

# Lastenheft: Plattform zum Vergleich von Spiele-KIs

## 1 Einleitung

*Anleitungshistorie**Blocksa*

In einer Zeit, in der künstliche Intelligenz (KI) zunehmend Einzug in verschiedene Bereiche unserer Gesellschaft hält, wird die Bedeutung des Verständnisses und der Anwendung von KI-Konzepten immer deutlicher. Insbesondere im Bildungsumfeld ist es entscheidend, Studierenden praxisnahe Erfahrungen zu ermöglichen, um ihre Fähigkeiten in der KI-Entwicklung zu vertiefen. Hierbei bietet das alpha-zero Framework [1, 2] einen interessanten Ansatz. Dieses Framework hat gezeigt, wie KI durch selbständiges Spielen von Schach, Shogi und Go meisterhaft werden kann.

Trotz dieser Möglichkeiten besteht jedoch bisher ein Mangel an einer benutzerfreundlichen Plattform, die es ermöglicht, das Verhalten von KIs durch spielerische Erfahrungen zu erkunden, verschiedene KIs miteinander zu vergleichen und eigene KIs zu testen. Dies ist besonders bedeutsam, um den Lernprozess im Bereich der KI-Entwicklung spielerisch zu gestalten und Interessierten praxisnahe Erfahrungen zu vermitteln. Das Projekt setzt an der Stelle an, um diese Lücke zu schließen und den Lernprozess im Bereich der KI-Entwicklung spielerisch zu gestalten.

Das Ziel ist es, eine intuitive und zugängliche Plattform zu entwickeln, die es den Benutzern erlaubt, Spiele gegen KIs zu spielen und ihre eigenen KIs zu testen und zu vergleichen. Durch die direkte praktische Anwendung von KI-Konzepten erhalten die Nutzer nicht nur einen tieferen Einblick in die Welt der KI-Entwicklung, sondern können auch ihre Fähigkeiten auf eine unterhaltsame und interaktive Weise verbessern.

Die Plattform soll einen ansprechenden Zugang zur Welt der KI-Entwicklung bieten und ist sowohl für Anfänger als auch für fortgeschrittene Benutzer konzipiert. Sie soll dabei nicht nur Wissen vermitteln, sondern auch als Werkzeug dienen, um praxisnahe Erfahrungen zu sammeln und die Fähigkeiten im Bereich der KI-Entwicklung zu erweitern.

## 2 Konzept

Das Projekt zielt auf die Entwicklung einer benutzerfreundlichen Plattform hin, die es den Nutzern ermöglicht, Spiele gegen KIs zu spielen und eigene KIs zu testen und zu vergleichen. Dabei soll besonderer Wert auf intuitive Bedienbarkeit und hohe Nutzerfreundlichkeit gelegt werden.

Die Plattform bietet Anfängern und fortgeschrittenen Benutzern gleichermaßen die Möglichkeit, praxisnahe Erfahrungen im Bereich der KI-Entwicklung zu sammeln. Durch Spiele gegen KIs und die Möglichkeit, eigene KIs zu testen und zu vergleichen, können die Benutzer ihre Fähigkeiten erweitern und ihr Verständnis für KI-Konzepte vertiefen.

Zudem soll die Plattform sich an Studierende, Entwickler und KI-Enthusiasten, die ihr Wissen und ihre Fähigkeiten im Bereich der KI-Entwicklung erweitern möchten, richten.

## 3 Funktionale Anforderungen

### 3.1 Must-Haves

- Konfiguration von zu spielenden Spielen mit Auswahl des Spiels, der Wahl des Spielmodis (Spieler gegen Spieler, Spieler gegen bereitgestellte KI und Spieler-KI gegen bereitgestellte KI) und bei

einfacheren Spielen dynamische Spielfeldoptionen. (Zeitabschätzung: 10 Stunden)

- Umgesetzte Spiele (Zeitabschätzung: 8 bis 15 Stunden pro Spiel (abhängig vom Schwierigkeitsgrad)):

- Othello (Reversi)
- Tic-Tac-Toe
- Vier gewinnt
- Nim
- Dame
- Go
- Waldmeister

Spiel Brett Konfiguration

per API / GUI ?

GUI - Ausgabe Spielzustand?

- Möglichkeit zur Erstellung oder Beitritt zu virtuellen identifizierbaren Räumen (Lobby), die noch nicht voll besetzt sind, um einen Mehrspielerbetrieb zu ermöglichen. (Zeitabschätzung: 20 Stunden)
- Bereitstellung von generierten Schlüsseln zum KI-Beitritt über ein Application Programming Interface (API). (Zeitabschätzung: 15 Stunden)

- Graphische Darstellung aktiver Spiele in den Lobbys mit PyGame. (Zeitabschätzung: 8 Stunden)

→ Welche werden unterstützt? +?

- Interaktion mit der Spieloberfläche im Spielmodus mit Mauseingabe. (Zeitabschätzung: 15 Stunden)

- Möglichkeiten des Spieleabbruchs und des Verlassens einer Lobby mit Rückkehr zur Hauptseite. (Zeitabschätzung: 5 Stunden)

- API-Bereitstellung für Spiele mit Spieler-KI-Interaktion, inklusive Anleitung und erforderlichen Dateien zur Verbindung. (Zeitabschätzung: 40 Stunden) — ?

- Für statistische Auswertungen eine Möglichkeit zur Ausführung vieler Spiele ohne graphische Oberfläche für die Spieler-KIs. (Zeitabschätzung: 15 Stunden)

- Zeitstrahl und Historie für KI-Spiele. Dabei besteht die Möglichkeit X Schritte zurückzugehen und das Spielfeld und KI-Verhalten zu analysieren. (Zeitabschätzung: 10 Stunden)

- Darstellung von Impressum und Datenschutzerklärung auf der Webseite. (Zeitabschätzung: 8 Stunden)

### 3.2 Nice-to-Haves

- Einstellbare Schwierigkeitsgrade für die vorab trainierten bereitgestellten KI-Implementierungen. (Zeitabschätzung: 4 Stunden) ⇒ Must have & Spiel weniger

- Implementierung einer Benutzeridentifikation (Benutzername) zur Speicherung von Spieler- oder KI-Punkten. Dazu eine Anzeigetafel mit Punktedarstellung. (Zeitabschätzung: 15 Stunden)

- Farbmusterauswahl der Webseite (Dark- oder Light-Mode). (Zeitabschätzung: 5 Stunden)

- Zusätzliche Konfiguration von zu spielenden Spielen mit Wahl des Spielmodis (Spieler-KI gegen Spieler-KI). (Zeitabschätzung: 15 bis 20 Stunden)

Schon Punkt 1 Must have

- Implementierung weiterer Spiele, z.B. (Zeitabschätzung: 10 bis 15 Stunden pro Spiel (abhängig vom Schwierigkeitsgrad)):

- Schach

- Mühle
- Halma
- Abalone
- Hex

- Möglichkeit zum Abspeichern von Spielen für spätere Analyse oder Fortsetzung des Spiels, sowie Lademöglichkeit abgespeicherter Spiele. (Zeitabschätzung: 3 Stunden)

↔ Zeitstrahl - Feature

### 3.3 If-Time-Allows

- Barrierefreiheit der Webseite ausbauen. (Zeitabschätzung: 20 Stunden)
  - Vollständig mausfreie Bedienung.
- Möglichkeit in der Spieleoberfläche einzelne Züge zurückzunehmen. (Zeitabschätzung: 6 Stunden) → Must-Have
- Bereitstellung der Webseite für mobile Geräte mit zusätzlicher Toucheingabe für die Spieleoberfläche. (Zeitabschätzung: .. Stunden) ??
- Mehrsprachigenunterstützung der Webseite und Anleitungen (Englisch, Deutsch). (Zeitabschätzung: 10 Stunden)
- Implementierung weiterer Spiele. (Zeitabschätzung: 15 Stunden pro Spiel):
  - Spiele mit Zufallskomponente.
- Zur Förderung des Spielespaßes eine Erweiterung der Punkteanzeigen mit Ranglisten und Erfolge. (Zeitabschätzung: 30 Stunden)

## 4 Qualitative Anforderungen

### 4.1 Benutzerfreundlichkeit

- Die Benutzeroberfläche soll intuitiv gestaltet sein, um eine einfache Navigation für Benutzer zu ermöglichen.
- Es sollen klare Anleitungen und Hilfestellungen zur Verfügung gestellt werden, um den Einstieg für neue Benutzer zu erleichtern.

→ in Frontend

### 4.2 Skalierbarkeit

- Die Plattform soll skalierbar sein, um eine steigende Anzahl von Benutzern und Spielen zu unterstützen, ohne dass die Leistung beeinträchtigt wird.
- Die Architektur der Plattform sollte flexibel genug sein, um zukünftige Erweiterungen und Anpassungen problemlos zu ermöglichen.

→ Technik?

### 4.3 Performance und Stabilität

- Die Plattform sollte eine schnelle Ladezeit und eine reibungslose Benutzererfahrung bieten, auch bei einer hohen Anzahl gleichzeitiger Benutzer.

## 5 Stakeholder und ihre Ziele

- **Studierende:** Ziel ist es, praxisnahe Erfahrungen im Bereich der KI-Entwicklung zu sammeln und ihre Fähigkeiten zu vertiefen.
- **Entwickelnde:** Ziel ist es, ihre Kenntnisse und Fähigkeiten im Bereich der KI-Entwicklung zu erweitern und neue Anwendungen zu erforschen.
- **KI-Enthusiasten:** Ziel ist es, mehr über die Entwicklung und Anwendung von KI zu erfahren und eigene KIs zu testen und zu verbessern. Zudem die Möglichkeit in verschiedenen Spielen gegen vortrainierte KIs und andere Spieler anzutreten.

*Dozenten in Pls?*

## 6 Geplante Abnahmetests

- Testen der Plattformfunktionalitäten, einschließlich Spielekonfiguration, Lobbyerstellung, API-Bereitstellung usw. *geplant*
- Testen der Benutzerfreundlichkeit durch Benutzerfeedback und Usability-Tests.
- Testen der Skalierbarkeit und Performance unter Last durch Lasttests und Leistungstests.

*Dauer?*

## 7 Dokumentation

Am Ende des Projekts wird eine umfassende Dokumentation erstellt, die folgende Punkte umfasst:

- Benutzerhandbuch: Enthält Anleitungen zur Nutzung der Plattform und ihrer Funktionen.
- Entwicklerdokumentation: Enthält Informationen zur Plattformarchitektur, API-Spezifikationen und Implementierungsdetails.
- Weiterentwicklungsdokumentation: Anleitungen zum möglichen Erweitern und Weiterentwickeln der Plattform und seiner Systeme (z.B. Spielerepertoire erweitern, Webseitenzugänglichkeit ausbauen).
- Testdokumentation: Enthält Berichte über durchgeführte Tests und deren Ergebnisse.

*Dauer?*

## 8 Literaturverzeichnis

[1] SILVER, David, et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. Science, 2018, 362. Jg., Nr. 6419, S. 1140-1144.

[2] suragnair, et. al. Alpha Zero General. [Online]. Verfügbar unter: <https://github.com/suragnair/alpha-zero-general>. [Letztes Zugriffsdatum: 25.04.2024].