



AUSTIN

USER TESTING PLAN

## IS480 – Final Year Project AY2011/2012 Term 2

**Project Name:** AUSTIN

**Date:** 20 September 2011

**Project Sponsor:** ESPN Starsports

**Project Client:** Nigel Harper

**Project Supervisor:** Professor Kyong

**Team Members:** Aw Ming Kun

Cassie

Victor li

Melvir Singh

## User Testing (UT)

### Project Overview:

An interactive web portal designed for ESPN Star Sports Staff/Clients to view and add video metadata in a guided environment. The web portal will be a comprehensive and flexible tool to replace existing complicated methods. The web portal will consist of Create Video Metadata, Dashboard and Manage Settings components.

Brief Description of the modules:

<b>Create Video Metadata</b>	This component allows users to select videos from FTP servers and input metadata around this video. Examples of such metadata includes Title, Description, Creation Date and etc.
<b>Dashboard</b>	This component allows users to view all video details and status that are in the system currently. These include videos that have not yet been started on being encoded, those being encoded, those that have been encoded successfully and those that have resulted in errors.
<b>Manage Settings</b>	This component allows users to customize dynamically the different fields relating to video metadata as well as miscellaneous settings.

### Purpose:

User Testing is performed to verify that the system meets the deliverables of the project. This is to ensure that the web portal will fulfill user expectations as defined in the business requirements and functional specifications.

### Objective:

To ensure that the web portal is bug-free and gain more feedback in terms of the usefulness and the usability of the AUSTIN web portal.

### Instructions:

For each of the tests below, run the test according to the test procedures. Observe the expected results. If the actual result on the AUSTIN web portal matches with the expected results, the test passes. Else if there are any discrepancies or differences from the expected results, note down the actual output in the "Actual Results" column.

Estimated time: 30minutes

### Test Cases for Admin users:

In order to test the following functions, please access the web portal, which has been deployed on ESPN Star Sport’s server at [deployment server here]:

**Module: Create Video Metadata**

S/N	Description of Test Cases	Actual Results	Pass/Fail
1	<p><b><u>Test Function:</u></b> List Files from FTP Server</p> <p><b><u>Test Procedure:</u></b> 1. Select appropriate FTP server from Source.</p> <p><b><u>Expected Results:</u></b> You’ll see that videos and images are populated in dropdown box and they are selectable.</p>	As Expected	Pass
2	<p><b><u>Test Function:</u></b> Preview Image Thumbnails (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b> 1. Select any image from the images populated by the FTP server.</p> <p><b><u>Expected Results:</u></b> You should see the image being previewed in a standard-sized box.</p>	As Expected	Pass
3	<p><b><u>Test Function:</u></b> View full-resolution version of image</p> <p><b><u>Test Procedure:</u></b> 1. Click on ‘Show me full resolution version of image’.</p> <p><b><u>Expected Results:</u></b> You should be able to see a popup window showing full size of image.</p>	As Expected	Pass
4	<p><b><u>Test Function:</u></b> Preview Image Thumbnails (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b> 1. Select any image from the images populated by the FTP server.</p> <p><b><u>Expected Results:</u></b> You should see the image being previewed in a standard-sized box.</p>	As Expected	Pass

User Testing PLAN

4	<p><b><u>Test Function:</u></b> Display metadata look and feel on Mobile/TV (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Click 'Next' to go to the next screen of filling up video metadata.</li> <li>2. Fill up 'Description' textbox.</li> </ol> <p><b><u>Expected Results:</u></b> You should see that the mobile and smarttv textbox gets filled in real-time with various sizes.</p>	As Expected	Pass
5	<p><b><u>Test Function:</u></b> jQuery Datepicker (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Click on 'Start Date' textbox.</li> </ol> <p><b><u>Expected Results:</u></b> You should be able to pick a date from the jQuery Datepicker instead of having to manually fill in the date.</p>	As Expected	Pass
6	<p><b><u>Test Function:</u></b> Video Tagging (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Type any tag you'd like to create into Tag Video textbox.</li> <li>2. Hit enter after every individual tag.</li> </ol> <p><b><u>Expected Results:</u></b> You should see the tags being added dynamically in real-time after hitting enter.</p>	As Expected	Pass
7	<p><b><u>Test Function:</u></b> jQuery Validation (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Leave some required field such as 'Title' or 'Description' empty.</li> <li>2. Click 'Confirm' to go to the next screen.</li> </ol> <p><b><u>Expected Results:</u></b> AUSTIN Web Portal does not allow you to move ahead with empty required fields. It prompts you</p>	As Expected	Pass

## User Testing PLAN

---

	to fill up empty required field and this error goes away in real-time.		
8	<p><b><u>Test Function:</u></b> Confirm Video Details (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>Once all the required fields have been filled up, click 'Confirm' to go into the next screen.</li> </ol> <p><b><u>Expected Results:</u></b> You should see the details that you've entered being previewed in the next screen. This preview also applies to videos/images.</p>	As Expected	Pass
9	<p><b><u>Test Function:</u></b> Save Video Details (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>Click 'Back'.</li> </ol> <p><b><u>Expected Results:</u></b> You should see the fields in the previous page were successfully saved and are being held in cookies until you close the entire browser.</p>	As Expected	Pass
10	<p><b><u>Test Function:</u></b> Submit Video Details (continue form the screen you saw in previous scenario)</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>Click 'Confirm'.</li> <li>Once you've checked all the details are correct, click 'Submt'.</li> </ol> <p><b><u>Expected Results:</u></b> You should see the new form telling you that your data has been saved successfully to the database and that Windows Service will now kick in to work with the encoder depending on the queue.</p>	As Expected	Pass

**Module: Dashboard**

S/N	Description of Test Cases	Actual Results	Pass/Fail
1	<p><b><u>Test Function:</u></b> Login</p> <p><b><u>Test Procedure:</u></b> 1. Load any form in the Web Portal without inputting user credentials.</p> <p><b><u>Expected Results:</u></b> You should be able to see a form telling you to login first.</p>	None	Fail
2	<p><b><u>Test Function:</u></b> Display all videos</p> <p><b><u>Test Procedure:</u></b> 1. On loading the 'Dashboard', there should be a data grid that by default loads all videos in the system, past and present.</p> <p><b><u>Expected Results:</u></b> You should be able to see 1. A data grid that shows all videos 2. That data grid is sorted according to the latest video created.</p>	As Expected	Pass
3	<p><b><u>Test Function:</u></b> List status and count of videos</p> <p><b><u>Test Procedure:</u></b> 1. On loading the 'Dashboard', there should be a portion on top that lists the different statuses with regards to videos. 2. Ensure that the count number is shown too.</p> <p><b><u>Expected Results:</u></b> You should be able to see 1. That the total count of videos in the system is split up into the different statuses in the system, which also has their respective count number. 2. There would be an option to 'View' the videos tagged to those statuses.</p>	As Expected	Pass
4	<p><b><u>Test Function:</u></b> View all videos currently tagged to a certain status</p>	As Expected	Pass

## User Testing PLAN

---

	<p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Click on 'View' for the different statuses that you'd like to test</li> <li>2. For this test, please go through the 'View' button for all the various statuses.</li> </ol> <p><b><u>Expected Results:</u></b> You should be able to see</p> <ol style="list-style-type: none"> <li>1. A data grid that displays videos according to the status clicked.</li> <li>2. That data grid is sorted according to the latest video created.</li> </ol>		
5	<p><b><u>Test Function:</u></b> Search videos.</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Click on 'Search'.</li> <li>2. Search on one or more fields based on your own criterion.</li> </ol> <p><b><u>Expected Results:</u></b> You should be able to see</p> <ol style="list-style-type: none"> <li>1. A data grid that populates based on your search result.</li> <li>2. That data grid is sorted according to the latest video created.</li> </ol>	As Expected	Pass
6	<p><b><u>Test Function:</u></b> Paging</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading the 'Dashboard', there should be a data grid that by default loads all videos in the system, past and present.</li> </ol> <p><b><u>Expected Results:</u></b> You should be able to see</p> <ol style="list-style-type: none"> <li>1. That data grid includes numeric paging.</li> <li>2. Each page click loads next page of videos and work appropriately going backwards and forwards.</li> </ol>	Paging does not work	Fail
7	<p><b><u>Test Function:</u></b> Re-queue</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading the 'Dashboard', there should be a data grid that by default loads all</li> </ol>	Requeue does not work	Fail

	<p>videos in the system, past and present.</p> <p>2. This data grid should also have a button that allows you to re-queue a video. This function basically restarts the video encoding process.</p> <p><b>Expected Results:</b> You should be able to see</p> <ol style="list-style-type: none"> <li>1. The 'Requeue' button and it is clickable.</li> <li>2. On clicking the button, the status of that video should be changed to 'Not Started', meaning it has been added to the queue for the Windows Service to work on.</li> </ol>		
--	--	--	--

**Module: Manage Settings**

S/N	Description of Test Cases	Actual Results	Pass/Fail
1	<p><b>Test Function:</b> Create Category</p> <p><b>Test Procedure:</b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Category'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol> <p><b>Expected Results:</b> You'll see that the newly added 'Category' appears in the refreshed data grid below.</p>	As Expected	Pass
2	<p><b>Test Function:</b> Edit Category</p> <p><b>Test Procedure:</b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Category'.</li> <li>3. In the data grid that populates below, choose 1 of the types and click on 'Edit'.</li> </ol>	As Expected	Pass

	<p>4. After changing the appropriate data, click on 'Update'.</p> <p>5. Do test that the 'Cancel' works accordingly too.</p> <p><b>Expected Results:</b> You'll see that the edited 'Category' has been updated in the refreshed data grid.</p>		
3	<p><b>Test Function:</b> Create Sport</p> <p><b>Test Procedure:</b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Sport'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol> <p><b>Expected Results:</b> You'll see that the newly added 'Sport' appears in the refreshed data grid below.</p>	As Expected	Pass
4	<p><b>Test Function:</b> Edit Sport</p> <p><b>Test Procedure:</b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Sport'.</li> <li>3. In the data grid that populates below, choose 1 of the types and click on 'Edit'.</li> <li>4. After changing the appropriate data, click on 'Update'.</li> <li>5. Do test that the 'Cancel' works accordingly too.</li> </ol> <p><b>Expected Results:</b> You'll see that the edited 'Sport' has been updated in the refreshed data grid.</p>	As Expected	Pass
5	<p><b>Test Function:</b> Create Channel</p>	As Expected	Pass

	<p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Channel'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the newly added 'Channel' appears in the refreshed data grid below.</p>		
6	<p><b><u>Test Function:</u></b> Edit Channel</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Channel'.</li> <li>3. In the data grid that populates below, choose 1 of the types and click on 'Edit'.</li> <li>4. After changing the appropriate data, click on 'Update'.</li> <li>5. Do test that the 'Cancel' works accordingly too.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the edited 'Channel' has been updated in the refreshed data grid.</p>	As Expected	Pass
7	<p><b><u>Test Function:</u></b> Create Profile</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Profile'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the newly added 'Profile' appears in</p>	As Expected	Pass

	the refreshed data grid below.		
8	<p><b><u>Test Function:</u></b> Edit Profile</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Profile'.</li> <li>3. In the data grid that populates below, choose 1 of the types and click on 'Edit'.</li> <li>4. After changing the appropriate data, click on 'Update'.</li> <li>5. Do test that the 'Cancel' works accordingly too.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the edited 'Profile' has been updated in the refreshed data grid.</p>	As Expected	Pass
9	<p><b><u>Test Function:</u></b> Create Service</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Service'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the newly added 'Service' appears in the refreshed data grid below.</p>	As Expected	Pass
10	<p><b><u>Test Function:</u></b> Edit Service</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Service'.</li> <li>3. In the data grid that populates below,</li> </ol>	As Expected	Pass

	<p>choose 1 of the types and click on 'Edit'.</p> <ol style="list-style-type: none"> <li>4. After changing the appropriate data, click on 'Update'.</li> <li>5. Do test that the 'Cancel' works accordingly too.</li> </ol> <p><b>Expected Results:</b> You'll see that the edited 'Service' has been updated in the refreshed data grid.</p>		
11	<p><b>Test Function:</b> Create ServiceType</p> <p><b>Test Procedure:</b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'ServiceType'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol> <p><b>Expected Results:</b> You'll see that the newly added 'ServiceType' appears in the refreshed data grid below.</p>	As Expected	Pass
12	<p><b>Test Function:</b> Edit ServiceType</p> <p><b>Test Procedure:</b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'ServiceType'.</li> <li>3. In the data grid that populates below, choose 1 of the types and click on 'Edit'.</li> <li>4. After changing the appropriate data, click on 'Update'.</li> <li>5. Do test that the 'Cancel' works accordingly too.</li> </ol> <p><b>Expected Results:</b> You'll see that the edited 'ServiceType' has been updated in the refreshed data grid.</p>	As Expected	Pass
13	<p><b>Test Function:</b> Create Status</p>	As Expected	Pass

	<p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Status'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the newly added 'Status' appears in the refreshed data grid below.</p>		
14	<p><b><u>Test Function:</u></b> Edit Status</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>6. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>7. Click on 'Status'.</li> <li>8. In the data grid that populates below, choose 1 of the types and click on 'Edit'.</li> <li>9. After changing the appropriate data, click on 'Update'.</li> <li>10. Do test that the 'Cancel' works accordingly too.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the edited 'Status' has been updated in the refreshed data grid.</p>	As Expected	Pass
15	<p><b><u>Test Function:</u></b> Create Tag</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Tag'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol>	As Expected	Pass

	<p><b><u>Expected Results:</u></b> You'll see that the newly added 'Tag' appears in the refreshed data grid below.</p>		
16	<p><b><u>Test Function:</u></b> Edit Tag</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Tag'.</li> <li>3. In the data grid that populates below, choose 1 of the types and click on 'Edit'.</li> <li>4. After changing the appropriate data, click on 'Update'.</li> <li>5. Do test that the 'Cancel' works accordingly too.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the edited 'Tag' has been updated in the refreshed data grid.</p>	As Expected	Pass
17	<p><b><u>Test Function:</u></b> Create Type</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> <li>2. Click on 'Type'.</li> <li>3. Fill up the textboxes.</li> <li>4. Click on 'Add'.</li> </ol> <p><b><u>Expected Results:</u></b> You'll see that the newly added 'Type' appears in the refreshed data grid below.</p>	As Expected	Pass
18	<p><b><u>Test Function:</u></b> Edit Type</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. On loading 'Manage Settings', there should be a navigation bar full of tabs relating to the different settings that can be customized such as Type, Category, Profile, etc.</li> </ol>	As Expected	Pass

## User Testing PLAN

---

	<ol style="list-style-type: none"><li>2. Click on 'Type'.</li><li>3. In the data grid that populates below, choose 1 of the types and click on 'Edit'.</li><li>4. After changing the appropriate data, click on 'Update'.</li><li>5. Do test that the 'Cancel' works accordingly too.</li></ol> <p><b><u>Expected Results:</u></b> You'll see that the edited 'Type' has been updated in the refreshed data grid.</p>		

**Module: AUSTIN Service – Basic/Monitor Service (Scenario 1: Normal Flow)**

1	<p><b><u>Test Function:</u></b> Move Video Asset from its respective Source folder to respective Watch folder(s) for encoding</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Create a new video asset entry via AUSTIN Web Portal</li> <li>2. Selected the number of service – make sure that folders are created on network</li> <li>3. Confirm the video asset entry</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Verify Video Asset are sent to its respective watch folders based on the profile settings for encoding E.g., iPad and MobileTV Watch folders</li> <li>• Dashboard to reflect <b>Sent to Watch Folder</b></li> </ul>		
---	---	--	--

**Module: AUSTIN Service – Basic/Monitor Service (Scenario 1: Exception Flow) (File Not Found)**

2	<p><b><u>Test Function:</u></b> Video Asset Not Found in Source Folder</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Delete the video asset or rename the video asset in the Source folder</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Dashboard will reflect Video Asset not found in Source Folder</li> <li>• Email with video assets details and error will be sent out</li> </ul>		
---	---	--	--

Module: AUSTIN Service – Basic Service (Scenario 1: Exception Flow) (File In Use)

3	<p><b><u>Test Function:</u></b> Video Asset is in use in Source Folder</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"><li>1. Open/Play the video asset in the Source Folder</li></ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"><li>• Dashboard will reflect Video Asset in use in Source Folder</li><li>• Email with video assets details and error will be sent out</li></ul>		
---	--	--	--

Module: AUSTIN Service – Verify Service (Scenario 1: Normal Flow)

<p>1</p>	<p><b><u>Test Function:</u></b> Verify Video Asset integrity in <u>Watch Folder</u> to ensure video asset is not damaged in transport</p> <p><b><u>Test Procedure (A):</u></b></p> <ol style="list-style-type: none"> <li>1. After the video asset is transferred from source to watch folder(s), check the log to see the video MD-5 is similar</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Verify MD-5 hash similar in log</li> <li>• Dashboard to reflect <b>Sent to Watch Folder</b></li> </ul> <p><b><u>Test Procedure (B):</u></b></p> <ol style="list-style-type: none"> <li>1. Replace the video asset in watch folder with other video asset</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Verify MD-5 hash different in log</li> <li>• Dashboard to reflect <b>File Damaged in Watch Folder</b></li> <li>• Email with video assets details and error will be sent out</li> </ul>		
<p>2</p>	<p><b><u>Test Function:</u></b> Verify Video Asset integrity in <u>FTP Folder</u> to ensure video asset is not damaged in transport</p> <p><b><u>Test Procedure (A):</u></b></p> <ol style="list-style-type: none"> <li>1. After the video asset is transferred from output to FTP folder(s), check the log to see the video MD-5 is similar</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Verify MD-5 hash similar in log</li> <li>• Dashboard to reflect <b>Sent to FTP Folder</b></li> </ul> <p><b><u>Test Procedure (B):</u></b></p> <ol style="list-style-type: none"> <li>1. Replace the video asset in FTP folder with other video asset</li> </ol> <p><b><u>Expected Results:</u></b></p>	<p>*Pending Fawad to activate hash function on FTP server. Hashing is only done on local pc</p>	

	<ul style="list-style-type: none"> <li>• Verify MD-5 hash different in log</li> <li>• Dashboard to reflect <b>File Damaged in Watch Folder</b></li> <li>• Email with video assets details and error will be sent out</li> </ul>		
--	---	--	--

**Module: AUSTIN Service – Verify Service (Scenario 1: Exception Flow) (File Not Found)**

2	<p><b><u>Test Function:</u></b> Video Asset Not Found in Watch Folder</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Delete the video asset or rename the video asset in the Watch folder</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Dashboard will reflect <u>Video Asset not found in Watch Folder</u></li> <li>• Email with video assets details and error will be sent out</li> </ul>		
---	---	--	--

**Module: AUSTIN Service – Publish Service (Scenario 1: Normal Flow)**

1	<p><b><u>Test Function:</u></b> After encoded is completed (file not in use in output folder), encoded asset and its XML file is pushed to FTP folders based on Profile settings</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. After encoding is completed, observe the output folder</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Dashboard to reflect <b>Sent to FTP Folder</b></li> <li>• Video asset meta data generated</li> <li>• Encoded video asset and metadata pushed to the respective FTP folders based on the profile settings</li> </ul>		
2	<p><b><u>Test Function:</u></b> Verify meta data of video asset (XML) is created in the respective output folders</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. After encoding is completed, observe the output folder</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Video asset metadata is generated</li> <li>• Verify metadata is correct and accurate – similar to video asset</li> </ul>		
3	<p><b><u>Test Function:</u></b> Verify meta data of video asset (XML) generated is based on the configuration file (another XML) in the outputfolder</p> <p><b><u>Test Procedure:</u></b></p> <ol style="list-style-type: none"> <li>1. Edit the output configuration file to show and hide some fields</li> <li>2. Set “true” to “false”</li> </ol> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"> <li>• Verify video asset metadata is consistent with output configuration file</li> </ul>		

Module: AUSTIN Service – Publish Service (**Scenario 2: Exception Flow**) (File Not Found)

2	<p><b><u>Test Function:</u></b> Video Asset Not Found in Output Folder</p> <p><b><u>Test Procedure:</u></b></p> <p>2. Delete the encoded video asset or rename the video asset in the Output folder</p> <p><b><u>Expected Results:</u></b></p> <ul style="list-style-type: none"><li>• Dashboard will reflect <u>Video Asset not found in Output Folder</u></li><li>• Email with video assets details and error will be sent out</li></ul>		
---	--	--	--

Once you have completed the above test cases, the AUSTIN Web Portal/AUSTIN Service user testing is completed.

.....End of Test Cases.....

**Survey Questions**

1. How would you rate the ease of use of the AUSTIN Web Portal?

1	2	3	4
Very Good	Good	Bad	Very Bad

2. How would you rate the overall look (User Interface) of the AUSTIN Web Portal?

1	2	3	4
Very Good	Good	Bad	Very Bad

3. Was it easy/fast to locate the functions on the AUSTIN Web Portal?

1	2	3	4
Very Easy	Relatively Easy	A bit difficult	Very Difficult

4. Was it easy to understand where you are in the page?

1	2	3	4
Very Easy	Relatively Easy	A bit difficult	Very Difficult

5. If you have encountered errors, were the error messages easy to understand?

1	2	3	4
Very Easy	Relatively Easy	A bit difficult	Very Difficult

6. Are all the functions of the web portal useful?

Create Video

Metadata

1 Very Good	2 Good	3 Bad	4 Very Bad
----------------	-----------	----------	---------------

Dashboard

1 Very Good	2 Good	3 Bad	4 Very Bad
----------------	-----------	----------	---------------

Manage Settings

1 Very Good	2 Good	3 Bad	4 Very Bad
----------------	-----------	----------	---------------

Any suggestions on how can it be improved?

User Interface can be improved very much!

Name:

Company:

Signature: