ONLINE APPOINTMENT FOR HEALTHCARE CLINIC

Technical Design Document (TDD)

|  |  |  |  |
| --- | --- | --- | --- |
| Author | Version | Date | Description |
| Zohaib Waqar | 1.0 |  |  |

Table of Contents

**System Architecture Overview 2**

**Component Breakdown 2**

**Data Design 2**

**Technology Stack4**

**Assumptions & Constraints 4**

1. **System Architecture Overview**

The system is structured as a traditional MVC (Model-View-Controller) web application. It is organized as follows:

* 1. **Frontend:** Server-rendered web pages using HTML/CSS/JavaScript using Razor syntax.
  2. **Backend:** A monolithic backend using a web framework that adheres to MVC architecture (e.g. ASP.NET Core MVC).
  3. **Database:** Microsoft SQL Server used for data persistence.
  4. **APIs:** Currently, no RESTful APIs are exposed externally.

1. **Component Breakdown**

|  |  |
| --- | --- |
| Module | Responsibility |
| Register | Allows new users (patients, doctors, staff) to create accounts. |
| Login | Authenticates users and provides role-based access to the system. |
| Appointment Booking | Allows users (patients) to view available slots and book appointments. |
| Appointment History | Enables logged-in users to view past and upcoming appointments. |
| Profile | Displays and allows editing of user information (available only after login). |
| Schedule Availability | Enables doctors to set their available times for appointments. |
| Create Schedule | Enables staff to define time slots and manage availability calendars for doctors. |
| Reports | Allows admin/staff to generate reports (e.g., total appointments, no-shows, etc.). |

1. **Data Design**
   1. **User**

|  |  |  |
| --- | --- | --- |
| Field | Type | Notes |
| UserID | INT (PK) | Auto-increment |
| GovtID | VARCHAR | Unique identifier for login |
| PhoneNumber | VARCHAR |  |
| PasswordHash | VARCHAR | Store securely |
| Role | VARCHAR | 'Patient', 'Doctor', 'Staff' |

* 1. **Profile**

|  |  |  |
| --- | --- | --- |
| Field | Type | Notes |
| ProfileID | INT (PK) |  |
| UserID | INT (FK) | Linked to User |
| FirstName | VARCHAR |  |
| LastName | VARCHAR |  |

* 1. **Appointment**

|  |  |  |
| --- | --- | --- |
| Field | Type | Notes |
| AppointmentID | INT (PK) |  |
| PatientID | INT (FK) | Refers to UserID (Role: Patient) |
| DoctorID | INT (FK) | Refers to UserID (Role: Doctor) |
| Contact | VARCHAR | Patient contact |
| DateTime | DATETIME | Appointment date/time |
| Status | VARCHAR | 'Booked', 'Completed', 'No-show' |

* 1. **ScheduleAvailability (Doctor)**

|  |  |  |
| --- | --- | --- |
| Field | Type | Notes |
| AvailabilityID | INT (PK) |  |
| DoctorID | INT (FK) | Refers to UserID |
| Date | DATE |  |
| TimeSlot | VARCHAR | e.g., '9:00 AM - 11:00 AM' |

* 1. **Schedule (Created by Staff)**

|  |  |  |
| --- | --- | --- |
| Field | Type | Notes |
| ScheduleID | INT (PK) |  |
| DoctorID | INT (FK) |  |
| Date | DATE |  |
| RoomNumber | VARCHAR |  |

* 1. **ClinicStaffDetails**

|  |  |  |
| --- | --- | --- |
| Field | Type | Notes |
| StaffID | INT (FK) | From User (Role: Staff) |
| Experience | INT | In years |
| Qualification | VARCHAR |  |
| EmploymentType | VARCHAR | 'Full-time', 'Part-time' |

* 1. **Entity Relationships**

|  |  |  |  |
| --- | --- | --- | --- |
| From Entity | To Entity | Relationship Type | Cardinality |
| User | Profile | One-to-One | 1 : 1 |
| User (Patient) | Appointment | One-to-Many | 1 : ∞ |
| User (Doctor) | Appointment | One-to-Many | 1 : ∞ |
| User (Doctor) | ScheduleAvailability | One-to-Many | 1 : ∞ |
| User (Doctor) | Schedule | One-to-Many | 1 : ∞ |
| User (Staff) | ClinicStaffDetails | One-to-One | 1 : 1 |

* + 1. One **User** has one **Profile** containing personal data
    2. One **User (Patient)** can book multiple **Appointments** with doctors
    3. One **User (Doctor)** can have many appointments booked by **User (Patient)**
    4. One **User (Doctor)** can set multiple **ScheduleAvailability** time slots
    5. A **User (Doctor)** can be assigned to multiple **Schedule** slots by staff
    6. A **User (Staff)** has one record detailing to **ClinicStaffDetails**

For a visual representation of the system’s data structure and entity relationships, please refer to **Appendix A: Entity Relationship Diagram (ERD)**.

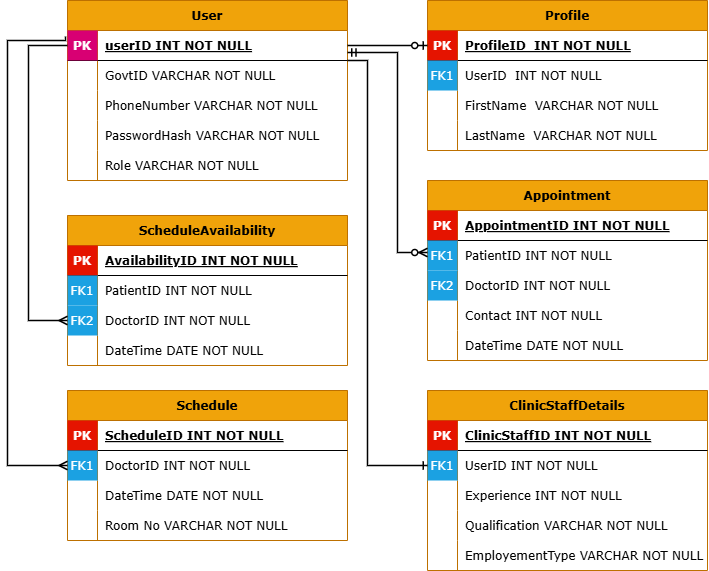
1. **Technology Stack**

|  |  |
| --- | --- |
| ****Layer**** | ****Technology**** |
| Frontend | HTML, CSS, JavaScript (or Razor if ASP.NET) |
| Backend | ASP.NET Core MVC (or similar MVC Framework) |
| Database | Microsoft SQL Server |
| ORM (if any) | Entity Framework or ADO.NET |
| Reporting Tools | In-app export features |
| Authentication | Form-based login with role-based access |

1. **Assumptions & Constraints**
   1. Only registered users can access Appointment History, Doctors Scheduling & Report, Finance Reports
   2. System roles are predefined: Patient, Doctor, Staff and Admin
   3. Doctors can only set their own availability
   4. Double-booking will be prevented via backend validation
   5. No external API integration is currently planned
   6. Reports are assumed to be downloadable in PDF/Excel formats.

**Appendix ‘A’**

**ENTITY RELATIONSHIP DIAGRAM**

****

***Figure A1*** - *This diagram illustrates the core entities of the Online Appointment Booking System and their relationships.*