Full Stack Engineer Take Home Assessment

Technical Take-Home Coding Assessment

Overview

We are excited to present you with this take-home assessment, which we've designed to showcase your abilities in developing a proof-of-concept (PoC) software system. At Abode, we are often tasked with building and delivering features with a great deal of autonomy. Your ability to make design decisions and reason through solving problems in the face of vague or open-ended requirements is just as important as your technical skills!

Your task is to create a simple yet effective and functional calendar event management system, leveraging React for the front-end and a suitable back-end technology of your choice for API development. This system should enable CRUD operations for events and incorporate a notification system to alert invitees before the event starts.

Objectives

Your PoC should aim to fulfill the following requirements:

Front-End Interface

- **Technology**: Use React to develop the front end.
- **Features**:
  - **Create Event**: Allow users to create an event by specifying details such as the event name, description, date, time, and invitees.
  - **Read Event**: Display a list of upcoming events, with the ability to view specific details for each event.

- **Update Event**: Enable editing of event details.
- **Delete Event**: Allow the deletion of events.

Back-End API

- **Technology**: You are free to choose the back-end technology (Node.js is recommended for its compatibility with React and ease of setup. We use NestJS with Typescript at Abode). You are free to choose how you want to implement a persistence layer.
- **Features**:
  - **CRUD Operations**: Develop APIs that facilitate creating, reading, updating, and deleting events. Ensure your API can handle requests effectively and return appropriate responses.

Notification System

- **Functionality**: Implement a notification system that sends out alerts to the list of invitees 30 minutes before their event is scheduled to begin.
- **Implementation**: You have the freedom to decide how notifications are sent (e.g., email simulation, console log, etc.). The key is to demonstrate the logic and mechanism behind scheduling and sending notifications.

Guidelines

- **Code Quality**: Write clean, modular, and maintainable code.
- **Scalability**: Design your solution with scalability in mind. Consider how your PoC could be expanded into a full-scale application.
- **Testing:** Tests, while not required, are certainly encouraged.
- **Documentation**: Include a README file that provides:
  - An overview of your solution.
  - Instructions on how to set up and run your project.
  - A brief discussion of your design choices and what you would have done with more time.

Evaluation Criteria

Your submission will be evaluated based on the following criteria:

- **Functionality**: Does the application work as required?
- **Code Quality**: Is the code clean, well-organized, and properly commented?
- **Design and Architecture**: How well thought out was the software design and architecture?
- **Creativity and Innovation**: How creatively did you approach the project? Were there any unique features or technologies used?
- **Scalability and Performance**: Is the system designed to scale? Were performance considerations taken into account?

Submission

Please provide a GitHub link (or equivalent) to your project repository. Ensure your repository is public or shareable with specific accounts (please include account details in your submission email if required).

After submission, your work will be evaluated by our Engineering team. If it meets requirements and demonstrates solid coding practices, you will be invited to a 1 hour second stage interview wherein members of the Engineering team will conduct a code review, ask questions about your solution, and generally discuss implementation details and technical approach. There will also be time for you to ask the team any questions you may have.

Deadline

Your completed assessment should be submitted within 1 week, ideally. We don't want you to spend an inordinate amount of time on this; we understand you are busy and a takehome represents a time commitment. Our goal with this assessment is to understand the tradeoffs you make, what you feel comfortable "hand-waving" away, and how you deliver a proof of concept system to illustrate an idea that can then be built into a production-ready feature.

If you have any questions, need to discuss timeline, etc. please feel free to reach out to james@abodehr.com