**API Project: Testing Trello API using Rest Assured**

**Project Title:** Automated API Testing of Trello API using Rest Assured

**Objective:** The objective of this project is to design, execute, and automate API tests for Trello using Rest Assured. You will create detailed test cases, set up the testing framework, and automate the testing process to ensure the functionality of Trello's API.

**Project Requirements:**

1. **Trello API Documentation Review:**
   * Review the Trello API documentation to understand the available endpoints, request methods, required parameters, and response structures.
   * Register for a Trello account and obtain an API key and token.
2. **Environment Setup:**
   * Set up a Java project using a build tool like Maven .
   * Add dependencies for Rest Assured, TestNG, and other necessary libraries in your pom.xml .
3. **Framework Setup:**
   * Create a setup test class to handle common configurations and setup.
   * Configure environment variables for API key, token, and base URL.
4. **Test Case Design:**
   * Design at least 10 test scenarios covering the following scenarios:
     + **Board Management:** Create, update, delete, and retrieve boards.
     + **List Management:** Create, update, delete, and retrieve lists on a board.
     + **Card Management:** Create, update, delete, and retrieve cards on a list.
     + **Negative Tests:** Test with invalid inputs and ensure proper error handling.
     + **Create integration flows**
5. **Request Creation and Validation:**
   * Use Rest Assured to create and send requests for each test case.
   * Validate responses using assertions to check status codes and data validation.
6. **Test Organization:**
   * Organize your tests into separate classes based on functionality (e.g., BoardTests, ListTests, CardTests).
7. **Automated Testing:**
   * Set up your testing framework to run tests automatically using a test runner like TestNG.
   * Configure the build tool to run tests as part of the build process.
8. **Reporting:**
   * Generate test reports using tools like Allure or ExtentReports.
   * Analyze the test results and identify any failed tests or performance issues.
   * Document the findings and suggest improvements.
9. **Project Submission:**
   * Prepare a project report summarizing your approach, test cases, and results.
   * Include presentation of your testing setup, test execution, and reports.
   * Submit the complete source code and the project report.
   * Implement parameterized tests to run with multiple sets of data.
   * Demonstrate the use of property files or environment variables for configuration.

**Deliverables:**

* Complete source code of the project.
* Project report (PDF or Word document).
* Test reports generated by the reporting tool.

**Evaluation Criteria:**

* Completeness and correctness of test cases.
* Effective use of Rest Assured and Java.
* Organization and structure of the testing framework.
* Quality of automated testing setup.
* Clarity and comprehensiveness of the project report.