

Project requirements

- Goal: To create a easy-to-use version control system with jokes/memes in merges that users find funny
- Non-Goal: Automatically generate memes based off the code being merged
- Non-functional requirement 1: Security
 - Functional requirements:
 - Use SSH keys to authenticate user
 - Securely store confidential information with private repos and ensure data is only accessible to the specific group of users
 - Non-functional requirement 2: Usability
 - Functional requirements:
 - A dashboard where users can create, edit and remove snickering concepts
 - Have an optional tutorial process providing step-by-step guide showing the essential tools of the system

Agile

- 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 ensure users feel comfortable using GiggleGit and decide if they want to have memes
- User story 2: As a team lead onboarding an experienced GiggleGit user, I want to manage access to repositories and snickering concepts
- User story 3: As a completely new GiggleGit user, I want to familiarize myself with meme-based merging with a testing platform
 - Task: Make a sandbox environment
 - Ticket 1: Create an separate sandbox environment where users can test meme-based merging without impacting their projects
 - Ticket 2: Build basic tutorial with instructions and visual cues to help users familiarize themselves with meme-based merging.

As a user I want to be able to authenticate on a new machine. It is not a user story because it only provides a feature and not the user's motivation or the value of the user's gain. It should tell why the user needs this functionality.