FULLTIME FORCE

Take-Home Test

Objective

The goal of this test will be to create an app that shows git commit history on the same project you are working on. You need to do well-stated and defined commits while working on this test and regularly push them up.

You will need to sign up for a free GitHub account if you do not already have one, then create a public repository to store your work so that we will be able to see and review it. You can find documentation on the GitHub API here: https://developer.github.com/v3/.

Choose the frontend and backend technologies that best suit your skill set in order to build the solution, Angular, Vue, or React frameworks or even Flutter, Swift or Android, if you like Mobile development. Pair them up with NodeJS, Python, or .Net, to create the perfect backbone.
Use all the technologies and skills that you want to showcase, but don't forget you must show on UI the methods that were worked on the backend.

Requirements

1. You must use the GitHub API – while you could make a scraper or devise another method to get the git commit history of your project, we would like to see how you work with an existing API.
2. Your GitHub repository must be public – this will allow us to look at it and verify the accuracy of your code. Once you complete this project, you will need to send over the link of the project to me (take-home-test@fulltimeforce.com).
3. The style of the page is up to you. Typically we use something like Tailwind CSS for a lot of our projects, then style it to meet the business want or need later on.
4. The framework we usually use for the backend is NestJS. You can use any other framework if you want but avoid Express.
5. The page does not need to be a real-time view (it's okay for us to refresh it to get the latest commits on your repository).
6. You MUST provide instructions in the README on how to install and run your project. We will not infer anything, and if you do not include these instructions, we will reject your submission.
7. We measure the results based on completeness, code cleanliness, technology assortment, software design and instructions quality given by the developer.

Try the best you can!!!
Good Luck!