Complete the following tasks:

1. **Complete these user stories:**
    - ○ **As a vanilla git power-user that has never seen GiggleGit before, I want to…**
    - ○ **As a team lead onboarding an experienced GiggleGit user, I want to…**

Theme: Get GiggleGit 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 GiggleGit before, I want to understand how it works and ensure how accurate it is compared to github.

Task: Test out GiggleGit demo and compare features with those of GitHub.

- ■ Ticket 1: Create repositories, branches, and other essential github features.
    - ○ Need to make sure that the big important aspects of Github are retained in GiggleGit. Also have to ensure that the feature that makes GiggleGit different (the memes), is working properly.
- ■ Ticket 2: Document the user's experience navigating through GiggleGit
    - ○ See if there are any major challenges that may come up as the user is using GiggleGit and record to improve it further.

User story 2: As a team lead onboarding an experienced GiggleGit user, I want to hear about their experience using GiggleGit and also walk them through some future ideas that might be implemented.

Task: Meet with the experienced GiggleGit user.

- ■ Ticket 1: Conduct a presentation/meeting to gather feedback
    - ○ Hear about what is working and what isn't. See if there are any good features that make it stand out from GitHub or if it is still not as good, and people would rather use the original.
- ■ Ticket 2: Test out new advancement features of GiggleGit
    - ○ Get a sense of whether the new features are needed or won't be useful to the users. If not useful then it is probably not worth implementing at the moment.

2. **Create a third user story, one task for this user story, and two associated tickets.**
    - ○ **Tasks should be a single phrase. (As should themes and epics. See those provided.)**
    - ○ **User stories should be one to three sentences.**
    - ○ **Tickets should have a title consisting of a single phrase and details that are long enough to sufficiently describe what needs to be done. You do not need to assign points to the tickets**

User story 3: As a user who plans on using GiggleGit as the main version control system, I want to feel comfortable saving my projects and working on a different site.

Task: Securely log and save all projects pushed

- ■ Ticket 1: Implement User Authentication
- ■ Ticket 2: Implement Recovery and Back Up Methods

3. **This is not a user story. Why not? What is it?**
    - **As a user I want to be able to authenticate on a new machine**

This is not a user story because it does not necessarily state what the user needs it for. This can be used as a ticket to help out a different user story.

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.

Complete the following tasks:

1. **List one goal and one non-goal**

Goal: Seamlessly sync SnickerSync with GiggleGit.
Non-goal: Implement advanced features that will make the platform annoying to use.

2. **Create two non-functional requirements. Here are suggestions of things to think about:**
    - **Who has access to what**
    - **PMs need to be able to maintain the different snickering concepts**
    - **A user study needs to have random assignments of users between control groups and variants**

1) Ensure that the system is secure from unauthorized users and the syncing works well.
2) Be able to adapt to different formats of projects/be compatible with other platforms

3. **For each non-functional requirement, create two functional requirements (for a grand total of four functional requirements).**

1) The system must be secure
2) The system must be able to maintain various formats