



## **Übungen zur Vorlesung Objektorientierte Komponenten-Architekturen**

**SS 2023**

### **Übungsblatt Nr. 6**

(Abgabe: Bearbeitung 1 und 2 bis Montag, 19.6.2023)

#### **Anmerkung:**

Nutzen sie die nächsten Wochen, um an ihren Semesterprojekten weiterzuarbeiten. Dazu stelle ich ihnen zwei Aufgaben bereit, die sie als Basis gerne verwenden können. Eine Pflichtabgabe ist *nicht mehr* notwendig.

#### **Aufgabe 1 (Weiter-Entwicklung der Software für das Unternehmen WirSchiffenDas; Bearbeitung bis: 19.6.2023)**

Die Geschäftsleitung der Firma WirSchiffenDas GmbH ist angetan von den neuen Technologien aus dem vierten und fünften Kapitel der Vorlesung, d.h., Docker, Apache Kafka und den Spring-Netflix-Stack! Es soll ihre Aufgabe sein, *mindestens* einer dieser Technologien sinnvoll in ihre Anwendung für die Fallstudie der Firma zu integrieren. Konkret sollten sie *mindestens eine* der folgenden technischen Anforderungen umsetzen:

- **MS\_TA1:** Verwendung des Framework Apache Kafka für eine Integration der Microservices zur Übermittlung *insbesondere* der „Bearbeitungs-Status-Nachrichten“.
- **MS\_TA2:** Verwendung der Technologien Docker und Docker Compose zur Bereitstellung und Komposition der Microservices in einsetzbare Docker Container.
- **MS\_TA3:** Verwendung des Spring-Netflix-Stacks, um wichtige Pattern wie Service Registry oder Circuit Breaker *sinnvoll* zu integrieren
- **MS\_TA4:** Integration der Plattform KNative zur Realisierung einer skalierbaren Funktion

Erweitern sie ihren bisherigen Prototypen entsprechend. Modellieren sie zudem die resultierende Software-Architektur mit Hilfe des 4-Sichten-Modells! Verwenden sie auch eine Farben-Legende, um Anpassungen oder Erweiterungen an ihrer Architektur gegenüber der Architektur aus der Übung Nr. 5 (siehe Hinweise in Kapitel 3, Abschnitt 2) hervorzuheben. Auch die Quelle von verwendeten Technologien (z.B. COTS, OSS) können sie optional mit einer Farben-Legende versehen. Bei den Technologien aus Kapitel 5 ggf. vorarbeiten.

Betrachten sie auch noch mal den Katalog nach (Schirgi and Brenner, 2021). Priorisieren sie noch mal die Lösungen für Anti Pattern bzw. Architecture Smells, die sie für ihr Konzept bzw. für ihren Prototyp als relevant ansehen. In einer möglichen Präsentation sollten sie die fünf höchst priorisierten Lösungen gemäß ihrer Bewertung vorführen und erläutern können. Es sollten für das finale Semesterprojekt *circa* drei Lösungen in ihr Konzept bzw. in den Prototyp einfließen.

Hinweise für die Verwendung von Kafka: Verwenden sie die Installation aus SEPP v 2.1:

<https://saschaalda.wordpress.com/sepp-software-engineering-deployment-pipeline-hbrs/>

Verwenden sie als Topic eine möglichst individuelle Bezeichnung, um mögliche Kollisionen zu vermeiden, z.B. `mytopic_alda`.

Die in der Vorlesung bzw. in dem Kapitel 5 gezeigten Source Codes finden sie unter:

Kafka-Demo (kein SpringBoot)

<https://git.fslab.de/salda2m/kafkademo>

Stream-Application:

<https://git.fslab.de/salda2m/kafkastreamapp>

**Aufgabe 2 (Erste Vorarbeiten für die Fallstudie „TheConfSystem“; Bearbeitung bis: 19.6.2023; eine Einführung zu dem Thema Java EE und Migration kommt am 19.6.2023, eine erste Einarbeitung über dieses Projekt ist denkbar)**

Das Start-Up Unternehmen YourConf KG aus Bonn hat sich zum Ziel gemacht, ein agiles und skalierbares Konferenzmanagement zu entwickeln, welches zuverlässig die Organisation und Durchführung auch von großen Konferenzen mit mehr als 5000 Teilnehmern (vgl. UN-Konferenzen in Bonn) ermöglicht.

Das Unternehmen hat als Grundlage für ein erstes Produkt eine Anwendung eines mittelständigen Unternehmens erworben, auf die man in Zukunft aufbauen möchte. Man ist sich bewusst, dass diese Anwendung auf eher suboptimalen Entwurfsentscheidungen aufbaut, man will aber auch nicht vollständig auf der grünen Wiese anfangen. Der komplette Source Code der Anwendung steht dem Start-Up zur Verfügung. Eine Teil-Lösung der Anwendung ist die User-seitige Bewertung von Firmen (z.B. Aussteller, Workshop-Organisatoren, Sponsoren) durch Reports, die Anwendung „RatingMgmt“. Der Source Code dieser Teillösung stellt das Unternehmen ihnen für eine erste Betrachtung bzw. Analyse bereit:

<https://git.fslab.de/rodene2s/masterproject>

In der `readme.md` des Repositorys finden sie einige grundlegende Informationen z.B. für den Aufbau der Anwendung sowie Angaben für eine mögliche Installation. Das Unternehmen bittet sie um Hilfe, um diese Teil-Lösung zu verstehen, Defizite herauszuarbeiten und um eine mögliche erste Ziel-Architektur zu entwerfen.

Ihre Aufgaben:

Analysieren sie den Source Code der Anwendung. Was sind die wichtigsten konzeptionellen und technischen Entwurfsentscheidungen, die hier (absichtlich oder unabsichtlich) getroffen wurden?

Modellieren sie die Software-Architektur der IST-Anwendung, also der bestehenden Anwendung. Fokussieren sie sich dabei auf die Baustein-Sicht sowie auf eine Verteilungs-sicht, die sie aber auch integrieren können (vgl. Kapitel 3, Folie 31).

Modellieren sie auch eine mögliche Software-Architektur einer zukünftigen ZIEL-Anwendung, nachdem die Migration vollständig abgeschlossen wurde.

Skizzieren sie auch einen *ersten möglichen* Prozess für die schrittweise (inkrementelle) Migration der IST-Anwendung auf die ZIEL-Anwendung. Die Beschreibung des Prozesses (textuell, UML-basiert, Skizzen-basiert) obliegt ihrer Entscheidung.

Als erste Orientierung für das Daten-Management hat man ihnen noch ein konzeptuelles Datenmodell mitgeliefert, welches sie als erste Orientierung verwenden können.

