SalesBricks Take Away Home Task Test Plan

1. Introduction

Objective: This test plan aims to validate the critical functionalities of the SalesBricks checkout flow. It ensures that users can configure orders, enter buyer details, and complete checkout reliably, enhancing the overall user experience.

Application Under Test (AUT): Salesbricks self-service checkout flow where buyers can:

- Configure orders by selecting add-ons and quantities.
- Enter buyer details, including personal and company information.
- Complete checkout via electronic payment using Stripe or through order form signature.

2. Scope/Test Items

Test Item ID	Feature	Description	Priority
TI-01	Order configuration functionalities	Validate selecting add-ons and quantities during order configuration.	High
TI-02	Buyer details input forms	Ensure that fields capture correct personal and company information.	High
TI-03	Payment processing	Test electronic payment via Stripe, ensuring all transactions are successful.	High
TI-04	Order form signature feature	Verify the functionality for completing orders through signature submission.	Medium

3. Risks and Assumptions

3.1. Risks:

- Payment gateway failures due to Stripe API downtime.
- Data loss during form submission caused by unstable network conditions.
- Incompatibility with specific browsers or devices.

3.2. Mitigation:

- Ensure test coverage includes Stripe's error-handling scenarios.
- Simulate network interruptions to test data resilience.
- Use BrowserStack for comprehensive cross-browser testing.
- Use anonymized test data for testing environments.

3.3. Assumptions:

- For the testing activities, it is assumed that the Stripe test environment is available and functioning properly. Additionally, it is assumed that all test accounts possess the necessary permissions to perform required actions.

4. Test Approach

Automated Testing:

- Focus on high-priority and repetitive test cases (e.g., payment processing and form validation).
- Utilize Playwright with TypeScript for reliable end-to-end tests.

Manual Testing:

- Exploratory and edge-case scenarios to identify unexpected behaviors.

4.1. Testing Levels

Testing Levels	Description
Smoke Testing	Verify that the main pages load and basic navigation works.
Functional Testing	Test each feature in detail.
Regression Testing	Re-run tests after changes to ensure existing functionality is intact.

4.2. Testing Types

Testing Types	Description
Automated Testing	Use Playwright with TypeScript for automating high-priority test cases.
Manual Testing	For exploratory testing and low-priority edge cases.

5. Test Environment

Test Environment	Description
Browsers	Latest versions of Chrome, Microsoft Edge, Firefox, and Safari
Devices	Desktop, tablets, and smartphones (if responsive design supported).
Testing Tools	BrowserStack or Sauce Labs for cross-browser compatibility.
Test Data	User realistic but fictitious data to simulate user input.

6. Test Schedule

Phase	Description	Estimated Duration
Phase 1	Create test cases and run e2e tests.	3 days
Phase 2	Develop and automate smoke test cases.	5 days
Phase 3	Execute regression tests as needed.	Ongoing

Note: Timelines may adjust depending on defect severity or environment readiness.

7. Deliverables

7.1. Test Plan Document

- Detailed testing strategy and scope.

7.2. Automated Test Scripts

- Playwright scripts stored in GitHub repository.

7.3. Test Execution Reports

- Summary of results with pass/fail status in CSV or HTML format.

7.4. Defect Reports (if any)

- Jira or other bug-tracking tool for defect management.

8. Responsibilities

Role	Responsibility
Test Engineer	Design, execute, and maintain automated and manual test cases.
Developer	Fix reported defects and provide necessary support for test environment setup.
QA Lead	Review test strategies and approve deliverables.

9. Exit Criteria:

- No Critical or High-Severity Defects Remain Unresolved
- All Planned Test Cases Executed
- Test Coverage Threshold Achieved (Automated tests achieve at least 80% code coverage on critical paths.)
- Stakeholder Approval Obtained
- Test Artifacts Delivered