Scoped App Test Plan

<Type your Application name here>

Template version: 4.0

Template effective date: January 26, 2014

# Table of Contents

1 Overview 3

2 Design document breakdown 3

3 Performance Test Cases 4

4 Load Test Cases 4

5 Functionality Test cases 5

6 Debugging demonstration 5

# Overview

Your Test Plan document is intended to enable ServiceNow Certification Engineers and Technical Support members to test your application and ascertain its quality. The document is used internally by ServiceNow only and is not shared with your customers.

To prepare your Test Plan, enter a thorough response in each section of this template document. Upload the completed document to your application record on the publisher portal.

ServiceNow performs a full run-through of your completed test plan with the intention of meeting the following high-level objectives:

1. All mitigating factors (processes) that were called out in the Design Document are demonstrated; either spelled out explicitly in this document or outlined for demonstration in the test plan run through.
2. All performance thresholds or constraints that were called out in the Design Document are demonstrated; either spelled out explicitly in this document or outlined for demonstration in the test plan run through.
3. All test cases are written in such a way that anyone, with little to no understanding of the application, can use the Test Plan to execute the full test suite. Any setup and configuration of the instance is outlined, and any dependencies on third-party or standalone software are specified. If customers will receive an installation guide with the application, the guide should be attached to the Design Document.

# Design document breakdown

In the Design Document, you identified the processes that are in place to ensure both platform security and performance. In this section, you demonstrate how the processes were tested or are enforced.

*<insert text here>*

*<example: >*

*“In the Application Design Document, section 6.2.1, we outlined the potential for large data exports to negatively impact the performance of the ServiceNow Platform. The following procedural mitigating factors are in place to illuminate the likelihood of this occurring:*

1. *The installation/configuration guide delivered to the customer outlines how many records should be exported at one time. The guide clearly states that no more than 50k records should be exported at any one time. Additionally, we call out breaking up the records into chunks so that the upper limit is never reached. See the attached copy of the installation/configuration guide for evidence of these instructions.*
2. *We have coded a configuration option for the user to specify how many records will be transferred at one time. To see this, navigate to the properties page on the Admin Module of the application. The user can specify how many records will be exported and when. Additionally, we have built-in logic to disallow any user-entered values greater than 50k. This further prevents the user from exporting too much data at any one time. We test this functionality in TC018. Please see section 4 below to see the results of that test case.*

# Performance Test Cases

Resource usage, scalability and reliability of the product are validated under this testing. Include all the test cases that were executed when performance testing was done. Also specify the yardstick used for the same. The primary goal of performance testing includes establishing the benchmark behavior of the system.

|  |  |  |  |
| --- | --- | --- | --- |
| *Test Case* | *Setup* | *Expected Result* | *Actual Result* |
| *PTC-001* | *Provide the setup sets associated with this test case* | *Provide the expected result associated with this test case* | *Pass/Fail*  *(This should represent the results of the most recent execution)* |
| *PTC-002* | *Provide the setup sets associated with this test case* | *Provide the expected result associated with this test case* | *Pass/Fail*  *(This should represent the results of the most recent execution)* |

# Load Test Cases

The goals of load testing are to expose the defects in application related to buffer overflow, memory leaks and mismanagement of memory. It can be used to check the any functionality of the application by flooding the system with 1000 users at a time.

|  |  |  |  |
| --- | --- | --- | --- |
| *Test Case* | *Setup* | *Expected Result* | *Actual Result* |
| *LTC-001* | *Provide the setup sets associated with this test case* | *Provide the expected result associated with this test case* | *Pass/Fail*  *(This should represent the results of the most recent execution)* |
| *LTC-002* | *Provide the setup sets associated with this test case* | *Provide the expected result associated with this test case* | *Pass/Fail*  *(This should represent the results of the most recent execution)* |

# Functionality Test cases

Include all test cases that were executed when testing your integration. Each function and each major use case of the integration should have been tested.

*<insert text here>*

*<example: >*

|  |  |  |  |
| --- | --- | --- | --- |
| *Test Case* | *Setup* | *Expected Result* | *Actual Result* |
| *TC-001* | *Provide the setup sets associated with this test case* | *Provide the expected result associated with this test case* | *Pass/Fail*  *(This should represent the results of the most recent execution)* |
| *TC-002* | *Provide the setup sets associated with this test case* | *Provide the expected result associated with this test case* | *Pass/Fail*  *(This should represent the results of the most recent execution)* |

# Debugging demonstration

In this step, we observe as you cause an error in your integration and work through the process of identifying and resolving the issue. Provide the methods used in this process in the “Debugging and Troubleshooting Tips" section of your Design Document. When you decide which failure scenario to choose, consider that this scenario should expose as many as possible of the debugging and troubleshooting methods in the integration.

Provide the following:

* High-level description of the scenario:
  + *<insert text here>*
* Steps to identify the issue in logs/UI actions/and so on:
  + *<insert text here>*
* Steps to correct the issue:
  + *<insert text here>*

End of Document