Agile

* + As a vanilla git power-user that has never seen GiggleGit before, I want to have as much control as vanilla git, so that I can manage my code effectively
  + As a team lead onboarding an experienced GiggleGit user, I want to understand how experienced GiggleGit users use GiggleGit, so that I can improve upon the application by setting goals.

1. A student new to git
   * As a student new to git, I want to be able to easily pick up and use GiggleGit, so that I can create and manage my code.
   * **Task:** Pick up and be introduced to GiggleGit
   * **Ticket 1:** Intuitive interface
     + Create an intuitive interface to help the user navigate and learn how to use GiggleGit effectively. Ensure that the interface is streamlined and not overtly cluttered.
   * **Ticket 2:** Documentation
     + Create a documentation for the different functions useable in GiggleGit. Ensure that the description for each function is clear and concise.
2. This is a task. It isn’t a user story as it doesn't have a user type, goal, or reason.

Formal Requirements

1. **Goal and Non-Goal**
   * **Goal:** Add a feedback button that links to a basic questionnaire/ response form with required questions on the design interface
   * **Non-goal:** SnickerSync does not have to support version control systems other than GiggleGit during its initial release
2. **Non-functional Requirements**
   * **Requirement 1:** SnickerSync must be able to run on multiple operating systems, such as MacOS, windows 11 and Linux
   * **Requirement 2:** Random assignments of users to snickering concepts in user studies
3. **Functional Requirements**
   * **1.1:** All core functionalities must operate consistently on the different operating systems
   * **1.2:** The installation package must include an interface to allow for seamless and easy installation
   * **2.1:** A randomization algorithm should be used to randomly assign users to different snickering concepts
   * **2.2:** A Google form or an alternative method linked from the feedback button should be used to audit and collect the recorded feedback for easily visualization and analysis

Part 3: Dependencies

