CAREPLUS

Your Care Start Here

TABLE OF CONTENT

1. Introduction of Project

2. Objectives of the Project

3. Scope of the Project

4. Data Flow Diagram (DFD)

5. Entity Relationship Diagram (ERD)

6. Class Diagram

7. Database Structure

8. Modules

9. Validation of The Project

10. Feasibility Study

11. Hardware and Software Requirements

12. Future Scope of The Project

13. Limitations of The Project

14. Bibliography

INTRODUCTION

The **CarePlus** project is a comprehensive healthcare management platform aimed at digitizing and streamlining operations in hospitals, clinics, and other healthcare providers. Its primary goal is to provide seamless management of critical healthcare functions, enhancing the interaction between patients, doctors, and administrators. The system efficiently handles patient appointments, maintains health records, enables doctor consultations, manages prescriptions, and incorporates a robust notification system for timely alerts. The platform is designed with three distinct roles: **Patient, Doctor, and Admin**, each with specific functionalities that contribute to the smooth functioning of the healthcare system.

Key features include:
- **Patients** can schedule, reschedule, or cancel appointments, view their medical history, receive personalized notifications (such as health tips, reminders, and updates), and access detailed information about prescribed medications.
- **Doctors** can manage their schedules, access patient medical records, provide prescriptions, and communicate directly with patients through a chat system. The registration process for doctors also includes an approval workflow overseen by administrators.
- **Admins** can manage the entire system, including verifying new doctor registrations, generating reports, and handling notifications. Admins have access to comprehensive dashboards that include key insights, such as patient-doctor activity, statistical reports, and more.

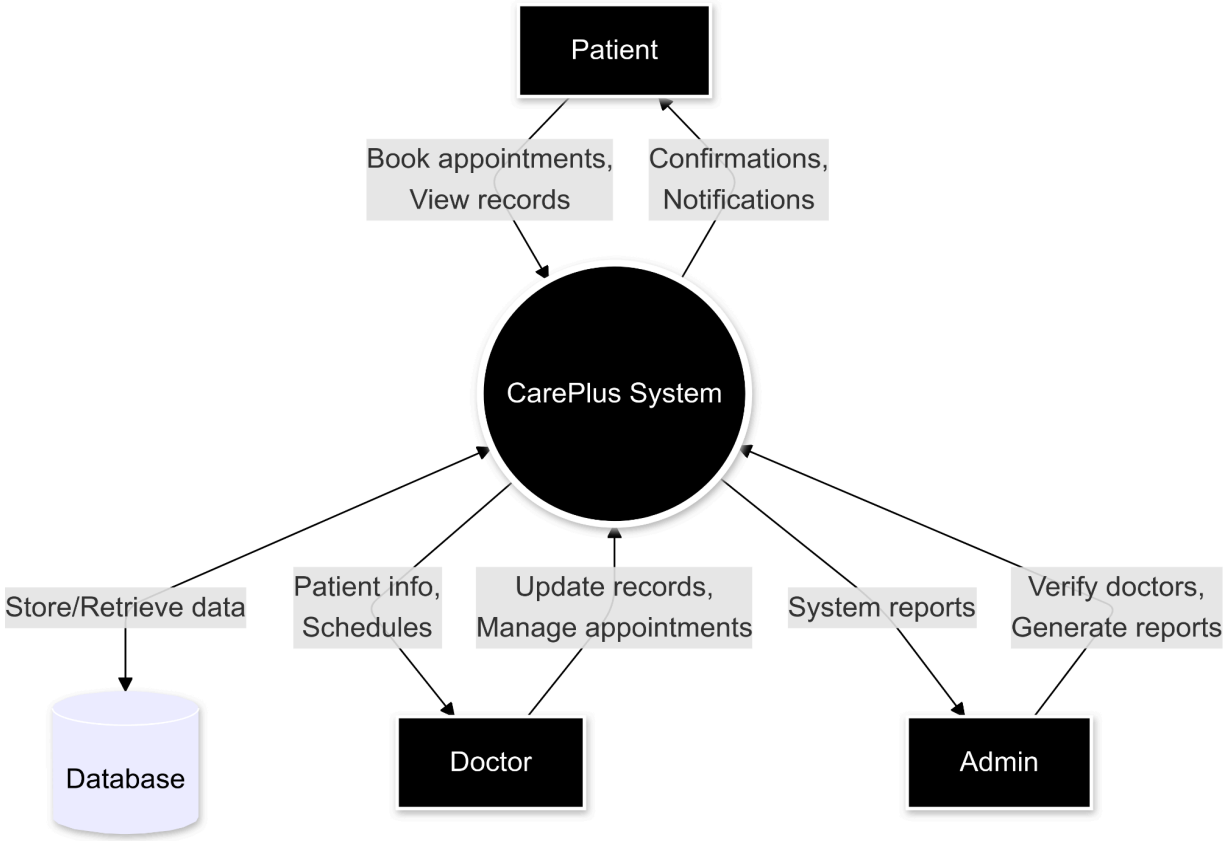
OBJECTIVES OF THE PROJECT

- To create a user-friendly **appointment scheduling** system that allows patients to easily book, reschedule, or cancel appointments, and helps doctors manage their availability.
- To create a seamless appointment scheduling system for patients and doctors.
- To manage patient records efficiently, including medical history, prescriptions, and notifications.
- To provide a platform for doctors to manage consultations and update patient records.
- To enable admins to oversee system operations, including managing doctor registrations and generating reports.
- To ensure the **efficient management of patient records**, including medical history, prescriptions, and notifications, so that both doctors and patients have easy access to vital health information.
- To provide a platform where **doctors can manage their consultations**, update patient records after each visit, and prescribe medications, all in one place.
- To empower **admins to oversee the entire system**, ensuring smooth operations by managing doctor registrations, verifying their credentials, and handling administrative tasks.

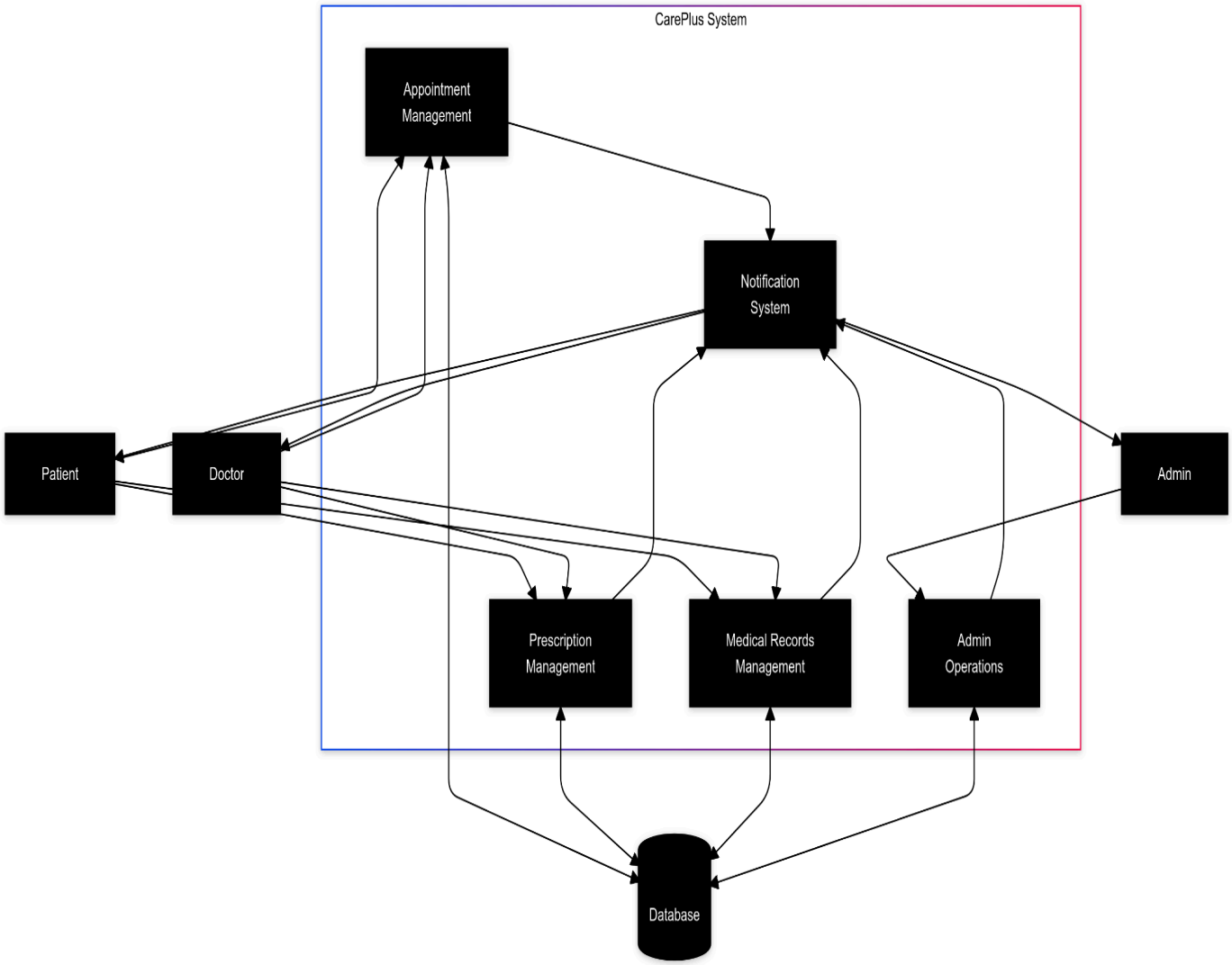
SCOPE OF THE PROJECT

- **Appointment Management**: Patients can book, reschedule, or cancel appointments, while doctors can manage their availability and view upcoming appointments.
- **Prescription Management**: Doctors can prescribe medications, and patients can view prescription details, including dosage and medication information.
- **Medical Records**: The system securely stores and updates patient medical history, accessible to both patients and doctors, ensuring up-to-date health data.
- **Notification System**: Automated notifications are sent to patients, doctors, and admins for appointments, prescription updates, and system alerts.
- **Doctor Registration and Verification**: Admins manage the doctor registration process, verifying credentials before granting access to the system.
- **Admin Dashboard and Reports**: Admins can generate reports on system activity, including appointment statistics and user data, with the ability to export reports in PDF and Excel formats.
- **Role-Based Access**: Each user role (Patient, Doctor, Admin) has access to specific features tailored to their needs.

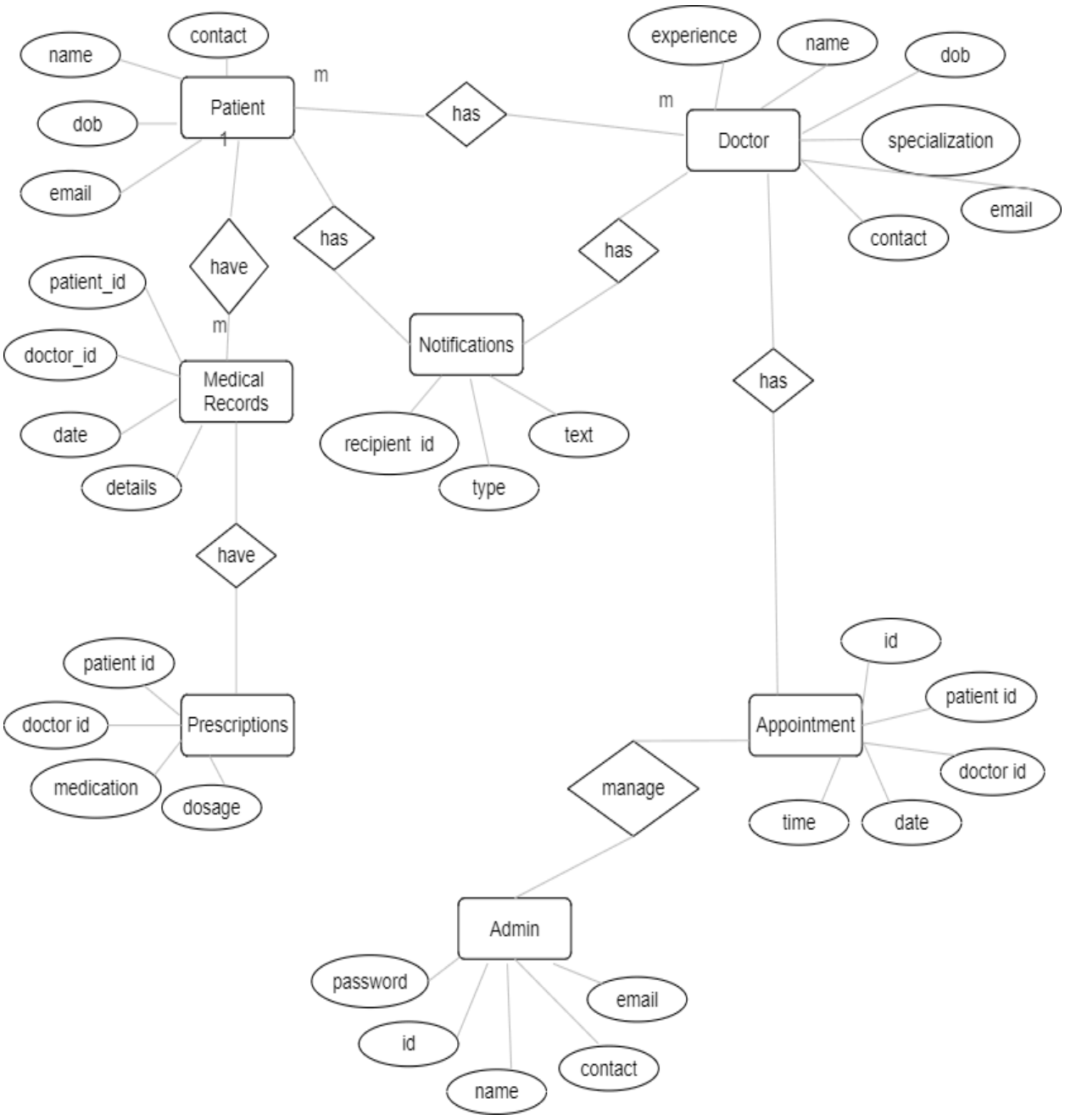
CONTEXT DFD (LEVEL 0)



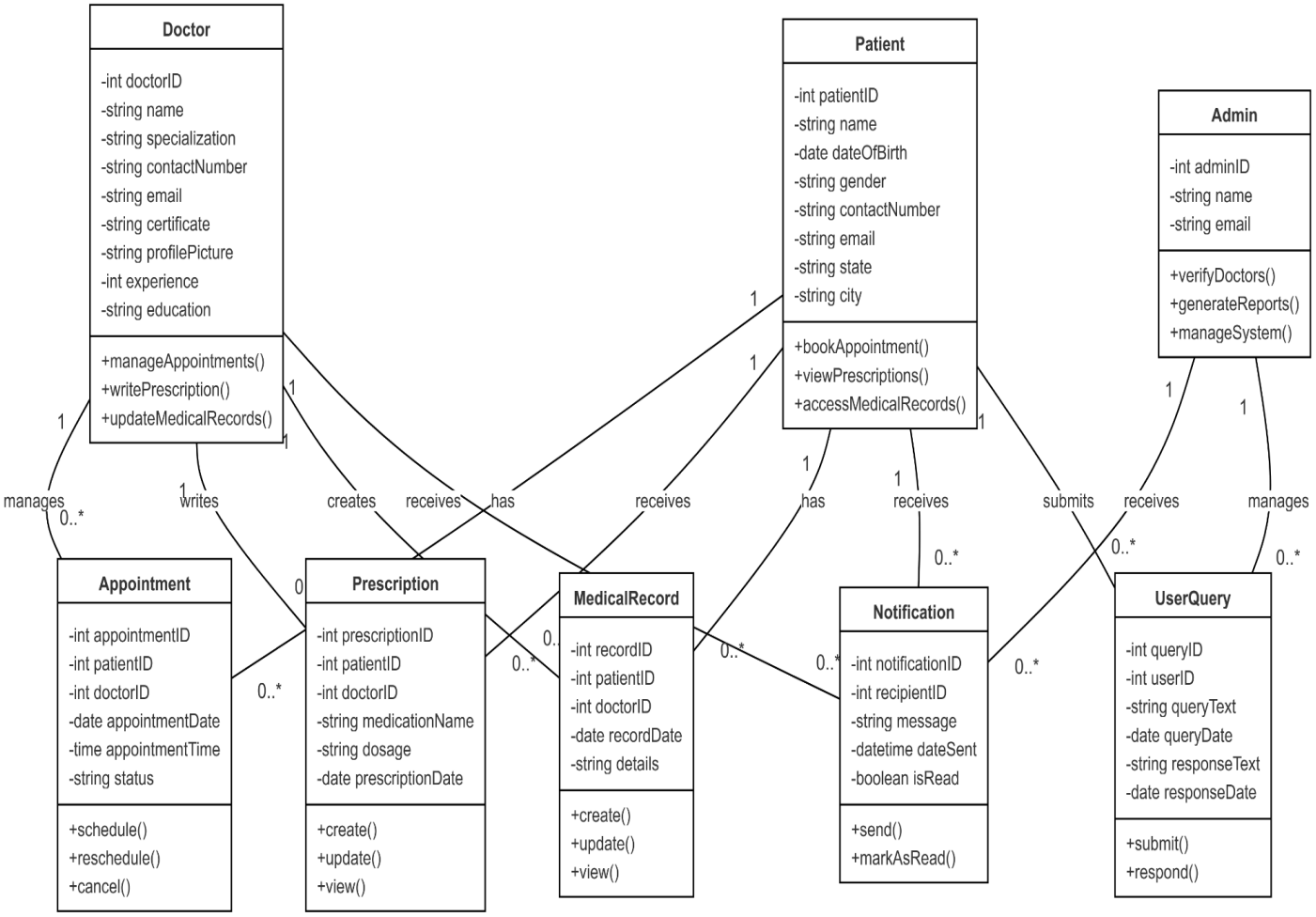
LEVEL 1 DFD DIAGRAM



ENTITY RELATIONSHIP DIAGRAM (ERD)



CLASS DIAGRAM



DATABASE STRUCTURE

Table: patients →

Field Name	Type	Constraints	Descriptions
id	int	Primary key	User Identity
name	varchar(50)	Not null	name
email	varchar(100)	Unique	email
dob	Date	Not null	dob
contact	int	Not null	Phone number
password	varchar(255)	Not null	encrypted
gender	varchar(20)	Not null	enum

Table: doctors→

Field Name	Type	Constraints	Descriptions
id	int	Primary key	User Identity
name	varchar(50)	Not null	name
email	varchar(100)	Unique	email
dob	Date	Not null	dob
contact	int	Not null	Phone number
password	varchar(255)	Not null	encrypted
gender	varchar(20)	Not null	enum
specialization	varchar(50)	Not null	enum

education	varchar(50)	Not null	Education detail
experience	varchar(50)	Not null	experience

Table: admin→

Field Name	Type	Constraints	Descriptions
id	int	Primary key	User Identity
name	varchar(50)	Not null	name
email	varchar(100)	Unique	email
contact	int	Not null	Phone number
password	varchar(255)	Not null	encrypted

Table: notifications→

Field Name	Type	Constraints	Descriptions
id	int	Primary key	Identity
recipient_id	int	Foreign key	id
recipient_type	varchar(100)	Unique	type
message	text	Not null	message
date	Date	Not null	Created date
isRead	Boolean	Not null	Read or not

Table: appointments→

Field Name	Type	Constraints	Descriptions
id	int	Primary key	Identity
patient_id	int	Foreign key	patient
doctor_id	int	Foreign key	doctor
appointment_date	Date	Not null	date
appointment_time	Time	Not null	time
status	varchar(15)	Not null	status

Table: medical_records →

Field Name	Type	Constraints	Descriptions
id	int	Primary key	Identity
patient_id	int	Foreign key	patient
doctor_id	int	Foreign key	doctor
date	Date	Not null	date
details	text	Not null	Appoint. overview

Table: prescriptions

Field Name	Type	Constraints	Descriptions
id	int	Primary key	Identity
patient_id	int	Foreign key	patient

doctor_id	int	Foreign key	doctor
medication_name	varchar(100)	Not null	Medicine name
dosage	varchar(100)	Not null	Dosage details
created_at	Date	Not null	date

Table: queries →

Field Name	Type	Constraints	Descriptions
id	int	Primary key	Identity
name	varchar(100)	Not null	name
email	varchar(100)	Not null	Contact info.
created_at	Date	Not null	date
query	text	Not null	Query details

MODULES AND THEIR DESCRIPTIONS

1. **Patient Module:**

- Appointment Management :-
    - Book, reschedule, and cancel appointments.
    - View appointment history and upcoming schedules.
- Prescription Management:-
    - View prescriptions, including medication name, dosage, and instructions.
    - Access detailed information about prescribed drugs.
- Notifications:-
    - Receive appointment confirmations, reminders, and health updates.
    - Mark notifications as read and categorize them for future reference.

2. **Doctor Module:**

- Appointment Management:-
    - View, accept, reschedule, or cancel appointments.
    - Access patient details and medical history before consultations.
- Patient Record Management:-
    - Update patient medical records and attach relevant documents.
    - Add diagnosis, notes, and treatment plans directly to the system
- Prescription Management:-

- Create and manage prescriptions with medication details.
- Review and renew previous prescriptions as needed.

3. **Admin Module:**

- Doctor Management:-
  - Verify and approve doctor registrations.
  - Suspend or deactivate doctor accounts if necessary.
- Appointment Supervision:-
  - Monitor and manage appointments across the system.
  - Resolve scheduling conflicts and notify affected users.
- Report Generation:-
  - Generate and export reports on appointments, prescriptions, and system performance.
  - Create custom reports based on specific criteria or time periods.
- Notification Control:-
  - Send system-wide notifications and announcements.
  - Manage the notification log for timely delivery.

VALIDATION OF THE PROJECT

- **User Authentication:**
  - Role-based access control for Patients, Doctors, and Admins.
  - Secure login with proper credential validation and password protection.
- **Data Integrity:**
  - Validations for appointment scheduling, prescriptions, and medical records.
  - Ensures correct data format for inputs like dates, dosages, and patient details.
- **Error Handling:**
  - User-friendly error messages for failed actions (e.g., incorrect appointment rescheduling or data entry errors).
  - Real-time validation to prevent invalid data submissions (e.g., incorrect prescription info).
- **Input Validation:**
  - Ensures mandatory fields are filled (e.g., appointment time, patient ID).
  - Restricts invalid inputs and provides feedback for corrections (e.g., invalid date formats).
- **Session Management:**
  - Tracks user sessions, ensuring secure navigation between roles and maintaining session integrity.

- **Notifications:**
    - Validates that notifications are sent to the correct user roles (Patients, Doctors, Admins) based on events like appointment changes or system updates.

FEASIBILITY STUDY

- **Technical Feasibility:**
  - Utilizes modern, scalable technologies like **React** for frontend, **Node.js** for backend, and **MySQL** for database management.
  - Easily deployable across various platforms with robust performance and security.
- **Operational Feasibility:**
  - Designed with a user-friendly interface for Patients, Doctors, and Admins.
  - Streamlines healthcare workflows, including appointment scheduling, record management, and notifications.
  - Minimal training required for users to effectively operate the system.
- **Economic Feasibility:**
  - Cost-efficient development, using open-source tools and minimal hardware.
  - Reduces administrative overhead for healthcare facilities by automating processes like appointment management and report generation.
  - Long-term savings through reduced manual errors and improved operational efficiency.

SOFTWARE REQUIREMENTS

- **Hardware:**
  - Processor: Intel i3 or above
  - RAM: 4GB minimum
  - Hard Disk: 50GB free space
- **Software:**
  - Frontend: React.js
  - Backend: Node.js, Express
  - Database: MySQL
  - Other: Git, Visual Studio Code

FUTURE SCOPE

- **AI Integration for Predictive Analytics**:
  - Implementing machine learning algorithms to analyze patient data and predict potential health risks or treatment outcomes.
  - Enabling doctors to make data-driven decisions, providing personalized healthcare recommendations based on historical data and trends.
- **Telemedicine Features for Online Consultations**:
  - Adding video consultation capabilities to allow patients to consult doctors remotely.
  - Integration of chat systems for instant messaging between patients and doctors, enabling real-time communication and remote diagnosis.
- **Mobile App Extension**:
  - Developing a mobile version of the *CAREPLUS* system for ease of use by patients and doctors.
  - Providing push notifications for appointment reminders, prescription updates, and urgent health tips directly to users' smartphones.
- **Automated Billing and Insurance Management**:
  - Introducing features for automating patient billing processes, integrating with insurance providers for claim management.
  - Reducing manual errors and improving the financial workflow within the healthcare facility.

LIMITATIONS OF THE PROJECT

- **Single Facility Support**: Supports only 1 clinic/hospital at a time.
- **AI Integration**: Less than 10% of functionalities use AI-driven analytics.
- **Internet Dependency**: 100% of core features require an active internet connection.

BIBLIOGRAPHY

1. Pressman, R. S. *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education, 2019.
2. Online tutorials and resources from platforms like MDN, W3Schools, and freeCodeCamp on Node.js, React.js, and MySQL.
3. Official documentation of React.js, Node.js, and MySQL technologies from their respective websites.
4. Stack Overflow and GitHub discussions for real-world problem-solving and implementation guidance.