

1. Complete these user stories:

- As a vanilla git power-user that has never seen GiggleGit before, I want to learn to use GiggleGit in a way that translates well from vanilla git
- As a team lead onboarding an experienced GiggleGit user, I want the experienced user on my team to have an accelerated onboarding experience to save their time
- As a rookie developer new to version control, I want GiggleGit to also teach me general git practices so I'm not confused later on.

Task: Create a quick-start guide for first-time git users.

Ticket 1: Create outline for general git guide

Description: Collate the essential information to setting up and using a version control system in an itemized manner.

Ticket 2: Create Figma mockup for quick-start guide

Description: Using Figma, create the mockup for how the quick start guide will appear in the GiggleGit onboarding process

- “As a user I want to be able to authenticate on a new machine”
It says nothing of the users motivations or specific intentions, so it's not a user story. It's almost closer to a technical requirement or theme.

CodeChuckle is introducing a new diff tool: SnickerSync—why merge in silence when you can sync with a snicker? The PMs have a solid understanding of what it means to "sync with a snicker" and now they want to run some user studies. Your team has already created a vanilla interface capable of syncing with the base GiggleGit packages.

Goal: Ensure that the set of snicker sounds are enjoyable for user's to hear

Non-goal: Get data for which snicker sounds most appeal to the study participants

Non-functional requirement: Data collation

Functional Requirement: Store user ratings of different sound profiles in a csv file

Functional Requirement: Save the study csv file on the cloud using dropbox

Non-functional requirement: Study Integrity

Functional Requirement: Study participants should be selected who have no ties to company or competitors

Functional Requirement: Initial snicker sounds should be randomized per user to avoid recency bias