Smart, Automated, and Reliable Security Service Platform for 6G

Deliverable D3.4

ROBUST-6G Final Report on Trustworthy and Sustainable AI Architecture and Requirements for Integrating Selected XAI measures

ROBUST-6G project has received funding from the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation programme under Grant Agreement No 101139068.

Date of delivery:		Version:	1.0
Project reference:	101139068	Call:	HORIZON-JU-SNS-2023
Start date of project:	01/01/2024	Duration:	30 months

Document properties:

Document Number:	D3.4
Document Title:	ROBUST-6G Final Report on Trustworthy and Sustainable AI Architecture and Requirements for Integrating Selected XAI measures
Editor(s):	Ioannis Pitsiorlas (EUR), Marios Kountouris (EUR), Bartlomiej Siniarski (UCD), Giovanni Perin (UNIPD), Manuel Gil Pérez (UMU)
Authors:	Ioannis Pitsiorlas (EUR), Marios Kountouris (EUR), Giovanni Perin, Michele Rossi (UNIPD), Manuel Gil Pérez (UMU), Enrique Tomás Martínez Beltrán (UMU), Fernando Torres Vega (UMU), Betul Guvenc Paltun (EBY), Eunjeong Jeong (LIU), Farah Abed Zadeh (UCD), Leyli Karaçay (EBY), Bartlomiej Siniarski (UCD)
Contractual Date of Delivery:	31/03/2026
Dissemination level:	PU[1]
Status:	Final
Version:	1.0
File Name:	ROBUST-6G D3.4_v1.0

Revision History

Revision	Date	Issued by	Description
0.1	10.01.2026	ROBUST-6G WP3	Initial draft version created. Basic structure and main objectives of D3.4 defined.
0.2	05.02.2026	ROBUST-6G WP3	Updated draft with extended content, added preliminary methodology and refined scope.
0.3	10.03.2026	ROBUST-6G WP3	Revised draft incorporating feedback, improved clarity, and expanded technical details.
0.4	27.03.2026	ROBUST-6G-WP3	Final version completed. Content reviewed, finalized, and approved for submission.
1.0	29.03.2026	ROBUST-6G-WP3	Final version reviewed by internal and external reviewers.

Abstract

Keywords

AI/ML prototype, Decentralized learning, Trustworthiness, Sustainability, Explainability, Services integration.

[1] SEN = Sensitive, only members of the consortium (including the Commission Services). Limited under the conditions of the Grant Agreement

PU = Public

Disclaimer

Executive Summary

This deliverable presents the final outcomes of the WP3 technical work on the ROBUST-6G trustworthy, sustainable, and explainable AI framework. Building upon the foundations established in earlier WP3 deliverables, particularly the intermediate architecture, requirements, and prototype developments reported in D3.2 and D3.3, D3.4 consolidates the final design of the proposed prototype. It further evaluates its readiness, feasibility, and resilience for integration into future 6G environments.

The deliverable moves beyond isolated component development and provides a coherent, integrated, and deployment-oriented view of the ROBUST-6G AI framework. In the final architecture, the **Decentralized Federated Learning (DFL) framework** is elevated from an optional training mechanism to the core operational backbone of the system, while trustworthiness, sustainability, and explainability are embedded directly into the learning lifecycle. In parallel, the Global Model Repository (GMR) assumes a strengthened role as a shared AI asset management and governance layer, supporting traceability, reproducibility, accessibility, and interoperability of WP3 outputs.

From a technical perspective, D3.4 reports the final integrated architecture of the WP3 prototype and its alignment with key 6G directions, including AI-native operation, distributed intelligence across the edge-cloud continuum, trustworthiness by design, sustainability as a core objective, and programmability through modular building blocks. It also presents the practical realization of the DFL framework, including deployment configuration, operational workflows, security mechanisms, and validation evidence.

The deliverable further consolidates the final state of the core AI modules developed in WP3. The trustworthy AI module addresses privacy-preserving and attack-resilient learning mechanisms, the sustainable AI module contributes scalable and energy-aware solutions for decentralized learning, and the explainable AI module provides XAI-based mechanisms for model enhancement, uncertainty estimation, incident reporting, interpretability, and accountability assessment. Together, these components strengthen the transparency, robustness, and operational viability of AI-enabled security processes in future 6G systems.

Finally, D3.4 addresses the integration and assessment perspective of the final prototype. It includes both an internal AI-pipeline usage example and an external integration view for selected XAI measures within the ROBUST-6G architecture, and it reports the fulfilment of the objectives defined in the DoA together with the corresponding KPI validation results. Overall, the deliverable serves as the final WP3 integration and assessment report, while also deriving lessons learned and recommendations for the feasible integration of selected XAI measures into future trustworthy AI-native 6G systems.

Table of Contents

List of Figures

List of Tables

Acronyms and abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
CPCF	Conformal Prediction Confidence Factor
CPU	Central Processing Unit
DFL	Decentralized Federated Learning
FL	Federated Learning
GMR	Global Model Repository
GPU	Graphics Processing Unit
GUI	Graphical User Interface
IDS	Intrusion Detection Systems
IID	Independent, Identically Distributed
ML	Machine Learning
NET	Network Layer
TLS	Transport Layer Security
SHAP	Shapley Additive Explanations
SHIELD	Selective Hidden Input Evaluation for Learning Dynamics
SNN	Spiking Neural Networks
VAE	Variational Autoencoder
VAoI	Version Age of Information
WP	Work Package
XAI	Explainable Artificial Intelligence
ZSM	Zero-touch Security Management layer

1 Introduction

1.1 Motivation, objectives and scope

Deliverable D3.4 - Final Report on ROBUST-6G Trustworthy and Sustainable AI Architecture and Requirements for Integrating Selected XAI Measures constitutes the concluding outcome of the WP3 technical work on the design, integration, prototyping, and final assessment of ROBUST-6G AI-native security enablers. Building on the foundations established and the results reported in earlier WP3 deliverables, and in particular on the intermediate architecture, requirements, and prototype developments of D3.2 and D3.3, this deliverable consolidates the final design of the proposed prototype and evaluates its readiness, feasibility, and resilience for integration into future 6G environments.

The motivation for this deliverable arises from the need to move beyond isolated component development and provide a coherent, integrated, and deployment-oriented view of the ROBUST-6G trustworthy, sustainable, and explainable AI framework. In the context of 6G, AI-driven security solutions must not only achieve strong technical performance but also demonstrate that they can be integrated into distributed and AI-native infrastructures in a way that is trustworthy, sustainable, explainable, and operationally viable. While Explainable AI (XAI) mechanisms can significantly enhance transparency, accountability, and confidence in AI-enabled security decisions, their integration may also affect the accuracy, latency, computational complexity, and energy footprint of the underlying AI processes. For this reason, a final assessment is necessary to determine whether and how the selected XAI models, measures, and modules support the ROBUST-6G architectural vision of trustworthy and sustainable AI-native security without introducing prohibitive costs or reducing operational effectiveness.

Accordingly, D3.4 has four main objectives. First, it documents the evolution and final integrated design of the ROBUST-6G prototype, including its architectural principles, its alignment with future 6G systems, and its positioning within the broader 6G ecosystem. Second, it presents the DFL framework and the associated operational workflows, deployment configurations, and validation evidence that support the practical realization of the prototype. Third, it consolidates the final state of the core AI modules developed in WP3, namely the trustworthy AI, sustainable AI, and explainable/fair AI components. Fourth, it provides an analysis of the feasibility and resilience of the proposed and prototyped security measures, with particular focus on their impact on AI-process accuracy, trustworthiness, and the sustainability of 6G through the implemented XAI means.

A further objective of this deliverable is to derive a set of requirements for conducting XAI integration feasibility, based on the knowledge gained from the architecture design, prototype implementation, and validation activities. In this sense, D3.4 not only reports the final prototype status, but also extracts lessons learned and translates them into requirements and guidance for future integration of XAI mechanisms in trustworthy AI-native 6G systems.

The scope of this deliverable therefore includes:

- the final integrated architecture of the ROBUST-6G trustworthy, sustainable, and explainable AI prototype
- the analysis of its alignment with key 6G architectural directions, including AI-native operation, distributed intelligence, edge-cloud continuity, trustworthiness, sustainability, and programmability
- the presentation of the **DFL** framework, including deployment, workflows, and validation evidence
- the consolidation of the core AI modules for trustworthy, sustainable, explainable, and fair AI
- the assessment of the feasibility, resilience, and operational readiness of the proposed security measures
- the definition of requirements and recommendations for the feasible integration of selected XAI measures into future 6G security architectures.

In summary, D3.4 serves as the final WP3 integration and assessment report. It brings together the architecture, prototype realization, module-level contributions, and deployment-oriented analysis to provide a consolidated view of how trustworthy, sustainable, explainable, and fair AI mechanisms can be realistically integrated into 6G security infrastructures while maintaining an appropriate balance between accuracy, transparency, resilience, and sustainability.

1.2 Document structure

Section 2 of this deliverable reports the evolution and final design of the ROBUST-6G Trustworthy, Sustainable and Explainable AI Prototype. In particular, it discusses the architectural consolidation of the prototype, its final integrated design, its alignment with emerging 6G architectural directions, the DFLframework and its validation evidence, as well as the final state of the core AI modules for trustworthy, sustainable, and explainable AI. **Section 3** addresses the use case and integration aspects, including both an internal AI-pipeline usage example and the external integration of selected XAI measures into the ROBUST-6G architecture. **Section 4** presents the fulfilment of the DoA objectives together with the corresponding KPI validation. Conclusions and perspectives are finally drawn in **Section 5**.

2 Evolution and Final Design of the Trustworthy, Sustainable and Explainable AI Prototype

2.1 Introduction

This section presents the architectural evolution and final consolidation of the WP3 “Trustworthy, Sustainable and Explainable AI Prototype”. The final version of the prototype represents the culmination of the work conducted throughout WP3, integrating threat-driven design principles, modular technical developments, and system-level architectural refinement into a coherent, AI-native framework tailored for future 6G networks. In the context of this deliverable, it is important to distinguish between the architecture and the prototype. The architecture refers to the logical design of the system, describing the structural components, their functional roles, and the interactions between them. The prototype, in contrast, refers to the concrete software implementation that instantiates this architecture. It consists of the deployed components, containerized services, a decentralised training framework, and supporting interfaces that realize the architectural design in practice.

The development trajectory followed a structured progression. Deliverable D3.1 established the threat landscape and security requirements for AI-enabled 6G systems. Deliverable D3.2 translated these requirements into concrete technical mechanisms addressing trustworthiness, sustainability, and explainability. Deliverable D3.3 presented the first integrated operational prototype. The final architecture refines and consolidates selected elements into a unified decentralized system governed through a shared model repository.

The resulting architecture is not merely an aggregation of modules, but a consolidated intelligence layer designed in alignment with the distributed, resilient, sustainable, and transparent nature envisioned for 6G infrastructures.

Threat-driven Foundations

The architectural direction of WP3 was fundamentally shaped by the threat analysis conducted in Deliverable D3.1. That analysis identified vulnerabilities inherent to AI-driven 6G systems, particularly those operating across distributed and heterogeneous environments. Among the most critical risks were adversarial model and data poisoning, privacy leakage in collaborative learning settings, manipulation of trust in distributed environments, excessive resource consumption, and the absence of explainability in automated decision-making processes.

These findings suggest that conventional centralized machine learning (ML) pipelines may be insufficient for 6G contexts. Instead, AI mechanisms for 6G are expected to evolve toward more decentralized designs that are resilient to adversarial behaviour, privacy-preserving, energy-aware, and inherently transparent. In this

light, trustworthiness, sustainability, and explainability emerge not merely as complementary features, but as essential design considerations.

D3.1 therefore established the foundational constraints that guided all subsequent architectural decisions. In particular, it emphasized that the prototype should natively incorporate these principles at the design level, rather than treating them as add-on or external services.

From components to modules

Deliverable D3.2 translated the threat-driven requirements into concrete technical contributions developed by consortium partners. Multiple mechanisms were designed to address specific dimensions of the problem space, including privacy-enhanced federated learning (FL), robust aggregation strategies, adversarial resilience techniques, energy-aware scheduling and model optimization methods, and explainability services based on feature attribution and uncertainty estimation.

At that stage, the system consisted of technically mature but largely independent building blocks. While each component addressed an important requirement, they operated as separate entities. The need for architectural coherence led to their consolidation into three thematic macro-modules: the Trustworthiness Module, the Sustainability Module, and the Explainability Module.

This conceptual consolidation marked a critical step forward. However, training workflows remained flexible and the DFL framework was not yet structurally central. The architecture was modular and functional, but not yet fully AI-native.

Initial integrated prototype

Deliverable D3.3 introduced the first integrated instantiation of the WP3 prototype. In this version, the three modules were operational and connected via defined interfaces. The DFL framework could be employed for distributed training, although its use was optional. Models trained either through DFL or through alternative pipelines were stored and versioned within the GMR, which acted as a central storage and access facility.

The prototype successfully demonstrated Application Programming Interface (API) exposure toward the zero-touch security management ZSM layer (WP4) and the Physical Network layer (WP5), enabling model discovery and retrieval on demand. Containerization ensured portability and reproducibility across environments.

From a functional standpoint, this version validated the feasibility of integrating trustworthiness, sustainability, and explainability services into a unified prototype. Nevertheless, architecturally the system still reflected a composition of modules around a training process, rather than a deeply integrated intelligence fabric. The three dimensions remained conceptually adjacent to the learning workflow rather than intrinsic to it. This observation motivated the architectural refinement that led to this final version.

2.2 Final integrated prototype

The final version as depicted in Figure 1 represents a structural consolidation of the WP3 prototype. The central design decision was to elevate the DFL framework from an optional training mechanism to the core operational backbone of the system.

In this final prototype, all model training processes are conducted within the DFL framework. Trustworthiness, sustainability, and explainability are no longer external services applied before or after training. Instead, they are modules embedded directly within the learning lifecycle. This transition reflects a fundamental architectural principle: AI mechanisms for 6G must be trustworthy, sustainable, and explainable by design.

Decentralized Federated Learning as Architectural Core

The DFL framework (discussed in fine details in Section 2.3) orchestrates collaborative training across distributed nodes without relying on a centralized aggregator. By enabling peer-to-peer coordination, it eliminates single points of failure and enhances resilience against selective attacks. Robust aggregation strategies and adversarial scenarios simulations are integrated within this framework, ensuring that resilience is assessed and enforced during training rather than retrospectively.

System-level and model-level metrics are continuously collected and persisted. The DFL framework interfaces directly with the Global Model Repository, ensuring that all intermediate and final model artifacts are

versioned and traceable. Through this structural centralization of DFL, the prototype aligns naturally with the distributed intelligence paradigm expected in 6G systems, where learning occurs across edge, core, terminals, UEs and cloud environments.

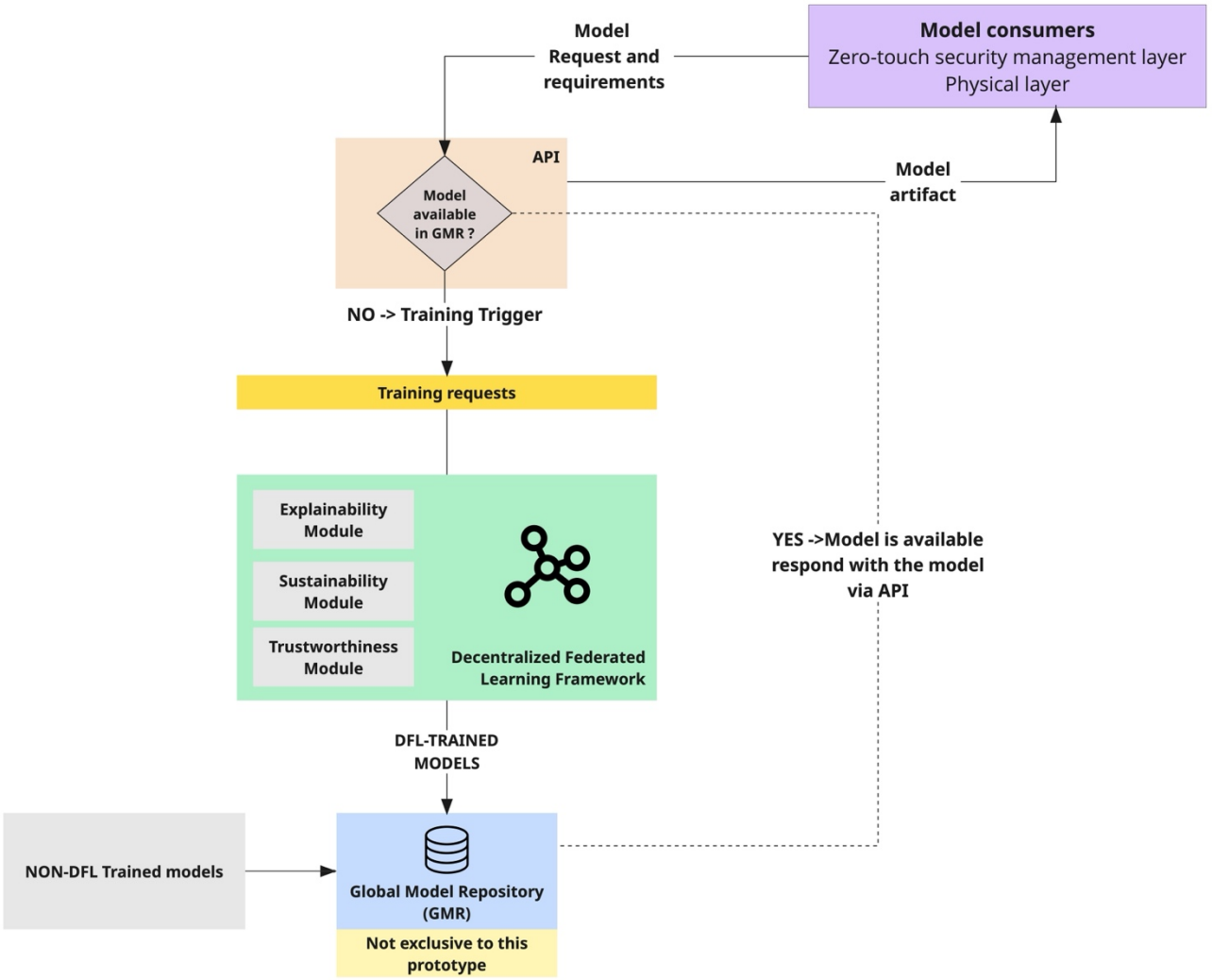

Figure 1 Final Architecture of Trustworthy, Sustainable and Explainable AI Prototype

Embedded Trustworthiness

In the final version, trustworthiness is not implemented as a peripheral validation step. Privacy-preserving mechanisms, adversarial robustness techniques, and trust evaluation metrics are integrated within the decentralized learning loop. This ensures that model resilience and data protection are inherent properties of the training process. The functionality of this module is described further in Section Trustworthy AI Module. The architecture thus directly addresses the threat vectors identified in D3.1 by embedding mitigation strategies into the operational core of the system.

Embedded Sustainability

Energy efficiency considerations are incorporated into training orchestration and aggregation decisions. Client scheduling, communication overhead reduction, and model optimization techniques such as pruning and quantization are integrated within the learning workflow. This enables adaptive trade-offs between accuracy and energy consumption, which are essential in 6G environments characterized by heterogeneous and resource-constrained nodes. Sustainability becomes a continuous optimization objective rather than a post-hoc evaluation metric. The functionality of this module is described further in Section Sustainable AI Module.

Embedded Explainability

Explainability services are integrated into the model lifecycle. Techniques such as feature attribution, uncertainty quantification, and confidence estimation generate artifacts that accompany the trained models. These artifacts are stored alongside performance and robustness metrics in the Global Model Repository. By embedding interpretability into the architecture, the prototype supports auditability, operator trust, and regulatory compliance, which are expected to play a significant role in AI governance for 6G networks. The functionality of this module is described further in Section Explainable AI Module.

Global Model Repository as Cross-Layer Governance Asset

While this prototype focuses on the integration of ZSM capabilities, it is important to highlight the role of the GMR as a shared governance asset across the broader architecture. The GMR is not confined to this prototype but operates as a common service that can be leveraged by multiple layers, including the Zero Touch Management layer, the network layer, and other components requiring centralized model storage and access.

The GMR provides a unified mechanism for storing, versioning, and annotating models produced within WP3, including metadata related to provenance, configuration, performance, robustness, and energy characteristics. This ensures traceability and reproducibility across the full model lifecycle. Through stable REST interfaces, the GMR enables different architectural components to retrieve production-ready models or raw artifacts for integration into their respective pipelines. As illustrated in Figure 2, the GMR is positioned as a central yet decoupled element, supporting interactions across layers rather than being embedded within a single functional block of the prototype. In this context, the GMR should be understood as a shared AI asset management layer, contributing to consistency, accessibility, and transparency of models across the system, rather than as a standalone component specific to this prototype implementation.

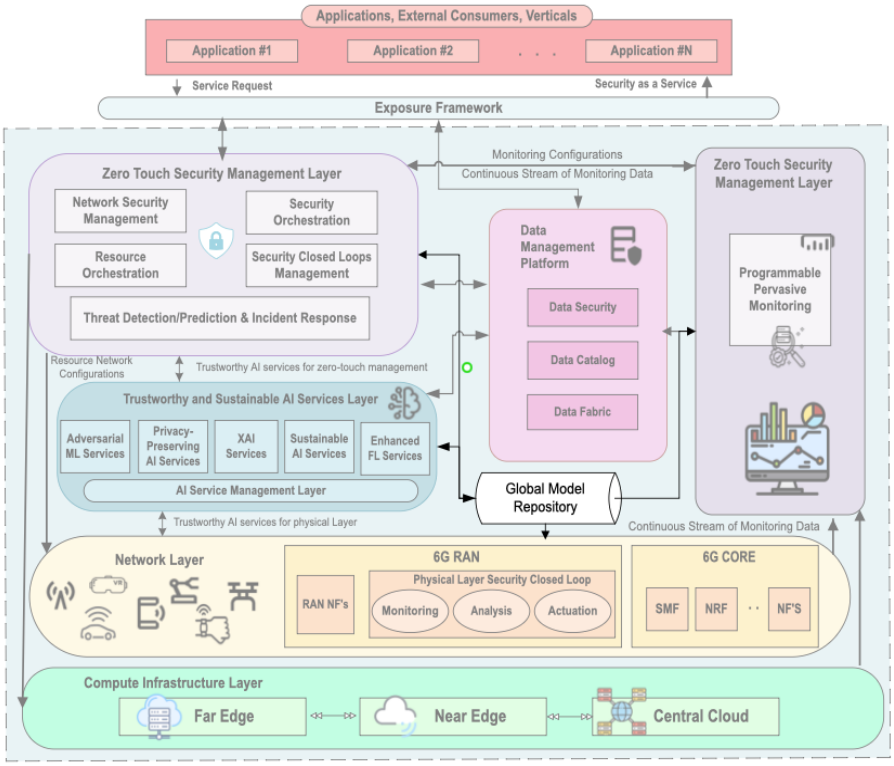

Figure 2 Placement of the GMR within the architecture

2.2.1 Alignment with future 6G networks

At the time of writing, a finalized and globally standardized 6G architecture does not yet exist. Standardization activities are ongoing, and multiple flagship research initiatives are contributing architectural blueprints, system visions, and enabling technology frameworks. Consequently, alignment with 6G must be assessed against emerging architectural directions rather than fixed specifications.

Among the most influential European efforts are the Hexa-X and Hexa-X-II flagship projects under the Smart Networks and Services Joint Undertaking (SNS JU). Hexa-X established the early vision of 6G as an AI-

native, sustainable, and trust-centric network of networks, emphasizing distributed intelligence across the edge-cloud continuum. Hexa-X-II advances this work toward a system blueprint, refining architectural enablers and validating key components that will inform pre-standardization discussions.

Across Hexa-X and Hexa-X-II documentation, several recurring architectural principles can be identified:

- AI integrated natively into network operation and control loops,
- Distributed and edge-centric intelligence rather than centralized control,
- Trustworthiness and security embedded by design,
- Sustainability as a structural objective rather than an optimization afterthought,
- Zero-touch automation and closed-loop orchestration.

While these initiatives do not yet define a final standardized 6G architecture, they provide the most concrete and structured architectural vision currently available at European level.

In parallel, SNS JU projects such as 6G-SANDBOX and 6G-BRICKS further refine practical and modular perspectives of 6G systems. 6G-SANDBOX promotes distributed experimentation infrastructures and programmable integration of AI-enabled services across heterogeneous environments. 6G-BRICKS focuses on modular 6G building blocks and programmable architecture elements designed to be composable and reusable across future deployments. These initiatives reinforce the view that 6G will not be a monolithic architecture but a flexible, programmable and AI-driven system.

Within this evolving landscape, the final version of WP3 prototype can be positioned as a concrete instantiation of one essential architectural layer anticipated in future 6G systems: the AI governance and distributed intelligence layer.

2.2.2 AI-Native Operation

Hexa-X and Hexa-X-II articulate the concept of AI-native networking, where artificial intelligence (AI) is embedded into network management, orchestration, and optimization processes rather than operating as an external analytics add-on. The WP3 prototype reflects this principle through its Decentralized Federated Learning backbone, which enables on-demand model training and lifecycle management integrated with higher-layer orchestration components.

Rather than training models offline and deploying them statically, the prototype supports dynamic model creation, validation, storage, and retrieval through defined APIs. This aligns with zero-touch and closed-loop control visions articulated in SNS architectural discussions.

2.2.3 Distributed Intelligence and Edge-Cloud Continuum

A defining characteristic of emerging 6G visions is the move toward distributed intelligence across edge, core, and cloud domains. The elimination of centralized aggregation within the DFL framework directly reflects this architectural direction. By enabling peer-to-peer coordination of model updates and resilience against selective node failures, the prototype embodies the distributed intelligence paradigm anticipated for 6G.

This design choice is consistent with Hexa-X system enablers emphasizing decentralization and with experimentation-driven architectures promoted in 6G-SANDBOX.

2.2.4 Trustworthiness by Design

Trust and security are recurrent themes across all major 6G flagship initiatives. Rather than treating security as a separate control plane function, emerging 6G visions promote trustworthiness as a structural property.

The WP3 prototype embeds privacy preservation, adversarial robustness, and trust evaluation mechanisms directly within the learning process. By integrating adversarial simulation and robust aggregation into

decentralized training, the architecture addresses the type of threat vectors identified in early 6G research while remaining aligned with trust-centric architectural principles under development within SNS JU.

Importantly, the prototype does not define trust as a compliance checklist but as a measurable and continuously monitored property of the AI lifecycle.

2.2.5 Sustainability as a Core Architectural Objective

Sustainability is positioned in Hexa-X and Hexa-X-II as a defining design axis for 6G. Energy efficiency, reduced environmental footprint, and adaptive resource usage are considered structural requirements.

The WP3 prototype integrates energy-aware orchestration and model optimization within its training backbone. Client scheduling, communication efficiency, and model compression techniques are embedded into the learning process itself. Sustainability therefore becomes an operational parameter continuously evaluated during model generation, consistent with the sustainability-by-design approach promoted in emerging 6G blueprints.

2.2.6 Programmability and modular building blocks

Projects such as 6G-BRICKS emphasize modular, programmable architectural components that can be composed into larger systems. The WP3 prototype reflects this philosophy through its containerized deployment model, API-driven interaction mechanisms, and centralized Global Model Repository.

The Global Model Repository acts as a structured AI asset management layer, enabling versioned, annotated, and reproducible model artifacts to be accessed by other architectural components. This modular governance layer aligns with programmable building-block perspectives currently being explored within the 6G research ecosystem.

2.2.7 Positioning withing the 6G ecosystem

It is important to clarify that the WP3 prototype does not claim to define a complete 6G architecture. Rather, it implements a focused yet critical subsystem: a distributed, trustworthy, sustainable, and explainable AI layer capable of supporting higher-level orchestration and network management functions.

Where flagship initiatives such as Hexa-X-II provide a broad system blueprint, the WP3 prototype contributes a concrete, operational example of how AI lifecycle management, decentralized learning, and governance mechanisms may be realized within such a blueprint.

In this sense, the prototype can be viewed as a compatible and complementary building block within the emerging 6G architectural ecosystem.

2.3 Decentralized federated learning framework

2.3.1 Introduction and Scope

This section focuses on the practical instantiation, real-world usage, and validation of the prototyping environment for the DFL framework, and on its tight integration with the GMR. While Deliverable D3.3 presented an initial integrated prototype in which DFL was optional, in this final architecture, the DFL framework has been elevated to the system's core operational backbone. In this final version, all model training processes are conducted directly within the DFL framework. The objective of this document is distinctly practical: to demonstrate the technological maturity of this component through tangible deployments.

Trustworthiness, explainability, and sustainability are no longer external services; since D3.3, they have been directly embedded into the training lifecycle as intrinsic structural properties aligned with future 6G networks. By structurally integrating all this functionality into the DFL, the prototype aligns naturally with the distributed intelligence paradigm expected in 6G systems, where learning occurs across edge, core, and cloud environments without centralized control. The framework now features greater modularity, integrating seamlessly with attack detection systems, and the explainability pipeline (XAI) relies on tools like ShaTS [FPM25] to generate artifacts stored directly in the GMR.

2.3.2 Framework Instantiation and Deployment Configuration

The technical implementation of the DFL framework and the GMR aligns with the distributed intelligence paradigm expected in 6G systems, orchestrating collaborative training across the edge-cloud continuum. The prototype is prepared for environments with high computational demands, offering hardware acceleration support (GPU) through dedicated containers using a common definition (e.g. Dockerfile-gpu), optimizing the training of complex models (e.g. PyTorch).

To guarantee reproducibility, portability, and agility aligned with zero-touch automation visions and closed-loop orchestration, all components have been containerized. The orchestration via docker-compose.yml seamlessly manages backend interfaces with FastAPI and Uvicorn, and persistence with PostgreSQL (see Figure 3). This API-driven modular deployment model follows the "programmable building blocks" philosophy anticipated for 6G networks, enabling other architectural components to access versioned and annotated models.

```
1
      ▷Run All Services
2     services:
3
                ▷Run Service
4                 particpant0:
5                   image: robust-gpu
6                   deploy:
7                     resources:
8                       reservations:
9                         devices:
10                          - driver: nvidia
11                            count: all
12                            capabilities: [gpu]
13                  volumes:
14                      - /home/umu/Documents/cyberdatalab/robust/ROBUST-6G_DFL_Framework:/robust
15                  extra_hosts:
16                      - "host.docker.internal:host-gateway"
17                  ipc: host
18                  privileged: true
19                  command:
20                      - /bin/bash
21                      - -c
22                      - |
23                        ifconfig && echo '192.168.50.1 host.docker.internal' >> /etc/hosts && python3.12 /robust/federation.py
24                  networks:
25                      robust:
26                        ipv4_address: 192.168.50.2
27
                ▷Run Service
28  >             particpant1: ···
53
                ▷Run Service
54  >             particpant2: ···
79
80
81    networks:
82        robust:
83            name: robust
84            driver: bridge
85            ipam:
86              config:
87                - subnet: 192.168.50.0/24
88                  gateway: 192.168.50.1
```

Figure 3 Code snippet of docker-compose.yml showing the environment configuration, ports, and volumes

Integration with remote clients and the continuous collection of system-level and model-level metrics are configured through environment variables. This establishes the initial P2P topology among the federated nodes without manual intervention, supporting programmable, modular building blocks for future 6G deployments.

2.3.3 Operational Workflows and Security Mechanisms

By enabling peer-to-peer coordination without relying on a centralized aggregator, the DFL framework eliminates single points of failure and enhances resilience. The complete lifecycle of a DFL node embeds trustworthiness structurally through several automated phases (see Figure 4):

1. Initialization and Node Recovery: Peer discovery and P2P topology are maintained via gossip (gossiper.py) and life signals (heartbeater.py), enabling dynamic reconfiguration when edge nodes fail.

2. Local Training and Robust Aggregation: Decentralized training is implemented using PyTorch Lightning (federation.py). Robust aggregation strategies are enforced here to assess and ensure resilience during the training phase. Scalable and customizable aggregation methods, such as DFedADMM [PBM+26], enable nodes to optimize the objective function while avoiding excessive displacement from neighbours' models, thereby ensuring sustainability and robustness by design.
3. Weight Encryption (Embedded Trustworthiness): As a critical privacy-preserving mechanism, the encrypter.py module encrypts model weights before transmission, protecting the network against interference or eavesdropping.
4. Governance in the GMR: Following a federated round, the final models, intermediate artifacts, and evaluation metrics are shipped to the GMR (via server.py and database_adapter.py). The GMR serves as the central AI asset management and governance layer, ensuring full traceability, consistency, and accessibility throughout the model lifecycle.

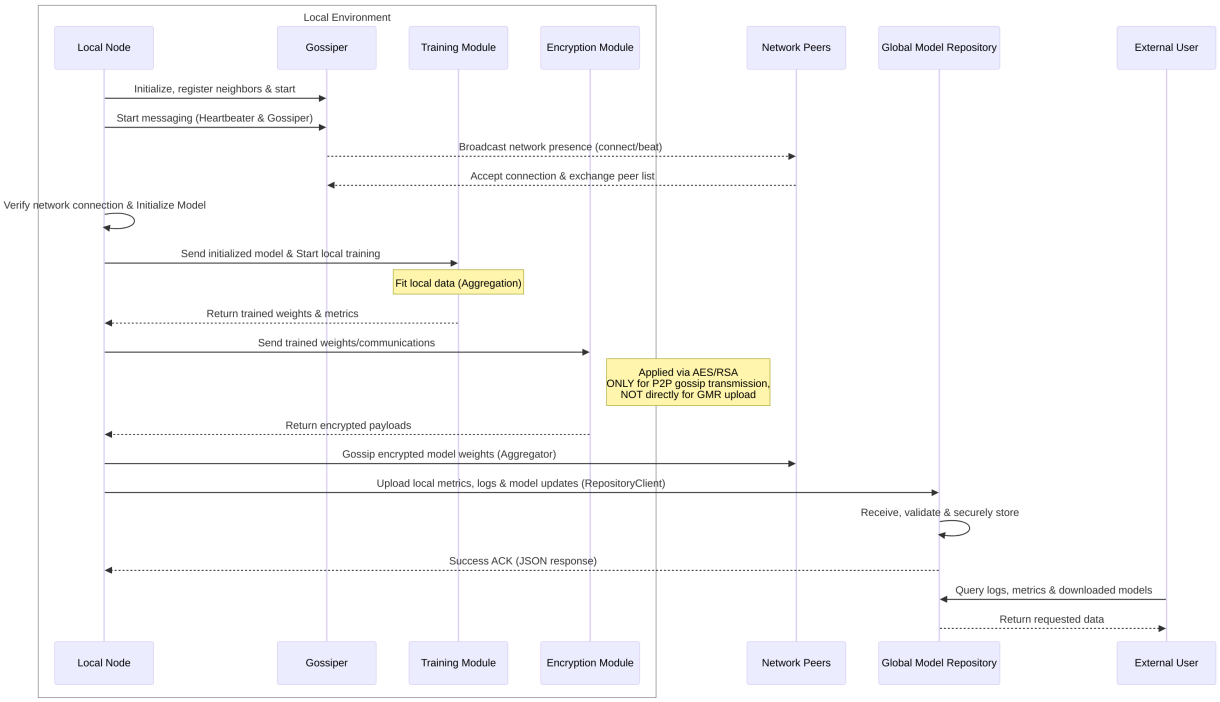

Figure 4 Lifecycle, from node initialization in the `gossiper` to the weight encryption and storage in the GMR

Daily management and scenario configurations are dynamically executed through an intuitive web interface, enabling administrators to simulate adversarial scenarios and observe the framework's resistance effortlessly.

2.3.4 Prototype Demonstration and Security Validation Evidence

The operational maturity is verified by the framework's ability to execute complex distributed tasks and withstand simulated adversarial behaviour natively. The frontend dashboard (templates/index.html) facilitates dynamic deployment launches, API-driven interactions, and deployment monitoring. Figure 5 shows the configuration selectors and the real-time metrics received after a federation deployment.

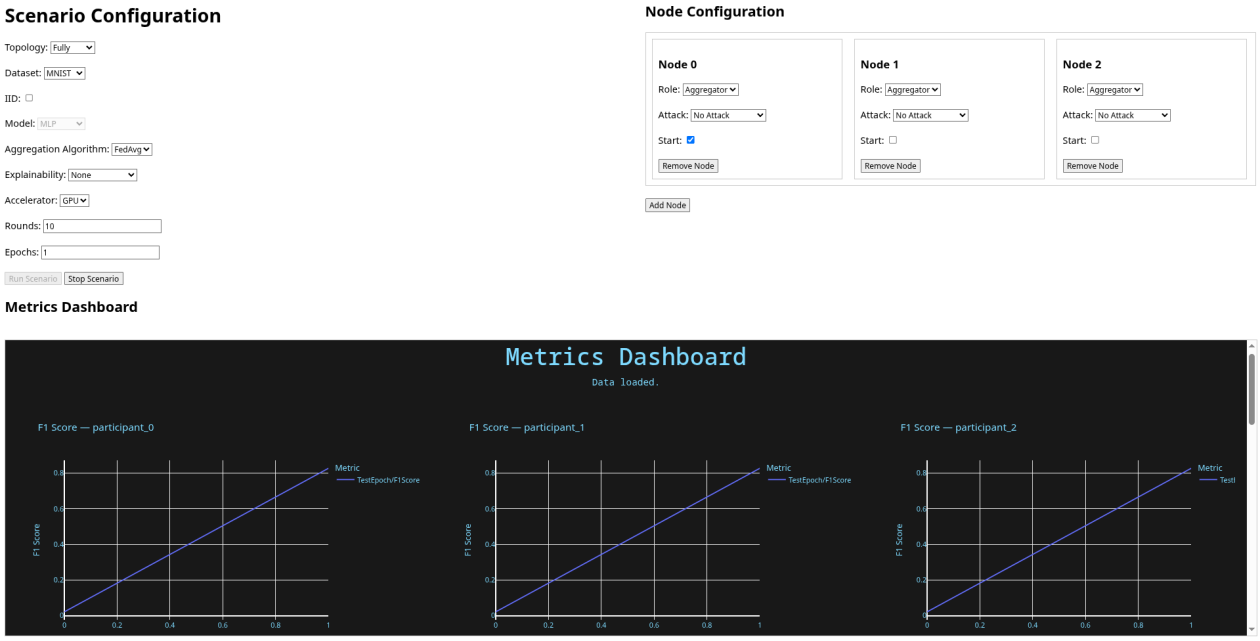

Figure 5 Frontend Web Interface

A fundamental aspect of this final architecture is its built-in capability to evaluate the impact of cyber threats. During pilot operations, the system successfully simulated adversarial vectors, such as Model Poisoning (attacks.py), natively integrated into the DFL framework. This permits researchers to observe the evolution of the attacks, assess model explainability, and test potential mitigation strategies. By leveraging embedded XAI services, operators can gain insights into feature-level relevance and associated confidence levels to interpret these security incidents effectively. Furthermore, empirical logs and captured metrics effectively validate the preservation of confidentiality and adversarial robustness, reinforcing the system's alignment with "Trustworthiness by Design" principles expected in 6G environments.

2.3.5 Conclusions and Operational Readiness

The evidence gathered from the generated logs, native containerized GPU acceleration, embedded cryptographic schemes, and the dynamic web interface demonstrates that the DFL framework has advanced significantly beyond Deliverable D3.3. Achieving a Technology Readiness Level (TRL) of 5 to 6, this DFL infrastructure, together with the associated GMR asset management layer, represents a fully operational, resilient, and transparent AI fabric. Evaluated under rigorous conditions and fully aligned with core 6G design parameters trustworthiness by design, decentralized edge-centric intelligence, sustainability, and zero-touch API programmability, this modular architecture is ready to perform as a crucial intelligence layer integrated with the overarching ROBUST-6G platform.

2.4 Core AI modules

2.4.1 Trustworthy AI Module

In FL, although encryption at the Transport Layer Security (TLS) level ensures that the communication channel between participants (i.e., clients) and the server is protected against external adversaries, such as eavesdropping or man-in-the-middle attacks, TLS only secures data in transit and does not address threats originating from within the system. In FL, the server can observe the participants' model weights in plaintext, which can expose sensitive information. Homomorphic encryption (HE) enables clients to encrypt their model updates before transmission. The server (aggregator) can then perform aggregation directly on the encrypted parameters without accessing their underlying values. This ensures end-to-end privacy for individual client updates, protecting them from an honest-but-curious aggregator. However, integrating HE into FL introduces a significant challenge: how can poisoning attacks be detected when all client updates are encrypted? Poisoning

remains one of the most critical threats in encrypted FL environment. A malicious client may submit manipulated, incorrect, or backdoored model updates during aggregation with the goal of degrading the global model's performance or embedding hidden malicious behaviours.

In ROBUST-6G, we propose a privacy-preserving framework to mitigate poisoning threats in FL. The approach leverages homomorphic encryption to ensure data privacy while detecting malicious updates through encrypted model distance analysis, enabling the system to identify and filter compromised clients without exposing any sensitive model information. To the best of our knowledge, this is the first framework capable of detecting poisoning attacks in fully homomorphically encrypted FL settings without requiring plaintext access, trusted hardware, or complex multi-party communication.

In our system model, Figure 6, we consider a horizontal FL scenario where a central server (aggregator) orchestrates multiple distributed clients to jointly train a global model without directly accessing their local datasets. The server operates under an honest-but-curious (semi-honest) assumption: it faithfully executes the defined protocol but may attempt to infer sensitive information from any data it can observe during the process.

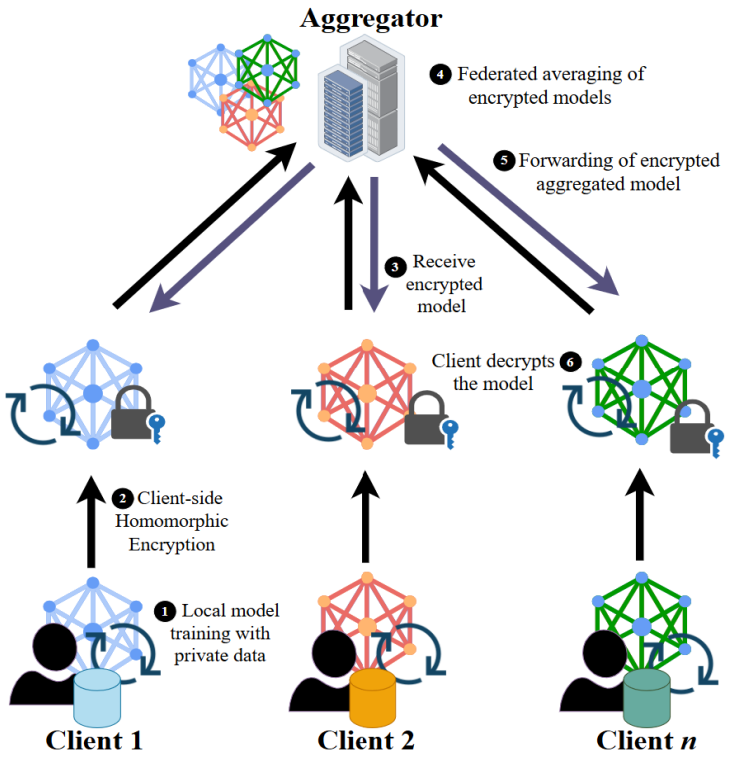

Figure 6 Overview of the system model

2.4.1.1 HE-based Poisoning Attack detection and Mitigation in Federated Learning

At the beginning of each FL round t, the server distributes the current global model $w_{glob}^{(t|1)}$ to all clients. Each client performs local training using its private dataset and produces an updated model $w_i^{(t)}$. To support encrypted computation, the model parameters are flattened into a vector and divided into fixed-size chunks. Each chunk is then encrypted using the shared public key and transmitted to the aggregator.

Once the encrypted updates are received, the server measures the deviation of each client update from the previous global model. For each client and model chunk k, the server computes the squared distance

$$d_{i,k} = \| w_{i,k}^{(t)} - w_{glob,k}^{(t|1)} \|^2$$

using homomorphic operations directly on encrypted values. This produces a client–chunk distance matrix D, where each element represents the deviation of a client update from the expected model behaviour.

Because the server does not possess the decryption key, a masked verification protocol is used to securely recover these distances. The server embeds the encrypted distances inside a larger matrix N containing random noise. In practice, this matrix is moderately expanded (e.g., 1.5×–3×) to introduce randomness and prevent inference of client identities or system structure, thereby ensuring strong privacy guarantees. This matrix N is sent to a selected subset of m clients required for decryption. If the models use the same secret key, then even a single client is sufficient for the decryption, given the process is repeated a minimum of two times. Since the real values are randomly scattered among masked entries, clients cannot infer which elements correspond to actual model distances. After decryption, the server removes the mask and reconstructs the true distance matrix.

To identify potential poisoning behaviour, the server computes a client-level anomaly score

$$s_i = \frac{1}{K}\sum_{k=1}^{K} d_{i,k}$$

where $K$is the number of model chunks. These scores capture how strongly each client update deviates from the global model.

The score vector s_i is then analysed using K-Means clustering to separate normal and anomalous updates. If the two clusters show sufficient separation, the cluster with higher variance is considered to contain poisoned updates. Clients in this cluster are excluded from aggregation, and only benign updates are used to compute the next encrypted global model.

Experimental Results

Impact of Poisoning on Federated Learning

The first experiment evaluates how poisoning affects a Neural Network (NN) model accuracy. A convolutional neural network (CNN) with two convolutional layers followed by a dropout layer is used for classification. A FL system with 20 clients was simulated, where 50% of the clients were malicious. Here, the poisoners are added with zero-mean Gaussian noise with a standard deviation of 10 for only 1 chunk of 400 parameters.

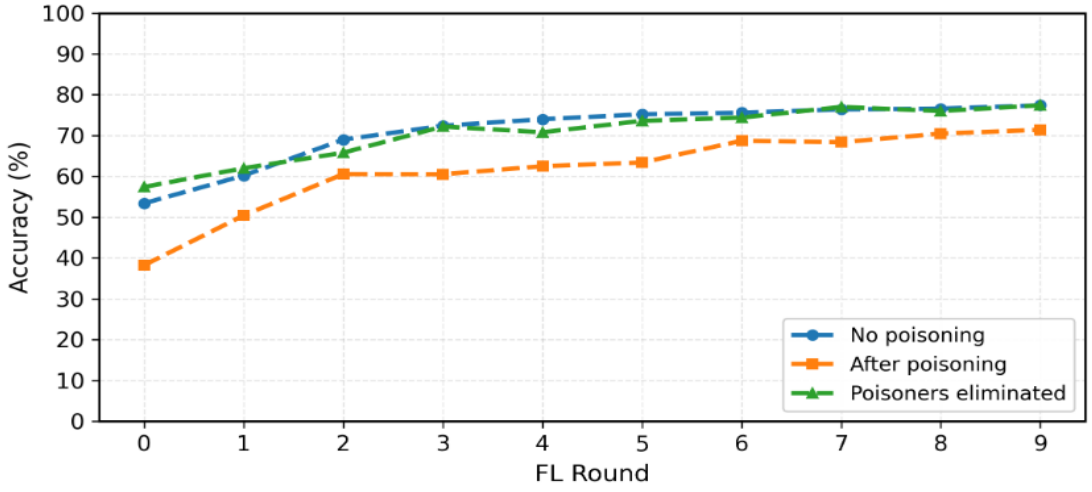

Figure 7 Effect of poisoned clients on the aggregated model accuracy

The impact of poisoning attacks on FL performance with sample dataset MNIST is illustrated in Figure 7, where the aggregated model accuracy is compared for three scenarios: (i) all benign clients, (ii) poisoned clients participating in training, and (iii) poisoned clients removed using the proposed detection method. The

results show that poisoning can reduce model accuracy by approximately 10%, while the proposed detection mechanism restores performance close to the benign baseline.

Table 1 Detection performance across multiple poisoning attack types and poisoning ratios

Metric	15% poi.	25% poi.	50% poi.	75% poi.	85% poi.
Gaussian dense model poisoning					
TPR	1.000	1.000	1.000	1.000	1.000
TNR	1.000	1.000	1.000	1.000	1.000
FPR	0.000	0.000	0.000	0.000	0.000
FNR	0.000	0.000	0.000	0.000	0.000
Acc	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
Uniform noise data poisoning					
TPR	0.600	0.840	0.900	1.000	1.000
TNR	0.600	0.800	0.900	1.000	1.000
FPR	0.400	0.200	0.100	0.000	0.000
FNR	0.400	0.160	0.100	0.000	0.000
Acc	**0.600**	**0.810**	**0.900**	**1.000**	**1.000**
Top-k model poisoning					
TPR	0.900	1.000	0.980	0.980	0.953
TNR	0.900	1.000	1.000	1.000	1.000
FPR	0.100	0.000	0.000	0.000	0.000
FNR	0.100	0.000	0.020	0.020	0.047
Acc	**0.900**	**1.000**	**0.990**	**0.990**	**0.965**

Table 1 presents the poisoning detection performance across different attack strategies, including Gaussian and uniform noise poisoning, and targeted top K parameter poisoning. The results show that the proposed method achieves perfect detection accuracy for most attack types such as Gaussian noise poisoning and uniform noise-based parameter manipulation even when the proportion of malicious clients increases.

Computational Overhead

The computational overhead of the proposed framework is summarised in Table 2, which reports the latency of the main steps in the encrypted poisoning detection pipeline for a FL setup with 20 clients. In this configuration, each model chunk contains 1000 parameters, and the resulting client–chunk distance matrix is embedded into a masked structure of size 2000 for secure verification. The evaluation is conducted on an MNIST-based model with approximately 100,000 parameters in total. In practice, evaluating a subset of chunks is sufficient for poisoning detection, however, analysing all chunks would provide maximum security, with the computational cost scaling linearly with the number of chunks. The results show that most of the processing time is associated with cryptographic operations required by the CKKS homomorphic encryption scheme. On the client side, the main cost comes from encrypting model chunks and decrypting the masked distance matrix, while on the server side the dominant operation is the homomorphic distance computation between client updates and the previous global model. Other steps such as distance matrix construction, verification, and poisoning detection incur relatively small overhead per chunk. Overall, the total processing time for handling encrypted updates from all clients is approximately 1.96 seconds, indicating that the proposed detection mechanism introduces linear additional latency beyond the necessary encryption operations. However, it should be noted this delay will only be applied during the training phase. Once the models are trained, there will be no delay as the models do not require HE-based communication at the inference time.

Table 2 Computational latency for HE-based poisoning detection operation

Comp.	Operation	Mean Time (s)	Std (s)
Client	Chunk Encryption	0.4099	0.0226
	Distance matrix decryption	0.3559	0.0635
Server	Encrypted distance computation	1.0837	0.0243
	Distance matrix construction	0.0628	0.0173
	Chunk Verification	0.0055	0.0020
	Poison detection	0.0478	0.1003

2.4.1.2 HE-based Aggregation Method for Decentralised FL

In HE-based traditional FL, the presence of a central aggregation server introduces a potential single point of failure and a trust dependency. To address this limitation, DFL replaces the central coordinator with peer-to-peer communication among participating clients. In this setting, homomorphic encryption enables clients to encrypt their locally trained model updates, exchange them with neighbouring peers, and perform aggregation over ciphertexts. However, unlike HE-based traditional FL where a shared secret key may be used for encrypting model updates, DFL must ensure that no single client can access the raw updates of others while still enabling global model convergence. Therefore, collaborative mechanisms such as secure Multi-Party Computation or Threshold Cryptography can be used to enable the decryption of the aggregated global update without revealing any client's contribution. Overall, integrating homomorphic encryption with DFL enhances privacy guarantees, eliminates reliance on a trusted central server, and improves system robustness.

The Trustworthy AI module aims to enhance privacy in model training across distributed nodes, where nodes might be "trusted but curious", meaning that they follow the protocol correctly but attempt to extract as much information as possible from the model updates exchanged during training. In the following, we briefly describe the multi-key HE-based functionality of the Trustworthy AI module in a decentralized federated learning setting.

- *Privacy Management Function:* decides about the privacy operation that is supported by all participants and shares the required parameters for the privacy operation before triggering the DFL process.
- *Joined Key Generation Function:* computes a joint public key by engaging related nodes.
- *Local Encryption Function:* Each client encodes its local update, encrypts using the joint public key, and sends ciphertext to the coordinator or peers, depending on protocol.
- *Collaborative Secure Aggregation Function*: runs at each node to aggregate received encrypted model updates from neighbours using secure aggregation method.
- *Collaborative Decryption Function:* this function is run at **e**ach node to compute a partial decryption share on aggregated encrypted result using node's own secret key and sends it to other nodes. This partial decryption does not leak any information related to the node's model update because the result still has other nodes' contributions mixed in.
- *Fusion and Decoding Function:* run at each node to combine all received shares and obtain the aggregated sum.

We conducted a PoC to evaluate the approach in DFL setting and to evaluate the associated computational overhead. In our implementation, we employ the CKKS Scheme using the OpenFHE C++ library with python bindings. Currently, Python supports only n-of-n threshold schemes, meaning that all clients participating in the selected DFL cluster must participate in the decryption process.

In OpenFHE, chunking is required because encrypted data structures in Homomorphic Encryption schemes have size and capacity limitations, especially when used for ML workloads such as FL. In schemes like the CKKS Scheme, a ciphertext can only encode a fixed number of values (called slots), determined by the polynomial modulus degree. When model update contains more parameters than the slot capacity, it cannot fit into a single ciphertext. Therefore, the vector must be split into chunks, where each chunk fits into one ciphertext and encrypted separately. CKKS also supports Single Instruction Multiple Data (SIMD)-style

packing, allowing operations such as addition, averaging, and aggregation to be performed simultaneously on multiple values. Consequently, chunking improves computational efficiency, reduces memory overhead, and facilitates parallel processing, which is particularly beneficial in distributed settings such as DFL.

Figure 8 depicts application of Chunking on model updates. At each federation round, clients first train locally, split their model into chunks$k$ chunks, encrypt each chunk, and then merge them into the final encrypted model. This approach handles CKKS capacity limits while ensuring secure model sharing.

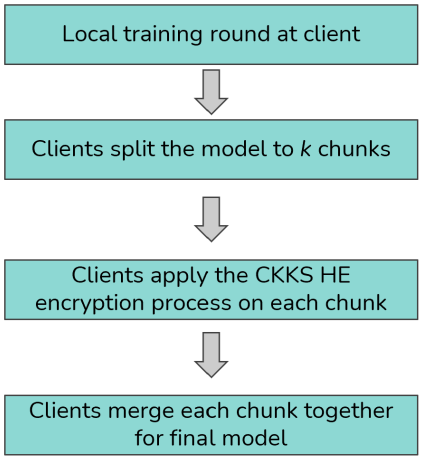

Figure 8 Application of Chunking on model updates

Figure 9 presents the latency of individual functions in our module as a function of chunk size, while keeping the number of clients fixed at 100, and median values computed over three runs. Chunk sizes range from 0 to 8000 slots. The total latency (blue line) increases slightly as the chunk size grows. This indicates that processing larger chunks adds computational cost, but the increase is moderate rather than dramatic. Most of the increase in total latency is driven by the encryption function, while other functions remain relatively stable.

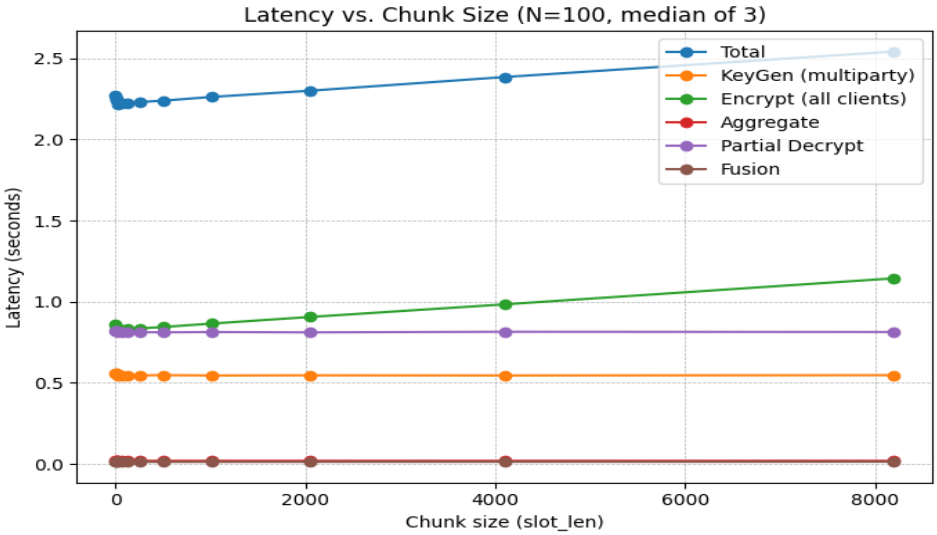

Figure 9 Latency vs. chunk size

Figure 10 shows how the latency of individual functions in our module changes as the number of clients (N) increases. The x-axis represents the number of clients, while the y-axis shows latency in seconds. The total latency (blue line) increases almost linearly with the number of clients. This indicates that the system scales

reasonably predictably as more clients participate. The encryption and decryption-related phases dominate the total runtime, while aggregation and fusion add negligible overhead.

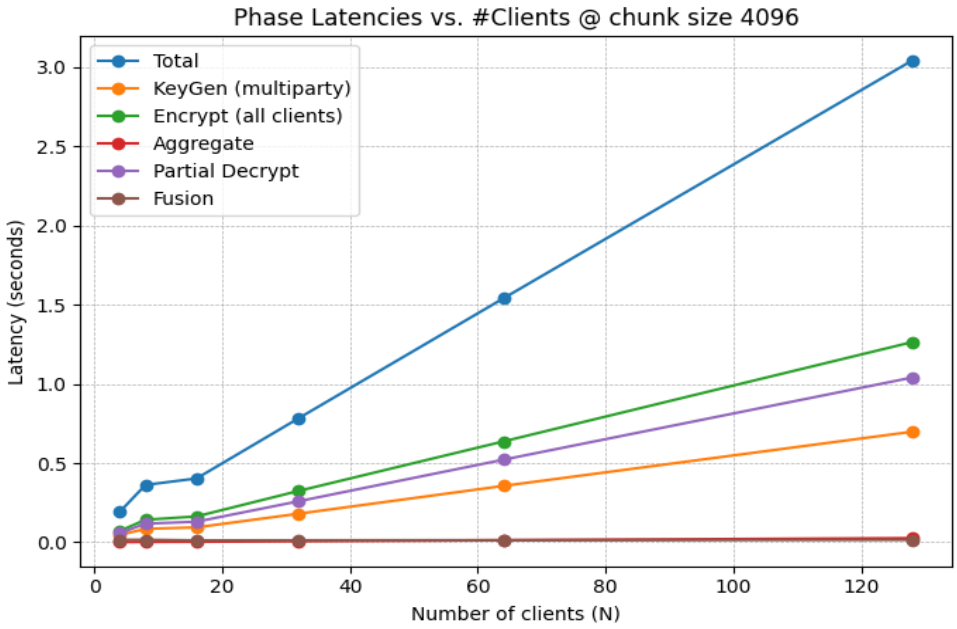

Figure 10 Latency with respect to number of clients

Integration in the prototype. Currently, the proposed solution is not integrated with the DFL framework.

2.4.1.3 Membership Inference Attack Mitigation for Decentralised FL

At the DFL clients, the received individual client models should be evaluated by each client to mitigate any threat from inference attacks. For this, a new algorithm was designed to early identify such anomalies from Layerwise Relevance Propagation-based XAI technique. In the LRP, the contribution of each neuron can be obtained via a relevance score, which is unique to a particular data instance. The difference between the contribution scores provides how deviating each data instance from each other, though they belong to the same class. This is illustrated in Figure 11, where the LRP scores are different.

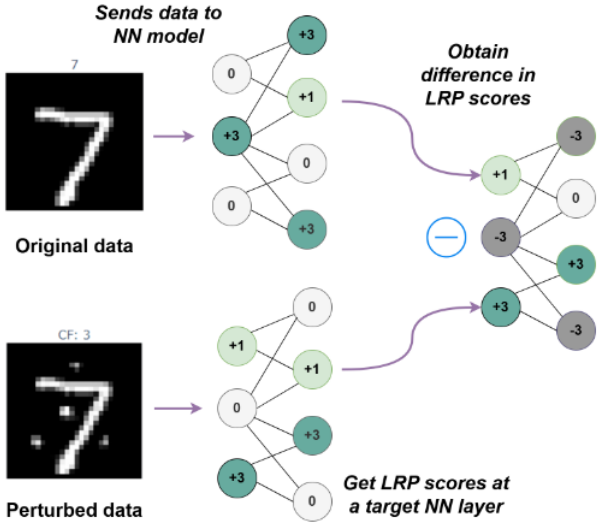

Figure 11 LRP-based score differences of a sample image with its perturbed variant

In membership inference attacks, an adversary aims to differentiate such subtle, unique variations among the data instances and attempts to infer if a client consists of that data instance in their private training data. Local client models trained with such unique variations in a private dataset create specific decision boundaries which the attackers exploit in the membership inference. Therefore, the proposed algorithm identifies such unique decision boundary variations via observing the differences in the LRP scores and trim these unique patterns to mitigate membership inference attacks.

The process is as follows:

1. Receive neighbouring models $M_i^{(t)}$ from peer client i at round t and perform local validation before aggregation.

2. Compute neuron relevance scores $R_j(x)$ for local samples $x \in D_{sub}$, where D_{sub} is the local client subset of data.

3. Identify distinctive neurons using statistical filtering (e.g., $z_j = \frac{R_j - \mu}{\sigma}$, retain if $| z_j | > z_{th}$).

4. Define sensitive neuron set $\mathcal{N}_s = \{j \mid\mid z_j \mid > z_{th}\}$.

5. Freeze common neuron parameters $\Theta_{common} = \Theta \setminus \Theta_s$ to preserve general behaviour.

6. Apply controlled perturbation $\mathcal{N}$ to unique sensitive parameters:

$$\Theta_s' = \Theta_s + \epsilon \cdot \mathcal{N}(0,1)$$

7. Update model parameters $\Theta' = \Theta_{common} \cup \Theta_s'$ and fine-tune using D_{sub}.

8. Share updated model $M^{(t+1)}$ for aggregation with other peers with reduced inference leakage and preserved utility.

With this approach, it was observed that the accuracy of the membership inference attacks is significantly reduced across multiple datasets. For the experiments, two neighbouring peers were considered, where one client is the adversary. The target peer continuously sends model updates trained to the attacker peer, where the attacker aims to infer the membership state of sample subset of data of the target. Here, a zero mean gaussian noise is added to the sensitive neuron weights.

Table 3 Membership inference accuracy without vs. with the defence

Dataset	Defence status	Attack acc.	Precision	Recall	F1
MNIST	No Defence	0.875	0.880	0.900	0.880
	With Defence	**0.400**	0.420	0.430	0.420
5G-NIDD	No Defence	0.967	0.960	0.970	0.970
	With Defence	**0.467**	0.470	0.470	0.460

From the results, it is observed that the attack accuracy has significantly degraded in both datasets, as the unique features that lead to membership inference has been eliminated. Therefore, the peers can forward their client model to unknown client peer in a decentralised FL system with minimum inference risk.

2.4.2 Sustainable AI Module

The Sustainable AI module aims to (i) reduce the ecological footprint and (ii) enhance the scalability of AI in 6G networks by

1. Optimizing training/scheduling algorithms for energy efficiency based on the network context:
    a) Scalable and highly customizable DFL aggregation methods,
    b) Energy- and semantics-aware client scheduling and optimization algorithms for FL and DFL.
2. Designing sustainable models for deployment of resource-constrained devices. For this, we have investigated spiking neural networks (SNNs).

In what follows, we briefly detail each functionality of the Sustainable AI module, its level of maturity, and status of integration into the prototype (specifically, in the source code of the DFL training framework).

2.4.2.1 Scalable aggregation methods for DFL

Regarding scalable and robust aggregation methods specifically tailored for DFL (1.a), we designed and developed two different algorithms based on dual optimization and message passing.

- DFedADMM is an algorithm based on the alternating direction method of multipliers (ADMM) that requires sending to the neighbors the local model parameters, exactly as in the decentralized version of federated averaging (FedAvg).
- DFedDRS is a more customizable alternative based on the Douglas-Rachford splitting, an equivalent formulation of the ADMM with a further hyperparameter to tune the convergence speed and practically more stable. Here, the nodes need to keep per-edge versions of the messages (i.e., the communicated local model differs based on the neighbor).

The main difference between this approach and conventional approaches is that we do not modify directly the local model with custom integration from neighbors' information but rather use this information to compute a penalty added to the objective function to avoid excessive displacement from the neighbours' models. Choosing the weight of this penalty trades scalability and model personalization for adherence to a strict version of the global model (and convergence speed).

Integration in the prototype. DFedADMM is already implemented in the source code of the DFL framework and is available for training models. DFedDRS is currently under implementation.

From a technical perspective, this is done by extending the class Aggregator with new classes that inherit the methods and that are fully compatible with the rest of the code. An adaptation to the local training phase is needed as the objective function is modified with an L2 penalty.

2.4.2.2 Energy- and semantics-aware client scheduling and optimization

On the scheduling/resource allocation side (labeled 1.b),

- The battery-aware cyclic scheduling scheme previously validated empirically [JP25a] has been extended with a formal convergence analysis, characterizing the impact of intermittent client participation under energy harvesting constraints on global model convergence [JP25b]. This provides analytical tools for predicting the accuracy-energy trade-off.
- A lightweight semantics-aware scheduling mechanism based on Version Age of Information (VAoI) has been developed, enabling clients to locally assess the timeliness and usefulness of their updates without costly model similarity computations, thereby reducing both energy consumption and communication overhead at the client side [JPY+26].
- An optimization framework for the allocation of uplink radio resources has been proposed for two concurrent set of devices: FL devices uploading their model to the access point and generative AI (GenAI) devices accessing the medium via random access (ALOHA or slotted-ALOHA) [PJP26].

Integration in the prototype. Currently, none of the proposed solutions are integrated with the DFL framework.

Nonetheless, a native way to enable integration is available in the framework source code through the function get_partial_aggregation(self, except_nodes) of the class Aggregator. Specifically, a set of nodes except_nodes can be passed to the function to exclude them from aggregation. This enables selecting only the nodes chosen from the developed client scheduling policies based on the energy context.

2.4.2.3 Sustainable ML models by design (SNNs)

Designing models that are natively sustainable and energy-efficient rather than optimized afterwards is of the utmost importance. Specifically, we investigated SNNs, a type of neural network closely mimicking how the

human brain works with three orders of magnitude better energy-delay-product (EDP) concerning classic neural networks [RBG+22].

In Figure 12 we show the working principles of an SNN. Input trains of current "spikes" are multiplied by the network parameters and sum up to generate the *membrane potential*, a state variable of neurons. Whenever the potential surpasses a firing threshold, a current spike is emitted at the neuron's output, and the potential is reset to a base value. The sparser are the signals and the neurons' firing rates, the more energy efficient is the SNN.

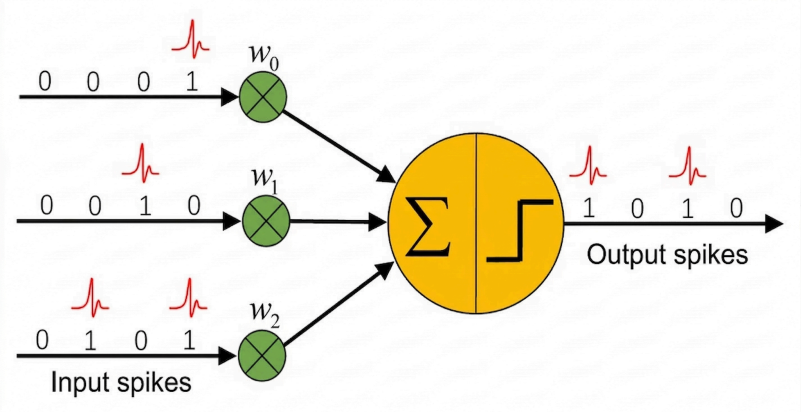

Figure 12 Spiking neural network working mechanism

For SNNs, we

- Designed LightSNN, a lightweight architecture search approach to improve the performance of candidate SNNs has been designed and tested [APM+25]. This allows choosing a promising SNN architecture before training and in an agnostic way with respect to the dataset.
- Proposed a novel convolutional spiking-based gate recurrent unit (GRU) cell to better process spatio-temporal data [ACR25]. This layer can be effectively used for tasks like network traffic recognition, attack and anomaly detection, and temporal series prediction.
- Proposed a novel training approach specifically tailored for SNNs that solves the non-differentiability of the Heaviside step function during backpropagation [PBM+26]. The method, based on the ADMM, completely replaces the need to backpropagate the gradient from the output with exact and/or approximate iterative solutions of convex subproblems.

Integration in the prototype. We made available directly in the source code of the DFL framework an SNN model with optimized architecture through LightSNN, and with a sample dataset for training. Training is performed, as of now, via the existing snnTorch tool, using standard surrogate backpropagation.

2.4.3 Explainable AI Module

Deploying security measures across evolving network environments demands not only robustness but also the verified reliability of the deployed AI/ML models. To systematically assess and ensure this, the Explainable AI module introduces strategic frameworks designed to objectives, specifically focusing on:

- The interpretability level of the AI/ML security models.
- The stability of model outputs via dynamic confidence scoring.
- The capacity for model optimization while preserving architectural transparency.

2.4.3.1 SHAP-Based Model Enhancement and Incident Reporting

Due to the sheer complexity and vast volume of heterogeneous data inherent to 6G networks, deploying security models that are highly accurate, lightweight, energy-efficient, and explainable is critical. To address

these requirements, we propose two distinct, yet integrable, services that leverage XAI for model optimization and post-hoc analysis.

Within the framework of the ROBUST-6G Architecture, the first service provides an XAI-driven model optimisation. Its primary function is to systematically refine security models, such as Intrusion Detection Systems (IDS), by identifying and retaining only the most impactful input features. Traditional IDS architectures deployed in high-dimensional network environments frequently struggle to identify minority attack types that lack distinguishable traffic patterns. Furthermore, processing raw, high-dimensional telemetry data imposes a severe computational overhead during both the training and inference phases. SHAPRefine resolves these critical bottlenecks by using Shapley Additive Explanations (SHAP)[LSM01] to distil the exact feature subspace required to train an optimal, lightweight, and highly accurate model. The framework is fundamentally model-agnostic, capable of operating across diverse behavioural architectures, and seamlessly handles both binary and multiclass threat vectors.

The process goes as follows:

1. An initial model is trained on network traffic to learn the patterns of available attack scenarios. This model can be of any architecture/behavioural background.
2. Shap values are then calculated dependent on the model and the training data.
3. Aggregation then begins across all available network scenarios. This aggregation accounts for the frequency of specific traffic classes, data distribution and each feature contribution to the model's prediction.
4. Then, each feature set is unified across the network scenarios/classes.
5. This final unified feature set is then used to select the features that are essential to develop a lightweight, high-performing model in comparison to using all the available features. Feature selection is done through an iterative performance-driven algorithm, which analyses feature interactions and how these interactions contribute to lower false negatives of the model's output.

By focusing on features with the highest impact, this approach produces lightweight, high-performing models tailored for real-time deployment in demanding network environments. Evaluation is done across diverse models with various underlying architectures, such as neural networks, tree based and linear models. Data was used from available open-source network intrusion datasets, including 5G based dataset to ensure its compatibility with next generation network environments. Results show up to 90% dimensionality reduction, 90% lower training time and energy consumption, detection capabilities improvement, with minority-class F1-scores boosted by up to 95%. To verify the deployment and efficiency of SHAPRefine's service, Figure 13 shows that the cumulative inference time grows far more slowly as the scale of data and the number of inference instances increase. As such, this service significantly lowers the computational overhead needed to run complex IDS on constrained edge computing nodes.

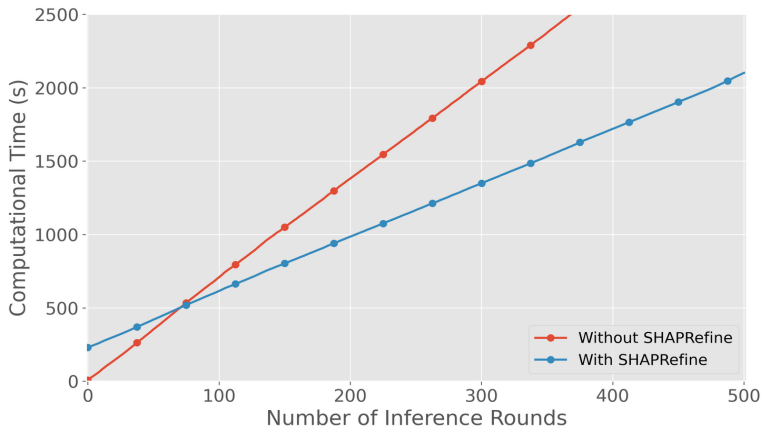

Figure 13 Cumulative inference time across deployment rounds for XGBoost models

Currently, this proposed service is not integrated with the DFL framework.However, in theory, this restricted input space can optimize the DFL pipeline, shrinking local model sizes to ensure ultra-low latency synchronization and reduced bandwidth consumption for GMR updates.

The core of explainability and interpretability is tackled in the second service, which utilises SHAP to generate detailed incident reports that decipher the reasoning behind a model's predictions in a human-understandable format. This second service can be applicable to any deployed model or integrated with optimised model after SHAPRefine's pipeline. The goal is to move beyond simple alerts and provide a comprehensive narrative that translates a model's complex decision-making process into transparent, human-readable, and actionable intelligence. It is designed to be consumed by both automated systems for orchestration and by human operators for in-depth analysis and final decision-making. This structured JSON report is the primary mechanism for communicating the model's reasoning and is crucial for integration with other system components and partners.

Instead of merely flagging a potential threat, the report provides a multi-layered explanation. It presents the model's final prediction alongside a full breakdown of its confidence across all potential outcomes, offering a nuanced view of the model's certainty. Critically, it quantifies the key pieces of evidence, the specific input features like traffic volume or connection states, that led to the conclusion. This evidence is then synthesized into a plain-language interpretation and a concrete recommendation, effectively telling the story behind the alert.

By providing human-understandable insights, this report serves as a vital bridge to other work packages. For the ZSM (WP4), it provides the necessary context for intelligent security orchestration. For the Trustworthiness Module (WP3), it offers the evidence required to validate the model's integrity. This integration is essential, as it ensures that explainability is not an isolated feature but a core enabler of system-wide trust.

In order for seamless integration, this component is set as a docker image with data source protocol to be determined to figure out the optimal strategy of delivering the data.

Integration in the prototype. Currently, none of the proposed solutions are integrated with the DFL framework. However, if other people need our component, it will be made available through the GMR. to be made available ensure traceability and reproducibility of these solutions, they will be stored in the centralised storage unit GMR. This allows for idea of a centralised storage and access of GMR

2.4.3.2 Explainability and Confidence Metrics

Transparency and accountability are central pillars of trustworthy AI, particularly in communication networks where AI/ML models are deeply embedded into operational workflows. Although such systems deliver substantial performance gains across multiple domains, they often operate as opaque decision mechanisms, limiting user confidence and complicating oversight processes. XAI aims to reduce this opacity by improving interpretability, clarifying model behaviour, and strengthening reliability in complex and distributed environments. In practice, AI systems must not only generate predictions but also provide insight into the reasoning behind those predictions and express when their outputs are uncertain.

This module addresses the explainability gap through two complementary strategies, each consisting of dedicated services that enhance transparency, reliability, and accountability.

The first strategy focuses on uncertainty quantification and confidence-aware explanations. The initial service develops model-agnostic confidence metrics based on conformal prediction. Conformal prediction provides statistically valid uncertainty guarantees without assumptions on the underlying model or data distribution, making it suitable for heterogeneous networked infrastructures. The proposed Conformal Prediction Confidence Factor (CPCF) quantifies prediction reliability across tasks and highlights how the introduction of new data or learning objectives affects previously acquired knowledge. By producing calibrated prediction sets and dynamic confidence intervals, this service translates abstract predictive uncertainty into interpretable and actionable metrics.

The second service introduces a model-specific confidence estimation mechanism based on Variational Autoencoders (VAEs). By leveraging latent space representations, this approach evaluates the proximity of

new inputs to previously learned patterns and generates reliability indicators that complement conformal uncertainty measures.

The third service extends the SHIELD (Selective Hidden Input Evaluation for Learning Dynamics) regularization framework by incorporating Bayesian neural network principles through the IVON (Improved Variational Online Newton) optimizer. This enhancement enables probabilistic interpretability, offering feature-level relevance insights alongside associated confidence levels. As a result, network operators gain not only information about influential inputs, but also an understanding of the stability and uncertainty of these explanations.

The second strategy emphasizes explanation-driven model refinement and post-hoc interpretability. The first service applies XAI-guided feature selection to optimize security models by retaining the most informative inputs while reducing noise and dimensionality. The second service generates structured, human-readable incident reports that explain model decisions in operationally meaningful terms. These services may operate jointly, producing explanations for optimized models, or independently, depending on deployment requirements.

Beyond individual prediction-level transparency, the Explainable AI module is designed to interoperate with the Fair AI functionality to strengthen system-level accountability. While explainability techniques clarify why decisions are made and how confident the model is in those decisions, fairness auditing evaluates whether such decisions exhibit systematic disparities across protected groups. The Fair AI functionality frames compliance as a tolerance-aware sequential hypothesis testing problem, assessing whether predefined fairness metrics (e.g., Statistical Parity or Equal Opportunity) remain within regulatory thresholds under realistic access constraints. In this way, transparency extends from interpreting single outputs to statistically validating population-level behaviour.

Together, these elements establish a comprehensive framework for explainable and accountable AI. The module produces not only optimized and interpretable models but also calibrated confidence indicators and structured explanation artifacts. These outputs are registered in the GMR, enabling versioned storage of models alongside their uncertainty metrics, explanation reports, and fairness audit results.

The core objectives of the Explainable AI module are therefore:

- Enhancing algorithmic transparency in distributed computing environments through the implementation of interpretability-first design principles.
- Developing explainable ML architectures suitable for deployment across different networked infrastructures.
- Optimizing security models by employing XAI-driven feature selection to reduce data dimensionality, remove noise, and enhance computational efficiency.
- Providing post-hoc, model-agnostic explanations for security incidents by analysing the features that drive a model's predictions.
- Producing standardized, machine-readable, and human-understandable metrics (e.g., JSON-based incident reports) for the Global Model Repository to ensure interoperability.
- Improving network operator trust and enabling more informed, rapid decision-making by translating complex model behaviour into actionable insights.
- Contributing to fair and accountable AI by coupling explanation mechanisms with statistically grounded fairness verification.

Integration in the prototype. Currently, the proposed service is not directly integrated with the prototype. Instead, it trains models independently and extracts explainability metrics, which are then stored together with the trained models in the GMR to ensure traceability, reproducibility, and centralised access. These models and their associated metrics can later be integrated into the overall system architecture if required by other components or stakeholders.

2.4.3.3 Explainability Trade-offs

XAI has emerged as an important component of modern AI/ML systems, particularly in mission-critical domains where transparency and accountability are required. XAI techniques aim to provide insights into how ML models generate predictions, enabling human operators to understand and validate automated decisions. This transparency is especially important in complex systems where black-box models are widely used, and decisions must be interpretable to ensure trust and reliability.

However, increasing transparency through explainability can introduce new security and privacy risks. When explanations reveal important features or decision patterns of a model, adversaries may exploit this information to conduct adversarial attacks. For example, explanation outputs can help attackers understand which input features most strongly influence a model's prediction, enabling more effective evasion attacks or facilitating model extraction attempts. In addition, repeated access to predictions and explanations through exposed interfaces may allow attackers to infer characteristics of the underlying training data. Such attacks may lead to leakage of sensitive information, including proprietary datasets or personal data used during model training.

Privacy concerns are particularly significant when explanations reveal relationships between input data and model outputs. Detailed explanations may allow adversaries to perform membership inference or data reconstruction attacks, potentially exposing whether a specific data sample was used during training. Consequently, improving explainability may inadvertently increase the risk of privacy leakage, especially in systems that provide explanations as part of AI-as-a-Service interfaces.

These challenges highlight the inherent trade-off between explainability, security, and privacy in AI-driven systems. While explainability improves transparency and trustworthiness, excessive disclosure of model information can expand the attack surface and expose sensitive data. Conversely, strict privacy protection mechanisms may limit the fidelity or usefulness of explanations. Therefore, secure AI system design must carefully balance these objectives by incorporating privacy-preserving explanation techniques, robust model training strategies, and secure mechanisms for delivering explanations. Achieving such a balance is essential for ensuring trustworthy and resilient AI systems in future network and intelligent infrastructure environments.

2.4.4 Explainability Module Offered Services

The Explainability module, will include four main services. The description of this module is as follow:

1. The arrival of incoming requests processed through the ROBUST-6G Network Layers, defining explainability requirements and providing the operational context for subsequent model deployment and assessment procedures.

2. The execution of model selection processes within the XAI framework, determining optimal architectures based on interpretability constraints and performance requirements.

3. The implementation of model training protocols enhanced with transparency-preserving algorithms and SHIELD-based regularization techniques, incorporating context-aware training optimization to ensure explainability is embedded throughout the learning process.

4. The generation of model outputs serving as the primary inference results, which subsequently feed into the explainability assessment pipeline for comprehensive interpretability evaluation.

5. The deployment of the Explainability Assessment component, which orchestrates a multi-faceted evaluation by coordinating the following services and analyses:

    a. Conformal Prediction-based Uncertainty Quantification: Implements the CPCF methodology to assess prediction reliability, providing dynamic confidence intervals to provide explanations for the quantification of the model's output confidence.

b. Latent Space-based Confidence Estimation: Implements a VAE methodology to assess prediction confidence through the proximity of the unknown representation and the training points on the Latent Space regarding the Mahalanobis distance.

c. Regularization-derived Explainability Metrics: Applies SHIELD techniques to evaluate selective hidden input evaluation to ensure transparent decision-making processes while enhancing overall model performance.

d. Feature Selection for Enhanced Efficiency Predictive Performance: Employs XAI-based techniques to identify and select the most critical input features, which improves model efficiency and predictive power while providing a clear understanding of the data driving the outcomes through an incident report.

e. SHAP-Based Incident Report & Analysis: Utilizes SHAP to generate detailed reports that explain individual predictions, offering precise, quantifiable insights into how each feature contributed to a specific result.

f. Fairness-aware Accountability Assessment**:** Extends interpretability from individual predictions to population-level behaviour by incorporating fairness compliance verification under predefined tolerance thresholds. Through a sequential, tolerance-aware statistical auditing procedure, the system evaluates whether group-level disparities (e.g., Statistical Parity or Equal Opportunity) remain within regulatory limits under realistic access constraints.

3 Use Case and Integration

This section details the pragmatic valorisation of the operational workflows introduced in section 2.2. The cornerstone here is the GMR, which acts not merely as a storage facility, but as the central "hub" that asynchronously interconnects the results of DFL training and analysis with the global architecture of the ROBUST-6G system, unifying intelligence storage.

3.1 Usage example – AI pipeline, for example using generic MNIST database

To validate the final development of this technical section, Figure 14 shows the complete AI pipeline representing the native and internal operations of the DFL prototype is executed, using an emulated cluster training with the MNIST dataset for demonstrative purposes.

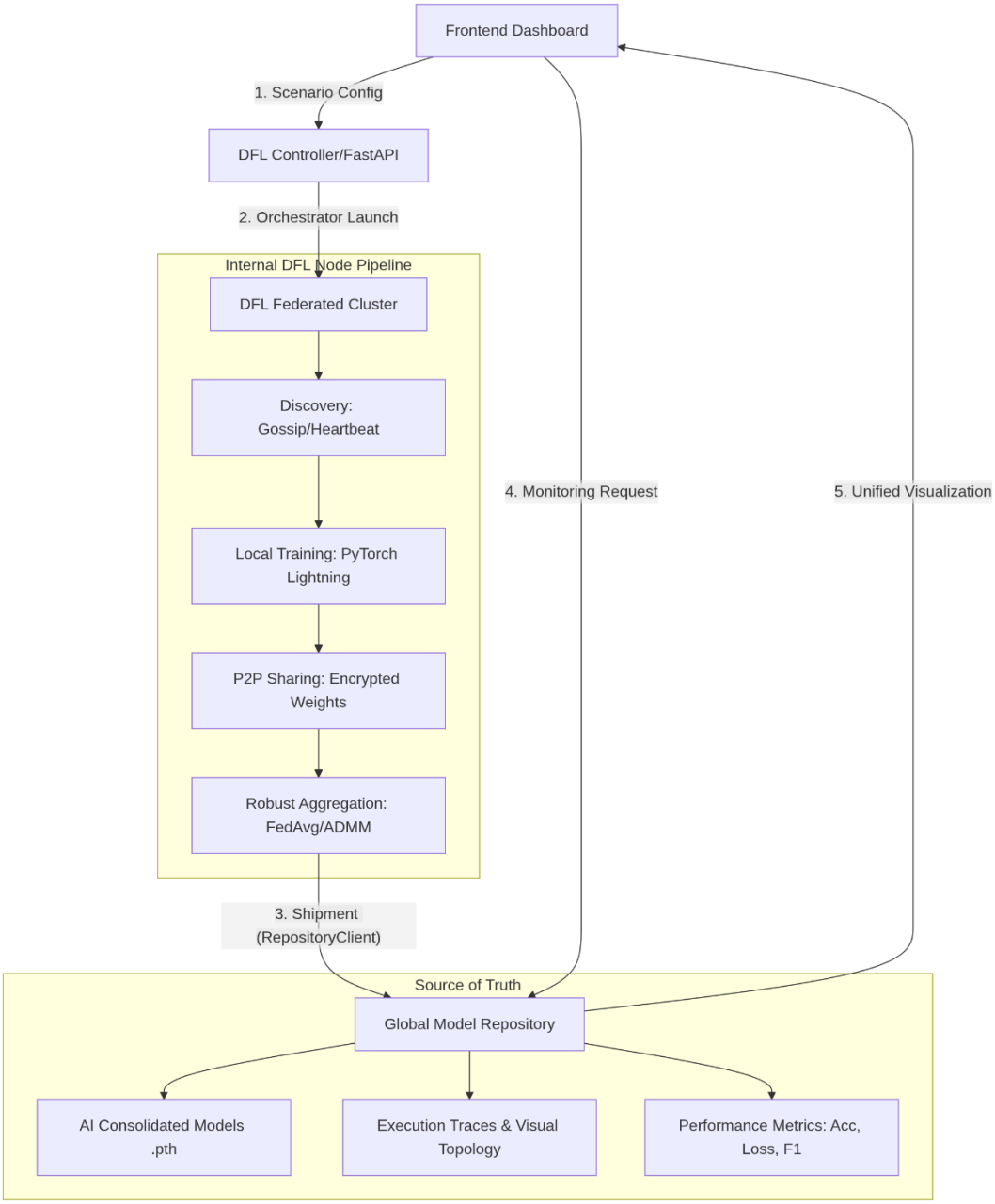

Figure 14 AI Pipeline

As the decentralized execution progresses across multiple nodes, the framework dynamically produces key components that act as the primary source of intelligence in the network:

- AI Models: The consolidated binary weight files, produced by federated aggregation, are serialized and stored in the secure collections of the GMR, making them immediately available for inference tasks.
- Execution Traces (Logs): Complete performance logs, debugging reports, visual representations of class distribution in the cluster, and the network topology are injected in real-time into the servers for auditing, if necessary.
- Global Performance Metrics: The baseline accuracy, loss function, and complex scores such as F1, Recall, and Precision are evaluated and packaged in metadata attached to the GMR storage entities. This allows the cluster state to be asynchronously queried with maximum reliability.

To illustrate these capabilities with concrete experimental results, Figure 15 shows the learning curves obtained during this validation experiment. In this emulated scenario, the workload was distributed across 3 nodes, simulating a fully decentralized training process.

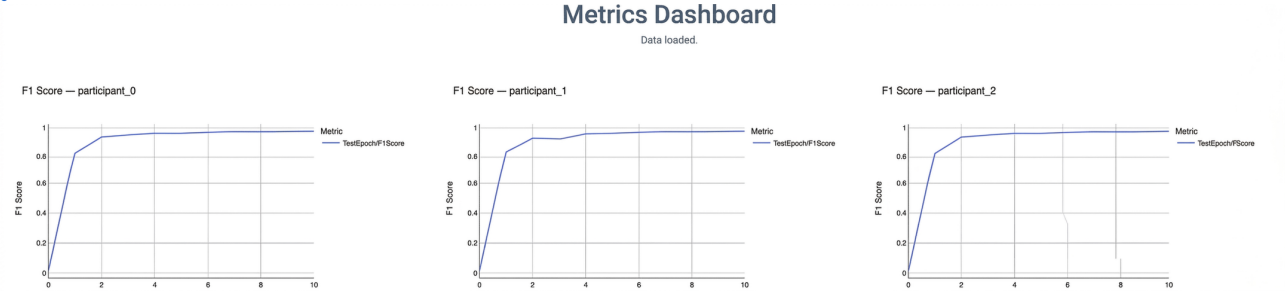

Figure 15 Test Epoch F1-Score curves across the participating nodes over 10 federated communication rounds

As observed in the graph, the global model converges stably after 10 federated aggregation rounds, achieving a global baseline accuracy of 97.49% and minimizing the test loss function to 0.0757. Following the pipeline's logic, the global performance metadata packaged in the GMR yielded the following consolidated test metrics for the final aggregated model:

- Precision: 97.54%
- Recall: 97.49%
- F1-Score: 97.50%

These validation scores demonstrate that the DFL prototype successfully coordinates the decentralized nodes to generate a robust model without compromising the pipeline's operational integrity or the privacy of local data.

3.2 (External) Integration of Selected XAI Measures in the ROBUST-6G Architecture

To satisfy the rigorous demands of European standards regarding the validation of transparent "Trustworthy AI", having a black-box model is insufficient. Therefore, the project evaluates the weights deposited in the GMR by connecting them to an external Explainable Artificial Intelligence pipeline (see Figure 16) enabled by the shats_explainer.py module [FPM25]. This module acts as an auxiliary component that integrates with the prototype's outputs.

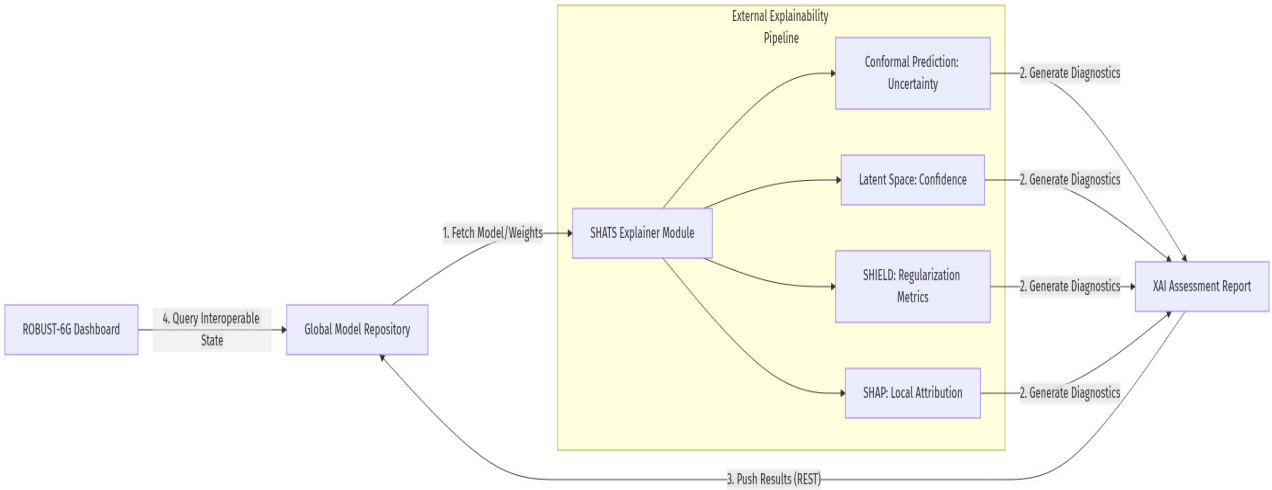

Figure 16 XAI pipeline

To achieve this, the Explainability Assessment component orchestrates a multi-faceted evaluation by coordinating the following services:

- Conformal Prediction-based Uncertainty Quantification: Implements the CPCF methodology to assess prediction reliability.

- Latent Space-based Confidence Estimation: Implements a VAE methodology to assess prediction confidence through pattern proximity in the Latent Space.
- Regularization-derived Explainability Metrics: Applies SHIELD techniques to ensure transparent decision-making processes.
- Feature Selection for Predictive Performance: Employs XAI-based techniques to identify and select critical input features.
- SHAP-Based Incident Report & Analysis: Utilizes SHAP [FPM25] to generate detailed reports.
- Fairness-aware Accountability Assessment: Incorporates fairness compliance verification to the interpretability pipeline.

Once evaluated, these diagnostics and visual explainability results are pushed back into the GMR ecosystem, intertwined with the base model to which they belong. From an architectural lens, the centralized repository significantly facilitates access to this information via REST by external interfaces, the Trustworthiness Evaluator, or the ROBUST-6G dashboard, enabling unified monitoring. The cybersecurity and reliability policies find their final unique catalogue (Source-of-truth) in the GMR, ensuring a highly interoperable model and the uninterrupted fostering of an explainable ecosystem.

3.2.1 Example of Visual XAI Artifact: SHAP Image Plot

To illustrate the practical output of this module and directly addressing the need for concrete assessment examples Figure 17 presents a detailed visual artifact generated for the prototype using the MNIST dataset.

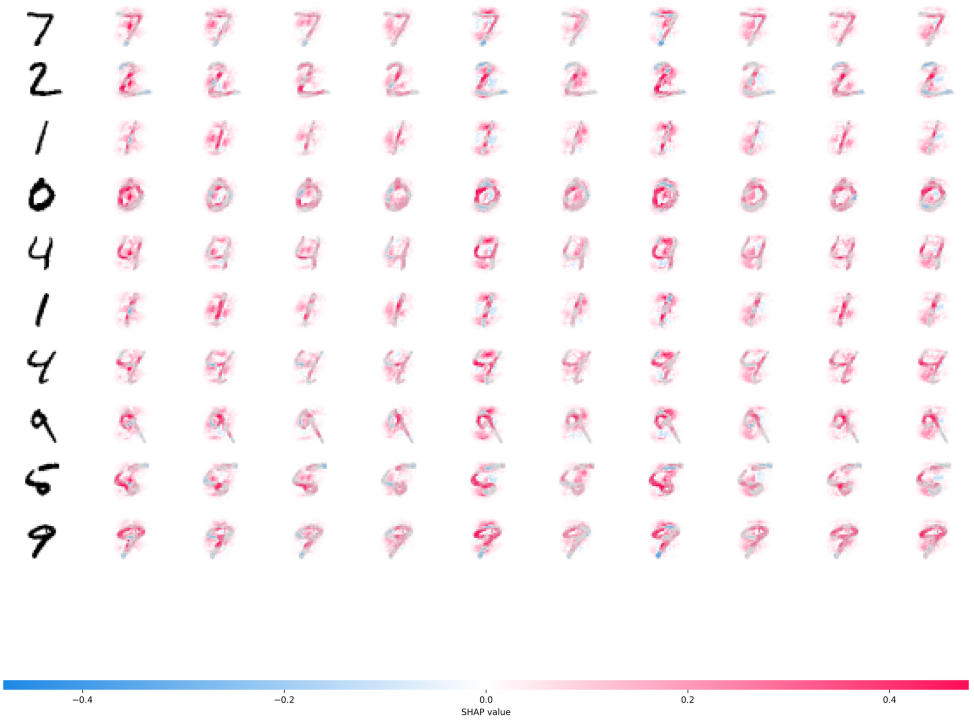

Figure 17 Local explainer artifact detailing the per-pixel SHAP contribution across 10 classes for multiple MNIST test samples

As part of the SHAP-Based Incident Report & Analysis service, this visual tool provides instance-level transparency. The visualization displays the raw input digit on the leftmost column. For each of the subsequent 10 columns (representing output classes 0 through 9), pixel-level SHAP values are overlaid.

Utilizing a localized color scale ranging from -0.1 (blue, negative contribution) to +0.1 (pink/red, positive contribution), the GMR repository now stores precise spatial maps detailing which pixel clusters (features) drove the model's reasoning. For example, looking at the first row (the digit '7'), the model identifies the top horizontal stroke (marked in red in the correct class column) as the definitive feature for that classification.

In contrast, other columns might show heavy blue overlay on those same features, indicating why the model rejected classifications like '1' or '4' for that specific sample.

By integrating these explicit diagnostics and visual metrics back into the centralized repository, the ROBUST-6G architecture ensures that both human operators and automated external interfaces (ROBUST-6G dashboard or Trustworthiness Evaluator) can instantly audit the rationale behind specific inferences as mandated by European standards for transparent and dependable AI ecosystems.

4 Fulfilment of DoA Objectives and KPI Validation

This section provides a consolidated assessment of WP3's achievements in relation to the commitments defined in the Description of Action (DoA), as well as the additional technical contributions delivered during the project. As Deliverable D3.4 represents the final release of the WP3 prototype, this section serves a complementary purpose, which is to demonstrate, in a traceable and measurable manner, that the quantitative objectives assigned to WP3 have been achieved based on the implemented architecture and experimental validation results. Table 3 summarises the fulfilment status of all DoA objectives relevant to WP3, explicitly linking each objective to the implemented mechanisms and validation evidence presented in this deliverable.

Table 4 Summary of objectives as specified in DoA

#	DoA Target	Validation description	Status
1	The final collaboratively trained AI/ML models shall achieve a composite trustworthiness score of ≥ 80%.	**Demonstrated in Section 2.4.1.1** The HE-based poisoning detection mechanism effectively isolates compromised clients, restoring model performance close to the benign baseline and maintaining high trust during aggregation (refer to Table 1). The evaluation of robustness for this KPI is ongoing and will be completed within the integration activities in WP6. This KPI is specific to Use Case 1, for which the final validation and consolidation of results are planned in final deliverable of WP6.	Partially achieved
2	The federated model must demonstrate an average accuracy improvement of ≥ 5% over standalone models	**Covered in Section 2.3.1. By incorporate.** To validate this KPI, a baseline non-aggregation scenario was executed wherein clients trained standalone models exclusively on their local, non-IID datasets, reflecting a highly fragmented and isolated learning environment. These individual local models were then evaluated against a common global test set to establish a baseline performance metric. Subsequently, the collaboratively aggregated DFL model was evaluated on the identical global test set. The comparative analysis confirms that the collaborative framework successfully overcomes the limitations of isolated, non-IID data, achieving a clear accuracy improvement over standalone models that meet the targeted threshold.	Achieved
3	The AI/ML models must achieve a minimum robustness score of ≥ 85% against a defined set of adversarial attacks.	**Detailed in Sections 2.3.4.** The system successfully simulated adversarial vectors natively in the DFL framework (e.g. Model Poisoning). It achieved strong detection accuracy against multiple attack types (Gaussian/uniform noise, targeted top K), validating strict adversarial resilience. The mitigation and evaluation of robustness for this KPI are ongoing and will be further refined and fully validated through the planned integration activities in WP6.	Partially achieved

4	Reduce time taken to share model updates to under 30 minutes, with 95% of the community acknowledging updates within that time frame.	**Covered in Sections 2.4.1.1 and 3.1.** The processing time for handling encrypted updates introduces minimal latency (~1.96 seconds for 20 clients). Aggregated model updates are immediately serialized and made available via the GMR, completing the synchronization loop well within the defined threshold.	Achieved
5	Target an average percentage of improvement of at least 5% in ML/DL model accuracy after incorporating trust-based updates.	**Covered in Section 2.4.1.1 By incorporate.** To validate this KPI, a baseline scenario was first established by training a global model using a vanilla FL setup, without incorporating trust-based updates. The resulting global model was evaluated on a common global test set to define the baseline performance metric. Subsequently, an enhanced federated learning model—integrating secure aggregation along with poisoning detection and removal mechanisms—was trained and evaluated on the same global test set. The comparative analysis demonstrates that incorporating the proposed trust-based updates leads to a measurable improvement in model performance, with accuracy increasing by at least 6.6%.	Achieved
6	Decrease membership inference attack precision from ~50% to ~20% and evasion-based adversarial attack success rate down from ~60% to ~20%	**Covered in Section 2.4.1.3.** Drop in the membership inference accuracy from a peak of 87.5% to 40% (47.5%) and 96.7% to 46.7% (50% drop). However, the attack accuracy was not dropped to a minimum of ~20%. Yet, >45% reduction in the attack success rate is obtained for both cases, where the adversary is unable to infer the membership state as accurate as the random guessing accuracy. **Covered in D3.2.** Drop in the Black-box evasion attack success rate from an overall average of about 88.8% to 14.8%.	Partially achieved
7	Decrease the overall energy consumption of the AI-based security solutions up to 10 times and aim at complete carbon neutrality when renewable resources are available.	**Description in Section 2.4.2:** Specifically, Task 3.3 partners worked on SNNs and energy-aware scheduling for federated learning, including favouring clients powered by renewables. The results published in the cited papers demonstrate that the KPI is achieved.	Achieved
8	Keeping the trade-off between various trustworthiness metrics and model performance within 10% (i.e. Not to decrease natural performance more than 10% while maintaining the robustness of the AI system or lowering the energy consumption of the AI system up to 10% without decreasing model performance by 10%)	All of the conducted experimental results for the proposed methods in WP3 demonstrate that enhancing the robustness, privacy, and sustainability of the AI system results in a performance degradation of no more than 10%.	Achieved

5 Conclusion and Perspectives

In this deliverable, the final outcome of the WP3 technical work has been presented. Building on the foundations established in D3.2 and on the first integrated prototype reported in D3.3, D3.4 consolidates the final design of the ROBUST-6G trustworthy, sustainable, and explainable AI prototype and assesses its readiness, feasibility, and resilience for integration into future 6G environments. The deliverable brings together the architectural evolution, the final prototype realization, the module-level technical contributions, and the deployment-oriented assessment into a coherent view of the WP3 results.

A central conclusion of this final phase is that the DFL framework has evolved from an optional training mechanism into the core operational backbone of the WP3 prototype. In the final architecture, model training is conducted within the DFL framework, while trustworthiness, sustainability, and explainability are embedded directly into the learning lifecycle rather than being treated as external add-on functions. In parallel, the GMR has assumed a strengthened role as a shared AI asset management and governance layer, ensuring traceability, reproducibility, accessibility, and interoperability of models, metrics, and explainability artifacts across the ROBUST-6G architecture.

The final prototype therefore represents more than an aggregation of individual technical components. It constitutes a consolidated AI-native security enabler aligned with the main architectural directions currently emerging for future 6G systems, including distributed intelligence, trustworthiness by design, sustainability as a structural objective, and programmability through modular building blocks. Within this integrated framework, the Trustworthy AI module contributes privacy-preserving and attack-resilient learning mechanisms, the Sustainable AI module contributes scalable aggregation, energy-aware optimization, and sustainable models by design, and the Explainable AI module contributes model enhancement, uncertainty-related measures, incident reporting, interpretability, and accountability-oriented assessment. Taken together, these contributions strengthen the robustness, transparency, and operational viability of AI-enabled security mechanisms in distributed 6G settings.

The deliverable has also demonstrated the practical integration perspective of the final WP3 prototype. The internal AI-pipeline example showed how models, execution traces, and performance metrics can be produced and stored through the DFL-GMR workflow, while the external integration path for selected XAI measures showed how explainability diagnostics can be computed and returned to the GMR as part of a broader monitoring and governance process. In addition, the fulfilment and KPI assessment section provides a consolidated view of WP3 achievements with respect to the DoA commitments, and the broader technical contributions delivered during the project.

From a forward-looking perspective, the work reported in D3.4 indicates that future AI-native 6G systems should not treat explainability, trustworthiness, and sustainability as isolated validation layers applied after model development. Rather, these properties should be designed jointly and assessed throughout the full AI lifecycle, from distributed training and secure aggregation to model storage, deployment, monitoring, and accountability assessment. The results of WP3 therefore provide a concrete basis for future research and integration activities in areas such as tighter coupling between XAI and operational orchestration, broader support for interoperable AI-governance interfaces, more systematic treatment of the trade-offs between explainability and performance, and extension of the proposed mechanisms to larger-scale and more heterogeneous 6G environments.

Overall, D3.4 serves as the final WP3 integration and assessment report, while also defining a practical perspective for the continuation of this line of work beyond the present prototype. The lessons learned in architecture design, prototype implementation, integration, and validation provide both a final consolidation of WP3 results and a foundation for the feasible integration of selected XAI measures into future trustworthy, sustainable, and AI-native 6G security architectures.

References

[JPY+26] E. Jeong, G. Perin, H. H. Yang, and N. Pappas, “Feature-Based Semantics-Aware Scheduling for Energy-Harvesting Federated Learning,” in Proc. of 2026 IEEE ICMLCN, Abu Dhabi, UAE, April 2026.

[PJP26] G. Perin, E. Jeong, and N. Pappas, “Federated Learning Meets Random Access: Energy-Efficient Uplink Resource Allocation,” submitted to 2026 IEEE ICC Workshops, Glasgow, UK, May 2026 (arXiv preprint available arXiv:2602.01913).

[PBM+26] G. Perin, C. Bidini, R. Mazzieri, and M. Rossi, “ADMM-Based Training for Spiking Neural Networks,” in Proc. of 2026 IEEE PerCom Workshops, Pisa, Italy, March 2026.

[APM+25] Y. Abdennadher, G. Perin, R. Mazzieri, J. Pegoraro, and M. Rossi, LightSNN: Lightweight Architecture Search for Sparse and Accurate Spiking Neural Networks, in Proc. of 2025 AMLDS, pp. 84-89, Tokyo, Japan, July 2025.

[ACR25] Y. Abdennadher, E. Cicciarella, and M. Rossi, Convolutional Spiking-Based GRU Cell for Spatio-Temporal Data, in Proc. of 2025 IEEE MLSP, Istanbul, Turkey, September 2025.

[JP25a] E. Jeong and N. Pappas, “Battery-Aware Cyclic Scheduling in Energy-Harvesting Federated Learning,” 2025 IEEE 26th International Workshop on Signal Processing and Artificial Intelligence for Wireless Communications (SPAWC), Surrey, United Kingdom, 2025, pp. 1-5, doi: 10.1109/SPAWC66079.2025.11143381.

[JP25b] E. Jeong and N. Pappas, “Computation-aware energy-harvesting federated learning: Cyclic scheduling with selective participation,” arXiv preprint arXiv:2511.11949, 2025.

[RBG+22] B. Rueckauer, C. Bybee, R. Goettsche, Y. Singh, J. Mishra, and A. Wild, “NxTF: An API and compiler for deep spiking neural networks on Intel Loihi,” in ACM Journal on Emerging Technologies in Computing Systems (JETC), 18(3), 1-22, 2022.

[FPM25] M. Franco de la Peña, Á. L. Perales Gómez, and L. Fernández Maimó, "ShaTS: A Shapley-based Explainability Method for Time Series Artificial Intelligence Models applied to Anomaly Detection in Industrial Internet of Things," arXiv preprint arXiv:2506.01450, 2025.

[LSM01] Shapley regression values: Lipovetsky, Stan, and Michael Conklin. "Analysis of regression in game theory approach." Applied Stochastic Models in Business and Industry 17.4 (2001): 319-330.