Fachhochschule Kiel
University of Applied Sciences

Data Economy – Data Platforms

Following is **Part 1** of the **Portfolio Examination** of the module **Data Economy** in **Summer Semester 2024**. Please **upload** your **documents** and **program files** to the corresponding course in **Moodle** Learn **latest until 17.05.2024 at 23:59**.

FIWARE Technology

Please prepare a presentation about the IT Architecture of a Data Platform using FIWARE Technology for parking and weather data and install, configure, and run the corresponding FIWARE components and prepare the Python programs due to the following guidelines:

1. Install, configure, and run the Docker images of the FIWARE Context Broker and the MongoDB database on your laptop.

2. Install, configure, and run the Docker images of FIWARE QuantumLeap and the CrateDB database on your laptop.

3. Create a Python program project in VS Code that will be used to simulate the status of parking spots and define a Dockerfile to generate a Docker image.

4. Create a Python program project in VS Code that will be used to collect weather data from an open web service and define a Dockerfile to generate a Docker image.

5. Create a Python program project in VS Code that will be used to visualize the status of the parking spots and the weather as real-time data from the Context Broker as well as time series data from QuantumLeap.

6. Prepare a (PowerPoint) presentation about the IT architecture that contains at least information about the installation and configuration of the FIWARE components as well as the logical data flow between the different programs in the data platform.

The **deliveries** are the presentation (as a PDF file).

Context Broker, Smart Data Models, and NGSI-V2

Please develop and run the Python programs to simulate the status of parking spots, to collect weather data, and to visualize the real-time data as well as the time series data using the Context Broker, QuantumLeap, the Smart Data Models, and NGSI-V2 due to the following guidelines:

1. Develop a Python program that creates the data model for an off-street parking space with 10 parking spots using the corresponding Smart Data Models and NGSI-V2. At runtime the program should randomly change the status of the different parking spots from free to occupied and vice versa in the Context Broker using NGSI-V2. The program should run as a Docker container.

2. Develop a Python program that creates the data model for weather data using the corresponding Smart Data Models and NGSI-V2. At runtime the program should collect

weather data from an open web service (e.g., Open-Meteo) and update the weather data for the corresponding location in the Context Broker using NGSI-V2. The program should run as a Docker container.

3. Configure the Context Broker and QuantumLeap to use subscriptions for all created data models so that new real-time data will be automatically stored in QuantumLeap as time series data.

4. Develop a Python program that retrieves real-time data from the Context Broker and time series data from QuantumLeap of the parking spots and the weather using NGSI-V2. The real-time data as well as the time series data should be visualized in a useful manner. The program should run as a Docker container.

5. Prepare a (Word) documentation that contains at least information about the setup to run the Docker containers of the Python programs, the structure and configuration of the programs, and some sample output.

The **deliveries** are the documentation (as a PDF file) as well as the Python program file(s) or Jupyter notebook(s) including the Dockerfile(s).