API documentation

The purpose of this Document is to enumerate the APIs which are exposed by Akraino Blue print project to the external projects Akraino/Non Akraino for interaction/integration.

This document should be used in conjunction with the architecture document to understand APIs at modular level and their interactions.

This document should function as a glossary of APIs with its functionality, interfaces, inputs and expected outcomes as the following example:

API 1: Kubernetes/k3 API

MicroMEC deploys k3, so the normal k3s interface is provided.

See https://rancher.com/docs/k3s/latest/en/ for reference.

API 2: OpenFAAS

MicroMEC also deploys OpenFAAS. For that, we again refer to official documentation in https://docs.openfaas.com/.

API 3: ETSI MEC11

Best overall reference is the ETSI MEC 11 documentation https://www.etsi.org/deliver/etsi_gs/MEC/001_099/011/01.01.01_60/gs_MEC011v010101p.pdf

The following interfaces have been implemented:

DNS rules

- /mp1/v1/applications/{appInstanceId}/dns_rules
- /mp1/v1/applications/{appInstanceId}/dns_rules/{dnsRuleID}

Traffic rules will fail

- /mp1/v1/applications/{appInstanceId}/traffic_rules
- /mp1/v1/applications/{appInstanceId}/traffic_rules/{traffic_rule_id}

Other

- /mp1/v1/applications/{appInstanceId}/dnsrules/{dnsRuleId}
- /mp1/v1/services
- /mp1/v1/timing/current_time

API 4: Available sensors

New APIs can be added to the platform for e.g. accessing different sensors. This, according to the ETSI MEC philosophy, is done with OpenAPI. One example was done in the Metropolia Innovation project during fall 2019 for accessing camera images. The code is available in https://github.com/Metropolia-Innovation-Project-2019-H2/uMEC-ETWS--API/tree/master/openapi.

The file `openapi.yaml` defines the interface. The OpenAPI application then generates the sceleton code for client and server code for the selected programming language, Python in this case. The functionality is implemented in `openapi/openapi_server/controllers/default_controller.py`.

API 5: Developer interface

There is a web UI for adding new applications so that both the MEC Service Registry and k3s are updated consistently.