

**Digital Egypt Pioneers Initiative (DEPI)**

**Test Plan Document**  
**Automation Exercise Website**

**Prepared by**

- 1.Ahmed Fayed
- 2.Alaa Mohamed
- 3.Essam Ehab
- 4.Rawan Mohamed
- 5.Yousef Samy

2024/2025

# **1. Introduction**

## **1.1 Purpose**

The purpose of this test plan is to define the strategy, scope, and objectives for testing the Automation Exercise Website. The website serves as a practice platform for UI and API automation, covering key functionalities such as user authentication, product management, shopping cart operations, checkout, and order processing.

## **1.2 Project Overview**

The Automation Exercise Website allows testers to practice automation testing by providing features like user registration, product browsing, shopping cart management, checkout, and order handling. This test plan ensures that these functionalities are thoroughly tested to meet quality standards.

## **1.3 References**

- Automation Exercise Website: <https://www.automationexercise.com>
- Software Requirements Specification (SRS)

# **2. Test Coverage**

## **2.1 Features to be Tested**

The following features will be tested:

- User Authentication
  - 1- Login
  - 2- Signup
  - 3- Logout
  - 4- Delete Account)
- Product Management
  - 1- Browsing
  - 2- Searching

- 3- Filtering by Category/Brand
  - 4- Viewing Product Details
- Shopping Cart
  - 1- Adding/Removing Items
  - 2- Viewing Cart
- Checkout & Order Management
  - 1- Checkout Process
  - 2- Payment Processing,
  - 3- Order Confirmation
  - 4- Invoice Generation
- Subscription
  - 1- Newsletter Subscription
- Support
  - 1- Contact Form Submission
- API Endpoints
  - 1- Authentication
  - 2- Product Management
  - 3- Cart Operations
  - 4- Order Processing

## **2.2 Features Not to be Tested**

- Internal database structures
- non-user-facing backend processes

## **2.3 Testing Levels**

- Unit Testing (Testing individual components)
- Integration Testing (Ensuring different modules interact correctly)
- System Testing (Validating end-to-end workflows)
- User Acceptance Testing (UAT) (Ensuring compliance with requirements)

## 2.4 Test Environment

- Browsers: Chrome, Firefox, Edge
- Devices: Desktop, Mobile (Responsive Testing)
- APIs: RESTful endpoints

## 3. Roles and Responsibilities

The testing team consists of 5 members, each assigned specific roles:

| <i><b>Role</b></i>                  | <i><b>Responsibilities</b></i>   | <i><b>MemberS</b></i>                     |
|-------------------------------------|--|---|
| <i><b>Test Coordinator</b></i>      | Ensures testing progress, assigns tasks, tracks defects, and reports status. | Alaa                                      |
| <i><b>Manual Test Owner</b></i>     | Creates test cases for UI, and functional. Logs defect                       | Ahmed<br>Essam<br>Rawan<br>Yousef         |
|                                     | Execute test cases and log defect  | Alaa<br>Essam<br>Rawan<br>Yousef          |
| <i><b>API Test Owner</b></i>        | Writes API Test Cases  | Ahmed<br>Alaa                             |
|                                     | Execute test cases to validate API requests and responses                    | Essam<br>Rawan<br>Yousef                  |
| <i><b>Automation Test Owner</b></i> | Develops and maintains automation scripts for UI and API testing             | Ahmed<br>Alaa<br>Essam<br>Rawan<br>Yousef |
| <i><b>Test Report Owner</b></i>     | Consolidates test execution results, prepares reports                        | Ahmed<br>Alaa<br>Essam<br>Rawan<br>Yousef |

## 4. Risks and Mitigation

### 4.1. Technical Risks

| Risk                         | Mitigation Strategy  |
|------------------------------|--|
| Browser Compatibility Issues | Conduct cross-browser testing and document inconsistencies.      |
| API Response Inconsistencies | Use predefined sample API responses for consistent testing.      |
| Automation Challenges        | Use stable locators and avoid hard-coded delays in test scripts. |
| Test Data Limitations        | Use mock data or generate synthetic test data.                   |

### 4.2. Resource & Team Risks

| Risk                            | Mitigation Strategy  |
|---------------------------------|--|
| Limited Access to Developers    | Focus on writing detailed defect reports with expected vs. actual results. |
| Knowledge Gaps in Testing Tools | Conduct knowledge-sharing sessions within the team.                        |
| Time Constraints                | Prioritize critical test cases and use automation for repetitive tests.    |

### 4.3. Documentation & Process Risks

| Risk                                | Mitigation Strategy  |
|-------------------------------------|--|
| Test Case Ambiguity                 | Follow structured test case templates with clear steps and expected results. |
| Defect Reporting Without Resolution | Focus on detailed defect analysis and categorize issues.                     |
| No UAT or Production Testing        | Perform exploratory testing to simulate different user behaviors.            |

### 4.4. Performance & Security Risks

| Risk                         | Mitigation Strategy  |
|------------------------------|--|
| Limited Performance Testing  | Use tools like Lighthouse to analyze page load times and browser performance.    |
| Security Testing Constraints | Focus on frontend security checks, such as form validation and URL manipulation. |

### 5. Success Metrics

Key Performance Indicators (KPIs) for testing:

- Test Execution Rate:** The percentage of test cases executed compared to the total planned test cases.
- Test Coverage Percentage:** The percentage of system functionalities covered by test cases.
- Automated Test Coverage:** The percentage of test cases that are automated versus manually executed.
- Defect Detection Rate:** The percentage of test cases that result in defect identification.
- API Response Time:** The time taken for an API request to receive a response from the server.
- Page Load Time:** The time taken for a webpage to load completely in the browser.
- Test Case Efficiency:** The number of defects detected per executed test case.

| Category               | KPI                      | Target  |
|------------------------|--------------------------|---------|
| Test Execution         | Test Execution Rate      | > 90%   |
|                        | Test Coverage Percentage | > 95%   |
|                        | Automated Test Coverage  | > 60%   |
| Defect Tracking        | Defect Detection Rate    | High    |
| Performance & Security | API Response Time        | < 500ms |
|                        | Page Load Time           | < 3 sec |
| Productivity           | Test Case Efficiency     | High    |

## **6. Test Deliverables.**

- Test Plan Document
- Software Requirements Specification
- Test Cases & Bug Report
- API Test Cases & API Testing Report.
- Automation Scripts & Test Execution Report.