

# **Architekturdokumentation**

## Indoor – Soccer – Club

erstellt von

Adam Grabowski,  
Max Kästner,  
Christina Sudareva und  
Marvin Königs

Abgabe: 20.01.2018

Hochschule Niederrhein  
Fachbereich Wirtschaftswissenschaften  
Studiengang Wirtschaftsinformatik

# Inhaltsverzeichnis

1. Einführung .....	1
2. Architekturtreiber .....	1
3. Einflussfaktoren .....	2
4. Sichten .....	2
4.1 Kontextsicht .....	2
4.2 Bausteinsicht .....	3
4.3 Laufzeitsicht .....	3
4.4 Verteilungssicht .....	4
5. Architekturaspekte .....	4
5.1 Persistenz .....	4
5.2 Ausnahme- und Fehlerbehandlung .....	4
5.3 Transaktionsbehandlung .....	4
6. Referenzen .....	5

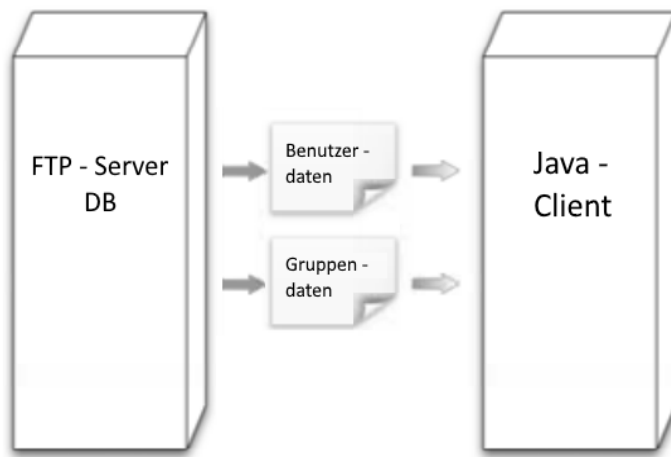
## 1. Einführung

Diese Dokumentation beschreibt die Architektur der Software – Client - Anwendung gemäß den methodischen und strukturellen Ansätzen der in den Vorlesung besprochenen Struktur.

Zweck des Systems:

Die Software – Client – Anwendung auf Basis eines Java – Clients und einem FTP – Server soll die Planung und Steuerung für Fußballspiele ermöglichen.

Die Erstellung von Gruppen und dem Suchen von Spielen in vor eingetragenen Hallen ermöglicht im Benutzer - Client eine Planung der zu spielenden Spiele.



## 2. Architekturtreiber

- Bestehende Spielerdaten sollen von dem System erfolgreich eingelesen und im Benutzer-Client dargestellt werden.
- Bestehende Spielerdaten sollen bearbeitbar sein. Anschließend müssen die Daten auf den Server hochgeladen werden.
- Erstellte Gruppen sollen auf dem FTP – Server gespeichert werden. Aus diesen sollen die Spiele geplant werden.
- Erstellte Hallen sollen bei einer Spielersuche von dem Client eingelesen, analysiert und ausgegeben werden.
- Gefilterte Hallen sollen in der Übersicht angezeigt werden. Der Benutzer soll hier eine Halle aussuchen können.

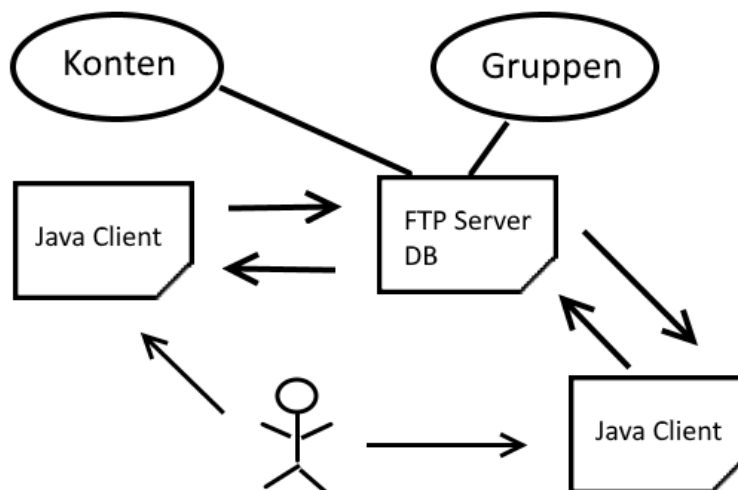
### 3. Einflussfaktoren

Hardware-Infrastruktur	FTP – Server von bPlaced, Windows oder Linux - Computer
Software-Infrastruktur	Windows – FTP – Server, Windows oder Linux – Betriebssystem
Systembetrieb	Jar - Format
Grafische Oberfläche	Javax.swing
Bibliotheken, Frameworks und Komponenten	Beliebige Java – Bibliotheken (Hauptsächlich Swing, AWT, FTP)
Programmiersprache	Java
Referenzarchitektur	Server-Client – Anwendung
Programmierschnittstellen	Jede Klasse hat seine eigene Funktion
Technische Kommunikation	Mit Host per ftp

### 4. Sichten

#### 4.1 Kontextsicht

Hier wird das System als eine „Blackbox“ gesehen. Zu sehen ist der Server – Client – Anwendung mit folgenden Schnittstellen.

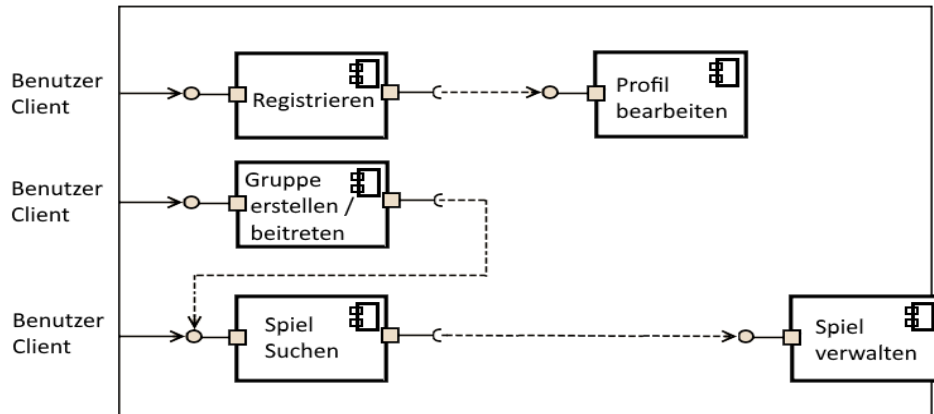


Der FTP – Server speichert alle Daten über die Konten und die Gruppen der Spieler. Die Client – Anwendungen greifen auf diese Daten zu und können diese Verwalten (Speichern, Ändern, Löschen).

## 4.2 Bausteinsicht

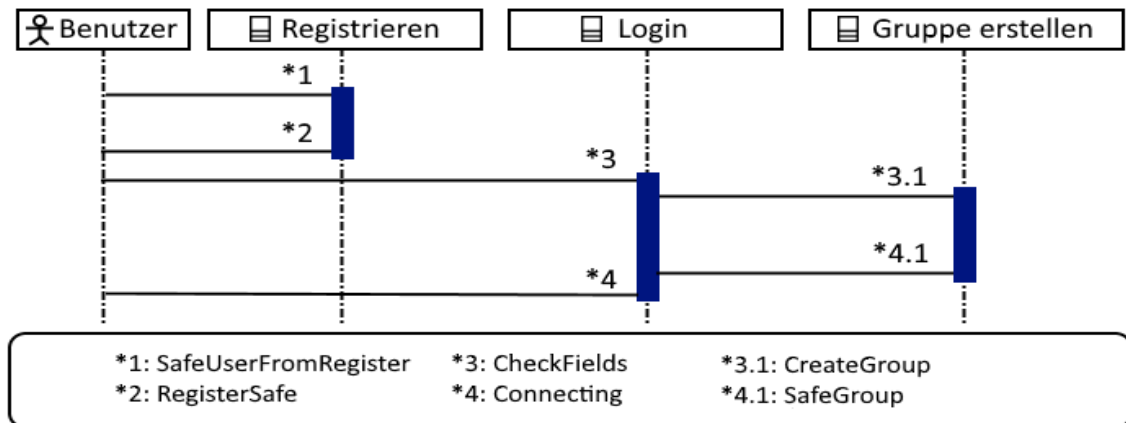
Alle Bausteine sind in einem gemeinsamen System zusammen gebaut.

Subsysteme hat die Anwendung nicht. Hier findet nur der Client (siehe Abbildung) und der Server eine Funktionalität.

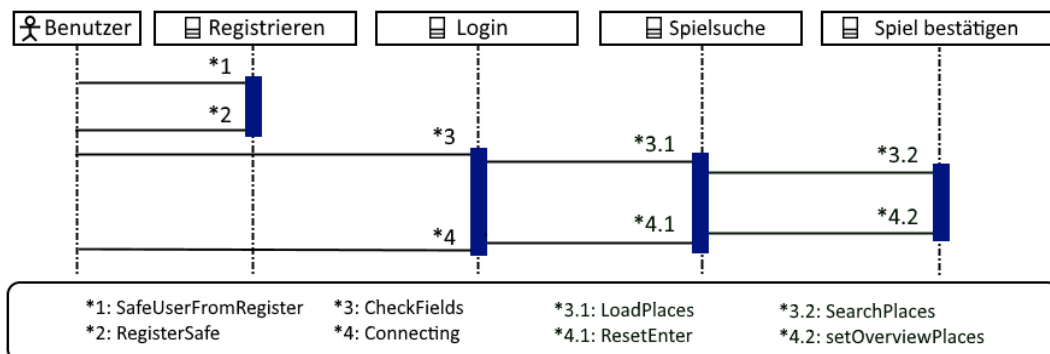


## 4.3 Laufzeitsicht

Die Funktion „Gruppe erstellen“ ist in der Laufzeitsicht wie folgt zu sehen:

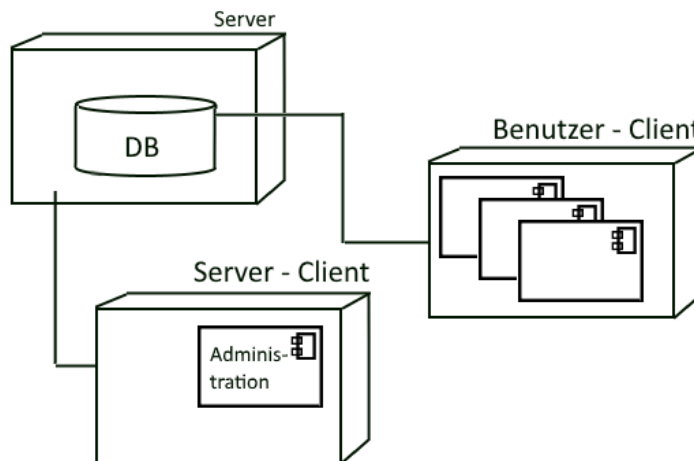


Die Funktion „Spiel suchen“ unterscheidet sich hier nur bedingt von der Funktion „Gruppe erstellen“. Einige Funktionen, wie das einmalige Registrieren und das Einloggen auf dem Benutzer-Client bleibt dabei zwingend erforderlich.



## 4.4 Verteilungssicht

---



## 5. Architekturaspekte

### 5.1 Persistenz

---

Die Daten in der Anwendung werden in soweit Persistent bereit gestellt, dass sie bei einer Änderung gespeichert werden und hochgeladen werden. Nach einem Absturz des Clients gehen so keine Daten verloren. Auch das kurzzeitige Abschalten des Server führt zu keinem Datenverlust.

So muss jede Änderung im Client gespeichert werden. Bei einer Error - Meldung können Daten entweder über den Benutzer – Client (normal - Anwender) oder über den Server – Client (Administrator, mit vollem Zugriff) verwaltet werden.

### 5.2 Ausnahme- und Fehlerbehandlung

---

Wie in der Persistenz beschrieben sind Daten über den Benutzer – Client oder über den Server – Client zu verwalten und fehlerhafte Daten zu korrigieren.

Die Korrektur soll allerdings nur bedingt vorkommen, da falsche Daten sofort von den Anwendungen erkannt werden sollen.

### 5.3 Transaktionsbehandlung

---

Die Daten werden über den Client je nach Bedarf im Sekundentakt aktualisiert.

Auch eine manuellen Aktualisierung ist (z.B. durch den Aufruf des Benutzerprofils) möglich.

So sind die Daten immer und von jedem Standort aus Korrekt und aktuell.

## 6. Referenzen

GitHub: <https://github.com/X20000N/Indoor-Soccer-Club>

Entwickler: Adam, Max, Christina und Marvin