

A Reflective Memo on Instructions

A Technical Writing Assignment

Connor Taffe

October 5, 2015

This assignment was illustrative of the orderly, though out nature of instructional documents. The knowledge gained can be most easily split into two categories: a description of the knowledge itself, and the future uses and implications of said knowledge.

0.1 Knowledge

This project provided a good proving ground for the practices we have learned so far in technical writing. Easy to read, concise instructions performed better in usability tests, therefore, almost as if by natural selection, improving instructions and showing which tactics were most effective.

0.2 Future implications

In computer science, especially the portion that is writing programs, there is a small portion of time dedicated to an extremely important task: documentation. Documentation can take several forms: comments in source code, UNIX manual pages, or “README” documents in source repositories.

- *Source code comments give meaning to code.* Source code can be difficult to decipher, especially in large collaborative projects that require a bit of background reading. In these types of environments, it is important that a developer can come upon a piece of code he has never touched and know what it does almost instantaneously. This is the purpose of source code comments. Source code comments outline the purpose of code. They must be concise, simple, short, and thusly easy to read.
- *Manual pages give descriptions of programs.* Programs in UNIX/Linux do not inherently have any user interface besides a blinking cursor on a black screen (nongraphical, I mean). Thusly, it is important that there be manual pages a user can read to learn the options and usage of a program. Writing manual pages requires concise writing and simple explanation.

Like instructions, they must convey the operations a user must perform in order for some goal to be achieved.

- *“README” documents give directions to developers.* “README” documents are historically meant to be read before mucking around in a source directory. They give instructions on compiling code, making changes and sometimes contributing back those changes. Developers must be attentive to these files as they will instruct all those who visit this folder afterwards.

In conclusion, it is easy to see that writing instructional material for others is a useful and valued skill in Computer Science. It is a most important part of conveying knowledge from one party to another. In an efficient company, no time can be wasted relearning or debugging what should have already been documented.