MEDIGENE: TELEHEALTH APPLICATION

INTRODUCTION

The post-pandemic world has redefined the way we live, work, and access essential services—especially healthcare. With people increasingly confined to remote locations, the demand for quick, affordable, and reliable healthcare has never been greater. The need of the hour is a seamless, digitally enabled solution that bridges the gap between patients and healthcare providers, ensuring quality care from anywhere in the world.

Today, we present a **Telehealth application** designed to address these critical challenges by offering:

- **Instant access** to health consultations with an experience mirroring in-person visits.
- **Efficient patient onboarding**, capturing essential details and specialty needs upfront.
- **Secure payment processing** before consultations to ensure a smooth experience.
- **Strict privacy compliance** for Protected Health Information (PHI), with secure data storage for future reference.
- **Real-time chat functionality** to enhance communication during consultations.
- **AI-powered transcription services** to overcome language and dialect barriers in remote consultations.

This solution not only meets the urgent demands of today's healthcare landscape but also paves the way for a more connected, accessible, and patient-centric future.

TECH STACK USED

FRONTEND:

NEXTJS: FRAMEWORK

TAILWINDCSS: STYLING

TYPESCRIPT: CODE LANGUAGE

MUI (MATERIAL UI): EFFECTS

BACKEND:

POSTGRESQL: DATABASE

RAZORPAY: PAYMENT INTEGRATION

WEB RTC VIDEO: VIDEO API INTEGRATION

MULTER: IMGAE UPLOAD

JWT: FOR SECURITY TOKEN

GOOGLEAPIS: LOCATION AND DISTANCE API

PROJECT FEATURES:

- Secure Video Consultations
- Doctor Specialty Filters & Booking
- Medical Record Management
- Payment Integration (Pre-consultation)
- PHI Compliance & Data Encryption

🔷 3. Functional Requirements

ID	Requirement Description
RQ-01	Users (patients/doctors) can register and log in securely
RQ-02	Patients select medical specialty and submit health info
RQ-03	Patients browse doctors and schedule appointments
RQ-04	Payments must be made before a consultation starts
RQ-05	Real-time video consultation using WebRTC/Agora
RQ-06	In-session chat system for doctor-patient communication
RQ-07	Secure upload and storage of medical records (PHI)
RQ-08	Admin dashboard to manage users and records
RQ-09	Transcription support during video calls (prototype/demo stage)
RQ-10	Frontend and backend must support mobile and desktop use

🔷 4. Manual Testing – Sample Test Cases

Test Case ID	Description	Steps	Expected Result	Status
TC-01	Patient Registration	Enter details, click Register	User created	✅ Pass
TC-02	Specialty Selection	Select "Cardiology", submit	Next page loads	✅ Pass
TC-03	Payment before booking	Choose time, enter card, pay	Payment success, consult scheduled	✅ Pass
TC-04	Video Call Launch	Click "Join"	Call starts	✅ Pass
TC-05	In-session chat	Send message during call	Message delivered	✅ Pass
TC-06	Medical Record Upload	Upload prescription	File saved securely	✅ Pass
TC-07	Transcription Accuracy	Speak during consult	Words transcribed correctly	⏳ In progress

---

◈ 5. Automated Testing Plan

۞ Unit Testing (Example Using Jest / Mocha)

Component	Test Description
Auth Module	Validate token issuance & expiration
Payment Handler	Simulate valid/invalid payment flows
Appointment Scheduler	Test time conflict resolution
Chat System	Check message storage and delivery
File Upload	Ensure only valid formats (PDF, JPEG) are stored securely

Sample Jest Unit Test

```
test('should not allow double booking for the same slot', () => {
const result = bookSlot(userId, doctorId, '2025-05-26 10:00');
  expect(result.success).toBe(false);
});
```

⚙ API Testing with Postman (Suggested)

- **POST /login** → Verify correct credentials returns token.
- **POST /appointment** → Check booking with/without payment.
- **GET /records/:id** → PHI must return only if authorized.
- **POST /chat** → Sends and receives messages securely.

⚙ UI Automation (Optional Tools)

Tool	Use
Selenium / Playwright	Test registration, booking flows
Katalon Recorder	Record-and-play login to consult flow

◈ 7. Security & PHI Consideration

- All PHI data is encrypted at rest and in transit.
- HTTPS enforced across all communications.
- Role-based access control (Doctor/Patient/Admin).
- Audit logs maintained for sensitive activity.
- Optional MFA for admin login.

◈ 8. Future Enhancements

- Doctor rating & feedback system.
- Real-time vitals integration via wearable sync.
- WhatsApp or SMS reminders for appointments.
