

---

---

## NSSC-HR SDG-0038 - Appendix B – Training Content Specifications

Prior to upload to SATERN, it is the responsibility of the NSSC to insure that modules (including videos):

- are accessible via the NASA and SATERN approved list of Web browsers and hardware as listed in the Client Reference Configuration in NASA-STD-2804M, Effective August, 2009
- are Section 508 compliant
- are Sharable Content Object Reference Module (SCORM) conformant
- meet NASA Internet Publishing Content Guidelines

Modules that do not meet these guidelines will not be accepted into the SATERN Learning Management System. The following sections will give a cursory look at the minimum browser sets, Section 508 Compliance, SCORM, and the NASA Internet Publishing Content Guidelines.

Approved list of Web browsers (from the Client Reference Configuration in NASA-STD-2804M, effective August, 2009):

<b>Windows XP</b>			
Browser	Interface Standard	Version	Effective Date
Mozilla Firefox	W3C and industry standards, including the following: HTML 4.01, XHTML 1.0 CSS 2 (Cascading Style Sheets) ECMA script (JavaScript) capability to run Java 2 applets, SSL version 2 and 3, TLS 1.0	3.0.x	October 1, 2008
Microsoft Internet Explorer		7.0.x	June, 2009
<b>Mac OS X</b>			
Mozilla Firefox	W3C and industry standards, including the following: HTML 4.01 XHTML 1.0 CSS 2 (Cascading Style Sheets) ECMA script (JavaScript) capability to run Java 2 applets, SSL version 2 and 3, TLS 1.0	3.0.x	October 1, 2008
Apple Safari		4.0.x	July 2009

### Section 508 Compliance

NASA Section 508 Web Policy states:

"all NASA Web sites, i.e., all Web sites in the 'nasa.gov' Internet domain whether hosted by NASA or for NASA under contracts, grants, or other agreements and all Web sites provided by or hosted by NASA contractors and grantees where the contract or grant requires the provision of Web services on behalf of NASA's missions, even if the Web sites are not in the 'nasa.gov' Internet domain must meet requirements set by Section 508 of the Rehabilitation Act. Owners of Web sites must make their Web sites accessible, or, if doing so constitutes an undue burden, provide other means to make the information or service provided at the Web site available."

## **Section 508 Standards Summary**

According to The U.S. Access Board, an independent Federal agency devoted to accessibility for people with disabilities, a Web site will be in compliance with the 508 standards if it meets paragraphs (a) through (p) of Section 1194.22. They are summarized as follows:

- (a) Text Tags - A text equivalent for every non-text element shall be provided (e.g., via "alt", "longdesc", or in element content).
- (b) Multimedia Presentations - Equivalent alternatives for any multimedia presentation shall be synchronized with the presentation.
- (c) Color - Web pages shall be designed so that all information conveyed with color is also available without color, for example from context or markup.
- (d) Readability (style sheets) - Documents shall be organized so they are readable without requiring an associated style sheet.
- (e) Server-Side Image Maps - Redundant text links shall be provided for each active region of a server-side image map.
- (f) Client-Side Image Maps - Client-side image maps shall be provided instead of server-side image maps except where the regions cannot be defined with an available geometric shape.
- (g) Data Table - Row and column headers shall be identified for data tables.
- (h) Multi-logic Row or Column Headers - Markup shall be used to associate data cells and header cells for data tables that have two or more logical levels of row or column headers.
- (i) Frames - Frames shall be titled with text that facilitates frame identification and navigation.
- (j) Flicker Rate - Pages shall be designed to avoid causing the screen to flicker with a frequency greater than 2 Hz and lower than 55 Hz.
- (k) Text-Only Alternative - A text-only page, with equivalent information or functionality, shall be provided to make a Web site comply with the provisions of this part, when compliance cannot be accomplished in any other way. The content of the text-only page shall be updated whenever the primary page changes.
- (l) Scripts - When pages utilize scripting languages to display content, or to create interface elements, the information provided by the script shall be identified with functional text that can be read by assistive technology.

(m) Applets and Plug-Ins - When a Web page requires that an applet, plug-in or other application be present on the client system to interpret page content, the page must provide a link to a plug-in or applet that complies with §1194.21(a) through (l).

(n) Electronic Forms - When electronic forms are designed to be completed online, the form shall allow people using assistive technology to access the information, field elements, and functionality required for completion and submission of the form, including all directions and cues.

(o) Navigation Links - A method shall be provided that permits users to skip repetitive navigation links.

(p) Time Delays - When a timed response is required, the user shall be alerted and given sufficient time to indicate more time is required.

### **Additional resources:**

There are several sources of information regarding Section 508 Web policy and compliance. Here are just a few resources that can be found via the Internet:

- NASA Section 508 Web site: (<http://www.section508.nasa.gov/>)
- The U.S. Access Board 508 Website: (<http://www.access-board.gov/508.htm>)
- The U.S. Access Board Guide to the Section 508 Standards: (<http://www.access-board.gov/sec508/guide/index.htm>)
- Web Accessibility Made Easy, U.S. Government Website Accessibility guidelines: (<http://www.hq.nasa.gov/webaccess/index.html>)
- Infocus: (<http://www.ssbtechnologies.com/products/infocus/index.html>)

### **Sharable Content Object Reference Model (SCORM)**

All online training developed for SATERN must be SCORM 1.2 conformant.

The Sharable Content Object Reference Model (SCORM) specification was designed by Advanced Distributed Learning (ADL) to provide a technical means to share distance learning courses (objects) among the different learning delivery environments. One objective of SCORM is the ability to move a course from one learning platform to another. Another is to create reusable chunks (SCOs) of Web-based learning content to enable its use in other courses. Once a common language or standard is defined, then all systems that are built using that language can "talk" to each other (in theory). SCORM addresses these issues by specifying a standard way for defining and accessing information about learning objects.

Developing to the SCORM standards allows your course to appear almost identical in any SCORM compatible LMS on the market. Using SCORM commands can also allow you to control how users advance through your course by only unlocking certain

sections once they have completed the sections that come before it. You can also set bookmarks in your courses to allow learners to jump directly to the bookmark the next time they enter the course.

ADL currently has two standards which are widely used, version 1.2 and version 1.3 (aka 2004). Technical specs for both versions can be downloaded from ADL's Web site. SATERN currently only supports the SCORM 1.2 standards. Using the SATERN customized version of ADL's JavaScript API interfaces (see below) should eliminate any problems with browser compatibility issues.

## **SCORM 1.2**

SCORM 1.2 is the easier of the two SCORM standards to program for. All that is required is a valid course Manifest file (named `imsmanifest.xml`), a call to the LMS to initialize the course (via JavaScript) and a call to the LMS to finish the course (again via JavaScript). Of course, there are other things that you can do with SCORM 1.2, but these are the only pieces that you must provide.

The manifest file is an XML file that must be named '`imsmanifest.xml`' and has to reside at the root directory of your content module. It can contain five parts: Manifest tag, Metadata (optional) tags, Organizations, Resources, and Sub-Manifests (optional). The format of the manifest file is very specific, so you must be careful when writing it.

The Manifest tag just contains references to the various SCORM reference documents that are stored online. This section rarely changes from module to module. Since the reference documents are kept online there is a potential that they could not be available all the time. Most developers tend to place local copies of these documents within their courses to avoid problems should the reference Web sites not be available at the time the course is run. It is the developer's responsibility to make sure that the local copies of the files are current with the ones on the reference Web sites.

The Metadata section of the manifest file is optional. This section allows the developer to add information specific to the module such as copyright notices, browsers supported, languages, etc.

The Organizations section allows the developer to decide how they want their content to be displayed. It is this section that allows you to generate the table of contents for the course which will be displayed by the LMS the course is loaded into. You can have multiple layouts for your course by using different Organization tags within this section. The Resources section is the most technical section of the manifest. It is in this section that you list every file used by the module, and how the LMS should treat them. The two main types of resources are Assets and Sharable Content Objects (SCO). An asset is like a library. It contains files which can be used by multiple SCOs. A SCO is an independent 'block' of the module which can stand by itself. These 'blocks' can be used by other courses in the future. If you do not set up an organization, you might want to consider organizing your resources into a single SCO or asset.

The Sub-Manifest is an optional section that allows you to incorporate other learning content into your course.

The second requirement for SCORM 1.2 is to have the course let the LMS know that the user has entered the course. This is done through a JavaScript call: `LMSInitialize()` and is usually done when the page loads. This call tells the LMS that the user is attempting the learning resource and sets flags accordingly.

The third requirement for SCORM 1.2 is to let the course know when the user has completed the course. This is done through the JavaScript call: `LMSFinish()` and is usually done on the last page of the learning material and is done when the page unloads. This call lets the LMS know that the user has probably gone through the course material, and the course is now complete for them.

## **SATERN and SCORM**

Communication between SATERN and the online training is done through a Java API that SATERN launches on the learner's computer. This is why the Java Runtime Environment (JRE) is required by SATERN. Content developers do not need to worry about the API, SATERN will start it automatically when the online training is launched. You will be able to communicate with the API via JavaScript calls to ADL's JavaScript libraries.

It is extremely important that you let SATERN know when the learner has exited your online training, even if they do not use the controls you put in your code to exit. When the training window is closed, you must terminate communication with SATERN. Failure to do so will cause problems if the learner tries to launch any other courses. Keep in mind that browser windows can be closed unexpectedly and in ways that you do not anticipate (such as in the middle of the SCO). It is up to you to make sure that communication is still concluded with the LMS in these situations.

To tell SATERN that the learner has completed the online training, you must pass a `lesson_status` of 'passed.' Any other value sent will not record a completion. ADL's model for SCORM, found in most of their examples, focuses on Microsoft Internet Explorer on the Windows platform.

You can implement SCORM controls in quite a few different ways. One of the easier methods is the 'SCORM wrapper' method where you use a frameset to mostly keep the SCORM controls separate from your online content. We've provided a sample working 'course' below that contains the SATERN modified API libraries as well as instructions for implementing the SCORM wrapper method.

## **SATERN SCORM Tips**

This section provides simple tips for designing your learning content in SCORM with the least amount of potential problems with SATERN.

- SATERN only grants credit for a `cmi.core.lesson_status` value of 'passed'.
- ADL's SCORM API interfaces that are used in their examples are usually tailored to users using Internet Explorer on a Windows platform.
- SATERN will display each item in the Organization area of your manifest on a 'Content Structure' page. These items should point to a specific Sharable Content Object (SCO) and not an asset.
- If you have more than one SCO in your learning content, DO NOT provide links between the two. Users must return to the 'Content Structure' page in order to proceed to the next SCO in your learning material; otherwise SATERN will not properly record an item's completion on the 'Content Structure' page.
- It is up to you to handle the communication between your SCO and the LMS. When the training window is closed, you must terminate communication with the LMS. Failure to do so will cause problems if the user tries to launch the course again. Keep in mind that browser windows can be closed unexpectedly and in ways that you do not anticipate (such as in the middle of the SCO). It is up to you to make sure that communication is still concluded with the LMS in these situations.
- If your content consists of a single SCO, it may be more convenient to place your learning content in a 'SCORM wrapper.' Create a two frame frameset containing one visible frame and one invisible frame. Place your learning content in the visible frame and all SCORM interaction in the hidden one. This gives you a central location to troubleshoot any SCORM issues that may arise in your learning content.
- When using ADL's Conformance Test Suite v. 1.2.7, test your archive file using the Content Package Conformance Test. Pay special attention to the note listed in the instructions. You have the option to just test your manifest file, or your entire learning content package. If you have an Organization section in your manifest, you probably want the Content Aggregation Package option.

### **NASA Internet Publishing Content Guidelines**

All online training on SATERN must meet the NASA Internet Publishing Content Guidelines found in section 11.3.9 of NPR 2810.1A (Security of Information Technology). These guidelines identify and categorize information that can and cannot be posted on the Internet. Visit <http://nodis3.gsfc.nasa.gov/> for the latest version of this NPR.

SATERN is not a publicly-accessible Web site but if your online training contains any of the prohibited information, you must take additional precautions when developing your training to make sure that only valid learners of SATERN can access your materials.

## Development Tools

### HTML

- CSE HTML Validator (<http://www.htmlvalidator.com/>) - Checks your HTML code for any problems that might affect how the code displays. The Lite version is free.
- Total Validator (<http://www.totalvalidator.com/>) - Checks your HTML code for any problems that might affect how the code displays, broken links, spelling, and Section 508 issues. There is a free version available.
- Arachnophilia (<http://www.arachnoid.com/arachnophilia/>) - A HTML editor which is good for people who are unfamiliar with HTML. The Web site also contains some basic HTML tutorials. The product is CareWare (free, but with a modest behavior request from the creator).

### Section 508

- InFocus (<https://www.ssbartgroup.com/>) - Tests your code for Section 508 issues. There is not a free version available.
- Total Validator (<http://www.totalvalidator.com/>) - Checks your HTML code for any problems that might affect how the code displays, broken links, spelling, and Section 508 issues. There is a free version available.

### SCORM

- ADL's Web site (<http://www.adlnet.gov/>) - Contains lots of tools to help with SCORM development, including SCORM examples, a sample SCORM conformant Learning Management System, and the SCORM Version 1.2 Conformance Test Suite Version 1.2.7.