Dear Deepgram Engineering Candidate,

In preparation for your upcoming interview, we would like you to prepare the following coding project.

> Build a simple API server to handle user audio projects. Your server should provide endpoints that allow a user to perform the following actions:
>
> 1. POST raw audio data and store it.
>   Eg: `$ curl -X POST --data-binary @myfile.wav http://localhost/post`
> 2. GET a list of stored files, GET the content of stored files, and GET metadata of stored files, such as the duration of the audio. The GET endpoint(s) should accept a query parameter that allows the user to filter results. Results should be returned as JSON.
>   Eg: `$ curl http://localhost/download?name=myfile.wav`
>   Eg: `$ curl http://localhost/list?maxduration=300`
>   Eg: `$ curl http://localhost/info?name=myfile.wav`

When you arrive for your interview, we will ask you to present your code and elaborate on some of the design decisions you have made, some of the things you would have done differently if you had more time, some things you learned about libraries you used, … Additionally, we will ask you to build upon the code you've worked on to add functionality.

Your code should minimally be able to handle the GET and POST requests described above. However, because we want this to be a point of departure for a slightly bigger project you will do during your in-person interview, here are some features and questions you might want to think about:

1. How to handle user authentication and data security?
2. How to build a simple browser UI to interface with your API?
3. How do you want to store audio data? For the purposes of this interview, just keeping them in memory is fine, but how else would you want to keep and serve audio data?
4. How to handle data integrity? How to make sure that users can't break your API by uploading rogue text data? How to make sure the metadata you calculate is correct and not thrown off by unmet expectations on the backend?

Thanks for your time and your earnest effort on this small project. We look forward to seeing what you've done.

The Deepgram team