Technical Assessment | Bosta

Product catalog service

Objective: Design a product catalog micro-service that will be used in an e-commerce platform, responsible for storing and searching in the catalog

To be delivered

- Write a complete system design document for the service
    - The document should include the following as well:
        - Database design
        - Infrastructure
- Implement only the **search REST API** that searches in the product catalog and ranks the results based on the best-selling products
    - Search should support any product attribute search, along with free text search

---

Requirements

- Product catalog should support product variants
    - T-Shirt is a product
    - T-Shirt (Red, Small, V-Neck) is a product and a variant to that product as well
    - T-Shirt (Green, Medium, Rounded-Neck) is a product and a variant to that product as well
    - T-Shirt (Green, Small, V-Neck) is a product and a variant to that product as well
- Products can be of any category, (example: Fashion, Electronics, Video games or Groceries)
- Products are linked to suppliers, each supplier can add their own product

System limitations

- 50 million products listed
- 10 million monthly active users
- Average user activity is 10 search per month

Notes

Please mention any assumptions made while designing the solution

Questions (*Answer without designing the solution, only mention what you think*)

- What would you change in the solution if the ranking will be based on most viewed items or the most/best reviewed?
- How would you think about linking relevant products together?
    - Supplier A has a Red T-shirt V-Neck
    - Supplier B has a Red T-shirt V-Neck & Green T-Shirt
    - How would the system know they are relevant to be recommended?
- How would you think about securing such service?
- What would be the type of servers needed for such like service? would it be RAM or CPU optimized types of machines and why?

Technical Requirements

- Programming: the task should be implemented using NodeJs
- Database: use any database system (SQL, NoSQL or both)
- Error Handling: the system should gracefully handle errors and provide meaningful feedback

Submission

- Codebase: share the code repository link
- Database: provide a schema diagram and any necessary setup scripts
- Solution should be dockerized and runnable with .readme file to set up and run the application
- Answers to the questions should be included in the design document