Quality Assurance

Table of Contents

[High Level Design 4](#_Toc318877399)

[High Level Design Checklist 5](#_Toc318877400)

[Instructional Design 6](#_Toc318877401)

[Instructional Design Checklist 7](#_Toc318877402)

[Instructional Design - Module Review 7](#_Toc318877403)

[Instructional Design - Module Review Checklist 7](#_Toc318877404)

[Instructional Design - Feedback from client 8](#_Toc318877405)

[Instructional Design - Module Review Checklist 8](#_Toc318877406)

[Programming - Module Build 9](#_Toc318877407)

[Programming - Module Build Checklist 10](#_Toc318877408)

[Graphics - Module Build 11](#_Toc318877409)

[Graphics - Module Build Checklist 11](#_Toc318877410)

[Quality Assurance - Module Review 12](#_Toc318877411)

[Quality Assurance - Module Review Checklist 12](#_Toc318877412)

[Savv-e Express Interface Checklist 13](#_Toc318877413)

[GLOBAL 13](#_Toc318877414)

[Standard Course structure 13](#_Toc318877415)

[FlyingStart structure 13](#_Toc318877416)

[Situational structure 13](#_Toc318877417)

[Adobe Captivate 14](#_Toc318877418)

[General 14](#_Toc318877419)

[Standard requirements 14](#_Toc318877420)

[Accessibility best practices 14](#_Toc318877421)

[Adobe Flash 17](#_Toc318877422)

[Accessibility best practices 17](#_Toc318877423)

[Web Design Best Practices Checklist 20](#_Toc318877424)

[Background Information 20](#_Toc318877425)

[Page Layout 20](#_Toc318877426)

[Browser Compatibility 20](#_Toc318877427)

[Navigation 20](#_Toc318877428)

[Color and Graphics 21](#_Toc318877429)

[Multimedia 21](#_Toc318877430)

[Content Presentation 21](#_Toc318877431)

[Functionality 21](#_Toc318877432)

[Accessibility 22](#_Toc318877433)

[Common Issues 23](#_Toc318877434)

[How to... 23](#_Toc318877435)

[Fix Notepad close button 23](#_Toc318877436)

[Flash jagged images 26](#_Toc318877437)

[Audio / Video 27](#_Toc318877438)

[Reference 29](#_Toc318877439)

## High Level Design

In the High Level Design it is essential that you ensure that the following points have been actioned and checked.

By applying these points it ensures that the document meets Savv-e’s standards.

1. Read and familiarise any client vendor information package which outlines communication, branding and formalised elearning requirements (e.g NAB / Westpac)
2. Copying Existing High Level Design
   * If using an existing High Level Design used for another similar project, ensure that all information from previous project is removed.
3. Check learning approach with the team (Should not exceed the proposed budget)
4. Course outline doesn’t exceed the scope
5. Technical specifications provided by the client are ok and accurate
   * Target audience, duration, module name, objectives and learning outcomes have been entered and are as relevant and possible.
6. Brainstorming - discuss with team for input, innovative ideas and development considerations.
7. Navigation – specified as agreed with the client.
8. Project and client details are correct
   * Ensure that all information is correct such as Company name, Client Contact.
9. Headers and Footers are correct
   * Correct address is used (not 100 Alexander Street)
10. Table of contents has been created / updated
    * Correct headings link to correct page (CTRL + SHIFT)
11. Formatting such as spelling, grammar, incorrect doubles spacing
12. Version control
    * For every edit that is made to the document, a new entry should be entered to ensure people are aware who worked on it last and this also allows us to find and backtrack information.

## High Level Design Checklist

|  |  |
| --- | --- |
| **🞎** | I have read and familiarised myself with the client/vendor information package |
| **🞎** | I have used an existing High Level Design used for another similar project, and I have removed all references to the previous project |
| **🞎** | I have check the learning approach with the team and it doesn’t exceed the proposed budget |
| **🞎** | I have checked the course outline and it doesn’t exceed scope |
| **🞎** | I have checked the technical specifications provided by the client are ok and accurate |
| **🞎** | I have discussed with the team and received appropriate information |
| **🞎** | I have specified the navigation as per the requirements of the client |
| **🞎** | I have checked that the project and client details are correct |
| **🞎** | I have checked that the Headers and Footers are correct |
| **🞎** | I have checked that the table of contents has been created / updated |
| **🞎** | I have checked that formatting such as spelling, grammar are correct |
| **🞎** | I have checked that the document version is up to date and is correct |

## Instructional Design

1. Read and familiarise any client vendor information package which outlines communication, branding and formalised elearning requirements (e.g NAB / Westpac)
2. Project name and client details are correct
   * If using an existing Script used for another similar project, ensure that all information from previous project is removed.
3. Headers and Footers are correct
4. Table of contents has been created / updated
   * This should be updated everytime to ensure the contents page matches up with the actual page.
5. Formatting such as spelling, grammar, incorrect doubles spacing
6. Bullet formatting and use of punctuation is consistent
   * This should be based on either the clients branding or preference. If the client has no preference, at Savv-e we use open punctuation.
7. Prompts – ensure they are clear and consistent throughout
   * Avoid scripting prompts that have positions such as "Click the buttons on the right to learn more about...", instead use "Click the buttons to learn more about..." as the positioning of the buttons can move if the graphics designer feels the screen etc looks / feels wrong
8. Include a resources page which summarises all links (doc, pdf and url)
9. Script page ID reference should auto increment
10. Audio/Voiceover script to include accurate screen references
11. Audio/Video script should not have and that sentence doesn’t go over 2 pages
    * This makes it hard when an actor is reading and has to flip pages on shoot
12. Audio/Video will be delivered in the correct format and quality
    * Video – MOV / AVI or MPEG
    * Stereo – Stereo / 44100 Hz / Constant Bitrate of 128 kbit/s
13. Description of visuals for graphics is clear. Not necessarily the actual image/video unless specified by client
14. Clear correct answer options for all questions in module and assessment
    * Ensure that the Correct Option is explained easily so that it can be programmed correctly.
15. Pools – Assessments questions should be in order ( Pool 1 - Q1, Q2) (Pool 2 – Q1, Q2), Not (Q1A, Q1B, Q2A, Q2B)
16. Assessment – Pass/Fail mark
17. Assessment Correct / Wrong Answer Feedback layout – The first line should be Correct / Incorrect, followed by the reason.
18. Assessment Next prompt – should say or be similar to: Click Next to continue
19. Assessment End prompt – Last prompt of assessment should say or be similar to: Click Next to see your results, NOT click next to continue
20. Global programming to be clear and accurate
21. Certificate – If required provide correct wording/layout and required fields.
22. Version control
    * This should be updated once the script has had a major update/review.

## Instructional Design Checklist

|  |  |
| --- | --- |
| **🞎** | I have read and familiarised myself with the client/vendor information package |
| **🞎** | I have checked that the project name and client details are correct |
| **🞎** | I have checked that the Headers and Footers are correct |
| **🞎** | I have checked that the table of contents is updated and correct |
| **🞎** | I have checked that formatting such as spelling, grammar are correct |
| **🞎** | I have checked that the bullet formatting and punctuation is consistent throughout the script |
| **🞎** | I have checked that the prompts are consistent throughout the script |
| **🞎** | I have checked that the resources page is up to date and links to source files |
| **🞎** | I have checked that page IDs are correct |
| **🞎** | I have checked that the Voiceover script is correct and that sentence doesn’t go over 2 pages |
| **🞎** | I have checked that the Video script is correct and that sentence doesn’t go over 2 pages |
| **🞎** | I have checked that the audio/video will be delivered in the correct specifications |
| **🞎** | I have checked that the images required have a clear description |
| **🞎** | I have checked that the Correct and Wrong Answer options are clearly displayed |
| **🞎** | I have organised the assessment into Pools. |
| **🞎** | I have formatted the Correct and Wrong answer feedback |
| **🞎** | I have checked that the assessment prompts are consistent throughout the script |
| **🞎** | I have checked that the assessment end prompt is along the lines of “Click Next to see your results” |
| **🞎** | I have checked that the global programming is clear |
| **🞎** | I have added the correct wording/layout and required fields for the certificate. |
| **🞎** | I have checked that the document version is up to date and is correct |

## Instructional Design - Module Review

1. Screen layout - text /image population looks correct and good
   * Example some screens may appear blank until a button is clicked, therefore possibility client will like another option or another template should be used.
2. Visuals, Navigation and functionality are as per script and clients expectations.
3. If changes are required, script must be updated by creating a new version.
4. Volume is consistent (voiceover, video etc are all on the same level)
5. Resources have been added and are correct
6. Help has been built

## Instructional Design - Module Review Checklist

|  |  |
| --- | --- |
| **🞎** | I have checked the screen layout and content |
| **🞎** | I have checked the visual, navigation and functionality as per requirements. |
| **🞎** | I have made some updates to some screens and have also updated the script |
| **🞎** | I have listened to the column in audio and video and its consistent through the course |
| **🞎** | I have checked that the resources are correct |
| **🞎** | I have checked that the help screen is correct |

## Instructional Design - Feedback from client

1. Review all feedback and determine:
   * If there is a scope changes
   * If feedback requires further clarification
   * Feedback can’t be implemented e.ge due to limitations of Savv-e Express.
2. Raise all relevant feedback in to Jira and assign to appropriate team member.
3. Update script as required.
4. Once feedback has been implemented, check to see if it has been actioned correctly.

## Instructional Design - Module Review Checklist

|  |  |
| --- | --- |
| **🞎** | I have reviewed all feedback |
| **🞎** | I have raised all relevant feedback and assigned to appropriate team member. |
| **🞎** | I have updated the script where relevant and have tracked changes on. |
| **🞎** | I have checked all edits and they have been actioned accordingly. |

## Programming - Module Build

1. Read and familiarise any client vendor information package which outlines communication, branding and formalised elearning requirements (e.g NAB / Westpac)
2. Create required files for scorm package
   * Modulename.html, begin-popup.htm, end-popup.htm and imanifest.xml
3. Update HTML and XML to have correct project name
4. Navigation
   * Remove/Hide "Next >" on Home Screen
   * Remove/Hide "Next >" on Final Screen
5. Breadcrumb
   * Ensure breadcrumb trail
     + Topic - Trail to show which screen they are on
     + Assessment - Question X of XX
6. Active topic
   * Active topic to be visible and Make Active Topic menu highlighted
7. Topic ticking / completion
   * Ensure each screen is marked as visited with a tick and tick the topic once all pages visited
8. Module completion
   * Once all topics are complete, Completion message should be displayed and/or the Print Certificate
9. Tracking of Data
   * The module can save and exit and on resume starts from last screen visited or home
   * Once module complete, completion status sent to LMS (IE)
10. Links
    * Clicking a link doesn’t close the module
    * Clicking a PDF link will open at 100% - filename.pdf#zoom=100
11. Transcript
    * Remove scroll bar if text fits in transcript/audio player
12. Global
    * Disable buttons on the screen behind for example click help, it loads over main interface, thus users should not be able to click on Save & Exit etc
    * Disable buttons on the screen behind
    * If audio/video is playing and click on resources or help, pause video/audio
    * Question and Answer
13. Question and Answer
    * The ‘Done’ button not activate until at least one answer has been selected.
14. Assessment
    * All questions should randomise unless specified
      + If a Question has the option – ALL OF THE ABOVE, this should not randomise and should be the last option on screen.
    * Pass – on pass correct message is displayed
    * Fail – on fail correct message is displayed
    * Re-attempt – If assessment has X amount of re-attempts, learner should be blocked out based on rules specified in script.
15. Edits
    * Once you have actioned and edit/bug/text update you will need to do some self QA to ensure that it has been done.

## Programming - Module Build Checklist

|  |  |
| --- | --- |
| **🞎** | I have read and familiarised myself with the client/vendor information package |
| **🞎** | I have checked that the copied files don’t have any reference to the previous project |
| **🞎** | I have created/added all required scorm files |
| **🞎** | I have updated all HTM / XML files with project details |
| **🞎** | I have applied navigation rules |
| **🞎** | I have checked breadcrumb location and style |
| **🞎** | I have checked that the active topic is highlighted in menu |
| **🞎** | I have checked that topics mark with TICK or PAUSE based on completion status |
| **🞎** | I have checked that module resumes from correct screen exited from |
| **🞎** | I have checked that module marks as complete once completion rule is met |
| **🞎** | I have checked that completion status and score if applicable is saved on the LMS |
| **🞎** | I have checked that links don’t force the module to close down on click |
| **🞎** | I have checked that PDF links open at 100%, not less or higher |
| **🞎** | I have checked that transcripts don’t have scroll bars where scroll bars are not required |
| **🞎** | I have disabled active buttons on the background screen when a popup appears |
| **🞎** | I have checked that audio/video pauses on screen popup such as Help/Resources |
| **🞎** | I have checked that audio/video doesn’t play on screen resume |
| **🞎** | I have checked that buttons are not active in a question until a selection is made |
| **🞎** | I have checked that assessment NEXT is not active until a question is answered |
| **🞎** | I have checked that assessment question randomise unless specified |
| **🞎** | I have disabled randomisation if a question has the option – All of the above |
| **🞎** | I have checked that correct answer feedback (CAF) is displayed when correct option(s) are selected |
| **🞎** | I have checked that wrong answer feedback (WAF) is displayed when wrong option(s) are selected |
| **🞎** | I have checked that if the assessment has a lockout or re-attempt amounts that it is coded as per the requirements |
| **🞎** | I have checked that all edits (bugs/scope change) have been actioned and checked before assigning to relevant team member |

## Graphics - Module Build

1. Read and familiarise any client vendor information package which outlines communication, branding and formalised elearning requirements (e.g NAB / Westpac)
2. If copying source files from another projects, ensure that other project **DATA** is removed
   * Ensure that client name or another module name doesn’t appear in all FLA/SWF.
3. Guide main interface from individual SWF files
   * This will ensure a smaller file, therefore optimising end package size.
4. Apply formatting to screens (Bold, Italic, Underline)
   * When building screens, ensure that the formatting is applied. Not just copy and paste, as copy and paste from Word to Flash loses the formatting.
5. Create and maintain a clean graphics folder as particular clients E.g. NAB will want to receive all files (RAW and Processed)
6. Optimise file size
   * Savv-e Express – JPG – save file at 80%
   * Custom – Publish output to 80%
7. Make images smooth in Flash – Tick allow smoothing
8. Naming files (assets)
   * Images, Video, SWF, Audio etc should be named correctly. Example CLIENT\_PROJECTNAME\_SCREENID / NAB-Banking\_1-1-2
9. Buttons
   * Buttons must have an up, over, selected and disabled state
10. Edits
    * Once you have actioned and edit/bug/text update you will need to do some self QA to ensure that it has been done.

## Graphics - Module Build Checklist

|  |  |
| --- | --- |
| **🞎** | I have read and familiarised myself with the client/vendor information package |
| **🞎** | I have checked that the copied files don’t have any reference to the previous project |
| **🞎** | I have guided the main interface from each individual SWFs for file optimisation |
| **🞎** | I have applied the formatting as per the script |
| **🞎** | I have created and maintain a clean graphics folder as some clients will require source files |
| **🞎** | I have applied the optimal image settings for file optimisation |
| **🞎** | I have applied the publish output settings for individual FLAs for file optimisation |
| **🞎** | I have ticked Allow smoothing on images in Flash |
| **🞎** | I have followed a standard naming convention for all assets |
| **🞎** | I have applied the correct video settings for optimal quality and file size |
| **🞎** | I have applied the correct audio settings for optimal quality and file size |
| **🞎** | I have created all buttons with an up, over, selected and disabled state |
| **🞎** | I have checked that all edits (bugs/scope change) have been actioned and checked before assigning to relevant team member |

## Quality Assurance - Module Review

1. Read and familiarise any client vendor information package which outlines communication, branding and formalised elearning requirements (e.g NAB / Westpac)
2. Locate and open proposal to obtain clients system specifications.
   * Screen resolution
   * Operating system
   * Plugins (Flash version)
3. Script to screen (ensure track changes are on)
   * Compare the screen to what is in the script. Any differences should be raised into Jira and assigned to the relevant team member.
4. Screen layout - text /image population looks correct and good
   * Example some screens may appear blank until a button is clicked, therefore possibility client will like another option.
5. Volume is consistent (voiceover, video etc are all on the same level)
   * Audio / video should all be at the same level so that no drastic change in volume occurs when you go from page to page.
6. Resources have been added and are correct
   * Resources in the script should be included in the module build. If for some reason resources have not been provided from the client, a placeholder should be added to ensure easy insertion down the track.
7. Help has been built and is correct.
8. Correct deliverable format (SCORM (LMS) / CD)
   * The correct delivery method has been programmed and tested (LMS, CD)
9. Test in Reload Scorm Player or on Clients Test LMS account as long as it has been provided
   * CBA - ASK-ASK and Cornerstone - SCORM
   * Westpac – Sandpit – SCORM (Completed Checklist)
   * St. George – eLuminate - SCORM
10. Package the module with all relevant files
    * LMS – include relevant files for example begin-popup.htm, main.htm, end-popup.htm, imanifest.xml etc.
    * CD – Should include autorun.inf, modulename.exe
11. Naming and Saving deliverable
    * The naming of the deliverable should follow the following naming convention:
      + YYYY-MM-DD-Client-Module\_name-TYPE-Delivery\_release.zip
      + 2012-02-26-NAB-Fraud\_awareness-LMS-FINAL.zip
    * The deliverable should be saved to 3 locations
      + Projects / Current / Client / Project Name / Dev / Delivery folder
      + Projects / Delivered / YYYY-MM-DD-Client-Module\_name-TYPE-Delivery\_release
      + Savv-e.net / Client / YYYY-MM-DD-Client-Module\_name-TYPE-Delivery\_release.zip
12. Creating and Sending deliverable email
    * When sending a client a deliverable we have a standard email that has all relevant data. Default text and 2 attachments. These have been created as these are the common questions client ask and by having the default text and attachments it has minimised support on extracting zip files etc.
    * Email should include:
      + Default text
      + 2 attachments
        - Extracting Package files.pdf
        - Clearing Your Browser’s Cache – All Browsers.pdf
        - Link to deliverable on savv-e.net
13. Send to all relevant client contacts and Savv-e team members.

## Quality Assurance - Module Review Checklist

|  |  |
| --- | --- |
| **🞎** | I have read and familiarised myself with the client/vendor information package |
| **🞎** | I have located the required documents to obtain the clients system specifications and will test the module based on these requirements |
| **🞎** | I have checked script to screen for each individual screen |
| **🞎** | I have checked the screen layout and content |
| **🞎** | I have listened to the audio and video and its consistent through the course |
| **🞎** | I have checked that all resources have been added |
| **🞎** | I have checked that resources is disabled if not resources are available |
| **🞎** | I have checked that help has been built and a correct image has been used |
| **🞎** | I have checked individual file sizes for optimal loading time |
| **🞎** | I have checked that all edits (bugs/scope change) have been actioned and closed |
| **🞎** | I have tested the module (if scorm) in reload to check scorm functionality (tracking/bookmarking/status and score) |
| **🞎** | I have tested the module on the clients LMS to check functionality (tracking/bookmarking/status and score) |
| **🞎** | I have packed the deliverable in the correct delivery format (SCORM / CD etc) |
| **🞎** | I have named and saved the deliverable to all locations |
| **🞎** | I have prepared the email with all relevant information |
| **🞎** | I have sent the deliverable to all relevant client contacts and Savv-e team members |

# 

# Savv-e Express Interface Checklist

## GLOBAL

### Standard Course structure

Savv-e Express has a default total of 73 templates. All these templates need to be skinned to ensure that the standard templates structure works.

* All default templates have been skinned
  + 4 X Homepage screens
  + 69 X Standard screens
* Create Help
  + Help should be updated with new look and feel
* Menu / Resources Menu
  + They should be consistent with the look and feel
* Print Certificate
  + If the Interface is a custom interface, please hardcode the client logo. This will ensure a crisp clear logo.
* Notepad close button should NOT match interface
  + It should be a generic X button

### Standard Course structure checklist

Once the standard course structure has been skinned for a new interface you will need to:

1. Publish to LMS (Scorm with POPUP and without POPUP)
2. Publish to CD

### FlyingStart structure

FlyingStart has a default total of 25 templates. All these templates need to be skinned to ensure that the standard FlyingStart structure works.

### Situational structure

Situational has a default total of 4 templates. All these templates need to be skinned to ensure that the standard Situational structure works.

# Adobe Captivate

## General

### Standard requirements

Captivate requires a few tweaks to make it meet the clients look and feel and to ensure a positive learner experience.

* End of simulation should NOT fade out
* End of simulation it should include a text popup box informing learner to click X to close the simulation
* Play bar interface should match clients look and feel
* If the simulation has a Closed Captioning, convert text to speech to ensure correct timing

### Accessibility best practices

Captivate allows you to present content to users with disabilities.

#### Adding a description to an Adobe Captivate movie

Every movie created in Adobe Captivate should include a movie description. For demonstrations that are short or that rely heavily on the mouse and are thus irrelevant to screen reader users, the movie description alone will suffice. Longer movies should provide a description including a content overview and context for the user, along with individual slide descriptions where necessary.

Follow these steps to add a description to an Adobe Captivate movie:

* Open the movie.
* Choose Edit > Preferences.
* Choose General from the list of categories.
* Type your description in the Description field.
* Click OK.

The following Adobe Captivate demonstration illustrates how to provide a description for an entire movie.

[Play the demo: Adding a description to an Adobe Captivate movie](http://www.adobe.com/accessibility/products/captivate/moviedescv05_demo.html)

#### Adding a description to an individual slide

An individual slide should include a description if it is an important step in the process or if it presents a significant concept visually. Some movies, particularly those of recorded demonstrations, may not need descriptions for each individual slide because often several slides cover one concept. In these cases, place the description on the last slide of that group.

When providing a description for an individual slide, it is important to pause the movie on that slide. This provides the user the opportunity to return to the top of the movie and read the description. Otherwise, the movie plays and moves past the slide containing the description.

To make it easier for a screen reader user to understand when a movie is paused, it is recommended that an audio cue, such as a tone, be associated with the Continue button.

To add a description to a slide, use the following steps:

Open the movie and select the appropriate slide.

* Choose Slide > Properties.
* Click the Accessibility button.
* Type the slide description in the Slide Accessibility Text dialog box.
* Click OK.

To add a pause button to a slide, use the following steps:

* Choose Insert > Button.
* Type "Continue" or other appropriate label as Button Text.
* Click the Select Keys button.
* Select the Attach a Shortcut option.
* Select Enter as shortcut key.
* Click OK.
* Select the audio tab.
* Click the Import button.
* Select Tone.mp3.
* Click Open.
* Click Apply.
* Click OK.
* Position the button on your screen.

The following Adobe Captivate demonstration illustrates how to add a description to an individual slide.

[Play the demo: Adding a description to an individual slide](http://www.adobe.com/accessibility/products/captivate/slide-desc-2_demo.html)

#### Adding quizzes

Adobe Captivate includes a set of accessible question types including multiple choice, true and false, and Likert scale. To ensure that your quiz is accessible, limit questions to these types. The short answer and matching question types are not accessible.

#### Providing equivalents for audio content

For people who are deaf or hard of hearing, the audio content of an Adobe Captivate movie is not accessible. To provide access to this information, place a transcript of the audio content in a caption on each screen.

The following Adobe Captivate demonstration illustrates how to add closed captions to an Adobe Captivate movie.

[Play the demo: Adding closed captions to an Adobe Captivate movie](http://www.adobe.com/accessibility/products/captivate/Adobe_Captivate_cc_tutorial_demo.html)

#### Adding click areas

Adding click areas to your movie is an easy way to create simulations that make Adobe Captivate movies much more than simple demonstrations. However, for people who have difficulty using the mouse, click areas are not meaningful ways of interacting with content. To allow people who rely on the keyboard to navigate content to access simulations containing click areas, these objects have been made keyboard accessible. With the 508 compliance option selected (which is enabled by default), click areas are accessible by default. In addition, they have a text equivalent for screen reader users to let them know that the button is a click area. Moving to the click area and pressing Enter will allow keyboard users to advance through a simulation even if they cannot use the mouse.

# Adobe Flash

### Accessibility best practices

Adobe Flash CS5.5 allows designers and developers to create powerful web applications that are exciting, engaging, and accessible to all.

The following are some practical suggestions for optimizing the accessibility of your rich media content.

#### Assign text equivalents for visual elements

Provide text equivalents for graphic elements in Flash Professional CS5.5. Provide names for graphic icons. Add text equivalents for gesturing animations that highlight an area of the page. When you use a feature such as Break Apart for text, be sure to provide a name or description. When a group of related graphic elements are used to convey a single idea, provide a single text equivalent and make the child objects inaccessible. Learn more about [providing text equivalents](http://www.adobe.com/accessibility/products/flash/text.html).

#### Animation

##### Make looping elements inaccessible

Movies that never stop moving cause screen readers to refresh frequently. Even in cases where the movies are at the bottom of a page, the screen reader can interpret motion as an update to the page and return to the top and start reading again. For this reason, child objects of movie clips or entire movies should be made inaccessible. Learn more about [handling animation](http://www.adobe.com/accessibility/products/flash/animation.html).

##### Allow users to control motion

Try not to present information in your movie that remains on the screen for only a short time. Screen readers may have a difficult time keeping up with quick changes in movies. You can resolve this type of problem by adding Next buttons that control movement. Learn more about [handling animation.](http://www.adobe.com/accessibility/products/flash/animation.html)

#### Use accessible components

A core set of accessible user interface components is provided in Flash to help you with accessible application development. These components can automate many of the most common accessibility practices related to labeling, keyboard access, and testing and help ensure a consistent user experience across rich applications. For each component, the designer or developer only needs to enable the accessibility object by using the command enableAccessibility(). This includes the accessibility object with the component as the movie is compiled. Learn more about [components and accessibility.](http://www.adobe.com/accessibility/products/flash/components.html)

#### Enable control over reading order

The default reading order of a movie created with Flash may not follow the same order that the designer would expect based on the visual layout. There are at least three means of controlling reading order. First, the designer or developer can limit the physical size of the stage and keep the layout simple. Second, the designer or developer can develop a secondary control that places a linear version of the content offstage. Third, the reading order can be specified using ActionScript®. Learn more about [controlling reading order](http://www.adobe.com/accessibility/products/flash/reading.html) in Flash.

#### Facilitate keyboard access to all controls

When adding buttons and other controls to movies, make sure that users can navigate through your movie effectively using only the keyboard. Keep in mind that applications need to be tested both with and without a screen reader. To facilitate keyboard access, try to keep scripts within frames as opposed to attaching them directly to objects. Also, avoid using empty movie clips as buttons. These "hit areas" are not recognized by screen readers. Finally, add keyboard shortcuts to commonly used buttons to promote access. Learn more about [facilitating keyboard access](http://www.adobe.com/accessibility/products/flash/keyboarding.html).

#### Provide captions

Flash Professional CS5.5 makes delivering audio content simple, and now delivering closed captions for users who are deaf or hard of hearing is easier than ever. Flash Professional CS5.5 offers a new component to display captions that are either contained in a World Wide Web Consortium (W3C) Timed Text XML file (TTML) or integrated with the FLV file as cue points. Several tools and professional services are available to assist Flash developers in creating or obtaining a TTML caption data file for a video. Learn more about [captions](http://www.adobe.com/accessibility/products/flash/captioning_tools.html).

#### Provide accessible video controls

Video playback controls need to be accessible for blind, low-vision, and keyboard-only users. Flash Professional CS5.5 provides built-in support for accessibility video player skins. Learn more about [accessible video controls](http://www.adobe.com/accessibility/products/flash/video.html) in Flash.

#### Enable control over audio playback

Music and audio that plays as the site loads presents a serious challenge to screen reader users. The audio from a movie can interfere with the end user's ability to hear the contents of a movie using a screen reader. As a result, it is important to make sure that the user has control over when music is played. The simplest strategy for handling audio playback is simply to allow the end user to control audio with a play and pause button. Learn more about how to [enable control over audio](http://www.adobe.com/accessibility/products/flash/audio.html).

#### Expose structure

Movies created with Flash can be complex in terms of layout, structure, and navigation. As a result, it can be very difficult for screen reader users to make sense of such a movie. As sites become more complex, try to add a description for the entire movie to help orient screen reader users to the structure of the site. This can be accomplished using either the root-level description or a separate information screen. The advantage of building a separate information screen is that it allows the user to go to the content without hearing the description at each refresh.

#### Expose state of controls

Flash allows an infinite variety of controls. For all controls, it is important to provide the user with feedback on the control as it changes. In the simple example below, notice that once the button is pressed it changes from a play to a pause button. As the state of the button changes, the accessibility information for this button should be updated as well.

#### Use color wisely

Flash allows designers to use a wide variety of color combinations. When making color choices for a movie, the designer should not rely on color alone to convey information. For example, it would not be appropriate to provide an instruction that reads, "Click the green button to go forward and the red button to go back." At the same time, it is important to make sure that there is sufficient contrast between foreground and background colors to make content easily readable.

#### Validate for accessibility

Accessibility guidelines provide only limited guidance for the design of accessible content. Many of the most common issues in accessible design are not reflected in accessibility standards such as Section 508 or the W3C guidelines. Flash designers and developers should rely on a variety of methods for validation of content for accessibility. First, try accessing your content using screen access technologies such as Window-Eyes from GW Micro. This will provide the best insight into the usability of a movie for screen users. Also, be sure to test your site using only the keyboard without a screen reader running. Keyboard access differs when a screen reader is not present. Second, use third-party repair and validation tools for Flash. These tools can help to identify common errors in design. Third, be sure to test the site with actual users. It is very important that before a site is deemed accessible it is evaluated either formally or informally by people with disabilities.

# Web Design Best Practices Checklist

## Background Information

* URL:
* Audience:
* Subject:
* Developer:

## Page Layout

1. Appealing to target audience
2. Consistent site header/logo
3. Consistent navigation area
4. Informative page title that includes the company/organization/site name
5. Page footer area — copyright, last update, contact e-mail address
6. Good use of basic design principles: repetition, contrast, proximity, and alignment
7. Displays without horizontal scrolling at 1024x768 and higher resolutions
8. Balance of text/graphics/white space on page
9. Good contrast between text and background
10. Repetitive information (header/logo and navigation) takes up no more than one-quarter to one-third of the top portion of the browser window at 1024x768 resolution
11. Home page has compelling, interesting information above the fold (before scrolling down) at 1024x768
12. Home page downloads within 10 seconds on dial-up connection

## Browser Compatibility

1. Displays on popular versions of Internet Explorer (8+)
2. Displays on popular versions of Firefox (5+)
3. Displays on popular versions of Opera (10+)
4. Display on popular versions of Safari (both Mac and Windows)

## Navigation

1. Main navigation links are clearly and consistently labeled
2. Navigation is easy to use for target audience
3. If images or Flash is the main navigation, clear text links are in the footer section of the page (accessibility)
4. Navigation is structured in an unordered list (accessibility)
5. Navigation aids, such as site map, skip navigation link, or breadcrumbs are used (accessibility)
6. All navigation hyperlinks "work" — are not broken

## Color and Graphics

1. Use of different colors in page backgrounds/text is limited to a maximum of three or four colors plus neutrals
2. Color is used consistently
3. Color has good contrast with associated text
4. Color is not used alone to convey meaning (accessibility)
5. Use of color and graphics enhances rather than distracts from the site
6. Graphics are optimized and do not significantly slow download
7. Each graphic used serves a clear purpose
8. Image tags use the alt attribute to configure alternate text to display if the browser or user agent does not support images (accessibility)
9. Animated images do not distract from the site and either do not repeat or only repeat a few times

## Multimedia

1. Each audio/video/Flash file used serves a clear purpose
2. The audio/video/Flash files used enhance rather than distract from the site
3. Captions are provided for each audio or video file used (accessibility)
4. Download times for audio or video files are indicated
5. Links to downloads for media plug-ins are provided

## Content Presentation

1. Common fonts such as Arial or Times New Roman are used OR no more than one web font used
2. Techniques of writing for the Web are used: headings, bullet points, short sentences in short paragraphs, use of white space, etc.
3. Fonts, font sizes, and font colors are consistently used
4. Content provides meaningful, useful information
5. Content is organized in a consistent manner
6. Information is easy to find (minimal clicks)
7. Timeliness: The date of the last revision and/or copyright date is accurate
8. Content does not include outdated material
9. Content is free of typographical and grammatical errors
10. Content provides links to other useful sites
11. Avoids the use of "Click here" when writing text for hyperlinks
12. If standard link colors are not used, hyperlinks use a consistent set of colors to indicate visited/nonvisited status
13. If graphics and/or media is used to convey meaning, the alternate text equivalent of the content is provided (accessibility)

## Functionality

1. All internal hyperlinks work
2. All external hyperlinks work
3. All forms function as expected
4. No JavaScript errors are generated

## Accessibility

1. If images or Flash is the main navigation, clear text links are in the footer section of the page
2. Navigation is structured in an unordered list
3. Navigation aids, such as site map, skip navigation link, or breadcrumbs are used
4. Color is not used alone to convey meaning
5. Image tags use the alt attribute to configure alternate text to display if the browser or user agent does not support images
6. If graphics and/or media is used to convey meaning, the alternate text equivalent of the content is provided
7. Captions are provided for each audio or video file used
8. Use attributes designed to improve accessibility such as title and summary when appropriate
9. Use the id and headers attributes to improve the accessibility of table data
10. If the site uses frames, use frame titles and place meaningful content in the noframes area
11. To assist screen readers configure the html element's lang and xml:lang attribute to indicate the spoken language of the page.

# Common Issues

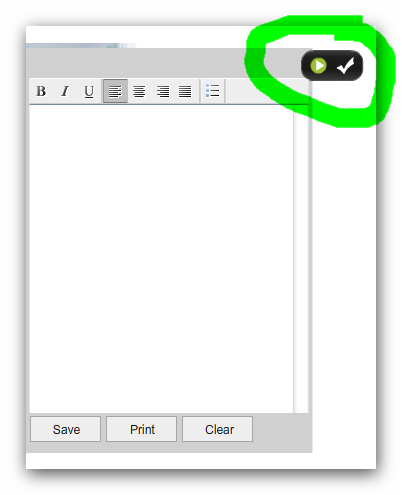
## How to...

### Fix Notepad close button

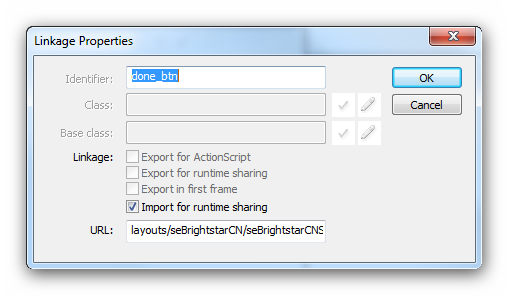
The Notepad close but is generic and doesn’t need to be skinned.

Sometimes they are skinned and the steps to fix it are below:

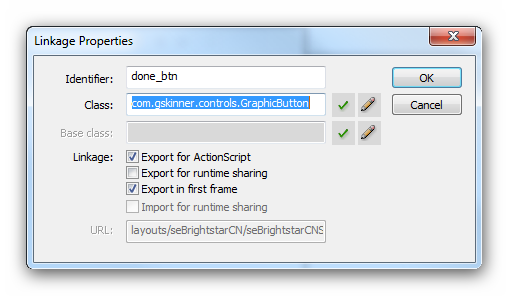
1. Image 1.0 shows the issue where the X (Close) button is skinned and needs to be removed.



1. Open the maininterface.fla for this interface and find **done\_btn** in the library, then Right click it and select Linkage... and you will see the screen similar to the one below.

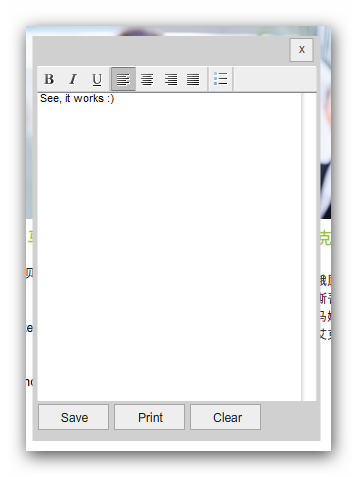


1. Now uncheck the **Import for runtime sharing**
2. Then tick **Export for ActionScript** and you will see the screen similar to the one below.



1. Save the maininterface.fla file and publish it.

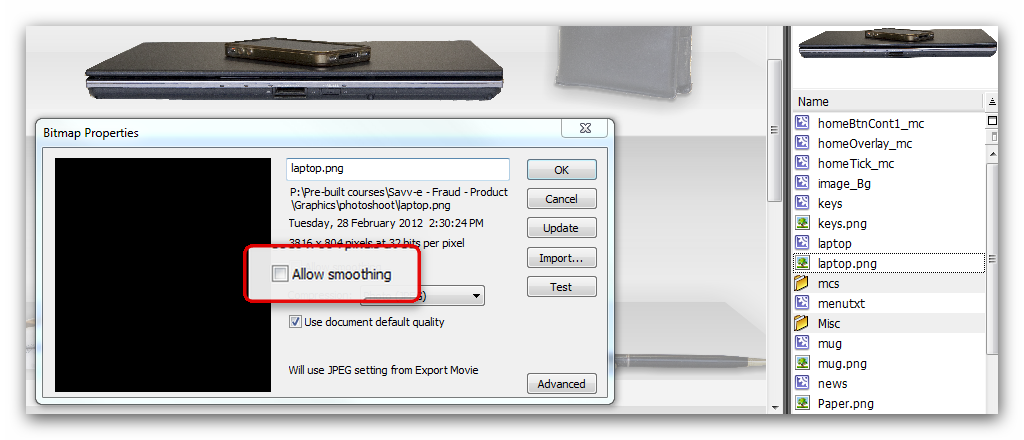
**NOTE:** Remember to clear your cache before previewing the maininterface and you will see the screen similar to the one below.



### Flash jagged images

When you import into Flash a large image and use flash to resize it, when the SWF is published it looks jagged. To fix this you need to **TICK** Allow smoothing on that image. To do this you will need to:

* Find image in library
* Right Click and select Properties
* Tick Allow smoothing



### Audio / Video

When building screens which require Audio or Video or both, please ensure that the levels match and the follow the following standard.

#### Savv-e Express

* + MP3

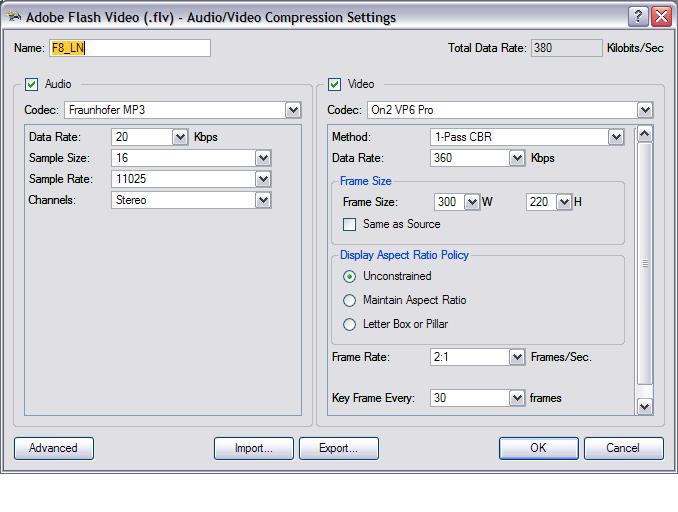
When converting WAV to MP3 please ensure you use the settings below. Please you use Adobe Audition 2 or higher as it has been tested and works.

|  |  |
| --- | --- |
| **Mode** | Stereo |
| **Sample Rate** | 44100 Hz |
| **Bitrate** | Constant Bitrate of 128 kbit/s (The Bitrate can be lower than 128 but audio quality will be sacrificed so this will need to be mention this to the client if they want a module to be a small size) |

* + FLV

When converting AVI/MOV/MPG etc to FLV please ensure you use the settings below. Please you use Sorenson Squeeze 5 or higher as it has been tested and works.

|  |  |
| --- | --- |
| **Audio** | |
| **Codec** | Fraunhofer MP3 |
| **Data Rate** | 20 |
| **Sample Size** | 16 |
| **Sample Rate** | 11025 |
| **Channels** | Stereo |
| **Video** | |
| **Codec** | On2 VP6 Pro |
| **Method** | 1-Pass CBR |
| **Data Rate** | 360 |
| **Frame Size** | This will depended on required size to be used for a specific template or screen |
| **Display Aspect Ration Policy** | Unconstrained |
| **Frame Rate** | 2:1 Frames/Dec. |
| **Key Frame Every** | 30 frames |



#### Custom

As Above.

# Reference

This document specifies 12 quality assurance standards for all Savv-e courses.

1. **Navigation - T**he course has a consistent and intuitive navigation system enabling students to quickly locate course information and materials.
2. **Learner Orientation -** A course orientation is used to familiarize the students with the course.
3. **Syllabus -** Students have easy access to a course syllabus which contains crucial course information and requirements they need to know about the course prior to starting.
4. **Instructor Response and Availability -** Instructor response time and availability is clearly communicated to the student.
5. **Course Resource Requirements -** Hardware, software, or specialized resources required are clearly communicated to the students.
6. **Technical Support -** Information regarding access to technical support is clearly communicated to the students.
7. **Accessibility Requirements -** The course adheres to University policies and guidelines regarding accessibility.
8. **Learning Objectives -** The course contains learning goals and objectives.
9. **Learning Activities and Assessment -** The course learning activities and assessment serve to stimulate student interactions with the course content and determine how well student performance achieves the course goals and learning objectives.
10. **Copyright Requirements -** The online course adheres to the current University policies for the use of third-party copyrighted material or is able to provide evidence of appropriate copyright clearance.
11. **Course Functionality -** All aspects of the course perform properly and support student progress.
12. **Student Input for Course Improvements -** Opportunities are provided to gather input from students on an on-going basis in order to inform course improvements.