ScreenCloud

Staff Engineer
Interview Challenge

This challenge is designed thoughtfully as follows;

1. First and foremost, by completing this task we intend to give you a **realistic preview of the Staff Engineer opportunity** at ScreenCloud and the sorts of tasks that you are likely to encounter - this is really important to us!
2. We'd like to see how you respond to the tasks that would come with this work, but **we're not looking for perfection**, neither are we looking for you to get all the details exactly right - you aren't (yet!) a ScreenClouder, therefore we don't expect you to have the inside scoop.
3. **Time Frame**: 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. We expect you to fit this around your life and recommend a timeframe of 4 hours. We set a deadline of a week.

ScreenCloud Order Management System

ScreenCloud's SCOS device is becoming increasingly popular, and to make sure we can handle future demand, we have decided to introduce a new order management system for our sales team.

SCOS Device information

Name	SCOS Station P1 Pro
Price	$150
Weight	365g
Volume discount	• 25+ units: 5% discount • 50+ units: 10% discount • 100+ units: 15% discount • 250+ units: 20% discount

The device ships from 6 different warehouses around the world. Below is a list of warehouses with geographical location and current stock.

Warehouses and stock

ScreenCloud

Name	Coordinates (latitude/longitude)	Stock
Los Angeles	33.9425, -118.408056	355
New York	40.639722, -73.778889	578
São Paulo	-23.435556, -46.473056	265
Paris	49.009722, 2.547778	694
Warsaw	52.165833, 20.967222	245
Hong Kong	22.308889, 113.914444	419

The cost of shipping an order is calculated based on the geographical distance between the warehouse and the shipping address, and the weight of the shipment. The rate is **$0.01 per kilogram per kilometer.**

A single order can be shipped from multiple warehouses. For example, if one warehouse does not have enough stock to fulfill the order, remaining units can be shipped from other warehouses. However, we always want the cost to be as low as possible. If shipping cost exceeds 15% of the order amount after discount, the order is considered invalid and should not be processed.

Your task is to design and implement the backend of the system.

Functional requirements

- A sales rep should be able to verify a potential order without submitting it, by inputting the number of devices and coordinates (latitude/longitude) of the shipping address. The rep should be able to see total price, discount and shipping cost, as well as an indication of the order's validity.
- A sales rep should be able to submit an order by inputting the number of devices and coordinates (latitude/longitude) of the shipping address. A successful order submission should result in warehouse inventory immediately being updated. The order must have an order number and store total price, discount and shipping cost as calculated at the time of submission.

Technical requirements

- The solution should be implemented in Typescript, and store data in a database. In the follow-up interview, we'll discuss your choices and thought processes behind your solution.
- The solution should expose its capabilities through a well-documented API.
- Extensive test coverage is not required, but a clear testing strategy should be demonstrated.

- You are free to use any libraries or frameworks. However, we recommend avoiding opinionated application frameworks like NestJS, as our goal is to evaluate your approach to designing the architecture and organizing the codebase.
- Approach the solution like you would a production system, and consider aspects such as performance, scalability, consistency and extensibility.
- It should be trivial to start the application and run tests in a local environment.
- Deploying the application to a cloud provider and demonstrating the use of a CD/CI pipeline is a plus.

There are two options for submitting the challenge:

1. Push the code to a public Github repo and email Imogen (imogen.king@screencloud.io) the link.
2. If you prefer the submission to be private, zip the project dir, upload it to https://wetransfer.com/ and email Imogen (imogen.king@screencloud.io) the link.

Let us know if you have any questions. And happy coding!