AI4Gov

Trusted AI for Transparent Public Governance fostering Democratic Values

Deliverable 2.4

Reference Architecture and Integration of AI4Gov Platform V2

31-12-2024

Version 1.0

Funded by the European Union	*Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the Agency. Neither the European Union nor the granting authority can be held responsible for them.*

PROPERTIES	
Dissemination level	Public
Version	1.0
Status	Final
Beneficiary	UPRC
License	This work is licensed under a Creative Commons Attribution-No Derivatives 4.0 International License (CC BY-ND 4.0). See: https://creativecommons.org/licenses/by-nd/4.0/

AUTHORS		
	Name	**Organisation**
Document leader	George Manias	UPRC
Participants	Dimitris Kotios	UPRC
	Spiros Batziopoulos	MAG
	Eri Goga	VIL
	Danai Kyrkou	VIL
Reviewers	Xanthi Papageorgiou	UBI
	Fabiana Fournier	IBM

VERSION HISTORY				
Version	**Date**	**Author**	**Organisation**	**Description**
0.1	08/08/2024	George Manias, Dimitris Kotios	UPRC	Initial draft version and ToCs
0.2	18/08/2024	George Manias, Dimitris Kotios	UPRC	Input in Sections 3 and 4
0.3	31/08/2024	George Manias	UPRC	Internal release in the context of Milestone No9
0.4	15/11/2024	George Manias	UPRC	Input update in Sections 1, 2, and 3
0.5	20/11/2024	Spiros Batziopoulos	MAG	Input in Section 5
0.6	25/11/2024	Danai Kyrkou, Eri Goga	VIL	Input in Sections 4.3
0.7	10/12/2024	George Manias	UPRC	Input in Sections 2 and 6
0.8	13/12/2024	George Manias	UPRC	Preparation for Internal Review
0.9.1	16/12/2024	Xanthi Papageorgiou	UBI	1st Internal Review
0.9.2	17/12/2024	Fabiana Fournier	IBM	2nd Internal Review
1.0	20/12/2024	George Manias	UPRC	Final version for submission

Table of Contents

List of figures

List of Tables

Abbreviations

Abbreviation	Description
AI	Artificial Intelligence
CI/CD	Continuous Integration/Continuous Deployment
D	Deliverable
DevOps	Development Operations
DGF	Data Governance Framework
DIDs	Decentralised Identifiers
DPB	Diputación Provincial de Badajoz
EBSI	European Blockchain Services Infrastructure
EU	European Union
GDPR	General Data Protection Regulation
GiST	Generalized Search Tree
GIN	Generalized Inverted Index
HRF	Holistic Regulatory Framework
JSI	Jozef Stefan Institute
JSONB	JavaScript Object Notation Binary
K8s	Kubernetes
LLM	Large Language Model
ML	Machine Learning
MT	Ministry of Tourism
NLP	Natural Language Processing
OECD	OECD Organisation for Economic Cooperation and Development
PRT	Policy Recommendation Toolkit
SDG	SDG Sustainable Development Goal
SLA	Service-Level Agreement
SSC	Sustainable Smart Cities
SWOT	Strengths, Weaknesses, Opportunities, and Threats
UBI	Ubitech
UPRC	University of Piraeus Research Centre
VUF	Virtualized Unbiasing Framework
VVV	Municipality of Vari-Voula-Vouliagmeni
XAI	eXplainable AI

Abstract

This is the second deliverable of the series that seeks to detail and specify the reference architecture and integration approach of the AI4Gov project and its integrated platform. In that context, the objective of this deliverable has a two-fold interpretation. On the one hand, it seeks to provide the consortium with a common high-level view about the whole solution and architecture. The details of each software artefact are delegated to the respective leader and will be further described in the related deliverables. On the other hand, it also seeks to describe and present the system and technical requirements that are being imposed by the integrated AI4Gov platform. The purpose of this series is to track those requirements throughout the project and update them during the progress of the project. To this end, a bottom-up approach is implemented that aims to identify and analyse the technical requirements with respect to the technical work packages that focus on the platform's technological needs. Technical requirements are also interlinked with the user requirements and scenarios as presented in the context of D6.2 – "Specification of UC Scenarios and Planning of Integration and Validation Activities V2", published in M18. The result of the above analysis consists in a list of measurable and unambiguous requirements that drove the design of the overall architecture of the AI4Gov platform, focusing on serving all different needs of the various use cases of the project.

Moreover, as the project progresses, this updated version provides an improved design of the platform's reference architecture and supports its software developers to adjust and revise the design, implementation, internal architecture and workflows of their components accordingly, by also introducing the key aspects of its overall integration strategy. The latter is applied by following the principles of an agile methodology presented in this deliverable and more specifically in Section 3. Moreover, for the platform to keep track of the latest technological advances, a state-of-the-art analysis has been performed regarding the major technologies that are envisioned to be exploited.

This deliverable is being released in M24 mainly focusing on the update of the specification and design of the project's reference architecture, and the establishment and description of the integration approach. The final updated version of this series of deliverables will be released in M36.

1 Introduction

1.1 Purpose and scope

This document introduces the second iteration of AI4Gov platform's reference architecture and its integration approach. Thus, the purpose of this document is two-fold. First, it provides an updated list of measurable and specific system requirements that drive the design of the architecture of the AI4Gov platform and are used as the basis for the implementation of the relevant functionalities that are offered by the various software components of the platform. Second, it provides the updated reference architecture and integration approach that drive the interactions among the different software artefacts developed by the AI4Gov partners. The latter is also a concrete list of the project's components and offers an overall analysis of the deployments and implementations that have been applied during the lifetime of the project.

This document is the product of the work carried out by the collaboration of T2.3 – "Reference Architecture Specification" and T2.4 – "Integration of AI4Gov Platform and Tools", in contrary to the initial version that mainly included the work from the first task, as T2.4 was initiated in M12 of the project. The main goals of these tasks are collaboratively to analyse the technical requirements and to conclude on agreements with the technical partners regarding the reference architecture and the integration approach that address the technical requirements as identified and presented in the context of this deliverable. All the component leaders were involved during the definition, refinement, and agreement on the common reference architecture and integration approach, as presented in this deliverable.

Starting with the analysis of the technical requirements, this document seeks to produce a measurable requirement set, which are tracked against the developments during the project lifecycle to ensure that the needs arising from the complex nature of AI4Gov are fully addressed and properly considered. Moreover, another important objective of both these tasks is to investigate and analyse the State-of-the-Art (SotA) techniques and approaches for the envisioned AI4Gov technologies and components. Both these two objectives represent a valuable input for the design of the platform's overall architecture and all project-related implementation activities. As a key content of this deliverable, this analysis seeks to address and highlight all the technical perspectives of the project and to act as a valuable input for its other technical-related tasks and its reference architecture.

Thus, as concerns the project's reference architecture, the main purpose of this document is to define several guidelines among the consortium and present a common view about the integrated AI4Gov platform. In parallel, its integration approach seeks to demonstrate that the combination of its different tools ensures a well-structured, scalable, and secure architecture design to support the platform's diverse needs. The main topics that this version of the document addresses concern the variety of software artefacts that are being developed, the communication pattern used, the diversity of graphical user interface provided, the infrastructure needed for testing and validation, the envisaged deployment, the integrated approach that is followed, and the key tools utilized to govern, manage, and monitor the overall deployments and implementations.

However, it should be noted that this document does not address the internal design choices of the components, or the different nature of the data to be ingested or analysed, but it aims to clarify the impact of the gathered requirements on the designed components and to highlight and describe some useful interactions. The next and final version of this series of deliverables, i.e., D2.5 (due on M36), will finalize the complementary work carried out in the context of the two aforementioned tasks, while it will provide the final details with regards to details related to resources allocation, usage, and will provide any further and last updates on the reference architecture and integration approach of the project.

1.2 Linking with other Tasks and Deliverables

The analysis and elicitation of the technical requirements of the corresponding technologies have been carried out considering the needs and concerns that have been identified by the current communities, end-users and related actors that are related to the AI4Gov pilots and providers, and as they have been reported as user requirements in the context of D6.1 – "Specification of UC Scenarios and Planning of Integration and Validation Activities V1" and D6.2 – "Specification of UC Scenarios and Planning of Integration and Validation Activities V2". As a result, the analysis that has been made in the context of current deliverable interlinks the user requirements (stakeholder requirements), with the technical requirements (system and software requirements). Moreover, this document recalls the principles and the assumptions made about data provenance, data security, and data privacy aspects as they are reported in D1.2 – "Data Management Plan V1" to further support the analysis of the technical requirements and the design of the reference architecture. It should be noted that all partners, and especially those that lead and integrate to the technical-related Work Packages (WPs) of the AI4Gov project, i.e., WP3 and WP4, provide content related to their contribution. The content and outcomes of this deliverable act as input for the design and implementation of the project key components, and as guidelines for the implementation of their internal workflows, as well as for the integration approaches and techniques that should be followed throughout the lifecycle of the project.

1.3 Updates since the previous version

In this section, the major updates incorporated in the context of this deliverable are highlighted, in comparison to the first version of the series of deliverables related to the Architecture and integration approach of the AI4Gov project.

Firstly, in Section 3 in two new subsections the updates of the architecture, the integration approach and their alignment with the recommendations received after the end of the first reporting period of the project are highlighted. In parallel, one of the key propositions of this deliverable is the alignment of the architectural design with the guidelines and mandates of the Holistic Regulatory Framework (HRF) and Data Governance Framework (DGF) of the project. The latter is also incorporated in Section 3 to drive and complete the overall update of the architecture as presented in Section 4.

Hence, in Section 4 an updated version of the project reference architecture is introduced to reflect on the updates, strategical decisions, and needs identified during the previous months of the project. This updated version of the architecture encapsulates revisions with regards to the specification of new components, new integration approaches between them, and the establishment of a robust architectural scheme that acts as the point of reference for the design and implementation of technical components.

Furthermore, the integration approach and the deployment, installation, configuration, and management of the K8s cluster is presented in detail in Section 5 to showcase the overall strategy of the project towards the implementation of a concrete orchestration mechanism for the deployment, utilization and integration of the different components. On top of this, the Interim Repository, as presented in the initial version of this series of deliverables, is updated to the utilization of a Data Lake that enhances the overall management, storing, and privacy of the data.

Finally, the different components of the project have been revised in terms of their functionalities and requirements, as additions and updates were identified and applied on them during the progress of the project and based on the updated pilot needs and scenarios identified under the scopes of D6.2 – "Specification of UC Scenarios and Planning of Integration and Validation Activities V2".

1.4 Document structure

The rest of this deliverable is organised and structured into five main sections. To better understand the software technology requirements, this deliverable provides in Section 2 a state-of-the-art analysis and specifies a list of the baseline technologies that are intended to be used in the development and implementation of the platform. The latter will result in the identification and provision of the components' technical requirements, also documented in the contents of this Section.

Afterwards, Section 3 has a fourfold objective:

- to provide a global overview on the adopted architecture definition approach;
- to describe, both for technical and non-technical audience, how AI4Gov fits into the existing environments, who and how will use the system;
- to present the main functionalities and components of the system;
- and to highlight the strategic decisions taken in alignment with the recommendations received from the first review of the project.

Section 4 presents the envisaged software artefacts and components that will compose the AI4Gov solution and provides the updates

Afterwards, Section 5 depicts the infrastructure needed by AI4Gov in order to perform its tasks and provide its features, as well as provides a detailed description and the plan followed in the context of project's integration strategy and approach.

Finally, Section 6 concludes this document and states the future work for the next version of this deliverables' series.

As previously noted, the next version of this deliverables' series, i.e., D2.5 (due on M36), will finalize the complementary work carried out in the context of T2.3 – "Reference Architecture Specification" and T2.4 – "Integration of AI4Gov Platform and Tools" with regards to the integration approach and all relevant implementations and interconnections. Thus, the final updates and revisions as concerns the architecture and the integration status will follow in the next and final version.

2 Technical Components and Requirements

This section provides a list of the initial technical component requirements for the AI4Gov project. It should be noted that a specific template was utilized under this scope to facilitate the process of collection from the project partners, as introduced in APPENDIX A – Architecture and Infrastructure Discussions. These requirements are related to specific components, which can be either a program, a software component, an existing product that will be used as part of the overall platform, or a set of combinations of all the above, that implements a specific functionality and provides a set of capabilities via well-defined interfaces. It should be noted that in the below subsections and especially in the "Reference Scenarios" input fields, the below custom coding is used for the mapping of technical components and requirements, as introduced in the context of D6.2 – "Specification of UC Scenarios and Planning of Integration and Validation Activities V2" scenarios. The latter is applied for the sake of brevity and for reasons of clarity and overall consistency in this document.

- **P1UC1**: Diputación Provincial de Badajoz (DPB) pilot - Use Case #1 Water management cycle – drinking water
- **P1UC2**: Diputación Provincial de Badajoz (DPB) pilot - Use Case #2 Water management cycle – Sewage water
- **P2UC1**: Joseph Stefan Institute (JSI) pilot - Use Case #1 IRCAI global top 100 projects
- **P2UC2**: Joseph Stefan Institute (JSI) pilot - Use Case #2 SDG Observatory
- **P2UC3**: Joseph Stefan Institute (JSI) pilot - Use Case #3 Bias analysis in the area of alcohol abuse in traffic – Slovenia
- **P2UC4:** Joseph Stefan Institute (JSI) pilot - Use Case #4 OECD policy documents analysis
- **P3UC1:** Municipality of Vari-Voula-Vouliagmeni (VVV) & Ministry of Tourism (MT) pilot - Use Case #1 Parking tickets monitoring
- **P3UC2**: Municipality of Vari-Voula-Vouliagmeni (VVV) & Ministry of Tourism (MT) pilot - Use Case #2 Waste management - Pay As You Throw (PAYT)

2.1 Decentralized blockchain-based infrastructure

2.1.1 Goals and Objectives

The main goal of the decentralised blockchain infrastructure (referred to as blockchain for shortness), is to provide transparency and traceability for storage and business logic execution. From a pilot execution point of view, the main concrete objectives can be defined as:

- Capability to ensure documents verification and versioning. By using blockchain, the information and the history of documents will be visible to all; this can be leveraged both by policymakers to ensure that they always have access to correct and verified current and historical data, and by programs that act on these data.
- Transparency in the execution of code running based on data shared via the blockchain. Tools such as the Policy Recommendation Toolkit can run as smart contracts

guaranteeing to all peers of the network that output was generated using universally agreed business logic, as this is encoded in the code of the smart contract.

2.1.2 State of the Art

Although originally created in the context of cryptocurrency, blockchain technology has been found to be useful in many other areas such as e-governance and supply chain (Naef, 2024). Most early, blockchain technologies were based on the concept of the public blockchain; in a public blockchain, everyone can access the blockchain using a public-private key pair (i.e., a wallet) that allows them to create and sign transactions. As validation of transactions must be performed by all or a majority of the peers, an incentive must be given to peers so that they participate in this validation process. A token that is mined by the "mining" peers is the typical reward mechanism that is implemented by most public blockchains, such as Bitcoin. Instead of only implementing direct transactions between peers, it is possible to also implement special programs that run on the blockchain and allow more complex transactions. Starting from the Ethereum network, fully Turing complete programs, called smart contracts, can run on the blockchain; the byte code of these contracts is written on the blockchain, and any peer can invoke these contracts by using their address.

Public blockchains are difficult to use in business-oriented scenarios. Firstly, the tokenisation schemes used by each blockchain may not be suitable in different business scenarios. Secondly, most business scenarios involve the participation of organisations and users with a verifiable identity, not being able to use an anonymous account. To address these limitations, permissioned blockchains which verify users' identity (e.g., through a certificate authority and a member's list) have been implemented. As validation is performed by special nodes called "orderers", through a variety of validation schemes (e.g., single node, majority voting etc.), a permissioned blockchains typically do not possess an innate token mechanism. The cost of running the "orderers" is taken by the organisation(s) using the blockchain, who, of course, have a strong incentive to do so, especially if the blockchain covers their business needs. Similarly, various implementations of private blockchains exist. Hyperledger Fabric, developed by the Linux Foundation, is an example of an implementation, used widely through a variety of public and private institutions. In the public services domain, in particular, EU has already deployed its own blockchain infrastructure based on Hyperledger Fabric. This infrastructure is called EBSI and has already demonstrated the capability of cross-border certification of official documents (Tan, 2023).

2.1.3 Background technology

Key baseline technologies that will be utilised for developing the project's blockchain infrastructure are listed in Table 1.

Table 1: T3.1 - Background Technology

Technology Name	Technology Description	Advancements / Usage
eIDAS authenticator	A prototype for implementing authentication and attribute authentication based on eIDAS regulation, using HyperLedger Fabric	The prototype will be developed into a full system that can be used as an entry point for user authentication, authorisation and credential management.
HyperLedger Fabric Orchestrator	A set of scripts for rapidly defining topology of a blockchain network based on HyperLedger Fabric and deploying it into a set of development machines or in a production environment	The orchestrator will be used for deploying all necessary blockchain infrastructure needed for the project.
Open Democracy DAO	A Decentralized Autonomous Organization realized by three processes that cover all main aspects of the service, mainly voting, consulting (opinion forming) and auditing	Utilized in align with the business logic implemented in smart contracts using the underlying Decentralized Data Governance infrastructure

2.1.4 Component to User Requirements

Below are the nine (9) tables, i.e., Tables 2 – 10, that map the respective User Requirements with the Blockchain Infrastructure that seeks to be designed and implemented in the context of the AI4Gov project.

Table 2: T3.1 - 1st Component Functionality

Title	Description
ID	U-REQ-T3.1-01
Source User Requirement	Access to current and historical drinking water quality reports and verification of the report's originator identity.

Use case quote	**P1UC1** As a worker at the municipal consortium for water management, I want to know as quickly as possible of any potential problems in the drinking water system, without having to check manually so that waste of water, power and time can be reduced all along the water treatment cycle.
Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	N/A

Table 3: T3.1 - 2nd Component Functionality

Title	**Description**
ID	U-REQ-T3.1-02
Source User Requirement	Access to current and historical sewage water quality reports and verification of the report's originator identity.
Use case quote	**P1UC1** As a worker at the municipal consortium for water management, I want to know as quickly as possible of any potential problems in the sewage water system, without having to check manually so that waste of water, power and time can be reduced all along the water treatment cycle.
Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	This user requirement mapping is very similar to U-REQ-T3.1-0.1 and it is mainly differentiated in the type of water being treated

Table 4: T3.1 - 3rd Component Functionality

Title	Description
ID	U-REQ-T3.1-03
Source User Requirement	Decentralized storage of reports and attachment of report metadata (e.g., annotations).
Use case quote	**P2UC1** As a reviewer of the IRCAI Top 100 program, I need to evaluate ethical considerations and bias in the submitted projects, so that we identify the AI projects that could be showcased for others (in terms of ethics).
Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	N/A

Table 5: T3.1 - 4th Component Functionality

Title	Description
ID	U-REQ-T3.1-04
Source User Requirement	Smart contracts to deliver data for visualization and rankings, based on blockchain anchored reports.
Use case quote	**P2UC2** As a decision maker, I want to be able to compare and rank different countries and regions to see their achievements about the SDG of my choice, so that I can identify the best performing countries and follow their practices.
Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	N/A

Table 6: T3.1 - 5th Component Functionality

Title	Description
ID	U-REQ-T3.1-05
Source User Requirement	Transparent storage and retrieval (via smart contracts) of the violation data together with the origin of the information.
Use case quote	**P3UC1** As a municipal officer in charge of the Municipal police, I want to know which are the areas, days and hours that violation of parking rules is observed, so that I allocate municipal police staff and equipment in an optimum way.
Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	N/A

Table 7: T3.1 - 6th Component Functionality

Title	Description
ID	U-REQ-T3.1-06
Source User Requirement	Transparent storage and retrieval (via smart contracts) of the violation data together with the origin of the information.
Use case quote	**P3UC1** As a member of the Municipal Council, I want to know which are the areas, days, and hours that violation of parking rules is observed, so that I allocate necessary funds and resources to address the problem.
Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	This user requirement mapping is nearly identical to U-REG-T3.1-

	05. The data are the same; only the relevant actions are differentiated which are not part of the specific user requirements.

Table 8: T3.1 - 7th Component Functionality

Title	Description
ID	U-REQ-T3.1-07
Source User Requirement	Transparent storage of Waste Management and Recycling Unit data together with resource allocation algorithms (via smart contracts).
Use case quote	**P3UC2** As a municipal officer in charge of the Municipal police, I want to know which are the areas, days, and hours that violation of parking rules is observed, so that I allocate municipal police staff and equipment in an optimum way.
Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	N/A

Table 9: T3.1 - 8th Component Functionality

Title	Description
ID	U-REQ-T3.1-08
Source User Requirement	Resource allocation algorithms (via smart contracts) for the Pay As You Throw System.
Use case quote	**P3UC2** As a member of the Municipal Council, I want to know which are the optimum areas and to predict the financial outcomes regarding the expansion of the Pay As You Throw System, so that I allocate necessary funds and resources.

Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	This user requirement mapping is very similar to U-REQ-T3.1-0.7

Table 10: T3.1 - 9th Component Functionality

Title	**Description**
ID	U-REQ-T3.1-09
Source User Requirement	Delivery of smart contracts utilised to facilitate actors' contributions to the AI4Gov distributed ledger.
Use case quote	**P1UC1** As a member of the local administration, I want to have a clear picture of the main areas for improvement within the drinking water treatment system so that I have the most information available when making decisions on infrastructure improvement and long-term strategies.
Generic/Specific	Generic
Task/Component	T3.1 - Decentralized Data Provenance and Reliability
Lead partner	UBI
Notes	Public actors and policy makers can share information and state as part of AI4Gov through secure state synchronization techniques.

2.1.5 Component to Technical Requirements

In Tables 11 – 13, the technical requirements are listed that will be covered through the design and implementation of the Blockchain Infrastructure. These requirements are mapped to specific framework functionalities and further address user requirements as identified in the context of pilot scenarios and further mapped above with this component.

Table 11: T3.1 - 1st Technical Requirement

Section	Description
Technical requirement code	T-REQ-T3.1-0.1
Type	FUNC
Short name	Authorization mechanism based on eIDAS.
Functionality ID	U-REQ-T3.1-01, U-REQ-T3.1-07, U-REQ-T3.1-08
Description & quantification	In case of access to decentralized data, eIDAS authorization provides means for cross-border user identification and user attributes management.
Additional information (source, characteristics, short description of the process)	Instead of using traditional organisation or country specific login mechanisms, the user is authenticated via eIDAS.
Priority	DES
Reference Scenarios	All
Success criteria	Successful login and attribute assignment

Table 12: T3.1 - 2nd Technical Requirement

Section	Description
Technical requirement code	T-REQ-T3.1-0.2
Type	DATA
Short name	Decentralized storage via anchoring to the blockchain files that are stored off-chain.
Functionality ID	U-REQ-T3.1-01, U-REQ-T3.1-02, U-REQ-T3.1-03, U-REQ-T3.1-05, U-REQ-T3.1-06, U-REQ-T3.1-07
Description & quantification	Data that need to be verified, monitored, and traced must be stored using a blockchain mechanism to ensure the above traits.

Additional information (source, characteristics, short description of the process)	Following standard practices, files are not stored directly in the blockchain, but are accessed via specific blockchain anchors that point to the files.
Priority	MAN
Reference Scenarios	All
Success criteria	Accessing the correct file and verifying its contents by sing the appropriate smart contract.

Table 13: T3.1 - 3rd Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T3.1-0.3
Type	FUNC
Short name	Ability to run globally verified and reproducible code via smart contracts.
Functionality ID	U-REQ-T3.1-01, U-REQ-T3.1-02, U-REQ-T3.1-03, U-REQ-T3.1-05, U-REQ-T3.1-07, U-REQ-T3.1-08
Description & quantification	Storage and delivery of commonly shared resources need to be performed in a way that ensures the endorsement of all to guarantee trust on data. Extraction of statistics and recommendations based on the data needs also to be transparent in the sense that each peer is able to reproduce the same results.
Additional information (source, characteristics, short description of the process)	Following standard practices, files are not stored directly in the blockchain, but are accessed via specific blockchain anchors that point to the files.
Priority	MAN
Reference Scenarios	All
Success criteria	Successful invocation of smart contract and validation from all relevant peers.

2.2 Policy Recommendation Toolkit

2.2.1 Goals and Objectives

The Policy Recommendation Toolkit (PRT), aims to offer organisations a high-level tool that will facilitate:

- Uploading of documents describing policies, which will be encoded into a JavaScript Object Notation (JSON) based descriptive language.
- Searching and viewing existing policy documents.
- Defining custom policies either by extending existing ones, or from scratch.
- The usage of a recommendation engine that will allow users to set hard and soft targets for new policies, where hard targets refer to specific, measurable, and enforceable goals in new policies (e.g., a 50% reduction in emissions by 2030), while soft targets are broader objectives meant to guide efforts without strict enforcement (e.g., promoting sustainability awareness). Afterwards, the recommendation engine will retrieve the set of policies that best matches the selected criteria.
- Visualisation of existing policies effectiveness against various defined criteria.

2.2.2 State of the Art

Technology-wise, the PRT is composed of two parts: an expert system part for recommendation and a representation one for illustrating entities, metrics, and rules. Expert systems have been widely used in enterprises and have been greatly enhanced by using Machine Learning. They have been employed in various domains and, in particular, in the domain of policy making, i.e., the PolicyCLOUD project has implemented a policy management system for tracking policies (Kyriazis, 2020).

For knowledge representation, Knowledge Graphs can be used to encode complex entities and relations. Rule languages, such as Drools or SWRL can also be embedded in Knowledge Graphs. This allows visual representation and explainability regarding the recommendation of the Toolkit.

To allow for transparent execution of the rule systems, the rules of the engine are to be encoded and executed via smart contracts so that recommendations, that are to be used by multiple organisations, can be validated by each of them. Regarding smart contracts, various Turing complete languages appropriate for different blockchain technologies exist. Solidity, which is used for the public Ethereum network, is historically the first of them. For permissioned blockchains, such as the one that is going to be used in AI4Gov, there exist various alternatives. The HyperLedger ecosystem supports multiple platforms for writing smart contracts (or chaincode as it is more precisely called within the context of the HyperLedger ecosystem) such as Go and Javascript, or even Solidity in the HyperLedger Besu platform.

2.2.3 Background technology

Key baseline technologies that will be utilised for developing the Policy Recommendation Toolkit are listed in the table below, i.e., Table 14.

Table 14: T3.3 - Background Technology

Technology Name	Technology Description	Advancements / Usage
Wastewater treatment DSS	A prototypical DSS for recommending best exploitation paths for wastewater extracted in small farming units.	The rule engine of the WasteWater DSS will be expanded to cover the domain of each pilot and it will be migrated into a smart contract to allow for concurrent and reproducible execution.

2.2.4 Component to User Requirements

Below follow the thirteen (13) tables, i.e., Tables 15 – 27, that map respective User Requirements with the PRT that seeks to be designed and implemented in the context of the AI4Gov project.

Table 15: T3.3 - 1st Component Functionality

Title	Description
ID	U-REQ-T3.3-01
Source User Requirement	Searching and retrieving documents relevant to existing policies.
Use case quote	**P1UC1** As a member of the local administration, I want to use large-scale data to evaluate the use of water throughout the province, so that I may pursue the correct policies and make the necessary budget allotments.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	N/A

Table 16: T3.3 - 2nd Component Functionality

Title	Description
ID	U-REQ-T3.3-02
Source User Requirement	Definition of new policies concerning drinking water management.
Use case quote	**P1UC1** As a member of the local administration, I want to use large-scale data to evaluate the use of water throughout the province, so that I may pursue the correct policies and make the necessary budget allotments.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	N/A

Table 17: T3.3 – 3rd Component Functionality

Title	Description
ID	U-REQ-T3.3-03
Source User Requirement	Recommendation of best policies based on set target KPIs.
Use case quote	**P1UC1** As a member of the local administration, I want to use large-scale data to evaluate the use of water throughout the province, so that I may pursue the correct policies and make the necessary budget allotments.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	N/A

Table 18: T3.3 - 4th Component Functionality

Title	Description
ID	U-REQ-T3.3-04
Source User Requirement	Searching and retrieving documents relevant to existing policies.
Use case quote	**P1UC2** As a member of the local administration, I want to use large scale data to evaluate the use of water throughout the province, so that I may pursue the correct policies and make the necessary budget allotments.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	Similar to U-REQ-T3.3-01 but applied for the sewage water case.

Table 19: T3.3 - 5th Component Functionality

Title	Description
ID	U-REQ-T3.3-05
Source User Requirement	Definition of new policies concerning sewage water management.
Use case quote	**P1UC2** As a member of the local administration, I want to use large-scale data to evaluate the use of water throughout the province, so that I may pursue the correct policies and make the necessary budget allotments.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	Similar to U-REQ-T3.3-02 but applied for the sewage water case.

Table 20: T3.3 - 6th Component Functionality

Title	Description
ID	U-REQ-T3.3-06
Source User Requirement	Recommendation of most relevant ethical and bias considerations.
Use case quote	**P2UC1** As an applicant to the IRCAI Top 100 projects, I need to understand which ethical and bias consideration I should consider whilest building my models, so that my created models are fair.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	N/A

Table 21: T3.3 - 7th Component Functionality

Title	Description
ID	U-REQ-T3.3-07
Source User Requirement	Rule based ranking engine, based on input scores.
Use case quote	**P2UC2** As a decision maker, I want to be able to compare and rank different countries and regions to see their achievements about the SDG of my choice, so that I can identify the best performing countries and follow their practices.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	N/A

Table 22: T3.3 - 8th Component Functionality

Title	Description
ID	U-REQ-T3.3-08
Source User Requirement	Searching and retrieving documents relevant to existing policies.
Use case quote	**P2UC2** As an expert for AI ethics, I would like to see which national policy documents are talking about AI ethics and are considering bias in AI, so that I can identify the current regulations and good practices.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	Similar to U-REQ-T3.3-01 and U-REQ-T3.3-04

Table 23: T3.3 - 9th Component Functionality

Title	Description
ID	U-REQ-T3.3-09
Source User Requirement	Ranking of policies based on relevance to dealing with bias in AI
Use case quote	**P2UC2** As a decision maker, I would like to identify best policy and legal practices dealing with bias in AI and AI ethics, so that I can follow and adapt those best practices for my country.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	N/A

Table 24: T3.3 - 10th Component Functionality

Title	Description
ID	U-REQ-T3.3-10
Source User Requirement	Rule-based strategy planner based on parking data information.
Use case quote	**P3UC1** As a member of the Municipal Council, I want to get information about the Parking problems in the city, so that I can facilitate the implementation of the Sustainable Urban Mobility Plan and Strategy2.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	N/A

Table 25: T3.3 - 11th Component Functionality

Title	Description
ID	U-REQ-T3.3-11
Source User Requirement	Rule based resource allocation.
Use case quote	**P3UC1** As a member of the Municipal Council, I want to know which are the areas, days, and hours that violation of parking rules is observed, so that I allocate necessary funds and resources to address the problem.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	N/A

Table 26: T3.3 - 12th Component Functionality

Title	Description
ID	U-REQ-T3.3-12
Source User Requirement	Fully autonomous citizen dApps deployment
Use case quote	**P1UC1, P1UC2, P3UC1, P3UC2** As a citizen, I want a fully autonomous application deployed on my Android device that provides real-time policy recommendations and personalized suggestions to address urban issues efficiently.
Generic/Specific	Specific
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	Focus on packaging and usability of dApps for Android devices.

Table 27: T3.3 - 13th Component Functionality

Title	Description
ID	U-REQ-T3.3-13
Source User Requirement	Autonomous citizen dApps functionality enhancement
Use case quote	**P1UC1, P1UC2, P3UC1, P3UC2** As a policy maker, I want a system that integrates fully autonomous citizen dApps to gather insights and feedback from Android users to refine urban policies and resource allocation.
Generic/Specific	Generic
Task/Component	T3.3 – Policy Recommendation Toolkit
Lead partner	UBI
Notes	Emphasis on citizen engagement and data-driven insights.

2.2.5 Component to Technical Requirements

In Tables 28 – 34 the technical requirements that will be covered through the design and implementation of the ***Policy Recommendation Toolkit*** *are listed*. These requirements are mapped to specific component functionalities and further address user requirements, as identified in the context of pilot scenarios.

Table 28: T3.3 - 1st Technical Requirement

Section	Description
Technical requirement code	T-REQ-T3.3-1
Type	FUN
Short name	Integration with the Policy Recommendation Engine
Functionality ID	U-REQ-T3.3-01, U-REQ-T3.3-03, U-REQ-T3.3-07
Description & quantification	The Policy Recommendation Frontend should communicate with the Policy Recommendation Engine to retrieve entities and policies (stored KPI's, Domains, Stakeholders, etc.).
Additional information (source, characteristics, short description of the process)	N/A
Priority	MAN
Reference Scenarios	All
Success criteria	Successfully store the created policy, its respective KPIs and rules.

Table 29: T3.3 - 2nd Technical Requirement

Section	Description
Technical requirement code	T-REQ-T3.3-2
Type	FUN
Short name	Integration with the Interactive Self-Explained

	Visualization Workbench
Functionality ID	U-REQ-T3.3-01, U-REQ-T3.3-02, U-REQ-T3.3-05, U-REQ-T3.3-06, U-REQ-T3.3-09
Description & quantification	The Policy Recommendation Frontend should communicate with the Interactive Self-Explained Visualization Workbench component to retrieve the list of available tools and visualizations and to send requests for analytics. It should also check the status of the pending jobs and retrieve the results from it.
Additional information (source, characteristics, short description of the process)	N/A
Priority	MAN
Reference Scenarios	All
Success criteria	Successfully visualize the results of the recommendations.

Table 30: T3.3 - 3rd Technical Requirement

Section	Description
Technical requirement code	T-REQ-T3.3-3
Type	FUN
Short name	Integration with blockchain and smart contracts
Functionality ID	U-REQ-T3.3-01, U-REQ-T3.3-02, U-REQ-T3.3-07, U-REQ-T3.3-10, U-REQ-T3.3-11
Description & quantification	Rules of the Policy Recommendation Engine to be encoded and executed via smart contracts
Additional information (source, characteristics, short description of the process)	N/A

Priority	MAN
Reference Scenarios	All
Success criteria	Successful encoding and execution of rules via the blockchain infrastructure.

Table 31: T3.3 - 4th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T3.3-4
Type	FUN
Short name	Knowledge Graphs Visualization
Functionality ID	U-REQ-T3.3-03, U-REQ-T3.3-04, U-REQ-T3.3-06, U-REQ-T3.3-08, U-REQ-T3.3-09
Description & quantification	Involves the utilisation graphical elements such as nodes, edges, and various visual cues to help policy makers better understand the relationships and connections between different pieces of information within the graph and related to their policies.
Additional information (source, characteristics, short description of the process)	N/A
Priority	MAN
Reference Scenarios	All
Success criteria	Knowledge Graphs Visualizations that can be used to encode complex entities and relations on policies.

Table 32: T3.3 - 5th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T3.3-5
Type	FUNC, USE
Short name	User Authentication & Authorization
Functionality ID	U-REQ-T3.3-01, U-REQ-T3.3-02, U-REQ-T3.3-04, U-REQ-T3.3-06, U-REQ-T3.3-08, U-REQ-T3.3-09
Description & quantification	PRT users should be able to authenticate using their credentials into the system (Login). The content will be dynamically changed depending on users' permissions (user role based).
Additional information (source, characteristics, short description of the process)	N/A
Priority	MAN
Reference Scenarios	All
Success criteria	A user supplies the credentials and enters (logins) into the platform. Logged in user can create a new policy model and submit it in the policy store.

Table 33: T3.3 - 6th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T3.3-6
Type	FUNC
Short name	Fully autonomous citizen dApps on Android
Functionality ID	U-REQ-T3.3-12, U-REQ-T3.3-13
Description & quantification	Development and deployment of autonomous citizen

	dApps packaged for Android devices, capable of delivering real-time policy recommendations, collecting user feedback, and integrating with smart contracts for process automation.
Additional information (source, characteristics, short description of the process)	Focus on usability, privacy-preserving mechanisms, and compatibility with blockchain-based policy rules.
Priority	MAN
Reference Scenarios	Urban policy recommendation, resource allocation, and citizen engagement scenarios.
Success criteria	Successful deployment of fully functional citizen dApps on Android devices with seamless integration with the policy recommendation toolkit and blockchain infrastructure.

Table 34: T3.3 - 7th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T3.3-7
Type	FUNC
Short name	Blockchain-based secure data exchange
Functionality ID	U-REQ-T3.3-02, U-REQ-T3.3-07, U-REQ-T3.3-13
Description & quantification	Implementation of blockchain-based mechanisms to securely exchange data between the citizen dApps, the Policy Recommendation Engine, and stakeholders, ensuring transparency and immutability of the exchanged data.
Additional information (source, characteristics, short description of the process)	Blockchain ensures data integrity and supports the execution of smart contracts to automate policy rules and citizen interactions.
Priority	MAN

Reference Scenarios	Secure communication in urban planning, fraud prevention in resource allocation, and automated policy enforcement scenarios.
Success criteria	Secure and transparent data exchange via blockchain with proper validation and execution of smart contracts in real-world scenarios.

2.3 Virtualized Unbiasing Framework (VUF) – Bias Detector Toolkit

2.3.1 Goals and Objectives

Virtualised Unbiasing Framework (VUF) encompasses the design, implementation, and integration of different key project's AI components, such as the XAI library, Bias Detector Toolkit, Adaptive analytics framework and Policy-Oriented Analytics and AI algorithms that are developed under the scoped of WP4 – "Trustworthy and Unbiased AI". However, the main technological outcome of this framework and task is represented by the Bias Detector Toolkit component. It is an information synthesis of methods to mitigate bias, displayed as an interactive catalogue - Bias Detector Toolkit. The catalogue will be augmented with an explanatory introduction for a thorough understanding of bias and with examples to raise awareness of bias mitigation.

In that direction, the key objectives of the Bias Detector Toolkit can be summarised as below:

- raising awareness of the importance of bias mitigation.
- information synthesis of debiasing processes.
- adapting the outputs of the component to the use case scenarios.

2.3.2 State of the Art

The increasing integration of AI systems into critical decision-making processes has brought the challenge of bias (e.g., bias in data, algorithmic bias) to the forefront, leading to the development of various frameworks and tools for its mitigation (Ferrara, 2023). In that context, recent advancements in bias mitigation focus on pre-processing (Duong, M. K., & Conrad, S., 2023), in-processing (Enock, F. E., Hewstone, M. R., Lockwood, P. L., & Sui, J., 2020), and post-processing strategies (Petersen, F., Mukherjee, D., Sun, Y., & Yurochkin, M., 2021). Such examples include reweighting samples or generating synthetic data to balance representation (Guo, D., Li, Z., Zhao, H., Zhou, M., & Zha, H., 2022), as well as the outputs of trained models are adjusted to meet fairness criteria without altering the underlying model (Xu, H., Liu, X., Li, Y., Jain, A., & Tang, J., 2024). Despite these significant advancements, challenges still persist and are related to the different variations of bias across applications and demographics (Lewicki, K., Lee, M. S. A., Cobbe, J., & Singh, J., 2023), the (Buijsman, 2024), and finally the low scalability and usability that describe these models (González-Sendino, 2024).

The Bias Detector Toolkit aims to address these challenges by integrating adaptability, performance balancing, and user-focused design. In that context, this component is designed to provide information, visualisations and tools for identification and mitigation of bias in AI models. It incorporates state-of-the-art techniques, algorithms, and data sources to provide accurate and reliable bias assessment and remediation capabilities. Going beyond a traditional catalogue, the Bias Detector Toolkit is going to have a powerful visual component, structuring the mitigation tools in an intuitive way, more approachable to both developers and policy making stakeholders.

2.3.3 Background technology

Key baseline technologies and platforms that will be utilised for developing the Bias Detector Toolkit are listed in Table 35.

Table 35: T4.1 - Background Technology

Technology Name	Technology Description	Advancements / Usage
IRCAI Top 100 projects platform	In-house developed web-based infrastructure for submitting the projects and for reviewers to evaluate the submitted projects.	IRCAI Top 100 projects platform
SDG observatories platform	In-house developed web based exploratory data visualization tools.	SDG observatories platform

2.3.4 Component to User Requirements

Below follow four (4) tables, i.e., Table 36 – 39, that map respective User Requirements with the Bias Detector Toolkit that seeks to be implemented in the context of the VUF framework of the AI4Gov project.

Table 36: T4.1 - 1st Component Functionality

Title	Description
ID	U-REQ-T4.1-01
Source User	Ensuring Ethical Excellence by Reviewing AI Projects for Inclusion

Requirement	in the IRCAI Top 100 Program
Use case quote	**P2UC1** - As an applicant to the IRCAI Top 100 projects, I need to understand which ethical and bias consideration I should think of when building my models, so that my created models are fair. **P2UC1** - As a reviewer of the IRCAI Top 100 program, I need to evaluate ethical considerations and bias in the submitted projects, so that we identify the AI projects that could be showcased for others (in terms of ethics).
Generic/Specific	Generic & Specific
Task/Component	T4.1 - Virtualized Unbiasing Framework (VUF) for AI & Big Data / Bias Detector Toolkit
Lead partner	JSI
Notes	N/A

Table 37: T4.1 - 2nd Component Functionality

Title	**Description**
ID	U-REQ-T4.1-02
Source User Requirement	Uncovering and Rectifying Data Biases for Comprehensive Sustainability Insights based on the SDG observatory
Use case quote	**P2UC2** - The development of the SDG observatory aims to address the problem of possible bias in data and in general as much as possible. This UC will develop a tool that will detect and eliminate biases to make the data as unbiased as possible, while identifying the topics and themes that are more prone to bias. Bias could also be in not showing some data, that are otherwise relevant, so this UC is interested in the lack of data as well (for instance, because it is not available).
Generic/Specific	Generic & Specific
Task/Component	T4.1 - Virtualized Unbiasing Framework (VUF) for AI & Big Data / Bias Detector Toolkit
Lead partner	JSI

Notes	N/A

Table 38: T4.1 - 3rd Component Functionality

Title	**Description**
ID	U-REQ-T4.1-03
Source User Requirement	Unbiased analytics on a municipality region
Use case quote	**P3UC1, P3UC2** Tourism-driven multi-domain policy management and optimization (VVV/MT), Use case #3.
Generic/Specific	Specific
Task/Component	T4.1 - Virtualized Unbiasing Framework (VUF) for AI & Big Data / Bias Detector Toolkit
Lead partner	JSI
Notes	A checklist for detecting bias in the policy use cases of the pilot.

Table 39: T4.1 - 4th Component Functionality

Title	**Description**
ID	U-REQ-T4.1-04
Source User Requirement	Unfolding the potentials of creating unbiased policies on a water cycle management scenario
Use case quote	**P1UC1**, **P1UC2** Policies for Sustainable Water Cycle Management at a Large Scale, Use case #1.
Generic/Specific	Specific
Task/Component	T4.1 - Virtualized Unbiasing Framework (VUF) for AI & Big Data / Bias Detector Toolkit
Lead partner	JSI
Notes	A checklist for detecting bias in the policy use cases of the pilot.

2.3.5 Component to Technical Requirements

In Tables 40 – 43, the technical requirements that will be covered through the design and implementation of the **VUF and Bias Detector Toolkit** are listed. These requirements are mapped to specific component functionalities and further address user requirements as identified in the context of pilot scenarios.

Table 40: T4.1 - 1st Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.1-1
Type	ENV
Short name	Pilots' requirements for VUF with Bias Detector Toolkit
Functionality ID	U-REQ-T4.1-01, U-REQ-T4.1-02
Description & quantification	Input from pilots.
Additional information (source, characteristics, short description of the process)	Sustainable Development and the European Green Deal (JSI), UC#1 and UC#2. Outputs will also be tailored for VVV/MT and DPB pilots.
Priority	OPT
Reference Scenarios	Primarily P2UC1, P2UC2, P2UC3, with possible enhancement on VVV/MT and DPB pilot scenarios.
Success criteria	Improved methodology for bias mitigation in pilots.

Table 41: T4.1 - 2nd Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.1-2
Type	FUN
Short name	Authentication for use cases
Functionality ID	U-REQ-T4.1-01, U-REQ-T4.1-02

Description & quantification	A system that will enable secure authentication for accessing the catalogue's part specially tailored for use cases.
Additional information (source, characteristics, short description of the process)	P2UC1, P2UC2. Outputs will be also tailored for VVV/MT and DPB pilot scenarios.
Priority	OPT
Reference Scenarios	P1UC1, P1UC2, P2UC1, P2UC2, P3UC1, P3UC2
Success criteria	Pilots can access their specific part of Bias Detector catalogue.

Table 42: T4.1 - 3rd Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.1-3
Type	FUN
Short name	Connection with APIs
Functionality ID	U-REQ-T4.1-01, U-REQ-T4.1-02, U-REQ-T4.1-03, U-REQ-T4.1-04
Description & quantification	Integration with external APIs
Additional information (source, characteristics, short description of the process)	
Priority	OPT
Reference Scenarios	All
Success criteria	API connection successfully implemented

Table 43: T4.1 - 4th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.1-4
Type	FUN
Short name	Database / input system for injection of resources into Bias Detector Catalogue webpage
Functionality ID	U-REQ-T4.1-01, U-REQ-T4.1-02, U-REQ-T4.1-03, U-REQ-T4.1-04
Description & quantification	A system that enables (semi)automatic injection of resources in the Bias Detector Catalogue
Additional information (source, characteristics, short description of the process)	
Priority	OPT
Reference Scenarios	All
Success criteria	Resource injection is (semi)automatic.

2.4 SAX/XAI Library (IBM)

2.4.1 Goals and Objectives

A crucial element in business process investigation and optimisation is the explainability of process outcomes and in-process decisions. The ultimate goal of process explainability is to produce sound (valid and true), context-aware (considering constraints of the business process execution and the external circumstances affecting this execution), and human-interpretable explanations. Such explanations should take into account the broad situations affecting the process execution and flow, the human preferences, the model performance and the explanation quality metrics: the causal execution relationships between process activities, which might differ from time-precedence relationships found in traditional process mining techniques, feature importance models which take into account the aforementioned true causal execution dependencies, while also accommodating user-preferences about these features to balance between accuracy and quality of the explanations as may be determined by various metrics.

Therefore, the goal of SAX/XAI library is to present a new toolset supporting new methodologies and techniques for process explainability, enabling generation of sound (valid and true), human-interpretable, situation-aware explanations for process-execution decisions and outcomes.

2.4.2 State of the Art

Business process management systems aim to monitor, adapt, and improve business processes with respect to various performance indicators. In the current state of the art, those systems are AI-empowered to learn, reason and act upon business process data within a set of process constraints and assumptions (Dumas, 2023). One of the main characteristics of such systems is the ability to reason and explain their decisions. Usually AI-powered systems employ ML techniques to analyse process data, predict the desired process behaviours in various circumstances and drive the execution of those behaviours. XAI techniques are then applied to the ML models to reason and explain the prediction outcomes. However, state of the art frameworks for explainability often fail to include the richness of contextual situations that affect or cause process outcomes (Amit, 2022), to correctly deduce the causal execution relationships between different process activities (Fournier, 2023) and to take those relationships into account when employing the XAI techniques on a different set of process attributes to determine which of those affected most the decisions. They also fail to consider not only technical facets but also to embed user-focused aspects of explainability.

2.4.3 Background technology

Key baseline technologies and platforms that are utilised for the developing the SAX library are listed in Table 44. The technologies are implemented in the SAX4BPM open source library accessible at: https://github.com/IBM/sax4bpm.

Table 44: T4.2 - Background Technology

Technology Name	Technology Description	Advancements / Usage
PM4PY	Open-source process mining platform written in Python providing process discovery by using process event logs as an input.	Process mining will be utilised to discover business process models, serving as basis for causal execution discovery and XAI on process data.
DirectLINGAM	A python open-source implementation of the LINGAM algorithm for non-gaussian acyclic causal discovery.	DirectLINGAM algorithm will be used to support discovery of temporal causal execution dependencies among process activities.
XAI techniques algorithms	A set of XAI processes and methods allowing model-agnostic and model-specific methods for model	XAI techniques will be used for process-specific explainability of ML prediction and decision-making

	explainability. Such models and techniques include SHAP, LIME algorithms, or more model-specific interpretability methods such as graphic tree or deep-learning explainability methods. We will rely on open-source python implementation of those algorithms which can be found in several open-source packages.	models for business process analysis.
LLMs and frameworks for development of LLM-driven applications	Open-source frameworks such as Langchain and/or Microsoft Guidance as well as open source LLM models such as HuggingFace and potential integration with IBM WatsonX.	Will be used for creation of human-understandable synthesis of various explainability views into business processes, including process models, causal dependency models and XAI.

2.4.4 Component to User Requirements

Table 45 and Table 46, map the respective User Requirements with the ***SAX4BPM*** services that seeks to be implemented in the context of the SAX/XAI Library component of the AI4Gov project.

Table 45: T4.2 - 1st Component Functionality

Title	**Description**
ID	U-REQ-T4.2-01
Source User Requirement	Enable true and sound explanations for process outcomes (e.g., bin levels) and in-process decisions (e.g., bin pickups) for process optimization and improvement.
Use case quote	**P2UC2** – Waste monitoring and optimization of allocation of stuff and resources – As a member of the Municipal Council, I want to know which are the optimum areas, so that I can allocate necessary funds and resources. Addresses goals of the use case: • A reliable and technologically advanced Management Information System. • A tool to monitor overall data from Telematics, recommend optimum areas and predict financial results.

	Optimization of garbage bins pickup scheduling and reduction of costs by creating a prediction model for optimized pickups scheduling considering various external factors (weather, location, local conditions, events).
Generic/Specific	Specific
Task/Component	T4.2 – SAX4BPM enabler services
Lead partner	IBM
Notes	None

Table 46: T4.2 - 2nd Component Functionality

Title	**Description**
ID	U-REQ-T4.2-02
Source User Requirement	Streamline and generate human-interpretable explanations for business process outcomes and in-process decisions considering user-preferences and quantitative metrics for explanation quality.
Use case quote	**P2UC2** – Waste monitoring and optimization of allocation of stuff and resources - As a member of the Municipal Council, I want to know which are the optimum areas, so that I can allocate necessary funds and resources. Addresses goals of the use case: • A reliable and technologically advanced Management Information System. • A tool to monitor overall data from Telematics, recommend optimum areas and predict financial results. • A tool to raise awareness of Citizens and businesses. Provide high-level human-interpretable context-aware explanations on garbage pickups optimization and prediction models to encourage trust and bring clarity to process outcomes and decisions.
Generic/Specific	Specific
Task/Component	T4.2 – SAX4BPM realisation services

Lead partner	IBM
Note	None

2.4.5 Component to Technical Requirements

Tables 47 - 49, illustrate the technical requirements that will be covered through the design and implementation of the ***SAX4BPM*** Services. These requirements are mapped to specific component functionalities and address user requirements, as identified in the context of pilot scenarios.

Table 47: T4.2 - 1st Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.2-01
Type	FUNC
Short name	Process and causal execution dependency analyser
Functionality ID	U-REQ-T4.2-01
Description & quantification	Process and causal execution dependency discovery module based on event log data to create true representation of process execution dependency to serve as basis for process decision and outcome explainability.
Additional information (source, characteristics, short description of the process)	Process event log data required to be fed into this component
Priority	MAN
Reference Scenarios	P2UC2
Success criteria	Successful discovery of process dependencies

Table 48: T4.2 - 2nd Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.2-02
Type	FUNC
Short name	Process-aware XAI
Functionality ID	U-REQ-T4.3-01, U-REQ-T4.3-02
Description & quantification	XAI module based on process temporal execution dependency module for process-aware explanations, and driven by user and metric-informed techniques providing quality-driven explanations.
Additional information (source, characteristics, short description of the process)	Process event log data, process and causal dependency models, contextual information, user-preferences
Priority	MAN
Reference Scenarios	P2UC2
Success criteria	Global and local process-aware feature importance designation for specific process decision and outcomes

Table 49: T4.2 - 3rd Technical Requirement

Section	**Description**
Technical requirement code	T-REQT4.2-03
Type	FUNC
Short name	Human-interpretable explanation synthesis.
Functionality ID	U-REQ-T4.2-02
Description & quantification	XAI user-and metric-informed techniques providing trade-off between explanation quality and performance

	(e.g., accuracy) and synthesizing all process perspectives into single human-interpretable view.
Additional information (source, characteristics, short description of the process)	XAI feature analysis, user-preferences, process-oriented user queries
Priority	MAN
Reference Scenarios	P2UC2
Success criteria	Human interpretable high-quality (based on qualitative and quantitative metrics) explanations for process outcomes and decisions

2.5 Policy-Oriented AI and NLP algorithms

2.5.1 Goals and Objectives

"Policy-Oriented AI and NLP algorithms" is being developed in the context of T4.3 – "Improve Citizen Engagement and Trust utilising NLP" and consists of two (2) sub-components, namely *Policy-Oriented Analytics and AI Algorithms* and *Adaptive Analytics Framework*. The *Policy-Oriented Analytics and AI Algorithms* aims to develop several NLP algorithms in order to analyse large volumes of text data and also assist the respective AI experts. This subcomponent consists of the following mechanisms:

- Question Answering Service
- Time Series Analyser

Additional to these two components, a Multilingual Bias Classification is implemented which is further enforced by the integration of a Retrieval-Augmented Generation (RAG) mechanism powered by the utilization of Large Language Models (LLMs). The latter is designed under the Adaptive Analytics Framework, however its main difference with the other components is that it is not exposed via an API through this framework, rather it directly integrates with the Visualization Workbench.

The scope of the Adaptive Analytics Framework subcomponent is to develop the needed ML models for performing predictive analytics and optimised resource allocation to satisfy the needs of the pilots and assist policy makers.

All the above should be executed in an efficient manner, utilising the least possible number of resources.

2.5.2 State of the Art

There have been several advancements in the fields of policy-oriented AI and NLP. To begin with, AI has been widely utilised in a variety of sectors to assist the corresponding policy makers. For instance, in the healthcare domain and during the recent COVID-19 pandemic a variety of approaches (Rahman, 2021) were introduced in order to help public health experts form specific policies, protocols and interventions that could potentially protect the citizens from the pandemic. In those approaches a variety of data analytics took place and a vast number of ML algorithms were developed (Mavrogiorgou, 2021) to monitor the progress of the pandemic and predict certain outcomes, thus enabling policy makers to make the right decisions. Towards the same direction, i.e., assisting in data-driven decision making and thus, creating more useful policies, AI is also being utilised in a variety of sectors such as e-Governance (Alexopoulos, 2019), transportation (Ağbulut, 2022) and environment (Hettinga, 2023). Regarding AI4Gov's pilots and the corresponding sectors, which are water management, sustainability, and tourism, there have also been some advancements in the literature. More specifically, regarding water management the authors in (Gino Sophia, 2020) proposed a genetic algorithm based on a fitness function that effectively manages water distribution with regression of 98%. In (Bhardwaj, 2022), the authors proposed a machine learning-based framework for the assessment of water quality by utilising a variety of ML-based algorithms such as logistic regression, naïve Bayes, ensemble-based approaches such as Random Forest and XGBoost to classify the data in appropriate classes and predict turbidity in a water sample. As for sustainability, a variety of approaches have been proposed as the ones presented in (Shafiq, 2020) and aim to the classification of Sustainable Smart Cities (SSC) network traffic. Those approaches are based on several ML techniques; most of them are more traditional while very few approaches utilise more advanced techniques such as reinforcement learning, ensemble learning and genetic algorithms. Regarding tourism, several approaches have been developed that aim to address different use case scenarios, mostly for predicting tourism flows and performing sentiment analysis such as the ones presented in (Xie, 2021), (Li, 2021) and (Puh, 2023).

As for NLP, there have been tremendous advancements, since the high availability of resources and data have provided new opportunities including Large Language Models (LLMs) that are capable of performing a wide variety of tasks in different domains. An example of such an LLM is ChatGPT (Geertsema, 2023) which is capable of a range of tasks, from providing information about a historical event to even debugging source code and/or generating new one. The reason for those capabilities is the fact that ChatGPT has been trained on an enormous number of data available on the internet. However, the aforementioned data are neither clean nor unbiased (Ray, 2023). As a result, the provided answers from such a tool are sometimes biased and/or wrong. In general terms, the performance of every ML model (either NLP related or not) that is created is highly correlated with the data that it has been trained on. This means that if the quality of the data is pure, then the results provided by a model and the model itself will be pure. To this end, the Policy-Oriented AI and NLP algorithms seek to implement appropriate AI techniques and NLP algorithms that satisfy the needs of the AI4Gov pilots whilst taking into consideration any potential challenges and ensuring the development of unbiased ML models.

2.5.3 Background technology

Key baseline technologies that will be utilised for developing Policy-Oriented AI and NLP algorithms are listed in Table 50.

Table 50: T4.3 - Background Technology

Technology Name	Technology Description	Advancements / Usage
TensorFlow	TensorFlow (TensorFlow, n.d.) is an open-source software library for machine learning and artificial intelligence.	TensorFlow will be utilised to perform data analysis on the available pilot data (streaming and non-streaming) and train specific ML algorithms for predictive analytics and resource allocation.
Natural Language Toolkit	The Natural Language Toolkit (Natural Language Toolkit, n.d.), or more commonly NLTK, is a suite of libraries for symbolic and statistical natural language processing.	NLTK will be utilised for processing text data and, along with PyTorch, will assist in the development of advanced NLP algorithms and models.
PyTorch	PyTorch (PyTorch, n.d.) is a machine learning framework based on the Torch library, used for applications such as computer vision and natural language processing.	Pytorch will be used to develop specific NLP algorithms that will meet the specifications of the AI4Gov pilots.
HuggingFace Transformers	An open-source library that provides state-of-the-art implementations of transformer models, such as BERT, GPT, and T5. It supports tasks like natural language processing (NLP), text classification, sentiment analysis, translation, and more.	HuggingFace has advanced NLP with pre-trained models and fine-tuning capabilities, enabling faster development of AI solutions. It's widely used in industry and research for chatbots, sentiment analysis, summarization, and multi-modal AI tasks.
Langchain	A framework designed for building applications with language models by integrating them with various data sources, tools, and memory capabilities. LangChain emphasizes chaining	Used for creating dynamic and adaptive LLM-based applications, including chatbots, question-answering systems, document analysis, and custom generative AI workflows. Its ability to combine LLMs with external tools like databases, APIs,

	language models together for complex workflows.	and vector stores has enabled more intelligent and context-aware applications.

2.5.4 Component to User Requirements

2.5.4.1 Question Answering Service

Table 51 maps respective User Requirements with the ***Question Answering Service*** that seeks to be implemented in the context of the Policy-Oriented AI and NLP algorithms component of the AI4Gov project.

Table 51: T4.3 - 1st Component Functionality

Title	**Description**
ID	U-REQ-T4.3-01
Source User Requirement	Analytical tool will raise awareness among developers of AI solutions
Use case quote	**P2UC4** OECD policy documents analysis US#1 - As an expert for AI ethics, I would like to see which national policy documents are examining AI ethics and are considering bias in AI, so that I can identify the current regulations and good practices.
Generic/Specific	Specific
Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Policy-Oriented Analytics and AI Algorithms Component
Lead partner	UPRC
Notes	None

2.5.4.2 *Time Series Analyser*

Tables 52 - 54, illustrate respective User Requirements with the ***Time Series Analyser*** Service that seeks to be implemented in the context of the Policy-Oriented AI and NLP algorithms component of the AI4Gov project.

Table 52: T4.3 - 2nd Component Functionality

Title	**Description**
ID	U-REQ-T4.3-02
Source User Requirement	Analysing the data over longer periods of time, the pilot would help policymakers identify recurring problems and overall trends
Use case quote	**P1UC1** Water management cycle – drinking water US#2 - As a member of the local administration, I want to have a clear picture of the main areas for improvement within the drinking water treatment system, so that I have relevant information available when making decisions on infrastructure improvement and long-term strategies.
Generic/Specific	Generic
Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Policy-Oriented Analytics and AI Algorithms Component
Lead partner	UPRC
Notes	None

Table 53: T4.3 - 3rd Component Functionality

Title	**Description**
ID	U-REQ-T4.3-03
Source User Requirement	Analysing the data over longer periods of time, the pilot would help policymakers identify recurring problems and overall trends
Use case quote	**P1UC2** Water management cycle – Sewage water US#2 - As a member of the local administration, I want to have a clear picture of the main areas for improvement within the sewage water treatment system, so that I have relevant information available

	when making decisions on infrastructure improvement and long-term strategies.
Generic/Specific	Generic
Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Policy-Oriented Analytics and AI Algorithms Component
Lead partner	UPRC
Notes	None

Table 54: T4.3 - 4th Component Functionality

Title	**Description**
ID	U-REQ-T4.3-04
Source User Requirement	Tool that monitors the number of parking tickets issued and analyses their time and spatial evolution in order to allocate municipal police staff, vehicles and equipment in an optimum way
Use case quote	**P3UC1** Parking Tickets Monitoring US#1 - As a municipal officer in charge of the Municipal police, I want to know which are the areas, days, and hours that violation of parking rules is observed, so that I allocate municipal police staff and equipment in an optimum way. **P3UC1** Parking Tickets Monitoring US#2 - As a member of the Municipal Council, I want to know which are the areas, days, and hours that violation of parking rules is observed, so that I allocate necessary funds and resources to address the problem.
Generic/Specific	Generic
Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Policy-Oriented Analytics and AI Algorithms Component
Lead partner	UPRC
Notes	None

2.5.4.3 Multilingual Bias Classification enhanced with RAG

Table 55 and Table 56 describe respective User Requirements with the Multilingual Bias Classification that seeks to be implemented in the context of the Policy-Oriented AI and NLP algorithms component of the AI4Gov project.

Table 55: T4.3 - 5th Component Functionality

Title	Description
ID	U-REQ-T4.3-05
Source User Requirement	Multilingual Bias Detection Pipeline
Use case quote	**P2UC4** OECD Policy Documents Analysis US#1 - As a content moderator, I want to detect and classify biases in multilingual articles to ensure balanced and unbiased reporting.
Generic/Specific	Generic
Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Policy-Oriented Analytics and AI Algorithms Component
Lead partner	UPRC
Notes	None

Table 56: T4.3 - 6th Component Functionality

Title	Description
ID	U-REQ-T4.3-06
Source User Requirement	Enhanced Bias Classification with Contextual Data
Use case quote	**P2UC4** OECD Policy Documents Analysis US#1 - As a policymaker, I want a system that can retrieve and integrate external contextual data to classify and explain biases in public policies for transparent decision-making.
Generic/Specific	Generic

Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Policy-Oriented Analytics and AI Algorithms Component
Lead partner	UPRC
Notes	None

2.5.4.4 Adaptive Analytics Framework

Table 57, Table 58 and Table 59, describe respective User Requirements with the ***Adaptive Analytics Framework*** that seeks to be implemented in the context of the Policy-Oriented AI and NLP algorithms component of the AI4Gov project.

Table 57: T4.3 - 7th Component Functionality

Title	**Description**
ID	U-REQ-T4.3-07
Source User Requirement	The pilot would provide a tool for technicians to detect possible sources of inefficiency within the system
Use case quote	**P1UC2** Water management cycle – drinking water US#2 - Analysis and cross-comparison of real-time operational data, detect possible sources of inefficiency within the system
Generic/Specific	Generic
Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Adaptive Analytics Framework Component
Lead partner	UPRC
Notes	None

Table 58: T4.3 - 8th Component Functionality

Title	Description
ID	U-REQ-T4.3-08
Source User Requirement	The pilot would provide a tool for technicians to detect possible sources of inefficiency within the system
Use case quote	**P1UC2** Water management cycle – Sewage water US#2 - analysis and cross-comparison of real-time operational data, detect possible sources of inefficiency within the system
Generic/Specific	Generic
Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Adaptive Analytics Framework Component
Lead partner	UPRC
Notes	None

Table 59: T4.3 - 9th Component Functionality

Title	Description
ID	U-REQ-T4.3-09
Source User Requirement	Innovative tool to streamline waste monitoring and optimize the allocation of staff and resources.
Use case quote	**P3UC2** Waste management – Pay As You Throw (PAYT) US#1 – As a municipal officer in charge of Waste Management and Recycling Unit, I want to know which are the areas where the Pay As You Throw System will be expanded so that I facilitate the Implementation of Pay As You Throw System. **P3UC2** Waste management – Pay As You Throw (PAYT) US#2 - As a member of the Municipal Council, I want to know which are the optimum areas, so that I can predict the financial outcomes.
Generic/Specific	Generic

Task/Component	T4.3 Improve Citizen Engagement and Trust utilising NLP / Adaptive Analytics Framework Component
Lead partner	UPRC
Notes	None

2.5.5 Component to Technical Requirements

2.5.5.1 Question Answering Service

Tables 60 – 65 list the technical requirements covered through the design and implementation of the ***Question Answering Service***. These requirements are mapped to specific component functionalities and address user requirements, as identified in the context of pilot scenarios.

Table 60: T4.3 - 1st Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.3-01
Type	FUNC
Short name	Text data feed
Functionality ID	U-REQ-T4.3-1
Description & quantification	Text data need to be fed into this component.
Additional information (source, characteristics, short description of the process)	Data source and data schema should be described.
Priority	MAN
Reference Scenarios	P2UC1, P2UC2, P2UC3
Success criteria	Successful ingest of text data in this component

Table 61: T4.3 - 2nd Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-02
Type	FUNC
Short name	APIs for communication with the AI4Gov frontend
Functionality ID	U-REQ-T4.3-1
Description & quantification	APIs for internal use should be developed.
Additional information (source, characteristics, short description of the process)	Specific APIs should be developed in order to allow the communication of this component with the frontend of the AI4Gov platform.
Priority	MAN
Reference Scenarios	All
Success criteria	Successful communication between this component and the Visualization Workbench.

Table 62: T4.3 - 3rd Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-03
Type	FUNC
Short name	Storage of documents.
Functionality ID	U-REQ-T4.3-1
Description & quantification	The OECD documents that this subcomponent uses, should be stored.
Additional information (source, characteristics, short description of the process)	The OECD documents, as well as any documents that originate from them after proper processing, should be stored.

Priority	MAN
Reference Scenarios	P2UC1, P2UC2, P2UC4
Success criteria	Successful storage of OECD Documents.

Table 63: T4.3 - 4th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-04
Type	FUNC
Short name	Text translation
Functionality ID	U-REQ-T4.3-1
Description & quantification	The text of the OECD documents should be translated to English.
Additional information (source, characteristics, short description of the process)	The OECD documents are multilingual and, as a result, they should be translated to English so that the question answering can take place.
Priority	MAN
Reference Scenarios	P2UC1, P2UC2, P2UC4
Success criteria	Successful translation of OECD Documents.

Table 64: T4.3 - 5th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-05
Type	PERF
Short name	Provided answers accuracy

Functionality ID	U-REQ-T4.3-1
Description & quantification	The accuracy of the provided answers should be sufficient.
Additional information (source, characteristics, short description of the process)	The accuracy, as well as the consistency of the provided answers should be sufficient.
Priority	MAN
Reference Scenarios	P2UC1, P2UC2, P2UC4
Success criteria	The accuracy of the used algorithms should be above 80% and the answers provided should be consistent.

Table 65: T4.3 - 6th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.3-06
Type	PERF
Short name	Time needed for providing answers
Functionality ID	U-REQ-T4.3-1
Description & quantification	This subcomponent should provide answers in real time, as fast as possible.
Additional information (source, characteristics, short description of the process)	As long as a user has asked a question, the whole process should last some seconds.
Priority	DES
Reference Scenarios	P2UC1, P2UC2, P2UC4
Success criteria	The whole question answering process should not exceed the limit of ten (10) seconds.

2.5.5.2 Time Series Analyser

Table 66 – 72 list the technical requirements that will be covered through the design and implementation of the ***Time Series Analyser***. These requirements are mapped to specific component functionalities and address user requirements, as identified in the context of pilot scenarios.

Table 66: T4.3 - 7th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-07
Type	FUN
Short name	Time series data provided
Functionality ID	U-REQ-T4.3-2, U-REQ-T4.3-3, U-REQ-T4.3-4
Description & quantification	Time series data should be provided to this subcomponent.
Additional information (source, characteristics, short description of the process)	The data that will be provided to this subcomponent should be time series, meaning that they should include specific variables such as date/timestamps.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful provision of data in terms on schema and data consistency

Table 67: T4.3 - 8th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-08
Type	FUN
Short name	Connection with external APIs
Functionality ID	U-REQ-T4.3-2, U-REQ-T4.3-3, U-REQ-T4.3-4

Description & quantification	Connection with external APIs in order to retrieve the time series data.
Additional information (source, characteristics, short description of the process)	The time series data are stored in external platforms, so dedicated APIs should be developed, so that they can be retrieved by this subcomponent.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful retrieval of time series data.

Table 68: T4.3 - 9th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.3-09
Type	FUNC
Short name	Historical data analysis
Functionality ID	U-REQ-T4.3-2, U-REQ-T4.3-3, U-REQ-T4.3-4
Description & quantification	Historical data should be analysed.
Additional information (source, characteristics, short description of the process)	The subcomponent should be able to analyse historical data that will be provided by the pilots in order to provide useful insights.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful analysis of data.

Table 69: T4.3 - 10th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-10
Type	FUNC
Short name	Descriptive data analysis
Functionality ID	U-REQ-T4.3-2, U-REQ-T4.3-3, U-REQ-T4.3-4
Description & quantification	Descriptive data analysis should data place.
Additional information (source, characteristics, short description of the process)	This component should perform descriptive analysis on data in order to discover possible trends, erroneous values, thus guiding their further analysis.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful analysis of data.

Table 70: T4.3 - 11th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-11
Type	FUNC
Short name	Data cleaning
Functionality ID	U-REQ-T4.3-2, U-REQ-T4.3-3, U-REQ-T4.3-4
Description & quantification	Data should be cleaned
Additional information (source, characteristics, short description of the process)	The provided data should be cleaned so that they are not erroneous, since this will affect the performance of the ML models that we be developed based on those data.
Priority	MAN

Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful cleaning of data and removal of erroneous values.

Table 71: T4.3 - 12th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-12
Type	FUN
Short name	Visualisation of time series analysis results.
Functionality ID	U-REQ-T4.3-2, U-REQ-T4.3-3, U-REQ-T4.3-4
Description & quantification	The results of the time series analysis should be visualised and available to the users.
Additional information (source, characteristics, short description of the process)	Dedicated API that will allow the communication of the Time Series Analyser with the Visualization Workbench should be developed, so that the results can be visualised.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful communication between this component and the Visualization Workbench.

Table 72: T4.3 - 13th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-13
Type	PERF

Short name	Performance of Time Series Analyser
Functionality ID	U-REQ-T4.3-2, U-REQ-T4.3-3, U-REQ-T4.3-4
Description & quantification	The accuracy of the used models should be sufficient.
Additional information (source, characteristics, short description of the process)	The models used for performing time series analysis should have sufficient performance and provide accurate results.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	The accuracy of the used models should be above 90%.

2.5.5.3 Multilingual Bias Classification

Tables 73 – 75 list the technical requirements that will be covered through the design and implementation of the ***Multilingual Bias Classification***. These requirements are mapped to specific component functionalities and address user requirements, as identified in the context of pilot scenarios.

Table 73: T4.3 – 14th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-14
Type	FUNC
Short name	Multilingual Bias Classification
Functionality ID	U-REQ-T4.3-05, U-REQ-T4.3-06
Description & quantification	Classify biases in multilingual content using pre-trained language models, supporting at least 10 languages, and achieving >85% F1-score for bias detection.
Additional information	Leverages transformer-based architectures like XLM-RoBERTa for multilingual support.
Priority	MAN
Reference Scenarios	Analysis of global news articles and social media content

	for biased narratives.
Success criteria	Multilingual bias classification system successfully deployed with consistent performance across supported languages.

Table 74: T4.3 – 15th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.3-15
Type	FUNC
Short name	RAG Integration for Bias Classification
Functionality ID	U-REQ-T4.3-05, U-REQ-T4.3-06
Description & quantification	Retrieve and integrate relevant external knowledge for bias classification, supporting at least five major data sources (e.g., Wikipedia, PubMed, or trusted news outlets).
Additional information	Focuses on improving contextual understanding in bias detection tasks through retrieved information.
Priority	OPT
Reference Scenarios	Context-aware classification of biases in political speeches and educational material.
Success criteria	Demonstrated improvement in contextual classification metrics by >10%.

Table 75: T4.3 - 16th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.3-16
Type	NON-FUNC

Short name	Explainable Bias Mitigation
Functionality ID	U-REQ-T4.3-05, U-REQ-T4.3-06
Description & quantification	Provide explainability for bias classification decisions, including highlighting key contributing factors in the text. Mitigate identified biases via suggested edits or alternative phrasing.
Additional information	Incorporates SHAP or LIME for explainability and AI-assisted text rewriting mechanisms for mitigation.
Priority	OPT
Reference Scenarios	Content moderation workflows and editorial guidelines for unbiased text creation.
Success criteria	Demonstrated explainability with >80% user satisfaction and actionable bias mitigation suggestions.

2.5.5.4 Adaptive Analytics Framework

Tables 76 – 80 list the technical requirements covered through the design and implementation of the ***Adaptive Analytics Framework***.. These requirements are mapped to specific component functionalities and address user requirements, as identified in the context of pilot scenarios.

Table 76: T4.3 - 14th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.3-17
Type	FUN
Short name	Appropriate data for predictions are provided
Functionality ID	U-REQ-T4.3-5, U-REQ-T4.3-6, U-REQ-T4.3-7
Description & quantification	Use case related data should be provided to the mechanism.
Additional information (source, characteristics, short description	The data provided should contain the appropriate variables. For example, in order to perform resource allocation in a certain area, the data should include

of the process)	coordinates.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	The appropriate data are provided.

Table 77: T4.3 - 15th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-15
Type	FUN
Short name	Connection with external APIs for streaming data
Functionality ID	U-REQ-T4.3-5, U-REQ-T4.3-6, U-REQ-T4.3-7
Description & quantification	Connection with external APIs in order to retrieve streaming data.
Additional information (source, characteristics, short description of the process)	The are generated by sensors, so they should be retrieved by the component.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful retrieval of streaming data.

Table 78: T4.3 - 16th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-16
Type	FUNC

Short name	Predictive analysis
Functionality ID	U-REQ-T4.3-5, U-REQ-T4.3-6, U-REQ-T4.3-7
Description & quantification	Predictive analysis should take place.
Additional information (source, characteristics, short description of the process)	Algorithms for predictive analytics should be developed in order to provide predictions based on the given data.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful development of predictive ML models.

Table 79: T4.3 - 17th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.3-17
Type	FUN
Short name	Visualisation of the Adaptive Analytics Framework's results.
Functionality ID	U-REQ-T4.3-5, U-REQ-T4.3-6, U-REQ-T4.3-7
Description & quantification	The results should be visualised and available to the users.
Additional information (source, characteristics, short description of the process)	Dedicated API should be provided that will allow the communication of the Adaptive Analytics Framework with the Visualization Workbench be developed, so that the results can be visualised.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	Successful communication between this component and the Visualization Workbench.

Table 80: T4.3 - 18th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.3-18
Type	PERF
Short name	Predictive ML models accuracy
Functionality ID	U-REQ-T4.3-5, U-REQ-T4.3-6, U-REQ-T4.3-7
Description & quantification	The predictive ML models that are developed are accurate.
Additional information (source, characteristics, short description of the process)	The accuracy and the corresponding metrics of the developed ML algorithms should be sufficient and above 90%.
Priority	MAN
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	The predictive ML models' metrics are sufficient.

2.6 Interactive Self-Explained Visualization Workbench

2.6.1 Goals and Objectives

The Interactive Self-Explained Visualization Workbench is a versatile web application built on the Next JS framework, which is based on React JS. It serves as a powerful tool for data visualisation and analysis within the context of AI policy-making and bias assessment. This component offers a user-friendly interface with intuitive design and customisable themes, making it easy for users to explore and interact with various data visualisations, including graphs and charts. It not only empowers users to visualise the results of background analytics processes, but also provides crucial insights into AI model bias and its creation. With robust performance and scalability, it ensures a seamless experience for users.

2.6.2 State of the Art

The field of visualisation applications and technologies has been rapidly evolving, driven by advances in computing power, data analytics, and user interface design. Data visualisation has also evolved significantly, with immersive 3D visualisations and interactive dashboards enabling data-driven decision-making across industries (Garzon-Orjuela, N., Parveen, S., Amin, D., Vornhagen, H., Blake, C., & Vellinga, A., 2023). Machine learning and AI-driven techniques have enhanced visual analytics by automating data exploration and generating meaningful insights (Rachakatla, S. K., Ravichandran, P., & Machireddy, J. R., 2023). Data visualisation tools like Tableau and Power BI are now used by over 100,000 organisations worldwide, highlighting the growing importance of data-driven decision support[1]. These advancements collectively underscore the transformative impact of visualisation technologies in various domains, promising exciting prospects for research and development in the foreseeable future. Regarding the AI4Gov's use case sectors, the following observations were made. In the realm of water management, these applications offer a comprehensive and real-time view of water resources, helping stakeholders make informed decisions about allocation and conservation. By visualising water distribution, usage patterns, and infrastructure, these tools empower authorities to optimize resource utilisation. In the context of sustainability, visualisation applications aid in communicating complex data related to environmental impacts and resource consumption. These tools enable stakeholders to grasp the intricacies of sustainability initiatives, fostering greater engagement and awareness among both businesses and the general public. By translating abstract concepts into visually intuitive representations, such as carbon emissions heat maps or interactive sustainability dashboards, these applications contribute to more informed choices and, ultimately, a more sustainable future. In the tourism sector, these applications enable tourism professionals to analyse visitor flows, anticipate trends, and optimise infrastructure investments, thus enhancing the overall tourism experience and the industry's contribution to local economies.

2.6.3 Background technology

Key baseline technologies that will be utilised for the design and implementation of the Interactive Self-Explained Visualization Workbench are listed in Table 81, below.

Table 81: T4.4 - Background Technology

Technology Name	Technology Description	Advancements / Usage
React.js / Next.js	React.js and Next.js provide a flexible framework for handling data.	Further develop data handling by integrating libraries like Redux for state management or Axios for making API requests to various data sources. Optimize performance by implementing code splitting and lazy loading for components using Next.js.
Data Visualisation	Utilise data visualisation libraries	Customize these libraries to

[1] https://www.mordorintelligence.com/industry-reports/data-visualization-applications-market-future-of-decision-making-industry#:~:text=The%20Data%20Visualization%20Market%20size,period%20(2024%2D2029).

Libraries	like D3.js / Chart.js to render graphs and charts.	handle different data formats and real-time data. Enhance interactivity by adding event handlers and user-friendly tooltips to provide detailed information.
UI Framework	Implement a UI framework to achieve a consistent and aesthetically pleasing design (Material- UI, Material Tailwind).	Customize the theme and styles to match the application's branding.
Real-time Data Updates	Real time data are used by the pilots	For real-time updates, it will integrate a WebSocket technology (e.g., Socket.io) to establish bidirectional communication between the server and client.

2.6.4 Component to User Requirements

Table 82 maps respective User Requirements with the ***Interactive Self-Explained Visualization Workbench*** that seeks to be implemented in the context of the AI4Gov project.

Table 82: T4.3 - 1st Component Functionality

Title	**Description**
ID	U-REQ-T4.4-01
Source User Requirement	Enhancing visual analytics, policy-making decisions and explainability via an interactive visualisation workbench.
Use case quote	**P1UC1** Water management cycle – Drinking water #UC1 – An interface for technicians to detect possible sources of inefficiency within the system and provide a tool for improving long-term investment strategies. **P1UC2** Water management cycle – Sewage water #UC2 – An interface for technicians to detect possible sources of inefficiency within the system and provide a tool for improving long-term investment strategies. **P2UC1** IRCAI global top 100 projects #UC1 – Have an interface to present rules and bias evaluation of any AI solution in general. **P2UC2** SDG Observatory #UC2 – Be able to compare and rank different countries and regions to see their achievements about the SDG of choice,

	and to identify the best performing countries and follow their practices. **P2UC4** OECD policy documents analysis #UC3 – A visual summary will be created presenting solutions and good practices. **P3UC1** Parking tickets monitoring #UC1 – A tool that monitors the number of parking tickets issued and analyses their time and spatial evolution. **P3UC2** Waste management - Pay As You Throw (PAYT) #UC2 – An innovative tool for waste monitoring and optimizing the allocation of staff and resources.
Generic/Specific	Generic
Task/Component	T4.4 - Interactive Self-Explained Visualization Workbench
Lead partner	UPRC
Notes	The project's User interface and frontend web-application.

2.6.5 Component to Technical Requirements

Tables 83 – 96 he technical requirements covered through the design and implementation of the ***Interactive Self-Explained Visualization Workbench* are listed**. These requirements are mapped to specific component functionalities and further address user requirements, as identified in the context of pilot scenarios.

Table 83: T4.4 – 1st Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.4-01
Type	DATA
Short name	Data Sources
Functionality ID	U-REQ-T4.4-01
Description & quantification	The application must support data retrieval from various sources, including AI model outputs, analytics results, external APIs and internal data stores.
Additional information (source, characteristics, short description of the process)	Data source must be described and documented in order to avoid importing errors or failing to meet certain requirements.

Priority	MAN
Reference Scenarios	All
Success criteria	No errors occur in the importing of data.

Table 84: T4.4 – 2nd Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.4-02
Type	DATA
Short name	Data Formats
Functionality ID	U-REQ-T4.4-01
Description & quantification	It should be able to handle different data formats (e.g., JSON, CSV, SQL) for visualisation and analytics.
Additional information (source, characteristics, short description of the process)	Data that need to be visualised may exist in various forms, so the application should be able to work with all major data types.
Priority	MAN
Reference Scenarios	All
Success criteria	No errors occur while using a dataset.

Table 85: T4.4 – 3rd Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.4-03
Type	DATA
Short name	Real-time Updates

Functionality ID	U-REQ-T4.4-01
Description & quantification	Support real-time or near-real-time data updates for dynamic visualisations.
Additional information (source, characteristics, short description of the process)	Some pilot data are sourced in real time, so a mechanism must exist that can manage real time visualisations.
Priority	DES
Reference Scenarios	P1UC1, P1UC2, P3UC1, P3UC2
Success criteria	No errors occur while visualising real time charts.

Table 86: T4.4 – 4th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.4-04
Type	FUNC
Short name	Information regarding bias in AI
Functionality ID	U-REQ-T4.4-01
Description & quantification	The application must provide visual information regarding bias in AI models.
Additional information (source, characteristics, short description of the process)	The interface that describes the importance of avoiding bias in AI models must be easily understandable by all readers.
Priority	MAN
Reference Scenarios	P2UC1, P2UC2, P2UC3
Success criteria	Bias-related data and visualisations are accessible and easy to understand.

Table 87: T4.4 – 5th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.4-05
Type	L&F
Short name	User Interface (UI)
Functionality ID	U-REQ-T4.4-01
Description & quantification	The UI should have an intuitive and modern design to ensure a pleasant user experience.
Additional information (source, characteristics, short description of the process)	Design an intuitive and user-friendly interface for easy navigation and interaction.
Priority	MAN
Reference Scenarios	All
Success criteria	All interfaces are well designed, and no flaws occur.

Table 88: T4.4 - 6th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.4-06
Type	L&F
Short name	Customization
Functionality ID	U-REQ-T4.4-01
Description & quantification	Allow users to customize charts and graphs to meet their preferences.
Additional information (source, characteristics, short description of the process)	Make use of interactive charts that allow the use of "what-if" scenarios for applicable use cases. Enable users to interact with the visualisations, such as zooming, panning, and selecting data points.

Priority	DES
Reference Scenarios	All
Success criteria	Graphs and charts allow users to further customize the result.

Table 89: T4.4 - 7th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.4-07
Type	USE
Short name	Interactivity
Functionality ID	U-REQ-T4.4-01
Description & quantification	Provide interactive features for users to AI bias insights effectively.
Additional information (source, characteristics, short description of the process)	Use the method of scrolly telling to describe a "story" of bias reduction in AI models.
Priority	MAN
Reference Scenarios	P2UC1, P2UC2, P2UC3
Success criteria	The interface is a story based scrolling experience.

Table 90: T4.4 - 8th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.4-08
Type	USE
Short name	User Guidance

Functionality ID	U-REQ-T4.4-01
Description & quantification	Include tooltips, guides, or onboarding to help users navigate the application and interpret visualisations.
Additional information (source, characteristics, short description of the process)	Include tooltips or pop-ups for detailed information.
Priority	MAN
Reference Scenarios	All
Success criteria	Users can easily navigate the application

Table 91: T4.4 - 9th Technical Requirement

Section	**Description**
Technical requirement code	T-REQ-T4.4-09
Type	USE
Short name	Access Control
Functionality ID	U-REQ-T4.4-01
Description & quantification	The application must support multiple user roles with appropriate access control.
Additional information (source, characteristics, short description of the process)	User roles allow users to access different parts or datasets of the application.
Priority	DES
Reference Scenarios	All
Success criteria	User roles are well-defined, and access control is effective

Table 92: T4.4 - 10th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.4-10
Type	PERF
Short name	Loading Time
Functionality ID	U-REQ-T4.4-01
Description & quantification	Optimize loading times for graphs and charts, especially when dealing with large datasets.
Additional information (source, characteristics, short description of the process)	Loading times must be kept at minimum when dealing with large amounts of data to be visualised.
Priority	MAN
Reference Scenarios	All
Success criteria	The application loads within an acceptable time frame for typical use cases.

Table 93: T4.4 - 11th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.4-11
Type	PERF
Short name	Error Handling
Functionality ID	U-REQ-T4.4-01
Description & quantification	Implement error handling and logging mechanisms for troubleshooting and maintenance.
Additional information (source, characteristics, short description of the process)	To better avoid mistakes, error handling mechanisms must be in place to better aid the development of the component.

Priority	MAN
Reference Scenarios	All
Success criteria	Errors and crashes are kept to a minimum.

Table 94: T4.4 - 12th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.4-12
Type	SUP
Short name	Codebase
Functionality ID	U-REQ-T4.4-01
Description & quantification	Maintain clean and well-documented code to facilitate future updates and enhancements.
Additional information (source, characteristics, short description of the process)	Having a clean codebase always allows for faster development, error handling and future upgrades to existing code.
Priority	DES
Reference Scenarios	All
Success criteria	No time is wasted in trying to understand previously written code.

Table 95: T4.4 - 13th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.4-13
Type	SUP
Short name	Version Control

Functionality ID	U-REQ-T4.4-01
Description & quantification	Implement version control (e.g., Git) for tracking changes and collaborating on development.
Additional information (source, characteristics, short description of the process)	By using version control, one can easily refer to previous versions of the application, or even have multiple ones at the same time.
Priority	MAN
Reference Scenarios	All
Success criteria	Code versions are kept in a remote registry such as GitLab.

Table 96: T4.4 - 14th Technical Requirement

Section	Description
Technical requirement code	T-REQ-T4.4-14
Type	SUP
Short name	User Support
Functionality ID	U-REQ-T4.4-01
Description & quantification	Provide user support and documentation for end-users.
Additional information (source, characteristics, short description of the process)	Develop an aiding mechanism that explains each interface's use case.
Priority	OPT
Reference Scenarios	All
Success criteria	End-users can access help resources and receive timely support when needed.

3 Architecture

The main goal of the reference architecture of AI4Gov is to design a limited and well-defined set of component functionalities satisfying the user requirements scheduled for the first period of the project.

3.1 Approach to the architecture specification

The architecture described in this document represents the first iteration and it would be extended by designing new features according to the consolidation or emergence of new user and technical requirements, which will occur optionally during the development of the project. The adopted methodology and techniques guarantee a good level of confidence in the architectural choices made so far. However, the choices made in the current version of the architecture can change as a consequence of revisions and updates of the overall requirements.

The approach adopted to describe the AI4Gov top-level architecture is based on existing literature and guidelines. The architecture reported here has the following objectives:

- It serves as the blueprint both for the system and the project developing it.
- It defines the work assignments, in terms of component functionalities, that must be carried out by separate design and implementation teams.
- It is the artifact that holds the key to post-deployment system understanding and/or mining efforts.

The definition of the AI4Gov architecture follows some key principles proposed and introduced by the GAIA-X architecture (Braud, 2021), which represents a novel model for the architectural design of software and cloud systems following a federated approach. This initiative aims to ensure data sovereignty, interoperability, and trustworthiness for businesses and organisations. The main quality attributes, considered in evaluating the alternative architectural choices are (i) the efficiency, (ii) the data privacy and security, (iii) the scalability and modifiability of the introduced solution, (iv) the interoperability and ease of realisation and (iv) the service reusability. This means that when multiple architectural alternatives were analysed, the solution considered most efficient, secure, scalable, interoperable, and reliable, in order of priority, has been adopted. The most efficient solution for each component is the one that minimises: (i) the time requested to perform its tasks, (ii) the amount of disk space requested to store internal data enabling its normal operation, and (iii) the overhead of communication with other components to exchange requested and provided data. When the simultaneous minimisation of these three parameters has not been possible because of conflicting conditions, 'the best' trade-off among them has been chosen. The criteria to define the best trade-off assigns the highest priority to the minimisation of the execution time requested to perform the tasks, secondly to the minimisation of the overhead of communication with other components, and finally to the minimisation of the amount of disk space required.

Accordingly, when high hardware requirements were expected, like CPU throughput and/or amount of memory, it was decided to adopt a dedicated server in order to minimise the execution time, even if such a choice does not optimise the efficiency in terms of communication overhead among distinct components (e.g., Decentralized blockchain-based infrastructure). For the components designed to be replicable, in case of high load, the dynamic horizontal scaling has to be taken into account (i.e., Policy-Oriented AI and NLP algorithms). Naturally, when possible, it was agreed to have more components running on the same server and in the same execution environment (e.g., Virtual Machine, Kubernetes nodes) to try to improve the communication as well (e.g., Policy Recommendation Toolkit).

Other quality attributes have been taken into consideration, like the modifiability of the system, which is one of the most important quality attributes considered during the design. Indeed, the adopted incremental approach implies continuous changes to the architecture and a highly modifiable system is strongly recommended. According to the adopted quality model, the modifiability is a complex attribute measured in terms of the extensibility of capabilities (i.e., the ability to add new functionalities with less impact on the overall system), the deletion of unwanted capabilities, the interoperability (i.e., the ability of the system to run under different executing environment) and the restructuring (i.e., the ability to support architectural configuration changes, such as optimizing system services, or creating reusable components).

The reusability of some of the main components is a highly desired aspect for AI4Gov, even if it is not mandatory. Thus, one secondary goal of the AI4Gov architecture is to reduce, as much as possible, the coupling between components/micro-services, while keeping each component as cohesive as possible. The reusability of the components is an aspect that will be further investigated during the next steps of the project.

The possible constraints of the infrastructure that will host the AI4Gov trial platform have been evaluated in terms of the number of required servers and, for each of them, the amount of the required computational resources and disk space. Moreover, the technologies needed for the correct operation of the components have been considered and they are further reported in Section 2. The objective of such investigation is to verify the feasibility of the demonstrator and to gather any technical requirements from the infrastructure providers (i.e., the pilots' infrastructure for the on-premises deployments). The result of such investigation leads to a gross grained estimation of the hardware requirements to execute AI4Gov components, provided by the component owners and submitted to the infrastructure provider for an acceptance validation. The estimated infrastructure requirements are reported in Infrastructure and Integration section.

The entire design phase, instead of focusing on specific tool/language and techniques that development teams use, or micro-managing the internal architecture of the components/micro-services, concentrates on the protocols and interactions between the various software artefacts and on the health and usefulness of the system as a whole.

3.2 Context

In the AI4Gov project, a software platform is being developed to provide different actors with several features. By analysing the list of pilot use cases, as introduced in the context of D6.1 – "Specification of UC Scenarios and Planning of Integration and Validation Activities V1", it is clear that the AI4Gov platform aims to support different categories of actors. More specifically, the below actors emerged:

- ***AI Model Builder***: AI researchers who can consult and test the state-of-the-art, unbiased techniques and approaches. The aim is to understand which ethical and bias considerations should be taken into account when building AI models so that they are fair, trustworthy and ethics-based.
- ***Ethical Expert***: ethical experts or reviewer that seeks to evaluate ethical considerations and bias in state-of-the-art AI projects and models.
- ***Individual/Citizen***: citizens or individuals that interact with the project's frontend applications. They will be trained and informed on AI, its replication in bias and ethics to increase their trust and awareness in policy-making procedures.
- ***Policy Maker***: the policy maker that seeks to provide enhanced, fair, unbiased, explainable, and trusted policies and predictions, and to track/monitor the day-to-day organisational operations.
- ***Admin***: the Identity or Data manager that will grant access and authenticate the users that will interact with the different AI4Gov components.
- ***External Sources***: other external system, such as: existing repositories, social networks, databases, etc.

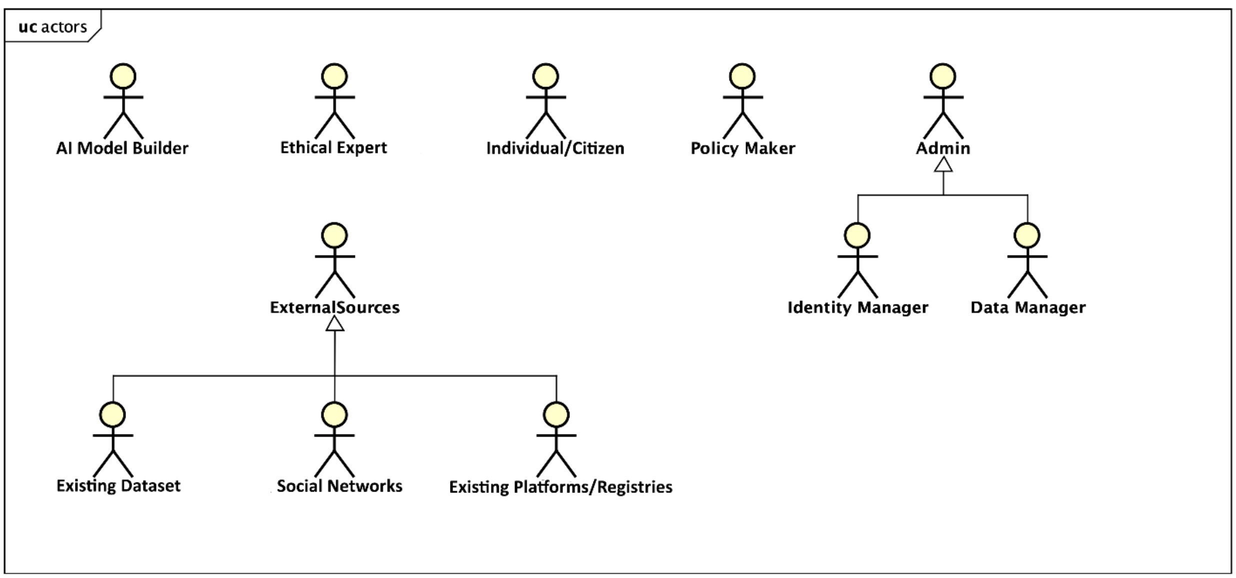

Figure 1: AI4Gov actors

Figure 1, places the actors in the context of the AI4Gov platform, while Figure 2 below indicates a snapshot that showcases all the involved actors that will integrate with the AI4Gov platform. This information is collected through a live document that will be populated throughout the whole lifecycle of the project. Any updates performed in this list will be reported in the next versions of this series of deliverables, i.e., D2.4 – "Reference Architecture and Integration of AI4Gov Platform V2" and D2.5 – "Reference Architecture and Integration of AI4Gov Platform V3".

Id	Name	Dataset to be used	Triggered By	*Other Actors involved in the UC*				Joint
01	Track Ethics/Bias on national AI-related policies	OECD	Policy Maker	Individual/ Citizen	Etchical Expert			Policy Maker, Individual/Citizen, Etchical Expert
02	Review ethical considerations and bias in other projects/documents	IRCAI Top100	Ethical Expert	Policy Maker	AI Researcher			Ethical Expert, Policy Maker, AI Researcher
03	Understand ethical/bias consideration should I think of when building my models	IRCAI Top100	Model Builders	AI Researche r	AI bias Reviewer			Model Builders, AI Researcher, AI bias Reviewer
04	Evaluate SDG's	SDG observatory	Policy Maker	AI Researche r	Admin			Policy Maker, AI Researcher, Admin
05	Monitor the Implementation of Pay As You Throw System	Waste Management	Policy Maker	Ethical Expert	AI bias Reviewer			Policy Maker, Ethical Expert, AI bias Reviewer
06	Monitor the scheduling/routing of garbage trucks and other vehicles	Telematic devices from tracks	Policy Maker	Admin				Policy Maker, Admin
07	Monitor and predict the traffic and parking	Ticketing issue data	Policy Maker					Policy Maker

Figure 2: Actors Involved in different Use Cases

3.3 Functional Overview

Starting from deliverable D6.1 – "Specification of UC Scenarios and Planning of Integration and Validation Activities V1" it is possible to enumerate the main use cases that drove the design phase, following their label with a short description of the envisaged behaviors. The overview of the principal function produced a list of common requirements, that also impact the overall architecture.

The first need, clear for all the pilots, is the ability of the platform to ingest data, coming from different sources and in divergent formats. These data, managed by following ethics principles and data governance approaches, are authenticated for their usage and access by integrating with the blockchain services of the project. Afterwards, those data are stored inside the AI4Gov storage repository for further queries and analytics. These actions will be implemented in the context of WP2 and WP3.

The same data will be accessed by all the components that provide the AI algorithms, analyses and aggregations that will be presented to the policy makers and all final users through the Visualization Workbench and the Policy Recommendation Toolkit. The latter will offer improved policy-oriented capabilities and processes. Those components, model and algorithms will be implemented and documented in the scope of WP3 and WP4.

3.4 Architecture Pillars

The architecture of AI4Gov includes five main pillars that realise the project's offerings, as depicted in Figure 3.

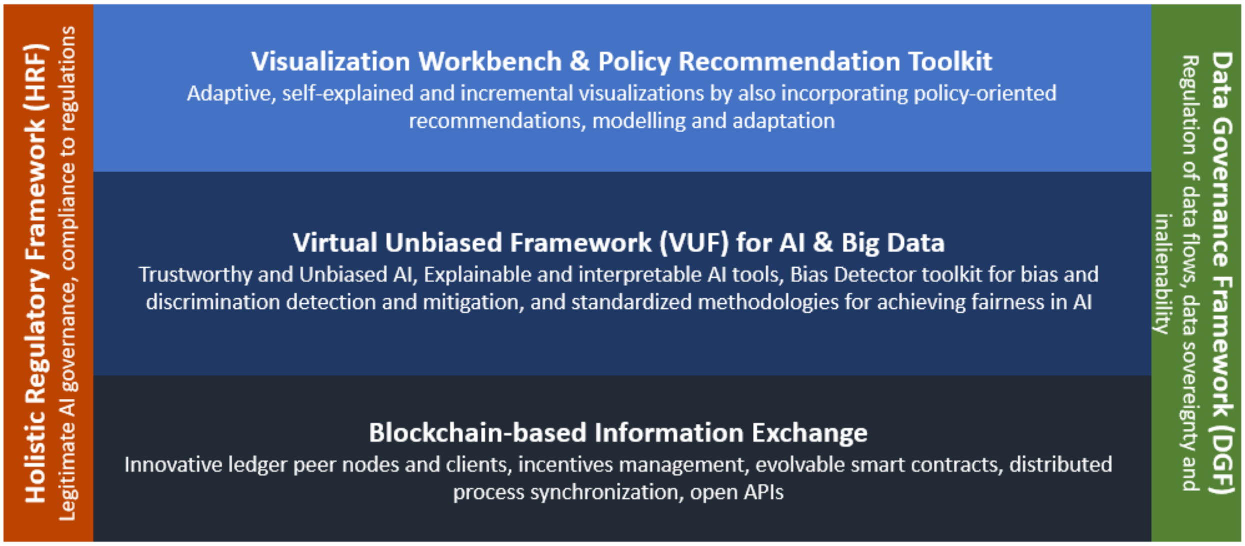

Figure 3: Architecture Pillars

These pillars are presented in the following paragraphs in a bottom-up manner:

1. The first pillar of the AI4Gov architecture is the Blockchain-based Information Exchange (BIE) block that will be based on the implementation of the project's distributed ledger infrastructure (blockchain). As a basis for the secure exchange of datasets and data across different sectors, AI4Gov will use a distributed ledger infrastructure (blockchain) within its holistic BIE component since it serves three complementary and important purposes, namely: (i) the security of the source data of the participating entities, (ii) the synchronisation of distributed processes that entail different actors, through enabling secure state sharing between them, and (iii) the incorporation as entities of the AI4Gov open platform and the data-driven research-oriented platform. In particular, there are three different types of blockchain infrastructures that could support the needs of AI4Gov, as outlined in Table 97. Despite their transparency benefits, public blockchains exhibit poor performance. On the other hand, permissioned blockchains are more appropriate in cases where networks of organisations need to establish and take advantage of a blockchain. Lastly, private blockchains might be appropriate for smaller scale enterprise use cases. The AI4Gov developments will take place over a permissioned blockchain infrastructure, given that permissioned blockchains will be the primary choice for demonstrating and validating the information exchange concept of the project. The following paragraphs highlight the project's approach towards the implementation and utilization of a permissioned blockchain. The AI4Gov blockchain infrastructure is fully decentralized, comprising distributed nodes or peer-nodes (or peers) and interconnected through the AI4Gov distributed ledger (or blockchain) protocols.

This pillar also incorporates technologies for interfacing and acquiring data from different sources and the project's Interim Repository (as described in Section **Error! Reference source not found.**), assessing their reliability and attaching the corresponding metadata to the sources and ensuring privacy enforcement for the collected data, using the developed smart contracts. The latter will identify attributes of data and stakeholders to ensure that all data decisions are according to the data governance rules specified by the data owners.

2. The second pillar of the architecture is the Virtual Unbiased Framework (VUF) for AI and Big Data that incorporates all data services/technologies provided by AI4Gov for the complete data path/lifecycle: modelling, cleaning, interoperability, linking/aggregation, storage, and advanced analytics, for constructing the required reusable models. Moreover, this framework also offers unbiased and explainability techniques and tools, such as the Bias Detector Toolkit and the eXplainability (XAI) Library of the project for bias and discrimination detection and mitigation, and methodologies for achieving interpretability and explainability of the developed AI models. Finally, in this pillar, policy-oriented AI and NLP techniques will allow stakeholders to acquire and analyse the corresponding information from citizens and individuals.
3. The third pillar refers to the frontend tools and mechanisms of the project and more specifically to the Visualization Workbench and Policy Recommendation Toolkit (PRT). This pillar will allow policy makers, ethical experts, researchers, and citizens to interact with the models and analytical tools as well as to specify their KPIs, requirements and constraints with respect to different policies (e.g., specification of the need for policies that can have a real-time impact due to emergencies). In addition, the toolkit facilitates the visualisation of policy monitoring in an adaptive and incremental way. Moreover, they will incorporate services for the identification of the required KPIs to model the policies and identify potential interdependencies with other policies within and across sectors at different levels (e.g., local, national, etc.). The PRT also includes tools for collecting evidence monitoring information both from the engaged citizens and from the population targeted by the policies, while also assessing the compliance to these policies and thus assessing the policies impact (based on the identified KPIs).
4. The fourth and fifth pillars of the architecture are not directly technical in terms of design and implementation but rather conceptual and transversal to all the other pillars. They govern the overall internal workflows and implementations of all internal components and micro-services and the establishment of both these frameworks assure that all the AI4Gov offerings conform to the required regulatory, ethical, legal and security aspects, thus ensuring the sustainability of the modelled policies in a trusted, fair, and interpretable way. These frameworks are vertically depicted in the Figure 3; the frameworks communicate in a bi-directional way with the other pillars by obtaining data from them and by specifying the data, analytical and policy-oriented tasks through the entire project lifecycle and roadmap.

Table 97: Types of Blockchain Infrastructures

Type	Definition	Characteristics	Suitability for AI4Gov
Public	Enable anyone to participate as a user or run a node.	Value anonymity, immutability, and transparency over efficiency	Medium to High
Permissioned	Operated by known entities such as public authorities.	Value immutability and efficiency over anonymity and transparency	High
Private	Operated by one organisation / entity.	Value efficiency over anonymity, immutability, and transparency	Medium

3.5 Revisions based on 1st Review Comments

In this section we highlight the specific technical-related updates and measures applied in the design and refinement of project's architecture and integration aligned to the comments received from the first reviewing period of the project. These measures revised the design of the architecture and further guided the integration approach of the project as they are presented in detail in later sections. To facilitate the identification of the recommendations and mitigation actions applied refer to , Table 98Table 97.

Table 98: Recommendations and Mitigation Actions

Recommendations	Mitigation Actions
It is crucial to maintain a dynamic perspective in order to adapt quickly to emerging technological changes, such as generative AI, and regulatory developments, to which we now need to be more attentive than ever.	Incorporation of emerging technologies and highlighting their role in the context of architecture. In that context, the VUF has been updated with the utilization of Bias Detection services and LLMs that are further embraced and demonstrated in the context of the project's use cases. SAX explanations are provided in natural language for interpretability, leveraging the power of LLMs.
Adapt and validate the use of blockchain technology to ensure traceability, security, and transparency. It is vital to monitor the integration and interoperability of technology components, adapting them to different use cases	Emphasized on the utilization of blockchain and integrate with the rest of AI solutions and components. Blockchain utilized in terms of data, policy, and models governance and integrated with several components (e.g., Visualization Workbench

	and PRT). Moreover, a Wallet and Mobile application was included in the architecture and offered for further exploitation to the citizens.
Continue the validation and refinement of tools such as VUF and XAI in real scenarios to assess their effectiveness. Differentiate clearly between the concept of bias and non-discrimination so as not to limit the potential of these tools.	A first round of the validation of technical components was completed and their genera in new use cases. The latter is further detailed in D6.4 – "Stakeholders' Feedback and Evaluation of the AI4Gov Use Cases V1". Discrete steps followed in the implementation of the tools towards the distinction between bias and discrimination, aslo in alignment with the HRF.
The technological approach between SIE, IBM, UBI, VIL, and others could be better integrated. During the review meeting, several inconsistencies were identified regarding XAI and SAX, data governance/ownership models (who owns the data), exploitation plan, blockchain/DAOs, etc. Clear roadmaps are needed to address AI's discrimination, bias, and data inequalities for the three use cases.	Improved integration of tools and frameworks, ensuring seamless collaboration and alignment of technological approaches such as XAI, SAX, and blockchain/DAO models. The incorporation of the Helix methodology principles, prioritized transparency, fairness, and strategies to mitigate discrimination, bias, and data inequalities.
Deepen the ethical and security analysis beyond the ALTAI guidelines, considering models developed in other countries and legal requirements for transparency and impact assessment of public algorithms.	The security and ethical aspects have been incorporated on the components development, and the architecture and integration approach have been aligned with the DGF and HRF of the project.

3.6 Alignment with the Helix methodology

One of the key comments received in the first review was the adoption of the Helix methodology core principles. In more detail, the Helix methodology offers a dynamic framework for innovation and collaboration, drawing from models like the Triple Helix, Quadruple Helix, and Quintuple Helix, each suited to different domains and approaches (Carayannis, 2022). The Triple Helix model focuses on the interplay between academia, industry, and government to foster knowledge-driven innovation (Etzkowitz H, 1995). Expanding on this, the Quadruple Helix model introduces civil society as a key stakeholder, emphasizing societal needs and public participation (Carayannis E. G., 2009). The Quintuple Helix model further incorporates environmental considerations, aligning innovation with sustainability and ecological awareness (Carayannis E. G., 2012). These models ensure adaptability across diverse domains by addressing technical, organizational, and operational dimensions. An overview of a Quintuple

Helix is presented in Figure 4, as introduced in a respective article (Barcellos-Paula, L., De la Vega, I., & Gil-Lafuente, A. M., 2021).

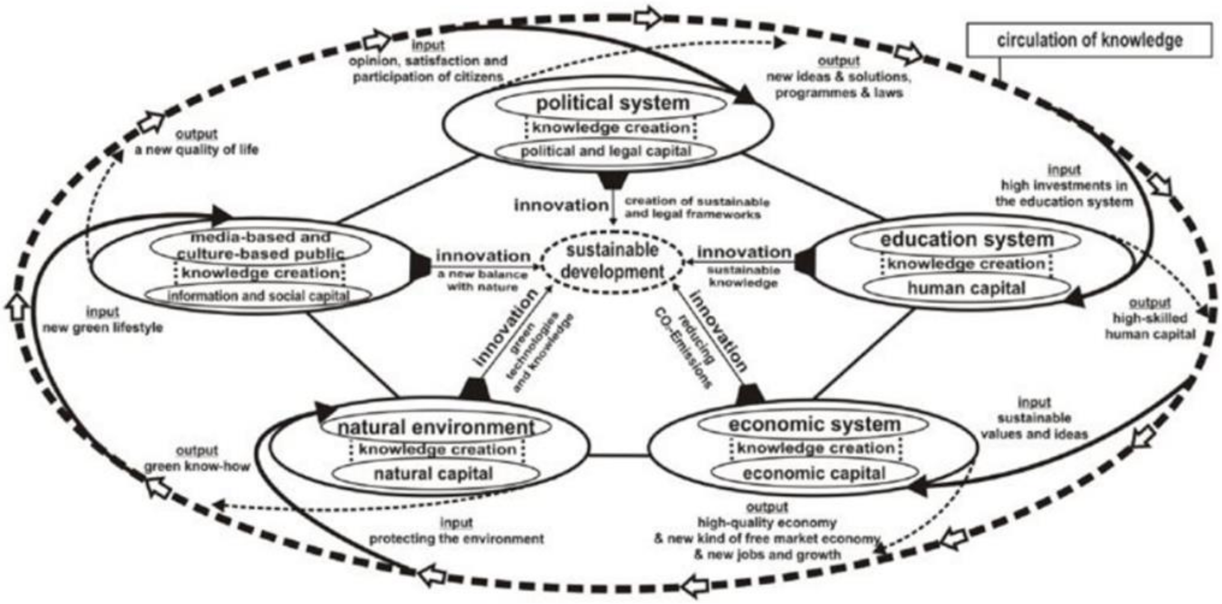

Figure 4: The Helix methodology

The methodology employs a multi-layered and interconnected approach, allowing for seamless integration across various levels of decision-making and implementation. In technical instances, the Helix methodology promotes tools and systems that enhance interoperability and collaboration. Organizationally, it facilitates governance structures that support stakeholder engagement and decision-making transparency. On an operational level, it ensures that workflows and processes remain adaptable to evolving requirements while maintaining alignment with core principles like inclusivity, sustainability, and innovation.

The translation of the Helix models into AI4Gov's practical and technical framework aims to provides a scalable and holistic approach to addressing complex challenges. It enables cross-sector collaboration and ensures that technological advancements are guided by societal, economic, and environmental goals, making it particularly effective for projects that require multidisciplinary coordination and adaptability.

The interconnection of the overall AI4Gov's architecture with this methodology started through the initial identification of the core principles of Helix Thinking: interdisciplinary collaboration, continuous feedback loops and iterations, adaptability, holistic integration, and feedback mechanisms. These principles form the foundation for achieving modularity, scalability, resilience, and flexibility in the project. These were translated into actionable layers and associated activities, illustrating how they are implemented to create a dynamic, multi-layered system that ensures innovation and effectiveness.

These activities are organized into four layers: **User Interaction, Orchestration & Decision-Making, Processing & Analytics, and Data Layer**. Each layer corresponds to specific principles of

Helix Thinking, showcasing the connection between theoretical frameworks and practical implementation, and are further. This alignment ensures that the project maintains a structured, yet adaptable approach to achieving its goals while promoting collaboration, transparency, and trust among stakeholders.

Table 99: Helix principles and corresponding activities

Helix Principle	Corresponding Activities	Relevant Layer(s)
Interdisciplinary Collaboration	Multi-stakeholder integration through Human-In-The-Loop (HITL) processes, multi-modal interfaces (Visualization Workbench), and feedback mechanisms like blockchain and participatory tools (i.e., PRT and Wallet dApp).	User Interaction, Processing & Analytics, Data Layer
Continuous Feedback Loops	Implementation of feedback systems such as explainability mechanisms (i.e., XAI, RAG, Minimal Sufficient Reasons (MSRs)), iterative policy refinement, and dynamic orchestration processes.	Orchestration & Decision-Making, Processing & Analytics
Adaptability	Scalability of orchestration engines (Kubernetes), AI/ML models, and resource management systems to adjust dynamically to changing demands.	Orchestration & Decision-Making, Processing & Analytics
Holistic Integration	Seamless connectivity across layers, incorporating AI/ML models, LLMs, chatbots, monitoring tools, and governance systems (i.e., Data Lake, policy graphs).	User Interaction, Orchestration & Decision-Making, Data Layer
Feedback Mechanism	Establishment of transparent redress mechanisms, logs, and continuous monitoring to track performance and improve trust among end-users.	User Interaction, Processing & Analytics

4 Software Architecture

In this section the main components that constitute the AI4Gov platform are detailed, considering the requirements that emerged in this deliverable.

4.1 Overall and Updates since D2.3

Figure 5 illustrates a detailed diagram of the overall architecture of the AI4Gov platform. A standard modelling language was not adopted for this specific diagram, to better explain also to non-technical readers the connections and the dependencies among components.

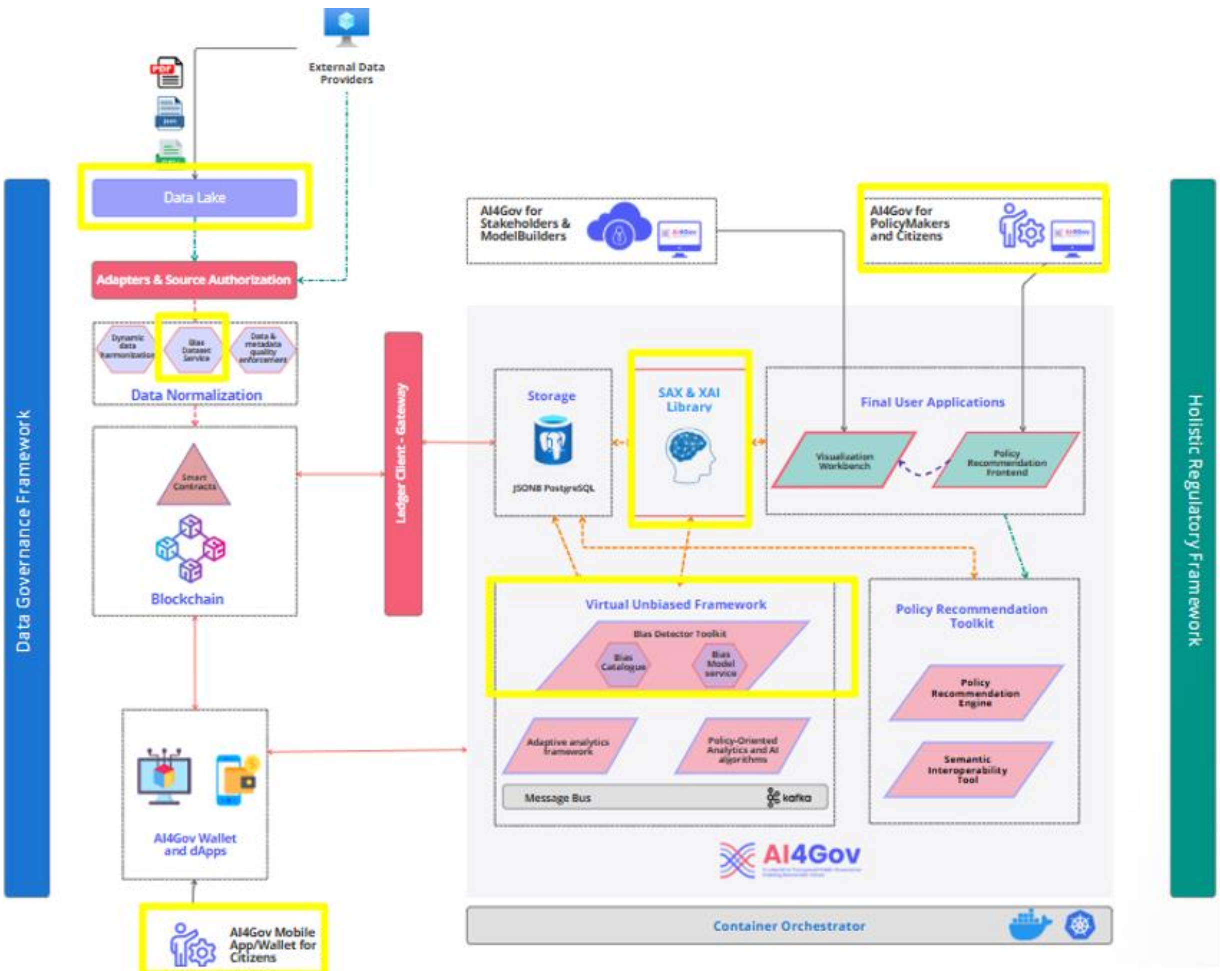

Figure 5: AI4Gov Reference Architecture

In particular, the shape of the software artefact mainly brings the concept of communication pattern, while a hexagon denotes the services that will be provided in terms of data normalisation, consistency, and quality. The triangle describes the smart contracts that will be implemented by the blockchain infrastructure and will govern the overall data and policy exchange across the whole lifecycle of the project. The rectangles define the tools and mechanisms that will be developed for the integration of the blockchain with the internal AI4Gov components and the project's Interim Repository. Moreover, the ping rectangles define the components that communicate a/synchronously, using the Kafka message broker or any API, while the green rectangles define the frontend applications. These details are further presented in a legend box that accompanies for further clarification and information to the end user as presented in the below figure, Figure 6.

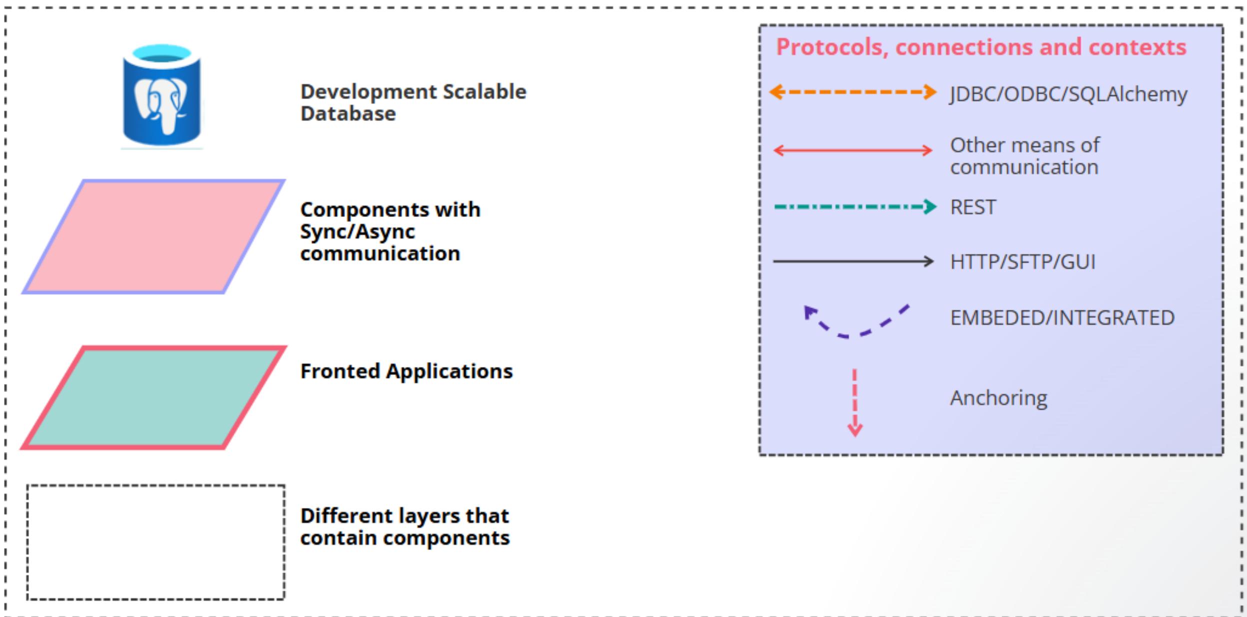

Figure 6: Architecture details

What is more the yellow rectangles presented in the above figure highlight the updates reflected in this second version of AI4Gov's Reference architecture in comparison to the initial one. The architecture has been enhanced with several key components and updates, reflecting a commitment to scalability, flexibility, and adherence to the project's DGF and HRF, as further detailed in later subsections. These updates ensure that the platform meets regulatory and ethical requirements, as well as delivers an improved environment for developers, and enhanced user experience and applications for stakeholders and citizens.

To address issues of fairness and accountability, the architecture now includes a Bias Detector Toolkit and a Bias Catalogue. These tools identify, document, and manage biases in datasets and AI models, enhancing trust and inclusivity. In that direction, the Bias Catalogue, Bias Dataset Service and Bias Model Service (subcomponents of the Bias Detection Toolkit) streamline the handling of datasets and models by providing mechanisms to detect and mitigate biases across the data pipeline. Together, these features ensure compliance with ethical AI principles and bolster trustworthiness in AI decision-making processes. Moreover, the SAX & XAI Library is offered outside of the VUF as it was decided not to be integrated and exposed through specific API endpoint through this framework. More details follow in the context of D4.2 – "Trustworthy, Explainable, and Unbiased AI V2" showing explainable AI capabilities, ensuring transparency and interpretability in the AI models deployed.

The integration of a Data Lake, instead of a simpler Interim Repository, demonstrates a significant step forward, serving as a centralized repository that supports both structured and unstructured data in alignment with the DGF and HRF. It facilitates data exploration and discovery, allowing stakeholders and citizens to interact with datasets transparently and effectively. Robust data governance and security mechanisms ensure that the Data Lake supports privacy, traceability, and interoperability, promoting secure collaboration among stakeholders. Its seamless integration with the AI4Gov Big Data Ecosystem further enhances the platform's capacity to process and analyze large-scale datasets, enabling scalable and flexible operations.

The architecture has been updated to include citizens as key stakeholders, supported by the Visualization Workbench, and the AI4Gov's Wallet and Mobile Application. These tools empower citizens by providing direct interfaces for visualization of the results, access to specific tools of the project, transparent policymaking, and secure access to the data. The newly introduced citizen-oriented approach fosters inclusivity and trust, making the architecture a model for transparency and ethical AI-driven policymaking. The updates also emphasize scalability and flexibility, ensuring that the system can dynamically adapt to growing demands while supporting diverse user needs and workflows.

The goal and the main envisaged interactions of the components are further described below. It should be noted that two conceptual frameworks will also be detailed. Although, these frameworks are not technical implementations, they will govern and regulate the functionality of all the different components and workflows of the AI4Gov platform. Thus, a brief description of them also follows, for more details refer to D2.2 – "AI4Gov Holistic Regulatory Framework V2" and D3.2 – "Decentralized Data Governance, Provenance and Reliability V2" respectively.

4.2 Data Governance Framework

The Data Governance Framework (DGF) of the AI4Gov platform will be aligned with the Data Governance Act (Commission, 2020) and Findability, Accessibility, Interoperability, and Reusability principles providing the capability of multiple entities' governance levels. It will act as project's values-based framework to inform data governance and regulate the management and processing of data from the AI and Big Data applications within the project, building upon prior knowledge gained by the PolicyCLOUD project (Kyriazis, 2020). Its main focus lies in governing the entire data pipeline and policy making procedures than just governing individual data elements. The framework and its accompanying tools will offer protection and privacy enforcement for the data and will ensure that decisions across the complete path follow specific protocols, regulations and legislations and are in-line with the Holistic Regulatory Framework (HRF). To this end, proper SLAs will be established as part of a smart contract between Data/Service Provider and Data/Service Consumer. Technical solutions will be accompanied by organisational rules or legal contracts and vice versa. Control points in different layers of the data-processing systems will be implemented in the above-listed technical layers. Moreover, regulatory support will be embedded in the governance, using the project's tools, including support for key regulations (e.g., GDPR, AI Regulation, EU AI Act) stemming from the HRF which will examine and assess the compliance of the project's technologies to applicable laws and regulations.

4.2.1 Alignment of the architecture with the DGF

The architecture presented above aligns closely with the DGF by adhering to its core principles of data FAIR-ification. The modular and scalable design ensures that data from multiple sources can be integrated seamlessly, enabling efficient governance at various levels. The inclusion of mechanisms for user registration, authentication, and access control ensures data accessibility while maintaining strict compliance with privacy and protection regulations, such as GDPR and the EU AI Act. The support of interoperability across heterogeneous datasets ensures that all data ingestion, storage, and processing steps align with specific protocols and legal requirements, the architecture fosters a unified and transparent governance model. While, the inclusion of traceability features provides accountability and visibility into the data pipeline, ensuring that decisions and actions taken are aligned with ethical and regulatory standards.

The integration of a Data Lake further strengthens alignment with the DGF, as also noted before, by enabling a centralized and flexible repository for managing structured and unstructured data. The Data Lake facilitates the seamless ingestion, storage, and processing of data from diverse sources while adhering to FAIR principles. What is more, the architecture also reflects the DGF's commitment to governing the entire data pipeline rather than focusing solely on individual data elements, as it incorporates mechanisms to enforce service-level agreements (SLAs) through smart contracts, it ensures that responsibilities and rights between data providers and consumers are clearly defined and upheld. Organizational rules and legal contracts are complemented by robust technical solutions, with control points and logging mechanisms embedded at every layer of the data processing system.

4.3 Holistic Regulatory Framework

The Holistic Regulatory Framework (HRF) in the AI4Gov Project refers to a comprehensive taxonomy and a set of guidelines intended to govern the use of AI and Big Data in the context of democracy and EU values. This framework aims to ensure that AI technologies, especially when applied to governmental processes or services, adhere to fundamental rights and values, do not perpetuate bias or discrimination, and respect existing laws, regulations, and ethical standards. It seeks to address these concerns, ensuring that the platform is just, equitable, and compliant with prevailing standards and laws. An introduction to the HRF is available in AI4Gov D2.1 - "AI4Gov Holistic Regulatory Framework V1", while its final version D2.2 - "AI4Gov Holistic Regulatory Framework V2" finalized and concluded the respective works in M18. Thus, this the updated version of AI4Gov platform provided in the context of this deliverable is aligned with the recommendations and propositions introduced by this specific framework of the project.

To ascertain a comprehensive understanding of the challenges and opportunities in integrating AI and Big Data in governance and policy making, a systematic procedure was initiated with the objective of constructing the AI-based Democracy HRF. The core procedure revolved around implementing the Delphi method to undertake a SWOT (Strengths, Weaknesses, Opportunities, and Threats) analysis for the AI-based Democracy HRF. Afterwards, a panel of experts with proficiency in AI governance and policy making was chosen to participate in expert panel. Leveraging the insights from the SWOT analysis, preliminary guidelines, and recommendations for the HRF were formulated. The focus was to ensure that the deployment of AI and Big Data in policy management remained democratic, transparent, and aligned with ethical standards.

Currently, the main aspects (in relation to the tasks of WP2) of AI4Gov's HRF are:

- **Defining Bias and Discrimination**: One of the main objectives of WP2 and of the HRF is to provide a holistic definition of bias, discrimination, unfairness, and non-inclusiveness. This involves aligning both technical definitions (from the AI/tech side) with social science definitions and understanding how these two aspects interact when AI and Big Data technologies are developed or deployed.
- **Ensuring Compliance with EU Values and Regulations**: The framework seeks to evaluate the AI4Gov Platform's alignment with current EU regulations concerning fundamental rights and values. This ensures that the technologies developed and deployed respect the rights of citizens and adhere to important regulations like the General Data Protection Regulation (GDPR).
- **Qualitative Analysis of Rights and Values**: Task T2.1 focuses on understanding how traditional (non-AI) biases might currently affect the rights and values of certain citizen groups, such as ethnic minorities, migrants, religious groups, and persons with disabilities. This task and this part of the HRF aims to uncover areas where discrimination might be overlooked, especially in relation to existing EU regulations on human rights protection.
- **Specification and Design of the Framework**: Task T2.2 emphasises the HRF need to proceed with a mapping of the existing processes to a policy management lifecycle and highlighting enhancements through proposed AI solutions. The AI4Gov HRF will be based on qualitative analyses of fundamental rights, EU values, legal activities, and ethical protocols. It will ensure that citizens are protected from potential abuses resulting from AI and Big Data use and that the framework adheres to existing laws, protocols, and ethical recommendations.
- **Reference Architecture**: In relation to Task T2.3 (the task of reference also for the current deliverable report) the HRF acts as the ethical and regulatory compass, ensuring that the reference architecture developed is not only technologically robust but also ethically sound, legally compliant, and fundamentally aligned with the overarching goals and values of the AI4Gov project. The HRF will play a role in how different components of the architecture will interact with each other, both in terms of data flow and functional hierarchy, ensuring that the interactions remain compliant with the ethical, legal, and functional standards the framework sets forth. In the realm of AI and Big Data, the flow and processing of information are of paramount importance. The HRF provides guidelines on how data should be collected, processed, stored, and shared, ensuring that privacy, fairness, and security are maintained throughout. Moreover, it needs to be noted that the HRF is not just a set of guidelines or a checklist for the project. It will serve as a foundational blueprint for the entirety of the AI4Gov architecture detailed in

Task T2.3. By aligning with the HRF, the project ensures that the architecture, and by extension, all its subsequent developments and deployments, operate within a framework that respects human rights, EU values, and ethical considerations.

In essence, the HRF in AI4Gov provides a thorough, multi-faceted approach to ensuring that AI- and data-related technologies are developed and used responsibly, ethically, and in line with the fundamental rights and values of the European Union (EU).

4.3.1 Alignment of the architecture with the HRF

One of the key propositions of the updated architecture that also demonstrates its alignment with the HRF is the inclusion of the citizens as end-users and their empowerment through enhanced accessibility to information, with the Visualization Workbench providing a direct interface for exploring data insights and analytics. This transparency fosters trust, allowing citizens to interact with and understand the data and policy decisions derived from it. Moreover, the inclusion of a Decentralized Wallet-based mobile application ensures that citizens can securely access their data, participate in data-sharing agreements, and monitor how their data contributes to policymaking. These tools enhance engagement, offering real-time updates and transparent views of how decisions are made, aligning with the HRF's focus on ethical and inclusiveness.

Moreover, the Decentralized Wallet-based mobile application enable users to consent to data-sharing agreements through smart contracts, reinforcing user control and data sovereignty. This feature also allows citizens to track the use of their data and the outcomes it informs, contributing to transparent and participatory policymaking processes. The Visualization Workbench further complements this approach by providing stakeholders and citizens with clear, actionable insights, bridging the gap between complex data pipelines and policy outcomes. Together, these components create a transparent ecosystem that empowers citizens, ensures compliance with regulatory frameworks, and supports informed decision-making while promoting accountability at all levels of governance.

Furthermore, in the context of D2.2 - "AI4Gov Holistic Regulatory Framework V2" key ethical and technical principles were highlighted as main priorities and key instances of the HRF. The proposed architecture aligns with these principles to ensure trustworthy and transparent AI systems. Table 100, highlights this interconnection and demonstrates how through the embedding of regulatory compliance tools and aligning workflows with applicable laws, the architecture creates a governance ecosystem that is adaptable, secure, and transparent, ultimately supporting informed policymaking and ethical data management. This dual-layered approach of combining legal and technical safeguards ensures compliance with both the DGF and the overarching HRF.

Table 100: Alignment of Architecture with HRF's main priorities

Category	Detailed Explanation
Human Oversight	Human-In-The-Loop (HITL), Human-On-The-Loop (HOTL), and Human-In-Control (HIC) ensure that humans are involved in key decision-making steps across the data lifecycle. This includes validation of data quality, oversight of AI processes, and final decision-making authority for critical outcomes.
Compliance to Ethical Guidelines	Compliance is designed to align with GDPR, the EU AI Act, and other human rights regulations. Tools like the Data Lake and the Bias Detection Toolkit ensure fairness, non-discrimination, and transparency in data governance and AI deployment.
Data Governance	Informed consent, privacy preservation, and data anonymization are prioritized. Encryption and FAIRification principles ensure that data remains findable, accessible, interoperable, and reusable while being securely governed by the Data Governance Framework (DGF) and blockchain.
Explainability (Transparency)	XAI and SAX approaches ensure clear and interpretable explanations of AI decisions, enabling citizens to understand how these decisions affect them and fostering trust in the AI system.
Redress Mechanisms	Feedback channels, such as complaint systems and appeal processes, are established to empower citizens to challenge or seek explanations for AI-based decisions that adversely affect them. These mechanisms enhance accountability and transparency.
Auditing	Continuous monitoring of AI performance is implemented using blockchain logs, the Visualization Workbench, AI models, and the Bias Detection Toolkit. This ensures the high quality of provided services, identifying errors or biases early in the pipeline.

4.4 Blockchain Framework

Generally, the blockchain framework consists of all necessary infrastructure that facilitates decentralized storage of documents and the execution of smart contracts; code that can ran and be validated by each one of the peers. It allows transparency in data storage and code execution and potential integration with EBSI, the EU blockchain infrastructure.

It is composed of three main packages:

- The ***blockchain*** which contains the core decentralization infrastructure together with the mechanisms of storing data in a decentralized manner and executing smart contracts.
- The ***Wallet and dApps***, which consist of the wallet application and the set of decentralized applications; it is basically a means of logging-in and running applications that rely on smart contracts.
- The ***Ledger client***, this is the main interface that allows centralized components to access the blockchains and post or retrieve information. As the blockchain is a closed system, any data posting should be validated by the blockchain.

Each one of these components is going to be analysed separately in the following sections.

4.4.1 Blockchain

4.4.1.1 Objective of the software Artefact

The objectives of the blockchain can be summarized as follows:

- Provision of a transparent storage mechanism for uploading and reading data and reports. Through the utilisation of blockchain technology, any data uploaded can be digitally signed by the user who uploads it, including any subsequent changes or versions of the file. Conversely, each peer can verify that the data were uploaded by the original user and universally agree upon its contents. Data security and accountability are thus enhanced.
- Provision of a mechanism to run code in a decentralized manner (smart contracts). Code that runs on the blockchain is open in the sense that all peers can check the implemented business logic, which is transparent in each execution. Any piece of code running on the blockchain must be validated by the network, thus consensus is reached regarding the results of the code. A recommendation engine implemented as a smart contract for example, will produce recommendations that can be proved to be generated by the underlying business logic without being tampered with.
- Enforcing data harmonization and metadata quality management. Relevant to the previous objective, blockchain will provide a set of smart contracts for validating that data conform to the commonly accepted norms (e.g., formats and schema) and that any normalization is approved by the number of peers defined by the blockchain governance model.

As a special note, it should be added that "data storage" in the case of blockchain does not refer to storing files directly into the blockchain. While this is a possibility, it is against standard practices. Instead, special pointers that point to off-chain data silos should be implemented. Robustness and transparency of data are further enhanced if the storage is also decentralized (e.g., IPFS). For the purposes of AI4Gov, the OpenDSU framework will be considered for storage. OpenDSU is a blockchain-agnostic and storage-agnostic framework that allows sharing of data and anchoring them via blockchain pointers.

4.4.1.2 Interactions with other components

The blockchain interacts with other components through the following interfaces:

- ***Wallet and dApps***: This is the main mechanism by which any component or piece of software that needs to run a smart contract can access the blockchain. Though it is technically possible to implement direct interfaces to centralized components, this will complicate the architecture, and it will be difficult to track and monitor all the applications that directly invoke calls to the blockchain gateway. Instead, there is a strict requirement that any component that needs to run a smart contract must do so via a specially implemented dApp. For example, if the Policy Recommendation toolkit needs to run a set of rules on the blockchain, a special dApp that will access the blockchain to run the code will be implemented and stored in the Wallet dApp marketplace.
- ***Ledger Client***: This is the main mechanism by which data that are saved in the AI4Gov storage are anchored into the blockchain.
- ***Adapters & Source Authorization***: This is the main mechanism for storing and retrieving information from external providers. As these data are not, strictly speaking, part of the AI4Gov ecosystem, this module will also implement extra authentication/authorization mechanisms to ensure the identity of the external provider and will also have to validate and align the external data.

4.4.2 AI4Gov Wallet and dApps

4.4.2.1 Objective of the software Artefact

The main objective of this component is to provide a mean of accessing the smart contracts of the blockchain both from a high-level and a low-level perspective:

- From a low-level perspective, the Wallet will provide the required keypairs by which users can access the blockchain and will also provide the mechanisms for invoking smart contracts. The OpenDSU[2] framework will also be considered here for this component as it will allow the deconstruction and dynamic reassembly of the dApps' execution environments, providing extra flexibility in dApps implementations.
- From a high-level perspective, the component will offer a Graphical user Interface (GUI) to end users that will allow them to access the dApp marketplace and find and execute any dApp needed.

[2] https://opendsu.com/

4.4.2.2 Interactions with other components

The Wallet and the dApps interact directly with the blockchain to perform identification operations and invoke smart contracts. For other components they only interact by implementing a special dApp that fulfils the required functionality. If the visualisation workbench, for example, needs some blockchain statistics and, therefore, access to blockchain data, it should do so by the appropriate dApp.

4.4.3 Ledger Client - Gateway

4.4.3.1 Objective of the software Artefact

The Ledger Client main objectives are:

- Store data into the blockchain by partitioning them appropriately in the offline storage and anchoring them via specifically implemented points into the blockchain.
- Retrieve requested information by authenticating both the user and the document via the appropriate Decentralized Identifiers (DIDs) and providing the information. If the file is shared, the Ledger Client will reconstruct the resource.

4.4.3.2 Interactions with other components

The Ledger Client interacts with the AI4Gov data storage and with the blockchain infrastructure.

4.5 Virtual Unbiased Framework

4.5.1 SAX/XAI Library (IBM)

4.5.1.1 Objective of the software Artefact

The main objective of the SAX/XAI library is to cater a toolkit of core enablement and realisation services for the generation of process and situation-aware, user-centric, and sound explanations via the acquisition of timestamped process event logs as the main input, complemented by possible user preferences about features to be considered and other relevant situational enrichments. The library applys a set of methods and techniques on those event logs for discovering sound view of the process model which in turn, drives sound analysis of various process features and external factors affecting process decisions and outcomes.

4.5.1.2 Interactions with other components

The core SAX/XAI library is inherently dependent on some core services that are also made intrinsically available via the utilisation of open-source libraries and/or APIs to external services providers. This includes a fundamental functionality for process discovery and causal discovery utilising the open-source LiNGAM library, XAI methods and LLMs. Additionally, SAX/XAI library can utilise the Data Storage component, either as a possible data source for processing event logs, or possible storage and retrieval solution for produced explanations. Alternatively, the same explanations can be stored on the blockchain as a means for trustworthy and verifiable storage for non-repudiation and provenance of business process analysis. SAX/XAI library may utilise the Bias Detector Toolkit in the XAI and LLM synthesis modules to verify usage of only certified, unbiased models and tools in its underlying implementation layers.

4.5.2 Bias Detector Toolkit

4.5.2.1 Objective of the software Artefact

The main objective of the Bias Detector Toolkit is to provide a catalogue of certified models, techniques and processes for bias mitigation. The audience for the Bias Detector Toolkit is primarily ML developers, who can access the bias mitigation knowledge in a condensed, user-friendly way. Moreover, the general public can benefit from the Bias Detector Toolkit, specifically regarding the raising of awareness and the explanatory part, where bias in AI is introduced and augmented with examples and the utilisation of the state-of-the-art scrolly telling technique.

4.5.2.2 Interactions with other components

Bias Detector Toolkit is being developed in close interaction with the project's Interactive Self-Explained Visualization Workbench.

4.5.3 Adaptive Analytics Framework

4.5.3.1 Objective of the software Artefact

As mentioned in Section 2.5.1, the "Adaptive Analytics Framework" is being developed in the context of T4.3 – "Improve Citizen Engagement and Trust utilising NLP". The scope of this component is to develop the needed ML models for efficiently performing predictive analytics and optimised resource allocation to satisfy the needs of the pilots and assist policy makers.

4.5.3.2 Interactions with other components

The results of the data analysis should be stored in the AI4Gov platform and be available to the users through the Visualization Workbench. Therefore, the Adaptive Analytics framework should be integrated with the Data storage component to store the aforementioned results, as well as the Visualization Workbench component in order for those results to be visualised. Moreover, this component should also be integrated with the XAI Library component since the developed ML models of the component should be fed into the XAI Library component. Lastly, this component should follow the guidelines proposed by the Bias Detector Toolkit component in order to address possible bias in the whole workflow of the Adaptive Analytics Framework.

4.5.4 Policy-Oriented Analytics and AI algorithms

4.5.4.1 Objective of the software Artefact

As also mentioned in section 2.5.1, "Policy-Oriented Analytics and AI Algorithms" is being developed in the context of T4.3 – "Improve Citizen Engagement and Trust utilising NLP". Its aim is to develop several NLP algorithms to analyse large volumes of text data and also assist the respective AI experts. This component consists of the following subcomponents:

- ***Question Answering Service***: this service will provide the necessary tool for allowing the AI experts, developers, and policy makers to perform queries on the OECD papers regarding, among others, raising awareness among them of AI solutions.
- ***Time Series Analyser***: this tool will support the analysis of time series and historical data to discover possible trends that will support the corresponding users in the water management cycle and the parking tickets monitoring use cases.

Lastly, the aforementioned subcomponents should follow the guidelines proposed by the Bias Detector Toolkit component in order to address possible bias in the whole workflow of the Policy-Oriented Analytics and AI algorithms component.

4.5.4.2 Interactions with other components

The Question Answering Service should interact with the Visualization Workbench component, since, through this, the users will be able to communicate with the Question Answering Service and perform their queries on the OECD papers. Furthermore, the Time Series Analyser subcomponent should also be integrated with the Visualization Workbench component since the results of these analyses should be visualised. Furthermore, the results should also be stored in the AI4Gov platform, meaning that there should also be an interaction between these components and the Data Storage component.

4.6 Policy Recommendation Toolkit

The Policy Recommendation Toolkit aims at facilitating the selection of the best available policies based on user needs, as these are input to the Toolkit via a set of target goals. It consists of two main aspects: the policy definition aspect, which is responsible for representing the policies and aligning the semantics of existing policies and user input, and the recommendation engine itself, which is responsible for providing the suggested policies.

4.6.1 Policy Recommendation Engine

4.6.1.1 Objective of the software Artefact

The main objectives of the Policy Recommendation are:

- Offer the functionality of searching the policy repository using appropriate filters.
- Consulting an ML based engine that, based on hard and soft constraints, will recommend the appropriate policies that best match the user defined target KPIs.
- Provide visualisation for comparing policies regarding defined criteria.

4.6.1.2 Interactions with other components

The Recommendation Toolkit naturally interacts with the Semantic Interoperability Tool to retrieve the common semantics for policy representation; this representation is important for unambiguously triggering the correct rules based on the proper values. It also interacts with the Visualization Workbench to provide its graphical output to end users. For implementing decentralized rule sets (via smart contracts), the toolkit will interact with the blockchain via an appropriately designed dApp.

4.6.2 Semantic Interoperability Toolkit

4.6.2.1 Objective of the software Artefact

The Semantic Interoperability Toolkit is responsible for mapping policy data into a common vocabulary by implementing both the model and the mechanisms for data uplifting and semantic enrichment.

4.6.2.2 Interactions with other components

The Toolkit interacts with the AI4Gov storage to save its models and, most importantly, access the document that needs uplifting and enrichment. Naturally, it interacts with the Policy Recommendation Toolkit to provide its interface to source data.

4.7 Final User Applications

4.7.1 Visualization Workbench

4.7.1.1 Objective of the software Artefact

The Visualization Workbench is being developed in the context of T4.4 – "Interactive Self-Explained Visualization Workbench". It is a versatile web application that serves as a powerful tool for data visualisation and analysis of AI policy-making and bias assessment. This component offers unique and user-friendly interfaces with intuitive design, making it easy for users to explore and interact with various use cases of the project. With robust performance and scalability, it ensures a seamless experience for users.

4.7.1.2 Interactions with other components

The Visualization Workbench will interact with the majority of the architecture's components, as it serves the role of the main platform of the project. One major component is the Policy Recommendation Toolkit, which will embed its frontend application within the Workbench. Also, there is direct communication with the Bias Detector Toolkit, which will incorporate all its bias prevention mechanisms within a separate interface of the Workbench. The Blockchain Framework component will also be a node of communication for the needs of the decentralised assets of the architecture as well as the authorisation and authentication of users by providing the needed access control and permissions. Regarding the scope of the NLP component, a prompt chat interface will facilitate direct communication between the two components. Lastly, all results from the analytics components that will be stored in project database must be visualised in the Workbench. Thus, an interconnection with the project's data storage is necessary for the visualisation requirements.

4.7.2 Policy Recommendation Frontend

4.7.2.1 Objective of the software Artefact

The Policy Recommendation Frontend will be integrated with the Visualization Workbench to offer an enhanced visualisation of the examined and executed policies. Its main objectives are:

- The provision of a user-friendly interface through the development of an intuitive frontend interface that allows policymakers and stakeholders to easily access and interact with policy recommendations.
- Enhanced policy visualisation by the implementation of knowledge graph visualisations to present complex policy insights in a visually informative manner, aiding policymakers in understanding key issues and trends.
- The provision of options for users to customise policy recommendations based on their specific needs and by setting their own KPIs and policy rules.
- Make the frontend accessible to a wide range of users, including those with disabilities, to ensure inclusivity in policymaking.
- Incorporate collaborative tools that enable policymakers to collaborate and share policy recommendations with colleagues and stakeholders.

4.7.2.2 Interactions with other components

The Policy Recommendation Frontend will be directly integrated with the Visualization Workbench, as well as with the Policy Recommendation Engine that acts as the backend of this tool. An integration with the project's data storage will offer the provision of adaptable policies.

4.8 Data Storage

4.8.1 Objective of the software Artefact

The utilisation of JSONB PostgreSQL, an advanced and versatile data type, offers a powerful and flexible solution for storing, querying, and manipulating semi-structured and nested data within relational databases (Lerner, 2014). JSONB stands for "binary JSON," and it provides the best of both worlds by combining the flexibility of JSON with the efficiency of a binary format. This combination is in alignment with the scope, the architecture principles and the quality measures that the AI4Gov project seeks to apply to its design and deployment. The latter is agreed between its consortium members based also on the divergent formats that describe the provided pilot datasets. These formats range from data collected from sensors to raw documents and text. One of the key advantages of using JSONB in PostgreSQL is its ability to accommodate alternate data structures. This dynamic nature will be particularly valuable in the scenarios that will be addressed in the context of the project. Moreover, developers can efficiently insert, update, and query JSONB data, all while maintaining the reliability and transactional capabilities of PostgreSQL, by also offering robust indexing and querying capabilities. With the introduction of the GIN (Generalized Inverted Index) and GiST (Generalized Search Tree) indexing methods for JSONB, querying JSONB data becomes highly efficient and scalable. This capability will be leveraged by the AI4Gov project as complex, ad-hoc queries against semi-structured data are required and as data retrieval and filtering are crucial for the provision of improved insights and policies.

4.8.2 Interactions with other components

The AI4Gov data storage will be integrated with almost all different components of the project for the storage and retrieval of the respective data and policies. It will consist of various components and will provide different data connectivity mechanisms, which allow the integration of it with different and diverse data providers, data consumers or other analytical processing frameworks. The exact nature and type of these mechanisms will be implemented in the context of T2.4 – "Integration of AI4Gov Platform and Tools" and reported in the next versions of this series of deliverables, i.e., in D2.4 – "Reference Architecture and Integration of AI4Gov Platform V2" and D2.5 – "Reference Architecture and Integration of AI4Gov Platform V3".

5 Infrastructure and Integration

In the context of the AI4Gov project, UBI partner will provide the needed computing and storage capabilities to set-up the cloud-based infrastructure to serve all the needed technologies and tools to be deployed and managed by the project's partners. Overall, this cloud infrastructure aims to help AI researchers, data analysts, policy makers and different stakeholders, to analyse a wide plethora of datasets from different data sources and facilitate the development of trusted, unbiased, and citizen-centric policies.

5.1 Infrastructure and Deployment

The provided infrastructure will help the project's partners to run computation – or data-intensive tasks and host online services in virtual machines or using virtualized containers on the provisioned IT resources. To facilitate the cloud provisioning of resources for the project and to collect all the technical needs, UPRC, as the project's technical coordinator, has invited project members to report technical needs in a shared document. A document listing partners' technical needs, was introduced during one of the AI4Gov consortium meetings organised. Figure 7 depicts a snapshot of this live document.

	Name	Task	Contributor	Type	Notes	Group	VCores [#]	RAM [GB]	Storage [GB]	GP U
OK	Blockchain Infrastructure	T3.1	UBI	Component	Will act as the ingestion mediate of the project and authentication mechanism	Infrastructure	8	24,0 GB	80,0 GB	0
OK	Policy Recommendation Toolkit	T3.3	UBI	Component		Final User Applications	2	6,0 GB	20,0 GB	0
OK	Unbiased Models	T4.1	JSI	Microservices		Analysis	4	20,0 GB	10,0 GB	1
OK	Bias Detection Toolkit	T4.1	JSI	Component		Analysis				
OK	XAI algorithms/Models	T4.2	IBM	Microservices		Analysis	2	16 GB	2 GB	0
OK	SAX Models	T4.2	IBM	Microservices		Analysis				
OK	XAI Library	T4.2	IBM	Component		Analysis				
OK	Policy-oriented analytics and AI Algorithms (NLP, Time series)	T4.3	UPRC	Microservices		Analysis	4	32,0 GB	10,0 GB	1
OK	Visualization Workbench	T4.4	UPRC	Component		Final User Applications	4	16,0 GB	1,0 GB	1
OK	Visual Analytics	T4.4	UPRC	Microservices		Analysis				
OK	Citizen-centric QA and chatbot applications	T4.3	UPRC	Component		Final User Applications	4	16,0 GB	10,0 GB	0
OK	AI4Gov Scalable DB	T2.4	MAG (???)		will provide REST APIs to the GUIs	Storage	2	8,0 GB	20,0 GB	0
OK	Kubernetes	T2.4	MAG		Requirements only for Master node	Infrastructure	4	16,0 GB	40,0 GB	0
OK	Kafka		MAG/UPRC			Infrastructure	2	4,0 GB	0,0 GB	0
	Total						**36**	**142,0 GB**	**191,0 GB**	**3**
	Max						**8**	**32,0 GB**	**80,0 GB**	**1**

Figure 7: Infrastructure Resources Needs

The envisaged infrastructure is provided to the technical partners as a testing and validation environment, accessible remotely and in which every component leader is responsible to test the component and the integration with the other software artefacts developed within the platform.

The overall design of the has resulted in the below infrastructure and integration schema, Figure 8, that is further analysed in later subsections.

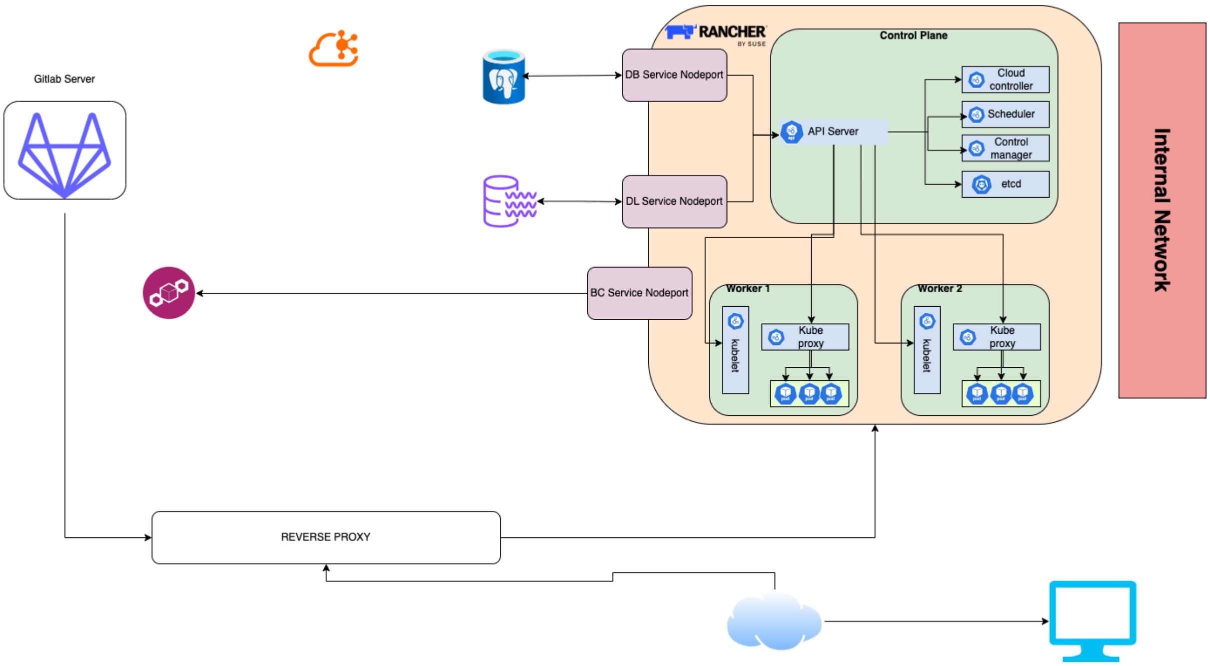

Figure 8: AI4Gov's Infrastructure Schema

Moreover, some indicative screenshots of the cluster follow in the below figures. More specifically, Figure 9 demonstrates the main dashboard for the overall monitoring of the cluster wellness and resources usage, while Figure 10 showcases the three nodes running in the cluster.

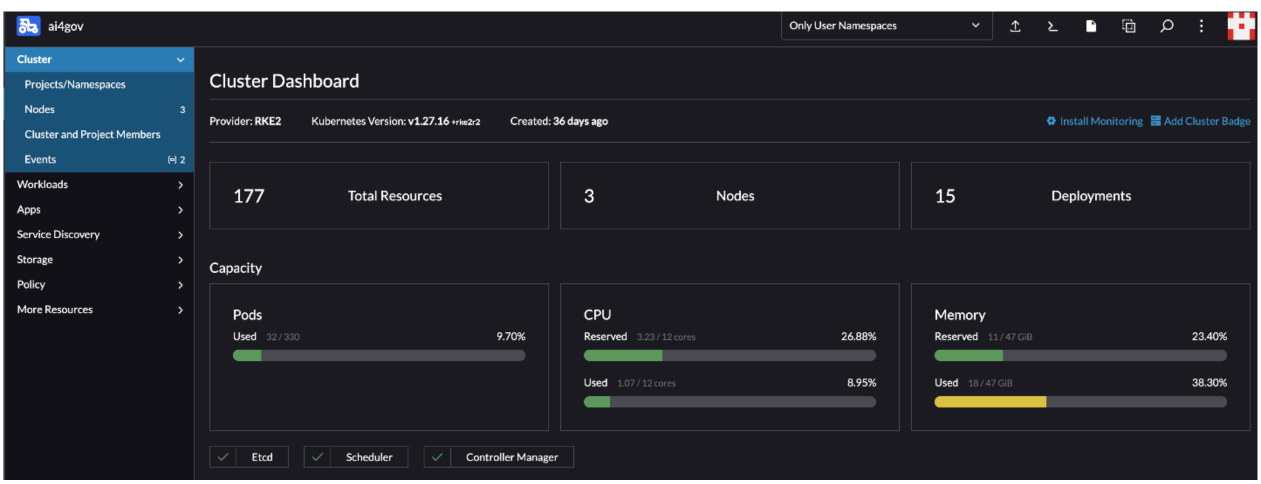

Figure 9: Cluster monitoring dashboard

Figure 10: Cluster nodes

The status of the AI4Gov infrastructure will be finally reported in the next and final version of this series of deliverables, i.e., in D2.5 – "Reference Architecture and Integration of AI4Gov Platform V3", which will be delivered in M36. The total amount of the final resources provisioned by the UBI will also be presented in this last version of deliverables for supporting.

Finally, it should be noted that at this phase of the project, it has been decided that the basic design principle will be service choreography (Leite, 2013) that allows for loosely-coupled microservices that can communicate with each other according to the data pipeline in which they are involved.

5.2 Integration Approach

Furthermore, the project's Consortium has identified a list of needed tools that must be provided to all partners in terms of the overall proposed architecture and the integration and deployment/management of their solutions/components. On top of this, the CI/CD approach represented below should be followed during the whole lifecycle of AI4Gov project. Figure 11 illustrates the CI/CD approach and the corresponding technologies/tools that have been identified.

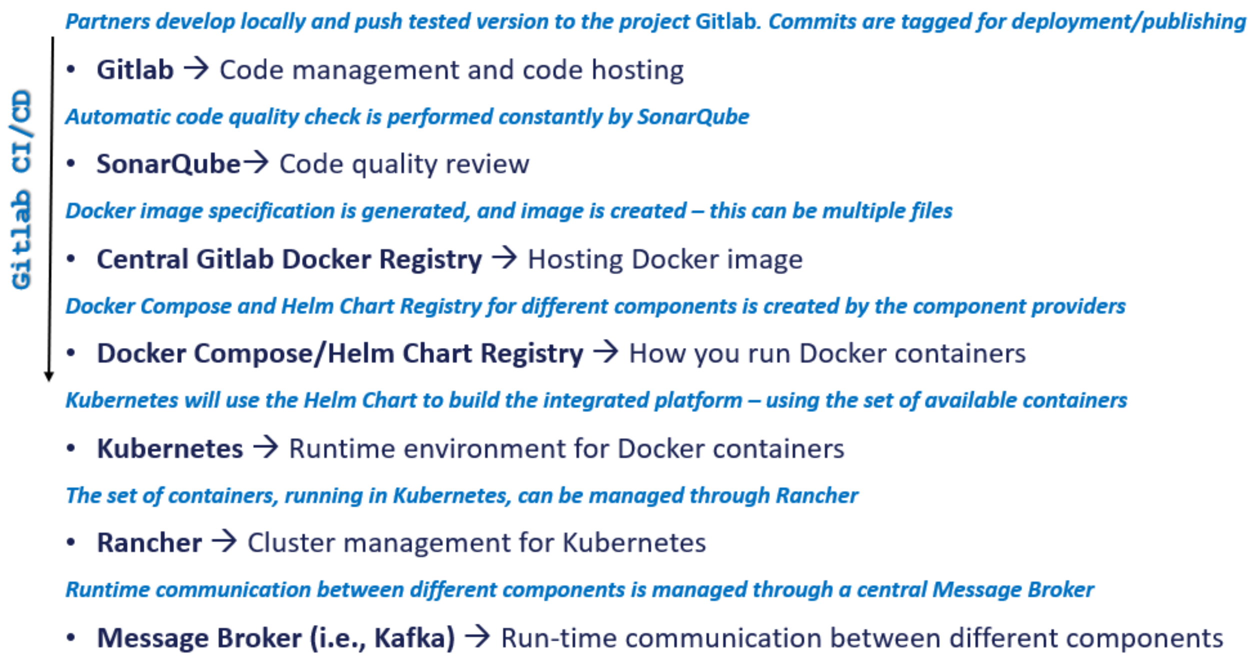

Figure 11: Identified Tools

At first, for enhanced collaboration between all project partners, the GitLab platform has already been installed. Already in use, the platform is accessible via the https://gitlab.ai4gov-project.eu/. Gitlab[3] is an all-encompassing, web-based DevOps platform that revolutionises the software development lifecycle. By utilising it, the AI4Gov team can effortlessly manage the entire workflow, from version control and source code management to continuous integration, delivery, and deployment. Additionally, it offers an integrated environment that combines collaboration, automation, and monitoring, enabling AI4Gov partners to work seamlessly together and streamline development processes, by also utilising the Central Gitlab Docker Registry. The Consortium agreed on the containerisation of its components and micro-services to encapsulate developed applications and their dependencies and to ensure consistent behaviour across different environments.

Available for all the components, but specifically for the communication and data exchange between the main analytical components, the Apache Kafka will be utilised (Garg, 2013). Apache Kafka is a message broker and stream processor which allows to publish, subscribe, archive and process streams of records in real time. It is designed to manage data streams from multiple sources by distributing them to multiple consumers.

Apart from the decision to use Kafka as the platform's message broker mechanism, regarding DevOps, the AI4Gov consortium agreed with the adoption of Kubernetes (Luksa, 2017). Kubernetes (K8s) is an open-source platform that automates Linux container operations. The integration between K8s and Kafka results in the simplification of the deployment of Kafka brokers as containerised pods, as each Kafka broker can run as a separate pod, ensuring

[3] https://about.gitlab.com/

scalability, fault tolerance and availability of the overall approach. On top of this, microservices deployed on top of K8s can easily consume and produce data to Kafka topics, allowing for real-time data processing and analysis of the processed data in the context of the project. Finally, K8s eliminates many of the manual processes involved in deploying and scaling containerised applications and allows the management of host clusters to run containers easily and efficiently. For enhanced management of the K8s cluster, the Rancher tool will be utilised.

5.2.1 Data Lake

The project Interim Repository functions as presented in the previous version of this series of deliverables (i.e., D2.3 – “Reference Architecture and Integration of AI4Gov Platform V1”) are foreseen to act as a temporary data storage layer. However, they are now conceptualized as a Data Lake architecture that leverages MinIO for object storage and PostgreSQL for structured data management. This Data Lake approach enables seamless integration of diverse datasets from multiple use cases while ensuring that data ingestion processes remain flexible and scalable. Datasets stored within the Data Lake are subject to legal and ethical audits, with their utilization governed by the project’s blockchain infrastructure and associated smart contracts. These mechanisms ensure robust access control, transparency, and compliance with data governance policies. The Data Lake also acts as a preparatory staging area for data ingestion into the AI4Gov environment, promoting collaboration among stakeholders through a shared, centralized repository that supports both structured and unstructured data.

The integration of the Data Lake with blockchain services further strengthens data governance. Through smart contracts, data owners can define granular access controls and permissions, ensuring only authorized users can access, modify, or use specific datasets. These permissions are automatically enforced by the blockchain, creating a transparent and decentralized framework for data management. As part of the Data Lake, MinIO supports the scalable storage of unstructured data such as logs, images, and documents, while PostgreSQL handles relational datasets requiring sophisticated query capabilities. Blockchain integration also introduces an immutable ledger for data provenance, recording every change or update made to a dataset with timestamps. This traceability fosters trust among stakeholders by enabling users to verify the origin and history of every data point.

The Data Lake supports the AI4Gov platform in meeting its compliance and governance objectives. The registration and authentication processes ensure that only verified users can access the repository, while data screening mechanisms address legal and ethical considerations prior to data ingestion. The repository’s architecture facilitates interoperability between disparate datasets and systems, promoting seamless integration through shared data governance frameworks enabled by blockchain. Additionally, this centralized environment operates under the strict oversight of the project’s DGF and HRF, ensuring that all data management practices align with the project's ethical and operational guidelines.

5.2.2 Kafka Message Bus

This is the connector to the Apache Kafka, which is the world's dominant event streaming platform. It allows data being pushed to a Kafka topic to be transparently stored into the big data platform in a predefined data table, using this connector. Apache Kafka is an important platform element of the overall AI4Gov integrated solution. The data will be exchanged using Kafka queues. This means that after the initial capture of the raw data from the Data Normalization and the blockchain, the data items will be placed inside a specific Kafka topic. Then, each of the data functions involved in the data pipeline that needs to be established will read data from one Kafka topic and will place the output into another Kafka topic, so that the next involved function can retrieve it. At the end of the data pipeline, a database connector will consume data from a specific data topic in a similar way and will connect to the Data Storage to persistently store the data. At this phase of the project, it has not been decided yet whether we will make use of static microservices that listen to pre-defined Kafka topics, or we will decide in favour of dynamic deployments of such data pipelines, probably taking advantage of a serverless platform provided under the scope of T2.4 - "Integration of AI4Gov Platform and Tools". More details about the deployment aspects of the data capture and ingestion pipelines will be provided in the next version of this series of deliverables.

5.2.3 Container Orchestration via K8s

Container orchestration via K8s represents a novel approach to how modern applications are managed and deployed. K8s is an open-source container orchestration platform that simplifies the complexities of deploying and scaling containerised applications. At its core, K8s automates container management tasks, such as load balancing, scaling, and failover, making it a vital tool for organisations striving to streamline their DevOps processes. It provides a declarative approach, allowing users to define their desired state, and takes care of ensuring that the system matches that desired state, all while maintaining high availability and resilience.

Kubernetes brings several transformative benefits to container orchestration. First and foremost, it allows for efficient scaling, enabling applications to automatically adapt to fluctuating workloads. Moreover, it provides robust tools for service discovery and load balancing, simplifying the process of managing containerized services across clusters of machines. Kubernetes also offers a rich ecosystem of extensions and integrations, making it highly customizable and adaptable to various use cases. As a result, it has become the de facto standard for container orchestration in the cloud-native landscape, empowering organisations to harness the full potential of containers and microservices architecture while improving reliability, scalability, and agility.

5.2.4 Rancher centralized cluster management

As presented in the previous subsection, the Rancher tool is utilized to act as immediate communication management mean for all the components of the cluster. It offers a centralized and robust platform for managing Kubernetes clusters, significantly reducing the complexity of handling multiple clusters or nodes. The incorporation of Rancher into the overall architecture allows to streamline administrative tasks such as provisioning, scaling, and monitoring. These tasks, which can become difficult to be managed in distributed systems, are now consolidated into a single, user-friendly interface. This centralization simplifies the operational overhead and ensures that cluster management is efficient, reliable, and scalable, providing a great alignment of the infrastructure with the guidelines derived from the HELIX and HRF frameworks and empowering AI4Gov partners to focus on research and innovation activities rather than infrastructure handling issues.

One of Rancher's key features is its robust implementation of Role-Based Access Control (RBAC) that enables fine-grained user access permissions management across different namespaces and projects. This ensures secure and efficient management of resources while supporting multi-tenant use cases. On top of this, Rancher's seamless integration with CI/CD pipelines further enhances productivity by enabling streamlined workload deployment. Its built-in catalogue of Helm charts accelerates application deployment, reducing setup times and simplifying the installation and configuration of project's components.

In addition to its operational benefits, Rancher enhances the interoperability of Kubernetes clusters, supporting hybrid environments and multi-cloud strategies. It allows consistent management across on-premises, cloud, and edge Kubernetes clusters. Finally, one of its key functionalities that is leveraged in the context of the project is its built-in monitoring and alerting system, powered by the utilization of Prometheus and Grafana. To this end, Rancher delivers real-time insights into cluster performance and workload health, eliminating the need for third-party tools and ensuring robust, proactive cluster management and monitoring.

To ensure the proper utilization of this tool, the RKE2 (Rancher Kubernetes Engine) was chosen as the Kubernetes distribution for this architecture due to its lightweight, production-ready features. The key benefits from this decision are that is provides enhanced security, simplicity, optimized performance and seamless Rancher integration. More specifically, RKE2 is CIS-compliant by default, ensuring a secure setup for Kubernetes clusters without additional configuration, while due to its minimal dependencies and modular design, it simplifies the setup and maintenance of the Kubernetes environment. Moreover**,** its lightweight design reduces resource overhead, making it an ideal choice for clusters with diverse workloads. Finally, RKE2 works natively with Rancher, ensuring compatibility and unlocking advanced management features.

5.3 Overview of external and internal services

The architecture integrates several critical components outside the Kubernetes cluster to support the system's functionality and scalability. More specifically, as presented above and in the context of the Data Lake implementation, PostgreSQL serves as a secure and robust relational database for storing application data, ensuring reliability in data management. This is complemented by the utilization of MinIO that provides efficient object storage for managing unstructured data and serves as a backend for data-intensive applications. For seamless asynchronous communication between services, Kafka is employed as a scalable messaging system, acting as a mediator between MinIO, analytical models, PostgreSQL, and external services. Moreover, to ensure data and policies integrity and traceability throughout the whole lifecycle of the project, a blockchain infrastructure is included for securely storing results and reports generated by the analytics application. This integration of different tools ensures a well-established, scalable, and secure architecture to support the platform's diverse needs.

What is more, within the Kubernetes cluster several critical components provide the AI4Gov's operations addressing the piloting needs. The Visualization Workbench serves as the front-end interface, enabling users to interact with the system's data and insights. It acts as the main entry point of the project and they key component that is further exposed for exploitation and utilization from the end users. In parallel, at the heart of data processing and analytical capabilities lie the "Policy-Oriented AI and NLP algorithms" and the "Adaptive Analytics Framework" components (further analyzed in D4.2 – "Trustworthy, Explainable, and Unbiased AI V2") that host the AI-powered models and even the Large Language Model (LLM) capabilities and execute the primary data analysis tasks. Finally, to ensure seamless integration with external blockchain infrastructure, the Blockchain Adapter acts as middleware, securely transmitting results and reports from the analytics application to the blockchain system. All these applications are exposed via NodePort services, providing external accessibility while maintaining controlled communication pathways.

To meet the demands of divergent workloads, all deployed applications within the Kubernetes cluster are configured with dynamic auto-scaling capabilities. This ensures efficient resource utilization and maintains consistent performance, even under varying traffic conditions. The integration of external infrastructure with the Kubernetes cluster creates a cohesive system capable of delivering high performance, scalability, and reliability while supporting advanced features such as LLM-based analytics, blockchain security, and real-time data visualization.

Moreover, an NGINX reverse proxy is also deployed in front of the Kubernetes cluster to efficiently manage incoming requests. The proxy is configured to route requests to the corresponding services inside the cluster, such as the Visualization Workbench, Policy-Oriented AI and NLP algorithms component, or Blockchain Adapter, as well as to direct traffic to external interfaces like the MinIO web interface when the latter is required. This setup simplifies request handling while maintaining a secure and organized entry point for all services.

5.4 Continuous Integration and Deployment (CI/CD)

To automate the integration and deployment processes, Fleet has been implemented as the CI/CD solution of the project. The integration of Fleet with GitLab and Rancher ensures that the overall architecture and integration approach offer:

- Seamless Integration: Code changes in GitLab repositories are automatically synchronized with the Rancher-managed Kubernetes cluster.
- Automated Deployments: Fleet pipelines enable automatic application deployments, ensuring faster and more reliable updates.
- Version Control: Every deployment is tied to a specific version, facilitating rollback and debugging in case of issues.

This CI/CD pipeline streamlines the software development lifecycle and reduces manual intervention, making the system more robust and efficient.

5.5 Alignment of the integration approach with the core principles of the Helix methodology

The overall system's design reflects a commitment to the core principles of the Helix methodology, as presented in Section 3, emphasizing modularity, scalability, resilience, and flexibility in integration. The adoption of a modular architecture allows the system to separate external components such as PostgreSQL, Kafka, and MinIO from internal applications like the Visualization Workbench, VUF, and Blockchain Adapter. This modular approach ensures that individual components can be independently scaled, updated, or replaced without disrupting the overall system. Such decoupling aligns with the Helix principle of adaptability, promoting seamless evolution as requirements change or technologies advance.

Scalability is a cornerstone of the system, supported by the architecture's ability to handle different workloads dynamically. Within the Kubernetes cluster, auto-scaling capabilities ensure efficient resource allocation based on traffic demands, while external components such as Kafka and MinIO leverage horizontal scaling to accommodate increasing loads. Resilience is equally prioritized, with RKE2 and Kubernetes providing high availability through features like pod replication, self-healing mechanisms, and fault-tolerant control planes. The integration of Kafka as a messaging intermediary further enhances the system's robustness, ensuring reliable message delivery even in the face of failures. These capabilities align with Helix's focus on building systems that are scalable and resilient to disruptions.

Furthermore, the system's architecture incorporates the flexibility aspect highlighted by the Helix methodology, by accommodating diverse workloads and deployment environments. NodePort services enable external accessibility for applications while maintaining compatibility with varied deployment strategies. In that direction, the inclusion of a reverse proxy enhances routing flexibility, facilitating seamless integration with external interfaces and ensuring efficient traffic management. To this end, the overall design and strategic decisions on the integration and architecture approach of the project demonstrate the project's ability to follow a pilot-agnostic approach and adapt to newly added and evolving use cases and integrate smoothly with external systems, reinforcing the Helix principle of creating flexible, wide adaptable, and future-ready solutions.

6 Conclusion

This document provides the updated view of the conceptual overall architecture and integration approach of the AI4Gov platform. It highlights the main components, their interconnections, and key capabilities, and offers details regarding the infrastructure and the approach that will be followed for the deployment of the platform. Furthermore, the technical partners of the consortium provided the updated sets of technical requirements as they were foreseen and updated at this phase of the project. Additionally, the document outlines a state-of-the-art analysis of the base technology sectors in which the AI4Gov project is involved in, or could possibly be exploited, along with a list of baseline technological tools and solutions that are incorporated into the overall platform. At this phase of the project, the outcomes of this deliverable provide a valuable insight for the progress of the tasks that are related to the design of the overall architecture of the platform and its integration.

Moreover, this deliverable seeks to encapsulate the overall strategy towards the alignment of the project's DGF and HRF with its architecture and integration strategy to further demonstrate the resilience, generalization, as well as compliant of its solutions with ethical and regulatory mandates.

All the details, and the internal design of each component, are demanded to the related tasks and deliverables that will follow during the lifecycle of the project. However, based on this document a higher-level description and presentation of all incorporated and integrated components is provided, by also detailing of how the AI4Gov solution will be provided to the end-users.

7 References

Ağbulut, Ü. (2022). Forecasting of transportation-related energy demand and CO2 emissions in Turkey with different machine learning algorithms. *Sustainable Production and Consumption*, (pp. 141-157).

Alexopoulos, C. L. (2019). How machine learning is changing e-government. . *12th international conference on theory and practice of electronic governance* , (pp. 354-363).

Amit, G. (2022). Model informed LIME Extension for Business Process Explainability.

Amit, G. (2022). Model informed LIME Extension for Business Process Explainability. . *PMAI@IJCAI'22. CEUR.*

Amit, G. (2022). Model informed LIME Extension for Business Process Explainability. . *PMAI@IJCAI'22. CEUR.*

Amit, G. (2022). Situation-Aware eXplainability for Business Processes Enabled by Complex Events., (pp. International Conference on Business Process Management,).

Barcellos-Paula, L., De la Vega, I., & Gil-Lafuente, A. M. (2021). The Quintuple Helix of innovation model and the SDGs: Latin-American countries' case and its forgotten effects. *Mathematics, 9*(4), 416.

Berg, V. d. (2022). *Artificial Intelligence and the Future of Public Policy.* Retrieved from https://ec.europa.eu/jrc/communities/sites/jrccties/files/06_berg.pdf

Bhardwaj, A. D. (2022). Smart IoT and machine learning-based framework for water quality assessment and device component monitoring. Environmental Science and Pollution Research.

Braud, A. F. (2021). The road to European digital sovereignty with Gaia-X and IDSA. *IEEE network, 35*(2), pp. 4-5.

Buijsman, S. (2024). Navigating fairness measures and trade-offs. *AI and Ethics, 4*(4), 1323-1334.

Carayannis, E. G. (2009). 'Mode 3'and'Quadruple Helix': toward a 21st century fractal innovation ecosystem. *International journal of technology management, 46(3-4)*, 201-234.

Carayannis, E. G. (2012). The Quintuple Helix innovation model: global warming as a challenge and driver for innovation. *Journal of innovation and entrepreneurship*, 1-12.

Carayannis, E. G. (2022). Helix trilogy: The triple, quadruple, and quintuple innovation helices from a theory, policy, and practice set of perspectives. *Journal of the Knowledge Economy, 13(3)*, 2272-2301.

ChatGPT. (n.d.). Retrieved from https://chat.openai.com

Commission, E. (2020). *Proposal for a Regulation of the European Parliament and of the Council on European data governance (Data Governance Act) — COM/2020/767 final.* Brussels, Belgium: European Commission.

Dumas, M. (2023). AI-augmented business process management systems: A research manifesto. *ACM Trans. Manage. Inf. Syst. 14(1)*.

Duong, M. K., & Conrad, S. (2023). Towards Fairness and Privacy: A Novel Data Pre-processing Optimization Framework for Non-binary Protected Attributes. *Australasian Conference on Data Science and Machine Learning* (pp. 105-120). Singapore: Sp.

Enock, F. E., Hewstone, M. R., Lockwood, P. L., & Sui, J. (2020). Overlap in processing advantages for minimal ingroups and the self. *Scientific Reports, 10*(1), 18933.

Etzkowitz H, L. L. (1995). The Triple Helix—University-industry government relations: A laboratory for knowledge based economic development. *EASST Review*, 14-19.

Ferrara, E. (2023). Fairness and bias in artificial intelligence: A brief survey of sources, impacts, and mitigation strategies. *Sci, 6*(1), 3.

Fournier, F. (2023). The WHY in Business Processes: Discovery of Causal Execution Dependencies, https://arxiv.org/abs/2310.14975,.

Garg, N. (2013). *Apache kafka.* Birmingham, UK: Packt Publishing.

Garzon-Orjuela, N., Parveen, S., Amin, D., Vornhagen, H., Blake, C., & Vellinga, A. (2023). The effectiveness of interactive dashboards to Optimise Antibiotic Prescribing in Primary Care: a systematic review. *Antibiotics, 12*(1), 136.

Geertsema, P. B. (2023). ChatGPT and Large Language Models: What are the Implications for Policy Makers? *SSRN 4424048*.

Gino Sophia, S. G. (2020). Water management using genetic algorithm-based machine learning. Soft computing.

González-Sendino, R. S. (2024). Mitigating bias in artificial intelligence: Fair data generation via causal models for transparent and explainable decision-making. *Future Generation Computer Systems, 155*, 384-401.

Guo, D., Li, Z., Zhao, H., Zhou, M., & Zha, H. (2022). Learning to re-weight examples with optimal transport for imbalanced classification. *Advances in Neural Information Processing Systems, 35*, 25517-25530.

Hettinga, S. V. (2023). Large scale energy labelling with models: The EU TABULA model versus machine learning with open data.

Kyriazis, D. B. (2020). Policycloud: analytics as a service facilitating efficient data-driven public policy management. *Artificial Intelligence Applications and Innovations: 16th IFIP WG 12.5 International Conference, AIAI 2020, Neos Marmaras, Greece, June 5–7, 2020, Proceedings, Part I 16*, pp. 141-150.

Leite, L. A. (2013). A systematic literature review of service choreography adaptation. *Service Oriented Computing and Applications*, pp. 199-216.

Lerner, R. M. (2014). *At the forge: PostgreSQL, the NoSQL database* (Vol. 247). Linux Journal.

Lewicki, K., Lee, M. S. A., Cobbe, J., & Singh, J. (2023). Out of Context: Investigating the Bias and Fairness Concerns of “Artificial Intelligence as a Service”. *2023 CHI Conference on Human Factors in Computing Systems*, (pp. 1-17).

Li, X. L. (2021). Machine learning in internet search query selection for tourism forecasting. *Journal of Travel Research*, 1213-1231.

Luksa, M. (2017). *Kubernetes in action.* Simon and Schuster.

Mavrogiorgou, A. K. (2021). beHEALTHIER: A microservices platform for analyzing and exploiting healthcare data. *2021 IEEE 34th International Symposium on Computer-Based Medical Systems (CBMS)* (pp. 283-288). IEEE.

Naef, S. W. (2024). Blockchain and network governance: Learning from applications in the supply chain sector. *Production Planning & Control, 25*(9), 932-946.

Petersen, F., Mukherjee, D., Sun, Y., & Yurochkin, M. (2021). Post-processing for individual fairness. *Advances in Neural Information Processing Systems, 34*, 25944-25955.

Puh, K. &. (2023). Predicting sentiment and rating of tourist reviews using machine learning. *Journal of Hospitality and Tourism Insights*, 1188-1204.

Rachakatla, S. K., Ravichandran, P., & Machireddy, J. R. (2023). AI-Driven Business Analytics: Leveraging Deep Learning and Big Data for Predictive Insights. *Journal of Deep Learning in Genomic Data Analysis, 3*(2), 1-22.

Rahman, M. M. (2021). Machine learning on the COVID-19 pandemic, human mobility and air quality: A review. Ieee Access.

Ray, P. P. (2023). *ChatGPT: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope.* Internet of Things and Cyber-Physical Systems.

Shafiq, M. T. (2020). Data mining and machine learning methods for sustainable smart cities traffic classification: A survey. . Sustainable Cities and Society.

Tan, E. L. (2023). Verification of Education Credentials on European Blockchain Services Infrastructure (EBSI): Action Research in a Cross-Border Use Case between Belgium and Italy. *Big Data and Cognitive Computing, 7*(2), p. 79.

Xie, G. Q. (2021). Forecasting Chinese cruise tourism demand with big data: An optimized machine learning approach. . Tourism Management.

Xu, H., Liu, X., Li, Y., Jain, A., & Tang, J. (2024). To be robust or to be fair: Towards fairness in adversarial training. In International conference on machine learning. *PMLR*, 11492-11501.

APPENDIX A – Architecture and Infrastructure Discussions

Template for the collection of Technical Requirements

The screenshots below (Figures 12-14) represent the template used for the collection of the Technical Requirements from AI4Gov partners.

1.1 Title of your component

1.1.1 Goals and Objectives

"Please add a small introductory text here regarding the goals and objectives of this component"

1.1.2 State of the Art

"Please fill in with the state of the art approach regarding your component"

1.1.3 Background technology

"Please fill in with text here regarding the already existed baseline technology of this component and how you plan to further develop them in order to fulfil the requirements of the AI4Gov platform. Please for this description use the following table"

Technology Name	Technology Description	Advancements / Usage

1.1.4 Component to User Requirements

"Please fill the following tables to establish a link between this component and the relevant user requirement(s) i.e., which user requirement(s) will be addressed through this component – use 1 table for each user requirement".

Title	Description
ID	*U-REQ-x (unique id in the format: "U-REQ-<TASK NUMBER>-NUMBER")*
Source User Requirement	*Please describe the user's need which this component satisfies (see D6.1 for more info)*
Use case quote	*This field is a segment from the user-scenario (e.g. relevant sentence from the Process Dialogue part of the user scenario) that can be related to this component (see D6.1 for more info)*
Generic/Specific	*Please fill choose Generic, if the user*

Figure 12: Template for the Collection of Technical Requirements (1/3)

	requirement is generic or Specific if this user requirement is pilot specific
Task/Component	*Please fill the name of relevant task and the relevant component name*
Lead partner	*Partner who lead the component*
Notes	*Additional notes*

Example:

Title	**Description**
ID	*U-REQ-T3.1-01*
Source User Requirement	*Delivery of smart contracts utilized to facilitate actors' contributions to the AI4Gov distributed ledger*
Use case quote	*Drinking water US#2 - As a member of the local administration I want to have a clear picture of the main areas for improvement within the drinking water treatment system*
Generic/Specific	*Generic*
Task/Component	*T3.1 - Decentralized Data Provenance and Reliability*
Lead partner	*UBI*
Notes	*Public actors and policy makers can share information and state as part of AI4Gov through secure state synchronization techniques*

1.1.5 Component to Technical Requirements

"Please fill the following tables with information regarding the component's requirements. Add as many tables as the requirements that are being imposed by the specific software component".

Section	**Description**
Technical requirement code	*T-REQ-Tx.x-xx*
Type	*(*see relevant table of the Technical*

Figure 13:Template for the Collection of Technical Requirements (2/3)

	Requirements Template document)
Short name	*Meaningful and not too long*
Functionality ID	*Respective user requirement that this requirement will address*
Description & quantification	*Description of the requirement and if possible quantification (especially in the case of non-functional requirements such as the ones related to performance)*
Additional information (source, characteristics, short description of the process)	
Priority	*This allows us to identify the priority of the requirements, which can be updated in the different iterations:* • *MAN: Mandatory requirement.* • *DES: Desirable requirement.* • *OPT: Optional requirement.* • *ENH: Possible future enhancement*
Reference Scenarios	*Pilot scenario in which this module/component may be utilized*
Success criteria	*This field contains information on how to assess the fulfilment of this requirement.*

Figure 14: Template for the Collection of Technical Requirements (3/3)