ProfessionalsBD

Requirement Analysis & System Design Report

Project Title: ProfessionalsBD – Online Professional Consultation Platform
Version: 1.0
Date: January 05, 2026

Prepared by:
Moniruzzaman Mahdi
Full-Stack Developer (MERN | Next.js | TypeScript | Prisma)

Contact:
Email: mahdimoniruzzaman@gmail.com
Portfolio: https://moniruzzaman-mahdi.vercel.app
GitHub: https://github.com/mahdimonir

Organization: KINGS ASSOCIATION
Status: Planed

Project Overview

ProfessionalsBD is a platform connecting users with professionals (specialists) for booked video consultations. It supports 4 user roles: **User**, **Professional**, **Admin**, **Moderator**.

Core Features

- Appointment booking
- Video meetings (via Stream.io with SFU topology for multi-participant support)
- Slot management (professionals can add specific slots or allow anytime booking)
- Email notifications (SMTP)
- Optional meeting recording and transcript storage (with admin approval, stored in Cloudinary/DB)

Suggested Additional Necessary Features

Authentication and Security

- Role-based access control (RBAC)
- Multi-factor authentication (MFA)
- Password reset via email
- JWT-based authentication
- Data encryption and CAPTCHA/rate limiting

Profile and Discovery

- Detailed profiles (bio, specialties, certifications via Cloudinary)
- Search and filters (by specialty, ratings, availability)
- Professional verification by admins
- Recommendations

Booking and Scheduling

- Calendar integration (Google/iCal)
- Timezone handling
- Waitlist and reminders (email)
- Cancellation/rescheduling policies

Video Meetings (Stream.io)

- Multi-participant support, screen sharing, chat
- Post-meeting feedback/ratings
- Recording/transcript generation (with admin approval)

Payments and Monetization

- Payment gateway (e.g., bKash, Rocket, SSL Commerz)
- Payouts with platform commission
- PDF invoices (Cloudinary storage)

Notifications and Communication

- Email notifications (SMTP)
- In-app messaging (Optional)

Analytics and Reporting

- Role-specific dashboards
- Exportable reports

Moderation and Compliance

- Dispute resolution
- Privacy features (GDPR-like)

Admin/Moderator Tools

- User management, content approval

General

- Mobile-responsive (PWA)
- SEO, accessibility

Overall System Architecture

- **Frontend**: Next.js (SSR/SSG, React)
- **Backend**: Node.js/Express, Prisma ORM, NeonDB (Postgres)
- **Integrations**: Stream.io (video), Cloudinary (media), Nodemailer (SMTP)
- **Non-Functional**: High performance, security (HTTPS, OWASP), scalability

Backend Requirements

Data Models (Prisma Schema Examples)

- User, ProfessionalProfile, Booking, Meeting, Notification, Review, Payment, etc.

APIs (RESTful)

- Auth: register/login/reset
- Users: profiles, search
- Professionals: slots/availability
- Bookings: create/check/join/cancel
- Meetings: video integration, recording approval
- Admin: user management, approvals
- Payments: integration

Business Logic

- Availability checks
- Secure video links
- Scheduled notifications

Frontend Requirements

Pages/Routes

- Public: Home, Register/Login
- User: Dashboard, Search, Booking, Meeting
- Professional: Dashboard, Profile/Slots
- Admin/Moderator: Management dashboards

Components

- SearchBar, Calendar, VideoPlayer (Stream.io SDK)
- Forms, Modals, Dashboards (charts)

User Flows

- Booking process
- Meeting join
- Admin approvals

References

1. Stream.io Documentation – Video Calling & SFU Topology
   https://getstream.io/video/docs/react/guides/video-calling/
   (Used for multi-participant video meetings with SFU support)

2. Cloudinary Documentation – Media Upload & Storage
   https://cloudinary.com/documentation
   (Reference for storing profile images, certifications, and meeting recordings)

3. Prisma ORM Documentation – Data Modeling with PostgreSQL
   https://www.prisma.io/docs
   (Used for defining database schema and relationships)

4. Next.js Official Documentation
   https://nextjs.org/docs
   (Framework for frontend development and SSR/SEO)

5. Best Practices for Appointment Booking Systems
   Calendly Help Center: https://help.calendly.com
   (Inspiration for slot management, availability, and booking flows)

6. Telemedicine Platform Features Reference
   Doxy.me Feature Overview: https://doxy.me/features
   (Ideas for secure video consultations, recording with consent, and admin controls)

7. Email Notification Standards
   Nodemailer Documentation: https://nodemailer.com
   (Used for SMTP-based email notifications and reminders)