

# Splatournament

## Inhalt

Was ist gut gelaufen?	2
Welche Arbeitspakete wurden erledigt?	2
1. Front-End	2
2. Back-End	2
Welche Probleme sind aufgetreten?	3
Wie liege ich im Zeitplan?	3
Was wird bis zur nächsten Besprechung umgesetzt?	3

## Was ist gut gelaufen?

Das Layout aller bisherigen Fragments ist gut zu implementieren gewesen und schaut bisher ganz gut aus. Die initiale Verbindung mit der Datenbank hat relativ schnell funktioniert und ich habe seither keine Schwierigkeiten mit der API generell gehabt.

## Welche Arbeitspakete wurden erledigt?

Bei Arbeitspaket 1 wurden folgende Punkte erledigt:

### 1. *Front-End*

Einrichten des Android-Studio-Projekts und Erstellen der folgenden Fragmente:

- ☒ ~~Login / Start fragment~~
- ☒ ~~Sign-Up fragment~~
- ☒ ~~Home fragment~~
- ☒ ~~Tournament List fragment~~
- ☐ Add Tournament fragment
- ☐ Leaderboard Fragment

4/6 = 66.6%

### 2. *Back-End*

Einrichten der Spring-Boot-API und Erstellen der folgenden Endpunkte:

- ☒ ~~Get List of Current Tournaments~~
- ☐ Get List of Past Tournaments
- ☒ ~~Create Tournament~~
- ☐ Get Leaderboard by Team
- ☐ Get Leaderboard by Player
- ☒ ~~Add new user~~
- ☒ ~~Verify user login~~

4/7 = 57%

## **Welche Probleme sind aufgetreten?**

Es sind Probleme mit den Arbeitspaketen aufgetreten, weil ich darin nicht festgehalten habe, dass man vor dem Erstellen eines Tournaments dafür mehrere Teams braucht und das nicht in den Arbeitspaketen enthalten war. Außerdem gab es ein paar Probleme unter Android, wobei eine LiveData ein Fragment immer wieder benachrichtigt hat obwohl es das nicht hätte tun sollen.

## **Wie liege ich im Zeitplan?**

Im Zeitplan liege ich eigentlich ganz gut, da das erste Arbeitspaket bis 9.5 ist und ich somit noch genügend Zeit zum Abschließen der restlichen Funktionalität habe.

## **Was wird bis zur nächsten Besprechung umgesetzt?**

- Die Restlichen offenen Punkte von Arbeitspaket 1
- Die restliche Tournament Funktionalität, dh. alles was noch mit Tournaments zusammenhängt und nicht Leaderboards und der Stat Generator