H2020-SC6-GOVERNANCE-2018-2019-2020

DT-GOVERNANCE-05-2018-2019-2020

D3.2: Design of the ACROSS Data Governance framework for data sovereignty – Final

Project Reference No	959157 — ACROSS — H2020-SC6-GOVERNANCE-2018-2019-2020
Deliverable	D3.2: Design of the ACROSS Data Governance framework for data sovereignty – Final
Work package	WP3: ACROSS Data Governance framework
Nature	Report
Dissemination Level	Public
Date	29/11/2022
Status	V1.0
Editor(s)	Valentín Sánchez (TEC)
Contributor(s)	Urtza Iturraspe (TEC), Enrique Areizaga, Idoia Murua
Reviewer(s)	Vincenzo Savarino (ENG), Jolien Clemens and Hans Graux (TLX)
Document description	It includes 1) the requirements, functional specification, the technical architecture, the design of the modules and a description its APIs; 2) the design of a generic data governance mechanism, such as a data governance data model for handling data access rights; 3) mock-ups of the user interface and 4) the relevant baseline technologies that will be used for the implementation of the data governance framework.

About

The project is co-funded by the European Commission's Horizon 2020 research and innovation framework programme. Spanning through three years, ACROSS consists of a consortium of 10 partners from 7 countries: Athens Technology Center (coordinator), Tecnalia, Dataport, Engineering, Fraunhofer, GRNET, TimeLex, The Lisbon Council, Waag and VARAM.

DISCLAIMER

Document Revision History

Version	Date	Modifications Introduced	
		Modification Reason	Modified by
V0.1	22/11/2022	Table of contents and initial version	Valentin Sánchez
V0.2	23/11/2022	First internal revision	Urtza Iturraspe, Idoia Murua, Enrique Areizaga
V0.3	25/11/2022	Second Internal review	Vincenzo Savarino
V0.4	29/11/2022	Third Internal review	Jolien Clemens and Hans Graux
V1.0	29/11/2022	Final version	Valentín Sánchez

Executive Summary

The main objective of the ACROSS project is to provide the means (tools, methods and techniques) to enable user-centric design and implementation of interoperable cross-border (digital) public services compliant with the current European regulations (e.g. the Single Digital Gateway (SDG) and Once-Only principle (OOP), European Interoperability Framework (EIF)) where the private sector can also interconnect their services **while ensuring the data sovereignty of the citizens, who can set the privacy level that will allow the public and private sector to access to their data based on their requirements**.

In order to ensure the protection of personal data (and documents) and its compliance with GDPR and other relevant regulations, especially when shared between organizations, ACROSS will design and implement **a data governance framework** where data subjects can control the use of their personal data empowering them**.**

The **data governance framework will** allow users to:

1) monitor which data are available to whom, and how they are used or how it has been accessed,
2) control: add, delete or change data, provide or block access to public bodies (where permitted or enforced by the law), businesses or data brokers, giving individuals the power to determine how their data can be used.

This report provides the final design of the ACROSS Data Governance framework for data sovereignty including the technical architecture, modules, APIs and graphical user interface.

The initial design (D3.1) was based on the data governance, security and privacy requirements from the use cases, considering both the technical and operational perspectives (WP6), the final user expectations regarding data privacy (WP2) and the ACROSS platform integration strategy (WP4 and WP5).

The final design is the update of the initial one based on the results of the usability test findings for the first version of the platform (D3.3). **User tests** via structured interviews and a **co-creation workshop** have helped in collecting valuable feedback about the Alpha version of the ACROSS platform from real users, as well as stakeholders from pilot countries and EU institutes who are involved in relevant national or European projects. **A set of recommendations for the evolution of ACROSS solution** have been reported here, mostly focus on optimizing already implemented features as well as making previous requirements more specific.

Table of Contents

List of Figures

List of Tables

List of Terms and Abbreviations

Abbreviation	Definition
IDSA	International Data Space Association
CPSV-AP	Core Public Service Vocabulary Application Profile
ABC	Attribute Based Credentials
GDPR	General Data Protection Regulation
DGA	Data Governance Act
PIMS	Personal Information Management System
SDGR	Single Digital Gateway Regulation
OOP	Once-only principle
UJSE	User Journey Service Engine
UI/UX	User Interface / User Experience

1 Introduction

1.1 Purpose and Scope

One of the ACROSS objectives is **to ensure the protection of personal data (**and documents**) and its compliance with GDPR and other applicable regulations, especially when shared between organizations.** This objective will be fulfilled by designing and implementing a private/personal data governance framework where data subjects can control the use of their personal data empowering them.

ACROSS will offer the citizen the possibility of defining which public and private organization will be allowed to *access which data and for what purpose* through the **ACROSS Data Governance Framework.** The main aim is to give the citizen the chance of **governing the access to** their data, profiting from a set of usage policies that implement levels of access and they can be the **sovereign owner** of such data.

The **data governance framework,** that allows users to

1) monitor which data are available to whom, and how they are used or how it has been accessed,
2) to control: add, delete or change data, provide or block access to public bodies (where permitted or enforced by the law) businesses or data brokers, giving individuals the power to determine how their data can be used.

From a technical point of view the Data governance framework includes:

1) A "private/personal data" governance platform including a Personal data management site which provides a user interface to define manage and control the use of personal data. (Data portal)
2) A set of APIs/libraries to interact with the ACROSS platform

The governance framework will be based on existing solutions:

1. **MyData**[1] model for human-centered personal data management and processing
2. Built on experiences around **Attribute-Based Credentials** approaches in the DECODE[2] project,
3. Built on the approach adopted in CaPe solution[3] for personal data management,
4. Include generic ***data usage policies*** when the private data needs to be transferred among several stakeholders (IDSA Data Sovereignty)

[1] https://mydata.org/
[2] http://decodeproject.eu/
[3] https://github.com/OPSILab/Cape

This deliverable includes the final version of the ACROSS personal data governance framework requirements, functional specification, the technical architecture, the design of the modules and a description of its APIs.

The initial design (D3.1) was based on the data governance, security and privacy requirements from the use cases, considering both the technical and operational perspectives (WP6), the final user expectations regarding data privacy (WP2) and the ACROSS platform integration strategy (WP4 and WP5).

The final design is the update of the initial one based on the results of the usability test findings for the first version of the platform (D3.3) that have been gathered in D6.2 Use case evaluation and impact assessment – Initial. **User tests** via structured interviews and a **co-creation workshop** have helped in collecting valuable feedback about the Alpha version of the ACROSS platform from real users, as well as stakeholders from pilot countries and EU institutes who are involved in relevant national or European projects. **A set of recommendations for the evolution of ACROSS solution** have been reported here, mostly focus on optimizing already implemented features as well as making previous requirements more specific.

Furthermore, some new functionalities have been included in the final design.

1. Data usage policies enforcement for Data access control
    a. Data usage policy editor
        i. Create, modify and delete policies associated to services
    b. Rest interface for policy enforcement (Usage control)
2. Rest interface for data consent definition from the User Journey Service Engine (UJSE)
3. My Personal Data view

Still to be done:

- Improve the user experience and usability of the Transparency dashboard
- The module used for logging is internal, not the one defined in the architecture, so both need to be integrated.

1.2 Approach for Work Package and Relation to other Work Packages and Deliverables

The goal of WP3 is to design, implement and deploy a “private/personal data” governance framework that allows the citizens to control how their data are created or used by businesses, governments, or data brokers, giving individuals the power to determine how their data can be used. The governance framework will be based on existing solutions such as MyData model for human-centred personal data

management and processing, CaPe suite for personal data management, and built on experiences around Attribute-Based Credentials approaches in the DECODE project, but it will also include generic data usage policies when the private data needs to be transferred among several stakeholders.

The services from this WP will be integrated into the platform created in WP5 and will demonstrate the functionality of the use cases in WP6.

WP5 aims at providing the architectural and implementation aspects for the delivery of the ACROSS tools taking into account the full range of requirements for such service. The design of the ACROSS platform will drive the design and implementation of the various components produced in the context of WPs **WP3**, WP4 & WP5.

WP2 and WP6 together have defined the so-called user journeys based on the results of several interviews with people from the three pilot countries. The aim of the interview process is to form potential user journeys, building on initial ideas. User journeys can include actions, touch points, emotions, pain points, and phases. This eventually results in concrete (socio-technical) requirements for the ACROSS platform modules. A specific section about data privacy issues has been included in the questionnaire in order to gather further requirements for the Data Governance Framework.

The ACROSS Personal Data Governance Framework is part of the so-called ACROSS platform (defined in WP5), useful in the context of cross-border public/private services for the citizens. However, it has been designed to be an independent framework which can be used also for individuals to manage their personal data according to the GDPR in any other context. Furthermore, it extends the MyData operator concept with Data usage policies enforcement, and data minimization techniques will be also integrated.

The following previous results of the project have been taken into account:

- D3.1 Design of the ACROSS Data Governance framework for data sovereignty – Initial [1]
- D3.3 Implementation of the ACROSS Data Governance framework for data sovereignty – Initial [2]
- D2.4 Report for cross-border service gap analysis – Final [3]
- D6.2 Use case evaluation and impact assessment – Initial [4]

1.3 Methodology and Structure of the Deliverable

This deliverable has been structured in the following sections:

- **Section 2** describes the previous work that has provided inputs to the final design and presents a summary of the new and updated functionalities included.

- **Section 3** is the central chapter of the deliverable, the Data Governance framework final design, which includes the framework architecture, both as an isolated product and integrated with the rest of the ACROSS platform, the detailed description of the modules and their interactions, the detailed description of the modules, APIs, the main data models and the user interface.
- Finally, some conclusions are drawn together with recommendations for future work.

2 Inputs and previous work

This section describes the previous work that has provide inputs to the final design and presents a summary of the new and updated functionalities included.

2.1 Inputs from other Work Packages

2.1.1 Requirements update and status

Next table shows the status of the requirements related to the Data Governance Framework for data sovereignty gathered in D6.2.

Table 1 Data Governance framework related requirements status

No.	Title	Description	Status - Comments	Relevance
Req_2	Data cockpit & interoperability tool	User story: As a user I want to be able control the flow of my data and documents through the platform. I want to be able to consent only to share data which I am comfortable sharing with. Technical components: Transparency dashboard, Digital wallet	**Partially implemented. Improvement of UI pending.**	**Medium**
Req_2.1	Support wallet for storing documents	Description: Place to store and access manually uploaded documents in case users want to reuse them. The wallet stores personal documents, credentials and attributes linked to the users' identity. The users will be able to share the documents with relevant parties on request and use them for authentication.	**Not implemented**	
Req_2.2	Consent panel	Description: Place which stores information on user consents	**Partially implemented.**	

		given regarding their personal data. Represented as list with tick/untick options.	**Improvement of UI pending**	
Req_2.3	Notification panel	Description: User is notified of any additional consents that need to be given (push notifications) to move forward with user journey.	**Partially implemented**	

The main conclusion from this analysis is that the transparency dashboard usability should be improved, specifically the process of accessing the personal data requirements of a service (i.e. the personal data that it must be able to access) and the way in which the consent is defined (granted and revoked). Some of the concepts used in the user interface are not clear for the end user, as for example the difference between consent "disabled" and "withdrawn".

The use cases will recommend some changes that will be evaluated and implemented in the next Data Governance framework implementation deliverables.

Regarding the personal wallet, two approaches have been identified: Define ACROSS as a trusted intermediary between the personal wallet and the services or to define a new channel for interacting with the services, like using a web page or email. The European Digital Identity Wallet[4] will make it possible for EU citizens, residents, and businesses to identify and authenticate themselves or provide confirmation of certain personal information. It will be used for both online and offline public and private services across the EU. ACROSS has been in contact with other initiatives and other projects (mGov4EU, IRMA) to look at other interactions for mobile wallet support.

However, the implementation and deployment of a European Identity Wallet is still under development and is out of the scope in ACROSS.

2.1.2 Pilot user tests and co-creation workshop recommendations

In Deliverable 6.2 usability test findings for the Alpha version of the platform are reported and recommendations for the next versions are proposed.

[4] European Digital Identity | European Commission (europa.eu)

The implementation and evaluation of use cases are essential for testing and validating the functionalities of the ACROSS Platform as well as for identifying the aspects that need to be tackled in order to apply improvements for offering functional, user-friendly and user-centric cross-border digital services.

User tests via structured interviews and the **co-creation workshop** that was held in June 2022 have helped in collecting valuable feedback about the Alpha version of the ACROSS platform from real users, as well as expert stakeholders from pilot countries and EU institutes who are involved in relevant national or European projects.

A set of recommendations for the future development of ACROSS solution, such as features of the user journey services engine and modelling tool, improvements on the UI/UX of the platform and its modules, transparent communication on data security and trustworthiness, mostly focus on optimizing already implemented features as well as making previous requirements more specific. Pilot partners will work in close collaboration with technical partners to give feedback on the progress of the implementation.

In the following table the recommendations for the future development of ACROSS solution mostly focus on optimizing already implemented features as well as making previous requirements more specific. Only the recommendations related to the User Journey Services Engine and the Transparency Dashboard has been included.

Table 2 Recommendations for ACROSS future development

No.	Title	Description/User Story	Relevance
Rec_B	Improvement of the UI/UX of the platform and its modules	User Story: As a user I want to make the least possible clicks for exploring user journeys, initiating one, using services, allowing access to my data, changing settings and accessibility options, finding examples etc. Description: The user interface and the user experience of the platform should be improved in order to take under consideration the difficulties that the initial users found during the user tests. Especially, improvement of the UI of the Transparency Dashboard is needed, Requirements about providing Information pages and widgets to the end users should also be considered. The information should be	High

		provided in such a way that users can easily understand and in a language understood by the user. This requirement is provided under the SDG regulation. Technical Components: Citizen Front End Components	
Rec_E	Concerns of gap analysis (D2.4) about Attribute Based Credentials (ABCs) and digital wallets	User Story: As a citizen I want to maintain the current level of access to my personal data from the service providers, after the adoption of ABCs and Digital Wallets. Description: The following recommendations described in D2.4, should be considered: • Thoughtfully implement certain types of credentials • Implement and enforce guidelines for requesting credentials • Ensure that use of a digital wallet is not required for access to basic, necessary, or public services. Does ACROSS incentivise and encourage the service providers and the wallet provider(s) who we work with to take the above steps? Technical Components: Data Access Control, Identity and Access Management, Transparency Dashboard	Medium
Rec_K	European Commission visual identity	User Story: As a user, I expect a consistent and familiar design in the European Union UI style on all subpages of the ACROSS platform in order to recognize ACROSS as a service of the EU and to trust the website. Description: ACROSS has to look and feel like it's part of the European Commission. Therefore, the UI of all ACROSS platform parts has to be	Medium

		consistent in the EU design in accordance with the guidelines set forth by the European Commission in this regard (while making it clear that, currently, it is only an EU funded research project, but not an official EU page).[5]. Technical Components: Citizen Front End	
Rec_L	Facilitating trust in the process and product	User Story: As a user, I would like to be able to recognize officially verified processes as such (e.g. via a symbol of the corresponding authority), to be sure that when I perform the process in ACROSS, I have done so according to the laws of the respective country and the processes are recognized by the respective authorities. Description: ACROSS has to provide a verification symbol to user journeys which are verified with state authorities and guarantee the minimum of actions to legally move abroad. It has to show disclaimers on actions needed to perform in-person after arrival (where justified by overriding reasons of public security, public health or the fight against fraud and when the objective cannot be fully achieved online, in accordance with applicable EU legislation). It has to include official contacts of each service owner and possibly their social media accounts for ability to reach out for support (contacting a human being, not an AI solution). Technical Components: Citizen Front End, UJMT, UJSE, Service Catalogue	Medium

[5] https://ec.europa.eu/info/sites/default/files/eu-emblem-rules_en.pdf.

Rec_M	Reusability of Components and Technologies	Components developed for ACROSS (like Dashboard, User Journey Engine, eIDAS proxy) should be well documented and ready-to-use by other platforms.	High
Rec_P	Login is still not user-friendly	As a user, I would like to be able to log in easily and without complications, without in-depth technical know-how, in order to experience a frustration-free use of the platform, while maintaining a privacy-secure login solution (e.g. multi-factor authentication where available)..	Medium
Rec_Q	European digital identity wallet integration	As a user, I would like to integrate my certificates directly from the European Digital Wallet on the platform in order to save unnecessary process steps.	Low
Rec_U	Guidance improvements	User story: As a user I want to be able to understand and learn how to use the platform in an easy manner. It would be great to have a short video on how to perform basic functions readily available upon login. Description: ACROSS has to create a video of the most typical use case of the platform. It would include information on how to create a user journey, start performing services and manage data consents. It would serve as the first support point for users before seeking help from virtual assistant and/or service owners. Other possibility would be to make a step-by-step explanation while the user explores the platform for the first time. Technical components: Citizen Front End Components	Medium

2.1.3 WP5 architecture

This deliverable has been aligned with D5.2: System Architecture & Implementation Plan – Final [5], which define the overall architecture of the ACROSS platform in terms of the supported functionalities, the respective processes and the components that realise them.

2.2 Evolution from the initial design of the ACROSS Data Governance framework for data sovereignty (D3.1)

This deliverable is an update of D3.1 Design of the ACROSS Data Governance framework for data sovereignty – Initial. Along with a more detailed information about the modules, APIs and user interface web application, this new version includes some new functionalities.

This section presents the main changes introduced in the Data Governance framework final design.

Two new user interfaces have been included:

- **My Personal Data View**: New dashboard that allows the user to know what services are able to use a specific personal data category. From this new interface, the user is also able to manage (including change or withdraw) the consent.
- **Manage Data usage policies**: A new user interface has been designed to allow the user to define and manage data usage policies associated to services.

Two new APIs:

- **Check Data usage policy**: API to enforce the data usage policies to be called from the UJSE.
- **Define consent**. Rest interface for consent personal data transfer (to be called from the UJSE): The whole set (including optional) or only the required personal data.

3 Data governance Framework final design

3.1 Architecture

ACROSS must offer secure storage for user data and documents, in the case it stores these data. This ensures the Cross borders public services adaptation. It must also provide a Transparency dashboard to the users to control how their (personal) data are used by public administrations, businesses, or data brokers to easily manage and handle sensitive information. The user must be able to define the rules on how data, used in cross border services, must be used (e.g., who can see my data and which parts, prohibit forwarding to 3rd parties and other participants, the purposes for which my data can be used, etc.)

The ACROSS Personal Data Governance Framework is part of the so-called ACROSS platform, useful in the context of cross-border public/private services for the citizens. However, it has been designed to be an independent framework useful also for the individuals to manage their personal data according to the GDPR in any other context. Furthermore, it extends the MyData operator concept with Data usage policies enforcement and data minimization techniques will be also integrated.

3.1.1 Data governance Framework design

The following figure provides the overall view of the main components of the Data Governance Framework seen as an isolated product. In the figure the security related modules have been included as an external security framework that provides identity and access control and logging and audit functionalities. The UJSE has been included as an example of an external application using the framework.

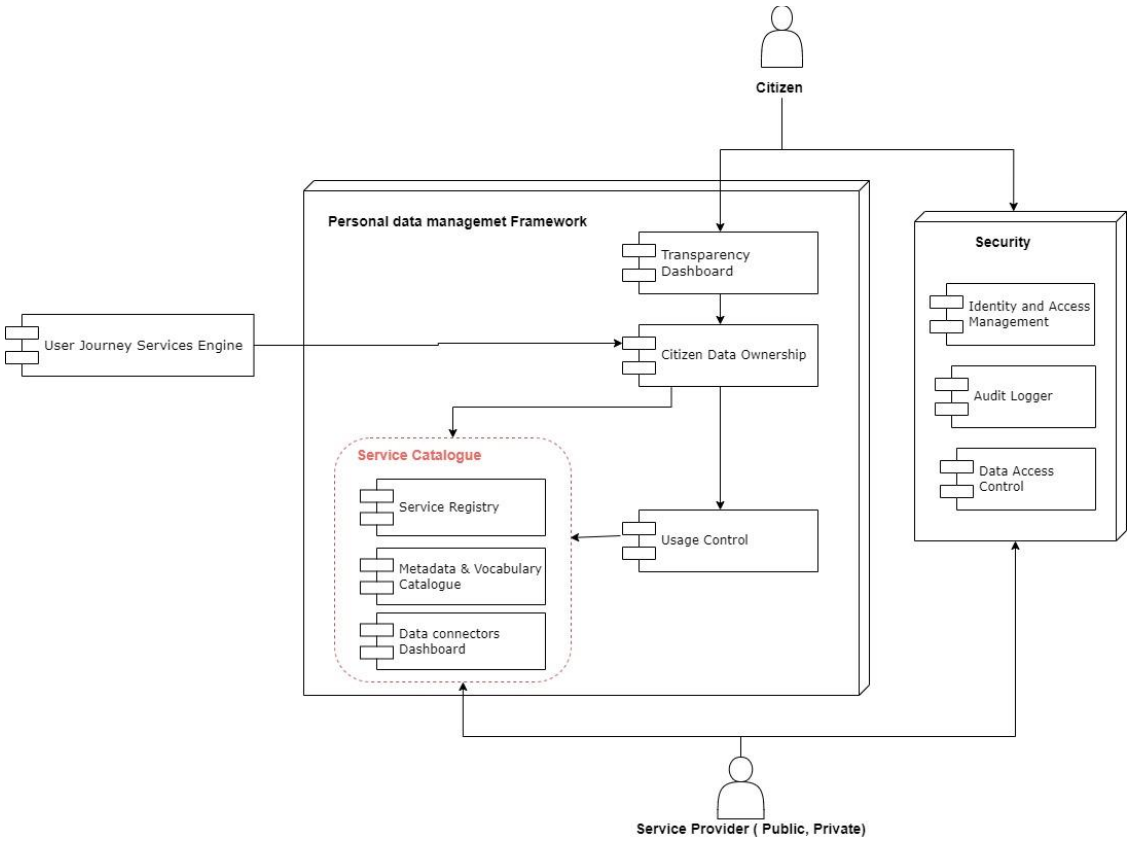


Figure 1 - Component View of Data Governance Framework

The Data Governance Framework will allow citizens / users to select a series of services from the service catalogue and allow the use of their data based on their explicitly, freely given and informed consent. To carry out this transfer of information in a secure way, the Usage Control module will be used, which will allow the usage of data based on previously defined usage policies.

The components in the Security layer will be used by all the components in the Data Governance Framework. This layer provides all the security features needed for a citizen and a service provider to be authenticated and authorized, and for logging all the interactions among all components of the framework.

In the current version of the framework an internal logging module has been implemented. This logging module stores detailed information about all the actions performed by the framework. Only a subset of this information will be sent to the external audit logger included in the ACROSS platform.

The service Catalogue aims at providing all functionality to register, model, map and publish and manage all the service information needed to support the uses of each service by the Data Governance Framework.

3.1.2 Data governance Framework integration with the ACROSS platform

The next figure shows the integrated view of the whole ACROSS platform, including the interaction between the User Journey Services Engine and the ACROSS Personal Data framework with the Usage Control module.

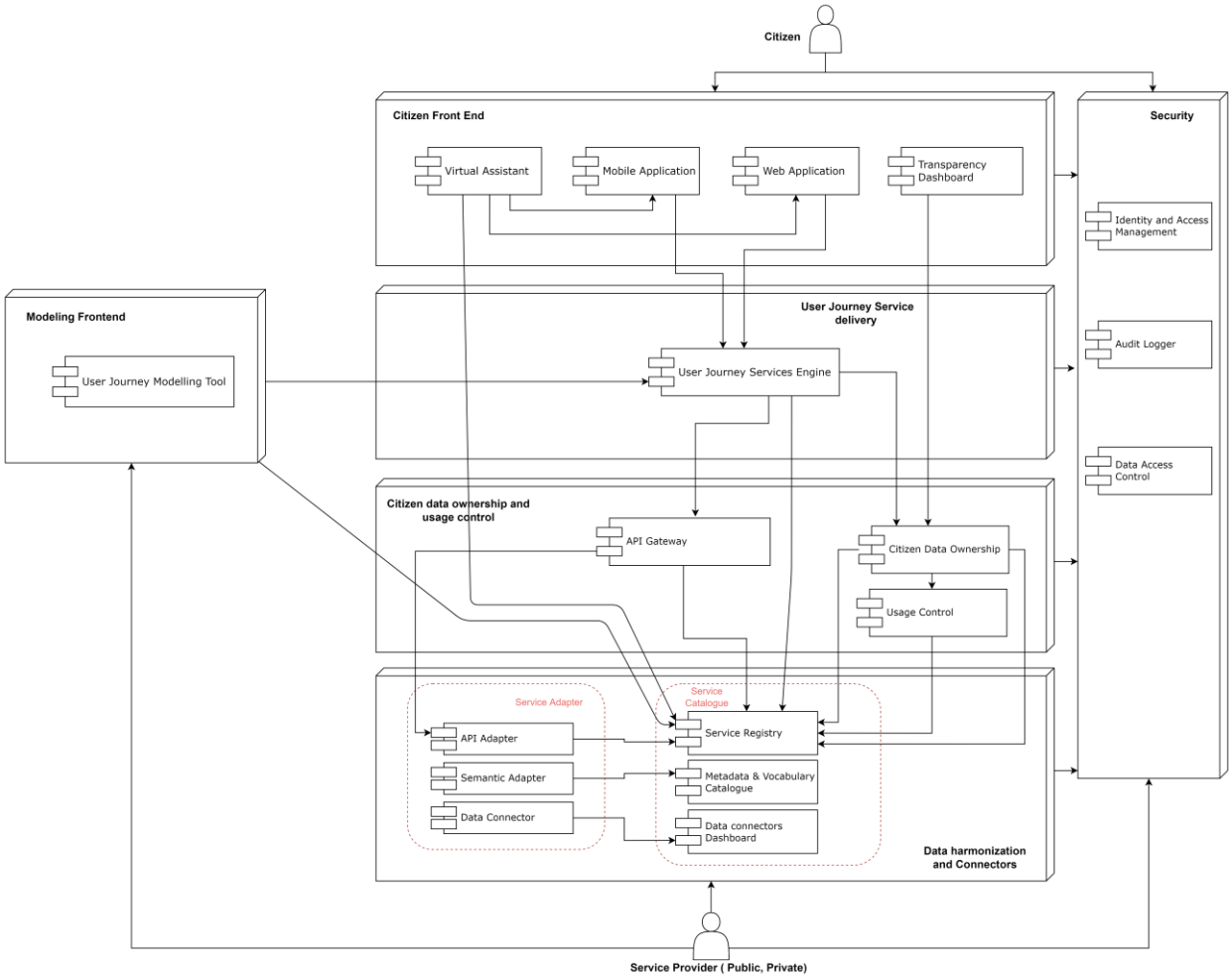


Figure 2 ACROSS platform architecture integrated view

3.1.3 Data governance framework scenario description

This section describes the workflow the users must follow to use final implementation of the ACROSS Data Governance framework for data sovereignty. Three types of users are envisaged: Administrator, End User (Citizen) and Service provider.

1. **Administrator:** Users management. Register and manage new users including end users and services providers. This functionality is provided by the external security package.
2. **Service provider:** Service description and registration. Each service provider has to register the services using the CSPV-AP extended model.
3. **End User:**
    a. Select services→ Select the services the user is going to use.
    b. Consent Management→ Define the personal data to be used by each selected service.
    c. Data Usage policies management→ Define the data usage policies applicable the data to be used by each service. This is an optional step.
    d. Monitor the data usage for each service
    e. Monitor the services using a specific personal data category

Once the end user has defined the consents for the selected services, the ACROSS User Journey engine will use the APIs to interact with the Personal Data Governance Framework.

The following logical interfaces have been defined:

- **Check consent**: The UJSE request to check the permissions to use personal data of a specific user by a service.
- **Check Data usage policy**: The UJSE request to enforce the data usage policies defined by a user for a service.
- **Pending services**: The UJSE inform the framework about the services included in a new workflow.
- **Notify data usage**: Each time a service is used the UJSE notifies the Data Governance framework about the personal data usage for logging and audit functions.
- **Define consent**: If a service called by the UJSE doesn't have the consent to use the personal data, this interface gives the user the possibility of creating the consent directly from the USJE.

Keycloak will be used for Identity and Access Management instead of the SSI Authentication included in the figure.

Next figure shows the whole scenario with the main actors, components and interactions.

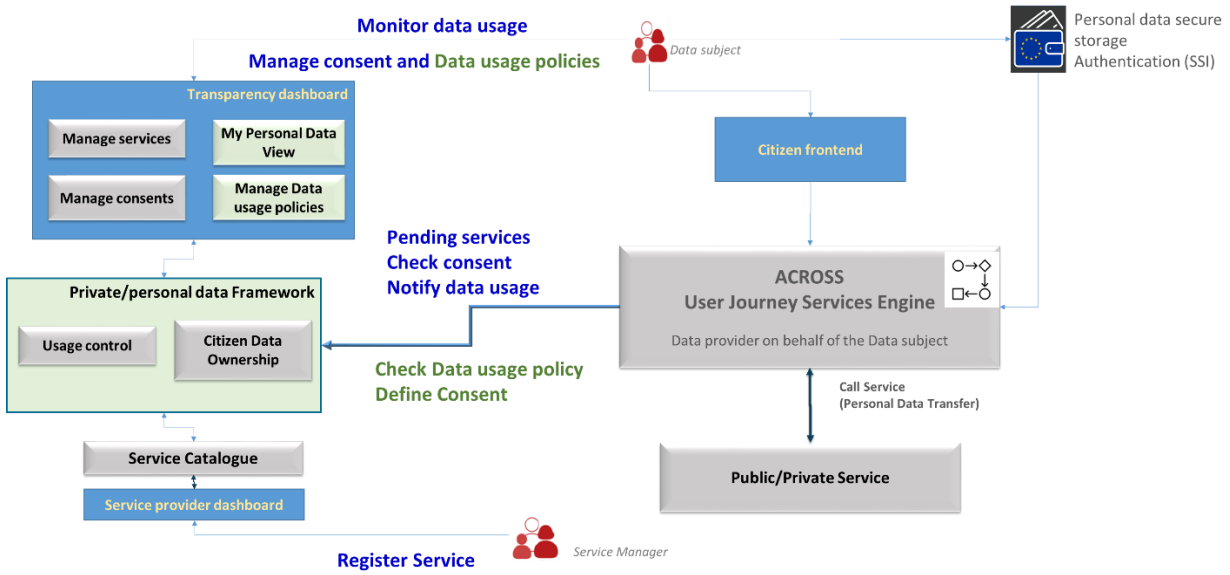


Figure 3 ACROSS Personal data governance framework scenario

Next, the ACROSS Personal data governance framework main functionalities are presented in more detail:

3.1.3.1 Administrator: Users management

The data Governance Framework must allow an end user to create a new account or to remove it. The first time a user enters the framework, the end user will have the opportunity to create an account in the framework. At least three types of users are envisaged: Administrator, End User (Citizen) and Service provider. The basic database structure needed to store the information about users has been implemented. However, the administration front-end has not been implemented.

3.1.3.2 Service provider: Service management

A service provider will be able to:

- Create and Edit all descriptions of Services that will be integrated with the framework, according to the Service Description Data Model defined
- Get an overview and manage the lifecycle of Services Descriptions (Create, Import, Export, Register, De-Register, Delete and so on).
- Get an overview and details of the Consents that End Users have given at corresponding registered Services, in particular:
    - Processing and Purpose details.
    - Consents history.
    - Consents raw data (JSON).

3.1.3.3 End User: Service, consent and data usage policies management

The end user will be able to:

- Get an overview of his personal data being processed by the Services he is linked to.
- Get an overview of previously registered Services by Service Providers, and ready to be selected and of already selected Services.
- Select a service.
- Unselect a Service. This will put all its active Consents (if any) in Disabled state
- Get an overview of given or pending Consents, where the following information will be provided:
    - Processed personal data
    - With which Organization data can be shared
    - Other info (such as the purposes for which the data can be used, e.g. by describing the relevant services)
- Manage the lifecycle of given Consents by changing its status:
    - **Disable**: disable the Consent. This means that no data sharing is possible anymore via the ACROSS Platform, but the characteristics of the prior data sharing relationship are stored. In that way, the user can reactivate the Consent at any time, and the same terms will apply as before.
    - **Activate**: enables the previously disabled Consent or pending Consent.
    - **Withdraw**: revoke the Consent entirely. No characteristics of the prior data sharing relationship are stored; therefore, a new one must be given if the user wants to resume sharing.
- Enable or disable each single Data Concept contained in the Resource Set regulated by that Consent (e.g.: his age).
- Manage (creation/modification/removal) of the data usage policy rules that will describe how the personal data should be used by the service.

3.2 Data Governance framework modules

This section includes the design of the Data governance framework modules including the description, their main functionalities and logical interfaces and the interaction with the other ACROSS modules.

The component cards are included in D5.2 System Architecture & Implementation Plan – Final [5] but are also added in this deliverable for readability. The logical interfaces described are implemented by the APIs included in section 3.4.

3.2.1 Transparency Dashboard

This module is a web application that uses a human centric approach to manage the potential of personal data and to facilitate its controlled flow from multiple data sources to applications and services. Citizens are able to opt-in (by giving their consent) and out (by withdrawing it) from the use of their personal data, in line with the requirements of the GDPR. The main objective is to give the individual control of their own data.

Table 3 Transparency Dashboard component card

<table>
<tr><td>Component Name</td><td colspan="2">Transparency Dashboard</td></tr>
<tr><td>Module Description</td><td colspan="2">A web application that uses a human centric approach to liberate the potential of personal data and to facilitate its controlled flow from multiple data sources to applications and services. Citizens must be able to opt-in and out from the use of their personal data, in line with the requirements of the GDPR.</td></tr>
<tr><td>Main functionalities</td><td colspan="2">The main objective is to give the individual control of their own data.
The component provides the following functionalities:
• Monitor which data are available and how they are used or how it can be accessed. It contains individual’s linked services, and data use related policies and consents.
• Users receive notifications about realised data processing at services.
• Give users control over their data allowing them to add as well as delete or modify information, and (subject to further discussions) exercise their rights as data subjects towards service providers.
• Enable the citizen to check the services that its personal data uses and make possible to remove this personal data from the previous created consent.</td></tr>
<tr><td>Main logical Interfaces</td><td>Interface name</td><td>Description</td></tr>
<tr><td></td><td>SearchConsent</td><td>Seach consents by different criteria.</td></tr>
<tr><td></td><td>ModifyConsent</td><td>Modify consent status (e.g.: withdraw), enable or disable specific data to which consent applies, change organizations to which data is shared.</td></tr>
<tr><td></td><td>ViewLogs</td><td>Show information about the events that have happened related to the linked services and the consents given/withdrawn.</td></tr>
<tr><td></td><td>SearchServices</td><td>Search services by different criteria.</td></tr>
<tr><td></td><td>LinkUserToService</td><td>Link a user to a service, so that he can manage the consents given to that service.</td></tr>
<tr><td>Interaction with other components</td><td>Interfacing Component</td><td>Interface Description</td></tr>
<tr><td></td><td>Audit Logger</td><td>It traces all security logs</td></tr>
</table>

	Identity and Access Management	Grants or denies other applications to access an exposed method (authentication and authorization)
	Citizen Data Ownership	Manages control over the data, being able to establish consents and data usage policies on them as well as modify or revoke them

3.2.2 User Journey Services Engine

The User Journey Services Engine is responsible for providing and initiating the User Journey Service workflows to the Citizens as well as invoking the specific services from which they are composed. The Citizens can access those workflows and specific services through the Mobile Application, the Web Application, or the Virtual Assistant. Thus, the User Journey Services Engine is responsible for performing the service orchestration by interpreting the provided workflow description and for intermediating between user interface requests and specific services.

The User Journey Service Engine module detailed description has been included since it is the main component interacting with the Data Governance framework via the Citizen Data Ownership.

Table 4 User Journey Services Engine component card

Component Name	**User Journey Services Engine**
Module Description	This component has two main responsibilities: 1. It manages machine-readable orchestration descriptions from the User Journey Modelling Tool and instantiates them into concrete User Journey Service Workflows (which include the concrete executable services) to the frontend components and thus the citizens. This instantiation will be carried out when the user specifies the abstract workflow (e.g.: work abroad, study abroad) he wants to carry out and the city/country from/to where to travel. 2. It performs the service orchestration and thus coordinates the communication with the API-Gateway that is necessary for the User Journey Service execution.
Main functionalities	The component provides the following functionalities: 1. It is able to receive and manage orchestration descriptions (abstract workflow templates) from the User Journey Modelling Tool. 2. It is able to provide an overview of the available User Journey Services. 3. Following a request from a citizen, it orchestrates the necessary service calls by interpreting the according service orchestration description and coordinating the communication with the API-Gateway. 4. It is responsible for intermediating user interface requests bidirectionally between the user interface components and the individual services executed within the concrete workflow's orchestration. 5. This intermediation is based on information received from the service registry.

Main logical Interfaces	**Interface name**	**Description**
	Manage workflow templates	Store, modify, delete workflow templates
	Get available workflow templates	Get the available workflow templates to instantiate.
	Instantiate workflow template	According to a workflow template and a destination country, provided as input parameters to the interface, translate workflow steps into concrete services.
	Steer workflow	Interface for interfacing from the UI to the User Journey Services Engine: start a workflow, kill a workflow, inform that a concrete workflow step has been completed offline (e.g.: by phone, etc.), etc.
	Monitor workflow status	Get status of workflow: before start, running at step xyz, completed, exited successfully, exited unsuccessfully at step xyz, etc.

Interaction with other components	**Interfacing Component**	**Interface Description**
	User Journey Modelling Tool	The component is able to receive and manage orchestration descriptions from the UJM backend tool.
	API Gateway	The component intermediates between user interface and individual services via the API Gateway.
	Web Application	The component is able to provide to the Web Application the available abstract workflows and the available User Journey Services. It is also able to send and receive user requests to and from the Web Application.
	Mobile Application	The component is able to provide to the Mobile Application the available abstract workflows and the available User Journey Services. It is also able to send and receive user requests to and from the Mobile Application.
	Service Registry	The component receives from the Service Catalogue (via service registry component) the necessary information for the invocation of service instance.
	Usage Control	Whenever the citizen inputs data for a service, before transferring it to that service, this data must be processed by the Usage Control component to enforce previously defined usage policies.

3.2.3 Citizen Data Ownership

The citizen data ownership ensures that the data provided by the citizens to the services, is used by the service providers taking into account the consents approved by them.

Table 5 Citizen Data Ownership component card

Component Name	**Citizen Data Ownership**	
Module Description	This component allows the citizens to manage their personal data and allows the organizations/services to fulfil the requirements in line with the GDPR. It will expose several interfaces for the Transparency Dashboard, so that the individuals can grant and withdraw their consents and receive notifications about how their data is being used. On the other hand, it will expose several interfaces for the services, so that they can be informed about the consents and data usage policies of the citizens, and they can send notifications about the data that is being used.	
Main functionalities	The component provides the following functionalities: • Enable the citizens to grant and withdraw their consents for data processing. • Enable the citizens to define data usage policies. • Provide to the citizens notifications about their data processing. • Provide to the services information about the consents and data usage policies of the citizens. • Enable the services to send the notifications about the usage of the data.	
Main logical Interfaces	**Interface name**	**Description**
	SearchConsent	Search consents by different criteria.
	ModifyConsent	Modify consent status (e.g.: withdraw), enable or disable specific data to which consent applies, change organizations to which data is shared.
	ViewLogs	Show information about the events that have happened related to the linked services and the consents given/withdrawn.
	SearchServices	Search services by different criteria.
	LinkUserToService	Link a user to a service, so that he can manage the consents given to that service.
Interaction with other components	**Interfacing Component**	**Interface Description**
	Transparency Dashboard	Read, insert, modify or remove consents of the citizens. Access notifications of usage of the citizens data.
	Service Registry	Access services related information saved in the registry and link services to citizens.
	Metadata and Vocabulary Catalogue	To map the different data models used by the different services involved in ACROSS.
	Audit Logger	It traces all security logs.

	Identity and Access Management	Grants or denies other applications to access an exposed method (authentication and authorization).
	Usage Control	List of consents established between the citizens and the services.

3.2.4 Usage Control

This component provides the enforcement mechanism to apply usage policies according to previously defined data usage policies. The available formats of data usage policies include GDPR consents and IDS data policies enforcement. The ACROSS personal data government framework strategy is to minimize the service providers adaptation needed to use the framework and it is not realistic to ask public and private services to use IDS connectors for data transfer. Therefore, the ACROSS Personal Data Governance Framework will assume the responsibility of performing data usage policies management and enforcement.

The User Journey Service Engine will call the framework before transferring the personal data to the service to enforce both personal data consents and data usage policies. Since ACROSS is not going to use IDS connectors for data transfer, the data usage policies are applied only in the data provider side. Therefore, no real "data usage control" can be applied, only a restricted set of IDS policies providing data access rules. Furthermore, the contract negotiation phase is not needed.

In order to be used within the ACROSS data governance framework the Data usage app needs to be adapted by changing the contract format and API. Contracts in IDS represent agreements between companies exchanging data and are defined for specific "artifacts" or data sets. In ACROSS the contract represents agreements between end-users and public/private services for using personal data.

Table 6 Usage Control component card

Component Name	**Usage Control**	
Module Description	The component provides the enforcement mechanism to apply usage policies according to previously defined consents.	
Main functionalities	The main objective is to provide the security and privacy aspects of the data shared among the different systems connected. The IDS data usage control mechanism has been studied and adapted to the platform.	
Main logical Interfaces	**Interface name**	**Description**
	UsageControlEnforcement	Apply usage policies so that data is used accordingly

Interaction with other components	**Interfacing Component**	**Interface Description**
	Audit Logger	It traces all security logs
	Identity and Access Management	Grants or denies other applications to access an exposed method (authentication and authorization).
	Metadata & Vocabulary Catalogue	It retrieves metadata and data mapping for semantic adaptation
	Citizen Data Ownership	Access the data usage policies established between the citizens and the services.

3.2.5 Service Catalogue

The Service Catalogue aims at providing all functionality to register, model, map and publish and manage all the service information needed to support the uses of each service by the Data Governance Framework. Its is composed of three modules: the service registry, the Metadata and Vocabulary Catalogue, and the Data Connectors Dashboard.

This service registry provides human and machine-readable description of services that will be available in ACROSS platform for user journey services provisioning. The registry enables the storage and publishing of service by providing general, technical and data processing information based on standard models (e.g., ISA^2, W3C DPV, etc.). The component provides the following functionalities:

- Publishing, searching, and retrieving of an already available service in the platform
- Service Description versioning
- API for programmatically interaction with the registry
- User Dashboard and service editor: manage the Semantic Descriptions and registrations of its own provided Services, so that it is available for the citizens through the Transparency Dashboard.

The metadata and vocabulary catalogue supports the storage of shared models and vocabularies used to describe public and private service available by means of the platform. The module also allows users to create service and data mappings by means of web interfaces (Service Registry component) between the stored metadata and vocabulary and any other data model for describing public and private services and related data processing.

The Data connectors dashboard provides a visual dashboard to manage the status of services connected to the platform by means of a specific data connector instance.

The detailed component card of this component has not been included since it is included in D5.2 and detailed in WP4.

3.3 Citizen Ownership and Data Usage Control interaction diagram

The main high-level interaction diagram of Citizen Ownership and Data Usage Control layer can be summarized in the image below (as described in D5.2):

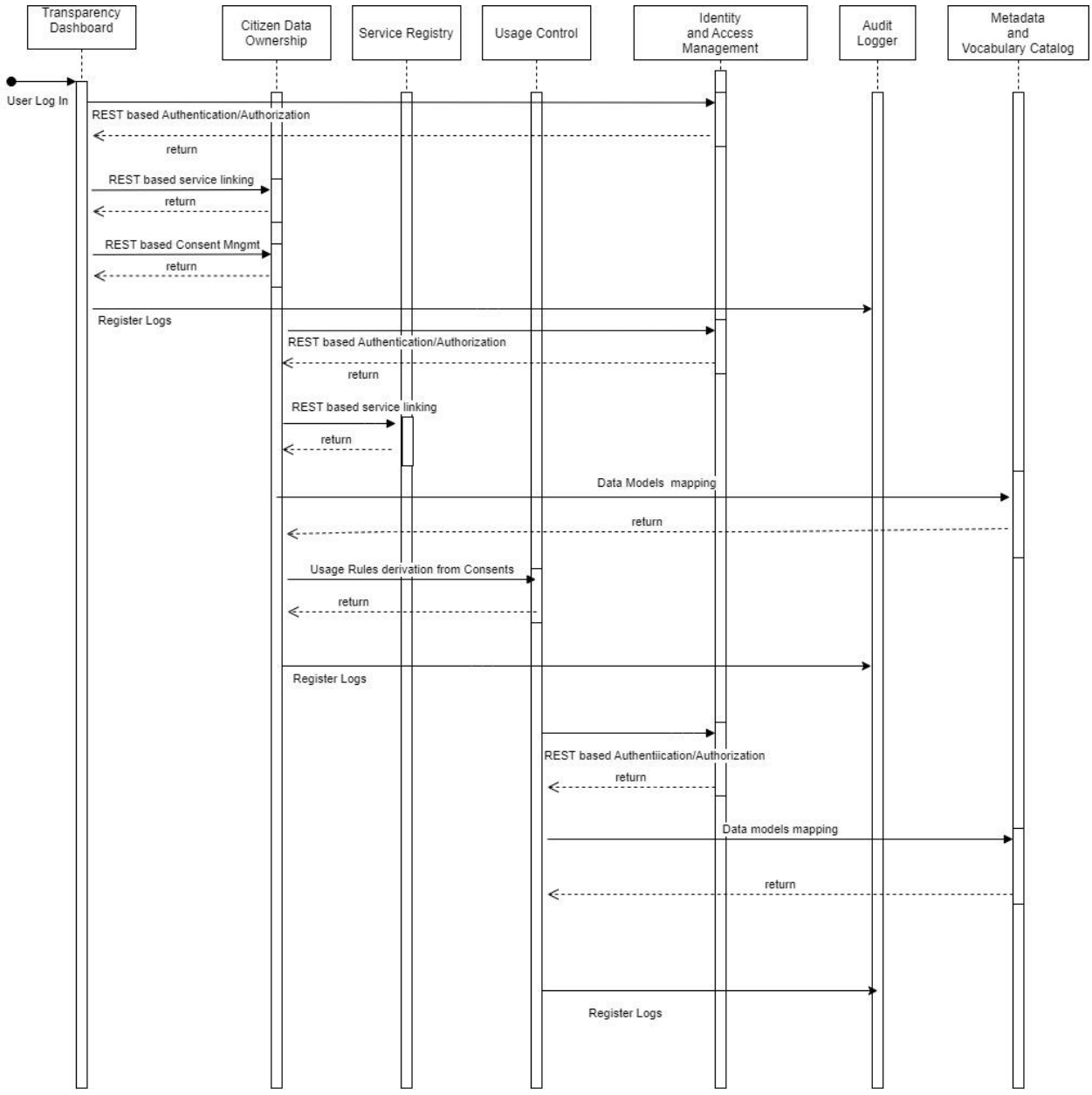


Figure 4 Citizen Ownership and Data usage control interaction flow

The main interaction scenarios are:

- Transparency Dashboard invocation. This scenario includes the functionalities that gives to individual control of their own data.
- Citizen Data Ownership invocation. Expose several APIs for the Transparency Dashboard so that the individuals can grant and withdraw their consents and receive notifications about how their data is being used.
- Service Registry invocation provide human and machine-readable description of services that will be available in ACROSS platform for user journey services provisioning.
- Usage Control invocation. This scenario provides the enforcement mechanism to apply usage policies according to previously defined consents.
- Identity and Access Management. This interaction scenario includes the interface with the Authentication node (eIDAS, SSO).
- Audit Logger invocation. It traces all security logs.
- Metadata and Vocabulary Catalogue invocation. Expose several APIs, through the Service Registry mediation, to support the storage of shared models and vocabularies used to describe public and private service available by means of the platform and allows users to create data mappings.

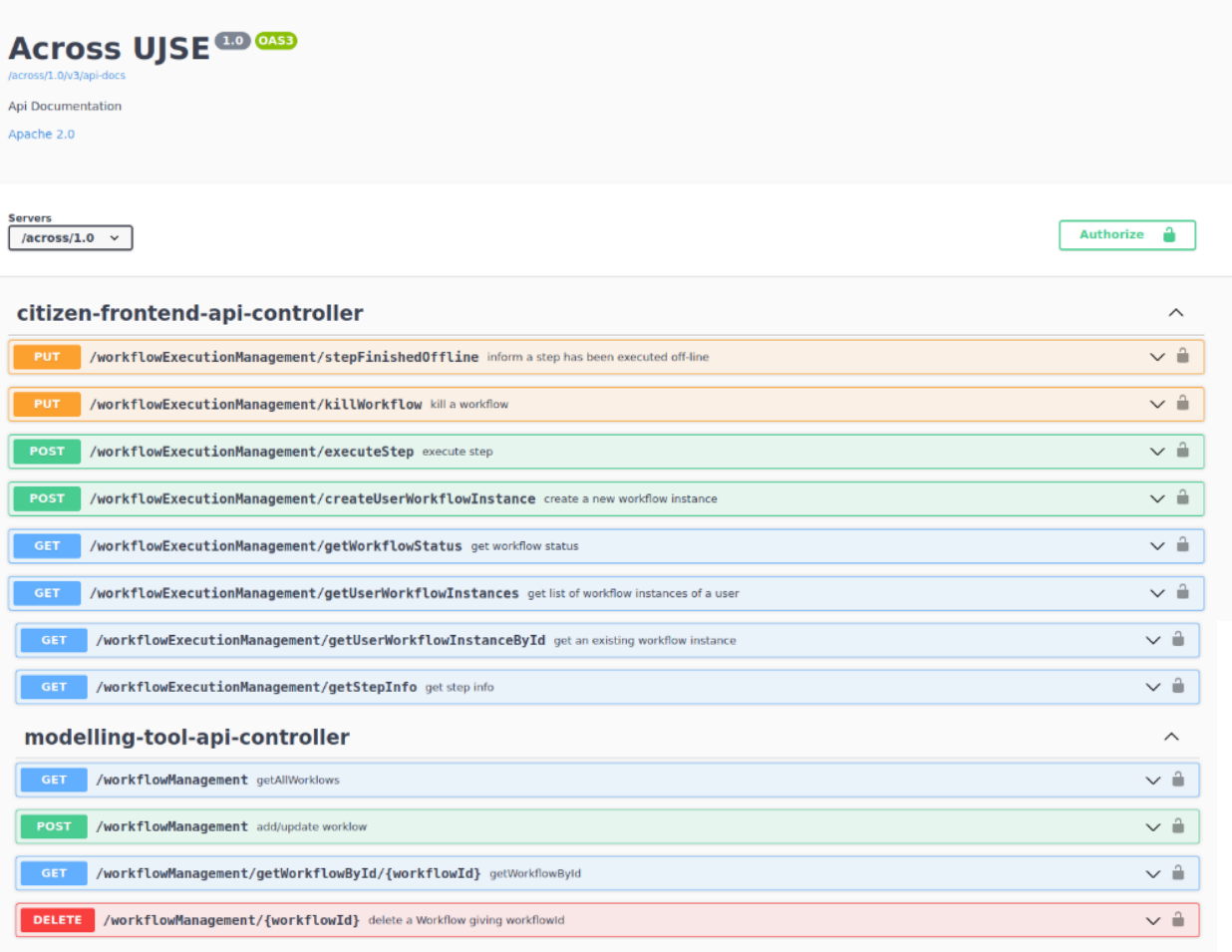
3.4 APIs

This section includes the detailed description of the rest APIs exposed by the Data Governance framework. The User Journey Service Engine APIs have been included since it is main component interacting with the Data Governance framework via the Citizen Data Ownership.

3.4.1 User Journey Service Engine

This component has two main responsibilities:

1. It manages machine-readable orchestration descriptions from the User Journey Modelling Tool and instantiates them into User Journey Services (which are the concrete executable workflows) to the frontend components - and thus the citizens.
2. It performs the service orchestration and thus coordinates the communication with the API-Gateway that is necessary for the User Journey Service execution. The orchestration might include processing of external events.



REST API: PUT /workflowExecutionManagement/stepFinishedOffline

Description: Puts as finished a concrete "serviceId" that belongs to a step.

Input parameters:

- workflowInstanceId: id of workflows instanciated by the user
- stepId: the id of the step of the workflow
- serviceId: id of the service

Interaction with other components: Citizen Front End

REST API: PUT /workflowExecutionManagement/killWorkflow

Description: Kill a concrete workflowInstaceId.

Input parameters:

- workflowInstanceId: id of workflows instanciated by the user

Interaction with other components: Citizen Front End

REST API: POST /workflowExecutionManagement/executeStep

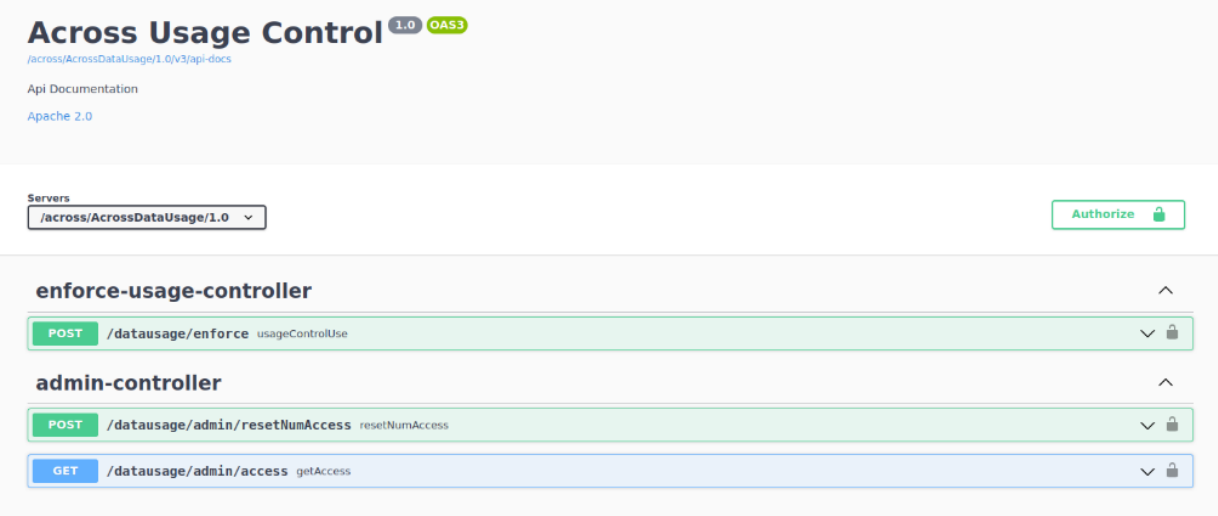
Execute a step that it is included in a workflow.

Input parameters: - workflowInstanceId: id of workflows instanciated by the user - stepId: the id of the step of the workflow - serviceId: id of the service - inputData **Interaction with other components:** Citizen Front End
REST API: POST /workflowExecutionManagement/createUserWorkflowInstance Create an instance of a workflow. **Input parameters:** - userId: User Id of the user - countryOrigin: country origin - countryDestination: country destination - workflowType: type of workflow study or work **Interaction with other components:** Citizen Front End
REST API: GET /workflowExecutionManagement/ Informs about status of the workflowInstanceId. **Input parameters:** - workflowInstanceId: id of the workflow **Interaction with other components:** Citizen Front End
REST API: GET /workflowExecutionManagement/getUserWorkflowInstances Retrieves all workflowInstances given an userId. **Input parameters:** - userId: User Id of the user **Interaction with other components:** Citizen Front End
REST API: GET /workflowExecutionManagement/getUserWorkflowInstanceById Retrieves all workflowInstances given a workflowInstanceId. **Input parameters:** - workflowInstanceId: id of the workflow **Interaction with other components:** Citizen Front End
REST API: GET /workflowExecutionManagement/getStepInfo Get information about invocation of a specific service in a concrete step. **Input parameters:** - workflowInstanceId: id of workflows instanciated by the user - stepId: the id of the step of the workflow

- serviceId: id of the service **Interaction with other components:** Citizen Front End
REST API: GET /workflowManagement/getAllWorkflows Get information about all the workflow uploaded into UJSE. **Input parameters:** - None **Interaction with other components:** Citizen Front End
REST API: POST /workflowManagement add/update worklow Upload new or updated workflow to UJSE. **Input parameters:** - userId: User Id of the user - countryOrigin: country origin - countryDestination: country destination - workflowType: type of workflow study or work - file: file in bpmn format of the workflow **Interaction with other components:** User Journey Modelling Tool
REST API: GET /workflowManagement/getWorkflowById/{workflowId} Retrieve information from specific workflowId. **Input parameters:** - workflowId: id of workflows **Interaction with other components:** User Journey Modelling Tool
REST API: DELETE /workflowManagement/{workflowId} Delete a workflow given its workflowId. **Input parameters:** - workflowId: id of workflows **Interaction with other components:** User Journey Modelling Tool

3.4.2 Usage Control API

The component provides the enforcement mechanism to apply usage policies according to previously defined data usage policies.



REST API: POST /datausage/enforce

Make the enforcement of the policies defined by an userId to a concrete service.

Input parameters:

- serviceId: id of the service
- userId: logged userId
- body: policies to be enforced

Interaction with other components: TransparencyDashboard

REST API: PUT /datausage/admin/resetNumAccess

Reset the number of accesses to specific service by a user when the policy regarding number of accesses has been deleted.

Input parameters:

- serviceId: id of the service
- userId: logged userId

Interaction with other components: Transparency Dashboard

GET /datausage/admin/access

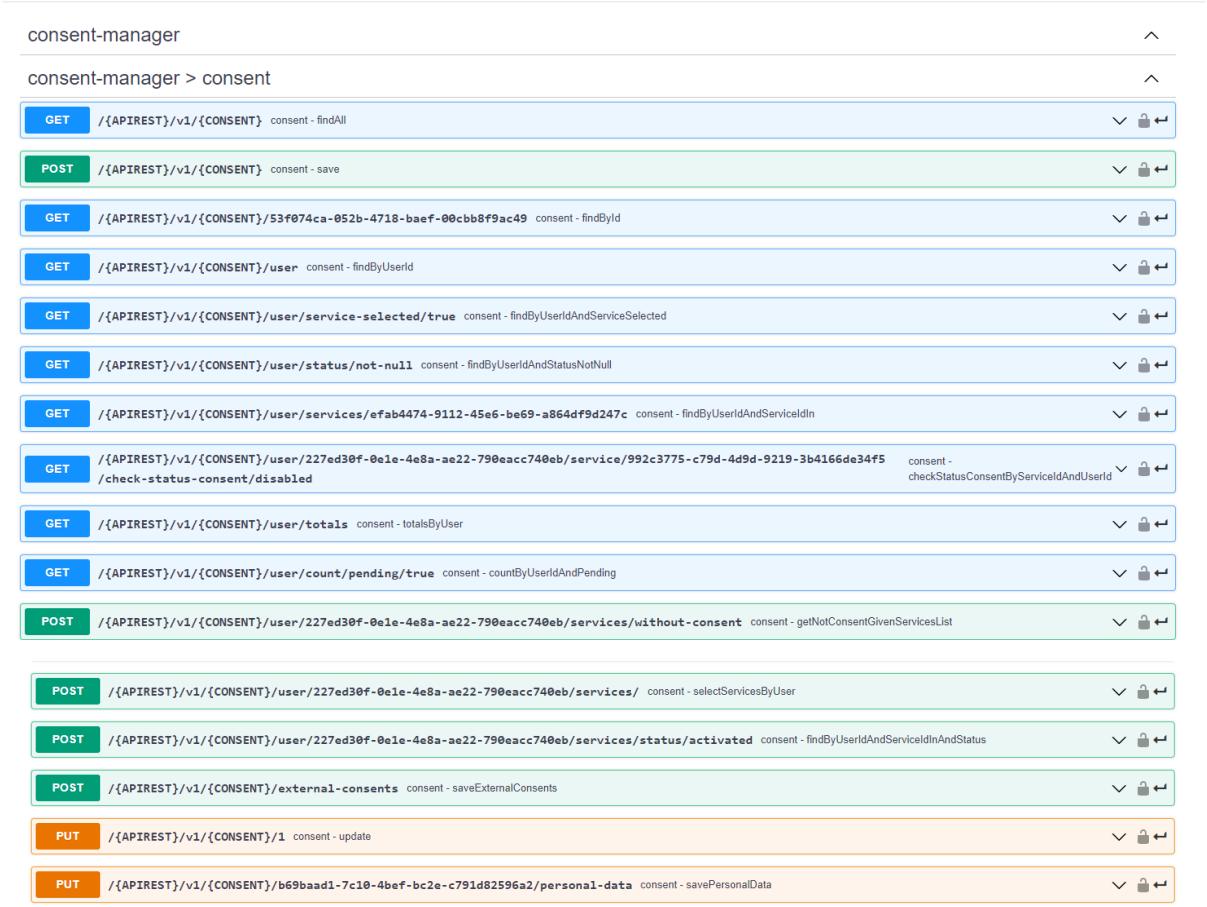
This API is called when the policy that checks the number of hits for a specific service is applied.

Input parameters:

- serviceId: id of the service
- userId: logged userId

Interaction with other components: UsageControl

3.4.3 Consent-manager: API rest to manage consents

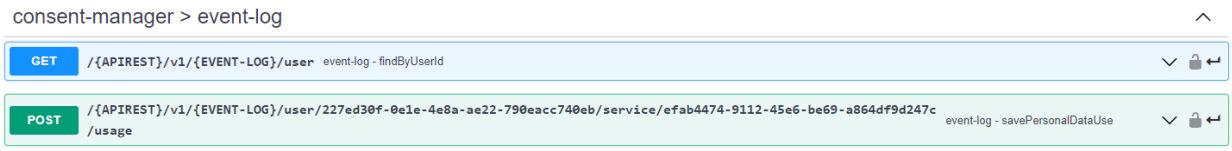


REST API CONSENT
REST API: GET /api/rest/v1/consent Get all given consents **Input parameters:** - None **Interaction with other components:** None
REST API: POST /api/rest/v1/consent Save a consent. **Input parameters:** - Body: consent in json format **Interaction with other components:** None
REST API: GET /api/rest/v1/consent/consentID Return consent information given a consentId. **Input parameters:**

- consent: id of the consent in the URL **Interaction with other components:** None
REST API: GET /api/rest/v1/consent/user/userId Get information given a user **Input parameters:** - user: id of the user **Interaction with other components:** None
REST API: GET /api/rest/v1/consent/user/userId/service-selected/true Get information when the service is selected **Input parameters:** - user: id of the user **Interaction with other components:** None
REST API: GET /api/rest/v1/consent/user/userId/status/not-null Get information giving userId when status is not null **Input parameters:** - user: id of the user **Interaction with other components:** None
REST API: GET /api/rest/v1/consent/user/userId/services/serviceId **Input parameters:** - userId: id of the user - serviceId: id of the service **Interaction with other components:** None
REST API: GET /api/rest/v1/consent/user/userId/check-status-consent/disabled Check if consent is activated or disabled **Input parameters:** - userId: id of the user - serviceId: id of the service **Interaction with other components:** User Journey Service Engine
REST API: GET /api/rest/v1/consent/user/userId/totals Get the number of total services by user and given consents. **Input parameters:** - userId: id of the user **Interaction with other components:** None
REST API: GET /api/rest/v1/consent/user/count/pending/true Get the number of services that are pending

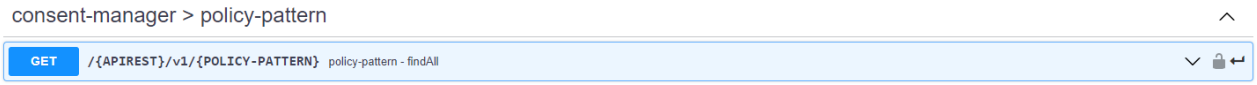
Input parameters: - userId: id of the user **Interaction with other components:** None
REST API: POST /api/rest/v1/consent/user/userId/services/without-consent Get the list of services without consent **Input parameters:** - userId: id of the user **Interaction with other components:** None
REST API: POST /api/rest/v1/consent/user/userId/services/ Get the selected services by the user **Input parameters:** - userId: id of the user **Interaction with other components:** None
REST API: POST /api/rest/v1/consent/user/userId/services/status/activated Get information about the services of the user when they are activated **Input parameters:** - userId: id of the user **Interaction with other components:** User Journey Service Engine
REST API: POST /api/rest/v1/consent/external-consents Create a new consent for the user **Input parameters:** • userId: id of the user • serviceIdList: list of the services **Interaction with other components:** User Journey Service
REST API: PUT /api/rest/v1/consent/1 Update an existing consent **Input parameters:** - consentId: id of the consent - consent: consent data in json format **Interaction with other components:** None
REST API: PUT /api/rest/v1/consent/userId/personal-data Save a concrete persona data **Input parameters:** -consentId: consent identifier - body: personal data information in json format **Interaction with other components:** None

3.4.4 Event-log: API to manage event logs into Transparency Dashboard



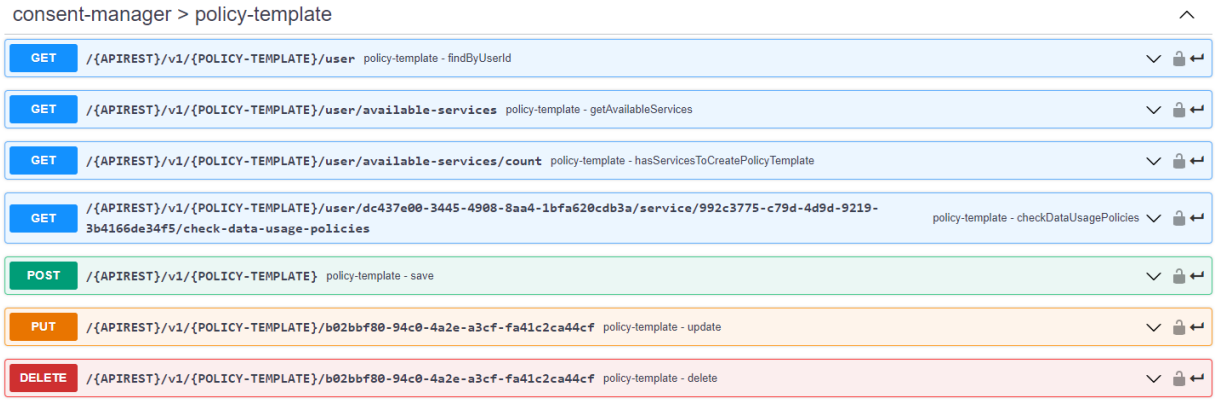
REST API EVENT-LOG
REST API: GET /api/rest/v1/{EVENT-LOG}/user Get all logs by user **Input parameters:** - userId: id of the user **Interaction with other components:** None
REST API: POST /api/rest/v1/{EVENT-LOG}/user/userId/service/serviceId/usage Save logs into Transparency Dashboard logs database. **Input parameters:** - userId: id of the user - serviceId: id of the service **Interaction with other components:** User Journey Service Engine

3.4.5 Policy pattern: API for policy patterns management



REST API POLICY PATTERN
REST API: GET /api/rest/v1/{POLICY-PATTERN} Get all policy patterns **Input parameters:** - None **Interaction with other components:** None

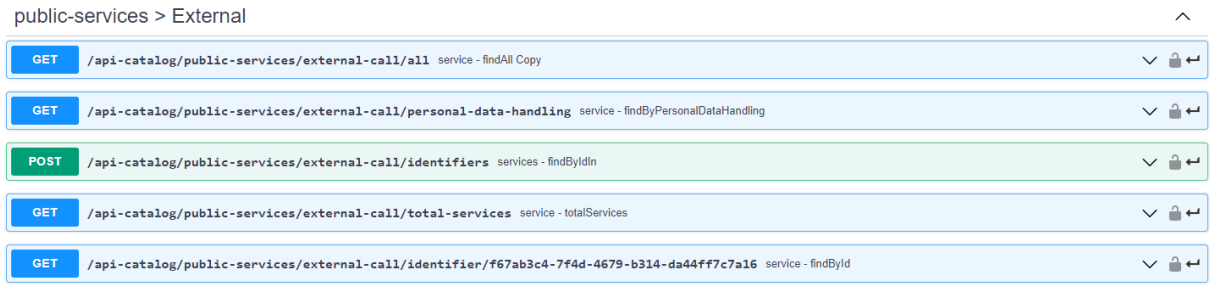
3.4.6 Policy-template: API for policies management



REST API POLICY TEMPLATE
REST API: GET /api/rest/v1/{POLICY-TEMPLATE}/user Get all policies by user **Input parameters:** - userId: id of the user **Interaction with other components:** None
REST API: GET/api/rest/v1/{POLICY-TEMPLATE}/user/available-services Get all available services by user **Input parameters:** - userId: id User **Interaction with other components:** None
REST API: GET/api/rest/v1/{POLICY-TEMPLATE}/user/available-services/count Return number of available services by user **Input parameters:** userId: id User **Interaction with other components:** None
REST API: GET /api/rest/v1/{POLICY-TEMPLATE}/user/userId/service/serviceId/check-data-usage-policies Check the policies enforcement by userId and serviceId **Input parameters:** - userId: id of the user - serviceId: id of the service **Interaction with other components:** User Journey Service Engine
REST API: POST /api/rest/v1/{POLICY-TEMPLATE} Save a policy **Input parameters:**

- policy: policy content **Interaction with other components:** None
REST API: PUT /api/rest/v1/{POLICY-TEMPLATE}/policy Update a policy **Input parameters:** - policy: policy content **Interaction with other components:** None
REST API: DELETE /api/rest/v1/{POLICY-TEMPLATE}/policyId Delete a policy **Input parameters:** - policy: policy content **Interaction with other components:** None

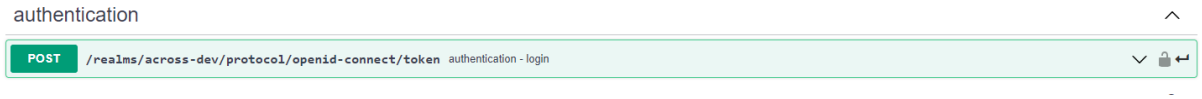
3.4.7 Public-service: API for public services management



REST API Public Services
REST API: GET /api-catalog/public-services/external-call/all Get all public services from the service catalogue **Input parameters:** - None **Interaction with other components:** Service Catalogue
REST API: GET /api-catalog/public-services/external-call/personal-data-handling Get the list of services who uses personal data services **Input parameters:** - None **Interaction with other components:** None
REST API: POST /api-catalog/public-services/external-call/identifiers Returns the identifiers of services.

Input parameters: - None **Interaction with other components:** Service Catalogue
REST API: GET /api-catalog/public-services/external-call/total-services Returns the number of total services into the service catalogue. **Input parameters:** - None **Interaction with other components:** Service Catalogue
REST API: GET /api-catalog/public-services/external-call/identifier/serviceId Returns the information a specific service **Input parameters:** - serviceId: id of the service **Interaction with other components:** Service Catalogue

3.4.8 Authentication: API for get token from Keycloak



REST API AUTHENTICATION
REST API: POST /realms/across-dev/protocol/openid-connect/token Get the token from Keycloak **Input parameters:** • client_id • username • password • grant_type **Interaction with other components:** None

3.5 Transparency dashboard user interface design

This section provides an overall overview of the Data Governance framework user interface design including the page general design and the main windows. A more detailed user interface design and user manual will be included in the implementation deliverables, including the modifications to improve the usability in the interface.

3.5.1 Transparency dashboard main page

Next figure shows the Data Governance framework main window. The main page structure is common for all pages:

- left frame with the vertical menu to access the framework functions
- a top frame with some context information (logged user, language and notifications icon) and login/logout function.
- a central frame which provides the interface for each functionality

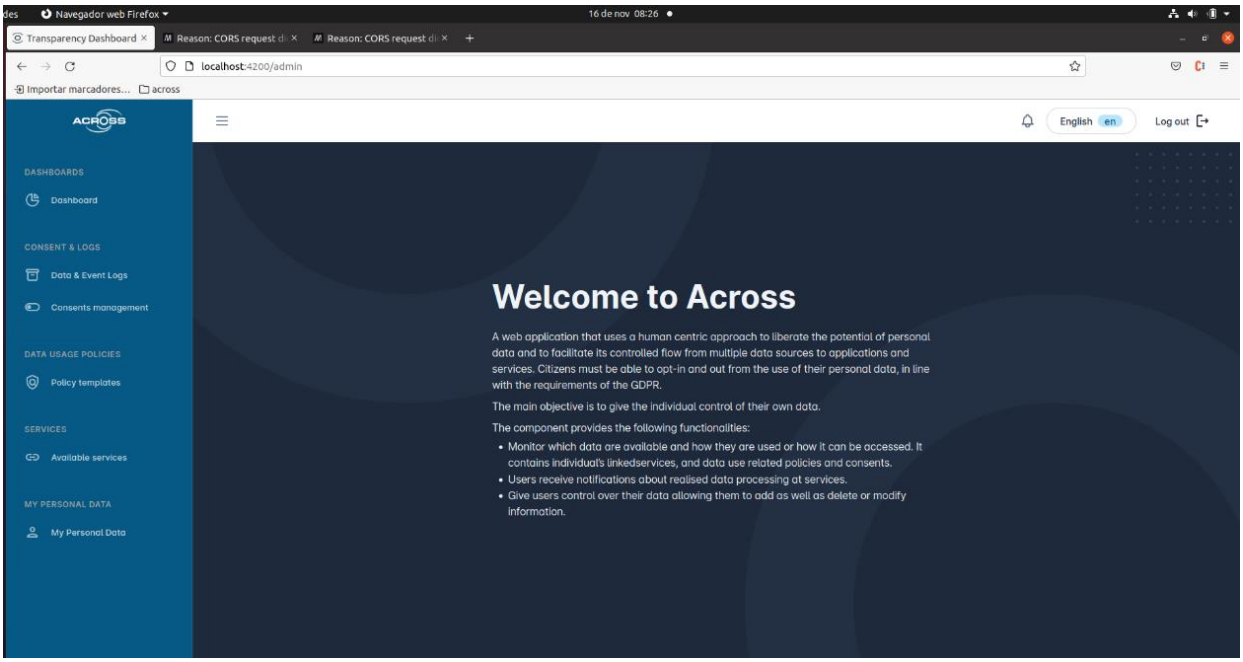


Figure 5 ACROSS main page

3.5.2 Dashboard

Next figure shows the dashboard page, which provides a summary view about the services the user is using along with the consents the user has defined for those services.

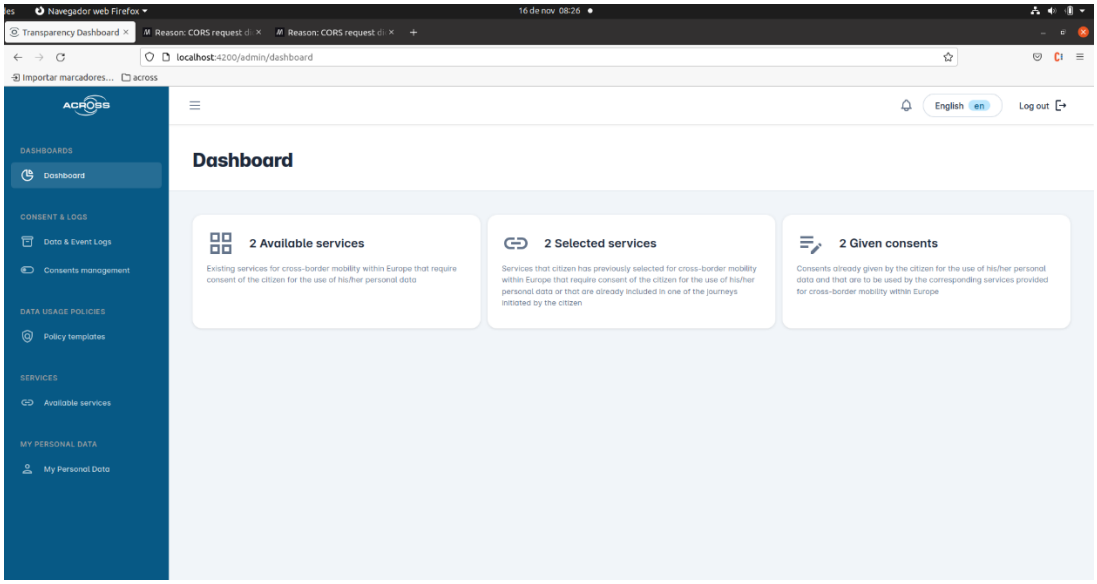


Figure 6 Data Governance framework Dashboard

3.5.3 Consent management

This window provides consent view and manage functionalities for the created consents.

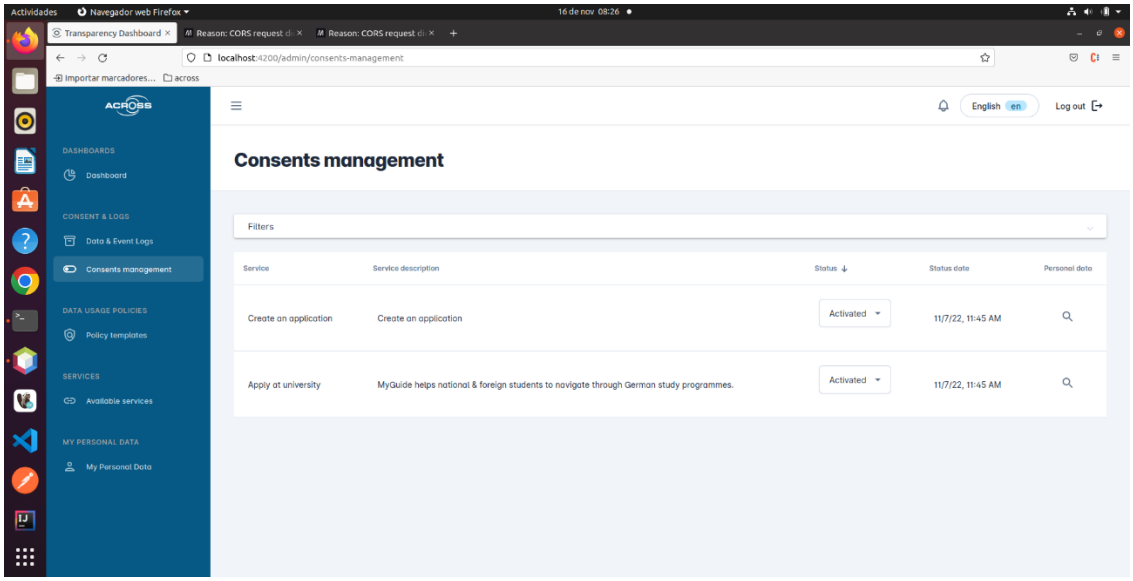


Figure 7 Consent management window

When defining a new consent all the personal data needed by the service, both mandatory and optional, are greyed out by default.

3.5.4 Service management

The available services window provides the list of services registered in the Service catalogue.

The following management functionalities are provided:

- Services browsing and filtering
- View services detailed information
- Services selection
- Consent creation

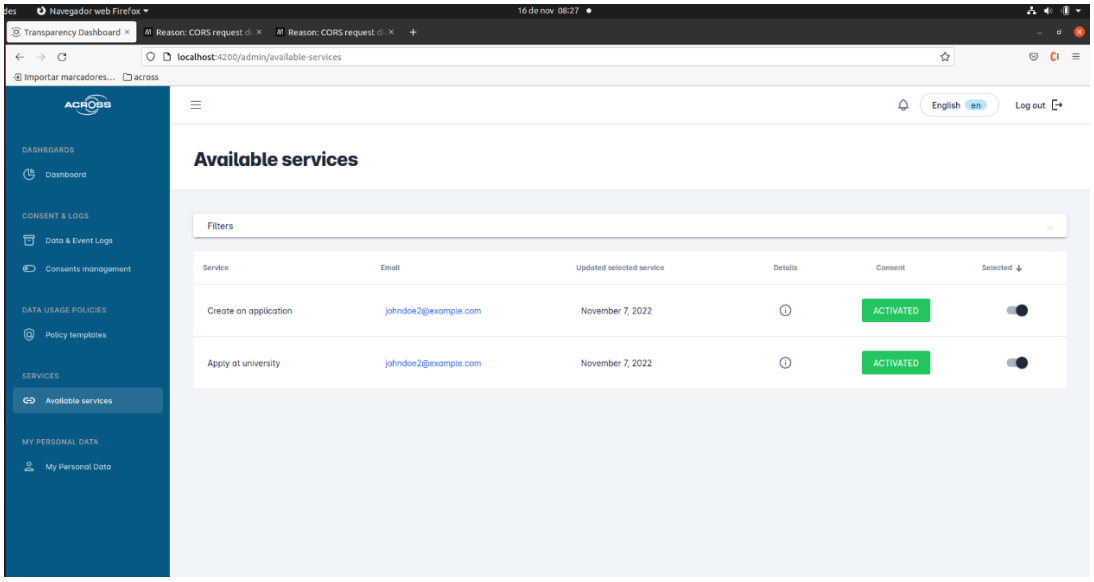


Figure 8 Service management window

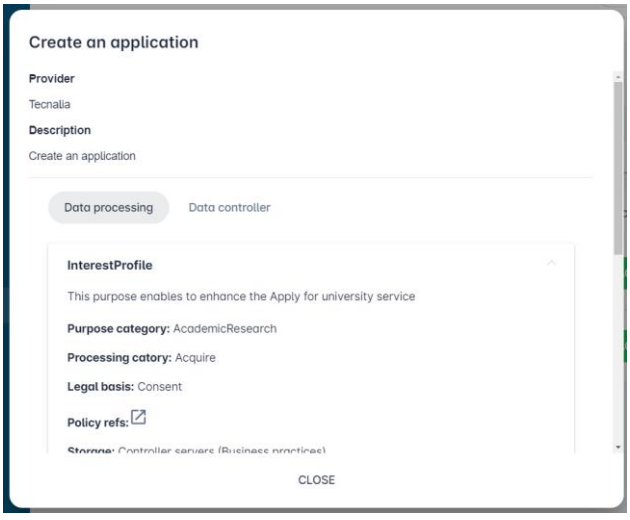


Figure 9 Service detailed information window

3.5.5 Data usage policy management

The Policy templates window access and management functionality for the data usage policies.

The following management functionalities are provided:

- Data usage policies browsing and filtering
- Data usage policy create/modify/delete functionalities

Each service is associated with a data usage policy that can be composed by one or several policy rules.

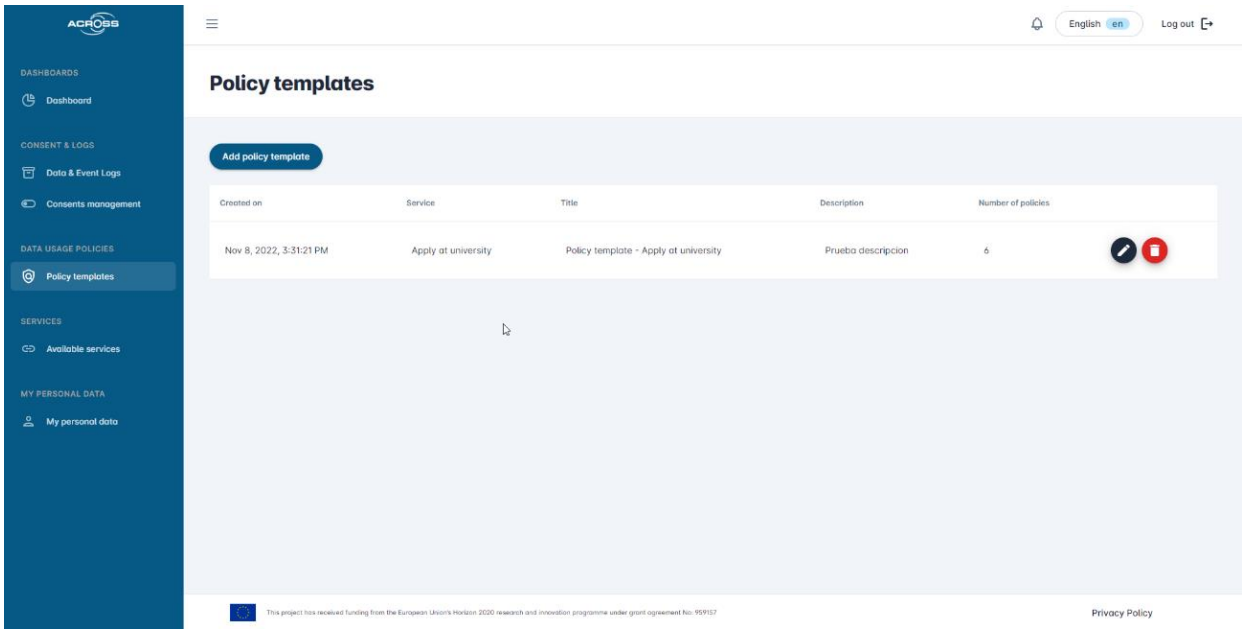


Figure 10 Data usage policy browse window

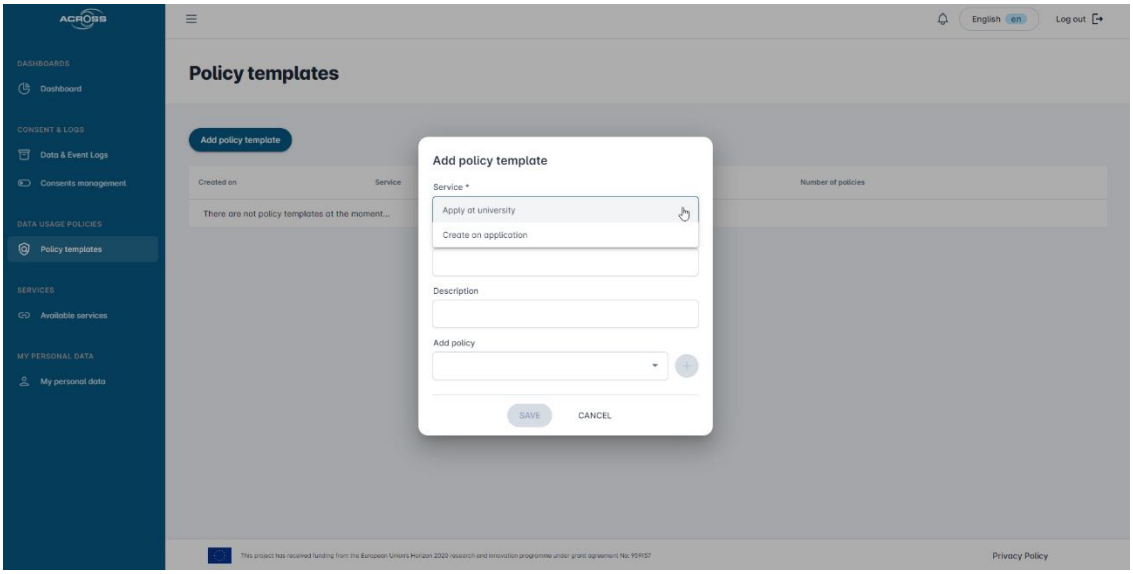


Figure 11 Add new data usage policy interface

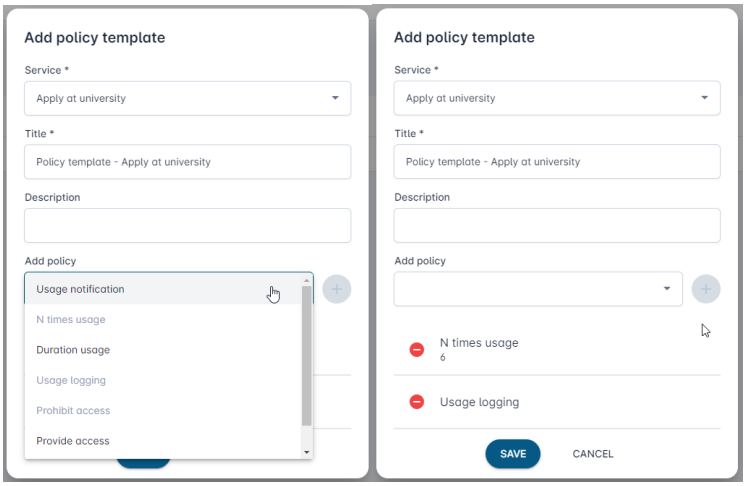


Figure 12 Add policy window examples: one policy composed by several policy rules

3.5.6 My personal data view

This new functionality allows the user to view the list of services that are using a specific personal data concept. In order to homogenise the personal data models, the ACROSS Personal Data Framework will use the personal data model defined on DPV Personal Data Category[6] taxonomy and classes.

DPV provides broad top-level personal data categories adapted from the taxonomy contributed by EnterPrivacy[7]. The top-level concepts in this taxonomy refer to the nature of information (financial, social, tracking) and to its inherent source (internal, external). Each top-level concept is represented in the DPV vocabulary as a Class and is further elaborated by subclasses for referring to specific categories of information - such as preferences or demographics.

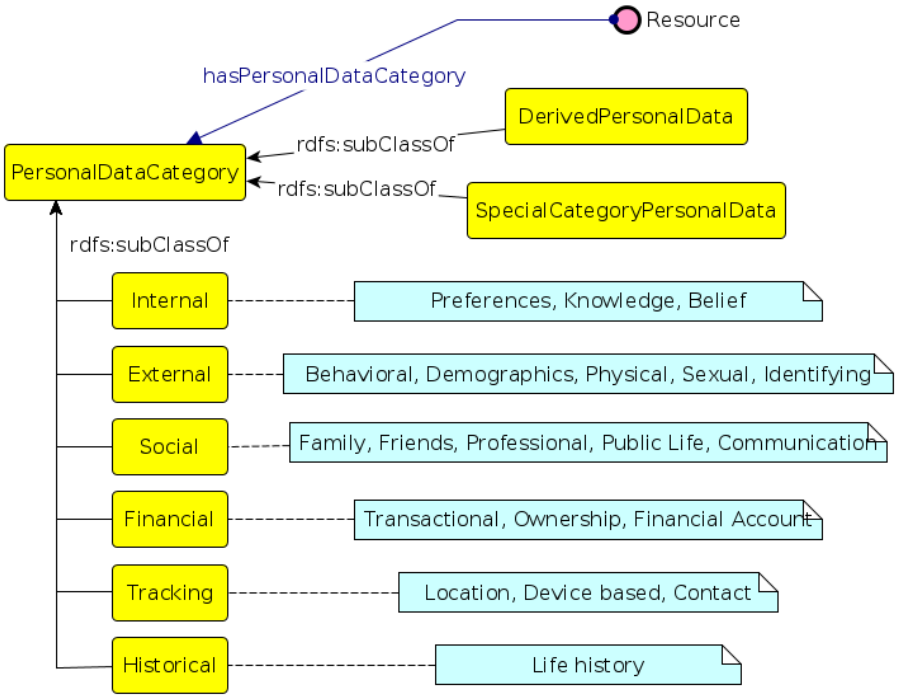


Figure 13 DPV taxonomy for personal data[8]

The framework will give the service provider a way to map its specific data model with the DPV Personal Data Categories data model. In this way, it would be possible to have a vision of where each type of data is being used in different services.

[6] https://w3c.github.io/dpv/dpv/#vocab-personal-data-categories
[7] https://enterprivacy.com/wp-content/uploads/2018/09/Categories-of-Personal-Information.pdf
[8] https://enterprivacy.com/wp-content/uploads/2018/09/Categories-of-Personal-Information.pdf

The model will be implemented but the frontend to define the mapping and the frontend for the end user are not included.

Next figures show the user interface of My Personal Data view. From this interface is possible to revoke or grant the consent to use a specific data category by a service.

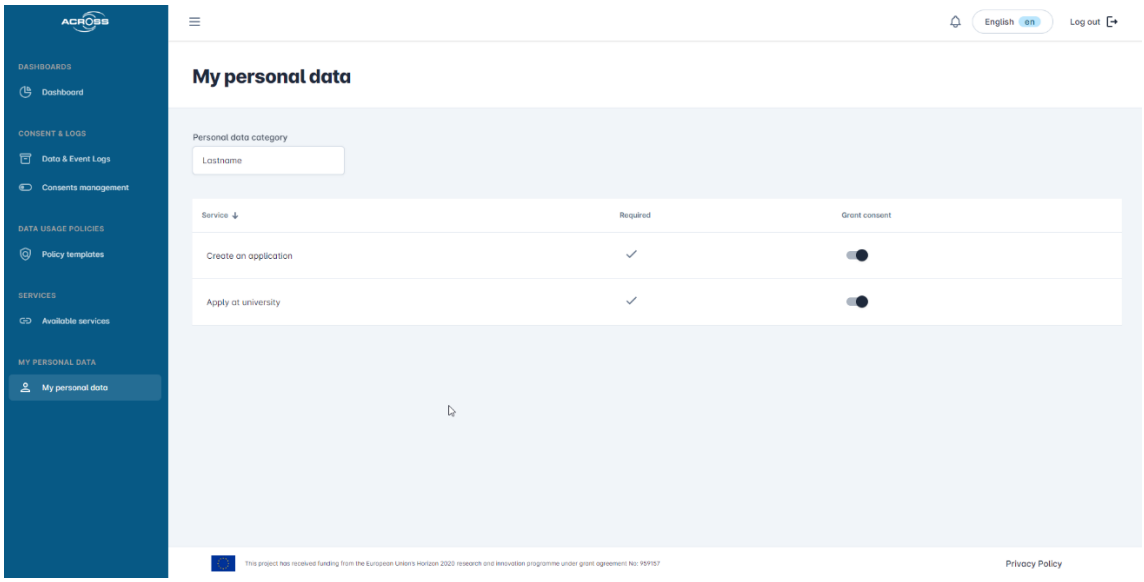


Figure 14 Personal data view interface: list of services using the "Last name" personal data category

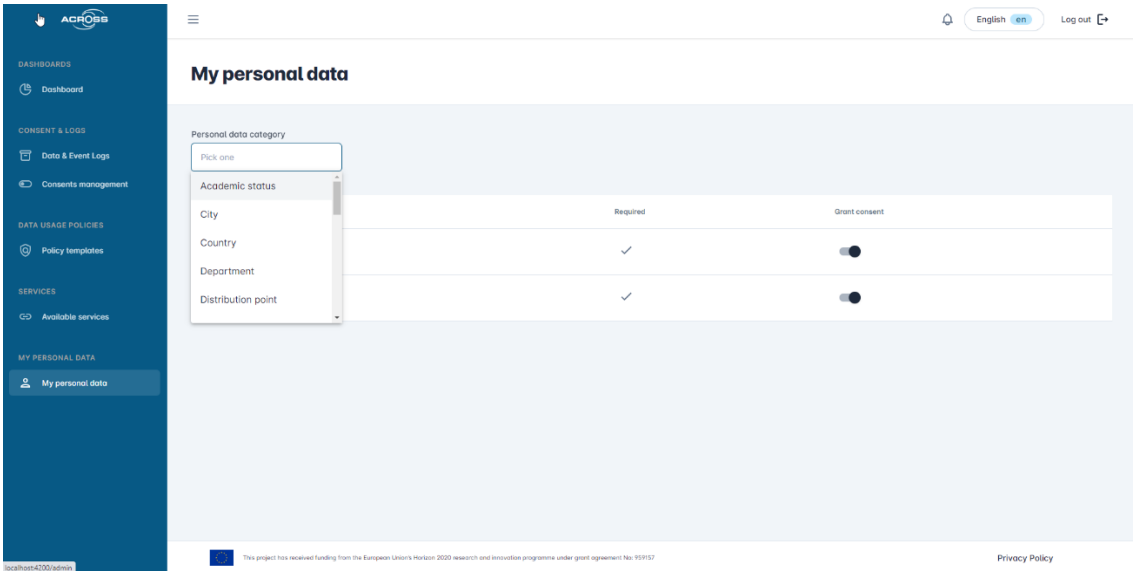


Figure 15 Personal data view interface: selecting a personal data category

3.5.7 Logging

This window provides a list of all the logs stored by the data governance framework.

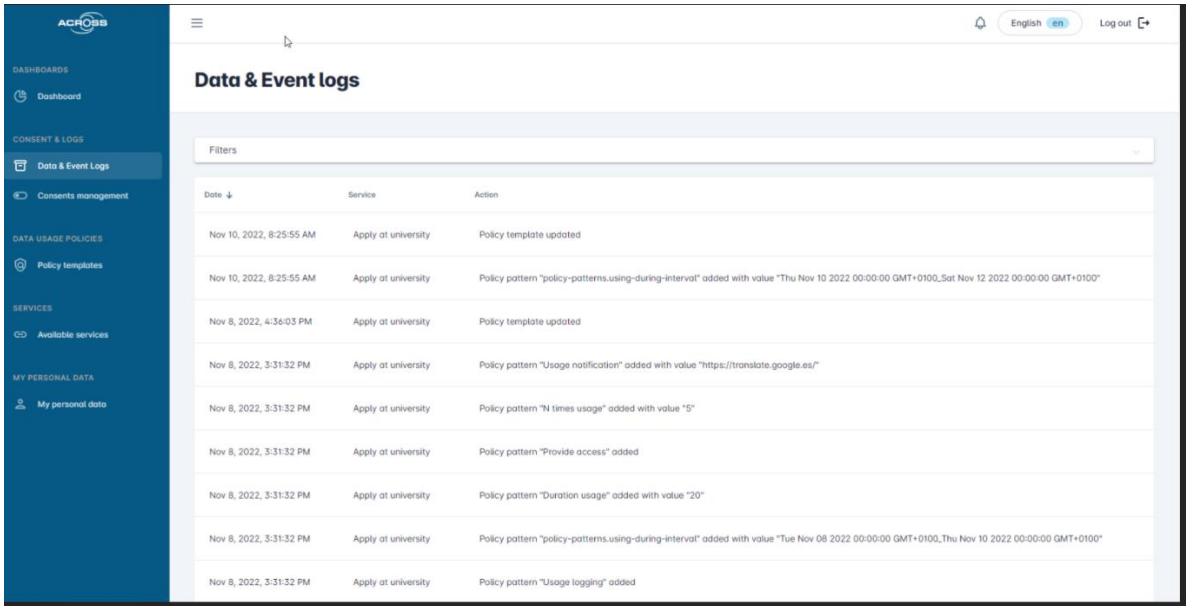


Figure 16 Data & event logs window

3.6 Data Governance framework data models

3.6.1 Service model

The service model is an extension of the CPSV-AP model. The detailed description of this data model is included in WP4.

3.6.2 Consent model

The consent model used includes the following information:

- ServiceId
- UserId
- Personal data permissions for each personal data category needed by the service, both required and optional.

3.6.3 Data usage policy data model

The ACROSS personal data government framework strategy is to minimize the service providers adaptation needed to use the framework and it is not realistic to ask public and private services to use IDS connectors for data transfer. Therefore, the ACROSS Personal Data Governance Framework will assume the responsibility of performing data usage policies management and enforcement.

The User Journey Service Engine will call the framework before transferring the personal data to the service to enforce both personal data consents and data usage policies.

Since ACROSS is not going to use IDS connectors for data transfer, the data usage policies are applied only in the data provider side. Therefore, not real "data usage control" can be applied, only a restricted set of IDS policies providing data access rules. Furthermore, the contract negotiation phase is not needed. Contracts in IDS represent agreements between companies exchanging data and are defined for specific "artifacts" or data sets. In ACROSS the contract represents agreements between end-users and public/private services for using personal data.

Although the contract related information included in the policies is not needed by ACROSS, it has been decided to adapt the ODRL based model defined by IDS, so that this the Data usage module can be used in another contexts, for example in the case of a service that uses IDS connectors to get the personal data.

3.6.3.1 Enforcement in the provider side

Provide Access: This policy gives permission to a specified data consumer to use the data.

```
{
  "@context":{
    "ids":"https://w3id.org/idsa/core/",
    "idsc":"https://w3id.org/idsa/code/"
  },
  "@type":"ids:ContractAgreement",
  "@id":"https://w3id.org/idsa/autogen/contractAgreement/contractAgree1",
  "ids:permission":[
    {
      "@type":"ids:Permission",
      "@id":"https://w3id.org/idsa/autogen/permission/perm1",
      "ids:target":{
        "@id":"https://across.tecnalia.digital.dev/serviceId"

      },
      "ids:title":[
        {
          "@value":"Example Usage Policy Provide Access",
          "@type":"http://www.w3.org/2001/XMLSchema#string"
        }
```

```
        ],
        "ids:description":[
          {
            "@value":"provide-access",
            "@type":"http://www.w3.org/2001/XMLSchema#string"
          }
        ],
        "ids:action":[
          {
            "@id":"idsc:USE"
          }
        ]
      }
    ],
    "ids:provider":{
      "@id":" https://across.tecnalia.digital.dev/ACROSS1" - > ACROSS URL
    },
    "ids:consumer":{
      "@id":"https://across.tecnalia.digital.dev/serviceId"· -> Service URL
    },
    "ids:contractStart":{
      "@value":"2021-02-18T10:15:21.137Z",  -> created On VALOR
      "@type":"http://www.w3.org/2001/XMLSchema#dateTimeStamp"
    }
  }
```


Prohibit Access: prohibits the data usage


```
{
  "@context":{
    "ids":"https://w3id.org/idsa/core/",
    "idsc":"https://w3id.org/idsa/code/"
  },
  "@type" : "ids:ContractAgreement",
  "@id" : "https://w3id.org/idsa/autogen/contractAgreement/contractAgree2",
"ids:prohibition"                    :                    [                    {
"@type" : "ids:Prohibition",
   "@id" : "https://w3id.org/idsa/autogen/permission/perm2",
   "ids:target" : {
        "@id":"https://across.tecnalia.digital.dev/serviceId" -> Service URL
   },
   "ids:title" : [ {
     "@value" : "Example Usage Policy Prohibit Access",
     "@type" : "http://www.w3.org/2001/XMLSchema#string"
   } ],
   "ids:description" : [ {
     "@value" : "prohibit-access",
     "@type" : "http://www.w3.org/2001/XMLSchema#string"
   } ],
   "ids:action" : [ {
```

```
    "@id" : "idsc:USE"
   } ]
  } ],
  "ids:provider":{
    "@id":" https://across.tecnalia.digital.dev/ACROSS1" - > ACROSS URL
  },
  "ids:consumer":{
    "@id":"https://across.tecnalia.digital.dev/serviceId"· -> Service URL
  },
  "ids:contractStart":{
    "@value":"2021-02-18T10:15:21.137Z",  -> created On VALOR
    "@type":"http://www.w3.org/2001/XMLSchema#dateTimeStamp"
  }
}
```

Usage During Interval: provides data usage within a specified time interval (start + end date)

```
{
  "@context":{
    "ids":"https://w3id.org/idsa/core/",
    "idsc":"https://w3id.org/idsa/code/"
  },
  "@type" : "ids:ContractAgreement",
  "@id" : "https://w3id.org/idsa/autogen/contractAgreement/contractAgree3",
  "ids:permission" : [ {
    "@type" : "ids:Permission",
    "@id" : "https://w3id.org/idsa/autogen/permission/perm3",
            "ids:target" : {
          "@id":"https://across.tecnalia.digital.dev/serviceId" -> Service URL
    },
    "ids:title" : [ {
      "@value" : "Example Usage Policy During Intercal",
      "@type" : "http://www.w3.org/2001/XMLSchema#string"
    } ],
    "ids:description" : [ {
      "@value" : "usage-during-interval",
      "@type" : "http://www.w3.org/2001/XMLSchema#string"
    } ],
    "ids:action" : [ {
      "@id" : "idsc:USE"
    } ],
    "ids:constraint" : [ {
      "@type" : "ids:Constraint",
      "@id": "https://w3id.org/idsa/autogen/constraint/0b7c4ca7-1f9e-4e30-8fa1-7551700c1980",
      "ids:rightOperand" : {
        "@value" : "2021-02-11T00:00:00Z",
        "@type" : "xsd:dateTimeStamp"
      },
      "ids:operator" : {
        "@id" : "idsc:AFTER"
```

```
    },
    "ids:leftOperand" : {
      "@id" : "idsc:POLICY_EVALUATION_TIME"
    }
   }, {
    "@type" : "ids:Constraint",
    "@id" : "https://w3id.org/idsa/autogen/constraint/9f2e0197-2ad9-442b-806b-5bb4951a2943",
    "ids:rightOperand" : {
      "@value" : "2022-12-11T00:00:00Z",
      "@type" : "xsd:dateTimeStamp"
    },
    "ids:operator" : {
      "@id" : "idsc:BEFORE"
    },
    "ids:leftOperand" : {
      "@id" : "idsc:POLICY_EVALUATION_TIME"
    }
   } ]
  } ],
  "ids:provider":{
    "@id":" https://across.tecnalia.digital.dev/ACROSS1" - > ACROSS URL
  },
  "ids:consumer":{
    "@id":"https://across.tecnalia.digital.dev/serviceId"· -> Service URL
  },
  "ids:contractStart":{
    "@value":"2021-02-18T10:15:21.137Z",  -> created On VALOR
    "@type":"http://www.w3.org/2001/XMLSchema#dateTimeStamp"
  }
}
```


N Times Usage: This policy restricts the numeric count of using your data by a specified data consume (provider side). The number of times used is updated and consulted via an external end point.


```
{
  "@context":{
    "ids":"https://w3id.org/idsa/core/",
    "idsc":"https://w3id.org/idsa/code/"
  },
  "@type" : "ids:ContractAgreement",
  "@id" : "https://w3id.org/idsa/autogen/contractAgreement/contractAgree4",
  "ids:permission" : [ {
    "@type" : "ids:Permission",
    "@id" : "https://w3id.org/idsa/autogen/permission/perm4",
    "ids:target" : {
          "@id":"https://across.tecnalia.digital.dev/serviceId" -> ServiceId URL
    },
    "ids:title" : [ {
      "@value" : "Example Usage Policy N Times Usage",
```

```
    "@type" : "http://www.w3.org/2001/XMLSchema#string"
   } ],
   "ids:description" : [ {
    "@value" : "n-times-usage",
    "@type" : "http://www.w3.org/2001/XMLSchema#string"
   } ],
   "ids:action" : [ {
    "@id" : "idsc:USE"
   } ],

   "ids:constraint" : [ {
    "@type" : "ids:Constraint",
    "@id":"https://w3id.org/idsa/autogen/constraint/2030a8f2-f03d-4af9-bce5-b9222e129dce",
    "ids:rightOperand" : {
     "@value" : "5",
     "@type" : "xsd:double"
    },
    "ids:operator" : {
     "@id" : "idsc:LTEQ"
    },
    "ids:leftOperand" : {
     "@id" : "idsc:COUNT"
    },
    "ids:pipEndpoint" : {
     "@id" : http://localhost:8080/acrossntec/ACROSSDataUsage/1.0/admin/api/access/ → "Example of
the PIP end point used to get the number of times the service has used the personal data"
    }
   } ]
}],
  "ids:provider":{
    "@id":" https://across.tecnalia.digital.dev/ACROSS1" - > ACROSS URL
  },
  "ids:consumer":{
    "@id":"https://across.tecnalia.digital.dev/serviceId"· -> Service URL
  },
  "ids:contractStart":{
    "@value":"2021-02-18T10:15:21.137Z",  -> created On VALOR
    "@type":"http://www.w3.org/2001/XMLSchema#dateTimeStamp"
  }
}
```


Duration Usage: allows data usage for a specified time period (xsd:duration) . For example, an instantiated policy from this policy class may allow a Data Consumer to use the data for a duration of three months. The permitted period may start from a given date and time.


```
{
  "@context":{
    "ids":"https://w3id.org/idsa/core/",
    "idsc":"https://w3id.org/idsa/code/"
  },
```

```
"@type" : "ids:ContractAgreement",
"@id" : "https://w3id.org/idsa/autogen/contractAgreement/contractAgree5",
"ids:permission" : [ {
 "@type" : "ids:Permission",
 "@id" : "https://w3id.org/idsa/autogen/permission/perm5",
 "ids:target" : {
       "@id":"https://across.tecnalia.digital.dev/serviceId" -> ServiceId URL
 },
 "ids:description" : [ {
  "@value" : "duration-usage",
  "@type" : "http://www.w3.org/2001/XMLSchema#string"
 } ],
 "ids:action" : [ {
  "@id" : "idsc:USE"
 } ],
 "ids:title" : [ {
  "@value" : "Example Usage Policy",
  "@type" : "http://www.w3.org/2001/XMLSchema#string"
 } ],
 "ids:constraint" : [ {
  "@type" : "ids:Constraint",
  "@id" : "https://w3id.org/idsa/autogen/constraint/a5aa4243-432f-4360-aff4-
c95da99eb266",
  "ids:rightOperand" : {
   "@value" : "PT4H",
   "@type" : "xsd:duration"
  },
  "ids:operator" : {
   "@id" : "idsc:SHORTER_EQ"
  },
  "ids:leftOperand" : {
   "@id" : "idsc:ELAPSED_TIME"
  }
 } ]
} ],
 "ids:provider":{
  "@id":" https://across.tecnalia.digital.dev/ACROSS1" - > ACROSS URL
 },
 "ids:consumer":{
  "@id":"https://across.tecnalia.digital.dev/serviceId"· -> Service URL
 },
 "ids:contractStart":{
  "@value":"2021-02-18T10:15:21.137Z",  -> created On VALOR
  "@type":"http://www.w3.org/2001/XMLSchema#dateTimeStamp"
 }
}
```

3.6.3.2 Enforcement in the consumer side

Next data usage policy rules are applicable only in the data consumer side. Therefore, it will not be applicable within the ACROSS platform since the service providers will not deploy IDS connectors for data transfer, only REST interfaces.

Role-restricted Data Usage & ACTION=USE:


```
{
  "@context":{
    "ids":"https://w3id.org/idsa/core/",
    "idsc":"https://w3id.org/idsa/code/"
  },
  "@type" : "ids:ContractAgreement",
  "@id" : "https://w3id.org/idsa/autogen/contractAgreement/contractAgree6",
  "ids:permission" : [ {
    "@type" : "ids:Permission",
    "@id" : "https://w3id.org/idsa/autogen/permission/perm6",
    "ids:target" : {
          "@id":"https://w3id.org/idsa/autogen/artifact/1"-> Service URL
    },
    "ids:description" : [ {
      "@value" : "role-restricted",
      "@type" : "http://www.w3.org/2001/XMLSchema#string"
    } ],
    "ids:action" : [ {
      "@id" : "idsc:USE"
    } ],
    "ids:title" : [ {
      "@value" : "Example Usage Policy Role",
      "@type" : "http://www.w3.org/2001/XMLSchema#string"
    } ],
    "ids:constraint" : [ {
      "@type" : "ids:Constraint",
      "@id" : "https://w3id.org/idsa/autogen/constraint/constraint6",
     "ids:rightOperandReference":{ "@id": "http://example.com/ids-role:riskManager"
},
      "ids:operator" : {       "@id" : "idsc:HAS_MEMBERSHIP"       },
      "ids:leftOperand" : {
        "@id" : "idsc:USER"
      },
      "ids:pipEndpoint" : {
        "@id": "http://localhost:8085/DataUsage/Pip/1.0/admin/api/role/"
      }
    } ]
  } ],
   "ids:provider":{
     "@id":"https://w3id.org/idsa/autogen/baseConnector/provider1" - > ACROSS URL
   },
   "ids:consumer":{
```

```
    "@id":"https://w3id.org/idsa/autogen/baseConnector/consumer1" -> Service URL
  },
  "ids:contractStart":{
    "@value":"2021-02-18T10:15:21.137Z",  -> created On
    "@type":"http://www.w3.org/2001/XMLSchema#dateTimeStamp"
  }
}
```

Purpose restricted data usage: This category represents a class of policy that restricts the usage of data to specific purposes. For example, the next example policy means that "If the purpose is marketing, then allow the usage of data".

```
{
  "@context":{
    "ids":"https://w3id.org/idsa/core/",
    "idsc":"https://w3id.org/idsa/code/"
  },
  "@type" : "ids:ContractAgreement",
  "@id" : "https://w3id.org/idsa/autogen/contractAgreement/contractAgree7",
  "ids:permission" : [ {
    "@type" : "ids:Permission",
    "@id" : "https://w3id.org/idsa/autogen/permission/perm7",
    "ids:target" : {
          "@id":"https://w3id.org/idsa/autogen/artifact/1"-> Service URL
    },
    "ids:description" : [ {
      "@value" : "purpose-restricted",
      "@type" : "http://www.w3.org/2001/XMLSchema#string"
    } ],
    "ids:action" : [ {
      "@id" : "idsc:USE"
    } ],
    "ids:title" : [ {
      "@value" : "Example Usage Policy Purpose",
      "@type" : "http://www.w3.org/2001/XMLSchema#string"
    } ],
    "ids:constraint" : [ {
      "@type" : "ids:Constraint",
      "@id" : "https://w3id.org/idsa/autogen/constraint/constraint7",

     "ids:rightOperandReference":{
"@id": "http://example.com/ids-purpose:Marketing"
},
      "ids:operator" : {
        "@id" :  "idsc:SAME_AS"
      },
      "ids:leftOperand" : {
        "@id" : "idsc:PURPOSE"
```

```
    },
    "ids:pipEndpoint" : {
      "@id": "http://localhost:8085/DataUsage/Pip/1.0/admin/api/purpose/"
    }
   } ]
  } ],
  "ids:provider":{
    "@id":"https://w3id.org/idsa/autogen/baseConnector/provider1" → ACROSS URL
  },
  "ids:consumer":{
    "@id":"https://w3id.org/idsa/autogen/baseConnector/consumer1" → Service URL
  },
  "ids:contractStart":{
    "@value":"2021-02-18T10:15:21.137Z", → created On
    "@type":"http://www.w3.org/2001/XMLSchema#dateTimeStamp"
  }
}
```

4 Conclusions and next steps

This section presents some conclusions gathered from the data governance framework design and its implementation. The concept of Personal Data Governance framework defined in ACROSS is perfectly aligned with several initiatives in the field of Personal and Private data management, including:

- **MyData initiative**: MyData operator
- **Data Governance Act**: Personal data sharing intermediary
- **Tech Dispatch published by the European Data Protection Supervisor**: Personal Information Management System (PIMS).

The ACROSS Personal Data Governance Framework is part of the so-called ACROSS platform, useful in the context of cross-border public/private services for the citizens. However, it has been designed to be an independent framework that can be used also for the individuals to manage their personal data according to the GDPR in any other context.

The ACROSS data governance framework does not cover the following functionalities included in most of the initiatives, although it could integrate them:

- Secure Data Storage (e.g., Personal wallet)
- Secure Data transfer among services (e.g., IDS connector)

ACROSS has extended the MyData operator concept with the Data usage policies management and enforcement functionalities, allowing the end user to define a more fine-grained control of the personal data being used by services.

Even though this deliverable contains the final version of the design, some of the information included will be a subject to refinements and modifications, mainly due to the changes in the user interface to improve the usability. These changes will be gathered in the next deliverables regarding the Data governance framework implementations.

5 References

[1] D3.1 Design of the ACROSS Data Governance framework for data sovereignty - Initial

[2] D3.3 Implementation of the ACROSS Data Governance framework for data sovereignty – Initial

[3] D2.4 Report for cross-border service gap analysis – Final

[4] D6.2 Use case evaluation and impact assessment – Initial

[5] D5.2 System Architecture & Implementation Plan – Final