

# Section 508 Electronic Information Accessibility Requirements for Software Development

Rebecca R. Ellis\*

*University of South Florida, Tampa, Florida, 33620*

**Section 508 Subpart B 1194.21 outlines requirements for operating system and software development in order to create a product that is accessible to users with various disabilities. This portion of Section 508 contains a variety of standards to enable those using assistive technology and with visual, hearing, cognitive and motor difficulties to access all information provided in software. The focus on requirements was limited to the Microsoft Windows® operating system as it is the predominant operating system used at this center. Compliance with this portion of the requirements can be obtained by integrating the requirements into the software development cycle early and by remediating issues in legacy software if possible. There are certain circumstances with software that may arise necessitating an exemption from these requirements, such as design or engineering software using dynamically changing graphics or numbers to convey information. These exceptions can be discussed with the Section 508 Coordinator and another method of accommodation used.**

## I. Introduction

**S**ECTION 508 is an amended portion of the Rehabilitation Act of 1973, which is applicable to electronic and information technology (EIT) that is developed, procured or maintained by federal agencies. Section 508 requires that federal employees and members of the public with disabilities have access to EIT that is comparable to those without disabilities unless and undue burden is placed on the agency. Some of examples of EIT covered by this section the Rehabilitation Act are operating systems and software, web based information, desktop and portable computers, telecommunications products and multimedia products. The term “undue burden” is further defined in Section 508 to mean that there would be significant difficulty or expense in implementing accessibility requirements, however if this is the case Section 508 mandates that other arrangements must be made to provide the individual with access to the necessary information<sup>1</sup>.

Within Section 508 there are four subsections: Subpart A, B, C and D. Subpart A contains general information regarding the scope, application and exceptions of Section 508, Subpart B contains information regarding the technical standards, Subpart C pertains to functional performance criteria and Subpart D contains instruction regarding product documentation, information and support<sup>1</sup>. This document will focus on a specific technical standard within Subpart B, however some requirements discussed will relate to other subparts and sections within Section 508.

A project was assigned with the goal of obtaining information on the requirements for software development contained within section 1194.21 within Subpart B. Research was to be conducted on the different standards set forth in 1194.21 and how those standards would impact a software development team and users of assistive technology. Research was focused entirely on software development on the Microsoft Windows® platform due to the majority of users at Kennedy Space Center using this operating system. The information collected was synthesized into a presentation given at a quarterly meeting regarding Section 508 compliancy. The results of the research conducted and recommendations made regarding implementation will be contained in the remainder of this report.

## II. User Difficulties and Assistive Technology

In order to understand the need for software development requirements for accessibility, one must first consider different types of issues affecting the end user and types of assistive technology used to remediate these issues. For

---

\* Section 508 Electronic Information Accessibility Requirements for Software Development Intern, IT-A, Kennedy Space Center, University of South Florida.

example, users with fine motor difficulty may need to use alternative input devices. Software developers must also be aware of the types of assistive technology that may be interfacing with their product in order to provide appropriate support for the technology in their program and for the user chosen accessibility options in the operating system. With background information on common difficulties and solutions that users have when interacting with software, software development can achieve compliance with ease.

Interacting with, and receiving information from, a computer is primarily done using visual stimuli, hence many of the requirements that will be discussed focus on alternate methods of inputting and receiving information by other methods. Similarly, many types of assistive technology used also provide alternative ways to receive and interpret information from a computer that would usually be displayed visually on a screen. Screen readers are commonly used to provide auditory explanations of what is occurring on screen and are helpful for those with low vision or vision impairments. They are able to read information available on the screen, read the status of elements such as checkboxes, combo boxes, etc., and provide auditory feedback on text entered by the user. The Microsoft Windows® operating system provides a screen reader as part of the accessibility package, however other commercial options such as JAWS (Job Access With Speech) and NVDA (Non-Visual Desktop Access) may be more user friendly and provide enhanced services. For users on the Macintosh operating system, the built in screen reader provides excellent access.

There are many other forms of assistive technology in use in addition to screen readers. Alternative keyboards may be used which may vary in size or provide alternative key placement to increase usability. Electronic pointing devices using ultrasound, infrared or brain wave input are being used, as well as sip and puff devices that operate by the user's inhales and exhales providing input to the software program. Joysticks can be used that are moved by a person's hands, feet or chin, and wands or sticks can be used to press keys using a person's mouth, head or hand. Trackballs, which are movable balls mounted on a stable base, can increase usability for someone who has difficulty using a mouse. Touch screens can be used instead of providing input with a keyboard or mouse and can be helpful for someone with low vision, motor or cognitive difficulties <sup>2</sup>.

Voice recognition programs can enable users to use speech as an input to operate a program if the keyboard and mouse cannot be used, and keyboard filters can enable a user to operate a keyboard successfully if using a wand or experiencing hand tremors. Refreshable braille displays can be used to output textual software information into braille format as an alternative to a screen reader. There are also assistive technologies available for users who may have cognitive impairments as well. These technologies can allow the user to see highlighted text as it is read, to have text reformatted or alternative methods of navigation <sup>2</sup>.

The assistive technologies discussed above are not the only options available, however they give the software developer an idea of the types of devices that may need to interact with software created. Developers have two options to provide compliance with Section 508 1194.21; they may write software that is compatible with assistive devices or they may make their program completely accessible without any need for additional assistive technology. As one can see from the list above, it would prove far simpler to provide support to assistive technologies within the program rather than attempt to write software for every specific user's need that may arise.

### **III. Section 508 1194.21 Provisions**

Section 508 1194.21 contains twelve different provisions, (a) through (l), regarding software development, some of which overlap in their specifications. This portion of the report will be dedicated to reviewing the provisions and their implications for software developers. It should be noted that these provisions also apply to software bundled with other products, such as printer or copy machine software used on a computer to access those devices, as this is often forgotten <sup>3</sup>. Microsoft Windows® provides Application Programming Interfaces (API's), which allow a program to use standardized code to perform certain functions and interact with the operating system. By using API's, the programmer can easily increase compatibility with current and future assistive technology. API's allow the assistive technology to communicate with other applications in a standardized way using the operating system to relay the information between them.

#### **A. Provision (a)**

Provision (a) states "When software is designed to run on a system that has a keyboard, product functions shall be executable from a keyboard where the function itself or the result of performing a function can be discerned textually."<sup>3</sup> This can be the most difficult provision to implement, especially in legacy software, as it requires that the entire program can be accessed and run from only a keyboard. It also requires that when a user takes an action there is textual feedback than can be accessed by a screen reader. For example, if the user presses the delete icon, a message box may ask if the user would like to delete this item. This provides a chance for a user to reverse an

action in the case it was taken in error and to tell when they have initiated an action within the software program. User who are navigating the program with a screen reader and those who are using alternative input devices are able to use the software program more effectively with these provisions in place. There is an exception, however, if the system in question was not designed for use with a keyboard. Kiosks and other devices without keyboards that use software to provide information would fall under a different standard other than 1194.21.<sup>3</sup>

#### **B. Provision (b)**

Provision (b) states "Applications shall not disrupt or disable activated features of other products that are identified as accessibility features, where those features are developed and documented according to industry standards. Applications also shall not disrupt or disable activated features of any operating system that are identified as accessibility features where the application programming interface for those accessibility features has been documented by the manufacturer of the operating system and is available to the product developer."<sup>3</sup> This provision is very important for users of assistive technology, as programs should be written to not interfere or disable their assistive technology or accessibility settings. Some examples of accessibility features that can be turned on in Microsoft Windows® are a screen reader, a screen magnifier, high contrast settings, and sticky or toggle keys. Sticky keys allow a key, such as shift, control or alt, to remain active until another key is pressed, which is useful if a person is only able to press one key at a time and needs to execute a command requiring multiple key combinations. Toggle keys will produce a sound when a key such as Caps Lock or Num Lock is pressed.<sup>3</sup>

#### **C. Provision (c)**

Provision (c) states "A well-defined on-screen indication of the current focus shall be provided that moves among interactive interface elements as the input focus changes. The focus shall be programmatically exposed so that assistive technology can track focus and focus changes."<sup>3</sup> This means that the focus, for example the placement of the cursor or caret, is clearly visible and is programmatically visible to assistive technology that is being used. When a user has a screen reader activated, the screen reader should be able to provide auditory output to enable the user to insert text in the correct location, or execute the right command by pressing enter.<sup>3</sup>

#### **D. Provision (d)**

Provision (d) states "Sufficient information about a user interface element including the identity, operation and state of the element shall be available to assistive technology. When an image represents a program element, the information conveyed by the image must also be available in text."<sup>3</sup> Images that are used in a program must be labeled with appropriate text so that if a screen reader is in use, the information contained in the image can be read to the user. User interface elements, for example checkboxes, should have their identity and state available to assistive technology. Using checkboxes as an example, if they were implemented correctly in the program a screen reader would be able to detect their state, checked or unchecked, and where the focus is. This is also useful for speech input devices so the user can instruct the program to check a checkbox using voice commands.<sup>3</sup>

#### **E. Provision (e)**

Provision (e) states "When bitmap images are used to identify controls, status indicators, or other programmatic elements, the meaning assigned to those images shall be consistent throughout an application's performance."<sup>3</sup> This provision is fairly straightforward, and most software developed will probably already be compliant. Images that are used in the program used to identify elements should not be changed partway through the program. Graphics should also be consistent with user expectations as well, for example a save icon should not be used to identify a delete button, and should instead be used to save the item. This provision does not apply to images that do not identify programmatic elements<sup>3</sup>.

#### **F. Provision (f)**

Provision (f) states "Textual information shall be provided through operating system functions for displaying text. The minimum information that shall be made available is text content, text input caret location, and text attributes."<sup>3</sup> Screen readers and other devices rely on text being provided through the operating system, and the developer can accomplish this by using API's provided in Microsoft Windows®. Developers do have the option to provide text through their own means if the assistive technology can still access the text through operating system functions<sup>3</sup>. If the screen reader is unable to access text through the operating system, compatibility becomes an issue and the user may not be able to use the assistive technology.

#### **G. Provision (g)**

Provision (g) states "Applications shall not override user-selected contrast and color selections and other individual display attributes."<sup>3</sup> Users may opt to use certain accessibility settings within Microsoft Windows®, such as certain contrast options and colors, and programs developed should not interfere with these settings. Programmers should allow the users contrast and color settings to be used in their program. There is an option to provide the user with custom contrast and color settings within a program developed, however if this option is chosen a variety of options must be made available. For example, some users have difficulty seeing certain sets of color, such as red and green, and may choose to avoid using these combinations when viewing information in a program. Other users may not be able to focus on high contrast, bright text, and may choose to use alternative contrast settings. The developer creating software should respect these choices, and failure to allow these settings to be used in the program can render them unusable.

#### **H. Provision (h)**

Provision (h) states "When animation is displayed, the information shall be displayable in at least one non-animated presentation mode at the option of the user."<sup>3</sup> Many images can easily be tagged in a way to present the information contained in the image in a text or audio format in order to comply with this provision. Some difficulty may be had when attempting display animations that contain unpredictable and spontaneous information, such as animated weather radar maps. In these cases it may be necessary to seek an exception with the Section 508 coordinator in order to find an alternative method to convey this information to the user.

#### **I. Provision (i)**

Provision (i) states "Color coding shall not be used as the only means of conveying information, indicating an action, prompting a response, or distinguishing a visual element."<sup>3</sup> Due to the difficulty that some users have with distinguishing colors, alternative methods of prompting the user to take an action or to differentiate items in a program should be used. For example, a message box may contain a red exclamation point indicating a user needs to correct an item. Some users, especially those using a screen reader, may not be able to interpret this correctly so alternative text may be needed. Color-coding in graphs and charts should also have alternative text in order to distinguish one element from another.

#### **J. Provision (j)**

Provision (j) states "When a product permits a user to adjust color and contrast settings, a variety of color selections capable of producing a range of contrast levels shall be provided."<sup>3</sup> This provision ties back into provision (g) regarding using a chosen color and contrast setting by the user. If the software developer decides to allow the option to adjust colors or contrast settings in the program, they must allow for a variety of color choices and a variety of contrast options. This is to accommodate those with difficulty viewing certain colors or contrasts.

#### **K. Provision (k)**

Provision (k) states "Software shall not use flashing or blinking text, objects, or other elements having a flash or blink frequency greater than 2 Hz and lower than 55 Hz."<sup>3</sup> This provision can be easily implemented by developers by avoiding flashing elements if possible. If these elements are used, the developer must take into account the cumulative effect of multiple flashing or blinking elements on screen within the program, and allow for variances with individual computer processing speeds and graphic card capabilities which can effect the blink rate.

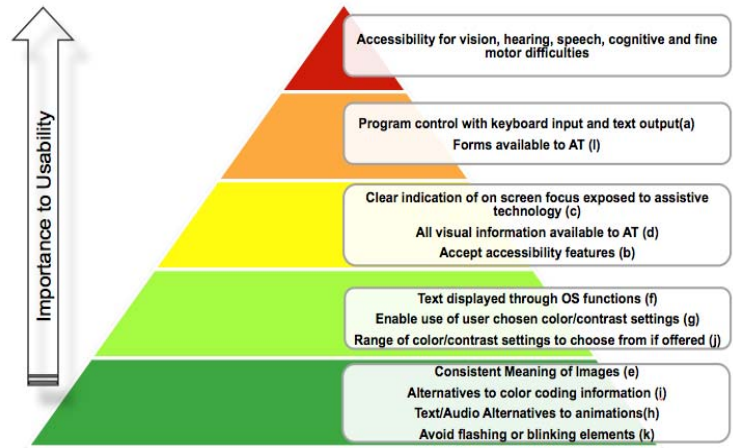
#### **L. Provision (l)**

Provision (l) states "When electronic forms are used, the form shall allow people using assistive technology to access the information, field elements, and functionality required for completion and submission of the form, including all directions and cues."<sup>3</sup> This provision is very similar to a provision regarding form accessibility and website development. Form elements should be clearly labeled, and any graphics should be labeled with alternative text. Shortcut keys to access different form elements can be helpful for users who rely on alternative input devices or keyboard input alone. All elements within the form, from filling it out to submission, should be available to assistive technology.

### **IV. Implementation of Requirements**

Initial implementation of the above requirements may be challenging for developers as all requirements are considered equally important. In order to create a path to follow for creating new software that is compliant and remediating legacy software, a hierarchy was created to show the final result desired, and how to achieve that goal.

Figure 1 shows that at the top of the pyramid, accessibility for all is the ultimate goal. However, in order to obtain that goal, a solid base must first be built. Some requirements will be simpler to implement, and if they are implemented correctly it allows additional requirements to be met as they build off one another. As the top of the pyramid is approached, the importance to usability also increases. Software developers can integrate testing of software with assistive technology and Section 508 requirements into the development life cycle. This will allow increased compliance with requirements, and is more cost effective as software can be created correctly from the start with minimal remediation necessary.



**Figure 1. Hierarchy of Requirements**

## V. Conclusion

While the requirements for compliance with Section 508 1194.21 are numerous, creating software that is compliant is necessary to provide access for users of all abilities. Software that is highly accessible for those using assistive technology will also increase usability for standard users by making a product consistent and simple to navigate. Many of the requirements can be fairly easy to implement, however there may be circumstances that arise where software will be unable to comply directly with some of the requirements due to varying factors. In cases such as these, there may be alternative options that can be used with assistance of the Section 508 Coordinator.

## Acknowledgments

The author thanks R. H. Liang for many stimulating discussions and guidance on the subject matter.

## References

- <sup>1</sup>United States Access Board, "Section 508 Standards for Electronic and Information Technology," URL: <http://www.access-board.gov/guidelines-and-standards/communications-and-it/about-the-section-508-standards/section-508-standards> [cited 2 July 2014]
- <sup>2</sup>Microsoft®, "Types of Assistive Technology Products," URL: <http://www.microsoft.com/enable/at/types.aspx> [cited 2 July 2014]
- <sup>3</sup>Internal Revenue Service, "Section 508 For Software Development" URL: <http://training.section508.gov/IRSCourse/mod02/020101--.html> [cited 17 July 2014]