

## PROJECT ABSTRACT

# Hospital Management System (MediCare+)

Hospitals run on coordinated workflows: registration and scheduling, doctors' daily queues, front-desk billing and pharmacy stock control.

---

## 01 Title

Hospital Management System (MediCare+)

## 02 Introduction / Background

Hospitals run on coordinated workflows: registration and scheduling, doctors' daily queues, front-desk billing and pharmacy stock control. MediCare+ digitizes this backbone in PHP/MySQL: online appointment booking, patient records, doctor schedules, e-prescriptions, billing and pharmacy stock, organized around four permission-gated roles — admin, doctor, receptionist and patient.

## 03 Problem Statement

Most academic hospital projects lack double-booking protection, connected pharmacy and billing. MediCare+ connects the patient journey: slot booking with a uniqueness constraint on doctor, date, slot; a receptionist desk with queue tokens; e-prescriptions flowing into the pharmacy module, which dispenses and decrements stock; billing consolidating consultation and pharmacy charges into one itemized bill.

## 04 Objectives

- Online appointment booking with department filters, doctor availability and slot-based scheduling.
- Patient registration with unique IDs and visit histories; doctor profiles.
- Role-separated dashboards for patient, receptionist, doctor and admin — each permission-gated.
- Pharmacy stock management: inventory, dispensing against prescriptions, low-stock alerts.
- Billing: itemized bills combining consultation and pharmacy charges, with payment recording.

## 05 Proposed System / Working

PHP web app with role-separated controllers/views, HTML/CSS/JS with Bootstrap, MySQL/InnoDB. Foreign keys enforce integrity across the patient → appointment → prescription → billing chain. Session-based authentication with bcrypt hashing and role middleware on every route enforces least privilege. Key tables: users, roles, patients, doctors, departments, doctor\_schedules, appointments, prescriptions, prescription\_items, medicines, pharmacy\_stock, bills, bill\_items,

payments. The appointment engine books slots against the unique constraint; the pharmacy module decrements a stock ledger on dispensing; bills aggregate charges server-side; every record timestamped and linked.

The demo runs the patient journey: a patient registers and books a doctor and slot; the receptionist confirms and issues a queue token; the doctor records a diagnosis with prescription; pharmacy dispenses and decrements stock; billing consolidates consultation and pharmacy charges into one bill. Admins view collection and appointment reports. Tests cover slot-conflict prevention, role authorization, bill-total math and stock integrity.

## 06 Technologies / Components Used

PHP · MERN stack · HTML · CSS · JavaScript · MySQL · Session-based authentication · XAMPP/localhost

## 07 Key Features

- Appointment booking with doctor availability, department filter and time slots.
- Patient registration with unique ID, medical history and visit records.
- Doctor dashboard: daily schedule, appointment queue, patient history, e-prescriptions.
- Receptionist desk: walk-in registration, scheduling, queue tokens and billing counter.
- Billing: consultation charges, itemized bills, payment recording, printable receipts.
- Pharmacy stock: inventory ledger, dispensing against prescriptions, low-stock alerts.
- Reports: daily collections, appointment summaries, patient visit history.

## 08 Applications / Use Cases

- Small clinics and single-branch hospitals still on paper registers

## 10 Conclusion

MediCare+ is an enterprise web application: four roles, a slot engine with concurrency safety, a connected prescription-to-pharmacy-to-billing chain and audit-friendly records on commodity PHP/MySQL hosting. The familiar domain and workflows make it one of the safest viva topics.