Part 1 - Analysis, Architecture Design and Detailed Design

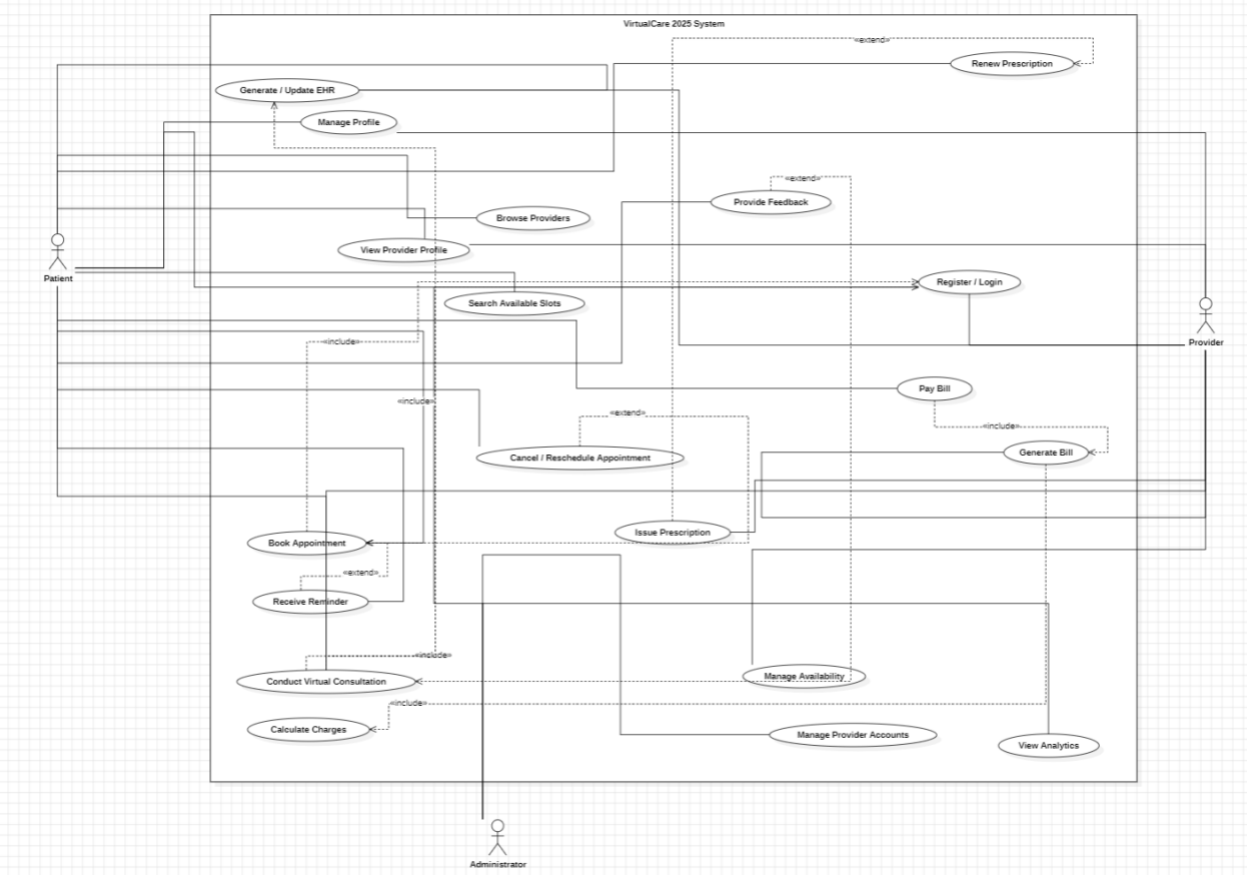

Figure 1: Use Case Diagram for Virtual Healthcare System (VHS)

Use Case Relationships:

- Actors ↔ Use Cases
- Patient → Register / Login
- Patient → Manage Profile
- Patient → Browse Providers
- Patient → View Provider Profile
- Patient → Search Available Slots
- Patient → Book Appointment
- Patient → Cancel / Reschedule Appointment
- Patient → Receive Reminder
- Patient → Conduct Virtual Consultation

- Patient → Generate / Update EHR
- Patient → Renew Prescription
- Patient → Pay Bill
- Patient → Provide Feedback
- Provider → Register / Login
- Provider → Manage Profile
- Provider → View Provider Profile
- Provider → Manage Availability
- Provider → Conduct Virtual Consultation
- Provider → Generate / Update EHR
- Provider → Issue Prescription
- Provider → Generate Bill
- Admin → Register / Login
- Admin → Manage Provider Accounts
- Admin → View Analytics

Include Relationships:

- Book Appointment → Register / Login
- Conduct Virtual Consultation → Register / Login
- Conduct Virtual Consultation → Generate / Update EHR
- Generate Bill → Calculate Charges
- Pay Bill → Generate Bill

Extend Relationships:

- Cancel / Reschedule Appointment → Book Appointment
- Receive Reminder → Book Appointment
- Provide Feedback → Conduct Virtual Consultation
- Renew Prescription → Issue Prescription

Use Case Diagram — Figure Explanation:

Figure 1: Virtual Care 2025 Use Case Diagram
This diagram illustrates the main actors (Patient, Provider, and Admin) and their interactions with the system.
It also shows the relationships among use cases, including **<<include>>** and **<<extend>>** dependencies between core and supporting functions.

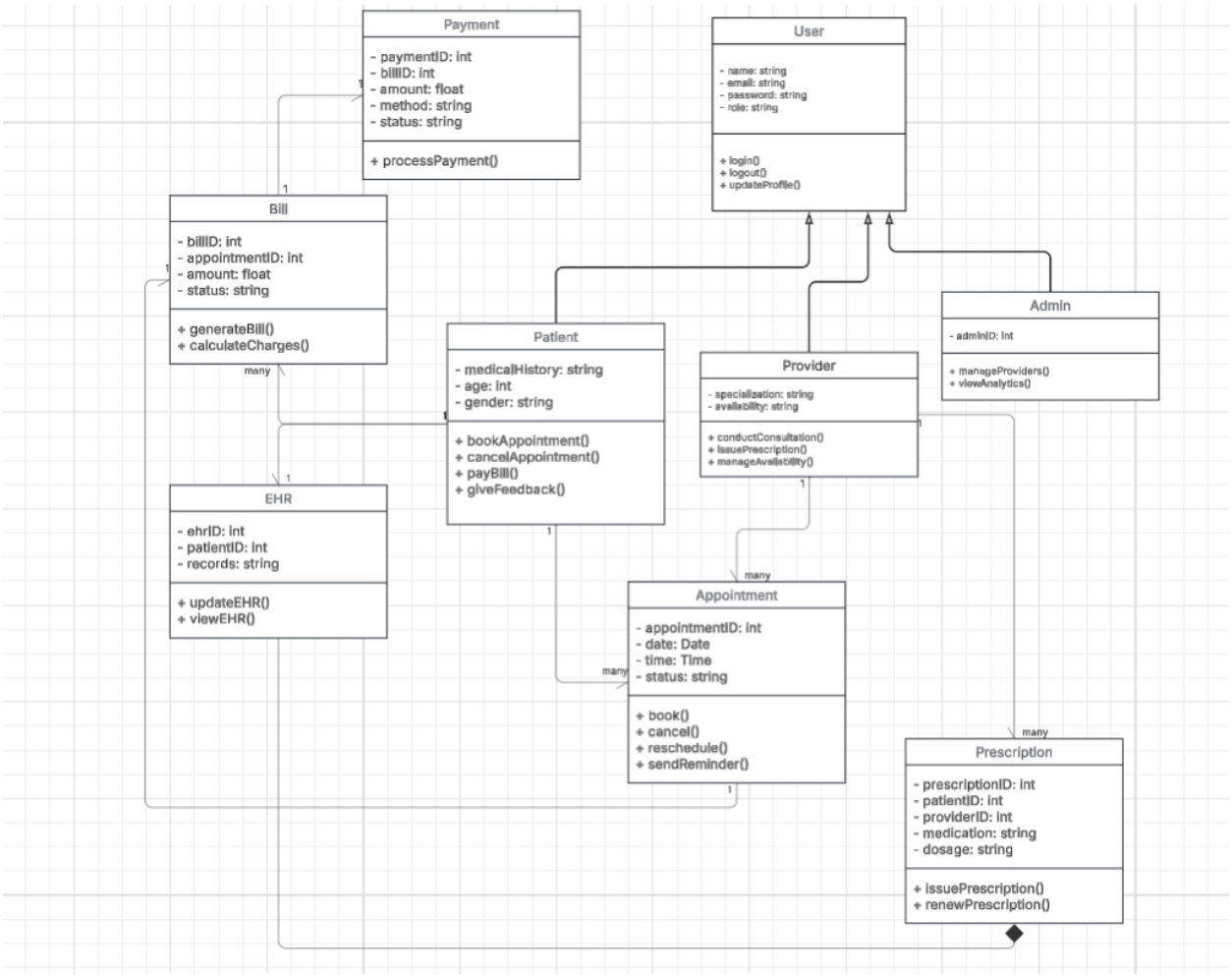

Figure 2: Class Diagram for Virtual Healthcare System (VHS)

Class Diagram Relationships :

1. **User** ⟶ inherits → Patient
2. **User** ⟶ inherits → Provider
3. **User** ⟶ inherits → Admin
4. **Patient** ⟶ 1 to many (0..*) → Appointment
5. **Provider** ⟶ 1 to many (0..*) → Appointment
6. **Patient** ⟶ 1 to 1 → EHR
7. **Provider** ⟶ 1 to many (0..*) → Prescription
8. **Patient** ⟶ 1 to many (0..*) → Bill
9. **Bill** ⟶ 1 to 1 → Payment
10. **Appointment** ⟶ 1 to 1 → Bill

11. **EHR *-- Prescription (Composition)**

Class Diagram — Figure Explanation:

Figure 2: VirtualCare 2025 Class Diagram
This diagram defines the main system classes and their relationships.
It includes inheritance from the User superclass (specialized into Patient, Provider, and Admin),
as well as associations between domain classes (Appointment, EHR, Prescription, Bill, and Payment).
Each class belongs to a subsystem in the architecture model:

- Auth Service: User, Admin
- Appointment Service: Appointment, Patient, Provider
- EHR Service: EHR, Prescription
- Billing Service: Bill, Payment

Class-to-Subsystem Mapping:

- **Auth Service:** `User`, `Admin`
- **Appointment Service:** `Appointment`, `Patient`, `Provider`
- **EHR Service:** `EHR`, `Prescription`
- **Billing Service:** `Bill`, `Payment`

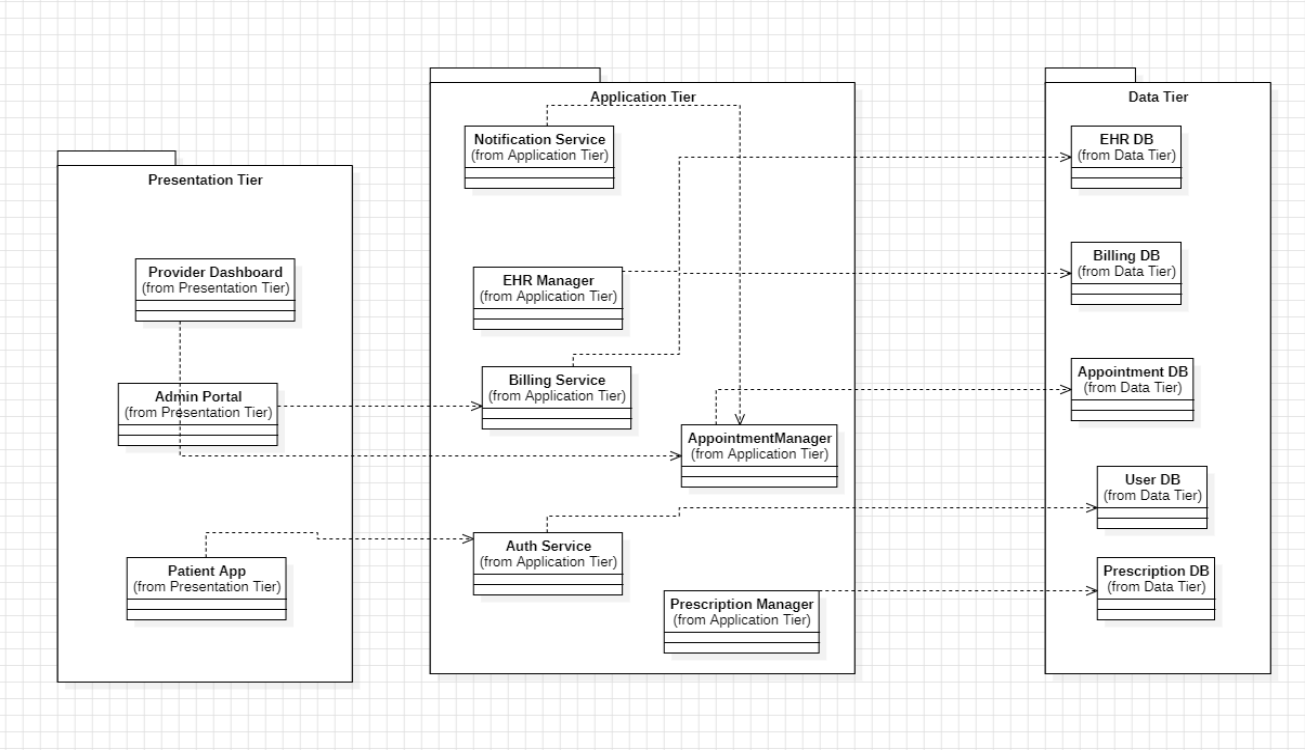

Figure 3: 3 Tier Architecture Model for Virtual Healthcare System (VHS)

Dependencies:

1. Patient App → Auth Service
2. Provider Dashboard → Appointment Manager
3. Admin Portal → Billing Service
4. Auth Service → User DB
5. Appointment Manager → Appointment DB
6. EHR Manager → EHRDB
7. Prescription Manager → Prescription DB
8. Billing Service → Billing DB
9. Notification Service → Appointment Manager

Architecture Diagram — Figure Explanation:

Figure 3: VirtualCare 2025 Architecture Model:
The system is structured using a 3-tier architecture:

1. Presentation Layer (User interfaces for Patients, Providers, Admins)
2. Application/Logic Layer (Core services such as Auth Service, Appointment Service, Consultation Service, EHR Service, Billing Service)

3. Data Layer (Database handling users, appointments, records, and billing data).
   Each subsystem depends on lower layers, maintaining modularity and separation of concerns.

Security Controls:

- **Authentication:** each user logs in securely with unique credentials
- **Authorization:** role-based access (Patient, Provider, Admin)
- **Encryption:** protects sensitive patient data during transmission
- **Data Privacy:** only authorized users can access health records