Agile:

As a vanilla git power-user that has never seen GiggleGit before, I want to be able to match my previous git experiences to functionalities in GiggleGit so that I can learn to use it easier.

As a team lead onboarding an experienced GiggleGit user, I want to conveniently construct and manage a team on GiggleGit, so that I can limit the amount of time wasted on team organization

1. User story: As a first-time git user who has never tried git or GiggleGit before, I want to have a clear stepwise tutorial and assistant to guide me, so that I can understand the functions of git faster.

* Task: Set up first-time user tutorial and assistance
  + Ticket 1: Implement a first-time user tutorial
    - Add a Boolean field to the user database to identify first-time users which prompts the tutorial automatically the first time and points out where to access the tutorial.
  + Ticket 2: Implement an interactive chatbot for questions
    - Implement a chatbot to capture keywords from user questions and list possible directions that may answer the question. Or import external AI API with memory set to our content.

This is not a complete user story because it lacks the benefits part of a user story. It only has a specific actor part and an action part.

Formal Requirements:

Goal: Have snickersync integrated with the original basic Giggle Git functionalities and allow users to change between the traditional merge style and the new snickersync merge style for better feedback.

Non-Goal: Replace the whole Giggle Git original functionalities with the new snickersync.

1. And 3.

Non-functional requirement 1: Security

* Functional requirements:
  + The GiggleGit administrators should have authenticated access to the code of snickersync for any changes and maintenance.
  + GitggleGit users should only have authenticated access to regular merge and compare functionalities of the snickersync.

Non-functional requirement 2: Relability

* Functional requirements:
  + The snickersync must have a random log for different control and variant groups of users to ensure the randomness of the user study.
  + The system must also have a database to store the log for user study analysis.