



D1.4

Requirements and Architecture (M30)

ITI

D1.4

Requirements and Architecture (M30)

Revision **v1.0**

Work package	WP1
Task	T1.3, T1.4, T1.5
Due date	30-06-2025
Submission date	26-06-2025
Deliverable lead	ITI
Version	v1.0
Authors	Jordi Arjona (ITI) Santiago Cáceres (ITI) Achilleas Marinakis (OTE) Vasileios Siopidis (CERTH) María José López (TEC) Tasos Nikolakopoulos (ICCS) All partners contributed to the collection of requirements.
Reviewers	Balasubramanian Chandramouli (CINECA) Roman Grzesiak (IDFS)

Abstract

This document provides an update on the technical and functional analysis of the components' requirements and the DATAMITE framework's architecture. A requirements analysis is carried out, and a detailed view of the architecture is presented following the C4 approach.

Keywords

Requirements, architecture, components, building blocks

Document revision history

Version	Date	Description of change	Contributor(s)
V0.1	12-05-2025	1 st version of the document based on the previous iteration	Santiago Cáceres (ITI)
V0.2	02-06-2025	Revision of the contents	Jordi Arjona (ITI), and Achilleas Marinakis (OTE)
V0.3	20-06-2025	Modifications based on peer-review comments	Jordi Arjona (ITI), Balasubramanian Chandramouli (CINECA), Roman Grzesiak (IDFS)
V1.0	26-06-2025	Document ready for delivery	Jordi Arjona (ITI), and 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

R

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

Executive Summary	10
1 Introduction	11
1.1 Deliverable Purpose and Scope	11
1.2 Document Structure	11
1.3 Document Dependencies	12
2 Requirements	13
2.1 Methodology	13
2.2 Analysis & Classification	16
3 Architecture	21
3.1 The C4 Approach	23
3.2 System Context Diagram	23
3.3 Container Diagram	25
3.4 Component Diagrams	27
3.4.1 Data Governance Module	27
3.4.2 Data Quality	32
3.4.3 Data Sharing	38
3.4.4 Data Security	41
3.4.5 Data Support Tools	44
3.4.6 Frontend	48
4 Conclusions	53
5 Appendix	54
5.1 Requirements Tables	54
5.2 Requirements removed compared to D1.2	115

List of Figures

Figure 1: DATAMITE's Architecture	10
Figure 2: Requirements' Priorities.....	17
Figure 3: Requirements' Source	17
Figure 4: Requirements' Type (All Sources)	18
Figure 5: Requirements' Type per Source	18
Figure 6: Requirements' Categories	19
Figure 7: Top 3 Popular Requirements' Categories from End Users (Percentage)	20
Figure 8: Initial Architecture.	21
Figure 9: DATAMITE's Architecture in M20.	22
Figure 10: DATAMITE's Architecture in M30.	22
Figure 11: Conventions followed to create the C4 Schemes.....	23
Figure 12: DATAMITE's System Context Diagram	24
Figure 13: DATAMITE's Container Diagram.....	26
Figure 14: Initial Version of the Data Governance Module Components Diagram.....	28
Figure 15: Current Data Governance Module Components Diagram in the Architecture.	28
Figure 17: Metadata Model	31
Figure 17: Initial Version of the Data Quality Module Components Diagram in the Architecture.	33
Figure 19: Final Data Quality Module Components Diagram in the Architecture.	34
Figure 19: Final DATAMITE Data Quality Vocabulary Schema.....	37
Figure 20: Initial Version of the Data Sharing Module Components Diagram.	39
Figure 21: Current Data Sharing Module Components Diagram in the Architecture.	39
Figure 22: Initial Version of the Data Security Module Components Diagram	42
Figure 24: Current Data Security Module Components Diagram in the Architecture.	43
Figure 24: Initial Version of the Data Support Tools Module Component Diagram.....	45
Figure 25: Current Data Support Tools Module Components Diagram in the Architecture	45
Figure 27: Initial Proposal of the Main Elements in DATAMITE's Frontend.	49
Figure 28: Current Frontend Components in DATAMITE's Architecture.	49



List of Tables

Table 1: Requirements' Fields.	16
Table 2: Data Governance related Requirements.	32
Table 3: Data Quality related Requirements.	38
Table 4: Data Sharing related Requirements.	41
Table 5: Data Security related Requirements.	44
Table 6: Data Support Tools related Requirements.	48
Table 7: Requirements related to the General Management of the Platform.	52
Table 8: Complete Requirements Tables	115
Table 9: Deprecated Requirements	120

Abbreviations

ABE	Attributes Based Encryption
ADA	Americans with Disabilities Act
ADLS	Azure Data Lake Storage
AGPL	Affero General Public License
AI	Artificial Intelligence
AIM	Agriculture Information Model
AIoD	AI-on-Demand
AMQP	Advanced Message Queuing Protocol
API	Application Programming Interface
AWS	Amazon Web Services
BAU	Business As Usual
BDVA	Big Data Value Association
BPN	Business Partner Number
BUFR	Binary Universal Form for the Representation of meteorological data
CA	Certificate Authority
CD	Continuous Development
CDH	Cloudera Distributed Hadoop
CDI	Contextual Decision Intelligence
CI	Continuous Integration
CKAN	Comprehensive Knowledge Archive Network
CMS	Content Management System
CSV	Comma Separated values
DAPS	Dynamic Attribute Provisioning Service
DB	Database
DBMS	Database Management Systems
DBT	Data Build Tool
DCAT	Data Catalogue Vocabulary
DCAT-AP	DCAT Application Profile
DEI	Diversity, Equity and Inclusion
DID	Decentralised Identifiers
DIHs	Digital Innovation Hubs
DLTs	Distributed Ledger Technologies
DoA	Description of Action
DQ	Data Quality
DQV	Data Quality Vocabulary
DS	Dataspace
DSBA	Data Spaces Business Alliance
DSC	Dataspace Connector
DSL	Domain Specific Languages

DSM	Digital Single Market
DSSC	Data Spaces Support Centre
DTD	Document Type Definition
EDC	Eclipse Dataspace Connector
EDIH	European Digital Innovation Hub
EOSC	European Open Science Cloud
EU	European Union
FAIR	Findable, Accessible, Interoperable and Reusable
FTP	File Transfer Protocol
GDPR	General Data Protection Regulation
GPL	General Public License
GPS	Global Positioning System
GRIB	General Regularly distributed Information in Binary form
GUI	Graphical User Interface
GXFS	Gaia-X Federation Services
HDFS	Hadoop Distributed File System
HPC	High Performance Computing
HTTP	Hypertext Transfer Protocol
IaaS	Infrastructure as a Service
IAM	Identity and Access Management
ICT	Information and Communication Technology
ID	Identifier
IDS	International Data Spaces
IDSA	International Data Spaces Association
IEC	International Electrotechnical Commission
IP	Internet Protocol Address
IP	Intellectual Property
ISO	International Organisation for Standardisation
JAAS	Java Authentication and Authorisation Service
JDBC	Java Database Connectivity
JSON	JavaScript Object Notation
JSON-LD	JSON Linked Data
JWT	JSON Web Token
KDE	Key Data Elements
KMS	Key Management System
KPI	Key Performance Indicator
LDAP	Lightweight Directory Access Protocol
ML	Machine Learning
MMLL	Musketeer Machine Learning Library
MPC	Multi-Party Computation
MQTT	Message Queuing Telemetry Transport
N/A	Not Applicable

netCDF	network Common Data Form
NGSI	Next Generation Services Interfaces
OAI-PMH	Open Archives Initiative Protocol for Metadata Harvesting
OAuth	Open Authorisation
ODBC	Open Database Connectivity
ODRL	Open Digital Rights Language
OGC	Open Geospatial Consortium
OpenAPI	Open Application Programming Interface
PaaS	Platform as a Service
ParIS	Participant Information System
PII	Personally Identifiable Information
RAM	Reference Architecture Model
RDBMS	Relational Database Management System
RDF	Resource Description Framework
REST	Representational State Transfer
SaaS	Software as a Service
SAML	Security Assertion Markup Language
SDGs	Sustainable Development Goals
SDK	Software Development Kit
SFTP	Secure File Transfer Protocol
SMEs	Small and Medium-sized Enterprises
SQL	Structured Query Language
SSE	Server-Side Encryption
SSH	Social Sciences and Humanities
SSO	Single Sign On
TEF	Testing and Experimentation Facility
UI	User Interface
VAPT	Vulnerability Assessment and Penetration Testing
VM	Virtual Machine
W3C	World Wide Web Consortium
WCAG	Web Content Accessibility Guidelines
WP	Work Package
XML	Extensible Markup Language
YAML	Yet Another Markup Language

Executive Summary

This document presents the final iteration of the technical and functional analysis of the requirements of all the components, as well as of the architecture of the DATAMITE framework.

To do so, Section 2 presents an analysis of the requirements collected, jointly with a description of the methodology that has been applied. These requirements are also listed in the appendix for completeness.

Most of the requirements are functional (73%) and are divided into the DATAMITE architecture categories: Data Monetisation, Interoperability, Data Sharing, Data Governance, Quality, and Security.

The architecture has now reached its final stage, depicted in Figure 1, that presents DATAMITE's architecture, with the result of the work performed during these months. The document presents its complete specification, including details of its different modules and linking them to the requirements. This architecture is first described following the C4 approach, and hence, different levels of detail are presented (system context, container and component diagrams). Similarly, the document presents the evolution of the different modules throughout the project.

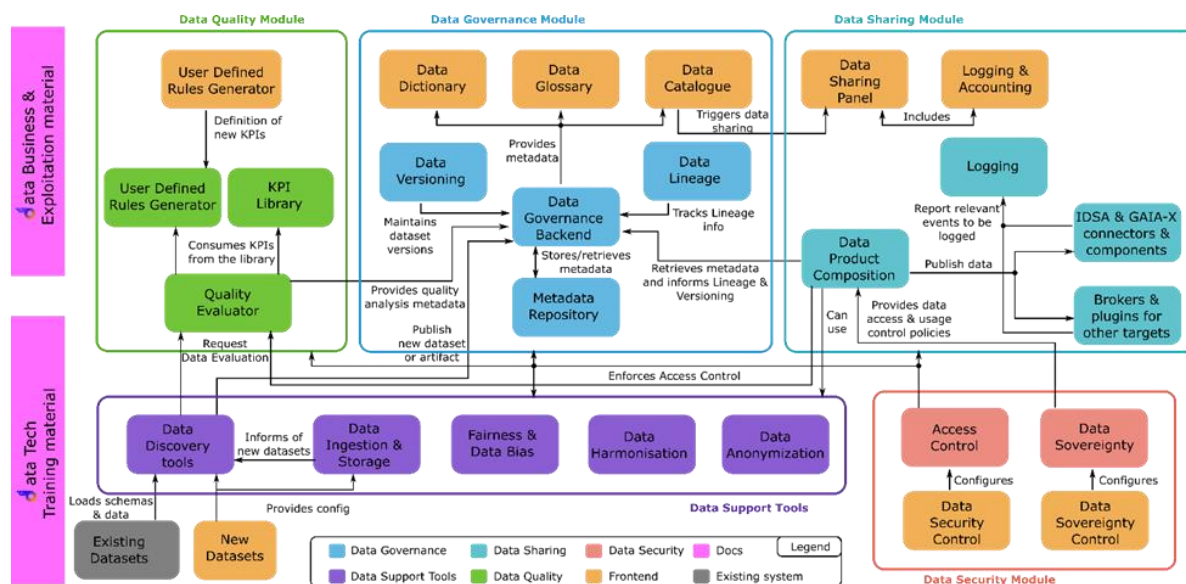


Figure 1: DATAMITE's Architecture

1 Introduction

DATAMITE is a project funded by the European Commission as part of the Horizon Europe programme 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 and the adherence to FAIR principles. Similarly, they 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 on 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:

Provides the analysis of the state of the art and integration of building blocks of ICT-13 Data Platforms components, the technical and functional analysis of the requirements of all the components, and the architecture of the DATAMITE framework.

State-of-the-art and integration of building blocks of ICT-13 was presented in D1.2 in M09, and as it has not been revisited, it is not included in this document.

An extensive list of the functionalities of the framework has been reviewed, listed, and described, as well as their relation to the requirements stated by pertinent stakeholders.

1.2 Document Structure

This deliverable is broken down into the following sections:

- **Executive Summary:** A summary of the contents and main findings is provided.

- **Section 1 Introduction:** It provides the deliverables' general context, dependencies, and structure.
- **Section 2 Requirements:** Presents the work performed in task 1.3 regarding the collection, revision, and analysis of the requirements.
- **Section 3 Architecture:** Introduces the DATAMITE architecture, describing the different modules and their relationship with the requirements previously presented.
- **Section 4 Conclusions:** Conclusions extracted from this work.

Appendixes:

- **Appendix:** Provides the list of requirements with all the details and the comparison with the requirements derived in M9 of the project.

1.3 Document Dependencies

This document is the final iteration of a series of living deliverables. It can be read as a standalone, and there is no need to consult the previous versions, as it contains all the information related to requirements and architecture.

2 Requirements

The main objective of Task 1.3 is to elicit and analyse the requirements for each one of the various software modules that compose the DATAMITE architecture, based on both internal technical and pilots' analysis, as well as taking into consideration feedback from relevant external stakeholders.

Defining high-quality requirements is crucial to ensure the successful delivery of the expected framework. In particular, the process of requirements collection reduces any potential gaps between the technical partners and their software development on one side and the pilot owners and their business priorities and needs on the other. The results of a thorough definition and analysis of requirements also serve as a basis for the evaluation of the final solution.

This section is structured as follows: the methodology for the requirements gathering is presented in subsection 2.1, while subsection 2.2 includes the analysis and classification of the collected requirements. Finally, the Requirements Tables (section 5.1 in the appendix) contain the complete definition to fully describe the requirements.

2.1 Methodology

Requirement analysis is a critical part of software engineering, as it determines the conditions that need to be satisfied by the system to be developed. Since DATAMITE follows a hybrid agile-waterfall approach for software development, the project's requirements are flexible and allow for change in accordance with the evolution of the architecture design and specification. To this end, since the beginning of the project, all partners were requested to provide their requirements based on the project's objectives, as well as on the individual partners' vision and expertise. This dynamic and recurring elicitation process resulted in an initial list of requirements, which was then enhanced according to the outcomes of a dedicated physical workshop that took place during the consortium plenary meeting in June 2023. As a result of two rounds of the requirements collection process, D1.2 presented the status of the list until M9 of the project's lifecycle. In M13, the third round was launched, where the technical partners, as well as the pilot owners, updated the requirements that were listed in D1.2 and also defined new additional requirements. Furthermore, for each requirement (either revised or new), a test case or acceptance criterion was specified. This information will be the base for the verification & validation procedure towards the fulfilment

of the requirements. This document presents the final status of the requirements collection process, also taking into account any duplicates, conflicts or overlapping identified in the initial input. Moreover, when comparing to the list presented in D1.2, there are some requirements that have been removed. A clear justification for the removal of each one of these requirements has been provided in Appendix 5.2.

In general, in trying to achieve the optimal definition of the requirements, the following rules and best practices¹ were followed during the elicitation process:

A requirement definition should be unambiguous, clear, and specific to ensure that it cannot be interpreted in various ways. For that reason, vague and general requirements descriptions have been avoided where this was possible.

The requirements must be consistent with each other, meaning that no requirement should conflict with any other requirement.

The feasibility of each requirement should be considered. There must be a clear mapping between the functionalities and the characteristics of the software modules that are going to be implemented and the requirements list. Those that cannot be implemented by any component should be eliminated or marked with lower priority.

Each requirement should be referred to using a unique identifier. This will enable the traceability and the testing of the requirements, also enhancing their consistency.

The requirements must be testable, either in a quantitative or in a qualitative way, so that it is possible to verify that the solution satisfies them, totally or partially.

The descriptions of the requirements must be flexible enough. However, it should be clear what constitutes a change of a requirement's goal, as distinguished from a valid interpretation of it. The key is to find a balance between adherence to a baseline and sufficient flexibility.

In some cases, those baseline requirements are agreed upon by many stakeholders (pilot owners, software modules developers, integrators, etc.), who also realise that they will evolve based on their flexibility. All these partners are involved in completing the requirement.

¹ Systems Engineering Guide, MITRE. Available online, <https://www.mitre.org/sites/default/files/publications/se-guide-book-interactive.pdf>

Requirements collection is an iterative process. At each step, the results must be validated against the stakeholders' needs, and if not, then proceed to further analysis, before adjusting the architecture design accordingly. To collect the requirements in a consolidated manner and have a homogeneous reading and understanding, a common fixed template has been created, whose fields are explained in Table 1:

Requirements Fields	
ID	<p>This is the unique identifier of each requirement: R{counter} (e.g., the first requirement in the list is R001) The ID is kept unchanged between the various iterations of the deliverable to enable traceability and consistency.</p>
Type	<p>This field determines whether the requirement is Functional or Non-Functional²³:</p> <p>Functional requirements are the ones that specify a function that a system or system component must be able to perform. These requirements should be action oriented and should describe the tasks or activities that the system performs during its operation.</p> <ul style="list-style-type: none"> • <p>Non-Functional requirements provide a description of a property or characteristic that a software system must exhibit or a constraint it must respect, other than an observable system behaviour. In a nutshell, they describe not what a software will do, but how the software will do it. These requirements include several aspects, such as development constraints, business rules, external interfaces, and quality attributes.</p>
Source	<p>This field specifies if the requirement is derived from:</p> <ul style="list-style-type: none"> • Internal Technical Analysis • Pilot Analysis • External relevant Stakeholders
Category	<p>This field specifies the category of the requirement: (i.e., Interoperability, Quality, Usability, etc.)</p>

² K. M. Adams, "Non-functional Requirements in Systems Analysis and Design," Springer, 2015

³ D. Z. N. N. Dewi Mairiza, "An Investigation into the Notion of Non-Functional requirements," in ACM Symposium on Applied Computing, 2010

Requirements Fields	
Description	This is the actual description of the requirement and will be used to determine the completion of it. The description must be linguistically aligned with the Priority field of the requirement.
Rationale	This field describes the need that the specific requirement is covering. It also shows the background goal for the requirement, which removes much of the ambiguity .
Priority	<p>Based on the MoSCoW method⁴:</p> <p>MUST: Describes a requirement that will gather the highest effort from the consortium. The vast majority of those requirements must be satisfied in the final solution for the project to be considered a success.</p> <ul style="list-style-type: none"> • <p>SHOULD: Represents a high-priority requirement that should be included in the solution if it is possible. This is often a critical requirement but one which can be satisfied in other ways if strictly necessary.</p> <ul style="list-style-type: none"> • <p>COULD: Describes a requirement which is considered desirable but not necessary. This will be included if time and resources permit it.</p> <ul style="list-style-type: none"> • <p>WON'T: Represents a requirement that stakeholders have agreed will not be implemented within a given release, but may be considered for the future.</p>
Test Case / Acceptance Criteria	This field describes the way to test the requirement, to ensure that it is developed and thus, fulfilled . This information will be the base for the verification & validation procedure .

Table 1: Requirements' Fields.

2.2 Analysis & Classification

In this subsection, the requirements of the project are presented at a higher level, and the corresponding metrics and figures reflect the overall expectations towards the project's objectives.

⁴ I. I. o. B. Analysis, A Guide to the Business Analysis Body of Knowledge (BABOK guide), 2nd ed., 2009

In total, 154 requirements have been defined, the relative majority (44%) of which **MUST** be completed for the project to be successful. 43% of the requirements are significant but not mandatory to be fulfilled (**SHOULD**), whereas 10% of them are nice to have but would not affect the project's outcomes and achievements if not implemented (**COULD**). Finally, 3% of the defined requirements cannot be fulfilled within the project's lifecycle, although they are still in the scope of DATAMITE (Figure 2). Then Figure 3, depicts the source of the collected requirements.

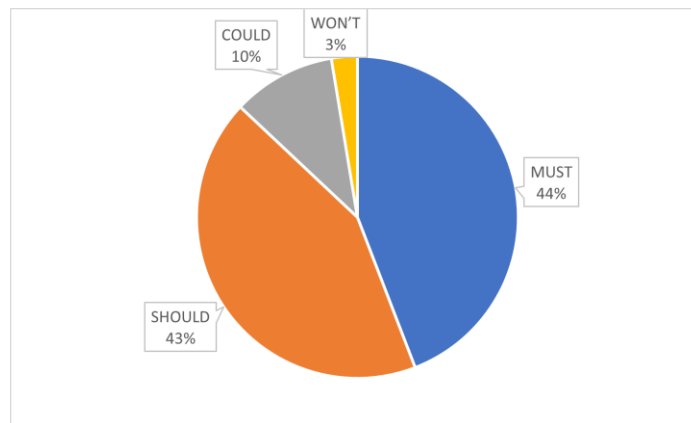


Figure 2: Requirements' Priorities

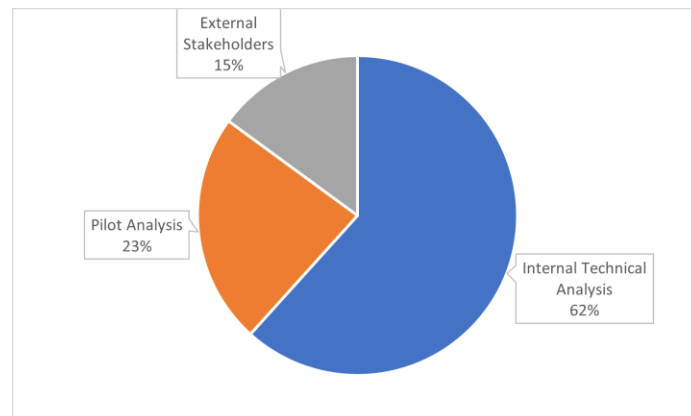


Figure 3: Requirements' Source

According to the definition provided in subsection 2.1, functional requirements describe the features that the DATAMITE framework should offer to its users, while non-functional requirements are quality attributes and constraints that it should comply with. Most partners focused on identifying all the functionalities that are expected from the various functional components of the DATAMITE architecture. Therefore, the collected requirements are functional in their majority (Figure 4). It is worth noting that, as shown in Figure 5, this trend applies to both

technical and business sources (Pilot Analysis and feedback from External relevant Stakeholders) of defining requirements.

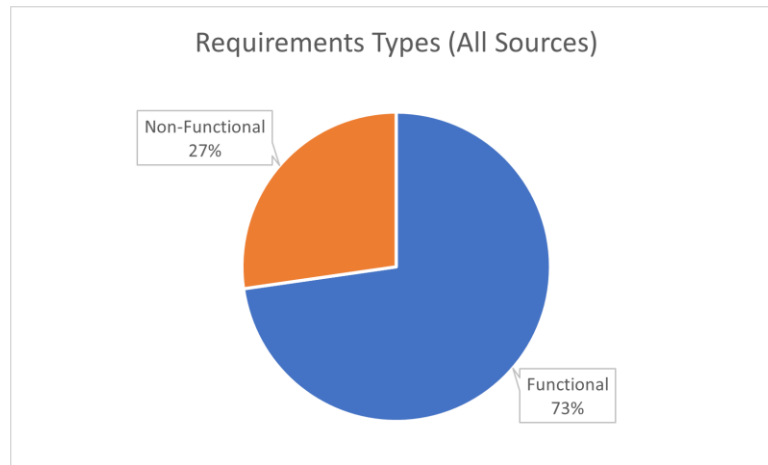


Figure 4: Requirements' Type (All Sources)

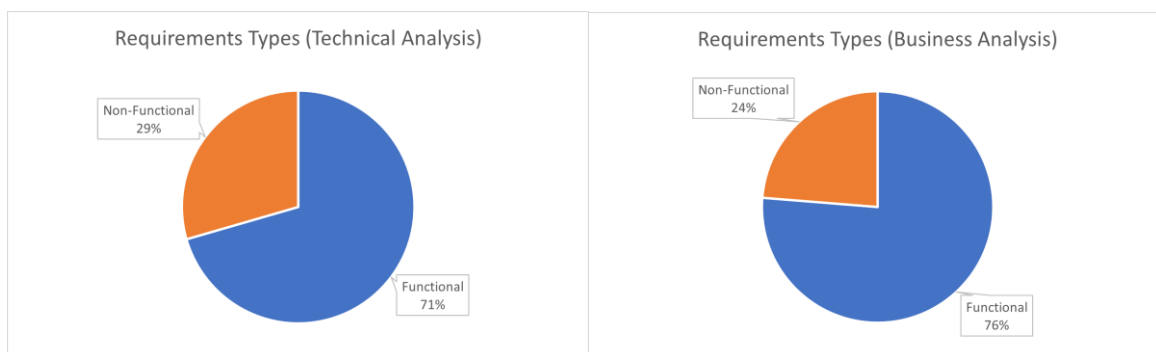


Figure 5: Requirements' Type per Source

Since the first round of the requirements elicitation process, six main categories have been identified, enabling the classification and analysis of the collected requirements. These categories are related to the project's key objectives and to the main software modules that compose the DATAMITE architecture:

- Data Monetisation
- Interoperability
- Data Sharing
- Data Governance
- Quality
- Security

If a requirement cannot be assigned to none of the above categories, then other labels are used, such as Functionality, Usability, Compatibility, Performance etc.

In more detail, **Figure 6** presents the categorisation of the defined requirements:

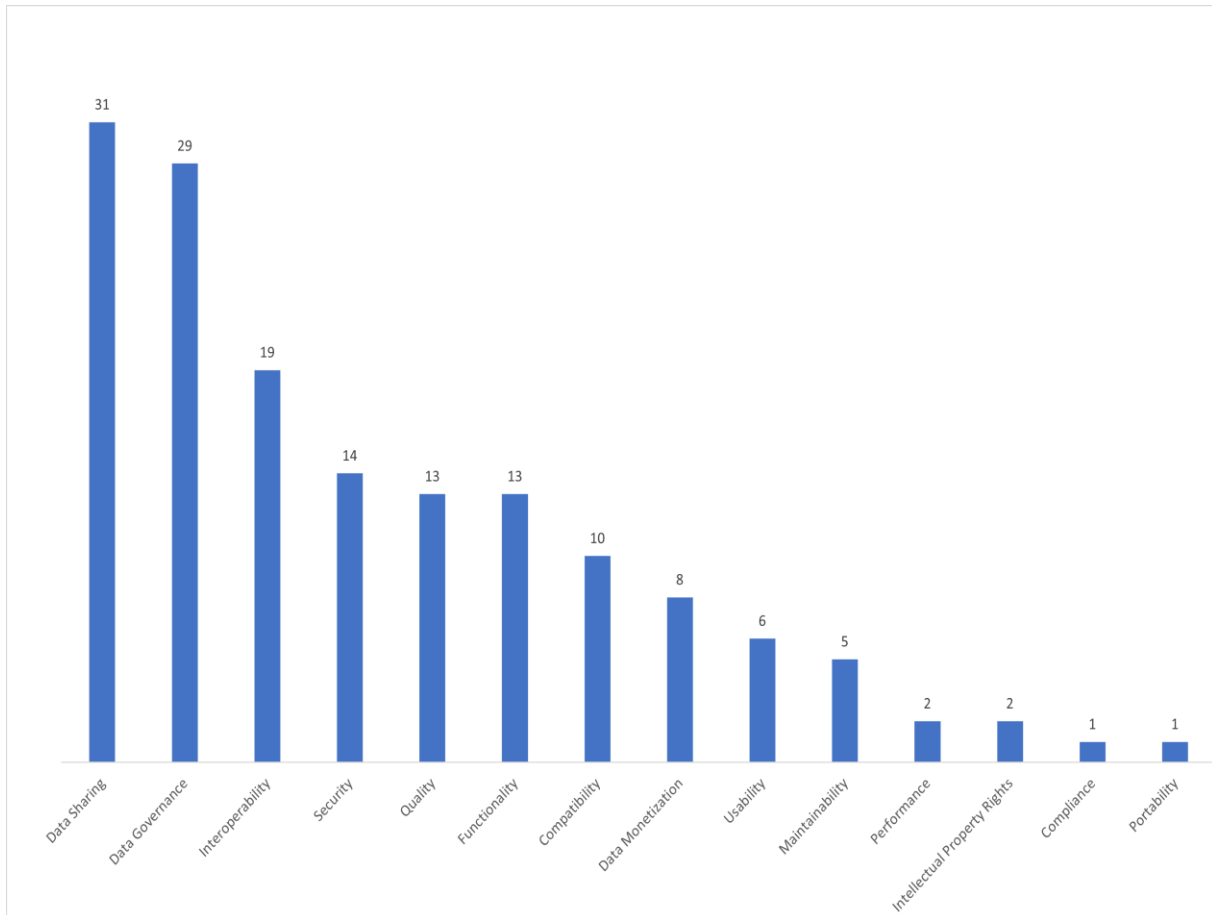


Figure 6: Requirements' Categories

Within the scope of Task 1.3, there is a need to analyse the challenges which the end users of the DATAMITE would face, to effectively use the expected framework. End users consist not only of the project's pilots but also of any external relevant stakeholders that could potentially adopt the proposed solution. To this end, Figure 7 depicts the most popular (in terms of percentage) categories of the requirements that were derived from pilot analysis as well as from external stakeholders' needs (only categories with significant numbers have been considered).

To have a better understanding of the importance of these findings, the metrics of Figure 7 should be combined with those from Figure 6 and Figure 3. More specifically, although 38% of the overall requirements derive from a non-technical source, pilots and external stakeholders have defined

the 100% (8 out of 8) of the total number of requirements concerning Data Monetisation, the 58% (18 out of 31) of requirements concerning Data Sharing, and the 43% (6 out of 14) of requirements concerning Security.

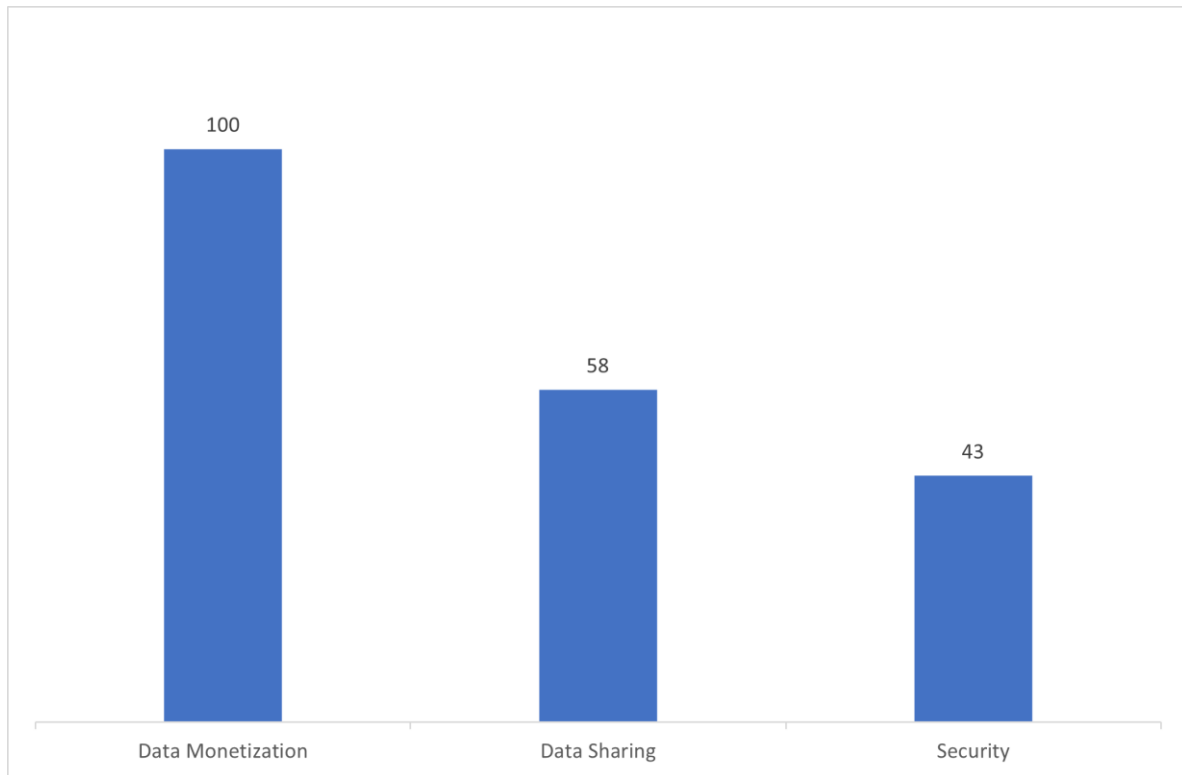


Figure 7: Top 3 Popular Requirements' Categories from End Users (Percentage)

3 Architecture

This section presents DATAMITE's architecture. DATAMITE has been devised as a modular solution, where each of its modules can be optionally deployed by the user according to his or her needs. However, modular does not mean independent, and there exist relations, complementarities, and synergies between them that make every module better in the presence of the others. These modules were presented in Figure 8. This was the initial DATAMITE high-level architecture presented in the proposal phase, and it conformed to our starting point. This architecture has evolved substantially during the course of the project, becoming more detailed and comprehensive. The final status of the architecture is presented in Figure 10, although Figure 9 has been included as well to show the changes between M20 and M30. The following subsections will provide an updated description of the high-level view of DATAMITE's architecture, as well as each one of the different modules and the relation among them, as well as other elements that can be relevant for its correct functioning (e.g., storage systems, computing resources, etc.).

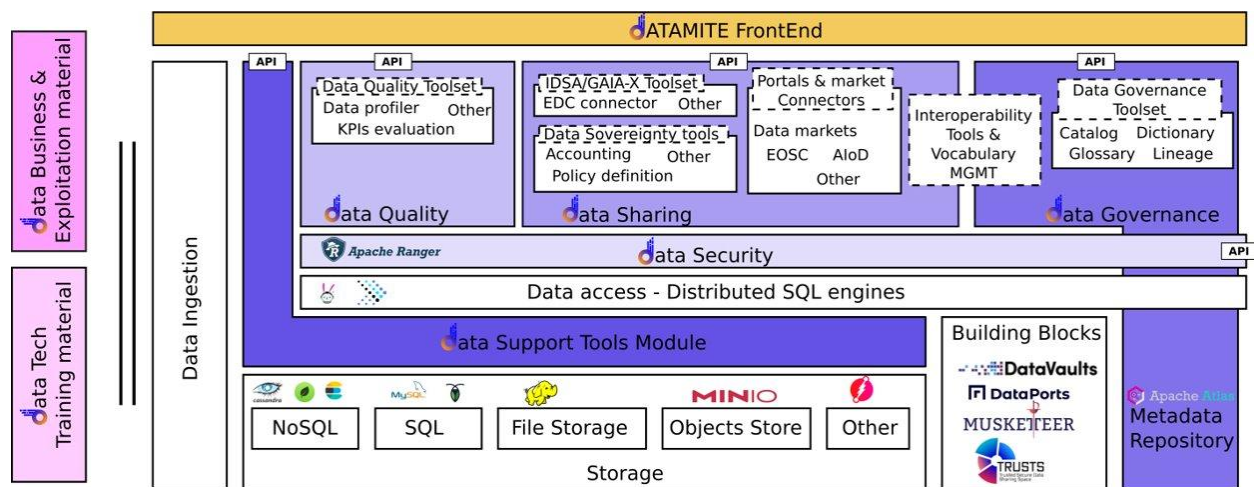


Figure 8: Initial Architecture.

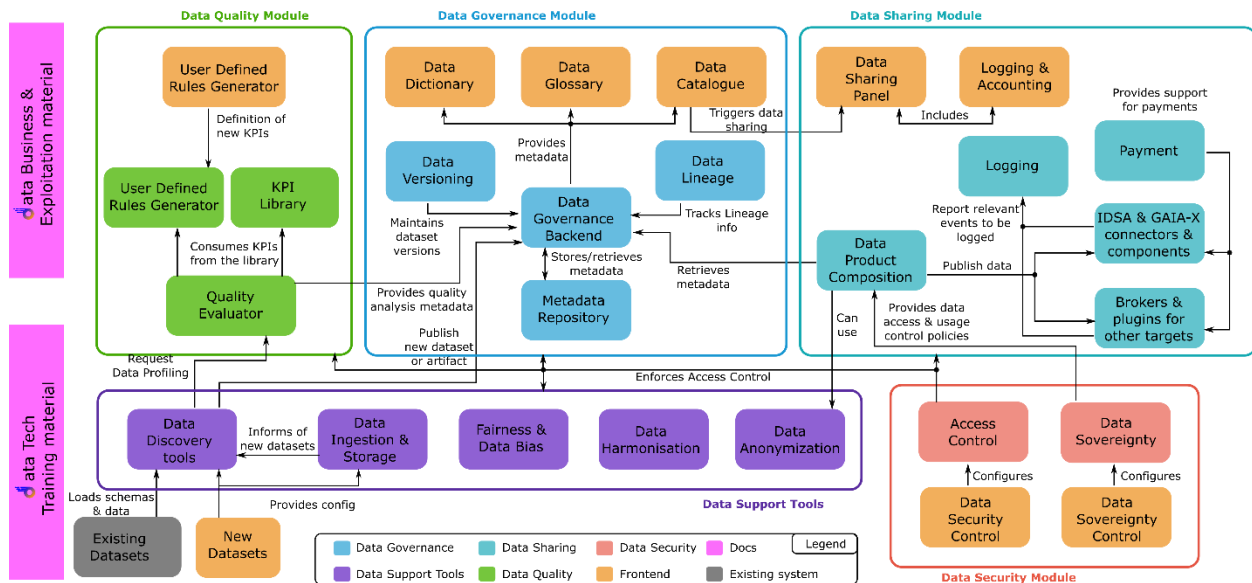


Figure 9: DATAMITE's Architecture in M20.

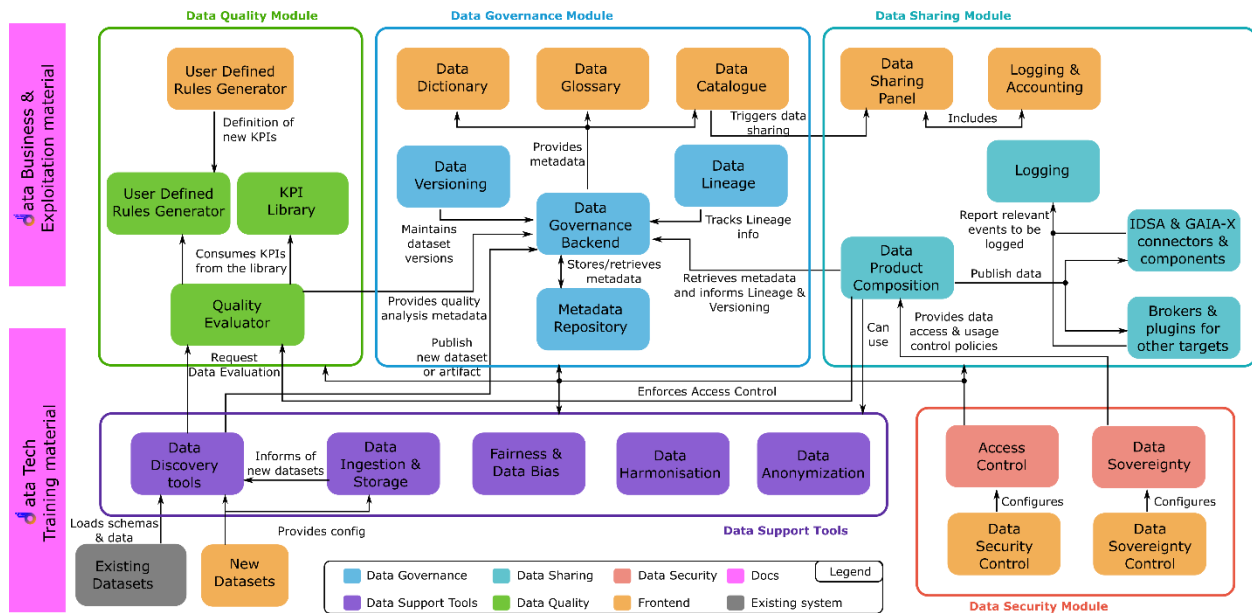


Figure 10: DATAMITE's Architecture in M30.

3.1 The C4 Approach

This architecture has been described following the C4 model⁵, an "abstraction-first" approach to diagramming software architecture that proposes up to four levels of diagrams: System Context, Container, Component and Code Diagrams.

The rationale behind this hierarchy would be that a software system is made up of one or more containers (applications and data stores), each of which contains one or more components, which in turn are implemented by one or more code elements (classes, interfaces, objects, etc). In DATAMITE, we will focus on the first three levels, which are presented in the following sections. Figure 11 presents the colour conventions used for the different modules at the different abstraction levels.

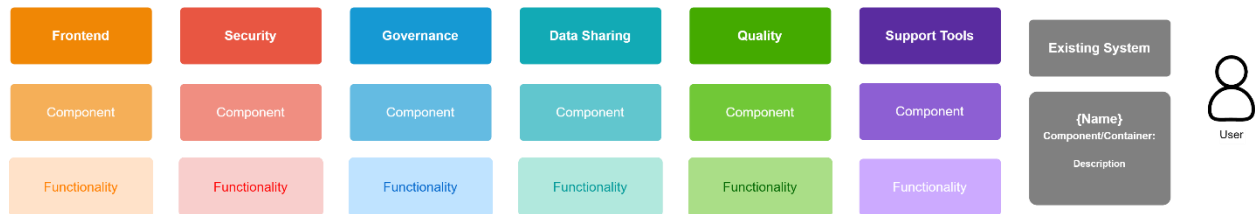


Figure 11: Conventions followed to create the C4 Schemes.

3.2 System Context Diagram

This section presents the System Context Diagram, depicted in Figure 12. Its goal is to provide a high-level view of the system and its interaction with other systems or main types of users. DATAMITE is conceived as a framework offering a series of solutions to improve how organisations manage their data from different points of view, i.e., governance, quality, security, and sharing, but with the primary goal of boosting their data monetisation capabilities.

⁵ <https://c4model.com/>

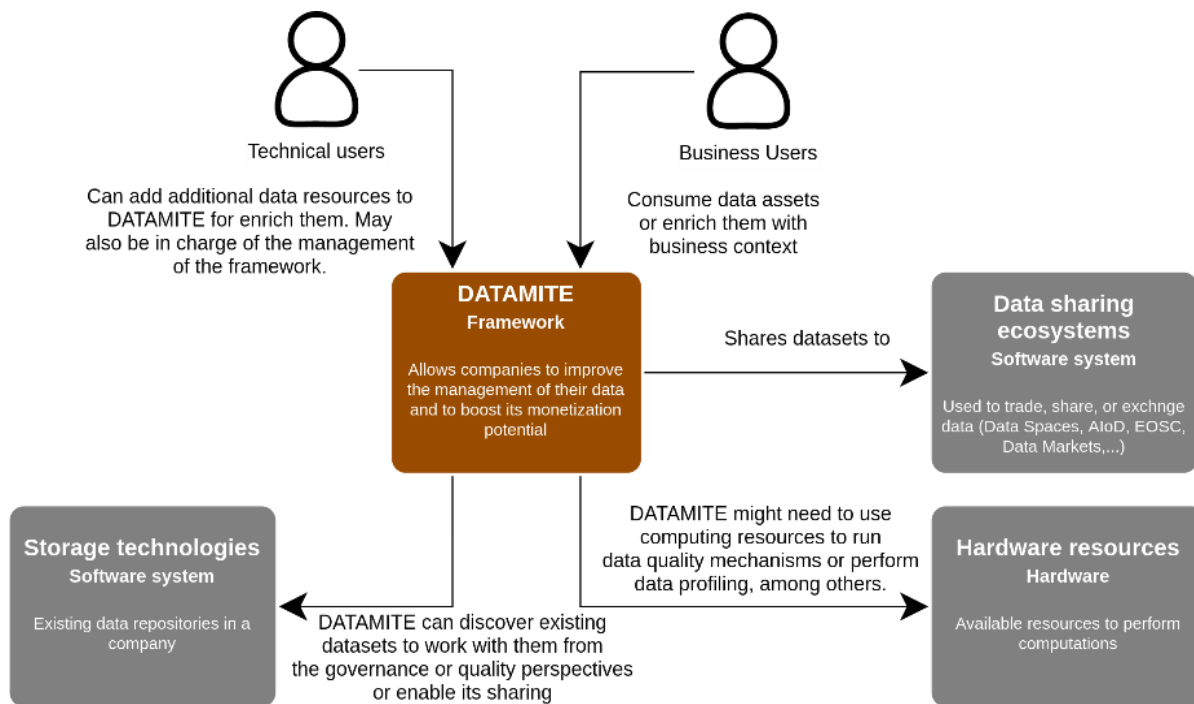


Figure 12: DATAMITE's System Context Diagram

In this regard, it is worth recollecting that, at least in the DATAMITE context, monetisation refers to those actions based on the use of data that allow decision-makers to increase their income. This monetisation can be internal and external. Internal monetisation relates to making informed decisions that will turn into improvements in the way a company does business, leading to larger revenues. These informed decisions can be based on forecasts, projections, and IA models built from company data and/or public sources, for instance. External monetisation involves utilising data to create products or services that can be sold to third parties. This can include data sharing or exchange to leverage synergies and collaborations with third parties.

Hence, DATAMITE, as a framework, will interact mainly with two kinds of users:

- **Technical users** are those in an organisation expected to work using data or computing resources. Would be in charge of deploying and managing DATAMITE in an organisation.
- **Business users** are those expected to consume data, not necessarily having a technical profile. They can span from data analysts to decision-makers who need to find, understand and use data in a way that is as easy as possible. Business users are

expected to be the main consumers of many of DATAMITE's functionalities in an organisation.

And three types of systems:

- **Data sharing ecosystems** can span marketplaces, data platforms or IDS dataspace, among others. It refers to any space or context where data can be published or shared for any purpose. DATAMITE will publish data into those ecosystems according to their data structures, vocabularies, or conditions.
- **Storage technologies** refer to the infrastructure holding the data of an organisation. DATAMITE will be able to discover these resources and facilitate their consumption by business users by improving its cataloguing, enriching them with metadata or providing additional information, such as quality KPIs.
- **Hardware resources.** Even when DATAMITE focuses not on the analysis or cleansing of data, hardware resources will be required for some DATAMITE functionalities, e.g., performing data profiling or data quality evaluation, which imply performing some computations.

3.3 Container Diagram

This section presents the container diagram for DATAMITE, shown in Figure 13. The container diagram presents a zoom-in into the framework, each one of its modules is equivalent to a container – i.e., a separately runnable/deployable unit executing code - in the C4 approach. This diagram shows the high-level architecture and how responsibilities are distributed among its different elements.

As shown in Figure 13, DATAMITE has five major modules devoted to Data Governance, Data Quality, Data Sharing, Data Security and Data Support Tools. An overview of these modules was already provided in D1.1⁶, and more details will be presented in their associated subsections. The components diagram presents the main interactions in this module.

It is worth emphasising the following aspects. One module with a particular centrality in the diagram is the Data Governance module. The Data Governance module, due to the central role

⁶ D1.1 – Roadmap and Pilots definition.

of the data catalogue, will be the point of access to most quality functionalities as well as for certain data-sharing aspects.

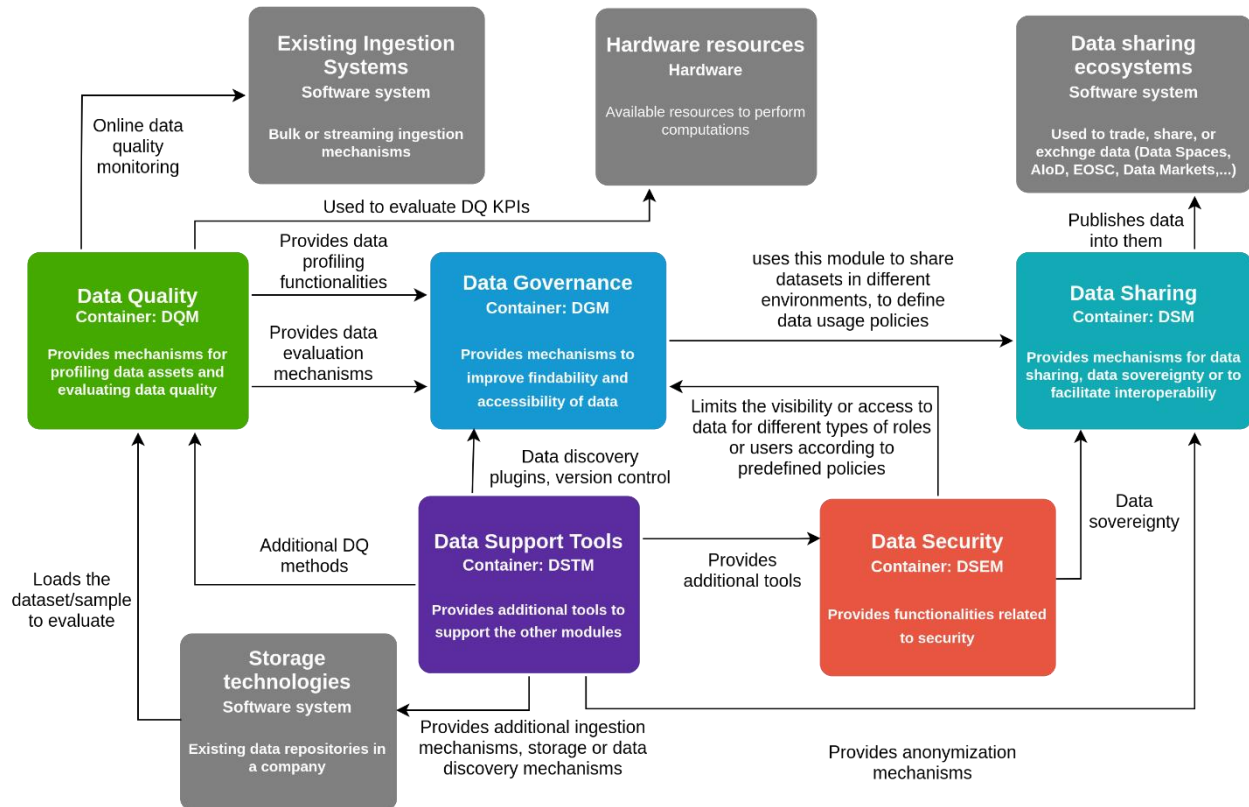


Figure 13: DATAMITE's Container Diagram.

Similarly, it can be appreciated that the role of Data Security and Data Support Tools is relatively horizontal. In the first case, Data Security must interact with the different modules to define and apply the security policies, controlling who can do what with the data. On the other hand, Data Support Tools will provide complementary tools to the different modules, extending their functionality. Examples of these tools could be a collection of plugins for different storage technologies to facilitate data discovery, anonymisation algorithms to be applied before sharing data or certain specific quality indicators.

Also, it is important to remark that the DATAMITE framework is a tool that companies deploy in parallel to their infrastructure and has little interaction with other systems, as its main goal is to provide tools to facilitate how data is consumed, to locate it, enrich it or share it; being the major interaction with data storage technologies. However, this becomes more complex when Data Quality comes into play, as it will require the evaluation of datasets or samples of them. This

implies access to computing resources - to perform the evaluation - and possibly replicating or moving data across the system. These are major complexities that must be handled appropriately by the consortium.

Finally, although depicted as another module, the Data Sharing module will be a rich and complex module, including multiple relevant tools such as the connectors to the different data sharing ecosystems, a clearing house, data sovereignty tools, or an IDS connector.

3.4 Component Diagrams

This subsection presents the different component diagrams for DATAMITE's modules. Each module will enumerate and describe its main functionalities, link these functionalities with the requirements that this module will cover and present the corresponding component diagram.

3.4.1 Data Governance Module

3.4.1.1 Initial Advances

The main goal of the Data Governance module is to assist organisations in managing, accessing, and enriching their data to enhance its usability, allowing them to define their vocabularies related to datasets to ease their consumption by non-technical personnel. It will offer several functionalities, at least a metadata repository, data catalogue, glossary, and data lineage capabilities. Finally, as was depicted in Figure 14, the previous version of the deliverable has been updated in Figure 15. It will have dependencies and interactions with other modules, such as the data quality module, as it will be consumed to populate the data dictionaries or the data sharing module regarding interoperability and vocabulary management aspects. This section elaborates on this module.

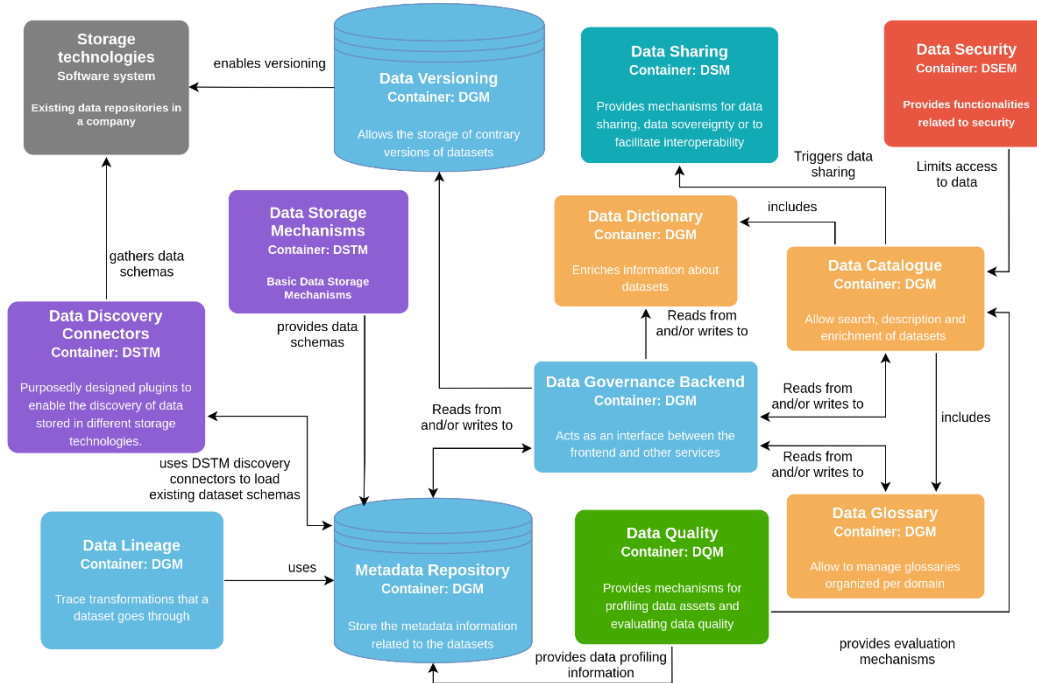


Figure 14: Initial Version of the Data Governance Module Components Diagram.

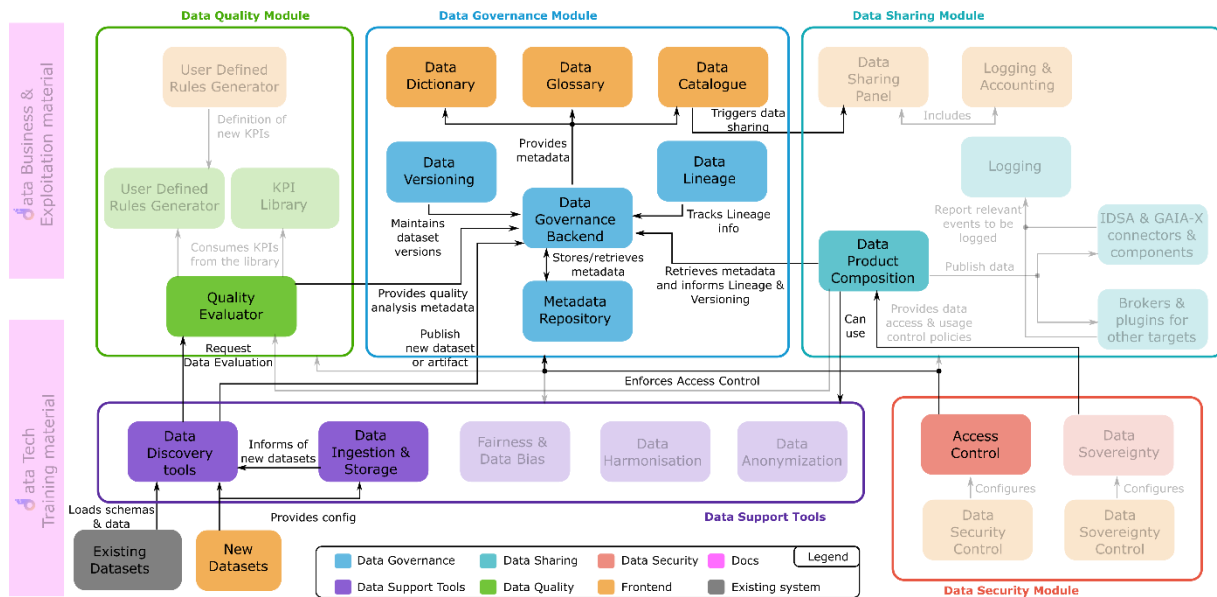


Figure 15: Current Data Governance Module Components Diagram in the Architecture.

3.4.1.2 Updates up to Month 20

The main changes between Figure 14 and Figure 15 are as follows: First, the Data Governance Backend becomes a complete façade for the metadata repository, so no other components will

directly interact with it. Hence, the data discovery tools or data ingestion and storage will report to the backend, not to the repository, and the same happens with the Quality Evaluator or other Governance components like Data Lineage. Second, even when not reflected in the architecture figure for the sake of simplicity, data versioning is still expected to interact with storage technologies. Access control will interact not only with the catalogue but also with all the components requiring any kind of authentication/authorisation. Finally, the Data Sharing module will also interact with the Data Governance Backend, given the addition of the Data Product Composition component, which will retrieve metadata to compose the data products from it.

3.4.1.3 Updates up to Month 30

There are no major changes in the Data Governance module at the architectural level. The most relevant change is due to the advances in the definition of the Data Versioning component. Data Lineage will capture, essentially, the origins from where datasets are created, but will also track the actions performed on a data product during its creation. This is, aggregation actions, anonymisation, filtering, etc. The user will have the possibility of saving the status of the data product during this process. Data versioning will be in charge of managing these intermediate versions of data products. These actions are performed through the Data Governance Backend, so the description of the connection between the Data Product Composition and Data Governance Backend has been modified.

3.4.1.4 Functionalities and Requirements

The updated Data Governance module is shown in Figure 15. This diagram depicts the most relevant functionalities and interactions in this module, which are described below these lines.

- **Metadata Repository:** The metadata repository is the core component of the data governance module. Its purposes are multiple, as reflected in the entities created to store the different kinds of metadata, which can be seen in Figure 17. First, it will store metadata associated with datasets. These metadata can be divided into the metadata being captured from the dataset itself (e.g., its structure), the quality metadata produced by DATAMITE by evaluating the dataset, and the metadata introduced by the user from frontends like the Data Catalogue. Second, it stores the data glossary and the vocabulary and terms within it. These terms can have relations or dependencies among themselves and will eventually be linked to datasets to enrich

them. Finally, it will be extended to consider additional types of data (semi-structured or non-structured) and data products.

- **Data Governance Backend:** It is the interface between the metadata repository and the rest of the DATAMITE components, like data discovery tools; other components within the governance module, like the data versioning and the data glossary, catalogue and dictionary; or components in other modules like the data product composition. It implements an API to be consumed by these components to store or offer information. This component is being completely developed within the project.
- **Data Catalogue, Data Glossary and Data Dictionary:** Even when they belong to Data Governance Module, they are described in Section 3.4.6, as they are part of DATAMITE's frontend.
- **Data Lineage:** It refers to the tracking and visualisation of data through various stages of its lifecycle within an organisation's systems and processes. Its purposes are multiple. On the one hand, it can assist with compliance, auditing, or assurance, as well as tracking the transformations that data goes through. Also, it can be linked with data versioning tools in troubleshooting or debugging. From the point of view of governance, it provides transparency in the use of data. Within DATAMITE it will track the changes or events on data ingested in the framework and regarding the construction of data products, feeding it from components like Data Discovery, Data Ingestion or the Data Product Composition.
- **Data Versioning:** Data versioning will incrementally store the different stages a data product goes through while being constructed by the Data Product Composition component. These versions will be accessible through the Data Lineage frontend.

Incorporating these functionalities within DATAMITE serves a distinct and well-defined purpose within its framework. This purpose is substantiated by their alignment with various requirements articulated by pertinent stakeholders, as described in Section 2. In Table 2 below, we illustrate the correlations between the functionalities offered by the Data Governance module and the requirements above.

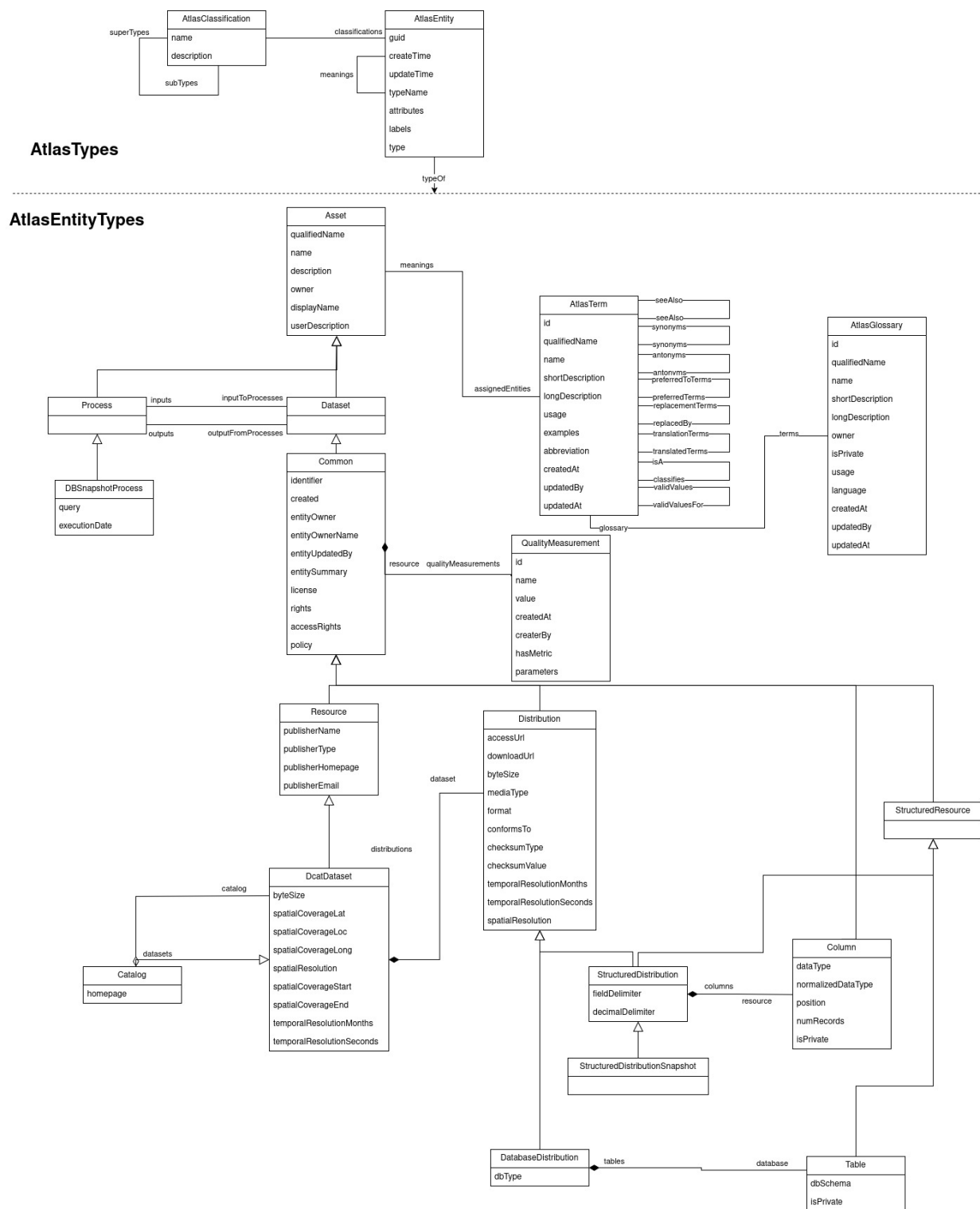


Figure 16: Metadata Model

Functionality	Associated Requirements
Metadata Repository	R026, R033, R037, R038, R039, R047, R068, R079, R080, R083, R084, R085, R088, R093, R094, R102, R147, R182
Data Governance Backend	R044, R047, R056, R070, R087, R095, R096, R103, R139, R149, R153, R172
Data Catalogue	R031, R056, R062, R063, R068, R079, R080, R082, R083, R096, R103, R105, R108, R110, R111, R112, R149, R172, R182
Data Glossary	R019, R020, R022, R023, R025, R044, R084, R085
Data Dictionary	R109
Data Lineage	R024, R064, R089, R090
Data Versioning	R064

Table 2: Data Governance related Requirements.

3.4.2 Data Quality

3.4.2.1 Initial Advances

This section presents DATAMITE's Data Quality module, its features, and its relation to the requirements defined by stakeholders, which are presented in Section 2. The Data Quality module's main purpose is to offer the necessary tools to companies so they can assess the quality of their datasets. This task can be performed in several ways. First, the data is profiled over the entire dataset or sample in a high-level or generic way. This profiling may give some first-hand, valid information to the organisation, but it does not necessarily indicate how good it is. To do so, it is necessary that the organisation can mark the specific fields that are relevant and what KPIs can be used to evaluate them. These KPIs, in addition, can be generic or inherent (e.g., statistics, averages, counts, distributions) or purposefully defined for that domain. It may also be important to monitor the quality of data that is being ingested in streaming to undertake corrective actions.

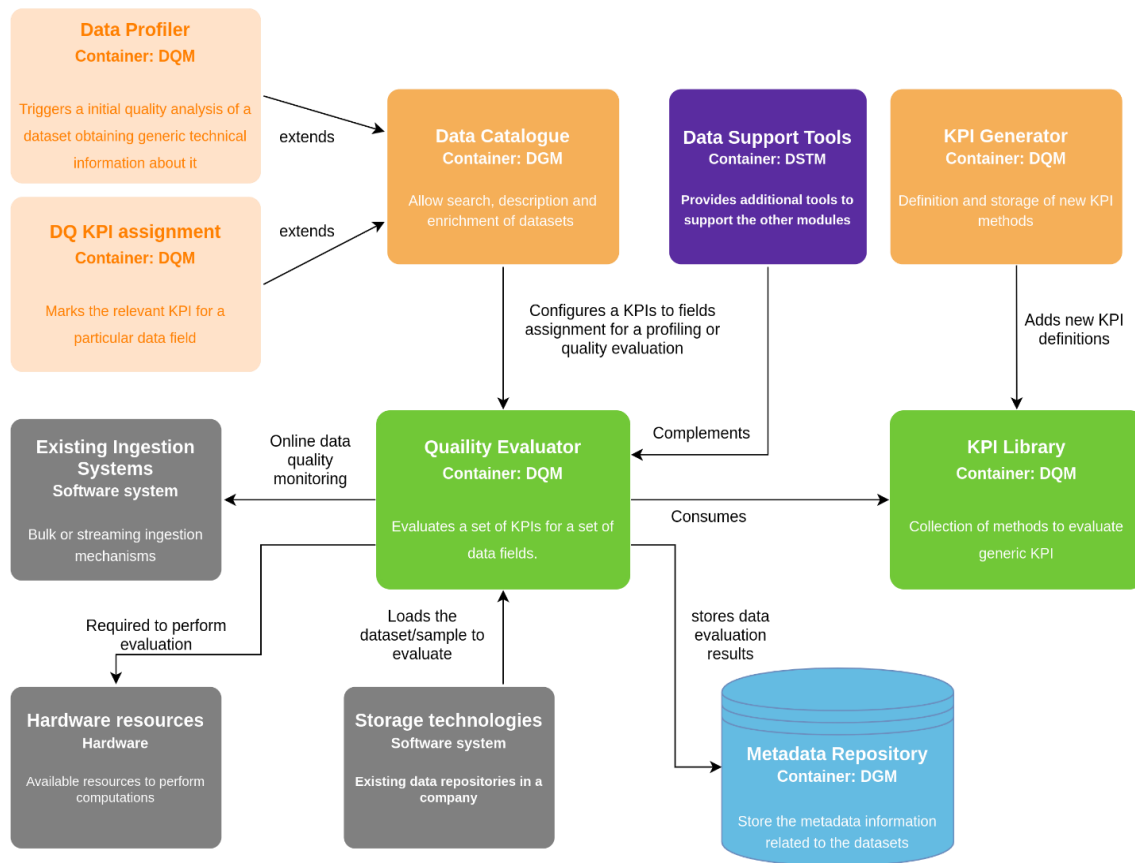


Figure 17: Initial Version of the Data Quality Module Components Diagram in the Architecture.

Finally, organisations should be able to perform these tasks in a user-friendly manner, simplifying the job of their technical personnel. Figure 17 shows the initial component diagram for the Data Quality Module, depicting its components, their relations and the relation with other components or modules in the architecture.

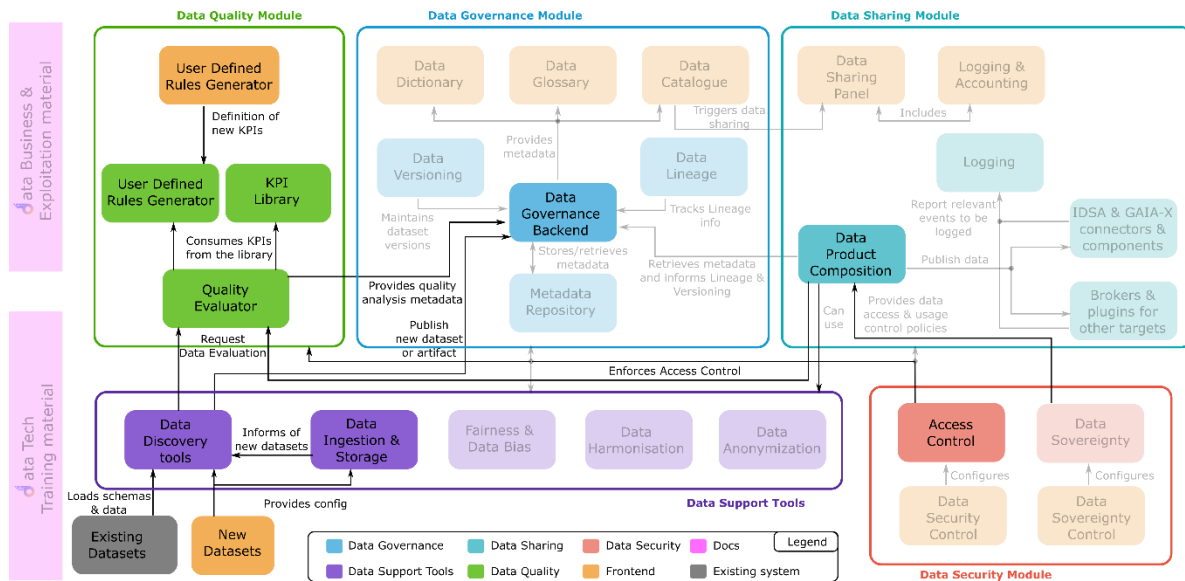


Figure 18: Final Data Quality Module Components Diagram in the Architecture.

3.4.2.2 Updates up to Month 20

Figure 19 shows the status of the Quality module at M20. Regarding the Quality Evaluator, its communication with other components is now clear and well-defined. The Quality Evaluator will receive data from the Data Discovery tools, along with additional information (metadata) regarding the dataset's columns. In addition, the Quality Evaluator shall consume the KPI Library and utilise it as a standard Python library in order to proceed with the automated KPI profiling process. Furthermore, the Evaluator will invoke the User-Defined Rules Generator in order to evaluate additional rules / KPIs that have been defined by the end-users. Finally, the Quality Evaluator will return the results by combining both user-defined rules and inherent KPIs.

As an evolution of the initially considered DQ KPI Assignment component, the User Defined Rules Generator has been included as a means to allow users to create indicators related to the requirements they have in their organisations. The front end is in charge of interacting with the users and the metadata information associated with a dataset and is managed by the Data Governance module. The backend is responsible for the storage of the indicators in the form of rules, following the format of the extended Data Quality Vocabulary (DQV), as well as for their evaluation.

3.4.2.3 Updates up to Month 30

Figure 19 shows the latest architecture for the Data Quality module. There has not been any major change since Month 20.

3.4.2.4 Functionalities and Requirements

Figure 17 and Figure 19 present the final version of the Data Quality module, showing its most relevant features and relations to elements from other modules and existing software or hardware.

These features are described in detail below these lines:

- **Quality Evaluator:** Its main purpose is to evaluate a set of KPIs, as well as user defined rules, for a set of data fields according to a provided configuration. This configuration can be generated in an automated fashion, with the purpose of performing data profiling and running a series of KPIs based on the data field types and the rules defined by the user to be part of that automated profiling. The KPI Library is responsible for the implementation (and inclusion) of generic KPIs that can be managed by the Evaluator and the User Defined Rules Generator Library is in charge of creating and evaluating the user rules. Apart from profiling, another approach is the on-demand execution of quality evaluation, which consists of the evaluation of the standard KPIs and the rules defined by the user for that purpose. The analysis will be performed by consuming hardware resources from the organisation and over data stored in the organisation's storage systems or data coming in streaming. The results of the data quality evaluation will be stored in the metadata repository from the Data Governance module.
- **KPI library:** Provides the different mechanisms or routines for evaluating a set of KPIs. These KPIs will be organised depending on whether they are thought for structured, semi-structured or non-structured data. The set of KPIs implemented will be extensive, including generic or inherent KPIs. Generic indicators will include general statistics (e.g., mean, quartiles), counting methods (e.g., repetitions, completeness), or graphical methods (e.g., histograms, violin plots), among others. The output of these KPIs is provided following the DQV specification..
- **User Defined Rules Generator (UDRG):** The functionalities of this element are provided by two components: a Graphical User Interface in the DATAMITE frontend and a backend integrated with the Quality Evaluator (QE) as a Library. These

functionalities are related to the UDRG backend, since the DATAMITE front end will interact with the components of the architecture for accessing the elements that will be part of these rules, including KPIs from the KPI Library, and columns of the dataset or artifact from the Data Governance component, as well as values and conditions established by the users. Once the user has selected the corresponding elements, the backend component will store them following the format defined as an extension of the DQV and related to the DATAMITE metadata model for each dataset.

ddqv: Datamite DQV

The relationship `ddqv:refersTo` from `ddqv:LeftOperand` class can have as range `ddqv:Metric` or `skos:Concept` type of classes, so it can use as `LeftOperand` the "ColumnValue", besides `Metric`.

Instances of `ddqv:Operator` class would be: `gt`, `lt`, `eq`, `gteq`, `lteq`, `in`, `regexMatch`.

Instances of `ddqv:LogicalOperator` class would be: `or`, and

`ddqv_datFormat` is an optional property, and is used to specify the date format in case the field it refers to is a date.

`ddqv_opModifier` is an optional property and is used to modify the value of the field it refers to, by using the arithmetic expression specified in this property.

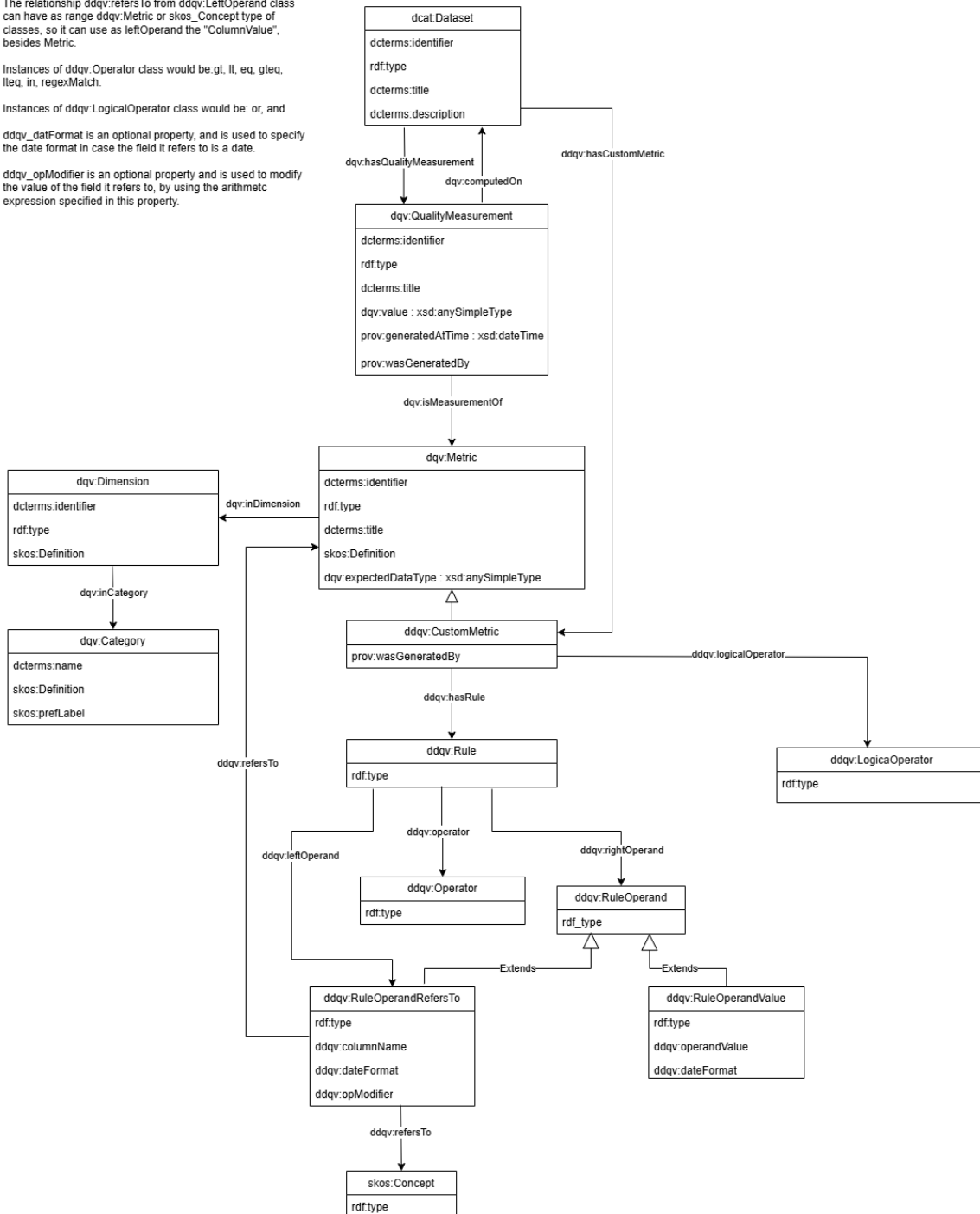


Figure 19: Final DATAMITE Data Quality Vocabulary Schema

The possibility of modifying the rules is presented to the users so that they can modify them or reuse them to create new ones. The evaluation functionality is performed when the QE invokes the UDRG, providing as input parameters, the values of the required KPIs (previously asked and provided by the UDRG component) as well as the dataset identifier together with the data itself. The result of this evaluation will be returned to the QE and stored as part of the quality metadata of the dataset that is being evaluated.

As in the previous section, Table 3 presents the alignment between the functionalities in the Data Quality module and the different requirements that were indicated by relevant stakeholders.

Functionality	Associated Requirements
KPI Library	R029, R099, R101
Quality Evaluator	R027, R029, R030, R098, R100, R105, R106, R110, R111
User Defined Rules Generator	R082, R100, R102

Table 3: Data Quality related Requirements.

3.4.3 Data Sharing

3.4.3.1 Initial Advances

The primary objective of the Data Sharing module is to facilitate secure and transparent data publication within DATAMITE's Data Sharing ecosystem. This module will provide a range of functionalities closely integrated with DATAMITE's IDSA & GAIA-X components and brokers for EU portals and data markets. These functionalities encompass data publication, tools for data and metadata representation, harmonisation, and interoperability. Additionally, a logging tool, similar to IDSA Clearing House, will meticulously record and log details throughout the data sharing process. Furthermore, the module will include features related to payment methods for data billing. Figure 20 depicts the dependencies and interactions within the Data Sharing module, as well as its connections to other DATAMITE modules. Specifically, DATAMITE's Data Sovereignty tools will play a crucial role in ensuring and enhancing data sovereignty throughout the data sharing process. Lastly, the representation, harmonisation and interoperability tools will collaborate with the metadata repository for storing metadata and the data glossary & catalogue management to organise model terms within glossaries efficiently.

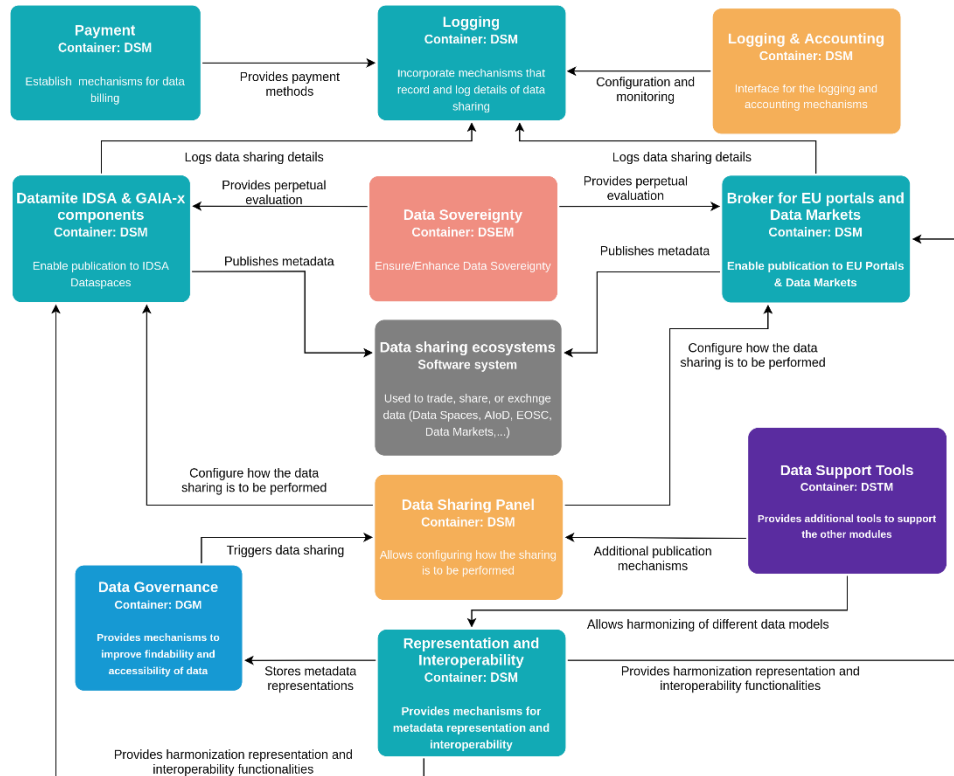


Figure 20: Initial Version of the Data Sharing Module Components Diagram.

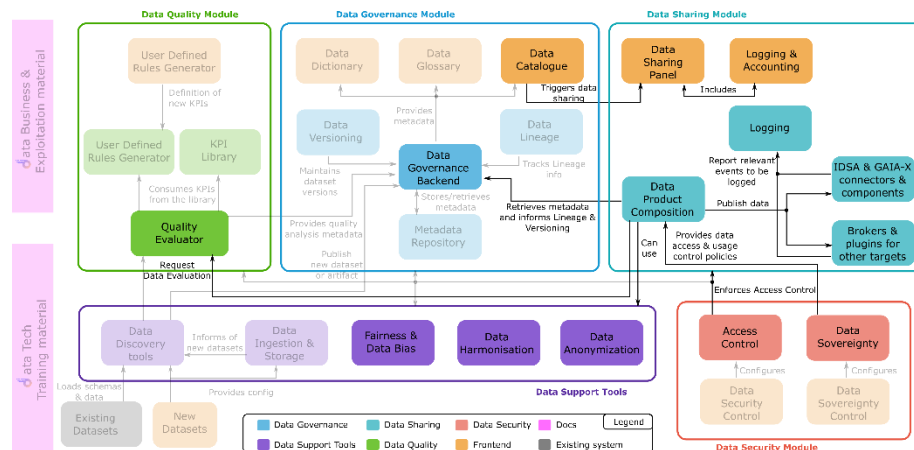


Figure 21: Current Data Sharing Module Components Diagram in the Architecture.

3.4.3.2 Updates up to Month 20

The Data Sharing module is now equipped and fully integrated with clear interactions with the other modules within the framework, enabling it to handle data exchanges and ensure data sovereignty throughout the data lifecycle. To meet the goal of producing valuable and shareable

data products, the Representation and Interoperability component has been replaced with the Data Product Composition component. This new component works with the Data Governance Backend to retrieve metadata from the Metadata Repository. Additionally, the Data Support Tools facilitate data product management and processing, while the Data Quality module ensures quality evaluation. Data sovereignty is maintained by relevant modules, and role-based access control is provided by the Access Control component within the Data Security module. Updates up to Month 30

The Data Product Composition component works in coordination with the Data Governance Backend to transform datasets from the Data Catalogue into data products. The Payment component has been removed. Although the DATAMITE framework initially considered to support payment mechanisms, due to the requirements from Pilot 5 to interact with marketplaces, none of the currently integrated data ecosystems offer charging capabilities, not even the aforementioned scenario in Pilot 5, that interacts with Pontus-X. The current status of the Data Sharing module is illustrated in Figure 20 and Figure 21.

3.4.3.3 Functionalities and Requirements

In this section, we delve into a more detailed analysis of the features and functionalities of each component within the Data Sharing module.

- **DATAMITE IDSA & GAIA-X Components:** The IDSA & GAIA-X components within DATAMITE are poised to bring a range of essential functionalities to the project. Their primary focus is to facilitate data publication to IDSA dataspace. Additionally, these components will significantly contribute to defining and enforcing data usage policies within the project, thereby safeguarding data sovereignty throughout the data sharing process. Lastly, the DATAMITE IDSA connector will play a crucial role in securing the data sharing process, ensuring the necessary credentials are in place for data transfer and onboarding within the IDSA dataspace.
- **Broker for EU portals and Data Markets:** Within DATAMITE, the EU Portals and Data Markets Broker is set to provide an array of vital features for the project. Their primary objective revolves around simplifying the process of publishing data to EU portals and Data Markets. To delve into the specifics, they utilise a comprehensive brokering service aimed at enabling the efficient publication of metadata to prominent European data portals.

- **Data Product Composition:** The primary objective is to aggregate data from various sources and/or retrieve data from the Data Ingestion & Storage system along with its metadata from the Metadata Repository. Additionally, it will provide various functionalities for creating data products, such as extracting specific subsets from datasets based on filters or aggregating datasets by rows or columns. The expected outcome is a compliant data product and its metadata, ready to be shared through DATAMITE's data sharing mechanisms.
- **Logging:** This component enhances transparency and security by incorporating mechanisms to record and log activities throughout data processing and sharing. As a result, it provides users with the ability to monitor actions performed on data within the framework. The auditable logging mechanism within DATAMITE aims to safeguard the confidentiality, integrity, and availability of shared data.

The inclusion of these functionalities within DATAMITE serves a specific and clearly defined role within its framework. Table 4 depicts the connections between the functionalities provided by the Data Sharing module and the DATAMITE 's requirements.

Functionality	Associated Requirements
DATAMITE IDSA & GAIA-X Components	R046, R048, R049, R050, R052, R057, R067, R069, R073, R076
Broker for EU portals and Data Markets	R011, R042, R048, R052, R057, R073, R113, R121,
Data Product Composition	R096, R113, R201, R202, R203, R204, R205
Logging	R013, R014, R016, R017, R018, R074, R091, R118 R122, R123, R124, R196, R197

Table 4: Data Sharing related Requirements.

3.4.4 Data Security

3.4.4.1 Initial Advances

The primary objective of the Data Security component is to bolster the DATAMITE project's security framework. This component encompasses a range of security-related functionalities, including data sovereignty, access control, and overall security measures. Access control and security, while distinct, complement each other within this component. Access control empowers

fine-grained access control and authorisation, ensuring that only authorised individuals or entities have access to specific data resources. On the other hand, the security aspect encompasses a variety of functions, including security analysis and privacy safeguards. For instance, it can perform security analysis to identify potential vulnerabilities and address them proactively. Additionally, the data sovereignty element within this component provides features aimed at enhancing DATAMITE's data sovereignty. Figure 22 provides a visual representation of the interactions within the Data Security component and its connections with other DATAMITE modules, showcasing how security is integrated into the broader DATAMITE ecosystem.

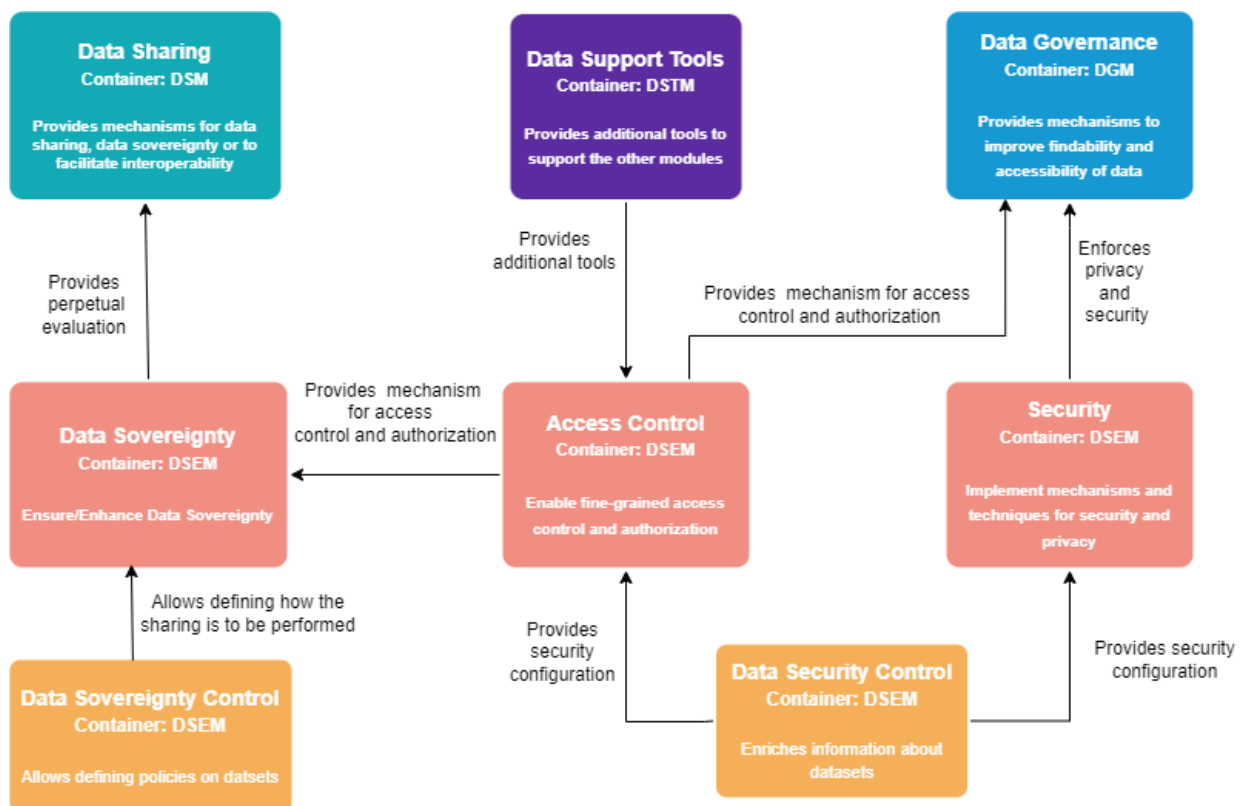


Figure 22: Initial Version of the Data Security Module Components Diagram

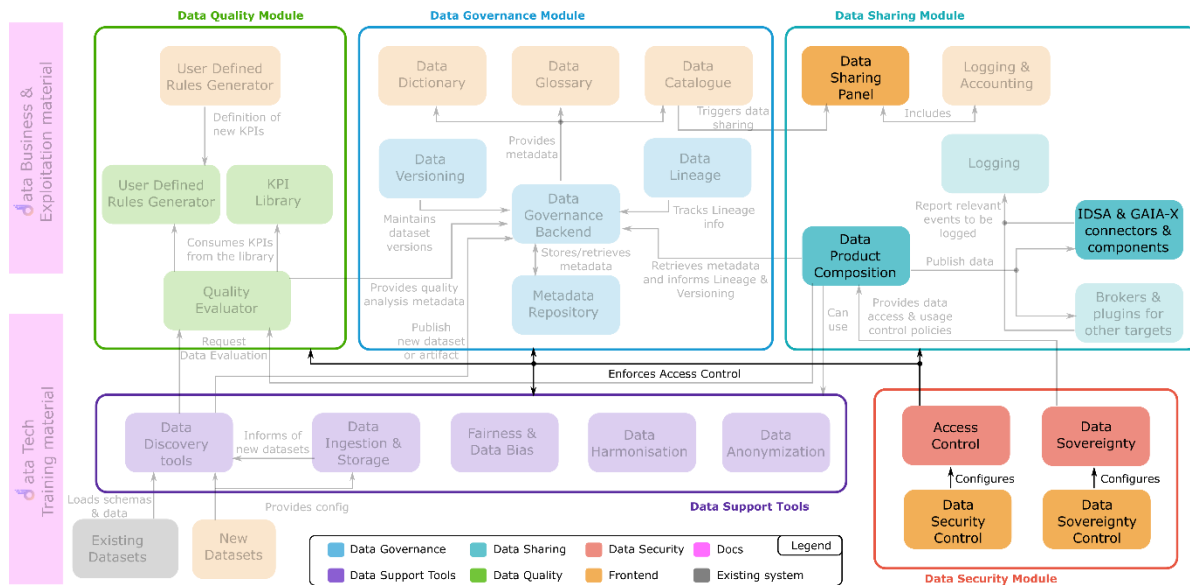


Figure 23: Current Data Security Module Components Diagram in the Architecture.

3.4.4.2 Updates up to Month 20

The update involves removing the Security component, as the Access Control component now fully covers the security aspect within the DATAMITE framework. The Access Control component is responsible for user management and ensures fine-grained access control. Data sharing mechanisms ensure security during the data exchange process. As a result, the Access Control module interacts with all other modules to enforce role-based access control. Additionally, the Data Sovereignty module interacts through the Data Sharing Panel with the Data Product Composition to pair the data product with usage policies and access rules for data consumption.

3.4.4.3 Updates up to Month 30

Figure 24 shows the latest architecture for the Data Security module. There has not been any major change since Month 20.

3.4.4.4 Functionalities and Requirements

In this section, we delve into a more detailed analysis of the features and functionalities of each component within the Data Security module.

- **Data Sovereignty:** Data sovereignty plays a crucial role in the data sharing process. To further bolster data sovereignty within this process, this component will introduce innovative features and functionalities to the DATAMITE framework. Its goal is to

establish transparent usage policies related to data and facilitate seamless communication between data providers and consumers. Specifically, the Data Sovereignty module will translate machine-readable usage policies stored in the ODRL information model into a human-readable format, enabling users to understand and choose the most appropriate data contract based on their needs. Additionally, the module will provide and enforce usage policies for automated data agreement processes, such as the exchange process with the EDC connector.

- **Access Control:** This component will encompass features pertaining to access control and authorisation mechanisms. Its objective is to implement mechanisms that provide fine-grained access control and authorisation capabilities, thereby ensuring the protection of sensitive data assets.

The Data Security module of DATAMITE has specific functionalities that correspond to the framework's requirements. Table 5 below shows how each functionality relates to a requirement of DATAMITE.

Functionality	Associated Requirements
Data Sovereignty	R009, R053, R054, R055, R056, R058, R059, R091, R120, R121, R122, R123, R174
Access Control	R059, R072, R116, R117, R118, R119, R133, R138, R152

Table 5: Data Security related Requirements.

3.4.5 Data Support Tools

3.4.5.1 Initial Advances

The main purpose of Data Support Tools is to provide the necessary interactions among the architectural modules of DATAMITE. Particularly, Data Support Tools constitute a collection of software tools and modules that enable or support dissimilar interconnections and functionalities that are needed among the different structural blocks within the DATAMITE's architecture.

Therefore, Data Support Tools feature a plethora of backed capabilities, such as interoperability, security, data quality, data sharing, and governance. These tools complement and form required synergies across the rest of the components in the DATAMITE's architecture. They also ensure that these components' interactions are adequately addressed where required, envisioning a unified architectural structure. Figure 24 and Figure 25 present the different components proposed to be part of the data support tools initially, and their latest version.

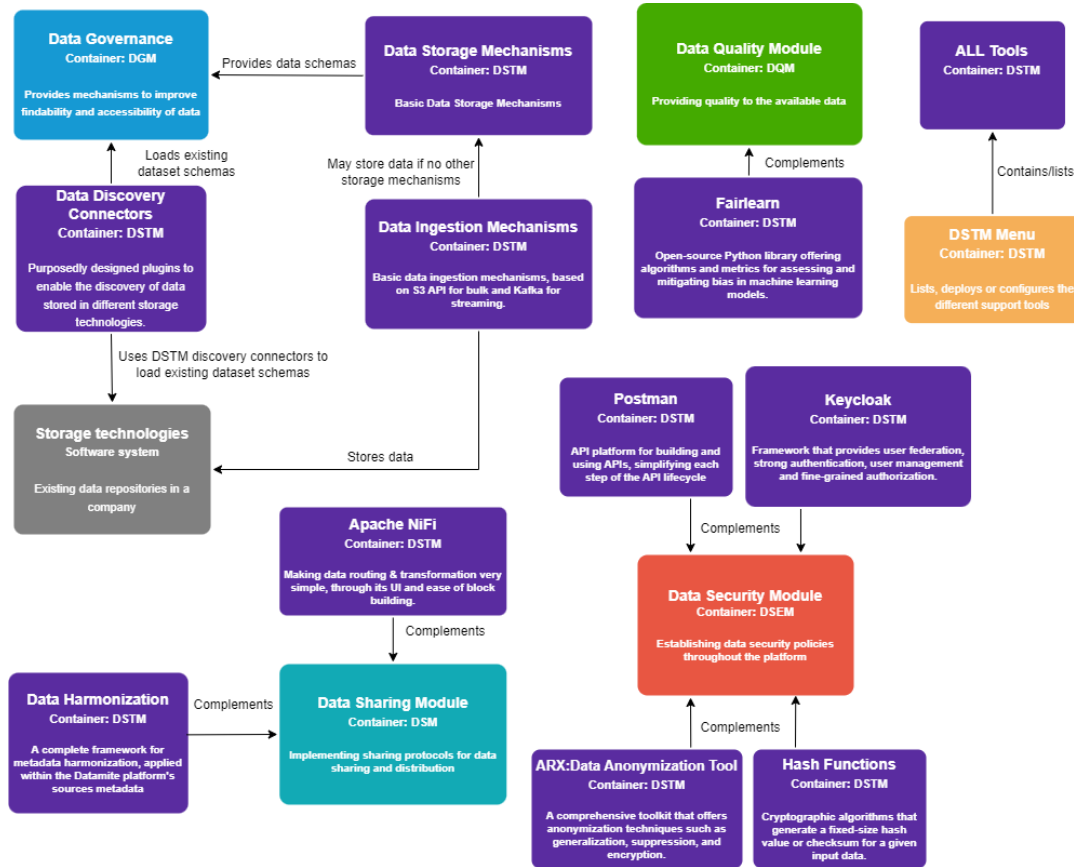


Figure 24: Initial Version of the Data Support Tools Module Component Diagram.

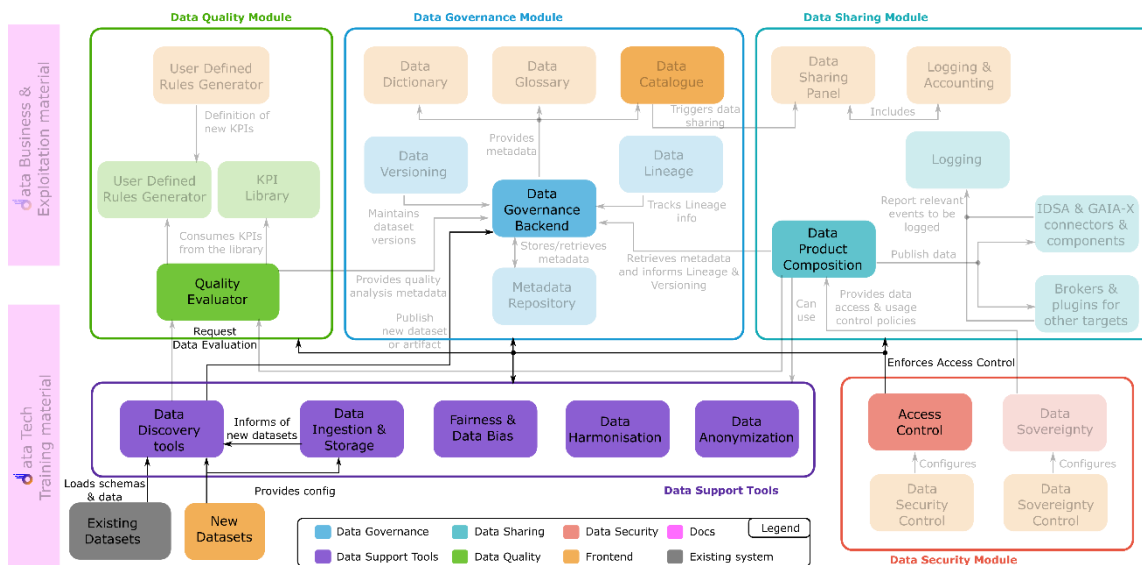


Figure 25: Current Data Support Tools Module Components Diagram in the Architecture

3.4.5.2 Updates up to Month 20

The data support tools have undergone a severe revision of the functionalities they comprehend. The initial version of the architecture and the component diagram are depicted in Figure 24, including as functionalities the Data Discovery connectors, Data Ingestion and Data Storage mechanisms, Data Harmonisation, Fairlearn, ARX data anonymisation tool, Hash functions, Keycloak, postman and Apache NiFi. Some of these functionalities have gone through some cosmetic changes but keep the same essence. This would be the case of Fairlearn, now Fairness and Data Bias; ARX data anonymisation tool and Hash functions, now denoted as Data Anonymisation; or Data Storage and Data Ingestion mechanisms, now unified in the same component. Data Discovery tools and Data Harmonisation remain as they were. However, the rest of the components have been eliminated or relocated. Postman was eliminated even when it can be leveraged as an external tool but does not belong to the framework. Similarly, Apache NiFi is more of a transversal tool to be used but not provided as a DATAMITE tool. Mage.ai is actually performing the role that Apache NiFi could have fulfilled, assisting with pipeline deployment. Keycloak is being used not as a tool to be offered but as part of the Data Security module. These tools can be used, in general, in isolation and on-demand, although they are being integrated into different flows. For instance, one potential option to consider is the integration of the Data Anonymisation tool into the data product composition flow. Data Harmonisation and Fairness and Data Bias have not yet been integrated into these flows, but they will.

3.4.5.3 Updates up to Month 30

Figure 25 shows the latest architecture for the Data Support Tools module. There has not been any major change since Month 20.

3.4.5.4 Functionalities and Requirements

The Data Support Tools module has a series of functionalities based on the interconnection with the other components/modules and the software it provides to assist them. These functionalities are:

- **Data Discovery Tools:** Data Support Tools encompass specialised plugins specifically crafted to facilitate the identification and retrieval of data stored within diverse storage technologies. These plugins serve as valuable instruments to empower users in the

seamless exploration and access of data residing across a wide array of storage platforms, ensuring efficient and comprehensive data management.

- **Data Ingestion and Storage:** Data Support Tools provide Data Ingestion and Storage Mechanisms to complement the data discovery tools as a source of new data in the framework. Data ingestion mechanisms will allow users to ingest data in bulk, utilising the S3 API for large-scale batch data transfers or streaming using a plugin-based approach offering different protocols (e.g., Kafka, MQTT, ModBus or OPC-UA). The storage mechanisms allow the user to store the data received through the ingestion mechanisms and to store data products created within the framework.
- **Fairness and Data Bias:** Data Support Tools include this data quality-related tool, an open-source Python library that provides a wide array of algorithms and metrics and offers a comprehensive framework for evaluating and mitigating bias in machine learning models. By leveraging Fairness and Data Bias, data practitioners can systematically assess model fairness, identify disparities, and implement interventions to rectify biases, thereby enhancing the ethical and equitable deployment of machine learning solutions.
- **Data Harmonisation:** An integral sub-component within Data Support Tools will comprise a comprehensive framework dedicated to Metadata Harmonisation, seamlessly applied in the DATAMITE platform's source metadata. This framework represents a pivotal aspect of data sharing and governance, ensuring consistency and coherence across diverse sources within the DATAMITE ecosystem. It not only harmonises metadata attributes but also establishes standardised protocols for data description, classification, and lineage tracking.
- **Data Anonymisation:** A comprehensive tool known for its wide range of anonymisation and pseudonymisation techniques including generalization, masking, suppression, format-preserving transformations, substitution, hashing, tokenization, and the application of privacy models such as k-anonymity, l-diversity, and differential privacy enables users to effectively safeguard sensitive information. Users can transform data, delete sensitive columns, generalize, and conceal specific fields. The anonymisation tool, part of the Data Support Tools suite, enhances data resilience, strengthens data security, and ensures the reliability of data operations ultimately contributing to a robust data ecosystem. It empowers organisations to comply with data privacy regulations by concealing personally identifiable information while preserving the utility of the data.

The inclusion of these functionalities within DATAMITE's requirements are depicted in **Table 6**.

Functionality	Associated Requirements
Data Discovery Connectors	R028, R045, R144, R158, R159
Data Ingestion and Storage Mechanisms	R028, R139, R141, R143, R156, R157, R162, R163
Data Harmonisation	R021, R035
Data Anonymisation	R130, R194, R195, R200, R204
Fairness & Data Bias	R101, R108
Other internal tools (e.g., Mage.ai)	R160, R198, R199

Table 6: Data Support Tools related Requirements.

3.4.6 Frontend

Besides the main DATAMITE modules that have been defined in the previous sections, there are two more relevant aspects to be considered: the front end and the general management of the platform.

The front end is not a module itself, but it provides the user interface for DATAMITE modules or the connection point between them. For instance, it allows for the configuration of data quality configurations and their execution from the data catalogue. Although the main components of the front end have already been pointed out in the previous sections, Figure 27 presents a recollection.

This is aligned, in fact, with the general management of the platform. There are features or functionalities which are not listed in the frontend diagram, even when they are already being considered. These features are related to the availability of the services, the authentication, cookie management, content management, the definition of users or the general configuration of the platform. They are present in the requirements proposed by the stakeholders and listed in Table 7.

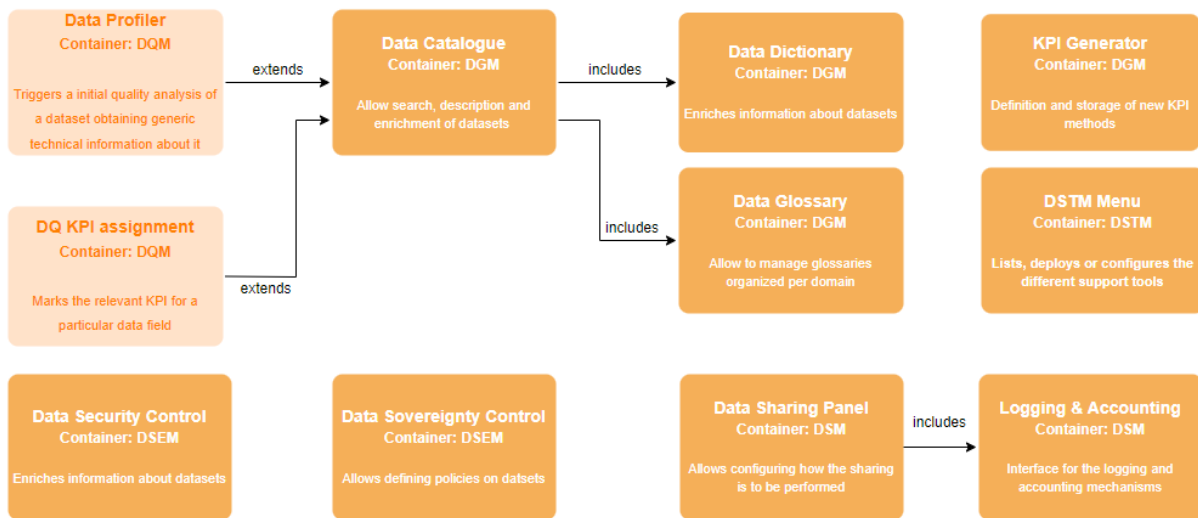


Figure 26: Initial Proposal of the Main Elements in DATAMITE's Frontend.

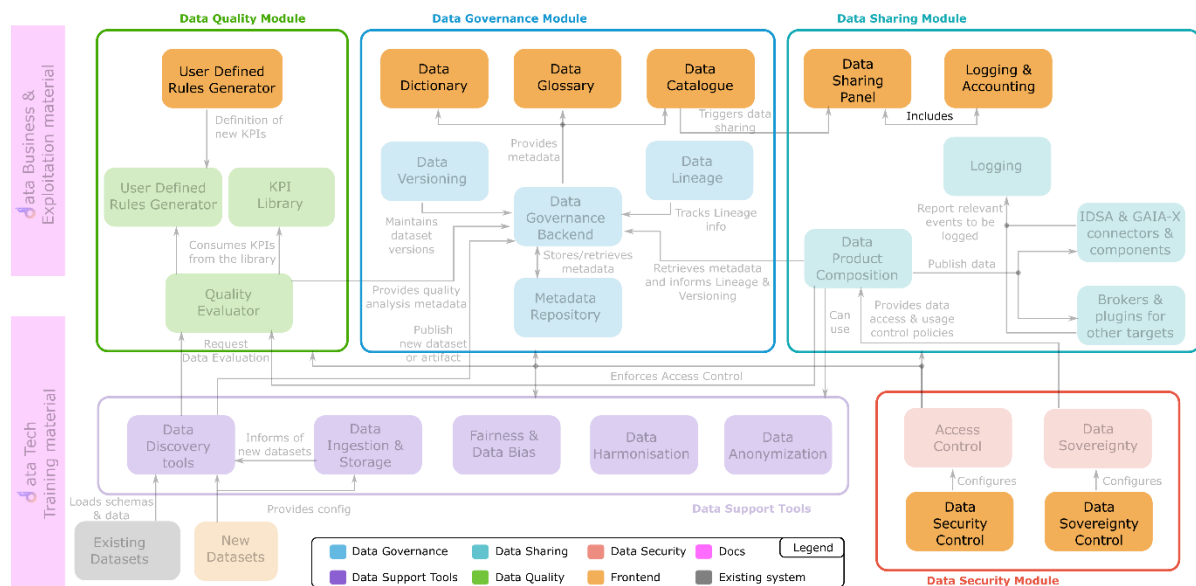


Figure 27: Current Frontend Components in DATAMITE's Architecture.

3.4.6.1 Updates up to Month 20

Figure 27 depicts the initial proposal of front-end interfaces for the different components in the architecture. As can be observed in Figure 28, these interfaces have not evolved significantly, during the last months, as the focus has been mostly on the development of the backend

components. Consequently, it can be observed that the most significant variations have been the following:

- The addition of the (add) new datasets interface indicates that this is the point of entry for new datasets into the system, either to store them or to connect to an existing database.
- The evolution of the data quality interfaces is related to the evolution of the KPI generation interface to the user-defined rules generator. There are some aspects to be defined yet on how the data quality interfaces will evolve, regarding the configuration of the profiling, for instance.
- The DSTM interfaces have not been included in Figure 28 for the sake of simplifying the figure, although, in fact, they will include their own interfaces to ease their configuration and usability.

3.4.6.2 Updates up to Month 30

- Several supporting features have been introduced that were not part of the original proposal. These include the FAQs and documentation area, which help users navigate the platform with ease, and the notification system, which keeps them informed in real time about key updates like data publications or rule checks. The inclusion of these elements reflects a broader focus on user experience.
- Security and sovereignty features have also started to take shape on the frontend. Integration with Keycloak for login and role management is developed.
- While the frontend has not undergone major visual changes, these targeted improvements are laying the foundation for a more functional, secure, and user-friendly experience in line with the DATAMITE framework's vision.

3.4.6.3 Functionalities and Requirements

The Frontend module has a series of functionalities based on the interconnection with the other components/modules. These functionalities are:

- **Data Catalogue:** It is the user interface to find datasets, manage them, enrich them with metadata or terms from the data glossary, perform data quality-related operations, trigger the creation of data products, or share them, among other actions. At the data product level, the user has the option to anonymise specific content in order to preserve privacy before sharing it.

- **Data Glossary:** Provides access to domain-specific vocabularies. The users themselves may have defined these vocabularies or may be standards from the community imported into the framework. Users may also extend these vocabularies with new terms.
- **Data Dictionary:** Provides an overview of the dataset, potentially including aspects like data types, statistics, completeness, or other indicators that can give an initial idea of its quality.
- **Data Sharing Panel:** The data products can be published in external platforms, like AI on Demand. Through this panel, the user can configure the metadata and monitor the publication status.
- **User-Defined Rules Generator:** In the KPI definition, the user has indicators that monitor the dataset quality. Users can define validation rules or constraints for artifacts based on project-specific or domain-specific requirements. Users may impose logical, statistical, or regulatory constraints on data attributes to guide their processing and interpretation.
- **Data Security Control:** Ensures controlled access to the framework, supporting login protocols and single sign-on mechanisms. Also, the framework has the role-based access control functionality where permissions are defined per user role, enforcing access restrictions for actions such as data modification, rule creation, or product publication.
- **Data Sovereignty Control:** Allows data owners to stay in control of where and how their data is stored and used. It helps ensure that data handling respects legal, organisational, and geographical boundaries. This way, organisations can confidently share and manage data across different environments, knowing that the rules they set are being followed. Regarding the data products, the interface allows users to define and associate usage policies and contractual terms with data products to ensure proper governance and compliance.
- **Logging & Accounting:** This feature supports secure data sharing via external portals. Within the data sharing panel, users receive an authentication token that allows them to connect with the external publication environment. At the same time, all significant actions—such as dataset sharing or publication—are recorded by the framework. This logging process ensures traceability, transparency, and accountability, all of which are



essential for maintaining trust and regulatory compliance within the DATAMITE framework.

- **New Datasets:** The framework includes a clear, guided process for adding new datasets. It begins by asking users to provide essential information such as the dataset's name, a short and full description, and to link it with relevant domains and glossary terms already defined in the framework. Users then choose the method of creation—whether by uploading in bulk, connecting a real-time data stream, or establishing a link to an existing database. In the next step, access permissions are configured, allowing roles previously defined in the system to be granted view, share, edit, or delete rights. Finally, a confirmation screen provides an overview of all inputs before submission, helping ensure the dataset is properly structured and securely registered.

The framework also includes a dedicated section for FAQs and documentation, providing users with thematically organised Frequently Asked Questions (FAQs) presented through an accordion interface, alongside detailed guidance on platform usage, such as policy configuration, security settings, and technical operations. A notifications system delivers real-time alerts and updates, including confirmations of data publication, outcomes of rule validations, and other relevant framework news. In addition, the framework frontend offers user-specific configuration settings, enabling the management of cookie preferences, language selection, and accessibility features, all in alignment with the WCAG 2.2 accessibility standard.

Functionality	Associated Requirements
General framework requirements	R002, R125, R154, R155, R165, R166, R167, R168, R169, R170, R173, R175, R176, R179, R180, R181, R183, R184, R185, R186, R188, R189, R190, R191

Table 7: Requirements related to the General Management of the Platform.

4 Conclusions

This document has updated the project requirements and the DATAMITE's architecture.

Regarding the requirements, Section 2 presented the methodology applied in the revision, which is a continuation of what was presented in the M9 deliverable D1.2, as well as how the analysis and classification have been performed, providing information on the different types of requirements, the stakeholders that were polled or the most relevant categories identified.

Section 3 introduced the current detailed architecture version, following the C4 approach. This section analysed the framework as a whole and performed a top-down analysis from this high-level view to a per-module analysis, detailing the components and some of the functionalities. Each of the Modules has evolved since the last version, and changes are specified for each one of them.

Finally, it is important to mention that in this analysis, a table relating each one of the DATAMITE modules to the corresponding reviewed requirements was presented.

5 Appendix

5.1 Requirements Tables

The following tables contain the full description of the collected requirements:

ID	R007
Type	Non-Functional
Source	External Stakeholders
Category	Data Monetisation
Description	The framework could provide the option to sell data according to various charging methods
Rationale	Exploitation of the available data
Priority	COULD
Test Case / Acceptance Criteria	Verify that a Data Consumer can choose the most suitable method from the available charging methods which are set by the Data Provider (e.g. payment per dataset / monthly subscription option), agree to the terms, pay and download the dataset

ID	R009
Type	Non-Functional
Source	Pilot Analysis
Category	Data Sharing
Description	The connectors from the data sharing module could be able to provide the pilot 4 data requesters characteristics in order to implement data access policies
Rationale	Define user access policies
Priority	COULD
Test Case / Acceptance Criteria	1st time a dataset is retrieved from dataspace from a specific user, the information about the retrieval could be available

ID	R011
----	------

Type	Functional
Source	External Stakeholders
Category	Data Monetisation
Description	The Data sharing module should allow publishing/advertising datasets to European Data Platforms and Markets (such as AIoD or EOSC)
Rationale	Access to diverse European Data Platforms and Markets will increase the chances of data monetisation for stakeholders
Priority	SHOULD
Test Case / Acceptance Criteria	All datasets flagged by data providers to be published to a specific EU data portal should be periodically published/updated to that specific EU data portal

ID	R012
Type	Functional
Source	External Stakeholders
Category	Data Monetisation
Description	As a data provider, I want to be able to select in which European Platforms to publish the metadata of my dataset(s)
Rationale	Exploitation
Priority	MUST
Test Case / Acceptance Criteria	Data providers must be able to choose zero or more EU data portals where to publish datasets that are being onboarded. Data providers must be able to select, for each onboarded dataset, zero or more EU data portals where DATAMITE should publish (the metadata) of the dataset

ID	R013
Type	Functional
Source	External Stakeholders
Category	Data Sharing
Description	The data logging tool from the data sharing module should provide and/or quantify data usage metrics (direct metrics, metrics of data shared by other providers/users, consumption (download) in order to analyse the benefits of sharing data
Rationale	Exploit the usage of datasets shared with different portals and data spaces

Priority	SHOULD
Test Case / Acceptance Criteria	1st time the dataset is retrieved from dataspace, the information about the retrieval should be available

ID	R014
Type	Functional
Source	Pilot Analysis
Category	Data Monetisation
Description	As the provider of domain service in agriculture, I would like to understand the statistics about the data usage (downloads, contracts) and have information about the clients and their profiles, to improve the offer and user experience and better targeting the clients
Rationale	Impacts exploitation as the clients' profiles will help to better fit their needs
Priority	SHOULD
Test Case / Acceptance Criteria	Client profile shown in the Data marketplace interface of the pilot 5 for marketplace owner/administrator role (can query logging tool API)

ID	R015
Type	Functional
Source	External Stakeholders
Category	Data Monetisation
Description	As a data provider, I want to be able to set a price for each license/usage policy (including free access, meaning zero price)
Rationale	Exploitation
Priority	COULD
Test Case / Acceptance Criteria	When a data provider onboards a dataset and selects the license/usage policy, including, if the policy is anything other than free access, it must be possible to also set the price for each unit under that usage policy

ID	R016
Type	Functional
Source	External Stakeholders

Category	Data Monetisation
Description	As a data provider, I want to be able to see the agreed policies (purchased contracts) for my dataset(s)
Rationale	Exploitation
Priority	SHOULD
Test Case / Acceptance Criteria	Data providers should be able to see, for each onboarded dataset, the agreed policies (purchased contracts), including the usage of the data under each agreed policy

ID	R017
Type	Functional
Source	External Stakeholders
Category	Data Monetisation
Description	As a user, I want to be able to see the data access policies I agreed to (contracts I purchased)
Rationale	Usability and availability
Priority	SHOULD
Test Case / Acceptance Criteria	Users that want to consume datasets and have agreed to the policies (purchased contracts) for one or more datasets, should be able to see the agreed policies (one or more per dataset)

ID	R018
Type	Functional
Source	External Stakeholders
Category	Data Monetisation
Description	As a user, I want to be able to see the resources I can still consume (download) according to my agreed policies (contracts)
Rationale	Usability and availability
Priority	SHOULD
Test Case / Acceptance Criteria	Users who want to consume datasets and have agreed to the policies (purchased contracts) for one or more datasets, should be able to see the usage of the data under each agreed policy (one or more per dataset), including the remaining data they can still consume under each agreed policy

ID	R019
Type	Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework requires a data glossary that can be used to define different vocabularies (probably coming from different domains), composed of terms that define concepts in a univocal way (to make it easier for companies to use a single term for each concept)
Rationale	The use of a glossary is needed to facilitate or enable semantic interoperability. This glossary may contain different vocabularies organised per domain
Priority	MUST
Test Case / Acceptance Criteria	There exist several vocabularies in the glossary, and they can be or are identified with at least one domain

ID	R020
Type	Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework should be able to establish relations between terms of different vocabularies
Rationale	Different vocabularies, or even the same vocabulary, may have terms that are similar and whose relation may be relevant to point out
Priority	SHOULD
Test Case / Acceptance Criteria	There exists the option to establish links between terms within the same or among different vocabularies (synonym, antonym...)

ID	R021
Type	Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework must provide mechanisms for data harmonisation

Rationale	Ensure data harmonisation within the framework for achieving consistency and coherence of diverse datasets. It enables the reconciliation of differences in data formats, structures, and semantics, promoting a unified view across various sources
Priority	MUST
Test Case / Acceptance Criteria	Demonstrate that the framework can harmonise and present data in a unified manner and eliminate inconsistencies

ID	R022
Type	Non-Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework must utilise ontologies and standard modelling vocabularies for improved interoperability
Rationale	Facilitate the adoption of common semantic standards and enhance interoperability across diverse domains
Priority	MUST
Test Case / Acceptance Criteria	Demonstrate the use of standard ontologies and vocabularies, relevant across various domains

ID	R023
Type	Non-Functional
Source	External Stakeholders
Category	Interoperability
Description	As a data hub provider/domain provider, I would like to provide the data harmonised in a standardised and interoperable way, making it compatible on syntax and semantic level, so that its usage can be maximised by other providers and potential users in the domain
Rationale	Common data format will improve compatibility and reuse
Priority	MUST
Test Case / Acceptance Criteria	Demonstrate data providers provide data based on standard data models (ontologies, vocabularies) and formats

ID	R024
Type	Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework must design an internal metadata model that covers various aspects (e.g., data quality) and is informative
Rationale	Ensure comprehensive and standardised metadata representation, enabling better data management and understanding
Priority	MUST
Test Case / Acceptance Criteria	The internal metadata model is nearly ready for publication with minor changes, and it provides informative details about the data

ID	R025
Type	Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework should organise model terms in glossaries for better clarity and reference
Rationale	Improve the consistency and understanding of terms used in the system and enhance communication among stakeholders
Priority	SHOULD
Test Case / Acceptance Criteria	Implement glossaries to enhance consistency, understanding of terms, and improve communication

ID	R026
Type	Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework should enable the registration of metadata representations with dedicated broker services
Rationale	Facilitate the discoverability and accessibility of structured metadata, promoting interoperability and data discovery

Priority	SHOULD
Test Case / Acceptance Criteria	Provide an internal metadata repository

ID	R027
Type	Non-Functional
Source	Internal Technical Analysis
Category	Quality
Description	The Data Quality Module must be independent of any data-source format
Rationale	The component's internal structure of data processing, as well as its storing mechanisms, must be implemented in a way that every future data source can be supported
Priority	MUST
Test Case / Acceptance Criteria	When the Data Quality Module is ready for operation, it must be able to seamlessly profile incoming datasets from different sources, formats, and types (e.g., JSON, CSV). For example, the Data Quality Module must be able to profile datasets from both the Telecommunications and the Energy sector

ID	R028
Type	Functional
Source	Pilot Analysis
Category	Interoperability
Description	Different data formats such as JSON, CSV and XML could be normalised to the internal data model (compliant to BUFR specifications), enabling a unified schema ingestion
Rationale	Currently, each format is ingested with a different schema, it would be useful to have one unified schema through a pluggable adapter
Priority	COULD
Test Case / Acceptance Criteria	It will be possible to ingest datasets in different formats or protocols, both for streaming and bulk update

ID	R029
----	------

Type	Non-Functional
Source	Internal Technical Analysis
Category	Quality
Description	The Data Quality Module should have an open API, so that new big vendors & providers can benefit and have the ability to publish and extend their services
Rationale	The Data Quality Module should be easy to understand, use, deploy and (in the future) improve
Priority	SHOULD
Test Case / Acceptance Criteria	For this requirement to be considered as accepted, the Data Quality Module has to provide the end-users with the ability to easily interact with it (the Data Quality Module). This can be achieved with the existence of a UI, with which the Data Quality Module will seamlessly communicate. In order to be a component that others will use (but furthermore easily maintain), it also has to be developed using a high-level programming language that is understandable by the majority of the new generation's software engineers. In addition, it has to be in a form of a package (such as a containerised Docker package), in order to be easily deployable

ID	R030
Type	Non-Functional
Source	Internal Technical Analysis
Category	Quality
Description	The Data Quality Module component should be Big Data compliant
Rationale	Big Data compliance is vital, in cases where components have to deal with large volumes of data, now or in the future
Priority	SHOULD
Test Case / Acceptance Criteria	The Data Quality Module's main functionality is to apply a series of pre-defined KPIs, in order to extract quality-related insights from a dataset. Furthermore, the module shall have an additional operational flow, where the end-user will define his/her own KPIs, in order to further profile the selected dataset. However, the Data Quality Module's aim is to (also) operate using whole datasets as inputs, rather than samples of those sets. This way, users will have a definitive quality overview of each dataset.

ID	R031
Type	Non-Functional

Source	Internal Technical Analysis
Category	Interoperability
Description	The pilot data should follow a specific structured format (e.g., DB schema, CSV file), especially for data harmonisation purposes
Rationale	To allow for the efficient and effective processing of the pilot data and interlinking with relevant data models
Priority	SHOULD
Test Case / Acceptance Criteria	The pilot data should be readable via the appropriate software tools or query languages

ID	R032
Type	Non-Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	For data harmonisation purposes, the pilot data values of non-numeric fields should belong to either local or international terminologies, classifications, vocabularies or coding systems, with the terms being well described in a format that both humans and software agents can understand
Rationale	To allow the semantic processing and interlinking of the data sources and the meaningful inference of results
Priority	SHOULD
Test Case / Acceptance Criteria	The distinct set of terms/codes used for the expression of the dataset should be well defined. A meaningful description should be provided for each of them

ID	R033
Type	Non-Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework should provide/support the appropriate metadata models for also capturing all the important parameters of pilot data and their features for data harmonisation purposes

Rationale	The metadata model/fields must be specified in advance so that all important parameters are recorded for each pilot data
Priority	SHOULD
Test Case / Acceptance Criteria	The metadata model should be developed taking into account all data description needs and relevant metadata models in existing publicly available frameworks and/or data markets

ID	R035
Type	Non-Functional
Source	External Stakeholders
Category	Interoperability
Description	The data support tools should provide means for data and metadata harmonisation based on a given description
Rationale	The data harmonisation will lead to better interoperability and easier exploitation of the datasets
Priority	SHOULD
Test Case / Acceptance Criteria	Given a set of descriptive guidelines, the tools should be able to adhere to and harmonise metadata or data of a given dataset

ID	R037
Type	Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	Each dataset should be accompanied by a common schema of metadata
Rationale	The common metadata schema should increase interoperability and allow for easy publishing to EU data portals
Priority	SHOULD
Test Case / Acceptance Criteria	The metadata schema should be descriptive enough for datasets of various fields as well as contain as much information as possible to allow easy publishing to EU data portals

ID	R038
Type	Functional

Source	Internal Technical Analysis
Category	Data Governance
Description	The data model of the dataset metadata in the DATAMITE catalogue must contain all fields needed to publish to the selected data markets
Rationale	This is a technical requirement that makes the requirement with ID R012 possible
Priority	MUST
Test Case / Acceptance Criteria	The metadata returned by the DATAMITE catalogue API must contain all mandatory fields of all the supported destinations EU data portals

ID	R039
Type	Functional
Source	Internal Technical Analysis
Category	Interoperability
Description	The framework must internally store the pilot metadata in accordance with the metadata model developed
Rationale	This is a technical requirement that makes the requirement with ID R038 possible
Priority	MUST
Test Case / Acceptance Criteria	The metadata captured/stored for a particular dataset must be accessible by the framework using the elements of the metadata model. Make sure the metadata of these datasets contain all the required fields mandated by the DATAMITE catalogue

ID	R042
Type	Non-Functional
Source	External Stakeholders
Category	Data Sharing
Description	The data sharing module should allow the user to publish data to open source repositories like OpenML or Zenodo in an easy manner
Rationale	Enhancing interoperability with other data platforms of wide usage such as OpenML or Zenodo, will increase the use of both DATAMITE and the data accessible through the framework
Priority	SHOULD

Test Case / Acceptance Criteria	Upon the user's decision of sharing data on the open source platforms, the data should be accessible from those platforms
---------------------------------	---

ID	R044
Type	Functional
Source	Pilot Analysis
Category	Data Governance
Description	As a data provider, I would be interested in providing information about the underlying data models and ontologies used by the published data as part of its metadata, so that consumers of my data can understand it better, e.g., what are the fields, what are their meaning, what are the units of measurement used, etc.
Rationale	Exploitation and reuse, as the consumers of data will be able to better understand the data and reuse it for their purposes
Priority	SHOULD
Test Case / Acceptance Criteria	Metadata contains fields pointing to the data model/vocabularies used by the dataset

ID	R045
Type	Functional
Source	Pilot Analysis
Category	Data Governance
Description	The end user should have the capability to access data products offered by different data providers and the response of the outcomes should be retrieved during a timeframe that will allow the business to operate as usual
Rationale	This business requirement will allow users to combine information in an efficient way
Priority	SHOULD
Test Case / Acceptance Criteria	Metadata from more than one provider in the catalogue

ID	R046
----	------

Type	Functional
Source	Pilot Analysis
Category	Interoperability
Description	As a data provider/provider of domain service, I would like to validate if the data are following properly the standard / ontology (like AIM in the case of the agriculture domain)
Rationale	Support automated data validation mechanisms, to ensure that datasets published on the platform adhere to quality standards, enhancing the trustworthiness of the data
Priority	MUST
Test Case / Acceptance Criteria	When the user uploads datasets, compatibility with standards and ontologies on their domain can be checked

ID	R047
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	As a DATAMITE developer, I want the DATAMITE catalogue to have an open API, so that I can use it to find datasets and obtain their metadata
Rationale	This is a critical requirement. Defining the datasets (their metadata) as the core entity that DATAMITE handles, and making the DATAMITE catalogue the central repository of these entities is the foundation of the entire project
Priority	MUST
Test Case / Acceptance Criteria	The DATAMITE catalogue must have an API to perform at least the following queries, needed to publish datasets to the platforms defined in Task 2.2: 1) list all datasets, 2) list datasets that should be published to a specific EU data portal, 3) get all the metadata of a specific dataset.

ID	R048
Type	Functional
Source	Pilot Analysis
Category	Data Sharing

Description	The framework should offer APIs and data connectors to enable seamless integration with external applications, facilitating data exchange and analysis
Rationale	To allow users to leverage the framework's data in their existing systems and workflows, promoting wider adoption
Priority	SHOULD
Test Case / Acceptance Criteria	Data will be shared or exchanged through IDS based connectors (e.g., the EDC connector) or connectors customised for different portals (e.g., the AloD)

ID	R049
Type	Functional
Source	Pilot Analysis
Category	Interoperability
Description	As an agriculture data provider as part of Pilot 5: Connecting eDWIN to Data Markets, I would like to transform existing data into AIM (Agriculture Information Model)-compliant format, to enable other systems to consume it
Rationale	Usability and interoperability, enabling different systems to interoperate and to analyse data produced by those systems in an integrated manner
Priority	SHOULD
Test Case / Acceptance Criteria	Connecting eDWIN to Data Markets, users can utilise conversion tool to adapt the data to standard formats.

ID	R050
Type	Functional
Source	Pilot Analysis
Category	Interoperability
Description	As an agriculture service provider, as part of Pilot 5: Connecting eDWIN to Data Markets, I would like to generate AIM-compliant data, to enable other systems to consume it
Rationale	Usability and interoperability, enabling different systems to interoperate and to analyse data produced by those systems in an integrated manner
Priority	SHOULD

Test Case / Acceptance Criteria	Connecting eDWIN to Data Markets, the user can generate the data supporting standard formats
---------------------------------	--

ID	R051
Type	Functional
Source	Pilot Analysis
Category	Interoperability
Description	As an agriculture service provider as part of Pilot 5: Connecting eDWIN to Data Markets, I would like to provide access to AIM data via standard APIs (e.g., OGC), to facilitate the interoperability with other systems/platforms. For example, measurements and features via OGC SensorThings API and OGC Features APIs
Rationale	Usability and interoperability, enabling different systems to use standard APIs to access data from different sources
Priority	SHOULD
Test Case / Acceptance Criteria	Connecting eDWIN to Data Markets, users can utilize the API to access the data.

ID	R052
Type	Non-Functional
Source	Pilot Analysis
Category	Data Sharing
Description	As a data provider, I want to be able to share data in a standardised form, both at the syntactic (e.g., JSON) and semantic level (e.g., AIM for agriculture that is based on OGC and W3C standards). This applies to all pilots
Rationale	Usability and interoperability, enabling different systems to interoperate and to analyse data produced by those systems in an integrated manner
Priority	SHOULD
Test Case / Acceptance Criteria	When sharing his data, the user can choose between different standard forms

ID	R053
----	------

Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	Convert common principles defined in the rulebook and other requirements into machine-readable policies
Rationale	The need and implementation for the policies is identified and defined during the definition of the common shared principles. These principles should be connected to the actual implementation of policies and the policy engine
Priority	SHOULD
Test Case / Acceptance Criteria	Commonly agreed usage control principles have been codified into machine readable policies to be executed by the policy engine

ID	R054
Type	Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	Collect machine-readable policies into a policy library, which collects together and lists machine-readable policies and their description in a human-understandable form
Rationale	Policies act as a bridge between the governance principles and the technical data infrastructure. Policies used as part of the usage control should be collected together and available for execution by the data sovereignty components, such as the policy engine. This collection should include both the actual machine-readable policy as well as the human-readable counterpart so that it is easier to understand what the policies mean and how the description is implemented in the policy language
Priority	SHOULD
Test Case / Acceptance Criteria	Machine-readable policies and their human-readable counterparts have been collected together and available for use by the policy engine

ID	R055
Type	Functional
Source	External Stakeholders

Category	Data Sharing
Description	The framework must provide different usage policies
Rationale	Share data under different usage policies, e.g. share a sample or historical data for research purposes under usage policy X and recent (live) data for commercial services under usage policy Y
Priority	MUST
Test Case / Acceptance Criteria	Verify that the Data Space supports different usage policies and the framework is able to enforce them

ID	R056
Type	Functional
Source	Pilot Analysis
Category	Data Sharing
Description	As a data provider, I would like to be able to publish part of the data as open data because of grant/funding requirements, legislation requirements or foster the reuse of the produced data
Rationale	Industrial availability. Open data is one of the recommendations and sometimes requirements of the EU policies and programs, and fosters the production of science
Priority	MUST
Test Case / Acceptance Criteria	The user has an option to choose to share his data publicly

ID	R057
Type	Functional
Source	Pilot Analysis
Category	Data Sharing
Description	As a data hub provider/domain provider (e.g. EDIH and TEF), I would like to be able to setup the dataspace for specific purpose (to share the data with specific business goals)
Rationale	Dataspace have varying requirements stemming e.g. from agreed business logic between dataspace participants. These rules should be reflected as part of the related policies provided that the underlying data infrastructure is able to execute such policies

Priority	SHOULD
Test Case / Acceptance Criteria	Dataset definition should include a set of policies relevant to that specific dataset.

ID	R058
Type	Functional
Source	External Stakeholders
Category	Data Sharing
Description	The framework could be able to share data within specific data periods and ensure that they will not be available outside that timeframe
Rationale	This feature ensures that the consumers cannot download the dataset after a specific time period
Priority	COULD
Test Case / Acceptance Criteria	<ol style="list-style-type: none"> 1. Verify that a Data Consumer cannot access a previously accessible dataset after a specified date. 2. Verify that a Data Consumer cannot access a shared dataset before the specified date when access is starting

ID	R059
Type	Non-Functional
Source	Pilot Analysis
Category	Security
Description	As a pilot provider, I would like a facility to handle heterogenous data with varying access policies
Rationale	Different datasets contain different types of access control requirements. These requirements must be converted into a set of policies that are associated with the dataset and executed when accessing the dataset
Priority	MUST
Test Case / Acceptance Criteria	Datasets must have an association with an applicable set of policies

ID	R062
Type	Functional

Source	External Stakeholders
Category	Data Governance
Description	As a data consumer, I want to be able to search/filter data and choose what to download
Rationale	Facilitate the user to select the appropriate subset of a dataset and download (even buy) only the desired, by him, subset of the original data
Priority	WON'T
Test Case / Acceptance Criteria	N/A

ID	R063
Type	Functional
Source	External Stakeholders
Category	Data Governance
Description	The framework has to estimate the size of a filtered dataset
Rationale	This feature calculates the size of a dataset (or data product) based on the selected (filtered) columns and rows. It will be used to display to the consumer the total size of what they are going to download
Priority	WON'T
Test Case / Acceptance Criteria	N/A

ID	R064
Type	Functional
Source	Pilot Analysis
Category	Data Governance
Description	The framework could track data changes through metadata
Rationale	Provide metadata that will track changes on different versions of the same dataset (e.g., new extraction date, new data size, added data fields)
Priority	COULD

Test Case / Acceptance Criteria	Upload different versions of the same dataset and verify that the changes can be seen on the metadata
---------------------------------	---

ID	R067
Type	Functional
Source	Pilot Analysis
Category	Data Sharing
Description	The connectors from the data sharing module that will be implemented in the pilot 4 must be able to provide information from the company database to the (energy) data space in order to explore new ways of providing this information
Rationale	Create technical conditions to share data with the demonstration endpoints
Priority	MUST
Test Case / Acceptance Criteria	After the connectors are implemented, the access to the data from the data spaces must be verified

ID	R068
Type	Functional
Source	Pilot Analysis
Category	Data Sharing
Description	The connectors from the data sharing module should be able to share the pilot 4 datasets and metadata to the (energy) data space in order to explore new ways of providing this information
Rationale	Create technical conditions to share data with the demonstration endpoints
Priority	SHOULD
Test Case / Acceptance Criteria	After the connectors are implemented, the access to the data from the data spaces should be verified

ID	R069
Type	Non-Functional
Source	Pilot Analysis
Category	Data Sharing

Description	The connectors from the data sharing module should be configured to the data space endpoints that will be used in pilot 4 in order to be able to connect and provide datasets to at least one (energy) data space
Rationale	Create technical conditions to share data with the demonstration endpoints
Priority	SHOULD
Test Case / Acceptance Criteria	After the connectors are implemented, the access to the data from the data spaces should be verified

ID	R070
Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	Data governance and data sharing, in general, should be based on a set of common and shared principles, e.g. a rulebook
Rationale	Common and shared ground rules lay the foundation for data-based collaboration and governance. These will be further refined into a set of machine-readable policies to be executed by the data sovereignty components, especially the policy engine
Priority	SHOULD
Test Case / Acceptance Criteria	Usage control-related principles have been defined, clarified, and available for converting them into machine readable policies to be executed by the policy engine

ID	R072
Type	Functional
Source	Pilot Analysis
Category	Security
Description	The framework must allow users to define and manage access control settings for their datasets, including sharing with specific individuals or groups
Rationale	To enable controlled sharing of data and ensure data security and privacy
Priority	MUST

Test Case / Acceptance Criteria	Users can define access policies to their datasets
---------------------------------	--

ID	R073
Type	Functional
Source	Pilot Analysis
Category	Data Sharing
Description	As the provider of domain service in agriculture, I would like to setup, configure and customise the catalogue of the DATAMITE framework based on DATAMITE components
Rationale	To facilitate the exploitation and monetisation of the datasets
Priority	MUST
Test Case / Acceptance Criteria	DATAMITE framework customised, deployed at PSNC and operational

ID	R074
Type	Functional
Source	Pilot Analysis
Category	Data Sharing
Description	As the provider of domain service in agriculture, I would like to integrate the catalogue of the DATAMITE framework with the accounting system (for the transactions)
Rationale	To facilitate monetisation of the datasets supporting charging, invoicing, and reporting
Priority	SHOULD
Test Case / Acceptance Criteria	In the pilot 5 end user interface, there is a user profile with all the transactions, which will be based on the successful transaction list for the Logging Module of DATAMITE (via API)

ID	R075
Type	Functional
Source	Pilot Analysis
Category	Data Sharing

Description	As the provider of domain service in agriculture, I would like to integrate the catalogue of the DATAMITE framework with the external billing system (like PayPal)
Rationale	To facilitate monetisation of the datasets supporting the most common payment methods
Priority	SHOULD
Test Case / Acceptance Criteria	There is a pilot 5 specific module implementing the payment functionality (to pay for selected datasets)

ID	R076
Type	Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	The framework will not be able to create data spaces from scratch
Rationale	The goal of the project is not to create data spaces, but to facilitate the access to them
Priority	WON'T
Test Case / Acceptance Criteria	N/A

ID	R079
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The framework requires a data catalogue that can be used by users to find data assets in their system
Rationale	A highly usable data catalogue is required to ease the findability and accessibility of data
Priority	MUST
Test Case / Acceptance Criteria	There exists a catalogue with different criteria for sorting the datasets as well as different means to perform a search for datasets

ID	R080
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The framework requires a data catalogue that can be used to edit or add information to data assets in the system, e.g. linking data entities/fields with terms in the data glossary
Rationale	A highly usable data catalogue is required to allow describing datasets and link them to terms in glossaries
Priority	MUST
Test Case / Acceptance Criteria	Datasets include a description that can be edited as well as other means of enriching it. This includes the capability of linking the dataset or its components (e.g., artifacts, fields) to terms from different vocabularies in the glossary

ID	R082
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The framework requires that the data catalogue can be linked to the data quality functionalities to run the corresponding routines and evaluate the data asset accordingly
Rationale	Data quality indicators are another relevant source of information of the datasets. Given this, it is important that the data quality features can be used or linked from the data catalogue
Priority	MUST
Test Case / Acceptance Criteria	It is possible to define quality indicators or to assign existing ones to fields of a dataset from the catalogue. It is also possible to run quality evaluations

ID	R083
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance

Description	The framework requires a data dictionary for each dataset that displays a set of basic indicators, which have been agreed beforehand, and that offer basic information about the dataset
Rationale	The data dictionary provides an initial overview from the data, potentially the results or part of the results from the data profile. This information is mostly technical
Priority	MUST
Test Case / Acceptance Criteria	The detail of a dataset in the data catalogue includes a tab or similar with data quality information resulting of the execution of a profile

ID	R084
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The data glossary must be able to import existing vocabularies. To import them, they must be in a format that is friendly to the glossary
Rationale	The reason is that particular data spaces or environments may impose existing vocabularies in order to publish data there. Similarly, there are many existing vocabularies that may be adopted by companies, and their use has to be facilitated
Priority	MUST
Test Case / Acceptance Criteria	It is possible to import existing vocabularies from the data glossary

ID	R085
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The data glossary must be able to group the vocabularies per domain. A vocabulary may be related to more than one domain
Rationale	Organising vocabularies per domain facilitates the task of finding them and the terms within
Priority	MUST

Test Case / Acceptance Criteria	In the data glossary, it is possible to filter or sort the existing vocabularies by domain
---------------------------------	--

ID	R087
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The framework must consider that data assets may be composed of data artifacts
Rationale	Datasets are not always atomic. A dataset coming in streaming is compacted every X time or every X size. Some datasets contain different files that can be related, etc. This has to be handled by the framework
Priority	MUST
Test Case / Acceptance Criteria	There are datasets in the data catalogue composed by more than one artifact

ID	R088
Type	Functional
Source	External Stakeholders
Category	Data Governance
Description	The data catalogue and the data dictionary must facilitate the addition of metadata and information, in general, at a fine or detailed level enough to achieve a sufficient level of interpretation of the datasets
Rationale	The catalogue must be the entry point for the metadata provided by users. It must offer an easy/usable interface thought for non-technical users
Priority	MUST
Test Case / Acceptance Criteria	It is possible to add/edit dataset/field descriptions or similar sources of information in a dataset in the catalogue. Similarly, it is possible to link those with terms from data vocabularies

ID	R089
Type	Functional
Source	Internal Technical Analysis

Category	Data Governance
Description	The framework should provide basic data lineage functionalities to record and make accessible the origin of the data and the processes through which they pass when it applies
Rationale	This could be interesting, especially within an organisation organisation, to reflect the changes that a dataset has suffered from its ingestion. It is basically a traceability tool
Priority	SHOULD
Test Case / Acceptance Criteria	It is possible to visualise information regarding the lineage of a dataset / data product when the associated transformations have happened within the framework

ID	R090
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The end user must have the capability to access a centralised data catalogue, where he/she will be able to identify who is the data owner of each dataset, what are the underline data, their definition, and their data types, and last but not least to capture data lineage. The framework must offer a full catalogue of business and technical metadata
Rationale	It will allow non-technical and technical users to access metadata of in scope information in a single point
Priority	MUST
Test Case / Acceptance Criteria	The data catalogue must include different kinds of metadata for describing a dataset, both technical and business, including terms linked from vocabularies in the glossary

ID	R091
Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	The framework must clarify who has control over the data and how ownership rights can be preserved to create a fair and transparent data market

Rationale	The question of data sovereignty and ownership is complex, especially when it comes to monetising data. The framework must include data ownership in the metadata schema for usability and safe handling purposes
Priority	MUST
Test Case / Acceptance Criteria	Implemented the functional mechanisms (e.g. user interfaces, APIs) that allow data owners to set and manage access controls, permissions and preferences for data sharing. Review of the clearly formulated Rulebook

ID	R093
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The framework should provide the capability to update technical or business metadata, by integrating and capturing the latest view from the metadata repository
Rationale	It will allow the framework to have the most refreshed and updated metadata information, reducing the probability of becoming a data swamp
Priority	SHOULD
Test Case / Acceptance Criteria	<p>Login to the DATAMITE portal as a data owner.</p> <p>Select the option that provides access to metadata management and updating.</p> <p>Choose a specific dataset and select the option "Update Metadata".</p> <p>Update metadata and then save the changes.</p> <p>Login to the DATAMITE portal as a data consumer and confirm the updated view of the metadata</p>

ID	R094
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The framework requires the implementation of an efficient metadata repository to store and index rich metadata to enable rapid search and retrieval of asset information

Rationale	In order to enrich datasets with metadata, this metadata has to be stored somewhere, as it links to the particular fields/columns/tables/X that it is enriching.
Priority	MUST
Test Case / Acceptance Criteria	An "overall performance index" is defined which is based on a combination of storage and indexing efficiency, search speed, relevance and completeness of search results, scalability, user-friendliness and system maintainability

ID	R095
Type	Functional
Source	External Stakeholders
Category	Data Governance
Description	The framework must provide a search option for metadata
Rationale	Allows consumers to find datasets of interest by searching specific keywords and matching them with the metadata of related datasets
Priority	MUST
Test Case / Acceptance Criteria	The data consumer searches for specific keywords in data catalogues (e.g. Energy, Power, Voltage, distribution grid) and the framework displays the datasets containing those keywords

ID	R096
Type	Functional
Source	Pilot Analysis
Category	Data Governance
Description	The framework should be able to map the connection between related datasets
Rationale	This is an important feature which will be used to link (bundle) different datasets into one data product (e.g. scada and telemetering files)
Priority	SHOULD
Test Case / Acceptance Criteria	1. Verify that a Data Provider can add to the metadata links to related datasets. 2. Verify that a Data Consumer can follow these links

ID	R098
----	------

Type	Functional
Source	Internal Technical Analysis
Category	Quality
Description	The framework (data quality module) must evaluate the quality of a dataset based on standard quality indicators
Rationale	This tool is a part of the data quality module and must be able to check and provide an overview of the quality of the datasets
Priority	MUST
Test Case / Acceptance Criteria	Verify that the dataset to be shared has successfully met the requirements defined by the KPIs

ID	R099
Type	Functional
Source	Internal Technical Analysis
Category	Quality
Description	A set of data quality indicators must be defined so that they are used by the data profiler and data dictionary. These indicators can be statistical indicators (e.g., mean, histograms), describe the type of dataset or others
Rationale	The quality tools must differentiate between generic indicators, those that do not need business context to be computed, and business based ones. Generic indicators are key to provide a solid basis for the data quality module as well as for governance tools such as the data dictionary
Priority	MUST
Test Case / Acceptance Criteria	There exists a KPI library with a wide set of implemented informative/generic quality indicators

ID	R100
Type	Functional
Source	Internal Technical Analysis
Category	Quality
Description	The framework should provide data quality monitoring of underlying data (sources and relevant pipelines). It should offer the capability to execute

	business defined data quality rules in order to monitor key data elements (KDE)
Rationale	Similarly, to the generic indicators, the framework should allow the definition of business/context based indicators by users. Probably through a GUI
Priority	SHOULD
Test Case / Acceptance Criteria	It is possible to define quality indicators through a user interface and store them within the KPI library

ID	R101
Type	Functional
Source	Internal Technical Analysis
Category	Quality
Description	The framework must implement basic and advanced quality mechanisms, including additional KPIs, such as the detection of data biases, advanced statistics, etc.
Rationale	Improve data quality by identifying and addressing biases, ensuring the reliability and fairness of data-driven processes
Priority	MUST
Test Case / Acceptance Criteria	There exists a tool to evaluate/detect biases in datasets, and it can be applied to one or more datasets in any of the pilots

ID	R102
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	There must be a description language (DSL) powerful and expressive enough to describe quality metadata - DSLQ
Rationale	Quality information is part of the definition of data products
Priority	MUST
Test Case / Acceptance Criteria	It is possible to model and visualise information regarding the quality of the data set/data product

ID	R103
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The searches allowed in the catalogue should consider parameters referring to dimensions or quality indicators
Rationale	Quality information is an important aspect to the user for searching in the catalogue
Priority	SHOULD
Test Case / Acceptance Criteria	It is possible to condition a search or filter it according to quality indicators

ID	R105
Type	Functional
Source	Internal Technical Analysis
Category	Quality
Description	The framework should have efficient provisions for checking data quality (e.g., to detect concept drift, missing data, inconsistent data, anomalies, outliers, completeness, validity, timeliness, consistency, etc.)
Rationale	80% of development effort in previous projects went into data integration and quality
Priority	SHOULD
Test Case / Acceptance Criteria	There should be a collection of indicators in different dimensions, allowing to evaluate the quality of a dataset from different perspectives

ID	R106
Type	Non-Functional
Source	Internal Technical Analysis
Category	Quality
Description	The framework should ensure that data is regularly updated and monitored to ensure it is current

Rationale	Keeping data current is critical to ensure its relevance and value. Outdated data can lead to incorrect conclusions or decisions. One potential quality indicator should be data freshness
Priority	SHOULD
Test Case / Acceptance Criteria	Review of the mechanism that continuously monitors the timeliness of the data. The effectiveness of this mechanism is measured by its ability to identify outdated or inconsistent data and trigger corresponding notifications or update processes.

ID	R108
Type	Functional
Source	External Stakeholders
Category	Quality
Description	The data supporting tools module should include tools that provide information about the fairness/bias of a dataset following indications of EU regulations
Rationale	Tools enhancing the quality of data will help the assimilation of the framework from a variety of users, in particular, those with no access to such tools
Priority	SHOULD
Test Case / Acceptance Criteria	The framework should have a data fairness evaluation tool that provides bias/unbiased metrics to assess the level of fairness of a dataset. This tool should be aligned with EU policy regarding fairness treatment

ID	R109
Type	Functional
Source	External Stakeholders
Category	Data Sharing
Description	Data sharing: The usage policy should be available on data publishing to EU portals
Rationale	The usage policy should be part of the metadata in order to publish to the EU data platforms
Priority	SHOULD
Test Case / Acceptance Criteria	If an EU data portal requires the usage policy of a dataset in order to be published, the framework should be able to provide the usage policy of a dataset when publishing that dataset to that EU data portal

ID	R110
Type	Functional
Source	Pilot Analysis
Category	Quality
Description	As a user of the data, I would like to know if the data is validated/curated and what is the date of validation
Rationale	Support automated data validation mechanisms, to ensure that datasets published on the platform adhere to quality standards, enhancing the trustworthiness of the data
Priority	SHOULD
Test Case / Acceptance Criteria	Metadata contains field(s) with information about the date of validation

ID	R111
Type	Functional
Source	Internal Technical Analysis
Category	Quality
Description	The framework should provide data quality reports allowing business and technical users to monitor data quality progress and keep them informed concerning the status of data that are ready to be consumed
Rationale	It will allow users to be sure about data freshness and avoid data synchronisation conflicts
Priority	SHOULD
Test Case / Acceptance Criteria	It should be possible to export the data quality information from a dataset into a report, either through the graphical interface or downloadable

ID	R112
Type	Functional
Source	External Stakeholders
Category	Quality
Description	Interfaces to compare the quality of different datasets, considering a future provision to third parties, through exchange or sale

Rationale	To be used in the data valuation process
Priority	MUST
Test Case / Acceptance Criteria	There is a collection of quality indicators that allow for comparing two different datasets in a generic way (e.g., completeness, uniqueness...)

ID	R113
Type	Functional
Source	Pilot Analysis
Category	Data Sharing
Description	A transformation pipeline system should be implemented. This tool should be totally adaptable in order to accomplish publication requirements
Rationale	Quality and flexibility requirements
Priority	SHOULD
Test Case / Acceptance Criteria	It should be possible to create a data product as a subset or an aggregation of datasets in the catalogue

ID	R116
Type	Functional
Source	Internal Technical Analysis
Category	Security
Description	Procedures must be provided to facilitate the definition of roles in data management and guarantee compliance with data accessibility policies designed
Rationale	Authentication and authorisation must be considered. Role based authorisation is a common approach that can be studied
Priority	MUST
Test Case / Acceptance Criteria	There are several roles (e.g., data owner, steward, admin) defined and they can be assigned to different users.

ID	R117
Type	Functional

Source	Internal Technical Analysis
Category	Security
Description	A tool to define fine-grained data access policies about who, what, how and when has access to data (clearly defined security policies and role-based permissions)
Rationale	Besides general authentication and authorisation mechanisms of the framework, we must consider the access to data. It is important to offer tools that allow a fine-grained control over who accesses what
Priority	MUST
Test Case / Acceptance Criteria	The tool must allow administrators to define and enforce fine-grained data access policies and role-based access control (RBAC)

ID	R118
Type	Functional
Source	Internal Technical Analysis
Category	Security
Description	The framework should have the capability to control/monitor internal users' access across all data sources, and enable reviewing which type of data has been accessed by whom, tracking users' actions and alerting possible privacy issues
Rationale	It will allow the admins to keep track of user actions and avoid internal information leakage
Priority	SHOULD
Test Case / Acceptance Criteria	Login to the DATAMITE portal as a data owner. Choose a dataset and a specific time period. Confirm that the framework provides a report regarding which users accessed the dataset during the selected time period, along with other data usage metrics (e.g., number of accesses, size of data, etc.)

ID	R119
Type	Functional
Source	Pilot Analysis
Category	Security
Description	Dataset owners must have control in two different points related to the dataset publishing: firstly, owners must have access control (who can

	access the dataset), and secondly, owners must have visualisation control (who can view metadata in marketplace tool)
Rationale	To tackle this requirement, the framework must allow flexibility to define data access and visualisation policies based on roles
Priority	MUST
Test Case / Acceptance Criteria	It is possible to define different levels of visibility for a dataset (e.g., public / private), or policies regarding data usage or access can be defined

ID	R120
Type	Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	The framework should ensure and enforce data sovereignty on the IDSA connectors
Rationale	Ensure that data owners/providers have control over their data and its usage in accordance with specified restrictions
Priority	SHOULD
Test Case / Acceptance Criteria	The connector will include a policy engine to check the data usage policies

ID	R121
Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	As a data provider, I want to be able to share data inside a sandbox owned by me (access control framework)
Rationale	Data availability control
Priority	WON'T
Test Case / Acceptance Criteria	N/A

ID	R122
----	------

Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	The data sharing module must include an efficient tool that uses the appropriate storage technologies (e.g., databases, blockchain) to support the secure storage of usage contracts
Rationale	Provide a secure tool that prevents unauthorised actions, describing permissions and obligations of participants in data exchange
Priority	MUST
Test Case / Acceptance Criteria	The tool efficiently processes and stores the contracts securely by preventing unauthorised access

ID	R123
Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	The data sharing module must describe the permissions and obligations of participants involved in data exchange through contracts
Rationale	Ensure clarity and transparency of data usage rights and responsibilities by explicitly defining them in contracts
Priority	MUST
Test Case / Acceptance Criteria	The contracts cover all relevant aspects of data usage rights and obligations, leaving no ambiguity or gaps in information and are easily accessible to all relevant parties, ensuring that participants can review the terms at any point

ID	R124
Type	Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	The data sharing module should perpetually monitor the evaluation of usage enforcement
Rationale	Enable continuous monitoring of data usage to ensure compliance with defined rules and policies

Priority	SHOULD
Test Case / Acceptance Criteria	Logs attempts by authorised/unauthorized users to access the data

ID	R125
Type	Functional
Source	Pilot Analysis
Category	Security
Description	As a System Manager, I want the data components that interact with raw data to be deployed in EDP internal network, so that I comply with applicable corporate security policies
Rationale	Internal security policies
Priority	MUST
Test Case / Acceptance Criteria	After the connectors are implemented, the access to the data from the data spaces must be verified

ID	R126
Type	Non-Functional
Source	Pilot Analysis
Category	Data Sharing
Description	The connectors from the data sharing modules to be implemented in pilot 4 should not require inbound firewall configuration (e.g. expose VM port to the internet)
Rationale	Ensure the viability of the connector's correct deployment
Priority	SHOULD
Test Case / Acceptance Criteria	After the connectors are implemented, the access to the data from the data spaces should be verified

ID	R130
Type	Functional
Source	External Stakeholders

Category	Security
Description	The framework (data support tools) could check if the anonymisation of a dataset was successful
Rationale	This will be used after the anonymisation of the dataset and will provide an overview of the quality of the anonymisation
Priority	COULD
Test Case / Acceptance Criteria	Verify that the anonymised datasets are indeed anonymised based on several tests (e.g., k-anonymity, t-closeness, l-diversity)

ID	R133
Type	Non-Functional
Source	Pilot Analysis
Category	Security
Description	As a data hub provider/domain provider, I would like to make sure the data are shared in a secure way (access control, authorisation, secure protocol for data accessing)
Rationale	Improving the framework and data trustworthiness
Priority	MUST
Test Case / Acceptance Criteria	Verify that authentication is needed to access the functionalities provided and that they are using secure mechanism for data sharing

ID	R138
Type	Functional
Source	Internal Technical Analysis
Category	Security
Description	The framework should be integrated with an IAM solution such as Keycloak or similar
Rationale	To enhance authentication and authorisation of users
Priority	SHOULD
Test Case / Acceptance Criteria	<ol style="list-style-type: none"> 1. Check if is available a login button/page to access the framework. 2. Check if the login redirects to no other domain/subdomain

ID	R139
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	A storage technology is required to receive the data captured with the data ingestion mechanisms. This may include a landing layer
Rationale	We may not want data to go directly to a storage layer, but it is worth considering an intermediate layer that allows compaction (in the case of streaming data), analysis of it or others
Priority	MUST
Test Case / Acceptance Criteria	The framework offers the possibility of storing data locally, without using external data storage technologies. In the case of streaming data, it may be stored as a temporary before being compacted and persisted permanently

ID	R141
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	The framework must provide a homogeneous/standard process to ingest data in bulk or streaming
Rationale	The framework must facilitate a mechanism for ingesting data in bulk or streaming in case the organisation does not have one
Priority	MUST
Test Case / Acceptance Criteria	It must be possible to create new datasets in the framework by uploading them or establishing streaming connections

ID	R143
Type	Functional
Source	Pilot Analysis
Category	Functionality
Description	A connector to the company data lake must be developed in the data support tools module in order to ingest and process the datasets for the pilot 4

Rationale	Ensure that the datasets can be retrieved from the internal systems to the pilot 4 sandbox in order to deploy the demonstration
Priority	MUST
Test Case / Acceptance Criteria	Data can be retrieved from E-REDES internal systems to the pilot 4 sandbox

ID	R144
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	The framework must be able to discover existing datasets stored in different storage technologies
Rationale	Organisations may have storage technologies already in place, so it is required to have data discovery mechanisms that allow to find and load these datasets into the data catalogue
Priority	MUST
Test Case / Acceptance Criteria	It is possible to connect to different storage technologies and import/discover datasets stored in there

ID	R146
Type	Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	The framework could support accounting/telemetry functionalities
Rationale	Accounting or data usage control can be very relevant for external processes such as billing (that we may not need to consider) as well as other related to data sovereignty. It may be a support tool or a part of the data sharing module
Priority	COULD
Test Case / Acceptance Criteria	There is a logging component that lists/logs events or interactions from external users (e.g., accesses or downloads) with respect to the dataset

ID	R147
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The data and its metadata must be persisted appropriately to its type and source/origin, in addition to being annotated with other considerations of sovereignty, privacy, openness, quality, provenance, etc. and also have a set of functionalities that allow discovering and consult this information
Rationale	The technology used to implement the catalogue must support an appropriate and extensible metadata model
Priority	MUST
Test Case / Acceptance Criteria	The metadata schema must be able to include all the information that is deemed as relevant by the pilots

ID	R149
Type	Functional
Source	Pilot Analysis
Category	Data Governance
Description	As a pilot provider, I would like to have data searching and filtering capability based on predefined criteria (like statistics about the data quality for each dataset)
Rationale	To enhance user experience by helping them find the data they need efficiently and improving overall framework usability
Priority	MUST
Test Case / Acceptance Criteria	Allow further advanced search criteria such as spatial and temporal coverage of datasets, but also filtering based on data quality and completeness

ID	R150
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	The framework frontend should be for mobile use. The mobile may offer fewer options, but it should be developed

Rationale	Portability
Priority	SHOULD
Test Case / Acceptance Criteria	1. Check if the framework is available to be tested on mobile devices. 2. Resolutions below 1280px width should offer a mobile version without breaking

ID	R152
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	We must have multiple user tiers, with multiple roles associated. Users must be able to create an account (at least for the lower tier). Others would be managed by admins
Rationale	Usability
Priority	MUST
Test Case / Acceptance Criteria	1. Framework must offer a form to create an account without login. 2. Admins must have an option in backoffice to create users and set their role

ID	R153
Type	Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	All data models should be created as independent components to be easily integrated in the framework
Rationale	Integration
Priority	SHOULD
Test Case / Acceptance Criteria	The admin should have an option at deployment and later to define which modules he/she wants to install/ add

ID	R154
Type	Functional

Source	Internal Technical Analysis
Category	Functionality
Description	The framework must provide a public privacy and cookie policy
Rationale	To improve users' experience with respect to GDPR
Priority	MUST
Test Case / Acceptance Criteria	Users without login must be able to see the private policy and cookie policy page

ID	R155
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	We should be able to monitor the framework health in one place (individual services status, APIs availability, other)
Rationale	To enable business continuity
Priority	SHOULD
Test Case / Acceptance Criteria	In the backoffice, there should be an area/page that provides an overview of the system status

ID	R156
Type	Functional
Source	Internal Technical Analysis
Category	Compatibility
Description	A storage technology is required to store the data assets stored in the system. The storage technology may or may not be the same used to receive the data from the ingestion mechanisms
Rationale	Besides the potential landing layer, the framework must offer a storage technology, generic enough, that it can be complemented with other storage technologies in the user organisation
Priority	MUST

Test Case / Acceptance Criteria	The framework offers the possibility of storing data locally without using external data storage technologies, especially for streaming data storage. In the case of streaming data, it may be stored as a temporary before being compacted and persisted permanently
---------------------------------	---

ID	R157
Type	Non-Functional
Source	Internal Technical Analysis
Category	Compatibility
Description	The framework data ingestion mechanisms in streaming should accept multiple formats/protocols (e.g. Kafka, MQTT, or others)
Rationale	The framework should not force an organisation to work with only one streaming data protocol. It should be as flexible as possible
Priority	SHOULD
Test Case / Acceptance Criteria	It is possible to ingest data through streaming and the connections can be created using a number of different communication protocols

ID	R158
Type	Functional
Source	Internal Technical Analysis
Category	Compatibility
Description	The framework must facilitate data connectors with core traditional databases (e.g., Oracle, Microsoft, SAP), ensuring seamless connectivity with the majority of operational systems currently in use. These connectors are to be designed to support integration with both on-premises and cloud infrastructures, equipped with functionalities to update or select tables and columns from the datastore
Rationale	It is critical to integrate core systems from different operational systems and offer a centralised interface with these sources
Priority	MUST
Test Case / Acceptance Criteria	It must be possible to connect and extract data/metadata from different existing storage technologies and work with them from DATAMITE

ID	R159
----	------

Type	Functional
Source	Internal Technical Analysis
Category	Compatibility
Description	The framework should enable data connectors with non-relational databases, big data infrastructures, and data storages (e.g., Cloudera, S3 AWS, ADLS Azure, HBase), promoting connectivity with both current and anticipated big data infrastructure organisations. These connectors are expected to support integration across on-premises and cloud infrastructures, with the solution's design aiming for generality and ease of integrating additional databases
Rationale	It will enable organisations to integrate with big data and data storage, enabling users to have access to detailed data
Priority	SHOULD
Test Case / Acceptance Criteria	It should be possible to connect and extract data/metadata from different BigData-like existing storage technologies and work with them from DATAMITE, e.g. S3 like repos

ID	R160
Type	Functional
Source	Pilot Analysis
Category	Compatibility
Description	All of the developed components from the different modules must be compatible with the Azure cloud (E-REDES infrastructure system) so that they can be integrated to deploy the demonstration
Rationale	Ensure the viability of the components' deployment
Priority	MUST
Test Case / Acceptance Criteria	The E-REDES infrastructure system is deployed for the pilot and the components are integrated

ID	R162
Type	Functional
Source	Internal Technical Analysis
Category	Compatibility

Description	Connection with different persistence technologies must be present (e.g. PostgreSQL, MariaDB, MongoDB, Cassandra, Elasticsearch, etc.)
Rationale	Allowing for discovering data from different sources
Priority	MUST
Test Case / Acceptance Criteria	The user will be able to setup new connections to different storage technologies to capture datasets stored in them and import their metadata in the catalogue

ID	R163
Type	Functional
Source	Internal Technical Analysis
Category	Compatibility
Description	The framework must be able to store datasets, at least, as objects (e.g., S3-like storage)
Rationale	The framework must provide some storage that is easily integrated with it in case the organisation does not have one
Priority	MUST
Test Case / Acceptance Criteria	Data uploaded as bulk or received through streaming connections are stored in an internal MinIO instance

ID	R165
Type	Non-Functional
Source	Pilot Analysis
Category	Compatibility
Description	The framework frontend must be compatible with modern web browsers
Rationale	To provide a consistent user experience
Priority	MUST
Test Case / Acceptance Criteria	The user must be able to use all framework functionalities without losing any experience in browsers such as: Chrome, Firefox, Safari (latest 2 versions)

ID	R166
Type	Functional

Source	Internal Technical Analysis
Category	Maintainability
Description	Set up Continuous Integration (CI) tools and processes
Rationale	Facilitate seamless integration and timely releases by automating the build, testing, and deployment processes
Priority	MUST
Test Case / Acceptance Criteria	The continuous integration will be implemented with the most suitable CI platform from (GitLab, CircleCI, Jenkins) and docker-compose

ID	R167
Type	Functional
Source	Internal Technical Analysis
Category	Maintainability
Description	Establish CI workflows to guide the integration
Rationale	Ensure consistency and reliability in integrating different software and tools
Priority	SHOULD
Test Case / Acceptance Criteria	Will setup the appropriate pipelines for CD (Continuous Development) with the expected final result to be with each git push on a predefined branch, the new version of the app will be automatically built and deployed on the (production/development) server (according to the branch)

ID	R168
Type	Functional
Source	Internal Technical Analysis
Category	Maintainability
Description	Implement version control to manage the source code and track changes
Rationale	Enable collaboration, version tracking, and easy rollback of changes, ensuring better code management and traceability
Priority	SHOULD
Test Case / Acceptance Criteria	Source code control: Will host code on GitLab to integrate the app with major software and services

ID	R169
Type	Non-Functional
Source	Internal Technical Analysis
Category	Maintainability
Description	Provide comprehensive documentation and guidelines for the CI tools and processes
Rationale	Enable the development team to effectively utilise the CI infrastructure, ensuring consistency and reducing onboarding and maintenance time
Priority	MUST
Test Case / Acceptance Criteria	Must provide all the required config files (docker, docker-compose, gitlab-ci.yml) and documentation so the development team will utilise the CI/CD infrastructure in a specific and easy way

ID	R170
Type	Non-Functional
Source	Internal Technical Analysis
Category	Maintainability
Description	It should be able to upgrade the whole framework (or part of it), enabling the correction of possible bugs and/or upgrading its current capabilities
Rationale	It will enable continuously evolving by offering a robust, maintainable and upgradable framework
Priority	SHOULD
Test Case / Acceptance Criteria	Upon new versions, components will allow its upgrade/deployment, minimising the impact on the user

ID	R172
Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The catalogue that is defined in the DATAMITE framework, should annotate/publish its services adequately, so that they can be compiled in GAIA-X/IDS data spaces
Rationale	GAIA-X/IDS compliance data services description

Priority	SHOULD
Test Case / Acceptance Criteria	The internal data product has the information needed by the external data sharing initiative

ID	R173
Type	Non-Functional
Source	Internal Technical Analysis
Category	Compliance
Description	The framework design could be aligned with low carbon guidelines for web development.
Rationale	Sustainability
Priority	COULD
Test Case / Acceptance Criteria	The framework could provide a high score here https://ecograder.com/

ID	R174
Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	It could be possible to define the conditions in which the data will be provided (establish some minimums/requirements in terms of quality, periodicity, precision, etc.)
Rationale	Agree a data service level agreement between stakeholders
Priority	COULD
Test Case / Acceptance Criteria	The conditions in which the data will be provided will be included in the external data product model

ID	R176
Type	Non-Functional
Source	Pilot Analysis
Category	Portability

Description	The framework should be able to operate both on prem and on cloud infrastructure, allowing organisation to select the preferred type based on the overall IT strategy
Rationale	It will enable the framework to be independent of the infrastructure type of the host
Priority	SHOULD
Test Case / Acceptance Criteria	Deploy the DATAMITE framework on-premises and on cloud infrastructure as well. Confirm that, in both cases, the framework is functioning properly

ID	R179
Type	Non-Functional
Source	Internal Technical Analysis
Category	Performance
Description	The framework frontend should be considered high performing under the PageSpeed Insights tool
Rationale	If the framework does not have a good performance, users will lack interest, and we can lose users over time
Priority	SHOULD
Test Case / Acceptance Criteria	The framework should provide a high score here https://pagespeed.web.dev/

ID	R180
Type	Non-Functional
Source	Internal Technical Analysis
Category	Performance
Description	Clear criteria must be defined to evaluate the quality of the DATAMITE framework
Rationale	Evaluation of the results of DATAMITE based on the requirements list. Maybe additionally oriented on the ISO 25010 (e.g., processing speed, scalability, resource utilisation, data quality, user experience and other relevant metrics)
Priority	MUST

Test Case / Acceptance Criteria	<p>Review of the framework with the previously defined and analysed requirements. And the requirements were met (for the most part).</p> <p>And perhaps an additional review of the overall quality index, which is based on critical quality dimensions: Functionality, Reliability, Usability, Efficiency, Maintainability, Portability and Security</p>
---------------------------------	--

ID	R181
Type	Non-Functional
Source	Internal Technical Analysis
Category	Usability
Description	To ensure easy and reusable implementation of some standard operations like anonymisation, the framework should provide libraries. It is important that these mechanisms are well documented and designed in a user-friendly way to allow developers a seamless integration process
Rationale	Maintainability and support
Priority	SHOULD
Test Case / Acceptance Criteria	Mechanisms for standard operations have been implemented

ID	R182
Type	Non-Functional
Source	Internal Technical Analysis
Category	Data Governance
Description	The framework should provide a unified user interface for business users to access core operations, including data and metadata. An elevated view for administrators is essential for full functionality access, with a key system feature being the transparency of policies and access rules. A dedicated form is advised to clearly present these restrictions to administrators. The complexity of integrating more databases necessitates a clear and user-friendly UI for efficient policy orchestration and monitoring
Rationale	It will provide ease of access to core functionalities to business users, enabling them to be part of BAU operations and offer advanced UI for admins
Priority	SHOULD

Test Case / Acceptance Criteria	DATAMITE frontend should be highly usable
---------------------------------	---

ID	R183
Type	Non-Functional
Source	External Stakeholders
Category	Usability
Description	Training material that enables companies to better understand the roles and responsibilities of individual internal departments within the data cycle
Rationale	There is a need to upskill both technical and business personnel. Similarly, these materials could be used in a DATAMITE open source community
Priority	MUST
Test Case / Acceptance Criteria	There is a collection of information related to the business side of data

ID	R184
Type	Non-Functional
Source	External Stakeholders
Category	Usability
Description	The framework must be easy to use to facilitate its adoption by non-technical personnel
Rationale	We want to aid companies, and upskill their personnel, we cannot have a steep learning curve
Priority	MUST
Test Case / Acceptance Criteria	DATAMITE frontend must be highly usable

ID	R187
Type	Non-Functional
Source	Internal Technical Analysis
Category	Usability

Description	The framework must be compliant with ADA and WCAG under European guidelines
Rationale	Usability
Priority	MUST
Test Case / Acceptance Criteria	The framework must be compliant with https://www.accessibilitychecker.org/

ID	R188
Type	Functional
Source	Pilot Analysis
Category	Usability
Description	The framework could support multiple languages for user interfaces and content to cater to a diverse user base
Rationale	To ensure accessibility and user-friendliness for users who prefer different languages
Priority	COULD
Test Case / Acceptance Criteria	<ol style="list-style-type: none"> 1. The user could see a language toggle element 2. On selection, the framework could provide a translated page in the selected language

ID	R189
Type	Non-Functional
Source	Internal Technical Analysis
Category	Usability
Description	The open source resulting framework must be fully downloadable without any dependency with a proprietary component
Rationale	Availability
Priority	MUST
Test Case / Acceptance Criteria	IP Review

ID	R190
----	------

Type	Non-Functional
Source	Internal Technical Analysis
Category	Intellectual Property Rights
Description	None of the mandatory (vs. optional) library dependencies must be based on a Strong Copyleft license (e.g. GPL or AGPL)
Rationale	Industrial availability
Priority	MUST
Test Case / Acceptance Criteria	IP Review

ID	R191
Type	Non-Functional
Source	Internal Technical Analysis
Category	Intellectual Property Rights
Description	Each source code files, script files, config files, must have their copyright header setup properly indicating the owner (institution or organisation), the license selected, and, optionally, the developer ID. The license file must be included in each repository created by the project
Rationale	Exploitation
Priority	MUST
Test Case / Acceptance Criteria	IP Review

ID	R193
Type	Functional
Source	Internal Technical Analysis
Category	Compatibility
Description	In order to deploy the demonstration, an Azure-base sandbox must be created in the E-REDES IT systems
Rationale	Create the infrastructure system to deploy the components
Priority	MUST

Test Case / Acceptance Criteria	All of the resources necessary are deployed and ready to use
---------------------------------	--

ID	R194
Type	Functional
Source	Internal Technical Analysis
Category	Security
Description	The data support module should provide a generic anonymisation tool
Rationale	The anonymisation tool should possess the capability to anonymise data from various domains. It should not be designed exclusively for a specific domain
Priority	SHOULD
Test Case / Acceptance Criteria	The framework should be implemented, providing the possibility for anonymisation from different domains-industries

ID	R195
Type	Functional
Source	Internal Technical Analysis
Category	Security
Description	The anonymisation tool provides a feature that informs the user about the most suitable anonymisation technique for each field in the imported data
Rationale	The tool could empower users with information, guiding them on the selection of appropriate anonymisation techniques for each field in the imported data. This feature enhances user understanding and ensures effective and context-specific data anonymisation
Priority	COULD
Test Case / Acceptance Criteria	After importing the dataset by the user, the tool will inform the user in every column (field) of the dataset which anonymisation technique is appropriate for this field. For this implementation, data harmonisation is required (e.g. all datasets should have field name and not dataset1=name and dataset2=title)

ID	R196
Type	Functional

Source	Internal Technical Analysis
Category	Data Sharing
Description	The data sharing module should ensure integrity in the data sharing process
Rationale	In addition to storing data agreements, their hash should also be stored
Priority	SHOULD
Test Case / Acceptance Criteria	Store both the data agreement and its hash for each instance

ID	R197
Type	Functional
Source	Internal Technical Analysis
Category	Data Sharing
Description	The data sharing module must provide a monitoring and auditing tool within the framework to transparently oversee transactions with external users, ensuring visibility and clarity in the handling of sensitive data sharing
Rationale	A transparent monitoring tool is essential to provide visibility into transactions with external users, fostering trust and clarity in data-sharing processes
Priority	MUST
Test Case / Acceptance Criteria	Monitor the transactions with the integrated EU Portals and Dataspaces

ID	R198
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	The framework should provide a workflow tool to orchestrate the processes needed to create a data product from a (set of) operational data source(s)
Rationale	It will allow to orchestrate the different processes in a graphical way, useful for non-technical users

Priority	SHOULD
Test Case / Acceptance Criteria	The framework is able to orchestrate some of the processes to deploy and define a data product, like the provision of new storage, the data quality profiler or the anonymisation module

ID	R199
Type	Functional
Source	Internal Technical Analysis
Category	Security
Description	Check security compliance requirements and track user activities and accesses leveraging on auditing and reporting capabilities
Rationale	To alert possible policy violation issues
Priority	COULD
Test Case / Acceptance Criteria	The framework could expose only policy compliant datasets to the user

ID	R200
Type	Functional
Source	Internal Technical Analysis
Category	Security
Description	The anonymisation tool provides a feature that informs the user about sensitive and non-sensitive values in the imported dataset
Rationale	The tool should recognise sensitive and quasi-identifiers values of a dataset. This feature helps users to recognize the type of the fields of the dataset
Priority	SHOULD
Test Case / Acceptance Criteria	After importing the dataset by user, the tool will inform the user near every column which fields are sensitive or quasi identifiers

ID	R201
Type	Functional
Source	Internal Technical Analysis

Category	Functionality
Description	The framework must aggregate data from various sources
Rationale	Merge different datasets into a new one
Priority	MUST
Test Case / Acceptance Criteria	Provide a tool that receives at least two datasets as input and exports a new dataset

ID	R202
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	The framework is designed to facilitate the creation of a new data product by extracting a subset from the original dataset
Rationale	This feature is essential to support custom data reduction, allowing users to define specific rules for extracting a subset to generate a new dataset from the original. E.g. create a subset of SCADA measurements of a specific year or a specific geographic area from a larger dataset
Priority	MUST
Test Case / Acceptance Criteria	Verify that the framework successfully exports new datasets created from a subset of the original dataset. Ensure that the exported dataset adheres to the specified rules for data reduction and accurately represents the desired subset

ID	R203
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	The framework could facilitate the exploration of data by providing an overview of the dataset
Rationale	Allow users to drill down into detailed information from visualisations or reports, such as creating a table with data summaries (volume, field names, etc.)
Priority	COULD

Test Case / Acceptance Criteria	Generate a summary of information based on the received data
---------------------------------	--

ID	R204
Type	Functional
Source	Internal Technical Analysis
Category	Functionality
Description	The framework could log the data preprocessing actions (e.g. aggregation, anonymisation)
Rationale	Maintain a database or storage system capable of storing the performed actions
Priority	COULD
Test Case / Acceptance Criteria	Represent actions to identify usage patterns and perform statistical analysis

ID	R205
Type	Functional
Source	Pilot Analysis
Category	Compatibility
Description	The framework could enable using common standardised (e.g. based on ISO/IEC 9075:2023) SQL queries to different RDBMS that are currently in use
Rationale	This business requirement will allow users to combine information in an efficient way
Priority	COULD
Test Case / Acceptance Criteria	Deploy two different RDBMS (MariaDB and PostgreSQL) and make a query that will fetch data and aggregate them

Table 8: Complete Requirements Tables

5.2 Requirements removed compared to D1.2

This section presents explanations about why some of the originally identified requirements have been removed.

ID	Description	Justification
R001	As a pilot provider, I would like to target more audience who can make use of my data/analysis	This is a business requirement from WP4
R002	As a pilot provider, I would like to define a business plan based on users' profiles and functionalities	This is a business requirement from WP4
R003	As a pilot provider, I would like to assess the FAIRNESS of both data and services	Merged with other requirement(s) due to overlapping
R004	The KPI value assessment tool (Task 4.2) could also include KPIs related to Task 4.4, for valuing datasets from a non-monetary perspective	This is a business requirement from WP4
R005	The KPI value assessment tool (Task 4.2) should retrieve as an input the key sector related with the dataset (e.g. automobile, agrifood, health, etc.), as the value of a dataset will differ depending on the sector it targets	This is a business requirement from WP4
R006	The DATAMITE team should push for the adoption of the KPI tool, and point out at the tool itself which marketplaces/data spaces are implementing it as a reference for dataset valuation	This is a business requirement from WP4
R008	As PT Pilot Provider, I want to make a business analysis of the costs of exploiting data sharing using the DATAMITE framework, so that we can assess if we can reduce our current costs of exploiting the open data platform	Out of scope of DATAMITE
R010	As a PT Pilot provider, I want to be able to measure the financial costs of technological infrastructure that supports the connection to the dataspace	Out of scope of DATAMITE
R034	Expression of pilot data using the elements of a different data model should be feasible	Merged with other requirement(s) due to overlapping
R036	As an EDIH, I want that my data is harmonised automatically independently of the market segment where I work, so that it can be consumed by several services without them having to adapt their tools (quite unlikely to happen)	Merged with other requirement(s) due to overlapping
R040	As a pilot provider, I want that the metadata of the pilots is agreed between all the pilots, so that I can satisfy my needs with a seamless approach during integration	Out of scope of DATAMITE
R041	As a data provider, I want that the metadata is synchronised / replicated / shared without modifications to the agreed policy, so that I can ensure the quality and use of my data	Out of scope of DATAMITE

ID	Description	Justification
R043	As a pilot provider, I want a connector that is compatible with the ALoD, so that I can push the metadata available to the AI community to be consumed by the services	Merged with other requirement(s) due to overlapping
R060	The system should present enough information from metadata in order to ease the access to thematic data on a democratic way	Too vague and generic requirement for this stage of the project
R061	As a pilot provider, I would like to improve metadata exposition for enhanced dissemination	Merged with other requirement(s) due to overlapping
R065	As a data provider, I want that my data to be available to service consumers, so that they can use it	Too vague and generic requirement for this stage of the project
R066	As an AI developer/service consumer, I want to be able to access datasets through the ALoD, so that I can retrieve the data from any of the federated sources on DATAMITE	Out of scope of DATAMITE
R071	As a data hub provider/domain provider, I would like to establish governance and policy for my data space	Merged with other requirement(s) due to overlapping
R077	Utilise tools or services from other EU initiatives and projects to enhance data governance	Too vague and generic requirement for this stage of the project
R078	As a PT Pilot Provider, I want a review of the existing open source Data Governance tools, how they compare to each other and how they compare to the Azure integrated Data Governance tools (e.g., Microsoft Purview and Unity Catalogue) and their role on the implementation and operation of Data Spaces, so that we can determine which one is best in terms of ease of deployment, costs, ease of use, data quality functionalities, lineage functionalities and integration with emerging Big Data development tools such as Databricks	Out of scope of DATAMITE
R081	The framework requires a data catalogue that allows linking data entities/fields with terms in the data glossary	Merged with other requirement(s) due to overlapping
R086	A uniform vocabulary/data format must be defined so that there is data quality, which depends on its accuracy, completeness, and consistency. This requires data cleansing, verification and corrective actions	Merged with other requirement(s) due to overlapping

ID	Description	Justification
R092	Data sovereignty should not contradict the requirements of the framework, but should complement and support them. This can ensure that the DATAMITE system, as a whole, functions efficiently and guarantees the protection of the data sovereignty of all stakeholders	Too vague and generic requirement for this stage of the project
R097	As PT Pilot Provider, I will identify the different types of datasets that will be shared in the project and their own specifications, so that we can deploy the demonstration	Out of scope of DATAMITE
R104	Measure quality correctly regardless of the type of data, its form of ingestion (dump, streaming, etc.) or other conditions, and allow tools that contrast the necessary definition of quality against the quality computed and offered by the provider	Too vague and generic requirement for this stage of the project
R107	As a user, I want that the system provides information about the data quality and completeness before to share it, so that I know the type of data that I have	Merged with other requirement(s) due to overlapping
R114	A database management system will offer efficient storage space utilisation by having a centralised storage for all data rather than having several copies of data for each application and user. This helps to control data redundancy and maintain data integrity. Additionally, it allows data sharing and avoidance of data inconsistency	Out of scope of DATAMITE
R115	As a system administrator, I need a data filtering system/policy	Out of scope of DATAMITE
R127	As PT Pilot Provider, I want to set up technical networking roles that allow for the necessary operation of components and follow the internal security policies of the company	Out of scope of DATAMITE
R128	The solution must offer data privacy functionalities that will enable data anonymisation/pseudo anonymisation of different data sources dynamically and/or encrypt sensitive information that is stored or processed in the system. This solution must be applied on top of data sources before data transfer and be aligned with data privacy requirements and legislation. Solution must also offer a deanonymisation process in order to support system to system integration with operational systems	Merged with other requirement(s) due to overlapping
R129	As a data provider, I want to be able to apply different Data Anonymisation Techniques to my data	Merged with other requirement(s) due to overlapping

ID	Description	Justification
R131	The system administrator needs to know if a data breach happened	Out of scope of DATAMITE
R132	As a pilot provider, I would like to enhance and validate the security level of the data access services through VAPT	Out of scope of DATAMITE
R134	To model the structural and behaviour aspects of the DATAMITE platform and annotate the resulting modules with security and privacy requirements	Out of scope of DATAMITE
R135	Analyse the structural aspects of software modules of the DATAMITE platform with respect to secure dependency policy	Out of scope of DATAMITE
R136	Analyse the structural aspects of software modules of the DATAMITE platform with respect to the secure links policy	Out of scope of DATAMITE
R137	Analyse the software modules of the DATAMITE platform with respect to the access control policies	Too vague and generic requirement for this stage of the project
R140	There must be a tool which notices the published datasets. This tool could be a front app or a marketplace system	A front app or marketplace is out of the scope of DATAMITE. This task belongs to the data catalogue, and it is defined in other requirements
R142	As PT Pilot Provider, I want to be able to feed data from the internal system, at least weekly, to the project's dedicated data lake, so that I am able to perform the demonstration	Out of scope of DATAMITE
R145	The framework should implement telemetry functionalities	Merged with other requirement(s) due to overlapping
R148	As an end user, I would like to have improved visualisation capabilities	Too vague and generic requirement for this stage of the project
R151	We must have a CMS to manage content that is not provided by the APIs	Out of scope of DATAMITE
R161	As PT Pilot Provider, I want to be able to set up the necessary IT infrastructure according to the processing needs of the components to be implemented, through the MS Azure environment	Out of scope of DATAMITE

ID	Description	Justification
R164	The framework could be able to load datasets into different storage technologies for its processing	Out of scope of DATAMITE. We have discarded to integrate other storage technologies in the framework that are not MiniIO
R171	As a pilot provider, I would like to optimise and standardise the data ingestion workflow	Too vague and generic requirement for this stage of the project
R175	The system should offer scalability capabilities in order to support increasing demand for data access and processing. Ideally solution should be able to scale up and if possible to scale down based on the selected infrastructure	Out of scope of DATAMITE
R177	The system should be able to host at least the initially identified volumes of data (2 TB for six months historicity) and also be able to host future needs (extending six months historicity)	Out of scope of DATAMITE
R178	The system should be able to respond on average in a similar timing with the current infrastructure	Out of scope of DATAMITE
R185	As a pilot provider, I would like to exploit non-monetary societal benefits, while providing a safe and publicly acceptable solution	This is a business requirement from WP4
R186	As a pilot provider, I would appreciate guidelines for reflections on the impacts of data sharing, before establishing the practice in my organisation	This is a business requirement from WP4
R192	Each code contributor must sign the Eclipse Contributor Agreement [https://www.eclipse.org/legal/ECA.php] (Developer Certificate of Origin) before pushing their code in the project	This is not a requirement for the DATAMITE framework

Table 9: Deprecated Requirements