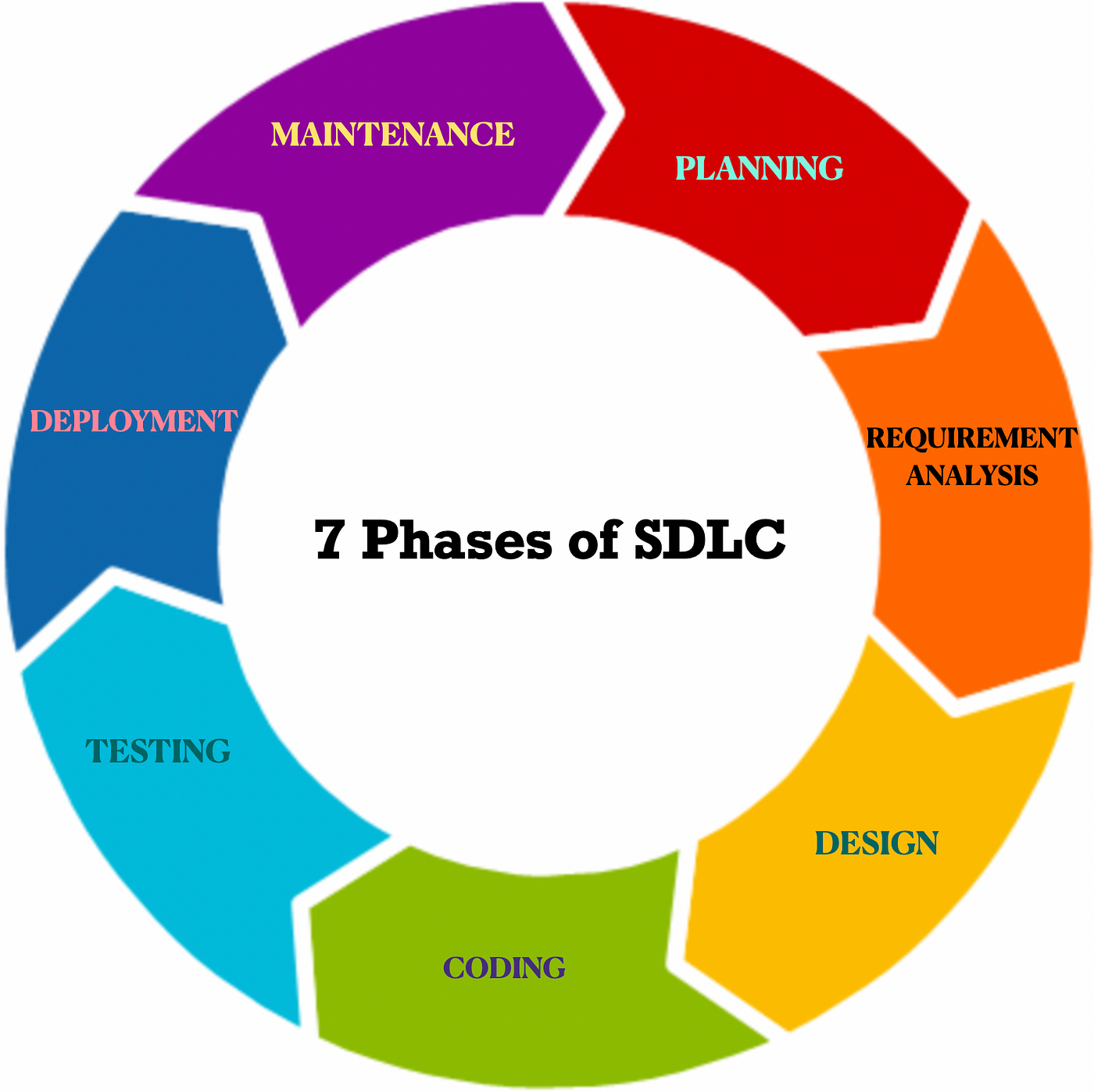
1. **SDLC OVERVIEW**





1. **Planning:** Decide what the system should do (store patient records, schedule appointments, manage billing, pharmacy, room allocation, and emergency services).
2. **Requirement Analysis:** Find out what the hospital staff needs (like keeping patient details, managing appointments, and tracking billing) and make sure everything works offline without the internet.
3. **Design:** Creating a database on the hospital’s computers to store all patient and hospital records.
4. **Coding:** Use a local database (like MySQL) to store hospital records and make sure the system runs fast.
5. **Testing:** Check if the system works properly offline and make sure patient records, billing, pharmacy and other services are saved and loaded correctly.
6. **Deployment:** Install the system on hospital computers, connect it to the local database and train hospital staff on how to use it easily.
7. **Maintenance:** Fix any problems that come up, provide updates and improvements and regularly back up data to keep it safe.

1. **REQUIREMENT GATHERING AND ANALYSIS**
   1. **Organization Details**

**1. Name of the Organization: -** Navrachana University

**2. Brief details of the Organization: -** Navrachana University is a premier multi-disciplinary private university located in Vadodara, Gujarat. NUV comprises five schools of learning spread across a sprawling new-age campus – School of Science (SoS), School of Engineering and Technology (SET), School of Business and Law(SBL), School of Environmental Design and Architecture (SEDA), and School of Liberal Studies and Education (SLSE). In 2021, NUV was ranked among Top 3 Pvt. Universities and Best in Vadodara by the Gujarat State Institutional Rating Framework (GSIRF).

**2.2 Meetings**

**1. Meetings with Principal: -**

**2. Principal Requirement in their words: -**

The system should work without the internet because hospitals may not always have a stable connection. It needs to store and manage patient records safely on hospital computers. Doctors and nurses should be able to find patient history quickly when needed. The billing system must create correct invoices without needing the internet. There should be a backup plan to keep data safe and prevent loss. The system should be easy to use so that hospital staff can learn it without much training. It should also allow the hospital to create reports on patient records, billing, and appointments.

**2.3 Data Which Will Be Input Into The System**

**1. Patient Information:** Name, age, gender, contact details, medical history, allergies, diagnosis, prescriptions.

**2. Doctor & Staff Details**: Name, specialization, shift timings, contact information, assigned wards.

**3. Appointment Data:** Patient name, doctor name, appointment date and time, reason for visit.

**4. Pharmacy Data:** Medicine name, stock levels, expiry date, price, supplier details.

**5. Emergency & Ambulance Records:** Emergency cases, patient condition, ambulance availability.

**6. Room & Bed Allocation:** Patient admission details, room number, bed number, discharge date.

**7. Billing & Payments:** Patient ID, services availed, medicine costs, consultation fees, total bill.

**2.4 Data Which Will Be Output From The System**

**1. Patient Reports:** Medical history, test results, treatment progress.

**2. Doctor & Staff Reports:** Attendance, assigned patients, working hours.

**3. Appointment Schedules:** Daily appointment lists for doctors.

**4. Pharmacy Reports:** Stock status, expired medicines, sales records.

**5. Emergency & Ambulance Reports:** Emergency cases handled.

**6. Room & Bed Allocation Reports:** Available rooms, occupied beds, discharge details.

**7. Billing Invoices:** Clear and detailed bills showing the cost of each service.

**2.5** **Type of Project**

Offline

**2.6 Method of Collecting Data**

* **Talking to Staff:** Had planned conversations with doctors, nurses, admin staff and pharmacists to understand their needs.
* **Surveys & Forms:** Gave hospital staff forms to fill out, asking about problems with patient records, billing, pharmacy and emergency services.
* **Watching How Work Happens:** Visited different hospital areas (like pharmacy, emergency, wards, and ambulance services) to see how they work.
* **Checking Documents:** Looked at patient records, appointment records, bills and pharmacy stock reports.

1. **SYSTEM REQUIREMENT SPECIFICATIONS**

**3.1 Introduction**

The Offline Hospital Management System for HealthFirst Hospital helps manage daily work like patient records, appointments, billing, pharmacy, emergency services, and inventory. It works without the internet. So, doctors, nurses and staff can always access important information without any interruptions.

**3.1.1 Purpose: -**

* Use a computer system instead of writing on paper.
* Works without the internet, so the hospital can use it anytime.
* Keep patient records safe with details about their visits and treatments.
* Manage hospital rooms, beds, and ambulances properly.
* Track medicines to know what is available and when they expire.
* Make billing easy by creating bills automatically.
* Keep hospital data private and safe.

**3.1.2** **Intended Audience and Reading Suggestions**

**Intended Audience:**

This SRS document is for:

* **Hospital Managers and Administrators:** To see how the system helps the hospital.
* **Doctors and Staff:** To check patient records and appointments.
* **Pharmacists:** To manage medicine inventory and prescriptions.
* **IT Team and Testers:** To build and test the system.
* **Receptionists and Billing Staff:** To use the system for daily hospital work.

**Reading Suggestions:**

* **Introduction and System Overview:** For everyone to understand the system.
* **System Overview:** Important for developers and testers.
* **System Features and Design:** Useful for hospital staff and managers.
* **Security and Maintenance:** IT team should read this to keep the system safe.
  + 1. **System Scope: -**

The HealthFirst Hospital Management System (HMS) helps hospitals work efficiently without the internet. It includes:

* **Patient Management:** Saves records, history, and appointments.
* **Doctor and Staff Management:** Manages schedules and duties.
* **Pharmacy Management:** Tracks medicines and prescriptions.
* **Billing:** Creates invoices for treatments and medicines.
* **Emergency:** Handles emergencies and ambulances.
* **Room and Bed Allocation:** Assigns and tracks availability.
* **Security:** Stores data safely with restricted access.

**3.2 Overall Description**

**3.2.1 System Function:-**

**(1) Patient Management Module**

* Add patients
* Update records
* Manage appointments

**(2) Doctor & Staff Management Module**

* Keep doctor and staff details
* Schedule shifts
* Track availability

**(3) Pharmacy Management Module**

* Manage medicines
* Generate stock alerts.

**(4) Billing & Payment Module**

* Create bills
* Track payment
* Generate invoices and receipts

**(5) Emergency & Ambulance Management Module**

* Record emergency cases
* Track ambulance availability

**(6) Room & Bed Management Module**

* Allocate rooms
* Check bed availability.
* Manage discharge and room cleaning

**(7) Security & User Management Module**

* Control user access
* Secure data
* Track system activity

**3.2.2 User classes and characteristics: -**

**(1) User Classes**

Administrator

Doctors

Hospital Staff (Includes Nurses, Receptionists, Pharmacists, Billing Staff, Emergency and Ambulance Staff)

**(2) User Characteristics**

Administrator

* Full control the system, manages users, security and hospital reports.

Doctors

* Access patient records, manage appointments, give treatments, and update progress.

Hospital Staff (Nurses, Receptionists, Pharmacists, Billing, Emergency Staff)

* Help with patient care, scheduling, admin work, medicine stock, billing, and emergencies.

**3.2.3 Operating Environment:-**

* This system will operate offline on PCs and local servers within the hospital.
* It will support Windows and Linux operating systems.
* Requires a minimum 4GB RAM, 500GB storage.

**3.2.4 Design and Implementation**

* GUI is only in English.
* No internet connection is required as the system is offline.
* User should have basic computer knowledge to operate the system.

**3.2.5 Assumptions and Dependencies**

* Hospital departments and staff are already added to the system.
* Only hospital admins can manage the database.
* Users have fixed roles and access based on their job.

**3.3 External Interface Requirement**

**3.3.1 User Interface:-**

* + - If a patient is not registered, hospital staff can register them in the system.
    - Once registered, the patient’s details can be accessed by authorized staff.
    - If login credentials (Username/Password) are incorrect, an error message will be displayed.
    - Doctors and staff can view, update, and manage patient records, appointments, and billing through a structured dashboard.

**3.3.2 Software Interface:-**

* + - **Operating System –** Windows 10 / Linux
    - **Front End –**Python
    - **Back End –** MySQL
    - **Software for Front End Development –** Visual Studio
    - **Database Management System –** MySQL

**3.4 System Features**

* Staff can register new patients if they are not in the system.
* Only authorized staff can access patient details.
* An error message appears for wrong login details.
* Doctors and staff can manage records, appointments and billing from a dashboard.

**3.5 Other Non-Functional Requirements**

**3.5.1 Performance Requirement: -**

* The system should work smoothly offline with little delay.
* It must handle a large amount of patient and hospital data without slowing down.
* Tasks like patient search, billing, and reports should be fast.

**3.5.2 Security Requirements**

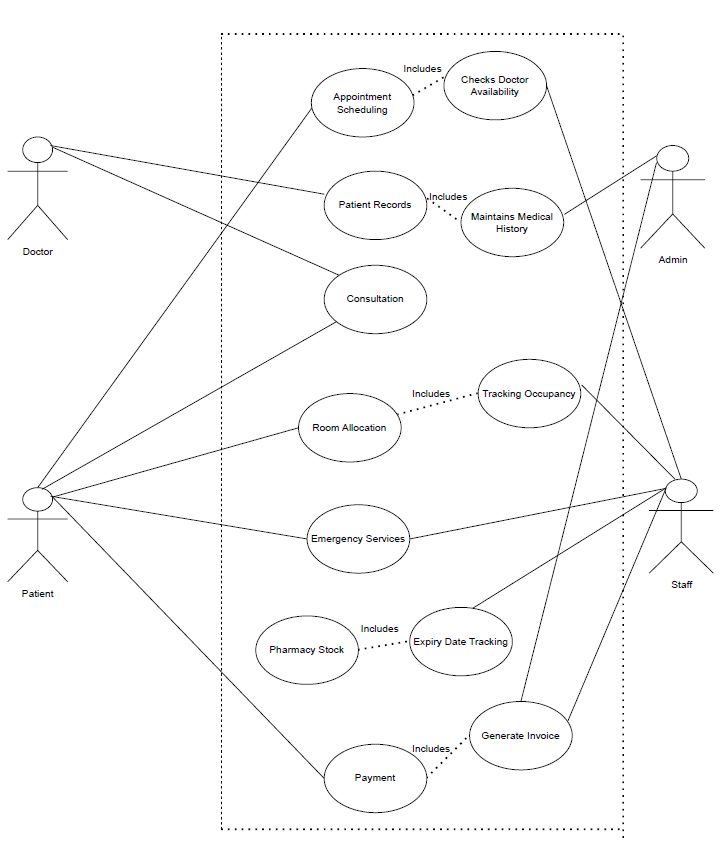
* Doctors have more access than hospital staff.
* Only authorized users can see patient and hospital data.
* Hospital management decides user access and permissions.
* Data is protected with encryption and authentication.

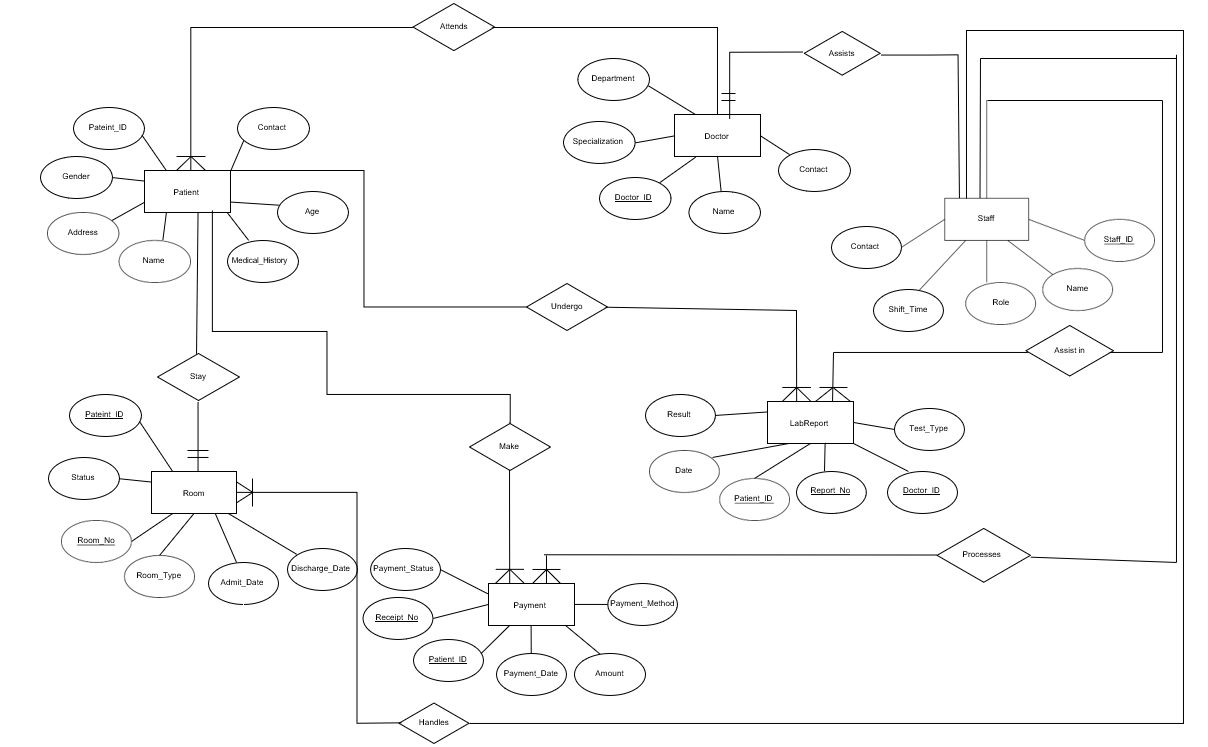
**3.5.3 Safety Requirements**

* Patient records, medical history and bills are safely stored to avoid data loss.
* Regular backups are done to recover data if the system stops working.
* Only approved staff can change or delete important hospital data.
* Login required to stop unknown people from using the system.

1. **SYSTEM ANALYSIS AND MODELING**

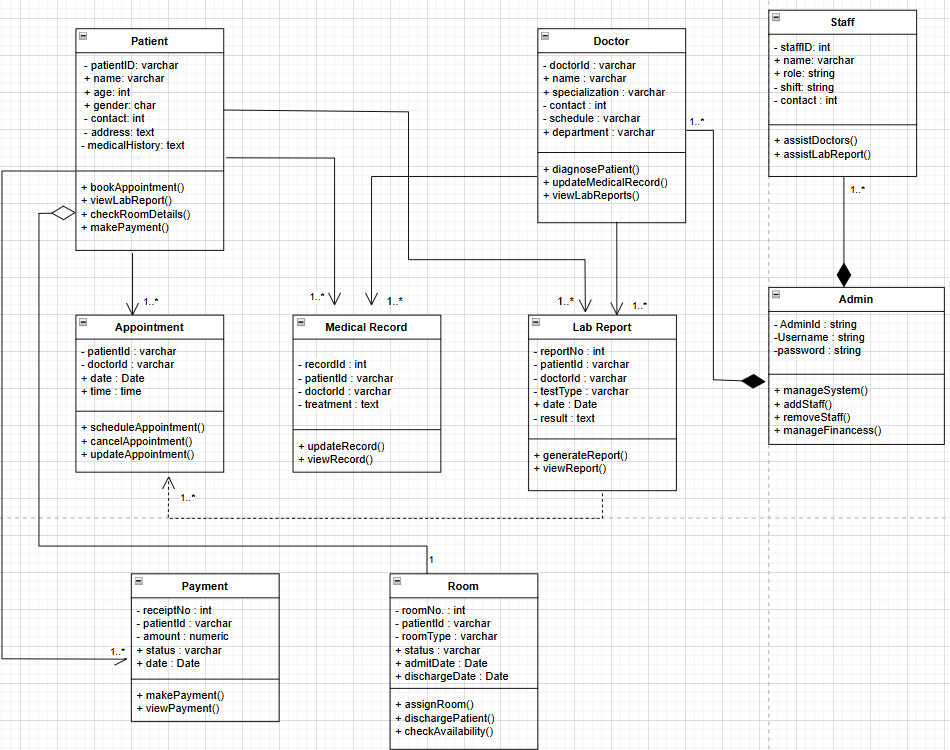
**4.1 Use Case Diagram**



**4.2 E-R Diagram**



**4.3 Class Diagram**

****

**4.4 Data Dictionary**

**Table 1 : Doctor**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No. | Column Name | Data Type | Null | Comments |
| 1 | Doctor\_ID | VARCHAR(100) | NO | Primary Key, Unique ID for doctor |
| 2 | Name | VARCHAR(100) | NO | Doctor's full name |
| 3 | Specialization | VARCHAR(100) | NO | Doctor's field of expertise |
| 4 | Contact | INT | NO | Contact number |
| 5 | Department | VARCHAR(100) | NO | Department assigned |

**Table 2 : Patient**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No. | Column Name | Data Type | Null | Comments |
| 1 | Patient\_ID | VARCHAR(100) | NO | Primary Key, Unique ID for patient |
| 2 | Name | VARCHAR(100) | NO | Patient's full name |
| 3 | Age | INT | NO | Patient's age |
| 4 | Gender | CHAR(1) | NO | M/F/O |
| 5 | Contact | INT | NO | Contact number |
| 6 | Address | TEXT | YES | Residential address |
| 7 | Medical\_History | TEXT | YES | Past medical conditions |

**Table 3 : Room**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No. | Column Name | Data Type | Null | Comments |
| 1 | Room\_No | INT | NO | Primary Key, Unique ID for room |
| 2 | Patient\_ID | VARCHAR(100) | YES | Foreign Key references Patient Table (NULL if available) |
| 3 | Room\_Type | VARCHAR(50) | NO | Type of room (General, ICU, Private, etc.) |
| 4 | Status | VARCHAR(20) | NO | Occupied / Available |
| 5 | Admit\_Date | DATE | YES | Date of patient admission |
| 6 | Discharge\_Date | DATE | YES | Date of patient discharge (NULL if still admitted) |

**Table 4 : Lab Report**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No. | Column Name | Data Type | Null | Comments |
| 1 | Date | DATE | NO | Date of the lab test |
| 2 | Patient\_ID | VARCHAR(100) | NO | Foreign Key (Patient Table) |
| 3 | Report\_ID | INT | NO | Primary Key, Unique ID |
| 4 | Doctor\_ID | VARCHAR(100) | NO | Foreign Key (Doctor Table) |
| 5 | Test\_Type | VARCHAR(100) | NO | Type of medical test |
| 6 | Results | TEXT | YES | Test results |

**Table 5 : Payment**

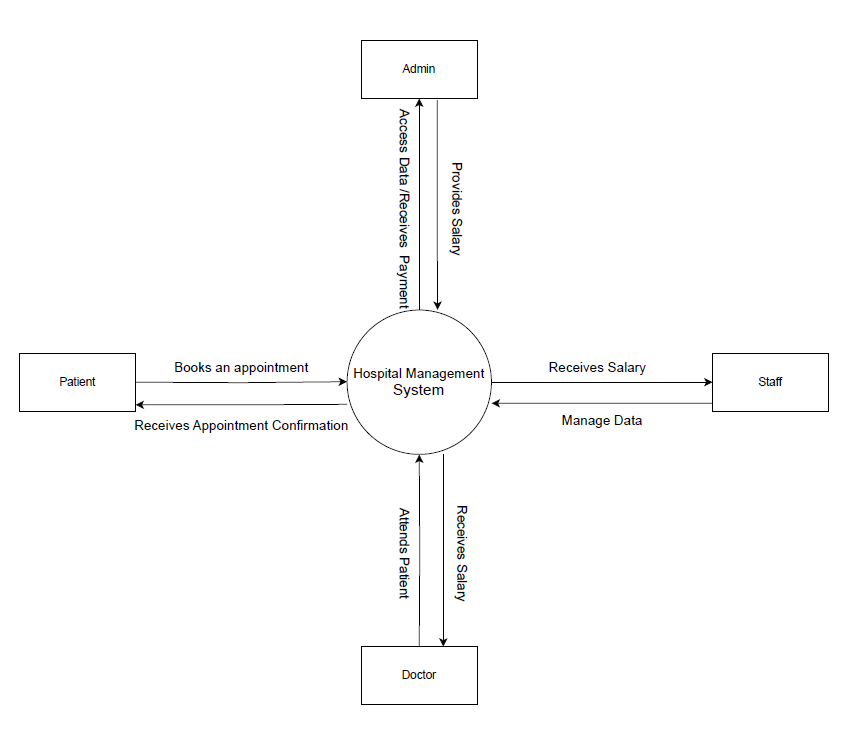
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No. | Column Name | Data Type | Null | Comments |
| 1 | Receipt\_No | INT | NO | Primary Key, Unique ID |
| 2 | Patient\_ID | VARCHAR(100) | NO | Foreign Key (Patient Table) |
| 3 | Amount | NUMERIC(10) | NO | Payment amount |
| 4 | Payment\_Method | VARCHAR(50) | NO | Cash/Card/Insurance |
| 5 | Payment\_Status | VARCHAR(50) | NO | Paid/Pending |
| 6 | Payment\_Date | DATE | YES | Date of payment |

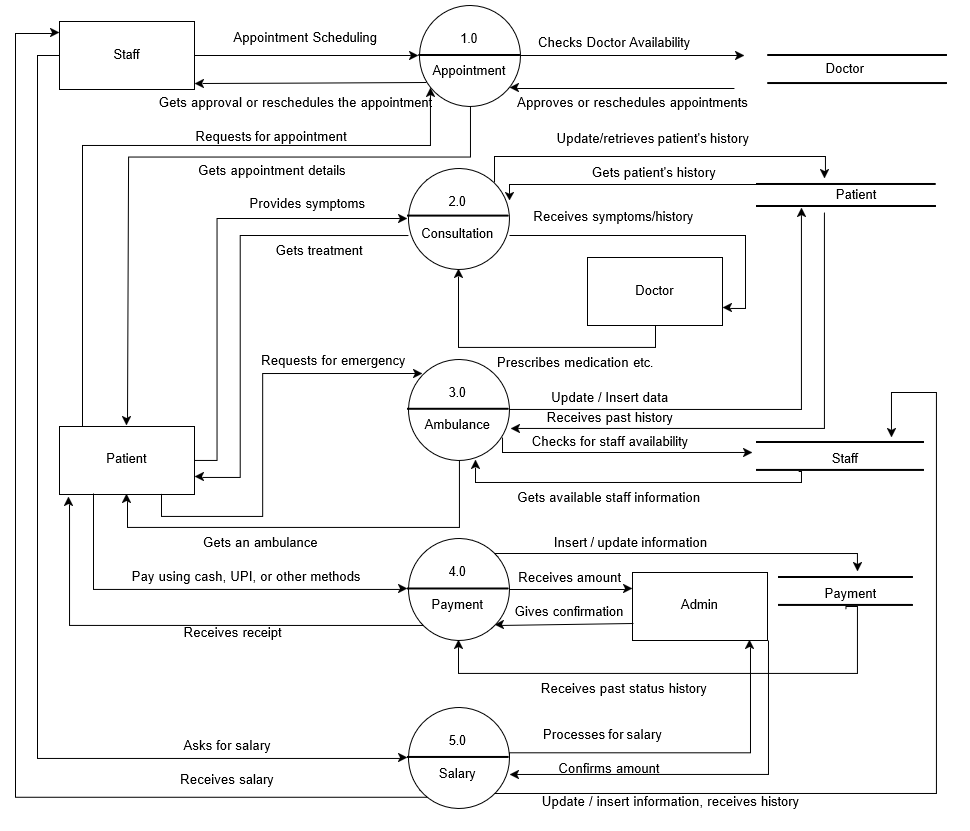
**Table 6 : Staff**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sr. No. | Column Name | Data Type | Null | Comments |
| 1 | Staff\_ID | INT | NO | Primary Key, Unique ID |
| 2 | Name | VARCHAR(100) | NO | Staff member's full name |
| 3 | Role | VARCHAR(50) | NO | Job role (Nurse/Receptionist etc.) |
| 4 | Contact | INT | NO | Contact number |
| 5 | Shift\_Time | VARCHAR(50) | NO | Shift timing (Morning/Night) |

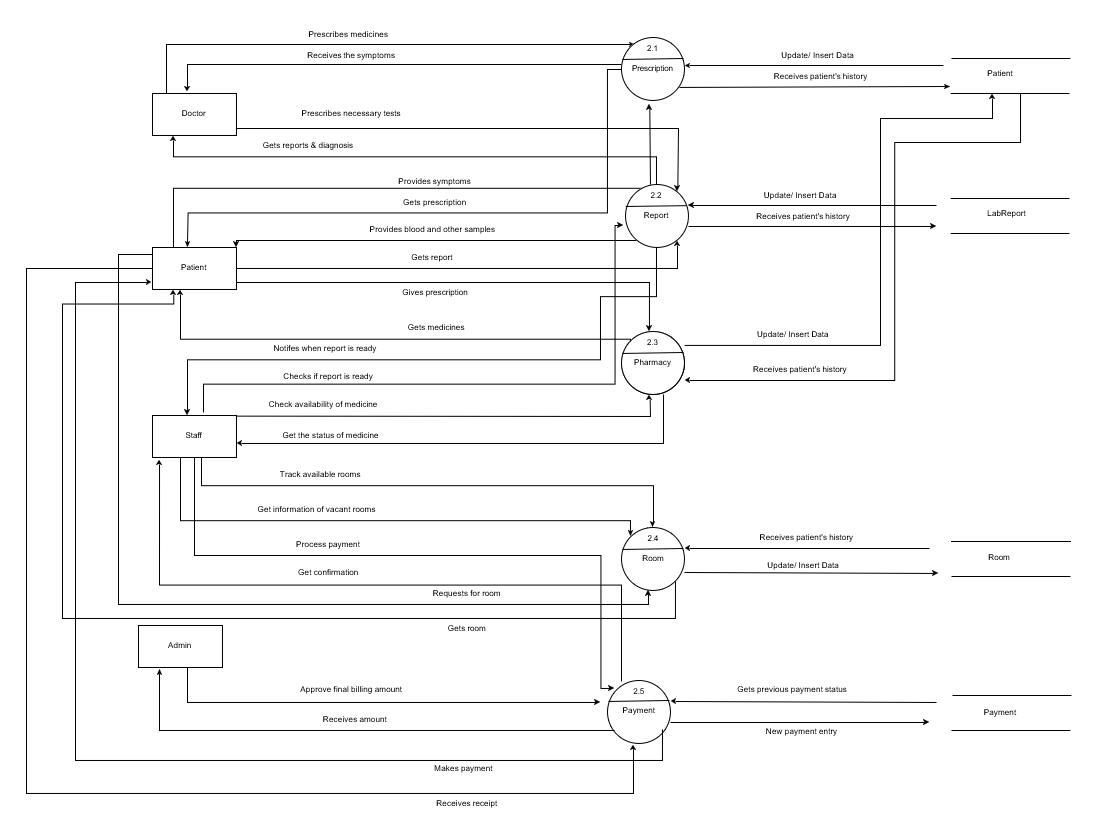
**4.5 Functional and Behavioral Modelling**

**4.5.1 Context Diagram (0-Level Diagram): -**

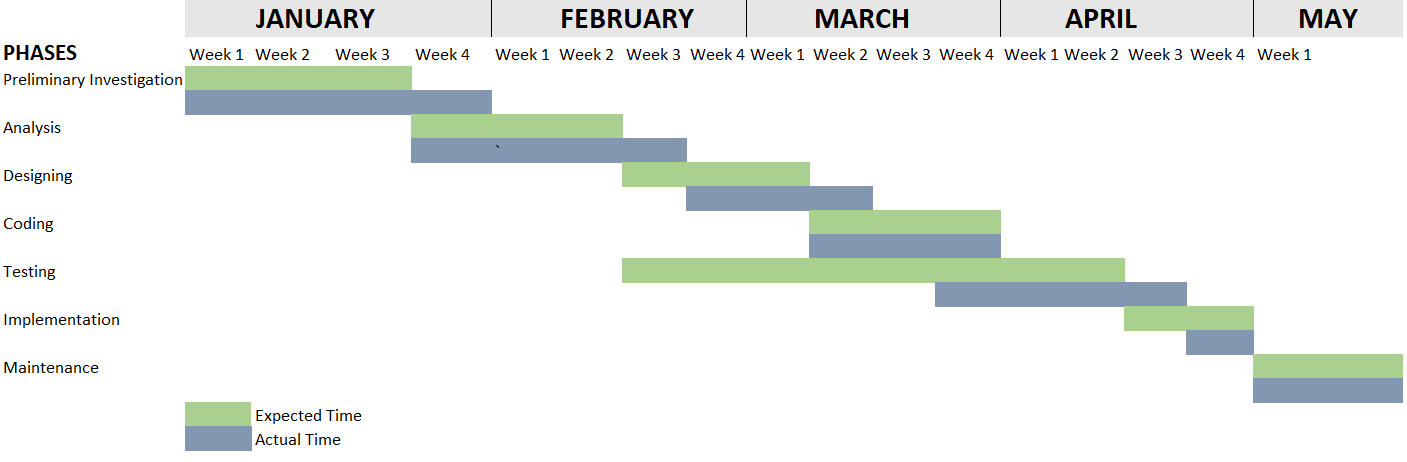


** 4.5.2 First Level Diagram (1-Level Diagram): -**

**4.5.3 Second Level Diagram (2-Level Diagram): -**



**4.6 Gantt Chart**

****

**5. TESTCASE**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID** | **Description** | **Steps** | **Test Data** | **Expected Output** | **Actual Output** | **Result** | **Remarks** |
| TC\_01 | Check **Doctor Login** with valid credentials | Open Login Page → Enter Email → Enter Password → Click Login | Email: [**doctor@gmail.com**](mailto:doctor@gmail.com) Password: **doc123** | Doctor should log in successfully | Same as expected | Pass | - |
| TC\_02 | Check **Doctor Login** with invalid credentials | Open Login Page → Enter Email → Enter Password → Click Login | Email: [**doctor@gmail.com**](mailto:doctor@gmail.com) Password: **wrongpass** | Doctor should **not** log in | Same as expected | Fail | Incorrect credentials |
| TC\_03 | Check **Patient Registration** with valid details | Open Registration Page → Enter Patient Details → Click Submit | Name: **John Doe** Age: **30** Contact: **9876543210** | Patient should be registered successfully | Same as expected | Pass | - |
| TC\_04 | Check **Patient Registration** with missing details | Open Registration Page → Enter **incomplete** details | Name: **John** Age: **N/A** | System should **not register** the patient and show an **error message** | Same as expected | Fail | Error: Missing Information |
| TC\_05 | Check **Appointment Scheduling** for registered patient | Open Appointment Module → Enter Patient ID → Select Doctor & Time Slot → Click Confirm | Patient ID: **P001** Doctor: **Dr. Smith** Time: **10:00 AM** | Appointment should be scheduled successfully | Same as expected | Pass | - |
| TC\_06 | Check **Medicine Inventory Update** by Staff | Open Pharmacy Module → Select Medicine → Update Stock → Save | Medicine: **Paracetamol** New Stock: **50** | Inventory should update with the new stock | Same as expected | Pass | - |
| TC\_07 | Check **Emergency Service Request** | Open Emergency Module → Enter Patient ID & Emergency Details → Click **Request Ambulance** |  | Ambulance should be **notified immediately** | Same as expected | Pass |  |
| TC\_08 | Check **Billing Module** for Patient Discharge | Open Billing Module Enter Patient ID → Add services → Generate Bill | **Patient ID:** P003 **Total Charges:** $500 | Invoice should be generated successfully | Same as expected | Pass | - |
| TC\_09 | Check **Access Control** for Hospital Staff | Hospital Staff enters login credentials | Email: [**staff@hospital.com**](mailto:staff@hospital.com) Password: **staff456** | Staff should log in successfully | Same as expected | Pass | - |
| TC\_10 | Check **Unauthorized Access for Staff** | Staff tries to access **patient records without permission** | Access Denied message should be shown | Same as expected | Pass | Unauthorized Access Blocked | - |

**6. LIMITATIONS AND FUTURE ENHANCEMENTS**

**6.1 Limitations:**

* Patients cannot change their own records.
* The system does not work online, only offline.
* It can only be used inside the hospital for adding and viewing data.

**6.2 Future Enhancements:**

* Patients may get a portal to see their medical history.
* The system can be made online so it works from anywhere.

**7. CONCLUSION**

* The HealthFirst Offline Hospital System has been successfully made. It helps with patient registration, appointments, billing, etc.
* The system works well and is easy for doctors, staff, and managers to use. It reduces paperwork, improves hospital work, and helps in better patient care.

1. **REFERENCE AND BIBLIOGRAPHY**

**Reference :**

WHO (World Health Organization) : *Digital Health Solutions for Hospitals*. Retrieved from [www.who.int](https://www.who.int)

ResearchGate: *Offline Medical Record Management*. Retrieved from [www.researchgate.net](https://www.researchgate.net)

**Bibliography :**

Government healthcare reports on hospital management.