Ref. Ares(2026)7544544 - 31/07/2026

SERBIAN ARTIFICIAL INTELLIGENCE FACTORY ANTENNA

WP3
Integrated HPC and AI runtime environment

Design of the AI Gateway

Deliverable due date	31/07/2026
Deliverable submission date	31/07/2026

Dissemination level (marked with "X")		
PU	Public, to be freely disseminated	X

EuroHPC Joint Undertaking

Project metadata

Project Acronym	SAIFA
Project Title	Serbian Artificial Intelligence Factory Antenna
Project Website	https://www.saifa.rs
Grant Agreement no.	101263280
Call identifier	HORIZON-JU-EUROHPC-2025-AIFA-01
Type of Action	HORIZON-JU-RIA
Project duration	May 1st, 2026 – April 30th, 2028 (36 months)
Coordinator	School of Electrical Engineering, University of Belgrade (ETF)

Document metadata

Deliverable no.	D3.1
Deliverable title	Design of the AI Gateway
Related WP no.	WP3
Related WP title	Integrated HPC and AI runtime environment
Lead beneficiary	School of Electrical Engineering, University of Belgrade (ETF)
Contributors	ETF, IPB, IMP, IVI, FSUKG, IMGGE
Deliverable type	R – Report; dissemination level PU – Public

Document revision history			
Version/name	**Date**	**Institution**	**Author(s)**
0.6 – source consolidation	22/07/2026	ETF	Working team
0.7 – updated version	29/07/2026	ETF	ETF, IPB, IMP, IVI, FSUKG, IMGGE
0.8 – final version	31/07/2026	ETF	ETF, IPB, IMP, IVI, FSUKG, IMGGE

COPYRIGHT

© 2026 Members of the SAIFA Consortium

DISCLAIMER

The information in this document reflects only the author's views and the European Community is not liable for any use that may be made of the information contained therein. The information in this document is provided "as is" without guarantee or warranty of any kind, express or implied, including but not limited to the fitness of the information for a particular purpose. The user thereof uses the information at his/ her sole risk and liability.

List of Abbreviations

ABAC	Attribute-Based Access Control
ACL	Access Control List
AI	Artificial Intelligence
API	Application Programming Interface
ART	ART sequencing-read simulator (software name)
BFF	Backend for Frontend
BI	Business Intelligence
CD	Continuous Delivery / Continuous Deployment
CE	Community Edition
CI	Continuous Integration
CI/CD	Continuous Integration and Continuous Delivery/Deployment
CKAN	Comprehensive Knowledge Archive Network
CPU	Central Processing Unit
DCAT	Data Catalog Vocabulary
DCAT-AP	Data Catalog Vocabulary Application Profile
DR	Disaster Recovery
EEG	Electroencephalography
EV	Electric Vehicle
FAIR	Findable, Accessible, Interoperable and Reusable
FTS	Full-Text Search
GDPR	General Data Protection Regulation
GPU	Graphics Processing Unit
HPC	High-Performance Computing
IAM	Identity and Access Management
ID	Identifier
IDE	Integrated Development Environment
IoT	Internet of Things
IS	Identity Server
IT	Information Technology
LLM	Large Language Model
LMS	Learning Management System
MLOps	Machine Learning Operations
OIDC	OpenID Connect
OPA	Open Policy Agent
OSI	Open Source Initiative

PEFT	Parameter-Efficient Fine-Tuning
PoC	Proof of Concept
RBAC	Role-Based Access Control
RIA	Research and Innovation Action
RNA	Ribonucleic Acid
RPO	Recovery Point Objective
RTO	Recovery Time Objective
S3	Simple Storage Service (S3-compatible object storage)
SAML2 / SAML 2.0	Security Assertion Markup Language 2.0
SLURM	Slurm Workload Manager (originally Simple Linux Utility for Resource Management)
SME	Small and Medium-Sized Enterprise
SSH	Secure Shell
SSO	Single Sign-On
TGI	Text Generation Inference
TRL	Transformer Reinforcement Learning (Hugging Face library)
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience
WORM	Write Once, Read Many

Executive Summary

Context. The Serbian AI Factory Antenna (SAIFA) is a HORIZON-JU-EuroHPC project (Grant Agreement No. 101263280) that establishes a national entry point to artificial intelligence and high-performance computing for research and clinical institutions, industry, the public sector and individual users. The AI Gateway is developed in Work Package 3, Task 3.4, as the unified access point for AI resources, tools and services. Deliverable D3.1, due in M03, provides the comprehensive initial plan for its development.

Scope. The design covers access to computing environments, datasets, pre-trained models, applications, reusable tools, modular workflows, training materials and support services. It also addresses authentication, resource scheduling, monitoring, secure sharing and interoperability with national infrastructure, Pharos, IT4LIA, the AI-on-Demand Platform and other AI Factory services. The design sets out a target direction and an evaluation process. It does not predetermine the final implementation scope, architecture or division of responsibilities, nor does it commit the project consortium to delivering every described capability or workflow in the beta or subsequent releases.

Design approach. The Gateway is conceived as a modular integration and control layer rather than a monolithic platform. Mature components are reused for identity management, cataloguing, notebooks, model management, execution, storage, monitoring and learning. SAIFA-specific development is concentrated on the unified portal, common metadata and authorisation rules, quota and accounting logic, the HPC bridge, audit, and federation adapters.

Users and use cases. The functional scope is illustrated through 21 Domain Use Stories covering language and culture, biomedicine and health, energy and manufacturing, environment, and cross-domain adoption. A representative set of system use cases specifies the principal interactions for authentication, discovery, data and model management, compute jobs, inference, experiments, training, federation and support. These descriptions remain subject to validation and prioritisation with domain owners, infrastructure operators and user-support teams.

Technology and risk. Candidate technologies are evaluated through short proofs of concept, mandatory elimination criteria and comparative assessment of usability, integration effort, operational complexity, performance, maturity, compliance and sustainability. The accompanying risk register combines proposal-level risks with Gateway-specific security, integration, data-integrity and operational risks, and links them to preventive controls and acceptance evidence.

Roadmap. The contractual path is D3.1 in M03, access to national infrastructure in M03, access and use policies and the online beta version in M12, and D3.4 - AI gateway functionalities and features - in M18. Task 3.4 continues through M36, covering further integration, hardening, interoperability, validation and operational improvement. Development and preliminary testing are carried out on the PARADOX and ITE infrastructures, followed by validation and benchmarking using the Greek and Italian AI Factory infrastructures.

Outcome. Together, these documents form a coherent design baseline that links the project structure, domain needs, system behaviour, candidate technologies, delivery plan and risk controls. The baseline supports implementation while preserving the flexibility needed to incorporate validated user requirements and evidence from the planned proofs of concept.

Table of Contents

1. Introduction 9
   1.1. Purpose and scope of the deliverable 9
   1.2. Relationship with other work packages and tasks 9
   1.3. AI Gateway goals and design principles 10
   1.4. Scope boundaries 10
   1.5. Document structure 10
2. SAIFA Structure and the Role of the AI Gateway 11
   2.1. National infrastructure and computing environment 11
   2.2. AI Gateway layer 12
   2.3. Federation with European AI infrastructures 12
   2.4. Target domains and project contributions 13
3. User Stories 14
   3.1. Energy, Manufacturing and Sustainability 14
   3.2. Environment and Sustainability 17
   3.3. Health 19
   3.4. Language and Culture 24
   3.5. Cross-Domain Gateway Adoption 27
4. System Use Cases 31
   4.1. System boundary 31
   4.2. Actors 32
   4.3. Representative system use cases 33
   4.4. Validation status and open design assumptions 47
5. Candidate Tools and Software Packages 49
   5.1. Reusable components and SAIFA-specific integration 49
   5.2. Cross-layer candidate and proof-of-concept baseline 51
   5.3. Technology-selection criteria 51
   5.4. Licensing and procurement 52
   5.5. Recommended decision statement 53
6. Risks and Mitigation Measures 54
   6.1. Assessment and governance 54
   6.2. Initial risk register 54
   6.3. Release and review rule 58
7. Roadmap and Timeline 59
   7.1. Indicative implementation stages 59
   7.2. Contractual deliverables and milestones 60

7.3. Testing and validation plan 61
8. Conclusion 62

List of Figures

Figure 2.1 Overview of the Serbian AI Factory Antenna structure. 11
Figure 2.2 Logical middleware role and responsibility boundary of the AI Gateway. 13
Figure 7.1 SAIFA implementation plan, highlighting WP3 and Task 3.4..... 60

List of Tables

Table 1.1 Practical goals given in Task 3.4. 10
Table 4.1 Human actors 32
Table 4.2 External system actors..... 32
Table 5.1 Candidate components and the corresponding SAIFA-specific integration..... 49
Table 5.2 Comparative technology-selection criteria. 51
Table 5.3 Technology domains and examples of mandatory criteria. 52
Table 7.1 Indicative AI Gateway implementation stages. 59
Table 7.2 Contractual deliverables and milestones relevant to the AI Gateway. 60

1. Introduction

SAIFA (Serbian AI Factory Antenna) is funded under HORIZON-JU-EuroHPC (Grant Agreement No. 101263280; call HORIZON-JU-EuroHPC-2025-AIFA-01; Type of Action HORIZON-JU-RIA). The project is coordinated by the School of Electrical Engineering, University of Belgrade (ETF) and runs for 36 months, from 1 May 2026 to 30 April 2029.

SAIFA establishes a national AI Factory Antenna that connects users in academia, research and clinical institutions, industry, public administration and other user communities with national and European AI resources. National computing is provided primarily through the PARADOX HPC cluster operated by IPB and the AI computing infrastructure operated by ITE, with links to Pharos, IT4LIA and other EuroHPC services.

Within this environment, the AI Gateway is the user-facing integration layer developed in Work Package 3, Task 3.4. It provides a coherent point of access to computing environments, data, models, tools, workflows, training and support while the specialised services remain operated by their respective platforms and work packages.

1.1. Purpose and scope of the deliverable

Deliverable D3.1, Design of the AI gateway, provides the initial comprehensive plan for the development of the component. It consolidates the Gateway role, representative domain and system use cases, candidate technologies, implementation stages, testing approach and risk controls. It also identifies where further validation, proofs of concept and architectural decisions are required.

The document is a design baseline, not a final low-level specification. It does not fix the production stack, detailed API contracts, deployment manifests, operating procedures or the complete beta backlog. The scenarios and capabilities below therefore describe candidate user-facing outcomes rather than implementation commitments. They will be refined during Task 3.4 on the basis of validated needs, infrastructure constraints, security and compliance review, and implementation evidence.

1.2. Relationship with other work packages and tasks

- WP1 provides project governance, quality assurance, risk management, legal and ethical oversight, and the Data Management Plan.
- WP2 identifies stakeholder needs, develops business and sustainability models, and coordinates liaison with national initiatives, AI Factories and AI Factory Antennas.
- T3.1 defines access and use policies; T3.2 provides national infrastructure resources; T3.3 develops and enables computing environments; T3.4 designs, implements and deploys the AI Gateway; and T3.5 addresses technical interoperability with AI Factories and antennas.
- WP4 develops and validates sector-specific AI applications in language and culture, energy and manufacturing, health, and environment. These applications supply domain requirements, workflows, datasets and models to be exposed through the Gateway.
- WP5 develops user-support procedures and training programmes and provides support and training services that the Gateway makes accessible to users.
- WP6 provides communication and dissemination channels for project results and services.

1.3. AI Gateway goals and design principles

The Task description leads to six practical goals for the AI Gateway that are given in Table 1.1.

Table 1.1 Practical goals given in Task 3.4.

Goal	Meaning for the design
Single entry point	Provide one coherent route to AI resources, tools, services, training and support.
User-centred access	Support users ranging from beginners to advanced AI developers through role-appropriate paths.
Experimentation and deployment	Connect data, models, workflows and computing environments across the AI lifecycle.
Secure and efficient operation	Integrate authentication, authorisation, resource scheduling, monitoring and accountability.
Reuse and modularity	Integrate mature platforms and develop custom software only for SAIFA-specific coordination and governance.
Interoperability	Connect national resources with PARADOX, ITE, Pharos, IT4LIA, the AI-on-Demand Platform and other European AI Factory services.

1.4. Scope boundaries

The AI Gateway is an integration and control layer. It does not replace the identity providers, schedulers, computing clusters, object stores, model-serving engines, learning-management systems, repositories or domain applications that it connects. Domain-specific methods, application logic, content and interpretation of results remain the responsibility of connected services, domain teams or authorised user-provided code.

Functions, products and end-to-end scenarios named in this deliverable are candidates unless explicitly identified as contractual milestones or existing infrastructure. Their inclusion does not commit SAIFA to implement all of them, assign them to the Gateway itself, or deliver them in the described form or scope.

1.5. Document structure

- Section 2 describes the SAIFA structure, infrastructure layers, the role of the AI Gateway and European federation.
- Section 3 presents the User Stories.
- Section 4 defines the system boundary, actors and representative System Use Cases.
- Section 5 lists candidate tools and the technology-selection approach.
- Section 6 consolidates risks and mitigation measures.
- Section 7 provides the roadmap, contractual milestones, and testing and validation plan.
- Section 8 concludes the design and identifies the immediate implementation focus.

2. SAIFA Structure and the Role of the AI Gateway

SAIFA is designed as a national single entry point for AI services, computing resources, data, models, tools, training and expert support. The AI Gateway provides the common user-facing access layer, while the underlying capabilities are delivered through the project infrastructure, sector-specific applications, support services and links to European AI Factory resources. This arrangement reduces fragmentation and enables users from research, industry and public administration to follow structured paths from initial exploration to validated AI workflows.

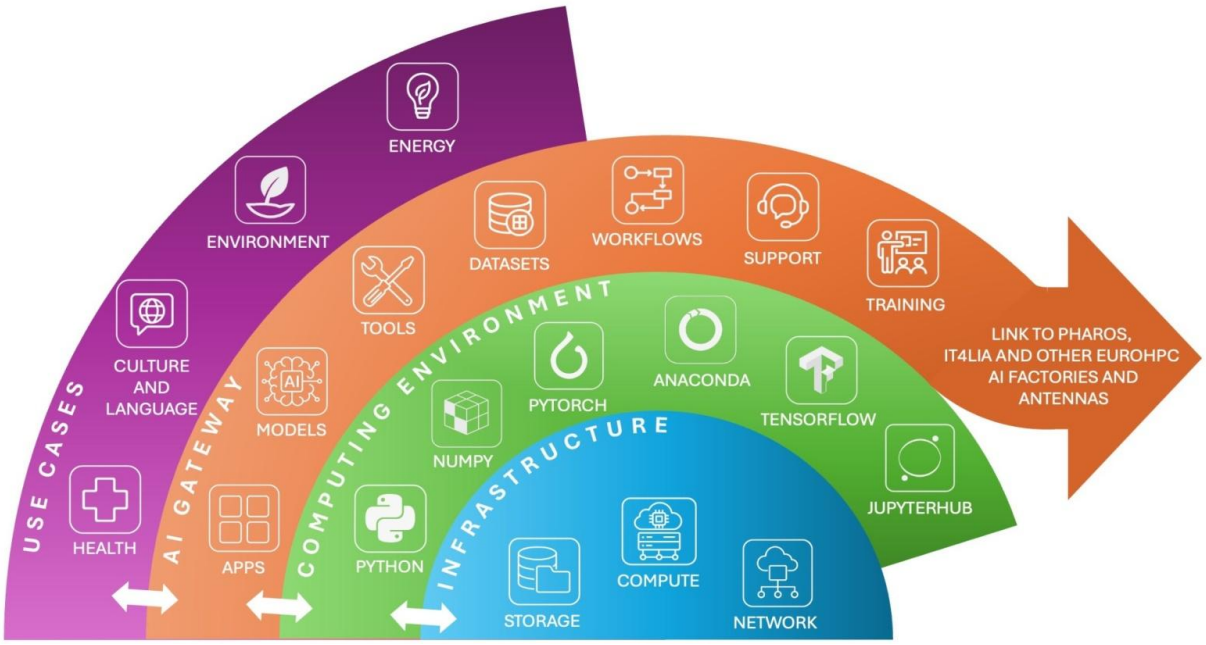


Figure 2.1 Overview of the Serbian AI Factory Antenna structure.

2.1. National infrastructure and computing environment

SAIFA uses the PARADOX HPC cluster operated by IPB and the AI computing infrastructure operated by ITE for development, integration and preliminary testing. Both provide dedicated computing time and storage for project activities. The planned 100 Gbps connection between the IPB and ITE data centres, established with AMRES, supports efficient transfer and synchronisation of project data. Final validation and performance testing will also use the Greek and Italian AI Factory infrastructures in accordance with the access and interoperability arrangements developed in WP3.

Existing scheduling and resource-management systems will be adapted to support priority-aware and workload-aware allocation. Users will submit jobs with defined CPU, GPU, memory and runtime needs, while the underlying schedulers remain responsible for queue management, backfilling, quality-of-service policies and fair use of the available capacity.

The computing environment will support containerised execution so that applications and their dependencies can be reproduced across compatible platforms. Kubernetes-based services and the existing cluster schedulers may be used for different workload types. Curated images for widely used AI frameworks will simplify onboarding and provide repeatable starting environments.

Advanced users may also register approved custom environments when a domain workflow requires specific libraries or tool versions. The approval and execution process must preserve security, reproducibility and compatibility with the selected computing backend.

2.2. AI Gateway layer

The AI Gateway sits above the computing and data infrastructure as middleware: a unified access and integration layer that accepts user or application requests, applies common policies and metadata, and forwards the requests to the appropriate connected service or execution environment. It may expose applications, workflows, reusable tools, pre-trained models, AI-ready datasets, support and training resources without implementing their domain-specific logic.

Typical Gateway responsibilities include identity and access enforcement, catalogue and metadata management, request validation and routing, quota and policy checks, submission and status aggregation, monitoring, audit and provenance, consistent portal/API access and common rules for controlled sharing. Domain applications, workflow engines, model-serving systems, compute backends and authorised user-provided code remain responsible for scientific or industrial algorithms, data preparation, model execution, domain validation and interpretation of results.

2.3. Federation with European AI infrastructures

The Gateway may provide technical interoperability with Pharos, IT4LIA, the AI-on-Demand Platform and other EuroHPC AI Factories and Antennas. Depending on validated partner interfaces and governance conditions, possible integration functions include:

- Catalogue federation. Approved models, datasets, workflows and services may be made discoverable through connected catalogues while preserving origin, version, provenance and usage conditions.
- Metadata and access mapping. Catalogue metadata and access requirements may be translated between SAIFA and partner platforms.
- Validation evidence exchange. Approved testing and performance evidence may be transferred to support interoperability and service improvement.
- Controlled data and model exchange. Synthetic or other publishable assets may be transferred only where legal, ethical, licensing and policy conditions permit.
- Documentation access. Relevant technical documentation and tutorials may be surfaced through the Gateway.
- Training-service access. Relevant partner training resources may be linked where integrations are available.
- External service discovery. Relevant partner services may be surfaced without reproducing their internal functionality.

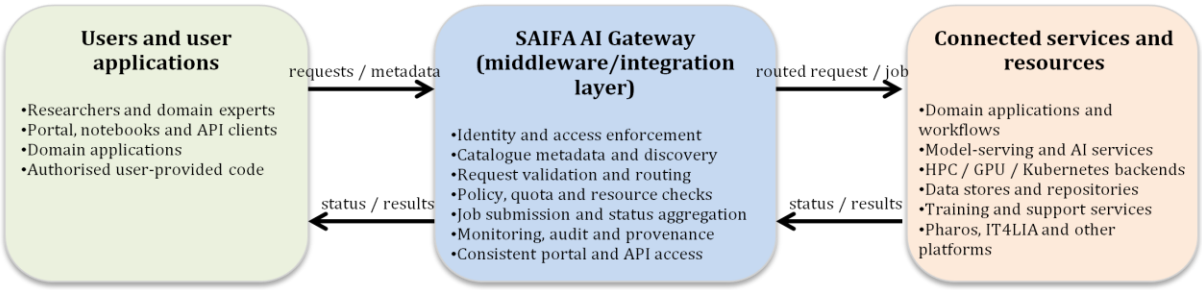


Figure 2.2 Logical middleware role and responsibility boundary of the AI Gateway.

2.4. Target domains and project contributions

The AI Gateway is intended to support four principal application areas: language and culture, energy and manufacturing, health, and environment. These domains provide candidate datasets, models, workflows and applications for validating Gateway integration patterns. Cross-domain services such as cybersecurity, trustworthy AI, training, support and interoperability may apply across all four areas.

3. User Stories

The following User Stories describe representative end-to-end situations in which stakeholder groups use the SAIFA AI Gateway to address scientific, industrial, public-sector or educational needs. They align the Gateway design with the target application areas and with cross-domain support, training and interoperability needs.

The stories are broader than individual Agile user stories and less technically prescriptive than system use cases. Each story identifies the actors, the current need, the intended use of the Gateway, the expected value and the principal Gateway capabilities involved.

Scope note. These stories represent user perspectives and desired modes of interaction rather than commitments to implement every capability. Their inclusion does not imply that all described functions will be delivered, or that any implemented function will follow the exact workflow, allocation of technical responsibility or scope presented here. Depending on the validated architecture, an activity may be performed by the Gateway itself, by another platform component or external service invoked through the Gateway, or by authorised user-provided code. Final scope and implementation responsibility will be determined through requirements validation, prioritisation, architecture decisions and acceptance planning.

The stories must be validated with domain owners and then decomposed into prioritised requirements, system use cases and acceptance criteria.

3.1. Energy, Manufacturing and Sustainability

DUS-EN-01 - Forecasting Renewable Generation, Demand and Load Flexibility

Primary actors: Energy researcher, distribution-system analyst, building or energy-community operator.

Current situation and need

Efficient scheduling of distributed energy resources requires accurate forecasts of renewable generation, consumption and demand-side flexibility. These signals are typically non-stationary and require large datasets and computationally intensive modelling.

Target scenario

An energy analyst creates a project and connects or uploads historical generation, consumption, and weather data. In case of a new upload, the analyst connects the data to an available semantic model to ensure interoperability with existing data processing and model-training workflow architectures.

The user then selects an approved forecasting workflow through the Gateway and specifies the prediction horizon, such as intraday or day-ahead forecasting. The Gateway routes the request, together with the validated dataset and selected configuration, to the appropriate forecasting application.

The forecasting application trains and benchmarks a predefined set of time-series, machine-learning, and deep-learning models, records the experiment parameters and results, and evaluates prediction quality. Through the Gateway, the analyst reviews the resulting forecasts, model-performance metrics, and uncertainty intervals, where available, and may use the selected forecasters within a subsequent scheduling or optimisation workflow.

Validated forecasting models and benchmark datasets may be registered and shared with authorised energy stakeholders.

Expected value and outputs

- renewable energy generation or consumption forecasts for a selected historical period;
- reusable energy datasets and models;
- improved basis for resource scheduling and energy management.

Main Gateway capabilities exercised

Support for time-series data processing, model training, GPU/HPC execution, model registration and version management, and API-based access.

DUS-EN-02 - Optimising Energy Flows in Energy Communities

Primary actors: Energy manager, building operator, energy-community coordinator, mobility or flexibility-service provider.

Current situation and need

Energy operators need to coordinate renewable generation, storage, building consumption and flexible assets within the energy communities while respecting cost, comfort, technical and sustainability constraints.

Target scenario

The user inputs the system configuration, including registering controllable assets, specifying building- or energy-community-level constraints, and uploading any available generation and demand forecasts. The application layer initiates an ontology-based validation process supported by the relevant services. This process maps the configured assets, features, and data properties to the corresponding ontology concepts and relationships and verifies that all required data and parameters are present, semantically consistent, and compliant with the defined constraints.

The Gateway enables the user to select and activate an approved optimization service. The validated configuration and input data are then routed to the appropriate optimisation application, which determines and evaluates alternative operational strategies and energy-flow configurations for the available energy resources. These may include power exchanges with the grid, renewable-energy allocation, storage charging and discharging policies, the dispatch of other controllable assets, while respecting technical, operational, economic, and comfort constraints.

Through the Gateway, the user can access and review the proposed optimisation results according to energy cost, self-consumption, peak demand, emissions, and compliance with operational limits.

Expected value and outputs

- optimised energy flows within the considered energy system, or energy community;
- improved renewable-energy self-consumption;
- more effective use of storage and demand flexibility;
- quantified costs and operational effects.

Main Gateway capabilities exercised

Ontology-based asset and data validation, forecast retrieval, selection of optimisation objectives, and operational policies, routing of validated inputs to the appropriate optimisation application, access to and review of optimisation results

DUS-IND-01 - Surrogate Modelling for Industrial Processes

Primary actors: Process engineer, industrial data scientist, plant energy manager.

Current situation and need

Industrial stakeholders need to maintain production quality and throughput while reducing costs and machine-energy consumption. Direct physical simulation or exhaustive experimentation may be too slow for operational optimisation. Thus, surrogate models can provide sufficiently fast estimates of process behaviour to support optimisation and decision-making

Target scenario

An industrial team uploads or securely connects process measurements, product-quality indicators, equipment states and energy-consumption data.

Through the Gateway, the team selects and activates an approved workflow for developing a surrogate model of the process. The Gateway routes the data and workflow configuration to the appropriate modelling application, which trains and evaluates models for predicting product quality, resource consumption, or potential faults. Where rare failure examples are insufficient, approved synthetic-data methods may be used to enrich the training set, with generated data clearly identified.

Validated surrogate models may then be used by optimisation applications to evaluate and improve control strategies for industrial processes.

Expected value and outputs

- validated surrogate process models;
- predictions of quality, machine-energy load or early fault detection;
- synthetic rare-event datasets.

Main Gateway capabilities exercised

Industrial data ingestion, support for model training and synthetic-data generation

DUS-IND-02 - Multi-Objective Optimisation of Industrial Processes

Primary actors: Process engineer, industrial data scientist, plant energy manager.

Current situation and need

Industrial operators need to identify control strategies that balance production quality, throughput, energy and resource consumption, operating costs, equipment constraints, and process safety. Testing a large number of control configurations directly on the physical process may be costly, slow, or operationally risky.

Target scenario

The user selects a validated surrogate model developed through DUS-IND-01 and defines the controllable process variables, operating limits, and available process and energy data.

The Gateway enables the user to select and activate an approved multi-objective optimisation service and configure the relevant objectives, constraints, and operational policies. These may include maintaining product quality and production throughput while minimising machine-energy consumption, resource use, operating costs, or the risk of process faults.

The validated surrogate model, optimisation configuration, and required input data are routed to the appropriate optimisation application. The application uses the surrogate model to efficiently evaluate process behaviour under different control settings and

identify candidate operating and control strategies that represent suitable trade-offs among the selected objectives.

Through the Gateway, the user can access and review the optimisation results, predicted process performance, and compliance with operational constraints.

Expected value and outputs

- optimised industrial operating and control strategies;
- faster evaluation of process configurations using validated surrogate models.

Main Gateway capabilities exercised

Retrieval and reuse of validated surrogate models, configuration and activation of multi-objective optimisation services, selection of objectives, constraints, and operational policies, routing of models and validated inputs to the appropriate optimisation application, and access to and review of optimisation results

3.2. Environment and Sustainability

DUS-ENV-01 - Development of Digital Twins for Flood Prediction and Prevention

Primary actors: Sector for emergency management and civil protection, public water-management enterprises, local self-government units, national hydrometeorological service, operators of hydrotechnical structures and critical infrastructure, insurance and risk-assessment bodies, digital-twin developers and solution integrators.

Current situation and need

Flood early warning relies mostly on physically based hydrological and hydraulic models, such as those operated within the most referent early warning and alerting systems in Europe. These models are time-consuming and expensive to develop, requiring scarce expert knowledge and numerous parameters that are difficult to establish. Keeping such models current also depends on regular, precise field measurements, which are hard to guarantee given recurring problems with monitoring equipment and data transmission.

Developing a digital twin capable of reliably predicting and helping prevent floods requires moving beyond these physically based approaches towards data-driven, automated machine-learning methods, combined with artificial intelligence and IoT-based sensor networks able to supply the model with continuously updated, high-resolution data. Based on hydrometeorological data collected across a given basin, together with data from such sensor networks, automated machine-learning pipelines can generate and continuously retrain models that predict water levels along its watercourses. Such models can be further improved by incorporating additional data on natural and social factors affecting the system, such as the condition of flood-defence structures and information on the operation of hydrotechnical infrastructure.

Target scenario

A developer building a flood-prediction digital twin signs in to the SAIFA AI Gateway and selects the MetaFOX AutoML service in the catalogue. The Gateway authenticates the user, verifies entitlements and quota, and opens the MetaFOX application running on SAIFA infrastructure. The developer connects historical hydrometeorological time-series and sensor data for the target basin, defines the variable to be predicted, such as water level at a given gauging station, and configures the AutoML job by selecting the model type, optimisation goal, evaluation metric, and termination criteria. Once submitted, the job runs asynchronously on orchestrated compute resources, where MetaFOX automatically performs feature engineering, model selection, and hyperparameter optimisation to produce the best-performing predictive model for the supplied data.

Throughout the run, the developer can monitor job status, progress logs, and the performance of the best model found so far directly through the Gateway, and can pause, resume, or stop the job as needed. Once the AutoML process completes, the optimised model is stored and made available in a standard, deployment-ready format together with its performance metrics and metadata, so it can be registered as a new or updated forecasting model within the flood-prediction digital twin. As new hydrometeorological data accumulates, the developer can resubmit the job to retrain the model, keeping predictions accurate as conditions on the basin change over time.

Expected value and outputs

• earlier, better-founded decisions on declaring flood-defence measures, including for basins without an existing physical model;

• faster, automated development of basin-specific predictive models, driven by AutoML feature engineering, model selection, and hyperparameter optimisation, without requiring specialised machine-learning expertise from digital-twin developers;

• reduced dependence on scarce hydrological-modelling expertise and on the availability of any single gauging station;

• predictive models that can be continuously monitored and retrained on demand as new hydrometeorological data arrives, keeping predictions accurate as basin conditions change;

Main Gateway capabilities exercised

Multi-source data integration, streaming and batch processing, HPC/GPU execution, digital-twin workflows, geospatial visualisation, alerts, APIs, monitoring and model retraining.

DUS-ENV-02 - Hydrological Forecasting Research

Primary actors: Researcher, hydrologist and domain expert, data curator, public-sector analyst, water-management practitioner, developer of early warning systems.

Current situation and need

Data-driven hydrological forecasting research faces two gaps. Observations relevant for a single basin are dispersed across the national hydrometeorological service, water-management enterprises, dedicated IoT sensor networks and public forecast providers, in different formats, spatial and temporal resolutions and update cycles. No automatically harvested, consolidated, versioned dataset exists, so researchers spend most of their effort on acquisition and cleaning rather than modelling. Where measured series are too incomplete to build and evaluate models directly, synthetic series from physically based models have to be used.

Producing a good predictive model per station requires the sustained effort of an expert team, or an extensive automated search that must be repeated whenever a station fails, is restored, or new data arrives. Research groups and public institutions lack the necessary computing capacity and, more importantly, AutoML and MLOps tooling that is integrated with it, so model building today remains manual and small-scale.

Target scenario

The researcher signs in to the SAIFA AI Gateway, creates a project, and searches the catalogue for hydrological and meteorological datasets by basin, station, variable, time span and completeness, reviewing each dataset's quality report. The researcher can upload additional data (sensor series, physically-modelled synthetic series, observations for another basin), which is validated, described with metadata, versioned and made discoverable under conditions the contributor sets. Over such a dataset, the researcher can develop, train and evaluate predictive models using the Gateway's standard machine-

learning environment, choosing the model architecture, features and training regime, and can invoke the AutoML service to search for candidate models automatically when a fully automated approach is preferred, or fine-tune an already registered model on a new basin or period. Trained models can be run over held-out or live data to generate forecasts, so the researcher can inspect predictions against the observed series before adopting a model. Candidates, whether hand-built or automatically generated, are compared over an agreed verification period, and the researcher can study design choices such as the number of upstream input stations, feature selection or robustness to a station failure.

The researcher can export a validated model in a portable, FAIR-documented format - with a model card, data description, inputs, evaluation results, licence and persistent identifier - and can be placed under an MLOps pipeline for performance monitoring, versioned deployment and rollback. The dataset, workflow, model version and quality report are preserved in the SAIFA repository and published with FAIR metadata where appropriate.

Expected value and outputs

• less researcher time spent on data acquisition and cleaning, more on modelling, thanks to a consolidated, versioned per-basin data pool of curated public and institutional datasets available through the catalogue;

• full control over model design, training and evaluation using standard machine-learning tooling and shared computing resources, with AutoML available as an optional accelerator rather than the only path to a model;

• the ability to run trained models over held-out or live data and directly compare predictions against observed series before adopting a model;

• portable, FAIR-documented, citable models and datasets that other researchers can reuse, audit and extend;

Main Gateway capabilities exercised

Catalogue search and dataset filtering, dataset curation and validation, dataset upload, model development and training, model inference, workflow execution, AutoML and model selection, fine-tuning, HPC and GPU orchestration, model export, MLOps pipelines, model versioning, data versioning, provenance and publication.

3.3. Health

DUS-BIO-01 - Generating Synthetic Whole-Genome Sequencing Data

Primary actors: Bioinformatician, genomic researcher, pipeline developer.

Current situation and need

Researchers need controlled synthetic sequencing datasets to develop, test and benchmark mapping, variant-calling and quality-control pipelines. Running the appropriate simulators with different platforms, error profiles and coverage parameters requires domain software, containers and substantial compute resources.

Target scenario

A researcher signs in, selects the WGS simulation workflow and uploads or selects a reference genome in FASTA format. The researcher specifies the sequencing platform, read type, coverage, error profile and other simulation parameters.

The Gateway selects an approved container containing tools such as InSilicoSeq, ART, SimLoRD or PBSIM3, verifies the required resources and submits the workload to an HPC environment. The user follows job status and logs without directly interacting with the cluster scheduler.

On completion, the system stores paired or long-read FASTQ files, execution logs, the simulation configuration, checksums and resource-consumption information. The resulting synthetic dataset can be used in another workflow or registered as a versioned research asset.

Expected value and outputs

- controlled synthetic FASTQ datasets;
- repeatable benchmark inputs;
- reduced setup time for sequencing simulators;
- traceable configuration and execution environment;
- support for development and validation of bioinformatics pipelines.

Main Gateway capabilities exercised

Dataset upload, container catalogue, job templates, HPC scheduling, large-file handling, monitoring, provenance and result registration.

DUS-BIO-02 - Mapping Spatial Transcriptomics and Cellular Interactions

Primary actors: Cancer researcher, oncologist, bioinformatician, computational biologist.

Current situation and need

Spatial transcriptomics combines large molecular matrices, spatial coordinates, single-cell reference data and tissue images. Analysis requires numerous tools, substantial memory and compute resources, and careful preservation of intermediate steps.

Target scenario

The researcher creates a protected health-research project and uploads or selects spatial transcriptomics, single-cell RNA sequencing and associated imaging data. Access is granted only to authorised project members.

The user selects an approved workflow for preprocessing, integration, cell-type mapping, spatial-neighbourhood detection and ligand-receptor analysis. Through the Gateway, the user may request validation of data classification and permissions, selection of an appropriate execution environment and execution of the workflow by integrated services or authorised user-provided code using CPU or GPU resources.

The researcher monitors the analysis, examines quality indicators and explores resulting spatial maps, cell niches, interaction networks and candidate biomarkers. The system records the complete lineage between input data, reference annotations, parameters, model versions and outputs.

Expected value and outputs

- spatial maps of cell populations;
- identified cellular niches;
- ligand-receptor interaction networks;
- candidate biomarkers and regulatory pathways;
- reproducible and access-controlled analysis records.

Main Gateway capabilities exercised

Sensitive-data governance, large dataset transfer, HPC/GPU execution, notebooks, workflow orchestration, interactive visualisation, lineage and protected collaboration.

DUS-BIO-03 - Integrating Multi-Omics Data and Developing Polygenic Risk Scores

Primary actors: Geneticist, clinical bioinformatician, biomedical researcher, data scientist.

Current situation and need

The development of risk-prediction models requires integration of genomic, transcriptomic, proteomic, metabolomic, clinical and phenotypic data. These data are heterogeneous, potentially sensitive and computationally demanding.

Target scenario

An authorised researcher creates a controlled project and registers the legal and ethical basis for processing the selected data. The researcher uploads or references the permitted datasets and chooses a multi-omics integration and risk-modelling workflow.

The Gateway provides access to bioinformatics workflows, including quality control, variant processing, annotation, multi-omics integration, feature selection, and model training. The user may compare several methods for polygenic-risk scoring and machine learning, execute cross-validation and evaluate performance, bias, uncertainty and subgroup behaviour.

Validated models are registered with their datasets, parameters, metrics, intended purpose and limitations. Only authorised users may access sensitive inputs, while approved aggregated models or research outputs can be shared more widely.

Expected value and outputs

- validated polygenic risk scores;
- prioritized genetic variants and biomarkers;
- annotated molecular networks;
- documented predictive models;
- evidence for risk stratification and preventive research.

Main Gateway capabilities exercised

Secure data projects, HPC/GPU training, experiment tracking, model registry, benchmarking, explainability, access control and audit.

DUS-HE-01 - Developing and Validating Models for Early Disease Detection

Primary actors: Clinical researcher, physician-scientist, medical data scientist.

Current situation and need

Clinical and biomedical teams need models capable of supporting earlier detection of conditions such as Parkinson's disease, diabetes and retinal disorders. Model development requires controlled data access, repeatable experiments and clinically meaningful evaluation.

Target scenario

A clinical research team creates a restricted project and selects an approved clinical or sensor dataset. The team chooses a baseline model or training template, defines the prediction task and runs several experiments using the Gateway.

The system tracks dataset versions, preprocessing steps, hyperparameters, metrics and model artefacts. Researchers compare candidate models, assess sensitivity, specificity, calibration, uncertainty and subgroup performance, and inspect explainability outputs.

A model that meets the research protocol's acceptance criteria is submitted for technical and domain review. The registered model remains clearly labelled as a research or decision-support asset and is not presented as an autonomous clinical diagnosis.

Expected value and outputs

- reproducible candidate diagnostic-support models;
- transparent performance and limitation reports;
- earlier identification of disease-related patterns;
- support for clinical research and decision-making;
- governed transition from experiment to validated prototype.

Main Gateway capabilities exercised

Protected datasets, model training, experiment tracking, benchmarking, explainability, review workflow, model registry and audit.

DUS-HE-02 - Analysing Therapy Response and Cognitive Engagement

Primary actors: Clinical researcher, therapist, neuroscientist.

Current situation and need

Research teams need repeatable methods for analysing EEG and related signals to understand therapeutic effects and cognitive responses. Existing analyses may be fragmented across local scripts and manually recorded experiments.

Target scenario

The researcher selects an approved EEG analysis workflow, provides a permitted dataset and records the relevant experimental protocol. Using an approved service, workflow or authorised user-provided code, the researcher may preprocess signals, train or apply an approved model and generate indicators of cognitive components and therapy-related changes.

Where supported by the selected workflow, the researcher may compare sessions or patient groups, review available model-confidence information and visual explanations, and link each result to the input dataset, protocol and model version. Results may be exported as research reports or retained for longitudinal studies.

Expected value and outputs

- quantitative therapy-response indicators;
- reproducible EEG analysis;
- comparison of sessions and intervention groups;
- improved evidence for therapy monitoring;
- documented model uncertainty and limitations.

Main Gateway capabilities exercised

Signal-data management, workflow execution, model training and inference.

DUS-HE-03 - Deriving Personalised Patient-Support Recommendations

Primary actors: Clinical researcher, therapist, behavioural specialist.

Current situation and need

Behavioural and functional assessment can involve lengthy manual observation. Research teams need tools for identifying patterns that may support more individualized interventions, particularly for children with autism and related conditions.

Target scenario

An authorised professional uploads or selects a consented behavioural dataset and chooses an approved assessment model. The Gateway processes the data and returns detected behavioural patterns, functional-performance indicators and model confidence.

The professional reviews the result rather than accepting it automatically, may add contextual observations and may compare results over time. Where the selected service or workflow produces recommendations, they are treated as decision-support outputs and remain subject to professional judgement and applicable clinical governance.

Expected value and outputs

- structured behavioural indicators;
- longitudinal comparison of functional performance;
- research support for personalised interventions;
- documented human review and oversight;
- controlled handling of sensitive data.

Main Gateway capabilities exercised

Sensitive-data access, inference, human validation, reporting and audit.

DUS-HE-04 - Monitoring the Robustness and Security of Digital-Health Systems

Primary actors: Healthcare IT operator, cybersecurity analyst, biomedical engineer.

Current situation and need

Connected healthcare systems and IoT medical devices may suffer from faults, anomalies and cyber threats that can compromise service reliability and patient safety.

Target scenario

An authorised operator registers a monitored digital-health environment and configures the permitted telemetry sources, providing the corresponding data as a dataset. Through the Gateway, the operator may invoke approved anomaly- and threat-detection services, workflows or authorised user-provided code to analyse incoming operational data.

Where anomalies are detected, the selected service or workflow may present them with severity, affected resources, supporting evidence and recommended investigation steps. The operator may confirm, dismiss or escalate an event. Confirmed incidents and model feedback may be retained to support model improvement and audit.

The environment is monitored without exposing sensitive patient information beyond the minimum required for the security task.

Expected value and outputs

- earlier detection of faults and suspicious behaviour;
- improved operational resilience;
- traceable security alerts and human responses;
- feedback data for improving detection models;
- stronger trust in digital-health infrastructure.

Main Gateway capabilities exercised

Streaming or batch inference, monitoring, alerting, security policies, human review, audit and controlled telemetry access.

3.4. Language and Culture

DUS-LC-01 - Detecting Hate Speech in Serbian-Language Content

Primary actors: Media analyst, social-platform moderator, public institution, researcher in language technologies.

Current situation and need

Institutions and online platforms need to identify harmful and hostile communication in Serbian-language content. Available multilingual moderation tools often have limited coverage of Serbian, domain-specific expressions, cultural context and language variants.

Target scenario

A user accesses the SAIFA AI Gateway and selects a Serbian-language hate-speech detection service from the catalogue. The user uploads or connects a collection of textual content, selects an available pre-trained model and configures the analysis parameters. The Gateway validates access to the data, executes inference on the appropriate computing environment and presents classified content, confidence values, aggregate statistics and explanations where available.

A qualified user may compare several model versions, evaluate them on an approved benchmark dataset and request fine-tuning on a domain-specific, legally usable dataset. Validated models and evaluation reports can subsequently be registered in the catalogue and shared with authorised users or published according to the defined governance process.

Expected value and outputs

- safer Serbian-language online environments;
- improved moderation and monitoring capabilities;
- reusable Serbian-language datasets and models;
- documented model performance, limitations and known biases;
- support for researchers, public institutions and media-monitoring organisations.

Main Gateway capabilities exercised

Catalogue search, dataset access, model inference, model evaluation, fine-tuning, model cards, access control, explainability, audit and publication.

DUS-LC-02 - Constructing and Exploring Domain Knowledge Graphs

Primary actors: Researcher, domain expert, data curator, public-sector analyst.

Current situation and need

The creation of domain-specific knowledge graphs requires substantial manual effort for document preparation, ontology development, entity recognition, relation extraction, normalisation and validation. Existing generic tools may not perform adequately on specialist domains or Serbian and other low-resource languages.

Target scenario

A researcher signs in to the Gateway and creates a knowledge-graph project. The researcher selects an existing document collection or uploads a new one, defines the relevant domain and selects an approved extraction workflow. The user may choose a transformer or LLM-based extraction model, configure entity and relation processing, and submit the workflow to the AI Factory infrastructure.

Requests are submitted through the AI Gateway, which forwards them to the appropriate AI services responsible for preprocessing, entity and relation extraction, normalization, graph construction, and community analysis.

The graph, its source dataset, workflow definition, model version and quality report can be preserved in the SAIFA repository and, where appropriate, published with FAIR metadata and a persistent identifier.

Expected value and outputs

- reusable domain knowledge graphs;
- reduced manual graph-construction effort;
- traceable links between source documents and graph entities;
- reproducible extraction workflows;
- interactive exploration and semantic discovery;
- publishable FAIR research assets.

Main Gateway capabilities exercised

Dataset upload, workflow execution, model selection, HPC/GPU orchestration, graph storage, provenance, versioning and publication.

DUS-LC-03 - Simulating Collective Social Behaviour with a Digital Twin

Primary actors: Computational social scientist, policy analyst, social-platform researcher, market analyst.

Current situation and need

Researchers and decision-makers lack accessible and scalable environments for studying polarization, misinformation, marginalization, narrative development and other emergent social phenomena. Existing systems frequently require manual infrastructure configuration and isolated deployments.

Target scenario

A computational social scientist logs in to the AI Gateway, creates a project and selects a predefined digital-twin template representing an online environment such as Reddit, X or Instagram. The researcher configures the simulated population, agent profiles, selected LLMs, moderation rules, interaction mechanisms and experiment duration.

The Gateway validates the project quota, deploys the required server and client components in a containerized environment, allocates GPU resources for LLM inference and starts the simulation. During execution, the researcher monitors simulation rounds, system health, trends, resource consumption and detailed logs.

The researcher can compare several runs, reproduce an earlier experiment, share results with project members and prepare selected outputs for publication.

Expected value and outputs

- reproducible simulations of collective online behaviour;
- evidence for research and policy analysis;

• reusable digital-twin templates;

• synthetic datasets, networks and experiment configurations;

• reduced dependence on manually managed computing infrastructure.

Main Gateway capabilities exercised

Project management, workflow templates, container deployment, LLM inference, GPU scheduling, runtime monitoring, result storage, comparison, collaboration and notifications.

DUS-LC-04 - Publishing and Reusing Digital-Society Research Assets

Primary actors: Researcher, data steward, project owner, external researcher.

Current situation and need

Simulation outputs, prompts, models, code and analysis notebooks are frequently stored separately, with incomplete metadata and weak provenance. This makes verification, citation and reuse difficult.

Target scenario

After completing a simulation or language-analysis workflow, the researcher reviews the generated datasets and associated artefacts. The Gateway automatically captures the workflow configuration, software environment, model versions, input sources and execution history.

The researcher prepares a dataset or model card, assigns access and licensing conditions, validates that sensitive information has been removed or appropriately protected, and submits the resource for review. After approval, the resource becomes searchable in the SAIFA catalogue. Publicly releasable assets may be exported to Zenodo or another compatible repository and assigned a persistent identifier.

Another researcher discovers the resource through full-text or semantic search, inspects its provenance and licence, downloads it or launches a new experiment using the resource without manually reconstructing its original context.

Expected value and outputs

• FAIR datasets and reproducible research packages;

• versioned models, prompts, workflows, notebooks and code;

• persistent identifiers and citation information;

• improved reuse of project results;

• traceability from published result to the originating experiment.

Main Gateway capabilities exercised

Resource review, licensing, dataset and model versioning, provenance, catalogue search, persistent identifiers, repository synchronisation and controlled download.

DUS-LC-05 - Prediction of homologous proteins domains

Primary actors: Molecular biologist, bioinformatician, biotechnology researcher.

Current situation and need

Searching large genomic and proteomic databases for proteins homologous to known human factors requires the coordinated use of several tools and databases, followed by filtering, sequence alignment and expert interpretation.

Target scenario

The researcher creates a project, uploads reference protein sequences and chooses relevant target databases. The user selects a validated discovery workflow containing sequence search, filtering, multiple-sequence alignment and optional AI-based candidate ranking.

The Gateway executes the workflow on an HPC resource, tracks the software and database versions and records the relationship between the input sequences, query parameters and generated candidates. Optional protein-language or embedding models can rank proteins by predicted structural or functional similarity.

The user inspects candidate lists and alignments, compares alternative search parameters, records expert validation and exports selected sequences and reports for further experimental investigation.

Expected value and outputs

- ranked candidate proteins;
- FASTA collections of selected sequences;
- multiple-sequence alignments;
- reproducible search and filtering reports;
- improved support for drug discovery, synthetic biology and cross-species research.

Main Gateway capabilities exercised

Reference-data access, HPC workflow execution, container management, model inference, provenance and collaboration.

3.5. Cross-Domain Gateway Adoption

These stories are not separate scientific verticals. They illustrate cross-domain patterns through which users may enter the SAIFA ecosystem, acquire skills, receive support, develop solutions and transfer approved assets into the European AI Factory network.

DUS-XD-01 - Developing an AI Solution from Data to a Validated Prototype

Primary actors: Research team, startup, SME, public institution, advanced AI developer.

Current situation and need

Organisations often have an idea and some data but lack integrated infrastructure, MLOps processes, expert support and a clear path from experimentation to a validated prototype.

Target scenario

An organisation contacts SAIFA through the Gateway and describes the intended problem, available data, expected users and operational constraints. With support from domain and technical experts, the organisation creates a project and identifies an appropriate starting workflow, dataset, model or notebook.

The team iteratively prepares data, runs experiments, evaluates models and deploys a controlled prototype. The Gateway may provide access to integrated services that preserve the relationships between data, code, experiments, models and deployment results. The team receives guidance on security, ethics, licensing and technical readiness.

At the end of the process, the organisation has a reproducible prototype and a documented plan for further validation, scaling or production deployment.

Expected value and outputs

- reduced barrier to AI adoption;
- reproducible prototype;
- reusable workflow and model assets;
- technical and governance readiness report;
- clearer route toward production or commercialization.

Main Gateway capabilities exercised

Onboarding, project workspace, data and model catalogue, notebooks, training, HPC/GPU execution, experiment tracking, consulting, support for deployment testing and documentation.

DUS-XD-02 - Learning AI and HPC through a Guided Practical Path

Primary actors: Student, early-career researcher, SME employee, public-sector professional, novice AI practitioner.

Current situation and need

Potential users have different levels of AI and HPC experience. Generic courses often do not connect learning directly to controlled access to real datasets, models and computing resources.

Target scenario

A user signs in and browses the catalogue of individual courses and learning paths, ranging from AI literacy to advanced model development and HPC deployment. Courses and paths can be filtered by role and, where relevant, domain of use.

The user follows modular material, opens guided notebooks and performs lab exercises on approved datasets, curated or anonymised for training use where the source domain requires it, using controlled resource templates. Progress is recorded, and the user receives feedback, support and, where the training programme defines a certification pathway, a certificate of completion.

A learner who completes the introductory path can move directly into a research or innovation project without changing identity, portal or working environment.

Expected value and outputs

- improved AI and HPC skills;
- practical experience on governed infrastructure;
- reusable educational notebooks and exercises;
- progression from beginner to advanced practitioner.

Main Gateway capabilities exercised

Learning management, role-adapted pathways, notebooks, controlled resource templates, training datasets, progress tracking, resource quotas, support, and federated identity.

DUS-XD-03 - Receiving Coordinated Technical and Domain Support

Primary actors: Any SAIFA user, organisation representative, developer or project owner.

Current situation and need

Users may not know whether a problem concerns access policy, infrastructure, data management, a domain model or application software. Fragmented support channels lead to delays and repeated explanations.

Target scenario

The user opens the helpdesk from the relevant resource, job or experiment page. The Gateway automatically includes permitted technical context such as the affected service, job identifier, error category and recent status information.

The request is routed to the appropriate support level or domain expert. The user follows its status, exchanges additional information and receives a documented resolution. Resolved cases contribute to a searchable knowledge base, while aggregated ticket trends inform platform improvements and training priorities.

Expected value and outputs

- one support entry point;
- faster and more consistent issue resolution;
- reduced user onboarding effort;
- reusable knowledge-base material;
- evidence about recurring platform and user needs.

Main Gateway capabilities exercised

Helpdesk, contextual ticket creation, expert routing, notifications, knowledge base, service monitoring and usage analytics.

DUS-XD-04 - Sharing a Project Resource Securely across Organisations and Sectors

Primary actors: Research consortium, startup-university team, public-private project, data or model owner.

Current situation and need

Cross-institutional teams need to share datasets, models, workflows and results, but must preserve ownership, access restrictions, licence conditions and accountability.

Target scenario

A project owner creates a shared workspace, invites members from participating organisations and assigns roles. Resource owners upload or register datasets, models, notebooks and workflows and define whether each resource is private, project-restricted, platform-visible or public.

The Gateway applies the relevant access rules to discovery, download, execution and publication. Every state-changing operation is audited. The team can review resources, reproduce experiments and transfer approved outputs to another project or publication process without sending uncontrolled file copies.

Expected value and outputs

- secure cross-organizational collaboration;
- consistent access rights and resource ownership;
- shared and reproducible project assets;
- reduced duplication of data and models;
- clear auditability and accountability.

Main Gateway capabilities exercised

Organisations, projects, membership, RBAC/ABAC, resource visibility, versioning, review, audit and secure sharing.

DUS-XD-05 - Federating Resources with Pharos, IT4LIA and Other AI Factory Services

Primary actors: SAIFA researcher, resource curator, EuroHPC user, partner AI Factory operator.

Current situation and need

Models, datasets and workflows developed nationally need a controlled path toward European discovery, validation and production-scale infrastructure. Users should not need to understand each partner platform's internal technical details.

Target scenario

A resource owner selects an approved SAIFA model, dataset or workflow and requests federation. The Gateway verifies metadata, provenance, licence, quality status and access policy and maps the resource to the schema required by the target AI Factory.

The resource is published or synchronized with a partner catalogue while retaining its SAIFA identity and provenance. A user can also discover a Pharos, IT4LIA, PARADOX or ITE resource through the SAIFA catalogue, authenticate through the supported European identity path and use or request access to the resource without creating an unrelated local workflow.

For compute-intensive final validation, an approved experiment may be transferred from a national development environment to a partner AI Factory, with results returned and linked to the original SAIFA project.

Expected value and outputs

- cross-factory discovery and reuse;
- production-oriented validation on PARADOX, ITE or EuroHPC resources;
- shared models, datasets, synthetic data and evaluation results;
- reduced fragmentation between national and European services;
- traceable provenance across federated platforms.

Main Gateway capabilities exercised

Federated identity, catalogue adapters, metadata mapping, publication review, provenance, data and model exchange, remote execution and synchronisation monitoring.

4. System Use Cases

System use cases describe how human actors and external systems may interact with the SAIFA AI Gateway to achieve defined objectives. The descriptions remain independent of final implementation products wherever possible: terms such as Identity and Access Management Service, Resource Catalogue, Compute Backend, Object Storage, Workflow Service and Federation Adapter denote logical components. This section presents a representative set of core use cases from the wider working catalogue. Exact roles, quota rules, workflow states, interface details, automated review procedures and responsibility allocation remain subject to validation. Inclusion here does not commit SAIFA to implement a use case, assign all of its steps to the Gateway, or include it in the beta or subsequent releases.

4.1. System boundary

The SAIFA AI Gateway system boundary includes the user-facing portal and APIs, together with the logical coordination services required to make the following capabilities accessible directly or through integrated components:

- authentication and authorisation;
- organisation, project and collaboration management;
- catalogue and resource discovery;
- dataset and model management;
- workflow and compute-job execution;
- inference;
- experiment tracking and model evaluation;
- training and practical lab environments;
- monitoring, usage accounting and audit;
- helpdesk and user support;
- federation with European AI Factory services.

The following elements are treated as systems external to the Gateway boundary:

- institutional identity providers participating in AMRES/eduGAIN;
- EuroHPC identity services;
- national HPC and AI infrastructure;
- Pharos, IT4LIA and other partner AI Factory services;
- external repositories such as Zenodo;
- domain-specific data sources and scientific databases;
- email and notification-delivery services.

The Gateway mediates access to these systems but does not replace their internal functionality.

4.2. Actors

Table 4.1 Human actors

Actor	Description
Anonymous visitor	A person who has not established an authenticated Gateway session.
Authenticated user	A registered user acting within an assigned role, organisation and resource allocation.
Researcher or AI practitioner	A user executing data, model, workflow, inference or HPC activities.
Student	A learner enrolled in a course or practical lab.
Mentor or instructor	A user who creates and manages courses and practical exercises.
Organisation representative	A user managing membership and allocated resources within an organisation.
Data administrator	A reviewer responsible for dataset publication and data-governance checks.
Model reviewer	A reviewer responsible for model evaluation and publication decisions.
Platform administrator	An operator managing users, resources, platform configuration, monitoring and federation.
Support-team member	A person resolving submitted user-support requests.

Table 4.2 External system actors

External actor	Responsibility
Institutional Identity Provider	Authenticates users through institutional credentials.
EuroHPC Identity Service	Authenticates users requiring federated European access.
Compute Backend	Executes CPU-, GPU- or HPC-based jobs.
Model Serving Environment	Executes model-inference requests.
External Resource Repository	Stores or publishes externally shared research resources.
Partner AI Factory Platform	Provides or receives federated models, datasets and services.

Notification Service	Delivers email or other configured notifications.
Helpdesk System	Records and manages user-support tickets.

4.3. Representative system use cases

IAM-UC-002 - Authenticate through an Institutional Identity Provider

Primary actor: Anonymous visitor

Supporting actors: Institutional Identity Provider, Gateway IAM service

Goal

The user establishes an authenticated Gateway session using an active institutional identity.

Preconditions

- The user is not currently authenticated.
- The user has an active account at an institution participating in the supported federation.
- The institutional identity provider is available.

Trigger

The user selects institutional sign-in from the Gateway login page.

Main flow

1. The Gateway presents the institutional-sign-in option.
2. The user selects the home institution.
3. The Gateway redirects the user to the institutional identity provider.
4. The user authenticates according to the institution's procedure.
5. The identity provider returns the required identity and affiliation attributes.
6. The Gateway validates the response and associates it with an existing account or creates the corresponding Gateway account.
7. The Gateway establishes the session.
8. The user is redirected to the role-appropriate Gateway view.

Alternative flows

- The institution is not available through the federation.
- Authentication is rejected by the institution.
- Mandatory identity attributes are missing.
- The returned email address conflicts with an existing account established through another method.
- The user cancels the external authentication process.

Postcondition

The user is authenticated and the institutional affiliation is available for subsequent authorisation and organisational-resource decisions.

CAT-UC-002 - Search and Filter the Resource Catalogue

Primary actor: Authenticated user

Supporting actors: Local catalogue, federated catalogue sources

Goal

The user discovers a model, dataset, workflow, course or other resource that the user is authorised to view.

Preconditions

- The user is authenticated.
- The user is authorised to access the catalogue.
- At least the local catalogue is available.

Trigger

The user opens the resource catalogue.

Main flow

1. The Gateway obtains the user's role, affiliation and relevant access attributes.
2. The Gateway presents only resources visible to that user.
3. Federated resources are marked with their source.
4. The user enters a search term.
5. The user optionally filters by resource type, domain, language or origin.
6. The system returns the matching authorised resources.
7. The user selects a resource.
8. The system displays its metadata, origin, version, licence and status.

Alternative flows

- No resource matches the selected criteria.
- A federated catalogue is temporarily unavailable; local results remain available.
- The user clears one or more filters.
- The selected resource has become unavailable or the user's access has changed.

Postcondition

The user sees either the details of an authorised resource or a filtered list. No inaccessible resource is exposed.

ORG-UC-007 - Create a Project Workspace

Primary actor: Authenticated user

Goal

The user creates a logical workspace for grouping collaborators, resources, experiments and outputs.

Preconditions

The user is authenticated.

Trigger

The user selects the action for creating a new project.

Main flow

1. The Gateway presents the project-creation form.
2. The user enters the project name and description.
3. The user selects the intended visibility.
4. The user confirms project creation.
5. The system validates the required fields.
6. The system creates the project and assigns the initiating user as its owner.
7. The system opens the project workspace.

Alternative flows

- A conflicting project name exists in the applicable scope.
- A required field is missing.
- The user cancels before confirmation.

Postcondition

A project workspace exists with the initiating user as owner and is visible according to the selected policy.

CAT-UC-oo8 - Upload and Register a Dataset

Primary actor: Authenticated user

Supporting actors: Dataset storage, resource catalogue

Goal

The user stores a dataset and creates a catalogue record containing the required metadata.

Preconditions

- The user is authenticated.
- The user has sufficient storage allocation.
- The selected file format is accepted for storage.

Trigger

The user selects the action for adding a dataset.

Main flow

1. The Gateway presents a dataset-registration form.
2. The user enters the name, description, licence, language and domain.
3. The user selects the intended visibility.
4. The user selects or drags the dataset file into the upload interface.
5. The system displays upload progress.
6. The system stores the successfully uploaded file.
7. The system creates the associated catalogue record.
8. Where supported, the system displays a safe sample or structural preview.

Alternative flows

- The user has insufficient storage allocation.
- A dataset with a conflicting name exists in the applicable scope.
- A mandatory metadata field is missing.
- The transfer is interrupted and must be resumed.
- The user cancels before completion.

Postcondition

The dataset and its metadata are stored. It remains subject to the applicable review and publication rules.

CAT-UC-003 - Submit a Resource for Review

Primary actor: Authenticated resource owner

Supporting actors: Data administrator or model reviewer

Goal

The user submits a draft dataset or model for formal review before publication.

Preconditions

- The user is authenticated and authorised to manage the resource.
- The resource exists in a draft state.
- The required artefact has already been stored or registered.

Trigger

The user selects the action to publish or submit the resource for review.

Main flow

1. The system presents the current resource metadata.
2. The user completes the description, domain tags, licence and requested visibility.
3. The user submits the resource.
4. The system validates mandatory metadata.
5. The system changes the resource status from draft to submitted or under review.
6. The system assigns the resource to the appropriate review queue.
7. The system notifies the responsible reviewer.
8. The user receives confirmation.

Alternative flows

- Required metadata or licence information is missing.
- The resource is already under review or published.
- The user saves the work without submission.
- A reviewer subsequently returns the resource for revision.

Postcondition

The resource is assigned to an appropriate reviewer but is not yet publicly available.

CAT-UC-004 / MOD-UC-011 - Review a Dataset or Model Version

Primary actors: Data administrator or model reviewer

Goal

An authorised reviewer determines whether a submitted dataset or model version may be published.

Preconditions

- The reviewer is authenticated and holds the appropriate review role.
- The submitted resource is in the review queue.
- Required metadata and supporting evidence are available.

Main flow

1. The reviewer opens the review queue.
2. The reviewer selects a submitted resource.
3. The system displays metadata, provenance, licence information and the applicable artefact or evaluation results.
4. The reviewer examines the evidence.
5. The reviewer records a comment.
6. The reviewer selects approve, reject or return for revision.
7. The system records the reviewer, decision, time and comment.
8. The system updates the resource status.
9. The system notifies the submitting user.

Alternative flows

- Licence or resource origin is unclear.
- Required evidence is incomplete.
- The resource is withdrawn during review.
- The reviewer leaves without confirming a decision.

Postcondition

The resource is approved, rejected or returned for revision, with the decision recorded for audit.

AIF-UC-001 - Submit a Compute-Intensive Job

Primary actor: Authenticated researcher or AI practitioner

Supporting actors: Authorisation service, resource-allocation service, compute backend

Goal

The user submits a training, fine-tuning, batch-processing or scientific-computing job.

Preconditions

- The user is authenticated and authorised to submit jobs.
- The user has access to all selected input resources.
- Sufficient resource allocation is available.

- At least one eligible compute environment is available.

Trigger

The user submits a completed job configuration through the portal, API or notebook environment.

Main flow

1. The Gateway presents or accepts the job type, inputs, parameters and output destination.

2. The user confirms the job request.

3. The system validates required fields.

4. The system validates access to each input.

5. The system estimates and reserves the required resource allocation.

6. The system determines an eligible compute environment.

7. Data-location restrictions are applied where compute-to-data execution is required.

8. The system creates the job record and places it in the queue.

9. The system returns the job identifier.

Alternative flows

- Required parameters are missing.
- The user cannot access an input resource.
- The available quota is insufficient.
- A competing request consumes the remaining quota before reservation completes.
- The required compute-to-data environment is unavailable.
- The user cancels before submission.

Postcondition

A queued job exists with linked inputs, configuration, output destination and reserved resources. Submission is recorded in the audit trail.

AIF-UC-002 - Monitor Job Status and Logs

Primary actor: Authenticated job owner

Supporting actor: Compute backend

Goal

The user follows the state and progress of a submitted job.

Preconditions

The user has an existing job that the user is authorised to view.

Trigger

The user opens the list of submitted jobs or selects a job from a related experiment.

Main flow

1. The system displays the user's jobs with type, submission time and current state.

2. The user selects a job.

3. The system displays its current state.

4. If available, the system displays queue position, estimated or actual resource use and execution logs.

5. The system refreshes status and logs while execution continues.

6. When execution ends, the system displays the final state.

7. For a successfully completed job, the system provides access to the result artefacts.

Alternative flows

- The job is still waiting and execution logs do not yet exist.
- The job fails and error information is displayed.
- Log delivery is delayed, while scheduler status remains authoritative.
- An administrator terminates the job.

Postcondition

The user has current information about the job. Viewing the status does not change its state.

AIF-UC-003 - Retrieve Job Results

Primary actor: Authenticated job owner

Supporting actor: Result storage

Goal

The user retrieves or registers the outputs of a successfully completed job.

Preconditions

- The job completed successfully.
- Result artefacts exist.
- The user is authorised to access them.

Main flow

1. The user opens the completed job.

2. The system displays the output artefacts, logs and metrics.

3. The user selects one artefact or the complete result set.

4. The system provides controlled access to the selected results.

5. The system records the access in the audit trail.

6. Where the result represents a reusable model or dataset, the user may initiate its registration as a new resource.

Alternative flows

- The job did not complete successfully.
- The result has expired according to the retention policy.
- The transfer is interrupted and resumed.
- The user no longer has access to the project or source resource.

Postcondition

The result is downloaded or registered as a draft resource. The original result remains stored according to the applicable retention policy.

INF-UC-001 - Invoke a Published Model

Primary actor: Authenticated user

Supporting actors: API access service, model-serving environment

Goal

The user submits input to an authorised model and receives an inference result.

Preconditions

- The user is authenticated through a session or scoped programmatic credential.
- The model is available to the user.
- The model is currently served.
- The user has remaining inference allocation.

Main flow

1. The user selects the model and obtains its documented invocation information.
2. The user sends an inference request through the supported interface.
3. The Gateway authenticates the request.
4. The Gateway checks role- and attribute-based access.
5. The Gateway checks applicable usage limits.
6. The Gateway forwards the authorised request to the internal serving environment.
7. The serving environment returns the model output.
8. The Gateway returns the response to the user and records resource consumption.

Alternative flows

- The credential or session is invalid.
- The user is not authorised for the model.
- The inference allocation is exhausted.
- The model-serving environment is unavailable.
- The request input is incompatible with the model.

Postcondition

The user receives an inference response, or a controlled error is returned without unauthorised execution.

WF-UC-001 - Define and Execute a Machine-Learning Pipeline

Primary actor: Authenticated researcher or AI practitioner

Supporting actors: Workflow service, compute backend

Goal

The user combines several processing, training, evaluation or registration activities into a reproducible workflow.

Preconditions

- The user is authorised to execute jobs.
- The required input datasets and models are available.

- The user has sufficient resource allocation.

Main flow

1. The user creates a pipeline definition.
2. The system presents the available processing steps.
3. The user arranges and connects the steps.
4. The user assigns inputs, parameters and eligible execution environments.
5. The user submits the pipeline.
6. The system validates the workflow structure, resource access and estimated resource requirements.
7. The system executes the steps in the defined dependency order.
8. Each output is passed to the dependent step.
9. The system presents the final result and records lineage across the full workflow.

Alternative flows

- The workflow graph is invalid or has missing inputs.
- Resource allocation is insufficient.
- One step fails.
- The user requests advanced branching or retry behaviour not supported in the current phase.

Postcondition

The completed workflow and its outputs are linked to the input resources, configurations and executed jobs.

MOD-UC-001 - Train a Model Using a Catalogue Dataset

Primary actor: Authenticated model developer

Supporting actors: Dataset catalogue, compute backend, experiment-tracking service

Goal

The user trains a model using an authorised dataset and captures the resulting artefacts and provenance.

Preconditions

- The user can access the selected dataset.
- Sufficient compute allocation is available.
- The selected training configuration is supported.

Main flow

1. The user selects a dataset.
2. The user initiates model training.
3. The system presents the training configuration.
4. The user selects the model architecture or base model and sets parameters.
5. The user submits the training job.
6. The job follows the standard compute-job lifecycle.

7. After successful completion, the system stores the training artefacts.

8. The system links the artefacts to the dataset, configuration and job.

9. The user is notified of completion.

Alternative flows

- The user cannot access the dataset.
- The training configuration is invalid.
- Resource allocation is insufficient.
- The job fails or is terminated.
- A partial checkpoint exists but cannot yet be treated as a registered model.

Postcondition

Training artefacts and their automatically captured lineage exist, but the model remains unregistered until the user initiates model registration.

MOD-UC-004 - Compare Experiment Runs

Primary actor: Authenticated model developer

Supporting actor: Experiment-tracking service

Goal

The user compares model-development runs and determines which result should be considered for further evaluation.

Preconditions

At least two runs are available to the user.

Main flow

1. The user opens the experiment view.

2. The system displays runs with parameters and metrics.

3. The user selects two or more runs.

4. The system displays a comparative view.

5. The user chooses the metrics relevant to the decision.

6. The user identifies the preferred run.

Alternative flows

- Fewer than two runs are selected.
- Runs do not share all metrics.
- Some tracking data are incomplete because of temporary infrastructure isolation.

Postcondition

The comparison is displayed. The system does not automatically promote any run; selection remains a human decision.

MOD-UC-008 - Evaluate a Model Version

Primary actor: Authenticated model developer

Supporting actors: Benchmark repository, compute backend

Goal

The user evaluates a specific model version using an identified benchmark and benchmark version.

Preconditions

- The model version exists.
- The user is authorised to evaluate it.
- An accessible benchmark is available.
- Sufficient compute allocation exists.

Main flow

1. The user selects the model version.
2. The user starts evaluation.
3. The system presents available benchmark sets and versions.
4. The user selects the benchmark and confirms submission.
5. The system creates an evaluation job.
6. The job follows the standard compute-job lifecycle.
7. After successful completion, the system stores the metrics.
8. The system links the results to both the model version and benchmark version.
9. The system notifies the user.

Alternative flows

- The benchmark is unavailable or inaccessible.
- The evaluation job fails or is terminated.
- The benchmark version changes after the job has started.

Postcondition

Versioned evaluation results exist and can support review, comparison and publication.

MOD-UC-009 - Submit a Model Version for Review

Primary actor: Model owner or authorised project member

Supporting actor: Model reviewer

Goal

The user submits an evaluated model version for formal review.

Preconditions

- The model version is in draft state.
- Mandatory metadata are complete.
- Evaluation results exist.

Main flow

1. The user selects the draft model version.
2. The user initiates submission for review.
3. The system verifies mandatory metadata.

4. The system verifies that evaluation results are attached.

5. The user optionally enters a submission comment.

6. The system moves the version to the review state.

7. The system places it in the reviewer queue.

8. The system records the submitter and time.

9. Reviewers are notified.

Alternative flows

- Evaluation results are missing.
- Mandatory metadata are incomplete.
- The user cancels submission.

Postcondition

The model version is awaiting an independent review decision.

EDU-UC-004 - Open a Preconfigured Practical-Lab Environment

Primary actor: Student

Supporting actors: Learning service, notebook environment

Goal

The student opens an interactive practical environment associated with an enrolled course.

Preconditions

- The student is authenticated and enrolled.
- The selected course contains a practical lab.
- The required resources are currently available.

Main flow

1. The student opens the relevant practical exercise.

2. The system displays its description.

3. The student starts the lab environment.

4. The system starts a preconfigured notebook session within the applicable resource limits.

5. The exercise dataset is mounted with the intended access mode.

6. The system loads the exercise notebook.

7. The student is informed that the environment is ready.

Alternative flows

- Resources are temporarily unavailable.
- An existing student session is resumed.
- The student remains inactive and the session is closed according to policy.

Postcondition

The student has an active practical environment and can work with the approved exercise materials and resources.

EDU-UC-005 - Execute a Controlled Compute Job from a Lab

Primary actor: Student

Supporting actors: Notebook environment, compute backend

Goal

The student executes a predefined compute exercise without directly configuring the underlying scheduler.

Preconditions

- The student has an active lab environment.
- The practical exercise contains an approved job template.
- The requested resources are within the student or course limits.

Main flow

1. The student executes the relevant notebook action.
2. The Gateway submits the predefined job configuration.
3. The system displays the queued state.
4. The system displays the running state.
5. When execution completes, the system transfers results to the student's working environment.
6. The student continues the exercise using the returned results.

Alternative flows

- The job fails and the relevant error information is shown.
- The student closes the notebook while the job continues.
- The job exceeds the permitted exercise limits.

Postcondition

The approved job is executed and its outputs are available in the practical environment.

FED-UC-001 - Access a Resource from a Partner AI Factory

Primary actor: Authenticated federated user

Supporting actors: Gateway identity broker, partner AI Factory

Goal

The user discovers and accesses a permitted partner resource through the SAIFA Gateway.

Preconditions

- The user has the required federated identity.
- The connection to the partner platform is active.
- The requested resource remains available.

Main flow

1. The user filters the catalogue by partner origin.
2. The system displays partner resources with source attribution.
3. The user opens a selected resource.

4. The system displays the metadata supplied by the partner.
5. The user requests access or use.
6. The Gateway brokers the user's identity to the partner.
7. The partner authorises the user.
8. The user receives access without creating an unrelated local partner account.

Alternative flows

- The user did not authenticate through the required identity route.
- The partner platform is unavailable.
- The resource has been removed or access has changed.

Postcondition

The user accesses the partner resource through a federated interaction while its source remains visible.

FED-UC-002 - Export an Approved Resource to a Partner Platform

Primary actor: Authenticated resource owner

Supporting actor: Partner AI Factory

Goal

The owner makes an approved public SAIFA resource available through a connected partner platform.

Preconditions

- The resource has been approved and published.
- The resource is eligible for external sharing.
- The user has the required federated identity and ownership rights.
- The partner connection is active.

Main flow

1. The owner opens the resource.
2. The owner selects export to a partner platform.
3. The system presents the eligible partner destinations.
4. The owner selects the destination.
5. The system validates the resource's publication and sharing status.
6. The Gateway transfers the required metadata and resource reference or artefact.
7. The partner confirms receipt.
8. The Gateway records the partner publication on the local resource.

Alternative flows

- The resource is not public or not approved.
- The partner is unavailable.
- The resource already exists at the partner.
- The partner rejects the metadata or licence.

Postcondition

The resource is available through the partner platform with SAIFA identified as its source.

SUP-UC-001 - Report a Problem

Primary actor: Authenticated user

Supporting actor: Helpdesk system

Goal

The user reports a problem with enough contextual information for the support team to investigate it.

Preconditions

The user is authenticated.

Main flow

1. The user selects the report-problem action from navigation, a job, a resource or an error page.

2. The system presents a support form.

3. The user enters a title, description and problem category.

4. The user may attach an image or other permitted evidence.

5. The system adds relevant technical context that the user is authorised to share.

6. The user submits the request.

7. The system creates the support ticket.

8. The system returns a ticket identifier.

9. The platform-support team is notified.

Alternative flows

- The report is initiated from a particular job and the job identifier is attached automatically.
- The user is anonymous and is directed to the public support contact.
- The ticketing backend is unavailable and the request cannot yet be registered.

Postcondition

A traceable support request exists and the user has an identifier for further communication.

4.4. Validation status and open design assumptions

The interaction patterns above form a design baseline, but several details remain assumptions rather than formally accepted requirements.

They include:

- the exact identity-brokering implementation;
- whether both local and institutional registration are required for the Beta;
- the detailed organisational and personal quota hierarchy;
- exact catalogue visibility states;

- the review lifecycle and names of lifecycle states;
- support for custom execution environments;
- whether a visual pipeline editor is required in the Beta;
- the precise boundary between the Gateway and external learning systems;
- automatic versus manual federation synchronisation;
- the conditions under which datasets or models may be exported;
- retention periods for jobs, logs and outputs;
- the degree of model-serving lifecycle management included in the beta baseline;
- whether teams inside organisations are introduced in the Beta or a later phase.

This section defines the candidate functional behaviour of the SAIFA AI Gateway through 20 representative use-case descriptions. Product names and specific technical mechanisms are implementation candidates rather than final architecture decisions. Each use case must be validated, prioritised and assigned either to the M12 beta baseline or to a later implementation stage before development is committed.

5. Candidate Tools and Software Packages

This section identifies software tools, frameworks and platforms that may support the domain and system use cases. All products are candidates for evaluation. Their inclusion does not mean that they will all be deployed or that the first-listed option has been selected. Final choices depend on validated requirements, proofs of concept, consultation with infrastructure operators, security and compliance review, interoperability and long-term operational sustainability.

Design principle: *reuse before build, integrate before replace, and develop custom software only where SAIFA-specific value is required*.

Custom development should focus on the unified portal and user experience, the canonical metadata model, common authorisation and resource-allocation rules, audit and accounting, the HPC bridge, federation adapters, and the connection between domain workflows and horizontal services.

5.1. Reusable components and SAIFA-specific integration

Table 5.1 records the leading candidates and alternatives for each layer, together with the integration responsibility retained by SAIFA. The table is an evaluation baseline, not a final architecture.

Table 5.1 Candidate components and the corresponding SAIFA-specific integration.

Layer or need	Candidate component(s)	SAIFA-specific integration
Portal and UX	Next.js/React (Nuxt/Vue)	SAIFA UI, BFF, tool integration
Identity and SSO	Keycloak (authentik, WSO2 IS)	role mapping, ABAC, account linking
Access policies	Open Policy Agent (Casbin, Cedar)	SAIFA RBAC/ABAC model
Courses (LMS)	Moodle (Open edX)	link to portal, quotas and lab exercises
Notebooks and IDE	JupyterHub + KubeSpawner (SlurmSpawner, code-server, JupyterLite)	launch from portal, quotas, dataset mount, audit
Dataset catalogue	service-catalogue/marketplace alternative: Waldur | CKAN or Dataverse (DataHub, OpenMetadata)	unified catalogue, ACL, federation adapters
Search	OpenSearch + pgvector (PostgreSQL FTS, Qdrant)	shared index and ACL filtering
Model registry and experiments	MLflow (ClearML, Aim, W&B)	model cards, approval, catalogue, access rights
LLM inference	vLLM (TGI, SGLang, Triton)	API proxy, quotas, routing, audit
Model serving orchestration	KServe (Ray Serve, BentoML)	serving profiles and deployment rules

Fine-tuning and training	Hugging Face TRL/PEFT (Axolotl, LLaMA-Factory)	approved templates, HPC submit, result registration
HPC jobs	allocation/accounting: Waldur | SLURM on existing clusters; Open OnDemand as a complement	HPC bridge and a unified job model
Kubernetes / batch	Kubernetes Jobs (Volcano, Kueue)	routing between Kubernetes and SLURM
Workflows	Argo Workflows (Airflow, Flyte, Kubeflow)	simple SAIFA workflow templates
Containers	Harbor + Trivy + BuildKit/Buildah; Apptainer on HPC (Quay, Kaniko)	image approval and HPC conversion
Storage	PostgreSQL (source of truth), S3 (MinIO/Ceph) for objects, Valkey/Redis for cache	SAIFA domains, metadata links, lifecycle
Observability	Prometheus + Grafana + Loki + OpenTelemetry	SAIFA dashboards and correlation with user/job
Audit	append-only (PostgreSQL/OpenSearch) + WORM archive (immudb)	shared audit events and policy
Secrets	HashiCorp Vault / OpenBao	access policies, rotation, HPC SSH certificates
Notifications and status	Novu/Gotify + email; Uptime Kuma for status	events, preferences, public SAIFA status
Ticketing	Zammad or GLPI (GitLab Issues for PoC)	portal button and context from the audit log
Collaboration and code	GitLab CE (Gitea/Forgejo) + Nextcloud	rights over SAIFA resources and repo-workflow linkage
Benchmarking and quality	lm-evaluation-harness + Great Expectations	SAIFA benchmark definitions, leaderboard, rules
Lineage	OpenLineage + Marquez (DataHub)	linking job-dataset-model-result
Federation	cross-site allocation: Waldur marketplace | DCAT/DCAT-AP + custom adapters (CKAN Harvester)	Pharos/IT4LIA adapters and schema mapping
Quota and accounting	own simple model (Waldur)	portal, reservations, unified accounting per factory

BI and reporting	Grafana -> Superset/Metabase	SAIFA metrics and permissions
CI/CD and GitOps	GitLab CI + Argo CD (Flux, Tekton)	SAIFA pipelines and cluster configuration
Backup and DR	Velero + PostgreSQL backup + replication	RPO/RTO policies and restore tests

5.2. Cross-layer candidate and proof-of-concept baseline

Waldur is a cross-layer candidate because it may cover service cataloguing, organisation and project allocations, accounting, and HPC resource provisioning through one platform. It should be evaluated against a simpler combination of specialised components; adoption in one layer must not automatically determine its use in the others.

A practical proof-of-concept baseline is a web portal; federated identity; a policy or backend authorisation module; PostgreSQL and S3-compatible object storage; JupyterHub for notebooks; a managed inference service; SLURM integration through an isolated bridge; and Prometheus/Grafana-based observability. For every component, the proof of concept must show the source of truth, identity propagation, resource charging, audit recording, access-control enforcement and behaviour during dependency failure.

5.3. Technology-selection criteria

Selection uses two steps. A candidate must first pass the mandatory criteria for its technology domain. Candidates that pass are then compared using the weighted criteria in Table 5.2.

Table 5.2 Comparative technology-selection criteria.

ID	Criterion	Assessment focus
C1	User experience	Usability of the resulting capability, especially for non-technical users (sector experts, students, SME users).
C2	Integration effort	How much custom code and glue is needed to place the component behind the unified portal, identity, quota and audit.
C3	Operational complexity	Day-2 cost: deployment, upgrades, scaling, backup/restore and the on-call burden for a small team.
C4	Performance and cost	Throughput/latency under load and the compute, GPU and storage cost of running the component at national scale.
C5	Maturity and community	Release cadence, community size, documentation quality and long-term maintenance outlook.
C6	Compliance	GDPR alignment, data sovereignty, self-hosting options, licensing and procurement implications.

The principal domains and example mandatory criteria are summarised in Table 5.3. Detailed test questions should be maintained in the technology-selection specification and executed during the proofs of concept.

Table 5.3 Technology domains and examples of mandatory criteria.

Domain	Scope	Example mandatory criterion
Authentication and access control	login (local/institutional/EuroHPC), API keys, RBAC/ABAC, session revocation, service-to-service auth	A1 (federation over SAML2 and OIDC to one identity); A3 (immediate revocation)
Resource catalogue	discovery, publication, moderation, search with ACL/ABAC	K1 (ACL/ABAC filter at the index level)
Inference serving	serving runtime behind the gateway, quotas/rate limits	I1 (runtime behind the gateway); I4 (quota charged before execution)
HPC orchestration	job submission/monitoring, SLURM abstraction, quota reservation	H1 (SSH+sbatch); H2 (SLURM abstracted); H4 (quota before occupation)
Storage	object store behind API, resumable transfer, integrity, immutable versions	S1, S2, S5, S7
Experiment tracking	metric logging from cluster jobs, reproducibility	E1 (logging over a partially isolated connection); E6 (reproducibility)
Education	LMS behind API, locked templates, read-only data	D1-D7
Federation	CatalogueSource abstraction, provenance	F1 (new partner = new adapter); F2 (provenance, no overwrite)

5.4. Licensing and procurement

Licensing is part of the selection decision and must be checked for the exact product version and deployment model used in each proof of concept. The following rules apply:

- Prefer OSI-approved and self-hostable components where they meet the functional and operational requirements.
- Treat network-copyleft, source-available and proprietary products as explicit review items and document the operational and distribution obligations.
- Prefer open alternatives such as OpenBao and Valkey when equivalent source-available options create avoidable restrictions.
- Record any paid licence, subscription or cloud dependency as a procurement item with an owner and an exit option.
- Repeat the licence review before production deployment because licences and product terms may change.

5.5. Recommended decision statement

The listed technologies form a structured set of implementation candidates mapped to the Gateway needs. Final selection will be based on validated requirements, mandatory criteria, comparative proof-of-concept evidence, security and compliance review, operational sustainability, licensing and procurement implications, and compatibility with national and European AI Factory infrastructures. SAIFA will reuse mature platforms where possible and develop only the integration and governance capabilities that are specific to the AI Gateway.

6. Risks and Mitigation Measures

The design, implementation and operation of the SAIFA AI Gateway depend on technical infrastructure, external identity and computing services, domain data, organisational procedures and cooperation with national and European partners. Risk management must therefore be integrated with requirements validation, architecture decisions, implementation planning and release acceptance. The register below combines the project-level risks stated in the proposal with Gateway-specific risks identified during the initial design.

The register is an initial baseline. It must be updated when stakeholder validation changes a requirement, a proof of concept produces new evidence, an architectural decision is approved, an external dependency changes, an acceptance test fails, a new domain workflow is introduced, or a risk becomes an issue. Risk levels inherited from the proposal are retained; levels for Gateway-specific risks remain subject to formal confirmation.

6.1. Assessment and governance

Each record should identify the risk, category, cause, consequence, initial level, preventive controls, contingency measures, evidence, owner, related design items, residual level and status.

Until the project adopts a common likelihood-impact matrix, the High, Moderate and Low levels used in the proposal should not be converted into invented numerical scores.

The Gateway team, infrastructure operators, WP4 domain owners, WP5 support and training teams, WP1 quality management, and the Legal and Ethics Board contribute evidence and corrective actions within their responsibilities.

A function is not accepted when a successful functional test leaves an unaddressed high security, privacy, ethical or domain-validity risk.

6.2. Initial risk register

R-01 - Personnel turnover

Category: Organisational Initial level: High (proposal)

Risk. Loss of key technical or domain staff may delay integration and reduce continuity of operation and support.

Principal controls

- Maintain shared documentation and architecture decision records.
- Use code review, paired ownership and repeatable deployment procedures.
- Plan knowledge transfer and succession for critical components.

Acceptance or mitigation evidence. Evidence: current documentation, named primary and backup owners, and successful handover exercises.

R-02 - Availability of computational resources

Category: Infrastructure / external dependency Initial level: Low (proposal)

Risk. Demand on national or European infrastructure may limit the capacity available for development, demonstrations and large domain workflows.

Principal controls

- Define a beta workload that fits confirmed allocations.
- Separate interactive and long-running batch workloads.
- Estimate and reserve resources before dispatch.
- Use smaller reference datasets for functional acceptance and pursue additional national or EuroHPC allocations.

Acceptance or mitigation evidence. Evidence: confirmed allocation, reference job, queue visibility and repeatable benchmark results.

R-03 - Overly complex user interfaces

Category: Usability / adoption Initial level: Moderate (proposal)

Risk. A single Gateway serving many domains, roles and skill levels may become difficult to use and may increase support demand.

Principal controls

- Use role-adapted views, guided and expert paths, predefined templates and progressive disclosure.
- Test agreed scenarios with representatives of the intended user groups.
- Provide documentation and direct links to WP5 support and training.

Acceptance or mitigation evidence. Release condition: the relevant user group can complete the normal flow with the agreed level of assistance.

R-04 - Dataset quality and availability

Category: Domain / data Initial level: Moderate (proposal)

Risk. Available datasets may be incomplete, unsuitable, legally restricted or insufficiently representative for the intended workflows.

Principal controls

- Validate data ownership, quality, representativeness and permitted use before onboarding.
- Version datasets and record provenance, licence, limitations and quality evidence.
- Use curated or synthetic data where appropriate and approved.

Acceptance or mitigation evidence. Evidence: data card, quality checks, approval record and reproducible dataset version.

R-05 - Technological obsolescence

Category: Technology / sustainability Initial level: Low (proposal)

Risk. Selected frameworks, interfaces or products may become unsupported or incompatible during the project.

Principal controls

- Separate logical interfaces from product-specific adapters.
- Use open formats, time-boxed proofs of concept and architecture decision records.
- Define exit criteria and review dependency support regularly.

Acceptance or mitigation evidence. Evidence: documented alternatives, migration path and periodic support-status review.

R-06 - Data privacy, ethics and unlawful publication

Category: Compliance / security Initial level: High (proposal)

Risk. Personal, sensitive or high-impact data and models may be processed, shared or published without the required legal, ethical or governance controls.

Principal controls

- Classify data and default sensitive resources to controlled visibility.
- Record legal basis, consent and access conditions where required.
- Apply publication review, audit logging, compute-to-data restrictions and Legal and Ethics Board involvement.

Acceptance or mitigation evidence. Release condition: no public access or external export is enabled until the required evidence and approval are complete.

R-07 - Interoperability with external infrastructures

Category: External dependency / integration Initial level: Moderate (proposal)

Risk. Identity, metadata, policy or API incompatibilities with Pharos, IT4LIA or other services may delay federation and validation.

Principal controls

- Use stable adapter interfaces and standard metadata and authentication protocols.
- Begin with read-only or mocked integration, preserve source provenance and isolate partner failures.
- Maintain jointly agreed interface documentation and regular technical coordination.

Acceptance or mitigation evidence. Evidence: partner sandbox or mock tests, schema validation and graceful local operation during partner unavailability.

R-08 - Cross-tenant or cross-project information leakage

Category: Security Initial level: High (Gateway-specific)

Risk. Incorrect authorisation, catalogue filtering or storage policy may expose another user or project resource or metadata.

Principal controls

- Enforce server-side authorisation for every action and access-aware filtering in the catalogue index.
- Separate catalogue visibility from artefact access and support immediate credential revocation.
- Run negative, direct-URL and cross-project access tests.

Acceptance or mitigation evidence. Release blocker: any confirmed unauthorised cross-tenant or cross-project access.

R-09 - Dependence on HPC operators and cluster interfaces

Category: Operations / external dependency Initial level: High (Gateway-specific)

Risk. Job submission, monitoring and accounting depend on cluster policies and interfaces outside the direct control of the Gateway team.

Principal controls

- Confirm one minimal supported job path first.
- Document the contract with each operator and keep cluster-specific logic in adapters.
- Use a test queue, reference job and reconciliation after connectivity loss.

Acceptance or mitigation evidence. Evidence: repeatable submission-monitor-retrieve test on each supported backend and agreed responsibility matrix.

R-10 - Loss of integrity during large transfers

Category: Technical / data integrity Initial level: Moderate (Gateway-specific)

Risk. Interrupted transfer of datasets or models may create incomplete, duplicated or corrupted artefacts.

Principal controls

- Use resumable multipart transfer, checksums, temporary state and immutable versions.
- Prevent serving or publication before integrity verification and clean abandoned partial transfers.

Acceptance or mitigation evidence. Evidence: representative interrupted-transfer and checksum-failure acceptance tests.

R-11 - Premature workflow-orchestration complexity

Category: Architecture / schedule Initial level: Moderate (Gateway-specific)

Risk. Introducing a general workflow engine before validated needs are known may delay the beta and increase operational burden.

Principal controls

- Support a reference linear workflow first.
- Separate workflow definition from execution and add advanced orchestration only after a proof of concept demonstrates the need.

Acceptance or mitigation evidence. Evidence: validated workflow requirements and comparison of simple and general orchestration options.

R-12 - Incorrect or stale monitoring information

Category: Operations Initial level: Low (Gateway-specific)

Risk. A status view may incorrectly report a service or job as healthy, failed or available.

Principal controls

- Use independent health checks, timestamps and clear operational/degraded/unavailable states.

- Test status propagation and allow operator annotation of incidents.

Acceptance or mitigation evidence. Evidence: comparison of alerts and status displays with controlled incidents and actual backend state.

6.3. Release and review rule

Every beta capability must be linked to its applicable risks, controls, acceptance evidence and accountable owner. Functional success alone does not justify release when a high residual risk remains. The register should be reviewed at each implementation gate and whenever external infrastructure, data conditions or legal and ethical requirements change.

7. Roadmap and Timeline

The AI Gateway is developed in Task 3.4 from M01 to M36. Deliverable D3.1, Design of the AI gateway, is due in M03. With the project starting on 1 May 2026, the principal month markers are M03 = July 2026, M12 = April 2027, M18 = October 2027 and M36 = April 2029. The contractual Gateway milestones and deliverables are separated below from the indicative internal implementation stages.

7.1. Indicative implementation stages

The stages in Table 7.1 organise the broad target scope into an implementable sequence. They are planning stages rather than additional contractual deliverables. Priorities may be adjusted after stakeholder validation, proof-of-concept results, infrastructure constraints and risk review.

Table 7.1 Indicative AI Gateway implementation stages.

Stage	Indicative window	Primary focus	Contractual anchor
Initial design	M01-M03	Consolidate the project context, domain and system use cases, candidate technologies, roadmap and risks.	D3.1 and MS01 in M03.
Beta implementation	M04-M12	Implement and integrate the minimum coherent user path: identity and access, portal and catalogue, basic data/model access, basic inference or job execution, monitoring, notifications, and links to support and training.	MS10 and MS11 in M12.
Functionality consolidation	M13-M18	Extend and validate Gateway functionalities and features, integrate selected domain workflows, strengthen monitoring and model/data lifecycle support, and execute federation proofs of concept.	D3.4 in M18.
Continued deployment and validation	M19-M36	Continue hardening, operational improvement, interoperability, domain integration, user support, training integration, validation and benchmarking.	T3.4 and T3.5 continue through M36; no separate M36 Gateway deliverable is listed in the Annex.

7.2. Contractual deliverables and milestones

The deliverables and milestones that directly frame the Gateway are listed in Table 7.2. They are taken from the SAIFA implementation details; the wider WP3 plan is shown in Figure 7.1.

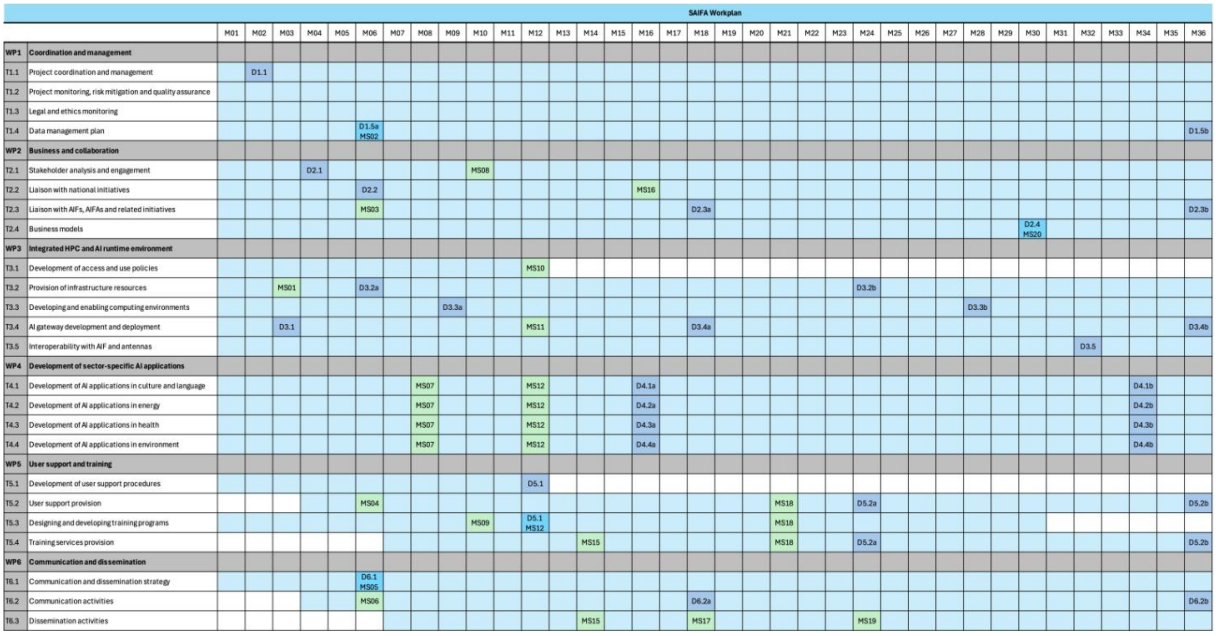


Figure 7.1 SAIFA implementation plan, highlighting WP3 and Task 3.4.

Table 7.2 Contractual deliverables and milestones relevant to the AI Gateway.

Figure ID	Deliverable or milestone	WP / Task	Due	Means of verification
D3.1	Design of the AI gateway	WP3 / T3.4	M03 (Jul 2026)	Comprehensive development plan submitted as a public deliverable.
MS01	Access to infrastructure provided	WP3	M03 (Jul 2026)	User accounts created on the national infrastructure resources.
MS10	Access and use policies	WP3	M12 (Apr 2027)	Access and use policies available online.
MS11	Beta version of AI gateway	WP3 / T3.4	M12 (Apr 2027)	AI Gateway available online.
D3.4	AI gateway functionalities and features	WP3 / T3.4	M18 (Oct 2027)	Detailed overview of implemented Gateway functionalities and features submitted.

7.3. Testing and validation plan

Testing follows the two-stage model described in WP3: development, integration and preliminary validation on the national infrastructure, followed by final testing and performance benchmarking using the Greek and Italian AI Factory infrastructures. The principal activities are:

- Continuous development and integration testing (M01-M36). Gateway services and domain workflows are integrated and tested on PARADOX and the ITE AI infrastructure provided through T3.2 and T3.3.
- Internal validation before wider access. Consortium users, WP4 domain owners and WP5 support and training teams validate the agreed scenarios before they are opened to broader user groups.
- Beta validation (M12). MS11 demonstrates an online beta version together with the access and use policies delivered under MS10. The beta must support at least one coherent end-to-end path through identity, discovery, execution and monitoring.
- Functionality and feature validation (M18). D3.4 records the implemented functions and evidence obtained from integration, user and interoperability testing.
- Greek and Italian infrastructure testing. Access policies and integration protocols developed with the Pharos and IT4LIA teams are used for final testing and performance benchmarking of selected applications and Gateway interactions.
- Security, compliance and operational acceptance. Access control, auditability, data governance, transfer integrity, recovery and relevant legal and ethical conditions are verified before a function is accepted for external use.

8. Conclusion

This deliverable defines the SAIFA AI Gateway as a modular integration and control layer that connects users with national and European computing resources, datasets, pre-trained models, tools, workflows, training and support. It presents 21 Domain Use Stories and a representative set of System Use Cases, identifies candidate technologies, and aligns implementation with risk and validation controls.

The design follows a reuse-oriented approach. Mature systems should provide identity, cataloguing, notebooks, execution, model and data management, storage, monitoring and learning functions, while SAIFA-specific development should focus on the unified portal, common metadata and access rules, resource accounting, audit, the HPC bridge and federation adapters. Candidate technologies remain subject to proof-of-concept evaluation and are not final procurement or architecture decisions.

The immediate path is to complete D3.1 in M03, confirm access to national infrastructure, validate and prioritise the beta baseline, and implement the online beta and access policies by M12. D3.4 in M18 will document the implemented Gateway functionalities and features. Task 3.4 then continues through M36 with further integration, hardening, interoperability, user support, training integration, testing and operational improvement.

The design remains intentionally adaptable and does not commit SAIFA to implement every described scenario or assign all processing to the Gateway. Architecture decisions, release scope, responsibility allocation and acceptance evidence should be updated as stakeholder needs are validated, proofs of concept are completed, and interfaces with PARADOX, ITE, Pharos, IT4LIA and other European AI services become sufficiently defined.