aiio Technical Task

This technical task has 2 parts that tests candidate's knowledge on React and Django

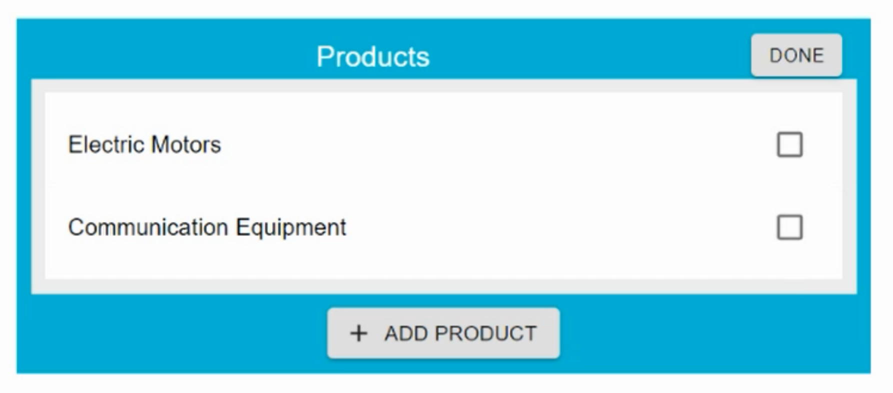
1. React
2. Django

Candidates applying for React engineer positions have to finish React part of this task and finishing Django part will be a bonus. But for Full Stack Developers Django part of the task is a must. In the attachments of the mail you will find the required three json data files needed to finish the task. One has the data of products, a file has the data of sub products, and the other file has data of sub categories of products.

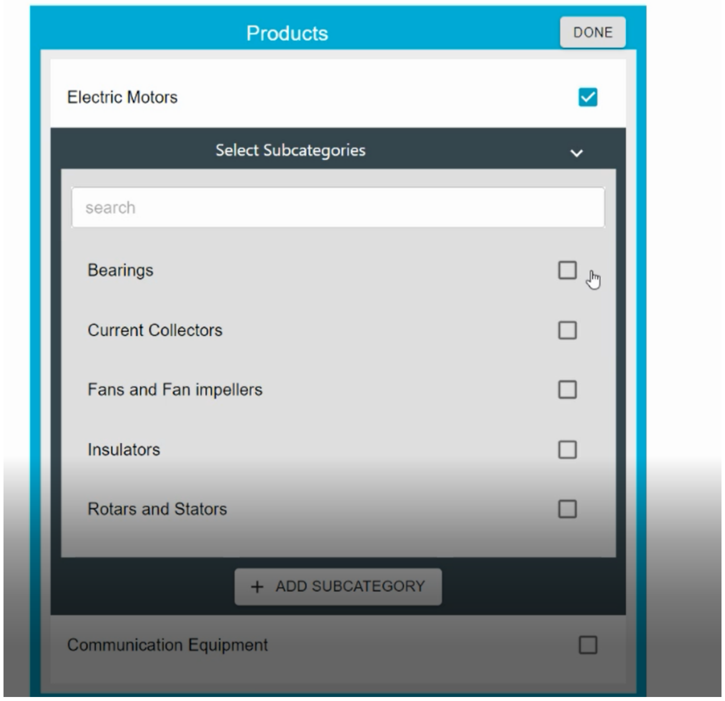
1. **React:**

   As mentioned above you have to understand the data models and build the following functionalities in react.

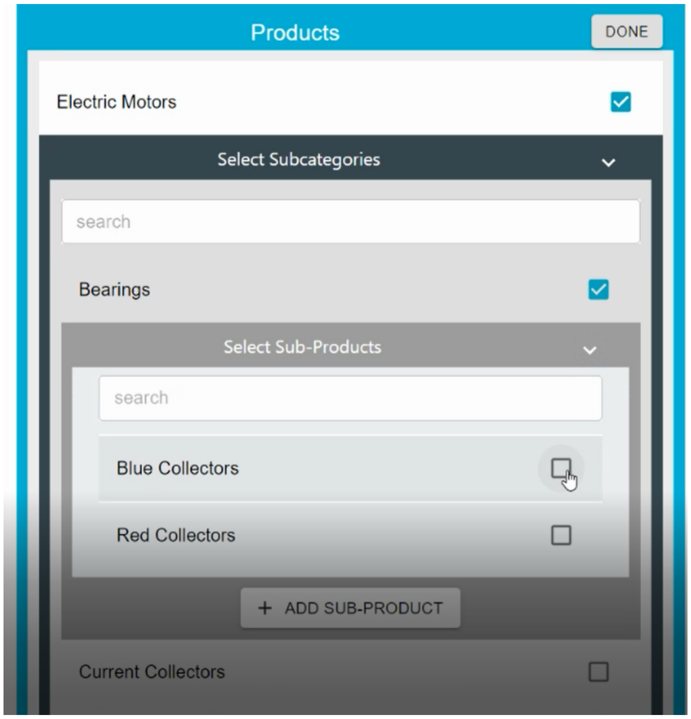
   1. Products being displayed as a list and expandable when checked to show subcategories as mentioned in the second image.



   2. Display the subcategories of the products with a search functionality of the subcategories and expandable when checked to show subproducts.



3. Display the subproducts with a search functionality when selected subcategories.



Products DONE

Electric Motors ☑

Select Subcategories

search

Bearings ☑

Select Sub-Products

search

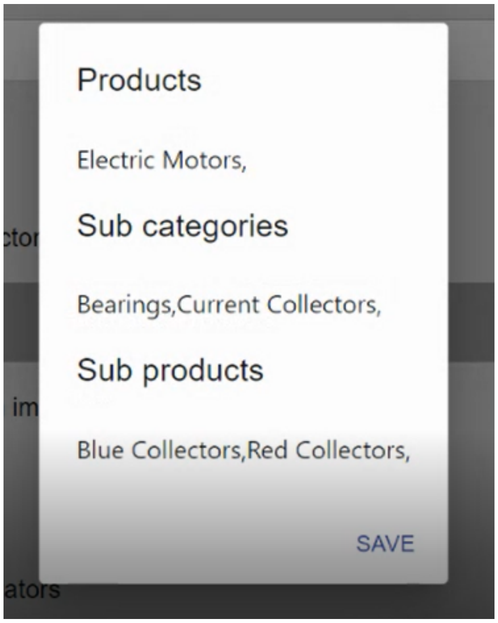
Blue Collectors □

Red Collectors □

+ ADD SUB-PRODUCT

Current Collectors □

4. Display the selected products, subcategories and sub products as below when clicked on "DONE".



Products

Electric Motors,

Sub categories

Bearings,Current Collectors,

Sub products

Blue Collectors,Red Collectors,

SAVE

Acceptance Criteria:

- Finishing the above functionalities. (Only React candidates are not needed to finish saving functionality and adding new products).
- Neat project structure.
- Proper State management. (Bonus points for using Context API or Redux for state management).
- Bonus points for loading the data using fetch or axios instead of using the static data from json.

2. Django:

Create a Django project, understand the data files and create models accordingly.

Acceptance Criteria:

1. Create API endpoints to fetch products, subcategories and sub products needed to show in the frontend part.
2. Create an API endpoint to add a new sub product.(Not needed for products and subcategories)
3. Create an endpoint to save the selected as shown in the save modal in the front end screen above. This can be a separate table in the backend.**(Optional and Bonus)**