

Hospital Management System Registration Block



Submitted by:

- Harsh Vardhan Patel (2206217)
- Aditi Kumari (2206250)
- Shivani Saroj (2206251)
- Priyanshu Raj (2206249)
- Jay vardhan Vashishtha (2206252)

Under the supervision of

DR. DASHARATH MAHTO

Department of Computer Science

National Institute of Technology Patna

Patna, Bihar

Session 2022-2026

Acknowledgement

We would like to express our deepest gratitude for providing the necessary resources and support for the successful completion of our project titled "Hospital Management System Registration Block."

The commitment to fostering a research-friendly environment has been instrumental in enabling our work. Our heartfelt thanks go to our mentors for their continuous guidance, insightful feedback, and unwavering support throughout the duration of this project. Their expertise and mentorship have been invaluable, enabling us to navigate the complexities of this research effectively.

We also wish to acknowledge the contributions of our colleagues and the entire team at NIT PATNA. Their collaboration and support have been vital in achieving the project's objectives. Lastly, we are immensely grateful to our families and friends for their encouragement and support, which have been a source of motivation and strength throughout this journey.

Thank you all for your invaluable contributions and support. This project is a testament to our collective efforts and dedication as a group from NIT Patna, Department of Computer Science and Engineering.

With sincere regards,

Declaration

We, hereby declare that this project report titled "Hospital Management System Registration Block" is an authentic record of my own work carried out under the guidance of Dr. Dashrath Mahto. The project was developed within a time frame of Labs using ReactJs, and Firebase. All the ReactJs files and components described in this report were created by me unless otherwise acknowledged in the references.

Date:

Signature:

Table Of Content

SNO	TOPIC	PAGE NO.
1.	PROBLEM STATEMENT	
2.	PROCESS MODEL	
3.	SOFTWARE REQUIREMENTS SPECIFICATION	
4.	SPECIFIC REQUIREMENT	
5.	ER DIAGRAM	
6.	LEVEL WISE DIAGRAM(DATAFLOW)	
7.	USE CASE DIAGRAM	
8.	USE CASE DESCRIPTION	
9.	DATA DICTIONARY	
10.	ER DIAGRAM	
11.	DATA DESIGN	
12.	COMPONENT LEVEL DIAGRAM	
13.	PROJECT SCHEDULING	
14.	TIMELINE CHART	
15.	FUNCTION POINT METRICS	
16.	COCOMO MODEL	
17.	SAMPLE SCREENSHOTS	
18.	RISK ANALYSIS	
19.	TESTING	

20.	CONCLUSION	
-----	------------	--

1 PROBLEM STATEMENT

Objective:

Design and implement a Registration Block module for a Hospital Management System (HMS) to efficiently manage the registration process of patients in the hospital lab. The system should streamline patient registration, manage patient data, and facilitate smooth communication between different departments.

Requirements:

1. Patient Registration:

- The system should allow the registration of new patients, including capturing essential details such as:
 - Full Name
 - Age
 - Gender
 - Contact Information (Phone Number, Email, Address)
 - Emergency Contact Details
 - Medical History (existing conditions, allergies, etc.)
 - Preferred Doctor or Department
- Provide an option to generate a unique Patient ID for each registered patient.

2. Appointment Scheduling:

- Allow patients to schedule appointments with doctors or lab tests directly during registration.
- Provide options for selecting preferred dates and times for appointments based on availability.
- Notify relevant departments (such as the lab or specific doctors) about the scheduled appointments.

3. Billing and Payment Integration:

- Include an interface for generating and managing bills related to registration fees, lab tests, and other services.

- Provide options for different payment methods (cash, credit/debit card, online payment, etc.).
- Ensure secure payment processing and maintain a record of all transactions.

4. Data Management and Security:

- Store all patient information securely with access control to protect sensitive data.
- Ensure compliance with data privacy regulations (such as HIPAA, GDPR) to safeguard patient data.
- Provide a searchable database to retrieve patient information quickly and accurately.

5. User Roles and Permissions:

- Define different user roles, such as Receptionist, Lab Technician, Doctor, and Administrator, with specific permissions:
 - Receptionists should handle patient registration and scheduling.
 - Lab Technicians should access patient data related to lab tests.
 - Doctors should access patient history and lab results.
 - Administrators should manage system settings and user roles.

6. Reporting and Analytics:

- Generate reports on patient registrations, lab tests, and appointment statistics.
- Include analytics to track patient flow, peak registration times, and other key performance indicators.

7. Notification System:

- Implement an automated notification system to send reminders to patients for appointments and test results via SMS or email.
- Notify staff members about new registrations, scheduled appointments, or any cancellations.

8. User Interface:

- Design a user-friendly interface for both staff and patients, with easy navigation and clear instructions.
- Ensure the system is accessible on various devices (desktop, tablet, mobile).

9. Scalability and Integration:

- Ensure the system can scale to handle a large number of patients and staff.
- Provide integration capabilities with other hospital systems, such as electronic health records (EHR), pharmacy, and billing systems.

Deliverables:

1. A fully functional Registration Block module integrated with the Hospital Management System.
2. A detailed report documenting the system architecture, data flow, user roles, and permissions.
3. User manuals and training materials for hospital staff.
4. Testing and validation results to ensure the system meets all requirements.

Constraints:

- The system must handle concurrent registrations efficiently.
- It must ensure a response time of under 2 seconds for key operations (like patient search, registration, or appointment scheduling).
- The solution should be built with a modern technology stack that supports scalability, security, and maintainability.

Conclusion:

This Registration Block module will streamline the hospital's patient registration process, improve operational efficiency, and enhance patient experience by minimizing wait times and providing easy access to healthcare services.

2. PROCESS MODEL

For the Hospital Management System Registration Block, we can use an appropriate software development process model to ensure that the module is built efficiently, meets all requirements, and can be easily maintained and scaled. Here are a few process models that are commonly used for such projects, with one being recommended as the most suitable:

1. Project Initiation:

- Define the scope and objectives of the Registration Block module.
- Identify the key stakeholders (e.g., hospital management, IT team, end-users like receptionists, lab technicians, and doctors).
- Develop a high-level project plan, including timelines, resources, and milestones.

2. Requirement Gathering:

- Conduct meetings with stakeholders to gather detailed requirements.
- Prioritize features based on importance, urgency, and potential impact.
- Create user stories and acceptance criteria for each requirement.

3. Sprint Planning:

- Break down the development process into multiple iterations (sprints), typically lasting 2-4 weeks.
- Select a set of user stories from the product backlog to be completed in the current sprint.
- Define the goals, deliverables, and tasks for the sprint.

4. Design and Prototyping:

- Design the system architecture, database schema, and user interface.
- Create prototypes or wireframes for the user interface to validate the design with stakeholders.
- Gather feedback and make necessary adjustments before implementation.

5. Development (Iterative):

- Develop the features selected for the current sprint using Agile practices (e.g., pair programming, test-driven development).
- Conduct daily stand-up meetings to monitor progress, identify blockers, and facilitate team communication.
- Integrate and test the developed features continuously to ensure functionality and quality.

6. Testing:

- Perform comprehensive testing for each sprint, including unit testing, integration testing, and user acceptance testing (UAT).
- Involve end-users (receptionists, lab technicians) to validate that the module meets their requirements.
- Capture feedback and address any defects or changes required.

7. Review and Retrospective:

- At the end of each sprint, conduct a sprint review meeting to demonstrate the completed features to stakeholders.
- Gather feedback and update the product backlog accordingly.
- Hold a sprint retrospective meeting to discuss what went well, what could be improved, and actions for the next sprint.

8. Deployment:

- Deploy the completed features of the Registration Block module in a production or staging environment.
- Monitor system performance and user feedback for any issues.

9. Maintenance and Support:

- Provide ongoing maintenance and support to address any bugs, feature requests, or performance issues.
- Plan for future updates or enhancements based on user feedback and evolving requirements.

CHAPTER 3

SOFTWARE REQUIREMENT SPECIFICATION

1. Product Perspective
 - a. System Interfaces
 - b. System Specifications
 - i. H/W Requirement
 - ii. S/W Requirement
 - iii. Communication Interfaces
2. Product functions
3. Data Flow Diagram (DFD)
 - a. Context Level Diagram
 - b. DFD Level – 1
 - c. DFD Level – 2
4. Use Case Diagram
5. Use Case Description
6. User characteristics
7. Constraints
8. Assumptions and dependencies

1. Product Perspective

This Hospital Patient Info Management System is a self-contained system that manages activities of the hospital.

Due to improperly managed details medical center faces quite a lot of difficulties in accessing past data as well as managing present data. The fully functional automated hospital management system which will be developed through this project will eliminate the disadvantages caused by the manual system by improving the reliability, efficiency and performance. The usage of a database to store patient, employee, stock details etc. will accommodate easy access, retrieval, and search and manipulation of data. The access limitations provided through access privilege levels will enhance the security of the system. The system will facilitate concurrent access and convenient management of activities of the medical center.

❖ System Interfaces

❖ User Interfaces

- This section provides a detailed description of all inputs into and outputs from the system. It also gives a description of the hardware, software and communication interfaces and provides basic prototypes of the user interface.
- The protocol used shall be HTTP.
- The Port number used will be 80.
- There shall be logical address of the system in IPv4 format.

❖ Hardware Interfaces

- Laptop/Desktop PC-Purpose of this is to give information when Patients ask information about doctors, medicine available lab tests etc. To perform such Action it need very efficient computer otherwise due to that reason patients have to wait for a long time to get what they ask for.
- Laser Printer (B/W) - This device is for printing patients' info etc.
- Wi-Fi router - Wi-Fi router is used to for internetwork operations inside of a hospital and simply data transmission from pc to server.

❖ Software Interfaces

- JDK 1.8 - Java is fast, secure, and reliable. From laptops to data centers, game consoles to scientific supercomputers, cell phones to the Internet,
- Mysql server - Database connectivity and management
- OS Windows 7/8/8.1- Very user friendly and common OS
- JRE 1.8 - JAVA Runtime Environment for run Java Application and System

2.1.2 System Specifications

2.1.2.1 H/W Requirement

Core i5 processor

2GB Ram.

20GB of hard disk space in terminal machines

1TB hard disk space in Server Machine

2.1.2.2 S/W Requirement

Windows 7 or above operating system

JRE 1.8

2.1.3 Communication Interfaces

NIC (Network Interface Card) – It is a computer hardware component that allows a computer to connect to a network. NICs may be used for both wired and wireless connections.

CAT 5 network cable- for high signal integrity

TCP/IP protocol- Internet service provider to access and share information over the Internet

Ethernet Communications Interface- Ethernet is a frame-based computer network technology for local area networks (LANs)

Ubiquitous, easy to set up and easy to use. Low cost and high data transmission rate.

2.3 DATA FLOW DIAGRAM (DFD)

CONTEXT LEVEL DIAGRAM

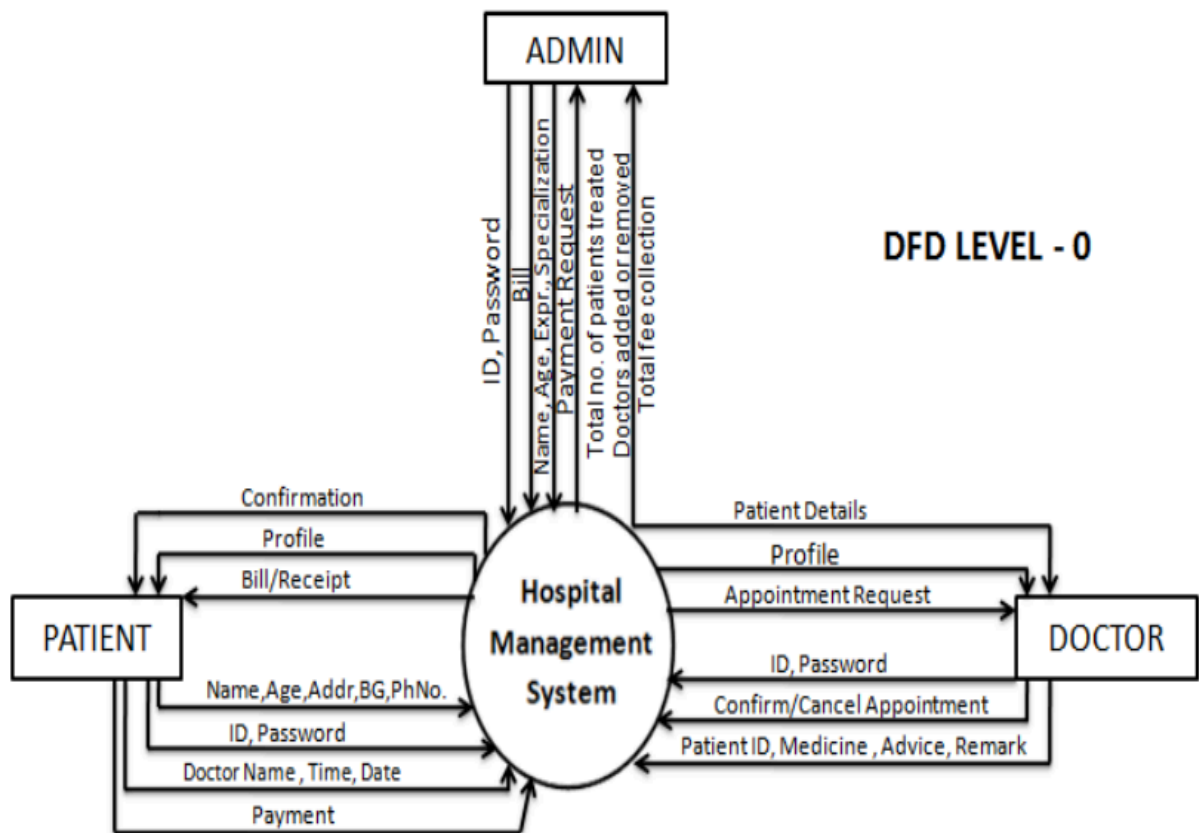


FIGURE 2.1 CONTEXT LEVEL DFD

DFD LEVEL – 1

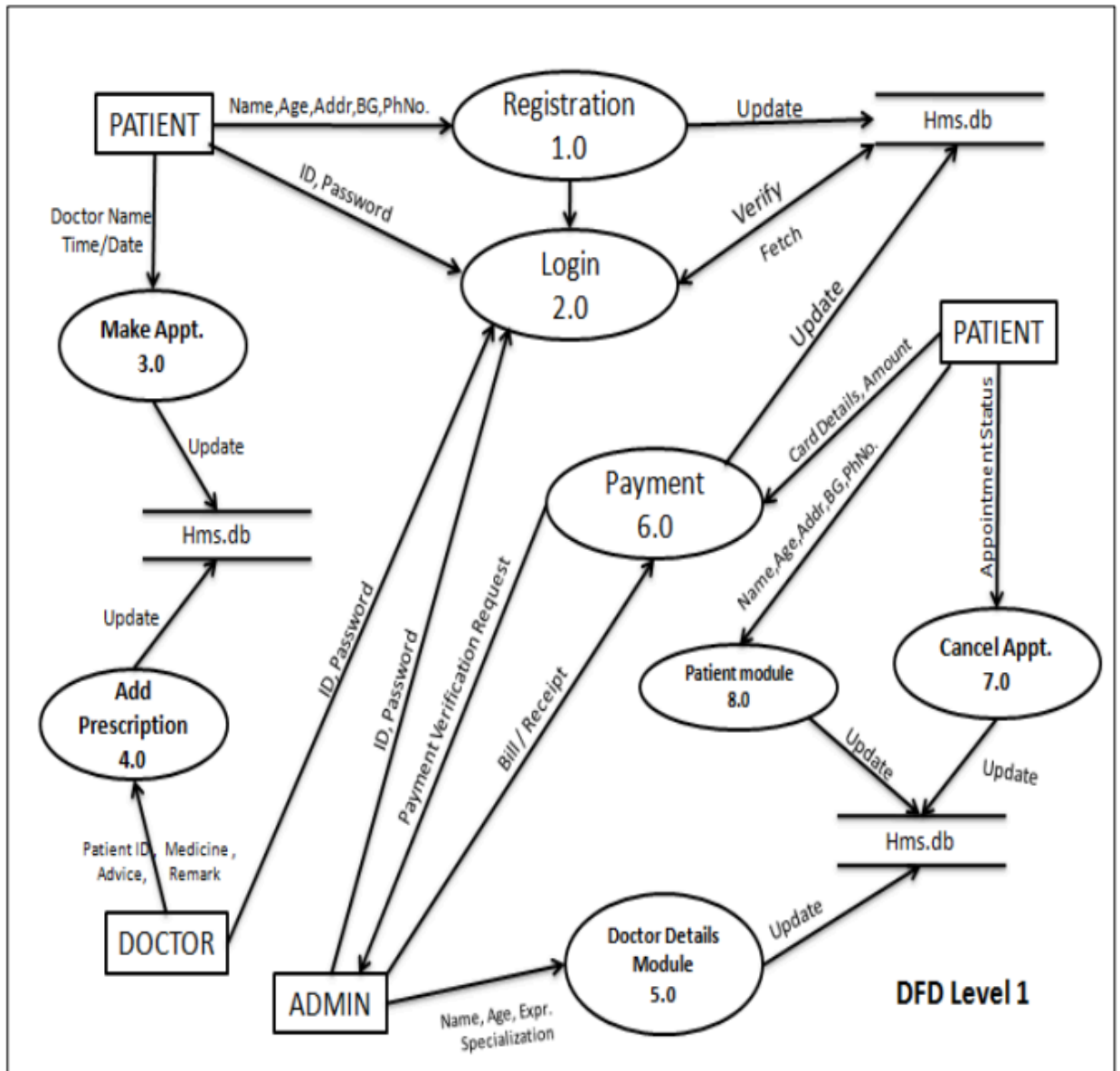


FIGURE 2.2 LEVEL – 1 DFD

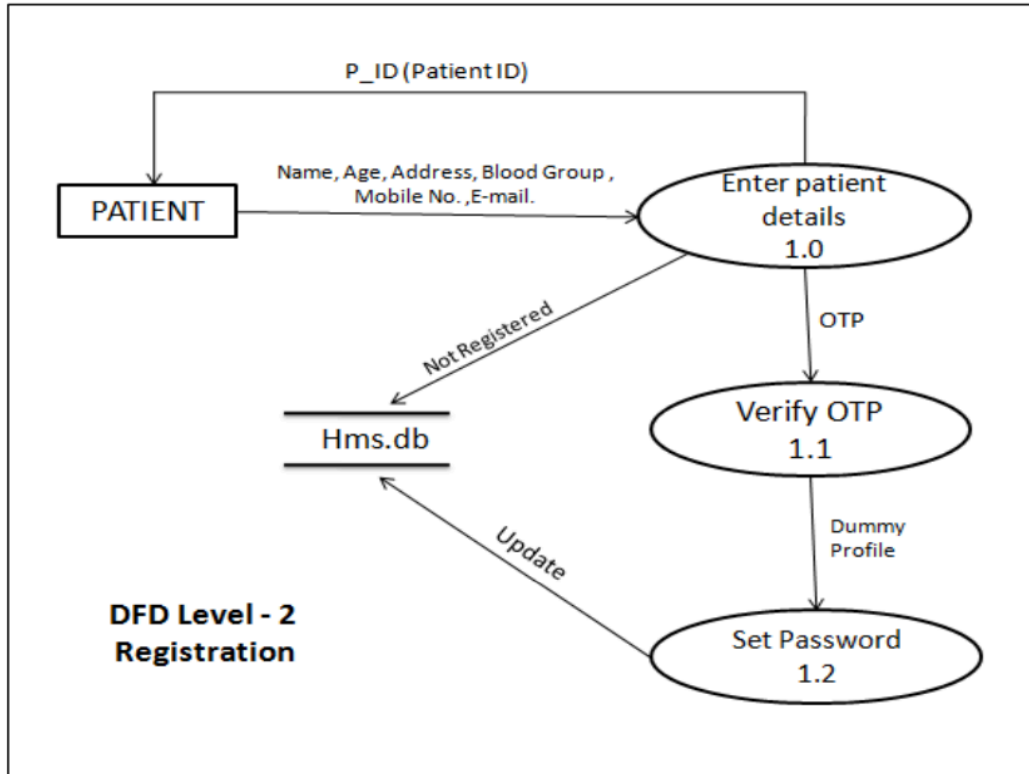
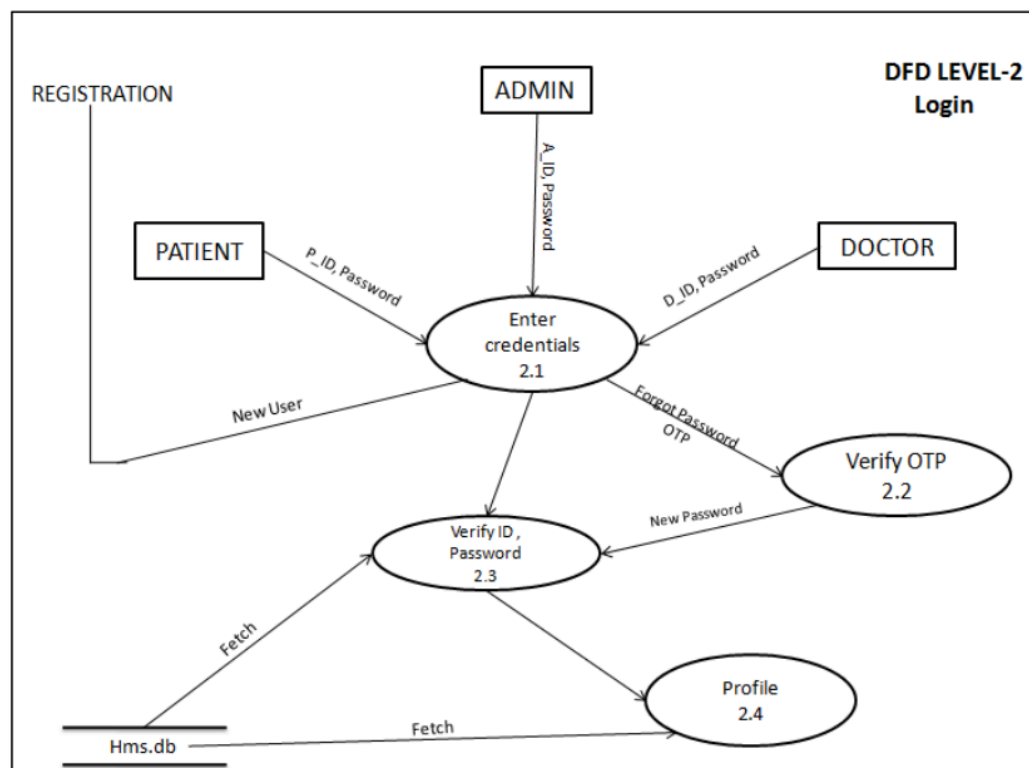


FIGURE 2.3 LEVEL – 2 Registration



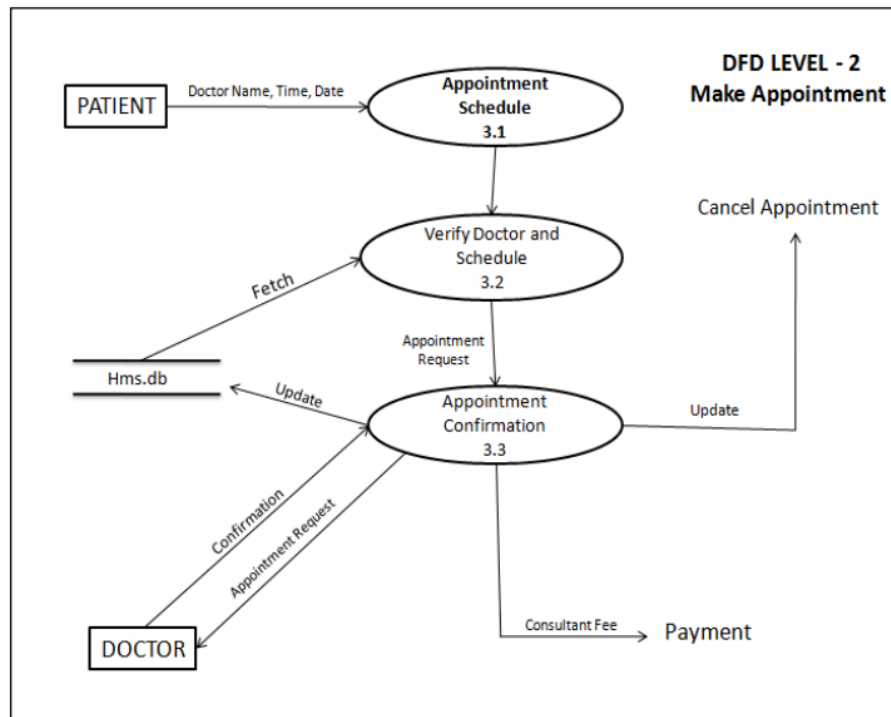


FIGURE 2.5 LEVEL – 2 Make Appointment

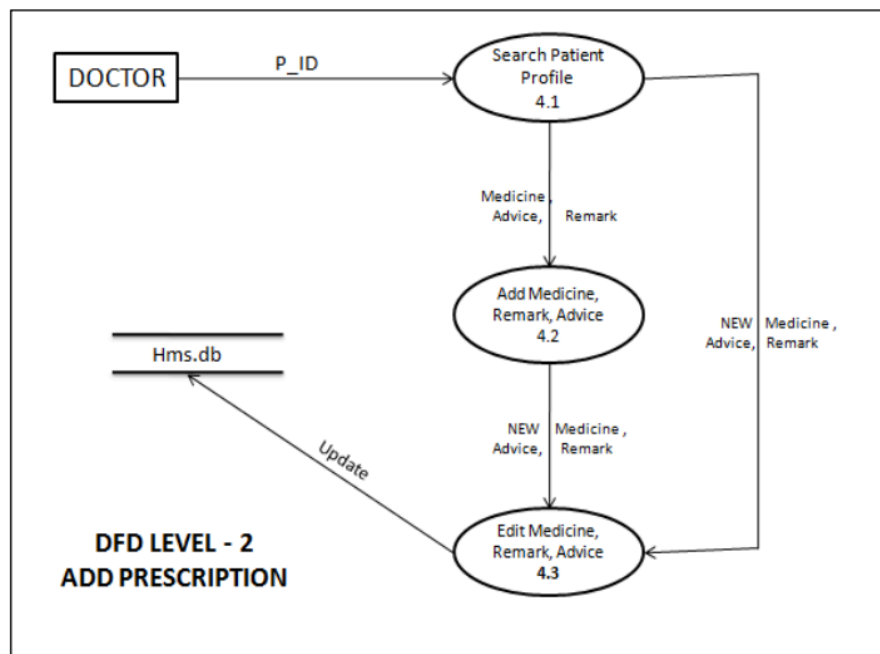


FIGURE 2.6 LEVEL – 2 Add Description

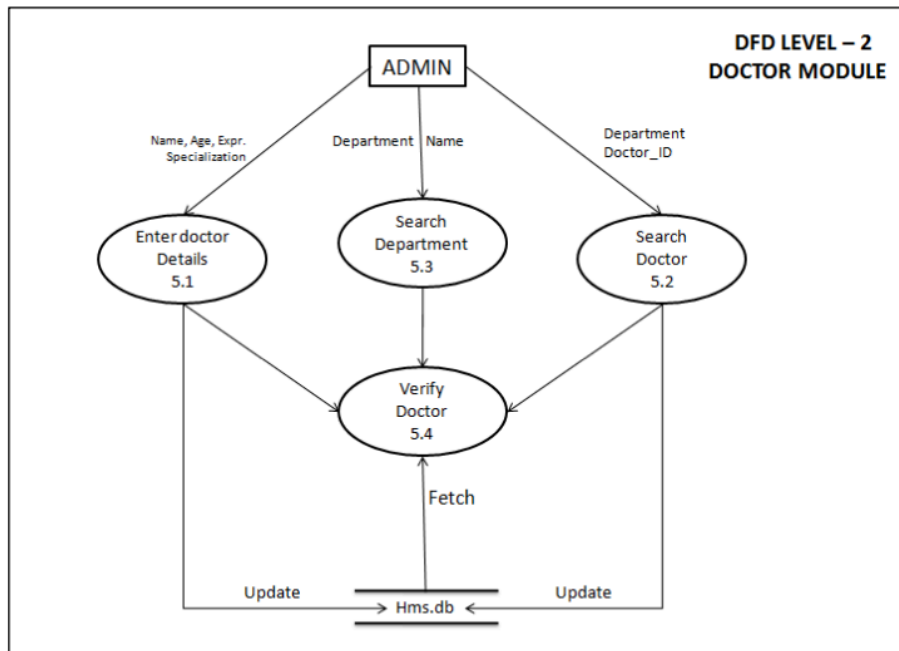


FIGURE 2.7 LEVEL – 2 Doctor Module

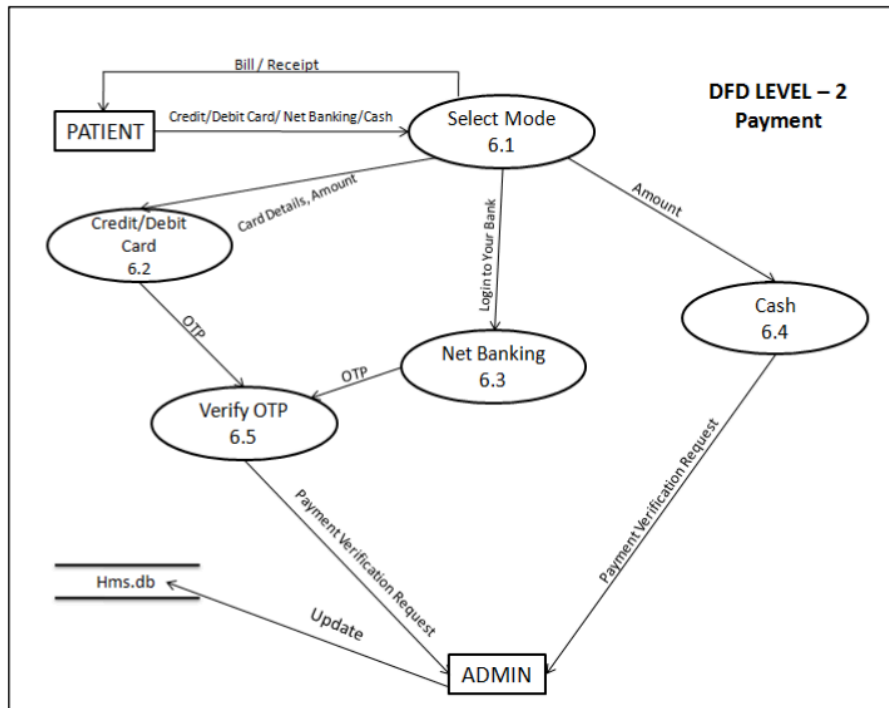


FIGURE 2.8 LEVEL – 2 Payment

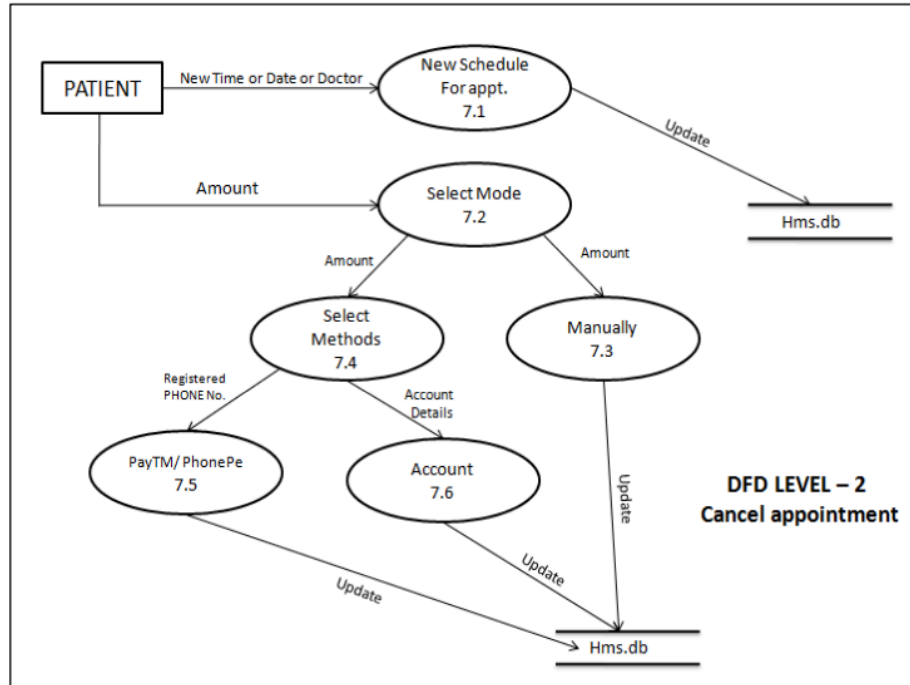


FIGURE 2.9 LEVEL – 2 Cancel Appointment

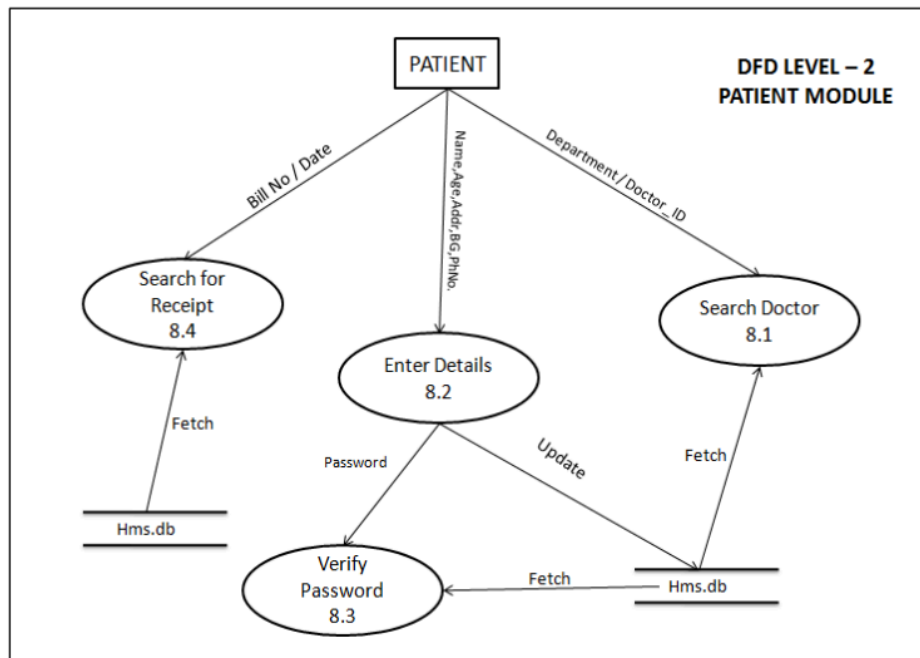
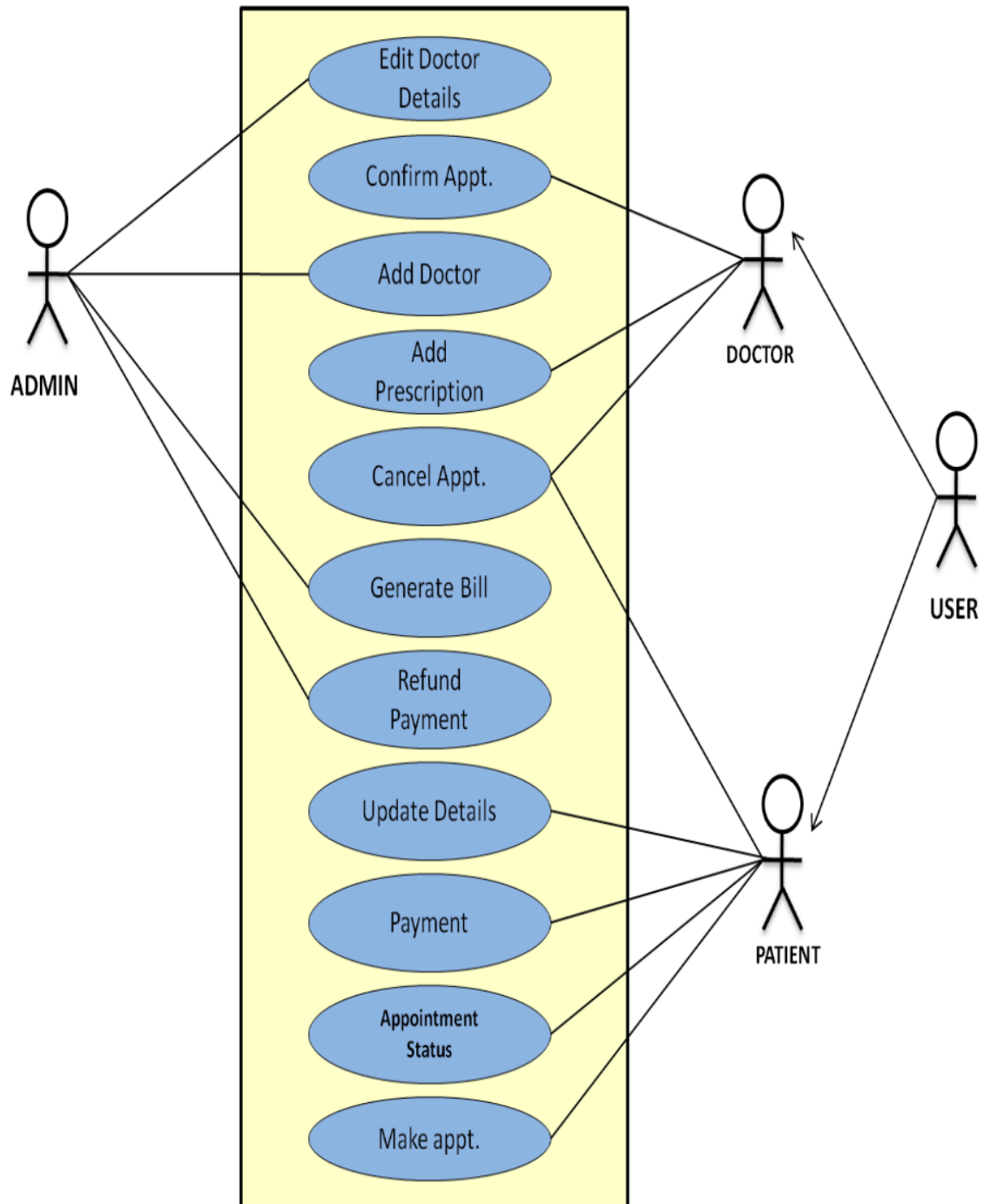


FIGURE 2.10 LEVEL – 2 Patient Module

2.4 USE CASE DIAGRAM



2.5 USE CASE DESCRIPTION

(1) PATIENT

*** REGISTRATION**

DESCRIPTION - The new patient can register themselves and add their details like name, age , gender, blood group etc. The patient entry will be made in the hms database.

PRE -CONDITION – The patient must be a new patient, If necessary fields left by user

then prompt user to fill the necessary fields.

MAIN FLOW OF EVENTS

1. Patient selects sign up in login module.
2. A registration form get displayed
3. Patient fills the required details.

POST CONDITIONS - Patient record is added to hms database.

*** UPDATION**

DESCRIPTION-The patient should be enabled to update his/her details and the changes

should reflect in hms database.

PRE-CONDITION – The patient must be a registered patient, The patient cannot update

details after treatment starts.

MAIN FLOW OF EVENTS

1. Patient logs in to the system.
2. Patient view his record
3. Patient selects update details.
4. Now patient may change the necessary fields.
5. Pop of update details.

POST CONDITION - The record of patient is updated in hms database

CHAPTER 4 SPECIFIC REQUIREMENT

3.1 Performance requirements

3.2 Safety requirements

3.3 Security constraints

3.4 Software system attributes

3.4.1 Usability

3.4.2 Availability

3.4.3 Correctness

3.4.4 Maintainability

3.4.5 Accessibility

3.5 Functional Requirements

3.1 PERFORMANCE REQUIREMENTS

- o Response time-** The system will give responses within 1 second after checking the patient information and other information.
- o Capacity-** The system must support 1000 people at a time
- o User interface-** User interface screen will response within 5 seconds

3.2 SAFETY REQUIREMENTS

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed up log, up to the time of failure. All the administrative and data entry operators have unique logins so system can understand who is login in to system right now no intruders allowed except system administrative nobody cannot change record and valuable data.

3.3 SECURITY REQUIREMENTS

1. Want take the responsibility of failures due to hardware malfunctioning.
2. Warranty period of maintaining the software would be one year.
3. Additional payments will be analyzed and charged for further maintenance.
4. If any error occur due to a user's improper use. Warranty will not be allocated to it.
5. No money back returns for the software.

3.4 SOFTWARE SYSTEM ATTRIBUTES

3.4.1 Usability: Software can be used again and again without distortion.

3.4.2 Availability: The system shall be available all the time.

3.4.3 Correctness: Bug free software which fulfills the correct need/requirements of the client.

3.4.4 Maintainability: The ability to maintain, modify information and update fix problems of the system.

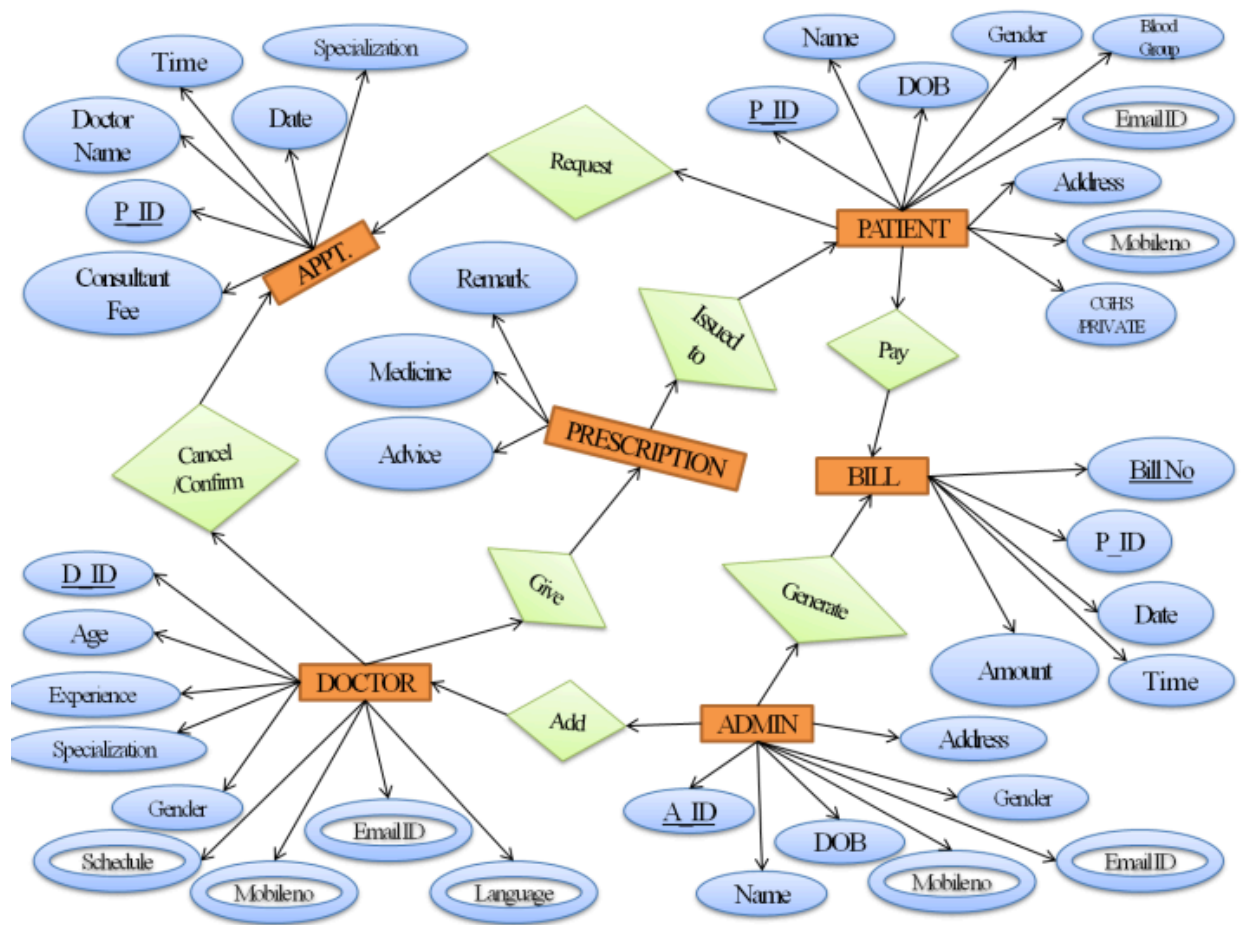
3.4.5 Accessibility: Administrator and many other users can access the system but the access level is controlled for each user according to their work scope.

3.5 FUNCTIONAL REQUIREMENTS

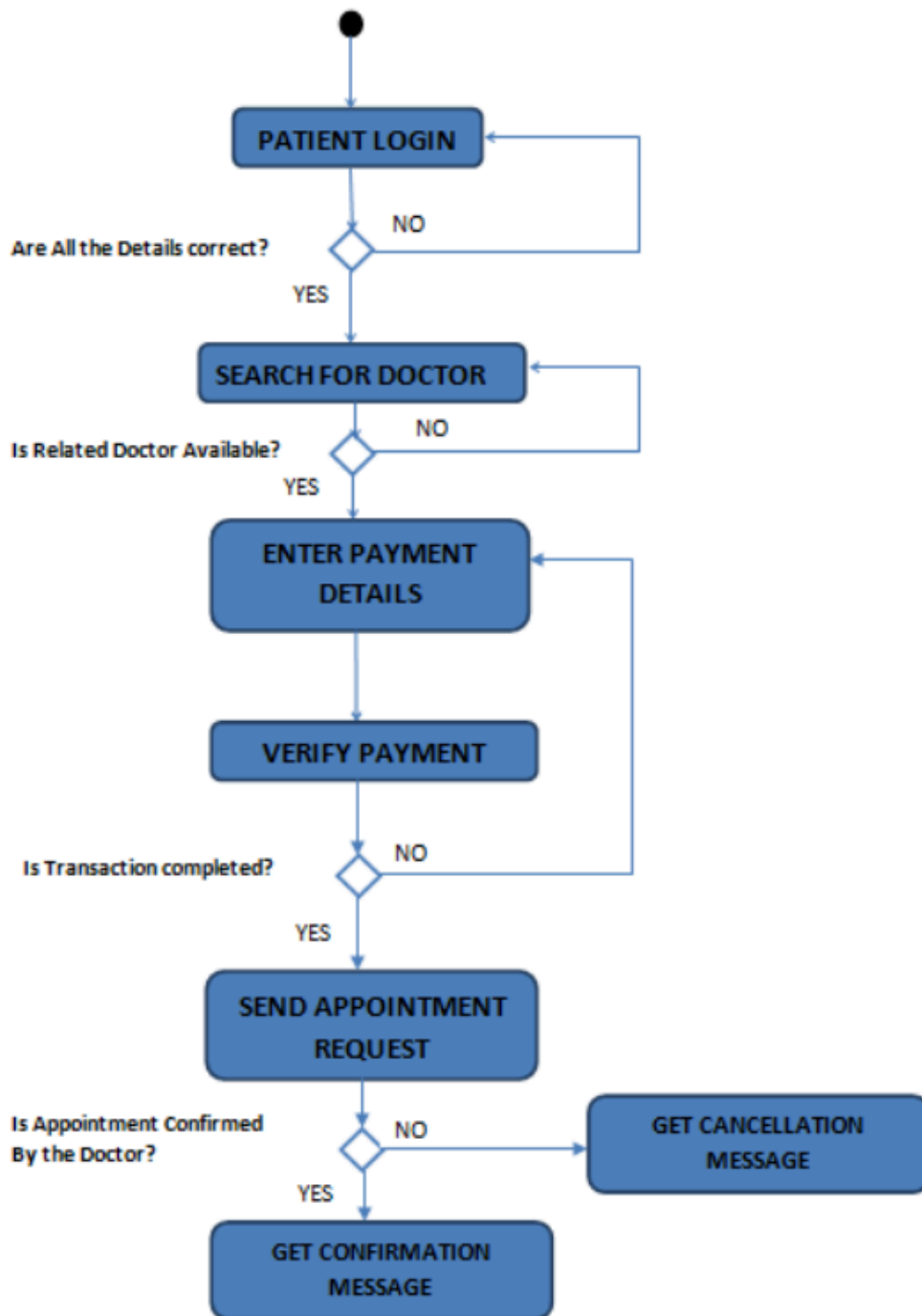
S.No.	MODULE NAME	APPLICABLE ROLES	DESCRIPTION
1.	LOGIN	PATIENT DOCTOR ADMIN	PATIENT: Can login using unique Id and Password after this system shall show his/her profile. DOCTOR: Can login using unique Id and Password after this system shall show his/her profile. ADMIN: Can login using unique Id and Password after this system shall show a profile with links to maintain the website.
2.	REGISTRATION	PATIENT	PATIENT: Can Register by filling all the required details, after this the system will verify the details and check if already registered or not.
3.	MAKE APPT.	PATIENT	PATIENT: Can Select doctor, date time and make an appointment request after this system shall show a confirmation for appointment request.

4.	CANCEL APPT.	PATIENT DOCTOR	PATIENT : Can Cancel appointment if want to by just one click after this system shall ask for re-schedule or refund of payment. DOCTOR : Can Cancel appointment if want to by just one click after this system shall send a message to the patient.
5.	PAYMENT	PATIENT	PATIENT : Enter payment details and make payment after this system shall show the generated bill by the hospital.
6.	DOCTOR MODULE	ADMIN	ADMIN : Can add a new doctor by filling all the details after this system shall show a confirmation message. Can Remove a doctor by just one click after this system shall show confirmation message
7.	PATIENT MODULE	PATIENT	PATIENT : Can view payment history or can search for a particular bill also after this system shall show a bill or history. Can also See or search for a doctor by entering dept. name or doctor id if known after this system will check for the doctor if found shall show doctor's profile. Can also update details after this system shall ask for re-enter password and after verifying password shall update details.
8.	ADD PRESCRIPTION	DOCTOR	DOCTOR : Enter Patient Id and after this all the treatment details and medicine, remark and advice for the patient after this system shall show a message for update.

5 . ER DIAGRAM



6. LEVEL WISE DIAGRAM(DATAFLOW)



Workflow

- Patient:-

- Login/Sign Up
- Input his/her details in form
- Search For Doctor
- Make Appointment
- Making Payment
- Verify Payment

- Doctor:-

- Give Response to the User (Accept/Reject).

Tech-stacks to be Used:-

- For Frontend :- ReactJs, CSS
- For Backend :- NodeJs
- Database :- Google Firebase
- For User Authentication :- Firebase Authentication.