<Date>

**Version <X.x>**

**NOTES TO THE AUTHOR/TEMPLATE INSTRUCTIONS**

This template includes instructions and boilerplate text for this document. The author should note that:

* Each section provides instructions or describes the intent, assumptions, and context for content included in that section. Instructional text appears in *blue italicized text* throughout this template.
* Replace instructional text in each section with project specific information.
* Use or modify boilerplate examples of wording and formats for text and tables as appropriate.
* Search and replace all text enclosed in angle brackets - < > - with project specific information (e.g., <Project Name> or <Project Acronym>.

Follow these steps when using this template:

1. All documents must be compliant with Section 508 requirements. Refer to [FNS 504-508 Compliance Reference Library](https://fncspro.usda.net/offices/oit/Documents/Forms/AllItems.aspx?RootFolder=%2Foffices%2Foit%2FDocuments%2F504%2D508%20Compliance%20%2D%20Accessibility&View=%7BB47848DF%2D6059%2D4D43%2DAB0C%2D8ECEDC92AD4C%7D) or [Section508.gov](https://section508.gov/) for more information.
2. Modify any boilerplate text, as appropriate, for your specific project.
3. Use Styles for new sections such as Heading 1, Heading 2 and Std Para.
4. Place Table captions and descriptions *above* the table and centered. All tables must have an associated tag providing appropriate alternative text for Section 508 compliance.
5. Place Figure captions and descriptions *below* the figure and centered. All figures must have an associated tag providing appropriate alternative text for Section 508 compliance.
6. Update the Table of Contents and any List of Tables or List of Figures by right-clicking it and selecting Update field / Update entire table.
7. Delete this “Notes to the Author/Template Instructions” page and all instructional text to the author before finalizing the draft of the document.

**SDLC TEMPLATE REVISION HISTORY**

| VERSION | DATE | CHANGE DESCRIPTION |
| --- | --- | --- |
| 1.0 | 03/13/2013 | Created the document. |
| 1.1 | 09/09/2013 | Re-formatted the document. |
| 1.2 | 03/23/2017 | Re-formatted the document. |
| 1.3 | 04/06/2017 | Updated the document to reflect changes |
| 1.4 | 11/02/2020 | Updated to reflect new FNS Agile SDLC processes and comply with Section 508 standards. |

**SDLC TEMPLATE CONTACT INFORMATION**

| RESPONSIBILILTY | CONTACT PERSON | EMAIL ADDRESS |
| --- | --- | --- |
| Portfolio Management Division Director, Chief Portfolio Officer | Joe Shaw | [Joseph.Shaw@usda.gov](mailto:Joseph.Shaw@usda.gov) |
| IT Governance Manager | Kevin Russ | [Kevin.Russ@usda.gov](mailto:Kevin.Russ@usda.gov) |
| SDLC Lead | Max Mounger | [Max.Mounger@usda.gov](mailto:Max.Mounger@usda.gov) |

Complete the below table to track the version history of this document.

**DOCUMENT REVISION HISTORY**

| **VERSION** | **DATE** | **AUTHOR** | **CHANGE DESCRIPTION** |
| --- | --- | --- | --- |
| x.x |  |  |  |
| x.x |  |  |  |
| x.x |  |  |  |

*List the relevant area of responsibility, contact person and email address for this document.*

**CONTACT INFORMATION**

| **RESPONSIBILILTY** | **CONTACT PERSON** | **EMAIL ADDRESS** |
| --- | --- | --- |
| Area of Responsibility |  |  |
| Area of Responsibility |  |  |
| Area of Responsibility |  |  |
| Area of Responsibility |  |  |
| Area of Responsibility |  |  |

Table of Contents

[1. Introduction 1](#_Toc55236834)

[1.1 Scope 1](#_Toc55236835)

[2. Assumptions and Constraints 1](#_Toc55236836)

[2.1 Assumptions 1](#_Toc55236837)

[2.2 Constraints 1](#_Toc55236838)

[3. Test Overview 1](#_Toc55236839)

[3.1 Test Levels 1](#_Toc55236840)

[3.1.1 Unit Testing 2](#_Toc55236841)

[3.1.2 Integration Testing 2](#_Toc55236842)

[3.1.3 Functional Testing 2](#_Toc55236843)

[3.1.4 Regression Testing 2](#_Toc55236844)

[3.1.5 System Testing 2](#_Toc55236845)

[3.1.6 Performance Testing 3](#_Toc55236846)

[3.1.7 Load Testing 3](#_Toc55236847)

[3.1.8 508 Testing 3](#_Toc55236848)

[3.1.9 User Acceptance Testing 4](#_Toc55236849)

[3.2 Organization and Locations 4](#_Toc55236850)

[3.3 Schedule 5](#_Toc55236851)

[4. Test Environment 5](#_Toc55236852)

[4.1 Equipment 5](#_Toc55236853)

[4.2 Software 6](#_Toc55236854)

[4.2.1 Software Resources 6](#_Toc55236855)

[4.2.2 Support Software 6](#_Toc55236856)

[4.3 Test Material 7](#_Toc55236857)

[4.4 Test Training 7](#_Toc55236858)

[5. Test Description 7](#_Toc55236859)

[5.1 Covered User Stories 8](#_Toc55236860)

[5.2 Test Case Description 8](#_Toc55236861)

[5.3 Test Data Handling 9](#_Toc55236862)

[5.4 Test Cases 9](#_Toc55236863)

[6. Appendix A: Test Cases (IF APPLICABLE) 10](#_Toc55236864)

[7. Appendix B: Test Scripts 11](#_Toc55236865)

[Attachment A – Test Analysis Report 12](#_Toc55236866)

**LIST OF TABLES**

[Table 1 - <Project Name> Testing 4](#_Toc54676724)

[Table 2 - <Project Name> Testing Schedule 5](#_Toc54676725)

[Table 3 - Equipment Environment 5](#_Toc54676726)

[Table 4 - Software Resources 6](#_Toc54676727)

[Table 5 - Covered User Stories 8](#_Toc54676728)

[Table 6 - Test Case Description 8](#_Toc54676729)

**LIST OF FIGURES**

No table of figures entries found.

*List the acronym reference and definition or description for each acronym contained in this document.*

**ACRONYM LIST**

| **REFERENCE** | **DEFINITION** |
| --- | --- |
| FNS | Food and Nutrition Service |
| ICT | Information and Communication Technology |
| IT | Information Technology |
| ITG | Information Technology Governance |
| OIT | Office of Information Technology |
| PM | Project Manager |
| TAR | Test Analysis Report |

# Introduction

Create this document for a new system and update annually at a minimum. A project may determine that a test plan or an abbreviated test plan including just the test execution and test case information is required for each release.

Develop or update test cases and test scripts for each release. Use the test case template (Excel spreadsheet) if appropriate.

Also describe the purpose of this the test plan.

This test plan describes the testing scope and activities and outlines the testing definitions and test execution activities for the <Project Name> application.

## Scope

Provide a description of the intended scope of the test plan.

# Assumptions and Constraints

This section describes the assumptions and constraints that may influence testing on this product.

## Assumptions

Describe any assumptions that may influence testing on this product.

This section describes any assumptions that may influence testing on this product.

## Constraints

Describe any constraints imposed on testing, whether they exist because of a lack of specialized test equipment, or limits on time or resources. Indicate what steps, if any, are necessary to reduce program risk of the test constraint.

This section describes constraints imposed on the testing and any steps necessary to reduce the program risk due to test constraints.

# Test Overview

This section describes the levels of formal testing that take place during the system development process: integration, functional, regression, system, load, performance, section 508 compliance, security, and user acceptance tests.

It also addresses the planning to execute this testing, including test schedules.

## Test Levels

Describe the types of testing performed on this product, who conducts the tests, the location of the test cases, and any tools used during the testing.

### Unit Testing

Describe how to conduct unit testing on this product.

Unit testing is the testing of individual components (units) of the software. The objective of unit testing is to test the functionality of the code, which implements and maps to the functional requirements/user stories identified in <insert document name>. Unit test cases use a set of test cases that focus on the control structure of the procedural design. Developers conduct unit testing in the <insert environment name>. Unit test cases are documented <insert additional information and location.>

### Integration Testing

Describe how to conduct integration testing on this product. Conduct integration testing after unit testing and focus on the interfacing between any interconnected modules. Test cases simulate interaction between two modules.

System integration is the process of testing the integrated and configuration managed system for integration with other major components and interfacing systems to make certain that they operate to specifications. It also ensures that appropriate interfaces execute as designed without adversely affecting other components. Focused attention is on the physical mechanics of the interfaces, such as data transport management.

<Insert who conducts integration testing> conducts integration testing in the <insert environment name>. Integration test cases and test scripts are located <insert additional information and location.>

### Functional Testing

Describe how to conduct functional testing on this product. Include a description of the test coverage and location of test cases and test scripts.

Functional system testing validates that the product is meeting business requirements. It includes validating via the user interface as well as reports that the data meets expectations and that it flows correctly through the system. The main focus is to verify that the fully integrated system works per requirements/user stories.

### Regression Testing

Describe how to conduct regression testing on this product. Include a description of the test coverage and location of test cases, test scripts, and data.

Regression testing is re-running functional and non-functional tests to ensure that previously developed and tested software still performs after an introduced change. Regression testing ensures that no other previously working functionality fails as a result of the fixes or newly added functionalities. It is a quality control measure which ensures that the new or modified code still complies with its specified requirements, and there are no affects to untouched code. Integration test cases and test scripts are located <insert additional information and location.>.

### System Testing

System testing, conducted after integration testing, includes both functional and non-functional testing. Describe how to conduct system testing on this product. Include a description of the test coverage and location of test cases and test scripts.

System testing is testing conducted on a complete integrated system to evaluate the system's compliance with its specified requirements/user stories. System testing covers the entire system in the context of functional and non-functional requirements or user stories. System testing tests not only the design, but also the behavior and expectations of the product owner and users.

### Performance Testing

Describe the approach for conducting performance testing on this product. In performance testing, the load on which the system is tested is normal.

Performance testing is a type of software testing which determines system performance in terms of sensitivity, reactivity, and stability under a particular workload. Performance testing is the process of determining the system’s performance that includes speed and reliability under varying load. In performance testing the load on which the system is tested is normal.

### Load Testing

Describe the approach for conducting load testing on this product. Identify how you determined peak load for the product. Only sustainability of the system is tested during load testing.

Load testing is a type of software testing which determines the performance of a system, software product or software application under real life-based load conditions. Load testing is the process of determining the behavior of a system when multiple users access it at the same time. It checks the behavior of the system under extreme load.

During load testing, peak load is used. Peak load for <Project Name> is <insert the determined peak load for the system>.

### 508 Testing

Describe the approach for conducting Section 508 testing on this product. Use the Accessibility Conformance Report contained within the Voluntary Product Accessibility Template to document your approach. Include references to tools used to validate conformance. 508 testing for FNS’ purposes is contained within functional testing, but called out in a separate section to highlight the level of importance. For more information refer to [Section 508 ICT Testing Baseline](https://section508coordinators.github.io/ICTTestingBaseline/).

The primary focus of 508 testing is to ensure <System Name> satisfies the baseline requirements of Revised 508 Standards. Testing covers all the guidelines identified for revised section 508 that has incorporated the Web Content Accessibility Guidelines (WCAG) conformance level A and AA.

The following are the Section 508 Information and Communication Technology (ICT) Testing Baselines. The ICT Testing Baseline is not a test process itself.

1. Conforming Alternate Version and Non-Interference
2. Auto-Playing and Auto-Updating Content
3. Flashing
4. Keyboard Access and Focus
5. Forms
6. Links and Buttons
7. Images
8. Adjustable Time Limits
9. Repetitive Content
10. Content Structure
11. Language
12. Page Titles, Frames, and iFrames
13. Sensory Characteristics and Contrast
14. Tables
15. CSS Content and Positioning
16. Pre-Recorded Audio-Only, Video-Only, and Animations
17. Synchronized Media
18. Resize Text
19. Multiple Ways
20. Parsing

### User Acceptance Testing

Conduct User Acceptance Testing after system testing to check whether the software meets the requirements/user stories based on product owner and user validation. Describe the approach for conducting user acceptance testing on this product. Include a description of the test coverage and location of test cases for this test. Place test cases in a centralized location, preferably in the OIT project space. Update test cases (test case repository) for each release to reflect changes. The requirements for conducting UAT and developing associated test cases are product-specific.

User Acceptance Testing (UAT) occurs at the end of the whole testing process when the product is ready to deliver. UAT validates that the system is acceptable or not by the Product Owner and fulfills system performance requirements.

## Organization and Locations

Describe the participating organizations and the location where the software testing occurs. Incorporate 508 testing into functional testing.

Table - <Project Name> Testing

| **Test Level** | **Responsible Group** | **Location** |
| --- | --- | --- |
| Test Level | Who is responsible for testing | In what location is the testing completed? |
| Unit |  |  |
| Integration |  |  |
| Functional |  |  |
| Regression |  |  |
| System |  |  |
| Performance |  |  |
| Load |  |  |
| 508 Testing |  |  |
| User Acceptance Testing |  |  |

## Schedule

Outline the high-level schedule for the testing levels. Do not include unit testing as this is on-going during development activities. If all identified information is in the project schedule, you can reference that or use the table below to provide the information.

Table - <Project Name> Testing Schedule

| **Testing Level** | **Planned Start Date** | **Planned Completion Date** |
| --- | --- | --- |
| Integration |  |  |
| Functional, including 508 Testing |  |  |
| Regression |  |  |
| System |  |  |
| Performance |  |  |
| Load |  |  |
| User Acceptance Testing |  |  |

# Test Environment

This section discusses the requirements for the test environment, if applicable.

## Equipment

Describe the hardware configurations of the independent test environment, if applicable.

The <Project Name> test environment consists of <high level description of test equipment>. The table below lists the minimum equipment requirements for testing.

Table - Equipment Environment

| **Component** | **Description** |
| --- | --- |
| XX Laptop | Processor:  Memory:  SDRAM  Mouse  Hard drive: |
| Peripheral device | Laser printer |
| Workstation #1 |  |
| Laptop #2 |  |

## Software

This section lists other software needed to support testing that is not part of the tested software. This should include debugging software and programming aids; any drivers or system software used in conjunction with the new software to ensure compatibility and integration; and any software required to operate the equipment and record test results.

### Software Resources

The table below, Software Resources, provides a detailed list of the software required for testing and gives a brief description of its use.

Table - Software Resources

| **Software** | **Description** |
| --- | --- |
| Chrome | Browser, version X.X or greater |
| Software #1 |  |
| Software #2 |  |

### Support Software

Describe the support software used. Identify the test tool and the description and test activity information. Use the table for organization.

The following table lists the software tools used for supporting the different types of testing.

Table - Test Tools

| **Test Tool** | **Description and Test Activity** |
| --- | --- |
| Name of Test Tool | Description of the test tool and the testing activities associated with it. |
| **JIRA** | JIRA captures a backlog of the release requirements as well as the Master Product Backlog containing high level requirements for the system. Customized reports and queries enable team members to run reports that show only the data they need to see. Test case information and design elements mapped into Jira generate the Product Traceability Matrix. Items that must be entered in the database are:   * Master Product Backlog * Release Product Backlog * Design elements mapped to user stories * Test cases mapped to user stories |
| Test Tool #2 |  |

## Test Material

In this section, lists the materials needed for the test, such as system documentation, user manuals, test procedures, and worksheets.

Use the following materials for testing:

* Operating software
* Data provided by the <agile team>
* Product Traceability Matrix
* Test Plan for <Project Name, Release X.x>
* Hardware, as specified in Section 4.1– Equipment
* Software, as specified in Section 4.2 – Software
* Test Cases, as specified in Section 5.4 – Test Cases

## Test Training

This section describes or references the plan for providing training in the use of the tested software. Specify the types of training, personnel targeted for training, and the training staff.

<Example – Test team personnel are expected to be familiar with the application prior to performing testing.

or

The agile team provides a release demonstration to the Product Owner and testing stakeholders prior to the commencement of UAT. The agile team presents a review of the applicable user stories, along with the associated test scripts, using information identified in Section 5.2 – Test Case Description.>.

# Test Description

In this section, describe the testing approach. Boilerplate language provided below.

In the subsections, describe the user stories to be tested and identify how the test cases relate or track back to the user stories. Provide the test cases if applicable for User Acceptance Testing. FNS recommends storing User Acceptance Testing test cases in a centralized location such as the project SharePoint location for ease of use by release or sprint as appropriate for your project. This also reduces the effort to copy and paste the actual test cases within the test plan, as you can provide the location instead.

The <insert test levels> for <Project Name, Release X.X> verifies that all input and output records, external and internal interfaces, as well as system processing is conducted in accordance with the user stories and acceptance criteria and testing techniques described in this section. Section 5.1 – Covered User Stories identifies user stories associated with <Project Name, Release X.X>.

Test case procedures are the primary method of verifying that the functional requirements documented in the Product Backlog, Product Traceability Matrix, users stories and change requests supported by the application. These test cases cover essentially all the requirements/user stories and change requests allocated to <Project Name, Release X.X> and are organized by <describe method, e.g., test case, view, functionality etc.>. <Add additional instructions as necessary>. The test cases contain individual steps and actual results compared with the expected results. <Insert tracking tool, Jira> and the test analysis report (TAR), as appropriate, documents any discrepancies and deviations. Output captured includes screen shots, system messages, reports, forms, and other printed output as described in the expected results section of each test case.

All record data is monitored and any type of problem that may occur during testing, whether a software or hardware problem, is noted. Generating defect or trouble tickets for all software or hardware problems occurs immediately and tickets are assigned a priority. Retesting or regression testing occurs after resolving the defect. Section 5.2 – Test Case Description lists test cases and their sources. Section 5.4 – Test Cases outlines test case information.

## Covered User Stories

In this section, list the requirements or user stories to be tested. You can extract this information from the tool you use to manage user stories (release backlog).

<Project Name, Release X.x> - User Stories lists and describes the requirements, user stories, or change requests to be tested.

Table - Covered User Stories

| **User Story ID** | **Description** |
| --- | --- |
| ABC-1023 | Provide Coordinators ability to delete their Reviews in the Search Reviews Page through “Delete This Review” button displayed in the “Details” section of the Review. |
| <User story #1> |  |
| <User story #2> |  |

## Test Case Description

In this section, describe each test performed and the guidelines and approach used in the development of the test cases. Include a description of each test case executed and traceability to the requirements/user story validated by the test cases. Note – the below table is a different layout than the Product Traceability Matrix. This table is focused by test cases to user stories, not user stories to test cases. There is usually a many-to-many relationship between user stories and test cases.

Table 6 – Test Case Description outlines each test case by identification number, provides a succinct description, and maps to the user stories associated with the test case.

Table - Test Case Description

| **Test Case ID** | **Test Case Description** | **User Story Traceability** |
| --- | --- | --- |
| TC-ABC-58 | TC\_Search Reviews Menu: FiscalYearButton | ABC-203 |
| TC #1 |  |  |
| TC #2 |  |  |

## Test Data Handling

Describe how test data is developed or generated and who is responsible. Describe how to decouple data from production and how to anonymize the data, if applicable. Provide additional information for areas such as multiple data sets, and repeatability of regression tests, as applicable for your project.

## Test Cases

Test cases include the step number, the steps or actions taken, the expected results, the actual results, whether the step passed or failed, the defect tracking number (e.g. Jira) and comments to provide any clarification. Reference the test case template for a recommended format if a test case management tool is not used.

FNS recommends a centralized library for test cases, such as the project specific SharePoint location, with accessibility by the Product Owner, IT project manager, and other project stakeholders. For organization and maintenance, store test cases and results in a release folder. Store sprint test cases and results in the same manner if applicable for your project.

Provide the location of the test cases in the centralized library or embed any files within this section or in an appendix, as applicable.

The test cases and test results for <Project Name, Release X.x> are located at <insert link>.

# Appendix A: Test Cases (IF APPLICABLE)

A test case is documentation which specifies input values, expected output, and the preconditions for executing the test. It is associated with manual testing.

List all the test cases for the system if the location is not provided earlier in the document or provide the appropriate link.

# Appendix B: Test Scripts

A test script in software testing is a set of instructions performed on the system to test that the system functions as expected. It is associated with automation testing.

Describe the test scripts for the system if applicable and their location.

Attachment A – Test Analysis Report

Insert the link to or embed the object for the finalized test analysis report (TAR) associated with this test plan. The finalized TAR summarizes the results of the testing.

<Insert final TAR link or object if applicable>