



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

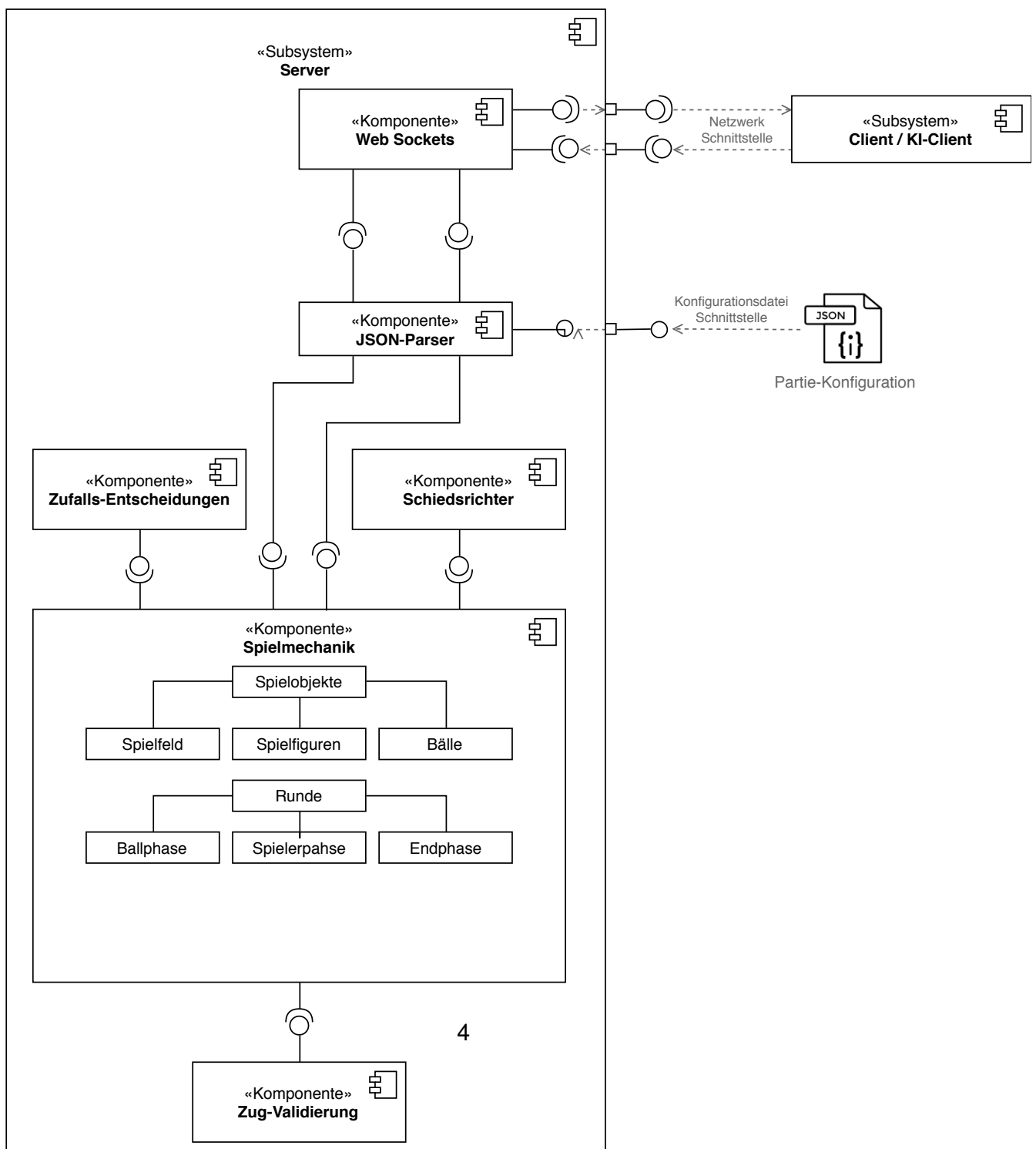
1	Server	4
1.1	UML2-Komponentendiagramm	4

Teil I

Architekturentwurf

1 Server

1.1 UML2-Komponentendiagramm



1.2 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 bildet die zentrale Komponente des Server Subsystems. Die Spielmechanik bildet dabei die komplette Partie intern ab, d.h. in der Spielmechanik ist zu jeder Zeit der aktuellste Status eines Spielobjektes hinterlegt. Während eines Spiels senden die Clients ihre gewünschten Züge an den Server. Die Spielmechanik wertet die Züge aus, lässt diese prüfen und aktualisiert dann gegebenenfalls die aktuelle Spielsituation. Die aktualisierte Spielsituation wird im Anschluss wieder vom Server aus an zu den einzelnen Clients ausgegeben. Als zentrale Komponente des Server Subsystems ist es zwingend notwendig, dass diese Komponente als Einheit gesehen wird, das ohne diesen Teil kein das Spielen nicht möglich wäre.

Schiedsrichter Die Schiedsrichter Komponente des Server Subsystem stellt der Spielmechanik keine Schnittstelle zur Verfügung. Möchte ein Spieler einen verbotenen Zug tätigen wird in der Schiedsrichter Komponente die Entscheidung getroffen, ob der Zug von der Spielmechanik trotzdem ausgeführt wird oder ob der Spieler bestraft wird. Diese Komponente ist aus der Spielmechanik ausgegliedert, da das Spiel grundsätzlich auch ohne diese Komponente möglich ist und daher diese Komponente in einem nachgelagerten Entwicklungsschritt implementiert werden kann.

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.

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

Web Sockets Die Web Sockets bilden 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 ein 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 wandelt die JSON-Daten in interne Objekte der Software um und umgekehrt. Da auch in den Client-Anwendungen eine ähnliche Komponente von Nöten ist, bietet es sich an diese Funktionalitäten in einer eigenen Komponente auszulagern.

1.3 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
Web Sockets	FA55
JSON-Konverter	FA53 FA57