



ulm university universität
uulm

**Fakultät für
Ingenieurwissenschaften,
Informatik und
Psychologie**

Institut für Software-
technik und Program-
miersprachen

Softwaregrundprojekt Meilenstein 5

Softwaregrundprojekt an der Universität Ulm

Vorgelegt von:

Gruppe 10

Dozent:

Florian Ege

Betreuer:

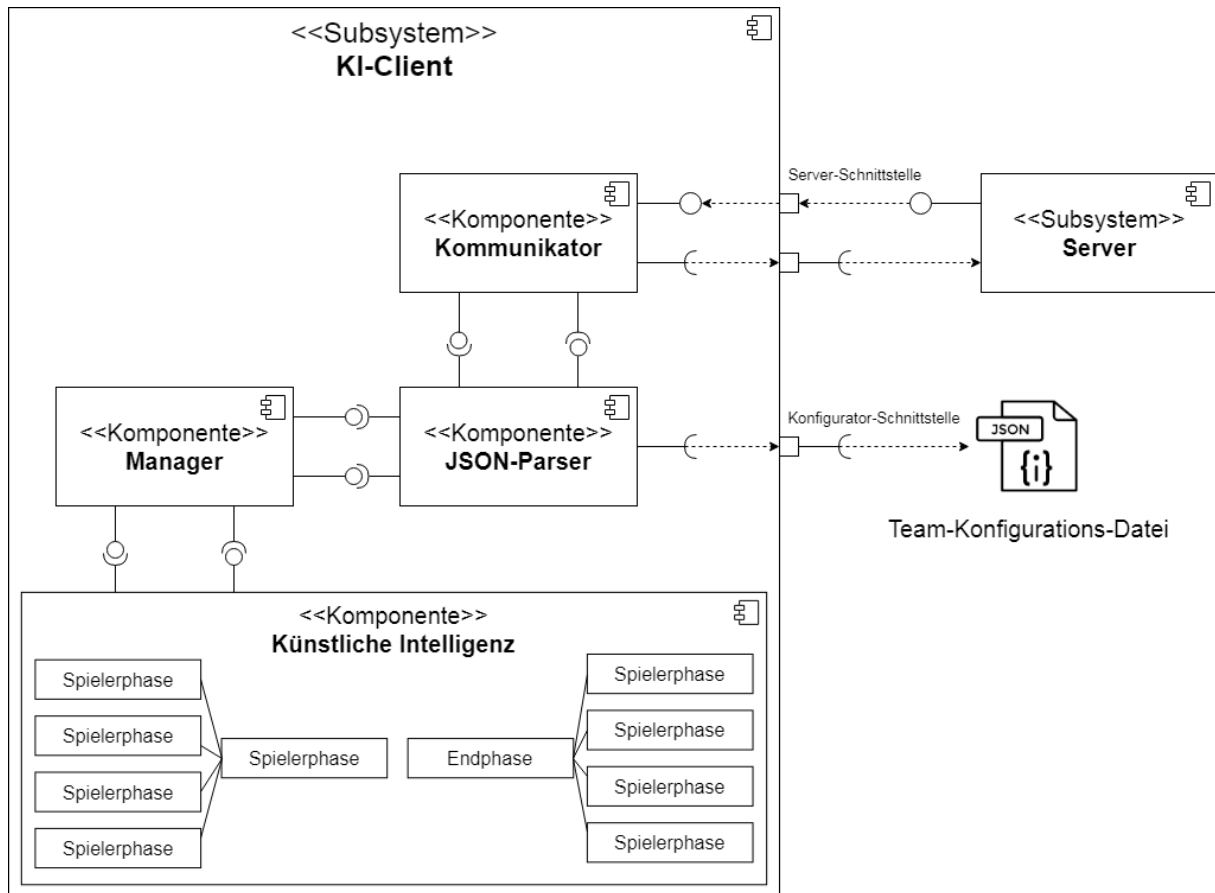
Stefanos Mytilineos

2019

Inhaltsverzeichnis

1	KI-Client	3
2	Server	7
2.1	UML2-Komponentendiagramm	7
2.2	Beschreibungen	8
2.3	Zuordnung der Funktionalen Anforderungen	9

1 KI-Client



KI-Client: Das Subsystem KI-Client ist eine Kommandozeilenanwendung, die sich wie ein Nutzer-Client mit einem Server verbindet, einer Partie beiträgt und, gesteuert von einer KI, einen menschlichen Spieler simuliert.

Manager: Diese Komponente verwaltet die Daten des KI-Client, insbesondere die aktuelle Spielsituation und verarbeitet die Anwendungsparameter.

Der Manager empfängt vom JSON-Parser die aktuelle Spielsituation und aktualisiert seine gespeicherten Daten entsprechend. Er übergibt seine Daten an die KI, damit diese Entscheidungen über die durchzuführenden Aktionen treffen kann. Anschließend empfängt er die Entscheidungen der KI und aktualisiert die gespeicherte Spielsituation entsprechend, bevor er sie dem JSON-Parser übergibt.

Zu Beginn einer Partie erhält der Manager vom JSON-Parser die Daten aus der zu verwendende Team-Konfigurationsdatei.

Der Manager ist die zentrale Komponente des Subsystems und behandelt den Programmverlauf und die Parameter, damit die KI sich ausschließlich mit dem Entscheidungen befassen kann.

JSON-Parser: Diese Komponente fungiert als Dolmetscher für die Kommunikation zwischen KI-Client, Server und Team-Konfigurator.

Der JSON-Parser empfängt die vom Kommunikator kommenden Nachrichten im JSON-Format und extrahiert daraus Daten über die aktuelle Spielsituation, die er anschließend dem Manager übergibt. Andersherum empfängt er Daten vom Manager, verpackt sie in einer Nachricht im JSON-Format und sendet sie an den Kommunikator.

Außerdem liest der JSON-Parser Team-Konfigurationsdateien, extrahiert die Daten und gibt sie an den Manager weiter.

Das Übersetzen der JSON-Dateien wird in eine eigene Komponente ausgelagert, damit der Manager unabhängig von den Konventionen der JSON-Nachrichten ist und das Subsystem leicht an diese angepasst werden kann.

Künstliche Intelligenz Diese Komponente trifft Entscheidungen über durchzuführende Aktionen anhand der aktuellen Spielsituation.

Die künstliche Intelligenz, kurz KI, empfängt Daten vom Manager und verarbeitet sie in ihrer jeweiligen Logik, um die nächste durchzuführende Aktion zu ermitteln. Sobald sie eine Entscheidung getroffen hat, teilt sie diese dem Manager zur weiteren Verarbeitung mit.

Die interne Struktur der künstlichen Intelligenz enthält eine Logik für jede Spielfiguren-Rolle und jede mögliche Einmischung, da jede Spielfigur anhand ihrer Aufgabe und Spezialisierung handeln muss.

Die KI soll dabei unabhängig von den anderen Komponenten sein, damit sie problemlos zu jedem Zeitpunkt optimiert und für unterschiedliche Schwierigkeitsstufen ausgetauscht werden kann.

Kommunikator Diese Komponente ist dafür zuständig, mit dem Server zu kommunizieren.

Der Kommunikator stellt eine Verbindung mit dem angegebenen Server her, hält sie aufrecht und versucht, sie bei Verbindungsabbruch wiederherzustellen. Er empfängt Nachrichten vom Server und gibt sie unverarbeitet an den JSON-Parser weiter und überträgt umgekehrt die vom JSON-Parser kommenden Daten an den Server.

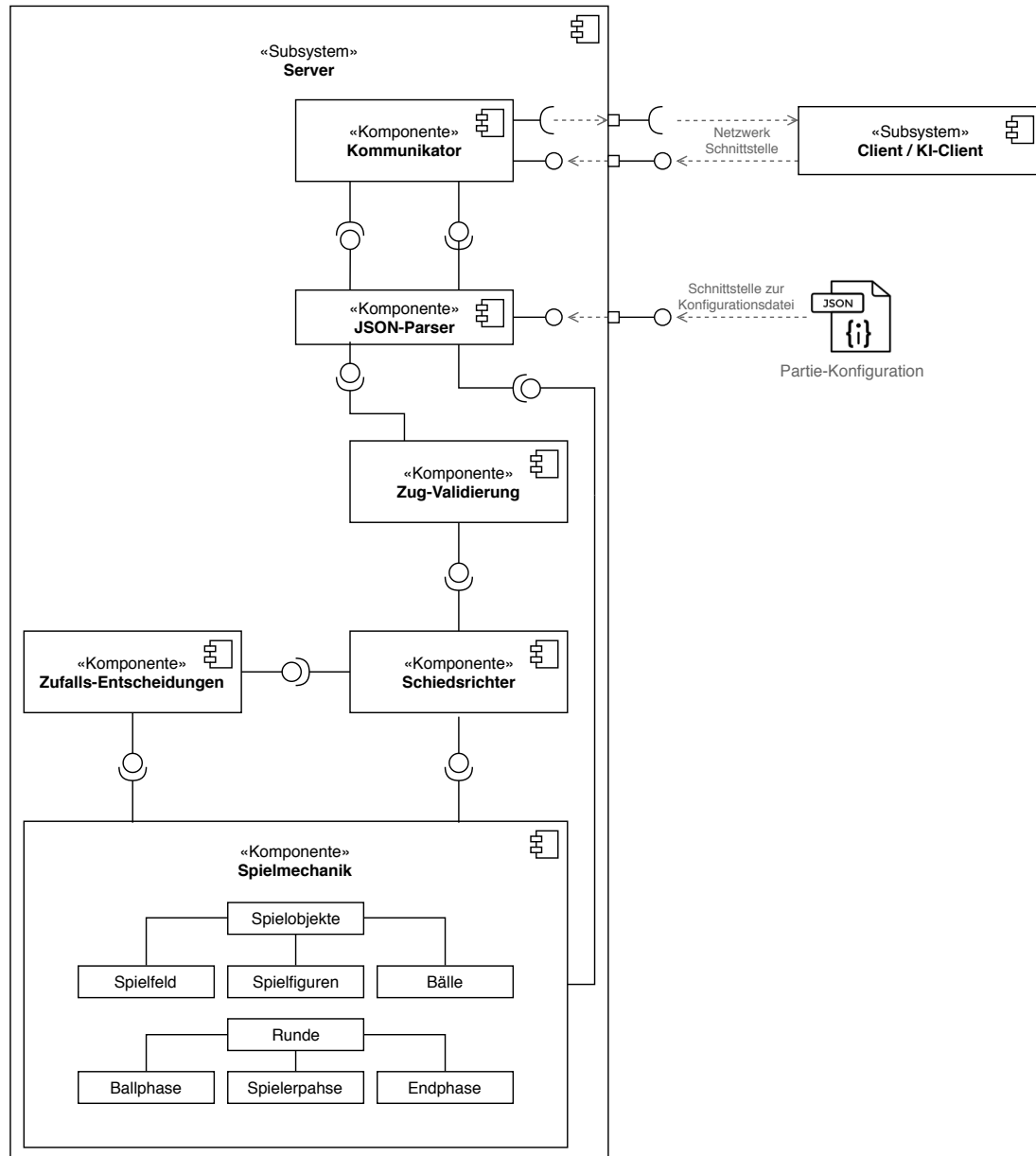
Diese Funktionen werden von einer separaten Komponente übernommen, da sie in anderen Subsystemen ebenfalls gebraucht werden und der Kommunikator dort wiederverwendet werden kann.

Die funktionalen Anforderungen gemäß dem Pflichtenheft werden den Komponenten folgendermaßen zugeteilt:

Komponente	Abgedeckte funktionale Anforderungen
Manager	FA1 - FA8 FA10 - FA20 FA44 - FA47 FA73 - FA75
JSON-Parser	FA54
Künstliche Intelligenz	FA17 - FA 21 FA26 - FA28 FA30 - FA35 FA37 - FA42 FA46 - FA47 FA50
Kommunikator	FA55

2 Server

2.1 UML2-Komponentendiagramm



2.2 Beschreibungen

Server Beim Subsystem Server handelt es sich um die Konsolenanwendung, die als Schnittstelle für zwei oder mehr Clients dient. Die Serveranwendung kümmert sich dabei primär um die Spielsteuerung und agiert dabei unabhängig von den Client-Anwendungen.

Spielmechanik Die Spielmechanik ist die zentrale Komponente des Server Subsystems. Diese bildet dabei die komplette Partie intern ab. In der Spielmechanik sind also zu jeder Zeit die aktuellsten Status der Spielobjekte hinterlegt. Während eines Spiels senden die Clients ihre gewünschten Züge an den Server. Die Spielmechanik wertet die Züge, nachdem sie validiert und vom Schiedsrichter geprüft wurden, aus und aktualisiert dann gegebenenfalls die aktuelle Spielsituation. Diese wird im Anschluss wieder vom Server aus zu den einzelnen Clients ausgegeben. Als zentrale Komponente des Server Subsystems ist es zwingend notwendig, dass diese Komponente als Einheit gesehen wird, da ohne diesen Teil kein das Spielen nicht möglich wäre.

Zug-Validierung Die Zug-Validierung prüft, ob die Züge, die von einem Spieler über seinen Client übermittelt werden, grundsätzlich möglich sind. Dabei wird jedoch nicht geprüft ob der gewünschte Zug ein Foul darstellt. Notwendig ist diese Prüfung, da nicht sichergestellt ist, dass jede Client-Anwendung tatsächlich prüft ob die Züge die ein Spieler tätigen will auch grundsätzlich möglich sind. Die Komponente ist aus der eigentlichen Spielmechanik ausgegliedert, da die Zug-Validierung auch in anderen Subsystem eingesetzt werden könnte, z.b. einer Client-Anwendung.

Schiedsrichter Die Schiedsrichter Komponente des Server Subsystems. Möchte ein Spieler einen verbotenen Zug tätigen, wird in der Schiedsrichter Komponente die Entscheidung getroffen, ob der Spieler bestraft wird. Diese Komponente ist aus der Spielmechanik ausgegliedert, da da das Spiel grundsätzlich auch ohne diese Komponente möglich ist und daher diese Komponente in einem nachgelagerten Entwicklungsschritt implementiert werden kann.

Zufalls-Entscheidungen Dies Komponente stellt der Spielmechanik und der Schiedsrichter Komponente eine Schnittstelle zur Verfügung, die es erlaubt die vielen Zufalls-Ereignisse in einer Partie auszuwerten.

Kommunikator Der Kommunikator bildet die Komponente, die sich um die Datenübertragung zwischen der Server- und den Client-Anwendungen kümmert. Dabei werden dort zum einen die Status der Verbindungen zu den Clients überwacht und verwaltet. Zum anderen wird die Datenübertragung in beide Richtungen bereitgestellt. Da auch in den Client-Anwendungen eine ähnliche Komponente von Nöten ist, bietet es sich an, diese Funktionalitäten in einer eigenen Komponente auszulagern.

JSON-Parser Der JSON-Parser (de-) serialisiert die Objekte der Spiellogik. Da auch in den Client-Anwendungen eine ähnliche Komponente von Nöten ist, bietet es sich

an, diese Funktionalitäten in einer eigenen Komponente auszulagern.

2.3 Zuordnung der Funktionalen Anforderungen

Die funktionalen Anforderungen gemäß dem Pflichtenheft werden den Komponenten folgendermaßen zugeteilt:

Komponente	Abgedeckte funktionale Anforderungen
Spielmechanik	FA1 - FA36 FA43 - FA52 FA56 FA59
Schiedsrichter	FA36 - FA42
Zug-Validierung	FA67
Zufalls-Entscheidungen	FA58
Kommunikator	FA55
JSON-Konverter	FA53 FA57