Hospital Management System

Course: Information Systems — CMPn325

Under Supervision: Dr. Khaled Morsy

T.A Hadeer Kamal

Students:

ID	Name	Section
4240102	John Emil Ramzy	**TH-5**
4240624	Abdullah Mohamed Ahmed	**TH-7**
4240622	Hazem Mohamed Abd Elhaseb	**TH-7**
4240223	Belal Ahmed Mahmoud	**TH-4**
4240233	Mohamed Abd ElNasser Galal	**TH-5**
4240225	Youssef Omar Said	**TH-7**

Level: Level 2 Computer

Term: First Term: 2025–2026

Abstract

The Hospital Management System is a web-based application designed to streamline hospital operations through a centralized digital platform. The system provides secure access through user authentication, including sign up and login functionality, and offers separate dashboards for patients, doctors, and administrators.

Patients can request appointments with doctors and view their medical information. Doctors can manage their schedules and add prescriptions for patients after consultations. The administrator oversees the entire system, with the ability to add and manage doctors, patients, appointments, prescriptions, and medicines. The system enhances efficiency, reduces human error, and improves communication within the hospital.

Table of Contents

4. Outline of Systems Study

4.1 Objectives of the Study

- Identify current issues in patient registration, appointments
- Collect user requirements to design an appropriate information system that supports hospital activities.
- Implement secure role-based access for all users using JWT tokens.

4.2 Scope

The scope of the system includes internal hospital operations covering patient management, appointments, medical staff, laboratory services, pharmacy. The system provides secure access using **JWT-based authentication**, ensuring only authorized patients, doctors, and administrators can access their respective dashboards and features.

---

5. Detailed Results of Systems Study

5.1 Current System Description

The proposed Hospital Management System is a web-based application providing separate dashboards for patients, doctors, and administrators. Users access the system through sign up and login pages, and authentication is secured via **JWT tokens**.

- **Patient Dashboard:** Allows patients to view their profile and request appointments with doctors.
- **Doctor Dashboard:** Allows doctors to view their appointments and add prescriptions for patients.
- **Admin Dashboard:** Allows administrators to manage doctors, patients, appointments, prescriptions, and medicines.

The system ensures secure, role-based access and centralizes hospital data, improving efficiency and communication.

5.2 Problems and Opportunities

- Lack of centralized system for managing appointments.
- No direct interaction between patients and doctors for appointment scheduling.

- Manual prescription documentation causes errors.
- No role-based dashboards for system users.
- Difficulty managing hospital data such as medicines and registered users.

---

6. Functional Requirements

6.1 Input Requirements

- Patient registration via sign up.
- Login credentials for authentication (JWT).
- Appointment requests submitted by patients.
- Prescription details entered by doctors.
- Doctor, patient, medicine, and appointment data entered by admin.

6.2 Output Requirements

- Patient dashboard displaying appointments and profile information.
- Doctor dashboard displaying scheduled appointments and patient details.
- Admin dashboard showing system statistics and management tools.
- Prescription records.
- Medicine lists.

6.3 Processing Requirements

- Authenticate users using **JWT tokens**.
- Authorize access based on user roles (patient, doctor, admin).
- Process patient appointment requests.
- Allow doctors to create prescriptions.
- Allow admin to add doctors, patients, appointments, prescriptions, and medicines.
- Update dashboards dynamically.

6.4 Storage Requirements

- Relational database storage (Oracle DB)

6.5 Control Requirements

- Role-based access control using JWT.
- Audit logging for all data changes.

- Encryption of sensitive patient data.
- Input validation mechanisms to ensure data integrity.

---

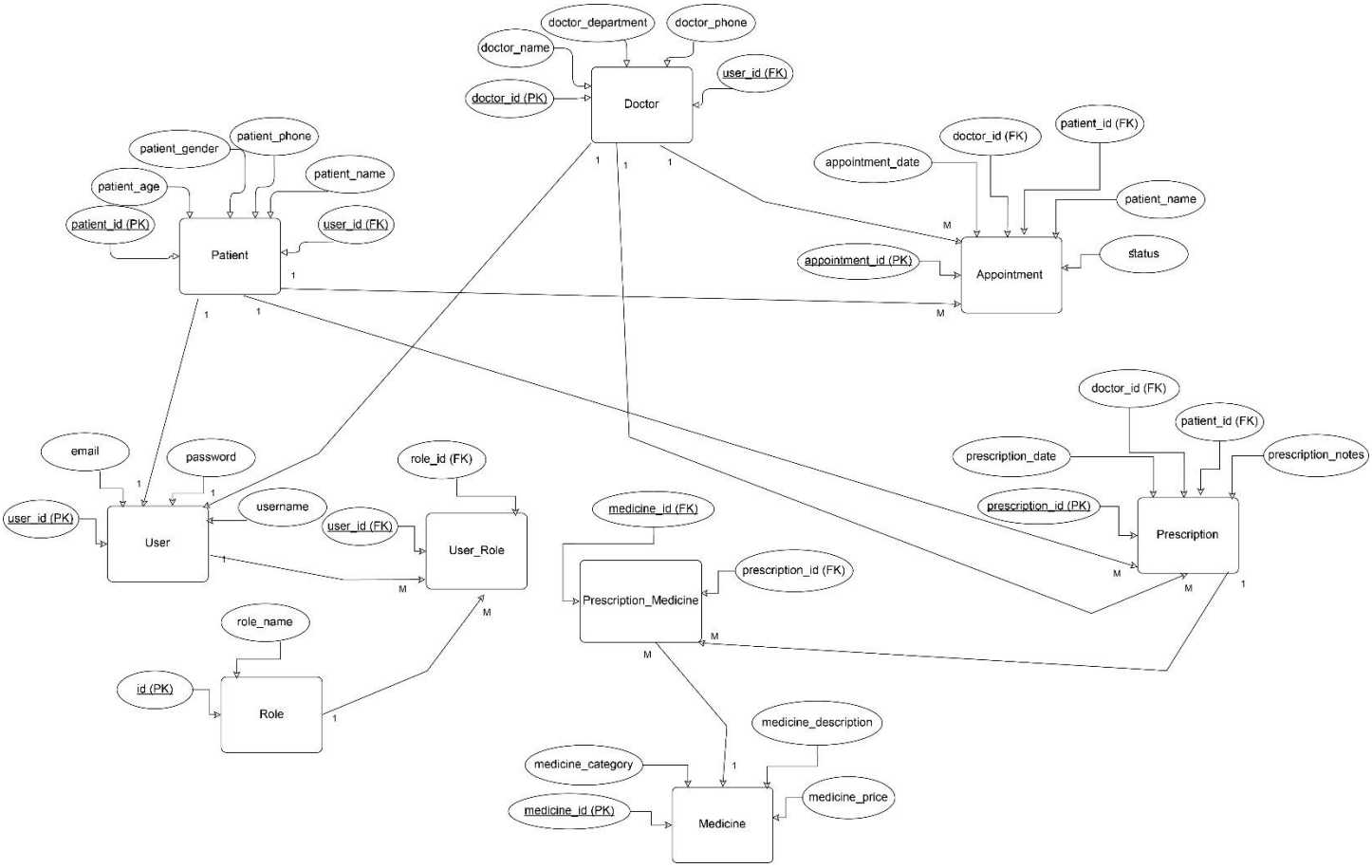
7. System Design

7.1 User Interface Design

- Sign up page.
- Login page.
- Patient dashboard: view profile, request appointments.
- Doctor dashboard: view appointments, add prescriptions.
- Admin dashboard: add doctor, patient, appointment, prescription, medicines.

7.2 Data Design

- ERD to represent patients, doctors, appointments, prescriptions and medicines



- Tables include primary keys, foreign keys, and relationships.
- Database supports role-based access and **JWT authentication**.

7.3 Process Design

- User signup and login using **JWT authentication**.
- Patient adds appointment.
- Admin adds appointments.
- Doctor adds prescriptions for patients.
- Admin manages doctors, patients, appointments, prescriptions, and medicines.
- Dashboards update dynamically based on user role.

---

8. System Specifications

- **User Interface:** Dashboards for patients, doctors, and admins, login/sign up pages.
- **Database:** Relational tables with relationships, primary/foreign keys (Oracle DB).
- **Software:** Backend + frontend, REST APIs, JWT authentication.
- **Hardware:** Server for app and database.
- **Personnel:** Admins, doctors, staff, developers.
- **Documentation:** User manuals and system administrator guides.

---

9. Implementation Plan

- Develop frontend and backend modules.
- Implement JWT authentication for secure access.
- Create patient, doctor, and admin dashboards.
- Build database with all necessary tables and relationships.
- Test appointment scheduling, prescription creation, and admin management functionalities.
- Deploy the system and provide user training.

---

10. Testing & Acceptance Criteria

- Users can sign up and log in securely with JWT.
- Patients can request appointments successfully.
- Doctors can add prescriptions for their patients.
- Admin can manage all users, appointments, prescriptions, and medicines.
- Dashboards reflect real-time data updates.
- Data security and validation are enforced.
