Vanilla Steel

Full-Stack Developer - Coding Challenge

Instructions & Timeframe

As the next step in the interview process, we'd like you to complete a coding challenge. We would like the take home challenge to be completed within 3-5 days. If you need more time, please reach out to us. Once we receive the result, we will schedule the next step of the interview process.
There will be two evaluations:

- Product evaluation - This will assess whether the application works as specified. Bonus points for creativity, scrappiness and user delight.
- Coding evaluation - This will assess the code quality and cloud setup.

The Project

You will be building a basic version of a Material Inventory Dashboard and Matching Function. There are two csv files provided, inventory and preference. **Inventory** contains a list of live material from different suppliers working with Vanilla Steel and includes information that is relevant for a purchase decision. **Preference** is a list of buyer preferences with different detail levels and reflects the specific purchase needs for a buyer. You will need to take the inventory data and persist it in a backend database. You will then develop a backend API, to be called upon by your frontend, which will display the data. A second page should offer functionality to upload the preference CSV and show exact matches of preferences with inventory data. The preferences data should also be persisted in your database with an upload date.

The result should be a dashboard with two pages (the application can be single-page or multiple pages):

- A main page that displays inventory information in a clean, user-friendly format
- A second page that allows upload of preferences and shows the matches

Because this position requires familiarity with web components, we would like you to use the Google Material 3 design system (https://m3.material.io/) for this project.

Vanilla Steel

Requirements

Backend

- The backend of this project can be done in Ruby or Python. You are permitted to use generators such as Ruby on Rails, Flask, etc.
- Inventory and preference is persisted in a database.
- Each data point should have a corresponding column in the database.
- You may use any database tool of your choice, such as PostgresQL, MongoDB, SQLite, etc.
- API provides frontend with inventory data and matching of preferences data.

Frontend

- The frontend of this project can be done in React, Vue, or Angular. You are permitted to use any of these frameworks CLI generators to quickly start a project
- The application utilizes Google Material 3 UI design system
- Main Page/Dashboard Requirements
    - The dashboard page cleanly displays the data from inventory data in a table with a summary pane
    - The following should be displayed in the Inventory pane:
        - Display the total number of Line Items.
        - Display the total volume in tons of the inventory.
    - For each Line Item, display:
        - Product Number
        - Form & Choice (concatenated, separated by a space)
        - Grade & Surface (concatenated, separated by a space)
        - Finish
        - Dimensions (mm)
        - Quantity
        - Total Weight (tons)
        - Location
    - For Dimensions, use dimension labels and concatenate the content: L = Length, W = Width, H = Height, T = Thickness, OD = Outer Diameter, Wt = Wall Thickness, Tw = Web Thickness, Tf = Flange Thickness
    - Always sort by weight (highest to lowest)
    - Allow additional sorting on Form & Choice.
- For the preference matching page
    - Apply Material, Form, Choice, Grade and Dimensions

- Match only items with weight > 10 tons

How to submit this challenge

- Download the repository here: https://tinyurl.com/bdef4b47
- Work on your solution.
- Deploy the frontend and backend of your application using free services (Netlify, Heroku, etc).
- Provide access to Balaji and Cliff to a demo-ready version of your solution.

Additional Information

- You are free to use any third-party libraries.
- Have fun and be as creative as you like!
- Please feel free to reach out to ask any questions (balaji@vanillasteel.com).