Project Proposal: Testing AutomationExercise.com (Manual, Automation, and API)

1. Project Description

AutomationExercise.com is an e-commerce website that provides a test environment for quality assurance engineers. This project aims to conduct manual testing, automated UI testing, and API testing on the platform to ensure its functionality, usability, and performance.

2. Group Members & Roles

- Manual Testing Lead: [Abdo Mohamed] - Responsible for executing test cases manually and reporting bugs.
- Automation Testing Engineer: [Abdelrahman Ahmed – Amar Ashraf] - Develops Selenium test scripts and maintains automation frameworks.
- API Tester: [Toni Ihab - Youssef Yahia] - Conducts API testing using Postman and Rest Assured.
- Documentation & Reporting: [Amar Ashraf] - Manages test case documentation and generates reports.

3. Team Leader

- Team Leader: [Abdo Mohamed] - Oversees testing activities, ensures quality standards, and coordinates between members.

4. Objectives

- Perform manual testing to identify UI/UX issues and validate business logic.
- Develop automated test scripts for functional and regression testing.
- Test the website's API endpoints to ensure data integrity and proper responses.
- Document test cases, test scripts, and results for reporting.

5. Scope of Testing

5.1 Manual Testing

- Functional Testing: Verify core functionalities such as user registration, login, product search, cart operations, and checkout.
- UI/UX Testing: Evaluate navigation, responsiveness, and accessibility.
- Cross-Browser Testing: Test compatibility across multiple browsers (Chrome, Firefox, Edge).
- Bug Reporting: Log defects and suggest improvements.

5.2 Automation Testing

- Use Selenium with Python/Java for UI automation.
- Automate key workflows like login, product search, add-to-cart, and checkout.
- Implement test data-driven approach for reusable test scripts.
- Integrate with TestNG/JUnit for test execution and reporting.

5.3 **API Testing**

- Test available API endpoints using Postman or RestAssured.
- Validate response status codes, headers, and payloads.
- Perform positive, negative, and edge-case testing.
- Automate API tests using RestAssured in Java or Requests in Python.

6. **Tools & Technologies**

- Manual Testing Excel for documentation, JIRA for bug tracking.
- Automation Testing: Selenium WebDriver.
- API Testing: Postman.
- Performance Testing: JMeter (for load and stress testing API endpoints).

7. **Milestones & Deadlines**

Phase	Duration
Test Planning	1 week
Manual Testing	2 weeks
Automation Scripting	3 weeks
API Testing	2 weeks
Reporting & Closure	1 week

8. **Key Performance Indicators (KPIs)**

- Bug Detection Rate: Number of defects found per test execution cycle.
- Test Coverage Percentage: Percentage of test cases executed successfully.
- Automation Success Rate: Percentage of automated test cases that pass without manual intervention.
- Response Time for API Requests: Average response time for API calls.
- Regression Stability: Number of new issues introduced after updates.

9. **Deliverables**

- Test Plan outlining the testing approach.
- Test Cases & Test Scripts for manual and automated tests.
- Bug Reports with severity levels.
- Test Summary Report with findings and recommendations.

10. **Conclusion**

This project aims to provide a comprehensive evaluation of AutomationExercise.com through manual, automation, and API testing. The findings will help improve the website's reliability, user experience, and performance.