# Human factors guidance notes

# Testing methods for plain language requirements

This VVSG 2.0 Requirements Guidance Note summarizes major changes that were made in sets of requirements that work together and which may be organized under different principles.  These changes were based on current research and changes in technology in conjunction with input from the NIST Human Factors Public Working Group (HF PWG). The guidance also serves to inform readers who were not participants in the HF PWG.

This guidance note applies to:

| **VVSG 2.0 Requirement** | **VVSG 1.1 Requirement** |
| --- | --- |
| 7.3-O – Plain language | 3.2.4.c, 3.2.8.a, 3.2.4.c,ii, 3.2.4.c.iii, 3.2.4.c.v, 3.2.4.c.vi, 3.2.4.c.vii |
| Related requirements |  |
| 7.3-J – Warnings, alerts, and instructions | 3.2.4.c.i, 3.2.4.c.iv |
| 7.3-M – Instructions for voters | 3.2.4.a, 3.2.4.b, 3.2.4.e.iv, 7.8.6.g |
| 7.3-N – instructions for election workers | 3.2.8.1.c, 3.2.8.1.c.i, 3.2.8.1.c.ii, 3.2.8.1.c.iii |
| 8.3-A – Usability for voters | 3.3.1.a, 3.3.3.a, 3.3.3.a.i , 3.3.3.10.a, 3.3.3.10.b, 3.3.3.10.c, 3.2.7.a.iv |
| 8.4-A – Usability for election workers | 3.2.8.1.a, 3.2.8.1.b, 3.2.8.1.b.i |
| 2.2-A – User-centered design process | New |

## Why the requirements were updated

The original plain language requirements in VVSG 1.0 and 1.1 were easy to understand, but hard to test.

These plain language guidelines were incorporated into requirements in VVSG 1.0 and 1.1 as a “SHALL” requirement for plain language with a list of “SHOULD” guidelines under it. Structurally, this was a challenge for conformance testing because it was not clear how to interpret the guidelines. For example, would a single instance of not meeting the guideline be cause to fail the higher level requirement?

## How the requirements have changed

For VVSG 2.0, the requirement itself has been simplified to focus on the goal of plain language. This simplification is paired with a more robust approach to testing the requirement.

To make the structure and goal of the requirement clearer, 7.3-O has been simplified to the core requirement for plain language, with the guidance moved to the discussion.

Several VVSG 2.0 principles require voting system messages, notices, and documentation to be written clearly, so voters and election workers understand the information they need to ensure a successful voting experience. This includes all the information:

* voters will see on a ballot, including instructions and ballot choices
* election workers will see, such as error messages and documentation that comes with the system

The primary plain language requirement is 7.3-O:

“Information and instructions for the voter must be written clearly, following the best practices for plain language. Messages generated by the voting system for election workers in support of operation, maintenance, or safety of the system must also follow plain language best practices.”

Three other requirements rely on plain language for creating the required instructions:

* 7.3-J – Warnings, alerts and notifications
* 7.3-M – Instructions for voters
* 7.3-N – Instructions for poll workers

In addition, three requirements cover evaluation of usability of the voting system, which will also test the usability and plain language of information in the voting system:

* 8.3-A – Usability for voters includes any instructions, messages or notification voters encounter in using the voting system
* 8.4-A – Usability for election workers relies on plain language in writing usable instructions for setup, operation, and shutdown of the voting system
* 2.2-A requires documentation of a user-centered design approach that includes considering plain language.

## Recommendations for changing the test method for plain language

Making this change requires a more robust and detailed test method to ensure that all of the information in the voting system meets best practices for plain language and the intent of the requirement.

In discussing how to present the plain language requirements more effectively in VVSG 2.0, the Public Working Group recommended a two-part testing method:

* Using an automated test as part of the “pre-flight” checks – with readily available tools that vendors can use to prepare for entering the certification process.
* Developing a test method for human review, based on a manual review that focuses on the effective application of plain language best practices.

## How to create and evaluate information in plain language

Creating information in plain language is a multi-step process that includes reviewing the text against plain language guidelines or using an evaluation program that makes recommendations, editing the document, having subject matter experts review the document, and having readers try using the information in a usability test.

The challenge for testing is that there are no absolute plain language requirements – for each best practice guideline, there are always exceptions where breaking the rule makes the information clearer. Modern testing tools acknowledge this by providing a range. For example, they might suggest no more than a few passive sentences in a document, based on the overall length of the text.

Similarly, grade level ratings can tell you if a text contains too many multi-syllable words or long sentences, but cannot tell you whether it is understandable. The grade-level algorithms are particularly difficult to use when assessing election information which may contain legally required words that might not be completely “plain” (for example, “jurisdiction”). A manual review can determine whether these words are used appropriately or explained in context.

A manual review also looks for best practices that cannot be automatically tested, such as whether steps in a process are in the right order, or an error message includes information about how to correct the problem.

The recommended two-part evaluation method allows for a multi-step evaluation process that mirrors the process of creating the information:

* An initial automated test with a software program provides an overview of how well the information meets basic plain language best practices.
* A manual review looks at the problems found in the automated test to see if they can be justified or if they should be revised.
* The manual review also looks at the text for best practices in organizing the information.
* Finally, the usability tests in 8.3-A and 8.4-A test the information as it is used by voters or election workers to complete typical election tasks.

## Guidelines for writing instructions and messages

A white paper, *NISTIR 7596-Guidelines for Editing Clear Instructions and Messages for Voters and Poll Workers* (2009), compiled the following list of guidelines for writing clear instructions and messages for voters and election workers.

**Guidelines for clear instructions on ballots placement**

1. Put instructions where they are needed – not all together at the top.
2. Put instructions before they are needed – not after.

**Order**

1. Put instructions in logical order. First task, first; last task, last.
2. Put warnings about consequences before – not after – the voter is likely to act.
3. On electronic interfaces, wait to highlight the option to vote until voters have been through all the races and measures.
4. On electronic interfaces, match the order of buttons to the order of the instructions

**Sentences**

1. Start each instruction on a new line.
2. Write directly to the voter.
3. Keep each instruction as short as possible.
4. Watch the tone. Help voters; don’t threaten them.
5. Write in the positive.
6. Put the context before the action.
7. Be consistent in the way you give instructions.

**Words**

1. Do not use gender-based pronouns.
2. Use simple English words that voters know
3. Be consistent in the words you use.
4. For electronic interfaces, do not use technical, computer jargon.
5. For electronic interfaces, do be explicit in naming buttons.

**Topics**

1. Cover all important situations.
2. Consider voters' likely mistakes.

Additional guidelines for plain language include the [Federal Plain Language guidelines](https://www.plainlanguage.gov/guidelines/) found on<https://www.plainlanguage.gov> along with other tools and resources.

## Plain language evaluation tools

As part of the initial work on the test methods required by this approach, we found several commercial tools that can help begin a document evaluation quickly and easily. We looked for tools that could handle short texts like error messages effectively and which evaluated information using the best plain language practices for your voting system, such as those listed here.

* Sentence and word length
* Passive verbs and hidden verbs
* Adverbs
* Words easily misused
* Complex phrases
* Duplicate or unnecessary words
* Jargon

We tried using these tools that can help you ensure a clearly written document, and we describe how you will find them useful.

We began by consulting plain language experts, asking what tools they use. We then tested the tools using sample ballot language. We tested them for:

* ease of use
* usefulness of the feedback
* ease of making suggested changes

We are sharing those results below. The tools reviewed include:

* Hemingway Editor
* Visible Thread
* Editsaurus
* White Smoke
* Style Writer

Other programs, not reviewed in detail, include:

* [readable.io](https://readable.io/)
* [Word Rake](https://www.wordrake.com/)
* [Acrolinx](https://www.acrolinx.com/)
* [Grammarly](https://app.grammarly.com/)

These summaries are not recommendations or endorsements, but examples of available products.

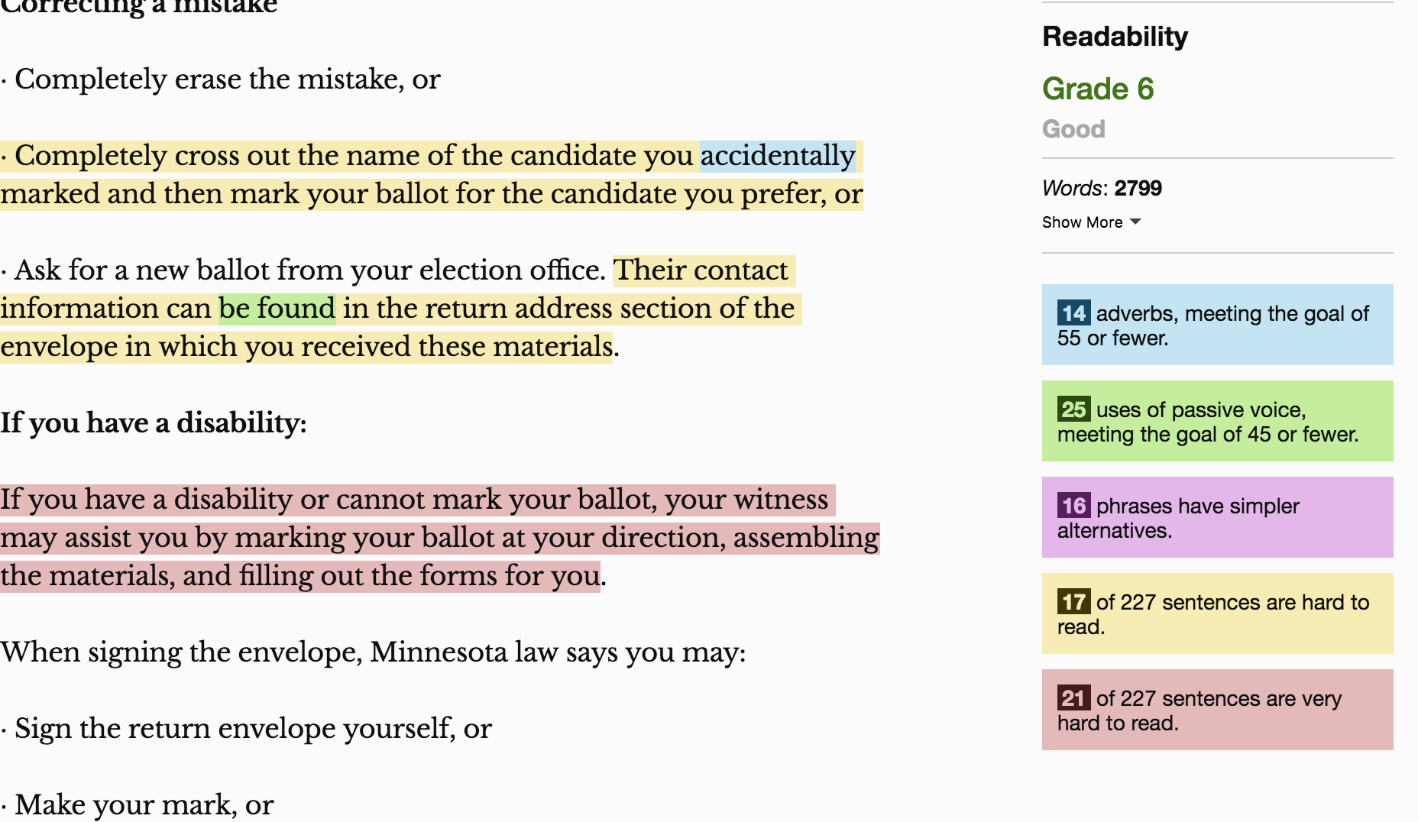
We continue to look for useful plain language evaluation tools. If you have any contributions to this list, please contact Sharon Laskowski at NIST.

### Hemingway Editor

[Hemingway Editor](http://hemingwayapp.com/) highlights many plain language criteria such as excess words, passive voice, hard to read sentences, and phrases with simpler alternatives. It color codes problem areas and gives a readability score.

**Cost** – There is a free online interface. To purchase: $20

**Sample of the Hemingway Editor results**



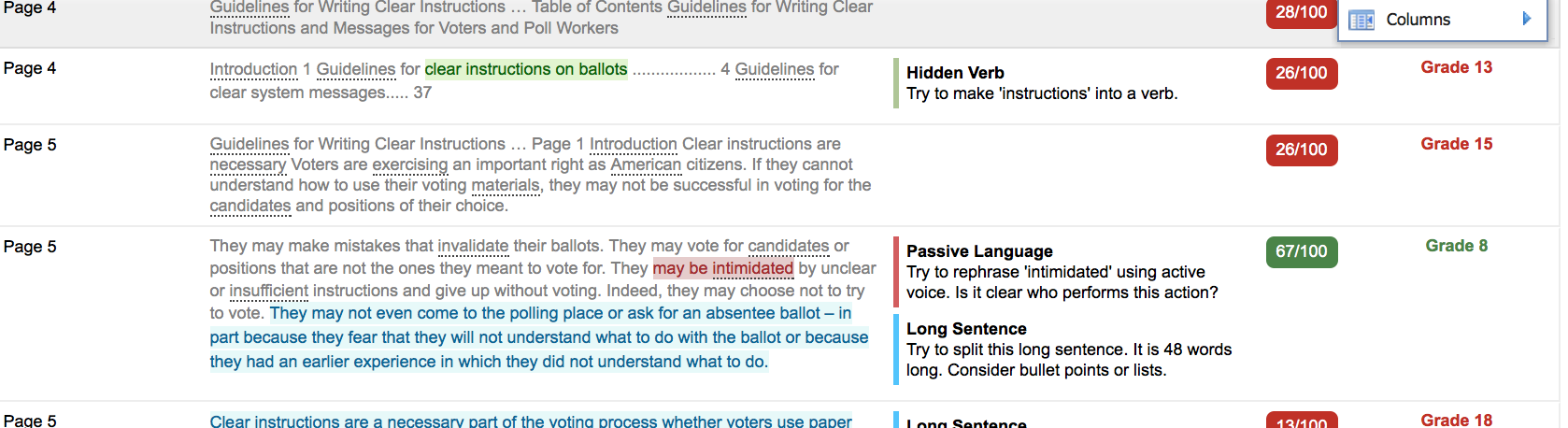
### Visible Thread

[VisibleThread](http://www.visiblethread.com/solutions/government-communications/) is a web-based readability tool for PCs or Macs. It gives a detailed analysis of the document and suggests how to improve it. Its criteria can be adjusted for things you decide you don’t need pointed out. The initial report only shows the sentences that have problems and tells you what page they’re on, rather than showing the entire document. The higher the readability score, the easier it is to read the document.

**Cost** – Website shows “Readability content creators – free to analyze any text; $45 monthly for premium readability content.”

**Sample of the Visible Thread results**

The report shows: Location - Document content – Suggestions – Readability - Reading level

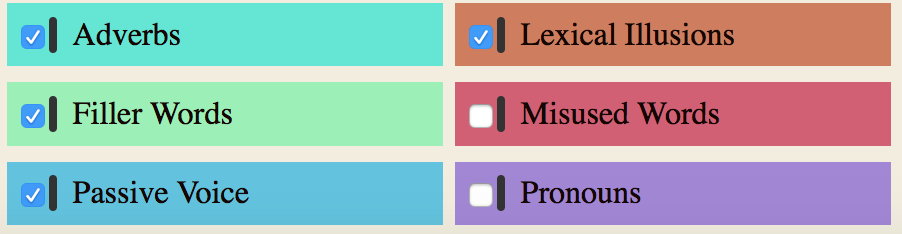


### Editsaurus

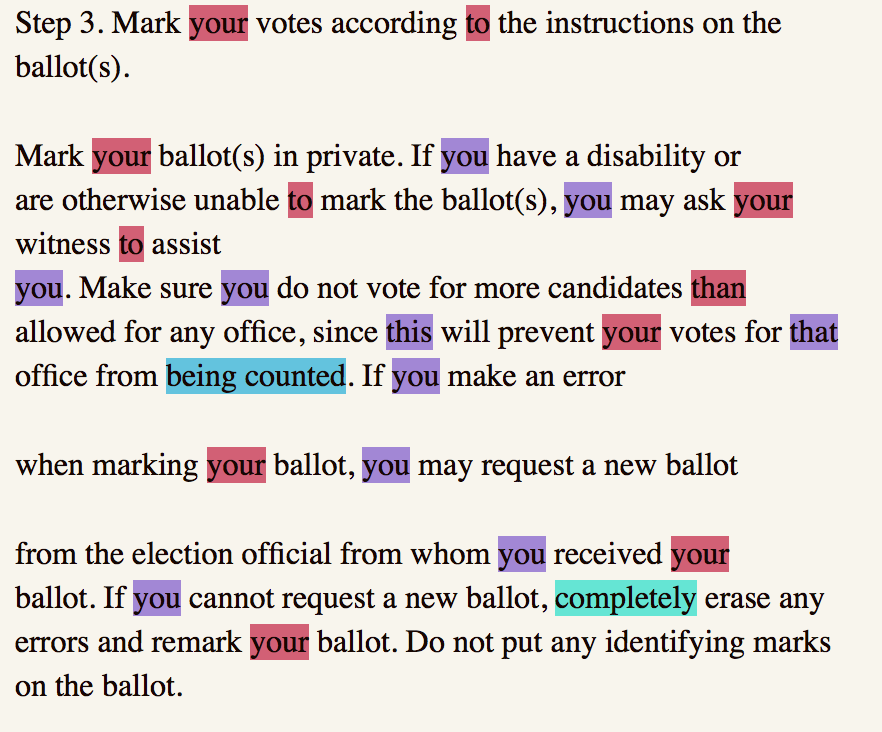
[Editsaurus](https://editsaurus.tylerwalters.com/) is a tool that lets you check the features you want to review among adverbs, filler words, passive, lexical illusions, misused words and pronouns. It shows your text with the types of problems highlighted in the corresponding color. It shows the original document side-by-side with the same document highlighted for possible problems.

**Cost** - Free

**Sample of the Editsaurus filters**



**Sample of the Editsaurus results**

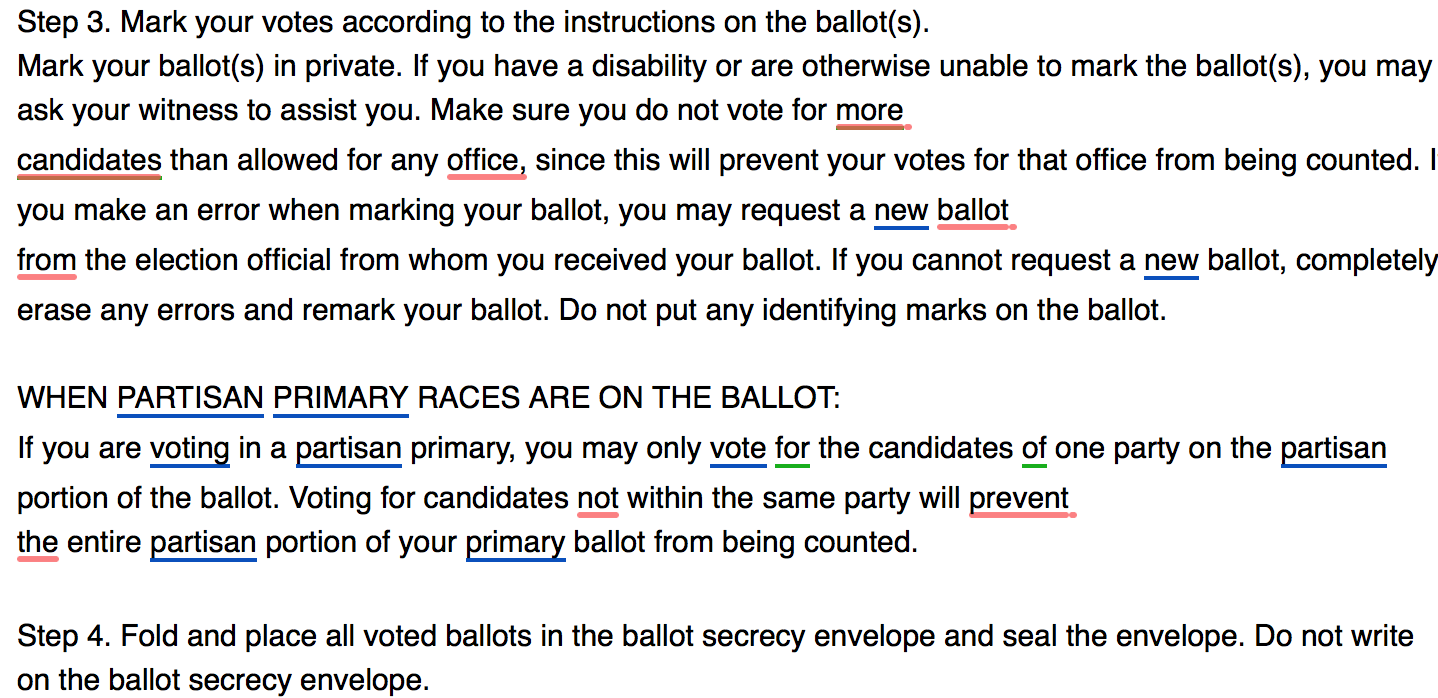


### White Smoke

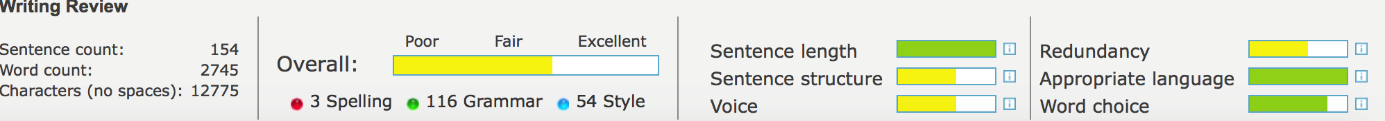
[White Smoke](http://www.whitesmoke.com/) is a web-based program that is simple to download. It underlines words in various colors, and indicates the problem when you scroll over the word.

**Cost** - $80 or $120 per year

**Sample of the White Smoke markup**



**Sample of the White Smoke summary score**

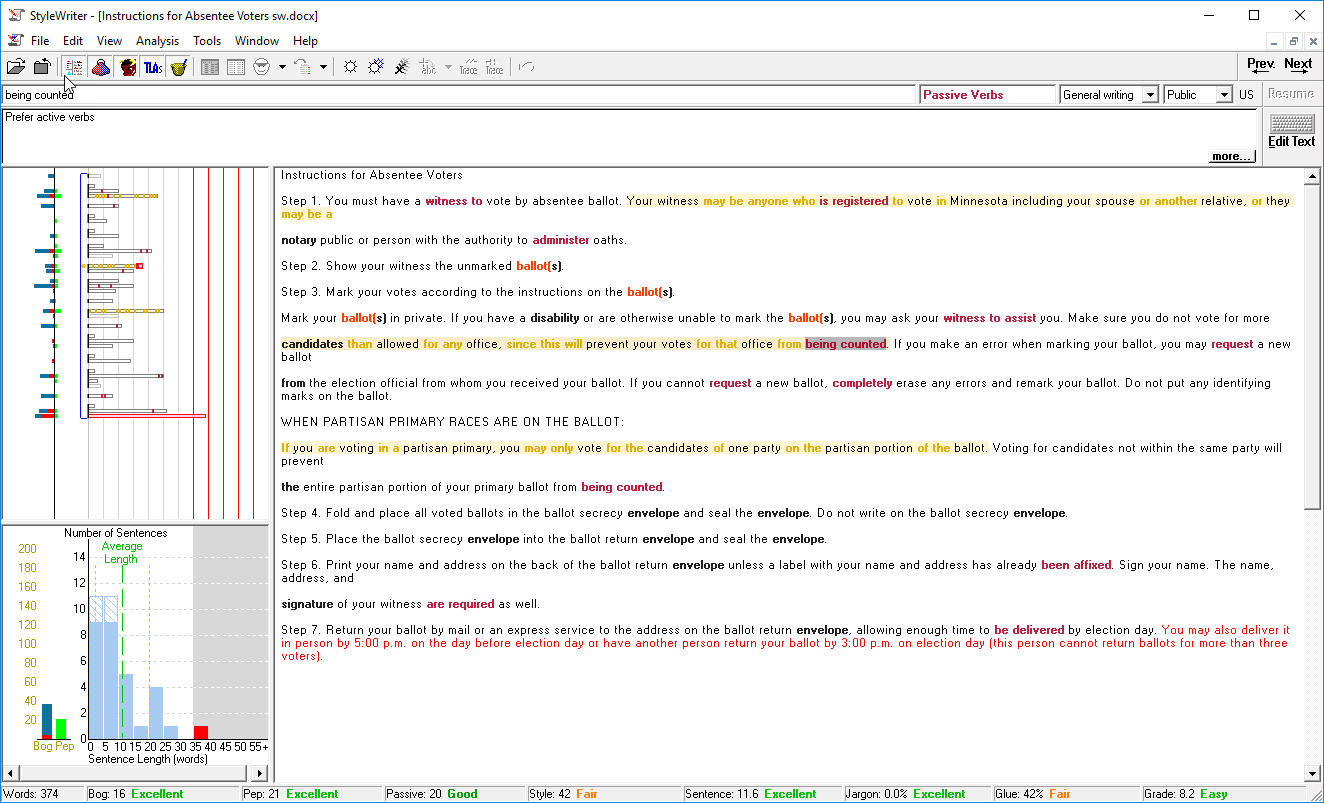


### Style Writer

[StyleWriter](http://www.stylewriter-usa.com/) is one of the first plain language programs available, and many Federal agencies have used this. It has a unique scoring system that provides encouraging feedback to writers. It might be especially useful for a documentation team. It was designed for PCs and only runs on a Mac in a virtual Windows environment.

**Cost -** 3 versions: $90, $150, $190

**Sample of Style Writer’s markup**



## Additional resources

 NISTIR 7596 - Guidelines for Writing Clear Instructions and Messages for Voters and Poll Workers. Authors: Redish and Laskowski, May 2009

NISTIR 7556 Report of Findings: Use of Language in Ballot Instructions. Authors: Redish, Chisnell, Newby, Laskowski, and Lowry, December 2008

NISTIR 7519 - Style Guide for Voting System Documentation. Authors: Chisnell, Becker, Laskowski, Lowry, August 2008

[Federal Plain Language guidelines](https://www.plainlanguage.gov/guidelines/) - <https://www.plainlanguage.gov>