

Engineering Intern Hiring - Assignment



Build a Note-Taking Web App, using React.js, that accepts text, URL, and audio as input and transcribes it to text using [Browser Web Speech API](#) and then briefly explain your Code in a video presentation.

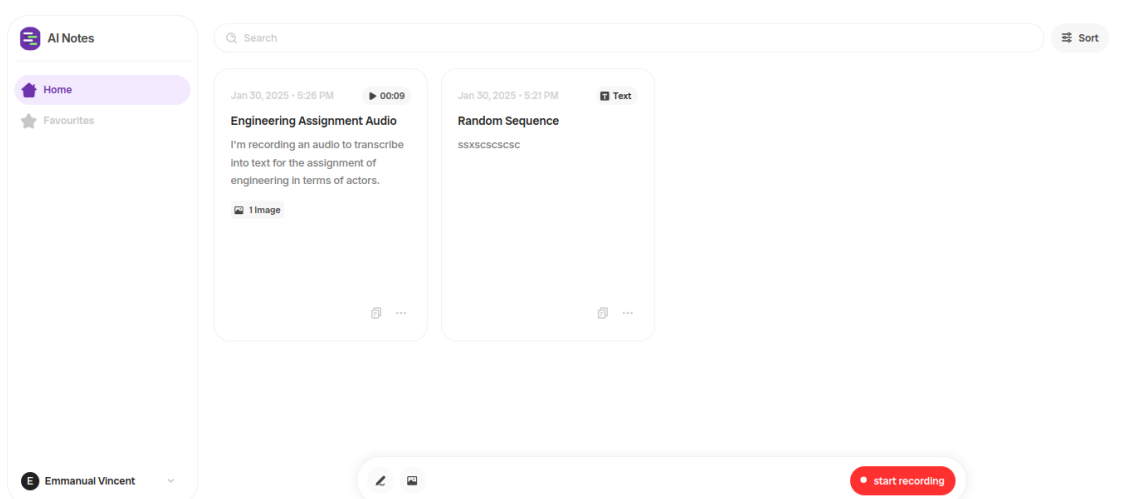
Note: Please make sure you have talked to this [Fullstack Intern qualification chatbot here](#) before working on this Coding challenge.

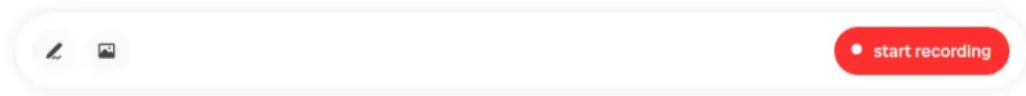
Functional Requirements:

Please build a single-page React.js app with the following specs:

1. It is mandatory to build the UI using Nextjs or React, feel free to use a React UI library like [Shadcn](#), [Tailwind](#), or any other of your choice to develop the project and match the design as closely as possible.

2. Use ChatGPT, Cursor, or any other AI tool to help you generate the code, **it is imperative that whatever code you generate is structured and its logic well understood by you**. So much so that any modification to the app can be done by you with ease during the interview.
3. The App should have authentication using JWT, both sign-up and login should be functional.
 - The design can be kept simple, focusing solely on ensuring the required fields and functionality work as expected.
4. The main page of the app should display all the notes of the user, these notes should be stored in MongoDB and the note list should be sortable from old to new.
5. Note Creation
 - a. A note can be created using a text, and audio record button
 - b. When you click on the start recording button it should record your voice (max duration 1min) and when you stop recording it should transcribe the audio into text using Browser Web Speech API check this [documentation](#) and create a note using the transcribed text.
 - c. The design for this is on the Main Page at the bottom.



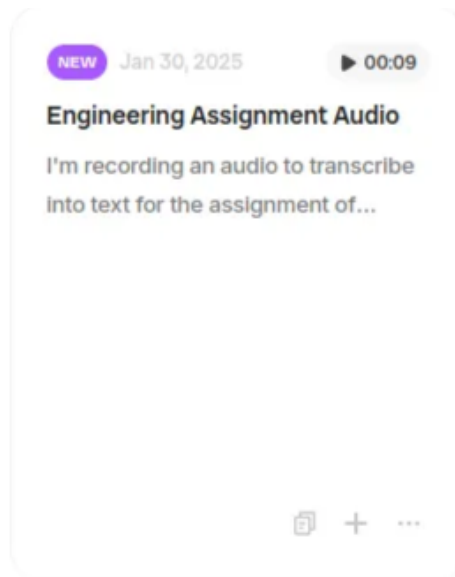


6. Search Feature

- a. There should be an option to search notes at the top of the title and the content of the note should be searchable across all the notes a user has.
- b. Typing anything in the search bar should update the page view with the note search results in real-time

7. Note Card Functionality

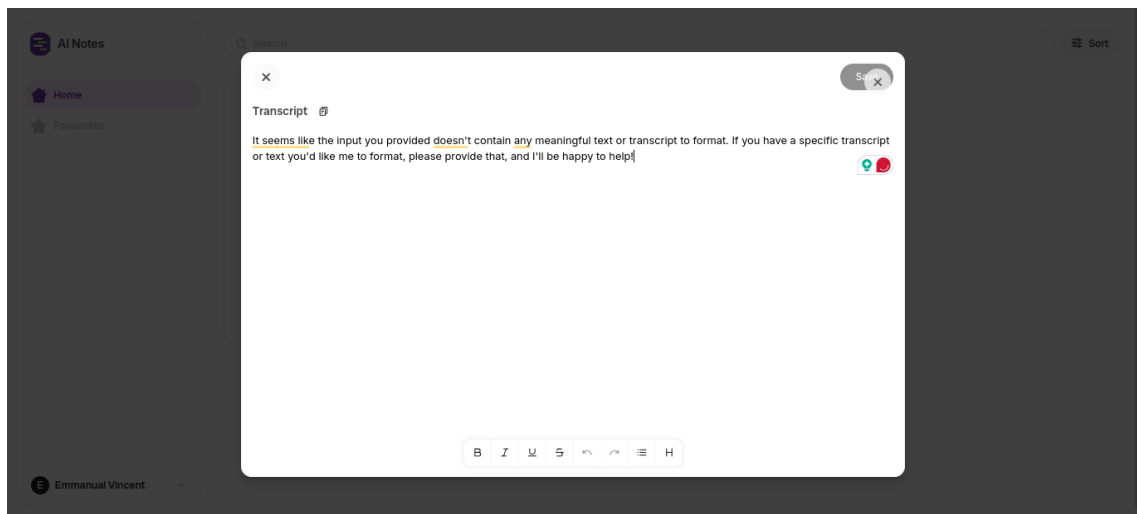
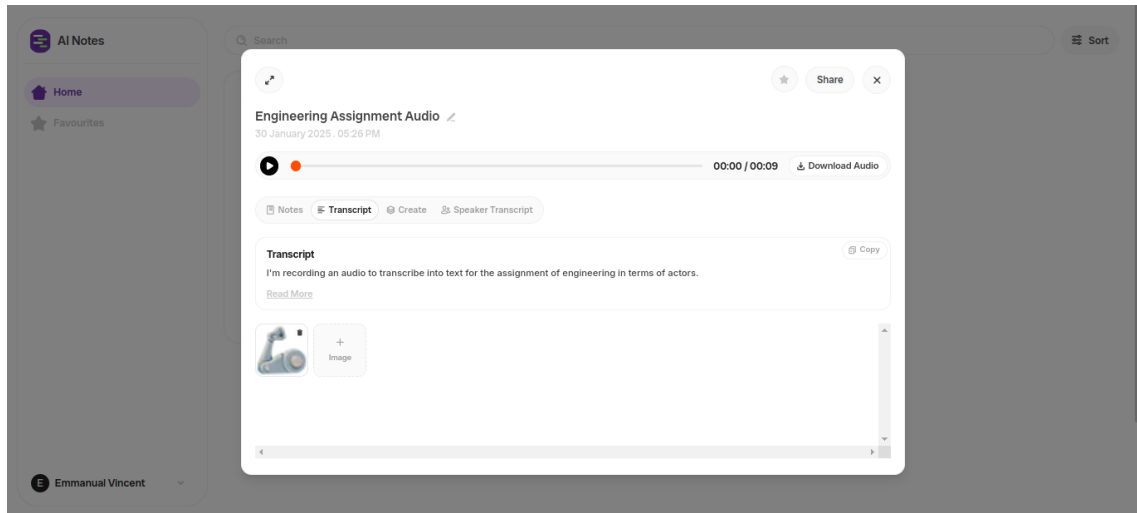
- a. Copy to clipboard button
- b. Delete note button
- c. Rename note button



8. On clicking on any note card

- a. It should display a modal that has the content of the note
- b. If it is an audio note it will have the transcription
- c. Ability to edit the note

- d. Button to make the modal go fullscreen.
- e. Button to favorite your note.
- f. Ability to upload an image in the note.



Video Explanation & Presentation:

1. Once done, you need to make a video explanation using the Loom video recorder. and explain your Application Code in a short 5-minute video. You can explain the library you used, the design pattern, the react states/hooks used, etc.
 - a. ***Please give a short 30 sec. introduction about yourself at the beginning.***

- b. ***Make sure that your camera is on and your Face is visible in the video when explaining your Application Code.***

Evaluation Criteria:

Your submission will be evaluated on the following parameters

1. Output - Ease of use, user experience, and performance of the app.
2. Code Quality - Ease of understanding and readability of the code. use of proper data structures, modular and reusable functions, easy-to-understand variables, and function names.
3. Presentation - How do you communicate & explain your code in the Loom video.
4. Speed - The time you take to build and deliver the app.
5. Creativity - feel free to add extra features/functionality. You can also add more features after the 1st submission and re-submit on the same email thread.

Hosting & Submitting the Application Code:

1. You need to host the code on codesandbox.io, run it, and test the app working on their web IDE.
 - a. ***Please make sure that the Codesandbox App Editor link is shared and publicly readable. open the link in incognito to make sure, before submission.***
 - b. ***You can host your code on your Github account and import it into codesandbox.io easily.***
2. Once done, you can send the publicly available Codesandbox link and the link of the Loom video where you have explained your code to the below emails:
 - Email: vinit@hellotars.com , emmanual@hellotars.com (send to both)
 - Subject: [Fullstack Intern Code Challenge Submission](#)

- Body: Should include
 - **Your Full Name**
 - **Your LinkedIn profile link** (Create it if you don't have it)
 - **Link for the Codesandbox App Code** (Publicly Readable)
 - **Link of the Loom Video**
 - (Optional) Any other relevant social links like Github, Medium, Dev.to, etc.

Notes:

1. We think that ideally, you should be able to build this app in 4-5 hours. It would be great if you could let us know the time it took you to build it in the email itself.
2. Once you submit the code, we will review the app functionality and the Code itself and will respond to you within 3-5 days about the next steps.
 - If qualified for the next round, we will set up a 45-minute Google Meet call with you to talk about you and we will also walk through your code.
3. If you have any specific questions about the Coding Challenge, You can drop me an email at vinit@hellotars.com with the subject: Query about Tars Code Challenge