**Agile:**

**1.**  
As a vanilla Git power-user who has never seen GiggleGit before, I want to be able to easily disable memes so that I can use vanilla Git if needed.  
As a team lead onboarding an experienced GiggleGit user, I want to have better memes so I can laugh multiple times during merges.

**2.**  
As a returning user who has used GiggleGit multiple times, I want to be able to save my favorite memes so I can see them more often when merging.

* **Task:** Make memes savable.
  + **Ticket 1:** Design and create a database large enough to store the memes.  
    Figure out how many memes can be favorited (infinite or finite). This will help us determine the type of database to use.
  + **Ticket 2:** Implement a heart button on memes.  
    Create a flag on the meme object that indicates if it is favorited. Add it to the database when the heart button is clicked on the meme UI.

**3.**  
The reason it is not a user story is because we need to identify the type of user telling the story and the reason behind their need.

**Project Requirements:**

Goal: Provide an easy-to-use interface for SnickerSync so that users may engage in user research and see the merging of modifications using the "sync with a snicker" idea.

Non-Goal: creating a completely new backend for merge conflict

Non-functional requirement 1: Access Control & Permissions

Functional requirement: Create a dedicated admin interface that allows PMs to add, modify, and delete snickering concepts without impacting the core functionality of GiggleGit.

Functional requirement: Implement role-based access so that only PMs and authorized personnel have write permissions to edit SnickerSync features, while regular users have read-only access to the snickering concepts.

Non-functional requirement 2: User Study Randomization & Data Integrity

Functional requirement: Implement a random assignment module that distributes users evenly between control and experimental groups based on predetermined probabilities.

Functional requirements: Create a logging system that tracks user assignments and flags any anomalies (e.g., a user appearing in both control and experimental groups), ensuring data integrity throughout the study.