Techstack:

I use a monolithic repo that uses NextJS for serving both the frontend and backend, and tailwind.css for frontend styling.
This project serves as a simple standalone application that once running, handles both frontend and backend requests and does not rely on any external DB or third party API.

Approach:

I focused on the frontend and achieved all the 3 functionality requirements.
In order to provide a simple web app that listens to the user, displays notes based on the audio, and allows manual input for the notes, I designed the FE to have the following components:

1. **Audio recorder:** responsible for collecting audio data and visualising audio input from the user
2. **NotePad:** responsible for editing and displaying notes. Notes includes both user's hand written notes and LLM generated notes.
3. **Root page:** responsible for assembling the inner components, managing the exchange of data/states between inner components, and potentially manages high level states if necessary.

For the BE, I wrote a simple endpoint that takes in the raw audio data and synchronously returns a mock AI-generated text as the session note.

Assumption:

1. Recording duration is short
2. LLM processing time is short
3. Network is very stable
4. User doesn't close the tab during using the web app
5. User uses mainstream web browsers