Overwatch

Iterative system integration and fine-tuning

Deliverable 4.2

Work package: **WP4**
Dissemination level: Public
Lead partner: ENG
Due date: 30/04/2025
Submission date: 24/11/2025

The OVERWATCH project has received funding from the Horizon Europe call "HORIZON-EUSPA-2021", topic HORIZON-EUSPA-2021-SPACE-02-52,

under agreement No. 101082320

Deliverable	Report on integration and fine-tuning
Deliverable No.	4.2
Work Package	4
Dissemination Level	PU
Author(s)	ISQ, ENGINEERING, INESCTEC, HOLO-LIGHT, LINKS
Co-Author(s)	
Date	
Status	Ongoing
Version	V2
Reviewed by (if applicable)	CINAMIL (Josè Alberto de Jesus Borges, Daniela Fraga)
Information used for referencing	N/A

Deliverable abstract	
Keywords	Validation, Tests, SAT, FAT

Table of Contents

Document revision history 9
Acronyms and abbreviations 10
Executive Summary 11
Use Cases 12
System Architecture 13
1. Meet the functional requirements and the end user requirements within the components: 15
  1.1. Augmented Reality technologies 15
  1.2. Artificial Intelligence-based backend Management System 18
  1.3. Drone terrain mapping and navigation 23
  1.4. Earth Observation technologies 24
  1.5. Fallback communication system 26
2. Meet the technical requirements (testing, performance, reliability and availability) for each involved module 29
  2.1. AR backend - Technical requirements met 29
    2.1.1. Testing requirements 29
    2.1.2. Performance requirements 31
    2.1.3. Reliability requirements 33
    2.1.4. Availability requirement 33
  2.2. AR Frontend - Technical requirements met 34
    2.2.1. Testing requirements 34
    2.2.2. Performance requirements 34
    2.2.3. Availability requirements 35
  2.3. Management backend - Technical requirements met 35
    2.3.1. Testing requirements 35
    2.3.2. Performance requirements 36
    2.3.3. Reliability requirements 36
    2.3.4. Availability requirements 37
  2.4. Web-based dashboard backend - Technical requirements met 37
    2.4.1. Testing requirements 37
    2.4.2. Performance requirements 37
    2.4.3. Reliability requirements 38
    2.4.4. Availability requirements 38
  2.5. Mapping from EO - Technical requirements met 39
    2.5.1. Testing requirements 39
    2.5.2. Performance requirements 39

2.5.3. Reliability requirements 40
2.5.4. Availability requirements 40
2.6. Drone navigation – Technical requirements met 41
2.6.1. Testing requirements 41
2.6.2. Performance requirements 41
2.6.3. Reliability requirements 41
2.6.4. Availability requirements 42
2.7. Drone terrain mapping – Technical requirements met 42
2.7.1. Testing requirements 42
2.7.2. Performance requirements 44
2.7.3. Reliability requirements 45
2.7.4. Availability requirements 45
2.8. Fallback connectivity module – Technical requirements met 46
2.8.1. Testing requirements 46
2.8.2. Performance requirements 47
2.8.3. Reliability requirements 48
2.8.4. Availability requirements 49
3. Fine-tuning in an operational environment integrated 52
3.1. Image Services timing 52
3.2. Processing times for the algorithm 53
3.3. System responsiveness 56
3.4. User Usability Feedback outcome 63
3.4.1. Evaluation of the Platform’s User Experience 63
3.4.2. UEQ Results 64
3.4.3. Final considerations 65
3.5. Highlighting on the Progress Beyond the State-of-the-Art 66
3.6. Feedback from Pilots (gathered during D4.3 / T4.4) 66
3.6.1. AR Feedback 66
3.6.1.1. Interface and Usability 67
3.6.2. Performance 67
3.6.2.1. Visualisation 67
3.6.2.2. Additional Features 67
3.6.2.3. Synthesis 67
3.6.3. Web Dashboard Feedback 68
3.6.3.1. Layout and Organisation 68
3.6.3.2. Filters and Search 68
3.6.3.3. Integration 68

3.6.3.4. Alerts and Notifications 68
3.6.3.5. Synthesis 69
3.6.4. Fine-tuning actions for feedback 69
3.6.4.1. For AR Applications 69
3.6.4.2. For Web Dashboards 69
3.6.4.3. Synthesis 70
3.7. Known Limitations 70
3.7.1. Earth Observation and Early Warning 70
3.7.2. Drone Operations and Environmental Constraints 70
3.7.3. Augmented Reality (AR) Latency and Connectivity 71
3.7.4. Thermal Sensor Calibration 71
Conclusion 72
Appendix A: Mapping of Validation Activities to D4.1 Methodology 1

Figures

Figure 1: Overwatch system architecture. 13
Figure 2 - EO Services response time averages 52
Figure 3 - Total processing times for wildfire and waterbody delineation algorithms tested on three areas of increasing sise. Average total processing times tends to be proportional to the sise of the requested area. 55
Figure 4 - Sise in Megabytes and square meters of the satellite images used for testing the performances of Damage Localisation and Change detection pipelines. 55
Figure 5 - Average processing time in seconds of the Damage Localisation and Change detection tested on three area of interest of increasing sise. 56
Figure 6 - Response Time Over Time 59
Figure 7 - Transactions per seconds 60
Figure 8 - Response Time Percentiles 61
Figure 9 - loader.io stress test results 62
Figure 10 - Planned Activities and correlation with User Experience Questionnaire 63
Figure 11 - Comparison of UEQ scores between the first and second test for flooding scenario 65
Figure 122 – Comparison of UEQ scores between the first and second test for the wildfire scenario 65

Tables

Table 1 - Functional and end user requirement for AR 15
Table 2 - Functional and end user requirement for AIMS 18
Table 3 - Functional and end user requirement for DT 23
Table 4 - Functional and end user requirement for DN 24
Table 5 - Functional and end user requirement for EO 24

Table 6 - Functional and end user requirement for FC 26
Table 7 - Testing requirements met for AR backend 29
Table 8 - Performance requirements met for AR backend 31
Table 9 - Reliability requirements met for AR backend 33
Table 10 - Availability requirements met for AR backend 33
Table 11 - Testing requirements met for AR frontend 34
Table 12 - Performance requirements met for AR frontend 35
Table 13 - Availability requirements met for AR frontend 35
Table 14 - Testing requirements met for MB 35
Table 15 - Performance requirements met for MB 36
Table 16 - Reliability requirements met for MB 36
Table 17 - Availability requirements met for MB 37
Table 18 - Testing requirements met for web dashboard 37
Table 19 - Performance requirements met for web dashboard 37
Table 20 - Reliability requirements met for web dashboard 38
Table 21 - Availability requirements met for web dashboard 38
Table 22 - Testing requirements met for Mapping from EO 39
Table 23 - Performance requirements met for Mapping from EO 39
Table 24 - Reliability requirements met for Mapping from EO 40
Table 25 - Availability requirements met for Mapping from EO 40
Table 26 - Testing requirements met for drone navigation 41
Table 27 - Performance requirements met for drone navigation 41
Table 28 - Reliability requirements met for drone navigation 41
Table 29 - Availability requirements met for drone navigation 42
Table 30 - Testing requirements met for drone terrain mapping 42
Table 31 - Performance requirements met for drone terrain mapping 44
Table 32 - Reliability requirements met for drone terrain mapping 45
Table 33 - Availability requirements met for drone terrain mapping 45
Table 34 - Testing requirements met for fallback connectivity 46
Table 35 - Performance requirements met for fallback connectivity 47
Table 36 - Reliability requirements met for fallback connectivity 48
Table 37 - Availability requirements met for fallback connectivity 49
Table 38 - JMeter tests main statistics 57
Table 39 - Number of participants per session and scenario 64

Document revision history

Version	Date	Description	Editor
V1	20/09/2023	First version	ISQ
V2	16/09/2025	First release ready for contribution	ENG
V2.1	04/12/2025	Version with revisions	ENG

Acronyms and abbreviations

AI	Artificial Intelligence
AR	Augmented Reality
BPL	Broadband over Power Lines
C2	Command and Control
CH	Chapter
CBI	Composite Burned Index
COTS	Commercial of the Shelf
D	Deliverable
DM	Decision Making
DSM	Digital Surface Model
DTM	Digital Terrain Model
ECS	Emergency Communication Systems
EGNOS	European Geostationary Navigation Overlay Service
EUB	End-Users Board
EO	Earth Observation
EMS	Emergency Management and Security
ER	Emergency Responders
FAT	Factory Acceptance Test
FR	First Responder
FSX	Full-Scale eXercise [an exercise that involves multiple organisations or functions and includes actual activities (ISO22300:2021, n.d.)]
GCS	Ground Control Station
GIS	Geographic Information Systems
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
LEO	Low Earth Orbit
LIDAR	Laser Imaging, Detection, and Ranging
HAS	High Accuracy Service
ISAR	Interactive Streaming for Augmented Reality
OSNMA	Open Service Navigation Message Authentication
NBR	Normalised Burn Ratio
NGO	Non-Governmental Organisations
R&D	Research and Development
SA	Situational Awareness
SAR	Search and Rescue
SART	Situation Assessment and Reconnaissance Teams
SAT	Site Acceptance Test
SDK	Software Development Kit
SotA	State of the Art
TO	Theatre of Operations
VR	Virtual Reality
XR	Extended Reality

Executive Summary

This document presents the results of the "Iterative system integration and fine-tuning" for the OVERWATCH project. It details the validation activities covering Earth Observation (EO), Remote Sensing, Machine Learning (ML), Augmented Reality (AR), and Fallback Connectivity modules.

Validation Status (FAT/SAT) Following the methodology defined in D4.1, the consortium successfully executed a series of Factory Acceptance Tests (FAT) and preliminary Site Acceptance Tests (SAT). The validation focused on verifying functional and technical requirements for each module. The technical verification of the AR Backend/Frontend, AI-based Management Backend (AIMS), Drone Navigation/Mapping, and Fallback Connectivity system has been completed.

Regarding performance, stress tests (using JMeter and loader.io) confirmed the system's responsiveness and stability under load, with average response times well within operational thresholds.

Modules Needing Remediation & Key Risks While the core integration is stable, specific constraints were identified during the fine-tuning phase regarding environmental limitations for drones and thermal sensor calibration. These risks are currently managed through specific operational protocols.

Deployment Recommendation Based on the successful verification of core functionalities, the OVERWATCH platform is recommended for deployment ("GO") in the upcoming Pilot demonstrations. Two additional versions of this document are expected to report on the final outcomes of the Full-Scale Exercises and the last iteration of fine-tuning.

Use Cases

As per deliverable 1.1, wildfires and floods are two of the most common and dangerous natural disasters that pose significant challenges to first responders and other civil protection stakeholders. The technologies developed in the project will be tested and validated through extensive testing in two pilots (FSX) in Portugal and Poland.

The OVERWATCH main modules, including Earth Observation, Drones, Fallback Communication, an Artificial Intelligence-based Backend Management System, and Augmented Reality, will be tested in these pilots, contributing to the validation of these integrated applications.

The use cases will primarily focus on wildfire and flood scenarios. They will serve as validation of the expected outputs of the OVERWATCH project, including rapid data collection, pre-, during, and post-response management with real-time data to produce updated maps and visualisations, a robust communication network, and support for situation assessment through actionable intelligence.

System Architecture

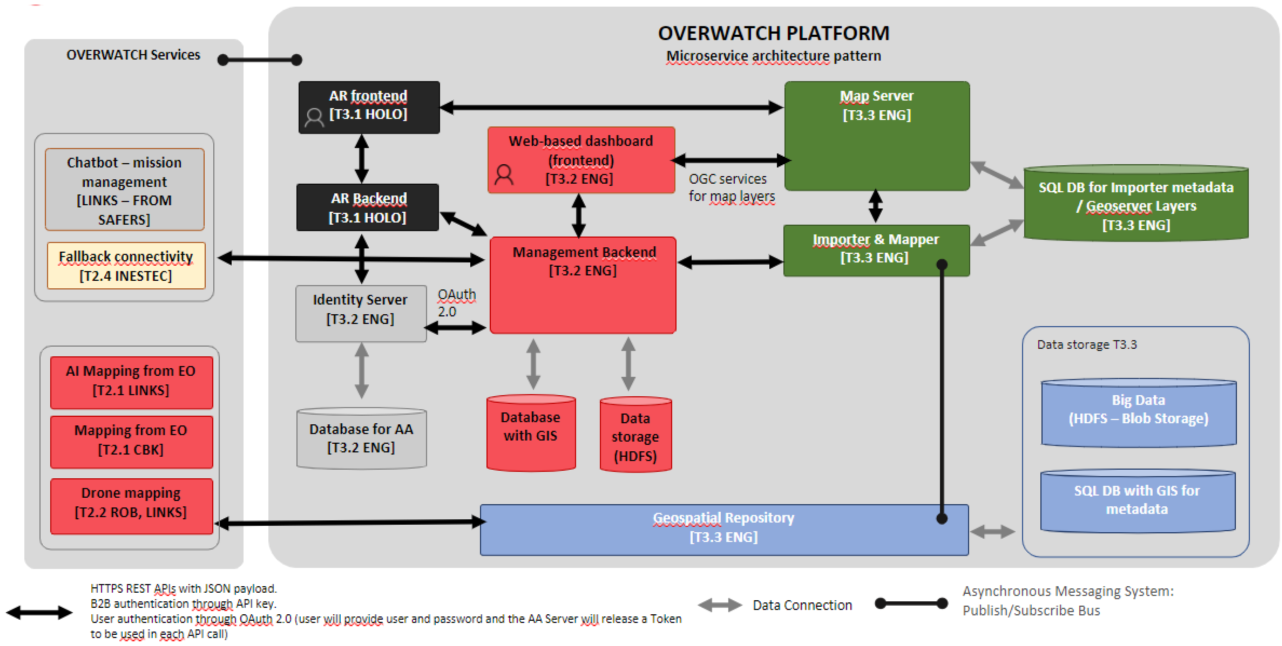


Figure 1: Overwatch system architecture.

Overwatch architecture is divided into two main components: the first provides useful data based on imagery from satellite Earth observation, drone acquisition, or legacy systems, while the second gathers the information and serves the results to the final user.

GIS layers and images are the primary data sources for the architecture; therefore, they have a specific, standardised pipeline. It begins by ingesting data, utilising a Big Data storage solution to receive and store this data through the Geospatial Repository. In an event-driven manner, the Geospatial Repository will share the signalling of the operation using a message broker, maintaining a loose coupling among the components that provide data from other components.

Once images and other GIS-enabled data have been stored, the underlying storage infrastructure triggers the processing phase that performs the required operations to import and harmonise the data into map layers. This will provide a mechanism for cataloguing and querying the imported data, along with metadata information, exploiting OGC-compliant services such as GeoServer [2].

The delivery of this information is possible because the management system can query and visualise the content that has been imported through OGC services, which provide the capability to serve and deliver maps to clients or end-users in a format that can be easily rendered and displayed on devices or applications, such as AR and web dashboards.

The two parts that comprise the architecture are connected using technologies and mechanisms that also provide fallback connectivity. In operational scenarios, the network connectivity is a crucial component to ensure the correct interactions between modules. However, disruptions or failures can occur due to various reasons, such as network outages, hardware failures, or environmental factors, above all, especially in crises. By considering the possibility of such disruptions, system designers incorporate backup or alternative connectivity options to mitigate the impact on the overall system functionality. The symbol ●━━● represents the presence of a message bus that will allow the exchange of all metadata between services and the backend, as well as all updates forwarded from services to the geospatial repository. The message bus is a standard component in the architecture and is based on a publish/subscribe mechanism.

1. Meet the functional requirements and the end user requirements within the components:

This chapter provides an overview of the functional and end-user requirements that have been successfully met during the development and implementation of the components: Augmented Reality (AR), AIMS (Automated Information Management System), Drone, Earth Observation (EO), and Flight Controller (FC).

These requirements were initially captured during stakeholder meetings and workshops conducted earlier in the project, as detailed in D1.2. While the original requirements were often defined at a high level, they have been carefully analysed, validated, and translated into actionable engineering specifications during the design and implementation phases.

The focus of this chapter is to highlight how the identified requirements (both functional and non-functional) were addressed and satisfied, showcasing the alignment between user expectations and the delivered components. This ensures that the system meets its intended purpose while maintaining a user-centric approach to design and functionality.

1.1. Augmented Reality technologies

Table 1 - Functional and end-user requirements for AR

Identifiers	Functional requirements	End user requirements	Description End user requirements	Requirement met (HOW)
AR-01	Map data format.	**AR-M-000** **AR-M-010**	Data provided must be able to integrate into Unity and must have selectable layers.	This requirement is achieved by integrating the Mapbox Static Tiles API and Sentinel Imagery via HTTPS requests.
AR-02	Layer Selection Functionality.	**AR-M-010** **AR-M-130**	The application must have the ability to select different layers.	The AR app allows users to select and toggle multiple layers, including elevation, contour, and Sentinel imagery layers.
AR-03	Remote Rendering.	**AR-M-020** **AR-M-030** **AR-M-090** **AR-M-100** **AR-S-020** **AR-C-010** **AR-SCA**	Remote application rendering must be supported. Client application (AR frontend) must send input methods to the AR backend server.	The AR app utilises the Hololight Stream SDK, where the AR app runs on a server (Windows PC) and is streamed to the client app on the AR device, providing full support for remote rendering.

AR-04	Metadata availability.	**AR-M-040**	The OVERWATCH must provide relevant metadata.	The AR app provides metadata (images, descriptions) regarding situations, missions, drones, and fire/flood events.
AR-05	Measuring features.	**AR-M-040**	The application should provide measuring features (e.g., distance, volume, etc.)	The AR app enables users to measure the distance (in a straight line and in altitude) between two selected points.
AR-06	Multi-platform support.	**AR-M-050** **AR-M-030**	The solution must function with multiple device types (e.g., HoloLens2, iOS device).	The current solution is compatible with Microsoft HoloLens 2 and Meta Quest 3.
AR-07	Image stability.	**AR-M-060**	The visualised content must remain stable in space through, for example, reprojection.	The requirement was met by developing improved pose prediction algorithms on the backend (server) and doing additional post-processing on the frontend (client).
AR-08	Object Recognition.	**AR-M-070**	Where relevant, the application should provide object recognition and tracking through third-party plugins.	As reported in D3.1, this requirement has been removed.
AR-09	Multi-user.	**AR-M-080** **AR-S-050**	The application must support multiple users in a single session and enable interaction between them.	The AR app currently supports up to 10 users with data synchronised between them. For demonstration purposes, two users will be used.
AR-10	Training module.	**AR-M-110**	The solution must provide some form of training (e.g., tutorial, user guide, on-site training session), dependent on pilot execution logistics.	The solution will provide a user guide and usage instructions within the application.
AR-11	External visualisation.	**AR-M-120**	AR application, as well as users' point of view (POV),	As the AR application runs on the server, the user's point of view is accessible

			must be accessible and shareable with additional screens.	and shareable, along with additional screens from the server.
AR-12	Possibility to integrate with external interfaces.	**AR-S-000** **AR-S-010** **AR-S-060**	The application should interface with additional sources (e.g., weather data, GPS, scenario simulation).	The AR application can interface with additional sources, including weather, altitude, and scenario simulation, via HTTPS requests.
AR-13	Protocol generation.	**AR-S-040**	The application should generate a protocol that includes relevant metadata (e.g., timestamps, measured values, pulled weather information) in the form of an XML or CSV file.	N.A.
AR-14	Research into the possibility of integrating with external interfaces.	**AR-C-000** **AR-C-030** **AR-C-040**	The application could interface with additional sources (e.g., weather, GPS, scenario simulation, etc.).	The research has been carried out, and it is reported in AR-12.
AR-15	Hardware Access based on biometric data.	**AR-C-060** **AR-SEC**	The used hardware could manage access based on biometric data (e.g., retina, fingerprint, etc.).	This requirement is achieved based on the chosen hardware. E.g., the server, Windows PC, as well as HoloLens 2, can be accessed via retina scan.
AR-16	User-friendly UI design.	**AR-ACC** **AR-USBT**	The UI must be minimalist and comply with user accessibility needs.	The user interface of the AR app has been refined through several design iterations to ensure ease of use. Feedback from end users during review meetings has helped achieve a simple and user-friendly UI.
AR-17	Network stability.	**AR-MNT** **AR-PERF** **AR-REL**	The network must be stable and robust.	The AR app is optimised to be stable under variable network conditions, leveraging the adaptive

				streaming technology. A dedicated router is used to ensure further stability. This requirement was successfully tested on-site during pilot demonstration. The network was a 5G connection and provided a stable streaming experience for users.

1.2. Artificial Intelligence-based backend Management System

Table 2 - Functional and end-user requirements for AIMS

Identifiers	Functional requirements	End user requirements	Description End user requirements	Requirement met (HOW)
AIMS.FR.1	The AIMS manages and ingests AI algorithm results	**AIMS-M-000**	The AIMS must manage the OVERWATCH system using artificial intelligence algorithms, e.g., to analyse historical/current data, identify patterns and trends relevant to improve risk assessment and decision-making.	The AIMS effectively meets the requirements by ingesting and analysing high-resolution satellite imagery from Earth Observation (EO) services. It utilises AI algorithms to identify patterns and trends, directly supporting the OVERWATCH system's risk assessment.
AIMS.FR.2	The AIMS supports the C2 in providing information on wildfires and floods	**AIMS-M-010**	The AIMS must support the C2 in managing emergencies along the response and recovery phases by providing information related to wildfires and floods.	AIMS provides C2 with essential wildfire and flood information via a comprehensive web dashboard. The system supports full access and navigation, including all interactions with key entities and relationships that define the emergency management process for both the response and recovery phases.

AIMS.FR.3	The AIMS will support the assessment and update of the SA by providing alerts and information on the current situation.	**AIMS-M-020**	The AIMS must have the ability to continuously assess the TO and update the SA of the C2 decision-makers and other emergency responders.	AIMS is continuously accessible online, providing real-time updates and alerts. This enables the continuous evaluation of information, directly supporting decision-makers and other emergency responders in their control and decision-making processes.
		AIMS-S-010	The AIMS should be able to provide real-time updates and alerts to emergency responders based on changing conditions.	The system leverages SSE[1] for live traffic; the traffic from the MB is forwarded to the WD.
		AIMS-S-060	The AIMS should be able to provide early warning alerts to emergency responders based on detected risks.	AIMS supports SA by sending early warning alerts to the WD via a secure channel. This ensures all alerts on detected risks are delivered to emergency responders promptly.
		AIMS-S-070	The AIMS should provide situational awareness to residents in the TO or potential emergency evacuation areas, enabling them to make informed decisions about evacuation.	AIMS provides crucial information on affected areas, enabling residents to assess their situation and make informed decisions about potential evacuation.
AIMS.FR.4	The AIMS stores information to manage the allocation of resources for	**AIMS-M-030**	The AIMS must facilitate and support C2 in allocating resources (personnel,	AIMS stores information from Earth Observation services, detailing the impact and potential growth

---

[1] Server-Sent Events (SSE) is an HTTP-based protocol that allows servers to push updates to clients over a single, long-lived connection. This is particularly useful for applications that require real-time updates, such as live feeds, notifications, or stock price updates.

	tasks during emergencies.		equipment, supplies, etc.) based on, for example, the severity, potential spread of the natural hazard, and the impact on human life and property.	of natural hazards to support C2 in resource allocation.
AIMS.FR.5	The AIMS stores the necessary information and supports the management and coordination of UAV usage.	**AIMS-M-040**	The AIMS must support the management and coordination of the operation and use of autonomous vehicles in the environment, e.g., the missions and airspace used by drones.	AIMS supports the management and coordination of UAVs by allowing drone mission creation through its dedicated APIs, directly fulfilling the user's needs.
AIMS.FR.6	The AIMS ingests and manages satellite imagery.	**AIMS-M-060**	The AIMS must access and ingest satellite imagery and other remote sensing data to supplement on-the-ground data collection efforts.	AIMS effectively stores and provides access to satellite data from remote sensing services, including metadata and binary imagery.
AIMS.FR.7	The AIMS ingests Copernicus EMS data elaborated by the AI mapping for EO.	**AIMS-M-070**	The AIMS must collect and analyse data from the Copernicus EMS.	AIMS stores and provides access to data from the Copernicus EMS, including its mapping and early warning components, for analysis and use.
AIMS.FR.8	The AIMS ingests metadata on resources to provide information to Ers.	**AIMS-M-120**	The AIMS must be able to identify resources such as water, fuel, roads, or alternative routes to escape danger and emergency progression, providing valuable information to emergency responders.	AIMS sends metadata to the Web Dashboard via a specific WMS, enabling emergency responders to identify and access crucial resources and information easily.

AIMS.FR.9	The AIMS integrates existing emergency management systems, storing the information that the SA will visualise.	**AIMS-M-130**	The AIMS must support integration with existing emergency management systems.	AIMS integrates with existing systems through its APIs, enabling it to leverage WMS data from services like EFFIS and EFAS for enhanced situational awareness.
		AIMS-C-050	The AIMS could support visualising the results of modelling the development of the emergency provided by third-party services.	AIMS integrates with existing systems using its APIs, allowing it to leverage external WMS data from services that comply with the OGC standard.
AIMS.FR.10	AIMS integrates external and data sources.	**AIMS-S-000**	The AIMS should be able to integrate with multiple external sensors and data sources to enhance data collection and analysis.	AIMS effectively integrates with external WMS sources through its APIs. This allows it to leverage data from services like Direção-Geral do Território “Dados Albertos”, fulfilling the need for enhanced situational awareness.
AIMS.FR.11	AIMS prepares data for visualisation on the web-based dashboard and AR backend.	**AIMS-M-150**	The AIMS must be able to provide real-time situational awareness to emergency responders.	AIMS support the Web Dashboard through the real-time channel via the HTTP SSE mechanism.
AIMS.FR.12	AIMS dispatches information for reports and visualisations to the web-based dashboard.	**AIMS-M-160**	The AIMS must be able to generate reports and visualisations that help emergency responders make informed decisions quickly.	The connection to the Web Dashboard is always active and is enabled through the platform's internet access.
		AIMS-M-170	The AIMS must be able to generate reports and visualisations for	The EO service outcome leverages remote and recent past events; this data information is

			post-action review and lessons learned.	available through AIMS's API.
		AIMS-M-180	The AIMS must provide situational awareness to all emergency responders, enabling them to make informed decisions in real-time and coordinated responses.	AIMS provides access to several users, including responders, and is managed through the Identity Server.
AIMS.FR.13	AIMS enables the sharing of information with ERAs.	**AIMS-M-200**	The AIMS must have the capability to share data and information with other emergency response agencies and organisations.	The AIMS provides the endpoint for downloading the metadata of the data layer shown in the Web Dashboard.
AIMS.FR.14	AIMS supports the SA with the visualisation of air quality monitoring	**AIMS-C-020**	The AIMS could have the capability to provide real-time air quality monitoring for responders and the public.	This feature is partially applicable, as air quality information can be obtained through an external WMS service.
AIMS.FR.15	AIMS models the development of the emergency.	**AIMS-C-060**	The AIMS could model the development of the emergency under different conditions to provide SA support in identifying effective routes for emergency responders to access/leave the affected area.	AIMS models emergency development by identifying affected areas and routes using Earth Observation (EO) imagery, which helps emergency responders manage the area effectively.
AIMS.FR.16	AIMS supports SA to provide recommendations.	**AIMS-C-080**	The AIMS could provide recommendations to emergency responders on the most effective response strategies.	AIMS provides high-level information that, while not direct recommendations, serves as a comparable strategic guide for emergency responders, aiding their decision-making process.

1.3. Drone terrain mapping and navigation

Table 3 - Functional and end-user requirements for DT

Identifiers	Functional requirements	End user requirements	Description End user requirements	Requirement met (HOW)
DT-01	Autonomous Flight and Navigation	**DT-M-020, DT-M-030, DT-M-040, DT-M-050, DT-M-060, DT-M-100,** **DT-S-010**	The drones must be capable of autonomous or remote flight, using flight control and stabilisation systems, algorithms, GNSS signals, and obstacle detection to navigate safely in diverse conditions. They should also support switching between piloted and autonomous modes of operation whenever necessary.	The utilised drones are customised commercial drones. The manufacturer assures the autonomous flight capabilities.
DT-02	Drone safety	**DT-M-070, DT-M-090**	The drones must have the ability to return to base or land safely in case of a malfunction or loss of communication and should be able to communicate their presence to other drones or the respective GCS to help coordinate sharing the same airspace.	This is met because the drone protocol establishs that the drone will return “home” in case of lost of communication
DT-03	Emergency Response and Environmental Adaptability	**DT-M-000, DT-M-010, DT-M-120,** **DT-S-020**	The drones must support emergency responders with functional capabilities and operate effectively in a diverse range of visibility conditions and adverse weather conditions like moderate wind,	The drone is capable of support emergency responders to identify and detect fire in a set of visible conditions like moderate wind, and are able to

			informing the GCS accordingly.	communicate this data to the GCS
DT-04	Data Relay/Streaming and Real-Time Data Processing	**DT-M-080, DT-M-110** **DT-S-020**	The drones must be capable of relaying/streaming data to the GCS/Geospatial repository and have on-board real-time processing capabilities of data from sensors.	The drones are able to use wifi /4G or 5g networks to communicate data to the GCS. The on-board processing is done for fire detection. However, based on the current drone configuration, it is not possible to orthorectify the images on board.

Table 4 - Functional and end-user requirements for DN

Identifiers	Functional requirements	End user requirements	Description End user requirements	Requirement met (HOW)
ASDM-1 (Accurate Secure Drone Mapping)	The system shall provide accurate and secure positioning information to the system.	**DT-M-020** **DT-M-050** **DT-M-110**	The drones must be able to fly autonomously or remotely piloted, with the ability to take off, hover, and land safely. The drones must use signals from GNSS to navigate in the airspace. The drones must have on-board real-time data processing capabilities from their sensors.	The drones fly both autonomously and remotely piloted using GNSS data. Furthermore, they are capable of processing visual and thermal data on board using embedded systems.

1.4. Earth Observation technologies

Table 5 - Functional and end-user requirements for EO

Identifiers	Functional requirements	End user requirements	Description End user requirements	Requirement met (HOW)
EO-1	The system shall collect multiple EO data sources	**EO-M-000** **EO-M-010**	The tool shall be capable of gathering and cross-referencing	The system leverages a variety of data sources,

	(EMS, GIS data, OSM).	**EO-M-020** **EO-S-050**	data from various sources.	mainly derived from satellite and EMS data.
EO-2	The system shall provide a range of thematic layers for different hazard types (fire, flood, damage assessment)	**EO-M-000** **EO-M-020** **EO-S-010** **EO-C-050**	The tool shall be capable of providing different output mappings.	The system provides thematic layers for wildfire, flood, and water extent delineation, as well as damage assessment.
EO-3	The service should support change detection using EO for different purposes (land cover, land use, environmental changes that could exacerbate hazards).	**EO-S-020** **EO-S-040**	Similarly, the different mappings may utilise pre- and post-hazard imagery to achieve a more precise output.	The system serves thematic layers produced by comparing pre- and post-hazard satellite imagery, enabling the detection of changes.
EO-4	The service should enable mappings on historical data and past satellite acquisitions.	**EO-C-000**	The module shall be able to retrieve historical data and provide the same mappings.	The service allows any date range, provided that satellite images are available.
EO-5	The service may provide a precise delineation and localisation of hazards and their current status.	**EO-C-030**	While the delineation and subsequent localisation are certainly feasible, the current status depends on the availability of satellite images.	The service provides delineation capabilities and estimates the current status, assuming the availability of satellite images.
EO-6	The system may provide helpful information for evacuation and other emergency measures based on EO.	**EO-S-000**	The detection of affected infrastructures can be provided using EO data after the event occurs, considering the limitations of the temporal and spatial resolution of the considered satellite network (e.g., for Sentinel-2, 10m and 5-6 days). Such detection could be taken into account in conjunction with other	The damage assessment layer, given a pair of high-resolution satellite images, provides information about infrastructure damage. The wildfire and flood layers provide a delineation map of a resolution of 10 meters. Providing optimal routes and

			local information (e.g., road closures) to determine the evacuation route. The implementation of a service aimed at providing an evacuation route, given the use of satellite mapping, is out of scope in OVERWATCH.	similar planning is outside the scope of this service.
EO-7	The system may provide alerts in the event of a detected emergency or escalation.	**EO-S-030**	Typically, alerts are issued according to the forecasted weather conditions during the early warning phase, while EO can provide detection capabilities (in the response or post-disaster phase). Therefore, we cannot implement any early warning alerting using EO data. Other instruments, such as MODIS and VIIRS, may also be evaluated for this purpose.	The system is used to outline and identify features after an event has occurred. The system lacks the capability for early warning alerts, which are typically issued based on forecasted conditions prior to an event.
EO-8	The service could provide ad-hoc layers to identify potential risks (e.g., landslides, wildfires, etc.).	**EO-C-010** **EO-C-020**	The feasibility of this requirement needs to be assessed for each kind of hazard (wildfire, landslide).	The system does not provide risk layers to identify potential hazards such as landslides or wildfires. This functionality is considered out of scope.

1.5. Fallback communication system

Table 6 - Functional and end-user requirements for FC

Identifiers	Functional requirements	End user requirements	Description End user requirements	Requirement met (HOW)
FCS-FR-1	Provide external connection with other OVERWATCH sub-modules and	FCS-FM-000 FCS-FM-010	Provide an external connection between the fallback connectivity and other Overwatch sub-	The fallback connectivity system can provide a wireless network that other Overwatch

	legacy systems (namely C2).		modules, C2 and other legacy systems.	systems can connect to.
		FCS-FS-010		
FCS-FR-2	Provide Connectivity (network and Internet access) to other systems.	FCS-FM-030	The fallback connectivity system must be able to provide support for reliable communications, e.g., Wi-Fi, Ethernet, and (if available) 5G, to provide redundant communication coverage to nearby local systems, allowing them to communicate locally and with the Internet (e.g., upload and download data from cloud systems).	The fallback connectivity system can provide Internet connectivity through 4G and Starlink.
		FCS-C-000		
FCS-FR-3	Tethered drone GCS functionalities.	FCS-FM-020	These requirements relate to the functional requirements of the tethered drone GCS and capabilities to system a reliable and functional system. Namely, the capability to ensure communication, a user-friendly interface, log all activities, and receive data.	The system logs the available bandwidth and all other communication-related items. The drone logs all drone-related activities (flight, position, altitude). The user interface is provided through the FCS GCS.
		FCS-FM-040		
		FCS-FM-060		
		FCS-FM-070		
		FCS-FM-080		
		FCS-FS-020		
FCS-FR-4	Tethered drone functional capabilities.	FCS-FM-050	The tethered drone must remain airborne for extended periods of time and land safely whenever it loses connection to GCS.	The tethered drone is expected to have 2 to 3 hours of autonomy, depending on weather conditions.

2. Meet the technical requirements (testing, performance, reliability and availability) for each involved module

This chapter outlines the process to ensure that each module involved in the system meets its defined technical requirements. The goal is to validate that all modules operate as expected, both individually and in conjunction with other system components, ensuring seamless integration and compliance with project specifications.

To ensure consistency with the D4.1, the following testing phases will be applied:

- **Factory Acceptance Tests (FAT):**
    - Conducted in a controlled environment to verify that each module meets its standalone technical requirements.
    - Focus on module-specific functionality, performance, and compliance with the design specifications.
- **Site Acceptance Tests (SAT):**
    - Performed in the deployment environment to ensure that modules integrate correctly with the system.
    - Validate end-to-end functionality, including communication and interoperability between modules.

2.1. AR backend - Technical requirements met

2.1.1. Testing requirements

Table 7 - Testing requirements met for AR backend

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	Stress-test network stability during single-user and multi-user use.	AR frontend. AR backend. Management backend.	Network.	Yes, this has been tested in the lab environment and will be demonstrated live during the pilot demonstration.
#2	Connectivity between the AR backend and the Management backend.	AR backend and Management backend.	Push and pull request.	Yes, this requirement is met, and the connection between the AR backend and Management Backend is

				established via server-side events (SSE), achieving the necessary push/pull requests.
#3	Latency of data transfer between the AR backend and the Management backend, to ensure a fluid display of all map tiles.	AR backend and Management backend.	Connection between the AR backend and the Management backend	Yes, the connection between the AR backend and the Management backend is established, ensuring a stable display of data.
#4	After receiving a visual alert from the operator in a scarce connectivity context, a mission should be created.	Mission Management (sample)	Create Mission	N.A. This starts from the Web Dashboard.
#5	Bandwidth tests to confirm the impact of the resolution drop.	AR frontend.	WebRTC	N.A.
#6	Turn map data layers on and off.	AR backend. AR frontend input methods. Management backend.		Yes, there is a feature to toggle the map data layers on/off.
#7	Turn the display of additional content on and off, such as weather information.	AR backend AR frontend input method External component		Yes, the app is designed to enable users to turn additional content on and off as needed.
#8	Ensure that during a multi-user session, all relevant movements and inputs are displayed to all users.	AR backend AR frontend	Photon	Yes, during a multi-user session, all relevant movements and data are synced via Photon Unity Network.

#9	Ensure data transfer between all interfaces.	AR backend		Yes, there is data transfer between all interfaces from the AR backend.
#10	Ensure simple usability of the UI elements.	AR frontend	MRTK, etc.	Yes, the AR app offers simple UI usability with options to drag, move, and scale.
#11	Ensure transmission of all relevant input data.	AR frontend AR backend	Custom send function, etc.	Transmission of all relevant input data occurs via streaming.
#12	Verify measurement results and ensure that the deviation remains within an acceptable range.	AR backend AR frontend input method		Yes, measurement results are within an acceptable range of deviation.
#13	Verify that the recorded details relevant to the protocol are accurately captured and stored.	AR backend		N.A.
#14	Verify input methods.	AR backend AR frontend	Identify if the selected method works as intended.	N.A.

2.1.2. Performance requirements

Table 8 - Performance requirements met for AR backend

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	Ensure the round-trip time remains within the 50ms threshold.	AR backend Network		Lab tests ensured a sub 50ms latency. This was not measure during the Pilot. However, based on network and infrastructure setup, the AR streaming experience was stable and smooth. This can further be measure and reported during the

				Poland Pilot demonstration.
#2	Impact of fast map interaction on GPU and CPU.	AR backend	Verify if the data is updating fluently or if there are long delays present.	Yes, the data is updated fluently without any delays.
#3	Image quality of the map tiles and layers.	AR backend AR frontend Management backend	Assess if long buffering moments are present.	Yes, this requirement is met, and there is no long buffering for the data to load.
#4	The application is compatible with multiple platforms.	AR frontend	Verify if the appropriate UI elements are displayed on the various selected platforms, i.e. iOS VS HoloLens 2	The iOS environment was not tested, as it was outside the scope. Meta Quest and HoloLens 2 were tested, and the UI is consistent; the application can be used on both devices.
#5	The system must be able to visualise processed map data and display the results of relevant analysis to improve first responders' situational awareness.	AR backend Management backend		The AR app can visualise and display map data as well as situations, missions, drone, and fire positions.
#6	The system must be able to stream scalable volumes of data to accommodate on-site commanders operating from satellite terminals.	AR backend	MRTK, etc.	Yes, the system is capable of streaming scalable volumes of data.
#7	Impact of 2+ users on the network.	AR backend AR frontend		Yes, this requirement was tested in the lab, and two or more users were able to connect and collaborate seamlessly.
#8	Verify that the visual quality in unfavourable environments is appropriate.	AR frontend		Yes, the visual quality in bright lighting conditions is lower due to the device's limited light sensitivity.

2.1.3. Reliability requirements

Reliability is an important aspect during an emergency. By utilising the remote rendering technology of the AR Backend, the primary focus will be on the Network requirements.

Table 9 - Reliability requirements met for AR backend

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The network must be at least Wi-Fi 5Ghz.	AR backend AR frontend		Yes. The devices are connected using WiFi 5Ghz.
#2	Bandwidth should be at least 40 Mbps to ensure clear image quality.	AR frontend	Identify the lowest acceptable resolution	Yes. This is device-dependent, i.e., it differs for HoloLens 2 and Meta Quest. The system optimises based on network conditions.
#3	Handling of poor network quality to reduce the risk of shutdown.	AR backend AR frontend	Stabilise the connection between the frontend and backend to avoid unwanted closures.	Yes. The backend is always running, and in the event of a poor network, the frontend can re-establish the connection.

2.1.4. Availability requirement

Table 10 - Availability requirements met for AR backend

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	If Cloud is being used, then measures must be taken to ensure server availability.	AR backend instance.		The cloud is not used, and hence this requirement is not met. The server is not running on the cloud but locally on a windows PC close to the device.
#2	The system should be compatible with intended devices and platforms, ensuring that emergency responders can access them from predefined locations and devices.	AR frontend.		Yes, the system adopts a device-agnostic approach, allowing users to access it from various devices. As long as the server and client are in the same network, the user can access from predefined locations.
#3	The system must always be available, even during peak usage or unexpected events, to ensure that emergency	AR backend AR frontend Management backend		Yes, the server, i.e., the AR application, is always running on the server. In the event of any disconnection, the client, i.e., the application on the headset, can

	responders can rely on it when needed.			re-establish the connection with the server.

2.2. AR Frontend - Technical requirements met

2.2.1. Testing requirements

Table 11 - Testing requirements met for AR frontend

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	Ensure a successful connection between the frontend and backend application.	AR frontend AR backend	Signalling server	Yes, there is a successful connection between the frontend and the backend.
#2	Transmission of all relevant input data.	AR frontend	Data stream, Custom send, Audio, Hand gestures, Gaze	Yes, all relevant input data are transmitted.
#3	Test the impact of loss of connection.	AR frontend		Yes, the impact is low since the backend is always running, and in the case of disconnection, the frontend can re-establish the connection.
#4	Image quality in unfavourable environments is appropriate.	AR frontend	Visual quality outdoors or in a room with bright light	Yes, the visual quality in bright lighting conditions is lower due to the device’s limited light sensitivity
#5	Set up a realistic test environment, considering factors such as people count, used platforms, network quality, speed, and urgency.	AR frontend AR backend Management backend		Yes, the setup has been used on the field for the demo test pilot.

2.2.2. Performance requirements

Most of the requirements for the AR Backend are similar to those present in the AR Frontend. Therefore, the section below will focus on only those relevant to the AR frontend.

Table 12 - Performance requirements met for AR frontend

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	Latency between UI input in the AR frontend and recognition in the AR backend	AR frontend AR backend	Custom send, Hand gestures, Gaze	Yes, the streaming solution optimises based on network conditions and user inputs, and data is seamlessly synced between the frontend and backend. However, onsite conditions at the Pilot site are yet to be tested.
#2	If a direct interface to the AR frontend is required, test the performance of the feature.	AR frontend	No feature currently directly interfaces with the AR frontend	There is no direct interface with Frontend.

2.2.3. Availability requirements

Table 13 - Availability requirements met for AR frontend

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The defined device list must include available clients.	AR frontend		Yes, clients are available (HoloLens 2, Meta Quest)

2.3. Management backend - Technical requirements met

2.3.1. Testing requirements

Table 14 - Testing requirements met for MB

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The system must be available at all times, even during peak usage or unexpected events.	Backend management	Service is up and running	Yes, the BM is accessible online and has been tested for peak usage.

#2	The AIMS system must be highly reliable, with minimal downtime and a low probability of failure.	Backend management	Test the throughput	Yes, downtime and failures are minimal.
#3	The system must ensure the availability of maps for the 2D front end and the AR front end.	Backend management	Test map layers	Yes, access to the maps is guaranteed through the proxy module of BM
#4	The OVERWATCH must ensure that the privacy of individuals is protected and that data is used only for its intended purposes.	Backend management	Test services providing sensitive data	Yes, each data point is user-aware, and the data is protected

2.3.2. Performance requirements

Table 15 - Performance requirements met for MB

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The system must be able to process and analyse large amounts of data in real-time, providing timely insights to emergency responders.	Backend management	Test requests on metadata and on maps	Yes, the EO services can provide insights in the order of minutes, provided that valid data is available.
#2	The system must be able to stream scalable volumes of data to accommodate on-site commanders operating from satellite terminals.	Backend management	Test the throughput	Yes, access to the data stored in the Data Lake and in the Geospatial repository is always guaranteed.

2.3.3. Reliability requirements

Table 16 - Reliability requirements met for MB

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The AIMS system must be highly reliable, with minimal downtime and a low probability of failure, to ensure it can provide the necessary information to emergency responders in critical situations.	Backend management	Test status of the backend	Yes, the BM status is under the control of its logs through the log analyser Grafana, Loki, and Promtail.

	The system should be easy to maintain, with minimal downtime required for maintenance and updates.	Backend management		Yes, maintenance is provided through SSH access to the premises that host the system.

2.3.4. Availability requirements

Table 17 - Availability requirements met for MB

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The AIMS must incorporate a backup and restore for all its contents	Backend management	n/a	n/a

2.4. Web-based dashboard backend - Technical requirements met

2.4.1. Testing requirements

Table 18 - Testing requirements met for web dashboard

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The system must be available at all times, during test usage	Dashboard	Management Backend	Yes

2.4.2. Performance requirements

Table 19 - Performance requirements met for web dashboard

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The system must be able to visualise large amounts of data in real-time, providing timely insights to emergency responders.	Dashboard	Management Backend	Yes
#2	The system should be easy to use and intuitive, requiring minimal training for emergency responders to utilise it effectively.	Dashboard	Management Backend	Yes, the ease of use has been proven through the user experience questionnaire.

2.4.3. Reliability requirements

Table 20 - Reliability requirements met for web dashboard

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The system should provide a user-friendly interface that enables emergency responders to access and utilise it efficiently.	Dashboard	Management Backend	Yes
#2	The system must be highly reliable, with minimal downtime and a low probability of failure, to ensure it can provide the necessary information to emergency responders in critical situations.	Dashboard	Management Backend	Yes

2.4.4. Availability requirements

Table 21 - Availability requirements met for web dashboard

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The system should be compatible with intended devices and platforms, ensuring that emergency responders can access them from predefined locations and devices.	Dashboard	Management Backend	Yes
#2	The system must be available at all times, even during peak usage or unexpected events, to ensure that emergency responders can rely on it when needed.	Dashboard	Management Backend	Yes

2.5. Mapping from EO - Technical requirements met

2.5.1. Testing requirements

Table 22 - Testing requirements met for Mapping from EO

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	Test the retrieval of EO imagery from external services.	Mapping from EO	Retrieve EO data.	Yes, the data is stored in the Geospatial Repository.
#2	Test the creation of a map request, either manually or directly from the main frontend UI.	Mapping from EO	Create a map request.	Yes
#3	Test the sending of update notifications for a given request.	Mapping from EO	Update map request.	Yes, through the SSE channel
#4	Test the upload of the retrieved EO data and layers produced by the service.	Mapping from EO	Conclude the map request.	Yes

2.5.2. Performance requirements

Table 23 - Performance requirements met for Mapping from EO

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The service shall be able to handle multiple subsequent requests	Mapping from EO	Handle map requests	Yes, each request is unique with a specific identifier
#2	The service shall be able to produce the required mapping in a relatively short amount of time (minutes or hours at most)	Mapping from EO	AI-based request processing	Yes, each request is continuously reported in terms of timing.
#3	The service shall be able to provide the required products on demand within 15 days.	Mapping from EO	Manual map request processing	Yes, actually the data is stored for more than 15 days

#4	The service shall be able to handle large data files and outputs (several GB)	Mapping from EO	Data management and upload	Yes, through the data lake plugin of the Geospatial Repository

2.5.3. Reliability requirements

Table 24 - Reliability requirements met for Mapping from EO

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The service shall be able to handle failures from external tools (e.g., EO data providers, ML models) and provide meaningful updates.	Mapping from EO	Map request processing.	Yes, if a request fails, a message is reported in the request's status information.
#2	The service shall be able to provide a retry mechanism for (possibly) recoverable errors.	Mapping from EO	Map request processing.	The service is stateless, but deterministic. Internally, the system provides a short-term retry mechanism to enhance resilience against transient issues, such as connection errors. For retries after a more extended period, a retry mechanism may be implemented on the requester's side by resending the original request's payload.

2.5.4. Availability requirements

Table 25 - Availability requirements met for Mapping from EO

Id	Description	Family of services	Service to test	Requirement Meet (Y/N)
#1	The service shall have an adequate bandwidth to download the required EO data and upload the resulting outputs.	Mapping from EO	Map request processing, data upload.	Yes, the tests did not report any bottleneck.
#2	The service shall maintain a stable connection with the message bus to receive map requests and send updates.	Mapping from EO	Map request creation, status updates.	Yes, the MB connection is guaranteed by the ACK policy.

2.6. Drone navigation – Technical requirements met

2.6.1. Testing requirements

Table 26 - Testing requirements met for drone navigation

Id	Description	Family of services	Service to test	Requirement Meet
#1	Test the validity of the outputs of the GNSS receiver selected under nominal conditions.	Accurate Positioning.	Create a position request.	N.A.
#2	Test the validity of the GNSS receiver's outputs, selected using OSNMA, under nominal conditions.	Accurate Positioning.	Create a position request with OSNMA.	N.A.
#3	Test the validity of the GNSS receiver's outputs, selected using OSNMA, under interfered conditions.	Accurate Positioning.	Create a position request with OSNMA under the interference.	N.A.

2.6.2. Performance requirements

Table 27 - Performance requirements met for drone navigation

Id	Description	Family of services	Service to test	Requirement Meet
#1	The module shall provide accurate positioning information	Accurate Positioning	The ability of the vehicle to use the outputs from the GNSS receiver correctly	Yes, the drone utilises the outputs from the GNSS receiver to obtain accurate positioning information. This is already assembled and implemented in the DJI drone from the manufacturer.

2.6.3. Reliability requirements

Table 28 - Reliability requirements met for drone navigation

Id	Description	Family of services	Service to test	Requirement Meet
#1	Under high interference, the module should alert the system	Accurate Positioning	Navigation capabilities under interference	This is already assembled and implemented in the DJI drone from the manufacturer.

2.6.4. Availability requirements

Table 29 - Availability requirements met for drone navigation

Id	Description	Family of services	Service to test	Requirement Meet
#1	The GNSS receiver module should provide, on its own, the processing power needed for calculating the PVT solution	Accurate Positioning	Data processing	This is already assembled and implemented in the DJI drone from the manufacturer.

2.7. Drone terrain mapping – Technical requirements met

2.7.1. Testing requirements

Table 30 - Testing requirements met for drone terrain mapping

Id	Description	Family of services	Service to test	Requirement Meet
#1	Test the creation of a mission after the operator triggers the visual alert in a context with scarce connectivity.	DJI M350 RTK	Create Mission	Yes, the mission is created in the Overwatch dashboard and sent to the drone.
#2	Test the drone's performance in various visibility conditions, including daytime, low-light/low-visibility, fog or smog, and nighttime.	DJI M350 RTK	Drone Sensors	The requirement is met under visibility conditions as specified for the drone: daytime, nighttime, and low-light environments. It features IP55 protection, allowing operation in light rain, fog, and smog, while maintaining stability. DJI does not

				specify a minimum visibility range; however, the system is engineered for reliable performance in complex conditions within these parameters.
#3	Test the drone's ability to switch between autonomous and piloted flight modes under diverse conditions.	DJI M350 RTK	Drone Navigation	Yes, all the flights are performed with a safety pilot, and the operating mode can be switched from autonomous to piloted under diverse conditions.
#4	Test the drone's ability to return to base or land safely in case of a loss of communication.	DJI M350 RTK	Drone Safety	Yes, the drone has always been connected to the remote control, which can be used to land safely and return to the takeoff position in case of communication loss after completing the mission.
#5	Test the drone's capability to relay or stream data to the Ground Control Station (GCS).	DJI M350 RTK	Data transmission	Yes, the drone was able to stream data to the GCS, namely RGB, thermal and orthorectified images. The drone position, status, and fire detection are

				also sent to the GCS.
#6	Verify the drone's ability to detect adverse weather conditions, such as strong winds, and provide relevant information to the GCS for informed decision-making.	DJI M350 RTK	Drone Sensors	This cannot be applied to the INESC TEC drone.
#7	Test the real-time mapping module deployed on the onboard computer, including its ability to make automatic adjustments to the drone's navigation path.	DJI M350 RTK		This is performed in the drone ground station after the flight, and then the results are sent to the GCS.

2.7.2. Performance requirements

Table 31 - Performance requirements met for drone terrain mapping

Id	Description	Family of services	Service to test	Requirement Meet
#1	Maximum fight time with the payload.	VTOL drone system	Drone Specifications	Approximately 35 minutes
#2	Maximum flight range with payload.	VTOL drone system	Drone Specifications	The drone can fly approximately 35 minutes at a speed of 23m/s (maximum speed), which translates to a distance of approximately 40 km.
#3	Wind speed resistance.	VTOL drone system	Drone Specifications	8/ms with payload
#4	Radio communication distance	VTOL drone system	Drone Specifications	2 km line of sight, but it depends on the antennas used.
#5	The system should generate and	VTOL drone system	Data transmission	N/A

	stream real-time maps to a ground station within a specified timeframe.			

2.7.3. Reliability requirements

Table 32 - Reliability requirements met for drone terrain mapping

Id	Description	Family of services	Service to test	Requirement Meet
#1	The drone's autonomous flight and navigation system should maintain consistent performance in diverse conditions.	VTOL drone system	Drone Navigation	The DJI control module does this.
#2	The drone's safety features, such as return-to-home and safe landing protocols, should function reliably in the event of a loss of communication.	VTOL drone system	Drone Safety	Yes, the drone has automatic procedures for a safe landing in case of communication loss.
#	The drone should withstand light rain, dust and smoke.	VTOL drone system	Drone Safety	No, the drone cannot fly in the rain.

2.7.4. Availability requirements

Table 33 - Availability requirements met for drone terrain mapping

Id	Description	Family of services	Service to test	Requirement Meet
#1	Checking the drone's ability to reliably transmit collected data to the geospatial repository,	VTOL drone system	Data transmission	Yes, the drone can transmit information using 4G and the fallback communications.

	whether over 4G or via a ground station.			
#2	Ensuring the drone can effectively interface with other modules in the OVERWATCH project, including data sharing and coordination.	VTOL drone system	Data transmission	Yes, the dashboard is able to specify the mission to be deployed on the drone and receive information from it.

2.8. Fallback connectivity module – Technical requirements met

2.8.1. Testing requirements

Table 34 - Testing requirements met for fallback connectivity

Id	Description	Family of services	Service to test	Requirement Meet
#1	Tethered drone flight capability	Fallback Connectivity System	All functions that make the tethered drone capable of flying, e.g., batteries, motors, autopilot, and missions	Yes, already tested in flight.
#2	Tethered system enrollment	Fallback Connectivity System	Deploy and land the drone with a tethered cable	Yes, already tested
#3	Tethered system power and fibre-optic	Fallback Connectivity System	Provide power and connectivity to the drone	Yes, already tested
#4	Fallback connectivity GCS functions	Fallback Connectivity System	Control the drone and provide a connection to external systems	Yes, already tested
#5	Tethered drone sensors	Fallback Connectivity System	Provide data and video to the GCS	Not going to be used
#6	Wireless Network Broadcast	Fallback Connectivity System	Check if the fallback network is correctly broadcasting a	Yes, already tested

			local wireless signal	
#7	LEO Satellite Connectivity	Fallback Connectivity System	Check if the LEO satellite connection service is operational and providing a functional backhaul Internet link.	Yes, already tested
#8	Cloud backend Connectivity	Fallback Connectivity System	Check if the connection with the cloud system is established	Yes, tested

2.8.2. Performance requirements

Table 35 - Performance requirements met for fallback connectivity

Id	Description	Family of services	Service to test	Requirement Meet
#1	Tethered drone autonomy	Fallback Connectivity System	Test tethered drone range and autonomy	2 to 3 hours, depending on weather conditions. The tether GCS only powers up the motors.
#2	Tethered drone maximum altitude capability	Fallback Connectivity System	Test tethered drone maximum altitude	100 meters
#3	Communications maximum throughput	Fallback Connectivity System	Test the maximum network throughput achievable for each communications link technology (local access and backhaul links)	The backhaul LEO link consistently achieved an Internet throughput averaging 293 Mbit/s for the Downlink, and 46 Mbit/s for the Uplink. The local access network, based on Wi-Fi 6, was capable of

				delivering an average throughput to the Internet (utilising the backhaul) of 277 Mbit/s for the downlink and 42 Mbit/s for the Uplink, thereby not representing a bottleneck to the Fallback Communications System.
#4	Communications maximum coverage range	Fallback Connectivity System	Test the maximum coverage range of each wireless communications access technology.	The Wi-Fi 6 range was only tested up to a radius of 350 m due to the field trial logistics. At this range, we obtained a Downlink throughput of 42.1-94 Mbit/s and an Uplink throughput of 23.2–33.9 Mbit/s using a regular single-antenna mobile phone. Based on the received signal strength, and depending on the terminal, the maximum coverage is expected to reach a radius of between 600 m and 1 km.

2.8.3. Reliability requirements

Table 36 - Reliability requirements met for fallback connectivity

Id	Description	Family of services	Service to test	Requirement Meet

#1	The Fallback connectivity system should be able to provide at least two alternative communication means (Wi-Fi and Ethernet, and, if possible, 5G) for establishing local connectivity with nearby systems in a reliable manner.	Fallback Connectivity System	Communications links (backhaul and local access) Reliability and Resilience	The Fallback Connectivity System was capable of providing both Wi-Fi 6 and Ethernet connectivity to the local systems in a reliable and resilient manner. No communication outages were detected. All critical ground systems utilised Ethernet connectivity but were also configured to use Wi-Fi 6 as a fallback technology. Access to the cloud systems was successfully tested using both wired and wireless local access technologies, while the backhaul link to the Internet was always Wireless through the LEO link.

2.8.4. Availability requirements

Table 37 - Availability requirements met for fallback connectivity

Id	Description	Family of services	Service to test	Requirement Meet

#1	Tethered Drone Power Availability.	Fallback Connectivity System.	Power-on of the communications payload and ground components.	Yes, the drone can be powered up by a tethered system. A generator powers up the ground components.
#2	Communications minimum network throughput.	Fallback Connectivity System.	Verify that the minimum expected network throughput for each access and backhaul link technology is being met.	Yes, all field tests exceeded the minimum expected throughput for each link technology.
#3	Communications minimum coverage range	Fallback Connectivity System	Test if the minimum expected coverage range for each wireless communications access link is being maintained.	The backhaul link, based on a LEO access, did not show any limitations, as it only requires a clear Line of Sight to the sky. The fibre link of the Tether system, connecting the air and ground components, maintained a stable full-duplex 1 Gbit/s link as expected. For the local wired links, Ethernet Cat6 UTP cables ensure a reliable connection up to 100 m, and fibre can be used for longer connections if needed. For the local Wi-Fi 6 links, we could always exceed the 100 m radius, with tests of up to 350 m distance (radius) clearly showing that the link still had SNR margin to increase the distance.
#4	Communications delay below the maximum threshold	Fallback Connectivity System	Test if the local and Internet connection delays are below the maximum delay threshold tolerable by the systems.	The average round-trip time delay to the Internet was 29 ms, with a minimum of 19 ms and a maximum of 38 ms. This delay (accounting to the local, LEO access, and Internet delay) is equivalent to a wired Internet access, and is tolerable by the systems, such as the augmented reality and dashboard access. The local access

				network round-trip time was below 2 ms.

3. Fine-tuning in an operational environment integrated

Content from the on-field demonstration will be an outcome of T4.4 activities (provided for May 2025). D4.4 will follow the procedures outlined in D4.1.

3.1. Image Services timing

The analysis of response times for Earth Observation services in wildfire monitoring has revealed significant operational characteristics that can be implemented in integrated operational environments.

The study involved a total of 60 requests distributed across three areas of interest with different extents:

- 99 km² (4 wildfires found in the last 5 years)
- 64 km² (4 wildfires found in the last 8 years)
- 39 km² (4 wildfires found in the last 8 years)

The performance of four service types:

- Sentinel-1 RAW image acquisition
- Burned Area Delineation
- Severity Estimation
- Active Fire & Smoke detection

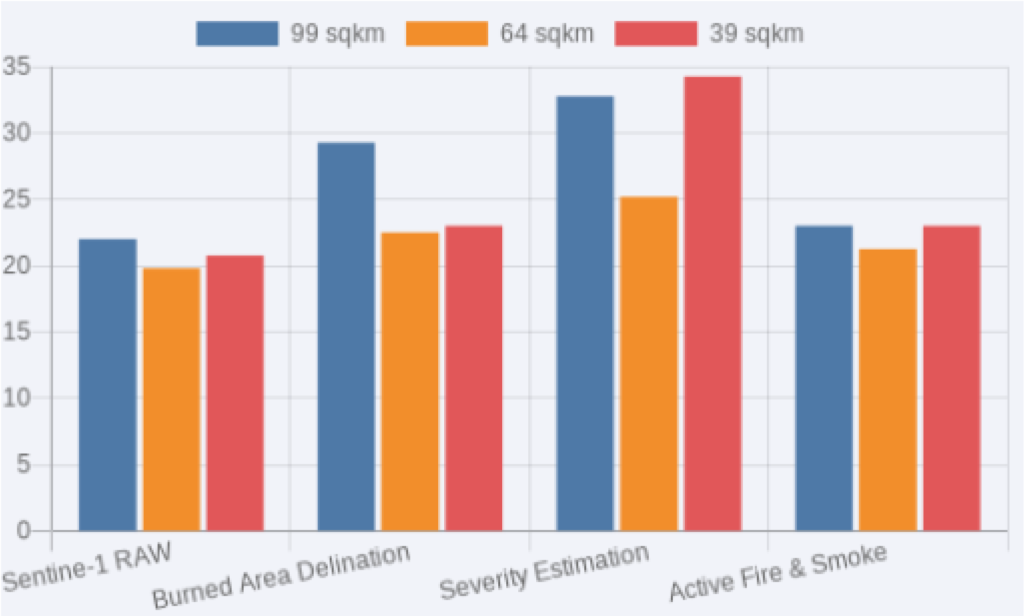


Figure 2 - EO Services response time averages

Figure 2 shows the mean processing times in seconds (Y-axis) for different Earth Observation services (X-axis) across three area extents.

The results demonstrate a clear differentiation in processing times among the different services. The Severity Estimation service proves to be the most computationally intensive, requiring, on average, the highest processing times (approximately 33 seconds for the 99 km² area and 34 seconds for the

39 km² area), followed by the Active Fire & Smoke service (approximately 23 seconds across all areas). The Burned Area Delineation service shows intermediate times (ranging from 22 to 29 seconds), while Sentinel-1 RAW image acquisition emerges as the most efficient, with times consistently below 23 seconds for all configurations.

A particularly relevant aspect revealed by the analysis is the substantial independence of processing times from the area of interest. This apparently counterintuitive characteristic can be explained through the parallel processing architecture implemented in the system. Areas are automatically subdivided into optimally-sised tiles, each processed in parallel through independent download and processing operations. This parallelisation strategy enables consistent response times regardless of the total area sise, effectively leveraging available computational resources.

The design choice to implement a limit on the number of parallel processes, defined during the development phase, ensures system stability and prevents undesired effects such as resource overload or overall performance degradation. This configuration guarantees predictable and reliable service behaviour, a fundamental characteristic for operational applications in critical scenarios such as wildfire monitoring.

3.2. Processing times for the algorithm

In the realm of digital image analysis, 'inference' refers to the process of deriving meaningful information from images using computational techniques. This often involves the application of machine learning models or algorithms to interpret and analyse visual data, enabling tasks such as object detection, classification, and segmentation.

Once the required geospatial data is collected, the inference computation phase employs machine learning models or other algorithms to produce the desired results. This phase involves preprocessing the input data, applying the relevant model for inference, and storing the results to enhance efficiency. It generates outputs, such as GeoTIFF files, in formats suitable for subsequent tasks. Typically, each pipeline is linked to a specific model, which is then applied to one or more images obtained during the earlier data acquisition stage.

a. Each processing pipeline has unique performance characteristics based on the data it handles, the complexity of its operations, and whether it relies on deep learning inference. As summarised in Figure 3, all wildfire and flood mapping algorithms were tested on three regions of different sises (a-e). Damage assessment pipelines (f-g) were tested on three different high resolution satellite images of different sises (see Figure 4) to estimate average processing times summarised in Figure 5, The following list summarises the purpose, steps and processing times of each pipeline, along with how much of the total time is consumed by model inference on average: Burned Area Delineation: This algorithm retrieves Sentinel-2 imagery, preprocesses it, and applies a deep learning model to identify areas affected by fire (see Deliverable 2.1 for technical details). The results are then uploaded to the data lake for further use. Processing times increase with area sise, ranging from approximately 9.09 seconds for a 39 km^2 area, to 9.35 seconds for a 64 km^2 area, and 15.82 seconds for a 99 km^2 area. Inference accounts for roughly 40% to 50% of the total processing time.
b. Severity Estimation: Building upon the burned area delineation, this algorithm first identifies burned zones and then estimates the severity of fire damage using an additional deep-learning model. Processing durations are 17.31 seconds for a 39 km^2 area, 18.22 seconds for 64 km^2 ones, and 31.36 seconds for 99 km^2 areas. Inference is much slower compared to other pipelines because it leverages two models.
c. Sentinel-2 RAW: Designed primarily for display and visual inspection of the satellite image, this pipeline does not involve any deep learning model. It focuses on retrieving Sentinel-2

imagery, applying basic preprocessing steps, and formatting the data for visualisation purposes. The absence of inference makes it computationally lightweight and relatively fast, with processing time primarily consumed by I/O operations and image processing.

d. Active Fire and Smoke Detection: This algorithm detects active fires and smoke using selected Sentinel-2 bands (B02, B03, B11, B12) and applies thresholding and band combinations to produce the result. No deep learning models are used here, which results in significantly faster processing compared to AI-based pipelines. The speed advantage of this pipeline is apparent when compared to others in Figure 3.
e. Waterbody Delineation: Using Sentinel-1 Radiometrically Terrain Corrected data, this algorithm performs waterbody segmentation via a deep learning model (see Deliverable 2.1 for technical details). The model requires a minimum input sise of 224×224 pixels. When dealing with small areas that fall below this threshold, padding must be added to meet the input sise requirement. This sometimes results in longer processing times for small areas compared to larger ones. Specifically, the processing time is 3.35 seconds for a 39 km² area, 1.74 seconds for a 64 km² area, and 1.88 seconds for a 99 km² area.
f. Damage Localisation: This algorithm processes a pair of images provided by the user, performs necessary preprocessing, and feeds them to a deep learning model to detect structural or environmental damage (refer to Deliverable 2.1 for further information). Processing times scale with image sise, starting at 2.60 seconds for 0.08 km^2 areas, 6.88 seconds for 0.37 km^2, and reaching 21.72 seconds for 1.59 km^2 areas. The increased complexity with area sise reflects both the growing data volume and longer model inference times.
g. Change Detection: Functionally similar to damage localisation, this pipeline also takes two user-supplied images, preprocesses them, and uses a dedicated model to identify changes over time. The only significant difference is the model used, which is optimised for temporal change analysis rather than damage classification. Processing times are slightly higher, with 3.12 seconds for a 0.08 km^2 area, 8.54 seconds for 0.37 km^2, and 28.17 seconds for 1.59 km^2 area.

Pipelines that incorporate deep learning models typically exhibit longer processing times, with inference often accounting for the majority of the computational load, especially in multi-stage workflows such as severity estimation. Conversely, pipelines based on traditional image processing or spectral analysis methods, such as those used in Sentinel-2 RAW and active fire/smoke detection, are more efficient due to the absence of heavy inference workloads. These differences underscore the importance of tailoring resource allocation to the processing demands of each pipeline.

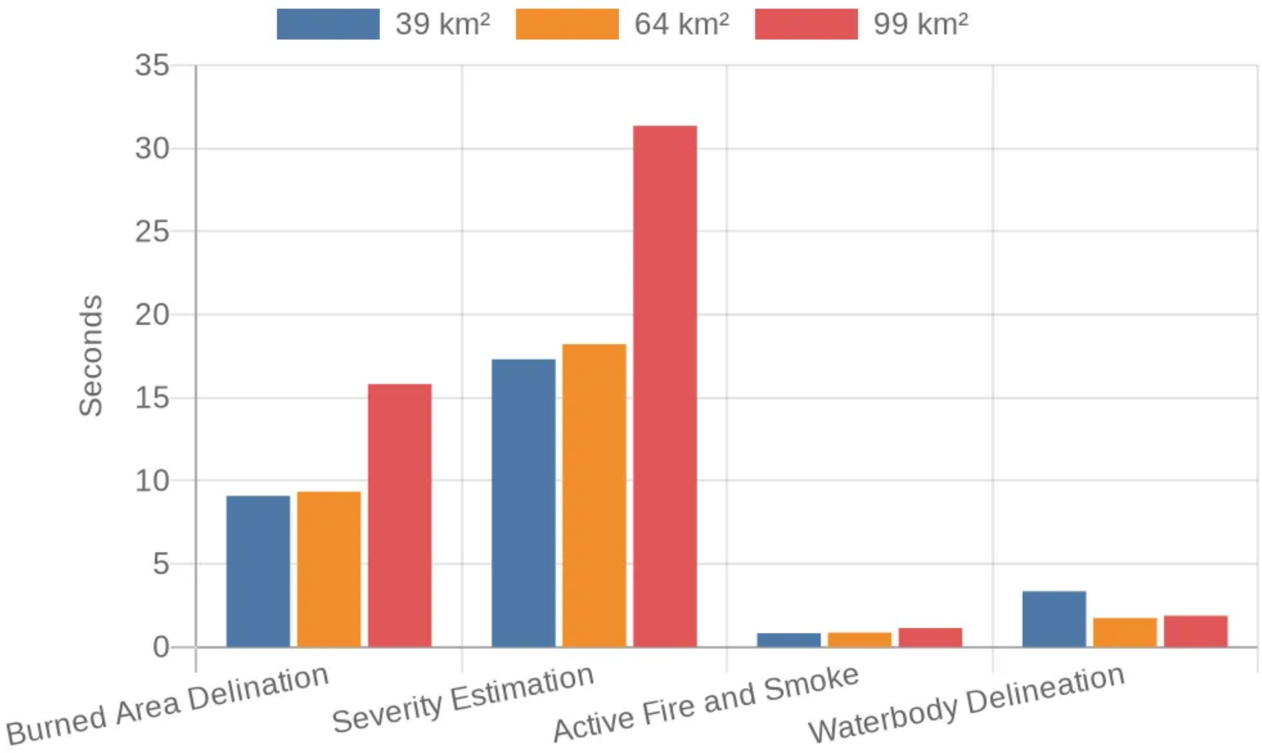


Figure 3 - Total processing times for wildfire and waterbody delineation algorithms tested on three areas of increasing sise. Average total processing times tend to be proportional to the sise of the requested area.

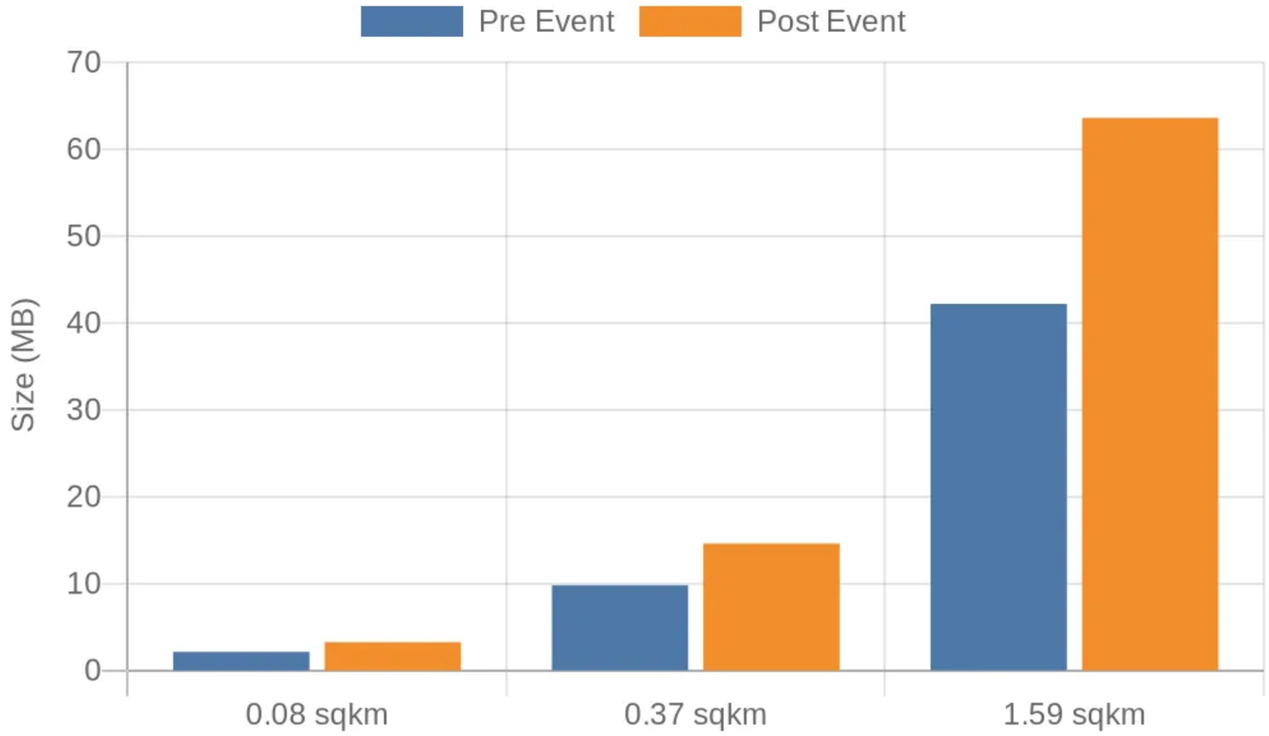


Figure 4 - Sise in Megabytes and square meters of the satellite images used for testing the performance of the Damage Localisation and Change Detection pipelines.

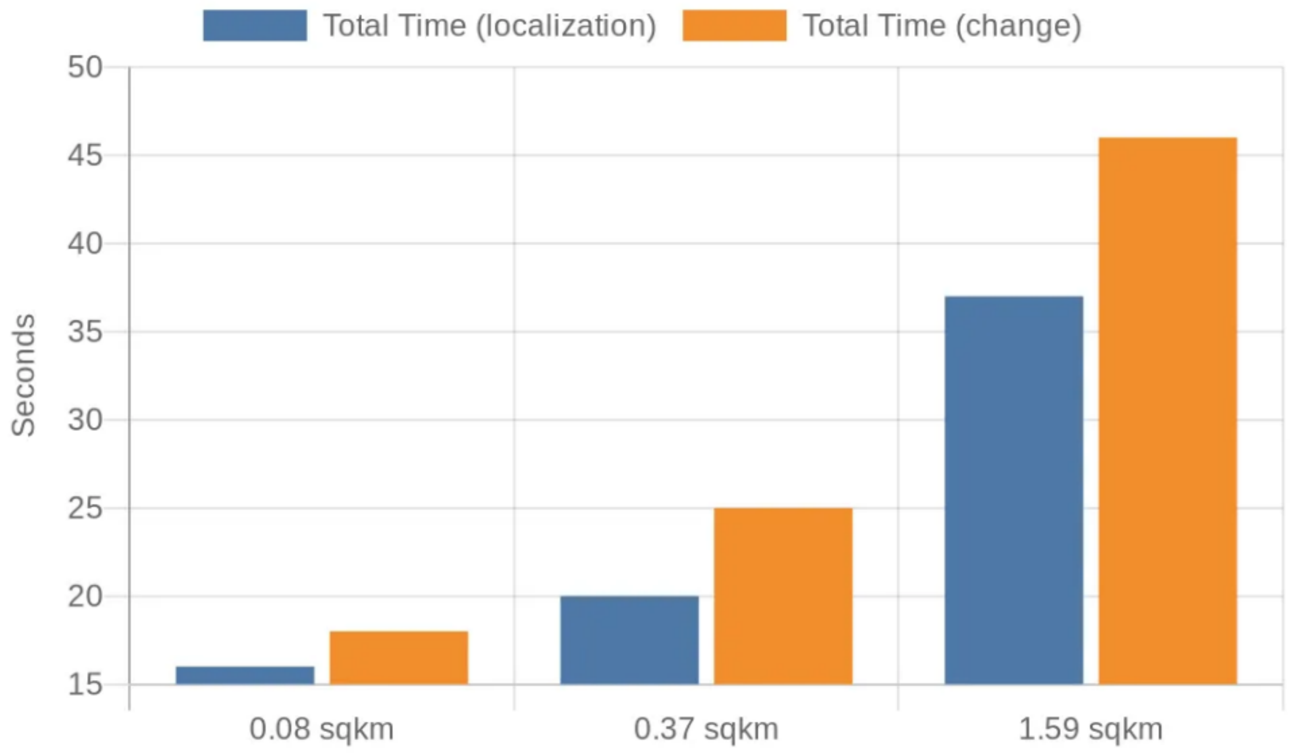


Figure 5 - Average processing time in seconds of the Damage Localisation and Change detection tested on three areas of interest of increasing sise.

3.3. System responsiveness

In an iterative system integration process, ensuring the system responsiveness of a web dashboard is critical. Responsiveness is not just about speed; it is about the system's ability to consistently and reliably handle user requests, particularly under increasing load. To thoroughly evaluate this aspect, we utilised Apache JMeter, an open-source tool for load and performance testing. JMeter allowed us to simulate a realistic user workload and gather essential performance metrics. By analysing data such as average response times, percentile values (like the 90th and 95th), and error rates, we were able to measure how the dashboard performs under various conditions precisely. This data provides a clear, quantitative basis for identifying potential bottlenecks and ensuring that our integrated system meets the performance expectations of end-users at each stage of development.

To accurately measure the performance and responsiveness of the web dashboard, a detailed test scenario was developed to simulate a typical user navigation flow. This approach ensures that the performance metrics gathered are relevant to real-world usage and reflect the system's behaviour under a representative workload.

The simulated user journey was designed as an iterative process, comprising a series of sequential HTTP requests that mimic a standard workflow. The following five distinct steps defined this scenario:

1. Opening the Web Page and Reaching the Landing Page: The test begins by simulating a user's initial access to the web dashboard's main URL. This step measures the time required for the initial page load, including all associated static assets such as CSS, JavaScript, and images.

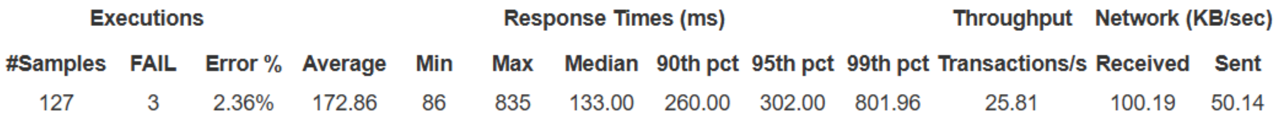
2. Login Authentication: Following the landing page load, the scenario proceeds with the user logging in. This involves submitting authentication credentials and validating a successful login. This step is critical for evaluating the performance of the system's authentication services.
3. Viewing Main User’s Situation Awareness: After a successful login, the user navigates to the main dashboard to view the primary "situation awareness" displays. This action simulates the loading of key graphical and data-rich components, which provide insights into the performance of the core dashboard functionalities.
4. Viewing Layers from Previous Analysis: The user then selects a specific situation awareness display and requests the visualisation of layers derived from previous analyses. This step tests the system's ability to efficiently retrieve and render multiple data layers, which can significantly impact performance.
5. Viewing Details and Metadata: Finally, the test simulates the user's request for detailed information and metadata related to the first layer of situational awareness. This final action measures the performance of retrieving granular data and highlights the system's efficiency in handling specific, detailed queries.

This structured navigation scenario provides a comprehensive view of the system's performance, from initial access and authentication to detailed data retrieval. By analysing the response times and throughput for each of these steps, we can identify performance bottlenecks and confirm the system's readiness to handle production-level workloads.

To execute the test scenario, the following parameters were configured within the Thread Group settings in Apache JMeter:

- Number of Threads (Users): 5 - This value simulates the number of concurrent users accessing the web dashboard simultaneously. A value of five was chosen to represent a small but manageable load, providing a baseline for performance analysis.
- Ramp-up Period (in seconds): 1 - The ramp-up period defines the time it takes for JMeter to "ramp up" to the full number of users specified in the thread group. In this test, all five users were started within a one-second interval. This gradual increase helps to avoid a sudden, unrealistic load spike on the server, ensuring a more accurate measurement of its behaviour under typical conditions.
- Loop Count: 5 - This setting specifies the number of times the test scenario (the user journey from step 1 to 5) will be executed by each user. Each of the five users will perform the full navigation sequence five times. This helps to generate a sufficient volume of data points for reliable statistical analysis.

In summary, the test was configured to simulate five concurrent users, with each user initiating their actions within a one-second ramp-up period and then repeating the entire navigation flow five times. This controlled execution process enabled the collection of consistent and repeatable performance data.



Executions			Response Times (ms)							Throughput	Network (KB/sec)	
#Samples	FAIL	Error %	Average	Min	Max	Median	90th pct	95th pct	99th pct	Transactions/s	Received	Sent
127	3	2.36%	172.86	86	835	133.00	260.00	302.00	801.96	25.81	100.19	50.14

Table 38 - JMeter tests main statistics

The comprehensive overview of the test results, as depicted in Table 38, provides a precise and quantitative measure of the system's performance. The data is organised into four main categories: Executions, Response Times, Throughput, and Network traffic, each offering a specific insight into the system's behaviour under load.

1. **Executions** provides a summary of the total number of requests made and their success rate. The number of samples indicates the total number of requests sent during the test, in this case, 127. Of these, FAIL indicates that three requests were unsuccessful, resulting in an Error Rate of 2.36%. This metric is crucial for identifying reliability issues, as a high error rate can indicate that the system is unable to handle the given workload.
2. **Response Times (ms)** is a critical value for assessing system responsiveness, with all values measured in milliseconds (ms):

- *Average*: The average response time for all requests was 172.86 ms. This value provides a quick, general overview of the system's speed.
- *Min* and *Max*: These values represent the fastest and slowest response times recorded during the test (86 ms and 835 ms, respectively). The significant gap between the minimum and maximum values can indicate performance inconsistencies or occasional spikes in response time.
- *Median*: The median, or 50th percentile, of 133.00 ms shows that half of the requests were completed in less than this time. This is often a more reliable indicator of typical user experience than the average, as a few extremely slow requests do not skew it.
- *Percentiles* (90th, 95th, 99th pct): These metrics are key to understanding the user experience for the majority of users. For example, the 90th percentile of 260.00 ms indicates that 90% of all requests were completed in 260.00 ms or less. Similarly, the 95th percentile shows that 95% of requests took 302.00 ms or less, while the 99th percentile highlights that 99% of responses were received within 801.96 ms.

3. Throughput measures the system's ability to handle a workload over time. The Transactions/s value of 25.81 represents the number of requests the server was able to process successfully per second. This is a vital metric for determining the system's capacity and scalability.
4. Network (KB/sec) This section provides a summary of network traffic. The Received and Sent values, measured in kilobytes per second (KB/sec), indicate the amount of data transferred to and from the server during the test. This data can help identify bandwidth usage and potential network-related bottlenecks.

Based on the values from the JMeter report, the system's performance can be considered very good for navigation, especially for the vast majority of users.

The main strengths are:

Average Response Time: The average response time of 172.86 ms is excellent. Users generally perceive response times below 200 ms as "instantaneous," which means the system is highly responsive for a typical user.

Reliability (Error %): The low error rate of 2.36% is a positive sign. With only three failed requests out of 127 total, the system demonstrates strong stability under the simulated load, without frequent failures.

User Experience (Percentiles): The 90th and 95th percentile values (260 ms and 302 ms, respectively) are fantastic. This indicates that 95% of all requests were completed in approximately one-third of a second, ensuring a very smooth and responsive experience for nearly all users.

Indeed, the points to monitor are:

Maximum Values (Max and 99th Percentile): The maximum value of 835 ms and the 99th percentile of 801.96 ms are the only values that warrant further attention. Although these times are rare (affecting only 1% of the requests), they suggest that a tiny percentage of users may have experienced a noticeable delay (almost one second).

The data confirms that the system is highly performant and robust under the tested load. The navigation experience is exceptionally fluid for the overwhelming majority of users.

The following charts (shown in Figure 6, Figure 7 and Figure 8) provide a visual representation of the system's performance. It's essential to note that each coloured line on these graphs represents a specific, single API call to a unique resource or endpoint, rather than the overall performance of the user flow. Since the test simulates a fixed navigation sequence, these individual calls are not repeated across iterations for a single user thread, which is why the number of lines is fixed. Instead of analysing each line, the key is to interpret them as belonging to a specific logical group of actions, which we defined in the test methodology. For clarity, we have categorised the lines according to these groups:

Landing Page: Represents the initial page load.
Login: The authentication call to the Identity Server.
Situation Awareness: Calls to retrieve the main dashboard displays.
Layers: Calls to load specific data layers from previous analyses.
Layer Details: The final calls to retrieve detailed metadata for a layer.

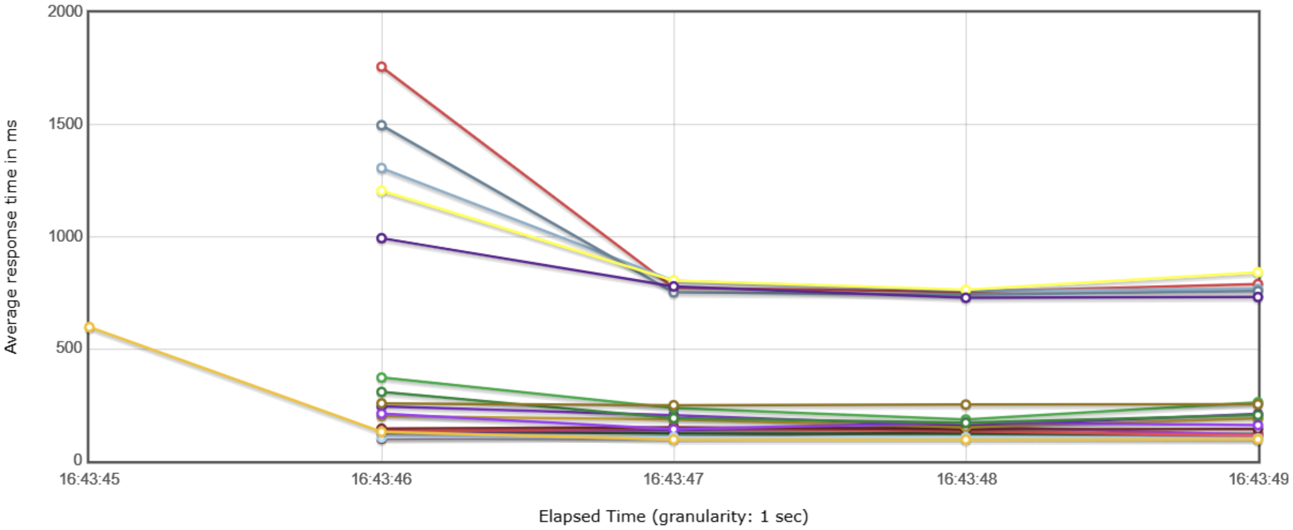


Figure 6 - Response Time Over Time

The Response Times Over Time graph in Figure 6 provides one of the most insightful views into the system's performance, illustrating how responsiveness evolves over the test duration. This chart reveals two key behaviours: an initial "warm-up" phase followed by a period of remarkable stability.

The most notable feature of the graph is the significant drop in response times between the beginning of the test and the moment the system reaches a stable state. This initial spike, evident in the higher cluster of lines, represents the system's warm-up phase. During this period, components such as the Identity Server and the web dashboard are busy initialising connections, loading data into cache, and preparing resources to handle the full user load. The subsequent drop in times demonstrates that once the system is "warmed up," it becomes highly efficient and ready to serve requests.

Following the warm-up, the graph shows a clear period of sustained stability. The response times for all requests flatten out, indicating that the system is handling the workload consistently without any signs of degradation. This is a very positive result, as it confirms that the web dashboard can maintain high performance under the tested load.

Finally, the chart distinctly separates the requests into two performance clusters. The upper group of lines represents the more complex operations, specifically the login phase, which is routed through a dedicated Identity Server portal. Longer response times for these requests are expected, as they involve authentication protocols and data verification. The lower, faster group of lines corresponds to the requests made to the web dashboard portal itself. The consistently low response times for these operations are a testament to the dashboard's excellent efficiency in handling internal navigation and data visualisation, ensuring a smooth and responsive experience for authenticated users.

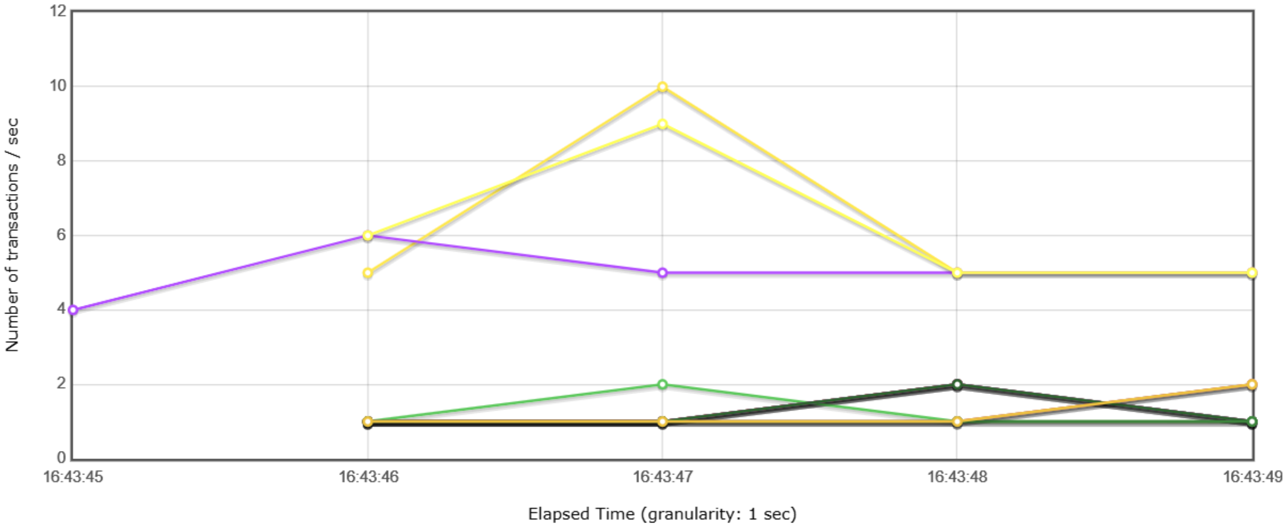


Figure 7 - Transactions per second

The Throughput graph in Figure 7 shows the transactions per second, a crucial metric for evaluating the system's capacity and overall efficiency under simulated load. This chart provides a clear visual representation of how different parts of the system handle concurrent requests.

As pointed out, the upper lines in the graph correspond to the login and landing page processes. The throughput for these initial requests shows a significant early peak, reaching up to 10 transactions per second. This initial burst is a typical and positive sign, indicating that the system's authentication and initial page-loading components are highly efficient at the beginning of the test. The subsequent drop to a more consistent throughput of around five transactions per second is expected as the test transitions from the initial login phase to the more complex navigation steps.

The lower, flatter lines in the graph represent the throughput for the subsequent actions within the web dashboard itself. Actions, such as viewing situation awareness layers and details, have a lower throughput, stabilising at around 1-2 transactions per second. This is a common and logical outcome, as these requests are often more complex and data-intensive compared to the simple login and landing page calls. The consistent throughput indicates that the system reliably processes these more demanding requests without degradation.

This graph demonstrates that the system is robust. It can handle high volumes of initial requests efficiently and then maintains a stable, albeit lower, throughput for more complex data-centric operations. This performance pattern confirms that the architecture is well-balanced and can effectively manage the different types of workloads presented by the user's navigation flow.

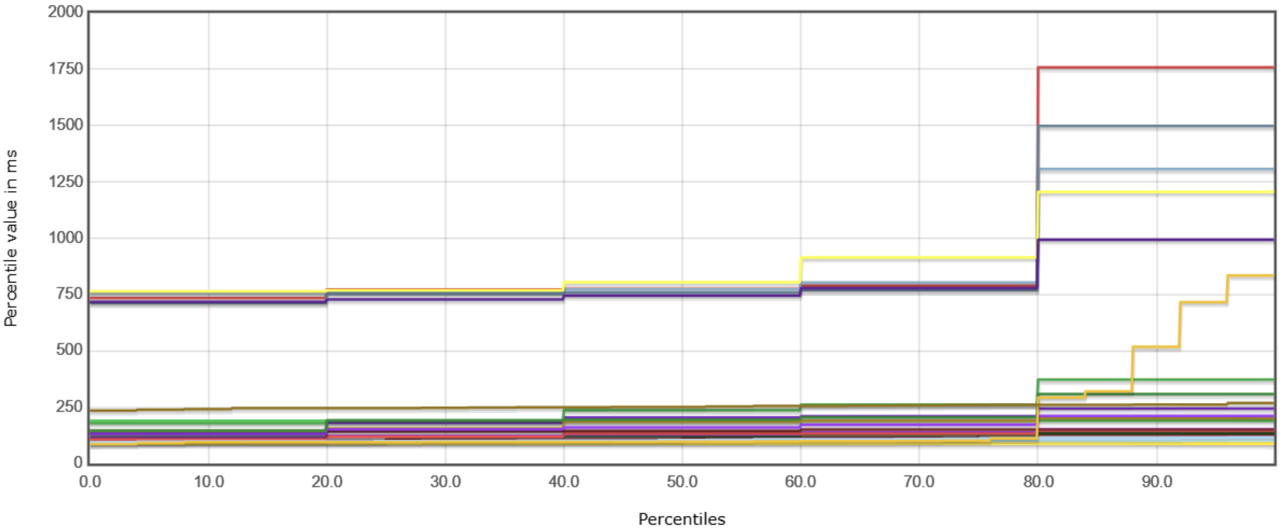


Figure 8 - Response Time Percentiles

The Response Time Percentiles graph, shown in Figure 8, is one of the most powerful tools for understanding the actual user experience, as it illustrates how consistently the system performs across its user base. Rather than just an average, this chart reveals the distribution of response times.

The graph is designed to be read from left to right. The horizontal axis represents the percentage of requests, while the vertical axis shows the response time in milliseconds. Each colored line represents a different request type from the test.

For the overwhelming majority of requests (up to the 80th percentile), the response times are very low and consistent, remaining well below the 300 ms mark. This indicates that most users experienced a swift and fluid navigation, with no noticeable delays. This is a clear indicator of strong performance and a highly responsive system.

However, the graph also reveals some "tail-end" latency. After the 80th percentile, there is a notable rise in response times. This means that a small percentage of requests, the slowest 20%, took significantly longer to complete. For a few requests (the yellow, red, and grey lines), the response times jump dramatically, reaching up to 1750 ms at the 99th percentile. This confirms the data from the initial statistics table, showing that while most users had an excellent experience, a tiny fraction encountered a substantial delay.

This chart visually confirms your system’s performance is solid for the average user, but it also highlights the need to investigate the outliers to ensure consistent performance for everyone.

In today's digital landscape, ensuring that web applications and APIs are robust and performant is crucial. System responsiveness has a direct impact on user experience and business success. To ensure that a service can handle high volumes of traffic without performance degradation, load testing is a crucial practice.

This document uses loader.io, a cloud-based load testing service, to simulate user traffic and measure system behaviour under stress. By generating thousands of simultaneous connections, loader.io provides valuable insights into an application's scalability and stability, identifying potential bottlenecks before they affect real-world users. The following sections analyse the results of a load test conducted using this tool, demonstrating the application's ability to maintain consistent and responsive performance.

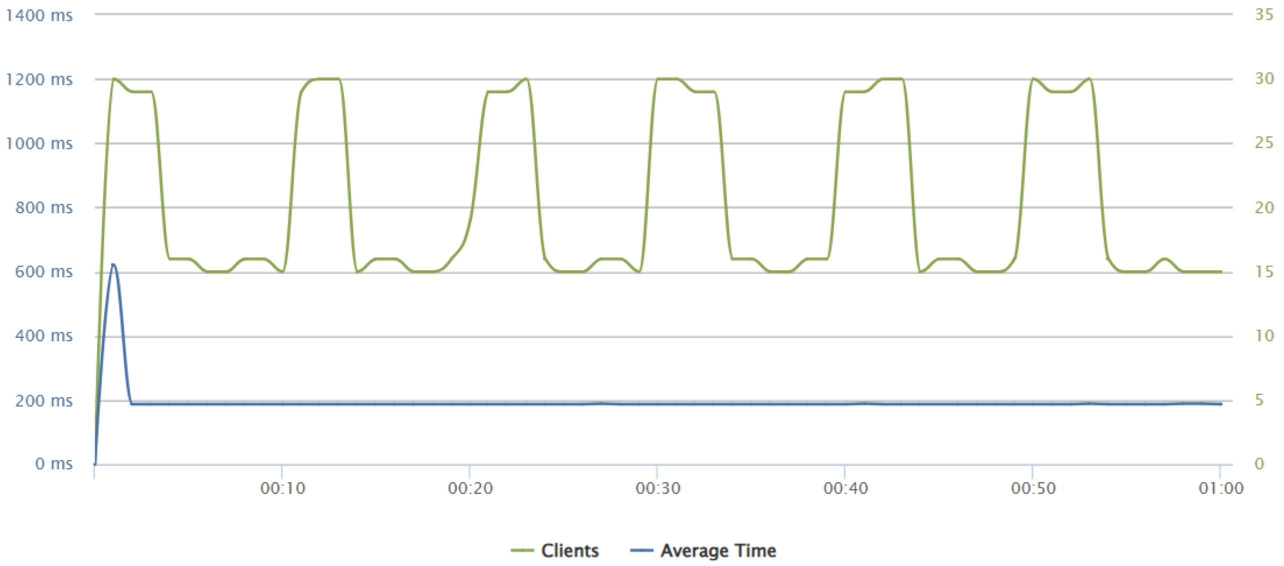


Figure 9 - loader.io stress test results

This graph displays the results of a load test, showing the relationship between the number of virtual users and the average response time of the system.

The green line, labelled "Clients," represents the number of concurrent virtual users accessing the application. As the graph shows, the test was executed in a series of repeated load cycles. In each cycle, the number of clients gradually increases to a peak of approximately 30 and then decreases.

The blue line, labelled "Average Time," indicates the average response time in milliseconds (ms) as the load changes. The average response time remains remarkably stable, hovering around 200 ms, even during the peak load periods when the number of clients is at its highest.

The test results demonstrate that the system exhibits excellent responsiveness under the simulated load. The average response time remains consistent and low, indicating that the application is well-optimised and scalable, successfully handling the increase in concurrent users without any significant degradation in performance.

3.4. User Usability Feedback outcome

3.4.1. Evaluation of the Platform's User Experience

To ensure that the OVERWATCH platform was truly usable, intuitive, and effective for end users, a process for collecting and analysing User Experience (UX) feedback was planned. The goal was to evaluate the perceived quality of the interface, identify potential issues, and implement targeted improvements while monitoring their effectiveness over time.

Figure 10 - Planned Activities and correlation with User Experience Questionnaire

The UX evaluation process was structured into several phases:

1. **Demonstration sessions**
   Guided sessions were organised for both the Flooding Pilot and the WildFire Pilot, where a sample of users had the opportunity to interact with the platform. During these sessions, users carried out realistic tasks related to the prevention and management of floods and wildfires, simulating the platform's use in operational contexts.
   To ensure the objective comparability of the two test phases, a standardised sequence of operations has been established to be carried out until the results are obtained.
   The sequence of operations is outlined as follows:
   - Login
   - Creation of a Situation Awareness
     - Select "Fire" or "Flood" (this will impact the types of services)
   - Request for Layers
     - Drawing an area within or close to the interested area
     - Select a past time window (1 week, for instance, for EO services)
   - Checking the status of the requests and visualisation of the EO services outcome: the WMS layers
   - Drone Deployment
     - Drawing an area to scan or defining the point to scan
2. **Feedback collection**
   At the end of each session, users were asked to complete an online evaluation questionnaire. Google Forms was used to distribute the survey and collect responses.
   The following table shows the number of users who participated:

Table 39 - Number of participants per session and scenario

Scenario	Wildfire	Flooding
First round	11	11
Second round	10	3

3. **Analysis tool**
   The data collected was analysed using the UEQ (User Experience Questionnaire[2]), which evaluates UX across six main dimensions:
   - Attractiveness
   - Perspicuity (clarity, comprehensibility)
   - Efficiency
   - Dependability (reliability)
   - Stimulation (engagement, motivation)
   - Novelty (innovation, originality)
4. **Evaluation cycles**

   The questionnaire was administered twice, two months apart:

   - First test: conducted after the initial release of the platform.
   - Second test: repeated after the introduction of improvements derived from the first feedback cycle.

This approach enabled the assessment not only of the initial state of the interface but also of the effectiveness of corrective actions.

3.4.2. UEQ Results

Data analysis revealed a clear improvement in users' perception between the first and second evaluation cycles. In particular:

- Perspicuity and Efficiency showed significant growth, highlighting increased ease of use and faster interaction with the platform.
- Stimulation and Novelty also recorded higher values, indicating greater user satisfaction and engagement.
- Dependability remained stable but positive, confirming users' trust in the platform's functional consistency.
- Overall, attractiveness benefited from the improvements, with a noticeably higher average rating.

---

[2] https://www.ueq-online.org

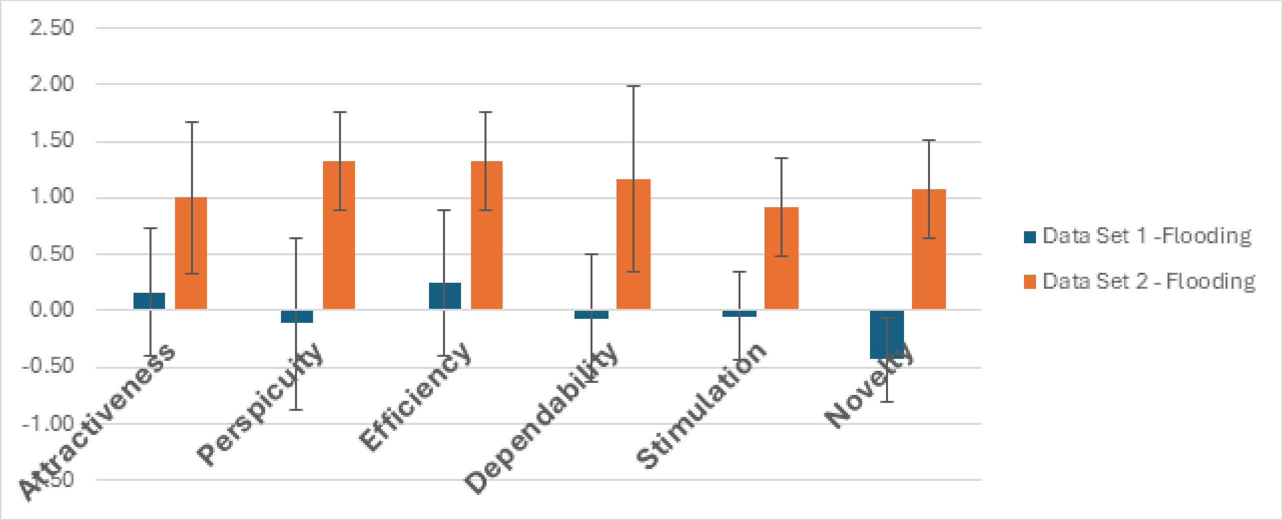


Figure 11 - Comparison of UEQ scores between the first and second test for the flooding scenario

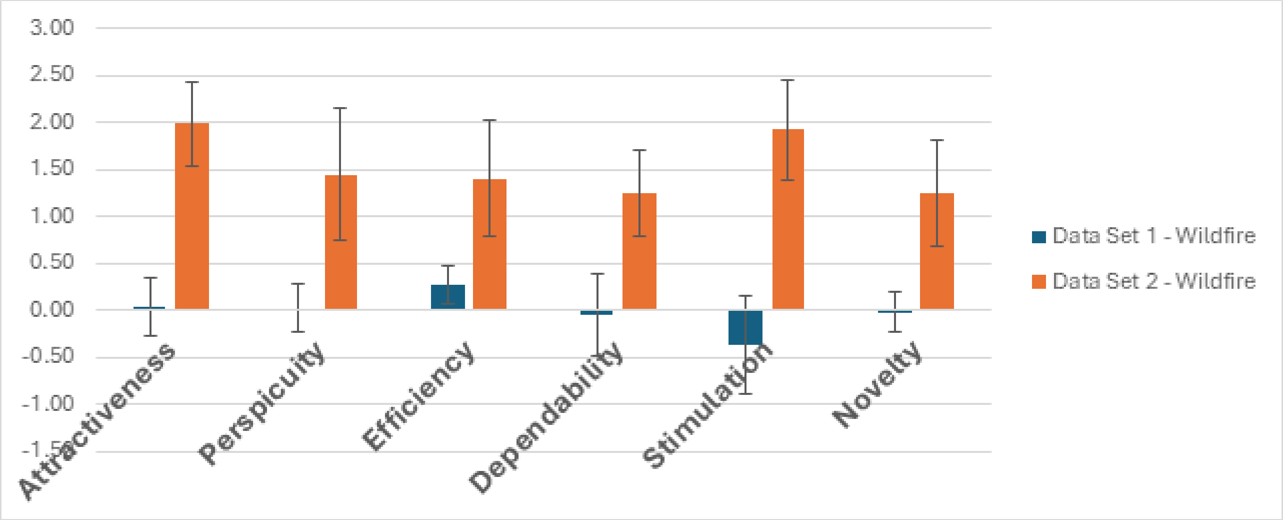


Figure 122 – Comparison of UEQ scores between the first and second tests for the wildfire scenario

3.4.3. Final considerations

The iterative approach, based on feedback collection through demonstration sessions and UEQ questionnaires, proved effective: the changes made to the platform resulted in a tangible improvement in user experience.

The sharp increase in scores between the first and second tests confirms the importance of adopting user-centred design methodologies and actively involving end-users throughout the entire development cycle of tools aimed at managing complex scenarios, such as wildfires and floods.

3.5. Highlighting on the Progress Beyond the State-of-the-Art

The main progress beyond the state-of-the-art (SOTA) of OVERWATCH lies in the design and development of a holistic and fully integrated platform. Unlike the pre-existing SOTA, which was characterised by fragmented or siloed solutions for individual aspects (e.g., data processing, data ingestion, orthorectification, mission planning, C2), OVERWATCH unifies these functionalities into a single, end-to-end workflow. This workflow manages the entire data lifecycle, from generation (via satellite imagery and high-resolution video feeds from drones) to processing and user presentation (via a Web Dashboard and AR headsets). Overcoming this fragmentation enables a previously unattainable operational synergy, aimed at advanced and unified decision support.

A second, and crucial, advancement is the platform's validation in realistic operational scenarios (fire and flood management), conducted in close collaboration with strategic end-users (e.g., Civil Protection agencies). While the SOTA in this sector is often confined to laboratory demonstrations (low TRLs - Technology Readiness Levels), OVERWATCH has proven its effectiveness, robustness, and usability in the field. The direct feedback from operators, who actively used the 2D/3D situational awareness and mission management features, moves the SOTA from a "proof-of-concept" to "operational validation".

Furthermore, OVERWATCH has innovated the data value chain in disaster management. By implementing edge processing algorithms directly on board the assets (drones), the platform overcomes the SOTA limitations related to the latency and network overhead of purely centralised processing systems. Colored and thermal images are transformed into exploitable information almost instantaneously. This enables near real-time situational awareness for the command centre, a decisive capability for intervention in critical scenarios.

Finally, the platform demonstrated its full capability to manage and control complex assets (such as drones) through the integration of a high-fidelity simulation environment. Far from being a limitation, this approach represents a methodological advancement. It enabled the validation of mission management and data processing logic in complex and high-risk scenarios (such as flooding/fire) in a safe, controlled, and repeatable manner. This was a fundamental and necessary step to de-risk the subsequent full operational deployment with tangible assets.

3.6. Feedback from Pilots (gathered during D4.3 / T4.4)

The feedback summarised below was provided directly by the commanders and firefighters who participated in the pilot test phase. Their observations reflect operational experience during real-world testing, ensuring that the recommendations are grounded in practical use and aligned with mission requirements.

3.6.1. AR Feedback

Operational systems in dynamic environments demand interfaces that are intuitive, responsive, and visually clear. However, the assessment by end users revealed several areas for improvement, particularly in usability, performance, visualisation, and feature integration.

3.6.1.1. Interface and Usability

Navigation was identified as a critical pain point. Users reported that accessing essential functions required too many steps, which can hinder decision-making during time-sensitive operations. In high-pressure contexts, every second counts, and complex menu structures introduce unnecessary delays. To address this, a simplified menu design and more direct command pathways are recommended. These adjustments should be tailored to the operational context, ensuring that frequently used functions are immediately accessible. The expected benefit is clear: faster decision-making and improved efficiency in dynamic scenarios where rapid responses are vital.

3.6.2. Performance

Another challenge lies in system performance, particularly in environments with limited connectivity where users reported delays in updating real-time information. The defined latency threshold for fluid AR rendering and data synchronization is 50 ms (RTT), as outlined in the technical requirements (Section 2.1.2). Validation tests utilizing the dedicated Fallback Connectivity System demonstrated robust performance, achieving an average round-trip time of 29 ms (max 38 ms). Mitigation Strategy: The reported delays occurred in scenarios exceeding these optimal conditions. To address this, the additional technical day requested will be specifically dedicated to stress-testing the synchronization protocols under degraded network conditions (simulating bandwidths below the 40 Mbps requirement). This session will focus on fine-tuning the data buffering and retry mechanisms to ensure that information remains reliable even when connectivity fluctuates beyond the ideal thresholds.

3.6.2.1. Visualisation

Visual clarity is fundamental for situational awareness, yet overlapping graphic elements have made interpretation difficult in complex operational displays. This issue not only slows comprehension but also contributes to user fatigue over extended periods. A dynamic approach to visualisation is recommended, incorporating automatic adjustments to opacity, contrast, and element positioning based on context. Such adaptive visualisation techniques will declutter the interface, allowing critical information to stand out without overwhelming the user. The anticipated outcome is improved readability, reduced cognitive load, and sustained operator performance during prolonged missions.

3.6.2.2. Additional Features

End users have requested the inclusion of visual indicators for force status, such as “ready,” “moving,” or “unavailable”, alongside critical alerts. These features would provide commanders and operators with immediate insight into unit readiness and operational posture. By integrating these indicators into the main display, the system can deliver greater situational awareness and enable faster responses to emerging events. This enhancement aligns with the overarching goal of supporting informed, timely decisions in complex operational environments.

3.6.2.3. Synthesis

Addressing these areas, interface simplification, performance optimisation, visualisation refinement, and feature expansion, will significantly improve system usability and operational effectiveness. The combined impact of these measures is a platform that not only meets technical requirements but also empowers users to act decisively in dynamic, high-stakes scenarios.

3.6.3. Web Dashboard Feedback

Modern operational platforms must strike a balance between functionality and clarity, ensuring that users can access critical information quickly and efficiently. Feedback from the pilot phase highlighted several areas for improvement, particularly in layout organisation, filtering capabilities, system integration, and alert management.

3.6.3.1. Layout and Organisation

While the interface was considered functional, users noted that certain areas displayed excessive information, making it difficult to prioritise what truly matters during high-pressure situations. Overloaded screens can lead to cognitive fatigue and slower decision-making. To address this, a reorganisation strategy based on levels of criticality is recommended. Grouping information by operational importance and implementing modular dashboards will allow users to customise their view according to mission needs. This approach ensures that critical events remain visible while secondary data is accessible but not overwhelming. The expected benefit is improved information management and a sharper focus on priority actions.

3.6.3.2. Filters and Search

Efficient navigation within complex systems requires robust filtering and search capabilities. During the pilot, commanders emphasised the need for advanced filters to locate units, resources, or specific events quickly. The proposed solution involves implementing an innovative search engine that utilises multiple criteria, including location, status, and resource type. This functionality will significantly reduce the time spent searching for operational elements, enabling faster responses and better resource allocation. Ultimately, this enhancement translates into greater operational efficiency and improved mission execution.

3.6.3.3. Integration

Interoperability remains a cornerstone of modern defence systems. Participants requested seamless integration with other platforms, particularly the ability to export reports in standard formats such as PDF and CSV. This capability would simplify information sharing across different command levels and allied systems, fostering collaboration and reducing administrative burden. By enabling standardised data exchange, the platform will support joint operations and ensure that decision-makers have access to consistent, reliable information. The expected benefit is enhanced interoperability and streamlined reporting processes.

3.6.3.4. Alerts and Notifications

Timely alerts are essential for maintaining situational awareness. The pilot revealed a need for configurable alerts that allow users to define thresholds and priorities based on operational context. Additionally, an accessible event history should be included to support traceability and post-action analysis. These features will enable proactive monitoring, ensuring that critical developments are never overlooked. By improving alert customisation and historical tracking, the system will provide better oversight and accountability, reinforcing operational effectiveness.

3.6.3.5. Synthesis

Implementing these improvements, criticality-based layout, advanced search tools, system integration, and configurable alerts, will transform the platform into a more intuitive, interoperable, and mission-focused solution. These changes directly address feedback from commanders who participated in the pilot, ensuring that the system evolves in line with real operational needs and enhances decision-making in complex environments.

3.6.4. Fine-tuning actions for feedback

Enhancing AR and Web Dashboard Capabilities for Operational Efficiency

Operational platforms must deliver clarity, speed, and adaptability across different interfaces. Feedback from the pilot phase highlighted key improvements for both Augmented Reality (AR) applications and Web Dashboards, focusing on usability, performance, visualisation, and interoperability.

3.6.4.1. For AR Applications

AR systems play a critical role in providing real-time situational awareness, but their effectiveness depends on intuitive design and reliable performance. Users noted that the current menus were too long and required multiple steps to execute essential commands. To address this, interface simplification is recommended, with shorter menus and contextual commands that adapt to the operational scenario. This will reduce cognitive load and accelerate decision-making in dynamic environments.

Performance optimisation is another priority. In areas with limited connectivity, delays in data synchronisation can compromise mission success. Enhancing offline data management and improving synchronisation protocols will ensure that operators have access to accurate information even under constrained conditions. These changes will reduce latency and strengthen operational reliability.

Visualisation also requires refinement. Overlapping elements and static displays can hinder clarity in complex environments. Implementing dynamic visual adjustments, such as adaptive contrast and opacity, will enable the interface to respond to varying lighting conditions and operational contexts, thereby improving readability and reducing user fatigue. Finally, the inclusion of visual indicators for critical states and urgent events will provide immediate awareness of force readiness and emerging threats, enabling faster and more informed responses.

3.6.4.2. For Web Dashboards

The web-based command interface serves as the backbone for operational coordination, but its current layout can overwhelm users with excessive information. A reorganisation strategy using modular dashboards is recommended, prioritising critical events while allowing secondary data to remain accessible. This approach will enhance information management and enable commanders to maintain focus on high-priority actions.

Advanced filtering and search capabilities are essential for efficient navigation. Implementing smart search functions with multiple criteria, such as location, status, and resource type, will enable users to locate units, assets, or specific events quickly. This enhancement will streamline operational management and reduce time spent on manual searches.

Integration with other systems is another critical requirement. The ability to export reports in standard formats, such as PDF and CSV, will facilitate interoperability and simplify information sharing across different command levels and allied platforms. This feature will support joint operations and ensure consistent reporting standards across all operations.

Finally, configurable alerts and notifications will enhance monitoring and traceability. Allowing users to personalise thresholds and priorities, combined with an accessible event history, will improve oversight and accountability. These features will ensure that critical developments are never missed and that post-action analysis is fully supported.

3.6.4.3. Synthesis

By implementing these improvements, the platform will evolve into a more intuitive, interoperable, and mission-focused solution, featuring simplified AR interfaces, optimised performance, adaptive visualisation, modular dashboards, advanced search tools, system integration, and configurable alerts. These recommendations reflect feedback from commanders who participated in the pilot, ensuring alignment with real operational needs.

3.7. Known Limitations

The iterative integration and validation phases conducted in WP4 have highlighted specific technical and operational constraints. Identifying these limitations is crucial for defining the operational boundaries of the OVERWATCH system and ensuring the safety and reliability of the deployed assets during the pilots and future real-world scenarios.

The following subsections detail the identified constraints, their operational thresholds, and the mitigation strategies adopted.

3.7.1. Earth Observation and Early Warning

The current Earth Observation solutions are purely observational: they are producing thematic layers, based on the acquisitions of Sentinel satellites, after an event has happeded. Most polar orbiting satellites, including Sentinel-1 and Sentinel-2, which represent the backbone of our system, do not have revisit times short enough to provide detection or early warning capabitilies. This limits what can be done with EO using these sensors to post-event (in-event at most) monitoring (see EO-7). Active fire detection or warning systems could be implemented by switching to other satellite constellations (e.g., EUMETSAT MTG), or by implementing forecast solutions.

3.7.2. Drone Operations and Environmental Constraints

The current drone integrated is subject to strict environmental limitations. Due to payload constraints required to carry the specific sensor suite (RGB and thermal cameras, onboard edge computing modules), the drones cannot be fully weather-sealed. Consequently, operations are severely limited by adverse weather conditions, specifically heavy rain and high wind speeds. Additionally, reliance on GNSS for navigation makes the system susceptible to degradation during electromagnetic storms or in environments with high electromagnetic interference (EMI).

The operational thresholds are:

- Zero tolerance to precipitation, flights must be grounded in the event of rain to prevent electrical short-circuits and sensor damage.

- Wind speed limitation, because operations are restricted to wind speeds below **8 m/s** (approx. 28 km/h), as validated during the testing phase (Table 31 - Performance requirements met for drone terrain mapping).

The primary mitigation strategy relies on rigorous mission planning and meteorological monitoring. Before any deployment, a "Go/No-Go" protocol is enforced based on real-time weather forecasts at the theatre of operations. In the event of sudden environmental changes or GNSS loss, the drone failsafe protocols (Return-to-Home or safe landing) are automatically triggered.

3.7.3. Augmented Reality (AR) Latency and Connectivity

The Augmented Reality module relies on remote rendering to display complex geospatial layers and high-fidelity 3D maps on the headset. High network latency can lead to a misalignment between the digital overlay and the physical world (registration errors), potentially causing motion sickness for the operator and reducing situational awareness.

The operational threshold is:

- A latency between AR frontend and backend under 50ms threshold to ensure a fluid user experience and stable holographic projections.

To mitigate network instability inherent in public infrastructures during crisis scenarios, the system architecture mandates the deployment of a dedicated ad-hoc field network. During the pilot tests, the use of a local, high-bandwidth communication bubble (utilizing a dedicated routers) successfully guaranteed the required throughput and low latency, validating the system's performance even in areas with poor external internet coverage.

3.7.4. Thermal Sensor Calibration

The fine-tuning of the thermal imagery processing algorithms for wildfire detection encountered logistical constraints regarding sensor calibration. Obtaining ground-truth data requires performing controlled burns (prescribed fires).

There is a conflict between the operational wildfire season (when the system is tested) and safety regulations, which strictly prohibit starting controlled fires during high-risk periods to prevent accidental spread. This prevented the simultaneous execution of thermal sensor calibration and live pilot testing in the target environment.

Current validation relies on manufacturer specifications and laboratory-simulated data. The precise fine-tuning and calibration of thermal thresholding algorithms have been postponed to a window outside the critical fire season, where controlled burns can be performed safely under the supervision of civil protection authorities.

Conclusion

Conclusions will be discussed in the final version of the present deliverable, due by M42 and corresponding to the end of the project, when also the Flood pilot demonstration will be held and more inputs collected and integrated.

Appendix A: Mapping of Validation Activities to D4.1 Methodology

During the rehearsal and demonstration phases, a consolidated script comprising 7 Operational Walkthrough Steps was defined and executed. These steps were instrumental in verifying the end-to-end basic flows of the Use Cases and Scenarios originally defined in WP1, ensuring that the system supports the complete data lifecycle from acquisition to visualization in a realistic crisis management environment.

Furthermore, this operational workflow serves as the foundation for the specific validation rules used to confirm the compliance status reported in the Test Matrix Results of D4.1. The following table illustrates the direct mapping between these operational steps and the formal verification requirements (VRFs), demonstrating how the practical execution of the pilot scenarios fulfills the methodological compliance criteria and validates the integration results presented in this document.

Operational Walkthrough Step	Detailed Protocol Description	D4.1 Reference (VRF ID)	Acceptance Criteria	Supporting Evidence
1. Creation of a Situation Awareness	User logs into the Web Dashboard, defines the crisis perimeter, and initializes a new operational scenario.	**VRF-1** (Request Overwatch mapping)		Screenshot
2. Request Mapping Services	User selects the area of interest and specific EO services (e.g., Flood/Wildfire delineation) via the Dashboard form.	**VRF-1** (Request Overwatch mapping)		Screenshot
3. Ingestion of the EO Service outcome	System processes the request; the output is automatically uploaded to the Geospatial Repository and converted into WMS layers.	**VRF-3** (Visualize EO result onto repository)		Screenshot
4. Visualization of the WMS layers	User verifies that the generated layers are correctly displayed on both the Web Dashboard and AR devices.	**VRF-4** (Dashboard visualization) **VRF-5** (AR visualization)		Screenshot

5. Drone Deployment	Drone takes off to perform mapping or navigation tasks, specifically testing accuracy against Ground Control Points (GCP) and avoiding hazard zones.	**VRF-6** (Mapping Accuracy) **VRF-7** (Navigation/Obstacle avoidance)		Logs + Screenshots
6. In-field communication (Fallback)	Drone and ground station connectivity is maintained via the Fallback Connectivity module (Wi-Fi/LEO backhaul) during the flight.	**VRF-11** (Fallback Internet connectivity)		Logs
7. Visualization of Drone flight result	Real-time or quasi-real-time visualization of the drone-acquired data (AI mapping) on the platform interfaces.	**VRF-9** (Quasi Real-time visualization)		Logs + Screenshots

Supporting evidence will be added in the second issue of the deliverable.