SovereignEDGE.EU COGNIT

A Cognitive Serverless Framework for the Cloud-Edge Continuum

D5.12 COGNIT Framework - Demo - c

Version 2.0

1 March 2026

Abstract

COGNIT is an AI-Enabled Adaptive Serverless Framework for the Cognitive Cloud-Edge Continuum that enables the seamless, transparent, and trustworthy integration of data processing resources from providers and on-premises data centers in the cloud-edge continuum, and their automatic and intelligent adaptation to optimise where and how data is processed according to application requirements, changes in application demands and behaviour, and the operation of the infrastructure in terms of the main environmental sustainability metrics. This document provides an overview of the final demonstrations belonging to the four project Use Cases, demonstrating many of the capabilities of the COGNIT Framework. The demonstrations have been run using both the shared COGNIT testbed hosted by RISE and on local UC clusters.

Copyright © 2025 SovereignEdge.Cognit. All rights reserved.

This project is funded by the European Union's Horizon Europe research and innovation programme under Grant Agreement 101092711 – SovereignEdge.Cognit

This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

Deliverable Metadata

Project Title:	A Cognitive Serverless Framework for the Cloud-Edge Continuum
Project Acronym:	SovereignEdge.Cognit
Call:	HORIZON-CL4-2022-DATA-01-02
Grant Agreement:	101092711
WP number and Title:	WP5. Adaptive Serverless Framework Integration and Validation
Nature:	DEM: Demonstrator, Pilot, Prototype
Dissemination Level:	PU: Public
Version:	2.0
Contractual Date of Delivery:	30/09/2025
Actual Date of Delivery:	01/03/2026
Lead Author:	Thomas Ohlson Timoudas (RISE), Joel Höglund (RISE)
Authors:	Daniel Clavijo (OpenNebula), Marco Mancini (OpenNebula), Alberto P. Martí (OpenNebula), Idoia de la Iglesia (Ikerlan), Fátima Fernández (Ikerlan), Constantino Vázquez (OpenNebula), Pavel Czerny (OpenNebula), Francesco Renzi (Nature 4.0 s.r.l.), Filippo Tagliacarne (Nature 4.0 s.r.l.), Agnieszka Frąc (Atende Industries), Mateusz Kobak (Phoenix Systems)., Malik Bouhou (CETIC), Nikolaos Matskanis (CETIC), Antonio Lalaguna(ACISA), Carlos López(ACISA).
Status:	Submitted

Document History

Version	Issue Date	Status[1]	Content and changes
0.1	19/12/2025	Draft	Initial Draft
0.2	22/12/2025	Peer-Reviewed	Reviewed Draft
1.0	31/12/2025	Submitted	Final Version
1.1	23/02/2026	Draft	Initial Draft
1.2	28/02/2026	Peer-Reviewed	Reviewed Draft
2.0	01/03/2026	Submitted	Final Version

Peer Review History

Version	Peer Review Date	Reviewed By
0.2	22/12/2025	Antonio Álvarez (OpenNebula)
1.2	24/02/2026	Antonio Álvarez (OpenNebula)
1.2	28/02/2026	Marco Mancini (OpenNebula)

Summary of Changes from Previous Versions

This is the second version of Deliverable D5.12

[1] A deliverable can be in one of these stages: Draft, Peer-Reviewed, Submitted, and Approved.

Executive Summary

This deliverable, D5.12, presents the final version of the Use Case demonstrations, using the final COGNIT components and architecture.

Use Case #1: Smart Cities. The demo from the smart city use case demonstrates how prioritization of vehicles such as buses in intersections with traffic lights is enhanced through the COGNIT function offloading capabilities, giving access to more powerful compute resources than what is available at the devices doing the vehicle detection.

Use Case #2: Wildfire Detection. The wildfire use case demos illustrate: The offloading of the image recognition function to COGNIT by an actual device, enabling functionality not available on the resource-constrained device. The successful scaling of the COGNIT compute resources needed to handle a (simulated) wildfire event, rapidly requiring increased resources.

Use Case #3: Smart Energy. The smart energy use case demos illustrate: How the COGNIT function offloading enables more advanced control functionality in Smart Meters, performing smart home control and appliance management. How the COGNIT Framework successfully scales to accommodate a large number of households equipped with smart energy meters operating simultaneously.

Use Case #4: Cybersecurity. The cybersecurity demos illustrate: How the COGNIT framework can enable increased cybersecurity protection through the use of an anomaly detection function that is offloaded to the COGNIT Framework and the use of Confidential Computing-enabled hardware. How the COGNIT Framework can switch the target endpoint for function offloading of anomaly detection capabilities, based on keeping the acceptable latency requirements in dynamic environments, thereby maintaining security for moving vehicles.

Together, the results from the Use Case demonstrations presented in this report illustrate the enhanced flexibility, efficiency, and adaptive capacity of the COGNIT Framework in dynamic cloud-edge environments.

The work reported in this document complements Deliverable D5.6 *Use Cases - Scientific Report*, which presents more details about the work done by all Use Case partners, and Deliverable D5.9 *COGNIT Framework - Software Source*, which provides easy access to the project software resources.

This deliverable has been released at the end of the final Research & Innovation Cycle, also coming with the conclusion of the project.

Table of Contents

Abbreviations and Acronyms **5**
1. Introduction **6**
2. ICE EdgeCluster setup **6**
3. Use Case #1: Smart Cities **7**
3.1 Main Scenario Description – Vehicle Prioritization 7
3.2 Demo Flow 7
3.3 Alternative scenarios 13
3.4 Impact, added value and lessons learned 14
3.5 Video of demonstration 14
4. Use Case #2: Wildfire Detection **15**
4.1 Main Scenario Description 15
4.2 Demo Flow 15
4.3 Alternative scenarios 19
4.4 Impact, added value and lessons learned 19
4.5 Video of Demonstration 19
5. Use Case #3: Smart Energy **20**
5.1 Main Scenario Description 20
5.2 Demo Flow 21
5.3 Alternative scenario: Running a vast number of simulated applications – scalability demo 24
5.4 Impact, added value and lessons learned 28
5.5 Video demonstration 28
6. Use Case #4: Cybersecurity **29**
6.1 Scenario I Description - Confidential Computing (CC) 29
6.2 Demo I Flow 29
6.3 Scenario II Description - Anomaly Detection with Mobility 31
6.4 Demo II Flow 31
Figure 6.9: Edge cluster switching - CETIC to SE. Step 6.2 35
6.5 Impact, added value and lessons learned 36
6.6 Video of demonstrator 36
7. Conclusion **37**

Abbreviations and Acronyms

AI	Artificial Intelligence
API	Application Programming Interface
CC	Confidential Computing
DaaS	Data as a Service
EV	Electric Vehicle
FaaS	Function as a Service
IoT	Internet of Things
IP	Internet Protocol
M-Hub	Mobility Hub (advanced TLC)
OS	Operating System
RSU	Road Side Unit
RTOS	Real-Time Operating System
SEM	Smart Energy Meter
SR	Serverless Runtime
SSH	Secure Shell
SUMO	Simulation of Urban Mobility[2]
TRL	Technology Readiness Level
TTC	TreeTalker Cyber
VM	Virtual Machine

[2] An open source, highly portable, microscopic and continuous multi-modal traffic simulation package designed to handle large networks: https://eclipse.dev/sumo/

1. Introduction

This document presents the third and final version of the COGNIT Framework Demo. It showcases the culmination of the project's development and integration efforts. Each of the four different use cases presents its demos, which highlight how the COGNIT Framework has been applied in its respective domain. These demonstrators illustrate the versatility, scalability, and operational maturity of the COGNIT Framework in real-world and experimental settings.

The document is organised accordingly, with section 2 briefly presenting the shared test infrastructure and sections 3–6 presenting the use case demos.

2. ICE EdgeCluster setup

The ICE cluster, accessible for all use cases, consists of two hosts: sm07 and sm15, as reported in Table 2.1.

Table 2.1. The characteristics of the hosts related to the Edge Cluster

Host ID	Host Name	CPUs	Memory [GB]
0	sm07	32	1032
1	sm15	256	773

The two hosts present different characteristics in terms of CPU and Energy Consumption. The host sm15 has lower power consumption for the same amount of CPU usage than sm07.

3. Use Case #1: Smart Cities

3.1 Main Scenario Description – Vehicle Prioritization

Our demo scenario begins with the detection of a specialised vehicle, such as a city bus or emergency vehicle, specifically configured for this project. This will be detected by detectors and Road Side Units (RSUs), triggering priority processing in the Mobility Hub (M-Hub) edge. This will create a request to the COGNIT device client, which will offload the request to the serverless runtime. The runtime, in turn, will process the request by running a Simulation of Urban Mobility (SUMO) simulation to estimate delays for the buses. Afterwards, the device will compare this value with a threshold to make a decision, which will be returned by the priority system.

3.2 Demo Flow

Below follows a high-level walkthrough of what happens during the demo.

The big picture of this demo is depicted in this figure:

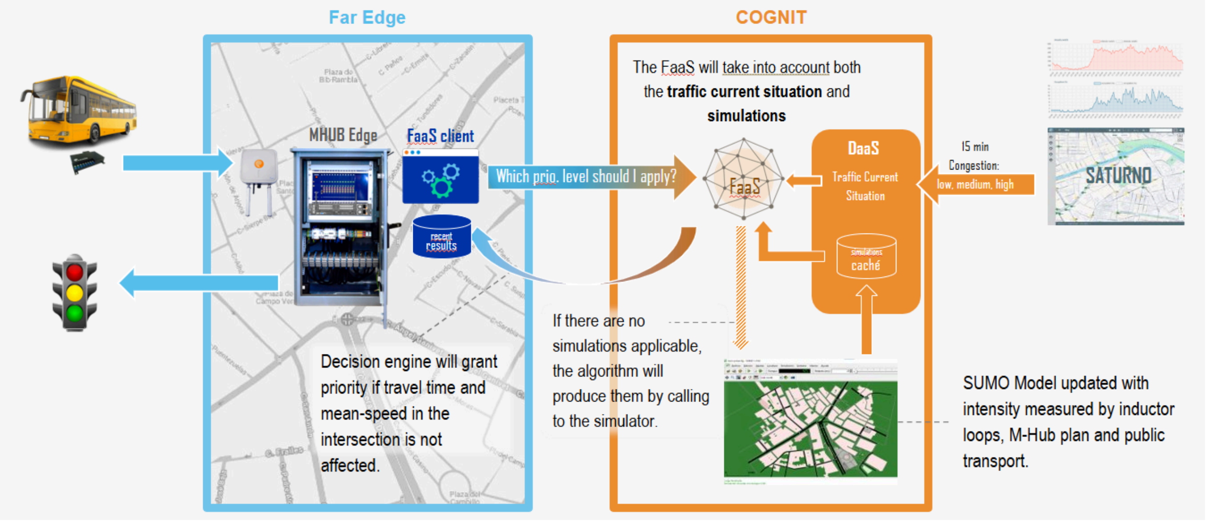

Far Edge
COGNIT
MHUB Edge
FaaS client
recent results
Which prio. level should I apply?
The FaaS will take into account both the **traffic current situation** and **simulations**
FaaS
DaaS
Traffic Current Situation
simulations caché
15 min
Congestion:
low, medium, high
SATURNO
Decision engine will grant priority if travel time and mean-speed in the intersection is not affected.
If there are no simulations applicable, the algorithm will produce them by calling to the simulator.
SUMO Model updated with intensity measured by inductor loops, M-Hub plan and public transport.

Figure 3.1: Elements involved in the demo

The elements involved in the demo are:

- The buses in the city of Granada, in the scope of this project, trigger detection events captured by the RSU.
- The RSU forwards the detection to the M-Hub Edge.
- The M-Hub Edge handles the request to the Priority subsystem.
- The Priority subsystem makes a request to the COGNIT Device Client.
- The COGNIT Device Client prepares the FaaS request for the COGNIT Serverless Runtime.
- The COGNIT Serverless Runtime executes the requested Function, which:
    - Obtains the status of the traffic condition from the DaaS. This status has been provided by the Saturno Traffic System, which collects traffic data from the sensors deployed on the roads. Saturno periodically updates the traffic status in the DaaS.

- If so configured, tries to obtain the result of a previous simulation in similar circumstances. For the purpose of this demo, we disabled this optional feature to always force the execution of the simulation.
- Calls the SUMO simulation of the junction, according to the traffic conditions.

- The Priority subsystem makes a decision about granting or not granting the priority according to bus time delays in the simulation and predefined thresholds.

In the following figures, we will show in detail some real-world traces from the demo.

These are the different elements shown in the demo:

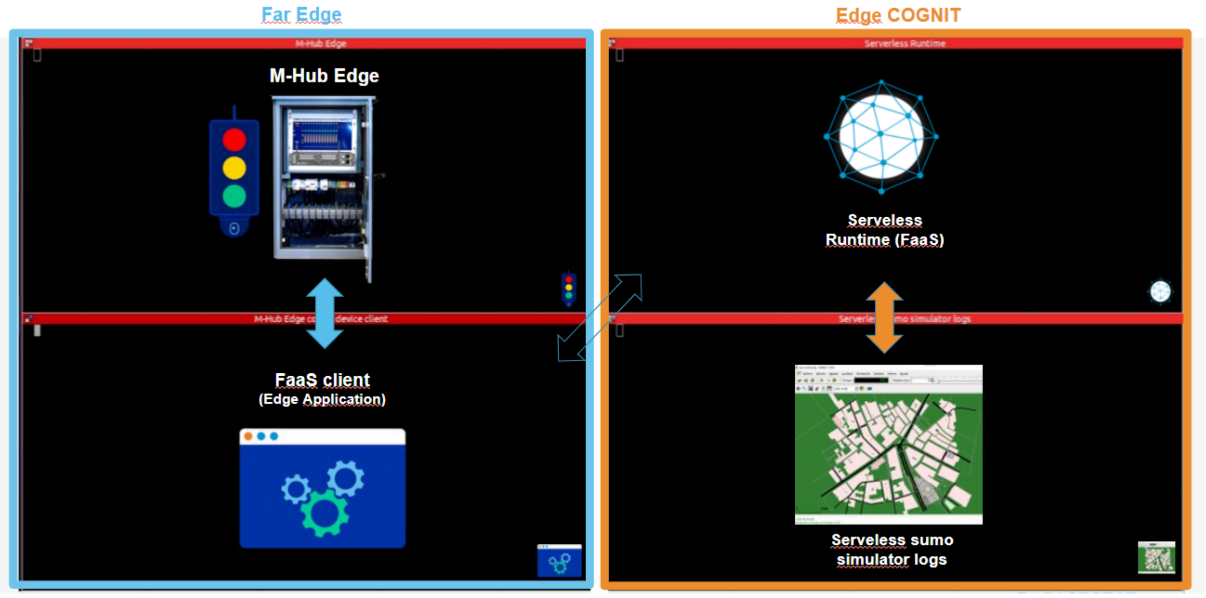
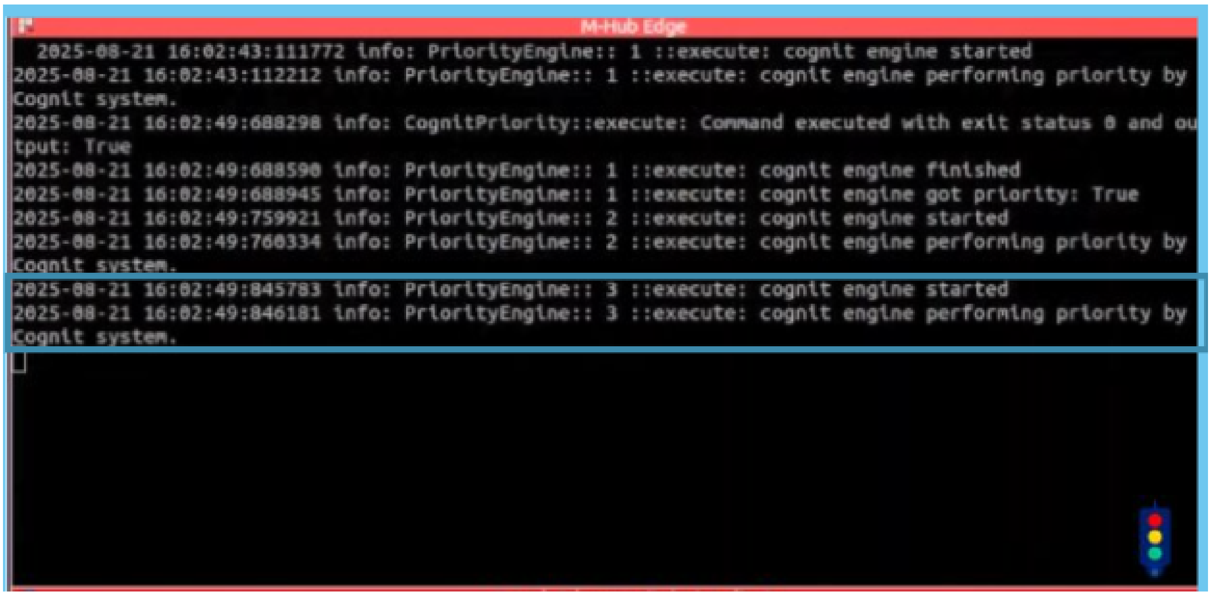


Figure 3.2: elements in the demo

The screens for the M-Hub Edge and the FaaS client, to the left, top and down, show logs captured from the device.

The screens for the Serverless Runtime (FaaS) and the Serverless sumo simulator, to the right top and down, show logs captured in the edge cluster, where the FaaS instances are run.

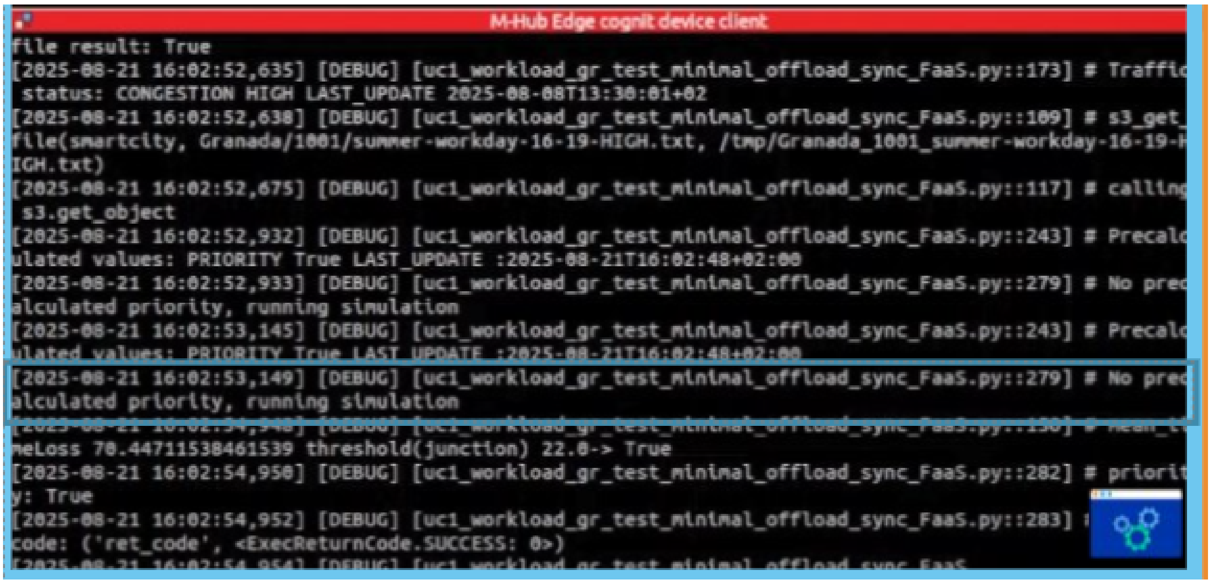
In the next figure, the Priority subsystem receives a detection and prepares a request to the COGNIT device client.



```
 2025-08-21 16:02:43:111772 info: PriorityEngine:: 1 ::execute: cognit engine started
2025-08-21 16:02:43:112212 info: PriorityEngine:: 1 ::execute: cognit engine performing priority by
Cognit system.
2025-08-21 16:02:49:688298 info: CognitPriority::execute: Command executed with exit status 0 and ou
tput: True
2025-08-21 16:02:49:688590 info: PriorityEngine:: 1 ::execute: cognit engine finished
2025-08-21 16:02:49:688945 info: PriorityEngine:: 1 ::execute: cognit engine got priority: True
2025-08-21 16:02:49:759921 info: PriorityEngine:: 2 ::execute: cognit engine started
2025-08-21 16:02:49:760334 info: PriorityEngine:: 2 ::execute: cognit engine performing priority by
Cognit system.
2025-08-21 16:02:49:845783 info: PriorityEngine:: 3 ::execute: cognit engine started
2025-08-21 16:02:49:846181 info: PriorityEngine:: 3 ::execute: cognit engine performing priority by
Cognit system.
```

Figure 3.3: Priority subsystem gets a detection and calls the device client

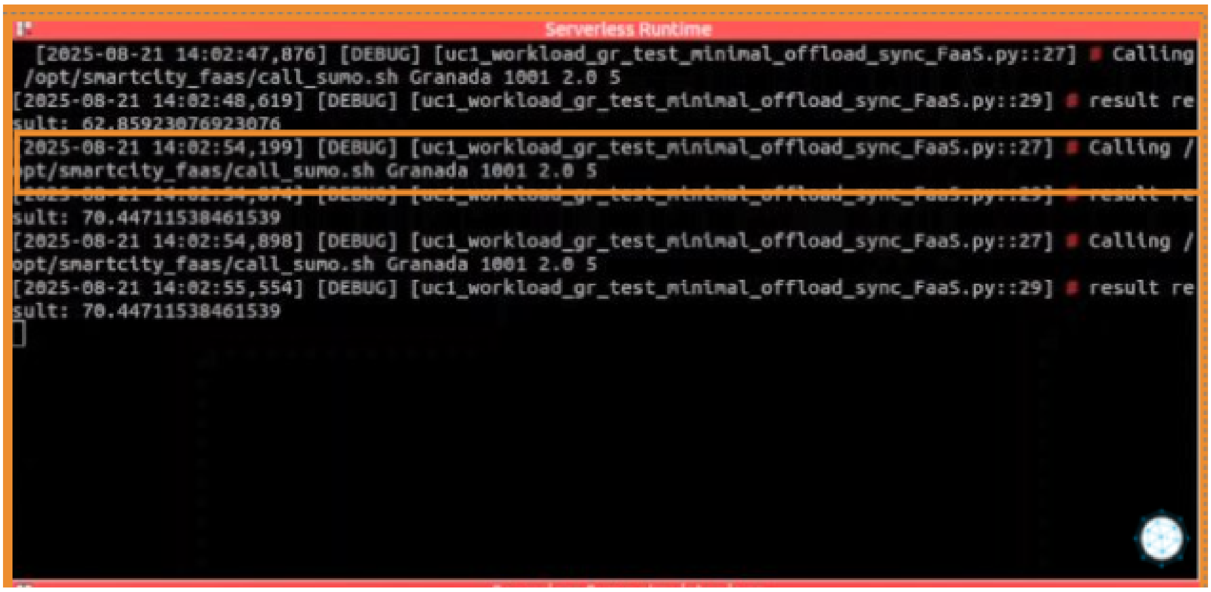
In the next figure, the Device Client gets called and prepares the FaaS request. As there is no previous simulation execution, it proceeds to run the FaaS to request a simulation.



```
file result: True
[2025-08-21 16:02:52,635] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::173] # Traffic
 status: CONGESTION HIGH LAST_UPDATE 2025-08-08T13:30:01+02
[2025-08-21 16:02:52,638] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::109] # s3_get_
file(smartcity, Granada/1001/summer-workday-16-19-HIGH.txt, /tmp/Granada_1001_summer-workday-16-19-H
IGH.txt)
[2025-08-21 16:02:52,675] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::117] # calling
 s3.get_object
[2025-08-21 16:02:52,932] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::243] # Precalc
ulated values: PRIORITY True LAST_UPDATE :2025-08-21T16:02:48+02:00
[2025-08-21 16:02:52,933] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::279] # No prec
alculated priority, running simulation
[2025-08-21 16:02:53,145] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::243] # Precalc
ulated values: PRIORITY True LAST_UPDATE :2025-08-21T16:02:48+02:00
[2025-08-21 16:02:53,149] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::279] # No prec
alculated priority, running simulation
[2025-08-21 16:02:54,948] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::130] # mean_ti
meLoss 70.44711538461539 threshold(junction) 22.0-> True
[2025-08-21 16:02:54,950] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::282] # priorit
y: True
[2025-08-21 16:02:54,952] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::283] #
code: ('ret_code', <ExecReturnCode.SUCCESS: 0>)
[2025-08-21 16:02:54,954] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS
```

Figure 3.4: Device client execution

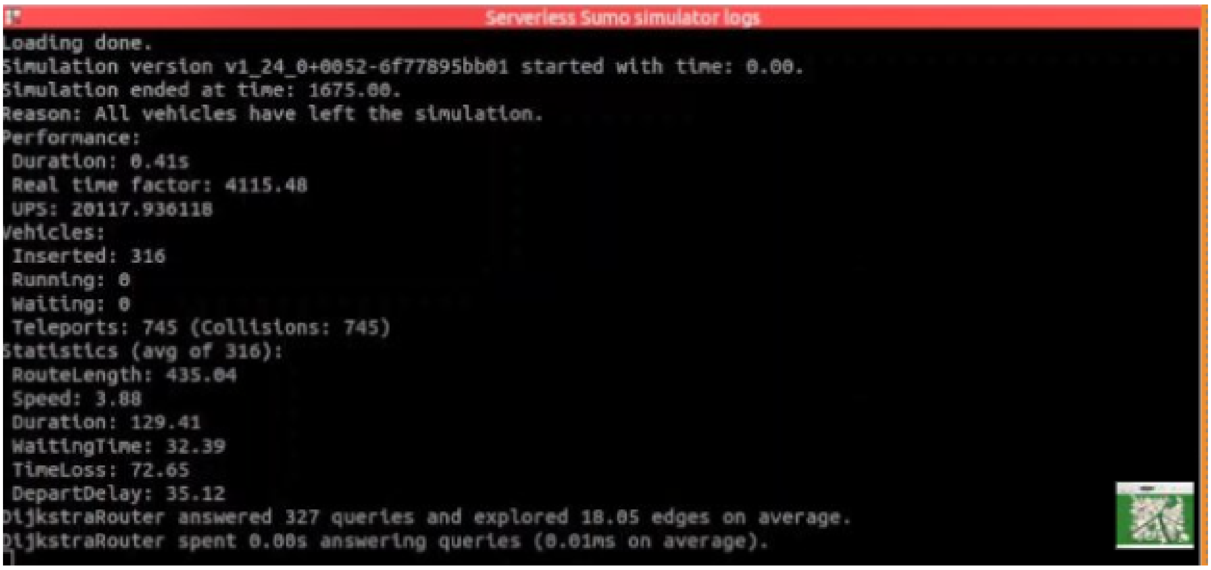
The Serverless Runtime executes the request function and calls the SUMO simulation.



```
 [2025-08-21 14:02:47,876] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] Calling
 /opt/smartcity_faas/call_sumo.sh Granada 1001 2.0 5
[2025-08-21 14:02:48,619] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] result re
sult: 62.85923076923076
[2025-08-21 14:02:54,199] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] Calling /
opt/smartcity_faas/call_sumo.sh Granada 1001 2.0 5
[2025-08-21 14:02:54,874] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] result re
sult: 70.44711538461539
[2025-08-21 14:02:54,898] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] Calling /
opt/smartcity_faas/call_sumo.sh Granada 1001 2.0 5
[2025-08-21 14:02:55,554] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] result re
sult: 70.44711538461539
```

Figure 3.5: Serverless Runtime calling SUMO simulation

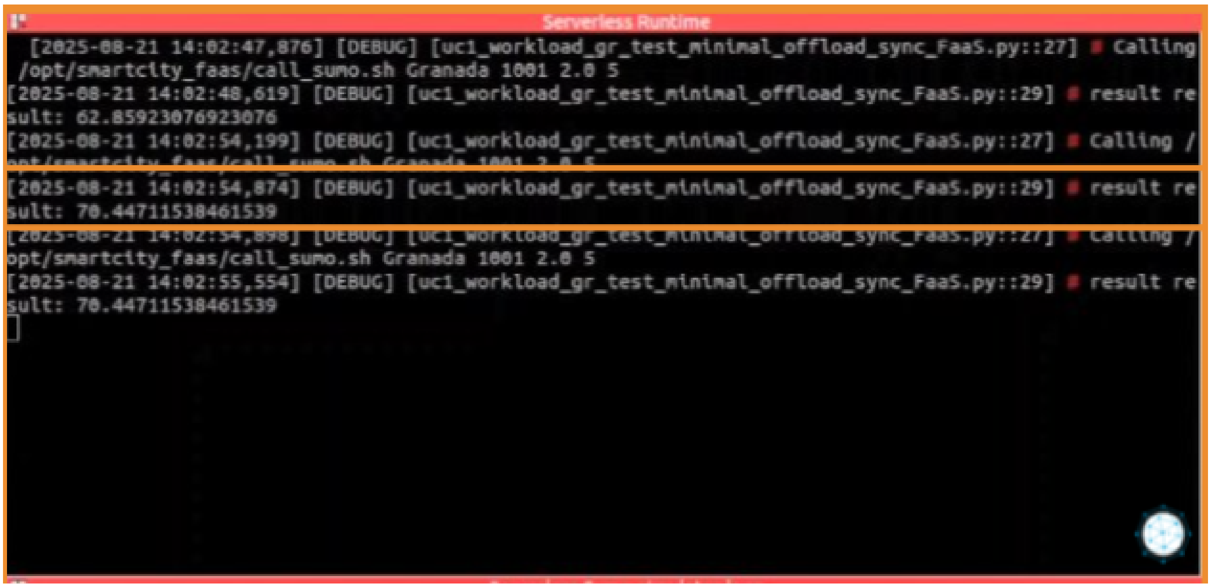
The SUMO simulation is executed:



```
Loading done.
Simulation version v1_24_0+0052-6f77895bb01 started with time: 0.00.
Simulation ended at time: 1675.00.
Reason: All vehicles have left the simulation.
Performance:
 Duration: 0.41s
 Real time factor: 4115.48
 UPS: 20117.936118
Vehicles:
 Inserted: 316
 Running: 0
 Waiting: 0
 Teleports: 745 (Collisions: 745)
Statistics (avg of 316):
 RouteLength: 435.04
 Speed: 3.88
 Duration: 129.41
 WaitingTime: 32.39
 TimeLoss: 72.65
 DepartDelay: 35.12
DijkstraRouter answered 327 queries and explored 18.05 edges on average.
DijkstraRouter spent 0.00s answering queries (0.01ms on average).
```

Figure 3.6: SUMO simulation execution

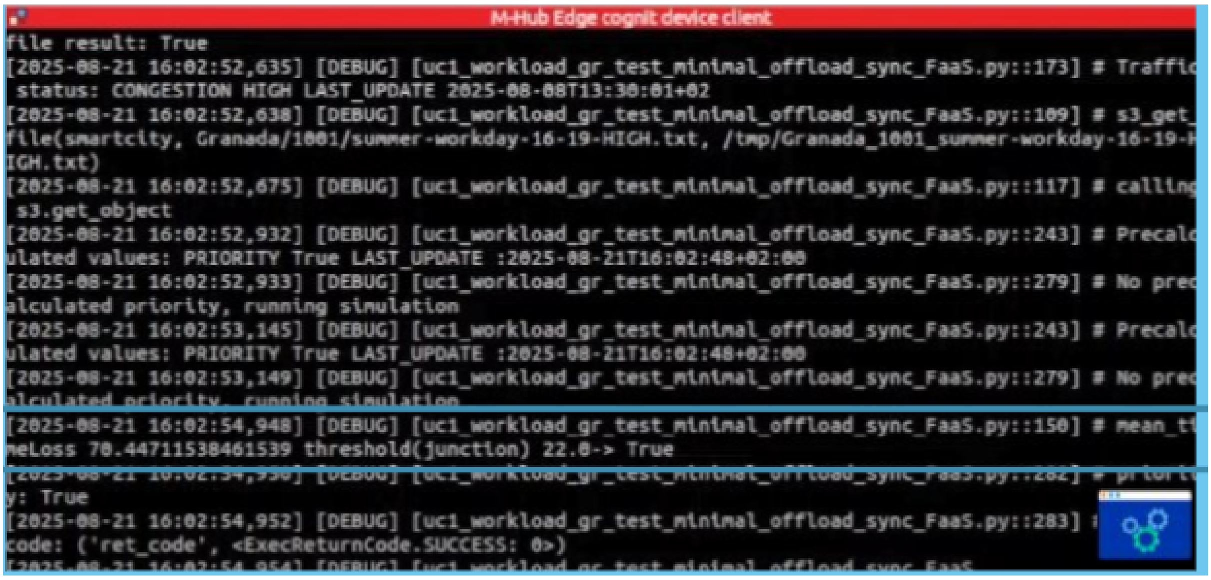
In the next figure, the response from SUMO is parsed, and the value of interest is extracted.



```
 [2025-08-21 14:02:47,876] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] ■ Calling
 /opt/smartcity_faas/call_sumo.sh Granada 1001 2.0 5
[2025-08-21 14:02:48,619] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] ■ result re
sult: 62.85923076923076
[2025-08-21 14:02:54,199] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] ■ Calling /
opt/smartcity_faas/call_sumo.sh Granada 1001 2.0 5
[2025-08-21 14:02:54,874] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] ■ result re
sult: 70.44711538461539
[2025-08-21 14:02:54,898] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] ■ Calling /
opt/smartcity_faas/call_sumo.sh Granada 1001 2.0 5
[2025-08-21 14:02:55,554] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] ■ result re
sult: 70.44711538461539
```

Figure 3.7: The FaaS obtain the result of the simulation

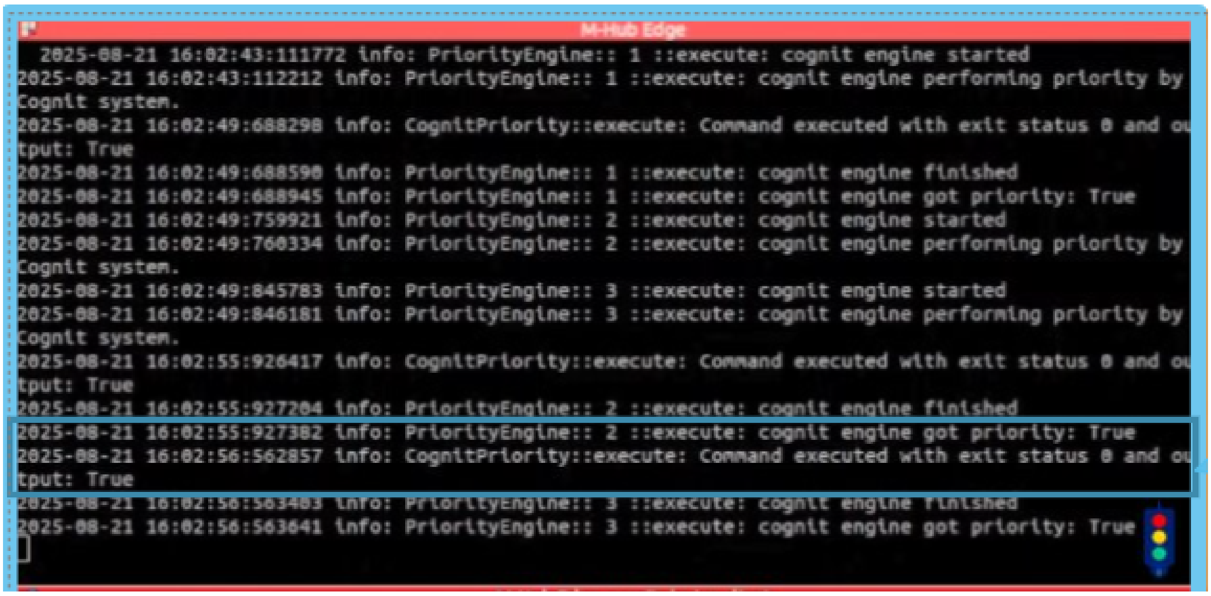
In the next figure, the Device Client checks the result received from the FaaS and, according to the threshold, decides whether to grant or not grant the priority.



```
file result: True
[2025-08-21 16:02:52,635] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::173] # Traffic
 status: CONGESTION HIGH LAST_UPDATE 2025-08-08T13:30:01+02
[2025-08-21 16:02:52,638] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::109] # s3_get_
file(smartcity, Granada/1001/summer-workday-16-19-HIGH.txt, /tmp/Granada_1001_summer-workday-16-19-H
IGH.txt)
[2025-08-21 16:02:52,675] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::117] # calling
 s3.get_object
[2025-08-21 16:02:52,932] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::243] # Precalc
ulated values: PRIORITY True LAST_UPDATE :2025-08-21T16:02:48+02:00
[2025-08-21 16:02:52,933] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::279] # No prec
alculated priority, running simulation
[2025-08-21 16:02:53,145] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::243] # Precalc
ulated values: PRIORITY True LAST_UPDATE :2025-08-21T16:02:48+02:00
[2025-08-21 16:02:53,149] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::279] # No prec
alculated priority, running simulation
[2025-08-21 16:02:54,948] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::150] # mean_ti
meLoss 70.44711538461539 threshold(junction) 22.0-> True
[2025-08-21 16:02:54,950] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::282] # priorit
y: True
[2025-08-21 16:02:54,952] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::283] #
code: ('ret_code', <ExecReturnCode.SUCCESS: 0>)
[2025-08-21 16:02:54,954] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS
```

Figure 3.8: The Device Client calculates the priority

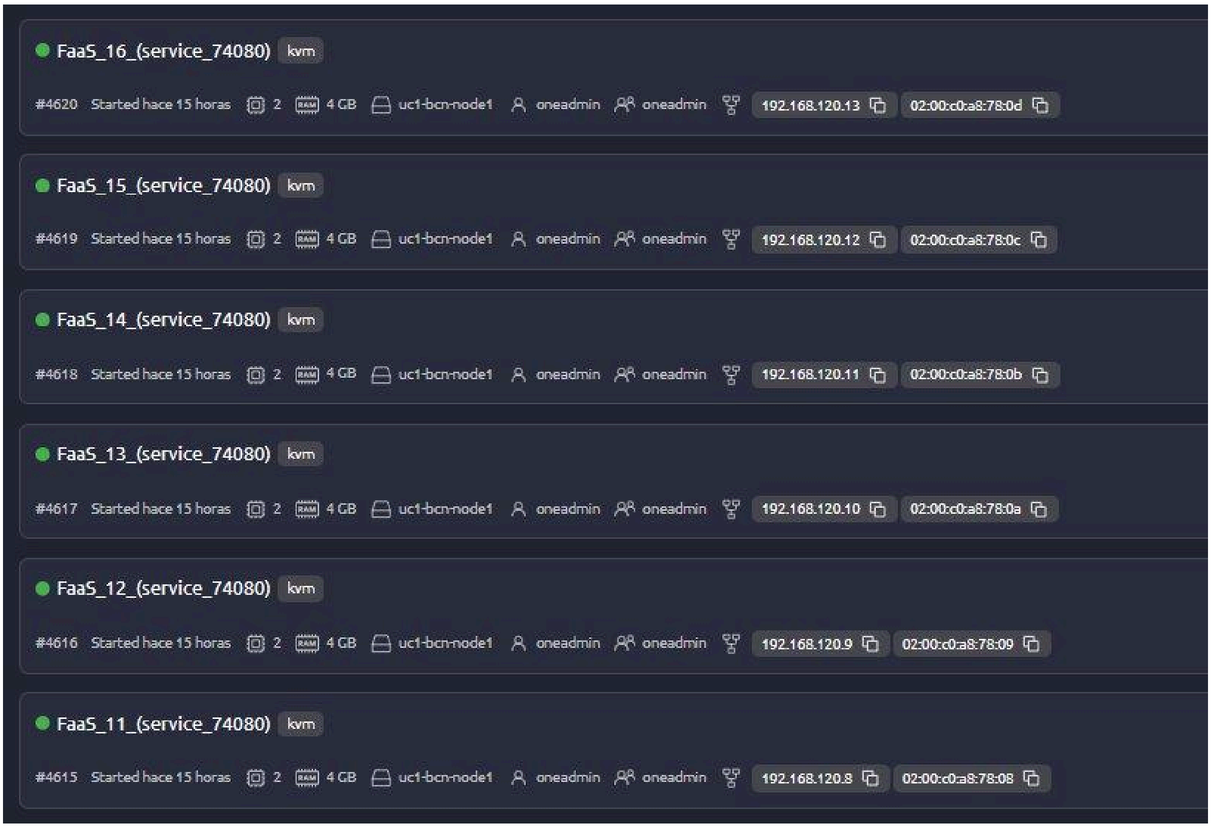
In the next figure, the priority subsystem receives the result of the FaaS.



```
 2025-08-21 16:02:43:111772 info: PriorityEngine:: 1 ::execute: cognit engine started
2025-08-21 16:02:43:112212 info: PriorityEngine:: 1 ::execute: cognit engine performing priority by
Cognit system.
2025-08-21 16:02:49:688298 info: CognitPriority::execute: Command executed with exit status 0 and ou
tput: True
2025-08-21 16:02:49:688590 info: PriorityEngine:: 1 ::execute: cognit engine finished
2025-08-21 16:02:49:688945 info: PriorityEngine:: 1 ::execute: cognit engine got priority: True
2025-08-21 16:02:49:759921 info: PriorityEngine:: 2 ::execute: cognit engine started
2025-08-21 16:02:49:760334 info: PriorityEngine:: 2 ::execute: cognit engine performing priority by
Cognit system.
2025-08-21 16:02:49:845783 info: PriorityEngine:: 3 ::execute: cognit engine started
2025-08-21 16:02:49:846181 info: PriorityEngine:: 3 ::execute: cognit engine performing priority by
Cognit system.
2025-08-21 16:02:55:926417 info: CognitPriority::execute: Command executed with exit status 0 and ou
tput: True
2025-08-21 16:02:55:927204 info: PriorityEngine:: 2 ::execute: cognit engine finished
2025-08-21 16:02:55:927382 info: PriorityEngine:: 2 ::execute: cognit engine got priority: True
2025-08-21 16:02:56:562857 info: CognitPriority::execute: Command executed with exit status 0 and ou
tput: True
2025-08-21 16:02:56:563483 info: PriorityEngine:: 3 ::execute: cognit engine finished
2025-08-21 16:02:56:563641 info: PriorityEngine:: 3 ::execute: cognit engine got priority: True
```

Figure 3.9: The Priority Subsystem gets the result of the priority granted

Finally, to illustrate the scaling capability, we show evidence of how the AI Orchestrator, developed under WP4, scales up the number of instances in the cluster as demand increases.



● FaaS_16_(service_74080) kvm

#4620 Started hace 15 horas 2 4 GB uc1-bcn-node1 oneadmin oneadmin 192.168.120.13 02:00:c0:a8:78:0d

● FaaS_15_(service_74080) kvm

#4619 Started hace 15 horas 2 4 GB uc1-bcn-node1 oneadmin oneadmin 192.168.120.12 02:00:c0:a8:78:0c

● FaaS_14_(service_74080) kvm

#4618 Started hace 15 horas 2 4 GB uc1-bcn-node1 oneadmin oneadmin 192.168.120.11 02:00:c0:a8:78:0b

● FaaS_13_(service_74080) kvm

#4617 Started hace 15 horas 2 4 GB uc1-bcn-node1 oneadmin oneadmin 192.168.120.10 02:00:c0:a8:78:0a

● FaaS_12_(service_74080) kvm

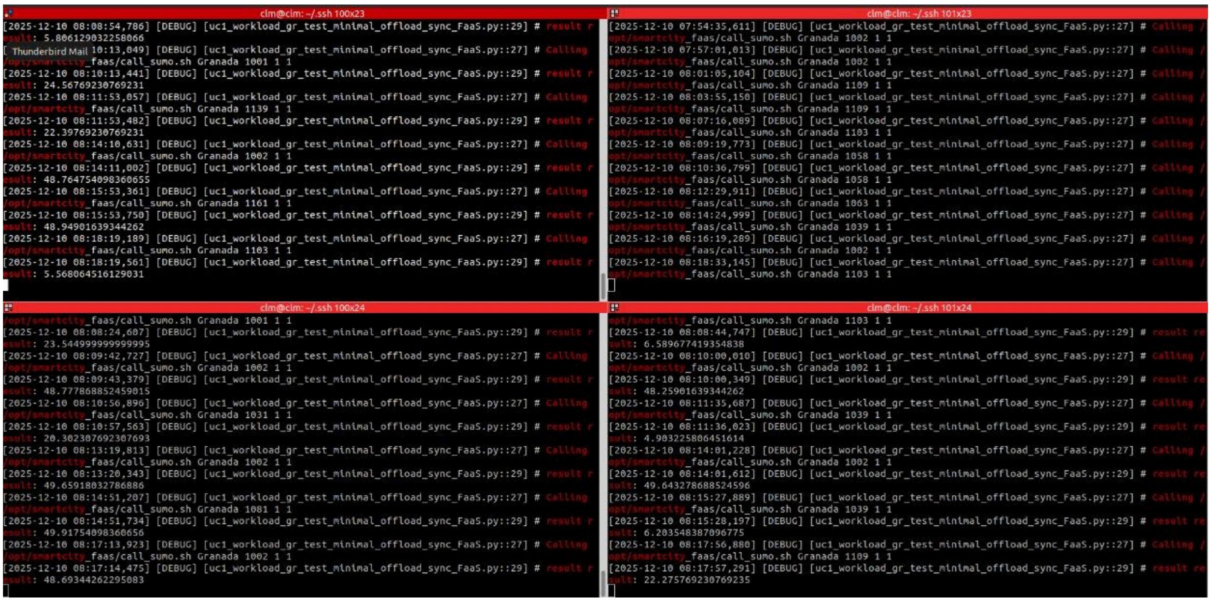
#4616 Started hace 15 horas 2 4 GB uc1-bcn-node1 oneadmin oneadmin 192.168.120.9 02:00:c0:a8:78:09

● FaaS_11_(service_74080) kvm

#4615 Started hace 15 horas 2 4 GB uc1-bcn-node1 oneadmin oneadmin 192.168.120.8 02:00:c0:a8:78:08

Figure 3.10: FaaS instances for Smartcity in the cluster

Traces of different instances processing requests from different devices.



```
[2025-12-10 08:08:54,786] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 5.806129032258066
[ Thunderbird Mail 10:13,049] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1001 1 1
[2025-12-10 08:10:13,441] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 24.56769230769231
[2025-12-10 08:11:53,057] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1139 1 1
[2025-12-10 08:11:53,482] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 22.39769230769231
[2025-12-10 08:14:10,631] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:14:11,002] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 48.764754098360655
[2025-12-10 08:15:53,361] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1161 1 1
[2025-12-10 08:15:53,750] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 48.94901639344262
[2025-12-10 08:18:19,189] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1103 1 1
[2025-12-10 08:18:19,561] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 5.568064516129031
```

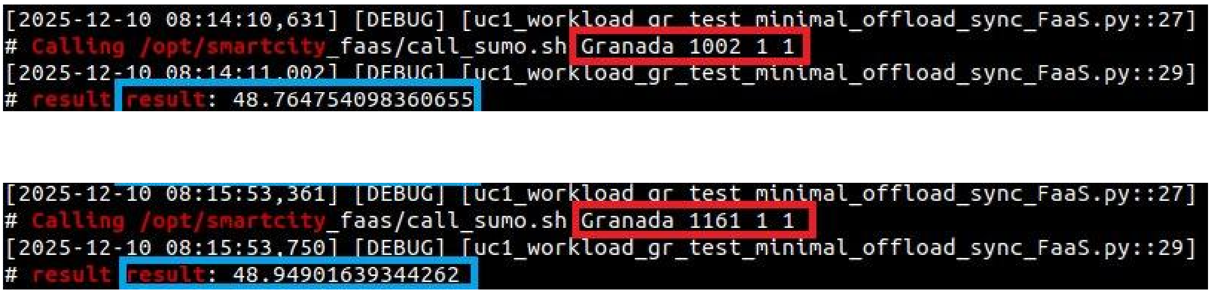
```
[2025-12-10 07:54:35,611] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 07:57:01,013] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:01:05,104] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1109 1 1
[2025-12-10 08:03:55,150] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1109 1 1
[2025-12-10 08:07:16,089] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1103 1 1
[2025-12-10 08:09:19,773] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1058 1 1
[2025-12-10 08:10:36,799] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1058 1 1
[2025-12-10 08:12:29,911] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1063 1 1
[2025-12-10 08:14:24,999] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1039 1 1
[2025-12-10 08:16:19,289] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:18:33,145] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1103 1 1
```

```
/opt/smartcity_faas/call_sumo.sh Granada 1001 1 1
[2025-12-10 08:08:24,607] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 23.544999999999995
[2025-12-10 08:09:42,727] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:09:43,379] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 48.777868852459015
[2025-12-10 08:10:56,896] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1031 1 1
[2025-12-10 08:10:57,563] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 20.302307692307693
[2025-12-10 08:13:19,813] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:13:20,343] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 49.65918032786886
[2025-12-10 08:14:51,207] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1081 1 1
[2025-12-10 08:14:51,734] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 49.91754098360656
[2025-12-10 08:17:13,923] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling
/opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:17:14,475] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result r
esult: 48.69344262295083
```

```
opt/smartcity_faas/call_sumo.sh Granada 1103 1 1
[2025-12-10 08:08:44,747] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result re
sult: 6.589677419354838
[2025-12-10 08:10:00,010] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:10:00,349] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result re
sult: 48.25901639344262
[2025-12-10 08:11:35,687] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1039 1 1
[2025-12-10 08:11:36,023] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result re
sult: 4.903225806451614
[2025-12-10 08:14:01,228] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:14:01,612] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result re
sult: 49.643278688524596
[2025-12-10 08:15:27,889] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1039 1 1
[2025-12-10 08:15:28,197] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result re
sult: 6.203548387096775
[2025-12-10 08:17:56,880] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27] # Calling /
opt/smartcity_faas/call_sumo.sh Granada 1109 1 1
[2025-12-10 08:17:57,291] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29] # result re
sult: 22.275769230769235
```

Figure 3.11: Traces of Serverless Runtime instances

And finally, the details of two requests from different devices:



```
[2025-12-10 08:14:10,631] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27]
# Calling /opt/smartcity_faas/call_sumo.sh Granada 1002 1 1
[2025-12-10 08:14:11,002] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29]
# result result: 48.764754098360655
```

```
[2025-12-10 08:15:53,361] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::27]
# Calling /opt/smartcity_faas/call_sumo.sh Granada 1161 1 1
[2025-12-10 08:15:53,750] [DEBUG] [uc1_workload_gr_test_minimal_offload_sync_FaaS.py::29]
# result result: 48.94901639344262
```

Figure 3.12: Requests from two devices

3.3 Alternative scenarios

The demo we presented above shows all the functionality and elements included in our use case.

Although initially we thought of other scenarios showing partial functionality, or simpler scenarios, finally we are showing the complete case involving real components in the field, and for this reason, we do not need those alternative scenarios, which were a way to show partial results in case we did not fully reach the goals with our more ambitious integrated scenario.

For showcasing purposes in different fora, we are showing two variants of our demo: one in which the priority is granted and another in which the priority request is rejected.

3.4 Impact, added value and lessons learned

The presented demo shows the capability of the Device Client to offload functions to the COGNIT Framework in a real-world scenario.

The Cloud-Edge Manager seamlessly deploys the required images (including the SUMO simulation) as the Serverless Runtimes without requiring any administrative action;;the distribution of the requests is driven by the application requirements configured for each device.

The AI-Enabled Orchestrator scales the number of Serverless Runtime instances up and down to adapt to changes in demand from the devices.

This Serverless Runtime provided an efficient and seamless way to execute simulations, which would have been a challenge if made on the devices. The centrally managed distributed system (Cloud-Edge Manager and AI-Enabled Orchestrator) also greatly simplified the administrative tasks required to manage this system. The capability to scale the system according to demand using the AI Orchestrator helps us find a balance between committed resources and efficiency.

In this project, one of the most challenging aspects has been the fact that all the partners have been evolving their developments according to the findings made, the needs or opportunities for improvement detected. With so many changes from different partners, and so many moving parts, a lot of coordination from all sides has been required.

3.5 Video of demonstration

Demonstration video of use-case 1 (Released August 2025, updated December 2025)

4. Use Case #2: Wildfire Detection

The Wildfire detection scenario is designed to demonstrate the application of the COGNIT Framework to IoT devices deployed in remote areas, in high numbers, and above all, with unpredictable request peaks. The TreeTalker Fire devices are equipped with various gas and infrared sensors for the early detection of a possible flame, the confirmation of which is entrusted to an image recognition algorithm using an RGB camera. A device that recognizes a flame can wake up nearby devices, resulting in a potential cascade effect. The reaction of COGNIT to this sudden peak of requests and the integration with the device is what is demonstrated by the demo.

4.1 Main Scenario Description

The demo is divided into two parts:

- The first part demonstrates the offloading of the image recognition function to COGNIT by an actual device when a possible fire is detected or following the wake-up signal from a nearby device.
- The second is a simulation of a TreeTalker Fire network to show the behaviour of COGNIT during a wildfire event with multiple functions offloaded at the same time.

The functions are offloaded to an on-premises server to simulate a local edge node of the civil protection to resemble a full-fledged deployment of a TreeTalker Fire network.

4.2 Demo Flow

The behaviour of a device during its different operational modes is presented below. Moreover, the scalability test is reported with the dashboard programmed to visually show the fire spread. Both demonstrations make use of the Python client as the device is capable of executing Python code.

The device characteristics and the software logic are described in the deliverable *“D5.6 Use Cases - Scientific Report - f”*.

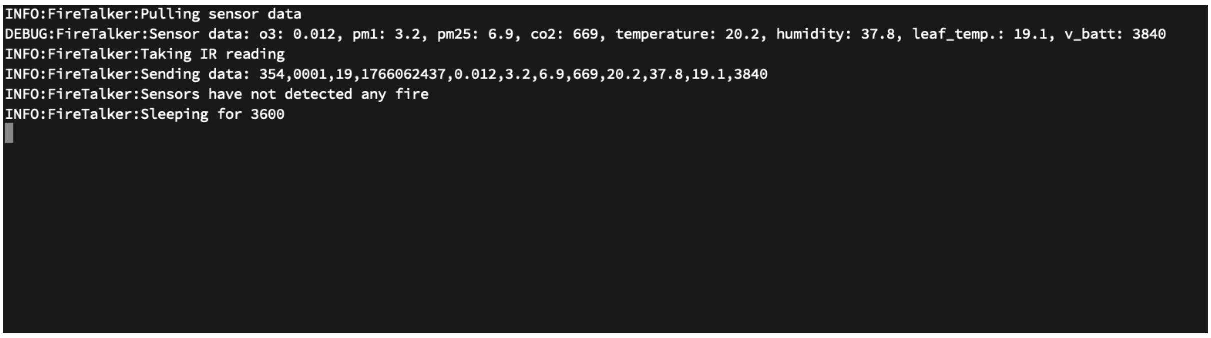
Device function offload

The device function offload demo aims to demonstrate the behaviour and interaction with the COGNIT Framework of a TreeTalker Fire to prove the use of COGNIT on the actual device.

Three different device prompts are shown based on the different operational modes of the device:

- Default mode: standard mode of the device;
- Fire mode: when a flame is detected by the infrared camera;
- Wake-up mode: when the wake-up signal is received by nearby devices.

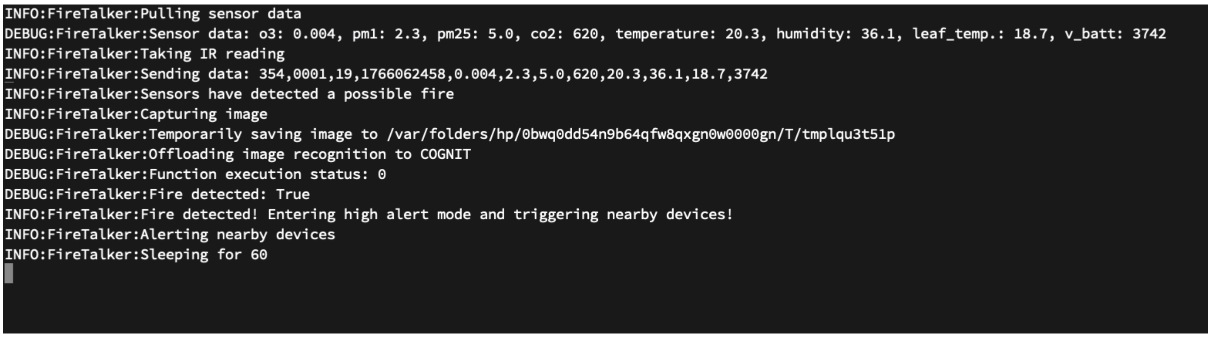
During default mode, the TreeTalker Fire wakes up every hour to collect sensor data (everything except for the RGB camera) before returning to sleep. The collected data contains concentrations for O_3, PM, and CO_2, air temperature and humidity, leaf temperature and an infrared image. This data is sent to a remote server for further processing, such as the development of fire risk maps and air quality monitoring. The logs for this default mode can be seen in Figure 4.1. To ensure better temporal coverage, devices are set to be activated at different times during installation.



```
INFO:FireTalker:Pulling sensor data
DEBUG:FireTalker:Sensor data: o3: 0.012, pm1: 3.2, pm25: 6.9, co2: 669, temperature: 20.2, humidity: 37.8, leaf_temp.: 19.1, v_batt: 3840
INFO:FireTalker:Taking IR reading
INFO:FireTalker:Sending data: 354,0001,19,1766062437,0.012,3.2,6.9,669,20.2,37.8,19.1,3840
INFO:FireTalker:Sensors have not detected any fire
INFO:FireTalker:Sleeping for 3600
```

Figure 4.1: Output prompt of a TreeTalker Fire device during default mode

The second mode shown in Figure 4.2 represents a device where both the sensors and the camera have detected a flame. To achieve this, a heating source was placed in front of the device to trigger the IR camera, along with a picture of a fire to trigger the RGB camera to recognize a fire.

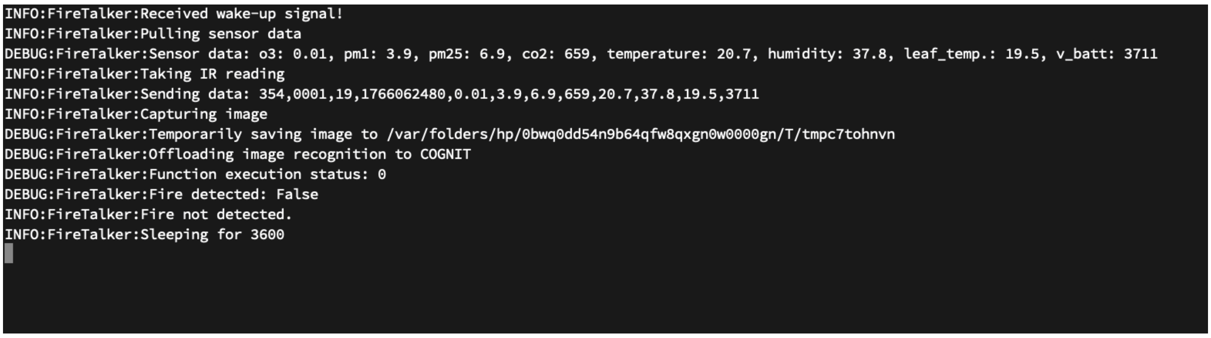


```
INFO:FireTalker:Pulling sensor data
DEBUG:FireTalker:Sensor data: o3: 0.004, pm1: 2.3, pm25: 5.0, co2: 620, temperature: 20.3, humidity: 36.1, leaf_temp.: 18.7, v_batt: 3742
INFO:FireTalker:Taking IR reading
INFO:FireTalker:Sending data: 354,0001,19,1766062458,0.004,2.3,5.0,620,20.3,36.1,18.7,3742
INFO:FireTalker:Sensors have detected a possible fire
INFO:FireTalker:Capturing image
DEBUG:FireTalker:Temporarily saving image to /var/folders/hp/0bwq0dd54n9b64qfw8qxgn0w0000gn/T/tmplqu3t51p
DEBUG:FireTalker:Offloading image recognition to COGNIT
DEBUG:FireTalker:Function execution status: 0
DEBUG:FireTalker:Fire detected: True
INFO:FireTalker:Fire detected! Entering high alert mode and triggering nearby devices!
INFO:FireTalker:Alerting nearby devices
INFO:FireTalker:Sleeping for 60
```

Figure 4.2: Output prompt of a TreeTalker Fire device entering high alert mode

In this second case, after waking up, the device detects a possible flame, so instead of returning to sleep, the RGB image is captured, and the image recognition function is offloaded to COGNIT. The status of the function reports possible errors during the offload of the function, and the result of the image recognition function is also reported. When the flame is detected, the device enters high alert mode and a distress signal is sent to nearby devices. The device enters sleep mode for 60 seconds, then the cycle is repeated until a flame is not detected for 10 consecutive cycles.

The last case is when a distress signal is received by the device. The corresponding prompt is shown in Figure 4.3.



```
INFO:FireTalker:Received wake-up signal!
INFO:FireTalker:Pulling sensor data
DEBUG:FireTalker:Sensor data: o3: 0.01, pm1: 3.9, pm25: 6.9, co2: 659, temperature: 20.7, humidity: 37.8, leaf_temp.: 19.5, v_batt: 3711
INFO:FireTalker:Taking IR reading
INFO:FireTalker:Sending data: 354,0001,19,1766062480,0.01,3.9,6.9,659,20.7,37.8,19.5,3711
INFO:FireTalker:Capturing image
DEBUG:FireTalker:Temporarily saving image to /var/folders/hp/0bwq0dd54n9b64qfw8qxgn0w0000gn/T/tmpc7tohnvn
DEBUG:FireTalker:Offloading image recognition to COGNIT
DEBUG:FireTalker:Function execution status: 0
DEBUG:FireTalker:Fire detected: False
INFO:FireTalker:Fire not detected.
INFO:FireTalker:Sleeping for 3600
```

Figure 4.3: Output prompt of a TreeTalker Fire device receiving a distress signal

This case has a similar cycle to the case in which a possible flame is detected by the sensor, with the only difference being that the image is directly offloaded to the COGNIT framework without checking if the sensors detect a possible flame or not. If a flame is not detected, the device does not enter high alert mode but instead returns to sleep for one hour or until another distress call is received.

TreeTalker Fire network simulation

The TreeTalker Fire network simulation aims to test the behaviour of the COGNIT Framework in a real scenario, and this is very important to evaluate its scalability feature, which is a core functionality for UC2. The test of a forest-wide fire was impossible to perform in a real environment, so a simulation was programmed instead. The chosen area for the simulation is reported in Figure 4.4.

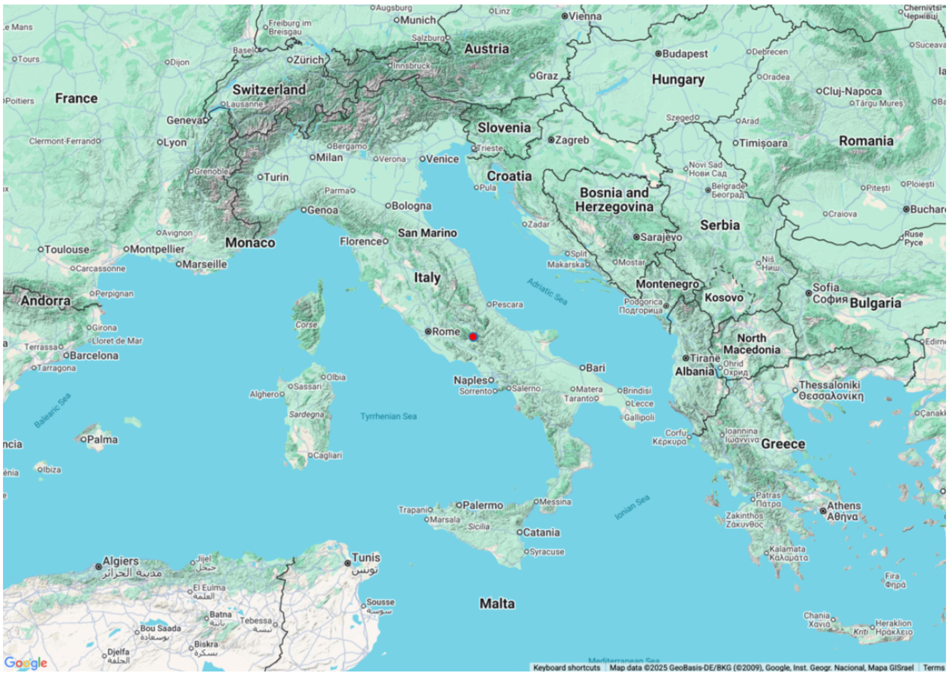
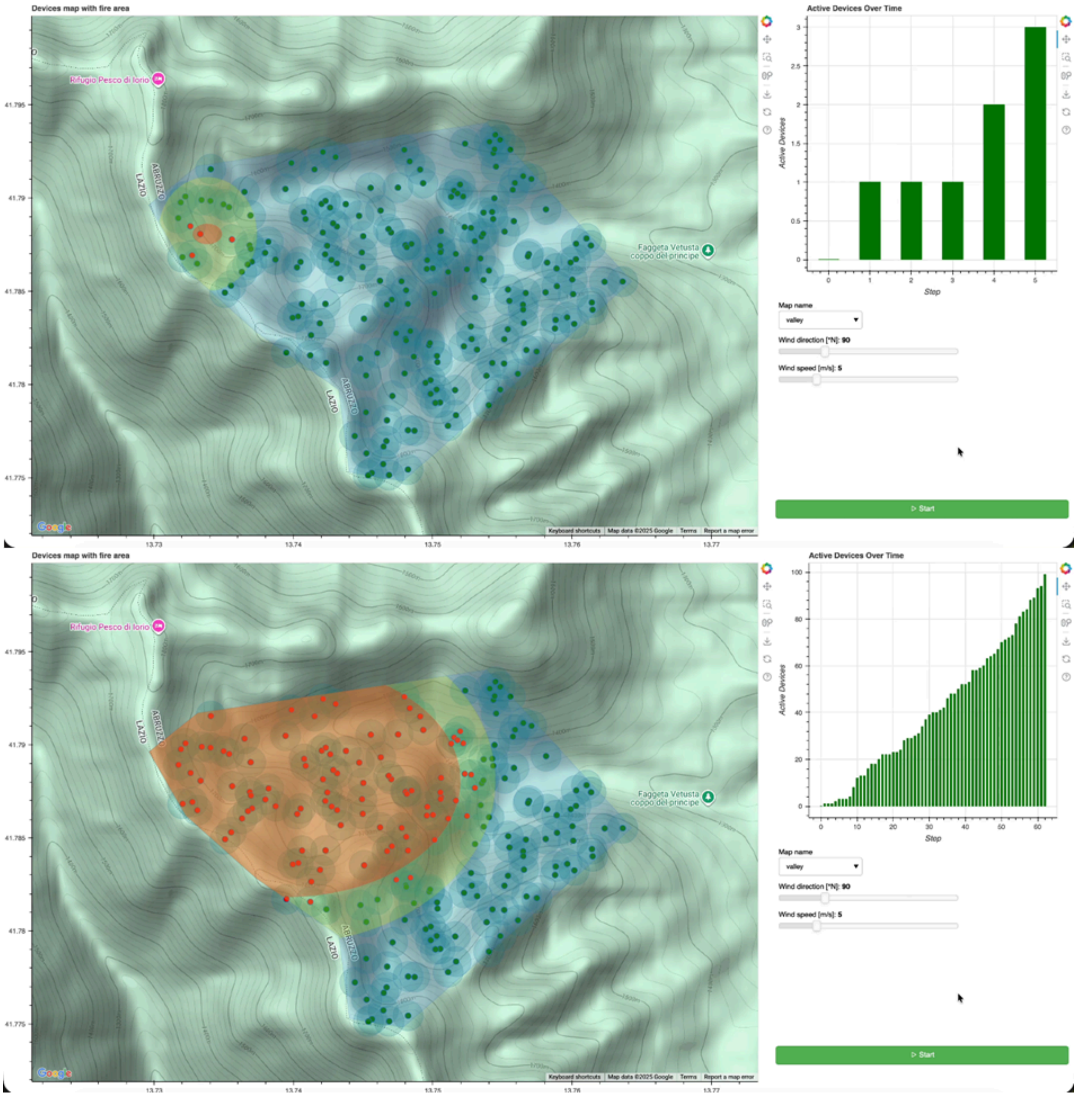


Figure 4.4: Location of the fire area with respect to Italy

The simulator takes as input a map from a selected set, a number of devices, along with wind angle and velocity. Both devices and the starting point for the fire are distributed randomly in the selected area. Each device has its own area of detection that is represented by a circle around the device itself. The fire spread is approximated with an elliptical geometry that has one focus at the starting point of the fire and elongates along the main direction of the wind. The simulation is performed in steps simulating the spread of the fire, the inner ellipse representing the area where the flame can be seen by the devices, while the outer ellipse is the area the wake-up signals of the devices in the inner ellipse can reach. In each iteration, the new area engulfed by the fire is calculated, and all the devices inside the two ellipses send an offload request to COGNIT at the same time. The offload request function of the software has been parallelized, and every device has its own device ID to best simulate a realistic deployment. For the devices located in the inner area, the offloaded function is passed along with a picture of a wildfire, while the devices located in the outer ellipse offload the function with an image of a forest. After the result of all the offloaded functions is returned, the next step is performed until all the devices are in high alert mode. Two screenshots of the simulation in different phases are reported in Figure 4.5.



Map name: valley

Wind direction [°N]: 90

Wind speed [m/s]: 5

Figure 4.5: Dashboard showing the simulation at the beginning and after several steps. On the right, the number of devices sending concurrent requests can be seen, as well as the bars to select wind speed and angle.

The results of the simulation show that all the device functions are successfully managed by the COGNIT Framework and the successful integration with the use case edge node, demonstrating the successful integration of the framework's strengths with respect to the wildfire use case.

4.3 Alternative scenarios

The scenario described successfully demonstrates the advantages the application of COGNIT brings to the wildfire use case, thus no alternative scenarios have been considered.

4.4 Impact, added value and lessons learned

The demos of the wildfire use case successfully tested the applicability of the COGNIT Framework to remotely located IoT devices such as the TreeTalker Fire.

The first demo shows how the device can offload the image recognition function and retrieve the result to take the needed steps based on it. This result demonstrates how complex algorithms can now be made available in IoT devices thanks to COGNIT.

The second demo focuses on how the COGNIT Framework reacts to sudden peaks of requests and how it would react during a wildfire event. The increasing number of requests was always successfully managed by the architecture in a seamless way, proving its scalability features.

The two demos also show the integration of an edge node within the framework architecture, which is an important feature for its application in organizations such as civil protection. The possibility of optimizing the use of distributed server resources is particularly useful for big organizations that have distributed offices, in particular for events with low probability and high impact, where dedicating fixed resources could become expensive and highly inefficient, while using the resources for everyday tasks and sharing them when a high-priority, massive event happens is valuable.

During the development of the demos, a comparison with the execution of the image recognition algorithm on the device was performed. The average time required by the device to perform the image recognition function is 50 seconds, compared with the 6.7 seconds required by COGNIT. Thus, improving the response time of the system as well as the power consumption, which was reduced almost tenfold.

Overall, the COGNIT framework proved to be a valuable asset for IoT devices, in particular in applications that present a high peak of requests and for companies and organizations with distributed server resources.

4.5 Video of Demonstration

Demonstration video of use-case 2 (Released August 2025, updated: December 2025)

5. Use Case #3: Smart Energy

As the Energy Use Case, we have been aiming to demonstrate that Smart Energy Meters (SEMs) powered by the capabilities of the COGNIT framework can play a significant role in the process of energy industry transformation. The main concept is that the meter is managing the energetically important devices in a household by offloading the AI decision function to the COGNIT Framework. Below, we present the demo application we have developed with a step-by-step description.

5.1 Main Scenario Description

During the testing phase, two validation setups were developed: a *simulation environment demo* and a *laboratory environment demo.* The former represents a full simulation of the use-case environment (including households and energy meters) executed on a PC using Python, and it can be easily scaled to scenarios involving a large number of concurrently simulated users. The latter setup simulates only the household component, with the demo application running on actual smart energy meters.

Videos illustrating examples of executions of the demos in both environments are publicly available:

- A fully simulated Python-based demo,
- A laboratory demo implemented in C

This section focuses on presenting the application flow for the second setup.

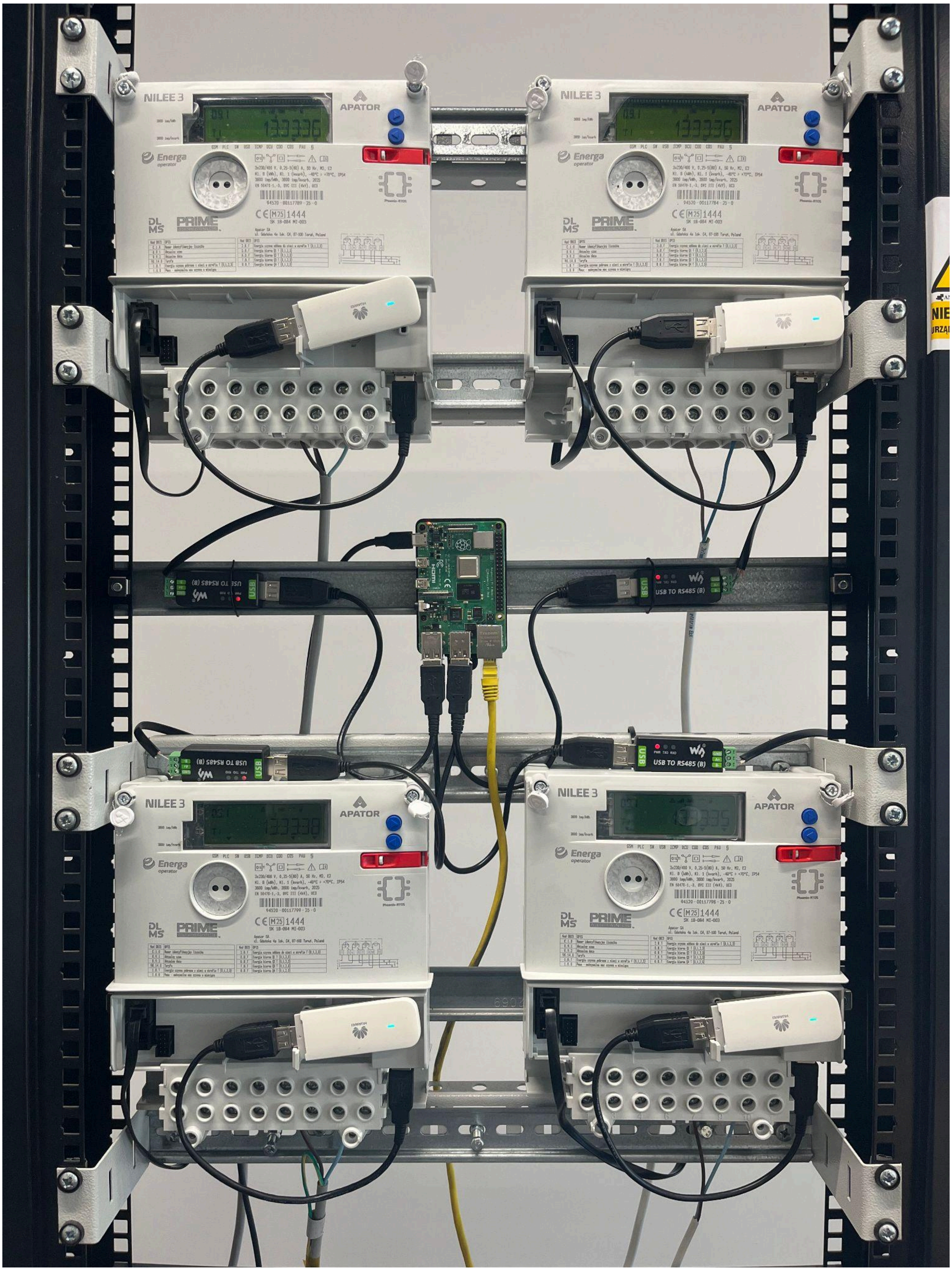


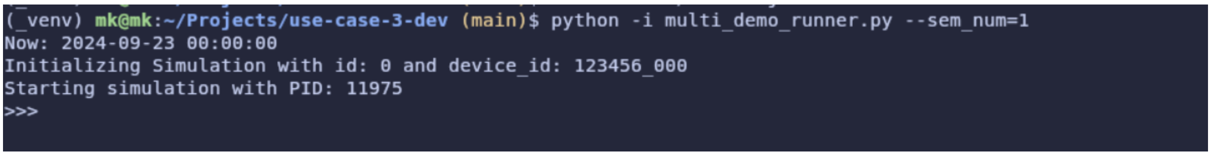
Figure 5.1: The setup of the laboratory environment demo.

5.2 Demo Flow

Simulation of household devices

The first step is to launch the household simulation process. This part of the UC3 demo's software is located (and well described) in the use-case-3 GitHub repository (links to different repositories are available in deliverable D5.9).

The simulation is started by running the Python script as illustrated in Figure 5.2 below.

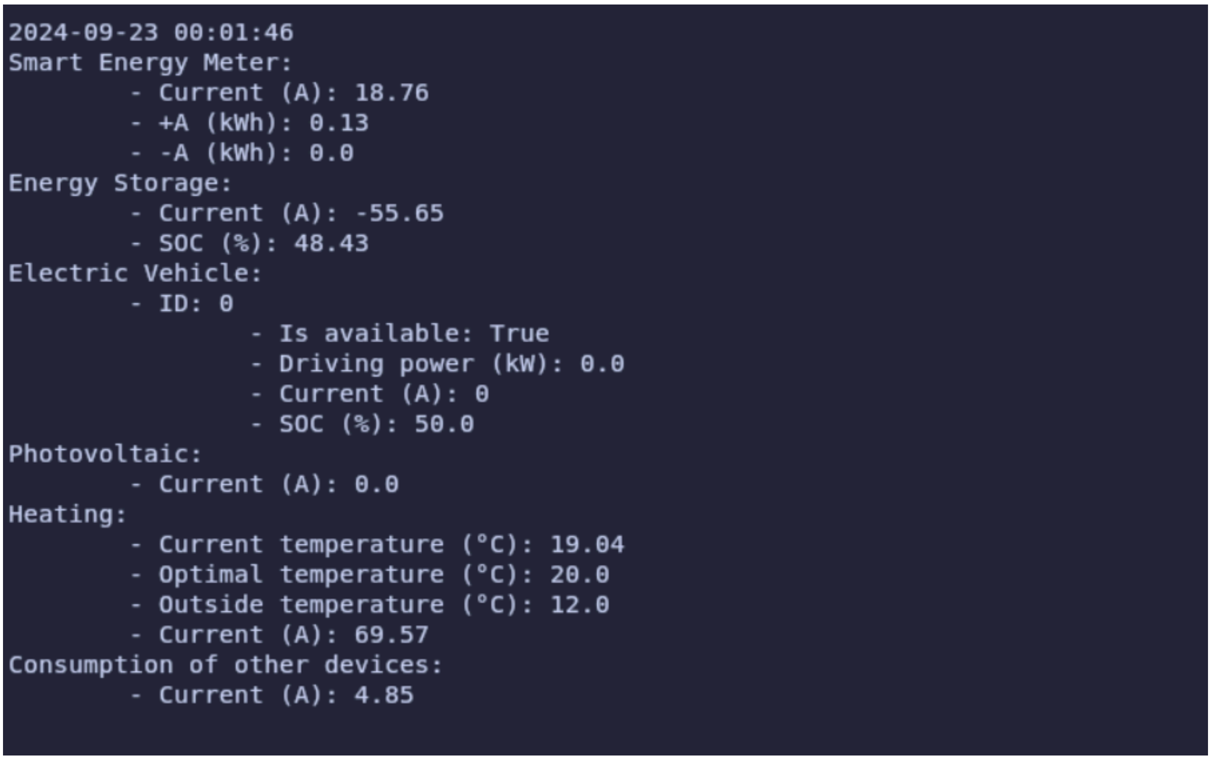


```
(_venv) mk@mk:~/Projects/use-case-3-dev (main)$ python -i multi_demo_runner.py --sem_num=1
Now: 2024-09-23 00:00:00
Initializing Simulation with id: 0 and device_id: 123456_000
Starting simulation with PID: 11975
>>>
```

Figure 5.2: Starting the household simulation process.

This spawns the devices' simulators for the scenario configuration determined in the file `scenario/{sem_id}.json`. The specifications and parameters of the household devices (electric vehicle, energy storage, heating facility) can be found there.

During the simulation, the state of the household appliances can be monitored live via the logs in file `logs/{pid}/{device_id}/simulation.log`.



```
2024-09-23 00:01:46
Smart Energy Meter:
        - Current (A): 18.76
        - +A (kWh): 0.13
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): -55.65
        - SOC (%): 48.43
Electric Vehicle:
        - ID: 0
                - Is available: True
                - Driving power (kW): 0.0
                - Current (A): 0
                - SOC (%): 50.0
Photovoltaic:
        - Current (A): 0.0
Heating:
        - Current temperature (°C): 19.04
        - Optimal temperature (°C): 20.0
        - Outside temperature (°C): 12.0
        - Current (A): 69.57
Consumption of other devices:
        - Current (A): 4.85
```

Figure 5.3: Live logs presenting the state of household appliances simulators.

Setting up the demo application on a SEM

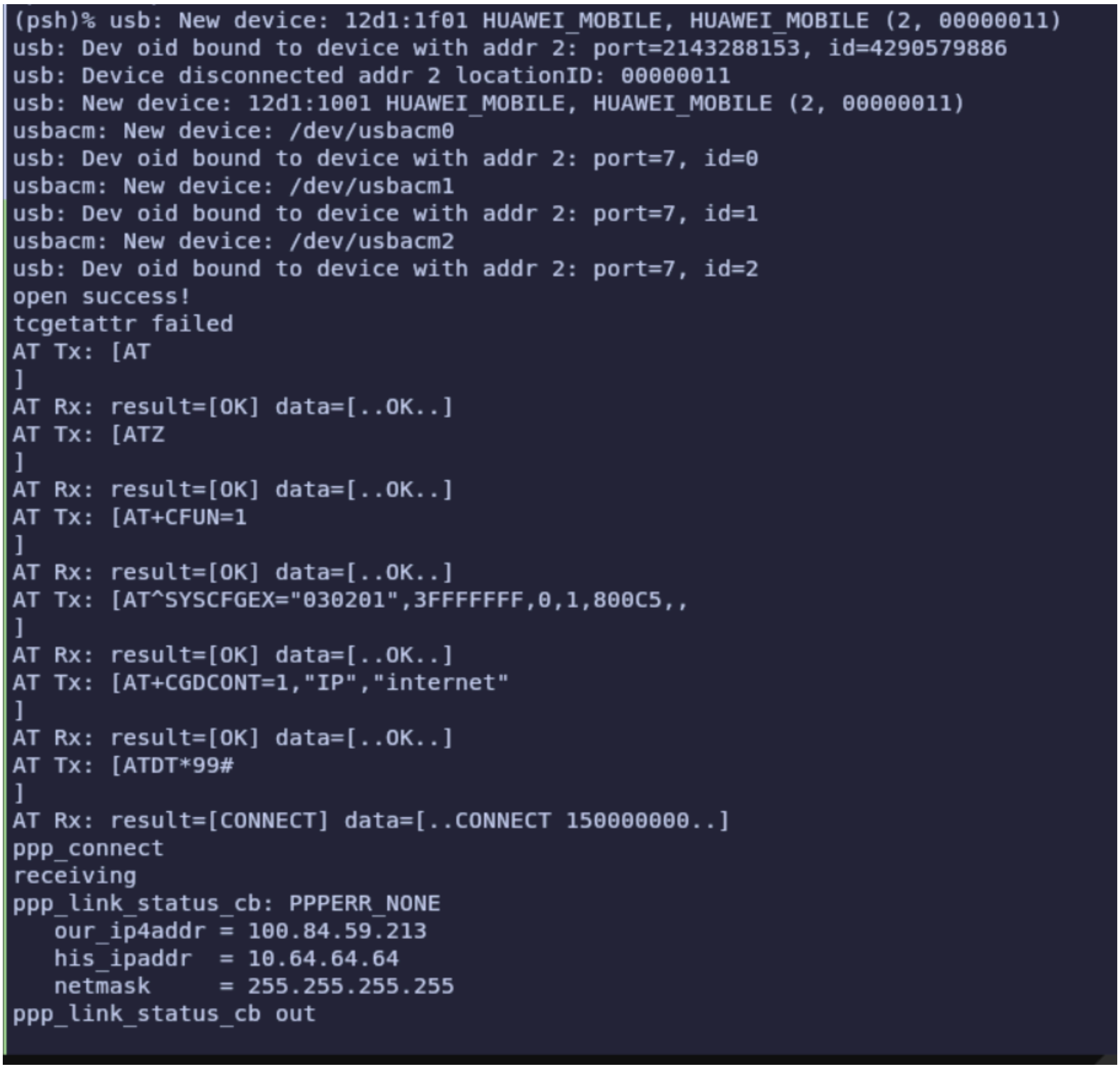
The demo application for the Smart Energy Meter is designed to be run with the Phoenix-RTOS operating system. Its software can be found in the `use-case-3-phoenix-demo` repository at the COGNIT project's GitHub, where it is well depicted how to launch this application on the `ia32` architecture emulator using Qemu.

Connection with household simulators – Modbus

The connection between SEM and simulators of household appliances is done over the Modbus protocol, which is one of the standard protocols used for managing various devices. The household simulation process (described above) provides a Modbus server for each of the appliances. The Modbus client (SEM) connects to the host, which runs the household simulation, via the RS-485 interface. In this way, the demo application can collect data from household appliances and manage their configuration.

Internet connection – USB modem

Another interface that the demo application needs is an Internet connection. This is provided by the USB GSM modem that is plugged into the meter and by the Phoenix-RTOS networking stack.

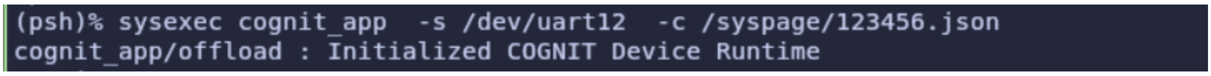


```
(psh)% usb: New device: 12d1:1f01 HUAWEI_MOBILE, HUAWEI_MOBILE (2, 00000011)
usb: Dev oid bound to device with addr 2: port=2143288153, id=4290579886
usb: Device disconnected addr 2 locationID: 00000011
usb: New device: 12d1:1001 HUAWEI_MOBILE, HUAWEI_MOBILE (2, 00000011)
usbacm: New device: /dev/usbacm0
usb: Dev oid bound to device with addr 2: port=7, id=0
usbacm: New device: /dev/usbacm1
usb: Dev oid bound to device with addr 2: port=7, id=1
usbacm: New device: /dev/usbacm2
usb: Dev oid bound to device with addr 2: port=7, id=2
open success!
tcgetattr failed
AT Tx: [AT
]
AT Rx: result=[OK] data=[..OK..]
AT Tx: [ATZ
]
AT Rx: result=[OK] data=[..OK..]
AT Tx: [AT+CFUN=1
]
AT Rx: result=[OK] data=[..OK..]
AT Tx: [AT^SYSCFGEX="030201",3FFFFFFF,0,1,800C5,,
]
AT Rx: result=[OK] data=[..OK..]
AT Tx: [AT+CGDCONT=1,"IP","internet"
]
AT Rx: result=[OK] data=[..OK..]
AT Tx: [ATDT*99#
]
AT Rx: result=[CONNECT] data=[..CONNECT 150000000..]
ppp_connect
receiving
ppp_link_status_cb: PPPERR_NONE
   our_ip4addr = 100.84.59.213
   his_ipaddr  = 10.64.64.64
   netmask     = 255.255.255.255
ppp_link_status_cb out
```

Figure 5.4: Establishing an internet connection through the GSM modem.

Starting the application – initializing the Device Runtime

We start the application through the psh – the Phoenix-RTOS shell. During initialization, the program establishes the connection with the household simulation and starts the COGNIT Device Runtime. Since SEMs are strictly limited by the resources in terms of RAM, we use the version of the COGNIT Device Runtime written in the C language.



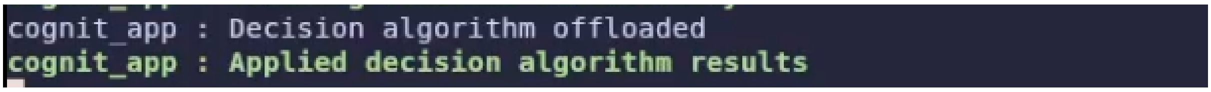
```
(psh)% sysexec cognit_app  -s /dev/uart12  -c /syspage/123456.json
cognit_app/offload : Initialized COGNIT Device Runtime
```

Figure 5.5: Initialization of the demo application and COGNIT Device Runtime.

Decision making

The AI decision algorithm and the three types of functions that are offloaded from SEM (decision, evaluation and training functions) are well described in the D5.6 document. Here, we only show how they are offloaded from the device.

The decision function is offloaded periodically with a frequency defined by the end user. In the most standard case, this should be done every 60 minutes. The offloading of the decision function can also be triggered immediately when manually requested.



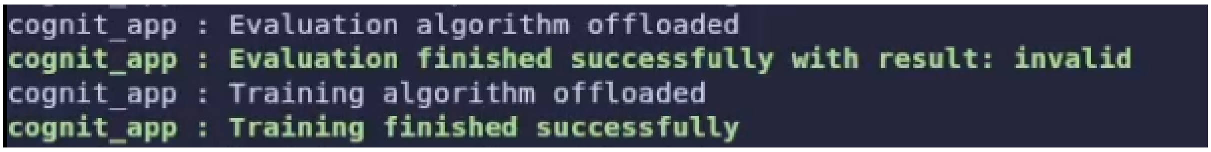
```
cognit_app : Decision algorithm offloaded
cognit_app : Applied decision algorithm results
```

Figure 5.6: Decision function offloading.

When the decision should be performed, the application first acquires the state of the household simulation. Then this state is used as the argument to the decision function, which is offloaded to the COGNIT Serverless Runtime. Finally, the result of the decision is obtained from the offloaded function and applied to the simulators of the household appliances.

Evaluation and retraining

The evaluation function is offloaded periodically or triggered manually by the user. It checks whether the AI model is performing well and if it should be retrained to better adjust it to the user's habits and needs. If the evaluation function's return value indicates that retraining is necessary, then retraining is offloaded as well.

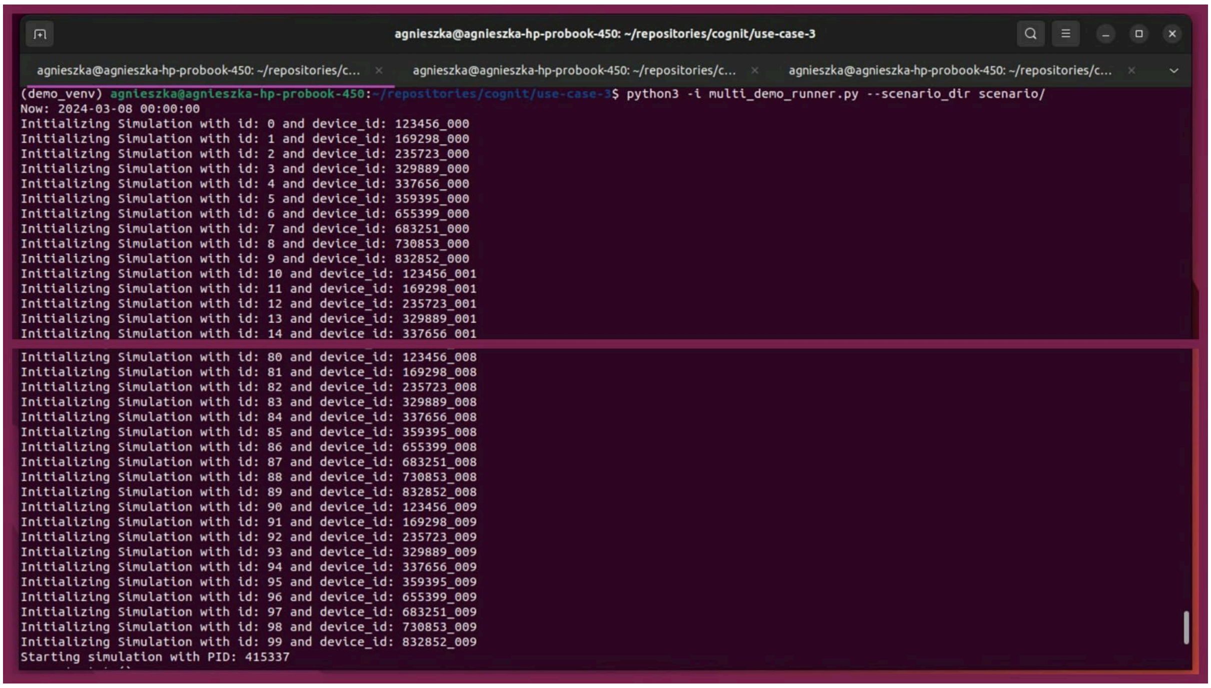


```
cognit_app : Evaluation algorithm offloaded
cognit_app : Evaluation finished successfully with result: invalid
cognit_app : Training algorithm offloaded
cognit_app : Training finished successfully
```

Figure 5.7: AI model evaluation triggering retraining.

5.3 Alternative scenario: Running a vast number of simulated applications – scalability demo

The Python-based demo enables the simulation of a large number of households equipped with smart energy meters operating simultaneously. The simulation process is initiated in the same manner as the laboratory simulation described in Section 5.2. Figure 5.8 presents the command execution along with logs confirming the correct initialization of all simulated devices. As shown, the simulation started on 08.03.2024 at 00:00, which represents the simulation's virtual time and is treated as the current time within the simulated environment.



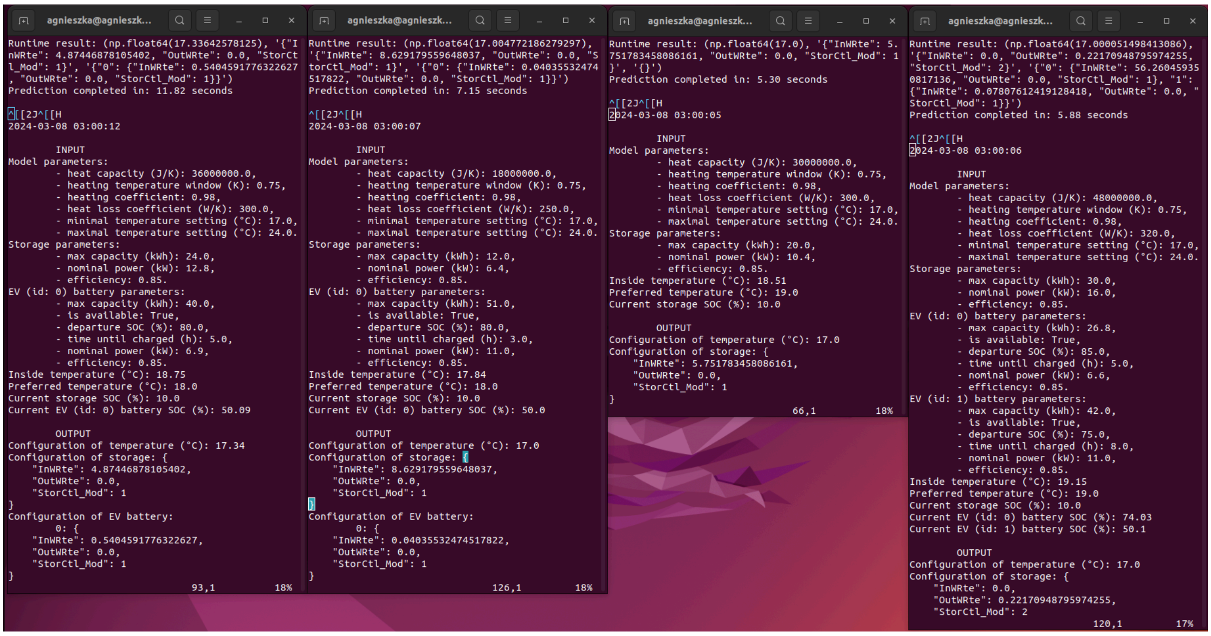
agnieszka@agnieszka-hp-probook-450: ~/repositories/cognit/use-case-3
agnieszka@agnieszka-hp-probook-450: ~/repositories/c...
agnieszka@agnieszka-hp-probook-450: ~/repositories/c...
agnieszka@agnieszka-hp-probook-450: ~/repositories/c...

```
(demo_venv) agnieszka@agnieszka-hp-probook-450:~/repositories/cognit/use-case-3$ python3 -i multi_demo_runner.py --scenario_dir scenario/
Now: 2024-03-08 00:00:00
Initializing Simulation with id: 0 and device_id: 123456_000
Initializing Simulation with id: 1 and device_id: 169298_000
Initializing Simulation with id: 2 and device_id: 235723_000
Initializing Simulation with id: 3 and device_id: 329889_000
Initializing Simulation with id: 4 and device_id: 337656_000
Initializing Simulation with id: 5 and device_id: 359395_000
Initializing Simulation with id: 6 and device_id: 655399_000
Initializing Simulation with id: 7 and device_id: 683251_000
Initializing Simulation with id: 8 and device_id: 730853_000
Initializing Simulation with id: 9 and device_id: 832852_000
Initializing Simulation with id: 10 and device_id: 123456_001
Initializing Simulation with id: 11 and device_id: 169298_001
Initializing Simulation with id: 12 and device_id: 235723_001
Initializing Simulation with id: 13 and device_id: 329889_001
Initializing Simulation with id: 14 and device_id: 337656_001
```

```
Initializing Simulation with id: 80 and device_id: 123456_008
Initializing Simulation with id: 81 and device_id: 169298_008
Initializing Simulation with id: 82 and device_id: 235723_008
Initializing Simulation with id: 83 and device_id: 329889_008
Initializing Simulation with id: 84 and device_id: 337656_008
Initializing Simulation with id: 85 and device_id: 359395_008
Initializing Simulation with id: 86 and device_id: 655399_008
Initializing Simulation with id: 87 and device_id: 683251_008
Initializing Simulation with id: 88 and device_id: 730853_008
Initializing Simulation with id: 89 and device_id: 832852_008
Initializing Simulation with id: 90 and device_id: 123456_009
Initializing Simulation with id: 91 and device_id: 169298_009
Initializing Simulation with id: 92 and device_id: 235723_009
Initializing Simulation with id: 93 and device_id: 329889_009
Initializing Simulation with id: 94 and device_id: 337656_009
Initializing Simulation with id: 95 and device_id: 359395_009
Initializing Simulation with id: 96 and device_id: 655399_009
Initializing Simulation with id: 97 and device_id: 683251_009
Initializing Simulation with id: 98 and device_id: 730853_009
Initializing Simulation with id: 99 and device_id: 832852_009
Starting simulation with PID: 415337
```

Figure 5.8: Starting the scalability demo with initialization logs.

Household configurations differ in terms of the number and types of appliances, their technical specifications, and residents' preferences, such as EV usage schedules and comfort temperature settings. Additionally, households may exhibit different computation offloading patterns depending on user requirements. However, for the purpose of this scenario, an identical offloading configuration was applied to all households. The objective was to evaluate a worst-case scenario with a high degree of overlapping requests and to verify the capability of the COGNIT infrastructure to execute a large number of functions concurrently within the expected time constraints.



```
Runtime result: (np.float64(17.33642578125), '{"I
nWRte": 4.87446878105402, "OutWRte": 0.0, "StorCt
l_Mod": 1}', '{"0": {"InWRte": 0.5404591776322627
, "OutWRte": 0.0, "StorCtl_Mod": 1}}')
Prediction completed in: 11.82 seconds

^[[2J^[[H
2024-03-08 03:00:12

        INPUT
Model parameters:
        - heat capacity (J/K): 36000000.0,
        - heating temperature window (K): 0.75,
        - heating coefficient: 0.98,
        - heat loss coefficient (W/K): 300.0,
        - minimal temperature setting (°C): 17.0,
        - maximal temperature setting (°C): 24.0.
Storage parameters:
        - max capacity (kWh): 24.0,
        - nominal power (kW): 12.8,
        - efficiency: 0.85.
EV (id: 0) battery parameters:
        - max capacity (kWh): 40.0,
        - is available: True,
        - departure SOC (%): 80.0,
        - time until charged (h): 5.0,
        - nominal power (kW): 6.9,
        - efficiency: 0.85.
Inside temperature (°C): 18.75
Preferred temperature (°C): 18.0
Current storage SOC (%): 10.0
Current EV (id: 0) battery SOC (%): 50.09

        OUTPUT
Configuration of temperature (°C): 17.34
Configuration of storage: {
    "InWRte": 4.87446878105402,
    "OutWRte": 0.0,
    "StorCtl_Mod": 1
}
Configuration of EV battery:
        0: {
    "InWRte": 0.5404591776322627,
    "OutWRte": 0.0,
    "StorCtl_Mod": 1
}
                                    93,1          18%
```

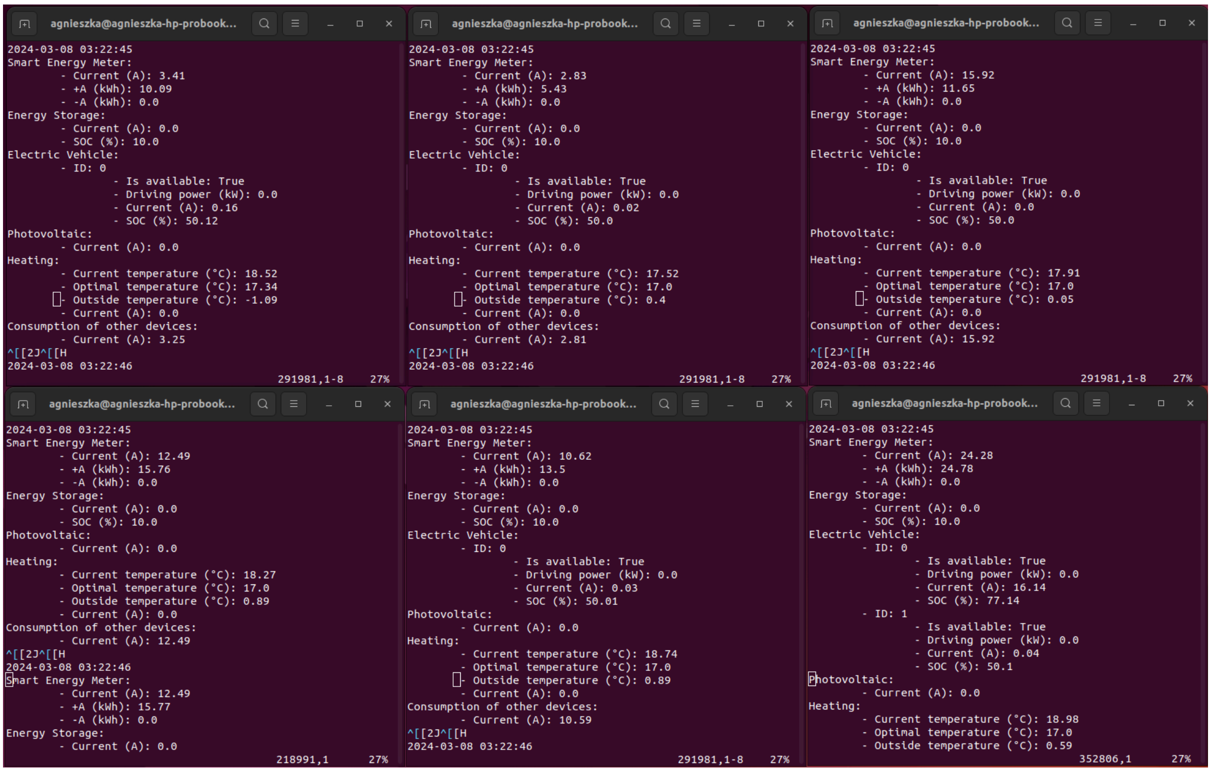
```
Runtime result: (np.float64(17.004772186279297),
'{"InWRte": 8.629179559648037, "OutWRte": 0.0, "S
torCtl_Mod": 1}', '{"0": {"InWRte": 0.04035532474
517822, "OutWRte": 0.0, "StorCtl_Mod": 1}}')
Prediction completed in: 7.15 seconds

^[[2J^[[H
2024-03-08 03:00:07

        INPUT
Model parameters:
        - heat capacity (J/K): 18000000.0,
        - heating temperature window (K): 0.75,
        - heating coefficient: 0.98,
        - heat loss coefficient (W/K): 250.0,
        - minimal temperature setting (°C): 17.0,
        - maximal temperature setting (°C): 24.0.
Storage parameters:
        - max capacity (kWh): 12.0,
        - nominal power (kW): 6.4,
        - efficiency: 0.85.
EV (id: 0) battery parameters:
        - max capacity (kWh): 51.0,
        - is available: True,
        - departure SOC (%): 80.0,
        - time until charged (h): 3.0,
        - nominal power (kW): 11.0,
        - efficiency: 0.85.
Inside temperature (°C): 17.84
Preferred temperature (°C): 18.0
Current storage SOC (%): 10.0
Current EV (id: 0) battery SOC (%): 50.0

        OUTPUT
Configuration of temperature (°C): 17.0
Configuration of storage: {
    "InWRte": 8.629179559648037,
    "OutWRte": 0.0,
    "StorCtl_Mod": 1
}
Configuration of EV battery:
        0: {
    "InWRte": 0.04035532474517822,
    "OutWRte": 0.0,
    "StorCtl_Mod": 1
}
                                    126,1         18%
```

```
Runtime result: (np.float64(17.0), '{"InWRte": 5.
751783458086161, "OutWRte": 0.0, "StorCtl_Mod": 1
}', '{}')
Prediction completed in: 5.30 seconds

^[[2J^[[H
2024-03-08 03:00:05

        INPUT
Model parameters:
        - heat capacity (J/K): 30000000.0,
        - heating temperature window (K): 0.75,
        - heating coefficient: 0.98,
        - heat loss coefficient (W/K): 300.0,
        - minimal temperature setting (°C): 17.0,
        - maximal temperature setting (°C): 24.0.
Storage parameters:
        - max capacity (kWh): 20.0,
        - nominal power (kW): 10.4,
        - efficiency: 0.85.
Inside temperature (°C): 18.51
Preferred temperature (°C): 19.0
Current storage SOC (%): 10.0

        OUTPUT
Configuration of temperature (°C): 17.0
Configuration of storage: {
    "InWRte": 5.751783458086161,
    "OutWRte": 0.0,
    "StorCtl_Mod": 1
}
                                    66,1          18%
```

```
Runtime result: (np.float64(17.000051498413086),
'{"InWRte": 0.0, "OutWRte": 0.22170948795974255,
"StorCtl_Mod": 2}', '{"0": {"InWRte": 56.26045935
0817136, "OutWRte": 0.0, "StorCtl_Mod": 1}, "1":
{"InWRte": 0.07807612419128418, "OutWRte": 0.0, "
StorCtl_Mod": 1}}')
Prediction completed in: 5.88 seconds

^[[2J^[[H
2024-03-08 03:00:06

        INPUT
Model parameters:
        - heat capacity (J/K): 48000000.0,
        - heating temperature window (K): 0.75,
        - heating coefficient: 0.98,
        - heat loss coefficient (W/K): 320.0,
        - minimal temperature setting (°C): 17.0,
        - maximal temperature setting (°C): 24.0.
Storage parameters:
        - max capacity (kWh): 30.0,
        - nominal power (kW): 16.0,
        - efficiency: 0.85.
EV (id: 0) battery parameters:
        - max capacity (kWh): 26.8,
        - is available: True,
        - departure SOC (%): 85.0,
        - time until charged (h): 5.0,
        - nominal power (kW): 6.6,
        - efficiency: 0.85.
EV (id: 1) battery parameters:
        - max capacity (kWh): 42.0,
        - is available: True,
        - departure SOC (%): 75.0,
        - time until charged (h): 8.0,
        - nominal power (kW): 11.0,
        - efficiency: 0.85.
Inside temperature (°C): 19.15
Preferred temperature (°C): 19.0
Current storage SOC (%): 10.0
Current EV (id: 0) battery SOC (%): 74.03
Current EV (id: 1) battery SOC (%): 50.1

        OUTPUT
Configuration of temperature (°C): 17.0
Configuration of storage: {
    "InWRte": 0.0,
    "OutWRte": 0.22170948795974255,
    "StorCtl_Mod": 2
                                    120,1         17%
```

Figure 5.9: Examples of user_app logs regarding execution of decision-making functions from the test.

Figure 5.9 shows excerpts from the user_app logs corresponding to the execution of decision-making functions at the same simulation time step for several randomly selected households. The terminal outputs indicate that different inputs provided to individually trained AI models resulted in diverse control decisions for end devices. The examples include households with one EV, multiple EVs, or no EV present. The function execution time is measured from the moment the task is offloaded by the Device Runtime Client to the COGNIT Frontend until the results are received.



```
2024-03-08 03:22:45
Smart Energy Meter:
        - Current (A): 3.41
        - +A (kWh): 10.09
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): 0.0
        - SOC (%): 10.0
Electric Vehicle:
        - ID: 0
                - Is available: True
                - Driving power (kW): 0.0
                - Current (A): 0.16
                - SOC (%): 50.12
Photovoltaic:
        - Current (A): 0.0
Heating:
        - Current temperature (°C): 18.52
        - Optimal temperature (°C): 17.34
        - Outside temperature (°C): -1.09
        - Current (A): 0.0
Consumption of other devices:
        - Current (A): 3.25
^[[2J^[[H
2024-03-08 03:22:46
                                              291981,1-8    27%
```

```
2024-03-08 03:22:45
Smart Energy Meter:
        - Current (A): 2.83
        - +A (kWh): 5.43
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): 0.0
        - SOC (%): 10.0
Electric Vehicle:
        - ID: 0
                - Is available: True
                - Driving power (kW): 0.0
                - Current (A): 0.02
                - SOC (%): 50.0
Photovoltaic:
        - Current (A): 0.0
Heating:
        - Current temperature (°C): 17.52
        - Optimal temperature (°C): 17.0
        - Outside temperature (°C): 0.4
        - Current (A): 0.0
Consumption of other devices:
        - Current (A): 2.81
^[[2J^[[H
2024-03-08 03:22:46
                                              291981,1-8    27%
```

```
2024-03-08 03:22:45
Smart Energy Meter:
        - Current (A): 15.92
        - +A (kWh): 11.65
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): 0.0
        - SOC (%): 10.0
Electric Vehicle:
        - ID: 0
                - Is available: True
                - Driving power (kW): 0.0
                - Current (A): 0.0
                - SOC (%): 50.0
Photovoltaic:
        - Current (A): 0.0
Heating:
        - Current temperature (°C): 17.91
        - Optimal temperature (°C): 17.0
        - Outside temperature (°C): 0.05
        - Current (A): 0.0
Consumption of other devices:
        - Current (A): 15.92
^[[2J^[[H
2024-03-08 03:22:46
                                              291981,1-8    27%
```

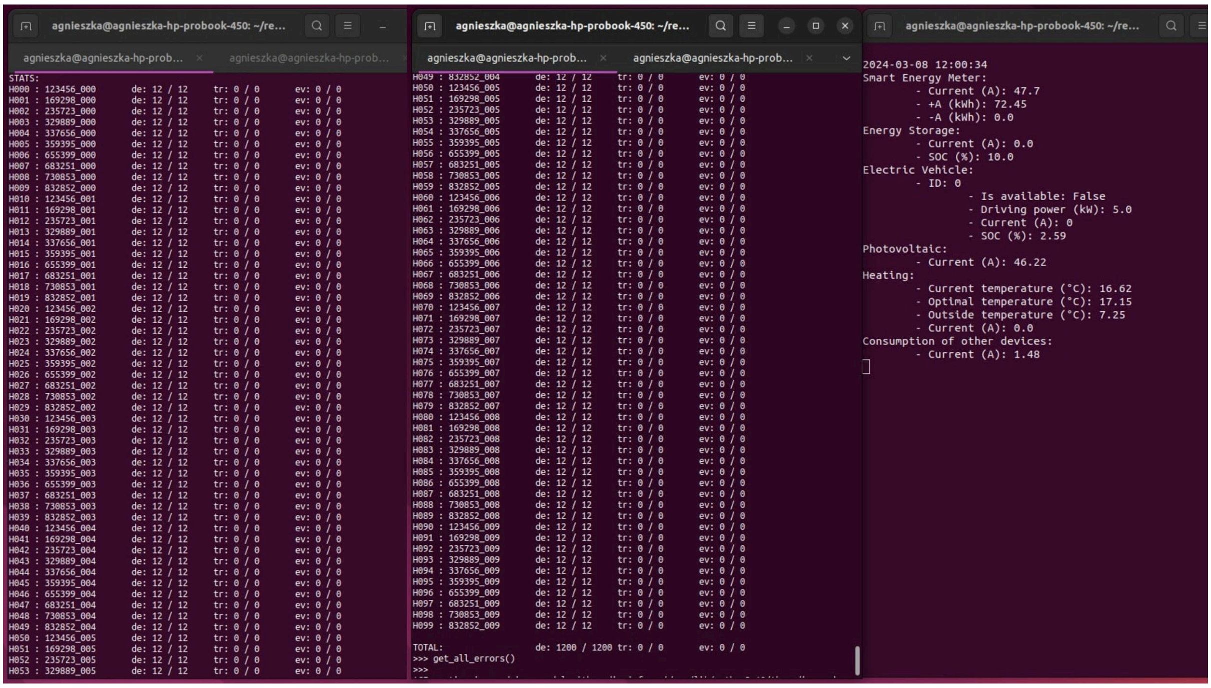
```
2024-03-08 03:22:45
Smart Energy Meter:
        - Current (A): 12.49
        - +A (kWh): 15.76
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): 0.0
        - SOC (%): 10.0
Photovoltaic:
        - Current (A): 0.0
Heating:
        - Current temperature (°C): 18.27
        - Optimal temperature (°C): 17.0
        - Outside temperature (°C): 0.89
        - Current (A): 0.0
Consumption of other devices:
        - Current (A): 12.49
^[[2J^[[H
2024-03-08 03:22:46
Smart Energy Meter:
        - Current (A): 12.49
        - +A (kWh): 15.77
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): 0.0
                                              218991,1      27%
```

```
2024-03-08 03:22:45
Smart Energy Meter:
        - Current (A): 10.62
        - +A (kWh): 13.5
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): 0.0
        - SOC (%): 10.0
Electric Vehicle:
        - ID: 0
                - Is available: True
                - Driving power (kW): 0.0
                - Current (A): 0.03
                - SOC (%): 50.01
Photovoltaic:
        - Current (A): 0.0
Heating:
        - Current temperature (°C): 18.74
        - Optimal temperature (°C): 17.0
        - Outside temperature (°C): 0.89
        - Current (A): 0.0
Consumption of other devices:
        - Current (A): 10.59
^[[2J^[[H
2024-03-08 03:22:46
                                              291981,1-8    27%
```

```
2024-03-08 03:22:45
Smart Energy Meter:
        - Current (A): 24.28
        - +A (kWh): 24.78
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): 0.0
        - SOC (%): 10.0
Electric Vehicle:
        - ID: 0
                - Is available: True
                - Driving power (kW): 0.0
                - Current (A): 16.14
                - SOC (%): 77.14
        - ID: 1
                - Is available: True
                - Driving power (kW): 0.0
                - Current (A): 0.04
                - SOC (%): 50.1
Photovoltaic:
        - Current (A): 0.0
Heating:
        - Current temperature (°C): 18.98
        - Optimal temperature (°C): 17.0
        - Outside temperature (°C): 0.59
                                              352806,1      27%
```

Figure 5.10: Examples of simulation logs from the test.

Figure 5.10 presents simulation logs for a subset of randomly selected households at the same time step influenced by the decisions illustrated in Figure 5.9. The observed differences in environmental states are a consequence of heterogeneous household configurations and individualized end-device management decisions produced by the AI models.


agnieszka@agnieszka-hp-probook-450: ~/re...
agnieszka@agnieszka-hp-prob...
agnieszka@agnieszka-hp-prob...

```
STATS:
H000 : 123456_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H001 : 169298_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H002 : 235723_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H003 : 329889_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H004 : 337656_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H005 : 359395_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H006 : 655399_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H007 : 683251_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H008 : 730853_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H009 : 832852_000      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H010 : 123456_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H011 : 169298_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H012 : 235723_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H013 : 329889_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H014 : 337656_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H015 : 359395_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H016 : 655399_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H017 : 683251_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H018 : 730853_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H019 : 832852_001      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H020 : 123456_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H021 : 169298_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H022 : 235723_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H023 : 329889_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H024 : 337656_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H025 : 359395_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H026 : 655399_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H027 : 683251_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H028 : 730853_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H029 : 832852_002      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H030 : 123456_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H031 : 169298_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H032 : 235723_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H033 : 329889_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H034 : 337656_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H035 : 359395_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H036 : 655399_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H037 : 683251_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H038 : 730853_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H039 : 832852_003      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H040 : 123456_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H041 : 169298_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H042 : 235723_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H043 : 329889_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H044 : 337656_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H045 : 359395_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H046 : 655399_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H047 : 683251_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H048 : 730853_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H049 : 832852_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H050 : 123456_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H051 : 169298_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H052 : 235723_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H053 : 329889_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
```

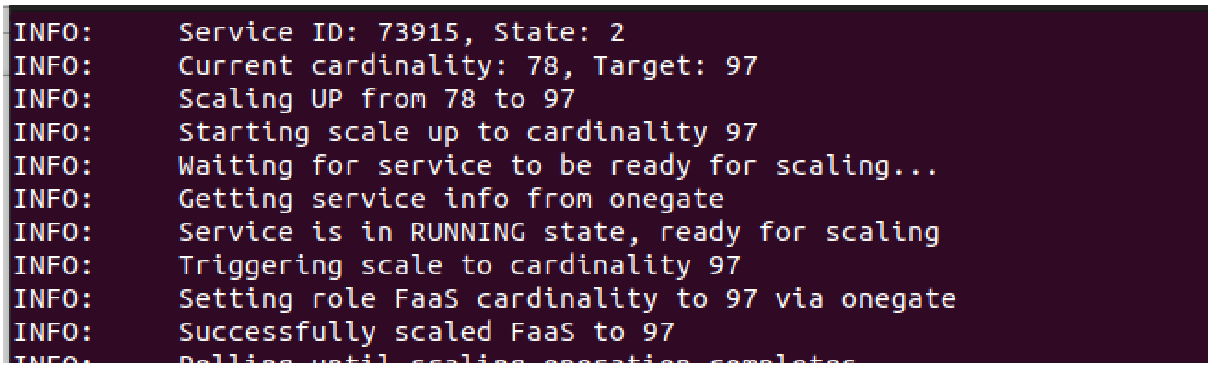
```
H049 : 832852_004      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H050 : 123456_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H051 : 169298_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H052 : 235723_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H053 : 329889_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H054 : 337656_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H055 : 359395_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H056 : 655399_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H057 : 683251_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H058 : 730853_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H059 : 832852_005      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H060 : 123456_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H061 : 169298_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H062 : 235723_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H063 : 329889_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H064 : 337656_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H065 : 359395_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H066 : 655399_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H067 : 683251_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H068 : 730853_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H069 : 832852_006      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H070 : 123456_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H071 : 169298_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H072 : 235723_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H073 : 329889_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H074 : 337656_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H075 : 359395_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H076 : 655399_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H077 : 683251_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H078 : 730853_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H079 : 832852_007      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H080 : 123456_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H081 : 169298_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H082 : 235723_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H083 : 329889_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H084 : 337656_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H085 : 359395_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H086 : 655399_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H087 : 683251_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H088 : 730853_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H089 : 832852_008      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H090 : 123456_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H091 : 169298_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H092 : 235723_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H093 : 329889_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H094 : 337656_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H095 : 359395_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H096 : 655399_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H097 : 683251_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H098 : 730853_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0
H099 : 832852_009      de: 12 / 12     tr: 0 / 0       ev: 0 / 0

TOTAL:                 de: 1200 / 1200 tr: 0 / 0       ev: 0 / 0
>>> get_all_errors()
>>>
```

```
2024-03-08 12:00:34
Smart Energy Meter:
        - Current (A): 47.7
        - +A (kWh): 72.45
        - -A (kWh): 0.0
Energy Storage:
        - Current (A): 0.0
        - SOC (%): 10.0
Electric Vehicle:
        - ID: 0
                - Is available: False
                - Driving power (kW): 5.0
                - Current (A): 0
                - SOC (%): 2.59
Photovoltaic:
        - Current (A): 46.22
Heating:
        - Current temperature (°C): 16.62
        - Optimal temperature (°C): 17.15
        - Outside temperature (°C): 7.25
        - Current (A): 0.0
Consumption of other devices:
        - Current (A): 1.48
```

Figure 5.11: Statistics logs from the test right before its termination, with an excerpt from simulation logs.

The simulation was terminated after 12 hours, as shown in Figure 5.11. This duration covered 12 hourly time steps for all 100 households, resulting in a total of 1,200 offloaded decision-making function executions.

All function executions were completed successfully, demonstrating that the COGNIT Infrastructure scaled correctly and provisioned a sufficient number of virtual machine instances with the required computational resources. Logs from the FaaS Frontend for the relevant deployment flavour, illustrating successful scaling under this test scenario, are presented in Figure 5.12.



```
INFO:     Service ID: 73915, State: 2
INFO:     Current cardinality: 78, Target: 97
INFO:     Scaling UP from 78 to 97
INFO:     Starting scale up to cardinality 97
INFO:     Waiting for service to be ready for scaling...
INFO:     Getting service info from onegate
INFO:     Service is in RUNNING state, ready for scaling
INFO:     Triggering scale to cardinality 97
INFO:     Setting role FaaS cardinality to 97 via onegate
INFO:     Successfully scaled FaaS to 97
INFO:     Polling until scaling operation completes
```

Figure 5.12: A snippet of the Edge Cluster Frontend logs for service with ID 73915 (Flavour EnergyTorch located in a cluster at Phoenix Systems) regarding scaling during the test.

5.4 Impact, added value and lessons learned

In the main demo, described in Section 5.2, we have shown and validated that the COGNIT Framework indeed can be used to enhance the computational potential of edge IoT devices with strictly constrained resources. In our case, it was providing the power of AI decision-making to a smart energy meter.

In the alternative scenario, described in Section 5.3, we have presented and tested the scalability capabilities of the COGNIT Framework. A significant number of concurrent offloading requests are handled well, providing a smooth end-user experience.

5.5 Video demonstration

Demonstration video of use-case 3 (Initial release August 2025, updated December 2025)

6. Use Case #4: Cybersecurity

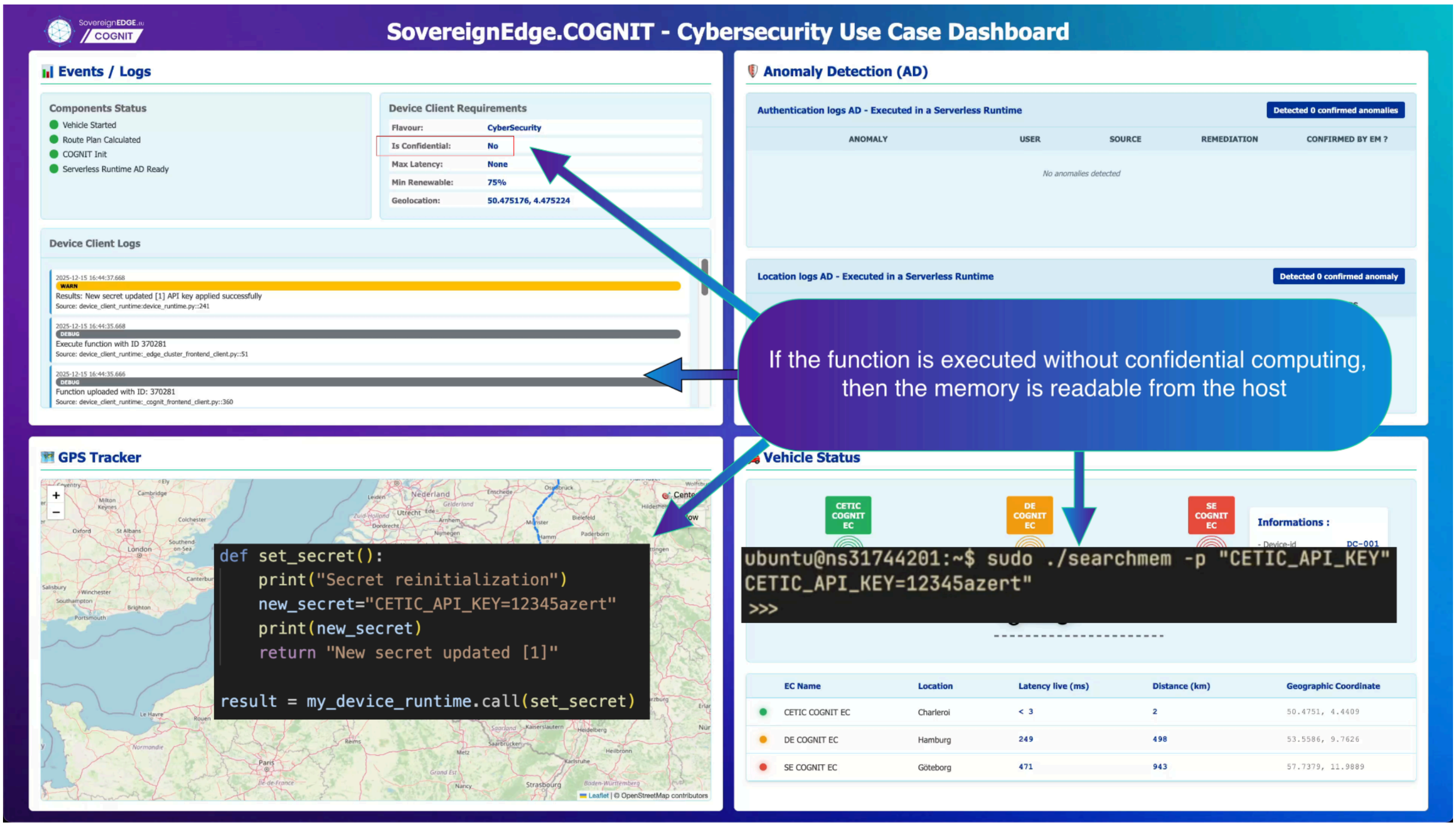
To enable real-time monitoring and visualization of the cybersecurity use case, we developed a dedicated web-based dashboard and deployed it in the RISE ICE infrastructure. The dashboard provides a single interface to observe Device Client status, track anomaly-detection outcomes, and monitor interactions between the vehicle and the COGNIT edge infrastructure (Edge Cluster selection, handovers, and Serverless Runtime activity). It was introduced specifically to make the "behind-the-scenes" orchestration and device behavior visible and auditable during demonstrations.

6.1 Scenario I Description - Confidential Computing (CC)

The Confidential Computing scenario demonstrates data-in-use protection with AMD SEV on a CC-capable edge node (cetic-node2-cc). The same serverless function is executed twice in otherwise identical conditions: first without confidential computing and then with it enabled. Host-side memory inspection illustrates the difference: plaintext is readable without CC and unreadable with CC. When the CC requirement (IS_CONFIDENTIAL) is set, the orchestrator restricts placement to CC-capable infrastructure and prepares the appropriate SR flavour. With the flag unset, the function runs on a standard path.

6.2 Demo I Flow

1. Device Client Requirement "IS_CONFIDENTIAL" is set to False.
2. Function is offloaded and executed in a Serverless Runtime on a non-CC edge node. "CETIC_API_KEY" as a secret is set in the Serverless Runtime VM process.
3. From the host, the memory-scanning script targeting the Serverless Runtime VM is run. The value of the "CETIC_API_KEY" is retrieved.
4. Device Client Requirement "IS_CONFIDENTIAL" is set to True.
5. Function is offloaded and executed in a Serverless Runtime on cetic-node2-cc edge node, with CC capabilities. "CETIC_API_KEY" as a secret is set in the Serverless Runtime VM process.
6. From the host (cetic-node2-cc), the memory-scanning script targeting the Serverless Runtime VM is run. The value of the "CETIC_API_KEY" is not retrieved.
7. The execution/startup differences and dashboard data for both executions are captured.



Components Status
- Vehicle Started
- Route Plan Calculated
- COGNIT Init
- Serverless Runtime AD Ready

Device Client Requirements

Flavour:	CyberSecurity
Is Confidential:	No
Max Latency:	None
Min Renewable:	75%
Geolocation:	50.475176, 4.475224

Device Client Logs

```
2025-12-15 16:44:37.668
WARN
Results: New secret updated [1] API key applied successfully
Source: device_client_runtime:device_runtime.py::241

2025-12-15 16:44:35.668
DEBUG
Execute function with ID 370281
Source: device_client_runtime:_edge_cluster_frontend_client.py::51

2025-12-15 16:44:35.666
DEBUG
Function uploaded with ID: 370281
Source: device_client_runtime:_cognit_frontend_client.py::360
```

Authentication logs AD - Executed in a Serverless Runtime — Detected 0 confirmed anomalies

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?
No anomalies detected				

Location logs AD - Executed in a Serverless Runtime — Detected 0 confirmed anomaly

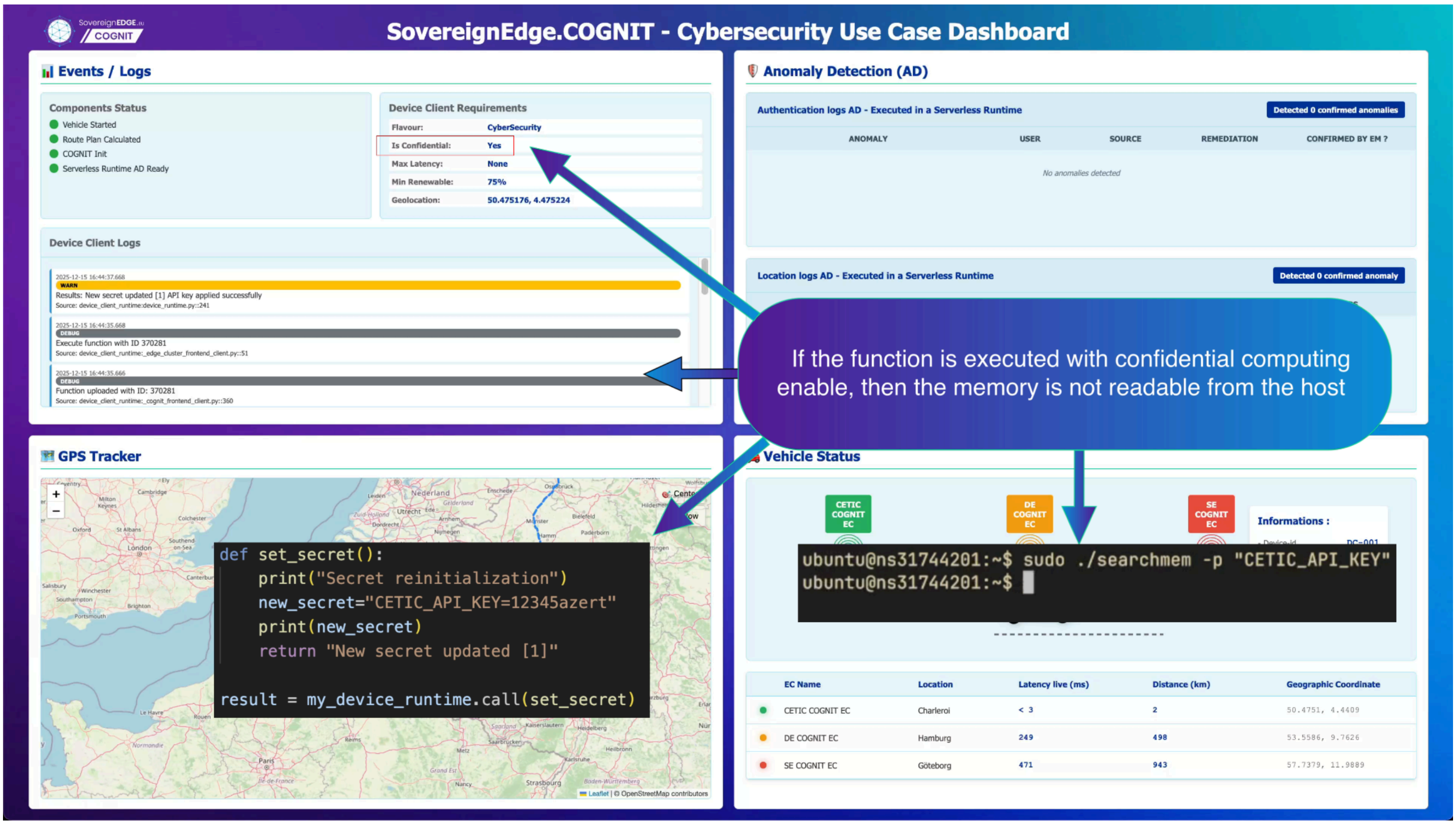
If the function is executed without confidential computing, then the memory is readable from the host

```python
def set_secret():
    print("Secret reinitialization")
    new_secret="CETIC_API_KEY=12345azert"
    print(new_secret)
    return "New secret updated [1]"

result = my_device_runtime.call(set_secret)
```

```
ubuntu@ns31744201:~$ sudo ./searchmem -p "CETIC_API_KEY"
CETIC_API_KEY=12345azert"
>>>
```

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
🟢	CETIC COGNIT EC	Charleroi	< 3	2	50.4751, 4.4409
🟠	DE COGNIT EC	Hamburg	249	498	53.5586, 9.7626
🔴	SE COGNIT EC	Göteborg	471	943	57.7379, 11.9889

Figure 6.1: Confidential computing disabled



Components Status
- Vehicle Started
- Route Plan Calculated
- COGNIT Init
- Serverless Runtime AD Ready

Device Client Requirements

Flavour:	CyberSecurity
Is Confidential:	Yes
Max Latency:	None
Min Renewable:	75%
Geolocation:	50.475176, 4.475224

Device Client Logs

```
2025-12-15 16:44:37.668
WARN
Results: New secret updated [1] API key applied successfully
Source: device_client_runtime:device_runtime.py::241

2025-12-15 16:44:35.668
DEBUG
Execute function with ID 370281
Source: device_client_runtime:_edge_cluster_frontend_client.py::51

2025-12-15 16:44:35.666
DEBUG
Function uploaded with ID: 370281
Source: device_client_runtime:_cognit_frontend_client.py::360
```

Authentication logs AD - Executed in a Serverless Runtime — Detected 0 confirmed anomalies

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?
No anomalies detected				

Location logs AD - Executed in a Serverless Runtime — Detected 0 confirmed anomaly

If the function is executed with confidential computing enable, then the memory is not readable from the host

```python
def set_secret():
    print("Secret reinitialization")
    new_secret="CETIC_API_KEY=12345azert"
    print(new_secret)
    return "New secret updated [1]"

result = my_device_runtime.call(set_secret)
```

```
ubuntu@ns31744201:~$ sudo ./searchmem -p "CETIC_API_KEY"
ubuntu@ns31744201:~$ ▌
```

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
🟢	CETIC COGNIT EC	Charleroi	< 3	2	50.4751, 4.4409
🟠	DE COGNIT EC	Hamburg	249	498	53.5586, 9.7626
🔴	SE COGNIT EC	Göteborg	471	943	57.7379, 11.9889

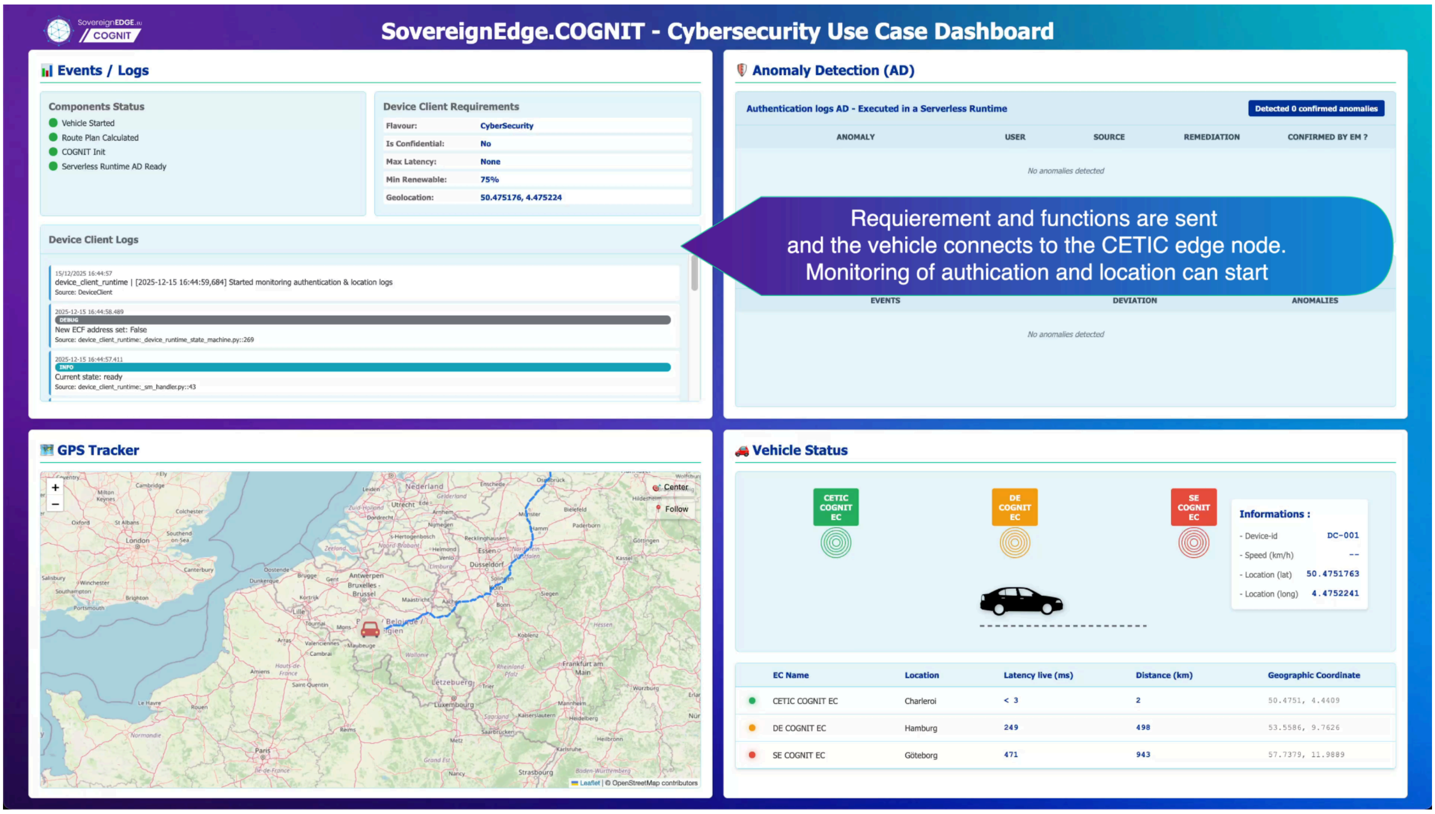
Figure 6.2: Confidential computing enabled

6.3 Scenario II Description - Anomaly Detection with Mobility

The Anomaly Detection with Mobility scenario demonstrates the edge-centric cybersecurity pipeline for a moving device. A rover-simulator (Device Client) authenticates to the COGNIT Frontend and offloads an Anomaly Detection function to a Serverless Runtime on the currently selected Edge Cluster. The proximity is evaluated against the latency of the pre-placed and other available SRs to select the edge node with the lowest latency. The Device Client switches to the nearest Edge Cluster on the next invocation. Geographically proximate clusters emulate credible Response times without relying on a live 5G layer. During the “journey,” the Anomaly Detection processes authentication/GPS events. Detections trigger closed-loop remediation on the device (e.g., temporary/permanent SSH blocks, IP-specific or global). The dashboard provides situational awareness (initialization/logs, anomalies, route/EC status).

6.4 Demo II Flow

1. The rover simulator is started and authenticated with the COGNIT Frontend.
2. Device Client Requirements (FLAVOUR, IS_CONFIDENTIAL, GEOLOCATION) are set, and the initialization function is offloaded and executed. Monitoring of authentication and geolocation starts. The rover begins its journey.
3. The initial Edge Cluster is defined, and the Serverless runtime is ready to execute the functions.
4. Anomaly Detection - Authentication
    - 4.1. A login attempt is simulated. An entry appears in the authentication log (auth.log). This entry is captured and passed as a parameter to the function that is executed in the serverless runtime.
    - 4.2. The result is received by the Device Client, and a first response is provided. User john.doe is blocked for 1 hour.
    - 4.3. The anomaly is then sent to the Embedding model for further analysis and confirmation. The anomaly is confirmed by the Embedding model. (More details can be found in D5.6.)
5. Anomaly Detection - Location
    - 5.1. A spoofing attempt is simulated, tricking the device into believing it is in New York. The location is captured and passed as a parameter to a function executed in the Serverless Runtime.
    - 5.2. The result is received by the Device Client, and a first response is provided. The Device Client then switches to a different geolocation mechanism.
6. Mobility - Edge cluster switching
    - 6.1. The rover moves away from the CETIC edge cluster (cetic-node2-cc), it will switch over and connect to the DE edge cluster (p02r11srv15)
    - 6.2. The vehicle gradually moves away from the DE edge cluster, it switches and connects to the SE edge cluster (p02r11srv01)



SovereignEdge.COGNIT - Cybersecurity Use Case Dashboard

Events / Logs

Components Status

- ● Vehicle Started
- ● Route Plan Calculated
- ● COGNIT Init
- ● Serverless Runtime AD Ready

Device Client Requirements

Flavour:	CyberSecurity
Is Confidential:	No
Max Latency:	None
Min Renewable:	75%
Geolocation:	50.475176, 4.475224

Device Client Logs

```
15/12/2025 16:44:57
device_client_runtime | [2025-12-15 16:44:59,684] Started monitoring authentication & location logs
Source: DeviceClient

2025-12-15 16:44:58.489
DEBUG
New ECF address set: False
Source: device_client_runtime:_device_runtime_state_machine.py::269

2025-12-15 16:44:57.411
INFO
Current state: ready
Source: device_client_runtime:_sm_handler.py::43
```

Anomaly Detection (AD)

Authentication logs AD - Executed in a Serverless Runtime — Detected 0 confirmed anomalies

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?

No anomalies detected

EVENTS	DEVIATION	ANOMALIES

No anomalies detected

Requierement and functions are sent and the vehicle connects to the CETIC edge node. Monitoring of authication and location can start

GPS Tracker

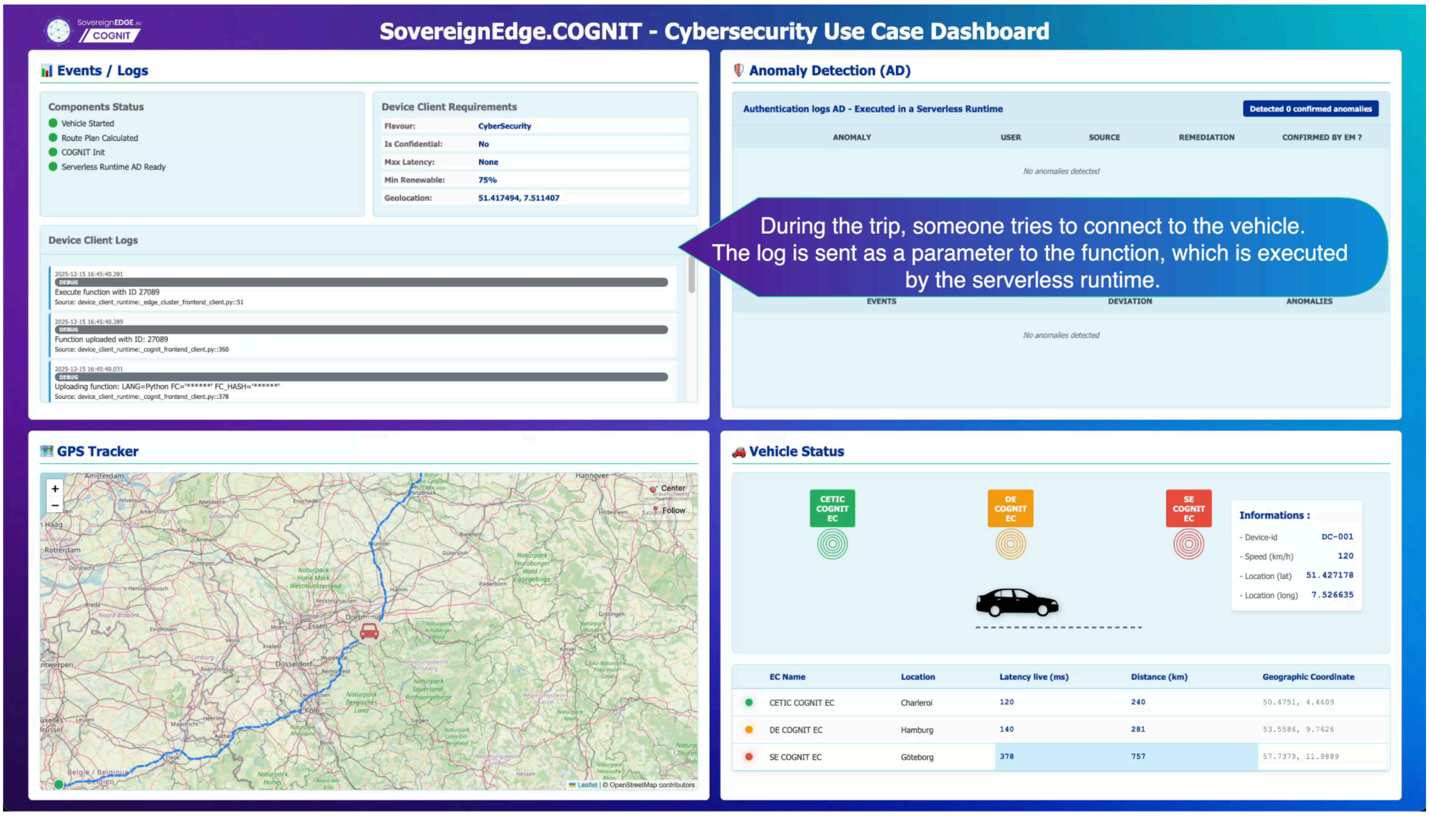
Vehicle Status

Informations :

- Device-id	DC-001
- Speed (km/h)	--
- Location (lat)	50.4751763
- Location (long)	4.4752241

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
●	CETIC COGNIT EC	Charleroi	< 3	2	50.4751, 4.4409
●	DE COGNIT EC	Hamburg	249	498	53.5586, 9.7626
●	SE COGNIT EC	Göteborg	471	943	57.7379, 11.9889

Figure 6.3: Running. Steps 1, 2 and 3



SovereignEdge.COGNIT - Cybersecurity Use Case Dashboard

Events / Logs

Components Status

- ● Vehicle Started
- ● Route Plan Calculated
- ● COGNIT Init
- ● Serverless Runtime AD Ready

Device Client Requirements

Flavour:	CyberSecurity
Is Confidential:	No
Max Latency:	None
Min Renewable:	75%
Geolocation:	51.417494, 7.511407

Device Client Logs

```
2025-12-15 16:45:40.291
DEBUG
Execute function with ID 27089
Source: device_client_runtime:_edge_cluster_frontend_client.py::51

2025-12-15 16:45:40.289
DEBUG
Function uploaded with ID: 27089
Source: device_client_runtime:_cognit_frontend_client.py::360

2025-12-15 16:45:40.031
DEBUG
Uploading function: LANG=Python FC='******' FC_HASH='******'
Source: device_client_runtime:_cognit_frontend_client.py::378
```

Anomaly Detection (AD)

Authentication logs AD - Executed in a Serverless Runtime — Detected 0 confirmed anomalies

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?

No anomalies detected

EVENTS	DEVIATION	ANOMALIES

No anomalies detected

During the trip, someone tries to connect to the vehicle. The log is sent as a parameter to the function, which is executed by the serverless runtime.

GPS Tracker

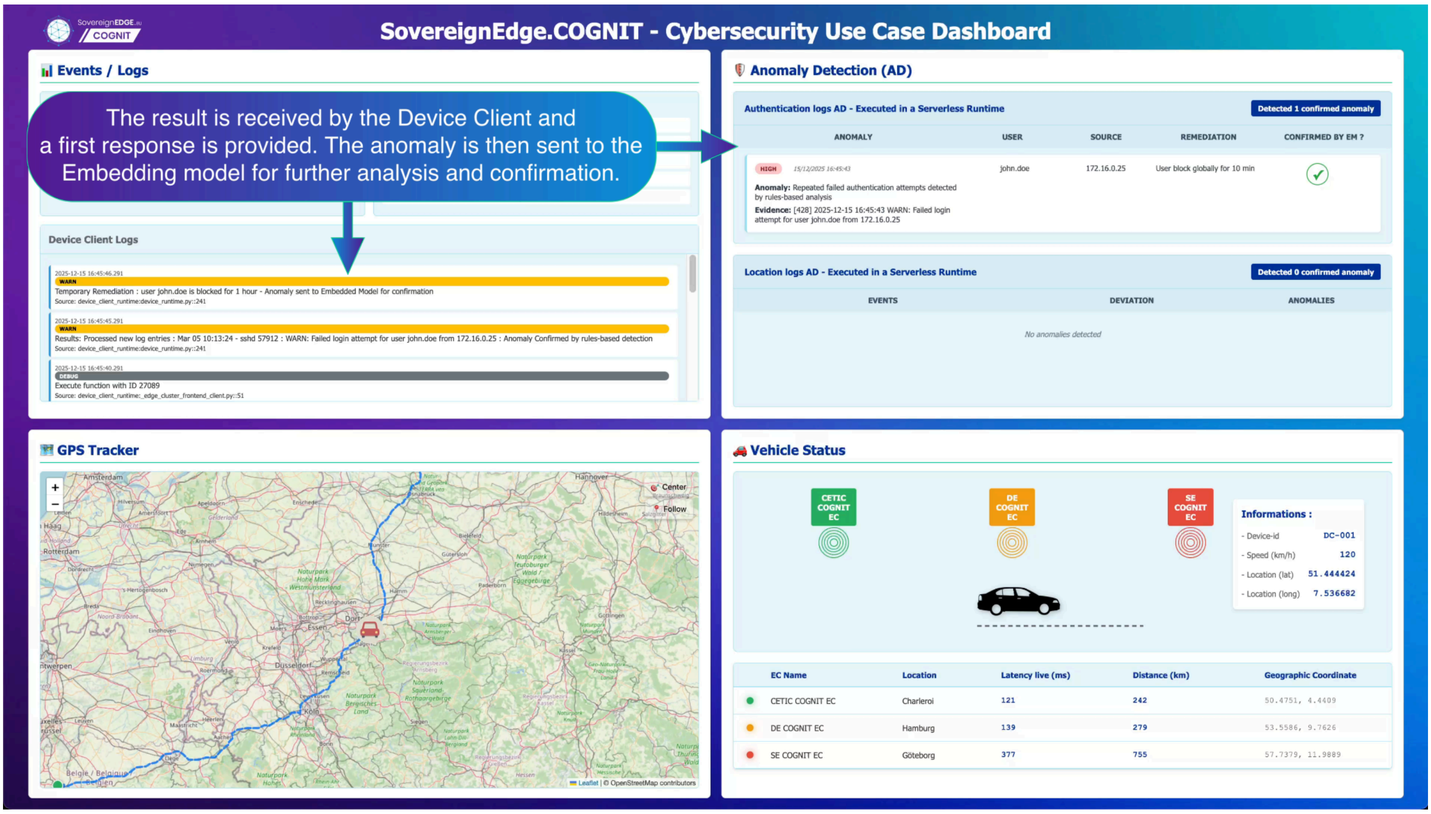
Vehicle Status

Informations :

- Device-id	DC-001
- Speed (km/h)	120
- Location (lat)	51.427178
- Location (long)	7.526635

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
●	CETIC COGNIT EC	Charleroi	120	240	50.4751, 4.4409
●	DE COGNIT EC	Hamburg	140	281	53.5586, 9.7626
●	SE COGNIT EC	Göteborg	378	757	57.7379, 11.9889

Figure 6.4: Connection attempt - attack simulation. Step 4.1



SovereignEdge.COGNIT - Cybersecurity Use Case Dashboard

Events / Logs

The result is received by the Device Client and a first response is provided. The anomaly is then sent to the Embedding model for further analysis and confirmation.

Device Client Logs

```
2025-12-15 16:45:46.291
WARN
Temporary Remediation : user john.doe is blocked for 1 hour - Anomaly sent to Embedded Model for confirmation
Source: device_client_runtime:device_runtime.py::241

2025-12-15 16:45:45.291
WARN
Results: Processed new log entries : Mar 05 10:13:24 - sshd 57912 : WARN: Failed login attempt for user john.doe from 172.16.0.25 : Anomaly Confirmed by rules-based detection
Source: device_client_runtime:device_runtime.py::241

2025-12-15 16:45:40.291
DEBUG
Execute function with ID 27089
Source: device_client_runtime:_edge_cluster_frontend_client.py::51
```

Anomaly Detection (AD)

Authentication logs AD - Executed in a Serverless Runtime

Detected 1 confirmed anomaly

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?
HIGH 15/12/2025 16:45:43 **Anomaly:** Repeated failed authentication attempts detected by rules-based analysis **Evidence:** [428] 2025-12-15 16:45:43 WARN: Failed login attempt for user john.doe from 172.16.0.25	john.doe	172.16.0.25	User block globally for 10 min	✓

Location logs AD - Executed in a Serverless Runtime

Detected 0 confirmed anomaly

EVENTS	DEVIATION	ANOMALIES

No anomalies detected

GPS Tracker

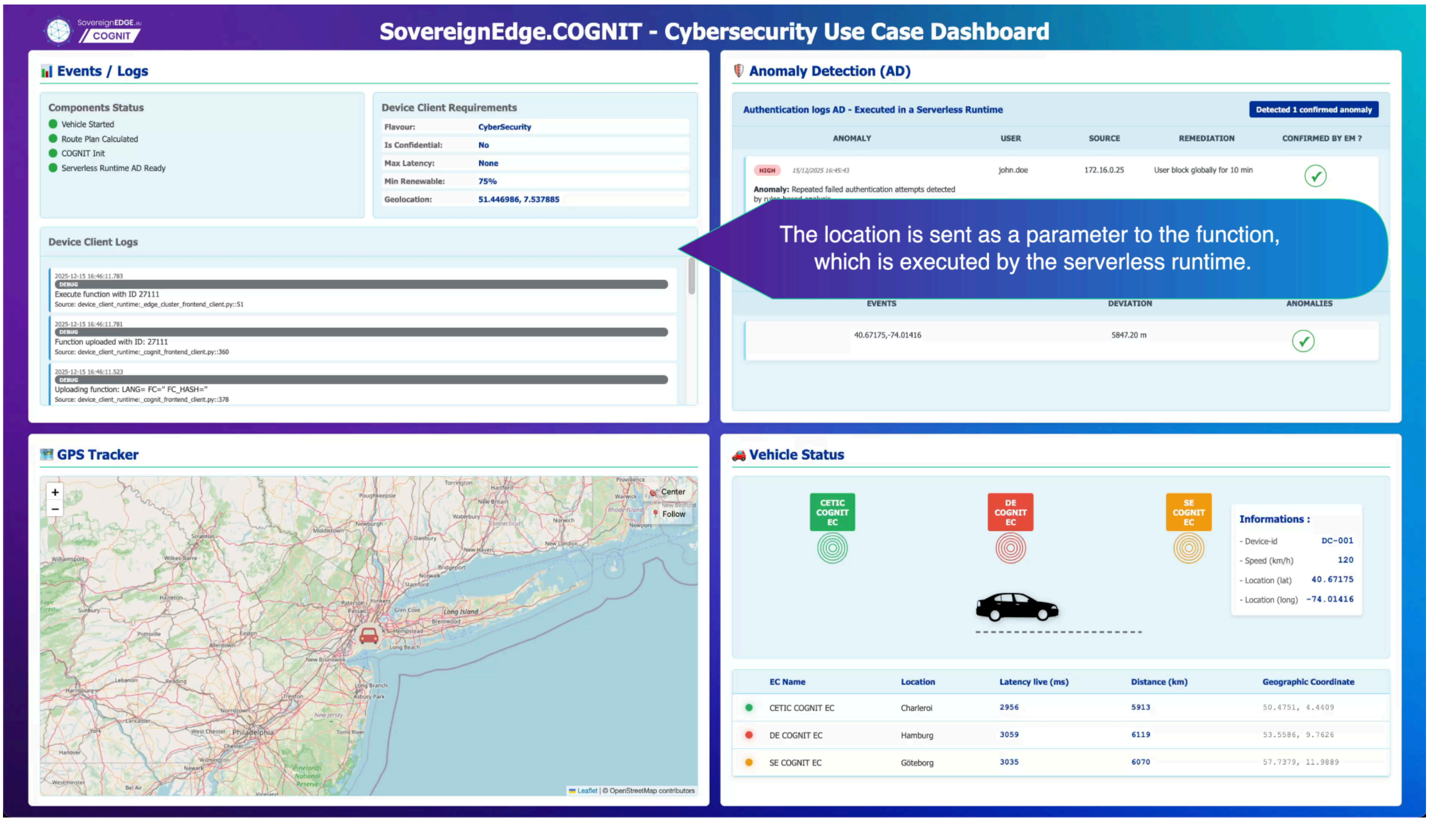
Vehicle Status

Informations :

- Device-id	DC-001
- Speed (km/h)	120
- Location (lat)	51.444424
- Location (long)	7.536682

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
●	CETIC COGNIT EC	Charleroi	121	242	50.4751, 4.4409
●	DE COGNIT EC	Hamburg	139	279	53.5586, 9.7626
●	SE COGNIT EC	Göteborg	377	755	57.7379, 11.9889

Figure 6.5: Connection attempt - attack simulation - result and remediation. Step 4.2 & 4.3



SovereignEdge.COGNIT - Cybersecurity Use Case Dashboard

Events / Logs

Components Status

- ● Vehicle Started
- ● Route Plan Calculated
- ● COGNIT Init
- ● Serverless Runtime AD Ready

Device Client Requirements

Flavour:	CyberSecurity
Is Confidential:	No
Max Latency:	None
Min Renewable:	75%
Geolocation:	51.446986, 7.537885

Device Client Logs

```
2025-12-15 16:46:11.783
DEBUG
Execute function with ID 27111
Source: device_client_runtime:_edge_cluster_frontend_client.py::51

2025-12-15 16:46:11.781
DEBUG
Function uploaded with ID: 27111
Source: device_client_runtime:_cognit_frontend_client.py::360

2025-12-15 16:46:11.523
DEBUG
Uploading function: LANG= FC=" FC_HASH="
Source: device_client_runtime:_cognit_frontend_client.py::378
```

Anomaly Detection (AD)

Authentication logs AD - Executed in a Serverless Runtime — Detected 1 confirmed anomaly

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?
HIGH 15/12/2025 16:45:43 **Anomaly:** Repeated failed authentication attempts detected by rules-based analysis	john.doe	172.16.0.25	User block globally for 10 min	✓

The location is sent as a parameter to the function, which is executed by the serverless runtime.

EVENTS	DEVIATION	ANOMALIES
40.67175,-74.01416	5847.20 m	✓

GPS Tracker

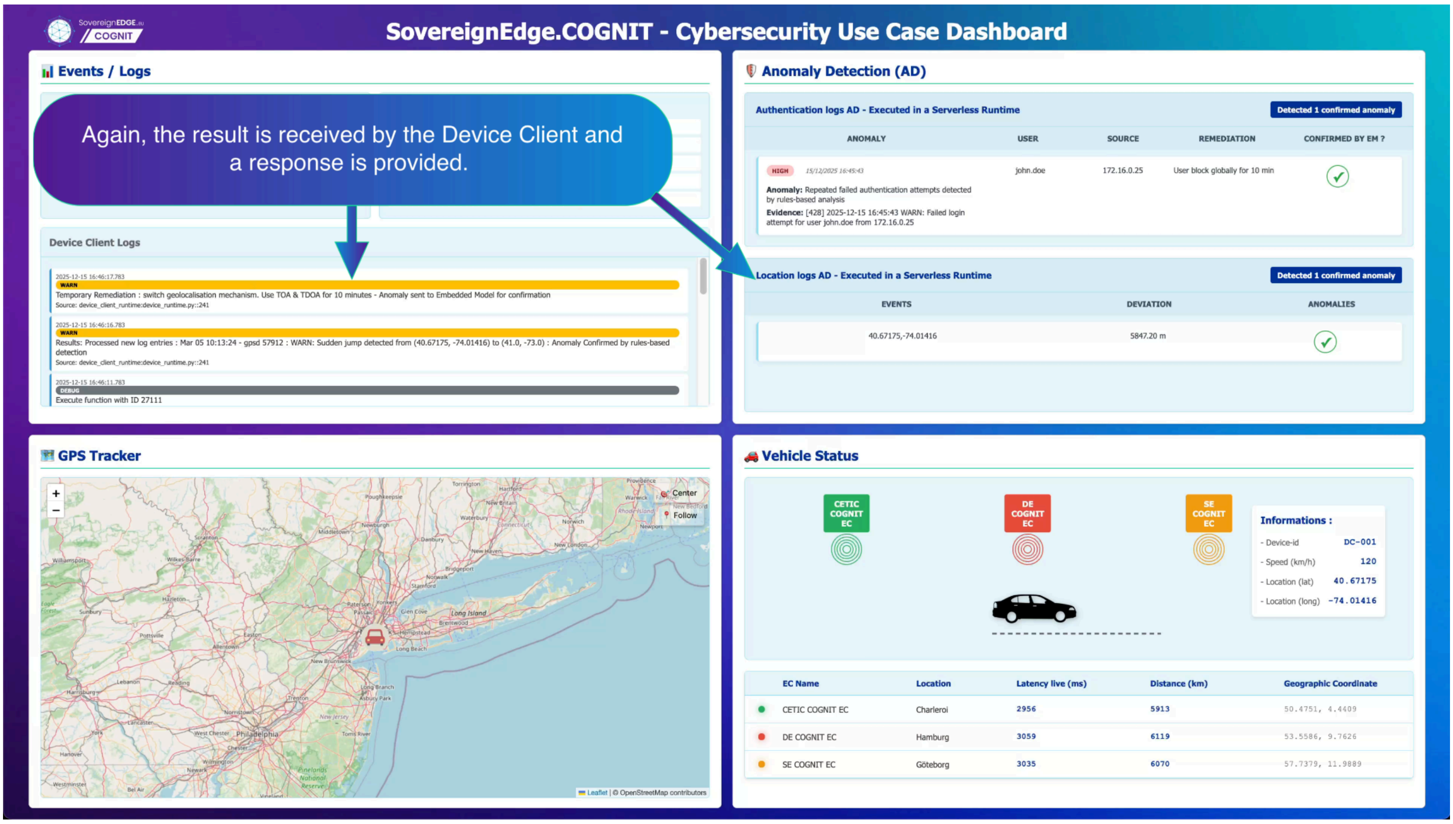
Vehicle Status

Informations :

- Device-id	DC-001
- Speed (km/h)	120
- Location (lat)	40.67175
- Location (long)	-74.01416

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
●	CETIC COGNIT EC	Charleroi	2956	5913	50.4751, 4.4409
●	DE COGNIT EC	Hamburg	3059	6119	53.5586, 9.7626
●	SE COGNIT EC	Göteborg	3035	6070	57.7379, 11.9889

Figure 6.6: Spoofing - attack simulation. Step 5.1



SovereignEdge.COGNIT - Cybersecurity Use Case Dashboard

Events / Logs

Again, the result is received by the Device Client and a response is provided.

Device Client Logs

```
2025-12-15 16:46:17.783
WARN
Temporary Remediation : switch geolocalisation mechanism. Use TOA & TDOA for 10 minutes - Anomaly sent to Embedded Model for confirmation
Source: device_client_runtime:device_runtime.py::241

2025-12-15 16:46:16.783
WARN
Results: Processed new log entries : Mar 05 10:13:24 - gpsd 57912 : WARN: Sudden jump detected from (40.67175, -74.01416) to (41.0, -73.0) : Anomaly Confirmed by rules-based detection
Source: device_client_runtime:device_runtime.py::241

2025-12-15 16:46:11.783
DEBUG
Execute function with ID 27111
```

Anomaly Detection (AD)

Authentication logs AD - Executed in a Serverless Runtime — Detected 1 confirmed anomaly

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?
HIGH 15/12/2025 16:45:43 **Anomaly:** Repeated failed authentication attempts detected by rules-based analysis **Evidence:** [428] 2025-12-15 16:45:43 WARN: Failed login attempt for user john.doe from 172.16.0.25	john.doe	172.16.0.25	User block globally for 10 min	✓

Location logs AD - Executed in a Serverless Runtime — Detected 1 confirmed anomaly

EVENTS	DEVIATION	ANOMALIES
40.67175,-74.01416	5847.20 m	✓

GPS Tracker

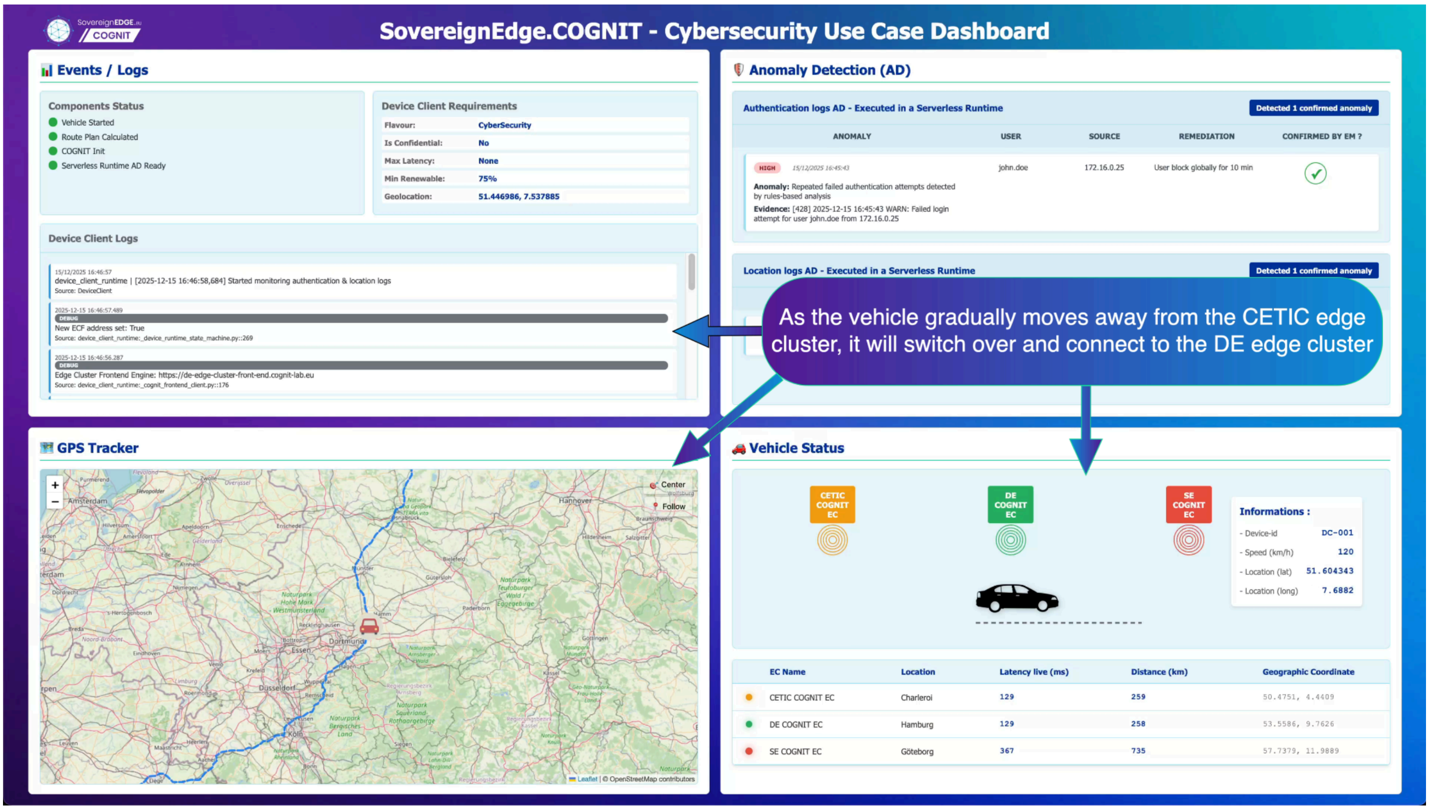
Vehicle Status

Informations :

- Device-id	DC-001
- Speed (km/h)	120
- Location (lat)	40.67175
- Location (long)	-74.01416

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
●	CETIC COGNIT EC	Charleroi	2956	5913	50.4751, 4.4409
●	DE COGNIT EC	Hamburg	3059	6119	53.5586, 9.7626
●	SE COGNIT EC	Göteborg	3035	6070	57.7379, 11.9889

Figure 6.7: Spoofing - attack simulation - result and remediation. Step 5.2



SovereignEdge.COGNIT - Cybersecurity Use Case Dashboard

Events / Logs

Components Status
- ● Vehicle Started
- ● Route Plan Calculated
- ● COGNIT Init
- ● Serverless Runtime AD Ready

Device Client Requirements

Flavour:	CyberSecurity
Is Confidential:	No
Max Latency:	None
Min Renewable:	75%
Geolocation:	51.446986, 7.537885

Device Client Logs

```
15/12/2025 16:46:57
device_client_runtime | [2025-12-15 16:46:58,684] Started monitoring authentication & location logs
Source: DeviceClient

2025-12-15 16:46:57.489
DEBUG
New ECF address set: True
Source: device_client_runtime:_device_runtime_state_machine.py::269

2025-12-15 16:46:56.287
DEBUG
Edge Cluster Frontend Engine: https://de-edge-cluster-front-end.cognit-lab.eu
Source: device_client_runtime:_cognit_frontend_client.py::176
```

Anomaly Detection (AD)

Authentication logs AD - Executed in a Serverless Runtime — Detected 1 confirmed anomaly

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?
HIGH 15/12/2025 16:45:43 **Anomaly:** Repeated failed authentication attempts detected by rules-based analysis **Evidence:** [428] 2025-12-15 16:45:43 WARN: Failed login attempt for user john.doe from 172.16.0.25	john.doe	172.16.0.25	User block globally for 10 min	✓

Location logs AD - Executed in a Serverless Runtime — Detected 1 confirmed anomaly

As the vehicle gradually moves away from the CETIC edge cluster, it will switch over and connect to the DE edge cluster

GPS Tracker

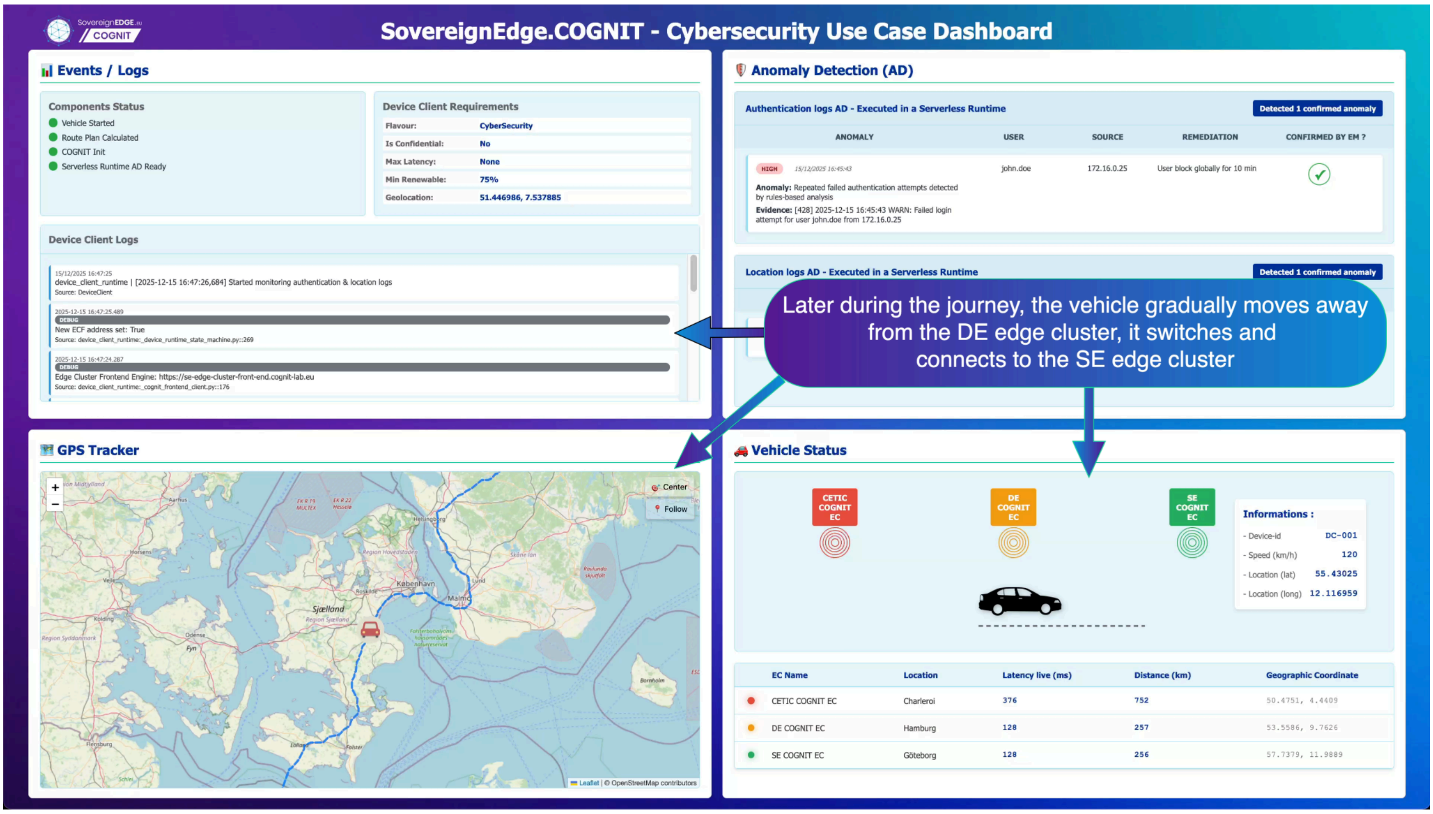
Vehicle Status

Informations :

- Device-id	DC-001
- Speed (km/h)	120
- Location (lat)	51.604343
- Location (long)	7.6882

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
●	CETIC COGNIT EC	Charleroi	129	259	50.4751, 4.4409
●	DE COGNIT EC	Hamburg	129	258	53.5586, 9.7626
●	SE COGNIT EC	Göteborg	367	735	57.7379, 11.9889

Figure 6.8: Edge cluster switching - CETIC to DE. Step 6.1



SovereignEdge.COGNIT - Cybersecurity Use Case Dashboard

Events / Logs

Components Status
- ● Vehicle Started
- ● Route Plan Calculated
- ● COGNIT Init
- ● Serverless Runtime AD Ready

Device Client Requirements

Flavour:	CyberSecurity
Is Confidential:	No
Max Latency:	None
Min Renewable:	75%
Geolocation:	51.446986, 7.537885

Device Client Logs

```
15/12/2025 16:47:25
device_client_runtime | [2025-12-15 16:47:26,684] Started monitoring authentication & location logs
Source: DeviceClient

2025-12-15 16:47:25.489
DEBUG
New ECF address set: True
Source: device_client_runtime:_device_runtime_state_machine.py::269

2025-12-15 16:47:24.287
DEBUG
Edge Cluster Frontend Engine: https://se-edge-cluster-front-end.cognit-lab.eu
Source: device_client_runtime:_cognit_frontend_client.py::176
```

Anomaly Detection (AD)

Authentication logs AD - Executed in a Serverless Runtime — Detected 1 confirmed anomaly

ANOMALY	USER	SOURCE	REMEDIATION	CONFIRMED BY EM ?
HIGH 15/12/2025 16:45:43 **Anomaly:** Repeated failed authentication attempts detected by rules-based analysis **Evidence:** [428] 2025-12-15 16:45:43 WARN: Failed login attempt for user john.doe from 172.16.0.25	john.doe	172.16.0.25	User block globally for 10 min	✓

Location logs AD - Executed in a Serverless Runtime — Detected 1 confirmed anomaly

Later during the journey, the vehicle gradually moves away from the DE edge cluster, it switches and connects to the SE edge cluster

GPS Tracker

Vehicle Status

Informations :

- Device-id	DC-001
- Speed (km/h)	120
- Location (lat)	55.43025
- Location (long)	12.116959

	EC Name	Location	Latency live (ms)	Distance (km)	Geographic Coordinate
●	CETIC COGNIT EC	Charleroi	376	752	50.4751, 4.4409
●	DE COGNIT EC	Hamburg	128	257	53.5586, 9.7626
●	SE COGNIT EC	Göteborg	128	256	57.7379, 11.9889

Figure 6.9: Edge cluster switching - CETIC to SE. Step 6.2

6.5 Impact, added value and lessons learned

The demonstrations validate a coherent multi-edge model in which the Device Client offloads functions to Serverless Runtimes while COGNIT performs a switch on next-call handovers. We showed end-to-end anomaly detection with closed-loop remediation during mobility and executed the same workload with and without confidential computing (AMD SEV), evidencing data-in-use protection on compatible hardware. This kept device logic lightweight yet responsive, with decisions and outcomes made auditable through the dashboard.

COGNIT's added value lies in the combination of edge-appropriate placement, explicit edge requirements and a portable execution abstraction that moves functions between devices without application modification. Enabling confidential computing follows the same orchestration path. The key takeaways demonstrate a lightweight, two-step Anomaly detection chain (decision tree + embedded model) that adapts to edge constraints. Gravity-based remediation can be safely automated at the edge of the network. Confidential Computing significantly enhances edge trust. Overall, UC4 is validated at TRL 5 and is readily transferable to domains requiring low-latency analytics, controlled transfers, and data protection during use.

6.6 Video of demonstrator

Demonstration video of use-case 4 (Initial release August 2025, updated December 2025).

7. Conclusion

This document demonstrates the capabilities of the final release of the COGNIT Framework, related to the final COGNIT Architecture (presented in D2.6). The capabilities are demonstrated through the four different use cases, which together highlight the flexibility of function offloading to expand the functionality available for resource-constrained devices, the scalability of the framework and the optimisation of the Serverless Runtime workloads.