

# Team Reflection: Group 23 Sprint 3

Our team has been doing great this week. Throughout the week we have encountered several problems slowing down our progress with the conversion from command line to GUI, but fortunately we have all been able to solve these together. We have improved our communication within the team by having daily scrums so that our progress and eventual problems could be more clearly tracked and handled accordingly. We can improve our agile teamwork going forward by circulating the people we work with, to improve the overall cohesion within the group.

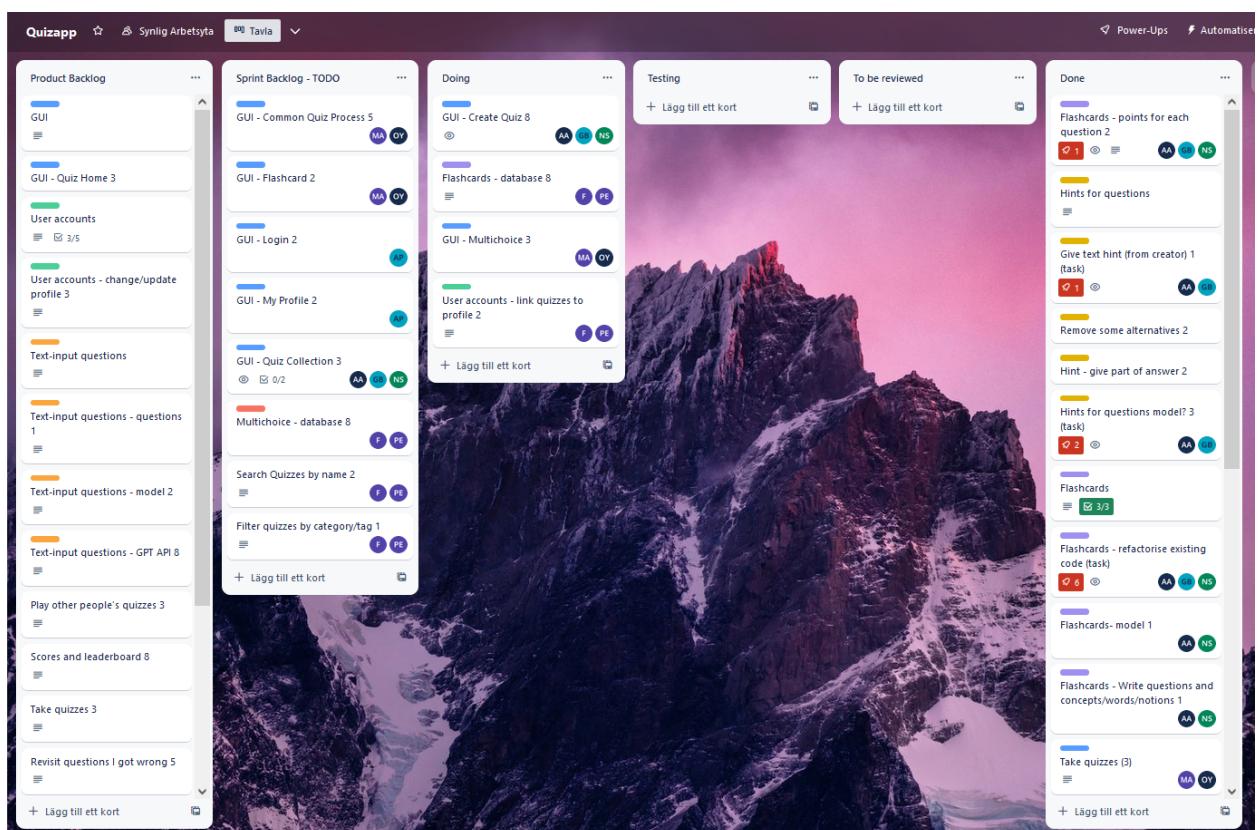
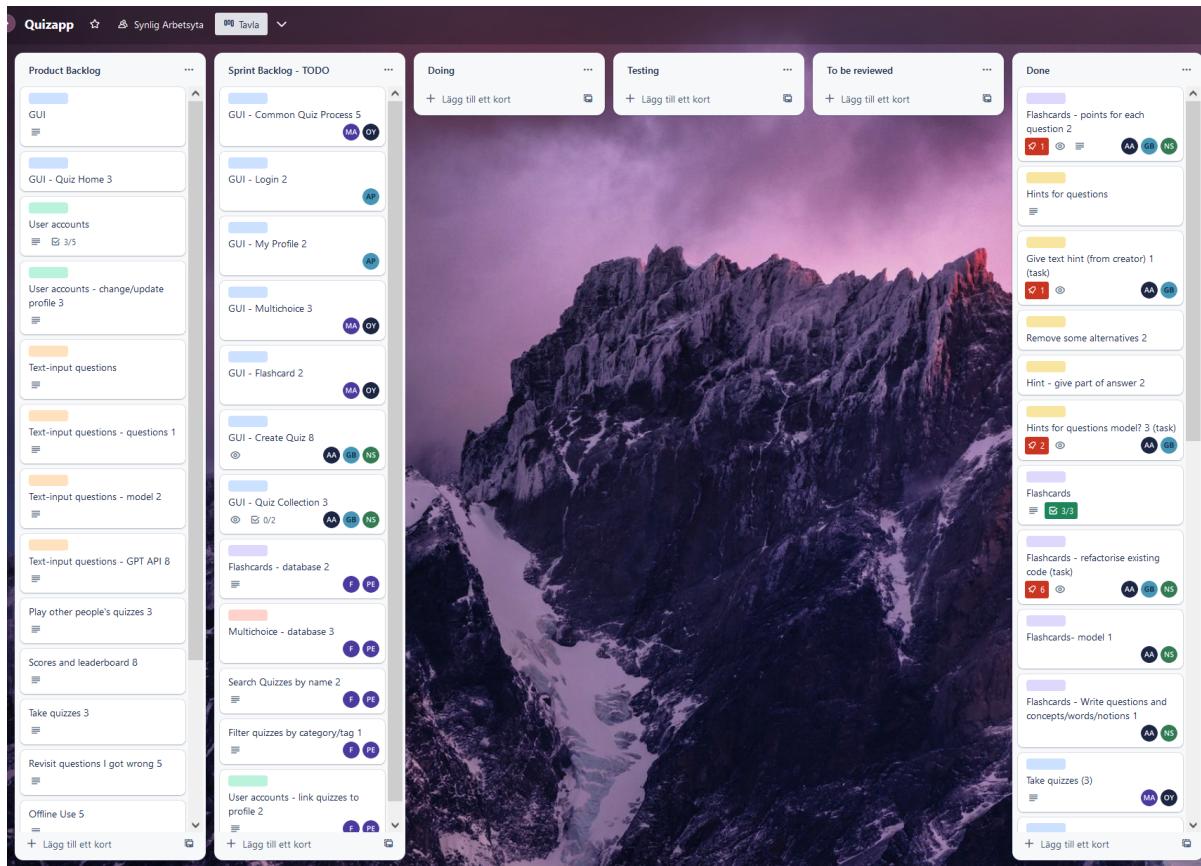
This sprint our team implemented quite a few user stories. We increased the number of story points taken during our sprint planning this week in order to evaluate how this would affect the outcome of the sprint. This would give us a clearer measurement of how many points we can undertake in one week to further enhance our future sprint planning. The user stories we implemented this week were, for example:

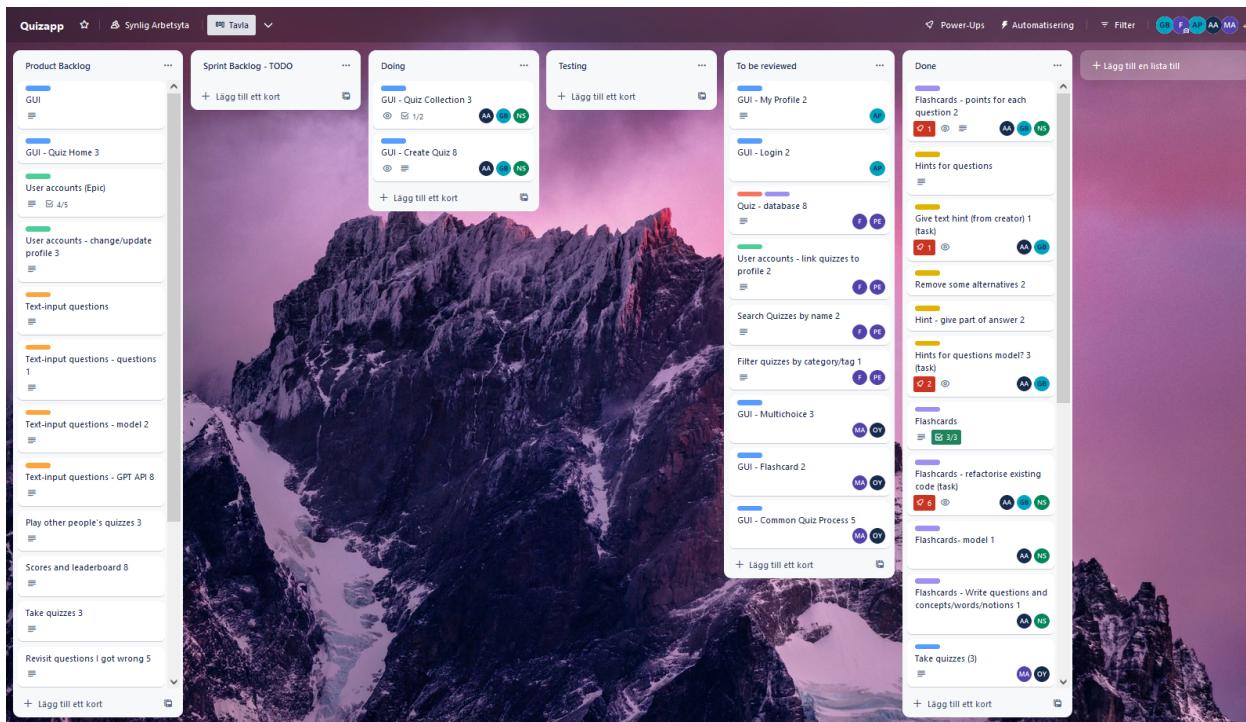
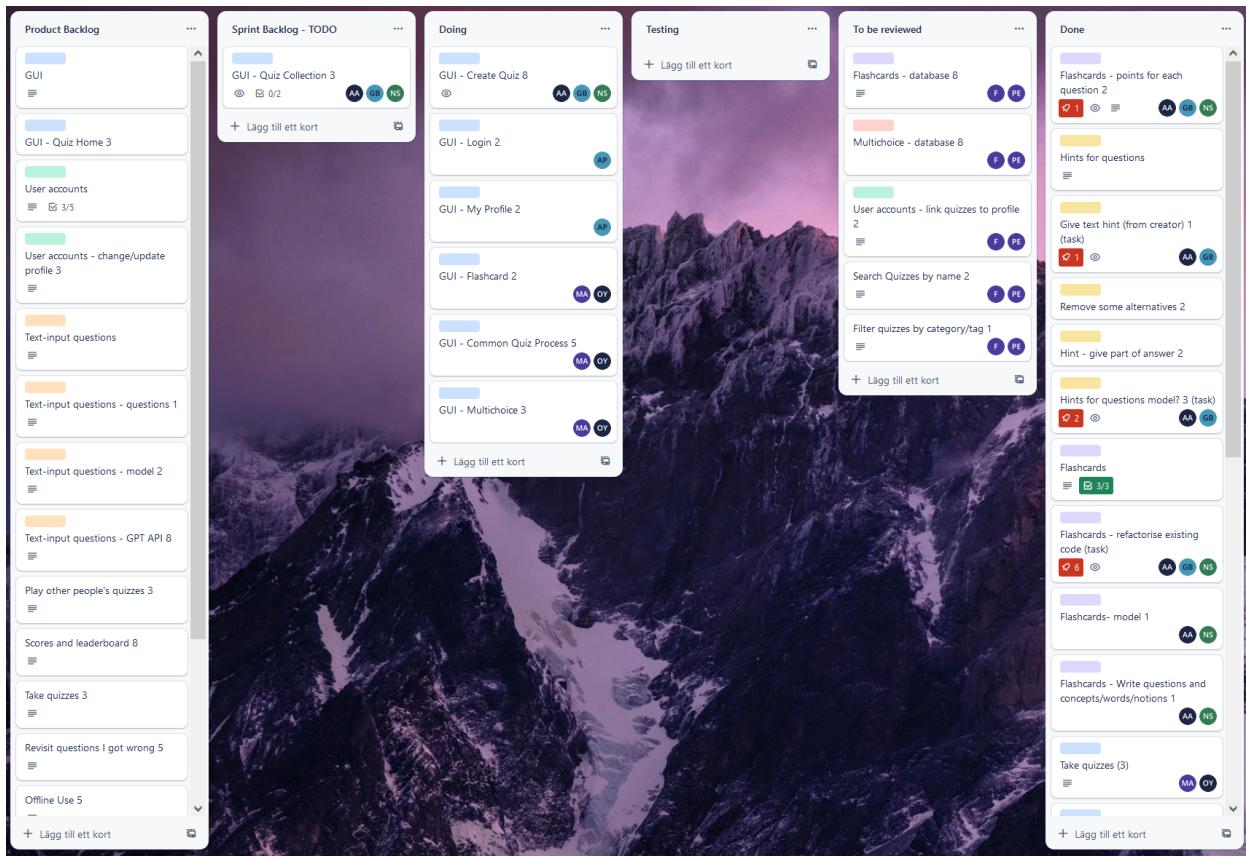
- As a student that uses the application often. I want my quizzes to be saved, so that I won't have to create them all over again each time. (Quiz -database 8p)
- As a creative individual I want an interactive way to create my own quizzes so that they fit my preferences (GUI - create Quiz 8p)
- As a busy student who values time, I want to be able to search for specific quizzes, so that I can save time (Search quizzes by name 2p)

During our first sprint not only did we assign points to user stories but we also decided to assign them to tasks, because the team felt like it would make our efforts more effective, knowing approximately how much the points of the task needed to be done. We feel like this has had a positive effect on the team and therefore decided to stick with this approach.

Since our DOD included testing the code with JUnit tests, which is not applicable to GUI related user stories, we needed to add some acceptance criterias specifically for the GUI tasks and user stories to still ensure the quality of the application. For example, another person has to do a GUI-review to ensure that it is intuitive and working great.

Next sprint we will focus on improving the connection between the different GUI elements worked on by the members of our team since we took on a little too much work upon us this sprint. The tasks and user stories we tried to implement this week were all finished, however the final merge between all of the branches brought up some unforeseen GUI difficulties as some of those tasks and user stories depended on each other. The merging of user stories was pushed to next week as a result of this.





My Quiz

Add a new Flashcard

<p><b>TERM</b></p> <p>What color is the ocean</p>	<p><b>DEFINITION</b></p> <p>Blue</p> <p>X</p> <p>CHOOSE HINT</p>
<p><b>TERM</b></p> <p>What is the capital of Norway?</p>	<p><b>DEFINITION</b></p> <p>Oslo</p> <p>X</p> <p>CHOOSE HINT</p>
<p><b>TERM</b></p> <p>What is a flashcard?</p>	<p><b>DEFINITION</b></p> <p>What is a flashcard?</p>

[BACK](#)

[NEXT](#)



# TAGS

**Applied tags**

Medicine	Architecture	Art	Music
Theater	Geology	Nutrition	Cybersecurity
Biomechanics	Pathology	Semiotics	Virology
Forensic	Pharmacology	Anatomy	Ophthalmology
Mycology	Entomology	Biochemistry	Immunology
Horticulture	Robotics	Dentistry	Epidemiology

Accounting  
Psychology  
History

## How the database looks like (using firebase):

The screenshot shows the Firebase Realtime Database interface. The path is `users > OQUyXsAvbYIWryZZgAqb`. The left sidebar shows the `quizapp-a1bfd` project with a `quizzes` collection and a `users` collection. The `users` collection has several documents, one of which is selected and expanded: `OQUyXsAvbYIWryZZgAqb`. This document contains fields: `email: "test"`, `id: "OQUyXsAvbYIWryZZgAqb"`, `name: "test"`, and `password: "7iaw3Ur350mqGo7jwQrpkj9hiYB3Lkc/iBml1JQODbJ6wYX4oOIabXv2hUruwNRwc9rHW3S"`.

The screenshot shows the Firebase Realtime Database interface. The path is `quizzes > 0SzkoPnKcUU5...`. The left sidebar shows the `quizapp-a1bfd` project with a `quizzes` collection and a `users` collection. The `quizzes` collection has one document, `0SzkoPnKcUU50u4W296W`, which is expanded. This document contains fields: `createdBy: "OQUyXsAvbYIWryZZgAqb"`, `id: "0SzkoPnKcUU50u4W296W"`, `name: "a"`, and `quiz: "[{"class": "com.example.quizapp.model.Flashcard", "data": {"question": "a", "answer": "b"}}]"`. A `tags` field is also present with a single entry: `"MATH"`.

Email

Log in      Register

<p>This is a test quiz name</p> <p>Points: 0/2</p> <p><input type="text"/></p> <p><b>This is a test question</b></p> <p><small>Hint Show Answer</small></p> <p><small>Previous</small>      <small>Next</small></p>	<p>This is a test quiz name</p> <p>Points: 0/2</p> <p><input type="text"/></p> <p><b>This is a test question</b></p> <p><small>Hint Show Answer</small></p> <p><small>Previous</small>      <small>Next</small></p>
<p>This is a test quiz name</p> <p>Points: 0/2</p> <p><input type="text"/></p> <p><b>test question Second</b></p> <p><input type="radio"/> One <input type="radio"/> Two <input type="radio"/> Three <input checked="" type="radio"/> Four</p> <p><small>Hint Show Answer</small></p> <p><small>Previous</small>      <small>Next</small></p>	<p>This is a test quiz name</p> <p>Points: 0/2</p> <p><input type="text"/></p> <p><b>test question</b></p> <p><input checked="" type="radio"/> 1 <input type="radio"/> 2 <input type="radio"/> 3 <input type="radio"/> 4</p> <p><small>Hint Show Answer</small></p> <p><small>Previous</small>      <small>Next</small></p>