▯ PROMPT: Build the Complete Web Admin Dashboard UI for Cybersathi App

You are a full-stack UI/UX engineer, frontend architect, and product designer.
Your task is to **design and develop the complete frontend (web-based Admin Dashboard)** for the *Cybersathi WhatsApp Chatbot System*.

Use the previous requirements of the Cybersathi project (guidelines derived from PS2.pdf + conversation) and implement a full UI that is:

- clean
- responsive
- minimal
- secure
- audit-friendly
- functional for government agents handling cybercrime complaints

Follow all details below.

▯ 1. Pages to Build (Mandatory)

1. Login Page

Purpose: Allow authorized admin officers to access the dashboard.

Requirements:

- Username + password fields
- Show/hide password toggle
- Submit button
- Minimal branding ("Cybersathi Admin Console")
- Left-side informational block explaining system purpose
- Handle invalid credentials error
- Dark/light clean color palette (Blue–Gray theme)

---

2. Admin Dashboard (Main Page)

Purpose: Show an overview of incoming cyber complaints from WhatsApp Bot.

Required UI Elements:

- Top navigation bar (logo + admin profile + logout)
-

Sidebar menu:

  - Dashboard
  - Complaints
  - Ticket Lookup
  - Export Data
  - Settings

- Summary Cards (4):
  - Total complaints
  - New complaints
  - In-progress
  - Closed

- Graphs (optional for now):
  - Monthly complaint volume

- Complaint categories distribution

Data Table of Complaints:
Columns:

- Ticket ID
- Name
- Phone (masked)
- Complaint Type
- District
- Status (color badge: new=blue, progress=yellow, closed=green)
- Date
- Actions (View button)

Filters (top-right):

- Date range
-

District

- Status

- Complaint Type

---

3. Complaint Detail Page

Purpose: Review a single complaint in-depth.

Required Sections:

1. **Citizen Information Card**

    - Name

    - Guardian name

    - DOB

    - Phone

    - Email

    - Gender

    -

Address fields (Village, PO, PS, District, PIN)

2. **Complaint Information Card**

   - Ticket ID
   - Complaint type
   - Sub-type
   - Submission date
   - Current status
   - "Change Status" dropdown
   - Notes section (add note, list previous notes)

3. **Uploaded Attachments Section**

   - Thumbnail grid
   - Click to open viewer
   - Metadata (file type, uploaded timestamp)

4. **Chat Transcript View**

   - Messages displayed like a chat
   - Timestamps
   - Media icons for attachments
   - Auto-scroll container

5. **Authority Actions Panel**

   - Assign officer
   - Mark as priority
   - Close ticket

---

4. Ticket Status Lookup Page

Purpose: Allow admin to quickly search a ticket.

Components:

- Search bar for Ticket ID
- Result section showing:
  - Ticket summary
  - Status
  - Created date
  - Assigned officer
  - Quick link to full detail page

---

5. Data Export Page

Purpose: Export complaints to CSV/Excel.

Components:

- Date-range selector
- Status filter
-

Format selector (CSV/XLSX)

- "Export" button
- Download link after export

---

6. Settings Page

Purpose: Manage system config (for local demo).

Sections:

- Update admin password
- Update WhatsApp API keys (masked)
- Language settings (English/Odia toggle)
- Data retention policy (informational)

⊠ 2. Component Architecture

Core Components to Build

- <Navbar />

- <Sidebar />
- <StatsCard />
- <ComplaintsTable />
- <FiltersPanel />
- <ChatTranscript />
- <AttachmentPreview />
- <StatusBadge />
- <Pagination />
- <Loader />
- <ErrorMessage />
- <Modal />
- <Input />, <Select />, <Textarea />

▯ 3. UX & Branding Guidelines

Color Palette

Primary Blue: #2563EB Dark Blue: #1E3A8A Gray-100: #F3F4F6 Gray-300: #D1D5DB Gray-700: #374151 Gray-900: #111827 Success Green: #10B981 Warning Yellow: #FACC15 Danger Red: #EF4444

Typography

- Use **Inter** or **Roboto**
- Titles: text-2xl font-semibold
- Section headers: text-xl font-medium
- Paragraphs: text-sm

Styling Rules:

- Use **Tailwind CSS** everywhere
- Rounded corners: rounded-xl
- Shadows: soft shadows shadow-md

- Cards: use bg-white + padding

Layout

- Full-height sidebar
- Main content scrollable
- Responsive:
  - Mobile □ Sidebar collapsible
  - Tables □ horizontal scroll
  - Cards stack vertically

□ 4. Security & UX for Government Use

- Mask phone numbers: 98******21
- Mask email: sh***@gmail.com
- Role-based access (basic)

- Prevent unintentional status changes (confirm modal)
- Attachment viewer with download disabled
- Add loader animations for heavy data

---

5. Integration Expectations

Backend endpoints assumed:

GET /admin/complaints GET /admin/complaints/:ticket_id POST /admin/complaints/:ticket_id/status GET /admin/export POST /admin/login

Frontend must:

- Fetch complaint list
- Apply filters via query params
- Fetch detail page on click
- Post status updates

- Download exported files
- Display errors gracefully

□ 6. Folder Structure (Vite + React + Tailwind)

/src /components /pages Login.jsx Dashboard.jsx Complaints.jsx ComplaintDetail.jsx TicketLookup.jsx Export.jsx Settings.jsx /hooks /utils /styles

---

□ 7. Final Output Requirements

Replit AI must deliver:

□ Complete responsive UI

□ All pages & components listed

□ Fully functioning admin console

□ Fully styled Tailwind implementation

□ Integrated with backend APIs (use placeholders if needed)

□ Commented, production-ready code

□ Clean, consistent UI following the Cybersathi theme