ScreenCloud

Engineering Interview Challenge

There is one key element to this challenge, and it's been designed thoughtfully as follows;

1. We'd like to see how you respond to these tasks but **we're not looking for perfection,** neither are we looking for you to get all the details exactly right.
2. While of course there is an element of assessment (this is part of an interview process after all!) please enjoy this task, **show us who you are and how you think.**

Please take a look at this challenge, and send it back to us within a week. If for any reason you need more time, please let us know as soon as possible.

Backend Challenge (SwiftCloud)

Copious levels of user research have highlighted a blank space in the market for us.

Everything changed after 2020 with us all spending more time inside and so music streaming has gone through the roof.

We want to build an app that lets users spot the trends and quirks about what they've been listening to most.

...Ready for it?

SwiftCloud!

The #1 app for Taylor Swifties!

We've pulled together data on all of Taylor Swift's songs, along with play counts for a user in the past 3 months:

https://docs.google.com/spreadsheets/d/1iNGwJWu4ghwM_jP3U81SRU9oneYqN4DTjW7j9t3IMh8/edit?gid=619956793#gid=619956793

Product & Design are still figuring out the frontend, but we know this is going to make sparks fly with our users. May even be 22 releases to come!

Your task is to design and implement a flexible API which exposes this data in different shapes and queries to enable an awesome frontend.

Some examples could be:

- What songs were written in a given year?
- Which songs or albums were most popular last month? What about over all months?

- Searches and sorts?

Feel free to mix and match the data any way you can to enable interesting queries.

We don't want any bad blood in the team later though, so don't forget to document the APIs for your future user and check it will work as expected when the team comes to hit it.

Technical Notes

- Ideally we'd like to see JavaScript or TypeScript. You can write in another language if you haven't worked with these before, but the vast majority of our work at ScreenCloud is in JS/TS so you'd need to pick it up when you join.
- The data is a public sheet so you can query Google Sheets directly, or you could import a CSV inside your project, or even migrate it into a full database. Entirely up to you!
- Deploying and running your API is as important as writing it. Brownie points if you host your app somewhere (which we can then chat about in the interview too) and for code we'd be happy to maintain in a real app.
- We know ~200 songs doesn't strictly need this level of server-side logic. Assume it could grow into 2 million songs some day, Taylor does seem to write a lot!

Guidance (Taylor's version)

The challenge is a chance for you to show off, so pick the tools/frameworks/libraries etc that you know best. We've deliberately asked you to work on this at home so you aren't under any fake pressures and can use the laptop/editor etc that you like best.

- Please put your finished work on Github, or if you need it to be private, please zip your folder, upload it to https://wetransfer.com/ and email us the link.
- Include a readme with setup instructions and anything else you'd like us to know.
- Please make sure your project is easy to run. We don't mind a little setup, but if you don't get time to host it, we'd really appreciate not needing to jump through too many hoops to run locally.
- Don't feel you need to spend forever on this. We hope the challenge touches a few different areas and lets you show off some creativity in each. When you reach a point you're happy with, just let us know in the readme what you would do next if this were a real project.
  - e.g. a few tests in a key area would show your approach to testing, no need to cover the whole application.

If you have any questions, just email imogen.king@screencloud.io & when you have completed the task please send it to her.

Meanwhile I'm just gonna shake, shake, shake.... 🕺 💃