

## Software Test Plan

Release	Revisions	Approved	Date
1	Initial release	BES	10/24/2008

# Table Of Contents

<b>Table Of Contents</b>	<b>2</b>
<b>Introduction</b>	<b>3</b>
Purpose . . . . .	3
Scope . . . . .	3
Assumptions and Constraints . . . . .	3
<b>Test Plan</b>	<b>3</b>
Items Tested . . . . .	3
Items Not Tested . . . . .	4
Deliverables . . . . .	4
Techniques . . . . .	4
Environment and tools . . . . .	4
Tasks . . . . .	5
Organization . . . . .	5
<b>Templates</b>	<b>5</b>
Test Case Template . . . . .	5
Test Procedure Template . . . . .	6
Test Log Template . . . . .	6
Test Incident Report Template . . . . .	6
<b>Test Summary Report</b>	<b>7</b>

# Introduction

## Purpose

The purpose of this document is to define the overall system test strategy, the activities performed to prepare and conduct system and acceptance testing, and the format of other testing documents.

## Scope

This plan covers testing for the Cascadia system, including the application server, bridge components, and client applications.

## Assumptions and Constraints

It is assumed that not all features are available from all UI platforms. For instance, though it is not explicitly stated in the Terms of Reference, it is unlikely that a user is required to be able to post a new MLS listing over SMS.

Although it is not specified, it is assumed that all non-client systems will require an administration interface.

It is assumed that the product architecture will separate the main application server from the bridge components. This assumption may be invalidated when the SADD becomes available.

# Test Plan

## Items Tested

- Identification – Account database, credentials, and subscription-level access rights; personal profiles and preferences access from all UI platforms.
- Property database integration – Integration points with the existing database are subject to rigorous testing.
- Posting – Creation of new listings, including input validation.
- Searching – On-demand searches from all relevant UI platforms.
- Notifications – Delivery and options access from all relevant UI platforms.
- Service directory – Access to MLS service referrals from all relevant UI platforms.
- Messaging – User configuration, message delivery, and automatic selection of communications channel.

- Calendaring – Appointment scheduling, notification, and reminders on all relevant UI platforms.
- Forms Management – Interface and processing for all supported forms, on all relevant UI platforms.
- Performance – Response time of database and intranet queries on all relevant UI platforms.
- Fault Tolerance – Single-failure response of CPU and storage systems, up-time, and live upgrades.
- Scalability – Load testing with 5 years of projected growth.

## Items Not Tested

- Property database – Oracle on a Red Hat environment is a well-known configuration of a commercial product. Since this subsystem has well-known performance and reliability characteristics, and is not under the direct control of the development team, it will not be tested.

## Deliverables

The following items are required as part of this plan:

- Software Test Description (STD)
- Software Test Report (STR)
- Test/Requirements Matrix

## Techniques

PARTITIONING will be used to construct test suites, and ensure that all requirements are covered by the test procedures.

OPERATIONAL PROFILING will be used to prioritize test suites and partitions according to empirical and predicted feature demand.

## Environment and tools

Three primary test environments are anticipated for system testing.

1. Mobile client environment – Requires an application server and a mobile phone/PDA with Internet, e-mail, IM, and SMS access.
2. Desktop client environment – Requires an application server and a desktop PC with Internet, e-mail, IM, and fax access.
3. Server environment – Requires the application server and a workstation with access to the administration interfaces.

## Tasks

STRUCTURAL TESTING – In accordance with industry best practices for this type of project, a set of automated unit tests will be produced and run regularly in order to maintain low-level code consistency.

AUTOMATED ACCEPTANCE TESTING – A set of automated acceptance tests will be developed and used to guide both requirements elicitation and ongoing development.

ISOLATION TESTING – Each independent subsystem will be tested in isolation of other subsystems by constructing mocks for the system dependencies.

INTEGRATION TESTING – Two or more subsystems are tested in combination, using mocks for missing subsystems.

## Organization

The development and test teams is organized into the following divisions:

- Server team – develops/tests the primary application server which connects to the main database.
- Bridge team – develops/tests the systems which bridge SMS, IM, fax, and e-mail to the application server.
- Client team – Develops/tests the web and native applications for use on PCs, laptops, and mobile phones.
- Integration team – Develops mocks for integration testing, and tests the various subsystems in combination with each other.

## Templates

The following sections are to be used as templates. Text in *italics* is to be replaced with information specific to the item being produced.

### Test Case Template

#### 1. Test Case Name

- Requirements Tested
  - *List requirements tested by this case.*
- Relevant Operational Profiles
  - *List OPs used to construct this case.*
- Environment/Setup
  - *List environment, setup and tools requirements.*

## Test Procedure Template

### 1. Procedure Name

- Test Case: *Index and name of test case.*
- Expected outcomes:
  - *List expected outcomes.*
- Notes
  - *List any special considerations.*
- Test Steps
  - ☐ *Step 1*
  - ☐ *...*

## Test Log Template

### Test run dated *date, time*

- Performed by *Name*
- Procedures
  - *List all test procedures completed.*
- Notes
  - *List all deviations from standard environment, incidents logged, or other anomalies.*

## Test Incident Report Template

- Defect summary (10 words or less)
  - Subsystem(s): *Apparent location of failure*
  - Versions
    - \* *Versions of all systems involved in test*
  - Environment
    - \* *Describe test environment.*
  - Reproduction Steps
    - \* *Steps*
    - \* *...*
  - Expected: *expected results*
  - Actual: *actual results*

## Test Summary Report

The project manager will be kept apprised of test team progress through the use of weekly status reports delivered via email. As work items are completed, they will be delivered via email to the customer, and inserted into version control; monthly meetings with the customer will be held to review new work products and revisions to already-delivered items.

Once testing activities have begun, weekly status reports from testers will include test logs. Incident reports are to be input into the defect tracking system, and their status is to be kept up-to-date as development progresses. Automated testing results will be continuously available through the project portal hosted on the intranet.