Group-9
Deepshikha Dhammi – 30140157
Roshan Patel – 30184010

FINAL REPORT

HealthHub is a Hospital Database Management System developed to help hospitals overcome their day-to-day challenges: scattered patient records, difficulties in appointment scheduling, and billing and resource management issues. These usually delay treatment for patients, increase wait times, and raise extra workload for the staff. HealthHub integrates one place for everything, from maintaining EHRs to appointment scheduling, billing, insurance processing, and resource management. HealthHub reduces mistakes, helps with better decision-making, and lets hospitals handle both routine and emergencies faster by keeping all patient and hospital data in one place. This would let doctors find out all information about a patient in an instant, administrators monitor resources, and accounting staff process payments without problems. Designed with health regulations in mind, HealthHub ensures that data is kept secure, and reliable, and has improved the working way of a hospital. With automation, real-time updates, and safe technology combined, HealthHub creates more efficiency in managing a hospital, focusing on what matters most: excellent patient care. This all-inclusive system reduces errors, saves time, and assists hospitals in providing better care.

Introduction: Managing patient data is one big challenge for hospitals. Many times, the information is scattered at every other place, and coordinating between doctors, nurses, and staff is a problem. This leads to delays in admitting patients, longer waiting queues, and errors in billing. Even emergency services can be delayed due to improper communication and deployment of resources. To add to that, hospitals are struggling with how to handle old systems as they work towards adopting new technologies that would best facilitate their care.

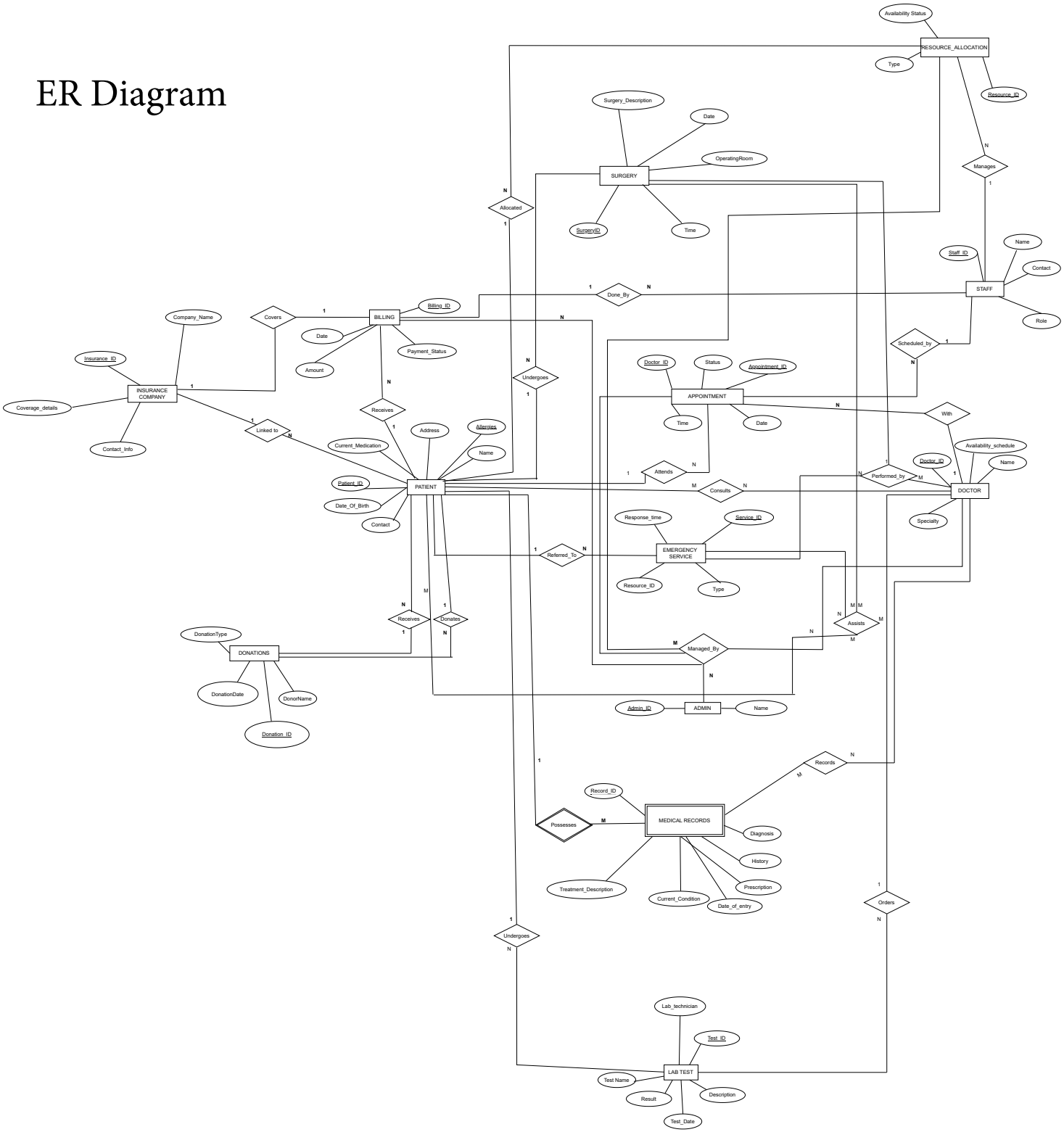
This is quite serious, as healthcare is something everybody depends on deeply. When hospitals are not able to manage patient data, it directly influences the quality of treatment patients receive. These could be the reasons that make healthcare more affordable, faster, and better for all. Our project, HealthHub, is a hospital management system meant to solve these challenges. It creates one platform where all the data about a hospital is stored and controlled in one place. With HealthHub, hospitals can efficiently manage patient records, handle appointment scheduling, manage billing, and do resource allocation.

Project Design:

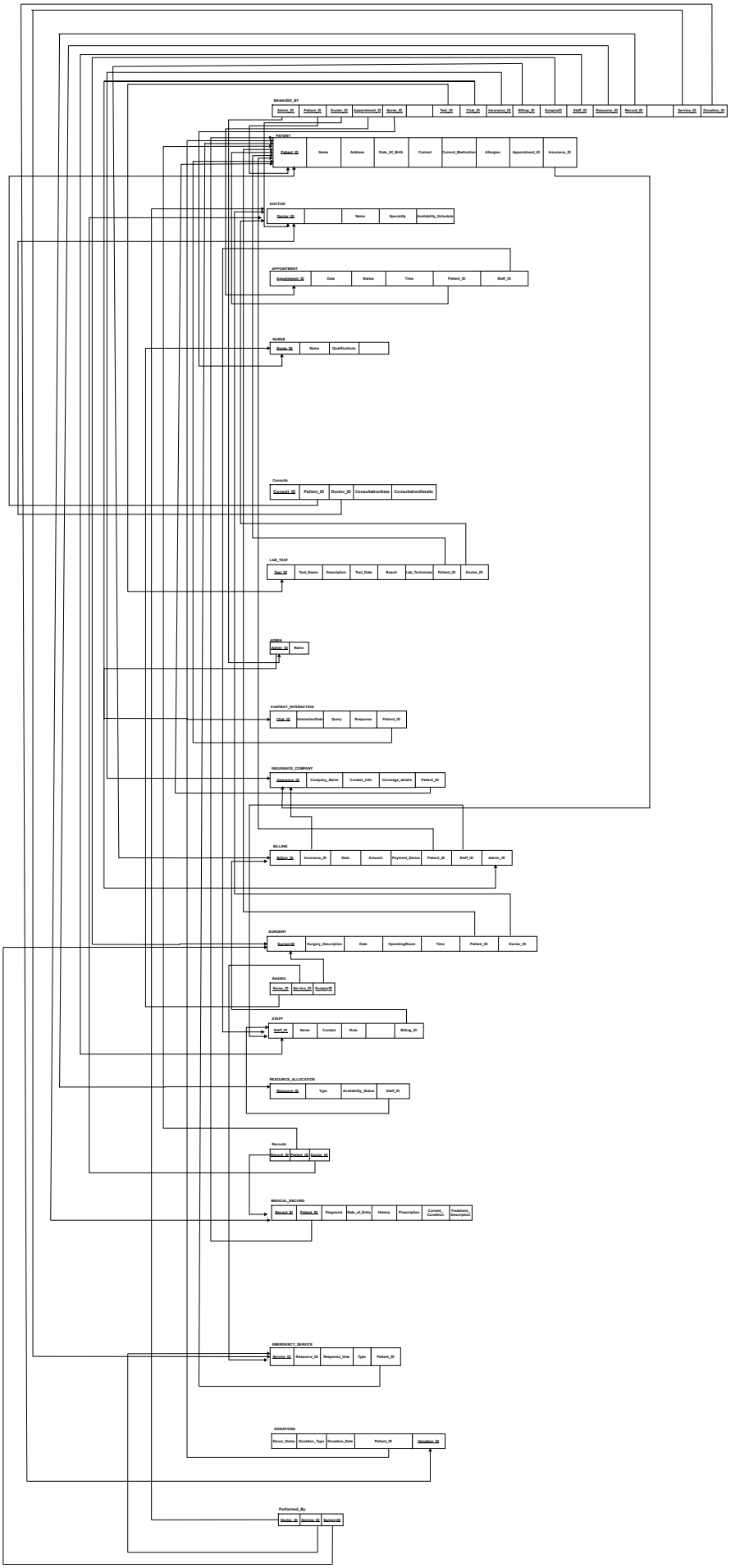
1. Patient Registration:
    - A patient enters their personal information on the website, creating a profile.
    - This information is stored securely in the database under the “Patient Information” module.

2. Appointment Booking:
    - The patient selects a doctor using the “Doctors Information” section.
    - The system checks the doctor’s availability and confirms the appointment.
3. Doctor Interaction:
    - On the scheduled date, the doctor logs in to view the patient’s records and updates medical history, diagnoses, and treatment plans.
4. Lab Tests and Results:
    - If required, the doctor schedules a lab test.
    - Lab technicians update the system with test results, which are accessible to both the doctor and patient.
5. Billing and Insurance Claims:
    - Once the consultation or tests are completed, a bill is generated.
    - If the patient has insurance, the claim is processed directly from the system.
6. Resource Management and Emergencies:
    - During surgeries or emergencies, resources like operating rooms and staff are allocated using the system.
    - Logs are maintained for auditing and tracking purposes.
7. Donation Management:
    - If a patient or donor logs a donation, the system records the details and matches it to a recipient if necessary.

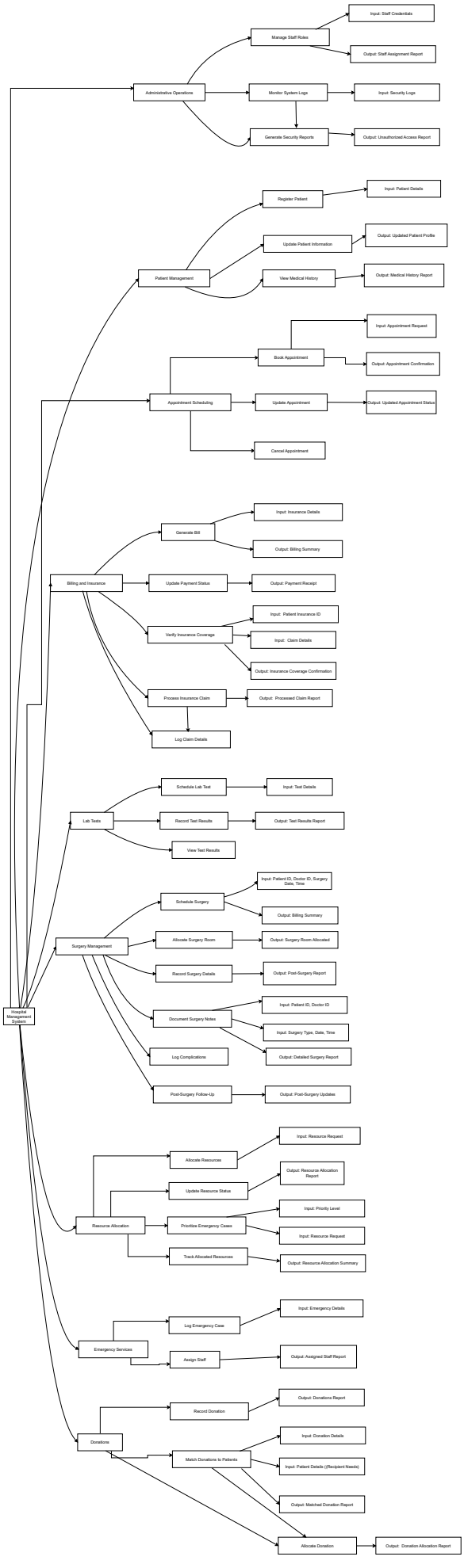
ER Diagram


Availability Status
RESOURCE_ALLOCATION
Type
Resource_ID
Surgery_Description
Date
OperatingRoom
SURGERY
Manages
Allocated
SurgeryID
Time
Staff_ID
Name
Contact
STAFF
Role
Done_By
Billing_ID
Company_Name
Covers
BILLING
Date
Amount
Payment_Status
Scheduled_by
Insurance_ID
INSURANCE COMPANY
Coverage_details
Contact_Info
Undergoes
Doctor_ID
Status
Appointment_ID
APPOINTMENT
Time
Date
With
Receives
Linked to
Address
Allergies
Name
Current_Medication
Availability_schedule
Doctor_ID
Name
Performed_by
DOCTOR
Specialty
Attends
Consults
Patient_ID
Date_Of_Birth
Contact
PATIENT
Response_time
Service_ID
Referred_To
EMERGENCY SERVICE
Resource_ID
Type
Receives
Donates
Assists
DonationType
DONATIONS
DonationDate
DonorName
Donation_ID
Managed_By
Admin_ID
ADMIN
Name
Records
Record_ID
Possesses
MEDICAL RECORDS
Diagnosis
History
Prescription
Date_of_entry
Current_Condition
Treatment_Description
Orders
Undergoes
Lab_technician
Test_ID
LAB TEST
Test Name
Result
Description
Test_Date

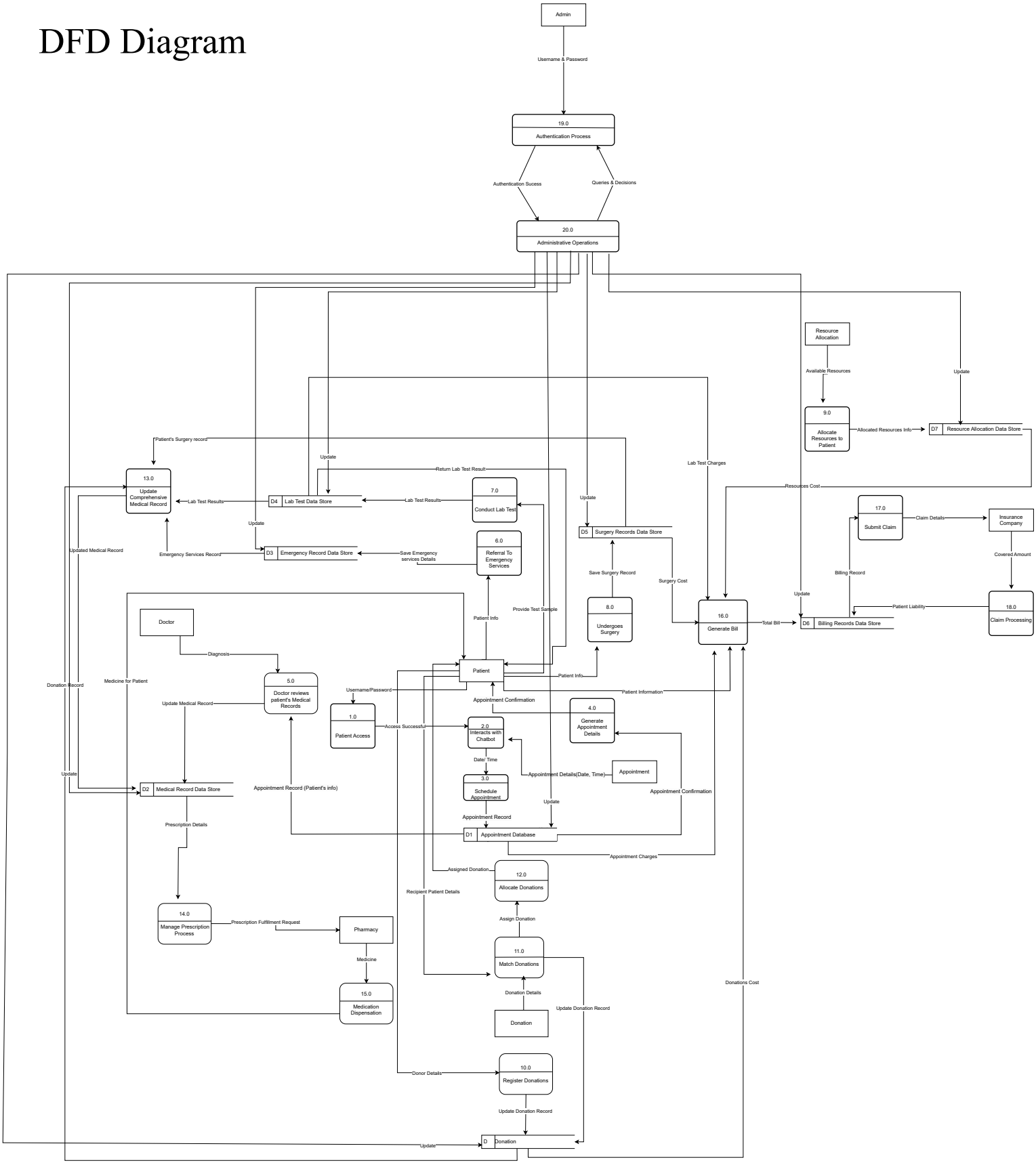
RM Diagram


MANAGED_BY
Admin_ID
Patient_ID
Doctor_ID
Appointment_ID
Nurse_ID
Test_ID
Chat_ID
Insurance_IN
Billing_ID
SurgeryID
Staff_ID
Resource_ID
Record_ID
Service_ID
Donation_ID
PATIENT
Patient_ID
Name
Address
Date_Of_Birth
Contact
Current_Medication
Allergies
Appointment_ID
Insurance_ID
DOCTOR
Doctor_ID
Name
Specialty
Availability_Schedule
APPOINTMENT
Appointment_ID
Date
Status
Time
Patient_ID
Staff_ID
NURSE
Nurse_ID
Name
Qualifications
Consults
Consult_ID
Patient_ID
Doctor_ID
ConsultationDate
ConsultationDetails
LAB_TEST
Test_ID
Test_Name
Description
Test_Date
Result
Lab_Technician
Patient_ID
Doctor_ID
ADMIN
Admin_ID
Name
CHATBOT_INTERACTION
Chat_ID
InteractionDate
Query
Response
Patient_ID
INSURANCE_COMPANY
Insurance_ID
Company_Name
Contact_Info
Coverage_details
Patient_ID
BILLING
Billing_ID
Insurance_ID
Date
Amount
Payment_Status
Patient_ID
Staff_ID
Admin_ID
SURGERY
SurgeryID
Surgery_Description
Date
OperatingRoom
Time
Patient_ID
Doctor_ID
Assists
Nurse_ID
Service_ID
SurgeryID
STAFF
Staff_ID
Name
Contact
Role
Billing_ID
RESOURCE_ALLOCATION
Resource_ID
Type
Availability_Status
Staff_ID
Records
Record_ID
Patient_ID
Doctor_ID
MEDICAL_RECORD
Record_ID
Patient_ID
Diagnosis
Date_of_Entry
History
Prescription
Current_Condition
Treatment_Description
EMERGENCY_SERVICE
Service_ID
Resource_ID
Response_time
Type
Patient_ID
DONATIONS
Donor_Name
Donation_Type
Donation_Date
Patient_ID
Donation_ID
Performed_By
Doctor_ID
Service_ID
SurgeryID

HIPO Diagram



DFD Diagram


Admin
Username & Password
19.0
Authentication Process
Authentication Sucess
Queries & Decisions
20.0
Administrative Operations
Resource Allocation
Available Resources
Update
9.0
Allocate Resources to Patient
Allocated Resources Info
D7 Resource Allocation Data Store
Patient's Surgery record
Update
Return Lab Test Result
Lab Test Charges
Resources Cost
13.0
Update Comprehensive Medical Record
Lab Test Results
D4 Lab Test Data Store
Lab Test Results
7.0
Conduct Lab Test
Update
17.0
Submit Claim
Claim Details
Insurance Company
Update
D5 Surgery Records Data Store
Updated Medical Record
Emergency Services Record
D3 Emergency Record Data Store
Save Emergency services Details
6.0
Referral To Emergency Services
Covered Amount
Billing Record
Save Surgery Record
Surgery Cost
Update
Patient Liability
18.0
Claim Processing
Doctor
Provide Test Sample
Patient Info
8.0
Undergoes Surgery
16.0
Generate Bill
Total Bill
D6 Billing Records Data Store
Diagnosis
Patient
Donation Record
Medicine for Patient
5.0
Doctor reviews patient's Medical Records
Username/Password
Patient Info
Patient Information
Update Medical Record
1.0
Patient Access
Access Successful
2.0
Interacts with Chatbot
Appointment Confirmation
4.0
Generate Appointment Details
Update
Date/ Time
Appointment Details(Date, Time)
Appointment
3.0
Schedule Appointment
Appointment Confirmation
D2 Medical Record Data Store
Appointment Record (Patient's info)
Appointment Record
D1 Appointment Database
Prescription Details
Appointment Charges
Assigned Donation
12.0
Allocate Donations
Recipient Patient Details
14.0
Manage Prescription Process
Prescription Fulfillment Request
Pharmacy
Assign Donation
11.0
Match Donations
Medicine
15.0
Medication Dispensation
Donation Details
Donations Cost
Update Donation Record
Donation
Donor Details
10.0
Register Donations
Update Donation Record
D Donation
Update

SQL QUERIES

Lab Test:

```
CREATE TABLE Lab_Tests (
    TestID INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY, -- Auto-incrementing Test ID
    PatientID INT NOT NULL,                              -- Reference to Patients table
    DoctorID INT NOT NULL,                               -- Reference to Doctors table
    TestType VARCHAR(255) NOT NULL,                     -- Type of test (e.g., Blood Test, X-Ray)
    TestDate DATE DEFAULT SYSDATE,                      -- Date of the test
    Results CLOB,                                        -- Test results
    Status VARCHAR(50) DEFAULT 'Pending',               -- Test status (Pending/Completed)
    FOREIGN KEY (PatientID) REFERENCES Patients(PATIENT_ID),
    FOREIGN KEY (DoctorID) REFERENCES Doctors(DOCTOR_ID)
);
```

Emergency Service:

```
SELECT
    es.Service_ID,
    p.Name AS Patient_Name,
    p.Contact AS Patient_Contact,
    es.Type AS Emergency_Type,
    d.Name AS Doctor_Name,
    r.Type AS Resource_Type,
    es.Response_Time,
    es.Date_of_Entry
FROM
    YOUR_SCHEMA.EMERGENCY_SERVICE es
JOIN
    YOUR_SCHEMA.PATIENT p ON es.Patient_ID = p.Patient_ID
JOIN
    YOUR_SCHEMA.DOCTOR d ON es.Doctor_ID = d.Doctor_ID
JOIN
    YOUR_SCHEMA.RESOURCE_ALLOCATION r ON es.Resource_ID = r.Resource_ID
WHERE
    es.Date_of_Entry >= SYSDATE - 30
ORDER BY
    es.Date_of_Entry DESC;
```

Medical Record:

```
CREATE TABLE Medical_Records (
    PatientID INT NOT NULL,             -- Reference to the patient
    DoctorID INT NOT NULL,              -- Reference to the doctor
    Diagnosis VARCHAR(255),             -- Diagnosis information
    Treatment CLOB,                     -- Details of the treatment
    Date_of_Record DATE DEFAULT SYSDATE, -- Automatically use the current date
    Notes CLOB,                         -- Additional notes (optional)
    Blood_Pressure VARCHAR(10),         -- Example: 120/80
    Heart_Rate INT,                     -- Heartbeats per minute
    Temperature DECIMAL(5,2),           -- Body temperature (e.g., 98.6)
    Oxygen_Saturation DECIMAL(5,2),     -- Oxygen level in percentage
    PRIMARY KEY (PatientID, Date_of_Record) -- Ensures one record per patient per day
);
```

Log Donation:

```
CREATE TABLE DONATIONS (
    DONATION_ID NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY, -- Unique ID for each
    DONOR_NAME VARCHAR2(100) NOT NULL,                               -- Name of the donor
    DONATION_TYPE VARCHAR2(50) NOT NULL,                             -- Type of donation (e.g., Blood, Money, Equipment)
    DONATION_DATE DATE DEFAULT SYSDATE,                              -- Date of the donation
    PATIENT_ID NUMBER,                                               -- Optional: Patient ID for specific patient-related donations
    AMOUNT NUMBER                                                   -- Amount donated (for monetary donations)
);
```

Billing:

```
CREATE TABLE Billing (
    Billing_ID NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY, -- Unique Billing ID
    Patient_ID NUMBER NOT NULL, -- Foreign key to Patient table
    Service_Type VARCHAR(50) NOT NULL, -- Foreign key to Charges table
    Insurance_ID NUMBER, -- Foreign key to Insurance_Company table
    Billing_Date DATE DEFAULT SYSDATE, -- Date of the billing record
    Amount NUMBER NOT NULL, -- Amount associated with the billing
    Payment_Status VARCHAR2(20) DEFAULT 'Pending', -- Payment status (e.g., Pending, Paid)
    CONSTRAINT FK_Patient_ID FOREIGN KEY (Patient_ID) REFERENCES Patient (Patient_ID), -- Links to Patient table
    CONSTRAINT FK_Service_Type FOREIGN KEY (Service_Type) REFERENCES Charges (Service_Type), -- Links to Charges table
    CONSTRAINT FK_Insurance_ID FOREIGN KEY (Insurance_ID) REFERENCES Insurance_Company (Insurance_ID) -- Links to Insurance_Company table
);
```

Insurance Company:

```
CREATE TABLE Insurance_Company (
    Insurance_ID NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY, -- Unique identifier for insurance company
    Company_Name VARCHAR2(100) NOT NULL, -- Name of the insurance company
    Coverage_Amount NUMBER(10, 2) NOT NULL, -- Coverage amount provided by the insurance
    Contact_Info VARCHAR2(150) -- Contact information (phone/email)
);
```

Surgery:

```
CREATE TABLE Surgery (
     Patient_ID INT NOT NULL, -- Foreign key referencing Patient table
    Doctor_ID INT NOT NULL, -- Foreign key referencing Doctor table
    Surgery_ID INT GENERATED ALWAYS AS IDENTITY PRIMARY KEY, -- Unique identifier for the surgery
    Surgery_Description VARCHAR(255) NOT NULL, -- Description of the surgery

    Surgery_Date DATE NOT NULL, -- Date of the surgery
    Surgery_Time TIMESTAMP NOT NULL, -- Date and time of the surgery
    Operating_Room VARCHAR(50) NOT NULL, -- Operating room number or name

    CONSTRAINT FK_Patient FOREIGN KEY (Patient_ID) REFERENCES Patient(Patient_ID),
    CONSTRAINT FK_Doctor FOREIGN KEY (Doctor_ID) REFERENCES DOCTORS(DOCTOR_ID)
);
```

Appointments:

```
CREATE TABLE APPOINTMENTS (
    APPOINTMENT_ID NUMBER PRIMARY KEY,
    PATIENT_ID NUMBER,
    DOCTOR_ID NUMBER,
    APPOINTMENT_DATE DATE,
    APPOINTMENT_TIME VARCHAR2(50),
    STATUS VARCHAR2(50),
    FOREIGN KEY (PATIENT_ID) REFERENCES PATIENTS (PATIENT_ID),
    FOREIGN KEY (DOCTOR_ID) REFERENCES DOCTORS (DOCTOR_ID)
);
```

Patient:

```
CREATE TABLE Patient (
    Patient_ID NUMBER PRIMARY KEY,          -- Primary Key for identifying patients
    Patient_Name VARCHAR2(255) NOT NULL,     -- Patient's name, not null
    Date_of_Birth DATE NOT NULL,             -- Date of birth, not null
    Contact_Number VARCHAR2(15),             -- Contact number, can be null
    Email VARCHAR2(255),                     -- Email address, can be null
    Address VARCHAR2(500)                    -- Address, can be null
);
```

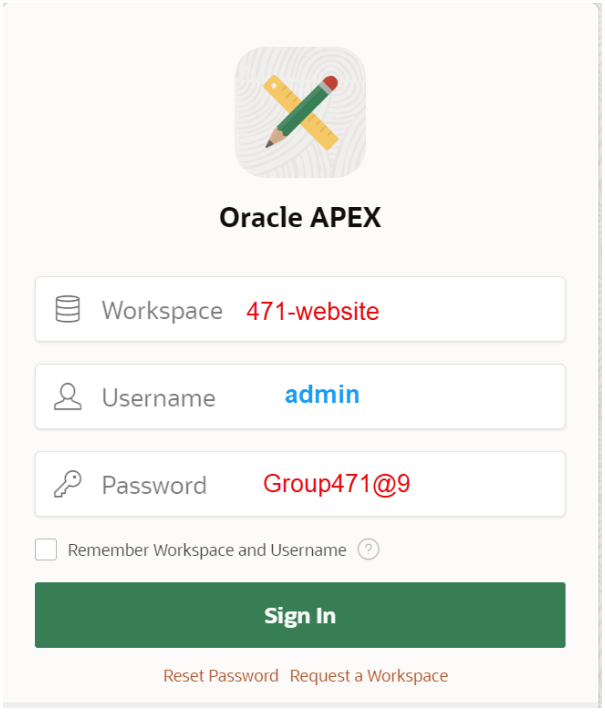
Doctor:

```
CREATE TABLE DOCTORS (
    DOCTOR_ID NUMBER PRIMARY KEY,
    NAME VARCHAR2(100) NOT NULL,
    SPECIALTY VARCHAR2(100),
    AVAILABILITY_SCHEDULE VARCHAR2(255)
);
INSERT INTO DOCTORS (DOCTOR_ID, NAME, SPECIALTY, AVAILABILITY_SCHEDULE)
VALUES (1, 'Dr. John Smith', 'Cardiologist', 'Mon-Fri 9:00 AM - 5:00 PM');

INSERT INTO DOCTORS (DOCTOR_ID, NAME, SPECIALTY, AVAILABILITY_SCHEDULE)
VALUES (2, 'Dr. Emily Davis', 'Dermatologist', 'Mon, Wed, Fri 10:00 AM - 2:00 PM');

INSERT INTO DOCTORS (DOCTOR_ID, NAME, SPECIALTY, AVAILABILITY_SCHEDULE)
VALUES (3, 'Dr. Robert Lee', 'Orthopedic Surgeon', 'Tue-Thu 8:00 AM - 12:00 PM');
```

HEALTH HUB

Manual

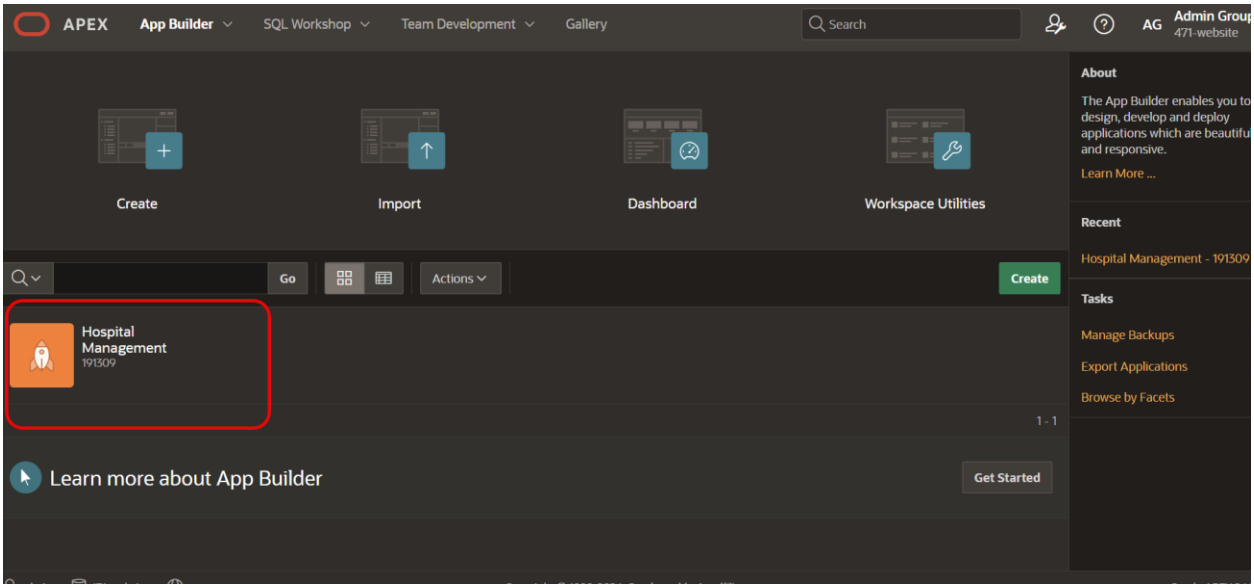
Steps to Start Using Oracle APEX Starting from the Login Page:

1. **Access the Login Page:**
    - Open your web browser and navigate to the Oracle APEX login URL provided by your administrator (or hosted service).
    - You should see the login page similar to the one in the uploaded screenshot.
    - 

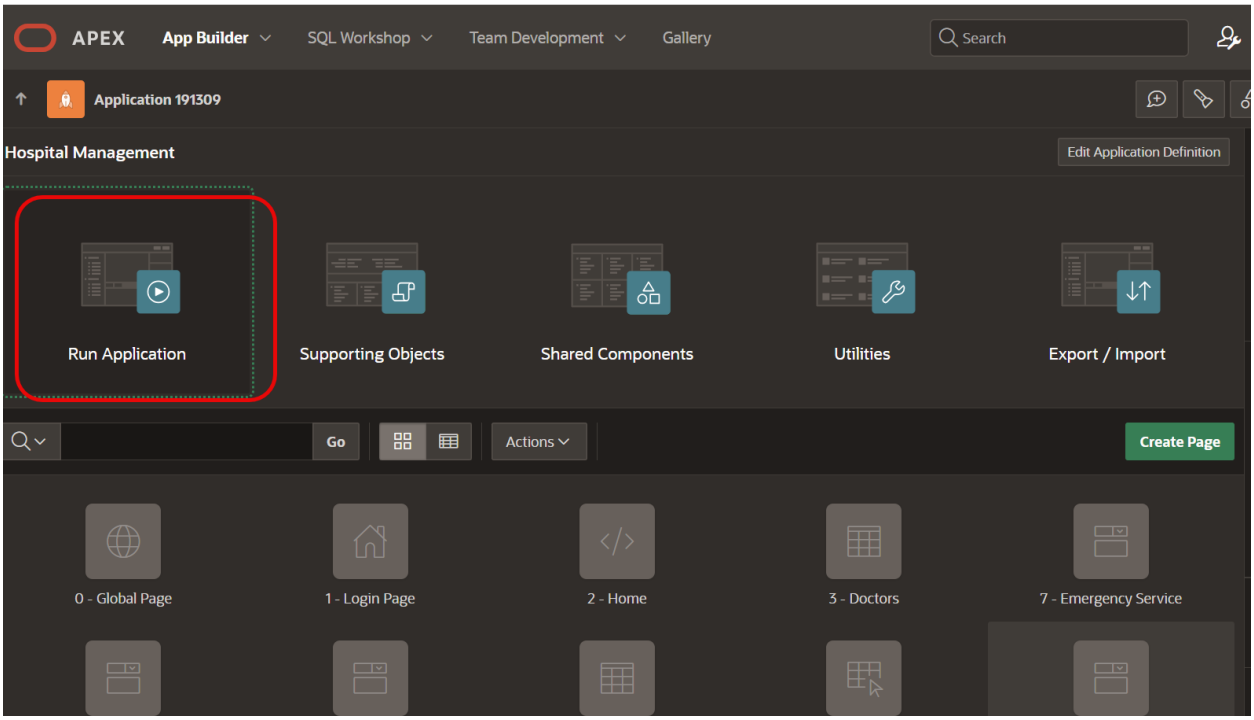
2. **Enter Workspace Information:**
    - Locate the **Workspace** field.
    - Enter the workspace name provided to you, in this case, 471-website.
3. **Enter Your Username:**
    - In the **Username** field, enter your username. For example, admin as shown.
4. **Enter Your Password:**
    - In the **Password** field, enter the password provided to you. For example, Group471@9 as displayed.
5. **Optional: Remember Workspace and Username:**
    - Check the box labeled **"Remember Workspace and Username"** if you are using a secure, personal device and want the browser to save your credentials.
6. **Click "Sign In":**

- Press the **Sign In** button. If the credentials are correct, you will be logged into Oracle APEX

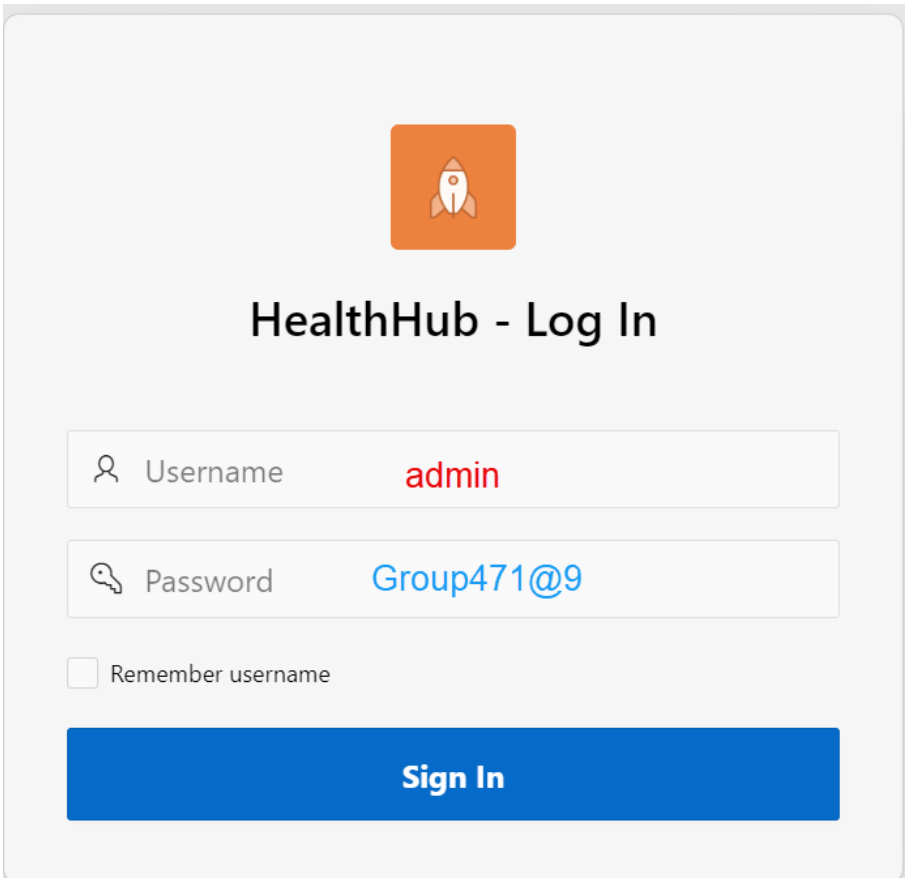
7. **Click on the 'Hospital Management' icon to open the application.**



8. Click on 'Run Application' to launch and preview the Hospital Management application



9. Enter the username admin and the password Group471@9, then click 'Sign In' to access the HealthHub application



10. After logging in, you will be redirected to the HealthHub dashboard.

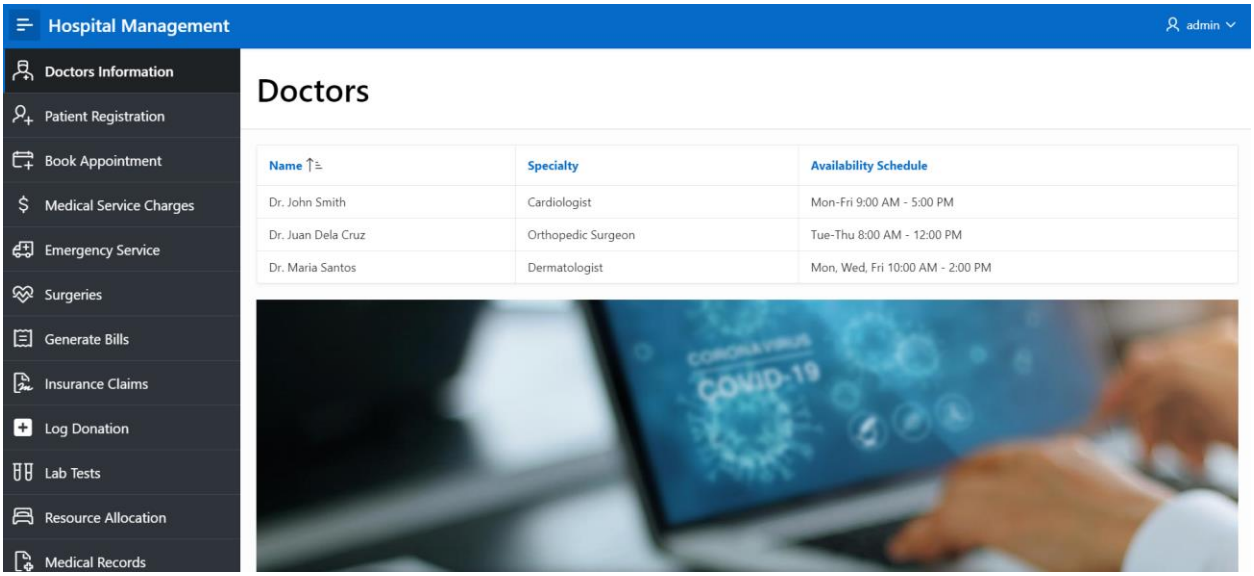
11. Click on the menu icon on the top left corner of the screen to expand the navigation menu, and then select the desired module, such as 'Book Appointment' or 'Lab Tests,' to proceed with its functionality.

Doctor’s Information

1. **Select "Doctors Information":**
   From the expanded menu, click on **"Doctors Information"** (first option in the list).

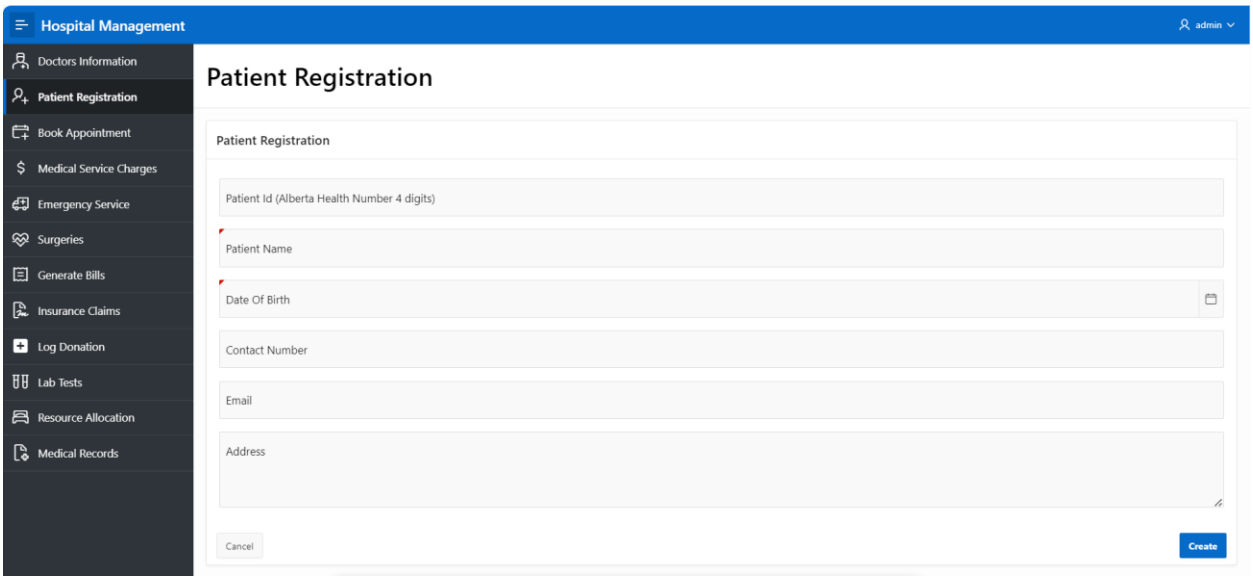
   View Doctors Details:

   You will be redirected to the **Doctors Information** page, where you can:
   View a list of all doctors.
   Search for a doctor by name, specialty, or department.

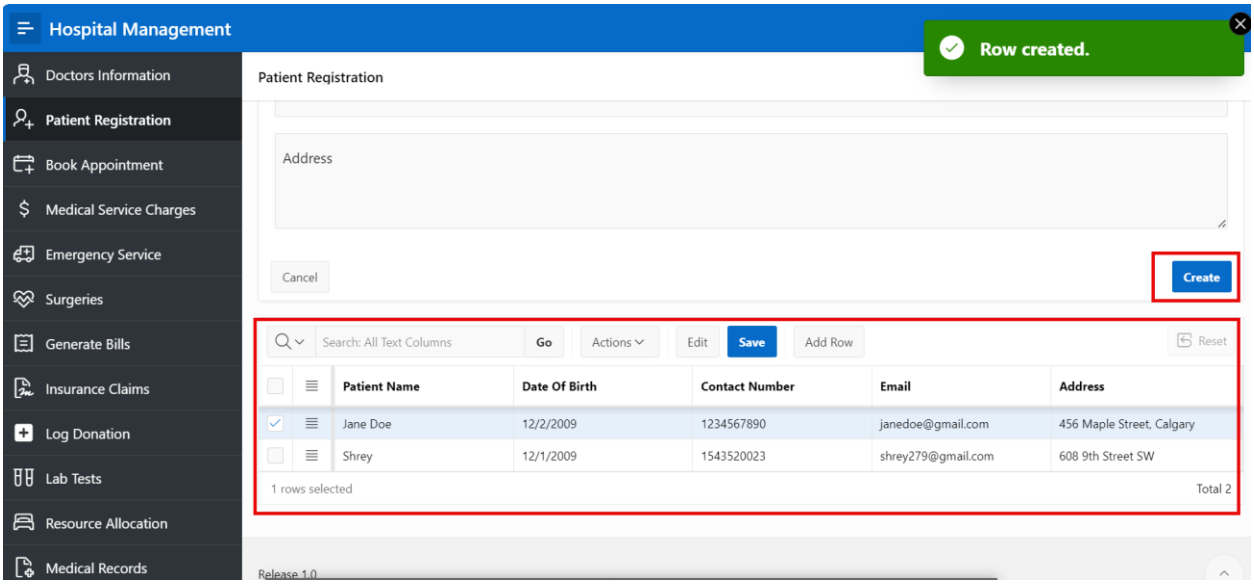


PATIENT REGISTRATION

1. On the 'Patient Registration' page, you can register new patients by filling in the required details such as Patient ID (Alberta Health Number, 4 digits), Name, Date of Birth, Contact Number, Email, and Address. Once all fields are completed, click the 'Create' button to save the patient's information.

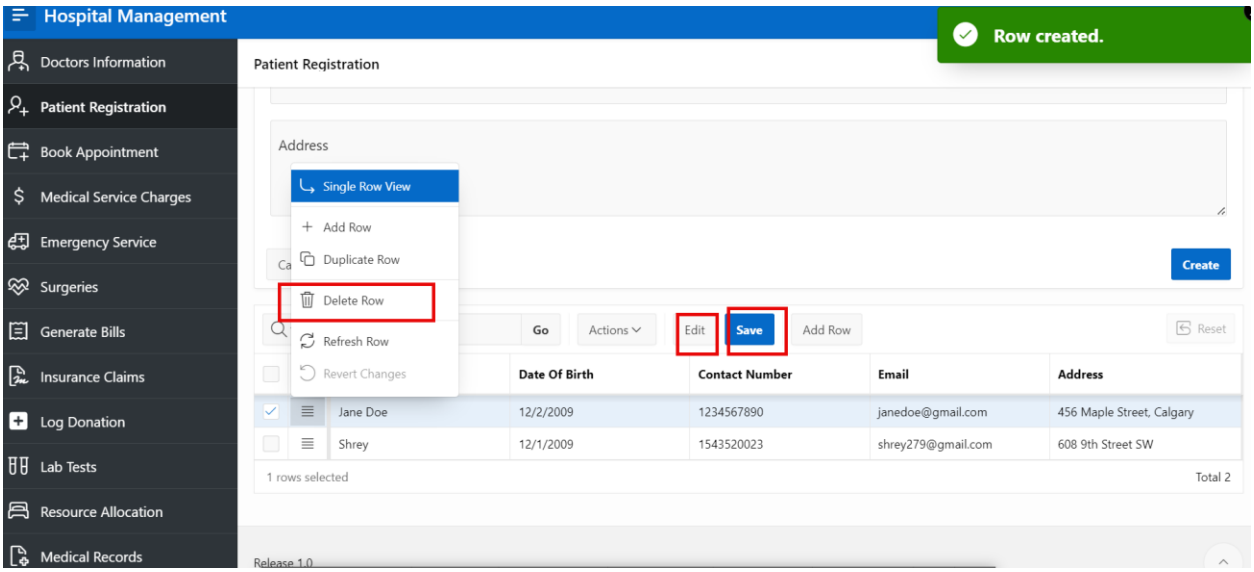


2. After filling out the Patient Registration form, click the 'Create' button to save the patient's details. The newly added patient will appear in the table below, along with their name, date of birth, contact number, email, and address. You can use the table to view, search, edit, or manage patient records.



To manage patient records:

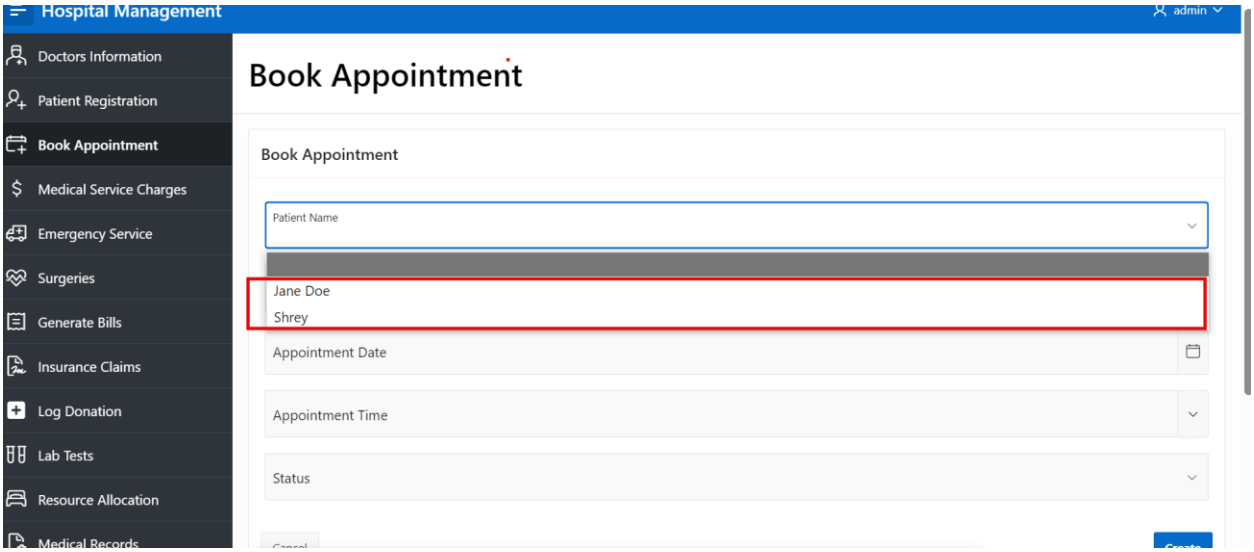
- **Edit a Record:** Select a patient row, click the **'Edit'** button, make the necessary changes in the fields, and then click **'Save'** to update the information.
- **Delete a Record:** Use the **'Actions'** menu, choose **'Delete Row'**, and confirm to remove the selected patient's record from the table

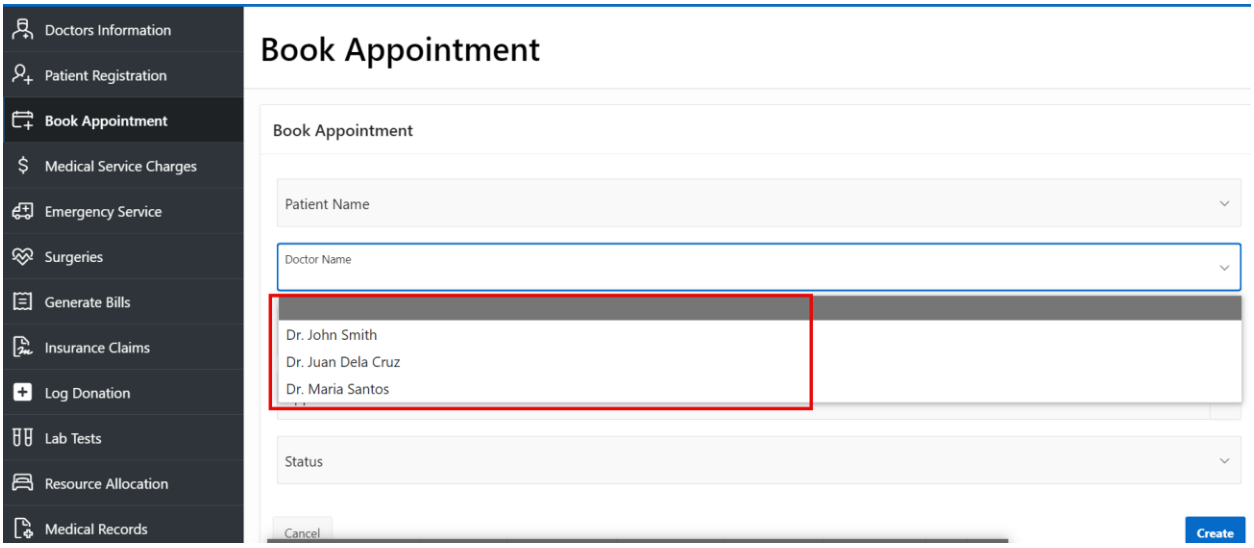


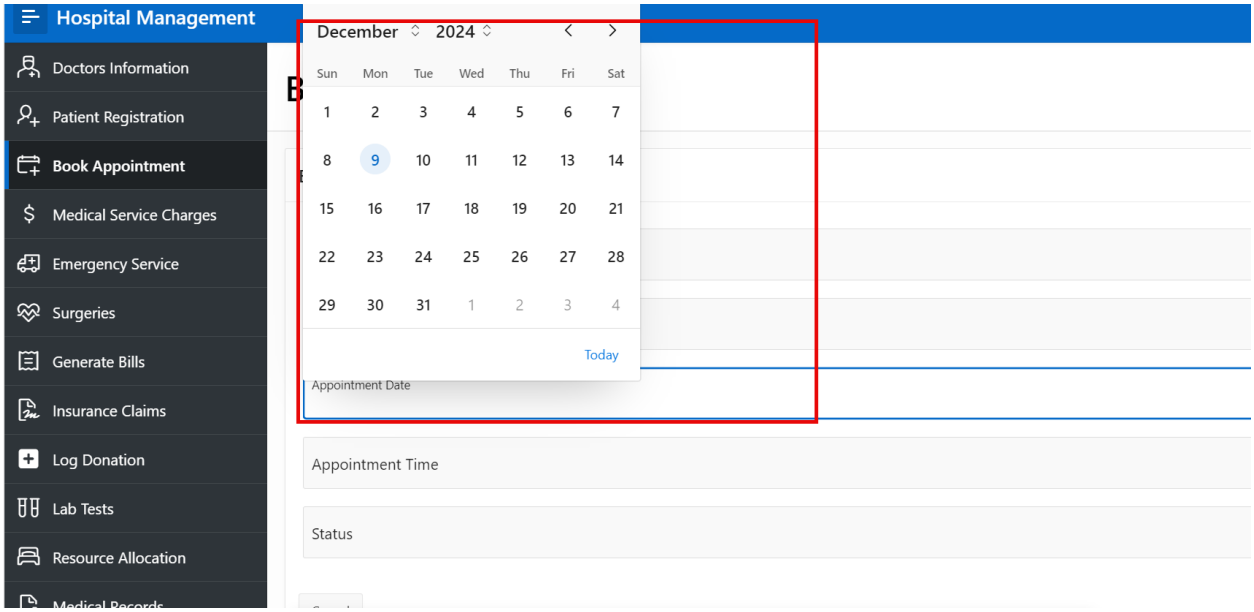
Book an Appointment:

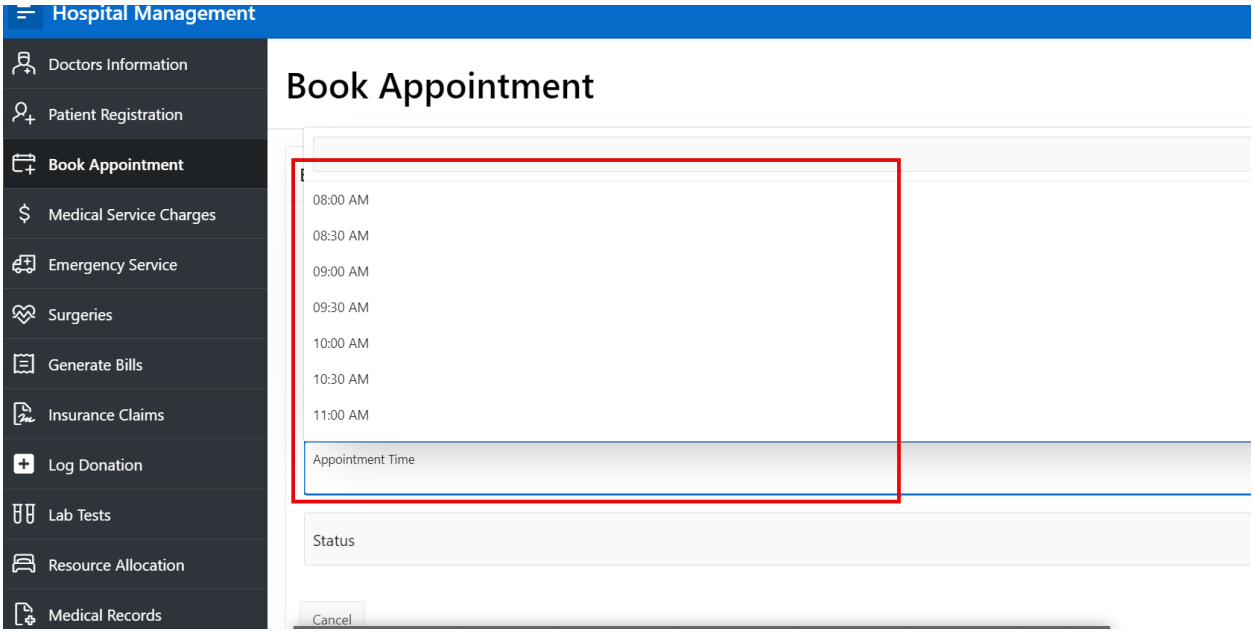
1. **Select Patient Name:**
    - Open the **Patient Name** dropdown.
    - Choose the patient for the appointment, such as:
        - **Jane Doe**
        - **Shrey**
2. **Select Doctor Name:**
    - Open the **Doctor Name** dropdown.
    - Choose the doctor you want to book an appointment with:
        - **Dr. John Smith (Cardiologist)**
        - **Dr. Juan Dela Cruz (Orthopedic Surgeon)**
        - **Dr. Maria Santos (Dermatologist)**
3. **Choose Appointment Date:**
    - Click on the **Appointment Date** field to open the calendar.
    - Select the desired date for the appointment.
4. **Choose Appointment Time:**
    - Open the **Appointment Time** dropdown.
    - Pick a time slot that works for the appointment, such as:
        - **08:00 AM**
        - **08:30 AM**
        - **09:00 AM**
        - **09:30 AM**
        - **10:00 AM**
5. **Set Appointment Status:**
    - Open the **Status** dropdown.
    - Choose the status of the appointment:
        - **Confirmed**: If the appointment is finalized.
        - **Pending**: If the appointment requires confirmation.

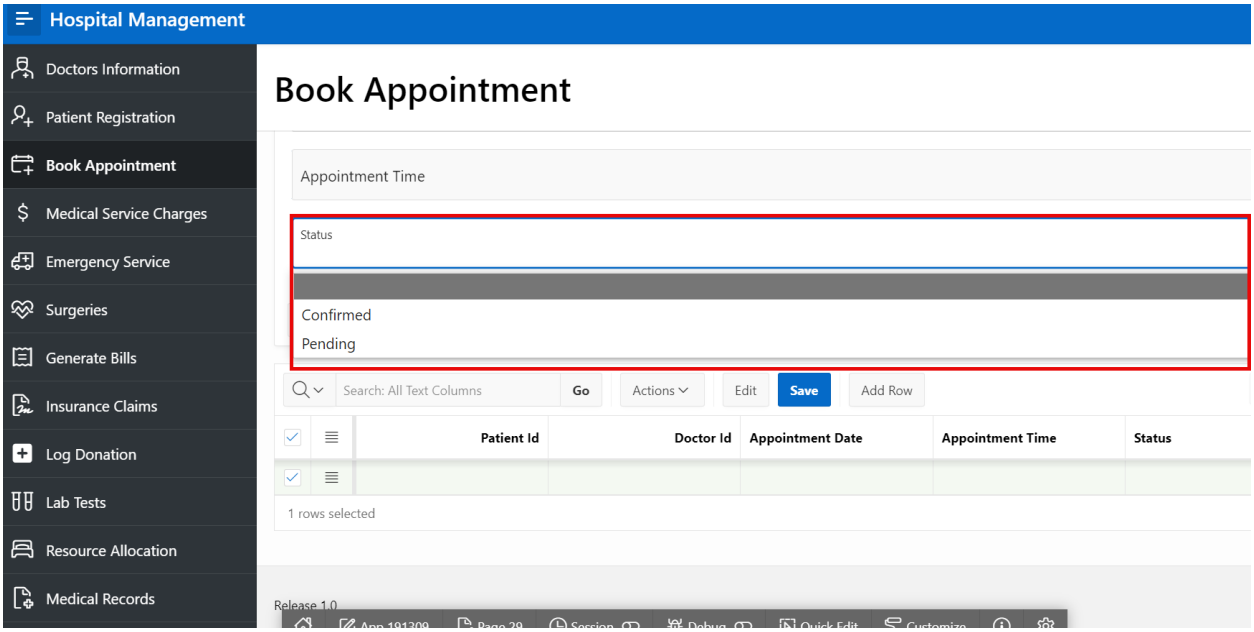
6. **Save the Appointment:**
    - After completing all fields, click the **Create** button to save the appointment.
    - The newly booked appointment will be displayed in the table below for reference and management.






Hospital Management
Doctors Information
Patient Registration
Book Appointment
Medical Service Charges
Emergency Service
Surgeries
Generate Bills
Insurance Claims
Log Donation
Lab Tests
Resource Allocation
December
2024
Sun Mon Tue Wed Thu Fri Sat
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31 1 2 3 4
Today
Appointment Date
Appointment Time
Status


Hospital Management
Doctors Information
Patient Registration
Book Appointment
Medical Service Charges
Emergency Service
Surgeries
Generate Bills
Insurance Claims
Log Donation
Lab Tests
Resource Allocation
Medical Records
Book Appointment
08:00 AM
08:30 AM
09:00 AM
09:30 AM
10:00 AM
10:30 AM
11:00 AM
Appointment Time
Status
Cancel

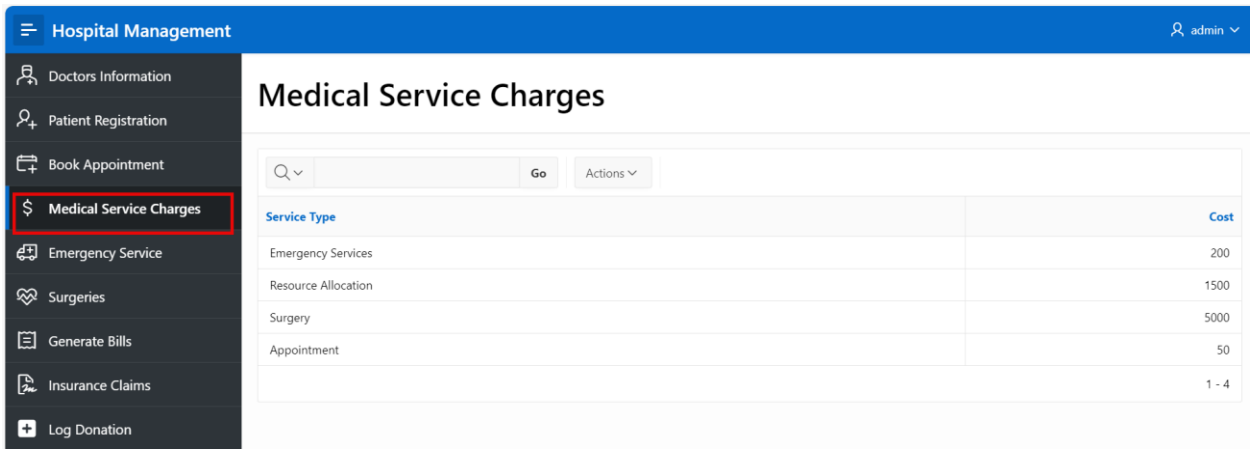

Hospital Management
Doctors Information
Patient Registration
Book Appointment
Medical Service Charges
Emergency Service
Surgeries
Generate Bills
Insurance Claims
Log Donation
Lab Tests
Resource Allocation
Medical Records
Book Appointment
Appointment Time
Status
Confirmed
Pending
Search: All Text Columns
Go
Actions
Edit
Save
Add Row
Patient Id
Doctor Id
Appointment Date
Appointment Time
Status
1 rows selected
Release 1.0

Medical Service Charges

The **Medical Service Charges** page displays the list of service types along with their associated costs. This allows administrators or users to view and manage the pricing for various medical services offered.

Details on the Page:

- **Service Type**: Lists the type of medical services provided (e.g., Emergency Services, Surgery, Appointments).
- **Cost**: Displays the cost associated with each service type (e.g., 200 for Emergency Services, 5000 for Surgery).



Emergency Service

The **Emergency Service** page allows users to log details of emergency cases and assign relevant patient and doctor information.

Fields in the Form:

1. **Patient ID**:
    - A dropdown to select the **Patient ID** from the list of registered patients.
    - This ensures the emergency case is linked to an existing patient.
2. **Emergency Type**:
    - A text field to specify the type of emergency (e.g., Cardiac Arrest, Accident, etc.).
3. **Doctor ID**:
    - A dropdown to select the **Doctor ID** from the list of available doctors.

- This field assigns the case to a specific doctor.

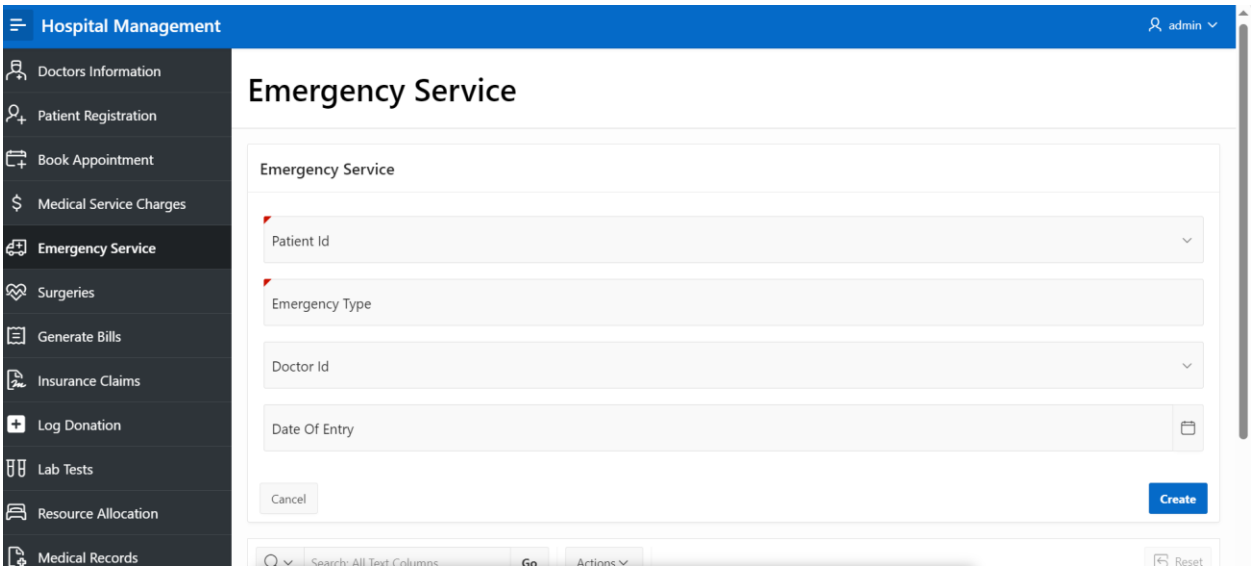
4. **Date of Entry**:
    - A date picker to log the date of the emergency service entry.

How to Use the Form:

1. **Select Patient**:
    - Use the **Patient ID** dropdown to select the patient involved in the emergency.
2. **Specify Emergency Type**:
    - Enter the nature of the emergency in the **Emergency Type** field (e.g., "Fracture" or "Severe Allergic Reaction").
3. **Assign Doctor**:
    - Choose the **Doctor ID** from the dropdown to assign a doctor to the case.
4. **Set Date of Entry**:
    - Use the date picker to specify the date of the emergency.
5. **Save the Record**:
    - Click the **Create** button to save the details of the emergency service.

Post-Submission Actions:

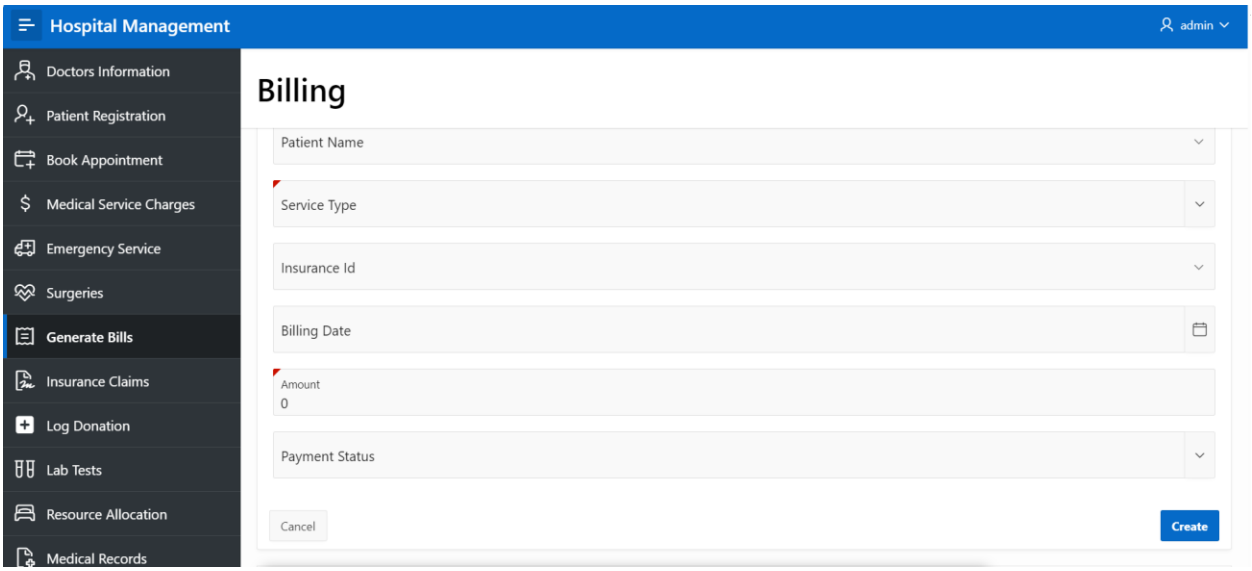
- Once the form is submitted, the newly created record will be displayed in the table below for reference and further updates if necessary.



Surgeries Page

- **Select Patient**:
- Use the **Patient ID** dropdown to choose the patient undergoing the surgery.
- **Assign Doctor**:
- Use the **Doctor ID** dropdown to assign the doctor responsible for the procedure.
- **Describe the Surgery**:
- Fill in the **Surgery Description** field with details about the surgery (e.g., procedure name or notes).
- **Set Surgery Date and Time**:
- Use the **Surgery Date** field to select the date of the surgery.
- Use the **Surgery Time** field to specify the exact time.
- **Specify Operating Room**:
- Enter the name or number of the operating room in the **Operating Room** field.
- **Save the Record**:
- Click the **Create** button to save the surgery details

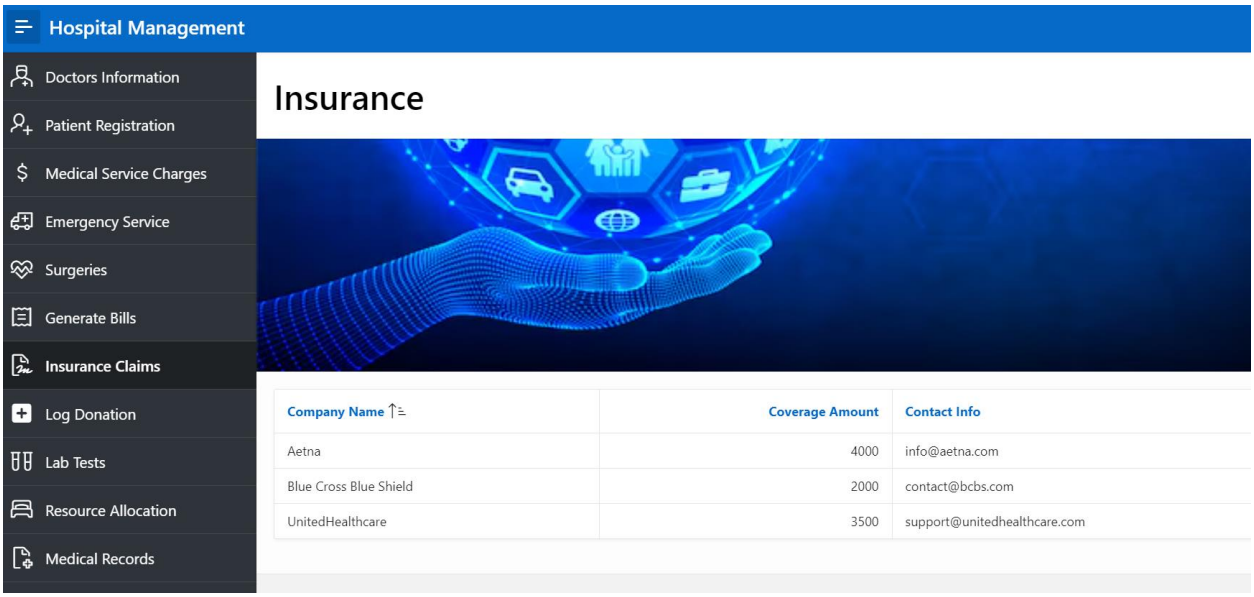
Generate Bills



Fields in the Form:

1. **Patient Name**:
    - A dropdown to select the patient's name from the list of registered patients.
    - Links the bill to the specific patient.
2. **Service Type**:
    - A dropdown to select the type of service rendered (e.g., Emergency Services, Surgery, etc.).
    - This can be linked to the **Medical Service Charges** table to fetch the cost automatically.
3. **Insurance ID**:
    - A dropdown to select the insurance details for the patient if applicable.
4. **Billing Date**:
    - A date picker to specify the date the bill is generated.
5. **Amount**:
    - Displays the total amount based on the selected service type (e.g., cost fetched from the **Medical Service Charges** table).
    - This field may be auto-populated and read-only.
6. **Payment Status**:
    - A dropdown to specify the payment status, such as **Paid**, **Pending**

INSURANCE CLAIMS



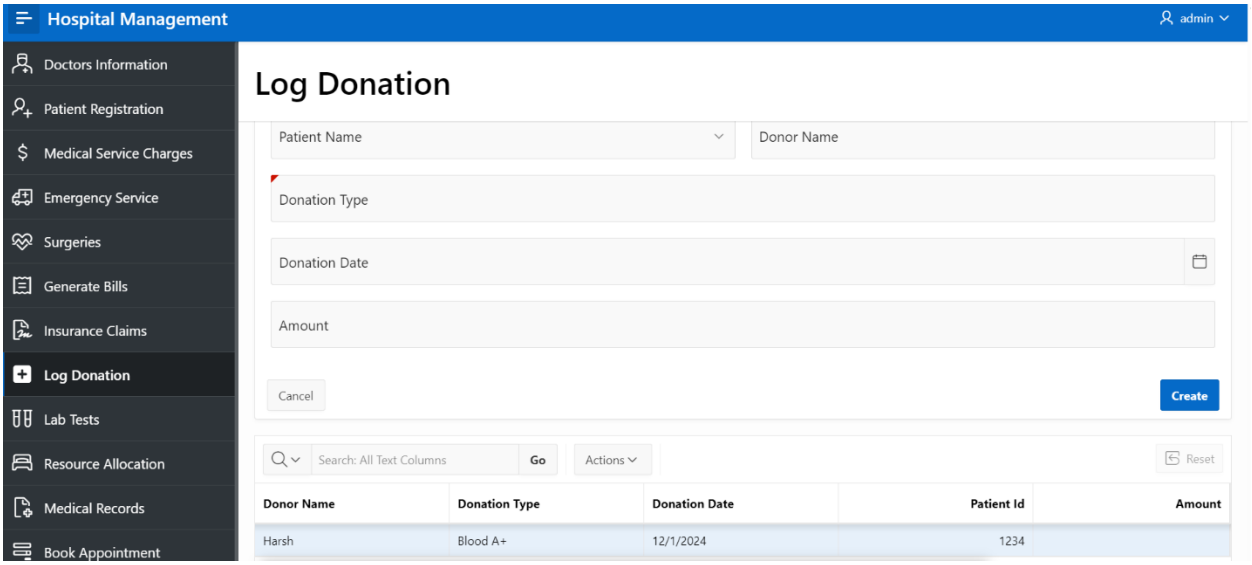
Process for Insurance Verification and Billing:

1. **Verify Patient's Insurance Status**
    - Check the hospital system or ask the patient directly if they have insurance coverage.
    - Retrieve the insurance company details, including the **Company Name**, **Coverage Amount**, and **Contact Information**, from the "Insurance Claims" section of the system.
2. **Send Verification Email**
    - Use the email address listed in the "Contact Info" column to send a verification email.
    - Wait for a confirmation email from the insurance company.
3. **Generate an Invoice**
4. **Send Invoice to the Insurance Company**
    - Email the invoice as an attachment to the insurance company's provided email address.
    - Include a clear subject and body summarizing the invoice
5. **Record the Action**

- Document the verification and invoice details in the hospital management system or manually in a log file/spreadsheet.
- Include the following:
  - Date of verification and invoice sending.
  - Patient details and service details.
  - Insurance company details.
  - Confirmation status and claim ID (if applicable).

6. **Follow Up if Necessary**
   - If payment is not received within the stipulated time, follow up with the insurance company via email or phone.

LOG DONATION



- Patient **Name** (Optional):

  Select or type the name of the patient associated with the donation (if applicable).

- **Donor Name**:

  Enter the name of the donor contributing the donation.

- **Donation Type**:

  Choose the type of donation from the dropdown (e.g., Cash, Blood, Organ, Medical Equipment, etc.).

- **Donation Date**:

  Use the date picker to select the date on which the donation was made.

- **Amount**:

  If the donation is monetary, enter the amount donated. Leave it blank if not applicable.