

## **Agile**

### **User Story 1**

As a vanilla git power-user that has never seen GiggleGit before, I want to understand the differences between traditional Git and GiggleGit so that I can smoothly transition.

### **User Story 2**

As a team lead onboarding an experienced GiggleGit user, I want to provide detailed onboarding resources so that the team member can add GiggleGit to our workflow.

### **User Story 3**

As a developer setting up GiggleGit for the first time, I want to ensure that the installation process is correct and smooth so that I do not run into issues later.

### **Task**

Installation and Setup Documentation

### **Ticket 1**

Write step-by-step guides for installing GiggleGit on all operating systems and include solutions to common issues.

### **Ticket 2**

Create YouTube videos to walkthrough the installation process.

### **Statement Evaluation**

“As a user I want to be able to authenticate on a new machine” is not a user story because it does not follow the form of a user story, which includes an action that leads to some benefit. The given statement has an action but it does not state the benefit or reason.

## **Formal Requirements**

### **Goal**

Provide a user-friendly experience using SnickerSync to make it easy for users to understand the tool.

### **Non-Goal**

Redesign the user interface of GiggleGit.

### **Non-Functional Requirement 1**

Usability

### **Functional Requirement 1**

Create an interactive tutorial for when users first use the tool.

**Functional Requirement 2**

Create a help button that answers frequently asked questions and can provide guidance of specific issues.

**Non-Functional Requirement 1**

Security

**Functional Requirement 1**

Create a secure login system so that only authorized users can access sensitive information.

**Functional Requirement 2**

Create a two-factor authentication system so that there is added security to users' accounts and information.