San José State
UNIVERSITY

**Accessible Technology Initiative (ATI) Task Force**

# Web Accessibility Training Manual for Webmasters

**Web site:** http://www.sjsu.edu/accessibility
**Email:** atiweb@sjsu.edu

# Table of Contents

**Section 1: Overview**
Executive Summary ....02
University Template ....02
Special Thanks ....02
Webmaster Instructions ....03
Help ....03

**Section 2: Install Web Auditing Tools**
HiSoftware AccVerify (Required) ....04
Additional Tools (Recommended) ....20

**Section 3: Conduct Manual Accessibility Evaluation**
Retrofit to Law — Build to Best Practice....22
Syntax Validation ....23
Semantic Validation ....27
Checkpoint A: Text Equivalents ....30
Checkpoint B: Multimedia Equivalents ....35
Checkpoint C: Color ....36
Checkpoint D: Styles ....40
Checkpoint E: Server-Side Image Maps ....44
Checkpoint F: Client-Side Image Maps ....45
Checkpoint G: Simple Tables ....47
Checkpoint H: Complex Tables ....49
Checkpoint I: Frames ....52
Checkpoint J: Flicker....54
Checkpoint K: Text Only ....55
Checkpoint L: Scripts ....56
Checkpoint M: Applets and Plug-Ins ....61
Checkpoint N: Forms ....63
Checkpoint O: Skip Links ....68
Checkpoint P: Timed Response ....72
User Validation ....74

**Section 4: Run Accessibility Report with Interview Wizard**
Global Settings....78
Run Report....92
Understanding and Evaluating Your Report....103

# Section 1: Overview

### Executive Summary

The California State University (CSU) system has launched a system-wide Accessible Technology Initiative (ATI) to ensure that each CSU campus follows state and federal laws mandating access to technology as well as to ensure a culture of inclusive learning.

The San José State University web site, along with all related subsites and affiliated web sites, must be made accessible to the disabled per Section 508 standards. Universal accessibility to the web means that all people, regardless of their physical or developmental abilities or impairments, have access to web-based information and services.

Webmasters (those employees responsible for maintaining a university subsite) must attend training, submit a preliminary ***Before Retrofit*** report, then retrofit their web page(s) for Section 508 compliance and submit an ***After Retrofit*** report.

These ***Before Retrofit*** and ***After Retrofit*** reports will be used to determine campus funding needs for evaluating and retrofitting all campus web pages. For this reason, it is important that University Webmasters submit a preliminary report ***before*** making any needed accessibility retrofits.

For additional information, visit SJSU Accessible Technology Initiative.

### University Web Templates

SJSU Web Services Unit has developed a series of accessible web templates that are Section 508 compliant. If you are currently using a University Template for your subsite, you can request an upgrade to the accessible templates. If you are not currently using the University Template, you may wish to consider doing so during your next subsite redesign. For more information, visit the SJSU Web Services Unit.

### Special Thanks

***Tom Jewett***, CSU Web Accessibility Project Lead, has been very generous with his time in training the SJSU ATI Web Team. Tom wrote much of the information provided in this training manual (for which we are very grateful). He is a Lecturer at CSU Long Beach, Department of Computer Engineering and Computer Science. For more information, visit Tom Jewett Web Design/Specializing in Accessibility.

### Webmaster Instructions

Webmasters (those employees responsible for maintaining a university subsite) are encouraged to follow these steps:

**Step 1: Join the Web Mailing List**
The Web Mailing List is open for anyone maintaining web pages, web site(s) or providing content for any domains inside SJSU.EDU. We recommend that you join this mailing list so that you can be advised about accessibility issues and due dates. To sign up, visit SJSU Accessibility Web Mailing List.

**Step 2: Install Needed Software:** Complete the ***HiSoftware Web Auditing Tools Request Form*** available at SJSU Accessible Technology Initiative. Present the form with your Tower ID to the University Help Desk located on the first floor of Clark Hall. They will provide you with the CD. Refer to Section 2 of this training manual for installation instructions and recommendations for additional tools.

**Step 3: Attend Training:** Learn how to evaluate your web sites for compliance with accessibility standards and ways to resolve accessibility problems. Register online at the SJSU Accessible Technology Initiative.

**Step 4: Perform Preliminary Accessibility Evaluation:** Refer to Sections 3 and 4 of this training manual for instructions on how to perform automated and manual accessibility evaluations of your home page and how to submit your ***Before Retrofit*** report to the ATI Task Force.

**Step 5: Retrofit Your Home Page:** Refer to Section 4 of this training manual for help in understanding your AccVerify report and retrofitting your home page for Section 508 compliance. If you need assistance in doing this, please contact the SJSU ATI Task Force at atiweb@sjsu.edu.

**Step 6: Perform Second Accessibility Evaluation:** After retrofitting your home page for accessibility, perform a second automated and manual accessibility evaluation as described in sections 3 and 4 of this training manual. Submit your ***After Retrofit*** report to the ATI Task Force at atiweb@sjsu.edu.

**Step 7: Evaluate and Retrofit Entire Subsite:** Use the information in this training manual and your experience from retrofitting your home page to make any needed retrofits to your entire subsite. For due dates on completing this task, visit SJSU Accessible Technology Initiative.

### Help

For additional information, visit SJSU Accessible Technology Initiative. If you need help evaluating and/or retrofitting your web site to Section 508 standards, please contact the SJSU ATI Task Force at atiweb@sjsu.edu for assistance.

Also, we highly recommend the book, ***Web Accessibility: Web Standards and Regulatory Compliance***, by Jim Thatcher, Andrew Kirkpatrick, Richard Rutter, Christian Heilmann, Cynthia Waddell, Michael R. Burks, Shawn Lawton Henry, Bruce Lawson, Mark Urban, and Patrick H. Lauke, published by Friends of Ed, July, 2006, ISBN: 159059638.

# Section 2: Install Web Auditing Tools

This section explains how to download and install HiSoftware AccVerify. This software is required for all SJSU Webmasters and is available for free from the University Help Desk in Clark Hall. This section also provides information about obtaining recommended software tools for use in manual accessibility evaluations of web sites.

### Installing AccVerify

AccVerify/Repair is a comprehensive toolset for validating and repairing the accessibility, quality, privacy and usability of web sites. AccVerify and AccRepair support standards-based validation (Section 508, WCAG 1.0), automated repairs and custom testing in a variety of reporting formats.

**System requirements.**

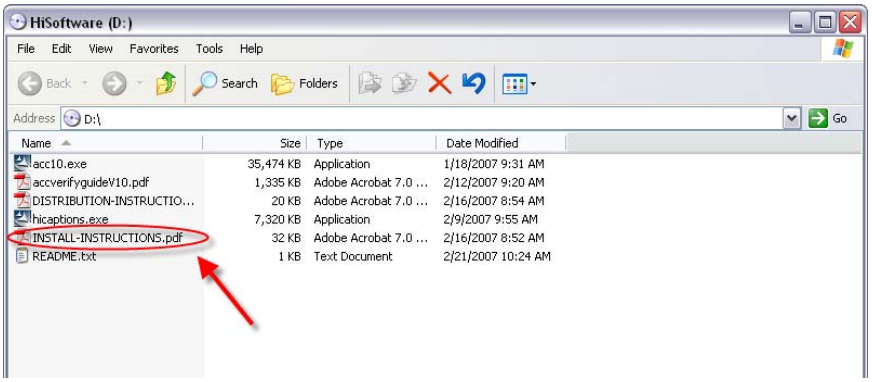
AccVerify/Repair Single User/Single PC Desktop License System Requirements:

- Windows 95 or later
- 32 MB RAM
- 20 MB of free hard drive space for installation
- VGA Monitor, Super VGA recommended

1. Insert the CD provided by the SJSU Help Desk.

**The HiSoftware window opens.**

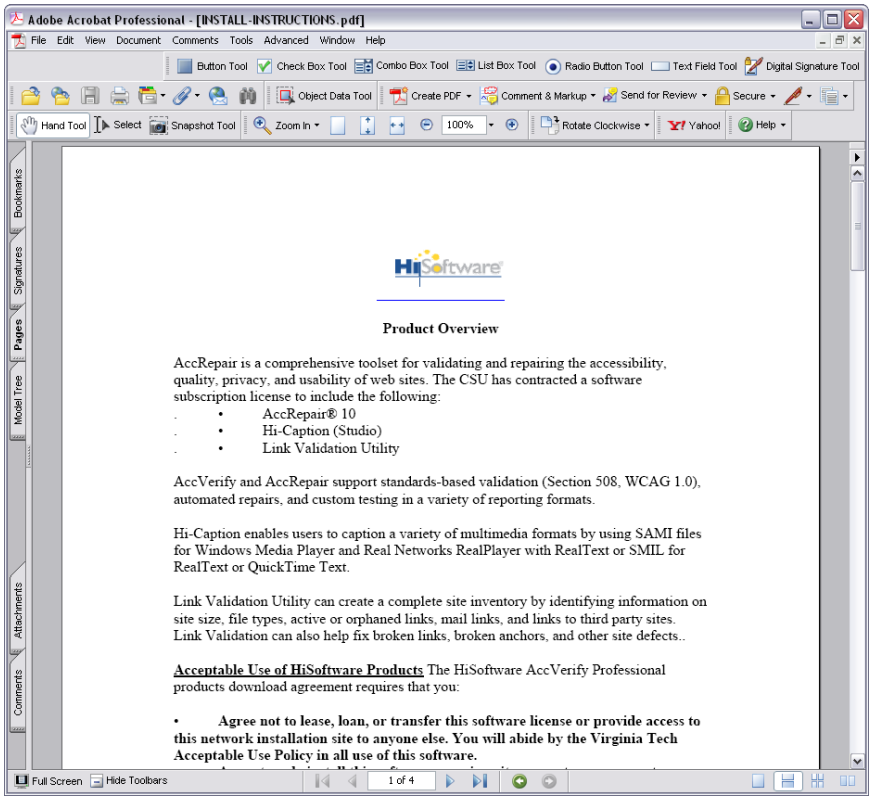
2. Double click on the **INSTALL-INSTRUCTIONS.pdf** file.



**The PDF document opens**.

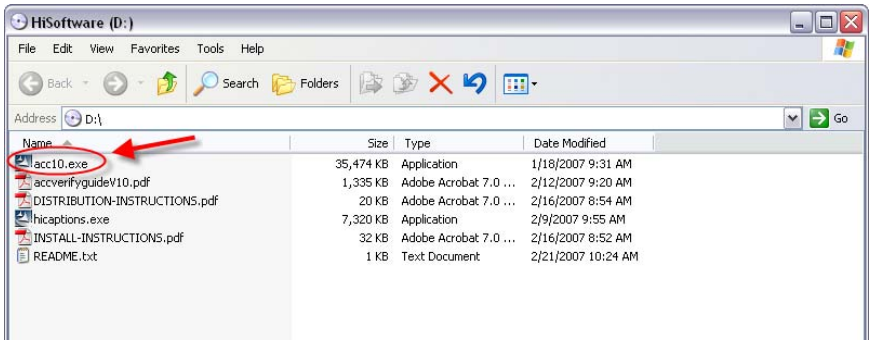
3. Make a note of installation password, company name, and license key code on page 2. You will need this information for the installation.

4. Close or minimize this document.



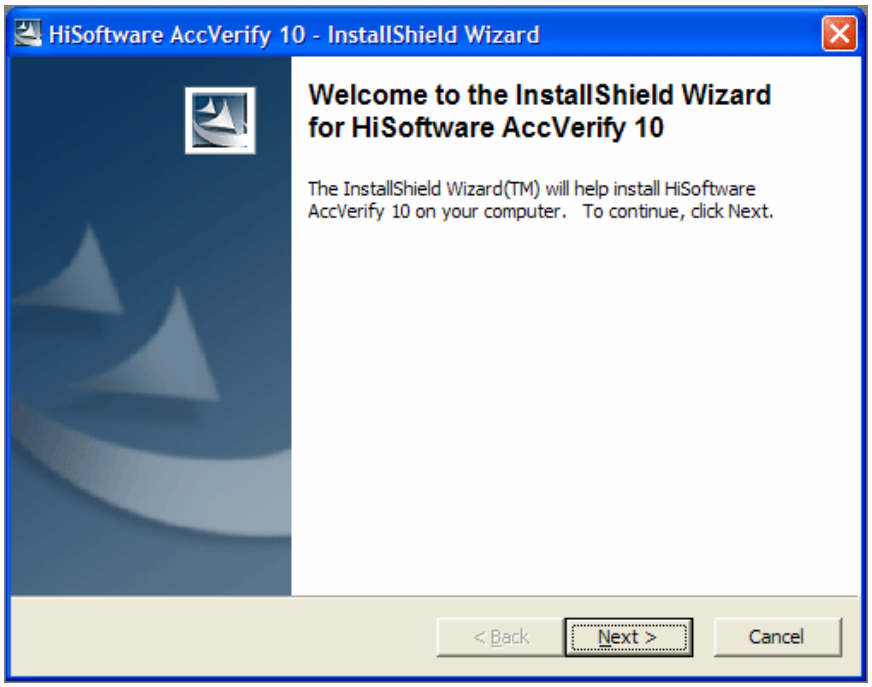

**The HiSoftware window displays.**

5. Double click on the **acc10.exe** file.



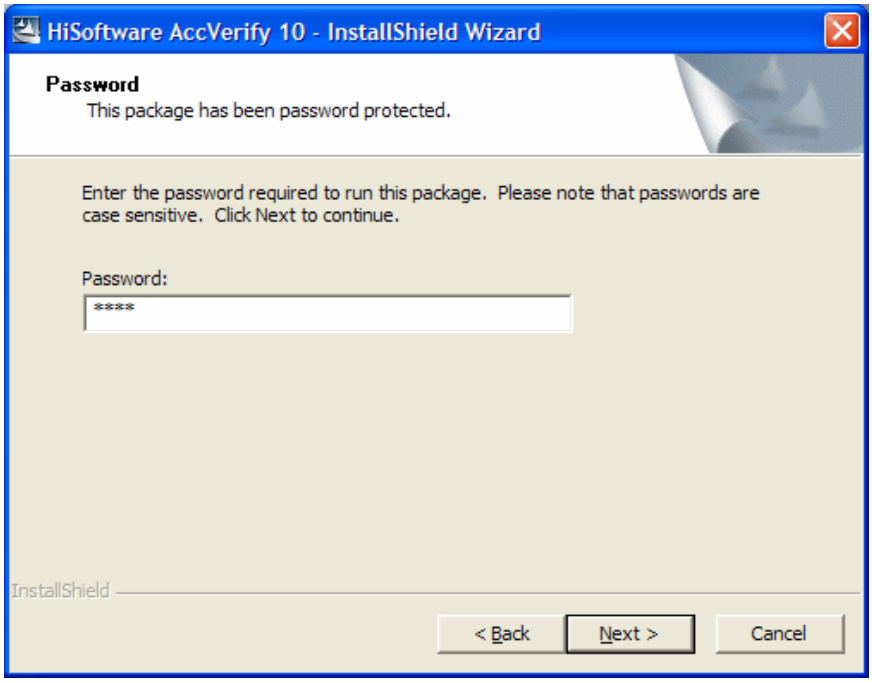
**The HiSoftware AccVerify 10 InstallShield Wizard window displays.**

6. Click on the Next button.




**The Password window displays.**

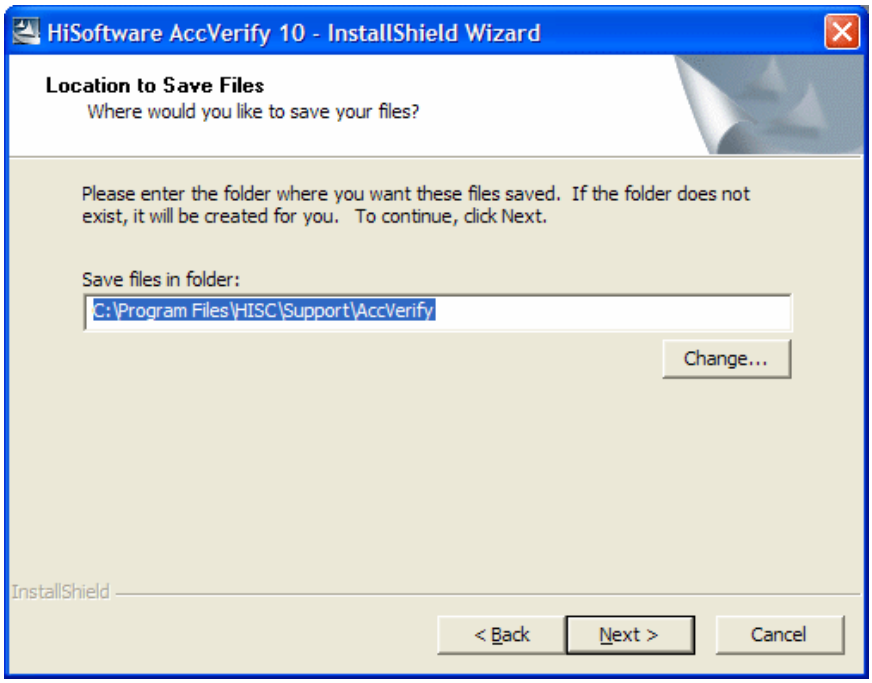
7. **Password:** Enter password provided in HiSoftware installation instructions.



**The Location to Save Files window displays.**

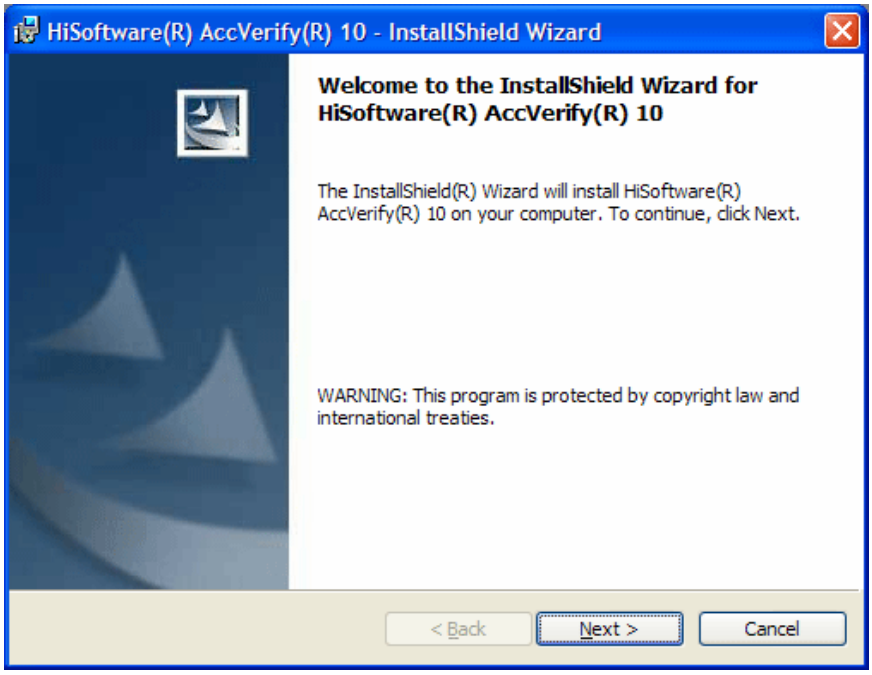
8. If you would like to change location, click on the **Change** button and select desired location.

9. Click on the **Next** button.




**The Welcome window displays.**

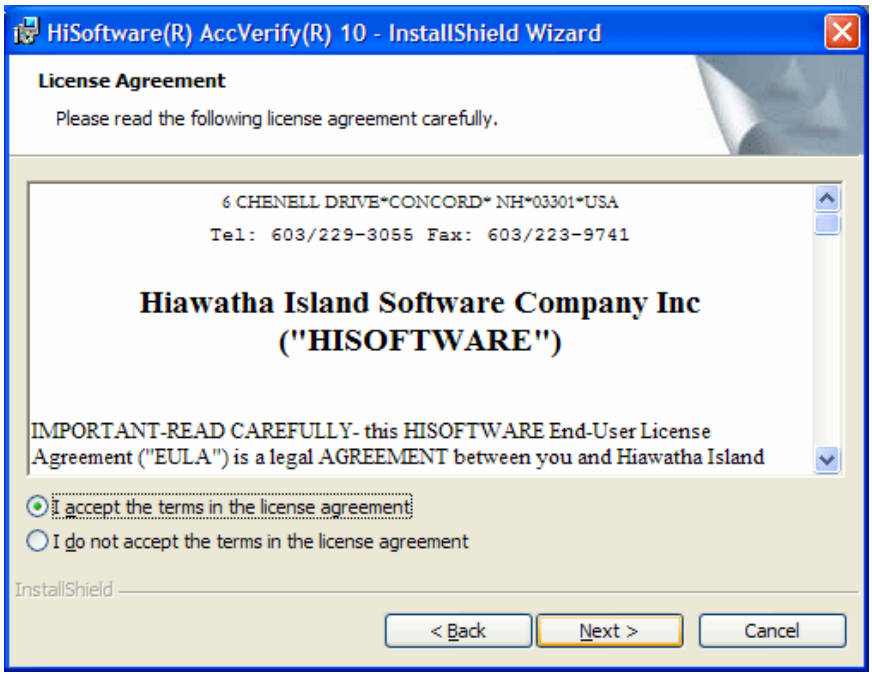
10. Click on the **Next** button.



**The License Agreement page displays**.

11. Select **I accept the terms in the license agreement**.

12. Click on the **Next** button.

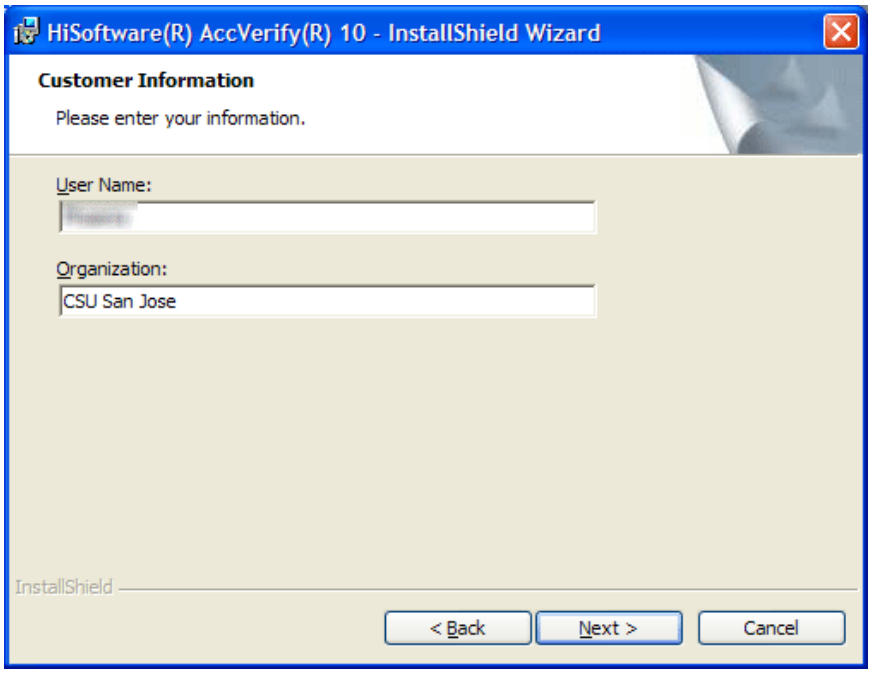



**The Customer Information window displays.**

13. **User Name:** Enter your first and last name.

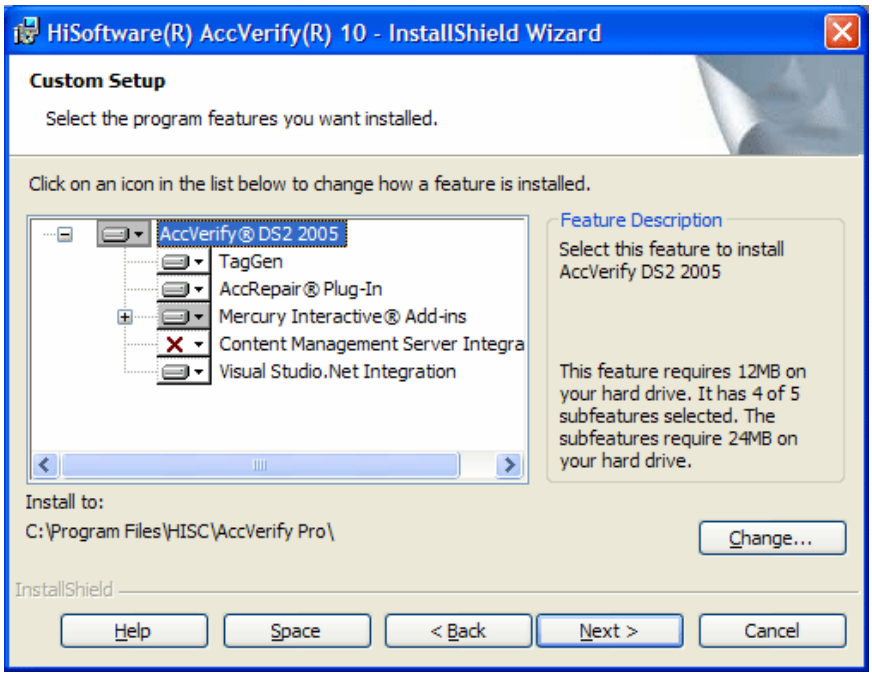
14. **Organization:** CSU San Jose.

15. Click on the **Next** button.



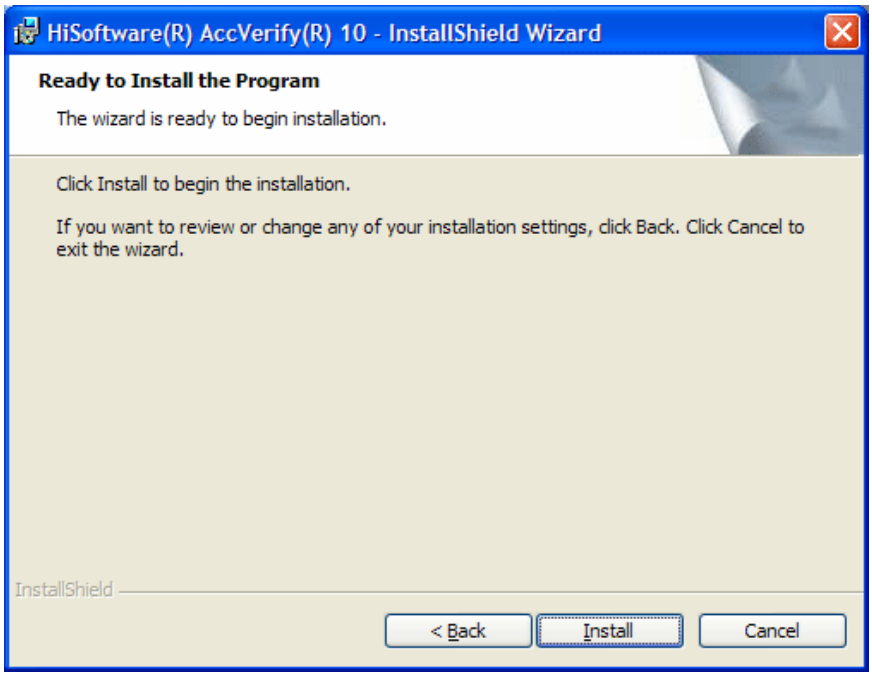
**The Custom Setup window displays.**

16. Click on the **Next** button.



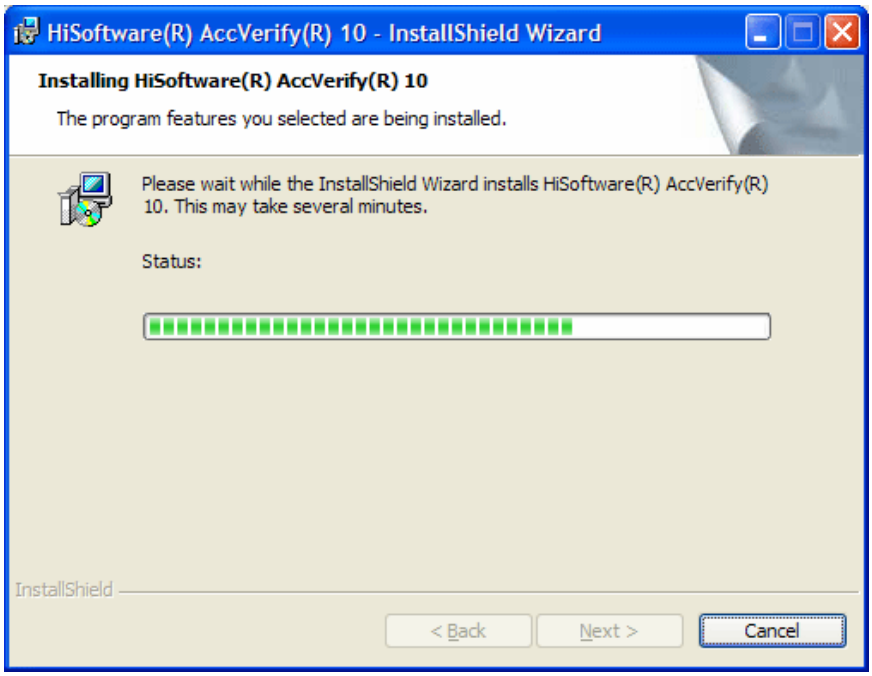

**The Ready to Install the Program window displays.**

17. Click on the **Install** button.



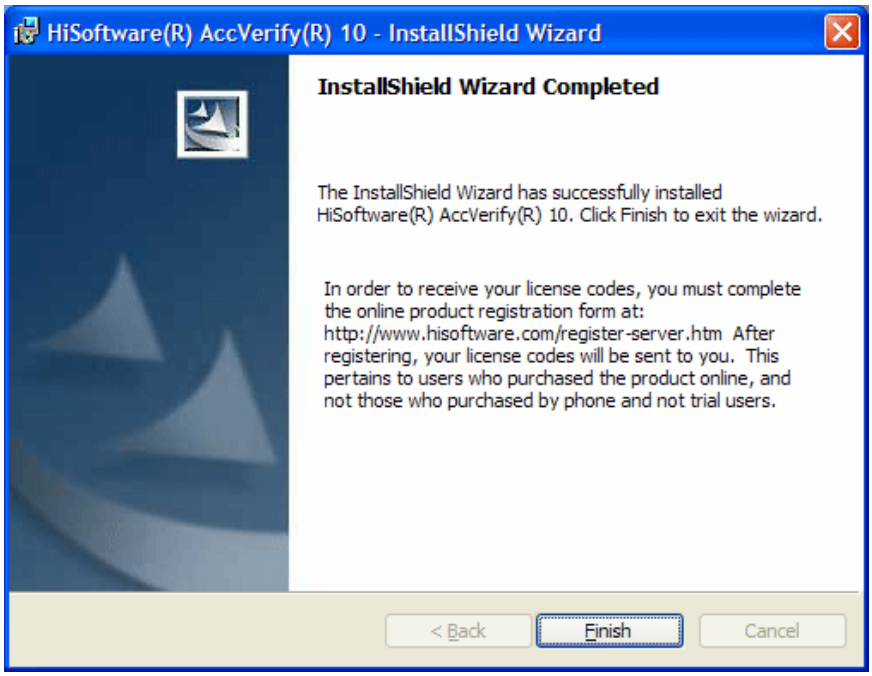
**The Installing HiSoftware AccVerify window displays**.

18. Wait for installation to be completed.



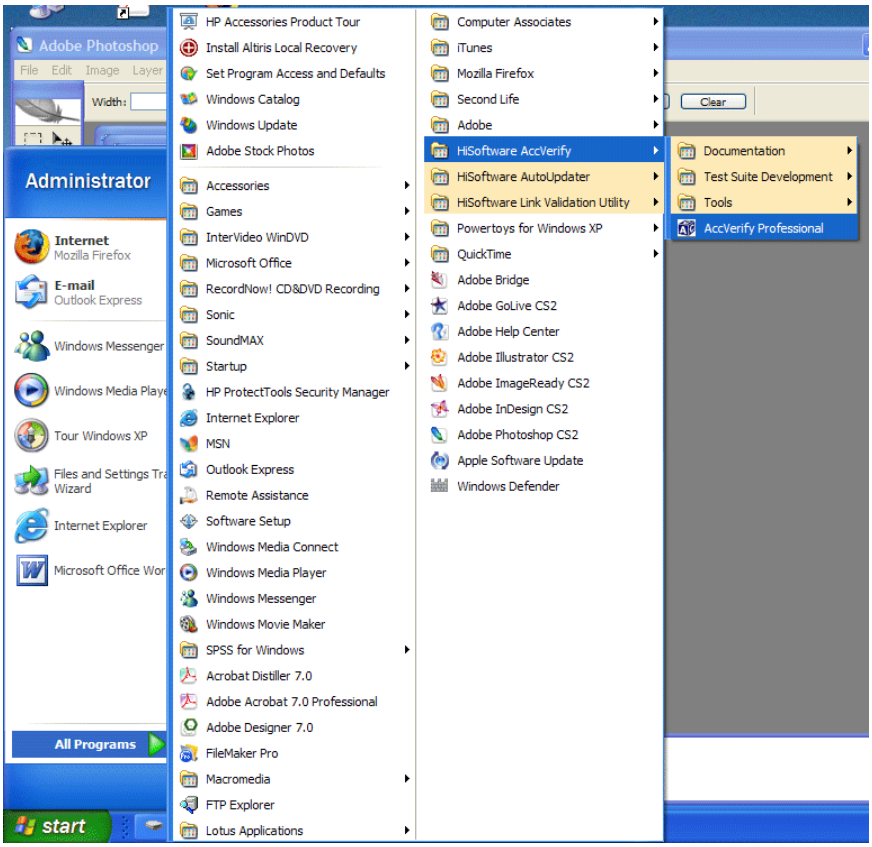

**The Installation Wizard Completed window displays**.

19. Click on the **Finish** button.



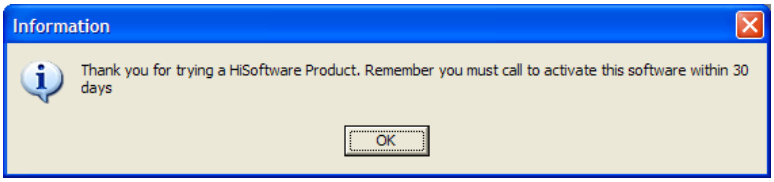
**Start HiSoftware AccVerify.**

20. Navigate to **Start** > **Programs** > **HiSoftware AccVerify** > **AccVerify Professional**.




**The Information window displays.**

21. Click on the **OK** button.



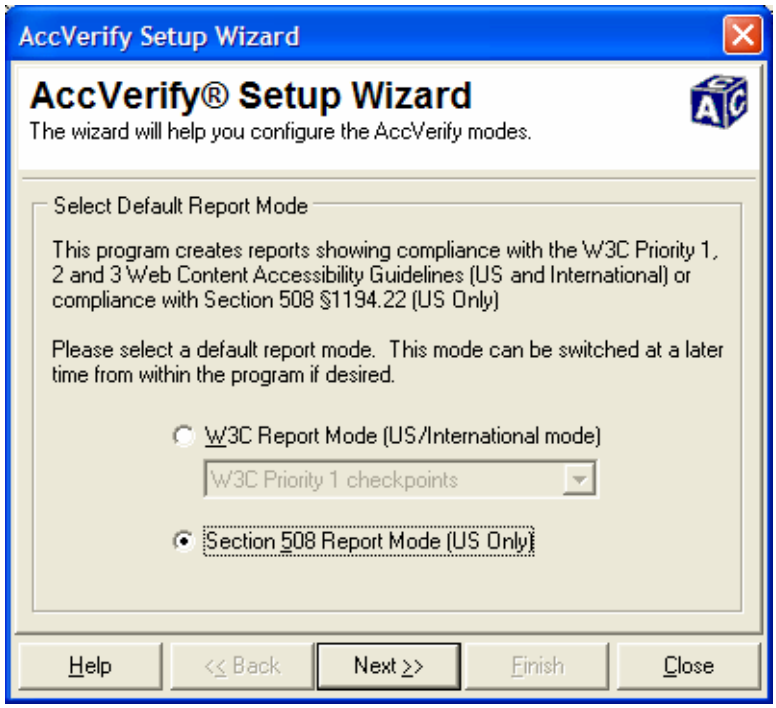

**The HiSoftware AccVerify window displays.**

22. Click on the **OK** button.

**The Select Default Report Mode window displays.**

23. Select **Section 508 Report Mode**.

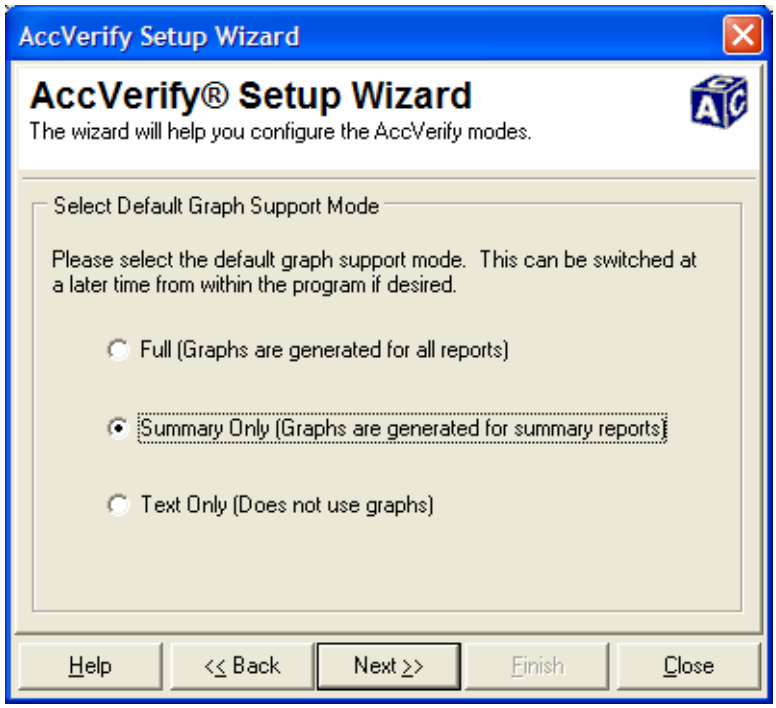
24. Click on the **Next** button.




**The Select Default Graph Support Mode window displays.**

25. Select **Summary Only**.

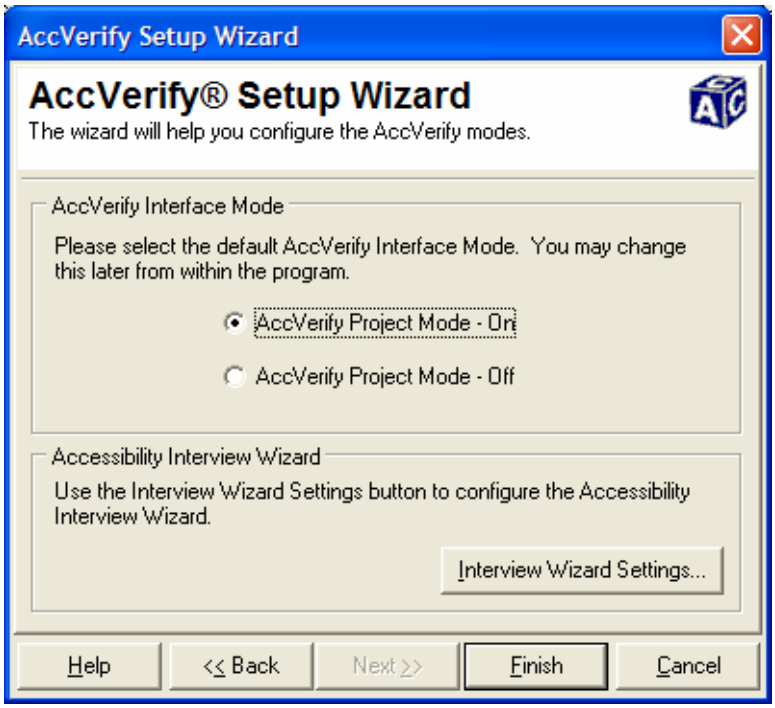
26. Click on the **Next** button.



**The AccVerify Interface Mode window displays.**

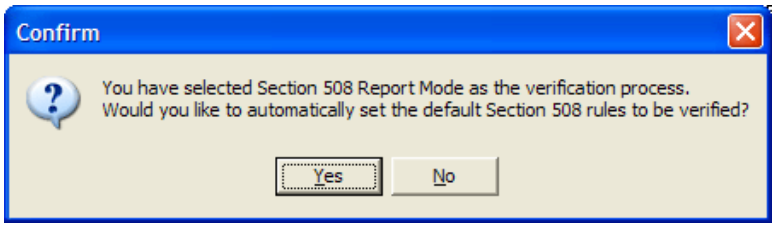
27. Select **AccVerify Project Mode – On**.

28. Click on the **Finish** button.



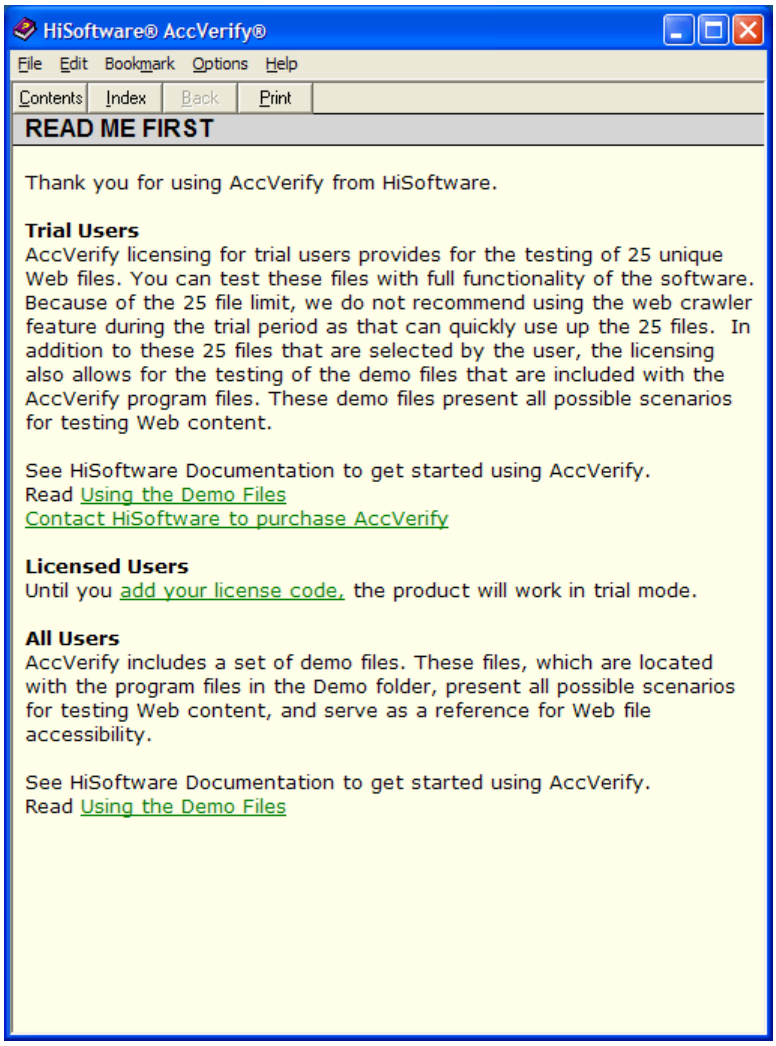

**The Confirm window displays**.

29. Click on the **Yes** button.



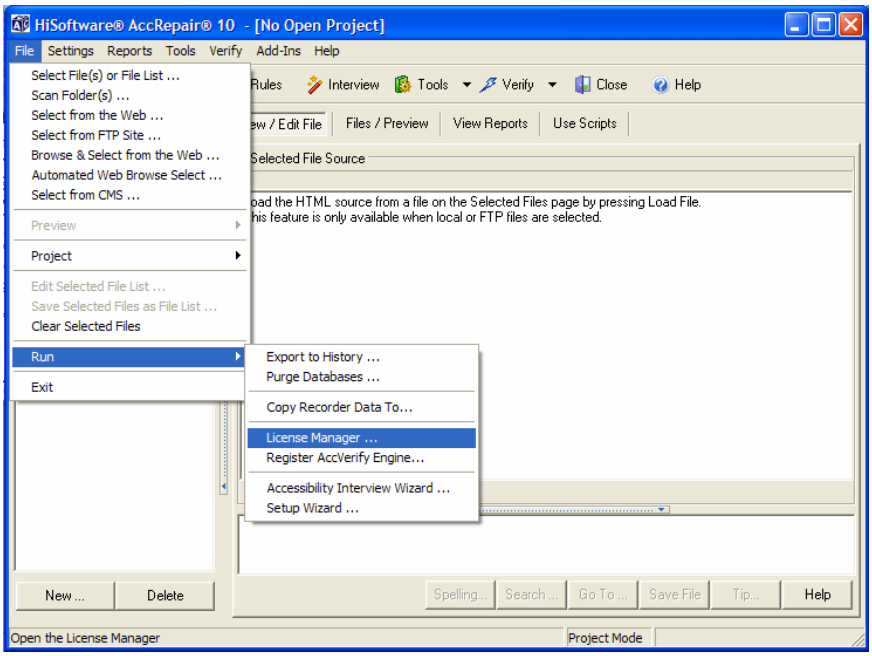
**The Help window displays.**

30. Close this window if you don't need it.



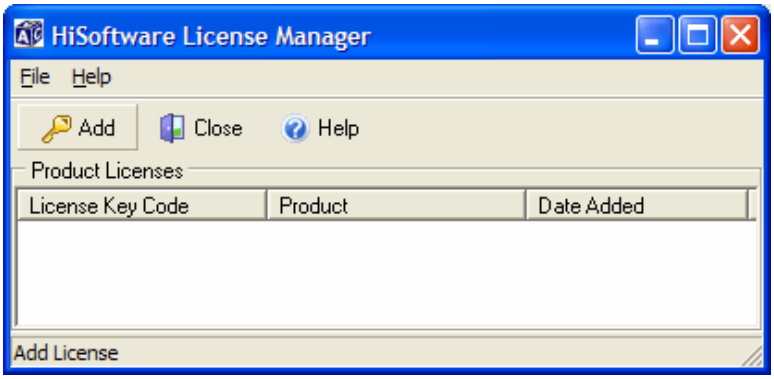
**The HiSoftware AccRepair window displays.**

31. Navigate to **File** > **Run** > **License Manager**.




**The License Manager window displays.**

32. Click on the **Add** button.

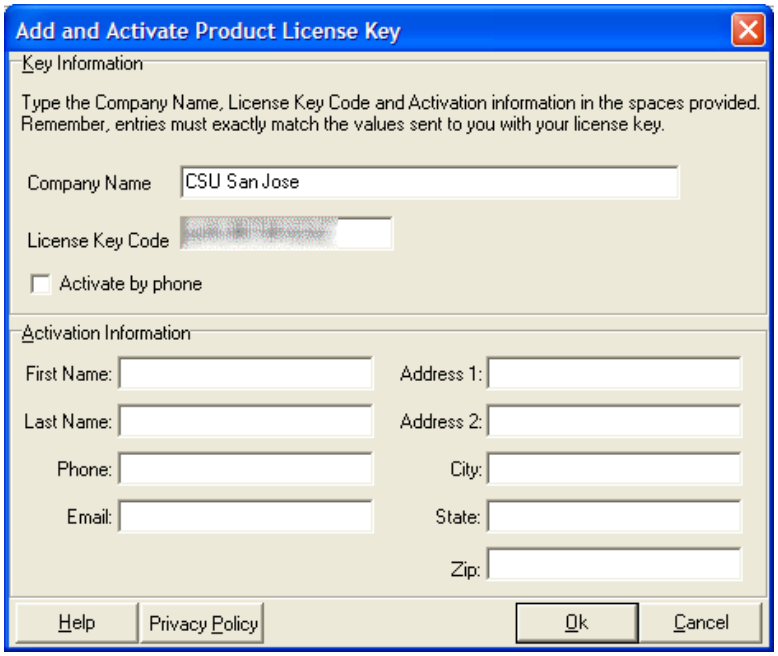


**The Add and Activate Product License Key window displays.**

33. **Company Name:** Enter CSU San Jose.

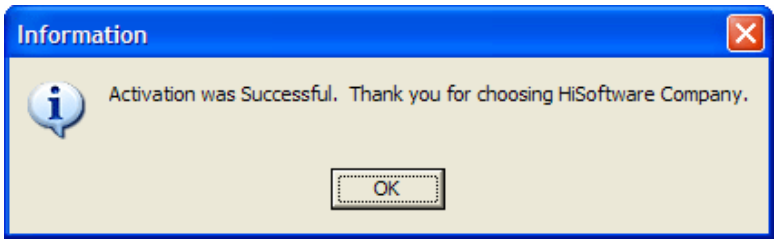
34. **License Key Code:** Enter key code provided in HiSoftware installation instructions.

35. Click on the **OK** button.



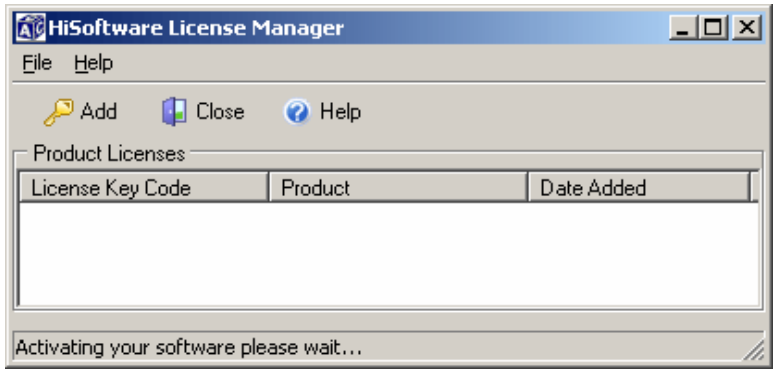

**The Information window displays.**

36. Click on the **OK** button.



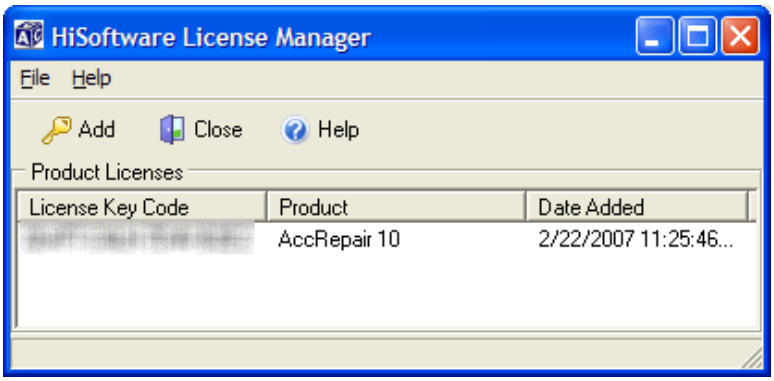

**The License Manager window displays.**

37. Wait while software is being activated.



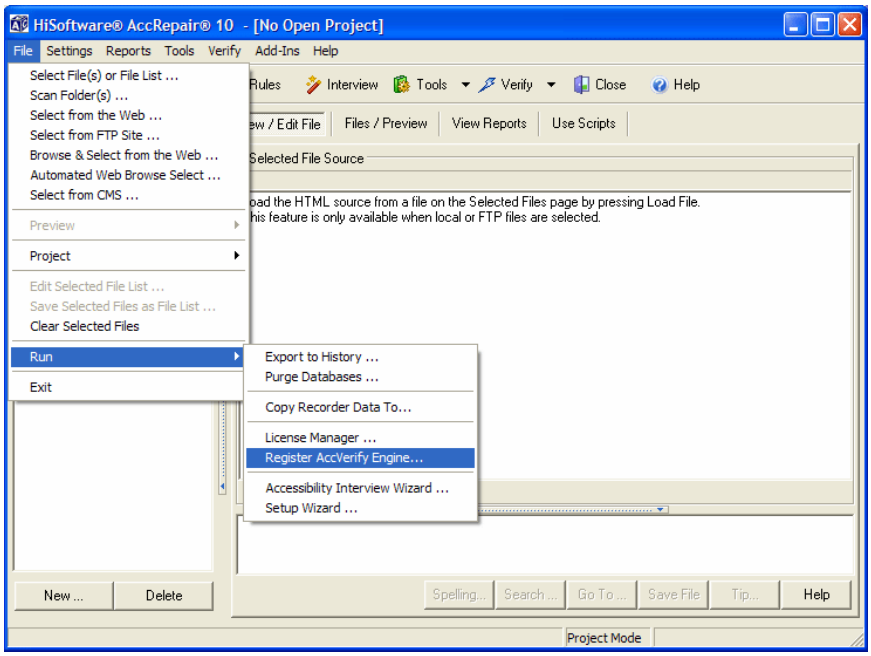
**The license key code appears**.

38. Click on the **Close** button.




**The HiSoftware AccRepair window displays.**

39. Navigate to **File** > **Run** > **Register AccVerify Engine**.



**The Unlock Information window displays.**

40. **Serial Number:** This number is defined by the system.

41. **Name:** Your name.

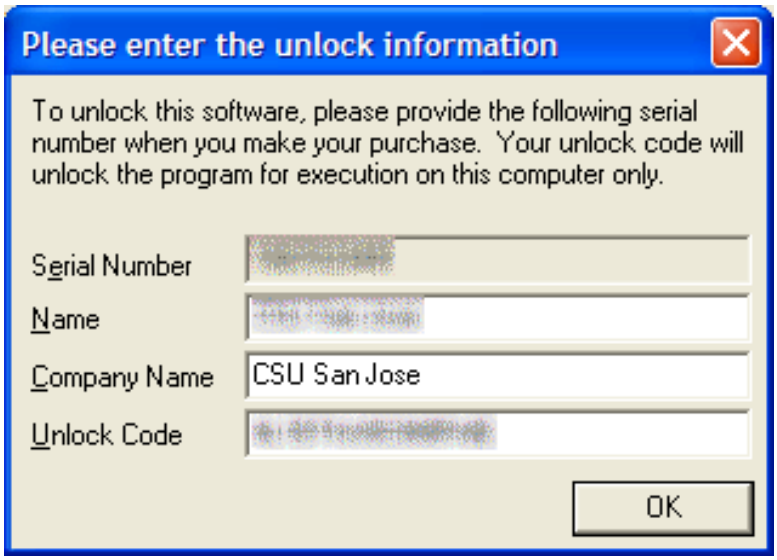
42. **Company Name:** CSU San Jose

43. **Unlock Code:** Contact HiSoftware for an unlock code. Enter your unlock code.

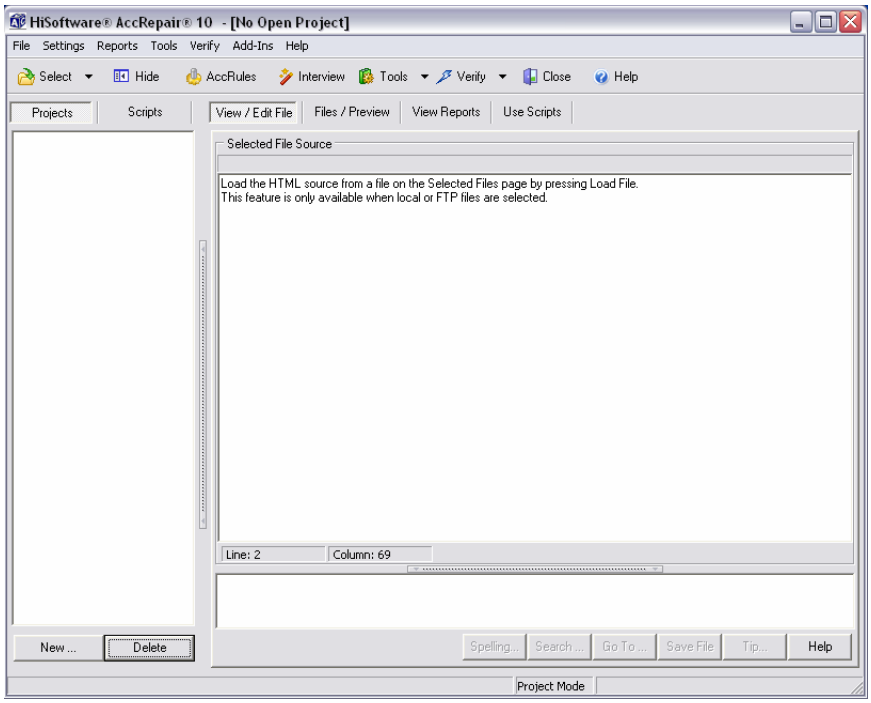
**HiSoftware**
**Ph:** 603-578-1870
**Email:**
support@hisoftware.com

*Note: HiSoftware phones are available from 8am to 4pm Eastern Time Only.*

44. When ready, click on the **OK** button.



**The Main Menu displays.**



| | |
|---|---|
| **Additional Tools** | The following information is available at Tom Jewett Web Design/Section 508 Manual Evaluation Tools (http://www.tomjewett.com/accessibility/508-eval-tools.html). He recommends the following developer tools for assistance in your manual evaluation of accessibility: |
| **1. Firefox:** | Download the Firefox browser for free at Mozilla Firefox Download (http://www.mozilla.com/en-US/firefox/). |
| **2. FF/Developer:** | To download the Firefox web development toolbar, open your Firefox browser, then open the add-ons page at Mozilla Firefox Add-ons (https://addons.mozilla.org/en-US/firefox/addon/60) and click on **Install now.** (It installs automatically.) Then close and reopen Firefox to finish the installation. (Note: there are similar toolbars for Internet Explorer, but the Firefox version seems to be the most complete and easiest to use.) |
| **3. FF/Illinois:** | To download the University of Illinois Accessibility Extension, open your Firefox browser, and then go to Firefox Accessibility Extension (http://firefox.cita.uiuc.edu/). This is very similar to the toolbar listed above, but contains a few functions that are available nowhere else. |
| **4. FF/Fangs:** | Install the Fangs text-reader simulation for Firefox like the developer toolbar above, from Sourceforge at SourceForce Fangs (http://sourceforge.net/projects/fangs). To use, right-click the page and select **View Fangs** from the context menu. You may have to click on the **Reload** button to bring up the voice-formatted text. |
| **5. FF/Firevox:** | Charles Chen's free page reader, available at FireVox Downloads (http://firevox.clcworld.net/downloads.html). |
| **6. IE/AIS:** | The Microsoft Internet Explorer Accessible Information Solutions (AIS) toolbar is available from Vision Australia at Web Accessibility Toolbar (http://www.visionaustralia.org.au/info.aspx?page=614). Many of its functions open in a new window; you will have to disable popup blockers to use them. |

| | |
|---|---|
| **7. Colour Contrast Analyser:** | Use Colour Contrast Analyser version 1.1; (yes, it's UK spelling, and yes, the version has to be 1.1, not 1.0). This is not the same as the one linked from the IE/AIS toolbar. Go to the Vision Australia Download the CCA (http://www.visionaustralia.org.au/info.aspx?page=628#download) and click the first link under **Download**. Save the .zip file and extract it to a convenient place. You may want to make a shortcut to the .exe file on your desktop. |
| **8. Low Vision Stylesheet:** (lowvis.css) | Go to Tom Jewett Web Design/Section 508 Manual Evaluation Tools (http://www.tomjewett.com/accessibility/508-eval-tools.html) and click on lowvis.css. Save to any convenient folder on your computer.<br><br>Note: This low-vision style sheet was developed by accessibility expert ***Dr. Wayne Dick*** for his personal use. (He also uses a voice reader along with it.) It represents just one possible adaptation for low vision; other users will require very different styles or techniques. |
| **9. HiSoftware Link Validation Utility:** | Insert the HiSoftware AccVerify CD provided by the SJSU Help Desk. Double click on the **INSTALL-INSTRUCTIONS.pdf** file and follow instructions for installation. |
| **10. HiSoftware HiCaption:** | Insert the HiSoftware AccVerify CD provided by the SJSU Help Desk. Double click on the **INSTALL-INSTRUCTIONS.pdf** file and follow instructions for installation. |
| **11. Jaws®** | This is the most popular full-screen text reader. It is available in a demo version that runs for 40 minutes at a time before requiring the computer to be re-booted. To download, go to Freedom Scientific JAWS for Windows (http://www.freedomscientific.com/fs_downloads/jaws.asp). |
| **12. Adobe Acrobat** | The ATI Task Force will require you to convert your reports into PDF documents before submitting them. We recommend upgrading to version 8.0 as this version helps you create accessible PDF documents. Contact SJSU Purchasing at 408-924-1560 for special contract pricing. |

## Section 3: Conduct Manual Accessibility Evaluation

This section explains how to manually check a web page for Section 508 compliance and best practices. As you perform each checkpoint evaluation, make a note of whether your web page passes, fails or if the Checkpoint is not applicable. This information will be entered into the AccVerify Interview Wizard in Section 4.

| | |
|---|---|
| **Retrofit to Law — Build to Best Practice** | ***Retrofit to Law*** means that existing web pages must be retrofit for Section 508 compliance.<br><br>***Build to Best Practice*** means that any new web pages you create must be built for Section 508 compliance and must also follow best practices as established by the ATI Task Force. Best Practices are noted in each of the Checkpoints in this section. |
| 1. What are the Section 508 Checkpoints? | • Manual Checkpoints B, C, D, E, G, H, J, K, L, O and P (as described later in this section) are part of Section 508 compliance.<br><br>• Automated Checkpoints A, F, I, M and N (as described later in this section) are part of Section 508 compliance.<br><br>• Additionally, Syntax Validation, Semantic Validation and User Validation (as described later in this section) are part of Section 508 compliance.<br><br>***Important Note: Although HiSoftware AccVerify evaluates the automated checkpoints for you, all Webmasters must still manually review these checkpoints to review warnings and to ensure accuracy.*** |
| 2. What are the Best Practice Checkpoints? | Best Practices are additional requirements (determined by the CSU ATI Task Force) that go beyond Section 508 compliance.<br><br>These additional requirements apply to all new CSU web pages. They consist of ***additional questions*** to some of the Checkpoints as well as additional questions for Syntax Validation, Semantic Validation and User Validation (as described later in this section). |

### Syntax Validation

Contemporary, maintainable, accessible web design is based on complete separation of content (HTML/XHTML) from presentation (CSS) with both conforming to W3C standards; valid code contributes to accurate presentation by both standard browsers and assistive technologies.

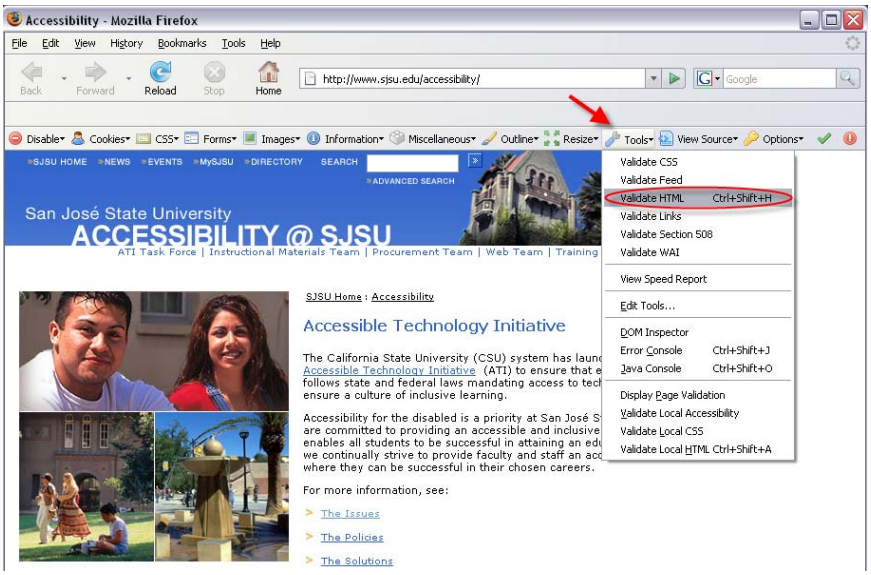
*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

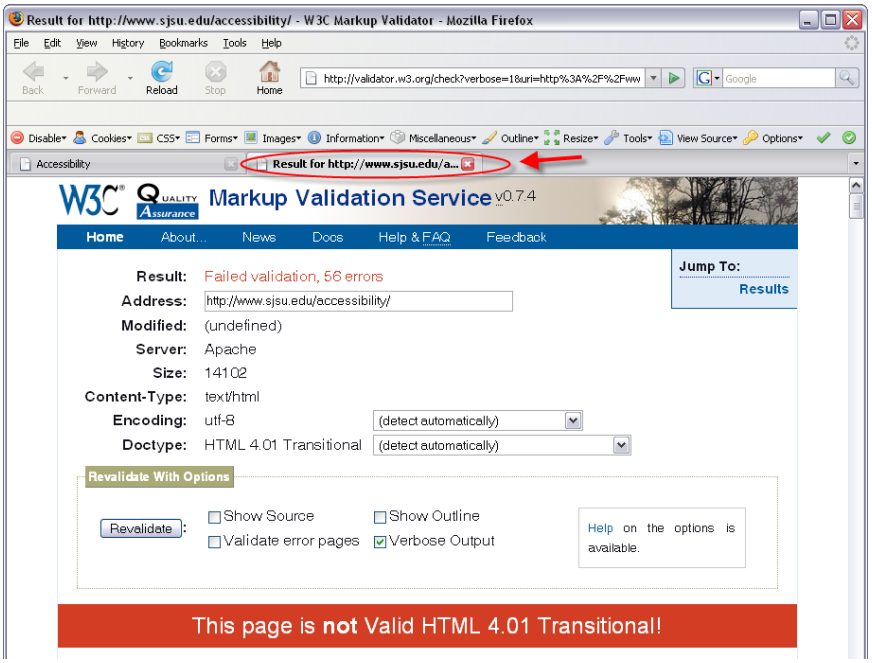
1. **Syntax Question 1:** Is the page valid HTML? (If yes, then pass.)

2. To answer this question, use the **FF/Developer toolbar**.

3. Click on **Tools** > **W3C HTML Validator** > **Validate HTML**.



**The HTML Validator results display**.

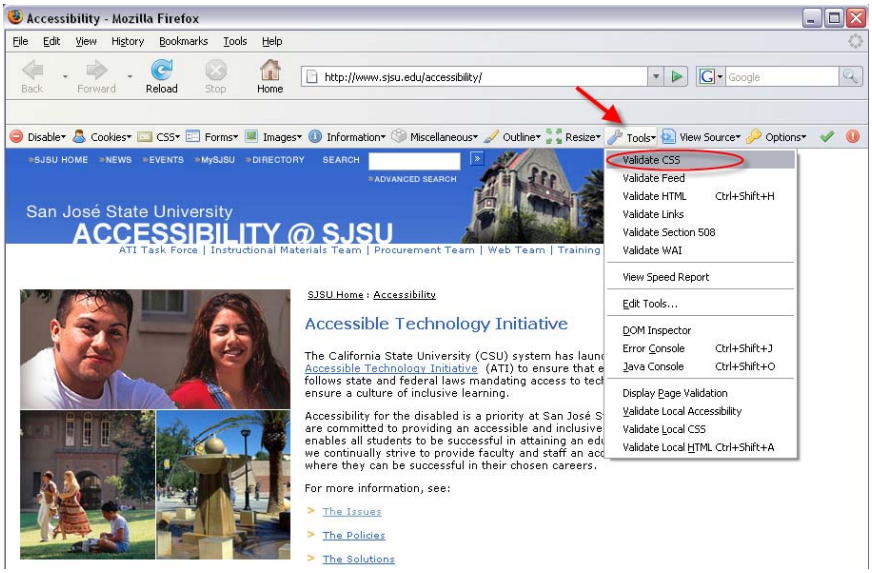



**The web page displays.**

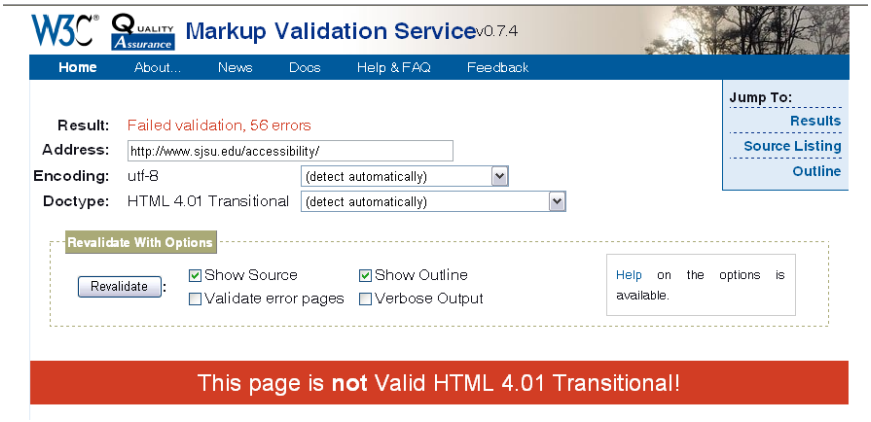
4. **Syntax Question 2:** Does the page use valid CSS? (If yes, then pass.)

5. To answer this question, use the **FF/Developer toolbar**.

6. Click on **Tools** > **W3C CSS Validator** > **Validate HTML**.



**The CSS Validator results display.**




**The web page displays.**

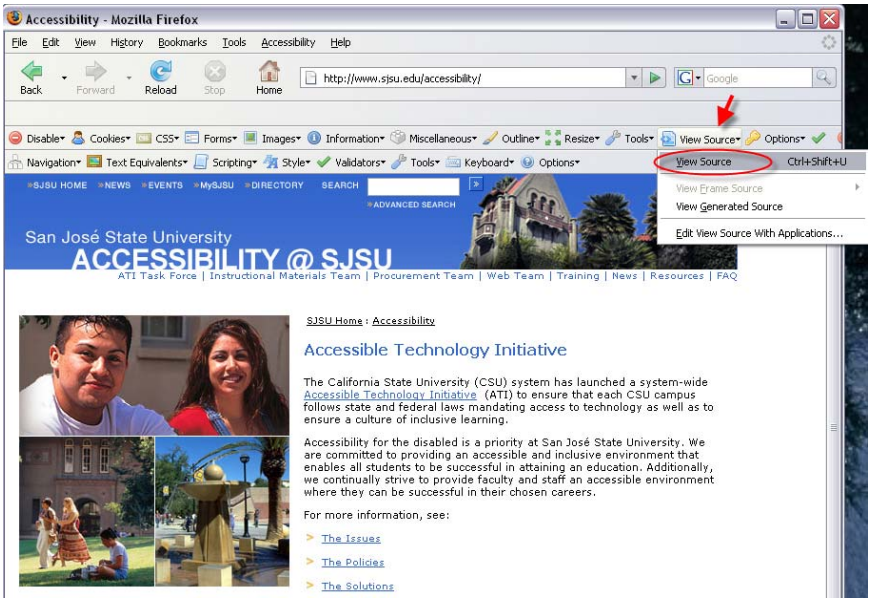
7. **Syntax Question 3 (Best Practice):** Best Practice: Is the CSS entirely separate from HTML code? (If yes, then pass.)

8. **Syntax Question 4 (Best Practice):** Is the JavaScript entirely separate from HTML and CSS? (If yes, then pass.)

9. To answer these questions, use the **FF/Developer toolbar**.

10. Click on **View Source** > **View Source**.

*Note: As an alternate, use the IE Menu and click on View > Source.*

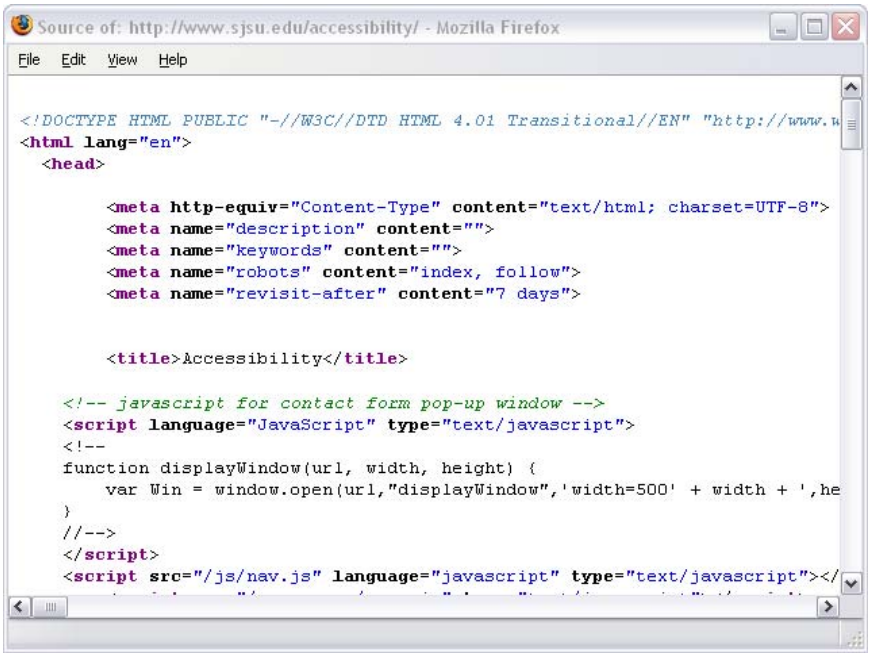


**The source code displays.**

11. For question 3, look for "style=".

12. For question 4, look for script type="Text/javascript"

*References: Web Accessibility, pages 86–89 and Chapter 9, CSS for accessible web pages.*

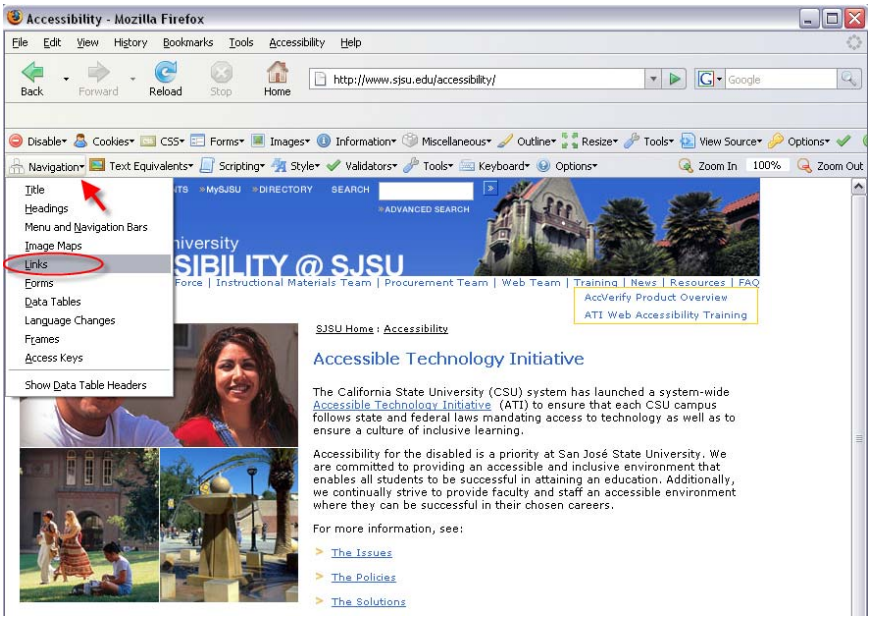


### Semantic Validation

Page elements that function as links (including text, images and buttons) should be recognizable and usable by sighted viewers and assistive technology. All visitors are helped by logically organized page content that uses section headings and subheadings in their proper order. Visitors using assistive technology can use the headings for easy navigation between sections of the page.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**




1. **Semantic Question 1:** Does the text of each link describe where the link goes? (If yes, then pass.)

2. **Semantic Question 2:** Do any links with the same text point to different places? (If no, then pass.)

3. To answer these questions, use the **FF/Developer toolbar**.

4. Click on **Navigation** > **Links**.

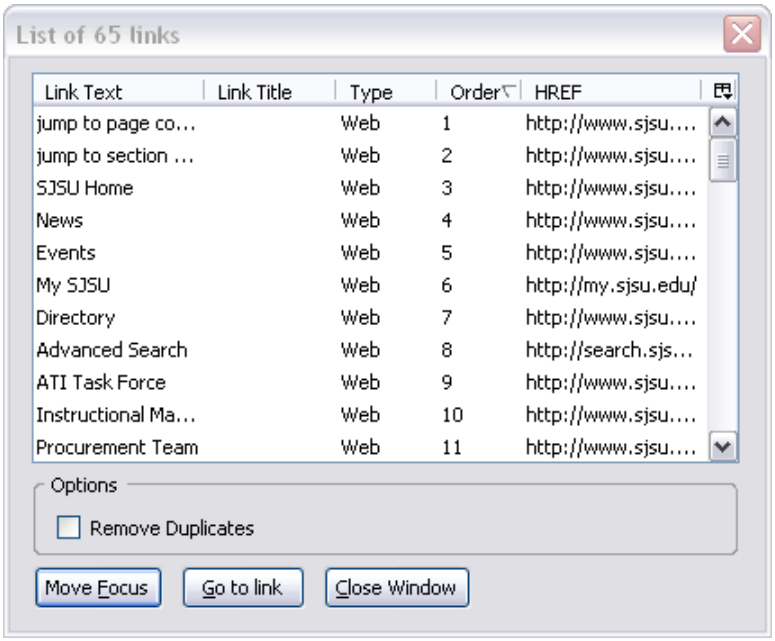
**The list of links displays.**

5. For question 1, click on the **HREF column** to put list in order.

6. For question 2, click on **Link Text** column to put list in order.

7. Check the Remove Duplicates box. There should be no remaining links with the same name. If there are, it is because they point to different places.

8. When done, close window.



**The web page displays.**

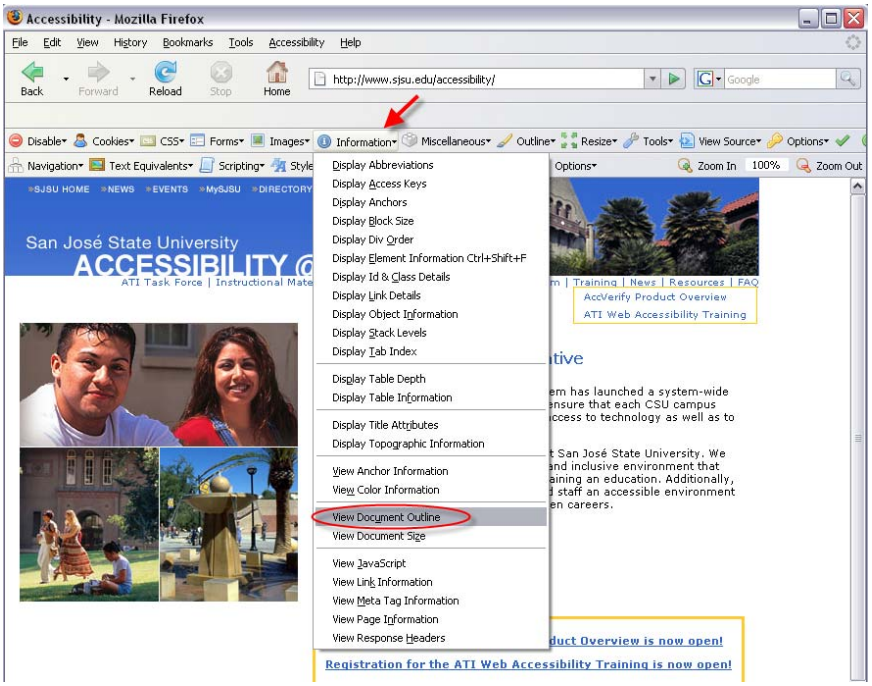
9. **Semantic Question 3 (Best Practice):** Best practice: Are headings properly nested by level? (If yes, then pass.)

10. **Semantic Question 4 (Best Practice):** Best practice: Do headings accurately reflect the structure of the document content? (If yes, then pass.)

**11.** To answer these questions, use the **FF/Developer toolbar.**

12. Click on **Information** > **View Document Outline**.

*Note: As an alternative, use the IE/AIS toolbar to click on Structure > Headings.*




**The document outline displays.**

*Reference: Web Accessibility, pages 206–210.*

**Checkpoint A**

Assistive technologies such as screen readers can't read images; neither can search engines. Some sighted visitors might have images disabled for various reasons, or images might fail to load over a network. If alt-text is not given for images used as links, screen readers will read the whole URL instead, which is horribly confusing for listeners! The title= attribute (best practice) is used in Firefox for mouseover (alt= is used in IE); many low-vision users benefit from having this available.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

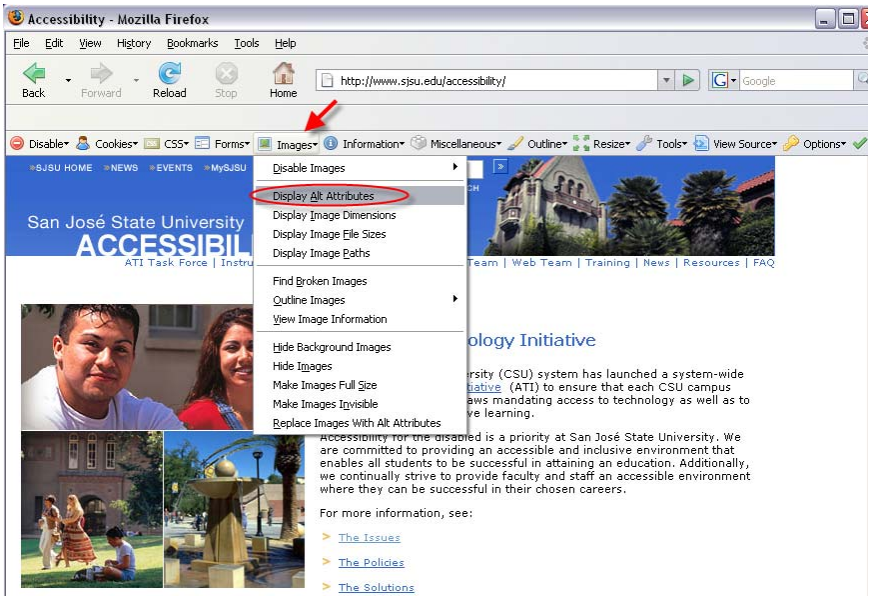
**The web page displays.**

1. **Question A1:** Do images that convey content have equivalent alt-text? (If yes, then pass.)

2. To answer this question, use the **FF/Developer toolbar**.

3. Click on **Images** > **Display Alt Attributes**.

*Note: As an alternative, use the IE/AIS Toolbar and click on **Images** > **Toggle Image/Alt**.*



**The Alt Attributes display.**

4. **Question A2:** Do purely decorative images have empty alt-text (alt="")? (If yes, then pass.)

5. When done, turn off alt attributes by reloading the page.

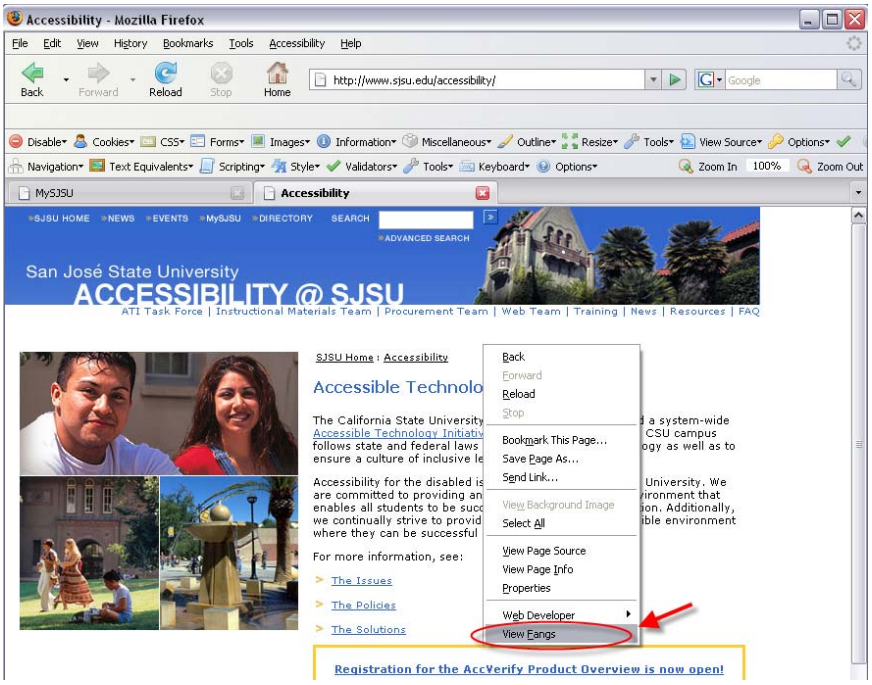
*Note: When retrofitting, include title= as well as alt= attribute.*

**The web page displays.**

6. **Question A3:** Does alternate text make sense in the context of the page as spoken? (If yes, then pass.)

7. **Question A4:** Do images that convey complex content have longdesc attributes or equivalent text content elsewhere on the page? (If yes, then pass.)

8. To answer these questions, right click anywhere on the page and select **View Fangs**.

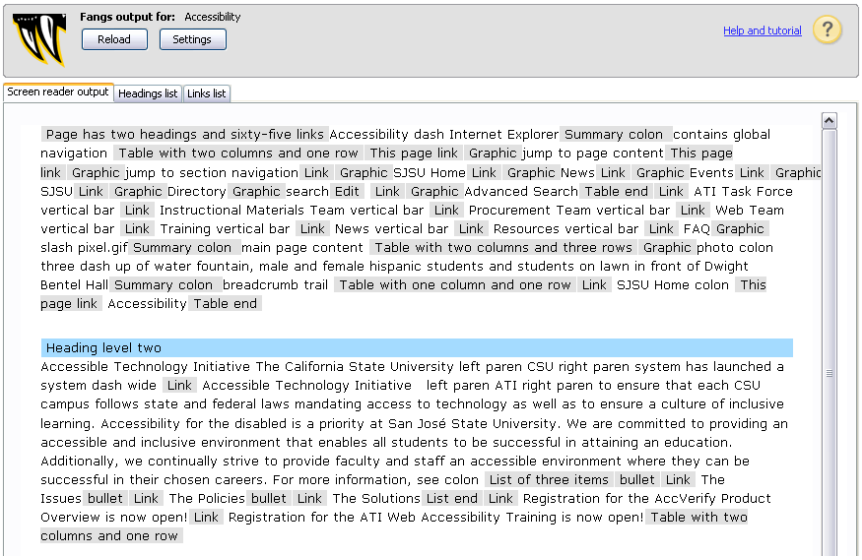


**The Fangs window displays.**

9. If window is blank, then click on the **Reload** button.

**The Fangs window displays.**

10. When done, close Fangs window.

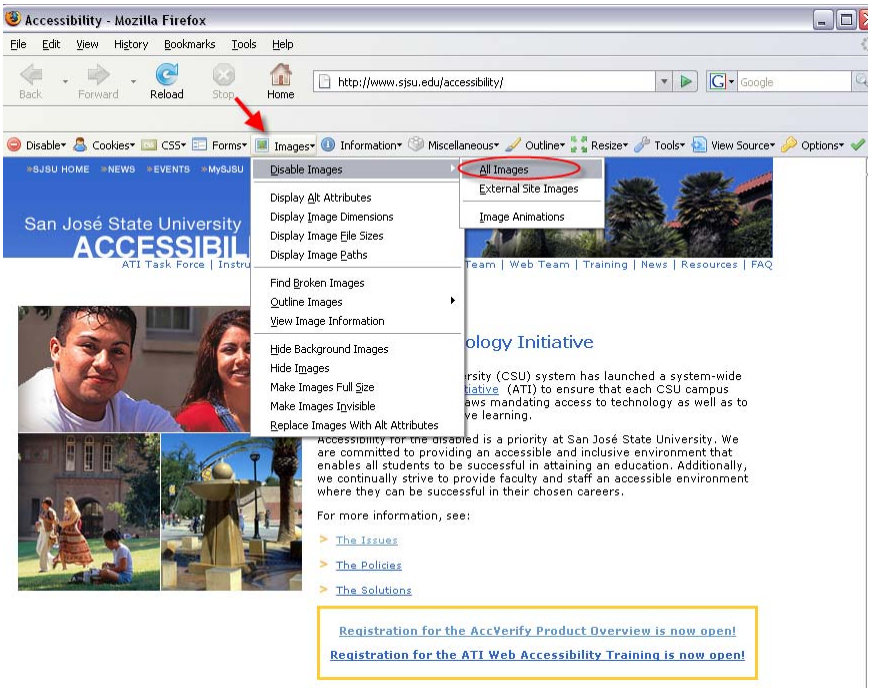


**The web page displays.**

11. **Question A5:** Does text content contained in images disappear when images are not available? (If no, then pass.)

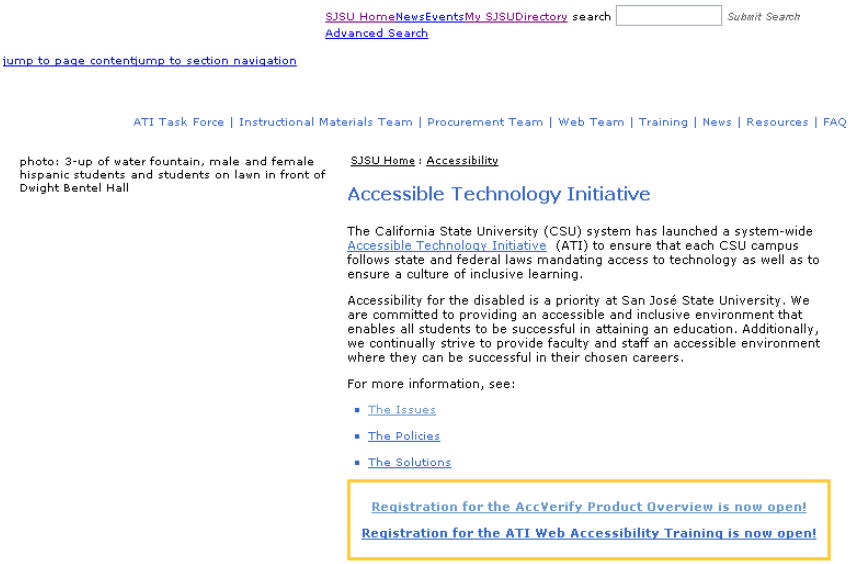
12. To answer this question, use the **FF/Developer toolbar**.

13. Click on **Images** > **Disable Images** > **All Images**.




**The web page displays without images.**

14. When done, turn images back on by closing the tab or reloading the page.

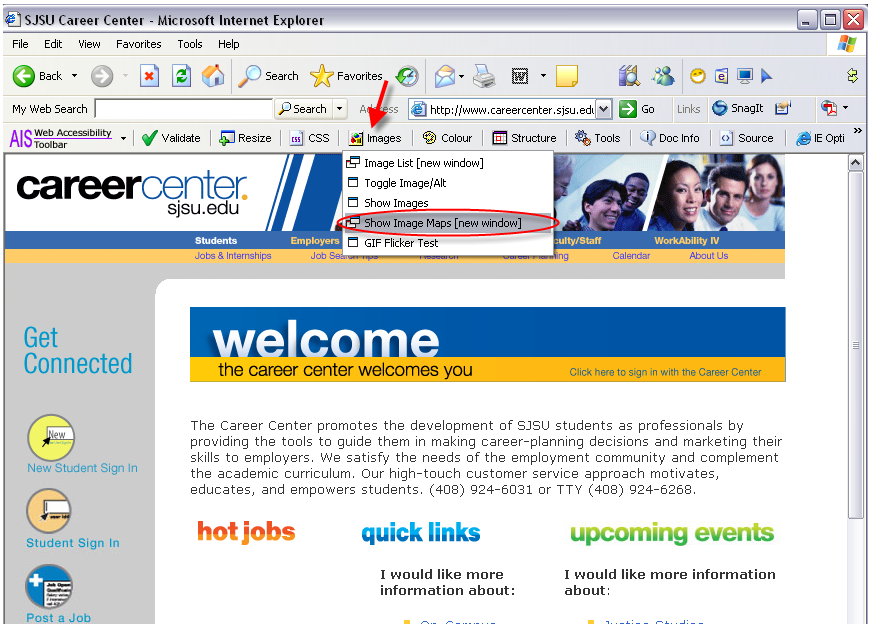


**The web page displays.**

15. **Question A6:** Does image map area alt-text describe the link destination correctly? (If yes, then pass.)

16. To answer this question, use the **IE/AIS toolbar**.

17. Click on **Images** > **Show Image Maps**.

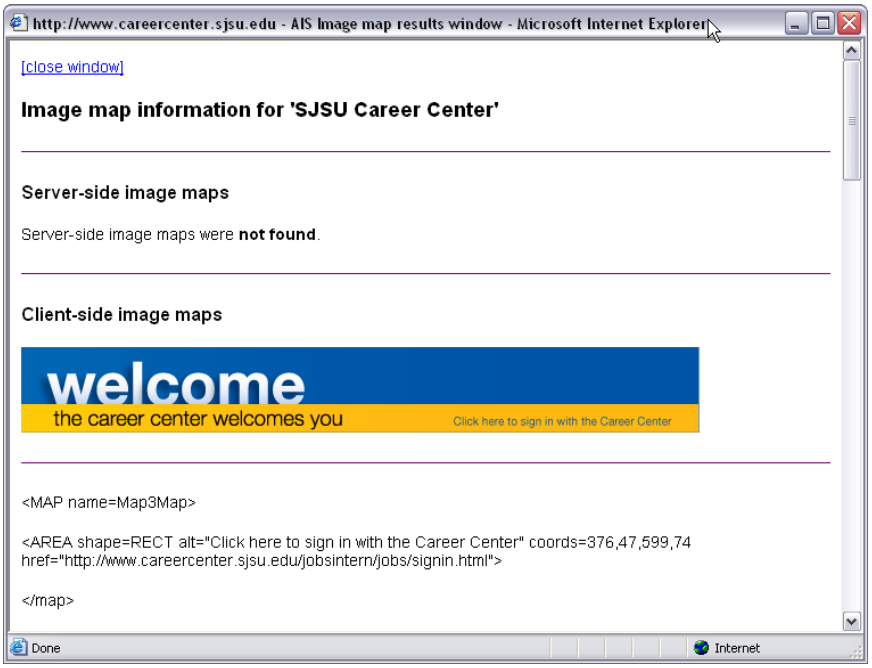



**A new window opens.**

18. When done, close window.

*Note: When retrofitting, include title= as well as alt= attribute.*

*Reference: Web Accessibility, see **text equivalents** in index.*



### Checkpoint B

### Multimedia Equivalents

For a multimedia presentation, a visual equivalent of audio content is needed as well as an audio equivalent of visual content. In addition, most assistive technologies are unlikely to have plug-ins that can deal with multimedia formats.

Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).

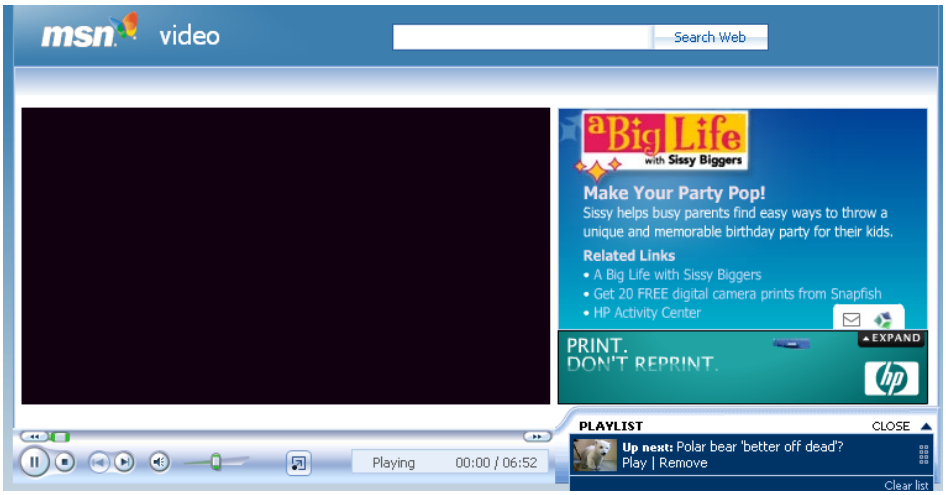
**The web page displays.**

1. **Question B1:** Is captioning provided for all spoken content? (If yes, then pass.)

2. To answer this question, make a visual and audio inspection of any multimedia on your web page.

3. Note whether your web page passes, fails, or is not applicable for Checkpoint B.

*Reference: Web Accessibility, Chapter 11, Accessible Flash.*



**Checkpoint C**

**Color**

Visitors with both full vision and various forms of color blindness require sufficient contrast to read text easily. If images are used as background, contrast may change when they are unavailable. Color-blind users may be unable to distinguish information conveyed by color without alternate coding of that information; blind users obviously cannot use color information at all.

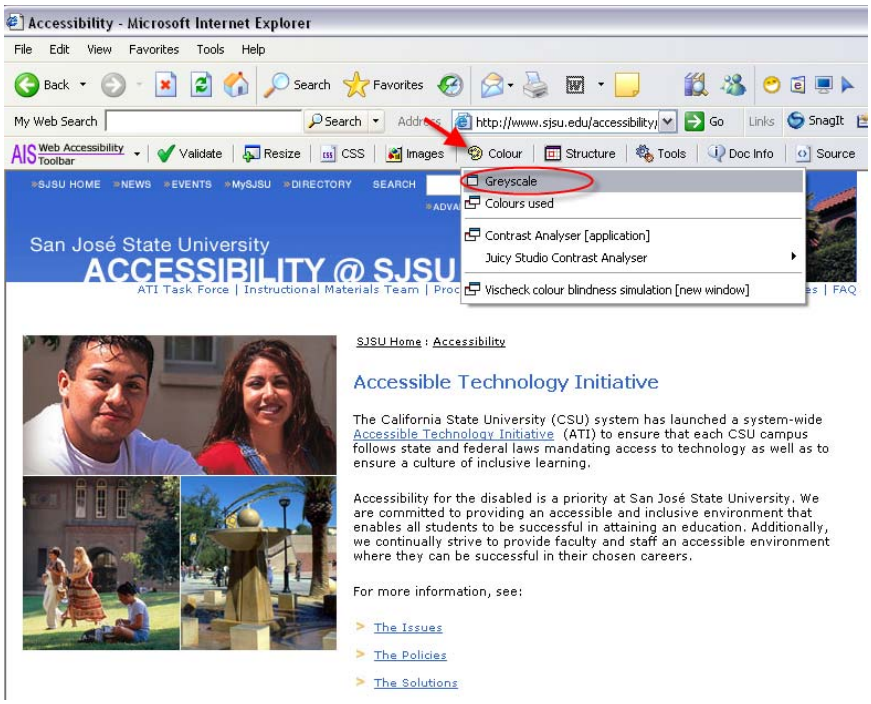
*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

1. **Question C1:** Is information conveyed by color also conveyed by context, markup, graphic coding, or other means? (If yes, then pass.)

2. To answer this question, use the **IE/AIS Toolbar**.

3. Click on **Colour** > **Greyscale**.



**The web page displays in greyscale.**

4. When done, turn off greyscale by refreshing the page.

**The web page displays.**

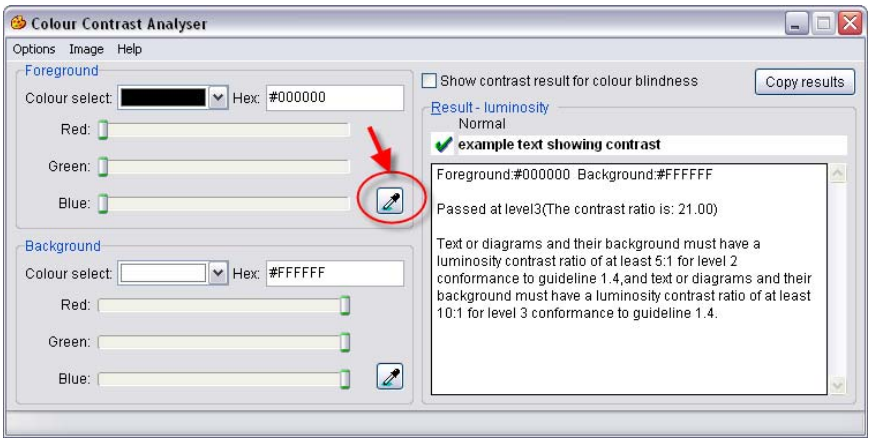
5. **Question C2:** Does the contrast between foreground and background colors have a luminosity contrast ratio of at least 5:1 for both normal vision and color blindness? (If yes, then pass.)

6. To answer the question, open the **Colour Contrast Analyser** tool.

**The Colour Contrast Analyser displays.**

7. Click on the foreground eyedropper icon.

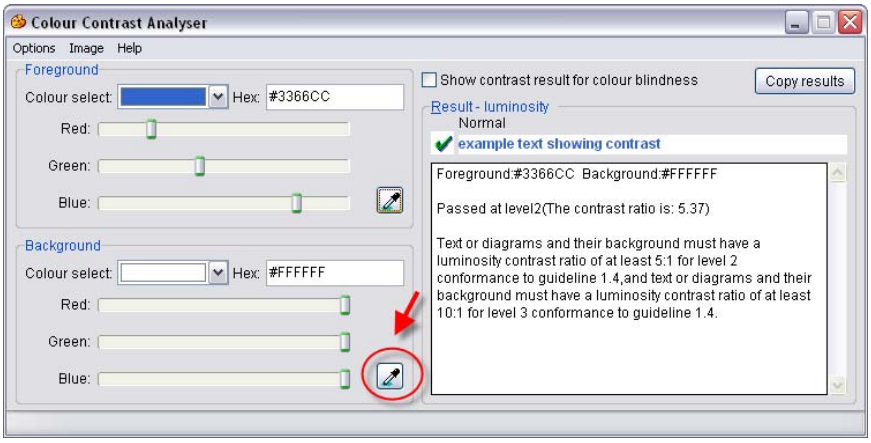
8. Go to web page and select foreground color.



**The foreground color displays.**

9. Click on background eyedropper icon.

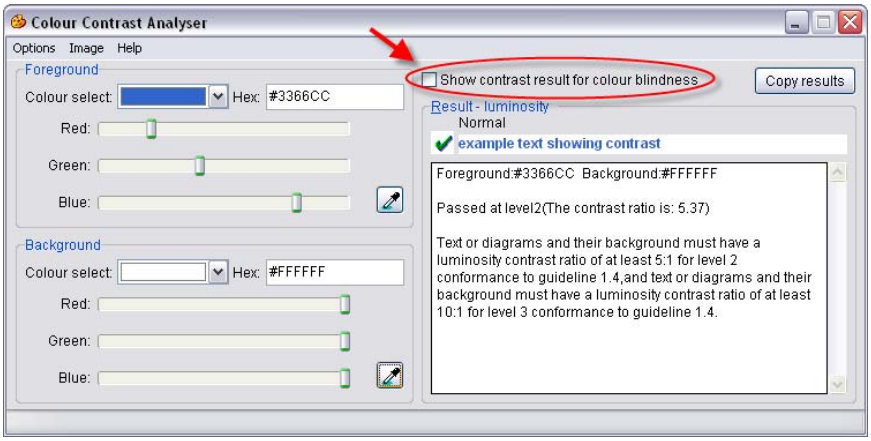
10. Go to web page and select background color.




**The background color displays.**

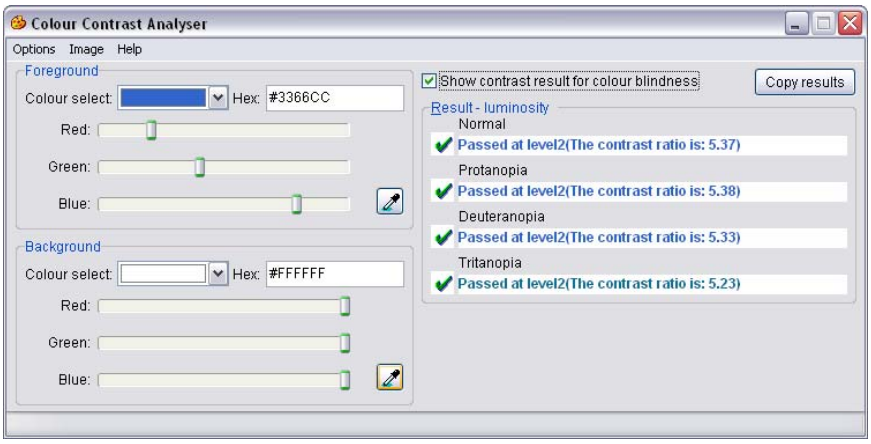
11. Review results in window.

12. Select **Show contrast result for colour blindness**.




**The color blind results display.**

13. When done, close **Colour Contrast Analyser** window.

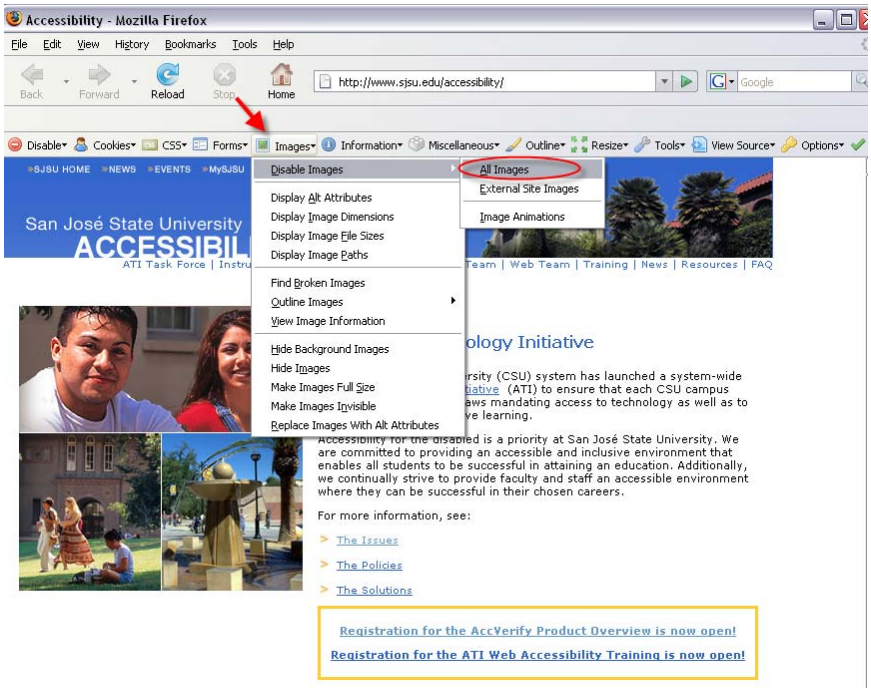


**The web page displays.**

14. **Question C3:** Is a correct contrast ratio maintained when images are not available? (If yes, then pass.)

**15.** To answer this question, use the **FF Developer toolbar**.

**16.** Click on **Images** > **Disable Images** > **All Images**.

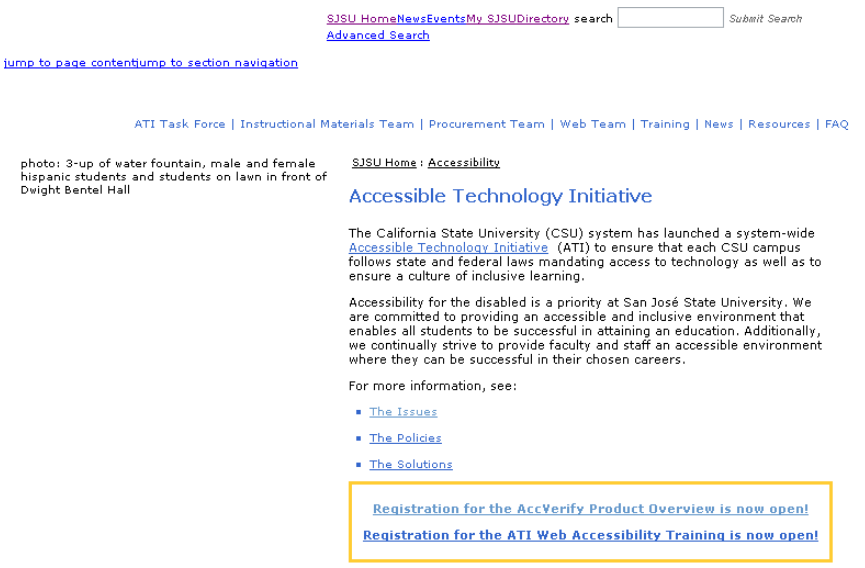



**The web page displays without images.**

17. Note whether your web page passes, fails, or is not applicable for Checkpoint C.

**18.** When done, turn off Disable Images by closing the tab or reloading the page.

*Reference: Web Accessibility, see **Color** in index.*



**Checkpoint D**

**Styles**

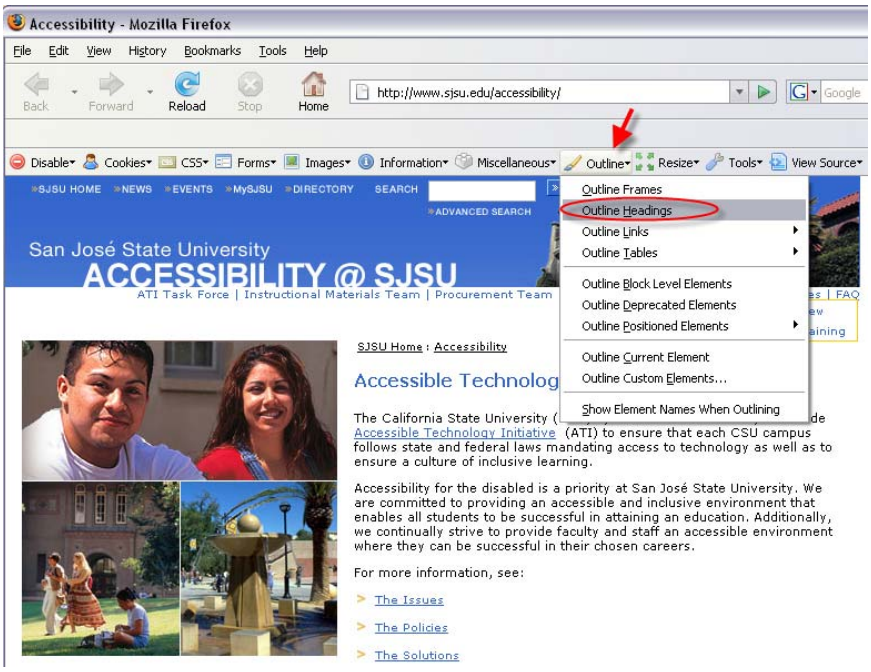
Many low vision visitors will disable the page author's styles, and may substitute a custom style sheet of their own. If styles are used to substitute for semantic markup, low-vision and text-only readers will be unable to understand the structure of the page. (Example: large bold text used instead of a heading.) Elements that are hidden from sighted users will also be hidden from low-vision users who may need to see them; badly hidden element may be inaccessible to all users.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

1. **Question D1:** Are styles used to simulate headings or other semantic markup? (If no, then pass.)

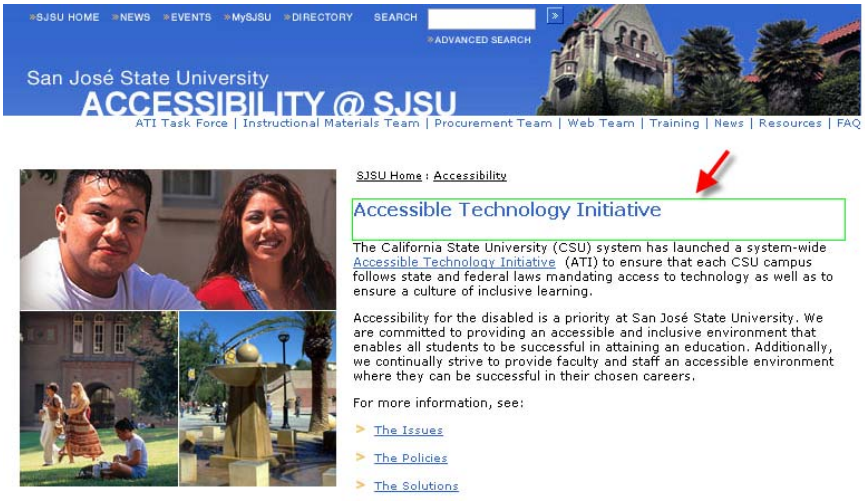
2. To answer this question, use the **FF/Developer toolbar**.

3. Click on **Outline** > **Outline Headings**.




**The web page displays with headings outlined.**

4. When done, turn off Outline Headings by closing the tab or reloading the page.



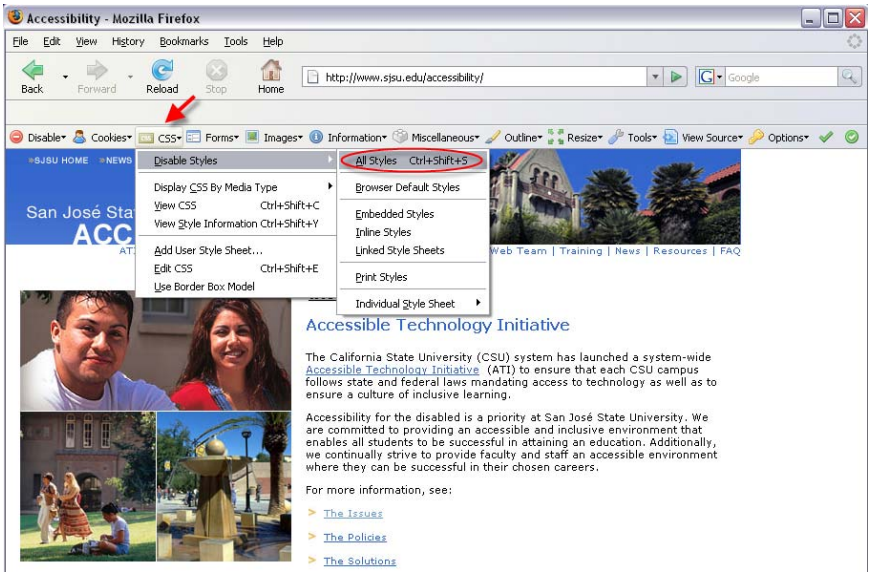
**The web page displays.**

5. **Question D2:** With all styles disabled, is color and font information rendered in the browser's default style? (If yes, then pass.)

6. To answer this question, use the **FF/Developer toolbar**.

7. Click on **CSS** > **Disable Styles** > **All Styles**.

*Note: As an alternate, use the **IE/AIS toolbar** and click on **CSS** > **Disable CSS**.*

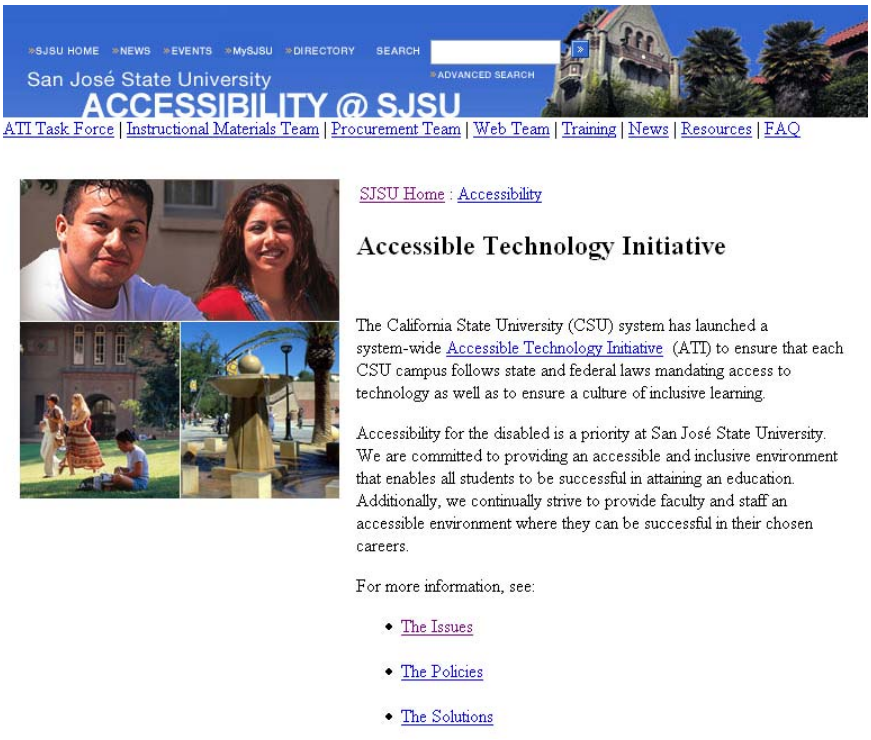



**The web page displays without styles.**

8. **Question D3:** With all styles disabled, are headings, paragraphs and lists obvious and sensible? (If yes, then pass.)

9. **Question D4:** With all styles disabled, does the order of the page content make sense as read? (If yes, then pass.)

10. To answer these questions, leave styles disabled.



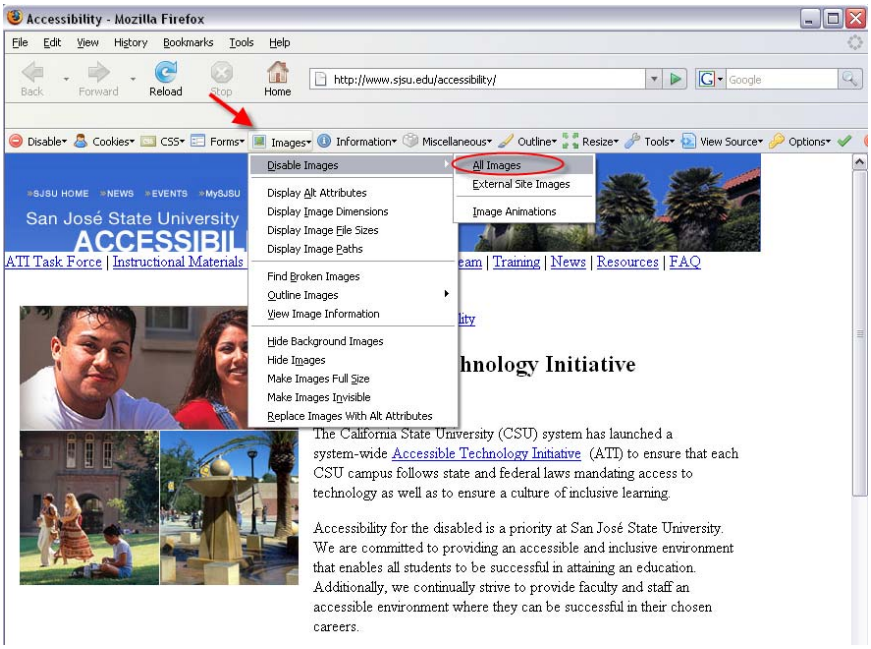
**The web page displays without styles.**

11. **Question D5 (Best Practice):** With all styles disabled, is most text (other than logos and banners) rendered in text rather than images? (If yes, then pass.)

12. To answer this question, leave the styles disabled and use the **FF/Developer toolbar** to click on **Images** > **Disable Images** > **All Images**.

**The web page displays with both styles and images disabled.**

**13.** When done, turn images back on by closing the tab or reloading the page.



**The web page displays with styles disabled.**

14. **Question D6 (Best Practice):** With all styles disabled, does any content appear that was invisible before? (If no, then pass.)

15. To answer this question, leave the styles disabled.

16. Note whether your web page passes, fails, or is not applicable for Checkpoint D.

17. **When done, turn styles back on by closing the tab or reloading the page.**

*Reference: Web Accessibility, Chapter 9, CSS for Accessible Web Pages.*

ATI Task Force | Instructional Materials Team | Procurement Team | Web Team | Training | News | Resources | FAQ

SJSU Home : Accessibility

## Accessible Technology Initiative

The California State University (CSU) system has launched a system-wide Accessible Technology Initiative (ATI) to ensure that each CSU campus follows state and federal laws mandating access to technology as well as to ensure a culture of inclusive learning.

Accessibility for the disabled is a priority at San José State University. We are committed to providing an accessible and inclusive environment that enables all students to be successful in attaining an education. Additionally, we continually strive to provide faculty and staff an accessible environment where they can be successful in their chosen careers.

For more information, see:

- The Issues
- The Policies
- The Solutions

**Checkpoint E**

**server-side image maps**

This checkpoint was written at a time when server-side image maps were already becoming obsolete.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

1. **Question E1:** Does this page use server-side image maps? (If no, then pass.)

2. To answer this question, use the **IE/AIS Toolbar**.

3. Click on **Images** > **Show Image Maps**.

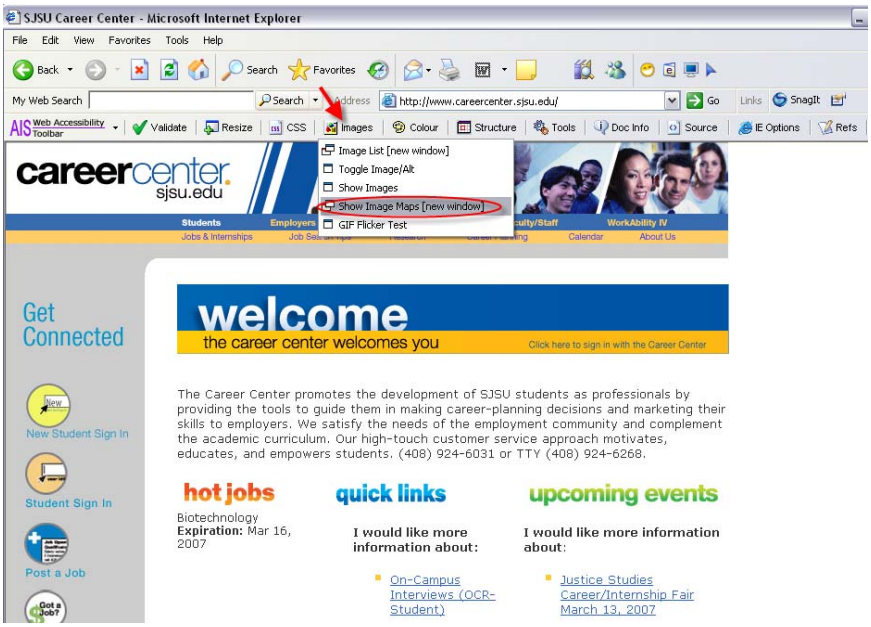
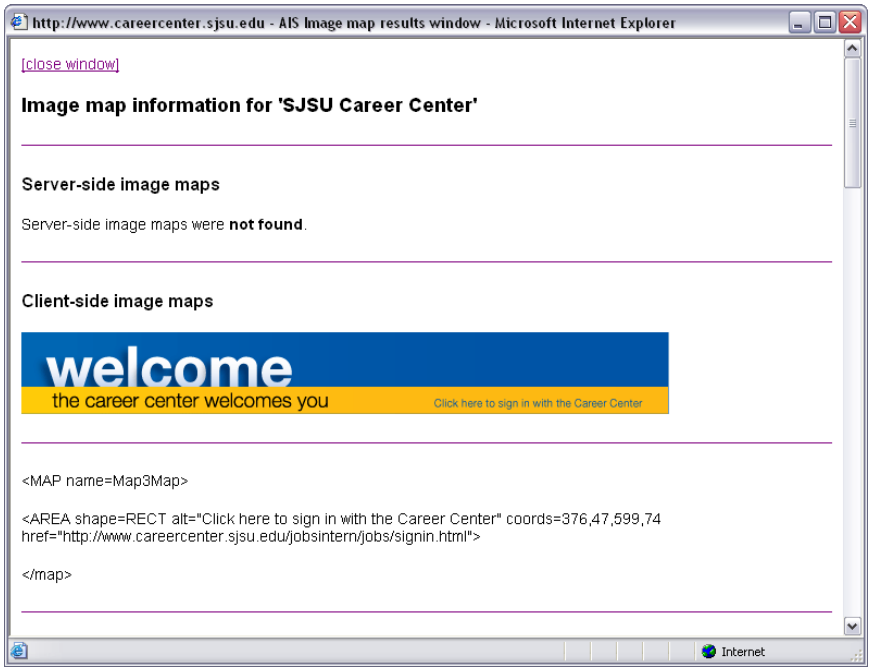



**Image Map information opens in a new window.**

4. Note whether your web page passes, fails, or is not applicable for Checkpoint E.

5. When done, close window.

*Note: When retrofitting, replace server-side image maps. There is no reason to use these today, despite the exception allowed in checkpoint F.*



**Checkpoint F**

Image maps are designed for fully-sighted readers who can also use a mouse. Each clickable area must be identified with its link target, and must be reachable via its tab order, just like all other links. Frequently, an alternate means of selection may prove easiest to use for all visitors. (Example: a drop-down list of states in addition to an image map of the country.)

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

1. **Question F1:** Does image map area alt-text describe the link destination correctly? (If yes, then pass.)

2. To answer this question, use the **IE/AIS toolbar**.

3. Click on **Image** > **Show Image Maps**.

*Note: As an alternate, use the FF/Illinois toolbar and click on Navigation > Image Maps.*

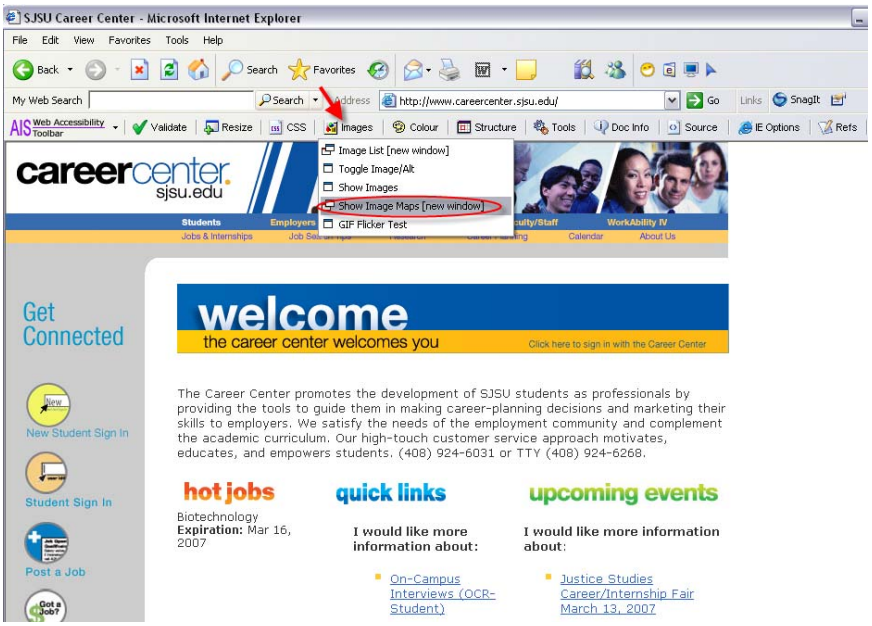
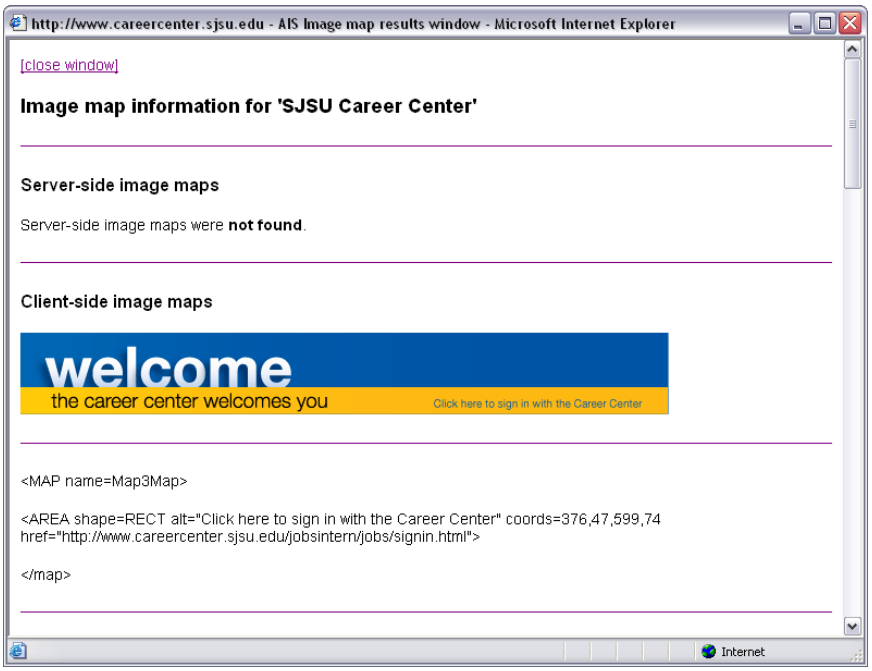


**Image Map information opens in a new window.**

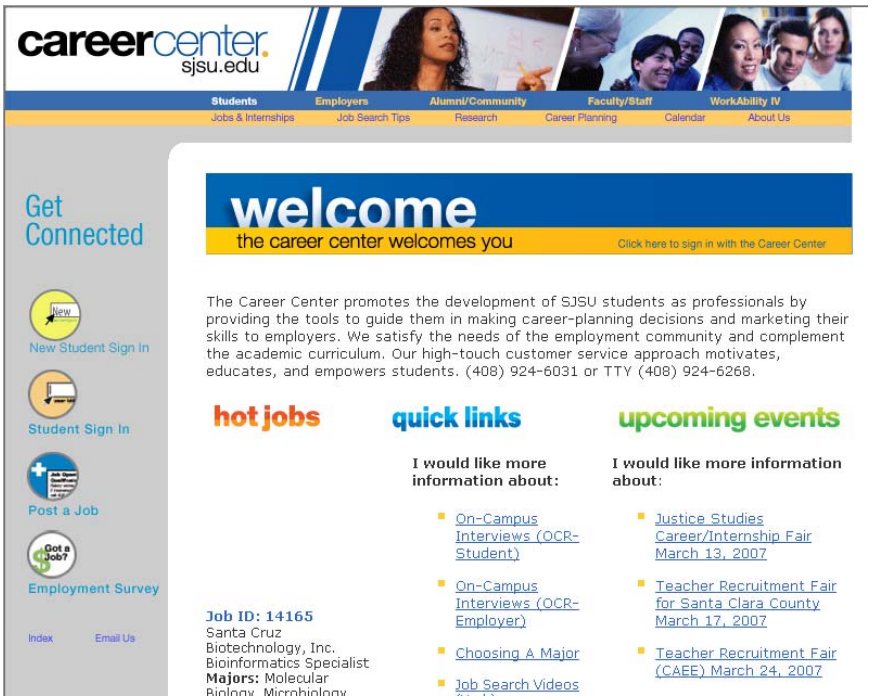
**4.** When done, close window.




**The web page displays.**

5. **Question F2 (Best Practice):** Is there a non-graphic alternative to the image map, for example a drop-down selection control? (fix no)

6. To answer this question, make a visual inspection of your web page.



**Checkpoint G**

**Simple Tables**

Simple data tables are those consisting of only one column header, and possibly one row header, for each data cell. Unless the HTML code explicitly associates each data cell with its column (and row, if applicable) header, visitors using text readers will hear only a string of unintelligible data values. Technically, this is done with the <th scope= ...> element. Text-only visitors also might need assistance in understanding the purpose of the table, which may be provided to them with the summary attribute of the <table> element. On the other hand, text-only visitors should never have to hear summaries such as, "this table is for layout only".

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

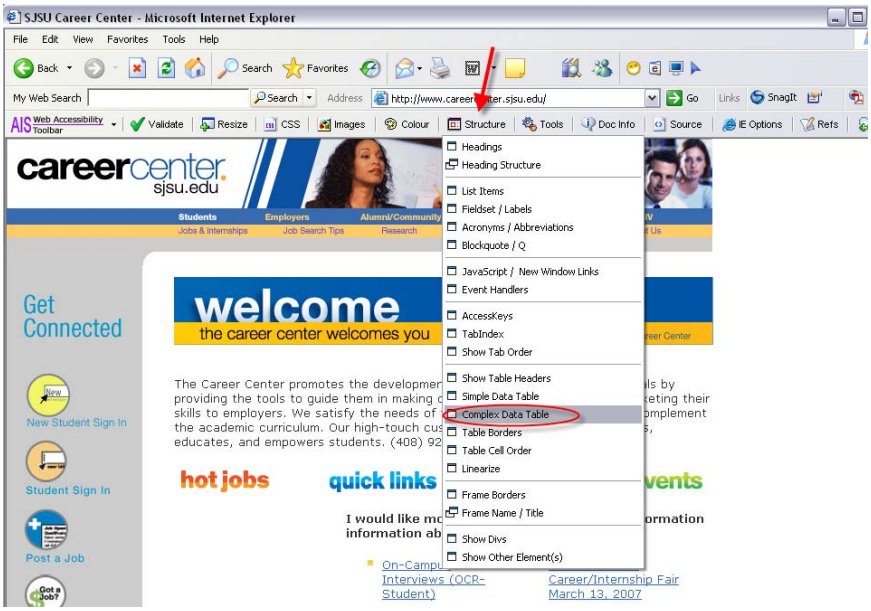
**The web page displays.**

1. **Question G1:** For tables containing data, are <th> elements used in the first row (and first column, if applicable)? (If yes, then pass.)

2. To answer this question, use the **IE/AIS toolbar**.

3. Click on **Structure** > **Simple Data Table.**

*Note: As an alternative, use the **FF/Developer toolbar** to click on **Outline** > **Outline Tables** > **Table Cells**.*



**The web page displays with each table cell labeled with either <td> or <th>.**

4. **Question G2:** Do <th> elements contain the scope= (col or row) attribute? (If yes, then pass.)

5. **Question G3:** For tables that are used for layout only, are <th> elements or the summary= attribute used at all? (If no, then pass.)

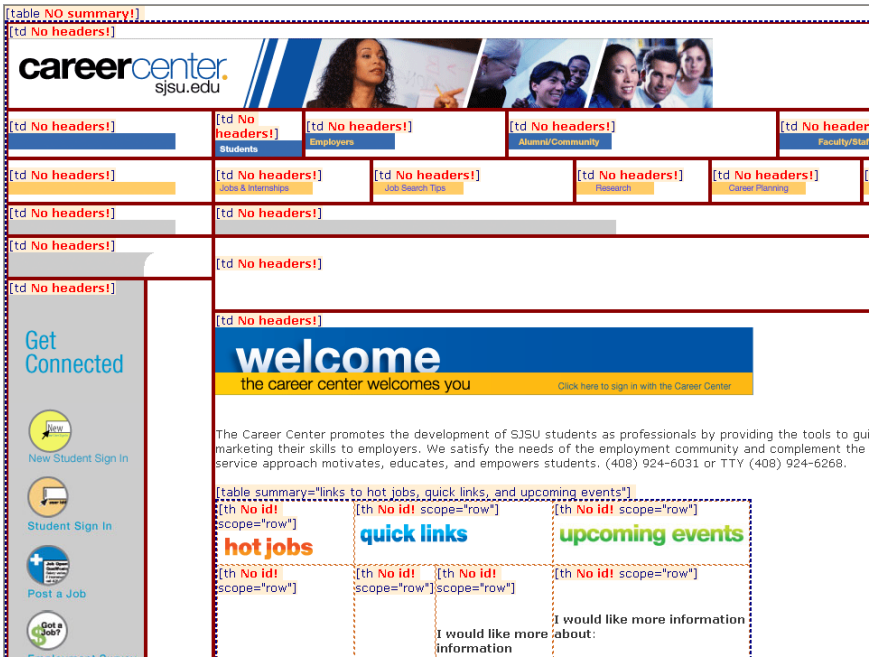
6. **Question G4 (Best Practice):** For tables containing data, is the summary= attribute used to explain the meaning of the table if it is not otherwise evident from context? (If yes, then pass.)

7. Note whether your web page passes, fails, or is not applicable for Checkpoint G.

8. When done, turn off table structure by refreshing the page.

*Note: When retrofitting, don't use tables for layout only.*

*Reference: Web Accessibility, pages 158–173.*



**Checkpoint H**

**Complex Tables**

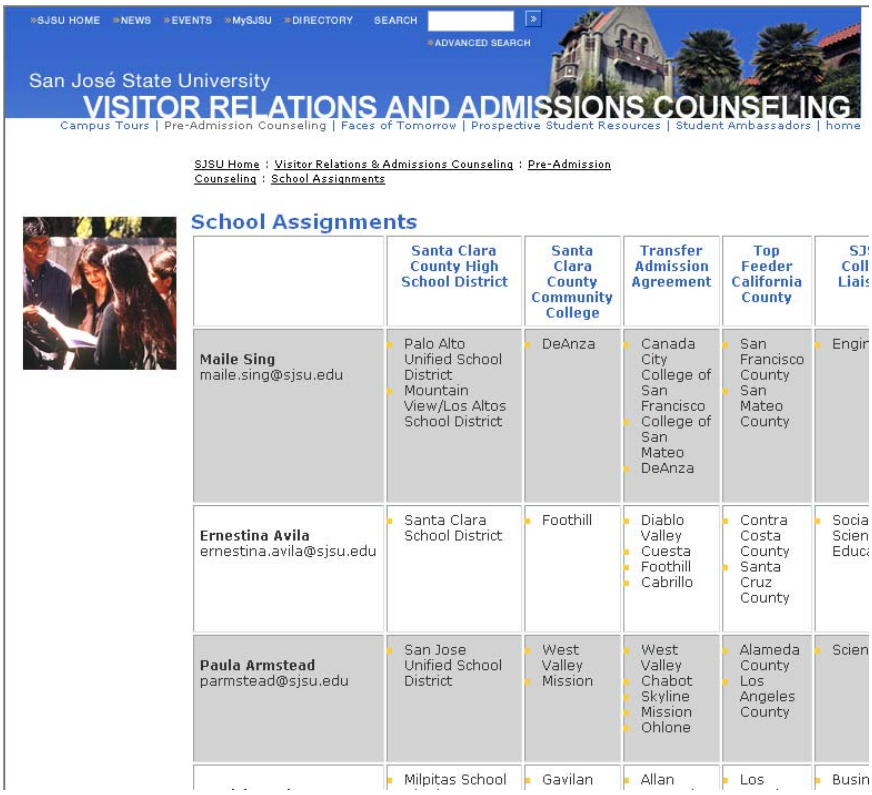
Complex data tables are those in which there is more than one level of header for any column or row. All of the simple table information also applies to these. For text readers to work properly on multiple levels of headings, id= attributes must be added to the <th> elements, and matching headers= attributes to each data cell. Additional markup such as the <thead> element can be used to further clarify the table structure.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

1. **Question H1:** Are all of the Checkpoint G items done correctly? (If yes, then pass.)

2. To answer this question, see Checkpoint G on page 34.



**The web page displays.**

3. **Question H2:** Are <thead> and <tbody> elements used to clarify the table structure? (If yes, then pass.)

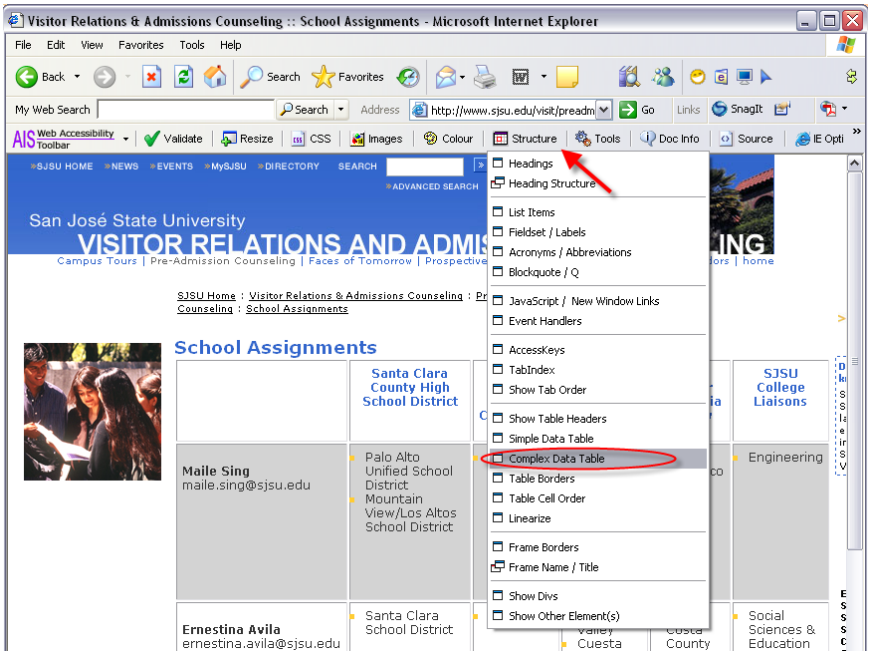
4. **Question H3:** Does each <th> element contain the id= attribute? (If yes, then pass.)

5. **Question H4:** Does each <td> element contain a headers= attribute that associates it with its column and row headers? (If yes, then pass.)

6. To answer these questions, use the **IE/AIS toolbar**.

7. Click on **Structure** > **Complex Data Table**.

*Note: As an alternative, use the **FF/Developer** toolbar to click on **View Source** > **View Source**.*



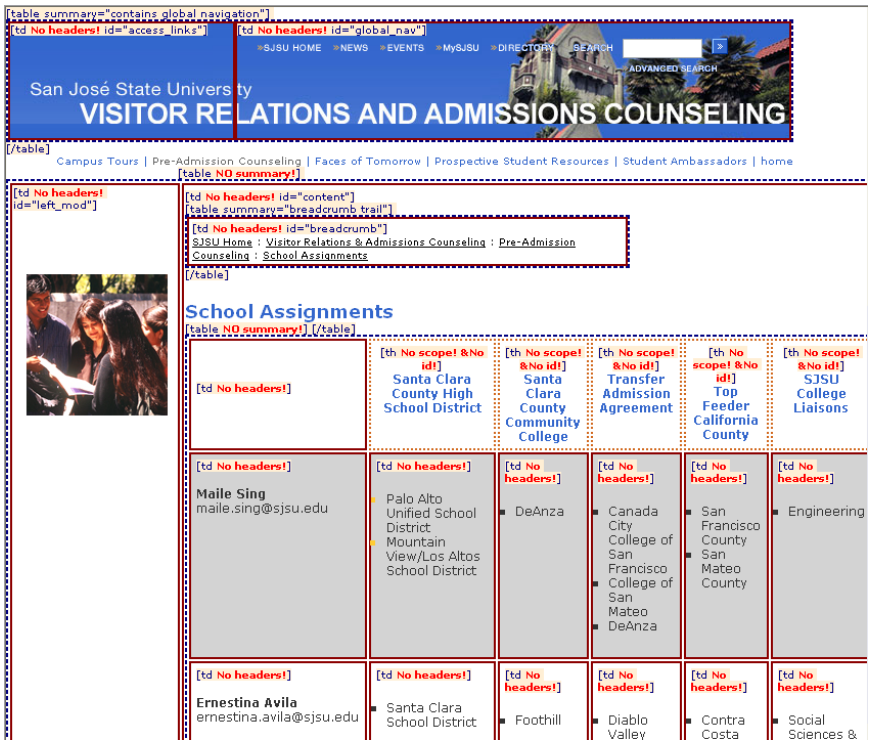
**The web page displays showing table structure.**

8. Note whether your web page passes, fails, or is not applicable for Checkpoint H.

9. When done, turn off table structure by refreshing the page.

*Note: When retrofitting, find a simpler way to present the data to all readers.*

*Reference: Web Accessibility, pages 158–173.*



**Checkpoint I**

Untitled frames cannot be navigated by assistive technology, leaving some visitors unable to reach major portions of the page content. Some assistive technologies might not support frames at all, and it is almost impossible to provide truly equivalent content with the <noframes> element. For these reasons and many others, the use of frames is deprecated by the W3C in favor of element positioning with style sheets.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

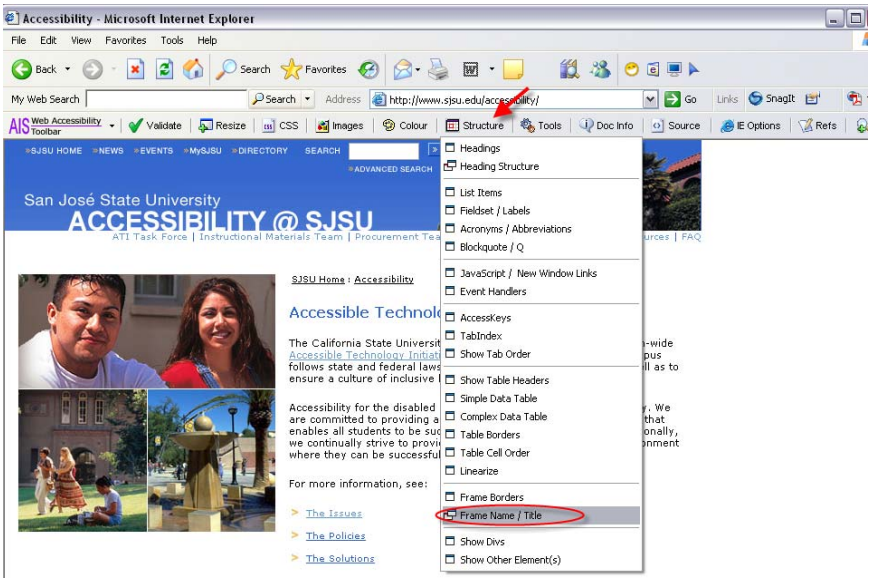
1. **Question I1:** Does each <frame> element have a meaningful title= attribute? (If yes, then pass.)

2. **Question I2:** Does the page have equivalent content in a <noframes> element for user agents that do not support frames? (If yes, then pass.)

3. To answer these questions, use the **IE/AIS toolbar**.

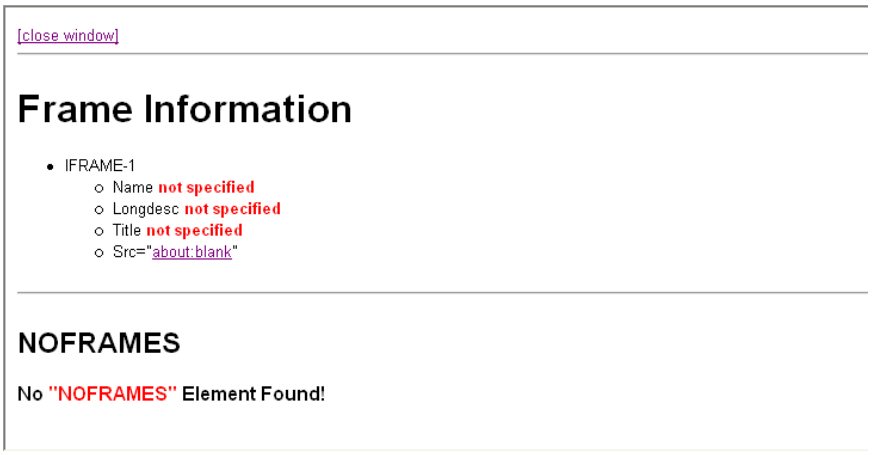
4. Click on **Structure** > **Frames**.

*Note: As an alternate, use the FF/Illinois toolbar and click on Navigation > Frames.*



**The frame information displays.**

*Note: When retrofitting, don't use frames.*



**Checkpoint J**

**Flicker**

At worst, flickering and flashing page elements can cause epileptic seizures in some viewers. They are also disruptive to many readers with cognitive disabilities or low vision. At best, they have become an irritant to nearly everyone. Unfortunately, user agents still do not provide easy-to-use, comprehensive solutions to these problems.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at [Tom Jewett Web Design/Specializing in Accessibility](http://www.tomjewett.com/accessibility/) (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

1. **Question J1:** Does any page element flicker at an unhealthy rate? (If no, then pass.)

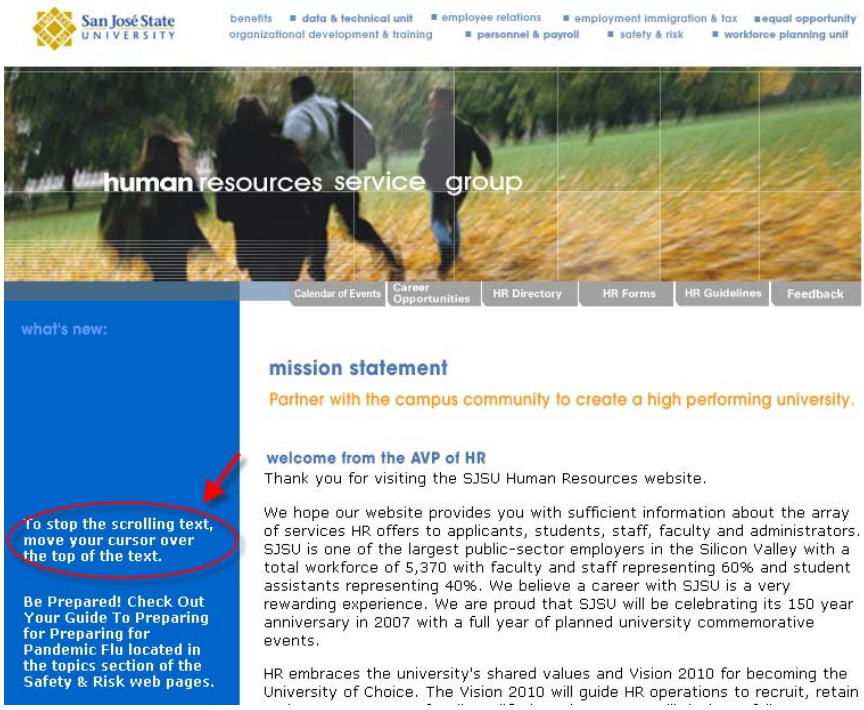
2. **Question J2 (Best Practice):** Does any element on the page move at all? (If no, then pass*.)

3. To answer this question, make a visual inspection of any movement on your web page.

4. Note whether your web page passes, fails, or is not applicable for Checkpoint J.

**Best practice is to have no moving elements at all unless the moving element conveys information relevant to the page content, and then refer to checkpoints B, L, and M.*

*Reference: The glossary of the new WCAG 2.0 Guidelines gives precise definitions of the terms blink, general flash threshold, and red flash threshold.*



**Checkpoint K**

**Text Only**

This checkpoint was written before it became truly feasible to write a single version of a web page that is as accessible to disabled users as it is to the majority. The limitations of text-only pages were well known even at that time, though: they include the difficulty and cost of maintenance and the difficulty of insuring that content is truly equal in different versions of the page. Modern development standards have rendered the text-only page obsolete.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

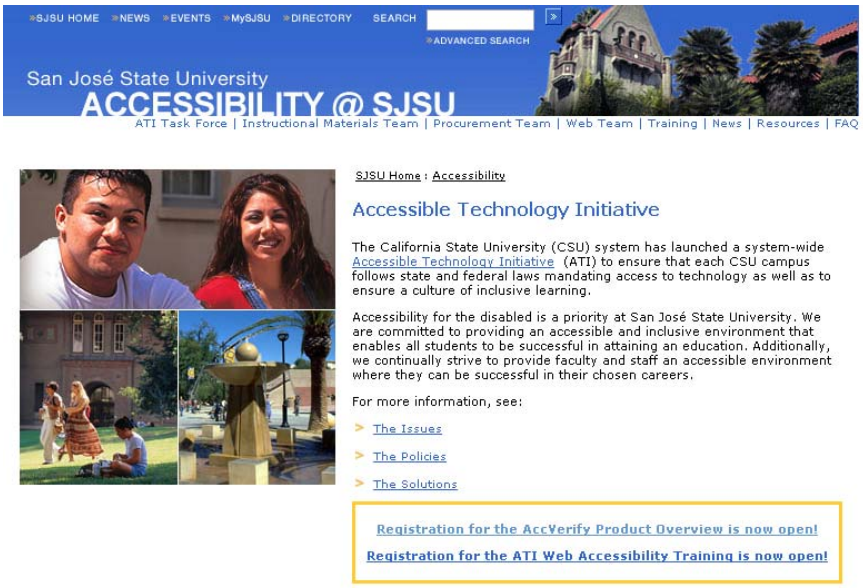
**The web page displays.**

1. **Question K1:** Does this page have a text-only version? (If no, then pass.)

2. To answer this question, make a visual inspection of your web page.

3. Note whether your web page passes, fails, or is not applicable for Checkpoint K.

*Note: When retrofitting, fix by making the page compliant with the rest of the checkpoints There is no reason today to have separate versions, despite the grudging exception contained in this checkpoint.*



**Checkpoint L**

**Scripts**

Many users of assistive technology will not have scripting available; even users of standard browsers may have scripting disabled for security or other reasons. Keyboard-only users will not be able to access functionality that is provided only with mouse-triggered events. The now obsolete "<noscript>" alternative is almost never able to provide truly equivalent content and functionality.

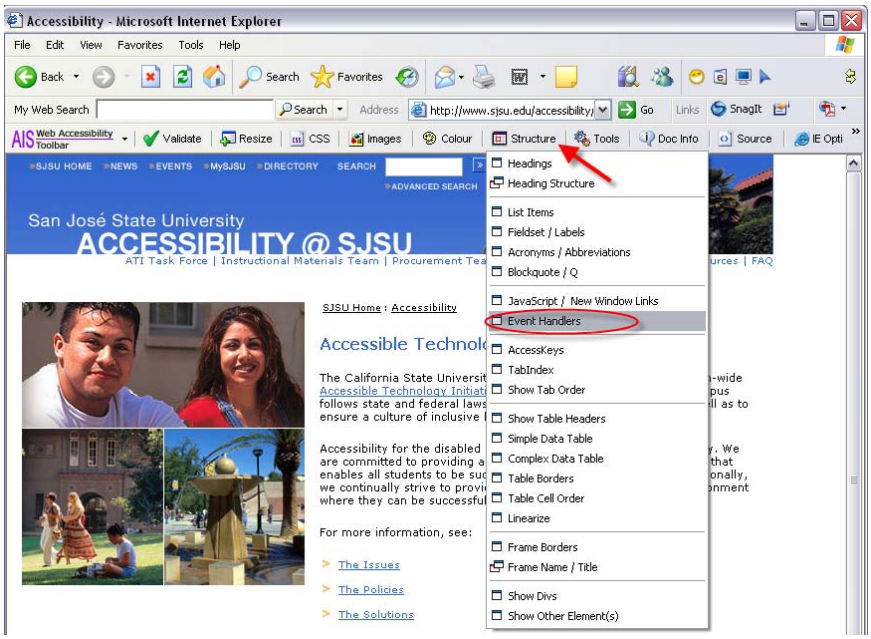
*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

1. **Question L1:** Are HTML event handlers accessible to both mouse and keyboard users? (If yes, then pass.)

2. To answer this question, use the **IE/AIS toolbar**.

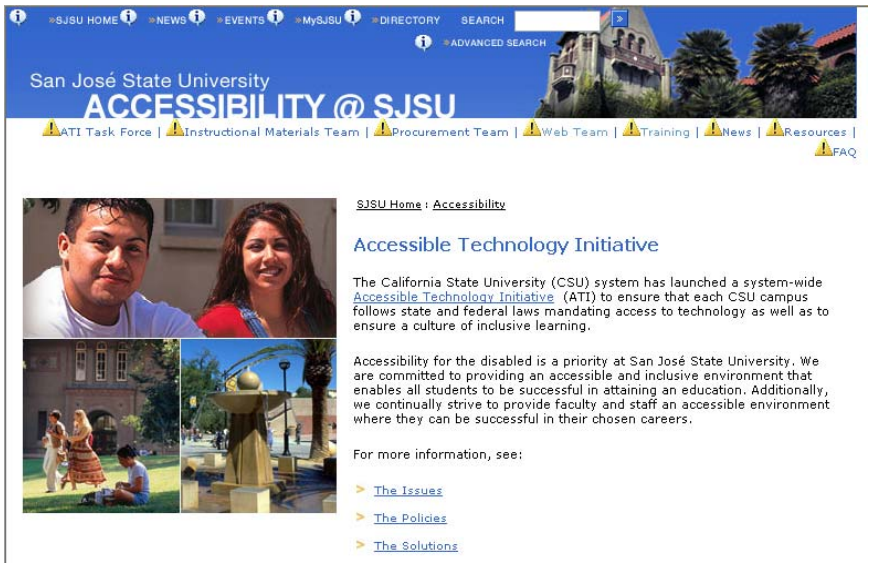
3. Click on **Structure** > **Event Handlers**.




**The Microsoft Internet Explorer window displays with list of event handlers.**

**The web page displays with icons next to all event handlers.**

4. When done, turn off event handlers by refreshing the page.



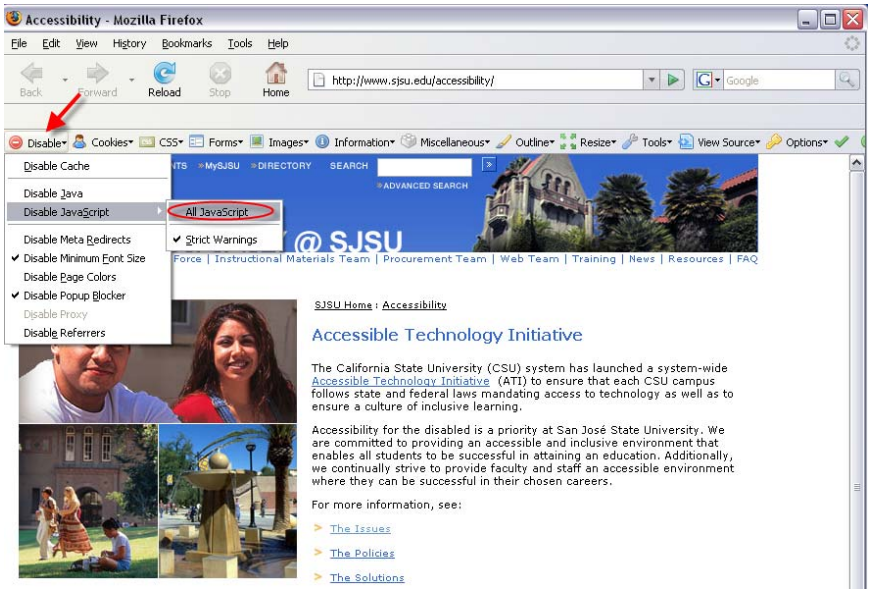

**The web page displays.**

5. **Question L2:** Is all content and functionality generated by scripts also provided to user agents that do not support scripts? (If yes, then pass.)

6. To answer this question, use the **FF/Developer toolbar**.

7. Click on **Disable** > **Disable JavaScript** > **All JavaScript**.

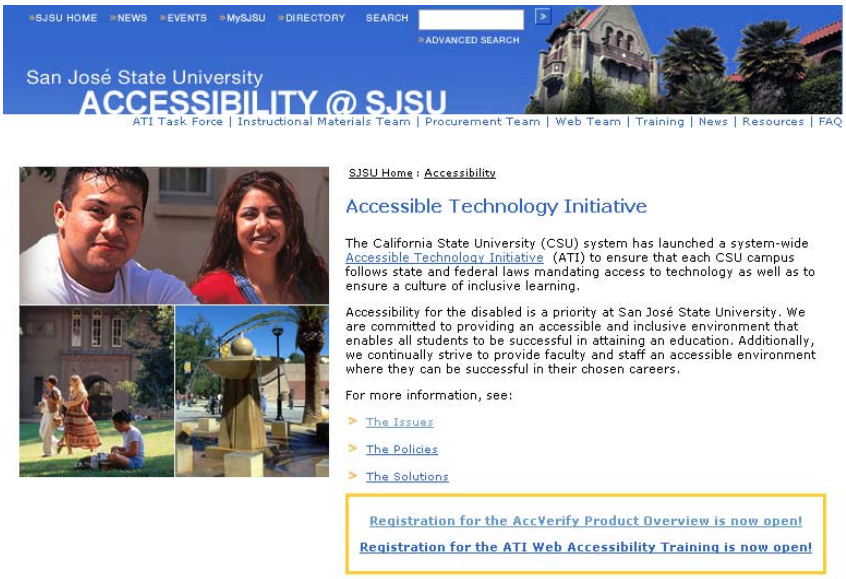
*Note: As an alternative, use the Internet Explorer menu to navigate to **IE Options** > **Toggle JavaScript**, (then refresh page).*



**The web page displays without JavaScript.**

8. Note whether your web page passes, fails, or is not applicable for Checkpoint K.

9. When done, turn JavaScript back on by closing the tab or reloading the page.

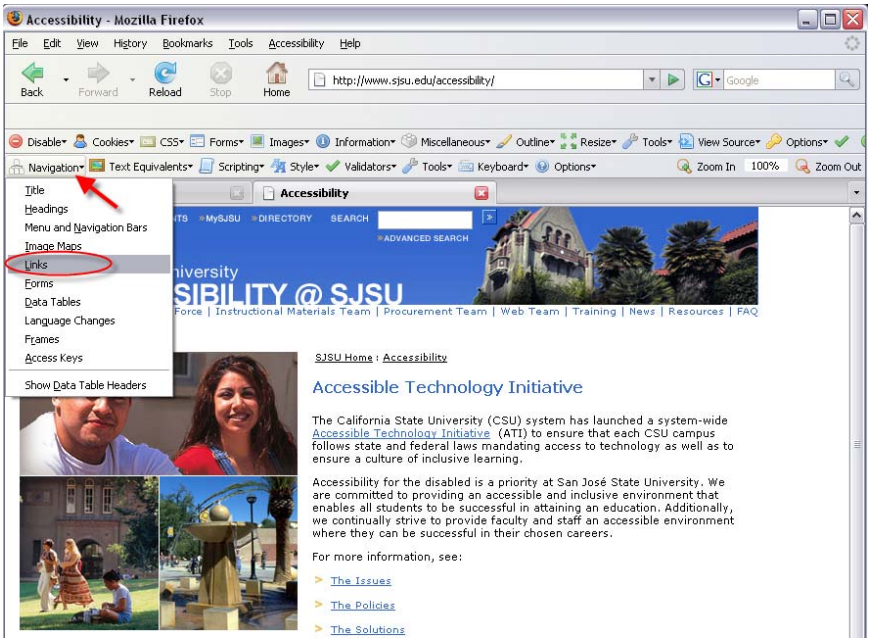



**The web page displays.**

10. **Question L3 (Best Practice):** Is "href='javascript:...'" used in any anchor element? (If no, then pass.)

11. To answer this question, use the **FF/Illinois toolbar**.

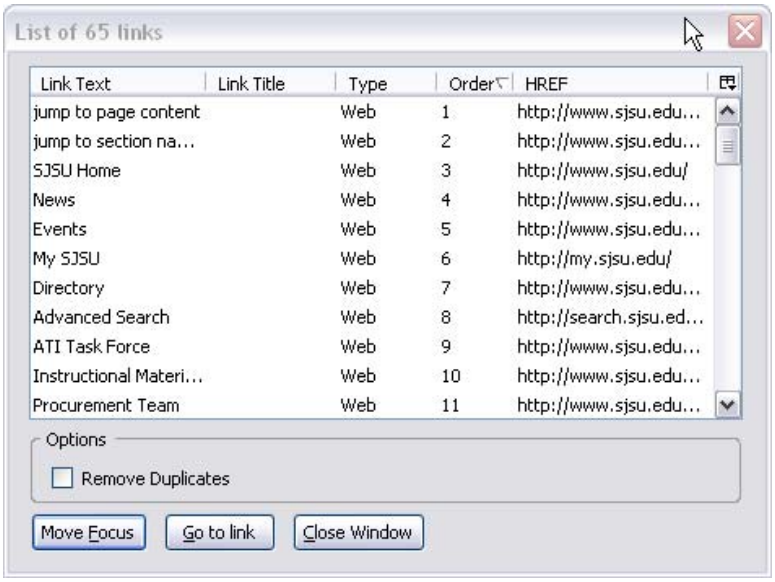
12. Click on **Navigation** > **Links**.



**A new window opens displaying a list of links.**

13. Check to make sure that javascript is not in the HREF column.

14. When done, close window.

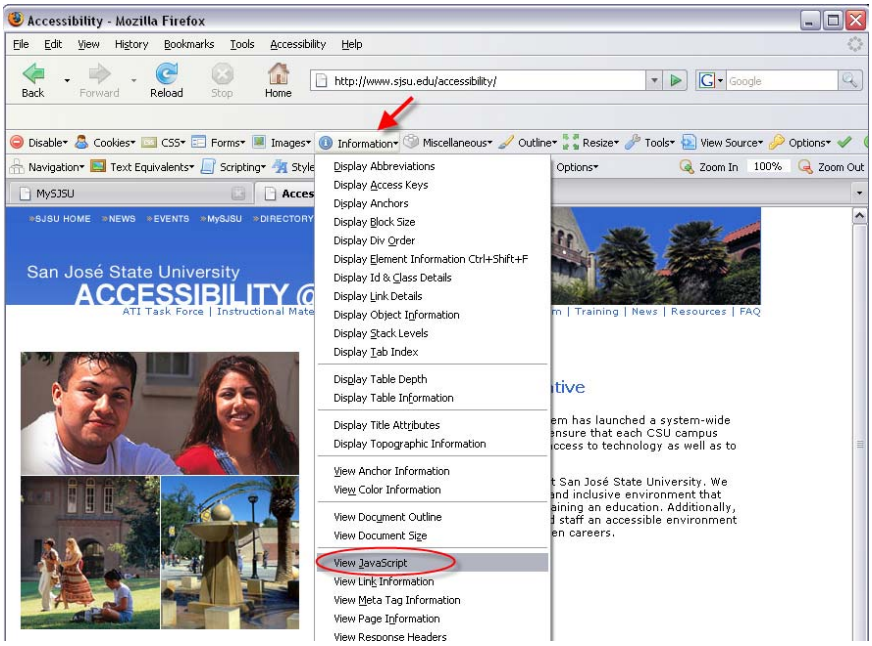



**The web page displays.**

15. **Question L4 (Best Practice):** Is modern "unobtrusive" JavaScript used in place of older coding techniques? (If yes, then pass.)

16. To answer this question, use the **FF/Developer toolbar**.

17. Click on **Information** > **View JavaScript**.

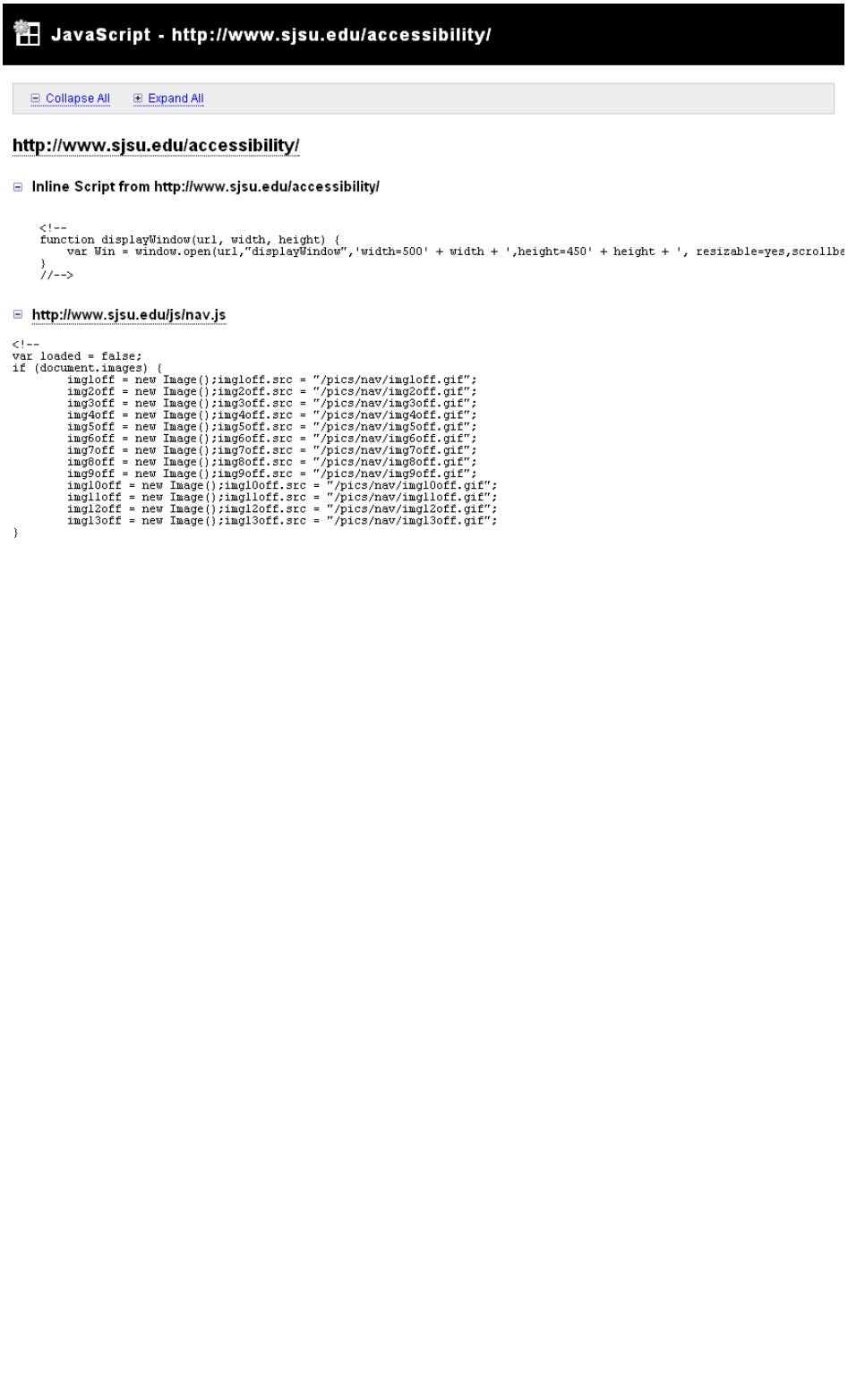


**The JavaScript displays.**

18. Note whether your web page passes, fails, or is not applicable for Checkpoint K.

19. When done, turn view JavaScript off by closing the tab or reloading the page.

*Reference: Web Accessibility, Chapter 10, Accessible JavaScript.*



**Checkpoint M**

This checkpoint has become substantially more complicated to evaluate since it was written. There are two problems here.

- The first, *applets*, can be thought of as true computer applications running within a browser; for example, Java or ActiveX controls—these will have to meet the separate Section 508 standards for all computer software.

- The second, *plug-ins*, could mean anything from Flash to Quick-time to PDF or even Word. Most browsers now provide built-in rendering for many of these formats, but many assistive technologies probably won't.

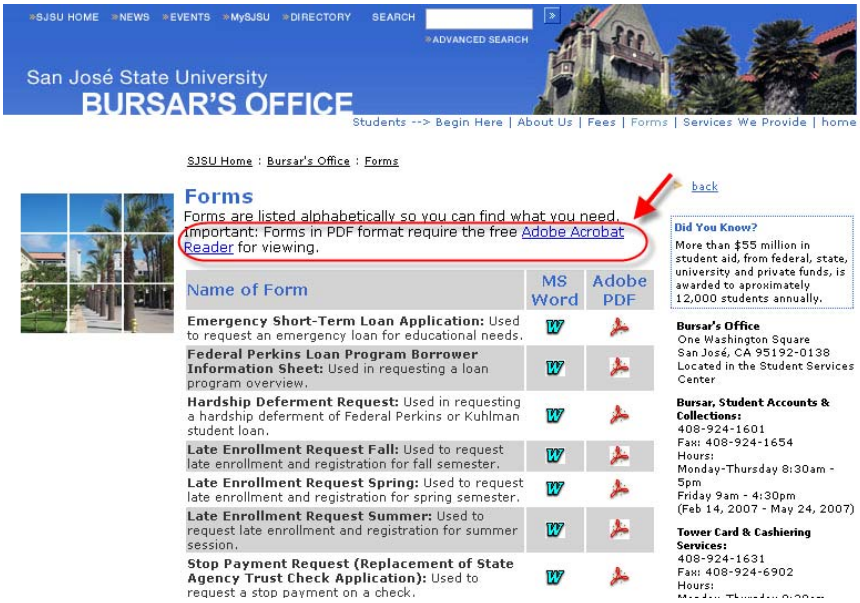
Equivalent content will have to be provided for a large range of disabilities; for example, captioning of video and transcripts of audio—and formats such as PDF will have to be coded with current accessibility techniques.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

1. **Question M1:** Are links provided to any special readers or plug-ins that are required to interpret page content? (If yes, then pass.)

2. To answer this question, make a visual, audio, and functional inspection depending on content.



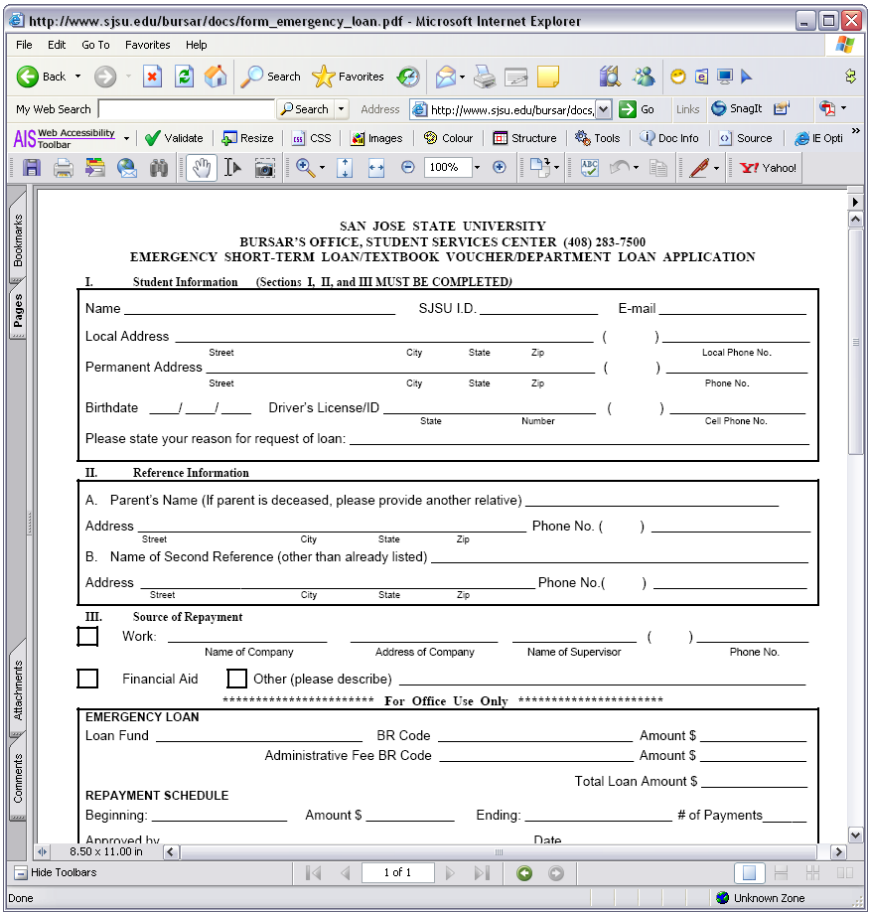
**The document displays.**

3. **Question M2:** Do special readers or plug-ins comply with the requirements of Section 508 paragraphs 1194.21(a)-(l), the Subpart B technical standards for software? (If yes, then pass.)

4. **Question M3:** Are documents that are presented in commonly-supported formats such as PDF and Word constructed so as to be accessible? (If yes, then pass.)

5. To answer these questions, make a visual, audio, and functional inspection depending on content.

*References: Web Accessibility, Chapter 11, Accessible Flash and Chapter 12, PDF Accessibility.*



**Checkpoint N**

Visitors using assistive technology need to access form controls independently of how they are presented on the screen. They also need to receive all of the cues that normal browsers provide—for example, which fields are mandatory and what, if any, errors have been made in completing the form. Proper semantic markup (XHTML) helps assistive technology to present form information and controls accessibly.

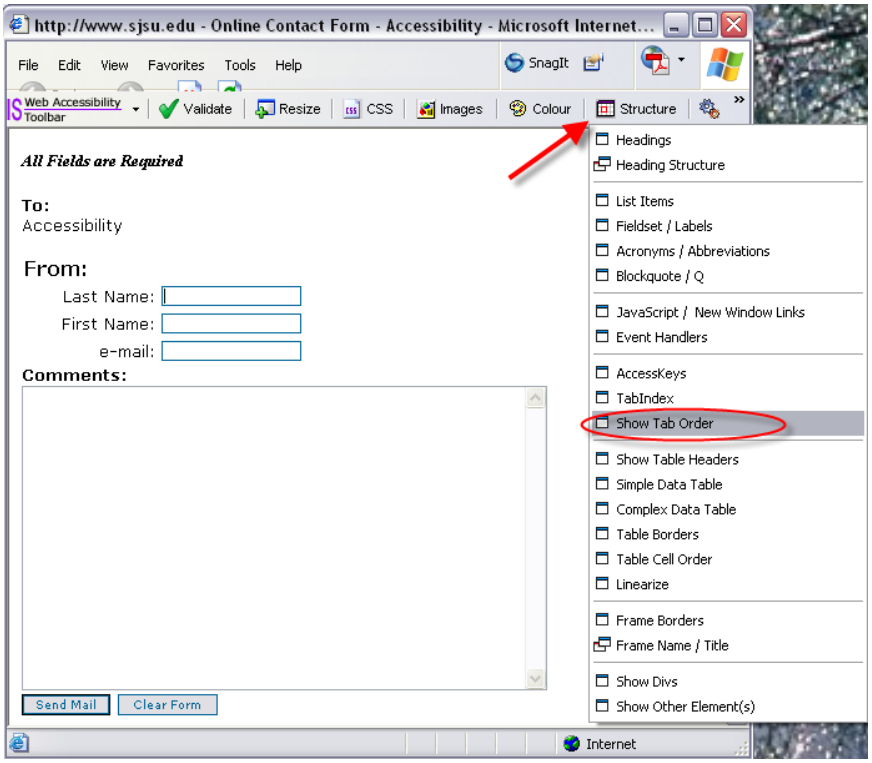
*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The form displays.**

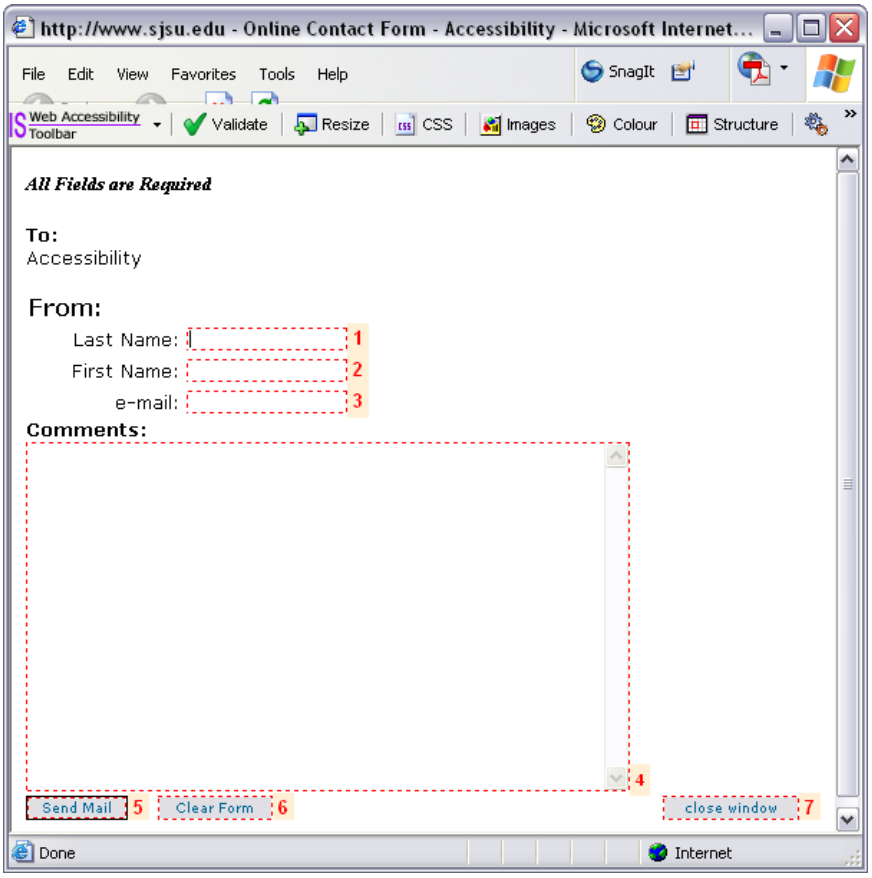
1. **Question N1:** Is the tab order to reach the form and the tab order between form elements consistent with the normal order of entering form data? (If yes, then pass.)

2. To answer this question, use the **IE/AIS toolbar**.

3. Click on **Structure** > **Show Tab Order**.



**The form displays with tab order marked.**

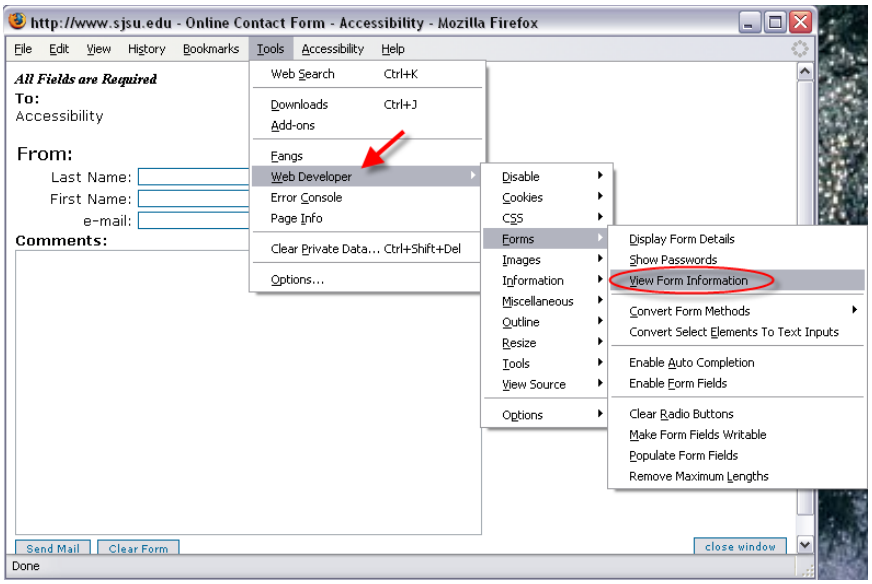



**The form displays.**

4. **Question N2:** Does each <input> element or control (except buttons) have an associated and visible <label> element? (If yes, then pass.)

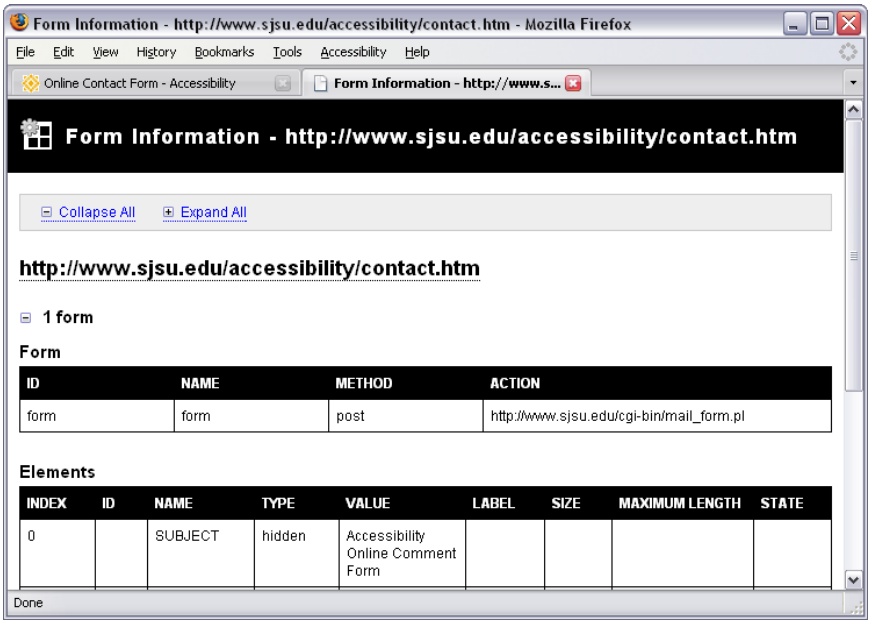
5. To answer this question, use the **FF/Developer toolbar**.

6. Click on **Forms** > **View Form Information**.



**A new window opens displaying form information.**

7. Check to make sure form elements have labels, except for hidden, submit, reset.



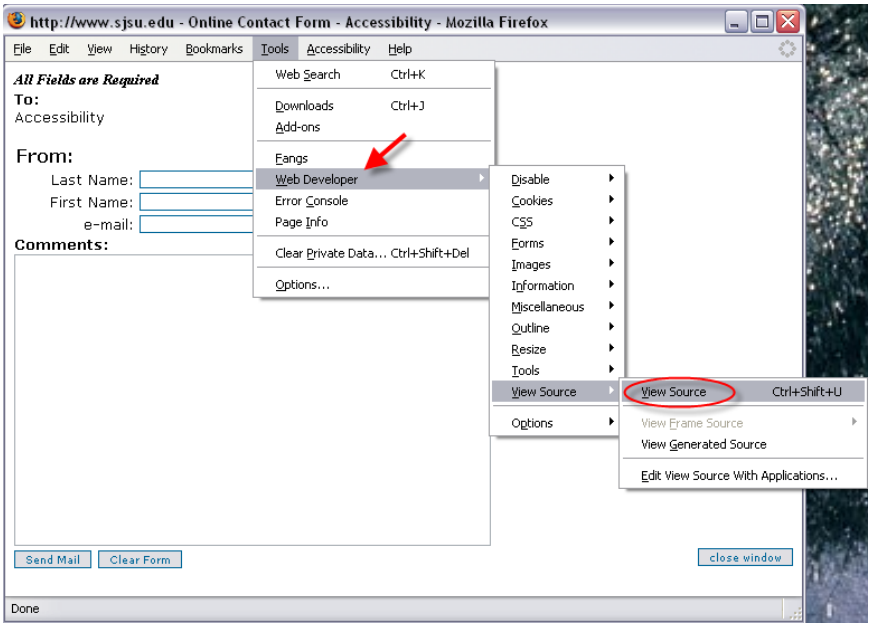

**The form displays.**

8. **Question N3**: Are logically-related groups of form elements identified with appropriate <fieldset>, <legend>, or <caption> elements? (If yes, then pass.)

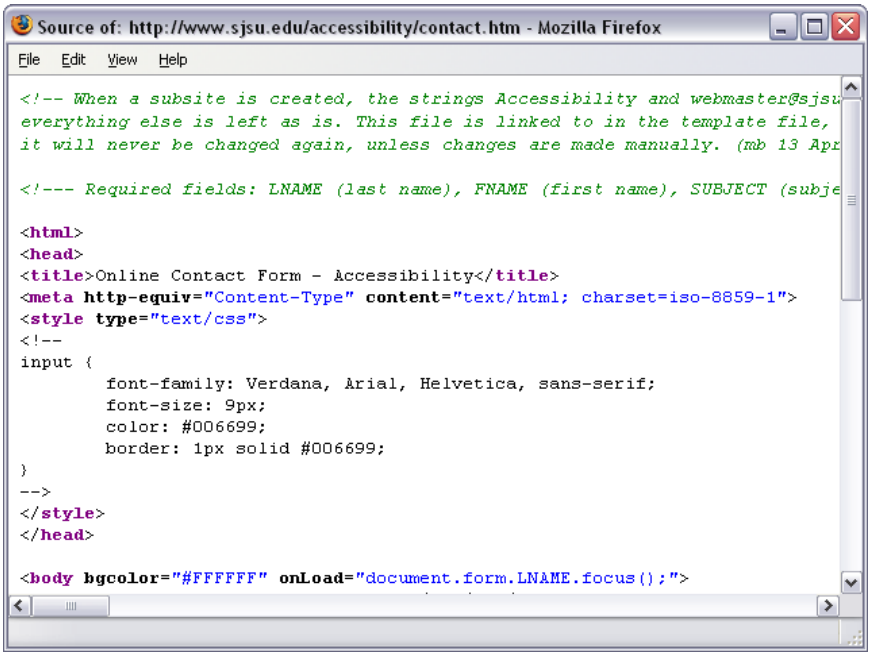
9. To answer this question, use the **FF/Developer toolbar**.

10. Click on **View Source** > **View Source**.

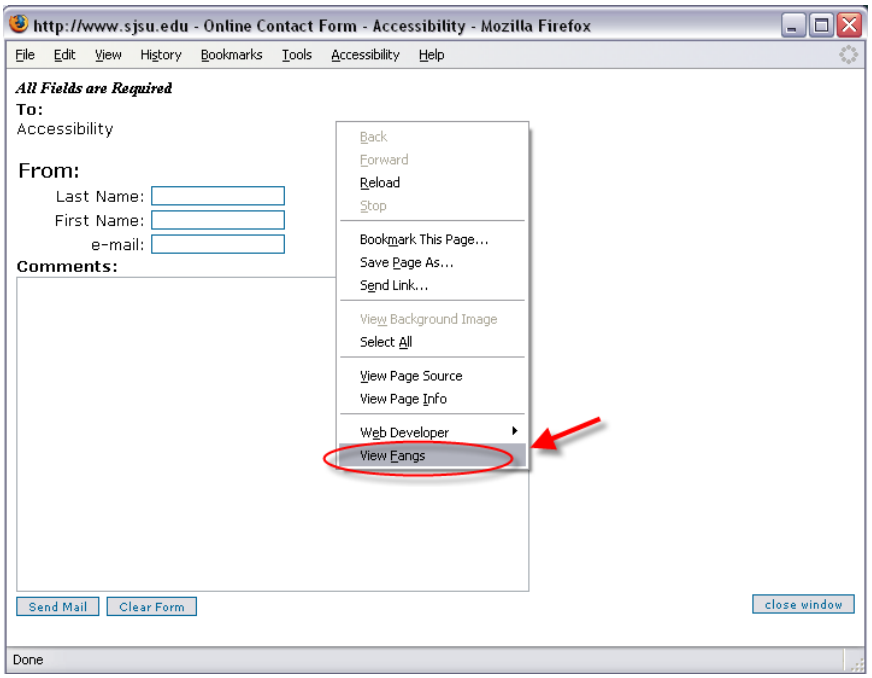
*Note: As an alternate, use the IE Menu to click on View > Source.*



**A new window opens displaying source code.**




**The form displays.**




11. **Question N4:** Are all cues for filling out the form (mandatory fields, help boxes, error messages, and so on) available to users of assistive technology? (If yes, then pass.)

12. **Question N5:** Is placeholder text, if used, redundant or distracting to users of assistive technology? (If no, then pass.)

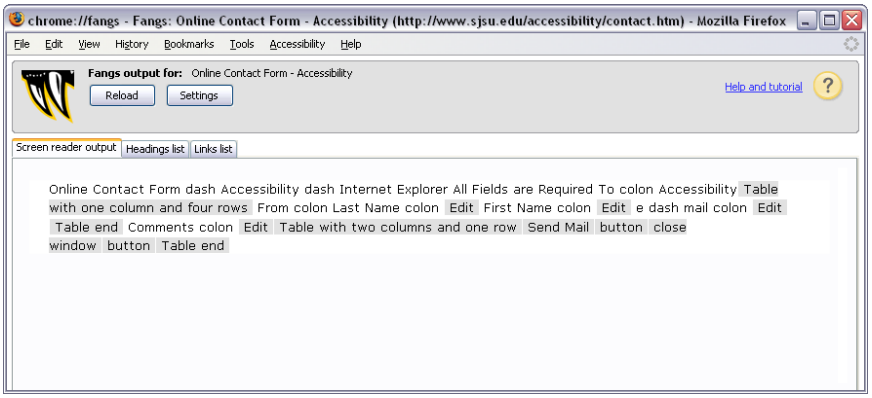
13. To answer these questions, right click anywhere on the page and select **Fangs**.

**A new window opens displaying Fangs.**

14. If window is blank, click on the Reload button.

*Note: When retrofitting, most properly labeled form elements no longer benefit from placeholder text.*

*Reference: Web Accessibility, Chapter 8, Accessible data input.*



**Checkpoint O**

**Skip Links**

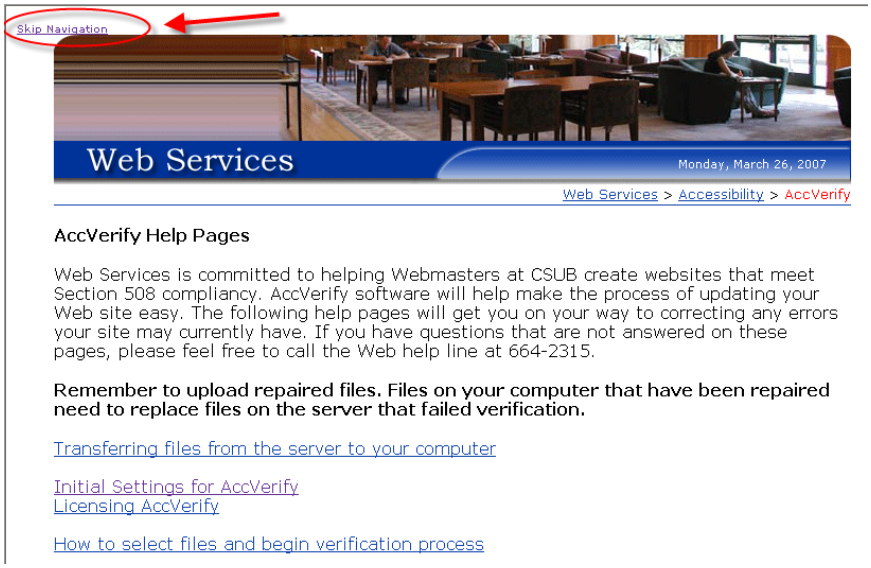
This may be the most misunderstood and poorly implemented of all Section 508 checkpoints. Conventional wisdom is to hide a ***skip*** link from sighted viewers, often with an awkward and outdated single-pixel .gif image. Yet low-vision users may benefit from a visible way to bypass repetitive links, and fully-sighted viewers are not bothered in the least by its presence. Worse, pages might contain multiple groups of repetitive links, which a user might want to ***skip*** in whole or only in part. Since assistive technologies now support navigation by headings and lists, these constructs can be used to organize links beneficially for all users.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

**The web page displays.**

1. **Question O1:** Does each group of repetitive navigation links have a visible and audible way to skip reading? (If yes, then pass.)

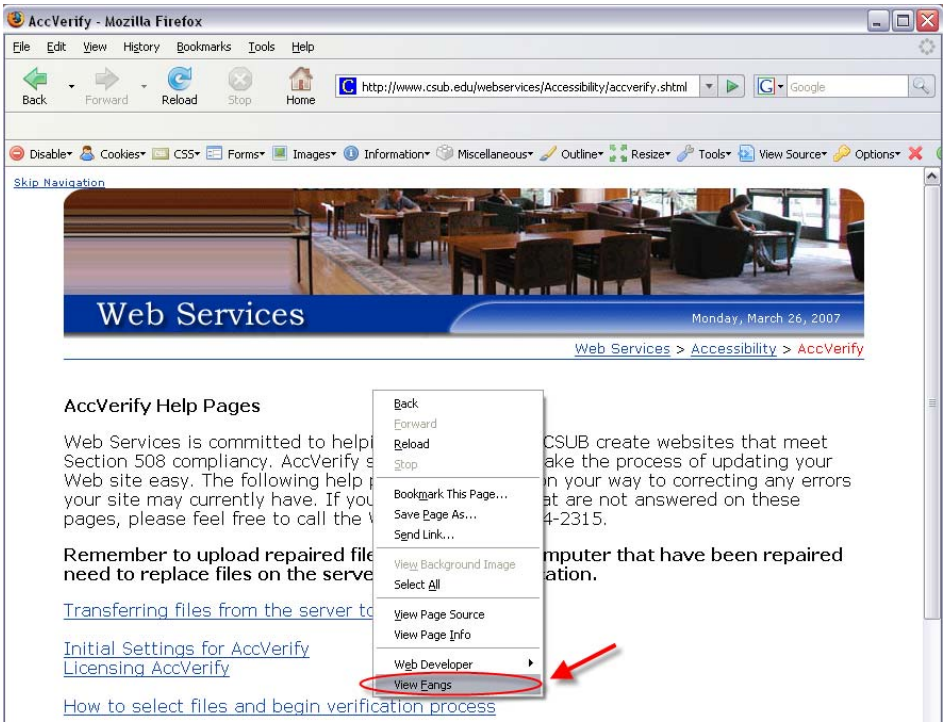
2. To answer the visible part of the question, make a visual inspection of any skip links on your web page.



**The web page displays.**

3. To answer the audible part of the question, use **Fangs**.

4. Right click anywhere on the page and select **View Fangs**.




**The Fangs window displays.**

5. If window is empty, click on **Settings**.

**The Fangs Settings window displays.**

6. Use the drop down menu to select **Sectioned**.

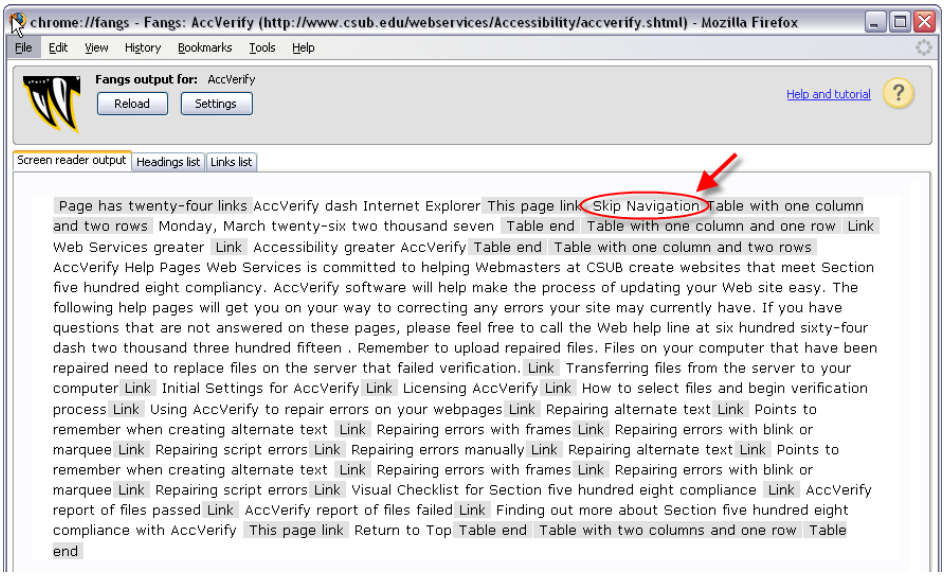
7. Click on the **OK** button

**The Fangs window displays.**

8. Click on the **Reload** button.

**The Fangs window displays.**

9. When done, close Fangs window.

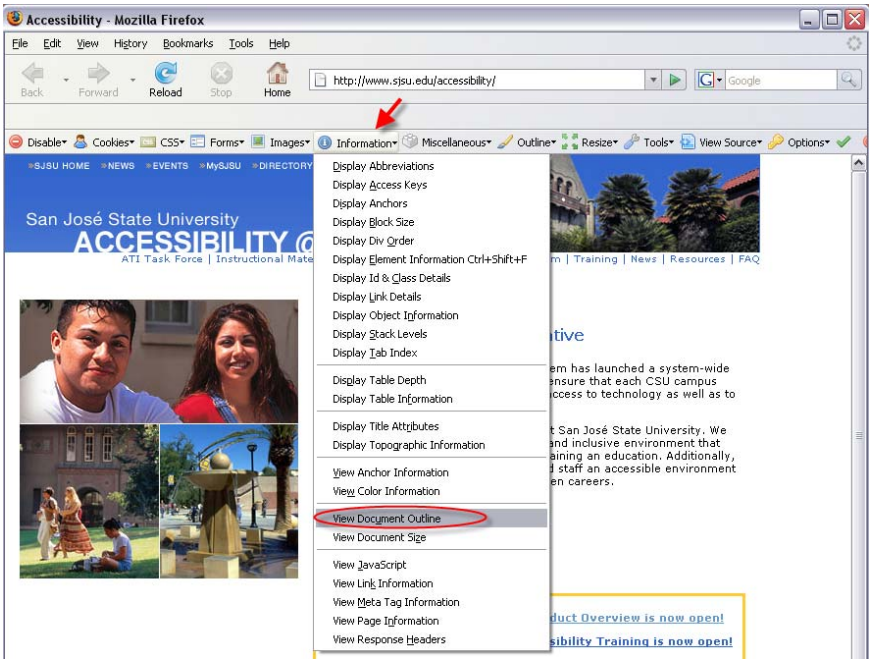


**The web page displays.**

10. **Question O2:** Are multiple groups of links navigable from one to another by means of headings or other methods that can be used by assistive technologies? (If yes, then pass.)

11. To answer this question, use the **FF/Developer toolbar**.

12. Click on **Information** > **View Document Outline**.




**The document outline displays**.

13. Note whether your web page passes, fails, or is not applicable for Checkpoint O.

14. When done, turn off document outline by closing the tab or reloading the page.

*Reference: Web Accessibility, pages 184–196.*

**Checkpoint P**

**Timed Response**

Users may have a variety of difficulties when a timed response is required from a web form. In education, one possible reason requiring such a response would be in an examination, where it would be impractical to allow all students to request more time, as called for in the checkpoint. Other accommodations, such as a customized version of the test or a special testing center, may have to be provided in this case.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at Tom Jewett Web Design/Specializing in Accessibility (http://www.tomjewett.com/accessibility/).*

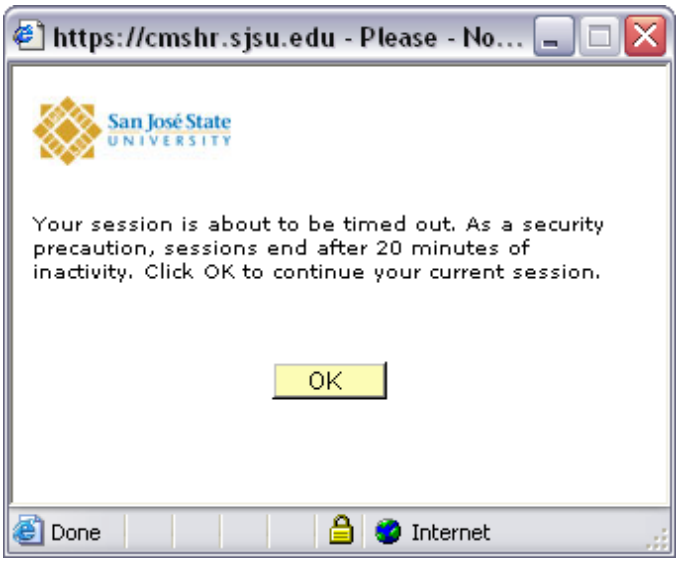
**The web page displays.**

1. **Question P1:** If a timed response is required, is the user alerted and given sufficient time to indicate that more time is required? (If yes, then pass.)

2. **Question P2:** In a timed environment such as an examination, are users who need more time accommodated in an appropriate alternative way?* (If yes, then pass.)

3. To answer this question, make a visual inspection of your web page.

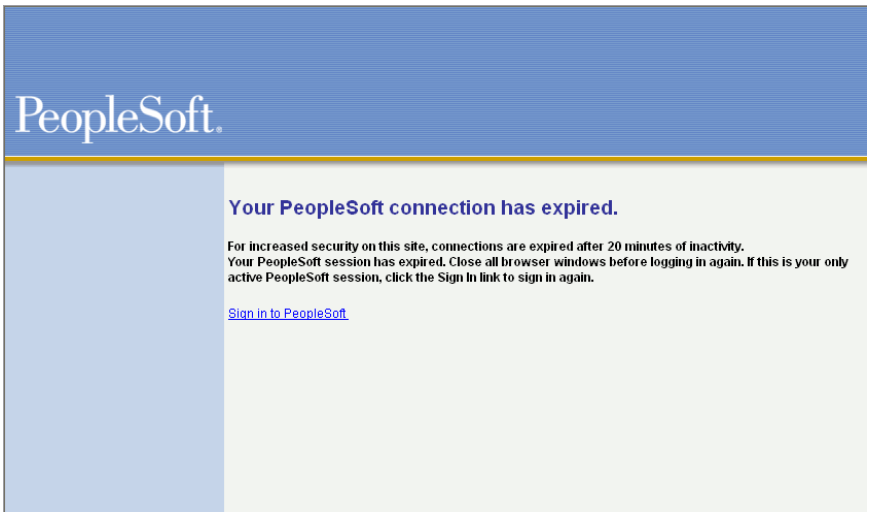
**This may be may be an organizational rather than technical solution.*



**The web page displays.**

4. Note whether your web page passes, fails, or is not applicable for Checkpoint P.

*Note: You may also need to do a functional inspection, depending on content.*



### User Validation

Users with mildly low vision or some cognitive disabilities will benefit from text that is simply resized in a standard browser. Text that is styled to prevent this, or page layout that requires horizontal scrolling for large text, will be a barrier to these users. Even after all checkpoints appear to have been met, practical use by persons with actual disabilities, using their normal assistive technology or accommodations, may discover additional barriers that need to be fixed.

*Source: Tom Jewett, CSU Long Beach, Department of Computer Engineering and Computer Science at http://www.tomjewett.com/accessibility/.*

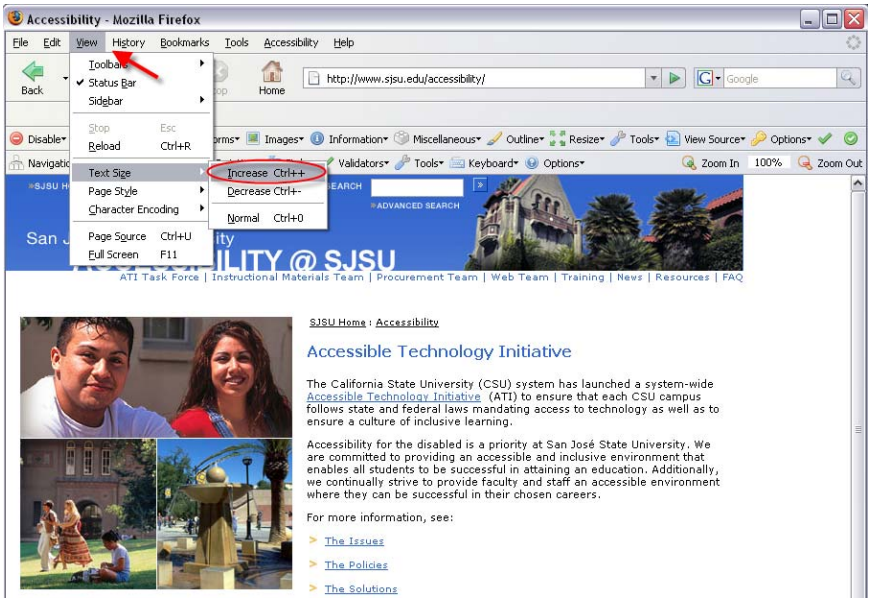
**The web page displays.**

1. **User Question 1:** Does all text (except in banner graphics) enlarge or shrink in response to changes in browser text size settings? (If yes, then pass.)

2. **User Question 2:** Does text enlarged with browser settings wrap within columns? (If yes, then pass)

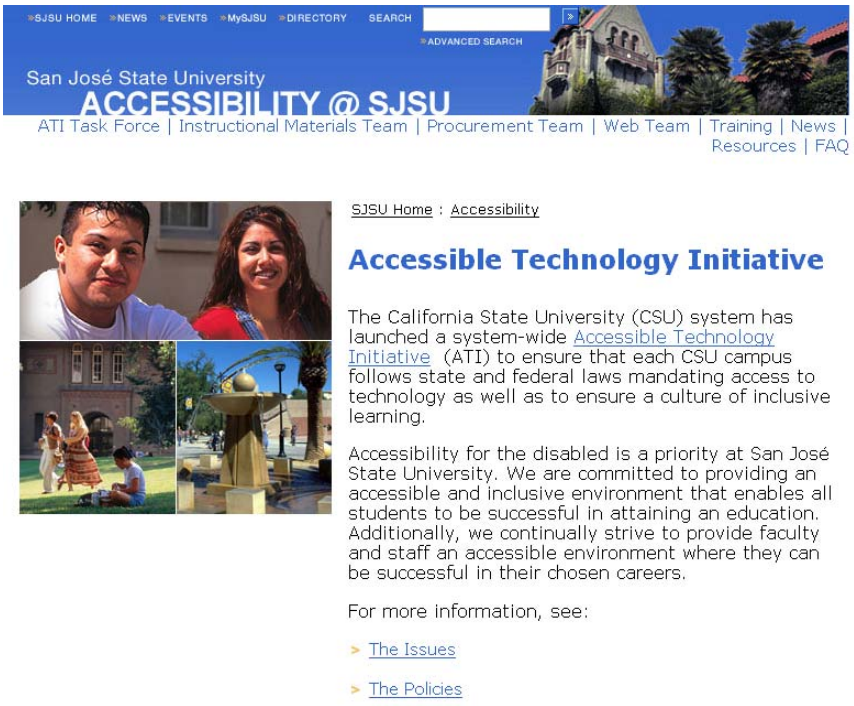
3. **User Question 3:** Do columns, page elements, or text lines overlap each other when text is enlarged with browser settings? (If no, then pass)

4. To answer these questions, use the **FF Menu** to click on **View** > **Text Size** > **Increase or Decrease** (or Ctrl-+ and Ctrl--)



**The web page displays with text enlarged.**

*Note: As an alternate, use the IE Menu to click on Page > Text Size > Larger, etc. (Not Ctrl-+, which zooms whole page.)*



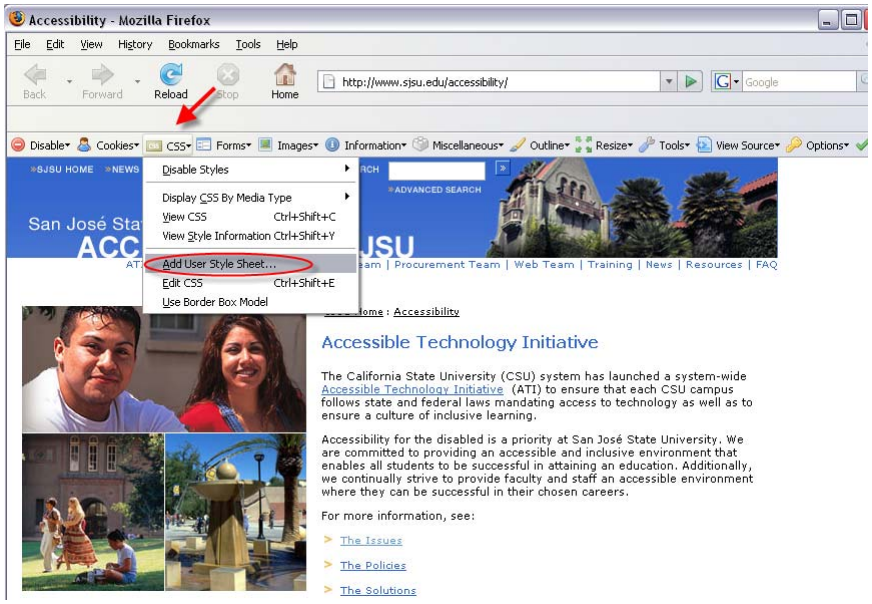

**The web page displays.**

5. **User Question 4:** Do actual low vision users encounter any remaining barriers? (If no, then pass.)

6. To answer this question, ask actual vow vision users to view page.

7. As an alternate, use a low vision style sheet.

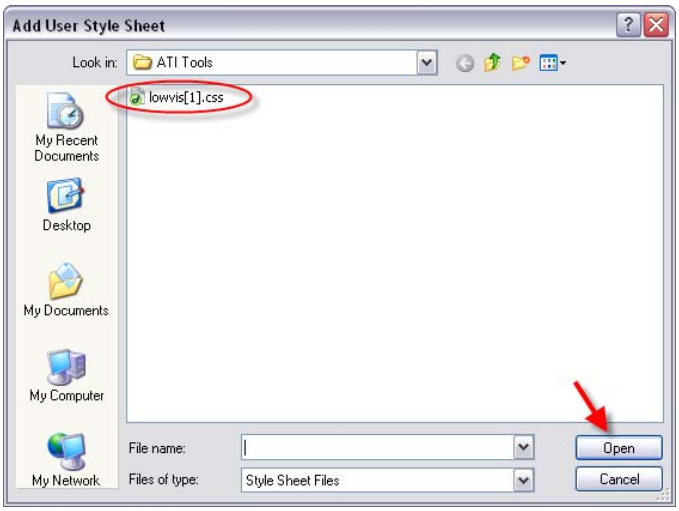
8. On the **FF/Developer toolbar**, click on **CSS** > **Add User Style Sheet**.



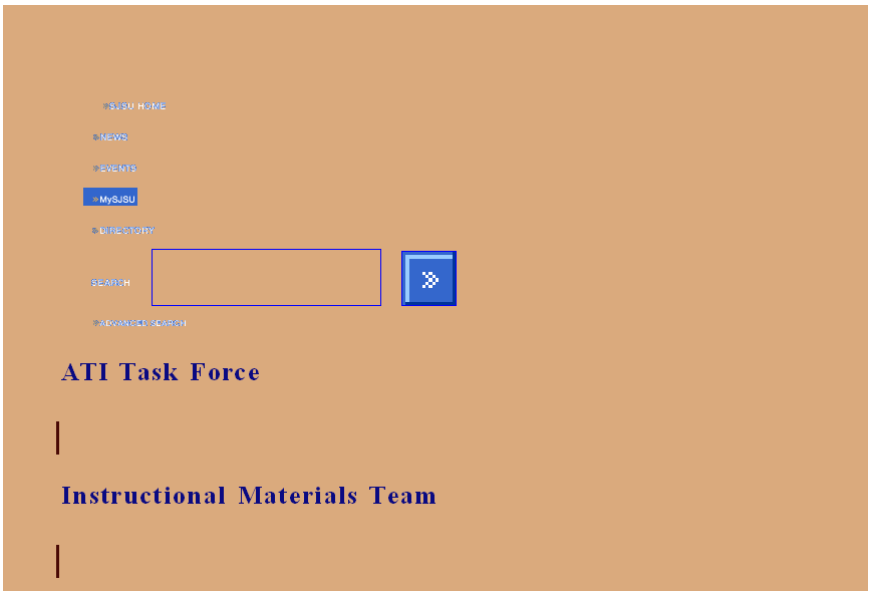
**The Add User Style Sheet window opens.**

9. Browse to file named **lowvis.css**.

10. Click on the **Open** button.




**The web page displays using the low vision style sheet.**



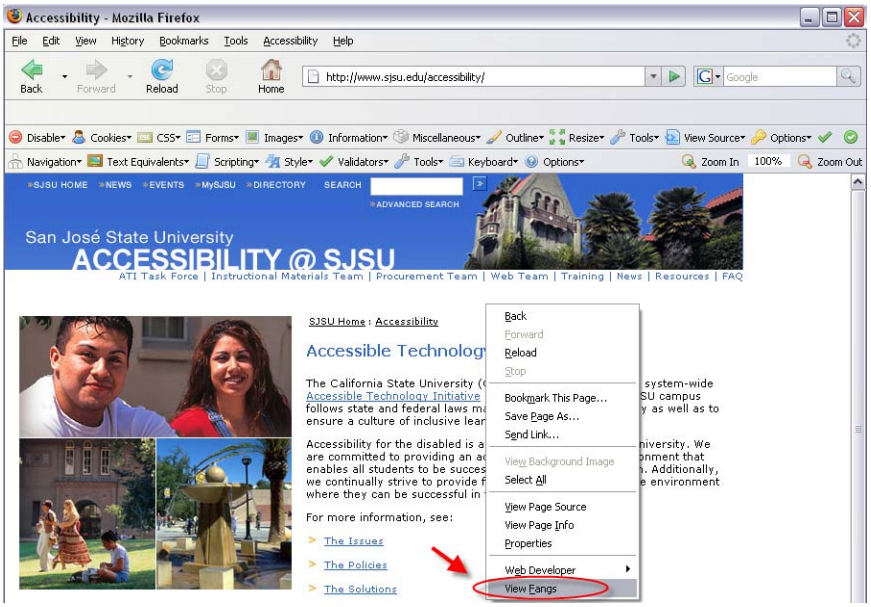
**The web page displays.**

11. **User Question 5:** Do actual blind users encounter any remaining barriers? (If no, then pass.)

12. **User Question 6:** Do actual users with other disabilities encounter any remaining barriers? (If no, then pass.)

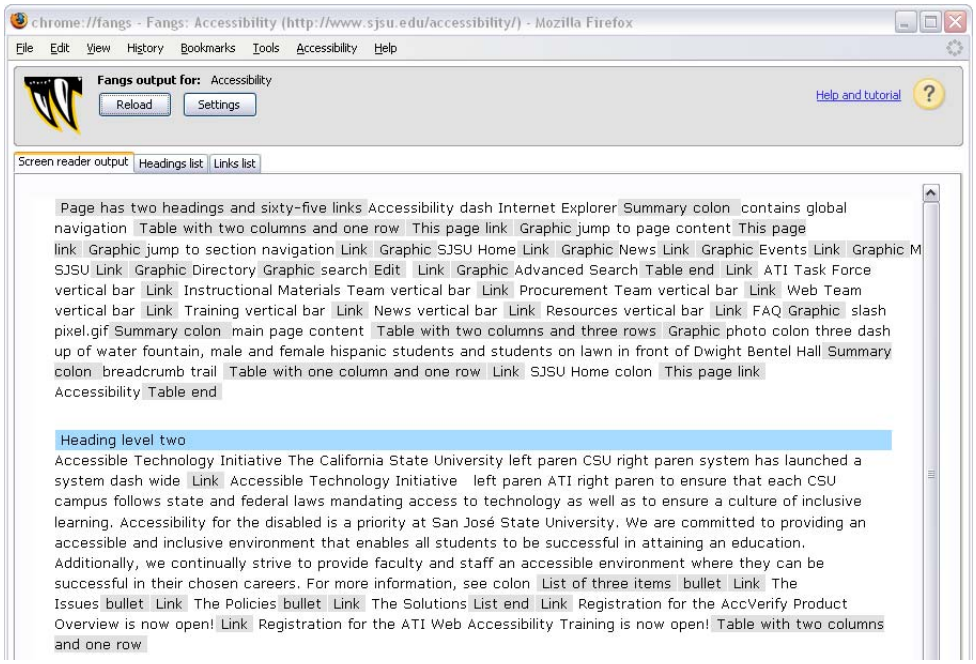
13. To answer these questions, ask actual blind user to view page.

14. As an alternate, right click anywhere on the page and select **View Fangs**.




**The Fangs screen reader displays.**

15. If the window is blank, click on the Reload button.

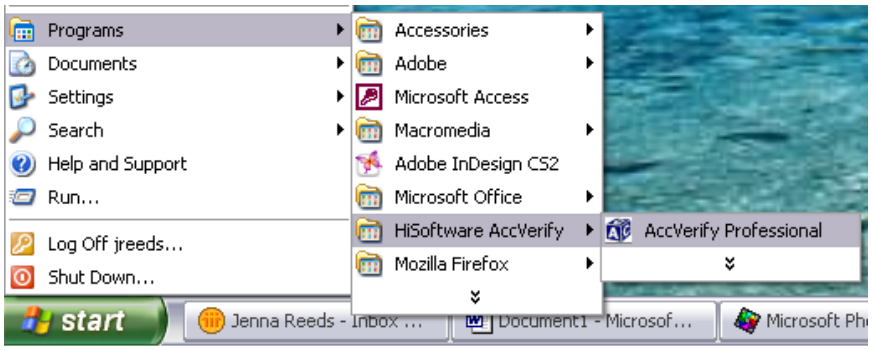


# Section 4: Run Accessibility Report with Interview Wizard

This section explains how to perform global settings in HiSoftware AccVerify, how to run an accessibility check on a single web page and how to understand and evaluate your report.

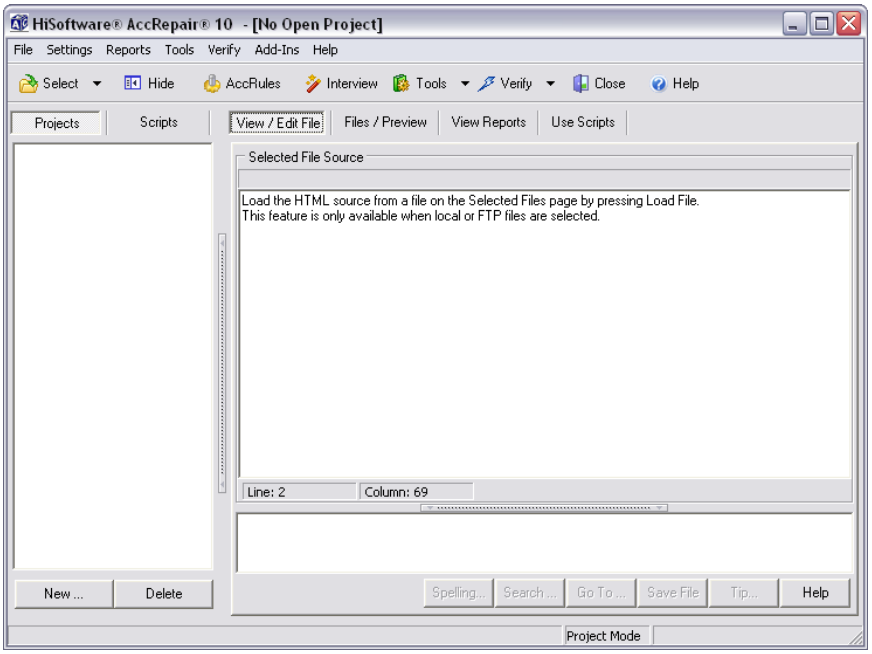
### Global Settings

1. Navigate to **Start** > **Programs** > **HiSoftware AccVerify** > **AccVerify Professional**.



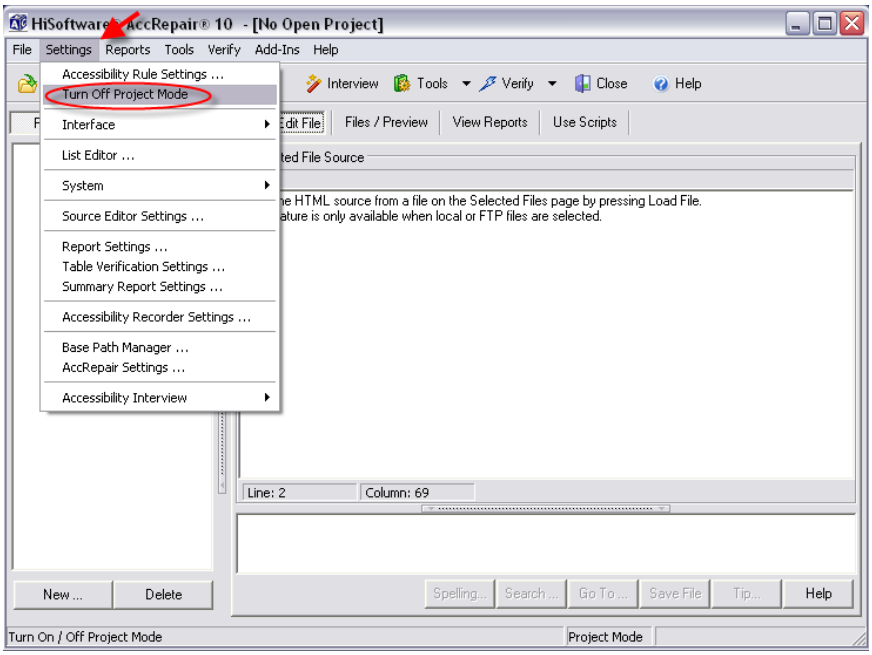

**The AccVerify window opens.**

*Note: When Project Mode is on, AccVerify displays three windows.*



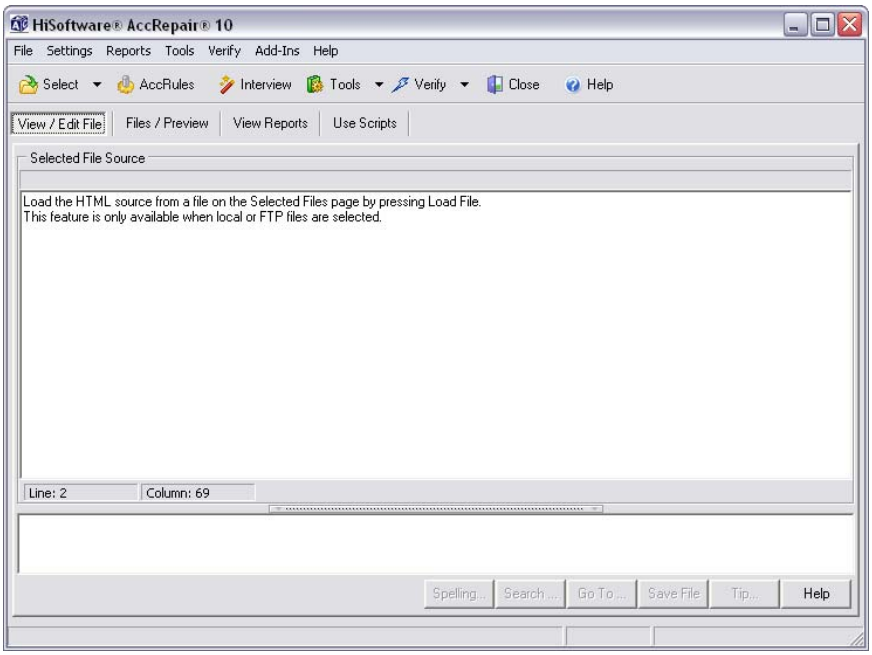
**The AccVerify Main Menu displays.**

2. Click on **Settings** > **Turn Off Project Mode**.



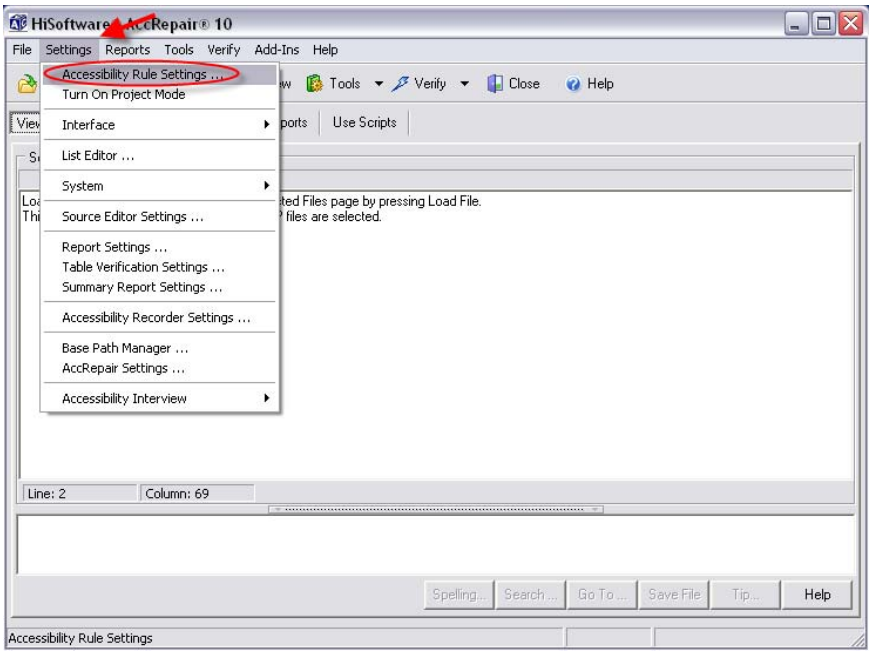

**Project Mode is now off.**

*Note: When Project Mode is off, AccVerify displays only two windows.*



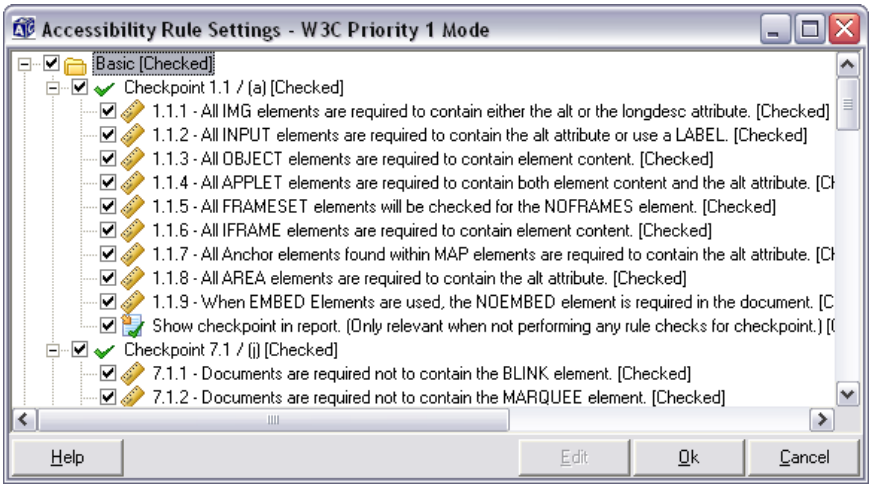
**The Main Menu displays.**

3. Click on **Settings** > **Accessibility Rule Settings**.




**The Accessibility Rule Settings window opens.**

4. Leave default checks and add the following checks:
   - Rule 6.3.3
   - Rule (n).1.
5. Click on the **OK** button.



### Global Settings Administration

### Edit Accessibility Checks

**Basic**

**Checkpoint**

☑ 1.1 / (a) Provide a text equivalent for every non-text element (e.g. via "alt", "longdesc", or in element content). This includes: images, graphical representations of text (including symbols), image map regions and animations.

**Advanced Rule Settings for Checkpoint 1.1 / (a)**

- ☑ 1.1.1 - All IMG elements are required to contain either the alt or the longdesc attribute.
- ☑ 1.1.2 - All INPUT elements are required to contain the alt attribute or use a LABEL.
- ☑ 1.1.3 - All OBJECT elements are required to contain element content.
- ☑ 1.1.4 - All APPLET elements are required to contain both element content and the alt attribute.
- ☑ 1.1.5 - All FRAMESET elements will be checked for the NOFRAMES element.
- ☑ 1.1.6 - All IFRAME elements are required to contain element content.
- ☑ 1.1.7 - All Anchor elements found within MAP elements are required to contain the alt attribute.
- ☑ 1.1.8 - All AREA elements are required to contain the alt attribute.
- ☑ 1.1.9 - When EMBED Elements are used, the NOEMBED element is required in the document.

☑ 7.1 / (j) Until user agents allow users to control flickering, avoid causing the screen to flicker

**Advanced Rule Settings for Checkpoint 7.1 / (j)**

- ☑ 7.1.1 - Documents are required not to contain the BLINK element.
- ☑ 7.1.2 - Documents are required not to contain the MARQUEE element.

☑ 2.1 / (c) Ensure that all information conveyed with color is also available without color, for example from context or markup.

☑ 6.1 / (d) Organize documents so they may be read without style sheets. For example, when an HTML document is rendered without associated style sheets, it must still be possible to read the document.

**Image Maps**

**Checkpoint**

☑ 1.2 / (e) Provide redundant text links for each active region of a server-side image map.

**Advanced Rule Settings for Checkpoint 1.2 / (e)**

- ☑ 1.2.1 - Locate any IMG element that contains the 'ismap' attribute.
- ☑ 1.2.2 - Locate any INPUT element that contains the 'ismap' attribute.

☑ 9.1 / (f) Provide client-side image maps instead of server-side image maps except where the regions cannot be defined with an available geometric shape.

**Advanced Rule Settings for Checkpoint 9.1 / (f)**

- ☑ 9.1.1 - No IMG element should contain the 'ismap' attribute.
- ☑ 9.1.2 - No INPUT element should contain the 'ismap' attribute.

**Tables**

**Checkpoint**

☑ 5.1 / (g) For data tables, identify row and column headers.

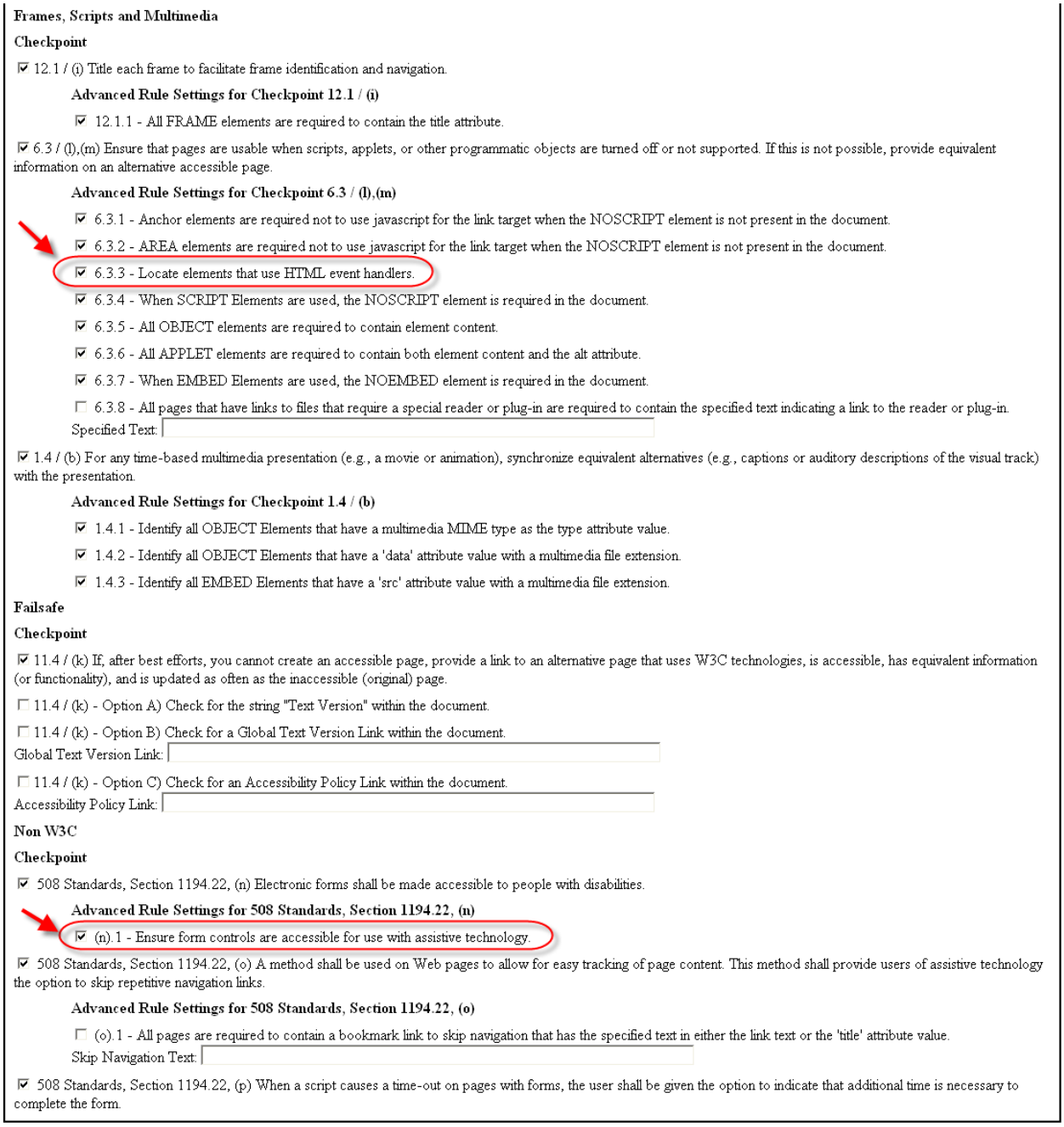
**Advanced Rule Settings for Checkpoint 5.1 / (g)**

- ☑ 5.1.1 - Identify all Data TABLE elements.

☑ 5.2 / (h) For data tables that have two or more logical levels of row or column headers, use markup to associate data cells and header cells.

**Advanced Rule Settings for Checkpoint 5.2 / (h)**

- ☑ 5.2.1 - Identify all Data TABLE elements.



**Frames, Scripts and Multimedia**

**Checkpoint**

☑ 12.1 / (i) Title each frame to facilitate frame identification and navigation.

**Advanced Rule Settings for Checkpoint 12.1 / (i)**

☑ 12.1.1 - All FRAME elements are required to contain the title attribute.

☑ 6.3 / (l),(m) Ensure that pages are usable when scripts, applets, or other programmatic objects are turned off or not supported. If this is not possible, provide equivalent information on an alternative accessible page.

**Advanced Rule Settings for Checkpoint 6.3 / (l),(m)**

☑ 6.3.1 - Anchor elements are required not to use javascript for the link target when the NOSCRIPT element is not present in the document.

☑ 6.3.2 - AREA elements are required not to use javascript for the link target when the NOSCRIPT element is not present in the document.

☑ 6.3.3 - Locate elements that use HTML event handlers.

☑ 6.3.4 - When SCRIPT Elements are used, the NOSCRIPT element is required in the document.

☑ 6.3.5 - All OBJECT elements are required to contain element content.

☑ 6.3.6 - All APPLET elements are required to contain both element content and the alt attribute.

☑ 6.3.7 - When EMBED Elements are used, the NOEMBED element is required in the document.

☐ 6.3.8 - All pages that have links to files that require a special reader or plug-in are required to contain the specified text indicating a link to the reader or plug-in.

Specified Text:

☑ 1.4 / (b) For any time-based multimedia presentation (e.g., a movie or animation), synchronize equivalent alternatives (e.g., captions or auditory descriptions of the visual track) with the presentation.

**Advanced Rule Settings for Checkpoint 1.4 / (b)**

☑ 1.4.1 - Identify all OBJECT Elements that have a multimedia MIME type as the type attribute value.

☑ 1.4.2 - Identify all OBJECT Elements that have a 'data' attribute value with a multimedia file extension.

☑ 1.4.3 - Identify all EMBED Elements that have a 'src' attribute value with a multimedia file extension.

**Failsafe**

**Checkpoint**

☑ 11.4 / (k) If, after best efforts, you cannot create an accessible page, provide a link to an alternative page that uses W3C technologies, is accessible, has equivalent information (or functionality), and is updated as often as the inaccessible (original) page.

☐ 11.4 / (k) - Option A) Check for the string "Text Version" within the document.

☐ 11.4 / (k) - Option B) Check for a Global Text Version Link within the document.

Global Text Version Link:

☐ 11.4 / (k) - Option C) Check for an Accessibility Policy Link within the document.

Accessibility Policy Link:

**Non W3C**

**Checkpoint**

☑ 508 Standards, Section 1194.22, (n) Electronic forms shall be made accessible to people with disabilities.

**Advanced Rule Settings for 508 Standards, Section 1194.22, (n)**

☑ (n).1 - Ensure form controls are accessible for use with assistive technology.

☑ 508 Standards, Section 1194.22, (o) A method shall be used on Web pages to allow for easy tracking of page content. This method shall provide users of assistive technology the option to skip repetitive navigation links.

**Advanced Rule Settings for 508 Standards, Section 1194.22, (o)**

☐ (o).1 - All pages are required to contain a bookmark link to skip navigation that has the specified text in either the link text or the 'title' attribute value.

Skip Navigation Text:

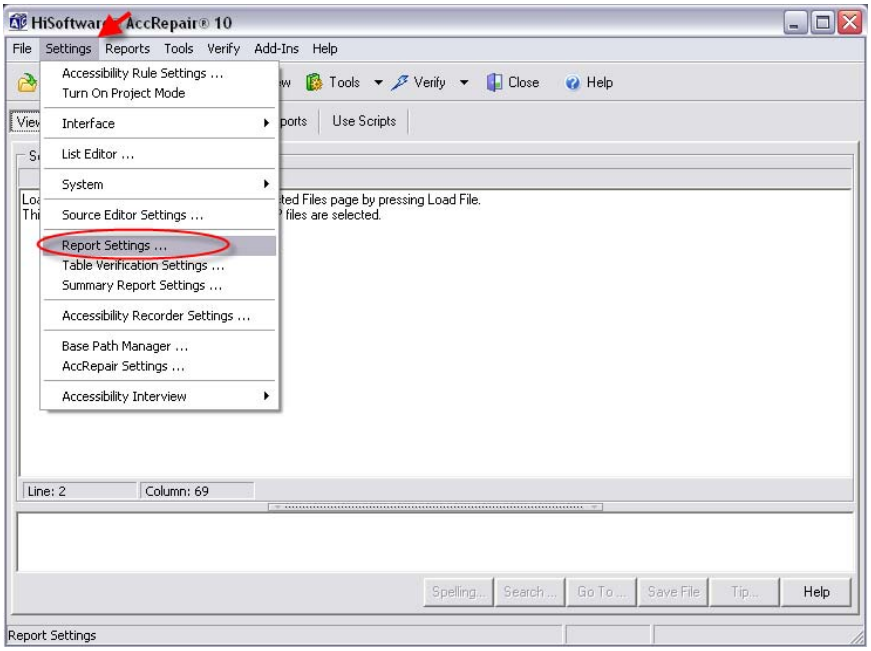
☑ 508 Standards, Section 1194.22, (p) When a script causes a time-out on pages with forms, the user shall be given the option to indicate that additional time is necessary to complete the form.

**Accessibility Rule Settings Saved Successfully!**

**6.** Click on the **OK** button.

**The AccVerify Main Menu displays**.

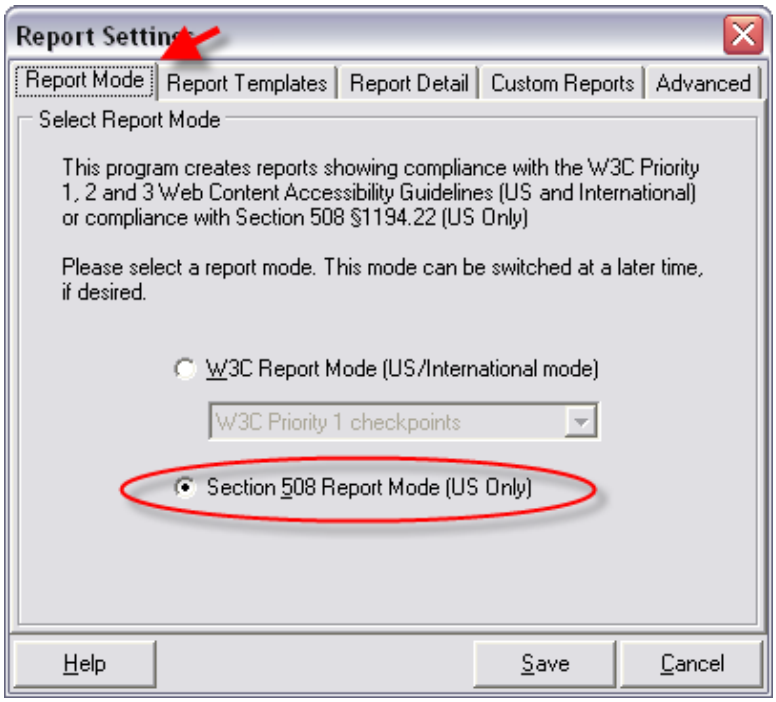
7. Click on **Settings** > **Report Settings**.



**The Report Settings window displays.**

8. Under the Report Mode top tab, select **Section 508 Report Mode**.

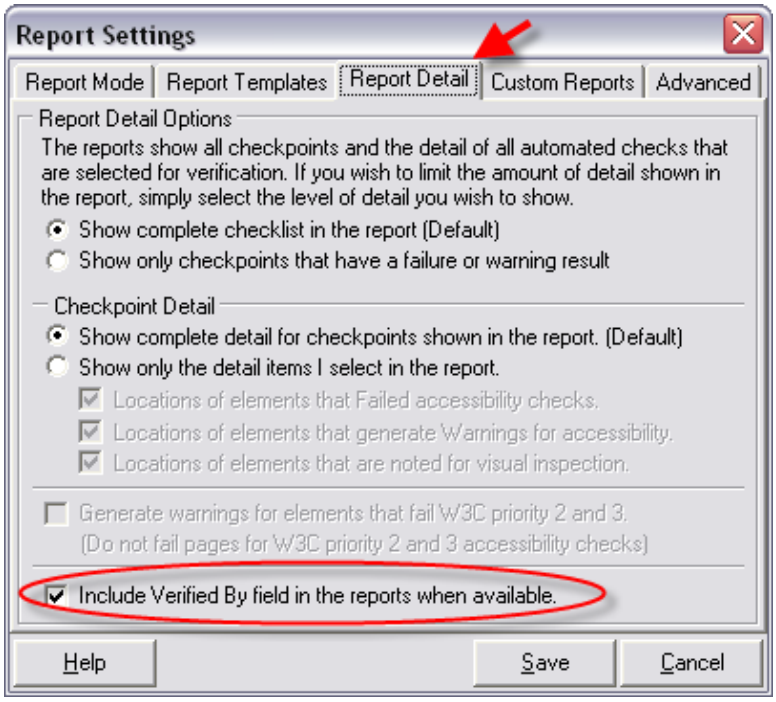
**9.** Click on the **Report Detail** tab.




**The Report Detail tab displays.**

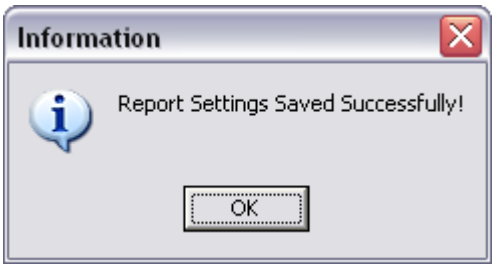
10. Under the Report Detail top tab, select **Include Verified By field in the reports when available**.

11. Click on the **Save** button.



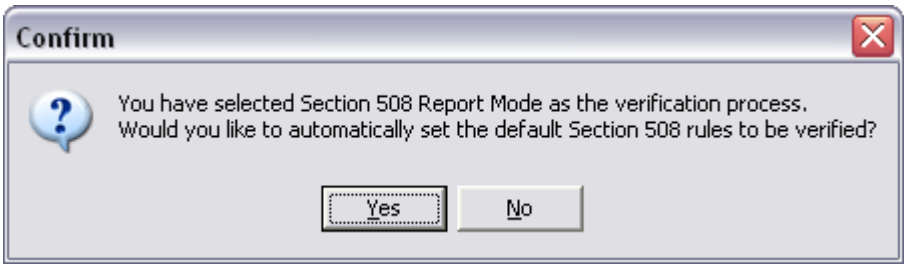
**Report Settings Saved Successfully!**

12. Click on the **OK** button.



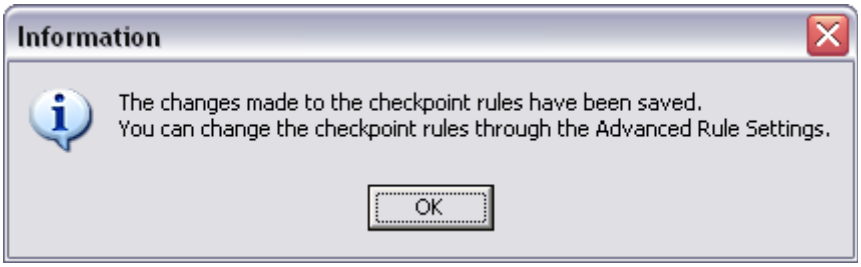

**The Confirm window displays.**

13. Click on the **Yes** button.



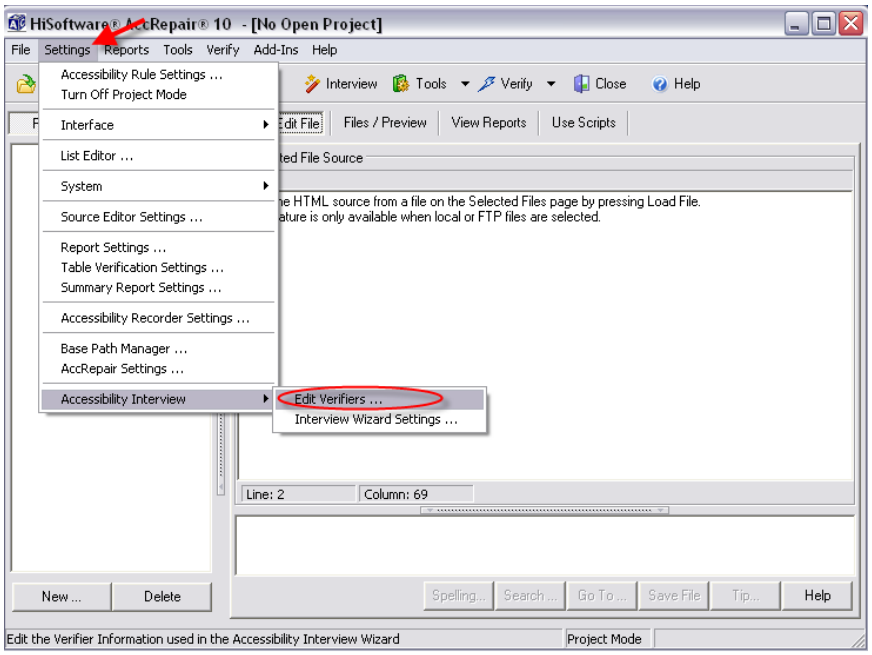

**The Information window displays**.

14. Click on the **OK** button.



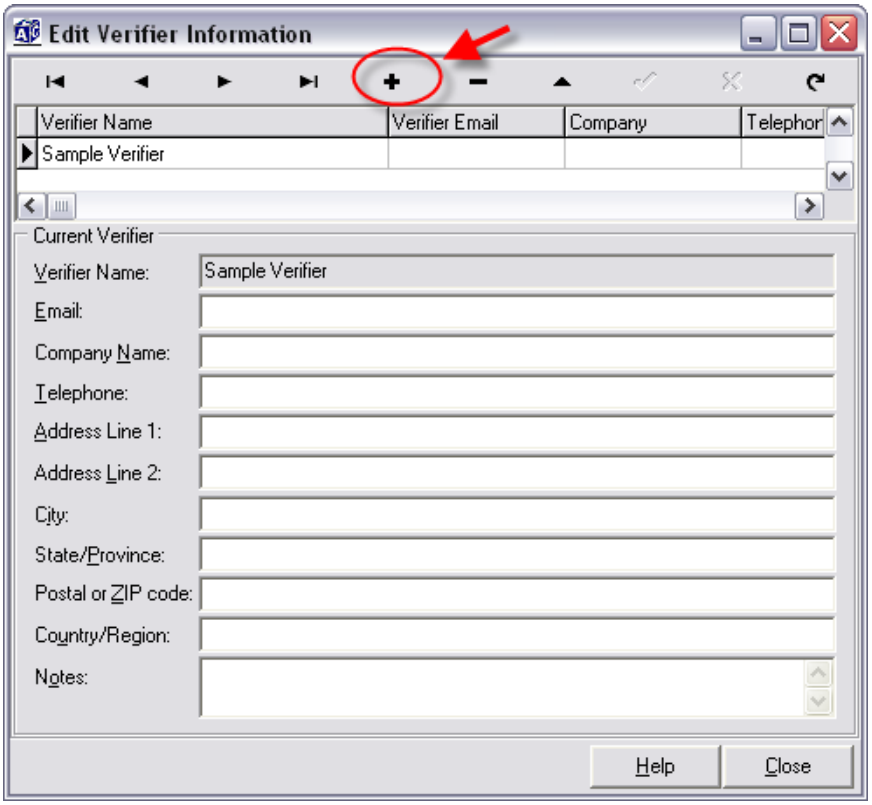
**The AccVerify Main Menu displays.**

15. Click on **Settings** > **Accessibility Interview** > **Edit Verifiers**.



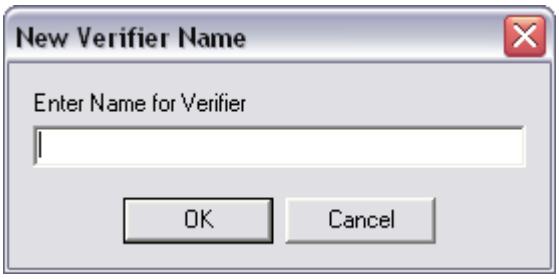

**The Edit Verifier Information window displays**.

16. Click on the add (+) button.



**The New Verifier Name window displays.**

17. Enter your first and last name.

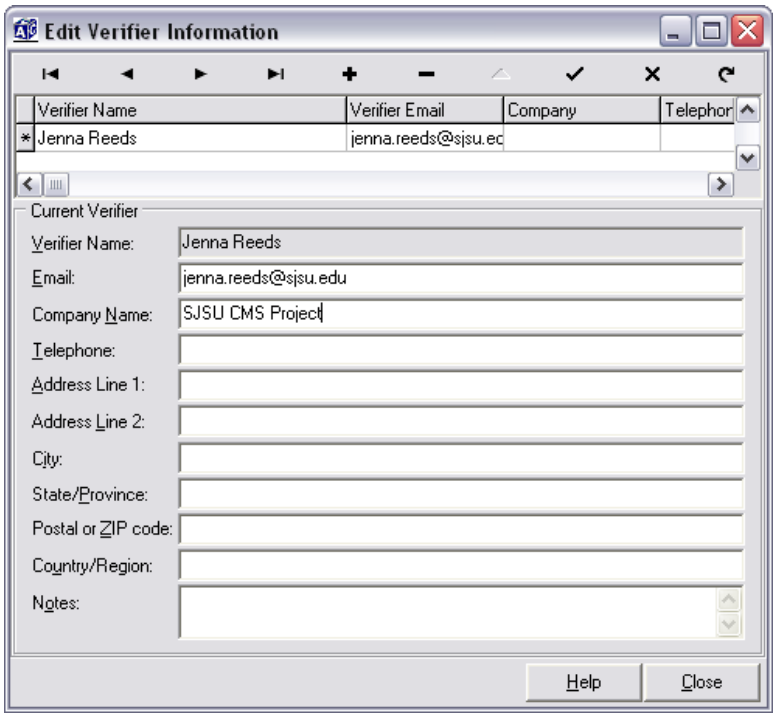



**The Enter Verifier Information window displays.**

18. **Email:** Enter your email address.

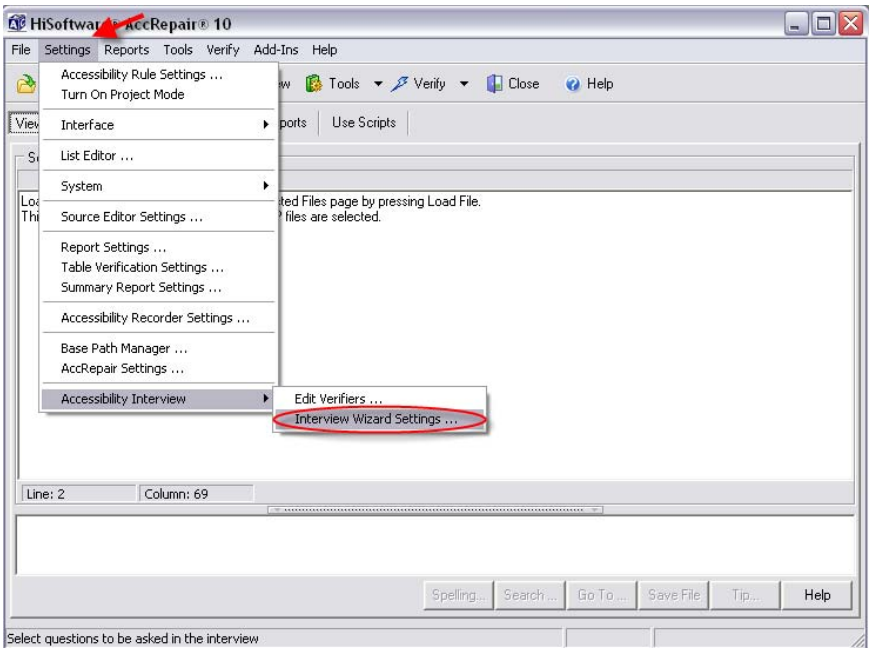
19. **Company Name:** Enter **SJSU** and your department name.

20. Click on the **Close** button.



**The AccVerify Main Menu displays.**

21. Click on **Settings** > **Accessibility Interview** > **Interview Wizard Settings**.

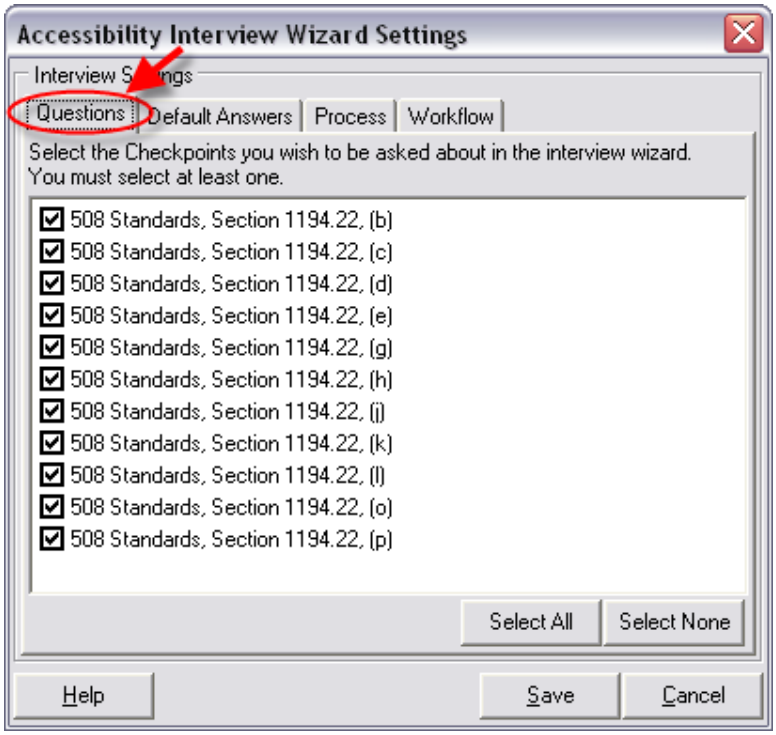



**The Accessibility Interview Wizard Settings window displays.**

22. Select the **Questions** top tab.

23. Click on the **Select All** button.

24. Select the **Process** top tab.



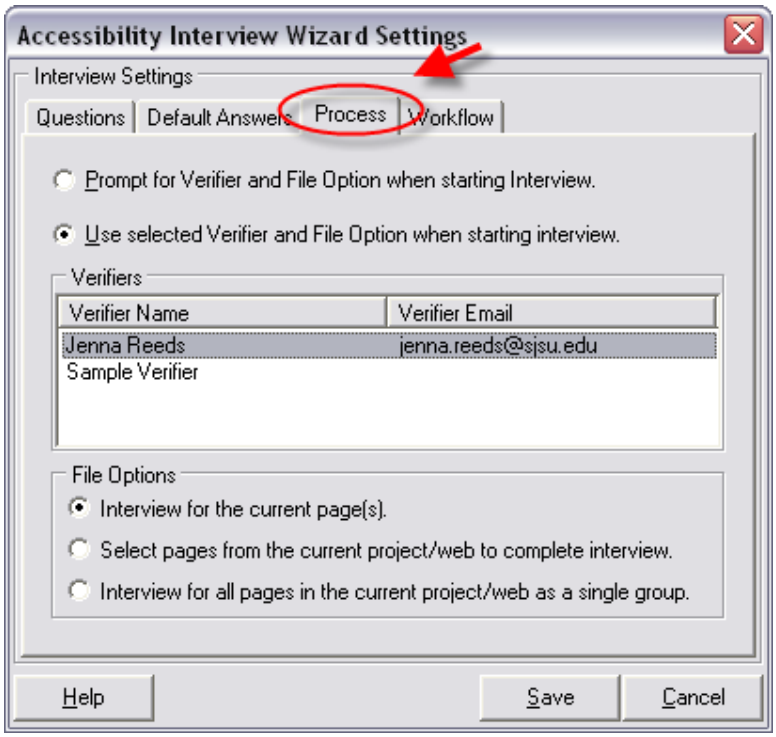
**The Process top tab displays.**

25. Select **Use selected Verifier and File Option when starting interview**.

26. **Verifier Name:** Select your name.

27. Select **Interview for current page(s)**.

28. Select the **Workflow** top tab.

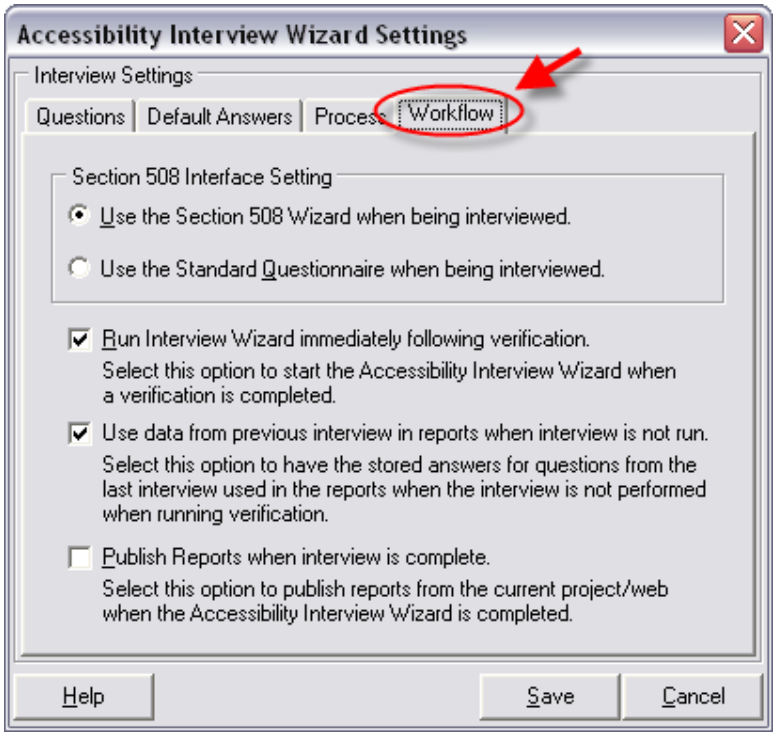



**The Workflow top tab displays**.

29. Select the following:

    - Use the Section 508 Wizard when being interviewed.
    - Run Interview Wizard immediately following verification.
    - Use data from previous interview in reports when interview is not run.

30. Click on the **Save** button.

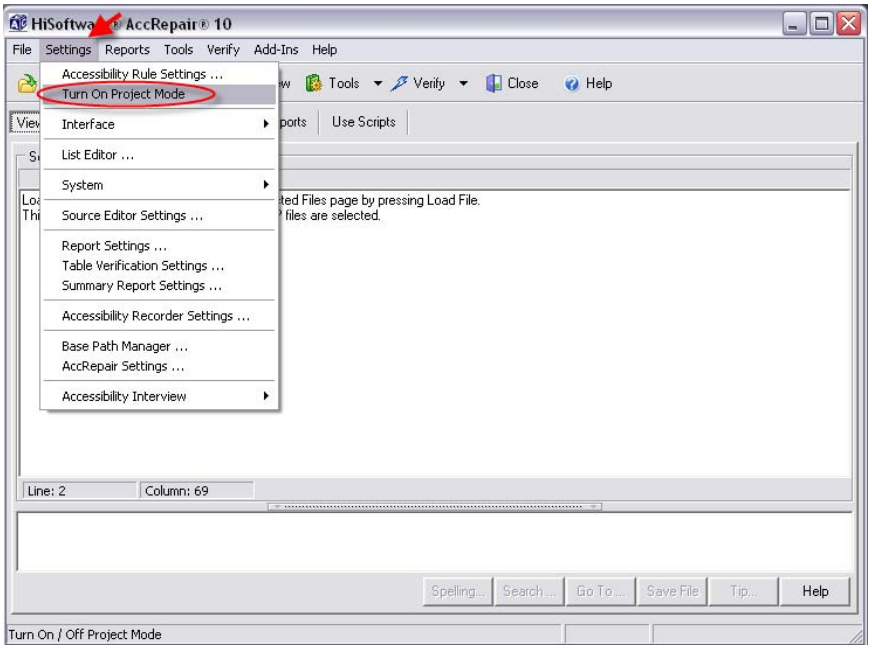


**Accessibility Interview Wizard Settings Saved Successfully!**

31. Click on the **OK** button.

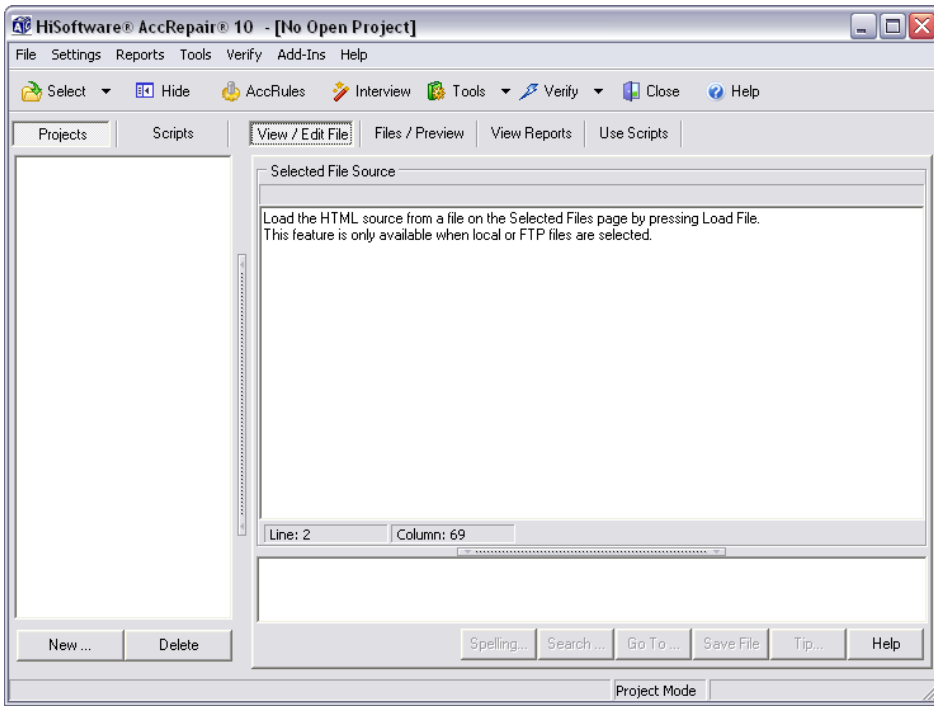
**The AccVerfy Main Menu displays.**

32. Click on **Settings** > **Turn On Project Mode**.



**Project Mode is now on.**

*Note: When Project Mode is on, AccVerify displays three windows.*

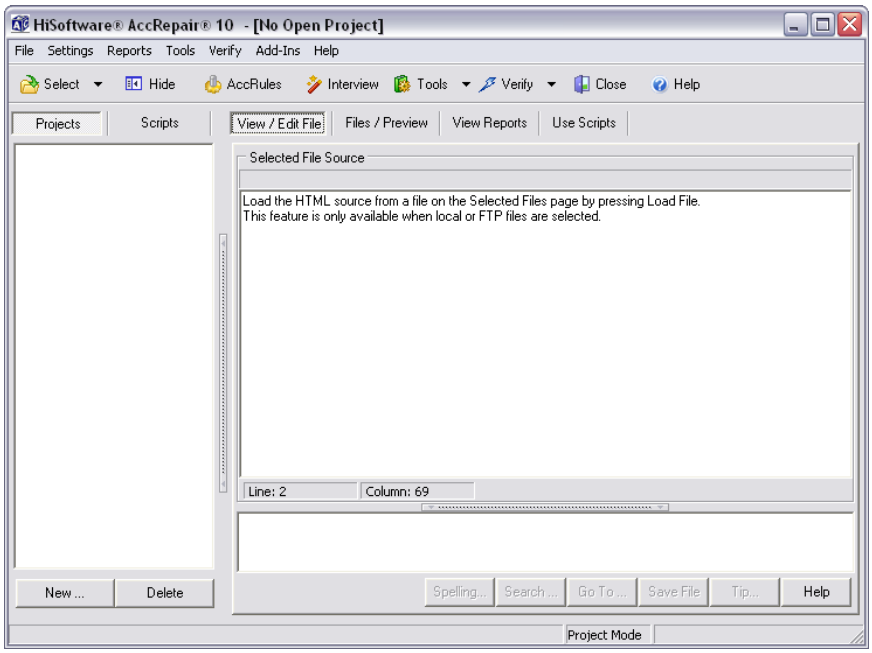


## Run Report

**The AccVerify Main Menu displays.**

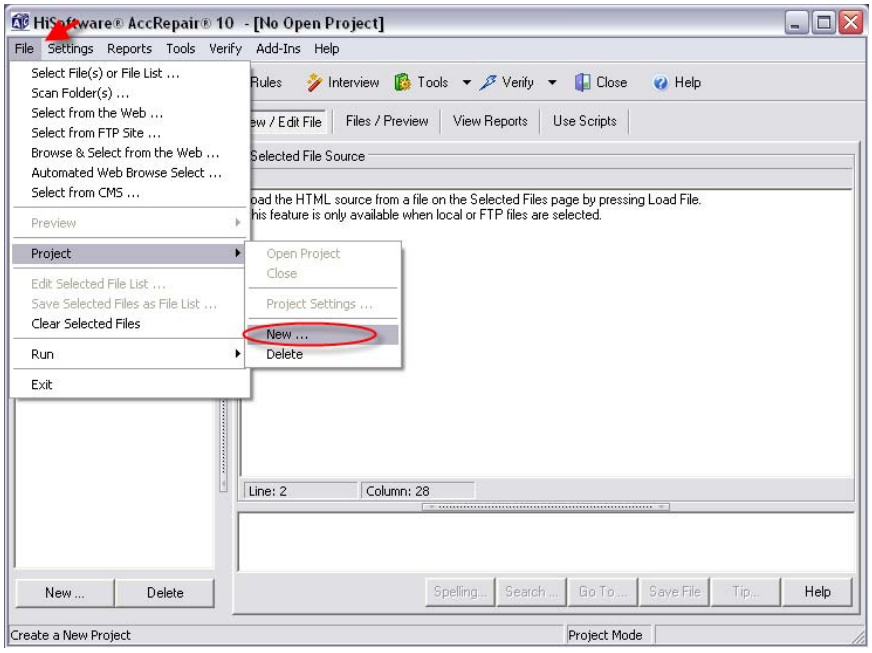
1. Check to ensure that **Project Mode** is on.

*Note: When Project Mode is on, AccVerify displays three windows.*



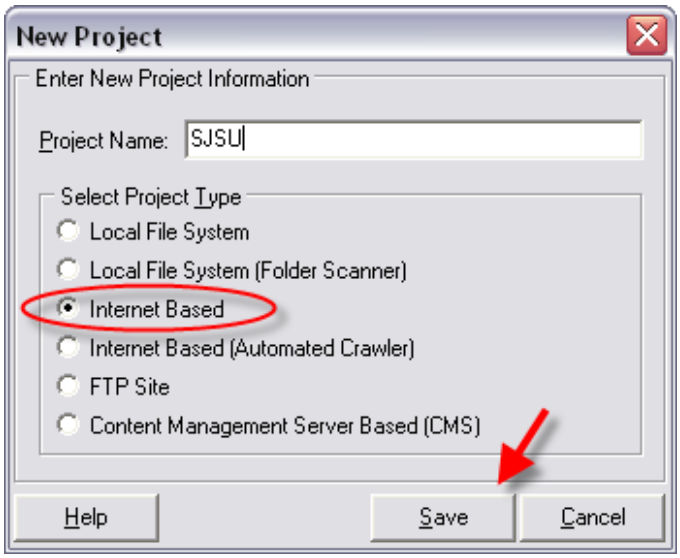

**The AccVerify Main Menu displays with Project Mode on.**

2. Click on **File** > **Project** > **New**.



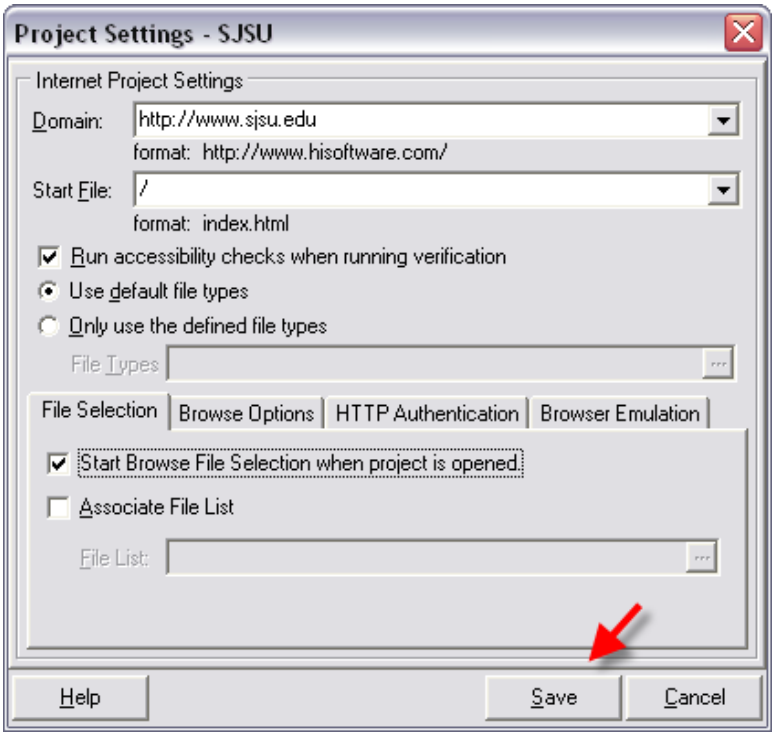
**The New Project window displays.**

3. **Project Name:** Enter your project name.
4. Select **Internet Based**.
5. Click on the **Save** button.



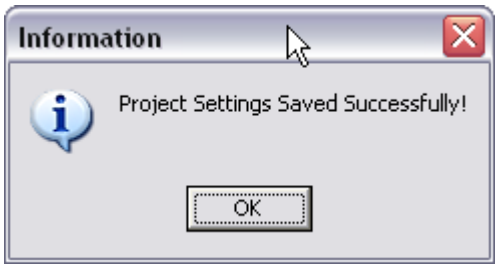

**The Project Settings window displays.**

6. **Domain:** Type in the URL of the web page you want to check.
7. **Start File:** Type in a (slash (/).
8. Make the following selections:
   - Run accessibility checks when running verification.
   - Use default file types.
   - Start Browse File Selection when project is opened.
9. Click on the **Save** button.



**Project Settings Saved Successfully!**

10. Click on the **OK** button.




**The Browse and Select Files from Web for Verification window opens.**

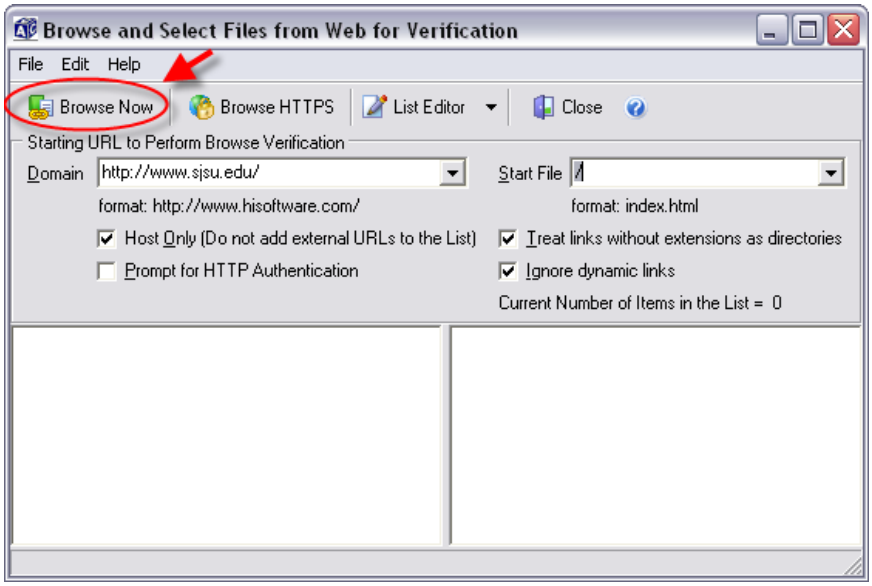
11. **Domain:** Use the drop down menu to select the URL for the web page you want to check.

12. **Start File:** Type in a slash (/).

13. Make the following selections:

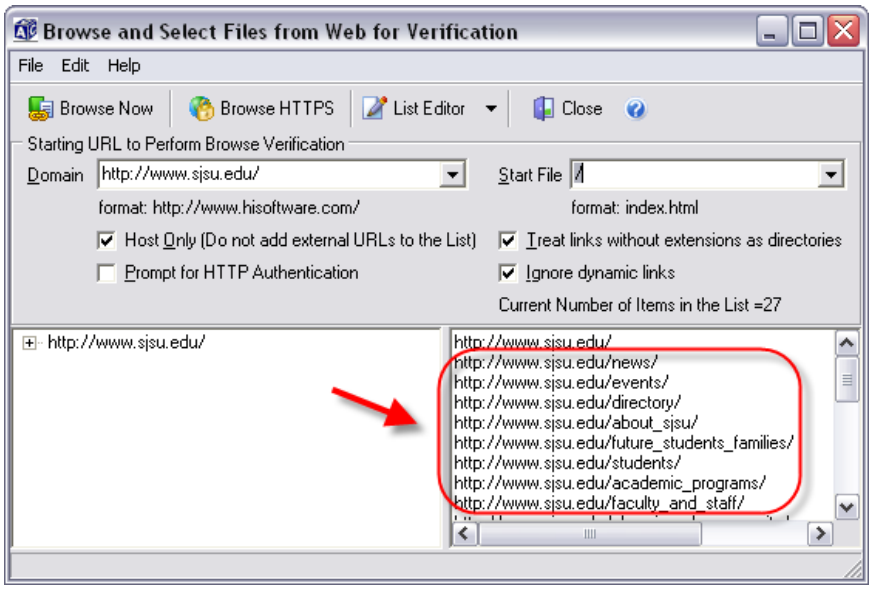
    - Host Only (Do not add external URLs to the List).
    - Treat links without extensions as directories.
    - Ignore dynamic links.

14. Click on the **Browse Now** button.



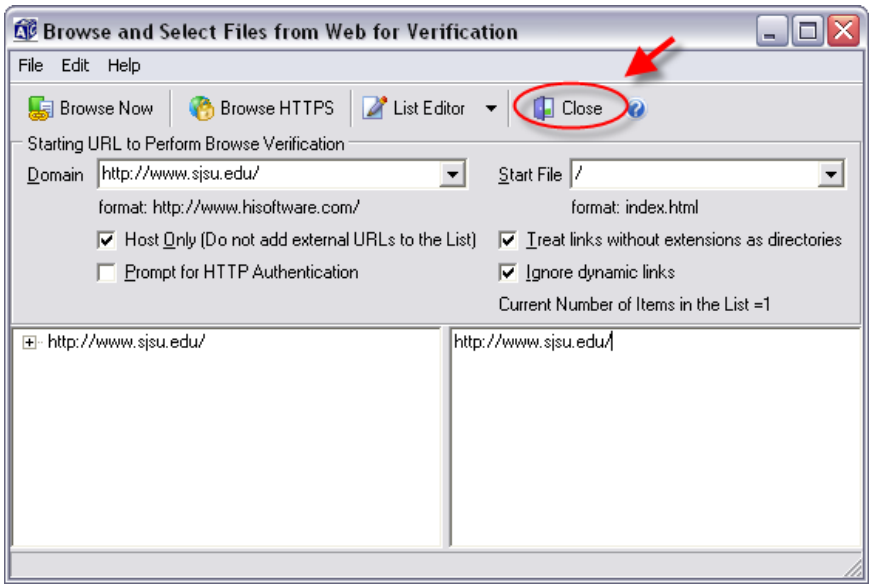
**The URL displays at both bottom left and bottom right windows.**

15. If extra web pages show in the bottom right window, select them, and then click on delete key.




**One web page shows in both right and left windows.**

16. Click on the **Close** button.

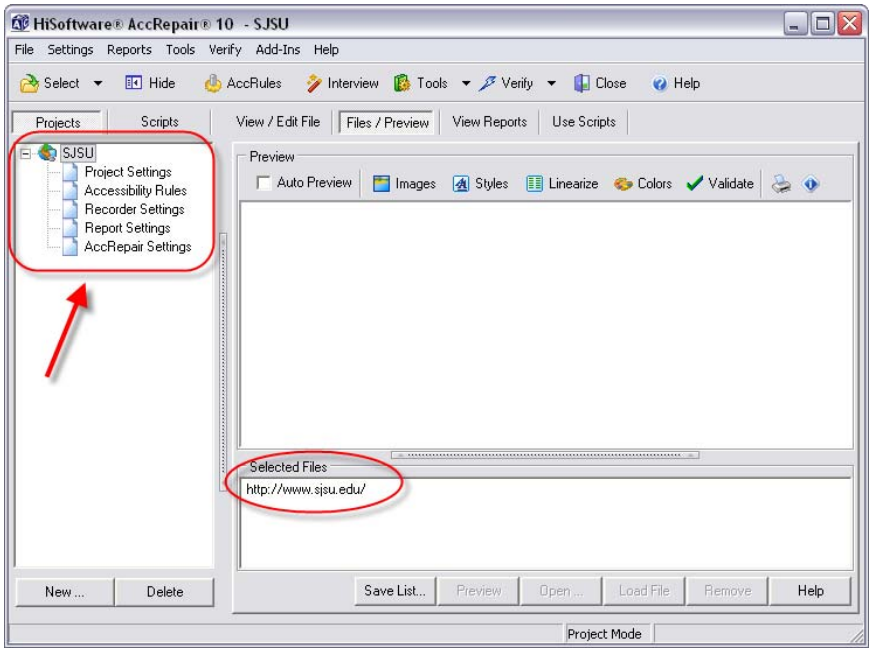


**The Information window displays.**

17. Click on the **OK** button.

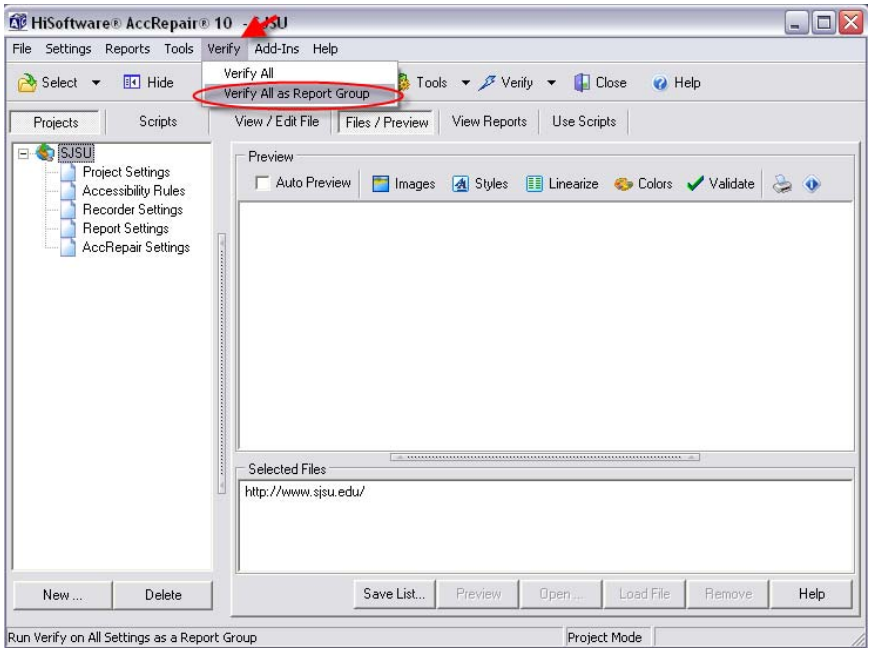
**The Main Menu displays.**

*Note: Your project should be listed in the left window and the URL shows in the bottom window.*



**The Main Menu displays.**

**18.** Click on **Verify** > **Verify All as Report Group**.




**The Report Group window displays**.

19. **Report Group Name:** Type in your project name.

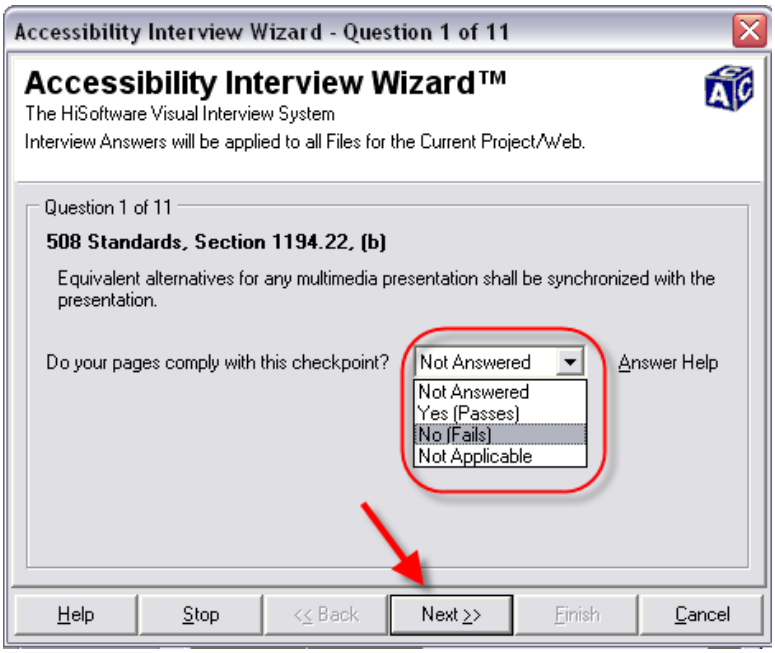
**20.** Click on the **Ok** button.

**The Accessibility Interview Wizard window displays.**

21. Use the drop down menu to select results determined from your manual evaluation in Section 2.

22. Click on the **Next** button.

23. Repeat for each Checkpoint.

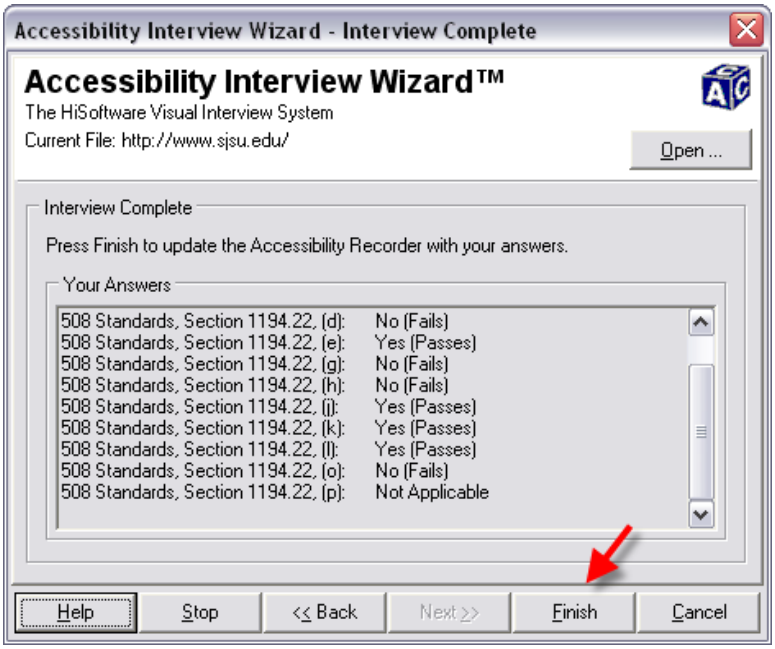



**The Accessibility Interview answers display.**

24. Check the list to make sure your answers are correct.

25. To make changes, click on the **Back** button.

26. When ready, click on the **Finish** button.



**The Information window displays.**

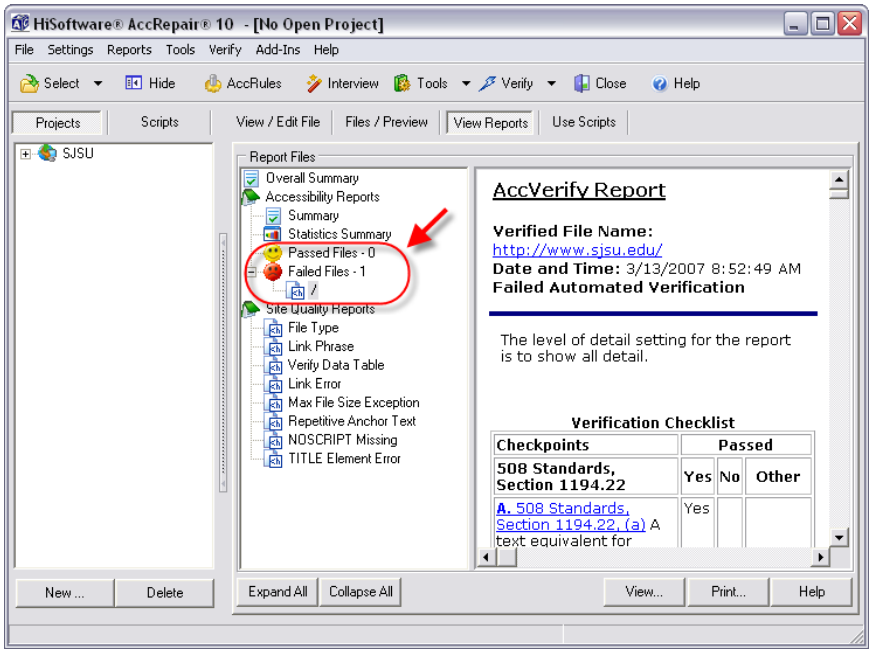
27. Click on the **OK** button.

**The AccVerify Main Menu displays.**

28. Open the **Passed Files** folder.

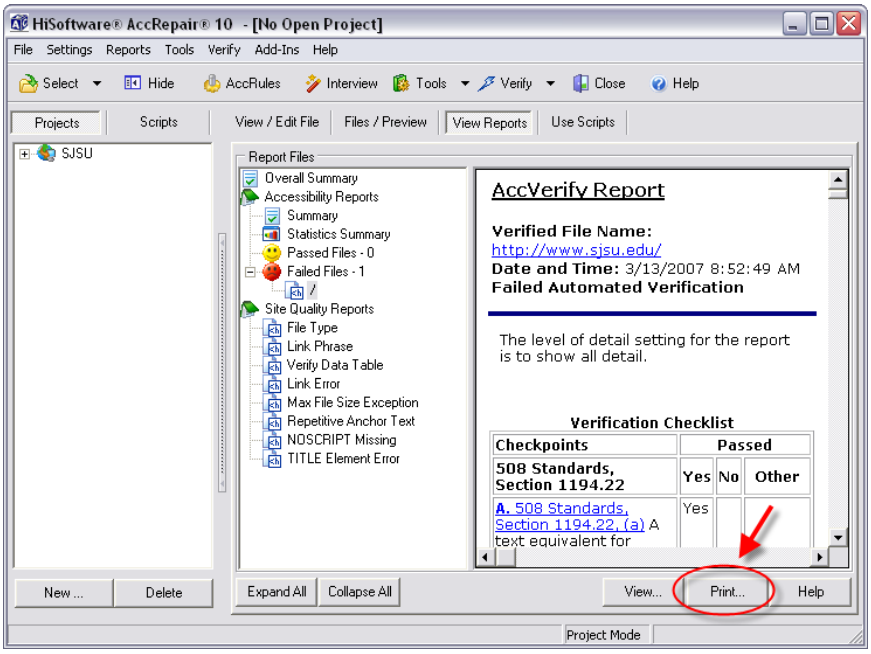
29. Open the **Failed Files** folder.

30. Select your file.



**The Report results display.**

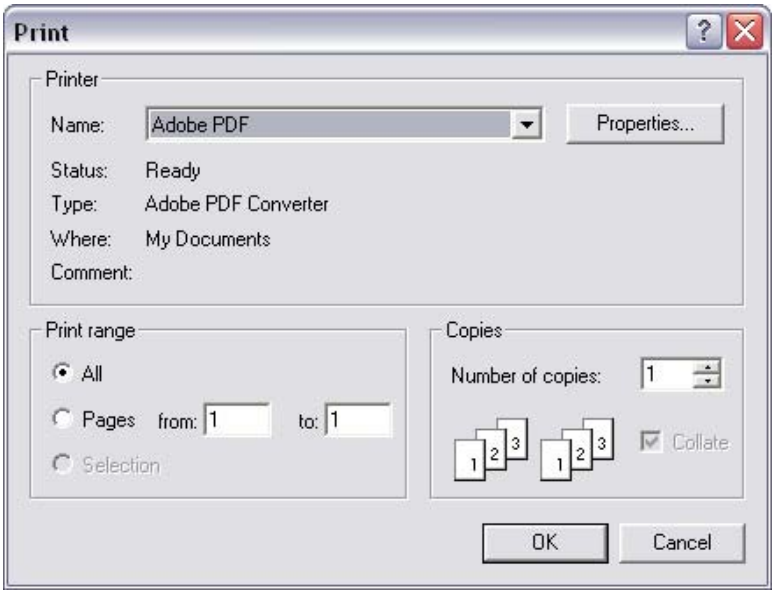
31. Click on the **Print** button




**The Print window opens**.

32. Select **Adobe PDF**.

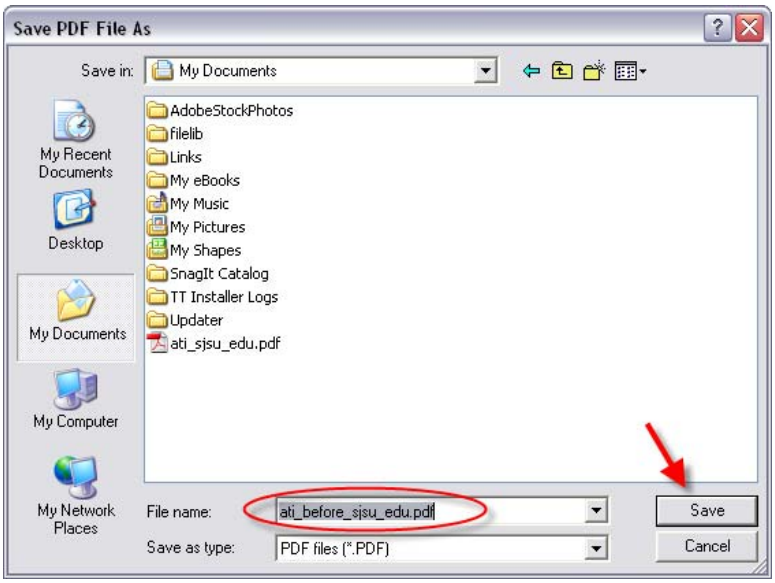
33. Click on the **OK** button.



**The Save PDF File As window opens.**

34. Select any file location you wish.

35. **File Name:** Use the format ***ati*** plus ***before/after*** plus your ***url*** (without the www) when naming your report files.



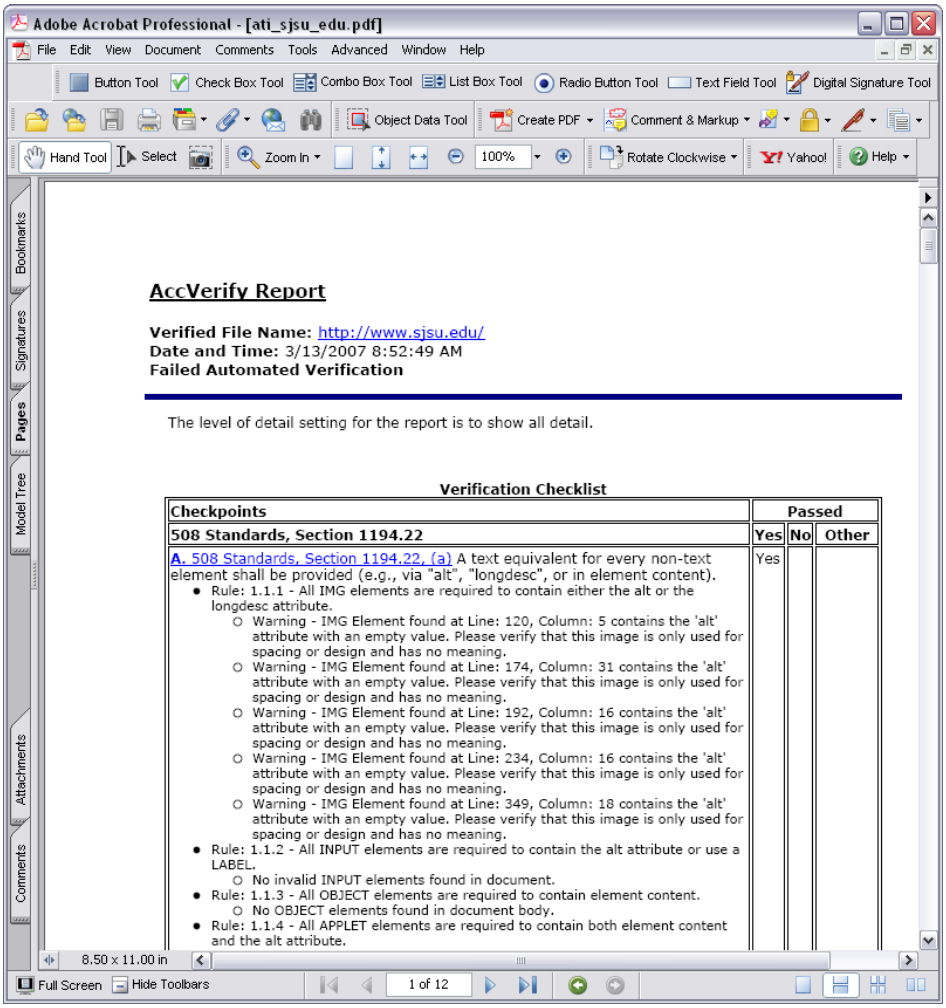

**Example #1:**
URL: http://www.sjsu.edu/hr
Before Retrofit Report: ati_before_sjsu_edu_hr.pdf
After Retrofit Report: ati_after_sjsu_edu_hr.pdf

**Example #2:**
URL**:** http://www.careercenter.sjsu. edu
Before Retrofit Report: ati_before_careercenter_sjsu_edu.pdf
After Retrofit Report: ati_after_sjsu_edu_careercenter.pdf

**Example #3:**
URL: http://my.sjsu.edu
Before Retrofit Report: ati_before_my_sjsu_edu.pdf
After Retrofit Report: ati_after_my_sjsu_edu.pdf

**The AccVerify Report displays.**

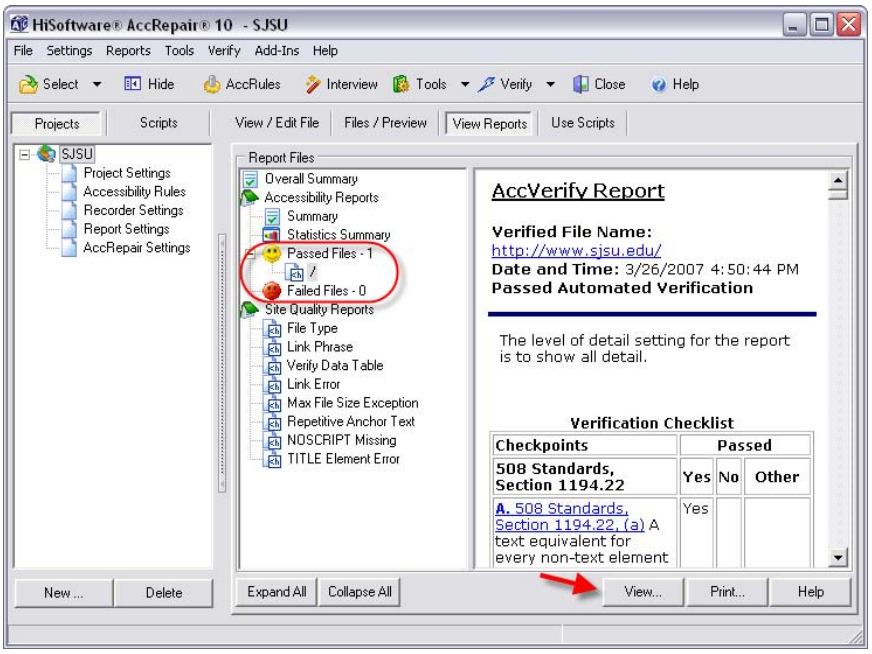
36. Email your report to the ATI Task Force at atiweb@sjsu.edu.



## Understanding and Evaluating Your AccVerify Report

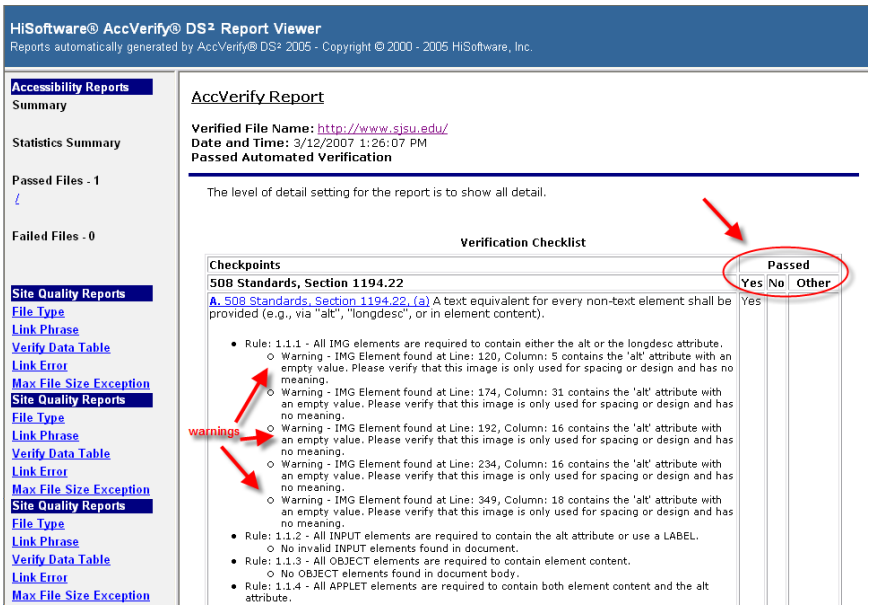
### The AccVerify Main Menu displays.

1. Under the Report Files column, select your report.
2. Click on the **View** button.




### The AccVerify Report displays.

3. To the right of each Checkpoint, the report will indicate if your web page passed, failed or was not applicable (other).
4. Read the information provided for each Checkpoint. It includes warnings and indications about retrofitting your web pages.



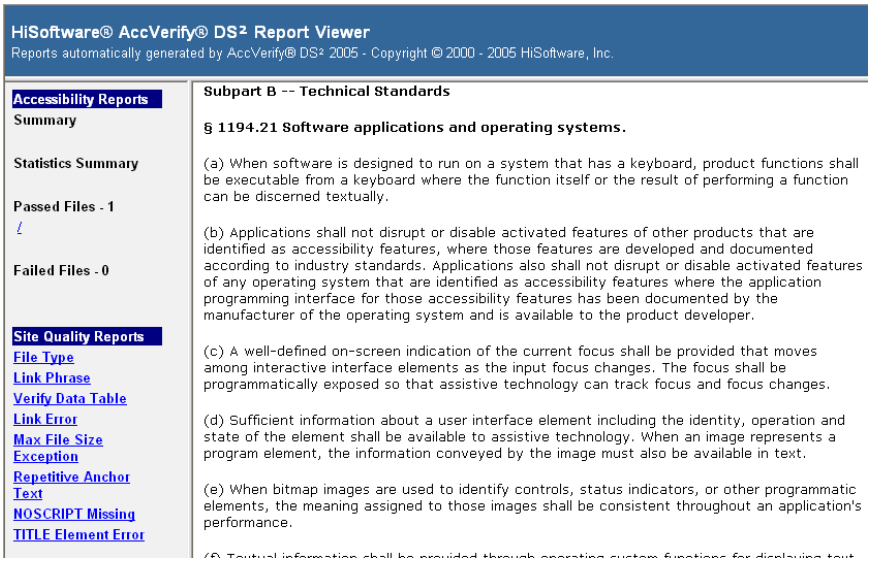
**The AccVerify Report displays.**

5. Notice your report provides a link for each Checkpoint.

6. To obtain more detailed Section 508/Code 11135 information, click on this link.

**Detailed Section 508/Code 11135 information for that Checkpoint displays.**

*Note: Although HiSoftware AccVerify evaluates the automated checkpoints for you, all Webmasters must still manually review these checkpoints to review warnings and to ensure accuracy***.**




7. For assistance in how to make repairs, refer to the book, "Web Accessibility: Web Standards and Regulatory Compliance".

**Web Accessibility: Web Standards and Regulatory Compliance**

By Jim Thatcher, Andrew Kirkpatrick, Richard Rutter, Christian Heilmann, Cynthia Waddell, Michael R. Burks, Shawn Lawton Henry, Bruce Lawson, Mark Urban and Patrick H. Lauke

Published by Friends of Ed, July, 2006, ISBN: 159059638