Acceptance Test Plan for Mashbot

George D'Andrea Andrew Gall Josiah Kiehl Cody Ray Vito Salerno

January 7, 2010

Revision History

Name	Date	Reason for Changes	Version
George D'Andrea, Andrew Gall, Josiah Kiehl, Cody Ray, Vito Salerno	1 December 2009	Initial Version	1.0

Contents

1 Introduction

1.1 Background

This document describes the battery of tests the Mashbot product will be required to pass in order to be considered successful. Mashbot is a tool for managing advertising campaigns and collecting data from social networks. It is described in detail in the Software Requirements Specification for Mashbot.

1.2 Structure of Document

- Section 2 - Describes the overall approach to the Acceptance Test Plan
- Section 3 - Describes in more detail features covered or not covered by the Acceptance Test Plan
- Section 4 - Describes the criteria which must be satisfied to begin and complete the Acceptance Test Plan
- Section 5 - Describes the roles and responsibilities of the staff members involved in the Acceptance Test plan and procedures for reporting test results and testing problems
- Section 6 - Describes the actual test cases in the Acceptance Test Plan

1.3 References

Software Requirements Specification for Mashbot

1.4 Glossary

- Test Team Leader - The person in charge of all the testers.
- Project Leader - The person in charge of the whole project.
- Clients Representatives - Clients Representatives are people who overlook the Acceptance Test Plan execution on behalf of the customers
- Software Requirements Specification - A Software Requirements Specification is a document which describes the behaviour of a system.
- Functional Requirements - Functional Requirements define the internal workings of the software.
- Unit Tests - A procedure in the software to validate that individual modules and other units of source code are working properly
- Integration Test - Integration Test is a test phase which is employed after the Unit Tests are validated and tests how all the different modules of a software system fit and work together with each other.
- System Test - System Test is conducted after Integration Test has been conducted to evaluate the systems compliance with its specified requirements

2 Test Approach and Restraints

2.1 Introduction

This section describes the overall approach, techniques and testing tools to be used in Acceptance Test Plan for the Mashbot and any constraints that may apply.

2.2 Test Objectives

The Acceptance Test Plan process will examine Mashbot and verify whether it fulfills the requirements set forth in the Software Requirements Specification.

2.3 Test Structure

The Acceptance Test Plan will consist of a subset of test cases and methods, previously used in the Unit Tests, Integration Test and System Test for Mashbot. The test cases will be carefully selected to allow for the verification of the functional requirements of Mashbot as listed in the Software Requirements Document. It is essential that all appropriate Unit Tests, Integration Tests and System Tests were successfully performed on Mashbot prior to the Acceptance Tests and their results were reported and presented to the client.

3 Test Assumptions and Exclusions

3.1 Introduction

This section describes details about what aspects of Mashbot will actually be covered within the scope of this Acceptance Test Plan document and what will not be covered.

3.2 Assumptions

The Acceptance Test Plan covers:

- Functional requirements of Mashbot listed in the Software Requirements Specification
- Usability of Mashbot
- Consistency of user documentation for Mashbot

3.3 Exclusions

The Acceptance Test Plan does not cover:

- Non-functional requirements of Mashbot besides usability, as listed in the Software Requirements Specification
- Quality of code

4 Entry and Exit Criteria

4.1 Introduction

This section specifies criteria that must be met in order for the Acceptance Test Plan to be executed. Additionally, it specifies the criteria that must be met for success and failure of that acceptance test plan.

4.2 Entry Criteria

The Acceptance Test Plan can be executed when these preconditions are met:

- Mashbot has successfully passed testing including:
  - Integration Testing
  - Systems Testing

  - Unit Testing

- A testing environment has been setup which reflects the system requirements as stated in our Software Requirements Specification.
- Copies of the latest versions of both the Software Requirements Specification and other user-related documentation has been received.
- The latest released version of Mashbot has been appropriately resourced.
- Consent of the following people to begin the acceptance test plan has been given:
  - Client
  - Project Leader
  - Test Team Leader

4.3 Exit Criteria

A run of the Acceptance Test Plan has several possible outcomes:

- Success: All **Priority 1** requirements wer tested and performed as required.
- Failure:
  - At least one of our **Priority 1** requirement deviated from expected behavior.
  - The Acceptance Test Plan was rescheduled on approval from the Client and Test Team Leader.

5 Testing Participants

5.1 Introduction

This section escribes the roles and responsibilities of people involved in the Acceptance Test Plan process, as well as how to report results.

5.2 Roles and Responsibilities

Test Team Leader George D'Andrea

User Representative A member of the Drexel College of Engineering faculty or the advisor to this project who will overview the testing process

Tester A person who will execute the tests

5.3 Training Requirements

Everyone involved with the test process should be familiar with Mashbot, its user interace, the documentation, and the Software Requirements Specification.

5.4 Problem Reporting

Any problems discovered must be reported to the Test Team Leader, then eventually the project Administrative Lead, to be fixed.

5.5 Progress Reporting

Following the test procedure, the Test Team Leader will compile a report to submit to the Administrative Lead.

6 Testing Project Test Cases

6.1 Introduction

The test cases are divided into sections covering parts of the functionality and use cases in the Software Requirements Specification. Each test case has the following format:

- Name - The name of the test case
- Preconditions - Conditions needed to initiate the test case
- Actions - The actions expected from a tester
- Post conditions - The expected outcome of the test case

6.2 Test Cases

6.2.1 User Account Creation

Preconditions	The user visits the Mashbot registration page in a web browser.
Actions	The tester enters a username, password, and relevant information in the appropriate fields and submits for registration.
Postconditions	The tester is registered in the system if all field verifications are successful. or The tester is not registered in the system if any field verification is not successful.

6.2.2 User Account Modification

Preconditions	The user visits their profile / settings page in a web browser.
Actions	The tester enters updated information in any field, except for username, and submits for update.
Postconditions	The tester is registered in the system if all field verifications are successful. or The tester is not registered in the system if any field verification is not successful.

6.2.3 Logging In

Preconditions	The user has visited the Mashbot web page in a web browser.
Actions	The tester enters his username and password in the appropriate fields and submits for verification.
Postconditions	The tester is logged in to the system if the verification is successful. or The tester is not logged in to the system if the verification is not successful.

6.2.4 Logging Out

Preconditions	The tester is logged into the Mashbot web application.
Actions	The tester clicks the logout button.
Postconditions	The tester is logged out of the system and their session is destroyed.

6.2.5 Login Timeout

Preconditions	The tester is logged into the Mashbot web application.
Actions	The tester does not take any action for a predefined amount of time.
Postconditions	The tester is logged out of the system and their session is destroyed.

6.2.6 Campaign Creation

Preconditions	User is logged into the Mashbot web application.
Action	User selects “Create” tab.
Action	User fills out all fields presented on the “Create” view.
Action	User clicks submit button.
Postconditions	New campaign is created.

6.2.7 Campaign Modification

Precondition	User is logged into the Mashbot web application
Precondition	At least one campaign has been created.
Precondition	User is viewing the list of all campaigns.
Action	User clicks to edit the campaign.
Reaction	User is redirected to the “Edit” view for a campaign.
Action	User changes fields.
Action	User presses “Save” button.
Postcondition	Campaign is updated with changed fields.
Postcondition	User is taken to view list of all campaigns

6.2.8 Add Campaign Content

Precondition	User is logged into the Mashbot web application
Precondition	User is viewing list of campaigns.
Action	User selects a campaign to which to add content.
Reaction	User is taken to “Add Content” view.
Action	User clicks “Add New Content” button.
Reaction	A view to select content type appears.
Action	User selects content type.
Reaction	User is taken to page to edit content for given content type.
Action	User completes content.
Postcondition	Content is added to the campaign.
Postcondition	User is taken to the “Add Content” view.

6.2.9 Modify Campaign Content

Precondition	User is logged into the Mashbot web application.
Precondition	User is viewing a campaign.
Action	User clicks on a piece of content to edit.
Reaction	User is given the “Edit Content” view.
Action	User completes changes.
Action	User clicks the save button.
Postcondition	User is directed back to the Campaign view.
Postcondition	Content is saved.

6.2.10 Content Scheduler

Schedule Content

Precondition	User is logged in to the Mashbot web application.
Precondition	User is on the "Schedule" view.
Action	User clicks and drags a piece of content from the Content Bucket to the calendar.
Postcondition	Content is scheduled for the date the date at which the user dropped the content.
Postcondition	A view is given to add the time for the date for which the content is scheduled.

Unschedule Content

Precondition	User is logged in to the Mashbot web application.
Precondition	User is on the "Schedule" view.
Precondition	At least one piece of content is already scheduled.
Action	User clicks and drags scheduled content from view to Content Bucket.
Postcondition	Content is removed from the calendar and is unscheduled.

Edit Scheduled Time

Precondition	User is logged in to the Mashbot web application.
Precondition	User is on the "Schedule" view.
Action	User clicks a piece of content on the Calendar.
Postcondition	User is given the "Edit Scheduled Time" view.

Edit Scheduled Date

Precondition	User is logged in to the Mashbot web application.
Precondition	User is on the "Schedule" view.
Precondition	At least one piece of content is already scheduled.
Action	User drags content from one date to another.
Postcondition	Content's date is changed to date on which the content is dropped.

6.2.11 Content Publisher

Precondition	Content is scheduled to be published.
Action	Scheduled time of publishing occurs.
Postcondition	Content is published to the service it was scheduled to be published to, and at the time it was scheduled for.

6.2.12 Adding External Service Account

Precondition	User is logged in and viewing the External Service Account control panel.
Action	User clicks "Add Account".
Reaction	User is given fields to authenticate the account. or User is redirected to an authentication page on the external service itself.
Action	User fills in account credentials in provided fields. or User affirms authentication on authentication page.
Postcondition	User is approved to use the account.
Postcondition	Authentication credentials are stored for future use.