



# **Data Collection, Security, Storage & Management Services Bundles - Release 1.00**

D5.5

June 2022

# Deliverable

PROJECT ACRONYM	GRANT AGREEMENT #	PROJECT TITLE
TWINERGY	957736	Intelligent interconnection of prosumers in positive energy communities with twins of things for digital energy markets

## DELIVERABLE REFERENCE NUMBER AND TITLE

# D5.5 Data Collection, Security, Storage & Management Services Bundles – Release 1.00

Revision: v1.0

### AUTHORS

Anastasios Tsitsanis	Marios Phinikettos
Suite5	Suite5



Funded by the Horizon 2020 programme of the European Union  
Grant Agreement No 957736

## DISSEMINATION LEVEL

- ✓ P Public
- C Confidential, only for members of the consortium and the Commission Services

# Version History

REVISION	DATE	AUTHORS	ORGANISATION	DESCRIPTION
V0.1	12.07.2022	Anastasios Tsitsanis Marios Phinikettos	Suite5	1 <sup>st</sup> draft of the deliverable D5.5 - Data Collection, Security, Storage & Management Services Bundles – Release 1.00
V0.2	15.07.2022	Konstantinos Kotsalos	ED	Peer review of the document. Minor typos/syntactic improvements
V0.3	15.07.2022	Moisés Antón García Ana Isabel Martínez García	ETRA	Peer review of the document. Minor comments/Corrections
V0.4	19.07. 2022	Fynn Christian Bollhöfer	TH-OWL	Peer review of the document. Formatting suggestions for correction
V0.5	21.07.2022	Anastasios Tsitsanis Marios Phinikettos	Suite5	Consolidated Version sent to PC
V1.0	22.07.2022	Anastasios Tsitsanis Marios Phinikettos	Suite5	Final version submitted to EC by the PC

## Statement of Originality

This deliverable contains original unpublished work except where clearly indicated otherwise.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

---

# Legal Disclaimer

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The above referenced authors will have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials subject to any liability which is mandatory due to applicable law. The sole responsibility for the content of this publication lies with the authors. It does not necessarily reflect the opinion of the European Union. Neither the CINEA nor the European Commission is responsible for any use that may be made of the information contained therein.

© 2022 by TwinERGY Consortium

# Executive Summary

Deliverable D5.5 "Data Collection, Security, Storage & Management Services Bundles – Release 1.00" summarizes three (3) tasks, namely Task 5.2 "Data Management Platform Backbone Infrastructure," Task 5.3 "Core Data Ingestion, Curation, and Management Services," and Task 5.4 "Data security, encryption, and privacy mechanisms" of WP5 "Data Collection and Communication Platform", as depicted in the TwinERGY DoA [1] and builds on D5.2 "Data Collection, Security, Storage & Management Services Bundles – Beta Release, [2]". The current deliverable provides the description Release 1.00 of the respective services bundles. The data availability in the pilot sites, the data requirements of the TwinERGY modules, and the nine (9) use cases of the TwinERGY project were analysed, as explained in the deliverable D5.3 "TwinERGY Integrated Data Management Platform – Alpha, Mock-ups Release, [3]" in order to define the functional requirements for the development of the TwinERGY CDMP and the different services that are offered by the Platform. Each service introduces its own set of functionalities.

This deliverable dives into the development details of the services, covering the following aspects:

- The current status of various functionalities provided from each service, as well as the functionalities introduced in D5.2 "Data Collection, Security, Storage & Management Services Bundles – Beta Release".
- The frameworks and libraries used to implement each service, as well as their versions.
- The interface specifications that were created, in order to document the technical details of each service's APIs.

The current deliverable is the 1.00 version of TwinERGY Core Data Management Platform's services on M20. The delivered services are going to be represented with appropriate screenshots in D5.6 "TwinERGY Integrated Data Management Platform–Release 1.00" [3], which will illustrate the development activities of the TwinERGY Integrated Data Management Platform Release 1.00 on M24 (October 2022).

# Index

1. Introduction.....	9
1.1 Purpose of this deliverable.....	9
1.2 Scope of this deliverable.....	9
1.3 Structure of the document.....	10
1.4 Abbreviation List .....	10
2. TwinERGY Core Data Management Platform and Services Overview .....	12
2.1 TwinERGY Core Data Management Platform.....	12
2.2 Services overview .....	12
2.2.1 Data Collection Service .....	14
2.2.1.1 State of Development .....	15
2.2.1.2 Technological Infrastructure.....	16
2.2.1.3 Interface Specification.....	17
2.2.2 Data Security Service.....	21
2.2.2.1 State of Development .....	21
2.2.2.2 Technological Infrastructure.....	22
2.2.2.3 Interface specification .....	22
2.2.3 Data Storage Service .....	24
2.2.3.1 State of Development .....	25
2.2.3.2 Technological Infrastructure.....	25
2.2.3.3 Interface Specification.....	25
2.2.4 Platform Management Service.....	26
2.2.4.1 State of Development .....	26
2.2.4.2 Technological Infrastructure.....	26
2.2.4.3 Interface Specification.....	27
2.3 Services Installation and Access.....	28
2.3.1 Services Installation.....	28
2.3.2 Licensing and access .....	28
3. Conclusions .....	29
REFERENCES .....	30

---

Annex.....31

---

# List of Figures

Figure 1 The TwinERGY Core Data Management Platform conceptual architecture..... 12



# List of Tables

Table 1 Requirements 019 and 020, that led to the enhancements on file ingestion methods .....	13
Table 2 Requirements 017 and 018, that led to UI/UX enhancements of the TwinERGY CDMP .....	14
Table 3 Start a new data ingestion job interface specification .....	17
Table 4 Retrieve a data ingestion job interface specification .....	17
Table 5 Delete a data ingestion job interface specification .....	18
Table 6 Ingester Configuration interface specification .....	18
Table 7 Ingester Retrieval interface specification .....	18
Table 8 Mapper Configuration interface specification .....	19
Table 9 Mapper Retrieval interface specification.....	19
Table 10 Curator Configuration interface specification .....	19
Table 11 Curator Retrieval interface specification.....	20
Table 12 Metadata Editor Configuration interface specification.....	20
Table 13 Metadata Editor Retrieval interface specification .....	21
Table 14 Access Policy Creation interface specification .....	22
Table 15 Access Policy Retrieval interface specification.....	23
Table 16 Access Policy Update interface specification .....	23
Table 17 Access Policy Deletion interface specification .....	23
Table 18 Anonymization Handler Configuration interface specification .....	24
Table 19 Anonymization Handler Retrieval interface specification.....	24
Table 20 User Retrieval interface specification .....	27
Table 21 User Update interface specification .....	27
Table 22 User Deletion interface specification .....	28
Table 23 User Password Update interface specification.....	28
Table 24 Functional Requirements of the TwinERGY CDMP .....	32

---

# 1. Introduction

## 1.1 Purpose of this deliverable

The backbone of the project is an "open," modular, and interoperable big data management platform, as shown in the TwinERGY DoA [5], which enables open standards-based data collection and management communication throughout the project's energy value chain. The TwinERGY Data Management Platform complies with current open energy standards (such as OpenADR, USEF, etc.) and includes a homogenized Common Information Model that guarantees semantic interoperability of the digitalized energy assets used in the pilot sites as well as seamless integration, communication, and operation on top of any Energy Management System and Smart Home systems and devices. In order to protect end-user data confidentiality and the non-repudiation of DER assets, the TwinERGY Data Management Platform is equipped with the necessary data security, privacy, authentication, and authorization procedures.

As described in deliverable D5.3 "TwinERGY Integrated Data Management Platform – Alpha, Mock-ups Release", along with relevant mockups, a thorough analysis of the data availability in the pilot sites, the data requirements of the TwinERGY modules, and the way that the TwinERGY use cases are covered was conducted, in order to define the functional requirements for the development of the TwinERGY CDMP and the various services offered by the platform.

The various functionalities introduced to the Data Collection, Data Security, Data Storage, and Platform Management services, as well as those implemented during the beta release, are thoroughly detailed in this deliverable. As an enhanced version of the TwinERGY Core Data Management Platform Services Bundles, D5.5 is a more developed and feature-rich version of D5.2, detailing the upgrades and improvements made in comparison to the beta release, such as enhancements on file ingestion methods, CIM improvements, UI/UX enhancements and bug fixing and component refactoring, as detailed in section 2.2.1 Outline of services updates. D5.6 - "TwinERGY Integrated Data Management Platform– Release 1.00" will go through how the various services are implemented in the TwinERGY CDMP's User Interface.

## 1.2 Scope of this deliverable

The major scope of D5.5 is to provide a comprehensive overview of the TwinERGY Core Data Management Platform Services version 1.00. The deliverable goes through the

TwinERGY CDMP's services in greater detail, as well as their functionalities. Data Collection, Data Security, Data Storage, and Platform Management services, are all described in detail by:

- Depicting the implemented functionalities for each service, as well as the functionalities introduced in the beta release.
- Describing the libraries and frameworks used to implement each service.
- Documenting the development of relevant APIs that make communication and information sharing across services easier.

### 1.3 Structure of the document

The structure of the document is as follows:

Section 2 provides a brief description of the TwinERGY Core Data Management Platform conceptual architecture, the services and their functionalities, setting the basis for the next section. It also provides an overview of the platform's services and shows how the functionalities of each service are developed. Additionally, the technological infrastructure that was used to establish each service and the specifications of the various interfaces that have been created, together with the appropriate licensing and access methods are reported.

Deliverable D5.5 "Data Collection, Security, Storage, and Management Services Bundles – Release 1.00" is concluded with Section 3, which provides a brief synopsis of what has been covered in the various sections of the deliverable.

### 1.4 Abbreviation List

Acronym	Full Name
API	Application Programming Interface
CDMP	Core Data Management Platform
CIM	Common Information Model
CSV	Comma-separated values
D	Deliverable
DCAT	Data Catalog Vocabulary

DoA	Description of Action
DER	Distributed Energy Resources
DMP	Data Management Platform
GDPR	General Data Protection Regulation
JSON	JavaScript Object Notation
M	Month
UI	User Interface
WP	Work Package
XML	Extensible Mark-up Language

## 2. TwinERGY Core Data Management Platform and Services Overview

### 2.1 TwinERGY Core Data Management Platform

Data Collection, Data Security, Data Storage, and Platform Management are all part of the TwinERGY Core Data Management Platform, as depicted in Figure 1. The platform's services and their respective functionalities are detailed and further examined in the sections that follow throughout this deliverable.

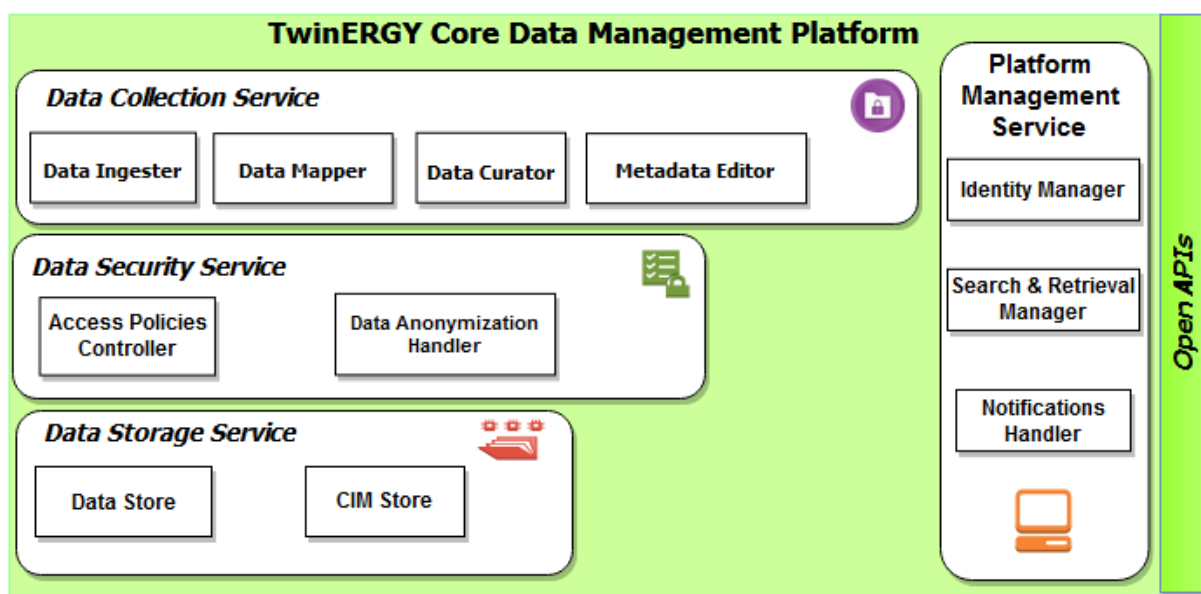


Figure 1 The TwinERGY Core Data Management Platform conceptual architecture

The above architecture of the CDMP was introduced, in order to address the requirements that were identified, after the completion of the activities and evaluation of the findings and feedback obtained from the TwinERGY Uses Cases elaboration, the TwinERGY pilot sites data availability activities and the identification of the various data requirements of the different modules to be developed in the context of the TwinERGY project.

In the following sections, a detailed overview of the services and of their functionalities is provided, along with the state of development of each service's functionalities and the interfaces specifications created to describe the technical details of each service's APIs.

### 2.2 Services updates and overview

#### 2.2.1 Outline of services updates

Between the beta release of the Data Collection, Security, Storage & Management Services Bundles and release 1.00, a number of enhancements were made to the CDMP,

its services, and their respective functionalities. By simultaneously addressing the needs that emerged from the elaboration of the TwinERGY Use Cases, the availability of data in the pilot sites, and the identification of the various data requirements of the modules, these improvements were meant to enhance the Core Data Management Platform's overall performance. More specifically, these improvements refer to:

1. Enhancements on file ingestion methods, enabling platform users to append files to an already created data ingestion job. Parallel to this, changes were made to the platform's overall data ingestion management when it comes to handling large amount of data. This was also part of the TwinERGY Core Data Management Platform's refactoring process with the aim of improving the platform's performance, while maintaining its functionalities, by improving the scheduling process of the data ingestion jobs for optimized execution, which enables the ingestion and the faster upload of larger data files, addressing also Req\_019 and Req\_020 of the TwinERGY CDMP Functional Requirements, as shown below in the respective rows of the requirements list. It is important to note that Req\_020 is a newly added requirement that enhances the initial list of requirements. The entire list of requirements can be found in the Annex section.

Req_019	The platform should be able to handle big volumes of data.
Req_020	The platform should allow users to append new data to already created datasets.

*Table 1 Requirements 019 and 020, that led to the enhancements on file ingestion methods*

2. CIM improvements, with the extension of the underlying TwinERGY Common Information Model (creation of additional fields, based on pilots' data samples) in order to support the modelling requirements of the anticipated data to be handled within the platform.
3. UI/UX enhancements, guaranteeing a more user-friendly UI and improving the overall user experience. The TwinERGY CDMP's various services have undergone the necessary changes which enable users of the platform to receive more messages when leveraging its various functionalities. Additionally, the changes made, allowed for more user-friendly user flows and concise messaging in the event of platform functionality errors along with the proper notifications. These enhancements also address requirements Req\_017 and Req\_018 of the TwinERGY CDMP Functional Requirements, as shown below in the corresponding rows of the requirements list. These changes will be made visible to the platform users in the TwinERGY Integrated Data Management Platform– Release 1.00, on M24 (October 2022).

Req_017	The platform should ensure an intuitive, easy-to-use user interface.
Req_018	The platform should ensure a user-interface with minimum latency.

*Table 2 Requirements 017 and 018, that led to UI/UX enhancements of the TwinERGY CDMP*

4. Bug fixing of identified errors and refactoring of specific components for the overall increase of the performance of the TwinERGY Core Data Management Platform. The TwinERGY Platform was rigorously examined by the involved development team after the TwinERGY Integrated Platform Beta Release (D5.4 - M16) in order to identify any bugs and prioritize them for immediate fix or inclusion in the backlog for the upcoming TwinERGY CDMP release (M24 - October 2022).

In the following sections, an updated detailed overview of the services and of their functionalities is provided, along with the state of development of each service's functionalities and the interfaces specifications created to describe the technical details of each service's APIs.

### *2.2.2 Data Collection Service*

The Data Collection Service introduces a variety of functionalities that are relevant to the data collection process, including: i) Data Ingestor, which introduces proper procedures and methods for ingesting data into the CDMP; ii) Data Mapper, which allows ingested datasets to be aligned with the CIM; iii) Data Curator, which ensures the datasets quality; and iv) Metadata Editor, which enables the users of the platform to define specific aspects of the ingested dataset, in order to provide a detailed profile of the given dataset.

The appropriate handling of data management procedures enables data collection setup, allowing users to specify the data ingestion method, according to which the data is uploaded to the TwinERGY CDMP. The platform integrates data from a variety of sources in a straightforward manner. Users uploading data use the platform to clarify the way and the type of data they own and are ready to share, as well as to define the parameters of the data ingestion process, outline their needs and preferences on the way that the data are inserted in the platform (users APIs, or file uploading), and perform proper mapping actions on all fields of the data, all of which are based on the TwinERGY Common Information Model.

The Data Mapper functionality determines how the fields of a dataset are aligned with the concepts of the TwinERGY CIM. When the CIM and the dataset are matched, a consistent domain-specific understanding of the data is established, making it easier to use for data-related services in the CDMP. To guarantee adherence with the CIM, the application of appropriate mapping and transformation routines on datasets allows for the changing of field names and the conversion of measurement units.

Additionally, during the curation configuration process, the Data Curator functionality allows the user to specify the restrictions and limitations that the data ingested into the platform may have, as well as the precise activities undertaken if any of these constraints are not satisfied. This configuration refers to a set of curation routines that are performed by carrying out the required data curation processes.

Any result of data ingestion is saved as a dataset in the TwinERGY Core Data Management Platform and with the Metadata Editor functionality, the users of the platform can specify a comprehensive dataset profile, by determining the dataset title, description and additional metadata in alignment with the DCAT [7] recommendations.

### 2.2.2.1 State of Development

The Data Collection Service release, as specified in this deliverable, introduces a range of functionalities that are described below.

The functionalities that were part of the beta release of the Data Collection Services are depicted as follows:

- Data ingestion process configuration: describes in detail the needs and preferences on the way that the data are ingested into the TwinERGY Core Data Management Platform (file uploading, API, streaming data).
- Ingestion of data from files: allows data retrieval from files in popular file formats (e.g., csv).
- API Data Ingestion: This functionality allows data to be retrieved from both pilot system APIs and Open Data APIs (e.g., weather data, other local sources).
- Reliable and safe data upload: Data is uploaded to the TwinERGY Core Data Management Platform using reliable and secure processes.
- Mapping activities on ingested data: This functionality ensures that the data values are matched to the CIM provisions (e.g., data types, measurement units) by performing the necessary mapping actions on the ingested data.
- Dataset & metadata creation: Any outcome of a data ingestion is recorded as a dataset in the Data Storage Service of the TwinERGY Core Data Management Platform with this functionality. Users of the platform can give a dataset a title and associated metadata, as well as describe what is contained inside it.

The functionalities that are part of the 1.00 release, extending the functionalities of the beta release, of the Data Collection Services are depicted as follows:

- Appending data in data files: With this functionality, the TwinERGY CDMP users are able to append new data in already stored datasets within the platform.
- Large file uploading: This functionality ensures that bigger volumes of data can be ingested in the TwinERGY CDMP, in a single dataset.
- Faster uploading: This functionality ensures that data are ingested faster in the Core Data Management Platform, due to the refactoring performed.



### 2.2.2.2 Technological Infrastructure

The back - end of the Data Collection Service is developed with **Python 3.10**<sup>1</sup>, using also a range of open-source technologies, namely **Django 4.0**<sup>2</sup>, **Pandas 1.3.0**<sup>3</sup> and **Docker 20.10.11**<sup>4</sup>.

**Python** is a high-level, general-purpose programming language, which emphasises code readability with the use of indentation. Its language constructs as well as its object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects<sup>5</sup>.

**Django** is a high-level Python web framework that enables rapid development of secure and maintainable websites. Built by experienced developers, Django takes care of much of the hassle of web development, so you can focus on writing your app without needing to reinvent the wheel. It is free and open source, has a thriving and active community, great documentation, and many options for free and paid-for support<sup>6</sup>.

**Pandas** is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series<sup>7</sup>.

**Docker** is an open-source containerization platform. It enables developers to package applications into containers—standardized executable components combining application source code with the operating system (OS) libraries and dependencies required to run that code in any environment<sup>8</sup>.

The front- end of the Data Collection Service is developed with **Vue.js V2**<sup>9</sup>.

**Vue** is a progressive framework for building user interfaces. Unlike other monolithic frameworks, Vue is designed from the ground up to be incrementally adoptable. The core library is focused on the view layer only and is easy to pick up and integrate with other libraries or existing projects.

<sup>1</sup> <https://www.python.org/downloads/release/python-3100/>

<sup>2</sup> <https://www.djangoproject.com/weblog/2021/dec/07/django-40-released/>

<sup>3</sup> <https://pandas.pydata.org/docs/whatsnew/index.html>

<sup>4</sup> <https://docs.docker.com/engine/release-notes/>

<sup>5</sup> <https://analyticsindiamag.com/python-3-10-and-its-new-features/>

<sup>6</sup> <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Django/Introduction>

<sup>7</sup> [https://en.wikipedia.org/wiki/Pandas\\_\(software\)](https://en.wikipedia.org/wiki/Pandas_(software))

<sup>8</sup> <https://www.ibm.com/in-en/cloud/learn/docker>

<sup>9</sup> <https://vuejs.org/v2/guide/>

### 2.2.2.3 Interface Specification

<b>Name</b>	Start a new data ingestion job		
<b>Description</b>	Initiates the data ingestion process		
<b>Reference Code</b>	DCS01		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	POST	N/A	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twenergy-api/data-collection/		
<b>Interaction with Services</b>	Data Collection Service		

*Table 3 Start a new data ingestion job interface specification*

<b>Name</b>	Retrieve a data ingestion job		
<b>Description</b>	Initiate the data ingestion retrieval process		
<b>Reference Code</b>	DCS02		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	GET	id: The identifier of the data collection	id (string) title (string) user (string) configuration (json) created (datetime) status (string)
<b>Endpoint URI</b>	/twenergy-api/data-collection/{id}		
<b>Interaction with Services</b>	Data Collection Service		

*Table 4 Retrieve a data ingestion job interface specification*

<b>Name</b>	Delete a data ingestion job		
<b>Description</b>	Deletes a data ingestion job		
<b>Reference Code</b>	DCS03		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	DELETE	id: The identifier of the data collection	200 OK 401 Unauthorized 403 Forbidden 404 Not Found

<b>Endpoint URI</b>	/twinegy-api/data-collection/{id}
<b>Interaction with components</b>	Data Collection Service

*Table 5 Delete a data ingestion job interface specification*

<b>Name</b>	Ingestor Configuration		
<b>Description</b>	Creates the data ingestor settings		
<b>Reference Code</b>	DCS04		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	POST	id: The identifier of the data ingestor configuration	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twinegy-api/data-collection/data-ingester-config/{id}		
<b>Interaction with Services</b>	Data Collection Service		

*Table 6 Ingestor Configuration interface specification*

<b>Name</b>	Ingestor Retrieval		
<b>Description</b>	Retrieves the data ingestor settings		
<b>Reference Code</b>	DCS05		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	GET	id: The identifier of the data ingestor retrieval	id (string) step-id (string) title (string) configuration (json) created (datetime) status (string)
<b>Endpoint URI</b>	/twinegy-api/data-collection/data-ingester-config/{id}		
<b>Interaction with Services</b>	Data Collection Service		

*Table 7 Ingestor Retrieval interface specification*

<b>Name</b>	Mapper Configuration		
<b>Description</b>	Creates the data mapper settings		
<b>Reference Code</b>	DCS06		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method

	POST	id: The identifier of the data mapper configuration	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twenergy-api/data-collection/ data-mapper-config /{id}		
<b>Interaction with Services</b>	Data Collection Service		

*Table 8 Mapper Configuration interface specification*

<b>Name</b>	Mapper Retrieval		
<b>Description</b>	Retrieves the data mapper settings		
<b>Reference Code</b>	DCS07		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	GET	id: The identifier of the data mapper retrieval	id (string) step-id (string) title (string) configuration (json) created (datetime) status (string)
<b>Endpoint URI</b>	/twenergy-api/data-collection/data-mapper-config/{id}		
<b>Interaction with Services</b>	Data Collection Service		

*Table 9 Mapper Retrieval interface specification*

<b>Name</b>	Curator Configuration		
<b>Description</b>	Creates the data curator settings		
<b>Reference Code</b>	DCS08		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	POST	id: The identifier of the data curator	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twenergy-api/data-collection/data-curator-config /{id}		
<b>Interaction with Services</b>	Data Collection Service		

*Table 10 Curator Configuration interface specification*

<b>Name</b>	Curator Retrieval
<b>Description</b>	Retrieves the data curator settings

<b>Reference Code</b>	DCS09		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	GET	id: The identifier of the data curator retrieval	id (string) step-id (string) title (string) configuration (json) created (datetime) status (string)
<b>Endpoint URI</b>	/twenergy-api/data-collection/data-curator-config/{id}		
<b>Interaction with Services</b>	Data Collection Service		

*Table 11 Curator Retrieval interface specification*

<b>Name</b>	Metadata Editor Configuration		
<b>Description</b>	Creates the metadata editor settings		
<b>Reference Code</b>	DCS10		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	POST	id: The identifier of the metadata editor configuration	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twenergy-api/data-collection/metadataeditor-config/{id}		
<b>Interaction with Services</b>	Data Collection Service		

*Table 12 Metadata Editor Configuration interface specification*

<b>Name</b>	Metadata Editor Retrieval		
<b>Description</b>	Retrieves the metadata editor settings		
<b>Reference Code</b>	DCS11		
<b>Service</b>	Data Collection		
<b>Type of interface</b>	REST		
<b>Inputs</b>	Methods of the interface	Parameters of the method	Return Values of the method
	GET	id: The identifier of the metadata editor retrieval	id (string) step-id (string) title (string) configuration (json) created (datetime) status (string)
<b>Endpoint URI</b>	/twenergy-api/data-collection/metadataeditor-config/{id}		

<b>Interaction with Services</b>	Data Collection Service
----------------------------------	-------------------------

*Table 13 Metadata Editor Retrieval interface specification*

### 2.2.3 Data Security Service

The TwinERGY CDMP users' data security and privacy concerns about data that are ingested and handled in the TwinERGY Core Data Management Platform are addressed by the Data Security Service. As a result, it includes a number of functions and functionalities that are easily configured in the TwinERGY Platform, such as the Access Policies Controller and the Data Anonymization Handler.

The Access Policies Controller functionality enhances user confidence in the entire TwinERGY Core Data Management Platform by allowing users to create access policies that flexibly and clearly regulate access requests to data within the TwinERGY CDMP. The access policy controller effectively handles policy enforcement at run-time so that only authorized platform users can view and access the data. This functionality allows platform users to define and configure access policy routines, allowing these routines to be based on data fields and requestor properties. Routines are saved and users uploading data can easily modify them. After the access policy is defined and configured, the policy is applied when access to the data contained in the CDMP is requested.

As for the Anonymization handler functionality, it ensures the protection of the data from inadvertent disclosure of personal and company information. Before the data becomes available on the TwinERGY platform, anonymization actions are performed and reflected in the defined part of the data (i.e., fields). The users of the CDMP are provided with the appropriate interventions on the data in case they need to anonymize fields in their data that contain personally identifiable information.

#### 2.2.3.1 State of Development

The Data Security Service release, as specified in this deliverable, introduces a range of functionalities that are described below.

The functionalities that were part of the beta release of the Data Security Services are depicted as follows:

- Backend access policy management and handling: Policies are used to determine when access is permitted.
- Backend enforcement of data access policies: The access policies that have been set are applied when a request is made to access data that has been ingested into the TwinERGY CDMP.

### 2.2.3.2 Technological Infrastructure

The back - end of the Data Security Service is developed with **Python 3.10**, using also a range of open-source technologies, namely **Django 4.0**, and **Pandas 1.3.0**, as documented in 2.2.2.2 Technological Infrastructure.

The access policy management library of the Data Security Service is **CASL v4**<sup>10</sup>. **CASL** is an isomorphic authorization JavaScript library which restricts what resources a given client is allowed to access. It is designed to be incrementally adoptable and can easily scale between a simple claim based and fully featured subject and attribute-based authorization. It makes it easy to manage and share permissions across UI components, API services, and database queries.

### 2.2.3.3 Interface specification

<b>Name</b>	Access Policy Creation		
<b>Description</b>	Creates an access policy for a dataset		
<b>Reference Code</b>	DSS01		
<b>Service</b>	Data Security		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	POST	id: The identifier of the dataset	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twinegy-api/dataset/{id}/policy		
<b>Interaction with Services</b>	Data Security Service		

*Table 14 Access Policy Creation interface specification*

<b>Name</b>	Access Policy Retrieval		
<b>Description</b>	Retrieves all access policies for a dataset		
<b>Reference Code</b>	DSS02		
<b>Service</b>	Data Security		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>

<sup>10</sup> <https://casl.js.org/v4/en/guide/intro>

	GET	id: The identifier of the dataset	id (string) step-id (string) title (string) policies (json) created (datetime) status (string)
<b>Endpoint URI</b>	/twenergy-api/dataset/{id}/policy		
<b>Interaction with Services</b>	Data Security Service		

*Table 15 Access Policy Retrieval interface specification*

<b>Name</b>	Access Policy Update		
<b>Description</b>	Updates an access policy for a dataset		
<b>Reference Code</b>	DSS03		
<b>Service</b>	Data Security		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	POST	id: The identifier of the dataset  policy_id: The identifier of the access policy	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twenergy-api/dataset/{id}/policy/{id}		
<b>Interaction with Services</b>	Data Security Service		

*Table 16 Access Policy Update interface specification*

<b>Name</b>	Access Policy Deletion		
<b>Description</b>	Deletes an access policy for a dataset		
<b>Reference Code</b>	DSS04		
<b>Service</b>	Data Security		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	DELETE	id: The identifier of the dataset  policy_id: The identifier of the access policy	200 OK 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twenergy-api/dataset/{id}/policy/{id}		
<b>Interaction with Services</b>	Data Security Service		

*Table 17 Access Policy Deletion interface specification*



<b>Name</b>	Anonymization Handler Configuration		
<b>Description</b>	Creates the data anonymization handler settings		
<b>Reference Code</b>	DSS05		
<b>Service</b>	Data Security		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	POST	id: The identifier of the data anonymization handler configuration	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twinegy-api/data-collection/data-anonymization-handler-config/{id}		
<b>Interaction with Services</b>	Data Security Service		

*Table 18 Anonymization Handler Configuration interface specification*

<b>Name</b>	Anonymization Handler Retrieval		
<b>Description</b>	Retrieves the data anonymization handler settings		
<b>Reference Code</b>	DSS06		
<b>Service</b>	Data Security		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	GET	id: The identifier of the data anonymization handler retrieval	id (string) step-id (string) title (string) configuration (json) created (datetime) status (string)
<b>Endpoint URI</b>	/twinegy-api/data-collection/data-anonymization-handler-config/{id}		
<b>Interaction with Services</b>	Data Security Service		

*Table 19 Anonymization Handler Retrieval interface specification*

### 2.2.4 Data Storage Service

This service meets the demand for dependable data storage and indexing by providing resilient measures and different indexing methodologies. Depending on the type of information stored in the TwinERGY Platform and the means for retrieval, different storage solutions are utilised to meet various needs. The Data Storage Service is in charge of securely storing a wide range of data, as well as the metadata that goes with it, in a reliable and secure manner.

#### 2.2.4.1 State of Development

The Data Storage Service release, as specified in this deliverable, introduces a range of functionalities that are described below.

The functionalities that were part of the beta release of the Data Storage Services are depicted as follows:

- **Data Store:** This functionality entails storing various types of data (e.g., datasets) together with their associated metadata, making it accessible to all TwinERGY Core Data Management Platform services.
- **CIM Store:** Storage of the TwinERGY Common Information Model, as well as all necessary concepts and fields.

The functionalities that are part of the 1.00 release, extending the functionalities of the beta release, of the Data Storage Services are depicted as follows:

- **CIM Extension:** Extension of the TwinERGY CIM with the creation of additional fields, based on the data samples from the TwinERGY pilot sites.

#### 2.2.4.2 Technological Infrastructure

The back - end of the Data Storage Service is developed with the utilization of **PostgreSQL 14**<sup>11</sup> and **Elasticsearch 7.14**<sup>12</sup>.

**PostgreSQL** is a powerful, open source object-relational database system with over 30 years of active development that has earned it a strong reputation for reliability, feature robustness, and performance<sup>13</sup>.

**Elasticsearch** is a distributed, free and open search and analytics engine for all types of data, including textual, numerical, geospatial, structured, and unstructured. Known for its simple REST APIs, distributed nature, speed, and scalability, Elasticsearch is the central component of the Elastic Stack, a set of free and open tools for data ingestion, enrichment, storage, analysis, and visualization<sup>14</sup>.

#### 2.2.4.3 Interface Specification

There are no external APIs that are exposed by the Data Storage Service to accompany its initial version except for the APIs that have been presented in sections 2.2.2.3, 2.2.3.3 and 2.2.5.3.

<sup>11</sup> <https://www.postgresql.org/about/news/postgresql-14-released-2318/>

<sup>12</sup> <https://www.elastic.co/blog/whats-new-elasticsearch-7-14-0>

<sup>13</sup> <https://www.postgresql.org/>

<sup>14</sup> <https://www.elastic.co/what-is/elasticsearch>

## 2.2.5 Platform Management Service

This service establishes the tools and processes for users to register on the platform in a safe and reliable manner. Through appropriate authentication and security measures, users are given access to the data that they are authorized to use. When data ingestion takes place in the platform, users are alerted via platform notifications based on their preferences. The service has three main functionalities: the (i) Identity Manager, the (ii) Search & Retrieval Manager, and the (iii) Notifications Handler.

Summarized, with the development of the Platform Management Service:

- Users register, authorize, and authenticate themselves using the TwinERGY CDMP's identity management functionality, which is interconnected with security-related processes.
- Users are able to search for and find data that may be proven valuable, as well as determine and define which of these available data are relevant to their needs, using the Search & Retrieval functionality.
- The Notifications handling functionality is in charge of detecting events related to the status and progress of data ingestion in order to provide the TwinERGY Core Data Management Platform's users with appropriate material.

### 2.2.5.1 State of Development

The Platform Management Service release, as specified in this deliverable, introduces a range of functionalities that are described below.

The functionalities that were part of the beta release of the Platform Management Services are depicted as follows:

- Identity Management: This functionality enables the creation and management of Core Data Management Platform users allowing for the establishment and controlling of identifying information for those who are permitted to use the platform.

### 2.2.5.2 Technological Infrastructure

The back-end of the Platform Governance Module is developed with **Python 3.10**, using also a range of open-source technologies, namely **Django 4.0**, as documented in 2.2.2.2 Technological Infrastructure

**Websockets 10.1**<sup>15</sup> and **Kubernetes 1.22.2**<sup>16</sup> are utilized for server-side notifications implementation and for the system deployment respectively.

<sup>15</sup> <https://websockets.readthedocs.io/en/stable/>

<sup>16</sup> <https://kubernetes.io/blog/2021/08/04/kubernetes-1-22-release-announcement/>

**Websockets** is a library for building WebSocket servers and clients in Python with a focus on correctness, simplicity, robustness, and performance.

**Kubernetes** is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation. It has a large, rapidly growing ecosystem. Kubernetes services, support, and tools are widely available<sup>17</sup>.

### 2.2.5.3 Interface Specification

<b>Name</b>	User Retrieval		
<b>Description</b>	Retrieves the specifics for a user		
<b>Reference Code</b>	PMS01		
<b>Service</b>	Platform Management		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	GET	id: The identifier of the user	username (string) name (string) e-mail (string)
<b>Endpoint URI</b>	/twenergy-api/user/{id}		
<b>Interaction with Services</b>	Platform Management Service		

*Table 20 User Retrieval interface specification*

<b>Name</b>	User Update		
<b>Description</b>	Updates the specifics for a user		
<b>Reference Code</b>	PMS02		
<b>Service</b>	Platform Management		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	POST	id: The identifier of the user	200 OK 201 Created 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twenergy-api/user/{id}		
<b>Interaction with Services</b>	Platform Management Service		

*Table 21 User Update interface specification*

<b>Name</b>	User Deletion
<b>Description</b>	Deletes a user

<sup>17</sup> <https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/>

<b>Reference Code</b>	PMS03		
<b>Service</b>	Platform Management		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	DELETE	id: The identifier of the user	200 OK 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twinergy-api/user/{id}		
<b>Interaction with Services</b>	Platform Management Service		

*Table 22 User Deletion interface specification*

<b>Name</b>	User Password Update		
<b>Description</b>	Updates the password of a user		
<b>Reference Code</b>	PMS04		
<b>Service</b>	Platform Management		
<b>Type of interface</b>	REST		
<b>Inputs</b>	<u>Methods of the interface</u>	<u>Parameters of the method</u>	<u>Return Values of the method</u>
	PUT	id: The identifier of the user	200 OK 401 Unauthorized 403 Forbidden 404 Not Found
<b>Endpoint URI</b>	/twinergy-api/user/{id}/password		
<b>Interaction with Services</b>	Platform Management Service		

*Table 23 User Password Update interface specification*

## 2.3 Services Installation and Access

### 2.3.1 Services Installation

A private repository contains the Data Collection, Data Security, Data Storage, and Platform Management Services. Because all elements are already packed as Docker containers, the relevant private code repository includes explicit instructions for the service's deployment.

### 2.3.2 Licensing and access

Only closed-source code has been used in the development of the Data Collection, Data Security, Data Storage, and Platform Management services. The versions of the different services that are to be deployed will be accessed through the TwinERGY project's Core Data Management Platform.

---

## 3. Conclusions

Deliverable D5.5, "Data Collection, Security, Storage, and Management Services Bundles - Release 1.00," described the enhancements and improvements made in comparison to Deliverable D5.2, "Data Collection, Security, Storage, and Management Services Bundles - Beta Release," and served as the updated version of the TwinERGY Core Data Management Platform Services Bundles. It also contained a concise report outlining the implementation status of each service's functionalities, the technological infrastructure used, and the established interface specifications. The deliverable described the updated version of the four (4) developed TwinERGY CDMP services, namely the Data collection, Data security, Data storage, and Platform Management Services, and provided more information for each of the services, including:

- The current status of various functionalities provided from each service, as well as the functionalities introduced in D5.2
- The technological infrastructure of each service, including the version used.
- The interface specifications that were created.

The current deliverable is the 1.00 release of the services made available on M20, in compliance with the TwinERGY DoA, which provides back-end implementation details and sets the stage for the subsequent 1.00 release of the TwinERGY Integrated Data Management Platform to be made available on M24 (D5.6 – October 2022). It is important to note that the TwinERGY Platform's services are constantly being developed and will continue so till M32. Future releases of the TwinERGY Core Data Management Platform will take into account any alterations and enhancements made as the project progresses, as well as any additional requirements.

---

# REFERENCES

- [1] DoA Part A, TwinERGY Consortium Agreement No 957736
  
- [2] TwinERGY Consortium. (2020). TwinERGY D5.2 "Data Collection, Security, Storage & Management Services Bundles – Beta Release"
  
- [3] TwinERGY Consortium. (2020). TwinERGY D5.3 "TwinERGY Integrated Data Management Platform – Alpha, Mockups Release"
  
- [4] TwinERGY Consortium. (2020). TwinERGY D5.6 "TwinERGY Integrated Data Management Platform– Release 1.00"
  
- [5] DoA Part B, TwinERGY Consortium Agreement No 957736
  
- [6] TwinERGY Consortium. (2020). TwinERGY D5.4 "TwinERGY Integrated Platform– Beta Release"
  
- [7] Data Catalog Vocabulary (DCAT) - Version 2 (w3.org)

# Annex

Requirement_ID	Description	Requirement Status
Req_001	The platform shall have access to DER management system data.	Ongoing
Req_002	The platform shall provide the possibility for the ingestion of real-time datasets.	Ongoing
Req_003	The platform shall have access to Energy Management Systems data.	Ongoing
Req_004	The platform shall have access to Smart Home Systems data.	Ongoing
Req_005	The platform shall provide the possibility to communicate with the DERs to receive data.	Ongoing
Req_006	The platform shall provide the possibility to communicate with the relevant modules to send data.	Ongoing
Req_007	The platform might have access to grid data.	Ongoing
Req_008	The platform shall have access to sensor data.	Ongoing
Req_009	The platform shall have the capability to consume data provided from gateways.	Ongoing
Req_010	The platform shall allow the mapping and storage of all available datasets under a common information model (CIM).	Completed
Req_011	The platform should provide the possibility to a user to upload data files (csv, json).	Completed
Req_012	The platform shall provide the definition of metadata of available datasets.	Completed



Req_013	The platform should allow the user to define data anonymization routines.	Pending
Req_014	The platform should allow the user to define data curation routines on the data that the user owns.	Completed
Req_015	The platform shall formulate and enforce a data access control decision based on the associated data access policies.	Completed
Req_016	The platform should enable the periodic data retrieval from Open Data APIs.	Completed
Req_017	The platform should ensure an intuitive, easy-to-use user interface.	Ongoing
Req_018	The platform should ensure a user-interface with minimum latency.	Ongoing
Req_019	The platform should be able to handle big volumes of data.	Ongoing
Req_020	The platform should allow users to append new data to already created datasets.	Ongoing

*Table 24 Functional Requirements of the TwinERGY CDMP*