

Frontend-setup

LabLens Frontend

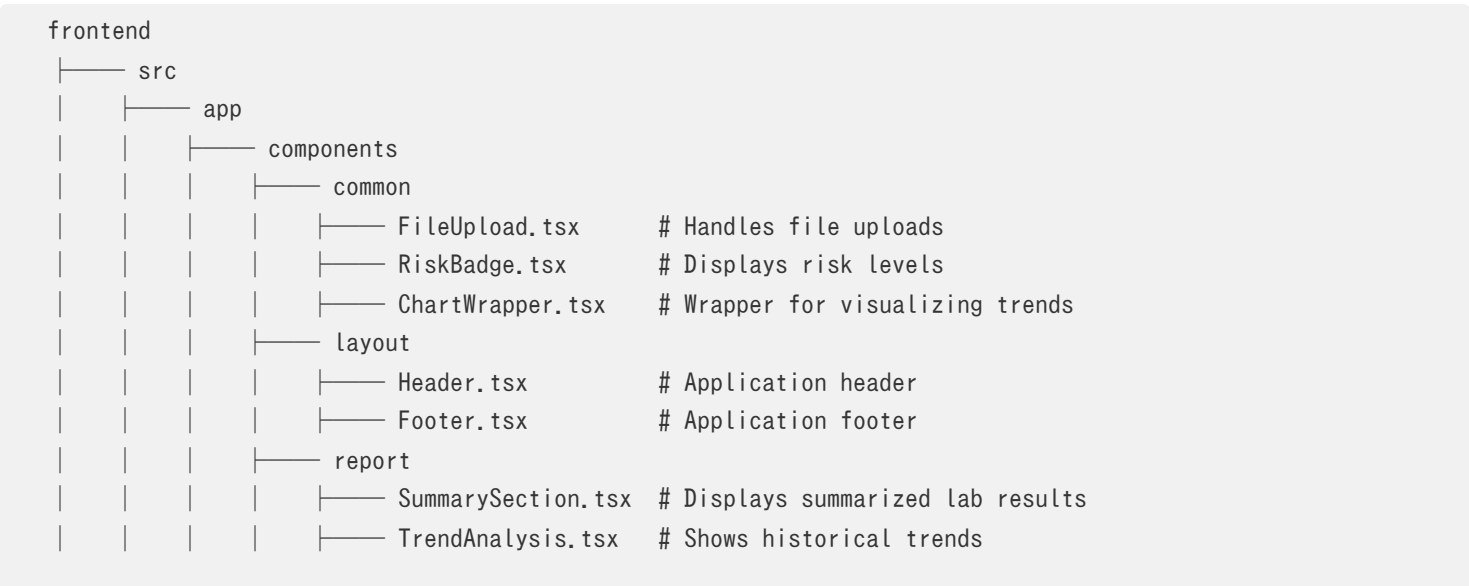
Overview

LabLens is an AI-powered medical report analysis system that helps users analyze lab reports, detect abnormalities, generate health insights, and provide dietary recommendations based on lab findings. This project is built using **Next.js 13+** with the App Router structure.

Features

- **PDF Upload & Analysis:** Users can upload lab reports in PDF format for AI-based analysis.
- **Health Insights & Trends:** AI interprets lab results, detects abnormalities, and generates health recommendations.
- **Diet Planning:** Personalized dietary suggestions based on lab findings.
- **User Authentication:** Secure login/signup system using NextAuth.js and Google authentication.
- **Interactive Dashboard:** Displays historical lab reports, downloadable PDFs, and trend visualizations.
- **Conversational AI:** ElevenLabs ConvAI integration for AI-powered interactions.

Project Structure



```
| | | | | lib
| | | | | | api
| | | | | | | client.ts      # API request configurations
| | | | | | | endpoints.ts   # API endpoint definitions
| | | | | | | hooks
| | | | | | | | useReportData.ts # Hook to fetch and manage report data
| | | | | | | | useFileUpload.ts # Hook to manage file uploads
| | | | | | | types
| | | | | | | | report.ts      # Type definitions for reports
| | | | | | styles
| | | | | | | globals.css     # Global styles
| | | | | | | tailwind.config.js # TailwindCSS configuration
| | | | | | public/assets/images # Static images
| | | | | | .env.local         # Environment variables
| | | | | next.config.js      # Next.js configuration
| | | | | package.json        # Dependencies
| | | | | tsconfig.json       # TypeScript configuration
| | | | | jest.config.js      # Jest testing configuration
```

Key Components

Landing Page ([page.tsx](#))

- **Features:** Displays key features of the platform, including AI-based lab analysis, diet planning, and trend visualization.
- **ElevenLabs ConvAI:** Integrates AI chatbot for interactive user experience.
- **Animated UI:** Uses Framer Motion for smooth animations.

Dashboard ([dashboard/page.tsx](#))

- **Lab Report History:** Lists previous reports and provides download/view options.
- **PDF Upload:** Users can upload new lab reports for analysis.
- **Search Functionality:** Allows searching reports by filename.

Diet Plan ([dietplan/page.tsx](#))

- **Personalized Meal Recommendations:** Based on AI-analyzed lab reports.
- **General Nutrition Tips:** Displays additional health advice.
- **Tab-Based UI:** Users can switch between recommendations and general nutrition tips.

Report Analysis ([report/page.tsx](#))

- **Detailed Report Summary:** Displays extracted lab data with highlighted abnormalities.

- **AI Recommendations:** Suggests lifestyle changes, immediate actions, and follow-up tests.
- **Trend Analysis:** Shows historical lab trends using visual graphs.
- **Audio Summary:** Uses browser speech synthesis to read the report.

Authentication (`auth/page.tsx`)

- **Login & Signup Forms:** Email/password authentication and Google OAuth.
 - **OTP Verification:** Secure sign-up with one-time password verification.
 - **Animated Forms:** Framer Motion-based UI transitions.
-

Setup & Installation

1. Clone the Repository

```
git clone https://github.com/your-repo/lablens-frontend.git
cd lablens-frontend
```

2. Install Dependencies

```
yarn install # or npm install
```

3. Set Up Environment Variables

Create a `.env.local` file and configure API keys:

```
NEXT_PUBLIC_API_BASE_URL=https://your-api-url.com
NEXT_PUBLIC_GOOGLE_CLIENT_ID=your-google-client-id
NEXT_PUBLIC_GOOGLE_SECRET=your-google-secret
NEXT_PUBLIC_ELEVENLABS_AGENT_ID=LeQc9M40BavEckRhBEZb
```

4. Run Development Server

```
yarn dev # or npm run dev
```

The app will be available at <http://localhost:3000>

5. Build & Deploy

```
yarn build # or npm run build
yarn start # or npm start
```

Technologies Used

- **Framework:** Next.js 13+ with App Router

- **UI Styling:** Tailwind CSS
 - **State Management:** React Hooks
 - **Authentication:** NextAuth.js with Google OAuth
 - **API Integration:** Fetch API for communicating with the backend
 - **AI Integration:** ElevenLabs ConvAI for interactive conversations
 - **Visualization:** Framer Motion & custom charts for trend analysis
-

Future Enhancements

- **Enhanced AI Recommendations:** Improve Flan-T5 model predictions.
- **More Data Visualizations:** Advanced graphs for deeper health insights.
- **Multi-User Support:** Enable family accounts for tracking multiple users.