

Lina Park  
CASCS 411  
Professor Golbus  
10 February 2025

## **411: HW1 Requirements**

### **Agile:**

- Theme: Get GiggieGit demo into a stable enough alpha to start onboarding some adventurous clients
- Epic: Onboarding experience
- User story 1: As a vanilla git power-user that has never seen Giggie before, I want to be able to learn GiggieGit commands with ease,
  - Task: Make resources/instructions with in depth resources/instructions
    - Ticket 1: Step by Step Instruction List
      - Create a good Step by Step Instruction List for new Users to connect their github account to GiggieGit. These instructions should guide the user through basic uses such as repository selection, meme selection, or merging.
    - Ticket 2: Have Git Integration Document
      - Write a document with comprehensive explanations on how to use Giggie Git. Make sure to use clear and simple connotations.
- User story 2: As a team lead onboarding experience GiggieGit user, I want to make sure that my team can easily use Giggie Git and continue using advanced features. (I don't want to loose fancy features nor have new problems)
  - Task: Make sure that GiggieGit has the advanced features without new problems
    - Ticket 1: Team Collaboration
      - Simple and clean setups/interfaces are crucial for team collaboration. With work being done throughout a team, Giggiegit needs to allow for easy management.
    - Ticket 2: Advanced Features

- When new team members join or initial team members need to do something they haven't done before, there needs to be a place for good instructions to securely and efficiently use GiggleGit Tools. This can be done through advanced features such as branching, forking, merging, different memes so that members can collaborate while maintaining order.
- User story 3: As a user who has been using Github, I want to be able to easily transfer repositories to Giggle Git.
  - Task: Set up Github Connection/integration
    - Ticket 1: Sync Github to Giggle Git
      - Create a way for users to sync their Github to GiggleGit. Allowing users to transfer or simply sync their current Github Account with their GiggleGit account and vice versa.
    - Ticket 2: Ease of Connection.
      - Implement OAuth login for Github so that users can easily connect their Github account to their GiggleGit account.
- Not User Story: As a user I want to be able to authenticate on a new machine.
  - For a user story to be a user story, there needs to be previous context. In this phrase, there is a want but no why. Also, it is lacking in detail, and therefore cannot be interpreted as a clear set of instruction or requirements leading to confusion.

### **Formal Requirements:**

- Goal: Create an engaging and entertaining experience while using SnickerSync when making a merge.
- Non-Goal: Although the entertaining experience comes with SinckerSync, prioritize the user study on the engagement of users produced from the vanilla interface.
  - Non-functional requirement 1: Random/Non-biased
    - Functional requirements:
      - Use a control group and a variant group to measure the engagements of users to make sure that no one is biased.

- Make sure that users are assigned randomly to make sure that there is no relationship between the user and whether they are put in the control group or variant group.
- Non-functional requirement 2: Repeatability
  - Functional requirements:
    - Previous user assignments and their results will be stored locally
    - New or Returning Users from Github/Giggle Git should be able to easily use SnickerSync