Benutzerhandbuch

Softwaretechnik-Projekt SoSe2024

Thema

Plattform zum Vergleich von Spiele-KIs: **KIMaster**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Änderungshistorie** | | | | |
| **Version** | **Datum** | **Kapitel** | **Änderung** | **Name** |
| 0.1 | 04.07.2024 | Alle | Anlegen und Füllen | Justine Buß |
| 0.2 | 20.07.2024 | 4 | Schreiben | Thorben Jones |

|  |  |  |
| --- | --- | --- |
| Herausgeber | [Technische Hochschule Mittelhessen – FB06 Mathematik, Naturwissenschaften und Informatik](https://www.thm.de/mni/) | |
| Dateiname | Benutzerhandbuch\_ver\_0.2 | |
| Dokumentenbezeichnung | Technische Dokumentation: Plattform zum Vergleich von Spiele-KIs | |
| Version | 0.2 | |
| Stand | Sonntag, 21. Juli 2024 | |
| Status | In Bearbeitung | |
| Autoren | Justine Buß, Thorben Jones, Alexander Roos, Maximilian Bachmann, Omar Karkotli, Sven Reinhard, Pascal Waldschmidt | |
| Freigegeben von |  |  |
| Ansprechpartner | Justine Buß | [justine.buss@mni.thm.de](mailto:justine.buss@mni.thm.de) |
|  | Thorben Jones | [thorben.jones@mni.thm.de](mailto:thorben.jones@mni.thm.de) |
| Kurzinfo | „Technische Hochschule Mittelhessen Softwaretechnik-Projekt. Technische Dokumentation.“ | |

Inhaltsverzeichnis

[1 Einführung 6](#_Toc172457603)

[1.1 Kurzübersicht der Software 6](#_Toc172457604)

[1.2 Zielgruppen 6](#_Toc172457605)

[2 Einrichtung 6](#_Toc172457606)

[2.1 Systemanforderungen 6](#_Toc172457607)

[2.2 Ersteinrichtung und Konfiguration 7](#_Toc172457608)

[3 Benutzeranleitung für die Webseite 7](#_Toc172457609)

[3.1 Überblick über die Benutzeroberfläche 7](#_Toc172457610)

[3.2 Hauptfunktionen und deren Nutzung 7](#_Toc172457611)

[4 Benutzeranleitung für die Schnittstelle 7](#_Toc172457612)

[4.1 Überblick über die Schnittstelle 7](#_Toc172457613)

[4.2 Voraussetzungen 7](#_Toc172457614)

[4.3 Einrichtung 8](#_Toc172457615)

[4.4 Nutzung der Hauptfunktionen 8](#_Toc172457616)

[5 Häufige Probleme und Lösungen 10](#_Toc172457617)

[6 Glossar 10](#_Toc172457618)

[7 Index 11](#_Toc172457619)

Abbildungssverzeichnis

**Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.**

Tabellenverzeichnis

**Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.**

# 1 Einführung

Dieses Benutzerhandbuch dient als Leitfaden für die Navigation und Nutzung der Plattform KIMaster. Es richtet sich sowohl an Nutzer, die über die Webseite spielen möchten, als auch an solche, die komplexere Anwendungsfälle, wie die Anbindung an die Schnittstelle und die Implementierung eigener Künstlicher Intelligenzen (KI), realisieren wollen.

## 1.1 Kurzübersicht der Software

Die Software stellt eine vielseitige Plattform bereit, auf der Nutzer in verschiedenen Spielen gegeneinander antreten können. Ein Alleinstellungsmerkmal ist die Möglichkeit, gegen vortrainierte KIs zu spielen, die für jedes Spiel verfügbar sind. Die Plattform kann über eine Webseite genutzt werden oder über ein Terminal, wenn die Anbindung an die bereitgestellte Schnittstelle erfolgt.

Ein zusätzliches Feature der Software ist die Option, eigene KI-Implementierungen zu integrieren. Entwickler können ihre KIs gegen andere Spieler, deren KIs oder vortrainierte KIs testen. Dies eröffnet vielfältige Möglichkeiten für Entwicklungen im Bereich der KI.

## 1.2 Zielgruppen

Dieses Benutzerhandbuch richtet sich an eine vielfältige Nutzergruppe:

* **Allgemeiner Spieler**: Personen, die auf der Webseite spielen und gegen andere Nutzer oder vortrainierte KIs antreten möchten.
* **Studierende**: Nutzer, die erste Erfahrungen mit KI sammeln oder spezielle KI-Module belegen und ihre eigenen KIs entwickeln möchten.
* **Dozenten:** Lehrkräfte, die die Plattform für Lehr- und Forschungszwecke im Bereich KI verwenden.
* **Entwickler**: Nutzer, die die Plattform über die Schnittstelle nutzen und eigene KI-Implementierungen integrieren möchten.

# 2 Einrichtung

## 2.1 Systemanforderungen

## 2.2 Ersteinrichtung und Konfiguration

# 3 Benutzeranleitung für die Webseite

## 3.1 Überblick über die Benutzeroberfläche

## 3.2 Hauptfunktionen und deren Nutzung

(Auswahl und Start eines Spiels, Spielen gegen andere Nutzer, Spielen gegen vortrainierte KIs, Nutzung der verschiedenen Spielmodi)

# 4 Benutzeranleitung für die Schnittstelle

Die Schnittstelle bietet Methoden zum Verbinden, Senden und Empfangen von Nachrichten über eine WebSocket-Verbindung und ist so gestaltet, dass sie einfach zu erweitern und anzupassen ist.

## 4.1 Überblick über die Schnittstelle

Die KIMaster-Schnittstelle ermöglicht es, einfach und effizient mit einem WebSocket-Server zu kommunizieren. Es können verschiedene Befehle gesendet und Antworten erhalten werden in Echtzeit.

**Hauptfunktionen:**

* Verbindung zu einem WebSocket-Server herstellen.
* Befehle an den Server senden.
* Nachrichten vom Server empfangen.

Es werden Klassen und Code-Abschnitte zur einfacheren Anbindung bereitgestellt. Dabei enthält dieKIMaster-Klasse die grundlegende Infrastruktur für die WebSocket-Kommunikation, einschließlich Verbindungsmanagement, Befehlssendung und Nachrichtenerhalt. Die Example-Klasse erweitert diese Funktionalität, indem sie spezifische Befehle und Szenarien implementiert, die für die Anwendung erforderlich sind.

KIMaster**-Klasse:**

* **\_\_init**\_\_: Initialisiert die Klasse mit einer Liste von URIs.
* **connect**: Stellt eine Verbindung zu einem WebSocket-Server her.
* **send\_cmd**: Sendet Befehle an den Server.
* **receive**: Empfängt Nachrichten vom Server.
* **close**: Schließt die Verbindung.
* **run**: Führt eine Coroutine aus.
* **handler**: Verwaltet das Senden und Empfangen von Nachrichten.
* **print\_message**: Formatiert und druckt Nachrichten.
* **show**: Zeigt ein Bild aus einem Byte-Stream an.

Example-Klasse**:**

* **get\_commands**: Gibt eine Liste von Befehlen zurück.
* **main:** Haupt-Coroutine zum Verwalten der Verbindung und Kommunikation.

Ein Bild, das Text, Screenshot, Software enthält.

Automatisch generierte Beschreibung

## 4.2 Kommandos

Die Schnittstelle bietet verschiedene Befehle, die an den WebSocket-Server gesendet werden können. Diese Befehle werden als JSON-Nachrichten formatiert und enthalten spezifische Schlüssel und Daten, die vom Server verarbeitet werden. Alle Befehle sind in vollem Umfang mit ihren Möglichkeiten, Variationen und optionalen Parametern in der commands.md enthalten.

* Struktur einer zu sendenden Nachricht im JSON-Format:

{“command“: command,

“command\_key“: commandy\_key,

“data“: data}

* Struktur einer zu empfangenden Nachricht im JSON-Format:

{“response\_code“: response\_code,

“response\_msg“: response\_msg,

“data“: data}

* Beispiel für Kommandos:
  + Lobby: Statusabfrage, Positionierung, Erstellung und Verlassen von Lobbys.
  + Play: Erstellen eines Spiels mit bestimmten Parametern (z.B. Spielname, Schwierigkeitsgrad, Modus)

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

## 4.3 Voraussetzungen

Bevor begonnen werden kann, muss sichergestellt werden, dass die folgenden Voraussetzungen erfüllt sind:

* Python ist auf dem Computer installiert.

## 4.4 Einrichtung

1. **Installieren der notwendigen Bibliotheken**: Um die KIMaster-Schnittstelle zu verwenden, müssen die erforderlichen Python-Bibliotheken installiert sein. Ist dies nicht der Fall muss folgendes durchgeführt werden: Das Terminal öffnen und ausführen des folgenden Befehls zur Installation der notwendigen Bibliotheken:

Ein Bild, das Text, Schrift, Screenshot, Grafiken enthält.

Automatisch generierte Beschreibung

1. **Verbindung zum WebSocket-Server**: Verwenden Sie die KIMaster-Klasse, um eine Verbindung zum Server herzustellen. Dies wird durch die Methodeconnect() der KIMaster-Klasse hergestellt. Diese Methode versucht, eine Verbindung zu einer der URIs herzustellen.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

1. **Einbindung der Schnittstelle: Die Schnittstelle muss in das eigene Projekt eingebunden werden und die Verbindung muss gestartet werden. Dazu muss eine Instanz der** Example**-Klasse erstellt werden und die** start**-Methode aufgerufen werden. Dies stellt eine Verbindung her und startet die Kommunikation**

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

## 4.4 Nutzung der Hauptfunktionen

DieKIMaster-Schnittstelle bietet mehrere Hauptfunktionen, die für die Kommunikation mit dem WebSocket-Server verwendet werden.

* **Befehle senden:** Die Methode send\_cmd() wird verwendet, um Befehle an den WebSocket-Server zu senden. Sie nimmt den Befehl, den Befehlsschlüssel und optional zusätzliche Daten als Parameter entgegen.

Ein Bild, das Text, Screenshot, Software, Schrift enthält.

Automatisch generierte Beschreibung

* **Nachrichten empfangen**: Die Methodereceive() wird verwendet, um Nachrichten vom WebSocket-Server zu empfangen. Sie kann Nachrichten im JSON-Format, als Zeichenkette oder als Byte-Stream zurückgeben.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

* **Hauptprogramm**: Das Hauptprogramm initialisiert die Example-Klasse, startet die WebSocket-Verbindung und verwaltet die asynchronen Aufgaben für das Senden und Empfangen von Nachrichten. Dazu muss dieses ausgeführt werden.

Ein Bild, das Text, Screenshot, Schrift enthält.

Automatisch generierte Beschreibung

# 5 Häufige Probleme und Lösungen

# 6 Glossar

# 7 Index