

PSE

# PFLICHTENHEFT

Rukiye Devran, Tim Groß, Daniel Helmig, Orkhan Aliev,  
Florian Weber

## Inhaltsverzeichnis

# 1 Zielbestimmung

Schach ist weltweit bekannt und hat über viele Jahre hinweg eine tiefe kulturelle Bedeutung erlangt. Dadurch ist es zu einer Sportart geworden, die viele begeistert. Deshalb soll im Rahmen der Praxis der Softwareentwicklung eine Schach-App entwickelt werden, die es ermöglicht zu jeder Zeit Schach zu spielen. Spieler sollen Gegner durch eine Spielsuche finden. Außerdem können sie sich in einer Rangliste vergleichen und es wird eine Statistik über jeden Spieler geführt. Die App soll somit ein Schachspiel simulieren, wo man trotz weiter Entfernungen gegeneinander spielen kann, ohne sich gegenüber zu sitzen. Letztendlich kann man durch häufiges Spielen bessere Spielfähigkeiten erlangen, sich mit anderen Spielern vergleichen und gegen seine Freunde spielen.

## 1.1 Musskriterien

### KM1010 *Schachregeln*

- Alle Schachregeln sollen implementiert werden.
- Das Spielende soll definiert sein.

### KM1020 *Einhaltung der Schachregeln*

- Der Nutzer soll nur gültige Züge, die den Schachregeln entsprechen, ausführen können.

### KM1030 *Spielsuche*

- Es soll dem Spieler möglich sein mit anderen Spielern zu spielen und zwar:
  - Gegen einen zufälligen Gegner
  - Gegen einen bestimmten Spieler, der mithilfe einer Suchfunktion bestimmt wird.

### KM1040 *Graphical User Interface (GUI)*

- Die Oberfläche besteht aus:
  - Login Seite: Hier soll sich der Nutzer mit Hilfe eines externen Kontos anmelden können.

- Hauptmenü: Hier soll der Nutzer dazu in der Lage sein, Spiele zu starten und seine Statistiken aufzurufen.
- Statistik Seite: Der Nutzer kann seine Spielstatistiken einsehen.
- Schachbrett: Hier kann der Nutzer in einer Partie seine Züge durchführen.
- Diverse Meldungen an den Spieler

#### **KM1050 *Schachfiguren bewegen***

- Es muss möglich sein, jede seiner Figuren anzutippen und damit auszuwählen.
- Für eine ausgewählte Schachfigur müssen mögliche Züge angezeigt werden.
- Nur gültige Züge sollen ausgeführt werden können.

### **1.2 Wunschkriterien**

#### **KW1010 *Account/Gastzugang***

- Das Erstellen eines Accounts mit wählbarem und eindeutigen Benutzernamen soll möglich sein.
- Außerdem soll es die Möglichkeit geben, ohne Accounterstellung mittels eines Gastkontos zu spielen. Der Name wird automatisch vergeben.
- Gastkonten werden nach dem Schließen der App nicht gespeichert und nicht in der Spielerliste angezeigt.

#### **KW1020 *Spiel speichern***

- Beide Spieler sollen ein Spiel nach Beenden der Partie speichern können.
- Es soll eine Textdatei mit dem Partieverlauf in algebraischer Notation erzeugt und auf dem Mobilgerät abgespeichert werden.
- Benutzer können die Textdatei öffnen und die Partie auf einem Brett nachspielen.

#### **KW1030 *Einstellbare Bedenkzeit und Spielfarbe***

- Beide Spieler einer Partie sollen eine maximale Bedenkzeit haben.
- Beim Herausfordern eines zuvor gewählten Spielers soll die Bedenkzeit sowie die gewünschte Spielfarbe einstellbar sein.
- Zur Wahl der Bedenkzeit soll eine Liste mit möglichen Zeiten erscheinen.
- Bei der Farbwahl soll zwischen Weiß, Schwarz und zufällig gewählt werden können.

#### **KW1040 *Spielerchat***

Spieler können einen Chat mit dem entsprechenden Gegner führen.

- Dazu soll es während einer Partie ein Chatsymbol geben, welches ein Chatfenster öffnet.
- Dort können Nachrichten an den Gegner verschickt werden können, sowie Spielmeldungen erscheinen.

#### **KW1050 *Elosystem und Rangliste***

- Es soll für jeden Spieler eine Elozahl zur Messung der Spielerstärke existieren.
- Nach jeder Partie sollen die Werte beider Spieler entsprechend einer Formel aktualisiert werden.
- Es soll eine Rangliste existieren, in welcher alle Spieler anhand ihrer Elozahl in absteigender Reihenfolge gelistet werden .

#### **KW1060 *Anmeldung mit Facebook oder Google***

- Benutzer sollen die Möglichkeit haben, sich mit ihrem Google bzw. Facebook-Konto anzumelden.
- Der Account ist dann mit dem jeweiligen Konto verknüpft.

#### **KW1070 *Zwei Spieler auf einem Gerät***

- Spieler sollen die Möglichkeit haben, zu zweit auf einem Gerät gegeneinander zu spielen.
- Die Symbole der schwarzen Figuren sollen zur besseren Benutzbarkeit auf den

Kopf gedreht sein.

#### **KW1080 *Revanche-Button***

- Nach Beenden einer Partie sollen beide Spieler die Möglichkeit haben, einen Rückkampf zu fordern.
- Akzeptiert der Gegner die Herausforderung, so soll eine neue Partie gestartet werden.
- Die Spieler sollen bei der Revanche die jeweils andersfarbigen Spielfiguren zugeteilt bekommen wie bei der vorherigen Partie, die Bedenkzeit soll dieselbe sein.

### **1.3 Abgrenzkriterien**

**KA1010 *Schach-Engine:*** Es soll keine selbst spielende Schach-Engine implementiert werden.

**KA1020 *Zurücknahme:*** Die Spieler können Züge nicht zurücknehmen.

**KA1030 *Spielmodi:*** Bei der Spielsuche soll es keine Möglichkeit zur Modifikation des Spielmodus geben.

**KA1040 *Fremde Partien:*** Spiele von anderen Spielern können nicht live verfolgt werden.

**KA1050 *Partien einlesen:*** Es soll nicht möglich sein, andere Partien einzulesen.

**KA1060 *Mehrere Partien:*** Es ist nicht machbar, mehrere Partien gleichzeitig zu spielen.

**KA1070 *Bedenkzeit:*** Der Spieler ist nicht dazu in der Lage, seinem Gegner mehr Bedenkzeit zu gewähren.

## **2 Produkteinsatz**

### **2.1 Anwendungsbereich**

Privatpersonen sollen in der Lage sein mit anderen Personen Schach zu spielen. Die Anwendung soll dies schnell, einfach und mobil ermöglichen.

### **2.2 Zielgruppe**

Die Anwendung richtet sich an Personen mit einem Android Smartphone, die unterwegs eine Partie Schach spielen möchten.

### **2.3 Betriebsbedingungen**

Die Anwendung soll täglich 24 Stunden verfügbar sein.

Es sollen alle Versionen ab Android 4.4 unterstützt werden.

Der Server soll wartungsfrei laufen.

## **3 Produktumgebung**

### **3.1 Software**

Eine App für Mobilgeräte mit Android Betriebssystem ab Version 4.4.

Ein Java Server zur Verwaltung von Partien und Spielsuche.

### **3.2 Hardware**

Ein Internetfähiges Smartphone mit:

- Android Betriebssystem
- Touchscreen

Ein virtueller Computer



## 4 Funktionelle Anforderungen

### 4.1 Benutzerfunktionen

**F1010 *Anmelden:*** Ein Android Nutzer der auch einen Google Account besitzt, kann sich auf der Hauptseite des Apps erfolgreich anmelden. Für die Anmeldung im System sind folgende Angaben erforderlich:

- eMail Adresse, die mit Google Konto in Verbindung steht
- Google Konto Passwort

**F1020 *Gastzugang:*** Der Benutzer der keinen Google Account besitzt kann sich als Gast im System anmelden. Bei der Anmeldung bekommt er einen eindeutigen Benutzernamen vom System.

**F1030 *Abmelden:*** Der Benutzer, der sich bereits mit Google Account angemeldet hat, kann sich wieder vom System abmelden.

**F1040 *Anzeige des eigenen, persönlichen Profils:*** Der Benutzer kann sich sein persönliches Profil anzeigen lassen. Dabei sieht er seine Statistiken:

- Wie viele Spiele gespielt wurden
- Wie viele mal gewonnen wurde
- Wie viele mal verloren wurde
- Wie viele Spiele Unentschieden geblieben sind
- Elo-Zahl

**F1050 *Suche nach Benutzern:*** Der Benutzer kann mit der Suchfunktion nach anderen Benutzern des Systems anhand einer Benutzernamen suchen.

**F1060 *Anzeigen der persönlichen Profile anderer Benutzer:*** Der Benutzer kann sich von anderen Benutzern die persönlichen Profile anzeigen lassen wobei er Benutzernamen und die Statistiken sieht und die andere Benutzer können genauso sein Profil anzeigen lassen.

**F1070 *Multiplayer auf einem Gerät:*** Mit einer Multiplayer Funktion können zwei Spieler auf einem Gerät gegeneinander spielen

**F1080 *Chatten:*** Spieler können während des Spiels miteinander

## 4.2 Initialisierung

**F2010 *Eröffnung eines Spieles:*** Der Benutzer kann neue Spiele erzeugen ohne dabei einen anderen Benutzer als Gegner angeben zu müssen. Dann bekommt er vom System einen Gegner der auch genauso ein Spiel eröffnet hat. Die Farbe ist zufällig und es steht eine feste Zeitgrenze von 15 Minuten für jeden Spieler.

**F2020 *Spieleinstellungen bestimmen:*** Bei der Herausforderung einer bestimmten Person zum Spiel **F2050** kann der Herausforderer Spieleinstellungen aufstellen:

1. Zeit
2. Farbe

**F2030 *Herausfordern:*** Nachdem ein entsprechender Gegner ausgesucht wurde **F1050** kann der Benutzer ihn zum Spiel herausfordern.

**F2040 *Akzeptieren einer Herausforderung:*** Der Benutzer kann Herausforderung zum Spiel **F2030** annehmen.

**F2050 *Ablehnung einer Herausforderung:*** Der Benutzer kann Herausforderung zum Spiel **F2030** ablehnen.

## 4.3 Spielverlauf

**F3010 *Bewegungsmöglichkeiten:*** Der Benutzer kann während des Spiels falls er dran ist, einen Zug seiner Wahl unter Beibehaltung der Schach Spielregeln ziehen.

**F3020 *Unentschieden(Remis) bieten:*** Ein Spieler, der Remis anbieten möchte, tut dies, nachdem er einen Zug auf dem Schachbrett ausgeführt und bevor es seine Uhr angehalten und die des Gegners in Gang gesetzt hat.

**F3030 *Remis annehmen:*** Der Benutzer kann, wenn ihm Unentschieden angeboten wurde **F3020**, das Angebot annehmen.

**F3040 *Remis ablehnen:*** Der Benutzer kann, wenn ihm Unentschieden angeboten wurde **F3020**, das Angebot ablehnen.

**F3050 *Unentschieden bleiben:*** Ein Spiel bleibt Unentschieden falls eine von beiden Spielern Remis angeboten hat und der andere Spieler den Angebot angenommen hat oder es nach Schachregeln ein Remis Situation auftritt:

1. Falls es Patt vorliegt
2. Falls uns eine tote Stellung vorliegt
3. Wenn eine identische Stellung mit gleichen Zugmöglichkeiten und demselben Spieler am Zug mindestens zum dritten Mal auf dem Schachbrett entstanden ist

**F3060 *Aufgeben:*** Während des Spiels kann jeder Spieler jederzeit aufgeben.

**F3070 *Spiel enden:*** Spiel endet falls eine von beiden Benutzern den anderen Schachmatt gesetzt hat oder eine von Unentschieden Situationen auftritt **F3050**.

**F3080 *Rückkampf anbieten:*** Nachdem ein Spiel zu Ende gekommen ist, muss jeder Spieler den anderen Spieler einen Rückkampf anbieten zu können.

## 5 Nichtfunktionale Anforderungen

**NF1010** Das starten der App soll auf aktuellen Geräten maximal 3 Sekunden dauern.

**NF1020** Der Erstellungsprozess **F2010** einer neuen Partie darf nicht länger als 5 Sekunden dauern.

**NF1030** Die Ermittlung **F3010** an möglichen gültigen Zügen darf nicht länger als 0,1 Sekunden dauern.

**NF1040** Die Weiterleitung und Überprüfung einzelner Züge soll auf dem Server nicht länger als 2 Sekunden dauern.

**NF1050** Bei Übertragungen zwischen zwei Geräten sollen keine Pakete unbemerkt verloren gehen.

**NF1060** Nachrichten/Spielzüge sollen unverändert am Empfänger eintreffen. Sollten Änderungen vorgenommen worden sein, soll dies vom Empfänger erkannt werden können.

**NF1070** Der Server soll wartungsfrei und ohne Neustarts auskommen.

## 6 Produktdaten

### 6.1 System-Daten auf mobilen Geräten

#### PD1010 Die Einstellungen

Die Einstellungen beinhalten:

- Benutzername
- Verbindungsdaten vom Server
- Sonstige Einstellungen

#### PD1020 Spieldaten von der aktuell laufender Partie

Zu den Spieldaten gehört:

- Partiekennung
- Aktueller Zustand des Schachbretts

### 6.2 System-Daten auf zentralem Server

#### PD2010 Server-Einstellungen

#### PD2020 Partie-Informationen

Für jede Partie wird folgendes gespeichert:

- Partiekennung
- Benutzername beider Spieler
- Aktueller Zustand des Schachbrettes
- Alle bisherigen Züge

## 6.3 Benutzer-Daten auf mobilen Geräten

### PD3010 Benutzerinformationen

Folgende Daten werden über jeden Benutzer gespeichert:

- Benutzername
- Eigene Spielstatistiken

## 6.4 Benutzer-Daten auf zentralem Server

### PD4010 Benutzerinformationen

Folgende Daten werden über jeden Benutzer gespeichert:

- Benutzername
- eMail-Adresse
- Passwort
- Spielstatistiken

Zusätzlich wird eine globales Leaderboard über die besten Spieler gespeichert.

## 7 GUI Entwürfe

## 8 Globale Testfälle

**T1010** Ein Android Nutzer laden die Applikation und registriert sich im System **F1010**

**T1020** Ein bereits registrierter Nutzer meldet sich mit seinen Benutzernamen und seiner Passwort in der Applikation an **F1020**

**T1030** Ein bereits registrierter Nutzer meldet sich vom System ab **F1030**

**T1040** Ein Gastspieler bekommt vom System einen eindeutigen Benutzernamen **F1040**

**T1050** Ein bereits registrierter Benutzer fordert seine Passwort **F1050**

**T1060** Ein bereits registrierter Benutzer ändert seine Passwort **F1060**

**T2010** Der Benutzer erzeugt ein neues Spiel **F2010**

**T2020** Der Benutzer führt ein schon eröffnetes Spiel fort **F2020**

**T2030** Der Benutzer fordert unter Angabe eines gültigen Benutzernamens einen anderen Benutzer heraus **F2030**

**T2040** Der Benutzer nimmt die Herausforderung an **F2040**

**T2050** Der Benutzer lehnt die Herausforderung ab **F2050**

**T2060** Der Benutzer schaut seine Statistiken an **F2060**

**T2070** Der Benutzer sendet eine Freundschaftsanfrage **F2070**

**T2080** Der Benutzer akzeptiert die Freundschaftsanfrage **F2080**

**T2090** Der Benutzer lehnt die Freundschaftsanfrage ab **F2090**

**T3010** Der Benutzer macht irgendeinen Zug seiner Wahl, wobei die Schachregeln eingehalten werden müssen **F3010**

**T3020** Der Benutzer bietet ein Remis an, vorausgesetzt er ist am Zug **F3020**

**T3030** Der Benutzer akzeptiert ein Remis **F3030**

**T3040** Der Benutzer lehnt ein Remis ab **F3040**

**T3050** Der Benutzer tauschen Nachrichten aus **F3050**



## 9 Systemmodelle

### 9.1 Szenarien

#### **Szenario 1: „Spiele gegen zufälligen Spieler“**

Max Mustermann möchte eine Runde Schach in seiner Mittagspause spielen. Er holt sein Smartphone aus der Tasche und klickt auf das Appsymbol. Da er die App noch nie verwendet hat muss er sich zuerst anmelden. Nachdem er dies getan hat befindet er sich im Hauptmenü. Da keiner seiner Kollegen Zeit hat, möchte er gegen einen zufälligen Gegner spielen. Nun klickt er auf „Spiel suchen“ und bekommt die Meldung, dass er nun der Spielsuche hinzugefügt wurde. Nach kurzer Zeit bekommt er einen Gegner zugewiesen und die Partie beginnt. Vor ihm erscheint das Schachbrett in seiner Ausgangsposition.

Herr Mustermann hat die Farbe Weiß zugewiesen bekommen und sein Gegner Schwarz. Er wählt einen Bauern an und bekommt seine möglichen Züge mit dieser Schachfigur angezeigt. Max Mustermann macht seinen Zug. Als nächstes zieht sein Gegner. Nun ist er wieder am Zug. Beide ziehen nun immer abwechselnd. Nach 35 Zügen hat Herr Mustermann seinen Gegner Schachmatt gesetzt. Die Partie ist somit beendet und Max Mustermann bekommt die Meldung, dass er gewonnen hat. Ihm steht nun zur Auswahl ob er eine Revanche anbieten möchte, die Partie auf seinem Gerät speichern will, oder einfach nur ins Hauptmenü möchte. Er klickt auf den Button „Zum Hauptmenü“ und wird zum Hauptmenü weitergeleitet.

#### **Szenario 2: „Spiele gegen einen Freund“**

Magnus und Fabiano wollen in ihrer Freizeit gegeneinander Schach spielen, haben aber gerade kein Schachbrett zur Hand. Beide öffnen auf ihren Android-Gerät ihre Schach-App, und Magnus klickt auf den Button „Spieler suchen“. Es öffnet sich eine Spielerübersicht mit einer Suchleiste, in welcher er den Spielernamen von Fabiano eingibt. In der Spielerübersicht erscheint Fabianos Profil, welches Magnus anklickt. Daraufhin öffnet sich Fabianos Profilübersicht mit seinen Statistiken. Auf Fabianos Profil betätigt Magnus den Button „Herausfordern“, und es erscheint ein Auswahlfenster. Magnus wählt als Bedenkzeit 2 Stunden und die zufällige Farbenverteilung. Fabiano erhält daraufhin eine Mitteilung, dass er von Magnus zu einem Spiel mit Bedenkzeit 2h herausgefordert wurde, und hat die Möglichkeit, dieses anzunehmen oder abzulehnen. Er klickt auf „Annehmen“ und beiden Spielern öffnet sich ein Schachbrett in Ausgangsposition.

Magnus bekommt die schwarzen Figuren zugeteilt, Fabiano die weißen. Nach 115 Zügen ist die Stellung immer noch ausgeglichen und Fabiano klickt auf „Remis anbieten“. Magnus erhält eine Benachrichtigung mit der Möglichkeit, das Remis anzunehmen oder abzulehnen. Auch er sieht in der Stellung keine Gewinnmöglichkeit und klickt deshalb auf „Akzeptieren“. Das Spiel wird beendet und die Statistiken beider Spieler werden aktualisiert. Beide Spieler erhalten eine Mitteilung mit der Möglichkeit, eine Revanche zu

fordern und das Spiel abzuspeichern. Da Magnus das Spiel gerne noch einmal ansehen möchte, klickt er auf „Spiel speichern“ und erhält eine Textdatei mit der Notation der Partie auf sein Smartphone.

### **Szenario 3: „Statistiken einsehen“**

Magnus behauptet, er wäre ein besserer Spieler als Fabiano. Dieser zweifelt das an, und bittet Magnus, dessen Statistiken mit seinen zu vergleichen. Beide öffnen ihre Schachapp und sind direkt mit ihrem Account angemeldet, da sie dies vorher eingestellt haben. Anschließend klicken sie im Hauptmenü ihrer Schach-App auf „Statistiken“. Sie können nun jeweils ihre Anzahl an gespielten, gewonnenen, remisierten und verlorenen Spielen, sowie ihre Elozahl einsehen. Beide kommen zu dem Schluss, dass Magnus bessere Werte vorzuweisen hat.

## 9.2 Anwendungsfälle

1. **Zustand:** Die App ist geschlossen und beendet.

**Aktion:** Die App wird erstmalig geöffnet.

**Reaktion:** Es erscheint ein Fenster zur Eingabe eines Spielernamens.

2. **Zustand:** Die App ist geschlossen und beendet.

**Aktion:** Die App wird zum wiederholten Mal geöffnet.

**Reaktion:** Es erscheint das Hauptmenü und der Benutzer ist unter seinem Spielernamen eingeloggt.

3. **Zustand:** Die App befindet sich im Hauptmenü.

**Aktion:** Der Benutzer klickt auf den Button „Statistiken“.

**Reaktion:** Es öffnet sich eine Accountübersicht mit allen gespeicherten Statistiken.

4. **Zustand:** Die App befindet sich im Hauptmenü.

**Aktion:** Der Benutzer klickt auf den Button „Spiel suchen“.

**Reaktion:** Der Benutzer kommt in eine Warteschlange für suchende Spieler.

5. **Zustand:** Die App befindet sich im Hauptmenü.

**Aktion:** Der Benutzer klickt auf den Button „Spieler suchen“.

**Reaktion:** Es erscheint eine Spielerübersicht mit Suchmöglichkeit.

6. **Zustand:** Die App befindet sich im Hauptmenü.

**Aktion:** Der Benutzer klickt auf den Button „Rangliste“.

**Reaktion:** Die Top 10 Spieler werden nach Elozahl sortiert aufgelistet.

7. **Zustand:** Die App befindet sich im Spielersuchmenü.

**Aktion:** Der Benutzer klickt auf einen anderen Spieler.

**Reaktion:** Es erscheint eine Accountübersicht des Spielers mit Möglichkeit zur Herausforderung.

8. **Zustand:** Ein Spiel ist am laufen und der Nutzer ist am Zug.

**Aktion:** Der Benutzer klickt auf eine eigene Figur.

**Reaktion:** Es werden alle möglichen Zugfelder markiert.

9. **Zustand:** Ein Spiel ist am laufen und der Nutzer ist am Zug.

**Aktion:** Der Benutzer klickt auf eine gegnerische Figur oder ein leeres Feld.

**Reaktion:** Nichts passiert.

10. **Zustand:** Eine Schachfigur wurde angeklickt.

**Aktion:** Der Benutzer klickt auf ein markiertes Feld.

**Reaktion:** Die Figur wird gezogen und die Daten an den Server gesendet.

11. **Zustand:** Ein Spiel ist am laufen.

**Aktion:** Es wird ein Zug ausgeführt, welcher einen Spieler Matt oder Patt setzt.

**Reaktion:** Es erscheint eine Benachrichtigung, die Elozahlen und Statistiken der Spieler werden aktualisiert, das Spiel wird beendet und abgespeichert.

12. **Zustand:** Ein Spiel ist am laufen.

**Aktion:** Ein Spieler klickt auf den Button „Aufgeben“.

**Reaktion:** Es erscheint eine Benachrichtigung, die Elozahlen und Statistiken der Spieler werden aktualisiert, das Spiel wird beendet und abgespeichert.

13. **Zustand:** Ein Spiel ist am laufen.

**Aktion:** Ein Spieler bietet Remis an.

**Reaktion:** Der andere Spieler erhält eine Benachrichtigung mit Auswahlmöglichkeit.

14. **Zustand:** Remis wurde angeboten.

**Aktion:** Der Spieler klickt auf den Button „Annehmen“.

**Reaktion:** Es erscheint eine Benachrichtigung, die Elozahlen und Statistiken der Spieler werden aktualisiert, das Spiel wird beendet und abgespeichert.

15. **Zustand:** Remis wurde angeboten.

**Aktion:** Der Spieler klickt auf den Button „Ablehnen“.

**Reaktion:** Die Benachrichtigung schließt sich, der andere Spieler erhält eine Mitteilung.

### 9.3 Anwendungsfalldiagramme

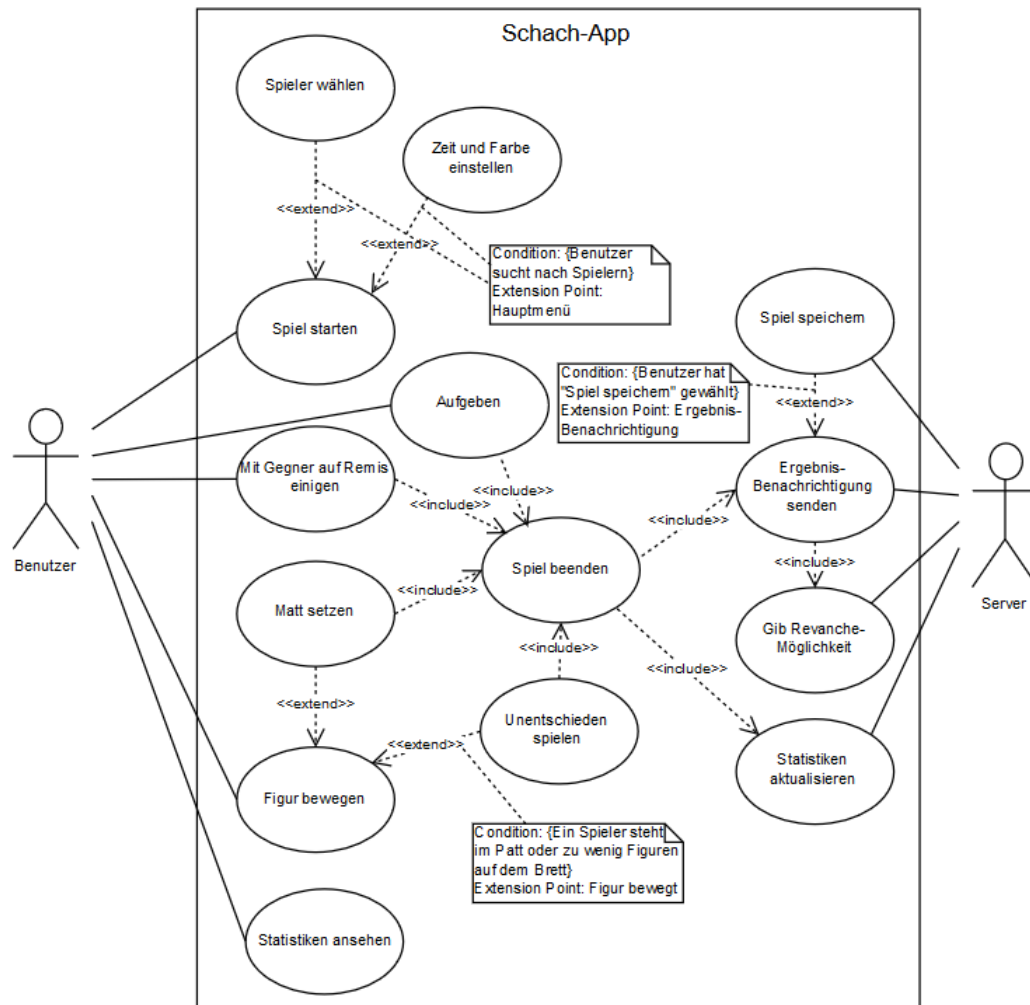


Abbildung 1: Schach-App

