DOKRAVE

ILLUSTRATIVE EXAMPLE | SYNTHETIC QA DEMONSTRATION

E-commerce to ERP Integration

Integration, data synchronisation and exception-handling testing

Scenarios	Passed	Failed	With workaround	Findings
6	4	1	1	1

Overview

A synthetic demonstration of validating data exchange between an e-commerce platform and an ERP system. It focuses on synchronisation direction, product and customer data, order import and handling missing product records.

Testing objective

Verify that supported e-commerce data is transferred to the ERP as specified, synchronisation occurs only in the intended direction, and exceptions are handled in accordance with integration requirements.

Scope

In scope

- Product synchronisation from the e-commerce platform to the ERP.
- Confirmation that ERP-side product changes do not trigger unsupported reverse updates.
- Order import for an existing customer and a newly created customer.
- Order handling when an order contains a product not yet available in the ERP.
- Retry behaviour after the missing-product condition is addressed.

Out of scope

- Testing all features of the e-commerce platform or ERP.
- Payment-provider, tax-calculation or shipping-carrier certification.
- Performance, volume or resilience testing beyond the stated scenarios.

Test approach

1. Confirm intended data-flow direction and expected mapping rules.
2. Create and update controlled test records in the source system.
3. Verify corresponding records and values in the ERP.
4. Check that ERP actions do not create unintended reverse synchronisation.
5. Create orders for existing and new customers, then verify imported order details.
6. Introduce a missing-product condition and compare observed behaviour with the agreed exception rule.

Test scenarios and results

ID	Scenario	Steps and expected result	Result
INT-01	New product synchronisation	**Steps:** Create two products in the e-commerce platform. **Expected:** Corresponding products are created in the ERP with the expected mapped information.	**PASS** Pass in this illustrative scenario.
INT-02	No reverse product synchronisation	**Steps:** Create a counterpart product in the ERP, edit it and delete it, then check the e-commerce platform. **Expected:** ERP-side changes and deletions do not trigger an update in the e-commerce platform.	**PASS** Pass in this illustrative scenario.
INT-03	Existing customer order	**Steps:** Place an order using a customer already present in the ERP. **Expected:** The order is imported and associated with the correct customer.	**PASS** Pass in this illustrative scenario.
INT-04	New customer order	**Steps:** Place an order using a customer not yet present in the ERP. **Expected:** The customer is created or matched according to integration rules, and the order is imported correctly.	**PASS** Pass in this illustrative scenario.
INT-05	Order with missing product	**Steps:** Place an order containing a product that does not exist in the ERP. **Expected:** The integration follows the specified exception rule, such as creating a designated placeholder product while retaining the original product description.	**FAIL** Failed. The expected placeholder was not created and the order remained pending.
INT-06	Retry after prerequisite is addressed	**Steps:** After the missing-product condition is addressed, retry the pending order. **Expected:** The order is imported without duplication and retains correct customer, product and order information.	**PASS WITH WORKAROUND** Pass after manually creating the required placeholder. This is a workaround, not evidence that automatic exception handling was fixed.

Findings

INT-F01 Missing-product exception not handled as specified

Severity / impact: Medium to high, depending on business context

When an order contained a product absent from the ERP, the expected placeholder-product handling did not occur and the order remained pending.

Retest / follow-up: A placeholder was created manually and the order was retried successfully. This showed a workaround and recovery path; it did not verify a permanent correction. Recommendation: create the placeholder product, or confirm the exception rule is implemented, before the integration goes live.

Outcome

This demonstration covers integration behaviour across systems, including data-flow direction, customer handling, order import and exception recovery. It shows why testing must verify not only successful synchronisation but also behaviour when required data is missing.

Tools and techniques

Illustrative tools: e-commerce test environment, ERP test environment, integration logs and application-level data verification.

Important note. Synthetic demonstration. Platform and system references are generic. The missing-product behaviour and manual workaround are illustrative, not a claim about a named customer or production integration.