARING : 5 AVAILABLE

DORMITORY : 16 AVAILABLE

ICU : 3 AVAILABLE

BOOKINGS

BACK TO DASHBOARD



Patient name	Doctor fees	Bed Charges	Medicine cost	Miscellaneous	Total cost
D.Sharma	1000	1500	600	2500	5600

Technologies used:

Python and its packages used

- Tkinter 8.6 and ttk : For creating the GUI.
- Datetime: For handling date and time.

Database

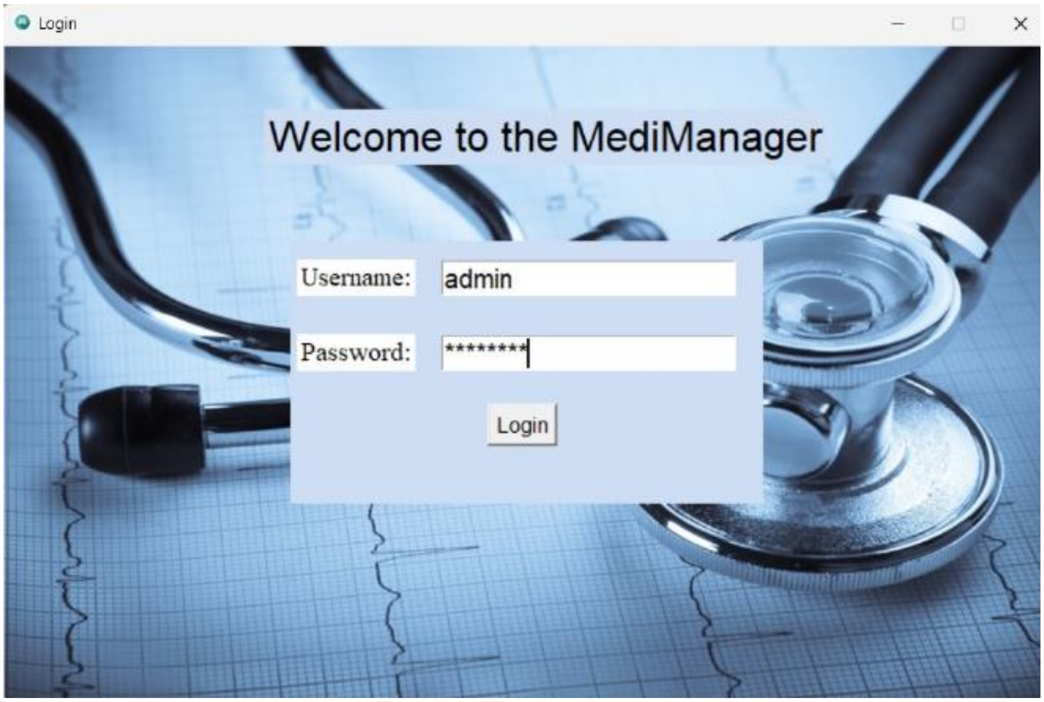
- Sqlite3

Third party package and its version: Pillow 10.4.0

Final UI screenshots

Login Window:

1. Accepts username and password.
2. Validates against user credentials.
3. Opens the dashboard upon successful login.



Login

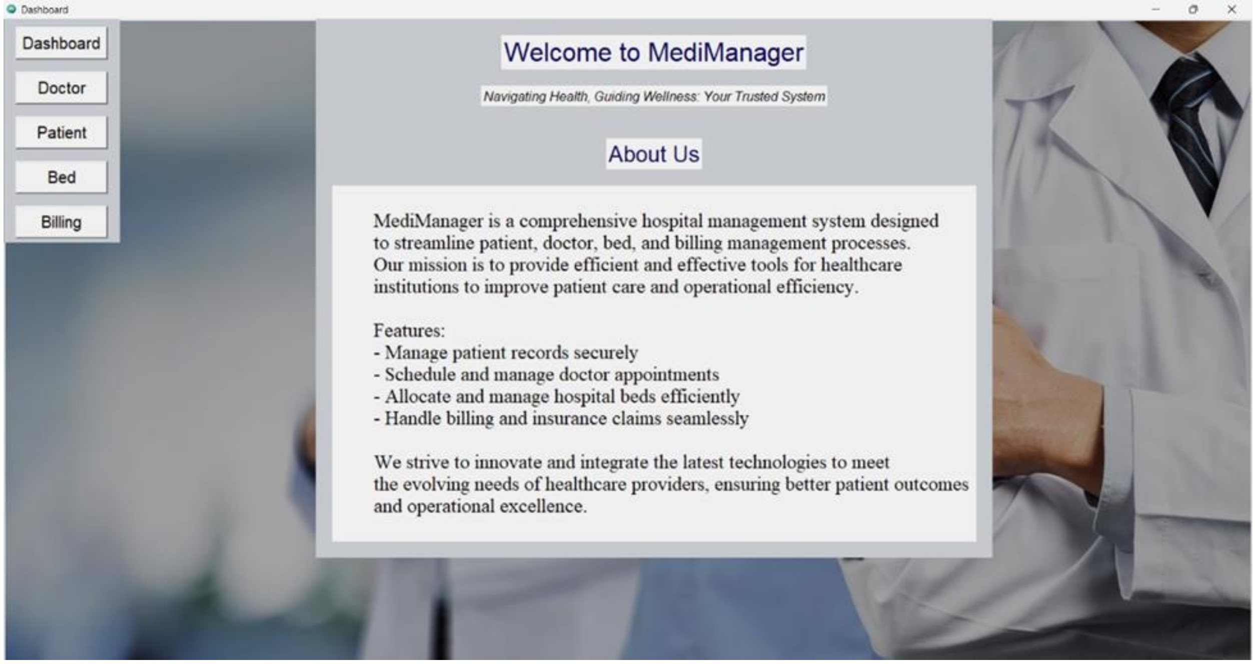
Welcome to the MediManager

Username: admin

Password: ********

Dashboard:

1. Provides navigation to different sections: Dashboard, Doctor, Patient, Bed, and Billing.
2. Initially displays a welcome message.



Welcome to MediManager

Navigating Health, Guiding Wellness: Your Trusted System

About Us

MediManager is a comprehensive hospital management system designed
to streamline patient, doctor, bed, and billing management processes.
Our mission is to provide efficient and effective tools for healthcare
institutions to improve patient care and operational efficiency.

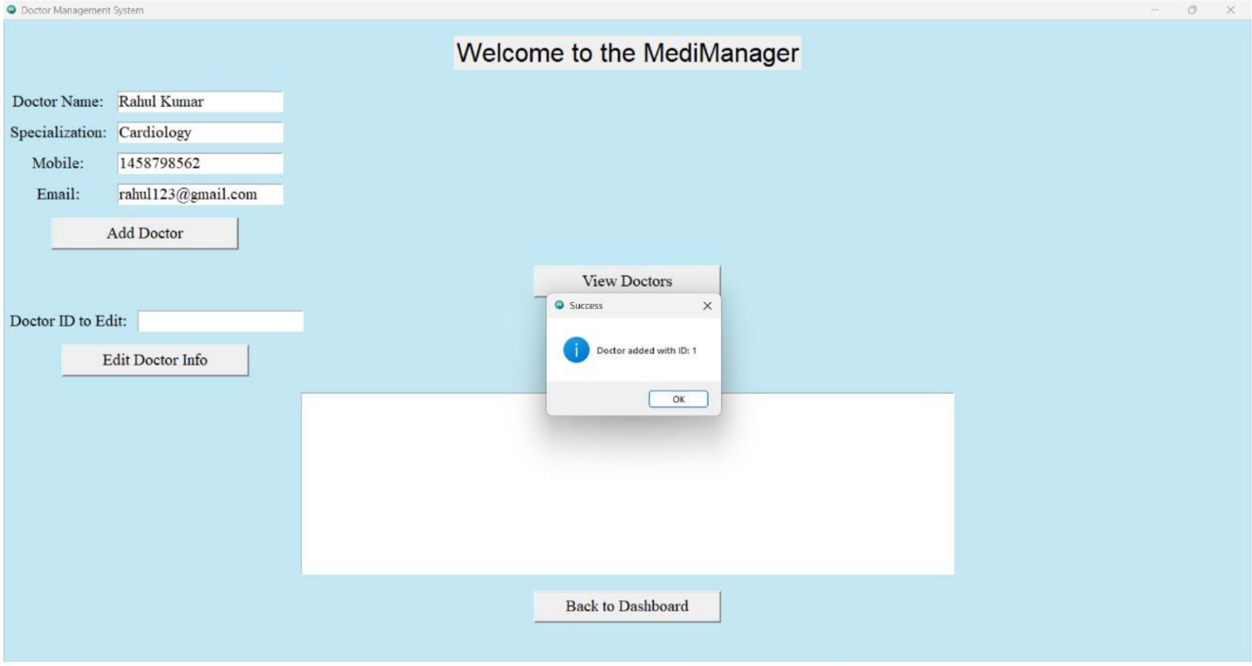
Features:
- Manage patient records securely
- Schedule and manage doctor appointments
- Allocate and manage hospital beds efficiently
- Handle billing and insurance claims seamlessly

We strive to innovate and integrate the latest technologies to meet
the evolving needs of healthcare providers, ensuring better patient outcomes
and operational excellence.

Doctor Management:

1. Allows adding doctors with details (name, specialization, mobile, email).
2. Displays a list of all doctors.
3. Allows editing doctor details by doctor ID.

Add doctor:



Welcome to the MediManager

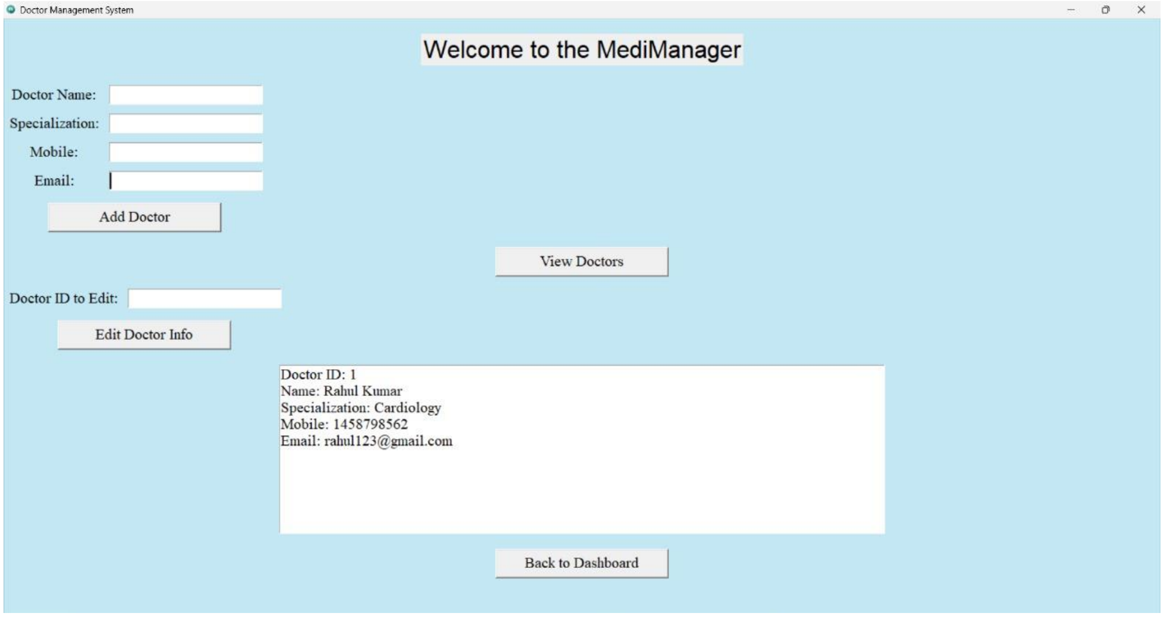
Doctor Name: Rahul Kumar
Specialization: Cardiology
Mobile: 1458798562
Email: rahul123@gmail.com

Doctor ID to Edit:

Success

Doctor added with ID: 1

View Doctors:

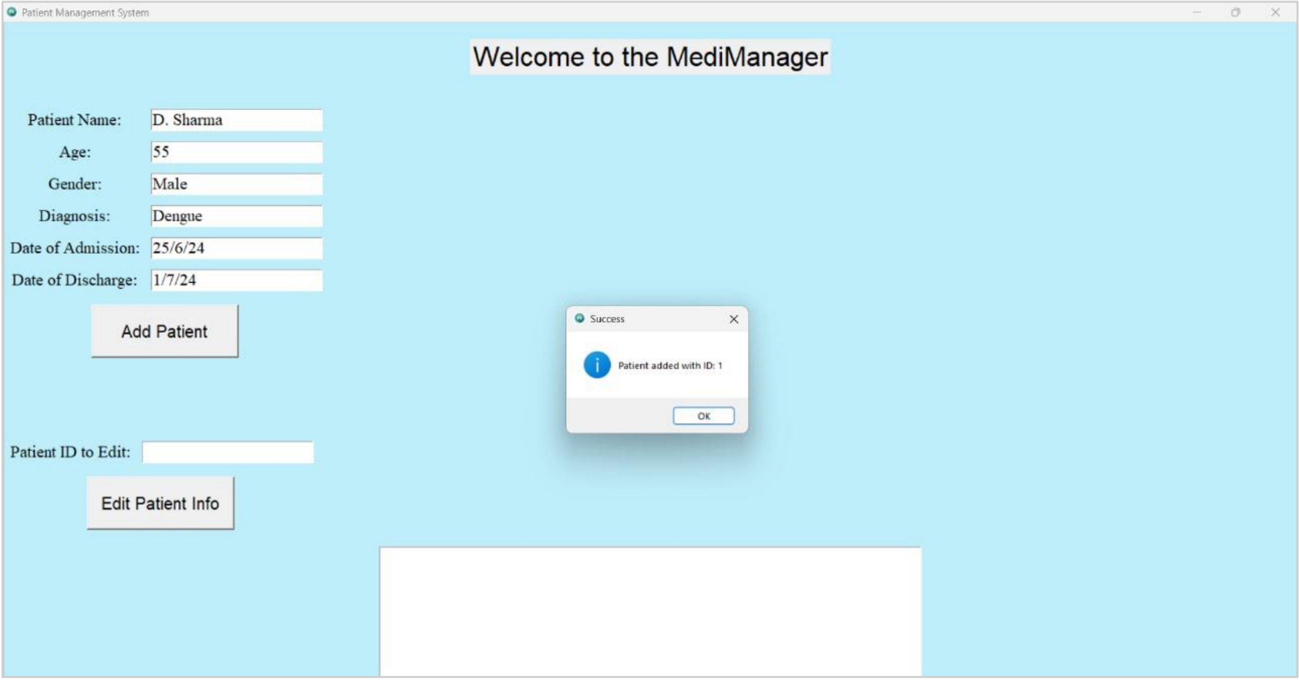


```
Doctor ID: 1
Name: Rahul Kumar
Specialization: Cardiology
Mobile: 1458798562
Email: rahul123@gmail.com
```

Patient Management:

1. Allows adding patients with details (name, age, gender, diagnosis, admission date, discharge date)Displays a list of all patients.
2. Allows editing patient details by patient ID.

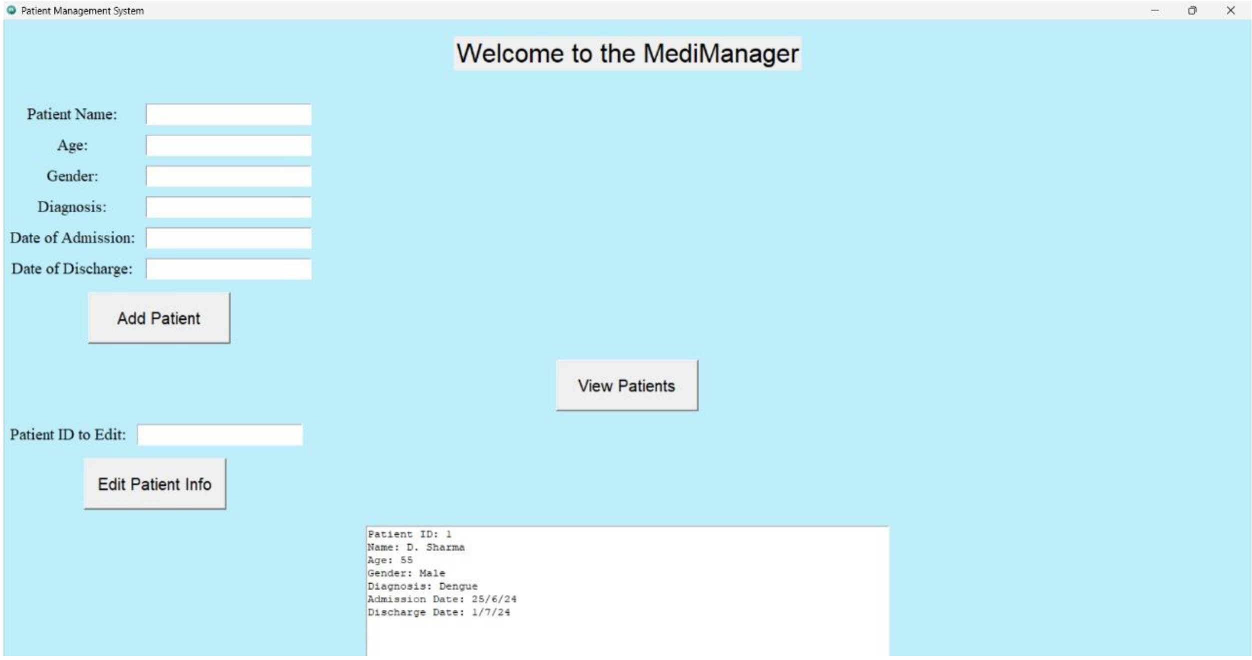
Add Patient:



Patient Name: D. Sharma
Age: 55
Gender: Male
Diagnosis: Dengue
Date of Admission: 25/6/24
Date of Discharge: 1/7/24

Patient added with ID: 1

View Patients:

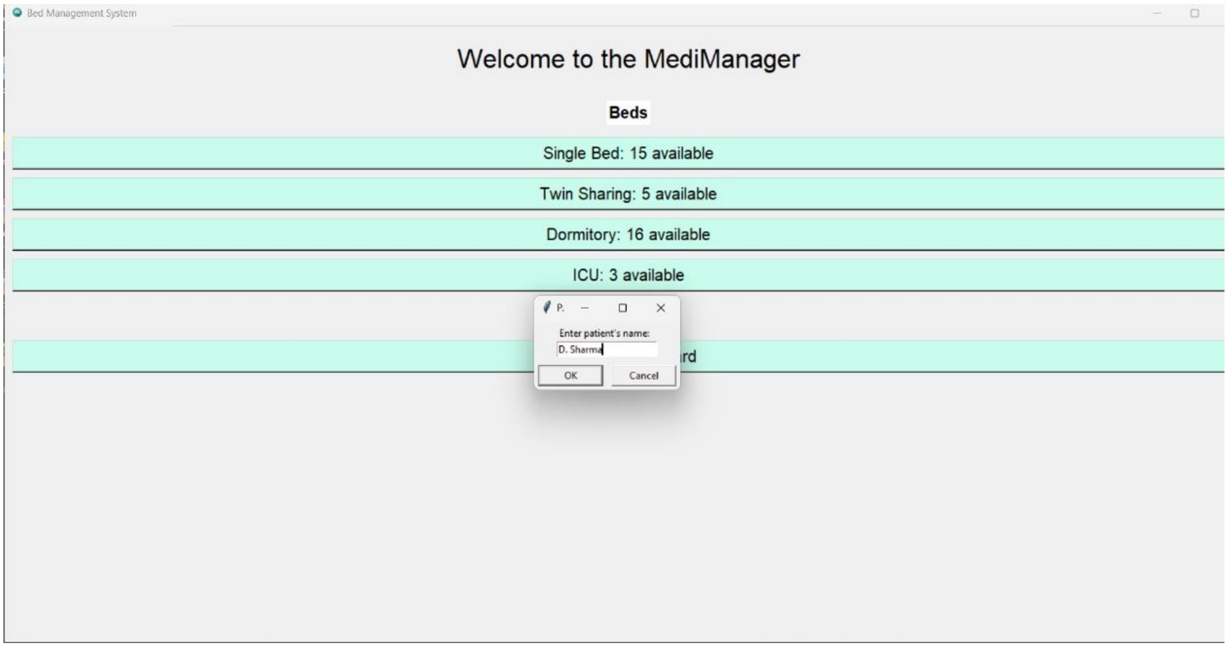


```
Patient ID: 1
Name: D. Sharma
Age: 55
Gender: Male
Diagnosis: Dengue
Admission Date: 25/6/24
Discharge Date: 1/7/24
```

Bed Management:

1. Displays available beds and bookings.
2. Allows booking beds by specifying the patient’s name.
3. Updates the number of available beds accordingly.

Select the type of beds from the options and enter the patient’s name:



Single Bed: 15 available

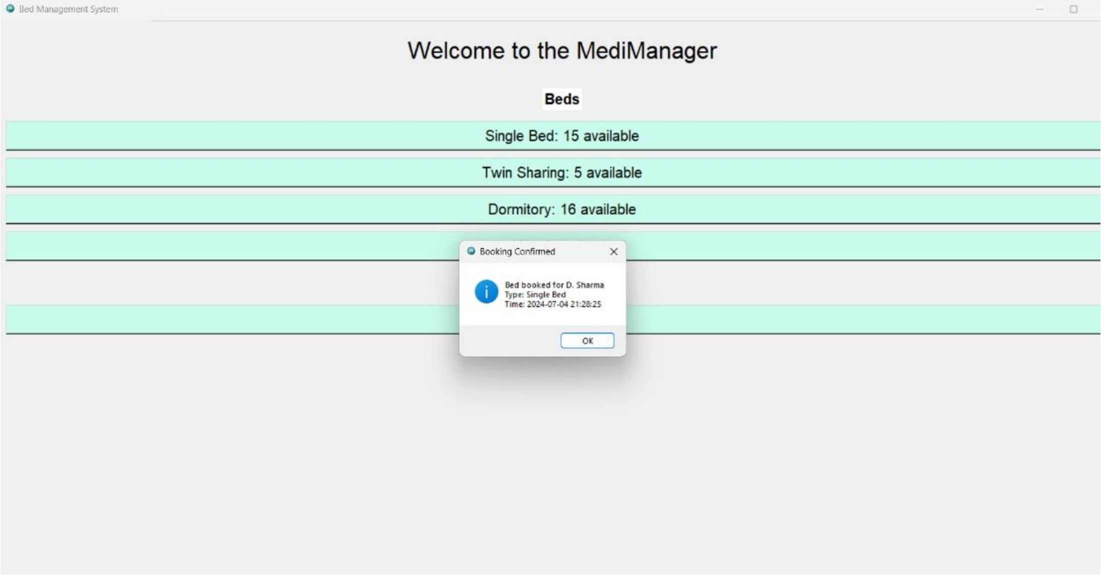
Twin Sharing: 5 available

Dormitory: 16 available

ICU: 3 available

Enter patient's name: D. Sharma

The type of bed booked, date and time of the bed booked is displayed .



Welcome to the MediManager

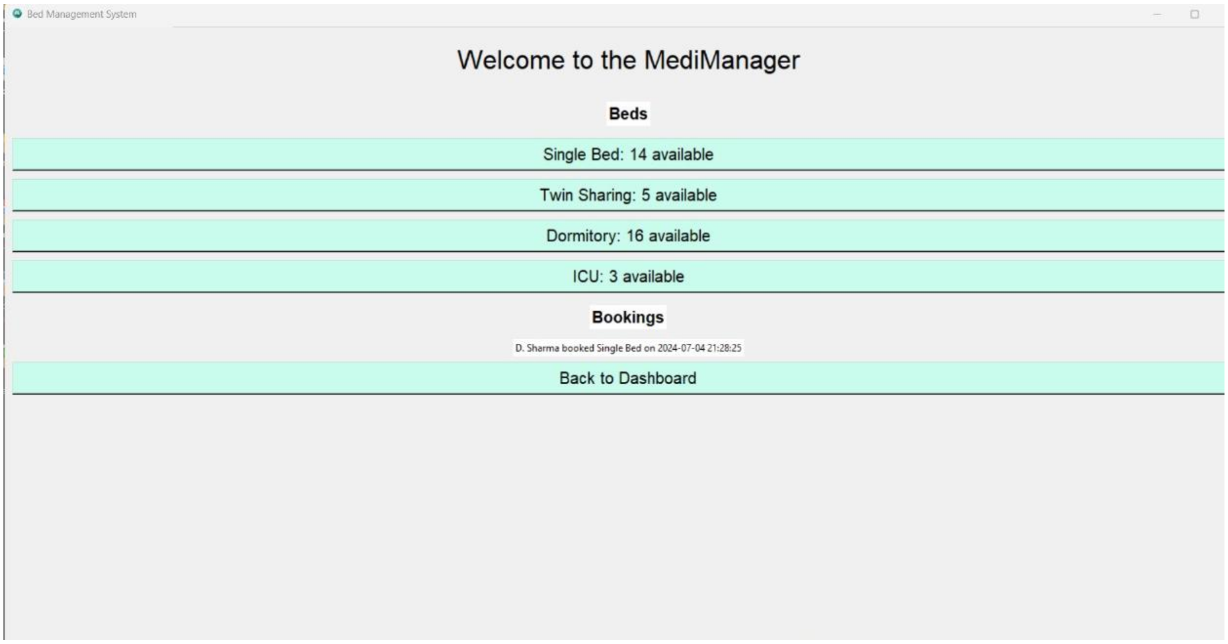
Beds

Single Bed: 15 available

Twin Sharing: 5 available

Dormitory: 16 available

Bed booked for D. Sharma
Type: Single Bed
Time: 2024-07-04 21:28:25



Welcome to the MediManager

Beds

Single Bed: 14 available

Twin Sharing: 5 available

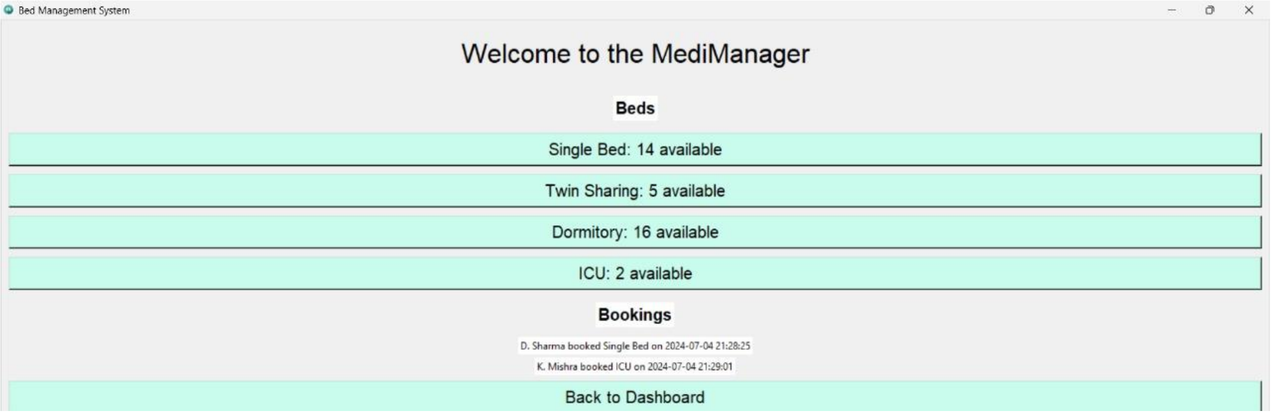
Dormitory: 16 available

ICU: 3 available

Bookings

D. Sharma booked Single Bed on 2024-07-04 21:28:25

The list of bookings along with patient name, bed type, date and time is displayed.


Bed Management System

Welcome to the MediManager

Beds

Single Bed: 14 available

Twin Sharing: 5 available

Dormitory: 16 available

ICU: 2 available

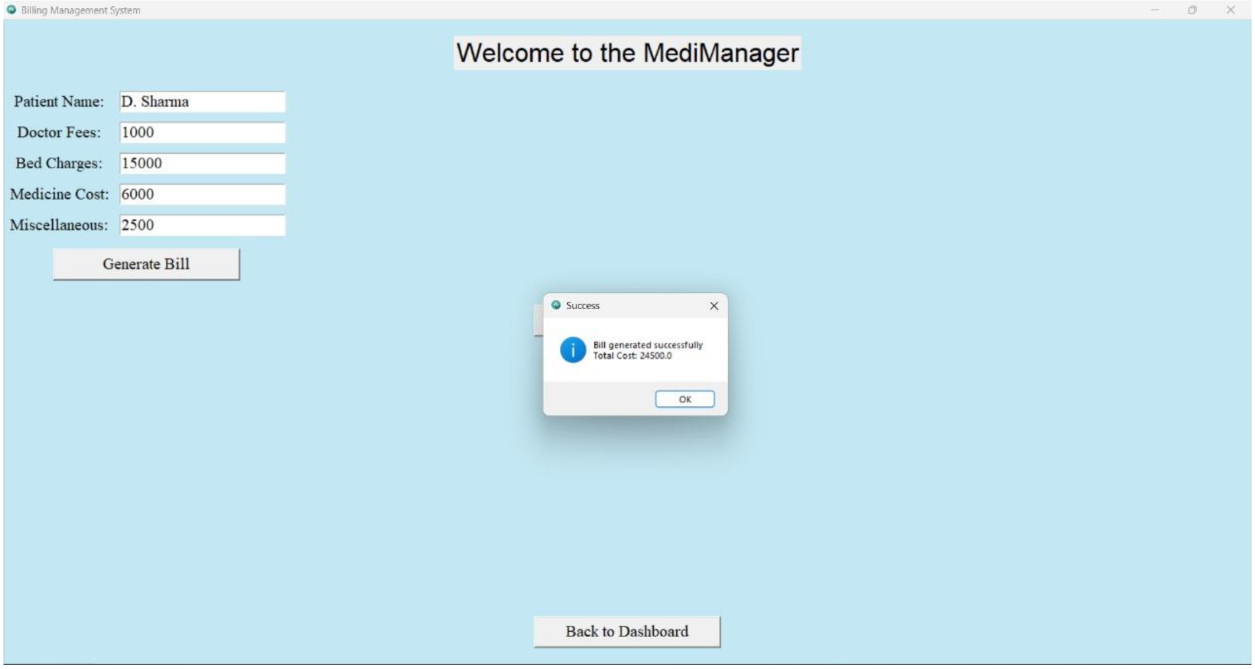
Bookings

D. Sharma booked Single Bed on 2024-07-04 21:28:25

K. Mishra booked ICU on 2024-07-04 21:29:01

Billing Management:

1. Provide a detailed breakdown of charges for medical services, medications, and other fees.


Billing Management System

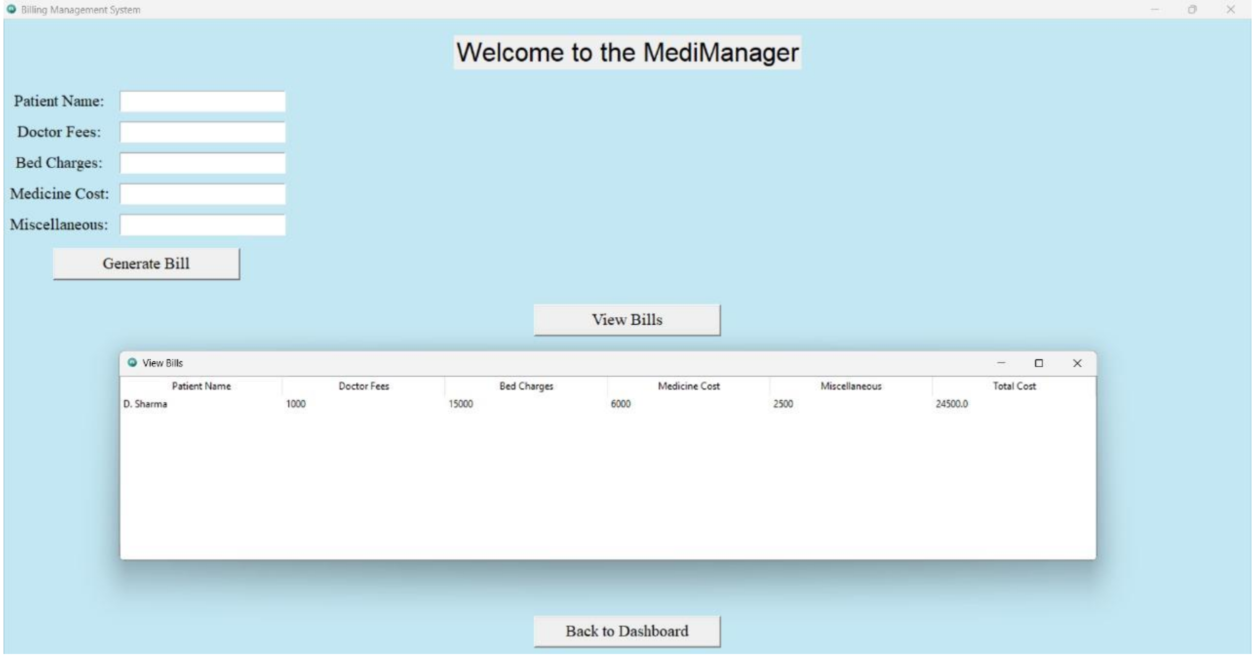
Welcome to the MediManager

Patient Name:	D. Sharma
Doctor Fees:	1000
Bed Charges:	15000
Medicine Cost:	6000
Miscellaneous:	2500

Success

Bill generated successfully
Total Cost: 24500.0

When View Bills button is clicked displays the bill of each patient.



Patient Name	Doctor Fees	Bed Charges	Medicine Cost	Miscellaneous	Total Cost
D. Sharma	1000	15000	6000	2500	24500.0

Database structure:

Table Doctor:

Column name	Data type
Name	Text
Specialization	Text
Mobile	Integer
Email	Text

Table Patient:

Column name	Data type
Name	Text
Age	Integer
Gender	Text
Diagnosis	Text
Admission date	Text
Discharge date	Text

Table Beds:

Column name	Data type
ICU	Integer
Single bed	Integer
Integer	Integer
Integer	Integer

Table Billing:

Column name	Data type
Patient name	Text
Doctor fees	Float
Bed charges	Float
Medicine cost	Float
Miscellaneous	Float

Doctor Table: Used to manage doctor profiles, contact details, and specialization.
Patient Table: Maintains patient records, including admission, diagnosis, and discharge information.
Beds Table: Tracks bed availability across different categories, aiding in patient accommodation planning.
Billing Table: Records financial transactions related to patient treatments, facilitating billing and financial reporting.

Database name - medimanager_db

List of classes and functions used:

Classes:

1. LoginWindow:
   - Represents the login window GUI where users input their credentials to access the system.
2. Dashboard:

- Represents the main dashboard GUI with navigation buttons to manage doctors, patients, beds, and billing.

3. DoctorManagement:
    - Manages the GUI and functionality related to adding, viewing, and editing doctor information.
4. PatientManagement:
    - Manages the GUI and functionality related to adding, viewing, and editing patient information.
5. BedManagement:
    - Manages the GUI and functionality related to viewing available beds and booking them.

6. BillingManagement
    - Manages the GUI and functionality related to viewing total bill and generates a bill.

Functions:

Class: LoginWindow

- set_full_size
- login
- open_dashboard

Class: Dashboard

- set_full_size
- show_dashboard
- open_doctor
- open_patient
- open_bed
- open_billing

Class: DoctorManagement

- set_full_size

- add_doctor
- view_doctors
- edit_doctor
- clear_doctor_entries
- back_to_dashboard

Class: PatientManagement

- set_full_size
- add_patient
- view_patients
- edit_patient
- clear_patient_entries
- back_to_dashboard

Class: BedManagement

- set_full_size
- show_beds_section
- book_bed
- back_to_dashboard

File structure and list of supporting files and dependencies:

1. medimanager (main file)
2. medimanager_data (for database connectivity)
3. medimanager_db (database)
4. medimanager_dbsqbpro (database)
5. doctor.jpg (background image)
6. hospital.jpg (background image)
7. icon.icon (icon image)

Outcome of the project

In this project, we've learned how to build a hospital system using Python and Tkinter. We focused on creating a simple and helpful interface for tasks like registering patients, scheduling appointments, handling bills, and managing beds. We practiced setting up a database to store and organize information efficiently using Sqlite3. Managing user logins and keeping track of doctor and patient details taught us a lot about designing software that's useful in healthcare settings. This project was a great way to improve our skills while understanding how important it is to design systems that make hospitals run smoother and better

Contribution of the members:

1. Debadrita Chattopadhyay: Front end, Database connectivity
2. Debasrita Chattopadhyay: Front end, Documentation
3. Mohika Jugele: Back end, Documentation