TECHNICAL TEST QA AUTOMATION

1. What do you know about Behavior-Driven Development (BDD)? Explain in detail what BDD is, its advantages, and how BDD testing works in practice.

Answer :

1. Behavior-Driven Development (BDD) is a software development methodology that enhances collaboration among developers, Quality Assurance (QA) professionals, and business stakeholders. It focuses on defining the expected behavior of software through user stories written in a language that everyone can understand, often using a format like Gherkin.
2. Advantages of BDD :
3. Improved Communication: BDD bridges the gap between technical and non-technical team members by using a common language
4. Clear Requirements: It ensures that development efforts are aligned with user needs and business objectives
5. Living Documentation: The executable specifications act as living documentation that can be easily updated
6. Enhanced Collaboration: Encourages collaboration and cross-functional teamwork, reducing miscommunication
7. Focused Development: Helps in prioritizing business-critical features and ensures that all development work is directly tied to business objectives
8. Practice Example

A common practice in BDD is writing scenarios using the “Given-When-Then” format. Here’s a simple example for a login feature:

Feature: User Login

Scenario: Successful login with valid credentials

Given the user is on the login page

When the user enters a valid username and password

Then the user should be redirected to the dashboard

Scenario: Unsuccessful login with invalid credentials

Given the user is on the login page

When the user enters an invalid username or password

Then an error message should be displayed

In this example:

Given describes the initial context.

When specifies the action taken by the user.

Then outlines the expected outcome.

This format ensures that everyone involved in the project has a clear understanding of the requirements and expected behavior of the software.

The presence of gherkin cucumber syntax helps a QA's knowledge of test case scenarios, resulting in clear and understandable debug code.

2. Create BDD Test Scenarios Using Gherkin Syntax for "Clio" Application

• Objective: Use Gherkin syntax to create test scenarios for the application "Clio." These scenarios will serve both as test documentation and for automating the given functionalities.

Website : <https://www.clio.com/>

• Instructions:

○ Register for a free trial of the "Clio" app to perform these tasks.

○ If you find any bugs during testing, write a detailed bug report.

Done use [syauqi247@gmail.com](mailto:syauqi247@gmail.com) account

• Scenarios to Automate:

2a. Create a New Company:

○ Develop a test scenario using Gherkin syntax to automate the process of creating a new

company in Clio.

2b. Create a New Matter:

○ Write a Gherkin scenario that automates the steps to create a new matter in Clio.

2c. Create a New Task:

○ Use Gherkin syntax to automate the creation of a new task in Clio.

2d. Record Time Entry to a Matter:

○ Automate a scenario to record time for a matter in Clio using Gherkin syntax.

- All 2a-2d on “clio project” folder or on github <https://github.com/syauqijufri>

• Additional Notes:

○ Ensure that each scenario is detailed and includes clear steps (Given-When-Then). *On Feature folder*

○ Highlight any bugs found with steps to reproduce, expected vs. actual results, and any other relevant details.

-Website not get a bug.