Ref. Ares(2025)2064303 - 14/03/2025

NEMO

Next Generation Meta Operating System

D5.3 NEMO Living Labs use cases evaluation results - Initial version

Document Identification			
Status	Final	Due Date	28/02/2025
Version	13.0	Submission Date	14/03/2025

Related WP	WP5	**Document Reference**	
Related Deliverable(s)	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**	Ilias Seitanidis (SYN) Jonathan Klimt (RWTH)

Keywords:
NEMO, trial, living lab, pilot, validation, intermediate results, DMP, data management

(*) Dissemination level: **(PU)** Public, fully open, e.g., web (Deliverables flagged as public will be automatically published in CORDIS project's page). **(SEN)** Sensitive, limited under the conditions of the Grant Agreement. **(Classified EU-R)** EU RESTRICTED under the Commission Decision No2015/444. **(Classified EU-C)** EU CONFIDENTIAL under the Commission Decision No2015/444. **(Classified EU-S)** EU SECRET under the Commission Decision No2015/444.

Document Information

List of Contributors	
Name	**Partner**
Mircea Vasile	SIM
Andreea Paunescu	SIM
Costas Vrioni	MAG
Mika Skarp	CMC
Sonja Waechter	CONTI
Antonis Gonos	ESOFT
Prashanth Kumar Pedholla	ASM
Rizzi Gianluca	WIND
Mohammad Ghoreishi	ASM
Dimitrios Christopoulos	FHW
George Sofianopoulos	FHW
Theo Kakardakos	NOVO
Francesco Bellesini	EMOT
Ilias Seitanidis	SYN
Matija Cankar	COMS
Denis Sodin	COMS
Javier Serrano	UPM
Alberto del Rio	UPM

Document History			
Version	**Date**	**Change editors**	**Changes**
1.0	01/11/2024	Prashanth Kumar. P	Table of Contents
2.0	14/11/2024	Dimitrios Christopoulos	Introduction and Trial Site descriptions.
3.0	20/01/2025	Prashanth Kumar. P	TOC changes for Smart Energy & Smart Mobility Trial
4.0	21/02/2025	A. Gonos (ESOFT), Ilias Seitanidis (SYN)	Contribution to Smart Farming Trial
5.0	26/02/2025	Mohammad. G (ASM), Matija Cankar (COMS), Denis Sodin (COMS)	Contribution to Smart Energy/Mobility Trial
6.0	28/02/2025	Dimitrios Christopoulos (FHW), Costas Vrioni (MAG)	Contribution to Smart Media & XR Trial
7.0	28/02/2025	Mircea Vasile, A. Paunescu (SIM), S. Waechter (CONTI)	Contribution to Smart Industry Trial
8.0	28/02/2025	F. Bellesini (EMOT)	Contribution to Smart Energy/Mobility Trial

Document History			
Version	**Date**	**Change editors**	**Changes**
9.0	03/03/2025	Alberto del Rio (UPM), Javier Serrano (UPM)	Contribution to Smart Media & XR Trial
10.0	05/03/2025	Ilias Seitanidis (SYN)	Peer review
11.0	07/03/2025	Jonathan Klimt (RWTH)	Peer review
12.0	12/03/2025	Prashanth Kumar. P (ASM), Dimitrios Christopoulos (FHW)	Addressing peer review comments, Consolidated version
13.0	14/03/2025	ATOS	Quality check & submission to EC

Quality Control		
Role	**Name (Organization)**	**Approval Date**
Deliverable leader	P. Prashanth Kumar (ASM)	14/03/2025
Quality manager	R. Valle Soriano (ATOS)	14/03/2025
Project Coordinator	E. Pages (ATOS)	14/03/2025
Technical Coordinator	Ilias Seitanidis (SYN)	14/03/2025

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

Abbreviation / acronym	Description
3GPP	3rd Generation Partnership Project
AAA	Authentication, Authorization, and Accounting
aAGV	automated Autonomous Guided Vehicle
ADAS	Advanced Driver Assistance System
AF	Application Function
AGV	Automated Guided Vehicle
AGLV	Automated Guided Land Vehicle
AI	Artificial Intelligence
API	Application Programming Interface
AR	Augmented Reality
ASBS	Automated Sorting and Booking Station
AV	Audiovisual
CDN	Content Delivery Network
CFDRL	Cybersecure Federated Deep Reinforcement Learning
CMDT	Cybersecure Microservices' Digital Twin
Cobot	Collaborative Robot
CPO	Charging Point Operator
CPU	Central Processing Unit
DLT	Distributed Ledger Technology
DMP	Data Management Plan
DoA	Description of Action
DR	Demand Response
DSO	Distribution System Operator
Dx.y	Deliverable number y belonging to WP x
EC	European Commission
EDA	Electrodermal Activity
EV	Electric vehicle
FAIR	Findability, Accessibility, Interoperability, and Reusability
FL	Federated Learning
FML	Federated Machine Learning
FOTA	Firmware over the Air
GDPR	General Data Protection Regulation
GPS	Global Positioning System
GPU	Graphics Processing Unit
HMD	Head Mounted Display
HPC	High Performance Computing
HR	Heart Rate

Abbreviation / acronym	Description
HTTP	Hypertext Transfer Protocol
HW	Hardware
IaaS	Infrastructure-as-a-Service
IDM	Identity Management
IoT	Internet of Things
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
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
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

Abbreviation / acronym	Description
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
XR	Extended Reality

Executive Summary

The NEMO project aims to revolutionize the management and orchestration of applications and resources across the IoT-edge-cloud continuum by introducing a meta-Operating System (meta-OS). Through this approach, NEMO provides a unique framework for businesses to optimize service deployment and resource management across different verticals.

To validate these capabilities, NEMO is piloted in five verticals—Smart Farming, Smart Energy, Smart Industry, and Smart Media—across nine use cases. These pilots provide real-world environments with diverse requirements to test, evaluate, and refine the NEMO platform.

Deliverable D5.3 - NEMO Living Labs Use Cases Evaluation Results-Initial Version, presents the outcomes of the trials, focusing on the development and deployment of applications and services within the Living Labs. It provides an evaluation of the initial implementation efforts, documenting:

- Technical developments carried out by each pilot, including the design, installation, and configuration of hardware and software components.
- Implementation of trial-specific applications and services, which will be integrated into the NEMO framework for validation.
- Intermediate results, demonstrating the feasibility and performance of key functionalities before full-scale NEMO integration.

A key aspect of this deliverable is to assess the technical advancements within each pilot while aligning them with NEMO's objectives. These developments will serve as the foundation for the integration and validation phase, ensuring that NEMO provides tangible benefits to application providers, infrastructure owners, and end-users.

Additionally, an update to the project's Data Management Plan (DMP) has been conducted, based on feedback from Consortium members. The updated DMP incorporates more detailed questionnaires, gathering input on ethics, used and generated datasets, and data management processes, compared to the second version of the DMP, which was submitted in Month 18.

1 Introduction

The NEMO project aims to revolutionize the way computing workloads, network resources, and applications are orchestrated across highly distributed environments, spanning IoT, edge, and cloud. In modern digital ecosystems, businesses and industries rely on hyper-distributed applications that operate across diverse infrastructures with varying levels of computational demands, data exchange needs, and security requirements. Managing such complex environments efficiently, while ensuring cybersecurity, flexibility, and interoperability, remains a significant challenge.

To address this, NEMO introduces a meta-Operating System (meta-OS), which provides a cybersecure, flexible, and efficient orchestration framework for applications and resources operating across different layers of the cloud-edge-IoT continuum. The NEMO meta-OS is being validated through five application verticals—Smart Farming, Smart Energy, Smart Industry, Smart Media and XR which include a total of nine use cases. Each of these use cases presents unique challenges related to computational performance, networking, security, data privacy, and multi-stakeholder interactions. These use cases serve as testbeds for evaluating NEMO's ability to seamlessly integrate, manage, and optimize application execution across different infrastructure layers while ensuring compliance with data protection and privacy regulations such as GDPR.

1.1 Objectives of the document

The successful validation of NEMO innovations requires a structured approach to development, integration, and testing. This deliverable, D5.3 - NEMO Living Labs Use Cases Evaluation Results (Initial Version), presents the progress made in the five Living Lab trials—Smart Farming, Smart Energy & Mobility, Smart Industry, and Smart Media & XR by documenting the advancements in application and service development, their alignment with NEMO functionalities, and preliminary validation efforts.

This document provides insights into the status of use case-specific applications and services, tracking their integration into the NEMO meta-OS and evaluating their technical feasibility through initial testing. The primary focus is on demonstrating how NEMO's capabilities, such as intent-based orchestration, workload migration, and cybersecurity enhancements, can support and optimize different applications across IoT, edge, and cloud infrastructures.

Additionally, this deliverable provides an update to the Data Management Plan (DMP), ensuring that the handling of data, ethics compliance, and GDPR considerations align with the project's evolving needs. The updated DMP is based on feedback from consortium members and incorporates refinements to ethics and data management processes following previous evaluations in D5.1 [4] and D5.2 [5].

1.2 Connection to Other Work Packages and Deliverables

This document, D5.3 – NEMO Living Labs Use Cases Evaluation Results (Initial Version), is the third deliverable in Work Package 5 (WP5) and serves as an intermediate evaluation of the progress made in the NEMO Living Labs. It builds upon the preparatory activities outlined in D5.2 [5], providing updates on pilot-specific application and service developments, their alignment with NEMO functionalities, and the early validation efforts conducted in each trial.

This deliverable is linked to D1.3 [3] – Refinement of Functional and Non-Functional Requirements of the NEMO meta-OS. Following the pilot preparation activities and initial technical validation, the use case definitions and specifications were revisited, leading to refinements in the functional and non-functional requirements of the NEMO platform. These refinements directly influence the validation scenarios presented in D5.3

The final outcomes and validation results from the NEMO Living Labs will be consolidated in D5.4 – NEMO Living Labs Use Cases Evaluation Results (Final Version), due in M36. This forthcoming deliverable will document the outcomes from the fully integrated applications and services, further assessing NEMO's impact across its application verticals.

1.3 Document Structure and Overview

The rest of this document is structured in the following manner:

- Sections 2 to 5 present the intermediate results and validation efforts from the five trials: Smart Farming, Smart Energy & Mobility, Smart Industry, and Smart Media & XR. Each chapter documents the progress made in each pilot, updates on technical validation, use case sequence diagrams, and insights into the integration of pilot-specific applications with the NEMO meta-OS.
- Section 6 outlines the updated Data Management Plan (DMP), reflecting refinements made in response to feedback from consortium partners. This section highlights any changes in data collection, management, and compliance aspects following the ongoing pilot activities.
- Section 7 presents the conclusions and summarizes the next steps in the pilot implementation process.
- Detailed information on the datasets identified in the NEMO use cases are provided in the Annex: Data Inventory and FAIR Data Considerations. This annex outlines key attributes of each dataset, including its source, format, accessibility, and alignment with FAIR (Findability, Accessibility, Interoperability, and Reusability) principles.

The structure of this deliverable follows a systematic approach to present intermediate results of the trials:

Trial Site Description and Updates: This section provides an overview of each pilot site, emphasizing new updates since the last deliverable. If no significant changes have occurred, a brief summary of the existing infrastructure and objectives is included.
Technical Validation: This section documents the updated test scenarios that assess how NEMO functionalities integrate into the pilots. These scenarios focus on how NEMO components will be piloted within the use cases.
Use Case Sequence Diagrams: To visualize the interactions between NEMO and the different pilot components, updated sequence diagrams are included. These illustrate communication flows, data exchanges, and decision-making processes within each use case. Only new updates since the last deliverable are included here.
Timeline of Activities: This section presents an updated roadmap for the pilots, detailing key milestones, integration efforts, and deployment timelines to ensure alignment with the project's overall objectives.
Intermediate Results: As NEMO progresses, this section summarizes intermediate findings, documenting application/service development progress, results. These results will serve as a basis for the final evaluation phase in D5.4 - NEMO Living Labs Use Cases Evaluation Results (Final Version).

2 Smart farming Trial

2.1 Introduction and Use case Applications

Agriculture remains a crucial domain for human survival and economic stability, providing food worldwide. With the increasing demand for enhanced production both in volume and quality, several actions have to be taken in order to assist the farmer in this goal. In this context, NEMO through the Smart Farming pilot is aiming to enhance the Agriculture 4.0 functionalities needed by the farmer through its platform capabilities.

The test environment is in Agia Sofia estate, an organic olive farm located on the Myrtoan Sea and expanding over 200ha. The farm is using state-of-the-art processes for bottling olive oil focusing on the quality perspective. In this context, the use of smart farming techniques within the NEMO framework targets to provide high quality standards during the production process as well. The two scenarios under investigation, Aerial Precision Bio-Spraying and Terrestrial weed management, aim to improve the traditional farming procedures used focusing on the efficiency and quality of the final product.

In the smart farming pilot two applications are implemented, relevant to validating NEMO, as follows.

SF_01: Aerial Precision Bio-Spraying: This Use Case targets to protect olive trees from insects and diseases, such as the olive fruit fly. This is achieved by conducting targeted aerial spraying on the affected trees with the use of ML and UAV technologies. This procedure takes place in the entire IoT-edge-cloud continuum by utilizing the available resources demonstrating the benefits of the NEMO meta-OS framework.

SF_02: Terrestrial weed management. This Use Case aims to the autonomous weed mowing. This is achieved by deploying autonomous robots that are capable of detecting obstacles such as trees, humans, etc. The process of detecting and classifying the detected object is using ML components that can be deployed across the IoT-edge-cloud continuum with several benefits in the domains of energy and finance.

2.2 SF_01 Aerial Precision Bio-Spraying

2.2.1 Trial Site Description Updates

In this section an overview of the advanced precision agriculture solution utilizing drone technology for targeted bio-spraying of olive trees will be presented. The process begins with a drone flight, which is conducted within a predefined territory defined by two corner coordinates that mark the field's boundaries. These coordinates establish the operational limits for the drone, ensuring that image acquisition is confined within the designated area, maximizing efficiency and preventing unnecessary data collection outside the target zone. During the flight high-resolution images are captured with optimal overlap to ensure comprehensive coverage of the area. This ensures that the subsequent processing steps can work with detailed and redundant data, minimizing gaps and errors in the analysis. Once the images are collected, they undergo georeferencing and transformation into orthophoto maps**,** creating an accurate, spatially consistent representation of the surveyed region. These orthophoto maps serve as the foundational dataset for the next step, which involves segmentation techniques applied to extract the exact locations of olive trees**.** Through machine learning algorithms, individual trees are identified and distinguished from other land features, ensuring precise targeting.

Once the tree locations are extracted, they are then used as waypoints for a custom waypoint navigation mission, where each identified tree is assigned a specific GPS coordinate. This step ensures that the spraying drone can navigate efficiently, reducing unnecessary flight time and optimizing the use of resources. After defining the waypoints, the system proceeds with a custom waypoint mission for micro-adjustments and precision bio-spraying, where the drone adapts its path dynamically based on captured images and real-time olive-trees segmentation results. This level of fine-tuned control allows for the precise application of treatments, minimizing waste and improving efficiency. The process is cyclical, meaning it can be continuously refined with each mission, leveraging new data to enhance accuracy and effectiveness over time. By integrating aerial imaging, geospatial analysis, and autonomous drone navigation, this workflow exemplifies a modern approach to sustainable and efficient precision agriculture.

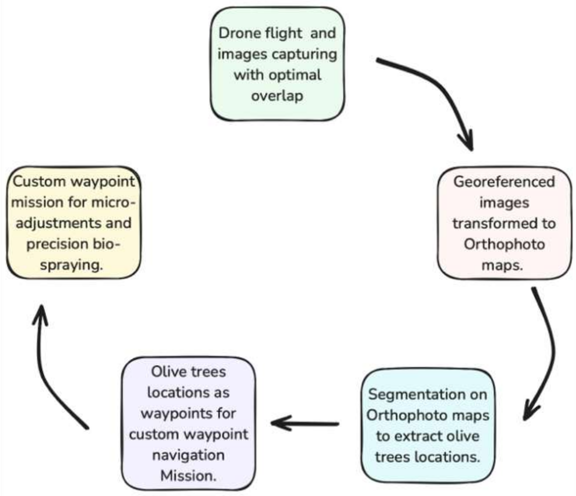


Figure 1 Schematic of the process followed in the Aerial Precision Bio-Spraying use case

Orthophoto Maps

The creation of Orthophoto maps using OpenDroneMap[1] (ODM) involves a series of photogrammetric processing steps that transform individual drone-captured images into a geospatially accurate mosaic. The process begins with image alignment, where feature matching techniques such as SIFT (Scale-Invariant Feature Transform [1]) or ORB (Oriented FAST and Rotated BRIEF [2]) detect common key points across overlapping images. These matched points allow the software to triangulate the position of each image relative to the others and the ground. This computationally intensive procedure takes place in cloud infrastructure.

Once alignment is complete, the images are georeferenced using metadata from the drone's onboard GPS and IMU (Inertial Measurement Unit) sensors. The transformation into an Orthophoto requires correcting distortions caused by the camera's perspective and terrain elevation changes, a process known as orthorectification. ODM uses Structure-from-Motion (SfM) [3] and Multi-View Stereo (MVS) algorithms to generate a sparse point cloud, which is then densified into a 3D model before being flattened into a high-resolution GeoTIFF[2] orthophoto.

A crucial factor in Orthophoto quality is the image overlapping. The general industry standard recommends at least 70-80% front overlap (along the flight path) and 60-70% side overlap (between adjacent flight lines) to ensure accurate reconstruction. Insufficient overlap can lead to gaps, poor

1 https://www.opendronemap.org/
2 https://docs.ogc.org/is/19-008r4/19-008r4.html

alignment, or distortion in the final map. OpenDroneMap optimizes this process by leveraging progressive bundle adjustment, a method that refines camera positions iteratively to minimize projection errors.

Once processing is complete, the final Orthophoto map is stored in GeoTIFF format, where each pixel is assigned precise GPS coordinates. This allows seamless integration into GIS software for further analysis, such as segmentation, classification, or precision agriculture applications. By leveraging ODM's open-source capabilities, users can achieve high-accuracy mapping without relying on expensive proprietary photogrammetry software.

Figure 2 Isolation of the area of interest based on the Orthophoto maps

Custom Waypoint navigation mission, Microservices Approach and NEMO integration

The custom waypoint mission system, Figure 1, follows a microservices-based architecture, where each processing stage—waypoint navigation, image capturing, segmentation, center extraction, and final navigation—is implemented as an independent service pod, Figure 2. This modular approach enhances scalability, fault tolerance, and flexibility, as each microservice can be deployed, updated, or scaled independently based on workload demands. By decoupling functionalities into distinct services, the system ensures that failures in one component do not affect the entire pipeline, making it more resilient and maintainable.

The system operates as a triggered task pipeline, where each task is activated upon reaching a specific waypoint, ensuring a seamless and automated mission execution. The communication between the components has the following structured flow:

1. **Waypoint Navigation & Task Triggering:**
   - The Waypoints Navigation Service Pod sends commands to the drone hardware to move to the next waypoint.

- Once the drone reaches the waypoint, it triggers the next processing stage via an API call.

2. **Image Capturing:**
   - The Image Capture Service Pod receives a trigger and commands the drone's camera (via Jetson Orin Nano) to capture an image.
   - The captured image is then stored in MinIO cloud Storage.
3. **Image Segmentation & Mask Generation:**
   - The Image Segmentation Service Pod fetches the captured image from the storage and applies a pretrained YOLO model for olive tree segmentation stored in MinIO cloud storage to identify and segment olive trees.
   - The output of this step is a predicted segmentation mask, which is stored back in MinIO cloud Storage.
4. **Center Coordinates Extraction:**
   - The Center Extraction Service Pod retrieves the predicted segmentation mask and extracts the center coordinates of the segmented object.
   - These coordinates are then stored in MinIO cloud Storage for further use.
5. **Navigation to Center Coordinates:**
   - The Navigate to Center Service Pod retrieves the extracted center coordinates and commands the drone to move precisely to that location before resuming waypoint navigation.
   - This ensures fine-grained navigation, allowing for precision actions such as targeted bio-spraying.

In the workflow presented in Figure 3, the Context Broker plays a pivotal role orchestrating the different services in terms of communication and synchronizations. It is upon the Context Broker to ensure that each task triggers the next step automatically in a streamlined sequence. The NEMO Meta-OS dynamically manages and distributes computational loads between the UAV and the edge nodes, or the edge and the cloud nodes based on the current system usage and available resources. Computationally expensive tasks, such as image segmentation using deep learning models (YOLO), are offloaded to more powerful cloud or edge nodes, while lightweight tasks, like waypoint navigation and image capturing, remain on the drone's onboard Jetson Orin Nano. This real-time load migration prevents performance bottlenecks, ensures optimal resource utilization, and minimizes energy consumption by running tasks on the most suitable hardware. To achieve this, K3s- a lightweight Kubernetes distribution-runs on the Jetson Nano, transforming it into a K3s node that is incorporated as part of the NEMO cluster. This allows the drone to locally manage its own processing tasks while offloading the energy and computationally intensive tasks to more appropriate infrastructure based on NEMO's meta-orchestrator.

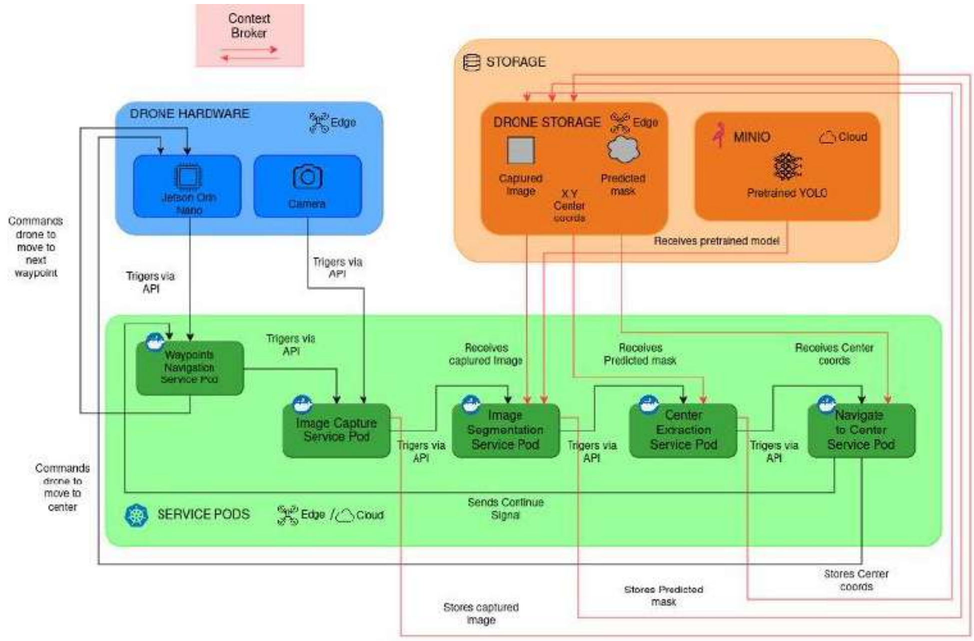
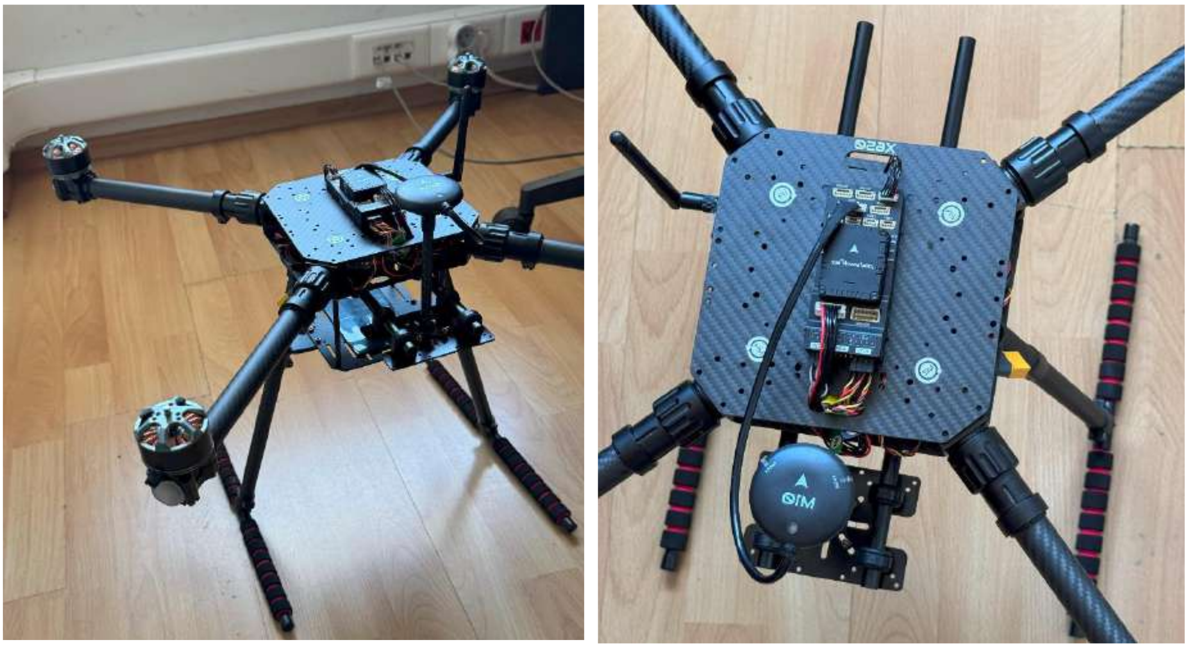


Figure 3 Aerial Precision Bio-Spraying Architecture

Use Case Equipment

In the previous section the advancements in the software developed for this particular use case were described. In this Section the two main Hardware platforms of the SF_01 use case will be presented, the drone responsible for the image acquisition and the AI/ML processing board. The Holybro X650 Dev Kit[3], Figure 4, is a versatile and open-source development platform designed for researchers and developers working on autonomous UAV applications. It features a carbon fiber foldable frame with a 650mm wheelbase, making it portable yet robust for various testing environments. The drone is powered by T-Motor KV330 motors and Tekko32 F4 45A ESCs, ensuring stable and efficient flight performance. It comes with the Pixhawk 6X flight controller, running PX4 open-source firmware, providing extensive support for advanced navigation, obstacle avoidance, and autonomous flight missions. The kit includes GPS module, telemetry radios, and multiple connectivity options such as CAN, UART, and I2C, enabling seamless integration with sensors and additional peripherals. With support for ROS 2 [4] and MAVSDK[4], the X650 is an ideal platform for AI-driven aerial robotics, precision agriculture, and research applications.

[3] https://holybro.com/products/x650-development-kit
[4] https://mavsdk.mavlink.io/main/en/index.html

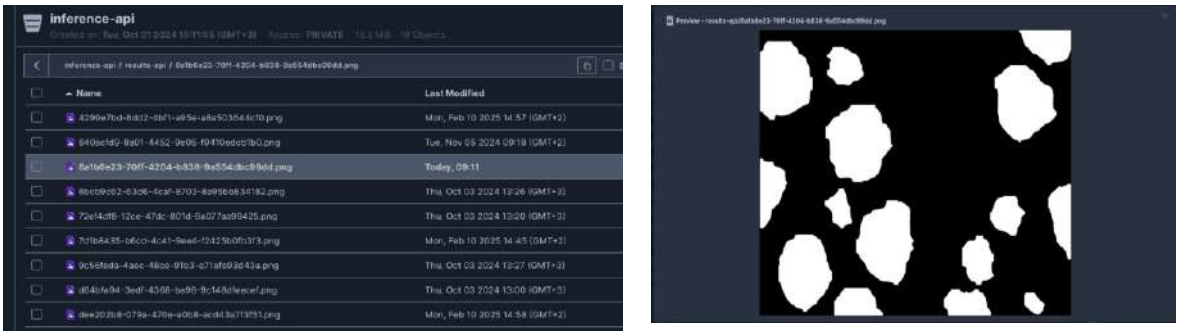


(i) Side view of the drone (ii) Top view of the drone

Figure 4 Drone deployed in the Aerial Precision Bio-Spraying use case

As a companion computer on the drone to host the ML algorithms an Nvidia Jetson Orin Nano 8GB, Figure 5, is selected based on the NVIDIA Ampere architecture with 1024 CUDA cores and 32 tensor cores offering AI performance of 76 INT8 TOPS. In the Jetson, a k3s cluster is installed and configured with one node. As future work it is planned to integrate this k3s instance with the NEMO Kubernetes infrastructure. All the modules described in the previous section have been deployed and tested in the K3s on Jetson. The Image Segmentation Service Pod that loads a pretrained olive tree segmentation model and an image from the MinIO bucket, performs the segmentation and stores the segmented images back to the MinIO bucket, Figure 6 using a POST REST request, Figure 7.

Figure 5 Jetson Orin used in the use case for the inference part



(i) MinIO inference results data storage (ii) Inference result image

Figure 6 MinIO data storage housing the inference results

```
/ # curl -X POST \
>   http://inference-api-service/inference \
>   -H 'Content-Type: application/json' \
>   -d '{"image_minio_url": "minio://inference-api/images/IMG_240417_072655_0040_RGB.JPG"}'
{"result_minio_url":"minio://inference-api/results-api/6a1b6e23-70ff-4204-b838-9a554dbc99dd.png"}/ #
```

Figure 7 REST API for inference result retrieval

2.2.2 Technical validation

The Smart Farming trial's technical validation will be carried out through the test scenario detailed in the table below, addressing SF_01

SF01_Test_Scenario_1	
Scenario ID	SF01_Test_Scenario_1 App owner as application provider App user as resource provider and app user NEMO-offered Federated Learning (FL) services (PaaS) Local training in own resources only (private IaaS, SaaS)
Objective	FL will be used for training image segmentation ML models used in Smart Farming application for olive tree identification by UAV. Local trainings, pertaining to different app users, i.e., Smart Farmers, will be performed in clusters of different administrative domains. Secure communication of ML model parameters between participants and aggregator nodes will be achieved via privacy preserving FL techniques and secure micro-slice creation through mNCC. Trained model will be stored and served by NEMO. Execution of training workloads will be constrained in user-defined clusters. Also, tokens will be calculated for the provision or use of NEMO apps and resources.
Features to be tested	MLOps (FL, model storage, model sharing) Meta-Orchestrator, PPEF, CFDRL (Observability, Workload deployment & migration, limit execution within a cluster set) CMDT (workload discovery) Intent-based API (app and resource selection) LCM (app deployment and LCM visualization) IdM & Access Control (users) MOCA (tokens' calculation)

SF01_Test_Scenario_1	
	mNCC (secure micro-slice creation)
Requirements addressed[5]	SF_01_FR01, SF_01_FR02, SF_01_FR03, SF_01_FR04, SF_01_FR07, SF_01_FR08, SF_01_FR011, SF_01_FR012, SF_01_FR014, SF_01_NFR01, SF_01_NFR02, SF_01_NFR03
KPIs[2]	KPI_SF_01_1, KPI_SF_01_2
Prerequisites	The NEMO platform should be installed and configured. At least 3 users are registered and signed in with relevant access. One of them (let assume they are called OWNER) is acting as Smart Farming Application Owner (SFAO), who uploads their Smart Farming App in NEMO, available for deployment by third parties and has access as NEMO partner (see NEMO roles in D1.2 [2]). The rest of the users (let assume they are USER_A and USER_B) should be registered both as NEMO cluster providers, having provided their resources in NEMO, and as consumers, wishing to deploy the Smart Farming app in their resources only through NEMO. Thus, individual clusters in different administrative domains should be available for each of these cluster providers, aimed for simulating the resources (at least 1 edge/cloud server and 1 UAV) of the Smart Farming application users. FL training should be supported for at least one ML task in the Smart Farming application. The clusters should be configured appropriately, in order to be integrated into NEMO resources and be able to run local training tasks as FL participants. NEMO should be configured, in order to allow restricting FL training tasks in user-defined resources. The OWNER should be able to initiate an FL training task. Also, 1 user is registered as meta-OS Provider (ADMIN).
Test steps	<u>Resource onboarding</u> 1. USER_A signs in the NEMO dashboard and accesses the resource onboarding functionality. 2. USER_A selects the option to add their resources into NEMO and provides relevant info. 3. USER_A is notified about the result through the dashboard. 4. USER_A is informed about their credits. The same process is followed by USER_B. <u>Workload registration</u> 1. The OWNER signs in NEMO as meta-OS consumer. 2. The OWNER uploads their application into NEMO and is notified about the result. 3. As soon as the upload is successful, it will be available for third parties to deploy and use it. 4. The OWNER enables and configures the plugin that allows FL training to be initiated for an ML task in their application. 5. The OWNER visualizes lifecycle data and credits through the NEMO dashboard. <u>Workload execution</u> 1. USER_A accesses the workload execution functionality in the NEMO dashboard.

[5] Based on naming adopted in D1.1.

SF01_Test_Scenario_1	
	2. USER_A selects the Smart Farming app to be executed in their resources only. 3. USER_A enables FL training for the app's ML tasks. The same steps are followed by USER_B. FL training 1. The OWNER sends a request for FL training for their app. 2. The FL training process is executed with the user-defined parameters (rounds, epochs, FL participants, etc.) 3. The training process is completed, and the aggregate model is stored. 4. USER_A and USER_B are notified about the newly trained model. Model sharing & deployment 1. USER_A sends a request through the NEMO dashboard to deploy the new model on their UAV. 2. USER_A checks that new models are deployed on their devices through the dashboard.
Success state	The training process is completed successfully, and OWNER, USER_A and USER_B get notified accordingly. The final ML model is stored in NEMO resources. The final ML model gets deployed on the selected UAV.
Failure state	The training process has not been completed; and/or The final ML model is not accessible; and/or The final model has not been deployed on the UAV.
Responsible for testing and implementation	ESOFT, SYN
Risks	No risks identified so far.

Table 1 Test Scenario SF 01

2.2.3 Use case diagrams

The use case diagrams presented in deliverable D5.2 [5] provide a detailed description of how the several entities interact with each other. These diagrams are foreseen to be updated in deliverable D5.4 with the finalization of the piloting use case as new interactions may be needed as the project progresses.

2.2.4 Timeline of activities

Phase	Estimated start date	Estimated end date	Notes
Trial set-up and equipment procurement	M5	M18	Completed
Initial implementation and validation	M19	M30	Completed
Final implementation and validation	M31	M36	Validation is mainly on the basis of the final integrated prototype with NEMO platform

Table 2 Timeline of activities for SF 01

2.2.5 Intermediate Results

A custom dataset has been carefully curated for olive tree segmentation using drone imagery, ensuring a robust and diverse representation of different environmental and operational conditions. It consists of a total of 938 images, split into 655 for training, 141 for validation, and 142 for testing, making it well-structured for deep learning model development and evaluation. The dataset is object-dense**,** with a total of 9,025 bounding boxes, averaging 12.76 annotations per image across all splits. However, when broken down by subset, the training set averages 9.54 trees per image, while the validation and test sets maintain a slightly higher density of 9.86 and 9.75 trees per image, respectively**.**

A key strength of this dataset lies in its diverse environmental conditions, as it integrates images captured during various weather conditions, such as sunny, cloudy, and afternoon flights. This variation in lighting and shadow conditions enhances model generalization, helping to prevent biases toward specific illumination patterns. Additionally, multiple flight heights were incorporated, leading to variations in tree size and scale within the images. This ensures that the dataset is well-suited for models that need to generalize across different resolutions and perspectives, making it particularly valuable for real-world drone applications where altitude changes can significantly affect object appearance.

The dataset was collected using a Parrot Sequoia multispectral camera, known for its high accuracy in agricultural applications. Each image is captured at a high resolution of 4608 × 3456 pixels, providing detailed tree features for precise segmentation. The high resolution ensures that even small olive trees or partial tree canopies are accurately represented in the dataset.

The dataset follows the Ultralytics[6] YOLO segmentation format, designed for training segmentation models such as YOLOv11-seg. Each image in the dataset has an accompanying text file that contains the segmentation mask information. The format is as follows:

- **One text file per image**: Each image has a corresponding txt file format with the same name, storing the segmentation information.
- **One row per object**: Each row in the text file represents a single segmentation instance (i.e., a tree canopy).
- **Object information per row**: Each row contains:
    - Class index: An integer representing the object class. In this dataset, olive trees are labelled as class 0.
    - Polygon coordinates: A series of normalized (x,y) coordinates representing the vertices of the segmentation mask polygon. The coordinates are relative to the image dimensions and range from 0 to 1.

Example YOLO Segmentation Annotation Format:

0 0.4151 0.0124 0.4047 0.0176 0.3926 0.0185 0.3904 0.0197 0.3889 0.0223 0.3830 0.0443
0 0.681 0.485 0.670 0.487 0.676 0.487

[6] https://docs.ultralytics.com/

In this example:

- The first row represents a segmentation mask for an olive tree (class 0), defined by a polygon with vertices at (0.4151, 0.0124), (0.4047, 0.0176), (0.3926, 0.0185), etc.
- The second row is another segmentation instance, also labelled as class 0, with a polygon defined by fewer points.

By merging data from multiple flights, this dataset promotes robust segmentation performance, reducing the risk of overfitting to a single environmental condition. The combination of varied lighting**,** diverse shadow patterns**,** and multi-altitude imagery makes it highly representative of real-world agricultural settings, ensuring that models trained on it can perform reliably in operational drone**-**based precision agriculture tasks.

Figure 8 Images from the training dataset used

Transition to YOLOv11-seg for Segmentation

In our previous implementation, we leveraged the YOLOv8-Nano and RepViT-SAM models to develop a drone-based precision agriculture solution for olive tree segmentation, proposing two distinct approaches to address the challenge of limited and variable-quality data. In the context of continuous service improvement and optimization the YOLOv11-Seg was tested and implemented in this particular use case. YOLOv11-Seg introduces significant advancements in accuracy, robustness, and efficiency, making it a more suitable choice for UAV-based agricultural applications.

The transition from YOLOv8-Seg to YOLOv11-Seg brings enhanced mask precision, improved small object detection, and superior generalization across varying lighting conditions and flight altitudes. Additionally, YOLOv11-Seg optimizes real-time inference, enabling efficient onboard UAV processing without heavy reliance on cloud resources. These improvements make YOLOv11-Seg an ideal model for olive tree segmentation, ensuring higher detection accuracy, reduced segmentation errors, and greater reliability in precision spraying workflows.

To effectively adapt YOLOv11-Seg for olive tree segmentation, we fine-tuned the YOLOv11-Nano model on our custom dataset. The Nano variant was selected over larger alternatives (Small, Medium, Large) due to its lightweight architecture, which enables real-time inference on edge devices such as the Jetson Orin Nano, while maintaining high segmentation accuracy. Training was conducted for 74 epochs, with early stopping applied after 15 epochs, using an image size of 1024 pixels. The model achieved a mean Average Precision (mAP50) of 0.959 and mAP50-95 of 0.728 on the validation set, and mAP50 of 0.914 and mAP50-95 of 0.76 on the test set, with an average inference time of 4.9 ms per image on an RTX A4500 GPU. This represents a significant improvement over our previous implementation using YOLOv8n-Seg, which, after fine-tuning, achieved an mAP50 of 0.761. These results demonstrate a strong balance between accuracy and computational efficiency, confirming YOLOv11-Nano as a practical and effective solution for autonomous UAV-based olive tree segmentation, enabling real-time decision-making and precision agriculture applications without compromising performance.

2.3 SF_02 Terrestrial weed management

2.3.1 Trial Site Description Updates

This scenario deals with terrestrial weed management, based on AGLV moving autonomously on the olive grove, and applying weed controls, respecting organic crop development practices. The AGVL aims to facilitate the weed control process by automating the process done by the workers. To achieve this goal, the use of an autonomous mobile robot or Automated Guided Land Vehicle (AGLV) that can safely navigate on the farm is recommended. The key factors driving this idea are:

- A Smart Agriculture AGLV design, focused on hardware that can be programmatically controlled.
- Autonomous operation to guarantee the AGLV can move safely between locations in the designated area, always avoiding obstacles like people, trees, or anything else in its path while reaching its destination.

Equipment used in this Use case

In this context, a custom AGVL was created with an onboard Jetson Xavier NX, a Zed2i stereo camera, the TP-Link AC 1300 Wi-Fi adapter and the SynBoard as shown in Figure 9. The Jetson board is directly connected with the Wi-Fi adapter in order to ensure a seamless connectivity with the local network and benefit from the IEEE 802.11ac standard capabilities. The Jetson performs all the processes for detecting an object and sending the relevant command to the microcontroller that handles the drivetrain of the robot. This is achieved using the UART protocol for bi-directional communication. The drivetrain board of the robot is the SynBoard custom PCB.

Figure 9 AGVL used in the weed management use case

Software

The Operating System running on the Jetson is ROS[7]. ROS operates on a distributed system architecture, where multiple nodes (software processes) can communicate with each other through a communication framework. The communication framework enables nodes to send and receive messages, and to subscribe to and publish data streams. The way that the AGVL's software is orchestrated leverages from the inherited properties of the OS. The software presented later in this section follows the microservices-architecture by splitting each logical functionality into a dedicated container that is resilient and autonomous.

The software used for this use case was running both on the Jetson board and the SynBoard. The AGLV's autonomous navigation system is powered by a collision avoidance service that combines an obstacle detection algorithm with movement commands to guide the AGLV. This service has two main functions: navigating without obstacles and avoiding obstacles when detected.

For navigation, the AGLV uses odometry data to track its position. In our setup, the Zed2i stereo camera provides this information through its built-in sensors, including stereo cameras, an Inertial Measurement Unit (IMU), and a barometer. The IMU delivers inertial odometry data, which includes the camera's linear and angular acceleration and rotation, while the barometer provides altitude measurements. These data, combined with odometry data from the camera, are available via the "/zed2i/zed_node/odom" topic

[7] https://www.ros.org/

in ROS after launching the ZED-ROS-Wrapper[8] package. By subscribing to this topic, we can calculate the AGLV's current position and orientation using the transformations library. With the goal coordinates known, we compute the necessary yaw for the AGLV to move in the correct direction. Once aligned, the AGLV moves forward, and we monitor the distance to the goal, stopping it if it gets too close.

When an obstacle is detected, the system divides the image with the obstacle into 3 columns and 5 points sections: left, middle left, centre, middle right, and right. The algorithm then calculates the distance between the camera and five key points in the bounding box. If any of these distances are below a set threshold of 0.8 meters, those points are flagged. The system determines which part of the image (left, middle, or right) the closest points belong to, and based on this, it decides which way the AGLV should turn to avoid the obstacle. For example, if the closest points are in the middle and right sections, the AGLV will turn left to avoid the obstacle. The AGLV continues turning in this direction until no points are below the threshold (i.e., no more obstacles). After that, it moves forward for a short time before using the first part of the algorithm again, unless another obstacle is detected. The Zed2i stereo camera also provides a depth map through the "/zed2i/zed_node/depth/depth_registered" topic, which shows the distance from the camera to each pixel in the image. This allows to calculate the distance to any detected obstacle based on its position in the image. If no obstacles have been detected, the AGLV relies on the navigation system to move. If an obstacle has been detected, the system switches to obstacle avoidance, guiding the AGLV to safely move around it.

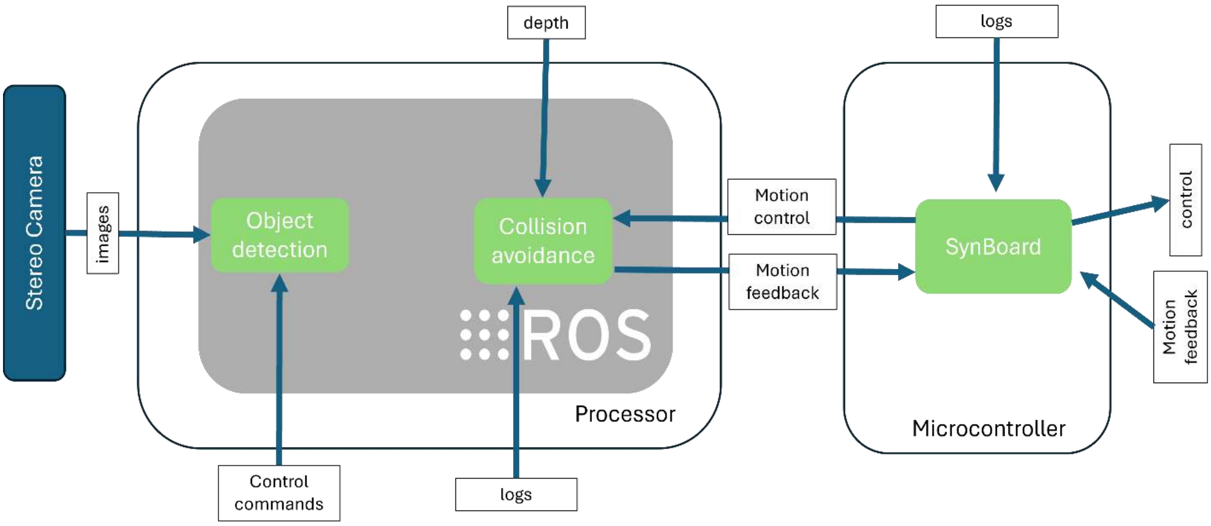


Figure 10 Logical diagram for the collision avoidance process

2.3.2 Technical validation

The Smart Farming trial's technical validation will be carried out through the test scenario detailed in the table below, addressing SF_02

Scenario: SF02_Test_Scenario_1	
Scenario ID	SF02_Test_Scenario_1 App owner as application provider. AGLV owner as resource provider.

[8] https://github.com/stereolabs/zed-ros-wrapper

Scenario: SF02_Test_Scenario_1	
	App user as app user. Execution in NEMO resources (IaaS, SaaS).
Objective	The objective is to validate the NEMO capabilities in supporting seamless operation of AGLV-based terrestrial weed management, combined with SynField data, while ensuring labor staff safety and safe AGLV movement across the olive grove.
Description	This scenario deals with terrestrial weed management, based on AGLV moving autonomously on the olive grove, and applying weed controls, respecting organic crop development practices. The scenario assumes a Smart Farmer, who desires to make use of a Smart Farming application, available through NEMO by a third party. As the Smart Farmer does not own an AGLV, they may use one of those available in NEMO resources for their area, in order to apply weed control automatically via this AGLV. The Smart Farmer selects the AGLV out of the available NEMO resources and then deploy the app on it via NEMO. The app execution should be flexibly managed in order to allow it to be performed either on the AGLV or in some edge or cloud device provided by NEMO, according to Smart Farmer's preferences or application or resource requirements. For this, ML techniques to lower the ML model size might be needed to allow its execution in IoT, edge or cloud devices, with minimal impact on performance. The scenario also involves the calculation of tokens to be credited or charged, based on the use of NEMO apps and resources.
Features to be tested	Meta-Orchestrator, PPEF, CFDRL (Observability, Workload deployment & migration) mNCC (inter-domain communication during migration) CMDT (workload discovery) Intent-based API (app and resource selection) LCM (app deployment and LCM visualization) IdM & Access Control (users) MOCA (tokens' calculation)
Requirements addressed	SF_02_FR01, SF_02_FR03, SF_02_FR04, SF_02_FR05, SF_02_FR06, SF_02_FR07, SF_02_FR08, SF_02_FR09, SF_02_FR10, SF_02_FR11, SF_02_FR12, SF_02_NFR01, SF_02_NFR02, SF_02_NFR03
KPIs	KPI_SF_02_2
Prerequisites	The NEMO framework should be deployed and functional. One registered user (let assume they are called OWNER) is acting as Smart Farming Application Owner (SFAO), who has uploaded their ML-based Weed Management App in NEMO, available for deployment by third parties. At least 1 user (let assume they are called USER_A) is registered in NEMO as a consumer for executing the aforementioned apps in their resources. At least 1 user (let assume they are called PROVIDER) is registered as a partner, having already integrated their resources (at least 1 AGLV) to NEMO. The monitoring app (e.g. SynField®) is already running in NEMO resources. USER_A is subscribed to the monitoring app. The AGLV must be able to provide its data to edge/cloud device and receive commands. The AGLV App may run an ML-based obstacle avoidance service, based on data collected by the AGLV. The Weed Management app is already registered into NEMO and available for deployment by third parties. Also, 1 user is registered as meta-OS Provider (ADMIN).

Scenario: SF02_Test_Scenario_1	
Test steps	Resource provisioning 1. The PROVIDER logs into NEMO dashboard. 2. The PROVIDER adds their AGLV in NEMO resources. 3. The PROVIDER is notified/informed of tokens and operations' results through the NEMO dashboard. Resource & workload selection 1. USER_A logs into NEMO dashboard. 2. USER_A visualizes the available apps with available devices in their region. 3. USER_A selects the Weed Management app. 4. USER_A visualizes the list of available compatible devices in their region. 5. USER_A selects the AGLV to lease. 6. USER_A requests deployment of the Weed Management app in the AGLV. 7. USER_A is notified/informed of tokens and operations' results through the NEMO dashboard. App execution 1. USER_A signs in the Weed Management app. 2. USER_A configures a new route for the AGLV. 3. The AGLV starts its route. 4. Upon detection of obstacle, the AGLV changes its trajectory. 5. When NEMO identifies a risk in the app execution on the AGLV (e.g. due to its power exceeding a predefined limit), NEMO undertakes reallocation of the workload to some other device (e.g. edge or could). 6. USER_A monitors the application, workload and device performance through the NEMO (LCM) dashboard. 7. The route completes and application logs are collected.
Success state	All obstacles are avoided and AGLV operation stops when the route is covered. The AGLV operation is seamless.
Failure state	The AGLV stops before the route is completed; and/or The AGLV collides with a tree or human; and/or The AGLV operation is interrupted or lags.
Responsible for testing and implementation	ESOFT, SYN
Risks	No risks identified so far.

Table 3 Test scenario SF 02

2.3.3 Use case diagrams

The use case diagrams presented in deliverable D5.2 [5] provide a detailed description of how the several entities interact with each other. These diagrams are foreseen to be updated in deliverable D5.4 with the finalization of the piloting use case as new interactions may be needed as the project progresses.

2.3.4 Timeline of activities

Phase	Estimated start date	Estimated end date	Notes
Trial set-up and equipment procurement	M5	M18	Completed
Initial implementation and validation	M19	M30	Completed
Final implementation and validation	M31	M36	Validation is mainly on the basis of the final integrated prototype with NEMO platform

Table 4 Timeline of activities SF 02

2.3.5 Intermediate Results

During the testing phase the object detection and avoidance capabilities of the robot were tested. The Yolo v5 object detection algorithm used was trained with a combination of the COCO dataset and a custom dataset created by images of the testing area. This detection algorithm provides the relevant input to the navigation algorithm. The navigation algorithm is evaluated under three scenarios, namely "1-static object", "2-static objects" and "moving object". Under these scenarios the following metrics are evaluated:

- The number of times that it was not possible to avoid an obstacle over the number of times an obstacle was detected.
- The distance between the robot and the object.
- The time required for the robot to detect the object, calculate the new path to be followed and ultimately avoid the object.
-

Figure 11, Figure 12, Figure 13 and Figure 14 demonstrate the effectiveness of the detection and navigation algorithms in the previously mentioned metrics. Overall, the behavior of the robot was satisfactory, and the tests were successful.

Figure 11Results obtained with one stationary obstacle

Figure 12 Results obtained with two stationary obstacles

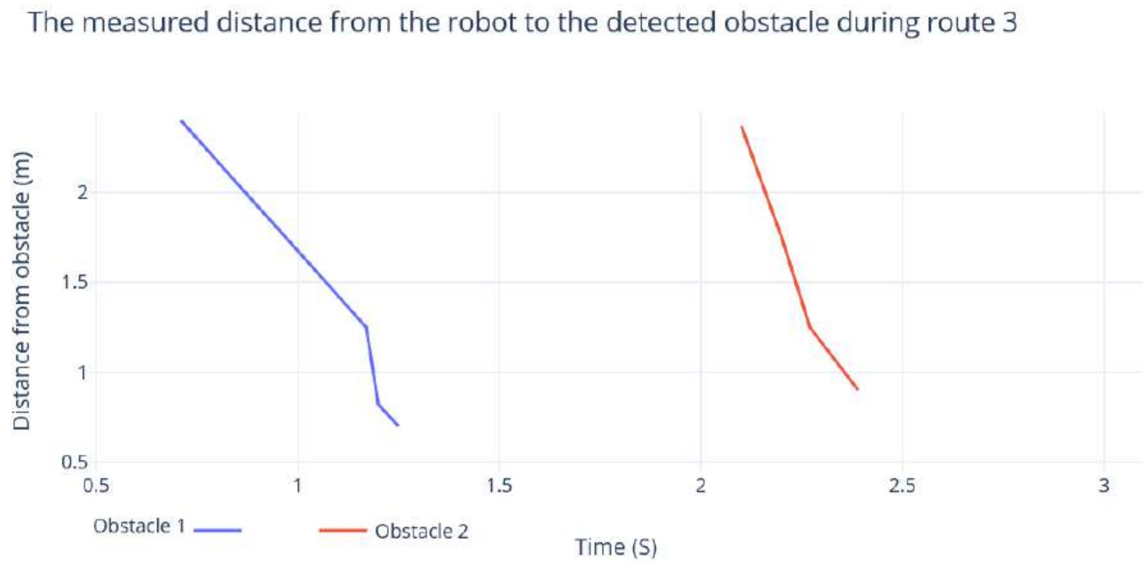


Figure 13 Results obtained with two stationary obstacles in a different route

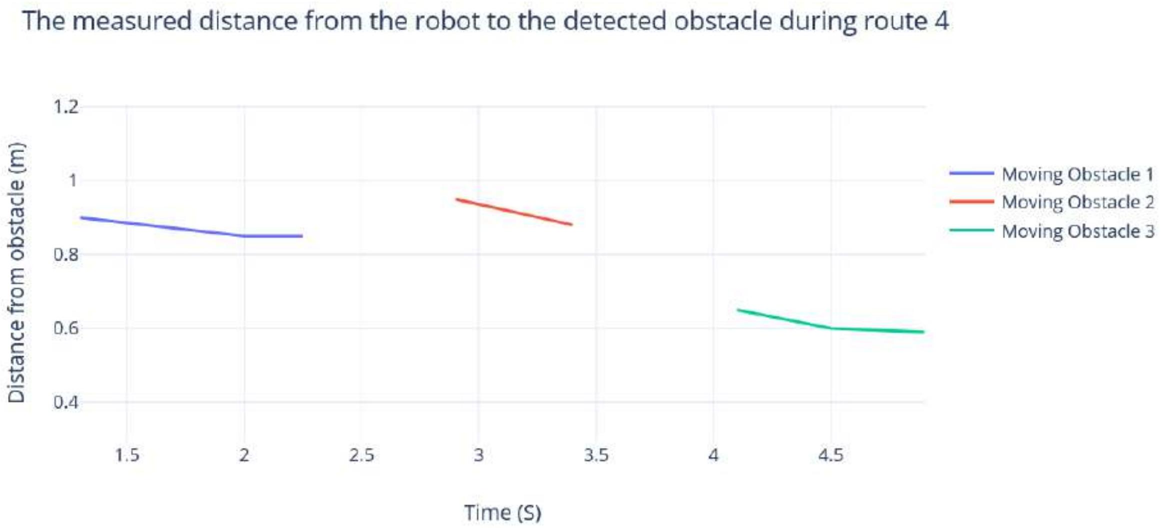


Figure 14 Results obtained with three moving obstacles

The overall performance of the navigation algorithm is demonstrated in Figure 15, which shows the total number of obstacles detected and avoided across the different routes the robot was tested on. The results indicate that the robot successfully avoided all obstacles on every route, leading to zero collisions. This means the robot can efficiently move from the starting point to the destination while steering clear of any obstacles. Additionally, Figure 16 presents the robot's response time measurements. Regardless of the route type (whether it involves static or moving obstacles) or the number of obstacles, the robot consistently achieves response times of under one second. This shows that the robot can react quickly to avoid any obstacle in its path.

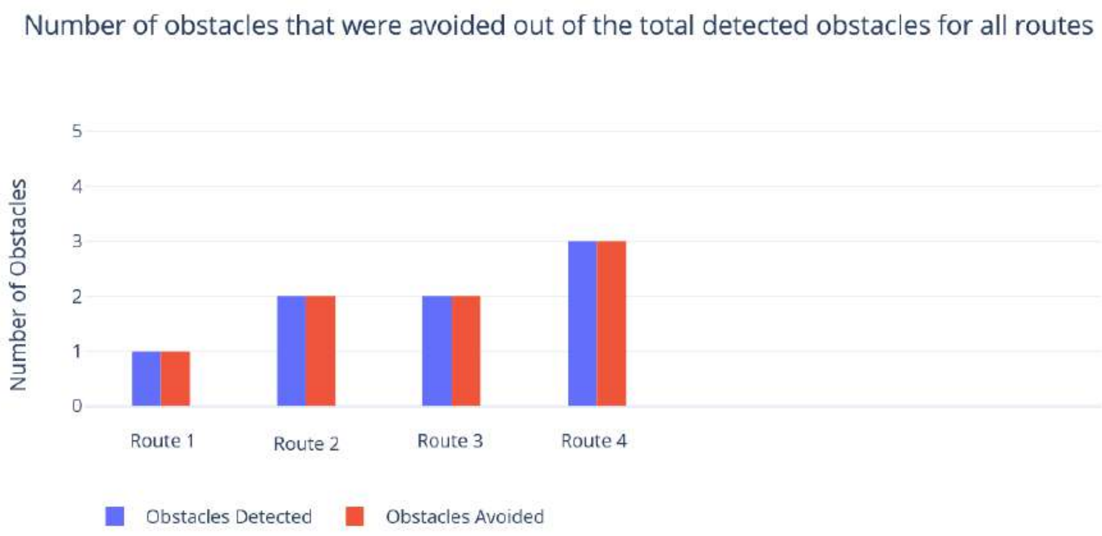


Figure 15 Success of the obstacle avoidance algorithm

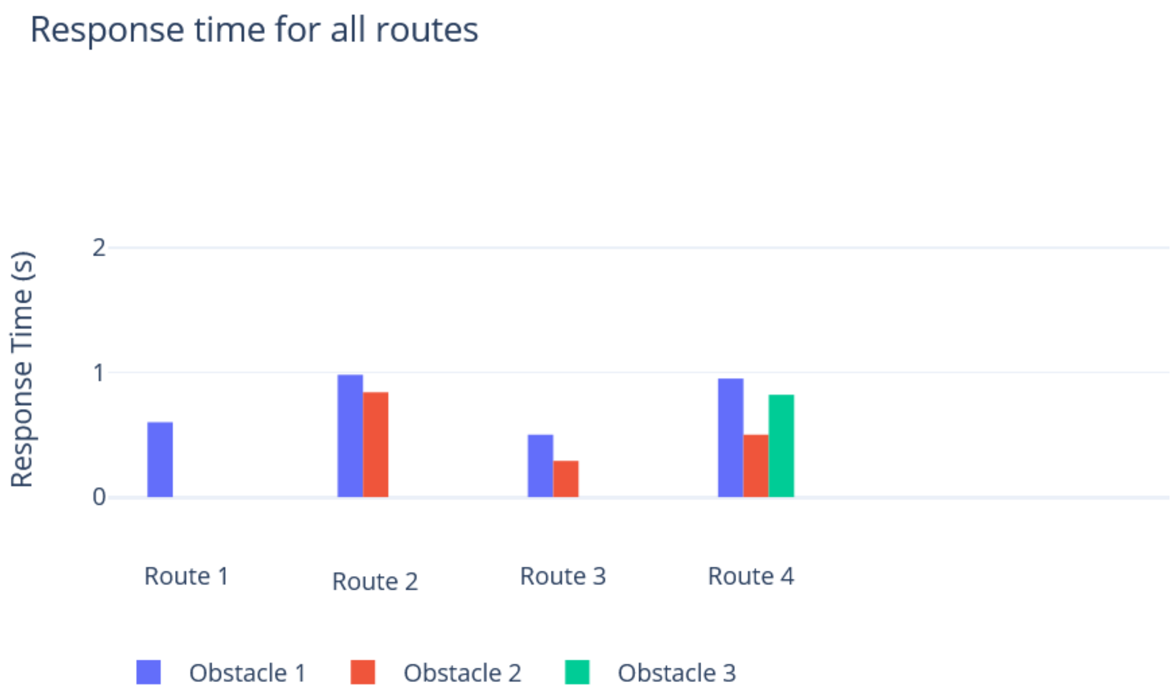


Figure 16 Time required for each trial for the AGVL to detect and avoid collision

3 Smart Energy & Smart Mobility Trial

3.1 Introduction and Use case Applications

The Smart Energy Living Lab is located in Terni, central Italy, and is managed by ASM Terni. As part of the ongoing transformation of the electric distribution network, this trial focuses on addressing challenges associated with the increasing penetration of renewable energy sources (RES) and the evolution of urban mobility. While the integration of RES is essential for reducing carbon emissions and dependency on fossil fuels, it introduces grid stability issues, including voltage fluctuations, current imbalances, and reverse power flow. These challenges necessitate advanced monitoring, control, and optimization strategies to ensure a stable and efficient electrical distribution system.

A key component of the trial site is its real-time monitoring infrastructure, which includes Power Quality Analysers (PQAs) and Phasor Measurement Units (PMUs) deployed at different grid levels. These devices play a crucial role in tracking power flow, voltage variations, and disturbances in both the medium-voltage (MV) and low-voltage (LV) segments of the network. PMUs provide high-resolution synchronized measurements that enable fault detection, classification, and localization, supporting faster mitigation strategies. Data collected from PMUs and PQAs are processed both locally at the edge—for real-time event detection—and remotely in the cloud, where down sampled data is stored for further analysis. This dual-layer approach minimizes communication overhead while ensuring timely and reliable fault response mechanisms.

Beyond grid monitoring, the Smart Energy Living Lab is an experimental hub for integrating electric vehicle (EV) charging, parking systems, and smart grid management. The growing adoption of EVs presents both opportunities and challenges for the distribution network. If unmanaged, EV charging can exacerbate grid congestion; however, when strategically coordinated, it can improve grid balancing by shifting demand to periods of peak RES generation. Through predictive modelling and demand-response mechanisms, the site leverages data from EV chargers, parking sensors, and mobility platforms to optimize charging schedules, reduce energy waste, and enhance eco-mobility services. Additionally, crowd-sourced data, environmental monitoring, and traffic flow analysis support real-time decision-making, further improving the efficiency of urban mobility.

Based on the modification reported in D5.2 [5], the trial is validated through two use cases described in the following subsections.

3.2 SE_01 Smart Grid

3.2.1 Trial Site Description Updates

In addition to D5.2[5], the trial site for SE_01 focuses on a specific portion of electrical distribution grid equipped with PQAs and PMUs at substation level to support real-time monitoring and fault detection, classification, and localization. This feeder, which is part of a medium-voltage (MV) grid segment, contains a high level of renewable energy sources, making it particularly susceptible to fluctuations and disturbances. The PQA and PMU infrastructure continuously monitors the feeder, where the former meters are used to track the power flows of the MV/LV transformer stations in the feeder and the latter to track the voltage and current profiles of the primary and secondary substation.

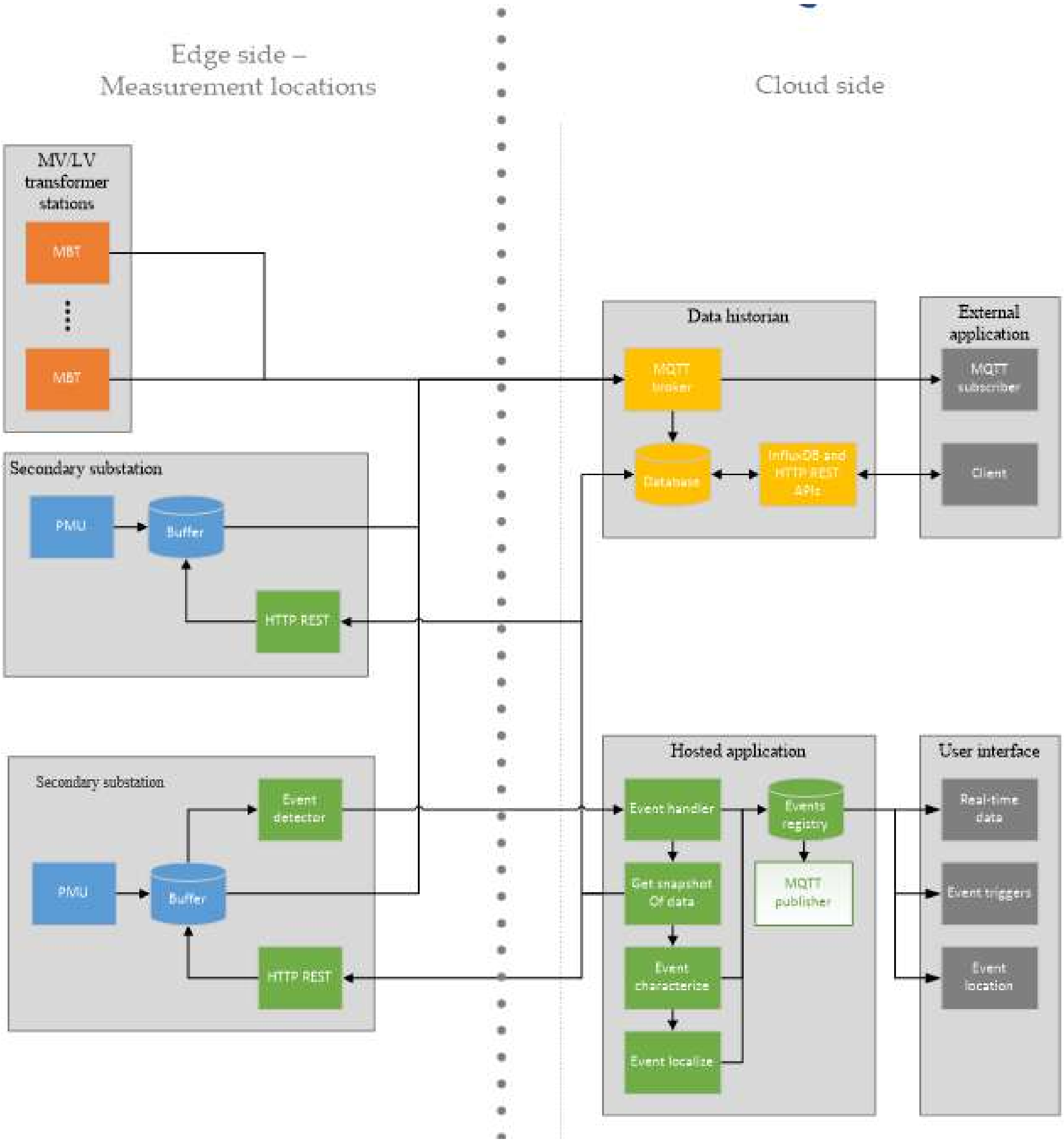


Figure 17 High level architecture involving PMUs and Application

The PMU measurements at full granularity (50 Hz) are stored in a circular buffer at the edge of the grid, while the downsampled measurements are sent together with the PQA data to the data historian in the cloud. Downsampling is employed to reduce data traffic over LTE, thereby lowering the operational expenses for the DSO associated with data transmission. The data stored in the cloud can be retrieved manually on request or automatically by a hosted application (fault localization algorithm).

The PMU in the secondary substation is also accompanied by an event detector, which is used to detect faults at full data granularity to enable a rapid response. Whenever an event is detected, a precise timestamp of the event is automatically reported and stored in the cloud of the proposed framework. A data query (the full resolution of the data for a short time window around the timestamp) is then issued to the PMU and PQA databases. Once all the required data has been reported back from the edge to the cloud, the fault is classified and localized. The results for the corresponding event are then automatically sent to the GUI of the DSO, where they are visualized and stored in the event log.

The SE_01 trial demonstrates how the NEMO architecture supports improved fault detection and resilience in urban grids with a high proportion of renewables and provides a model for effective feeder management.

3.2.2 Technical validation

The Smart Energy trial's technical validation will be conducted through the test scenario detailed in the following subsections, addressing SE_01

Scenario: SE01_Test_Scenario_1	
Scenario ID	SE01_Test_Scenario_1
Objective	Hierarchical grid disturbance mitigation
Description	Observe MV grid segment in real-time utilizing the MBT and PMU infrastructure. Complement PMU with edge gateway/computing for data pre-processing and disturbance events detection. Upon local event detection, trigger disturbance localization and classification procedures. Provide the operator with insights to make informed decisions.
Features to be tested	AIoT Architecture CMDT Micro-services Secure Execution Environment MLOps via CF-DRL PRESS, Safety & Policy enforcement framework Cybersecurity & Digital Identity attestation meta-Orchestrator
Requirements addressed	- SE_01_FR01: The platform has the capability to monitor the real-time data from the sensors deployed in the grid - SE_01_FR03: High-tech power sensors should be useful to elaborate on new strategies, in order to improve the power quality in a secure way - SE_01_FR04: Based on sampled data, phasors are calculated with high precision and the synchronization process must be very fast. - SE_01_NFR01: Secure communication of sensitive data related to the infrastructure should be provided.
KPIs	- KPI_SE_01_1: Time granularity for monitoring (<1 s) - KPI_SE_01_2: Information exchanged by devices (> 100.000 measurements/min)
Prerequisites	The NEMO platform should be installed and configured. All the sensors are deployed in the electrical grid and collect the data continuously. The topology and line parameters of the green feeder are provided by DSO.
Test steps	• Deployment of MBT/PMU infrastructure • Integration • Verification of measurements quality • Development of the fault classification and localization model. • Evaluation and refinement of the model based on results.

Scenario: SE01_Test_Scenario_1	
	Technical validation includes: • Reuse the NEMO functionality of the integration of so-called far-edge device management (firmware update) for managing the PMU Gateways. • Deployment and use of the high-insensitive data processing gateway engine • Gathering crucial information from the edge devices (low intensity logs and high intensity parts if the disturbance is made) and its on edge processing • Monitoring and alerting feature for Grid Disturbance Mitigation System • The graphical interface for review of the gathered data.
Success state	Improved observability of the grid in the steady-state and additional mitigation information in case of disturbance. The final model is deployed and the fault classification and localization are accurate. Technical validation will prove the usability of the services built in WP2 (CMDT), WP3 (MetaOrcestrator, Nerves) and WP4 (Integration of NEMO results).
Failure state	The model fails to classify the fault or the fault location is false
Responsible for testing and implementation	ASM
Risks	Potential delays due to partner withdrawal have been mitigated so far

Table 5 Test Scenario SE_01

Technical validation includes:

- Reuse the NEMO functionality of the integration of so-called far-edge device management (firmware update) for managing the PMU Gateways.
- Deployment and use of the high-insensitive data processing gateway engine
- Gathering crucial information from the edge devices (low intensity logs and high intensity parts if the disturbance is made) and its on edge processing
- Monitoring and alerting feature for Grid Disturbance Mitigation System
- The graphical interface for review of the gathered data.

Technical validation will prove the usability of the services built in WP2 (CMDT), WP3 (MetaOrcestrator, NERVES) and WP4 (Integration of NEMO results).

3.2.3 Use case diagrams

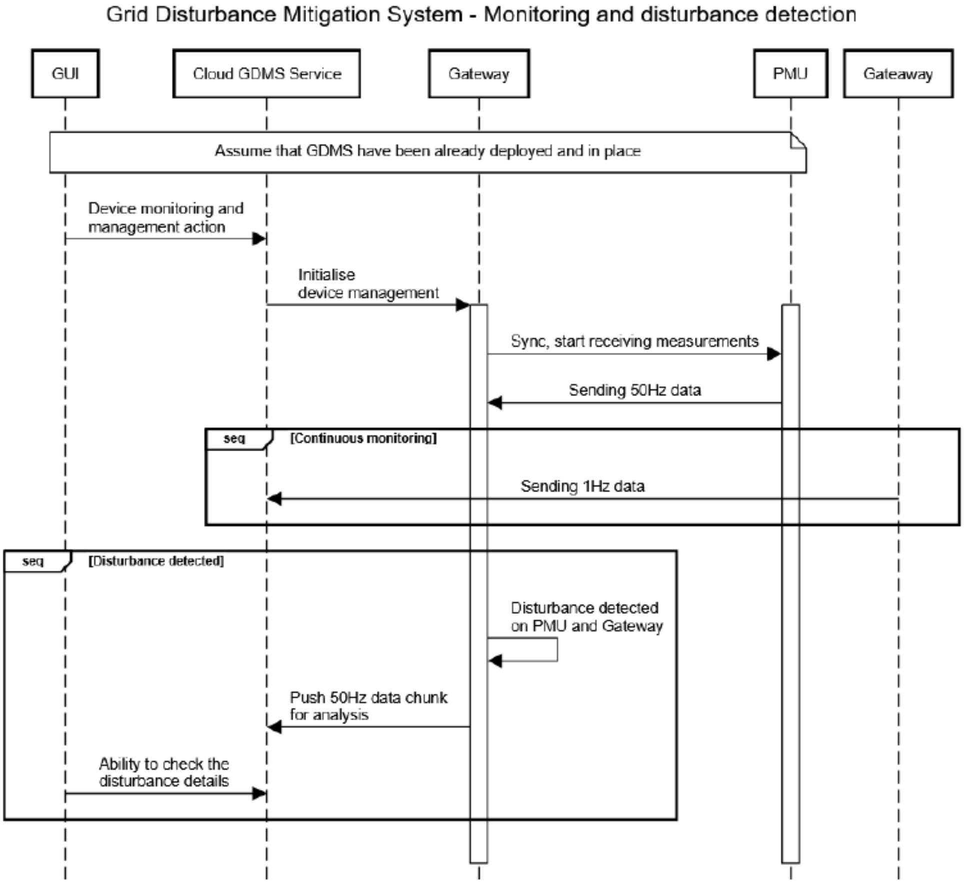


Figure 18 Use case sequence diagram for SE 01

3.2.4 Timeline of activities

Phase	Estimated start date	Estimated end date	Notes
Trial set-up and equipment procurement	M5	M18	Completed
Additional Hardware Deployment	M27	M29	Completed
Initial implementation and validation	M30	M31	Completed
Final implementation and validation	M32	M36	Validation is mainly on the basis of the final integrated prototype with NEMO platform

3.2.5 Intermediate Results

This Use case integrates Phasor Measurement Units (PMUs) and Edge Gateways for real-time grid monitoring and disturbance detection. The Figure 19 illustrates how the NEMO cloud infrastructure interacts with far-edge devices deployed at substations.

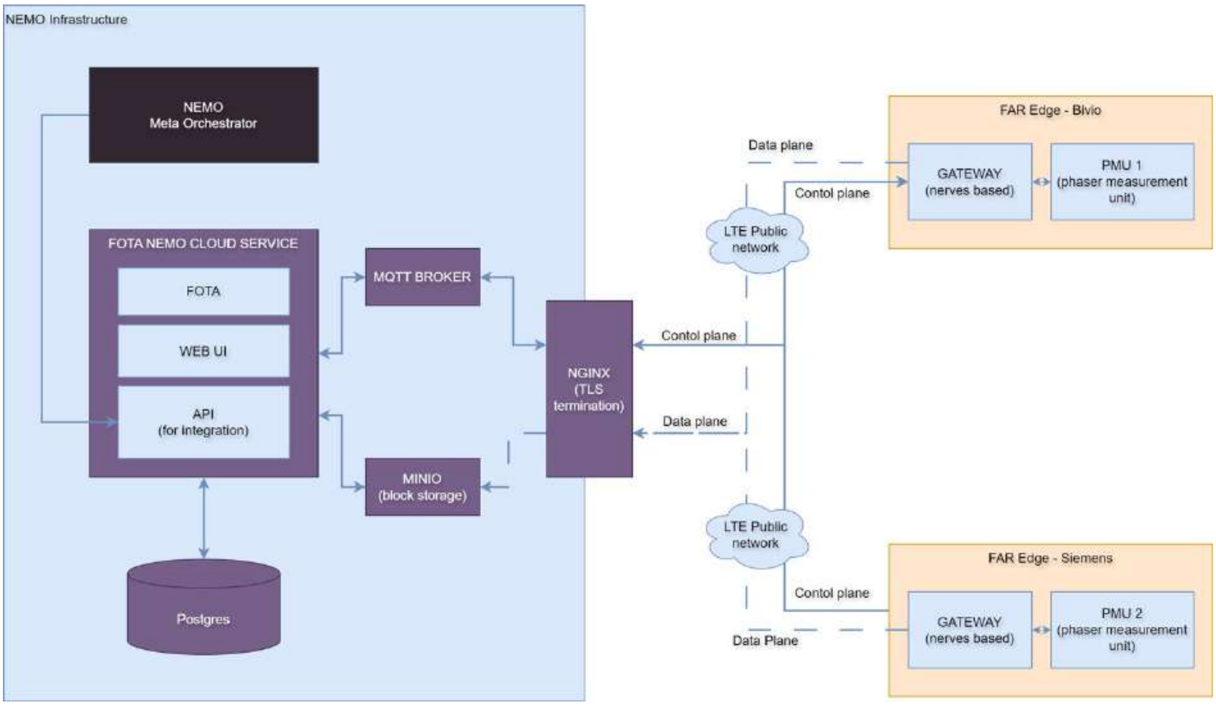


Figure 19 NEMO Deployment schema

SSE, PMU+NERVES deployment:

- **Cloud part** (cloud services)
  - Kubernetes manifest files to define deployment and interactions
  - UI [container]
  - FOTA API [container]
  - Message exchange (MQTT/Rabbit) [containerized service, part of IBMC]
  - Object Storage database (MinIO) [containerized service, IBMC]
  - Proxy/entrypoint (NGINX) [containerized service part of Kubernetes]
- **Edge part**
  - Gateway node [Nerves-based, bare-metal solution]
  - PMU units (on-premises, COMS proprietary solution)

The NEMO infrastructure includes the NEMO Meta Orchestrator (MO), which is responsible for managing workloads and ensuring efficient data exchange between the edge and cloud. Below the orchestrator, the FOTA (Firmware Over the Air) NEMO Cloud Service provides essential functionalities, including remote firmware updates, a graphical user interface (GUI) for managing devices, and an API for integration. This API enables the NEMO Meta Orchestrator to initiate firmware upgrades and manage edge devices remotely.

To facilitate data communication between the edge and cloud, an MQTT broker is integrated into the system. This broker ensures continuous monitoring of far-edge activities and allows for bidirectional communication. Through MQTT, NEMO can send commands to edge devices and receive status updates and event data. Additionally, a NGINX server with TLS termination is deployed to enhance security, ensuring that all data exchanges between edge and cloud components remain protected.

At the far edge, PMUs are deployed at two substations, Bivio and Siemens. These PMUs generate high-resolution measurements at 50 Hz, capturing critical grid parameters. The data is first processed by an Edge Gateway, which plays a crucial role in optimizing bandwidth usage. Under normal conditions, the gateway downsamples the 50 Hz data to 1 Hz and periodically transmits it through the MQTT broker, allowing operators to monitor real-time grid conditions. However, if the gateway detects a grid disturbance or anomaly, it immediately sends the full 50 Hz dataset to the MINIO storage within the NEMO infrastructure for advanced analysis. This selective data transmission strategy helps optimize network bandwidth while ensuring that critical high-resolution data is available when needed.

The firmware management feature in NEMO further enhances system efficiency. The FOTA service allows operators to remotely update the firmware of edge gateways without requiring physical intervention. This capability ensures that PMUs and edge devices remain secure, up-to-date, and aligned with evolving grid management requirements.

Overall, this architecture supports real-time grid monitoring, efficient event-driven data transmission, and remote firmware updates while maintaining secure and scalable communication between edge and cloud components. By leveraging NEMO's orchestration and monitoring capabilities, the Smart Energy trial ensures that grid operators can quickly detect disturbances, analyze high-resolution data, and maintain a robust decision-support system for effective grid management.

The intermediate results of the SE_01 trial site demonstrate the practical functionality of the developed framework, with a key emphasis on the GUI, designed for the DSO. The GUI serves as an intuitive and centralized platform for accessing real-time measurements and fault localization results, ensuring efficient grid management and fault response. The GUI, as shown in the Figure 20, summarizes the relevant information for the DSO and enables seamless navigation and data visualization. Key features of the GUI include:

- Real-time monitoring:
    - The GUI provides a live display of the voltage, current and power profiles measured by the PMUs and PQAs at various points on the grid.
- Fault localization results:
    - When a fault is detected, the GUI promptly displays the results of the fault localization algorithm.
    - The visualization highlights the affected grid segment and allows the DSO to quickly identify and address the issue.
- Event log and data retrieval (in progress):
    - Each detected event is automatically stored in the event log, which is accessible via the GUI.
    - The DSO can query the data in full resolution for desired events, including timestamps and high granularity (50 Hz), which are fetched from the edge on demand.
- Customizable notifications:
    - The GUI supports customizable notifications that ensure the DSO is immediately alerted to significant disturbances or anomalies detected in the feeder.
- Data flow and accessibility (To be integrated in next period).
    - The GUI establishes a connection between the cloud-based data storage and the DSO's operational interface. Down-sampled measurements (sent over LTE to reduce costs) are

readily available for routine monitoring. In contrast, high-granularity data stored in edge devices is fetched only when required, e.g. during fault investigation. This architecture ensures cost-effective data management without compromising the accuracy or responsiveness of the system.

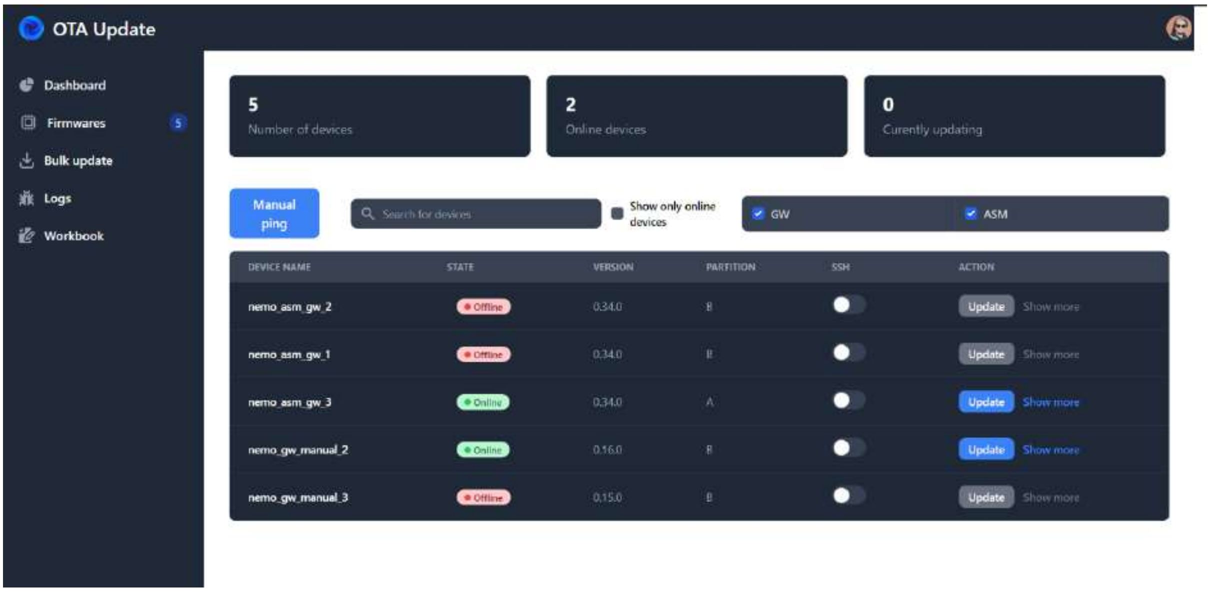


Figure 20 GUI for SE 01

3.3 SE_02 Smart Mobility/City

3.3.1 Trial Site Description Updates

The SE_02 Use case focuses on enhancing Renewable Energy Sources (RES) integration within urban mobility infrastructure by coordinating smart grid resources, parking, and electric vehicle (EV) charging. This use case aims to balance RES load demand with the increased need for EV charging in urban areas, optimizing energy use across these interconnected systems. Leveraging data from EV chargers, parking sensors, and public transit systems, SE_02 utilizes predictive models to manage EV charging schedules and parking availability. By strategically scheduling EV charging during periods of high renewable energy generation and low grid demand, SE_02 alleviates grid congestion while supporting sustainable mobility.

In addition to integrating smart EV chargers and parking systems, SE_02 incorporates crowd-sourced information, weather and environmental data, as well as CCTV footage and traffic metrics. These comprehensive data inputs enable the system to optimize urban mobility, adjusting public transport schedules, managing traffic flow dynamically, and making real-time parking recommendations. These coordinated efforts support eco-mobility, reduce energy waste, and enhance the efficiency of urban transportation, all while facilitating greater RES integration into the grid to create a more sustainable and resilient urban ecosystem.

Figure 21 ASM Living Lab

3.3.2 Technical validation

The Smart Energy trial's technical validation will be conducted through the test scenario detailed in the following subsections, addressing SE_02

Scenario: SE02_Test_Scenario_1	
Scenario ID	SE_02_Test_Scenario_1
Objective	• Improve Renewable Energy Sources (RES) load balancing via EV chargers. • Predict traffic flow/parking prediction via EV chargers and parking positions for Mobility. • Support citizens eco-mobility in a smart city scenario combining crowd sourcing. info and public transportation, weather/noise data, along with historical data and analysis of CCTV/traffic.
Description	Currently the transition to renewable energy and electric mobility is proceeding in parallel, creating new opportunities and new obstacles: by increasing the number of electric vehicles, the amount of electricity that must be supplied increases and, therefore, a necessary strengthening of power lines follows; moreover, this energy will progressively come from intermittent and non-programmable renewable energy plants, resulting in an energy balancing challenge. In this context, a cooperation mechanism between DSO (Distribution System Operator), eMSP (electric Mobility Service Provider) and EV users allows both a power lines improvement limitation and grid balancing service by coordinating EV charging. DSO monitors the electricity grid via distributed IoT smart meters and, thanks to accurate forecasting systems developed by COM, DSO is able to identify how, when and where to charge electric vehicles for grid balancing. eMSP will then be able to offer advantageous charging price at DSO-selected charging stations, attracting more EV users. In particular, ASM and EMOT, supported by COM, W3 and TSG realize driver-friendly scenarios for smart city mobility and dispatchable charging of EVs based on RES demand-response along with human-centred smart micro-contracts and micro-payments. The use case

Scenario: SE02_Test_Scenario_1	
	will utilize basic geography, street-level, public transportation, weather and noise data, along with historical data and analysis of CCTV/traffic cameras to model and train distributed AI models on traffic flow and parking prediction in a greedy layer wise fashion. W3 5G network is also utilized together with TSN by CMC and a novel twin Green Data Centres infrastructure.
Features to be tested	AIoT Architecture Cybersecure Microservices Digital Twin (CMDT) MLOps via CF-DRL Federated meta Network Cluster Controller (mNCC) Micro-services Secure Execution Environment (SEE) PRESS, Safety & Policy enforcement framework Cybersecurity & Digital Identity attestation meta-Orchestrator Intent-based migration SDK
Requirements addressed	SE_02_FR01, SE_02_FR02, SE_02_FR03, SE_02_FR04, SE_02_FR05, SE_02_FR06, SE_02_FR07, SE_02_FR08, SE_02_FR09, SE_02_FR10, SE_02_FR11, SE_02_FR12, SE_02_FR13, SE_02_NFR01, SE_02_NFR02, SE_02_NFR03, SE_02_NFR04, SE_02_NFR05, SE_02_NFR06, SE_02_NFR07
KPIs	KPI_SE_02_1, KPI_SE_02_2, KPI_SE_02_3, KPI_SE_02_4
Prerequisites	The NEMO platform should be installed and configured. Consumption and production data should be constantly collected. DSO and CPO platforms must implement a Demand/Response (DR) marketplace.

Table 6 Test scenario SE 02

3.3.3 Use case diagrams

The use case diagram presented in deliverable D5.2 [5]provide a detailed description of how the Smart Mobility/City Trial several entities interact with each other.

3.3.4 Timeline of activities

Phase	Estimated start date	Estimated end date	Notes
Trial set-up and equipment procurement	M5	M18	Completed
Initial implementation and validation	M19	M30	Completed
Final implementation and validation	M31	M36	Validation is mainly on the basis of the final integrated prototype with NEMO platform

Table 7 Timeline of activities SE 02

3.3.5 Intermediate Results

The intermediate results in this use case comprise three major developments:

- A Graphical User Interface (GUI) for the Distribution System Operator (DSO) – to visualize congestion forecasts and request flexibility from charging point operators (CPOs)/EV fleet managers.
- A Graphical User Interface (GUI) for the Charging Point Operator to visualize charging stations and electric vehicles status and, based on that, to provide flexibility offer for distribution system operator.
- A Flexibility Marketplace – where CPOs/EV fleet managers can bid to provide flexibility by charging EVs at lower costs when surplus energy is available.

The Flexibility Management Interface provides DSOs with a centralized tool for managing transformer congestion levels and optimizing flexibility requests. The interface as shown Figure 22 presents a structured list of transformer stations, each identified by name and location, allowing the DSO to quickly assess grid conditions and make informed decisions.

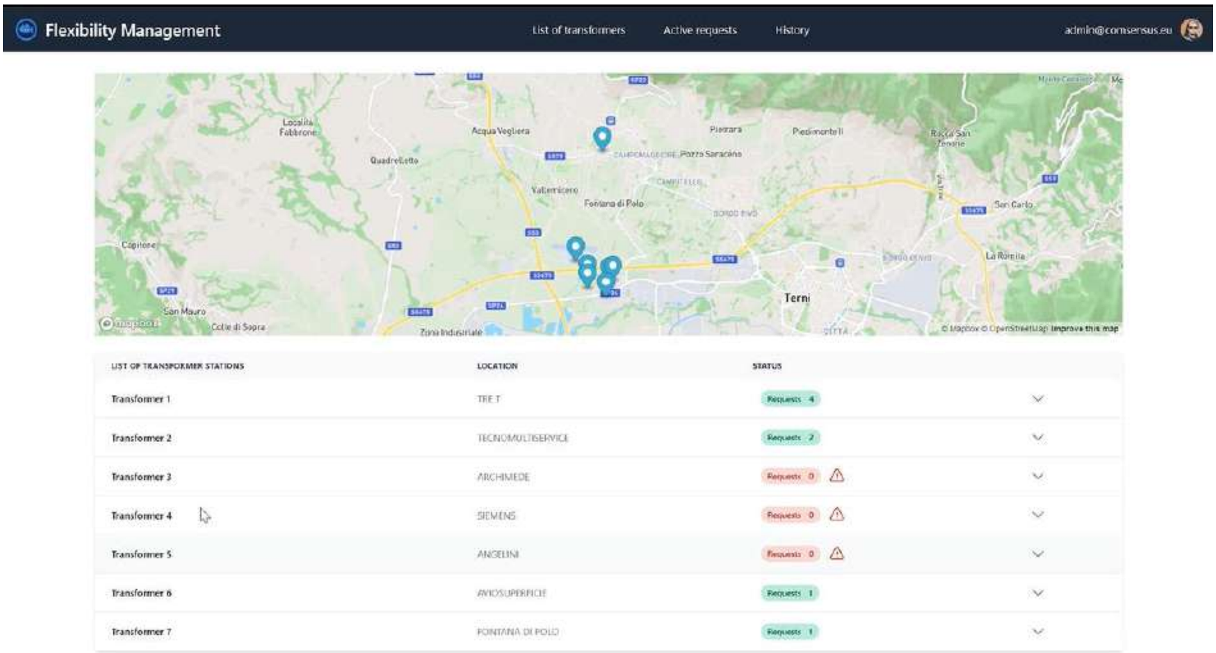


Figure 22 Map-Based Visualization

The map-based visualization enhances situational awareness by displaying the transformer stations geographically. This feature allows DSOs to assess congestion patterns across different regions, enabling proactive congestion management. Clicking on a transformer station as shown in Figure 23 brings up its historical data, providing deeper insights into grid stability at that specific location. The combination of tabular and geographic views ensures that flexibility decisions are made based on both real-time and historical grid conditions.

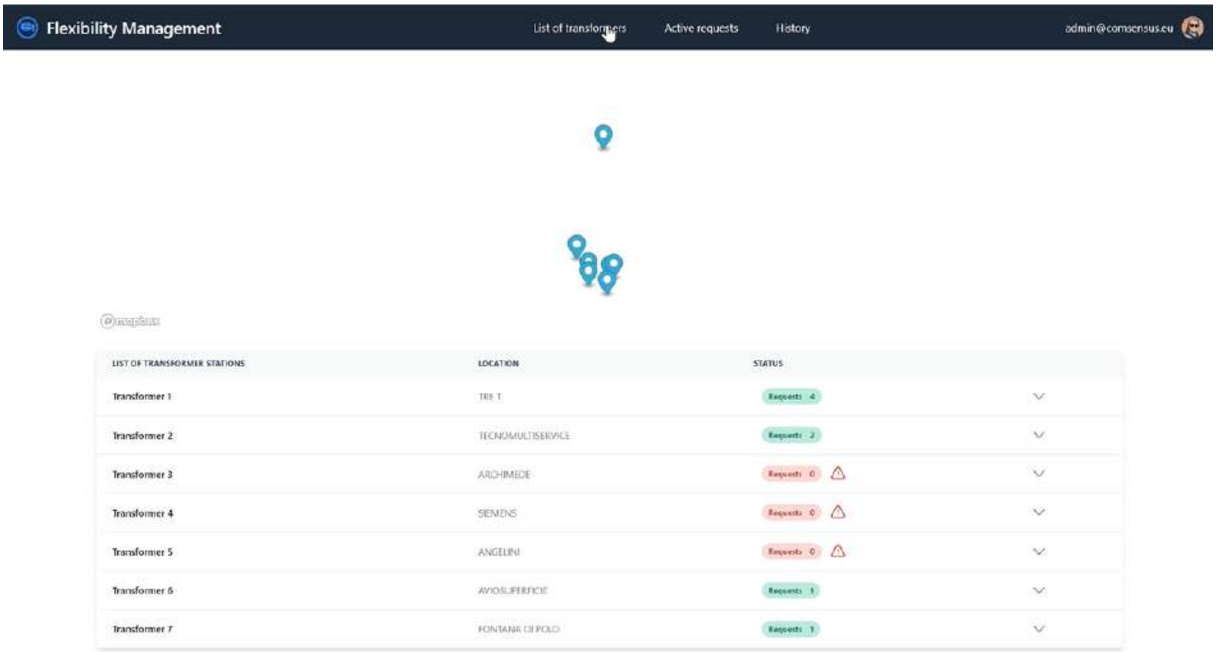


Figure 23 Map-Based Visualization

When congestion is detected at a station, the DSO can create a flexibility request through an interactive submission form as shown in Figure 26. The request requires the DSO to input start and end times, the amount of flexibility needed (kWh), and the maximum acceptable offer price (€ per kWh). Once submitted, this request is dispatched to potential flexibility providers, such as Charging Point Operators/EV fleet managers, who can then respond with their bids.

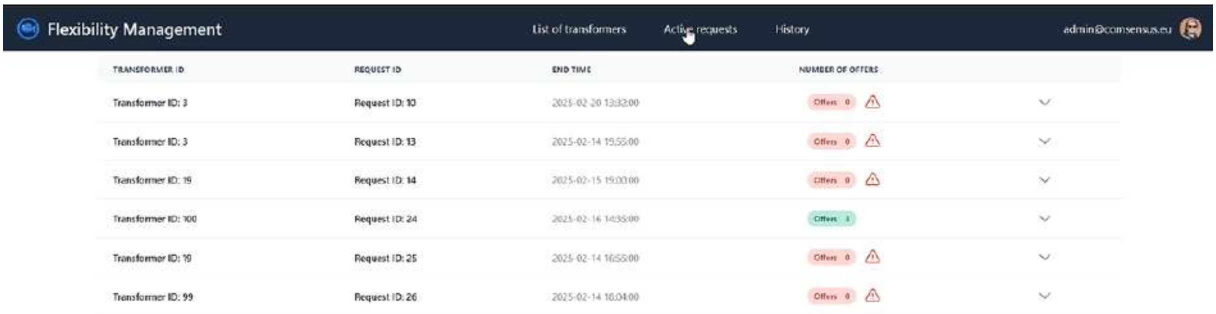


Figure 24 Active requests

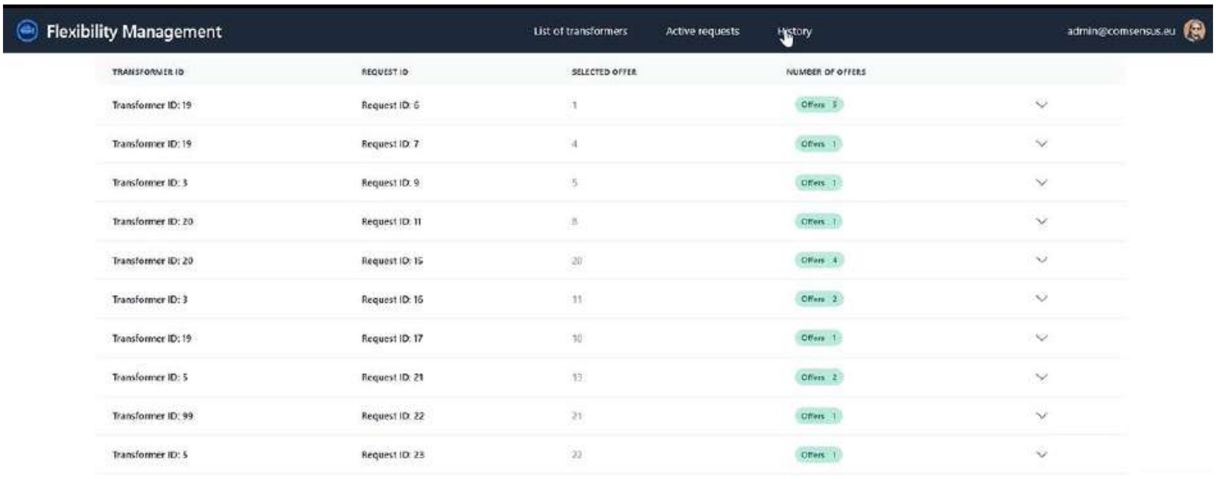


Figure 25 History of requests

A dedicated Active Requests tab as shown in Figure 28 allows the DSO to track all ongoing flexibility requests. Each request is displayed with its unique request ID, deadline, and the number of bids received. If no bids are received, the DSO can adjust the request parameters to increase engagement from flexibility providers. Once bids are available, the DSO can evaluate different offers, considering parameters such as the bid price, available flexibility, and response time before selecting the most optimal offer.

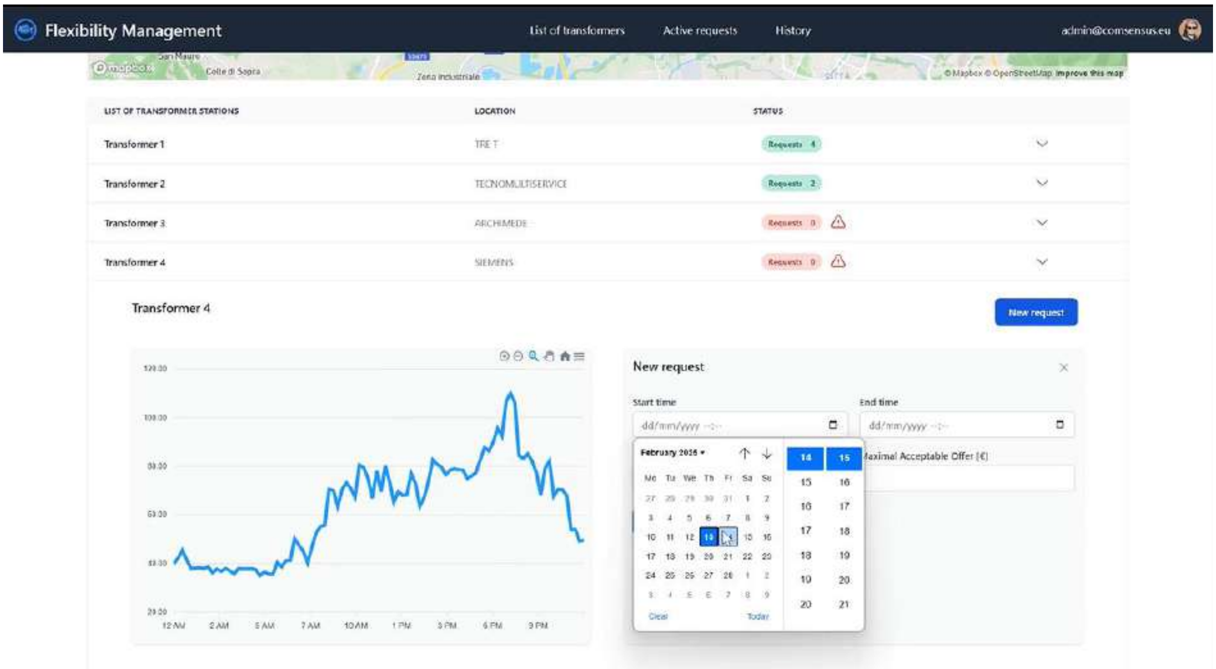


Figure 26 Flexibility Request Submission

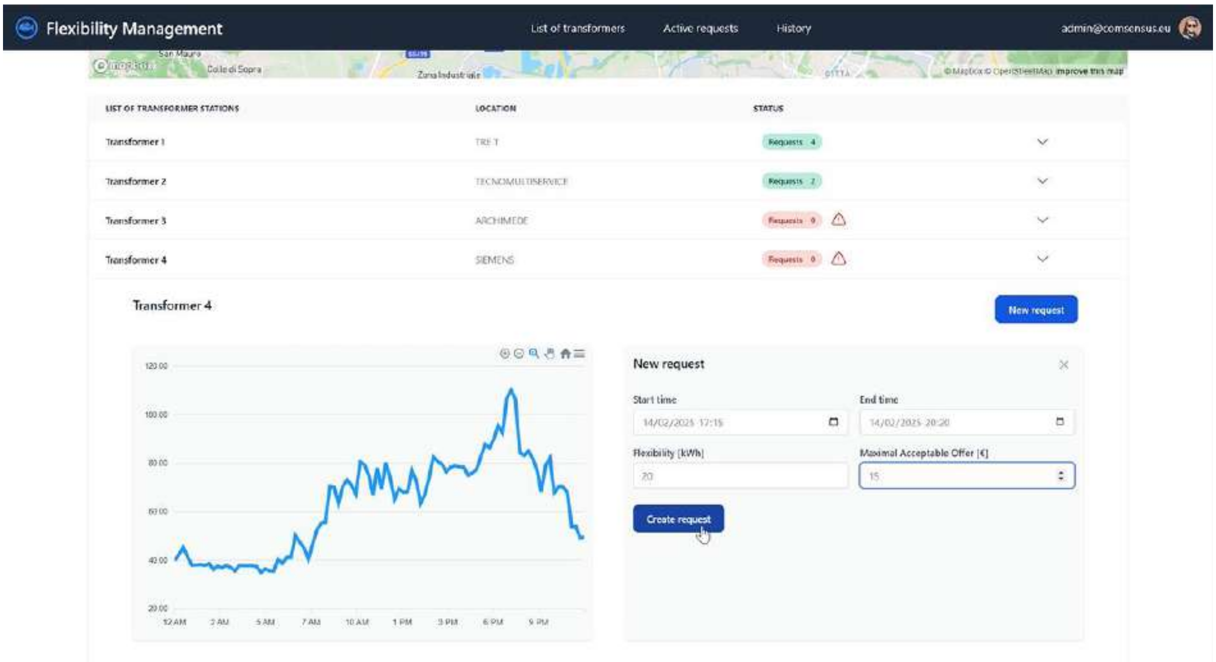


Figure 27 Flexibility Request Submission

To ensure transparency and learning from past transactions, the History tab maintains a record of previous flexibility requests as shown in Figure 25. It logs crucial details such as selected bids, their respective prices, and the overall impact of flexibility activation on the grid. By reviewing past transactions, DSOs can refine their approach to flexibility procurement, ensuring more effective congestion mitigation strategies in future operations.

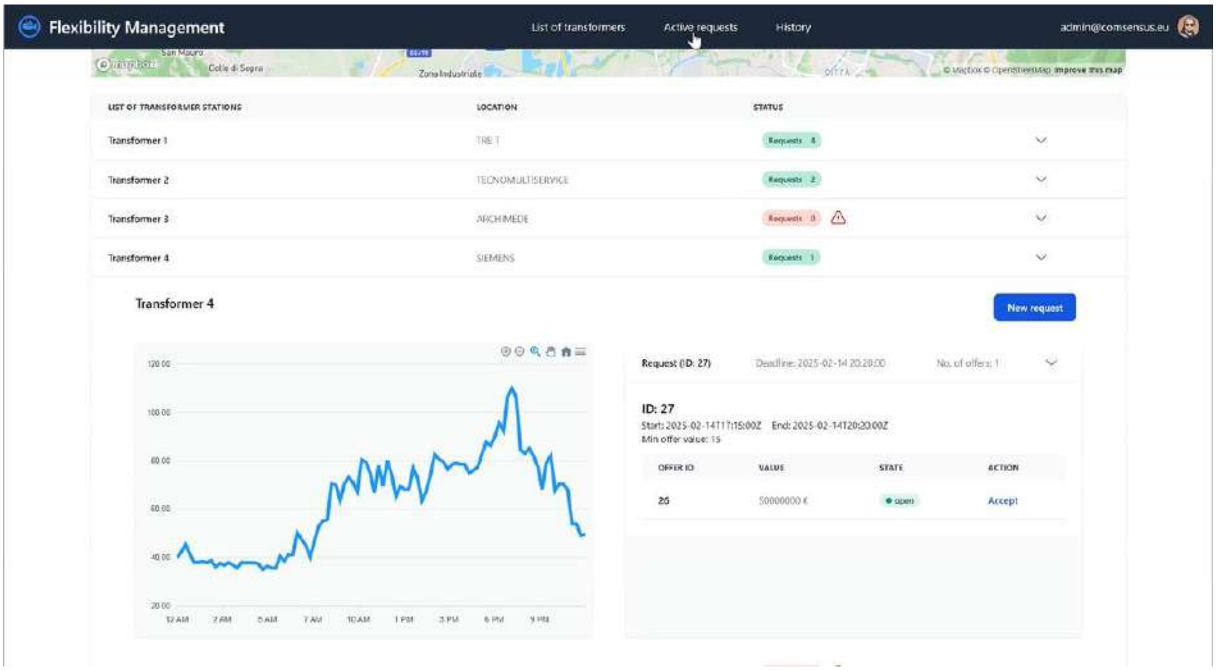


Figure 28 Active Requests

The GUI, therefore, acts as an intelligent decision-support system, seamlessly integrating real-time grid monitoring, flexibility request issuance, and market-based procurement. It provides intuitive navigation, structured data representation, and interactive features that empower DSOs to manage flexibility in a more efficient, transparent, and cost-effective manner.

Parallelly, an electric mobility platform has been developed to host charging stations and electric vehicles and to enable P2P energy flexibility trading. Web application front-end as shown in Figure 29 is based on Flutter v3, Flutter is an open-source UI software development kit created by Google. It can be used to develop cross-platform applications from a single codebase for the web, Fuchsia, Android, iOS, Linux, macOS, and Windows[9]. Flutter apps are written in the Dart language. Dart is a programming language used to develop web and mobile apps as well as server and desktop applications. Dart is an object-oriented, class-based, garbage-collected language with C-style syntax. It can compile to machine code, JavaScript, or WebAssembly. It supports interfaces, mixins, abstract classes, reified generics, and type inference[10] . All the computation logic is handled by the back-end server in Django, which the web app calls to get data and results. Django is a free and open-source, Python-based web framework that runs on a web server[11] .

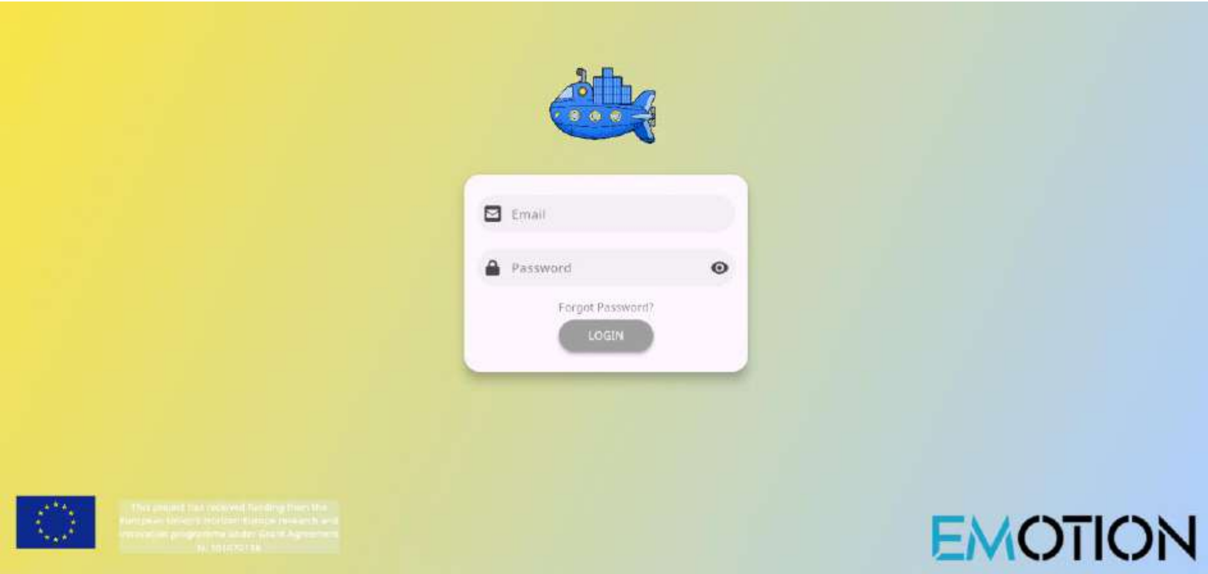


Figure 29 EMOT electric mobility platform login

[9] https://flutter.dev/
[10] https://dart.dev/
[11] https://www.djangoproject.com/

As shown in the Figure 30, each registered charging station is provided with the following technical details:

- Model;
- Country;
- City;
- Address;
- Maximum value of power output in alternate current;
- Maximum value of power output in direct current;
- Plugs connector type.

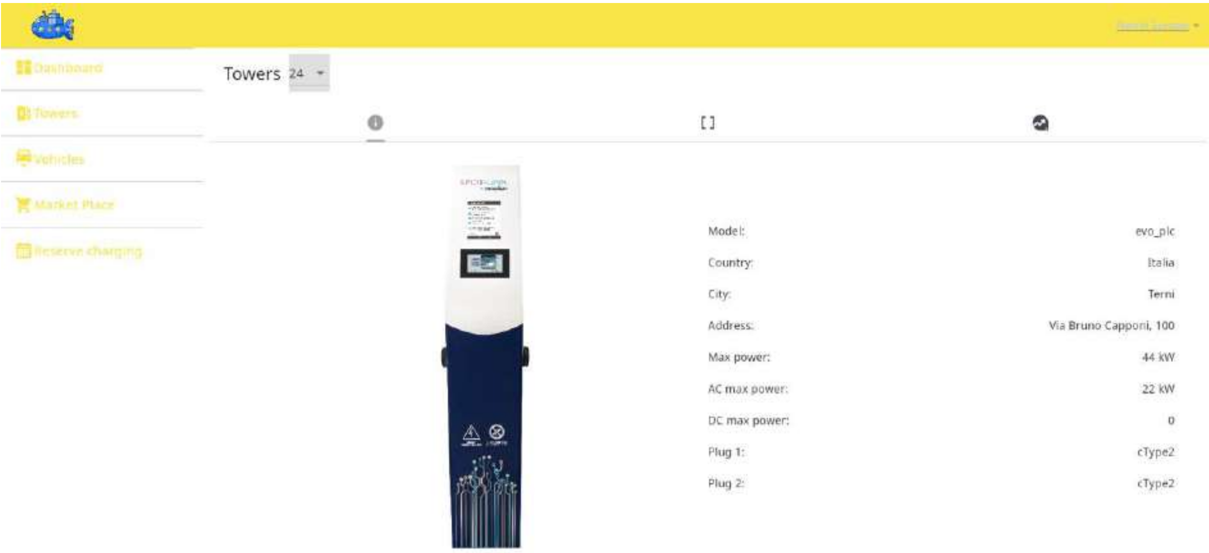


Figure 30 Charging station technical details in the Italian pilot electric mobility platform

As shown in Figure 31 Regarding electric vehicles, technical details include:

- Model;
- Battery capacity;
- Maximum value of charging power.

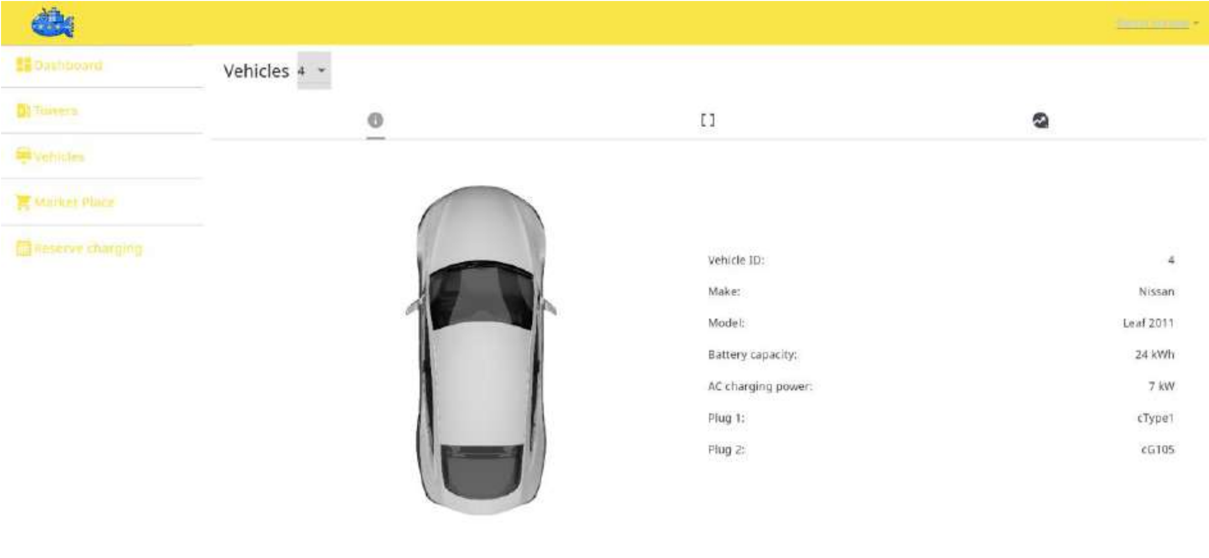


Figure 31 Electric vehicle technical details in the Italian pilot electric mobility platform

Data collected from charging stations and electric vehicles are accessible via electric mobility platform, both real-time and historical data as shown in Figure 32 Figure 33 Figure 34

Figure 32 Charging station historical data in electric mobility platform

Figure 33 Charging station real-time data in electric mobility platform

Figure 34 Electric vehicle historical data in electric mobility platform

Moreover, electric mobility platform integrates P2P energy flexibility marketplace to enable charging point operator to provide flexibility offers to distribution system operator as shown in Figure 35

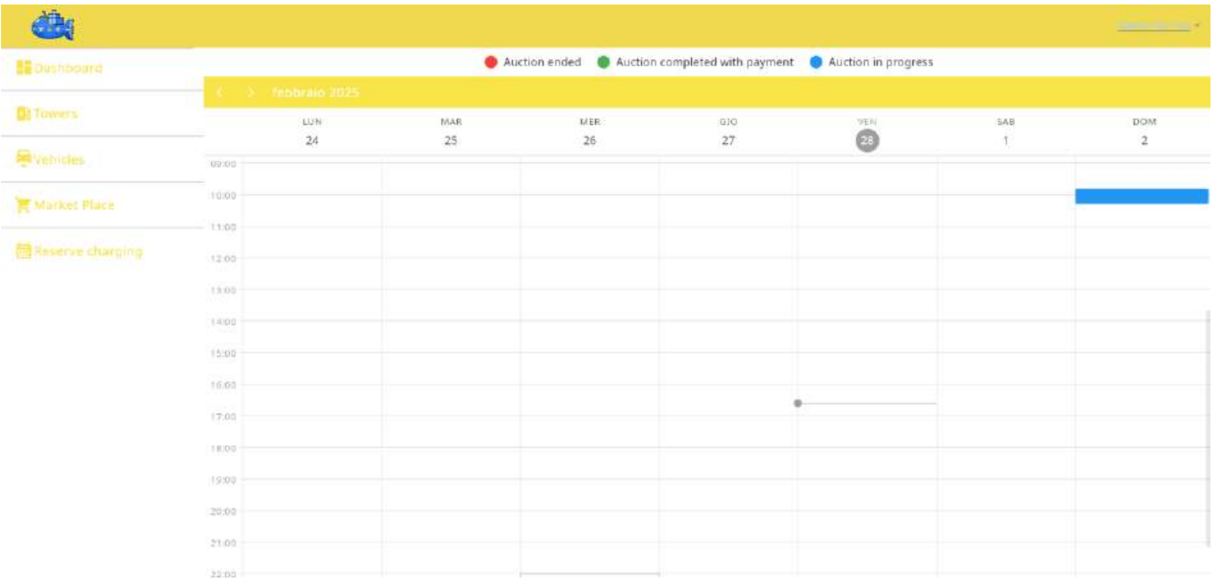


Figure 35 Marketplace calendar in the Italian pilot electric mobility platform

After DSO creates an energy flexibility request as shown in Figure 36, an indication of such creation appears on the electric mobility platform. CPO, following a verification of its ability to satisfy the energy flexibility request, can provide an offer in the dedicated section of the electric mobility platform. Thereby, charging point operator trades energy flexibility via the marketplace component in an automated, decentralized and flexible way using Ethereum smart contracts as shown in Figure 37 Figure 38 Figure 39

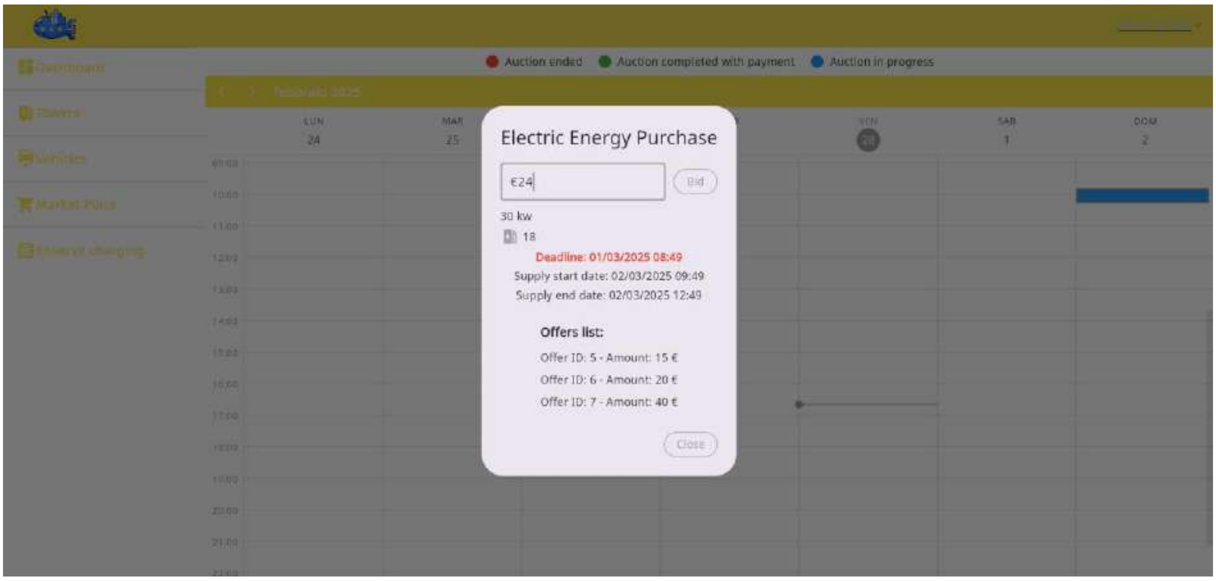


Figure 36 CPO energy flexibility offer in the Italian pilot electric mobility platform

```
❯ https emotion-projects.eu/marketplace/request/1 --verify=no
HTTP/1.1 200 OK
Content-Length: 306
Content-Type: application/json
Date: Fri, 27 Sep 2024 15:03:55 GMT
Server: Werkzeug/1.0.1 Python/3.10.14

{
    "deadline": "2024-09-25 16:45:00",
    "decided": null,
    "decision": [],
    "extra": [
        3,
        1727250911,
        19,
        78
    ],
    "from": "0×4005Ea3d4F2598b183A1680dbd9718dcb0941494",
    "id": 1,
    "offers": [
        1,
        2,
        3
    ],
    "state": "open"
}
```

Figure 37 DSO flexibility request in blockchain-based marketplace

```
HTTP/1.1 200 OK
Content-Length: 1071
Content-Type: application/json
Date: Fri, 27 Sep 2024 15:08:15 GMT
Server: Werkzeug/1.0.1 Python/3.10.14

{
    "offers": [
        {
            "author": "0x4005Ea3d4F2598b183A1680dbd9718dcb0941494",
            "extra": [
                100000000
            ],
            "id": 1,
            "request_id": 1,
            "state": "open"
        },
        {
            "author": "0x4005Ea3d4F2598b183A1680dbd9718dcb0941494",
            "extra": null,
            "id": 2,
            "request_id": 1,
            "state": "pending"
        },
        {
            "author": "0x4005Ea3d4F2598b183A1680dbd9718dcb0941494",
            "extra": null,
            "id": 3,
            "request_id": 1,
            "state": "pending"
        },
        {
            "author": "0x4005Ea3d4F2598b183A1680dbd9718dcb0941494",
            "extra": [
                23
            ],
            "id": 5,
            "request_id": 1,
            "state": "open"
        },
        {
            "author": "0x4005Ea3d4F2598b183A1680dbd9718dcb0941494",
            "extra": null,
            "id": 4,
            "request_id": 2,
            "state": "pending"
        }
    ]
}
```


Figure 38 CPO flexibility offers in blockchain-based marketplace


```
❯ https emotion-projects.eu/marketplace/info --verify=no
HTTP/1.1 200 OK
Content-Length: 181
Content-Type: application/json
Date: Fri, 27 Sep 2024 15:02:46 GMT
Server: Werkzeug/1.0.1 Python/3.10.14

{
    "contract": {
        "address": "0x7c4959b7Ab174aEC63eD0F1c36dAAEEaB8690d02",
        "network": "1720594411590"
    },
    "type": "eu.sofie-iot.EnergyMarketplace.energy"
}
```


Figure 39 Energy Flexibility Provision Smart Contract

Micropayment in token demonstration will be reported in D5.4, together with GUIs and marketplace containerized version deployed in NEMO environment.

Finally, WINDTRE has guaranteed connectivity between EMOT substations and the network with 5G router. Router installed are Telsey 5G in Figure 40 that guarantee connectivity to the substation in WI-FI mode.

Figure 40 Tesley Router

Telsey router parameters:

- **Mobile**

5G/LTE Cat.: 12 FDD-LTE DL up to 600Mbps, UL up to 150Mbps

- **Wi-Fi**

802.11b/g/n/a/ac/ax, 2.4 GHz or 5 GHz, up to 16 client connected

WINDTRE has analysed 5G coverage in trial area as shown in Figure 41 Figure 42 to verify level of signal, antenna TR005 is bts that ensures 5G coverage in ASM Terni, TDD and FDD are available.

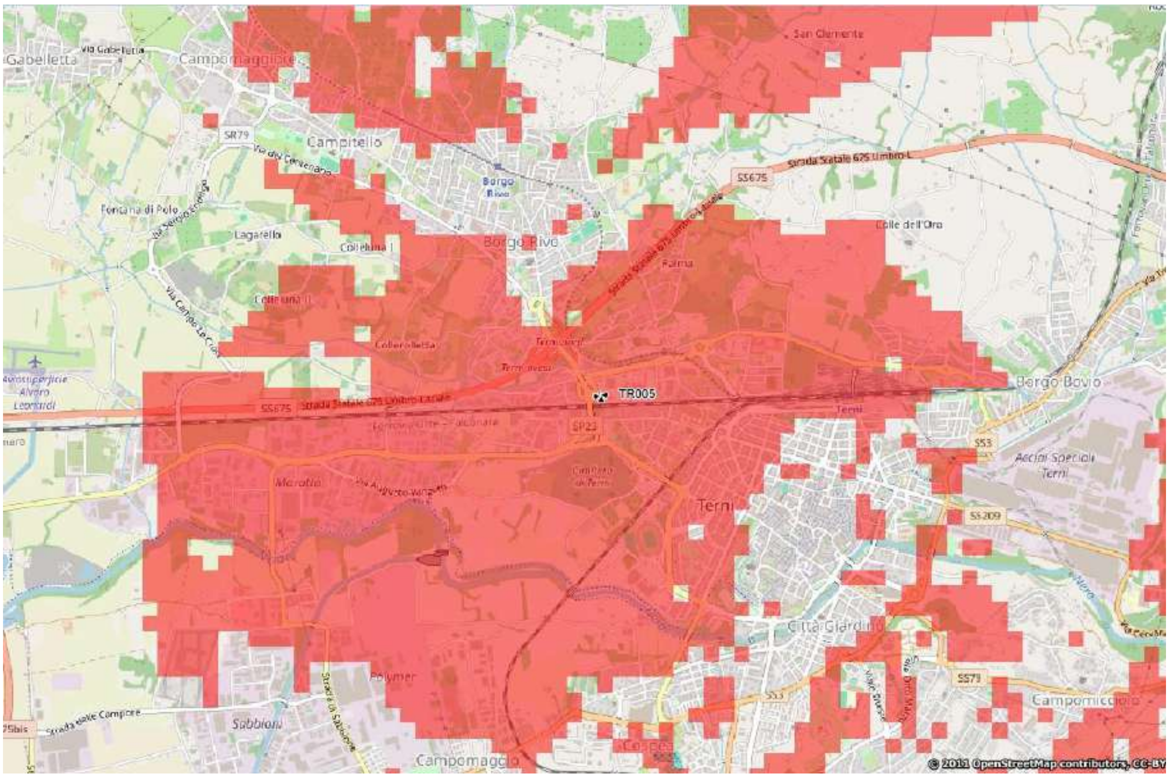


Figure 41 W3 TDD coverage

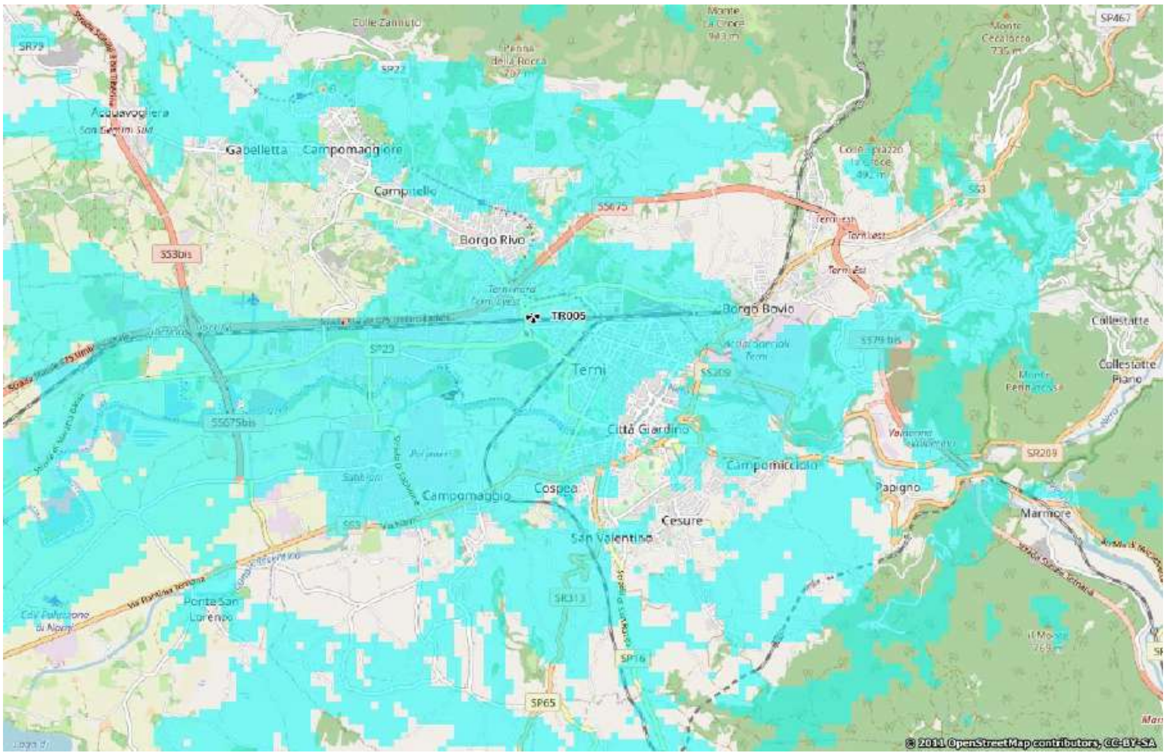


Figure 42 W3 FDD coverage

EMOT has required to use connectivity with public IP address therefore the following APN is configured: *myinternet.wind*.

4 Smart Manufacturing & Industry 4.0 Trial

4.1 Introduction

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, CONTI has already undergone feasibility and profitability assessment for the application of flexible Cobots in upcoming assembly facilities. In parallel, CONTI 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 leadplant of CONTI for advanced driver assistance systems (BU ADAS) is located in Ingolstadt. The plant has a high level of automation in manufacturing and logistics as well as high specialization in Cobots and AGVs. Innovative manufacturing processes are set up in Ingolstadt for the first time and upon successful verification are transferred to the other ADAS sites.

4.2 Use Case Applications

CONTI considers two applications to be implemented, relevant to validating NEMO, as follows.

SM_01: Fully automated indoor logistics/supply chain: This Use Case 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 (one Cobot and several types of AGVs), CONTI aims to fully automate controlled material picking from Auto Store and autonomous transfer to the production line.

SM_02: Human-centred indoor factory environment safety. This Use Case 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 uploads to the edge cloud facilities, where AI functions will detect the position of each body and build a "safety shell" around it to ensure human-centred 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.

4.2.1 Trial Site Description Updates

The Automated Sorting and Booking Station (ASBS) is a first of this type of artificial intelligence (AI) supported technology in Continental to ensure fully automated material supply that completely covers the entire demand from AutoStore for ADAS Plant Ingolstadt.

The technology is meanwhile successfully tested at the smart technology application room in Ingolstadt. With a target of at least 3000 picks per day, it offers a substantial improvement over manual picking process. Additionally, it ensures a complete touchless material flow and thereby technical cleanliness. A unique combination of a Light Field Camera system in combination with image evaluation with neural networks allow for a 100 % fulfilment of the AutoStore demand for ADAS Plant Ingolstadt. One of the challenges in identifying the materials is that unfortunately there are no awareness of what kind of material is in the box placed from the AutoStore. Glossy film packaging and transparent reels brings classic 3D-sensors (Triangulation) to its technical limits. This could not be set to known and shared sensors. After a thorough market analysis, an evaluation of a light field sensor from a startup company named HD Vision was done. This sensor uses the physical conditions of light field refraction and thus also recognizes transparent or reflective surfaces. HD Vision provided a stable and secure solution. Since not only reels but also Printed Circuit Board packages and various mechanical components (e.g. lenses, flexcables, etc.) are stored in the AutoStore, standard grabbers and suction systems available on

the market were not suitable. A new tool has been developed and purpose - built in house. Also, the project automated the booking in SAP from the warehouse to production.
During image evaluation, the material label is read and transmitted to SAP. The business software developed by the central 4.0 team for sorting incoming goods is extended also for outgoing goods by adapting the middleware.

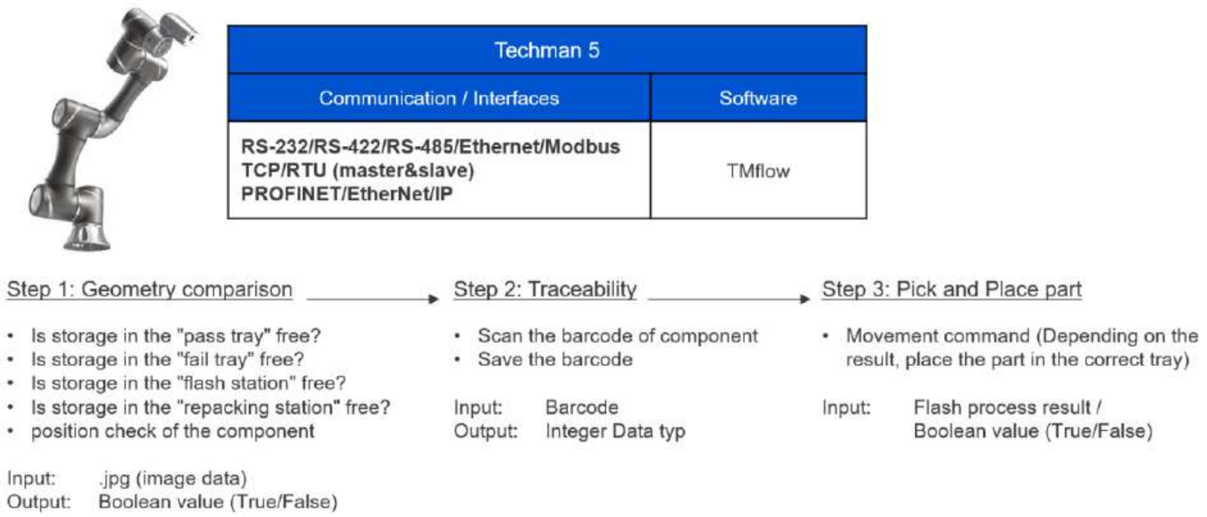
In the last step, the robot places the reels inside a transfer box. The robot always has two free transfer boxes for placing the materials, if a new box is needed for a new destination, it is provided via one of the two box towers. A total of ten sorting targets are available to the robot. When the robot is done, the transfer box is made available for further transport for the automated Autonomous Guided Vehicle (aAGV) via a conveyor line. The transfer boxes are then automatically transferred to the aAGV which brings it to production.

4.2.2 Technical validation

The technical validation of the Trial #4 is done in Continental Ingolstadt in the Smart Technology and Application Room with the involvement of the technical experts and those responsible from the relevant areas such as logistics and production as well as the employees involved from the warehouse, material planning and the operators.

The central process "Bin Picking / Pick and Place" in the Use Cas Smart Industry 4.0 is recreated in the STAR - Smart Technology & Application Room to test the NEMO integration. The hardware setup for the demonstration has been completed.

Figure 43 Hardware set up at CONTI Living lab



The Smart Manufacturing trial's technical validation will be carried out through the test scenario detailed in the table below, addressing both SM_01 and SM_02

Scenario: SM01_Test_Scenario_1	
Scenario ID	SM01_Test_Scenario_1 (SM01_TS01)
Objective	• More efficient use of employees and improved ergonomics • More flexible material removal • Better just-in-time delivery (no longer clocked in 30 min.) • Expansion to include backend production (till date only front end)
Description	The current situation refers to an inefficient operation because the materials are removed from the AutoStore every 30 minutes manually by an operator. For this, employees have to interrupt their other work. The process is not ergonomic and very time consuming. The future situation, generated by the Use Case trial, corresponds to a fully automated indoor logistics with focus on human and environment safety.
Features to be tested	• Correct component recognition in AutoStore • Correct gripping / removing of the components • Efficient and effective transfer to AGV • Required communication between robot and AGV The above functionalities will be tested within the following components of the NEMO platform: • AIoT Architecture (Cybersecure distributed learning framework, Federated meta Network Cluster Controller, micro-services, Cybersecurity & Unified/Federated Access Control...) • IoT/5G Time Sensitive Networking (TSN) • SEE and SLO meta-Orchestrator • MLOps (FL, model storage, model sharing) • Meta-Orchestrator, PPEF, CFDRL (Observability, Workload deployment & migration, limit execution within a cluster set) • CMDT (workload discovery) • Intent-based API (app and resource selection) • LCM (app deployment and LCM visualization) • IdM & Access Control (users)

Scenario: SM01_Test_Scenario_1	
Requirements addressed	• **SM_01_FR01:** Bin Picking: The platform must ensure secure part recognition and handle position determination (no component destruction) • **SM_01_FR02**: Camera: Must ensure recognition of black parts in black trays • **SM_01_FR03:** Bin picking: The application should provide a sufficiently short cycle time • **SM_01_FR04:** Bin Picking: The application must ensure error-free component assignment / QR code read
KPIs	• **KPI_SM_01_2:** Different types of sensors' data to be analysed (> 10) • **KPI_SM_01_3:** System reaction in emergency cases (< 0,5 sec) • **KPI_SM_01_4:** ADAS supply chain improvement: accuracy (> 30 %) • **KPI_SM_01_5:** Cost and time reduction (> 20%)
Prerequisites	Facility conditions at the Ingolstadt location due to the historical structural development of the production and logistics areas present challenges and limitations in the implementation of automated material transport. Innovative and intelligent solutions are being sought. There is a well-organized process flow in place, which allows the full automation of material supply. The implementation and evaluation of the NEMO solution comprise several development phases and complexity steps. The NEMO platform should be installed and configured. Users are allowed to be registered and signed in with relevant access. USER_1 uploads their Smart Industry App in NEMO, available for deployment and access as NEMO partner. USER_2 should be registered as NEMO cluster provider, having provided its resources in NEMO, and as consumer, wishing to deploy the Smart Industry app in its resources only through NEMO. FL training should be supported for at least one ML task in the Smart Industry application. The clusters should be configured appropriately to be integrated into NEMO resources and be able to run local training tasks as FL participants. The USER_1 should be able to initiate an FL training task. Also, 1 user is registered as meta-OS Provider (ADMIN).
Test steps	Step-by-step implementation of the trial: 1. Automatic component removal from storage 2. Automatic component provision to AGV 3. Extension to backend materials (optional) On the Nemo platform there are a few steps to test: <u>Workload registration</u> 1. USER_1 signs in NEMO as meta-OS consumer. 2. USER_1 uploads their application into NEMO and is notified about the result. 3. As soon as the upload is successful, it will be available for third parties to deploy and use it. 4. USER_1 enables and configures the plugin that allows FL training to be initiated for an ML task in their application. 5. USER_1 visualizes lifecycle data and credits through the NEMO dashboard.

Scenario: SM01_Test_Scenario_1	
	<u>Resource onboarding</u> 1. USER_2 signs in the NEMO dashboard and accesses the resource onboarding functionality. 2. USER_2 selects the option to add their resources into NEMO and provides relevant info. 3. USER_2 is notified about the result through the dashboard. 4. USER_2 is informed about their credits. <u>Workload execution</u> 1. USER_2 accesses the workload execution functionality in the NEMO dashboard. 2. USER_2 selects the Smart Industry app to be executed in their resources only. 3. USER_2 starts to run the Smart Industry app and select the pictures of the electronic devices that will be used from training 4. USER_2 enables FL training for the app's ML tasks. <u>FL training</u> 1. USER_1 sends a request for FL training for their app. 2. The FL training process is executed with the user-defined parameters (rounds, epochs, FL participants, etc.) 3. The training process completes and the aggregate model is stored. 4. USER_2 is notified about the newly trained model. 5. USER_2 saves the pictures and the trained model into the storage. <u>Model sharing & deployment</u> 1. USER_2 sends a request through the NEMO dashboard to deploy the new model on their edge. 2. USER_2 checks that new model is deployed on their devices through the dashboard.
Success state	The first three trial steps have successfully been realized. These steps contribute mainly to the first application "Fully automated indoor logistics/supply chain" In general, the project progress can be assessed as positive. So far, the set trial goals have been achieved. The trial is running according to schedule. The success rate will be measured according to the KPIs.
Failure state	Currently, both applications corresponding to the trial work according to the planning.
Responsible for testing and implementation	The core team for testing and implementation are the employees in Smart Technology & Application Room in strong cooperation with technical experts and those responsible from the relevant areas such as logistics and production as well as the employees involved from the warehouse, material planning and the operators.
Risks	• Automatic component recognition □□□Failure • Automatic picking process □□ Scrap

Table 8 Test scenario SM 01

Scenario: SM02_Test_Scenario_1	
Scenario ID	SM02_Test_Scenario_1 (SM02_TS01)
Objective	• Improve safety of operators working on the manufacturing processes • Smart detection of the position of each body and build a human-centred safety environment around it • More efficient use of employees and improved ergonomics
Description	This scenario is focused on localizing the AGVs with high precision, by collecting and processing real-time data from smart sensors and devices (safety cameras, radar and Lidar). The private wireless network (TSN) will support great amount of data uploads to the edge cloud facilities, where AI algorithms will detect the position of each body and build a "safety shell" around, thus ensuring the human-centred safety.
Features to be tested	• Identifiying the position of AGVs within the manufacturing processes • Detecting the position of each body and build a safety area around it • Processing the real-time data in order to avoid collisions between AGVs, or between a worker and an AGV The above functionalities will be tested within the following components of the NEMO platform: • AIoT Architecture (Cybersecure distributed learning framework, Federated meta Network Cluster Controller, micro-services, Cybersecurity & Unified/Federated Access Control...) • IoT/5G Time Sensitive Networking (TSN) • SEE and SLO meta-Orchestrator
Requirements addressed	• **SM_02_FR01:** The application should provide information about the localization of AGVs • **SM_02_FR02:** The application must provide information about the localization of the human worker • **SM_02_FR03:** The application has the capability of detecting / identifying the human body • **SM_02_FR04:** The application will send alerts in case of potential collisions between human workers and AGVs
KPIs	• **KPI_SM_02_1:** Different types of AGV and Cobots to be addressed (> 4) • **KPI_SM_02_3:** Improve human collision avoidance and manufacturing safety (30 %) • **KPI_SM_02_4:** Cost and time reduction (> 20%)
Prerequisites	Facility conditions at the Ingolstadt location due to the historical structural development of the production and logistics areas present challenges and limitations in the implementation of automated material transport. Innovative and intelligent solutions are being sought. There is a well-organized production workflow in place, which involves both human staff and robots. Data are collected via AGVs and Cobots and the devices are connected to the NEMO platform.
Test steps	Step-by-step implementation of the trial: 1. Automatic material transport to production, on the same floor 2. Automatic material transport to production on another level (floor jump) 3. Extension to backend materials (optional)

Scenario: SM02_Test_Scenario_1	
Success state	The current focus of the trial is in the second application “Human-centred indoor factory environment safety” addressing the step 2 Automatic material transport to production on another level (floor jump). The Benchmark for a suitable technology was already evaluated with the SERVUS-System
Failure state	Currently, both applications corresponding to the trial work according to the planning.
Responsible for testing and implementation	The core team for testing and implementation are the employees in Smart Technology & Application Room in strong cooperation with technical experts and those responsible from the relevant areas such as logistics and production as well as the employees involved from the warehouse, material planning and the operators.
Risks	No risks identified so far

Table 9 Test scenario SM 02

4.2.3 Use case diagrams

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 Sensorik system for assessing the working environment.

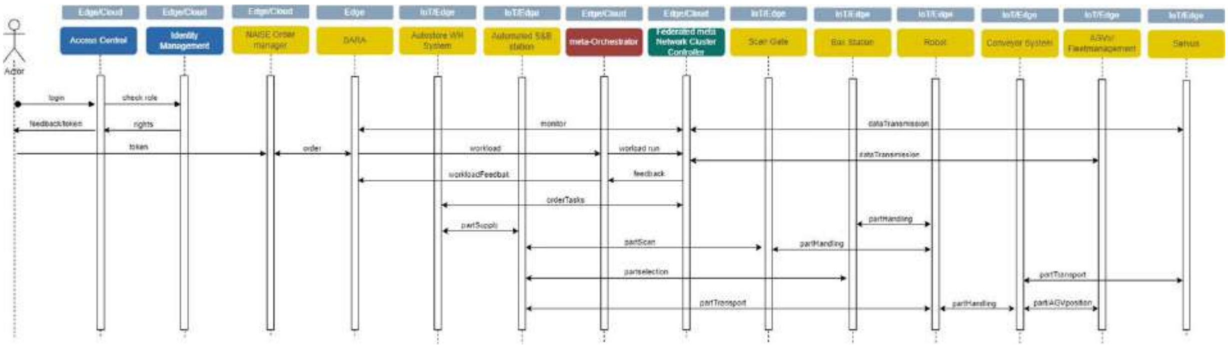


Figure 44 Sequence diagram for application execution in SM01_Test_Scenario_1

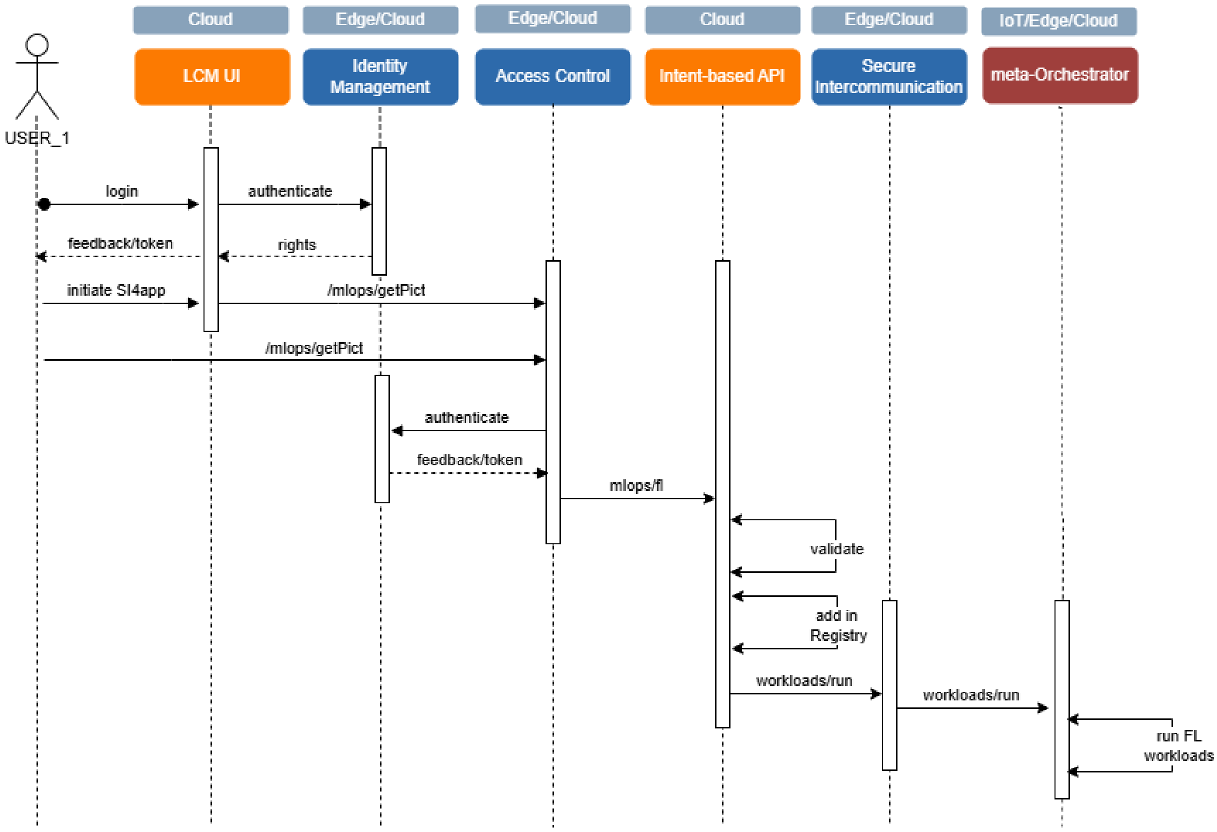
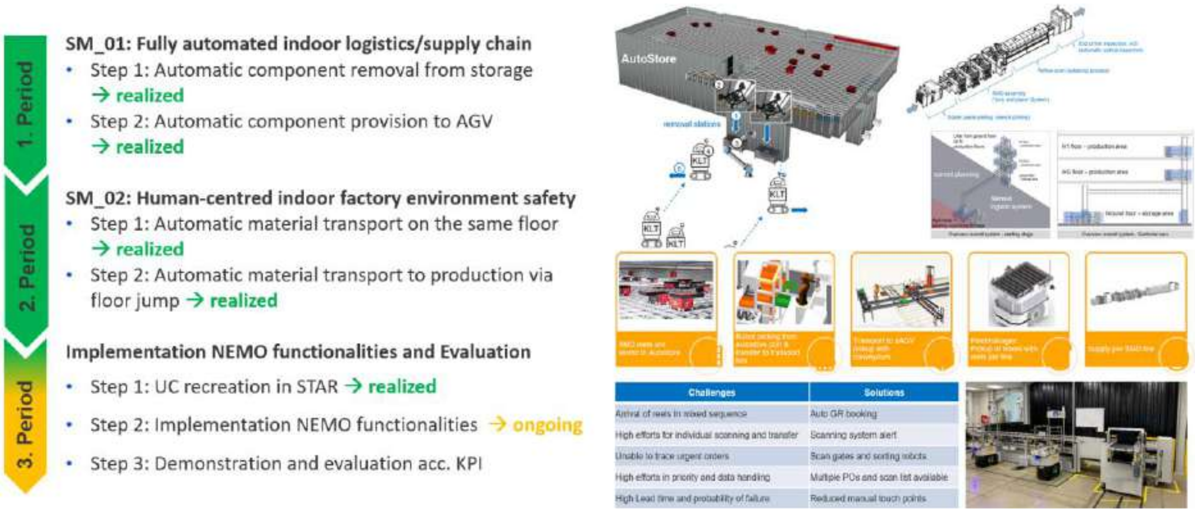


Figure 45 Sequence diagram for FL training in SM01_Test_Scenario_1

4.2.4 Timeline of activities



SM_01: Fully automated indoor logistics/supply chain	status
SM_01_FR01: Bin Picking: The platform must ensure secure part recognition and handle position determination (no component destruction)	✓
SM_01_FR02: Camera: Must ensure recognition of black parts in black trays	✓
SM_01_FR03: Bin picking: The application should provide a sufficiently short cycle time	✓
SM_01_FR04: Bin Picking: The application must ensure error-free component assignment / QR code read	✓
KPI_SM_01_2: Different types of sensors' data to be analysed (> 10)	✓
KPI_SM_01_3: System reaction in emergency cases (< 0,5 sec)	⏲
KPI_SM_01_4: ADAS supply chain improvement: accuracy (> 30 %)	⏲
KPI_SM_01_5: Cost and time reduction (> 20%)	✓

SM_02: Human-centred indoor factory environment safety	status
SM_02_FR01: The application should provide information about the localization of AGVs	⏲
SM_02_FR02: The application must provide information about the localization of the human worker	⏲
SM_02_FR03: The application has the capability of detecting / identifying the human body	✓
SM_02_FR04: The application will send alerts in case of potential collisions between human workers and AGVs	✓
KPI_SM_02_1: Different types of AGV and Cobots to be addressed (> 4)	✓
KPI_SM_02_3: Improve human collision avoidance and manufacturing safety (30 %)	⏲
KPI_SM_02_4: Cost and time reduction (> 20%)	⏲

4.2.5 Intermediate Results

Two applications are planned to be demonstrated, validating NEMO. These refer to the fully automated indoor logistics/supply chain and the human-centred indoor factory environment safety. The concept for both applications has been developed. The Automated Sorting and Booking Station (ASBS) scans, sorts and handles all reels, drypacks and PCBs automatically. The robot sorts the reels into boxes with the help of scanning stations, and these reels can then be sent to the warehouse or production. The ASBS consists of conveyor system to transport parts, scan gate to register them, robot to sort and put away the parts, and box station for automatic supply of boxes. The process starts with the image recording of the material in the AutoStore box. The camera software transfers all relevant data to the PLC. The data includes the position of the material, the size of the material and the label information. The size of the material is relevant so that the gripper is set to the correct size. Also, the project automates the booking in SAP from warehouse to production. In the last step, the robot places the reels inside a transfer box. The robot always has two free transfer boxes for placing the materials, if a new box is needed for a new destination, it is provided via one of the two box towers. A total of ten sorting targets are available to the robot. When the robot is done, the transfer box is made available for further transport for the aAGV via a conveyor line. The transfer boxes are then automatically transferred to the aAGV. The orders for the aAGV are created via the NAiSE Order manager. The Servus transport system to realize the floor jump has also been implemented.

Challenges	Solutions
Arrival of reels in mixed sequence	Auto GR booking
High efforts for individual scanning and transfer	Scanning system alert
Unable to trace urgent orders	Scan gates and sorting robots
High efforts in priority and data handling	Multiple POs and scan list available
High Lead time and probability of failure	Reduced manual touch points

Table 10 Challenges emerged and solutions developed

5 Smart Media & XR Trial

5.1 Introduction and Use case Applications

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.

Spectators can enhance their contributions and interact with other users in response to specific race incidents. The trial emphasizes real-time, user-generated content processing and rendering, leveraging Federated Learning (FL) hosted across IoT nodes (smartphones), edge devices, and cloud infrastructure. AI models will be trained to recognize Racing Bib Numbers, street numbers, and race landmarks, enabling better identification of runners and their precise positioning within each video stream.

The trial takes place in both a controlled stadium environment and its surrounding areas, with mobile base stations providing extended network coverage several kilometers beyond the stadium. The deployment involves both commercial networks and a dedicated network layer built on top of existing infrastructure. The trial infrastructure includes two compute nodes and one control node for deploying the trial's components as virtual machines, with an additional GPU available to support AI computations.

Key capabilities explored in this trial include:

- Geolocation and identification of key places.
- Real-time race situation analysis, depending on the final pilot scenario.
- Integration of wired cameras for enhanced coverage.
- The potential inclusion of drones to capture additional media perspectives.

Usually Cloud-Edge-IoT solutions target the industrial sectors. But in doing so we ignore the fact that any new technology needs also citizen acceptance, therefore we should also look into solutions for media and entertainment. To showcase the potential of cloud technologies in these areas two use cases were created as part of the NEMO project use case trials. The Smart Media & XR Trial will feature three use cases that span sports and Culture. The use case's location will be Athens Greece. The main objectives are:

- To validate the NEMO for live events in Smart/Media city environments where we have many user, live content and the need of 5G usage.
- To validate it for Extended Reality environment where we need high interactivity, low latency and high accuracy for gesture/emotion analysis.

5.2 SC_01 Round of Athens Race

5.2.1 Trial Site Description and Updates

The Trial stadium description is well documented in D5.2 [5] The trial will take place outdoors. The use case will be hosted in the Municipality stadium of Egaleo in Athens and we will deploy a hybrid scenario where part of the race will happen inside the stadium and the other half outside. This is done on purpose to stress the KPI specified and to test in a more "realistic" environment.

A survey for any 5G weak spots is already performed but OTE's mobile network on the day of the pilot will be enhanced with mobile stations as to provide the best possible 5G coverage. The computing resources needed for Edge and Cloud nodes will be hosted on the OTE premises that are also based in Athens.

Components will send metadata information to a central Information bus, where other components will retrieve this information. Mobile app will consume this data to enhance the user experience of users, by providing visual content, in addition to runner's detection, quality of experience values and other useful information.

5.2.2 Technical validation

The Smart Media trial's technical validation will be carried out through the test scenario detailed in the table below, addressing Use case SC_01

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 using FL hosted locally on the IoT nodes (smart phones), in the edge and at the cloud. The AI/Models will be trained to recognize the Racing Bib Numbers on each athlete and street numbers and landmarks of the race 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
KPIs	KPI_SC_01_1-7

Scenario: SC01_Test_Scenario_1	
Prerequisites	The NEMO platform should be accessible and running. More than 10 users should be available for using the App and the Trial infrastructure. More than 5 video feeds as sources available at any time. Sources can be Cameras located at the event, Drones, Runner cams and user content provided through the App. A 4-5G network coverage should exist at all places in order to provide high bandwidth and multipath connectivity. Stadium and location for run. User with Phones and App installed. Media Managers for receiving streams and distribution of processed streams available through NEMO components. Production Control system for stream authoring as NEMO plugin service. An AI component as a NEMO resource for automatic annotation and authoring. Network Probe for Media adaptation on network quality.
Test steps	NEMO IS deployed and operational. This includes: • 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. • GoPro cameras on runners, smartphone cameras of spectators, (optional) professional and drones IP cameras are connected to platform and send real time data. • The platform users (professionals) are registered to the platform. Users open App and register as meta-OS consumer. Additional cameras and Runner Cams get ready and broadcast as met-OS providers. Production control commences and AI component start analyzing video feed at the same time NEMO collects monitoring data and workloads. When power of AI reaches a predefined limit, NEMO undertakes reallocation to other devices. Users interact 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 have access to enriched content that includes: • 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. • General overview of the event situation (classification, groups, time advantage, etc.). • Runners’ numbers recognition for race positioning tracking.
Failure state	AI component fails to identify runner BiB numbers. Production control is not optimal and does not provide qualitative stream. App fails to stream or view specific events. GPS signal is not enhanced.

Scenario: SC01_Test_Scenario_1	
Responsible for testing and implementation	UPM, NOVO,OTE, TID
Risks	• High computing resources. • High network latency. • High video latency (both for technical director and video delivery). • Synchronization of events with media content. • Right access to content with privacy preservation. • Quality of Experience (QoE) optimization.

Table 11 Test scenario for SC 01

5.2.3 Use case sequence diagrams

The use case diagrams presented in deliverable D5.2 [5]provide a detailed description of how the components interact with each other. Only the updates are documented below:

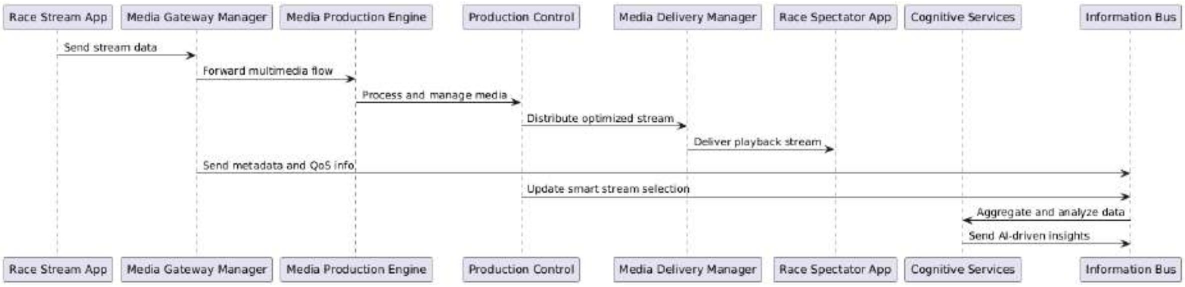


Figure 46 Use case sequence diagram for SC 01

5.2.4 Timeline of activities

Phase	Start Date	End Date	Notes
Deliver of beta version of components. Local Testing.	M5	M29	Completed
Testing Deployment on OTE Servers in Athens. Testing media round trip.	M29	M31	In Progress
Integration with NEMO	M30	M32	In Progress
Final Pilot Tests in Athens	M33 (7,8 May 2025)	M33 (7,8 May 2025)	To be performed

Table 12 Timeline of activities SC 01

5.2.5 Intermediate Results

The Intermediate results of this Use case is documented below, other than what was documented in the previous deliverable, the intermediate results include the progress on the developments from all the partners involved in the Use case and the integration status of the Use case specific components. Figure 47, Figure 48 are related to the Production control deployment on Proxmox in Greece and Connection with remote studio in Spain.

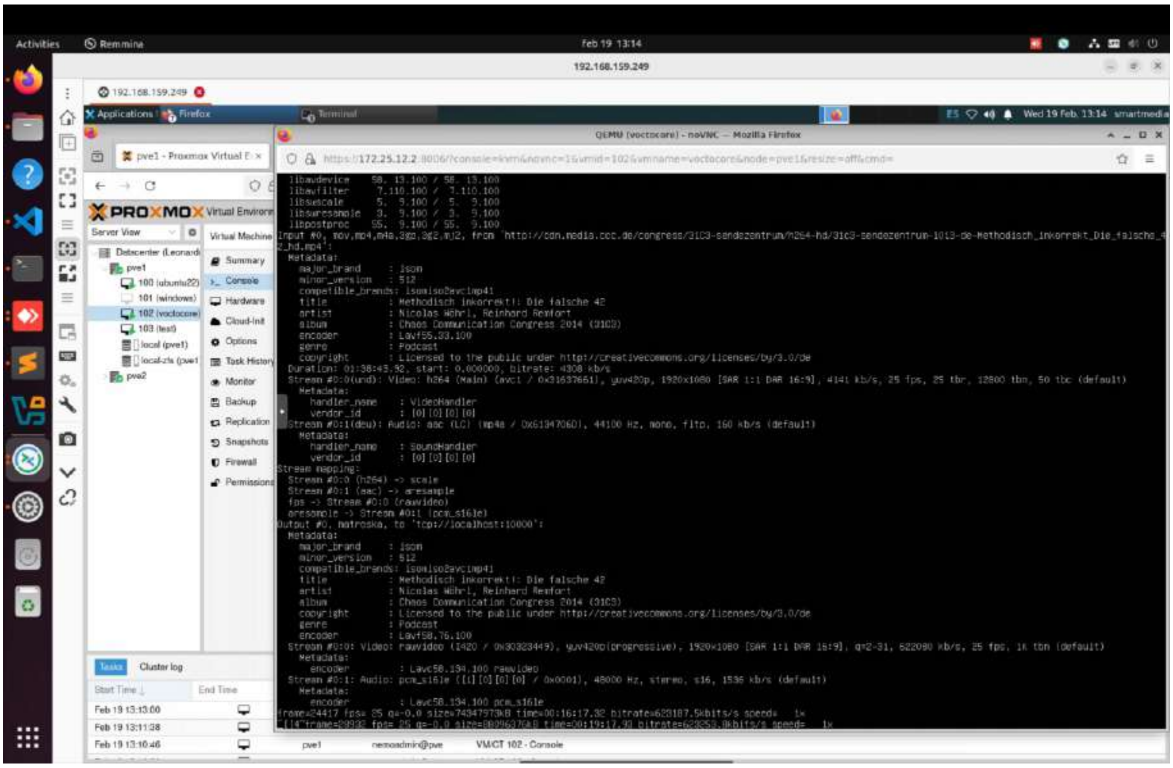


Figure 47 Production control deployment on Proxmox in Greece

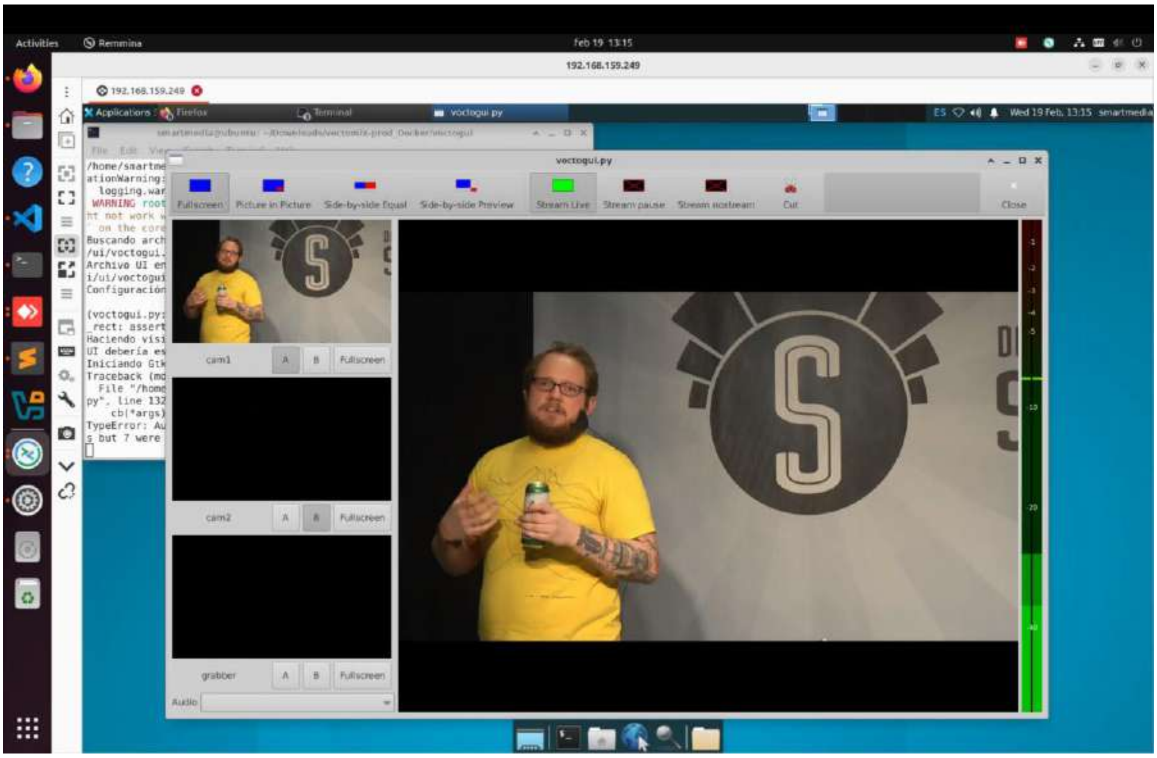


Figure 48 Connection with remote studio in Spain

Other components (AI Engine, Network Probe, and Media Production Engine) have been tested locally per partner and prepared to be deployed via NEMO Kubernetes on OTE premises once the connection between Proxmox (Greece) and TID Lab (Spain) have tested network configuration.

Finally, a RTMP server will deliver the information to the users in the mobile app as shown in Figure 49. Specifically,

- **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.

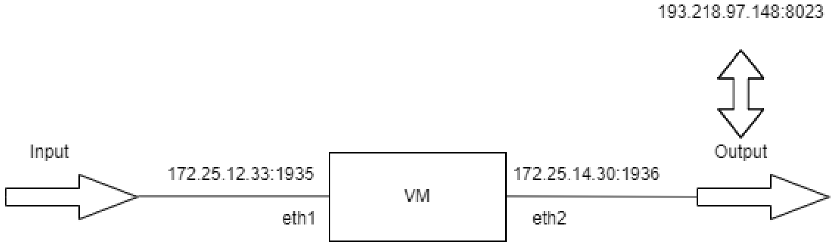


Figure 49 Schema of interaction between RTMP server and mobile app

NWDAF

NWDAF is a network function that collects data from different network functions (data providers) in 5G core and performs data analytics and provides an insight to consumer network functions.

NWDAF represents operator managed network analytics logical function. The NWDAF includes the following functionality:

- Support data collection from NFs and AFs;
- Support data collection from OAM;
- NWDAF service registration and metadata exposure to NFs/AFs;
- Support analytics information provisioning to NFs, AF.

The details of the NWDAF functionality are defined in TS 23.288, TS 29.520

Exposure of analytics: NWDAF analytics may be securely exposed by NEF for external party, as specified in TS 23.288.

The following events can be subscribed by a NF consumer (Event ID is defined in clause 4.15.1):

- Service Experience information, as defined in clause 6.4.2, TS 23.288.
- UE Mobility information, as defined in clause 6.7.2.2, TS 23.288
- UE Communication information, as defined in clause 6.7.3.2, TS 23.288
- Exceptions information, as defined in clause 6.7.5.2, TS 23.288

Retrieval of data from external party by NWDAF: Data provided to the external party may be collected via NEF for analytics generation purpose.

The application from NOVO

The NOVO app will be developed for the Android OS, targeting a wide range of devices running Android.

1. Technology Stack: For the frontend development, React Native will be used. React Native is a popular framework for building cross-platform mobile applications using JavaScript and React. It allows for faster development and better performance compared to traditional native app development.
2. Backend Development: For the backend infrastructure, Laravel will be used. Laravel is a PHP framework that provides a robust set of tools for building web applications. It is known for its simplicity, elegance, and ease of use. Laravel will be used to develop the APIs and manage the database for the NOVO app.
3. UI/UX Design: The UX/UI design of the NOVO app will follow modern user-friendly guidelines to ensure that users can easily navigate through the available functions and contribute to the event. The design will be intuitive and visually appealing, with a focus on simplicity and ease of use.

Hereby, we provide NOVO's mobile app wireframe, outlining its key screens and functionalities.
The home screen, illustrated in Figure 50Figure 50, features fundamental action buttons:

1. Watch live streaming.
2. Capture.
3. Upload from disk.

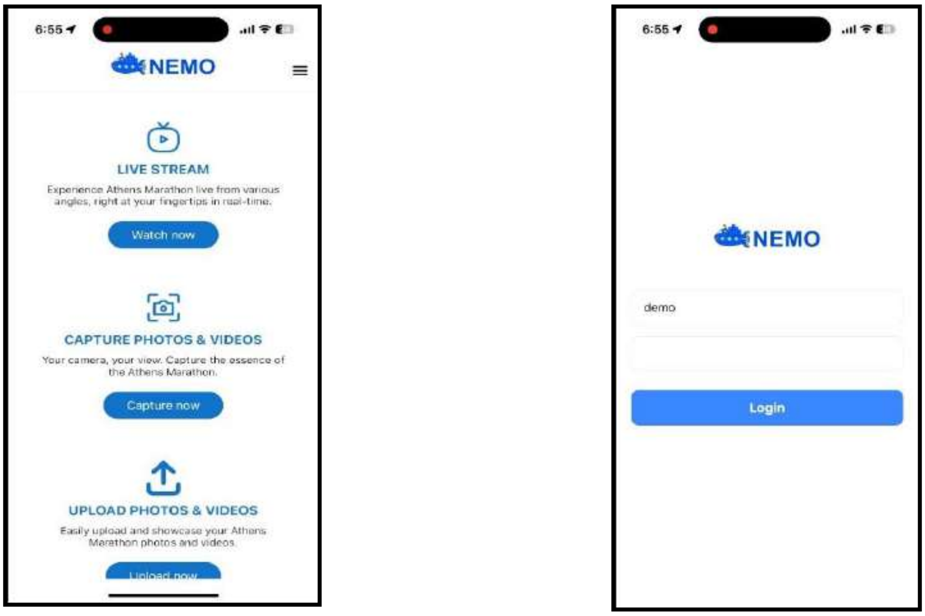


Figure 50 NOVO app Home Screen design and Navigation Menu design

The Navigation menu, illustrated in Figure 51, encompasses basic user actions.
The Streaming interface, illustrated in Figure 51, allows users to select the streaming location/camera.

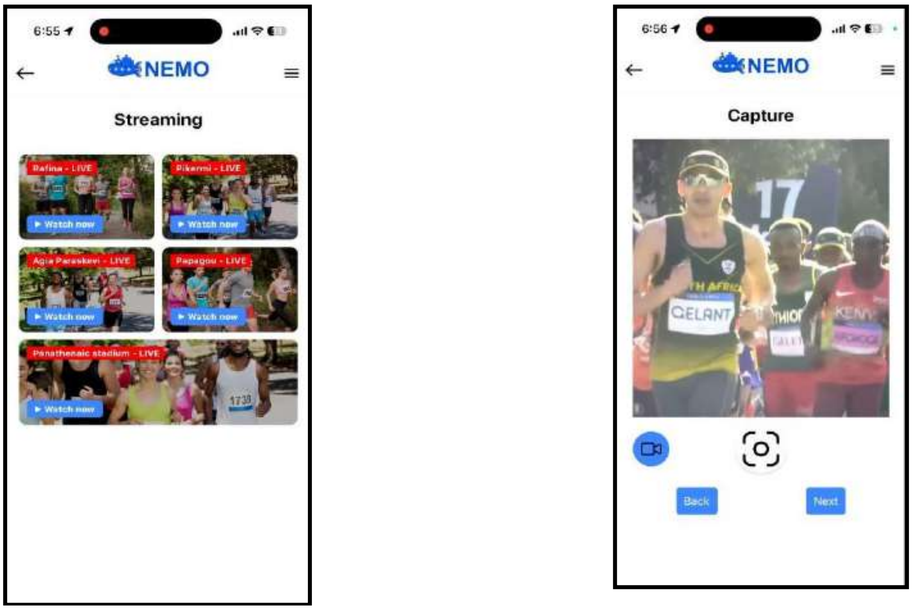


Figure 51 NOVO app Streaming interface and Capture interface

The Capture interface, illustrated in Figure, guides users through the process of capturing content:

1. Choose location (automatically detect through GPS).
2. Take picture from camera.
3. Insert comment/annotate.
4. Upload.

Lastly, the UI flow diagram of the application is illustrated in Figure 52

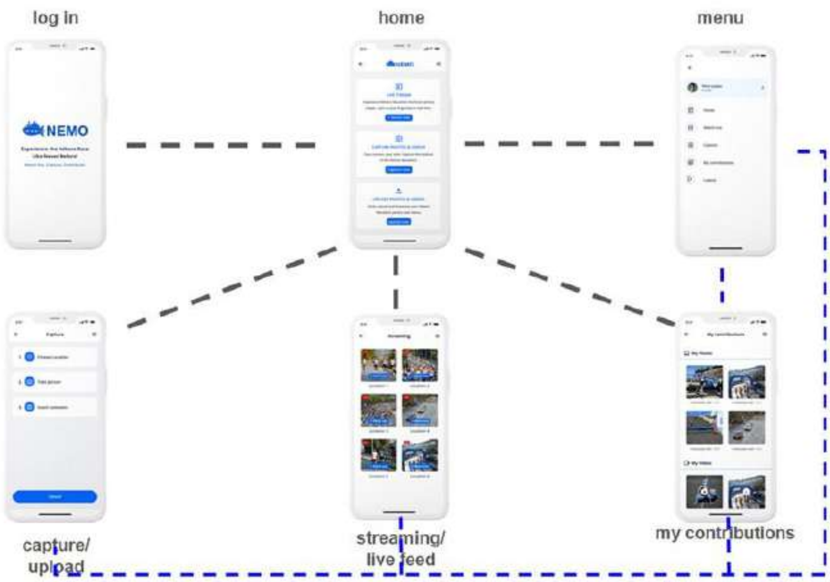


Figure 52 NOVO app flow diagram

5.3 XR_01 VR Experience about ancient Workshop of sculptor Phidias

5.3.1 Trial Site Description and Updates

The XR use cases focus on enhancing educational VR experiences. The pilot will be hosted at the Hellenic Cosmos Cultural Center of the Foundation of the Hellenic World, which is based in Athens and features exhibitions, educational programs and public access to single person systems using Head Mounted Displays.

This use case will enhance the experience of a Head Mounted Display VR application that presents everyday life in the workshop of a famous sculptor Phidias. Biometric Data of the user as he interacts and navigates the virtual environment will be captured by an IoT Smart Watch and then analyzed on a AI/ML node on the edge-cloud continuum to estimate the Emotional status of the individual. This status will allow the application to adapt the experience or even notify museum staff is assistance is required in case of nausea. The architectural diagram Figure 53 and Table 13 depicted below shows the final software that was used to create the components and its interconnection.

Emotion Recognition ML App	**Input data**: Smart Watch (Samsumg Galaxy 5) Application based on 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
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	**Input:** Inference from ML Applications (HTTP REST API) **Technologies:** Docker/Kubernetes (Helm), Django 5.0.1
Message Broker	**Input:** Data from IoT devices (wearables, cameras etc) **Technologies:** RabbitMQ /Docker/Kubernetes
Data Lake	**Input:** Data from IoT devices **Technologies:** InfluxDB 2.7.5 / Postgres 16.1.0
VR Applications	Unity 3D - VR Application for Dome Theatres (PC) Unity 3D - VR Application for Head Mounted Displays (Meta Quest 2 - Android)
Smart devices	Unity 3D - 2D PC application for Smart TV output Unity 3D – 2D Android Application for Smart Phone

Table 13 Listing of Software used to implement the architectural diagram below

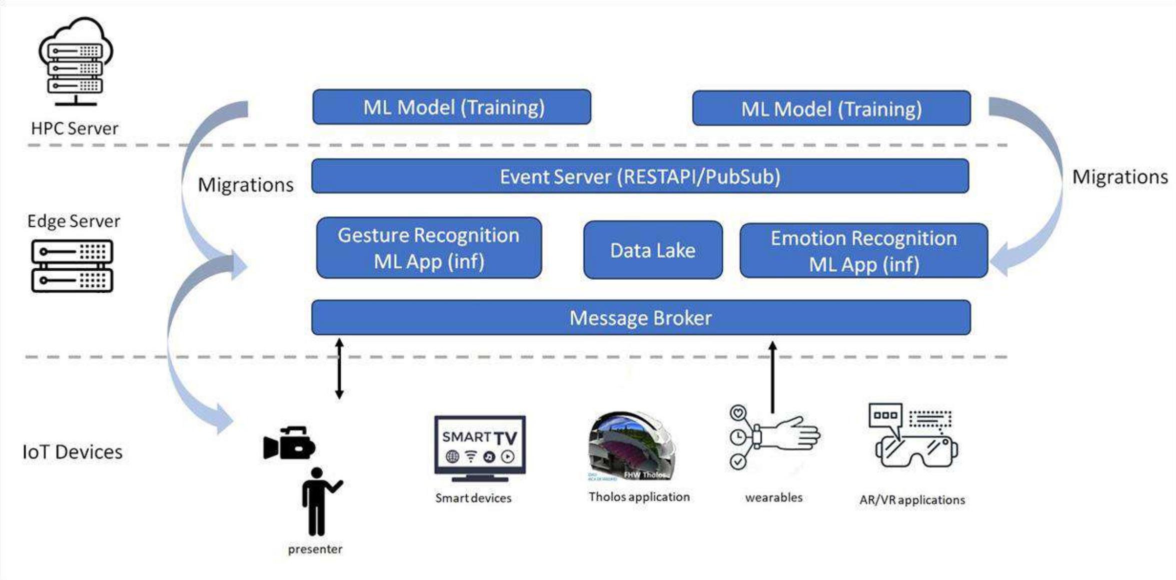


Figure 53 Latest version of the Pilot Architecture for XR01 and XR02 use cases

This section below details a comprehensive approach to detecting stress using data captured by wearable devices. It begins with an in-depth overview of the dataset (collected from smartwatches and a complementary public dataset) which includes crucial physiological and motion-based signals (such as heart rate, accelerometer data and skin temperature). The study outlines the rigorous exploratory data analysis and preprocessing steps that ensure data quality, address class imbalance, and convert raw sensor streams into a format suitable for time-series modeling. A hybrid deep learning model combining convolutional and LSTM layers is then trained using these preprocessed signals. Finally, the evaluation results) including high accuracy, precision, recall, and F1 scores (demonstrate the model's strong capability in real-time stress classification, highlighting the promise of wearable devices for continuous stress monitoring in both controlled and real-world environments.

Dataset Overview

A key application of Smart XR is the continuous monitoring of a user's stress levels while they wear a virtual reality headset. To achieve this, we selected the Samsung Galaxy S6 smartwatch as our primary data collection device due to its ability to capture crucial physiological and motion-based signals for stress detection, including heart rate (HR), accelerometer data (X, Y, Z), and body temperature (TEMP). These metrics enable real-time tracking of subtle physiological changes, providing data-driven insights into the user's emotional state during VR immersion.

To complement this smartwatch data, we utilized a publicly available dataset specifically designed for continuous stress monitoring among hospital nurses. This dataset closely aligns with the smartwatch's capabilities while also including electrodermal activity (EDA), a critical marker of sympathetic nervous system activation. The data was gathered using the Empatica E4 wearable device, which records EDA at 4 Hz and blood volume pulse (BVP) at 64 Hz, enabling the extraction of HR metrics. These high-resolution time series measurements provide precise tracking of physiological fluctuations throughout a nurse's shift.

Collected in a real-world, high-stress work environment, the dataset consists of approximately 11.5 million records spanning nine physiological and contextual features. These include orientation data (X, Y, Z), electrodermal activity (EDA), heart rate (HR), skin temperature (TEMP), a subject identifier (ID), timestamps (Datetime), and categorical stress state labels (low stress, medium stress and high stress). Table 14 provides an overview of randomly selected samples from the dataset.

X	Y	Z	EDA	HR	TEMP	id	Datetime	label
-13	-61	5	6.769995	99.43	31.17	15	03:00.0	2
-20	-69	-3	6.769995	99.43	31.17	15	03:00.0	2
12	60	20	0.087115	69.72	29.93	5C	48:06.6	1
11	59	20	0.087115	69.72	29.93	5C	48:06.6	1
-36	-4	55	0.292055	83.12	28.99	5C	57:19.8	0
-34	-4	54	0.292055	83.12	28.99	5C	57:19.8	0

Table 14 Overview of the Stress Monitoring Dataset

By combining high-frequency physiological signals, orientation information, and self-reported contextual data, the dataset provides a robust ground for investigating how stress manifests in biometric readings. It also underscores the viability of wearable devices for continuous stress monitoring in real-world environments, reinforcing the potential to translate such insights from hospital settings to broader applications, including VR-based scenarios and everyday consumer stress monitoring.

Exploratory Data Analysis

To ensure the dataset was suitable for stress monitoring using smartwatch sensor data, an exploratory data analysis (EDA) process was conducted. Given that the smartwatch does not provide electrodermal activity (EDA) data, this feature was removed from the dataset along with the id and datetime columns, which were not relevant for numerical analysis. The dataset was then checked for missing values, revealing that none were present, ensuring that data completeness was not a concern.

Following data cleaning, the distribution of stress levels in the label column was analyzed to assess class imbalance. The dataset contained three stress levels: low stress (0), medium stress (1), and high stress (2). A frequency analysis revealed a significant imbalance, with high-stress instances (label 2) being the most prevalent, followed by low-stress instances (label 0), and a considerably smaller proportion of medium-stress instances (label 1). This imbalance was further visualized using a bar chart, as shown in Figure 54, confirming that high-stress data points dominated the dataset.

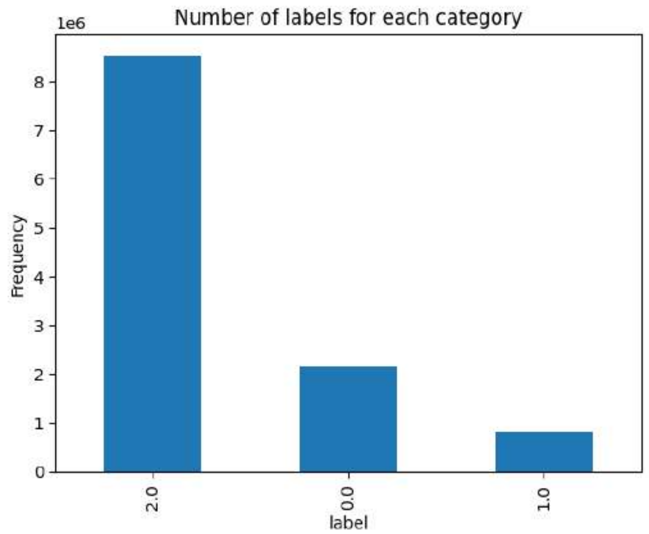


Figure 54 high-stress data points

To further investigate the characteristics of the dataset, as we see in Figure 55, histograms were plotted for each feature, revealing the underlying distributions of physiological and motion-related signals. The heart rate (HR) distribution was right-skewed, with most values concentrated between 60 and 100 beats per minute. Skin temperature (TEMP) exhibited a bimodal distribution, suggesting possible variations in external conditions or individual differences in physiological responses. The orientation features (X, Y, Z) showed near-normal distributions but with some skewness, potentially due to variations in movement intensity.

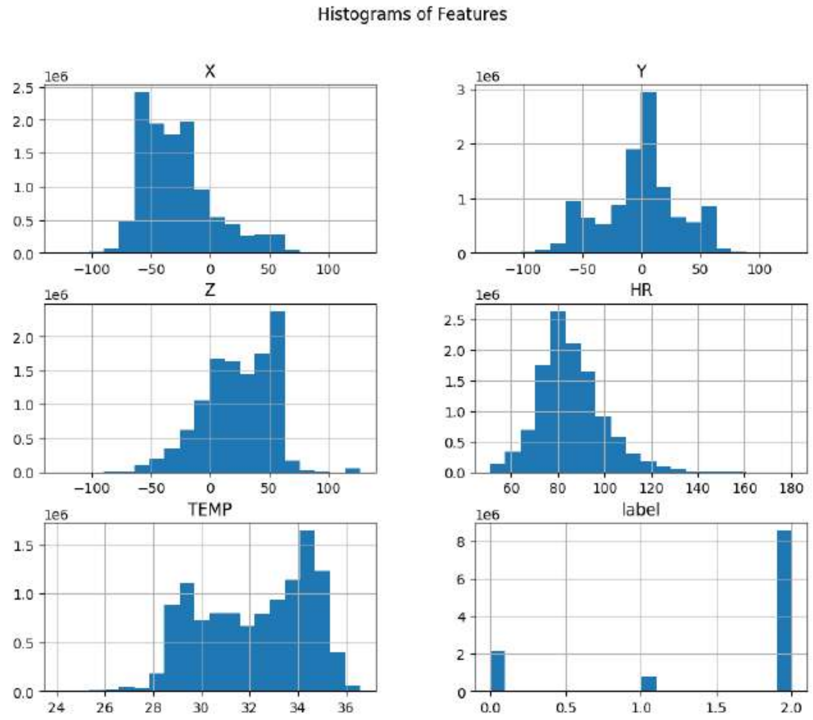


Figure 55 Histograms of Features

Overall, the EDA process ensured that the dataset was appropriately structured, cleaned, and preprocessed for stress monitoring applications. The removal of the EDA feature aligned the dataset with the available smartwatch sensors, while the distribution analysis provided insights into physiological variations. The observed class imbalance in stress levels suggested a potential need for data balancing techniques

Dataset preprocessing

Before training the model, several preprocessing steps were applied to ensure that the dataset was properly structured for effective stress classification. Given the nature of physiological signals, raw sensor data required transformation to align with the temporal and statistical properties needed for deep learning models. The preprocessing phase involved addressing class imbalance, standardizing numerical features, encoding categorical labels, and structuring the dataset into a time-series format suitable for sequence-based learning.

The dataset initially exhibited a significant class imbalance, with a disproportionately large number of high-stress instances compared to medium and low-stress instances. To mitigate the effects of this imbalance and ensure that the model did not become biased toward the dominant class, random undersampling was applied. This technique involved downsampling the majority class to match the number of instances in the minority class, ensuring an even class distribution across all stress levels.

Although undersampling reduces the total dataset size, it helps prevent model bias and enhances generalization across different stress states.

Once the dataset was balanced, feature scaling was applied to standardize the numerical attributes. All physiological and motion-related features—X, Y, Z (orientation data), HR(heart rate) and TEMP (skin temperature)—were transformed to a distribution with zero mean and unit variance. This standardization process was necessary to prevent features with larger numerical ranges from disproportionately influencing model learning. The labels were then one-hot encoded to represent the three stress levels as categorical variables, preparing the dataset for multi-class classification.

To effectively model the temporal dependencies in physiological signals, the dataset was further processed into a time-series format. A sliding window approach was applied, where sequences of 32 consecutive time steps were used as input samples, with the corresponding stress level at the final time step serving as the target label. This window size was not arbitrarily chosen; rather, it was selected because the smartwatch collects and transmits 32 measurements per second for each physiological feature. By aligning the window length with the smartwatch's data transmission rate, the model effectively learns from one-second intervals of physiological changes, ensuring that stress predictions are based on meaningful, real-time physiological patterns rather than isolated measurements. This transformation allowed the model to capture short-term fluctuations and trends in biometric signals while preserving the temporal structure of the data.

Model Training

To develop an accurate stress classification model, a deep learning approach was employed using a combination of convolutional and recurrent neural network architectures. A hybrid deep learning architecture combining convolutional (CNN) and recurrent (LSTM) layers was designed to capture both spatial and sequential patterns in the data. The model began with a 1D convolutional layer (Conv1D) to extract local feature representations, followed by a max-pooling layer to reduce dimensionality and enhance computational efficiency. Two long short-term memory (LSTM) layers were incorporated to capture long-term dependencies in the sequential data, with batch normalization and dropout layers added to prevent overfitting. The final layers consisted of fully connected (dense) layers, culminating in a softmax activation function for multi-class classification. The complete architecture of the proposed CNN-LSTM model is illustrated in Figure 56

The model was compiled using categorical cross-entropy as the loss function, Adam optimization, and accuracy as the primary performance metric. Training was conducted using early stopping to prevent overfitting by monitoring validation loss and halting training if performance began to deteriorate. The model was trained with a batch size of 32, a validation split of 20%.

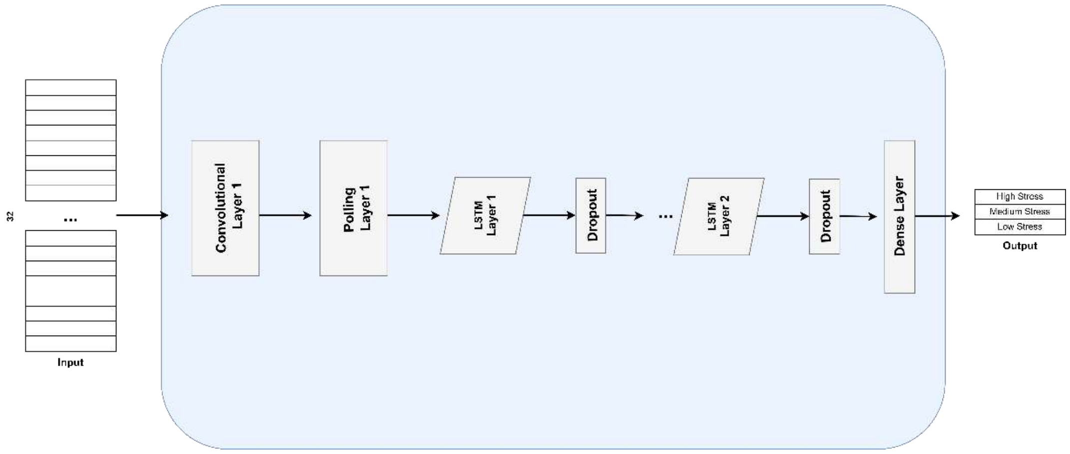


Figure 56 Architecture of the CNN-LSTM Model for Stress Classification

Model Evaluation Results

To assess the performance of the proposed model, the evaluation was conducted using a separate test dataset. The model was assessed using key classification metrics, including accuracy, precision, recall, and F1-score. Additionally, a confusion matrix was generated to visualize the classification performance across different stress levels.

The model achieved a test accuracy of 95.84%, indicating its effectiveness in distinguishing between different stress states. The precision, recall, and F1-score were 95.91%, 95.84%, and 95.84%, respectively, demonstrating a well-balanced performance across all classes. These results suggest that the model can reliably classify stress levels based on physiological and motion-based signals captured by the smartwatch. The complete evaluation metrics are summarized in Table 15.

Metric	Value
Precision	95.91%
Recall	95.91%
F1 Score	95.84%
TestLoss	0.11
TestAccuracy	95.84%

Table 15 Table Evaluation metrics

To further analyze the model's classification performance, Figure 57 presents the confusion matrix, illustrating the distribution of true and predicted labels. The majority of samples were correctly classified, with some misclassifications occurring primarily between adjacent stress levels. This suggests that while the model effectively distinguishes between stress categories, minor overlaps exist between medium and high-stress instances.

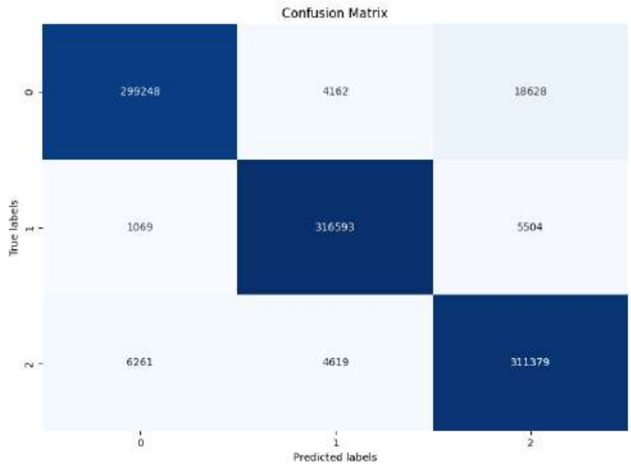


Figure 57 Confusion Matrix of the Stress Detection Model

Additionally, Figure 58 shows the training and validation accuracy over successive epochs, demonstrating the model's learning progression. The training curve indicates steady improvement in accuracy, while the validation curve remains closely aligned, suggesting minimal overfitting. The early stopping mechanism was triggered at epoch 7, confirming that additional training beyond this point would not have significantly improved performance. The final accuracy stabilized near 96%, reinforcing the model's generalization capability on unseen data.

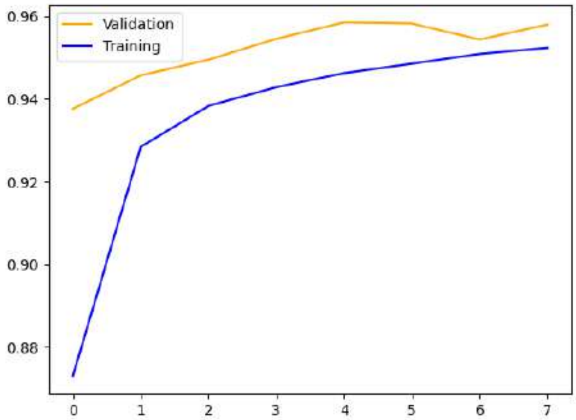


Figure 58 Training and Validation Accuracy Curves of the Stress Detection Model

The results confirm that the proposed CNN-LSTM architecture effectively captures temporal patterns in physiological signals, leading to accurate stress classification. The use of early stopping ensured optimal training, preventing overfitting while maintaining high generalization performance.

Adaptive Model Retraining

To enhance the accuracy of our stress detection model over time, we implement a data collection pipeline that stores real-time physiological and motion data from the smartwatch into a centralized data lake. This data, initially unlabeled, is preserved for future model retraining. To address the absence of labels, we incorporate the original labeled dataset into the data lake and leverage semi-supervised learning techniques to iteratively refine the model. Specifically, methods such as self-training, pseudo-labeling, and consistency regularization (e.g., Mean Teacher, FixMatch, and MixMatch) can be employed to infer labels for the newly collected data. This approach enables the model to progressively adapt to unseen variations in physiological responses, leading to improved generalization and robustness in stress detection.

Furthermore, the same methodology can be applied to a gesture recognition model, where motion sensor data from the smartwatch is continuously stored and later refined using semi-supervised learning techniques to improve the accuracy of gesture classification. By utilizing this iterative learning framework, both models benefit from an adaptive learning process, ensuring enhanced performance in real-world applications.

5.3.2 Technical validation

The scenario will validate the NEMO meta-OS with Meta Head Mounted Display (HMD) VR devices. The immersive VR experience inside the headsets will be enhanced using emotion detection AI/ML functionalities. Thus, the NEMO ecosystem will be 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. The KPI_XR_01.06 specified in D1.1 [1], D5.2 [5] targeting multipath connectivity using 360 degree video can't be verified in this trial as there is no such functionality, it will be deprecated from this trial and evaluated in the Smart Media City sibling trial mentioned in the previous section.

Scenario: XR01_Test_Scenario_1	
Scenario ID	XR01_Test_Scenario_1
Objective	VR Experience about ancient workshop of sculptor Phidias. Enhance experience with smart meter biometric data (Accelerometer, Heart Rate, Skin temperature).
Description	The XR application will be based on heterogeneous IoT devices (Wearables and VR headsets) and is going to collect and analyze anonymous biometric data from the user in order to estimate their emotional status during the VR experience in order to adapt the experience to the user. The specific use case is going to use state-of-the-art machine learning algorithms that are going to be trained and executed in the IoT-to-Edge-to-Cloud continuum.
Features to be 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 should be installed and configured, including at least 2 clusters, 10020wearable device providing biometric measures, 1 VR headset, 1 edge PC device for info presentation, 1 edges server. The ML application for emotional detection should run on the edge server. At least 1 user is registered as NEMO consumer and has access to workload LCM information, as well as at least 1 user registered as meta-OS Provider.

Scenario: XR01_Test_Scenario_1	
Test steps	• The Smart XR application owner signs in NEMO as meta-OS consumer. • The application owner uploads the workload descriptor on the NEMO platform. • The application owner deploys the application over the continuum via the NEMO API. • The XR application is started and test scenario 1 is performed. • The application owner monitors the workload execution and confirms that data from the wearables are sent, analyzed, and consumed properly • The application owner retrains the ML model (this process requires the ML micro-service to be migrate to High Performance Computer (HPC) infrastructure). • The application owner terminates the execution and collects the logs.
Success state	• Application running in the VR headset consumes successfully events by ML emotional detection service. • Application running in VR headset stops on detection of user's emotional event. • Successfully training of the ML models in central/edge HPC nodes.
Failure state	AI-ML application fails to deploy or to detect emotions, VR app and IoT devices do not receive status changes or do not adapt correctly.
Responsible for testing and implementation	MAG, FHW
Risks	No risks identified so far.

Table 16 Test scenario 1 for the ancient workshop use cases

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_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 Gesture and Emotion components with Unity 3D Apps.
KPI_XR_01.3	Conditional tasks/Micro-service migration. Success rate. - 100%. ML training 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	Description	How it will be tested
KPI_XR_01.5	Does the user get better experience >=25%	Qualitative user and administrator study to prove better QoE.

Table 17 Test XR01 scenario 1 KPI validation methodology

5.3.3 Use case sequence diagrams

The use case diagrams presented in the previous deliverable D5.2 [5]provide a detailed description of how the several components interact with each other. These diagrams are still valid and are to be updated in deliverable D5.4 with the finalization of the piloting use case as new interactions may be needed as the project progresses.

5.3.4 Timeline of activities

Phase	Start Date	End Date	Notes
Delivery of beta version of components. Local Testing.	M5	M27	Completed
Testing Deployment via NMEO meta-OS	M27	M29	Completed
Pre-Trial at Site test at the Hellenic Cosmos Cultural Center	M30	M30	Completed
Final Integration with NEMO	M30	M32	In Progress
Final Pilot Tests	M32	M33	To be performed

Table 18 Timeline of activities for XR01

5.3.5 Intermediate Results

At the end of M30 the integration status with NEMO is as outlined below:

- The Helm chart has been created and tested in our local environment (Done).
- Integrating testing on the OnLAB cluster (in progress).
- Migration testing to a high-Performance Server using the Meta-Orchestrator (MO) (in progress).
- End-to-end testing (using the intent-based API and MO) (in progress).
- On site Pre-Trial tests at the Hellenic Cosmos Cultural center in a local environment setup (Done).

An important part of the XR Pilot is the qualitative evaluation in order to fulfill also KPI_XR_01.5. Since the XR pilot target directly the usage of NEMO in a cultural setting, an evaluation will proof that besides the technical markers also the experience and operation of the VR experience has benefited from NEMO affordances. Two questionnaires will be designed, one for HMD operators (museum educators) and one for visitors. An information sheet and consent form is prepared in which the participant agrees to participate in the evaluation and have his data collected. Consent for media capture for media posts is also asked as well and the visitor is informed that there is a possibility of motion sickness. The questionnaires will only have participant codes (no id, no names). The questions will be multiple choice and open ended for collection of qualitative data and a post-use interview with the operators will also be performed.

During M30 a complete set of Pre-Trial tests were performed at the Hellenic Cosmos Culture center of the Foundation of the Hellenic World. The tests were performed by FHW and MAG and were carried out with a local environment setup with components mainly being located on the Edge.

The integration with NEMO, the migration, intent based API, MOCA and MO components were not tested. During the trial the Event Server RestAPI, the Emotion Recognition ML App, the Data Lake were run on a server located at the edge. The Message Broker was located at the cloud on MAG servers. The smart wearable (Samsung Galaxy 5) was worn by a test user during the VR experience as shown in Figure 60. The VR experience running on Pico G4 Head Mounted Displays as shown in Figure 59 was running a Unity3D application presenting an interactive storytelling application about a visit at the ancient workshop of the ancient artist Phidias.

Both the VR app and an external PC device were subscribed to the event server for receiving messages. The external smart device PC was connected to a smart TV for monitoring the visitor and was executing a Unity3D application which informed about the emotional status of the user. Upon receiving a high stress indicator, the device prompted to the museum educator that the visitor needed attention. Similarly on receiving a high stress indicator the VR application executing on the headset displayed a message to the user that high stress was sensed and a member of staff will come shortly. The IoT smart watch connected to the user's hand was running a dedicated application that communicated with the message broker transmitting all the needed bio-data during the interaction of the user with the VR application. Note that during the pre-trial we used a Pico headsets but for the final pilot trial a Meta headsets will be used to use the latest version of the VR app.

Figure 59 The user wearing a Head Mounted Display device during the VR experience

Figure 60 The Smart Wearble IoT device

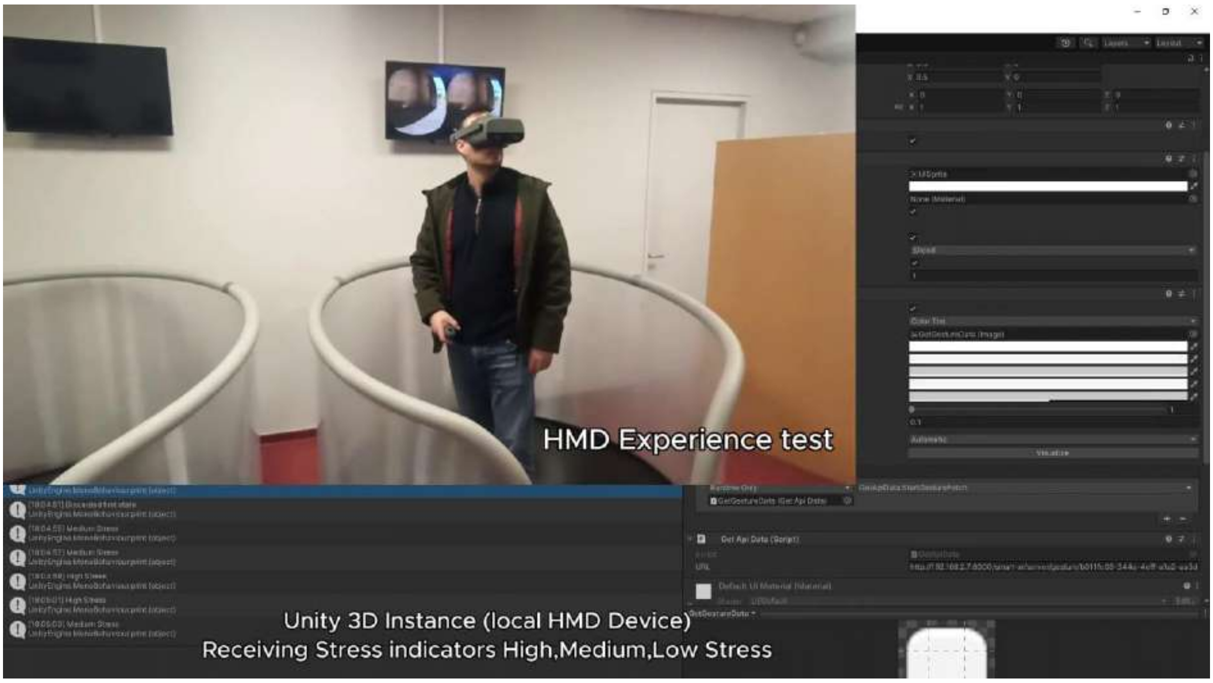


Figure 61 Unity VR application Debug Messages for receiving the emotion estimation

5.4 XR_02 Enhance AV experience in the Tholos Dome VR Theatre

5.4.1 Trial Site Description and Updates

The XR use cases focus on enhancing educational VR experiences. The pilot will be hosted at the Hellenic Cosmos Cultural Center of the Foundation of the Hellenic World, which is based in Athens and features exhibitions, educational programs and public access to projection based VR Systems for large audiences.

The second experience that will be enhanced will be for a much larger VR system. A large scale realtime VR Dome where live guided tours are provided for 132 visitors through interactive environments. For the VR Dome the video of the museum educator who is doing the live Dome show presentation is captured during the show, using a video capture PC. This video is sent to AI/ML nodes for analysis on gestures. The presenter thus can trigger events by performing gestures. These events could trigger actions in the real-time show that is presented or notify external staff for assistance or display info on external screens such as when the show is about to end. Thus, it provides another interaction capability.

For the implementation of this component, the MediaPipe library was utilized. MediaPipe, developed by Google, is an open-source framework designed for real-time perception tasks, including hand tracking, pose estimation, and object detection. It provides efficient, cross-platform solutions optimized for mobile and web applications. Specifically, for gesture recognition, MediaPipe's Hand Tracking module offers robust hand landmark detection and tracking capabilities, allowing for precise gesture classification in real time. The framework employs machine learning models to detect hand key points with high accuracy, making it a suitable choice for gesture-based interaction systems.

Specifically, the first step in the gesture recognition pipeline is detecting whether a hand is present in the input frame. By using a machine learning-based detector that identifies the bounding box of the hand in the image or video feed. This detection is optimized for real-time performance, allowing the system to process multiple frames per second without significant latency. Once a hand is detected, MediaPipe applies a pose estimation model to extract 21 key landmarks on the hand and with the extracted hand landmarks, the next step is the gesture classification.

Dataset Creation

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.
The selected gestures for this dataset are:

- ThumbsUp
- ThumbsDown
- Victory
- UnknownClass

The Unknown Class plays a critical role in improving the model's reliability. In real-world scenarios, the system might encounter hand positions or gestures that it has not been explicitly trained on. Without an Unknown class, the model might mistakenly classify such inputs as one of the known gestures, leading to errors. To mitigate this, we included a set of random, non-relevant hand positions in the dataset, allowing the model to correctly identify unfamiliar gestures as "Unknown."

To ensure robustness, we captured gesture images from multiple angles and using both hands (left and right). This was done to improve the model's ability to generalize across different hand orientations and perspectives. If images were captured only from a single angle or a single hand, the model might struggle to recognize gestures performed differently from the training data. Figure 62 presents a set of example images from the dataset, demonstrating the variation in angles, hand positioning, and lighting conditions.

(a) Thumpsup

(b) Victory

(c) Thumpsdown (d) Unknown

Figure 62 Images of the gesture recognition dataset

The dataset was carefully constructed to ensure a balanced representation of each class. Specifically, 300 samples were collected for the Thumbs Up gesture, 300 samples for the Thumbs Down gesture, and 300 samples for the Victory Sign. Additionally, 350 samples were included for the Unknown class to account for hand positions and gestures that do not belong to the predefined categories. This distribution ensures that the model is not biased toward any specific class while still providing additional examples for the Unknown category to improve generalization.

Once the images were collected, they were automatically processed to extract relevant hand keypoints. Instead of using raw images, we converted each image into a numerical representation by extracting the (x, y) coordinates of key hand landmarks. This was done using the MediaPipe framework, which provides real-time hand tracking and gesture analysis. Each image was converted into a structured format suitable for machine learning training. The dataset structure follows this format, first column is the gesture label, and second column onward is normalized keypoint coordinates (x, y positions of hand landmarks).

Each row in the dataset represents a single gesture sample, with the first value indicating the gesture class and the remaining values representing the normalized coordinates of key hand landmarks. By using this structured dataset, the model can efficiently learn the spatial distribution of hand movements, leading to improved recognition accuracy across different real-world conditions. Before training, the dataset was split into training, validation, and test sets to ensure proper model evaluation and generalization. Each gesture class contained a total of 300 samples, out of which 240 samples were allocated for training and validation, while 60 samples were reserved exclusively for testing.

Model training

To train the gesture recognition model, we employed a feedforward neural network. The model was designed to take hand keypoint coordinates as input and classify them into one of the predefined gesture categories. The training process involved multiple steps, including model architecture definition, compilation, and training with early stopping and checkpointing mechanisms.

The model follows a sequential architecture composed of fully connected (dense) layers. It takes a 42-dimensional input, representing the (x, y) coordinates of 21 hand landmarks. To prevent overfitting, a dropout layer with a 20% dropout rate is applied, randomly deactivating neurons during training. The first hidden layer consists of 20 neurons with ReLU activation, allowing the model to learn complex patterns in the data. Another dropout layer with a 40% dropout rate follows to further improve generalization. The second hidden layer, with 10 neurons and ReLU activation, refines the learned features. Finally, the output layer applies a softmax activation function, producing class probabilities for the predefined gesture categories. This ensures the model assigns a confidence score to each class, making it suitable for multi-class classification.

The model was trained using the training dataset employing a maximum of 500 epochs to allow sufficient learning while mitigating underfitting. A batch size of 128 was used to balance computational efficiency and model generalization. To monitor the model's performance and prevent overfitting, a separate validation dataset was utilized throughout training. Additionally, early stopping was applied with a patience value of 20 epochs, allowing the training process to halt automatically if validation loss did not improve over 20 consecutive epochs. This approach prevented unnecessary computations and reduced the risk of overfitting.

Evaluation Results

The training process yielded strong results, with the model achieving a training accuracy of 91.62% and a validation accuracy of 99.14%. Early stopping was triggered at epoch 100, indicating that the model had reached optimal performance before the maximum epoch count. Final evaluation on the test dataset confirmed the model's effectiveness, achieving a test accuracy of 98.43%, demonstrating its robustness in recognizing the predefined gesture classes. To further assess the model's performance, a confusion matrix was generated, providing insight into its classification accuracy for each gesture category. The confusion matrix, shown in Figure 63, illustrates the number of correct and misclassified predictions for each class.

To evaluate the real-time performance of the gesture recognition system, a live test was conducted using a webcam, as shown in Figure 64. The system continuously captures frames from the camera, detects hand landmarks, and processes the extracted key points through the trained neural network to classify gestures in real time. In this instance, the model successfully identifies the all the gestures, displaying both the detected hand landmarks and the classification result on the interface. Additionally, the terminal output logs show the detected gestures over time, confirming the accuracy and responsiveness of the model. These results demonstrate the model's effectiveness in live conditions, making it suitable for interactive applications.

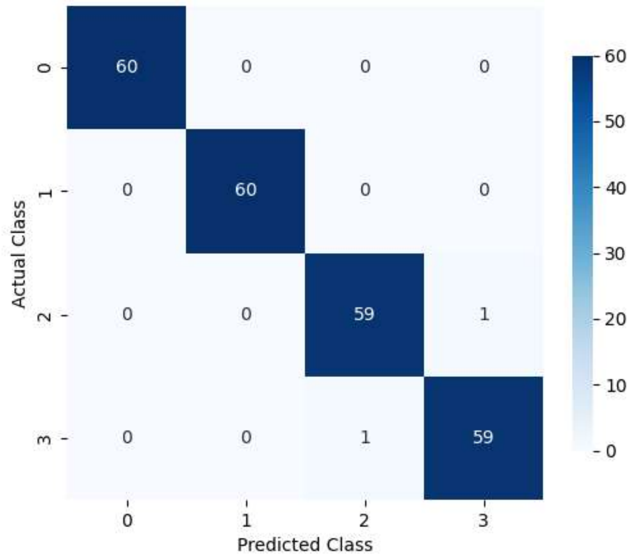
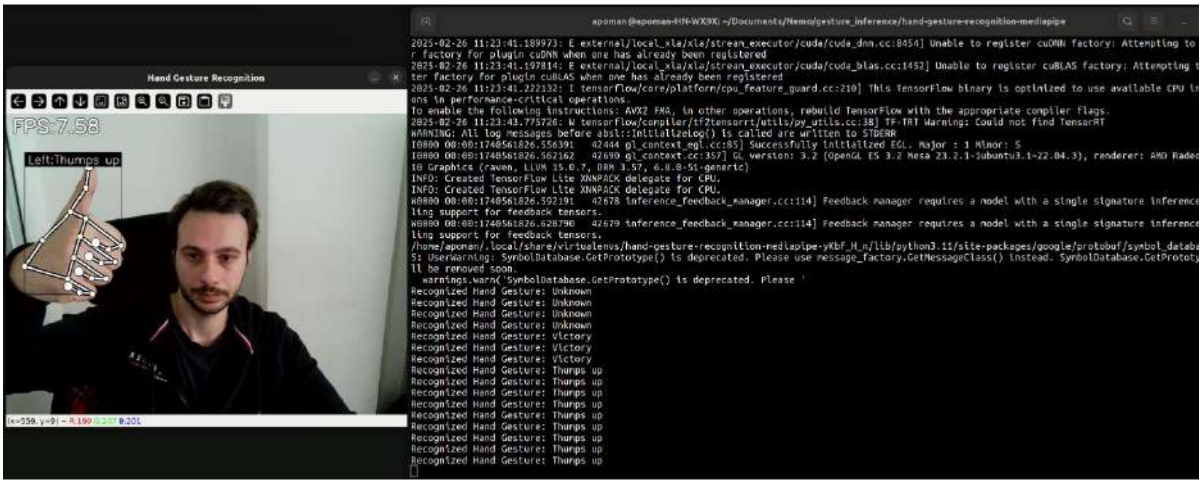
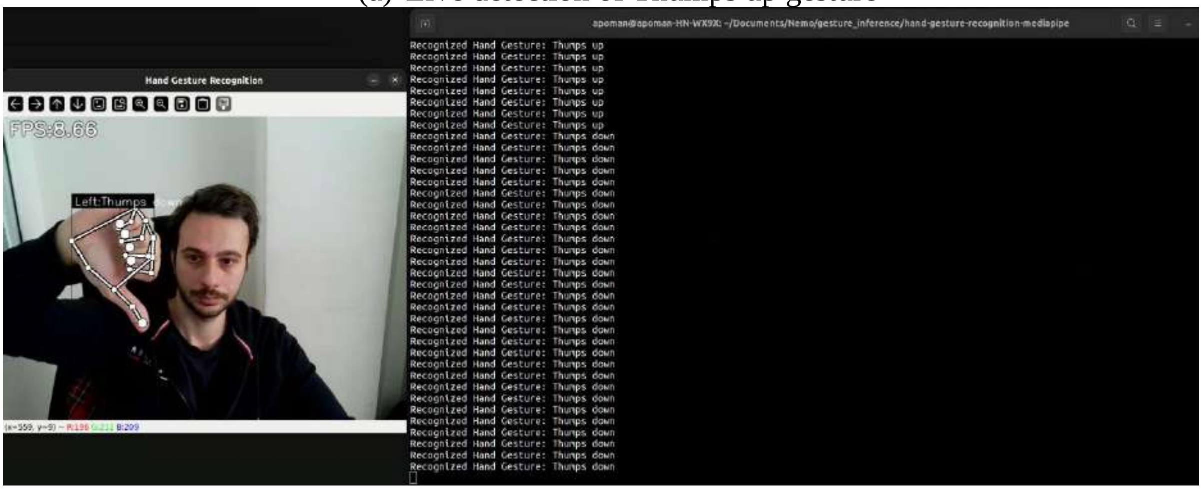


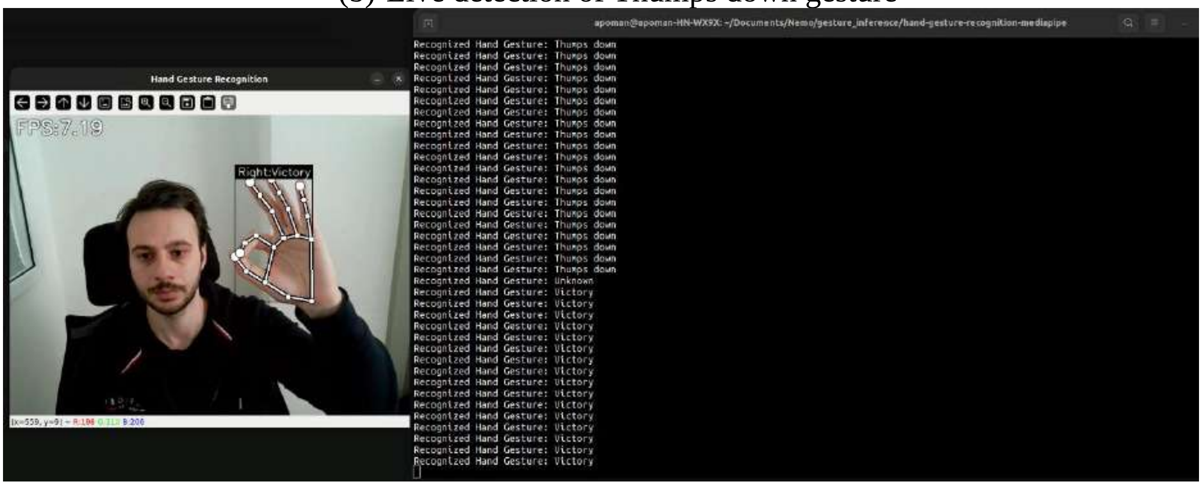
Figure 63 Confusion Matrix of the Gesture Recognition Model



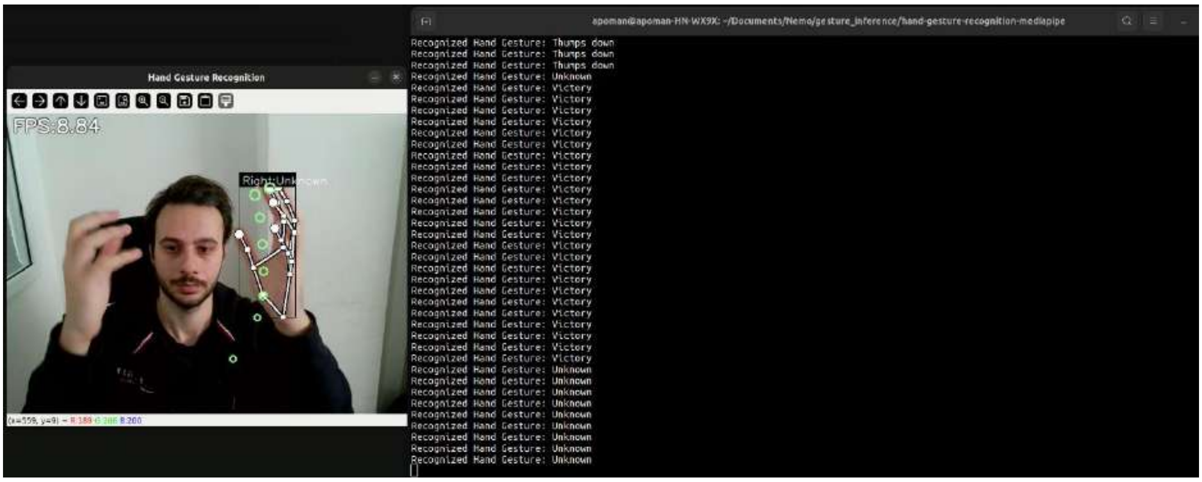
(a) Live detection of Thumps up gesture



(b) Live detection of Thumps down gesture



(c) Live detection of Victory gesture



(d) Live detection of Unknown gesture

Figure 64 Real-TimeGestureRecognition

5.4.2 Technical validation

The scenario will validate the NEMO meta-OS within a large-scale Digital Dome VR Theatre. The immersive VR experience running on the Dome will be enhanced using gesture detection AI/ML functionalities. Thus, the NEMO ecosystem will be validated 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. This pilot will be only based on gesture recognition and will 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 [5] has been removed as deprecated.

Scenario: XR02_Test_Scenario_1	
Scenario ID	XR02_Test_Scenario_1
Objective	Enhance experience in the Tholos Dome VR Theatre. Analyze gestures of museum-educator presenter. Gesture based recognition by using ML in IoT-to-Edge-to-Cloud continuum.
Description	Enhance the user's audio-visual (AV) experience in the Tholos Dome VR Theatre. This scenario will analyze the physical position of the presenter and perform gesture recognition based on state-of-the-art machine learning algorithms. The system trains and executes ML models in the IoT-to-Edge-to-Cloud continuum and it will trigger events in real time that are going to be 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.
Features to be 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,

Scenario: XR02_Test_Scenario_1	
	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 should be installed and configured, including at least 2 clusters, 1 wearable device providing biometric measures, 1 Dome Theatre driven by a realtime-rendering PC, 1 edge PC device for info presentation, 1 edges server who also handles video capturing. The ML application for gesture detection should run on the edge server. At least 1 user is registered as NEMO consumer and has access to workload LCM information, as well as at least 1 user registered as meta-OS Provider.
Test steps	• The Smart XR application owner signs in NEMO as meta-OS consumer. • The application owner uploads the workload descriptor on the NEMO platform. • The application owner deploys the application over the continuum via the NEMO API. • The DOME VR application is started and test scenario 1 is performed. • The application owner monitors the workload execution and confirms that image data from the camera are sent, analyzed, and consumed properly • The application owner retrains the ML model (this process requires the ML micro-service to be migrate to High Performance Computer (HPC) infrastructure). • The application owner terminates the execution and collects the logs.
Success state	• VR Dome application consumes successfully events by ML gesture recognition service. • VR Dome application reacts properly to the presenter signals. • Subscribed IoT device such a smart TV show info.
Failure state	AI-ML app fails to deploy or to detect gesture, external IoT devices do not receive status changes.
Responsible for testing and implementation	MAG, FHW
Risks	No risks identified so far.

Table 19 Test scenario 1 for the use case in Tholos Dome VR Theatre

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	Description	How it will be tested
KPI_XR_02.4	End to end low latency migration to avoid dizziness and motion-sickness < 20ms. Latency between user gesture and recognition. Measure latency. 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.

Table 20 Test XR02 scenario 1 KPI validation methodology

5.4.3 Use case sequence diagrams

The use case diagrams presented in the previous deliverable D5.2 [5]provide a detailed description of how the several components interact with each other. These diagrams are still valid and are to be updated in deliverable D5.4 with the finalization of the piloting use case as new interactions may be needed as the project progresses.

5.4.4 Timeline of activities

Phase	Start Date	End Date	Notes
Delivery of beta version of components. Local Testing.	M5	M27	Completed
Testing Deployment via NMEO meta-OS	M27	M29	Completed
Pre-Trial at Site test at the Hellenic Cosmos Cultural Center	M30	M30	Completed
Final Integration with NEMO	D30	D32	In Progress
Final Pilot Tests	D32	D33	To be performed

Table 21 Timeline of activities for XR02

5.4.5 Intermediate Results

At the end of M30 the integration status with NEMO is as follows:

- The Helm chart has been created and tested in our local environment (Done).
- Integrating testing on the OneLAB cluster (in progress).
- Migration testing to a high Performance Server using the Meta-Orchestrator (MO) (in progress).
- End-to-end testing (using the intent-based API and MO) (in progress).

- On site Pre-Trial tests at the Hellenic Cosmos Cultural center in a local environment setup (Done).

An important part of the XR Pilot is the qualitative evaluation in order to fulfill also KPI_XR_02.5. Since the XR pilots target directly the usage of NEMO in a cultural setting, an evaluation will proof that besides the technical markers also the experience and operation of the VR theatre has benefited from NEMO affordances. One questionnaire will be designed for the DOME operators (museum educators) that present the show. An information sheet and consent form is prepared in which the participant agrees to participate in the evaluation and have his data collected. Consent for media capture for media posts is also asked. The questionnaire will only have participant code (no id, no names). The questions will be multiple choice and open ended for collection of qualitative data and a post-use interview will also be performed.

During M30 a complete set of Pre-Trial tests were performed at the Hellenic Cosmos Culture center of the Foundation of the Hellenic World. The tests were performed by FHW and MAG and were carried out with a local environment setup with components and devices mainly located on the edge.

The integration with NEMO, the migration, the intent based API, MOCA and MO components were not tested. During the trial the Event Server RestAPI, the Emotion Recognition ML App, the Data Lake were run on a server located at the edge. The Message Broker was located at the cloud on MAG servers. A PC running a capture python app and an usb video camera was used to record the gestures. The Dome was powered by a Super Graphics Rendering PC running a 3D Unity application presenting the architectural walkthrough of an ancient 10th century monastery in Greece. An external PC was used to connect to a smart TV for displaying the state of gestures.

Both the rendering PC as well as the PC connected to the smart TV were subscribed to the event server in order to get the gesture recognitions. The pc device connected to the camera, would capture the video and send a stream of images to the Message Broker for subsequent ML based gesture recognition. During the trial the navigation of the Dome experience was performed using gestures. Gestures were also received by a PC located outside of the Dome displaying the results to a smart TV. In the final trial the PC located outside of the Dome will interpret the gestures and will be used for communicating the various states of the show. For an example a victory gesture would indicate an ongoing show, and a thumbs up that the show will end in 5 min.

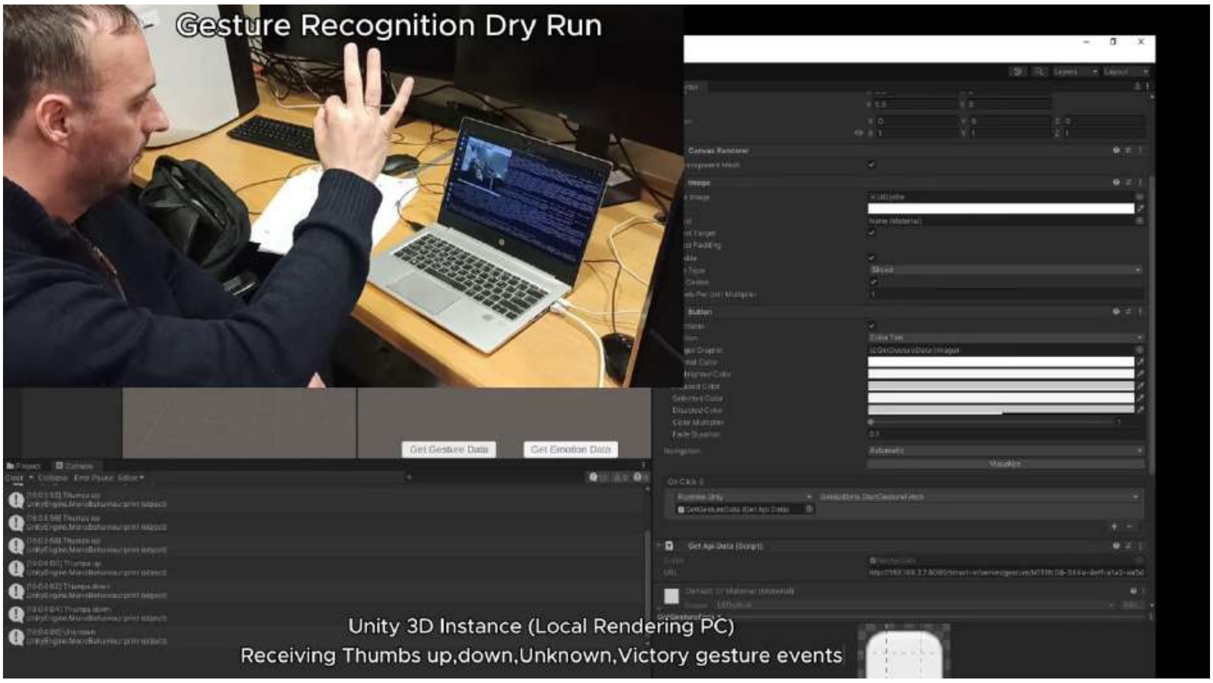


Figure 65 Gesture recognition in front of the video camera PC.
Debug log messages on the unity 3D application which executes on the smart TV PC.

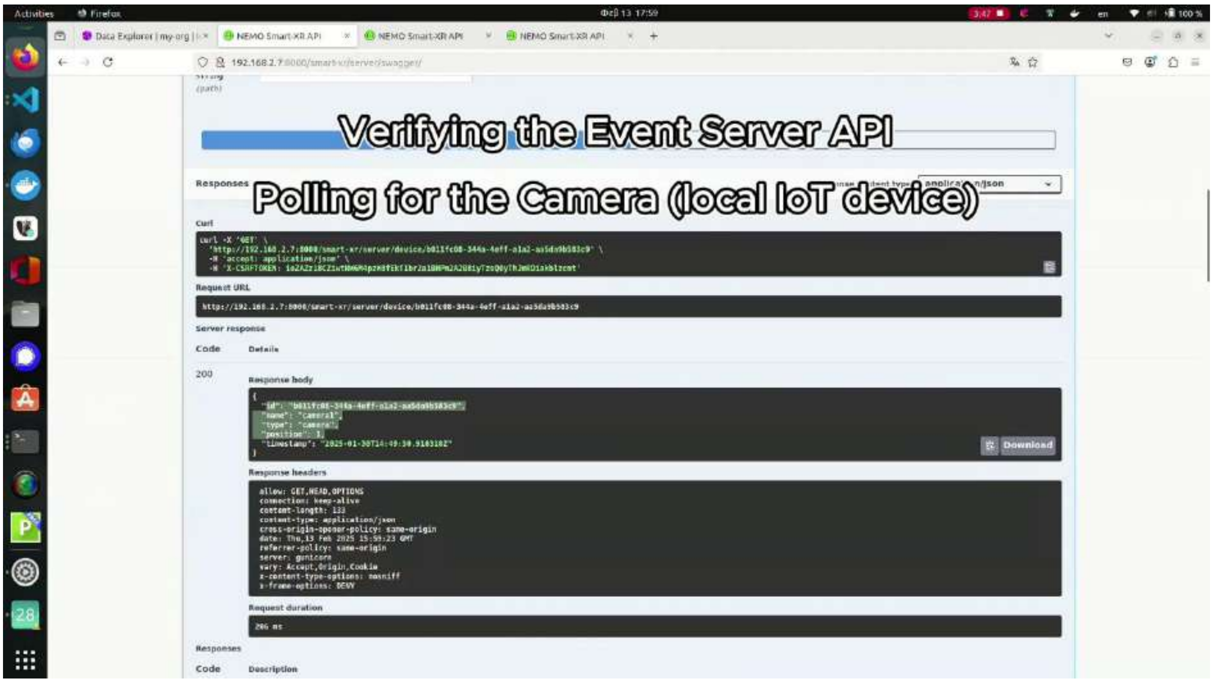


Figure 66 The Rest API exposing the Camera IoT device for gesture reception.

Figure 67 Pre-Trial inside the Dome. Navigating the 3d experience with gestures.

6 Data Management Plan Updates

The NEMO project follows a structured and GDPR-compliant data management strategy, ensuring that all collected datasets are properly stored, accessed, and utilized while adhering to privacy, security, and legal requirements. As the project progresses, continuous updates to the Data Management Plan (DMP) are necessary to align with evolving data collection needs, regulatory compliance (e.g., GDPR), and best practices in research data management. Proper data governance facilitates interoperability, collaboration, and long-term sustainability of the project's outcomes, while also safeguarding sensitive and proprietary information.

To assess and incorporate necessary updates, a structured Ethics and Data Management questionnaire was circulated among the project partners, collecting insights into data handling practices, potential modifications in data collection, and compliance considerations. Table 22 provides an overview of the latest updates on the DMP, reflecting the adjustments reported by the NEMO trials as of M30.

Dataset	Description	Scope	Data Availability
Aerial olive tree image dataset	This dataset contains aerial images of olive groves, including images of trees (aerial leaf images) and control images (random images taken over an olive grove, but do not contain olive trees). It is being used for image classification to assist aerial precision spraying for the Smart Farming LL trials.	WP5/T5.2	Public
Terrestrial olive tree image dataset	This dataset contains terrestrial images taken using the AGVL from the piloting site. These images were used for the training of the LL model. The dataset is enhanced by the real-time video snippets taken during the validation phase at the farm.	WP5/T5.2	Confidential
Power Quality Analyser data	This dataset consists of real-time measurements collected from Power Quality Analyzers (PQAs) installed in the MV/LV substations of Terni electrical distribution distribution network. It includes parameters such as voltage, current, power, and environmental conditions, recorded and processed locally by the devices.	WP5/T5.3	Confidential
Historical Consumption and Production data (Wally)	Time-series data of energy consumption loads. Supports demand-response strategies and RES integration.	WP5/T5.3	Confidential
Phasor Measurement Unit (PMU) data	This dataset consists of real-time measurements and high-resolution synchrophasor data from PMUs, including voltage, current, frequency, phase angle. The measurement is related to a specific portion of the electrical distribution grid and it is used for	WP5/T5.3	Confidential

Dataset	Description	Scope	Data Availability
	real-time grid monitoring, fault detection, and disturbance classification.		
Historical and Real-time Charging station data	Time-series data of charging stations. Supports demand-response strategies and RES integration.	WP5/T5.3	Confidential
Historical and Real-time electric vehicle data	Time-series data of electric vehicles. Supports demand-response strategies and RES integration.	WP5/T5.3	Confidential
Video Quality Probes	This dataset contains Quality of Experience (QoE) measurements and additional image KPIs (Key Performance Indicators) related to media streams. It includes metrics such as resolution, bitrate, frame loss, latency, and image artifacts.	WP5/T5.5	Confidential
AI Engine Runner Recognition	This dataset consists of AI-based analysis of race streaming footage, specifically recognizing runners and extracting their bib numbers using computer vision techniques. The dataset includes labeled images, bounding box coordinates, and extracted bib numbers.	WP5/T5.5	Confidential
Smart Watch Data	This dataset will have data originating from, orientation data Accelerometer (AC), Heart Rate (HR), skin temperature (TEMP) from the smart watch.	WP5/T5.5	Confidential
Camera Video Data	This dataset will have data originating from a camera video stream. To ensure privacy and compliance with data protection regulations, we do not retain any images containing faces when training our gesture recognition model. Instead, we extract key landmark points from each image and store them in a structured CSV file.	WP5/T5.5	Confidential

Table 22 Data management plan updates

7 Conclusions

Deliverable D5.3, reported an intermediate evaluation of the NEMO Living Labs, summarizing the progress made across the five verticals—Smart Farming, Smart Energy & Mobility, Smart Industry, and Smart Media & XR. It provides an overview of the technical developments, application implementations, and validation efforts conducted so far. Each trial has advanced in developing all the core components of Use case specific applications and services, which are now being integrated into the NEMO meta-OS to demonstrate its capabilities in managing distributed workloads across IoT, edge, and cloud environments. In addition, the deliverable presented an updated version of the Data Management Plan (DMP), incorporating the latest feedback from consortium partners.

In the final deliverable, D5.4 "NEMO Living Labs Use Cases Evaluation Results – Final Version", due in Month 36, the complete outcomes of the pilots will be presented. This will include consolidated validation results, performance benchmarks, impact assessments, and insights into NEMO's role in enabling distributed computing and resource orchestration.

At this stage, no significant modifications to the DMP are anticipated. However, any emerging requirements or refinements identified during the final validation phase of the project will be addressed in the concluding deliverable of Work Package 5 (D5.4), ensuring that the NEMO data management strategy remains robust and aligned with the overall project objectives.

8 References

[1] NEMO, "D1.1 - Definition and analysis of use cases and GDPR compliance," HORIZON - 101070118 - NEMO Deliverable Report, 2023.
D1.1 Definition and analysis of use cases and GDPR compliance

[2] NEMO, "D1.2 - NEMO meta-architecture, components and benchmarking. Initial version," HORIZON - 101070118 - NEMO Deliverable Report, 2023.
D1.2 NEMO meta-architecture, components and benchmarking. Initial version

[3] NEMO, "D1.3 - NEMO meta-architecture, components and benchmarking. Final version," HORIZON - 101070118 - NEMO Deliverable Report, 2024.
NEMO_D1.3-NEMO-meta-architecture-components-andbenchmarking.Final-version_V1.1.pdf

[4] NEMO, "D5.1 - Living Labs and Data Management Plan (DMP). Initial version," HORIZON - 101070118 - NEMO Deliverable Report, 2023.
D5.1 Living Labs and data management plan (DMP). Initial version

[5] 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

Annex. Data Summary and FAIR Data Considerations

To maintain clarity and provide structured documentation, the detailed information of datasets from all the pilots has been compiled in this section based on the data collection templates provided to the pilot owners as shown in. This annex presents a detailed inventory of datasets used within each trial, following a standardized structure that includes dataset descriptions, ownership and licensing information, storage mechanisms, access conditions, and metadata considerations. Additionally, the FAIR (Findability, Accessibility, Interoperability, and Reusability) principles have been taken into account to ensure proper data governance, facilitating future reuse and integration.

Dataset Name (The name that clearly identifies the dataset)	**URL to the data or published descriptor (doi, zenodo)**		
Description (A brief summary describing the data content) **Purpose of Data Collection:** (A short introduction text explaining the purpose of the data collection and the relation to the objectives of the UC) **Data Origin: Data Origin** should be one of the following: • **Observational:** Data captured in real time, often not reproducible (e.g. sensor readings, images, telemetries, sample data). • **Experimental:** Data from lab equipment, often reproducible, but with high costs (e.g. chromatograms, magnetic fields readings). • **Simulation:** Data generated by computational models (e.g. climate models, economic models, materials models). • **Derived / Compiled:** Data coming from analysis or compilation. Reproducible but with high costs (e.g. results of text and data mining, compiled databases). • **Reference / Canonical:** Collection or conglomeration of smaller datasets published and curated (e.g. chemical	**Ownership & Licensing** (Information about who owns the data and any intellectual property licenses or agreements)	**Reuse Conditions** Specific rules about how the data can be reused (e.g., open for reuse, restricted to project partners)	**Data Availability** (Public/Confidential) There may be several reasons for not sharing them: ethical reasons, protection of personal data, the involvement of intellectual and/or industrial property rights, commercial interests, etc. You must specify the reasons why a dataset will not be shared.
	Storage Details (Information about the storage medium (e.g., cloud server, on-premise server, database))	**Access Mechanisms** How the dataset can be accessed, including methods like APIs, message brokers, SCADA connections, etc.	**Metadata information** Description of data format and explain how the data and metadata are assigned to a globally unique and eternally persistent identifier. This identifier could be a DOI (Digital

Dataset Name (The name that clearly identifies the dataset)	URL to the data or published descriptor (doi, zenodo)		
structures, gene sequence databanks, spatial data portals). **Data size: (**Provide the information on the approximate volume of the datasets. Consider the implications of data volumes in terms of storage, backup, cost and access. Estimate the volume of data in MB/GB/TB and how this will grow to make sure any additional storage and technical support required can be provided)			**Object Identifier)**, **Handle**, or a unique **URL** from an institutional repository.

Table 23 Data Collection Template Shared with Pilot Leaders

For publicly available data that can be shared, the NEMO consortium will ensure its availability on Zenodo. The exact details regarding data publication and access will be documented in the final deliverable of WP5, D5.4 – NEMO Living Labs Use Cases Evaluation Results (Final Version). The detailed information of datasets gathered from the Use cases are documented below:

Dataset Name Aerial olive tree image dataset.	**URL to the data or published descriptor (doi, zenodo)** **TBD**		
Description **This dataset contains aerial images of olive groves, including images of trees (aerial leaf images) and control images (random images taken over an olive grove, but do not contain olive trees). It is being used for image classification to assist aerial precision spraying for the Smart Farming LL trials.** **Purpose of Data Collection:** **This dataset contains aerial images from the piloting site. These images were used for the training of the LL model. Moreover, as the real-time images taken by the drone for piloting purposes are stored in an on-premises datastore, the dataset contains the images used for the evaluation of the model.**	**Ownership & Licensing** The data are under the ownership of Synelixis S.A.	**Reuse Conditions** The dataset that will be made available, includes olive tree images without humans. This dataset is envisioned to be used from research institutes for training and validation of LL models.	**Data Availability** (Public/Confidential) The dataset is planned to be publicly available in popular open repositories, e.g. Zenodo, to encourage researchers to develop cutting edge classification models and increase the impact of the project.
	Storage Details	**Access Mechanisms**	**Metadata information**

Dataset Name Aerial olive tree image dataset.	URL to the data or published descriptor (doi, zenodo) TBD		
Data Origin: **The dataset consists of real-time images from the drone deployed in the area. These images reflect actual operating conditions and are hard to be reproduced.** **Data size:** **The dataset consists of training and real-time subsections. The training dataset is 2.9 GB, while each image taken by the drone is of ~31.85 MB (4608 × 3456 pixels)/image.**	The data are being stored in an on-premises datastore (MinIO).	The dataset can be accessed through a REST API in a read-only way.	The images comprising this dataset include GeoTIFF metadata. These metadata are being used for the creation of the Orthophoto maps and later for the identification of the point of interest for the aerial precision spraying.

Table 24 Dataset gathered in Use case SF_01

Dataset Name **Terrestrial olive tree image dataset.**	URL to the data or published descriptor (doi, zenodo) TBD		
Description: **The dataset consists of images acquired from camera on AGLV to assist terrestrial bio-spraying for the Smart Farming use cases.** **Purpose of Data Collection:** **This dataset contains terrestrial images taken using the AGVL from the piloting site. These images were used for the training of the LL model. The dataset is enhanced by the real-time video snippets taken during the validation phase at the farm.** **Data Origin: Observational** **The dataset consists of real-time images from the AGVL deployed in the area. These images reflect actual operating conditions and are hard to be reproduced.**	**Ownership & Licensing** The data are under the ownership of Synelixis S.A. **Storage Details** The video segments are stored in a Minlo datastore deployed	**Reuse Conditions** Access to the data is restricted due to the presence of humans in some of the images taken. **Access Mechanisms** The data are accessible only through the MinIO User Interface and only through the same Local Area	**Data Availability** (Public/Confidential) The dataset is confidential and will not be published. **Metadata information** No metadata is included in this dataset.

Dataset Name Terrestrial olive tree image dataset.	URL to the data or published descriptor (doi, zenodo) TBD		
Data size: The training dataset 1.5 million object instances, 80 object classes, object segmentation, and more.	on-premises.	Network that the AGVL is.	

Table 25 Dataset gathered in Use case SF_02

Dataset Name Power Quality Analyser data	URL to the data or published descriptor (doi, zenodo) TBD		
Description: This dataset consists of real-time measurements collected from Power Quality Analyzers (PQAs) installed in the MV/LV substations of Terni electrical distribution distribution network. It includes parameters such as voltage, current, power, and environmental conditions, recorded and processed locally by the devices. Purpose of Data Collection: This dataset is used in the application being developed for the grid operator to enhance grid visibility and support decision-making. The data is integrated into a Graphical User Interface, which serves as a centralized platform for real-time monitoring of power quality. By providing accurate measurements of voltage, current, power, and environmental conditions, the application helps the Distribution System Operator improve operational efficiency, detect anomalies, and ensure better power quality management.	**Ownership & Licensing** ASM, as the grid operator, owns the data, they have full control over how it is used and shared.	**Reuse Conditions** The dataset is strictly restricted to the consortium and can only be accessed and used by project partners. Any external use or sharing is subject to prior approval from ASM, the grid operator.	**Data Availability** (Public/Confidential) The dataset is classified as confidential, as it originates from critical energy infrastructure. Any misuse or unauthorized access could jeopardize grid security, operational stability, and regulatory compliance. Therefore, access is strictly limited to authorized consortium members, and appropriate security measures are in place to protect the data.
Data Origin: Observational The dataset consists of real-time sensor readings from Power Quality Analyzers deployed in the grid. These measurements reflect actual operating conditions and are not fully reproducible.	**Storage Details** The data is stored in an on-premises	**Access Mechanisms** Access to the dataset is strictly confined to the	**Metadata information** The dataset is provided in both JSON and CSV

Dataset Name Power Quality Analyser data	URL to the data or published descriptor (doi, zenodo) TBD		
Data size: The dataset generates approximately 1 MB per day per analyzer.	database managed by the grid operator. Historical data is archived within this database, ensuring long-term storage, accessibility, and retrieval for analysis and decision-making.	consortium, with ASM providing real-time data through an MQTT broker and historical data stored in an on-premises database for analytics. To ensure security, partners must provide their public IP, which is whitelisted to grant read-only access.	formats, structured to include essential metadata fields such as device ID, timestamp, and detailed power profiles—including active and reactive power, as well as voltage measurements across different phases. Although no globally unique and persistent identifier (like a DOI or Handle) has been assigned, the comprehensive metadata ensures that each record is self-descriptive and can be effectively parsed, queried, and analyzed.

Table 26 Dataset gathered in Use case SE_01

Dataset Name Historical Consumption and Production data (Wally)	URL to the data or published descriptor (doi, zenodo) TBD		
Description: Time-series data of loads wallys. Supports demand-response strategies and RES integration. Purpose of Data Collection: Supports forecasting and flexibility planning by	**Ownership & Licensing** ASM, as the grid operator, owns the data, they	**Reuse Conditions** The dataset is strictly restricted to the consortium and can only be accessed and used by project	**Data Availability** (Public/Confidential) The dataset is classified as confidential, as it originates from critical energy infrastructure. Any

Dataset Name Historical Consumption and Production data (Wally)	URL to the data or published descriptor (doi, zenodo) TBD		
analyzing energy consumption and production trends. Data Origin: Observational These measurements reflect actual operating conditions and are not fully reproducible. Data size: The dataset generates approximately 40 KB per day per analyzer.	have full control over how it is used and shared.	partners. Any external use or sharing is subject to prior approval from ASM, the grid operator.	misuse or unauthorized access could jeopardize grid security, operational stability, and regulatory compliance. Therefore, access is strictly limited to authorized consortium members, and appropriate security measures are in place to protect the data.
	Storage Details The data is stored in an on-premises database managed by the grid operator. Historical data is archived within this database, ensuring long-term storage, accessibility, and retrieval for analysis and decision-making.	**Access Mechanisms** Access to the dataset is strictly confined to the consortium, with ASM providing real-time data through an MQTT broker and historical data stored in an on-premises database for analytics. To ensure security, partners must provide their public IP, which is whitelisted to grant read-only access.	**Metadata information** The dataset is provided in both JSON and CSV formats, structured to include essential metadata fields such as device ID, timestamp, and detailed power profiles—including active and reactive power, as well as voltage measurements across different phases. Although no globally unique and persistent identifier (like a DOI or Handle) has been assigned, the comprehensive

Dataset Name Historical Consumption and Production data (Wally)	URL to the data or published descriptor (doi, zenodo) TBD		
			metadata ensures that each record is self-descriptive and can be effectively parsed, queried, and analyzed.

Table 27 Dataset gathered in Use case SE_01

Dataset Name Historical and Real-time Charging station data	URL to the data or published descriptor (doi, zenodo) TBD		
Description: **Time-series data of charging stations. Supports demand-response strategies and RES integration.** **Purpose of Data Collection: Supports flexibility planning by analyzing energy consumption trend and provides energy flexibility provision verification.** **Data Origin: Observational** **These measurements reflect actual operating conditions and are not fully reproducible.** **Data size: The dataset generates approximately 40 KB per day per device.**	**Ownership & Licensing** EMOT, as charging point operator, owns the data and has full control over how it is used and shared.	**Reuse Conditions** The dataset is strictly restricted to the consortium and can only be accessed and used by project partners. Any external use or sharing is subject to prior approval from EMOT, the charging point operator.	**Data Availability** (Public/Confidential) The dataset is classified as confidential, as it is related to energy flexibility provision to the grid and any misuse or unauthorized access could jeopardize grid security, operational stability, and regulatory compliance. Therefore, access is strictly limited to authorized consortium members, and appropriate security measures are in place to protect the data.
	Storage	**Access**	**Metadata**

Dataset Name Historical and Real-time Charging station data	URL to the data or published descriptor (doi, zenodo) TBD		
	Details The data is stored in an on-premises database managed by the charging point operator. Historical data is archived within this database, ensuring long-term storage, accessibility, and retrieval for analysis and decision-making.	**Mechanisms** Access to the dataset is strictly confined to the consortium, with EMOT providing real-time data through an MQTT broker and historical data stored in an on-premises database for analytics. To ensure security, partners authorized must receive credentials to access.	**information** The dataset is provided in both JSON and CSV formats, structured to include essential metadata fields such as device ID, timestamp, and detailed power profiles—including active and reactive power, as well as voltage measurements across different phases. Although no globally unique and persistent identifier (like a DOI or Handle) has been assigned, the comprehensive metadata ensures that each record is self-descriptive and can be effectively parsed, queried, and analyzed.

Table 28 Dataset gathered in Use case SE_02

Dataset Name Historical and Real-time electric vehicle data	URL to the data or published descriptor (doi, zenodo) TBD		
Description: Time-series data of electric vehicles. Supports demand-response strategies and RES integration. Purpose of Data Collection: Supports flexibility	**Ownership & Licensing** EMOT, as charging	**Reuse Conditions** The dataset is strictly restricted to the consortium and can only be	**Data Availability** (Public/Confidential) The dataset is classified as

Dataset Name Historical and Real-time electric vehicle data	URL to the data or published descriptor (doi, zenodo) TBD		
planning by analyzing electric vehicle State of Charge trend and provides energy flexibility provision verification. Data Origin: Observational These measurements reflect actual operating conditions and are not fully reproducible. Data size: The dataset generates approximately 40 KB per day per device.	point operator, owns the data and has full control over how it is used and shared.	accessed and used by project partners. Any external use or sharing is subject to prior approval from EMOT, the charging point operator.	confidential, as it is related to energy flexibility provision to the grid and any misuse or unauthorized access could jeopardize grid security, operational stability, and regulatory compliance. Therefore, access is strictly limited to authorized consortium members, and appropriate security measures are in place to protect the data.
	Storage Details The data is stored in an on-premises database managed by the charging point operator. Historical data is archived within this database, ensuring long-term	**Access Mechanisms** Access to the dataset is strictly confined to the consortium, with EMOT providing real-time data through an MQTT broker and historical data stored in an on-premises database for analytics. To ensure security, partners authorized must receive credentials to access.	**Metadata information** The dataset is provided in both JSON and CSV formats, structured to include essential metadata fields such as device ID, timestamp, and detailed power profiles—including active and reactive power, as well as voltage measurements across different phases. Although no globally unique

Dataset Name Historical and Real-time electric vehicle data	URL to the data or published descriptor (doi, zenodo) TBD		
	storage, accessibility, and retrieval for analysis and decision-making.		and persistent identifier (like a DOI or Handle) has been assigned, the comprehensive metadata ensures that each record is self-descriptive and can be effectively parsed, queried, and analyzed.

Table 29 Dataset gathered in Use case SE_02

Dataset Name: Video Quality Probes	(To be determined – can be a DOI or institutional repository link) TBD		
Description: This dataset contains Quality of Experience (QoE) measurements and additional image KPIs (Key Performance Indicators) related to media streams. It includes metrics such as resolution, bitrate, frame loss, latency, and image artifacts. **Purpose of Data Collection:** The dataset is collected to evaluate and optimize video streaming quality. It allows for real-time monitoring of media quality to ensure a high QoE for end users. The data is used for adaptive bitrate selection and troubleshooting network-related issues in media delivery. **Data Origin:**Observational (Real-time sensor data collected from video quality probes monitoring live media streams) **Data size:** Estimated to be in the range of 1-2 GB, depending on the number of monitored streams. Growth will depend on streaming volume and retention policies.	**Ownership & Licensing:** Owned by Universidad Politécnica de Madrid (UPM) or project consortium members. Licensing to be defined (e.g., open-source, research-only, or commercial license).	**Reuse Conditions** Restricted to project partners; data can be shared upon request with necessary agreements in place.	**Data Availability** (Public/Confidential) Confidential – Not publicly available due to intellectual property concerns and possible inclusion of sensitive streaming data.
	Storage Details Stored on on-premise servers and	**Access Mechanisms** Accessible via secured APIs and message brokers	**Metadata information** Data is stored in structured format (JSON/CSV) with

Dataset Name: Video Quality Probes	(To be determined – can be a DOI or institutional repository link) TBD		
	backed up to secure cloud storage for redundancy.	for real-time analytics, with authentication mechanisms in place.	metadata including timestamps, stream identifiers, and KPI descriptions. Persistent identifiers (DOI or institutional repository links) will be assigned where applicable.

Table 30 Dataset gathered in Use case SC_01

Dataset Name: AI Engine Runner Recognition	(To be determined – can be a DOI or institutional repository link) TBD		
Description: This dataset consists of AI-based analysis of race streaming footage, specifically recognizing runners and extracting their bib numbers using computer vision techniques. The dataset includes labeled images, bounding box coordinates, and extracted bib numbers. **Purpose of Data Collection:** The purpose of this dataset is to enable automated identification of runners in live race footage. This aids in generating real-time race analytics, providing personalized insights for spectators, and facilitating race tracking. **Data Origin:**Observational (Captured from live video streams and analyzed in real time) **Data size:** Estimated to be in the range of 1-2 GB.Growth will depend on streaming volume and retention policies.	**Ownership & Licensing:** Owned by Universidad Politécnica de Madrid (UPM) or project consortium members. Licensing will follow project agreements (e.g., research use, restricted distribution).	**Reuse Conditions** Restricted to project partners; data can be shared upon request for research purposes with necessary agreements.	**Data Availability** (Public/Confidential) Confidential – Not publicly available due to privacy concerns related to identifiable personal data in race footage.
	Storage Details Stored on on-premise servers and	**Access Mechanisms** Accessible via secured APIs and message brokers	**Metadata information** Data is stored in structured format (JSON/CSV) with

Dataset Name: AI Engine Runner Recognition	(To be determined – can be a DOI or institutional repository link) TBD		
	backed up to secure cloud storage for redundancy.	for real-time analytics, with authentication mechanisms in place.	metadata including timestamps and bib numbers,. Persistent identifiers (DOI or institutional repository links) will be assigned where applicable.

Table 31 Dataset gathered in Use case SC_01

Dataset Name Smart Watch Data	URL to the data or published descriptor (doi, zenodo) TBD		
Description This dataset will have data originating from, orientation data Accelerometer (AC), Heart Rate (HR), skin temperature (TEMP) from the smart watch. **Purpose of Data Collection:** These readings will be transferred to a ML node for emotion estimation for adapting the VR experience. **Data Origin: Data Origin** is observational from sensor readings **Data size:** • **For training:** ○ 11.5 million samples ○ 821MB • **For inference:**5 KB/s	**Ownership & Licensing** Mag and FHW own the data with full control	**Reuse Conditions** Restricted to the pilot trials.Reuse of the data is not possible	**Data Availability** (Public/Confidential) The data is strictly confidential and anonymized. Sharing is not allowed as it entails private biomarker data. An information sheet stating that is signed by participants.
	Storage Details Stored on the Edge components, is migrated to the cloud for processing before transferred to the edge again.	**Access Mechanisms** The actual dataset can be accessed only by internal components of the trial. Its processed results are available to pilot trial devices via a REST API.	**Metadata information** The dataset is provided in JSON formats, structured to include essential metadata fields such as device ID, timestamp, sensor reading and

Dataset Name Smart Watch Data	URL to the data or published descriptor (doi, zenodo) TBD		
			computed results of the ML nodes. Although no globally unique and persistent identifier (like a DOI or Handle) has been assigned, the comprehensive metadata ensures that each record is self-descriptive and can be effectively parsed, queried, and analyzed.

Table 32 Dataset gathered in Use case XR_01

Dataset Name Camera Video Data	URL to the data or published descriptor (doi, zenodo) TBD		
Description This dataset will have data originating from a camera video stream.To ensure privacy and compliance with data protection regulations, we do not retain any images containing faces when training our gesture recognition model. Instead, we extract key landmark points from each image and store them in a structured CSV file. Each row in the CSV file represents a single frame and contains numerical values corresponding to the normalized coordinates of specific key points detected in the image. These key points are typically derived from a human pose estimation model and include information about hand, arm,	**Ownership & Licensing** Mag and FHW own the data with full control	**Reuse Conditions** Restricted to the pilot trials.Reuse of the data is not possible	**Data Availability** (Public/Confidential) The data is strictly confidential and anonymized. Sharing is not allowed as it entails private biomarker data. An information sheet stating that is signed by participants.
	Storage Details Stored on the Edge components, is migrated to the cloud for processing	**Access Mechanisms** The actual dataset can be accessed only by internal	**Metadata information** The dataset is provided in JSON formats, structured

Dataset Name Camera Video Data	URL to the data or published descriptor (doi, zenodo) TBD		

and body positions while completely omitting facial features.
The format of the CSV file consists of:
1. A series of numerical values representing the X and Y coordinates of key points.
2. All values are normalized to a consistent coordinate space, ensuring model generalization.
3. The data is structured in a way that allows efficient processing for machine learning tasks without retaining any personally identifiable visual information.
This approach allows us to train the model effectively while maintaining strict privacy standards by avoiding the storage of raw images with identifiable human features.
Purpose of Data Collection: The video stream will be transferred to a ML node for gesture for adapting the VR experience.
Data Origin: Data Origin is observational from video camera
Data size:
• **For training:**
  ○ 200 samples (custom data created by MAG)
  ○ 40MB
• **For inference:**200 KB/s
	before transferred to the edge again.	components of the trial. Its processed results are available to pilot trial devices via a REST API.	to include essential metadata fields such as device ID, timestamp, sensor reading and computed results of the ML nodes. Although no globally unique and persistent identifier (like a DOI or Handle) has been assigned, the comprehensive metadata ensures that each record is self-descriptive and can be effectively parsed, queried, and analyzed.

Table 33 Dataset gathered in Use case XR_02