

## 1. Agile Slide:

### a. Creating valid user stories:

- i. As a vanilla git power-user that has never seen GiggleGit before, I know more about GiggleGit and what it does (a quick guide or demo of the product).
- ii. As a team lead onboarding an experienced GiggleGit user, I want the user to be able to give feedback or suggestions so we can improve on our product.

### b. As a GiggleGit user, I want a clear and helpful set of instructions on how to use this product and access all of its features.

- i. Task: Implement a quick walkthrough/ PDF instructions the users
- ii. Ticket1: Add an easy to access link to instructions for new users
- iii. Ticket2: Create a downloadable walkthrough/ PDF instructions for the user to save on their devices

### c. This is not an user story because it seems to be incomplete and not descriptive enough.

## 2. Formal Requirement Slide:

### a. The goal is to evaluate the users response and feedback to syncing with snickers. A non goal is making sure the tool works as efficiently as possible.

### b. One non-functional requirement is making sure the tool is up and running as fast as it can all the time and never crashes. Another non-functional requirement is the tool should have special roles or permissions for the PM so they access and change the options through this tool (other regular users would not be able to do this).

### c. For the first non-functional requirement, a functional requirement is a monitor that tracks the system and checks for any potential error that may occur and another functional requirement is handling the error/crash if it does happen (this can be providing the user with an error message and getting the tool back and running as soon as possible to satisfy the user). For the second non-functional requirement, a functional requirement would be to properly assign the right user with the right role so no one is giving permission and access to features they wouldn't be using. Another functional requirement for this is making sure people without the right role don't have permission to do certain actions (basically adding security to certain actions and abilities).

## 3. Partitioning Functionality: On GitHub

## 4. Importable Module:

```
from wildlife_tracker.animal_management.animal import Animal
from wildlife_tracker.habitat_management.habitat import Habitat
from wildlife_tracker.migration_tracking.migration import Migration
from wildlife_tracker.migration_tracking.migration_path import MigrationPath
from wildlife_tracker.formal_specification import Formal_specification
```

5. Dependencies:

