SCAD P06 – Modern Microservice Development

This is a one-week lab, i.e. SW 8, submission before SW 9 lecture.

1. Task Overview

Use a programming language or (micro-)framework specifically designed for microservice development to build a microservice-based application, consisting of at least three services. Each microservice should be made discoverable by an appropriate service description written in an SDL, and/or perform messaging with code generated based on an IDL. The registration of the service description into a service registry is optional. No stateful processing is required, but the microservices should communicate with each other. The pattern of communication can be chosen freely (e.g. sequence or fan-out or load balancing or anything else). The packaging of the microservices (e.g. in containers) is also optional.

Note that from a terminology perspective, microservice-based does not equal cloud-native. There will not be any necessary cloud deployment in this lab, although of course some of the frameworks provide decent cloud integration and exploitation, and most cloud-native applications follow a microservices style.

2. Frameworks

You are free to choose a framework/language combination if it is part of the list below, or if the lecturer gives you the green light on your own suggestion.

- Ballerina - https://ballerina.io/
- Jolie - https://www.jolie-lang.org/
- Go-Micro Framework - https://go-micro.dev/
- Quarkus - https://quarkus.io/

Get into the languages by going through the respective tutorials. Copy and paste from the tutorials or sample code is allowed. Do not aim at a complex functionality. The goal is to understand how these languages and frameworks are better suited for microservice development than conventional counterparts.

For the service description and discovery, OpenAPI or RAML are the recommended SDL options. You can choose if you want to separate the modelling from the development or use top-down/bottom-up coupling of service description (discoverable interface) and service implementation.

3. Submission

Submission items - what to put into the Git repository

- Code for the three microservices
- Declarative service descriptions (SDLs) and/or messaging interface descriptions (IDLs)
- README with brief explanation of language/framework choices, intended application functionality, how to get the app started, and feedback on suitability of programming language/framework in contrast to already known counterparts