

Interview Problem

Intuit supports below products for Small Business owners:

1. Quickbooks (QB) for Accounting
2. QB Payroll for paying small business employees
3. QB Payments for small business to receive payments from their customers
4. TSheets for time-tracking

Customers can subscribe to any/all of the above products and by doing so, maintain a shared/common business profile across all of the products.

Example: Customer Ganesh runs a small flower shop and he has 4 employees in his shop. Ganesh has subscribed to all of the above products. Ganesh has a common business profile among all of these products.

A business profile is composed of below fields:

Field Name
Company Name
Legal Name
Business Address (Line1, Line 2, City, State, Zip, Country)
Legal Address
Tax Identifiers (PAN, EIN)
Email
Website

Customer can update their business profile from any of the above product workflows. Design and build a Business Profile service where every profile Create/Update request needs to be approved by all the subscribed products of the user.

In our use-case, for example, when Ganesh updates his company address via API, all of the above products (#1, #2, #3, #4) have to validate the data before data can be saved in the database.

Assume each of the above products has exposed a REST API that can be invoked to validate the data. For the purpose of implementation, you can use a dummy Validate API and use the same API for mocking each of the products.

Considerations

1. New products get added every 3 months once
2. The load on this to-be-built system will be high (1000 tps)
3. Response time of CRUD operation should be < 100m

SE2

1. Build a spring boot microservice with REST/GraphQL API's for CRUD operation
 - a. Have Well-defined API contract
 - b. Have well-defined HTTP status code and error messages.
2. Use HTTP as the method of interacting between all products
 - . Ensure the validations are done parallelly and with retries if required.
 - a. When any of the interaction fails, the API response will be a failure.
3. Use Dynamo or equivalent database for persistence
4. As a surprise, you can also build UI experience in React to update the profile fields.

Aim for:

1. Use the Spring framework as much as possible
2. Implement Hystrix for HTTP interactions
3. Apply design patterns as much as possible
4. Try following TDD or have good coverage for Unit Tests.