**Project Overview**

I am developing a website named **"Nepals Care"** to automate clinic management processes. The primary goal is to streamline operations such as appointment booking, online consultations, medicine management, and user management for three key user roles: doctors, patients, and admins.

**Technology Stack**

I am using the following technologies, with which I am comfortable:

* **Backend:** Django, Django REST Framework (DRF), and additional Django libraries as needed.
* **Frontend:** HTML, Tailwind CSS, JavaScript.
* **Programming:** Proficient in Python and adaptable to new libraries or frameworks when required.

**Current Progress**

* **Frontend:** I have completed approximately **60-70%** of the frontend using HTML, Tailwind CSS, and JavaScript. The user interface (UI) is functional, utilizing dummy data stored in JavaScript arrays and rendered dynamically.
* **Backend:** I have initialized a Django project and app, set up basic views and URLs for page navigation, and I am confident in my Django skills to expand this further.

**Core Features (Must-Haves)**

The website must include the following essential functionalities:

1. **Appointment Booking:** Enable scheduling, managing time slots, issuing tickets, and tracking bookings.
2. **Video Consulting:** Facilitate online consultation sessions between doctors and patients.
3. **Medicine Management:** Manage prescriptions, track medication usage, and handle refills.
4. **User Management:** Provide role-specific features for doctors, patients, and admins.

**Challenges Faced**

Initially, I developed the frontend without a clear plan, leading to inefficiencies and a harder development process. I aim to avoid this mistake with the backend by creating a structured and efficient plan from the start.

**Objectives**

I need assistance with:

* Brainstorming and logic building for the backend.
* Creating a detailed backend development plan.
* Analyzing my current frontend to identify areas for updates, additions, or removals to ensure seamless integration with the backend.

**Frontend Description**

Below is a detailed breakdown of the frontend I’ve built so far, including its structure and the dummy JSON data used to render it dynamically.

**A) Home Page**

* **Purpose:** Introduces the system and its workflow.
* **Components:**
  1. **Static Content:** Describes how the system works:
     + Create an account and configure prescription settings.
     + Accept bookings and manage your schedule.
     + Automate payments and oversee your practice.
  2. **Doctors List:** Displays each doctor’s image, name, and specialty.
  3. **Patient Feedback:** Shows profile pictures, names, ratings, and review messages.
* **Sample JSON:**

json

CollapseWrapCopy

const homeData = {

doctors: [

{

id: "dr-001",

name: "Dr. Emily Smith",

specialty: "General Medicine",

image: "{% static 'doc-1.png' %}"

},

{

id: "dr-002",

name: "Dr. Sarah Johnson",

specialty: "Cardiology",

image: "{% static 'doc-2.png' %}"

}

],

feedback: [

{

patientName: "John Doe",

profilePic: "{% static 'patient-1.png' %}",

rating: 4.8,

review: "Great experience, very professional!"

},

{

patientName: "Jane Smith",

profilePic: "{% static 'patient-2.png' %}",

rating: 4.5,

review: "Helped me recover quickly, highly recommend."

}

]

};

**B) Login/Signup Page**

* **Purpose:** Handles user authentication and registration.
* **Components:**
  + **Registration Form:** Fields include Full Name, Email, Password, Confirm Password, and an agreement checkbox.
  + **Login Form:** Fields include Email, Password, and a "Remember Me" checkbox.
* **Sample JSON:**

json

CollapseWrapCopy

const authData = {

registration: {

fullName: "John Doe",

email: "john.doe@example.com",

password: "securePass123",

confirmPassword: "securePass123",

agreeTerms: true

},

login: {

email: "john.doe@example.com",

password: "securePass123",

rememberMe: true

}

};

**C) Patient Section**

* **Purpose:** Provides patients with tools to manage their healthcare.
* **Shared Component:**
  + **Navbar (Repeated Across Pages):**
    - Notifications (e.g., "Appointment Approved," "Appointment Canceled").
    - User Banner: Profile picture, name, email, buttons for Edit Profile, Change Password, and Logout.
    - User ID Card: Profile picture, name, patient ID, date of birth, blood group, phone number, email, address, emergency contact, medical info (allergies, current medications), last updated date, card ID.
  + **Sample JSON for Navbar:**

json

CollapseWrapCopy

const navbarData = {

notifications: [

{ id: 1, message: "Appointment Approved", time: "2025-03-23 10:00 AM" },

{ id: 2, message: "Appointment Canceled", time: "2025-03-22 02:00 PM" }

],

userBanner: {

profilePic: "{% static 'patient-1.png' %}",

name: "John Doe",

email: "john.doe@example.com"

},

userIdCard: {

profilePic: "{% static 'patient-1.png' %}",

name: "John Doe",

patientId: "PAT-001",

dob: "1990-05-15",

bloodGroup: "O+",

phone: "+977-9876543210",

email: "john.doe@example.com",

address: "Kathmandu, Nepal",

emergencyContact: "+977-9123456789",

medicalInfo: {

allergies: "Peanuts, Penicillin",

currentMedications: "Amlodipine 10mg"

},

lastUpdated: "2025-03-20",

cardId: "CARD-001"

}

};

**C.a) Patient Dashboard**

* **Purpose:** Offers a quick overview of the patient’s health status and activities.
* **Components:**
  + **Summary Divs:**
    1. Upcoming Appointments: Date, time, type, "View All" button.
    2. Recent Lab Results: Last two results (e.g., Cholesterol, Blood Pressure), "View All" button.
    3. Prescription Alerts: Medication name, refill due date, percentage left, progress bar, "View All" button.
    4. Health Reminders: Upcoming tasks (e.g., Flu Shot), "View All" button.
  + **Scroll Section:** Totals for appointments, online meetings, and medicines.
  + **Overview & Health Timeline:**
    1. Appointments Summary: Three recent/upcoming appointments with date, time, doctor, type, "View All" button.
    2. Recent Activity: Three activities (e.g., appointment scheduled), "View All" button.
    3. Health Timeline: Vertical graph with entries (e.g., "Annual Check-up"), date, description, actions (e.g., "View Full Report").
* **Sample JSON:**

json

CollapseWrapCopy

const patientDashboard = {

summary: {

upcomingAppointments: [

{ date: "2025-03-25", time: "10:00 AM", type: "General Check-up" }

],

recentLabResults: [

{ name: "Cholesterol", result: "Normal", date: "2025-03-10" },

{ name: "Blood Pressure", result: "Elevated", date: "2025-03-10" }

],

prescriptionAlerts: [

{ name: "Lisinopril", refillDue: "2025-03-26", remaining: "25%", progress: 25 }

],

healthReminders: [

{ task: "Flu Shot", status: "Due Soon" },

{ task: "Dental Check-up", status: "Upcoming" }

]

},

scrollSection: {

totalAppointments: 5,

totalOnlineMeetings: 2,

totalMedicines: 3

},

overview: {

appointments: [

{ date: "2025-03-25", time: "10:00 AM", doctor: "Dr. Emily Smith", type: "General" }

],

recentActivity: [

{ action: "Appointment Scheduled", date: "2025-03-20" }

],

healthTimeline: [

{

event: "Annual Check-up",

date: "2025-03-10",

description: "Routine check-up completed",

actions: ["Blood Work", "View Full Report"]

}

]

}

};

**C.b) View Appointments**

* **Purpose:** Allows patients to review and manage their appointments.
* **Components:**
  + **Filter Header:** Search, consultation type, status (pending, completed, etc.), time range (today, last 30 days).
  + **List Table:** Columns for patient info (name, email), doctor info (name, email, specialty), date/time, status, actions (view, cancel).
  + **Entries Filter:** Default 10 entries per page, with pagination.
* **Sample JSON:**

json

CollapseWrapCopy

const appointmentsList = [

{

patient: { name: "John Doe", email: "john.doe@example.com" },

doctor: { name: "Dr. Emily Smith", email: "emily.smith@nepals.care", specialty: "General Medicine" },

date: "2025-03-25",

time: "10:00 AM",

status: "Confirmed",

actions: ["View", "Cancel"]

},

{

patient: { name: "John Doe", email: "john.doe@example.com" },

doctor: { name: "Dr. Sarah Johnson", email: "sarah.j@nepals.care", specialty: "Cardiology" },

date: "2025-03-20",

time: "02:00 PM",

status: "Completed",

actions: ["View"]

}

];

**C.c) Book an Appointment**

* **Purpose:** Guides patients through booking a new appointment.
* **Components:**
  + **Step-by-Step Process:**
    1. Select Doctor: List with name, specialty, rating, reviews, experience, availability.
    2. Select Date & Type: Choose date and type (General, Follow-up, Specialist, Online).
    3. Select Time Slot: Available slots (e.g., 9:00 AM, 10:00 AM).
    4. Describe Reason: Text input or file upload (documents, voice clips).
    5. Confirm Booking: Displays consultation fee (no payment integration yet).
* **Sample JSON:**

json

CollapseWrapCopy

const doctors = [

{

id: "dr-001",

name: "Dr. Emily Smith",

specialty: "General Medicine",

experience: 12,

rating: 4.9,

reviews: 124,

fees: 150,

image: "{% static 'doc-1.png' %}",

availability: {

consultation: {

"2025-03-25": ["09:00 AM", "10:00 AM", "11:00 AM"],

"2025-03-26": ["09:00 AM", "10:00 AM"]

},

followup: {

"2025-03-25": ["02:00 PM"]

},

online: {

"2025-03-25": ["09:00 AM", "10:00 AM", "02:00 PM"]

}

},

appointmentTypes: ["consultation", "followup", "online"]

}

];

**C.d) Document Management**

* **Purpose:** Manages patient-uploaded documents.
* **Components:**
  + **Filter Section:** Search, document type (MRI, blood report, etc.), sort (newest first, oldest first, A-Z).
  + **Document List:** Thumbnails, name, type, uploader’s name, date, actions (download, edit, delete).
  + **Add New Document:** Fields for file name, type, attachment, save button.
* **Sample JSON:**

json

CollapseWrapCopy

const documents = [

{

id: 1,

thumbnail: "{% static 'doc-thumb-1.png' %}",

name: "Blood Report March 2025",

type: "Blood Report",

uploader: "John Doe",

date: "2025-03-10",

actions: ["Download", "Edit", "Delete"]

},

{

id: 2,

thumbnail: "{% static 'doc-thumb-2.png' %}",

name: "MRI Scan",

type: "MRI",

uploader: "John Doe",

date: "2025-02-15",

actions: ["Download", "Edit", "Delete"]

}

];

**C.e) Prescription Management**

* **Purpose:** Tracks and manages patient prescriptions.
* **Components:**
  + **Summary Divs:** Active Prescriptions, Pending Refills, Recent Orders, Alerts.
  + **Prescription Lists:**
    - **Active Prescriptions:** Table with medication name, dosage, doctor, status, details dropdown (instructions, refill/report buttons).
    - **Prescription History:** Table with medication, dosage, doctor, date range, status, reason.
    - **Today’s Medications:** List with name, time, taken status.
* **Sample JSON:**

json

CollapseWrapCopy

const prescriptionData = {

activePrescriptions: [

{

id: 1,

name: "Amlodipine",

dosage: "10mg",

frequency: "Once daily",

doctor: "Dr. Sarah Johnson",

specialty: "Cardiology",

startDate: "2025-01-15",

endDate Roscoe: "2025-03-20",

status: "Active",

refillStatus: "Available",

instructions: "Take with or without food, at the same time each day.",

sideEffects: "May cause dizziness or swelling."

},

{

id: 2,

name: "Metformin",

dosage: "500mg",

frequency: "Twice daily",

doctor: "Dr. Michael Chen",

specialty: "Endocrinology",

startDate: "2025-02-10",

endDate: "2025-05-10",

status: "Active",

refillStatus: "Pending",

instructions: "Take with meals to reduce stomach upset.",

sideEffects: "May cause nausea or diarrhea."

}

],

historyPrescriptions: [

{

id: 3,

name: "Amoxicillin",

dosage: "500mg",

frequency: "Three times daily",

doctor: "Dr. James Wilson",

specialty: "General Practice",

startDate: "2024-12-05",

endDate: "2024-12-15",

status: "Completed",

reason: "Full course completed"

}

],

todayMedications: [

{ name: "Amlodipine", time: "8:00 AM", taken: true },

{ name: "Metformin", time: "8:00 AM", taken: true },

{ name: "Metformin", time: "8:00 PM", taken: false }

]

};

**C.f) Lab Report Management**

* **Purpose:** Displays and organizes patient lab results.
* **Components:**
  + **Summary Buttons:** Recent Lab Report, Abnormal Findings, Pending Lab Orders.
  + **Filter Header:** Search, sort (newest/oldest), status (all, abnormal, pending).
  + **Report List:** Table with type, date, doctor, status, dropdown for parameters (e.g., glucose, value, range, status), notes.
* **Sample JSON:**

json

CollapseWrapCopy

const labReports = [

{

id: 1,

date: "2025-03-10",

type: "Blood Test",

status: "abnormal",

doctor: "Dr. Sarah Johnson",

parameters: [

{ name: "Hemoglobin", value: "10.2", unit: "g/dL", reference: "12.0-15.5", status: "low" },

{ name: "Glucose", value: "98", unit: "mg/dL", reference: "70-100", status: "normal" }

],

notes: "Mild anemia detected."

},

{

id: 2,

date: "2025-02-25",

type: "Lipid Panel",

status: "normal",

doctor: "Dr. Robert Chen",

parameters: [

{ name: "Cholesterol", value: "185", unit: "mg/dL", reference: "<200", status: "normal" }

],

notes: "All levels normal."

}

];

**D) Doctor Section**

* **Purpose:** Provides tools for doctors to manage patients and schedules.
* **Shared Component:**
  + **Navbar:** Similar to patient navbar, with notifications and doctor card.
  + **Sample JSON:**

json

CollapseWrapCopy

const doctorNavbar = {

notifications: [

{ id: 1, message: "New Appointment Request", time: "2025-03-23 09:00 AM" }

],

doctorCard: {

profilePic: "{% static 'doc-1.png' %}",

name: "Dr. Emily Smith",

specialty: "General Medicine",

email: "emily.smith@nepals.care"

}

};

**D.a) Dashboard**

* **Purpose:** Gives doctors an overview of their day.
* **Components:**
  + **Header:** Current status dropdown (Busy, Available, Away, Offline).
  + **Summary Divs:** Total Patients, Today’s Appointments (with pending count), Unread Messages, Pending Lab Results.
  + **Appointment Schedule:**
    - Filter Header: Calendar date picker, filter (all, completed, upcoming), search.
    - Slider Calendar: Time slots with patient name, type, status, notes.
* **Sample JSON:**

json

CollapseWrapCopy

const doctorDashboard = {

status: "Available",

summary: {

totalPatients: 15,

todayAppointments: { total: 4, pending: 2 },

unreadMessages: 3,

pendingLabResults: 1

},

appointments: [

{

id: 1,

patientName: "John Doe",

time: "09:00 - 09:30",

date: "2025-03-25",

type: "Check-up",

status: "upcoming",

notes: "New patient"

},

{

id: 2,

patientName: "Jane Smith",

time: "10:00 - 10:45",

date: "2025-03-25",

type: "Consult",

status: "upcoming",

notes: "Follow-up"

}

]

};

**D.b) Patient Management**

* **Purpose:** Allows doctors to manage patient appointments and records.
* **Components:**
  + **Filter Header:** Search, filter (upcoming, completed, canceled).
  + **List Table:** Columns for patient, appointment date/time, status, issue, actions (view, approve, decline).
  + **View Details:** Patient image, name, age, phone, gender, email, appointment details, action buttons.
* **Sample JSON:**

json

CollapseWrapCopy

const patientManagement = [

{

patient: { name: "John Doe", image: "{% static 'patient-1.png' %}", age: 34, phone: "+977-9876543210", gender: "Male", email: "john.doe@example.com" },

appointment: { date: "2025-03-25", time: "10:00 AM", status: "Pending", issue: "Routine Check-up" },

actions: ["View", "Approve", "Decline"]

}

];

**D.c) Edit Your Schedule**

* **Purpose:** Lets doctors set their availability.
* **Components:**
  + **Header:** Weekly hours, save button, stats tooltip.
  + **Tabs:**
    - Weekly Calendar: Grid with time slots (rows) and days (columns), editable statuses (Free, Available, Break, Booked).
    - Settings & Presets: Quick presets (e.g., Morning Shift), break settings.
* **Sample JSON:**

json

CollapseWrapCopy

const scheduleData = {

"2025-03-25": [

{ hour: 9, status: "available", duration: 30 },

{ hour: 10, status: "available", duration: 30 },

{ hour: 11, status: "booked", duration: 30 },

{ hour: 12, status: "break", duration: 30 }

],

"2025-03-26": [

{ hour: 9, status: "available", duration: 30 },

{ hour: 10, status: "available", duration: 30 }

]

};

**Missing Features**

The following planned features are not yet implemented:

* **Message Section:** A messaging system for doctor-patient communication.
* **Video Call Section:** Interface for online consultations.

**What I Have and What I’m Planning**

**Current State**

* **Frontend:** 60-70% complete with a functional UI using dummy JSON data for dynamic rendering across Home, Login/Signup, Patient Section (Dashboard, Appointments, Booking, Documents, Prescriptions, Lab Reports), and Doctor Section (Dashboard, Patient Management, Schedule).
* **Backend:** Basic Django setup with views and URLs for navigation, ready for expansion.

**Planning Moving Forward**

* **Backend Development:**
  + Design database models for Users (Doctors, Patients, Admins), Appointments, Prescriptions, Lab Reports, Documents, Messages, and Schedules.
  + Create APIs using Django REST Framework to replace dummy data with real data, or just send the context in views for other
  + Implement business logic for appointment booking, video consulting, medicine management, and user management.
* **Frontend Enhancements:**
  + Add Message Section with a chat interface.
  + Add Video Call Section integrated with appointments.
  + Replace dummy data with API calls for real-time data.
* **Additional Features:**
  + Integrate payment processing (e.g., Stripe) for consultation fees.
  + Add real-time notifications using Django Channels or similar.
* **Goals:**
  + Ensure a structured backend plan to avoid past planning mistakes.
  + Align frontend with backend for a cohesive system.