GUIDELINES FOR WRITING ACCESSIBLE PROCEDURES

REAL, NOT TECHNICAL, ACCESSIBILITY

RELATED DOCUMENTS

Plagiarism Prevention Using Turnitin: A Pedagogical Approach Using Turnitin
Plagiarism Prevention Using Turnitin: Instructor Procedures
Plagiarism Prevention Using Turnitin: Sample Syllabus Statements
Plagiarism Prevention Using Turnitin: sample student instructions
TABLE OF CONTENTS

PREFACE .......................................................................................................................... 3
Accessible Procedures ................................................................................................. 4
Using a Screen Reader ................................................................................................. 4
Alternate Text ................................................................................................................ 4
System Accessibility Improvements ........................................................................... 5
Accessible Language ..................................................................................................... 5
Good Faith Effort .......................................................................................................... 5

These guidelines are provided by the Center for Teaching and Faculty Development as part of their accessibility resources for Universal Design for Learning. Universal Design for Learning is a process of making course concepts accessible and skills attainable regardless of learning style, physical, or sensory abilities. Providing accessible instructional materials is for the benefit of everyone as accessibility removes the barriers individuals may face from obtaining the information.

For additional information on Universal Design for Learning, please visit the Universal Design for Learning web site online at http://ctfd.sfsu.edu/udl. For additional accessibility resources, please visit the Center for Teaching and Faculty web site online at http://ctfd.sfsu.edu/accessibility-resources.htm.
PREFACE

This document is based on my experience developing user procedures for the integration of the commercial plagiarism detection product Turnitin with iLearn, San Francisco State’s version of the Moodle open-source learning management system. While there is some documentation available for Moodle, and detailed documentation on Turnitin, there were no written procedures for the combined product. Further, software developers, both commercial and open-source, try to reach the widest possible user audience. Their documentation therefore features all possibilities. I am trying to offer instructions to support a specific pedagogical approach, so my goal has been far more restrictive and prescriptive than typical software procedures.
ACCESSIBLE PROCEDURES

Document accessibility has been defined mostly in technical terms: text must be formatted with styles, graphics need alternate text descriptions, tables (if used at all) must have headings, and colors must work for readers with visual limitations. And while we must follow these guidelines for our texts to be at all accessible, we must also go beyond mere technical accessibility to make our content equally welcoming and useful for all. This true accessibility is particularly important for procedures for using software. Most current instructions, even for accessible systems, assume a sighted user with a mouse, leaving those who employ screen readers to fend for themselves. Here are a few points that may serve as the beginning of the exploration into true accessibility

USING A SCREEN READER

While screen reading software works immediately, it is expensive and requires hours of training for users to become fluent. Still, procedures writers and systems designers must take the time to at least set with experienced users, watching them work through the procedures and processes. We do this for the average user as part of usability analysis; we must do this for every method of navigation. Guidelines for accessibility only become truly meaningful after you have experienced the open floodgates of the machine reading hundreds of links, or discovered that “you can’t get there from here.”

ALTERNATE TEXT

We know that we need useful alternate text to describe graphics, and accessibility guides give both examples and rules. For simple images, we are advised to create brief labels. **Yet when we are writing user procedures, the alternate text needs to match the alternate text in the software program.** To link to its academic integrity report, Turnitin provides sighted-users what appears to be a button that indicates what percent of the text has matches on its database.  

While Turnitin’s own documentation calls this a report box, screen readers pick up “download link.” To serve both users, we need the alternate text to refer to something like “download link for the report.”
SYSTEM ACCESSIBILITY IMPROVEMENTS

We may uncover inequalities through the deliberate iterative process of documenting system procedure. Students submit their essays to Turnitin and then wait for the software to produce a report. While the report is being processed, sighted students are given the visual clue of a grayed-out report box; those using screen readers get nothing. A truly accessible system would provide report status information for all users.

Similarly, integrating two accessible systems doesn’t automatically result in an optimally accessible system. Turnitin opens up in a frame, but the first version of the integration didn’t give a title to the frame, so someone with a screen reader had no quick way to navigate to the frame.

ACCESSIBLE LANGUAGE

Most user procedures say “click” even though the term only applies to someone working with a mouse. Perhaps it is my background—feminist, human resources specialist, writing instructor—but I feel such language is as exclusionary as the age- and gendered-terminology that we as a society have largely dropped. We receive deliveries from a postal carrier, not a mailman, and buy items from sales clerks, not sales girls. I have chosen the comparatively dull term “select” over the snappier “click” as a move toward a language of inclusion.

GOOD FAITH EFFORT

These suggestions are just a start toward making system documentation as accessible as the systems themselves. As we continue to think about all students and all teachers, we will undoubtedly become aware of the many more improvements needed.