## **1. Patient List View Screen**

https://www.figma.com/design/qaC2rkSu6YUKMixLw4PyaL/Sample-UI?node-id=26-33645&t=yYPZDYY6ERACNaL9-4

First remove the existing content from App.tsx

Create a Patient List View component with following details and add in App.tsx:

TECHNICAL REQUIREMENTS:

- Use React functional components with TypeScript

- Implement React hooks (useState, useEffect, useMemo)

- Use React Material UI

- Include proper prop types and interfaces

- Add loading states and error handling

- Implement pagination (10 patients per page)

- Add search functionality with debouncing (300ms)

- Include sorting by Date of birth and Last Visit

DESIGN SPECIFICATIONS:

- Use color codes from figma only

- Table layout with columns: Patient Id, Avatar & Name, Date of Birth, Contact Details, Last Visit, Actions

- Action buttons: View, Edit, Delete with icons

- Search bar at the top with filter dropdown

- Pagination controls at bottom

- Add Patient button (primary CTA) in top-right

RESPONSIVE DESIGN:

- Desktop Full table view with all columns

- Design should be responsive

DATA STRUCTURE:

interface Patient {

id: string;

name: string;

dateOfBirth: string;

contactDetails: string;

lastVisit:string

}

Include mock data for 15+ patients and implement all interactive features.

Show 10 Patients per page

Special Note:

Do not add any additional fields

Use color codes from figma only

Patient List View should occupy full width, check for and remove any max-width, display: flex, or centering styles on the body, #root, or parent containers.

## **2. Add Patient Modal/Screen**

Design and implement an “Add Patient” modal form in React + TypeScript, matching the following Figma design:

@<https://www.figma.com/design/qaC2rkSu6YUKMixLw4PyaL/Sample-UI?node-id=26-34402&t=yYPZDYY6ERACNaL9-4>  
  
When I click On Add Patient button this modal should get open   
If I click on Save patient that patient should appear in Patient listing at top

UI/UX & LAYOUT REQUIREMENTS

Single-step modal: All fields visible at once (no stepper, no multi-step logic).

Modal shape: Horizontally oriented, wide and short (not tall/vertical).

Width: 80%

Height: Max 80vh, with vertical scroll if content overflows.

Padding: Ample horizontal padding (e.g., 56px left/right), less vertical padding.

Header: Bold title (“Add Patient”) left-aligned, close (X) button right-aligned, with a bottom border.

Form layout:

Two-column grid for fields on desktop (use CSS grid).

Single column on tablet/mobile.

Consistent spacing between fields (e.g., 28px row, 40px column).

Buttons: “Submit” (primary, right), “Cancel” and “

Avatar upload: Drag & drop or click to upload, with preview.

Clean, modern look: Subtle shadows, 16px border-radius, light gray backgrounds, Inter font, etc.

Responsive: Modal and form adapt to all screen sizes.

DATA STRUCTURE:

interface Patient {

id: string;

firstName: string;

middleName?: string;

lastName: string;

dateOfBirth: string;

gender: 'male' | 'female' | 'other';

maritalStatus: 'single' | 'married' | 'divorced' | 'widowed';

timezone: string;

language: string;

ssn: string;

race: string;

ethnicity: string;

profilePicture?: string;

}

FUNCTIONAL REQUIREMENTS

React functional component with TypeScript

Use React Hook Form for form state/validation

Yup for schema validation

Field masking for phone and date fields

Auto-save draft to localStorage

File upload for avatar (jpg/png, max 5MB)

Proper form submission with loading state and success/error toasts

Validation: All required fields, email/phone/date formats, file size/type, real-time feedback

On submit: Add new patient to the list in the parent component and close the modal

FORM FIELDS

Personal Info: First Name, Middle Name, Last Name, Date Of Birth\* (MM/DD/YYYY), Gender\*, Marital Status, Timezone, Language, SSN, Race, Ethnicity

ACCESSIBILITY & INTERACTION

Keyboard navigation: All fields and buttons accessible via Tab

Close modal: X button, overlay click, or Esc key

Disabled submit until form is valid

Error styling: Red border and message under invalid fields

Success confirmation: Toast or modal after submit

DO NOT

Do not use a stepper or multi-step logic

Do not stack all fields vertically on desktop

Do not use a narrow or tall modal

Summary:

> “I want a single-step, horizontally-aligned, wide Add Patient modal, matching the Figma design, with all fields visible at once, two-column layout, and all the modern UX, validation, and accessibility features described above.”

## **3. Patient Details View Screen**

https://www.figma.com/design/qaC2rkSu6YUKMixLw4PyaL/Sample-UI?node-id=26-34734&t=7gu5EFTLf05acDyX-4

Create a comprehensive Patient Details view component

When I click on eye button in patient listing for patient then Patient View Modal should open

TECHNICAL REQUIREMENTS:

- React functional component with TypeScript

- Implement lazy loading for heavy data sections

- Add print functionality

- Add breadcrumb navigation

LAYOUT SECTIONS:

1. View Screen fields : First Name, Middle Name, Last Name, Date Of Birth\* (MM/DD/YYYY), Gender\*, Marital Status, Timezone, Language, SSN, Race, Ethnicity

DESIGN SPECIFICATIONS:

- Card-based layout with clear visual hierarchy

- Profile header with large avatar and key patient info

- Tabbed interface with lazy loading of tab content

- Action buttons: Edit Profile

RESPONSIVE DESIGN:

- Desktop: Sidebar navigation + main content area

DATA STRUCTURE:

interface Patient {

id: string;

firstName: string;

middleName?: string;

lastName: string;

dateOfBirth: string;

gender: 'male' | 'female' | 'other';

maritalStatus: 'single' | 'married' | 'divorced' | 'widowed';

timezone: string;

language: string;

ssn: string;

race: string;

ethnicity: string;

profilePicture?: string;

}

Include loading skeletons, error boundaries, and offline functionality indicators.

## **4. Edit Patient Modal/Screen**

Create an Edit Patient form component that extends the Add Patient functionality:

TECHNICAL REQUIREMENTS:

- Inherit from Add Patient component but with pre-populated data

- Implement optimistic updates for better UX

- Add change tracking to highlight modified fields

- Include confirmation dialog for unsaved changes

- Implement partial updates (only send changed fields)

- Add audit trail logging for medical compliance

- Handle concurrent editing scenarios

ADDITIONAL FEATURES:

- Pre-fill all form fields with existing patient data

- Highlight changed fields with subtle visual indicators

- Show "last modified" timestamp and user info

- Add "Revert Changes" button to reset to original values

- Implement auto-save every 30 seconds for drafts

- Add version history dropdown (last 5 changes)

DESIGN SPECIFICATIONS:

- Same layout as Add Patient but with "Edit Patient" title

- Changed fields indicator (subtle left border or background)

- Last modified info at top of form

- Additional action buttons: Revert, Save, Cancel

- Confirmation modal for destructive actions

- Visual diff indicator for changed vs original values

VALIDATION & SECURITY:

- Same validation rules as Add Patient

- Additional check for data conflicts (if record was updated by another user)

- Implement optimistic locking mechanism

- Show warning if patient has appointments scheduled before major changes

RESPONSIVE DESIGN:

- Same responsive behavior as Add Patient form

- Additional consideration for change indicators on mobile

- Swipe gestures for mobile version history

DATA FLOW:

- Fetch current patient data on component mount

- Track changes in separate state object

- Implement undo/redo functionality

- Show unsaved changes warning on navigation

Include proper error handling for network failures and data conflicts.