

# Connections Academy

## CMS 3.5 Membership Management TEST PLAN

### Revision History

Rev:	Date:	Submitted By:	Review By:
1	9/18/2009	Lyle Albright	Jonathan Wilt (QA Team Lead/Mgr) Project Stakeholders

# TABLE OF CONTENTS

<b>OVERVIEW OF PROJECT</b> .....	<b>3</b>
<b>OVERVIEW OF THE DOCUMENT</b> .....	<b>3</b>
<b>TEST PLAN</b> .....	<b>3</b>
<b>1 OBJECTIVES</b> .....	<b>5</b>
<b>2 INTRODUCTION</b> .....	<b>5</b>
<b>3 FEATURES TO BE TESTED</b> .....	<b>5</b>
<b>4 FEATURES <u>NOT</u> TO BE TESTED</b> .....	<b>6</b>
<b>5 APPROACH</b> .....	<b>6</b>
<b>6 TYPES OF TESTS</b> .....	<b>7</b>
<b>7 TEST PROCEDURES</b> .....	<b>8</b>
<b>7.1 Software Releases</b> .....	<b>8</b>
<b>7.2 Problem Reporting</b> .....	<b>8</b>
<b>7.3 Response to Problems Reported</b> .....	<b>9</b>
<b>8 SUSPENSION CRITERIA &amp; RESUMPTION REQUIREMENTS</b> .....	<b>9</b>
<b>8.1 Suspension Criteria</b> .....	<b>9</b>
<b>8.2 Resumption Requirements</b> .....	<b>9</b>
<b>9 TEST DELIVERABLES</b> .....	<b>9</b>
<b>9.1 Test Deliverables Prior to Testing</b> .....	<b>9</b>
<b>9.2 Test Deliverables After Testing</b> .....	<b>10</b>
<b>10 TESTING TASKS</b> .....	<b>10</b>
<b>11 ENVIRONMENTAL NEEDS</b> .....	<b>10</b>
<b>11.1 Physical characteristics</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.2 Communications software</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.3 System software</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.4 Modes of use</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.5 Security</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.6 Test Tools</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.7 Availability</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.8 Volume</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.9 Response Time</b> .....	<b>Error! Bookmark not defined.</b>
<b>11.10 Staffing</b> .....	<b>Error! Bookmark not defined.</b>
<b>12 PASS/FAIL CRITERIA</b> .....	<b>11</b>

<b>13</b>	<b>SYSTEM OVERVIEW</b> .....	<b>11</b>
<b>14</b>	<b>DIAGRAMS</b> .....	<b>11</b>
	<b>14.1 System Overview</b> .....	<b>11</b>
<b>15</b>	<b>RESPONSIBILITIES</b> .....	<b>11</b>
<b>16</b>	<b>STAFFING AND TRAINING NEEDS</b> .....	<b>12</b>
	<b>16.1 Staffing Needs</b> .....	Error! Bookmark not defined.
	<b>16.2 Training Requirements</b> .....	Error! Bookmark not defined.
<b>17</b>	<b>SCHEDULE</b> .....	<b>12</b>
	<b>17.1 Project Schedule</b> .....	<b>12</b>
	<b>17.2 Testing Milestones and Item Delivery Schedule</b> .....	<b>12</b>
<b>18</b>	<b>RISKS ASSUMPTIONS</b> .....	<b>13</b>
	<b>18.1 High-Risk Assumptions</b> .....	<b>13</b>
<b>19</b>	<b>APPROVALS</b> .....	<b>14</b>
<b>20</b>	<b>TEST CASES</b> .....	<b>14</b>
	<b>20.1 Test Case Dependencies</b> .....	<b>14</b>
	<b>20.2 Special Requirements</b> .....	<b>14</b>
<b>21</b>	<b>TEST REPORTS</b> .....	<b>14</b>
<b>22</b>	<b>DEFINITIONS, ACRONYMS, AND ABBREVIATIONS</b> .....	<b>14</b>
<b>23</b>	<b>REFERENCE DOCUMENTS</b> .....	<b>14</b>
<b>24</b>	<b>ATTACHMENTS</b> .....	<b>14</b>
	<b>24.1.1 TEST CASES</b> .....	Error! Bookmark not defined.
	<b>24.1.2 TEST REPORTS</b> .....	Error! Bookmark not defined.

## OVERVIEW OF PROJECT

Client XYZ has requested a simple membership management feature added into their public facing website. The website is built on a popular Content Management System architecture and is user and role authenticated. Client XYZ manually maintains two distinct membership levels through an MS Access database. The database is currently passed from member to member via CD as the non-profit organization has no brick and mortar building or network to share the database.

## OVERVIEW OF THE DOCUMENT

This section provides an overview of the Software Test Plan and will define the following for the (project) during the (phase) Phase:

- Test Plans
- Test Designs
- Test Case Specifications
- Test Procedures
- Test Cases and Reports
- Reference documents
- Definitions

## TEST PLAN

This section summarizes test team's objectives as well as the software items and features to be tested. A justification of the need for testing may also be included.

The structure of this chapter is as follows:

1. **Objectives-** Details the objective's of the testing effort.
2. **Introduction-** Summarizes the software items and software features.
3. **Features to be Tested-** Identifies the features to be tested.
4. **Features not to be Tested-** Identifies the features not to be tested.
5. **Approach-** Specifies the major activities, methods and tools that will be used to test the designed groups or failed testing.
6. **Types of Tests-** Describes the types of tests to be performed for this project.
7. **Test Procedures-** Describes the established QC test procedures.
8. **Suspension Criteria and Resumption Requirements-** Specifies the criteria used to suspend all, or a part of the testing activities on the test items associated with this plan.
9. **Test Deliverables-** Identifies the items that must be delivered before testing begins and the items that must be delivered when testing is finished.
10. **Testing Tasks-** Specifies the tasks necessary to be completed before testing begins.
11. **Environmental Needs-** Describes the necessary configurations or properties of the test environment.
12. **Pass/Fail Criteria-** Specifies the criteria to be used to decide whether each test item has passed or failed.

13. **System Overview-** Provides an overview of the system to be tested.
14. **Diagrams-** Displays system flow &/or data flow under test through diagrams.
15. **Responsibilities-** Specifies roles and responsibilities.
16. **Staffing and Training Needs-** Describes staffing needs according to skill level and identifies training requirements.
17. **Schedule-** Documents test milestones identified in the software project schedule and all item delivery events.
18. **Risks and Contingencies-** Identifies the high risk assumptions and contingency plan for each.
19. **Approvals-** Specifies the names and titles of all persons who must approve this plan.
20. **Test Cases-** Describes test case development approach; test cases are included as attachments.
21. **Test Reports-**Identifies and describes test reports; test reports are included as attachments.
22. **Definitions, Acronyms, and abbreviations-** Provides the definitions of al term, acronyms, and abbreviations used in the test plan.
23. **Reference Documents-** Specifies reference documents used in writing the test plan.
24. **Attachments-** Includes any applicable attachments to this test plan such as test cases and test reports.

## 1 OBJECTIVES

The testing team will conduct an exhaustive, risk-based, change test approach on the new feature of the CMS system. To make sure this new feature is accessible to potential operators, the testing team will ensure compatibility with the top 80% of popular browser and operating system (platform) combinations. Due to time constraints, to facilitate this objective, the testing team will need to conduct parallel automated browser and platform testing on our preconfigured virtual environment platform to ensure one hundred percent coverage of the new feature.

It is also the objective of this test effort to verify high level CMS system regression functions as a result of the new feature being added to the system. Quality Control will execute 100% of our system acceptance tests, an some portion of functional regression tests against targeted components of the system that are suspect to the nature of the change being made (to be determined via code review with Software Development).

An acceptance level system performance test will also be in scope for this release to verify the new feature has no significant performance impact on the system.

## 2 INTRODUCTION

The software items to be tested for this release consist of the entire 3.5 version of the CMS product. That includes all web files, configuration files and data files to be installed per the release notes as well as the CMS SQL database.

All existing features of the CMS product will be acceptance tested, whereas the new feature (membership management) will be exhaustively tested.

In execution of this system level, end-to-end test effort, Quality Control will also execute basic acceptance test of the backend server services and maintenance jobs.

## 3 FEATURES TO BE TESTED

The specific features to be tested are as follows:

- CMS core capabilities (database connectivity, authentication/security, I/O, logging, reports, etc.)
- User/Group Permissions
- Content
- Events
- Forum
- Theme
- Contact Management
- Module Management

## Membership Management

### **4 FEATURES NOT TO BE TESTED**

All features of the CMS are to be tested.

### **5 APPROACH**

Quality Control will conduct exhaustive change testing against the new membership management module, and acceptance level regression testing against all other features of the system. Additional regression tests will be added to the test set against those features of the CMS suspected to have interoperability issues as a result of adding the membership management feature.

Quality Control will break down the test effort into a traditional waterfall test method that consists of three planned releases from development before release to production. The release will be labeled for this plan as follows:

1. Initial Release
2. Fix Cycle 1
3. Fix Cycle 2

On the initial release of the CMS 3.5 code to Quality Control, we will execute 100% of our planned change test cases against the code, and 100% of the planned acceptance regression cases. Since the code is least stable during the initial release, this approach will help to ensure the greatest run-time execution on the new feature and drive the highest defect detection rates of the test effort. The result will help to ensure the development team will have as many defects detected as early in the test process as possible for lead time to fix.

Fix Cycle 1 will execute only 50% of the planned change test cases, to include all high priority and some low priority tests. QC will execute 100% of the planned acceptance regression cases in this cycle. The approach here is that the code has become more stable in the second release from Software Development, allowing the Quality Control team to accept a high level of lowered risk in order to shorten the overall test execution time to meet the project time constraints.

Fix Cycle 3 will execute only 25% of the planned change test cases and 100% of the planned acceptance regression cases. Again, the code is most stable in this release, and a bit more risk can be taken in the interest of the project time constraints.

Each release, regardless of cycle, will employ an automated test framework against a virtual environment of in scope browser and platform combinations. This will ensure adequate platform compatibility compliance per the project requirements.

QC work will be structured in order that:

- duplication of effort will not occur
- testing will be performed as much as possible in a parallel effort
- functional testing, interface testing, and end-to-end testing for data integrity will try to employ the same test cases
- all problems and problem resolutions will be documented throughout the process
- implementation of necessary repairs to software will occur in a controlled manner
- the most critical functionality will be tested first – the project stakeholders will decide what is most critical
- prior to external site alpha/beta deployment, all known problems will be reviewed with alpha/beta testers
- during alpha/beta testing, all problems encountered will be identified, documented, and reported as they occur
- testing will be performed on a variety of PC configurations by Development, QC, alpha/beta users, subject matter experts, and Integration
- the test and repair cycle allows time for the full suite of tests

## 6 TYPES OF TESTS

This release will consist of the following test types-

- *Documentation Tests* – to ensure project documents have the complete and accurate information required for Quality Control.
  - *External Acceptance tests (Product should work on an acceptance level before delivery to Quality Control)*
  - *Review of Unit test reports (Adequate unit testing for unit coverage to reduce unnecessary critical and high defects during final systems testing)*
  - *Release Notes*
- *Baseline Testing:* Will be conducted on all test beds to ensure the product worked properly before the update was applied as well as the opportunity to verify an prior regression defects being addressed in this release. This approach will allow the tester first hand verification of the defect and a more connected understanding of its manifestation.
- *Acceptance Testing:*
- *Installation Testing* – Ensuring the product has been properly installed so that useful testing can follow. The review an installation packaging logic and installation features as well as uninstall o rollback testing to protect the business environment in the event of a catastrophic CMS failure related to the update.
- *Functional Testing*

- *Change Testing* – Tests designed to specifically test those features, behaviors of the system that are new to the changes being made.
- *Regression Testing* – To ensure that any changes to the system and/or software do not corrupt or otherwise have an adverse impact on any segment of the application, and that the changes follow accepted business rules. Regression testing is performed on modified software to provide confidence that modifications behave as intended and do not adversely affect the functionality of unmodified codes. Regression testing will be performed each time a new version of the program is completed by the Development team and provided to the QC team.
- *Interoperability Testing* - To ensure that after the upgrade has been applied all aspects of the system(s) continue to operate as designed without the introduction of any new defects.
- *Performance Testing* – To ensure that the system continues to perform at a comparable rate after the upgrade (new feature) has been applied.
- *Compatibility Testing* – Will take place to ensure the system is compatible with all in scope browser and operating system combinations. This will take place in a virtual environment and will be conducted through the use of an automation tool.

## 7 TEST PROCEDURES

Internal Quality Control testing materials such as the daily testing log will be housed on the QC server 'srv123.' Documents for external publication such as test plans and test cases will be kept in XYZ source control.

Internal testing of the CMS 3.5 application will begin once the Testing Team is assembled and will continue through the final release of the application.

Testing will take place in the QC Test Lab, and on desktops.

### 7.1 Software Releases

Upon release of new software the Testing Team uses installation documents prepared by the Development Team to install or update the software in the lab. The Testing Team runs a full suite of tests using the test scripts found later in this document.

### 7.2 Problem Reporting

Problems, issues and desired enhancements to the software will be logged in HP TestDirector software.

### **7.3 Response to Problems Reported**

After receiving and reviewing the problems reported by the test team, the Development Team adds comments and returns them to the Test Team. The Test Team then retests against the problem log to verify Developer comments.

Continuing with the processes for problem documentation and reporting, the Testing Team runs a full suite of tests on the integrated system.

## **8 SUSPENSION CRITERIA & RESUMPTION REQUIREMENTS**

*This section specifies the criteria that will be used to suspend all, or a part of the testing activities on the test items associated with the plan. In addition, it specifies the testing activities that must be repeated when testing is resumed.*

### **8.1 Suspension Criteria**

Specified are the criteria that will be used to suspend all, or part of the testing activities on the test items associated with this plan here.

- A new build cannot be installed.
- The documentation is so lacking that testing cannot be performed.
- The software is failing at a rate that testing cannot be performed.

### **8.2 Resumption Requirements**

Specified is the resumption requirements for testing activities that must be repeated when testing is resumed.

- A new installation procedure document is sent.
- The programming team rectifies issues preventing the installation.
- A new release containing software fixes is delivered to QC.

## **9 TEST DELIVERABLES**

### **9.1 Test Deliverables Prior to Testing**

Identified are the items that must be delivered before testing begins which should include:

- Test Plan
- Test Cases
- Test Procedures
- Test Input Data
- Test Tools

## 9.2 Test Deliverables After Testing

Identified are the items that must be delivered when testing is finished which should include:

- Test reports
- Test output data
- Problem reports

## 10 TESTING TASKS

Identifies are the set of tasks necessary to prepare for and perform testing.

- Assemble testing team
- Determine testing procedures and approach
- Configure QC integration lab
- Training
- Implement automated testing tools

## 11 ENVIRONMENTAL NEEDS

*This section specifies the necessary and desired properties of the test environment. The testing environment is set up to mirror the production environment. Additionally, the test environment includes testing tools, which will be excluded from production. The testing lab is located in XXXX.*

The test environment will include the following:

- Physical characteristics of the facilities including hardware
- Communications software
- System software
- Modes of use (i.e., standalone, networked)
- Security
- Test Tools
- Availability

- Volume
- Response Time
- Staffing

## **12 PASS/FAIL CRITERIA**

This section specifies the criteria to be used to decide whether each test item has passed or failed testing.

## **13 SYSTEM OVERVIEW**

*This section will provide an overview of the system in test.*

Enter the description of the system.

## **14 DIAGRAMS**

*This section lists applicable diagrams used in the testing process.*

### **14.1 System Overview**

This section will provide an overview of the system flow diagrams.

*Figure 1:*

*Figure 2:*

## **15 RESPONSIBILITIES**

This section defines the team members responsible for managing, designing, preparing, executing, witnessing, and checking tests.

Team members include developers, operations staff, user representatives, technical support staff, data administration staff, independent verification personnel and quality control staff.

<b>Team Member Name</b>	<b>Role</b>	<b>Responsibilities</b>
Lyle Albright	QC Analyst	<ul style="list-style-type: none"> <li>• Develop test plans</li> <li>• Manage/Assist testers</li> <li>• Maintain project schedules</li> <li>• Analyze results Develop test cases</li> <li>• Perform testing</li> <li>• Maintain testing log</li> </ul>
	QC Lab Technician	<ul style="list-style-type: none"> <li>• Manage Integrated Lab</li> <li>• Prepare technical documentation</li> </ul>
Jonathan Wilt	QC Project Manager	<ul style="list-style-type: none"> <li>• Manage QC testing team</li> <li>• Develop project schedules</li> <li>• Configure project</li> <li>• Set project goals</li> </ul>
TBD	Development QC Analyst	<ul style="list-style-type: none"> <li>• Perform component testing at development level</li> <li>• Liaison between QC and MN developers</li> </ul>
TBD	QC Analyst	•
TBD	QC Director	•

## 16 STAFFING AND TRAINING NEEDS

*This release requires no additional staffing or training needs..*

## 17. SCHEDULE

This section contains information about the project schedule, test milestones and item delivery events.

### 16.1 Project Schedule

Please see the attached QC MS-Project Project schedule.

### 16.2 Testing Milestones and Item Delivery Schedule

This section includes test milestones identified in the software project schedule and all item delivery events, for example:

## **18 APPROVALS**

This section specifies the name and title for all persons who must approve this plan.

## **19 TEST CASES**

This section will contain a description of the procedures for writing test cases. Identify the documentation used to write test cases here.

Test cases will be found as attachments.

### **19.1 Test Case Dependencies**

This section describes any test case condition that must be performed before a test case can be executed.

- Test cases will be release dependent.

### **19.2 Special Requirements**

This section identifies any special requirements for the execution of a test case.

## **20 TEST REPORTS**

This section will contain any applicable test reports.

## **21 DEFINITIONS, ACRONYMS, AND ABBREVIATIONS**

This section should provide the definitions of all terms, acronyms, and abbreviations used in the plan.

## **22 REFERENCE DOCUMENTS**

This section provides a complete list of all reference documents.

## **23 ATTACHMENTS**

This section specifies the supporting materials to the test plan.