**Product Backlog 📋**

1. **Epic: Project Setup and Environment Preparation**
   * User Story: As a team, we need to set up the project environment and understand the requirements to start working efficiently.
     + Tasks:
       - Research and study the project scope and objectives.
       - Prepare the development and testing environment (e.g., install tools like Java, Maven, Git).
       - Understand use case scenarios for the Swag Labs website.
2. **Epic: Manual Testing and Test Case Development**
   * User Story: As testers, we want to perform manual testing and document test cases to ensure the website functionality is thoroughly validated.
     + Tasks:
       - Conduct manual testing on the Swag Labs website.
       - Write detailed test cases for all features (e.g., login, products, cart).
       - Review and evaluate test cases for accuracy and coverage.
3. **Epic: API Testing with Postman**
   * User Story: As testers, we need to validate the API endpoints of the Swag Labs website using Postman to ensure backend functionality.
     + Tasks:
       - Set up Postman for API testing.
       - Identify and document all API endpoints to be tested.
       - Write assertions for each endpoint to verify responses (e.g., status codes, data).
4. **Epic: Automation Testing with Selenium**
   * User Story: As automation testers, we want to build an automation testing framework using Selenium to automate testing of the Swag Labs website.
     + Tasks:
       - Set up a Selenium project with Maven and TestNG.
       - Implement the Page Object Model (POM) design pattern for better code organization.
       - Write automation scripts for key features (e.g., login, products, cart, checkout).
       - Integrate Allure for test reporting.
5. **Epic: Team Collaboration and Progress Tracking**
   * User Story: As a team, we want to use collaboration tools to track progress and sync tasks effectively.
     + Tasks:
       - Set up Trello/Notion boards for task management.
       - Assign tasks to team members and track progress.
       - Conduct regular sync meetings to ensure alignment.

**Sprint Backlog 📅**

**Sprint 1 (Week 1): Project Setup and Initial Research**

* **Goal:** Prepare the project environment and understand requirements.
* **Tasks:**
  + Research and study the project scope and objectives for Swag Labs.
  + Install and configure necessary tools (Java 24, Git 2.45.1, Node.js 21.5.0, npm 10.8.2, Maven).
  + Identify and document use case scenarios (e.g., login, add to cart, checkout).
  + Set up Trello/Notion boards for task management and team collaboration.
  + Assign initial tasks to team members and conduct a kickoff sync meeting.

**Sprint 2 (Week 2): Manual Testing and Test Case Development**

* **Goal:** Perform manual testing and document test cases.
* **Tasks:**
  + Conduct manual testing on Swag Labs website features (e.g., login, products page, cart page).
  + Write test cases for each feature (e.g., verify product images, verify add to cart functionality).
  + Review test cases as a team to ensure they cover all scenarios.
  + Update Trello/Notion with progress and document any issues faced.
  + Sync with the team to discuss findings and plan for API testing.

**Sprint 3 (Week 3): API Testing with Postman**

* **Goal:** Validate API endpoints using Postman.
* **Tasks:**
  + Install and set up Postman for API testing.
  + Identify all API endpoints for Swag Labs (if applicable, or simulate with mock APIs).
  + Write assertions for each endpoint (e.g., check for 200 status, validate response data).
  + Test API responses and document results in Trello/Notion.
  + Conduct a sync meeting to review API testing progress and prepare for automation.

**Sprint 4 (Week 4): Automation Testing with Selenium**

* **Goal:** Build and execute the automation testing framework.
* **Tasks:**
  + Set up a Selenium project with Maven and TestNG (already done as per pom.xml).
  + Implement Page Object Model (POM) for pages like ProductsPage, CartPage, etc.
  + Write automation scripts for key features (e.g., ProductsTest.java, CartPageTest.java).
  + Integrate Allure for test reporting (configure testng.xml with Allure listener).
  + Run tests and generate Allure reports (mvn clean test, allure serve target/allure-results).
  + Update Trello/Notion with final progress and conduct a wrap-up sync meeting.