Technical Assessment: Technical Mentor

Date issued: 25th February, 2021
Submission Date: A week after receiving.

Submission summary: After working on this project, you will be submitting the following:

1. A project documentation
2. A link to the github repository where you stored your project (for both Frontend and Backend).
   (NB: Copy pasting code is forbidden. Incase of any copy pasted code, you'll be disqualified immediately)
3. A link to the live application with all the intended capabilities/functionalities.

Below is a detailed description.

Introduction

As described in the Job Description shared with the application form, the mission of the Technical Mentor is to, "To provide a safe space for participants to drive their own learning by removing any obstacles on their learning path, being observant and jumping in to guide when needed. You will be the point of truth when some issues arise between peers, or peers with staff."

We are looking for a doer. A person who can thrive and grow both technically and professionally. Someone who deeply cares about the success of learners and would do everything within their ability to make sure we have satisfied customers. This assessment will give us a better sense of your alignment with our mission and a deeper sense of your skillset and fit for the role.

Context and overview

As mentioned above, you will be responsible for developing a mini application that will give us a clear idea of your technical ability. The role itself doesn't require you to develop any application but we want to be sure that you have what it takes to support very junior individuals who are at the early stages of your journey into software engineering.

Overview of the application:

- Section 1: An overview of the project and platform choice related questions
- Section 2: Setup and development of the Backend (backend, exposing apis, e.t.c)
- Section 3: Development of the frontend (connecting it with backend and integrating a 3rd party API)
- Section 4: Deployment of the application and testing live application

- Section 5: Creating a presentation/project documentation and submitting project (including the **documentation/presentation** link, **github link** of the application and a **link** to the live application)

Section 1: An overview of the project and platform choice related questions

Scenario: As a Technical Mentor, you've been requested to develop a mini-app of an information management system. The application should be able to:

1. Signup or login a user
2. Have at least one page (ie Home page which will display data from 3rd party api - more details in section 3)
3. The application should be fully decoupled (The frontend and backend be in different codebase). We highly recommend Quasar or React for frontend, and Nodejs, Django or Laravel for Backend. You're free use any other frameworks of your choice.

Reflection Question: (You can respond during your presentation/documentation)
What informed the choice of your frameworks? What informed the choice of your database?

Section 2: Setup and development of the Backend (backend, exposing apis, e.t.c)

Given the description of the platform above, go ahead and develop the Backend of the application. The Backend should: Be fully decoupled (not monolithic), with authenticable apis, a database for data storage.

Reflection Question: (You can respond during your presentation/documentation)
Are we able to test and authenticate your apis using postman?
Which authentication did you use?
Which database did you use? What's the schema?

Section 3: Development of the frontend (connecting it with backend and integrating a 3rd party API)

Given that you have a functional backend in place, go ahead and develop a frontend that will be used by an application user. The frontend (which will be fully decoupled from the backend) should allow the user to:-

- Signup and Login to view the Homepage. And logout when done
- Home page should have two sections:
    - 1: Fixed-source: A page section made of cards or other appropriate widgets to that display data from from the api: https://official-joke-api.appspot.com/jokes/ten

- 2. Random-sources : A page section made of appropriate widgets that displays data from the api below (Note: You'll go through the documentation to integrate the API): https://kitsu.docs.apiary.io/#introduction/json:api
  Note: You don't have to display all the variables from the API. Select what's appropriate for you.

Section 4: Deployment of the application and testing live application

Build and deploy your application to be accessible via a live link. FYI: There are free hosting and deployment platforms you can use. Take advantage of those if you don't have a server.

> **Reflection Question**: (You can respond during your presentation/documentation)
> Describe and demonstrate the types of tests that you used?
> Is your application functional, secure and scalable? Why or why not?

Section 5: Creating a basic documentation of the application project

Prepare a basic documentation of the application not exceeding two pages or four slides to describe your implementation process. You can also respond to the reflection questions provided per section.

All the best.