



**University of Central Punjab**  
**Faculty of Information and Technology**  
**Introduction to Data Base**

**Project**

**Project Title:** Hospital Management System

**Group Members**

- Faaiz Ashiq (L1f22bscs0579)
- Asad Faisal Butt (L1f22bscs0574)
- M. Ammar Dogar (L1f22bscs0077)

## *Project Overview:*

The Hospital Management System (HMS) is designed to improve the efficiency and quality of healthcare management within a hospital. This system will manage patient records, doctor and nurse details, appointments, treatments, prescriptions, and room allocations. The HMS will provide a user-friendly interface for hospital staff to access and update information swiftly, ensuring better patient care and streamlined operations.

## *Objectives:*

- **Efficient Patient Management:** Maintain comprehensive records of patient information, medical history, and treatment details.
- **Appointment Scheduling:** Streamline the process of scheduling appointments with doctors and managing appointment data.
- **Treatment and Prescription Management:** Track treatments administered to patients and manage prescriptions accurately.
- **Resource Allocation:** Manage room availability and allocation for patients.
- **Staff Management:** Maintain details of doctors, nurses, and other medical staff.
- **Reporting:** Generate various reports for administrative and medical purposes.

## *Scope:*

The system will cover the following functionalities:

- **Patient Management:** Registration, updates, and record-keeping.
- **Doctor and Nurse Management:** Profiles, specializations, and contact details.
- **Appointment Management:** Scheduling, rescheduling, and cancellations.
- **Treatment Records:** Documentation of treatments administered.
- **Prescription Management:** Managing medications prescribed to patients.
- **Room Management:** Room assignment and availability status.
- **Reports:** Generation of reports such as patient history, staff duty rosters, and room occupancy.

## Sample Working of Project:

### Insertion Sample:

This Picture shows a sample to insert data into Data Base.

- ```
INSERT INTO Treatments (TreatmentID, PatientID, DoctorID, TreatmentDate, Description) VALUES
(1, 1, 1, '2024-07-10', 'Initial evaluation and treatment plan'),
(2, 2, 2, '2024-07-11', 'Diagnosis and follow-up plan'),
(3, 3, 3, '2024-07-12', 'Consultation and treatment options discussed'),
(4, 4, 4, '2024-07-13', 'Examination and treatment recommendations'),
(5, 5, 5, '2024-07-14', 'Oncology screening and treatment strategy'),
(6, 6, 6, '2024-07-15', 'Skin condition assessment and treatment'),
(7, 7, 7, '2024-07-16', 'Heart condition evaluation and treatment'),
(8, 8, 8, '2024-07-17', 'Neurological assessment and treatment'),
(9, 9, 9, '2024-07-18', 'Follow-up and orthopedic care'),
(10, 10, 10, '2024-07-19', 'Pediatric assessment and treatment');
```
- ```
INSERT INTO Rooms (RoomID, RoomNumber, RoomType, Availability) VALUES
(1, '101', 'Single', TRUE),
(2, '102', 'Double', TRUE),
(3, '103', 'Suite', TRUE),
(4, '104', 'ICU', FALSE),
(5, '105', 'Single', TRUE),
(6, '106', 'Double', FALSE),
(7, '107', 'Suite', TRUE),
(8, '108', 'ICU', TRUE),
(9, '109', 'Single', TRUE),
```

### Working using Queries:

#### Join Queries

#### 1. Patients and their Appointments:

```
SELECT p.Name, a.AppointmentDate, a.Reason
FROM Patients p
JOIN Appointments a ON p.PatientID = a.PatientID;
```

*Reason: To retrieve a list of patients along with their appointment details.*

#### 2. Doctors and their Specialties:

```
SELECT d.Name, d.Specialty, a.AppointmentDate
```

```
FROM Doctors d
JOIN Appointments a ON d.DoctorID = a.DoctorID;
```

*Reason: To see which doctor is handling which appointment based on their specialty.*

**3. Patients and their Treatments:**

```
SELECT p.Name, t.TreatmentDate, t.Description
FROM Patients p
JOIN Treatments t ON p.PatientID = t.PatientID;
```

*Reason: To get information about the treatments received by each patient.*

**4. Doctors and their Treatments:**

```
SELECT d.Name, t.TreatmentDate, t.Description
FROM Doctors d
JOIN Treatments t ON d.DoctorID = t.DoctorID;
```

*Reason: To list the treatments administered by each doctor.*

**5. Treatments and Prescribed Medications:**

```
SELECT t.Description, m.Name, p.Dosage
FROM Treatments t
JOIN Prescriptions p ON t.TreatmentID = p.TreatmentID
JOIN Medications m ON p.MedicationID = m.MedicationID;
```

*Reason: To see which medications are prescribed for specific treatments.*

**6. Patients and their Room Assignments:**

```
SELECT p.Name, r.RoomNumber, r.RoomType
FROM Patients p
JOIN PatientRooms pr ON p.PatientID = pr.PatientID
JOIN Rooms r ON pr.RoomID = r.RoomID;
```

*Reason: To know which patients are assigned to which rooms.*

7.

**Doctors and Patients through Appointments:**

```
SELECT d.Name AS DoctorName, p.Name AS PatientName, a.AppointmentDate
FROM Doctors d
JOIN Appointments a ON d.DoctorID = a.DoctorID
JOIN Patients p ON a.PatientID = p.PatientID;
```

*Reason: To see which doctors are attending to which patients and when.*

8. **Nurses and Patients through Room Assignments:**

```
SELECT n.Name AS NurseName, p.Name AS PatientName, r.RoomNumber
FROM Nurses n
JOIN PatientRooms pr ON n.NurseID = pr.NurseID
JOIN Patients p ON pr.PatientID = p.PatientID
JOIN Rooms r ON pr.RoomID = r.RoomID;
```

*Reason: To find out which nurses are assigned to which patients in specific rooms.*

9. **Patients and their Prescriptions:**

```
SELECT p.Name AS PatientName, m.Name AS MedicationName, pr.Dosage,
pr.Frequency
FROM Patients p
JOIN Treatments t ON p.PatientID = t.PatientID
JOIN Prescriptions pr ON t.TreatmentID = pr.TreatmentID
JOIN Medications m ON pr.MedicationID = m.MedicationID;
```

*Reason: To get details about medications prescribed to patients.*

10. **Doctors and Prescribed Medications:**

```
SELECT d.Name AS DoctorName, m.Name AS MedicationName, pr.Dosage,
pr.Frequency
FROM Doctors d
JOIN Treatments t ON d.DoctorID = t.DoctorID
JOIN Prescriptions pr ON t.TreatmentID = pr.TreatmentID
JOIN Medications m ON pr.MedicationID = m.MedicationID;
```

*Reason: To see which doctors are prescribing which medications.*

11. **Available Rooms:**

```
SELECT RoomNumber, RoomType
FROM Rooms
```

WHERE Availability = TRUE;

*Reason: To find out which rooms are currently available.*

## 12. Patients and Doctors Interaction Count:

```
SELECT p.Name AS PatientName, d.Name AS DoctorName, COUNT(*) AS  
InteractionCount  
FROM Patients p  
JOIN Appointments a ON p.PatientID = a.PatientID  
JOIN Doctors d ON a.DoctorID = d.DoctorID  
GROUP BY p.Name, d.Name;
```

*Reason: To know how many times a patient has interacted with a specific doctor.*

## 13. Treatments and the Number of Medications Prescribed:

```
SELECT t.Description, COUNT(pr.MedicationID) AS MedicationCount  
FROM Treatments t  
JOIN Prescriptions pr ON t.TreatmentID = pr.TreatmentID  
GROUP BY t.Description;
```

*Reason: To see the number of medications prescribed for each treatment.*

## 14. Doctors and Patients Treated:

```
SELECT d.Name AS DoctorName, COUNT(t.PatientID) AS PatientsTreated  
FROM Doctors d  
JOIN Treatments t ON d.DoctorID = t.DoctorID  
GROUP BY d.Name;
```

*Reason: To know how many patients each doctor has treated.*

## 15. Patients with Multiple Appointments:

```
SELECT p.Name AS PatientName, COUNT(a.AppointmentID) AS AppointmentCount  
FROM Patients p  
JOIN Appointments a ON p.PatientID = a.PatientID  
GROUP BY p.Name  
HAVING COUNT(a.AppointmentID) > 1;
```

*Reason: To find patients who have had more than one appointment.*

## Aggregate Functions

### 1. Total Number of Patients:

```
SELECT COUNT(*) AS TotalPatients  
FROM Patients;
```

*Reason: To get the total count of patients in the system.*

**2. Average Age of Patients:**

```
SELECT AVG(YEAR(CURDATE()) - YEAR(DOB)) AS AverageAge  
FROM Patients;
```

*Reason: To find the average age of patients.*

**3. Total Number of Appointments per Doctor:**

```
SELECT d.Name, COUNT(a.AppointmentID) AS TotalAppointments  
FROM Doctors d  
JOIN Appointments a ON d.DoctorID = a.DoctorID  
GROUP BY d.Name;
```

*Reason: To know the number of appointments handled by each doctor.*

**4. Total Medications Prescribed:**

```
SELECT COUNT(*) AS TotalMedications  
FROM Prescriptions;
```

*Reason: To get the total number of medications prescribed.*

**5. Maximum Number of Patients Treated by a Doctor:**

```
SELECT MAX(PatientCount) AS MaxPatientsTreated  
FROM (SELECT COUNT(t.PatientID) AS PatientCount  
      FROM Doctors d  
      JOIN Treatments t ON d.DoctorID = t.DoctorID  
      GROUP BY d.Name) AS PatientCounts;
```

*Reason: To find the maximum number of patients treated by any single doctor.*

## Correlated Queries

**1. Patients with More Appointments than the Average:**

```
SELECT p.Name  
FROM Patients p
```

```
WHERE (SELECT COUNT(*) FROM Appointments a WHERE a.PatientID =  
p.PatientID) >  
      (SELECT AVG(AppointmentCount) FROM (SELECT COUNT(*) AS  
AppointmentCount FROM Appointments GROUP BY PatientID) AS  
AvgAppointments);
```

*Reason: To find patients who have more appointments than the average number of appointments per patient.*

**2. Doctors Treating More Than the Average Number of Patients:**

```
SELECT d.Name  
FROM Doctors d  
WHERE (SELECT COUNT(*) FROM Treatments t WHERE t.DoctorID = d.DoctorID)  
>  
      (SELECT AVG(TreatmentCount) FROM (SELECT COUNT(*) AS TreatmentCount  
FROM Treatments GROUP BY DoctorID) AS AvgTreatments);
```

*Reason: To find doctors who have treated more patients than the average number.*

**3. Rooms Occupied by More Than One Patient:**

```
SELECT r.RoomNumber  
FROM Rooms r  
WHERE (SELECT COUNT(*) FROM PatientRooms pr WHERE pr.RoomID =  
r.RoomID) > 1;
```

*Reason: To find rooms that have been assigned to more than one patient.*

**4. Patients with More Prescriptions than Average:**

```
SELECT p.Name FROM Patients p WHERE (SELECT COUNT(*) FROM Prescriptions  
pr JOIN Treatments t ON pr.TreatmentID = t.TreatmentID WHERE t.PatientID =  
p.PatientID) > (SELECT AVG(PrescriptionCount) FROM (SELECT COUNT(*) AS  
PrescriptionCount FROM Prescriptions GROUP BY TreatmentID) AS  
AvgPrescriptions);
```

*Reason: To identify patients with more prescriptions than the average.*

**5. Doctors with More Treatments than the Average:**

```
SELECT d.Name  
FROM Doctors d  
WHERE (SELECT COUNT(*) FROM Treatments t WHERE t.DoctorID = d.DoctorID)  
>
```



(SELECT AVG(TreatmentCount) FROM (SELECT COUNT(\*) AS TreatmentCount FROM Treatments GROUP BY DoctorID) AS AvgTreatments);

Reason: To identify doctors who have administered more treatments than the average.

**Proofs:**

**Tables:**

Tables_in_hospital
appointments
doctors
medications
nurses
patientrooms
patients
prescriptions
rooms
treatments

	Name	AppointmentDate	Reason
▶	John Doe	2024-07-10	Routine check-up
	Jane Smith	2024-07-11	Neurological evaluation
	Alice Brown	2024-07-12	Orthopedic consultation
	Bob White	2024-07-13	Pediatric follow-up
	Ann Green	2024-07-14	Oncology screening
	Tom Black	2024-07-15	Skin rash evaluation
	Harry Potter	2024-07-16	Cardiology consultation
	Hermione Granger	2024-07-17	Neurology follow-up
	Ron Weasley	2024-07-18	Orthopedic follow-up
	Draco Malfoy	2024-07-19	Pediatric examination

	Name	Specialty	AppointmentDate
▶	Dr. Alice Brown	Cardiology	2024-07-10
	Dr. Bob White	Neurology	2024-07-11
	Dr. Ann Green	Orthopedics	2024-07-12
	Dr. Tom Black	Pediatrics	2024-07-13
	Dr. John Doe	Oncology	2024-07-14
	Dr. Jane Smith	Dermatology	2024-07-15
	Dr. Richard Roe	Cardiology	2024-07-16
	Dr. Mary Major	Neurology	2024-07-17
	Dr. Henry Jones	Orthopedics	2024-07-18
	Dr. Linda Lane	Pediatrics	2024-07-19

	Name	TreatmentDate	Description
▶	John Doe	2024-07-10	Initial evaluation and treatment plan
	Jane Smith	2024-07-11	Diagnosis and follow-up plan
	Alice Brown	2024-07-12	Consultation and treatment options discussed
	Bob White	2024-07-13	Examination and treatment recommendations
	Ann Green	2024-07-14	Oncology screening and treatment strategy
	Tom Black	2024-07-15	Skin condition assessment and treatment
	Harry Potter	2024-07-16	Heart condition evaluation and treatment
	Hermione Granger	2024-07-17	Neurological assessment and treatment
	Ron Weasley	2024-07-18	Follow-up and orthopedic care
	Draco Malfoy	2024-07-19	Pediatric assessment and treatment

Neurologi

Result GridFilter Rows:Export:Wrap Cell Content:

⌵

	Name	TreatmentDate	Description
▶	Dr. Alice Brown	2024-07-10	Initial evaluation and treatment plan
	Dr. Bob White	2024-07-11	Diagnosis and follow-up plan
	Dr. Ann Green	2024-07-12	Consultation and treatment
	Dr. Tom Black	2024-07-13	Examination and treatment recommendations
	Dr. John Doe	2024-07-14	Oncology screening and treatment strategy
	Dr. Jane Smith	2024-07-15	Skin condition assessment and treatment
	Dr. Richard Roe	2024-07-16	Heart condition evaluation and treatment
	Dr. Mary Major	2024-07-17	Neurological assessment and treatment
	Dr. Henry Jones	2024-07-18	Follow-up and orthopedic care
	Dr. Linda Lane	2024-07-19	Pediatric assessment and treatment

Consultation and treatment opti

	Description	Name	Dosage
▶	Initial evaluation and treatment plan	Aspirin	100mg
	Diagnosis and follow-up plan	Ibuprofen	200mg
	Consultation and treatment options discussed	Acetaminophen	500mg
	Examination and treatment recommendations	Metformin	500mg
	Oncology screening and treatment strategy	Amoxicillin	250mg
	Skin condition assessment and treatment	Lisinopril	10mg
	Heart condition evaluation and treatment	Simvastatin	20mg
	Neurological assessment and treatment	Omeprazole	20mg
	Follow-up and orthopedic care	Losartan	50mg
	Pediatric assessment and treatment	Hydrochlorothiazide	25mg

	Name	RoomNumber	RoomType
▶	John Doe	101	Single
	Jane Smith	102	Double
	Alice Brown	103	Suite
	Bob White	104	ICU
	Ann Green	105	Single
	Tom Black	106	Double
	Harry Potter	107	Suite
	Hermione Granger	108	ICU
	Ron Weasley	109	Single
	Draco Malfoy	110	Double

Result 9

	PatientName	MedicationName	Dosage	Frequency
▶	John Doe	Aspirin	100mg	Once daily
	Jane Smith	Ibuprofen	200mg	Twice daily
	Alice Brown	Acetaminophen	500mg	Once daily
	Bob White	Metformin	500mg	Twice daily
	Ann Green	Amoxicillin	250mg	Three times daily
	Tom Black	Lisinopril	10mg	Once daily
	Harry Potter	Simvastatin	20mg	Once daily
	Hermione Granger	Omeprazole	20mg	Once daily
	Ron Weasley	Losartan	50mg	Once daily
	Draco Malfoy	Hydrochlorothiazide	25mg	Once daily

## Conclusion

The Hospital Management System (HMS) project effectively enhances hospital operations by integrating patient management, staff details, appointments, treatments, prescriptions, and room allocations. Key achievements include:

- **Efficient Patient Management:** Comprehensive records and quick access to patient information.
- **Streamlined Appointment Scheduling:** Simplified and error-reduced scheduling processes.
- **Accurate Treatment and Prescription Management:** Ensures correct medications and follow-up care.
- **Effective Resource Allocation:** Optimized room availability and allocation.
- **Comprehensive Staff Management:** Detailed profiles for better coordination.
- **Robust Reporting:** Valuable insights through various reports.

Technical highlights include the use of join queries, aggregate functions, and correlated queries to provide comprehensive data integration, summarization, and complex data retrieval. The HMS improves operational efficiency and patient care, demonstrating the transformative power of database management in healthcare.