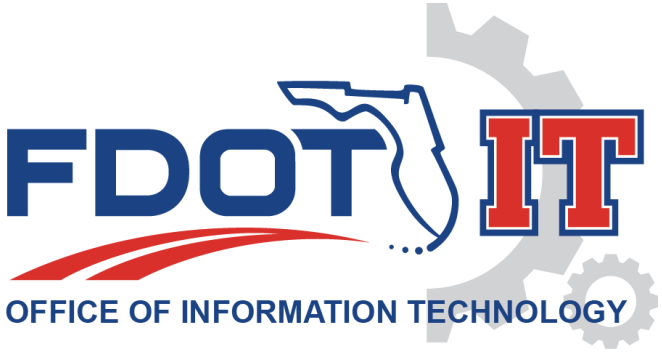

Florida Department of Transportation
Office of Information Technology (OIT)

Application Testing Standards

Table of Contents

Document Revision History

The following table lists the draft versions of this document. The 'Author(s)' column represents the person or people who made the updates to the draft or version; the 'Date' column indicates the date draft or version was completed; the 'Version'/Description' column provides a high-level description of and changes made to the draft or version

Version	Date	Name	Description
1.0	6/12/2017	Preeti Zutshi, Bill Lucas	Ready for TSRT review
1.1	7/26/2017	Stephanie Taylor	Corrected minor grammatical errors

About The Standards

Scope

The following standards apply to testing deliverables that are developed or maintained by vendors, staff, or consultants employed by, or contracted with, OIT.

Structure

Supporting details for each standard are included below. The supporting details provide standards that Applications Services staff have developed over the years. It also includes references, links, and techniques that can be used in conjunction with the referenced standards.

Standards

All projects that include testing deliverables are required to follow these standards. All projects that are subject to OIT oversight will be reviewed against these standards for compliance. All exceptions to these standards must be requested through the Technical Standards Review Team (TSRT) with final approval by the Enterprise Architect. Please see the OIT Method & Practice entitled TSRT Exception Request Process for detailed information regarding exception requests.

Application Testing Standards

1.0 General Standards

All projects must include the following unless otherwise stated in the procurement instrument or project management plan:

1.1 Test Plan

All Projects must include a test plan. See Section 2.0 for details.

1.2 Functional Test Cases/Test Scripts

All projects must include functional test cases/test scripts. See Section 3.0 for details.

1.3 Testing Types

All projects must perform certain specific types of testing. See Section 4.0 for details.

1.4 Defect Tracking

All projects must perform defect tracking. See Section 5.0 for details.

1.5 Traceability Matrix

All projects must include a Traceability Matrix that shows that Testing Requirements trace to all defined Functional and Non-Functional Application Requirements. See Section 6.0 for details.

2.0 Test Plan

The Test Plan must include the following sections:

- Purpose
- Testing Types
- Scope
- Entry/Exit Criteria
- Traceability to Requirements
- Pass/Fail Criteria
- Suspension/Resumption Criteria
- Defect Tracking
- Test Environment
- Schedule
- Roles and Responsibilities
- Risk and Contingencies
- Approvals
- Glossary

Note: Any test plan template may be used as long as it includes all of the required sections listed above. A Test Plan template may be found in the active templates folder of the FDOT Project Delivery Methodology SharePoint Site.

3.0 Functional Test Cases / Test Scripts

All test cases must include the following information:

3.1 Test Case Identifier

All test cases must be assigned a unique identifier. The test ID must follow the following formula:

- Whole Number = Test type (ex. 1.0, 2.0, 3.0)
- Decimal Place = All other test types (ex. 1.01, 2.01, 3.01)

3.2 Requirement ID

The Requirement ID must be included as provided in the Requirement Deliverable document.

3.3 Test Description

All test cases must be given a test description. The description must be written in plain English and must describe a specific test.

3.4 Test Created By

All test cases must contain the name of the individual who created the test case. If the test case was modified the name of the person making the modification must be included.

3.5 Test Executed By

Each test case must indicate the person assigned to perform the test.

3.6 Pre-Condition

Each test case must describe any preconditions that must be met prior to performing the test. For example, Unit testing must be completed prior to System testing.

3.7 Step Number

Each test case must include the individual step numbers required to complete the test. The test step ID must follow the below format:

- Whole Number = Test step number (ex. 1, 2, 3)

3.8 Test Step

Each test case must include a numbered list of the steps to perform in the system[1] which makes it easier to understand the test case.

3.9 Test Data

The test data can be entered directly in the test data field or a separate file that contains test data for one or more test cases can be referred.

3.10 Expected Results

All tests must provide the expected outcome of the test to be performed.

3.11 Outcome

All tests must provide the outcome of the test performed.

3.12 Pass/Fail

All tests must provide a pass/fail indicator. The format of the indicator must be as follows:

- Pass
- Fail

3.13 Comments

Comments may or may not be provided.

Example:

Project Name:						
Version:						
Test Case ID: _____ Test Created By: _____ Requirement ID: _____ Test Priority : _____ Test Description: _____ Test Executed By: _____						
Preconditions:						
Test Steps						
Step #	**Test Step**	**Test Data**	**Expected Results**	**Actual Results**	**Status(Pass/Fail)**	**Comments**
Post Conditions/Results:						

Note: SharePoint, TFS, Test Manager or some other Test Management tool may be used for designing/managing the test cases. A Test Case template may be found in the active templates folder of the FDOT Project Delivery Methodology SharePoint Site.

4.0 Testing Types

All projects must perform a variety of types of testing to ensure that code is error free and working correctly in the FDOT environment. At a minimum, all projects must perform and produce results for Unit Testing, System Testing, User Acceptance Testing, and Post Implementation Testing. Other types of testing are highly recommended for every project, and may be required per the procurement instrument and/or project plan for specific projects. All test results and defects must be tracked in a log.

4.1 Unit Testing (REQUIRED)

Unit testing involves the testing of individual units or modules of source code to determine whether they perform according to defined requirements and are error free.

4.2 System Testing (REQUIRED)

System testing includes testing of the entire, integrated system, prior to User Acceptance Testing. System Testing should be performed by someone other than the developer who wrote the code in order to provide an unbiased, second opinion. System testing should test all code, modules, and interfaces with other applications, per predefined Test Cases, to ensure that all parts of the application are functioning correctly and per defined requirements.

4.3 Performance/Load Testing (RECOMMENDED)

Performance/Load testing helps determine the responsiveness, throughput, reliability and/or scalability of a system under a given workload. Performance/Load testing may be performed manually, or through the use of an automated testing tool. While Performance/Load testing is not required for all projects, it should be performed for any project that has defined requirements related to system performance.

4.4 Regression Testing (RECOMMENDED)

Regression Testing verifies that software that was previously developed and tested still performs correctly after it was changed or interfaced with other software. A full regression test is recommended to be performed for any major upgrades to an existing application.

4.5 Application Security Scanning (RECOMMENDED)

All applications should be scanned for security vulnerabilities using an Application Security Scanning tool such as OWASP/ZAP to reveal any potential security threats, prior to being migrated to the production environment. Vulnerabilities must be documented and should be reviewed by the FDOT Security Team.

4.6 User Acceptance Testing/UAT (REQUIRED)

Acceptance Testing is performed by the ACTUAL users of the Application. During UAT, users test the APPLICATION to make sure it can handle required tasks in real-world scenarios according to documented requirements. UAT must not be initiated until programs are in place, functioning correctly according to specification, having passed all reviews (i.e. code review, web standards review, etc.) and fully integrated in the system test environment. UAT signoff must be obtained before any new or modified data structure, program, or other application component is implemented in the production environment.

4.7 Post Implementation Testing (REQUIRED)

This type of testing occurs immediately after the application has been migrated to Production to ensure that it functions as expected in the new environment.

5.0 Defect Tracking

All projects must include a defect tracking mechanism to manage the defects/bugs found during testing. Documentation of defects must include the following information:

- Site Address
- Business Area
- Time and Date the issue occurred
- Detailed description of the issue
- Detailed steps to recreate the issue
- Screen shots of the issue
- Severity of the issue
- Priority of the issue
- Status of defect

Note: SharePoint, TFS, Test Manager, Test Director or some other Defect Management tool may be used for managing the system defects/bugs.

6.0 Traceability Matrix

All projects must demonstrate that requirements have been adequately covered by testing through the use of a Traceability Matrix. The Traceability Matrix must show the relationship between test cases and defined requirements.