|  |
| --- |
| SClogoVertSmallPanTest Design Specification  Sitecore Media Uploader |
|  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Feature Responsible | | | | |
|  | | |  |
| Product Manager: | Gabriel Aguilo / [gaa@sitecore.net](mailto:gaa@sitecore.net) | |
| ProJECT Manager: | Nataliya Yatsenko / [nyt@sitecore.net](mailto:nyt@sitecore.net) | |
| Development: | Alexander Dodatko / [adk@sitecore.net](mailto:adk@sitecore.net); Igor Khomich / igk@sitecore.net | |
| Test: | Alina Zozulya / [alr@sitecore.net](mailto:alr@sitecore.net); Elena Taran / [elt@sitecore.net](mailto:elt@sitecore.net) | |

|  |  |  |
| --- | --- | --- |
| Document Finalization | | |
| ACTIVITY | PARTICIPANTS | DATE |
| REVIEW | [Names/email alias] |  |
| Review Meeting | [Names/email alias] |  |
| APPROVAL | [Names/email alias] |  |

|  |  |  |
| --- | --- | --- |
| Revision History | | |
| **Date** | **Author** | **Changes Made** |
| [2014-04-08] | Elena Taran  ([elt@sitecore.net](mailto:elt@sitecore.net)) | Initial version of document created |
|  |  |  |
|  |  |  |

# Contents

[1 Contents 2](#_Toc384721935)

[2 Document Scope 2](#_Toc384721936)

[2.1 Introduction 2](#_Toc384721937)

[2.2 Feature Functionalities /Requirements Covered 2](#_Toc384721938)

[3 Feature Test Brief 3](#_Toc384721939)

[3.1 Scenarios Covered 3](#_Toc384721940)

[3.2 Testing Approach 3](#_Toc384721941)

[3.3 Testability and Test Constraints 5](#_Toc384721942)

[3.4 Test Data Requirements 6](#_Toc384721943)

[3.5 Test Environment Requirements 6](#_Toc384721944)

[3.6 Cross Feature Dependencies/Assumptions 6](#_Toc384721945)

[3.7 Risks 6](#_Toc384721946)

[4 Tests Identification 6](#_Toc384721947)

[4.1 Test Cases Outline 6](#_Toc384721948)

[4.2 Suspension and Resumption Criteria 6](#_Toc384721949)

[4.3 Exit Criteria 6](#_Toc384721950)

[5 Test tools 7](#_Toc384721951)

[6 References 7](#_Toc384721952)

[6.1 Documents 7](#_Toc384721953)

[6.2 Test Types & Methodologies for reference: 7](#_Toc384721954)

# Document Scope

## Introduction

This document is created to describe our plans for Sitecore Media Uploader.

Project consists from iOS Application

Sitecore Media Uploader should work on the following configurations:

* Sitecore CMS version 6.6 SP1 or later;

Sitecore Media Uploader should support iPad, iPhone with iOS version 7.0 and higher

Application should support Portrait orientation.

Sitecore Item Web API 1.2

## Feature Functionalities /Requirements Covered

|  |  |  |
| --- | --- | --- |
| **Feature or Area** | **Description** | [**Risks**](#Risks) **/ Dependencies** |
| **View default screen after installing Sitecore Up** | When user installs the app – application must show:   * Page with three folders: Upload, Browse,   MyUploads.   * **Settings** button at top panel * **Sitecore** logotype at the bottom | Application should look well in different screen sizes. |
| **Editing “Settings”** | User can edit saved information on view “Settings”.  User can input Url site without ‘http(s)://’ and taping on buttons (‘http’ and ‘https’) can select it. | Check Item Web API security and Sitecore security of the User to access an edited instance and folder (they may have different access) |
| **My Uploads** | User can view pending files to be upload later, all uploaded files and not completed uploads.  User can cancel or apply uploading media for pending files. User can see progress indicate during media is uploading.  Media files are sorted by time in My Uploads.  User can clear the uploading history.  User can retry the uploading for failed uploads | Check working with uploading many pending files at once.  Check working with unstable or unavailable internet connection and with insufficient local memory. |

# Feature Test Brief

## Scenarios Covered

Test plan for full smoke testing and regression testing for Sitecore Media Uploader (Test cases in TFS)

<http://tfs4dk1.dk.sitecore.net/tfs/PD01/Product_Mobile_Apps/_testManagement#planId=14&suiteId=37> (for Android and iOS)

## Testing Approach

1. The next test activities should be performed for every sprint:

* **Smoke/Sanity**

All the manual test scenarios for smoke testing should be executed;

* **Regression**

All the manual test scenarios for regression should be executed for configurations listed in “Configuration/Compatibility testing” chapter;

* **Testing bug fixes/feature implementations**

1. The next test activities should be performed every sprint:

* **Security/Security Vulnerability**

Security in application should be provided with/without rights to Write/Read media files from CMS or Item Web API module.

Security Vulnerability will not be performed for this product;

* **Localization**

English version should be localized. Localization for English should be tested;

* **Performance/Load**

Performance testing will be provided for reading/creating many items, large media files in CMS and Uploader.

1. Media Sizes:
2. *Images:*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| File extension | File name | Original size | Resolution | Internet connection (3G) | Internet connection (WI-FI) |
| png | 80 KB .png | 80 KB | 191x167 (75 dpi) |  |  |
| png | green\_mineraly\_big\_picture2.png | 74 MB | 5496x10288 (300 dpi) |  |  |
| jpg | 2 MB (2).jpg | 2 MB | 1344x10288 |  |  |
| jpg | green\_mineraly1 (2).jpg | 4.2 MB | 4459x5880 (300 dpi) |  |  |
| gif | green\_mineraly1.gif | 10 MB | 4080x5880 (75 dpi) |  |  |
| gif | 15 MB.gif | 15 MB | 3752x10288 (300 dpi) |  |  |

1. *Videos:*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| File extension | Original size | File name | Size after compressing | Resolution | Video length | Compressing time | Internet connection (3G) | Internet connection (WI-FI) |
| MOV | 45 MB | IMG\_0994.MOV |  | 1280x720 | 59 sec |  |  |  |
| MOV | 251 MB | IMG\_0995.MOV |  | 1280x720 | 4 min 38 sec |  |  |  |
| MOV | 362 MB | IMG\_0997.MOV |  | 1280x720 | 6 min 5 sec |  |  |  |
| MOV | 900 MB | IMG\_0996.MOV |  | 1280x720 | 11 min 53 sec |  |  |  |

1. Uploading several media items at once:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| File extension | File name | Original size | Resolution | Count | Internet connection (3G) | Internet connection (WI-FI) |
| png | 2\_22MB.png | 2.22 MB | 1920x1200 | 3 |  |  |
| png | 2\_22MB.png | 2.22 MB | 1920x1200 | 10 |  |  |
| png | 2\_22MB.png | 2.22 MB | 1920x1200 | 50 |  |  |
| jpg | 1\_91MB.jpg | 1.91 MB | 3264x2448 | 1 |  |  |
| jpeg | 3\_26MB.jpg | 3.26 MB | 3264x2448 | 20 |  |  |
| MOV | IMG\_0994.MOV | 45 MB (59 sec) |  | 2 |  |  |

* **Equivalence Partitioning**

Checking loading media files with different size.

* **Configuration/Compatibility testing**

Configuration testing will be provided for following configurations:

|  |  |
| --- | --- |
| Mobile Devices | iPad 3 Retina  iPad Air  iPhone 5  iPad 2 |
| OS versions | iOS 7.1  iOS 7.0.2 |
| CMS versions | CMS  6.6.0 Update 6 rev.130529 (Service Pack -1)  CMS 7.0 Update 4 rev.140120  CMS 7.1 Update 2 rev.140324  CMS 7.2 Update 2 rev.140314 |
| Item Web API | [Sitecore Item Web API module v1.2.0 rev.140128](http://sdn.sitecore.net/downloads/SitecoreItemWebAPI120rev140128zip.download) |

* **Usability testing**

Navigation controls and design of application should be iOS specific. Interface should be user-friendly and comfortable organizing. UI appearance should be high quality designed.

* **Installation/Uninstallation**

Installation will be tested using Testflight app during the development.

Uninstallation testing should check if all saved data is removed after the application uninstall;

* **Documentation**

All documentation that is going to be published on SDN and AppStore should be checked.

All main features, changes, fixes should be described in documentation. It should correspond to the latest application changes.

Testers should pay attention not to the grammar, style etc., but to the correctly described features of the application. Attention should also be paid to features or information which is omitted from the documentation, but which, in the expert evaluation of the tester, should be included.

All procedures described in the document should be tested. All code samples should be tested and, it should be ensured, at a minimum that these samples compile without error.

* **Recovery**

Testing application with device low battery charge, not stable network connection, turn off the device, lack of free disk memory of the device, interruption testing (incoming calls, alerts, messages, press “Menu” button, screen lock, portrait/landscape orientation, change user location);

* **Fault Injection**

During this testing activity all input fields (instance parameters, uploading media parameters) should be filled with failed/invalid parameters (manual);

Checking application behaviour when we upload wrong beaten files, word, excel, png files, files with different extensions.

* **Exploratory**

1. The test cases / smoke tests should be created for new features as early as possible

Test cases / smoke tests should be automated as soon as UI freeze is stated for a new feature.

## Testability and Test Constraints

1. Functional testing:

* Main functional requirements should be covered with unit tests;

1. UI testing:

* All UI features should be tested manually (appropriated test cases should be created in [TFS](http://tfs4dk1.dk.sitecore.net/tfs/PD01/Product_Mobile/IOS%20Developers/_testManagement));

1. Other testing types:

* Other testing activities as security testing, stress and load testing, performance testing could be executed manually;

## Test Data Requirements

For testing Sitecore Media Uploader it’s necessary to have:

1. Stable builds of Sitecore CMS 6.6.0 SP1 and later;
2. Stable build of Sitecore Item Web API Module v1.2.0 rev.140128;
3. Stable build of Sitecore Media Uploader;
4. Different media files in devices media library.

## Test Environment Requirements

1. Several iOS devices (iPad, iPhone) to test application on different iOS versions (7.0 and later) and device simulator;
2. Mac machine with installed Xcode;
3. Jenkins CI server;
4. Win PC to handle testing CMS instances;

## Cross Feature Dependencies/Assumptions

1. Sitecore Media Uploader retrieves data from Sitecore instance so it depends on CMS features. On tests failing you should check backend first.
2. Sitecore Media Uploader uses Sitecore security model to access to Media Library, so it depends on changes in Sitecore Security model.
3. Application retrieves data using Sitecore Item Web API, so it depends on API stability.
4. Application uses Mobile SDK framework and it depends on framework stability.

## Risks

1. Incomplete requirements may affect estimates;
2. Depending on using Sitecore Item Web API service quality and possibilities;
3. Parallel projects may get a higher business value, which would lead to delay in Sitecore Media Uploader testing.

# Tests Identification

## Test Cases Outline

All test cases for full smoke testing and regression testing for Sitecore Mobile Media Uploader are stored in [TFS](http://tfs4dk1.dk.sitecore.net/tfs/PD01/Product_Mobile_Apps/_testManagement#planId=14&suiteId=37);

## Suspension and Resumption Criteria

When the Smoke tests fail due to product defect, further testing will be suspended and resumption of testing can happen only upon the product defect being fixed in the next build.

## Exit Criteria

Existing features: All the smoke and regression manual tests passed successfully. Test cases for regression testing are written according to issues in change log.

New features: All the smoke and regression manual tests passed successfully; test cases created according to new features should be passed as well.

Configuration activity passed successfully.

# Test tools

* Access to test cases in [TFS](http://tfs4dk1.dk.sitecore.net/tfs/PD01/Product_Mobile_Apps/_testManagement#planId=14&suiteId=37);
* Several iOS devices (iPad 3 Retina, iPad 2, iPad Air, iPhone 5) to test application on different hardware and OS versions;
* Mac machine with installed Xcode, [AntEater](https://www.redant.com/anteater/);
* Jenkins CI server;
* TestFlight for built applications distribution;

# References

## Documents

|  |  |  |
| --- | --- | --- |
| **Document** | **Author** | **Location** |
| **Functional / Requirement Spec** | GAA | <http://tfs4dk1.dk.sitecore.net/tfs/PD01/Product_Mobile/IOS%20Developers/_backlogs/backlog/Backlog%20items> |
| **Test Case** | ALR, ELT | <http://tfs4dk1.dk.sitecore.net/tfs/PD01/Product_Mobile_Apps/_testManagement#planId=14&suiteId=37> |
| **Test Design Specification** | ALR, ELT | <http://tfs4dk1.dk.sitecore.net/tfs/PD01/Product_Mobile/IOS%20Developers/_git/iOS_Media_Uploader#path=%2FTesting+Documentation%2FiOS_Media_Uploader+TDS.docx&version=GBmaster&_a=contents> |

## Test Types & Methodologies for reference:

|  |
| --- |
| **Test Types & Methodologies** |
| Functional |  |
| Performance |  |
| Scalability |  |
| Installation/Uninstallation |  |
| Upgrade & Backward Compatibility |  |
| Configuration |  |
| Smoke/Sanity |  |
| Regression |  |
| Security |  |
| Exploratory |  |
| Documentation |  |
| Integration |  |
| System |  |
| Recovery |  |
| Compatibility |  |
| User Experience |  |
| User Interface |  |
| Equivalence Partitioning |  |
| Boundary Value |  |
| Fault Injection |  |

**Note:** Refer to the document ‘Sitecore Products QA Overview’ for a detailed description of these test types.

Look for the heading ‘Test Types & Methodologies’ in the document [here](http://intranet.dk.sitecore.net/Departments/Quality-Assurance/Processes/Sitecore-Products-QA-Overview.aspx).