

Agile

- Complete these user stories:
 - As a vanilla git power-user that has never seen GiggleGit before, I want the documentation to be readily available to transition to GiggleGit faster.
 - As a team lead onboarding an experienced GiggleGit user, I want to be able to easily give them permission to work on repositories, allowing them to quickly be productive.
- Third user story, task, and two tickets:
 - User story: As a new user only familiar with git, I want GiggleGit to correct me when I accidentally use vanilla git commands, leading me towards the appropriate GiggleGit command, allowing me to learn GiggleGit faster and stick with the service.
 - Task: Create GiggleGit correction mechanism.
 - Ticket 1: Map most git commands to GiggleGit commands – Create a program that maps git commands to the GiggleGit alternatives will be created, providing feedback to the user on what GiggleGit command they should use.
 - Ticket 2: Implement mapping into GiggleGit interpreter – Add a feature to the GiggleGit CLI that whenever there is an error, it will check the git to GiggleGit mapping and if the user put in a git command in the mapping, it will output what command they need.
- Why is the following not a user story? – As a user I want to be able to authenticate on a new machine

- That is not a user story because user stories must have a benefit, and in this case the user story does not document a benefit of wanting to be able to authenticate on a new machine. This merely describes a feature.

Formal Requirements

- List one goal and non goal
 - Goal: Create the ability for users to create their own profiles to test out the system.
 - Non goal: Have users connect with other users, implementing a social network like system.
- Two non functional requirements
 - 1: Security
 - 2: Performance
- Two functional requirements for each non functional requirement
 - NFR 1:
 - Hash passwords before they are stored in the database.
 - Implement https communication between front and back end.
 - NFR 2:
 - Create indexes on the users table in the relational database to speed up logging in.
 - Install a redis cache to cache frequent data.