

User Testing Report

To: Ana Leighe Cooke

From: Team Git

May 5, 2017

Professor Cooke,

The purpose of this memo is to cover the objectives, users, tasks, and results of our user testing.

Objectives

Our intended goal is for our manual to be primarily used as an introduction the basics of version control with Git for beginners. Our secondary goal is for our manual to serve as a quick reference for Git commands for intermediate to advanced users.

Our user tests were split between two groups: beginner and intermediate users of Git. Our objective with the beginner group was to test the immediate use of our manual for someone with little to no knowledge of Git. Our objective for the intermediate group was to test how useful our manual would serve as a quick reference.

Test Subjects and Testing Procedure

User Testing Overview

We targeted individuals that were either new to or experienced with Git. The reason for this is that we envision our manual primary usage to be an introduction to the basics of Git for beginners as well as a quick reference for Git commands for intermediate to advanced users.

The reason we chose the coaching method for beginners is to better simulate a first-time experience with using Git. Beginners will be handed a computer with a directory with several files and a list of tasks to accomplish. The manual will be available as their primary source of help.

The think-aloud protocol is better for getting feedback from more experienced users since they can read the instructions and simultaneously compare it to their previous experience using and learning Git.

Intended user testers

User	Level	Method	Contributor
Tim Becker	intermediate	coaching	Evan
Cole Heathershaw	intermediate	think aloud	Fred
Riley Xu	intermediate	think aloud	Sunny
Xinni Wu	beginner	coaching	Ziyang
Di Wong	intermediate	think aloud	Ziyang
George Situ	beginner	coaching	Nishad

We plan on testing the whole draft on our users, since it is attended to be modularized for easier skimming with users being able to pick out sections that they need at the time.

Individual User Feedback

Tim Becker

Cole Heathershaw

Overall, Cole was off-put by the overall tone of the manual. In particular, he found some sections were “too-personal.” Part of the inconsistency across sections in terms of tone is due to the way we allocated sections to individual group members. Furthermore, Cole disagreed with the inclusion of the “Why use Git?” section, which he considered unnecessary considering that most people opening a manual know why they are using the manual.

A larger portion of his feedback stemmed from his differentiation between guides and manuals, with manuals addressing more expert users and guides being closer to an introduction. In this regard, our manual was more of a guide since our target audience consists of beginner and intermediate users of Git.

For some other feedback points, Cole mentioned the following:

- Get rid of the use of “we”

- Add warnings to some sections such as “Deleting branches”
- Fix dangling sections
- Be more direct instead of suggesting in some sections (i.e. removing “perhaps,” “want,”)
- In “Merging branches,” what is the word “This” referring to?
- The Problem/Solution/Description separation works sometimes, but in some sections the problem statement is so short to the point that it is redundant with the section header.

Riley Xu

We used the think-aloud protocol with Riley. Since he is an experienced Git user, we had him read through the manual and voice his opinions on the various sections. His feedback was more focused on the content of the manual, but he also gave information on the organization of the manual. He was reasonably impressed with the manual and liked the visuals.

Riley mentioned the following:

- Explain the difference between rebasing and merging clearly.
- Emphasize that while switching branches the changes made to current branch are reverted.
- Specify when to delete branches and when to merge them.
- Fix grammatical errors.
- Include a cheatsheet at the end i.e. a list of common git commands and references to them in the manual.
- More detail needed in git add (e.g. git add ., git add -A).
- Did not introduce Git commit hashes, therefore the section on reverting changes and git history can be confusing to readers.
- In the images in installation section, zoom in on the relevant section.
- It was unclear how to actually make a Git alias.

Xinni Wu

Di Wong

George Situ

1 Revisions