



**FH**

University of  
Applied Sciences

**TECHNIKUM**

**WIEN**

# Softwarekomponentensysteme

Einführung

Bernhard Löwenstein

# Inhalte

- Entwicklung von Microservices mittels Spring

# Technologien

- Java (wahlweise: Kotlin)
- Maven (wahlweise: Gradle)
- Java Persistence API
- Spring Boot
- Config Server, Eureka, Ribbon, Zuul
- Kafka (wahlweise: RabbitMQ)
- MySQL (wahlweise: beliebiges relationales Datenbank-system)

# Übungsablauf

- Teams zu jeweils max. 2 Studierenden
- Abgabegespräch mit jedem Team am Semesterende
- alle Sourcen in englischer Sprache verfassen

# Übungsaufgabe (1)

- Zur Belebung des österreichischen Tourismus soll ein Reiseportal erstellt werden, auf dem patriotische Autoren reißerische Blogbeiträge zu vorgegebenen österreichischen Sehenswürdigkeiten veröffentlichen können. Diese Artikel sollen Lust auf einen Urlaub in der Alpenrepublik machen.
- Ein Bonussystem soll zum Schreiben von Blogbeiträgen motivieren. Konkret soll 1 Cent pro Artikelaufruf ausgeschüttet werden. Die Auszahlung der Provision an die Autoren erfolgt monatlich.
- Anhand der statistisch gesammelten Daten soll man feststellen können, welche rot-weiß-roten Sehenswürdigkeiten bei den Lesern gerade besonders beliebt sind.
- Es ist die Umsetzung in Form einer Microservice-Architektur erwünscht. Zur Diskussion mit den Stackholdern soll es ein zweckmäßiges UML-Diagramm geben.

# Übungsaufgabe (2)

- Punkteverteilung:
  - UML-Diagramme 10%
  - Frontend 20%
  - Backend 70 %
- Hinweise zum Frontend:
  - Es soll eine Seite zur Eingabe eines Blogbeitrags durch einen Autor geben. Jedem Artikel wird dabei eine konkrete Sehenswürdigkeit zugeordnet.
  - Zum Lesen von Artikeln soll es im klassischen Zeitungsstile eine Übersichtsseite mit den Headlines und eine Detailseite, auf der ein konkreter Artikel dargestellt wird, geben.
  - Es soll eine Seite geben, auf der man sich die Gesamtstatistik sowie Monatsstatistiken ansehen kann.
  - Auf die Validierung der eingegebenen Daten und das Styling der Seiten kann vollständig verzichtet werden.
- Hinweise zum Backend:
  - Die Sehenswürdigkeiten und Autoren sind vorab in der Datenbank anzulegen. Die Verwaltung dieser Daten ist nicht Teil der Übungsaufgabe.
  - Die Auszahlung der Provision an die Autoren ist nur fiktiv durchzuführen. Es reicht hierzu die Ausgabe einer entsprechenden Textzeile auf der Standardkonsole.
  - Es sollen alle vorgestellten [Technologien](#) sinnvoll zum Einsatz kommen.
  - Die Tabellen unterschiedlicher Microservices können in einer gemeinsamen Datenbank gespeichert werden.
  - Es soll wirklich nur jene Funktionalität umgesetzt werden, die tatsächlich benötigt wird.
- Hinweise zum Abgabegespräch:
  - Mögliche Punkteabzüge können durch die anschließende Korrektur und das Nachbringen der Änderungen innerhalb von einer Woche kompensiert werden.
  - Wenn ein Team seine Sourcen nicht erklären kann, wird die Abgabe automatisch negativ bewertet.

# Leistungsbeurteilung

1 = Sehr gut	$\geq 88 \%$
2 = Gut	$\geq 75 \%$
3 = Befriedigend	$\geq 63 \%$
4 = Genügend	$\geq 50 \%$
5 = Nicht genügend	$< 50 \%$