Scope and Deliverables #DoctorApp

Front End will be up cycled from an existing app.

MVP SCOPE (Back-End & System-Level Functionality)

These are the essential backend deliverables to support a working platform that connects doctors and patients internationally.

1. Authentication & User Management
- Sign up/login for doctors and patients (email/phone/password)
- Role-based access control (doctor, patient, admin)
- Password recovery and basic profile management
- Account verification

2. Patient Module
- Upload and manage health records (images, PDFs, notes)
- View appointment history & upcoming appointments
- Join video calls (Via API preferably)
- Access prescriptions or treatment notes from doctors
- Notifications/reminders (email (optional) and push)

3. Doctor Module
- View patient profiles and consultation history
- Record notes, diagnosis, and prescriptions post-consultation
- Manage availability (set available time slots)
- Join video calls
- View dashboard (appointments, earnings, etc.)

4. Appointment Management
- Patient books appointments with doctor (from available slots)
- Appointment status flow (requested → accepted → completed/cancelled)
- Timezone handling and conversion
- Calendar integration (optional)

5. Video Calling Integration
- Integration with 3rd party video service (Gmeet preferred)

6. Prescription & Notes
- Doctor can write and save prescriptions or notes
- Patient can view/download prescriptions
- PDF generation of prescription (optional)

7. Admin Dashboard (Basic)

- View & manage users (doctors/patients)
- View appointments
- Ban/suspend users if needed
- View usage stats (basic metrics)

8. Notifications (System-level)
- Appointment confirmations & reminders (email + push)
- Missed appointment alerts
- Admin notices if needed

9. Security & Compliance
- Secure API
- Data encryption (at rest & in transit)
- GDPR/HIPAA-aligned data practices (Phase 2)
- Rate limiting, audit logging

————

Developer Deliverables

1. Backend Codebase
- RESTful (or GraphQL) APIs with clear documentation (Swagger/Postman collection)
- Organized code with modular architecture

2. Database
- Schema definition & setup (PostgreSQL, MongoDB)
- Seed/test data
- Admin credentials for DB access
- Secure backup policy

3. API Documentation
- For frontend team to integrate with
- Clear instructions on auth, data structures, error handling

4. DevOps & Deployment
- Staging + Production environment setup
- CI/CD pipelines
- SSL, DNS setup
- Monitoring setup (e.g., logs, uptime)

5. Video Integration
- Setup & working integration of Google Meets

6. Testing & QA
- Unit tests for key modules
- End-to-end testing for critical user journeys

7. Admin Access

- Basic web dashboard (or even a no-code tool) for admin functionality
- Credentials and training for you to manage users

8. Payment Gateway Integration