

XFall Arbeitsgruppe

# Testserver

Konzept

- ENTWURF -

Stand 2019-08-13

© 2019 FJD Information Technologies AG. All rights reserved.



## Historie

Datum	Bearbeiter	Änderung
2019-08-13	Andreas Huber, FJD	Dokument erstellt

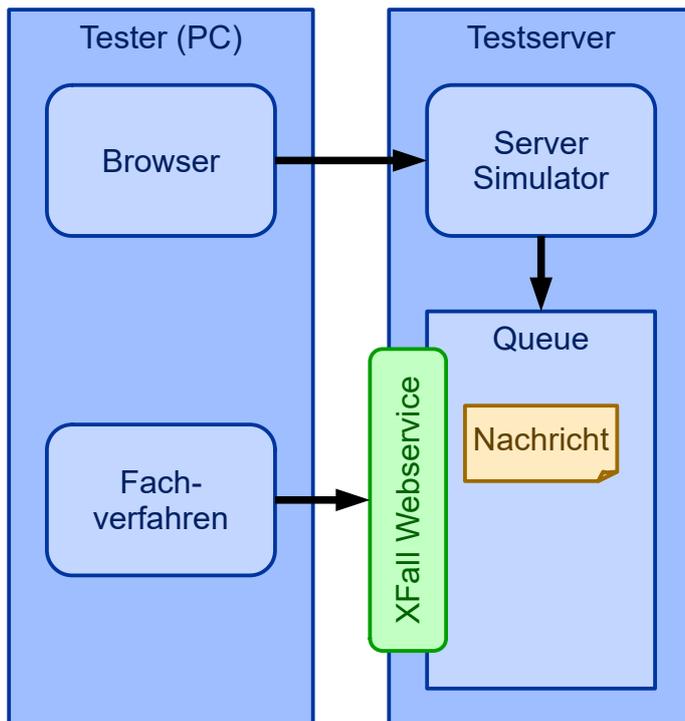
# Inhaltsverzeichnis

1 Konzeption.....	4
1.1 Architektur.....	4
1.1.1 Server Simulator.....	4
1.1.2 Client Simulator.....	4
1.2 Module.....	5
1.3 Funktionalitäten.....	5
1.3.1 Server Simulator – Oberfläche.....	5
1.3.2 Server Simulator – Webservice.....	5
1.3.3 Client Simulator.....	5
1.4 Datenhaltung.....	6
1.5 Löschkonzept.....	6
1.6 Schnittstellen.....	6
1.6.1 Weboberfläche.....	6
1.6.2 Webservice (Server).....	6
1.6.3 Webservice (Client).....	6
1.7 Sicherheit.....	6
1.8 Datenschutz.....	6
2 Umsetzung.....	7
2.1 Vorbedingungen.....	7
2.2 Meilensteine.....	7
2.2.1 Level 1.....	7
2.2.2 Level 2.....	7
2.2.3 Level 3.....	7
3 Anhang.....	8
3.1 Abkürzungsverzeichnis.....	8
3.2 Begriffsdefinitionen.....	8
3.3 Offene Punkte.....	9

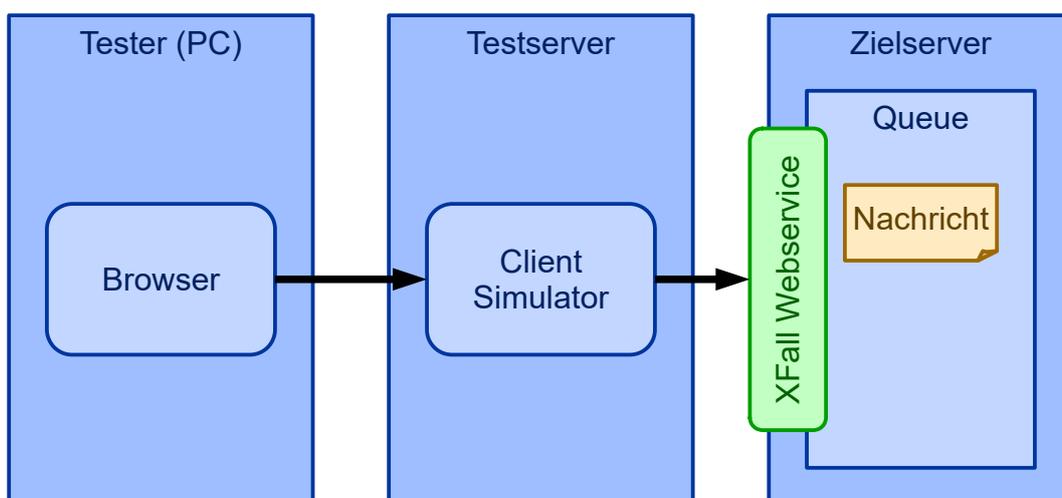
# 1 Konzeption

## 1.1 Architektur

### 1.1.1 Server Simulator



### 1.1.2 Client Simulator



## 1.2 Module

Server-Simulator Oberfläche

Server-Simulator Webservice

Client-Simulator Oberfläche

## 1.3 Funktionalitäten

### 1.3.1 Server Simulator – Oberfläche

- Konfiguration
  - Authentifikation einstellen (Art, Token, Header, Login, Passwort, Zertifikat)
  - Webservice-URL abrufen
- Queue einsehen
- Nachricht in die Queue legen
- Nachricht in der Queue einsehen
- Nachricht aus der Queue entfernen

### 1.3.2 Server Simulator – Webservice

Level 1

- Authentifikation
- Abfrage der Queue (Funktion „queue“)
- Nachricht abholen (Funktion „receive“)
- Abholung der Nachricht bestätigen (Funktion „confirm“)
- Datei abholen (Funktion „receiveFile“)

Level 2

- Versenden von Nachrichten ohne Transaktion (Funktion „send“)

Level 3

- Transaktion anlegen (Funktion „openTransaction“)
- Datei hochladen (Funktion „sendFile“)
- Transaktion commiten (Funktion „commitTransaction“)

### 1.3.3 Client Simulator

Client-Simulator Oberfläche

Level 1

- Konfiguration
  - URL

- Authentifikation
- Login & Passwort
- HTTP-Header & Token
- Client-Zertifikat
- Abfrage der Queue (Funktion „queue“)
- Nachricht abholen (Funktion „receive“)
- Abholung der Nachricht bestätigen (Funktion „confirm“)
- Datei abholen (Funktion „receiveFile“)

#### Level 2

- Versenden von Nachrichten ohne Transaktion (Funktion „send“)

#### Level 3

- Transaktion anlegen (Funktion „openTransaction“)
- Datei hochladen (Funktion „sendFile“)
- Transaktion commiten (Funktion „commitTransaction“)

## 1.4 Datenhaltung

Die Konfiguration und beim Server die Queue müssen für eine kurze Zeit (~1 Tag) gespeichert werden.

## 1.5 Löschkonzept

Da in der Queue Nachrichten liegen bleiben könnten, sollten diese nach einer Weile (3 Tage?) automatisch entfernt werden.

## 1.6 Schnittstellen

### 1.6.1 Weboberfläche

### 1.6.2 Webservice (Server)

### 1.6.3 Webservice (Client)

## 1.7 Sicherheit

Sicherheitstechnische Bewertung

## 1.8 Datenschutz

- nicht relevant da nur Testdaten -

## 2 Umsetzung

### 2.1 Vorbedingungen

### 2.2 Meilensteine

#### 2.2.1 Level 1

Minimaler Satz an Funktionen zum Abholen von Anträgen

#### 2.2.2 Level 2

Zusätzliche Funktionen zur bidirektionalen Kommunikation (Antworten)

#### 2.2.3 Level 3

Transaktionen

## 3 Anhang

### 3.1 Abkürzungsverzeichnis

Abkürzung	Langbezeichnung
AMS	Antragsmanagementsystem
AS	Antragsteller
GovOS	Governmental Operating System
GovAMS	GovOS Antragsmanagementsystem
ZS	Zuständige Stelle

### 3.2 Begriffsdefinitionen

**Hinweis:** Aus Gründen der bessern Lesbarkeit werden in diesem Dokument die männliche Form verwendet. Es ist jedoch immer, wenn nicht anders vermerkt, eine Person jeglichen Geschlechts gemeint.

Begriff	Erläuterung
Antragsteller	Der Antragsteller nutzt das GovAMS, um eine von einem Mandanten angebotene Leistung nachzufragen bzw. zu beantragen. Wird hier synonym zu „Bürger“ benutzt.
App	Die einzelnen Vorgänge in GovOS werden in „Apps“ gekapselt. Diese sind vergleichbar mit den Apps auf einem Smartphone, nur dass sie auf einem Server im Internet laufen und über einen Browser statt einem Handy zugänglich sind.
Bürger	Der Bürger nutzt das GovAMS, um eine von einem Mandanten angebotene Leistung nachzufragen bzw. zu beantragen. Wird hier synonym zu „Antragsteller“ benutzt.
Kunde	Der Kunde ist der Vertragspartner oder Auftraggeber der FJD.
Mandant	Als Mandant wird die Behörde bezeichnet, die GovOS bzw. GovAMS nutzt, um behördliche Leistungen anzubieten. Wird hier synonym zu „Zuständige Stelle“ benutzt.
Zuständige Stelle	Als Zuständige Stelle wird die Behörde bezeichnet, die GovOS bzw. GovAMS nutzt, um behördliche Leistungen anzubieten. Wird hier synonym zu „Mandant“ benutzt.

### 3.3 Offene Punkte

- Soll der Push-Modus auch abgedeckt werden? In diesem Fall bräuchten wir einen dritten Simulator.
- Welche Authentifikationen bieten wir an?
  - Token in der URL
  - Token im HTTP-Header
  - Basic Authentication (Spezialfall von Token im HTTP-Header)
  - Client Zertifikat
- Mit welcher Technik wird der Testserver realisiert?
  - Java oder eher PHP oder Node.js?
- Wie wird die Weboberfläche realisiert?
  - Am besten ein JavaScript-Framework