



D2.5

Data Sharing and Sovereignty Mechanisms

CERTH

D2.5

Data Sharing and Sovereignty Mechanisms

Revision **v1.0**

Work package	WP2
Task	T2.1, T2.2, T2.3, T2.4
Due date	30-09-2025
Submission date	24-09-2025
Deliverable lead	CERTH
Version	v1.0
Authors	Konstantinos Votis (CERTH), Vasileios Siopidis (CERTH), Nikolaos Tepelidis (CERTH), Charalampos Tsakiridis (CERTH), Efthymios Chondrogiannis (ICCS), Efstathios Karanastasis (ICCS), Raul Palma (PSNC), Engfer Sebastian (FHG), Panagiotis Kyziropoulos (UCC), Gorka Zarate (TEC), Urtza Iturraspe (TEC), Valentin Sanchez (TEC), Joel Himanen (1001 LAKES)
Reviewers	Federico Pinuela (GIDI), Engfer Sebastian (FHG)

Abstract

This deliverable outlines the final development status of the data sharing and data sovereignty mechanisms. It includes references to the repository, open API specifications, online documentation, and other resources for each component related to WP2.

Keywords

Data Sharing, Data Sovereignty, repository, API, software

Document revision history

Version	Date	Description of change	Contributor(s)
v0.1	18-07-2025	Deliverable template	Vasileios Siopidis (CERTH)
v0.2	10-09-2025	Content from authors	Deliverable authors
v0.3	21-09-2025	Internal review	Sebastian Engfer (FHG), Federico Piñuela (GIDI)
v1	24-09-2025	Final version	Vasileios Siopidis (CERTH) & Santiago Cáceres (ITI)

Disclaimer

The information, documentation and figures available in this deliverable are provided by the DATAMITE project's consortium under EC grant agreement **101092989** and do not necessarily reflect the views of the European Commission. The European Commission is not liable for any use that may be made of the information contained herein.

Copyright notice

© DATAMITE 2023-2025

Project co-funded by the European Commission in the Horizon Europe Programme

Nature of the deliverable

OTHER

Dissemination level

PU Public, fully open. e.g., website

✓

CL Classified information as referred to in Commission Decision 2001/844/EC

SEN Confidential to DATAMITE project and Commission Services

* Deliverable types:

R: document, report (excluding periodic and final reports).

DEM: demonstrator, pilot, prototype, plan designs.

DEC: websites, patent filings, press and media actions, videos, etc.

OTHER: software, technical diagrams, etc.

Table of contents

1	Introduction.....	6
1.1	Deliverable Purpose and Scope.....	6
1.2	Document Structure	6
1.3	Document Dependencies	7
2	Data Sharing Module	8
2.1	IDSA Gaia-X Connectors and Components.....	8
2.1.1	Gaia-X Data Product on Boarding CES Publishing.....	8
2.1.2	Logging service extension	9
2.2	Eclipse Dataspace Connector.....	11
2.3	Brokers to EU Portals.....	13
2.4	Data Product Composition	16
2.5	Logging.....	17
3	Data Security Module.....	20
3.1	Policy Library	20
3.2	Policy Engine.....	22
4	Data Support Tools	26
4.1	Data Harmonisation	26
4.1.1	Data Harmonisation Tools	26
4.1.2	Data Harmonisation Pipelines	27

List of figures

Figure 1. SWAGGER Documentation Gaia-X Data Product on Boarding Ces Publishing	9
Figure 2. Architecture diagram for Logging service extension module	10
Figure 3. Connector Base Architecture and Communication Flow	11
Figure 4. Control plane sequence for requesting data	12
Figure 5. Data plane signaling sequence	12
Figure 6. Component setup of the consumer/provider docker-demonstrator-deployment	13
Figure 7. System Context Diagram	13
Figure 8. Component Diagram	14
Figure 9. Flow Diagram	15
Figure 10. The workflow of data product composition tool.....	16
Figure 11. SWAGGER Documentation for Data Product Composition	17
Figure 12. The workflow of logging tool	18
Figure 13. SWAGGER Documentation for logging tool	19
Figure 14. High-level overview of DATAMITE policy management	21
Figure 15. The architecture of Policy Library backend service	22
Figure 16. Time Interval Usage: Allows data usage for a specific time interval.....	24
Figure 17. Location Usage: Allows data usage for a specific location	25
Figure 18. Purpose Usage: Allows data usage for specific purposes.....	25
Figure 19. Data Harmonisation Tools – Web Application – Home Page	26
Figure 20. Data Harmonisation Pipelines Service architecture.....	28
Figure 21. SWAGGER Documentation for Data Harmonization Pipelines	29

1 Introduction

DATAMITE is a project funded by the European Commission as part of the Horizon Europe program and coordinated by the ITI - Technological Institute of Informatics. DATAMITE empowers European companies by delivering a modular, open-source and multi-domain Framework to improve DATA Monetising, Interoperability, Trading and Exchange in the form of software modules, training, and business materials.

DATAMITE unleashes the monetisation potential at two levels. At the internal level, users will have tools to improve the quality management of their data, the adherence to FAIR principles, and will be able to upskill on technical and business aspects, thanks to the multiple open-source training materials the project will generate. Therefore, data will become trustworthy and more reliable also in other paradigms like AI.

At the external level, keeping users in control of their data will provide new sources of revenue and interaction with other stakeholders. The architecture envisioned for DATAMITE enables DIHs' sandboxing, becoming a potential instructor in their onboarding of SMEs and low-tech SMEs into the data economy. Together, DATAMITE's solutions will function as a catalyst to boost data monetisation in the European productive fabric.

1.1 Deliverable Purpose and Scope

Specifically, the Grant Agreement states the following regarding this Deliverable:

“Describes the status of development, the applied technologies, the communication channels, and encountered issues and obstacles”.

Hence, the purpose of this document is to present the final development status of the various components related to WP2. For each component, this document provides references to the DATAMITE GitLab repository, which includes the code, license, and online documentation. An overview of the current design, implementation status, and next steps for each component has already been described in *D2.4 - Data Sharing and Sovereignty Mechanisms Report (M33)*.

1.2 Document Structure

This deliverable is broken down into the following sections:

- **Section 1 Introduction** provides the deliverable's general context, dependencies, and structure.

- **Section 2 Data Sharing Module, Section 3 Data Security Module, and Section 4 Data Support tools** provide references to the code, open API specifications, online documentation, demos, and other resources for relevant projects in the DATAMITE GitLab repository.

1.3 Document Dependencies

This document is the final instalment in a series of deliverables. It presents the completed development status of the components and provides the relevant references to the DATAMITE repository.

2 Data Sharing Module

The Data Sharing module has published software for the following components:

- IDSA Gaia-X Connectors and Components
- Eclipse Dataspace Connector
- Broker to EU Portals
- Data Product Composition
- Logging

Links to code and documentation are provided in the following subsections.

2.1 IDSA Gaia-X Connectors and Components

To participate in a Gaia-X-based data space, several steps must be followed, starting with obtaining a valid certificate recognised by the Gaia-X Trust Anchor. It is also necessary to generate a DID (Decentralized Identifier) document and a verifiable credential for the legal participant, ultimately enabling the description and offering of data products. The developed API allows all these actions to be carried out.

2.1.1 Gaia-X Data Product on Boarding CES Publishing

The following resources are available:

- Repository: [Gaia-X Data Product Repository](#)
- Online Documentation: [Gaia-X Data Product Documentation](#)
- Open API specification: [Gaia-X Data Product Open API specification](#)
- User manual: [Gaia-X Data Product User Manual](#)

The component is being developed by TECNALIA and is available through [MIT license](#).

As can be seen, the API is divided into three distinct components:

- **DATAMITE API:** an independent API that operates even outside the framework, allowing a company to generate its participant credentials and even define and publish its data products.
- **DATAMITE Internal API:** API that has been particularly useful during development for performing faster checks and validations.

- **DATAMITE Internal Data Product:** API directly linked to the framework, enabling an internal product to be published as a Gaia-X compliant data product.

The following images are illustrative screenshots of the component.

Datamite API		^
POST	/diddocument Create Did	▼
POST	/legalparticipant Createlegalparticipant	▼
GET	/dataproducs Getdataproducslist	▼
POST	/dataproducs Createdataproducswithresources	▼
POST	/publish-dataproduct Createandpublishdataproducs	▼
GET	/dataproducs/{dataProductNme} Getdataproducsbyname	▼
GET	/cesrecords Getcesrecords	▼
Datamite testing API		^
POST	/compliance Calltoconpliance	▼
POST	/credential-event-service Calltocredentialeventservice	▼
POST	/sign-vc Signvc	▼
POST	/sign-vp Signvp	▼
POST	/loadNeo4j Loadneo4j	▼
Datamite internal data product		^
GET	/export-dataproduct-composition-to-gaiax ExporttoGaiaXmodel	▼
POST	/publish-dataproduct-internal Createandpublishdataproducs	▼

Figure 1. SWAGGER Documentation Gaia-X Data Product on Boarding Ces Publishing

2.1.2 Logging service extension

The following resources are available:

- Repository: [Logging Service Extension Repository](#)
- Online Documentation: [Logging Service Extension Documentation](#)

The **Logging Service Extension** captures and processes events generated in the different EDC components. These events include asset lifecycle changes (*AssetCreated*, *AssetUpdated*, *AssetDeleted*), policy definition updates (*PolicyDefinitionCreated*, *PolicyDefinitionUpdated*, *PolicyDefinitionDeleted*), and contract definition modifications (*ContractDefinitionCreated*, *ContractDefinitionUpdated*, *ContractDefinitionDeleted*). Any unrecognised events are also logged for monitoring purposes.

Once processed, the extension forwards the event data to the **DATAMITE Logging API**, available at: <https://datamite.api.iti.gr/docs/logging#/EDC/>.

The component is being developed by TECNALIA and is available through [MIT license](#).

The following images are illustrative screenshots of the component.

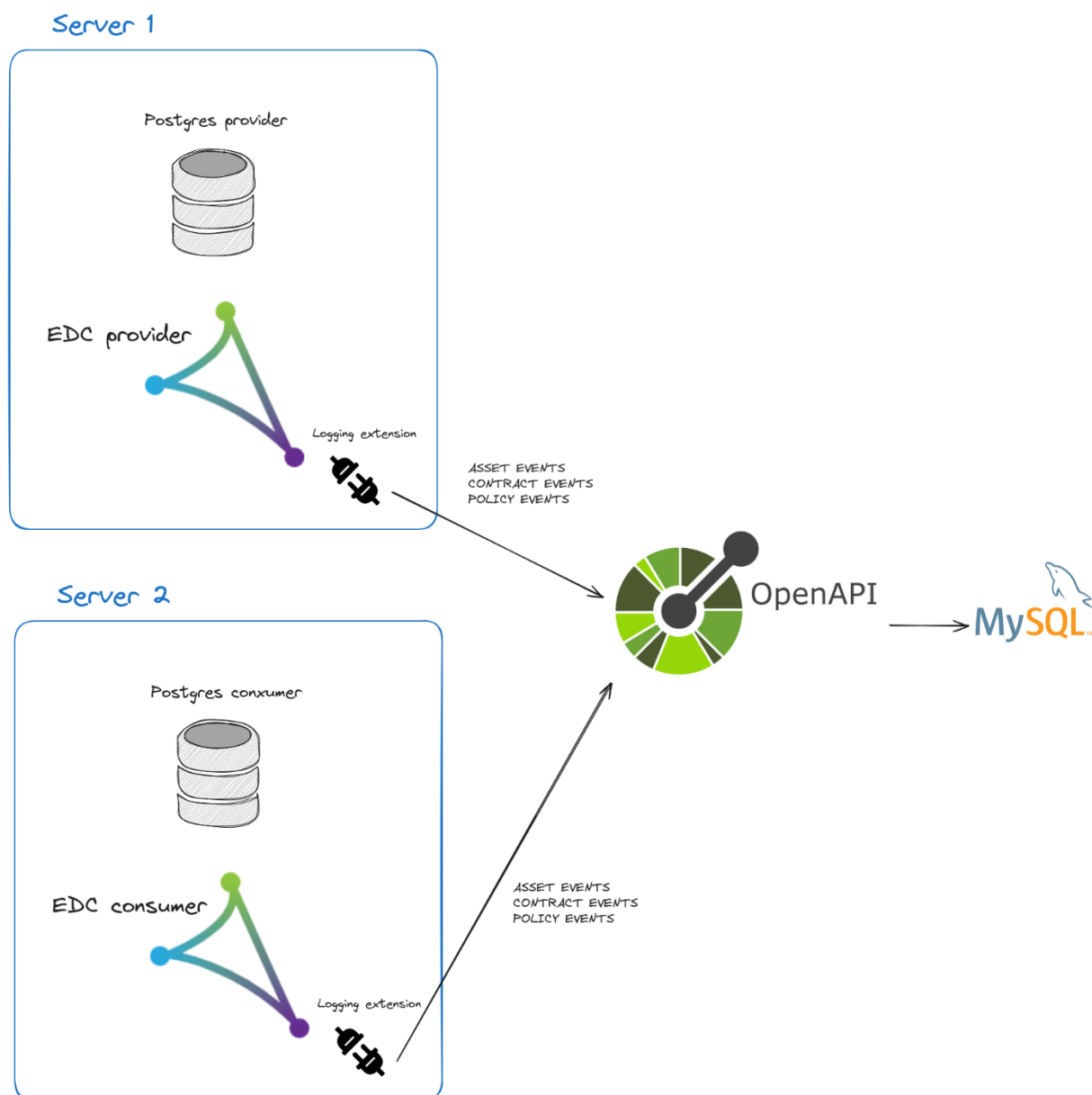


Figure 2. Architecture diagram for Logging service extension module

2.2 Eclipse Dataspace Connector

The following resources are available:

- Repository: [EDC Repository](#)
- Online Documentation: [EDC Documentation](#)
- Open API specification: [EDC Open API specification](#)
- User manual: [EDC manual](#)

The component is being developed by Fraunhofer and is available through [MIT license](#).

The following images are illustrative screenshots of the component.

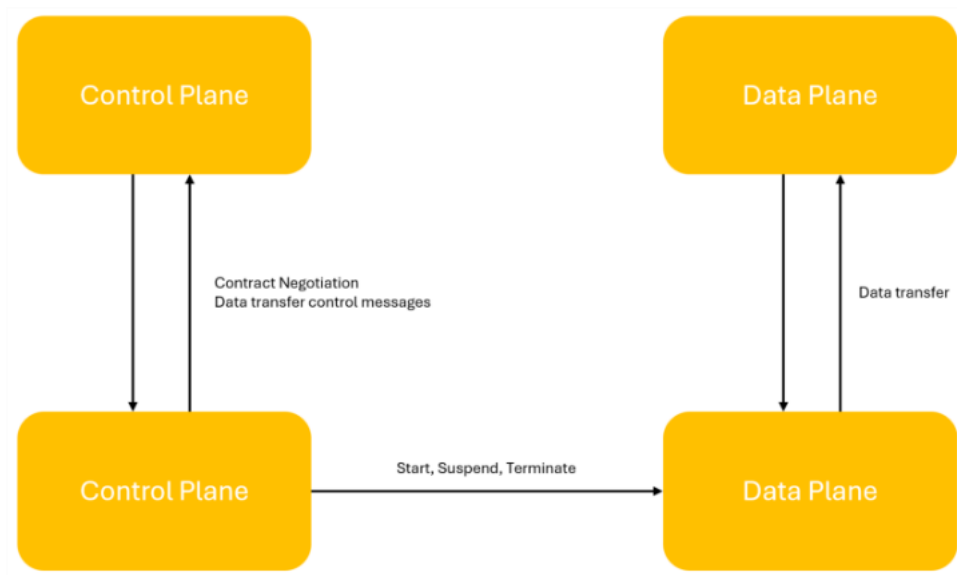


Figure 3. Connector Base Architecture and Communication Flow

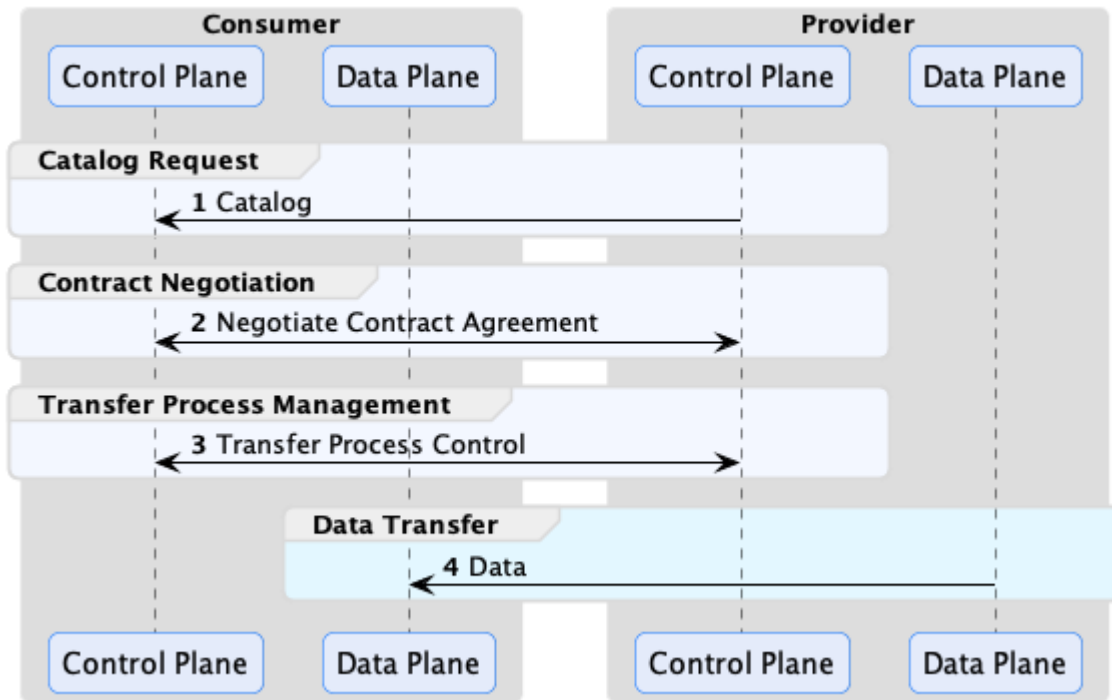


Figure 4. Control plane sequence for requesting data

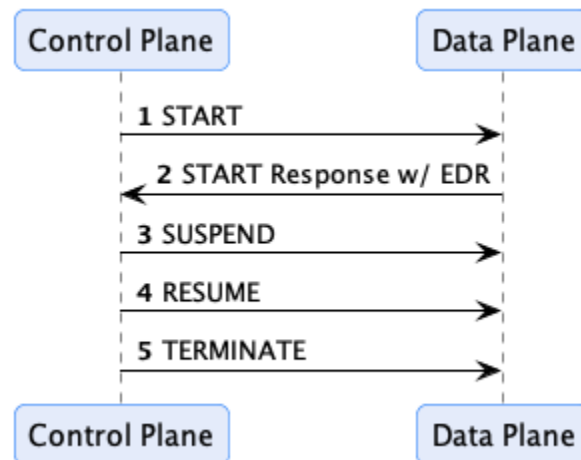


Figure 5. Data plane signaling sequence

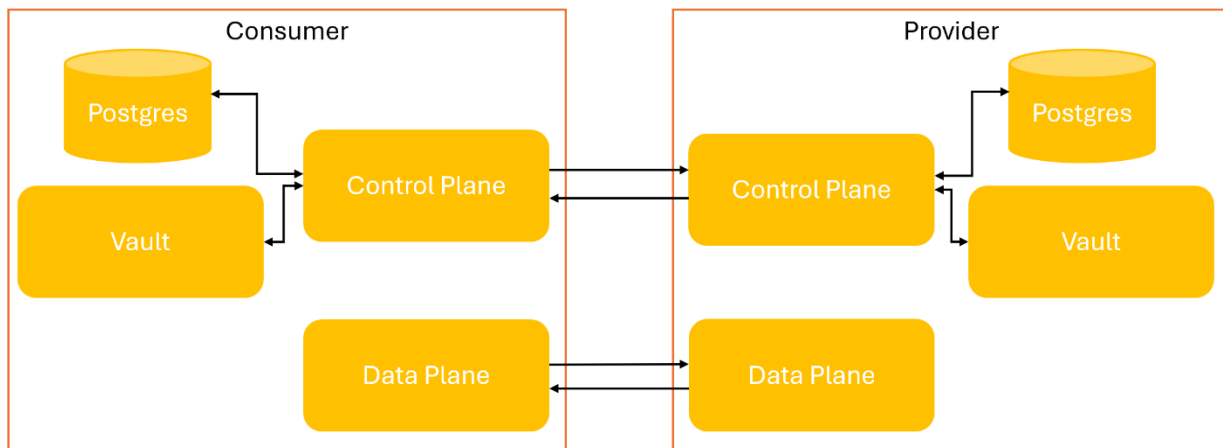


Figure 6. Component setup of the consumer/provider docker-demonstrator-deployment

2.3 Brokers to EU Portals

The following resources are available:

- Repository: [Broker to EU Portals Repository](#)
- Online Documentation: Broker to EU [Portals](#) Documentation
- Open API specification: Broker to EU [Portals](#) API Specification
- User manual: [Broker user manual](#)
- Demo: [Demo](#)

The component is being developed by UCC and is available through [MIT license](#).

The following images are illustrative screenshots of the component.

System Context Diagram

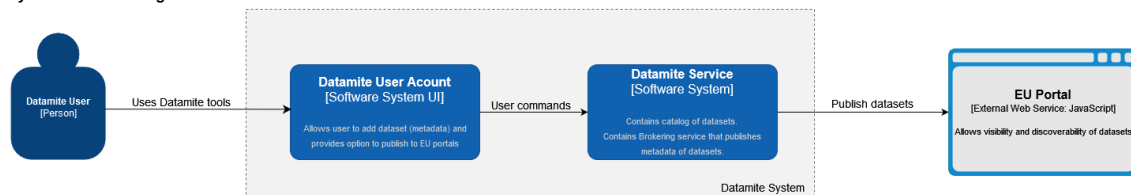


Figure 7. System Context Diagram



Component Diagram

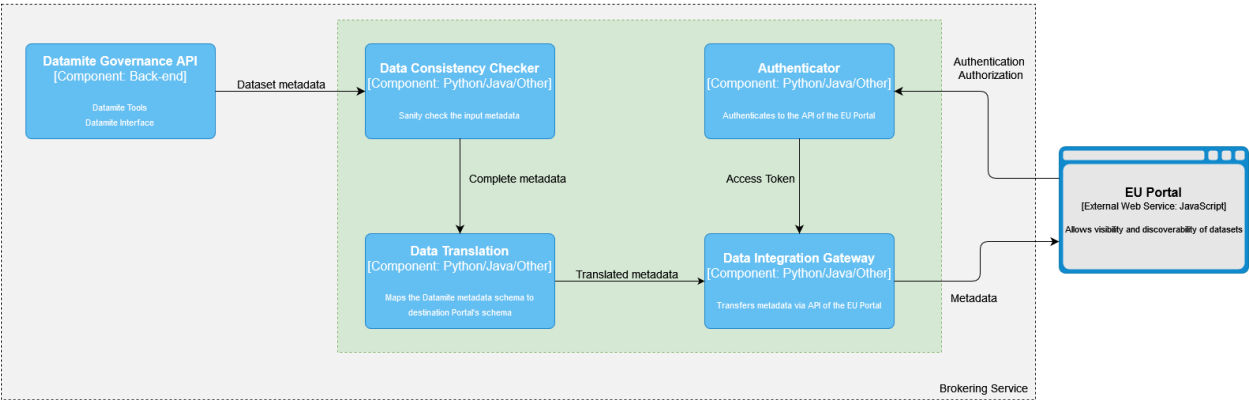


Figure 8. Component Diagram

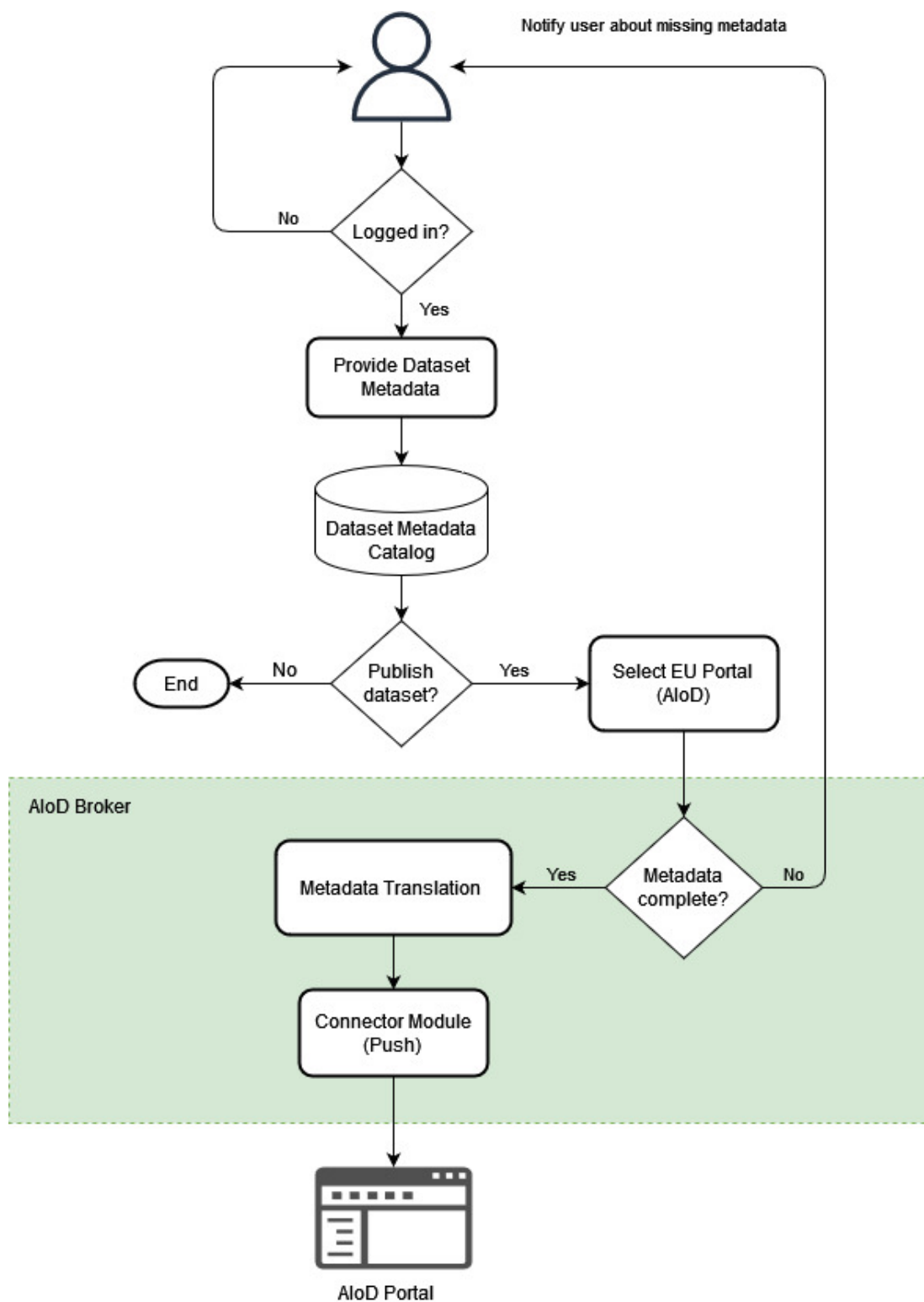


Figure 9. Flow Diagram

2.4 Data Product Composition

The following resources are available:

- Repository: [Data Product Composition Repository](#)
- Online Documentation: [Data Product Composition Documentation](#)
- Open API specification: [Data Product Composition API](#)
- User manual: [Data Product Readthedocs Tool](#)
- Demo: [Demo Video](#)

The component is being developed by CERTH and is available through [MIT license](#).

The following images are illustrative screenshots of the component.

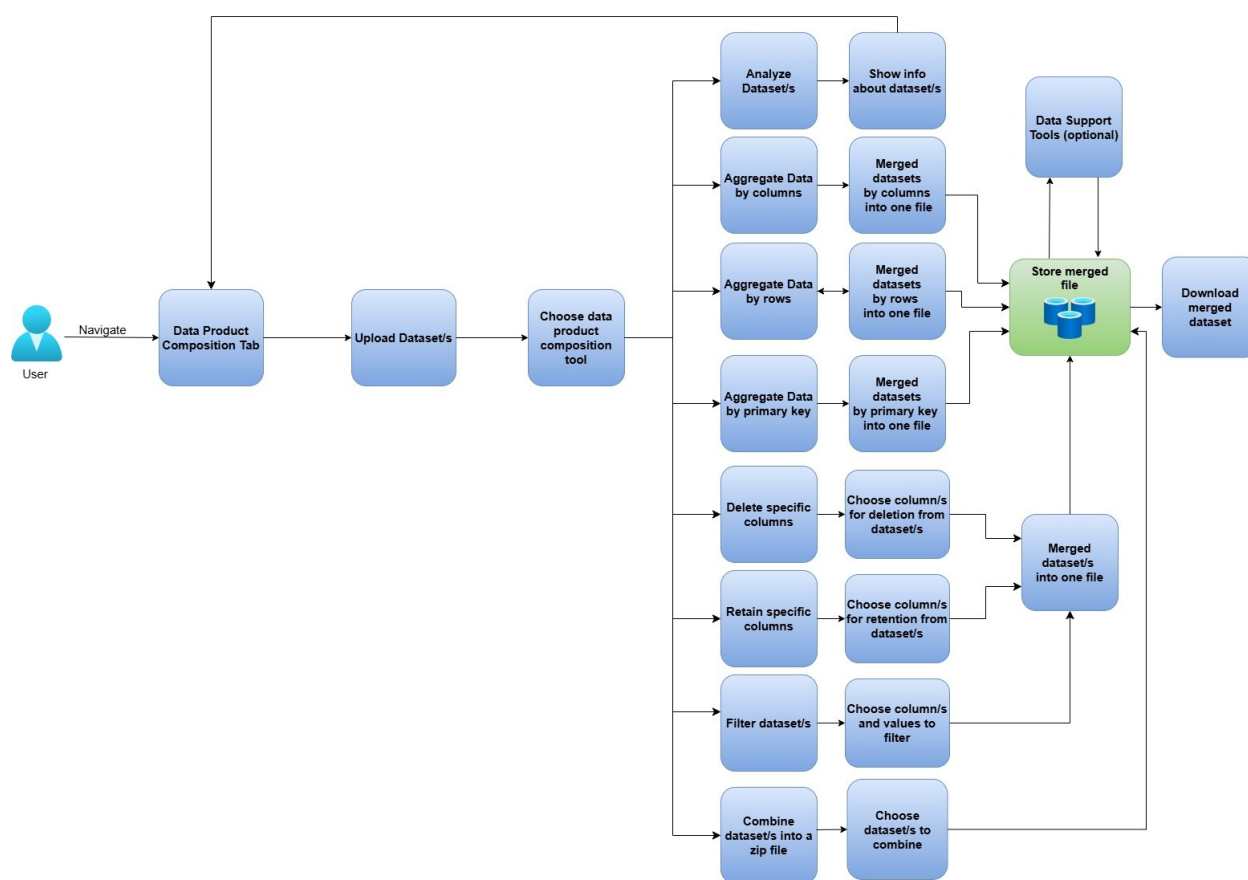


Figure 10. The workflow of data product composition tool

Load data product Everything about load process of data product		^
POST	/upload_dataset Upload dataset(s) (csv, xlsx and json formats support)	▼
GET	/load_data_product Get and load data product by unique ID or file name	▼
Retrieve data product Everything about retrieve process of data product		^
GET	/retrieve_data_product Get data product by unique ID or file name	▼
Data Product Info Analyze dataset/s		^
POST	/analyze_datasets Analyze datasets	▼
Aggregate data by primary key Merge datasets by primary key		^
GET	/show_datasets Show imported and filtered (if filter module was used) files with columns	▼
POST	/merge_datasets_byPrimaryKey Merge datasets by primary key	▼
GET	/download_merged_dataset_byPrimaryKey Download merged dataset	▼
Aggregate data by columns Merge datasets by columns		^
GET	/show_datasets Show imported and filtered (if filter module was used) files with columns	▼
POST	/merge_datasets_bycolumns Merge datasets by columns	▼
GET	/download_merged_dataset_bycolumns Download merged dataset	▼
Aggregate data by rows Merge datasets by rows		^
GET	/show_datasets Show imported and filtered (if filter module was used) files with columns	▼
POST	/merge_datasets_byrows Merge datasets by rows	▼
GET	/download_merged_dataset_byrows Download merged dataset	▼
Delete specific columns Delete columns from one or multiple datasets		^
GET	/get_all_datasets Show all files and the columns	▼
POST	/delete_columns Delete specific columns	▼
GET	/download_deleted_dataset Download deleted dataset	▼

Figure 11. SWAGGER Documentation for Data Product Composition

2.5 Logging

The following resources are available:

- Repository: [Logging Repository](#)
- Online Documentation: [Logging Documentation](#)
- Open API specification: [Logging Open API specification](#)
- User manual: [Logging User manual](#)
- Demo: [Demo Video](#)

The component is being developed by CERTH and is available through [MIT license](#).

The following images are illustrative screenshots of the component.

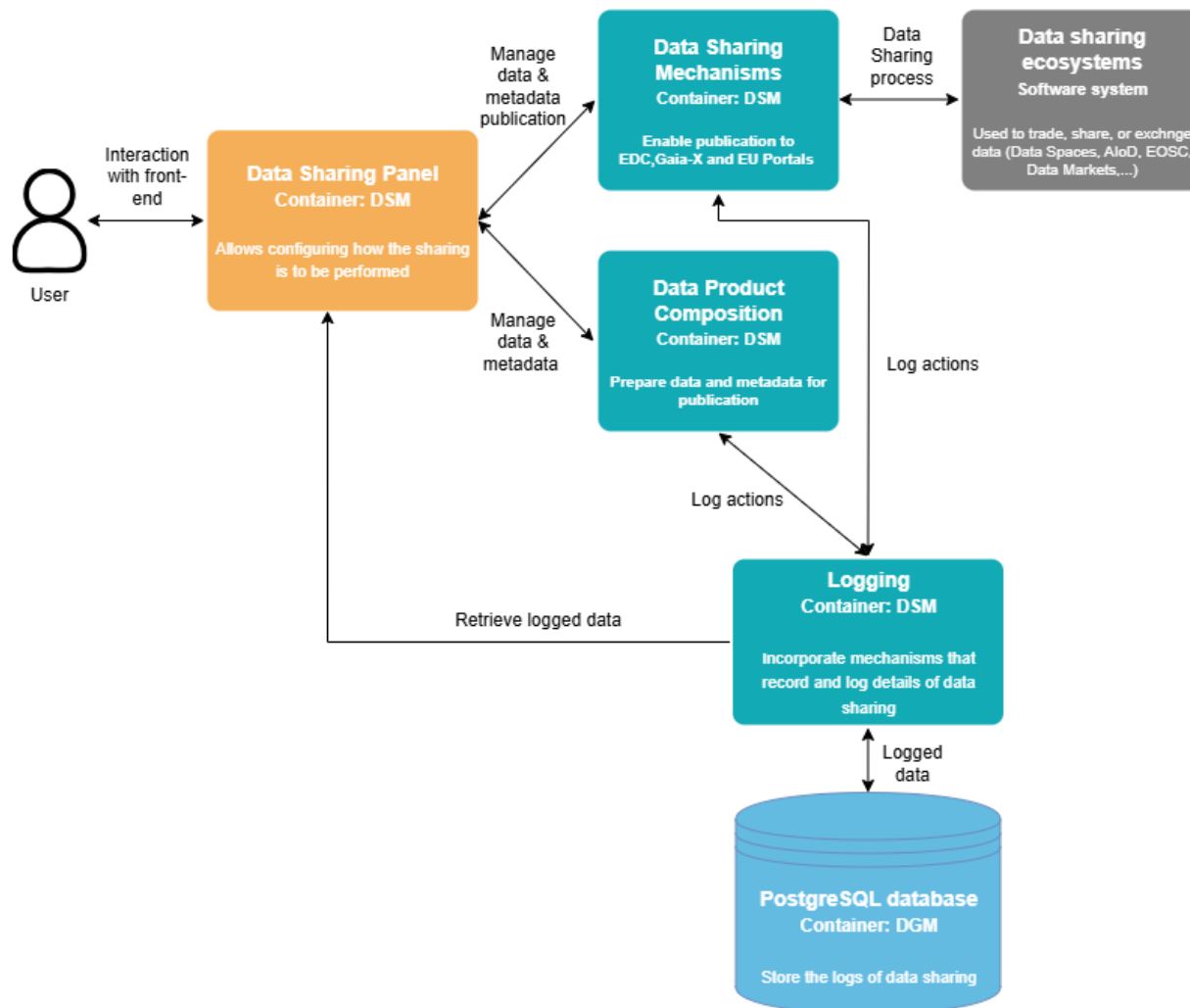


Figure 12. The workflow of logging tool



Data Product Composition <small>Operations related to Data Product Composition</small>			^
POST	/dataProductLog/create	To log information when a data product is created	v
POST	/dataProductLog/update	To log information when a data product is updated	v
POST	/dataProductLog/delete	To log information when a data product is deleted	v
POST	/dataProductLog/updateMetadata	To log information when metadata are updated	v
POST	/dataProductLog/anonymization	To log information when a data product is anonymized	v
POST	/dataProductLog/download	To log information when a data product is downloaded	v
GET	/dataProductLog/getall	Returns all the logs about Data Product composition actions	v
GET	/dataProductLog/searchWithFilters	Filters the Data Product Composition logs by the below fields, empty the values from fields not desired, each filter is applied with and operator	v
GET	/dataProductLog/searchHistory	Returns the history of a selected Data Product	v
GET	/dataProductLog/searchHistoryWithFilters	Filters the history of Data Product Composition logs by the below fields, empty the values from fields not desired, each filter is applied with and operator	v
Publications <small>Operations related to publications</small>			^
GET	/publicationsLog/getall	Returns all the logs about publications	v
GET	/publicationsLog/searchWithFilters	Filters the publications logs by the below fields, empty the values from fields not desired, each filter is applied with and operator	v
GET	/publicationsLog/searchHistory	Returns the history of publications	v
GET	/publicationsLog/searchHistoryWithFilters	Filters the publications history by the below fields, empty the values from fields not desired, each filter is applied with and operator	v

Figure 13. SWAGGER Documentation for logging tool

3 Data Security Module

The Data Security module has published software for the following components:

- Policy Library
- Policy Engine

Links to code and documentation are provided in the following subsections.

3.1 Policy Library

The following resources are available:

- Repository: [Policy Library Repository](#)
- Online Documentation: [Policy Library Documentation](#)
- Open API specification: [Policy Library Open API specification](#)

The component is being developed by 1001 LAKES and is available through the [MIT license](#).

The following images are illustrative diagrams of the component.

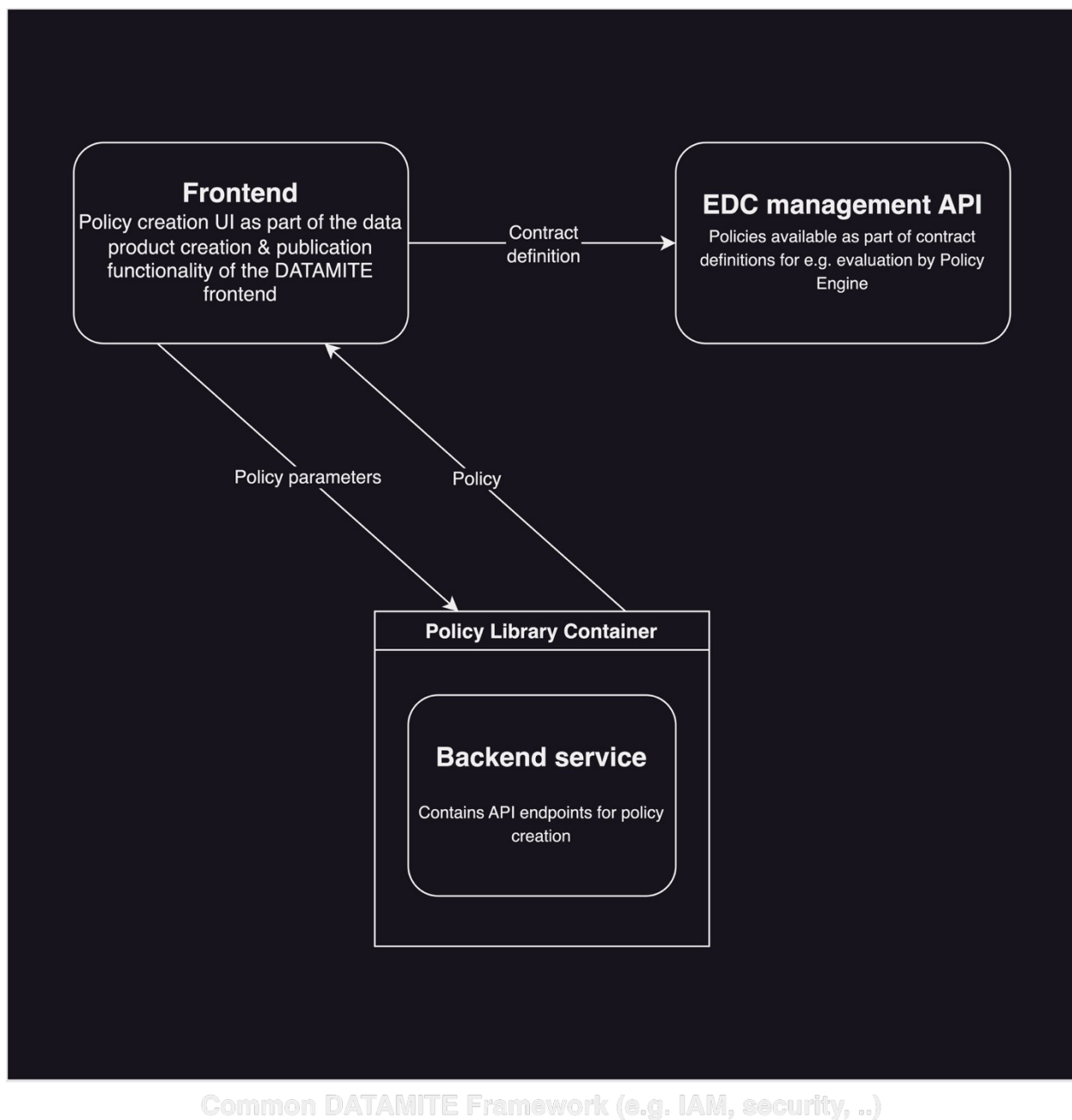


Figure 14. High-level overview of DATAMITE policy management

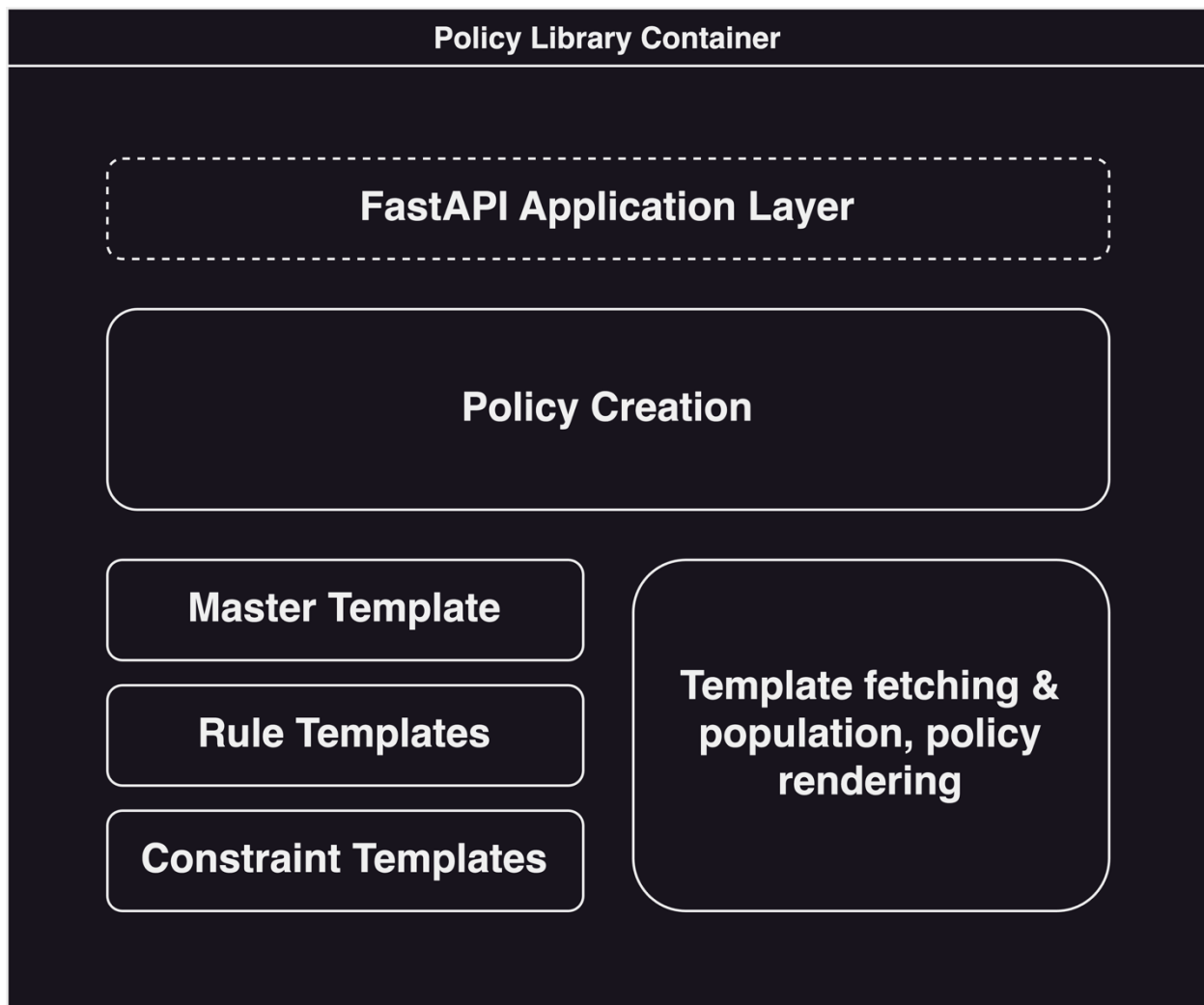


Figure 15. The architecture of Policy Library backend service

3.2 Policy Engine

The **Policy Engine** is an extension integrated within the EDC, designed to verify compliance with established policies. If the conditions are met, the EDC proceeds with the data transfer; otherwise, the transfer is blocked.

Three types of policies have been defined:

- **Time Interval:** This policy checks, using the system's date and time, whether the asset can be transferred within the time window specified in the policy.

- **Location and Purpose:** These two policies rely on Keycloak, where the user's location and intended purpose are defined. The policy compares these attributes to determine if they match the specified conditions.

For these last two policies, new operators have been introduced in addition to the existing EQ and NEQ, including: `hasPart`, `isNoneOf`, `isPartOf`, and `isAnyOf`.

The following resources are available:

- Repository: [Policy Engine Repository](#)
- Online Documentation: [Policy Engine Documentation](#)
- User manual: [Policy Engine user manual](#)

The component is being developed by TECNALIA and is available through [MIT license](#).

The following images are illustrative screenshots of the component.

```
{
  "@context": {
    "@vocab": "https://w3id.org/edc/v0.0.1/ns/"
  },
  "@id": "aPolicyTimeInterval",
  "@type": "edc:PolicyDefinition",
  "policy": {
    "@context": "http://www.w3.org/ns/odrl:jsonld",
    "@type": "Set",
    "permission": [
      {
        "action": "use",
        "constraint": [
          {
            "@type": "AtomicConstraint",
            "leftOperand": "https://w3id.org/edc/v0.0.1/ns/timeInterval",
            "rightOperand": {
              "@type": "xsd:date",
              "@value": "2022-12-31T23:00:00.000Z"
            },
            "operator": {
              "@id": "odrl:gteq"
            }
          },
          {
            "@type": "AtomicConstraint",
            "leftOperand": "https://w3id.org/edc/v0.0.1/ns/timeInterval",
            "rightOperand": {
              "@type": "xsd:date",
              "@value": "2024-09-30T23:00:00.000Z"
            },
            "operator": {
              "@id": "odrl:lteq"
            }
          }
        ]
      }
    ],
    "prohibition": [],
    "obligation": []
  }
}
```

Figure 16. Time Interval Usage: Allows data usage for a specific time interval


```
{
  "@context": {
    "@vocab": "https://w3id.org/edc/v0.0.1/ns/"
  },
  "@id": "aPolicyRegion",
  "@type": "edc:PolicyDefinition",
  "policy": {
    "@context": "http://www.w3.org/ns/odrl.jsonld",
    "@type": "Set",
    "permission": [
      {
        "action": "use",
        "constraint": {
          "@type": "AtomicConstraint",
          "leftOperand": "https://w3id.org/edc/v0.0.1/ns/regionLocation",
          "operator": "odrl:eq",
          "rightOperand": "eu"
        }
      }
    ]
  }
}
```

Figure 17. Location Usage: Allows data usage for a specific location

```
{
  "@context": {
    "@vocab": "https://w3id.org/edc/v0.0.1/ns/"
  },
  "@id": "policy_id",
  "@type": "edc:PolicyDefinition",
  "policy": {
    "@context": "http://www.w3.org/ns/odrl.jsonld",
    "@type": "Set",
    "permission": [
      {
        "action": "use",
        "constraint": {
          "@type": "AtomicConstraint",
          "odrl:leftOperand": "https://w3id.org/edc/v0.0.1/ns/purpose",
          "rightOperand": {
            "@type": "xsd:string",
            "@value": "marketing"
          },
          "operator": {
            "@id": "odrl:eq"
          }
        }
      }
    ]
  }
}
```

Figure 18. Purpose Usage: Allows data usage for specific purposes

4 Data Support Tools

Links to code and documentation for the Data Harmonisation are provided.

4.1 Data Harmonisation

4.1.1 Data Harmonisation Tools

The three software tools developed for Data Harmonisation have been integrated into a Web Application, which can be used either as a stand-alone tool or in conjunction with the other components of the DATAMITE platform, as presented in D2.4.

The developed GUI (Figure 19) guides the user through the Data Harmonisation process by enabling them to: (i) provide the existing dataset (either by uploading the corresponding file or selecting one from those already available in the DATAMITE repositories), (ii) specify mappings to the elements of a predefined data model (the target model), and (iii) use these mappings for data transformation.

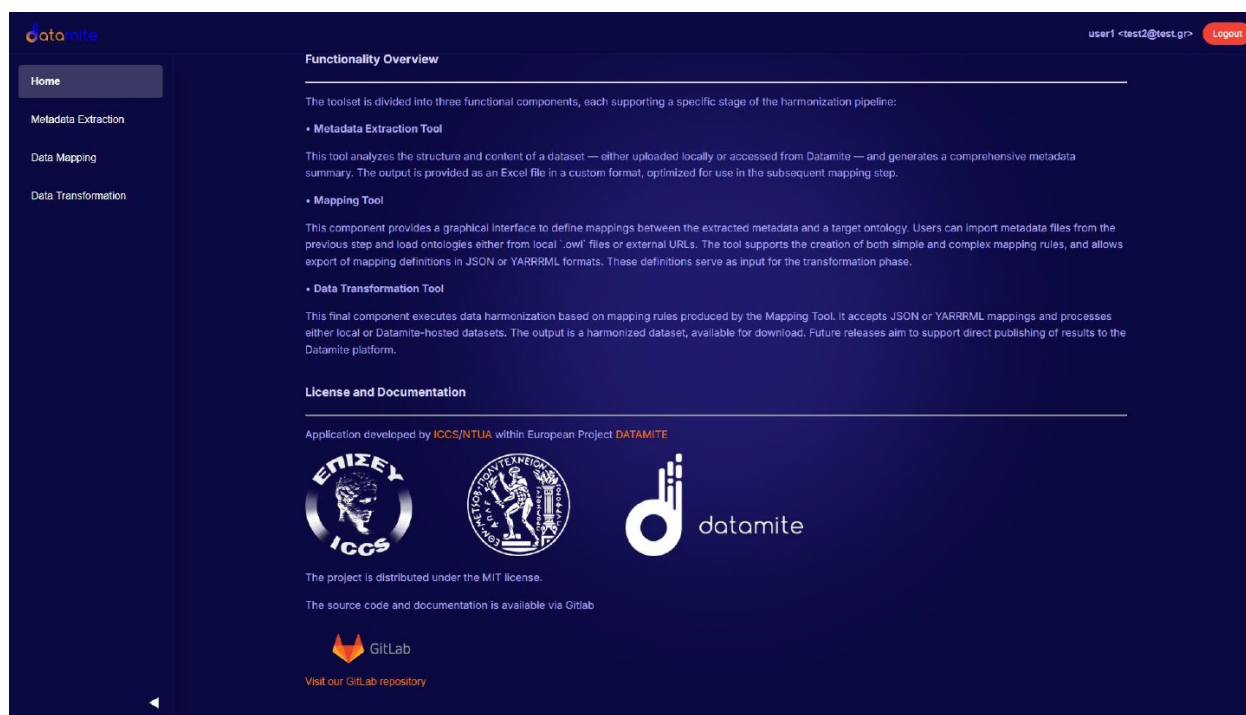


Figure 19. Data Harmonisation Tools – Web Application – Home Page

The following resources are available:

- Repository: [Data Harmonisation Tools Repository](#)
- Online Documentation: [Data Harmonisation Tools Documentation](#)
- Open API specification: It has its own GUI that interacts DATAMITE REST services
- User manual: [Data Harmonisation Tools Read the Docs](#)
- Demo: [Demo Video](#)

The component is being developed by ICCS and is available through [MIT license](#).

4.1.2 Data Harmonisation Pipelines

The data harmonisation toolset includes a CLI tool, a Web Service and a simple Web interface.

The following resources are available:

- Repository: [CLI](#), [Web service](#), [GUI](#)
- Online Documentation: [Pipelines Documentation](#)
- Open API specification: [Pipelines API](#)
- User manual: [Pipelines readthedocs tool](#)
- Demo: [Demo video](#)

The components are being developed by PSNC and are available through [MIT license](#).

The following images are illustrative screenshots of the component.

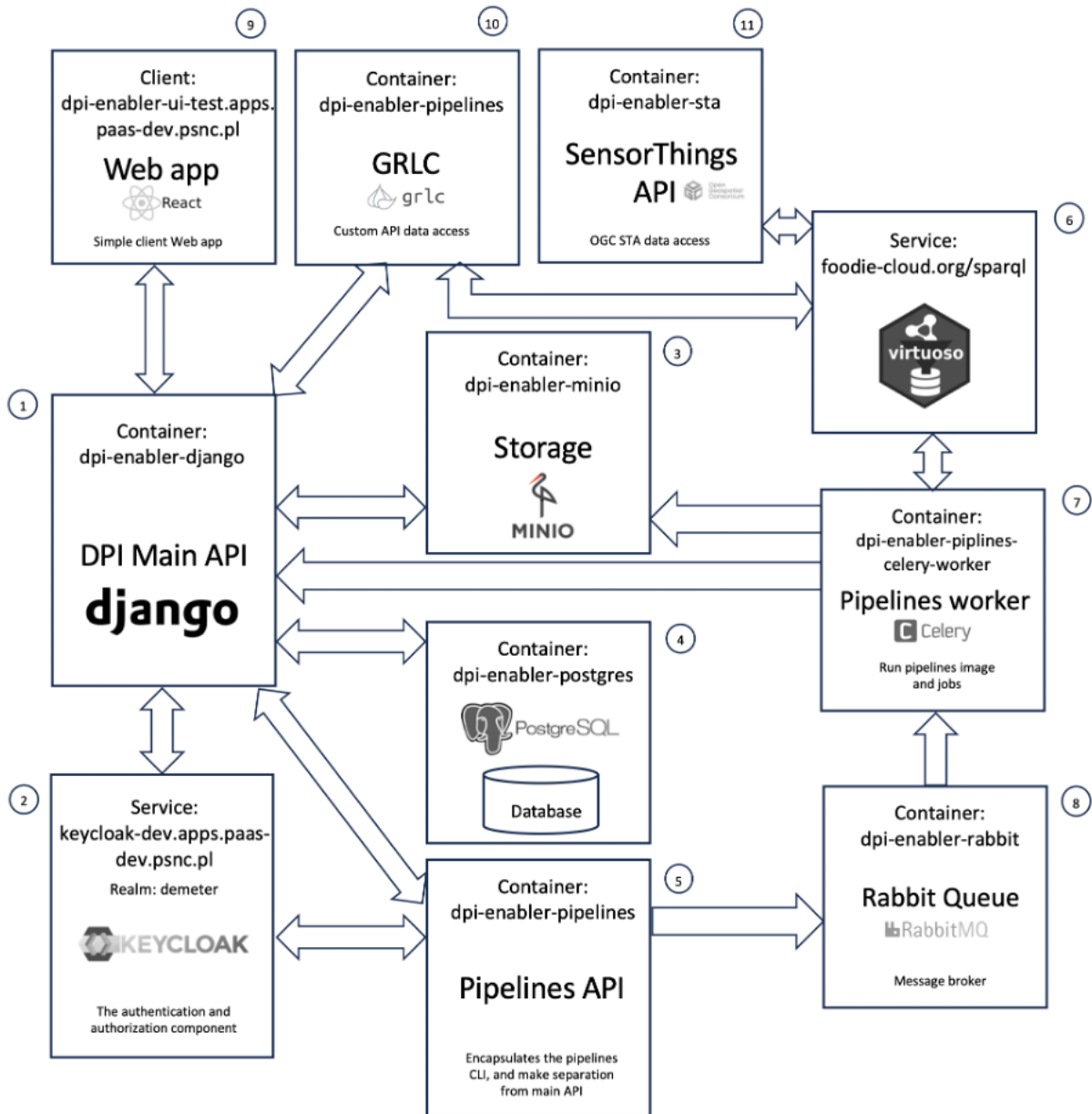


Figure 20. Data Harmonisation Pipelines Service architecture

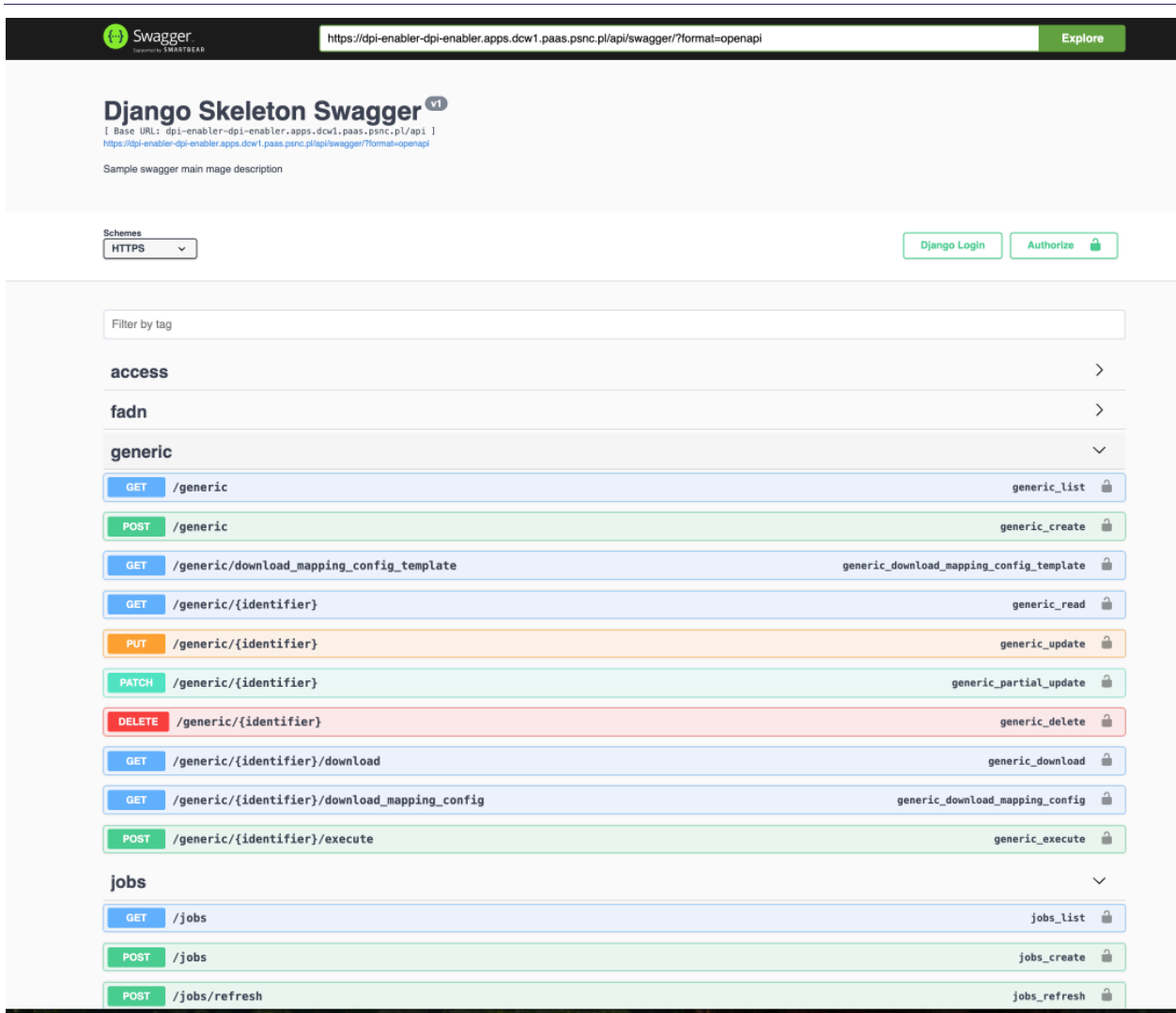


Figure 21. SWAGGER Documentation for Data Harmonization Pipelines