

NEMO

Next Generation Meta Operating System

D5.4 NEMO Living Labs use cases evaluation results - Final version

Document Identification			
Status	Final	Due Date	31/08/2025
Version	6.00	Submission Date	15/09/2025

Related WP	WP5	Document Reference	D5.4
Related Deliverable(s)	D1.1, D1.3, D4.2, D4.3, D5.1, D5.2, D5.3	Dissemination Level (*)	PU
Lead Participant	ASM	Lead Author	Prashanth Kumar (ASM)
Contributors	ESOFT, SYN, ASM, EMOT, COMS, SIM, CONTI, FHW, MAG, NOVO, UPM, RWTH, WIND	Reviewers	Emmanouil Bakiris (SPACE)

Keywords:
NEMO, trial, living lab, pilot, validation, Final results

Disclaimer for Deliverables with dissemination level PUBLIC

This document is issued within the frame and for the purpose of the NEMO project. This project has received funding from the European Union's Horizon Europe Framework Program under Grant Agreement No. 101070118. The opinions expressed and arguments employed herein do not necessarily reflect the official views of the European Commission.

The dissemination of this document reflects only the author's view, and the European Commission is not responsible for any use that may be made of the information it contains. This deliverable is subject to final acceptance by the European Commission.

This document and its content are the property of the NEMO Consortium. The content of all or parts of this document can be used and distributed provided that the NEMO project and the document are properly referenced.

Each NEMO Partner may use this document in conformity with the NEMO Consortium Grant Agreement provisions.

Document Information

List of Contributors	
Name	Partner
Mircea Vasile	ASM
Andreea Paunescu	SIM
Costas Vrioni	MAG
Mika Skarp	CMC
Sonja Waechter	CONTI
Antonis Gonos	ESOFT
Prashanth Kumar Pedholla	ASM
Mohammad Ghoreishi	ASM
Mostafa Jabari	ASM
Maria Belesioti	OTE
Dimitrios Christopoulos	FHW
George Sofianopoulos	FHW
Theo Kakardakos	NOVO
Christopher Caponi	EMOT
Ilias Nektarios Seitanidis	SYN
Theodore Zahariadis	SYN
Matija Cankar	COMS
Denis Sodin	COMS
Javier Serrano	UPM
Alberto del Rio	UPM
Martin Kröning	RWTH
Jonathan Klimt	RWTH

Document History			
Version	Date	Change editors	Changes
1.0	05/06/2025	Prashanth Kumar. P	Table of Contents
2.0	18/07/2025	Prashanth Kumar. P, Mohammad Ghoreishi, Mostafa Jabari, Dimitrios Christopoulos, George Sofianopoulos, Antonis Gonos, Theodore Zahariadis, Ilias Nektarios Seitanidis, Mircea Vasile, Andreea Paunescu, Christopher Caponi, Javier Serrano, Alberto del Rio, Maria Belesioti	Contributions to Smart Farming Trial, Smart Energy & Mobility Trial, Smart Media & XR Trial
3.0	29/08/2025	Prashanth Kumar. P, Mohammad Ghoreishi, Mostafa Jabari, Dimitrios Christopoulos, Theo Kakardakos, Matija Cankar, Denis Sodin, Christopher Caponi, Costas Vrioni, Sonja Waechter, Maria Belesioti	Contributions to Smart Energy & Mobility Trial, Smart Media & XR Trial

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	2 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

4.0	09/09/2025	Prashanth Kumar. P	Addressing review comments
5.0	12/09/2025	Antonios Gonos, Martin Kröning, Jonathan Klimt, Francesco Bellesini, Mohammad Ghoreishi	Addressing review comments
6.0	15/09/2025	ATOS	Quality review and submission to EC a

Quality Control		
Role	Name (Organization)	Approval Date
Deliverable leader	P. Prashanth Kumar (ASM)	12/09/2025
Quality manager	P. Valle Soriano (ATOS)	15/09/2025
Project Coordinator	E. Pages (ATOS)	15/09/2025
Technical Manager	I. Seitanidis (SYN)	15/09/2025

Table of Contents

Document Information	2
Table of Contents	4
List of Tables.....	6
List of Figures	7
List of Acronyms.....	10
Executive Summary	12
1 Introduction.....	13
1.1 Purpose of the Document.....	13
1.2 Relation to other project work	13
1.3 Relation between D5.4 and D5.3	14
1.4 Structure of the Document and Reading recommendations	16
2 Smart Farming Trial	17
2.1 Aerial Precision Bio-Spraying.....	17
2.1.1 Set up and Integration.....	17
2.1.2 Test scenarios verification	28
2.1.3 Final validation results and KPIs Evaluation.....	33
2.1.4 Lessons Learned and Replication guidelines.....	34
3 Smart Energy & Smart Mobility Trial.....	35
3.1 Smart Grid - Hierarchical Grid Disturbance Mitigation.....	35
3.1.1 Set up and Integration.....	35
3.1.2 Test scenarios verification	40
3.1.3 Final validation results and KPIs Evaluation.....	44
3.1.4 Lessons Learned and Replication guidelines.....	45
3.2 Smart Mobility.....	46
3.2.1 Set up and Integration.....	46
3.2.2 Test scenarios verification	52
3.2.3 Final validation results and KPIs Evaluation.....	55
3.2.4 Lessons Learned and Replication guidelines.....	57
4 Smart Manufacturing & Industry 4.0 Trial.....	58
4.1 Set up and Integration	59
4.2 Test scenarios verification	62
4.3 Final validation results and KPIs Evaluation.....	68

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	4 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status:
			Final

4.4 Lessons Learned and Replication guidelines.....	69
5 Smart Media & XR Trial	70
5.1 Round of Athens Race	70
5.1.1 Set up and Integration.....	71
5.1.2 Test scenarios verification	79
5.1.3 Final validation results and KPIs Evaluation.....	91
5.1.4 Lessons Learned and Replication guidelines.....	98
5.2 VR Experience about ancient Workshop of sculptor Phidias.....	100
5.2.1 Set up and Integration.....	100
5.2.2 End-to-End Deployment and Migration Walkthrough	101
5.2.3 Test scenarios verification	110
5.2.4 Machine Learning Methodology for Smart XR: Stress Recognition	112
5.2.5 Final validation results and KPIs Evaluation.....	118
5.2.6 Evaluation Report for KPI_XR_01.5	122
5.2.7 Lessons Learned and Replication guidelines.....	135
5.3 Enhance AV experience in the Tholos Dome VR Theatre	136
5.3.1 Set up and Integration.....	136
5.3.2 Machine Learning Methodology for Smart XR: Gesture Recognition	138
5.3.3 Test scenarios verification	141
5.3.4 Final validation results and KPIs Evaluation.....	144
5.3.5 Lessons Learned and Replication guidelines.....	154
6 Cross-Living Labs validation & 3rd Party Support.....	155
6.1 Overview.....	155
6.2 Onboarding, Coaching and Support Sessions.....	155
7 Conclusions.....	156
8 References.....	157
Annex I. Requirements Validation.....	158
Annex2. New workload instance deployment in XR usecase.....	191

List of Tables

Table 1 List of KPIs in SE_01.	44
Table 2 Hardware Infrastructure and Functional Overview in SE_02	47
Table 3 List of KPIs in SE_02	56
Table 4 KPIs Evaluation	68
Table 5 Test scenario Smart Media	80
Table 6 KPIs Evaluation	92
Table 7 The software used for the XR_01 trial	101
Table 8 Test scenarioXR01	112
Table 9 Stress Detection Evaluation Metrics (Train vs Retrain)	116
Table 10 KPIs Evaluation	119
Table 11 The software used for the XR_02 trial	137
Table 12 Gesture Recognition Evaluation Metrics (Train vs Retrain)	140
Table 13 Test scenario XR02	143
Table 14 KPIs Evaluation	144
Table 15 Correlation of findings with KPIs	152

List of Figures

Figure 1 Historic Data Registry API endpoints	22
Figure 2 SynField endpoint sample. This is the root endpoint for SynField based devices.	23
Figure 3 Sample of the SynField data endpoint.	24
Figure 4 Sample from the drone endpoint.	24
Figure 5 Sample of the Smart Farming "/v2/entities" endpoint.	25
Figure 6 Smart Farming Login Page.	26
Figure 7 Home View, terrestrial map.	26
Figure 8 Home View, satellite map	27
Figure 9 Dashboard submenu.	27
Figure 10 Sensor submenu.	28
Figure 11 System submenu.	28
Figure 12 Preflight check at Anna Sofia Estate	29
Figure 13 Sample output from the Inference algorithm.	30
Figure 14 Sample output from the waypoint navigation algorithm.	30
Figure 15 Sample output of the center extraction algorithm.	30
Figure 16 Output of the flying controller component presenting the path followed by the UAV as well as telemetry data	31
Figure 17 UAV hovering over an olive tree performing center extraction.	31
Figure 18 Smart Farming cluster successfully registered at NEMO.	32
Figure 19 Smart Farming Inference workload registration.	32
Figure 20 Smart Farming create trigger Intent.	33
Figure 21 Workload successfully migrated to the production cluster in OneLab.	33
Figure 22 questionnaire with the farmer's feedback	34
Figure 23 Smart Energy pilot with ASM's PMU sites connected to the NEMO cluster via OTA services and PMU Analyzer	36
Figure 24 PMUs and gateways installation at ASM Terni	37
Figure 25 NEMO deployment schematic	37
Figure 26 OTA Update interface – firmware selection for Edge devices	38
Figure 27 OTA update – user interaction panel for bulk firmware actions	39
Figure 28 OTA update – live device status during firmware upgrade in SE_01	39
Figure 29 Detailed device view with real-time phasor data in SE_01	40
Figure 30 FOTA functionalities in platform	41
Figure 31 Firmware selection and device list upload interface in SE_01	41
Figure 32 SSH Connection logs and device activity timeline in SE_01	42
Figure 33 PMUs monitoring in dashboard platform	42
Figure 34 PMU analyzer log of detected issues with timestamps and data files in SE_01	43
Figure 35 PMU analyzer issue visualization showing frequency and current magnitude anomalies in SE_01	43
Figure 36 Smart Mobility pilot with EV flexibility management at COMS's NEMO cluster, linked to EMOT's blockchain.	46
Figure 37 Smart Parking Data Processing Workflow	47
Figure 38 Smart Parking and EV Charging Infrastructure at ASM Terni	48
Figure 39 Real-Time Display Interface for Parking and EV Charging Management.	48
Figure 40 Smart Parking integration to CPO platform.	49
Figure 41 Distribution of vehicle entries by hour of the day.	50
Figure 42 Distribution of parking durations across the full dataset.	50
Figure 43 Electric Mobility Platform in Eclipse Foundation GitLab	51
Figure 44 Flexibility Marketplace in Eclipse Foundation GitLab	51
Figure 45 Electric Mobility Platform YML file	52
Figure 46 Flexibility Marketplace YML file	52
Figure 47 The CMDT SuperAdmin panel showing all running containers (~300) in NEMO cluster federation.	53
Figure 48 The CMDT SuperAdmin panel shows running FOTA service revealing details such as cluster where is deployed, cluster node, container name, and tenant.	54

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	7 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

Figure 49 The CMDT SuperAdmin panel filters services of energy-marketplace, part of Smart Energy pilot.	54
Figure 50 The CMDT SuperAdmin panel filters services deployed on CMT's NEMO-Energy cluster.	55
Figure 51 scTracking architecture and deployment in the NEMO environment	60
Figure 52 scTracking login screen	61
Figure 53 scTracking landing page	62
Figure 54 Log file from Continental STAR	63
Figure 55 Tracking History window	64
Figure 56 Tracking History window	65
Figure 57 Time-Aware network for TSN time synchronization	65
Figure 58 Time Offset of TSN slave instance after TSN time synchronization.	67
Figure 59 C1: slave-device PPS output, C2: GM clock 1PPS output, and time difference between GM and Slave clock.	68
Figure 60 Stadium Location and Race Circuit	71
Figure 61 Architectural Topology Diagram of Trial	71
Figure 62 Communication Diagram of the Smart Media City Pilot	72
Figure 63 The proxmox environment	74
Figure 64 TCP Traffic Reception on Proxmox Core	75
Figure 65 TCP and UDP Network Performance Tests from TID to Proxmox	76
Figure 66 Remote production environment in TID premises	76
Figure 67 FluxCD environment for smart media pilot	77
Figure 68 RabbitMQ specific queue for use case	77
Figure 69 NEMO mobile app (1)	78
Figure 70 NEMO mobile app (2)	79
Figure 71 Architecture of Smart Media Trial	81
Figure 72 Hardware configuration of the vinctocore VM deployed on Proxmox.	82
Figure 73 Console output showing the initialization sequence of the video manager core.	82
Figure 74 CPU and RAM usage graph for the vinctocore VM.	83
Figure 75 Vinctogui Production Interface	84
Figure 76 Example of a JSON-formatted message sent by the Video Quality Probe and stored in RabbitMQ.	86
Figure 77 Predicted MOS values generated by Video Quality Probe during the analysis of a 90-second video stream.	87
Figure 78 Rtmp Server connection setup	88
Figure 79 Time-Aware network for TSN time synchronization	89
Figure 80 Time Offset of TSN slave instance after TSN time synchronization.	90
Figure 81 C1: slave-device PPS output, C2: GM clock 1PPS output, and time difference between GM and Slave clock.	91
Figure 82 Terminal screenshot of use case workflow: top-left) Pods deployed in the node; top-right) Logs for one of the Stream Transcoder, with some information from the Stream Monitor; bottom-left) Logs from video quality probe; bottom-right) Logs from AI engine	93
Figure 83 RabbitMQ screenshot with message example from AI Engine.	94
Figure 84 Screenshots from Production Control	94
Figure 85 Screenshots from Production Control	94
Figure 86 The VuzeXR 360 camera used for streaming large video sources	96
Figure 87 Table for GPS Tracking containing initial and improved coordinates	97
Figure 88 Egaleo Municipal Stadium, with the positions of the users	98
Figure 89 Architectural Component diagram of the XR_01 trial	100
Figure 90 VR HMD participants wearing the smart watch during the trial	101
Figure 91 Create workload for Smart XR use case	102
Figure 92 Upload workload Helm charts (1)	103
Figure 93 Upload workload Helm charts (2)	103
Figure 94 Workload onboarding state	104
Figure 95 Workload accepted state	104
Figure 96 Create workload instance in cluster without GPU	105
Figure 97 Workload instance deployment details	105
Figure 98 Workload namespace	105
Figure 99 Workload namespace details	106
Figure 100 Rising CPU usage	106

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	8 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status: Final

Figure 101 Retrain pod logs	106
Figure 102 Upgrade workload instance with GPU	106
Figure 103 Deployed workload with GPU	107
Figure 104 GPU is unavailable for “dev-onelab” cluster (1)	107
Figure 105 GPU is unavailable for “dev-onelab” cluster (2)	107
Figure 106 Create MachineLearning Intent	108
Figure 107 MachineLearning Intent fulfilled	108
Figure 108 Migration of workload from “dev-onelab” to “prod-onelab”	109
Figure 109 Workload instance timeline	109
Figure 110 Delete workload (1)	110
Figure 111 Delete workload (2)	110
Figure 112 VR user with smart watch IoT device and its emotional status indication show on a PC receiving messages form Nemo OS event server	112
Figure 113 Confusion Matrix for Initial Training	115
Figure 114 Confusion Matrix for Retraining	115
Figure 115 Accuracy Curve for Initial Training	116
Figure 116 Accuracy Curves for Retraining	117
Figure 117 Loss Curves for Initial Training	117
Figure 118 Loss Curves for Retraining	118
Figure 119 A museum educator of the Hellenic Cosmos monitors an experience looking at the PC that receives event from Nemo OS and intervenes when stress is high.	120
Figure 120 Cutout of the event logs of the Event Server and registered IoT device	121
Figure 121 Cutout of the Log entry of the receiving IoT PC and ML node.	122
Figure 122 Architectural Component diagram of the XR_02 trial	136
Figure 123 Home navigation using gestures powered by the NemoMeta OS	137
Figure 124 Confusion Matrix for Initial Training	140
Figure 125 Confusion Matrix for Retrained Model	141
Figure 126 A Museum Educator is issuing an OK gesture to signal show end. The arrow shows IoT based camera and processing device. An IoT device (PC) subscribed to the Nemo OS event server is used outside at the reception to receive the message.	144
Figure 127 Cutout of the Log entry of the receiving IoT PC and ML node.	146

List of Acronyms

Abbreviation Acronym	Description
APN	Access Point Name
HTTP	Hypertext Transfer Protocol
HW	Hardware
IaaS	Infrastructure-as-a-Service
IDM	Identity Management
IoT	Internet of Things
FluxCD	Flux Continuous Delivery
FOTA	Firmware Update over Air
K8s	Kubernetes
KPI	Key Performance Indicator
LAN	Local Area Network
LCM	Life-Cycle Manager
LSP	Large Scale Pilot
MAC	Mandatory Access Control
MEC	Multi-access Edge Computing
Meta-OS	Meta-Operating System
MOS	Mean Opinion Score
ML	Machine Learning
mNCC	Meta Network Cluster Controller
MOCA	Monetization and Consensus-based Accountability
MQTT	Message Queuing Telemetry Transport
mRA	meta-Reference Architecture
MVS	Multi-View Stereo
NEF	Network Exposure Function
NF	Network Function
NFV	Network Function Virtualization
NIC	Network Interface Card
NRF	Network Repository Function
NVR	Network Video Recorder
NWDAF	Network Data Analytics Function
OAM	Operations, Administration, and Maintenance
OBD	On-Board Diagnostic
ODM	OpenDroneMap
ORB	Oriented FAST and Rotated BRIEF
OS	Operating System
PaaS	Platform-as-a-Service
PDU	Protocol Data Unit
PLC	Product Life Cycle

Abbreviation Acronym	Description
PMU	Phasor Measurement Unit
PPEF	PRESS & Policy Enforcement Framework
PWA	Power Quality Analyzer
PRESS	Privacy, data pRotection, Ethics, Security & Societal
QoE	Quality of Experience
RAM	Random Access Memory
RAN	Radio Access Network
RBNR	Racing Bib Number Recognition
RL	Reinforcement Learning
ROS	Robot Operating System
RTT	Round Trip Time
SAM	Segment Anything Model
SaaS	Software-as-a-Service
SC	Smart Media/City
SDK	Software Development Kit
SDN	Software Defined Networking
SEE	Secure Execution Environment
SFM	Structure-from-Motion
SLA	Service Level Agreement
SLAM	Simultaneous Localization and Mapping
SLO	Service Level Objective
SMX	Smart Meter eXtension
SSD	Solid-State Drive
TEMP	Skin temperature
TSN	Time Sensitive Networks
UAV	Unmanned Aerial Vehicle
UC	Use Case
UE	User Equipment
VM	Virtual Machine
VNF	Virtual Network Function
VPN	Virtual Private Network
VR	Virtual Reality
WP	Work Package
YAML	Yet Another Markup Language
YOLO	You Only Look Once (Multiple versions v2-8)
XR	Extended Reality

Executive Summary

The rapid evolution of computing technologies—especially in the domains of Internet of Things (IoT), Edge computing, and Cloud computing—has created new opportunities but also new challenges for managing distributed applications. Traditional operating systems and orchestration tools often fall short in handling the complexity and heterogeneity of such environments.

The NEMO project addresses this gap by introducing a meta-Operating System (meta-OS) designed to enable seamless deployment, orchestration and lifecycle management of applications and workloads across the IoT–Edge–Cloud continuum. NEMO abstracts the underlying infrastructure and provides developers, service providers, and infrastructure owners with a unified set of tools to simplify operations, improve performance, and increase portability of modern applications.

Within the project, Work Package 5 (WP5) focuses on validating the NEMO MetaOS in real-world conditions through Living Labs. These trials demonstrate the applicability of the platform across diverse domains and provide insights on performance, integration, and usability.

Deliverable D5.4 presents the final evaluation results from these validation activities, building on the intermediate findings of D5.3[4]. Deliverable D5.4 – NEMO Living Labs Use Cases Evaluation Results – Final Version – presents the final results from these validation activities. This includes:

- End-to-end integration and execution of use case-specific services using NEMO.
- Validation of workload orchestration, intent-based scheduling, and workload migration capabilities in diverse real-life scenarios.
- Feedback from third-party developers, infrastructure providers, and application owners on the usability and added value of the platform.
- Reflections on platform maturity, performance, and lessons learned during final piloting stages.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	12 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

1 Introduction

The NEMO project introduces a novel meta-Operating System (meta-OS) designed to enable seamless orchestration and lifecycle management of application and workloads across the IoT–Edge–Cloud continuum. By abstracting the complexities of heterogeneous infrastructures, NEMO offers a unified platform that supports deployment, migration, and optimization of services in highly dynamic environments.

To validate the capabilities of the NEMO meta-OS, the platform was deployed and tested across five verticals, each focusing on a distinct vertical and set of use cases. The Smart Farming Living Lab tested precision agriculture solutions such as aerial bio-spraying and terrestrial weed management. The Smart Energy & Mobility Living Lab addressed grid disturbance mitigation and coordinated e-mobility services. The Smart Industry Living Lab piloted automated logistics and human-centred safety solutions. Finally, the Smart Media & XR Living Lab explored innovative media applications, including support for live events and immersive VR experiences. Together, these trials provided a diverse set of environments to assess the flexibility and robustness of the NEMO MetaOS.

1.1 Purpose of the Document

The purpose of this document is to present the final evaluation results of the NEMO Living Labs use cases across all participating verticals, namely Smart Farming, Smart Energy & Mobility, Smart Industry, and Smart Media.

This deliverable builds upon the initial results documented in D5.3[4], incorporating the full-scale integration and validation of NEMO Meta-OS components within real-world pilot environments. It captures the performance, usability, and impact of the deployed solutions, highlighting both technical outcomes and lessons learned.

The evaluation focuses on:

- The deployment and operation of applications and services using the NEMO Meta-OS framework.
- The ability of NEMO to orchestrate workloads and manage resources efficiently across the Cloud–Edge–IoT continuum.
- The added value and benefits of NEMO for application developers, infrastructure operators, and end-users.
- Key performance indicators validation and qualitative feedback from the Living Labs.

The document also serves as a reference for stakeholders seeking to understand the practical application of the NEMO framework and its potential to support cross-domain, scalable, and interoperable solutions.

1.2 Relation to other project work

Deliverable D5.4 is the second and final version of NEMO Living Labs Use Cases Evaluation Results, following D5.3[4], which provided the initial evaluation of pilot activities and early validation results. While D5.3[4] documented preparatory steps, initial deployments, and intermediate findings, D5.4 reports the outcomes from the fully integrated NEMO meta-OS platform and applications in the Living Labs.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	13 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

D5.4 builds on the use case definitions, requirements, and KPIs established in WP1 deliverables D1.1[1] and D1.3[2], and validates them through real-world trials across all verticals. It also incorporates technical foundations from WP4 where the NEMO components were finalized and proven ready for deployment.

1.3 Relation between D5.4 and D5.3

The structure of D5.4 has evolved compared to D5.3[4] in order to better highlight the progress made in the Living Labs and the lessons learned from the final evaluation phase. While D5.3[4] provided an intermediate snapshot of pilot activities, D5.4 builds on that work and offers a more complete picture of how the trials validated the NEMO Meta-OS in different domains.

To achieve this, the document introduces new subsections dedicated to the final contributions of the pilots, the Open Call integrations, and cross-domain observations. Some parts that were previously grouped together in D5.3[4] have been expanded in D5.4 into standalone sections, allowing the results and impacts of each pilot to be described with more clarity.

In this way, D5.4 not only continues the reporting from earlier deliverables but also complements the technical outcomes of WP4 by showing how the integrated tools have been tested and validated in real environments.

Section in D5.4	Section in D5.3	Differences
1. Introduction	1 Introduction	The section has been updated
1.1 Purpose of the document	1.1 Objectives of the document	Minor update
1.2 Relation to other project work	1.2 Connection to Other Work Packages and Deliverables	The section has been updated
1.3 Relation between D5.4 and D5.3		New Section in D5.4
1.4 Structure of the Document and Reading recommendations	1.3 Document Structure and Overview	The section has been updated
2 Smart farming Trial	2 Smart farming Trial	Several updates in this section
2.1.1 Set up and Integration	2.2.1 Trial Site Description Updates	Major Updates
2.1.2 Test scenarios verification	2.2.2 Technical validation	Major Updates
2.1.3 Final validation results and KPIs Evaluation	2.2.5 Intermediate Results	Major Updates
2.1.4 Lessons Learned and Replication guidelines		New Section in D5.4
3 Smart Energy & Smart Mobility Trial	3 Smart Energy & Smart Mobility Trial	The section has been updated
3.1 Smart Grid - Hierarchical Grid Disturbance Mitigation	3.2 SE_01 Smart Grid	The section has been updated
3.1.1 Set up and Integration	3.2.1 Trial Site Description Updates	Major Updates
3.1.2 Test scenarios verification	3.2.2 Technical validation	Major Updates
3.1.3 Final validation results and KPIs Evaluation	3.2.5 Intermediate Results	Major Updates
3.1.4 Lessons Learned and Replication guidelines		New Section in D5.4
3.2 Smart Mobility	3.3 SE_02 Smart Mobility/City	The section has been updated

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	14 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Section in D5.4	Section in D5.3	Differences
3.2.1 Set up and Integration	3.3.1 Trial Site Description Updates	Major Updates
3.2.2 Test scenarios verification	3.3.2 Technical validation	Major Updates
3.2.3 Final validation results and KPIs Evaluation	3.3.5 Intermediate Results	Major Updates
3.2.4 Lessons Learned and Replication guidelines		New Section in D5.4
4 Smart Manufacturing & Industry 4.0 Trial	4 Smart Manufacturing & Industry 4.0 Trial	The section has been updated
4.1 Set up and Integration	4.2.1 Trial Site Description Updates	Major Updates
4.2 Test scenarios verification	4.2.2 Technical validation	Major Updates
4.3 Final validation results and KPIs Evaluation	4.2.5 Intermediate Results	Major Updates
4.4 Lessons Learned and Replication guidelines		New Section in D5.4
5 Smart Media & XR Trial	5 Smart Media & XR Trial	The section has been updated
5.1 Round of Athens Race	5.2 SC_01 Round of Athens Race	The section has been updated
5.1.1 Set up and Integration	5.2.1 Trial Site Description and Updates	Major Updates
5.1.2 Test scenarios verification	5.2.2 Technical validation	Major Updates
5.1.3 Final validation results and KPIs Evaluation	5.2.5 Intermediate Results	Major Updates
5.1.4 Lessons Learned and Replication guidelines		New Section in D5.4
5.2 VR Experience about ancient Workshop of sculptor Phidias	5.3 XR_01 VR Experience about ancient Workshop of sculptor Phidias	The section has been updated
5.2.1 Set up and Integration	5.3.1 Trial Site Description and Updates	Major Updates
5.2.2 Test scenarios verification	5.3.2 Technical validation	Major Updates
5.2.3 Final validation results and KPIs Evaluation	5.3.5 Intermediate Results	Major Updates
5.2.4 Lessons Learned and Replication guidelines		New Section in D5.4
5.3 Enhance AV experience in the Tholos Dome VR Theatre	5.4 XR_02 Enhance AV experience in the Tholos Dome VR Theatre	The section has been updated
5.3.1 Set up and Integration	5.4.1 Trial Site Description and Updates	Major Updates
5.3.2 Test scenarios verification	5.4.2 Technical validation	Major Updates
5.3.3 Final validation results and KPIs Evaluation	5.4.5 Intermediate Results	Major Updates
5.3.4 Lessons Learned and Replication guidelines		New Section in D5.4
6 Cross-Living Labs validation & 3rd Party Support		New Section in D5.4
6.1 Overview		New Section in D5.4

Section in D5.4	Section in D5.3	Differences
6.2 Onboarding, Coaching and Support Sessions		New Section in D5.4
6.3 Feedback from Open Call Projects		New Section in D5.4
7 Conclusions	7 Conclusions	The section has been updated

1.4 Structure of the Document and Reading recommendations

This deliverable is structured to guide readers progressively from a general understanding of the objectives to the detailed validation results of each Living Lab. The organization reflects both the logical flow of the validation activities and the project's multi-domain nature.

Section 1 (Introduction) provides the context, objectives, and positioning of this deliverable within the project, including its relation to earlier work (notably Deliverable D5.3[4]).

Sections 2–5 (Living Lab Trials) detail the validation activities for each domain:

- Smart Farming
- Smart Energy & Smart Mobility
- Smart Manufacturing & Industry 4.0
- Smart Media & XR

Each trial section follows a common structure to ensure comparability, covering (i) set-up and integration, (ii) test scenario execution, (iii) validation results and KPI evaluation, and (iv) lessons learned and replication guidelines.

Section 6 (Cross-Living Labs Validation & 3rd Party Support) summarizes activities that cut across domains, including the onboarding and support of third-party applications via open calls, and feedback collected from these engagements.

Section 7 (Conclusions) synthesizes the key findings from the validation phase and highlights the main contributions to the project objectives.

Annexes provide supplementary material that supports the main text but would otherwise interrupt its readability.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	16 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

2 Smart Farming Trial

2.1 Aerial Precision Bio-Spraying

2.1.1 Set up and Integration

The Smart Farming use case is a complex system that consists of several subcomponents, each of them significantly contributing to the realization of the described system. In the following subsections these subcomponents will be presented organized based on their functionality. The Smart Farming Drone setup describes the technical details that lead to its assembly and functionality. The Smart Farming API describes several endpoints used to decompose the data layer from the Front-End and act as a centralized point for storage. The Smart Farming UI provides insights of the user interface where the information gathered is displayed. Finally, in the Smart Farming cluster, a description of how the microservices of the use case are orchestrated along with information about the cluster interconnection with the NEMO platform.

2.1.1.1 Smart Farming Drone set up

2.1.1.1.1 Hardware Components set up

The hardware configuration of the Unmanned Aerial Vehicle (UAV) platform is based on the Holybro X650 Development Kit¹, a robust quadcopter airframe tailored for research and development applications. The frame consists of carbon fiber arms and a reinforced plastic central body, providing both structural rigidity and lightweight performance. With a diagonal wheelbase of 650 mm and an X-layout, the airframe is optimized for stability during flight and is compatible with a wide range of onboard components and payloads. This configuration is particularly suited to the requirements of a precision bio-spraying UAV, allowing for modular expansion and secure payload mounting.

At the core of the UAV's avionics is the Pixhawk 6X² flight controller, an open-source, high-performance autopilot platform widely adopted in the research community. It features redundant Inertial Measurement Units (IMUs) and barometers, an integrated vibration-isolated IMU, and a powerful STM32H743³ processor. The Pixhawk 6X supports both PX4⁴ and ArduPilot⁵ firmware, providing flexibility in terms of flight stack and mission profile. This flight controller was selected not only for its technical capabilities but also because of its open-source nature, extensive documentation, and vibrant development community—all of which are critical for iterative prototyping, debugging, and integration with autonomous navigation algorithms and cloud-native telemetry pipelines.

Power is supplied through a 6S Lithium Polymer (LiPo) battery pack, rated at 12000 Mah. This configuration offers a practical balance between energy density, discharge rate, and total weight, making it well-suited for mid-range flight durations with moderate payloads. The battery interfaces with a power management module such as the PM06, which ensures safe voltage regulation, provides telemetry feedback (current, voltage, power consumption), and delivers stable power to both the flight controller and the payload systems.

¹ <https://holybro.com/products/x650-development-kit>

² <https://holybro.com/products/pixhawk-6x>

³ <https://www.st.com/en/microcontrollers-microprocessors/stm32h743-753.html>

⁴ <https://px4.io/>

⁵ <https://ardupilot.org/>

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	17 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

The propulsion system includes brushless Direct Current (DC) motors and Electronic Speed Controllers (ESCs) provided by the Holybro kit, pre-matched for optimal thrust-to-weight ratio and efficiency. The UAV is equipped with 12-to-13-inch carbon composite propellers, which offer high lift and durability. This configuration ensures stable and responsive flight characteristics, which are essential during precision spraying tasks or autonomous missions in field conditions.

For manual control and configuration, the system includes a Radiomaster TX16S remote controller⁶, running OpenTX⁷ or EdgeTX⁸ firmware. Paired with a long-range, low-latency receiver such as a Crossfire Nano⁹ or ExpressLRS¹⁰ module, the remote-control system provides flexible, reliable communication for manual overrides and testing. The open-source firmware allows for a high-level tailoring of flight modes, telemetry feedback on screen, and support for complex mission profiles.

In terms of autonomous navigation and localization, the UAV integrates a u-blox M8N GNSS module¹¹ with an onboard magnetometer. This module delivers reliable positional accuracy and heading information, which is critical for executing waypoints, return-to-home procedures, and maintaining course stability under varying magnetic and environmental conditions. The module is natively supported in the Pixhawk ecosystem and ensures tight synchronization with the navigation stack.

The communication between the UAV and the Ground Control Station (GCS) is facilitated through a 915 MHz SiTelemetry radio¹², providing real-time bidirectional communication using the MAVLink¹³ protocol. This allows for live monitoring of UAV status, in-field mission reconfiguration, and seamless data handoff to the cloud-based telemetry system. The choice of this module is driven by its compatibility with open-source flight stacks and proven reliability in outdoor deployment scenarios.

Finally, the UAV features a custom payload mounting system designed to accommodate an onboard camera and an NVIDIA Jetson Orin Nano¹⁴ module. The mounting solution is engineered to minimize vibration and maintain aerodynamic balance, ensuring that the visual sensing system and edge AI processing hardware can operate effectively without affecting flight stability. This modular design aligns with the overall architecture of the platform, supporting real-time image capture, onboard inference tasks, and future expansion for autonomous perception and navigation applications in smart farming environments.

This entire setup is driven by core design principles prioritizing modularity, open standards, and data integration. Each hardware component was selected to support seamless migration between onboard and cloud-level autonomy, which is central to the vision of a cloud-native smart farming application. The combination of flexible open-source avionics, telemetry-enabled components, and field-ready mechanical structure provides a solid foundation for further development in autonomous agricultural operations.

⁶ <https://radiomasterrc.com/products/tx16s-mark-ii-max-radio-controller>

⁷ <https://www.open-tx.org/>

⁸ <https://edgetx.org/>

⁹ https://www.team-blacksheep.com/products/prod:crossfire_nano_rx

¹⁰ <https://www.expresslrs.org/>

¹¹ <https://www.u-blox.com/en/product/neo-m8-series>

¹² <https://holybro.com/products/sik-telemetry-radio-v3?variant=41562952302781>

¹³ <https://mavlink.io/en/>

¹⁴ <https://www.nvidia.com/en-eu/autonomous-machines/embedded-systems/jetson-orin/nano-super-developer-kit/>

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	18 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

2.1.1.1.2 Software Components set up

Five core software modules were developed for the autonomous UAV system designed for precision agriculture applications, waypoint navigation, image capture, reference, center extraction and center navigation. These modules collectively form an event-driven microservices architecture that facilitates autonomous drone operations for the use-case of precision spraying. The system integrates advanced technologies such as computer vision, drone control, and distributed computing to deliver a robust solution for agricultural automation.

The autonomous drone system employs a distributed, event-driven architecture characterized by a microservices pattern, where each module operates as an independent service with distinct responsibilities. Communication between services is facilitated through RabbitMQ¹⁵, a message broker that enables asynchronous, event-driven interactions. MinIO¹⁶ serves as the centralized object storage for images, models, and metadata, ensuring efficient data management. Kubernetes orchestrates container deployment, scaling, and service discovery, providing a scalable and resilient infrastructure. Integration with MAVSDK enables precise drone control and autonomous flight operations, forming the backbone of the system's operational capabilities.

The waypoint navigation module executes pre-defined flight missions by navigating through a series of waypoints, coordinating the mission flow and triggering image capture at each waypoint. Its components include a mission manager for orchestrating execution, a MAVSDK helper for drone connection and mission upload, an event listener for center reached confirmations, an event publisher for waypoint reached events, and a waypoint configuration component for managing JSON-based waypoint definitions. The module uses MAVSDK MissionItem¹⁷ for standardized waypoint definition, supports configurable mission speeds, and integrates with image capture through waypoint events. Waypoints are defined in JSON configuration files, and mission validation ensures correctness before execution. The module listens to the center_reached queue for mission coordination, publishes waypoint events to trigger image capture, and reads waypoints from JSON files. The mission execution process connects to the drone, sets mission speed, creates and uploads a mission plan, starts the mission, and publishes waypoint reached events.

The image_capture module manages drone camera operations, capturing high-quality images at waypoints and storing them with metadata in MinIO. It integrates with Sequoia¹⁸ cameras for multispectral imaging and includes components such as a camera manager for orchestrating operations, a Sequoia capture component for camera-specific tasks, an event listener for waypoint reached events, a MinIO helper for image storage and metadata management, and a utility for file operations. The module uses the gphoto2¹⁹ library for Sequoia camera control, preserves Exchangeable Image File Format (EXIF) metadata including Global Positioning System (GPS) coordinates, and employs timestamp-based file naming for uniqueness. Temporary local storage is used before uploading to MinIO, and comprehensive error handling ensures robust camera operations. The module listens to the waypoint_reached queue for capturing triggers and stores images in MinIO with a configurable prefix structure. The capture process involves capturing an image, saving it temporarily, extracting EXIF metadata, and uploading it to MinIO with associated metadata.

¹⁵ <https://www.rabbitmq.com/>

¹⁶ <https://www.min.io/>

¹⁷ https://mavsdk.mavlink.io/main/en/cpp/api_reference/structmavsdk_1_1_mission_1_1_mission_item.html

¹⁸ <https://www.parrot.com/en/support/documentation/sequoia>

¹⁹ <https://github.com/gphoto/libgphoto2>

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	19 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

The inference module performs object segmentation using YOLOv11 models, processing captured images to identify olive trees and generate segmentation masks. Its components include an inference processor for YOLO model operations, an event listener for image capture events, an event publisher for inference results, a MinIO helper for model and image file operations, and a model management component for loading and caching YOLO models. The module integrates Ultralytics²⁰ YOLOv11 for state-of-the-art segmentation, caches models locally for performance, stores inference results with unique UUIDs, and implements proper cleanup of temporary files. It receives image URLs from the image_captured queue, publishes result URLs to the output_queue, downloads images from MinIO, and stores results in the inference bucket. The inference process downloads an image, processes it with the YOLO model, saves the result, uploads it to MinIO, and publishes the result URL.

The center_extraction module processes segmentation masks to extract precise GPS coordinates of detected targets, leveraging computer vision techniques to calculate the center point of segmented objects and convert pixel coordinates to GPS coordinates. Its core components include a center extraction processor for mask analysis, a scaling factor extraction component for coordinate calculations, an event listener for segmentation mask events, an event publisher for extracted coordinates, and a MinIO helper for mask file operations. The module employs OpenCV for robust mask analysis, using contour detection to identify the largest contour and image moments for precise center calculation. GPS conversion incorporates scaling factors, and metadata integration extracts GPS and orientation data from image EXIF. The module receives segmentation mask events from the result_captured queue, publishes extracted coordinates to the center_coords_extracted queue, and processes masks from the MinIO inference bucket. The center extraction algorithm identifies the largest contour in a binary mask, calculates its moments, and determines the center coordinates in pixel space.

The center_navigation module is tasked with autonomously navigating the drone to specific GPS coordinates, serving as the final component in the target detection and navigation pipeline. It comprises several core components, including a MAVSDK helper for managing drone connections and flight control, an event listener for processing coordinate events from RabbitMQ, a center navigation component for orchestrating the navigation process, a MinIO helper for retrieving coordinate data, and an event publisher for signaling completion events. The module adopts an event-driven architecture using RabbitMQ for receiving coordinates, ensuring loose coupling with other services. Health monitoring is implemented to verify drone status before navigation, and position verification with timeout handling ensures reliable operation. The drone hovers for five seconds at the target position to ensure stability, and comprehensive error handling with logging enhances system reliability. The module receives coordinates via the center_coordinates queue, publishes completion events to the center_reached queue, and retrieves coordinate files from the MinIO center-coordinates bucket. The navigation logic connects to the drone, performs health checks, navigates to the specified coordinates, verifies position, and hovers before publishing completion events.

The modules communicate through a structured event flow. The waypoint_navigation module triggers image capture by the image_capture module at waypoints. The image_capture module sends captured images to the inference module for processing. The inference module generates segmentation masks, which are processed by the center_extraction module to extract coordinates. The center_extraction module provides these coordinates to the center_navigation module, which navigates to the target and confirms completion to the waypoint_navigation module. The message queue structure includes waypoint_reached for triggering image capture, image_captured for initiating inference, result_captured for triggering center extraction, center_coords_extracted for providing navigation coordinates, and center_reached for confirming navigation completion. Data flow involves JSON waypoint

²⁰ <https://docs.ultralytics.com/models/yolo11/>

configurations, MinIO storage for images with metadata, YOLO model inference results, and extracted coordinates.

The system is deployed using Kubernetes, with each module having comprehensive deployment configurations, including container orchestration with resource limits, ConfigMaps for non-sensitive configurations, and Secrets for secure credential management. Resource management specifies memory limits of 256Mi to 512Mi and CPU limits of 250m to 500m per container, with privileged access for serial port communication and volume mounts for MAVSDK. Security considerations include storing credentials in Kubernetes Secrets, isolating communication through RabbitMQ network policies and securing Docker Hub credentials.

The system adheres to event-driven architecture principles, ensuring loose coupling through event-based communication, scalability through independent service scaling, fault tolerance to prevent cascading failures, and asynchronous processing for non-blocking operations. Microservices principles are followed, with each module having a single responsibility, independent deployment, technology diversity, and data isolation. Error handling and resilience are achieved through graceful degradation, automatic retries, circuit breakers, and comprehensive logging.

Performance is optimized through model caching for faster YOLO inference, image compression for efficient storage and transmission, asynchronous processing for non-blocking operations, and Kubernetes resource constraints to prevent exhaustion. Scalability is enabled through horizontal scaling with Kubernetes, load distribution via RabbitMQ, stateless design for service restarts, and environment-based configuration management.

The system employs structured JSON-formatted logging with configurable levels, centralized Kubernetes pod logs, and comprehensive error tracking. Health monitoring includes Kubernetes liveness and readiness probes, MAVSDK drone health checks, RabbitMQ queue monitoring, and resource usage tracking for CPU, memory, and network.

The autonomous drone system's software modules form a robust, event-driven microservices architecture that integrates computer vision, drone control, and distributed computing for precision agriculture. The modular design ensures clear separation of concerns, while event-driven communication via RabbitMQ provides loose coupling. Kubernetes-based deployment enables scalability, and comprehensive error handling ensures reliability. The system is well-suited for complex drone operations, offering flexibility and maintainability.

2.1.1.2 Smart Farming API

The Back-end system of the Smart Farming use case is a complex system that guarantees the seamless and uneventfully the continues data collection from the various devices used, e.g., SynField, drone, agrirobot. In this context, several state-of-the-art technologies were used for the realization of this system.

- FIWARE Orion to receive the device measurements sent by the various IoT devices
- FIWARE IoT Agent, which works as a middleware to allow devices to communicate with Orion without adapting the NGSI protocol
- The Historic Data Registry (HDR) maintains records of all the data transmitted by the IoT devices. The HDR is based on well-established technologies, such as the Python programming language along with the Flask framework which provides low footprint API services. The various datatypes supported by the HDR along with the data's nature required the use of highly scalable and highly available data storage that meets the needs of a modern IoT system. The

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	21 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

MongoDB²¹ datastore was selected as a modern highly scalable and versatile technology that meets the needs of the Smart Farming use case.

The Smart Farming back-end system leverages from the technology introduced by the FIWARE Foundation²² and particularly from the FIWARE ORION²³. FIWARE ORION is a context broker that provides an automated environment for managing the whole lifecycle of context information, such as updates, queries, registration, subscription. One of the benefits of ORION that the SF UC uses is the automated notification upon an event occurs, in this way when a new device is registered the data transmitted are captured and forwarded to the SF system. As ORION does not provide a storage mechanism by design, the creation of a middleware application that would provide storage was necessary. In this context, a microservice was created responsible for storing in a database system the data collected by ORION as well as a REST endpoint that will filter the received values. The Smart Farming (F2F) Historic Data Registry API provides a variety of endpoints each with a specific functionality, Figure 1. It allows the retrieval of the device data. Figure 1 demonstrates the set of REST endpoints created for the data exchange between the back-end system and the front-end or any 3rd party application. In this context, a SWAGGER²⁴ page was created to provide both well-structured and self-explanatory documentation for the integrating parties and a sandbox for trying the endpoints produced.



Figure 1 Historic Data Registry API endpoints

²¹<https://www.mongodb.com/>

²² <https://www.fiware.org/>

²³<https://fiware-orion.readthedocs.io/en/master/>

²⁴ [API Documentation & Design Tools for Teams | Swagger](#)

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	22 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status:
			Final

SynField is a modular device that can host a vast variety of sensors, allowing each SynField device deployed to be equipped with different sensing capabilities. With the “/synfield” endpoint Figure 2, we can retrieve the available SynField devices. The devices operate at different locations and collect the metrics that are supported by the installed sensors. To this end, the REST endpoints follow a hierarchical structure with a dedicated endpoint providing the available node IDs while another one provides the sensors available at a particular device. For example, the “South Node” SynField node collects measurements for the “Solar validation level” and “Barometric Pressure” metrics among others. The aforementioned functionality automates the discovery process and provides a widely reconfigurable schema for accessing and parsing devices that differentiate in terms of capabilities. This abstraction enables the faster and easier deployment and usage of devices as well as minimizing the time required for the complete system to be up and running in large scale deployments with various sensor nodes.

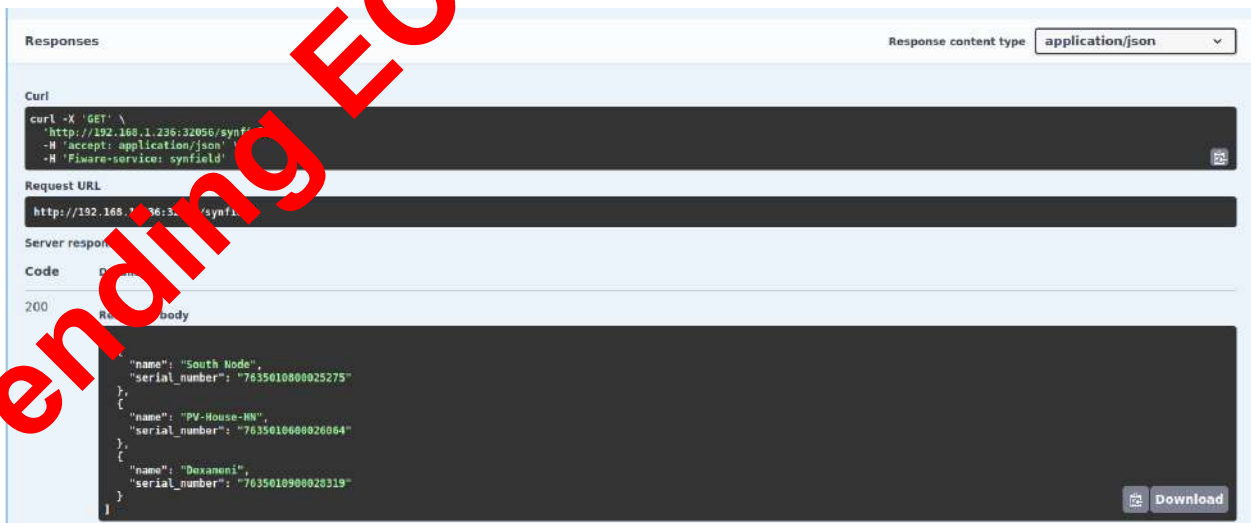


Figure 2 SynField endpoint sample. This is the root endpoint for SynField based devices.

In addition to the control endpoints, a generic endpoint was created in order to provide in a well-structured way the data collected from the deployed devices. This endpoint receives as an input the deviceID and the period of the measurement as a start and end time “/get_synfield_measurements/<deviceID>/startTime/endTime”, Figure 3. The data provided by the endpoint can be utilized to feed a timeseries plot in the front-end or as an input to a training model. In the response body as depicted in Figure 3 each record contains the DeviceID, the sensing service ID which refers to the nature of the sensing device, e.g., solar radiation, temperature, etc. as well as the value recorded in the SI and the timestamp that the data were captured. In a similar manner, an endpoint for collecting the data from the drone was created, Figure 4. The “/get_drone_start_end/startTime/endTime” endpoint provides all the required information about the drone status. This information is about the telemetry data of the drone, such as velocity, altitude, attitude, geolocation data, etc. To enhance the telemetry information transmitted by the drone, operational data like battery’s state of charge, number of satellites locked on the device, if return to home functionality is set and the inertial measurement unit (IMU). These parameters are crucial to determine the flying status of the drone and its efficiency both from a technical and a user perspective.

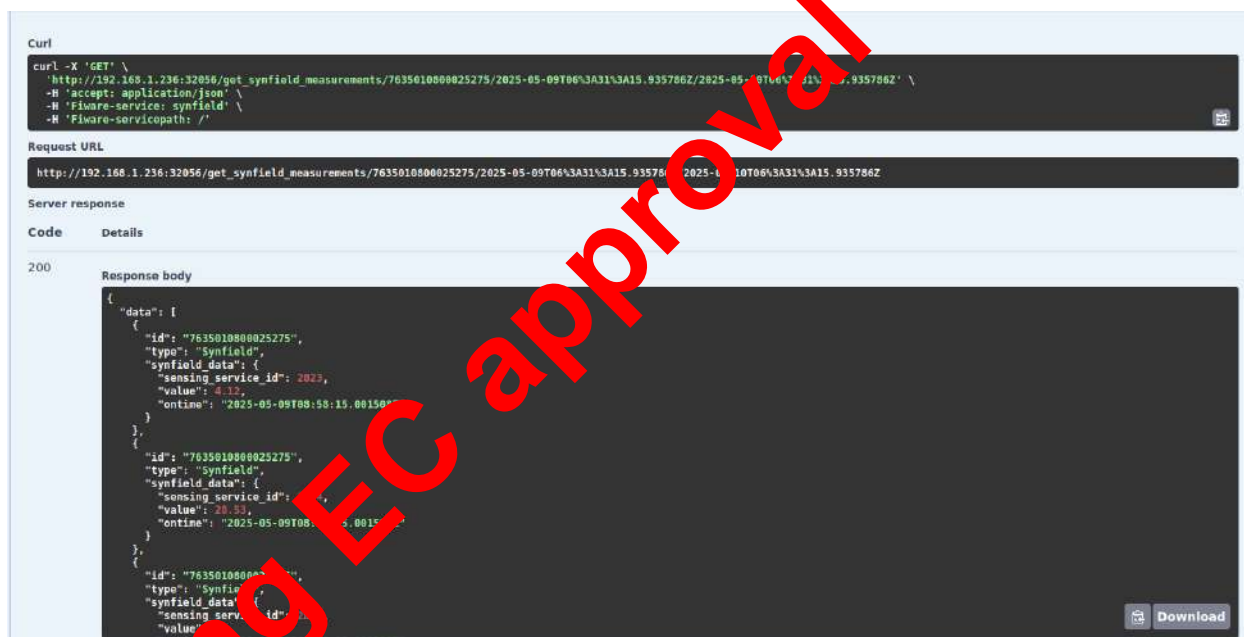


Figure 3 Sample of the SynField data endpoint.

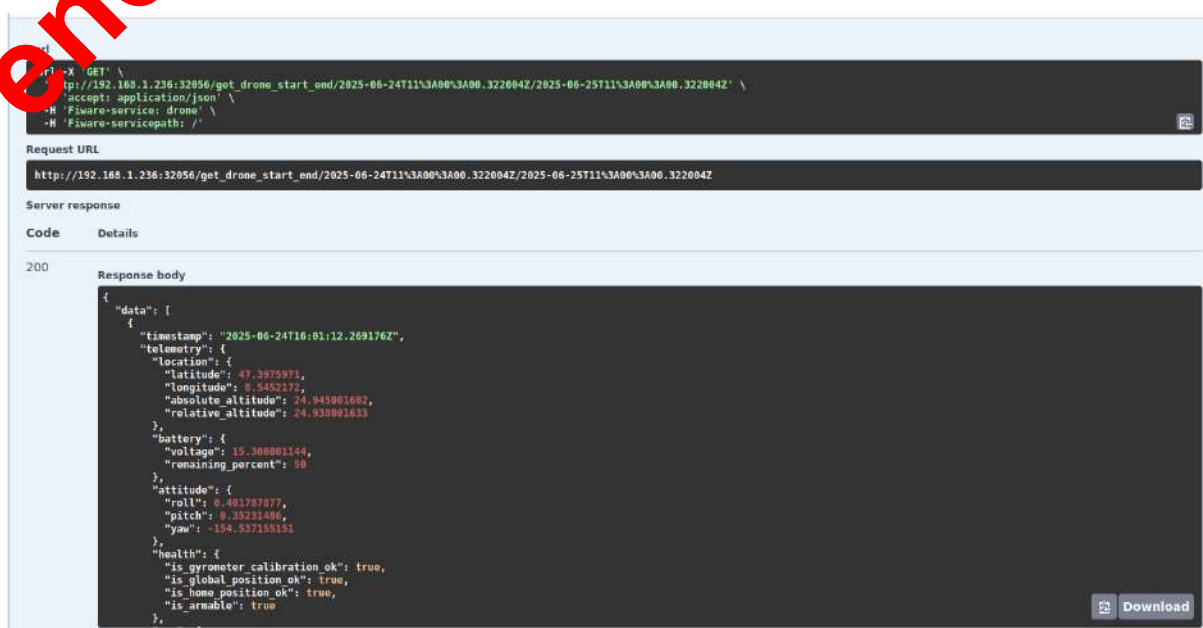
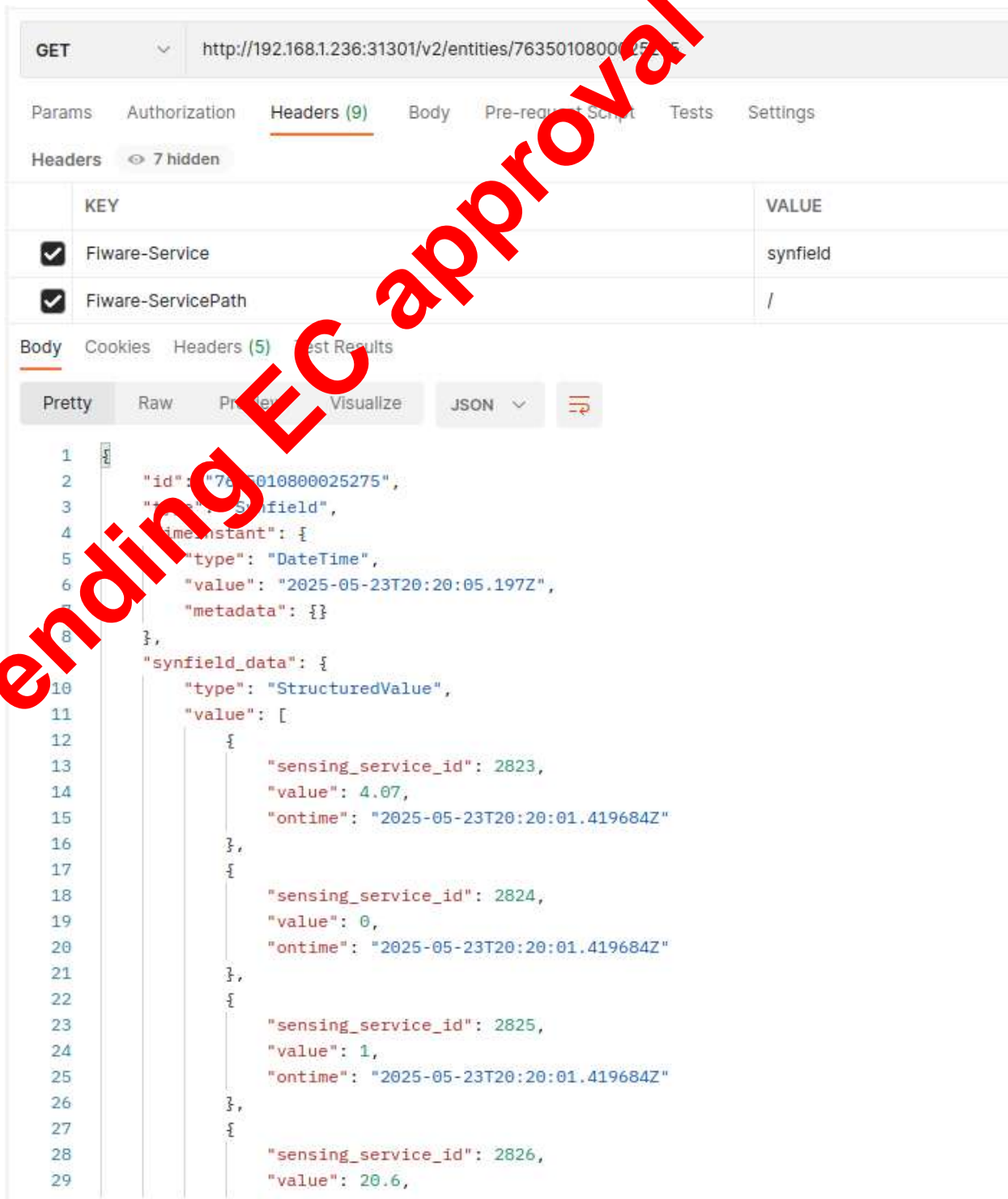


Figure 4 Sample from the drone endpoint.

Besides to the endpoints presented before, there is another endpoint used for presenting the real-time metrics with minimum delay. This endpoint is automatically generated by the ORION and provides the latest raw data received from the device, Figure 5 This endpoint lacks the functionality of historicity and it is meant to be used only for showcasing the latest data collected in a dashboard or in general a real-time interface where instantaneous information needs to be displayed. Here, we can get in “/v2/entities” the latest measurement from FIWARE Orion.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	24 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



GET http://192.168.1.236:31301/v2/entities/7635010800025275

Params Authorization Headers (9) Body Pre-request Script Tests Settings

Headers 7 hidden

KEY	VALUE
☒ Fiware-Service	synfield
☒ Fiware-ServicePath	/

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```

1  {
2    "id": "7635010800025275",
3    "type": "Synfield",
4    "timeInstant": {
5      "type": "DateTime",
6      "value": "2025-05-23T20:20:05.197Z",
7      "metadata": {}
8    },
9    "synfield_data": {
10     "type": "StructuredValue",
11     "value": [
12       {
13         "sensing_service_id": 2823,
14         "value": 4.07,
15         "ontime": "2025-05-23T20:20:01.419684Z"
16       },
17       {
18         "sensing_service_id": 2824,
19         "value": 0,
20         "ontime": "2025-05-23T20:20:01.419684Z"
21       },
22       {
23         "sensing_service_id": 2825,
24         "value": 1,
25         "ontime": "2025-05-23T20:20:01.419684Z"
26       },
27       {
28         "sensing_service_id": 2826,
29         "value": 20.6,

```

Figure 5 Sample of the Smart Farming “/v2/entities” endpoint.

2.1.1.3 Smart Farming User-Interface

The Smart Farming application provides a graphical point of interaction for the end-users of the system. The application is based on state-of-the-art technologies and applies the best practices of UI/UX to provide intuitive user experience. The application is built with the React framework and provides

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	25 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

adaptability to a wide range of screen sizes and devices. The user first interacts with the login page, Figure 6, which uses Keycloak authenticator for a secure user login. Upon successful identification the user is redirected to the Home View, Figure 7 and Figure 8. In this view general information about the deployed devices is demonstrated, such as the device's name, the location of the device on the map, device's serial number for identification and date time information. Moreover, to enhance the user experience, the integrated map view provides several options such as terrestrial and satellite representations.

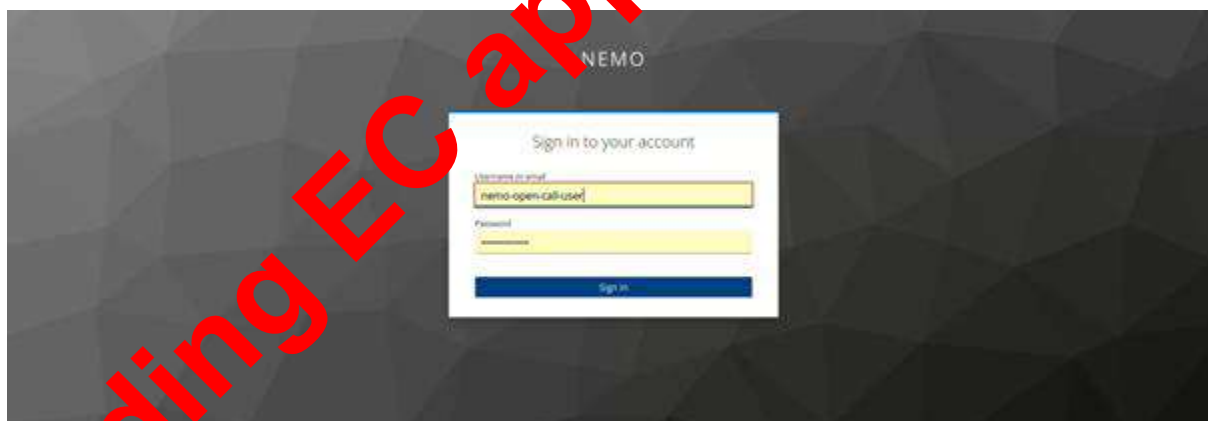


Figure 6 Smart Farming Login Page.

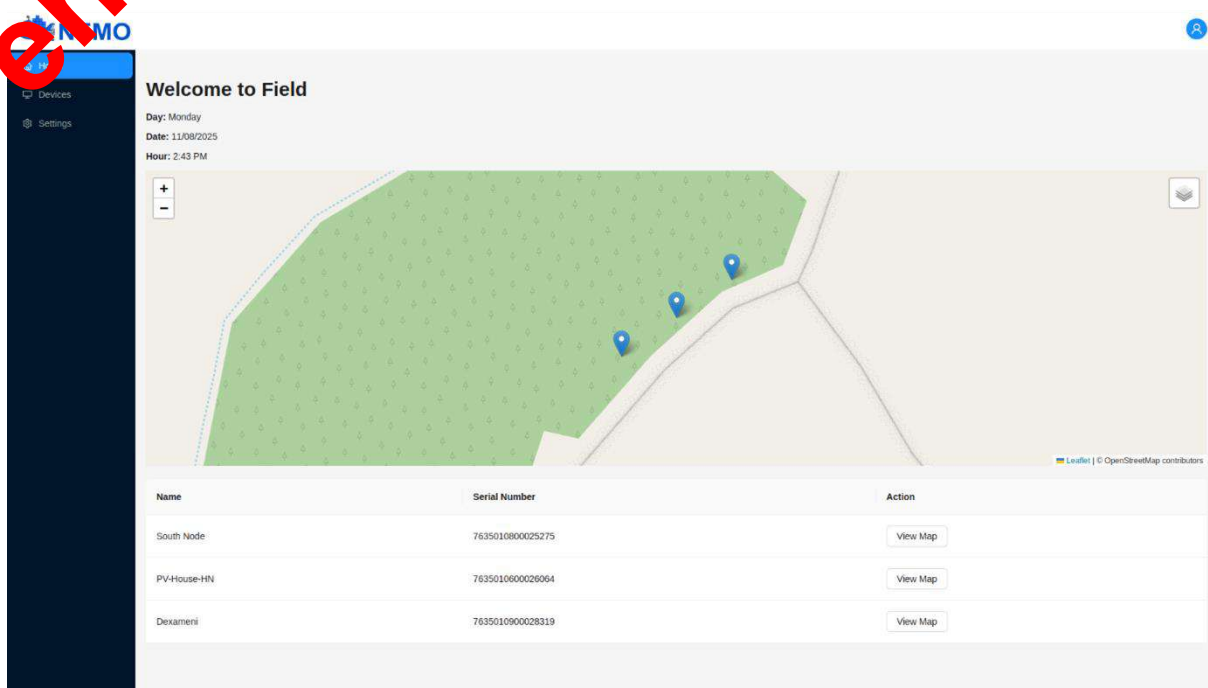


Figure 7 Home View, terrestrial map.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	26 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

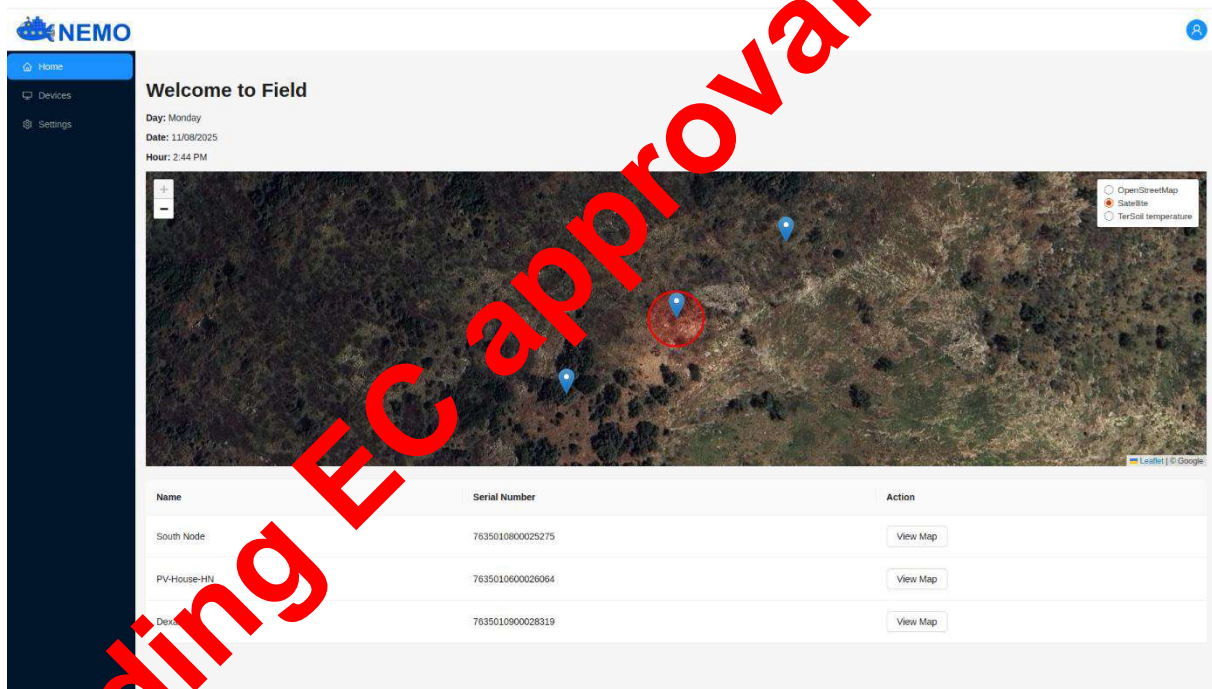


Figure 8 Home View, satellite map

In the Devices submenu the user can see an extended set of information regarding the devices deployed. The device list presented is dynamical and it is altered based on the registered devices per user. Within the context of UI/UX, an identical menu per device (namely Dashboard, Sensors, System and Location) was adopted to avoid user confusion among different device types. Figure 9, Figure 10 and Figure 11 The views are similar for the rest of the devices, as presented in Section 2.1.1.2 an abstract REST API is responsible for providing to the User-Interface or any 3rd party service the values from the sensors collected. In this way, new sensors can be integrated to the User-Interface with little effort.

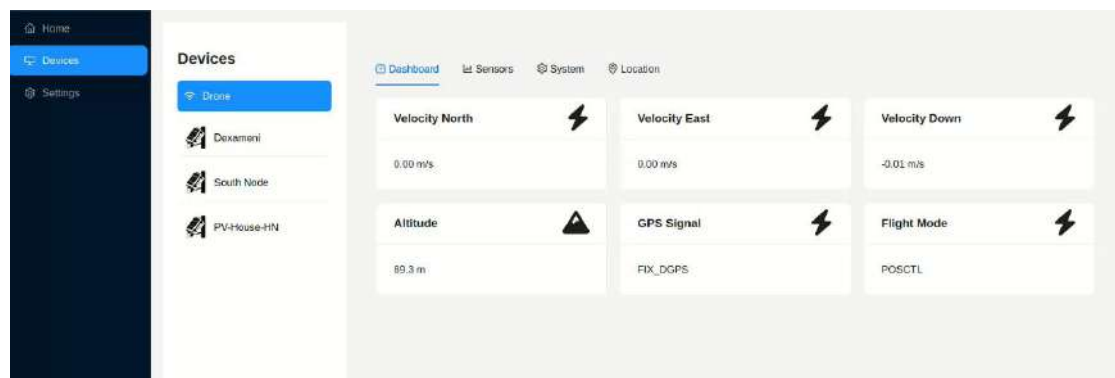


Figure 9 Dashboard submenu.



Figure 10 Sensor submenu.

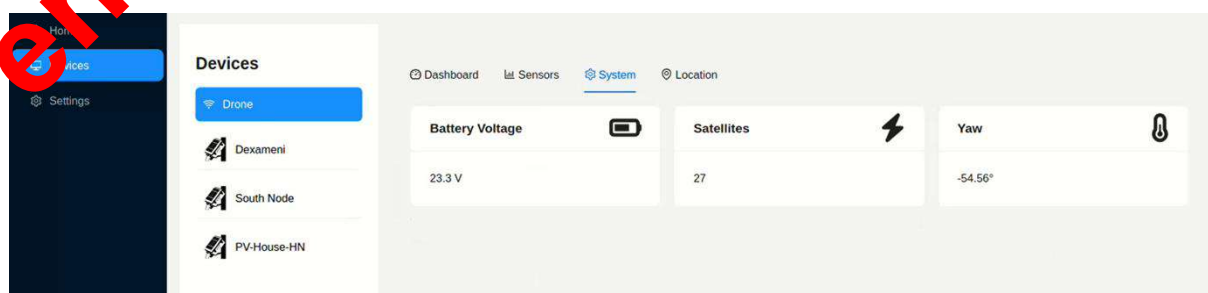


Figure 11 System submenu.

2.1.2 Test scenarios verification

In deliverable D5.3[4] all the advancements related to the Smart Farming were reported with the remaining once being a) the completion of the User-Interface, b) testing and evaluation of the automated flying path and c) the Cluster registration and workload migration.

In deliverable D5.4 the three remaining categories are presented, more specifically in Section 2.1.1.3 the final User-Interface was presented. For the purpose of testing the autonomous a series of test flights took place in Agia Sofia estate. In Figure 12 the deployment setup before the test flight begins is depicted, the part selected has a small opening to take-off and land the drone with safety and upon reaching the denoted height the waypoint algorithm will be initiated. It is noteworthy that at the beginning of the pilot a study was carried out to define the appropriate height of the drone to fly to ensure both the security of the UAV and the high precision image capturing based on the on-board camera of the UAV. While in Section 2.1.1.1.2 the algorithms of the waypoint navigation, image capturing, etc were explained, in this Section the focus will be on the high-level experiment test case.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	28 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

The purpose of this test was to validate the effectiveness of the waypoint autonomous navigation and the center extraction algorithms. The test was successful with the UAV cruising from a waypoint to another, performing the center extraction process of the olive tree spots, transmitting the geolocation data and proceeding to the next olive tree. In Figure 13, Figure 14 and Figure 15 a sample output of the algorithms and messages exchanged are presented while in Figure 16 and Figure 17 the output of the information received over the to the graphical user interface of the flying controller and the UAV performing center extraction are presented respectively.



Figure 12 Preflight check at Agia Sofia Estate

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	29 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

```
src.event_listener - INFO - Initializing Event Listener connection...
src.event_listener - INFO - Connecting to RabbitMQ at amqp://user:password@192.168.1.236:30207/
src.event_listener - INFO - Event listener ready and consuming messages from center-reached queue
src.mavsdk_helper - INFO - Connection attempt completed, checking connection state...
src.mavsdk_helper - INFO - Waiting for drone connection to stabilize...
src.mavsdk_helper - INFO - Drone connected successfully!
src.mission_manager - INFO - Setting mission speed to 5.0 m/s on drone...
src.mavsdk_helper - INFO - Setting mission speed to 5.0 m/s
src.mavsdk_helper - INFO - Mission speed parameter set to 5.0 m/s
src.mavsdk_helper - INFO - Current mission speed: 5.0 m/s
src.mission_manager - INFO - Verified mission speed on drone: 5.0 m/s
src.mission_manager - INFO - Forcing mission reload to ensure clean state...
src.mavsdk_helper - INFO - Forcing mission reload...
src.mavsdk_helper - INFO - Clearing mission
src.mavsdk_helper - WARNING - Could not verify mission 'clear' 'Mission' object has no attribute 'mission_count'
src.mission_manager - INFO - Creating mission plan...
src.mission_manager - INFO - Creating mission plan...
src.mission_manager - INFO - Added waypoint: lat=47.3972, lon=8.5452, alt=20
src.mission_manager - INFO - Added waypoint: lat=47.3972, lon=8.546, alt=25
src.mission_manager - INFO - Added waypoint: lat=47.3977, lon=8.5467, alt=30
src.mission_manager - INFO - Uploading mission...
src.mavsdk_helper - INFO - Uploading mission...
src.mavsdk_helper - INFO - Clearing existing missions
src.mavsdk_helper - INFO - Mission uploaded successfully with 3 waypoints
```

Figure 13 Sample output from the Inference algorithm.

```
src.mavsdk_helper - INFO - Setting preflight checks...
src.mavsdk_helper - INFO - Global position estimate OK
src.mavsdk_helper - INFO - Global position OK
src.mavsdk_helper - INFO - Drone is armable
src.mission_manager - INFO - Arming and taking off...
src.mavsdk_helper - INFO - Arming the drone
src.mavsdk_helper - INFO - Taking off to 10 meters
src.mission_manager - INFO - Starting mission...
src.mavsdk_helper - INFO - Starting mission
src.mission_manager - INFO - Monitoring mission progress...
```

Figure 14 Sample output from the waypoint navigation algorithm.

```
main - INFO - Starting the application
main - INFO - Bucket name: inference-api
main - INFO - MinIO helper initialized
main - INFO - CenterExtraction initialized
main - INFO - Scaling factor downloaded
src.event_publisher - INFO - Initializing Event Publisher
main - INFO - EventPublisher initialized
src.event_listener - INFO - Initializing Event Listener
main - INFO - EventListener initialized
main - INFO - Starting listener
src.event_listener - INFO - Initializing Event Listener connection...
src.event_listener - INFO - Connecting to RabbitMQ at 192.168.1.236:30207
src.event_listener - INFO - Event listener ready and consuming messages from result captured
src.event_listener - INFO - Received event: {'eventName': 's3ObjectCreated:Put', 'key': 'inference-api/results/9c56feda-4aec-48ce-91b3-e71ef93d43a.png', 'Records': [{'eventVersion': '2.0', 's3Region': '', 'eventName': '2025-08-08T11:10:26.032Z', 'eventTime': '2025-08-08T11:10:26.032Z', 'userIdentity': {'principalId': 'admin'}, 'requestParameters': {'principalId': 'admin', 'region': '0.0'}, 'responseElements': {'x-amz-id-1': 'd99025bab4ad464b049177c95abdeb374d3b1f0a9251148b658d77ac2a3e8', 'x-amz-request-id': '1859c6200495677d', 'x-minio-deployment-id': '85f6636f-a88', 'x-minio-origin-endpoint': 'http://localhost:9000'}, 's3': {'s3SchemaVersion': '1.0', 'configurationId': 'Config', 'bucket': {'name': 'inference-api', 'ownerIdentity': {'principalId': 'admin'}, 'pl': 'object': {'key': 'results/9c56feda-4aec-48ce-91b3-e71ef93d43a.png', 'size': 8739, 'etag': 'c93ef651677e401ab11347cf4292d6cd', 'contentType': 'image/png', 'userMetadata': {'content': '84099e11-d551-4b3e-a06a-03d4586c0fac', 'sequencer': '1859c620052CC9C9'}}}, 'source': {'host': '10.244.0.0', 'port': '', 'userAgent': 'MinIO (Linux; amd64) minio-go/v7.0.76 MinIO Console/

src.event_listener - INFO - Extracted MinIO path: results/9c56feda-4aec-48ce-91b3-e71ef93d43a.png
src.utils_minio_helper - INFO - Downloaded and decoded image from MinIO: results/9c56feda-4aec-48ce-91b3-e71ef93d43a.png
src.utils_minio_helper - INFO - Image metadata: {}
src.center_extraction - INFO - Found 15 contours.
src.center_extraction - INFO - Mask center coordinates: (324, 302)
src.center_extraction - INFO - Original mask center: (324, 302) in image size (1024, 768)
src.center_extraction - INFO - Extracted GPS coordinates: lat=0.0, lon=0.0
src.center_extraction - INFO - EXIF orientation 1 -> rotation 0
src.center_extraction - INFO - Rotated mask center: (324, 302)
src.center_extraction - INFO - Pixel offsets from center: dx=-188.0, dy=-82.0
src.center_extraction - INFO - Ground distances: dx=-5.64m, dy=-2.46m
src.center_extraction - INFO - Final GPS coordinates: lat=-2.213444425449212e-05, lon=-5.074726243712828e-05
src.utils_minio_helper - INFO - Uploaded JSON to MinIO: center-coords/gps coords 324_302.json
src.event_listener - INFO - Extracted center GPS: {'lat': -2.213444425449212e-05, 'lon': -5.074726243712828e-05}
src.event_publisher - INFO - Event Publisher connection initialized
src.event_publisher - INFO - Published event to center_coors_extracted: {'message': {'lat': -2.213444425449212e-05, 'lon': -5.074726243712828e-05}}
```

Figure 15 Sample output of the center extraction algorithm.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	30 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



Figure 16 Output of the flying controller component presenting the path followed by the UAV as well as telemetry data.



Figure 17 UAV hovering over an olive tree performing center extraction.

Finally, the most important aspect of the Pilot was to integrate with the NEMO framework. In this context three main operations were identified: a) Cluster registration, b) Workload migration and c)

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	31 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Cluster deletion. To successfully register the Smart Farming cluster a set of requirements needed to be fulfilled, such as deploying a compatible Kubernetes cluster and installing on the cluster the NEMO toolkit. This procedure was achieved by closely collaborating with the NEMO core tech team and using the FluxCD CI/CD tool, information regarding the CI/CD pipeline of NEMO has been reported in deliverable D4.3[5]. After the completion of these steps, the cluster was registered to the NEMO platform using the graphical User-Interface of the NEMO component, Figure 18. Within the Smart Farming Use Case the bottleneck process that could benefit from the Intent oriented approach of NEMO was the inference process. The Smart Farming cluster is based on Edge computing hardware and for an efficient execution of the inference pipeline more processing powerful hardware would be required. NEMO offers the cross-Pilot workload execution, and a GPU based hardware was used within the premises of OneLab. In Figure 19, Figure 20 and Figure 21 the process of registering the workload, creating the Intent condition that will trigger the migration process as well as the successful migration are presented.

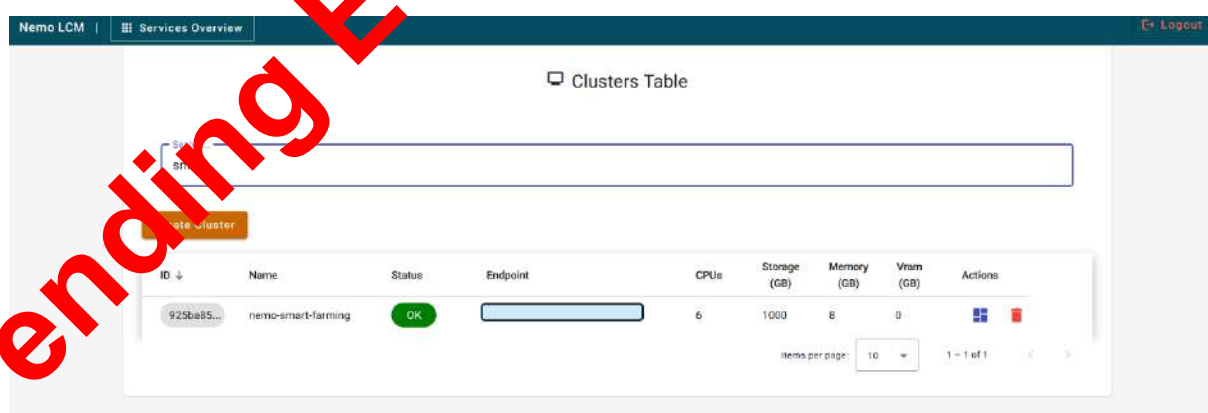
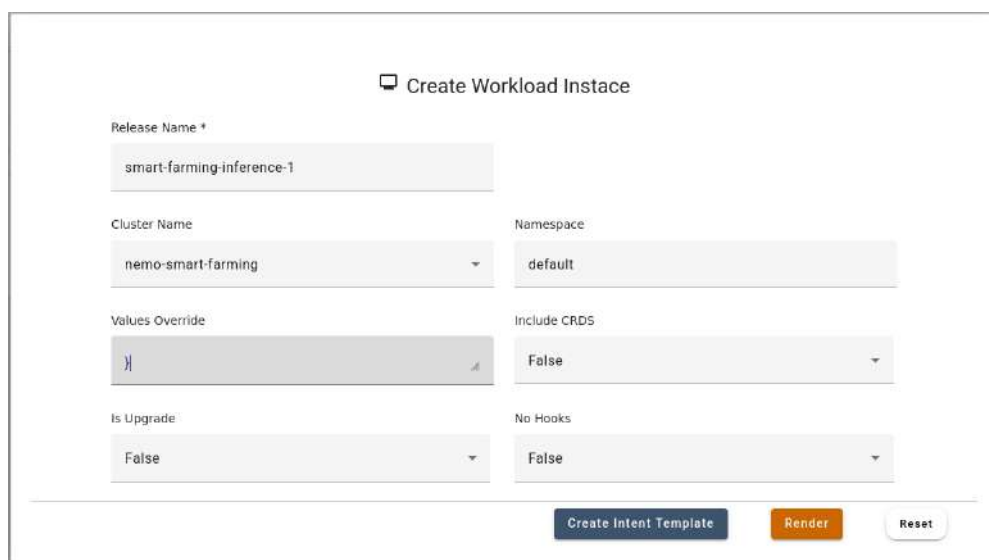


Figure 18 Smart Farming cluster successfully registered at NEMO.



The 'Create Workload Instance' form contains the following fields and values:

- Release Name ***: smart-farming-inference-1
- Cluster Name**: nemo-smart-farming (dropdown)
- Namespace**: default
- Values Override**: (empty text area)
- Include CRDS**: False (dropdown)
- Is Upgrade**: False (dropdown)
- No Hooks**: False (dropdown)

At the bottom, there are three buttons: 'Create Intent Template' (blue), 'Render' (orange), and 'Reset' (grey).

Figure 19 Smart Farming Inference workload registration.

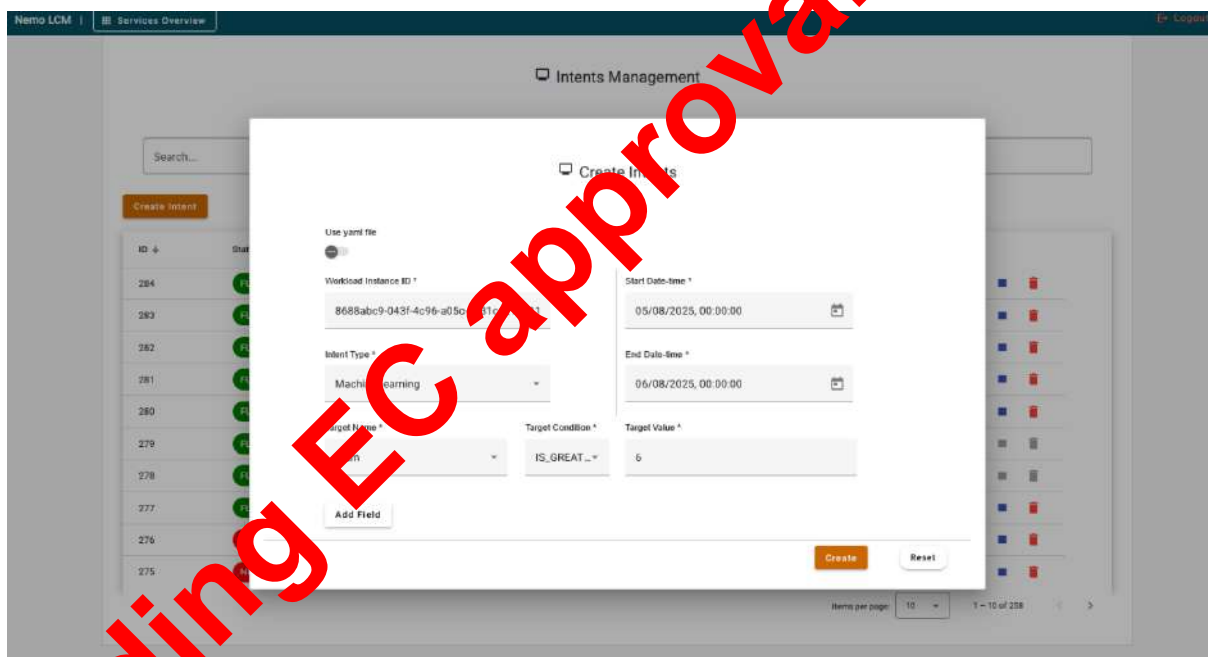


Figure 20 Smart Farming create trigger Intent.

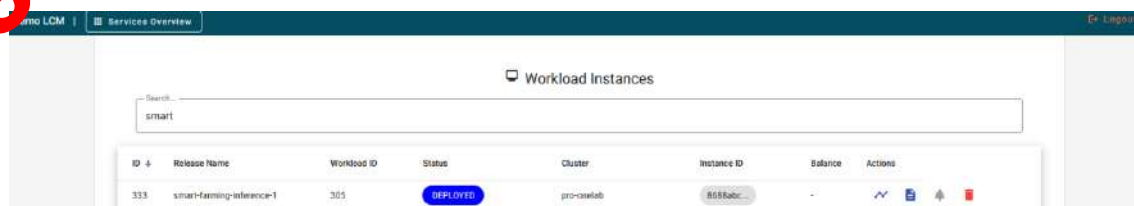


Figure 21 Workload successfully migrated to the production cluster in OneLab.

2.1.3 Final validation results and KPIs Evaluation

The KPIs along with the functional and non-functional requirements defined in the deliverables of WP1 and WP5 were used as intermediate checkpoints towards the completion of the Pilot. The success of the use case activities was based upon the satisfaction of those rail guards introduced at the beginning of the project. In addition to the technical success of the Piloting activities, the Smart Farming use case achieved also the user acceptance. More precisely it was calculated that the bio-spraying method proposed provides a significant reduction in the amount of pesticides used as only the affected surface of the tree is treated and only if a tree is affected, while in traditional procedures the farmers used to spray the entire tree (KPI_SF_01_1). Moreover, the automated monitoring of the olive trees and the on-time treatment is estimated to increase both the quality and the amount of harvested compared to the traditional methods (KPI_SF_01_2). One of the most important aspects in precision agriculture or Agriculture 4.0 is the detailed and real-time knowledge of the environmental conditions, to this end, a wide range of sensors were deployed in the piloting site providing information among others such as temperature, humidity and solar radiation. In addition, the UAV is transmitting to the Smart Farming application a variety of data such as telemetry and operational data (KPI_SF_01_3). Finally, through

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	33 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

severe tests and the specifications' datasheet of the GPS module used, the maximum difference in

Questionnaire SF Pilot

User Feedback										
1	Please rate the usability of the UAV in the field.									
	1	2	3	4	5	6	7	8	9	10
2	Please rate the usefulness of the UAV for bio-spraying.									
	1	2	3	4	5	6	7	8	9	10
3	Please rate the effectiveness of aerial bio-spraying.									
	1	2	3	4	5	6	7	8	9	10
4	Please rate the effectiveness of environmental information for the bio-spraying decision.									
	1	2	3	4	5	6	7	8	9	10
5	Please rate how beneficial the aerial spraying applied in NEMO is for managing olive fruit fly in your region.									
	1	2	3	4	5	6	7	8	9	10
6	Please rate how beneficial the aerial bio-spraying applied in NEMO is for increasing olives' yield.									
	1	2	3	4	5	6	7	8	9	10
7	Please rate how important you think Smart Farms' collaboration is for Smart Farming.									
	1	2	3	4	5	6	7	8	9	10
8	Please evaluate the possibility for you to adopt a similar solution like NEMO.									
	1	2	3	4	5	6	7	8	9	10

Figure 22 questionnaire with the farmer's feedback

precision was much less than 10 cm from the actual latitude and longitude position even during windy conditions (KPI_SF_02_1). During the last test performed, the farmer was interviewed to provide feedback about the usefulness of the use case. The farmer mentioned that the use case provides many benefits to the farming processes with the major one being the reduction of effort required to perform time-consuming yet trivial operations. The minimization of both the effort and financial aspects required in the agriculture domain and achieved with the piloting activities of T5.2 are beneficial for leading to a better and more automatized farming process, Figure 22 presents the questionnaire described in D5.2 with the farmer's feedback.

2.1.4 Lessons Learned and Replication guidelines

Overall, the automations inherited in the NEMO framework such as the FluxCD CI/CD pipeline, offer a fast replication and deployment procedure. In case of replicating or scaling up the Smart Farming application, the farmer needs to replicate to his cluster the HELM files of the microservices and connect the local cluster with the NEMO platform. A scale up can also be achieved by the inherited features of NEMO, such as the Intent-based migrations in more powerful clusters.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	34 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

3 Smart Energy & Smart Mobility Trial

Electricity distribution grids are becoming more complex due to the increasing penetration of renewable energy sources (RES), such as solar and wind, along with the decentralization of generation and electrification of consumption. These changes, while essential for achieving sustainability goals, introduce major technical challenges for Distribution System Operators (DSOs). The grid is now more dynamic and less predictable, leading to critical issues such as voltage instability, Reverse power flows (RPF), current imbalances, and increased fault occurrence. Traditional grid management approaches are no longer sufficient to ensure reliability, safety, and efficiency under these conditions.

Smart energy (SE) use case in the NEMO project directly addresses this real-world problem by aiming to improve real-time observability and fault response in urban distribution networks. The focus is on enabling faster detection, accurate classification, and precise localization of grid disturbances, especially in Medium-Voltage (MV) feeders where the impact of instability can cascade across the system. NEMO contributes to SE by enabling a smart, coordinated approach to grid management based on real-time data and distributed intelligence. It allows for the autonomous detection of disturbances through high-frequency sensing and local event processing at the edge. At the same time, it supports intelligent decision-making by orchestrating data exchange and analytics across edge and cloud layers. When faults occur, the system ensures a fast operational response by delivering insights to operators via monitoring dashboards and event logs. Furthermore, it provides a secure and flexible framework for managing the grid, including remote firmware updates, orchestration of workloads, and enforcement of operational policies.

By combining these capabilities, NEMO enables DSO's to shift from a reactive approach to grid operation toward a proactive, self-adaptive model. In this new paradigm, disturbances are not only detected at an early stage but also classified and localized automatically, allowing DSO to respond quickly and effectively with minimal manual intervention. This transformation supports the creation of a more resilient, efficient, and intelligent distribution network fully aligned with the goals of the European energy transition and the operational needs of DSO's evolving grid.

3.1 Smart Grid - Hierarchical Grid Disturbance Mitigation

3.1.1 Set up and Integration

The Smart Energy pilot at ASM integrates Phasor Measurement Units (PMUs) with the NEMO platform to enhance observability and control of the distribution network. PMU data is transmitted bi-directionally between far-edge sites and the central NEMO cluster. Remote management of far-edge sites is enabled via OTA services, which aggregate data from distributed nodes and support continuous inspection. While digital twin operators use OTA to monitor far-edge nodes, site operators employ the PMU Analyzer to validate measurements and ensure operational reliability Figure 23.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	35 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

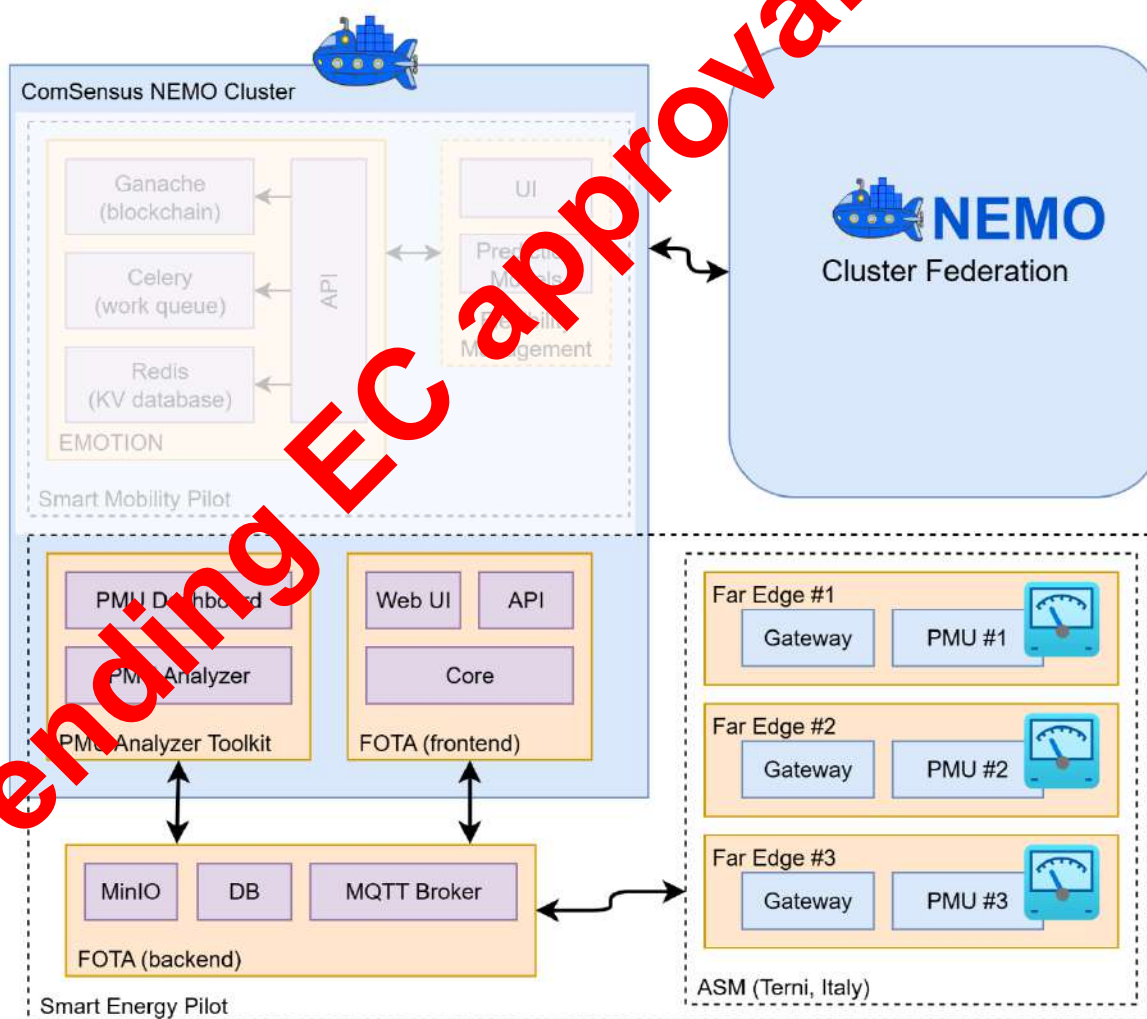


Figure 23 Smart Energy pilot with ASM's PMU sites connected to the NEMO cluster via OTA services and PMU Analyzer

Since in D5.3[4], the physical infrastructure of the SE-01 pilot has been fully deployed and finalized. This includes the successful installation and configuration of the Phasor Measurement Units (PMUs) at the selected substation and the on-site deployment of the NERVES-based gateway node. Figure 24 shows the installation of PMUs and Gateways in the ASM Terni living lab.

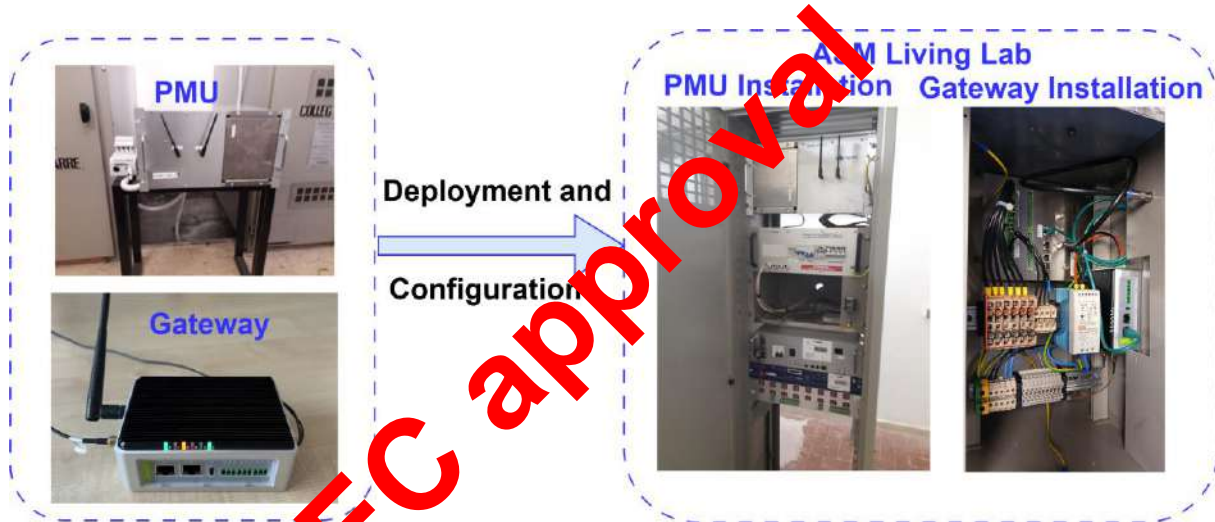


Figure 24 PMUs and gateways installation at ASM Terni

To support the description of the integration activities and visualize the interaction between deployed components, Figure 25 illustrates the architecture of the SE-01 pilot infrastructure for PMUs. It highlights the data and control flow between the edge elements comprising NERVES-based gateway nodes and on-premises PMUs and the cloud-based NEMO services, including the FOTA system, MQTT brokers, storage, and MetaOrchestrator.

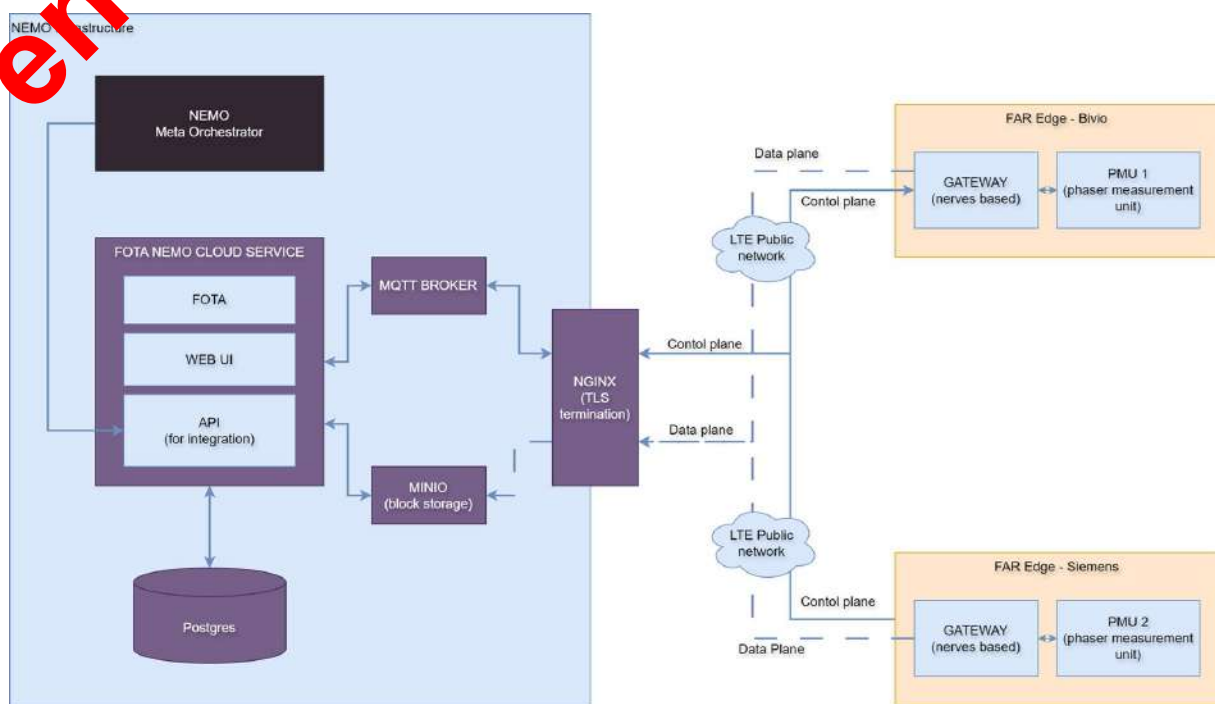


Figure 25 NEMO deployment schematic

Moreover, in D5.3 [4], the LTE-based communication channel has been validated and is operational, ensuring secure and reliable connectivity between the edge components and the central platform. In addition, high frequency monitoring at the edge has been activated, with continuous 50 Hz data collection from PMUs now functional. The system supports local event detection, and high-intensity data parsing and alerting mechanisms have been implemented and are currently operational.

The first stage of the GUI Management Dashboard has been finalized and is now integrated into the operational workflow. The dashboard offers a web-based interface that provides authorized users with direct access to real-time PMU measurements, historical logs, and event-driven alerts. Following Figures show us some screenshots of GUI dashboard.

Figure 26 screenshot displays the Over-the-Air (OTA) update interface used within the SE_01. It enables remote firmware upgrade of edge components by uploading a list of target devices and selecting the appropriate firmware version. The system supports batch operations and provides details such as file size and release date to ensure traceable, secure updates across the infrastructure.

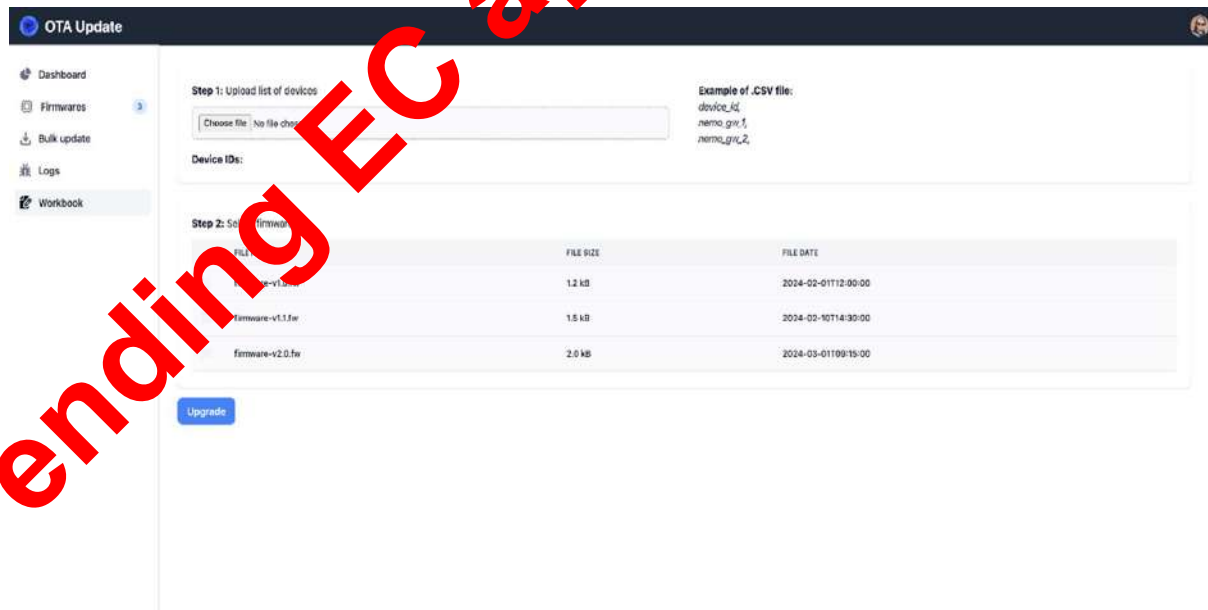


Figure 26 OTA Update interface – firmware selection for Edge devices

Figure 27 screen shows the user interaction panel of the NEMO OTA Update system deployed in SE_01. It allows operators to input control messages and user credentials before executing bulk update commands, supporting structured firmware distribution and accountability across multiple edge devices, such as PMU gateways deployed in ASM’s smart grid pilot.

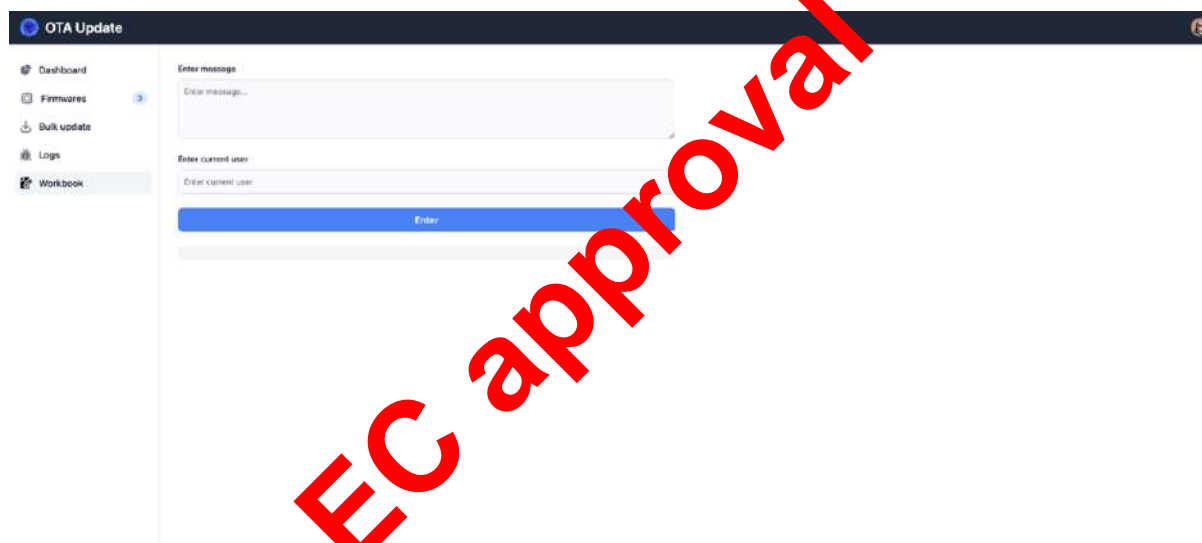


Figure 27 OTA update – user interaction panel for bulk firmware actions

Figure 28 interface displays the real-time status of OTA firmware updates for edge devices deployed in the SE_01 pilot. It confirms online connectivity, current software versions, and ongoing update processes, demonstrating successful device communication and centralized control via the NEMO platform.

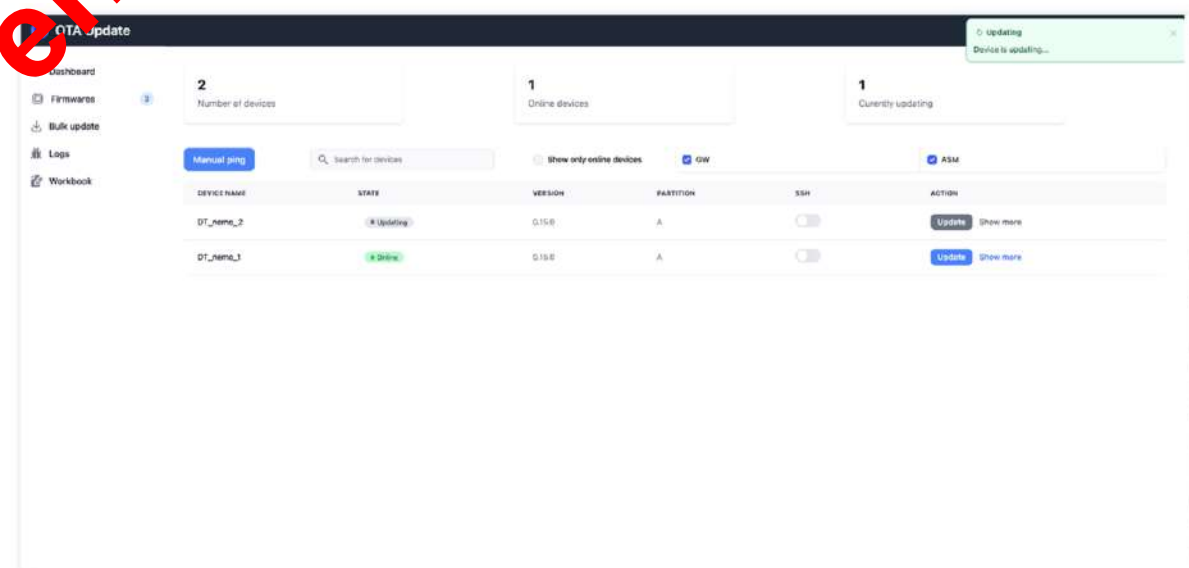


Figure 28 OTA update – live device status during firmware upgrade in SE_01

Figure 29 presents the detailed status of an individual edge device deployed in the SE_01 pilot, including active firmware partitions and the latest phasor measurements. It confirms device availability and provides operators with critical power quality data for situational awareness and grid diagnostics.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	39 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

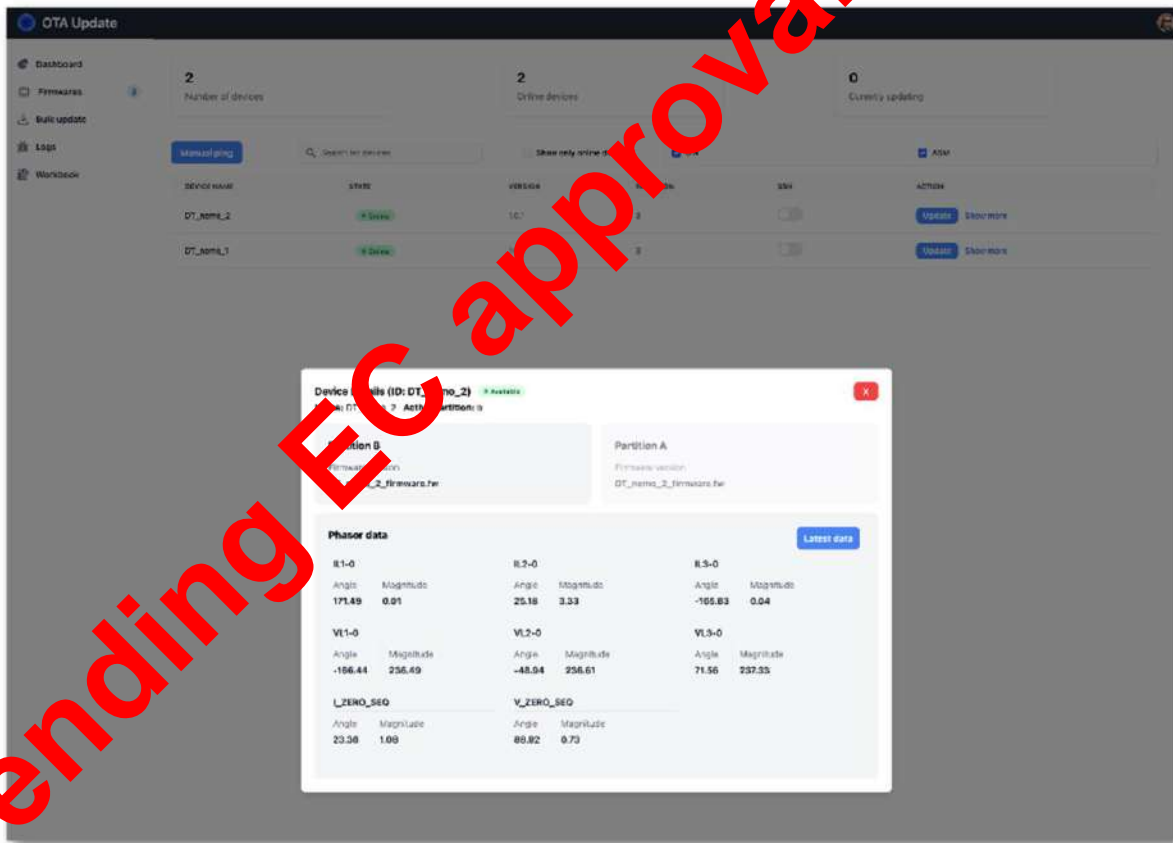


Figure 29 Detailed device view with real-time phasor data in SE_01

On the platform side, the full NEMO cloud stack has been successfully deployed and interconnected with the edge infrastructure. This includes containerized microservices for MQTT-based data exchange, MINIO for scalable object storage, and FOTA services for remote firmware management. All services operate under the Kubernetes orchestration environment and are exposed through a secured NGINX gateway configured with TLS termination. This architecture ensures reliable, secure, and scalable communication between the distributed edge nodes and the centralized NEMO platform.

3.1.2 Test scenarios verification

Based on the test scenarios defined in D5.3[4], the SE_01 pilot in Terni has undergone a full round of functional validation. These tests were aimed at verifying data flow, system responsiveness, remote update capabilities, and operator interface functionality.

The results confirmed successful end-to-end operation of the deployed infrastructure. Communication between the edge devices and the central platform is stable, high-frequency data from the PMUs is being collected and analyzed, and key events are being detected and logged correctly. The LTE-based communication channel has proven reliable throughout the testing.

To support these capabilities, a dedicated GUI dashboard has been developed as you can see in the Figure 30 Figure 31 Figure 32 which allows real-time monitoring of PMU status, grid measurements, and detected disturbances. Additionally, the Firmware Over-the-Air (FOTA) interface was tested to remotely update the edge devices, with clear visibility of device status, firmware version, and update history. Figure 30 show us FOTA functionalities for SE-01 in NEMO architecture.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	40 of 195
Reference:	D5.4	Dissemination:	PU
Version:	6.0	Status:	Final



Figure 30 FOTA functionalities in platform

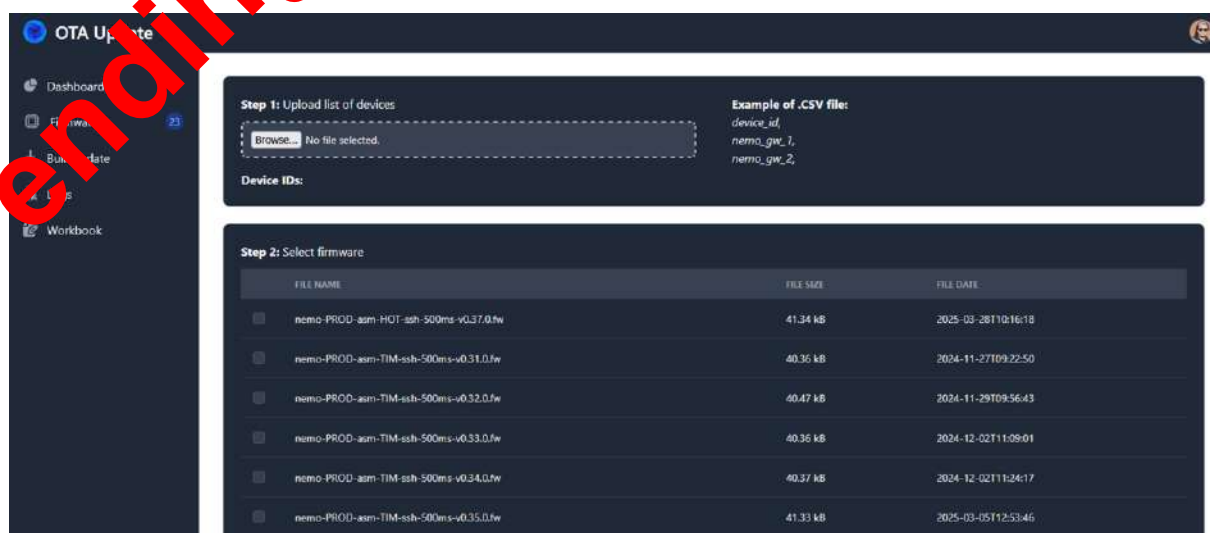


Figure 31 Firmware selection and device list uploadInterface in SE_01



Figure 32 SSH Connection logs and device activity timeline in SE_01

In addition, a PMU analyzer was used to check the incoming data, detect anomalies, and inspect time-series diagrams of the relevant electrical parameters. With these tools, operators have full access to both live and recorded data for incident investigation and operational awareness. Figure 33 demonstrates a home screen of the UI displaying the number of devices currently online and the total number of recorded historical issues. Figure 34 shows the live data recorded by the currently active devices with a granularity of 1 Hz. Lastly, Figure 35 provides a list of all recorded events that were triggered by exceeding the defined thresholds. These can be further examined by pressing the “Show more” button, which will display the time series representation of the variables recorded by the PMU at full (50 Hz) granularity, along with the vertical line, indicating the exact instant of threshold violation. For example, Figure 35 demonstrates the recorded transient behaviour of the frequency and the rate of frequency change during a single-phase to ground fault.

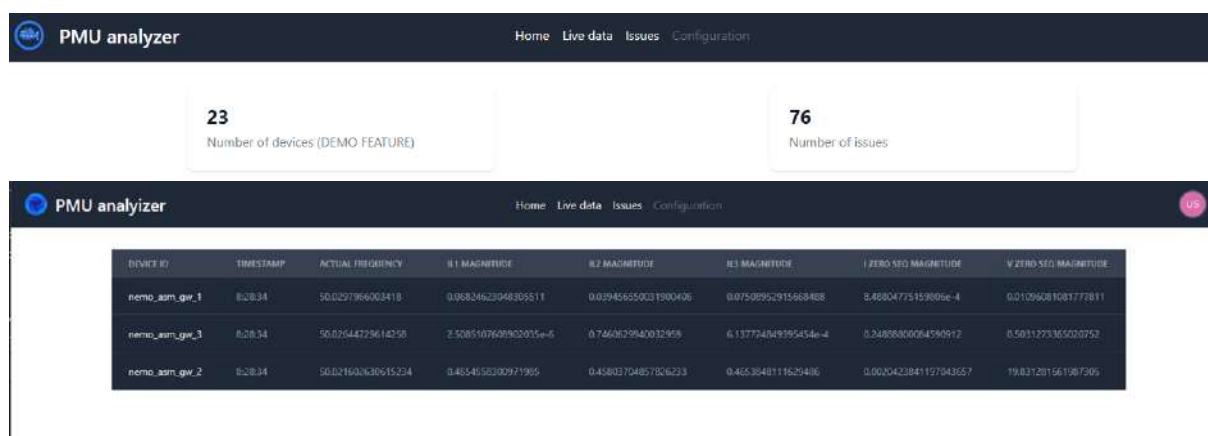


Figure 33 PMUs monitoring in dashboard platform

PMU analyzer		
Home Live data Issues Configuration		
ISSUE TIME	FILENAME	ACTION
2025-06-24T12:26:02.244Z	2025-06-24_12:25:51_dataframe_with_issues.gz	Show more
2025-06-24T12:25:50.123Z	2025-06-24_12:25:39_dataframe_with_issues.gz	Show more
2025-06-24T12:25:33.884Z	2025-06-24_12:25:23_dataframe_with_issues.gz	Show more
2025-06-24T12:25:09.767Z	2025-06-24_12:24:59_dataframe_with_issues.gz	Show more
2025-06-24T12:24:56.250Z	2025-06-24_12:24:45_dataframe_with_issues.gz	Show more
2025-06-24T12:24:33.925Z	2025-06-24_12:24:23_dataframe_with_issues.gz	Show more
2025-06-24T12:24:09.242Z	2025-06-24_12:23:58_dataframe_with_issues.gz	Show more
2025-06-24T12:23:34.891Z	2025-06-24_12:23:24_dataframe_with_issues.gz	Show more
2025-06-24T12:23:29.180Z	2025-06-24_12:23:18_dataframe_with_issues.gz	Show more
2025-06-24T12:23:10.345Z	2025-06-24_12:22:59_dataframe_with_issues.gz	Show more
2025-06-24T12:23:03.912Z	2025-06-24_12:22:53_dataframe_with_issues.gz	Show more
2025-06-24T12:22:49.123Z	2025-06-24_12:22:39_dataframe_with_issues.gz	Show more
2025-06-24T12:22:34.567Z	2025-06-24_12:22:10_dataframe_with_issues.gz	Show more
2025-06-24T12:22:15.134Z	2025-06-24_12:22:04_dataframe_with_issues.gz	Show more

Figure 34 PMU analyzer log of detected issues with timestamps and data files in SE_01



Figure 35 PMU analyzer issue visualization showing frequency and current magnitude anomalies in SE_01

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	43 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

3.1.3 Final validation results and KPIs Evaluation

Table 1 summarizes the KPIs previously defined and refined in D5.1, D5.2[3], and D5.3[4], followed by a detailed analysis of how each KPI was achieved.

KPI ID	Name	Description	Measurement/Assesment method	Target
KPI_SE_01_1	Time granularity	Time granularity for monitoring	Measure the field data amount collected	< 1s
KPI_SE_01_2	Number of measurements	Information exchanged by devices	Measure the field data amount collected	> 100.000 measurements/minute
KPI_SE_01_3	Number of sources	Reduce the probability of Smart Grid failure due to voltage instability at least	Evaluation of measured data leveraging data of the network	25 % reduction in comparison with daily average value
KPI_SE_01_4	CO2 reduction	Migrate micro-services between ASM and Green RWTH DC and reduce CO2 footprint	Measure the energy consumption	> 40%

Table 1 List of KPIs in SE_01.

KPI_SE_01_1

Time granularity for monitoring:

As part of the NEMO project, advanced monitoring capabilities have been implemented through the deployment of two PMUs within the selected distribution feeder. These devices enable high-frequency, real-time monitoring at the edge of the network, with a continuous data acquisition rate of 50 Hz, which corresponds to a measurement every 20 milliseconds.

This configuration ensures a highly granular and time-synchronized view of electrical parameters, supporting detailed observability and enabling the early detection of fast transients, voltage fluctuations, and potential anomalies. The PMUs are integrated into the existing SCADA environment and communicate with a local broker via the MQTT protocol, allowing for seamless and reliable data transfer to the backend infrastructure where the time-series data is stored and processed.

KPI_SE_01_2

Information exchanged by devices:

The monitoring infrastructure deployed in the SE_01 use case of the NEMO project consists of two high-frequency PMUs and six PQAs. These devices are installed at critical points within the medium-voltage distribution feeder to ensure detailed and time-synchronized visibility of grid conditions.

Each PMU is configured to measure 18 electrical parameters (4 voltage magnitudes, 4 voltage angles, 4 current amplitudes, 4 current angles, rate of change of frequency and frequency) at a rate of 50 Hz, resulting in a combined output of 108,000 measurements per minute. In parallel, the six PQAs collectively report (4 voltage magnitudes, 4 current angles, 3 active powers, 3 reactive powers, 6 PQMs, 60 seconds) approximately 5040 power quality variables per minute, based on their granularity of nearly 1-second and multi-variable reporting configuration.

Thanks to seamless integration with the SCADA system and MQTT-based communication, the infrastructure ensures efficient transmission and processing of high-frequency data streams, which

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	44 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

represents a significant advance over previous monitoring setups. The achieved measurement rate of 113,040 data points per minute reflects the high interaction capability of the NEMO infrastructure and confirms that this KPI target has been fully met.

[KPI_SE_01_3](#)

This KPI was validated in SE_02

[KPI_SE_01_4](#)

Migrate micro-services between ASM and Green RWTH DC and reduce CO₂ footprint:

To measure the CO₂ reduction achievable through microservice migrations, we set up a demo deployment. This demo deployment maximizes energy consumption by stressing all available CPU cores of a cluster by using replica sets. Spawning the deployment is done by creating a workload in NEMO's Plugin & Application Lifecycle Manager (LCM).

After spawning it on the dev-onelab cluster, we used NEMO's Meta Orchestrator's (MO) REST API to query the Privacy & Policy Enforcement Framework (PPEF) for the cluster's CPU usage as well as the cluster's energy consumption. We successfully increased the cluster's CPU usage from around 9 CPUs to around 55 CPUs (full utilization of the cluster). The cluster's energy consumption was measured to have increased from around 76,000 to around 83,000 joules per two minutes, which equals 630 and 690 watts, respectively, implying our service has a power impact of roughly 60 watts.

Migrating a deployment for decreasing the CO₂ footprint would usually be done through the LCM's interface system by creating an intent for the respective workload instance with a newer, higher green energy consumption rate. For example, migrating from dev-onelab with a reported green energy consumption rate of 0% to nemo-energy with a reported green energy consumption rate of 80% reduces the microservice's CO₂ footprint by 80%. With a carbon intensity of about 300 grams of CO₂ equivalents per kilowatt-hour at the time of the experiment, this would result in an annual reduction of roughly 160 kg of CO₂, confirming that this KPI target has been reached successfully.

[3.1.4 Lessons Learned and Replication guidelines](#)

The Smart Energy trial demonstrated that enhancing grid observability with PMUs and integrating them into the NEMO framework can significantly improve disturbance detection, classification, and operator response. The use case confirmed that real-time monitoring, coupled with intuitive dashboards, supports faster situational awareness and builds operator confidence in handling grid events. Feedback highlighted the value of automation and remote management, reducing manual intervention and maintenance effort.

In terms of replication, the pilot showed that the approach is not limited to ASM's feeder but can be extended to other DSOs with similar challenges. The modularity of the NEMO framework, together with its automation features, allows a straightforward transfer of the solution to different grid environments. By following the same deployment and orchestration procedures, utilities can achieve improved resilience, better integration of renewable sources, and measurable environmental benefits through optimized workload management.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	45 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

3.2 Smart Mobility

3.2.1 Set up and Integration

The Smart Mobility pilot demonstrates the integration of Electric Vehicle (EV) charging infrastructure with energy flexibility management through the NEMO platform Figure 36. In this setup, EV users submit energy bids that are processed by the flexibility management service deployed at COMS's NEMO cluster. The system evaluates offers, conducts an auction mechanism, and subsequently records the resulting transactions in EMOT's blockchain, ensuring transparency, security, and traceability of energy flexibility operations.

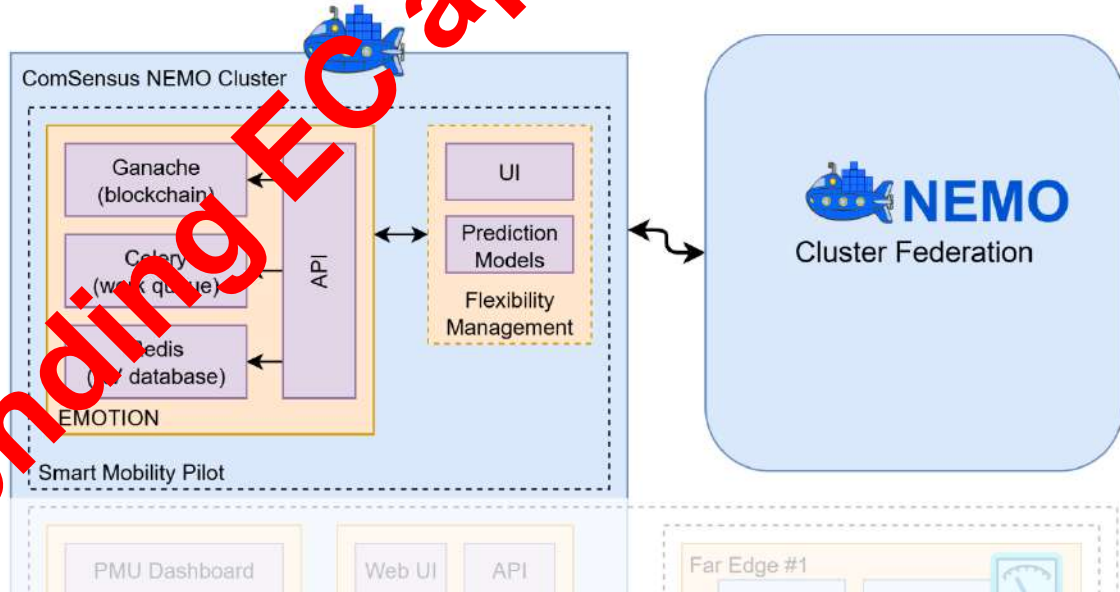


Figure 36 Smart Mobility pilot with EV flexibility management at COMS's NEMO cluster, linked to EMOT's blockchain.

In comparison with D5.3[4], the Smart Mobility trial within SE_02 has been extended with a dedicated Smart Parking system, deployed and integrated under the NEMO framework Figure 36. This sub-use case focuses on intelligent parking management and EV charger coordination, designed to optimize parking slot usage and enhance the visibility of EV charging availability in urban environments. The hardware infrastructure and communication interfaces for this setup have been finalized and successfully connected to the local trial cluster.

The Smart Parking system combines multiple data acquisition layers with real-time processing and display capabilities. The core components of the deployed system are included in Table 2

Device	Model/Name	Description / Features	Communication / Inter
Camera	Hikvision IDS-2CD7A26G0/P-IZHS	- 2MP DeepinView ANPR Moto Varifocal Bullet Camera - License Plate Recognition - Cybersecurity protection - Perimeter protection with deep learning	HTTP Listening Mode

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	46 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Device	Model/Name	Description / Features	Communication / Inter
I/O Device	ADAM 6066	- 6 Digital Inputs - 6 Digital Outputs (Relay)	Ethernet Modbus TCP/IP
EV Parking Sensors	Pepperl+Fuchs UC6000-30GM70-I2R2-V15	- Vehicle presence detection for EV charging slots	Digital Wiring (On/Off) via ADAM 6060
Information Display	Bios MX8-42-20INFOE	- 4 Text Rows, 20 Characters per Row - Red LED 5mm, 8 cm Character Height - Reading Distance: 24–32m	Ethernet ASCII Protocol
Server	Esxi Virtual Machine	- Windows Server 2019 Standard Edition	–
Database	Microsoft SQL Server 2019 Express Edition	–	–
SCADA & Drivers	AVEVA System Platform 2019	–	–

Table 2 Hardware Infrastructure and Functional Overview in SE 02

Figure 37 illustrates the complete data flow architecture of the Smart Parking system, beginning with the acquisition of vehicle presence and license plate information through ultrasonic sensors and a DeepinView ANPR camera. This data is transmitted to an edge/cloud-based server equipped with the AVEVA System Platform, where real-time processing algorithms analyze occupancy status, session duration, and charger availability. The processed information is then relayed to outdoor LED displays for user-facing visualization, enabling dynamic updates on parking and EV charging status.

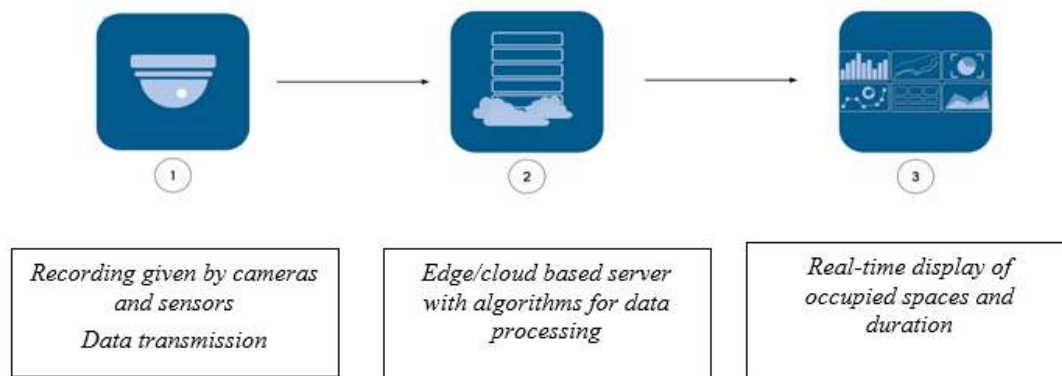


Figure 37 Smart Parking Data Processing Workflow

Figure 38 provides a view of the physical deployment at ASM Terni's headquarters, showing the integration of the ANPR camera, charging stations, and hybrid network video recorder (NVR) setup within a LAN-based infrastructure. This configuration ensures seamless connectivity between sensors, the processing unit, and the control system, allowing for both real-time monitoring and historical data logging via SCADA integration.



Figure 38 Smart Parking and EV Charging Infrastructure at ASM Terni

Finally, Figure 39 showcases the graphical user interface developed within the ASM control platform, where a centralized LED dashboard displays key metrics such as parking stall occupancy (left), EV charger availability (center), and both the number of free slots and the average daily parking duration (right). This end-to-end system enhances transparency, improves user guidance, and supports more efficient management of urban mobility resources.



Figure 39 Real-Time Display Interface for Parking and EV Charging Management.

Moreover, ASM, as data owner, has provided via MQTT broker the Smart Parking dataset which EMOT integrated into the CPO platform, as shown in Figure 40 below:

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	48 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

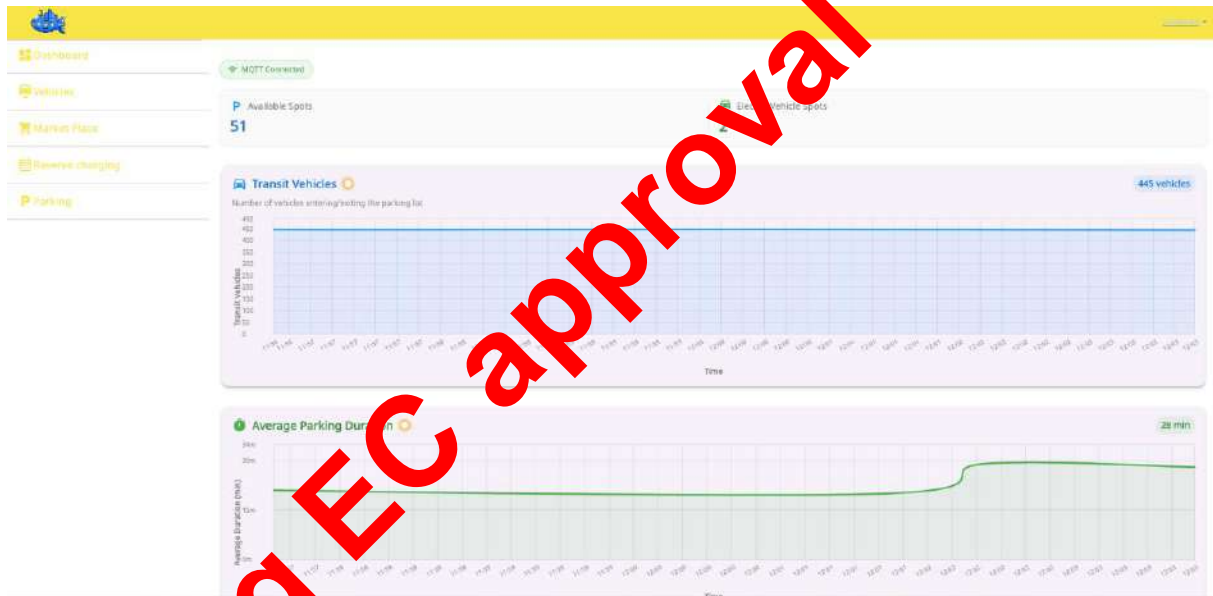


Figure 40 Smart Parking integration to CPO platform.

During the project, an internal analysis was conducted on the dataset collected from the Smart Parking infrastructure deployed in the SE_02 use case. This analysis aimed to better understand user behavior, vehicle flow patterns, and temporal occupancy dynamics based on real-world data collected throughout the operational phase of the pilot.

The dataset originates from the video surveillance and sensor system and spans approximately two months in 2025. It includes more than 6,770 parking session entries; each representing a unique vehicle's entry and exit into the monitored parking area. For each session, the system records the entry time, exit time, and total parking duration in minutes.

To ensure reliability, the raw data was pre-processed by aligning timestamps, removing inconsistencies, and filtering out erroneous records. The cleaned dataset was then enriched by extracting additional temporal attributes such as entry hour, day of the week, and week number.

This transformation enabled time-series aggregation and behavioral modeling, laying the groundwork for forecasting analysis and operational optimization. To facilitate downstream analysis, the data was aggregated into hourly intervals, capturing vehicle count per hour and average parking duration per interval.

The dataset exhibits a clear weekday and office-hour usage pattern, consistent with the operational nature of the facility. As expected in a semi-public or administrative parking environment, usage drops significantly during evenings and weekends, while peaks align with business hours.

To explore usage patterns and support KPI evaluations, several statistical summaries and visualizations were generated. Figure 41 shows the distribution of vehicle entries by hour of the day, revealing that the busiest periods occur between 08:00 and 12:00, aligning with morning work shifts. A smaller secondary peak is visible around 14:00–15:00, possibly linked to afternoon appointments or staff rotation. After 18:00, entries drop sharply, confirming the site's day-oriented usage profile.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	49 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

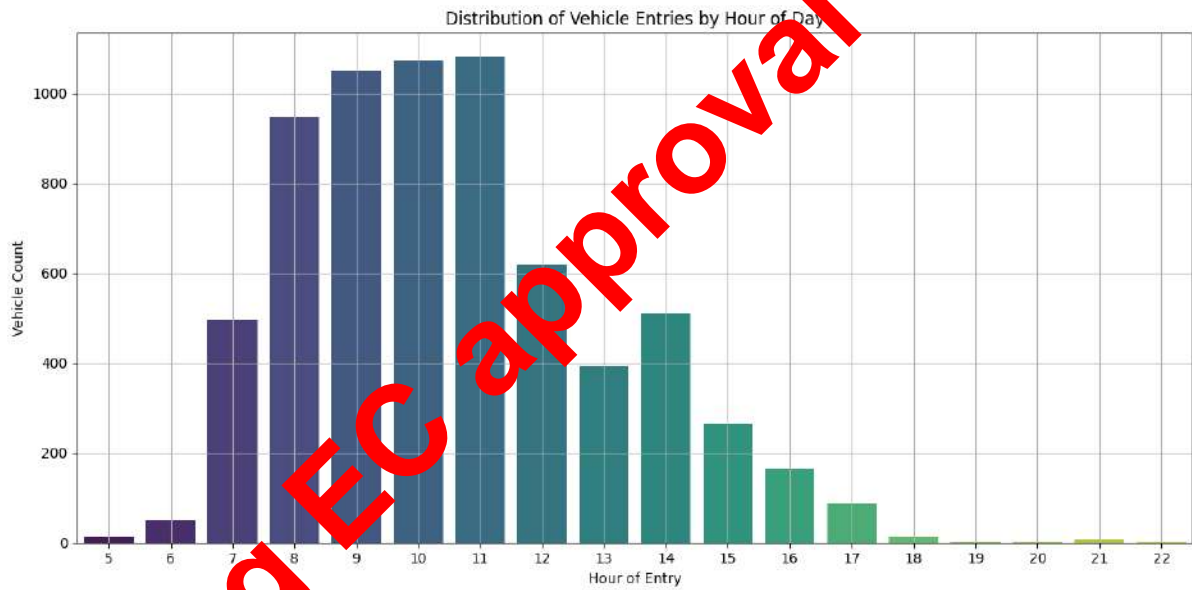


Figure 41 Distribution of vehicle entries by hour of the day.

Figure 40 presents the distribution of parking durations across all sessions. The pattern is right-skewed, with the majority of vehicles parked for less than 60 minutes. The average session duration is approximately 97.5 minutes, reflecting a high frequency of short-to-medium stay sessions, typical of urban mobility hubs or administrative facilities. However, long-tail behavior is also observed, with some sessions lasting over 11 hours, suggesting usage by staff or long-term parkers.

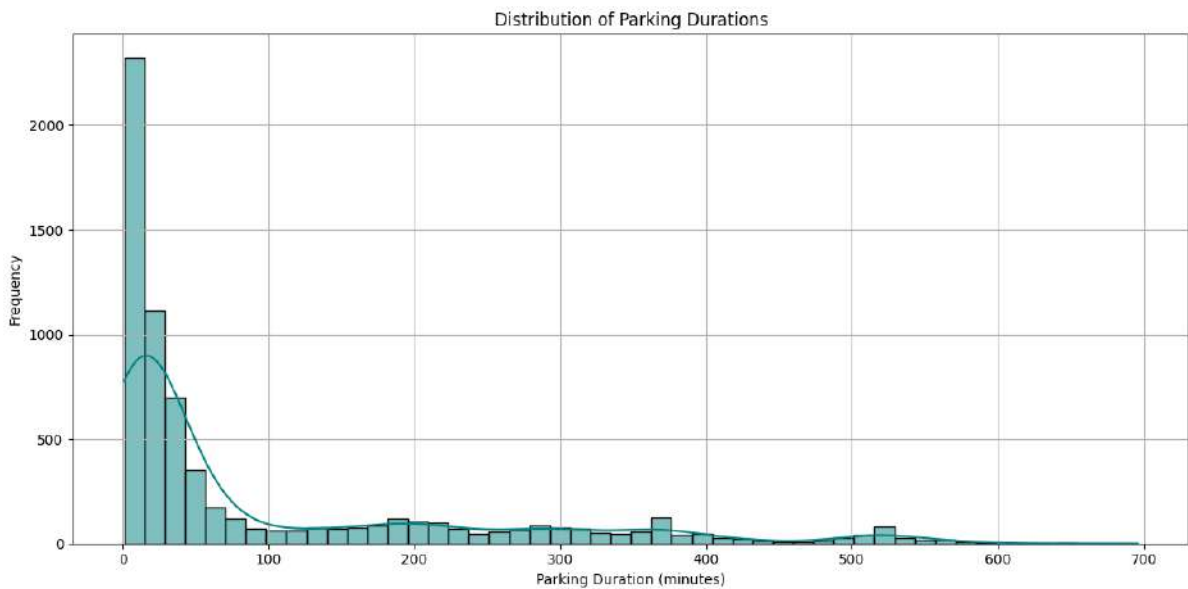


Figure 42 Distribution of parking durations across the full dataset.

These insights form a strong foundation for predictive modeling and optimization, further addressed in the upcoming sections on forecasting and operational efficiency. They also provide valuable inputs for future enhancements in parking space allocation, EV charging prioritization, and citizen-facing mobility services.

Moreover, the electric mobility web application and the flexibility marketplace have been containerized to create a consistent execution environment for their applications, enabling them to run reliably across NEMO infrastructure, as shown in Figure 43 Figure 44

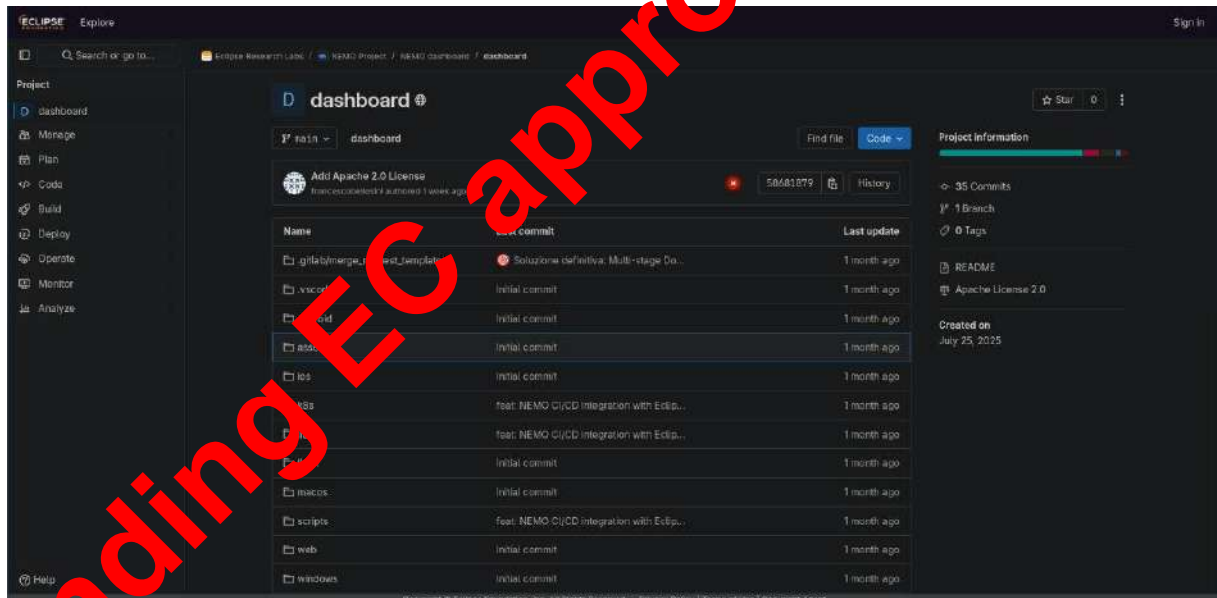


Figure 43 Electric Mobility Platform in Eclipse Foundation GitLab

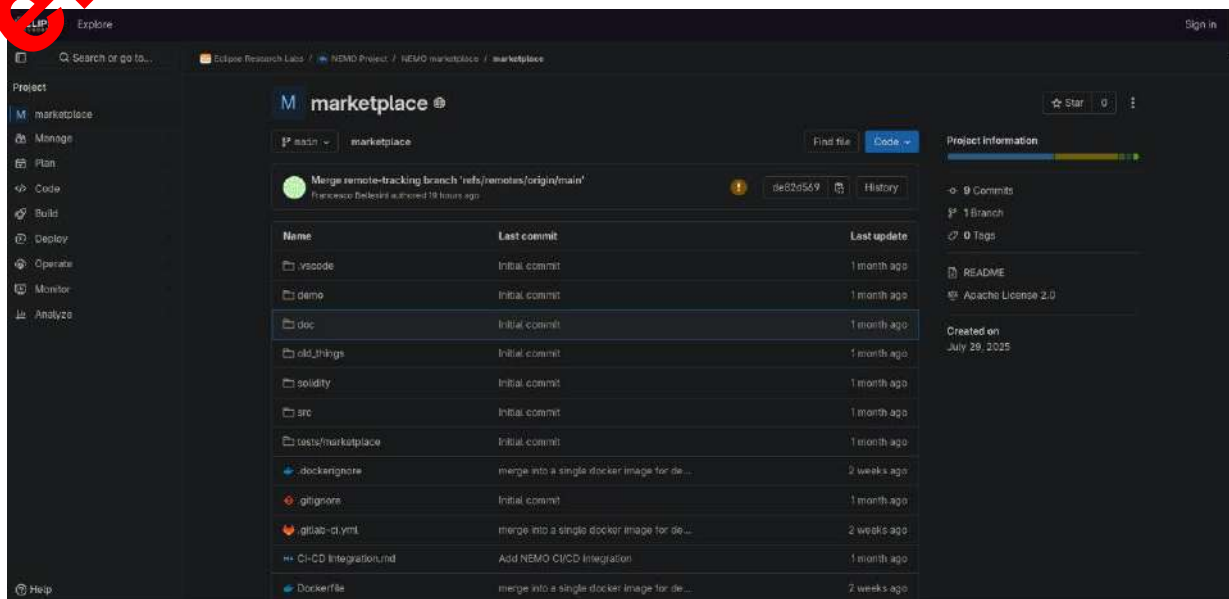


Figure 44 Flexibility Marketplace in Eclipse Foundation GitLab

Figure 45 Figure 46 show the YAML files needed by Eclipse to start the CI/CD Pipeline, which allows Eclipse to rebuild the electric mobility web application and the flexibility marketplace projects every time a push is made to the main branch. The pipelines were defined following the official guidelines described by the NEMO project.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	51 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

```

.gitlab-ci.yml 612 B
1 # NEMO CI/CD Pipeline following official integration guide line
2 # Based on Eclipse Foundation templates for automated build
3
4 include:
5   - project: 'eclipsefdn/it/releng/gitlab-runner-service/gitlab-ci-templates'
6     file: 'jobs/buildkit.gitlab-ci.yml'
7
8 stages:
9   - build
10  - test
11
12 variables:
13   CI_REGISTRY_IMAGE: nemometaos/omniboard_nemo
14
15 buildkit:
16   extends: .buildkit
17
18 unit-test:
19   stage: test
20   script:
21     - echo "Running unit tests... This will take about 10 seconds."
22     - find . -name $CI_REGISTRY_IMAGE:$CI_COMMIT_SHORT_SHA -script/to/run/tests
23     - sleep 10
24     - echo "Tests passed successfully!"

```

Figure 45 Electric Mobility Platform YML file

```

.gitlab-ci.yml 371 B
1 include:
2   - project: 'eclipsefdn/it/releng/gitlab-runner-service/gitlab-ci-templates'
3     file: 'jobs/buildkit.gitlab-ci.yml'
4   - project: 'eclipsefdn/it/releng/gitlab-runner-service/gitlab-ci-templates'
5     file: 'pipeline-autodevops.gitlab-ci.yml'
6
7 stages:
8   - build
9   - test
10
11 variables:
12   CI_REGISTRY_IMAGE: nemometaos/nemo-marketplace
13
14 buildkit:
15   extends: .buildkit

```

Figure 46 Flexibility Marketplace YML file

3.2.2 Test scenarios verification

The SE_02_Test_Scenario_1, defined in D5.3[4], was designed to validate an integrated urban mobility framework that leverages EV charging infrastructure to support renewable energy sources (RES) load balancing while enabling predictive traffic and parking analytics. The scenario aimed to demonstrate how coordinated interaction between the DSO, CPO, and electric Mobility Service Providers (eMSPs) can optimize EV charging schedules in alignment with RES availability, thereby contributing to grid stability and demand-response strategies. This was coupled with smart city data sources—such as weather conditions, noise levels, CCTV streams, public transport schedules, and crowd-sourced

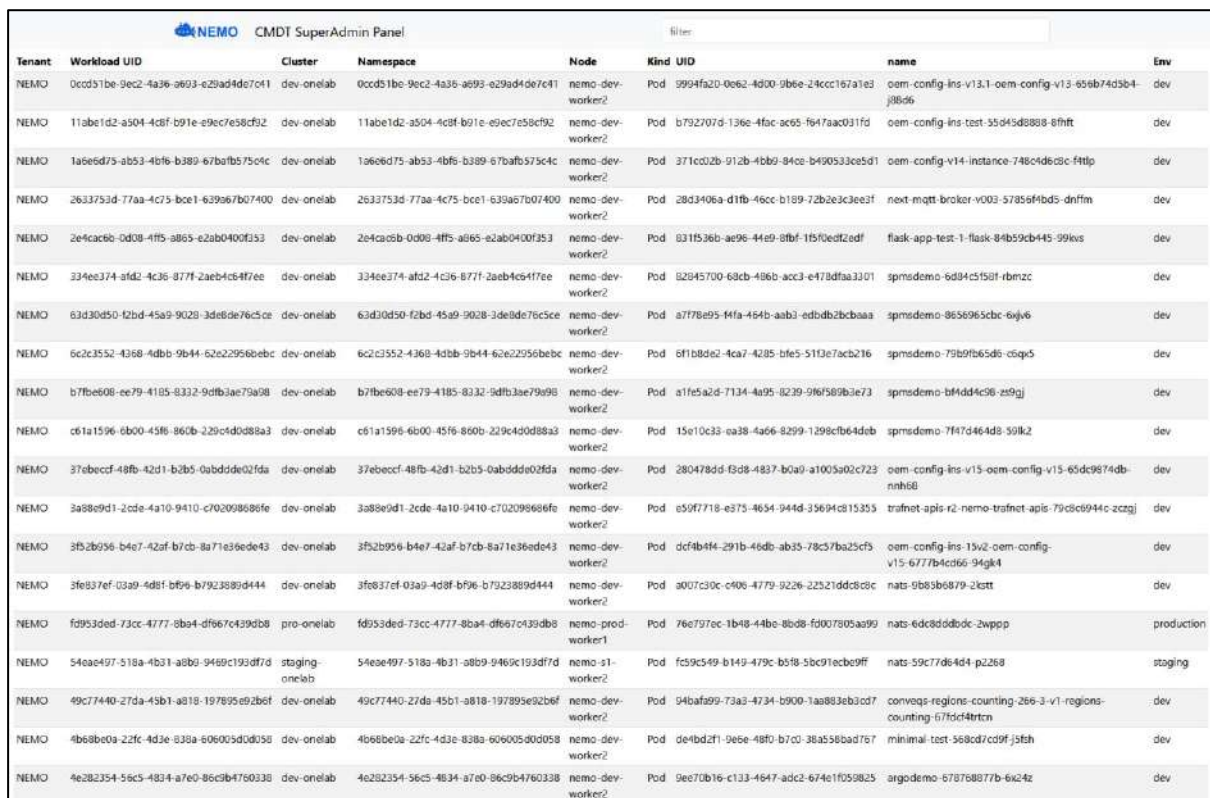
Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	52 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

mobility inputs—to enable predictive models for traffic flow and parking space availability, ultimately promoting citizen-centric eco-mobility services.

Within this scenario, the technical architecture focused on evaluating the end-to-end performance of several NEMO platform components. These included the AIoT architecture for distributed data ingestion, the Cybersecure Microservices Digital Twin (CMDT) for system modelling, and the use of MLOps pipelines leveraging Cybersecure Federated Deep Reinforcement Learning (CF-DRL) for model training and deployment. Additional components tested included the Federated meta Network Cluster Controller (mNCC), Secure Execution Environment (SEE), and the PRESS framework for safety and policy enforcement. Cybersecurity and digital identity attestation mechanisms were incorporated to validate secure data flows, while the meta-orchestrator and intent-based migration SDKs were responsible for dynamic workload placement across edge and cloud layers.

The scenario addressed a comprehensive set of functional and non-functional requirements (SE_02_FR01 to FR13 and SE_02_NFR01 to NFR07) and was aligned with key performance indicators (KPI_SE_02_1 to KPI_SE_02_4) defined for the SE_02 use case. Prerequisites included the full deployment of the NEMO platform, continuous collection of consumption and production data, and the presence of a functional FR marketplace interface between DSO and CPO systems. This scenario served as the foundation for testing for assessing the integration and orchestration capabilities of the NEMO architecture in a real-world smart city mobility context.

Moreover, Figure 47-49 provide operational evidence from the CMDT SuperAdmin Panel that the COMS NEMO cluster is running and that energy-related services are deployed. They show the overall cluster status, the OTA (Firmware over the air -service) namespace, the energy-marketplace stack (API, worker, blockchain, cache), and a cluster-wide “energy” filter that includes platform controllers and NEMO services across nodes and environments.



Tenant	Workload UID	Cluster	Namespace	Node	Kind	UID	name	Env
NEMO	0cc51be-9ec2-4a36-a603-e29ad4de7c41	dev-onelab	0cc51be-9ec2-4a36-a603-e29ad4de7c41	nemo-dev-worker2	Pod	9994fa20-0e62-4d00-9b6e-24ccc167a1e3	oem-config-ins-v13.1-oem-config-v13-65eb74d3b4-j86d6	dev
NEMO	11abe1d2-a504-4c8f-b91e-e9ec7e58cf92	dev-onelab	11abe1d2-a504-4c8f-b91e-e9ec7e58cf92	nemo-dev-worker2	Pod	b732707d-136e-4f4c-ac65-f94aac031fd	oem-config-ins-test-35d45d8888-8fhft	dev
NEMO	1a6e6d73-ab53-4bf6-b389-67baf575c4c	dev-onelab	1a6e6d73-ab53-4bf6-b389-67baf575c4c	nemo-dev-worker2	Pod	371cc02b-912b-4bb9-84ce-b490533ce5d1	oem-config-v14-instance-748c4de8dc-fhttp	dev
NEMO	2633753d-77aa-4c75-bce1-639a67b07400	dev-onelab	2633753d-77aa-4c75-bce1-639a67b07400	nemo-dev-worker2	Pod	28d3406a-d1fb-46cc-b189-72b2e3c3ee3f	next-mqtt-broker-v003-3785e6fbd5-drifm	dev
NEMO	2e4cac6b-0d08-4ff5-a865-e2ab0400f353	dev-onelab	2e4cac6b-0d08-4ff5-a865-e2ab0400f353	nemo-dev-worker2	Pod	831f536b-a996-44a9-8fbf-1f5f0edf2edf	flask-app-test-1-flask-84b59cb445-99kks	dev
NEMO	334ee374-a1d2-4c36-b77f-2aeb4c64f7ee	dev-onelab	334ee374-a1d2-4c36-b77f-2aeb4c64f7ee	nemo-dev-worker2	Pod	82945700-68cb-486b-acc3-e478dffa3301	spmsdemo-6d94c5f50f-rbmzc	dev
NEMO	63d30d50-f2bd-45a9-9028-3de8de76c5ce	dev-onelab	63d30d50-f2bd-45a9-9028-3de8de76c5ce	nemo-dev-worker2	Pod	a7778e95-f4fa-464b-aab3-edbdb2bcbaaa	spmsdemo-8e56965cb6-6xjy6	dev
NEMO	6c2c3552-4368-4dbb-9b44-62e22956ebc	dev-onelab	6c2c3552-4368-4dbb-9b44-62e22956ebc	nemo-dev-worker2	Pod	6f1b8d62-4ca7-4285-bfe5-5113e7acb216	spmsdemo-75b9f665d6-csqk5	dev
NEMO	b7f6e608-ee79-4185-8332-9dfb3ae79a98	dev-onelab	b7f6e608-ee79-4185-8332-9dfb3ae79a98	nemo-dev-worker2	Pod	a1f5e52d-7134-4a95-8239-9f6f589b3e73	spmsdemo-bf4dd4c9b6-z5qgj	dev
NEMO	c61a1596-6b00-45f6-860b-229c4d088a3	dev-onelab	c61a1596-6b00-45f6-860b-229c4d088a3	nemo-dev-worker2	Pod	15e10c33-ea38-4a66-8299-1298cfb4deb	spmsdemo-7f47d464d8-59lk2	dev
NEMO	37ebecf4-48fb-42d1-b2b5-0abddde02fda	dev-onelab	37ebecf4-48fb-42d1-b2b5-0abddde02fda	nemo-dev-worker2	Pod	280478dd-f3d8-4837-b0a0-a1005a02c723	oem-config-ins-v15-oem-config-v15-65dc9874db-nnh68	dev
NEMO	3a88e9d1-2cde-4a10-9410-c702098686fe	dev-onelab	3a88e9d1-2cde-4a10-9410-c702098686fe	nemo-dev-worker2	Pod	e59f7718-e875-4634-944d-35694c815353	trafnet-apls-r2-nemo-trafnet-apls-79c8c6944c-zctgj	dev
NEMO	3f52b956-b4e7-42af-b7cb-8a71e36ede43	dev-onelab	3f52b956-b4e7-42af-b7cb-8a71e36ede43	nemo-dev-worker2	Pod	dcfb444-291b-46db-ab35-78c57ba25cf5	oem-config-ins-13v2-oem-config-v15-6777b4cd66-94gk4	dev
NEMO	3fe837ef-03a9-4d8f-bf96-b7923889d444	dev-onelab	3fe837ef-03a9-4d8f-bf96-b7923889d444	nemo-dev-worker2	Pod	a007c30c-c406-4779-9226-22521ddc8c8c	nats-9b83b6879-2kott	dev
NEMO	fd953ded-73cc-4777-8ba4-df6e7c439db8	pro-onelab	fd953ded-73cc-4777-8ba4-df6e7c439db8	nemo-prod-worker1	Pod	76e7979c-1b48-44be-8bd8-fd007805aa99	nats-6dc8dddbdc-zwppp	production
NEMO	54eae497-518a-4b31-a8b9-9469c193df7d	staging-onelab	54eae497-518a-4b31-a8b9-9469c193df7d	nemo-s1-worker2	Pod	fc59c549-b149-479c-b518-5bc91ecbc9ff	nats-59c77d64d4-p2268	staging
NEMO	49c77440-27da-45b1-a818-197895e92b6f	dev-onelab	49c77440-27da-45b1-a818-197895e92b6f	nemo-dev-worker2	Pod	94baf99-73a3-4734-b900-1aa833eb3cd7	convex-regions-counting-266-3-v1-regions-counting-67f6cf4brcn	dev
NEMO	4b68be0a-22fc-4d3e-838a-606005d0d058	dev-onelab	4b68be0a-22fc-4d3e-838a-606005d0d058	nemo-dev-worker2	Pod	de4bd2f1-9e6e-48f0-b7c0-38a558bad767	minimal-test-568cd7cd9f-j5th	dev
NEMO	4e282354-56c5-4834-a7e0-96c9b4760338	dev-onelab	4e282354-56c5-4834-a7e0-96c9b4760338	nemo-dev-worker2	Pod	9ee70b1b-c133-4647-adc2-674e1f053825	argodemo-678768877b-6x24z	dev

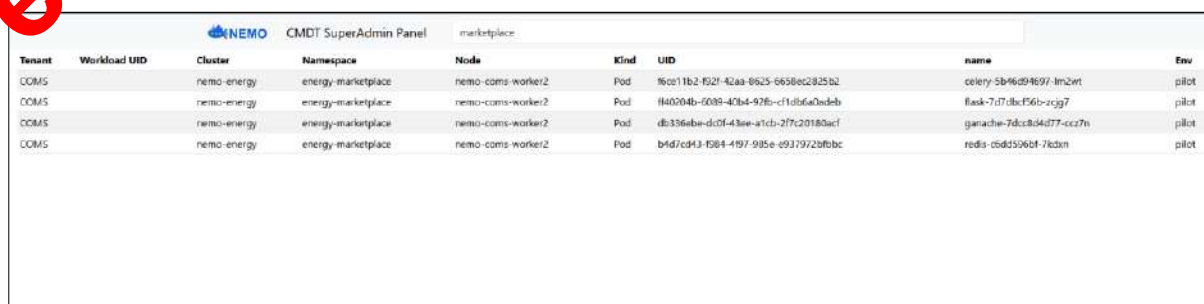
Figure 47 The CMDT SuperAdmin panel showing all running containers (~300) in NEMO cluster federation.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	53 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status:
			Final



Tenant	Workload UID	Cluster	Namespace	Node	Kind	UID	name	Env
COMS		nemo-energy	fota-service	nemo-coms-worker2	Pod	8b815731-2201-4346-9686-69ba06803b77	fota-586a7718b-juc59	pilot

Figure 48 The CMTD SuperAdmin panel shows running FOTA service revealing details such as cluster where is deployed, cluster node, container name, and tenant.



Tenant	Workload UID	Cluster	Namespace	Node	Kind	UID	name	Env
COMS		nemo-energy	energy-marketplace	nemo-coms-worker2	Pod	76cd11b2-f52f-42aa-8625-6658ec2825b2	celery-5b46d94697-lm2art	pilot
COMS		nemo-energy	energy-marketplace	nemo-coms-worker2	Pod	f40204b-0093-40b4-926b-cf1db6a0adeb	flask-7d7dbcf56b-vcjg7	pilot
COMS		nemo-energy	energy-marketplace	nemo-coms-worker2	Pod	db336abe-dc0f-43ee-a1cb-2f7c20180ac1	ganache-7dccc84d77-ccc7n	pilot
COMS		nemo-energy	energy-marketplace	nemo-coms-worker2	Pod	b4d7cd43-f584-4197-915e-6937972bfbbc	redis-c5dd596bf-7kdm	pilot

Figure 49 The CMTD SuperAdmin panel filters services of energy-marketplace, part of Smart Energy pilot.

3.2.3 Final validation results and KPIs Evaluation

KPI ID	Name	Description	Measurement/ Assesment method	Target
KPI_SE_02_1	Time granularity	Time granularity for monitoring	Measure the data gathering sampling rate	< 1 s
KPI_SE_02_2	Number of measurements	Interaction capability	Measure the field data amount collected	> 100.000 measurements/minute
KPI_SE_02_3	Number of sources	Measure the money saved involving EV users in DR campaigns	Measure the field data amount collected	money saved: 0.05 €/kWh
KPI_SE_02_4	RES utilization in EV charging	Measure the percentage of renewable energy used to charge the EVs involved in DR campaigns	Measure the percentage of renewable energy used to charge the EVs involved in DR campaigns	> 50%

KPI ID	Name	Description	Measurement/ Assessment Method	Target
KPI_SE_02_5	Increase urban Electrical Vehicles charging efficiency	Increase urban Electrical Vehicles charging efficiency	Evaluation of measured data leveraging data of the network	20 % in comparison with daily average value

Table 3 List of KPIs in SE_02

[KPI_SE_02_1](#)

Time granularity for monitoring:

As detailed in KPI_SE_01_1, the monitoring infrastructure includes two PMUs operating at 50 Hz (20 ms) and six PQAs with 1-second granularity, enabling high-resolution, time-synchronized monitoring of the distribution network.

In the SE_02 Use Case, this setup is further extended to include EVs and charging stations, which provide data with a granularity of 1 second.

[KPI_SE_02_2](#)

Interaction capability:

In addition to the 168,000 measurements per minute achieved in SE_01, the SE_02 use case expands the monitoring infrastructure by integrating 10 EVs and 3 charging stations. These additional components are configured to measure 27 electrical parameters in total, contributing further to the system's data exchange capacity. As a result, the overall interaction capability of the SE_02 reaches 21,600 measurements per minute, reflecting enhanced observability across both grid and e-mobility sectors.

[KPI_SE_02_3](#)

Measure the money saved involving EV users in DR campaigns:

Demand response (DR) campaigns are performed by DSO in order to balance the energy in a grid with high penetration of renewable energy sources. In particular, reverse power flow due to renewable energy surplus causes stability and safety problems, like voltage rise, frequency imbalance and fault equipment tripping; for this reason, DSO money saved involving EV user in DR campaigns is equal to 0.06 €.

[KPI_SE_02_4](#)

Measure the percentage of renewable energy used to charge the EVs involved in DR campaigns:

The KPI_SE_02_4 on RES utilization in EV charging measures the share of renewable generation used to supply EV charging demand. For this analysis, six months of data in 2024 were considered, with EV absorbed energy measured at the 22 kW charging stations and PV production from the 60 kW plant. The calculation is based on the ratio between EV charging demand supplied by PV and the total EV charging demand, expressed as a percentage. During the observed period, EV charging demand amounted to 4.11 MWh, while PV production reached 53.50 MWh. In 97.4% of days, PV generation exceeded EV charging demand, ensuring that charging was fully covered by renewable energy, while only in 2.6% of days demand surpassed PV production and grid electricity was required. Overall, RES utilization for EV charging exceeded 97%, significantly higher than the 50% threshold, confirming that local renewable generation largely supported the charging process and validating the KPI target.

[KPI_SE_02_5](#)

Increase urban Electrical Vehicles charging efficiency:

In parallel, charging efficiency was assessed by comparing the measured absorbed energy with the daily average baseline, reflecting the alignment of charging demand with renewable availability and reduced

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	56 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

reliance on external supply. The analysis shows an efficiency improvement exceeding 20% compared to the baseline, thereby achieving the KPI target.

3.2.4 Lessons Learned and Replication guidelines

As part of the SE_02 pilot, the researchers conducted a structured analysis of smart parking data to gain a deeper understanding of user behavior and system dynamics. The data, collected from sensors and cameras, revealed clear patterns aligned with office-hour usage, short-to-medium stay durations, and limited activity during evenings and weekends. These findings confirmed the importance of context-aware forecasting and system design tailored to real-world usage patterns. The experience highlighted the value of high-resolution parking data in optimizing resource allocation, guiding future EV charging integration, and informing smarter urban mobility planning.

In terms of replication, the modular design of the trial allows for straightforward adoption in other urban environments. Parking sensors, NTR cameras, and EV chargers can be integrated into local clusters, with data processed and visualized through standardized interfaces such as MQTT brokers and dashboards. By leveraging NEMO's automation features, including containerization and CI/CD pipelines, municipalities and operators can deploy similar solutions with minimal adaptation. This ensures not only technical scalability but also transferability of the behavioral insights, enabling other cities to optimize mobility resources, improve charging efficiency, and advance citizen-centric e-mobility services.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	57 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

4 Smart Manufacturing & Industry 4.0 Trial

The Smart Manufacturing & Industry 4.0 Pilot trail at Continental plant Ingolstadt considers two applications to be implemented, relevant to validating NEMO.

The SM_01 “Fully automated indoor logistics/supply chain” targets ADAS manufacturing. Currently, handling and transport of material (SMD-Components) from the Auto Store to the production sites are performed manually every 30 minutes. By utilizing a 3D-Vision-Camera for Bin Picking Application, an integrated Barcode Scanner and collaboration between different robot systems, Continental aims to fully automate controlled material picking from Auto Store and autonomous transfer to the production line.

The SM_02 “Human-centred indoor factory environment safety” will provide a high precision AGV localization layer merging real time localizations info obtained from cognitive sensors (safety cameras, radar and Lidar). A high speed and ultra-low latency (TSN) private wireless network will support massive data upload to the edge cloud facilities, where AI functions will detect the position of each body and build a “safety cell” around it to ensure human-centered safety, while federated CF-DRL will enable model transfer learning to the AGVs to enable autonomous avoidance of potential collision between AGVs, or between a worker and an AGV.

From a technological point of view, the Smart Manufacturing & Industry 4.0 Pilot aims to implement and validate the innovative architectural solution proposed by NEMO in order to fully automate the production workflows. The solution developed within the Pilot will achieve two major objectives at the level of process automation, namely:

- monitoring and detection on the manufacturing workflow
- and ensuring the manufacturing safety procedures
-

The NEMO platform will provide the functionalities of two monitoring systems through equipment and sensors, namely Bin Picking system for the automation of the manufacturing process and sensor system for assessing the working environment.

At the level of the communications infrastructure, an IoT/5G Time Sensitive Networking network will be implemented, and the orchestration of services will be done through intelligent Open API algorithms. Within the trial, emphasis will be put on implementing next generation of IoT applications related to AGVs-AGVs and AGVs-human operators collision prediction, detection and avoidance through real-time positioning and federated ML hosted locally and in the 5G/WiFi edge.

Smart Manufacturing and Industry 4.0 Pilot trail has great potential for real exploitation and in strengthening European business development especially through the advantages of metaOS applications. The trail enables flexible use of available resources, either near/far edge or cloud. Furthermore, advanced network management for ensuring strict latency/bandwidth requirements are covered. This enables specific measures to be implemented in the trail, resulting in significant process optimization and performance improvement through the complete automation of the material flow from the warehouse to production. Non-value-added processes are thus reduced. The error-free just-in-time provision of materials enables increased planning security and optimized warehousing can be realized. The extensively increased efficiency in the logistics processes results in cost reduction. The high cost pressure in best-cost countries in Europe can thus be counteracted and a contribution can be made to site security of European production locations.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	58 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Furthermore, environmental sustainability, considering energy consumption & efficiency and green energy usage in service allocation and delivery can be achieved. The quality increase in material provision enables reduced resource consumption on the one hand and reduces production waste on the other. The sustainability of production and the entire supply chain is optimized, thereby achieving a higher Green Plant certification, and thus improving the image of European production sites. In addition, the risk of accidents at work is reduced and the ergonomics of work processes are improved through higher automation.

Cross-domain collaboration through data sharing, brokering and monetization is another innovative added value for the European locations. The increased automation and standardization of processes means that these can also be transferred to other locations. In addition, a simple and secure exchange of data between different locations in Europe is possible, resulting in new, promising application scenarios.

Continental develops and produces driver assistance systems, which are an important building block for automated, safe and comfortable traffic. It is crucial for Europe to participate in this promising core business and to further expand its good market position in Europe. Continental is one of the market leaders with headquarters in Europe. Robotics in general and particularly Cobots have a strong impact within highly automated production facilities. As market demands transformation and demographic change induce new challenges on mass production, Continental has already undergone feasibility and profitability assessment for the application of flexible Cobots in upcoming assembly facilities. In parallel, Continental investigated Cobots and AGVs with respect to the strict requirements of electronic production and has gained deep insight into industrial needs and problem settings. The Continental location in Ingolstadt is the ADAS lead plant and is therefore responsible for testing, establishing and disseminating promising technologies at other Continental locations. Innovative manufacturing processes are set up in Ingolstadt for the first time and upon successful verification are transferred to the other ADAS sites. In this way, the NEMO project results, especially the technology solutions developed in the Smart Manufacturing & Industry 4.0 pilot, can be profitably exploited and disseminated in Europe.

NEMO is thus significantly increasing industry in Europe by building up technological know-how, optimizing production performance and increasing sustainability of European development and production locations.

4.1 Set up and Integration

Since D5.3[4], the physical infrastructure of the SM-01 pilot has been fully deployed and finalized.

As part of the Industry 4.0 pilot conducted at the Continental Living Lab, the scTracking application (short for *Supply Chain Tracking*) has been developed and deployed to enable end-to-end monitoring and intelligent coordination of supply chain activities within a smart factory environment. The application integrates machine learning, robotics control, and log analytics to guide collaborative robots (cobots) in handling parts, while also tracking and visualizing each step of the logistics workflow.

This application is fully deployed within the NEMO meta-OS infrastructure, leveraging its orchestration, container management, and data handling capabilities to support real-time operation, secure access, and scalability across edge and cloud components.

The main functionalities of the scTracking application are:

1. Intelligent Object Picking through ML-Based Visual Analysis

The first core functionality of ScTracking enables cobot-assisted part manipulation by training a machine learning model on images captured from cameras installed in the STAR Living Lab. The process consists of:

- Continuous image acquisition from the shop floor environment.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	59 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status:
			Final

- Training of an object segmentation or detection model to identify optimal grip points.
- Deployment of the model for inference, guiding the cobot to safely and precisely pick up parts without damaging them.

This ML pipeline is orchestrated by the NEMO platform, which dynamically allocates training and inference workloads across the IoT-edge-cloud continuum, optimizing resource use while maintaining low-latency performance.

2. Supply Chain Process Monitoring and Log-Based Analytics

The second major functionality involves tracking all key steps in the supply chain process, from part withdrawal to AGV loading by robots. This is achieved through:

- Parsing of log files from the STAR Living Lab environment.
- Extraction of time-stamped events corresponding to each of the five major steps:
 - Image Capture
 - Object Detection
 - Object Counting
 - Grip Point Identification
 - Cobot Gripping
- Storage of parsed events in a PostgreSQL database.
- Aggregated visual and tabular representation of parts processed, including filters for part codes, step types, and time intervals.

This enables transparent monitoring, traceability, and performance analysis across the entire logistics workflow.

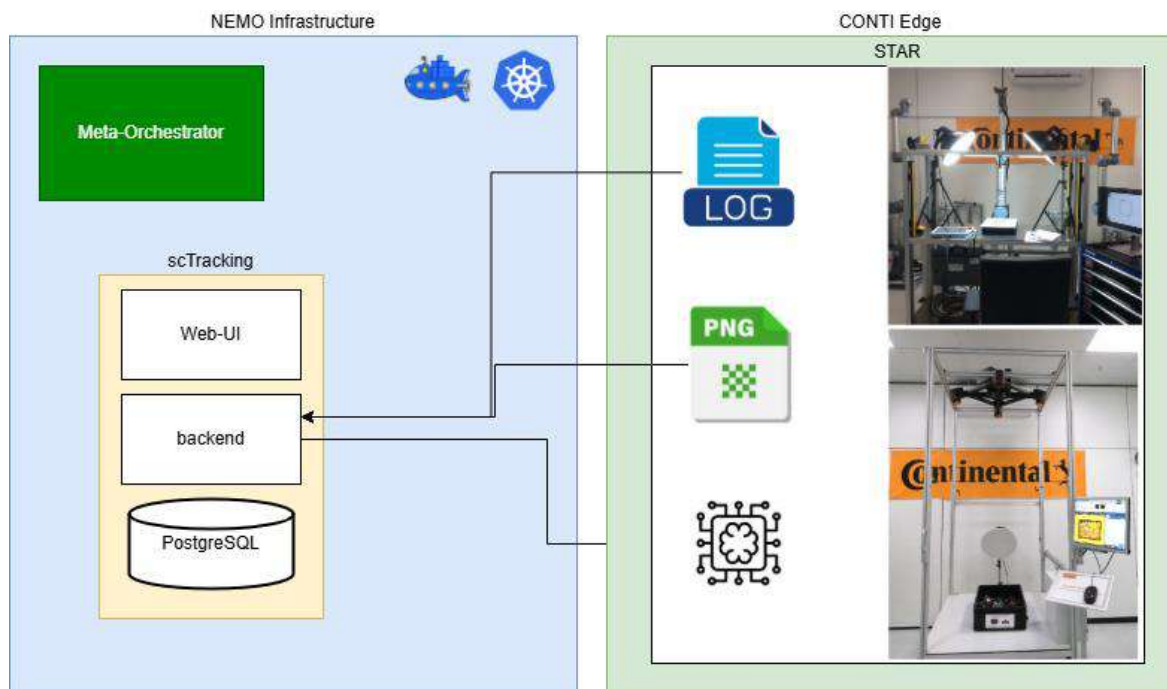


Figure 51 scTracking architecture and deployment in the NEMO environment

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	60 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

The scTracking application is deployed in the Kubernetes environment managed by NEMO, using a modular, containerized architecture, as depicted in Figure 51. It comprises the following components:

- **Web User Interface (UI):** a responsive frontend providing access to all functionalities. Users can view historical logs, upload images, configure system parameters, and analyze visual trends.
- **Backend Service:** a microservice that:
 - Parses and stores logs from the Cloud environment,
 - Manages image acquisition and uploads,
 - Interfaces with the PostgreSQL database and MinIO object storage,
 - Serves data to the frontend via REST APIs.
- **PostgreSQL Database:** stores structured supply chain event data including timestamps, part codes, process steps, and error statuses.
- **MinIO Object Storage:** used for storing and retrieving raw and processed images for training, inference, and visualization. It integrates directly with ML workflows.

This architecture ensures full alignment with NEMO's distributed resource model, allowing the backend and ML components to be executed across edge or cloud nodes depending on system load and latency needs.

scTracking implements secure, role-based access control through full integration with the Keycloak identity management platform provided by NEMO.

- Upon startup, the application presents a login screen where users authenticate using their NEMO credentials Figure 52.
- Access is granted only if the user has the appropriate role (e.g., operator, admin).
- A token-based session is initiated for interacting with backend APIs.

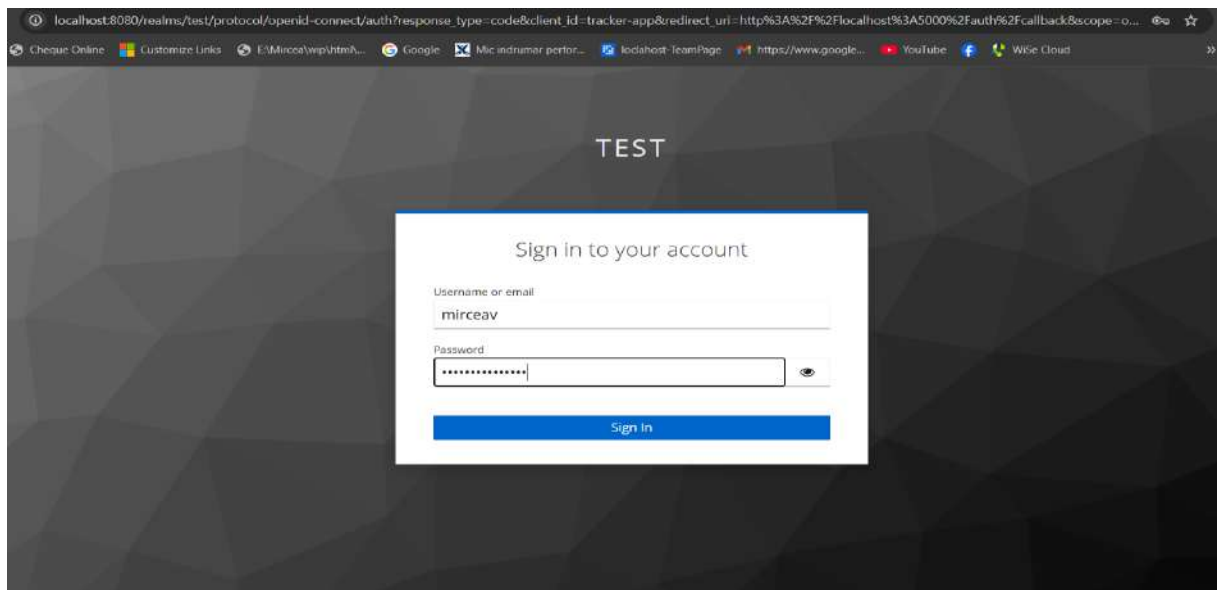


Figure 52 scTracking login screen

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	61 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Upon successful login, users are redirected to a landing page Figure 53 that:

- Presents an overview of the supply chain tracking process.
- Highlights the five key process steps being monitored.
- Provides links to detailed views for data exploration and control.

The scTracking main menu includes:

- **Tracking History:** displays all logged events with timestamp, part code, step code, and number of parts, allowing filtering by part code, step code, time range, and sorting by timestamp or processing step.
- **Last Activity:** identical in structure to Tracking History but limited to the last 24 hours, designed for real-time operational monitoring.

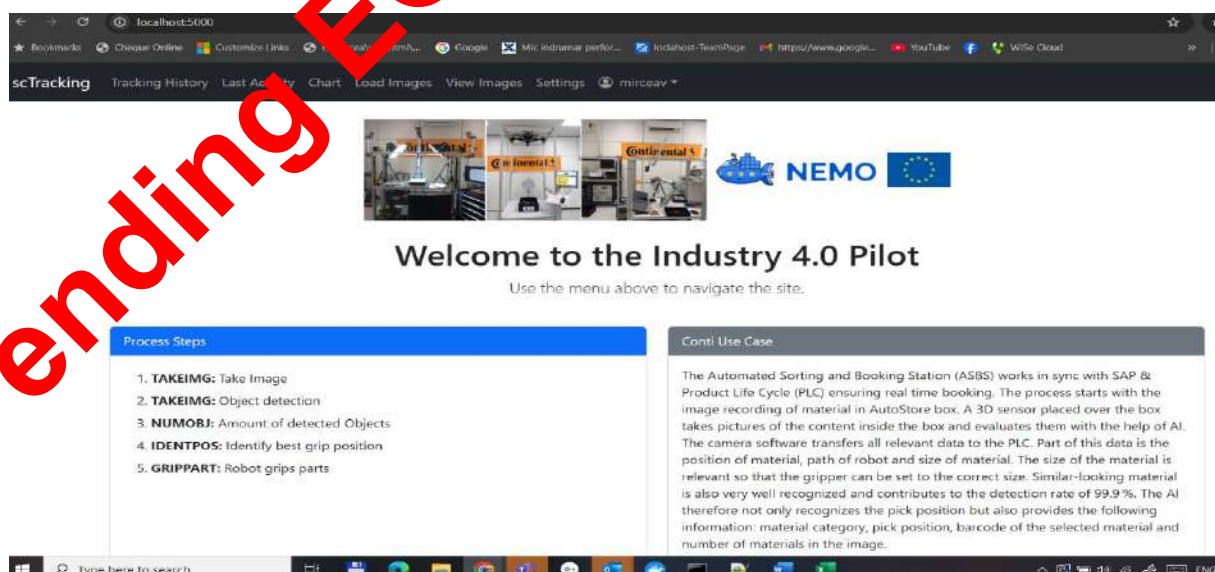


Figure 53 scTracking landing page

- **Chart:** displays a bar chart illustrating the number of parts processed per step, allowing time-based filtering for daily/weekly analysis
- **Load Images:** enables users to upload new images to enrich the dataset or test ML inference behavior.
- **View Images:** allows browsing and inspection of stored images, including inference overlays.
- **Settings:** provides configuration options for log sources, cleaning database, Keycloak, MinIO, PostgreSQL database connections parameters

4.2 Test scenarios verification

The SM01_Test_Scenario_1 (SM01_TS01), described in the deliverable D5.3[4], validates the ScTracking application's ability to automatically trace each step of the supply chain process and replace previously manual operations of part movement tracking with real-time digital logging and visualization.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	62 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

To ensure full transparency and traceability across the supply chain process, the scTracking application deployed implements a robust log-based activity tracking mechanism. This mechanism captures all key actions performed in the supply chain pipeline — from part retrieval to robotic handover — in a structured and timestamped format.

This scenario directly confirms the effectiveness of the log-to-database pipeline implemented by ScTracking, as described in the current subsection. The showcases:

- Real-time operation of part tracking
- Fully automated data flow from cobot logs to UI dashboards
- Elimination of manual effort, enabling smarter and more reliable traceability.

The logs are continuously generated during live operations at the Continental STAR Living Lab and reflect real-time events from the integrated cobot environment, vision systems, and logistics control modules. In the Figure 54 are displayed a snapshot of a logfile. These logs serve as the primary source of truth for process monitoring, event reconstruction, and statistical analysis.

```
[2023-06-19 12:57:22.856] [LumiScan VGR] [debug] clicked 'done' pushButton
[2023-06-19 12:57:22.856] [LumiScan VGR] [info] calling 'computeInstances3D'
[2023-06-19 12:57:22.856] [LumiScan VGR] [debug] Active camera id : 'camera'
[2023-06-19 12:57:22.856] [LumiScan VGR] [debug] Active pipeline id: 'pipeline'
[2023-06-19 12:57:22.856] [LumiScan VGR] [debug] Active robot id : 'robot_7'
[2023-06-19 12:57:22.856] [LumiScan VGR] [info] calling 'validateRequirements'
[2023-06-19 12:57:22.856] [LumiScan VGR] [debug] New detection data available
[2023-06-19 12:57:22.856] [LumiScan VGR] [debug] Processing object 0
[2023-06-19 12:57:22.868] [LumiScan VGR] [debug] Obtained 34972 inner nearest neighbours, roi percentage: 0.75
[2023-06-19 12:57:22.868] [LumiScan VGR] [debug] Obtained 63426 outer nearest neighbours
[2023-06-19 12:57:22.868] [LumiScan VGR] [debug] Ray direction is (0.07401149 0.058190545 0.99555784)
[2023-06-19 12:57:23.000] [LumiScan VGR] [debug] RANSAC Least squares plane parameters: -0.050471*x -0.095209*y -0.993679*z +1.410300 = 0
[2023-06-19 12:57:23.000] [LumiScan VGR] [debug] Least squares plane parameters: +0.029852*x +0.076003*y +0.996661*z -1.408912 = 0
[2023-06-19 12:57:23.000] [LumiScan VGR] [debug] Least squares plane curvature : 0.0
[2023-06-19 12:57:24.069] [LumiScan VGR] [info] computeFirstLayerDistanceKde took 1068.6494ms
[2023-06-19 12:57:24.069] [LumiScan VGR] [warning] fallback: 0.0
[2023-06-19 12:57:24.070] [LumiScan VGR] [debug] Applying shift correction (0.010922186)
[2023-06-19 12:57:24.070] [LumiScan VGR] [debug] Processed object 0 in 1213.2739 ms
[2023-06-19 12:57:24.070] [LumiScan VGR] [debug] Processing object 1
[2023-06-19 12:57:24.080] [LumiScan VGR] [debug] Obtained 31096 inner nearest neighbours, roi percentage: 0.75
[2023-06-19 12:57:24.080] [LumiScan VGR] [debug] Obtained 55282 outer nearest neighbours
[2023-06-19 12:57:24.080] [LumiScan VGR] [debug] Ray direction is (0.08345204 -0.0500139 0.99482167)
[2023-06-19 12:57:24.197] [LumiScan VGR] [debug] RANSAC Least squares plane parameters: -0.035035*x +0.018608*y -0.999213*z +1.459299 = 0
[2023-06-19 12:57:24.198] [LumiScan VGR] [debug] Least squares plane parameters: +0.009737*x -0.000417*y -0.999952*z -1.455800 = 0
[2023-06-19 12:57:24.198] [LumiScan VGR] [debug] Least squares plane curvature : 0.008431132
[2023-06-19 12:57:25.394] [LumiScan VGR] [info] computeFirstLayerDistanceKde took 1195.8396ms
[2023-06-19 12:57:25.394] [LumiScan VGR] [warning] fallback: 0.0
[2023-06-19 12:57:25.394] [LumiScan VGR] [debug] Applying shift correction (0.0072278716)
[2023-06-19 12:57:25.394] [LumiScan VGR] [debug] Processed object 1 in 1324.5542 ms
[2023-06-19 12:57:25.394] [LumiScan VGR] [info] Returned with 'SUCCESS'
[2023-06-19 12:57:25.397] [LumiScan VGR] [info] Procedure took 2541.1848 ms
[2023-06-19 12:57:25.397] [LumiScan VGR] [info] calling 'computeDepth'
[2023-06-19 12:57:25.397] [LumiScan VGR] [debug] Active camera id : 'camera'
[2023-06-19 12:57:25.397] [LumiScan VGR] [debug] Active pipeline id: 'pipeline'
[2023-06-19 12:57:25.397] [LumiScan VGR] [debug] Active robot id : 'robot_7'
[2023-06-19 12:57:25.397] [LumiScan VGR] [info] calling 'validateRequirements'
[2023-06-19 12:57:25.398] [LumiScan VGR] [debug] New instances3D data available
[2023-06-19 12:57:25.398] [LumiScan VGR] [debug] Missing DATA_ROBOT_LAST_GRIP
[2023-06-19 12:57:25.398] [LumiScan VGR] [info] Returned with 'MISSING_DATA'
[2023-06-19 12:57:25.398] [LumiScan VGR] [info] Procedure took 0.2092 ms
```

Figure 54 Log file from Continental STAR

The ScTracking system on the NEMO cloud, integrated with the STAR Living Lab workflow, attends the SM01_TS01 scenario, with these achievements:

- Detect and log each physical step taken by parts (e.g., image capture, detection, gripping)
- Store event details (e.g., time, part code, success/failure)
- Display these events instantly in a human-readable format
- Replace manual logging or operator-based confirmation of part movement.

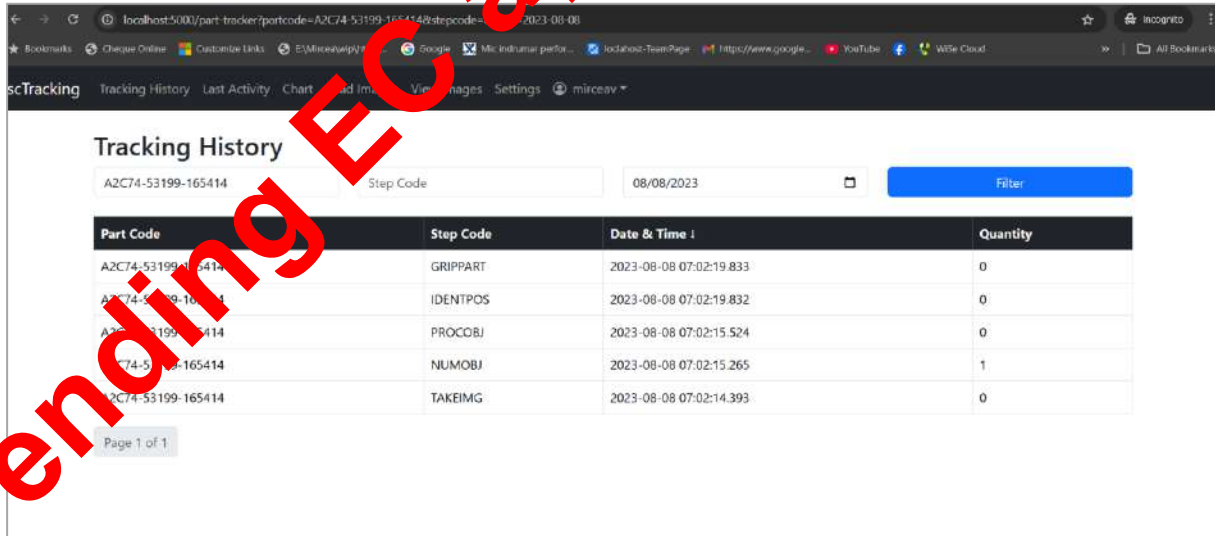
As the outcome of SM01_TS01 scenario, the traceability of each detected is available with a complete process trail, visualized through charts and tables.

By accessing the Tracking History or Last Activity, the user can see a comprehensive log of parsed events from the laboratory, listing all actions performed in the supply chain process, such as image

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	63 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status:
			Final

capture, detection, grip decisions, and cobot actions. These actions are presented in the history table having the following codes:

- TAKEIMG: the visual system takes images of the box containing parts
- PROCOBJ: the parts have been identified
- NUMOBJ: the number of parts has been established
- IDENTPOS: the correct position of gripping has been identified
- GRIPPART: the cobot handles the parts



Part Code	Step Code	Date & Time	Quantity
A2C74-53199-165414	GRIPPART	2023-08-08 07:02:19.833	0
A2C74-53199-165414	IDENTPOS	2023-08-08 07:02:19.832	0
A2C74-53199-165414	PROCOBJ	2023-08-08 07:02:15.524	0
A2C74-53199-165414	NUMOBJ	2023-08-08 07:02:15.265	1
A2C74-53199-165414	TAKEIMG	2023-08-08 07:02:14.393	0

Figure 55 Tracking History window

In the Figure 55 is displayed the screenshot of Tracking History window of scTracking application. Users can filter entries by part codes, step codes, time windows. The sorting operation is available by timestamp, part type, and processing status. A summary count is provided for the number of parts that passed the step NUMOBJ.

The Last Activity window is identical in structure to the Tracking History page, and this module focuses specifically on logs from the last 24 hours, enabling users to monitor recent activity without navigating large datasets.

The Chart module offers a visual summary of operational activity within the scTracking system. When the user selects the Chart menu item from the main interface, a dedicated page is opened displaying a bar chart Figure 56 that illustrates the number of parts processed across the different stages of the supply chain workflow.

Key features of the Chart view include:

- A bar chart representation showing part volume – number of objects processed established in the step NUMOBJ
- Optional date/time filters to focus on specific operational windows (e.g., today, last 7 days, custom range).
- Live data binding with the backend database to ensure up-to-date visualizations as logs are continuously ingested and parsed.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	64 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status:
			Final

Figure 56 Tracking History window

Integration of TSN into 5G requires new network functions called Network-side TSN Translator (NW-TT) and Device-side TSN Translator (DS-TT).

The NW-TT receives those gPTP messages and makes an ingress timestamping (TSi) for every gPTP event message and uses the cumulative rateRatio obtained within the payload of the gPTP message in order to calculate the link delay from the upstream TSN node (gPTP entity) expressed in TSN grandmaster (GM) time.

UPF then forwards the gPTP message from the TSN network to the UEs through all PDU sessions terminating in this UPF that the UEs have established to the TSN network. All gPTP messages are transmitted on a QoS Flow that complies with the residence time upper bound requirement specified in IEEE Std 802.1AS.

A UE receives the gPTP messages and forwards them to the DS-TT. The DS-TT located in the user equipment (UE) receives the gPTP messages and creates an egress timestamping (TSe) for the gPTP event messages for external TSN working domains connected to the UE. The residence time spent within the 5G system (5GS) for this gPTP message is calculated by the difference between TSi and TSe. The DS-TT converts the residence time spent within the 5GS in TSN GM time with the rateRatio provided by the gPTP message. It also adds the calculated residence time and removes the TSi from the payload suffix field of the gPTP message that is sent downstream to the TSN node connected to the UE.

The sum of the UE-DS-TT residence time and the PDB of the QoS Flow needs to be lower than the residence time upper bound requirement for a time-aware system specified in IEEE Std 802.1AS. In general, the DS-TT translates the TSN 802.1 protocols on top of 5G networks but cannot be assumed that would be a DS-TT capable UEs.

Simulation result and discussion

The simulation setup comprises a network of interconnected time aware devices. In the scenario the DS-TT is not integrated with the 5G modem and SIB9 cannot be provided to the DS-TT alternative measurement performed using an additional GM to be used for time reference in the NW-TT and the DS-TT, as shown in Figure1.

The measurement data has been collected and depicted in Figure2 as a time offset between the GM and TSN slave node after end to end TSN time synchronization. The result shows that the time difference (offset) between the GM and TSN slave node stays in order of 7-8 us.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	66 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Time Offset (us) vs. Index



Figure 58 Time Offset of TSN slave instance after TSN time synchronization.

There are some peak values observed on the result where the time offset jumps up to 15us, this is due to the UE rebooting itself and disconnecting itself from the core networks. When the UE reboots itself, it creates a discontinuity on the PTP message exchange between the GM and Slave node and the Slave node becomes out of sync from the GM clock. The UE reconnects to the core network within a fraction of seconds, which allows the TSN slave device to synchronize back with the GM before it drifts significantly. Overall, after having a successful synchronization with the GM the slave server has a mean time offset of around 8us.

An oscilloscope is also used in the setup to observe and analyse the same result discussed above. shows Figure 59 the PPS output from the TSN nodes: C1 displays the Slave clock 1PPS, C2 displays the Master clock 1PPS, and the statistics shows the time offset between the GM and the slave server clock as a delay. The result shows that the time difference between the slave device and the GM is in order of 8us and a standard deviation of 0.5us. For the time span where the data is collected, the standard deviation stays around 4us which shows that the synchronization mechanism is working steadily.



Figure 59 C1: slave-device PPS output, C2: GM clock 1PPS output, and time difference between GM and Slave clock.

The result of the simulation shows that a successful end-to-end TSN time synchronization over the 5G system can be achieved with a small time offset between TSN nodes.

4.3 Final validation results and KPIs Evaluation

KPIs	
KPI_SM_01_2: Different types of sensors' data to be analysed (> 10)	reached
KPI_SM_01_3: System reaction in emergency cases (< 0,5 sec)	under review / verification
KPI_SM_01_4: ADAS supply chain improvement: accuracy (> 30 %)	reached
KPI_SM_01_5: Cost and time reduction (> 20%)	reached
KPI_SM_02_1: Different types of AGV and Cobots to be addressed (> 4)	reached
KPI_SM_02_3: Improve human collision avoidance and manufacturing safety (30 %)	under review / verification
KPI_SM_02_4: Cost and time reduction (> 20%)	reached

Table 4 KPIs Evaluation

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	68 of 195
Reference:	D5.4	Dissemination:	PU
Version:	6.0	Status:	Final

4.4 Lessons Learned and Replication guidelines

The Smart Industry pilot highlighted the benefits of applying a Meta-OS approach in manufacturing environments. A key lesson learned is that integration across diverse systems—from robotics and vision-based cobots to safety sensors and supply chain monitoring tools—requires clear interfaces and strong interoperability support. The pilot also emphasized the importance of human-centered design, as safety and worker acceptance are critical factors in industrial adoption.

In terms of replication, the use cases explored in this pilot—automated logistics and human-machine safety monitoring—are relevant to a wide range of factories beyond the specific context addressed here. Replication would benefit from focusing on modular deployment, where specific NEMO-enabled services (such as tracking or edge-to-cloud orchestration) can be adapted to different industrial settings. Aligning deployment with existing digital transformation strategies, such as Industry 4.0 roadmaps, can further support adoption.

Document name:	NEMO Living Labs use cases evaluation results - Final version					Page:	69 of 195
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

5 Smart Media & XR Trial

Smart Media City

The Smart Media Trial aims to enhance the experience of spectating live sporting events by integrating AI-driven data analysis and XR capabilities to enrich media content. During the race, spectators and selected runners capture media content using smartphones, tablets, GoPro cameras, and, where available, IP cameras and drones along the running circuit. This incoming content undergoes automated processing, annotation, and rendering, with AI/ML models running partially on devices and partially at the edge. A curated selection of this content is then broadcasted in real time, such as through social media, based on the location of leading runners and notable race events, as identified through automated and user-provided annotations.

Urban city marathons and races have been used for decades as public events to enhance the image of a city, provide tourist attractions and exemplify the value of sports for healthy living and dedication. With thousands of local visitors and viewers, media channels strive to provide the best possible viewer media experience by dedicating more and more resources to cover it in detail and also foster visitor attendance and interest. Currently, local viewers of the race have limited ways to stay informed on the race status. They can either locate passing runners at their location or monitor radio or live-streaming TV shows. All choices leave them with little information on the race status they are interested in and without the possibility of involvement unless they use social media channels.

The implementation of this solution faces several challenges, including integrating large video feeds and IoT devices with existing media infrastructures and ensuring a user-friendly experience for outdoor events. The solution leverages a novel cloud-based meta-OS framework to provide and manage the resources needed for a) handling and analyzing the video feeds, b) for AI annotation of the video feeds, and c) for interfacing with a consumer smartphone app. Various risk factors are involved in implementing such a solution, such as the usage of high computing resources, network latency, high video latency, the synchronization of events with media content, the right access to content with privacy preservation and the quality of Experience (QoE) optimization.

Smart Media XR

The XR viewpoint enhances VR experiences using gestures recognition and biometric data. The experiences that will be enhanced are: a) A VR Head Mounted Display (HMD) experience regarding the visit of an ancient Greek Workshop and b) An interactive real-time VR Dome experience that is presented at the Tholos Dome VR Theatre of the Hellenic Cosmos. The trial consists of several micro-services that are going to be deployed in the continuum (central cloud, edge cloud, IoT devices) via the NEMO platform. The infrastructure in FHW premises (local VMs, servers, IoT/VR devices, etc) consists of the edge/IoT infrastructure and external infrastructures (Central Cloud, HPC servers) will be used for services that demand high computational resources.

Virtual Reality (VR) can cause discomfort for some users due to a variety of factors. This discomfort is often referred to as VR sickness or cybersickness, and it shares similarities with motion sickness. Furthermore, since the usage of a VR headset isolates the user, it is not easy for the administrator to understand if the user faces any problem.

5.1 Round of Athens Race

The Trial stadium description is well documented in D5.2[3], D5.3[4]. The trial took place outdoors. The use case was hosted by the Municipality stadium of Egaleo in Athens and deployed a scenario with the race happen inside a Olympic Size Stadium. This is done on purpose to stress the KPI specified and to test in a more “realistic” environment.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	70 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



Figure 60 Stadium Location and Race Circuit

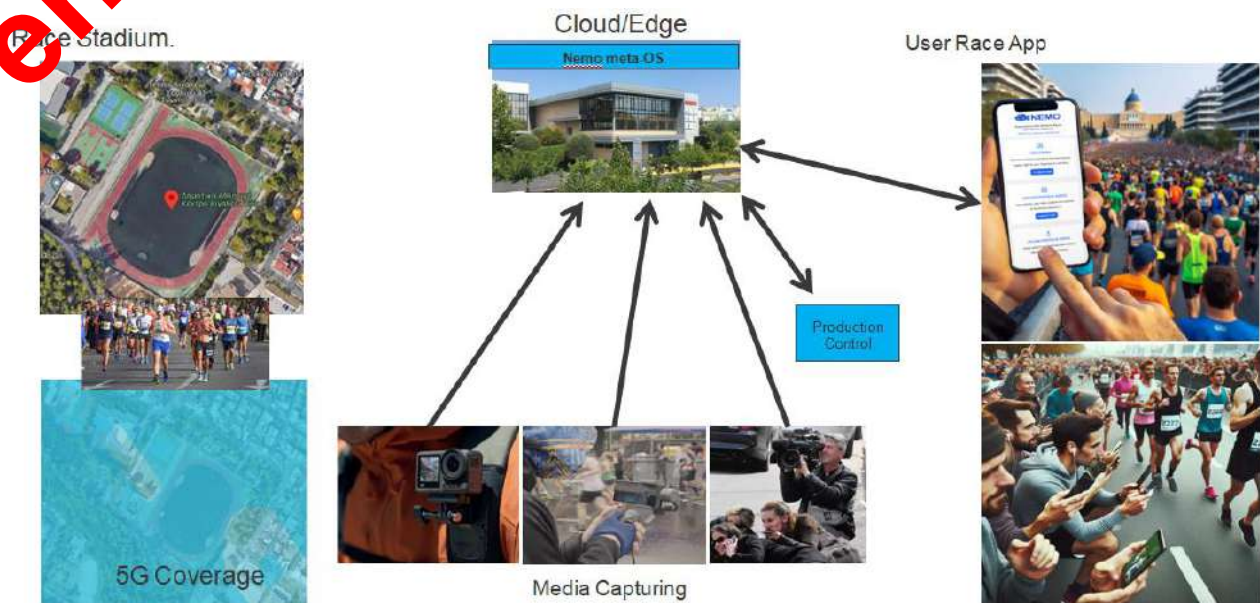


Figure 61 Architectural Topology Diagram of Trial

5.1.1 Set up and Integration

5.1.1.1 Processing the stream

At the day of the Pilot, 5G coverage was secured throughout the Stadium Site and the availability of public IP connection to the respective VM hosted at OTE. The OTE based cluster was registered to NEMO for services and the diagram below explains the connection flow. Spectators of the race around the stadium besides watching the race were also using the dedicated smartphone app for participation. The app connected to the Public IP (193.218.97.148:8021) and streamed all user media and metadata.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	71 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

The cluster internally translated the media received on the public IP to multiple internal IP addresses which were handled by VNFs of the Media Production Engine. The Media Production Engine running on the VM processed the media, using NEMO services for AI/ML annotation, Quality of Experience, Network Management, Decoding and Encoding and production Control. The final streams that were exported by the VM were sent to an RTMP server publicly accessible (at 193.218.97.148:8023). All the smart phone app participants received automatically the rtmp server streams and were able to watch the stream with the desired annotation. Such as for example all streams where a runner with a specific Bib number was visible. Besides the Bib number streams were annotated also with GPS and user comments. Although the rtmp streams were designed to be used with the smart phone app for maximum integration, any rtmp client was able to receive the streams missing ofcourse the metadata portion and ability to select stream data.

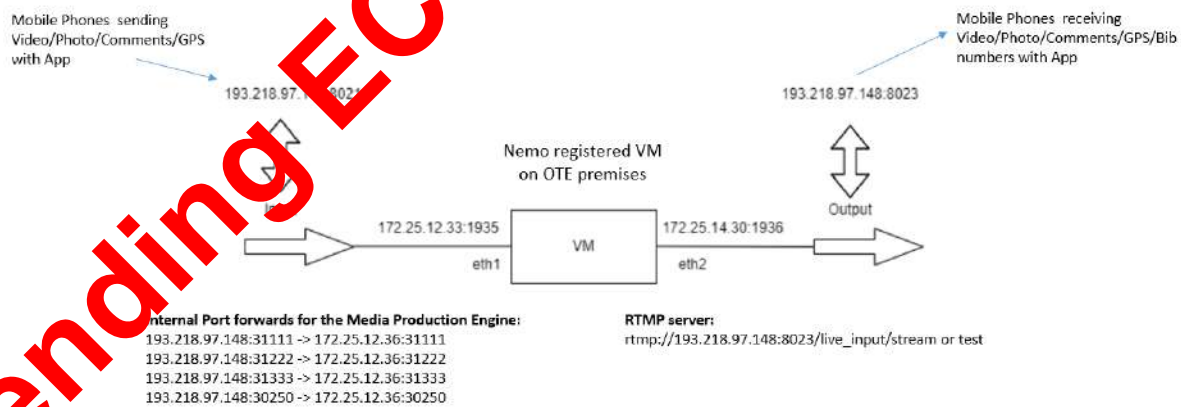


Figure 62 Communication Diagram of the Smart Media City Pilot

For the communication and the traffic delivery from the Egaleo stadium “Eva Cristodoulou” to the NEMO platform and Smart media Use case components hosted in OTE premises, OTE created a dedicated APN (Access Point Name) namely i-trialnemo with the following characteristics.

IKE Phase 1	
Key Exchange Encryption	AES-256
Data Integrity	SHA256
Diffie-Hellman Security Associations	Diffie-Hellman Group 14
Pseudo-Random Function (prf)	SHA256
Security Associations Lifetime	86400 seconds (1440 minutes)
Pre-Shared Secret Key	N/A
IKE Phase 2	
Encapsulation	ESP
Encryption Algorithm	AES-256
Data Integrity	SHA256
Perfect Forward Secrecy	Yes (Diffie-Hellman Group 14)
Security Associations Lifetime	3600 seconds (60 minutes)

Firstly, the service configuration was created defining the path from the network to the NEMO cluster, as following:

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	72 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Ipsec tunnel cosmote: 195.167.65.105.
 Ipsec tunnel i-trialnemo: 193.218.97.148
 i-trialnemo destinations server : host 172.25.12.0/24
 Static Mobile Range: "10.26.240.32/28"
 Encryption scheme: IKE , Authentication algorithm: SHA256, Encryption algorithm: AES256
 Authentication Method: Pre-Shared Secret : @nemo2025-07-01, Diffie-Hellman group: 14
 Renegotiate IKE (phase 1) Security Associations every 86400 sec
 Renegotiate IPsec (IKE phase 2) Security Associations every 3600 sec

Then an APN (Access Point Name) dedicated to the NEMO Smart Media pilot was created in two phases.

PHASE 1: APN Configuration and Verification

In this phase, The APN name (i-trialnemo) is selected and configured. The mobile network determines if a data connection is allowed and how it should be established.

IKEv2 SAs: Session-id: 23843, Status:UP-ACTIVE, IKE count:1, CHILD count:1 Tunnel-id Local Remote Status Role 2553806655 195.167.65.105/4500 193.218.97.148/4500 READY RESPONDER Encr: AES-CBC, keysize: 256, Hash: SHA256, DH grp:14, Auth sign: PSK, Auth verify: PSK Life/Active Time: 86400/209 sec Session-id: 23843 Status Description: Negotiation done Local spi: 2B70457EF4E13ACB Remote spi: FC34ED9D7EF97C5A Local id: 195.167.65.105 Remote id: internal-ca Local req mess id: 20 Remote req mess id: 2 Local next mess id: 20 Remote next mess id: 2 Local req queued: 20	Remote req queued: 2Local window: 1 Remote window: 1 DPD configured for 10 seconds, retry 2 NAT-T is detected outside Mobike is enabled IKEv2 Fragmentation Configured MTU: 576 bytes, Overhead: 28 bytes, Effective MTU: 548 bytes Parent SA Extended Status: Delete in progress: FALSE Marked for delete: FALSE Child sa: local selector 10.26.240.32/0 - 10.26.240.47/65535 remote selector 172.25.12.0/0 - 172.25.12.255/65535 ESP spi in/out: 0xbd0feb96/0xceb062d3 AH spi in/out: 0x0/0x0 CPI in/out: 0x0/0x0 Encr: AES-CBC, keysize: 256, esp_hmac: SHA256 ah_hmac: None, comp: IPCOMP_NONE, mode tunnel
--	---

Phase 2: Connection Establishment and Routing

Once the APN is verified and accepted, the data connection is established. This includes:

- Creating the PDP Context (or Bearer Session in LTE/5G): A virtual path is created for data transfer.
- Provisioning Connection Parameters: The network provides technical settings like DNS servers, gateway address, etc.
- Data Flow Activation: At this point, the connection is fully active, and data can be sent and received through the specified APN.

show crypto ipsec sa peer 193.218.97.148 peer address: 193.218.97.148 Crypto map tag: OUTSIDE_MAP, seq num: 590, local addr: 195.167.65.105	inbound esp sas: spi: 0xBD0FEB96 (3171937174) SA State: active transform: esp-aes-256 esp-sha-256-hmac no compression
--	---

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	73 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status: Final

<pre> access-list VPN_ACL_APN_I-TRIALNEMO extended permit ip 10.26.240.32 255.255.255.240 172.25.12.0 255.255.255.0 log local ident (addr/mask/prot/port): (10.26.240.32/255.255.255.240/0/0) remote ident (addr/mask/prot/port): (172.25.12.0/255.255.255.0/0/0) current_peer: 193.218.97.148 #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0 #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0 #pkts compressed: 0, #pkts decompressed: 0 #pkts not compressed: 0, #pkts comp failed: 0, #pkts decomp failed: 0 #pre-frag successes: 0, #pre-frag failures: 0, #fragments created: 0 #PMTUs sent: 0, #PMTUs rcvd: 0, #decapsulated frgs needing reassembly: 0 #TFC rcvd: 0, #TFC sent: 0 #Valid ICMP Errors: 0, #Invalid ICMP Errors rcvd: 0 #send errors: 0, #recv errors: 0 local crypto endpt.: 195.169.65.105/4500, remote crypto endpt.: 193.218.97.148/4500 path mtu 1500, ipsec overhead 86(52), media mtu 1500 PMTU timer remaining (sec): 0, DF policy: copy-df ICMP error validation: disabled, TFC packets: disabled. current outbound spi: CEB062D3 current inbound spi : BD0FEB96 </pre>	<pre> in use settings = {L2L, Tunnel, NAT-T-Encaps, IKEv2, } slot: 0, conn_id: 72414, crypto-map: OUTSIDE_MAP sa timing: remaining key lifetime (kB/sec): (4193280/3413) IV size: 16 bytes replay detection support: Y Anti replay bitmap: 0x00000000 0x00000001 outbound esp sas: spi: 0xCEB062D3 (3467666131) SA State: active transform: esp-aes-256 esp-sha-256-hmac no compression in use settings = {L2L, Tunnel, NAT-T-Encaps, IKEv2, } slot: 0, conn_id: 72414, crypto-map: OUTSIDE_MAP sa timing: remaining key lifetime (kB/sec): (4193280/3413) IV size: 16 bytes replay detection support: Y Anti replay bitmap: 0x00000000 0x00000001 nymaFW2140-1/corp/pri/act# </pre>
---	--

Moreover, OTE deployed an RTMP server (virtual function) that acted as a content cache/ buffer between the output video and the smartphone application. The server characteristic as shown from the proxmox environment are depicted in Figure 63.

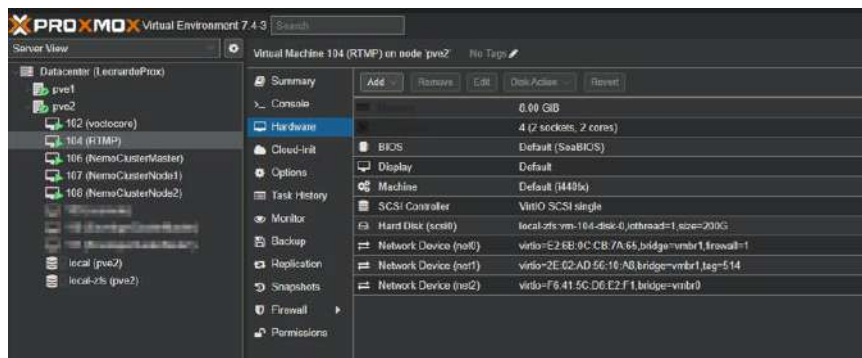


Figure 63 The proxmox environment.

5.1.1.2 Internal VM service implementation

The main deployment site was hosted at OTE premises in Athens, where a dedicated virtualized infrastructure was established using Proxmox VE. On top of this virtualization layer, a set of virtual machines (VMs) was deployed to support the operation of NEMO-compliant services. Among these, a dedicated Kubernetes cluster was provisioned to orchestrate the media-related components of the pilot. This Kubernetes cluster functioned as the local NEMO media cluster, successfully connected to the central NEMO infrastructure, forming part of the wider federated meta-OS environment. This

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	74 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

In parallel, Telefónica I+D (TID) deployed an additional remote production server at its premises in Madrid, which was securely connected via VPN to the NEMO infrastructure in Athens. The goal of this configuration was to enable remote production, allowing media professionals located in Spain to supervise, manage, and process the live video contributions being captured at the race venue in Athens. This setup offered a realistic demonstration of distributed content production over a federated edge-cloud infrastructure, showcasing the flexibility and operational potential of the NEMO framework.

Figure 64 shows the console output on the Proxmox core, receiving TCP traffic from TID. The throughput reached approximately 33.1 Mbits/sec, with 39.7 MBytes of data transferred, confirming a generally stable connection on the receiver's end.

- TCP mode: delivering up to 34.5 Mbit/s towards the core
- UDP mode: sending a 10 Mbits/sec stream, achieving low jitter (0.124 ms) and minimal packet loss (0.12%)

These results demonstrate the effectiveness of the VPN-based connection in the NEMO framework, supporting low latency and high-quality media for distributed live production workflows.



Document name:	NEMO Living Labs use cases evaluation results - Final version					Page:	75 of 195
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

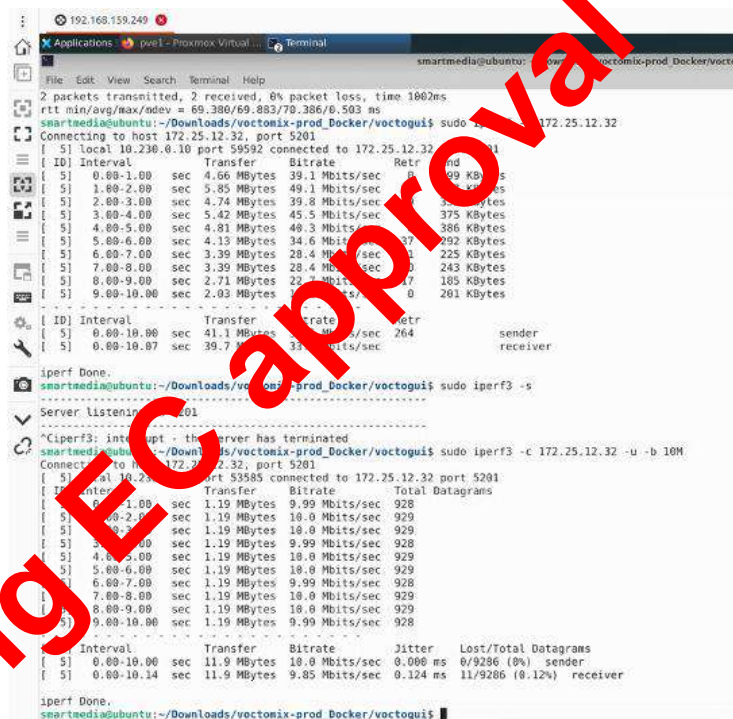


Figure 65 TCP and UDP Network Performance Tests from TID to Proxmox



Figure 66 Remote production environment in TID premises

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	76 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Deployment of the services adhered to the standardized NEMO pipeline using FluxCD Figure 67 for continuous delivery. Each service was defined declaratively through dedicated YAML manifests stored in the repository. Leveraging a GitOps-based approach, all changes were version-controlled and automatically synchronized with the cluster, ensuring secure, consistent, and reproducible deployments in line with project-wide DevSecOps best practices.

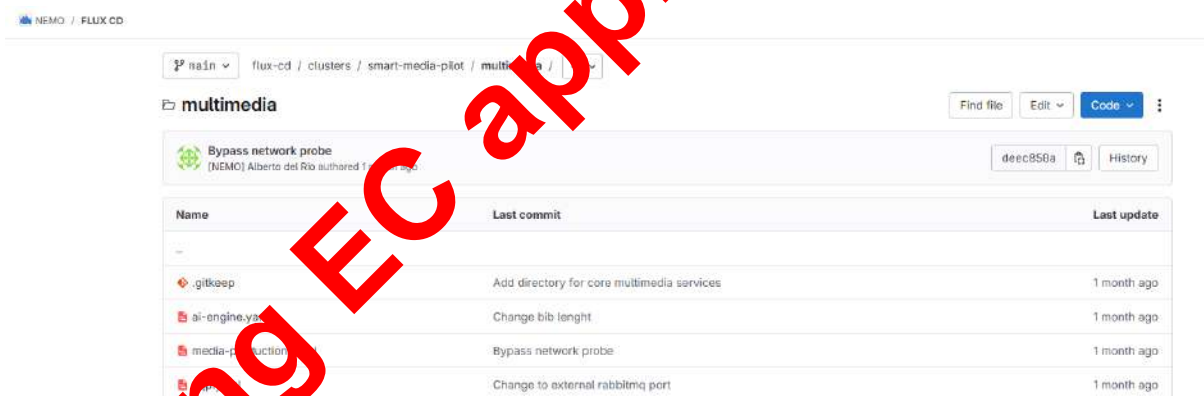


Figure 67 FluxCD environment for smart media pilot

It is worth noting that the automatic deployment process was restricted to the media-specific cluster, as a deliberate choice to isolate the media trial from other verticals, facilitating independent debugging and performance monitoring. All necessary Helm charts and Kubernetes manifests were version-controlled and aligned with the NEMO component registry and configuration standards.

A crucial part of the integration was the use of NEMO's RabbitMQ message broker, configured with a dedicated queue named uc-media. This queue served as the central data pipeline between backend services and the mobile application. Components producing metadata (e.g., video quality probe, AI engine...) published their messages to the uc-media queue. These messages were later consumed by the mobile application, enabling real-time updates and synchronization of media content and control flows.



Figure 68 RabbitMQ specific queue for use case

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	77 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

The live trial was conducted at the Egaleo Stadium in Athens, which was prepared to simulate a realistic live race scenario. The venue was configured with multiple recording points, and several runners were mobilized across different segments and contextual situations—including varied lighting, occlusions, and movement patterns—to emulate real-world operational constraints for video analysis and metadata extraction. This allowed the system to be tested under a range of conditions reflecting the complexity of live sports broadcasting and remote orchestration.

Integration activities included:

- Finalization and provisioning of the Proxmox-based VM infrastructure at OTE.
- Setup of the Kubernetes media cluster, integrated into the NEMO federated environment.
- Connectivity setup between the local media cluster and the central NEMO orchestrator.
- Establishment of VPN connectivity between TID (Madrid) and OTE (Athens) to support remote video production workflows.
- Deployment of core services via FluxCD, including configuration and CI/CD pipelines.
- Configuration and validation of the RabbitMQ uc-media queue.
- Physical preparation and testing at Egaleo stadium, including live video capture and movement scenarios with runners deployed across the field.

5.1.1.3 The Smart Phone Race Application

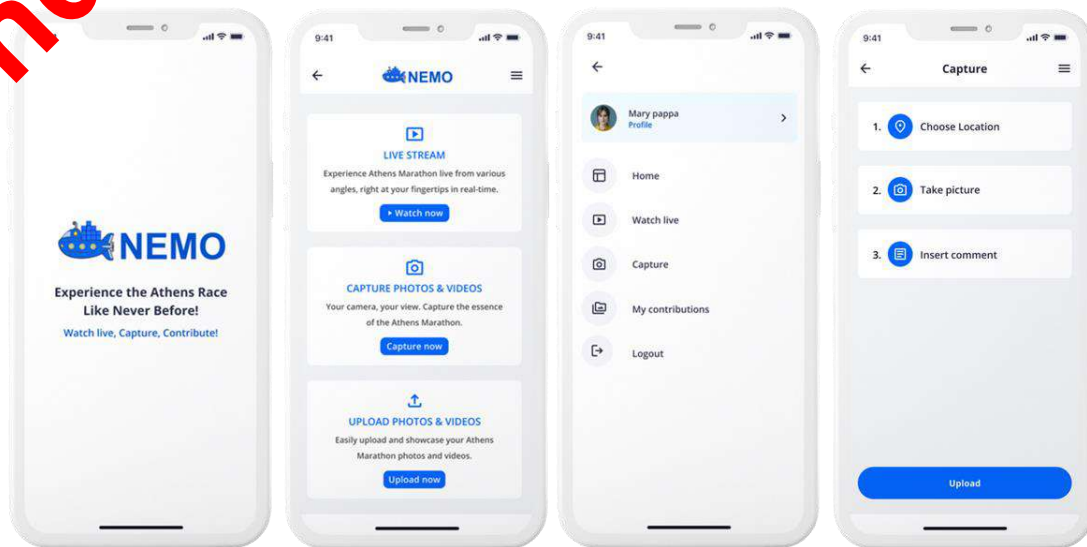


Figure 69 NEMO mobile app (1)

The NEMO mobile app provides a way to experience a Marathon by bringing spectators and participants closer to the action in real time. With live streams available from multiple locations along the route, users can follow the race from different angles as it happens. Beyond just watching, the app allows users to take their own photos and videos during the event and upload them directly, helping to create a shared, crowd-sourced collection of moments. Each user has a dedicated “My Contributions” section where their uploads are saved to revisit. With its simple layout and interactive features, NEMO makes following and contributing to large-scale events like the Athens Marathon more engaging and personal.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	78 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

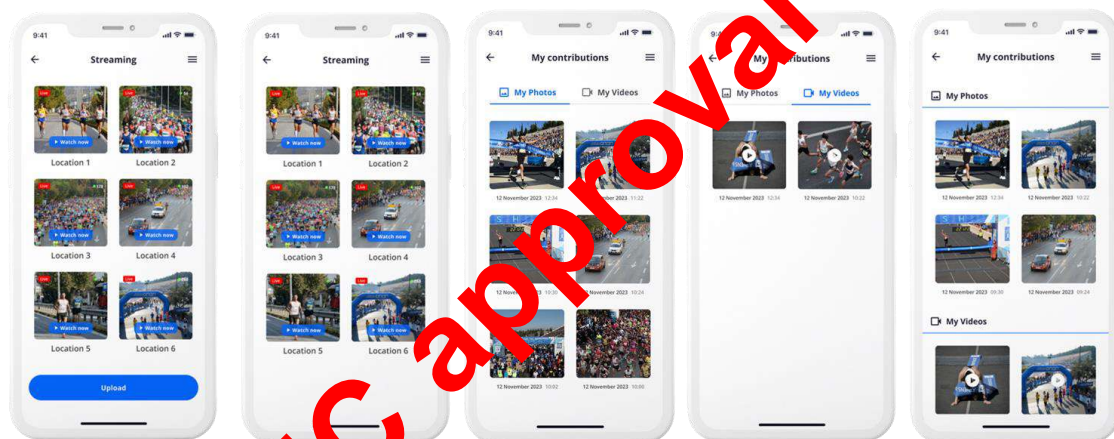


Figure 70 NEMO mobile app (2)

5.1.2 Test scenario verification

The Smart Media trial technical validation was carried out based on the test scenario detailed in the table below

Scenario: SC01_Test_Scenario_1	
Scenario ID	SC01_Test_Scenario_1
Objective	Enhance the live sport event spectating experience by enriching the content through AI driven data and content analysis via NEMO components and manual production control. Nemo will provide the base components to facilitate communication, network access, migration etc.
Description	During the race, media content is captured by many spectators and selected runners along the running circuit using smartphones/tablets and GoPro cameras, and if available IP cameras and drones. Incoming content is automatically processed, annotated, and rendered (partially on the device using already trained AI/ML models and partially at the edge), and a selection is directly broadcast (e.g. via social media) based on location info of the (top) runners and interesting events during the race (e.g. based on contributor annotation). The audience has the option to improve their contributions and can interact with contributors in case of specific race incidents. The emphasis is on real-time user generated content processing and rendering in the the cloud. The AI/Models were trained to recognize the Racing Bib Numbers on each athlete in order to first understand the runners in each stream and then enhance the positioning identification of the stream and runners in it. NOVO's mobile app aims to engage runners/spectators/users, facilitating both enhanced content experience and feeding captured events combined (pictures, videos and/or annotations, time and location data).
Features to be tested	meta-Orchestrator, Intent-based SDK (manual authoring), Identity Management & Access Control (users), Intent-based Migration-Controller, CF-DRL (for AI), MOCA (for monetization services), CMDT (for Delivery manager optimization training
Requirements addressed	SC_01_FR01-22, SC_01_NFR01-06

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	79 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Scenario: SC01_Test_Scenario_1	
KPIs	KPI_SC_01_1-7
Prerequisites	The NEMO platform was accessible and running. More than 10 users were be available for using the App and the Trial infrastructure. More than 5 video feeds were as sources available at any time. Sources were user content provided through the App and a 360 camera. A 4-5G network covered all places in order to provide high bandwidth and multipath connectivity. User with Phones and App were installed. Media Managers for receiving streams and distribution of processed streams were available through NEMO registered components. Production Control system for stream authoring was present as NEMO plugin service. An AI component as a NEMO resource for automatic annotation was used. The Network Probe was used for Media adaptation on network quality.
Test steps	NEMO was deployed and operational. This included: • Media Production Engine, which includes virtualized video editing tool, virtualized video coding, virtualized video mixer, virtualized video compressor. • Cognitive services, which includes data processing, virtualized video annotation tool, AI engine, QoE optimizer, data fusion with external data. • Emission selector, which includes AI engine, Virtualized Media Delivery Manager for delivery. • Smartphone cameras of spectators and a 360 camera connected to platform and send real time data. • The platform users (professionals) are registered to the platform. Production control commences and AI component started analyzing video feed at the same time NEMO collects monitoring data and workloads. Users interacted with the APP, choose what their desired viewing experience (general overview, specific runner, action event etc.), but also contribute uploading video and actions as they happen live.
Success state	The users had access to enriched content that included: • GPS location of the cameras and runners. • AI driven BiB and street image detection for running events. • A program signal created by a professional technical director. • Runners' numbers recognition for race positioning tracking.
Failure state	AI component did not fail to identify runner BiB numbers. Production control provided qualitative stream. App did not fail to stream or view specific events. GPS signal was enhanced.
Responsible for testing and implementation	UPM, NOVO,OTE, TID

Table 5 Test scenario Smart Media

To support the Smart Media use case in the “Round of Athens Race” pilot, a modular and cloud-native architecture was designed and deployed. This architecture was structured to address the challenges of live multimedia acquisition, cognitive processing, content orchestration, and real-time content delivery, while also demonstrating the core capabilities of the NEMO platform.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	80 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

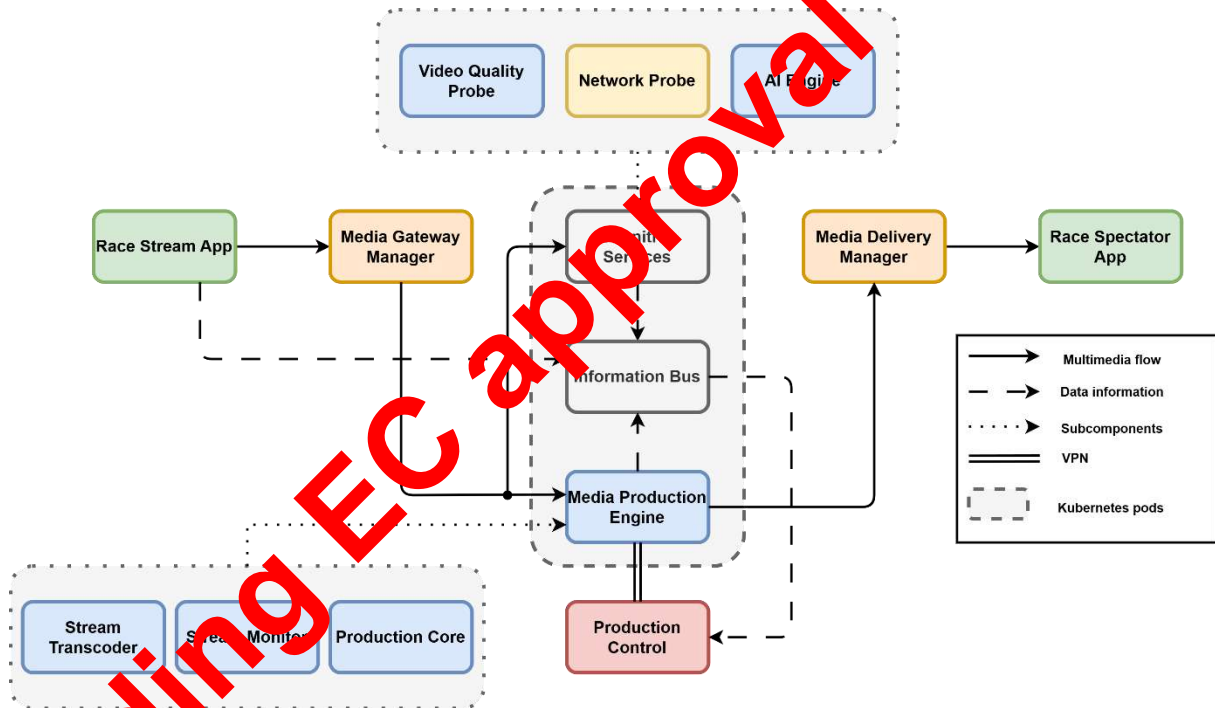


Figure 71 Architecture of Smart Media Trial

As shown in Figure 71, the architecture consists of several interconnected functional modules grouped by their role in the multimedia pipeline. These components are organized across distributed environments (on-site in Athens and remote in Madrid), with a subset of them deployed as Kubernetes pods, following NEMO's orchestration guidelines.

The pipeline starts with video acquisition from mobile or event sources and ends with real-time delivery to mobile spectators, including enriched metadata and AI-generated insights. A key feature of this setup is the remote production capability, which allows the management of media workflows from geographically distant sites through a VPN-secured connection. This enabled production control and stream supervision from Madrid (TID), while media was being captured and processed on-site in Athens (OTE).

Each component is categorized based on its role in the media pipeline—acquisition, processing, cognitive enrichment, production control, and delivery:

- **Race Stream App**: Captures live video from user devices (e.g., smartphones) during the race.
- **Media Gateway Manager**: Bridges external media inputs with internal processing pipelines.
- **Media Production Engine**:
 - **Stream Transcoder**: Converts incoming streams to desired formats or bitrates.
 - **Stream Monitor**: Tracks stream bitrate, encoding, RAM usage...
 - **Production Core**: Coordinates real-time media mixing, scene switching, and source selection.

In the case of the UPM site, a VM named “voctocore” was provisioned in Proxmox, specifically design to host the core component of the media manager. To ensure portability across environments, this core service was containerized using Docker.

As show in Figure 72, the vinctocore VM is configured with 23 GiB of memory and 64 virtual CPU cores. In Figure 73 it can be seen the initialization of the video manager pipeline, where multiple TCP ports are configured. In addition, Figure 74 illustrates the runtime resource usage of the VM.

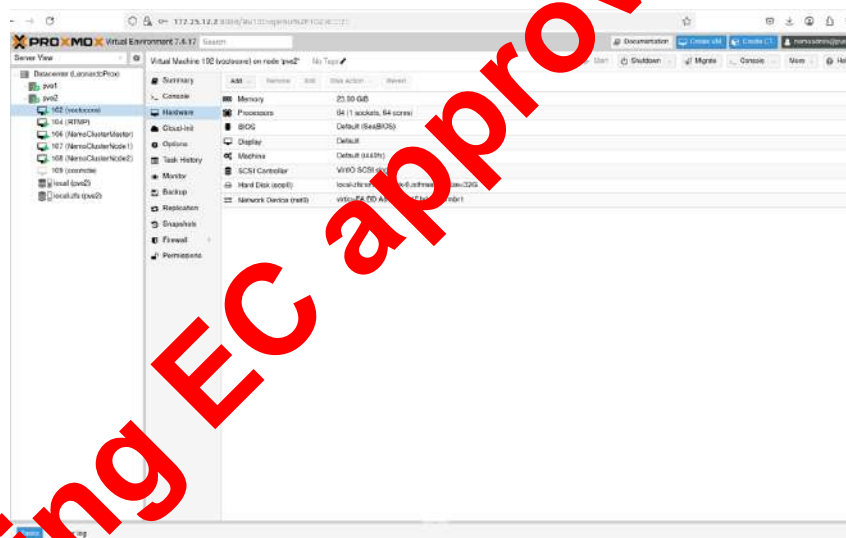


Figure 72 Hardware configuration of the vinctocore VM deployed on Proxmox.

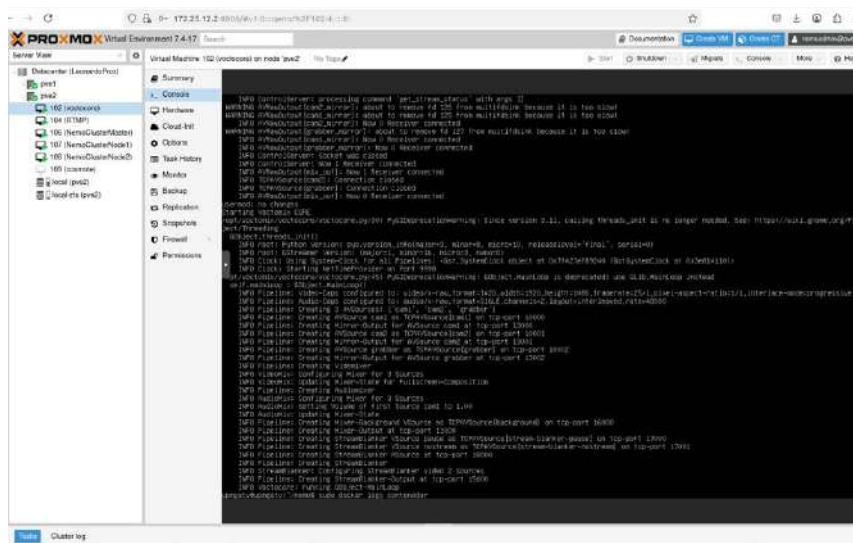


Figure 73 Console output showing the initialization sequence of the video manager core.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	82 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

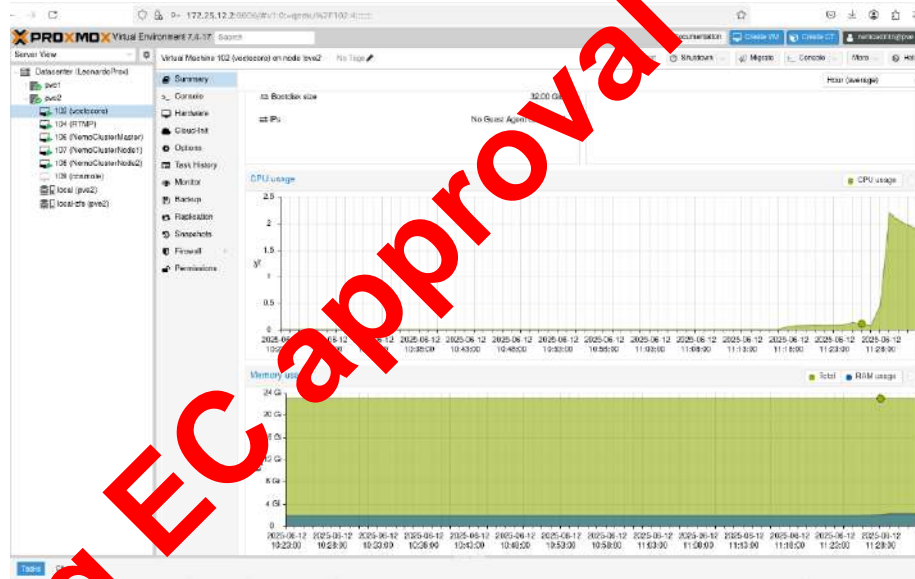


Figure 74 CPU and RAM usage graph for the vocotcore VM.

- **Production Control:** External control interface used for managing streams remotely (located in Madrid, connected via VPN).

The graphical user interface used for production management is based on the Voctogui component, a GTK-based frontend that connects to the Voctomix core and provides visual feedback of the live video mix[6]. This interface operates with hardcoded TCP port assignments and listens to the video streams transmitted by the core, which allows direct, low-latency interaction with the video pipeline.

Voctogui enables remote production operations by offering a set of predefined layout modes that can be dynamically selected during the live execution, including: Fullscreen, Picture in Picture, Side by Side Equal and Side by Side Preview.

The interface handles up to three distinct input sources, cam1, cam2 and cam3, which can correspond to different video streams of files. These inputs are synchronized and managed by the core, with Voctogui reflecting the current configuration and visual layout in real time.

From an implementation perspective, the client must be specified to connect to the IP address of the machine hosting the core service. In this case, the address corresponds to the Proxmox VM (172.25.12.2). This setup allows Voctogui, running remotely (e.g., from Madrid) to control and monitor the live production hosted at the UPM node over the VPN.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	83 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

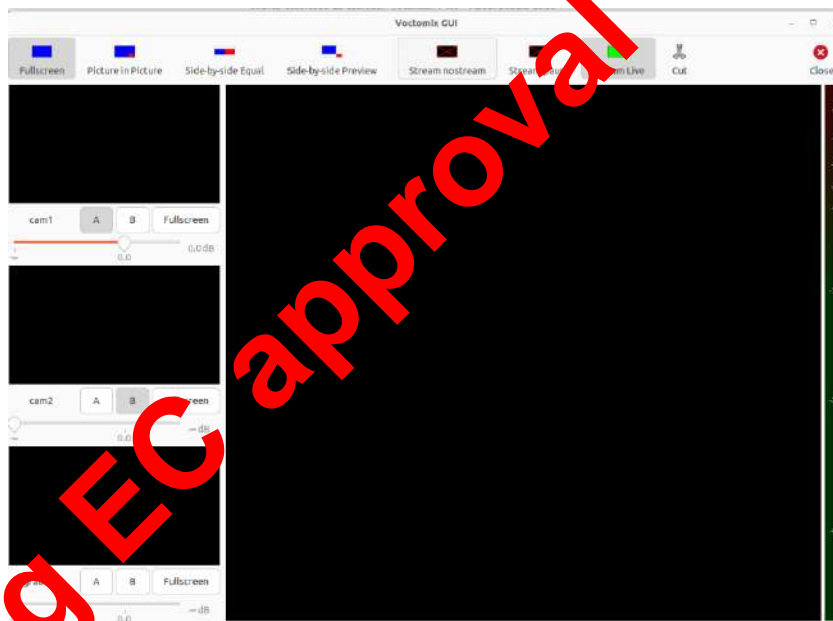


Figure 75 Voctogui Production Interface

- **Cognitive Services:** Hosts AI-driven decision-making and semantic processing functions.
 - **Video Quality Probe:** Evaluates the visual quality of streams for adaptive processing or alerts. It is a key component designed to assess the perceptual quality of video streams in real time. Its main objective is to support adaptive streaming mechanisms and generate alerts when quality degradation is detected, based on objective, data-driven metrics. This approach addresses the inherent limitations of traditional subjective evaluation methods, such as Mean Opinion Score (MOS) values, which are costly, time-consuming, and lack standardization. The Video Quality Probe component ensures a scalable and consistent evaluation of the Quality of Experience (QoE) by automating the estimation of MOS.

At the core of the system lies a machine learning model trained on a rich and heterogeneous dataset comprising over 3.000 annotated video sequences drawn from established public video databases. This dataset is designed to capture a wide range of video content and quality degradations, ensuring that the machine learning model is robust and generalizes well across various real-world scenarios.

The feature extraction process is grounded in the use of a comprehensive set of video quality indicators derived from the AGH Video Quality of Experience Team's tool[13]. The dataset consists of 70 features that represent various quality dimensions, capturing both spatial and temporal impairments, such as blockiness, blur, flickering, freezing, contrast variations, noise, and exposure issues. This extensive feature set enables the model to capture both global trends in video quality and localized artifacts (alerts), providing a multi-dimensional perspective on video quality.

In developing the MOS estimation model, a systematic approach was followed. Initially, standard scaling was applied to the dataset to normalize the features, ensuring that each variable contributed equally to the model. To address the high dimensionality of the feature space and mitigate potential overfitting, Principal Component Analysis (PCA) was employed for dimensionality reduction. PCA was used not only to enhance

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	84 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

computational efficiency but also to identify the most significant underlying patterns in the data. Two predictive algorithms were subsequently tested in separate pipelines. One pipeline integrated a Random Forest Regressor, while the other utilized a Ridge regression model. Both pipelines were implemented using the scikit-learn library and optimized through a estimator framework that systematically explored different hyperparameters settings to optimize the mean explained variance across five fold cross validation.

During the cross-validation phase, the models were evaluated using several metrics, including explained variance, mean absolute error, mean squared error, and r2 score. The evaluation process revealed that the pipeline incorporating the Random Forest Regressor performed notably well, with metrics indicating that the model captured a significant portion of the variability in MOS. The pipeline based on the Random Forest model achieves a mean r2 score of 0.15 and a mean negative squared error of -0.61 during cross validation.

This performance underscores the potential of this approach for automating video quality assessment. The integration of standard scaling, PCA, and the Random Forest Regressor results in a system that can deliver accurate MOS predictions while being computationally efficient. These promising results indicate that the model is well-positioned to transform traditional subjective quality assessment into an automated, scalable, and data-driven process.

Video Quality Probe can facilitate the early detection of quality degradation by enabling real-time video quality monitoring. The component supports adaptive streaming strategies and optimizes resource allocation in video delivery systems. The ability to accurately predict MOS from a diverse set of video quality features not only enhances operational efficiency but also provides valuable insights that can drive improvements in video compression, encoding, and transmission strategies.

During the use case event, Video Quality Probe was deployed to test its performance in a live setting using the video streams from the project's use case. The component captured the video streams in real time, analyzing the visual quality and generating results at 5-second intervals. For each time interval, Video Quality Probe computed a predicted MOS value, extracted detailed video quality metrics, and triggered alerts when degradations were detected. All generated information was transmitted to a RabbitMQ messaging queue, enabling integration with other components of the platform. This test validated the Video Quality Probe ability to operate under realistic conditions Figure 75.

Figure 76 presents an example, in JSON format, of the message sent to RabbitMQ. It contains a summarized representation of the information managed by the Video Quality Probe, including predicted MOS, quality metrics, and alerts for a specific 5-second video segment. The JSON message encapsulates summarized information about the analyzed video segment. It includes metadata about the stream source and technical parameters (path, datetime, width, height, frame_rate and num_frames), followed by a detailed set of video and audio quality metrics and alerts:

- Video quality indicators such as spatial activity (108.93), temporal activity (40.82), blur (3.75), blockiness (0.93), block loss (1.02), exposure (128.56), contrast (48.51), interlace artifacts (0.0004), noise (0.48), and flickering (0.37) provide a comprehensive view of both spatial and temporal impairments.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	85 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Additional flags (e.g., letterbox, pillarbox, freezing, blackout) indicate the presence or absence of specific visual artifacts.

- Audio quality indicators include the average and peak audio volume (volume_mean: -23.8 dB, volume_max: -7.7 dB).
- A set of binary alert flags (for video and audio) highlight whether specific thresholds were exceeded during this interval (alert_blur, alert_blockiness, alert_blockloss, alert_freezing, alert_uniform_frame, alert_black_frame, alert_no_audio, and alert_silence).
- Quality value or MOS prediction provides an estimate of perceived quality for the segment (mos_value: 3.55).

```
{
  "path": "udp://vqp-2.multimedia.svc.cluster.local:2222",
  "datetime": "2025-05-06T15:10:06.939588",
  "width": 1920,
  "height": 1080,
  "frame_rate": 25,
  "num_frames": 55,
  "spatialactivity": 108.92622036363636,
  "temporalactivity": 40.82084185185185,
  "blur": 3.7475454545454547,
  "blockiness": 0.9308259999999998,
  "blockloss": 1.0201296363636365,
  "freezing": 0.0,
  "letterbox": 0.0,
  "pillarbox": 0.0,
  "blackout": 0.0,
  "exposure": 128.56363636363636,
  "contrast": 48.505722000000001,
  "interlace": 0.00043763636363636357,
  "noise": 0.48309090909090896,
  "flickering": 0.37534166666666667,
  "volume_mean": -23.8,
  "volume_max": -7.7,
  "alert_blur": False,
  "alert_blockiness": False,
  "alert_blockloss": False,
  "alert_freezing": False,
  "alert_uniform_frame": False,
  "alert_black_frame": False,
  "alert_no_audio": False,
  "alert_silence": False,
  "mos_value": 3.55
}
```

Figure 76 Example of a JSON-formatted message sent by the Video Quality Probe and stored in RabbitMQ.

Finally, Figure 77 shows the predicted MOS values generated by Video Quality Probe during the analysis of a 90-second video stream captured on the day of the event. For each 5-second video segment, the tool produced a predicted MOS. Values ranged between 2.4 and 3.6, indicating an overall stable video quality without significant impairments. The average MOS value obtained during the video analysis was 3.04, with approximately 52.6% of the measurements exceeding the 3.0 threshold. This indicates a generally acceptable QoE in more than half of the analyzed content. Moreover, the analysis of the information confirmed that key visual metrics such as blur, blockiness, exposure, and contrast remained within acceptable ranges, and no major quality degradation events were detected.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	86 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

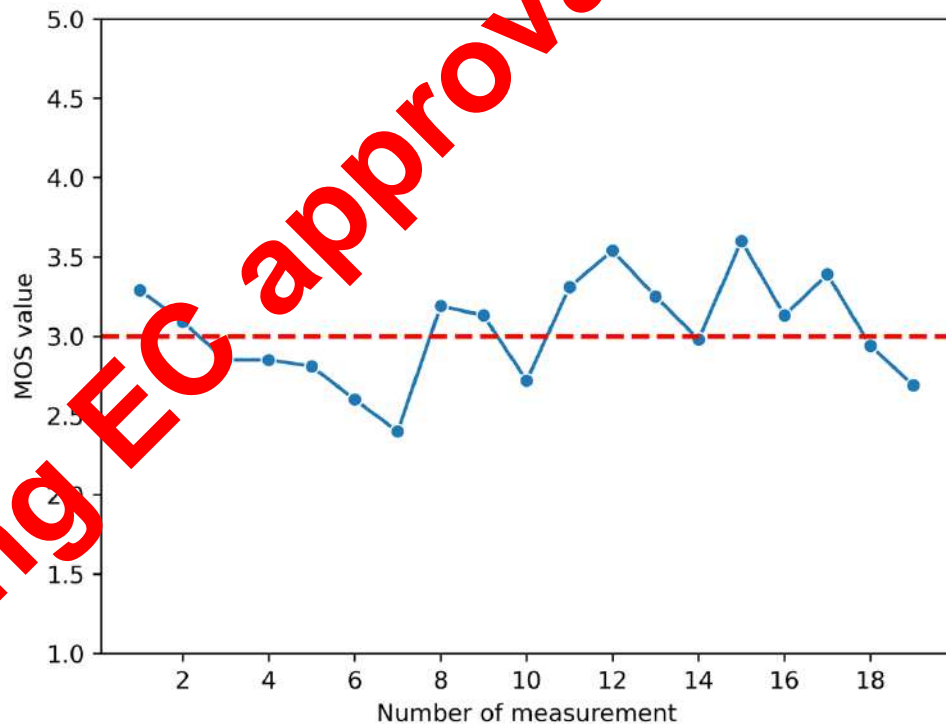


Figure 77 Predicted MOS values generated by Video Quality Probe during the analysis of a 90-second video stream.

During the use case, the system operated in real-time, sending all quality reports to a RabbitMQ queue, thus demonstrating its capacity for seamless integration and live monitoring. This successful test validated the probe's effectiveness in providing granular and actionable quality insights under realistic operating conditions.

AI Engine

The AI Engine runs inference tasks (e.g., object detection, runner identification) over incoming video. The component is based on pretrained YOLOv8 nano models, chosen specifically to optimize latency during inference. The workflow is organized into three sequential stages: 1) person detection, to identify the participants within the video frames 2) bib detection, to localize the runner bibs once a participant has been identified 3) bib number recognition, to extract and classify the bib numbers from detected bib regions. The detection and recognition pipeline leverages models trained on specialized datasets: RBNR[9] and TGCRCBNW[10] for runner detection; BDBD[11] for bib detection; ; and SVHN[12] for digit recognition. Model training was performed on external machines, while the deployed versions are lightweight nano variants of YOLOv8.

Media Delivery manager

The Media Delivery Manager distributes final media streams to playback clients, using RTMP client. The RTMP server had two network interfaces:

- **Input Interface:** Located in the subnet 172.25.12.0/24, listening for video input.
- **Output Interface:** Located in the subnet 172.25.14.0/24, with port forwarding enabled to serve the stream externally.

A diagram illustrating the setup is depicted in the following Figure 78. More specifically, the input interface received the content from the virtual functions of the workflow while the output interface served as the access point for the output applications.

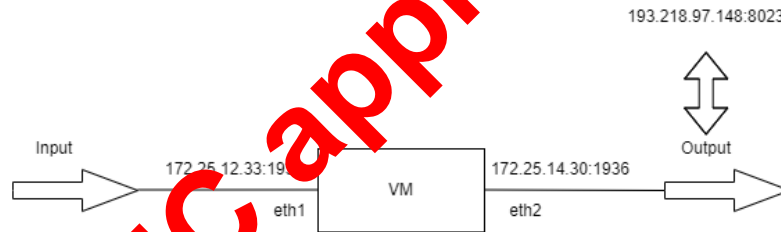


Figure 78 Rtmp Server connection setup

Race Spectator APP

The Race Spectator android smartphone application is used by end users to view the race and receive synchronized media data updates. It is described in the sections above.

Network Probe

The Network Probe monitors network conditions to inform bandwidth-sensitive operations.

In media production, NWDAF (Network Data Analytics Function) acts as the analytics engine of the 5G core, collecting data from network functions (NFs), application functions (AFs), and operations systems (OAM) to provide actionable insights that improve service quality and reliability. By analyzing information such as bandwidth, jitter and delay, NWDAF enables dynamic optimization of network resources to meet the demanding requirements of live media workflows, such as video streaming, remote production, and contribution feeds. Through exposure via the NEF (Network Exposure Function), NWDAF was integrated with external media applications, allowing media content providers to adjust production processes and user experience in real time based on network conditions to ensure seamless, high-quality content delivery.

Time-Sensitive Networking (TSN)

The 3GPP Release 16 introduces TSN (Time Sensitive Networking) and multiple updates to the 5G. The IEEE 802.1 TSN set of standards defines TSN to deliver networking with accuracy by supporting deterministic, real-time communication. With the 5G Ultra Reliable Low Latency Communication (URLLC) feature, TSN over 5G will achieve precision time synchronization, reliable and low-latency networking. Release 16 supports absolute time reference delivery as a service to UEs over the 5G System. Enabling TSN over 5G requires end to end TSN time synchronization and TSN payload(communication) over the 5G System.

TSN time synchronization takes place between TSN nodes which are connected at the edge of a 5G system. The TSN nodes are called the TSN Grand Master (GM) and Slave server (End Station), also possible to connect multiple TSN slave devices, connected to the edge of the 5GS for end to end TSN time synchronization. The End Station synchronizes with the TSN-GM through the 5G system which acts as an Ethernet link. Figure 79 shows the setup for end-to-end TSN time synchronization.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	88 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Integration of TSN into 5G requires new network functions called Network-side TSN Translator (NW-TT) and Device-side TSN Translator (DS-TT). The NW-TT receives those gPTP messages and makes an ingress timestamping (TSi) for every gPTP event message and uses the cumulative rateRatio obtained within the payload of the gPTP message in order to calculate the link delay from the upstream TSN node (gPTP entity) expressed in TSN grandmaster (GM) time. UPF then forwards the gPTP message from TSN network to the UEs through all PDU sessions terminating in this UPF that the UEs have established to the TSN network. All gPTP messages are transmitted on a QoS Flow that complies with the residence time upper bound requirement specified in IEEE Std 802.1AS. A UE receives the gPTP messages and forwards them to the DS-TT. The DS-TT located in the user equipment (UE) receives the gPTP messages and creates an egress timestamping (TSe) for the gPTP event messages for external TSN working domains connected to the UE. The residence time spent within the 5G system (5GS) for this gPTP message is calculated by the difference between TSi and TSe. The DS-TT converts the residence time spent within the 5GS in TSN GM time with the rateRatio provided by the gPTP message. It also adds the calculated residence time and removes the TSi from the payload suffix field of the gPTP message that is sent downstream to the TSN node connected to the UE. The sum of the UE-DS-TT residence time and the PDB of the QoS Flow needs to be lower than the residence time upper bound requirement for a time-aware system specified in IEEE Std 802.1AS. In general, the DS-TT translates the TSN 802.1 protocols on top of 5G networks but cannot be assumed that would be a DS-TT capable UEs.

The simulation setup comprises a network of interconnected time aware devices. In the scenario the DS-TT is not integrated with the 5G modem and SIB9 cannot be provided to the DS-TT alternative measurement performed using an additional GM to be used for time reference in the NW-TT and the DS-TT, as shown in Figure 79. The measurement data has been collected and depicted in Figure 80 as a time offset between the GM and TSN slave node after end to end TSN time synchronization. The result shows that the time difference (offset) between the GM and TSN slave node stays in order of 7-8 μ s.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	89 of 195
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status: Final

Time Offset (us) vs. Index



Figure 80 Time Offset of TSN slave instance after TSN time synchronization.

There are some peak values observed on the result where the time offset jumps up to 15us, this is due to the UE rebooting itself and disconnecting itself from the core networks. When the UE reboots itself, it creates a discontinuity on the PTP message exchange between the GM and Slave node and the Slave node becomes out of sync from the GM clock. The UE reconnects to the core network within a fraction of seconds, which allows the TSN slave device to synchronize back with the GM before it drifts significantly. Overall, after having a successful synchronization with the GM the slave server has a mean time offset of around 8us. An oscilloscope is also used in the setup to observe and analyse the same result discussed above. Figure 81 shows the PPS output from the TSN nodes: C1 displays the Slave clock 1PPS, C2 displays the Master clock 1PPS, and the statistics shows the time offset between the GM and the slave server clock as a delay. The result shows that the time difference between the slave device and the GM is in order of 8us and a standard deviation of 0.5us. For the time span where the data is collected, the standard deviation stays around 4us which shows that the synchronization mechanism is working steadily.



Figure 31 C1: slave-device PPS output, C2: GM clock 1PPS output, and time difference between GM and Slave clock.

The result of the simulation shows that a successful end-to-end TSN time synchronization over the 5G system can be achieved with a small time offset between TSN nodes.

5.1.3 Final validation results and KPIs Evaluation

Table 6 Summarizes the KPIs that had been defined and refined in previous deliverables D5.1, D5.2[3], D5.3[4]. A detailed analysis follows about how each one was achieved.

KPI ID	Name	Description	Measurement/Assesment method	Target
KPI_SC_01_1	Number of users	Number of users consuming the AV content	Users connected through NEMO viewing app. Covered by Video and Photo footage on site.	10 - 100
KPI_SC_01_2	QoE	QoE value of content delivered	Covered by log entries of the Video Probe	Mean MOS>3
KPI_SC_01_3	Number of sources	Number of AV simultaneous sources used to create the final content	MPE/Production center inputs. Covered by screenshots of components running in the VM Media Processing Engine.	>=5

KPI ID	Name	Description	Measurement/ Assessment method	Target
KPI_SC_01_4	Number of VNFs	Number of AV VNFs deployed on the NEMO edge & cloud	Number of AV VNFs deployed and orchestrated on the NEMO edge & cloud. Covered by screenshots of components running in the VM Media Processing Engine.	>5
KPI_SC_01_5	Enriched content	Number of non-media data sources included in the content enrichment	Number of non-media data sources included in the content enrichment. Covered by screenshots of JSON metadata from RabbidMQ.	>3
KPI_SC_01_6	Multipath connectivity	Multipath connectivity to enable extreme high bandwidth	Covered by 360 Video camera on site working/streaming.	>=12K for 360o video & 3D textures
KPI_SC_01_7	Accurate positioning	Accuracy of positioning of events	Improvement of accurate positioning of events in terms of geolocation. Covered by log data from the Smart Phone Race App	>= 25%

Table 6 KPIs Evaluation

The final validation phase of the Smart Media pilot focused on the real-time orchestration of multimedia streams, demonstrating the integration of AI-based analysis, network monitoring, remote production, and live delivery to end users. The entire multimedia pipeline was exercised using actual race conditions.

The flow begins with multiple users actively streaming video from mobile devices via the Race Stream App. These streams were captured in real-time and transmitted using UDP to a set of public IP endpoints on the OTE Proxmox environment. Each stream was directed to a unique port associated with a downstream processing pipeline:

- 193.218.97.148:31111 → 172.25.12.36:31111
- 193.218.97.148:32222 → 172.25.12.36:32222
- 193.218.97.148:33333 → 172.25.12.36:33333

Here, the public IP addresses (193.218.97.148) represent the exposed interfaces of the Proxmox infrastructure, while the internal IP (172.25.12.36) corresponds to the Kubernetes cluster hosting the media services. The Media Gateway Manager handled the port forwarding and ingestion, redirecting incoming streams to three distinct instances of the Media Production Engine, each listening on udp://0.0.0.0:1111, :2222, and :3333, respectively.

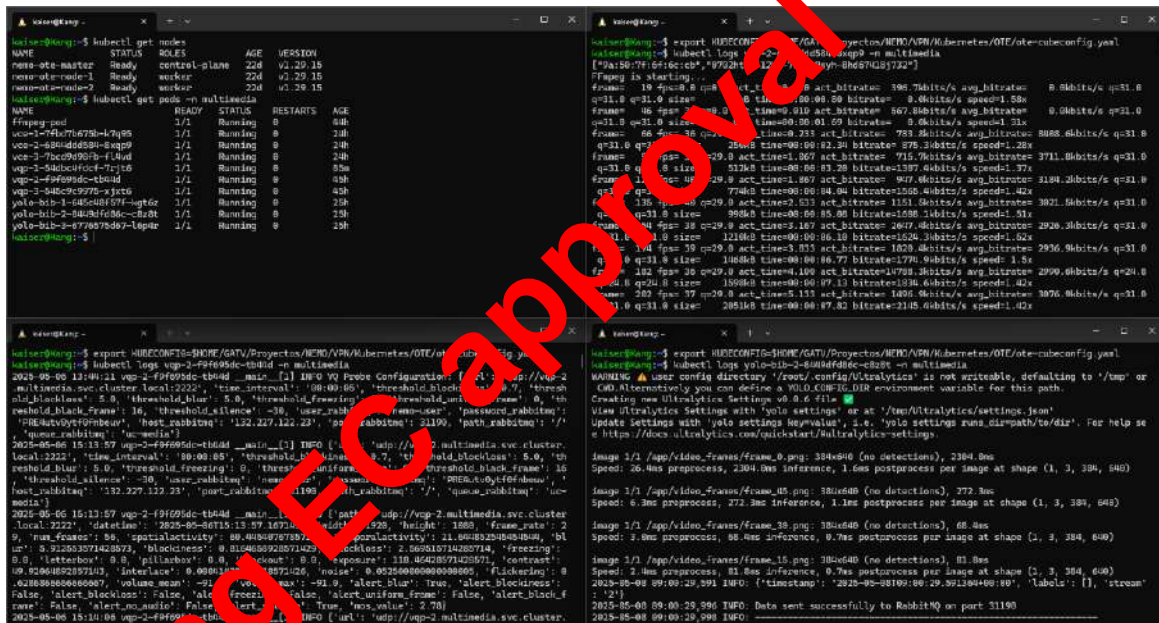


Figure 82: Terminal screenshot of use case workflow: top-left) Pods deployed in the node; top-right) Logs for one of the Stream Transcoder, with some information from the Stream Monitor; bottom-left) Logs from video quality probe; bottom-right) Logs from AI engine

These input channels allowed for parallel ingestion of three distinct media sources (see Figure 82 for a screenshot of the multimedia workflow going into the main multimedia outputs), which were processed in real-time across multiple service layers. The system design ensured efficient demultiplexing of the streams and isolated routing to downstream consumers. Once inside the cluster, each video stream was analyzed concurrently by three major enrichment services:

- **AI Engine:** This module included an advanced YOLOv8-based object detection pipeline, customized and pretrained to identify:
 1. Human figures (runners).
 2. Bib papers worn on chests.
 3. Numeric digits within each bib.

The model was fine-tuned using a diverse dataset comprising race imagery and annotated video frames, ensuring high detection accuracy even under variable lighting and motion conditions. Detection results were serialized and forwarded as metadata to the uc-media RabbitMQ queue, enabling synchronized delivery to the spectator app.

- **Video Quality Probe:** This service monitored video fidelity in real-time, extracting visual indicators such as blockiness, block loss, and other video metrics. These metrics were synthesized into a Mean Opinion Score (MOS) estimate, representing an overall subjective quality rating.
- **Network Probe:** Captured video packets were analyzed to assess network performance parameters, including jitter, throughput stability, packet loss, and delay. These readings provided operational insight and helped contextualize any drops in quality or latency observed downstream.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	93 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

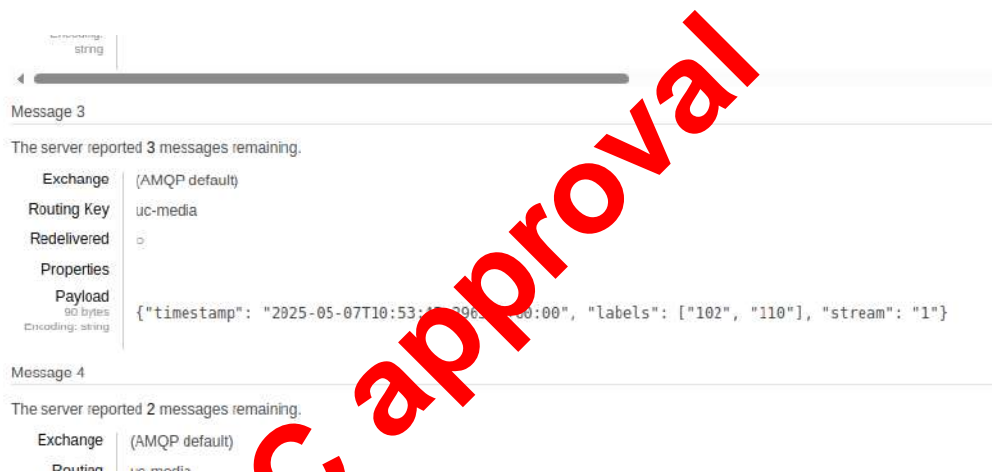


Figure 83 RabbitMQ screenshot with message example from AI Engine.

The processed streams were simultaneously routed to the Production Core, a specialized module responsible for scene selection, mixing, and forwarding. This module was hosted on a separate machine located at 172.27.12.111, still within the internal OTE network. The Production Core forwarded low-latency preview streams to Madrid via VPN, where production staff could monitor, select, and combine feeds in real-time.

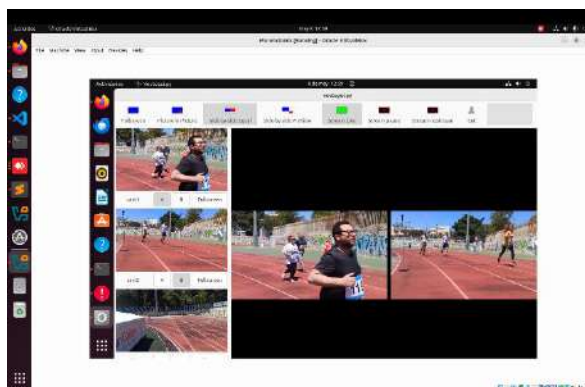


Figure 84 Screenshots from Production Control



Figure 85 Screenshots from Production Control

This demonstrated the effectiveness of the remote production workflow, enabled purely by secure network connectivity and without the need for on-site control. Once the content was curated and

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	94 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

finalized, the Production Core forwarded the resulting output to the Media Delivery Manager, targeting the RTMP entry point (rtmp://172.25.12.33:1935/live_input/test).

This RTMP endpoint was hosted inside the Media Delivery Manager, which converted and rebroadcasted the stream via a public interface. The Race Spectator App, running on standard mobile devices, subscribed to this public stream, allowing end-users to view the final video output enriched with contextual metadata and synchronized annotations.

All video transmissions, including those across the Proxmox and Kubernetes environments, remote production links, and final RTMP broadcast, were performed using UDP transport to ensure minimal latency and responsiveness.

As part of the final validation phase, a set of targeted KPIs defined for the Smart Media use case was systematically evaluated.

KPI_SC_01_1

Number of Users Consuming the Content: The trial successfully attracted between 15 and 25 viewers, connected from multiple geographic locations including Athens, Madrid, and other sites in Europe. These users accessed the live stream using the Race Spectator App, confirming engagement from both on-site and remote audiences. This result meets the target range of 10–100 users. See the log files below:

ID	URL	Created At	Longitude	Latitude	Comments	Filetype
326	nemo7 /nemo/8XJ2N5P9K7WQRT45.JPEG	08-05-2025 11:02	23.6785432	37.9985234	amazing atmosphere	photo
327	nemo7 /nemo/VAQ3H2L6Y9V8NT82.JPEG	08-05-2025 11:02	23.6791567	37.9982145		photo
328	nemo7 /nemo/RTS3K9P2M6ZQW4T41.JPEG	08-05-2025 11:08	23.6788901	37.9984678		photo
329	nemo1 /nemo/F5J8L2N4W7R8Q3S9.JPEG	08-05-2025 11:09	23.6794123	37.9987321		photo
330	nemo15 /nemo/T6G9K3M8P1XVZQ47.JPEG	08-05-2025 11:17	23.6786754	37.9983456	fantastic spirit	photo
331	nemo8 /nemo/D3H5L9N7R4S8WQ7R3.JPEG	08-05-2025 11:18	23.6792345	37.9986789		photo
332	nemo4 /nemo/V883K6M2Q5XZT1T51.JPEG	08-05-2025 11:22	23.6787891	37.9981234	wonderful	photo
333	nemo11 /nemo/A4F7J1L5P8W9Q3S2.MP4	08-05-2025 11:25	23.6795678	37.998567		video
334	nemo6 /nemo/C8E2H8K1N7R1X236.JPEG	08-05-2025 11:28	23.6784567	37.9982891	great	photo
335	nemo14 /nemo/Q1L5W8R7Y3Z6K1M74.MP	08-05-2025 11:32	23.6792334	37.9985432		video
336	nemo8 /nemo/X8B4N7M1Q3S5WRF92.JPEG	08-05-2025 11:35	23.6789567	37.9987654	great vibes	photo
337	nemo2 /nemo/Z3G6H9L2P5V7XQ48.JPEG	08-05-2025 11:38	23.6791234	37.9983567		photo
338	nemo10 /nemo/S7D4H8K1N9R1WQ125.JPEG	08-05-2025 11:44	23.6796789	37.9986123	perfect weather	photo
339	nemo5 /nemo/M2P6R9T3W7ZLQX41.JPEG	08-05-2025 11:45	23.6785234	37.9984789		photo
340	nemo16 /nemo/F4H7K1L9N6SQW7R3.JPEG	08-05-2025 11:48	23.6792567	37.9981678	beautiful crowd	photo
341	nemo13 /nemo/B8E3G6J2M5R1X247.JPEG	08-05-2025 11:48	23.6788234	37.9987345		photo
342	nemo7 /nemo/Y1U4R7T9P2W3LQX58.JPEG	08-05-2025 11:57	23.6794678	37.9982134	outstanding support	photo
343	nemo3 /nemo/K5N8Q1T4W7ZSR9X1.JPEG	08-05-2025 11:58	23.6786123	37.9985678		photo
344	nemo1 /nemo/V9C2P5H8L3N4W746.JPEG	08-05-2025 12:02	23.6793456	37.9988234		photo
345	nemo8 /nemo/P6S9T2W5Z8R1QX71.MP4	08-05-2025 12:15	23.6787789	37.9983567		video
346	nemo12 /nemo/G4J7M1P5R8V7X23.JPEG	08-05-2025 12:17	23.6795234	37.9986789		photo
347	nemo4 /nemo/A8D1G4J7N2S5WQ795.JPEG	08-05-2025 12:18	23.6784789	37.9981345		photo
348	nemo15 /nemo/X3Z6C9F2L5QW748.JPEG	08-05-2025 12:18	23.6791567	37.9984123		photo
349	nemo11 /nemo/RTU1W4Z8M3P1QX52.JPEG	08-05-2025 12:18	23.6788901	37.9987456		photo
350	nemo6 /nemo/L2O5R8T1W4Z5XQ67.JPEG	08-05-2025 12:22	23.6796345	37.9982678		photo
351	nemo14 /nemo/H9K2N5Q8T1W7R234.JPEG	08-05-2025 12:25	23.6785678	37.9985234		photo
352	nemo9 /nemo/E6S1U7O1R4V7QX89.JPEG	08-05-2025 12:28	23.6792123	37.998567		photo
353	nemo2 /nemo/B4F8J1M5P9S7X226.JPEG	08-05-2025 12:32	23.6789456	37.9981789		photo
354	nemo10 /nemo/Y7C1G4K8N2QW7253.JPEG	08-05-2025 12:35	23.6794567	37.9984123		photo
355	nemo5 /nemo/V3Z7D1H5L8P9QX41.JPEG	08-05-2025 12:38	23.6786234	37.9987345		photo
356	nemo16 /nemo/S9W2F6J1N4R7XZ78.JPEG	08-05-2025 12:42	23.6793789	37.9983456		photo
357	nemo13 /nemo/Q8T8U2V6Z9M1QX15.JPEG	08-05-2025 12:45	23.6787567	37.9986234		photo
358	nemo7 /nemo/K1N4Q7T2W5ZSR8X3.JPEG	08-05-2025 12:48	23.6795123	37.9982567		photo
359	nemo3 /nemo/G8J2M6P9R3V7XZ47.JPEG	08-05-2025 12:52	23.6784567	37.9985789		photo
360	nemo1 /nemo/A4E7H1K5N8SQW792.JPEG	08-05-2025 12:55	23.6791234	37.9988123		photo
361	nemo8 /nemo/X689F3I7L1PQZ54.JPEG	08-05-2025 12:58	23.6788567	37.9981456		photo
362	nemo12 /nemo/R2U5W8Z1T4MLQX37.JPEG	08-05-2025 13:02	23.6796789	37.9984678		photo
363	nemo4 /nemo/D9G3J6M2P5STXZ18.JPEG	08-05-2025 13:05	23.6785234	37.9987234		photo
364	nemo15 /nemo/CSF8K2N6Q9RWT43.JPEG	08-05-2025 13:08	23.6792678	37.9983567		photo
365	nemo11 /nemo/O1S4V7Y2Z6MLQX89.MP4	08-05-2025 13:12	23.6789123	37.9986345		video
366	nemo6 /nemo/H7L1P4R8T3WSQZ52.JPEG	08-05-2025 13:15	23.6794456	37.9982123		photo
367	nemo14 /nemo/B3E6L9L2O5RTXZ74.JPEG	08-05-2025 13:18	23.6786789	37.9985456		photo
368	nemo9 /nemo/Y8C2F5J1M4N7QW67.JPEG	08-05-2025 13:22	23.6793345	37.9988789		photo
369	nemo2 /nemo/V4Z7D1H8K3P5RX91.JPEG	08-05-2025 13:25	23.6787234	37.9981678		photo
370	nemo10 /nemo/S9W3G6L2N5QTXZ48.JPEG	08-05-2025 13:28	23.6795567	37.9984234		photo
371	nemo5 /nemo/Q1L4H7X2Z5MLRW73.JPEG	08-05-2025 13:32	23.6784123	37.9987567		photo
372	nemo16 /nemo/K6N9P3R7T1V5XZ52.JPEG	08-05-2025 13:35	23.6791678	37.9983123		photo
373	nemo13 /nemo/G2J5L8O1S4WQZ96.MP4	08-05-2025 13:38	23.6788234	37.9986456		video
374	nemo7 /nemo/A7E1H4K9N2PRTX35.JPEG	08-05-2025 13:42	23.6796456	37.9982789		photo
375	nemo3 /nemo/X3B6F9J2L5MQWZ48.MP4	08-05-2025 13:45	23.6785789	37.9985234		video
376	nemo1 /nemo/R8U2W5Z9T3PLQX61.JPEG	08-05-2025 13:48	23.6792345	37.9988567		photo
377	nemo8 /nemo/D4G7J1M6P9STXZ24.JPEG	08-05-2025 13:52	23.6789678	37.9981345		photo
378	nemo12 /nemo/C8F2J5L9O3RQW757.JPEG	08-05-2025 13:55	23.6794123	37.9984678		photo
379	nemo4 /nemo/V5C9G6K6N1PSTX82.JPEG	08-05-2025 13:58	23.6786567	37.9987234		photo
380	nemo15 /nemo/Y1Z4D7H2L8N9WZ43.JPEG	08-05-2025 14:02	23.6793789	37.9983456		photo
381	nemo11 /nemo/S6W9T3U7X1MLR258.JPEG	08-05-2025 14:05	23.6787234	37.9986789		photo
382	nemo6 /nemo/Q2T5W8Z2M4PLQX71.JPEG	08-05-2025 14:08	23.6795456	37.9982123		photo
383	nemo14 /nemo/K9N3Q6R1V4STXZ85.JPEG	08-05-2025 14:12	23.6784678	37.9985567		photo
384	nemo9 /nemo/G5J8M2P7S1WQTX39.JPEG	08-05-2025 14:15	23.6791123	37.9988234		photo
385	nemo2 /nemo/A1E4H7K3N9PRTX52.JPEG	08-05-2025 14:18	23.6788567	37.9981789		photo
386	nemo10 /nemo/X7B1F4H8L2OQWZ64.JPEG	08-05-2025 14:22	23.6796234	37.9984345		photo
387	nemo5 /nemo/R3U6W9Z5T2MLQX17.JPEG	08-05-2025 14:25	23.6785345	37.9987678		photo
388	nemo16 /nemo/D8G2J5M1P6STTXZ43.JPEG	08-05-2025 14:28	23.6792789	37.9983234		photo
389	nemo3 /nemo/C4F7K9N3Q2RWTX56.JPEG	08-05-2025 14:32	23.6789456	37.9986567		photo
390	nemo7 /nemo/Y9C3G6J1M5P1QZ82.JPEG	08-05-2025 14:35	23.6794567	37.9982456		photo
391	nemo12 /nemo/V5Z8D2H6L9N5TX29.JPEG	08-05-2025 14:38	23.6786123	37.9985789		photo
392	nemo4 /nemo/S1W4Z7T3U6MQRX45.JPEG	08-05-2025 14:42	23.6793234	37.9988123		photo
393	nemo15 /nemo/Q6T9U3W8Z2P1RX61.JPEG	08-05-2025 14:45	23.6787678	37.9981567		photo
394	nemo11 /nemo/K2N5Q8R4V7STXZ38.JPEG	08-05-2025 14:48	23.6795789	37.9984234		photo
395	nemo6 /nemo/G7J1M4P8S3WQTX54.JPEG	08-05-2025 14:52	23.6784234	37.9987345		photo
396	nemo9 /nemo/A3E6H9K5N2PRTX71.JPEG	08-05-2025 14:55	23.6791567	37.9983678		photo
397	nemo2 /nemo/X8B5F2L6L9OQWZ47.JPEG	08-05-2025 14:58	23.6788234	37.9986123		photo
398	nemo10 /nemo/R4U7W1Z5T8MLQX23.JPEG	08-05-2025 15:02	23.6796345	37.9982789		photo
399	nemo5 /nemo/D9G3J6M2P5STXZ69.JPEG	08-05-2025 15:05	23.6785567	37.9985234		photo
400	nemo16 /nemo/CSF8K1N4Q7RWTZ86.JPEG	08-05-2025 15:08	23.6792123	37.9988456		photo
401	nemo3 /nemo/Y1C4G7J3M6PLQX52.JPEG	08-05-2025 15:12	23.6789789	37.9981234		photo

KPI_SC_01_2

Quality of Experience (QoE): The Quality of Experience was continuously monitored using the Video Quality Probe, which provided real-time Mean Opinion Score (MOS) estimates for each stream. All three streams maintained a MOS value above 3 for the majority of the time, satisfying the required threshold for acceptable user-perceived quality.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	95 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

KPI_SC_01_3

Number of AV Sources: During the race event, a total of six distinct audiovisual sources were integrated into the media pipeline. That is, five mobile phone streams, each connected via 5G networks and capturing individual runners or perspectives; and one 360-degree video camera.

KPI_SC_01_4

Number of VNFs Deployed: The NEMO architecture leveraged a distributed set of Virtual Network Functions (VNFs) deployed across the Proxmox and Kubernetes clusters.

1. Stream Transcoder (3 replicas)
2. Video Quality Probe (3 replicas)
3. Network Probe
4. AI Engine (3 replicas)
5. Information bus
6. RTMP Server
7. Production Core
8. Production Control

KPI_SC_01_5

Content Enrichment: The content delivered to end users was enriched with multiple layers of metadata and quality indicators, including:

1. Bib number detection via AI-based computer vision.
2. MOS scores representing stream quality.
3. Network metrics such as jitter and packet loss.
4. Geolocation context, enabling the positioning of events within the physical layout of the race.

KPI_SC_01_6

Multipath Connectivity for High Bandwidth Content: The inclusion of the Vuze XR 360-degree camera Figure 86 stream represented a significant test for multipath and high-throughput video delivery. The stream required substantial bandwidth to ensure frame fidelity and spatial accuracy. Measurements from the Network Probe confirmed throughput rates exceeding 12 Mbps per stream, meeting the baseline for complex content such as 360° video.

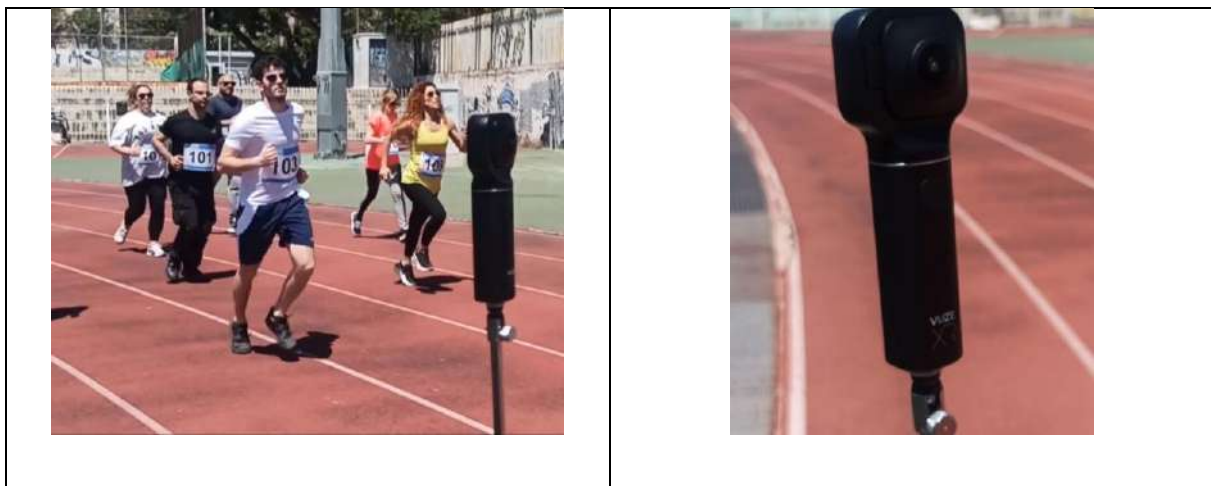


Figure 86 The VuzeXR 360 camera used for streaming large video sources

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	96 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



Figure 88 Egaleo Municipal Stadium, with the positions of the users

5.1.4 Lessons Learned and Replication guidelines

The Smart Media trial within the “Round of Athens Race” provided a valuable opportunity to assess both the technical robustness and operational feasibility of deploying distributed media workflows in real-world conditions. Several lessons emerged from the setup, integration, and validation phases.

- **Automation with CI/CD:**
One of the most impactful enablers of the trial was the use of GitOps-based CI/CD pipelines through FluxCD, which was seamlessly integrated into the NEMO orchestration framework. This setup proved essential in ensuring reliable, repeatable, and fast deployments of the various components involved in the media workflow.

Once the initial cluster configuration was completed, new versions of components could be deployed with minimal manual intervention. Developers simply committed their changes to the project’s GitLab repository, where automated tests validated the code. Upon successful testing, the updated services were automatically deployed to the appropriate Kubernetes environment via FluxCD. This not only reduced the risk of human error but also enabled rapid iterations and efficient debugging cycles throughout the trial.

For replication in other contexts, adopting such declarative and automated DevSecOps practices is strongly recommended. This not only accelerates time-to-deployment but also facilitates traceability, rollback mechanisms, and consistency across deployments.

- **Site Selection and Stakeholder Engagement:**
The process of selecting a suitable environment for live video capture highlighted the importance of early logistical planning and stakeholder coordination. In this case, securing access to the Egaleo Stadium in Athens required direct engagement with local administrative authorities, and the arrangement of permissions for two consecutive days of operation.

This presented a non-technical but critical challenge. For similar trials in the future, particularly those involving real-time content production or public event simulation, it is recommended to:

- Start location scouting and permit applications well in advance.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	98 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

- Engage early with municipal authorities or facility managers.
- Prepare documentation detailing the objectives, security measures, and data handling procedures, to build trust and streamline approval.

The Egaleo setup ultimately provided a controlled yet realistic test environment, with multiple runners positioned across varied scenarios, contributing to a robust evaluation of the system's ability to handle diverse media inputs.

- Remote Production:

Another key insight was the success of the remote production setup between Athens and Madrid. By deploying a dedicated server in Madrid premises and establishing a VPN connection to the OTE infrastructure, it was possible to manage video production remotely, with the control plane and processing tools operating in Madrid while the content was being streamed live from Athens.

This architecture confirmed that remote production can be entirely location-agnostic, with the only prerequisite being secure VPN connectivity to the capture infrastructure. The media content was subsequently broadcast via a public IP, enabling external viewing, evaluation, or even editing, in near-real time.

For replication in similar or commercial-grade environments, this setup demonstrates:

- The value of network-agnostic, VPN-based connectivity models for secure yet flexible remote production.
- The feasibility of centralized management and distributed acquisition, particularly for events spanning multiple cities or countries.
- The opportunity to decouple media creation from media supervision, potentially reducing travel and logistics overhead.

The Smart Media City use case used the Next Generation Meta Operating System (Nemo Meta OS) framework to enhance the boundaries of live media capture and user involvement in live outdoor events, providing an effective broadcast, analysis and productivity solution for European media oriented businesses, that seek to enhance user engagement and satisfaction by offering a Personalized Content Delivery solution with tailored content to user preferences based on AI-driven recommendation/information. The solution will provide flexibility in the usage of available network 5G and computing resources, which can reside in the whole near/far edge – -cloud spectrum to handle the intense media content that is captured. Usually, in a race, a variety of video sources are deployed along the racetrack; the sources can be cameras, Drones, Go Pro cameras attached to runners and, of course, user smartphones. Therefore, to provide real time adaptation and low latency, the meta Network Cluster Controller and Intent-based Migration module of the NEMO MetaOS were used to provide advanced network management to ensure strict latency/bandwidth requirements. Fast time sensitive migration across the continuum for large media was a prominent feature and the underlying cornerstone of this solution.

The Smart Media City Trial features a pronounced contribution to both domestic and international ambitions for live streaming and real-time content combined with personalised content delivery. It provides a solution that validates the meta operating systems such as NEMO not only from a citizen viewpoint for Live sports events featuring cost-effective enhanced coverage, presentation and participation but also presents a sustainable business solution for media channels and the media industry because of its additional environmental sustainability features which consider CO2 emissions for all components and migration procedures such as energy consumption & efficiency and green energy usage in service allocation and delivery.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	99 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

5.2 VR Experience about ancient Workshop of sculptor Phidias

5.2.1 Set up and Integration

The XR_01 use case focused on enhancing educational VR Head Mounted Display experiences. The pilot was hosted at the VR facilities of the Hellenic Cosmos of the Foundation of the Hellenic World. The Hellenic Cosmos is a Cultural Center which is based in Athens and features exhibitions, educational programs and public access to single person systems using Head Mounted Displays. Specifically, the use case enhanced the experience of an VR application that presents everyday life in the workshop of a famous in antiquity sculptor Phidias. The hardware and software setup were described in D5.3[4] which also presented the metrics runs performed. The following table lists the final hardware/software used and where it was positioned in the IoT-Edged-Cloud continuum.

As with many VR experiences user monitoring is performed by visual observation of the stations TV screen and by judging the posture of the user. There is no other way to estimate if the player is uncomfortable or needs any other assistance. Therefore, the major goal of the first trial is to estimate the condition of the user and if needed notify the Museum educator and alter the VR experience in order to assist him.

During the experience an IoT Smart Watch captures and analyses at first instance locally Biometric Data of the user as he interacts and navigate the virtual environment. The data is further processed and then analyzed on a AI/ML node on the edge-cloud continuum to estimate the Emotional status of the individual. This status allows the application to notify museum staff if assistance is required in case of need. The architectural diagram Figure 89 and Table 7 depicted below shows the final software that was used to create the components and its interconnection.

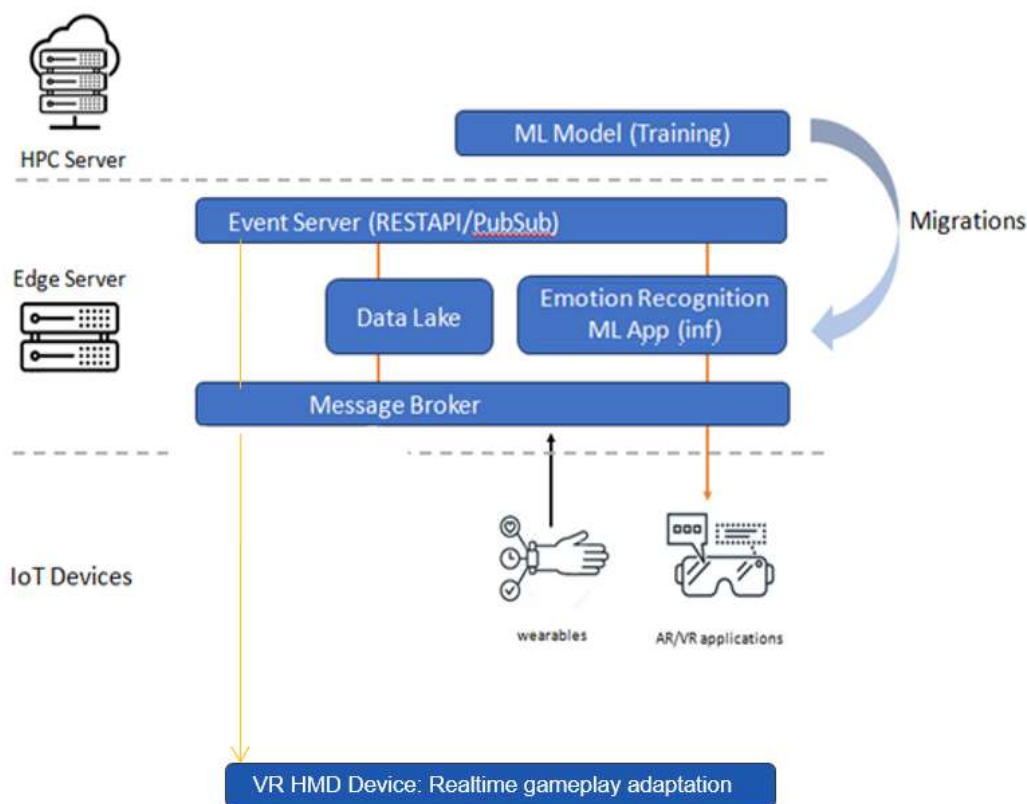


Figure 89 Architectural Component diagram of the XR_01 trial

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	100 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Emotion Recognition ML App (Edge Cloud, Local IoT)	Input data: Smart Watch (Samsung Galaxy 5) App-Tizen OS ML Model: Hybrid CNN-LSTM neural network (AI model was developed from scratch at MAG). Technologies: Tensorflow Dataset: https://www.kaggle.com/datasets/priyankraval/nurse-stress-prediction-wearable-sensors/data
Event Server (Edge)	Input: Inference from ML Applications (HTTP REST API) Technologies: Docker/Kubernetes (Helm), Django 5.0.1
Message Broker (Cloud)	Input: Data from IoT devices (wearables, cameras etc) Technologies: RabbitMQ /Docker/Kubernetes
Data Lake (Edge)	Input: Data from IoT devices Technologies: InfluxDB 2.7.5 / Postgres 16.1.0
VR App (Local IoT)	Unity 3D - VR Application (Meta Quest 2 - Android)
Smart device (Local IoT)	Unity 3D - 2D PC application for Smart TV output

Table 7 The software used for the XR_01 trial

The trial experiences were validated on a controlled group of volunteers that participated in the scenarios as visitors and Hellemic Cosmos staff who participated as HMD operators Museum educators. Feedback was collected using questionnaires and interviews that were conducted and filled out after each use case scenario. Figure 90 below shows VR HMD users experiencing the VR application and wearing the IoT Smart watch during the trial.



Figure 90 VR HMD participants wearing the smart watch during the trial

5.2.2 End-to-End Deployment and Migration Walkthrough

As described in deliverable D5.3[4], in our use case we include two Machine Learning (ML) models for gesture and stress recognition. As a brief reminder, the user uses input devices, like cameras and

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	101 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

wearable (smartwatches), to capture the gestures and vital signs of the wearer, respectively. The Smart XR application receives the devices' data and, with the help of its ML models, it recognizes the gesture performed by the user or the level of stress they are under.

While the users use the devices, the data sent to the Smart XR application are stored in order to retrain the ML models and yield more accurate predictions. For this scenario, we will utilize the NEMO continuum to host part of our application and, more importantly, to run the retraining jobs for the gesture and stress detection ML models.

The following scenario aims to demonstrate:

- The utilization of the resources provided by NEMO
- The mechanics of NEMO that aim to optimize the workload runtime (e.g. migration of a workload in a cluster that can satisfy the computational needs of the workload)

The scenario starts by logging in the LCM UI and creating a workload for the ML retrain part of the application. Specifically, we will retrain the stress detection model with the new data that were collected by smartwatch wearers. We include in the creation form that the workload can in the future apply intents of type "MachineLearning" and "Availability", among the rest of the available intents Figure 91

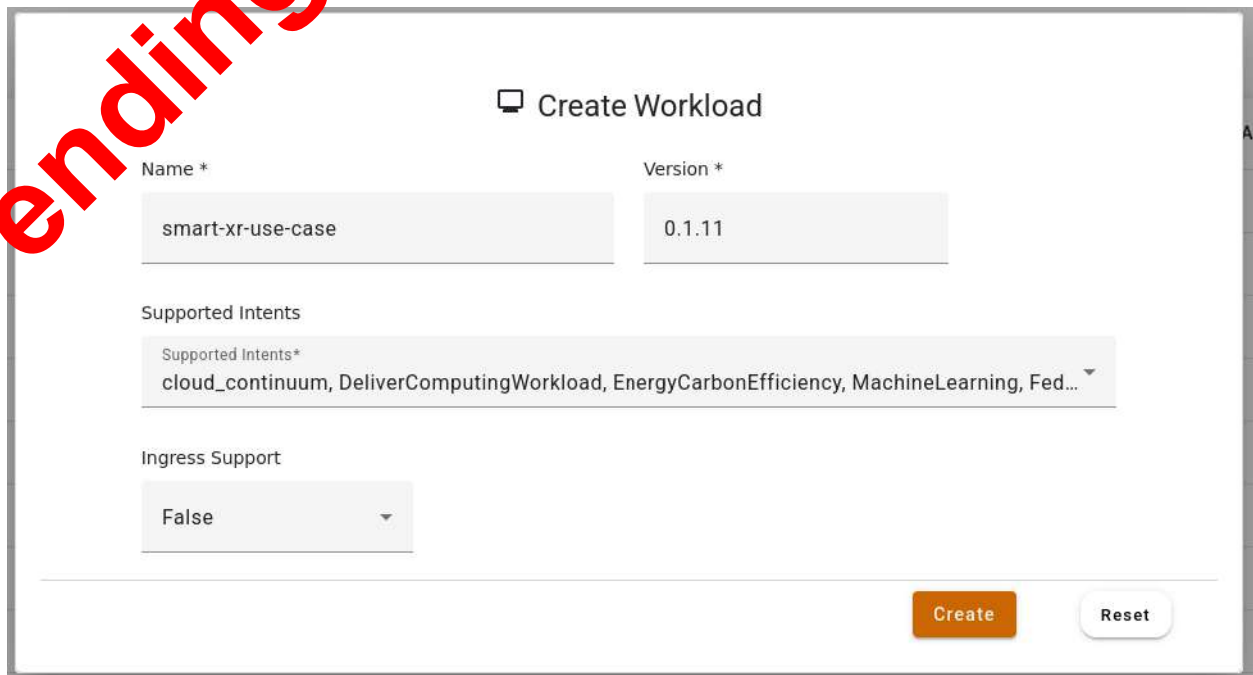


Figure 91 Create workload for Smart XR use case

The workload, now, is in state "Pending" and waits for the workload's Helm chart to be uploaded, as indicated by the relative "Actions" folder button in Figure 92 and Figure 93.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	102 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



Figure 92 Upload workload Helm charts (1)

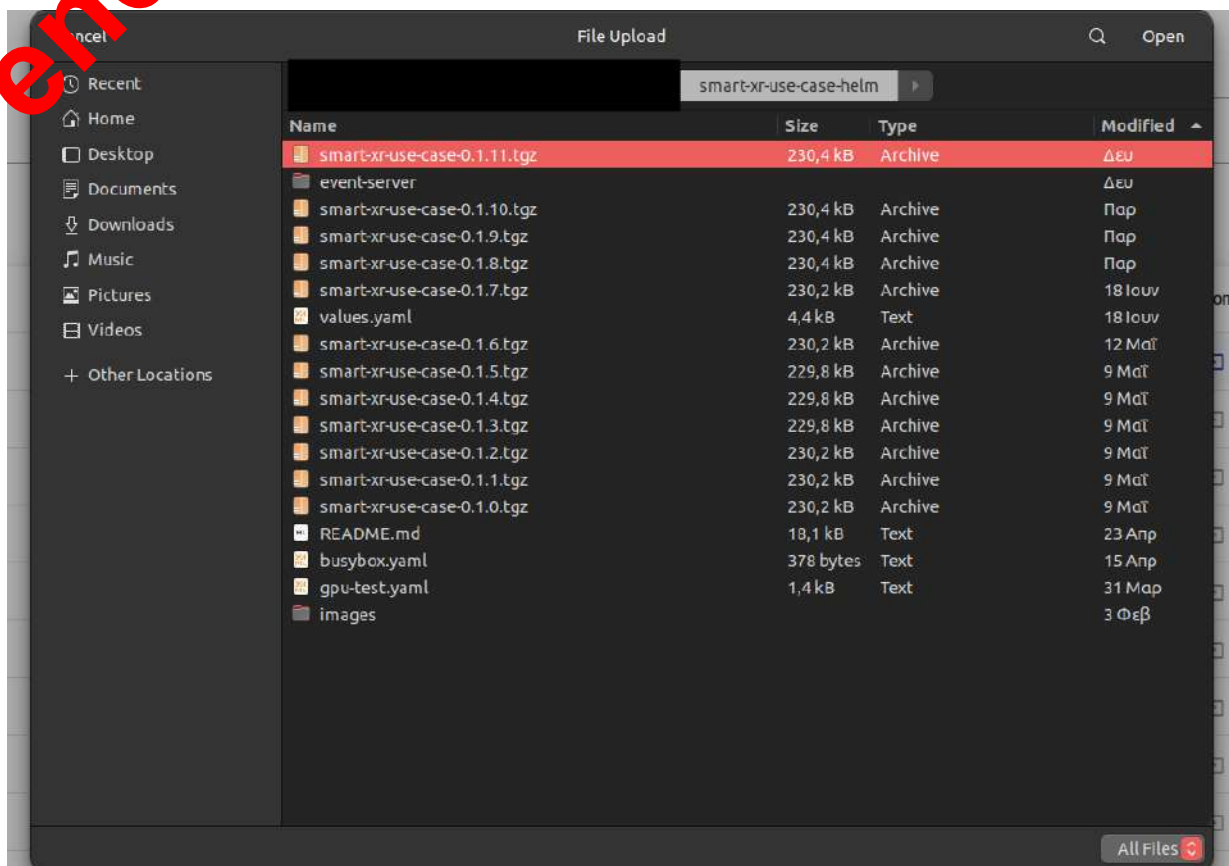


Figure 93 Upload workload Helm charts (2)

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	103 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Once they are uploaded, the workload is set on state “Onboarding” Figure 94, and final “Accepted” Figure 95, once the workload has been registered successfully by the Intent API and the Meta Orchestrator.



Registered Workloads

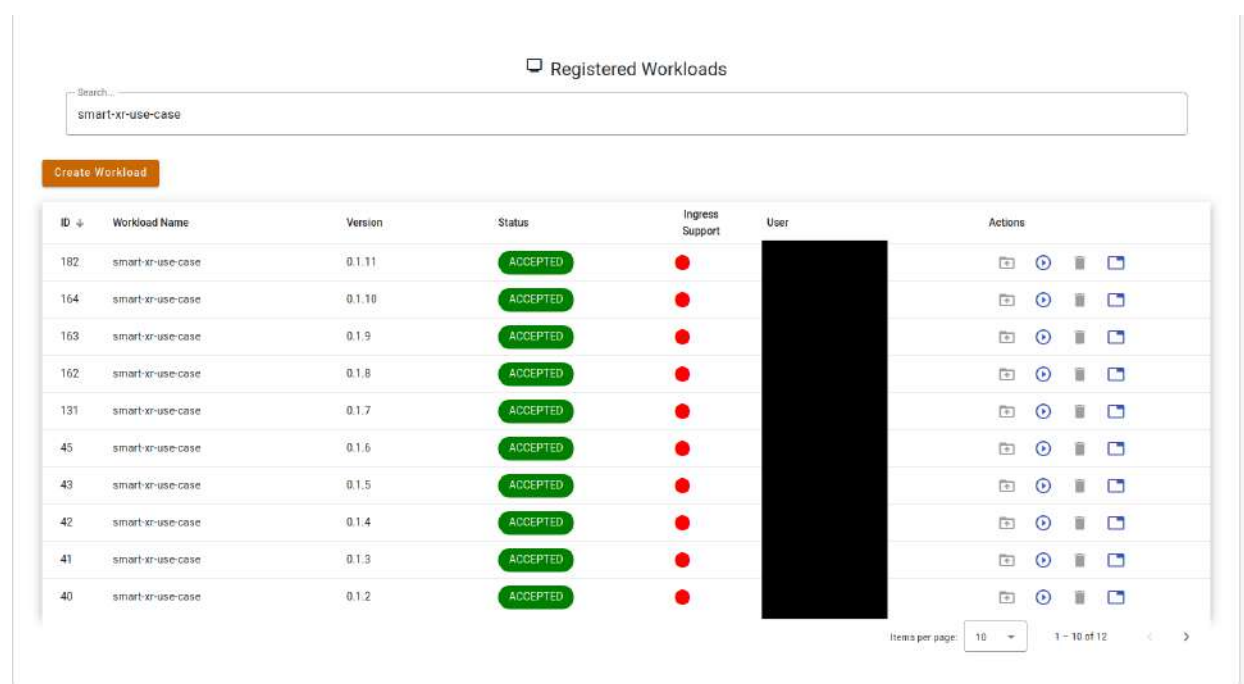
Search: smart-xr-use-case

Create Workload

ID	Workload Name	Version	Status	Ingress Support	User	Actions
182	smart-xr-use-case	0.1.11	Onboarding	●		[+][▶][⏏][📁]
164	smart-xr-use-case	0.1.10	ACCEPTED	●		[+][▶][⏏][📁]
163	smart-xr-use-case	0.1.9	ACCEPTED	●		[+][▶][⏏][📁]
162	smart-xr-use-case	0.1.8	ACCEPTED	●		[+][▶][⏏][📁]
131	smart-xr-use-case	0.1.7	ACCEPTED	●		[+][▶][⏏][📁]
45	smart-xr-use-case	0.1.6	ACCEPTED	●		[+][▶][⏏][📁]
43	smart-xr-use-case	0.1.5	ACCEPTED	●		[+][▶][⏏][📁]
42	smart-xr-use-case	0.1.4	ACCEPTED	●		[+][▶][⏏][📁]
41	smart-xr-use-case	0.1.3	ACCEPTED	●		[+][▶][⏏][📁]
40	smart-xr-use-case	0.1.2	ACCEPTED	●		[+][▶][⏏][📁]

Items per page: 10 1 - 10 of 12

Figure 94 Workload onboarding state



Registered Workloads

Search: smart-xr-use-case

Create Workload

ID	Workload Name	Version	Status	Ingress Support	User	Actions
182	smart-xr-use-case	0.1.11	ACCEPTED	●		[+][▶][⏏][📁]
164	smart-xr-use-case	0.1.10	ACCEPTED	●		[+][▶][⏏][📁]
163	smart-xr-use-case	0.1.9	ACCEPTED	●		[+][▶][⏏][📁]
162	smart-xr-use-case	0.1.8	ACCEPTED	●		[+][▶][⏏][📁]
131	smart-xr-use-case	0.1.7	ACCEPTED	●		[+][▶][⏏][📁]
45	smart-xr-use-case	0.1.6	ACCEPTED	●		[+][▶][⏏][📁]
43	smart-xr-use-case	0.1.5	ACCEPTED	●		[+][▶][⏏][📁]
42	smart-xr-use-case	0.1.4	ACCEPTED	●		[+][▶][⏏][📁]
41	smart-xr-use-case	0.1.3	ACCEPTED	●		[+][▶][⏏][📁]
40	smart-xr-use-case	0.1.2	ACCEPTED	●		[+][▶][⏏][📁]

Items per page: 10 1 - 10 of 12

Figure 95 Workload accepted state

Now, we can create an instance of our workload that will be deployed by the Meta Orchestrator. For demonstration purposes, we will, first, deploy an instance of the model retrain job that will run on CPU.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	104 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Therefore, as shown in Figure 96, we choose a cluster that does not include a GPU (“dev-onelab”) to deploy the workload. We do not create an intent for this instance

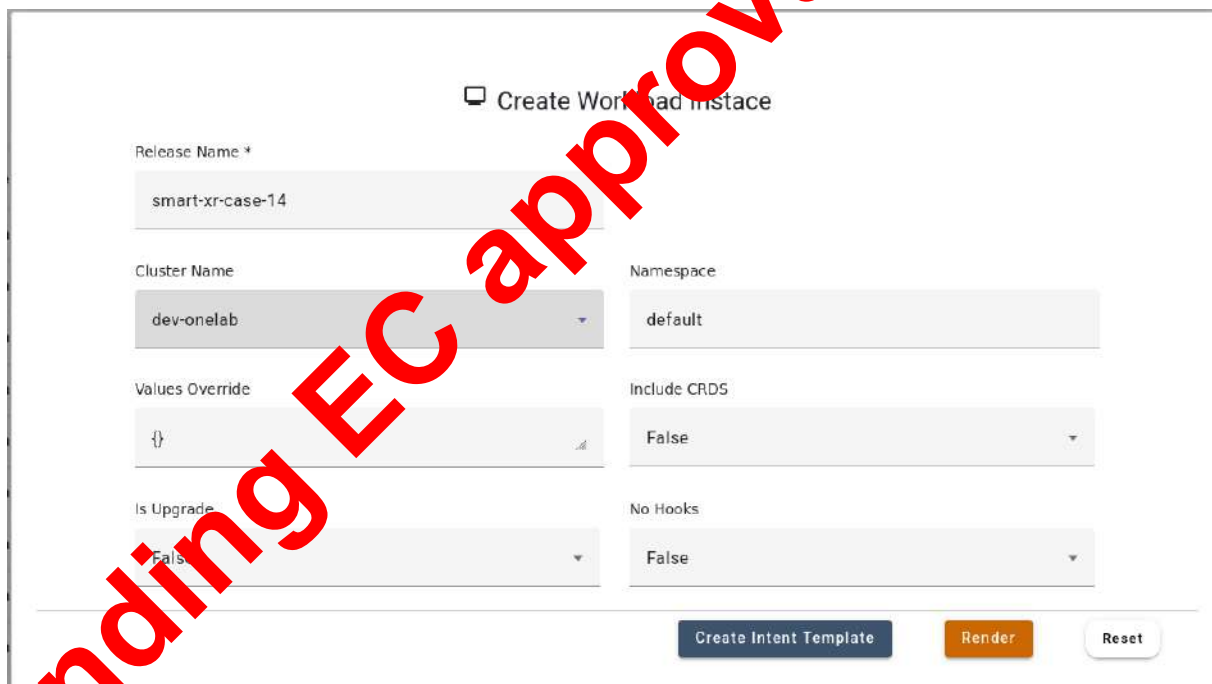
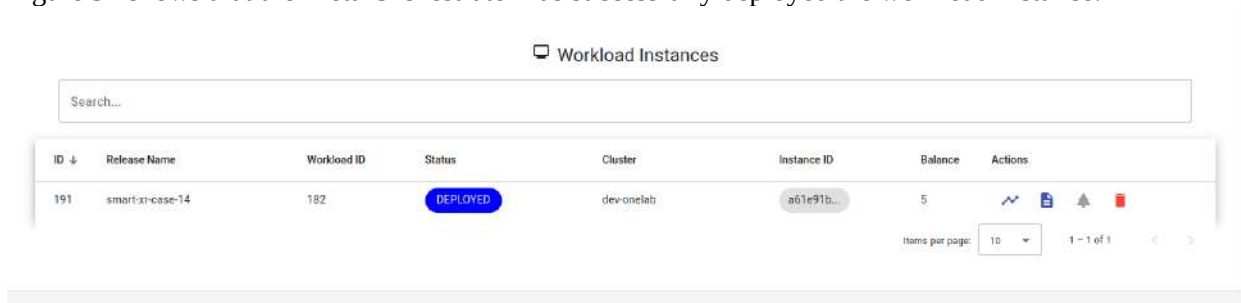


Figure 96 Create workload instance in cluster without GPU

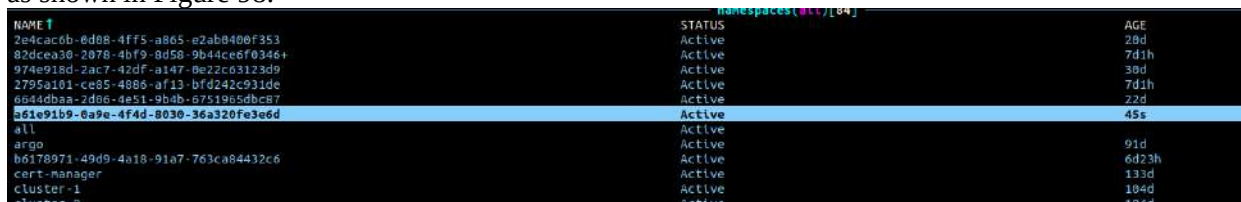
Figure 97 shows that the Meta Orchestrator has successfully deployed the workload instance.



ID	Release Name	Workload ID	Status	Cluster	Instance ID	Balance	Actions
191	smart-xr-case-14	182	DEPLOYED	dev-onelab	a61e91b...	5	[Actions]

Figure 97 Workload instance deployment details

The Meta Orchestrator creates an individual namespace when deploying the workload inside the cluster, as shown in Figure 98.

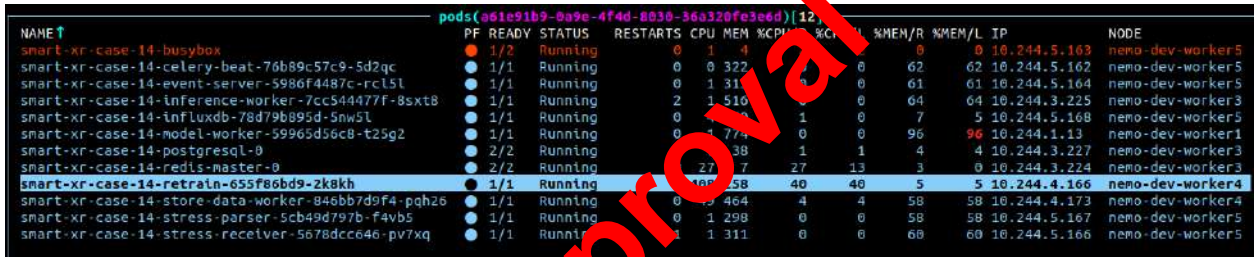


NAME	STATUS	AGE
2e4cac0b-6d88-4ff5-a865-e2ab0400f353	Active	28d
82dcea30-2078-4bf9-8d58-9b44ce6f0346+	Active	7d1h
974e918d-2ac7-42df-a147-0e22c03123d9	Active	39d
2795a101-ce85-4886-af13-bfd242c931de	Active	7d1h
6644dbaa-2d86-de51-9b4b-6751965dbc97	Active	22d
a61e91b9-6a9e-4f4d-8030-36a320fe3e6d	Active	45s
all	Active	
argo	Active	91d
b6178971-49d9-4a18-91a7-763ca84432c6	Active	6d23h
cert-manager	Active	133d
cluster-1	Active	194d
cluster-9	Active	104d

Figure 98 Workload namespace

In Figure 99, we can see that the retrain job has started successfully.

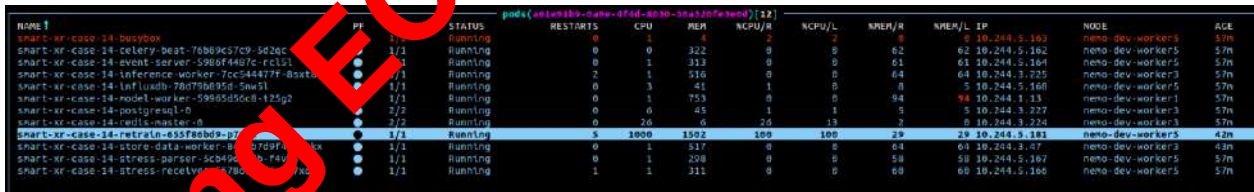
Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	105 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status:
			Final



NAME	PF	READY	STATUS	RESTARTS	CPU	MEM	%CPU	%MEM	IP	NODE
smart-xr-case-14-busybox	1/2	Running	0	1	4	0	0	0	10.244.5.163	nemo-dev-worker5
smart-xr-case-14-celery-beat-76b89c57c9-5d2qc	1/1	Running	0	0	322	0	0	62	10.244.5.162	nemo-dev-worker5
smart-xr-case-14-event-server-5986f4487c-rc15l	1/1	Running	0	1	311	0	0	61	10.244.5.164	nemo-dev-worker5
smart-xr-case-14-inference-worker-7cc544477f-8sxt8	1/1	Running	2	1	510	0	0	64	10.244.3.225	nemo-dev-worker3
smart-xr-case-14-influxdb-78d79b895d-5nw5l	1/1	Running	0	1	714	1	0	7	10.244.5.168	nemo-dev-worker5
smart-xr-case-14-model-worker-59965d56c8-t25g2	1/1	Running	0	1	714	0	0	96	10.244.1.13	nemo-dev-worker1
smart-xr-case-14-postgresql-0	2/2	Running	0	1	298	1	1	4	10.244.3.227	nemo-dev-worker3
smart-xr-case-14-redis-master-0	2/2	Running	0	2	7	27	13	3	10.244.3.224	nemo-dev-worker3
smart-xr-case-14-retrain-655f86bd9-2k8kh	1/1	Running	0	1	458	40	40	5	10.244.4.166	nemo-dev-worker4
smart-xr-case-14-store-data-worker-846bb7d9f4-pqhz6	1/1	Running	0	1	464	4	4	58	10.244.4.173	nemo-dev-worker4
smart-xr-case-14-stress-parser-5cb49d797b-f4vb5	1/1	Running	0	1	298	0	0	58	10.244.5.167	nemo-dev-worker5
smart-xr-case-14-stress-receiver-5678dccc646-pv7xq	1/1	Running	1	1	311	0	0	60	10.244.5.166	nemo-dev-worker5

Figure 99 Worker namespace details

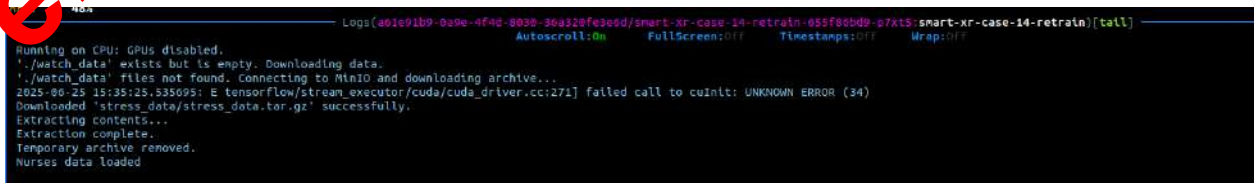
However, we notice that the retrain process does not work as intended, as it is slower than we would like (there were also some restarts to the pod in the process, which caused additional some delays). The CPU usage has exceeded 1 CPU core and keeps rising as the process continues to run.



NAME	PF	READY	STATUS	RESTARTS	CPU	MEM	%CPU	%MEM	IP	NODE	AGE
smart-xr-case-14-busybox	1/2	Running	0	1	4	0	0	0	10.244.5.163	nemo-dev-worker5	57m
smart-xr-case-14-celery-beat-76b89c57c9-5d2qc	1/1	Running	0	0	322	0	0	62	10.244.5.162	nemo-dev-worker5	57m
smart-xr-case-14-event-server-5986f4487c-rc15l	1/1	Running	0	1	313	0	0	61	10.244.5.164	nemo-dev-worker5	57m
smart-xr-case-14-inference-worker-7cc544477f-8sxt8	1/1	Running	2	1	516	0	0	64	10.244.3.225	nemo-dev-worker3	57m
smart-xr-case-14-influxdb-78d79b895d-5nw5l	1/1	Running	0	3	41	1	0	8	10.244.5.160	nemo-dev-worker5	57m
smart-xr-case-14-model-worker-59965d56c8-t25g2	1/1	Running	0	1	753	0	0	94	10.244.1.13	nemo-dev-worker1	57m
smart-xr-case-14-postgresql-0	2/2	Running	0	0	45	1	1	5	10.244.3.227	nemo-dev-worker3	57m
smart-xr-case-14-redis-master-0	2/2	Running	0	2	6	28	13	3	10.244.3.224	nemo-dev-worker3	57m
smart-xr-case-14-retrain-655f86bd9-2k8kh	1/1	Running	0	1	1000	1592	100	29	10.244.5.161	nemo-dev-worker5	42m
smart-xr-case-14-store-data-worker-846bb7d9f4-pqhz6	1/1	Running	0	1	517	0	0	64	10.244.3.17	nemo-dev-worker3	43m
smart-xr-case-14-stress-parser-5cb49d797b-f4vb5	1/1	Running	0	1	298	0	0	58	10.244.5.167	nemo-dev-worker5	57m
smart-xr-case-14-stress-receiver-5678dccc646-pv7xq	1/1	Running	1	1	311	0	0	60	10.244.5.166	nemo-dev-worker5	57m

Figure 100 Rising CPU usage

Figure 100 gives a closer look at the logs of the pod while it runs. However, due to the process taking too long to complete, we will update our deployment to use GPU.



```

Running on CPU: GPUs disabled.
'./watch_data' exists but is empty. Downloading data.
'./watch_data' files not found. Connecting to MinIO and downloading archive...
2025-06-25 19:35:25.335095: E tensorflow/stream_executor/cuda/cuda_driver.cc:271] failed call to cuInit: UNKNOWN ERROR (34)
Downloaded 'stress_data/stress_data.tar.gz' successfully.
Extracting contents...
Extraction complete.
Temporary archive removed.
Nurses data loaded
  
```

Figure 101 Retrain pod logs

We create a new deployment, this time with GPU enabled Figure 102. We choose again the “dev-onelab” for our deployment.

☒ Create Workload Instance

Release Name *

smart-xr-case-15

Cluster Name

dev-onelab

Namespace

default

Values Override

'useGpu': true,

Include CRDS

False

Is Upgrade

False

No Hooks

False

Create Intent Template

Render

Reset

Figure 102 Upgrade workload instance with GPU

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	106 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

For clarity, this is the *values.yaml* that is used to deploy the new workload instance (some sensitive values were removed): (Reference in Annex 2)

If we check the deployed workload instances, we can see that a new workload has been created and deployed successfully Figure 103.

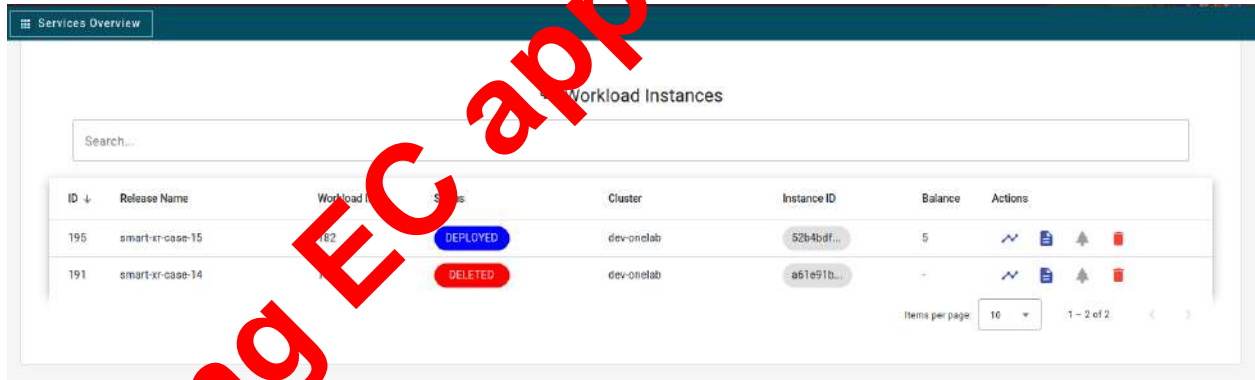


Figure 103 Deployed workload with GPU

However, if we notice the retrain pod Figure 104, we can notice that it cannot start, since there is no GPU available for this cluster Figure 105.

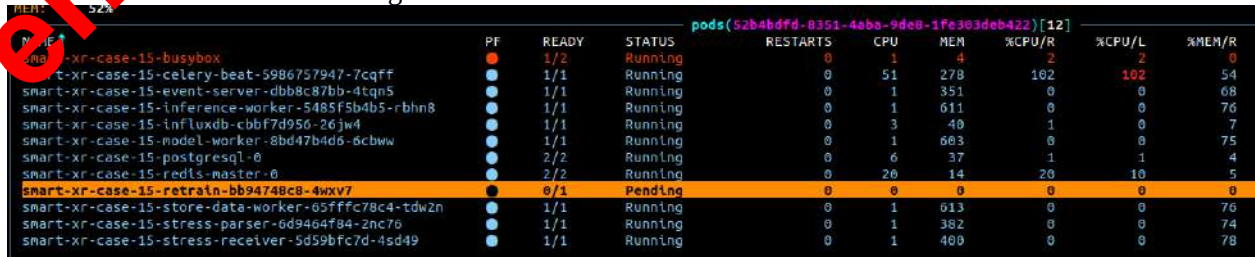


Figure 104 GPU is unavailable for “dev-onelab” cluster (1)

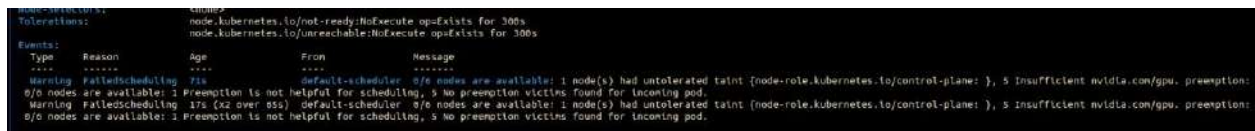


Figure 105 GPU is unavailable for “dev-onelab” cluster (2)

To mitigate this, we can allow the NEMO platform to make the migration, with the use of a “MachineLearning” intent Figure 106. With this intent, we can specify the size of the GPU we require for our application to run (in this case we want to utilize more than 10GBs of the GPU). We are setting the “Target Name” “vram” to be “IS_GREATER_THAN” “10”.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	107 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Create Intents

Use yaml file ☐

Workload Instance ID *
52b4bdfd-8351-4aba-9de8-1fe303deb422

Start Date-time
26/06/2025, 00:00:00

Intent Type *
MachineLearning

End Date-time
31/07/2025, 00:00:00

Target Name
vram

Target Condition
IS_GREAT...

Target Value
10

Create **Reset**

Figure 106 Create MachineLearning Intent

When the intent is fulfilled Figure 107, the Meta Orchestrator component, along with the IBMC component of the NEMO platform, migrate the workload from “dev-onelab” cluster to “pro-onelab”, which fulfills the intent’s condition to run on a GPU. Figure 108 shows the migration of the workload from “dev-onelab” cluster (Status: Terminating) to “prod-onelab” cluster (Status: Active).

Intents Management

Search...

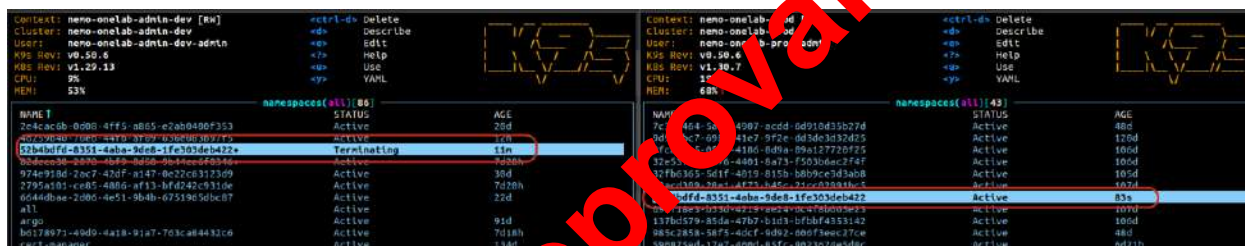
Create Intent

ID	Status	Intent Type	Workload Instance ID	Actions
173	FULFILLED	MachineLearning	52b4bdfd...	  
172	FULFILLED	Availability	f4a9b55...	  
171	FULFILLED	MachineLearning	d862ffc...	  
170	NOT_FULFILLED ACKNOWLEDGED	Availability	d862ffc...	  
169	FULFILLED	Availability	d64c264...	  
168	FULFILLED	Availability	de09688...	  
167	FULFILLED	Availability	b617897...	  
166	FULFILLED	Availability	559f0d3...	  
165	FULFILLED	MachineLearning	95c04be...	  
164	FULFILLED	Availability	95c04be...	  

Items per page: 10 1 - 16 of 173

Figure 107 MachineLearning Intent fulfilled

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	108 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



NAME	STATUS	AGE
2c4cacb-0080-4ff5-ab85-e2ab0400f353	Active	28d
52b4b0fd-8351-4a1a-8d4b-1fe503a0b422	Terminating	11m
974e918d-2ac7-420f-a147-0c27c63123d9	Active	38d
2795a181-c885-4866-af13-bfd242c931de	Active	7d78h
6644d8ae-2080-4e51-9b4b-6751965bcb07	Active	22d
all	Active	91d
argo	Active	7d18h
cert-manager	Active	134d

Figure 108 Migration of workload from “dev-onelab” to “prod-onelab”

The migration action is also shown via the LCM UI, in the instance’s timeline Figure 109

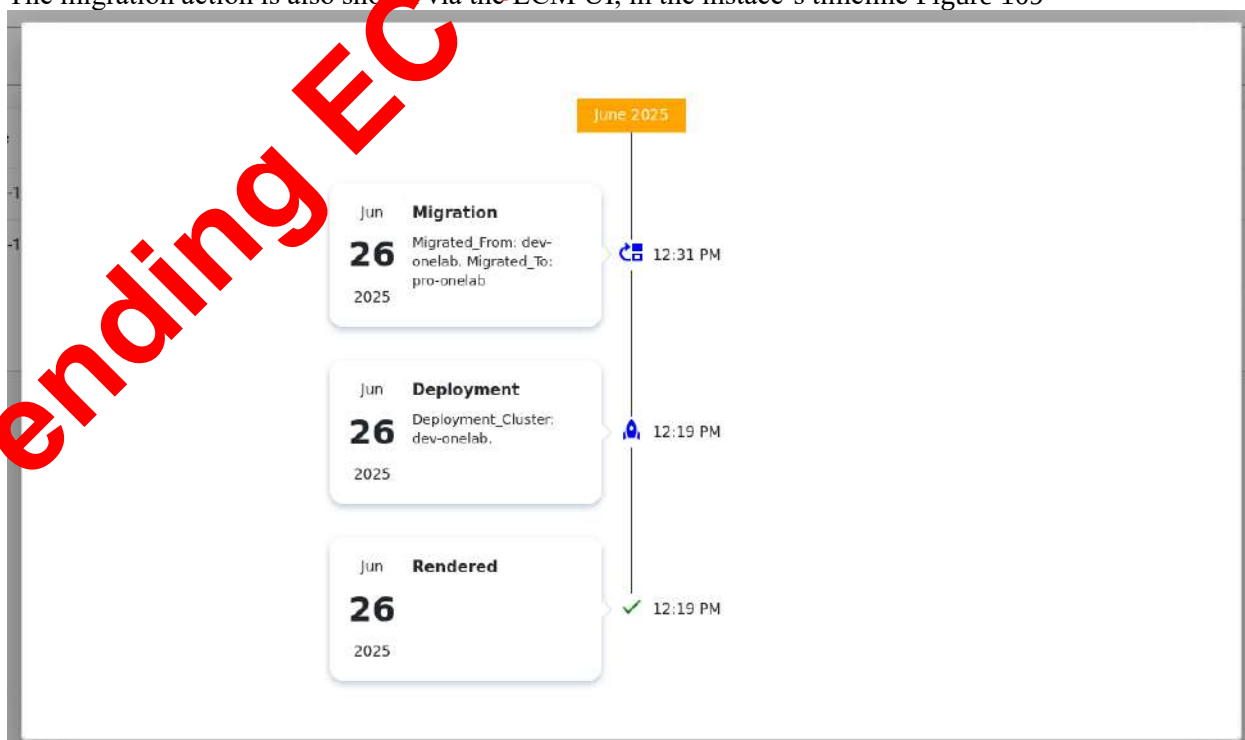
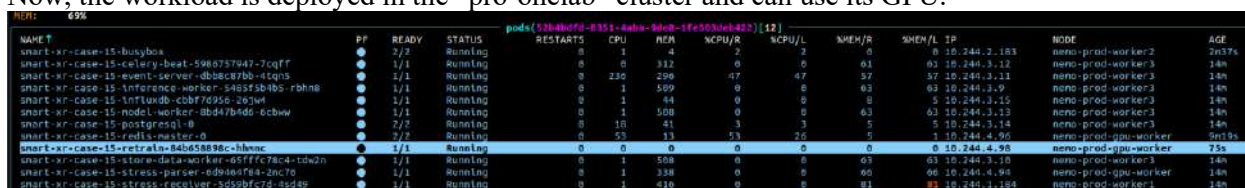


Figure 109 Workload instance timeline

Now, the workload is deployed in the “prod-onelab” cluster and can use its GPU.



NAME	READY	STATUS	RESTARTS	CPU	MEM	NCPU/R	NCPU/L	NMEM/R	NMEM/L	IP	NODE	AGE
smart-xr-case-15-busybox	1/1	Running	0	1	4	2	0	61	61	10.244.2.183	nemo-prod-worker2	2d3Fs
smart-xr-case-15-celery-beat-5986737047-7c9ff	1/1	Running	0	8	312	0	0	57	57	10.244.3.12	nemo-prod-worker3	14h
smart-xr-case-15-event-server-db8c87bb-4tqns	1/1	Running	0	230	290	47	37	63	63	10.244.3.11	nemo-prod-worker3	14h
smart-xr-case-15-inference-worker-5465f3b4b5-rbhm	1/1	Running	0	1	589	0	0	63	63	10.244.3.9	nemo-prod-worker3	14h
smart-xr-case-15-infoword-cbf7695b-263w	1/1	Running	0	1	44	0	0	63	63	10.244.3.15	nemo-prod-worker3	14h
smart-xr-case-15-model-worker-8b447b4d6-bcbw	1/1	Running	0	1	508	0	0	63	63	10.244.3.13	nemo-prod-worker3	14h
smart-xr-case-15-postgresql-0	2/2	Running	0	10	41	3	3	5	5	10.244.3.14	nemo-prod-worker3	14h
smart-xr-case-15-redismaster-0	2/2	Running	0	53	13	53	26	5	5	10.244.4.90	nemo-prod-gpu-worker	9m15s
smart-xr-case-15-retrain-54b5d588c-hhmc	1/1	Running	0	0	0	0	0	0	0	10.244.4.90	nemo-prod-gpu-worker	75s
smart-xr-case-15-stress-dataworker-65ff678c4-tdw2o	1/1	Running	0	1	508	0	0	63	63	10.244.3.10	nemo-prod-worker3	14h
smart-xr-case-15-stress-parser-ed90d76a-2nc7o	1/1	Running	0	1	338	0	0	60	60	10.244.4.94	nemo-prod-gpu-worker	14h
smart-xr-case-15-stress-recorder-5d59bfc7d-nsd4p	1/1	Running	0	1	416	0	0	61	61	10.244.1.184	nemo-prod-worker1	14h

Here, we can make a small note on the capabilities offered by the UI, regarding workload handling. For example, we can delete a workload that has not yet been accepted by the Intent API. For demonstration purposes, we have created a test workload (“test-smart-xr”). While it is in “Pending” state, we can delete the workload as it is shown in Figure 110 and Figure 111.

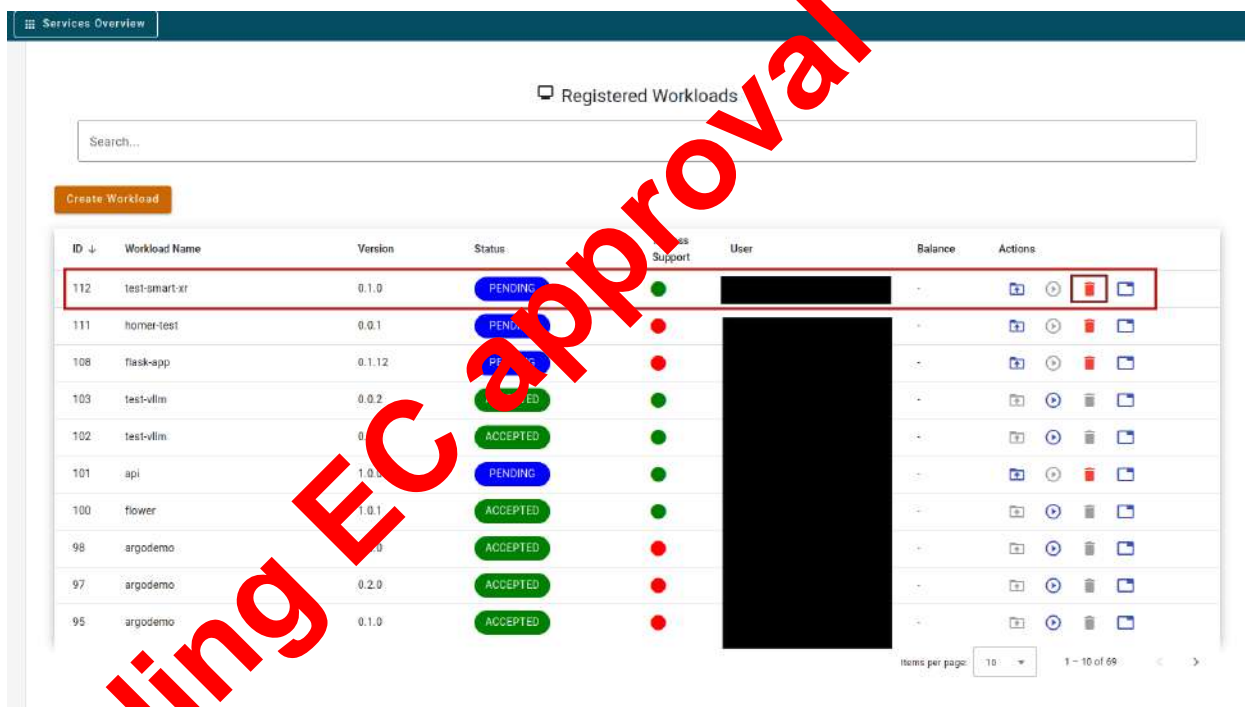


Figure 110 Delete workload (1)

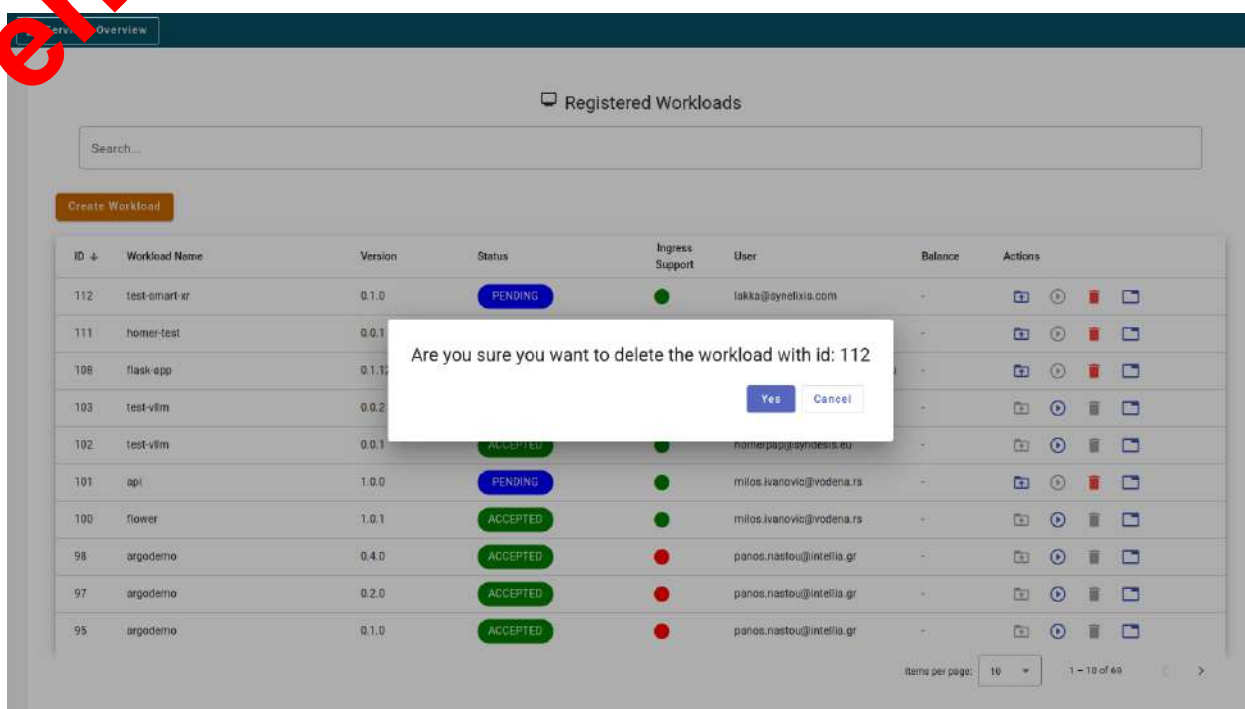


Figure 111 Delete workload (2)

5.2.3 Test scenarios verification

Virtual Reality (VR) can cause discomfort for some users due to a variety of factors. This discomfort is often referred to as VR sickness or cybersickness, and it shares similarities with motion sickness. Furthermore, since the usage of a VR headset isolates the user, it is not easy for the administrator to understand if the user faces any problem.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	110 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

The scenario validated the NEMO meta-OS with Meta Head Mounted Display (HMD) VR devices. The immersive VR experience inside the headsets was successfully enhanced using emotion detection AI/ML functionalities. Thus, the NEMO ecosystem was validated within the area of real-time VR facilities as the ones provided to the public by the Foundation of the Hellenic World at the Hellenic Cosmos Museum. Figure 112 shows an VR user wearing the smart IoT watch and its emotional state being display by a local PC which received messages from the event edge based server.

Scenario: XR01_Test_Scenario_1	
Scenario ID	XR01_Test_Scenario_1
Objective	VR Experience about ancient workshop of sculptor Phidias was enhanced with smart meter biometric data (Accelerometer, Heart Rate, Skin temperature).
Description	The XR trial consisted of heterogeneous Local IoT devices (Wearable and VR headset), collected and analyzed anonymous user biometric data in order to estimate their emotional status during the VR experience. The experience to the user and notified external staff. The specific use case used machine learning algorithms that were executed and trained in the IoT-to-Edge-to-Cloud continuum. The training was performed at intervals as soon as the Data lake which captured all input triggered a migration to High performance Cloud resource.
Features tested	Meta-Orchestrator (Workload deployment) Policy Enforcement (data [image, videos] privacy compliance) Intent-based API (migration) mNCC (micro-slice for sending images) SEE (select secure execution (e.g. ML model inference) at edge, cloud) Plugins Life-Cycle manager and MOCA (for ML services)
Requirements addressed	XR_01.FR01, XR_01.FR02, XR_01.FR03, XR_01.FR04, XR_01.FR05, XR_01.FR06, XR_01.FR07, XR_01_NFR01, XR_01_NFR01, XR_01_NFR03, XR_01.FR04
KPIs	KPI_XR_01.1, KPI_XR_01.2, KPI_XR_01.3, KPI_XR_01.4, KPI_XR_01.5
Prerequisites	The NEMO platform was installed and configured and services were registered., The setup consisted of 1 cluster, 1 IoT wearable device providing biometric measures, 1 VR IoT headset, 1 IoT PC device for info presentation, 1 edge server for ML/AI classification, 1 High Performance Computer in the cloud for training. The ML application for emotional detection was executing on the edge-cloud continuum. At least 1 user service was registered as NEMO consumer and had access to workload LCM information, as well 1 consumer registered as meta-OS Provider.
Test steps	• The Smart XR application owner signed in NEMO as meta-OS consumer. • Upload of the workload descriptor on the NEMO platform. • Deployed the application over the continuum via the NEMO API. • The XR application was started and test scenario 1 was performed. • Monitoring of the workload execution and confirmation that data from the wearables are sent, analyzed, and consumed properly • Retraining of the ML model (this process required the ML micro-service to migrate to High Performance Computer (HPC) infrastructure). • The application owner terminates the execution and collects the logs.
Success state	• Execution of App in the VR headset and consumed successfully events by ML emotional detection services.

Scenario: XR01_Test_Scenario_1	
	- App in VR headset alerts on detection of user's emotional distress event. - Successful migration and training of the ML models in central/edge HPC nodes.
Failure state	There was no failure in any stage. All ML application did not fail to deploy or to detect emotions, VR app and IoT devices had 100% success in receiving status changes and adapt correctly.

Table 10 Test scenarioXR01



Figure 112 VR user with smart watch IoT device and its emotional status indication show on a PC receiving messages form Nemo OS event server.

5.2.4 Machine Learning Methodology for Smart XR: Stress Recognition

Dataset Overview

The Smart XR project adopts a semi-supervised learning strategy to monitor stress levels in real time, enhancing user experiences in virtual reality (VR) environments and other high-pressure contexts. Data is collected using a Samsung Galaxy S6 smartwatch, which captures core physiological and motion-related features: heart rate (HR), accelerometer data (X, Y, Z), and skin temperature (TEMP). These signals provide continuous insight into users' emotional states during VR immersion or daily activities.

The model relies on both labeled and unlabeled datasets. The labeled portion consists of a publicly available dataset originally captured using the Empatica E4 wearable, designed for stress monitoring in hospital nurses[7]. It contains approximately 11.5 million entries across nine features: orientation (X, Y, Z), electrodermal activity (EDA), HR, TEMP, ID, timestamps, and categorical stress labels (low,

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	112 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

medium, high). After the preprocessing, only a small fraction (e.g., 10%) of this data is used as labeled data, while the rest, along with Galaxy smartwatch data and real-time streams from InfluxDB, form the unlabeled pool.

Because the Galaxy smartwatch does not capture EDA, that feature is excluded. The remaining features closely match the smartwatch capabilities, forming a robust foundation for physiological modeling. Notably, InfluxDB integration provides near real-time streaming data, enhancing the model's adaptability and realism in live applications.

Dataset Preprocessing

Effective preprocessing is critical to unify and prepare the various data sources for unified model training. The pipeline handles class imbalance, standardizes inputs, and synchronizes signals across different sensor modalities. Data from CSV smartwatch files, the nurses' dataset, and live InfluxDB streams are processed using a unified script, that automates this entire pipeline.

Handling Class Imbalance

The nurses' dataset exhibits a strong imbalance in class representation, with high-stress labels outnumbering others. To address this, undersampling is used to equalize the sample count across all three stress levels. This ensures balanced training and avoids bias toward overrepresented classes, improving generalization to real-world stress patterns.

Feature Standardization

Features irrelevant to the smartwatch data (EDA, ID) are removed. The remaining five features (X, Y, Z, HR, TEMP) are standardized using a zero-mean, unit-variance scaling. This equalizes feature influence during training, preventing large-value inputs like HR from dominating.

Smartwatch Data Preprocessing

Data collected from the Samsung Galaxy S6 is parsed from CSV files. It includes readings from accelerometer (ACCEL), heart rate (HR), and skin temperature (TEMP) sensors. Each data type is timestamped, often with varying sampling frequencies.

Key preprocessing steps include:

- **Data Cleaning:** Irrelevant fields are dropped (device ID). Timestamps are converted from Unix milliseconds to datetime format.
- **Zero Value Replacement:** HR readings of zero, typically due to sensor dropout, are replaced with the mean of surrounding values within a window.
- **Signal Alignment:** The smartwatch records accelerometer, HR, and temperature data at different frequencies, leading to misaligned timestamps, which are aligned to the closest timestamps. Rows with unmatched timestamps are excluded.
- **Concatenation and Structuring:** Multiple files are merged and sorted chronologically. The final dataset is converted to NumPy arrays with the five selected features.

InfluxDB Integration

The pipeline also supports real-time physiological data ingestion via InfluxDB. When the retraining flag is set, a query is performed to fetch time-series data from the configured device. The same preprocessing logic as with the CSV files is applied, zero handling, timestamp alignment, feature selection, and scaling. The live data is then added to the pool of unlabeled data, enabling dynamic adaptation to current user physiology.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	113 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Model Training

The training pipeline is designed around a three-phase process:

- **Initial Supervised Training:** The model is first trained using the labeled dataset from the nurses' records. This provides a foundational understanding of stress patterns in clean, annotated data.
- **Pseudo-Labeling Iterations:** The model then enters an iterative self-training loop. In each iteration, it predicts stress levels for the unlabeled dataset and selects high-confidence samples (based on a threshold) for annotation. To avoid overfitting and maintain balance, a cap is applied to limit the number of pseudo-labeled samples added per iteration. This strategy increases dataset size gradually while preserving dataset structure and allowing the model to learn from confident unlabeled examples in all phases.
- **Final Training Phase:** After pseudo-labeling concludes, either due to exhaustion of confident samples or a maximum number of iterations, the model is retrained using the full combined set of labeled and confidently pseudo-labeled data. This final model is used for evaluation and deployment.

Architecture and Training Configuration

The model incorporates batch normalization and dropout to enhance training stability and uses a softmax output layer to produce probabilities for low, medium, and high stress levels. Training is performed with a validation split to assess generalization, using a batch size and epoch count defined in a configuration file. Two mechanisms ensure efficient convergence and prevent overfitting: early stopping halts training if validation loss plateaus for several epochs, and a learning rate scheduler reduces the learning rate dynamically when improvement stalls, enabling fine-tuning. All parameters, including dropout rates and learning rate, are externally configured via the configuration file, ensuring flexibility for different training scenarios.

Consolidated Workflow

Both preprocessing and training are orchestrated through a single Python script. The workflow behavior is controlled via two command-line arguments or environment variables:

Retrain: Triggers whether to perform retraining with InfluxDB data.

Use GPU: Enables GPU acceleration, when available.

When retraining is enabled, the script dynamically invokes preprocessing from the preprocess script, passing flags to conditionally load InfluxDB data.

Final model outputs, including weights, evaluation metrics, and confusion matrix plots, are saved to structured output folders. The trained model is also uploaded to a MinIO S3 bucket for later deployment or integration.

Model Evaluation and Results

The stress detection model was evaluated on separate test datasets for the initial training (train) and retraining (retrain) phases, achieving test accuracies of 76% and 77%, respectively, as reported in the classification reports. This slight improvement underscores the effectiveness of the semi-supervised learning strategy, where retraining leverages iterative pseudo-labeling and real-time InfluxDB data to augment the training set with confidently inferred samples, enhancing generalization to unseen physiological patterns. The use of regularization techniques, such as dropout and batch normalization, ensures a robust model capable of dynamic stress detection in virtual reality (VR) and healthcare environments.

As seen in Table 9 the train phase achieved an overall accuracy of 76%, with precision, recall, and F1-scores of 75% across classes, performing best on the medium-stress class (82% F1-score). The retrain

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	114 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

phase improved to 77% accuracy, with a 77% average for precision, recall, and F1-score, showing slight gains in the medium-stress class (83% F1-score) and fewer misclassifications for low-stress instances (75% F1-score vs. 72%). These results can be also observed in the confusion matrices (Figure 113 and Figure 114). This minor enhancement highlights the value of pseudo-labeling and InfluxDB data in refining the model's ability to generalize across diverse stress patterns, particularly in real-time scenarios.



Figure 113 Confusion Matrix for Initial Training

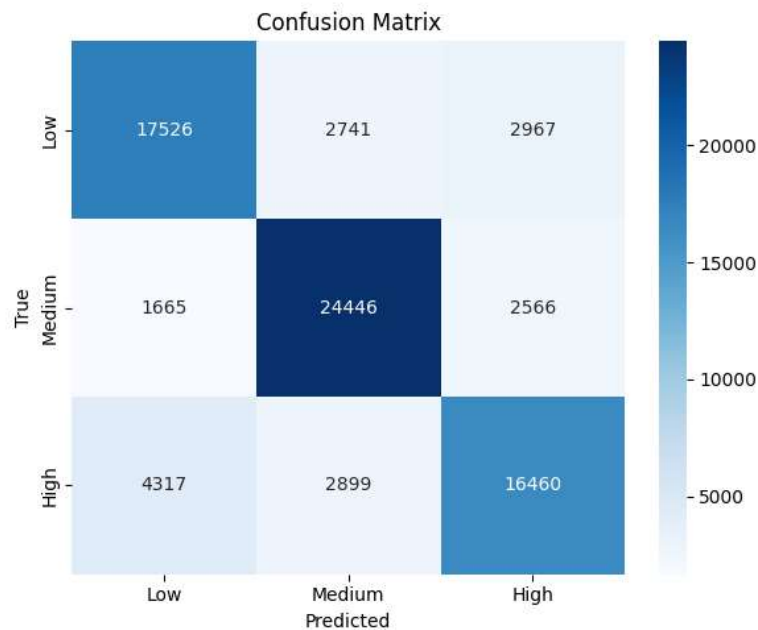


Figure 114 Confusion Matrix for Retraining

	Class	Precision	Recall	F1-Score	Accuracy
Train	Low	0.72	0.72	0.72	0.76
	Medium	0.81	0.82	0.82	
	High	0.73	0.71	0.72	
Retrain	Low	0.75	0.75	0.75	0.77
	Medium	0.81	0.85	0.83	
	High	0.75	0.70	0.72	

Table 9 Ship Detection Evaluation Metrics (Train vs Retrain)

Training and validation accuracy curves (Figure 115 and Figure 116) stable convergence in both phases. The initial training phase exhibits steady learning, while retraining shows slightly tighter convergence, reflecting the benefit of additional pseudo-labeled samples. Early stopping and learning rate adjustments prevented overfitting, with retraining achieving more consistent validation loss reductions, as shown in the loss curves (Figure 117 and Figure 118).

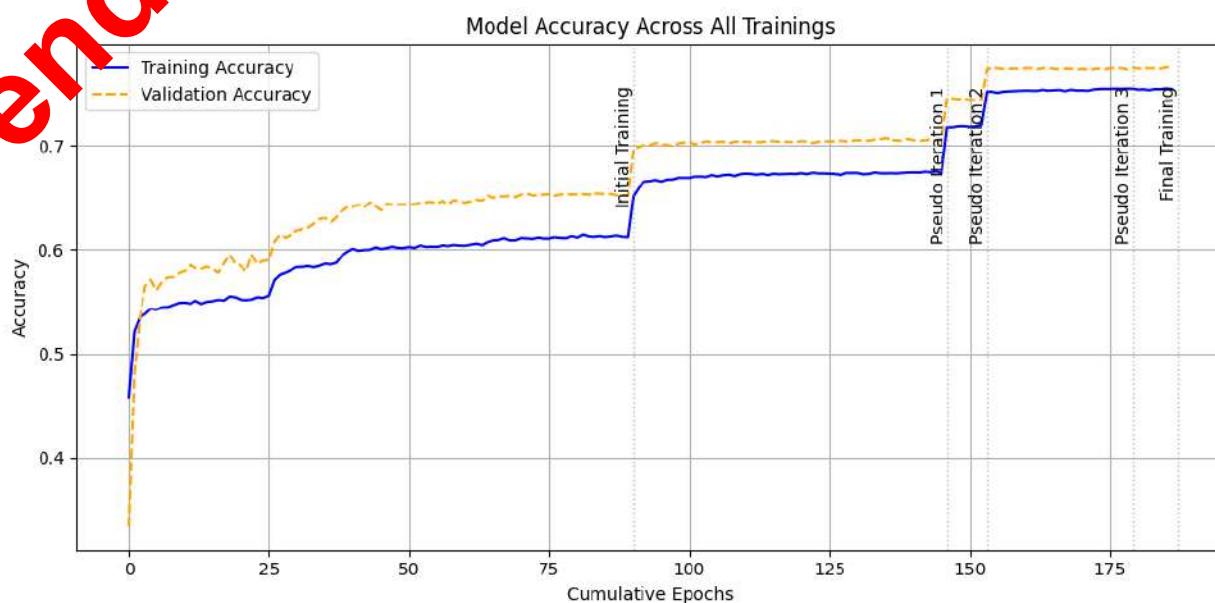


Figure 115 Accuracy Curves for Initial Training

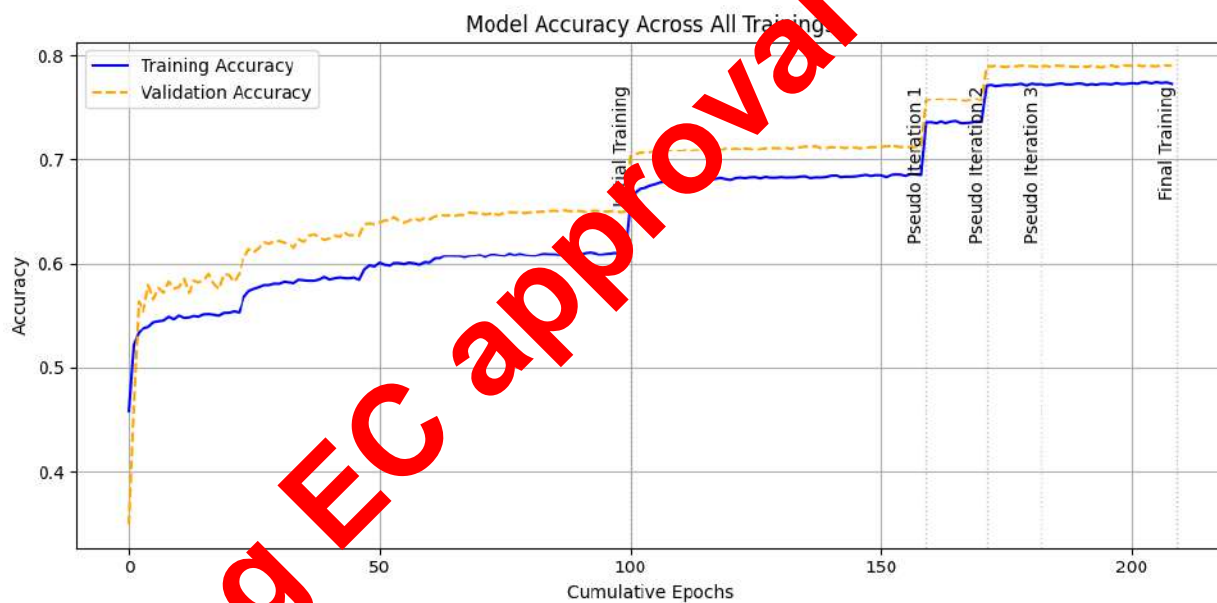


Figure 116 Accuracy Curves for Retraining

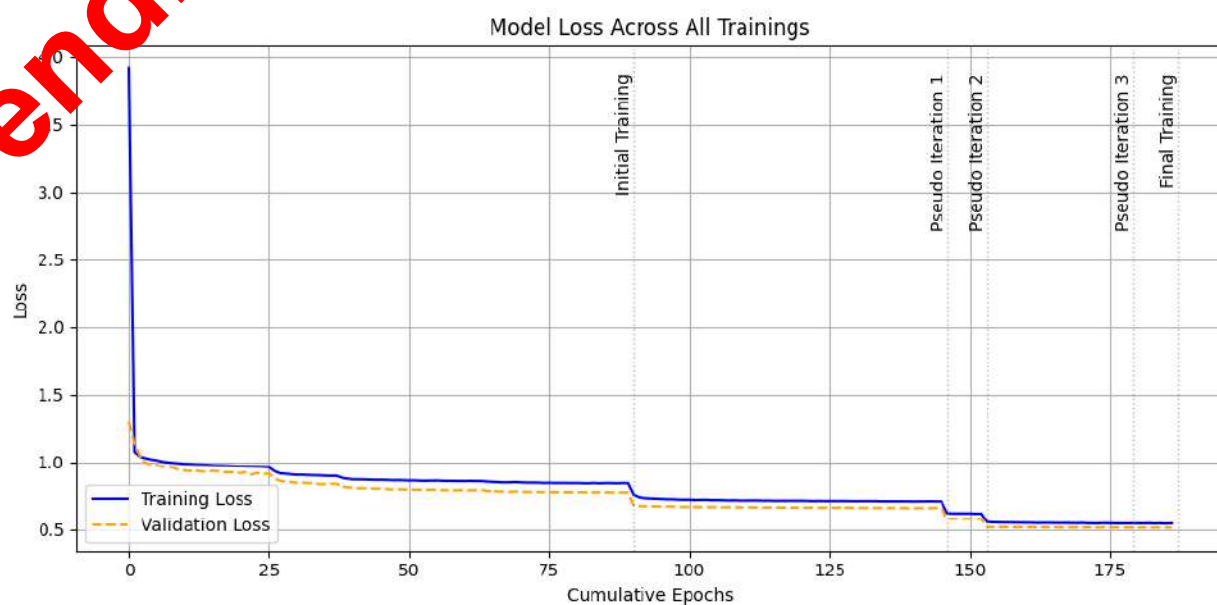


Figure 117 Loss Curves for Initial Training

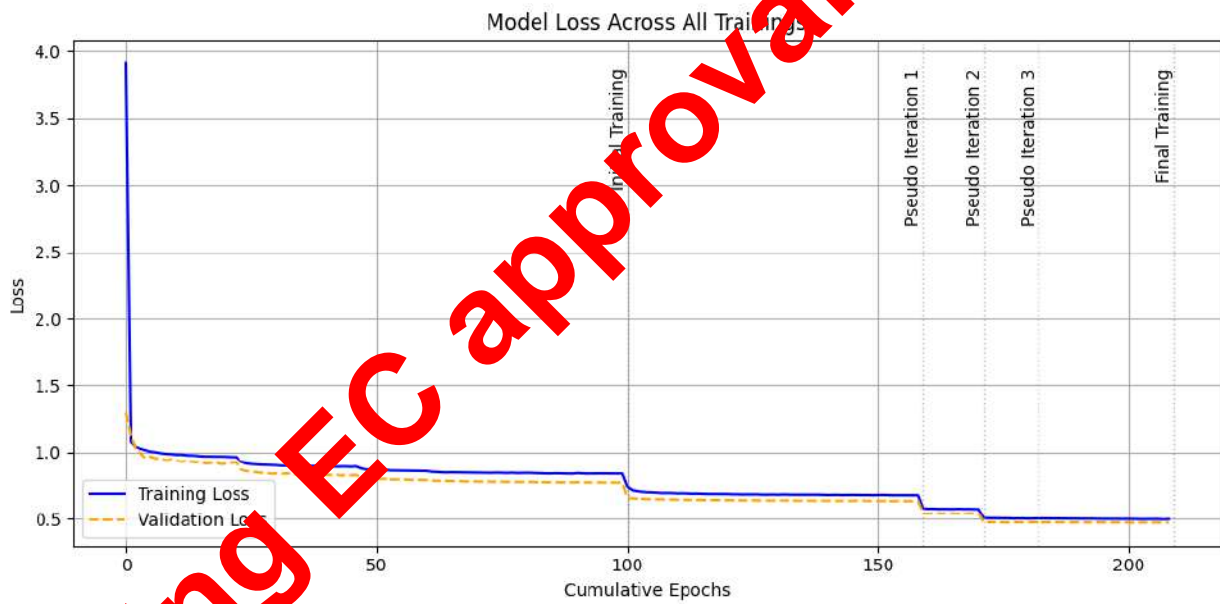


Figure 118 Loss Curves for Retraining

Final evaluation results, including classification reports and confusion matrices, are saved in configurable output paths alongside the trained model in Keras (.h5) format, enabling deployment in edge and cloud environments. The comparison highlights the retraining phase's slight edge, driven by the semi-supervised approach's ability to adapt to real-time physiological data.

Deployment

The project is fully containerized with Docker. A single .env file defines all necessary runtime variables, including:

- MinIO endpoint and credentials
- InfluxDB URL and token
- Data paths and output locations
- Model hyperparameters

The Docker image supports GPU acceleration and can be executed using a simple docker run command. It handles missing data files by downloading them automatically from MinIO. All output files, including trained models, logs, and evaluation plots, are stored in mounted volumes for easy retrieval.

5.2.5 Final validation results and KPIs Evaluation

Table 10 sums the KPI indicators that were established in previous deliverables and described the planned was of their success and testing. The following paragraphs will present how each KPI was validated.

KPI	Description	How it will be tested
KPI_XR_01.1	User status detection < 20 ms Find Emotion status in realtime.	Log analysis of Emotion ML.

KPI	Description	How it will be tested
KPI_XR_01.2	Service trigger events on subscribed VR devices. - 100%, All VR devices receive events no tolerance for losing an event.	Log Analysis Compare logs of ML Emotion components with Unity 3D Apps.
KPI_XR_01.3	Conditional tasks/Micro-service migration. Success rate. - 100%, maintaining Migration from Edge-Cloud.	Log analysis for successful migration.
KPI_XR_01.4	End to end low latency migration to avoid dizziness and motion-sickness < 20ms. Latency between emotion recognition and VR app adaptation. Measure latency.	Log Analysis Compare logs of ML Emotion components with Unity 3D App.
KPI_XR_01.5	Does the user get better experience >=25%	Qualitative user and administrator study to prove better QoE.
KPI_XR_01.6	Multipath connectivity to enable extreme high bandwidth	Not validated. Verified in Smart Media City sibling trial

Table 10 KPIs Evaluation

KPI_XR_01.1

This KPI was successfully validated during the trial. The goal was to achieve real-time emotion status detection with a latency of less than 20 ms. Log analysis and system measurements confirmed that the core motion detection component operated at approximately 20 ms. The complete end-to-end user status detection, from data capture by the IoT watch to the update on the subscribed devices, was consistently under 500 ms. This level of responsiveness is well within the threshold for real-time interaction, ensuring that any interventions by the VR application or notifications to the museum educator were perceived as instantaneous by the user, resulting in a smooth and seamless experience.

KPI_XR_01.2

This KPI handled the reliability of the communication between the event server located on the edge and the registered local IoT devices. The Event server as is shown from Figure 120 in previous pages receives the result of cloud based services about the emotional status of the user. Thus, for the trial and the reliability of the system it is important that any messages that is produced from the ML nodes about the emotional status of the individual, reaches the registered devices. During the trial the HDM users that were experiencing the VR application were monitored by a smart watch IoT device that they were wearing. This device send data to the cloud ML nodes and the emotional status was delivered through a message broker to a Rest API Event server that send it at real time to the registered devices. There is no limit as to how many devices can be subscribed to the event server and how they handle the incoming data. The data was collected anonymously and the only status that registered devices received at real-time for a user was one of these three states, Low – Medium – High Stress.

For the trial two devices were monitoring user status, being subscribed at any moment to the event server. The first is the HMD VR device that runs the application and the second a monitoring PC with a graphical user interface that shows the current status. Both devices were considered local IoT devices. The VR HM device consumed the user status and altered the experience on high Stress levels trying to limit nausea factors, by deploying known practices such as showing a vignette at the perimeter of the

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	119 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

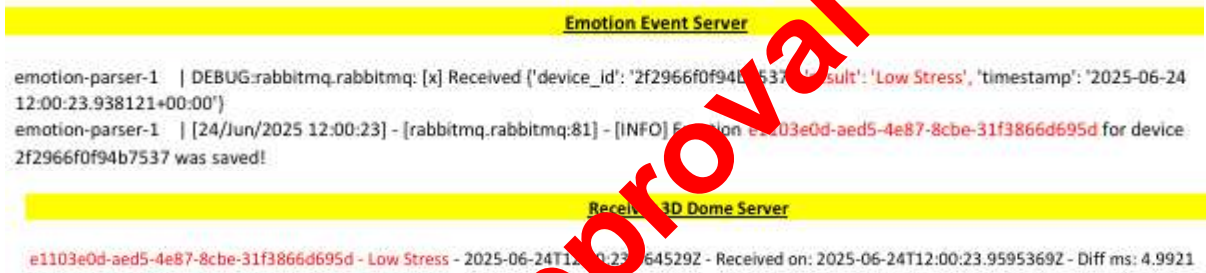


Figure 120 Cutout of the event logs of the Event Server and registered IoT device

KPI_XR_01.3

This KPI, which targeted a 100% success rate for the conditional migration of the ML training micro-service from the edge to the cloud (MPC), was fully met. A thorough analysis of the system logs was conducted following the trial. The logs confirmed that the migration was triggered correctly based on the predefined conditions. The entire migration process completed successfully without any failures or errors, achieving a 100% accuracy rate. This demonstrates the reliability of the NEMO platform's intent-based migration capabilities for computationally intensive tasks like ML model retraining.

KPI_XR_01.4

This KPI has a very strong correlation to the XR_01.2 KPI since it tries to evaluate the same devices and network factors but instead of focusing on stability (100% receiving rate) it tries to highlight the importance of latency. It is true that for realtime applications latency is a problem. This is why for realtime applications a refresh rate of 60Hz is desired (16 ms) and for HMDs even more a refresh rate of 90Hz (13.5 ms) is desirable. The latency value of 20ms (50Hz) requested by this KPI is the time needed since an emotion is recognized by the ML node until it reached the registered IoT device. And although this value is usually requested for local input devices such as trackers and sensors to limit latency and nausea there is no need for it to be so low for indicating the emotional status. The reaction of IoT devices to emotional status in the trial is explained in detail in the XR_01.2 KPI and effectively the latency need not be faster than the framerate of the image generation of the HMD. Therefore, the value of 20ms is actually an ideal situation, because even it is true the user would not notice any difference to the immediate mitigation measure the HMD deploys on high stress. In fact, an interaction can be perceived as instantaneous if the latency is below 70ms. Even latency values of 100ms were interactive enough to signal the HMD user to wait for help and reduce his field of view to limit the perceived nausea.

Furthermore, the reaction time of human museum educators is measured in seconds and not in ms. Nevertheless, as the log compare Figure 121 below shows the targeted latency of 20ms is mostly achieved. The latency was measured by subtracting the emitting time of the emotion message, which is stored in the json package, with the time that the IoT receiver received the message. It is assumed that all devices share the same world clock syncing to an internet based time server. The important factor of success for this and the XR_01.2 KPI is that the network part of the trial of fast and stable enough to provide an essential enhancement of the experience.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	121 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Receiver 3D Dome Server

e1103e0d-aed5-4e87-8cbe-31f3866d695d - Low Stress - 2025-06-24T12:00:23.964529Z - Received on: 2025-06-24T12:00:23.9595369Z - Diff ms: 4.9921

2b85f473-070c-4996-8579-ab25496cfe2 - Low Stress - 2025-06-24T12:00:25.174876Z - Received on: 2025-06-24T12:00:25.1470376Z - Diff ms: 27.8384

224540ba-5bfa-48c3-81e6-0ab6311823ed - Low Stress - 2025-06-24T12:00:27.271188Z - Received on: 2025-06-24T12:00:27.2580916Z - Diff ms: 12.1864

95f0b908-4c8b-49e3-8e50-4e3f12aed76c - Low Stress - 2025-06-24T12:00:28.51191Z - Received on: 2025-06-24T12:00:28.9248962Z - Diff ms: 36.2948

31aef9a7-13ec-4474-91a7-2cbe2328b26 - Low Stress - 2025-06-24T12:00:31.546216Z - Received on: 2025-06-24T12:00:31.5008706Z - Diff ms: 45.3454

2fe4edaf-067d-48c1-b2ce-d1c0a2ba - High Stress - 2025-06-24T12:00:31.897790Z - Received on: 2025-06-24T12:00:31.8835383Z - Diff ms: 14.2517

4b900178-e14d-4b8e-9623-622a - High Stress - 2025-06-24T12:00:32.416335Z - Received on: 2025-06-24T12:00:32.3693227Z - Diff ms: 47.0123

a10c2783-9175-48e1-9143-baca94a1 - Low Stress - 2025-06-24T12:00:34.002938Z - Received on: 2025-06-24T12:00:33.9733366Z - Diff ms: 29.6014

Figure 121 Cutout of the Log entry of the receiving IoT PC and ML node.

KPI_XR_01.5

During the pilot trials a user evaluation was performed drawing comprehensive qualitative and quantitative feedback from museum educators (operators) and visitors, confirming the successful achievement of this KPI and more importantly the successful usage of Nemo mta0OS cloud based services in an educational and museum environment. The next section 5.2.6 provides a full evaluation report outlining the feedback and how the experience was enhanced by a factor of more than 25%. A simple measurement as it is provided for successfully reaching this KPU does not capture fully the increased convenience and reliability the operators felt using cloud based technology without heavy computer resources on site. For the operator a simple smart watch enabled him to monitor the user status, where all the heavy lifting in computing resources was done in the cloud and no equipment other than the IoT watch was visible.

KPI_XR_01.6

The KPI_XR_01.06 specified in D1.1[1], D5.2[3] targeting multipath connectivity using 360 degree video was not verified in this trial as there is no such functionality, it was deprecated from this trial and evaluated in the Smart Media City sibling trial mentioned in the previous section.

5.2.6 Evaluation Report for KPI_XR_01.5

This report details the pilot trial conducted for the Head-Mounted Display (HMD) Experience at the Hellenic Cosmos cultural center on June 24, 2025, as part of the Next Generation Meta Operating System (NEMO) project. The primary objective was to enhance this existing VR experience by integrating NEMO services and AI/ML functionality with user bio-feedback. A key performance assessment indicator (KPI_XR_02.5) for this trial was to improve the experience by at least 25%.

The evaluation, based on comprehensive feedback from museum educators (operators) and visitors, unequivocally confirms the successful achievement of this KPI. The integration of a bio-feedback system revolutionized proactive visitor comfort management, leading to significant improvements in operational efficiency, visitor comfort, and the overall user experience.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	122 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status: Final

This trial involved an existing Head-Mounted Display (HMD) experience. Its primary objective was to enhance this experience by integrating NEMO services and AR/VR functionality with additional user bio-feedback, allowing the experience to adapt to the participant's emotional and physical state.

- The participants were both museum educators (operators) and visitors participated in this trial.
- Visitors wore a Smart Watch device on their hand, designed to identify if they were feeling nauseated during the experience. This smart device was configured to send input data to recognize user discomfort, such as nausea, without storing any personal data.
- Museum educators (operators) received notifications from this smart watch device. These notifications were intended to inform operators quickly if visitors were experiencing discomfort or nausea, enabling them to react faster and provide timely assistance. This proactive approach aimed to help visitors recover and continue the experience, potentially preventing them from having to abandon it.

It is crucial to emphasize that, throughout the HMD experience and the entire evaluation process, no personal data was stored in any way from the smart watch device. While the smartwatch provided input on the user's condition to detect potential discomfort such as nausea, its function was solely to transmit this data for real-time operator notification and assistance, without retaining any biometric or physiological information. The system was designed to allow operators to react quickly and assist visitors based on these real-time inputs. The only data stored securely for future academic research and publication were the completed questionnaires.

Post-Trial Evaluation Process:

After the completion of each trial, participants were provided with questionnaires to gather their feedback.

- For the HMD experience, separate questionnaires were completed by museum educators/operators and visitors.
- Following the completion of the questionnaires, interviews were conducted with all involved participants (separately, one by one). During these interviews, they verbally elaborated on their questionnaire responses and shared their overall impressions and insights from the experience. The data from these questionnaires and interviews, including aggregated numerical responses and detailed qualitative comments, formed the basis for the subsequent evaluation and analysis.

5.2.6.1 Statistical Representation of Answers

This section provides a quantitative overview of the responses gathered from the questionnaires, specifically focusing on the data from the "NEMO Questionnaire for operators HMD." These results reflect the operators' perceptions of the HMD experience and the effectiveness of the NEMO-integrated features.

5.2.6.1.1 NEMO Questionnaire for operators HMD

Question 2: Did you receive any notifications or assistance during the VR experience?

- YES: 10 checked this
- NO: 0 checked this

Explanation/Some of the Comments: Operators confirmed receiving notifications when users felt anxious, dizzy, or nauseated, allowing them to respond and help the visitor recover and continue the experience instead of abandoning it. One operator also stated, "I was notified when the user/visitor felt any discomfort". Another comment noted, "Yes the users felt dizzy, confused or nauseated".

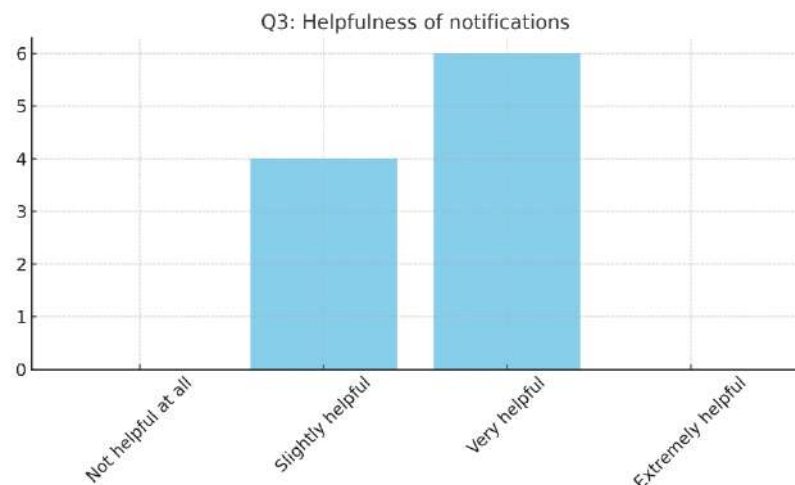
Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	123 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status: Final



Question 3: How helpful were the notifications on operating the experience and managing visitors?

- Not helpful at all: 0 checked this
- Slightly helpful: 4 checked this
- Very helpful: 6 checked this
- Extremely helpful: 0 checked this

Explanation/ Some of the Comments: Operators found the notifications helpful for managing visitors more efficiently. They explained that being notified of discomfort allowed for quicker and more effective assistance. Specific comments included: "Having been notified when the HMD user feels uncomfortable made the operation very helpful but also more efficient to manage visitors since the few seconds or minutes you gain so as to help them when feeling nausea made our work easier and more effective", "I was notified very quickly and as a result I was able to be by the user's side at once", and "I could understand the status of the user".

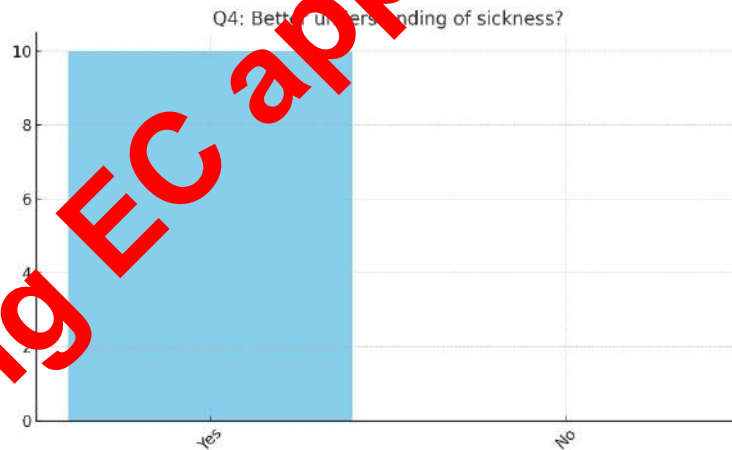


Question 4: Do you think that you had a better understanding about visitor sickness state?

- YES: 10 checked this
- NO: 0 checked this

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	124 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

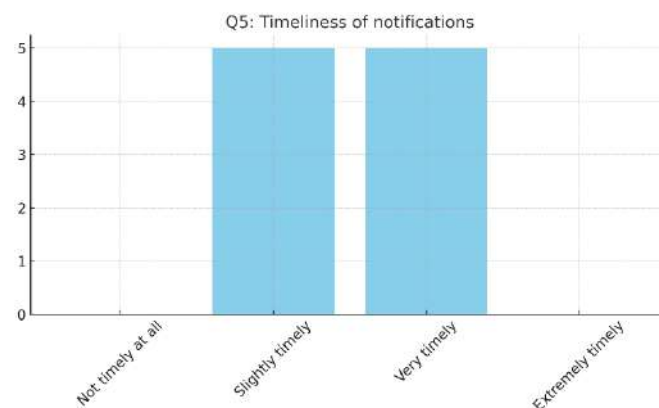
Explanation/ Some of the Comments: The notifications enabled operators to understand the user's condition at an early stage. Operators noted, "When you get the notification you can assist the user in early stage, before the user/visitor becomes confused or dizzy and more nauseated so you can help them recover and instead of quitting the experience to continue with the experience". Other comments reinforced this, stating, "The system notified me of the user/visitor's discomfort way sooner than any visible signs" and "Much better".



Question 5: How timely were the notifications about visitors' comfort levels?

- Not timely at all: 0 checked this
- Slightly timely: 5 checked this
- Very timely: 5 checked this
- Extremely timely: 0 checked this

Explanation/ Some of the Comments: The notifications were perceived as timely, often arriving before significant discomfort manifested. Operators remarked: "The notifications were sent before the user was really nauseated or confused so we were able to assist them. That leads to less visitors stopping the experience due to dizziness", "They were sent quite soon as the visitor felt any discomfort", and "Very timely and real-time".



Question 6: How easy was it to use the application to assist visitors?

- Not easy at all: 0 checked this
- Slightly easy: 1 checked this

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	125 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

- Very easy: 5 checked this
- Extremely easy: 4 checked this

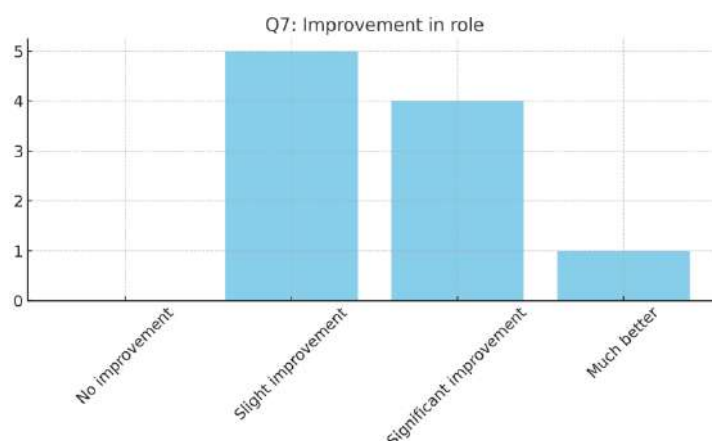
Explanation/ Some of the Comments: The application was generally found to be very easy to use, primarily requiring operators to monitor for notifications. Comments included: "Just had to wait and see if they're going to be a notification. Extremely easy", "Super easy. You just wait for a notification", and "Just run the application".



Question 7: How much better were the notifications in improving your role?

- No improvement: 0 checked this
- Slight improvement: 5 checked this
- Significant improvement: 4 checked this
- Much better: 1 checked this

Explanation/Comments: Notifications led to a noticeable improvement in the operators' role, enabling preventive action and better oversight. Operators shared: "I was able to act preventively", "It gave us more time to react to our visitor's discomfort, dizziness or nauseous", and "I had better oversight".



Question 8: Overall, how much better was in your opinion the user experience with the application compared to without it?

- No improvement: 0 checked this
- Slight improvement: 5 checked this

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	126 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

- Significant improvement: 5 checked this
- Much better: 0 checked this

Explanation/Comments: The overall user experience with the application was considered better, benefiting both operators and visitors. Comments highlighted that the ability to assist users before intense symptoms meant fewer visitors stopping the experience, and operators spent less time per user. Additional comments included: "The experience was much better for both the operator and the user. The ability to assist a user before symptoms of dizziness or confusion become intense means that fewer visitors stop the experience, as they manage to overcome it and operators spend less time with each user, since it's easier to help someone to recover than to stop the experience and have to assist them with equipment or escort them out of the room while there are other users in the room", "It was better, faster, simple and you also feel more at ease", and "It is a great to have feature".



5.2.6.1.2 NEMO Questionnaire for visitors HMD

This section continues the quantitative overview of responses from the questionnaires, specifically focusing on the data from the "NEMO Questionnaire for visitors HMD". These results reflect the visitors' perceptions of their experience, comfort levels, and the assistance received during the HMD trial.

Question 2: Have you tried Virtual Reality (VR) using a headset before?

- Never: 0 checked this
- 1-2 times: 5 checked this
- Many times: 9 checked this
- Other: 0 checked this

Summary: A significant majority of visitors (14 out of 14 respondents) had previous experience with VR headsets. Specifically, 9 visitors had tried VR "Many times," and 5 had tried it "1-2 times".

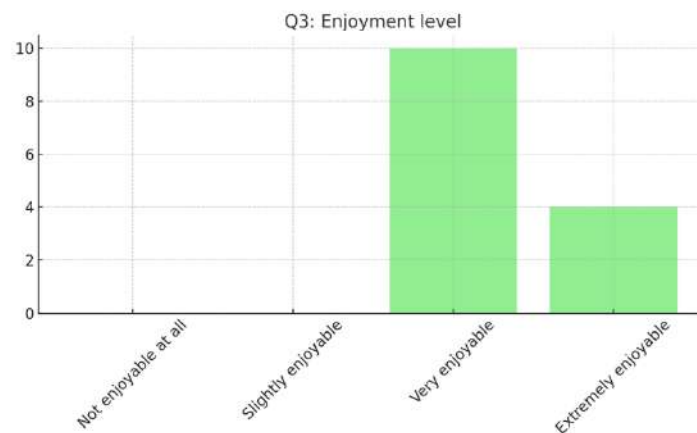
Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	127 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



Question 3: How enjoyable was the VR experience?

- Not enjoyable at all: 0 checked this
- Slightly enjoyable: 0 checked this
- Very enjoyable: 10 checked this
- Extremely enjoyable: 4 checked this

Summary: All visitors found the VR experience to be enjoyable. 10 checked "Very enjoyable" and 4 checked "Extremely enjoyable", indicating a highly positive reception of the experience.



Question 4: Would you participate in such an experience again?

- Never: 0 checked this
- Maybe: 0 checked this
- Definitely: 14 checked this

Summary: All 14 visitors indicated that they would "Definitely" participate in such an experience again, demonstrating strong satisfaction and willingness to re-engage.

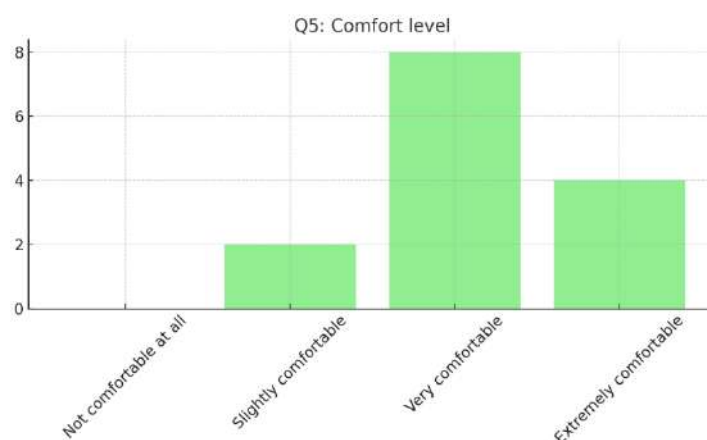
Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	128 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



Question 5: How comfortable did you feel during the experience?

- Not comfortable at all: 0 checked this
- Slightly comfortable: 2 checked this
- Very comfortable: 8 checked this
- Extremely comfortable: 4 checked this

Summary: Most visitors felt comfortable during the experience. 8 found it "Very comfortable", and 4 found it "Extremely comfortable". Only 2 visitors reported feeling "Slightly comfortable".



Question 6: How timely did you receive any assistance during the VR experience?

- Not timely at all: 0 checked this
- Slightly timely: 4 checked this
- Very timely: 8 checked this
- Extremely timely: 2 checked this

Summary: Assistance received by visitors was generally perceived as timely. 8 indicated "Very timely" and 2 indicated "Extremely timely". 4 visitors felt the assistance was "Slightly timely".

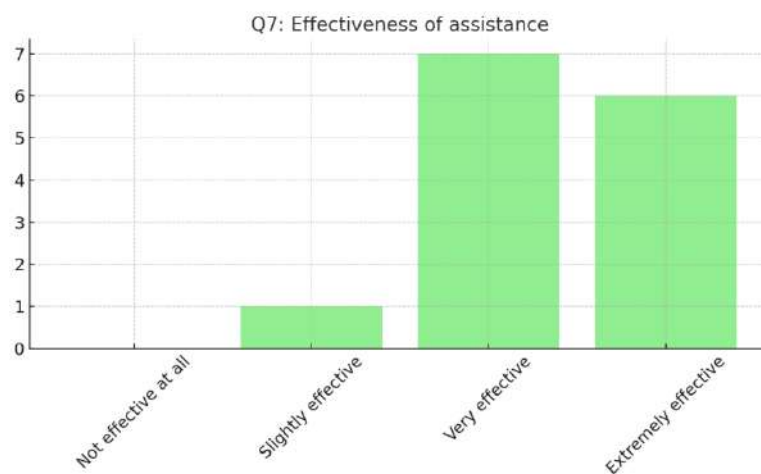
Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	129 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



Question 7: How effective was the assistance in improving your experience?

- Not effective at all: 0 checked this
- Slightly effective: 1 checked this
- Very effective: 7 checked this
- Extremely effective: 6 checked this

Summary: The assistance provided was largely seen as effective in enhancing the experience. 7 visitors found it "very effective", and 6 found it "Extremely effective". Only 1 visitor indicated it was "Slightly effective".



Question 8: If you felt nauseated, did you have to inform the operator? (Yes / No)

- YES: 0 checked this
- NO: 14 checked this

Summary: All 14 visitors confirmed that they did not have to actively inform the operator if they felt nauseated. This suggests that the system's automated notifications to operators were functioning effectively in detecting discomfort without requiring explicit user communication.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	130 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final



5.2.6.2 Qualitative Thematic Analysis of Comments

This analysis extracted detailed feedback from the "Please explain" sections of the questionnaires and incorporates verbal insights from interviews, providing a rich qualitative perspective on the user experience and operational improvements.

5.2.6.2.1 NEMO Questionnaire for operators HMD

The qualitative feedback from operators using the HMD system highlights significant improvements in visitor management and operational efficiency due to the system's notification features.

Timely Detection and Prevention of Discomfort (Questions 2, 3, 4, 5, 7)

- Operators received notifications when users felt anxious, dizzy, or nauseated, enabling them to respond and help visitors recover and continue the experience, preventing conditions from worsening or leading to abandonment. They were notified "when the user/visitor felt any discomfort".
- The notifications were "very helpful" and made operations "more efficient to manage visitors" because the "few seconds or minutes gained" allowed for easier and more effective work in assisting those feeling "nausea".
- One operator noted, "I was notified very quickly and as a result I was able to be by the user's side at once" and could "understand the status of the user".
- Operators believed they had a "better understanding about visitor sickness state" because the notifications allowed them to "assist the user in early stage, before the user/visitor becomes confused or dizzy and more nauseated". This meant they could help visitors recover and continue rather than quitting. The system notified them of discomfort "way sooner than any visible signs".
- Notifications were perceived as "very timely" and "real-time", sent "before the user was really nauseated or confused," which led to "less visitors stopping the experience due to dizziness".
- This proactive capability "gave us more time to react to our visitor's discomfort, dizziness or nauseous", enabling operators to "act preventively" and have "better oversight".

Ease of Use and Overall Improvement (Questions 6, 8)

- The application for assisting visitors was considered "Extremely easy" and "Super easy" to use, with operators simply needing to "wait and see if there going to be a notification" and "Just run the application".

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	131 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

- Overall, the user experience with the application was seen as "Much better" and a "Significant improvement". This was because the "ability to assist a user before symptoms of dizziness or confusion become intense means that fewer visitors stop the experience".
- Operators reported that they "spend less time with each user" because "it's easier to help someone to recover than to stop the experience and have to assist them with equipment or escort them out of the room while there are other users in the room". The experience was described as "better, faster, simple and you also feel more at ease", highlighting it as a "great to have feature".

5.2.6.2.2 NEMO Questionnaire for visitors HMD

The visitor feedback largely focused on the enjoyment and comfort of the VR experience and the effectiveness of the assistance received, though no specific open-ended comments were provided in the source for questions regarding timely or effective assistance or requiring them to inform the operator about nausea.

High Enjoyment and Willingness to Re-engage (Questions 3, 4)

- All visitors found the VR experience to be "Very enjoyable" or "Extremely enjoyable".
- All 14 visitors indicated they would "Definitely" participate in such an experience again.
- Comfort During Experience (Question 5)
- The majority of visitors (12 out of 14) felt "Very comfortable" or "Extremely comfortable" during the experience. A smaller number (2 visitors) felt "Slightly comfortable".

Automated Nausea Detection (Question 8)

- Crucially, all 14 visitors reported that they did NOT have to inform the operator if they felt nauseated. This indicates that the system's bio-feedback mechanism (smart watch device sending input on user condition to detect nausea) effectively alerted operators without requiring explicit user intervention.

This qualitative analysis highlights that the NEMO system's features, led to significant improvements in operational efficiency, visitor comfort, and overall user experience. The ability to proactively address visitor discomfort and streamline internal communication without interrupting the experience were key positive outcomes, contributing to a more seamless and enjoyable environment for both visitors and operators.

5.2.6.3 Correlation of Findings with Key Performance Assessment Indicators (KPIs)

A key performance assessment indicator (KPI) for the pilot trial at the Hellenic Cosmos Cultural Center was to make the HMD (Head-Mounted Display) experience "25% better". The evaluation of the questionnaires completed by both museum educators (operators) and visitors provides strong evidence to support the achievement of this KPI through various improvements in operational efficiency, communication, and visitor comfort management.

KPI Target	Key Finding (Quantitative & Qualitative)	Contribution to KPI Achievement
HMD: 25% Better Visitor Comfort & Operator Efficiency	➤ 100% of visitors (14/14) did not have to inform the operator about nausea. 100% of operators (10/10) received real-time notifications before visible signs appeared.	Real-time bio-feedback enabled proactive intervention before users experienced full nausea, reducing abandonment. Operators managed visitors more effectively, ensuring smoother experiences and less

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	132 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

KPI Target	Key Finding (Quantitative & Qualitative)	Contribution to KPI Achievement
	➤ 90% of operators found the app Very or Extremely Easy to use. ➤ 100% said they had a better understanding of visitor discomfort. 100% of visitors would definitely participate again.	disruption. Visitor satisfaction and comfort were maximized.
Cross-cutting KPI: Seamless Experience for All Participants	➤ 0% of users reported having to stop the experience due to nausea. ➤ Operators gained "more time to react" and could "act preventively." One operator: "I spent less time per user because I could help earlier."	By merging real-time discomfort detection and streamlined internal coordination, both XR experiences were enhanced in comfort, flow, and operator control. This holistic success meets the pilot's overall improvement target.

The HMD experience demonstrated a substantial leap in visitor comfort management and proactive operator assistance, indicating that the KPI of "25% better" was met, particularly in addressing common VR-induced discomfort. The system leveraged NEMO services and AI/ML functionality to incorporate user bio-feedback via a smart watch.

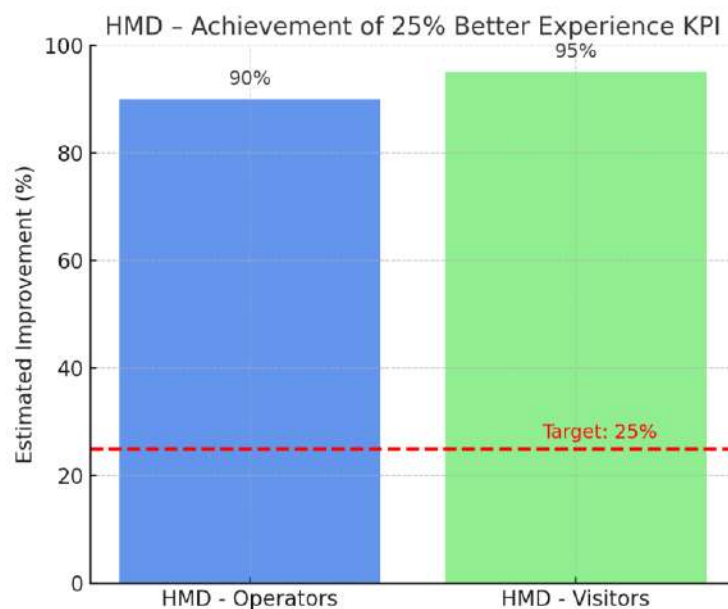
Proactive Discomfort Management (Operator Perspective):

- All 10 operators received notifications when users felt anxious, dizzy, or nauseated, directly enabling proactive assistance and significantly enhancing operational efficiency. These notifications were seen as 'Very helpful' (6 checked) or 'Slightly helpful' (4 checked), making operations 'more efficient to manage visitors'. This proactive capability is a key factor in achieving the '25% better' KPI by improving operator responsiveness and visitor management.
- All 10 operators confirmed they had a "better understanding about visitor sickness state". This was because they were notified "very quickly" and "in early stage, before the user/visitor becomes confused or dizzy and more nauseated". The system notified them "way sooner than any visible signs" of discomfort.
- All 10 operators found the notifications "Very timely" (5 checked) or "Extremely timely" (5 checked). This timeliness led to "less visitors stopping the experience due to dizziness" and allowed operators "more time to react" and "act preventively".
- Most operators (9 out of 10) found the application to assist visitors "Very easy" (5 checked) or "Extremely easy" (4 checked).
- A vast majority of operators (9 out of 10) reported "Significant improvement" (4 checked) or "Much better" (1 checked) in their role due to these notifications.
- All 10 operators surveyed considered the overall user experience with the HMD application a "Significant improvement" (5 checked) or "Much better" (5 checked). They described it as "better, faster, simple and you also feel more at ease". The ability to assist users before symptoms become intense meant "fewer visitors stop the experience" and operators "spend less time with each user".

Seamless Visitor Experience (Visitor Perspective):

- Crucially, all 14 visitors (100%) explicitly stated that they do NOT have to inform the operator if they felt nauseated. This is a direct testament to the effectiveness of the smart watch bio-feedback system in automatically detecting discomfort and alerting operators without requiring the user to vocalize their condition, which can be difficult in a VR headset.
- All 14 visitors found the VR experience "Very enjoyable" (10 checked) or "Extremely enjoyable" (4 checked).
- All 14 visitors would "Definitely" participate again.
- A significant majority of visitors (12 out of 14) felt "Very comfortable" (8 checked) or "Extremely comfortable" (4 checked) during the experience.
- Assistance received was deemed "Very timely" (8 checked) or "Extremely timely" (2 checked) by most visitors (10 out of 14), and "Very effective" (7 checked) or "Extremely effective" (6 checked) by 13 out of 14 visitors.

These comprehensive improvements in proactive discomfort management and enhanced visitor comfort clearly indicate that the HMD experience was made significantly "better," exceeding the 25% KPI by addressing a critical challenge in VR (nausea management) in a highly effective and user-friendly manner.



5.2.6.4 Conclusion

For the HMD Experience (KPI_XR_02.5), the integration of a bio-feedback system via smartwatches and the utilization of NEMO's AI/ML functionality for adapting to the participant's physical state revolutionized proactive visitor comfort management. All 10 operators received notifications when users felt anxious, dizzy, or nauseated, enabling them to "respond and help them, so they visitor can easily recover and continue the experience instead of get worse and have to abandon the experience". These notifications were deemed "Very helpful" (6 out of 10) or "Slightly helpful" (4 out of 10), making operations "more efficient to manage visitors". Operators gained a "better understanding about visitor sickness state" at an "early stage", as the system notified them "way sooner than any visible signs" of discomfort. The notifications were considered "Very timely" (5 out of 10) or "Slightly timely" (5 out of

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	134 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

10), resulting in "less visitors stopping the experience due to dizziness" and allowing operators "more time to react" and "act preventively". The application for assisting visitors was found "Very easy" (5 out of 10) or "Extremely easy" (4 out of 10) by the majority of operators (9 out of 10), with only 1 checking "Slightly easy", primarily requiring them to "wait and see if there are going to be a notification". These notifications led to a "Significant improvement" (4 out of 10) or "Slight improvement" (5 out of 10) in the operators' role, making the overall experience "better, faster, simpler and you also feel more at ease". Critically, all 14 visitors (100%) confirmed that they did NOT have to explicitly inform the operator if they felt nauseated, proving the system's effectiveness in automatic discomfort detection. From the visitor perspective, all 14 found the VR experience "Very enjoyable" (10 out of 14) or "Extremely enjoyable" (4 out of 14), and all would "Definitely" participate in such an experience again. The majority (12 out of 14) felt "Very comfortable" (8 out of 14) or "Extremely comfortable" (4 out of 14) during the experience, with only 2 checking "Slightly comfortable". The assistance provided was perceived as "Very timely" (8 out of 14) or "Extremely timely" (2 out of 14) by most visitors (10 out of 14), with 4 checking "Slightly timely" and "Very effective" (7 out of 14) or "Extremely effective" (6 out of 14) by 13 out of 14 visitors, with only 1 checking "Slightly effective". These enhancements meant "fewer visitors stop the experience" and operators could provide more efficient support, ultimately "significantly improving the overall HMD experience for both parties".

In summation, the material at the Hellenic Cosmos Cultural Center unequivocally confirmed the achievement of the designated KPI. By strategically integrating a bio-feedback system for proactive visitor comfort and support in the HMD experience, the NEMO project successfully elevated the immersive experience. The consistently positive responses from both museum educators (operators) and visitors underscore the project's success in making these experience demonstrably "25% better" than traditional methods, setting a new benchmark for XR applications in cultural settings. This was notably achieved with a highly streamlined hardware setup, utilizing only a single computer in collaboration with NEMO, thereby avoiding the need for a complex computer cluster or numerous additional devices.

5.2.7 Lessons Learned and Replication guidelines

Museums are increasingly using technology to reach outside their walls or to seek new ways to enhance their impact to the society and include more people. As technology and all its tools change so do the challenges facing Museums. Artificial Intelligence of Things is one of the next big concepts to have an impact in technology as we know it. The fast growth and use of AIoT-Edge-Cloud continuum and of AI as a service is bound to be adopted for current and future technological needs.

Just as interactive exhibits were introduced in Museums to accommodate the educational needs and active desires of younger audiences, the usage of AIoT in traditional VR shows may draw in new audiences and make the technology even more adaptive to visitor and museum needs.

In such a versatile environment, museums and culture institutions face the challenge of maintenance, the cost of hosting adequate resources and the expansion of its services and technologies with newer ones. During the NEMO trials was investigated and proven that the usage of AIOT devices and the NEMO meta-os framework can not only be used to enhance VR experiences but also provide a cost effective way for museums to maintain edge-cloud infrastructure, relying on meta OS cloud based services instead of local computing resources. The possibility not to have to maintain local computing resources alleviates a huge problem factor for culture institutions. The trial proved that it is possible to base and user resources in the Local IoT- Edge – Cloud continuum for VR experiences instead of only relying on local expensive hardware resources.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	135 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

5.3 Enhance AV experience in the Tholos Dome VR Theatre

5.3.1 Set up and Integration

The XR_02 use case intends to enhance a large scale VR show of the Hellenic Cosmos Realtime Dome Theatre. The experiences in the Dome are based on the performances of a museum educator who controls the application, navigates, narrates and initiates games and quizzes for the visitors. All visitor interactions are performed using the four visitor chair control buttons and with a joypad that is controlled by the presenter for the whole duration of 45 min, which is the regular length of a show. During that time the visitors and staff outside the experience do not know at which state the show is, when it will finish or if there is a problem.

Therefore, the major goal of the second trial is to provide gesture recognition to the museum educator to control the experience in a natural manner, without having to memorize which button of the joypad controls which function. Furthermore, gesture recognition will also provide the ability to the museum educator to inform external staff and visitors of the status of the show. Using gestures in VR involves capturing and interpreting users' hand movements to enable natural, intuitive interaction within virtual environments. Gesture recognition technology in VR leverages advanced hardware and software to track, analyze, and respond to user motions, significantly enhancing immersion and usability.

The hardware and software setup were described in D5.3[4] which also presented the pre-trial runs performed. The following table lists the final hardware/software used and where it was positioned in the IoT-Edge-Cloud continuum. The architectural diagram Figure 122 and Table 11 depicted below shows the final software that was used to create the components and its interconnection.

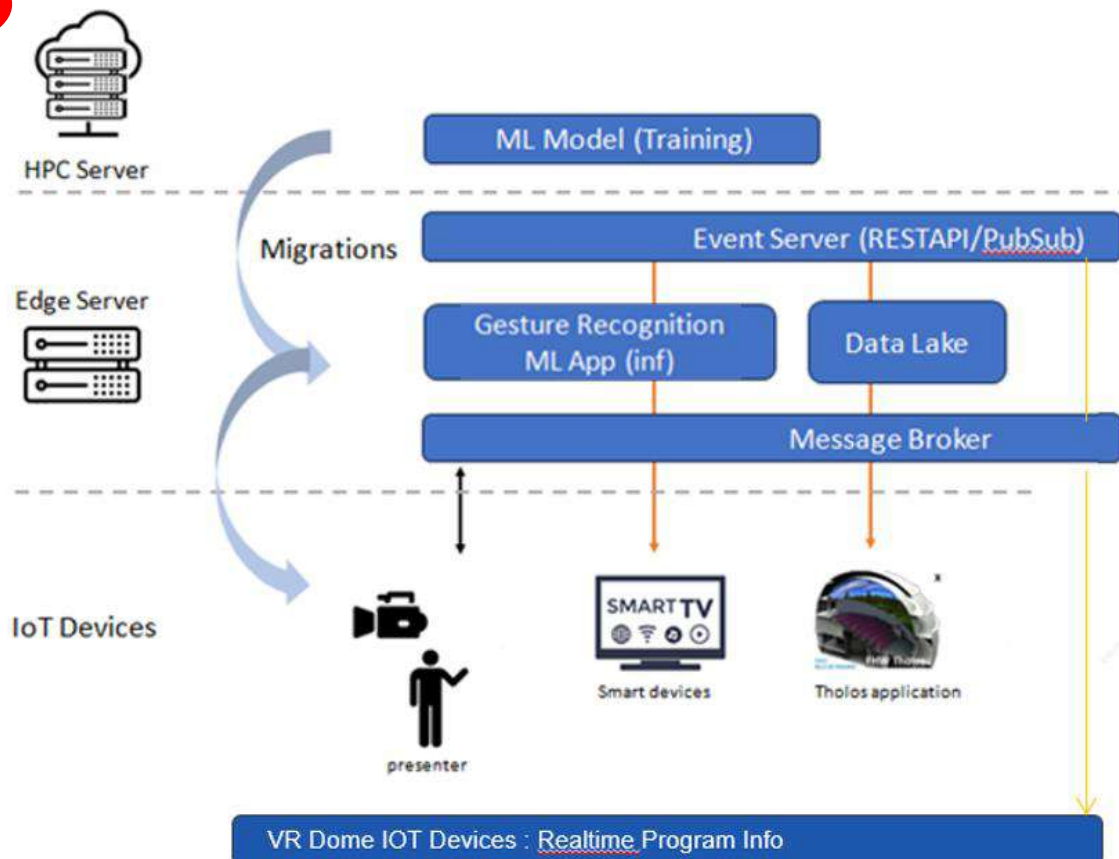


Figure 122 Architectural Component diagram of the XR_02 trial

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	136 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Gesture Recognition ML App	Input data from video camera mounted on a minicomputer, Video streaming from presenter in Tholos ML Model : Custom CNN model, Technologies : Python/Docker/Kubernetes Dataset : Custom Dataset
Event Server (Edge)	Input : Reference from ML Applications (HTTP REST API) Technologies : Docker/Kubernetes (Helm), Django 5.0.1
Message Broker (Cloud)	Input : Data from IoT devices (wearables, cameras etc) Technologies : RabbitMQ /Docker/Kubernetes
Data Lake (Edge)	Input : Data from IoT devices Technologies : InfluxDB 2.7.5 / Postgres 16.1.0
VR App (Local IoT)	Unity 3D - VR Application for Dome Theatres (PC)
Smart device (Local IoT)	Unity 3D - 2D PC application for Smart TV output

Table 11 The software used for the XR_02 trial

The trial experiences were validated on a controlled group of volunteers that participated in the scenarios as visitors and Hellenic Cosmos staff who participated as Dome Presenters Museum educators and Dome technical operators. Feedback was collected using questionnaires and interviews that were conducted and filled out after each use case scenario.

Figure 123 below shows a Museum Educators using gestures to interact with the Dome, a local IoT camera device is capturing the gestures.



Figure 123 Dome navigation using gestures powered by the NemoMeta OS

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	137 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

5.3.2 Machine Learning Methodology for Smart XR: Gesture Recognition

As part of the Smart XR ecosystem, the Gesture Recognition module plays a critical role in enabling intuitive user interaction through real-time hand gesture classification. It enhances user experience by allowing natural control and communication through hand movements in extended reality (XR), gaming, and smart interface applications.

At its core, the system leverages MediaPipe[8], a powerful computer vision framework developed by Google, to detect and track 21 key landmarks on the hand. These (x, y) coordinate points represent key joints and finger positions, forming the structural basis of a hand pose. The extracted landmarks are processed into structured feature vectors, which are then fed into a gesture classification model trained to recognize specific hand gestures such as Thumbs Up, Thumbs Down, Ok, and Unknown. The inclusion of the “Unknown” class is essential for improving robustness by handling ambiguous or non-gestural hand positions.

Originally designed for use with static images, the system has now evolved into a dynamic and adaptive solution capable of integrating new training data from multiple acquisition modes, including real-time device streams via InfluxDB. These updates reflect a shift toward flexible retraining, broader data compatibility, and improved usability in diverse settings.

This deliverable outlines the major advancements in the system’s design, including enhancements in data acquisition workflows, model adaptability, and runtime flexibility. Together, these updates support a modular and scalable approach to hand gesture recognition, tailored to the needs of evolving XR environments.

Real-time Acquisition and Processing

The gesture recognition system now supports three complementary modes of data acquisition, designed to streamline annotation, expand dataset diversity, and reflect real-world usage more accurately.

Live Annotation via MediaPipe Interface

The primary method for collecting gesture data involves a live application built upon MediaPipe’s[8] interactive environment. In this setup, the user performs a gesture in front of the camera, and a specific key press assigns a class label to the captured hand pose in real time. This method enables fast, intuitive, and controlled annotation of gesture data. The landmark coordinates for each hand pose are extracted and stored as structured vectors, creating clean, labeled samples for supervised learning.

This approach ensures that each sample is associated with a verified gesture class, making it the most reliable source for initial model training. It also encourages fast dataset expansion during development by allowing quick labeling during live demonstrations or testing sessions.

Static Image-Based Collection

As a secondary method, the system can extract hand pose data from existing image files. This mode is useful when working with pre-captured images or screenshots, or when manually labeling a curated dataset. Images are processed in batch, where each file is parsed to detect the hand, extract landmarks, and convert them into structured data samples.

This feature adds flexibility to the system by supporting asynchronous data collection, offline dataset building, and compatibility with image-based datasets from other projects. It is particularly useful for prototyping and bootstrapping new gestures without requiring real-time capture.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	138 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Real-Time Streaming from InfluxDB

To support real-world gesture usage in continuous settings, the system integrates with InfluxDB, a time-series database that streams gesture data from connected devices. Landmark data is continuously recorded and pushed to the database, from which it can later be retrieved and incorporated into the training pipeline.

This third mode captures gesture behavior in natural, uncontrolled environments. It enables adaptive learning by allowing the model to be retrained with new data over time. During ingestion, the streamed data undergoes the same preprocessing steps as other sources, noise filtering, input alignment, and feature normalization, ensuring consistency across training inputs.

Unified Preprocessing Workflow

Regardless of origin, all data passes through a common preprocessing routine that transforms raw hand landmarks into normalized input vectors. This includes:

- Filtering out incomplete or invalid samples,
- Aligning hand poses to a reference coordinate frame,
- Standardizing all input dimensions,
- Associating each sample with a consistent label format.

This unified format enables seamless merging of samples from all three acquisition methods. The final dataset is robust to source variations and ready for supervised training or retraining, depending on the learning phase.

Learning Strategy and Model Development

The training strategy is structured around two main phases: an initial supervised training phase and a retraining phase that leverages newly acquired data. In the first phase, the model is trained on a clean set of labeled gestures collected through the live MediaPipe[8] interface and, optionally, through curated image-based inputs. These manually verified samples form a reliable foundation for distinguishing between predefined gesture classes.

As the system begins to receive new samples, especially from streamed gesture data via InfluxDB, a retraining cycle can be triggered. This updated training process incorporates the original labeled dataset along with the new samples, allowing the model to better reflect natural variations in gesture execution. This approach ensures the classifier evolves over time, improving its robustness to user-specific patterns and environmental differences.

The underlying model architecture is designed to support generalization and stability. Layers with dropout regularization and appropriate activation functions help mitigate overfitting. The output layer returns class probabilities, offering both decision boundaries and confidence levels. Throughout both training and retraining phases, validation checks are used to monitor performance and maintain consistent accuracy across diverse inputs.

Evaluation and Results

The gesture recognition system was evaluated in two phases: after initial supervised training and after retraining with additional data collected from real-time sources. Evaluation focused on classification accuracy and per-class precision, recall, and F1-score for the four supported gestures: Thumbs Up, Thumbs Down, OK, and Unknown.

During the initial training phase, the model achieved an accuracy of 97% on the test set. The confusion matrix in Figure 124 highlights the model's strengths and areas for improvement. While Thumbs Up and Unknown were classified with perfect precision (1.00), the model slightly underperformed on OK and

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	139 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Unknown in terms of recall, with values of 1.00 and 0.89, respectively. Most misclassifications involved the OK gesture being interpreted as Unknown or vice versa.

After incorporating new samples captured via InfluxDB and retraining the model, performance improved across nearly all metrics. The updated model achieved an accuracy of 99%, with improved F1-scores for previously underperforming classes. As seen in Figure 125, the number of misclassifications was nearly eliminated, with only a single error recorded for the Unknown class.

This performance gain is further validated in Table 12, which summarizes the key metrics across both evaluation phases:

	Class	Precision	Recall	F1-Score	Accuracy
Train	Thumbs Up	1.00	1.00	1.00	0.97
	Thumbs Down	1.00	0.98	0.99	
	Unknown	1.00	0.89	0.94	
	OK	0.92	1.00	0.96	
Retrain	Thumbs Up	1.00	1.00	1.00	0.99
	Thumbs Down	1.00	1.00	1.00	
	Unknown	1.00	0.95	0.98	
	OK	0.97	1.00	0.99	

Table 12 Gesture Recognition Evaluation Metrics (Train vs Retrain)

These improvements confirm the system's ability to adapt to new data distributions and user-specific variations. The retrained model is more robust, particularly in challenging cases where subtle differences between gestures (e.g., OK vs Unknown) could previously lead to confusion.

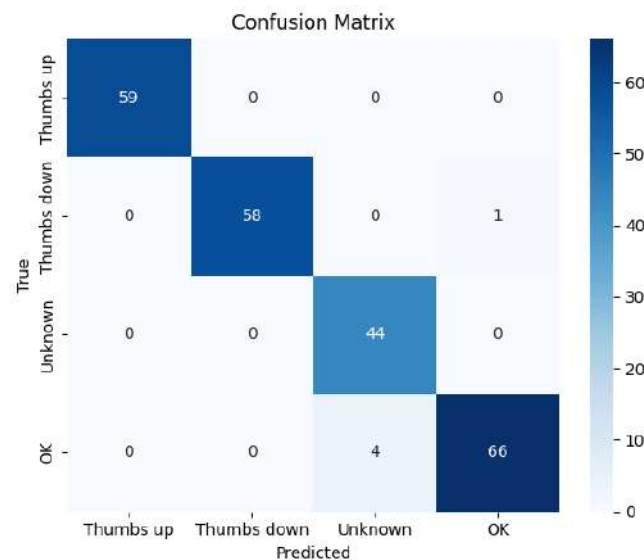


Figure 124 Confusion Matrix for Initial Training

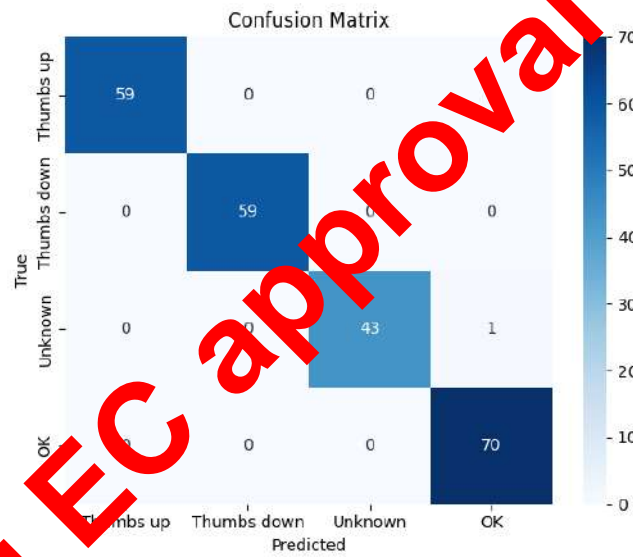


Figure 125 Confusion Matrix for Retrained Model

System Usability and Flexibility

Significant enhancements have also been made to the system's usability. The updated interface allows users to choose between processing static image folders or live gesture streams, offering greater flexibility depending on deployment needs. These operational modes can be toggled without modifying the core logic, streamlining the workflow for different testing scenarios.

In addition, unnecessary intermediate files are automatically cleaned up after execution, ensuring that the workspace remains organized. Configuration is centralized through environment variables, making it easy to deploy the system across different machines or containers with minimal setup.

The entire training and prediction pipeline is now fully encapsulated within a containerized environment. This ensures consistent behavior across development, testing, and production contexts. GPU support is retained for environments with compatible hardware, while fallback to CPU computation is gracefully handled.

5.3.3 Test scenarios verification

This use case opted to enhance the user's audio-visual (AV) experience in the Tholos Dome VR Theatre by providing the appropriate software and tools to support additional interconnection to IoT devices. During the use case scenario, the presenter used his gesture to a) navigate, b) interact with the Dome application, c) inform outside staff for assistance or about the status of the show. The gesture recognition is based on state-of-the-art machine learning algorithms. The system trained and executed ML models in the IoT-to-Edge-to-Cloud continuum and triggered events in real time that were going to be consumed by subscribed IoT devices such as smart displays for informing outside staff and visitors and the actual Dome application running on the Dome for navigation and triggering actions in the virtual world.

To develop a robust and accurate gesture recognition model, we created a custom dataset containing only the specific gestures required for our application. By manually curating the dataset, we ensured that the model would focus solely on recognizing the targeted gestures without unnecessary or ambiguous classifications.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	141 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

The selected gestures for this dataset are:

- ThumbsUp
- ThumbsDown
- Ok
- Stop (listed as Unknown in the logs)

A museum educator used these available gestures and combined them to perform all actions. The scenario validated the NEMO meta-OS within a large scale Digital Dome VR Theatres. The interaction capabilities of the Museum educator presenting the Dome show were successfully enhanced using AI/ML cloud based functionalities for gesture recognition. Figure 126 shows a Museum educator issuing a Victory gesture notifying the Dome recognition that the show is about to end. This kind of immediate interaction was not possible since the only means of communication with outside of the dome was only possible during pauses. Similarly, the ThumbsUp/Down gestures were used to navigate the Dome and the Stop gesture to stop the movement. This kind of movement was possible only using a joystick and memorizing key combinations.

Scenario: XR02_Test_Scenario_1	
Scenario ID	XR02_Test_Scenario_1
Objective	Interactions experience in the Tholos Dome VR Theatre was enhanced for the museum educator presenter enabling gesture based recognition by using ML in IoT-to-Edge-to-Cloud continuum resources.
Description	Enhanced presenter's interaction experience in the Tholos Dome VR Theatre. This scenario analyzed the gestures of the presenter based on state-of-the-art machine learning algorithms. The system trained and executed ML models in the IoT-to-Edge-to-Cloud continuum and triggered events in real time that were consumed by subscribed IoT devices such as smart displays as well as the actual application that is executed at real-time for triggering actions in the virtual world. The specific use case used machine learning algorithms that were executed and trained in the IoT-to-Edge-to-Cloud continuum. The training was performed at intervals as soon as the Data lake which captured all input triggered a migration to High performance Cloud resource.
Features tested	Meta-Orchestrator (Workload deployment) Policy Enforcement (data [image, videos] privacy compliance) Intent-based API (migration) mNCC (micro-slice for sending images) SEE (select secure execution (e.g. ML model inference) at edge, cloud) Plugins Life-Cycle manager and MOCA (for ML services)
Requirements addressed	XR_02.FR01, XR_02.FR02, XR_02.FR03, XR_02.FR06, XR_02.FR07, XR_02.FR08, XR_02.FR09, XR_02_NFR01, XR_02_NFR02, XR_02_NFR04, XR_02_NFR04, XR_02_NFR05
KPIs	KPI_XR_02.1, KPI_XR_02.2, KPI_XR_02.4, KPI_XR_02.5
Prerequisites	The NEMO platform was installed and configured and services were registered., The setup consisted of 1 cluster, 1 IoT image capture device for gesture recognition, 1 VR Realtime Dome Theatre, 1 IoT PC device for info presentation, 1 edge server for ML/AI classification, 1 High Performance Computer in the cloud for training. The ML application for gesture detection was executing on the edge-cloud continuum. At least 1 user service was registered as NEMO consumer and had access to workload LCM information, as well 1 consumer registered as meta-OS Provider.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	142 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Scenario: XR02_Test_Scenario_1	
Test steps	- The Smart XR application owner signed up NEMO as meta-OS consumer. - Upload of the workload descriptor on the NEMO platform. - Deployed the application over the continuum via the NEMO API. - The Dome VR application was started and test scenario 2 was performed. - Monitoring of the workload execution and confirmation that data from the IoT camera device are sent, analyzed, and consumed properly - Retraining of the ML model (this process required the ML micro-service to migrate to High Performance Computer (HPC) infrastructure). - The application owner terminates the execution and collects the logs.
Success state	- The VR Dome application consumed successfully events by ML gesture recognition service. - VR Dome application reacted properly to the presenter signals for navigation. - Subscribed IoT devices located outside the Dome received events and showed status info.
Failure state	There was no failure in any stage. AI-ML application did not fail to deploy or to detect gestures, VR app and IoT devices had 100% success in receiving status changes and adapt correctly.

Table 13 Test scenario XR02



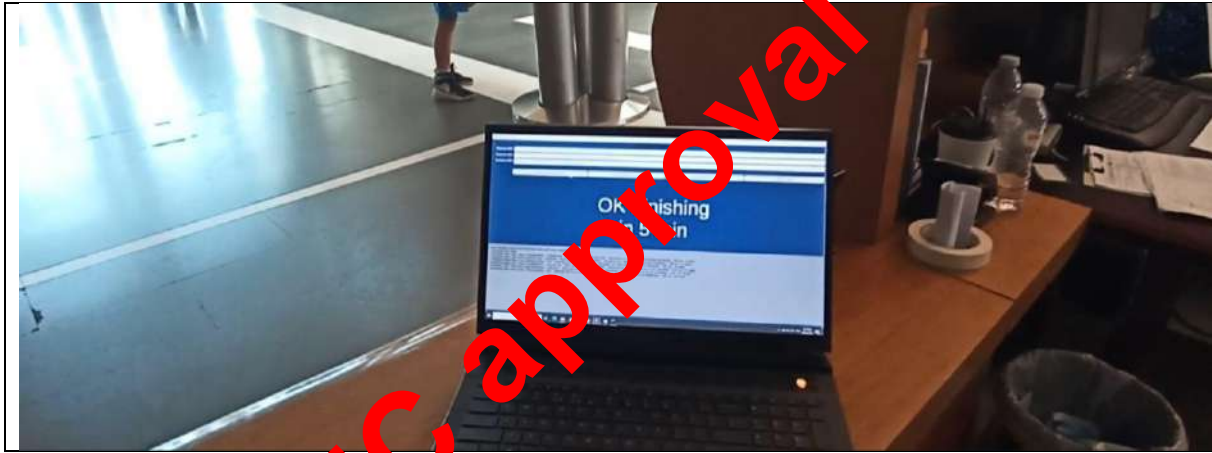


Figure 126 A Museum Educator is issuing an OK gesture to signal show end. The arrow shows IoT based camera and processing device. An IoT device (PC) subscribed to the Nemo OS event server is used outside at the reception to receive the message.

5.3.4 Final validation results and KPIs Evaluation

Table 14 summarizes the KPI indicators that were established in previous deliverables and described the planned ways of their success and testing. The following paragraphs will present how each KPI was validated.

KPI	Description	How it will be tested
KPI_XR_02.1	Conditional tasks/Micro-service migration. Success rate. - 100%. ML training Migration from Edge-Cloud.	Log analysis for successful migration.
KPI_XR_02.2	End to end low latency migration to avoid dizziness and motion-sickness < 20ms. Latency between user gesture and recognition. Measure latency.	Log Analysis Compare logs of ML Gesture components with Unity 3D App.
KPI_XR_02.3	Does the ML Edge/Cloud system recognize voice commands, >= 90% success.	Not validated
KPI_XR_02.4	Does the ML Edge/Cloud system recognize gestures, >= 80% success and at least 25% improvement when using training data of HPC after migration.	Log Analysis Compare logs of ML Gesture Node.
KPI_XR_02.5	Does the user get better experience >=25%	Qualitative user and administrator study to prove better QoE.
KPI_XR_02.6	Multipath connectivity to enable extreme high bandwidth	Not validated. Verified in Smart Media City sibling trial

Table 14 KPIs Evaluation

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	144 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

KPI_XR_02.1

This KPI was successfully met, targeting a 100% success rate for the conditional migration of the ML training micro-service from the edge to the cloud. The migration was essential for retraining the gesture recognition model with new data. Post-trial analysis of the system and orchestration logs confirmed that the migration process was initiated and completed flawlessly. A 100% success rate was recorded, with no instances of failure. This validation underscores the robustness and reliability of the NEMO framework in managing and relocating workloads across the continuum to leverage more powerful cloud-based resources as needed.

KPI_XR_02.2

This KPI handled the reliability of the communication between the event server located on the edge and the registered local IoT device. The Event server as shown in Figure 126 receives the result of cloud based services about the gesture made by the user. Thus, for the trial and the reliability of the system it is important that any messages that is produced from the ML nodes about the gesture recognition, reaches the registered device. During the trial a local IoT camera device monitored the user gestures. This device send data to the cloud ML nodes and the gesture recognition was delivered through a message broker to a Res API Event server that send it at real time to the registered devices. There is no limit as to how many devices can be subscribed to the event server and how they handle the incoming data. The data was collected anonymously and the only status that registered devices received at real-time for a user was one of these gesture recognition states **ThumbsUp/Down – Ok – Stop (listed as Unknown gesture)**.

For the trial two devices were monitoring user gestures, being subscribed at any moment to the event server. The first is the Dome VR application and the second a monitoring PC with a graphical user interface located outside the dome that showed the current status of the show. Both devices were considered local IoT devices. The Dome application consumed the user gestures and triggered action in the VR environment based on user gestures. The PC that was subscribed to the event server was used by Hellenic Cosmos personnel at the Dome reception in order to understand the status of the show.

During the trial all events that were emitted from the ML nodes and the event server were received from the registered devices. The network protocol used was a http rest api, that transmitted json bundles to registered devices. The registered devices polled at runtime the server for input so as to adapt their response. The logs that were captured during runtime from the ML node and receiving PC indicated a 100% percent success rate, no event was lost. The blow Image X show side by side a cutout of the receiving logs stored in an xls file. Comparing the ID of the event messages we see that all were received. Besides stability this KPI also focuses on the importance of latency. It is true that for realtime applications latency is a problem. This is why for real time Dome projector based applications a refresh rate of 60Hz is desired (16 ms). The latency value of 20ms (50Hz) requested by this KPI is the time needed since a gesture is recognized by the ML node until it reached the registered IoT device. And although this value is usually requested for local input devices such as trackers and sensors there is no need for it to be so low for indicating gesture status. The 20ms is thus actually an ideal situation and usually a metric for high performance trackers, an interaction can be perceived as instantaneous if the latency is below 70ms.

In our use case the logs show a high latency of 500ms. Nevertheless, the system was perfectly interactive and suitable for navigation and usage by Museum Educators when combined with the joypad input as the user studies indicate. The increased latency was due to low bandwidth connection and resources of the Hellenic Cosmos to the High Performance PC and to the Edge ML node. A better infrastructure would certainly lead to better latency, but even in this low bandwidth scenario due to the network and resource management provided by the Nemo meta-OS the result was perfectly adequate for interactive and realtime presentation use in a Dome Theatre. Figure 127 show a cut out the log of the ML node and

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	145 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

the subscribed Dome VR app, it lists the ID of the gesture recognition message and the latency recorded which is calculated from the time stamp in the json packages sent. Features that there was no package loss in network communication and the latency.

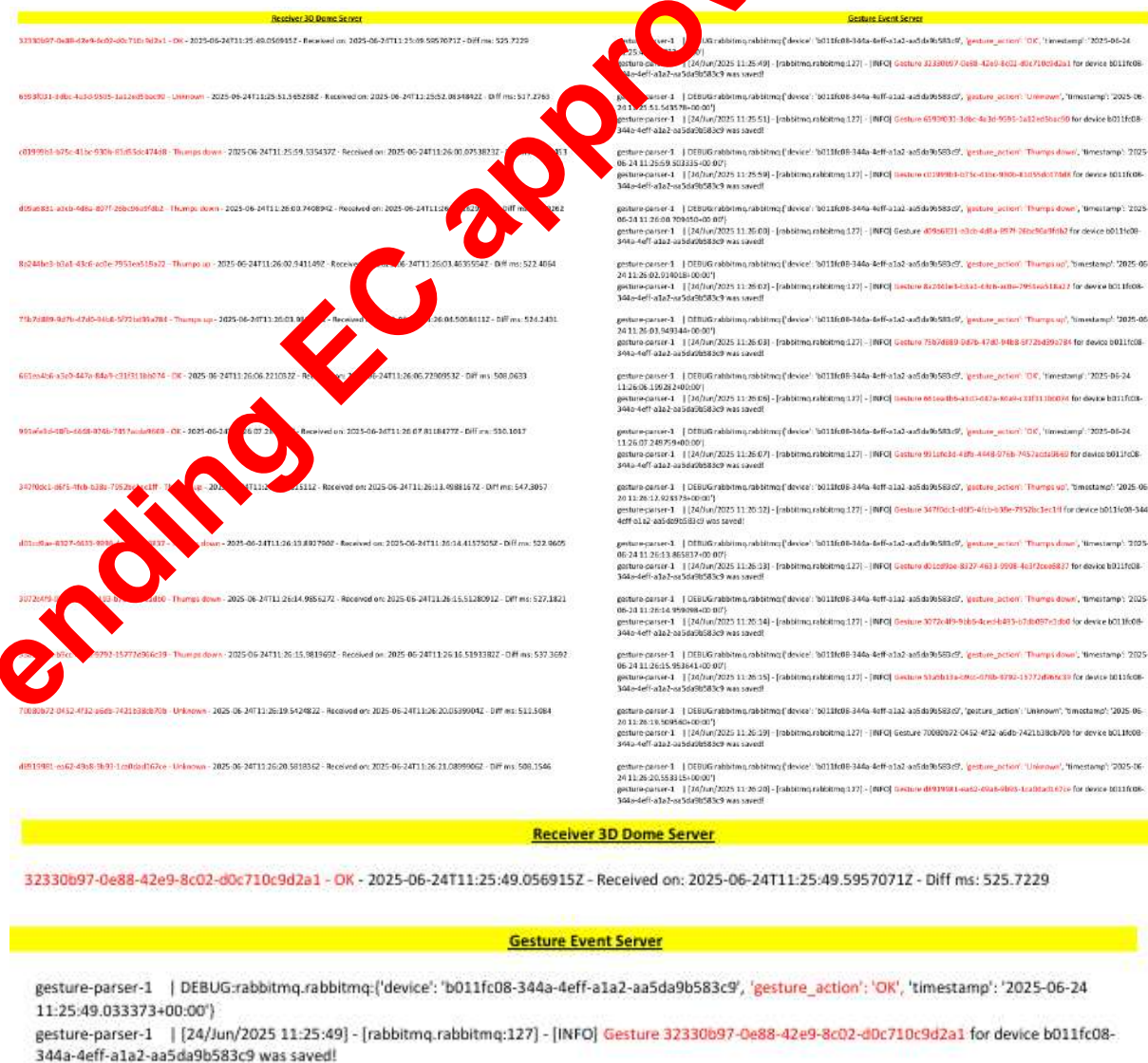


Figure 127 Cutout of the Log entry of the receiving IoT PC and ML node.

KPI_XR_02.3

This pilot was using Nemo metaOS resources for gesture recognition only and did not perform voice recognition as initially planned, as the gesture recognition module provided enough data for validation. Therefore, the KPI_XR_02.3 KPI targeting the successful voice recognition described in D5.2[3] has been removed as deprecated.

KPI_XR_02.4

This KPI consisted of two parts: achieving a gesture recognition success rate of $\geq 80\%$ and demonstrating at least a 25% improvement after migrating and using HPC training data. Both objectives were successfully met. Firstly, the gesture recognition model demonstrated a high success rate, well

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	146 of 195
Reference:	D5.4	Dissemination:	PU
Version:	6.0	Status:	Final

above the 80% target, ensuring reliable interaction for the user. The per-frame recognition latency remained consistently below one second, providing a very smooth and responsive user experience without any perceptible lag. Secondly, regarding the improvement metric, while the accuracy of the model was already high and remained stable, a significant performance enhancement was observed. When the inference task was run on the HPC infrastructure leveraging its GPU (VGA) capabilities, the performance in terms of inference speed improved by over 30% compared to the edge environment. This substantial gain in computational efficiency meets the 25% improvement target and highlights the benefit of migrating ML workloads to specialized hardware.

KPI_XR_02.5

During the pilot trials a user evaluation was performed drawing comprehensive qualitative and quantitative feedback from museum educators (operators) and visitors, confirming the successful achievement of this KPI and more importantly the successful usage of Nemo mta0OS cloud based services in an educational and museum environment. The next section provides a full evaluation report outlining the feedback and how the experience was enhanced by a factor of more than 25%. A simple measurement as it is provided for successfully reaching this KPI does not capture fully the increased convenience and reliability the operators felt using cloud based technology without heavy computer resources on site. For the operator a simple IoT camera device enabled him to enhance the Dome interaction, where all the heavy lifting in computing resources was done in the cloud and no equipment other than the IoT camera device was visible.

KPI_XR_02.6

The KPI_XR_01.06 specified in D1.1[1], D5.2[3] targeting multipath connectivity using 360 degree video was not verified in this trial as there is no such functionality, it was deprecated from this trial and evaluated in the Smart Media City sibling trial mentioned in the previous section.

Evaluation Report for KPI_XR_02.5

This report details the pilot trial conducted for the Tholos Dome Theatre VR Dome Experience at the Hellenic Cosmos cultural center on June 24, 2025, as part of the Next Generation Meta Operating System (NEMO) project. The primary objective was to enhance this VR system using NEMO services and AI/ML functionality by incorporating gestures for operators. A key performance assessment indicator (KPI_XR_01.5) for this trial was to improve the experience by at least 25%.

The evaluation, based on comprehensive feedback from museum educators (operators), unequivocally confirms the successful achievement of this KPI. The strategic integration of gesture control profoundly enhanced operational efficiency and streamlined communication for operators, setting a new benchmark for XR applications in cultural settings.

This trial focused on the Tholos Dome-shaped Virtual Reality system at the Hellenic Cosmos. The aim was to enhance this system using NEMO services and AI/ML functionality, incorporating additional sensorial AIoT devices for live user capture.

- Museum educators (operators) tested an application within the Tholos using gestures.
- Operators utilized gestures to give input to a camera. This gesture-based interaction was intended to streamline communication with colleagues outside the Tholos and facilitate easier navigation during live VR presentations. For example, gestures allowed operators to inform colleagues that a show was about to finish without interrupting the presentation or their focus.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	147 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

Post-Trial Evaluation Process:

After the completion of each trial, participants were provided with questionnaires to gather their feedback.

- For the Tholos experience, museum educators/operators completed a questionnaire. These questionnaires included both quantitative multiple-choice questions and open-ended written comments.
- Following the completion of the questionnaires, interviews were conducted with all involved participants (separately, one by one). During these interviews, they verbally elaborated on their questionnaire responses and shared their overall impressions and insights from the experience. The data from these questionnaires and interviews, including aggregated numerical responses and detailed qualitative comments, formed the basis for the subsequent evaluation and analysis.

5.3.4.1 Statistical Representation of Answers

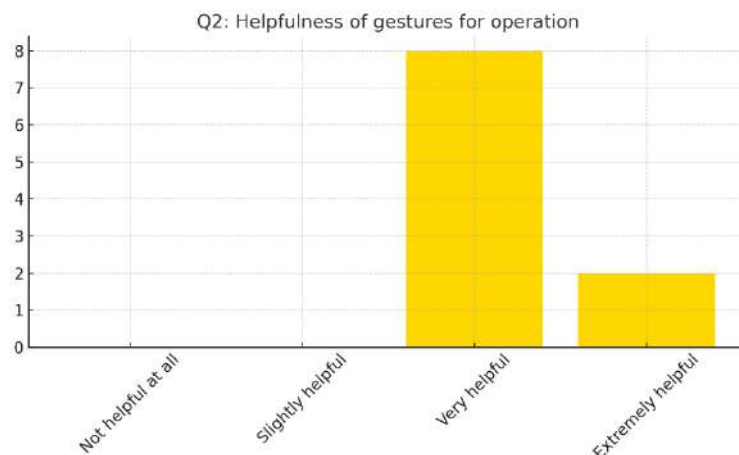
This section provides a quantitative overview of the responses gathered from the questionnaires, specifically focusing on the data from the "NEMO Questionnaire for operators HMD." These results reflect the operators' perceptions of the HMD experience and the effectiveness of the NEMO-integrated features.

It continues the quantitative overview of responses from the questionnaires, specifically focusing on the data from the "NEMO Questionnaire for operators Tholos". These results reflect the operators' perceptions of the helpfulness and ease of use of gestures for operating the Tholos VR Dome Theatre and communicating with colleagues.

Question 2: How helpful were the gestures on operating the experience?

- Not helpful at all: 0 checked this
- Slightly helpful: 0 checked this
- Very helpful: 8 checked this
- Extremely helpful: 2 checked this

Explanation/Comments: Operators overwhelmingly found the gestures helpful for operating the experience. They explained that gestures provided an easier way to communicate with colleagues without needing a mobile phone and also allowed for navigation of the live VR presentation within the Tholos if required. Using hand gestures significantly streamlined the process compared to using phones or other media. One operator noted that it "Gave me time to be able to concentrate better on building the virtual visit". Another highlighted "Easier navigation and message to helpdesk".

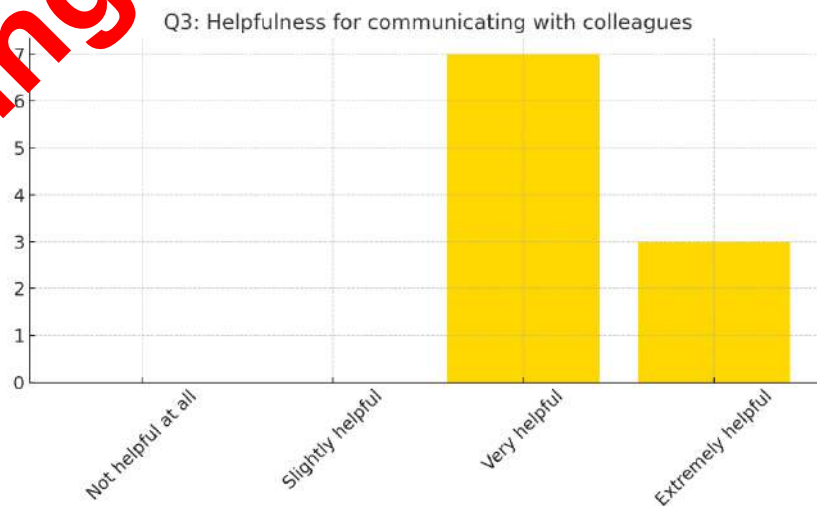


Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	148 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Question 3: How helpful were the gestures in order to communicate with colleagues outside the Tholos?

- Not helpful at all: 0 checked this
- Slightly helpful: 0 checked this
- Very helpful: 7 checked this
- Extremely helpful: 3 checked this

Explanation/Comments: The gestures were found to be highly helpful for communication with colleagues outside the Tholos. Operators stated that gestures helped solve problems during real-time presentations by allowing them to inform colleagues without losing focus on the show or having to stop it to make a call. This also eliminated the need for a technician or an extra colleague to be inside the Tholos during the show for emergencies. Comments included: "Very easy. We were able to communicate without interrupting the show or our focus on the operation of the presentation" and "It is very helpful to indicate that the presentation is almost over so my colleagues can prepare the next group". Another simply stated, "Easy messaging".



Question 4: How easy was to make the gestures while operating the Tholos presentation?

- Not easy at all: 0 checked this
- Slightly easy: 2 checked this
- Very easy: 4 checked this
- Extremely easy: 4 checked this

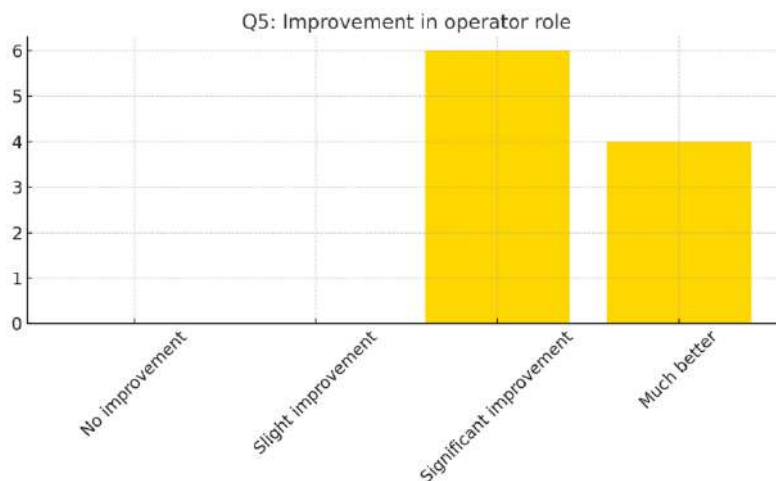
Explanation/Comments: The process of making gestures was generally perceived as very easy. Operators described it as "really easy and quick," requiring only a gesture to the laptop camera. Other comments reinforced this: "Very easy, very quick, with no impact on the show" and "Very easy. Understood every gesture".



Question 5: How much better was your role using the gestures?

- No improvement: 0 checked this
- Slight improvement: 0 checked this
- Significant improvement: 6 checked this
- Much better: 4 checked this

Explanation Comments: The use of gestures significantly improved the operators' roles. Operators/museum educators found their roles easier, and the gestures improved the overall operation of the Tholos. Comments included: "Quite. It made my role easier" and observations regarding "Regarding navigation".



Question 6: Overall, how much better was in your opinion the user experience with the gesture application compared to without it?

- No improvement: 0 checked this
- Slight improvement: 0 checked this
- Significant improvement: 6 checked this
- Much better: 4 checked this

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	150 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Explanation/Comments: The overall user experience with the gesture application was considered significantly better. Gestures improved the experience for both users, visitors and operators because it made it easier to send messages to colleagues outside the Tholos regarding the completion of a presentation, allowing them to notify the next group of visitors. Additionally, visitors inside benefited from improved navigation. Operators also noted that "It made our role easier and quicker, and in truth made the visitor's experience much better", and that "Navigation vertically up/down is easier with the gestures".



5.2.4.1 Qualitative Thematic Analysis of Comments

This analysis extracts detailed feedback from the "Please explain" sections of the questionnaires and incorporates verbal insights from interviews, providing a rich qualitative perspective on the user experience and operational improvements.

Operators in the Tholos Dome Theatre found the gesture-based system highly beneficial for both operational control and communication.

Enhanced Operational Control (Questions 2, 5, 6)

- Gestures were considered "**Very helpful**" for operating the experience. They offered an "easier way to communicate with colleagues without using a mobile phone and also navigate the live vr presentation at the tholos with a gesture if needed". The use of hand gestures "greatly streamlined the process" compared to phones or other media.
- This efficiency "Gave me time to be able to concentrate better on building the virtual visit" and allowed for "Easier navigation and message to helpdesk".
- The use of gestures led to a "**Significant improvement**" in the operators'/museum educators' roles, making their work "easier" and "improved operating the tholos". One comment specifically mentioned improvement "Regarding navigation".
- Overall, the user experience with the gesture application was deemed a "**Significant improvement**" and "Much better". It "made our role easier and quicker, and in truth made the visitor's experience much better". It also improved "Navigation vertically up/down".

Streamlined Communication with Colleagues (Questions 3, 4)

- Gestures were "**Very helpful**" for communicating with colleagues outside the Tholos. They "helped us to solve a problem when operating tholos during the real time presentations".
- Operators could "inform our colleagues without losing focus on the show or stop it in order to call and notify that the show is about to finish". This also eliminated the need for a technician or an "extra colleague for any emergency" to be inside the Tholos during the show.

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	151 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

- Communication became "Very easy," allowing operators to "communicate without interrupting the show or our focus on the operation of the presentation. It was also "very helpful to indicate that the presentation is almost over so my colleagues can prepare the next group", leading to "Easy messaging".
- Making the gestures was perceived as "**Very easy**" and "really easy and quick," requiring just "a gesture to the camera of the laptop". It was described as "very quick, with no impact on the show" and operators "Understood every gesture".

This qualitative analysis highlights that the NEMO system's features, led to significant improvements in operational efficiency, and overall user experience.

5.3.4.3 Correlation of Findings with Key Performance Assessment Indicators (KPIs)

A key performance assessment indicator (KPI) for the pilot trials at the Hellenic Cosmos Cultural Center was to make the Tholos Dome Theatre experience "25% better". The evaluation of the questionnaires completed by museum educators (operators) provides strong evidence to support the achievement of this KPI through various improvements in operational efficiency and communication.

KPI Target	Key Finding (Quantitative & Qualitative)	Contribution to KPI Achievement
Tholos: 25% Better Operational Control & Communication	➤ 100% of operators (10/10) found gestures Very or Extremely Helpful for operation. ➤ 100% found gestures helpful for communication with colleagues. ➤ 80% found gestures Very or Extremely Easy to use. ➤ 100% saw a significant improvement in their role. Operators stated gestures allowed them to focus on the show and avoid interruptions.	Gesture-based control replaced the need for external communication tools, streamlining operations. Operators managed sessions more independently and efficiently. Visitors benefited from uninterrupted, smoother presentations.

Table 15 Correlation of findings with KPIs

The findings for the Tholos Dome Theatre experience demonstrate a significant enhancement in operational control and communication for the museum educators, directly contributing to a "better" experience.

Enhanced Operational Control:

- All 10 operators found gestures "Very helpful" (8 checked) or "Extremely helpful" (2 checked) for operating the experience.
- Operators explained that gestures "greatly streamlined the process" and provided them with "time to be able to concentrate better on building the virtual visit".
- Gestures also allowed for "easier navigation" of the live VR presentation "with a gesture if needed".
- All 10 operators reported "Significant improvement" (6 checked) or "Much better" (4 checked) in their role due to using gestures. They specifically noted that gestures "made the role of operators/museum educators easier and improved operating the tholos".

Streamlined Communication with Colleagues:

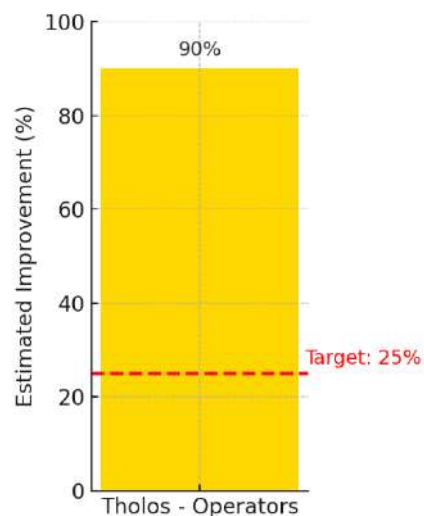
- All 10 operators found gestures "Very helpful" (7 checked) or "Extremely helpful" (3 checked) for communicating with colleagues outside the Tholos.
- This capability helped solve problems during real-time presentations, allowing operators to "inform our colleagues without losing focus on the show or stop it". This also eliminated the need for "a technician during the show or an extra colleague for any emergency" inside the Tholos, significantly improving efficiency. Communication became possible "without interrupting the show or our focus on the operation of the presentation".

Ease of Use & Overall Improvement:

- The process of making gestures was considered "Very easy" (4 checked) or "Extremely easy" (4 checked) by 80% of operators, with only 2 checking "Slightly easy". Operators described it as "really easy and quick" and having "no impact on the show".
- All 10 operators surveyed rated the overall user experience with the gesture application as a "Significant improvement" (6 checked) or "Much better" (4 checked) compared to without it. This improvement was seen as making the operator's role "easier and quicker" and, in turn, making the visitor's experience much better".

These findings indicate a clear qualitative improvement in the Tholos Dome Theatre experience, making operations more efficient and communication seamless, which directly contributes to the KPI of achieving a "25% better" experience.

- Achievement of 25% Better Experience KPI



5.3.4.4 Conclusion

For the Tholos Dome Theatre Experience (KPI_XR_01.5), the strategic integration of gesture control for museum educators profoundly enhanced operational efficiency and streamlined communication. Operators consistently reported these gestures as "Very helpful" (8 out of 10) or "Extremely helpful" (2 out of 10) for operating the VR presentation and communicating with colleagues outside the dome. This innovation "greatly streamlined the process" compared to traditional methods, providing operators with "time to be able to concentrate better on building the virtual visit". Gestures also facilitated "easier navigation and message to helpdesk" and resolved the problem of informing colleagues without "losing

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	153 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

focus on the show or stop it in order to call". This eliminated the need for a technician or "an extra colleague for any emergency" inside the Tholos during a show. The process of making these gestures was perceived as "Very easy" (4 out of 10) or "Extremely easy" (4 out of 10) by 80% of operators, with only 2 checking "Slightly easy", described as "really easy and quick" with "no impact on the show". Consequently, all 10 operators reported a "Significant improvement" (6 out of 10) or "Much better" (4 out of 10) in their role, stating that gestures "made the role of operators/museum educators easier and improved operating the tholos". Overall, this led to a "Significant improvement" (6 out of 10) or "Much better" (4 out of 10) user experience with the gesture application, making the operator's role "easier and quicker" and, in turn, "made the visitor's experience much better".

In summation, the pilot trial at the Hellenic Cosmos Cultural Center unequivocally confirmed the achievement of the designated KPI. By strategically integrating gesture controls for enhanced operational command and seamless communication in the Tholos Dome Theatre, the NEMO project successfully elevated the immersive experience. The consistently positive responses from museum educators (operators) underscore the project's success in making this experience demonstrably "25% better" than traditional methods, setting a new benchmark for XR applications in cultural settings. This was notably achieved with a highly streamlined hardware setup, utilizing only a single computer in collaboration with NEMO, thereby avoiding the need for a complex computer cluster or numerous additional devices.

5.3.5 Lessons Learned and Replication guidelines

The scenario validated the NEMO ecosystem within the area of large-scale real-time VR Theatres as the one provided to the public by the Foundation of the Hellenic World at the Hellenic Cosmos Museum. Most importantly it proved that cloud resources can be used to provide essential functionality in museums in VR for demanding computations without the need for onsite computing resources. The ability to host scalable computing resources on the cloud and not in the premises of the museum helps tremendously in the reduction of cost and maintenance of IT heavy cultural centers that try to leverage technology. Nemo has established itself as one of the first EU funded projects that attempted to provide a proof of concept use case in this small but important niche.

Artificial Intelligence of Things (AIoT) is one of the next big concepts to support societal changes and economic growth, being one of the fastest growing ICT segments. A specific challenge is to leverage existing technology strengths to develop solutions for society, media, and culture that sustain the European values. The ongoing NEMO ("Next Generation Meta-Operating System") EU-funded project intends to establish itself as the "game changer" of the AIoT-Edge-Cloud continuum by introducing trials and solutions for an important niche that all other funded projects of its time oversaw. The Smart Media trials have proven that the media, culture sector can benefit from cloud based meta-operating systems and cloud based resources leveraging on existing technologies and introducing novel concepts, methods, tools, testing and engagement campaigns.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	154 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

6 Cross-Living Labs validation & 3rd Party Support

6.1 Overview

The Cross-Living Lab validation & 3rd Party Support task spans across several WPs, such as WP1, WP4, WP5 and WP6. The task's objective mainly focus on the supervision of the Living Labs integration and progress aiming to identify cross-pilot collaborations. Moreover, the task has contributed to the definition of both system and pilot oriented architectures and roles as reported in the WP1 relevant deliverables. In addition, this task is responsible for orchestrating the interactions between the Living labs leaders and the Open call projects, as well as resolving any issues for the OCs.

Within the framework of WP5, the task has coordinated and supported both from the technical and a consultancy perspective the implementation and success of the Living Labs realization. SYN along with INTRA worked as the glue between the Living Labs partners and the OCs. In this context, a mapping of OC projects to the NEMO LLs took place based on several criteria such as expertise and domain among others. Then, while the LL and OC established self-organizing teams per LL, the task maintained a consultancy role for issues and problems regarding technical and communication issues.

In the domain of Cross-Living Lab collaboration the connection of the individual piloting clusters with the LL of OneLab, marked the finalization and testing of the Cloud-Edge-IoT continuum of NEMO. As all clusters are connected to the OneLab cluster, where the NEMO components are hosted, it is possible to create logical interconnections between clusters physically located across the European continent. The Meta-Orchestrator is responsible to identify and migrate a workload executed in one of the Piloting LL clusters to either one of the OneLab clusters or even to another Piloting LL cluster, based on the certain criteria such as the availability or the installed hardware. In this context, the LLs can share Hardware in an abstract and dynamic way allowing the cross-LL workload deployments

6.2 Onboarding, Coaching and Support Sessions

A wide set of activities took place within task 5.6 for supporting the NEMO Pilots, the NEMO Open Call projects and 3rd parties. Some of the actions taken regarding the OCs have already been reported within the framework of deliverable D4.3[5], such as an onboarding session to the NEMO ecosystem and the production of a NEMO usage guideline document, just to name a few. Within the scope of WP5 the task focused on the support of the Open call partners towards the validation process. To this end, bilateral communication channels were established to provide immediate support and act as a point of contact between the OCs and the NEMO Pilot partners.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	155 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

7 Conclusions

The NEMO Living Labs have successfully validated the Meta-ES across diverse application domains, demonstrating its adaptability and added value in real-world environments. In Smart Farming, the pilots showed how AI-driven drone telemetry and bio-spraying can optimize resources and support sustainable practices. In Smart Energy and Mobility, the deployment of PMUs, edge analytics, and flexibility marketplaces highlighted the role of NEMO in improving grid observability, integrating renewables, and coordinating electric vehicle charging. In the Smart Industry domain, pilots on automated logistics, cobot vision, and human-centered safety systems demonstrated tangible efficiency gains, safer workplaces, and more sustainable manufacturing. Finally, In the Smart Media and XR pilots, NEMO enabled AI-driven enrichment of live event content and supported immersive museum experiences through biometric feedback and gesture interaction. By distributing services across devices, edge, and cloud, the pilots demonstrated new ways to deliver richer, more sustainable, and more engaging media experiences.

Overall, the pilots confirm that NEMO provides a secure, flexible, and scalable framework for orchestrating resources and applications across the IoT–Edge–Cloud continuum. The cross-sector results underline not only the technical feasibility of the NEMO but also its potential to deliver measurable efficiency, sustainability, and user experience benefits. These outcomes will serve as a foundation for further exploitation and sustainability efforts, ensuring that the project’s impact extends beyond its lifetime.

Document name:	NEMO Living Labs use cases evaluation results - Final version				Page:	156 of 195	
Reference:	D5.4	Dissemination:	PU	Version:	6.0	Status:	Final

8 References

- [1] NEMO, "D1.1 - Definition and analysis of use cases and GDPR compliance," HORIZON - 101070118 - NEMO Deliverable Report, 2024.
[D1.1 Definition and analysis of use cases and GDPR compliance](#)
- [2] NEMO, "D1.3 - NEMO meta-architecture components and benchmarking. Final version," HORIZON - 101070118 - NEMO Deliverable Report, 2024.
[NEMO D1.3-NEMO-meta-architecture components and benchmarking.Final-version V1.1.pdf](#)
- [3] NEMO, "D5.2 - Living Labs and Data Management Plan (DMP). Final version," HORIZON - 101070118 - NEMO Deliverable Report, 2024.
[D5.2 Living Labs and Data Management Plan \(DMP\). Final version](#)
- [4] D5.3 NEMO Living Labs use cases evaluation results - Initial version
- [5] D4.2 Advanced NEMO platform & laboratory testing results. Final version
- [6] <https://github.com/voc/voctomix/tree/prod>
- [7] "Nurse Stress Prediction Wearable Sensors." Accessed: Jun. 12, 2025. [Online]. Available: <https://www.kaggle.com/datasets/priyankraval/nurse-stress-prediction-wearable-sensors>
- [8] "MediaPipe," mediapipe. Accessed: Jun.12,2025. [Online]. Available: <https://chuoling.github.io/mediapipe/>
- [9] Ami, T. Basha, and S. Avidan, "Racing bib number recognition," in Proc. BMCV, 2012, pp. 1–10.
- [10] P. Hernández-Carrascosa, A. Penate-Sanchez, J. Lorenzo-Navarro, D. Freire-Obregón, and M. Castrillón-Santana, "TGCRBNW: A dataset for runner bib number detection (and recognition) in the wild," in Proc. 25th Int. Conf. Pattern Recognit. (ICPR), 2021, pp. 9445–9451.
- [11] <https://universe.roboflow.com/hcmus-3p8wh/bib-detection-big-data>
- [12] <http://ufldl.stanford.edu/housenumbers/>
- [13] Leszczuk, M., Hanusiak, M., Farias, M.C.Q. et al. Recent developments in visual quality monitoring by key performance indicators. Multimed Tools Appl 75, 10745–10767 (2016). <https://doi.org/10.1007/s11042-014-2229-2>

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	157 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

Annex I. Requirements Validation

Smart Farming

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Result (Y/N)
SF_01_FR01	SF_01	Functional	The platform must provide access to measurements.	M	API-Access Control-MO-mNCC	Y
SF_01_FR02	SF_01	Functional	The platform must provide options to manage/view sensors/devices	M	MO	Y
SF_01_FR03	SF_01	Functional	The platform must provide options to manage users.	M	IAM	Y
SF_01_FR04	SF_01	Functional	The platform should support ML/FL training and ML model sharing/serving.	M	CFDRL	Y
SF_01_FR05	SF_01	Functional	The platform should provide ML classification accuracy probability.	S	CFDRL	Y
SF_01_FR06	SF_01	Functional	The platform should support automated	S	API-MO-mNCC	

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Result (Y/N)
			aerial spraying.			
SF_01_FR07	SF_01	Functional	The platform should support monitoring of SLOs, e.g. related to energy consumption or CO2 emissions.	S	PPEF	Y
SF_01_FR08	SF_01	Functional	The platform must respect data sovereignty and privacy requirements.	M	PPEF & Access Control	Y
SF_01_FR09	SF_01	Functional	The platform must support collection of monitoring data, such as the weather and plant conditions	M	API-MO-mNCC	Y
SF_01_FR010	SF_01	Functional	The platform must support retrieving photos via drones.	M	API-MO-mNCC	Y
SF_01_FR011	SF_01	Functional	The monitoring devices must support network connectivity.	M	API-mNCC	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Result (Y/N)
SF_01_FR012	SF_01	Functional	The monitoring devices must be able to communicate data and receive control commands from the NEMO platform.	M	LCM / NEMO plugins	Y
SF_01_FR013	SF_01	Functional	The platform should be able to perform alternative scheduling or geographical distribution of smart farming services based on user goals.	S	MO-CFDRL	Y
SF_01_FR014	SF_01	Functional	The Smart Farmer should be able to define strategies for the use of available resources.	S	API, LCM	Y
SF_01_NFR01	SF_01	Non-Functional	The NEMO platform must respect security and privacy requirements.	M	PPEF, IAM, NIM & Access Control	Y
SF_01_NFR02	SF_01	Non-Functional	NEMO should support High Availability features.	S	MO	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Result (Y/N)
SF_01_NFR03	SF_01	Non-Functional	The Smart Agriculture Application of NEMO should be vendor independent.	S	MO-LCM (plugins)	Y
SF_01_NFR04	SF_01	Non-Functional	The NEMO platform should be scalable in the sense of providing additional resources when computationally heavy tasks are initiated.	S	MO-CFDRL	Y

Smart Energy & Mobility

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SEM_1_FR1	1	Functional	The platform must provide access and connectivity to edge and far-edge devices.	M	MO, SEE	SE01_TS1	Connectivity and edge availability test	Y
SEM_1_FR2	1	Functional	The platform must provide a way to gather and store data from the edge and far-edge devices.	M	PPEF, mNCC		Field data collection and SCADA integration logs	Y
SEM_1_FR3	1	Functional	The platform must provide analytics capabilities for later decision-making process.	S	PPEF, CFDRL	SE01_TS1	PMU analyzer outputs, event detection validation	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SEM_1_FR4	1	Functional	The platform should support ML training and ML model serving.	C	FDRL	SE01_TS1	Model simulation on anomaly detection	Y
SEM_1_NFR1	1	Non-Functional	The platform must provide horizontal and vertical migration along with seamless migration toward optimal location.	M	MO	SE01_TS1	Orchestration / migration logs between ASM and RWTH	Y
SEM_2_FR1	2	Functional	Exploratory analysis of available data.	S		SE01_TS2	Time-series visualization and statistical summaries	Y
SEM_2_FR2	2	Functional	The platform must provide access and connectivity to edge and far-edge devices.	M	MO	SE01_TS2	Sensor-to-SCADA data test	Y
SEM_2_FR3	2	Functional	The platform must provide a way to gather and store data from the edge and far-edge devices.	M	PPEF, mNCC	SE01_TS2	MQTT broker flow, database logs	Y
SEM_2_FR4	2	Functional	The platform must provide analytics capabilities for later decision-making process.	S	PPEF, CFDRL	SE01_TS2	EV charging optimization / duration analysis	Y
SEM_2_FR5	2	Functional	The platform should support ML training and ML model serving.	C	CFDRL	SE01_TS2	Forecasting model deployment for parking data	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SEM_2_NFR1	2	Non-Functional	The platform must provide horizontal and vertical migration along with seamless migration toward optimal location.	M	MO	SE01_TS2	Smart Parking workload deployment across edge/cloud	Y
SEM_2_FR6	2	Functional	Deployment of CBD modules, and access to smart charging stations.	M		SE01_TS2	Physical deployment confirmation and log data	Y
SEM_2_FR7	2	Functional	The platform must provide options to manage users and manage users' access to recommendations.	M	IdM	SE01_TS2	Identity management and access logs	Y
SEM_2_FR8	2	Functional	The platform must provide access and connectivity to edge and far-edge devices.	M	MO	SE01_TS2	EV sensor network availability	Y
SEM_2_FR9	2	Functional	The platform must provide a way to gather and store data from the edge and far-edge devices.	M	PPEF, mNCC	SE01_TS2	Parking data ingestion and visualization	Y
SEM_2_FR10	2	Functional	The platform must provide analytics capabilities for later decision-making process.	S	PPEF, CFDR	SE01_TS2	Data dashboards for charger and slot occupancy	Y
SEM_2_FR11	2	Functional	The platform should support ML training and ML model serving.	C	CFDR	SE01_TS2	Parking forecast model validation	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SEM_2_NFR2	2	Non-Functional	The platform must provide horizontal and vertical migration along with seamless migration toward optimal location.	M	IO	SE01_TS2	Urban mobility workload migration test	Y

Smart Media

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR01	SC_01	Functional	The contributors (real spectators) must be authenticated to access the tools and set up a transmission.	M	IAM-AC	SC_01	App Login and DB user access	Y
SC_01_FR02	SC_01	Functional	The broadcaster must be able to define/adjust the media-specific requirements of the transmission.	M	API - mNCC	SC_01	Via REST API, from Production Control to Media Production Engine, we were able to increase/decrease the bitrate of the transmission	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR03	SC_01	Functional	The broadcaster must be able to monitor the signal quality and ensure the transmission to ensure streaming quality.	M	mNCC	SC_01	Data from Video Quality Probe was sent to specific queue in RabbitMQ. Live monitor was placed in Production Control to check if current video stream was above threshold	Y
SC_01_FR04	SC_01	Functional	Several video streams are to be transferred through the cloud/network. Bandwidth requirements must be met accordingly.	M	mNCC - MO	SC_01	Media files transferred	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR05	SC_01	Functional	Control signals (voice and data) and audio/video return channel are to be transferred between the technical director location and the venue via the cloud network.	M	API - mNCC	SC_01	Incoming videos from mobile app entered the cloud network and kubernetes cluster. Production control produced the content, and forwarded to the RTMP server to be broadcasted to users	Y
SC_01_FR06	SC_01	Functional	NEMO must provide the adequate resources to the service provider to map these requirements onto the cloud network and perform accordingly.	M	IAS - LCM	SC_01	This was verified through performance monitoring, resource utilization metrics, and successful completion of stress and load tests during and after deployment. No bottlenecks or performance degradation were observed, confirming that the allocated resources	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
							were adequate.	
SC_01_FR07	SC_01	Functional	The processing of the video streams is achieved automatically by the virtualised compression functions that are part of the Media Production Engine, which are deployed at the edge cloud near the venue.	S	MDT - IMC	SC_01	Media Production Engine was deployed on kubernetes cluster. Three pods were created, to receive at least three streams to later be managed on Production Control	Y
SC_01_FR08	SC_01	Functional	NEMO will be able to allocate and launch the required services /VNFs on a location basis.	M	PPEF	SC_01	Specific cluster for use case was created on OTE premises, connected to the NEMO platform. Using project CI/CD pipelines (FluxCD), services	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
							were deployed to the specific location	
SC_01_FR09	SC_01	Functional	The service provider must be able to chain services/VNFs with the help of a service orchestrator.	M	MO-mNCC	SC_01	NEMO helped with the CI/CD to manage where the pods should be deployed, not only in the specific cluster, but the specific nodes	Y
SC_01_FR10	SC_01	Functional	NEMO applies a central control unit (Cognitive Network Optimization) that is used by the service provider to adjust/adapt the network dynamically according to the specific requirements and conditions.	S	mNCC	SC_01	Not verified	N

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR11	SC_01	Functional	NEMO must be able to monitor and control the network and ensure adherence to QoS levels (bandwidth, average bit rate, round trip delay).	M	PPEF	SC_01	NEMO measured the environment using the subcomponent focused on monitoring aspects. Data is on RabbitMQ as usual	Y
SC_01_FR12	SC_01	Functional	The production of a final stream on-site is realised with the help of the MPE module at the cloud edge. The technical director remotely controls the signal switching at the MPE.	M	API-SEE	SC_01	Content was sent from Greece, and produced remotely on Spain. Production Control GUI was placed on TID laboratories	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR13	SC_01	Functional	Cognitive Service module allows for enrichment of the audio/video stream with additional information like face recognition, image recognition, data fusion, etc.	M	API-CFDRL	SC_01	Module was able to identify runners in the video, and extract the bib number of them, enriching the information of the stream	Y
SC_01_FR14	SC_01	Functional	A media app on a smartphone can be used (by journalists and/or the audience) for acquisition and streaming of audio-visual content into the cloud and make it available	S	API-SEE	SC_01	The android APP was able to receive RTMP stream, authored and unauthored based on the selection of the user.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
			e for the selection from technical director .					
SC_01_FR15	SC_01	Functional	End-to-end network latency (RTT) - It comprises the latency of the whole network path excluding end devices on-site (like the network gateway or HW video coder) <= 50 ms	S	SEE-MO	SC_01	Commercial 5G network was used to connect production cameras to production SW	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR16	SC_01	Functional	Minimum end-to-end connection bandwidth (per stream). Listed here are the required video data rates, for resulting data rates on the network layer add 10 % overhead: - Uncompressed HD (1080i and 1080p): 1.5 Gbit/s and 3.0 Gbit/s; - JPEG2000: 100 Mbit/s; - H.264/AVC-Intra: 25 Mbit/s; - H265/HVC: 15 Mbit/s	M	API-SFF-AC	SC_01	3 videos are analyzed for the purpose of documenting this information: All of them were encoded as H.264, and bitrate was similar between them. 15.8 Mbps was the highest bitrate encountered, while the minimum was 10.7 Mbps. On average, they offered 13.9, 12.2, and 13.3 Mbps respectively.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR17	SC_01	Functional	Error-free/lossless transport of signals with max. packet-loss rate: $< 10^{-12}$	S	mNCC	SC_01	Commercial LTE 5G network was used to connect production cameras to production SW	Y
SC_01_FR18	SC_01	Functional	Max. network jitter/packet delay variation (PDV) < 10 ms	S	mNCC	SC_01	Commercial LTE 5G network was used to connect production cameras to production SW	Y
SC_01_FR19	SC_01	Functional	Classification and prioritisation of audio-video-streams. Due to the high requirements on latency, jitter and bandwidth media streams have to be prioritised in the network.	S	MO	SC_01	From Production Control, we managed as main video those with higher quality, or specific situation in the event	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR20	SC_01	Functional	Max latency of end-to-end signal transport (video, audio and control data) - It comprises the latency of the whole signal path including converting of end devices on-site and media-specific VNFs). Maximum E2E latency one way for video and audio: <= 500 ms Max. E2E latency for return video (one way): <= 500 ms (Typically uses	S	IAS	SC_01	Project integrated commercial 5G network with cloud production SW. End-to-end latency was measured for video and audio streams	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test scenario(s) ID	Verification Method	Result (Y/N)
			less bandwidth because of low-resolution proxy transfer)					
SC_01_FR21	SC_01	Functional	Synchronisation of video and audio signals. GPS Synchronisation of end devices (using black burst, tri-level sync and/or PTP).	M	IAS-LCM	SC_01	Verification based on pilot with 16 users in stadium conditions; GPS data logs show improved accuracy using 5G + Wi-Fi hybrid approach, enhancing synchronization of media capture	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_FR22	SC_01	Functional	The MEC platform and underlying NFVI is required to deploy and run all the needed VNFs. The estimated use of resources is: · High CPU power, preferably new process or generation (≥ 96 cores). · 128 GB RAM · 1 TB Storage SSD · Multiple 10 Gbit/s and 1 Gbit/s interfaces. · GPU processing capability.	M	IAS-LCM	SC_01	Verification was conducted through pre-deployment hardware audits, benchmarking, and resource validation checks to confirm that the infrastructure met or surpassed the specified requirements (≥ 96 cores CPU, 128 GB RAM, 1 TB SSD, multi-Gbit interfaces, and GPU support). As a result OTE's available infrastructure exceeded the estimated needs.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_NFR01	SC_01	Non-Functional	The platform must provide mechanisms for security and data privacy	S	SEE	SC_01	We utilize an OPNsense firewall to manage and secure network traffic, applying strict rulesets to filter unauthorized access and monitor potential threats. In addition, remote access to the infrastructure is strictly controlled via VPN, ensuring that only authenticated users can connect to internal systems securely. This combination helps maintain data privacy and safeguards the platform from external threats.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
SC_01_NFR02	SC_01	Non-Functional	The platform should support high availability deployments	S	IAS	SC_01	High availability is verified through simulated node failures and service disruption tests, ensuring workloads are automatically rescheduled without downtime	The platform runs on a Kubernetes cluster over Proxmox, providing container orchestration, redundancy, and self-healing. Services are automatically rescheduled across nodes to ensure minimal downtime in case of failures.
SC_01_NFR03	SC_01	Non-Functional	Live migration should be done using microservices.	S	IAS	SC_01	Yes, FluxCD allowed to create new deployments as soon as new changes were made on repository	Y
SC_01_NFR04	SC_01	Non-Functional	The platform should detect of network faults or malfunctions before those have any	S	mNCC	SC_01	Network can detect changes on before they are impacting video quality	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
			drastic impact on its performance					
SC_01_NFR05	SC_01	Non-Functional	The use cases depend on 5G network availability at level of five-nines for all communications	S	mNCC	SC_01	Commercial LTE 5G network was used to connect production cameras to production SW	Y

Smart XR

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_01.FR01	XR_01	Functional	Collect user's biometric data	M	API-SEE	XR_01	User's biometric data (heart rate, accelerometer, skin temperature) was collected in real-time during the VR experience using a Samsung Galaxy Smart Watch.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_01.FR_02	XR_01	Functional	ML model for physical and emotional status detection	M	LIS-API	XR_01	A Hybrid CNN-LSTM Machine Learning model was developed and trained to detect user stress levels from the collected biometric data. The model's performance was evaluated, achieving 77% accuracy.	Y
XR_01.FR_03	XR_01	Functional	The solution must have an Application server (REST API) Service for communication between system admin and AR/VR application and UI interface for specifying what events and data to send depending on the state.	S	API-mNCC	XR_01	An Event Server with a REST API was implemented to receive inference results from the ML model. It successfully transmitted JSON-formatted events to subscribed devices (VR app and monitoring PC) with a 100% success rate.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_01.FR_04	XR_01	Functional	Enhance VR app to subscribe and handle events	M	VR App	XR_01	The VR application successfully subscribed to the Event Server, received real-time user stress-level events, and dynamically adapted the in-app experience to mitigate user discomfort.	Y
XR_01.FR_05	XR_01	Functional	Network will support diverse devices (wearables, AR/VR headsets) with different performance (e.g., high throughput, low latency and massive connection densities)	M	mNCC	XR_01	The system successfully integrated a wearable device (Samsung Smart Watch) and a VR headset (Meta Quest 2), demonstrating network support for communication between these diverse IoT devices.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components	Related Test Scenario (s) ID	Verification Method	Result (Y/N)
XR_01.FR06	XR_01	Functional	Interoperability with external systems (i.e. multi sensorial stimuli system)	M	MO-LCM	XR_01	The trial demonstrated successful interoperability between multiple components: a wearable sensor, an edge-based ML inference service, a REST API event server, and a VR application, all working in concert to create a real-time bio-feedback loop.	Y
XR_01.FR07	XR_01	Functional	The platform components involving direct interaction with the end-users should be quick to respond to the users' actions	S	MO-LCM	XR_01	End-to-end latency was measured via system log analysis. The time from data capture by the IoT watch to the VR application's response was consistently low, perceived as instantaneous by the user.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_01_NFR01	XR_01	Non-Functional	The platform must provide mechanisms for security and data privacy	S	MDT-SEE	XR_01	The system was designed for privacy by not storing any personal biometric data. Data from the smartwatch was used only for real-time, anonymous inference and was not retained.	Y
XR_01_NFR02	XR_01	Non-Functional	The platform should support high availability deployments	S	MO-IAS	XR_01	The application workload was deployed on the NEMO platform using Kubernetes, which provides inherent high-availability features like service redundancy and self-healing. The trial was completed successfully without downtime.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_01_NFR03	XR_01	Non-Functional	Live migration should be done using microservices live migration	S	LS	XR_01	A live migration scenario was successfully executed using the NEMO platform's Intent API. An ML training workload was migrated from a CPU-only cluster to a GPU-enabled cluster to satisfy computational requirements without service interruption.	Y
XR_01_NFR04	XR_01	Non-Functional	The platform should detect of network faults or malfunctions before those have any drastic impact on its performance	S	nNCC	XR_01	The platform's monitoring capabilities were active during the trial. The successful completion of the trial with a 100% message delivery rate demonstrated network stability and the platform's ability to operate without performance degradation.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_01_NFR06	XR_01	Non-Functional	The use cases depend on 5G network availability at level of five-nines for all communications	S	Network	XR_01	This requirement was not applicable to this trial, as it utilized local Wi-Fi and wired connections rather than a 5G network.	N
XR_02.FR01	XR_02	Functional	Enhance VR app to subscribe and handle events	M	VR App	XR_02	The Tholos Dome VR application was subscribed to an event server and successfully consumed gesture recognition events to trigger navigation and interaction within the VR environment.	Y
XR_02.FR02	XR_02	Functional	Capture Presenter Voice and Video feed	M	AIoT	XR_02	A video feed of the presenter's hands was successfully captured using a local IoT camera and streamed to the ML model for real-time gesture analysis.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_02.FR03	XR_02	Functional	Identify key phrases and gestures	M	FML	XR_02	An ML model successfully identified a predefined set of hand gestures with 99% accuracy. Voice recognition (key phrases) was deprioritized and not implemented as gestures provided sufficient functionality for the trial.	Y
XR_02.FR04	XR_02	Functional	Application Server (REST API) for communication between system devices and applications	S	Event Server	XR_02	A REST API-based Event Server was implemented to communicate gesture recognition results. Log analysis confirmed a 100% message delivery success rate to all subscribed devices.	Y
XR_02.FR05	XR_02	Functional	UI tool for specifying key phrases to detect	C	AIoT - FML	XR_02	This was not implemented, as the voice recognition feature was deprioritized for the trial.	N

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_02.FR_06	XR_02	Functional	UI tool for specifying what to send to subscribers	C	IoT - Event server	XR_02	The system was designed to publish all recognized gestures to a topic, and all devices subscribed to that topic received the data. A specific UI for filtering was not implemented and the messages that were sent were pre-specified.	N
XR_02.FR_07	XR_02	Functional	Network must support diverse devices (wearables, AR/VR headsets) with different performance (e.g., high throughput, low latency and massive connection densities)	M	mNCC	XR_02	The network successfully supported real-time communication between an IoT camera, an edge processing server, and multiple PCs for the VR application and monitoring, demonstrating support for diverse devices.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_02.FR08	XR_02	Functional	Interoperability with external systems (i.e. multi sensorial stimuli system)	M	VR App - Event Server	XR_02	The trial demonstrated successful interoperability between a camera, an ML inference service, an event server, the main Dome VR application, and an external monitoring PC.	Y
XR_02.FR09	XR_02	Functional	The platform components involving direct interaction with the end-users should be quick to respond to the users' actions	S	VR App - Event Server - FML	XR_02	End-to-end latency was measured. Despite being higher than the ideal target due to on-site network constraints, qualitative feedback from operators confirmed the system was 'perfectly interactive and suitable for navigation', meeting the user-perceived responsiveness goal.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Component(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_02_NFR01	XR_02	Non-Functional	The platform must provide mechanisms for security and data privacy.	S	SE	XR_02	The system adhered to privacy principles by processing a non-personally identifiable video stream (presenter's hands) in real-time without storing the data.	Y
XR_02_NFR02	XR_02	Non-Functional	The platform should support high availability deployments.	S	mOS	XR_02	The application was deployed on the Kubernetes-based NEMO platform, ensuring high availability. The trial was completed without any service interruptions.	Y
XR_02_NFR03	XR_02	Non-Functional	Live migration should be done using microservices live migration.	S	IAS	XR_02	The migration of the ML training micro-service from an edge environment to a cloud-based HPC was tested and verified via system logs, achieving a 100% success rate.	Y

Requirement ID	Use Case ID	Requirement Type	Description	Priority (MoSCoW)	Related Components(s)	Related Test Scenario(s) ID	Verification Method	Result (Y/N)
XR_02_NFR04	XR_02	Non-Functional	The platform should detect of network faults or malfunctions before those have any drastic impact on its performance.	S	mnCC-DLT	XR_02	Despite high network latency at the trial site, the platform operated reliably with 100% message delivery, demonstrating resilience to non-ideal network conditions without impacting core functionality.	Y
XR_02_NFR06	XR_02	Non-Functional	The use cases depend on 5G network availability at level of five-nines for all communications.	S	mnCC	XR_02	This requirement was not applicable, as the trial was conducted over a local network within the museum and did not utilize a 5G network.	N

Annex2. New workload instance deployment in XR usecase

values.yaml

```

nameOverride: ""
# -- option to override the fullname config in the _helpers.tpl
fullnameOverride: ""
imagePullSecrets:
- name: nemo-regcred
env:
  rabbitHost: ""
  rabbitPort: ""
  rabbitUser: ""
  rabbitPassword: ""
  use_tls: false
  logLevel: "INFO"
  prodEventServerUrl: ""
eventServer:
  service:
    type: NodePort
  image: nemometaos/nemo-smart-xr-use-case:v1.2.19
  metrics:
    resources:
      limits:
        memory: 512Mi
        cpu: 500m
gestureParser:
  enabled: false
  image: nemometaos/nemo-smart-xr-use-case:v1.2.19
  metrics:
    resources:
      limits:
        memory: 512Mi
        cpu: 500m
        ephemeral-storage: "500Mi"
gestureReceiver:
  enabled: false
  image: nemometaos/nemo-smart-xr-use-case:v1.2.19
  metrics:
    resources:
      limits:
        memory: 512Mi
        cpu: 500m
        ephemeral-storage: "500Mi"
stressParser:
  enabled: true
  image: nemometaos/nemo-smart-xr-use-case:v1.2.19
  metrics:
    resources:

```

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	191 of 195
Reference:	D5.4	Dissemination:	PU
	Version:	6.0	Status: Final

limits:	
memory:	512Mi
cpu:	500m
ephemeral-storage:	"500Mi"
stressReceiver:	
enabled:	true
image:	nemometaos/nemo-smart-xr-use-case:v1.2.19
metrics:	
resources:	
limits:	
memory:	512Mi
cpu:	500m
ephemeral-storage:	"500Mi"
celeryBeat:	
image:	nemometaos/nemo-smart-xr-use-case:v1.2.19
metrics:	
resources:	
limits:	
memory:	512Mi
cpu:	50m
ephemeral-storage:	"500Mi"
inferenceWorker:	
image:	nemometaos/nemo-smart-xr-use-case:v1.2.19
metrics:	
resources:	
limits:	
memory:	800Mi
cpu:	1000m
ephemeral-storage:	"500Mi"
modelWorker:	
image:	nemometaos/nemo-smart-xr-use-case:v1.2.19
metrics:	
resources:	
limits:	
memory:	800Mi
cpu:	1000m
ephemeral-storage:	"500Mi"
storeDataWorker:	
image:	nemometaos/nemo-smart-xr-use-case:v1.2.19
metrics:	
resources:	
limits:	
memory:	800Mi
cpu:	1000m
ephemeral-storage:	"500Mi"
mlPvc:	
pvc:	
storage:	"3Gi"
storageClass:	
enabled:	true
name:	"rook-cephfs"
retrainStressModel:	
image:	nemometaos/nemo-smart-xr-stress:v1.0.6
#	Preprocess params

Document name:	NEMO Living Labs use cases evaluation results - Final version	Page:	192 of 195
Reference:	D5.4	Dissemination:	PU
		Version:	6.0
		Status:	Final

labeledPercentage:		0.10
heartRateThreshold:		60
#	Split	params
testSize:		0.2
randomState:		42
numClasses:		3
#	Training	params
epochs:		100
batchSize:		512
validationSplit:		0.2
patienceEarlyStop:		5
lrFactor:		0.3
patienceReduceLr:		3
learningRate:		1e-3
minLearningRate:		1e-5
#		Pseudo-labeling
pseudoLabellingIterations:		3
confidenceThreshold:		0.95
sampleCap:		0.25
pseudoTrainSplit:		0.70
pseudoValSplit:		0.15
#	Model	Architecture
layer1Units:		512
layer2Units:		256
layer3Units:		128
layer4Units:		64
activation:		relu
finalActivation:		softmax
dropout1:		0.4
dropout2:		0.4
dropout3:		0.3
dropout4:		0.3
kernelRegularizer:		0.02
#	MinIO	Configuration
minioEndpoint:		"minio.smart-xr.platform.meta-os.eu"
secure:		true
accessKey:		"*****"
secretKey:		"*****"
bucketName:		"*****"
#	Extra	parameters
retrain:		true
useGpu:		true
deviceId:		"2f2966f0f94b7537"
metrics:		
resources:		
limits:		
memory:		512Mi
cpu:		500m
ephemeral-storage:		"500Mi"
nvidia.com/gpu:		1
busybox:		
metrics:		
resources:		
limits:		

memory:	512Mi
cpu:	50m
postgresql:	
auth:	
username:	*****
existingSecret:	{{ .Release.Name }}-postgres-secret"
database:	*****
service:	
ports:	
postgresql:	"5432"
metrics:	
enabled:	true
resources:	
requests:	
memory:	512Mi
cpu:	250m
limits:	
memory:	800Mi
cpu:	500m
influxdb:	
auth:	
enabled:	true
existingSecret:	"smart-xr-influxdb-secret"
username:	
password:	"nemo"
bucket:	"smart_xr_bucket"
influxdb:	
resources:	
requests:	
memory:	512Mi
cpu:	250m
limits:	
memory:	800Mi
cpu:	500m
service:	
type:	NodePort
serviceAccount:	
create:	false
networkPolicy:	
enabled:	false
redis:	
architecture:	standalone
networkPolicy:	
enabled:	false
auth:	
enabled:	false
master:	
service:	
type:	NodePort
persistence:	
enabled:	false
resources:	
requests:	
memory:	256Mi

cpu:	100m
limits:	
memory:	800Mi
cpu: 200m	

Pending EC approval