

# Programmierprojekt - Moeraki Kemu

## 1. Treffen

April 26, 2017

## **Ansprechpartner**

### **Erreichbarkeiten der Betreuer**

#### **Niklas**

Mail [heinsohn@informatik.uni-tuebingen.de](mailto:heinsohn@informatik.uni-tuebingen.de)  
Telefon +49 7071 29-70484  
Büro C107

#### **Henry**

Mail [henry.foerster@student.uni-tuebingen.de](mailto:henry.foerster@student.uni-tuebingen.de)  
Telefon +49-7071-2970480  
Büro C 108

### **Bei Problemen**

Falls Beschwerden oder Probleme auftreten sollten, insbesondere wenn sich jemand schlecht behandelt fühlt oder sonst etwas auf dem Herzen hat, dass sie oder er nicht direkt mit den Betreuern besprechen möchte, zum Beispiel weil es diese unmittelbar betrifft, dann steht Olivia zur Verfügung als Mittelsfrau. Dafür hat sie sich dankbarerweise bereit erklärt. Sie ist erreichbar via [olivia.schuele@student.uni-tuebingen.de](mailto:olivia.schuele@student.uni-tuebingen.de)

## **Das Programm**

Unser Projekt ist natürlich die Implementierung des Moeraki Kemu (Regeln unter <http://www.kiehly.de/index.php/moeraki-kemu/spielbeschreibung/spielregeln>). Die Programmiersprache der Wahl ist Java. Die empfohlene Entwicklungsumgebung ist Eclipse.

### **Guter Code**

Guter Code tut was er soll - nicht mehr und nicht weniger. Zusätzliche Funktionalität bringt Fehleranfälligkeit mit sich, daher sollte Code knapp gehalten werden. Weiterhin ist auf vernünftige, lesbare Bezeichnungen für Klassen, Methoden und Variablen zu achten.

### **Anforderungen**

Die gestellten Anforderungen, d.h. die Mindestvoraussetzungen, welche das Spiel final erfüllen sollte, sind folgende

- Timer im 2-Spieler-Modus)
- Möglichkeit in Laufende Spiele einzusteigen, auch als Zuschauer
- Turniermodus (bsp. 3 Spieler, die abwechselnd gegeneinander spielen)

- Trainingsmodus (1vs1 lokal)
- Speichern und Wiederaufnehmen von Spielen
- Replays von Spielen
- Serverseitige Highscore
- Regelwerk einbinden
- Lobby mit Stärke von Spielern, Möglichkeit andere herauszufordern
- optional nach 3 Monaten und abhängig vom Fortschritt: rudimentäre KI

## Hauptkomponenten

Das Programm wird in seiner Implementierung in drei Hauptkomponenten aufgeteilt, die jeweils von 3er Teams hauptsächlich erarbeitet werden.

### GUI

Design, Designfunktionalität, Usability, etc.  
Verantwortliche: Jonas, Chris, Kim

### Backend

Speicher, Gewinnbedingungen, Logik, etc.  
Verantwortliche: Inge, Olivia, Felix

### Netzwerk

technischer Aspekt: TCP, Pakete senden/empfangen, etc.  
Verantwortliche: Sören, Erik, Markus

## Kommunikation

Als allumfassender zentraler Pfeiler muss und wird die Kommunikation bei einem Team dieser Größe im Vordergrund stehen. Sowohl innerhalb der Unter-Teams, als auch zwischen diesen, so insbesondere bei Neu-Implementierungen, muss die Kommunikation quantitativ und qualitativ hochwertig sein.

### Logbuch

Als ein Mittel zur Dokumentation von Problemen und deren Lösungen werden alle im Team angehalten, ein Logbuch zu führen. Hier sollen eben Probleme die beim Programmieren und Entwickeln entstehen aufgezeichnet werden. Etwaige Lösungsideen und auch die Begründungen für Entscheidungen für bestimmte Lösungen sind ebenfalls hilfreich. Diese Aufzeichnungen sollen als Hilfestellung für andere / bei sich wiederholenden Problemen unterstützend wirken.

### Regelmäßige Treffen

Mittwoch 16 Uhr, c.t. in C109 auf dem Sand  
Inhalte werden aktueller Stand und Ausblicke auf die Arbeit für die jeweils folgende Woche sein  
Weitere Treffen und Absprachen sind natürlich jederzeit möglich und gewünscht. Falls Räume dafür gebraucht werden, bitte an Niklas oder Henry wenden.

### Discord

Als Hilfestellung zur Kommunikation wird ein Discord-Server eingerichtet und beim Treffen am 03. Mai kurz vorgestellt.

# Ausblick und Arbeitsanweisungen

## Prototyp

In den kommenden 4-5 Wochen soll ein erster lauffähiger Prototyp erstellt werden. Dabei ist darauf zu achten, nicht zu viele Baustellen aufzumachen und sich vor allem auf die grundlegende Funktionalität zu konzentrieren. Eine gute erste Orientierung ist der Trainingsmodus. Sollte dieses Ziel erreicht werden, winkt ein Kasten Bier!

## Finales Projekt

Das Projekt sollte gegen Vorlesungsende, je nachdem spätestens bis Mitte August fertig sein. Die Bewertung des Projektes erfolgt dann als gemeinsame Note für das Team.

## Einführungsvorträge

Beim 2. Treffen am 03. Mai gibt es einige einführende Vorträge zu Software, die wir im Laufe des Semesters nutzen werden

### Git

Kim wird eine kurze Einführung in Version Control am Beispiel Git geben, insbesondere mit Hinblick auf ein bestehendes Eclipse-Plugin

### JUnit

Inge wird eine kurze Einführung in JUnit geben, ein Framework für Unit-Tests in Java-Umgebungen

### Javadoc

Sören wird eine kurze Einführung in Javadoc geben, ein Dokumentations-Tool zur automatischen Generierung von html-Dateien aus Java-Code

### Discord

Markus wird eine kurze Einführung in Discord geben, was uns als Basis zur Kommunikation dienen soll

Hinweis: Die Zusammenfassung des Treffens erhebt keinen Anspruch auf Vollständigkeit und soll lediglich zur Gedächtnisstütze dienen. Gerne dürfen Stellen erweitert oder korrigiert werden.