

# Pflichtenheft Mau-Mau

## Inhalt

1. Überblick .....	2
1.1    Allgemeine Beschreibung .....	2
1.2    Anwendungsbereiche .....	2
1.3    Zielgruppe .....	2
2. Funktionen .....	2
2.1    Muss-Anforderungen .....	2
2.1.1    Grafische Oberfläche .....	2
2.1.3    Benutzername eingeben .....	2
2.1.4    Karten mischen – Karten austeilen .....	2
2.1.5    Karten werfen/Kontrolle ob die Karte passt .....	2
2.1.6    Sonderkarten .....	3
2.1.7    Stabilität .....	3
3. Systemanforderungen .....	3
3.1    Software .....	3
3.2    Hardware .....	3
4. Use Case Diagramme .....	4
5. Klassendiagramm .....	5
6.1    Startbildschirm .....	6
6.2    Spielbildschirm .....	7
7. Entwicklungsumgebung .....	7
8. Projektplanung .....	7
8.1    Vorgehensmodell .....	7
8.2    Arbeitsteilungs Tabelle .....	8
9. Zeit- bzw. Kostenschätzung .....	8
9.1    Zeitschätzung .....	8
9.2    Kostenschätzung .....	8
Abbildung 1: UseCase StartScreen .....	4
Abbildung 2: UseCase PlayScreen .....	4
Abbildung 3: Klassendiagramm .....	5
Abbildung 4: Startbildschirm .....	6

## 1. Überblick

### 1.1 Allgemeine Beschreibung

„Mau-Mau“ ist ein Kartenspiel dar, welches das Spielen von Mau-Mau, dem traditionellen Südtiroler Kartenspiel, gegen deinen Freund ermöglicht.

Folgende Funktionen sollen im Programm realisiert werden:

- Jeder Benutzer soll seinen Benutzernamen eingeben können
- Karten mischen – Karten austeilen
- Karten werfen/Kontrolle ob die Karte passt
- Sonderkarten müssen ihre Funktion erfüllen
- Karte ziehen

### 1.2 Anwendungsbereiche

Personen können dieses Produkt nutzen, um ihr Spielkönnen auf die Probe zu stellen. Man spielt das Spiel auf einem Rechner, zu jeweils immer 2 Personen. Sobald ein Spieler gelegt hat, gibt er den Rechner weiter, und der nächste Spieler ist dran.

### 1.3 Zielgruppe

Mau-Mau ist für jene Personen gedacht, die in ihrer Freizeit gerne Karten spielen. Egal ob nur die Karten vergessen wurden oder die Benutzer digital begeistert sind; Mau-Mau ist für Jedermann geeignet!

## 2. Funktionen

### 2.1 Muss-Anforderungen

#### 2.1.1 Grafische Oberfläche

Das Spiel muss eine übersichtliche sowie gut strukturierte grafische Oberfläche gewähren. Alle Funktionen sollen durch die Oberfläche zugänglich und einfach anwendbar sein. Sie sollte einigermaßen realitätsgetreu sein bzw. dem echten Spiel so nahe wie möglich kommen.

#### 2.1.3 Benutzername eingeben

Nachdem man den Play – Button auf dem Startbildschirm gedrückt hat, können die zwei Spieler ihre Benutzernamen eingeben.

#### 2.1.4 Karten mischen – Karten austeilen

Nach jeder Runde sollen alle Karten zufällig gemischt werden, und danach wieder richtig ausgeteilt werden.

#### 2.1.5 Karten werfen/Kontrolle ob die Karte passt

Wenn ein Benutzer an der Reihe ist, kann er eine seiner übriggebliebenen Karten auswählen. Danach soll die ausgewählte Karte auf das Spielfeld abgeworfen werden. Das Programm kontrolliert dann, ob die ausgewählte Karte auf die, auf dem Stapel liegende Karte passt, d.h entweder gleiche Farbe oder gleicher Schlag.

### 2.1.6 Sonderkarten

Jede Karte mit einer besonderen Funktion, muss diese Funktion auch erfüllen:

- Ober: Sobald ein Ober auf dem Feld liegt, muss der nächste Spieler 2 Karte aufheben. Falls der andere Spieler auch einen Ober hat, kann dieser entscheiden, ob er 2 Karten aufhebt, oder seinen eigenen Ober darauflegt. Dann müsste der nächste 4 Karten aufheben.
- Ass: Wenn einer der Spieler ein Ass aufs Feld legt, ist der nächste Spieler ausgelassen, und der gleiche Spieler ist somit dann nochmal dran.
- Neuner: Wenn sein Spieler einen Neuner aufs Feld legt, darf er sich wünschen, mit welcher Farbe es danach weitergehen soll. Zudem kann man einen Neuner auf alle Karten drauflegen.

### 2.1.7 Stabilität

Das Programm soll fehlerfrei und stabil laufen. Es darf keine Möglichkeiten zu betrügen geben.

## 3. Systemanforderungen

### 3.1 Software

- Java 8 muss vorinstalliert sein

### 3.2 Hardware

Für Mau-Mau werden nur sehr niedrige Anforderungen benötigt. Im Grunde ist jeder Rechenfähige PC in der Lage das Spiel auszuführen.

Folgende Hardware Komponenten werden benötigt:

- Maus
- Tastatur
- Bildschirm
- Rechenfähiger PC

## 4. Use Case Diagramme

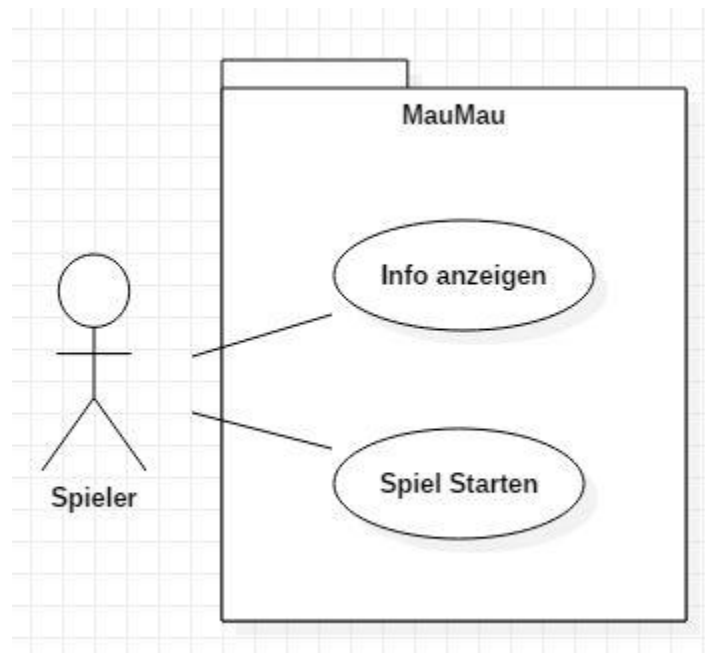


Abbildung 1: UseCase StartScreen

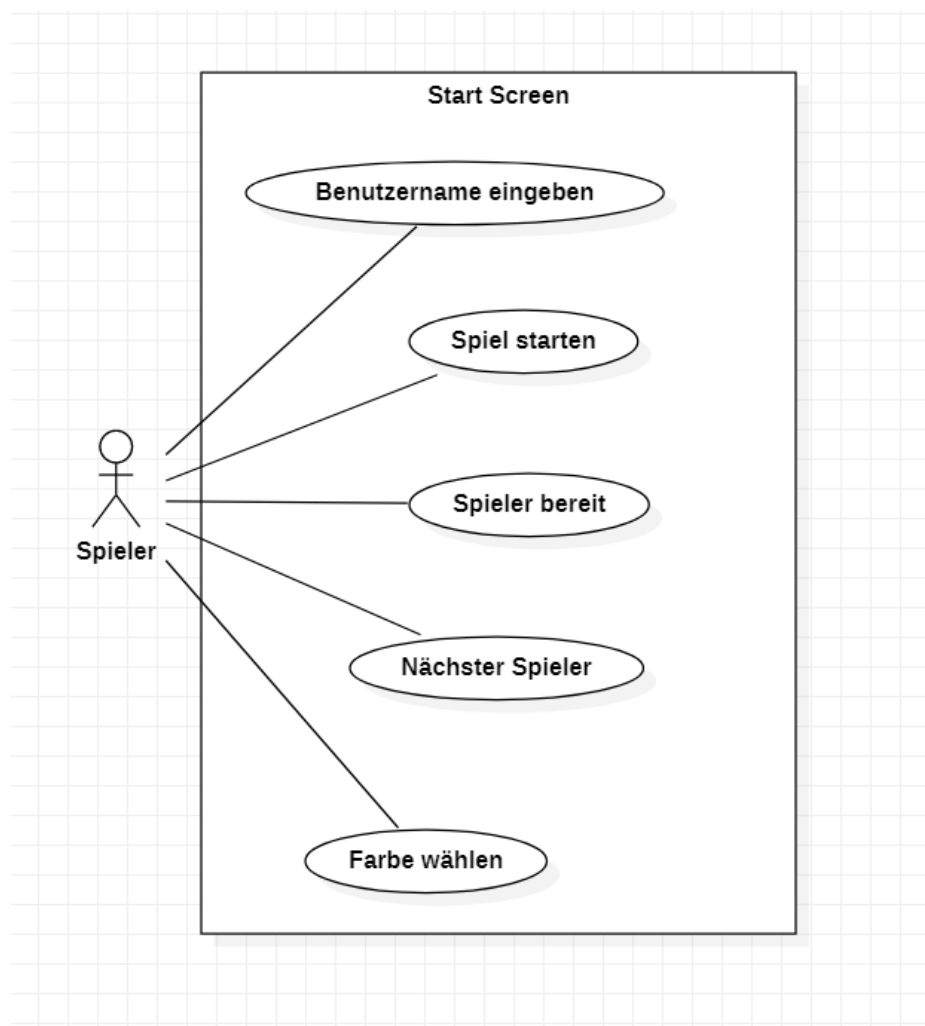


Abbildung 2: UseCase PlayScreen

## 5. Klassendiagramm

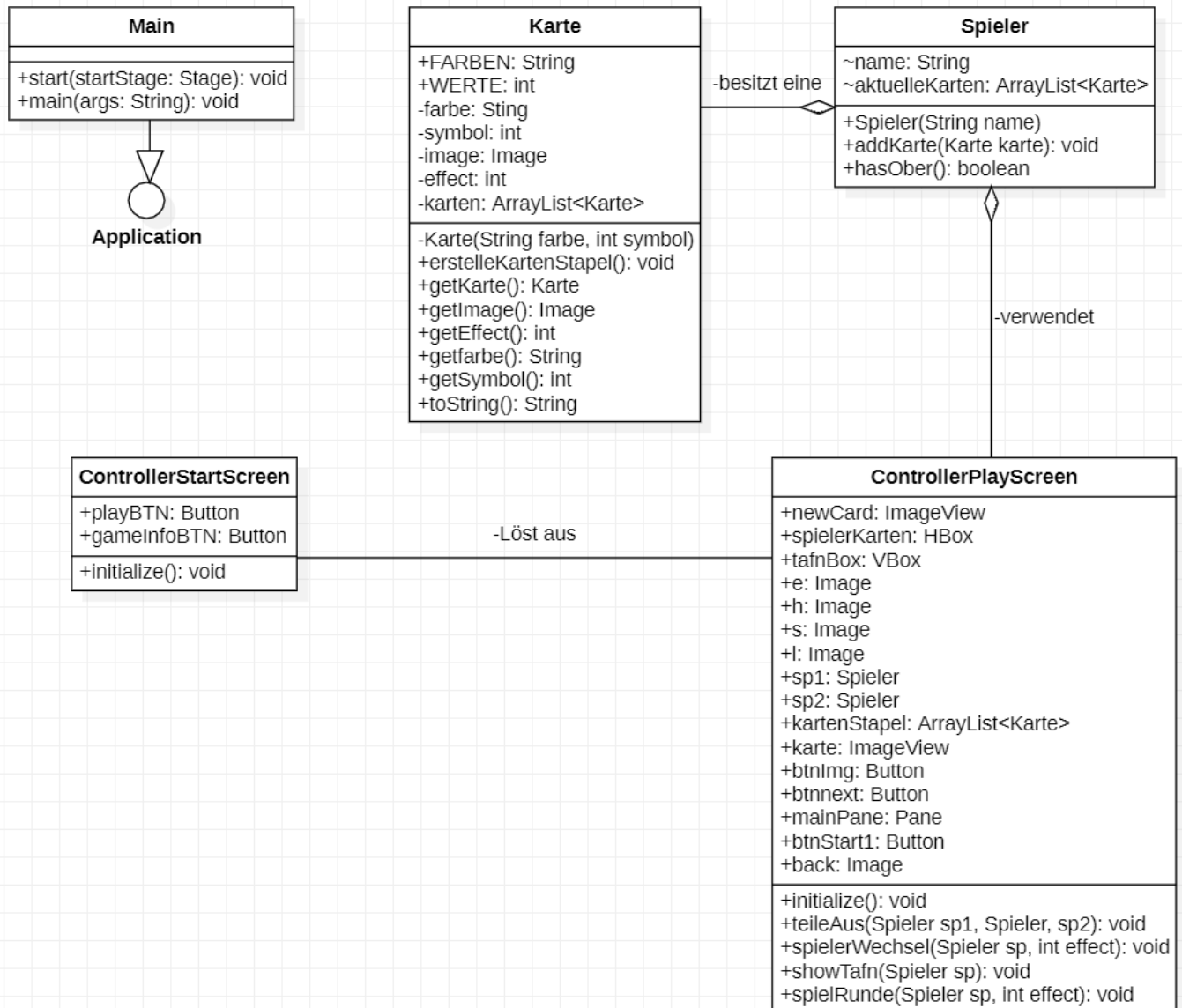


Abbildung 3: Klassendiagramm

## 6. Design/UI

---

*Alle Designs sind Prototypen des Endprodukts und können sich noch stark ändern*

---

### 6.1 Startbildschirm

Grundsätzlich haben wir uns für ein modernes, dunkles Design entschieden, dass bei langen Spielsessions die Augen weniger reizt und bei neueren Geräten mit OLED Technologie sogar Akku spart. Dennoch wollten wir die Originalität des Spiels beibehalten indem wir die Symbole der vier „Farben“ ins Design integrieren

Beim Starten des Programmes wird man anfangs von einem Startbildschirm begrüßt. Man hat die Möglichkeit das Spiel direkt zu starten oder sich zuerst unter „Game Info“ anzusehen wie man spielt.



Abbildung 4: Startbildschirm

Das Spiel wird direkt nach dem Drücken des „Play“ Buttons gestartet; Sobald man den Play Button gedrückt hat, kommt man zur Benutzernameneingabe.

Direkt nach der Eingabe der beiden Benutzernamen kann das Spiel beginnen.

## 6.2 Spielbildschirm

Das Design des Spielbildschirms soll ansprechend und einfach sein, sodass jedermann das Produkt verwenden kann. Ziel war es hier ebenfalls, eine moderne Oberfläche mit klassischen Elementen zu gestalten.

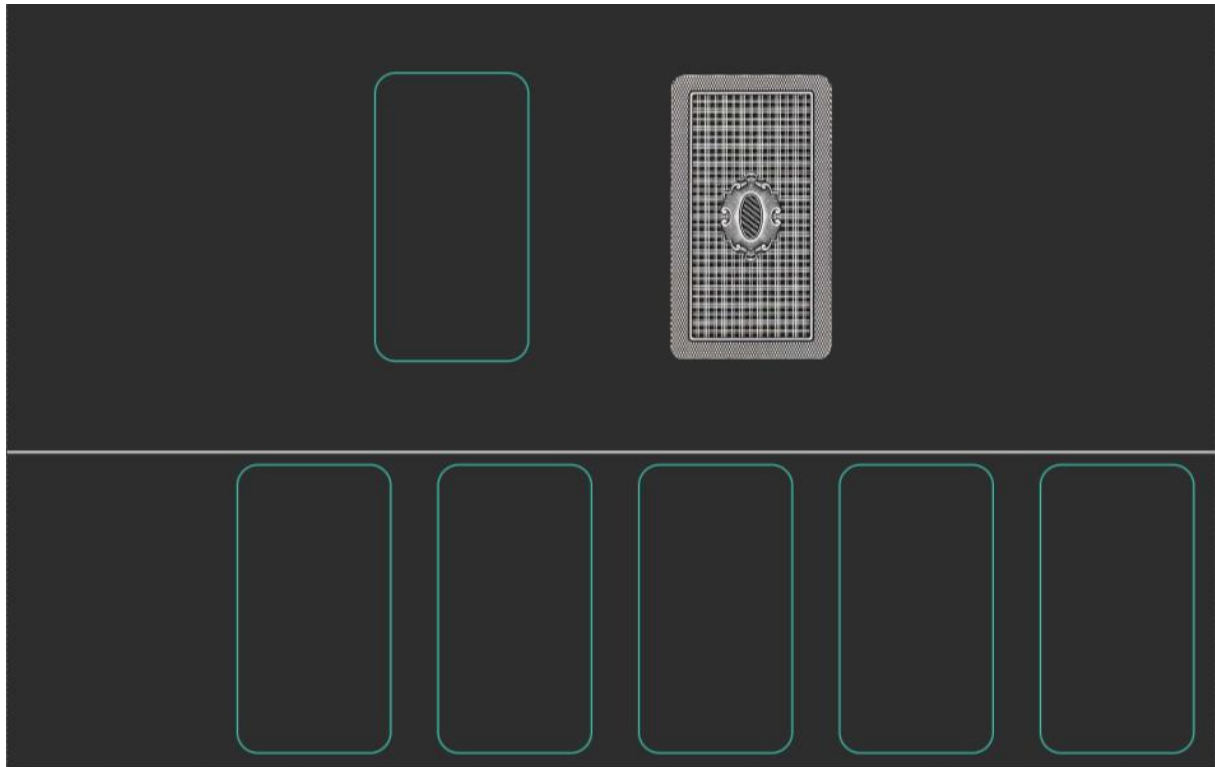


Abbildung 5: Spielbildschirm

## 7. Entwicklungsumgebung

- IntelliJ IDEA 2019.2.3
- Java Version <Version>
- JavaFX

## 8 Projektplanung

### 8.1 Vorgehensmodell

Zur Planung und Durchführung der Entwicklung verwenden wir das Vorgehensmodell „Extreme Programming“. Hauptmerkmale des agilen Vorgehensmodells sind Kundenfeedback und Kommunikation, Tests und Pair Programming. Diese Punkte werden versucht so gut wie möglich umzusetzen.

## 8.2 Arbeitsteilungs Tabelle

Datum	Person	Aufgabe
13.04.20	Philipp	Pflichtenheft
13.04.20	Moritz	Design
14.04.20	Matthias	Programmcode
17.04.20	Matthias, Moritz, Philipp	Programmcode
21.04.20	Philipp	Pflichtenheft
23.04.20	Matthias	Programmcode
23.04.20	Moritz	Abschlussdesign
27.04.20	Moritz	Git Commits
27.04.20	Matthias	Trello
27.04.20	Philipp	Pflichtenheft

## 9. Zeit- bzw. Kostenschätzung

### 9.1 Zeitschätzung

Das Programm wird voraussichtlich die ganze verfügbare Zeit in Anspruch nehmen. Es könnte sein, dass manche Nachmittage für dieses Projekt draufgehen.

### 9.2 Kostenschätzung

Normalerweise berechnet unsere Gruppe 60€ pro Arbeitsstunde, da dieses Projekt von Schule beauftrag wurde machen wir es Ehrenamtlich.