QikServe Engineer Technical Test

ⓘ Link to wiremock: https://cdn-dev.preoday.com/test/mock-api.zip

Feel free to spend as much time on as you need on the challenge as long as the following requirements are met!

Submission Guidelines

Please submit a git repository link containing:

- The solution you engineered
- A markdown file with answers to the Follow Up questions.
- Any documents, drawings, diagrams etc that can help us understand your thought process.

Requirements (MoSCoW)

The deliverables required following the MoSCOW prioritisation method. Interpret that as you see fit.

- MUST Have:
  - A java API that consumes the given wiremock and renders the content usable, dealing with the promotions as requested
  - Automated testing
- SHOULD:
  - A link to that java api hosted - if this is provided we will run tests against that
  - Internationalization
- COULD:
  - Multi tenancy of the wiremock - (The wiremock represents a POS that holds that data, we could have different wiremocks with different formats mapping to the same api)
- WONT HAVE:
  - A graphic interface that consumes that api - (Unless you're applying with a focus on a front-end or really want to impress, don't worry about building HTML).

Coding Task

Your task is to build the API for a checkout system for a local supermarket. Think about how the solution would be used to calculate the total cost of a basket which could contain any combination of items and promotions. Please bear in mind the following:

- Items can be added in any order
- Promotions should be applied whenever applicable.
- The store owner wants to display the customer's savings.
- Additional prices or promotion types might be introduced later.
- Use the Wiremock server with predefined API mappings. Utilize this API to fetch live data for your application.
- Please note that prices are denoted in pennies

Follow up questions

1. How long did you spend on the test? What would you add if you had more time?
2. What was the most useful feature that was added to the latest version of your chosen language? Please include a snippet of code that shows how you've used it.

3. What did you find most difficult?
4. What mechanism did you put in place to track down issues in production on this code? If you didn’t put anything, write down what you could do.
5. Explain your interpretation of the list of requirements and what was delivered or not delivered and why.