Test Plan – SOAP ISBN10 Validation API

Created by Srilatha Kore – Test Plan (SOAP Request)
TheTestingAcademy.com

Objective

The objective of this SOAP API test plan is to validate the ISBN10 number using the IsValidISBN10 API. The service accepts a 10-digit ISBN number via XML POST request and returns a boolean response—true if the ISBN is valid, otherwise false.

Scope

Scope of Test Plan for the ISBN10 Validation SOAP API:

1. **Functional Testing**
   - Validate correctness of API response for different types of ISBN numbers.
   - Ensure the response adheres to expected XML schema and data types.
2. **Data Validation Testing**
   - Validate the ISBN input constraints and formats.
   - Test correct and incorrect ISBN values (e.g., length, invalid characters).
3. **Error Handling Testing**
   - Validate error messages for malformed or missing parameters.
   - Ensure API responds with correct HTTP status codes.
4. **Security Testing**
   - Verify secure access if authentication is needed.

   - Check for input-based vulnerabilities.

5. **Integration Testing**
   - Test end-to-end flow where this API integrates with other systems.
6. **Compatibility Testing**
   - Validate API calls on different platforms/tools (Postman, SOAP UI, etc.).
7. **Documentation Review**
   - Ensure API documentation aligns with actual behavior.

Inclusions

- Send SOAP POST requests to the API endpoint using different test ISBN values.
- Validate API's ability to differentiate valid and invalid ISBN numbers.
- Include edge case, boundary, and malformed input testing.

Examples Covered:

- Valid ISBN (e.g., 0439023483)
- ISBN ending in X (e.g., 0306406152)
- Invalid ISBNs (e.g., short, long, alphanumeric, blank)
- Boundary & special character testing

Test Environments

Component	Description
API URL	http://webservices.daehosting.com/services/isbnservice.wso
OS	Windows 10, macOS

Browsers	Chrome, Firefox, Edge, Safari
Devices	Desktop, Mobile (for SOAP Tools)
Tools Used	SOAP UI, Postman, XML Editor
Protocol	SOAP 1.1
Response Format	XML

Defect Reporting Procedure

- **Tool**: JIRA
- **Template**: Include title, steps to reproduce, expected vs actual, screenshots, logs
- **POCs**:
  - Frontend: N/A
  - Backend: Web Service Dev
  - QA Lead: Srilatha Kore
- Severity & Priority defined during triage meetings

Test Strategy

1. **Test Case Design**
   - Use Equivalence Partitioning & Boundary Value Analysis.
   - Include both positive and negative test scenarios.
2. **Execution Flow**
   - Smoke test API endpoint for availability.
   - Proceed with functional, negative, and exploratory tests.

3. **Types of Testing**
    - Functional, Regression, Usability (Developer Ease), Exploratory
4. **Best Practices**
    - Context-Driven & Shift-Left Testing
    - End-to-End Flow Validation with Sample Requests

Test Schedule

Task	Duration
Test Plan Creation	Day 1
Test Case Preparation	Day 2
Execution & Defects	Day 3 – Day 4
Summary Report	Day 5
Sprint Duration	**1 Week Total**

Test Deliverables

- Test Plan Document
- Test Case Document (Spreadsheet or Tool-Based)
- Execution Logs & Screenshots
- Defect Reports
- Final Summary Report

Entry and Exit Criteria

Requirement Analysis

- **Entry**: Access to API WSDL & basic endpoint details
- **Exit**: All requirements understood and documented

Test Execution

- **Entry**: Test Cases Approved, API Reachable
- **Exit**: Test Logs & Defect Reports submitted

Test Closure

- **Entry**: Final Test Results Compiled
- **Exit**: Test Summary Report Shared

Tools

- JIRA (Bug Tracking)
- SOAP UI / Postman (Test Execution)
- Excel / Google Sheets (Test Case Documentation)
- Snipping Tool / Screenshot Tool

Risks and Mitigations

Risk	Mitigation
API Service Down	Use Mock Server or retry mechanism
Lack of Test Data	Generate synthetic ISBN data
Limited Access	Communicate with service owner early

Approvals

Team will send below documents for client or stakeholder approval:

- Test Plan
- Test Scenarios & Cases
- Execution Reports
- Final Summary

Testing will proceed to next phase only after approval.