Patient Record Keeping Web App

# Project Name:

Patient Record Keeping Web App

# Your Name:

[Abukar Abdirahman Ibrahim]

# Date:

[Date]

# Course/Institution:

[ASP.NET CORE MVC]

# Introduction

Overview of the Project: The Patient Record Keeping Web App is designed to provide a streamlined solution for healthcare providers to manage patient records, appointments, prescriptions, billing, and other medical data. It is a comprehensive platform that enables medical professionals to handle critical data efficiently and securely. The app is also designed with a user-friendly interface for easy navigation and interaction.

Importance: This web app plays a crucial role in simplifying the healthcare workflow. It helps medical institutions save time and resources by centralizing patient information, managing appointments, and generating essential reports. The app aims to improve the overall experience for both healthcare providers and patients.

Technologies Used: This project is developed using ASP.NET Core MVC for the backend, with a SQL Server database for storing data. The frontend is built with HTML, CSS, and JavaScript, ensuring a responsive and user-friendly interface.

# System Architecture

High-Level Overview: The system is based on a client-server architecture, with a clear separation of concerns. It follows the Model-View-Controller (MVC) architecture, where the data (Model), user interface (View), and application logic (Controller) are separated into different components to improve maintainability.

Components:   
1. Models: Includes entities such as Patient, Doctor, Appointment, Prescription, Billing, and Reports, each with its corresponding database table and logic.  
2. Views: Provides user interfaces for interacting with the system, including landing page, admin dashboard, and forms for managing patient, doctor, and other records.  
3. Controllers: Handle HTTP requests, manipulate data, and provide responses, ensuring proper interaction between the user interface and the database.

# Features

## Patient Module

The Patient Module allows medical staff to manage patient data. This includes creating new patient records, viewing existing records, updating information, and deleting records when necessary. The module ensures that patient information is stored securely and can be easily accessed by authorized personnel.

## Doctor Module

The Doctor Module enables users to manage doctor information. It supports CRUD operations for adding, viewing, editing, and deleting doctor records. This module ensures that doctors' credentials and contact information are up-to-date.

## Appointment Management

This feature allows users to schedule, view, update, and delete appointments. It ensures that both patients and doctors can manage their schedules effectively, minimizing errors and conflicts.

## Prescription Management

The Prescription Module allows doctors to create and manage prescriptions for patients. This includes prescribing medications, specifying dosages, and adding additional instructions for the patient's treatment.

## Billing

The Billing feature manages the invoicing process. It allows administrators to generate invoices for services rendered, track payments, and ensure accurate billing records.

## Reports

The Reports feature provides various reporting tools to help administrators generate valuable insights, such as patient treatment reports, billing statements, and appointment statistics.

## Admin Layout

The Admin Layout provides a centralized view for managing all aspects of the system. Administrators have the ability to view and modify patient and doctor data, manage appointments, and generate reports.

# User Authentication and Authorization

Login System: The login system allows users (patients, doctors, admins) to authenticate securely using their credentials. A secure login process ensures that only authorized users can access the system.

Register System: New users can register an account by providing necessary details. Registration is required for patients, doctors, and admins to gain access to their respective features.

Forgot Password: If users forget their passwords, the system provides a secure password recovery process to ensure they regain access to their accounts.

Role-Based Access Control (RBAC): The system implements role-based access control to ensure that users can only access features relevant to their role. For example, patients can only access their medical records, while admins can manage all aspects of the system.

# User Interface and Experience

Admin Dashboard: The Admin Dashboard provides a central hub for managing all the system's features. It allows admins to access patient records, manage appointments, generate reports, and more.

Landing Page: The Landing Page serves as the entry point for users, providing basic information about the application, with links to login, register, and access other system functionalities.

Responsive Design: The application is designed to be responsive, ensuring that it works seamlessly across a variety of devices, from desktop computers to mobile phones.

# Database Design

Tables: The app includes several key database tables, including Patient, Doctor, Appointment, Prescription, Billing, and Reports. Each table stores relevant information and is associated with one or more models in the system.

Relationships: There are several relationships between the tables. For example, the Patient table has a one-to-many relationship with the Appointment table, meaning a single patient can have multiple appointments.

Data Integrity: Constraints and validations are applied to ensure data integrity. For example, a prescription cannot be created without an associated patient and doctor, and appointment dates must be valid.

# Security Features

Password Security: Passwords are securely hashed using industry-standard algorithms to ensure they cannot be compromised. Sensitive data is stored securely to protect users' privacy.

Data Encryption: Sensitive data, such as personal patient information and billing records, is encrypted to prevent unauthorized access.

Role-Based Security: Different roles within the application (admin, patient, doctor) have varying levels of access to the system. This ensures that each user can only view or modify the data they are authorized to access.

# Future Enhancements

Potential Features: Some potential features that could be added in the future include telemedicine integration, advanced reporting capabilities, and mobile app support to allow users to access the platform on the go.

Scalability: Plans to scale the system to accommodate larger databases, more users, and additional functionality could be considered.

# Conclusion

The Patient Record Keeping Web App has successfully achieved its goal of streamlining the management of patient records, appointments, prescriptions, and billing. The system is secure, user-friendly, and designed to meet the needs of healthcare providers.

By providing a centralized platform for managing medical data, the application helps reduce errors, improve workflow efficiency, and deliver a better overall experience for both healthcare providers and patients.