DELIVERABLE D4.1

THEMIS 5.0 Trustworthiness Optimisation Platform Architecture and 1st release incl. Benchmark Training Datasets

DELIVERABLE DETAILS		
SUBMISSION DATE	NAME OF THE DELIVERABLE	WORK PACKAGE
31/03/2025	THEMIS 5.0 Trustworthiness Optimisation Platform Architecture and 1st release incl. Benchmark Training Datasets	WP4
DISSEMINATION LEVEL	AUTHOR(S)	LEAD BENEFICIARY
Public	ENG, ICCS, ATC, SINTEF, UoS, trustilio, IBM, MAG, I4RI, KUL also with all partners' contribution and input.	ENG

PROJECT DETAILS			
PROJECT ACRONYM	THEMIS 5.0	GRANT AGREEMENT	101121042
CALL IDENTIFIER	Human-02-01	PROJECT DURATION	01.10.2023 – 30.09.2026
PROJECT OFFICER	Anna Puig Centelles	PROJECT COORDINATOR	Maggioli SpA

THEMIS 5.0

QUALITY CONTROL ASSESSMENT				
VERSION	DATE	DESCRIPTION	NAME	ORG
0.1	16th Jan 2025	Table of contents	Rosario Catelli	ENG
0.5	28th Feb 2025	1st draft	Rosario Catelli	ENG
0.8	14th Mar 2025	Reviewed version	Arnaud Billion Giannis Stamatellos Faidra Alevizou	IBM IPT IPT
0.9	21st Mar 2025	Final changes and updates	Rosario Catelli	ENG
1.0	31st Mar 2025	Submission ready	Rosario Catelli	ENG

LIst of authors

AUTHORS NAMES	ORG
Rosario Catelli, Davide Profeta	ENG
Mattheos Fikardos, Katerina Lepenioti, Dimitris Apostolou, Gregoris Mentzas	ICCS
Adam Doulgerakis, Eleni Tsalapati, Giorgos Giotis	ATC
Brian Elvesæter, Pål Vegard Johnsen, Joel Bjervig	SINTEF
Giannis Stamatellos, Alexandra Psaltidou	IPT
Andrea Palumbo	KUL
Steve Taylor, Samuel Senior	UoS
Theo Fotis	trustilio
Arnaud Billion	IBM
George Karavokiros	MAG
Diego Argüello Ron	I4RI

DISCLAIMER

STATEMENT OF ORIGINALITY

This Deliverable contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both.

ACKNOWLEDGEMENT

This document is a deliverable of the THEMIS 5.0 project, which has received funding from the European Union's Horizon Europe research and innovation programme under Grant Agreement No. 101121042.

Table of contents

EXECUTIVE SUMMARY 9

1. INTRODUCTION 10

1.1. Purpose and scope 11
1.2. Contribution to other Deliverables 12
1.3. Structure of the Document 12

2. THEMIS 5.0 SERVICES, PLATFORM & ARCHITECTURE SPECS 13

2.1. Agile design and development approach 13
2.2. Architectural design 14
2.2.1. System Context level 14
2.2.2. Container level 15
2.2.3. Component level 27
2.3. Ensuring compliance with the GDPR and the AI Act in the design stage (KUL) 36
2.3.1. Introduction and scope of analysis 37
2.3.2. Curation of benchmark training datasets in compliance with the GDPR 37
2.3.3. Compliance with data quality and transparency requirements under the AI Act 39
2.4. Multilingualism support 39
2.4.1. i18next: hands on guidelines 40
2.4.2. Specific solutions for conversational agents 43

3. DEVELOP TRAINED AI MODELS FOR IMPLEMENTING THE THEMIS 5.0 TRUSTWORTHY SERVICES 45

3.1. Trustworthiness Assessment 45
3.1.1. Overview 45
3.1.2. Algorithm details 45
3.2. Decision impact of socio-technical system 49
3.2.1. Overview 49
3.3. Persona Predictor 49
3.3.1. Overview 49
3.3.2. Algorithm details 50
3.4. Risk Evaluation 51
3.4.1. Overview 51
3.4.2. Algorithm details 51
3.4.3. Technical Specifications 52
3.5. Solution Recommendation 52
3.5.1. Overview 52
3.5.2. Algorithm details 52
3.5.3. Technical Specifications 53
3.6. Conversational Agents 54
3.6.1. Persona conversational agent 54
3.6.2. Conversational Agent for Socio-Technical Evaluation 57

4. CO-CREATION OF BENCHMARKED DATA SETS AND TRAINING OF THEMIS 5.0 AI MODELS 58

4.1. Introduction 58
4.1.1. ICE/i4RI's Role as Facilitator for Trustworthiness Models 58
4.2. Stakeholder Engagement and Co-Creation Process 59
4.2.1. Identification of Stakeholders (ICCS, Domain Experts, TAIME Developers) 59
4.2.2. Integrating Moral, Legal, and Socio-technical Feedback into Dataset Creation 60
4.2.3. Governance Models for Deciding on Protected Attributes and Fairness Criteria 61
4.3. Benchmark Data Curation 61
4.3.1. Data Sources Relevant to THEMIS Trustworthiness Models (Fairness, Accuracy, Robustness) 62
4.3.2. Criteria for Dataset Updates Based on Performance Feedback & Assessments 63
4.4. Technical Infrastructure & Integration 64
4.4.1. ICE/i4RI Environment Overview (Containerization, CI/CD, APIs) 64
4.4.2. Integration Points with TAIME Models for Retraining 64
4.4.3. Dataset Versioning, Documentation, and Management Practices 65

4.5. Trustworthiness Metrics & Retraining Triggers ... 66
4.5.1. KPIs and Thresholds Defined by ICCS/SINTEF/IBM ... 66
4.5.2. Automated Dataset Refinement or Retraining Triggers ... 67
4.5.3. Incorporating TAI Cards and Assessment Results into Retraining Decisions ... 68
4.6. Ensuring Fairness, Non-Discrimination, and Legal Compliance ... 69
4.6.1. Adherence to GDPR and Additional Compliance Requirements ... 69
4.6.2. Strategies for Handling Morally Sensitive Data ... 70
4.6.3. Privacy-by-Design Principles and Audit Trails for Dataset Usage ... 70
4.7. Explainability & Transparency ... 71
4.7.1. Tools and Techniques (e.g., SHAP) for Explainable Datasets ... 72
4.7.2. Methods for Documenting Decisions and Data Transformations ... 72
4.7.3. Reporting Processes for Stakeholders to Understand Dataset Rationale ... 72
4.8. Quality Assurance & Validation ... 73
4.8.1. Data Validation Checks, Bias Detection, and Error Analysis ... 73
4.8.2. Continuous Monitoring and Feedback ... 74
4.9. Roadmap for Iterations & Future Work ... 76
4.9.1. Mechanisms for Iterative Improvement as Trustworthiness Criteria Evolve ... 76
4.9.2. Scalability Considerations and Long-term Maintenance Plans ... 76
4.9.3. Engaging Stakeholders in Ongoing Dataset Refinement ... 77

5. DEVELOPMENT OF THEMIS 5.0 ECOSYSTEM PLATFORM ... 77

5.1. SUT Common: quick usage of its features for fast prototyping of AI models from scratch ... 77
5.1.1. Workflow Management Overview ... 77
5.1.2. Data Insights and Analysis ... 79
5.1.3. Specifications about the backend ... 80
5.2. Cloud-edge resources management through K3S ... 83
5.2.1. K3s: key features and use cases ... 83
5.2.2. K3s: architectural overview ... 83

6. CONTINUOUS SYSTEM INTEGRATION, TESTING AND QUALITY ASSURANCE ... 86

6.1. CI\CD: THEMIS 5.0 approach ... 86
6.1.1. Technologies ... 88
6.1.2. DevOps Guidelines for THEMIS 5.0 ... 90

7. CONCLUSION ... 95

8. ANNEX 1: C4 MODEL – USE AND ADAPTATION IN THEMIS 5.0 PROJECT ... 96

9. ANNEX 2: METHOD CARDS ... 98

9.1. SafetyCage – Misclassification detection ... 98
9.1.1. Maximum Softmax Probability (MSP) ... 100
9.1.2. DOCTOR ... 102
9.1.3. Mahalanobis ... 104
9.1.4. SPARDACUS ... 106
9.1.5. Misclassification similarity explanations ... 108
9.2. OODGuard – Out-of-distribution (OOD) detection ... 109
9.2.1. Counterfactual explanations ... 111
9.3. FairnessEvaluator ... 113

10. ANNEX 3: USER REQUIREMENTS RESULT FROM LIVING LABS ... 115

10.1. Presentation of results on user requirements ... 115
10.2. Elaboration on recommendations for implementation of user requirements ... 116

11. ANNEX 4: UPDATED SYSTEM & USER REQUIREMENTS ... 117

11.1. System Requirements ... 117
11.2. User Requirements ... 119

12. ANNEX 5: PERSONA IDENTIFICATION METHODOLOGY ... 133

List of figures

Figure 1: Mapping of the conceptual framework to software components 11
Figure 2: System Context diagram of the THEMIS 5.0 platform 14
Figure 3: Container diagram of the TAUPOS 16
Figure 4: Container diagram of the TAIME 18
Figure 5: Container diagram of the SUT in the healthcare domain 20
Figure 6: Container diagram of the SUT in the critical infrastructure domain 22
Figure 7: Container diagram of the SUT in the disinformation in cyberspace domain 24
Figure 8: Container diagram of the SUT Common 25
Figure 9: Component diagram of the TAUPOS Persona Analyser 28
Figure 10: Component diagram of the TAUPOS TAI Lifecycle Actuator 29
Figure 11: Subcomponent diagram of the TAI Lifecycle Actuator Quantitative Assessor 31
Figure 12: Component diagram of the TAUPOS Risk Assessment\Spyderisk 34
Figure 13: Component diagram of the Super Decision Engine 35
Figure 14: Class description for the classifier of the Persona module 50
Figure 15: Class description for the classifier of the Disambiguation module 50
Figure 16: The pipeline of the Persona conversational agent 54
Figure 17: Chatbot's questions and conversation path, structured on JSON format files 55
Figure 18: The chatbot user interface 57
Figure 19: SUT Common workbench overview 78
Figure 20: SUT Common workbench – Example of workflow 79
Figure 21: SUT Common workbench – Example of dataset registration and selection 79
Figure 22: SUT Common workbench – Example of data filtering 80
Figure 23: SUT Common workbench – Example of data translation and statistical analysis visualisation 80
Figure 24: SUT Common workflow – Evaluation of design capabilities 82
Figure 25: SUT Common workflow – Evaluation of translation capabilities 82
Figure 26: SUT Common workflow – Evaluation of execution monitoring capabilities 82
Figure 27: SUT Common workflow – Evaluation of data interaction capabilities 83
Figure 28: CI\CD process from developer perspective 87
Figure 29: CI\CD process from DevOps perspective 87
Figure 30: ArgoCD – Example Application 89
Figure 31: C4 model abstractions 96
Figure 32: C4 model graphical elements 97

List of tables

Table 1: System Context – THEMIS 5.0 platform interactions 15
Table 2: Container – TAUPOS interactions 17
Table 3: Container – TAIME interactions 18
Table 4: Container – Healthcare domain SUT interactions 21
Table 5: Container – Critical infrastructure domain SUT interactions 23
Table 6: Container – Disinformation in cyberspace domain SUT interactions 24
Table 7: Container – SUT Common interactions 26
Table 8: Component – TAUPOS Persona Analyser interactions 28
Table 9: Component – TAUPOS TAI Lifecycle Actuator interactions 30
Table 10: Subcomponent – TAI Lifecycle Actuator Quantitative Assessor interactions 32
Table 11: Subcomponent – TAI Lifecycle Actuator Solution Navigator interactions 33
Table 12: Component – TAUPOS Risk Assessment (Spyderisk) interactions 35
Table 13: Component – TAUPOS Super Decision Engine interactions 36
Table 14: Trustworthiness Assessment – Fairness specification 45
Table 15: Trustworthiness Assessment – Accuracy specification 47
Table 16: Trustworthiness Assessment – Robustness specification 48
Table 17: Decision impact of socio-technical system specification 49
Table 18: Persona Predictor specification 51
Table 19: Risk evaluation specification 52
Table 20: Solution Recommendation specification 53
Table 21: A snapshot of stored chatbot questions, aligned with user responses and a target conversation group 56
Table 22: K3s and Kubernetes differences at a glance 85
Table 23: Method card for SafetyCage – misclassification detection 98
Table 24: Model card for Maximum Softmax Probability (MSP) 100
Table 25: Model card for DOCTOR 102
Table 26: Model card for Mahalanobis 104
Table 27: Model card for SPARDACUS 106
Table 28: Model card for misclassification similarity explanations 108
Table 29: Method card for OODGuard – Out-of-distribution (OOD) detection 109
Table 30: Model card for counterfactual explanations 111
Table 31: Model card for the FairnessEvaluator 113
Table 32: User requirements score 115
Table 33: System Requirements 117
Table 34: User Requirements 120
Table 35: Users Extended Characteristics 135
Table 36: Mapping and Clustering Matrix 141
Table 37: Mapping to clusters 143
Table 38: Probing Question to resolve conflicting TW allocation 144

Terms and Abbreviations

Abbreviation	Description
ABAC	Attribute-Based Access Control
AIS	Automatic Identification System
CA	Conversational Agent
CD	Continuous Development
CI	Continuous Integration
CNL	Controlled-Natural Language
DWT	Dead Weight Tonnage
ERB	Ethical Review Board
ETA	Estimated Time of Arrival
IDM	Identity Management
KG	Knowledge Graph
LIME	Local Interpretable Model-agnostic Explanations
LLM	Large Language Models
MCDM	Multi-criteria Decision-Making
MFA	Multi-Factor Authentication
PbD	Privacy-by-Design
PC	Pancreatic Cancer
PCS	Port Community System
RBAC	Role-Based Access Management
SHAP	SHapley Additive exPlanations
SMPC	Secure Multi-Party Computation
SUT	System Under Test
TAI	Trustworthy AI
TAIME	Trustworthy AI Models Engine
TAUPOS	Trustworthiness Assessor, User Profiler and Optimisation Suggester
TC	Trustworthy Characteristics
TOP	Trustworthiness Optimization Process
TPO	Trustworthiness & Profiling Optimizer
UI	User Interface
WP	Work Package
ZKP	Zero-knowledge Proofs

EXECUTIVE SUMMARY

The THEMIS 5.0 project aims to optimize trustworthiness in AI systems by developing a robust architecture and a structured framework for evaluating and enhancing AI trustworthiness. This deliverable, D4.1, presents the first release of the THEMIS 5.0 Platform, outlining its system architecture.

Ensuring AI trustworthiness is critical, as AI-driven solutions increasingly influence decision-making processes in various domains, including healthcare, critical infrastructure, and disinformation in cyberspace. Addressing concerns related to accuracy, fairness, and robustness, THEMIS 5.0 provides a structured methodology for assessing and improving AI models.

The deliverable details the design and implementation of a trustworthiness optimization framework, aligning with the conceptual models defined in prior project phases. It provides a structured approach to evaluating AI trustworthiness through quantitative and qualitative assessments. In defining the approach, partners have taken into account the provisions of the GDPR applicable to the activities carried out and foreseen under this deliverable, as well as the requirements that may apply in the future under the AI Act to the platform once it is placed on the market or put into service.

The THEMIS 5.0 platform is built using an agile design methodology, integrating modular software components based on the C4 model. The system architecture comprises the Trustworthiness Assessor, User Profiler, and Optimization Suggester (TAUPOS) and the Trustworthy AI Models Engine (TAIME) as architectural pillars. These components work together to assess AI models and recommend enhancements. Key features include:

- trustworthiness assessment, to evaluate fairness, accuracy, and robustness using benchmark training datasets;
- risk management and compliance, to ensure alignment with GDPR and AI Act requirements;
- multi-domain applicability, to supports diverse use cases, including healthcare, critical infrastructure and disinformation domains;
- continuous improvement mechanism, to use real-world feedback for ongoing model refinement.

The initial release of the THEMIS 5.0 platform successfully develops AI model assessment and improvement functionalities, such as development of a flexible and scalable architecture, implementation of trustworthiness assessment tools, compliance with ethical and legal frameworks, validation through pilot use cases.

The THEMIS 5.0 Platform provides a comprehensive framework for assessing and enhancing AI trustworthiness. Future work will focus on refining assessment methodologies, expanding dataset benchmarks, and better integrating functionalities and real-world use cases to further improve AI trustworthiness.

1. INTRODUCTION

This deliverable stems from the need for the THEMIS 5.0 project to structure and develop a system architecture within Work Package 4 that responds to the conceptual architecture theorized by Work Package 2. Specifically, the processes in place within Work Package 4 apply to the delicate phase of translating and transforming the broad and general requirements gathered, including through the efforts of Work Package 3, in order to implement a trustworthiness optimization process. While it proves to be absolutely necessary to remain on a general level of abstraction that enables the use of the THEMIS 5.0 platform in the most varied scenarios, it becomes particularly useful to have the possibility to establish a comparison with project use cases, which are totally different from each other, so that it is possible to have a reference for the elasticity expected from the platform in the operational phase.

To date, there is no one-size-fits-all solution regarding the measurement of trustworthiness: as already mentioned by Work Package 2, the broadening of the definition of the trustworthiness concept does not allow for taking into account all possible technical aspects that can be involved and therefore requires their restriction, which, in the case of the THEMIS 5.0 project, puts the focus on accuracy, robustness and fairness from a quantitative point of view and socio-technical aspects from a qualitative point of view. In order to cover all the different aspects involved, the following chapters provide an overview of all the topics covered in Work Package 4.

Chapter 2 details the architectural diagrams of the THEMIS 5.0 platform that go to meet the design specifications and requirements, on the basis of which the refinement and development work of the software related to it may evolve taking into account the inputs from Work Packages 2 and 3. At the same time, although they will be better fine-tuned at a later stage of software component integration, requirements related to both privacy and security (e.g., by providing both appropriate authentication and control mechanisms and by reducing the transit of privacy-sensitive data to the bare minimum or avoiding it altogether where possible) and multilingual support are taken into account during the platform design phase. Chapter 2 also provides an overview of the EU legal obligations that apply to the activities carried out under Work Package 4, as well as of the obligations that may apply to the platform in the future.

Chapter 3 provides a detailed view on the underlying algorithms utilized by the THEMIS 5.0 architecture to interpret data and assist with the decision-making process. The algorithms are grouped in sections following the decision-process described in D2.2, that include detail descriptions of their functionalities and specifications.

Chapter 4 focuses on the co-creation of benchmarked datasets and the training of THEMIS 5.0 AI models, a crucial process in ensuring trustworthiness, fairness, and robustness of the AI-driven decision support ecosystem. This chapter begins by outlining the role of ICE/i4RI as a facilitator for the trustworthiness models by providing an environment that supports containerized workflows, CI/CD pipelines, and automated dataset management. The stakeholder engagement and co-creation process are also detailed, highlighting how domain experts, developers, and legal scholars collaboratively define dataset requirements and fairness constraints. Additionally, dataset curation principles are described, specifying the sources of data relevant to THEMIS trustworthiness models and the criteria used for dataset updates based on performance feedback. Technical infrastructure and integration aspects are also explored, dataset versioning strategies, and governance models to ensure compliance with privacy and ethical standards. Furthermore, the chapter elaborates on trustworthiness metrics and retraining triggers, ensuring AI models remain aligned with predefined KPIs for fairness, accuracy, and robustness, while automated dataset refinements facilitate continuous improvement of AI model performance. Ethical, legal, and security considerations are also addressed, with adherence to GDPR and data protection-by-design principles, ensuring full transparency in data documentation and reporting processes. Lastly, the chapter concludes with a roadmap for iteration and future enhancements, ensuring the sustainability and scalability of the THEMIS 5.0 dataset curation and AI training process.

Chapter 5 illustrates the components of the THEMIS 5.0 platform that make it possible on the one hand to design from scratch AI models and employ the latest technologies to the user and on the other hand to manage cloud-edge infrastructural resources, hooking in some points to the results of tasks one through three.

Chapter 6 describes the guidelines to the continuous integration and delivery approach adopted in THEMIS 5.0, taking into account aspects of functional and integration testing possibilities with consequent validation objectives.

Finally, in chapter 7, with this initial release of the platform, conclusions are drawn and points of possible improvement in the future evolution are tried to be identified.

1.1. Purpose and scope

This deliverable is within the scope of software and infrastructure design and development, with the purpose of reporting the specifications of the first release of the THEMIS 5.0 platform, including the guidelines and constraints defined to ensure consistent development and integration, while presenting any early training datasets released, where available.

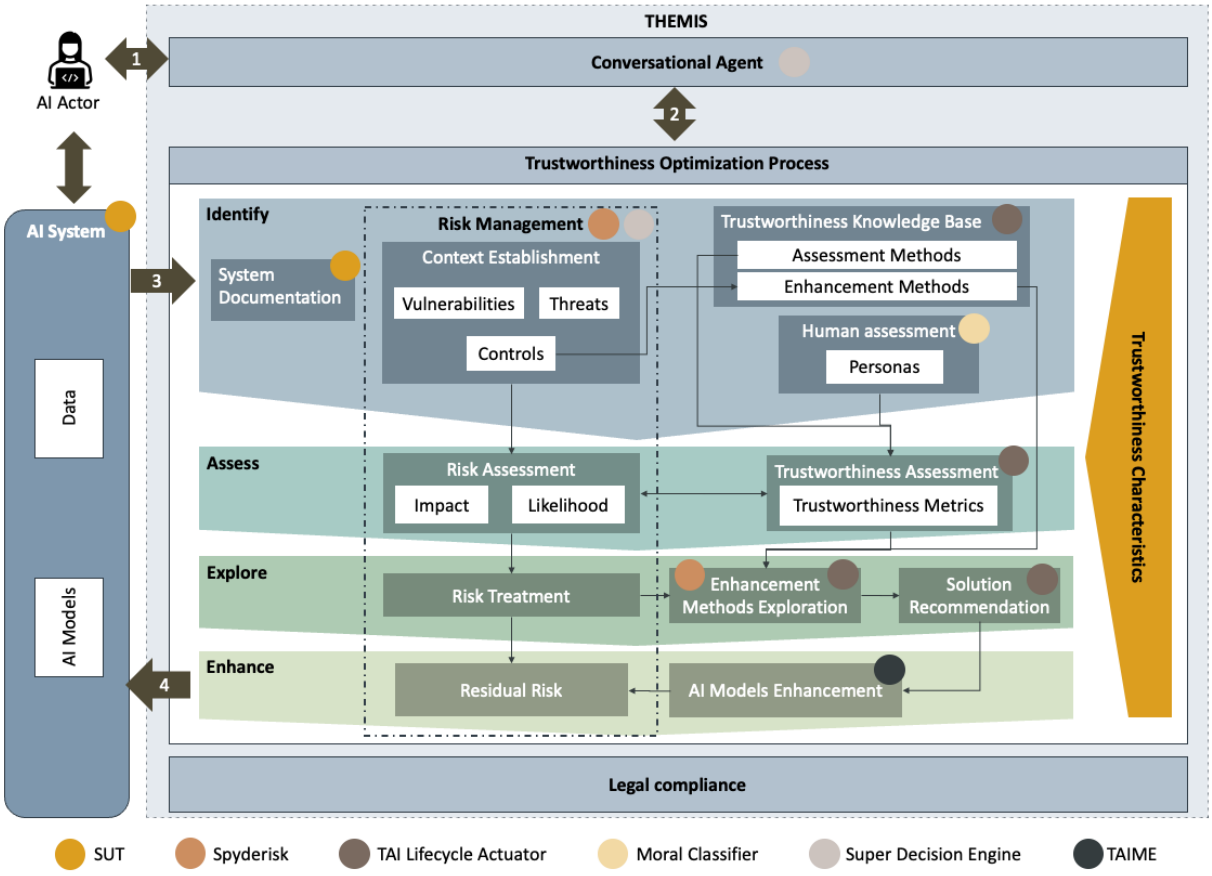

Figure 1: Mapping of the conceptual framework to software components

In this regard, Figure 1 shows the connection with the conceptual framework proposed by Work Package 2 and how it maps to software components developed within Work Package 4: at the bottom of the figure is a simple legend. In particular, the *System Under Test (SUT)* is shown as both *AI System* with its models and data whose trustworthiness wants to be assessed and *System Documentation* as starting stage of the *Identify* phase of the proposed *Trustworthiness Optimization Process (TOP)*.

This is where the *TAUPOS* software system come in: the latter is the acronym for *Trustworthiness Assessor, User Profiler and Optimisation Suggester*, one of the two main architectural pillars. In detail it is made up of four main components:

- *Spyderisk* and *Super Decision Engine* that contribute to the *Risk Management* phase, which is transversal to the whole TOP process;
- *TAI Lifecycle Actuator*, which owns the *Trustworthiness Knowledge Base* with specific *Assessment Methods* and *Enhancement Methods*, contributing to *Assess* and *Explore* phases in collaboration with Spyderisk;
- *Persona Analyser*, that preside over the *Human Assessment* stage within the *Identify* phase.

All TAUPOS components communicate and cooperate to support the decision processes and supply the *Solution Recommendation* as outcome of the *Explore* phase, downstream of the *Assessment* phase.

Finally, managing the *Residual Risk*, the *Enhance* phase leave floor to the *AI Models Enhancement* stage which is carried out by the other pillar of the architecture i.e. the *Trustworthy AI Models Engine (TAIME)*: based on the collected information, the final result of the THEMIS 5.0 platform is given: it provides suggestions and improvements over the assessed *Trustworthiness Characteristics* of the *AI System* to get better and trustworthier AI.

1.2. Contribution to other Deliverables

This deliverable builds on the theoretical and methodological efforts from Work Packages 2 and 3. In particular, Work Package 4 seeks to reflect in practical development what was outlined in the other two work packages, considering all those aspects related to the need to make the trustworthiness optimization process feasible and holding together all the inputs from the different stakeholders approached, so as to make the platform usable to both a broad and generalist and highly specialized audience.

1.3. Structure of the Document

This deliverable is organized as follows:

- Chapter 2, which covers aspects mainly related to Task 4.1.
- Chapter 3, which covers aspects mainly related to Task 4.2.
- Chapter 4, which covers aspects mainly related to Task 4.3.
- Chapter 5, which covers aspects mainly related to Task 4.4.
- Chapter 6, which covers aspects mainly related to Task 4.5.
- Chapter 7, which draws conclusions and tries to identify possible future areas for improvement

Finally, there are chapters devoted to annexes (referred to in previous chapters where appropriate).

2. THEMIS 5.0 SERVICES, PLATFORM & ARCHITECTURE SPECS

In this section, as previously mentioned, several key points of the project are addressed. In fact, based on several inputs from Work Packages 2 (see also Annex 4) and 3 (see also Annex 3), such as guidance on software development requirements and user preferences (for which references are provided in the appendix), the software architecture of the THEMIS 5.0 platform is developed, details of which are provided in the following sections regarding these aspects: the agile design and development approach, closely related to the C4 model employed (with appropriate adaptations) for the architectural diagrams that illustrate below the entire architecture of the constantly evolving THEMIS 5.0 platform, as well as the safeguards to comply by design and by default with the applicable legal framework on personal data processing and AI development, as well as multilingual support to make what is offered widely available.

2.1. Agile design and development approach

In the context of the project, the agile design and development methodology is closely related to the use of the C4 model (more details of which can be found in the Annex 1), which offers a great deal of flexibility in work. In particular, the C4 model fits well within agile development because it provides just enough architecture documentation without being overly rigid. Here's how it aligns with agile principles:

- Lightweight and incremental documentation:
  - agile values working software over comprehensive documentation, and C4 follows this by keeping diagrams simple and focused;
  - instead of creating all diagrams upfront (like in waterfall models), teams can evolve and update C4 diagrams iteratively as the system grows.
- Support collaboration and communication:
  - the Context and Container diagrams help cross-functional teams (developers, product owners, stakeholders) understand the system quickly;
  - instead of long documentation, agile teams can use C4 diagrams in sprint planning, backlog refinement, and design discussions.
- Enable Just-in-Time design:
  - agile promotes evolutionary architecture, meaning designs can change based on feedback;
  - teams can start with high-level Context/Container diagrams and refine them when needed (e.g., Component diagrams for complex parts).
- Work with agile artifacts
  - C4 diagrams can be integrated into user stories, epics, and sprint reviews to ensure clarity in system design;
  - can be combined with Architecture Decision Records to document design choices made during development.
- Support DevOps and CI/CD
  - Container and Component diagrams are useful for deployment strategies, microservices design, and monitoring architecture;
  - tools like Structurizr allow keeping C4 diagrams versioned alongside the codebase.

As can be imagined, this allows a solid basis to be established within the project which can then be appropriately extended within the development teams of the consortium partners and, of course, hook into other Work Packages where appropriate and necessary. For example, where development teams work in an agile workflow, C4 can be employed in this way:

1. Sprint zero/Inception: define a high-level Context and Container diagram to set the foundation;
2. During sprints: refine diagrams when necessary (e.g., add/update Components if new features impact architecture);
3. Sprint reviews: use diagrams to communicate changes to stakeholders and onboard new developers;
4. Continuous evolution: keep diagrams up to date with system changes but avoid over-documenting unnecessary details.

Finally, given the concurrence of WP4 with WP2 and WP3, the inputs of which are summarised in Annex 4 and Annex 3 respectively, the flexibility of the C4 model becomes a key element to manage future adaptations, tightening the broad requirements from the other WPs where necessary.

2.2. Architectural design

In this section, all the architectural diagrams constituting the THEMIS 5.0 platform developed during the first phase of the project are described, starting with the highest level (System Context level) and ending with the more detailed ones (Component level). Although the key concepts are rather clearly defined, refinements may be necessary for all diagrams in this section during the second part of the project, in order to better match the evolution of the THEMIS 5.0 platform architecture.

2.2.1. System Context level

The System Context level diagram is the highest-level diagram among those available to describe the architecture of the THEMIS 5.0 platform, that is designed to offer users and developers the possibility of verifying certain trustworthiness aspects of AI systems, both from a quantitative and qualitative point of view. Its purpose is to try to provide, at a glance, the conceptual idea underlying the realisation of the platform itself, letting it be understood which are the main actors at play and which are the main interacting software systems. With particular reference to the figure below, the fundamental building blocks and their interactions are better detailed below.

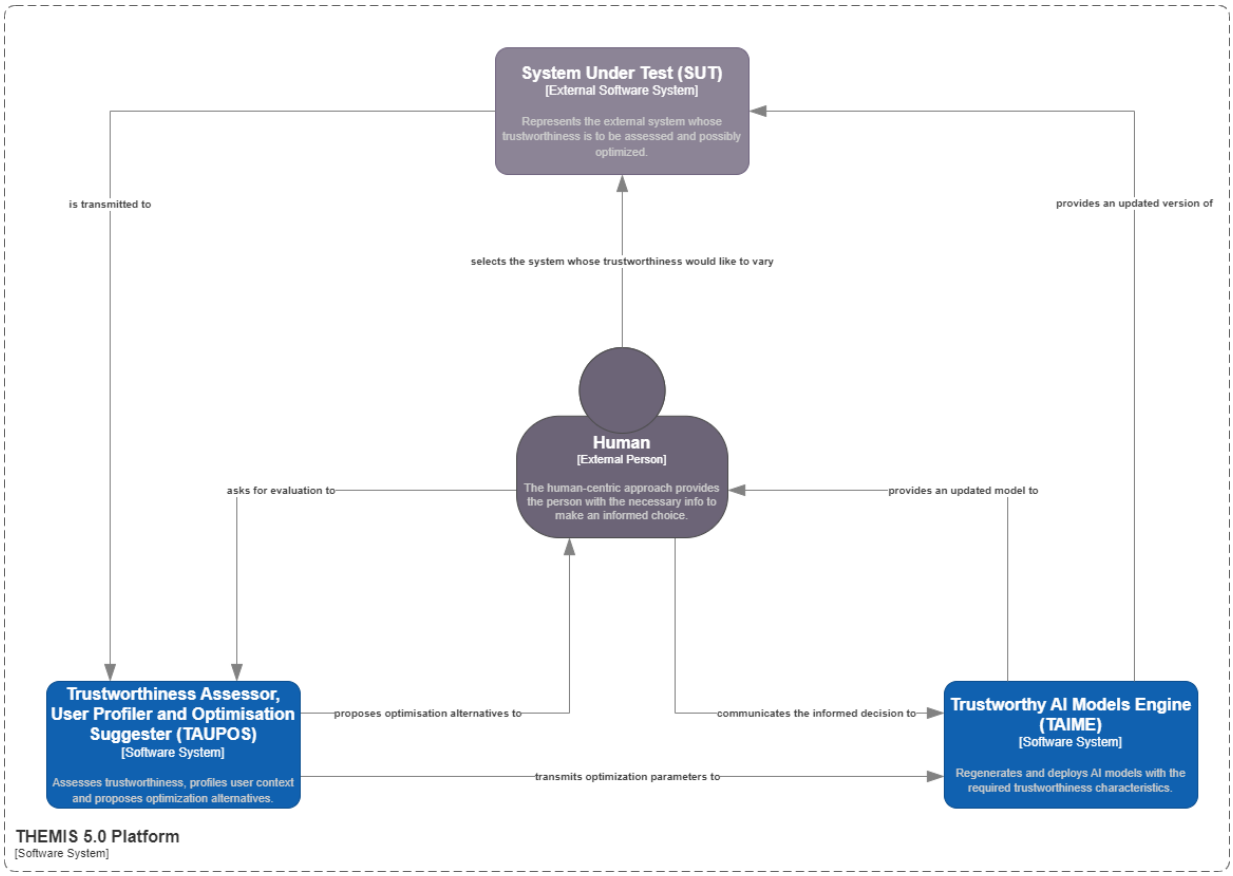

Figure 2: System Context diagram of the THEMIS 5.0 platform

2.2.1.1 Building Blocks description

The main building blocks of the THEMIS 5.0 platform are:

- **Human**. The human being is the main actor: they use the platform and proceed through the processes it provides. In addition, from the modelling point of view, the human is regarded as external to the platform itself: thus, a criterion of usability of the platform also becomes part of the entire trustworthiness assessment process.
- **Trustworthiness Assessor, User Profiler and Optimisation Suggester (TAUPOS)**. It is one of the two main building blocks constituting the THEMIS 5.0 platform: it assesses the trustworthiness of the given AI systems, profiles the user context and proposes trustworthiness optimization alternatives.
- **Trustworthy AI Models Engine (TAIME)**. It is the other of the two main building blocks constituting the THEMIS 5.0 platform: it regenerates and deploys AI models with the required trustworthiness characteristics.
- **System Under Test (SUT)**. It represents all external software systems both from use cases and from scratch design that can be assessed, and hopefully optimised, through the THEMIS 5.0 platform.

2.2.1.2 Interactions description

The following table specifies the I/O interactions of the System Context between its constituent building blocks: since this is the highest-level diagram, this table is quite simple and there are no external building blocks to be matched unlike the similar tables in the following sections. In particular, the table specifies the interactions among the highest-level building blocks, i.e. Human, TAUPOS, TAIME, and SUT, to verify with what is also documented in the diagram of the previous figure in which the I/O actions carried out are made explicit. This kind of table will assume a more informative role with the following diagrams, reporting what is illustrated with further details when needed.

Table 1: System Context – THEMIS 5.0 platform interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
Human	-	TAUPOS	-	TAUPOS
	-	TAIME	-	TAIME
	-	-	-	SUT
TAUPOS	-	Human	-	Human
	-	SUT	-	-
	-	-	-	TAIME
TAIME	-	Human	-	Human
	-	TAUPOS	-	-
	-	-	-	SUT
SUT	-	Human	-	-
	-	-	-	TAUPOS
	-	TAIME	-	-

2.2.2. Container level

The Container level diagrams are the second-level diagrams following the C4 model. They detail TAUPOS and TAIME, that are the two main pillars of the THEMIS 5.0 platform, and the SUT in its four iterations related to the three uses cases of the project plus the from scratch design capabilities.

2.2.2.1 *Container TAUPOS*

TAUPOS constitutes the foundation of the trustworthiness verification process. As anticipated in the introduction, a considerable part of the Trustworthiness Optimization Process workflows are managed within TAUPOS by several software tools, detailed below, which transform the input from the SUT i.e., the AI system to be audited, into a set of potentially applicable suggestions and tricks for such systems to improve their trustworthiness, however limited to the criteria examined within the THEMIS 5.0 project, including through the use of TAIME to which are transmitted.

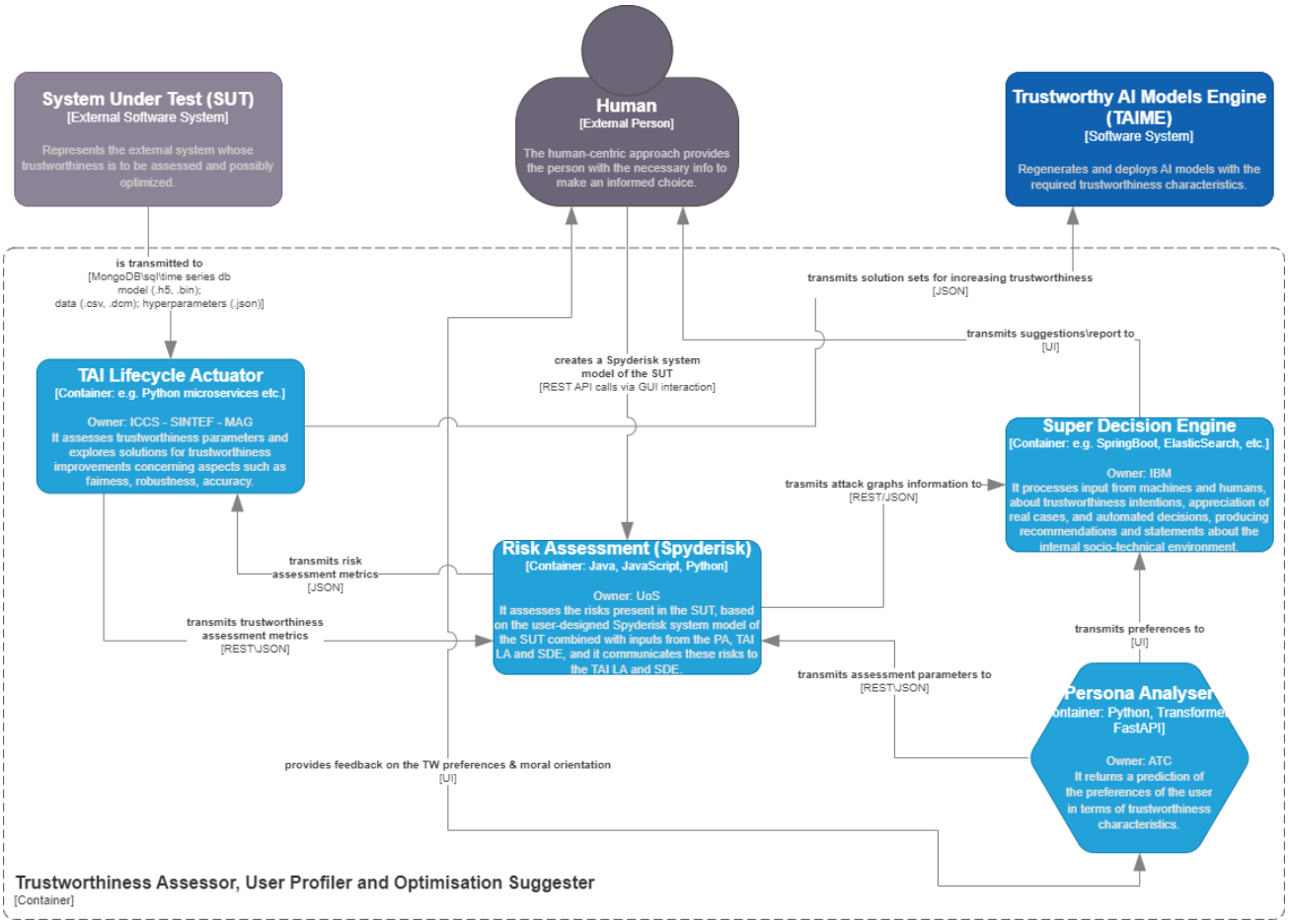

Figure 3: Container diagram of the TAUPOS

2.2.2.1.1 Building Blocks description

TAUPOS consists of four basic building blocks covering a wide range of functionality:

- Three generic-type containers:
  - The **TAI Lifecycle Actuator** deals with analysing trustworthiness parameters and exploring solutions to improve it by referring to aspects such as fairness, robustness, and accuracy;
  - The **Risk Assessment (Spyderisk)** deals with risk verification in both a broad and a narrower sense with particular reference to the trustworthiness of the analysed AI system;
  - The **Super Decision Engine** deals with processing information received from both other software and the human in order to delineate the context and the socio-technical environment of the system under analysis so as to produce appropriate recommendations;
- A microservice-type container named **Persona Analyser** that acts as a compass with respect to the trustworthiness preferences and the moral orientation of the potential user of the AI system to more finely define any suggestions to be made.

2.2.2.1.2 Interactions description

The following table report all interactions of the container TAUPOS with I/O details defined until now.

Table 2: Container – TAUPOS interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
Persona Analyser	-	Human	-	-
	-	-	-	Super Decision Engine
	-	-	REST/JSON	Persona Analyser
Risk Assessment (Spyderisk)	GUI interaction	Human	-	-
	REST/JSON	Persona Analyser	-	-
	REST/JSON	TAI Lifecycle Actuator	-	-
	-	-	REST/JSON	Super Decision Engine
	-	-	JSON	TAI Lifecycle Actuator
Super Decision Engine	-	Persona Analyser	-	-
	JSON	Risk Assessment (Spyderisk)	-	-
	-	-	-	Human
TAI Lifecycle Actuator	JSON	Risk Assessment (Spyderisk)	-	-
	-	SUT	-	-
	-	-	Risk Assessment (Spyderisk)	REST\JSON
	-	-	TAIME	JSON

2.2.2.2 Container TAIME

The TAIME lifecycle consists in an orchestrator pipeline managing the co-creation and benchmarking of datasets essential for THEMIS 5.0 model training and retraining. In this role, it integrates multiple modules, e.g. data collection, preprocessing, co-creation interfaces, and databases for storing curated datasets and hyperparameters. Within this workflow, TAIME dynamically refines datasets in continuous test loops — drawing on user feedback during both co-creation and broader deployment — to ensure that real-world nuances, fairness, and robustness are thoroughly captured.

Built using containerized, modular components it adheres to best practices in scalability and interoperability. By managing user insights, conducting iterative validations, storing intermediate results, and interfacing with external training or analytics backends, it ensures that updated benchmarks remain aligned with the evolving needs of end-users and the socio-technical contexts in which the System Under Test is deployed, reinforcing explainability, and trustworthiness for the AI models involved.

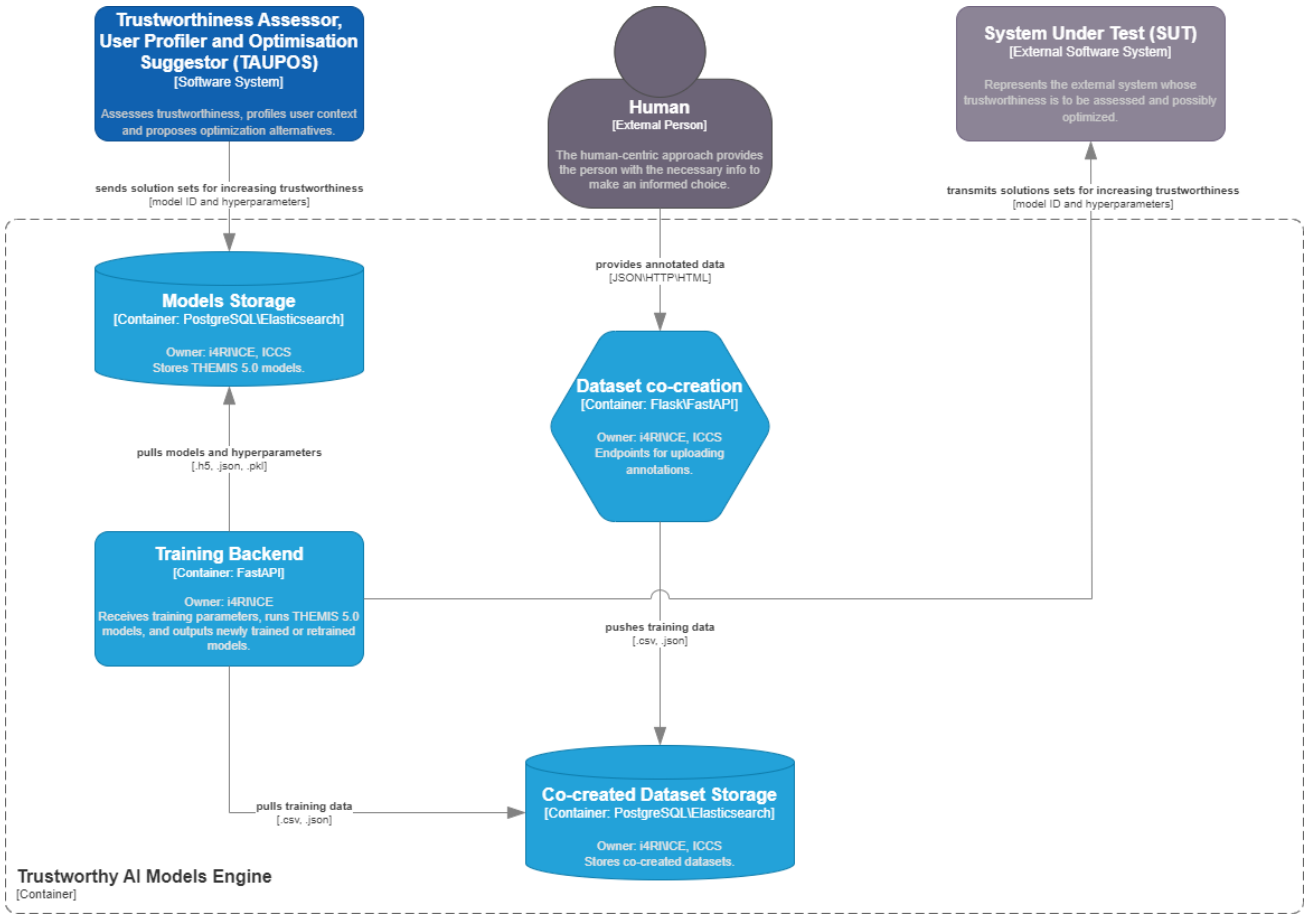

Figure 4: Container diagram of the TAIME

2.2.2.2.1 Building Blocks description

The TAIME container is composed of:

- A microservice-type container named **Dataset Co-creation**, that offers endpoints to upload or gather annotated data for training. It receives user-provided data, e.g. annotations, in JSON or HTTP format and outputs a resulting dataset, e.g. CSV or JSON, to the Co-created Dataset Storage.
- A generic-type container named **Training Backend**, that receives training parameters, runs THEMIS 5.0 models, and outputs newly trained or retrained models. It consumes datasets (CSV, JSON) and hyperparameters for training, then produces trained model artifacts.
- A storage-type container named **Co-created Dataset Storage**, that stores co-created datasets or other relevant data. It receives the final training data from the Dataset Co-creation microservice and may provide these datasets or artifacts to the Training Backend as needed.
- A storage-type container named **Models Storage**, that stores THEMIS 5.0 models that can be retrained for optimising trustworthiness.

2.2.2.2.2 Interactions description

The following table report all interactions of the container TAIME with I/O details defined until now. Since this container manages the last step of the TOP, some interactions could be subject to further refinements.

Table 3: Container – TAIME interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component

Co-created Dataset Storage	CSV/JSON	Dataset Co-creation	-	-
	-	-	CSV/JSON	Training Backend
Dataset Co-creation	JSON/HTTP	Human	-	-
	-	-	CSV/JSON	Co-created Dataset Storage
Models Storage	Model artifacts (h5, bin), data (pKL), hyperparameters (JSON)	TAUPOS	-	-
	-	-	Model artifacts (h5, bin), data (pKL), hyperparameters (JSON)	Training Backend
Training Backend	CSV/JSON	Co-created Dataset Storage	-	-
	Model artifacts (h5, bin), data (pKL), hyperparameters (JSON)	Models Storage	-	-
	-	-	Model artifacts (h5, bin), data (pKL), hyperparameters (JSON)	SUT

2.2.2.3 Containers SUT

Within the THEMIS 5.0 platform architecture, the acronym SUT i.e. System Under Test refers to the AI systems that undergo trustworthiness evaluation. Specifically, on the one hand, there are 3 iterations of SUT diagrams illustrating the systems related to the project use cases (related to the domains of health, critical infrastructure, and information media), while on the other hand, the SUT Common diagram acts as a tool for design from the ground up in addition to managing cross-cutting aspects such as access, authorization, and storage. Further details are provided in the following sections.

2.2.2.3.1 SUT: healthcare domain

The figure below depicts the high-level architecture of the pancreatic cancer (PC) predictor training pipeline. Specifically, it depicts the building blocks of the training pipeline as logical components. The pancreatic cancer predictor classifies individuals according to high, moderate or low risk of developing pancreatic cancer. The PC predictor lacks a UI layer, so the models are communicated to the TAUPOS components in a manual way. The PC predictor is trained on retrospective clinical data, containing data about gender, age, symptoms, morbidity history, comorbidities and physical findings. At the beginning, the pre-processing steps are applied (data cleaning, missing values imputation, embeddings creation). Next the training process taking place along with the hyper-parameter tuning. A SHAP (SHapley Additive exPlanations) analysis follows, where the feature impact on the predicted PC risk is computed. At the final stage a zip containing the trained model, the training/test datasets used along with a metadata file containing model information (feature data schemas, model type information etc) is extracted.

mora

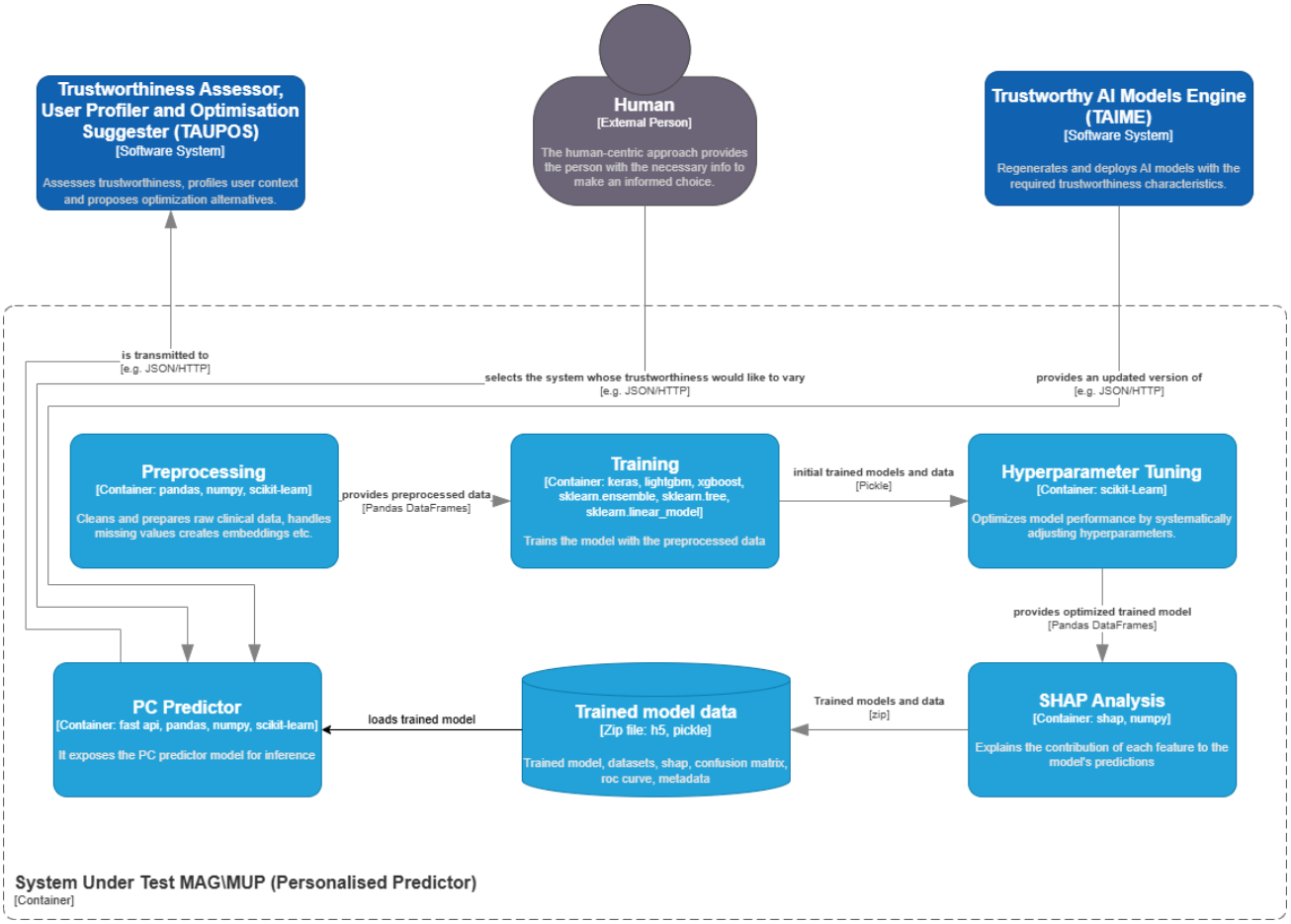

Figure 5: Container diagram of the SUT in the healthcare domain

2.2.2.3.1.1 Building Blocks description

The PC personalised predictor SUT is the Gradient Boosting classifier that results as the optimal model from the described training pipeline. It consists of the following building blocks:

Pre-processing: A request must go through all steps (data cleaning, missing values imputation, embeddings creation) to be processed by the PC predictor.

PC predictor: the predicted PC risk level is returned (low/moderate/high)

2.2.2.3.1.2 Interactions description

The PC Predictor SUT interacts with the TAUPOS component by providing the trained model for assessment. Moreover, it interacts with the TAIME component by means of retrieving an updated version of the model. It is also available to the Human actor to be selected for further analysis. The only SUT component that interacts with the THEMIS components is the PC Predictor, the rest of the components described in Figure 5 are in essence logical subcomponents of the training pipeline. Specifically, the Preprocessing module prepares the training data and feeds them into the Training module. The Training module provides the trained model along with the hyperparameters selected for optimization and feeds the Hyper Parameter Tuning module. Then the Hyper Parameter Tuning module provides the tuned model to the SHAP Analysis module to interpret the features importance. At the end, the SHAP Analysis module stores the tuned trained model in the repository. The PC Predictor is an Open API specification compatible component that exposes the inference endpoint of the trained model.

Table 4: Container – Healthcare domain SUT interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
PC Predictor	.pickle	TAIME	-	-
PC Predictor	-	-	.pickle	TAUPOS
PC Predictor	JSON	Human	-	-
Preprocessing			DataFrame	Training
Training			.Pickle, DataFrame	Hyper Parameter Tuning
Hyper Parameter Tuning			.Pickle, DataFrame	SHAP Analysis
SHAP Analysis			.pickle	Models DB
PC Predictor	.pickle/JSON	Models DB		

2.2.2.3.2 SUT: critical infrastructure domain

The figure below illustrates the high-level architecture, the components and external systems that compose the Estimated Time of Arrival (ETA) Berth prediction pipeline. The ETA Berth predictor service is trained using historical data of past port calls, weather conditions, Automatic Identification System (AIS) data (coordinates, direction and speed of vessels) and physical characteristics of vessels. The interaction with the user is done through PAULA (a web-based port CDM system property of Port of Valencia). To generate a prediction, the ETA Update Cron launches the ETA Berth prediction service, which collects the necessary data from Paula Database, Port Community System (PCS), Meteorological Service and AIS. The data is then processed and cleaned. The Dead Weight Tonnage (DWT) predictor service is used to fill a physical parameter of the vessel if its missing. When the ETA Berth is generated, it's returned to the ETA Update Cron, which writes it into PAULA Database.

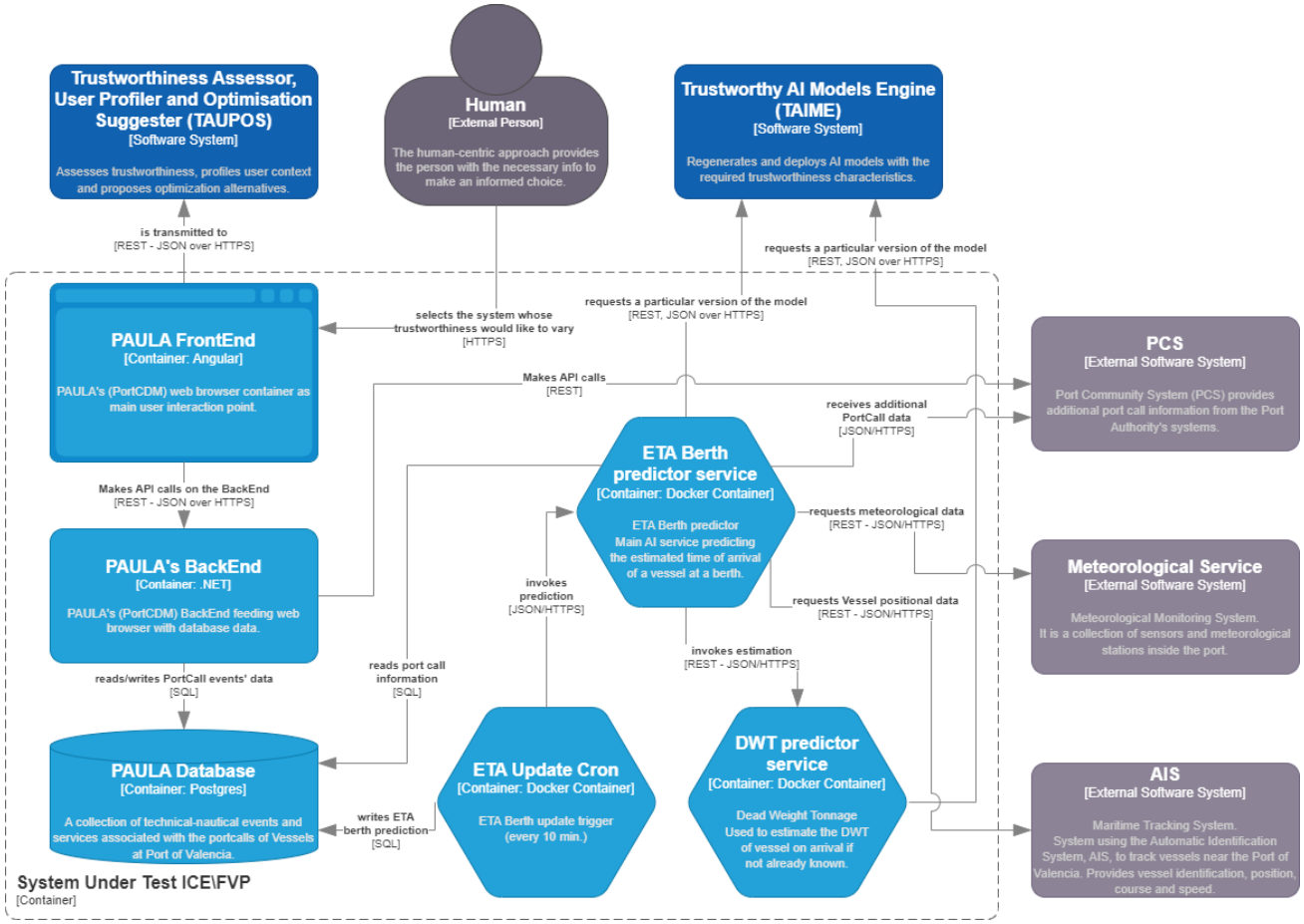

Figure 6: Container diagram of the SUT in the critical infrastructure domain

2.2.2.3.2.1 Building Blocks description

In the following the description of the system, its building blocks and functionalities.

- PAULA. Divided into three components:
  - Frontend: Web browser container made in Angular.
  - Backend: .NET container which handles the functionalities, operations, data ingestion, etc.
  - Database: A Postgres database containing all the data required for PAULA to function.
- PCS: The Port Community System of Port of Valencia. It's a platform for sharing information safely between the different agents involved in the port's operations.
- Meteorological service: An API which gives access to wind speed, wind direction and rain, collected by two weather stations at Port of Valencia.
- AIS: The Automatic Identification System is a tracking system used by vessel traffic services. We obtain the position, direction, speed and navigation status of vessels.
- DWT predictor service: an XGBoost model, trained on the physical characteristics of vessels to predict the Dead Weight Tonnage of a vessel. We use it in the data preprocessing phase to fill the DWT value if its missing in the PCS.
- ETA Update Cron: a Cron service that launches the ETA predictor service with a 10-minute frequency.
- ETA predictor service: The ETA predictor service gathers the necessary data, preprocess it, and obtains a prediction from the model.

2.2.2.3.2.2 Interactions description

The following table describes the interactions of the system with all I/O details.

Table 5: Container – Critical infrastructure domain SUT interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
ETA Update Cron	JSON/HTTPS	ETA Berth Prediction Service	SQL	PAULA Database
ETA Berth Prediction Service	SQL	PAULA Database	-	-
ETA Berth Prediction Service	JSON/HTTPS	PCS	-	-
ETA Berth Prediction Service	REST – JSON/HTTPS	Meteorological Service	-	-
ETA Berth Prediction Service	REST – JSON/HTTPS	AIS	-	-
ETA Berth Prediction Service	REST – JSON/HTTPS	DWT predictor service	-	-
ETA Berth Prediction Service	.ckpt	TAIME	-	-
DWT predictor service	.pickle	TAIME	-	-
ETA Berth Prediction Service	-	-	REST – JSON/HTTPS	TAUPOS

2.2.2.3.3 SUT: disinformation in cyberspace domain

This SUT aims to assist journalists in fighting disinformation spread across the web. It comprises of AI models that applies state of the art text analysis on news content by identifying disinformation signals. Specifically, a general-purpose language representation model, called DistilBERT, is fine-tuned on the tasks of hate speech and the fake news detection. An orchestration component, the AI Toolkit, handles the asynchronous execution of the AI models and exposes the results to the UI component, namely the Companion web application, using web sockets.

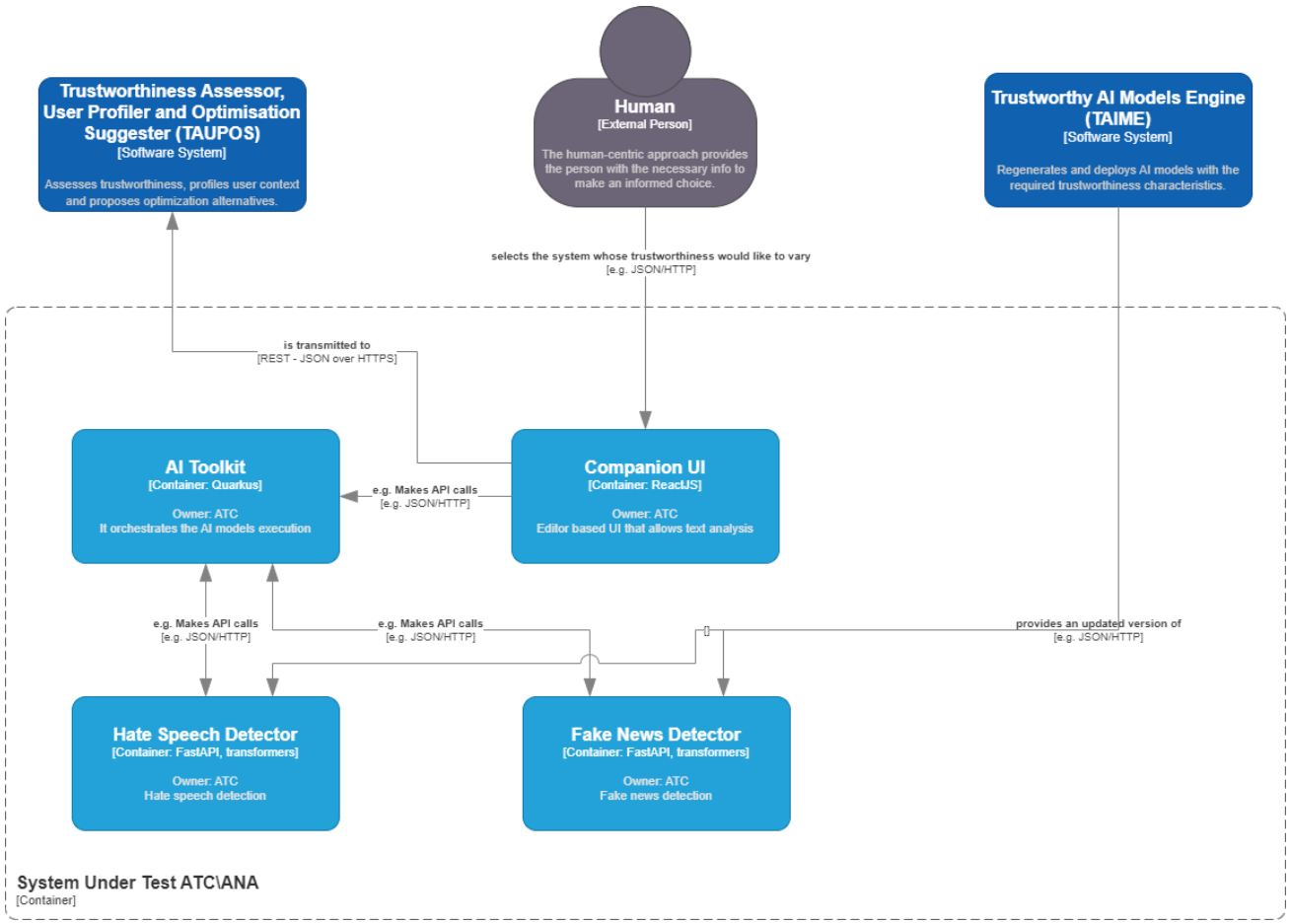

Figure 7: Container diagram of the SUT in the disinformation in cyberspace domain

2.2.2.3.3.1 Building Blocks description

The disinformation SUT consists of the following building blocks:

- A microservice-type container, named **Hate Speech Detector**, that detects hate speech signals.
- A microservice-type container, named **Fake News Detector**, that assesses whether a news content is fake or not.
- A microservice-type container, named **AI Toolkit**, that handles the asynchronous execution of the AI models and exposes the results through a web socket channel.
- A web-type container, named **Companion UI**, visualises the analysis results to the end user.

2.2.2.3.3.2 Interactions description

The table that follows depicts the interactions of the SUT system components including all I/O details. The serialization format of the fine-tuned models is .pth extension which typically contains a serialized PyTorch state dictionary. A PyTorch state dictionary is a Python dictionary that contains the state of a PyTorch model, including the model's weights, biases, and other parameters.

Table 6: Container – Disinformation in cyberspace domain SUT interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
Companion UI	JSON/HTTP	Human	-	-

Hate Speech Detector	Model serialization format: .pth	TAIME	-	-
Fake News Detector	Model serialization format: .pth	TAIME	-	-
Hate Speech Detector	-	-	JSON/HTTP	AI Toolkit
Fake News Detector	-	-	JSON/HTTP	AI Toolkit

2.2.2.3.4 SUT Common: design from scratch

The SUT Common, unlike the other SUTs, does not refer to a specific project use case, but rather provides the possibility for developers/users to design and execute a new pipeline training model from scratch to produce an AI trained model. This option makes it possible to test the platform without the need to have a specific model in advance or, from another point of view, allows you to try out models that you have within your own infrastructure but to which access is limited. Furthermore, in some respects, the SUT Common acts as a star centre for aspects such as access and authorisation, storage and serving of models.

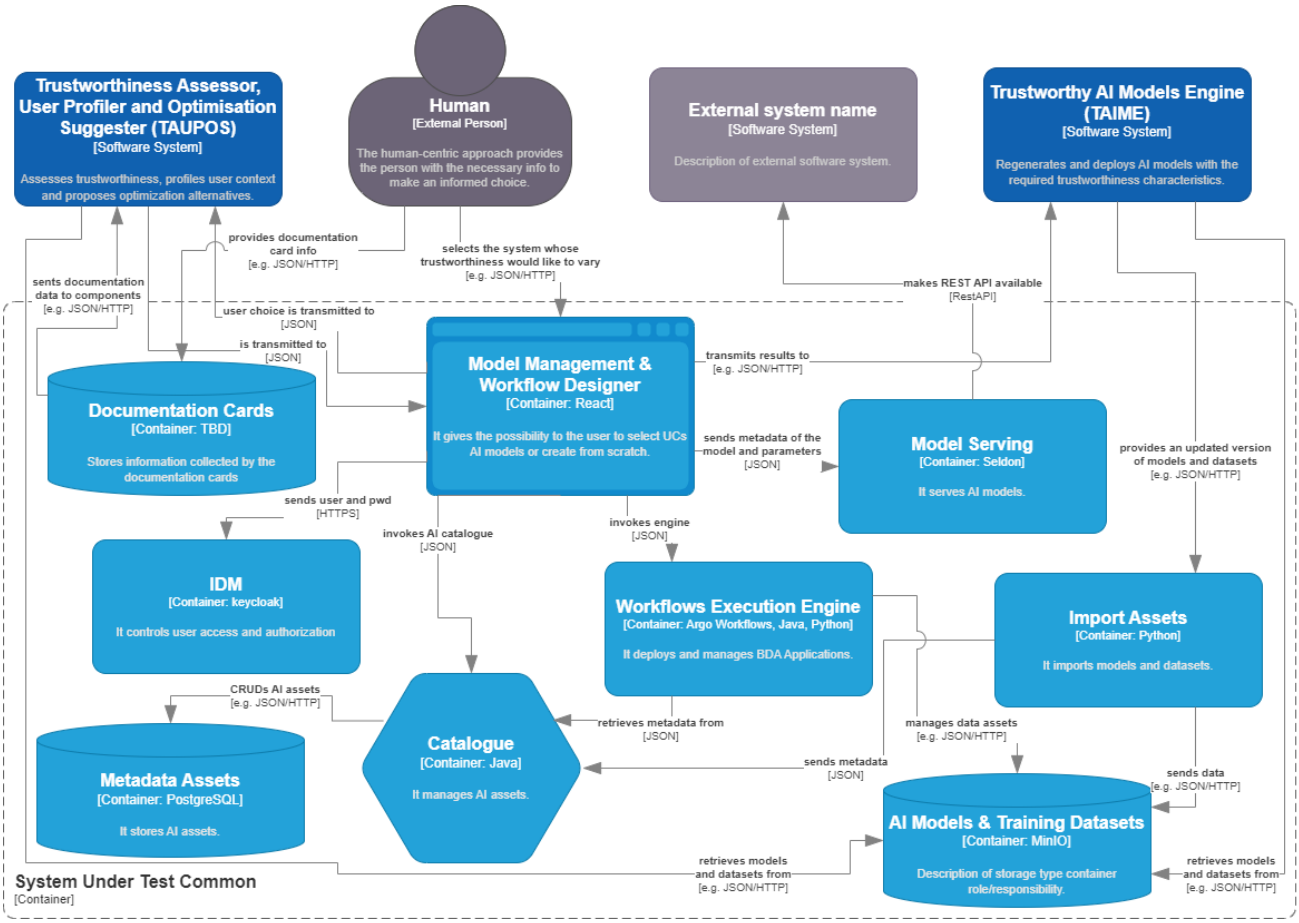

Figure 8: Container diagram of the SUT Common

2.2.2.3.4.1 Building Blocks description

The SUT Common consists of the following building blocks:

- A web browser-type container, named **Model Management & Workflow Designer**, which allows the selection of a registered trained model to be served within the platform or the design, execution and monitoring of a new training pipeline model to produce an AI model within the platform.

- A microservice-type container, named **AI Catalogue**, which manages AI-related assets, i.e. algorithms, models, datasets and metadata.
- Three storage-type containers:
    - The **Metadata Assets**, which takes care of storing the various assets available for design from scratch.
    - The **AI Models & Training Datasets**, which takes care of storing the different assets available from the design use cases.
    - The **Documentation cards**, which stores the information collected through the Identify stage of TOP in a structured format.
- Four generic-type containers, needed for:
    - Manage user access and authorisation, i.e., the building block named **IDM**.
    - Manage and provide applications for analysing big data related to models built from scratch, i.e., the **Workflows Execution Engine** building block.
    - Serve AI models externally, i.e., the **Model Serving** building block.
    - Importing updated models and datasets through trustworthiness assessment, i.e., the **Import Assets** building block.

2.2.2.3.4.2 Interactions description

The following table specifies the I/O interactions of the SUT Common, both between its constituent building blocks and with external building blocks. In particular, the table specifies the I/O formats and the relative actors with which the exchange of information in the defined format takes place, thus enabling a cross-check with what is also documented in the diagram of the previous figure in which the I/O actions carried out are made explicit.

Table 7: Container – SUT Common interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
AI Models & Training Datasets	JSON/HTTP	Import Assets	-	-
	JSON/HTTP	TAIME	-	-
	JSON/HTTP	TAUPOS	-	-
	JSON/HTTP	Workflow Execution Engine	-	-
Documentation Cards	JSON/HTTP	Human	-	-
	-	-	JSON/HTTP	TAUPOS
Catalogue	JSON	Import Assets	-	-
	-	-	JSON/HTTP	Metadata Assets
	JSON	Model Management & Workflow Designer	-	-
	JSON	Workflow Execution Engine	-	-
IDM	HTTP	Model Management & Workflow Designer	-	-

Import Assets	-	-	JSON/HTTP	AI Models & Training Datasets
	-	-	JSON	Catalogue
	JSON/HTTP	TAIME	-	-
Metadata Assets	JSON/HTTP	Catalogue	-	-
Model Management & Workflow Designer	-	-	JSON	Catalogue
	JSON/HTTP	Human	-	-
	-	-	HTTPS	IDM
	-	-	JSON	Model Serving
	-	-	JSON/HTTP	TAIME
	JSON	TAUPOS	-	-
	-	-	JSON	TAUPOS
	-	-	JSON	Workflow Execution Engine
Model Serving	-	-	RestAPI	External System
	JSON	Model Management & Workflow Designer	-	-
Workflow Execution Engine	-	-	JSON/HTTPS	AI Models & Training Datasets
	-	-	JSON	Catalogue
	JSON	Model Management & Workflow Designer	-	-

2.2.3. Component level

The Component level diagrams are the third-level diagrams following the C4 model. Within the THEMIS 5.0 project, this level is employed where software tools operated by different partners are present in the second-level diagrams: thus, the operation of these tools is clarified through the additional details provided below.

2.2.3.1 Components TAUPOS

This section clarifies how the TAUPOS container is built i.e. this component level is used to detail the TAUPOS container: it is the result of the joint effort of several partners who adopted owned solutions to tackle implementation challenges and, as said, needing more in-depth view to coordinate all the efforts and platform functionalities.

2.2.3.1.1 TAUPOS: Persona Analyser

Persona Analyser is a Python-based application that performs trustworthiness and moral orientation prediction of the end user. It is implemented as a chatbot backed by the Persona Predictor component. Persona Predictor constantly updates the current trustworthiness and moral orientation level of the user and guides the dialogue by proposing the next question. At the end of the interaction with the end user, it communicates the assessment results (moral orientation class, list of TW characteristics preference) to the Risk Assessment component.

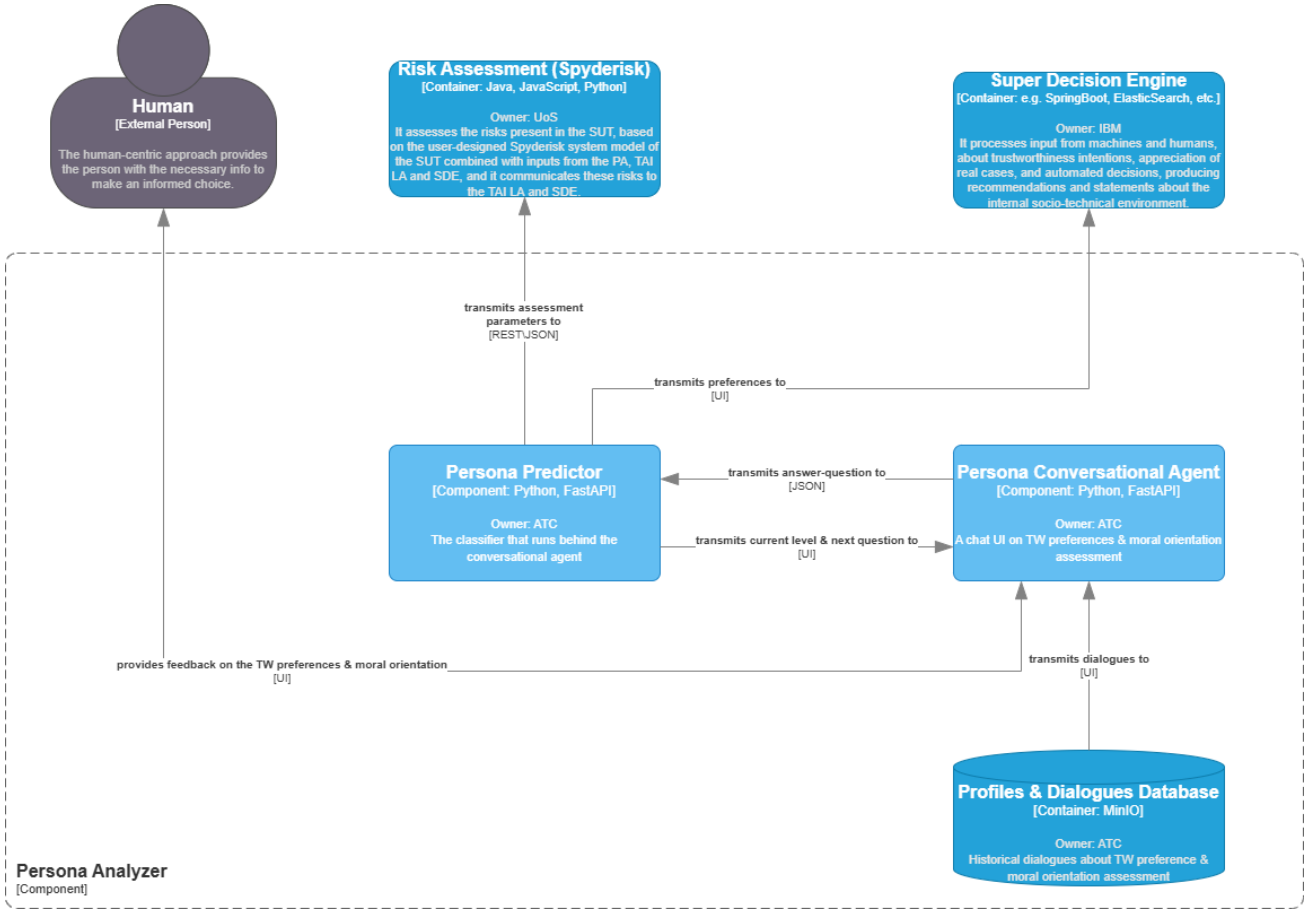

Figure 9: Component diagram of the TAUPOS Persona Analyser

2.2.3.1.1.1 Building Blocks description

Persona Analyser consists of the following building blocks:

- Persona Predictor: it assesses the current moral orientation level, the list of TW characteristic preferences and guides the dialogue by proposing the next question.
- Persona conversational agent: a chatbot application that performs the TW preferences and moral orientation assessment.
- Profiles & Dialogues DB: holds the co-created dataset that is used to train the classifier.

2.2.3.1.1.2 Interactions description

In the following table the interactions of the system with all I/O details.

Table 8: Component – TAUPOS Persona Analyser interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
Persona Predictor	JSON	Persona CA	-	-
Persona Predictor	-	-	REST/JSON	Super Decision Engine
Persona Predictor	-	-	REST/JSON	Risk Assessment
Persona CA	JSON	End user	-	-

Persona Predictor	JSON	Profiles & Dialogues DB	-	-

2.2.3.1.2 TAUPOS: TAI Lifecycle Actuator

The TAI Lifecycle Actuator is a Python-based orchestrator that manages the i) technical assessment and ii) solution exploration of the trustworthiness lifecycle for the Systems Under Test (SUT). It integrates multiple modules (assessment, exploration, decision-making), external databases (MongoDB, Neo4j), and external services (e.g., Risk Assessment).

The Lifecycle Actuator is built using FastAPI and is designed to support modularity, scalability, and containerized deployment with Docker. It manages user inputs, computes trustworthiness metrics, stores results, and interfaces with external systems for further risk assessments or recommendations.

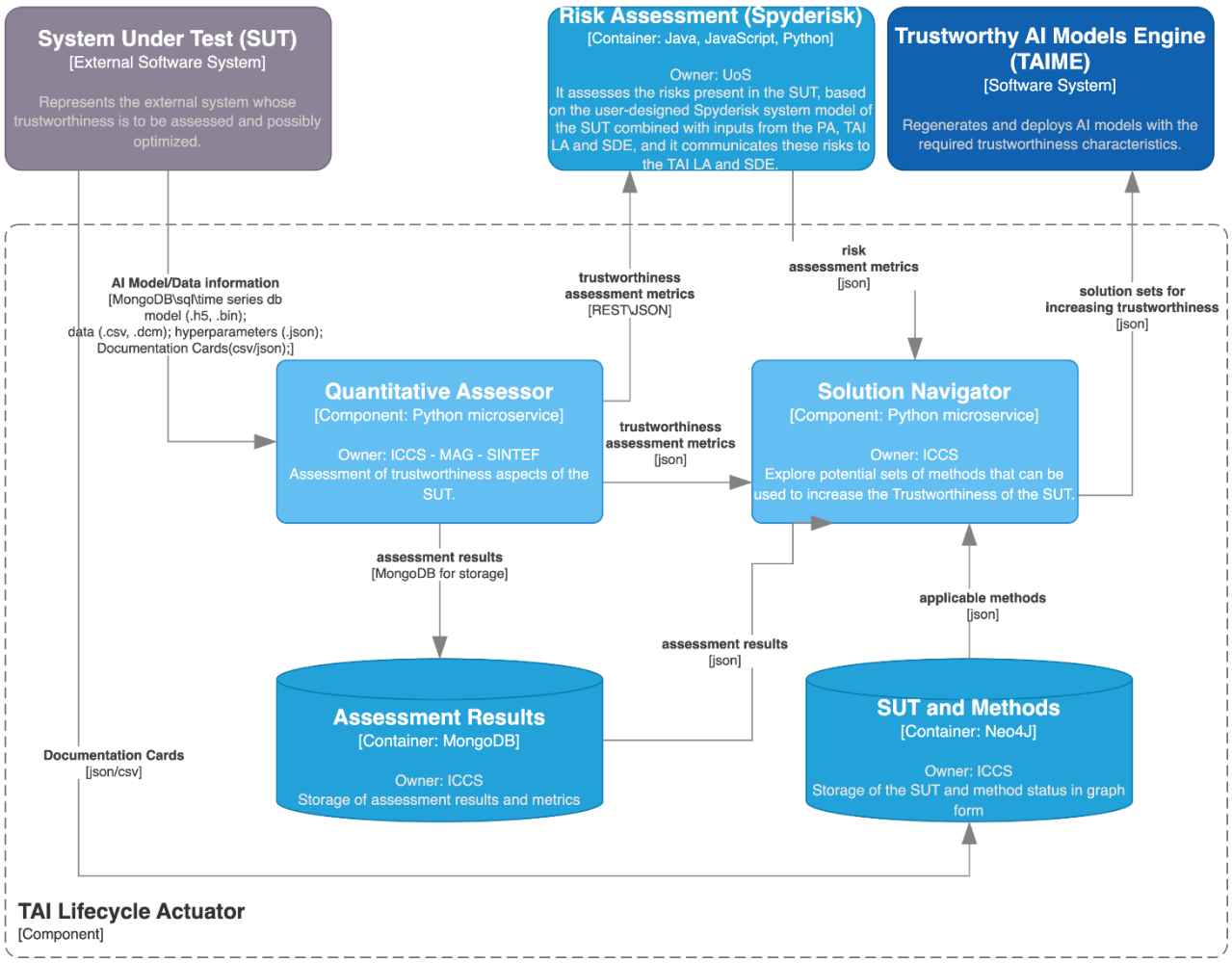

Figure 10: Component diagram of the TAUPOS TAI Lifecycle Actuator

2.2.3.1.2.1 Building Blocks description

The TAI Lifecycle Actuator consists of the following building blocks:

- **Quantitative Assessor** (see Section 2.2.3.1.2.3)
- **Solution Navigator** (see Section 2.2.3.1.2.4)
- **Assessment Results (MongoDB):** Provides persistent storage for requested assessments of a SUT. Instances of the database are accessed from both Quantitative Assessor and Solution Navigator and are accessible to the user via API calls to the respective components.

- **SUT and Methods (Neo4j):** Provides persistent storage for the details of the SUT, represented in a knowledge graph. Used internally by the Quantitative Assessor and Solution Navigator to retrieve SUT details and explore usable enhancement methods and relevant performance metrics.

2.2.3.1.2.2 Interactions description

In the following table the interactions of the system with all I/O details.

Table 9: Component – TAUPOS TAI Lifecycle Actuator interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
Quantitative Assessor	API Calls (POST, GET)	SUT	JSON/CSV	MongoDB, Risk Assessment
Solution Navigator	API Calls (POST, GET)	Quantitative Assessor, Risk Assessment	JSON/CSV	TAIME
Assessment Results (MongoDB)	Python Connector	Quantitative Assessor, Solution Navigator	{metrics, metadata, id}	Persistent Storage
SUT and Methods (Neo4j)	Python Connector	Quantitative Assessor, Solution Navigator	{SUT details, metrics}	Persistent Storage

2.2.3.1.2.3 TAI Lifecycle Actuator: Quantitative Assessor

The Quantitative Assessor is a core subcomponent of the TAI Lifecycle Actuator, responsible for evaluating trustworthiness characteristics such as fairness, robustness, and accuracy for datasets or models provided by the SUT. It operates through the **/assess** endpoint and relies on MongoDB for persistence and storage of results. Incoming requests specify which algorithms (from the Accuracy, Robustness, or Fairness pools) should be run, along with any required parameters (e.g., AI model files, data, and configuration variables). The response then includes the computed metrics.

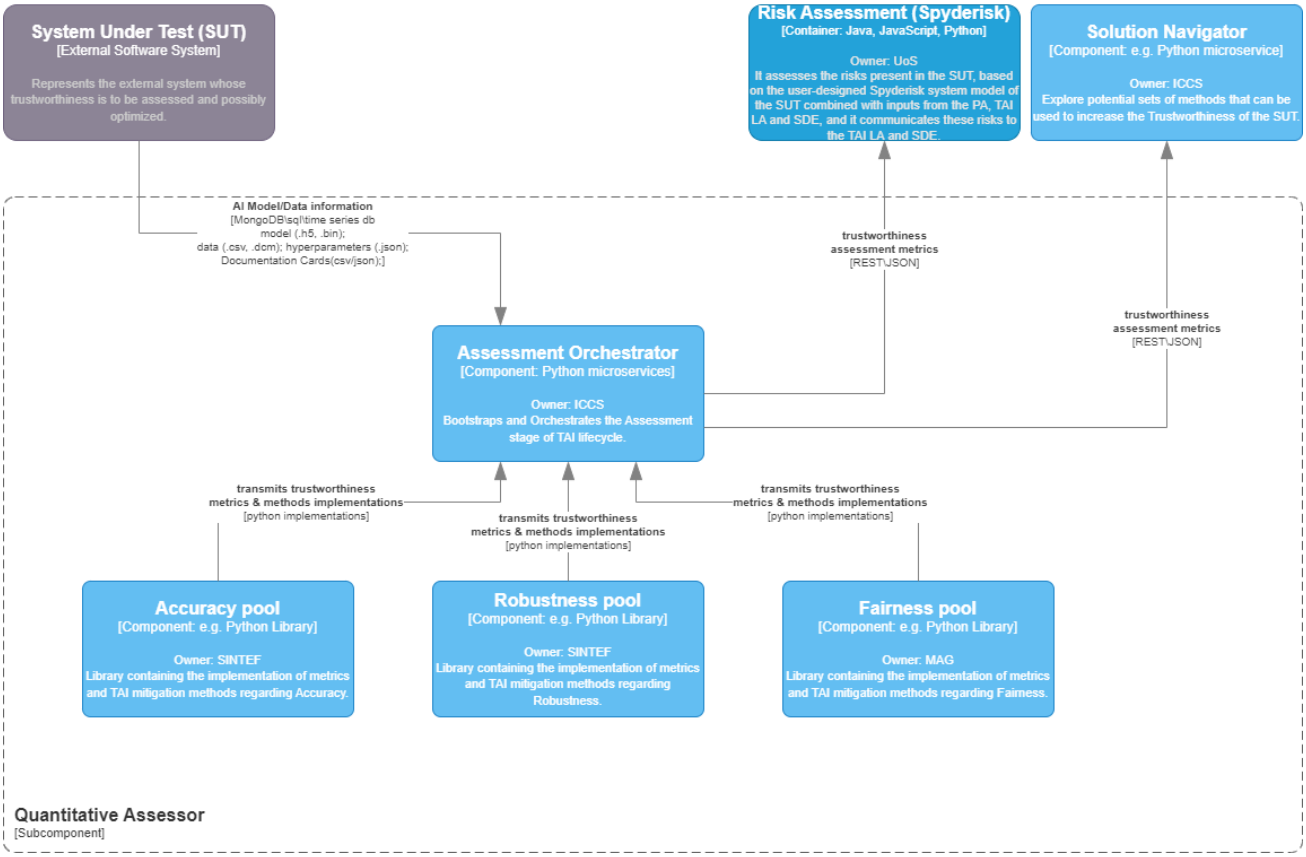

Figure 11: Subcomponent diagram of the TAI Lifecycle Actuator Quantitative Assessor

2.2.3.1.2.3.1 Building Blocks description

The Quantitative Assessor consists of the following key building blocks:

- **Assessment Orchestrator**
  - FastAPI Endpoint

    The **/assess/evaluate_metrics/** endpoint provides a RESTful interface for users to interact with the Quantitative Assessor. It serves as the primary entry point for submitting data, retrieving metrics, and downloading results. The endpoint supports the following functionalities:
    - Submitting SUTs for Assessment: Users can provide the SUT in the form of datasets or models, along with hyperparameters and the trustworthiness characteristic to evaluate (e.g., robustness, fairness, accuracy). The endpoint calculates metrics and stores the results in a MongoDB database for persistence.
    - Retrieving Previous Results: Users can query previously computed metrics by specifying a result ID or filtering by trustworthiness characteristics.
    - Downloading Results: The endpoint allows users to export filtered results in CSV format for offline analysis.
  - MongoDB Database

    The MongoDB database acts as the persistent storage layer for the Quantitative Assessor (see Figure 10), ensuring that all computed metrics and assessment results are saved for future retrieval. The database is accessed programmatically by the /assess endpoint to store, retrieve, and filter results. Each assessment result is stored as a document in the assess collection, with fields for the SUT data, hyperparameters, calculated metrics, and metadata.
- **Accuracy, Robustness and Fairness pools**

  These pools are implemented as Python libraries that provides methods for assessing accuracy, robustness and accuracy. Accuracy is implemented by the SafetyCage Python library (see Section

3.1.2.2), robustness is implemented by the OODGuard Python library (see Section 3.1.2.3), and fairness is implemented by the AIF360 Python library (see Section 3.1.2.1).

2.2.3.1.2.3.2 Interactions description

In the following table the interactions of the system with all I/O details.

Table 10: Subcomponent – TAI Lifecycle Actuator Quantitative Assessor interactions

Endpoints	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
/assess/evaluate_metrics	JSON (*dataset, model, category*)	User (via API)	JSON (trustworthiness_metrics, timestamp, id)	MongoDB/ User (via API)
MongoDB	Trustworthiness Metrics and Metadata	/assess	JSON/CSV (trustworthiness metrics, metadata, _id)	Persistent Storage/ User (via API)
/assess/ evaluate_metrics /results	Query Parameters (*category, result_id*)	MongoDB	JSON (list of entries)	User (via API)
/assess/ evaluate_metrics /result/download	Query Parameters (*category, result_id*)	MongoDB	CSV file	User (via API)

2.2.3.1.2.4 TAI Lifecycle Actuator: Solution Navigator

The Solution Navigator is a subcomponent of the TAI Lifecycle Actuator, responsible for exploring and proposing appropriate TAI mitigation methods for a SUT. It utilizes the results of the Quantitative Assessor and the Risk Assessment Components, to select methods expected to increase trustworthiness, or other metrics, of the system. It operates through the **/explore** endpoint and relies on MongoDB for persistent storage as well as Neo4j for SUT data storage and method exploration.

2.2.3.1.2.4.1 Building Blocks description

The Solution Navigator consists of the following key building blocks:

- **FastAPI Endpoint**
  The **/explore/** endpoint provides a RESTful interface for users to interact with the Solution Navigator. It serves as the primary entry point for submitting data, retrieving metrics, and downloading results. The endpoint supports the following functionalities:
  - Exploring applicable methods: The endpoint finds applicable methods that increase the trustworthiness of datasets and AI models based on the status of the SUT and the documented methods. The returned applicable methods can serve as controls for the Risk Management framework.
  - Submitting SUTs and methods for Assessment (If applicable): The endpoint (currently) calculates metrics relevant to the trustworthiness characteristic and the mitigation method selected by the user.
  - Retrieving Previous Results: Users can query previously computed metrics by specifying a result ID or filtering by trustworthiness characteristics.
  - Downloading Results: The endpoint allows users to export filtered results in CSV format for offline analysis.

- **MongoDB Database**sThe MongoDB database acts as the persistent storage layer for the Solution Navigator, ensuring that all computed metrics and assessment results are saved for future retrieval. The database is accessed programmatically by the /explore endpoint to store, retrieve, and filter results.s**Neo4j Knowledge Graph**

  The Knowledge Graph stores information about the SUT, derived from the relevant Use Case, Data and Model Cards, relating them to the relevant mitigation methods and trustworthiness characteristics from the Method cards.

2.2.3.1.2.4.2 Interactions description

In the following table the interactions of the system with all I/O details.

Table 11: Subcomponent – TAI Lifecycle Actuator Solution Navigator interactions

Endpoints	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
/explore/	JSON (characteristic, algorithm, dynamic_input)	User (via API)	JSON (exploration_result, timestamp, id)	MongoDB/ User (via API)
MongoDB	Trustworthiness Metrics for Method and Metadata	/explore	JSON/CSV (*characteristic, algorithm,* trustworthiness metrics, metadata, _id)	Persistent Storage/ User (via API)
/explore/results	Query Parameters (*characteristics, result_id*)	MongoDB	JSON (list of entries)	User (via API)
/explore/result/download	Query Parameters (*characteristics, result_id*)	MongoDB	CSV file	User (via API)
/explore/applicable-methods	Query Parameters (characteristics)	Neo4j	JSON	User (via API)
Neo4j	SUT data, Applicable Methods	/explore	JSON	Persistent Storage/ User (via API)

2.2.3.1.3 TAUPOS: Risk Assessment (Spyderisk)

The risk assessment of the TAUPOS is performed using Spyderisk. Spyderisk is an automated risk assessment tool that follows an ISO 27005 methodology. A user of the Spyderisk tool designs and creates a Spyderisk system model within it, runs the risk assessment, and then explores the risks present, along with controls to mitigate or blocks risks identified to bring their level of risk down to an acceptable level.

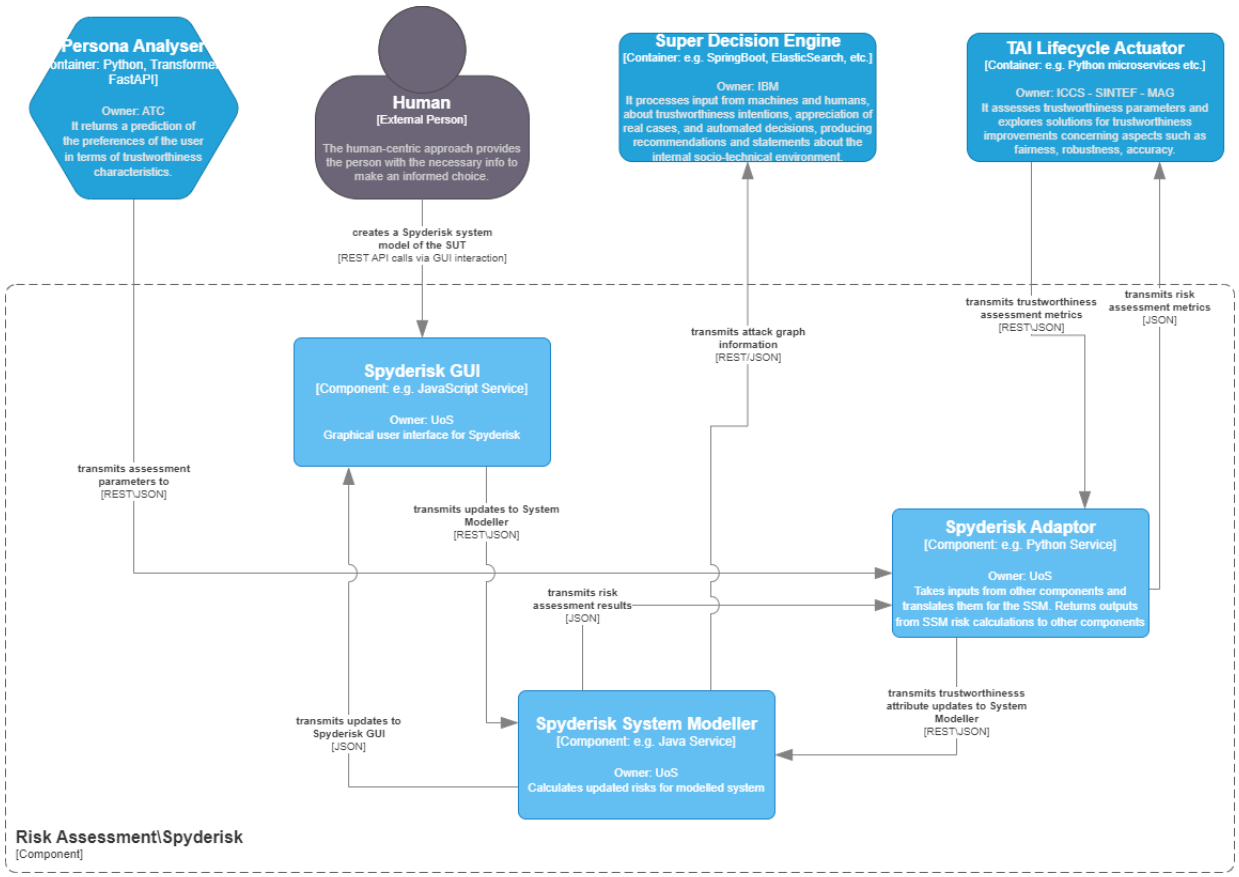

Figure 12: Component diagram of the TAUPOS Risk Assessment\Spyderisk

2.2.3.1.3.1 Building Blocks description

Spyderisk is comprised of several components:

- **The Spyderisk System Modeller**
  The Spyderisk System Modeller (SSM) is written in Java, performs the actual system model creation, validation, and risk calculation, along with the I/O in saving and retrieving system models. The SSM also contains the Spyderisk knowledge base that contains information on the different domain assets, asset relationships, trustworthiness attributes, threats, consequences, and controls. The SSM has a REST API, through which its actions are performed, and within the backend is also a MongoDB where user accounts, system models, and the knowledge base persist.
- **The Spyderisk graphical user interface**
  The Spyderisk graphical user interface (GUI) is written in JavaScript and allows the user of Spyderisk to interactively create system models, validate them, run risk calculations on them and explore the risks within them.
- **The Spyderisk Adaptor**
  The Spyderisk Adaptor is written in Python and provides additional functionality to Spyderisk that is tailored to specific needs of a domain or project. The Adaptor has a REST API to call this tailored functionality and communicates with the SSM using the SSM REST API.

2.2.3.1.3.2 Interactions description

In the following table the interactions of the system with all I/O details.

Table 12: Component – TAUPOS Risk Assessment (Spyderisk) interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
Spyderisk Adaptor	REST/JSON	Persona Analyser		
	REST/JSON	Lifecycle Actuator	JSON	Lifecycle Actuator
	JSON	Spyderisk System Modeller	REST/JSON	Spyderisk System Modeller
Spyderisk GUI	REST API Calls via GUI	User		
Spyderisk System Modeller	REST/JSON	Spyderisk GUI	REST/JSON	Spyderisk GUI
			REST/JSON	Super Decision Engine

2.2.3.1.4 TAUPOS: Super Decision Engine

Give contextual information related to the overall architecture of your software tool (use cross references to figures and tables). Short introduction of the following sections that will be used to detail the component diagram.

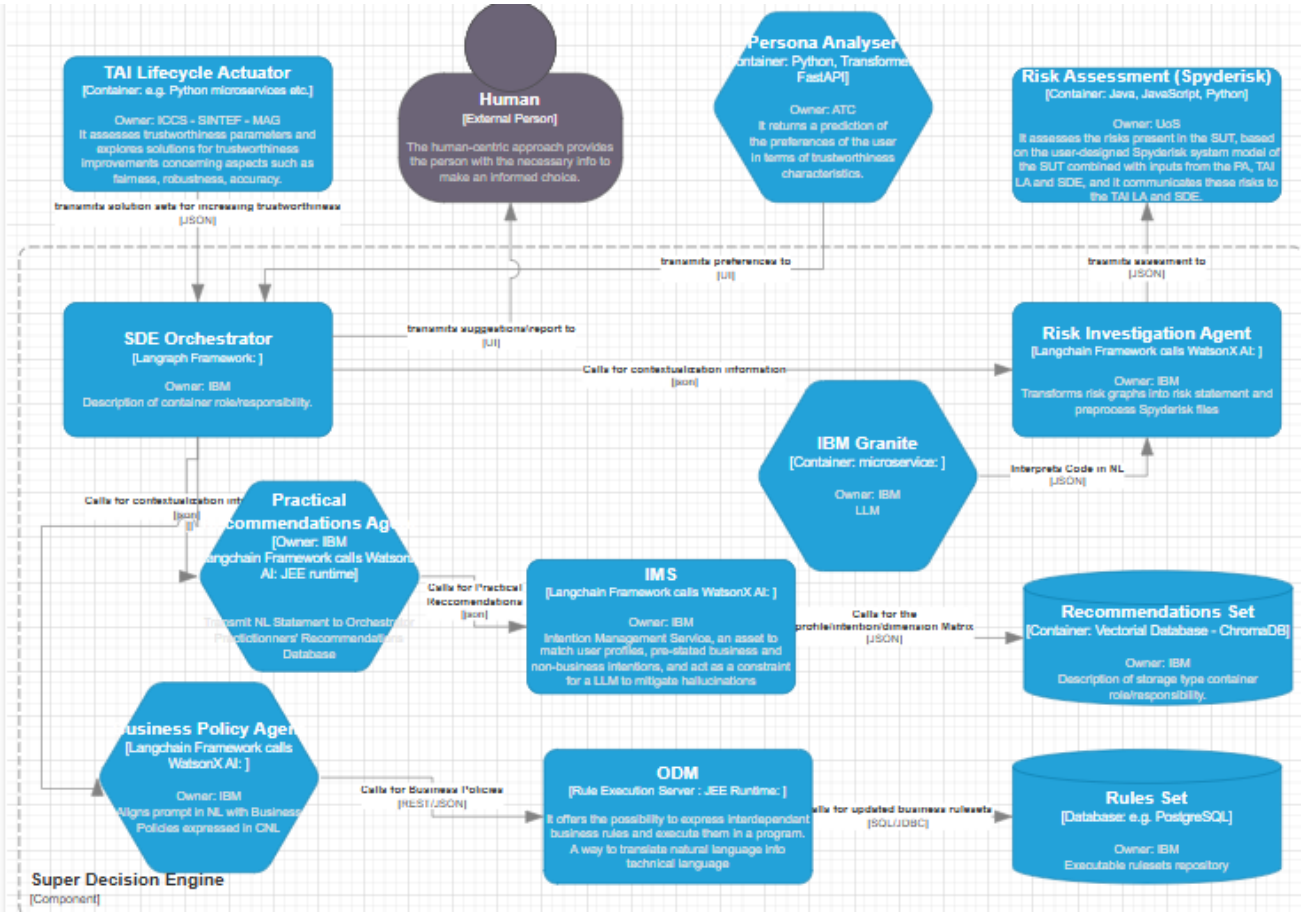

Figure 13: Component diagram of the Super Decision Engine

2.2.3.1.4.1 Building Blocks description

IBM SDE is composed of several software components

- Agentic AI Systems (AAIS).
- An orchestrator

Each AAIS consists in a chain of computation that mobilizes a library of prompts, Large Language Models (LLM) and a piece of more deterministic engine. The prompts library is tuned on a certain version of the model called, and a certain input format. There are three main AAIS corresponding to the three functional requirements of SDE:

- A Rules Agent that implements business policy.
- A Risk Agent that verifies adequacy with the Moral Compass model
- A Practical Agent (including a human-in-the-loop HITL feature) that verifies the feasibility of the recommendation

2.2.3.1.4.2 Interactions description

Input are instructions in NL since LLMs are used within agents for machine-to-machine data processing. Interoperations between different agents will be managed through the ad-hoc IBM orchestrator, which is not yet developed. CrewAIn Langgraph and IBM BeeAI agent are being tested for that purpose.

Table 13: Component – TAUPOS Super Decision Engine interactions

Building Block	Input format(s)	To be loaded from storage/to be received from component	Output format(s)	To be stored in storage/to be sent to component
IBM Orchestrator	Json and Tbd	SUT	Json	IBM SDE -
Business Policy Agent (BPA)	Json, NL Statements	IBM	NL statements	IBM SDE PRA
Practical Recommendations Agent (PRA)	NL statements	IBM	Automated SQL requests	IBM SDE RAA
Risk Assessment Agent (RAA)	NL Statement	IBM	NL statements	Orchestrator
Business Ruleset	PostgreSQL/JDBC	IBM	OWL	IBM WatsonX
Recommendations Database	Json	IBM	csv	IBM SDE
Spyderisk API	NL Statement	UoS	Json	UoS
Trustworthiness Assessment Connector	NL Statement	Tbd	NL Statement	Json

2.3. Ensuring compliance with the GDPR and the AI Act in the design stage (KUL)

This Section builds on the objective to ensure that legal requirements are complied with since the design stage. It outlines key considerations on the application of the GDPR and the AI Act to the selection and curation of benchmark training datasets, while outlining high-level recommendations for partners. The considerations and recommendations set out in this Section represent the basis for KUL support to partners in relation to the activities carried out under WP4.

2.3.1. *Introduction and scope of analysis*

The most relevant pieces of legislation to consider for the purposes of this deliverable are the General Data Protection Regulation (GDPR) and the Artificial Intelligence Act (AI Act). They acquire relevance due to the technology-specific scope of the AI Act and to the fact that personal data is being processed to train AI models for implementing THEMIS 5.0. On the latter point, the selection and curation of benchmark training datasets, as detailed in Section 4.3, includes the processing of personal data within the meaning of the GDPR. This does not exclude that other provisions of national and EU law could apply to the activities of the project, such as intellectual property law should the curation of benchmark datasets lead to a reproduction of subject matter protected by copyright. However, the scope of the legal assessment is limited to the GDPR and the AI Act, as the pieces of legislation that acquire most relevance and that pose the highest legal risks for the project. The GDPR and the AI Act come into play for different reasons in the context of the activities carried out under WP4.

On the one hand, the AI Act does not apply to any research, testing or development activity regarding AI systems or AI models that takes place before they are placed on the market or put into service. This is the research exemption laid down in Article 2(8) of the AI Act. Moreover, the AI Act does not apply in its entirety *ratione temporis* since all its provisions only apply from 2 August 2026[1], with a few exceptions. However, partners aim, with the support of KUL, to ensure that the AI models trained during the project are future-proof and present the requisite characteristics to comply with the AI Act when it applies to their placing on the market and deployment. To this end, this section illustrates the main provisions of relevance to the activity of curating benchmark training datasets, providing high-level guidance on how compliance can be ensured.

On the other hand, the GDPR applies to any processing of personal data taking place during the curation of benchmarking training datasets. While there are some exceptions provided by the GDPR for scientific research activities, this does not lead to a full exemption from the scope of application of the GDPR (see Recital 159 of the GDPR), but rather to the loosening of some requirements (e.g. on purpose limitation, storage limitation, and transparency requirements). Therefore, it must be assessed how the GDPR applies to the curation of benchmarking training datasets.

2.3.2. *Curation of benchmark training datasets in compliance with the GDPR*

The first step in assessing compliance with the GDPR is to establish whether there is any processing of personal data. As stated in Sections 4.3 and 4.6 below, preference is given during the project to the use of anonymised datasets for AI model training. Anonymous data from which it is not possible to identify data subjects does not qualify as personal data within the meaning of Article 4(1) of the GDPR. In this regard, it must be noted that there is a common misconception that anonymous data is a binary concept, and that it is always possible to achieve full anonymisation by reducing the risks of re-identification in a dataset to zero. On the contrary, there can be different degrees of anonymisation and the risks of re-identification do not need to be zero for a dataset to be outside the scope of the GDPR. A dataset can still be considered anonymous in the presence of residual risk below a certain threshold, if it passes the 'means reasonably likely to be used' test set out in Recital 26 of the GDPR. Therefore, for anonymised datasets a full assessment is needed to ascertain that the anonymisation techniques applied do not enable identification considering the means reasonably likely to be used in the circumstances of the case.

Project partners recognise and assume that in some cases anonymisation is not possible, and that personal data would be processed. In these cases, data should be pseudonymised where possible to increase the level of protection of personal data and to facilitate compliance with the GDPR, taking into account the legislative definition of pseudonymisation in Article 4(5) of the GDPR and the recently issued guidelines of the EDPB on pseudonymisation[2]. Partners should, in any case, bear in mind that pseudonymous data is still personal data,

[1] See Article 113 of the AI Act.

[2] EDPB, Guidelines 01/2025 on Pseudonymisation, 2025.

which is fully subject to the GDPR, though the pseudonymisation of data can serve as a technical and organisational measure to comply with GDPR requirements.[3]

The GDPR imposes several requirements that should be complied with by controller and processors in their processing operations involving personal data. The curation of benchmark training datasets would qualify as a processing of personal data under the GDPR, and thus partners acting as controllers for these operations must ensure, among others, that there is a **legal basis for the processing**[4], that they comply with the **principles relating to the processing of personal data**[5], that they provide the **required information to data subjects**[6], and that they **facilitate the exercise of data subject rights**[7]. In addition, controllers shall ensure that their processing operations and related technical and organisational measures implement the data protection principles of the GDPR by design[8] and by default[9]. The THEMIS 5.0 platform architecture has as one of its goals to respect data protection by design and by default: this means that relevant principles of the GDPR have been considered as integral parts of the system from the earliest stages of design and development and not as an aspect added as a later feature, but rather a fundamental part of the architecture itself. In other words, partners are aware that they must act responsibly as of the design stage, and continue to do so during the entire data processing lifecycle.

Multiple solutions to implement the principles of the GDPR by design and by default have been discussed over time in existing literature, such as federated machine learning as an alternative that facilitates data minimisation by avoiding the centralisation of training datasets in the hands of a single entity.[10] In implementing solutions for data protection by design, controllers must take into account the state of the art, the cost of implementation, and the nature, scope, context, and purposes as well as the risks posed by the processing.

In the context of Themis, the architecture incorporates the following safeguards and measures to ensure data protection by design and by default:

- Restricted access and control: the system will take into consideration implementing control mechanisms that allow only authorized persons to access and process data, protecting them from unauthorized access;
- Data minimization: the system will take into consideration collecting only the data strictly necessary for the purposes for which it is designed, avoiding excessive collection of information;
- Pseudonymization: where possible and anonymisation cannot be achieved, personal data is pseudonymised to reduce the risk that data subjects are identified;
- Accountability and monitoring: the system will take into consideration implementing mechanisms to monitor and document data processing so that practices are auditable and compliant with the GDPR;
- Security by default: the system will take into consideration ensuring that data is handled securely, with appropriate encryption measures, protection against vulnerabilities, and secure information management;
- Transparency: the system will take into consideration informing users in a clear and understandable way about how their data is processed and how they can exercise their data subjects' rights in accordance with the GDPR;
- Accuracy and facilitating the rectification of inaccurate information where possible: partners endeavour to ensure that any personal data being processed is accurate, and that data subjects can exercise their right to non-rectification where possible

[3] Such as Articles 25, 32 and 89 of the GDPR.
[4] Article 6 of the GDPR.
[5] Article 5 of the GDPR.
[6] Articles 12-14 of the GDPR.
[7] Articles 15-22 GDPR.
[8] See Article 25(1) of the GDPR.
[9] See Article 25(2) of the GDPR.
[10] Stephanie Rossello, Luis Muñoz-González, and Roberto Díaz Morales, "Data protection by design in AI? The case of federated learning" (2021) Computerrecht: Tijdschrift voor Informatica, Telecommunicatie en Recht, 3: 273.

2.3.3. *Compliance with data quality and transparency requirements under the AI Act*

In consideration of the pilots where the THEMIS 5.0 platform is to be tested, the AI models to be trained could be part of an AI system that qualifies, under the AI Act, as a minimal-risk system subject to transparency requirements, or as a high-risk system due to its full incorporation into a medical device. The question that this section aims to address is the following: which of the requirements that may potentially apply in future deployments of Themis require intervention at the design phase, to make sure that the design is future-proof when the AI Act will apply to it?

This question should be answered with a separate analysis for the requirements applicable to minimal-risk and high-risk AI systems, to assess whether the curation of benchmark training datasets affects compliance by design with the AI Act.

Starting from the transparency requirements, these are relevant to the AI-driven conversational agent. Under Article 50 of the AI Act, providers of AI systems should ensure that AI systems intended to interact directly with natural persons are designed and developed in such a way that the natural persons concerned are informed that they are interacting with an AI system, unless this is obvious from the point of view of a natural person who is reasonably well-informed, observant and circumspect, taking into account the circumstances and the context of use. Compliance with such transparency requirements would depend on the information provided in the context of deployment of the AI-driven conversational agent, and not on the composition of training datasets and how training takes place. It can thus be excluded that the transparency requirements are in the scope of our assessment on compliance by design.

As concerns, instead, compliance with the requirements on high-risk AI systems, interventions in the design phase can be valuable as well as necessary. In particular as concerns benchmark training datasets, they should comply with the data and data governance obligations laid down in Article 10 of the AI Act. Paragraphs 2 to 4 set out a number of requirements for training, validation and testing datasets. These require that datasets are subject to data governance and management practices appropriate for the intended purpose of high-risk AI systems, such as on data collection, bias detection and mitigation, availability, quantity and suitability of the requisite data. They also require that datasets be relevant, sufficiently representative, and to the best extent possible, free of errors and complete in view of the intended purpose, and that they take into account the characteristics or elements that are particular to the specific geographical, contextual, behavioural or functional setting within which the high-risk AI system is intended to be used.

Partners have been informed about these requirements, and how the co-creation of benchmarked datasets can achieve the requisite data quality to comply with Article 10 of the AI Act, in particular as concerns accuracy and representativeness. Compliance with Article 10 can also facilitate the respect of accuracy and robustness requirements laid down in Article 15 of the AI Act, which also applies to the provision of high-risk AI systems.

It should be noted that the integration of ethical considerations, including ethical conceptions on diversity and fairness, in the curation of datasets can contribute to, and facilitate, compliance with Article 10 of the AI Act. However, it is important to also stress that ethical and legal requirements on data quality and concepts on specific issues such as fairness and representativeness do not coincide. Ethical considerations can provide valuable guidance on the interpretation of legal requirements, but they pertain to a different domain and identical terms may take on a different meaning. Ethical and legal requirements and considerations should therefore be kept separate.

2.4. Multilingualism support

The multilingual support aspect of the THEMIS 5.0 platform is to all intents and purposes an important building block in building trust in AI systems: the general user can better understand and make even more informed choices if the different steps required in the use of the platform are available in his or her native language. For this reason, despite the unavailability of the tool originally planned in the proposal phase[11], the consortium found it necessary to evaluate the various possible and available alternatives and settled on i18next[12] for

[11] As written within T4.1 description, the tool originally planned was the EC CEF eTranslate Building Block, but it is no more available as can be verified at https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eTranslation.

[12] https://www.i18next.com/

general purpose use, while leaving room for specific implementations related to the conversational agents, as illustrated below. The latter is a popular internationalization framework for JavaScript applications, and it's widely used for translating text, managing localization, and handling multiple languages in frontend and backend applications. Additionally, given the already extensive use of the JSON format for the exchange of information between project software components, the use of this technological framework for internationalisation becomes fluid and versatile in its use by all partners involved.

2.4.1. i18next: hands on guidelines

As said, i18next is a powerful internationalization (i18n) framework for JavaScript applications. It supports frontend (React, Vue, Angular) and backend (Node.js, Express, Next.js, etc.) localization and provides features like:

- translation management (with JSON-based translation files);
- dynamic language switching;
- pluralization and context-based translations;
- date, time, and number formatting;
- language detection;
- middleware support for Express, Fastify, and other backends.

A brief in-depth view is given in the following.

2.4.1.1 Installation & setup

i18next can be installed along with useful plugins through a simple bash command:

```
npm install i18next react-i18next i18next-http-backend i18next-browser-languagedetector
```

where:

- i18next is the core i18n library;
- react-i18next is the react-specific integration;
- i18next-http-backend loads translations from an external server (optional);
- i18next-browser-languagedetector is able to detects user language automatically.

After installing, set up i18next in a config file (e.g., i18n.js):

```
import i18n from 'i18next';
import { initReactI18next } from 'react-i18next';

i18n.use(initReactI18next).init({
  resources: {
    en: { translation: { welcome: "Welcome!", goodbye: "Goodbye!" } },
    fr: { translation: { welcome: "Bienvenue!", goodbye: "Au revoir!" } }
  },
  lng: "en", // Default language
  fallbackLng: "en", // Fallback language if translation is missing
  interpolation: { escapeValue: false }
});

export default i18n;
```

Once i18next is configured, you can use translations in your React components:

```
import { useTranslation } from 'react-i18next';
```

```
function Welcome() {
  const { t, i18n } = useTranslation();

  return (
    <div>
      <h1>{t('welcome')}</h1>
      <button onClick={() => i18n.changeLanguage('fr')}>Switch to French</button>
    </div>
  );
}

export default Welcome;
```

where:

- the t('welcome') function retrieves the translated text;
- the i18n.changeLanguage('fr') method switches languages dynamically.

2.4.1.2 Advanced features

In this paragraph is shown how to use external translation files, activate language detection, support pluralization, context-based translation and formatting dates and numbers.

2.4.1.2.1 Using External Translation Files

Instead of defining translations in i18n.js, it is possible to store them in separate JSON files. In detail, within this file structure:

```
/public/locales/
  ├── en/translation.json
  ├── fr/translation.json
```

It is possible to have:

- en/translation.json

```
{
  "welcome": "Welcome!",
  "goodbye": "Goodbye!"
}
```

- fr/translation.json

```
{
  "welcome": "Bienvenue!",
  "goodbye": "Au revoir!"
}
```

Then, update i18n.js to load external JSON files:

```
import i18n from 'i18next';
import { initReactI18next } from 'react-i18next';
import HttpBackend from 'i18next-http-backend';
```

```
import LanguageDetector from 'i18next-browser-languagedetector';

i18n
  .use(HttpBackend) // Load translation files
  .use(LanguageDetector) // Detect browser language
  .use(initReactI18next)
  .init({
    fallbackLng: "en",
    interpolation: { escapeValue: false },
    backend: {
      loadPath: "/locales/{{lng}}/translation.json" // Fetch translations
    }
  });

export default i18n;
```

This way i18next will automatically load translations from JSON files.

2.4.1.2.2 Language detection

The i18next-browser-languagedetector plugin can detect a user's language from:

- Browser settings
- Local storage
- URL parameters
- Cookies

It is possible to configure it in i18n.js:

```
i18n
  .use(LanguageDetector)
  .init({
    detection: {
      order: ['querystring', 'cookie', 'localStorage', 'navigator', 'htmlTag'],
      caches: ['cookie']
    }
  });
```

Now, if a user's browser is set to French (fr), i18next will automatically switch to French translations.

2.4.1.2.3 Pluralization

Plural forms vary by language. i18next supports it using {{count}} variables, for instance:

- en/translation.json

```
{
  "item": "You have {{count}} item",
  "item_plural": "You have {{count}} items"
}
```

- Usage in React Component

```
<p>{t('item', { count: 1 })}</p>  // "You have 1 item"
<p>{t('item', { count: 5 })}</p>  // "You have 5 items"
```

i18next will automatically choose the singular or plural form based on the count.

2.4.1.2.4 Context-based translations

Use context for gender-specific or role-specific translations, for example:

- en/translation.json

```
{
  "friend": "Your friend is online",
  "friend_male": "Your boyfriend is online",
  "friend_female": "Your girlfriend is online"
}
```

- Usage in React

```
<p>{t('friend', { context: 'male' })}</p>  // "Your boyfriend is online"
```

2.4.1.2.5 Formatting Dates & Numbers

i18next can format dates, times, and numbers according to locale. In detail:

- Usage in React

```
const today = new Date();
<p>{t('date', { date: today, format: 'long' })}</p>
```

- Define format in i18n config

```
i18n.init({
  interpolation: {
    format: (value, format, lng) => {
      if (value instanceof Date) return new Intl.DateTimeFormat(lng).format(value);
      return value;
    }
  }
});
```

i18next will now format dates and numbers correctly based on the language.

2.4.2. Specific solutions for conversational agents

In this section, the specific solutions adopted for the conversational agents presented in Section 3.6 are described.

For what regards the conversational agent for socio-technical evaluation

The IBM Watson Language Translator can be a called as an API in course of the natural language processing feature covers the majority of EU spoken languages. It is to be noted that most of LLM technologies are natively trained in English, therefore the token granularity may not fit well to other languages.

For what regards the Persona conversation agent

For the 1st release of the Conversational Agent for Human in the Loop, the general-purpose employed LLM, Llama[13], while not specifically designed for multilingual tasks, can operate effectively in multilingual environments. However, for the next release, dedicated multilingual LLMs, such as BLOOM[14], which are trained on diverse multilingual datasets and provide responses in various languages, will be tested. If these models exhibit low performance, alternative solutions specifically designed for underrepresented languages, such as EuroLLM[15] or Teuken[16], will be evaluated. These models are trained with a particular focus on supporting lesser-represented languages. If none of these approaches deliver satisfactory results, an alternative pipeline will be considered. This would involve running the chatbot entirely in English and translating both user queries and chatbot responses into the target language. This method ensures that the chatbot benefits from the high accuracy and advanced reasoning capabilities of English-trained LLMs while still delivering responses in the user's preferred language. Translation tools, accessed through dedicated APIs such as Python's g*oogletrans*17 library or LibreTranslate[18], can facilitate this approach. Additionally, other solutions such as eTranslation[19], Digital Europe Programme Language Technology[20], or an internationalization (i18n) framework like *i18next*21 — which supports static content translation using pre-translated JSON files — can be leveraged.

[13] https://www.llama.com/

[14] https://bigscience.huggingface.co/blog/bloom

[15] https://sites.google.com/view/eurollm/home

[16] https://huggingface.co/openGPT-X/Teuken-7B-instruct-research-v0.4

[17] https://pypi.org/project/googletrans/

[18] https://github.com/LibreTranslate/LibreTranslate

[19] https://commission.europa.eu/resources-partners/etranslation_en

[20] https://language-tools.ec.europa.eu/

[21] https://www.i18next.com/

3. DEVELOP TRAINED AI MODELS FOR IMPLEMENTING THE THEMIS 5.0 TRUSTWORTHY SERVICES

The THEMIS 5.0 platform leverages a diverse range of algorithms—from rule-based approaches to state-of-the-art (SOTA) AI models requiring training—to interpret data and support decision-making processes. To effectively manage and oversee the development of these algorithms, it is critical to specify them in detail, thereby enhancing transparency and fostering a shared understanding among all components that incorporate them. Drawing on the work carried out in WP2 and the TOP, Method and Model cards are used to capture and document each algorithm's specifications. The following sections provide a comprehensive overview of the algorithms employed by the THEMIS 5.0 platform, while the respective documentation cards can be found in Annex 2 (Section 9).

3.1. Trustworthiness Assessment

3.1.1. Overview

One of the core functionalities of the THEMIS 5.0 is the assessment of trustworthiness across the TCs of fairness, accuracy, and robustness. Depending on the AI models and datasets of the AI System, different algorithms can be employed to execute its trustworthiness assessment across multiple metrics. In the following sub-sections details about the developed algorithms are provided.

3.1.2. Algorithm details

3.1.2.1 Fairness

The **FairnessEvaluator** is a component within THEMIS 5.0 designed to assess bias and promote equitable outcomes in ML model predictions. It leverages the AI Fairness 360 (AIF360) toolkit to compute a variety of observational, group-oriented fairness metrics such as Disparate Impact, Statistical Parity Difference, and Equal Opportunity Difference. These metrics—often categorized under group fairness—measure disparities between privileged and unprivileged groups by comparing outcome rates or error rates, although the tool can be extended to individual fairness metrics and even causality-based approaches in future iterations. By automatically preprocessing data (standardizing positive/negative labels and encoding categorical features), the FairnessEvaluator streamlines the evaluation process for protected attributes like race, gender, or age.

Users provide a dataset containing ground truth labels, predicted labels, and at least one protected attribute. The FairnessEvaluator then calculates the selected metrics, generating a structured report with numerical values and concise descriptions of each metric. This functionality supports regulatory compliance, transparency, and ethical standards by allowing human decision-makers to identify and address disparate impacts on unprivileged groups, thereby building trust in AI systems and ensuring alignment with human-centred goals.

The FairnessEvaluator is also designed for integration with the Quantitative Assessor, where it can complement other tools such as misclassification or out-of-distribution detection and explainability modules. The component maintains flexibility to handle CSV files or DataFrames, and it can accommodate scenarios where ground truth data is missing by providing preliminary fairness insights under certain constraints. Ultimately, it delivers a clear perspective on fairness risks, helping users refine their models and uphold ethical principles.

Additional information on the FairnessEvaluator is provided in Section 9.3.

Table 14: Trustworthiness Assessment – Fairness specification

Field	Value
Component	TAI Lifecycle Actuator (Quantitative Assessor)

Type of algorithm	Statistics-based
Licencing	Apache 2.0 licence
Computational Needs	Medium (low scale algorithms)
Training	Not required
Training Responsibility	-
Training Data	-
Benchmarking	Applicable
Benchmarking dataset	From THEMIS

3.1.2.2 Accuracy

Support for assessment of technical accuracy in the THEMIS 5.0 platform will be supported by a general toolbox named **SafetyCage** that provides various methods and associated ML models for misclassification detection. SafetyCage is implemented as a Python library, which will be made open source, and integrated as a core component of the **Quantitative Assessor**. The following ML-based misclassification detection methods will be supported by SafetyCage:

- **MSP (Maximum Softmax Probability)** labels the model prediction as correct or not[22]. A requirement is that the prediction model must have a softmax function applied on the output layer, such that the sum of all output neurons equals one. By doing so, each neuron activation value in the output layer can be interpreted as a probability, as is standard practice for neural network classifiers. The MSP detector works by comparing the value of the output neuron with the highest softmax probability, with a threshold $\alpha \in [0,1]$. If the MSP value is less than α , then the prediction is labelled as incorrect by the detector. Otherwise, the prediction is labelled correct. The threshold is tuned on a hold-out dataset, optimizing for what is desired (number of correct detections, or number of detections that correctly labelled predictions as incorrect.). See original work in.
- **DOCTOR:** DOCTOR for misclassification detection was proposed, based on an approximation of the misclassification probability, $Pe(x)$ for a particular sample x by only using the softmax output layer values, $\mathrm{P}_{\{\hat{Y}|\,\mathrm{X}\}}(\mathrm{c}\mid \mathrm{x})$ for each class c of total C classes. In particular $\mathrm{P}_{\{\hat{Y}|\,\mathrm{X}\}}(\mathrm{c}\mid \mathrm{x}) \approx 1 - \sum_{\{\mathrm{c}=1\}}^{\mathrm{C}} \mathrm{P}^2_{\{\hat{Y}|\,\mathrm{X}\}}(\mathrm{c}\mid \mathrm{x})$. The method flags a prediction as untrustworthy whenever the odds of a misclassification event is larger than some threshold $\alpha \in [0,\infty]$.
- **Mahalanobis method:** A data-driven approach which given historical information of correct and incorrect predictions, infers given an input sample x' whether the corresponding prediction y' is correct or incorrect by inspecting hidden and output layers in the ML model using the Mahalanobis method[23]. The output of the Mahalanobis detector is a p-value, and a prediction is deemed incorrect whenever p-value is less than some specified $\alpha \in [0,1]$.
- **SPARDACUS method:** A data-driven approach which given historical information of correct and incorrect predictions, infers given an input sample x' whether the corresponding prediction y' is correct or incorrect by inspecting hidden and output layers in the ML model using the SPARDA method[24] () that, for each layer, searches for the 1D direction that optimizes the Wasserstein distance between historical correct and incorrect predictions for each class. The output of the SPARDACUS detector is a p-value, and a prediction is deemed incorrect whenever p-value is less than some specified $\alpha \in [0,1]$.

[22] https://openreview.net/forum?id=Hkg4TI9xl
[23] https://proceedings.mlr.press/v233/johnsen24a/johnsen24a.pdf
[24] https://proceedings.neurips.cc/paper_files/paper/2015/file/83fa5a432ae55c253d0e60dbfa716723-Paper.pdf

More details about the SafetyCage Python library and its supported methods can be found in the method and associated model cards in Section 9.1.

Since each of the misclassification methods has its pros and cons, SafetyCage will facilitate an automatic procedure in which one uses the misclassification method that is most suited for each particular situation. Output for each method and each single pair of input sample and prediction from the prediction model is a raw value that quantifies uncertainty in prediction. For visualization for the user, we plan to facilitate a traffic light visualization in which green means that the prediction can be trusted, orange means that the prediction should be regarded as uncertain, and red means that the prediction is deemed to be incorrect. In addition, if historical data is available and for a particular input sample x flagged as giving a wrong prediction y by the detector, we will provide to the user other input samples z_1,…, z_n with corresponding true labels y_1,…, yn that are similar to x to be used as a decision-support for evaluating whether the prediction indeed is wrong or not.

The methodology for detecting **misclassification** consists of two models: (1) an ML model making classification predictions for a particular use case, and (2) a decision-supported ML based misclassification detection model which detects whether the classification model has made a correct or incorrect classification. For clarification, we establish the following naming standard for each respective misclassification detection model: The model which performs the prediction (read: cancer classification, fake news detection or estimation of time of arrival at a dock) is called the ***prediction model***. The ML-based methods that detect incorrectly and correctly made predictions by the prediction model is called the ***detector***, or the ***misclassification detector***.

A direct effect of the above-mentioned methods will deliver a higher accuracy during deployment, due to the mitigation of wrongly classified samples made by the model predictor, supplemented by model explanations made by another ML-based model (SHAP). These tools will help the end use (expert) to work with the AI models to make the right decisions.

The contribution to THEMIS is a decision support for the end user on whether the model prediction is reliable. Because different misclassification methods will be used in different scenarios, their respective outputs will have different meanings. SPARDACUS outputs a p value, while DOCTOR provides odds, and so on. Thus, a unified heuristic metric must be implemented and presented for the end user. Such a heuristic metric could be levels of confidence that the prediction is correct (red: not confident, yellow: somewhat confident, and green: confident). The number of levels is to be determined. An explanation provided by an algorithm such as SHAP would be accompanying the misclassification detection, such that the end user also can see how much and which parts of the input to the model drove the model to its decision. By doing so, the end user, which would also be the domain expert can reason about whether the misclassification is reasonable. Consider a model classifying a cancer sample as benign, when it is in fact malignant. The detector will flag the prediction as incorrect and be supplemented by another image depicting which parts of the image was used to drive the model to classify the cancer sample as benign.

Table 15: Trustworthiness Assessment – Accuracy specification

Field	Value
Component	TAI Lifecycle Actuator (Quantitative Assessor)
Type of algorithm	Artificial Intelligence
Licencing	Apache 2.0 licence
Computational Needs	Medium - High (low to large scale algorithms)
Training	Required
Training Responsibility	Partner

Training Data	From THEMIS
Benchmarking	Applicable
Benchmarking dataset	Open source

3.1.2.3 Robustness

Support for assessment of technical robustness in the THEMIS 5.0 platform will be supported by a general toolbox named **OODGuard** that provides various methods and associated ML models for out-of-distribution (ODD) detection. ODDGuard will be implemented as a Python library, which will be made open source, and integrated as a core component of the **Quantitative Assessor**. The following ML-based misclassification detection methods will be supported by OODGuard:

- **Machine learning based out-of-distribution (OOD) detection:** The purpose of OOD detection is to flag whenever an input sample from a particular model prediction is vastly different from the data used to train the prediction model. If this is the case, this is a good indication that the corresponding prediction should not be trusted due to the knowledge that ML models are not good at extrapolation. There are several SOTA approaches, some require historical data[25], however some do not[26].
- **Machine learning based counterfactual explanation:** A counterfactual explanation to an input sample x, is a minimal change of input from x to x' such that prediction changes from y to y'. This is useful for the user to build trust on the prediction, and to evaluate the robustness of the prediction model. The user will be presented for several counterfactual explanations. How to generate these counterfactuals may require generative AI methods.

More details about the OODGuard Python library and its supported methods can be found in the method and associated model cards in Section 9.2.

As described above, there exists many SOTA approaches for OOD detection and counterfactual explanations. The plan is to integrate these software packages together in a repository, and in the same spirit as the SafetyCage repository, facilitate an automatic procedure where the best suited method is applied for each specific situation based on the characteristics of the input sample x.

Table 16: Trustworthiness Assessment – Robustness specification

Field	Value
Component	TAI Lifecycle Actuator (Quantitative Assessor)
Type of algorithm	Artificial Intelligence
Licencing	Apache 2.0 licence
Computational Needs	Medium - High (low to large scale algorithms)
Training	Required
Training Responsibility	Partner
Training Data	From THEMIS
Benchmarking	Applicable
Benchmarking dataset	Open source

[25] https://proceedings.neurips.cc/paper_files/paper/2018/file/abdeb6f575ac5c6676b747bca8d09cc2-Paper.pdf
[26] https://arxiv.org/abs/2002.11297

3.2. Decision impact of socio-technical system

3.2.1. Overview

IBM's Super Decision Engine framework is based on a series of AI agents (Wooldridge M. et al. 1995) that proceed to ethical investigations that correspond (under the view of the final user of Themis systems) to stages of pre-processing for a trustworthiness assessment. The SUT is blind to impacts on the socio-technical system therefore the first stage is the preparation of data in view of their Natural Language Processing. In parallel, a knowledge modelling of black-box components of the SUT is prepared (esp. In a graph mode such as provided by UoS Spyderisk). Then several agents (composed of a LLM and other pieces of software) come into play: the "Business Policy Agent", the "Ethical Agent" and the "Practical Recommendations" Agent. The output of each phase is stored and a last "Verification Agent" processed the initial prompt together with output given by other steps. The result is issued to the user in the chosen language. The outcome of that agentic process is to augment the user's ability to evaluate and anticipate impacts on the STE, bring better measurements to the Trustworthiness Assessor and retroact on the SUT if necessary.

Table 17: Decision impact of socio-technical system specification

Field	Value
Component	Super Decision Engine
Type of algorithm	Rule Based and Artificial Intelligence
Licencing	Depending
Computational Needs	High (large scale algorithms)
Training	Maybe Required
Training Responsibility	Partner
Training Data	Open source
Benchmarking	Not Applicable
Benchmarking dataset	-

3.3. Persona Predictor

3.3.1. Overview

Themis 5.0 incorporates the human perspective into the evaluation of trustworthiness by assessing both human behavioural and moral value aspects. To facilitate this, it takes into consideration users' behavioural and moral value aspects. For this purpose, two classifiers have been developed: i) the Persona Predictor, for the identification of users' persona type, which is based on the Persona Identification Methodology (see Annex 5) and for the 1st release it is limited to the identification of the trustworthiness preferences, and ii) the Moral Orientation Predictor, for the identification of users' moral orientation (the Moral Orientation Methodology is described in Section 3.7.4.2 in D2.1) and for the identification of users' ethics approach (virtue ethics, deontological ethics, and utilitarian ethics).

All classifiers have been adapted for multilingual processing. Whenever a classification is required, the system ensures that the classifier interprets user responses within the context of the selected language. This setup guarantees native support for all target languages on open-ended user questions, while also seamless and accurate interactions across different language settings.

3.3.2. *Algorithm details*

To implement the classifiers in use, we have leveraged the recent advancements in LLMs, which are typically pre-trained on vast amounts of data, enabling them to interpret human language and perform complex tasks with impressive accuracy. In many cases, these models can handle such tasks without the need for further training or fine-tuning, demonstrating an excellent, out-of-the-box performance.

Llama 3 is an open-source model developed by Meta, known for outperforming many other open-source models and delivering results comparable to leading closed-source solutions like GPT-4[27]. For this task, we have utilized the model Llama 3.1[28] with 8 billion. It supports a 128K token context window and incorporates advanced tokenization techniques. Additionally, custom-built server clusters are used to enhance performance further, making it an excellent choice for powerful, open-source LLM applications and an ideal solution for our application. This LLM best meets our classification needs while avoiding the intensive time frames and resource requirements associated with larger models.

For each classifier, the Llama 3.1 model is tasked with performing the relevant classification. To facilitate this, we followed the prompting approach. Specifically, for the Persona Identfier, we provide the model with a comprehensive task description and a detailed explanation of each expected class, or the expected rules to be followed when classifying, along with the characteristics that define each class. This allows the LLM to better understand the specifics of each classification task. The rest of the classification process is then handled autonomously by the LLM.

Examples of class descriptions for different classifiers can be found in the figure below.

```
Given an input corresponding to a possible answer, classify it based on its predefined trustworthiness
  preference:

Mappings:
- "Less than 5 years of experience" -> "Robustness"
- "5-15 years of experience" -> "Accuracy"
- "More than 15 years of experience" -> "Fairness"

- "Very comfortable" -> "Robustness"
- "Somewhat comfortable" -> "Accuracy"
- "Not comfortable at all" -> "Fairness"

- "In favor" -> "Robustness"
- "Sceptical" -> "Fairness"
- "Neutral" -> "Accuracy"
```

Figure 14: Class description for the classifier of the Persona module

```
Fairness:
- Characteristics: High priority on transparency, ethical standards, and explainability.
- Behavior: Prefers AI systems that provide detailed explanations, maintain ethical integrity, and ensure trust.
- Needs: AI that offers assurances regarding bias, data integrity, and ethical decision-making.

Robustness:
- Characteristics: Focuses on efficiency, reliability, and adaptability, often over transparency.
- Behavior: Prefers tools that are functional, fast, and dependable, even in uncertain or changing environments.
- Needs: Reliable, advanced AI solutions that reduce complexity and handle innovative applications.

Accuracy:
- Characteristics: Requires tools that provide precise, structured, and actionable insights.
- Behavior: Values AI that enhances efficiency, minimizes ambiguity, and improves decision-making.
- Needs: AI that balances usability with high accuracy and clear, verifiable outputs.

Not Applicable:
- Characteristics: Doesn't fit into any of the above categories.
- Behavior: Responses do not indicate a clear preference for Fairness, Robustness, or Accuracy.
```

Figure 15: Class description for the classifier of the Disambiguation module

[27] https://openai.com/index/gpt-4/
[28] https://ai.meta.com/blog/meta-llama-3

Table 18: Persona Predictor specification

Field	Value
Component	Persona Predictor
Type of algorithm	Large Language Models (LLMs)
Licencing	Apache 2.0 licence
Computational Needs	Medium (low scale LLM models)
Training	Required
Training Responsibility	Partner
Training Data	From THEMIS
Benchmarking	Applicable
Benchmarking dataset	From THEMIS

3.4. Risk Evaluation

3.4.1. Overview

Risk of the AI SUT, and the wider system which encapsulates it, is evaluated using the Spyderisk tool. Spyderisk is a rules-based simulator that simulates and assesses the risks, and their risk levels, for a given modelled system. It does this by performing fuzzy risk calculations using linguistic likelihood variables and using a knowledge base containing information on domain assets, their relationships, threats that can affect them, and consequences that would result from the threats. The tool is designed to perform automated risk assessment to assist the AI actor in understanding the risks in their system and also the controls that can be used to mitigate and address these.

3.4.2. Algorithm details

Spyderisk is rules-based and so does not need any training, though extensions to the knowledge base may be needed to correctly capture required information about a domain. There are several general stages to the workflow involving Spyderisk, including system model design, risk calculation, and risk exploration.

In the system model design, the user of Spyderisk designs the system model, here representing the SUT. They then validate it, set impact levels for consequences they consider important, and apply initial controls that represent base controls that are, or will be, present in the system by default. For the case of the SUT, the user here is the person who will be performing and exploring the risk assessment of the SUT.

Impact levels of consequences related to the different AI trustworthiness characteristics can be set by the user, but they can also be set using the different user profiles. Here the user profiles give the user preferences of which AI trustworthiness characteristics are most important to that profile and so indicate which of the related consequences should have higher impact levels to give them more prominence.

With the system model designed, the user then performs risk calculations to determine the level of risk within the system, so that they can explore the risks present and understand them, allowing them to decide on appropriate controls to apply to mitigate or blocks the different risks, bringing the risk level own to an appropriate level.

The risk calculation itself follows an iterative algorithm that assesses all the possible threats in the system model to determine their likelihoods and how they interconnect. The algorithm starts with typically low threat likelihood levels and drives them up if preceding threat paths dictate it to, until it converges to a steady state where the different threat likelihoods do not increase any more. Here threat paths connect threats and consequences together in chain consisting of multiple threats and consequences, where prior threats occurring can result in consequences that trigger further threats, and so on. This allows the algorithm to

consider how threats may be interconnected and how the threats, consequences and risks propagate in the system.

With the risk calculation performed, the user can then explore the risks present and apply controls to address risks they deem to be too high. Risk exploration can take place by the user exploring threat paths (or attack paths) within the Spyderisk graphical interface, using attack path analysis algorithms to determine the root cause threat to a given path, and controls that can be applied to block or mitigate either the initial root cause threat to a given attack path or other threats along the path. This then results in the likelihood and risk of the final consequence a the end of the path being reduced. The intention here is that any controls applied will be implemented in the SUT itself, so that the updated, and reduced, risks in the risk evaluation reflect the actual risks and risk levels for the SUT.

In addition to this risk exploration, the Super Decision Engine tool can extract attack paths for given consequences of concern from Spyderisk so that the user can query this tool and explore the attack paths using natural language to better understand the paths, threats and risks present.

3.4.3. Technical Specifications

The Spyderisk backend is written in Java, and the graphical user interface is written in JavaScript. Additionally, the Spyderisk Adaptor tool may be used if needed, and this is written in Python. All these component parts of Spyderisk can be deployed through Docker, with there being Docker images for them.

Table 19: Risk evaluation specification

Field	Value
Component	Spyderisk (Risk Evaluation)
Type of algorithm	Rule based
Licencing	Apache 2.0
Computational Needs	Medium
Training	Not required
Training Responsibility	-
Training Data	-
Benchmarking	Not Applicable
Benchmarking dataset	-

3.5. Solution Recommendation

3.5.1. Overview

The developed algorithms are designed to assist the AI actor in selecting appropriate enhancement methods (or controls, in the context of risk management) to improve the trustworthiness of the AI system. These algorithms integrate inputs from the risk management framework, the SUT documentation, and user persona preferences. This comprehensive approach enables the formulation of solutions that combine multiple methods to mitigate risks associated with the AI system's trustworthiness.

3.5.2. Algorithm details

The implemented algorithms are rule-based and do not require any training. Their functionality is structured into two primary phases: **Method Exploration** and **Solution Recommendation**.

In the first phase, data from documentation cards and the risk management framework is integrated into a knowledge graph (KG). The KG's purpose is to establish a pathway linking the specific requirements and properties of the SUT to available enhancement methods. Initially, a graph is constructed to represent the current state of the SUT. Starting from a central "SUT" node, relationships are established with nodes representing its characteristics—for example, the stage of the AI system, the models employed, and the datasets utilized. The graph is then expanded to include recorded enhancement methods that share similar characteristics, thus creating pathways from the "SUT" node to applicable methods. The output of this phase is a set of applicable methods that can mitigate the trustworthiness limitations of the SUT.

In the second phase, given that the enhancement methods can be tested, solutions are created and evaluated. Depending on the number of applicable methods identified in the previous phase, solutions are created that incorporate multiple methods. This can be performed exhaustively (testing all combinations), heuristically, or by instructions from the human supervisor. All the solutions should be tested in a development environment based on which the residual risks can be calculated. These risks are used to determine the impact of each solution on the AI system across various metrics. With both the solutions and their impact, methodologies such as multi-criteria decision-making (MCDM) process can be employed to assist the supervisor in selecting the most appropriate solution. Notably, the supervisor's preferences across the different TCs can influence the recommended solutions. Furthermore, based on the documented requirements and KPIs of the SUT, solutions that do not meet performance thresholds can be automatically discarded.

3.5.3. Technical Specifications

The knowledge graph is constructed using Python in combination with the Neo4j graph database. Key entities, such as the SUT and the enhancement methods, are defined as Python classes and instantiated using data from the documentation cards. Once instantiated, these entities are loaded into the KG, which can then be queried to retrieve methods based on their relationships to the SUT and its assets.

The VIKOR methodology is currently under consideration for the MCDM process. This methodology is chosen for its ability to aggregate multiple solutions across different trustworthiness criteria, incorporate human preferences as weighting factors, and balance the potential regret associated with selecting a particular solution. The implementation of the VIKOR methodology is also executed in Python.

Table 20: Solution Recommendation specification

Field	Value
Component	TAI Lifecycle Actuator (Solution Navigator)
Type of algorithm	Rule based
Licencing	TBD
Computational Needs	Medium (low scale algorithms)
Training	Not required
Training Responsibility	-
Training Data	-
Benchmarking	Applicable
Benchmarking dataset	Open-source, THEMIS

3.6. Conversational Agents

The conversational agent contains two subcomponents: the Persona Conversational (named Conversational Agent for Human in the Loop in D2.2), presented in Section 3.6.1, and the Conversational Agent for Socio-Technical Evaluation, presented in Section 3.6.2).

3.6.1. Persona conversational agent

The persona conversational agent is a micro-service that engages the end-user in a text-based dialogue to incorporate the human perspective into the evaluation of trustworthiness by assessing both human behavioural and moral value aspects. Hence, the conversational agent collects user preferences, attitudes, and moral and professional characteristics through a set of pre-defined questions. These data will be leveraged to predict end-user's: i) traits with respect to trustworthy characteristics (TCs), and ii) moral orientation.

It uses a predefined conversation flow as its foundation. Additionally, various independent modules were integrated to enhance the overall conversation experience with more open-ended elements. The chatbot features a straightforward and user-friendly interface, where users first select their preferred language and then, a one-on-one conversation follows, with the chatbot posing questions and the user responding in free text until the conversation cycle concludes.

More specifically, the chatbot is structured into four distinct submodules, namely *Initial module, Persona module, Disambiguation module* and *Moral module,* each of which will be more thoroughly presented on the next subsection. A pre-determined conversation schema governs the sequence in which these modules are activated, while also, a specific conversation schema is applied within each module to determine the order of asked questions. This framework follows a tree-like structure, where each branch represents a unique conversation path, ultimately leading to a set of final nodes within each module. Following this approach, all the necessary information to classify a target user effectively, are systematically being collected.

3.6.1.1 Algorithm details

The overall pipeline can be seen in the figure below.

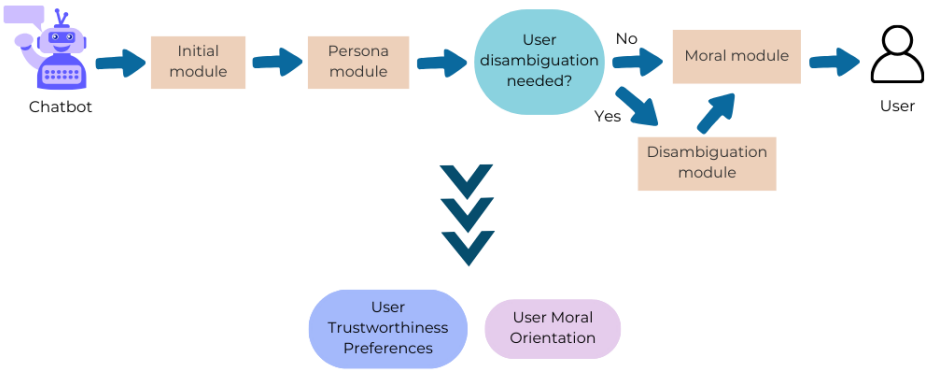

Figure 16: The pipeline of the Persona conversational agent

The conversation begins with the **Initial module**, where the user is firstly welcomed. Then, a question regarding the target domain is asked by the chatbot. Based on the response, the user is categorized into one of the three target domains—healthcare, media, or port management. The conversation then proceeds to the rest of the modules, while tailoring the conversation with domain-specific information for the selected domain.

The **Persona module** defines the users' persona by mapping their responses to an order of trustworthiness preferences, based on the selected domain. More precisely, the chatbot asks a series of questions regarding users' comfort with new technologies, their opinions on AI, and other related topics. Then, the Persona Predictor (described in Section 3.3) processes the responses to assign an appropriate trustworthiness

preference (i.e., Robustness, Accuracy, Fairness). Then, the count occurrences for each of the three target classes is identified. Eventually, based on the final percentages of the count occurrences, each trustworthiness preference is placed on an order of highest to lowest TP.

The **Disambiguation module** interacts with the Persona module in cases where a definitive classification cannot be reached. More precisely, for the case where two or more captured classes have the same count score, then the Disambiguation module is activated to refine the classification through additional questions. These clarification questions prompt the user to explicitly indicate their trustworthiness preferences and are grouped into three disambiguation categories: Fairness vs. Accuracy, Robustness vs. Accuracy, and Fairness vs. Robustness. Thus, for any tie-on class count occurrences, a relevant resolution follows to identify which of the two target TPs, each time, is mostly preferred by the user. The responses of these questions are analyzed, and the module sequentially processes each conflicting pair until it resolves all conflicts. Eventually, the final order of trustworthiness preferences is returned.

The **Moral module** assesses the user's moral orientation by presenting ethical scenarios and evaluating their preferences. Compared to the other modules, this module follows a more structured, tree-based conversation flow, where each user response—typically affirmative or negative—determines the next question dynamically. The Persona Predictor (described in Section 3.3) has been implemented to interpret whether a user's free-text response is affirmative or negative, maintaining flexibility while ensuring structured progression.

3.6.1.2 Conversation Scenario Structuring

To define the chatbot's conversation flow, all scenarios—including questions and information messages—are structured within dedicated JSON format files. These files follow a tree-structured format, where each entry contains a unique question identifier and a reference to the next question identifier, determining the chatbot's next step in the conversation.

The JSON files are organized separately for each supported language and maintain structured sets of questions grouped according to the different modules, while also distinguishing between the supported use cases. This structured approach ensures a modular and scalable conversation design.

An example of this JSON file structure is illustrated in the figure below.

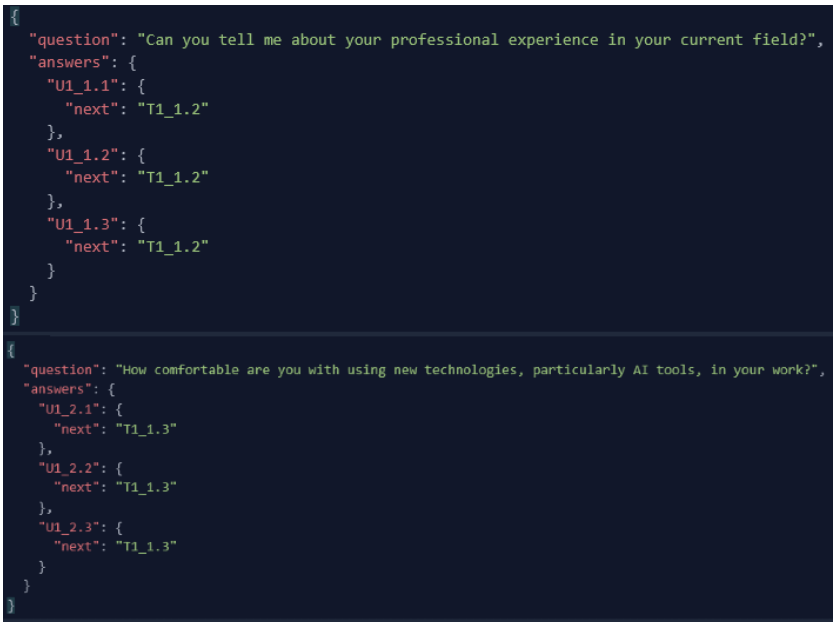

Figure 17: Chatbot's questions and conversation path, structured on JSON format files

3.6.1.3 Data Storage and Database

Concerning the specifics of data storing, each user response is stored in a dedicated database, which organizes the data into groups to facilitate further handling and analysis. For this purpose, SQLite[29] was chosen as the preferred database. SQLite is a highly efficient, lightweight, serverless database that is ACID-compliant, that can implement most SQL standards while also it can operate in-memory. It is also open source, making it a cost-effective solution.

SQLite proved to be an ideal choice for our needs, as the task did not require heavy database storage or complex processes, while it maintained a simplicity that was very useful for our requirements. More precisely, each chatbot question and its corresponding user response are stored in the database, alongside the target question group (i.e. a label, indicating the target module that the sub-conversation takes place). This grouping follows the previously mentioned JSON file structure, allowing the database to track and organize each conversation segment.

An example snapshot of our database can be seen in Table 21.

Table 21: A snapshot of stored chatbot questions, aligned with user responses and a target conversation group

Questions	Responses	Group
What sector do you work in?	health	Initial
Can you tell me about your professional experience in your current field?	6 years	Persona
How comfortable are you with using new technologies, particularly AI tools, in your work?	Very comfortable	Persona
Even though you are comfortable with technology, how important is it for you to understand the inner workings and decision-making process of AI systems you use?	I am interested only a little bit	Disambiguation
Do you believe that AI users in healthcare should be personally empowered with traits like integrity and fairness?	Yes	Moral

3.6.1.4 Web Service Integration

Ultimately, all the aforementioned components have been integrated into a complete pipeline, implemented as a Representational State Transfer (REST) service. The service was developed using the Flask microframework[30], a lightweight and flexible web framework that simplifies the creation and scaling of web applications.

Python was chosen as the primary programming language for the back end of the service, ensuring robust and efficient processing. On the frontend, HTML5, CSS3, and JavaScript were utilized to build the user interface, providing a seamless and responsive experience.

The final user interface can be seen in Figure 18 figure below.

[29] https://www.sqlite.org/

[30] https://flask.palletsprojects.com/en/stable/

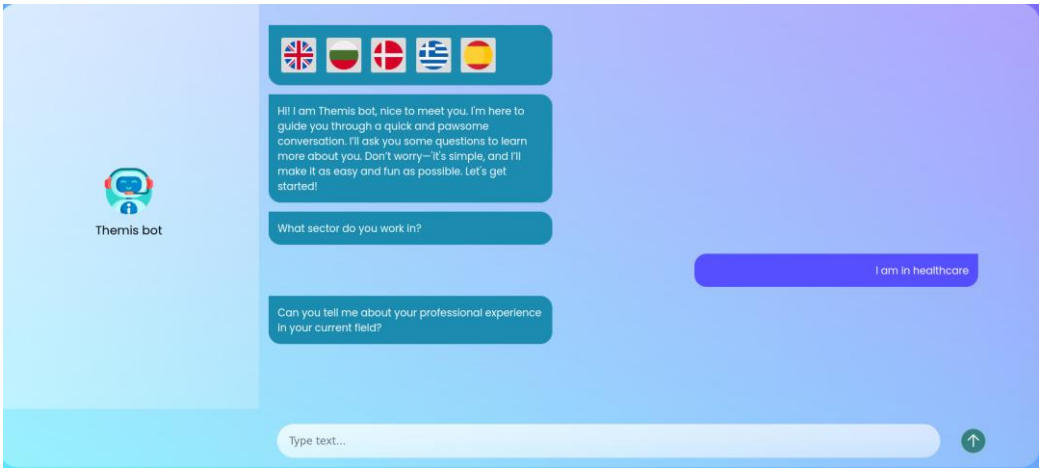

Figure 18: The chatbot user interface

3.6.2. Conversational Agent for Socio-Technical Evaluation

IBM's strategy with regards question answering feature in Themis is to avoid automation bias and complacency bias[31] to preserve the place of human within the computing loop. Namely, the AI recommendation inherits from all the credibility and veracity of the Socio-Technical System. In regards, trustworthiness reclaims a minimum level of doubt concerning computation bias and the chain-of-thought. As a consequence, User Experience and especially User Interface will not implement state-of-the-art practices but rather aim at decoupling the AI output from the (human user) belief.

3.6.2.1 Algorithm details

Different kinds of algorithms play into account in relation with IBM's three agents that are implemented under the crewAI framework. Three agents will be implemented. (1) the Business Policy Agent will be encoded with controlled-natural language (CNL) instructions to describe business rules, based on ontologies specific to the use case at stake, e.g. the Port of Valencia business and technical vocabulary. (2) the Risk Investigation Agent is informed with risks propagation diagrams taken from the University of Southampton Spyderisk framework, and bring some concepts taken specifically from the AI Trustworthiness field. (3) the Practical Recommendations Agent will be built under a human-in-the-loop perspective, to feed final recommendations with a more bottom-up and grassroot approach. Each agent will use algorithms beneath Mistral and Granite Large Language Models. The Risks Assessment Agent, the Practice Recommender Agent will implement RAG and fine-tuned LLMs, and the Business Policy Agents Rete algorithm.

3.6.2.2 Technical Specifications

At the development phase, Spyderisk Risk AssesGraphs is pre-process through Python scripting that produces Json files interoperable with the selected LLMs (optimised through an RDF-derived format). IBM Operational Decisions Engine will play the role of rules repository, to be interfaced with the selected LLMs. The orchestrator will be developed under the IBM Watson-X, either via CrewAI, Langgraph or IBM Bee. Generally speaking, interoperability will be ensured through Rest APIs. Code Engine or Openshift frameworks can be used for the production phase.

---

[31] From *Baudel, T., Verbockhaven, M., Cousergue, V., Roy, G., & Laarach, R. (2021). ObjectivAIze: Measuring performance and biases in augmented business decision systems. In Human-Computer Interaction–INTERACT 2021: 18th IFIP TC 13 International Conference, Bari, Italy, August 30–September 3, 2021, Proceedings, Part III 18 (pp. 300-320). Springer International Publishing.* "Automation bias is clearly related to complacency bias. It can also result from attention deficits, which are even more prevalent in assisted driving tasks".

4. CO-CREATION OF BENCHMARKED DATA SETS AND TRAINING OF THEMIS 5.0 AI MODELS

4.1. Introduction

The co-creation and benchmarking of datasets represent a vital methodology within the THEMIS 5.0 ecosystem, reflecting a commitment to building AI systems that are not only technically proficient but also ethically sound and human centred. By involving diverse stakeholders through collaborative workshops and iterative feedback, this approach ensures that datasets encapsulate the nuances of real-world contexts, including sector-specific requirements and demographic diversity. These datasets form the foundation for training AI models, fostering trustworthiness by embedding fairness, accuracy, and robustness into their very design. In practice, some project partners will integrate pre-existing AI models or services rather than retrain them from scratch. For example, IBM's Granite foundation model (a highly curated, transparent LLM) is used as a service and will not be re-trained on project data. Furthermore, this methodology embraces inclusivity, aligning technological development with the values, needs, and expectations of its users. By curating datasets that prioritize explainability and transparency, the co-creation process strengthens the relationship between AI systems and their stakeholders, enhancing acceptance and fostering confidence in the system's outputs. These datasets, once refined and validated, will be shared openly to advance research and innovation, ensuring that the collective effort contributes not only to the success of THEMIS 5.0 but also to the broader quest for ethical and reliable AI solutions. We anticipate using an open license such as Creative Commons Attribution (CC BY), which allows others to reuse and build upon the data with proper attribution. This approach follows the EU principle of making research outputs *"as open as possible, as closed as necessary,"* balancing openness with privacy/security needs. This human-centric methodology exemplifies how co-creation can bridge the gap between technical capability and societal responsibility.

THEMIS 5.0's datasets are intended to improve the trustworthiness of various AI software services. In practice, some project partners will call pre-existing AI models or services rather than retrain them from scratch. For example, IBM's Granite foundation model (a highly curated, transparent LLM) is used as a service and will not be re-trained on project data. Instead, THEMIS datasets will be used to enhance and evaluate such services' trustworthiness. This means the co-created data will help fine-tune certain model behaviours, calibrate outputs, or serve as test cases to assess fairness and explainability, even if the core model weights (as in Granite) remain unchanged. In summary, THEMIS 5.0 leverages its datasets to augment AI services across the board – whether by training new models where possible, or by informing and validating existing AI services – to ensure all integrated AI components exhibit improved transparency, fairness, and reliability.

4.1.1. ICE/i4RI's Role as Facilitator for Trustworthiness Models

In the co-creation and benchmarking of datasets for trustworthiness models within THEMIS 5.0, ICE/i4RI take on the crucial role of facilitators, orchestrating the development, validation, and refinement of training datasets tailored to the project's diverse applications. As the central hub for this task, ICE/i4RI ensures that the datasets align with the broader THEMIS 5.0 objectives of fairness, accuracy, and robustness, while meeting the sector-specific requirements of healthcare, port management, and media disinformation. This involves leveraging their expertise in training environments and benchmarking methodologies to curate datasets that reflect real-world challenges and user expectations.

Other partners play complementary and interconnected roles in this collaborative process. SINTEF and ATC contribute to the technical validation of the datasets, ensuring that they meet the performance benchmarks required for model training and optimization. IBM and ICCS, with their strong focus on AI architecture and algorithm development, work closely with ICE/i4RI to orchestrate the call of datasets into AI models, providing iterative feedback and enhancements to ensure that these models maintain high standards of explainability and technical accuracy. MAG and ENG bring their domain-specific insights, particularly in healthcare and port management, to ensure that the datasets address operational realities and user needs. Meanwhile, DBT facilitates the human-centric aspects of dataset creation, channelling inputs from co-creation workshops and aligning them with user-centred design principles.

This interplay of expertise creates a dynamic ecosystem where ICE/i4RI act as the linchpins, harmonizing the technical, ethical, and contextual contributions from all partners. Through ongoing collaboration, they enable the iterative refinement of datasets and ensure their readiness for deployment in training trustworthiness models, making this process a true embodiment of THEMIS 5.0's commitment to ethical and human-centred AI.

4.2. Stakeholder Engagement and Co-Creation Process

Building trust and ensuring real-world relevance in AI systems requires the active participation of those who will ultimately be affected by them. This section outlines how THEMIS 5.0 draws on the knowledge, experience, and perspectives of a diverse set of stakeholders—ranging from domain experts and technical developers to end-users and ethical advisors—to shape an AI ecosystem that is ethically grounded, technically robust, and contextually aware. By fostering collaborative workshops, iterative feedback loops, and continuous engagement, the project ensures that every voice is reflected in the design and deployment of trustworthy AI solutions.

4.2.1. Identification of Stakeholders (ICCS, Domain Experts, TAIME Developers)

The **Stakeholder Engagement and Co-Creation Process** is a cornerstone of the **THEMIS 5.0** methodology, ensuring that the ecosystem reflects the diverse insights, values, and practical needs of all its stakeholders. This inclusive process weaves together the perspectives of **ICCS**, **domain experts**, **TAIME developers**, **technical developers**, **ethical advisors**, and **end-users** across critical sectors such as healthcare, port management, and media disinformation. By foregrounding collaboration at every step, THEMIS 5.0 aspires to build AI systems that are not only cutting-edge in their capabilities but also deeply aligned with societal expectations for fairness, accuracy, and robustness.

Central to this engagement are **co-creation workshops**, which form a structured yet flexible environment where participants discuss and refine user requirements, pinpoint key trustworthiness parameters, and propose benchmarks. These workshops are carefully designed to be iterative: stakeholders revisit and refine their contributions, ensuring that as the project's technical architecture evolves, so do its guiding insights. Far from being a mere collection of ideas, these inputs flow directly into the **technical development pipeline**, influencing the very core of the THEMIS 5.0 platform—whether it's the design of trustworthiness metrics, user-facing interfaces, or data curation strategies in domains like healthcare and port management.

Throughout this process, **ICCS** plays a critical bridging role. Leveraging expertise in system modelling, AI trustworthiness assessment, and risk evaluation, ICCS ensures that stakeholder feedback aligns cohesively with the **socio-technical methodologies** and **ethical frameworks** foundational to THEMIS 5.0. Meanwhile, **IBM** contributes its extensive experience in AI lifecycle management, applying scalable, human-centred principles for application and workflow design. Their expertise translates the voices of stakeholders into robust, practical solutions—addressing everything from **multilingual support** to **bias mitigation** and **explainability** in AI outputs.

Equally vital are the **facilitators** at **ICE/i4RI**, who coordinate interactions among the various stakeholders and technical teams. They excel at translating complex technical concepts into accessible discussions, helping participants grasp the implications of algorithmic decisions or data choices without requiring deep AI expertise. Similarly, **DBT** and **SINTEF** organize and synthesize the feedback emerging from co-creation sessions, ensuring that user perspectives remain visible and actionable in subsequent project phases. This comprehensive feedback loop keeps the THEMIS 5.0 ecosystem responsive to evolving societal needs while maintaining alignment with regulatory imperatives and best practices.

In the specific context of Task 4.3—the co-creation of datasets—ICCS and IBM jointly provide the technical leadership needed to ensure these datasets align with the project's overarching goals of fairness, accuracy, robustness, and trustworthy user experience (T-UX). ICCS's expertise in trustworthiness standards shapes the data models and benchmarks, ensuring they reflect the complexity of real-world socio-technical

environments. Concurrently, IBM implements advanced techniques that not only safeguard datasets against bias but also mitigate user biases and automation biases, supporting informed human decision-making.

However, THEMIS 5.0 acknowledges that trustworthy AI cannot be achieved by dataset quality alone. High-quality data is essential, but user trust primarily depends on the quality of the human-AI interaction experience. Automation bias—where users over-trust automated suggestions without adequate scrutiny—poses a significant risk. THEMIS 5.0 proactively counters this by adopting a human-centred T-UX approach, ensuring AI outputs are not presented as infallible answers but rather as dialogue prompts inviting critical reflection.

For instance, the THEMIS conversational agent engages users interactively, clearly explaining the rationale behind recommendations in human-interpretable terms and integrating user-specific context, values, and feedback. It actively provides cues about uncertainty or potential errors to maintain user vigilance and critical engagement. Interface techniques, such as visualizing AI confidence levels, highlighting decision-relevant factors, and prompting users to confirm or question critical recommendations, reinforce active user participation. This design ensures users remain fully aware of their decision-making responsibility, effectively calibrating trust and reducing the likelihood of automation bias.

Through this integrated approach, THEMIS 5.0 establishes a high standard for co-created, trustworthy AI. It develops models and workflows that excel not only in technical performance but are also ethically grounded, inclusive, and attuned to the practical realities and concerns of end-users. By combining robust, bias-resistant datasets with a thoughtfully designed human-AI interaction experience, THEMIS 5.0 ensures trustworthiness extends beyond data quality alone, fostering genuine user understanding, engagement, and responsible use of AI systems.

4.2.2. Integrating Moral, Legal, and Socio-technical Feedback into Dataset Creation

In THEMIS 5.0, the integration of moral, legal, and socio-technical feedback into dataset creation ensures that the foundation of AI training aligns with ethical principles, legal requirements, and the complexities of real-world systems. This process is deeply embedded within the co-creation methodology, drawing from the diverse insights of stakeholders to construct datasets that are not only technically robust but also socially responsible and contextually aware.

Moral feedback is collected through structured engagements with ethical advisors and end-users, focusing on fairness, inclusivity, and the prevention of bias. These inputs guide the selection of data sources and the development of preprocessing strategies to ensure equitable representation across demographics, sectors, and use cases. For example, fairness metrics and transparency requirements derived from stakeholder feedback are directly applied during the curation and validation of datasets.

Legal considerations are integrated in close collaboration with KUL, to receive advice on how to ensure compliance with applicable EU law, and in particular the AI Act and the GDPR. This involves auditing datasets for adherence to data protection laws, ensuring data anonymization, and embedding mechanisms for accountability and traceability. Moreover, contribution from KUL includes advice on how the methodology followed for the creation of datasets can align with the data quality requirements laid down in the AI Act and ensure it even in the case that AI Act does not apply. These legal safeguards provide a foundation for the ethical deployment of AI models trained on the datasets.

The socio-technical dimension is addressed by aligning dataset characteristics with the operational realities of specific use cases, such as healthcare, port management, and media disinformation. Partners like MAG and SINTEF bring their domain expertise to tailor datasets to the socio-technical environments in which AI systems will operate. This includes ensuring that datasets capture contextual nuances, such as operational workflows and stakeholder expectations, while reflecting the broader societal impact of AI systems.

By harmonizing these diverse streams of feedback, THEMIS 5.0 ensures that its datasets not only meet the technical demands of AI training but also embody the moral and socio-technical values critical for trustworthy and human-centred AI. The human-centred approach of THEMIS 5.0 resonates with the self-determinational philosophical principle of information privacy, which emphasizes the right of individuals and groups to control

when, how, and what personal information is shared or communicated to others. This principle is embedded in the normative framework of information privacy within self-determination governance, where moral agency is ethically guided by the assessment of personal decisions within the context of a citizen's value system. This approach exemplifies the project's holistic commitment to building AI systems that are equitable, transparent, and aligned with societal priorities.

4.2.3. Governance Models for Deciding on Protected Attributes and Fairness Criteria

To ensure that fairness is systematically embedded into THEMIS 5.0's AI systems, a robust governance model guides the identification and incorporation of protected attributes and fairness criteria during dataset creation and model training. This governance structure reflects the project's commitment to transparency, inclusivity, and ethical rigor, balancing diverse stakeholder inputs with technical feasibility and legal mandates.

Central to the governance model is a multi-stakeholder decision-making framework. This framework engages domain experts, legal advisors, ethical scholars, and end-users in deliberations to define which attributes, such as gender, race, socioeconomic status, or regional differences, should be treated as protected. The model prioritizes attributes based on their relevance to the specific use cases in healthcare, port management, and media disinformation while ensuring alignment with overarching ethical principles and facilitating compliance with legislation, including the EU's AI Act and GDPR.

Fairness criteria are established through iterative co-creation workshops and feedback loops, enabling stakeholders to propose, debate, and refine metrics that align with their expectations and the socio-technical context of the systems under development. These criteria are operationalized using technical metrics, such as demographic parity or equalized odds, which are tailored to the specific risks and requirements of each use case. The governance model also ensures that these metrics are applied consistently across the project's lifecycle, from dataset creation to model evaluation and deployment.

ICE/i4RI play a pivotal role in facilitating this governance process, coordinating inputs from all partners and integrating the resulting decisions into the technical workflows. Partners like KUL and SINTEF contribute their expertise to navigate legal complexities and domain-specific challenges, while IBM Moreover, and ICCS develop technical mechanisms to encode these fairness criteria into the system's architecture. Furthermore, this collaborative governance framework explicitly integrates human oversight at each step, ensuring a central role for human judgment throughout the optimization process. By embedding continuous human-in-the-loop methodologies—such as co-creation activities, interactive reinforcement learning, and human-centred explainability—the THEMIS 5.0 framework prioritizes human agency and ensures that AI decisions align with ethical, legal, and social values, as well as organizational and end-user expectations

By instituting this governance model, THEMIS 5.0 ensures that decisions about protected attributes and fairness criteria are not made in isolation but reflect a collective, well-informed, and transparent process. This approach not only strengthens the trustworthiness of the resulting AI systems but also serves as a replicable framework for balancing ethical and technical priorities in the design of human-centred AI.

4.3. Benchmark Data Curation

The curation of benchmark data for THEMIS 5.0 forms the backbone of its trustworthiness optimization methodology, ensuring that the platform's AI models deliver suggestions that adhere to the highest standards of fairness, accuracy, and robustness. In machine learning, a benchmark dataset is defined as a standardized, widely recognized collection of data used as a reference point to objectively evaluate and compare the performance of AI models. Such datasets typically provide well-defined tasks or ground truth labels, enabling researchers to consistently measure aspects like classification accuracy, fairness metrics, and robustness against biases or errors. Key characteristics of effective benchmarks include consistency, broad acceptance in the research community, and diversity—ensuring fair, reproducible, and meaningful evaluations across various scenarios and contexts.

Following these best practices, THEMIS 5.0 applies a collaborative, human-centric approach to identifying and integrating diverse, high-quality data sources from critical domains such as healthcare, transportation, and

media. Guided by Task 4.3's co-creation framework, these benchmark datasets are iteratively refined through cycles of user feedback, model training, and performance validation. This dynamic process allows THEMIS 5.0 to adapt effectively to evolving socio-technical contexts.

Central to this effort is the active involvement of stakeholders—including developers, domain experts, and end-users from MAG, IBM, SINTEF, ATC, ENG, ICCS, and IPT—who contribute domain-specific insights, ensuring datasets authentically reflect real-world complexities. For instance, healthcare benchmark data curated by MUP integrates anonymized patient records and clinician inputs, specifically designed to evaluate and mitigate biases in cancer risk prediction models. Similarly, port management benchmark data from FVP includes resilience metrics for infrastructure, evaluating model robustness in handling climate-driven disruptions.

THEMIS 5.0 further commits to rigorous validation standards by clearly defining evaluation protocols, including fixed test splits and predefined success metrics. For example, the fairness and robustness datasets contain curated cases, including specific decision scenarios, diverse user profiles, and challenging edge cases that systematically test model biases and error rates. Likewise, benchmark datasets for the conversational agent incorporate standard dialogues to rigorously assess the agent's capability in handling user queries and providing transparent explanations.

Finally, aligning with open science principles and FAIR (Findable, Accessible, Interoperable, and Reusable) data practices as outlined in the Data Management Plan (D1.2 and D1.4), THEMIS 5.0 datasets are anonymized, comprehensively documented—including metadata on provenance, representativeness, and bias mitigation strategies—and will be released under open licenses. This approach ensures reproducibility and fosters broader research collaboration, advancing the project's vision of democratizing access to trustworthy, human-centred AI tools tailored to support Europe's digital transition.

4.3.1. Data Sources Relevant to THEMIS Trustworthiness Models (Fairness, Accuracy, Robustness)

Ensuring trustworthiness in AI involves addressing fairness, accuracy, and robustness in the underlying data. For the THEMIS project, our goal is to curate datasets that capture real-world complexity, adhere to ethical and regulatory standards, and support the varied needs of our partners. Below, we outline the main considerations and processes that shape how we select and prepare data for THEMIS 5.0 models.

The THEMIS consortium recognizes that high-quality data is fundamental to building AI systems that are fair, precise, and durable under real-world conditions. Our curation efforts go beyond gathering large volumes of information: we strive to represent the full diversity and complexity of human experiences. By doing so, we aim to reduce biases, ensure accuracy across different contexts, and bolster system resilience in the face of diverse, challenging inputs.

Data selection for THEMIS involves close collaboration between domain experts, ethical advisors, and technical teams. This iterative and multi-layered process guarantees that we capture every facet of trustworthiness. Domain experts help identify critical attributes and edge cases within healthcare, port management, or disinformation contexts. Ethical advisors ensure transparency, legal compliance (e.g., GDPR), and respect for data subjects. Technical teams then translate these insights into actionable requirements, focusing on the diversity and completeness needed to train and test robust AI models.

Another relevant aspect is the fairness aspect, which in AI starts with representative datasets. We actively seek inputs that span various demographics, socioeconomic levels, and geographic regions to minimize inherent biases. This includes demographic surveys, national and international socioeconomic data repositories, clinical records that capture diverse patient profiles, and historical media archives covering a broad range of narratives. By ensuring this inclusivity, we aim to develop models that treat all individuals and groups equitably, mitigating issues like algorithmic discrimination or underrepresentation.

Moreover, the curation process is designed to ensure high-quality annotations and reliable labels form the bedrock of accurate models. For THEMIS, we source data meticulously verified by domain experts, such as clinically validated patient records, port traffic logs cross-referenced by logistics authorities, or fact-checked media content gathered by reputable news organizations. These rigorous standards allow us to train and

evaluate models with confidence that the insights generated are both precise and meaningful. Accurate data ensures that AI outputs align closely with real-world observations and expert judgment.

Finally, compliance with legal, ethical, and technical standards is a core principle for THEMIS. We adhere strictly to the General Data Protection Regulation (GDPR) for all personal data. As we refine our trustworthiness models, we also align our data practices with the emerging guidance of the EU AI Act, ensuring the responsible, transparent, and secure handling of information. This commitment underpins every step of our data curation, from collection and processing to eventual archival or deletion. Ultimately, the creation of trustworthy AI in THEMIS 5.0 stems from an interdisciplinary approach that blends technical rigor with ethical responsibility. By uniting data scientists, legal advisors, industry experts, and researchers, our consortium crafts datasets that are fit for purpose, responsibly sourced, and attuned to the lived realities of diverse populations. This synergy ensures our efforts remain forward-looking and meaningful, fostering AI systems that can be genuinely trusted to uphold fairness, maintain accuracy, and demonstrate resilience in the face of evolving challenges.

4.3.2. Criteria for Dataset Updates Based on Performance Feedback & Assessments

In THEMIS, data is never static. We recognize that even the highest-quality initial datasets can become outdated or limited as real-world conditions shift and user demands evolve. This is why dynamic dataset management is a core principle in our co-creation approach. Iterative feedback loops—integrating user observations, domain experts' insights, and in-depth error analyses—guide timely refinements to the data. This constant re-evaluation safeguards trustworthiness across multiple fronts, ensuring that the Super Decision Engine continues to balance efficiency and equity, the Conversational Agent remains transparent and responsive to user queries, and domain-specific prediction models stay relevant in rapidly changing environment.

The datasets are updated based on:

- Model performance metrics: Performance monitoring is the first checkpoint for deciding whether and when to update datasets. Each THEMIS model is assessed using metrics tailored to its trustworthiness goals:
  - **Fairness Metrics**: For models like the **Persona Predictor** or **Fairness Optimization Model**, we track demographic parity gaps and equalized odds. Rising disparities—such as increased error rates for certain demographic groups—signal the need for more or differently distributed training examples.
  - **Accuracy Metrics**: Models like the **Domain-Specific Prediction Models** and **Explainability Enhancement Models** rely on precision-recall scores, F1 scores, or other domain-appropriate measures. When performance dips below target thresholds, an infusion of fresh or more representative data may help recalibrate their accuracy.
  - **Robustness Indicators**: For the **Robustness Evaluation Model**, we measure resilience under adversarial or noisy conditions. If stress tests reveal vulnerabilities—such as susceptibility to new types of disinformation in media sources—this signals the need for dataset updates that include these evolving real-world edge cases.

Substantial deviations in these metrics trigger a structured response, where data engineers, domain experts, and ethical advisors collaborate to determine the scope and nature of additional data requirements. This risk assessment framework incorporates human input facilitated through IBM's conversational agent.

- Error Analysis and User Feedback: Deployment in real-world scenarios can expose errors or biases not evident in controlled testing environments. For instance, the **Conversational Agent** might struggle with specific dialects, while the **Context-Aware Risk Assessor** might miss certain emergent risk factors. End-user feedback is crucial in identifying such blind spots. Through user reports, surveys, and interactions with the platform, we gather qualitative insights into where and how our datasets may be lacking. This continuous loop of user-centric feedback helps keep the THEMIS platform aligned with the evolving needs of healthcare providers, port authorities, media organizations, and other stakeholders.

- Contextual Relevance: AI does not exist in a vacuum. Shifts in regulatory guidelines (e.g., updates to the EU AI Act), socio-cultural trends, or operational changes in healthcare or logistics may all require data that reflects the new context. For example, if regulations impose stricter patient data standards, the **Healthcare Prediction Models** might need updated records that comply with these rules. Similarly, if an emerging disinformation tactic appears online, the **Media Disinformation Models** require new or different training data to capture these latest threats. Continually assessing such external factors ensures that our datasets stay both compliant and contextually meaningful.

4.4. Technical Infrastructure & Integration

The technical infrastructure of THEMIS 5.0 underpins every aspect of its trustworthiness framework. In this section, we present how the platform's components are structured, deployed, and integrated to support continuous data co-creation, model training, and benchmarking. By leveraging containerization, CI/CD pipelines, and well-defined APIs, the ICE/i4RI environment enables rapid iteration while maintaining high standards of scalability, security, and reliability. Additionally, we discuss how these technical elements interact with the TAIME subsystem, ensuring that trustworthiness assessments and improvements can be seamlessly implemented across diverse use cases.

4.4.1. ICE/i4RI Environment Overview (Containerization, CI/CD, APIs)

The **ICE/i4RI** environment forms the operational backbone for the **THEMIS 5.0** platform, ensuring that critical services—such as the **Dataset Co-Creation Microservice**, the **Training Backend**, and the **Storage Container**—work together seamlessly. Below is an overview of how containerization, continuous integration/continuous delivery (CI/CD), and APIs come together to create a stable, scalable, and human-friendly foundation for trustworthiness assessments.

ICE/i4RI relies on container technologies (e.g., Docker) to package each microservice with its dependencies. This approach makes deployments consistent across different environments and reduces potential conflicts caused by varying software libraries or configurations. Containers can be dynamically scaled to handle fluctuations in usage—an especially important feature when training large AI models or processing high volumes of data. By giving each microservice its own container, ICE/i4RI ensures that updates and bug fixes do not spill over and impact other parts of the platform.

Moreover, a CI/CD pipeline is put into place so that every time developers working on THEMIS 5.0 commit changes, an automated build process is triggered. This immediate feedback allows teams to catch errors or compatibility issues early, improving overall code quality. Once code is validated through automated testing, it seamlessly flows to pre-production environments for final verification. If everything checks out, the CI/CD pipeline can push these changes into production with minimal downtime. This approach is crucial for supporting the rapid iteration cycles required by trustworthiness optimization, where data sources and fairness metrics might evolve quickly.

All services within the ICE/i4RI environment expose RESTful APIs implemented by means of FastAPI, i.e. Python framework, following consistent naming conventions and documentation standards. This design choice makes it straightforward for components like the **Trustworthiness & Profiling Optimizer (TPO)** or the **Persona Predictor** to communicate with the **Storage Container** or the **Dataset Co-Creation Microservice**. ICE/i4RI offers user-facing endpoints that let **Human** stakeholders interact with the platform. For instance, an end-user can upload annotated data.

4.4.2. Integration Points with TAIME Models for Retraining

THEMIS 5.0 is designed as a human-centred trustworthiness optimization platform that evaluates and enhances the performance of SUTs in line with fairness, accuracy, and robustness. The co-created datasets in THEMIS are not used to train the SUTs themselves but rather to improve THEMIS's proprietary models. These models assess and optimize the trustworthiness of the SUTs.

Key components include the TAIME subsystem, which continuously monitors trustworthiness metrics for SUTs. If significant issues are detected (e.g., a dip in fairness or robustness), THEMIS triggers an analysis pipeline that provides actionable recommendations to address these concerns, such as mitigating biases or enhancing interpretability. These recommendations are informed by the TPO, leveraging both domain-specific insights and datasets co-created with stakeholders across diverse sectors.

The results are conveyed through transparent feedback loops, ensuring all actions comply with ethical, legal, and privacy standards. By guiding improvements without directly modifying the SUT, THEMIS fosters an ecosystem that supports responsible, adaptive, and context-aware AI implementations across healthcare, port management, and media use cases.

4.4.3. Dataset Versioning, Documentation, and Management Practices

4.4.3.1 Dataset Versioning

Dataset versioning is critical to maintaining the integrity and reproducibility of AI models and processes. It allows for tracking changes, ensuring backward compatibility, and enabling collaboration across distributed teams. The following principles define the versioning strategy:

- **Semantic Versioning**: Datasets will follow a semantic versioning scheme (e.g., v1.0.0), where:
  - The first digit indicates major changes (e.g., schema alterations).
  - The second digit reflects minor updates (e.g., addition of new data points or minor attribute changes).
  - The third digit signifies patch-level fixes (e.g., correcting mislabelled data or minor metadata adjustments).
- **Version Control Tools**: Datasets will be managed using version control systems (e.g., Git, DVC) to track changes, maintain commit histories, and ensure seamless collaboration among contributors.
- **Branching for Development**: Separate branches will be created for ongoing dataset modifications, enabling iterative improvements without disrupting the integrity of the primary dataset.

4.4.3.2 Dataset Documentation

Comprehensive and clear documentation is fundamental to ensuring datasets are interpretable, reusable, and trustworthy. Documentation will follow a standardized format, including the following key elements:

- **Dataset Description**: An overview of the dataset, including its purpose, scope, and context of use within the THEMIS 5.0 project. Properly explaining the scope within the specific model to be trained.
- **Schema Definition**: Detailed information about dataset attributes, including data types, formats, units of measurement, and constraints.
- **Provenance Information**: A log of the dataset's origin, including sources, collection methods, preprocessing steps, and any external data integration.
- **Version History**: A changelog capturing all modifications, including timestamps, contributors, and descriptions of changes.
- **Quality Metrics**: Information on dataset quality, such as completeness, consistency, and coverage, along with details of validation processes.
- **Ethical Considerations**: Documentation on how the dataset addresses privacy, fairness, and bias mitigation, in alignment with the THEMIS 5.0 ethical guidelines.
- **Usage Guidelines**: Instructions on accessing, using, and contributing to the dataset, including any licensing and compliance requirements.

All documentation will be stored in a centralized repository (e.g., hosted on GitHub or an internal platform), with accessibility based on role-specific permissions.

4.4.3.3 Dataset Management Practices

To ensure effective dataset management, a series of practices are planned to be implemented, although these are subjected to technical as well as contractual limitations:

- **Centralized Data Repository**: Datasets will be stored in a centralized, cloud-based repository that supports scalable access, version control, and secure collaboration.
- **Data Governance Framework**: A governance framework will define roles and responsibilities for dataset contributors, reviewers, and approvers to maintain accountability and compliance.
- **Data Access Controls**: Role-based access control (RBAC) mechanisms will be enforced to protect sensitive data and ensure ethical usage.
- **Automated Pipelines**: Data ingestion, preprocessing, and versioning workflows will be automated using tools like CI/CD pipelines to minimize manual errors and ensure consistency.
- **Quality Assurance Checks**: Regular quality audits will be conducted to verify dataset integrity, with predefined metrics for completeness, accuracy, and consistency.
- **Traceability and Auditability**: Metadata will be embedded within datasets to ensure traceability and auditability, providing a clear record of data lineage and transformation processes.
- **Stakeholder Collaboration**: Stakeholders will have access to a feedback loop to suggest improvements, report issues, and participate in iterative enhancements.

4.4.3.4 Integration with THEMIS 5.0 Ecosystem

The dataset versioning, documentation, and management practices will seamlessly integrate with the THEMIS 5.0 trustworthiness optimization processes:

- **Trustworthiness Metrics**: Datasets will include predefined metrics for assessing fairness, robustness, and accuracy during model development.
- **Explainability Integration**: Metadata will support explainability tools within THEMIS 5.0, enabling users to understand how datasets contribute to AI decisions.
- **Dynamic Updates**: Mechanisms will be implemented to incorporate feedback and new data into existing datasets without disrupting ongoing workflows.

4.5. Trustworthiness Metrics & Retraining Triggers

In Task 4.3, the co-creation of benchmarked data sets and the subsequent training of THEMIS 5.0 AI models rely on continuous feedback loops to maintain and improve the SUT's models trustworthiness. To manage such feedback effectively, clear Key Performance Indicators (KPIs) and thresholds are established. These KPIs reflect the technical accuracy, robustness, and fairness targets identified in the THEMIS 5.0 risk-based approach, while also linking to human-centred constraints (e.g., domain experts' acceptance criteria, business-level benchmarks, and legal or ethical guidelines).

4.5.1. KPIs and Thresholds Defined by ICCS/SINTEF/IBM

In THEMIS 5.0, KPIs and their corresponding thresholds act as pivotal benchmarks for monitoring and maintaining the trustworthiness of AI models across domains such as healthcare, port management, and media disinformation. While quantitative metrics provide essential insights into performance, it is equally important to recognize that trustworthiness cannot be fully reduced to numbers alone. Certain aspects—such as user perceptions, ethical considerations, or shifting social contexts—must also be captured through qualitative evaluations and natural language feedback, potentially facilitated by LLMs. By combining rigorous quantitative KPIs with more nuanced evaluative methods, THEMIS 5.0 can holistically assess and refine AI systems for fairness, accuracy, robustness, and explainability.

Accuracy and Robustness Metrics

Accuracy-Related KPIs: These may include Precision, Recall, F1 score, AUC-ROC, or domain-specific metrics (e.g., Mean Absolute Error in port ETA predictions; sensitivity/specificity in a health diagnosis model). Thresholds might be established through domain co-creation (e.g., 85% minimum precision for a classification task).

Robustness-Related KPIs: Tolerances to adversarial noise or out-of-distribution data are measured. For instance, the acceptable drop in performance under minor input perturbations (e.g., <5% reduction in F1 score after a shift in data distribution).

Environmental or Socio-Technical KPIs: Some SUTs (Systems Under Test) may require real-time reliability (e.g., max permissible failure rates in mission-critical port management tasks).

Fairness Metrics

To safeguard equitable performance across demographic groups or other subgroups, we may track metrics such as Demographic Parity, Equalized Odds, or Disparate Impact Ratios. Trigger points occur if any measured disparity exceeds a pre-agreed threshold (e.g., 80% rule for predictive parity).

Explainability and Transparency Indicators

While not always expressed as numeric KPIs, user feedback on system interpretability (collected in co-creation sessions) can form acceptance thresholds. For instance, if less than a certain percentage (e.g., 70%) of domain experts find the model's explanations understandable, re-check of the model or the data is triggered.

Compliance or Ethical Indicators

According to relevant ethical guidelines and legal frameworks (e.g., AIA or sector-specific regulations), certain model behaviours or performance levels must be upheld. A shortfall in these areas—detected through risk analysis or audits—can trigger refinement.

Examples of Thresholds

- *Accuracy Threshold:* 90% accuracy on real-time port traffic predictions.
- *Fairness Threshold:* No subgroup's false-negative rate is more than 1.5 times that of others.
- *Robustness Threshold:* <5% average performance degradation under mild distributional shifts.
- *Explainability Threshold:* Minimum 70% approval rating by co-creation participants in interpretability surveys.

Collectively, these KPIs and their thresholds ensure that the data curation and training processes remain aligned with THEMIS 5.0's trustworthiness objectives, providing a systematic way to detect when models must be retrained or data sets refined.

4.5.2. Automated Dataset Refinement or Retraining Triggers

While Task 4.3 focuses on curating data sets and ensuring that Themis platform models remain high-quality, the operational environment of the SUTs can change in ways that degrade model performance. Therefore, automation in detecting dataset drift, model underperformance, or new fairness issues is critical. The triggers described below help orchestrate timely and systematic retraining or refinement workflows.

Trigger Categories

1. **Performance Deviation Triggers**
    a. Continuously monitor the live model performance (e.g., via rolling window metrics). If the performance metric (e.g., F1) falls below a threshold for a sustained period or experiences a significant sudden drop, the system flags a retraining event.
    b. *Example:* For a port management model predicting vessel arrival times, if Mean Absolute Error increases by more than X% compared to the last validated baseline, an automated alert is generated.

2. **Drift Detection Triggers**
    a. *Data Drift:* If distributional changes in input features deviate beyond a set statistical distance (e.g., Jensen-Shannon divergence or Population Stability Index) from the training set distribution, an automated dataset update is triggered.
    b. *Concept Drift:* If the underlying relationships between inputs and outputs have changed (e.g., new maritime regulations altering shipping patterns), the drift detection module signals the platform to acquire fresh training data from newly collected instances and re-run training.
3. **Fairness or Ethical Violation Triggers**
    a. If real-time monitoring reveals that certain subgroups or variables are systematically disadvantaged beyond the fairness threshold set (e.g., differences in false positive rates among subpopulations), the system triggers a fairness re-check and potential mitigation steps (e.g., rebalancing the training set).
4. **Quality and Coverage Gaps in Dataset**
    a. Automated data profiling checks for missing data, outliers, or coverage gaps. If coverage in critical scenario categories (e.g., high-congestion days in port management) falls below a threshold, an updated data-collection process is auto-initiated.
5. **User-Initiated Alerts**
    a. If domain experts (e.g., doctors in a healthcare SUT or port operators) report suspicious predictions, they can flag anomalies to the THEMIS pipeline. The pipeline merges these flagged cases with standard metrics, potentially raising an immediate "high priority" retraining event.

4.5.3. Incorporating TAI Cards and Assessment Results into Retraining Decisions

A hallmark of the THEMIS 5.0 ecosystem is the structured documentation of AI systems and trustworthiness criteria via TAI Cards (e.g., Model Cards, Data Cards, Risk Knowledge Cards, Method Cards). During the iterative training loops in Task 4.3, these cards play a vital role:

- **Model Cards for Retraining History:** Each iteration of model training or fine-tuning updates the "Model Card," capturing new hyperparameters, dataset versions, performance metrics, and any fairness or robustness constraints applied. The training pipeline automatically references this card when setting up new runs, ensuring that prior insights (e.g., known edge cases, baseline comparisons) are not lost between iterations.
- **Risk Knowledge Cards for Triggering Retraining:** These cards document known vulnerabilities, relevant threat models, and recommended mitigation strategies. The pipeline can parse these risk indicators to decide, for instance, if the model is showing vulnerability to a known class imbalance or fairness shortcoming. If the risk knowledge card points out a recommended corrective step—such as "apply data balancing" or "train an adversarial model for out-of-distribution detection"—the pipeline executes these methods in the next retraining cycle.
- **Data Cards for Continuous Refinement:** The dataset and its various transformations are documented in "Data Cards." When user feedback identifies mislabelled entries or coverage gaps, the Data Cards record these updates. The pipeline consults the Data Cards to verify that the correct subsets or new data segments are included before retraining.
- **Method Cards for Evolving Techniques:** The relevant fairness or robustness "Method Cards" list the available interventions (e.g., rebalancing approach for fairness, data augmentation for robustness). As triggers arise, the pipeline checks these methods to decide whether a new technique—like domain adaptation or adversarial training—should be applied to meet updated thresholds.
- **Human-Centric Oversight:** Even though many triggers are automated, human reviewers—such as domain experts, data scientists, or risk officers—can intervene. The TAI Cards serve as an audit trail so that reviewers can confirm or override an automated decision. If a recommended approach is untested or fails to restore metrics above threshold, reviewers can revert to a previous model or dataset version—each documented in the relevant card.

4.6. Ensuring Fairness, Non-Discrimination, and Legal Compliance

Ensuring compliance with ethical guidelines, legal mandates, and robust security protocols is paramount to building trust in AI-driven systems. This section outlines the core principles and measures that THEMIS 5.0 adopts to protect individual rights, uphold legal standards (such as GDPR and the future EU AI Act), and safeguard morally sensitive data. We describe the governance frameworks, privacy-by-design strategies, and oversight mechanisms that guide data handling and model development, highlighting the pivotal role of continuous risk assessment and stakeholder engagement in preserving AI trustworthiness.

4.6.1. Adherence to GDPR and Additional Compliance Requirements

Compliance with the GDPR and the AI Act in THEMIS 5.0 is achieved through a comprehensive integration of legal, ethical, and technical safeguards that underpin the platform's AI model training and benchmarking processes, particularly within Task 4.3. As concerns the GDPR, the platform rigorously applies data protection and privacy-by-design principles by aiming to ensure that all personal data processing operations comply with the principles laid down in Article 5 of the GDPR, such as minimisation and storage limitation, and any other applicable requirements. In this regard, project partners have been informed about the relevant existing guidance on personal data processing in the context of AI models recently issued by the EDPB[32]. When possible, preference is given to the use of fully anonymized datasets, while taking into account the challenges related to anonymisation as detailed in available guidance provided by Working Party 29[33] and the EDPS[34]. Should anonymisation not be practical, any personal data being processed would be subject to pseudonymisation, taking into account the guidelines recently issued by the EDPB on pseudonymisation.[35] Pseudonymisation would facilitate compliance with multiple legal requirements of the GDPR, for instance as an appropriate technical and organisational measure to achieve the requisite security of data processing.

In addition, THEMIS 5.0 ensures transparency by clearly informing data subjects about the nature and purpose of data collection and processing, outlining their rights—including access, rectification, and erasure as stipulated in Articles 15–17 GDPR—and by establishing a lawful basis for processing either through explicit consent under Article 6(1)(a) or legitimate research interests under Article 6(1)(f). To maintain data security and integrity, the system implements robust encryption protocols for data at rest and in transit, fine-grained access controls to prevent unauthorized access, and incident reporting mechanisms that require immediate notification to the designated Data Protection Officer in compliance with GDPR Article 33, along with secure storage and retention policies that mandate the deletion of personal data once it is no longer needed.

Beyond GDPR, THEMIS 5.0 aligns with additional EU and international legal frameworks, including proactive measures to prepare for future compliance with the AI Act by incorporating risk assessment frameworks that identify AI-related risks early in development, and by conducting continuous ethical and legal impact assessments overseen by a dedicated Ethics and Legal Compliance Board. The platform further adopts a privacy-by-design approach with comprehensive audit mechanisms—such as version-controlled audit logs that track all dataset transformations and methods like SHAP to explain AI decisions and identify dataset biases—and implements granular consent systems that allow data subjects to revoke consent at any time via a centralized consent-revoke system compliant with Article 7(3) GDPR. By integrating these GDPR principles and putting in place measures to facilitate compliance with the recently adopted AI Act in the future, THEMIS 5.0 ensures that its AI benchmarking and training processes are legally robust, ethically sound, and privacy-

[32] EDPB, Opinion 28/2024 on certain data protection aspects related to the processing of personal data in the context of AI models, adopted on 17 December 2024.
[33] Article 29 Data Protection Working Party, Opinion 05/2014 on Anonymisation Techniques, 0829/14/EN WP216, 2014.
[34] AEPD-EDPS joint paper on 10 misunderstandings related to anonymisation, 2021.
[35] EDPB, Guidelines 01/2025 on Pseudonymisation, 2025.

conscious, while remaining adaptable to upcoming regulatory changes to support future deployment and scalability.

4.6.2. Strategies for Handling Morally Sensitive Data

Handling morally sensitive data in THEMIS 5.0 requires a structured approach that ensures ethical integrity, legal compliance, and societal trust, particularly given its diverse applications in healthcare, disinformation mitigation, and port management; accordingly, the platform integrates privacy-enhancing technologies, ethical oversight mechanisms, and strict governance frameworks to mitigate risks associated with processing such data. Morally sensitive data, defined as any dataset that could cause harm to individuals, communities, or societal structures if misused or exposed, is categorized in THEMIS 5.0 to include personally identifiable information (PII) such as names, addresses, and biometric data; health and biomedical data derived from medical records, patient diagnostics, or wearable health tracking devices; disinformation and misinformation indicators comprising AI-detected manipulative content with implications for freedom of speech and democracy; bias-prone data sources like socioeconomic data, demographic information, and historical datasets reflecting systemic inequalities; and high-risk decision-making data used in critical systems such as predictive policing, credit scoring, and hiring algorithms, all of which are collected, processed, and utilized under strict ethical guidelines and ongoing risk assessments. To prevent misuse, a multi-layered risk-aware data governance framework is employed that includes tiered data access control—utilizing role-based access management (RBAC) to restrict access to only authorized users, attribute-based access control (ABAC) to impose additional restrictions based on the nature and sensitivity of the dataset, and data partitioning and encryption with multi-factor authentication (MFA) for secure storage—along with ethical data filtering and processing measures such as pre-processing bias mitigation through de-biasing algorithms to ensure equitable representation, automated harm detection via AI-driven anomaly detection models that flag risks like algorithmic discrimination or incomplete representations, and synthetic data generation to replace actual sensitive data with AI-generated datasets that maintain statistical properties while eliminating individual traceability. Continuous risk auditing and explainability are maintained by classifying datasets based on risk levels to trigger stricter controls for high-risk data, integrating explainability tools like SHAP and LIME to ensure transparent and auditable processing, and employing automated ethical review alerts that prompt human-in-the-loop assessments when potential moral concerns arise. Moreover, ethical oversight and stakeholder involvement are ensured through the establishment of a multi-stakeholder Ethical Review Board (ERB) composed of ethicists, legal scholars, AI researchers, and community representatives who review new datasets for biases, societal risks, and legal implications, and conduct post-deployment impact assessments for high-risk AI applications, complemented by ethical AI auditing and redress mechanisms such as biannual ethical impact reports and an independent review process for affected individuals. To further mitigate risks, THEMIS 5.0 integrates state-of-the-art privacy-preserving AI methodologies aligned with GDPR, the AI Act, and ISO/IEC 27001 security standards, employing techniques like federated learning to process data locally without central transfer, differential privacy with noise injection to protect individual identities while enabling meaningful AI training, and zero-knowledge proofs (ZKP) to securely verify data properties without exposing raw content, while also ensuring compliance with legal frameworks through strict adherence to GDPR Articles 5–9, alignment with risk-classification methodologies under the EU AI Act (2024) for high-risk AI systems, and adherence to ISO/IEC 27001 standards for robust cybersecurity. In conclusion, THEMIS 5.0's approach to handling morally sensitive data is guided by privacy-first AI development, ethical governance, and legal compliance, and by integrating automated risk auditing, stakeholder involvement, privacy-enhancing technologies, and robust legal safeguards, the platform ensures that its AI models handle such data responsibly and transparently while maintaining public trust and regulatory compliance.

4.6.3. Privacy-by-Design Principles and Audit Trails for Dataset Usage

Ensuring data protection in AI model training and dataset usage is a foundational requirement for THEMIS 5.0, which is achieved by integrating Privacy-by-Design (PbD) principles and a comprehensive audit trail system that upholds data protection, user rights, and regulatory compliance in alignment with the GDPR. The PbD

framework, as mandated by Article 25 of the GDPR, embeds data protection into the AI system architecture from the outset by enforcing seven fundamental principles: firstly, a proactive and preventative approach is adopted through data protection risk assessments before any dataset processing to identify and mitigate vulnerabilities; secondly, data protection is maintained as the default setting so that users are automatically protected with anonymized or pseudonymized data processing unless explicit consent is provided for identifiable data; thirdly, data protection is embedded into the design, ensuring that security and compliance measures are an integral part of the AI model architecture rather than an afterthought; fourthly, a full functionality approach is taken, applying privacy-preserving AI techniques such as differential privacy and federated learning to maximize performance without compromising user privacy; fifthly, end-to-end security is guaranteed by ensuring secure data handling across the entire AI lifecycle with encryption at rest and in transit; sixthly, visibility and transparency are provided through clear audit logs, explainability tools, and real-time monitoring dashboards that allow stakeholders to verify data usage, access, and processing; and finally, user-centric control and data sovereignty are maintained by offering mechanisms for granular consent, revocation, and access requests in compliance with GDPR Articles 12–22. To further safeguard datasets used for AI model training, THEMIS 5.0 employs a multi-tiered access control framework that combines Role-Based Access Control (RBAC), which restricts dataset access to authorized personnel based on specific categories such as healthcare, media, or port management, with Attribute-Based Access Control (ABAC), which enforces granular security policies by allowing access only when certain conditions, like specific IP addresses or geographic zones, are met. In addition, robust data encryption and anonymization practices are implemented through AES-256 end-to-end encryption for data at rest and in transit, along with pseudonymization and anonymization techniques that replace or remove sensitive personal data using irreversible cryptographic hashing and tokenization. Enhancing these measures, privacy-preserving AI mechanisms such as federated learning enable local training on devices without centralizing raw data, differential privacy injects controlled random noise to prevent individual re-identification, and secure multi-party computation (SMPC) allows data processing without exposing raw dataset contents to any single entity. Complementing these privacy measures, THEMIS 5.0 maintains comprehensive audit trails for dataset usage through an immutable, blockchain-backed system that logs every dataset interaction in a verifiable manner; this includes version-controlled data logging that timestamps and records every dataset modification, detailed user activity logging that captures who accessed the dataset, when, and what actions (such as downloads, modifications, or analyses) were performed, and tamper-resistant storage ensured by cryptographic signing of audit logs. Real-time dataset monitoring is achieved through AI-powered anomaly detection that automatically alerts administrators to unusual activities like bulk data downloads outside standard business hours, supported by user-friendly dashboards that visually track data usage trends and access history. Data usage policy enforcement further guarantees that access requests are evaluated against data minimization standards and subjected to periodic reviews, ensuring that permissions are updated or revoked as necessary. In terms of legal and ethical standards, THEMIS 5.0 complies with GDPR by ensuring lawful data processing under Article 5, maintaining detailed records of processing activities as required by Article 30 (ROPA). By embedding these Privacy-by-Design principles and robust audit trail mechanisms, THEMIS 5.0 maintains data integrity, security, and transparency throughout its operations, thereby reinforcing its commitment to trustworthy AI development and full regulatory compliance.

4.7. Explainability & Transparency

Explainability and transparency are cornerstones of human-centred AI, directly impacting user trust and informed decision-making. In this section, we detail the tools and techniques THEMIS 5.0 employs to make AI behaviours understandable for diverse stakeholders—from domain experts to end-users. We also highlight documentation practices and reporting processes that illuminate how data and models evolve over time, ensuring that the reasoning behind model outputs is accessible, interpretable, and aligned with ethical and regulatory expectations.

4.7.1. *Tools and Techniques (e.g., SHAP) for Explainable Datasets*

Ensuring explainability and transparency in AI datasets is a core objective in THEMIS 5.0, particularly within Task 4.3, which focuses on the co-creation of benchmarked datasets and the training of AI models, as explainable datasets enhance trustworthiness by providing stakeholders with clear insights into how data is structured, processed, and used in training; to achieve this, key tools and techniques such as SHAP, LIME, counterfactual explanations, and feature attribution analysis are employed, with SHAP offering a method based on cooperative game theory to quantify feature importance, detect biases, and enhance transparency through human-readable explanations, while LIME assesses dataset transformations by generating perturbed versions to illustrate how feature changes influence model behaviour and bolster model trustworthiness, and counterfactual explanations generate alternative examples to validate fairness and demonstrate the impact of slight data changes on outcomes, further complemented by feature attribution techniques like Integrated Gradients that identify influential dataset features and evaluate robustness by highlighting elements that disproportionately affect decision-making; in addition to these methods, THEMIS 5.0 incorporates documentation techniques such as dataset versioning and maintaining detailed logs of updates and retraining triggers, automated explainability reports that outline dataset biases, key attributes, and the rationale behind retraining decisions, and audit trails that ensure accountability in data-driven decision-making, while reporting processes for stakeholders are enhanced through interactive dashboards with feature importance visualizations, automated fairness audits using both SHAP and counterfactual explanations, and real-time monitoring for continuous dataset assessment, ultimately integrating these approaches to not only improve AI model performance but also to align with ethical and legal requirements, thereby ensuring robust dataset explainability in critical AI-driven decision-making and contributing to the THEMIS 5.0 Trustworthiness Optimization Framework and human-centred AI transparency efforts.

4.7.2. *Methods for Documenting Decisions and Data Transformations*

Effective documentation of decisions and data transformations is essential for ensuring transparency, traceability, and accountability in AI systems, and in THEMIS 5.0, several structured methodologies are employed to meticulously record each stage of the data lifecycle, from collection to processing and analysis; Therefore, by detailing its origins and transformations, enabling stakeholders to understand the transformations it has undergone, and assess its impact on the final output, while data cards provide structured summaries containing essential information about datasets—including their origins, composition, intended use, and ethical considerations—that facilitate a comprehensive understanding of the dataset's lifecycle and support responsible AI development; moreover, model cards offer detailed documentation of AI models by outlining their development process, performance metrics, intended applications, and potential limitations, which aids in assessing the suitability of a model for specific tasks and ensures that stakeholders are aware of any constraints or ethical considerations associated with its deployment, and dynamic documentation continuously and automatically records data processes and decisions as they occur within the system, ensuring that documentation remains current with the latest system changes and provides a real-time overview of data workflows that enhances the system’s adaptability and responsiveness; additionally, AI traceability maintains a detailed record of how AI models are trained and how they process information to make decisions by documenting the data sources, feature selection, model parameters, and decision-making processes, thereby enabling thorough scrutiny and understanding of the AI system's operations, and by integrating these methodologies, THEMIS 5.0 ensures a robust framework for documenting decisions and data transformations that promotes transparency, accountability, and trustworthiness in its AI systems.

4.7.3. *Reporting Processes for Stakeholders to Understand Dataset Rationale*

Effectively communicating the rationale behind datasets is crucial for fostering stakeholder trust and ensuring informed decision-making within THEMIS 5.0, and by implementing structured reporting processes, stakeholders gain clear insights into dataset origins, transformations, and applications. To enhance transparency and understanding, methodologies such as Data Cards, which serve as structured summaries

offering essential information about datasets—including their origins, composition, intended use, and ethical considerations—are employed to facilitate a comprehensive understanding of the dataset's lifecycle and support responsible AI development. Additionally, interactive dashboards enable stakeholders to dynamically engage with data visualizations by providing real-time insights into dataset metrics, distributions, and transformations, while interactive elements like filters and drill-down capabilities empower users to customize their data exploration experience. Regular stakeholder workshops further foster direct communication between data practitioners and stakeholders, serving as platforms to present dataset rationales, discuss data collection methodologies, and address queries, thus promoting transparency and collaborative refinement of data practices. Comprehensive data documentation is maintained to detail data sources, collection methods, preprocessing steps, and transformation logic, offering stakeholders a clear narrative of the dataset's journey from raw data to its current state and ensuring traceability of each processing step. Finally, executive summaries distil complex data processes into concise narratives that highlight key aspects such as the dataset's purpose, scope, and significant transformations or decisions, ensuring that all stakeholders, regardless of technical expertise, can grasp the essential elements of the dataset rationale. By integrating these reporting processes, THEMIS 5.0 ensures that stakeholders are well-informed about dataset rationales, thereby fostering transparency, trust, and collaborative engagement in AI system development.

4.8. Quality Assurance & Validation

Data quality and reliable model performance are integral to achieving and maintaining AI trustworthiness. This section examines the strategies and mechanisms used within THEMIS 5.0 to validate data, detect biases, and analyse errors. By establishing robust monitoring frameworks and feedback loops with domain experts, the project aims to detect emergent issues early and refine datasets iteratively. Through these quality assurance measures, THEMIS 5.0 upholds high standards of fairness, accuracy, and resilience across all integrated AI services.

4.8.1. Data Validation Checks, Bias Detection, and Error Analysis

Ensuring the quality, fairness, and reliability of data used in the THEMIS 5.0 AI models is paramount for the trustworthiness and robustness of the system. This subsection outlines the methodologies applied for **data validation, bias detection, and error analysis** within the benchmarked datasets utilized for training THEMIS 5.0 AI models.

4.8.1.1 Data Validation Checks

The **integrity and consistency** of datasets are verified using an automated validation pipeline. This process includes:

- **Schema Validation:** Ensuring the structure of incoming data conforms to predefined schemas, including type constraints, missing values handling, and format compliance.
- **Statistical Consistency Checks:** Evaluating distributions of numerical and categorical variables to detect anomalies such as unexpected deviations from historical trends.
- **Duplication Detection:** Identifying and mitigating duplicate records that could distort model learning.
- **Data Drift Analysis:** Comparing the distribution of the training dataset with incoming real-world data to assess the representativeness of the dataset over time

4.8.1.2 *Bias Detection*

Bias can be introduced at any stage of an AI system’s lifecycle, including data collection, feature engineering, and model training. The THEMIS 5.0 framework applies **multi-faceted bias detection techniques** to mitigate systemic, computational, and human-induced biases. Key components of bias detection include:

- **Group Fairness Metrics:** Evaluating **demographic parity**, **equalized odds**, and **disparate impact** to identify disparities across protected attributes such as gender, ethnicity, or socioeconomic status.
- **Individual Fairness Analysis:** Utilizing **counterfactual fairness** methods to assess whether individuals with similar attributes receive consistent predictions.
- **Causal Bias Detection:** Leveraging causal inference models to identify underlying factors that contribute to biased decision-making.
- **Algorithmic Transparency Reports:** Generating explainability reports that highlight potential sources of bias and enable human review

Regarding *Bias Mitigation Strategies,* THEMIS 5.0 applies them based on resource constraints, domain requirements, and stakeholder feedback. Depending on the scenario, **pre-processing** methods like re-weighting or re-sampling may be used to address known class imbalances if sufficient data and labeling resources are available. **In-processing** approaches, such as injecting fairness constraints during model training, are considered when performance metrics and domain expectations remain consistent with these adjustments. Finally, **post-processing** steps—like calibrating model outputs to better align with fairness objectives—are adopted if residual biases persist. Each mitigation strategy is thus selected and tailored in collaboration with domain experts and technical teams, ensuring that the effort invested is proportionate, targeted, and aligned with the practical realities of the THEMIS 5.0 ecosystem.

4.8.1.3 Error Analysis

Robust error analysis is essential for identifying **systematic failures** and **edge cases** that may affect model reliability. THEMIS 5.0 incorporates the following approaches:

- **Misclassification Detection:** Utilizing confidence-based rejection mechanisms where predictions with high uncertainty are flagged for human review
- **Adversarial Robustness Testing:** Evaluating model responses to adversarial perturbations to assess stability under **out-of-distribution** conditions
- **Explainability-Aware Debugging:** Applying **SHAP (Shapley Additive Explanations)** and **LIME (Local Interpretable Model-agnostic Explanations)** to investigate prediction errors and improve interpretability
- **Human-in-the-Loop Evaluation:** Integrating domain experts in the error analysis loop to provide qualitative assessments and guide corrective actions

4.8.2. *Continuous Monitoring and Feedback*

To maintain long-term data quality and fairness, THEMIS 5.0 implements **automated monitoring pipelines** that:

- Continuously assess **model performance and fairness drift** using feedback loops.
- Trigger **automated dataset refinements** or **retraining** when KPIs fall below predefined thresholds
- Log detected anomalies and errors for ongoing assessment in **risk-aware AI decision systems**

4.8.2.1 Continuous Monitoring, Logging, and Reporting on Dataset Quality

To ensure that the co-created benchmark datasets remain reliable and representative of real-world scenarios, THEMIS 5.0 employs a continuous monitoring, logging, and reporting framework. This framework tracks dataset integrity, completeness, balance, and relevance, allowing prompt detection of dataset anomalies or drift.

1. **Monitoring Framework**
    a. **Live vs. Static Data**: Even though the co-created datasets may originate from human-centred and domain expert inputs, some SUT domains require periodic or continuous updates (e.g.,

real-time feed of shipping logs, evolving medical records). A monitoring service flags changes to dataset distributions (e.g., changes in class distribution or new feature ranges).

b. **Quality Metrics**: Key metrics include missing-value counts, average label confidence (for labelled data), ratio of minority to majority classes (for fairness analysis), and outlier detection. Any deviation beyond defined thresholds automatically raises alerts.

2. **Logging Practices**
   a. **Dataset Versions**: Each dataset update or augmentation is stored in a version-control system, along with time stamps, user IDs, and a summary of changes (e.g., new data for specific subpopulations, removed erroneous entries).
   b. **Metadata Capture**: Data lineage and provenance are logged, including original source information, transformations (e.g., normalization, feature engineering), and reasons behind these transformations (e.g., fairness enhancement).
   c. **Anomaly Reports**: If anomalies are detected (e.g., unexpected surge in missing fields), a structured incident log is produced. This log becomes input to potential data refinement or re-annotation efforts.
3. **Reporting Pipeline**
   a. **Scheduled Reports**: Regular summary reports outline the dataset's health indicators (e.g., distribution drifts, mislabelled data) and are shared with relevant stakeholders (e.g., data scientists, domain experts, WP4 developers).
   b. **Real-time Dashboards**: In higher criticality contexts, real-time dashboards provide immediate visibility into data ingestion and labelling performance.
   c. **Escalation Workflow**: When crucial data-quality issues emerge (e.g., a domain-specific concept not represented in the dataset), a pre-defined escalation workflow involves domain leads and the TAUPOS components to decide on potential re-annotation or model retraining.

Through these continuous monitoring and reporting processes, THEMIS 5.0 ensures that the co-created datasets retain validity, supporting robust model training and consistent alignment with trustworthiness metrics over time.

4.8.2.2 Feedback Loops with Domain Experts and TAUPOS

The co-creation of datasets in THEMIS 5.0 requires more than static, one-off data contributions. Robust feedback loops involving domain experts, end-users, and the TAUPOS are central to maintaining dataset relevance and improving trustworthiness.

1. **Domain Expert Interventions**
   a. **Periodic Review Sessions**: Experts (e.g., healthcare professionals, port management planners, content moderators) periodically review subsets of the dataset for accuracy, representation, and contextual correctness. These sessions reveal new domain requirements or highlight emergent edge cases.
   b. **Annotation Refinements**: Based on experts' review, certain data samples may need re-labelling. For instance, in disinformation detection tasks, evolving tactics might render older category definitions obsolete. Domain experts refine labels, accordingly, feeding improvements back into the dataset pipeline.
2. **Integration with TAUPOS**
   a. **Trustworthiness Triggers**: TAUPOS components (e.g., risk assessment, fairness monitors) can automatically alert domain experts if dataset fairness or accuracy metrics degrade. For example, if a certain demographic class becomes underrepresented, TAUPOS suggests collecting or annotating more samples.
   b. **Proposed Mitigation Methods**: TAUPOS might also recommend specific rebalancing or augmentation methods (e.g., synthetic oversampling) if domain experts confirm a documented gap or bias.

c. **Collaborative Decision-Making**: Any recommended dataset updates, quality-control steps, or model retraining events are presented to both technical leads and expert users. This ensures that final decisions reflect real-world constraints and prioritize trustworthiness attributes identified in WP2 (accuracy, robustness, fairness).

3. **Iterative Co-Creation Workflows**
   a. **Living Lab Approach**: As part of WP3's living labs, new or ambiguous data points can be flagged by participants. The feedback loop encourages iterative refinement, validating that these data truly reflect real phenomena or emergent categories in the domain.
   b. **User-Centric Dashboards**: A specialized user interface displays dataset summaries and trustworthiness indicators in intuitive forms (e.g., fairness metrics for each subgroup), allowing non-technical experts to suggest where data coverage is lacking or to confirm domain correctness.
   c. **Knowledge Base Updates**: If repeated domain feedback surfaces the same gaps, a knowledge repository records these as "recurrent issues." Over time, these issues become part of the system's risk knowledge, helping the platform proactively detect and address them.

By leveraging tight feedback loops with domain experts and integrating them with TAUPOS intelligence, THEMIS 5.0 ensures continuous alignment of co-created datasets with real-world conditions. This stakeholder-driven approach strengthens trustworthiness, maintains dataset relevance, and fosters ongoing improvement.

4.9. Roadmap for Iterations & Future Work

We have outlined a holistic approach to dataset quality assurance, co-creation strategies, and the associated triggers for retraining and refinement. By integrating domain expert feedback, automated data monitoring, and the use of TAI Cards, this framework provides the foundation for continuous dataset improvement. Moreover, explicit trustworthiness thresholds—covering accuracy, fairness, robustness, and socio-technical considerations—underscore the project's commitment to responsible AI practices.

4.9.1. Mechanisms for Iterative Improvement as Trustworthiness Criteria Evolve

Building on the current methodology, we plan iterative cycles of data expansion and model refinement. Each new cycle will:

- Re-evaluate domain requirements and data distribution with respect to evolving real-world conditions.
- Update TAI Cards and domain-specific acceptance criteria with newly uncovered user needs or compliance mandates.
- Identify additional data sources or underrepresented classes to enrich the benchmark datasets.
- Validate revised datasets through domain-expert workshops and live-lab pilots, ensuring alignment with SUT objectives.

Within these iterations, cross-functional collaboration—particularly between WP3 co-creation activities, WP4 model training, and WP5 pilot feedback—remains essential for capturing emergent use cases and trustworthiness needs that might otherwise be overlooked.

4.9.2. Scalability Considerations and Long-term Maintenance Plans

As THEMIS 5.0 scales across multiple use cases or regions, data volume and complexity will increase:

- **Infrastructure:** We plan to adopt cloud-based data pipelines that support efficient storage, versioning, and retrieval of increasingly large datasets.

- **Automated Quality Checks:** Automated data validation scripts will run in real time or as part of continuous integration (CI) workflows, detecting and reporting anomalies at scale.
- **Modular Architecture:** By designing microservices or containerized workflows, the solution can be readily extended to additional SUTs with minimal re-engineering.
- **Lifecycle Documentation:** Each dataset update will be accompanied by thorough logging and TAI Card updates, maintaining a comprehensive historical record of changes that supports future audits and compliance checks.

4.9.3. Engaging Stakeholders in Ongoing Dataset Refinement

Ultimately, the success of any dataset-driven solution depends on continuous stakeholder engagement:

- **Scheduled Feedback Cycles:** Domain experts, data scientists, and end-users will jointly review performance reports and co-creation outcomes, ensuring data remain valid for real-world conditions.
- **User-Centric Tools:** Dashboards or collaborative annotation platforms will be improved to lower barriers for non-technical experts, further democratizing data refinement efforts.
- **Community and Ecosystem Partnerships:** In cases where broader communities (e.g., healthcare practitioners, port management authorities) co-own data, we will foster partnerships to ensure that local knowledge and norms shape our evolving datasets.

In sum, we envision an agile approach that continuously strengthens co-created datasets and ensures the trustworthiness of the models trained on them. By iterating in close collaboration with domain experts and end-users, THEMIS 5.0 can remain responsive to shifting conditions and evolving criteria for trustworthy AI, ultimately delivering solutions that align with ethical standards, technical robustness, and real-world utility.

5. DEVELOPMENT OF THEMIS 5.0 ECOSYSTEM PLATFORM

The THEMIS 5.0 platform, developed within Work Package 4, will serve as a comprehensive environment for the assessment of both already developed and to be designed AI systems. Additionally, key platform capabilities include efficient cloud-edge resource management through Kubernetes, along with robust dataset and model handling using frameworks like MinIO, Redis, and Hive. This section will explore two core aspects of THEMIS 5.0: (1) the SUT Common interface, designed to facilitate rapid prototyping of AI models from scratch and built using open-source technologies, leveraging industry-leading machine learning frameworks like TensorFlow, SparkML, and FlinkML, and (2) the platform's cloud-edge resource management capabilities, with a focus on Kubernetes-based orchestration via K3S, that offer an intuitive, scalable infrastructure to support AI processing.

5.1. SUT Common: quick usage of its features for fast prototyping of AI models from scratch

This section outlines how the developed components can be utilized to address the point (1) above, by leveraging a workflow management environment for the application of machine learning and data analytics techniques across cloud and edge infrastructures.

5.1.1. Workflow Management Overview

The described component is designed to be adaptable across various applications by offering a flexible framework for building workflows that integrate machine learning and big data analytics tools. These workflows can process selected datasets and deliver the corresponding results.

The workbench provides the following main functionalities:

- user-friendly interface: an intuitive, graphical environment for constructing workflows using drag-and-drop components, enabling dynamic parameter tuning and advanced visual analytics;
- workflow management: capabilities to store, edit, reuse, and share workflows, while offering both private and public accessibility options;
- execution monitoring: tools to initiate, schedule, and track the performance of workflows securely within the project platform.

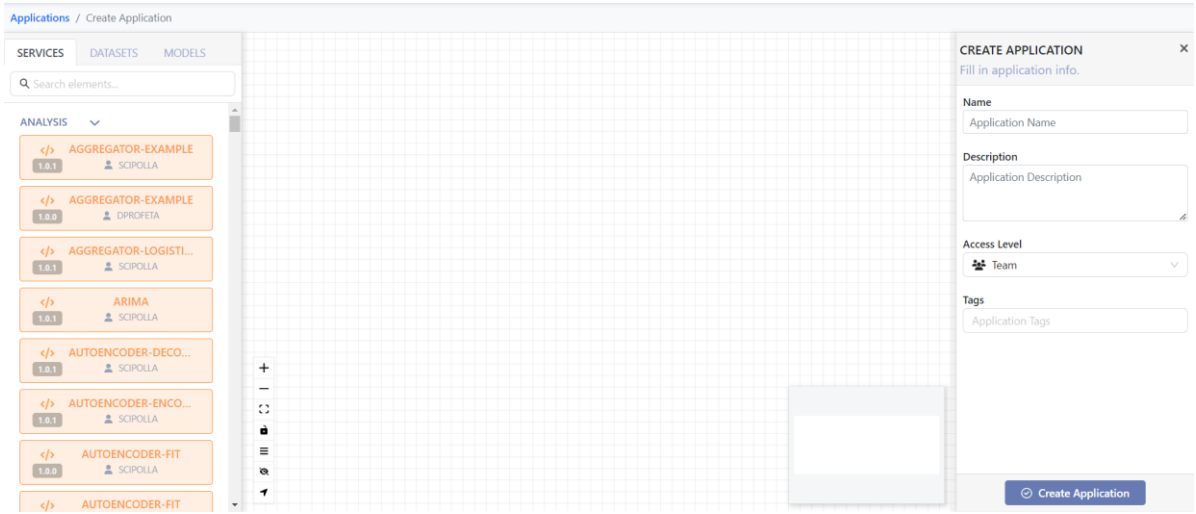

Figure 19: SUT Common workbench overview

The workflow environment allows users to create and execute applications by following key steps:

- data source selection: identify and choose one or more datasets to be used for processing;
- algorithm configuration: select machine learning or analytics algorithms from a library of state-of-the-art methods, adjusting their parameters as needed;
- results visualization: choose appropriate tools to present the outcomes with interactive and customizable visual representations;
- application profile definition: specify general application information, access control policies, and privacy settings, with options to set the application as private, public, or team-restricted.

When designing a workflow, users define the process steps, and the system automatically determines whether the operations should be executed in batch or streaming mode based on the nature of the data and the selected algorithms.

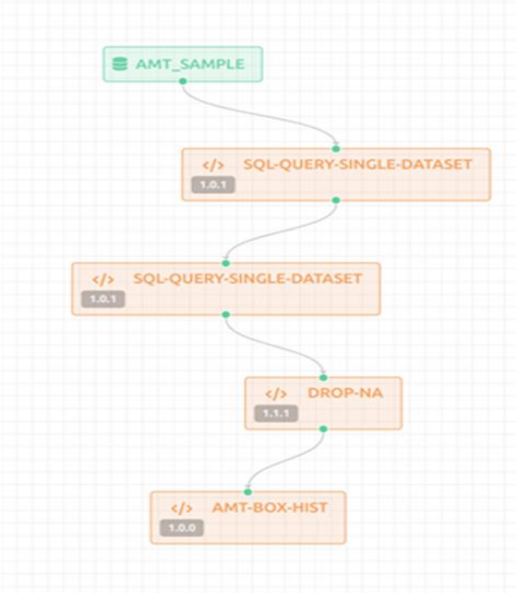

Figure 20: SUT Common workbench – Example of workflow

5.1.2. Data Insights and Analysis

The graphical interface simplifies the user interaction process. For demonstration purposes, the tool was applied to analyse sample data, focusing on statistical insights. In detail, the workflow involves several stages:

- dataset registration and selection: users can register and select datasets via a dedicated interface, for instance a sample dataset containing various records and attributes is show below;

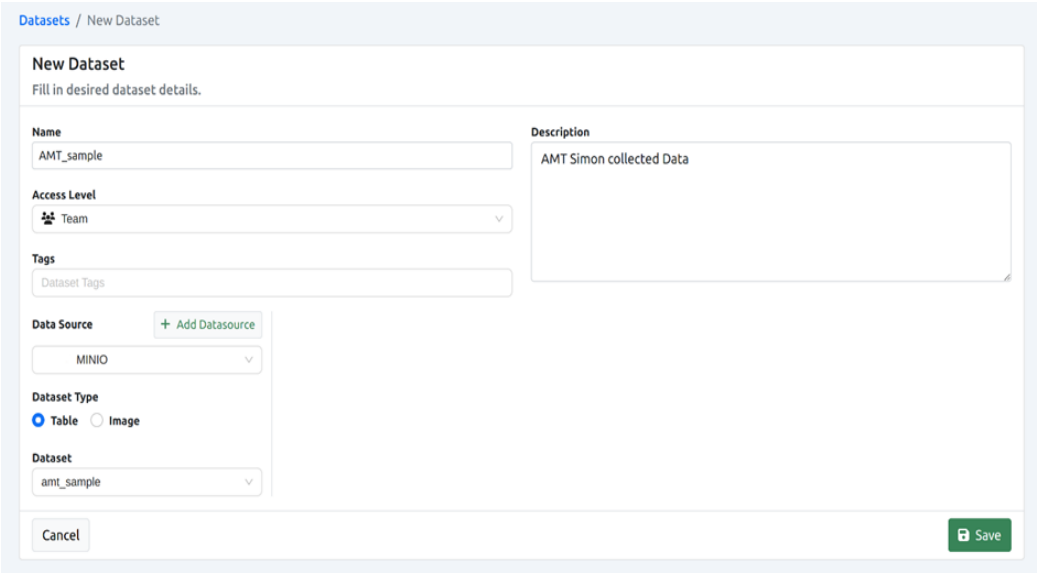

Figure 21: SUT Common workbench – Example of dataset registration and selection

- data filtering: SQL-like query services enable the selection of specific records, allowing for targeted data analysis, as shown below;

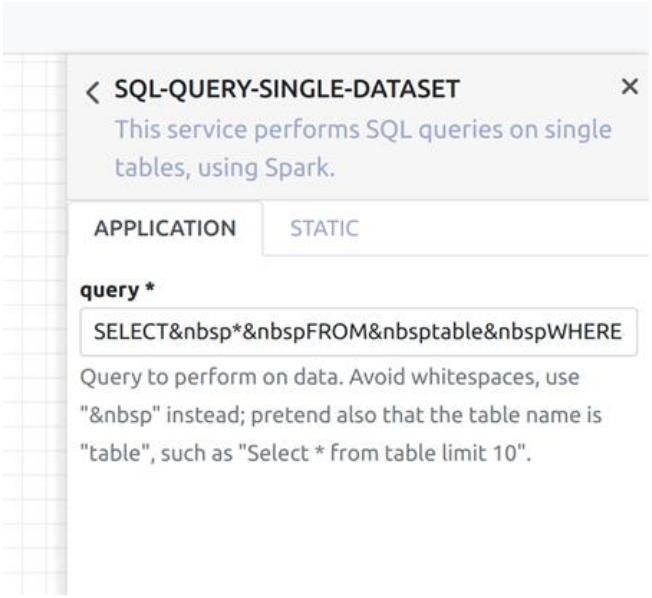

Figure 22: SUT Common workbench – Example of data filtering

- subset filtering: this step extracts specific records based on defined conditions, ensuring that only relevant data is retained;
- null value removal: the workflow includes services for detecting and eliminating null values to ensure data integrity during processing;
- date translation and statistical analysis: data are properly converted, and basic statistical measures like mean and median are computed, while outliers are identified, and results are visualized using boxplots and histograms, as shown below.

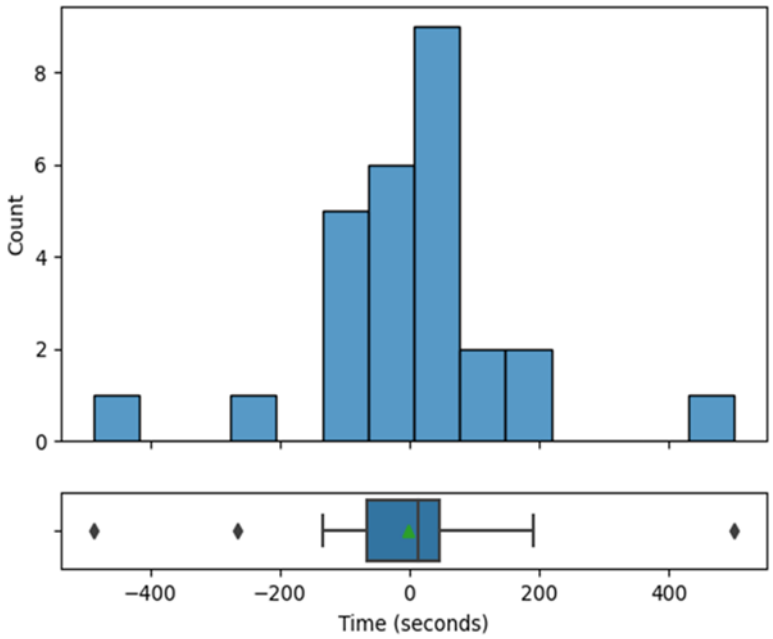

Figure 23: SUT Common workbench – Example of data translation and statistical analysis visualisation

The resulting insights offer a clear view of data distributions and anomalies, aiding in operational decision-making.

5.1.3. Specifications about the backend

The workbench, in its current stage, primarily revolves around its frontend: this component acts as the primary interface for users to access tools, data, algorithms, and application parameters. Users can fine-tune and

organize these resources into a coherent workflow through a streamlined, graphical interface. The entire process of constructing workflows, which utilize both cloud and edge resources, is simplified and transparent: users select and configure blocks to define the operational flow without needing to handle underlying technical complexities. The frontend coordinates with three core components:

- AI catalogue: this repository provides access to machine learning and big data analytics algorithms, that users can select from available modules to process data and extract insights. Once selected, these algorithms are passed to the orchestrator for execution.
- Workflow builder backend: this component translates the user-defined workflow into a YAML file, which communicates the necessary specifications to the resource allocator for processing and resource optimization. Integration with external systems like databases for data storage and Keycloak for authentication has also been implemented.
- Status manager: this module tracks and presents real-time workflow execution statuses by interacting with the Kubernetes cluster via APIs. It relays updates to the frontend and maintains the status information for services and workflows within the AI catalogue.

5.1.3.1 Component Description

The SUT Common facilitates the design, deployment, and execution of AI-driven workflows through a graphical interface and containerized microservices. The system leverages several open-source technologies, including:

- Resource allocator: it manages available computational resources;
- Argo Workflows engine: it orchestrates parallel job execution across allocated resources;
- Apache Kafka: it supports distributed event streaming;
- Workflow designer: it is a drag-and-drop interface for workflow composition, built with modern front-end frameworks like React and Redux;
- JupyterHub: it provides access to Jupyter environments for interactive data exploration.

The architecture supports the following core services:

- Workflow Builder: the React-based front end serves as the primary interface. Its backend manages requests, processes pipelines, communicates with the resource allocator and engine, and handles access control through OAuth 2.0 (Keycloak integration);
- Status Manager: this component oversees workflow status updates, transmitting event statuses to the front end via Kafka streams. It ensures real-time updates without page refreshes by interacting with the orchestrator client and monitoring resources via the status watcher;
- Catalogue: this module manages datasets, models, and related metadata using a PostgreSQL database. It supports standard operations like create, read, update, and delete.

5.1.3.2 Operational Evaluation

The functionality of the developed components was validated through the following steps:

1. Workflow design: users successfully constructed workflows using the graphical interface, as shown below;

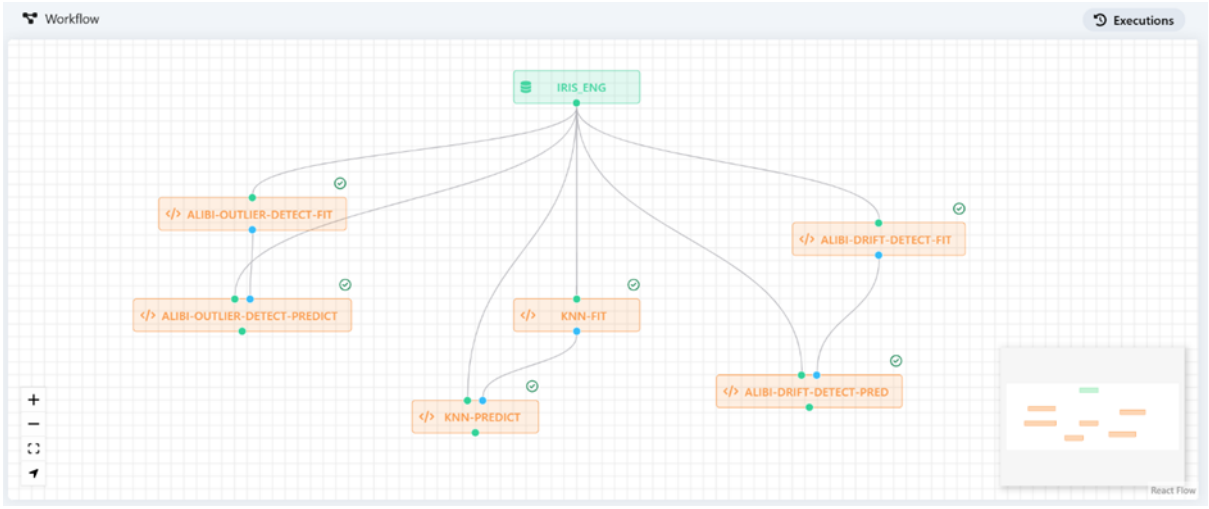

Figure 24: SUT Common workflow – Evaluation of design capabilities

2. Workflow translation: the designed workflows were correctly converted into Argo Workflows-compliant YAML specifications, as shown below;

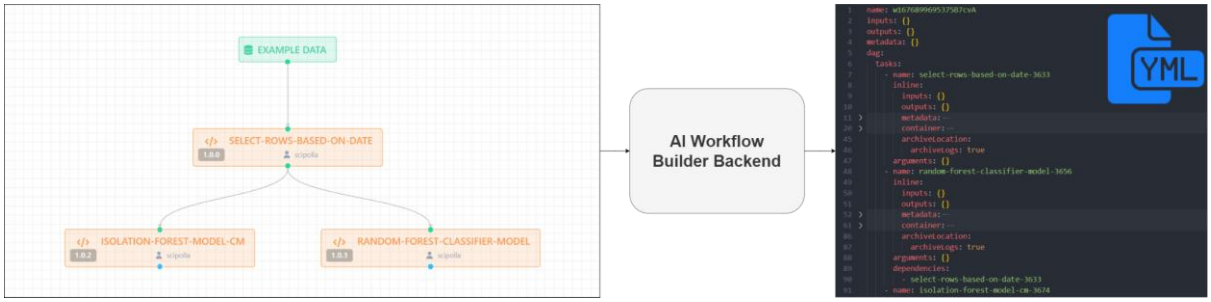

Figure 25: SUT Common workflow – Evaluation of translation capabilities

3. Execution monitoring: the system correctly launched and monitored workloads within the Kubernetes cluster, as shown below;

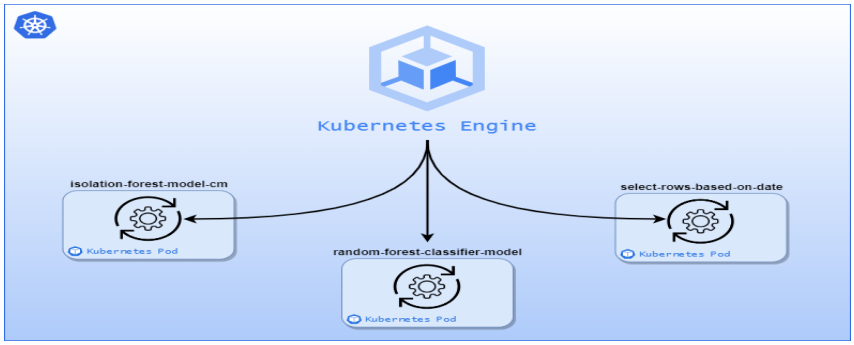

Figure 26: SUT Common workflow – Evaluation of execution monitoring capabilities

4. Data interaction via Jupyter: Jupyter Notebooks were tested with sample datasets to validate interactive analytical capabilities.

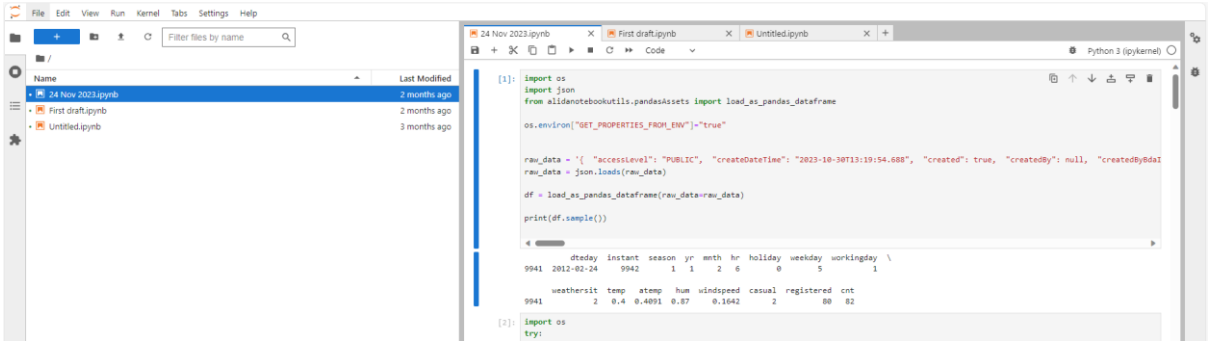

Figure 27: SUT Common workflow – Evaluation of data interaction capabilities

5.2. Cloud-edge resources management through K3S

This section outlines how the point (2) within the introductory paragraph has been addressed, by leveraging a cloud-edge resource manager based on K3s orchestration capabilities to support AI processing in an intuitive and scalable manner.

5.2.1. K3s: key features and use cases

K3s is a lightweight and certified Kubernetes distribution, designed for resource-constrained environments and edge computing. It is developed by Rancher Labs and it is optimized for simplicity, efficiency, and ease of deployment. Although its extended capabilities, its main key features can be summarised as follows:

- Lightweight. A single binary is about 100 MB with all dependencies, making it much smaller than standard Kubernetes.
- Less resource intensive. It uses minimal CPU and RAM, ideal for edge devices, IoT, and low-power hardware (like Raspberry Pi).
- Simplified installation. A single command (`curl -sfL https://get.k3s.io | sh -`) installs and runs K3s quickly.
- Built-in SQLite support. By default, it uses SQLite instead of etcd, though it supports external databases like MySQL and PostgreSQL.
- Integrated components. It comes with Traefik (ingress), Flannel (CNI), and CoreDNS pre-installed.
- Optimized for edge and IoT. It works well in environments where full Kubernetes would be overkill.

Given these features, it becomes easier to adapt it to different use cases, such as:

- Edge computing, running Kubernetes clusters on small devices (IoT, Raspberry Pi, ARM-based hardware).
- CI/CD pipelines: it is a lightweight Kubernetes for testing and development.
- Development and local testing: it is quick and easy setup for Kubernetes applications.
- Embedded and industrial applications: deploying containers in factories, retail, and remote sites.

This said, it is simply explained how a solution like this can fit in a research project like THEMIS 5.0 where it is needed to manage several changeable variables.

5.2.2. K3s: architectural overview

K3s simplifies Kubernetes by removing unnecessary components and optimizing it for edge computing, IoT, and small-scale clusters. Below is a brief breakdown of its architecture and key design principles.

5.2.2.1 Key architectural differences from Kubernetes

K3s maintains the core Kubernetes functionality while making several optimizations:

- Single binary (about 100MB). Instead of multiple binaries (like kube-apiserver, kube-controller-manager, kubelet, etc.), K3s compiles them into one lightweight binary.
- SQLite by default. Unlike Kubernetes, which relies on etcd for cluster state storage, K3s uses SQLite by default, reducing resource consumption. It also supports external databases like MySQL and PostgreSQL.
- Simplified networking. Flannel is the default CNI (Container Network Interface) for networking, simplifying setup.
- Built-in components. Includes Traefik as the default ingress controller and CoreDNS for service discovery.
- Removes legacy/unused features. Alpha features, in-tree cloud providers, and storage drivers are stripped out to reduce overhead.

5.2.2.2 Cluster architecture

K3s follows a server-agent architecture, like Kubernetes' control plane and worker nodes but optimized for lightweight deployments. In detail:

- Server Node (Control Plane)
  - Runs the API Server, Scheduler, and Controller Manager as a single process.
  - Uses SQLite (or external DB) to store cluster state.
  - Can be single-node or multi-master (with external database).
- Agent Nodes (Worker Nodes)
  - Runs the k3s agent, which includes:
    - kubelet (for managing containers).
    - A lightweight container runtime (containerd by default, replacing Docker).
    - Network and storage plugins.
  - Communicates with the server node via API.
- High Availability Mode
  - If you need multi-master (HA mode):
    - Multiple server nodes can share an external database (PostgreSQL/MySQL/MariaDB).
    - Load balancing is required for API server redundancy.
    - Worker nodes connect to any available server node.

5.2.2.3 Networking

K3s uses Flannel as the default CNI (Container Network Interface). In detail this means:

- Simple overlay networking (VXLAN backend).
- Supports WireGuard for encrypted networking.
- Can be replaced with other CNIs like Calico or Cilium.

For ingress, K3s includes Traefik, a lightweight ingress controller, out of the box.

In the SUT Common infrastructure, WireGuard has been configured in order to support the federation of additional Kubernetes nodes located elsewhere securely.

5.2.2.4 Storage

K3s implements several tricks for optimal storage management. In detail:

- Default storage backend: SQLite (lightweight but only for single-node clusters).
- For HA clusters, you need an external database (MySQL, PostgreSQL, etc.).
- Supports local storage (hostPath, local persistent volumes).
- Compatible with CSI (Container Storage Interface) plugins for cloud storage (EBS, Ceph, NFS, etc.).

5.2.2.5 Security and Upgrades

To always grant security, there are:

- Automatic TLS certificate rotation and secure API communication.
- Lightweight RBAC (Role-Based Access Control) like Kubernetes.
- Seamless upgrades with rolling restarts (k3s upgrade).

5.2.2.6 Differences at a glance

To sum up, K3s is a lightweight, optimized Kubernetes distribution that works well for edge computing, small clusters, and low-power environments. It simplifies installation, removes unnecessary components, and provides a fully functional Kubernetes environment with lower overhead. If necessary, it can be extended as required.

Table 22: K3s and Kubernetes differences at a glance

Feature	K3s	Kubernetes
Binary Size	About 100 MB	About 300+ MB
Storage Backend	SQLite (or external DB)	etcd
Resource Usage	Low (ideal for edge/IoT)	High
Networking	Flannel (default)	Various CNIs
Built-in Components	Traefik, CoreDNS	None (must install separately)
Ease of Setup	`curl -sfL https://get.k3s.io	sh -`
Best for	Edge, IoT, Dev/Test, CI/CD	Large-scale production

6. CONTINUOUS SYSTEM INTEGRATION, TESTING AND QUALITY ASSURANCE

The THEMIS 5.0 platform, with all its services, is developed following a logic of continuous integration, delivery and deployment, with automated and synchronized functional verification and integration based on the needs of the partners sharing the code in order to achieve high-quality code levels. The following sections detail the infrastructure adopted for the THEMIS 5.0 platform, with details on the technologies adopted e.g. GitLab, Argo, Kubernetes, as well as DevOps guidelines for software verification and container registry management equipped with practical templates for pragmatic implementation.

6.1. CI\CD: THEMIS 5.0 approach

The THEMIS 5.0 platform implements an advanced Continuous Integration and Continuous Delivery (CI/CD) strategy designed to automate, streamline, and enhance the development and deployment processes. This strategy underpins the integration of services, allowing the delivery of robust, scalable, and high-quality software. By leveraging automation, scalability, and resilience, this CI/CD approach enables rapid development cycles without compromising system reliability or stability. Central to this implementation are the open-source components deployed on Kubernetes infrastructure, alongside the integration of GitLab for Continuous Integration and Argo CD for Continuous Delivery.

The CI/CD methodology for THEMIS 5.0 is driven by several core objectives. One key goal is to enhance collaboration across distributed development teams by ensuring centralized version control and standardized workflows. Another priority is to reduce the time required to deliver new features or resolve defects by automating labour-intensive processes, such as testing, building, and deployment.

The CI/CD pipeline architecture for THEMIS 5.0 has been designed to accommodate diverse workflows that extend from code validation to production deployment. Initially, source code is managed through GitLab repositories, which serve as a centralized hub for version control, ensuring traceability and effective collaboration. Automated testing forms the next critical stage, where pipelines execute processes such as linting, unit tests, and integration tests to ensure code quality and functionality. Upon successful validation, Docker images are built and stored securely in a container registry, ensuring that deployment artifacts are readily available. Finally, the deployment process is managed by Argo CD, which synchronizes Kubernetes cluster with the desired state defined in Git repositories. The CI/CD pipeline, based on GitOps principles, is summarised in the following diagrams. The following figure describes the CI/CD from a developer perspective. The process starts with the developer versioning his source code within a Git repository by adopting, for instance, the Git Branch Workflow, which will be explained in more detail in a later section. The push into the GitLab repository triggers Continuous Integration (CI) pipelines executed by the GitLab Runner, which packages the application, test it, and finally push the containerised application as a Docker image into the GitLab Container Registry. GitLab has been then chosen both as a repository of the developers' source code, and as a place to define and execute mainly CI pipelines using GitLab Pipelines; finally, as a container registry to register the automatically produced Docker images within the GitLab pipelines; in the last step, the Argo CD framework, installed and configured within the THEMIS Kubernetes cluster, takes care of the Continuous Deployment part, adopting a ‘pull’ approach, monitors the Docker images within the GitLab Container Registry and automatically deploys the new Docker images.

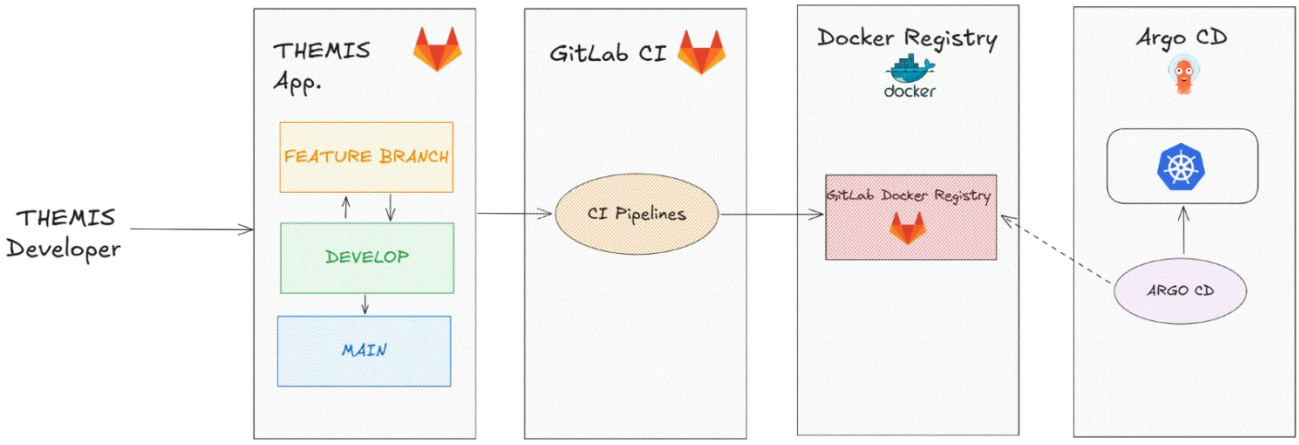

Figure 28: CI\CD process from developer perspective

Figure below describes the CI/CD process for the Kubernetes manifest of all applications in the THEMIS project. In this flow, DevOps engineers first collect the specifications and requirements of each application. These are then translated into Kubernetes manifest, all of which are collected within a Git repository, adopting the basic principles of GitOps. Indeed, on the Kubernetes cluster side, the Argo CD configured in the cluster takes care of checking for updates in the Kubernetes manifest repository and applying them directly within the Kubernetes cluster.

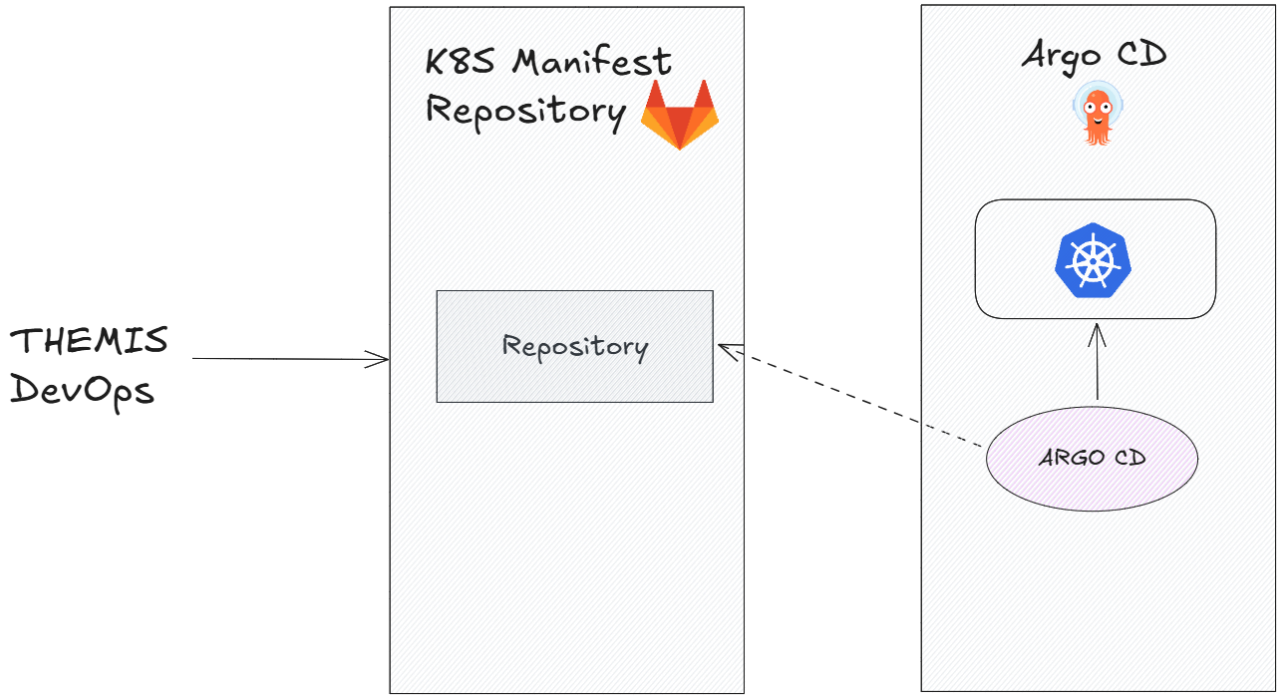

Figure 29: CI\CD process from DevOps perspective

As described above, GitOps principles form the foundation of THEMIS 5.0's CI\CD approach. These practices prioritize declarative configuration, ensuring that Kubernetes manifests and Helm charts stored in Git repositories act as the single source of truth for both application and infrastructure states. Changes made to these repositories automatically trigger updates in Kubernetes clusters, thereby eliminating the need for manual intervention. Integrated monitoring and feedback loops further enhance this approach, providing developers with actionable insights into deployment outcomes and overall system performance.

In the following, the technologies and frameworks used to support the DevOps processes just illustrated are explained and described in more detail.

6.1.1. Technologies

The implementation of CI\CD in THEMIS 5.0 relies on a well-curated set of tools and technologies. Among these, GitLab emerges as a comprehensive DevOps platform that seamlessly integrates version control with CI pipeline management. Argo CD, another critical component, is employed to manage continuous delivery through a declarative GitOps approach. Kubernetes serves as the backbone for deploying, scaling, and managing containerized applications, ensuring robust and scalable operations across all environments.

6.1.1.1 GitLab for Source Code and CI Management

GitLab (https://about.gitlab.com/) plays a pivotal role in managing the CI processes for THEMIS 5.0. As an all-encompassing DevOps platform, it provides a unified solution for source code management and CI pipeline automation. GitLab pipelines are modular and facilitate workflows that include linting, testing, and containerization. The integrated artifact management capabilities ensure that build outputs and Docker images are securely stored and easily accessible. Furthermore, GitLab's built-in container registry streamlines the storage and retrieval of deployment artifacts.

When it comes to Git repository management and DevOps integration, three major platforms dominate the landscape: Bitbucket, GitLab, and GitHub. We have chosen GitLab as it supports both public and private development branches and can be freely installed and hosted on premises, providing an end-to-end DevOps solution.

One of the most compelling advantages of GitLab is its built-in CI/CD pipeline. With GitLab CI/CD, teams can automate testing, security scanning, and deployment without needing to configure external tools. Furthermore, GitLab's Auto DevOps feature simplifies this process by automatically detecting project settings and generating pipelines with minimal user intervention.

Security is another key strength of GitLab. It offers native security scanning tools, including Static Application Security Testing (SAST), Dynamic Application Security Testing (DAST), container scanning, and dependency scanning. These security features help development teams identify vulnerabilities early in the software lifecycle, reducing risks and improving code quality. Unlike GitHub, which requires additional paid plans for advanced security, GitLab includes many of these features in its core offerings.

For organizations that require self-hosting, GitLab provides one of the best solutions available. While both GitHub and Bitbucket offer enterprise self-hosted versions, GitLab's self-hosted instance (GitLab Community Edition and GitLab Enterprise Edition) is more feature-complete and gives organizations full control over their data, compliance, and security settings.

Another area where GitLab excels is project management. While GitHub and Bitbucket require external tools like Jira for advanced issue tracking and Agile workflows, GitLab provides integrated project management features such as issue boards, milestones, epics, and roadmaps. This eliminates the need for additional software, streamlining development workflows and reducing costs for teams that want a single, unified platform.

6.1.1.2 Argo CD for Continuous Deployment

In the world of Kubernetes application deployment and continuous delivery, Argo CD (https://argoproj.github.io/cd/) has gained a significant position as a GitOps-based tool that ensures applications remain synchronised with their declared configurations.

The advent of Kubernetes, in fact, has prompted the implementation of frameworks/tools that are better suited to the continuous deployment of applications within Kubernetes clusters. Argo CD, in contrast to other non-native Kubernetes tools, is distinguished by its declarative approach, real-time synchronisation and user-friendly interface. This means that application definitions, configurations, and environments are stored in Git repositories, and Argo CD continuously monitors these repositories to ensure that the deployed state of an

application matches its desired state. If any drift occurs, Argo CD can automatically reconcile it or alert the team to take corrective action.

Unlike traditional CI/CD tools that execute scripts and deploy artifacts, Argo CD follows a declarative approach: developers define their Kubernetes resources in YAML files, commit them to Git, and let Argo CD handle the rest. This approach improves reliability, enhances security, and ensures auditability by making Git the single source of truth for application state.

One of its primary advantages is its support for declarative infrastructure management, allowing teams to utilize Kubernetes manifests, Helm charts, and Kustomize overlays to define system configurations.

Additionally, Argo CD integrates real-time health monitoring to assess application readiness and performance metrics. For complex architectures, it offers multi-cluster management, ensuring scalability and operational efficiency.

To summarize, main features of Argo CD include:

- A user-friendly UI that provides real-time insights into application states;
- Out-of-the-box multi-cluster management, making it easy to deploy applications across different Kubernetes environments;
- Integrated security features, such as RBAC and SSO support, ensuring that deployments remain secure, and access is controlled;
- Lightweight and Kubernetes-native design, with support for declarative deployment approaches via Kubernetes manifest, integration with Helm package manager and Kustomize.

In addition, the “Image Updater” plugin has been also installed in the Argo CD instance made available and installed in the SUT Common, which is able to check for new versions of container images distributed with Kubernetes workloads and automatically update them to the latest version allowed using Argo CD. It works by setting the appropriate parameters for Argo CD applications. A simple example will be described in the guidelines section.

In the following figure is shown the executed Argo CD example application that you can find on the provided Argo CD, and its source manifest file in the provided GitLab repository containing a complete example template.

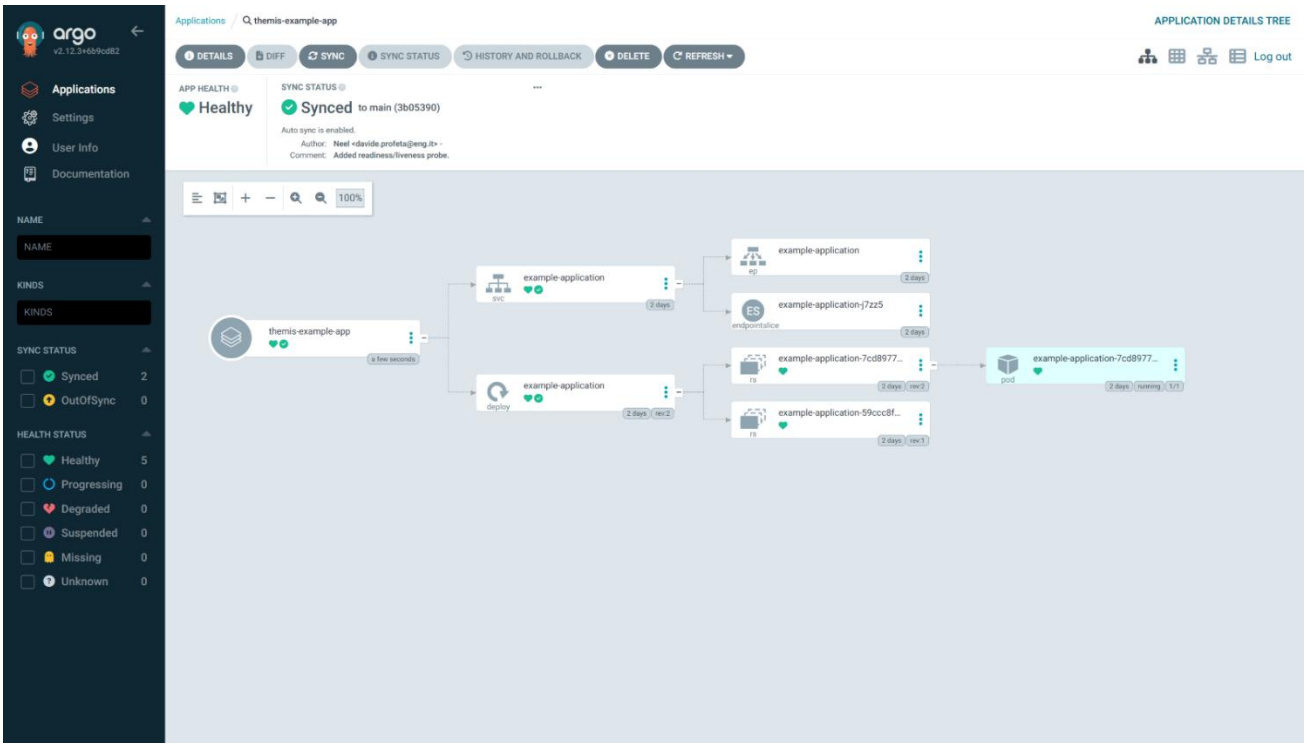

Figure 30: ArgoCD – Example Application

6.1.1.3 Kubernetes for Deployment and Scaling

Kubernetes (https://kubernetes.io/), often abbreviated as K8s, has become the de facto standard for container orchestration. Originally developed by Google and later open-sourced in 2014, Kubernetes automates the deployment, scaling, and management of containerized applications. It is now maintained by the Cloud Native Computing Foundation (CNCF) and is widely used across industries to build and manage scalable, resilient, and cloud-native applications.

While alternatives such as Docker Swarm, Nomad, and OpenShift exist, Kubernetes stands out due to its flexibility, scalability, and vast ecosystem of extensions and integrations. Kubernetes is a container orchestration platform that enables developers and DevOps teams to:

- Deploy and manage containerized applications automatically.
- Scale applications up or down based on demand.
- Ensure high availability by automatically restarting or relocating failed containers.
- Manage networking, storage, and security for containerized applications.

Instead of manually deploying containers on individual servers, Kubernetes provides an automated, declarative approach to managing infrastructure. Users define their desired application state in YAML manifests, and Kubernetes ensures that the cluster always aligns with this state.

In THEMIS 5.0, Kubernetes becomes the main place to deploy and scale the implemented applications for the project purposes. It provides an orchestrated, automated, and standardized environment where partners can deploy their applications inside a shared cluster, ensuring consistency, flexibility, and ease of management. The use of Argo CD also provides via a graphical interface to monitor the applications deployed within the Kubernetes cluster, providing greater control over the status of the cluster and applications.

6.1.2. DevOps Guidelines for THEMIS 5.0

So far, we have provided an overview of the approach and tools chosen to ensure the Continuous System Integration, Testing and QA (quality assurance) of the THEMIS 5.0 project's services and platforms. In this sub-section, we present guidelines, both methodological and practical with examples, of how to follow and practise the principles of DevOps within a research project such as THEMIS 5.0. But before proceeding, it is necessary to define and emphasise the importance of the basic DevOps principles that underpin the realisation of task T4.5.

DevOps is a set of practices and principles that aim to bridge the gap between software development and IT operations. In research projects, where multiple stakeholders collaborate on complex applications, adopting DevOps ensures faster deployments, improved collaboration, and more reliable software. By implementing DevOps, research teams can achieve continuous integration and deployment (CI/CD), monitoring, quality assurance and automation.

DevOps is built on several key principles that facilitate efficient software development and deployment. These principles ensure that research projects benefit from a streamlined workflow, reduced errors, and better reproducibility.

Traditionally, development and operations teams work in silos, leading to inefficiencies and misalignment. DevOps promotes a culture of shared responsibility, where developers, IT teams, and other stakeholders work together from the beginning. In THEMIS 5.0, this means:

- Improved coordination between developers, data scientists, and system administrators;
- Clear documentation and knowledge sharing;
- Faster resolution of issues due to open communication.

CI/CD is the backbone of DevOps, ensuring that software changes are integrated, tested, and deployed frequently. This is particularly useful in research projects where applications evolve rapidly.

- Continuous Integration (CI): Developers frequently merge their changes into a shared repository, where automated tests ensure stability;
- Continuous Deployment (CD): Once validated, the software is automatically deployed into THEMIS 5.0 Kubernetes environment;
- Benefits: faster iterations, immediate feedback, and reduction in manual errors.

Automation is a key driver of efficiency in DevOps. From code deployment to application deployment, automation eliminates human errors and speeds up workflows. In fact:

- Automated testing ensures code quality;
- Automated deployments reduce downtime;
- Automated scaling optimizes resource usage.

6.1.2.1 Principles of Software Verification: Unit, Functional and Integration Testing

Software verification within the THEMIS 5.0 CI/CD infrastructure is underpinned by several key principles. A comprehensive testing strategy is employed to validate the functionality, performance, and security of software components. Traceability is a critical aspect, with detailed logs maintained for all CI/CD activities to ensure accountability and reproducibility. Version control further strengthens the framework, enabling rollbacks and ensuring that every deployment is based on a known, validated state.

The CI/CD pipeline incorporates multiple levels of testing to ensure software reliability. Unit testing focuses on verifying the behaviour of individual components in isolation, ensuring they meet their functional requirements. Functional testing evaluates whether the software behaves as expected under predefined conditions, while integration testing validates the interaction and communication between various components of the system.

Automated testing is a core principle for the THEMIS 5.0 CI/CD strategy. By reducing human intervention, it minimizes errors and accelerates feedback cycles, enabling developers to address issues promptly. However, manual testing remains an important complementary approach, particularly for identifying edge cases and usability issues that may not be captured by automated tests.

Test execution in THEMIS 5.0 is either scheduled at predefined intervals or triggered by specific events, such as code commits or merge requests. This ensures that tests are consistently executed across various development and deployment stages, providing reliable insights into software quality.

There are several open-source frameworks and libraries to facilitate test automation. For Python-based applications, PyTest is used for unit testing, while frameworks such as Selenium are utilized for end-to-end testing of web interfaces. The orchestration of these test workflows can be as well managed into GitLab pipelines CI.

6.1.2.2 Container Registry for Code and Module Sharing

The integration of a secure container registry within GitLab facilitates the sharing of containerized applications as Docker images. This ensures that deployment artifacts are readily available and traceable across the development lifecycle.

A container registry serves as a centralized repository for Docker images. It ensures secure storage and versioning, making it easier to reproduce and manage software deployments. The registry also enforces access control measures, ensuring that only authorized personnel can access sensitive artifacts.

The GitLab-integrated container registry is seamlessly integrated into the CI/CD pipeline of THEMIS 5.0. Automated cleanup processes are implemented to remove unused images, optimizing storage. Monitoring tools track usage metrics and maintain comprehensive access logs, ensuring operational transparency and efficiency.

Developers working within the THEMIS 5.0 ecosystem are encouraged to adhere to best practices for containerization. Lightweight base images are recommended to reduce vulnerabilities and improve performance. Additionally, detailed documentation of the build and deployment processes ensures that team members can easily understand and replicate workflows.

The automated build workflow in THEMIS 5.0 is defined by a series of triggers and pipeline stages. Builds are initiated through events such as code commits, merge requests, or scheduled jobs. The pipeline progresses through various stages, including linting, testing, and packaging, culminating in the generation of deployment-ready artifacts stored in the GitLab container registry.

6.1.2.3 A complete DevOps template example

In this sub-section, a complete example is provided to demonstrate and explain how to structure a CI/CD process, starting with the definition of GitLab pipelines that will handle the CI part and then explaining how to use Argo CD to set up the CD part in conjunction with a Git repo containing the Kubernetes manifest\Helm charts of the applications to be deployed.

The GitLab repository made available for the THEMIS 5.0 project, hosted by ENGINEERING, is available at the following URL: https://gitlabpa.eng.it/. This will act as the central hub for the source code of each application, where CI pipelines can be defined and executed to generate and store Docker images within the same GitLab Container Registry.

Argo CD, on the other hand, is available at the following URL: https://argocd.themis.alidalab.it/ and is part of the project's SUT Common. It can be used to define Argo CD Applications that automatically apply Kubernetes manifests or Helm Charts stored in a Git repository to a target Kubernetes cluster. Additionally, thanks to the Argo CD Image Updater plugin, Docker images referenced in the Kubernetes manifests can be automatically updated according to the rules defined in the Argo CD application.

Access to the GitLab instance and Argo CD is private due to policy and security reasons, and access will be granted on an individual basis. At the following URL https://gitlabpa.eng.it/repos/alida/themis/devops-template is available a complete DevOps application example, which is now illustrated step by step.

6.1.2.3.1 Example Application Template

The first repository being illustrated is "example-application", which contains the source code of a possible application example developed for THEMIS 5.0. In this example, the application is a simple server exposing REST APIs and using FastAPI, a Python framework for building APIs.

The general project structure for an application is as follows:

- **Source code folder**: in this example, the "app" folder contains the application's source code, which consists of Python scripts using the FastAPI library to serve REST APIs.
- **Dockerfile**: this file describes how the application should be containerized as a Docker image. The produced Docker image includes any necessary scripts or compiled code and executes the command to launch the application (defined in the entrypoint section of the Dockerfile). As mentioned earlier, the Dockerfile should be written to include only the strictly necessary dependencies. In this example, a lightweight Python-slim base image is used which contains only the Python interpreter, along with the source code and only the required Python libraries, listed in a *requirements.txt* file.
- **README.md**: this is the standard markdown file used to document the repository. It describes the purpose of the application and how to run and test it locally. Proper documentation is crucial to assist DevOps Engineers in writing CI/CD pipelines and creating Kubernetes manifest files necessary for deploying the application in a Kubernetes cluster.
- **Docker-compose.yml**: this optional file allows developers to run and test the application locally. It defines the configuration of all services, networks, and volumes required by an application in a single YAML file, simplifying the management and deployment of complex applications.
- **.gitlab-ci.yml**: this is the main YAML configuration file for triggering GitLab Pipelines. GitLab projects using CI/CD features have a *.gitlab-ci.yml* file in their root directory. GitLab detects this file during push and merge operations and parses it to determine the pipeline workloads to execute. This file is necessary to activate the continuous integration processes that build, test, and finally push the Docker image to the GitLab Container Registry. In the example repository, templates have been defined in a file *.gitlab-script-template.yml*, which are then referenced and customized in the main *.gitlab-ci.yml* file. This file defines the execution stages: in the example, the application is tested by invoking an exposed endpoint, then the Docker image is built and pushed to the GitLab Container Registry. If any of these steps fail, the image will not be uploaded to the container registry and, consequently, will not be deployed in the target Kubernetes cluster via the Argo CD applications defined in the SUT Common. Only after all CI steps are completed can Argo CD detect changes in the Docker image and automatically update the component in the target Kubernetes cluster.

The GitLab pipeline is structured as follows:

- A **test** job is executed to test the application by building a containerized test version of the application. This job runs on every push/merge to both the development and main branches.
- The **build and push job** for the containerized application runs only when a push/merge occurs on the main branch and after successfully passing the *test* job. This job builds and pushes the containerised application into the GitLab Container Registry.

Kaniko is used instead of Docker-in-Docker for executing GitLab Pipelines jobs, offering better performance in terms of speed and greater security since it does not require privileged mode.

Additional optional files:

- **DockerfileTest and requirements-test.txt**: these files define the containerized test application. This container includes additional libraries required for testing, such as *pytest*, which is executed during the build phase of the application.

6.1.2.3.2 Example K8S Manifests Template

The other example template, named *example-k8s-manifest* and provided within the *DevOps Template* group on GitLab is now being presented. This project contains an example Helm chart for the example application described previously and a definition of an Argo CD application, which can be created and applyed via the graphical interface available in the Argo CD installed in the SUT Common (available at the following URL: https://argocd.themis.alidalab.it/).

The Argo CD application defines the synchronization methods with the target repository that contains the manifests/Helm charts to be applied, specifying also which Kubernetes cluster and namespace these objects should be deployed to.

A Helm chart (https://helm.sh/docs/) is a collection of YAML template files organized in a specific directory structure (see the official documentation for more details), used to describe a related set of Kubernetes resources.

Within the *templates* folder of the Helm chart, a *Deployment* and *Service* objects for the example application described in the previous section have been defined. These manifests utilize Helm variables, whose values are specified in the *values.yaml* file. This feature enables the dynamic modification of values during deployment, which may vary depending on the target cluster (infrastructure-related variables) or due to changes in the application itself.

For instances, Helm values can be used to:

- enable or disable certain optional components using flags;
- define, as in the example provided, the minimum and maximum resources allocated to each application;
- specify the URL of the Docker image where the application resides;
- set other important parameters, such as the name of the Kubernetes secret containing access credentials for the Docker registry that hosts the images to be deployed.

The project includes both a Helm Chart and an example Argo CD application definition and is structured as follows:

- *argo-cd-apps/*: it contains an Argo CD Application defined as a Kubernetes manifest.

The example Argo CD Application is set up to apply Helm Charts (see documentation: https://argo-cd.readthedocs.io/en/stable/user-guide/helm/) contained in this project.
The main elements to define are:

- **Git repository reference** containing the Helm chart to apply;
- **Branch** to be used and any custom values to be applied;
- **Synchronization policies**;
- **Kubernetes cluster and namespace** (in this case, the same cluster where Argo CD is installed in the **SUT Common**) where the Helm chart should be deployed;
- **Optional configurations for Argo CD Image Updater** (https://argocd-image-updater.readthedocs.io/), a tool that automatically updates Kubernetes workloads' container images managed by Argo CD;

- **Optional notification configurations** (e.g., email notifications for failed or successful synchronizations).

In the provided example, the Helm Chart defined within this repository is applied using an **automatic synchronization policy**. This means that changes made to the Helm chart following a **push/merge** will automatically be propagated and updated in the Kubernetes namespace specified in the Argo CD Application.

Additionally, by configuring the **"Image Updater"** subcomponent of Argo CD, updates to the **example application** ("example-application")—specifically the Docker image produced by the CI pipeline—will be automatically applied to Kubernetes whenever a new update is detected in the Docker registry.

- *helm-charts/example-chart/*: it contains the example Helm chart.

A Helm chart is generally structured as follows:

- *Chart.yaml*: it contains basic information about the chart, such as the name, version, and description.
- *values.yaml*: it contains default values for the chart's parameters. This file can be customized for different environments. In this example, default values include:
  - The Kubernetes Secret name for pulling Docker images;
  - The RAM and CPU resources for the application;
  - The port on which the containerized application listens.
- The Docker image URL (split into name and tag) is also defined to enable Argo CD Image Updater (see documentation: https://argocd-image-updater.readthedocs.io/en/stable/). This ensures that Kubernetes components are automatically updated when a new Docker image is pushed to the registry.
- Additionally, auto-update policies can be defined within the Argo CD Application based on:
  - The latest image digest (as used in this example);
  - Specific tags following semantic versioning (SemVer).
- *templates/*: it contains Kubernetes YAML manifests with placeholders for configurable values from values.yaml. In this example template, only a Service and a Deployment have been defined.
- Depending on the application and target cluster, other Kubernetes objects can be included, such as:
  - Ingresses: to expose the application externally via a hostname;
  - Persistent Volume Claims (PVCs): to allocate storage for stateful applications.
- *charts/*: it contains any dependent charts (subcharts). If this chart depends on other Helm charts (e.g., a PostgreSQL database), they will be included here.
- *templates/tests/*: it contains test definitions to verify the correct installation of the chart.
- *crds/*: it contains any required Custom Resource Definitions (CRDs).
- *README.md*: it provides documentation for the chart.

7. CONCLUSION

The THEMIS 5.0 project marks a significant step toward establishing a structured approach to AI trustworthiness. By integrating robust assessment mechanisms, legal compliance, and iterative model improvement, the platform lays the groundwork for reliable AI applications across multiple domains. The initial release demonstrates its potential, highlighting key achievements in ensuring fairness, robustness, and accuracy in AI decision-making. As AI systems continue to evolve, future iterations of THEMIS 5.0 will focus on refining methodologies, expanding dataset benchmarks, and incorporating emerging regulatory requirements to maintain compliance and ethical standards.

One of the main risks faced by THEMIS 5.0 is legal compliance. Besides the GDPR and other relevant pieces of EU law, the AI Act was recently adopted and will start to apply in its entirety in the two years. Compliance by design with key provisions of the AI Act is essential to ensure the usability of THEMIS 5.0 platform. For this reason, key applicable legal requirements have been described throughout the deliverable to ensure that technical partners take them into account in their contributions.

Furthermore, the THEMIS 5.0 platform's adaptability ensures its relevance across various sectors, from healthcare to critical infrastructures, reinforcing its role as a pivotal tool for AI governance. The continued engagement of stakeholders, researchers, and industry leaders will be essential in driving innovation and addressing the dynamic challenges posed by AI deployment. Establishing trust in AI requires sustained efforts in transparency, explainability, and continuous performance evaluation. Future research and development will aim to enhance interoperability, scalability, and the integration of advanced machine learning techniques to further bolster AI trustworthiness. Ultimately, THEMIS 5.0 is poised to set a new benchmark in the responsible development and application of AI technologies.

8. ANNEX 1: C4 MODEL – USE AND ADAPTATION IN THEMIS 5.0 PROJECT

In the THEMIS 5.0 project, it was decided to use the C4 model[36] as a reference for developing architectural diagrams, exploiting both part of its hierarchical abstractions (software systems, containers, components) and part of its corresponding hierarchical diagrams (system context, containers, components). An overview is provided in the figure below, taken directly from the C4 model website referenced earlier.

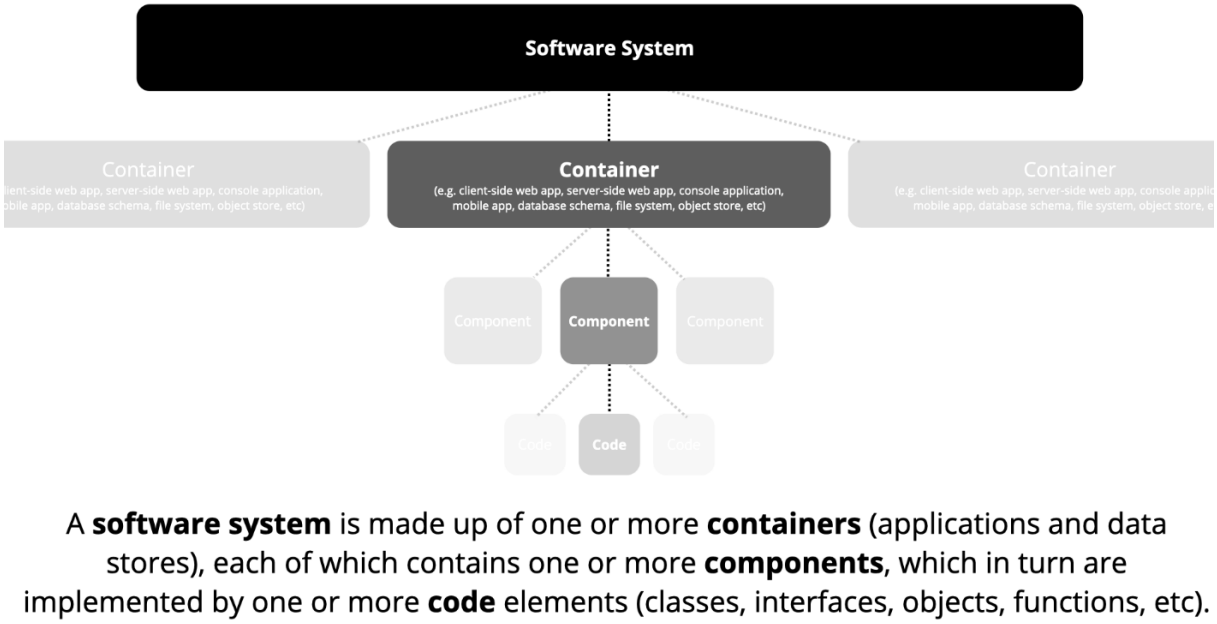

Figure 31: C4 model abstractions

In detail, the C4 model is a methodology for visualizing and documenting software architecture. It was created by Simon Brown and provides a structured way to model software systems at different levels of abstraction. The C4 stands for the four levels of architecture (hierarchical) diagrams:

- System Context Diagram: it shows the big picture, including users, external systems, and how they interact with your system.
- Container Diagram: it breaks the system into major components (e.g., applications, databases, microservices) and shows their interactions.
- Component Diagram: it zooms in on a single container, detailing the internal components and how they communicate.
- Code Diagram (Optional): it provides the lowest-level view, usually depicting class-level structures in a specific programming language.

The C4 model is widely used in modern software architecture because it balances clarity, simplicity, and technical depth. It often complements Unified Modelling Language and Entity-Relationship Diagrams but focuses more on software structure and communication.

This said, it is clear that the strength of this model lies in its great flexibility that allows it to be adapted according to specific needs. In particular, in the case of THEMIS 5.0, the first three hierarchical levels were used while introducing an additional level, a subcategory of the third and called “subcomponents,” where necessary. This made it possible to properly subdivide the work and assign it to those directly responsible.

[36] https://c4model.com/

In addition, since the C4 model is notation-independent, this made it possible to simplify and make interactions between software elements immediately visible, going to agree appropriately from the point of view of both “what” and “how” by reporting it directly on the connecting arrows.

Furthermore, it was decided to use its default colours to distinguish between internal elements and external elements, i.e. blue-based colours and grey-based colours. In the following figure are summarised the main graphical elements of the C4 model.

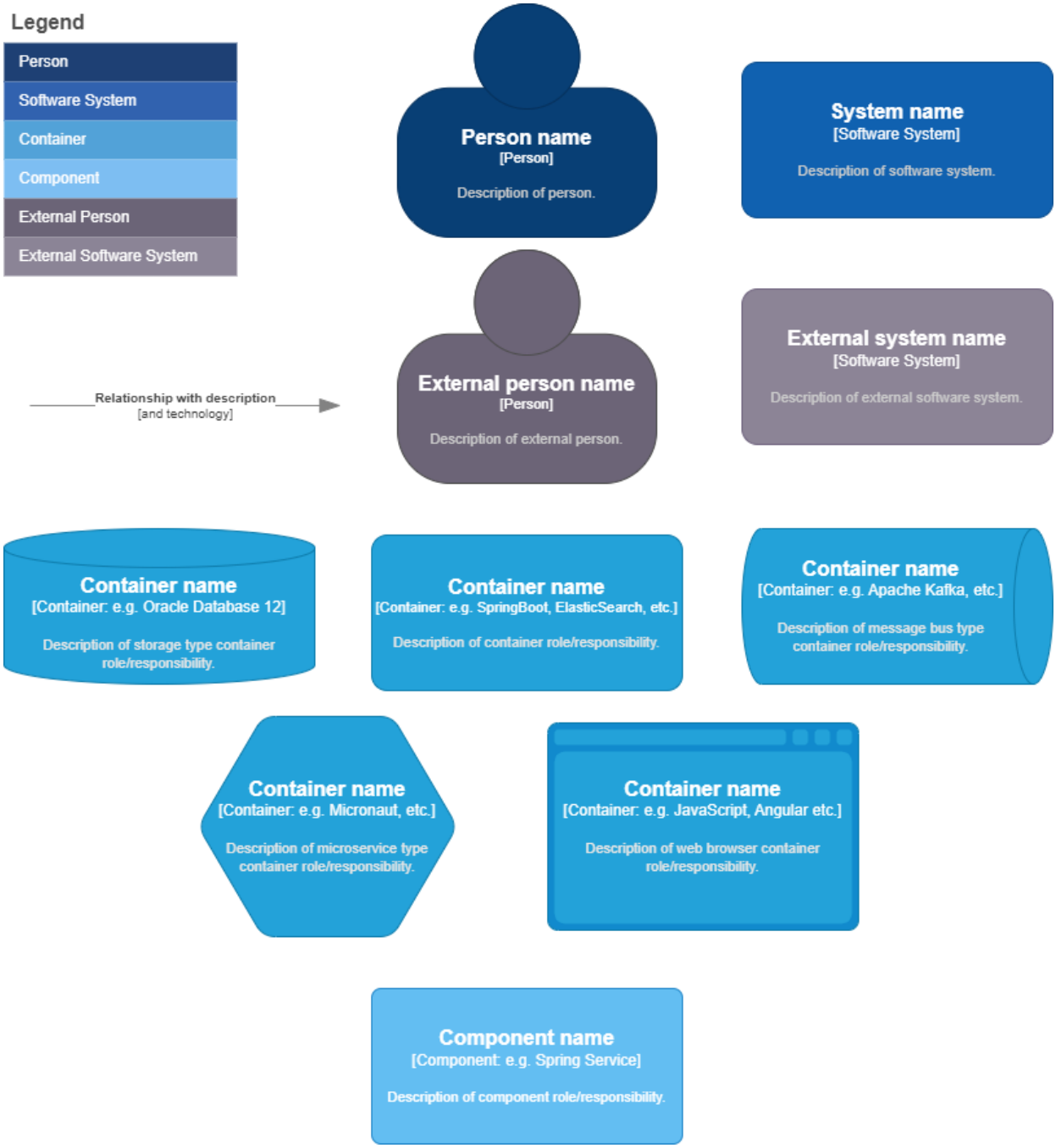

Figure 32: C4 model graphical elements

Should further specifications be necessary in the future, we will be able to rely on both the supplementary and further detail diagrams as made available by the C4 model.

9. ANNEX 2: METHOD CARDS

Deliverable D2.2 defines six different documentation cards (i.e., Use Case Card, Data Card, Model Card, AI Actor Card, Risk Card and Method Card) and proposes templates for how these should be documented. The Method Card is designed to host information about algorithms that relate to ML models and can be used to assess or enhance the trustworthiness of AI systems. For THEMIS 5.0 methods that are AI/ML-based, the Method Card will include references to relevant Model Cards. In the subsection below we provide examples of documented Method Cards, including referenced Model Cards, that are or will be supported by the THEMIS 5.0 platform.

9.1. SafetyCage – Misclassification detection

Table 23: Method card for SafetyCage – misclassification detection

Card section / field	SafetyCage – Misclassification detection
Method Details *	This section contains basic information about the method.
Method *	Misclassification detection.
Method name *	SafetyCage
Description *	Given an AI classification model and a particular prediction, assess the certainty in the prediction, and evaluate whether prediction is correct or wrong.
Trustworthiness characteristic *	Accuracy
Limitations *	To be used for classification models, and the method is inherently not to be used for regression models even though the approach also can be adapted for regression models. Note: Method is inherently not to be used for evaluating robustness, even though the framework also covers a method which is relevant for assessment of robustness. See the Mahalanobis SafetyCage model card for more details regarding this.
Limitations keywords	-
Conflicts	-
Conflicts keywords	-
Model-based *	Yes
Software available *	Yes
Contact info *	• Pål Vegard Johnsen (pal.johnsen@sintef.no) • Joel Bjervig (joel.bjervig@sintef.no)
Additional info	-
Stage details	This section contains information about the stages.
Stage	Develop
Sub-process	Model design, data preparation
Characteristics	Pre-processing, model centric, data centric
AI actors	ML engineer
Entity dimension	Model and data
Additional info	-
Software details	This section contains details about the available software code.
Code reference	-
Code repository *	• Private repositories at SINTEF's GitLab: (Access given to THEMIS 5.0 members on request) ○ https://gitlab.sintef.no/themis-sintef/themis-accuracy-robustness ○ https://gitlab.sintef.no/aai-internal/safetycage

	• Public repository on GitHub: Will be openly available at public launch summer 2025 ○ https://github.com/SINTEF
Input data *	The SafetyCage sofware requires the following input: 1) The model architecture from the AI model than can be loaded in the Python programming language. 2) The historical data that was used to train the AI model. 3) During deployment: Access to input data to model and the corresponding output or prediction from the AI model (prediction of the AI-model is accessible once the trained model is available (given by point 1).
Needed entities	1) Model: Python object with .predict(x) function. 2) Numpy array (python object) of inputs that are used during training of the AI model. 3) Input x as a numpy array to an AI-model, and numpy array y_hat which is the corresponding classification prediction.
Output *	For each prediction, a quantity that measures uncertainty in a prediction. Depending on the misclassification method used in SafetyCage, the value will be one of the following: 1) A p-value (between 0 and 1), where the uncertainty in the prediction is larger the smaller the p-value is. Hence, a small p-value indicates the prediction is wrong. This is the case when applying the Mahalanobis SafetyCage or the SPARDACUS SafetyCage. 2) A probability (between 0 and 1), where the uncertainty in the prediction is larger the smaller the probability is. Hence, a small probability indicates the prediction is wrong. This is the case for the maximum softmax probability (MSP) method. 3) An odds value (between zero and infinity), where the uncertainty in the prediction is larger the larger the odds value is. Hence, a large odds value indicates the prediction is wrong. This is the case for the DOCTOR method. Together with the raw value from SafetyCage, a set of tools including visualizations will also be provided for the end-user to finally make the decision as to whether the prediction is to trust or not. See output meaning for more details regarding these tools.
Explanations *	This section contains explanation about the output.
Output meaning *	The outputs are established for the purpose of giving the end-user the tools necessary to evaluating the trustworthiness in the prediction in terms of its accuracy. The outputs are the following: 1) A flag for each prediction that indicates whether a prediction is correct or not. The flagging procedure requires a thresholding in which a prediction is deemed as wrong whenever the output from SafetyCage is smaller/larger than this threshold. The threshold will be fitted according to some evaluation metric (for instance precision or recall). Some evaluation metrics may be regarded as more important than others. In light of this, a traffic light system will be established which indicates the risk connected to each prediction when one evaluation metric is more important than another. Green light means prediction can be regarded as safe with respect to the most important evaluation metric. Orange means that prediction should be carefully evaluated. Red means that prediction has high risk. 2) If prediction is deemed as being wrongly predicted, provide X historical inputs with similar prediction errors. This procedure will be based on a K-nearest neighbour (KNN) procedure. 3) If applicable, counterfactual explanations will be provided to further assess the trustworthiness of the prediction. See counterfactual explanation model card for more details.
Metrics *	Precision, recall, sensitivity, negative predictive value and Matthews correlation coefficient (MCC).
Metrics meaning *	The metrics measure the performance of the SafetyCage software on historical data. This means, the degree to which the SafetyCage is correctly flagging a prediction from an AI-model as both correct or wrong. The metrics evaluate different aspects of performance.
Ideal values *	All metrics are better the larger they are (all metrics have 1 as maximum value). The metrics are correlated to each other which means that an increase in the value of one metric often

	also means a change in the value of other metrics. The metric MCC is covering all aspects together and a value of 1 means that also the precision, recall, sensitivity and negative predictive values are also at their maximum equal to 1.
Threshold	The threshold will be fitted to the data. As of now, the threshold is fitted as to maximize the MCC performance metric of the SafetyCage framework, but according to the traffic light system we want to fit the threshold based on more than one metric. NB: The fitting is based on historical data.
Visualization	Visualizations will among others be: • A set of samples x that have similar predictions errors as the sample x' with prediction y' visualized in a user-friendly way. • The quantity of the output returned from SafetyCage together with a colour (green, orange or red) which indicates the risk of each prediction. • If applicable counterfactual explanations. See model card for more details.
Card references	This section contains references to other cards
Model card(s)	SafetyCage supports four different AI models for misclassification detection and one for misclassification explanations. These methods are further elaborated in the following model cards: 1) Maximum Softmax Probability (MSP) 2) DOCTOR 3) SPARDACUS 4) Mahalanobis 5) Misclassification similarity explanations

9.1.1. Maximum Softmax Probability (MSP)

Table 24: Model card for Maximum Softmax Probability (MSP)

Card section / field	Maximum Softmax Probability (MSP)
Model details *	This section contains basic information about the model.
Contact info *	Dan Hendrycks (hendrycks@berkeley.edu) and Kevin Gimpel (kgimpel@ttic.edu)
Model date *	Published at ICLR 2017
Model version *	Initial version as described in the paper
Model type *	Statistical method for detecting misclassification using softmax prediction probabilities
Information details *	MSP (Maximum Softmax Probability) is a method to detect samples misclassified by a model with a softmax output. Serves as a Simple baseline for detecting misclassified and out-of-distribution examples in neural networks
References	-
Citation	Hendrycks, D. and Gimpel, K., 2016. A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv preprint arXiv:1610.02136.
License	-
Intended use *	This section contains information about use cases that were envisioned during development.
Primary intended uses *	The primary use case is to detect when a machine learning classifier is likely to make an error, including when an example is misclassified or out-of-distribution
Primary intended users *	The intended users are machine learning practitioners, researchers, and anyone deploying machine learning classifiers in real-world tasks where reliability and safety are concerns
Out-of-scope use cases	The paper doesn't explicitly describe out-of-scope uses, but the method is designed to detect errors and out-of-distribution samples rather than for making predictions themselves
Factors	This section provides a summary of model performance across a variety of relevant factors.
Relevant factors	Model performance can vary based on: the type of task, specific datasets used for each task, including variations in the dataset size, complexity, and domain, neural network

	architecture, parameters, and training methods, and type and magnitude of distortions or noise used to create out-of-distribution examples
Evaluation factors	Error detection in various tasks, out-of-distribution detection using different datasets. Performance was compared across different types of out-of-distribution examples (e.g. noise, realistic images from different datasets, distorted audio).
Data	This section includes information about the data used by the model.
Datasets	• **Image datasets:** MNIST, CIFAR-10, CIFAR-100, SUN, Omniglot, notMNIST**.** • **Textual datasets:** IMDB, Customer Review dataset, Movie Review dataset, 20 Newsgroups, Reuters 8, Reuters 52, Wall Street Journal (WSJ), Tweets, English Web Treebank**.** • **Audio datasets:** TIMIT, Aurora-2, THCHS-30
Feature selection	-
Feature engineering	The datasets were chosen to cover a range of tasks, complexity, and research areas.
Pre-processing	Images are resized when necessary. For the speech recognition task, MFCCs, energy, and first and second deltas of a 25ms frame are used. Standard mirroring and cropping data augmentation are used in the computer vision tasks.
Data split	The choice was to align training and testing data distribution in the in-distribution case. The training datasets are the same as those used for evaluation. The text notes that the training data was preprocessed similarly to the evaluation data.
Model specification	This section contains general information about the training of model.
Computational needs	-
Input features	-
Model Output	-
Hyper-parameter optimization	-
Optimization approach	-
Metrics *	This section contains information about appropriate metrics.
Model performance measures *	AUROC, AUPR, Mean Predicted Class Probability
Decision thresholds	Threshold for classifying a prediction as false.
Variation approaches	-
Quantitative analyses	This section should contain information about the results of evaluating the model.
Unitary results	The performance of the baseline method and the abnormality module are reported in terms of AUROC, AUPR, and mean predicted class probability on various datasets and tasks
Intersectional results	The paper does not explicitly report results for the intersection of factors. The results show how performance is affected by different tasks, datasets, and the degree of distortion in out-of-distribution examples.
Ethical considerations	This section contains information about ethical considerations that went into the model development.
Data	The paper does not explicitly mention whether sensitive data is used, but some datasets may contain demographic information (e.g., social media posts), and some data could be biased.
Human life	The models are not intended to inform decisions about matters central to human life, but the ability of a model to detect errors is important in high-stakes applications such as medicine and autonomous driving.
Mitigations	The paper focuses on detection methods, not on mitigating risks through changes in model development. The proposed methods could be used to detect when a model is unreliable and trigger a different course of action.
Risks and harms	Overconfidence in misclassification detection method.

Use cases	The paper does not specifically discuss "fraught" use cases, but indicates the value of detecting errors and out-of-distribution samples in real-world applications.
Caveats and recommendations	This section should list additional concerns that were not covered as part of the ethical considerations.
Description	The softmax prediction probability is a misleading confidence proxy when viewed in isolation. The abnormality module outperforms the baseline method in some cases, suggesting the potential for further research. The paper recommends that detection methods should be tested across a variety of tasks and architectures and that AUROC and AUPR should be reported. The paper also recommends that researchers use their own architectures and datasets when comparing new methods to the softmax baseline

9.1.2. DOCTOR

Table 25: Model card for DOCTOR

Card section / field	DOCTOR
Model details *	This section contains basic information about the model.
Contact info *	Federica Granese (federica.granese@inria.fr), Marco Romanelli (marco.romanelli@centralesupelec.fr), Daniele Gorla (gorla@di.uniroma1.it), Catuscia Palamidessi (catuscia@lix.polytechnique.fr), Pablo Piantanida (pablo.piantanida@centralesupelec.fr)
Model date *	Presented at NeurIPS 2021.
Model version *	Not specified in the text.
Model type *	DOCTOR is a discriminator designed to detect whether a prediction from a deep neural network (DNN) classifier should be trusted. It uses soft-probabilities from the classifier's output.
Information details *	DOCTOR aims to identify whether a classifier's prediction should be trusted, regardless of whether the input is in-distribution or out-of-distribution. It is applicable to any pre-trained model. It does not require prior information about the underlying dataset. DOCTOR is implemented in two scenarios: Totally Black Box (TBB, where only soft predictions are available) and Partially Black Box (PBB, input perturbations are allowed). DOCTOR uses a discriminator based on the softmax probability, approximating the unknown error probability function.
References	-
Citation	Granese, F., Romanelli, M., Gorla, D., Palamidessi, C. and Piantanida, P., 2021. Doctor: A simple method for detecting misclassification errors. *Advances in Neural Information Processing Systems*, *34*, pp.5669-5681.
License	-
Intended use *	This section contains information about use cases that were envisioned during development.
Primary intended uses *	To identify whether a prediction from a DNN classifier should be trusted so that, consequently, it would be possible to accept it or to reject it. It is also intended for detecting misclassification errors.
Primary intended users *	Machine learning practitioners, researchers, and non-specialists who use DNNs as "black boxes" and need to assess the reliability of the model.
Out-of-scope use cases	DOCTOR is not designed for direct classification or prediction; it is intended for use as a method to validate the reliability of existing classifiers.
Factors	This section provides a summary of model performance across a variety of relevant factors.
Relevant factors	Type of dataset (image, text), specific datasets used for each task such as CIFAR10, CIFAR100, TinyImageNet, SVHN, IMDb, AmazonFashion, AmazonSoftware. The architecture of the underlying pre-trained model, and the presence of out-of-distribution samples.

Evaluation factors	Performance on in-distribution and out-of-distribution samples, performance in both TBB and PBB scenarios, comparison with other state-of-the-art methods like ODIN, softmax response, and Mahalanobis distance. The trade-off between Type-I and Type-II errors, and he performance of the Dα and Dβ discriminators.
Data	This section includes information about the data used by the model.
Datasets	• **Image datasets:** CIFAR10, CIFAR100, TinyImageNet, SVHN • **Textual datasets**: IMDb, AmazonFashion, AmazonSoftware
Feature selection	-
Feature engineering	These datasets were chosen to evaluate DOCTOR on both image and textual data and to compare its performance with other state-of-the-art methods. Datasets with varying accuracy levels were chosen to evaluate the method in different scenarios. In PBB, input samples are perturbed using gradients. Temperature scaling was applied in PBB when appropriate.
Pre-processing	DOCTOR is designed to work with pre-trained models without needing additional training data, making it flexible and easy to deploy. Since DOCTOR uses pre-trained models, no additional training pre-processing is required.
Data split	DOCTOR does not require training data. It is applied to pre-trained models.
Model specification	This section contains general information about the training of model.
Computational needs	-
Input features	-
Model Output	-
Hyper-parameter optimization	-
Optimization approach	-
Metrics *	This section contains information about appropriate metrics.
Model performance measures *	False rejection rate (FRR), true rejection rate (TRR), AUROC, FRR at 95% TRR.
Decision thresholds	Threshold for classifying a prediction as false.
Variation approaches	-
Quantitative analyses	This section should contain information about the results of evaluating the model.
Unitary results	Results include AUROC, FRR, and FRR at 95% TRR for different datasets and scenarios (TBB and PBB), for different methods, as shown in Table 1 and Table 2. Results show performance comparisons between Dα and Dβ. Results also show the performance of DOCTOR in comparison with other state-of-the-art methods.
Intersectional results	The experimental results compare performance across different datasets, architectures, and TBB/PBB scenarios, and under different parameter settings, including the addition of noise. The comparison between different methods and parameter choices is detailed in the experiments.
Ethical considerations	This section contains information about ethical considerations that went into the model development.
Data	The paper does not specifically discuss ethical implications of data usage, but the datasets themselves may contain biases.
Human life	The paper highlights the importance of detecting misclassifications in critical systems such as autonomous driving vehicles or industrial robots. DOCTOR is intended to improve the reliability of AI systems in such situations.
Mitigations	DOCTOR helps mitigate the risk of using unreliable classifiers by identifying predictions that should be rejected. The method can be used to avoid acting on erroneous predictions in sensitive or critical situations.
Risks and harms	Overconfidence in misclassification detection method.

Use cases	The paper does not mention specific use cases that are especially fraught, but the method is designed for all cases where the predictions of a classifier need to be evaluated for trustworthiness, in safety critical applications as well as more general tasks.
Caveats and recommendations	This section should list additional concerns that were not covered as part of the ethical considerations.
Description	DOCTOR uses all the information in the softmax output, which results in equal or better performance with respect to the other methods. The results in PBB, where a reduction in the false rejection rate is observed, are promising. DOCTOR does not require training data and can be easily deployed in real-world scenarios. DOCTOR does not exploit information across the layers, only soft-predictions are used. The calibration of the threshold (γ) between the desired fault rejection and acceptance rates is an important consideration that would require additional validation samples. Future work should investigate the extension of DOCTOR to white-box scenarios, combining the two discriminators and using information across different latent codes of the model.

9.1.3. Mahalanobis

Table 26: Model card for Mahalanobis

Card section / field	Description
Model details *	This section contains basic information about the model.
Contact info *	Pål Vegard Johnsen (pål.johnsen@sintef.no)
Model date *	2024
Model version *	Not specified
Model type *	Statistical inference framework for detection of misclassifications
Information details *	The method constructs a **probability distribution of pre-activation values** for correctly classified samples from the training set. A **hypothesis test** is performed to determine if a new sample originates from the same distribution, using pre-activation values from the neural network. The **Mahalanobis distance** is calculated, assuming a multivariate Gaussian distribution of the pre-activation random vector **Z**. **U = (Z − μ)TΣ−1(Z − μ)**, where **μ** is the mean and **Σ** is the covariance matrix. The SafetyCage is trained on the correctly classified samples in the training set of the classifier. The **p-value** is calculated as **p = 1 − Fχ2q(u)**, where **Fχ2q** is the cumulative of the Chi-squared distribution with **q** degrees of freedom, and **u** is the realization of the Mahalanobis distance.
References	-
Citation	-
License	-
Intended use *	This section contains information about use cases that were envisioned during development.
Primary intended uses *	To flag wrong or suspicious classifications.
Primary intended users *	Machine learning specialists, researchers, and end-users who may need to assess the reliability of ML model predictions.
Out-of-scope use cases	Designed for in-distribution detection of wrong classification.
Factors	This section provides a summary of model performance across a variety of relevant factors.
Relevant factors	The architecture of the neural network, datasets used to train and test the model, performance of the underlying classifier, choice of layer(s) used for misclassification detection, c**hoice of significance level α**.
Evaluation factors	Performance in detecting misclassifications on both well-performing and under-performing classifiers, comparison with other misclassification detection methods, the impact of using

	different layers or combinations of layers for detection, the sensitivity of the method to the **choice of threshold (α)**.
Data	This section includes information about the data used by the model.
Datasets	MNIST and CIFAR-10.
Feature selection	None.
Feature engineering	None.
Pre-processing	Standardizing image data to have a mean of zero and unitary standard deviation over each dataset partition (train and test).
Data split	66 % train data, 33 % test data.
Model specification	This section contains general information about the training of model.
Computational needs	The computational complexity of the algorithm (computational resources required for training/inference, time elapsed during training) is not specified in the excerpts.
Input features	Values from the dataset. Specifically, the pre-activation values **z** from the hidden layers of a feed-forward neural network.
Model Output	Prediction of whether a classification is wrong or suspicious, based on whether the pre-activation values **z*** originated from the probability distribution corresponding to **Ω(tr)ŷ***.
Hyper-parameter optimization	The significance level α can be optimized to maximize the MCC score.
Optimization approach	Details not provided in the excerpts.
Metrics *	This section contains information about appropriate metrics.
Model performance measures *	Matthews correlation coefficient (MCC), Precision, Recall, Specificity, Negative predictive value (NPV).
Decision thresholds	**Significance level α**, which affects the trade-off between precision and recall. Optimal α values are determined empirically to maximize MCC.
Variation approaches	-
Quantitative analyses	This section should contain information about the results of evaluating the model.
Unitary results	The SafetyCage achieves high recall but low precision. Performance is better on well-performing classifiers (MNIST) compared to poorly performing ones (CIFAR-10). Using the empirical cumulative distribution function (ECDF) or **χ2-asymptotic** yields comparable results. Performance varies depending on the layer(s) used for misclassification detection (penultimate, hidden, all) and the p-value combination test used (Fisher or Cauchy).
Intersectional results	The underlying classifier performance: The SafetyCage performs better with well-performing classifiers (e.g., MNIST) compared to poorly performing ones (e.g., CIFAR-10). Choice of layers: Performance varies depending on the layer(s) used for misclassification detection (penultimate, hidden, all). P-value combination test: The choice of the p-value combination test (Fisher or Cauchy) impacts the results when multiple layers are used. Choice of threshold: The sensitivity of the method to the choice of threshold α.
Ethical considerations	This section contains information about ethical considerations that went into the model development.
Data	No.
Human life	Depends on the use-case.
Mitigations	None.
Risks and harms	Over-reliability. This model can itself be wrong.
Use cases	-
Caveats and recommendations	This section should list additional concerns that were not covered as part of the ethical considerations.

Description	• **Distribution of pre-activation values:** The distribution of pre-activation values may deviate from a Normal distribution, which is an assumption of the Mahalanobis distance calculation. • **Overlapping distributions:** Distributions of correctly and wrongly predicted samples may overlap, especially with less-performing classifiers. • **Hypothesis test and distance measure:** Consider reformulating the hypothesis test or using a different distance measure to improve performance. • **Online learning:** The significance level α can be learned online with new data to maximize performance. • **Generalization:** The method can be generalized to other feed-forward layers and architectures.

9.1.4. SPARDACUS

Table 27: Model card for SPARDACUS

Card section / field	SPARDACUS
Model details *	This section contains basic information about the model.
Contact info *	Pål Vegard Johnsen (pal.johnsen@sintef.no)
Model date *	Presented at NLDL 2025.
Model version *	Not specified in the text.
Model type *	Uses a Sparse Difference Analysis (SPARDA) approach to project the activation values at each layer into one dimension and then uses the Wasserstein distance between the probability distribution functions (PDFs) (estimated via Gaussian mixture models) of correctly and wrongly classified samples to compute p-values which are then used to flag possible misclassifications.
Information details *	SPARDACUS is a method for detecting unreliable machine learning predictions across neural network layers by using SPARDA to maximize the Wasserstein distance between correct and incorrect distributions. It computes layer-specific p-values using the Cauchy combination test, without relying on pre-activation values or Gaussianity assumptions. With two statistical approaches (SC and SW), SPARDACUS extends the MSP-detector to provide a statistically rigorous technique for identifying potentially problematic predictions in safety-critical applications.
References	-
Citation	Johnsen, P.V., Remonato, F., Benedict, S. and Ndur-Osei, A., 2024. SPARDACUS SafetyCage: A new misclassification detector. *Northern Lights Deep Learning Conference 2025*. Available at: https://openreview.net/forum?id=3FswRo4Lhj
License	-
Intended use *	This section contains information about use cases that were envisioned during development.
Primary intended uses *	To detect and flag potential misclassifications from feed-forward neural networks, especially in situations where reliability is critical. This includes feature extraction and using intermediate layers in a neural.
Primary intended users *	Machine learning specialists, researchers, and end-users who may need to assess the reliability of ML model predictions.
Out-of-scope use cases	SPARDACUS is specifically designed for misclassification detection with in-distribution data and is not intended for out-of-distribution (OOD) detection.
Factors	This section provides a summary of model performance across a variety of relevant factors.
Relevant factors	The architecture of the neural network, including the number of layers and neurons. The specific datasets used to train and test the model (e.g., MNIST, CIFAR-10). The performance of the underlying classifier (e.g., well-performing vs. under-performing)). The choice of layer(s) used for misclassification detection (output layer, hidden layers, etc.). The choice of

	the statistic used (SC or SW), the threshold values αC and αW, and the regularization parameter λ used in the SPARDA projection.
Evaluation factors	Performance in detecting misclassifications on both well-performing and under-performing classifiers. Comparison with other misclassification detection methods (e.g., DOCTOR, MSP-detector, Mahalanobis SafetyCage). The impact of using different layers or combinations of layers for detection. The sensitivity of the method to the choice of threshold.
Data	This section includes information about the data used by the model.
Datasets	• **Image datasets**: MNIST and CIFAR-10
Feature selection	-
Feature engineering	Datasets were chosen to evaluate the method on both a well-performing and a poorly performing classifier. They are also established benchmarks for machine learning.
Pre-processing	Standardising image data. Mean zero and std unitary, over each dataset partition (train, and test).
Data split	The same datasets used for evaluation (MNIST and CIFAR-10) are used to train the underlying neural network classifier. SPARDACUS itself is trained on the activation values of the trained classifier.
Model specification	This section contains general information about the training of model.
Computational needs	-
Input features	-
Model Output	-
Hyper-parameter optimization	-
Optimization approach	-
Metrics *	This section contains information about appropriate metrics.
Model performance measures *	Matthews correlation coefficient (MCC), Precision, Recall, Specificity, Negative predictive value (NPV).
Decision thresholds	Threshold for classifying a prediction as false. Chosen to maximize a user specific performance metric.
Variation approaches	-
Quantitative analyses	This section should contain information about the results of evaluating the model.
Unitary results	Results are presented for different values of αC and αW. Tables show precision, recall, specificity, NPV, and MCC values for SPARDACUS, SafetyCage, DOCTOR, and MSP-detector methods
Intersectional results	• Comparison between different misclassification detection methods on the MNIST and CIFAR-10 datasets are provided. • The performance of SPARDACUS is evaluated using different layer aggregations. The results when using the SC and SW statistics are compared. • Results when parameters are estimated on the training set and test sets are presented. SPARDACUS is on par with MSP and DOCTOR, and outperforming Mahalanobis Safetycage.
Ethical considerations	This section contains information about ethical considerations that went into the model development.
Data	The paper does not discuss specific ethical considerations related to data use.
Human life	The paper emphasizes the relevance of misclassification detection for safety-critical applications. The method aims to improve the reliability of AI systems in such scenarios.
Mitigations	SPARDACUS helps mitigate the risk of using unreliable classifiers by identifying and flagging potential misclassifications. This reduces the risk of the model making erroneous predictions

Risks and harms	Overconfidence in misclassification detection method.
Use cases	The method is intended for use cases where model reliability is critical, such as autonomous driving, medical diagnosis, and other safety-critical applications.
Caveats and recommendations	This section should list additional concerns that were not covered as part of the ethical considerations.
Description	SPARDACUS demonstrates significant performance in misclassification detection, outperforming the Mahalanobis SafetyCage method and comparable to MSP-detector and DOCTOR methods. Its key strengths include flexibility across neural network layers, with the output layer being most valuable. While sensitive to threshold selection, SPARDACUS can be updated for new data and used to investigate classification errors. Future research should explore alternative statistical distances, optimization algorithms, and inner layer analysis.

9.1.5. Misclassification similarity explanations

Table 28: Model card for misclassification similarity explanations

Card section / field	Misclassification similarity explanations
Model details *	This section contains basic information about the model.
Contact info *	• Pål Vegard Johnsen (pal.johnsen@sintef.no) • Joel Bjervig (joel.bjervig@sintef.no)
Model date *	18.02.2025
Model version *	1
Model type *	K-nearest neighbours.
Information details *	Given AI-model with input x and corresponding class prediction i. If the output from SafetyCage is that the prediction is flagged as incorrect, one can use this model to help the user to further evaluate whether this prediction is likely to be wrong. This is done by historically looking at inputs x_1,…,x_k with the same wrong prediction. Using the K-nearest neighbour (KNN) algorithm, the model will provide examples to the user that are the most similar to the input x. For complex input data such as images or time series, and if the AI-model is a neural network, the KNN algorithm can be applied in one of the embedding layers.
References	-
Citation	Cover, Thomas M.; Hart, Peter E. (1967). "Nearest neighbor pattern classification". IEEE Transactions on Information Theory. 13 (1): 21–27
License	-
Intended use *	This section contains information about use cases that were envisioned during development.
Primary intended uses *	The method is inherently created for classification models and not regression models. However, adjustments are possible such that the method also is applicable for regression models
Primary intended users *	End-users
Out-of-scope use cases	Description of uses that the model should not be applied to.
Factors	This section provides a summary of model performance across a variety of relevant factors.
Relevant factors	KNN may not perform well if the dimension space is too large (curse of dimensionality). The result is sensitive to the choice of k. Choosing a too small k can give an overrepresentation, while choosing a too large k may give too large variation among the neighbours. The algorithm also depends on the choice of similarity metric. Most often the Euclidian distance is used, but depending on the use case, other well-known metrics or even a custom-made metric may be more suitable.
Evaluation factors	Metrics to report can be average distance to the k nearest neighbour, or the distance to the kth neighbour. Both are measures for degree of similarity.

Data	This section includes information about the data used by the model.
Datasets	Use case specific.
Feature selection	Use case specific.
Feature engineering	Use case specific.
Pre-processing	Use case specific.
Data split	Use case specific.
Model specification	This section contains general information about the training of model.
Computational needs	For one particular data point, the algorithm scales as O(k) increasing linearly as the number of k nearest neigbours.
Input features	For neural network models, the input can be in any of the embedding layers of the neural network instead.
Model Output	The prediction is the k nearest neighbours
Hyper-parameter optimization	Hyperparameters will be connected to the choice of similarity metric. This is an unsupervised method.
Optimization approach	Irrelevant as this is an unsupervised method
Metrics *	This section contains information about appropriate metrics.
Model performance measures *	See evaluation factors
Decision thresholds	-
Variation approaches	-
Quantitative analyses	This section should contain information about the results of evaluating the model.
Unitary results	-
Intersectional results	-
Ethical considerations	This section contains information about ethical considerations that went into the model development.
Data	-
Human life	-
Mitigations	-
Risks and harms	-
Use cases	-
Caveats and recommendations	This section should list additional concerns that were not covered as part of the ethical considerations.
Description	-

9.2. OODGuard – Out-of-distribution (OOD) detection

Table 29: Method card for OODGuard – Out-of-distribution (OOD) detection

Card section / field	OODGuard – Out-of-distribution (OOD) detection
Method Details *	This section contains basic information about the method.
Method *	Out-of-distribution (OOD) detection
Method name *	OODGuard
Description *	Given AI model with input x and prediction y', evaluate whether input x and output y' is similar to data (X_train,y_train) used to train the AI model. If not, it is an out-of-distribution

	sample, and we should not trust the corresponding prediction. This approach is based on the well-known fact that AI models are not behaving particularly good for extrapolated data.
Trustworthiness characteristic *	Robustness
Limitations *	The method is inherently to be used for evaluating the robustness of the model for the particular prediction y'. However, note there is a direct link to the assessment of accuracy, because the robustness is with respect to the prediction, and hence lack of robustness to the prediction means lack of accuracy.
Limitations keywords	-
Conflicts	Text-based conflict description.
Conflicts keywords	-
Model-based *	Yes
Software available *	Not yet. A first version of the software is planned released during 2025.
Contact info *	• Pål Vegard Johnsen (pal.johnsen@sintef.no) • Joel Bjervig (joel.bjervig@sintef.no)
Additional info	-
Stage details	This section contains information about the stages.
Stage	Design
Sub-process	Feature engineering
Characteristics	Model centric, data centric
AI actors	ML engineer
Entity dimension	Model and Data
Additional info	-
Software details	This section contains details about the available software code.
Code reference	-
Code repository *	• Private repositories at SINTEF's GitLab: (Access given to THEMIS 5.0 members on request) ○ https://gitlab.sintef.no/themis-sintef/themis-accuracy-robustness ○ https://gitlab.sintef.no/aai-internal/OODGuard
Input data *	Necessary input data is the same as the necessary input for the SafetyCage method: Namely the AI-model that can be loaded in a python environment and that can be called with .predict(x) given input x and output the corresponding prediction y_hat. Moreover, historical data used to train the AI-model must also be available.
Needed entities	1) AI-model as a python object; 2) Historical data used to train the AI model; and 3) During deployment, input data.
Output *	A single real value that quantifies the certainty that a sample is OOD. Included is a flagging procedure that flags a sample input to an AI-model as an OOD sample whenever value is smaller/larger than a given threshold.
Explanations *	This section contains explanation about the output.
Output meaning *	Given input x to AI-model, a variable that quantifies how far input x is from the historical data used to train the AI-model. For a sample that is flagged as being an OOD-sample, and if applicable, the output will also provide explanations as to why the sample is flagged as OOD for instance by referring to input variables that are far from the historical distribution function during training of the AI-model. If applicable, counterfactual explanations will be provided to further assess the robustness of the prediction.

Metrics *	Precision, recall, sensitivity, negative predictive value and Matthews correlation coefficient (MCC).
Metrics meaning *	The metrics measure the performance of the OODGuard software on historical data. This means, the degree to which the OODGuard is correctly flagging a prediction from an AI-model as OOD or not. The metrics evaluate different aspects of performance. Metrics to evaluate OOD-detection can be tricky in real life applications where so-called near-OOD samples are likely to occur more frequently than OOD-samples that can easily be regarded irrelevant for an AI-model, so-called far-OOD. To assess the OOD-method from historical samples, end-users would need to manually accept a sample as OOD, and hence not as a so-called in-distribution sample that is considered relevant for the trained AI-model. The definition of an OOD-sample may be difficult to grasp, but one way to define it is that the AI-model both turned out to give a wrong prediction AND that the sample possesses some characteristics that the AI-model should not be able to account for. If so, we can utilize the same metrics as those presented for the SafetyCage method: Precision, recall, sensitivity, negative predictive value and Matthews correlation coefficient.
Ideal values *	-
Threshold	The threshold is fitted to the historical data. The fitting procedure will be based on the assumption that ODD-samples will provide poor predictions.
Visualization	Histograms or density plots that can help the user understand why an input sample can be considered OOD with respect to the raw input data which is interpretable for the end-user.
Card references	This section contains references to other cards
Model card	OODGuard will include support for the following AI models: 1) Counterfactual explanations

9.2.1. Counterfactual explanations

Table 30: Model card for counterfactual explanations

Card section / field	Counterfactual explanations
Model details *	This section contains basic information about the model.
Contact info *	• Pål Vegard Johnsen (pal.johnsen@sintef.no) • Joel Bjervig (joel.bjervig@sintef.no)
Model date *	20.02.2025
Model version *	1
Model type *	Counterfactual explanations
Information details *	Given AI-model with input x and corresponding class prediction i. We seek for perturbations on x, giving x' such that the prediction changes from class prediction i to another pre-specified class prediction j. We want the difference between x and x' to be as small as possible. An option is to seek for several perturbed input samples x'_1,…m x'_K all giving the same changed class prediction j. The difference between x and the perturbed sample x'_k, and what is the cause of the difference, is a tool for measuring the robustness of the model.
References	-
Citation	Counterfactual explanations and how to find them: literature review and benchmarking, R Guidotti, Data Mining and Knowledge Discovery 38 (5), 2770-2824
License	-
Intended use *	This section contains information about use cases that were envisioned during development.
Primary intended uses *	The method is inherently created for classification models and not regression models.
Primary intended users *	End-users

Out-of-scope use cases	Not created for regression models
Factors	This section provides a summary of model performance across a variety of relevant factors.
Relevant factors	There are many different algorithms for finding counterfactual explanations. The model performance depends on several aspects, among them: The degree to which the perturbed sample gives the desired prediction class How to measure the similarity between input samples; the degree to which the perturbed x' is realistic and within the previously observed data manifold; how small the difference between the original sample and the perturbed sample is; how easy it is to understand the difference between x and x' (interpretability)
Evaluation factors	Metrics to report is whether perturbed sample gives desired class prediction. The similarity measure between the original sample and the perturbed sample. A measure for the degree to which the perturbed samples is within the data manifold (algorithms for out-of-distribution detection can be used for this).
Data	This section includes information about the data used by the model.
Datasets	Use case specific.
Feature selection	Use case specific.
Feature engineering	Use case specific.
Pre-processing	Use case specific.
Data split	Use case specific
Model specification	This section contains general information about the training of model.
Computational needs	Computational resources depends on the algorithm used, but all algorithms have in common that they are optimization algorithms which are non-convex and hence hard to solve. Time required to run the algorithm will hence also depend on the number of iterations pre-specified during the optimization search.
Input features	The input to the ML model, x.
Model Output	The output is the perturbed samples x'_1,…,x'_K.
Hyper-parameter optimization	Hyperparameters will depend on the counterfactual explanation algorithm chosen. There are typically several hyperparameters involved.
Optimization approach	Can vary.
Metrics *	This section contains information about appropriate metrics.
Model performance measures *	See evaluation factors.
Decision thresholds	-
Variation approaches	-
Quantitative analyses	This section should contain information about the results of evaluating the model.
Unitary results	-
Intersectional results	-
Ethical considerations	This section contains information about ethical considerations that went into the model development.
Data	-
Human life	-
Mitigations	-
Risks and harms	-
Use cases	-
Caveats and recommendations	This section should list additional concerns that were not covered as part of the ethical considerations.
Description	-

9.3. FairnessEvaluator

Table 31: Model card for the FairnessEvaluator

Card section / field	FairnessEvaluator
Model details *	This section contains basic information about the model.
Contact info *	George Karavokiros (george.karavokiros@maggioli.gr)
Model date *	2024-09-05 (indicative release date)
Model version *	1.0 (initial release within the THEMIS 5.0 framework).
Model type *	Statistics-based fairness evaluation tool.
Information details *	A Python-based fairness assessment component that leverages AI Fairness 360 (AIF360) to evaluate biases in ML models' predictions using statistical fairness metrics. Provides metrics such as Disparate Impact, Statistical Parity Difference, Equal Opportunity Difference, and others that compare outcomes for privileged vs. unprivileged groups.
References	AI Fairness 360 documentation, IBM Research, relevant academic work in algorithmic fairness.
Citation	Bellamy, R. K. E., et al. 2019. "AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias." IBM Research.
License	Apache 2.0 (aligned with AI Fairness 360 licensing).
Intended use *	This section contains information about use cases that were envisioned during development.
Primary intended uses *	To assess fairness by measuring potential bias in binary classification outputs; supports human-cantered decision-making and regulatory compliance efforts. Designed to be integrated with the Quantitative Assessor to provide a comprehensive evaluation of fairness alongside other trustworthiness metrics such as robustness and accuracy.
Primary intended users *	ML engineers, data scientists, fairness researchers, compliance officers, and any stakeholders interested in identifying and mitigating unfair outcomes.
Out-of-scope use cases	Direct bias mitigation; multi-class or regression tasks without additional customisation; purely rule-based or fully causal fairness analysis.
Factors	This section provides a summary of model performance across a variety of relevant factors.
Relevant factors	Data quality, choice of protected attributes, model thresholding, potential trade-offs between accuracy and fairness.
Evaluation factors	Disparities in statistical measures (e.g., error rates, selection rates) across protected groups.
Data	This section includes information about the data used by the model.
Datasets	User-provided datasets in CSV or pandas DataFrame format containing at least one protected attribute, binary ground truth labels, and model predictions.
Feature selection	Not performed; relies on already selected features from the user's predictive model.
Feature engineering	Not performed; tool focuses on fairness metrics, not data transformation beyond labeling/encoding.
Pre-processing	Maps positive/negative outcomes to 1/0; label-encodes categorical features; identifies privileged vs. unprivileged groups.
Data split	Not applicable.
Model specification	This section contains general information about the training of model.
Computational needs	Low to moderate CPU usage for running metrics; no GPU needed since no model training occurs.
Input features	Ground truth label, predicted label, protected attribute(s).
Model Output	Fairness metrics and a structured fairness report summarizing disparities across protected groups.

Hyper-parameter optimization	Not applicable; the FairnessEvaluator does not train or optimize models.
Optimization approach	Not applicable.
Metrics *	This section contains information about appropriate metrics.
Model performance measures *	Disparate Impact, Statistical Parity Difference, Equal Opportunity Difference, Equalized Odds, and other group fairness metrics included in AIF360.
Decision thresholds	Determined outside the FairnessEvaluator; the user must supply the predicted labels at a chosen threshold.
Variation approaches	Users can evaluate multiple thresholds or different models by repeatedly running the FairnessEvaluator.
Quantitative analyses	This section should contain information about the results of evaluating the model.
Unitary results	Generates numerical fairness metrics and differences/ratios indicating potential biases.
Intersectional results	Possible by defining multiple protected attributes separately and comparing results (though not fully automated).
Ethical considerations	This section contains information about ethical considerations that went into the model development.
Data	The FairnessEvaluator was developed with the ethical principle that fairness assessments depend on high-quality, representative data, ensuring that bias detection is meaningful and not distorted by incomplete or skewed datasets.
Human life	Recognizing the potential impact of biased AI decisions on individuals, the tool was designed to support fairness evaluations in critical domains like healthcare, hiring, and criminal justice, where fairness can have real-world consequences.
Mitigations	Provides insights to guide rebalancing data, changing models, or adjusting thresholds but does not directly perform bias mitigation.
Risks and harms	Ethical considerations included the risk of fairness-accuracy trade-offs and the potential for misinterpreting fairness metrics, which could lead to unintended negative consequences if used without proper domain expertise.
Use cases	Evaluating the fairness of AI models in liver cancer prediction
Caveats and recommendations	This section should list additional concerns that were not covered as part of the ethical considerations.
Description	Not a substitute for holistic bias control; recommended to combine results with other interpretability and risk assessment tools.

10. ANNEX 3: USER REQUIREMENTS RESULT FROM LIVING LABS

In D2.1 a list of user requirements based on co-creation phase A and interviews by ATC had generated a list of 81 user requirements. These user requirements have further been developed and investigated through their involvement in co-creation phase B, Living Labs.

The Living Labs took place in four European countries, Denmark, Bulgaria, Spain and Greece. A combined total of 79 people were involved across the four countries.

The participants were placed in mixed groups based on their belonging to different sectors (health, media and port management). Seven user requirements had been formulated based on the list from D2.1. At each workshop the participants were presented with the foundational thoughts behind the THEMIS 5.0 system. The participants were divided into smaller groups and was asked to rank the seven broadly formulated user requirements, from 1 (highest priority) to 7 (lowest priority) in relation to which user requirements they deemed the highest priority for their use of the THEMIS 5.0 system. The user requirements were intentionally formulated broadly, as the second step of the assignment was for the participants to give input on how they ideally would want the requirements to be implemented.

The seven user requirements that the participants were presented was the following, here presented in no particular order.

Access to the service: I want the AI to be available on my preferred device

Personalization: I want the AI to be tailored to my personal needs and professional profile

Presentation of results: I want the AI to show me its results in a user-friendly way

Length of interaction with chatbot: I want the interaction with the AI chatbot to be an appropriate length

User types: I want the AI to have different user types related to different organizational levels

Explainability: I want the AI to be able to explain its reasoning for the suggested results

Language: I want the interaction with the AI chatbot to use language that is user-friendly

10.1. Presentation of results on user requirements

The score of the user requirements were calculated by adding the priority they were given by each group, meaning that the lower the score, the higher it had been prioritised.

Table 32: User requirements score

Priority	User requirement	Accumulated score
1	Explainability	32
2	Presentation of results	33
3	Language	39
4	Access to service	40
5	Personalization	44
6	Length of interaction with chatbot	74*
7	User types	74*

**'Length of interaction with chatbot' and 'user types' has the same accumulated score, and both were ranked as priority 7 by five groups, but 'user types' were ranked as priority 6 by five groups, whereas 'length of interaction with chatbot' was ranked as priority 6 by four groups. Therefore, 'user types' was placed at priority 7 since it got the highest number of low priorities.*

10.2. Elaboration on recommendations for implementation of user requirements

Priority 1: Explainability

The chatbot should always provide reasoning for its output, including source descriptions and access links. Users should be able to ask about the credibility of the sources and how they were selected. Additionally, the chatbot should offer explainability through both text descriptions and other media, such as infographics or numbers. The chatbot must allow users to trace the steps taken by the AI to generate a response and be able to engage in a conversation to understand the AI's reasoning process.

Priority 2: Presentation of results

The chatbot's result presentation should be customizable based on user preferences, allowing choices between text, visuals, summaries, bullet points, tables, or infographics. The AI should learn from interactions to optimize presentation formats while ensuring clarity, coherence, and accessibility. Users should be able to distinguish AI-generated content from sourced information, and responses should be concise, reliable, and adaptable to different professional needs, languages, and contexts.

Priority 3: Language

The chatbot should support multiple languages, allowing users to choose their native or preferred language and request translations when needed. It must adapt to sector-specific terminology while offering the option to simplify language for clarity for individual user's needs. The AI should balance technical accuracy with plain, concise wording to avoid ambiguity and ensure accessibility. Personalization is key, enabling the chatbot to tailor its language based on user expertise and context to enhance understanding and prevent misunderstandings.

Priority 4: Access to service

The AI service should be accessible across multiple devices, including smartphones, PCs, and older-generation hardware, ensuring consistency in functionality and user experience. It must support shared access for professionals using common devices. The system should be open-source, multiplatform, easy to use, and responsive. Additionally, for niche industries, compatibility with specialized devices is necessary to optimize workflow efficiency.

Priority 5: Personalization

The chatbot should offer customizable user profiles that adapt dynamically based on interactions while allowing manual adjustments. Users can choose from predefined profiles or personalized settings to match their writing style, coding preferences, and industry needs. The system should balance flexibility with ease of use, learning user preferences over time while maintaining core concepts for consistency. It should also recognize whether a user prefers visual or text-based outputs to optimize the experience accordingly.

Priority 6: Length of interaction with chatbot

The chatbot should begin with an in-depth initial interaction to gather user preferences and ensure accurate personalization and recommendations. It should be clear for the user that spending more time on this setup improves the quality of future outputs.

Priority 7: User types

While the system should be widely accessible, access to sensitive information should be restricted when necessary to ensure security and privacy.

11. ANNEX 4: UPDATED SYSTEM & USER REQUIREMENTS

In this Annex, the updated System & User Requirements that were elaborated in the framework of T4.1 based on the outputs documented in the deliverable D2.1. are presented.

11.1. System Requirements

In D2.1, a comprehensive set of (55) system requirements are documented, originating from THEMIS 5.0 DoA that were analysed for overlaps, consolidated and summarised to produce a set with 28 system requirements, distributed to the four stages of the Trustworthiness Optimization Process (as described in D2.1 and D2.2 in detail). In the framework of T4.1, the consortium discussed extensively the System Requirements and further elaborated, summarised and defined the list of **28 System Requirements** as presented in the following table.

Table 33: System Requirements

ID	System Requirement	Source	Comments
OVERALL			
SR.1	[THEMIS 5.0 should be] … a micro-service-based platform supporting cycles of design, deployment, evaluation, and tuning of experiment variants of AI systems. These services will be cloud-based.	SR_12	-
SR.2	[THEMIS 5.0 should] …. interact with the user via an AI-driven conversational agent.	SR_01	-
SR.3	[THEMIS 5.0 should] … be able to explain in the necessary level of detail the outputs of AI systems.	SR_01	*While for SUTs this is not possible, outputs of the THEMIS 5.0 models (i.e. business impact, TW preferences identification) need to be presented and if needed explained to the users. The 'Conversational Agent for Human-in-the-Loop' will be able to indicate the characteristics that had a key role in the identification of the Users' persona and moral orientation. (Row 2, Row 19]*
SR.4	[THEMIS 5.0 should] … facilitate dialogues adjusted to users' preferences and traits.	SR_07	*Dialogues with the user will provide the input to the 'Conversational Agent for Human-in-the-Loop' to estimate their TW preferences. In some cases and depending on the input with regards to their traits and preferences, additional questions may be needed. [Row 19]*
SR.5	[THEMIS 5.0 should] … comply with the European legal and ethical framework	SR_03	*THEMIS 5.0 Ethical and Legal Assessment will provide guidance on how to comply with the European legal and ethical framework. [Row 18]*
IDENTIFY			
SR.6	[THEMIS 5.0 should] … capture users' role (job role) and *persona*, i.e., the preferences, requirements, objectives capabilities, motives and behavioural patterns, decision support needs, and legal, moral and ethical principles.	SR_14 SR_21 SR_22	*THEMIS 5.0 will capture the parameters that affect the users' TW preferences according to the established methodology.*

SR.7	[THEMIS 5.0 should] … capture business objectives and KPIs that the optimised AI system will need to support.	SR_22	*To be included in the socio-technical model.*
SR.8	[THEMIS 5.0 should] … enable the user via a GUI and a chatbot to create a qualitative model of the socio-technical environment using a combination of high-level knowledge specification and machine learning. In this GUI the user will insert: all the possible actions that the AI system can recommend, all the KPIs that might be affected by any of the possible actions, any external factors that might influence the actions or the actions' effect on the KPIs, and how the actions and the external factors affect the KPIs through pairwise relations.	SR_18 SR_19 SR_24 SR_47	*- (Row 5)*
	Generate a qualitative model based on user input	SR_24	Same as above? Consider to be omitted.
SR.9	[THEMIS 5.0 should] … generate a quantitative model or a simulation model from the qualitative model.	SR_24	-
SR.10	[THEMIS 5.0 should] … receive feedback from the users concerning their trust in the socio-technical model.	SR_47	*- (Row 6)*
SR.11	[THEMIS 5.0 should] … update either the qualitative or the quantitative model of the socio-technical environment based on user feedback.	SR_47	*- (Row 6)*
SR.12	[THEMIS 5.0 should] … enable the user to model an AI system in the context of a socio-technical environment in accordance with decision support needs and moral values.	SR_56	*To be included in the socio-technical model.*
ASSESS			
SR.13	[THEMIS 5.0 should] … perform a personalised AI Trustworthiness assessment of the AI system based on the knowledge captured in the IDENTIFY stage.	SR_13 SR_14 SR_54 SR_20	-
SR.14	[THEMIS 5.0 should] … consider vulnerabilities related to AI system fairness and technical accuracy and robustness, as well as to the EU legal framework for trusted AI.		*Vulnerabilities related to AI system fairness and technical accuracy and robustness.*
SR.15	[THEMIS 5.0 should] … perform and explain to the users a personalised AI Trustworthiness assessment based on at least 50 anomaly detection indicators and metrics: The fairness bias indicators will be defined according to NIST, including (a) systemic bias, (b) computational bias and (c) human bias Technical accuracy and robustness metrics will be able to adapt to the severity of risks in the business environment as well as to the users' needs and preferences. The indicators must be related to the embedded socio-technical environment.	SR_25 SR_26 SR_29 SR_30 SR_36	*Technical explanations of the SUT trustworthiness results will be provided. Will explore different ways (e.g. high level/qualitative) to display the results. (Row 3 & Row 4)*
SR.16	[THEMIS 5.0 should] … forecast risks and vulnerabilities based on users' persona and preferences	SR_03 SR_39	*Personas will be considered in risk assessment and enhancement suggestions. (Row 8) Similar to SR.19 below.*
SR.17	[THEMIS 5.0 risk assessment should] … be dynamic and be updated based on the implementation (or the projection) of trustworthiness optimisation measures.	SR_28	
EXPLORE			
SR.18	[THEMIS 5.0 should] … provide solutions that will improve the trustworthiness of the AI System.		
SR.19	[THEMIS 5.0 should] … propose mitigation measures based on users' persona and preferences.	SR_03 SR_39	*Personas will be considered in risk assessment and enhancement suggestions. (Row 8) Similar to SR.16 above.*

SR.20	The provided solutions will be explainable and transparent by providing the users with: input criteria used in the decision support process (training data sets, AI models, algorithms); the output of that process; and the perceived causal relationship between input and output, taking into consideration (i) the characteristics of the human recipients of decision support, (ii) the context and circumstances that triggered the decision support, and (iii) the user-defined objectives.	SR_07 SR_41 SR_42	-
	Identify where human-in-the-loop and human-in-command are needed and ensure human involvement in the AI-based decision support.	SR_35	
SR.21	THEMIS should incorporate alternative methods and algorithmic approaches for the trustworthiness assessment.	SR_27	-
SR.22	[THEMIS 5.0 should] … explore the adoption of negotiation strategies for the selection between multiple potential solutions, resulting from the different actors and objectives there may be.	SR_37	*Different solutions to address different vulnerabilities will be presented to the user, which according to their needs will negotiate the selection of the solutions based on their strategies. (Row 13)*
SR.23	[THEMIS 5.0 should] … Inform the user beforehand of the impact of the optimisation measures on the established business-related KPIs.	SR_38	*- (Row 11)*
SR.24	[THEMIS 5.0 should] … allow users to choose when and if a trustworthiness optimisation measure will be applied to the AI system under test.	SR_02 SR_48	*- (Row 9)*
SR.25	[THEMIS 5.0 should] … feature a fully equipped sandbox to design and deploy experiment variants and evaluate them against defined sets of criteria, through a no-code Big Data Analytics as a Service cloud-based designer (Interface). This sandbox must provide access to datasets for experimentation, Open APIs for experiment development, ML/DL algorithms models, advanced analytics, visualization configurations, human/machine interactions schemas, as well as the legal framework for trusted AI. A ModelOps approach should be followed for the operationalisation of the produced experimental variables.	SR_33 SR_34 SR_49 SR_51	*The TAIME trainer is an environment that retrains the THEMIS 5.0 models via a REST interface. The, new models are restricted to existing algorithms as the trainer is not intended to be a development environment. (Row 12)*
ENHANCE			
SR.26	Implement enhancement in improvement continuous cycles. THEMIS 5.0 must monitor AI systems' trustworthiness optimisation at each cycle by calculating quantitative and qualitative KPIs, benchmarking and measures on progress monitoring based on a defined set of criteria on the various process assets (AI models, algorithms, datasets, as well as specific configurations)	SR 02 SR 28	*Through the iterative TW assessment and risk assessment where the enhanced SUTs can be iteratively assessed and optimised.* *(Row 7).*
SR.27	Use feedback from users on the produced results to fine-tune the implemented AI models.	SR_02	-
SR.28	Ensure that the accuracy of the AI systems that have gone through one (or several) trustworthiness optimisation cycles will reach at least 70%.		-

11.2. User Requirements

In deliverable D2.1, a comprehensive set of (178) broad user requirements are documented, originating from the co-creation workshops and the use case partners and collected from:

- the two co-creation workshops with AI users and business stakeholders
- the user stories that were collected from the use case partners and their technical counterparts
- the user needs questionnaires filled in by the use case partners and their technical counterparts

These requirements were analysed for overlaps, consolidated and summarised to produce a set with 81 unique user requirements, distributed to the four stages of the Trustworthiness Optimization Process (as described in D2.1 and D2.2 in detail).

Furthermore, this consolidated set of user requirements was prioritised based on the MoSCoW method, that is one of the most common methods for software requirements prioritization (Achimugu et al, 2014). The term MoSCoW is an acronym for "Must have" (Mo), "Should have" (S), "Could have" (Co), and "Won't have" (W), each denoting a level of priority:

- "Must have" defines the requirements that must be included in the final product. In this case, all requirements mentioned in DoA are indicated as such.
- "Should have" defines high-priority requirements that should be included, if possible, within the delivery time frame.
- "Could have" requirements are desirable or nice to have requirements and could be included without incurring too much effort or cost.
- "Won't have" requirements are those recorded requirements, which, however, are out of the scope of the THEMIS 5.0 project.

This process continued well beyond the submission of D2.1 in the framework of T4.1 where partners discussed extensively on the User Requirements and updated the prioritisation taking into account their compliance with the scope of the THEMIS 5.0 project as well as the capacity of the consortium and the resources allocated to the project. The updated list of **59 User Requirements** is presented in the following table. Moreover, co-creation activities (i.e. Living Labs and Piloting) continuously provide input and feedback that the consortium will seek to incorporate in the THEMIS 5.0 platform.

Table 34: User Requirements

ID	User Requirement	User Req. ID	Relevant System Requirement from DoA	Top Level Analysis (Prioritization/Effort/Expertise)	Accepted (Yes/NO), Comments / Conditions / interpretations
OVERALL					
UR_1	THEMIS 5.0 interactions to be based on user's familiarity with AI.	UR_S1, UR_C13, UR_C38, UR_C39 UR_Q11, UR_Q23 UR_Q10	SR_07. THEMIS conversational interface must (i) adjust to the users' preferences and traits (...) SR_20. THEMIS personalised trustworthiness assessment will be based (i) on the users' (...) Technical awareness (...)	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_2	Provide to the user a user-manual with explanations of the various functionalities of THEMIS 5.0	UR_Q1, UR_Q12	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY *[Either as a "How-to" feature or through the conversational agent]*
UR_3	Provide to the user a user-manual with explanations of the terminology used by THEMIS 5.0.	UR_Q1, UR_Q12	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY *[Either as a "GLOSSARY" feature or through the conversational agent]*
	THEMIS 5.0 interactions to be based on user's:		-		
UR_4a	-Attitude towards AI (to provide	UR_C22, UR_C44	-	D2.1: COULD HAVE Effort: Reasonable with specific limitations	YES – LOW PRIORITY

	appropriate arguments)			Expertise: Existing	*[The identification of the users' persona and their TW preferences will consider the users' attitude towards AI]*
UR_4b	- Role within organization (to ensure alignment with organization guidelines).	UR_C3, UR_C4, UR_C21	-	D2.1: COULD HAVE Effort: Reasonable with specific limitations Expertise: Existing	YES – LOW PRIORITY *[The identification of the users' persona and their TW preferences will consider the users' role within organisation]*
UR_4c	-For clarity, it is advised that the user will select their role from predefined list	UR_C4	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES
UR_5	The interaction with the chatbot to be text-based	UR_C1	-	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES
UR_6	The chatbot to ask the same questions with multiple different ways (with paraphrases) to ensure clarity.	UR_C2	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES *[Users will be able to ask the conversational agent for explanations on the questions asked]*
UR_7	THEMIS 5.0 users can verify that functionalities are in line with organizational security guidelines	UR_C39UR_C42 UR_Q9	-	D2.1: SHOULD HAVE Effort: Excessive Expertise: Existing	NO (Row 2) *[Security measures will be put in place, but they cannot be customised to match those of the organisation's guidelines, which would result in a bespoke system for each customer.]*
UR_8	THEMIS 5.0 users can verify that functionalities are in line with organizational ethical guidelines	UR_C41	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY *[The conversational agent will identify the users' moral orientation and will investigate links with TW preferences. Also, risk assessment will also consider ethical aspects and impacts based on the Socio Technical model that is built for each use case.]*
UR_9	THEMIS 5.0 platform can be reached and used by laptops, computers and smartphones	UR_C43	-	D2.1: SHOULD HAVE Effort: Excessive Expertise: Existing	NO *[The effort for a cross-platform implementation is not in par with the impact achieved. It is anticipated that users will not need to use THEMIS outside their office settings]*

IDENTIFY					
	The chatbot to collect information about:		-		
UR_10a	-organizational ethical guidelines that the end-user must comply with.	UR_C7	SR_22. THEMIS conversational interface must be able to capture (…) the users' ethical values	D2.1: MUST HAVE-> COULD HAVE Effort: Excessive Expertise: Existing	YES – LOW PRIORITY *[The conversational agent will capture the moral orientation of the users, in the extend that any organizational ethical guidelines should be part of the socio-technical model that the risk assessment-based TW optimisation will be based upon, or prompted]*
UR_10b	-organizational security measures	UR_C8, UR_C24	-	D2.1: COULD HAVE Effort: Excessive Expertise: Existing	No *[Provided that any organizational security measures should be part of the socio-technical model that the risk assessment-based TW optimisation will be based upon.]*
UR_10c	-end-user's requirements with regards to trustworthiness characteristics	UR_C9	SR_39. Taking into account the users' profile and preferences, THEMIS will (i) forecast risks and vulnerabilities	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_10d	-end-user's climate concerns	UR_C10	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	No *[In the extend that any such concerns, should be part of the socio-technical model that the risk assessment-based TW optimisation will be based upon or prompted.]*
UR_10e	-where in the work process is the end-user using the SUT and the purpose of use.	UR_C11 UR_C12	SR_22. THEMIS conversational interface must be able to capture (i) the users' decision support needs, (…) objectives and KPIs that the optimised AI system will need to support.	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	No *[Any such concerns, should be part of the socio-technical model that the risk assessment-based TW optimisation will be based upon.]*
UR_10f	-organisation's rules and established procedures	UR_C23	-	D2.1: COULD HAVE Effort: Excessive Expertise: Existing	No (Row 22) *[If this is really important it should be part of the socio-technical model that the risk assessment-based TW optimisation will be based upon.]*

TRUSTWORTHINESS PREFERENCE PREDICTION					
UR_11	To first make the 20 most crucial questions for a quick start, and later refine the persona if necessary	UR_C14	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	No *[The persona identification is based on a methodology comprising specific questions. Having fewer questions will have unforeseeable results]*
UR_12	Interview the end-users only once, in the initialization of the platform and then update only if necessary (e.g., change of role)	UR_C17 UR_C18	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_13	Display the prediction of trustworthiness preferences to the user	UR_Q2	-	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY *[in the UI and/or through the conversational agent]*
UR_14	Enable the user to create various personas, based on role.	UR_C19	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY *[As long as they provide different responses (including their role) a different they will be assigned to a different user persona]*
UR_15	Enable the organization to define the same trustworthiness preferences among all employees (for the same SUT).	UR_C20 UR_Q14	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_16	Enable users to give feedback on the displayed predictions of preferences	UR_Q3	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_17	Provide users with explanations about the prediction of trustworthiness preferences, upon request.	UR_Q5	SR_01. THEMIS (..) should interact with the user via an AI-driven conversational interface that will be able to explain in the necessary level of detail the outputs of AI systems. *#Comment: Although SR_01 refers to the system under test, as THEMIS 5.0 platform is also an AI System that the user needs to trust we conclude that it refers to the THEMIS 5.0 platform as well.*	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY

UR_18	Enable users to adjust the estimated trustworthiness preferences.	UR_Q6	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_19	Enable users to copy the preferences of users from the same organization.	UR_Q7	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_20	Enable users to review their historical trustworthiness preferences.	UR_Q8	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_21	The identify stage should not last more than 5 mins	UR_Q16	-	D2.1: NA Effort: Reasonable Expertise: Existing	NO *[5 min. probably won't be enough for the whole duration of the IDENTIFY stage]*
UR_22	Trustworthiness assessment must take into account end-user's trustworthiness requirements.	UR_C25	SR_39. Taking into account the users' profile and preferences, THEMIS will (i) forecast risks and vulnerabilities	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_23	Consider for the trustworthiness assessment the purpose of use of the SUT & where in the work process the SUT is used.	UR_C26 UR_C27	SR_25. (...) trustworthiness assessment which will be based on (…) (iii) the embedding socio-technical environment. SR_24. THEMIS should (i) feature a GUI that will allow the users to create a qualitative model of the socio-technical environment by inserting (i.a) all the possible actions that the AI system can recommend , (i.b) all the KPIs that might be affected by any of the possible actions, (i.c) any external factors that might influence the actions or the actions' effect on the KPIs, and (i.d) how the actions and the external factors affect the KPIs through pairwise relations; THEMIS should be able to (ii) generate a qualitative model.	D2.1: MUST HAVE Effort: N/A Expertise: N/A	NO (Row 4): *[The quantitative TW assessment considers only the technical aspects of the SUT. Instead, "environmental/social" aspects should be part of the socio-technical model that the risk assessment-based TW optimisation will be based upon.]*

UR_24	Present AI system's trustworthiness assessment to the users via a chatbot-based interface, that is initiated by them and is tailored to their preferred level of detail.	UR_S2 UR_S3	SR_01. THEMIS cloud-based services must interact with the user via an AI-driven conversational interface that will be able to explain in the necessary level of detail the outputs of AI systems. *#Comment: Although SR_01 refers to the system under test, as THEMIS 5.0 platform is also an AI System that the user needs to trust we conclude that it refers to the THEMIS 5.0 platform as well.*	D2.1: MUST HAVE Effort: Excessive Expertise: Existing	YES – HIGH PRIORITY
UR_25	Present initially to the user the trustworthiness assessment results (chatbot or document) in the form of a high-level report with the most important results.	UR_S4 UR_Q22	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_26	Present to the user more details on specific results from the trustworthiness assessment overview on demand (chatbot or hyperlink in the document)	UR_S5	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_27	Present to the user the trustworthiness assessment results in a data-visualisation like dashboard (e.g. PowerBI) with descriptions	UR_Q25	-	D2.1: COULD HAVE Effort: Excessive Expertise: Existing	NO (Row 5): *[The trustworthiness assessment results will be presented to the user but the requirement for a data-visualization like dashboard is overly costly which is not justified by the additional anticipated impact.]*
UR_28	Present to the user the trustworthiness assessment results with quantitative metrics.	UR_Q26	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_29	Display to the user the trustworthiness assessment in context-specific way (General error track	UR_Q27	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY (Row 7) *[Based on the methods used and their specificities*

	record, Correlation of error with conditions and events, Training datasets details, e.g., 20% of the patients may be wrongly identified as of high risk)				*we will be able to provide (to some extend) the context-specific quantitative assessment.]*
UR_30	Provide to the user the trustworthiness assessment report in a user-friendly way to save in a digital format.	UR_C27	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_31	Present to the user which parts of the SUT suffer (e.g., model, deployment, datasets), but the level of detail will be based on user preferences.	UR_S7 UR_Q28 UR_Q29	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_32	Present to the user explanations on SUT trustworthiness assessment on demand, but the level of detail will be based on user preferences.	UR_Q30 UR_Q31 UR_Q35 UR_Q37	SR_25. THEMIS must provide explanations for the trustworthiness assessment (…)	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_33	Present to the user the factors that have mostly influenced the trustworthiness assessment.	UR_Q37	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_34	Display to the user the system's current trustworthiness assessment against previous trustworthiness assessments.	UR_Q5 UR_S13 UR_Q32	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_35	Enable the user to obtain a comparative analysis of multiple AI systems and services for the same task (selected by the end-user).	UR_Q34	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_36	Enable the user to choose which trustworthiness characteristics would like to be assessed	UR_Q36	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY (Row 6) *[Only specific methods that correspond to the selected TCs will be executed.]*
UR_37	Every time the SUT is updated, the system		SR_28. THEMIS trustworthiness	D2.1: MUST HAVE Effort: Reasonable	YES – HIGH PRIORITY

	signal to rerun the trustworthiness assessment		assessment will be dynamic and will be updated based on the implementation (or the projection) of trustworthiness optimisation measures.	Expertise: Existing	
UR_38	The Trustworthiness Assessment should not last more than an hour		-	D2.1: NA Effort: NA Expertise: Existing	YES – LOW PRIORITY
UR_39	Enable the user to perform experiments both for different scenarios (e.g., articles when trained only in posts), and on own datasets (e.g., sets of "hard" articles).	UR_Q43 UR_Q44 UR_S12	SR_33. THEMIS will provide access to datasets for experimentation, Open APIs for experiment development, ML/DL algorithms models, advanced analytics, visualization configurations, human/machine interactions schemas, as well as the legal framework for trusted AI. SR_34. THEMIS will feature a fully equipped sandbox to design and deploy experiment variants and evaluate them against defined sets of criteria.	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY (Row 14) *[The end user will be able to download the model and perform own tests to check the performance of the (enhanced) SUT in out of distribution data.]*
RISK ASSESSMENT					
UR_40	Present to the user the related risks stemming from trustworthiness vulnerabilities.	UR_S9	SR_31. THEMIS will implement trustworthiness assessment based on a risk assessment approach. SR_32. THEMIS (..) will be able to assess the AI system's accuracy and fairness their impact on the business-related risks and KPIs.	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_41	The system could have a BI tool to analyse risk assessment results	UR_Q46	-	D2.1: COULD HAVE Effort: Excessive Expertise: Existing	NO (Row 8): *[THEMIS will perform risk assessment and will present to the user its results, but the requirement for a BI tool to analyse risk assessment results is redundant and*

					not justified by the additional anticipated impact.]
UR_42	Provide on demand explanations to the user on how the presented risks have been calculated.	UR_Q47	SR_01. THEMIS (..) should interact with the user via an AI-driven conversational interface that will be able to explain in the necessary level of detail the outputs of AI systems. *#Comment: Although SR_01 refers to the system under test, as THEMIS 5.0 platform is also an AI System that the user needs to trust we conclude that it refers to the THEMIS 5.0 platform as well.*	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY (Row 9)
UR_43	Inform the user about the factors which have mostly influenced the risk assessment	UR_Q48	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY *[Similar with UR_42 above]*
UR_44	Present the risk assessment results at once (chatbot or document) with a high-level report with the most important results and present more details on demand.	UR_S10 UR_S11	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_45	The Risk Assessment should not last more than a few seconds		-	D2.1: NA Effort: NA Expertise: Existing	NO *[A few seconds may not be enough for the Risk Assessment.]*
EXPLORE					
UR_46	Inform the user about positive and negative business impacts for each enhancement measure	UR_S14, UR_C29, UR_Q54	SR_38. THEMIS will inform the user beforehand for the impact of the optimisation measures on the established business related KPIs.	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_47	Inform the user about positive and negative impacts in SUT's trustworthiness for each enhancement measure.	UR_S16, UR_Q59	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY (Row 13)
UR_48	Inform the user about the required time, and cost for the implementation of	UR_Q58	-	D2.1: WON'T HAVE Effort: N/A Expertise: N/A	NO *[Out of scope]*

	each enhancement measure.				
UR_49	Inform the user about the estimated changes in the usability of the tool (e.g., certain functionalities become slower by 10%).	UR_Q60	-	D2.1: WON'T HAVE Effort: N/A Expertise: N/A	NO *[Out of scope]*
UR_50	For each enhancement suggestion inform the user about the respective climate impact rating	UR_C30	-	D2.1: WON'T HAVE Effort: N/A Expertise: N/A	NO *[Out of scope]*
UR_51	For each enhancement suggestion provide explanations confirming that ethical/legal/security/ climate organization's restrictions are respected.	UR_C31 UR_Q6 UR_Q67	-	D2.1: WON'T HAVE Effort: N/A Expertise: N/A	NO *[Out of scope]*
UR_52	Present enhancement suggestions in tabular form with the pros and cons	UR_Q55	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_53	Present enhancement suggestions using a rating system	UR_C35	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_54	Provide the user with a visual representation of the trade-offs (when increasing fairness-accuracy drops, etc.) for each suggestion	UR_Q56	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY (Row 13) *[This is also related to UR_47 above]*
UR_55	All different types of roles that are using the SUT within the organization are considered for calculating enhancement suggestions, to ensure that no contradictory suggestions are made.	UR_C34	SR_37. THEMIS will explore the applicability implement negotiation strategies for the selection between multiple potential solutions, resulting from the different actors and objectives there may be	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY (Row 15) *[Different roles within an organisation may have different TW preferences or different priorities based on their role. Several persons from a single organisation may give input to the sociotechnical model that the risk assessment-based TW enhancement will be based upon.]*
UR_56	User's years of experience (to ensure motivation of decision	UR_C5, UR_C32	-	D2.1: WON'T HAVE Effort: N/A Expertise: N/A	NO

	making) are considered for calculating enhancement suggestions.				*[Unclear how years of experience is relevant]*
UR_57	User's professional interests (to ensure motivation of decision making) are considered for calculating enhancement suggestions.	UR_C6	SR_20. THEMIS personalised trustworthiness assessment will be based on (…) Motivation (…)	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY *[The users' professional interests are reflected in the sociotechnical model that the risk assessment-based TW enhancement will be based upon]*
UR_58	Exploration of solutions should be accessible only by specific roles.	UR_Q57	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY *[Authorised users should have the rights to explore and implement TW enhancements]*
UR_59	Provide the user with a testbed environment to allow the application of the provided solutions in a pipeline.	UR_S17	-	D2.1: SHOULD HAVE Effort: Excessive Expertise: Existing	NO (Row 10)
UR_60	Enable users to add new potential risks of trustworthiness vulnerabilities.	UR_Q61	-	D2.1: SHOULD HAVE Effort: Excessive Expertise: Existing	YES – LOW PRIORITY (Row 11) *[The risks are reflected in the sociotechnical model that the risk assessment-based TW enhancement will be based upon. The possibility for this model to be dynamic will be investigated.]*
UR_61	Enable only specific roles to add new potential risks of trustworthiness vulnerabilities.	UR_Q62	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY *[Depends on UR_60 above. If this is implemented, authorised users should have the right to add new risks]*
UR_62	Enable the user to select when and if a trustworthiness optimisation measure will be applied to the AI system under test.	UR_S15	SR_48. THEMIS users will be able to choose when and if a trustworthiness optimisation measure will be applied to the AI system under test.	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY (Row 12) *[After the EXPLORE phase a ranking of acceptable and applicable solutions will be provided. Then the human supervisor has to select which one to be implemented.]*
UR_63	The optimal solution should be directly	UR_Q64	-	D2.1: WON'T HAVE Effort: N/A	NO (Row 21)

	implemented without asking the user.			Expertise: N/A	*[Contradicts other user and system requirements (SR_48, UR_62)]*
UR_64	Enable the user to choose between allowing THEMIS 5.0 to perform the optimal solution automatically and allowing the user the one that prefers	UR_Q65	-	D2.1: COULD HAVE Effort: Excessive Expertise: N/A	NO *[Contradicts other user and system requirements (SR_48, UR_62)]*
UR_65	Present to the user how the changes made to the system will affect the risk assessment against previous assessments.	UR_Q45	-	D2.1: MUST HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_66	The explore stage should not last more than a few days	UR_Q71	-	D2.1: N/A Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY *[This timeframe sounds reasonable, although this also depends on the users]*
ENHANCE					
UR_67	Present to the user the new trustworthiness assessment results.	UR_S19	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_68	Present to the user an overview of the updated risk assessment of the AI system against previous values.	UR_Q72	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_69	Provide the user with a visual representation of trustworthiness parameters changes through time (e.g. in tabular form along with timestamps), given a predefined timeframe.	UR_Q73 UR_Q74	-	D2.1: COULD HAVE Effort: Reasonable Expertise: Existing	YES – LOW PRIORITY
UR_70	Present to the user upon request analytical details on what has changed to the SUT (e.g., fine-tuned in dataset X)	UR_Q75	-	D2.1: SHOULD HAVE Effort: Reasonable Expertise: Existing	YES – HIGH PRIORITY
UR_71	Present to the user upon request the sources used for generating the enhanced results.	UR_Q76	-	COULD HAVE	YES – LOW PRIORITY (Row 18) *[Present available information on datasets and models used for the enhanced SUT as well as*

					the any methods that have been used for training.]
UR_72	The enhancement stage should not last more than 24 hours	UR_Q77	-	D2.1: N/A Effort: N/A Expertise: Existing	YES – LOW PRIORITY *[This timeframe sounds reasonable.]*

12. ANNEX 5: PERSONA IDENTIFICATION METHODOLOGY

Building on the exploration of the qualitative data gathered during the co-creation workshops, trustilio conducted a comprehensive and systematic thematic analysis to identify emerging themes related to user preferences aiming to allocated them in relevant clusters.[1] These emerging themes were then cross-checked against the existing relevant literature to develop the final recommendations. Below, we present the process and results of this analysis per sector:

Healthcare Sector

The workshops characterized by a diverse range of professionals with varying levels of experience and attitudes towards AI technology reflecting on the sector. The clustering of users in this sector reflects these differences, particularly in terms of professional experience, trust in AI, and motivational traits as below:

In terms of **Professional Experience and Role Diversity,** the users can be allocated to two groups[2]:

Young Professionals: Often more open to adopting new technologies, including AI. They are generally more tech-savvy and adaptable but may lack the deep experience required to understand the full implications of AI decisions.

Experienced Professionals: These individuals tend to be more sceptical of AI, often due to a lack of transparency in how AI systems reach their conclusions. Their deep expertise makes them more critical of AI outputs, especially when AI recommendations conflict with their clinical judgment or experience. They expressed the opinion that AI systems should not only be accurate but also explainable.

Regarding **Trust and Transparency** there were two themes emerged:

Transparency in Decision-Making: where healthcare professionals demanded high levels of transparency from AI tools, especially regarding the data used for training these tools and the algorithms' decision-making processes. The theme is in alignment with existing literature referring to 'trust', as professionals wish to understand and verify the rationale behind AI-generated recommendations before applying them to patient care[3].

Bias in Data and Decision Support: Concerns about biased data were prevalent in the workshops with healthcare professionals, as diverse patient populations require AI systems trained on representative datasets. According to the literature, inaccurate or biased AI recommendations could have serious implications for patient safety and outcomes, hence the need for rigorous validation and ongoing training of AI systems.[4]

Exploring the shared motivations, the analysis identified the following **Motivational Trait** for **Patient-Centric Approach**. As expected and relevant to their profession, healthcare professionals expressed as their primary motivation the improvement of patient outcomes. They see AI tools as beneficial if they can enhance diagnostic accuracy, optimize treatment plans, and support patient-centred care and on the contrary, they perceive AI systems that are focusing more on cost-efficiency than patient welfare with resistance.

Reflecting on the emerged characterising's and needs as these expressed through their motivation, the healthcare professionals can be allocated to the following Clusters[5]:

Cluster 1. Young, Tech-Savvy Professionals:

- Characteristics: Open to innovation, adaptable, tech-oriented, willing to integrate AI into their workflows.

- **Needs**: AI tools that facilitate learning and skill development, transparency in AI operations to ensure trust, continuous training to avoid over-reliance on AI.

Cluster 2. Experienced, Sceptical Professionals:

- Characteristics: Critical of new technologies, experienced in their field, prioritize patient safety, cautious of AI integration.

- Needs: AI tools with high transparency and explainability, rigorous testing and validation of AI recommendations, assurance of AI complementing rather than replacing human judgment.

Port Management Sector

The workshops in the port management sector involved a range of professionals, each with distinct responsibilities and varying levels of familiarity with AI technologies. It emerged that the sector's complexity

necessitates AI tools that are reliable, transparent, and capable of handling diverse and dynamic conditions and the analysis regarding the participants roles identified two overarching main groups[6]:

Role and Function Diversity:

Operational Staff: Included roles such as port traffic controllers and shipping agents who manage day-to-day operations. These professionals expressed preference for AI tools that provide real-time data and decision support without replacing their expertise, and a need for user-friendly AI interfaces that accommodate varying levels of technical proficiency.

Management and Strategic Roles: Involved decision-makers who oversee broader port operations and strategy. These professionals were interested in AI tools that can enhance efficiency, optimize resource allocation, and provide predictive insights.

Regarding their **Technical Proficiency and Adaptation** there where two categories emerging:

Varying Levels of Technical Skills: The workshops included individuals with differing technical skills. Some had good knowledge in digital solutions and datasets, while others required additional training to effectively use AI tools. They expressed a preference for AI systems that are intuitive and easy to use, especially for less technically skilled staff.

Adaptability to Complex Scenarios: The participants valued AI tools that can handle complex scenarios and provide reliable solutions. In addition, they wished AI to support dynamic decision-making processes, especially in unpredictable situations such as severe weather or operational disruptions.

Furthermore, in this sector there was a new emerging theme which wasn't identified in the healthcare sector. The participants referred to the commercial aspect of the maritime industry expressing their considerations:

Commercial and Legal Considerations:

- Economic Impact and Legal Responsibility: Professionals in this sector were concerned about the commercial implications of AI errors, which could result in substantial economic losses. They referred also to a need for clarity on legal responsibilities when AI tools are involved in decision-making processes.

Reflecting on the emerged characterising's needs and considerations as these emerged by their roles and considerations, the maritime professionals can be allocated to the following Clusters:

Cluster 1. Technically Proficient Managers:

- Characteristics: Comfortable with technology, interested in AI for optimizing operations, require high reliability and precision from AI tools.

- Needs: Advanced AI tools that enhance decision-making, support for handling large datasets, clear legal frameworks defining AI responsibility.

Cluster 2. Operational Staff with Basic Technical Skills:

- Characteristics: Less familiar with AI, rely on hands-on experience, need intuitive and user-friendly AI systems.

- Needs: Training programs for AI adoption, easy-to-use AI interfaces, tools that provide clear and actionable insights without overwhelming technical detail.

Journalism/ Disinformation in Media Sector

The workshops in the media sector, included journalists and fact-checkers, and the overarching theme was a strong emphasis on ethical standards, accuracy, and transparency. There was scepticism around AI adoption due to concerns about potential bias and the impact on journalistic integrity concluding on the following emerging themes.[7,8]

Scepticism and Ethical Standards:

Cautious Attitude Towards AI: Media professionals, particularly journalists and fact-checkers, were wary of AI tools, fearing that they could compromise journalistic standards. The current AI tools are perceived as potentially generating biased or inaccurate information, which conflicts with the core values of journalism.

Adherence to Ethical Norms: There was a strong commitment to ethical standards and journalistic integrity. Media professionals demanded AI tools that are transparent, accurate, and capable of adhering to these

standards. They were particularly concerned about the "black box" nature of some AI systems, which could hinder accountability.

Transparency and Accuracy:

Need for Explainability: Journalists and fact-checkers referred to AI tools that provide clear explanations for their outputs. This transparency is essential for verifying the validity of information and maintaining trust in their work.

Data Bias and Integrity: Concerns about biased data were prevalent, especially if AI tools are trained on datasets that do not reflect diverse perspectives or are influenced by the developers' biases. Media professionals referred to AI tools that mitigate these biases and ensure accurate and reliable outputs.

Regarding their **Technical Proficiency and Tool Utility the results were inconclusive as there was a** wide range of technical proficiency among participants. While some expressed comfort in using AI tools for data analysis and fact-checking, others were less familiar with these technologies and expressed the preference for tools that are straightforward and easy to use.

As with the previous sectors, reflecting on the emerged characterising's and preferences, the media sector professionals can be allocated to the following Clusters

Cluster 1. Sceptical Journalists:

- Characteristics: Highly critical of AI, prioritize ethical standards, concerned about the impact of AI on journalistic integrity.

- Needs: AI tools that are transparent, provide clear justifications for outputs, adhere strictly to ethical standards, and allow human oversight and control.

Cluster 2. Fact-Checkers and Analysts:

- Characteristics: Require high accuracy and transparency, focus on verifying information, need tools that support detailed analysis and comparison of data.

- **Needs**: AI tools with robust data validation capabilities, clear explanations for AI-generated conclusions, tools that facilitate comprehensive fact-checking processes without automating critical thinking.

The next step in the analysis involved mapping the users' characteristics, which emerged from the thematic analysis, alongside the psychological, social, and behavioural parameters based on trustilio's previous work on human profiling and informed by relevant literature.

As we have discussed above, in the Healthcare Sector the participants were Clustered in Young, Tech-Savvy Professionals, and Experienced, Sceptical Professionals with the relevant characteristics. We conducted a mapping exercise of these against the Human Traits in the table below.

Table 35: Users Extended Characteristics

Personality Traits	Description & Examples
Extraversion	Gregariousness (e.g., Social engagement in attackers' groups) Assertiveness/Outspokenness (e.g., Leadership skills) Activity/Energy level (e.g., Enjoys a busy life) Positive Emotions/Mood (e.g., Happiness)
Conscientiousness	Orderliness/Neatness (e.g., Well-organized) Striving/Perseverance (e.g., Aims to achieve excellence) Self-Discipline (e.g., Persistent engagement to goals) Dutifulness/Carefulness (e.g., Strong sense of duty) Self-Efficacy (e.g., Confidence to achieve goals)
Openness to experiences	Intellect/Creativity Imaginative (e.g., Intellectual style) Scientifically Interested/Originality (e.g., Evidence-based) Adventurousness (e.g., Experiences of different things)

Social - Behavioural Traits	Description & Examples
Selected social exposure	Difficult to adapt to conventional social norms (e.g., Events) Easy to build virtual anonymous, professional relationships (e.g., Using anonymous identity has contacts with other attackers in the Deep Web) Easy to build strong e-bonds in hacking communities (e.g., These communities are closed to the public)
Not conventional relationships	Difficult to build physical relationships or contacts Easy to build professional (with other attackers) virtual, anonymous relationships under their moral code (us versus them approach)
Not talkative	Difficult to initiate small casual talks or social talks Difficult to express him/herself
Manipulative	Easy manipulating people via electronic means (e.g., phishing)
Technical Traits	**Description & Examples**
Networking skills	Knowledge in network architectures, systems, functional and operational aspects (e.g., DNS, HCP)
IT skills	Competencies in operating systems (e.g., languages, software and emerging technologies, programming)
Soft skills	Problem Solver (e.g. Understand, analyse and solve difficult problems) Social observer (e.g., Audits security behaviours)
Forensics skills	Know how to use security scripts, forensics tools (e.g., Intrusion detection/penetration tools)
Available resources	Available computing power (e.g., Owns/access to high computer processing power), time, economic support security communities
Position/access	Insider (e.g., Works in the organization) Supply chain partner (e.g., Part of the value chain) Outsider, external user, vendor/manufacturer Third party (e.g., physical access of the target via a network or local access)
Targeted Knowledge	Information/ measurements gathered about the targets (e.g., CVSS)
Motivational Traits	**Description & Examples**
Political	Political power (e.g., Espionage, fake news)
Personal	Personal satisfaction, feeling of accomplishment, boredom, competition
Social/Cultural	Whistleblower (warns of any digital wrongdoings)
Philosophical/ Theological	Humanitarian/activist goals (e.g., Stealing for societal benefit)
Trigger Traits	**Description & Examples**
Vulnerable assets	Open ports (e.g., Zero-day vulnerability) New non-certified technologies (e.g., App, AI systems)
Human weaknesses/errors	Vulnerable infrastructures (e.g., No access control in data centre) Unintentional human error (e.g., Distracted administrator) Intentional human error (e.g., Reckless but knowledge of risk)

Mapped Characteristics Healthcare Sector

Personality Traits

- Conscientiousness: Both clusters, but particularly experienced professionals, display high conscientiousness. This trait includes self-discipline and carefulness, aligning with their cautious approach to AI adoption and their focus on patient safety and ethical standards.

- Openness to Experiences: Young professionals exhibit openness to new technologies and innovative practices, reflecting an adventurous and imaginative nature. They are more willing to experiment with AI systems and integrate them into their workflows.[10]

Social-Behavioural Traits

- Selected Social Exposure: Experienced professionals might have a tendency towards selected social exposure, engaging primarily within trusted networks and communities in their field to discuss AI use and share experiences.

- Not Conventional Relationships: Young professionals may form non-traditional working relationships, especially in interdisciplinary and tech-driven environments where AI is increasingly relevant.

Technical Traits

- IT Skills: Young professionals are likely to possess stronger IT skills, making them more adept at understanding and utilizing AI tools.

- Soft Skills: Experienced professionals might exhibit strong problem-solving skills, crucial for critically assessing AI outputs and integrating them into patient care without over-reliance.

Motivational Traits:

- Personal: Both clusters are motivated by personal satisfaction in achieving accurate diagnoses and improving patient outcomes. However, experienced professionals might also focus on maintaining professional standards and ethical considerations.

- Philosophical/Theological: A commitment to humanitarian goals, such as improving patient care and ensuring fairness in AI applications, aligns with the healthcare sector's ethos.

Trigger Traits

- Human Weaknesses/Errors: The potential for unintentional errors, such as over-reliance on AI by young professionals, is a critical trigger trait that needs management through proper training and guidelines.[11]

Mapped Characteristics Port Management Sector

We conducted the same exercise in the port management sector, where we mapped the qualitative data of the user clusters (Technically Proficient Managers and Operational Staff with Basic Technical Skills) with the Human Traits in Table 1.

Personality Traits

- Conscientiousness: Both clusters show conscientiousness, but proficient managers particularly demonstrate self-efficacy and striving traits, aiming to optimize port operations through AI.

- Openness to Experiences: Proficient managers are open to leveraging AI for enhancing efficiency and decision-making, reflecting intellectual curiosity and an interest in innovative solutions.

Social-Behavioural Traits

- Manipulative: While not in a negative context, technically proficient managers use their networking and social skills to navigate complex operational environments and manage AI tools effectively.

- Not Talkative: Operational staff may display a reluctance to engage deeply with AI due to limited technical proficiency, preferring established practices over new technologies.

Technical Traits

- Networking Skills: Proficient managers demonstrated understanding of network architectures and systems, essential for integrating AI tools into port operations.

- **Available Resources**: Managers might have access to better computing resources and infrastructure, supporting more sophisticated AI applications.

Motivational Traits

- **Social/Cultural**: There is a focus on maintaining smooth port operations and meeting economic objectives, which aligns with a collective motivation to enhance organizational performance through AI.

Trigger Traits

- **Vulnerable Assets**: Operational staff with basic skills may be less aware of vulnerabilities that AI systems could exploit, necessitating training to improve their understanding and minimize risks.

As for the Clusters (Sceptical Journalists, Fact-Checkers and Analysts) in the disinformation in the media workshops, the exercise resulted on the following:

Personality Traits:

- **Agreeableness**: Sceptical journalists may score lower on trust due to concerns about AI's impact on journalistic integrity and ethical standards.
- **Conscientiousness**: High levels of self-discipline and duty are evident in both clusters, reflecting their commitment to ethical journalism and accuracy in reporting.

Social-Behavioural Traits:

- **Not Conventional Relationships**: Journalists and fact-checkers often work independently or within niche communities focused on maintaining high standards, which may not always conform to mainstream norms.
- **Not Talkative**: There might be limited interaction with AI developers or tech experts unless necessary, due to a focus on traditional journalism practices and scepticism towards AI.

Technical Traits:

- **Soft Skills**: Strong analytical skills are essential for fact-checkers and journalists, enabling them to critically evaluate AI-generated information and maintain high accuracy in their work.
- **Available Resources**: Access to reliable data and verification tools is crucial for these professionals to validate AI outputs and ensure the accuracy of information.

Motivational Traits:

- **Political**: Fact-checkers and analysts may be driven by a commitment to truth and combating misinformation, reflecting a social responsibility that aligns with broader societal goals.
- **Philosophical/Theological**: There is often a deep philosophical commitment to maintaining ethical standards and truth in media, guiding their cautious approach to AI adoption.

Trigger Traits:

- **Human Weaknesses/Errors**: The risk of intentional or unintentional bias in AI-generated content is a significant concern, prompting rigorous checks and balances by fact-checkers and journalists.

Designing Dialogues with a Chatbot aiming to allocate responders in Clusters

To create an effective dialogue for clustering users in the above user preference groups, a chatbot could ask questions that probe these areas.

Questions and Keywords

Here we present suggested questions and keywords for each attribute:

1. Professional Experience and Role

Question: "Can you tell me about your professional experience in your current field?"

Keywords: Years of experience, role, field, professional background.

Question: "What is your current role, and what are your primary responsibilities?"

Keywords: Role, responsibilities, daily tasks, position.

2. Technical Proficiency

Question: "How comfortable are you with using new technologies, particularly AI tools, in your work?"

Keywords: Comfort level, technology, AI tools, proficiency.

Question: "Do you have any formal training or certifications in AI or related technologies?"

Keywords: Training, certifications, AI, technology skills.

3. Attitudes towards AI

Question: "What are your thoughts on the adoption of AI in your industry? Are you generally in favour, sceptical, or neutral?"

Keywords: Adoption, AI, industry, favour, sceptical, neutral.

Question: "Have you had any positive or negative experiences with AI technologies at work?"

Keywords: Experience, positive, negative, AI technologies.

4. Trust and Transparency Requirements

Question: "How important is transparency in AI decision-making processes to you?"

Keywords: Transparency, AI decision-making, importance, trust.

Question: "Do you prefer AI systems that provide detailed explanations for their decisions?"

Keywords: Explanations, AI systems, decisions, preferences.

5. Motivational Traits

Question: "What motivates you most in your professional role? Is it innovation, efficiency, ethical considerations, or something else?"

Keywords: Motivation, innovation, efficiency, ethics, professional role.

Question: "Would you say your primary focus at work is improving processes, maintaining standards, or exploring new technologies?"

Keywords: Focus, improving processes, maintaining standards, exploring technologies.

6. Responsiveness to Uncertainty

Question: "How do you typically handle uncertainty in your work environment?"

Keywords: Uncertainty, handling, work environment, response.

Question: "Are you comfortable using AI tools in situations where outcomes are unpredictable?"

Keywords: AI tools, unpredictable, comfort, outcomes.

7. Commitment to Ethical Standards (Specific to the Disinformation Sector)

Question: "How important are ethical standards and integrity in your work with AI and technology?"

Keywords: Ethical standards, integrity, AI, technology.

Question: "Do you think AI tools should always allow for human oversight to maintain ethical practices?"

Keywords: AI tools, human oversight, ethical practices, maintain.

Responses mapping Clusters

To we need to analyse the potential responses based on the profiling methodology outlined in the THEMIS document.

1. Professional Experience and Role

Question: "Can you tell me about your professional experience in your current field?"

Possible Answers and Clustering:

Less than 5 years of experience: Likely to be clustered into "Young Tech-Savvy Professionals" in the healthcare sector or "Operational Staff with Basic Technical Skills" in the port management sector.

5-15 years of experience: May indicate mid-career professionals who are more experienced but still open to new technologies. Could fall into "Technically Proficient Managers" or "Sceptical Journalists" based on their attitudes towards AI.

More than 15 years of experience: Likely to be clustered as "Experienced Sceptical Professionals" in healthcare or "Experienced Managers" who require high transparency and reliability in AI tools.

2. Technical Proficiency

Question: "How comfortable are you with using new technologies, particularly AI tools, in your work?"

Possible Answers and Clustering:

Very comfortable: Could indicate "Young Tech-Savvy Professionals" in healthcare, "Technically Proficient Managers" in port management, or "Fact-Checkers and Analysts" in disinformation who are open to using new technologies.

Somewhat comfortable: May suggest users who are adaptable but cautious, potentially falling into clusters like "Operational Staff with Basic Technical Skills" or "Sceptical Journalists."

Not comfortable at all: Likely to be grouped into clusters requiring more support and training, such as "Operational Staff with Basic Technical Skills" or "Experienced Sceptical Professionals" who are resistant to AI.

3. Attitudes towards AI

Question: "What are your thoughts on the adoption of AI in your industry? Are you generally in favour, sceptical, or neutral?"

Possible Answers and Clustering:

In favour: Users who are enthusiastic about AI adoption are likely to be "Young Tech-Savvy Professionals" or "Technically Proficient Managers."

Sceptical: Indicates a cautious approach, likely clustering into "Experienced Sceptical Professionals" in healthcare or "Sceptical Journalists" in disinformation.

Neutral: May suggest a balanced view, potentially placing them in clusters like "Operational Staff with Basic Technical Skills" who are open to AI but not particularly enthusiastic.

4. Trust and Transparency Requirements

Question: "How important is transparency in AI decision-making processes to you?"

Possible Answers and Clustering:

Very important: Likely to fall into "Experienced Sceptical Professionals" or "Sceptical Journalists" who value transparency and ethical standards in AI.

Somewhat important: Could indicate users who value transparency but are not solely driven by it, such as "Young Tech-Savvy Professionals" or "Technically Proficient Managers."

Not important: Users who do not prioritize transparency might be more results-oriented, potentially fitting into "Young Tech-Savvy Professionals" or "Operational Staff with Basic Technical Skills."

5. Motivational Traits

Question: "What motivates you most in your professional role? Is it innovation, efficiency, ethical considerations, or something else?"

Possible Answers and Clustering:

Innovation: Suggests a cluster like "Young Tech-Savvy Professionals" or "Technically Proficient Managers" who are eager to integrate new technologies.

Efficiency: Could fit into "Technically Proficient Managers" or "Operational Staff with Basic Technical Skills," who focus on improving processes.

Ethical Considerations: Likely to be "Experienced Sceptical Professionals" or "Sceptical Journalists" who prioritize ethical standards and transparency.

Other (e.g., patient care, safety): Could fall into more specialized clusters based on sector-specific motivations, like "Fact-Checkers and Analysts" focusing on accuracy and integrity.

6. Responsiveness to Uncertainty

Question: "How do you typically handle uncertainty in your work environment?"

Possible Answers and Clustering:

Adapt well to uncertainty: Likely to be in clusters that value flexibility and quick decision-making, such as "Young Tech-Savvy Professionals" or "Technically Proficient Managers."

Prefer predictable environments: Indicates a preference for stability and may suggest clusters like "Experienced Sceptical Professionals" or "Sceptical Journalists" who require clarity and reliability.

Avoid uncertainty when possible: Could align with clusters that need more structured support and predictable outcomes, such as "Operational Staff with Basic Technical Skills."

7. Commitment to Ethical

Question: "How important are ethical standards and integrity in your work with AI and technology?"

Possible Answers and Clustering:

Extremely important: Suggests a high alignment with clusters like "Sceptical Journalists" and "Fact-Checkers and Analysts," who prioritize ethics and accuracy.

Moderately important: Could indicate a balanced approach, potentially clustering into groups that value ethics but are also pragmatic, such as "Technically Proficient Managers."

Not important: Unlikely to be a significant portion in sectors like disinformation; if present, might suggest a need for ethical training and awareness.

Example of Clustering Process

In a hypothetical scenario that a participant provides the following answers:

Experience: 10 years in healthcare

Technical Proficiency: Somewhat comfortable with AI

Attitudes towards AI: Sceptical

Trust and Transparency: Very important

Motivation: Ethical considerations

Uncertainty Handling: Prefers predictable environments

They would be allocated to the "Experienced Sceptical Professionals" cluster in the Healthcare Sector as we have seen above that this cluster is characterized by experienced professionals who are critical of new technologies and require high transparency and ethical standards.

Combining potential answers in the different sectors we propose the below Matrix as a tool for mapping and clustering (see Appendix for instructions on using the Matrix. Note that the Trustworthiness Preferences (TW) column includes comments made by participants during the co-creation workshops, as they were perceived and expressed by them. These comments are presented without deviating or conflicting with the definitions that were shared with participants during the workshop:

a) Fairness: The principle of justice in trustworthy AI also underlines fairness and the prevention of discrimination. Fairness of an AI system “ensures that there is an absence of any discrimination or favouritism toward an individual or a group based on any inherent or acquired characteristics that are irrelevant in the context of decision making”.

b) Robustness: Robustness concerns the AI-systems' ability to perform as expected under varying conditions. Robustness concerns the "sensitivity of the system’s outcome to a change in the input". It is accentuated that robustness "is a goal for appropriate system functionality in a broad set of conditions and circumstances, including uses of AI systems not initially anticipated". Robustness concerns system’s ability to deal with error at any point in the lifecycle and that the system is resilient to attacks.

c) Accuracy: Accuracy may be understood as the "closeness of results of observations, computations, or estimates to the true values or the values accepted as being true". For AI-systems it may be important to note that this particularly concerns how well the AI system "do on new (unseen) data compared to data on which it was trained and tested?"

Table 36: Mapping and Clustering Matrix

Question	Possible Answers	Healthcare Sector Clusters	Port Management Sector Clusters	Disinformation in Media Sector Clusters	Trustworthiness Preference
Professional Experience and Role	- Less than 5 years of experience	Young Tech-Savvy Professionals	Operational Staff with Basic Technical Skills	Young Media Professionals	**Robustness**: Needs reliable tools that facilitate learning and reduce complexity.

	- 5-15 years of experience	Mid-career Professionals (split based on attitudes)	Technically Proficient Managers or Operational Staff	Fact-Checkers and Analysts	**Accuracy**: Requires tools that provide precise and actionable insights to enhance decision-making.
	- More than 15 years of experience	Experienced Sceptical Professionals	Experienced Managers	Experienced Journalists or Analysts	**Fairness**: Demands high transparency and ethical considerations in AI outputs, especially concerning data integrity and bias.
Technical Proficiency	- Very comfortable	Young Tech-Savvy Professionals	Technically Proficient Managers	Fact-Checkers and Analysts	**Robustness**: Prefers advanced, reliable tools that can be trusted without needing extensive oversight.
	- Somewhat comfortable	Mid-career Professionals (open to training)	Operational Staff with Basic Technical Skills	General Media Professionals	**Accuracy**: Values tools that enhance accuracy and efficiency, especially in routine tasks.
	- Not comfortable at all	Experienced Sceptical Professionals	Operational Staff needing support	Sceptical Journalists with low technical skills	**Fairness**: Emphasizes the need for clear, transparent processes and assurances to build trust.
Attitudes towards AI	- In favor	Young Tech-Savvy Professionals	Technically Proficient Managers	Open Media Professionals	**Robustness**: Supports innovative tools that are adaptable and reliable in various situations.
	- Sceptical	Experienced Sceptical Professionals	Sceptical Managers	Sceptical Journalists	**Fairness**: Requires AI systems that provide detailed explanations and maintain ethical standards.
	- Neutral	Mid-career Professionals (adaptable but cautious)	Operational Staff with Basic Technical Skills	Neutral Media Professionals	**Accuracy**: Needs balanced tools that are accurate but not overly complex, focusing on usability.
Trust and Transparency Requirements	- Very important	Experienced Sceptical Professionals	Experienced Managers requiring transparency	Fact-Checkers and Analysts	**Fairness**: High demand for transparency and explainability to ensure trust and accountability.
	- Somewhat important	Young Tech-Savvy Professionals or Mid-career Professionals	Technically Proficient Managers	General Media Professionals	**Accuracy**: Interested in tools that are both efficient and provide clear, reliable results.
	- Not important	Young Tech-Savvy Professionals focused on innovation	Operational Staff focused on efficiency	Media Professionals focused on speed or innovation	**Robustness**: Less concerned with transparency; prioritizes functional reliability and speed.
Motivational Traits	- Innovation	Young Tech-Savvy Professionals	Technically Proficient Managers	Open Media Professionals	**Robustness**: Embraces new, reliable technologies that can handle innovative applications.
	- Efficiency	Technically Proficient Managers	Operational Staff with Basic Technical Skills	Media Professionals focused on output	**Accuracy**: Values precise and efficient tools that streamline workflows and improve productivity.

	- Ethical considerations	Experienced Sceptical Professionals	Sceptical Managers	Sceptical Journalists and Fact-Checkers	**Fairness**: Prioritizes tools that adhere to ethical standards and provide transparent operations.
	- Other (e.g., patient care, safety)	Experienced Sceptical Professionals	Experienced Managers	Analysts or Fact-Checkers with focus on integrity	**Fairness**: Emphasizes safety and integrity, requiring tools that are trustworthy and verifiable.
Responsiveness to Uncertainty	- Adapt well to uncertainty	Young Tech-Savvy Professionals	Technically Proficient Managers	Flexible Media Professionals	**Robustness**: Needs tools that are reliable even in uncertain or changing environments.
	- Prefer predictable environments	Experienced Sceptical Professionals	Sceptical Managers or Operational Staff	Sceptical Journalists	**Fairness**: Prefers AI systems that provide consistent, explainable, and ethical outputs.
	- Avoid uncertainty when possible	Operational Staff needing structured environments	Operational Staff with Basic Technical Skills	Media Professionals requiring structured work environments	**Accuracy**: Seeks tools that offer clear and structured outputs, minimizing ambiguity.
Commitment to Ethical Standards (Disinformation Sector)	- Extremely important	Not applicable	Not applicable	Sceptical Journalists and Fact-Checkers	**Fairness**: High priority on ethical standards and transparency to ensure integrity.
	- Moderately important	Not applicable	Not applicable	General Media Professionals	**Accuracy**: Balances ethics with functionality, requiring accurate and dependable tools.
	- Not important	Not applicable	Not applicable	Open Media Professionals	**Robustness**: Focuses on effective results over transparency or ethical considerations.

The proposed matrix and clustering methodology for grouping individuals based on user preferences and trustworthiness is derived from the qualitative analysis of data collected during the co-creation workshops.

The mapping to clusters though shows a variety in the allocation of groups/ 'personas' as these were expressed during the workshops. Ensuring the consistency of the clusters here is a refined allocation and definition:

Table 37: Mapping to clusters

Healthcare	
Description	**Cluster**
Young Professionals	Young, Tech-Savvy Professionals
Experienced Professionals	Experienced, Sceptical Professionals
Technically Proficient Managers	Technically Proficient Managers
Operational Staff with Basic Technical Skills	Operational Staff with Basic Technical Skills
Port Management	
Technically Proficient Managers	Technically Proficient Managers
Operational Staff with Basic Technical Skills	Operational Staff with Basic Technical Skills
Experienced Managers	Experienced Managers

Operational Staff Needing Support	Operational Staff Needing Support
Media	
Young Media Professionals	Fact-Checkers and Analysts
Experienced Journalists or Analysts	Fact-Checkers and Analysts
General Media Professionals	Sceptical Journalists or Fact-Checkers and Analysts
Sceptical Journalists with Low Technical Skills	Sceptical Journalists
Open Media Professionals	Fact-Checkers and Analysts
Neutral Media Professionals	Sceptical Journalists
Media Professionals Focused on Speed or Innovation	Fact-Checkers and Analysts

While existing literature suggests that individuals can be categorized and profiled according to their traits, human preferences often deviate from the straightforward, linear approach we propose. For instance, a participant may be both experienced and sceptical yet very comfortable with technology. This combination does not align into the matrix, highlighting the need to carefully consider both their comfort with technology and their overall attitude toward

In such a situation, the analysis shows an Experienced and Sceptical user which typically aligns with a preference for Fairness. Users who are sceptical often demand high transparency and ethical considerations from AI systems. They want clear explanations and assurances that the technology is being used responsibly. And Very Comfortable with Technology aligning with a preference for Robustness. These users trust technology more and are confident in its use. They might be more open to innovative applications and less concerned about transparency, as they feel capable of understanding or managing the technology.

Based on this analysis we propose to follow one the recommended approaches below (with Approach B as more accurate and preferable):

Approach A: the mapping and clustering can be informed by the analysed results of the workshops as these presented in D3.1 where we concluded that in relation to trustworthiness preferences, in the Healthcare sector is **Robustness**, in the media sector and port management sector is **Accuracy.**

Approach B: Given the combination of being experienced and sceptical but also very comfortable with technology, the responder could potentially align with either Fairness or Robustness. However, the deciding factor here is the balance between their scepticism and comfort with technology:

Primary Preference (Fairness): If the scepticism strongly influences their decision-making, the user might prioritize Fairness over Robustness. This would mean that even though they are comfortable with technology, their experience and scepticism lead them to prioritize ethical considerations, transparency, and the accountability of AI systems.

Secondary Preference (Robustness): If the user's comfort with technology significantly reduces their scepticism, making them more open to using advanced AI systems despite their experience, they might lean towards Robustness. This would imply that their familiarity with technology overrides some of their scepticism, leading them to prioritize reliable and efficient tools over transparency and ethical assurances.

To determine more accurately whether an experienced and sceptical user who is very comfortable with technology should be mapped to Fairness or Robustness, further probing questions could be asked to determine which attribute—scepticism or comfort with technology—has a more significant influence on their preferences. This approach addresses also the cases where alignment can relate to other combinations: Fairness/ Accuracy and Robustness/ Accuracy:

Table 38: Probing Question to resolve conflicting TW allocation

Fairness vs Accuracy	
Transparency vs Precision: Would you prefer an AI system that provides a fully explainable decision-making process, even if it means slightly lower accuracy, or one that consistently	**a)** Prioritize transparency and explainability, even if accuracy is slightly lower (**Fairness**) **b)** Prioritize high accuracy, even if transparency is limited (**Accuracy**)

delivers highly accurate results but with limited transparency	
Bias vs Performance: If an AI model showed slightly better performance but had the potential for hidden biases, would you still prefer to use it over a model that has lower performance but is designed to be bias-free?	**a)** I prefer an AI that is bias-free, even if it has slightly lower performance (**Fairness**) **b)** I prefer the higher-performing AI, even if it may contain hidden biases (**Accuracy**)
Trust in AI decisions: Would you trust an AI system that is known for making the most accurate predictions but does not always provide a clear justification for its decisions?	**a)** I need AI to provide clear justifications for its decisions (**Fairness**) **b)** I trust AI based on its high accuracy, even without justifications (**Accuracy**)
Robustness vs Accuracy	
Handling unexpected situations: Would you prefer an AI system that maintains reliable performance under different circumstances, even if its absolute accuracy is slightly lower, or one that delivers the highest accuracy under standard conditions but may struggle in uncertain scenarios?	**a)** Prefer a reliable AI that works under all conditions, even with slightly lower accuracy (**Robustness**) **b)** Prefer a highly accurate AI under normal conditions, even if it struggles with uncertainty (**Accuracy**)
Error tolerance: Would you rather have an AI that never makes small errors but occasionally fails in extreme conditions, or one that may have minor inaccuracies but performs consistently across all scenarios?	**a)** AI should be consistent in all situations, even if it makes minor errors (**Robustness**) **b)** AI should be highly accurate but might fail in extreme cases (**Accuracy**)
System performance: Would you prioritize an AI model that remains stable even under stressful and unpredictable conditions, even if it is not the absolute best at precision?	**a)** I prioritize stability over perfect accuracy (**Robustness**) **b)** I prioritize precision even if it makes the system less stable (**Accuracy**)
Fairness vs Robustness	
Scepticism and Trust: Even though you are comfortable with technology, how important is it for you to understand the inner workings and decision-making process of AI systems you use? **(same with T5 2.1)**	**a)** I need to understand AI decisions for trust (**Fairness**) **b)** As long as it works reliably, I don't need full transparency (**Robustness**)
Efficiency vs Ethical guidelines: How do you weigh the need for ethical guidelines and transparency against the efficiency and reliability of AI tools? **(same with T5 2.3)**	**a)** I prioritize ethical guidelines and transparency over efficiency (**Fairness**) **b)** I prioritize efficiency and reliability over transparency (**Robustness**)
AI Recommendations & Uncertainty: If an AI system made an unexpected recommendation that you didn't fully understand, would you trust it based on its past reliability, or would you require a detailed explanation before using its recommendation? **(same with T5 2.5)**	**a)** Require a detailed explanation before trusting AI (**Fairness**) **b)** Trust AI based on past reliability, even without an explanation (**Robustness**)

Step-by-Step Guide on Using the Matrix

Collect User Responses:

Start by gathering responses from users through a chatbot, survey, or direct interaction. The questions should cover areas such as professional experience, technical proficiency, attitudes towards AI, trust and transparency requirements, motivational traits, responsiveness to uncertainty, and commitment to ethical standards.

Map Responses to Clusters:

For each question, use the table to map the user's responses to the appropriate cluster in their sector. For example, if a user in the healthcare sector has over 15 years of experience and is sceptical about AI, they would be mapped to the "Experienced Sceptical Professionals" cluster.

Determine Trustworthiness Preference:

Once a user is assigned to a cluster, refer to the "Refined Trustworthiness Preference" column in the table. This column indicates the trustworthiness attribute (Fairness, Accuracy, Robustness) that is most relevant for that cluster based on the profiling methodology. For example, "Experienced Sceptical Professionals" in healthcare prioritize **Fairness** due to their emphasis on transparency and ethical considerations.

Practical Example of Using the Table

Scenario: Implementing AI in a Healthcare Organization

User Interaction:

A chatbot asks a healthcare professional, "How many years of experience do you have in your field?" and "How comfortable are you with using AI in your work?"

User Response:

The user responds, "Over 15 years of experience" and "I am somewhat comfortable with AI."

Mapping Responses:

According to the table, "Over 15 years of experience" and "somewhat comfortable with AI" places the user in the "Experienced Skeptical Professionals" cluster in the healthcare sector.

Determining Trustworthiness Preference:

The table indicates that this cluster prefers **Fairness**. This means they value transparency and ethical considerations in AI tools.

References

1. Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. Qualitative Research in Psychology, 3, 77–101.
2. Lambert, S.I., Madi, M., Sopka, S. *et al.* An integrative review on the acceptance of artificial intelligence among healthcare professionals in hospitals. *npj Digit. Med.* **6**, 111 (2023). https://doi.org/10.1038/s41746-023-00852-5
3. Steerling E, Siira E, Nilsen P, Svedberg P, Nygren J. Implementing AI in healthcare-the relevance of trust: a scoping review. Front Health Serv. 2023;3:1211150. Published 2023 Aug 24. doi:10.3389/frhs.2023.1211150
4. Challen R, Denny J, Pitt M, et al Artificial intelligence, bias and clinical safety BMJ Quality & Safety 2019;28:231-237.
5. Buck C, Doctor E, Hennrich J, Jöhnk J, Eymann T. General Practitioners' Attitudes Toward Artificial Intelligence-Enabled Systems: Interview Study. J Med Internet Res. 2022;24(1):e28916. Published 2022 Jan 27. doi:10.2196/28916
6. Jović M, Tijan E, Brčić D, Pucihar A. Digitalization in Maritime Transport and Seaports: Bibliometric, Content and Thematic Analysis. *Journal of Marine Science and Engineering*. 2022; 10(4):486. https://doi.org/10.3390/jmse10040486
7. Frechette, C., Scanlan C. and Scanlan, C. (2024) Two journalists talk to the bots - who talk back - about the pros and pitfalls of ai, Nieman Storyboard. Available at: https://niemanstoryboard.org/stories/journalism-ethics-artificial-intelligence-sources-fact-checking/ (Accessed: 03 September 2024).
8. Smith M. AI in journalism: how would public trust in the news be affected? | YouGov. yougov.co.uk. Published April 11, 2024. https://yougov.co.uk/technology/articles/49105-ai-in-journalism-how-would-public-trust-in-the-news-be-affec
9. K. Kioskli and N. Polemi, "Measuring Psychosocial and Behavioural Factors Improves Attack Potential Estimates," 2020 15th International Conference for Internet Technology and Secured Transactions (ICITST), London, United Kingdom, 2020, pp. 1-4, doi: 10.23919/ICITST51030.2020.9351343.
10. Hunady, J., Fendekova, J. and Orviska, M., 2020. Attitudes and trust to recent digital technologies: Using standards to improve them. In *25th EURAS Annual Standardisation Conference–Standards for Digital Transformation: Blockchain and Innovation, Glasgow, Scotland, June* (pp. 10-12).
11. Wooldridge, M., & Jennings, N. R. (1995). Intelligent agents: Theory and practice. The knowledge engineering review, 10(2), 115-152.

12. Wysocki, O., Davies, J.K., Vigo, M., Armstrong, A.C., Landers, D., Lee, R. and Freitas, A., 2023. Assessing the communication gap between AI models and healthcare professionals: Explainability, utility and trust in AI-driven clinical decision-making. *Artificial Intelligence*, *316*, p.103839.