Build a complete multipage hospital management website using only HTML, Tailwind CSS, and vanilla JavaScript — no frameworks and no backend server. All data must be stored and managed using LocalStorage.

🏥 Project Title:

IST Medical College & Hospital

🏗️ Core Requirements

- Use HTML + Tailwind CSS for layout and styling, and vanilla JS for all interactivity and data logic.

- ❌ No React, Vue, Angular, or any other JS frameworks.

- ❌ No backend/server — simulate all data using LocalStorage.

- The site must be multipage, with a clear separation between:

    - 🌐 Public Website Pages (information, sign up, login, doctor list, etc.)

    - 🩺 Doctor Dashboard (for managing appointments)

    - 🧑 Patient Dashboard (for booking and managing appointments, invoices, etc.)

- Design theme: blue and white, clean medical look, responsive layout.

🌐1️⃣ Public Website Navigation

A top sticky navigation bar will appear on all public pages.

Left Side:

- Hospital Logo + Name → `IST Medical College & Hospital`

Right Side Menu:

- 🏠 Home (`index.html`)

- 📝 About (`about.html`)

- 👨‍⚕️ Doctor List (`doctor-list.html`)

- 📅 Your Appointment (`appointment.html`)

- 💳 Pay Slip (`payslip.html`)

- 📜 History (`history.html`)

- 🪑 Seat Booking (`seat-booking.html`)

- 💬 Suggestions / Complaints (`suggestions.html`)

- 🔐 Login (`login.html`)

- 🆕 Sign Up (`signup.html`)

These are public pages, accessible to everyone. Some may redirect to login if accessed without being authenticated.

🧭②Dashboard Navigation

After login, users are redirected to their respective dashboard with sidebar navigation, separate from the public site.

👨‍⚕️ Doctor Dashboard (`doctor-dashboard.html`)

Sidebar Menu Items:

- Dashboard Overview

- Today's Appointments

- Appointment History

- Patient Details

- Generate Invoice / Pay Slip

- Suggestions / Complaints (View feedback)

- Logout

Doctors have predefined credentials (stored in JS).

🧑 Patient Dashboard (`patient-dashboard.html`)

Sidebar Menu Items:

- Dashboard Home (Welcome + Summary)

- Book Appointment

- Upcoming Appointments

- Appointment History

- Pay Slips / Invoices

- Seat Booking

- Submit Suggestions / Complaints

- Edit Profile

- Logout

Patients sign up through the public site and can manage their records from the dashboard.

📄 Page Separation Summary

Section	Purpose	Navigation Style	Accessible By
🌐 Website Pages	Public info, signup, doctor list, etc.	Top Navbar	Everyone
🩺 Doctor Dashboard	Manage appointments, invoices	Sidebar / Dashboard Navbar	Doctors only
🧍 Patient Dashboard	Book/manage appointments, history, etc.	Sidebar / Dashboard Navbar	Patients only

- If a logged-in user visits a public page, the Login/Sign Up buttons become "Go to Dashboard".

- Dashboard pages should block unauthenticated access via session checks in LocalStorage.

📁 Required Pages

Public Website Pages:

index.html

about.html

doctor-list.html

login.html

signup.html

appointment.html

payslip.html

history.html

seat-booking.html

suggestions.html

404.html

Dashboard Pages:

doctor-dashboard.html

patient-dashboard.html

JavaScript Files:

/js

├── data.js (initialize doctors, structure)

├── auth.js (login/signup/session logic)

├── appointments.js (appointment handling)

├── dashboard.js (dashboard functions)

├── invoice.js (invoice generator)

└── utils.js (helpers)

💾 Data Management (LocalStorage)

Store and manage:

- Doctors (with credentials)

- Patients (sign up data)

- Appointments (patient ↔ doctor)

- Invoices

- Suggestions / complaints

- Seat bookings

- Login session info

Example structure:

```
{
  doctors: [
    { id: 1, name: "Dr. Rahman", department: "Cardiology", password: "doc123" },
  ],
  patients: [
    { id: 1, name: "John", email: "john@mail.com", password: "1234", appointments: [] },
```

```
  ],

  appointments: [

    { id: 1, patientId: 1, doctorId: 2, date: "2025-10-20", status: "Pending" },

  ],

  invoices: [],

  suggestions: [],

  seatBookings: [],

  session: { userType: "patient", userId: 1 }

}
```

🧾 Invoice / Pay Slip Feature

- Doctors can generate invoices for completed appointments.

- Patients can view/download invoices from their dashboard.

- Invoice includes patient info, doctor info, date, services, amount, and unique ID.

- Use browser's print dialog for PDF download or print.

🎨 Design Guidelines

- Primary colors: `blue-600`, `blue-800`, `white`, `gray-100`

- Clean, modern medical font (sans-serif).

- Tailwind for layout, spacing, components.

- Dashboard pages use card layouts, tables, responsive grids.

- Public pages have a professional hero section, service highlights, and clean doctor list grid.

✨ Extra Features

- Form validation on signup/login and appointment forms.

- Toast/alert notifications for success/error.

- Persistent login using LocalStorage.

- Logout clears session but not stored data.

- 404 error page for broken links.

- Fully responsive across devices.

🧠 Goal:
Produce a fully functional, visually appealing, multipage hospital management website with a public website for info and login, and two dashboards (doctor + patient) for functionality — entirely built with HTML, Tailwind CSS, and vanilla JS, using LocalStorage as the data layer.