# Basic testing of Content Cue

Change history:

|  |  |  |
| --- | --- | --- |
| Date | Author | Change / State |
| 2010-08-04 | Pruchkovskaya T. | Created |

# Scope

The testing has been made for Product Content Cue. Some bugs were reported to developers. Here there is summary of work.

# List of Testing Scenario

List of description Testing Scenario

## Testing functionality of each tab

### Setting Tab

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Database description | Precedent conditions | Operation | Expectation | Actual result |
| 1 | Select or type the source of the documents | There are a lot of files and folders different types in definite directory | Necessary to show path to document and choose button keep or (and) include | User should be able to define if files from nested folder also should de available to be added into metadata file and to define if keeping folders hierarchy required. The application should first ask the user if they would like to do this. If the information is invalid, must be message about error | Functions |
| 2 | The number of files found in the source and size of all files | - | - | The Gui should highlight to the user the number of files found in the source and size of all size. If the document source is empty or does not contain any files to be uploaded, must be message - warning. | Functions |
| 3 | Select existing metadata file | User have metadata file | To select existing metadata file. | User should be able to select existing metadata file after that the data on Metadata tab should be reloaded. If metadata file XML is invalid or file specified does not exist - message about error. If metadata XML file is not defined - message "warning". | Functions |
| 4 | The log level (xml format) | The button with levels - low, medium, high | Choose level and press a button | User should to choose level | Functions |
| 5 | Logon settings | Not logged in | To log in | Should be a message - logged in as domain\_name/user\_name. If not specified the Content Cue tools assumes the current logged on user | Functions |
| 6 | Sharepoint Destination Url, Document library and folder | User had some library and folders | To enter the Destination site URL and Document Library | User should not enable to type the name of the library manually and be able to type the name of destination folder manually too. If the information is invalid, must be message about error | Functions |
| 7 | Save changes into settings file | User entered changes. | The new settings file is being saved first time. The user tries to close the application. The user tries to launch upload. The user tries to open another settings file. | The application should reminder the user to save changes (and provide with "Save file" dialog box if required) | Functions |
| 8 | Open existing settings file | User had existing settings file. | Open existing settings file | The application should validate the settings file format and show the error message in case the file format is not valid. | Functions |

### Metadata Tab

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Database description | Precedent conditions | Operation | Expectation | Actual result |
| 1 | The Mapping grid | User had information in the mapping grid. | To verify a mechanism to apply missing metadata. | The Mapping grid should display documents from the source and provide a mechanism to apply missing metadata. | Functions |
| 2 | Impossible to apply metadata | - | - | Where is not possible to apply metadata this files should be marked and moved to an "UploadFailed" folder as described in Functional requirements above. | Functions |
| 3 | No "MetadataMapping.xml" | - | - | In the case that there is NO "MetadataMapping.xml" the default is existing properties such as DocumentTitle and Author. | Functions |
| 4 | Upload of manage file | User had a manage file | Select files to be added into metadata from selected source folder. Remove files from metadata. | Manage files to be uploaded | Functions |
| 5 | Manage mapped fields | User had information in the mapping grid. | To add a new mapping field. To remove the existing mapping field. To select the mapping field name from the pre-populated list. | This function should be to work. | Functions |
| 6 | Import data from CSV file | The application should validate the CSV file format. | To press "import" in the main menu. | This function should be to work. Show the error message in case the file format is not valid. | Functions |
| 7 | Clear metadata | User had information in the mapping grid. | To press "clear" in the main menu. | The mapping grid should be empty. | Functions |
| 8 | Save metadata file | User had metadata file. | The new metadata file is being saved first time. The user tries to launch upload. The user tries to close the application. The user tries to launch upload. The user tries to open another settings file. | The application should reminder the user to save changes (and provide with "Save file" dialog box if required) | Functions |
| 9 | Metadata - unavailable status | The Metadata consists of a field which is absent or is unavailable status | - | If the Metadata consists of a field which is absent or is unavailable status in the designated document library - Read only field, invalid column should be highlighted red. | Functions |
| 10 | Document is absent | Document is absent in the defined source folder | - | Should be shown corresponding icon in the status column. | Functions |
| 11 | Document size > 50MB | Document size is greater than max allowed (50 MB) | - | Should be shown corresponding icon in the status column. | Functions |
| 12 | Mandatory field is empty | Mandatory field is empty | - | Should be shown corresponding icon in the status column, the empty cell should be highlighted red. | Functions |
| 13 | Upload documents | User had a documents where the minimum metadata requirements | To upload | The screen should allow the user to upload documents where the minimum metadata requirements have been met. (ie: All mandatory fields are provided). | Functions |

### Log Tab

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Database description | Precedent conditions | Operation | Expectation | Actual result |
| 1 | View application log file | User had completed application | User launches the application | The application creates a new log file for every work session. The log file name should have the following format: Date\_And\_Time\_Of\_Application\_Launching. The information within log file should be grouped by upload attempts. The log file should include the following information depending on defined logging level (low, medium, high). | Functions |

## Testing of Main menu

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № | Database description | Precedent conditions | Operation | Expectation | Actual result |
| 1 | [File] | [New], [Open], [Save], [Save as], [Launch Upload], [Exit] | All function should to perform | Creates new setting file. Shows "Open file" dialog where user can select required settings file. Saves changes in currently open settings file. Launches upload process. Close the application. | Functions |
| 2 | [Metadata] | [Import], [Add field], [Remove field], [Remove file], [Clear] | All function should to perform | Imports metadata from CSV file. Adds new mapping field column. Removes selected row with uploading file name. Clear metadata mapping. | Functions |
| 3 | [Help] | [View help], [About ContentCue] | All function should to perform | Shows help system window (link to html.file). | Functions |
| 4 | Tabs Section | [Settings], [Metadata], [Log] | All function should to perform | All function should to perform | Functions |
| 5 | Buttons Section | [Save settings], [Upload] | All function should to perform | The "Save settings] button allows the user to save current configuration settings. The "Upload" button allows the user to upload to the destination. | Functions |
| 6 | Status Bar | - | - | The status bar should display full path to currently open settings file. | Functions |

## Verifying Function scenarios requirement according to specification

|  |  |
| --- | --- |
| *Database description* | *Conformance specification* |
| 1. The tool will allow the user to select the source of the documents from a file share via the GUI. | Conforms |
| 2. The tool will allow the user to map the file share source to a SharePoint document library via the GUI. | Conforms |
| 3. The tool will allow the user to create a mapping XML file via the GUI on available metadata fields within a grid view (Schema Mapping Tab). | Conforms |
| 4. The tool should pick up existing document property mappings of the documents such as Title and Keywords where possible. | Conforms |
| 5. The mapping should show which fields can be mapped and which fields can’t be mapped. a. Ex: The document title field and document author should be populated. b. Ex: The CSV file can be created beforehand in excel to populate the column names (By means of import functionality). | Conforms |
| 6. The tool will allow the user to save the current settings in XML format. | Conforms |
| 7. The tool will allow the user to open a previous setting file and populate the settings. | Conforms |
| 8. The tool should show progress for each upload. | Conforms |
| 9. The tool should have configurable logging enabled – Low, Medium, High a. Low = Writes to the log number of successful uploads and number of unsuccessful uploads with the names of the files. b. Medium = Writes to the log number of successful uploads and number of unsuccessful uploads with the name of the files and the size of each file uploaded and if any schema errors are detected. c. High = Writes to the log number of successful uploads and number of unsuccessful uploads with the name of the file and the size of each file uploaded and if any schema errors are detected and the error and which document failed with the schema error. | Conforms |
| 10. The log file should be written to a folder in XML format. | Conforms |
| 11. The tool will allow the user to upload multiple documents in batches via the GUI. | Conforms |
| 12. The tool should continue uploading content when there are multiple upload errors and should write an error log on failed uploads. | Conforms |
| 13. The tool should provide a view of the documents in the source folder and have the ability to select only a few documents for upload. | Conforms |
| 14. The tool should allow the user to move the upload failure documents to a folder titled “Filed Upload” a. Optionally the tool should move these files automatically to this folder or provide a mechanism to easily move the failed documents. | Conforms |
| 15. The tool should have the following structure when deployed: ~:\ indicates a drive such as C:\ a. ~:\ContentCue.exe (Main executable) b. ~:\ContentCue.Config (Configuration file) c. ~:\Configuration \(All supporting files for the programme) d. ~\Logs\(All Log files) e. ~\FailedUpload\ Upload\_Date\_And\_Time\(All failed Conforms upload files) f. ~\Help\ (All help files) | Conforms |

## Verifying Non Function requirement according to specification

|  |  |
| --- | --- |
| *Database description* | *Conformance specification* |
| • The tool must work on x64 (64bit) platforms | Conforms |
| • The tool should work with SharePoint 2007 and SharePoint 2010\* including SharePoint Foundation | Conforms |
| • The tool should not require any pre-requisite software other than .Net Framework 3.5 to be present for the tool to function | Conforms |
| • The tool should upload all document (electronic) formats including images | Conforms |
| • The tool should warn when a single document size exceeds 50MB in size and reject uploading and provide the user with a warning | Conforms |
| • The current .csv Metadata mapping should be converted to XML format | Conforms |

# Stress Tests

|  |  |
| --- | --- |
| *Database description* | *Conformance specification* |
| Prepare library of 300 files for loading. They should be different types (user common types as .doc, .pdf, .jpg and so on. Prepare 10 files of different types with size greater than 50 MB. | The action of system conforms to specification |

# Smoke/Acceptance Test

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *№* | *Database description* | *Precedent conditions* | *Operation* | *Expectation* | *Actual result* |
| 1 | Select or type the source of the documents | There are a lot of files and folders different types in definite directory | Necessary to show path to document and choose button keep or (and) include | User should be able to define if files from nested folder also should de available to be added into metadata file and to define if keeping folders hierarchy required. The application should first ask the user if they would like to do this. If the information is invalid, must be message about error | Should conform to specification |
| 2 | Logon settings | Not logged in | To log in | Should be a message - logged in as domain\_name/user\_name. If not specified the ContentCue tools assumes the current logged on user | Should conform to specification |
| 3 | SharePoint Destination Url, Document library and folder | User had some library and folders | To enter the Destination site URL and Document Library | User should not enable to type the name of the library manually and be able to type the name of destination folder manually too. If the information is invalid, must be message about error | Should conform to specification |
| 4 | Save changes into settings file | User entered changes. | The new settings file is being saved first time. The user tries to close the application. The user tries to launch upload. The user tries to open another settings file. | The application should reminder the user to save changes (and provide with "Save file" dialog box if required) | Should conform to specification |
| 5 | Open existing settings file | User had existing settings file. | Open existing settings file | The application should validate the settings file format and show the error message in case the file format is not valid. | Should conform to specification |
| 6 | The Mapping grid | User had information in the mapping grid. | To verify a mechanism to apply missing metadata. | The Mapping grid should display documents from the source and provide a mechanism to apply missing metadata. | Should conform to specification |
| 7 | Impossible to apply metadata | - | - | Where is not possible to apply metadata this files should be marked and moved to an "UploadFailed" folder as described in Functional requirements above. | Should conform to specification |
| 8 | No "MetadataMapping.xml" | - | - | In the case that there is NO "MetadataMapping.xml" the default is existing properties such as DocumentTitle and Author. | Should conform to specification |
| 9 | Manage mapped fields | User had information in the mapping grid. | To add a new mapping field. To remove the existing mapping field. To select the mapping field name from the pre-populated list. | This function should be to work. | Should conform to specification |
| 10 | Clear metadata | User had information in the mapping grid. | To press "clear" in the main menu. | The mapping grid should be empty. | Should conform to specification |
| 11 | Save metadata file | User had metadata file. | The new metadata file is being saved first time. The user tries to launch upload. The user tries to close the application. The user tries to launch upload. The user tries to open another settings file. | The application should reminder the user to save changes (and provide with "Save file" dialog box if required) | Should conform to specification |
| 12 | Save metadata file | User had metadata file. | The new metadata file is being saved first time. The user tries to launch upload. The user tries to close the application. The user tries to launch upload. The user tries to open another settings file. | The application should reminder the user to save changes (and provide with "Save file" dialog box if required) | Should conform to specification |
| 13 | Document is absent | Document is absent in the defined source folder | - | Should be shown corresponding icon in the status column. | Should conform to specification |
| 14 | Document size > 50MB | Document size is greater than max allowed (50 MB) | - | Should be shown corresponding icon in the status column. | Should conform to specification |
| 15 | Mandatory field is empty | Mandatory field is empty | - | Should be shown corresponding icon in the status column, the empty cell should be highlighted red. | Should conform to specification |
| 16 | Upload documents | User had a documents where the minimum metadata requirements | To upload | The screen should allow the user to upload documents where the minimum metadata requirements have been met. (ie: All mandatory fields are provided). | Should conform to specification |
| 17 | View application log file | User had completed application | User launches the application | The application creates a new log file for every work session. The log file name should have the following format: Date\_And\_Time\_Of\_Application\_Launching. The information within log file should be grouped by upload attempts. The log file should include the following information depending on defined logging level (low, medium, high). | Should conform to specification |
| 18 | [File] | [New], [Open], [Save], [Save as], [Launch Upload], [Exit] | All function should to perform | Creates new setting file. Shows "Open file" dialog where user can select required settings file. Saves changes in currently open settings file. Launches upload process. Close the application. | Should conform to specification |
| 19 | [Metadata] | [Import], [Add field], [Remove field], [Remove file], [Clear] | All function should to perform | Imports metadata from CSV file. Adds new mapping field column. Removes selected row with uploading file name. Clear metadata mapping. | Should conform to specification |
| 20 | [Help] | [View help], [About ContentCue] | All function should to perform | Shows help system window (link to html.file). | Should conform to specification |
| 21 | Tabs Section | [Settings], [Metadata], [Log] | All function should to perform | All function should to perform | Should conform to specification |
| 22 | Buttons Section | [Save settings], [Upload] | All function should to perform | The "Save settings] button allows the user to save current configuration settings. The "Upload" button allows the user to upload to the destination. | Should conform to specification |
| 23 | Status Bar | - | - | The status bar should display full path to currently open settings file. | Should conform to specification |

# Conclusion

Consequently of testing a system was detected next mistakes: 1) the view of Main menu and Metadata are not correct; 2) the button ‘Save settings’ is not correct; 3) not message about hierarchy of folder; 4) not ‘warning’ in the Mapping grid when loaded files with size greater than 50 MB. All mistakes were removed.

The Product Content Cue is ready for construction QA-analysis.