AI-Based Consultation Chatbot Web Application

Project Status Document

Project Overview

This document provides a comprehensive overview of the AI-Based Consultation Chatbot Web Application, detailing the current implementation status, working features, and remaining tasks based on the Software Requirements Specification (SRS).

Technology Stack

- **Frontend**: React.js with Redux Toolkit for state management
- **Backend**: Firebase (Authentication, Firestore Database, Cloud Functions)
- **Authentication**: Firebase Authentication
- **Database**: Cloud Firestore
- **Styling**: CSS Modules/Tailwind CSS

Current Implementation Status

Authentication & User Management

- ✅ User registration and login functionality
- ✅ User profile management
- ✅ Admin and user role separation
- ✅ Authentication state persistence
- ❌ Multilingual login/registration support

Dashboard Interfaces

- ✅ User Dashboard with appointment history
- ✅ Admin Dashboard with statistics (users, appointments, revenue)
- ✅ Expert specialization display in both dashboards
- ❌ Advanced analytics and revenue reporting
- ❌ Session logs and detailed user activity

Appointment System

- ✅ Expert listing and selection
- ✅ Appointment scheduling with date/time selection
- ✅ Appointment status management (scheduled, completed, cancelled)
- ✅ Meeting link generation for virtual consultations
- ❌ Calendar integration (Google Calendar)
- ❌ Notification system for upcoming appointments

Payment Integration

- ✅ Basic payment data structure
- ❌ Razorpay integration for processing payments
- ❌ Subscription vs. pay-per-call options
- ❌ Invoice generation and history

AI Chatbot Functionality

- ❌ Text-based consultations via AI
- ❌ Integration with Large Language Model (LLM)
- ❌ Chat history storage
- ❌ AI response quality control

Voice Capabilities

- ❌ Speech-to-Text integration (Deepgram Nova-3)
- ❌ Text-to-Speech integration (ElevenLabs Turbo v2.5)
- ❌ Voice input processing
- ❌ Voice output for AI responses

Multilingual Support

- ❌ Multiple language support for UI
- ❌ Language detection and switching
- ❌ Multilingual voice processing

Expert Management

- ✅ Basic expert profiles with specialization
- ✅ Expert availability management
- ❌ Expert rating and review system
- ❌ Expert dashboard for managing appointments

UI/UX

- ✅ Responsive layout
- ✅ Dark/light mode support
- ✅ User-friendly navigation
- ❌ Accessibility features (WCAG 2.1 compliance)
- ❌ Mobile optimization

Data Structure

The application currently uses the following core data structures:

User

- User authentication details
- Profile information
- Payment history

- Appointment history

Expert

```
export interface Expert {
  id: string;
  name: string;
  specialization: string;
  experience: number;
  rating: number;
  photoURL: string;
  availability: {
    day: string;
    slots: string[];
  }[];
}
```

Appointment

```
export interface Appointment {
  id: string;
  userId: string;
  expertId: string;
  expertName?: string;
  expertSpecialization?: string;
  date: string;
  time: string;
  status: 'scheduled' | 'completed' | 'cancelled';
  meetingLink?: string;
  notes?: string;
  createdAt: number;
}
```

Critical Components Remaining

AI Integration

The core AI functionality is not yet implemented, including:

- Integration with an LLM for text consultations
- Speech-to-Text (STT) processing using Deepgram Nova-3
- Text-to-Speech (TTS) using ElevenLabs Turbo v2.5
- Real-time voice processing pipeline

Payment System

The payment integration is incomplete:

- Razorpay integration for handling transactions
- Subscription model implementation
- Pay-per-call pricing options
- Secure payment processing
- Invoice generation

Multilingual Support

The multilingual capabilities are not implemented:

- UI language switching
- Content translation
- Voice processing in multiple languages

Calendar Integration

The scheduling system needs to be connected to calendar services:

- Google Calendar API integration
- Calendar event creation and management
- Notification system for appointments

Next Steps

1. **Priority 1: Core AI Functionality**
    - Implement the LLM integration for text-based consultations
    - Set up the voice processing pipeline with STT and TTS
    - Create the chat interface for AI interactions
2. **Priority 2: Payment System**
    - Integrate Razorpay for payment processing
    - Implement subscription and pay-per-call options
    - Create secure payment flows and invoice generation
3. **Priority 3: Expert Consultation Enhancement**
    - Complete the expert management system
    - Implement the expert dashboard
    - Add rating and review functionality
4. **Priority 4: Multilingual Support**
    - Implement language switching in the UI
    - Set up multilingual voice processing
    - Test with various languages to ensure compatibility

Technical Debt and Considerations

1. **Performance Optimization**

    - The current implementation may need optimization for handling real-time voice processing
    - Database queries should be optimized for scalability

2. **Security Enhancements**
    - Payment security measures need implementation
    - Data encryption for sensitive user information
    - HIPAA compliance for medical consultations

3. **API Integration Requirements**
    - API keys needed for Razorpay, LLM, STT, and TTS services
    - Integration testing with each third-party service

Weekly Planning

Week 1 (Feb 19 – Feb 25, 2025)

- ✅ Kickoff meeting & project planning
- ✅ Define project scope
- ✅ Initial market research & competitor analysis

Week 2 (Feb 26 – Mar 3, 2025)

- ✅ Gather requirements for chatbot features & consultation workflow
- ✅ Identify necessary APIs for AI, payments, voice, and scheduling
- ✅ Define data security & compliance requirements

Week 3 (Mar 4 – Mar 10, 2025)

- ✅ Finalize technical specifications
- ✅ Design chatbot architecture for multi-domain consultation
- ✅ Plan database structure & backend workflow

Week 4 (Mar 11 – Mar 17, 2025)

- ⏳ Develop chatbot prototype (text-based consultation)
- ❌ Integrate first version of LLM API (OpenAI GPT/Gemini/LLaMA)
- ✅ Begin UI/UX wireframing

Week 5 (Mar 18 – Mar 24, 2025)

- ❌ Train chatbot to handle multiple domains (health, legal, finance, etc.)
- ❌ Develop AI response accuracy testing framework
- ✅ Continue UI/UX design work

Week 6 (Mar 25 – Mar 31, 2025)

- ⏳ Integrate secure payment system (Stripe/PayPal/Razorpay)
- ❌ Implement backend for payment processing and session tracking

- ❌ Gather user feedback on chatbot

Week 7 (Apr 1 – Apr 7, 2025)

- ❌ Integrate AI voice capabilities:
  - Speech recognition (Whisper/Deepgram)
  - Text-to-speech (Google TTS/Amazon Polly)
- ❌ Enhance chatbot response accuracy

Week 8 (Apr 8 – Apr 14, 2025)

- ❌ Test and optimize voice-based consultations
- ❌ Conduct usability testing for both text and voice interactions
- ❌ Implement chatbot conversation logging

Week 9 (Apr 15 – Apr 21, 2025)

- ❌ Develop audio call feature with AI model

Week 10 (Apr 22 – Apr 28, 2025)

- ❌ Continued development of audio call feature

Week 11 (Apr 29 – May 5, 2025)

- ✅ Develop admin panel for managing users, sessions, and payments
- ✅ Implement consultant dashboard for tracking their appointments and earnings

Week 12 (May 6 – May 12, 2025)

- ❌ Complete admin panel with analytics (session tracking, revenue insights)
- ❌ Full-system integration testing (chatbot, payments, scheduling)

Week 13 (May 13 – May 19, 2025)

- ❌ Final system testing & security audit
- ❌ Prepare documentation and user guides
- ❌ Launch beta testing with selected consultants & users
- ❌ Collect feedback and plan improvements

Conclusion

The AI-Based Consultation Chatbot Web Application has a functional foundation with user management, appointment scheduling, and dashboard interfaces implemented. However, the core AI functionality, payment processing, and multilingual support still need to be implemented to fulfill the SRS requirements completely.

The project has a clear roadmap for completion, with well-defined next steps and priorities. By addressing the remaining components in the suggested order, the application can be brought to full compliance with the SRS document.