

# Patient Admission and Room Allocation GUI

## Basic Layout:

1. Patient Admission Form
2. Patients List
3. Appointments List
4. Available Rooms List
5. Booked Rooms List

### Hospital Management System

#### Admit Patient

Name:

Phone Number:

Gender:

Age:

Select Patient Type:

#### Patients

Patient A - 1234567890 - Male - 45 - Inpatient	<input type="button" value="Delete"/>
Patient B - 1234567891 - Female - 32 - Inpatient	<input type="button" value="Delete"/>
Patient C - 1234567892 - Other - 28 - Inpatient	<input type="button" value="Delete"/>

#### Patients

Patient A - 1234567890 - Male - 45 - Inpatient	<input type="button" value="Delete"/>
Patient B - 1234567891 - Female - 32 - Inpatient	<input type="button" value="Delete"/>
Patient C - 1234567892 - Other - 28 - Inpatient	<input type="button" value="Delete"/>
Patient D - 1234567893 - Male - 50 - Inpatient	<input type="button" value="Delete"/>
Patient E - 1234567894 - Female - 60 - Inpatient	<input type="button" value="Delete"/>
Patient F - 1234567895 - Other - 22 - Inpatient	<input type="button" value="Delete"/>
Patient G - 1234567896 - Male - 38 - Inpatient	<input type="button" value="Delete"/>
Patient H - 1234567897 - Female - 41 - Inpatient	<input type="button" value="Delete"/>
Patient I - 1234567898 - Other - 55 - Inpatient	<input type="button" value="Delete"/>
Patient J - 1234567899 - Male - 29 - Inpatient	<input type="button" value="Delete"/>
Patient K - 1234567800 - Female - 35 - Outpatient	<input type="button" value="Delete"/>
Patient L - 1234567801 - Other - 40 - Outpatient	<input type="button" value="Delete"/>
Patient M - 1234567802 - Male - 27 - Outpatient	<input type="button" value="Delete"/>
Patient N - 1234567803 - Female - 33 - Outpatient	<input type="button" value="Delete"/>

## Appointments

Appointment ID: 1 - 2024-01-01T00:00:00.000Z - 1970-01-01T09:00:00.000Z - Checkup - Status: Scheduled

 Delete

Appointment ID: 2 - 2024-02-01T00:00:00.000Z - 1970-01-01T10:00:00.000Z - Follow-up - Status: Scheduled

 Delete

Appointment ID: 3 - 2024-03-01T00:00:00.000Z - 1970-01-01T11:00:00.000Z - Vaccination - Status: Scheduled

 Delete

Appointment ID: 4 - 2024-04-01T00:00:00.000Z - 1970-01-01T14:00:00.000Z - Consultation - Status: Scheduled

 Delete

Appointment ID: 5 - 2024-05-01T00:00:00.000Z - 1970-01-01T15:00:00.000Z - Pain Management - Status: Scheduled

 Delete

Appointment ID: 6 - 2024-06-01T00:00:00.000Z - 1970-01-01T16:00:00.000Z - Allergy Test - Status: Scheduled

 Delete

Appointment ID: 7 - 2024-07-01T00:00:00.000Z - 1970-01-01T17:00:00.000Z - Dermatology - Status: Scheduled

 Delete

Appointment ID: 8 - 2024-08-01T00:00:00.000Z - 1970-01-01T18:00:00.000Z - Cardio Check - Status: Scheduled

 Delete

Appointment ID: 9 - 2024-09-01T00:00:00.000Z - 1970-01-01T19:00:00.000Z - Ortho Consult - Status: Scheduled

 Delete

Appointment ID: 12 - 2024-10-01T00:00:00.000Z - 1970-01-01T20:00:00.000Z - Post-Op - Status: Scheduled

 Delete

## Appointments

Appointment ID: 1 - 2024-01-01T00:00:00.000Z - 1970-01-01T09:00:00.000Z - Checkup - Status: Scheduled

 Delete

Appointment ID: 2 - 2024-02-01T00:00:00.000Z - 1970-01-01T10:00:00.000Z - Follow-up - Status: Scheduled

 Delete

Appointment ID: 3 - 2024-03-01T00:00:00.000Z - 1970-01-01T11:00:00.000Z - Vaccination - Status: Scheduled

 Delete

Appointment ID: 4 - 2024-04-01T00:00:00.000Z - 1970-01-01T14:00:00.000Z - Consultation - Status: Scheduled

 Delete

Appointment ID: 5 - 2024-05-01T00:00:00.000Z - 1970-01-01T15:00:00.000Z - Pain Management - Status: Scheduled

 Delete

Appointment ID: 6 - 2024-06-01T00:00:00.000Z - 1970-01-01T16:00:00.000Z - Allergy Test - Status: Scheduled

 Delete

Appointment ID: 7 - 2024-07-01T00:00:00.000Z - 1970-01-01T17:00:00.000Z - Dermatology - Status: Scheduled

 Delete

Appointment ID: 8 - 2024-08-01T00:00:00.000Z - 1970-01-01T18:00:00.000Z - Cardio Check - Status: Scheduled

 Delete

Appointment ID: 9 - 2024-09-01T00:00:00.000Z - 1970-01-01T19:00:00.000Z - Ortho Consult - Status: Scheduled

 Delete

### Available Rooms

- Room 6 - Double - \$170/day
- Room 9 - Single - \$210/day

### Booked Rooms

- Room 1 - Single - \$200/day - Occupied
- Room 2 - Double - \$150/day - Occupied
- Room 4 - Emergency - \$300/day - Occupied
- Room 5 - Single - \$220/day - Occupied
- Room 7 - ICU - \$550/day - Occupied
- Room 10 - Double - \$160/day - Occupied
- Room 1 - Single - \$200/day - Occupied
- Room 7 - ICU - \$550/day - Occupied
- Room 4 - Emergency - \$300/day - Occupied

## Hospital Management System

### Admit Patient

Select Patient Type  
Inpatient  
Outpatient

### Patients

Patient C - 1234567892 - Other - 28 - Inpatient	<input type="button" value="Delete"/>
Patient D - 1234567893 - Male - 50 - Inpatient	<input type="button" value="Delete"/>
Patient E - 1234567894 - Female - 60 - Inpatient	<input type="button" value="Delete"/>

# Hospital Management System

Admit Patient

Patient A  
5181241456  
Female  
21  
Outpatient  
02/04/2025  
100

Admit Patient

Patients

Patient C - 1234567802 - Other - 28 - Inpatient Delete

Patient P - 1234567805 - Male - 50 - Outpatient Delete

Patient Q - 1234567806 - Female - 29 - Outpatient Delete

Patient R - 1234567807 - Other - 42 - Outpatient Delete

Patient S - 1234567808 - Male - 37 - Outpatient Delete

asdf - 1234567890 - Female - 21 - Inpatient Delete

sdfg - 1234567890 - Female - 21 - Inpatient Delete

Patient A - 5181241456 - Female - 21 - Outpatient Delete

# Hospital Management System

Admit Patient

Patient B  
5181241456  
Male  
21  
Inpatient  
Select Room  
Select Doctor

Admit Patient

Patients

Patient C - 1234567802 - Other - 28 - Inpatient Delete



# Hospital Management System

## Admit Patient

Patient B

5181241456

Male

21

Inpatient

✓ Select Room

- Room 1 - Single
- Room 2 - Double
- Room 4 - Emergency
- Room 6 - Double
- Room 7 - ICU
- Room 9 - Single

## Patients



# Hospital Management System

## Admit Patient

Patient B

5181241456

Male

21

Inpatient

Room 1 - Single

Dr. Samuel Lee

Admit Patient

## Available Rooms

 Room 1 - Single - \$200/day

 Room 2 - Double - \$150/day

 Room 4 - Emergency - \$300/day

 Room 6 - Double - \$170/day

 Room 7 - ICU - \$550/day

 Room 9 - Single - \$210/day

## Booked Rooms

 Room 5 - Single - \$220/day - Occupied

 Room 7 - ICU - \$550/day - Occupied

 Room 10 - Double - \$160/day - Occupied

## Available Rooms

 Room 2 - Double - \$150/day

 Room 4 - Emergency - \$300/day

 Room 6 - Double - \$170/day

 Room 7 - ICU - \$550/day

 Room 9 - Single - \$210/day

## Booked Rooms

 Room 5 - Single - \$220/day - Occupied

 Room 7 - ICU - \$550/day - Occupied

 Room 10 - Double - \$160/day - Occupied

 Room 1 - Single - \$200/day - Occupied

 Room 1 - Single - \$200/day - Occupied



# API Endpoints Overview

Feature	Endpoint	Method	Description
Register Patient	/patients	POST	Adds a new patient (Inpatient/Outpatient)
Get All Patients	/patients	GET	Fetches all patients
Update Patient	/patients/:id	PUT	Updates patient details
Delete Patient	/delete-patient/:id	DELETE	Deletes patient and all related records
Admit Patient	/admit	POST	Admits patient (assigns room if inpatient)
Get Available Rooms	/rooms	GET	Fetches available rooms
Get Booked Rooms	/bookedrooms	GET	Fetches occupied rooms and their patients
Discharge Inpatient	/discharge/:id	POST	Marks inpatient as discharged, frees room
Get Doctors	/doctors	GET	Fetches list of doctors
Get Appointments	/appointments	GET	Fetches all the appointments in the DB
Delete Appointments	/appointments/:id	DELETE	Deletes an appointment

## Handling Delete for patient:

The SQL Server trigger (`trg_CascadeDeletePatient`) is designed to handle the deletion of patient records from your hospital database system in an organized way. Here's how it works:

### Purpose

This trigger fires INSTEAD OF a direct DELETE operation on the Patient table. This means it intercepts any attempt to delete a patient and handles the deletion process in a custom way.

### Key Features

1. Patient Type Handling: The trigger recognizes that a patient can be either an Inpatient or Outpatient (but not both), and handles the deletion differently based on the patient type.

2. Room Management: For Inpatients specifically, it updates the Room status from "Occupied" to "Available" when the patient is deleted.
3. Handling of Optional Relationships: It safely handles cases where a patient may not have prescription medications or other optional related records.
4. Relational Integrity: The trigger maintains database integrity by deleting related records in a specific order:
  - o First deletes PrescriptionMedication links
  - o Then deletes Prescriptions
  - o Then deletes various records (medical records, appointments, care assignments, billing)
  - o Then conditionally deletes either from Inpatient or Outpatient table
  - o Finally deletes the Patient record itself
5. Conditional Processing: Using a cursor, it processes each deleted patient individually, checking their type and performing the appropriate deletion.

**Trigger used:**

```

CREATE TRIGGER trg_CascadeDeletePatient
ON Patient
INSTEAD OF DELETE
AS
BEGIN
SET NOCOUNT ON;

-- Only update rooms for inpatients

UPDATE Room
SET Status = 'Available'
FROM Room
INNER JOIN Inpatient ON Room.RoomID = Inpatient.RoomID
INNER JOIN deleted ON Inpatient.PatientID = deleted.PatientID

```

```

WHERE deleted.PatientType = 'Inpatient';

-- Delete PrescriptionMedication links only if they exist

DELETE pm

FROM PrescriptionMedication pm

INNER JOIN Prescription p ON pm.PrescriptionID = p.PrescriptionID

INNER JOIN deleted d ON p.PatientID = d.PatientID;

-- Delete Prescriptions if they exist

DELETE FROM Prescription WHERE PatientID IN (SELECT PatientID FROM deleted);

-- Delete other related records

DELETE FROM MedicalRecord WHERE PatientID IN (SELECT PatientID FROM deleted);

DELETE FROM Appointment WHERE PatientID IN (SELECT PatientID FROM deleted);

DELETE FROM CareAssignment WHERE PatientID IN (SELECT PatientID FROM deleted);

DELETE FROM Billing WHERE PatientID IN (SELECT PatientID FROM deleted);

-- Delete Inpatient or Outpatient records conditionally

DECLARE @PatientCursor CURSOR;

DECLARE @PatientID INT;

DECLARE @PatientType VARCHAR(20);

SET @PatientCursor = CURSOR FOR

    SELECT PatientID, PatientType

```

```
        FROM deleted;

OPEN @PatientCursor;

FETCH NEXT FROM @PatientCursor INTO @PatientID, @PatientType;

WHILE @@FETCH_STATUS = 0

BEGIN

    IF @PatientType = 'Inpatient'

        DELETE FROM Inpatient WHERE PatientID = @PatientID;

    ELSE

        DELETE FROM Outpatient WHERE PatientID = @PatientID;

    FETCH NEXT FROM @PatientCursor INTO @PatientID, @PatientType;

END;

CLOSE @PatientCursor;

DEALLOCATE @PatientCursor;

-- Delete the Patient records

DELETE FROM Patient WHERE PatientID IN (SELECT PatientID FROM deleted);

END;

GO
```