Take-Home Interview Question

Software Quality Automation Engineer

Objective

Demonstrate your ability to write an automated test using Node.js and Cucumber for a given Gherkin feature. If you are not comfortable with Node.js and Cucumber, you may use a programming language and testing framework you are more familiar with, provided it supports behavior-driven development (BDD) principles.

Instructions

1. Scenario
   - Write an automated test for the following Gherkin feature:

```
Feature: Cucumber Mathematics

  Scenario: Let's eat cucumbers!
    Given I have 5 cucumbers
    When I eat 4 cucumbers
    Then I have 1 cucumber

  Scenario: Let's eat carrots!
    Given I have 10 carrots
    When I eat 12 carrots
    Then...?

  Scenario: Let's make a salad!
    Given I have 8 cucumbers
    And I have 5 carrots
    When I make a salad with 3 cucumbers and 2 carrots
    Then I have 5 cucumbers
    And I have 3 carrots
    And I have 1 salad
```

2. Requirements
   - Create a new project using Node.js and include Cucumber.js as your test framework.
   - Use any additional libraries or tools you are comfortable with.
   - Implement the step definitions for the given Gherkin scenarios.
   - Complete the Gherkin scenarios where appropriate.
   - Ensure your project includes clear setup instructions (e.g., README.md) and can be run with a single command, such as *npm test* or *npx cucumber-js*.
   - Make this your own. Feel free to customize the scenarios to better suit your approach, as long as the core requirements being tested remain unchanged.
3. Deliverables
   - A GitHub repository (or a ZIP file) containing:

-
  - The complete project, including package.json and any necessary configuration files.
  - The README.md file with instructions on how to set up and run your test.
- Ensure that your code is clean, readable, and well-commented.

4. Evaluation Criteria
- Completeness: Does the test cover all steps of the scenarios?
- Code Quality: Is the code organized, modular, and readable?
- Functionality: Does the test run successfully and verify the scenarios' expected behavior?
- Documentation: Are the setup and usage instructions clear?

Feel free to include any questions, assumptions or notes in the README with your submission.