TwinERGY Pre-trial validation testing scenarios and results

D8.2

DECEMBER 2022

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

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**

D8.2

TwinERGY Pre-trial validation testing scenarios and results

Revision: v1.0

AUTHORS

Ana Isabel Martínez García Moisés Antón García
ETRA I+D

CONTRIBUTORS

Giselle Morabito	Ivan Sulev	Konstantinos Kotsalos Apostolos Kapetanios
IES	WEC	ED

Fynn Christian Bollhöfer Axel Balke	Giogos Papadopoulos Alexandros Tsitsanis	Mattia Repossi
TH OWL	S5	STAM

Konstantinos Plakas Chris Mounzouris John Gialelis
UoP

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

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Version History

REVISION	DATE	AUTHOR	ORG...	DESCRIPTION
V0.1	14.12.2022	Moisés Antón, Ana Isabel Martínez	ETRA ID	Version to be reviewed by partners
V0.2	21.12.2922	Konstantinos Kotsalos; Apostolos Kapetanios	ED	Reviewed version
V0.3	22.12.2022	Alexandros Tsitsanis	Suite5	Scalability considerations need to be covered as part of this task (see Review Template). Also, minor comments and corrections within the document
V0.4	22.12.2012	Athanasios Chassiakos	UoP	Reviewed version
V0.5	29.12.2022	Moisés Antón, Ana Isabel Martínez	ETRA	Final version with comments and corrections integrated submitted to PC
V1.0	31.12.2022	Athanasios Chassiakos, Stylianos Karatzas, Vasiliki Lazari, Antonis Papamanolis	UoP	Final version submitted to EC by the PC

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

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This deliverable contains original unpublished work except where clearly indicated otherwise.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.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

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 shall 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.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Executive summary

Integrated laboratory testing activities are the final stage of the TwinERGY solutions before testing their deployment in project pilot sites. It should be mentioned here that during this testing phase, GDPR principles and rules were accomplished. This is important in order to ensure that the privacy and data protection principles remain on top priorities, for reliable data handling during interoperable applications.

The integrated laboratory testing corresponds to task 8.2 "TwinERGY system modules integration and lab testing" and its main scope is to evaluate the interoperability capabilities and the functionalities of the developed tools in an integrated environment. In order to achieve that goal, the necessary laboratory infrastructure and the appropriate simulation environment are used.

The first step before the start of this activity was to ensure that all partners would use the same terminology. To achieve this goal, common definitions were agreed. Hereafter, a plan for the execution and documentation of the laboratory tests has been defined. This plan consists of the following steps:

1. Review the project requirements and Use Cases.
2. Define the features of the tool that need to be tested focusing on laboratory integration and interoperability features.
3. Define detailed test cases for the validation of the features.
4. Execute the test cases.
5. Document the results of the test cases and the test protocols using a common format.

This deliverable will allow the description of the executed validation of each software component in a laboratory environment The real communication between the modules in tested as well in this phase, as a simulation for the communication that the modules will use in the real case. This testing phase has allowed developers to detect future problems in the final integration that they can solve in this previous stage without the need of involving users and reducing the time of the preparation of the real environment. The continuation of these integration activities in the real-life environment will continue in Task 8.3 "TwinERGY system final version"

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Index

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

List of Figures

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

List of Tables

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

1. Introduction

1.1. Scope of the document

The purpose of this document is to summarise the results from Task 8.2 "TwinERGY System Modules integration and lab-testing". In this task, the TwinERGY solutions are tested in an integrated environment in laboratory conditions before deploying them in the real demonstration sites. At these laboratory tests, the interoperability between the solutions deployed and the hardware integration is examined in order to address possible issues. This deliverable describes the test plan and presents the results of the executed tests performed for the purposes of the integrated lab testing activities. The technical integration of the TwinERGY modules was implemented by the project tool developers. To follow and evaluate the integration, a set of tests were performed for each tool, and they are presented in this document.

1.2. Structure of the deliverable

This document starts with a short overview of the integrated laboratory testing activity that took place during the task development (M21-M27). Then the definition of the laboratory testing procedure and the test cases that will be examined are presented and after that, the results of lab-testing phase are shown using the appropriate format. Finally, the document closes with the conclusions regarding the evaluation of the lab-testing results.

1.3. Abbreviation list

Table 1: Abbreviation List

Acronym	Full Name
API	Application Programming Interface
CDMP	Core Data Management Platform
CDT	Consumer Digital Twin

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

CPO	Charging Point Operator
DER	Distributed Energy Resources
DLT	Distributed Ledger Technology
DR	Demand Response
DSO	Distribution System Operator
DNO/DSO	Distribution Network/System Operator
GDPR	General Data Protection Regulation
IoT	Internet of Things
JSON	JavaScript Object Notation
LEM	Local Energy Market
LV	Low Voltage
MQTT	MQ Telemetry Transport
MV	Medium Voltage
PoA	Prove of Authority
PUC	Primary Use Case
PV	Photovoltaic
RES	Renewable Energy Systems
RPOA	Raspberry Pi Oracle

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

RTU	Remote Terminal Unit
TEM	Transactive Energy Market
TVOC	Total Volatile Organic Compounds
UC	Use Case
USEF	Universal Smart Energy Framework
VPPs	Virtual Power Plants

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

2.Objectives

The objective of this deliverable is the description of the executed validation of each software component in a laboratory environment, being this the first step, before the integration in the real environment. In this step, the different modules that process the information from pilot sites and give services to end users, are tested using simulated assets and/or assets available in a laboratory. On the other hand, the real communication between these modules is tested in this phase. That means that modules are deployed in a test environment, but they are sending and receiving information by using the same channels as in the real case.

During this test phase, developers detect future problems in the final integration, and they can solve them without involving users, and more importantly, reducing time of preparation of the real environment. So, despite the fact that those new problems that may appear during the final integration and the first hours of TwinERGY real system, this step is totally necessary.

As starting point, we have considered the use cases defined during task T4.4, and described in the D4.4 "System Architecture" [1], so developers have defined the set of test cases to probe the communication between components, as well defined there. Some minor variations from the original architecture have been produced, but they are covered in this deliverable.

Another point to have into account is that scalability testing is not considered here, due we are most focused on integration. The scalability testing will be done during task T8.3, in first steps of the final deployment, where we can check the intensive use of the applications.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

3. Integrated lab-testing approach

The approach that is used in this document is followed for the integrated lab-testing activities. This approach is based on the NOBEL GRID [2] project where the followed methodology has been proven to be successful and sufficient for this kind of activities.

As it has been described in section *2.Objectives*, the test cases reflect the validation to be done over the use cases defined during task T4.4. Therefore, for each tool, the developer is also responsible to test and evaluate the interoperability of their tool with the others (regarding hardware requirements in a lab environment).

The NOBEL GRID approach [2] proposes a set of definitions that are presented in the following table (see Table 2). These definitions were developed considering the state of the art in software, smart grid, and system integration testing, especially with respect to the IEEE 829 Standard [3] for software test documentation.

Table 2: Definition of terms

Term	Definition
Device under test	a product or software which is verified by a certain test case. It is part of the test environment
Expected results	a description of the status of the test environment after a test case was carried out and pass criteria have been met
Features (not) to tested	a list of product requirements or specifications which are (not) covered by a certain test case
Pass/fail criteria	a definition of how to judge or measure if a product under test conforms to specifications and requirements that shall be validated by a certain test case
Retesting	re-execution of a test case that previously returned a "fail" result, to evaluate the effectiveness of intervening correction actions
Subsystem acceptance criteria	conditions to be fulfilled by a subsystem for including it into the system integration test. Conditions should include the availability of testing protocols for standalone subsystem tests. Also, subsystems should have similar level of maturity

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Suspension criteria	a description of conditions which indicate that the test was carried out incorrectly or that any situation was produced which renders the testing results unusable, making test continuation pointless and requiring the test to be halted and restarted
System integration test	a test designed to verify that a system made up of two or more interacting products (subsystems) conforms to system-wide specifications and requirements. The device under test is the system itself. It is specially designed for finding inconsistencies which emerge only through the subsystem interaction. The system integration test plan may define partial system integration tests which allow for adding subsystems subsequently
(System integration test) Level	The number of system layers which are included in a system integration test case minus one
System layer	a group of one or more subsystems which is defined prior to the system integration test. According to group definition for a given system should be used for all system integration test cases
Testing	set of activities conducted to facilitate discovery, validation and/or evaluation of properties of one or more test TwinERGY components
Test analysis	elaboration about why a test result emerged. It may also include a conclusion about what the test result implies for the future work
Test case	a collection of features (not) to be tested, testing procedures, and pass/fail criteria used for testing a system or device under test. Test cases may refer to specific types of product requirements, e.g. the function, reliability, stability, safety, or vulnerability.
Test case code	an identifier for a test case which is unique throughout the project.
Test specification	documentation of one or more test cases
Test coverage	a list of product requirements or specifications which are verified by a test plan
Test data	data created or selected which is needed for executing one or more test cases. It may be defined in the test case specification
Test environment	a list of all elements (software, hardware, information, external conditions) needed to carry out a test case, including the device or system under test and all elements needed to judge the test outcome
Test environment set-up process	a list of actions needed for establishing and maintaining a required test environment

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Test execution	the actions needed to carry out the testing procedures for a given test case
Test group	a collection of test cases which share at least one defined criterium. E.g. all test cases which relate to cybersecurity testing might be defined to make up a test group
Test method	a general definition of testing procedures and test environment for a test plan
Test plan	a strategy or list of tasks used to verify that a product conforms to design specifications and product requirements
Test preparation	a definition of steps which are needed to prepare a test environment for test execution
Testing procedures	a specific list of steps which are needed to carry out a test case
Test protocol	a summary of the test results of all test cases defined in a test plan. It may also contain the test analysis for said test cases
Test requirements	a definition stating the status of the test environment which is needed for carrying out a specific test case or a test group. Ideally, it is also stated how it can be checked if the test environment is ready for test execution
Test responsibilities	a definition stating which persons or organizations are needed for the test. It may also include an assignment of tasks to those persons or organizations
Test result	an indication of whether a specific test case has passed or failed. May also include any data that has been obtained through execution of the test case

3.1 Test plan

To provide a common framework for all involved partners, the testing activities are all based on the following test plan:

1. Review the project requirements from architecture.
2. Define the features of the tool that need to be tested - Focusing on laboratory integration and interoperability features.
3. Define detailed test cases for the validation of the defined features.
4. Execute the test cases.
5. Document the results of the test cases and the test protocols.

These steps are described in the next subsections.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

3.2 Review of project requirements from architecture

This step consists of the review of integration requirements that need to be probed, using the architecture specified during WP4 as a starting point (see D4.4 – TwinERGY Architecture [1]). Due to the fact that the architecture status is in its integration phase, requirements are not related to the internal working of each component, but to the communication among components.

Another thing to be considered is that, during the implementation of each module and the final architecture of TwinERGY, some variations appeared in the architecture have considered them in order to define the most appropriated testing. that need to be considered in During the next subsections (3.2.1 to 3.2.14), the architecture analysed as well as the requirements that are derived from it are exposed. These subsections describe the use cases in TwinERGY, the Core Data Management Platform (CDMP), the iSCAN & Digital Twin platforms, the Interoperability platform and the TwinERGY Identity management platform. Each of them is described and its requirements are exposed. These requirements are defined following the standard RFC 2119 [4], in which one can find:

- The identifier of the requirement
- The affected asset: component or use case
- The responsible partner
- A detailed description of the requirement
- A priority, being 1 the lowest and 5 the highest one.

Just as a summary of that standard, the description of the requirement and the priority are closely related. The description has to follow this schema: “Asset + AUXILIAR VERB + verb + action”, being AUXILIAR VERB: "MUST”, “SHALL", “HAVE TO”, "SHOULD", "RECOMMENDED", "MAY", "OPTIONAL", etc. Depending on the auxiliar verb that is used in the requirement sentence, the priory will change. For instance, “Energy and environmental sensors HAVE TO send data to Home & Tertiary Energy Management Module” corresponds with priory 5/4 (the highest one).

3.2.1 UC01 - Home Energy Management

The aim of this Use Case (UC) is to test the energy management of residential consumer premises and their ability of monitoring and controlling electrical loads to maximize self-consumption and self-sufficiency, reduce the costs for the users also enhancing their active role into the energy efficiency process. A bottom-up approach is envisaged

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

focusing on the power grid at the buildings level. The first step is to obtain greater facilities' observability; the necessary amount of monitored data, both static and dynamic, are to be gathered and then processed and analysed. Data gathering is crucial in the energy efficiency process and energy management; depending on the data availability different actions can be taken and, eventually, the existing monitoring system is going to be improved.

The key-parameters are:

- Energy demand
- Photovoltaic production
- Electrical storage state of charge
- Indoor parameters (such as temperature and humidity)
- Weather conditions

The data monitoring and data collection will be achieved through a monitoring system on edge and through online monitoring services. Several data sets are going to be made available through an online graphical interface to improve the users' awareness about their energy patterns.

Starting from the main electrical appliances inventory and their daily usage habits, the optimal loads distribution during the day are going to be delivered. In relation to the energy monitoring and forecasts and the energy costs information, these data can be used to find and shape the optimal load profile, reducing costs in the energy bill and maximizing self-consumption.

As a summary, the HLUC01- Home Energy Management is subdivided in the next primary use cases, as was explained in deliverable D4.4 - "System Architecture":

- PUC01.01. Increase the building observability - Data gathering from the home monitoring system
- PUC01.02. Data analysis - Behavioural rules analysis, minimization of energy costs and increase of self-consumption from PV
- PUC01.03. Optimal flexibility management system - Analysis of the optimal electrical appliances flexibility management
- PUC01.04. Control of the smart devices

These use cases consider the communication among the elements listed in the following Table 3:

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Table 3. Assets actuating during UC01 processes

Component	Component Type
Building level energy meters	Physical asset
Energy smart plugs	Physical asset
Environmental sensors	Physical asset
PV Panel and Inverter	Physical asset
PV smart meter	Physical asset
Home battery storage	Physical asset
Home storage smart meter	Physical asset
Raspberry Pi (gateway)	Physical asset
LV Grid	Physical asset
Core Data Management Platform	TwinERGY application
Digital Twin Platform	TwinERGY application
iSCAN	TwinERGY application
Interoperability Platform	TwinERGY application
Home & Tertiary Energy Monitoring Module	TwinERGY application
Risk Management Module	TwinERGY application
Consumer Demand Flexibility Profiling Module	TwinERGY application
RES Integration & DER Management Module	TwinERGY application
Keycloak	TwinERGY application
ENTSO-E transparency platform	External Application
Home Energy Management System (HEMS)	External Application

Based on the assets list, we consider the requirements presented in Table 4 that need to be validated during the testing.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Table 4. Integration requirements related to UC01

Req. Identifier	Asset	Responsible partner	Description	Priority
R01.001	Energy sensors	STAM	Energy and environmental sensors *HAVE TO* send data to Home & Tertiary Energy Management Module	5
R01.002	Smart plugs	STAM	Home & Tertiary Energy Management Module web application *HAS TO* show the logged user its appliances' telemetries data referring to the last 7 days	5
R01.003	Environmental sensor	STAM	Home & Tertiary Energy Management Module web application HAS TO show the logged user eventual anomalies that have been detected and are related to its appliances	5
R01.004	Comfort & Well-being web app	STAM	Home & Tertiary Energy Management Module web application HAS TO let the logged user browse its building possible threats, showing also countermeasures and financial impacts	5
R01.005	Physiological sensors	STAM	Home & Tertiary Energy Management Module HAS TO let the logged user control its appliances remotely through the use of input sliders	5
R01.006	DCMP	Suite5	The monitoring data *MUST* be available in the CDMP	5
R01.007	iScan	STAM	iScan MUST share the asset building configuration in a json file with an API REST	5
R01.008	H&T module	STAM	Home & Tertiary Energy Management Module MUST be able to read from the DCMP:	5

			• Energy demand • Energy generation • Voltages • Energy appliances demand	
R01.019	H&T module	STAM	The fault anomaly detection service in Home & Tertiary Energy Management Module MUST be able to read from the DCMP: • Energy demand • Energy generation • Indoor temperature	5

3.2.2 UC02 - RES generation in domestic and tertiary buildings

This use case has the goal to create further renewable sources and infrastructure to increase the RES share in public and private buildings. The use case is being applied to the pilot sites for the TwinERGY project, with different aims for each specific location in question.

The objectives in each of pilot sites are quite similar in the context of this use case, but the plan will be still to be as flexible as possible to allow for any specific requirements of one location so that the optimal solution can be found for local RES in the community.

As a summary, the UC02 was subdivided in the next primary use cases, as was explained in deliverable D4.4 - "System Architecture":

- PUC02.01: "Building Scale Demand Profile", where the information of energy demand is collected at the building level and provided through the building digital twin
- PUC02.02: "Community Scale Demand Profile", where the information of energy demand is collected at the Community level, which would aggregate the building level demand while also integrating V2G and community level renewables
- PUC02.03: "Next Day Optimised Operation of Smart Appliances for Single Customers", where demand profiles for single customers will be optimised based on the current RES production as well as the current energy storage to minimize either cost or carbon emissions based on the user's requirements.
- PUC02.04: "Next Day Optimised Operation of Smart Appliances at Community Level", where demand profiles for single customers will be optimised based on the current RES production as well as the current energy storage to minimize either cost or carbon emissions based on the Community requirements.
- PUC02.05: "Next Day Forecasting of RES Production", where the forecast of the RES production takes place, and an optimisation algorithm implements demand

shifting to maximise self-consumption of RES/minimise energy cost/ minimise carbon cost

- PUC02.06: "Peak Load Control at Community Level", where the optimisation algorithm will identify the relative flexibility in the demand profiles and provide feedback to the user as to how to adjust their consumption to reduce peak load or minimise cost/carbon as required
- PUC02.07: "Optimization of Future Design Scenarios using Disparate RES Solutions", where demand response opportunities at building and community level would be identified in terms of performance and financial requirements.

These use cases consider the communication among the elements listed in the following Table 5.

Table 5. Assets actuating during UC02 processes

Component	**Component Type**
Core Data Management Platform	TwinERGY application
iSCAN	TwinERGY application
Consumer Demand Flexibility Profiling Module	TwinERGY application
Transactive Energy Platform (TEM)	TwinERGY application

As the main goal of this phase is the testing of the communication between elements, we consider these requirements as the needed ones to be validated during the testing:

Table 6. Integration requirements related to UC02

Req. Identifier	**Asset**	**Responsible partner**	**Description**	**Priority**
R02.001	iSCAN	IES	iSCAN HAS TO allow users to define the building and data sources for the target building when automatically creating the iSCAN project	5
R02.002	iSCAN	IES	iSCAN HAS TO allow users to connect channels to the Data Sources	5

R02.003	Transactive Energy Platform	WEC	iSCAN HAS TO allow the Transactive Energy Platform to be integrated in a dedicated channel	5
R02.004	iSCAN	IES	iSCAN HAS TO initiate automatically the algorithm to populate channels with up-to-date data	5
R02.005	Consumer Demand Flexibility Profiling Module	IES	Consumer Demand Flexibility Profiling Module MUST optimize building and community demand profiles for cost and RES production based on user preferences in iSCAN.	5

3.2.3 UC03 - Grid capacity enhancement utilizing e-mobility

The aim of this Use Case is to test how the Electric Vehicles (VEs) can be a distributed storage asset, able to stabilize the grid and lead to more decarbonized neighbourhoods. At individual level, the project offers to the users:

1. Lower energy costs, by the reduction of the electricity bills due to smart charging in VEs.
2. Reduction of CO2 emissions by promoting the charging of VEs when there is a surplus of solar production in their domestic PV installations or when the RES production from the MV grid is high (registered in the country energy mix).

At community or neighbourhood level, TwinERGY offers ancillary services to the Distribution System Operator (DSO) for congestion and voltage management. This is done by considering V2G capabilities of EV batteries.

As a summary, the HLUC03 was subdivided in the next primary use cases:

- PUC03.01 – "Booking a charge session", where the user selects a charging point and reserves it in advance of the charge session.
- PUC03.02 – "Smart Charging to follow grid requests", where the current charge session is restricted by the grid status and DSO orders.
- PUC03.03 – "Smart Charging to maximize RES integration", where the current charge session considers the grid status and maximizes the injection of energy in those time slots where the energy is mainly generated by RES.
- PUC03.04 – "Smart Charging to minimize charge costs", where the current charge session considers the grid status and maximizes the injection of energy in those time slots where the energy is cheaper.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

- PUC03.05 – "Smart Charging to minimize time of charge", where the current charge session considers the grid status and maximizes the injection of energy in those time slots in which the energy availability is higher This implies that the duration of the session is reduced.
- PUC03.06 – "Grid Management", where the DSO sends orders to the Charge Points Operators (CPOs) in order to solve a problem in the grid.

These use cases consider the communication among the elements listed in next table:

Table 7. Assets actuating during UC03 processes

Component	Component Type
Charging Point	Physical asset
Charging Point Operator	Organization
Charging Point Operator Gateway	External application
Community battery storage	Physical asset
Community battery storage smart meter	Physical asset
Consumer Demand Flexibility Profiling Module	TwinERGY application
Core Data Management Platform	TwinERGY application
Digital Twin Platform	TwinERGY application
Electric Vehicle	Physical asset
ENTSO-E transparency platform	External Application
Grid gateway	External application
HEMS	External Application
Home battery storage	Physical asset
Home storage smart meter	Physical asset
Interoperability Platform	TwinERGY application
Inverter	Physical asset
iSCAN	TwinERGY application
LV Grid	Physical asset

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

MV Grid	Physical asset
MV-LV Transformer	Physical Asset
Neighbourhood Demand Flexibility Profiling Module	TwinERGY application
PV Panel	Physical asset
PV smart meter	Physical asset
Raspberry Pi	Physical asset
RES Integration & DER Management Module	TwinERGY application
Remote Terminal Unit	Physical asset
Transactive Energy Blockchain	TwinERGY application
Transactive Energy Platform	TwinERGY application
TwinEV Module	TwinERGY application

As the main goal of this phase is the test of the communication between elements, we consider these requirements the needed one to be validated during the testing:

Table 8. Integration requirements related to UC03

Req. Identifier	Asset	Responsible partner	Description	Priority
R03.001	Charge Point, Charge Point Operator, Charging Point Operator Gateway	ETRAID	The CP MUST be locked when TwinEV module sends a lock order to the CPO	5
R03.002	Charge Point, Charge Point Operator, Charging Point Operator Gateway	ETRAID	The CP MUST be unlocked when TwinEV module sends an unlock order to the CPO	5
R03.003	TwinEV module	ETRAID	A locked point MUST NOT be reserved when it is locked	5
R03.004	RES Integration & DER	TH-OWL	RES Integration & DER Management MUST send the RES production forecast after its calculations	5

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	Management module			
R03.005	Neighbourhood demand flexibility profiling module	IES	Neighbourhood demand flexibility profiling module MUST send the neighbourhood flexibility profile to Interoperability platform after its calculations	5
R03.006	RES Integration & DER Management module	TH-OWL	RES Integration & DER Management MUST send grid status after loading them from the grid	5
R03.007	Consumer Demand Flexibility Profiling module	IES	Consumer Demand Flexibility Profiling MUST send User appliances flexibility to Interoperability platform after its calculations	5
R03.008	Transactive Energy Platform	WEC	Transactive Energy Platform Must send LEM pricing to Interoperability platform after its calculations	5
R03.009	TwinEV module	ETRAID	For smart charging, TwinEV module MUST be able to read inputs • RES production forecast, • Neighbourhood flexibility profile, • User appliances flexibility • LEM pricing from Interoperability platform.	5
R03.010	TwinEV module	ETRAID	For both smart chargers to follow grid request and maximize RES integration, TwinEV module MUST read inputs Actual consumption and production from Core Data Management Platform	5
R03.011	TwinEV module	ETRAID	TwinEV module MUST send charge curve to CPO before starting a charge session	5
R03.012	TwinEV module	ETRAID	TwinEV module MUST be able to read grid status from Interoperability platform	5
R03.013	TwinEV module	ETRAID	When a grid operator set a restriction to charge points, TwinEV module MUST send minimal and maximal power to the affected charging points	5

R03.014	TwinEV module	ETRAID	When a charging session has finished, TwinEV module MUST send Charging session cost to Interoperability platform	5
R03.015	Transactive Energy Platform	WEC	Transactive Energy Platform HAS TO be able to load charging session cost from the Interoperability platform	5

3.2.4 UC04 - Prosumer's empowerment in local energy trading markets

The scope of this use case is to provide solutions to transactive energy use cases and enables grid decentralization and democratization by connecting the micro-grid operators to the DER managers and their customers. It aims for an integrated energy business model through energy service expansion, customer engagement and financial inclusion. It allows operators ~~them~~ to balance the grid and provide solutions to a number of grid challenges, such as grid power quality and reliability. The core of this use case is a transactional platform that offers its participants the opportunity to sell their flexible energy loads and excess capacity on an open market to the (micro) grid operators or to each other. Microgrid operators provide balancing and grid services at a local and micro-grid level.

A micro-grid could be a collection of a) IoT devices, b) buildings, c) neighbourhoods/substations, and d) regions that operate at a regional level to balance energy use in multiple neighbourhoods, districts and/or substations. It could potentially include the high voltage grid. Each component of the system (e.g., device, building, neighbourhood, distribution grid and transmission grid) is a self-contained ecosystem, replicated and nested within the next layer of the system, like in a fractal configuration. All components operate with identical information and control models and each have operational decision-making capabilities. This platform offers a path to grid decentralization, energy democratization, and a way to effectively leverage and monetize the emerging DER infrastructure. The Transactive Energy Platform (TEM), based on Hybrid Blockchain technologies, will be developed to solve current intractable optimization problems and create a premiere Transactive Energy (TE) protocol layer settlement process, marketplace, and governance framework to allow energy-related Apps to be written and interoperate with each other. As a summary, the UC04 was subdivided in the next primary use cases, as was explained in deliverable D4.4 - "System Architecture":

- PUC04.01. Recording transactions of energy: Recording transactions of energy from a RES to a private or public storage facility, recording transactions of energy between prosumers and Recording transactions of energy between prosumer

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

consortia. In addition, Demand Response events are simulated, and prosumers are rewarded for their positive response.

- PUC04.02. Calculation and broadcasting of LEM pricing compared to (Distribution Network/System Operators (DNO/DSO) pricing.

These use cases consider the communication among the elements listed in next table:

Table 9. Assets actuating during UC04 processes

Component	Component Type
Smart Meter	Physical Asset
Raspberry Pi	Physical Asset
Smart Battery	Physical Asset
Interoperability Platform	TwinERGY application
Raspberry Pi Oracle	External application
Transactive Energy Platform	TwinERGY application
Transactive Energy Blockchain Network	TwinERGY application
HEMS gateway	External application
HEMS	External Application
iSCAN	TwinERGY application

As the main goal of this phase is the test of the communication between elements, we consider these requirements as the needed ones to be validated during the testing:

Table 10. Integration requirements related to UC04

Req. Identifier	Asset	Responsible partner	Description	Priority
R04.001	Physiological smart meters	WEC	Physiological smart meters HAVE TO make activity readings and transmit to the Raspberry Pi Oracle application (RPOA) and cDT platform	5
R04.002	Raspberry Pi Oracle	WEC	RPOA HAS TO store and transmit activity readings to the Transactive Energy Blockchain network respective smart contract.	5

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R04.003	Raspberry Pi Oracle	WEC	RPOA data HAVE TO update the Smart meter data smart Contract and send the change (POST) on the Blockchain ledger.	5
R04.004	Transactive Energy Platform	WEC	Storage of Energy or Usage of stored energy MUST mint, burn or transfer kWh tokens	5
R04.005	Transactive Energy Platform	WEC	RPOA data HAVE TO be settled on a set interval to reflect accurate energy generation, usage and storage	5
R04.006	Transactive Energy Blockchain Network	WEC	Transactive Energy Platform HAS TO be able to submit sell orders to the P2P sale logic smart contract.	5
R04.007	Transactive Energy Platform	WEC	When a sale is completed, Transactive Energy Platform HAS TO instruct the HEMS to schedule an energy transfer between Smart Batteries.	5
R04.008	Transactive Energy Blockchain Network	WEC	When the transfer of energy is completed, the P2P sale logic smart contract HAS TO transfer kWh and TwinERGY tokens from and to the buyer and seller respectively.	5
R04.009	Transactive Energy Platform	WEC	Based on a synchronization of DER reward with the Social Network Module, Transactive Energy Platform MUST instruct the Smart Contract to mint and distribute DER tokens	5
R04.010	Transactive Energy Platform	WEC	Transactive Energy Platform HAS TO transmit the LEM price to the Smart Contract and store the data on the ledger	5

3.2.5 UC05 - Enhance grid flexibility through DER Management

The main goal of this Use Case is to test the capabilities of the respective TwinERGY modules to appropriately manage the usage of renewable energies and thus extending the periods over the time windows of actual production (e.g., night times, low wind speeds).

A successful implementation shall result in a higher ratio of self-consumption and a shift of consumption behaviour towards eco-friendlier times of the day.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

As a summary, the UC05 was subdivided in the next primary use cases, as was explained in deliverable D4.4 - "System Architecture":

- PUC05.01 – "Prediction of energy consumption and RES production", where a forecasting of consumption and production allows DSOs to proactively apply changes to the switching behaviour in the grid and distribute energy flows as well as energy reserves in battery stores more efficiently.
- PUC05.02 – "Utilizing the Virtual-Power-Plant" (VPP), focused on maximizing the production of renewable energy as well as acting as a grid stabilizing component, so while sun or wind powered energy producers can be part of the VPP by shutting down production in times of low energy demand

These use cases consider the communication among the elements listed in the next table:

Table 11. Assets actuating during UC05 processes

Component	Component Type
RES Integration & DER Management Module	TwinERGY application
Interoperability Platform	TwinERGY application
Core Data Management Platform	TwinERGY application
Consumer Demand Flexibility Profiling Module	TwinERGY application
H&T Energy Monitoring Module	TwinERGY application
Weather Forecast Service	External application
iSCAN	TwinERGY application
Wind Farm	Physical asset
PV Panels	Physical asset
Community Battery Storage	Physical asset
LV Grid	Physical asset

As the main goal of this phase is the test of the communication between elements, we consider these requirements the needed one to be validated during the testing:

Table 12. Integration requirements related to UC05

Req. Identifier	Asset	Responsible partner	Description	Priority

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R05.001	DER Management Module	TH OWL	The RES Integration & DER Management Module MUST get the PV and wind-power configuration of the respective pilot-site	5
R05.002	DER Management Module	TH OWL /SUITE5	The RES Integration & DER Management Module MUST get latest weather forecasts for the next 24 hours	5
R05.003	DER Management Module	TH OWL	The RES Integration & DER Management Module MUST calculate the proper Incentive Curve (ICC) for the given timeframe	5
R05.004	DER Management Module	TH OWL /ETRAID	The RES Integration & DER Management Module MUST send the ICC to the NATS server	5

3.2.6 UC06 - Consumer's engagement in Demand Side Management Programs utilizing feedback mechanisms

The aim of this Use Case is to test the capabilities of the respective TwinERGY modules to let the residents actively participate in demand side management by supplying information about the energy production and obtaining feedback based on recommendations for energy consumption. In dynamic energy tariffs, this use case can also result in lower costs for the end-user by the increased share of renewable energy.

As a summary, the UC06 was subdivided in the next primary use cases, as was explained in deliverable D4.4 - "System Architecture":

- PUC06.01. "Increase residential demand flexibility"
- PUC06.02. "Decrease residential energy use"

These use cases consider the communication among the elements listed in the following Table 13.

Table 13. Assets actuating during UC06 processes

Component	Component Type
DER Management Module	TwinERGY application
Interoperability Platform	TwinERGY application
Core Data Management Platform	TwinERGY application
Energy Light	Physical asset
Home & Tertiary Energy Monitoring Module	TwinERGY application

Consumer Demand Flexibility Profiling Module	TwinERGY application
HEMS	External application
HEMS Gateway	External application
Home Battery Storage	Physical asset
Energy asset & Controller	Physical assets

As the main goal of this phase is the test of the communication between elements, we consider these requirements the needed ones to be validated during the testing:

Table 14. Integration requirements related to UC06

Req. Identifier	Asset	Responsible partner	Description	Priority
R06.001	DER Management Module	TH OWL	The DER module MUST provide a value for the energy light based on the calculated Demand Response (DR)-Signal	5
R06.002	DER Management Module	TH OWL /ETRA	The DER module MUST communicate the DR-signal to the NATS server	5
R06.003	Energy Light	TH OWL /ETRAID	The Energy Light MUST get its control signal from the NATS server	5
R06.004	Energy Light	TH OWL	The Energy Light SHOULD refresh every 15 minutes	3
R06.005	H&T Module	STAM	The H&T module MUST get the latest DR-Signal from the NATS server to provide the consumer the proper information	4
R06.006	Demand Flexibility Module	IES	The Demand Flexibility module MUST get the Incentive Curve (ICC) from the NATS server to calculate flexibility	4

3.2.7 UC07 - Consumer's engagement in demand response programs utilizing a socio-economic context

This Use Case entails to enable a set of social context drivers for energy-related behaviour changes by exploiting social interaction and cultural values. The main objective is to validate a social comparison tool by comparing the energy use between several neighbouring end-users, focusing particularly on the individual's own behaviour which, to some extent, has implication on the consumption regular profile.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

The outcome will be a reasoned analysis into all kinds of barriers that customers and service providers face and how they can limit the implementation of TwinERGY business models.

Social engagement in this context is foreseen to take place via a gamification scheme based on social comparison among several end-users (i.e., prosumers and consumers). The increased awareness of end-users' profiles will be performed via comprehensive dashboards that will provide the monitoring of current and projected load and generation profiles. The social comparison will also lead social competition within the community or the neighbourhood providing rewards (i.e., TwinERGY points which can be redeemed in exchange of goods available in the transactive energy platform) to the participating households. The competition will be performed based on gamification functionalities which in turn will be relying on specific KPIs such as energy efficiency, participation on the local community, utilization of shared assets etc, as a matter of providing the corresponding rewards.

The whole UC07 is divided in the following sequence of the primary use cases (PUC):

- PUC07.01. "Social marketing to engage customers via competition"
- PUC07.02. "End-users' engagement on utilization of shared DERs"
- PUC07.03. "Enable co-creation for end consumers, service providers and public authorities".

These use cases consider the communication among the elements listed in next table:

Table 15. Assets actuating during UC07 processes

Component	Component Type
Social Network Module	TwinERGY application
Social Network Module GUI	TwinERGY application
Core Data Management Platform	TwinERGY application
Digital Twin Platform (Digital Twin Platform)	TwinERGY application
Comfort Well-being module	TwinERGY application
iSCAN	TwinERGY application
Transactive Energy Platform	TwinERGY application
Energy asset/appliance	Physical asset

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Asset Controller/Smart Meter	Physical asset
Raspberry-Pi	Physical asset
HEMS	External application
HEMS Gateway	External application

As the main goal of this phase is the test of the communication between elements, we consider these requirements the needed one to be validated during the testing:

Table 16. Integration requirements related to UC07

Req. Identifier	Asset	Responsible partner	Description	Priority
R07.001	iSCAN/ Digital Twin Platform	IES	iSCAN MUST form Digital Twin Platform optimized net profiles, DT model of community, disaggregated profiles of consumption.	5
R07.002	iSCAN	IES	iSCAN MUST publish disaggregated profiles, optimized net profiles, DT model of community	5
R07.003	Social Network Module GUI	ED	Social Network Module MUST retrieve disaggregated profiles, optimized net profiles, DT model of community (via Interoperability Platform),	5
R07.004	Social Network Module GUI	ED	Social Network Module MAY request for disaggregated profiles, optimized net profiles, DT model of community to the Digital Twin Platform through the iSCAN.	2
R07.005	Consumer Comfort/Well-being Module	UoP	Consumer Comfort/Well-being Module MUST publish at Interoperability Platform data analytics for individuals	5
R07.006	Energy asset/appliance	Pilot Leaders	Energy assets MUST establish successful interoperation with asset controller Raspberry Pi	5
R07.007	HEMS	Pilot Leaders	Asset Controllers/Pi's MUST send sensor status, actual energy consumption and production to HEMS	5
R07.008	HEMS gateway	Pilot Leaders	HEMS to HEMS Gateway MUST send to HEMS sensor status, actual consumption, production	5
R07.009	HEMS gateway	Pilot Leaders	HEMS to HEMS Gateway MUST publish to CDMP sensor status, actual consumption, production	5

R07.010	Social Network Module GUI	ED	Social Network Module GUI MUST request info for competitions badges/coins, Monitoring dashboards.	5
R07.011	Social Network Module - Transactive Energy Platform	ED-WEC	Social Network Module MUST exchange TwinERGY coins and goods with TEM	5
R07.012	Social Network Module	ED	Social Network Module MUST retrieve raw data (measurements for consumption, production) from CDMP	5
R07.013	Social Network Module	ED	Social Network Module MUST retrieve processed data directly from iSCAN	5

3.2.8 UC08 - Consumer's engagement in demand response programs utilizing personalized comfort/health-oriented services

Use Case 08 aims to show the feasibility of making optimal decisions of energy consumption allocation, while preserving acceptable Thermal Comfort level & Well-being Status with respect to the consumer's preferences.

The "Comfort & Well-being Module" processes a set of indoor environmental parameters (Indoor air temperature, Indoor relative humidity, and concentration of CO_2 and TVOCs), the individual's activity level and metabolic rate and clothing insulation. It aims to continuously assess the occupant's thermal comfort level and well-being status, which eventually is pushed to the Consumer Digital Twin (CDT). Moreover, the occupant inserts a set of preferences that reflect the relative priority and importance level over multiple criteria that affect occupant's energy behavior, namely the desired thermal comfort level and well-being status. Preferences can be updated at any time through the CDT's UI in a simple and user-friendly way.

As a summary, the Use Case 08 was subdivided in the next primary use cases, as was explained in deliverable D4.4 - "System Architecture":

- PUC08.01. "Well-being best practice for indoor environment conditions"
- PUC08.02. "Overall thermal comfort level and physiological parameters monitoring"
- PUC08.03. "Thermal comfort effect to demand response optimal solution

These use cases consider the communication among the elements listed in next table:

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Table 17. Assets actuating during UC08 processes

Component	Component Type
Physiological sensor	Physical asset
Environmental sensor	Physical asset
Wearable device	Physical asset
Consumer Comfort/Well-being Module	TwinERGY application
Consumer Digital Twin (CDT) platform	TwinERGY application
Smart Meter	Physical asset
Interoperability Platform	TwinERGY application
HEMS gateway	External application
HEMS	External Application
iSCAN	TwinERGY application
Consumer Demand Flexibility Profiling Module	TwinERGY application

As the main goal of this phase is the test of the communication between elements, we consider these requirements as the needed ones to be validated during the testing:

Table 18. Integration requirements related to UC08

Req. Identifier	Asset	Responsible partner	Description	Priority
R08.001	Physiological sensors	UoP	Physiological and activity readings MUST be aggregated, verified and transmitted to CDT platform	5
R08.002	Environmental sensors	UoP	Indoor environmental readings MUST be aggregated, verified and transmitted to CDT platform	5
R08.003	Comfort & Well-being application	UoP	Clothing insulation MUST be updated via the CDT platform	5
R08.004	Consumer Digital Twin platform	UoP	Thermal comfort and well-being status MUST be assessed	5

R08.005	Consumer Digital Twin platform	UoP	Preferences MUST be updated via the CDT platform	5

3.2.9 UC09 - Consumer's Engagement in Demand Response Programs Utilizing Digital Twin Prediction Capabilities for Dynamic VPPS

The main focus of consumer engagement in the demand response programs that utilise the digital twins of the TwinERGY pilot sites is a human-machine interface in the form of an online interactive dashboards. Bespoke dashboards will be created for each of the pilot sites, the design of which will be informed by their specific requirements. Dashboards will be created for both Consumer and Community Digital Twins, with specific users able to interact with and evaluate relevant information about their home/building/community. In terms of the quality of data that are displayed and the accuracy of the demand response programs, it is very important that the required information and data are provided by each of the pilot site. The quality of performance of this use case is directly proportional to the input data, and so the data collection exercise is a vital pre-requisite to this use case.

As a summary, the Use Case 09 was subdivided in the next primary use cases, as was explained in deliverable D4.4 - "System Architecture":

- PUC09.01: "Explicit Demand Response Automation and display at a consumer and community level", where the results of the explicit DR are shown to the user through the modules.
- PUC09.02: "Implicit Demand Response Calculation and Communication to the end-user at a consumer and community", where the results of the implicit DR are shown to the user through the modules.

These use cases consider the communication among the elements listed in next table:

Table 19. Assets actuating during UC09 processes

Component	Component Type
Core Data Management Platform	TwinERGY application
iSCAN	TwinERGY application
Consumer Demand Flexibility Profiling Module	TwinERGY application
Transactive Energy Market (TEM) Connected	TwinERGY application
Social Network Module Connected	TwinERGY application

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

As the main goal of this phase is the test of the communication between elements, we consider these requirements as the needed ones to be validated during the testing:

Table 20. Integration requirements related to UC09

Req. Identifier	Asset	Responsible partner	Description	Priority
R09.001	iSCAN	IES	iSCAN HAS TO create an iSCAN project when the user defines the building and data sources for the target building.	4
R09.002	iSCAN	IES	iSCAN HAS TO allow the user to connect the iSCAN Channels to the Data Sources	4
R09.003	iSCAN	IES	iSCAN HAS TO allow the Transactive Energy Module to be integrated in a dedicated channel	4
R09.004	iSCAN	IES	iSCAN HAS TO automatically initiate the algorithm to populate channels with up-to-date data.	4
R09.005	iSCAN	IES	iSCAN HAS TO optimize building and community demand profiles for cost and RES production based on user preferences.	4
R09.006	Social Network Module	IES/ED	The Social Network Module HAS TO display results of explicit and implicit demand response	4

3.2.10 Transactive Energy Blockchain Network

This section focuses on the Blockchain implementation of the Transactive Energy Module (TEM) lab test scenarios.

By definition, blockchain is a shared, immutable database that is shared among the nodes of a computer network, so the information is stored in a distributed way. That means that the main goal of blockchain networks is to allow digital information to be recorded and distributed but not edited, unlike traditional databases, where data can be created, edited and removed. In this way, a blockchain is the foundation for immutable records (ledgers) of transactions that cannot be altered, deleted, or destroyed. Therefore, blockchains are also known as a distributed ledger technology (DLT).

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Another main difference between a traditional database and blockchain is how the data is structured. While traditional databases store information in structured schemes called tables, blockchain collects information together in groups, known as blocks, that hold sets of information. Blocks have certain storage capacities and, when filled, are closed and linked to the previously filled block, forming a chain of data known as the blockchain. All new information that follows the freshly added block is compiled into a newly formed block that will then also be added to the chain once filled.

In TwinERGY Transactive Energy Market (TEM), blockchains cannot interact with data and systems outside their native Blockchain environment. Data outside the Blockchain is considered "off-chain" and data already stored on the Blockchain is considered "on-chain". Purposely isolated from external data and systems, the Blockchain obtains its most valuable properties. Interoperating with off-chain systems requires additional infrastructure known as "oracle". Oracles bridge the gap between the two environments.

Input oracles

An input oracle fetches data from the off-chain and delivers it onto a Blockchain network for smart contract consumption. The Transactive Energy Platform utilises different types of input oracles:

- Smart meter oracle: Delivers the meter readings of individual Smart meters to the Blockchain network
- LEM Price oracle: Delivers the calculated LEM price to the Blockchain network
- TEM oracle: Facilitates communication between the prosumers and the Blockchain network for trades, transactions, and rewards

Output oracles

Opposite of input oracles are "output oracles". These allow smart contracts to send commands to off-chain systems that trigger them to execute certain actions. The TEM utilises different types of output Oracles:

- Trade oracle: Notifies the TEM about the availability of new trade orders
- Transaction oracle: Instruct the IoT network to charge or discharge batteries when certain conditions are met
- DER oracle: Notifies the DER module to issue vouchers when prosumers have exchanged their reward tokens.

Blockchain consensus

The Transactive Energy Platform utilises a Proof of Authority (PoA) consensus mechanism. Using this consensus mechanism, the Transactive Energy Platform can keep

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

a tighter grip on both coin supply and coin trading and can ensure that all transactions occurring on the blockchain are genuine. In PoA the identities of validator nodes are publicly known, and thus it would be extremely detrimental to the validator to engage in fraudulent or malicious behaviour. They could be easily found, lose their reputation alongside the loss of validator status.

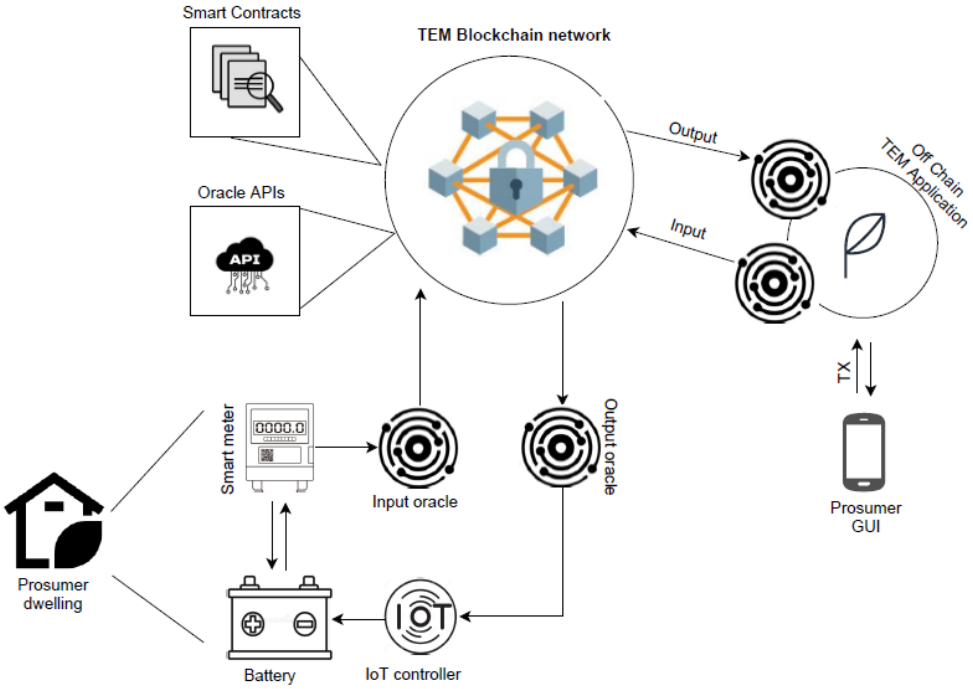

Figure 1. Workflow in Transactive Energy Blockchain network

The Transactive Energy Platform is participating in most of the Use cases of TwinERGY. However, the main use case that this platform is related is the UC04, previously described in section *3.2.4 UC04*. So, the requirements for this platform are the same as for UC04 (see *Table 10*).

3.2.11 Core Data Management Platform

As described in detail in the deliverables of WP5, the Core Data Management Platform (CDMP) is an "open", modular and interoperable platform, which enables open standards-based data collection and management communication across the TwinERGY project's energy value chain. The TwinERGY Core Big Data platform is made up of the following essential services, namely:

- Data Collection Service

- Data Security Service
- Data Storage Service
- Platform Management Service

It can be accessed at https://twinergy.s5labs.eu/

The conceptual architecture of the Core Data Management Platform is presented in Figure 2 below, and depicts, on top of the defined internal services, the available distributed systems of the pilots, embodying the source of the input data to be ingested into the platform, as well as the connection of the TwinERGY modules and Digital Twins via open APIs to the platform, to retrieve the necessary pilot's data for further processing.

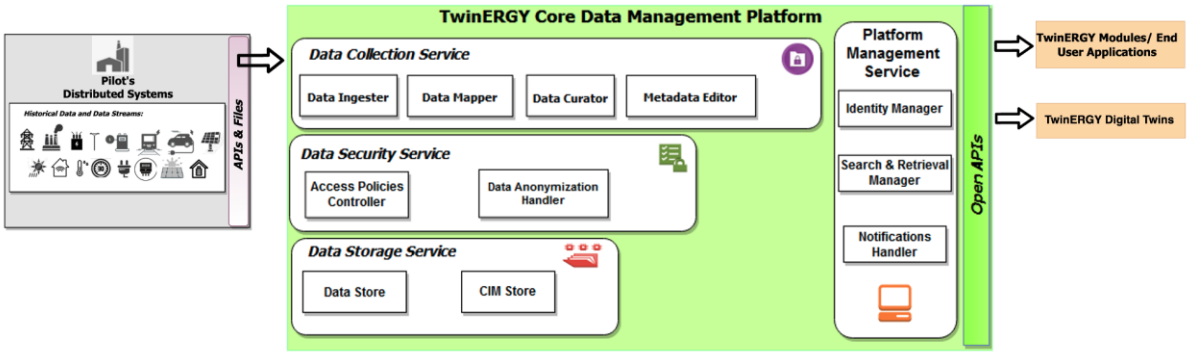

Figure 2: TwinERGY Core Data Management Platform Conceptual Architecture

Operation flows of the Core Data Management Platform

For the integration of the Core Data Management Platform with the Pilots, as well as the Digital Twins and the TwinERGY modules, as presented in WP6 and WP7 respectively, the following services and their respective functionalities are in focus, that offer the appropriate interfaces to realize the data ingestion of pilot's data and the functions for the retrieval of homogenized data from the platform:

- Data Collection Service – Data Ingester
- Platform Management Service – Search and Retrieval Manager

Data collection service – Data Ingester

The Data Ingester introduces proper procedures and methods for ingesting data into the CDMP. It offers the following functions:

- Data ingestion process configuration, that depicts the way that the data will be ingested into the TwinERGY Core Data Management Platform (file uploading, API).
- Ingestion of data from files, that allows data to be retrieved from files in popular file formats (e.g., csv, JSON).

- Ingestion of data via APIs, that 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, that enables data to be uploaded to the TwinERGY Core Data Management Platform using reliable and secure processes.

Platform Management Service – Search and Retrieval Manager

The Search and Retrieval Manager allows platform users to search and find data that may be proven valuable, as well as to define the necessary steps that allow the secure retrieval of that data via Open APIs. The search and retrieval Manager offers the following functions:

- Clear and comprehensive view of the available assets.
- Appropriate filters and free text search for the identification of assets that meet the user's criteria.
- Intuitive step by step guide to prepare the API for data retrieval.
- Single point of entry for apps (both TwinERGY modules and Digital Twins) to retrieve data from the CDMP that they are allowed to access.

Requirements of the CDMP

According to the previous description, the CDMP communicates with the elements listed in the following Table 21.Table 3

Table 21. Assets actuating with CDMP

Component	Component Type
Smart Devices and sensors at the pilot sites	Physical assets
Community Battery Storage	Physical asset
Inverter	Physical asset
Charging Point	Physical asset
MV-LV Transformer	Physical asset
TwinERGY system modules	TwinERGY applications
TwinERGY Digital Twin Platform	TwinERGY applications

The main goal of this phase is the testing of the communication between elements and, in this direction, the following requirement are deemed necessary (Table 22).

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Table 22. Integration requirements related to CDMP

Req. Identifier	Asset	Responsible partner	Description	Priority
RCDMP.001	Home Energy Management System gateway	All Pilots	Smart meters and sensors integrated MUST be able to send data to the HEMS of each pilot.	5
RCDMP.002	Home & Tertiary Real-time Energy Monitoring Module	STAM	Home & Tertiary Real-time Energy Monitoring Module MUST be able to send information from physical assets to the Core Data Management Platform	5
RCDMP.003	Consumer Comfort/Well-being Module	UoP	Consumer Comfort/Well-being Module MUST be able to deliver information from physical assets to the Core Data Management Platform	5
RCDMP.004	iSCAN	IES	iSCAN MUST be able to interchange information with Core Data Management Platform	5
RCDMP.005	RES integration & DER Management Module	TH OWL	RES Integration & DER Management Module MUST be able to retrieve information from Core Data Management Platform	5
RCDMP.006	Consumer Demand Flexibility Profiling Module	IES	Consumer Demand Flexibility Profiling Module MUST be able to retrieve information from Core Data Management Platform	5
RCDMP.007	Community battery storage gateway	All Pilots	The community battery storage gateway MUST be able send data to the CDMP	5
RCDMP.008	Grid gateway	All Pilots	The Grid gateway MUST be able to send data to the CDMP	5

RCDMP.009	TwinEV Module	ETRAID	TwinEV MUST be able to retrieve information from Core Data Management Platform	5
RCDMP.010	Transactive Energy Platform	WEC	Transactive Energy Platform MUST be able to retrieve information from Core Data Management Platform	5
RCDMP.011	Social Network Module	ED	Social Network Module MUST be able to retrieve information from Core Data Management Platform	5
RCDMP.012	Neighbourhood Demand Flexibility Profiling Module	IES	Neighbourhood Demand Flexibility Profiling Module MUST be able to retrieve information from Core Data Management Platform	5

3.2.12 & Digital Twin Platform

This section describes the requirements regarding the functionalities of iSCAN within the TwinERGY project. The main objective is to use the capabilities of the consumer and community Digital Twins methodology for calculating and profiling the potential flexibility at micro and macro level. The particular module is based on both physics-driven and data-driven modelling and simulation. Depending on the type of flexibility, the appropriate modelling tool was deployed to calculate the amount of flexibility and its controllability aspects.

DT components are intended to create further renewable sources and infrastructure, in order to increase the RES production, shared in public and private buildings. They are applied to the pilot sites for the TwinERGY project, with different aims for each specific to the location.

Considering the whole pilot site, the planification includes all requirements related to the whole community, so the RES production is considered as a whole.

The implementation of the use case is sub-divided into two different phases, namely Set-Up Phase and Optimisation Phase. The Set-Up phase is completed by the users and its results feed the Optimisation Phase. The methodology for each phase is outlined below.

Project Set Up – Completed by the User

- The user accesses the predefined JSON file and commences to provide information with respect to the building, its location and the relevant data sources that can be provided to the system.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

- This JSON contains the information allocated to a single building (each building requires its own dedicated JSON file).
- The user saves the JSON file and notes the file path/hyperlink that the file is stored.
- The user accesses the algorithm interface and provides the file path / hyper link to the JSON.
- The algorithm is initiated and reads the JSON file, setting up the iSCAN project for the building while creating the relevant channels as specified by the user in the JSON file.
- With the building level iSCAN project created, the algorithm sets up the community level iSCAN project.
- The community level project creates channels for aggregated demand for each building, total aggregated demand, optimised rescheduling of demand, cost analysis, renewable energy generation, flexibility assessment and end-user acceptance ratio.
- The algorithm creates a dedicated channel for Energy Tariff (from Transactive Energy Market) and the weather status for the location.
- At this point, the iSCAN project is set up. The user can insert additional JSON files for other buildings in the community.
- The user identifies the API associated with each channel within the iSCAN project.
- The user accesses the relevant data source (Core Data Management Platform, Transactive Energy Market, other) and provides the API for the channel to the data source.
- The user initiates a data transfer from the source to iSCAN and validates that data have been transferred.
- At this point, the Project Set Up is completed and the Optimisation Phase commences.

Optimization phase – Time Triggered

- At a user defined time (18:00 by default), the optimisation algorithm is automatically initiated and starts to run.
- The algorithm calls on the predefined JSON file from the project set-up phase and validates that the iSCAN project remains current.
- The algorithm calls on all data sources to push the most up-to-date data to the iSCAN project.
- The algorithm validates that the data are provided to the iSCAN channels.

- Where data are missing, the Building Digital Twin or the StRoBe library is called to fill in any data gaps and ensure a comprehensive data set is available for optimisation.
- The iSCAN channel is now populated with the most up to date data set.
- Building level time series data (from iSCAN) for the last 30 days is now accessed by the optimisation algorithm.
- Integrated python libraries identify the pattern of smart appliance use over the defined time period (30 days).
- The standard profile of energy use for the next day is calculated and the shiftable loads based on user preferences are defined. This is done for the three levels (consumer, building and community)
- The optimisation algorithm optimises the previous calculated demand profile for the next day based on shiftable loads, energy cost and RES production.
- The optimised demand profiles are processed over the specified time period and returned to the relevant iSCAN channel where it is available for others modules to use as required.

Requirements for iSCAN & Digital Twin Platform

Despite iSCAN & Digital Twin platforms appear in almost every use case, both components are highly related to the use case UC02 and UC09, so their requirements can be found in the description of these use cases (sections 3.2.2 and 3.2.9 above).

3.2.13 TwinERGY Identity Manager platform & Citizens platform

The TwinERGY Identity Manager platform (TwinERGY IdM platform) is a Keycloak-based server for users' authentication and managing access to modules belonging to WP6 and WP7, as well as to the protected resources accessed by these modules. In brief, Keycloak is a manager of access control based on Single Sign-On (SSO) [5] for web apps and RESTful web services. As such, Keycloak tries to create a trust relationship between an application (service provider) and an identity provider. This trust relationship is often based upon a certificate that is exchanged between the user and the authentication services. ....

More specifically, Keycloak is a server to which other applications point to and are secured by it. Browser applications redirect a user's browser from the application to the Keycloak authentication server where they enter their credentials. This redirection is important because users are completely isolated from applications and applications never see a user's credentials. Instead, applications are given an identity token that is

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

cryptographically signed. A token is a unique chain of text (an identifier) that represents sensitive data but does not contain any information of that sensitive data, so foreign applications cannot obtain information of the data by only using this token. The sensitive data can be username, address, email, and other profile data. They can also hold permission data so that applications can make authorization decisions. These tokens can also be used to make secure invocations on REST-based services.

Depending on the configuration, the communication between Keycloak and the clients asking for authentication takes place according to one of the two main supported SSO protocols: OpenID Connect and SAML, the first one being the preferred method.

Keycloak is based on the concept of realm, which manages a set of users, credentials, roles, and groups. A user belongs to and logs into a realm. Realms are isolated from one another and can only manage and authenticate the users that they control. Regarding user accessing, Keycloak includes two main functionalities: Single sign-on/out and access to protected resources.

Single Sign-On/Single Sign-Out

This feature allows users to:

1. Login the first time of access to one application/service and maintain that session for the rest of application/services.
2. Maintain the session and automatically renew it for all requests until a user express log out.
3. Logout once for an application, so the rest of applications will have the same session expired.

Of course, this is only possible if all application or services are accessed from the same device because they can store and share information about the open or expired session, managed by the Single-Sign on (SSO). Figure 3 shows the workflow executed.

A user that has not logged into the system yet makes a request via the graphical user interface (browser, mobile application, etc).

1. When the user makes a protected action through the interface, the associated backend the service/application checks in the shared memory if there is a valid session information stored.
   1.1. If there is no information about session or it is invalid/expired, the service or application asks to the interface to redirect to the Keycloak login page.
2. The interface redirects to Keycloak login.
3. The user introduces the credentials in the login page:

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

3.1. Keycloak validates the user credentials and, if they are valid, it generates a session code. If the Identity provider is another user, the validation will be done via the identity provider.
3.2. Keycloak asks the interface to redirect to the service/application with the session code.
3.3. The interface stores the session code in the shared device memory.

4. The interface tries the request in step 1:
4.1. The service/application checks in the shared memory if there is a valid session information stored.
4.2. If there is a valid session, the service/application executes the request.

5. When the user makes another protected action, the interface makes another request:
5.1. The service/application checks in the shared memory if there is a valid session information stored.
5.2. If there is a valid session, the service/application executes that new request.

6. The user opens another application or call another service, so the interface makes a request to a second service or to a second application.
6.1. The second service/application checks in the shared memory if there is a valid session information stored.
6.2. As there is a valid session, the second service/application executes the request. If the session has become invalid, the service/application, which received the request, repeats steps 1 to 3.3. Therefore, the responsible of asking for that renovation is the client application/service.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

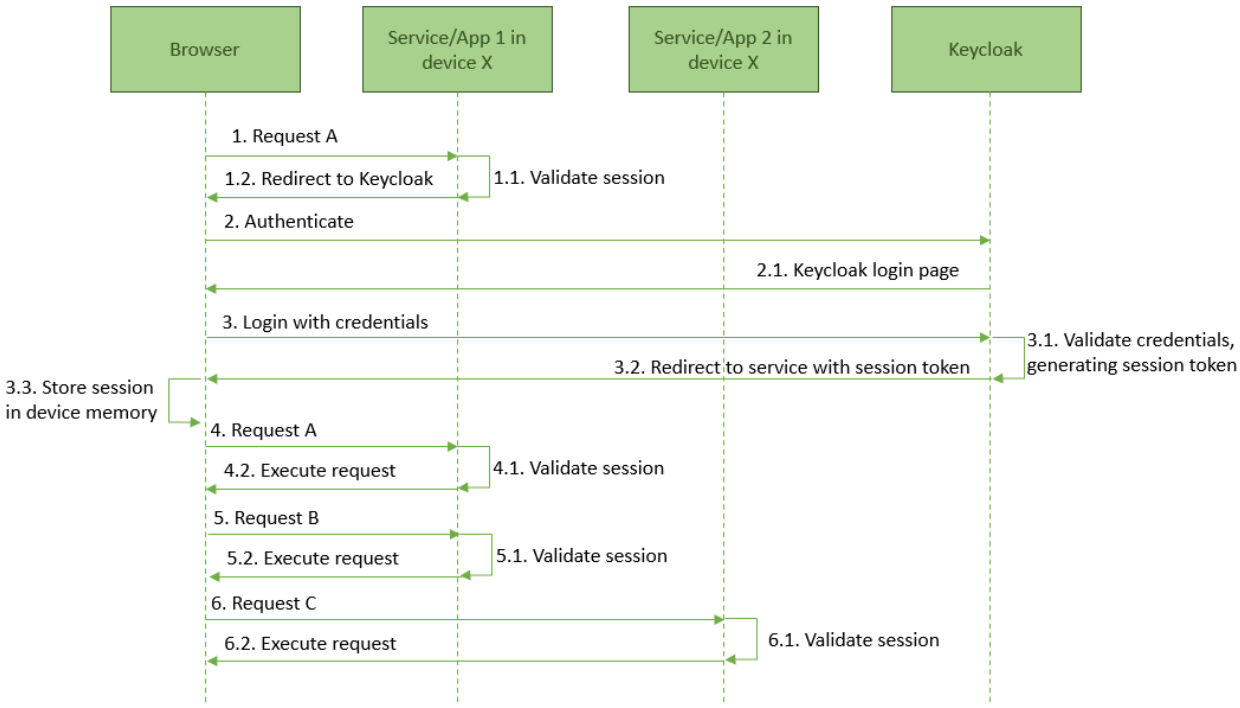

Figure 3. Example of workflow with Single Sign-On

Access to protected resources

The client applications must validate permissions for accessing protected resources via Keycloak. Figure 4 depicts the flow of actions that can be summarized as follows:

1. A user that has not logged into the system yet asks for a protected resource via the graphical interface (browser, mobile application, etc).
    1.1. The service/application checks that in the request there is no access code, so it generates one.
    1.2. The service/application asks to the interface to redirect to the Keycloak login page, with the access code.
2. The interface redirects to Keycloak login page with the access code.
3. The user introduces the credentials in the login page
    3.1. Keycloak validates the credentials and, if they are valid, it generates an authorization code. If the Identity provider is another one, the validation will be against the identity provider.
    3.2. Keycloak asks the interface to redirect to the service/application with the authorization code and the resource code.
4. The interface asks the application for the resource item, sending the authorization token and the resource code.
    4.1. The service asks Keycloak for validating the authorization code.

4.2. Keycloak validates the authorization code, generating an access token.

5. The service receives the confirmation that the resource is allowed to be accessed by the user and it is returned to them.

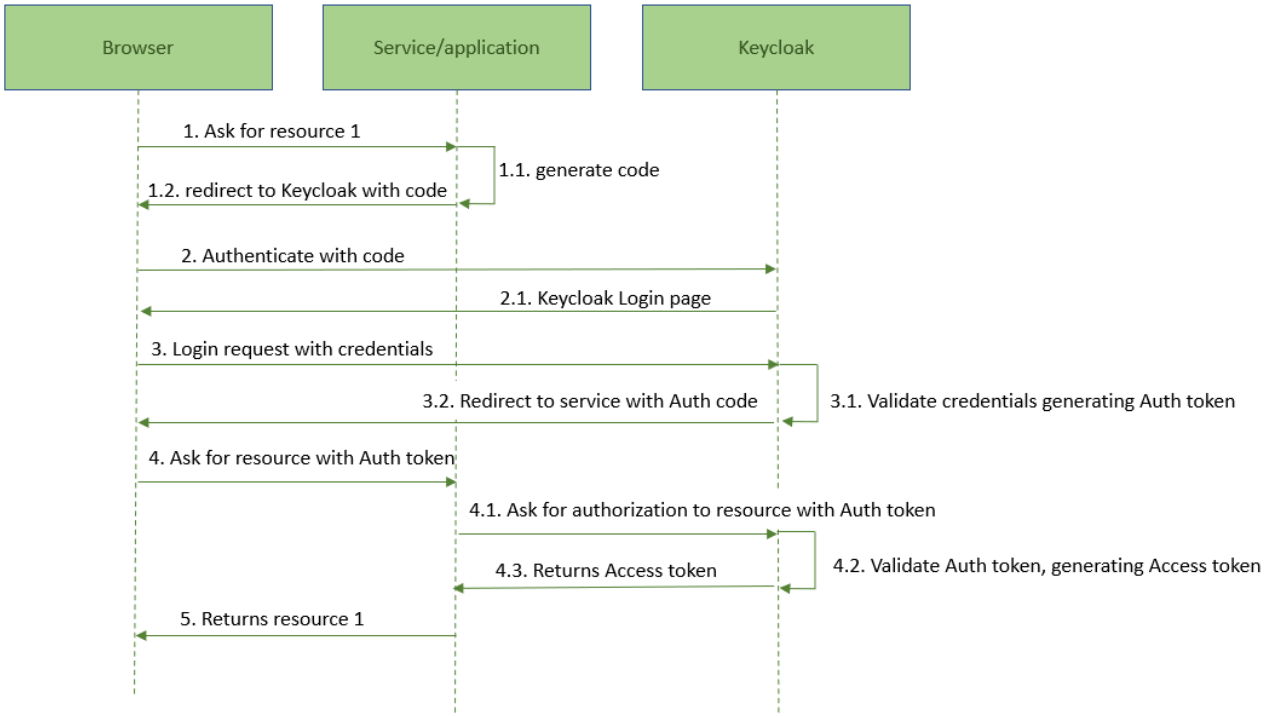

Figure 4. Keycloak operation phase

In case of users previously logged into the system, the steps 1 to 3.2 are omitted, because the browser asks for a resource with the authorization token.

TwinERGY Citizens Platform

Closely related to the previous component is the application where end-users register into TwinERGY Identity Management Platform, and maintain their related data used by the modules of WP7. Users can also fully remove their accounts, if they wish; in that case, they cannot further access any of the TwinERGY services.

Although this module was not defined during the system architecture design, as modules in WP7 have been developed and tested, the need of a common platform to maintain data related to users has appeared.

The platform is accessible in https://twinergy_citizens.tec.etra-id.com/, being deployed in ETRA premises. The information given by users is stored into a secured database, accessible by all modules.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

The registration process can be started by two ways:

1. The user goes to main page. The login page appears and, as user does not have an account, clicks on link "Register".
2. The user directly navigates to the register page (https://twinergy_citizens.tec.etra-id.com/register).

The registration page is a basic form including all fields required by the TwinERGY Identity Manager Platform in order to create an account: First name, last name, email, username, password and confirm password. After uploading that information, the account is temporarily created and the system sends a confirmation email to the address indicated. This email includes a code to be inserted in the registration page. After this confirmation, the account is created. If the account is not confirmed in 24 hours, it is deleted.

Once the account is created, the user is able to use the different services in TwinERGY. The platform also allows to fill extra information related to the shared data used by WP7 modules, such as: electric vehicles, charge points, storage units, etc.

Requirements for TwinERGY Identity Management Platform

As previously mentioned, the TwinERGY Identity Management platform & Citizens platform will communicate with all those components in TwinERGY which have any direct interaction with end users. This is summarized in next table. Please, consider that this is a first list which can be increased as real integration is finished.:

Table 23. Assets actuating with TwinERGY Identity Management platform

Component	Component Type
Consumer Comfort/Well-being module	TwinERGY application
Home & Tertiary real-time Energy Monitoring module	TwinERGY application
TwinEV module	TwinERGY application
Social Network module	TwinERGY application
Common platform for end-users shared data	TwinERGY application

As the main goal of this phase is the test of the communication between elements, we consider these requirements as the needed ones to be validated during the testing:

Table 24. Integration requirements related to TwinERGY Identity Management platform

Req. Identifier	Asset	Responsible partner	Description	Priority
RIdM.001	TwinERGY IdM platform	ETRAID	TwinERGY IdM platform HAS TO validate existing users	5
RIdM.002	TwinERGY IdM platform	ETRAID	TwinERGY IdM platform HAS TO reject unknown users or wrong credentials	5
RIdM.003	TwinERGY IdM platform	ETRAID	TwinERGY IdM platform HAS TO return user attributes with session token	5
RIdM.004	TwinERGY IdM platform	ETRAID	TwinERGY IdM platform HAS TO reject users in a Realm that try to connect from a client registered in another Realm	4
RIdM.005	Consumer Comfort/Well-being module	UoP	Consumer Comfort/Well-being module HAS TO validate an existing user through TwinERGY IdM platform	5
RIdM.006	Home & Tertiary real-time Energy Monitoring	STAM	Home & Tertiary real-time Energy Monitoring module HAS TO validate an existing user through TwinERGY IdM platform	5
RIdM.007	TwinEV module	ETRAID	TwinEV module HAS TO validate an existing user through TwinERGY IdM platform	5
RIdM.008	Social Network module	ED	Social Network module HAS TO validate an existing user through TwinERGY IdM platform	5
RIdM.009	Common platform for end-users shared data	ETRAID	Common platform for end-users shared data HAS TO validate an existing user through TwinERGY IdM platform	5

3.2.14 TwinERGY Interoperability Platform

As it was described in D7.1 "Modules' Interoperability" [1] , modules developed during WP7 conform a distributed ecosystem where all communicate with each other, being all of them highly interdependent. This means that every module needs information

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

generated by one or several others as well as it generate information that can be needed by the rest of the modules.

Moreover, the status of a module can affect the rest of them. For instance, if a module fails during its execution, the modules that are waiting for its inputs may hang. Thereforebe forced to stall their operation. Therefore, the waiting modules need to know this situation in order to follow a plan B: using previous data received from the module, making an estimation of what data would be received, or any other action. This point is important, so to ensure that the information is interchanged among the modules, since it is related to the tolerance to failures of the complete ecosystem. Besides, the modules need to receive the information promptly. It is of low importance is useless for modules to receive information with a large delay or corresponding to a previous time period. Therefore, all modules need to work with the latest data, corresponding to a real time schema. Such a situation implies that the WP7 needs a common platform of communication, where all modules can receive and send messages about their calculations and their status in a real time environment. This common channel of communication is the Interoperability Platform.

In summary, the Interoperability platform is composed by a NATS system, where WP7 modules connect to the server and send/receive messages to/from that server. All modules are connected through a client library, which allows to send messages, replay a request and subscribe to topics of messages for receiving what they need, among other functionalities. These functionalities are described below. As part of the work during Task T7.1, a web interface was developed, called "NATS console". This application shows the status of the NATS server, the list of connected clients and the list of subscriptions. Besides, it allows to send messages to the platform and load messages related to a topic. This application is the main communication (?) tool to be used during the integration phase.

Publish/subscribe

This feature is the basic way of communication in NATS, where a publisher sends a message with a subject and any active subscriber listening on that subject receives the message. Subscribers can also subscribe to messages by using regular expressions for the subject. For TwinERGY, this flow is used in almost all cases, where modules finalize their calculations and send the results to the rest of modules (Figure 5).

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

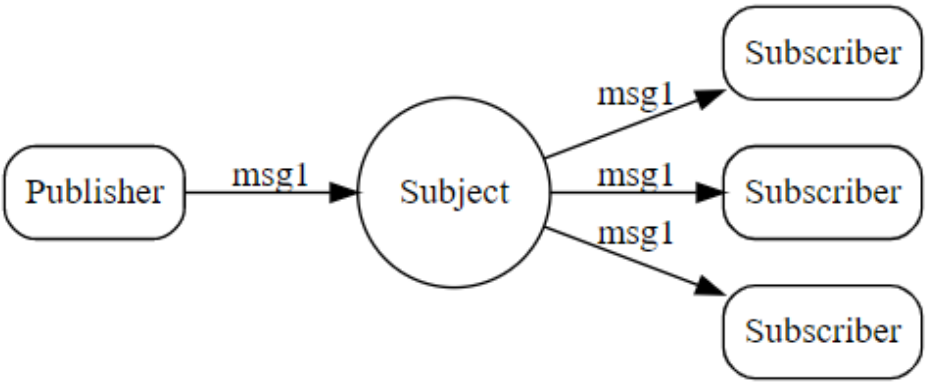

Figure 5. Publish/subscribe flow for NATS

Request/reply

Request-Reply is a common pattern in modern distributed systems. A request is sent and the application either waits on the response with a certain timeout or receives a response asynchronously (Figure 6).

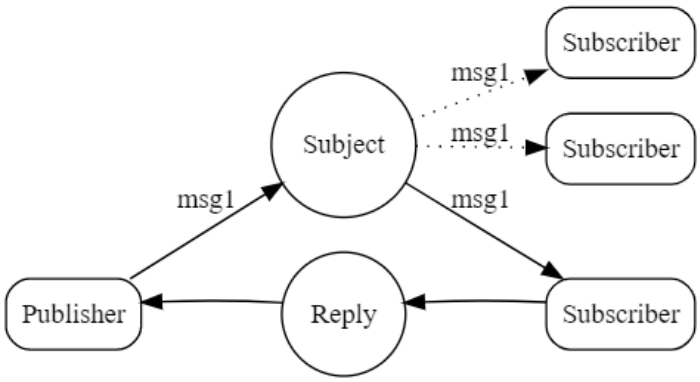

Figure 6. Request-reply flow

An example of request-reply is the following:

```
let nc = NATS.connect({url: "nats://demo.nats.io:4222"});

// set up a subscription to process the request
nc.subscribe('time', (msg, reply) => {
    if(reply) {
        nc.publish(reply, new Date().toLocaleTimeString());
    }
});

nc.requestOne('time', (msg) => {
    t.log('the time is', msg);
    nc.close();
});
```

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Requirements for TwinERGY Interoperability Platform

The TwinERGY Interoperability platform communicates with all modules belonging to WP7. This is summarized in next table:

Table 25. Assets actuating with TwinERGY Interoperability platform

Component	Component Type
Consumer Comfort/Well-being module	TwinERGY application
Consumer demand flexibility profiling module	TwinERGY application
Neighbourhood demand flexibility profiling module	TwinERGY application
Home & Tertiary real-time Energy Monitoring module	TwinERGY application
RES & DER management module	TwinERGY application
Risk Management & event handling module	TwinERGY application
TwinEV module	TwinERGY application
Social Network module	TwinERGY application
Transactive Energy module	TwinERGY application

As the main goal of this phase is the testing of the communication between elements, we consider the next requirements as the needed ones to be validated during the testing. Since the communication with this platform is tested in the sections for Use Cases, the requirements per each link of communication are omitted here.

Table 26. Integration requirements related to TwinERGY Interoperability platform

Req. Identifier	Asset	Responsible partner	Description	Priority
RIP.001	Interoperability Platform	ETRAID	Interoperability Platform HAS TO accept any client with recognized credentials	5
RIP.002	Interoperability Platform	ETRAID	Interoperability Platform HAS TO reject any client with wrong credentials	5
RIP.003	Interoperability Platform	ETRAID	Interoperability Platform HAS TO receive/deliver all messages sent through it	5

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

RIP.004	Interoperability Platform	ETRAID	Interoperability Platform HAS TO send all messages to clients that are subscribed to associated subjects	5

3.3 Definition of validation requirements

In the previous step, partners have defined the requirements representing the characteristics related to the integration that need to be tested. In this step, partners classify the requirements in some categories, that can be exposed in the next table:

Table 27. Requirements classification criteria

Test group	Common criteria
Visualisation and analysis	The feature under test provides visualization and/or analysis of the data collected.
Control	The feature under test provides control of assets.
Compliance	The feature under test relates to compliance of the tool with the Universal Smart Energy Framework (USEF) standard [6] or other standards.
Functionality	The feature under test is a complex function provided by a combination of software and communication between multiple TwinERGY subsystems.
Communication	The feature under test is basic data transmission between two communication endpoints, one being the tool.
Robustness and stability	The feature under test is related to fault tolerance and stability.
Cyber Security	The feature under test mitigates vulnerabilities of the software or malicious attacks aimed at it.

Hereafter, Table 28 summarized the classification of all requirements presented though subsections in chapter 3.2 in types indicated in Table 27.

Table 28. Requirements classification

Req. Identifier	Description	Test group

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R01.001	Energy and environmental sensors HAVE TO send data to Home & Tertiary Energy Management Module	Communication
R01.002	Home & Tertiary Energy Management Module web application HAS TO show the logged user its appliances' telemetries data referring to the last 7 days	Communication
R01.003	Home & Tertiary Energy Management Module web application HAS TO show the logged user eventual anomalies that have been detected related to its appliances	Communication
R01.004	Home & Tertiary Energy Management Module web application HAS TO let the logged user browse its building possible threats, showing them also potential countermeasures and financial impacts	Communication
R01.005	Home & Tertiary Energy Management Module HAS TO let the logged user control its appliances remotely through the use of input sliders	Communication
R01.006	The monitoring data MUST be available in the CDMP	Communication
R01.007	iSCAN MUST share the asset building configuration in a JSON file with an API REST	Communication
R01.008	Home & Tertiary Energy Management Module MUST be able to read • Energy demand • Energy generation • Voltages • Energy appliances demand from the DCMP.	Visualization and Analysis
R01.009	The fault anomaly detection service in Home & Tertiary Energy Management Module MUST be able to read • Energy demand • Energy generation • Indoor temperature from the DCMP.	Visualization and Analysis
R02.001	iSCAN HAS TO allow users to define the building and data sources for the target building when automatically creating the iSCAN project.	Control

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R02.002	iSCAN HAS TO allow users to connect channels to the Data Sources	Communication
R02.003	iSCAN HAS TO allow to the Transactive Energy Platform to be integrated in a dedicated channel	Communication
R02.004	iSCAN HAS TO automatically initiate the algorithm to populate channels with up-to-date data	Communication
R02.005	Consumer Demand Flexibility Profiling Module MUST optimize building and community demand profiles for cost and RES production based on user preferences in iSCAN	Communication
R03.001	The CP MUST be locked when TwinEV module sends a lock order to the CPO	Control
R03.002	The CP MUST be unlocked when TwinEV module sends an unlock order to the CPO	Control
R03.003	A locked point MUST NOT be reserved when it is locked	Robustness and stability
R03.004	RES Integration & DER Management MUST send the RES production forecast after its calculations	Communication
R03.005	Neighbourhood demand flexibility profiling module MUST send the neighbourhood flexibility profile to Interoperability platform after its calculations	Communication
R03.006	RES Integration & DER Management MUST send grid status after loading them from the grid	Communication
R03.007	Consumer Demand Flexibility Profiling must send User appliances flexibility to Interoperability platform after its calculations	Communication
R03.008	Transactive Energy Platform must send LEM pricing to Interoperability platform after its calculations	Communication
R03.009	For smart charging, TwinEV module MUST be able to read inputs • RES production forecast, • Neighbourhood flexibility profile, • User appliances flexibility • LEM pricing	Communication

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	from Interoperability platform.	
R03.010	For both smart chargers to follow grid request and maximize RES integration, TwinEV module MUST read inputs Actual consumption and production from Core Data Management Platform	Communication
R03.011	TwinEV module MUST send charge curve to CPO before starting a charge session	Communication
R03.012	TwinEV module MUST be able to read grid status from Interoperability platform	Communication
R03.013	When a grid operator set a restriction to charge points, TwinEV module MUST send minimal and maximal power to the affected charging points	Control
R03.014	When a charging session has finished, TwinEV module MUST send Charging session cost to Interoperability platform	Communication
R03.015	Transactive Energy Platform HAS TO be able to load charging session cost from the Interoperability platform	Communication
R04.001	Physiological smart meters HAVE TO MAKE activity readings and transmit to the Raspberry Pi Oracle application (RPOA) and cDT platform	Communication
R04.002	RPOA HAVE TO store and transmit activity readings to the Transactive Energy Blockchain network respective smart contract.	Communication
R04.003	RPOA data HAS TO update the Smart meter data smart Contract and POST on the Blockchain ledger.	Control
R04.004	Storage of Energy or Usage of stored energy MUST mint, burn or transfer kWh tokens	Control
R04.005	RPOA data HAS TO be settled on a set interval to reflect accurate energy generation, usage and storage	Control
R04.006	Transactive Energy Platform HAS TO be able to submit sell orders to the P2P sale logic smart contract.	Control
R04.007	When a sale is completed, Transactive Energy Platform HAS TO instruct the HEMS to schedule an energy transfer between Smart Batteries.	Communication

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R04.008	When the transfer of energy is completed, the P2P sale logic smart contract HAS TO transfer kWh and TwinERGY tokens from and to the buyer and seller respectively.	Communication
R04.009	Based on a synchronization of DER reward with the Social Network Module, Transactive Energy Platform MUST instruct the Smart Contract to mint and distribute DER tokens	Control
R04.010	Transactive Energy Platform HAS TO transmit the LEM price to the Smart Contract and store the data on the ledger	Communication
R05.001	The RES Integration & DER Management Module MUST get the PV and wind-power configuration of the respective pilot-site	Communication
R05.002	The RES Integration & DER Management Module MUST get latest weather forecasts for the next 24 hours	Communication
R05.003	The RES Integration & DER Management Module MUST calculate the proper Incentive Curve (ICC) for the given timeframe	Functionality
R05.004	The RES Integration & DER Management Module MUST send the ICC to the NATS server	Communication
R06.001	The DER module MUST provide a value for the energy light based on the calculated DR-Signal	Communication
R06.002	The DER module MUST communicate the DR-signal to the NATS server	Communication
R06.003	The Energy Light MUST get its control signal from the NATS server	Communication
R06.004	The Energy Light SHOULD refresh every 15 minutes	Robustness and stability
R06.005	The H&T module MUST get the latest DR-Signal from the NATS server to give the consumer proper information	Communication
R06.006	The Demand Flexibility module MUST get the ICC from the NATS server to calculate flexibility	Communication
R07.001	iSCAN MUST from Digital Twin Platform optimized net profiles, DT model of community, disaggregated profiles.	Communication

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R07.002	iSCAN MUST publish disaggregated profiles, optimized net profiles, DT model of community	Communication & Visualisation and analysis
R07.003	Social Network Module MUST retrieve (via Interoperability Platform) disaggregated profiles, optimized net profiles, DT model of community,	Communication
R07.004	Social Network Module MAY request for disaggregated profiles, optimized net profiles, DT model of community,	Communication
R07.005	Consumer Comfort/Well-being Module MUST publish at Interoperability Platform data analytics for individuals	Communication
R07.006	Successful MUST establish successful interoperation with asset controller /Pi	Communication
R07.007	Asset Controllers/Pi's MUST send to HEMS sensor status, actual consumption, production	Communication
R07.008	HEMS to HEMS Gateway MUST send to HEMS sensor status, actual consumption, production	Communication
R07.009	HEMS to HEMS Gateway MUST publish to CDMP sensor status, actual consumption, production	Communication
R07.010	Social Network Module GUI MUST request info for competitions badges/Coins, Monitoring dashboards.	Visualisation and analysis
R07.011	Social Network Module MUST exchange TwinERGY coins and goods with TEM	Communication
R07.012	Social Network Module MUST retrieve raw data (measurements for consumption, production) from CDMP	Communication
R07.013	Social Network Module MUST retrieve processed data directly from iSCAN	Communication
R08.001	Physiological and activity readings MUST be aggregated, verified and transmitted to CDT platform	Communication, Functionality
R08.002	Indoor Environmental readings MUST be aggregated, verified and transmitted to CDT platform	Communication, Functionality
R08.003	Clothing insulation MUST be updated via the CDT platform	Visualization and analysis

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R08.004	Thermal comfort and well-being status MUST be assessed	Visualization and analysis
R08.005	Preferences MUST be updated via the CDT platform	Visualization and analysis, Functionality
R09.001	iSCAN HAS TO create an iSCAN project when user defines the building and data sources for the target building.	Control
R09.002	iSCAN HAS TO allow user to connect the iSCAN Channels to the Data Sources	Communication
R09.003	iSCAN HAS TO allow Transactive Energy Module to be integrated in a dedicated channel	Communication
R09.004	iSCAN HAS TO initiate automatically the algorithm to populate channels with up-to-date data.	Communication
R09.005	iSCAN HAS TO optimize building and community demand profiles for cost and RES production based on user preferences.	Communication
R09.006	Social Network Module HAS TO display results of explicit and implicit demand response	Communication
RCDMP.001	Smart meters and sensors integrated MUST send data to the HEMS of each pilot.	Functionality
RCDMP.002	Home & Tertiary Real-time Energy Monitoring Module MUST send information from physical assets to the Core Data Management Platform	Functionality
RCDMP.003	Consumer Comfort/Well-being Module MUST be able to retrieve information from physical assets to the Core Data Management Platform	Functionality
RCDMP.004	iSCAN MUST be able to interchange information with Core Data Management Platform	Functionality
RCDMP.005	RES Integration & DER Management Module MUST be able to read information from Core Data Management Platform	Functionality
RCDMP.006	Consumer Demand Flexibility Profiling Module MUST be able to read information from Core Data Management Platform	Functionality

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

RCDMP.007	The community battery storage gateway MUST send data to the CDMP	Functionality
RCDMP.008	The Grid gateway MUST send data to the CDMP	Functionality
RCDMP.009	TwinEV MUST be able to read information from Core Data Management Platform	Functionality
RCDMP.010	Transactive Energy Platform MUST be able to read information from Core Data Management Platform	Functionality
RCDMP.011	Social Network Module MUST be able to read information from Core Data Management Platform	Functionality
RCDMP.012	Neighbourhood Demand Flexibility Profiling Module MUST be able to read information from Core Data Management Platform	Functionality
RIdM.001	TwinERGY IdM platform HAS TO validate existing users	Functionality
RIdM.002	TwinERGY IdM platform HAS TO reject unknown users or wrong credentials	Functionality
RIdM.003	TwinERGY IdM platform HAS TO return user attributes with session token	Functionality
RIdM.004	TwinERGY IdM platform HAS TO reject users in a Realm that try to connect from a client registered in another Realm	Functionality
RIdM.005	Consumer Comfort/Well-being module HAS TO validate an existing user through TwinERGY IdM platform	Functionality
RIdM.006	Home & Tertiary real-time Energy Monitoring module HAS TO validate an existing user through TwinERGY IdM platform	Functionality
RIdM.007	TwinEV module HAS TO validate an existing user through TwinERGY IdM platform	Functionality
RIdM.008	Social Network module HAS TO validate an existing user through TwinERGY IdM platform	Functionality
RIdM.009	Common platform for end-users shared data HAS TO validate an existing user through TwinERGY IdM platform	Functionality

RIP.001	Interoperability Platform HAS TO accept any client with recognized credentials	Control
RIP.002	Interoperability Platform HAS TO reject to any client with wrong credentials	Control
RIP.003	Interoperability Platform HAS TO receive all messages sent through it	Control
RIP.004	Interoperability Platform HAS TO send all messages to clients that are subscribed to associated subjects	Control

3.4 Test Case specification

Once the requirements have been classified, it is needed to create a set of test cases to probe these requirements. Despite the fact that each test case defines the way of validation of a requirement, two points need to further be clarified. Firstly, depending on the complexity of the requirement to be probed, one can define more than one test case for the same requirement. Secondly, although the most common situation is that a test case is related to a single requirement only, it is worth to mention that if one requirement is dependent of another one, the same test case can be established for both related test cases.

The test cases are defined by tables with the format of *Table 29*. Several pieces of information are included in these tables, and among them are: how to execute the test case, the dependencies with other test cases, the steps to follow, the expected results, and the criteria to indicate the success or failure of the test case.

Table 29. Test cases template

Name	*The test case code and name, which are unique to the TwinERGY project.*		
Module under test	*The devices or systems under test*	**Resp.**	*The main partner responsible for the test*
Module requirement	*The requirement, use case, or certification rule which is validated by the test case*		
Test environment	*The list of elements needed for the test execution*		

Features to be tested	*The list of features in software to be tested*
Features not to be tested	*Optional*
Preparation	*Short list of steps needed for preparing the test environment for test execution*
Dependencies	*(Optional) List of test case codes defining test cases which need to be passed before the test case at hand can be started*
Steps	*Testing procedures*
Pass criteria	*Expected (measurable) results, allowing to unambiguously judge if the test is passed or not passed (i.e. the product requirement was validated or not validated)*
Suspension criteria	*(Optional) Conditions under which continuation of the test is considered pointless because testing results would be invalid*
Results	*(Optional) Short list of results (system status after the execution or action done)*

In the next two tables (Table 30 and 31), we summarized the test cases defined to probe each requirement. The complete list of the test cases can be found in Annex 1. Complete description of Test Cases, where each test case is detailly described by filling the previous Table 29.

Table 30. Test cases for requirements in use cases

Req. Identifier	Requirement description	Test cases
R01.001	Energy and environmental sensors HAVE TO send data to Home & Tertiary Energy Management Module	TC01.001 - Retrieve telemetry data from sensors (through MQTT channel)

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

		TC01.002 - Retrieve telemetry data from sensors (though HTTP request)
R01.002	Home & Tertiary Energy Management Module web application HAS TO show the logged user its appliances' telemetries data referring to the last 7 days	TC01.003 - View monitored energy data last time series
R01.003	Home & Tertiary Energy Management Module web application HAS TO show the logged user eventual anomalies that have been detected related to its appliances	TC01.004 - Consult list of anomalies
R01.004	Home & Tertiary Energy Management Module web application HAS TO let the logged user browse its building possible threats, showing him also countermeasures and financial impacts	TC01.005 - Browse building risk evaluations
R01.005	Home & Tertiary Energy Management Module HAS TO let the logged user control its appliances remotely through the use of input sliders	TC01.006 - Control user assets remotely
R01.006	The monitoring data MUST be available in the CDMP	TC01.007. Custom query build on DCMP for Benetutti pilot
R01.007	iScan MUST share the asset building configuration in a json file with an API REST	TC01.008. API for retrieving the digital twin information from the interoperability platform
R01.008	Home & Tertiary Energy Management Module MUST be able to read • Energy demand • Energy generation • Voltages • Energy appliances demand from the DCMP.	TC01.003 - View monitored energy data last time series
R01.009	The fault anomaly detection service in Home & Tertiary Energy Management Module MUST be able to read • Energy demand • Energy generation • Indoor temperature from the DCMP.	TC01.004 - Consult list of anomalies
R02.001	iSCAN HAS TO allow users to define the building and data sources for the target building when automatically creating the iSCAN project.	TC02.001. Creation of the iSCAN Projects

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R02.002	iSCAN HAS TO allow users to connect channels to the Data Sources	TC02.002. Data Channels linked to Data Management Platform
R02.003	iSCAN HAS TO allow to The Transactive Energy Platform to be integrated in a dedicated channel	TC02.003. Transactive Energy Market (TEM) Connected
R02.004	iSCAN HAS TO initiate automatically the algorithm to populate channels with up-to-date data.	TC02.004. Automated Algorithm Initiation & Data Collection
R02.005	Consumer Demand Flexibility Profiling Module MUST optimize building and community demand profiles for cost and RES production based on user preferences in iSCAN	TC02.005. Demand Profile Optimisation
R03.001	The CP MUST be locked when TwinEV module sends a lock order to the CPO	TC03.001. Reserve a charge point
R03.002	The CP MUST be unlocked when TwinEV module sends an unlock order to the CPO	TC03.002. Cancel a reservation
R03.003	A locked point MUST NOT be reserved when it is locked	TC03.003. Reservation of available charging points (1) TC03.004. Reservation of available charging points (2)
R03.004	RES Integration & DER Management MUST send the RES production forecast after its calculations	TC03.005. Sends RES production forecasting
R03.005	Neighbourhood demand flexibility profiling module MUST send the neighbourhood flexibility profile to Interoperability platform after its calculations	TC03.006. Send neighbourhood flexibility profile
R03.006	RES Integration & DER Management MUST send grid status after loading them from the grid	TC03.007. Send Grid status
R03.007	Consumer Demand Flexibility Profiling must send User appliances flexibility to Interoperability platform after its calculations	TC03.008. Send user appliances flexibility
R03.008	Transactive Energy Platform must send LEM pricing to Interoperability platform after its calculations	TC03.009. Send LEM pricing
R03.009	For smart charging, TwinEV module MUST be able to read inputs • RES production forecast, • Neighbourhood flexibility profile, • User appliances flexibility	TC03.010. Load RES production forecast

	• LEM pricing from Interoperability platform.	TC03.011. Load neighbourhood flexibility profile TC03.012. Load user appliances flexibility TC03.013. Load LEM pricing
R03.010	For both smart chargers to follow grid request and maximize RES integration, TwinEV module MUST read inputs Actual consumption and production from Core Data Management Platform	TC03.0014. Load actual consumption / production
R03.011	TwinEV module MUST send charge curve to CPO before starting a charge session	TC03.0015. Sending of charge curve
R03.012	TwinEV module MUST be able to read grid status from Interoperability platform	TC03.016. Load grid status
R03.013	When a grid operator set a restriction to charge points, TwinEV module MUST send minimal and maximal power to the affected charging points	TC03.0017. Send minimal and maximal power to charge points
R03.014	When a charging session has finished, TwinEV module MUST send Charging session cost to Interoperability platform	TC03.0018. Send charging session costs
R03.015	Transactive Energy Platform HAS TO be able to load charging session cost from the Interoperability platform	TC03.0019. Load charging session costs
R04.001	Physiological smart meters HAVE TO MAKE activity readings and transmit to the Raspberry Pi Oracle application (RPOA) and cDT platform	TC04.001. Check readings availability
R04.002	RPOA HAVE TO store and transmit activity readings to the Transactive Energy Blockchain network respective smart contract.	TC04.002. Transmitting meter data to the Transactive Energy Blockchain platform.
R04.003	RPOA data HAS TO update the Smart meter data smart Contract and POST on the Blockchain ledger.	TC04.003. Storage of energy or consumption of stored energy
R04.004	Storage of Energy or Usage of stored energy MUST mint, burn or transfer kWh tokens	
R04.005	RPOA data HAS TO be settled on a set interval to reflect accurate energy generation, usage and storage	TC04.004. Settled of RPOA
R04.006	Transactive Energy Platform HAS TO be able to submit sell orders to the P2P sale logic smart contract.	TC04.005. Submission of sell orders

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R04.007	When a sale is completed, Transactive Energy Platform HAS TO instruct the HEMS to schedule an energy transfer between Smart Batteries.	
R04.008	When the transfer of energy is completed, the P2P sale logic smart contract HAS TO transfer kWh and TwinERGY tokens from and to the buyer and seller respectively.	
R04.009	Based on a synchronization of DER reward with the Social Network Module, Transactive Energy Platform MUST instruct the Smart Contract to mint and distribute DER tokens	TC04.006. Instruction on the Smart Contract to mint and distribute DER tokens
R04.010	Transactive Energy Platform HAS TO transmit the LEM price to the Smart Contract and store the data on the ledger	TC04.007. Transmission of LEM price to the LEM price – logic
R05.001	The RES Integration & DER Management Module MUST get the PV and wind-power configuration of the respective pilot-site	TC05.001 Initialise the module
R05.002	The RES Integration & DER Management Module MUST get latest weather forecasts for the next 24 hours	TC05.002 Download weather information
R05.003	The RES Integration & DER Management Module MUST calculate the proper Incentive Curve (ICC) for the given timeframe	TC05.003 Test the ICC implementation
R05.004	The RES Integration & DER Management Module MUST send the ICC to the NATS server	TC05.004 Test the NATS communication
R06.001	The DER module MUST provide a value for the energy light based on the calculated DR-Signal	TC06.001. Test DR and EL (Energy Light) signal calculation
R06.002	The DER module MUST communicate the DR-signal to the NATS server	TC06.002. Send DR signal via NATS
R06.003	The Energy Light MUST get its control signal from the NATS server	TC06.003. Access EL-signal via NATS
R06.004	The Energy Light SHOULD refresh every 15 minutes	TC06.004. Test refreshing of the EL-signal
R07.001	iSCAN MUST from Digital Twin Platform optimized net profiles, DT model of community, disaggregated profiles.	TC07.001. iSCAN retrieving data from Digital Twin Platform

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R07.002	iSCAN MUST publish disaggregated profiles, optimized net profiles, DT model of community	TC07.002. iSCAN publishing dataset through NATS
R07.003	Social Network Module MUST retrieves (via Interoperability Platform) disaggregated profiles, optimized net profiles, DT model of community,	TC07.003. Social Network Module subscribes (processed data) information profiles from NATS
R07.004	Social Network Module MAY request for disaggregated profiles, optimized net profiles, DT model of community,	TC07.004. Social Network Module retrieves (processed data) information profiles from NATS
R07.005	Consumer Comfort/Well-being Module MUST publish at Interoperability Platform data analytics for individuals	TC07.005. Comfort well-being publishes at NATS data analytics for individuals
R07.006	Successful MUST establish successful interoperation with asset controller /Pi	TC07.006. Interoperation of assets with controllers
R07.007	Asset Controllers/Pi's MUST send to HEMS sensor status, actual consumption, production	TC07.007. Asset Controllers/Pi's assumes connectivity with HEMS
R07.008	HEMS to HEMS Gateway MUST send to HEMS sensor status, actual consumption, production	TC07.008. HEMS to HEMS Gateway connectivity
R07.009	HEMS to HEMS Gateway MUST publish to CDMP sensor status, actual consumption, production	TC07.009. HEMS Gateway publishes to CDMP
R07.010	Social Network Module GUI MUST request info for competitions badges/Coins, Monitoring dashboards.	TC07.010. SOCIAL NETWORK MODULE GUI requests for competitions badges/coins, and monitoring features.
R07.011	Social Network Module MUST exchange TwinERGY coins and goods with TEM	TC07.011. Sync and preview information in regard to the balance of the consumers digital wallet and the TwinERGY point earned
R07.012	Social Network Module MUST retrieve raw data (measurements for consumption, production) from CDMP	TC07.012. SOCIAL NETWORK MODULE retrieving raw data from CDMP
R07.013	Social Network Module MUST retrieve processed data directly from iSCAN	TC07.013. SOCIAL NETWORK MODULE retrieving data from iSCAN

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

R08.001	Physiological and activity readings MUST be aggregated, verified and transmitted to CDT platform	TC08.001 check readings availability
R08.002	Indoor Environmental readings MUST be aggregated, verified and transmitted to CDT platform	TC08.002 check readings availability
R08.003	Clothing insulation MUST be updated via the CDT platform	TC08.003 check input availability
R08.004	Thermal comfort and well-being status MUST be assessed	TC08.004 Check execution of algorithms
R08.005	Preferences MUST be updated via the CDT platform	TC08.005 Check execution of algorithms
R09.001	iSCAN HAS TO create an iSCAN project when user defines the building and data sources for the target building.	TC09.001: Creation of the iSCAN Projects
R09.002	iSCAN HAS TO allow user to connect the iSCAN Channels to the Data Sources	TC09.002: Data Channels linked to Core Data Management Platform
R09.003	iSCAN HAS TO allow Transactive Energy Module to be integrated in a dedicated channel	TC09.003: Transactive Energy Market (TEM) Connected
R09.004	iSCAN HAS TO initiate automatically the algorithm to populate channels with up-to-date data.	TC09.004: Automated Algorithm Initiation & Data Collection
R09.005	iSCAN HAS TO optimize building and community demand profiles for cost and RES production based on user preferences.	TC09.005: Demand Profile Optimisation
R09.006	Social Network Module HAS TO display results of explicit and implicit demand response	TC09.006: Explicit & Implicit Demand Response Results display

Table 31. Test cases for requirements in common assets

Req. Identifier	Requirement description	Test cases
RCDMP.001	Smart meters and sensors integrated MUST send data to the HEMS of each pilot.	TCCDMP01. File ingestion TCCDMP02. Ingestion via data provider's available API

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

RCDMP.002	Home & Tertiary Real-time Energy Monitoring Module MUST send information from physical assets to the Core Data Management Platform	TCCDMP03. Retrieve data via APIs
RCDMP.003	Consumer Comfort/Well-being Module MUST be able to retrieve information from physical assets to the Core Data Management Platform	TCCDMP03. Retrieve data via APIs
RCDMP.004	iSCAN MUST be able to interchange information with Core Data Management Platform	TCCDMP01. File ingestion TCCDMP02. Ingestion via data provider's available API TCCDMP03. Retrieve data via APIs
RCDMP.005	RES Integration & DER Management Module MUST be able to read information from Core Data Management Platform	TCCDMP03. Retrieve data via APIs
RCDMP.006	Consumer Demand Flexibility Profiling Module MUST be able to read information from Core Data Management Platform	TCCDMP03. Retrieve data via APIs
RCDMP.007	The community battery storage gateway MUST send data to the CDMP	TCCDMP01. File ingestion TCCDMP02. Ingestion via data provider's available API
RCDMP.008	The Grid gateway MUST send data to the CDMP	TCCDMP01. File ingestion TCCDMP02. Ingestion via data provider's available API
RCDMP.009	TwinEV MUST be able to read information from Core Data Management Platform	TCCDMP01. File ingestion TCCDMP02. Ingestion via data provider's available API TCCDMP03. Retrieve data via APIs
RCDMP.010	Transactive Energy Platform MUST be able to read information from Core Data Management Platform	TCCDMP03. Retrieve data via APIs

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

RCDMP.011	Social Network Module MUST be able to read information from Core Data Management Platform	TCCDMP03. Retrieve data via APIs
RCDMP.012	Neighbourhood Demand Flexibility Profiling Module MUST be able to read information from Core Data Management Platform	TCCDMP03. Retrieve data via APIs
RIdM.001	TwinERGY IdM platform HAS TO validate existing users	TCIdM.001. Login with existing user
RIdM.002	TwinERGY IdM platform HAS TO reject unknown users or wrong credentials	TCIdM.002. Login with non-existing user TCIdM.003. Login with existing user with wrong password
RIdM.003	TwinERGY IdM platform HAS TO return user attributes with session token	TCIdM.004. Session token attributes
RIdM.004	TwinERGY IdM platform HAS TO reject users in a Realm that try to connect from a client registered in another Realm	TCIdM.005. Validation of existing users in another Realm
RIdM.005	Consumer Comfort/Well-being module HAS TO validate an existing user through TwinERGY IdM platform	TCIdM.001. Login with existing user
RIdM.006	Home & Tertiary real-time Energy Monitoring module HAS TO validate an existing user through TwinERGY IdM platform	
RIdM.007	TwinEV module HAS TO validate an existing user through TwinERGY IdM platform	
RIdM.008	Social Network module HAS TO validate an existing user through TwinERGY IdM platform	
RIdM.009	Common platform for end-users shared data HAS TO validate an existing user through TwinERGY IdM platform	
RIP.001	Interoperability Platform HAS TO accept any client with recognized credentials	TCIP.001. Connection with correct credentials
RIP.002	Interoperability Platform HAS TO reject to any client with wrong credentials	TCIP.002. Connection with wrong credentials
RIP.003	Interoperability Platform HAS TO receive all messages sent through it	TCIP.003. Reception of messages
RIP.004	Interoperability Platform HAS TO send all messages to clients that are subscribed to associated subjects	TCIP.004. Subscription to subject (1)

		TCIP.005. Subscription to subject (2)

3.5 Execution of the test cases

The main goal of this step is to execute the test cases in order to detect if any unexpected problem appears during this process. Since this testing is done in a lab environment, some of the test cases that are referred to physical assets are executed using a simulated asset or a limited situation where the asset to be accessed is limited. The subsections (3.5.1 to 3.5.13) summarizes the responsible of the execution in each test case

3.5.1. UC01 - Home Energy Management

All test cases associated with UC01 have been normally executed. The partner responsibilities for each test case are also indicated in the following Table 32.

Table 32. Test cases associated with UC01

Test case Id	Test case description	Responsible
TC01.001	Retrieve telemetry data from sensors (through MQTT channel)	STAM
TC01.002	Retrieve telemetry data from sensors (through HTTP request)	STAM
TC01.003	View monitored energy data last timeseries	STAM
TC01.004	Consult list of anomalies	STAM
TC01.005	Browse building risk evaluations	STAM
TC01.006	Control user assets remotely	STAM
TC01.007	Custom query build on DCMP for Benetutti pilot	STAM
TC01.008	API for retrieving the digital twin information from the interoperability platform	STAM

3.5.2. UC02 - RES generation in domestic and tertiary buildings

All the test cases have been executed over the final iSCAN platform, since there is no collision with real environment

Table 33. Test cases responsible in UC02

Test case Id	Test case description	Responsible

TC02.001	Creation of the iSCAN Projects	**IES**
TC02.002	Data Channels linked to Core Data Management Platform	**IES**
TC02.003	Transactive Energy Market (TEM) Connected	**IES**
TC02.004	Automated Algorithm Initiation & Data Collection	**IES**
TC02.005	Demand Profile Optimisation	**IES**

3.5.3. UC03 - Grid capacity enhancement utilizing e-mobility

As UCS03 is tested in the Greek and German pilots, the test cases are executed in collaboration between the partners indicated in next table, with the partner highlighted in bold being the test leader. Besides, some test cases related to charge points have been tested with simulated Charge Points in ETRA lab environment.

Table 34. Test cases responsible in UC03

Test case Id	Test case description	Responsible
TC03.001	Reserve a charge point	**ETRAID**, TH-OWL, MITYLINEOS
TC03.002	Cancel a reservation	**ETRAID**, TH-OWL, MITYLINEOS
TC03.003	Reservation of available charge points (1)	**ETRAID**
TC03.004	Reservation of available charge points (2)	**ETRAID**
TC03.005	Sends RES production forecasting	**TH-OWL**, ETRAID
TC03.006	Send neighbourhood flexibility profile	**IES**, ETRAID
TC03.007	Send Grid status	**TH-OWL**, ETRAID
TC03.008	Send user appliances flexibility	**IES**, ETRAID
TC03.009	Send LEM pricing	**WEC**, ETRAID
TC03.010	Load RES production forecast	**ETRAID**
TC03.011	Load neighbourhood flexibility profile	**ETRAID**
TC03.012	Load user appliances flexibility	**ETRAID**
TC03.013	Load LEM pricing	**ETRAID**

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

TC03.014	Load actual consumption / production	**ETRAID,** SUITE5
TC03.0015	Sending of charge curve	**ETRAID**, TH-OWL, MITYLINEOS
TC03.0016	Load grid status	**ETRAID**, TH-OWL
TC03.0017	Send minimal and maximal power to charge points	**ETRAID**, TH-OWL, MITYLINEOS
TC03.0018	Send charging session costs	**ETRAID**
TC03.0019	Load charging session costs	**WEC**

3.5.4. UC04 - Prosumer's empowerment in local energy trading markets

Due to the fact that some of the test cases need verification in pilot sites, the execution of the test cases for UC04 has been made available internally in a simulated environment.

Table 35. Test cases responsible in UC04

Test case identifier	Name	Partners involved
TC04.001	Test RPOA functionality	**WEC**
TC04.002	Test connection and transmitting to blockchain network	**WEC**
TC04.003	Test token Smart Contracts	**WEC**
TC04.004	Test settle mechanisms	**WEC**
TC04.005	Test P2P transaction Smart Contracts	**WEC, STAM**
TC04.006	Test DER Smart Contract	**WEC, SmartEN**
TC04.007	Test LEM price recording and broadcasting	**WE, UOP**

3.5.5. UC05- Enhance grid flexibility through DER Management

Due to the fact that some of the test cases need verification in pilot sites, the execution of the test cases for UC05 has been made with the cooperation of the responsible partners for the test case and the corresponding pilots. Although UC05 is implemented in the Italian and German pilots, the test case was executed by TH OWL in its own pilot and local server.

Table 36. Test cases responsible in UC05

Test case identifier	Name	Partners involved
TC05.001	Initialise the module	**TH-OWL**
TC05.002	Download weather information	**TH-OWL**
TC05.003	Test the ICC implementation	**TH-OWL**
TC05.004	Test the NATS communication	**TH-OWL**

3.5.6. UC06 - Consumer's engagement in Demand Side Management Programs Utilizing feedback mechanisms

Due to the fact that some of the test cases need the verification in pilot site, the execution of the test cases for UC06 has been made between the responsible partner of each test case and the pilots' responsible. As UC06 is implemented in Greek and Italian pilots, the test cases are executed in collaboration between the partners indicated in next table being the partner highlighted in bold the leader of the test

Table 37. Test cases responsible in UC06

Test case identifier	Name	Partners involved
TC06.001	Test DR and EL (Energy Light) signal calculation	**TH-OWL**
TC06.002	Send DR signal via NATS	**TH-OWL**
TC06.003	Access EL-signal via NATS	**TH-OWL**, MITYLINEOS, STAM
TC06.004	Test refreshing of the EL-signal	**TH-OWL**, MITYLINEOS, STAM

3.5.7. UC07 - Consumer's engagement in demand response programs utilizing a socio-economic context

The execution of the test cases for UC07 is performed between the responsible partners (ED, IES, WEC) and the pilots' responsible.

Table 38. Test cases responsible in UC07

Test case identifier	Name	Partners involved

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

TC07.001	Send Consumption Data	IES
TC07.002	Send Disaggregated Profile	IES
TC07.003	Send Generation Profile	ED
TC07.004	Sends forecasted PV profiles	ED
TC07.005	Send Shared DER Assets	UoP
TC07.006	Send Equipment Consumption	Pilot Leaders
TC07.007	Send forecasted energy behaviour	Pilot Leaders
TC07.008	Send optimized and baseline net curve for a household	Pilot Leaders
TC07.009	Send data relevant to consumer well-being parameters	Pilot Leaders
TC07.010	Preview user energy analytics and monitoring features for daily, weekly, monthly basis	ED
TC07.011	Sync and preview information in regard to the balance of the consumers digital wallet and the TwinERGY point earned	ED-WEC
TC07.012	SNM retrieving raw data from CDMP	ED, S5
TC07.013	SNM retrieving data from iSCAN	ED

3.5.8. UC08 - Consumer's engagement in demand response programs utilizing personalized comfort/health-oriented services

The testing of this use case has been performed in a lab environment with participants that can replicate the actual consumers.

Table 39. Test cases responsible in UC08

Test case identifier	Name	Partners involved
TC08.001	Check readings availability	UoP
TC08.002	Check readings availability	UoP
TC08.003	Check input availability	UoP
TC08.004	Check execution of algorithms	UoP

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

TC08.005	Check execution of algorithms	UoP

3.5.9. UC09 - Consumer's Engagement in Demand Response Programs Utilizing Digital Twin Prediction Capabilities for Dynamic VPPS

All the test cases have been executed over the final iSCAN platform, since there is no collision with real environment

Table 40. Test cases responsible in UC09

Test case Id	Test case description	Responsible
TC09.001	Creation of the iSCAN Projects	**IES**
TC09.002	Data Channels linked to Core Data Management Platform	**IES**
TC09.003	Transactive Energy Market (TEM) Connected	**IES**
TC09.004	Automated Algorithm Initiation & Data Collection	**IES**
TC09.005	Demand Profile Optimisation	**IES**
TC09.006	Explicit & Implicit Demand Response Results display	**IES**

3.5.10. Transactive Energy Blockchain Network

Due some of the test cases need the verification in pilot site, the execution of the test cases for UC04 has been made available internally in a simulated environment.

Table 41. Partners involved in execution of Blockchain network test cases

Test case identifier	Name	Partners involved
TC04.001	Test RPOA functionality	**WEC**
TC04.002	Test connection and transmitting to blockchain network	**WEC**
TC04.003	Test token Smart Contracts	**WEC**
TC04.004	Test settle mechanisms	**WEC**
TC04.005	Test P2P transaction Smart Contracts	**WEC,** STAM
TC04.006	Test DER Smart Contract	**WEC,** SmartEN

TC04.007	Test LEM price recording and broadcasting	**WEC,** UOP

3.5.11. Core Data Management Platform

This platform has been tested in its final deployment.

Table 42. Partners involved in execution of test cases related to the CDMP

Test case identifier	Name	Partners involved
TCCDMP.001	File ingestion	Suite5
TCCDMP.002	Ingestion via data provider's available API	Suite5
TCCDMP.003	Retrieve data via APIs	Suite5

3.5.12. TwinERGY Identity Management Platform & Citizens Platform

As explained before, the TwinERGY Identity Manager Platform (IdM Platform) acts as the validator of user identities for the TwinERGY modules that have direct interaction with end users (those with Graphical User Interfaces). The IdM responsible development partner -ETRAID- must ensure that this validation is enabled; however, the validation of each link between IdM Platform and individual TwinERGY modules requires the cooperation of the module developers. This implies that ETRAID has tested the communications through the related five test cases detailed in section 3.4. For that, ETRAID has created some test users, that simulate the different end users that use the services in TwinERGY. The rest of the communications links in the project architecture have been tested within the corresponding set of tests for the Use Cases where the modules are the main participating players.

Table 43. Partners involved in execution of test cases related to the Id Management platform

Test case identifier	Name	Partners involved
TCIdM.001	Login with existing user	ETRAID
TCIdM.002	Login with non-existing user	ETRA

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

TCIdM.003	TCIdM.003. Login with existing user with wrong password	ETRA
TCIdM.004	TCIdM.004. Session token attributes	ETRA
TCIdM.005	TCIdM.005. Validation of existing users in another Realm	ETRA

3.5.13. TwinERGY Interoperability Platform

The Interoperability Platform (IP) acts as a gateway of communication among modules. So, despite IP responsible -ETRAID- must ensure that connections and transmissions of messages are available, it is not responsible of each link between Interoperability Platform and another module in TwinERGY.

This implies that ETRAID has tested the communications through the five test cases detailed in section 3.4. For that, ETRAID has prepared a module with the code indicated in the test cases. The rest of communications links, as was commented before has been tested for each module in the corresponding set of tests for the Use Case where the module is the main participating component.

Table 44. Partners involved in execution of test cases related to the Interoperability platform

Test case identifier	**Name**	**Partners involved**
TCIP.001	Connection with correct credentials	ETRAID
TCIP.002	Connection with wrong credentials	ETRAID
TCIP.003	Reception of messages	ETRAID
TCIP.004	Subscription to subject (1)	ETRAID
TCIP.005.	Subscription to subject (2)	ETRAID

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

4. Integrated lab-testing results

In this section, the results of the execution of each test case are presented. The results are summarized in Table 45 with the following detailed information:

- The related component or Use case
- The Test case identifier
- The responsible partner
- The result of the execution, namely:
  - Success: the test case finishes in its defined pass criteria
  - Fail: the test case finished in its defined suspension criteria
  - Pending: the test case is blocked or it could not be executed due some external causes or unexpected event.

Table 45. Results of execution of test cases

Related asset/HLUC	Test case ID	Responsible	Results
CDMP	TCCDMP01	Suite5	Success
	TCCDMP02	Suite5	Success
	TCCDMP03	Suite5	Success
Id Management Platform and Citizens platform	TCIdM.001	ETRAID	Success
	TCIdM.002	ETRAID	Success
	TCIdM.003	ETRAID	Success
	TCIdM.004	ETRAID	Success
	TCIdM.005	ETRAID	Success
	TCIdM.006	ETRAID	Success
	TCIdM.007	ETRAID	Success
	TCIdM.008	ETRAID	Success
	TCIdM.009	ETRAID	Success
	TCIdM.010	ETRAID	Success

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Interoperability platform (NATS)	TCIP.001	ETRAID	Success
	TCIP.002	ETRAID	Success
	TCIP.003	ETRAID	Success
	TCIP.004	ETRAID	Success
	TCIP.005	ETRAID	Success
HLUC01	TC01.001	STAM	Success
	TC01.002	STAM	Success
	TC01.003	STAM	Success
	TC01.004	STAM	Success
	TC01.005	STAM	Success
	TC01.006	STAM	Success
	TC01.007	STAM	Success
	TC01.008	STAM	Fail
HLUC02	TC02.001	IES	Success
	TC02.002	IES	Success
	TC02.003	IES	Success
	TC02.004	IES	Success
	TC02.005	IES	Success
HLUC03	TC03.001	ETRAID	Success
	TC03.002	ETRAID	Success
	TC03.003	ETRAID	Success
	TC03.004	ETRAID	Success
	TC03.005	THOWL	Success
	TC03.006	IES	Success

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	TC03.007	THOWL	Success
	TC03.008	IES	Success
	TC03.009	WEC	Pending
	TC03.010	ETRAID	Success
	TC03.011	ETRAID	Success
	TC03.012	ETRAID	Success
	TC03.013	ETRAID	Success
	TC03.014	ETRAID	Success
	TC03.015	ETRAID	Success
	TC03.016	ETRAID	Success
	TC03.017	ETRAID	Success
	TC03.018	ETRAID	Success
	TC03.019	WEC	Pending
HLUC04 / Transactive Energy Platform & Market	TC04.001	WEC	Success
	TC04.002	WEC	Success
	TC04.003	WEC	Success
	TC04.004	WEC	Success
	TC04.005	WEC	Success
	TC04.006	WEC	Fail
	TC04.007	WEC	Fail
HLUC05	TC05.001	THOWL	Success
	TC05.002	THOWL	Success
	TC05.003	THOWL	Success

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	TC05.004	THOWL	Success
HLUC06	TC06.001	THOWL	Success
	TC06.002	THOWL	Success
	TC06.003	THOWL	Success
	TC06.004	THOWL	Fail
HLUC07	TC07.001	IES-ED	Success
	TC07.002	IES-ED	Success
	TC07.003	IES-ED	Success
	TC07.004	IES-ED	Success
	TC07.005	IES-ED	Success
	TC07.006	IES-ED	Success
	TC07.007	IES-ED	Success
	TC07.008	IES	Success
	TC07.009	ED	Success
	TC07.010	IES-ED	Success
	TC07.011	WEC-ED	Success
HLUC08	TC08.001	UoP	Success
	TC08.002	UoP	Success
	TC08.003	UoP	Success
	TC08.004	UoP	Success
	TC08.005	UoP	Success
HLUC09	TC09.001	IES	Success
	TC09.002	IES	Success

	TC09.003	IES	Success
	TC09.004	IES	Success
	TC09.005	IES	Success
	TC09.006	IES	Success

As it can be seen, some of the test cases has finished in a failed status. Here we present the causes and the actions to be executed in order to fix the detected problem.

Table 46. Causes and mitigation actions for pending or failed test cases.

Requirement	**Test case**	**Cause / Mitigation actions**
R01.007. iScan MUST share the asset building configuration in a json file with an API REST Prority 5.	TC01.008. API for retrieving the Digital Twin information from the interoperability platform	Not clear where this data are retrieved. For the time of delivering this document, these data will be managed locally without a need for integration (only for Benetutti pilot site)
R04.009. Based on a synchronization of DER reward with the Social Network Module, Transactive Energy Platform MUST instruct the Smart Contract to mint and distribute DER tokens. Prority 5.	TC04.006. Instruction on the Smart Contract to mint and distribute DER tokens	Instructions from Social Module will be sinchronised with the new ERC1155 / ERC 721 protocol and the test will be successful at that time
R04.010. Transactive Energy Platform HAS TO transmit the LEM price to the Smart Contract and store the data on the ledger Prority 5.	TC04.007. Transmission of LEM price to the LEM price – logic.	LEM price is posted to the Blockchain but not transmited to NATS because of shift of protocols from ERC20 to ERC1155 / ERC 721 / Polygon. Will be fully successful and posted to NATS with implementation of new ERC1155 / ERC 721 protocols
R06.004.	TC06.004.	In this moment, THOWL is studying where is the source of the problem

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

The Energy Light SHOULD refresh every 15 minutes. Prority 3.	Transactive Energy Platform HAS TO transmit the LEM price to the Smart Contract and store the data on the ledger	

So, responsible partners are in this moment in the way of discovering and fixed the detected problems. The three first requirement have a priority of 5, so they must be fixed before the end deployment in the real system. About the requirement R06.004, as it has a priority level 3, it is not needed to have it fixed before the final deployment. However, the intention is to have it fixed before it.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

5.Demonstration activities timeline and planning

In section 4, the results of the execution of the test cases have been presented. execute Despite a few test cases have finalized in the not expected status, among a large number of successful test cases, one can conclude, that almost all requirements are fulfilled. Based on that, delivery the consortium partners can now go on with demos in the real environment.

More specifically, following the final deployment of modules in the real environment, the next step is the execution of the corresponding tests by each pilot in the real environment, in collaboration with the developers of the platform. In this direction, test users will be created within the TwinERGY framework so that consortium partners can check whether the developed applications can communicate and work with real (test) data.

As potential problems may be expected in this first integration, after the validation in the real environment, developers may have to solve problems in the real environment before the validation with real end-users is generated. Therefore, for the rest of the project life, we pretend to use the defined test cases, to validate the requirements exposed in this deliverable, always where it is applicable.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

6. Conclusions

This document has summarized the work related to the testing in a lab environment, done during Task T8.2. The main objective of the testing is to detect the problems that can occur during the integration of components in TwinERGY, in order to avoid later problems when the final integration in the real system is produced. This is executed by following a test plan, being composed by these steps:

1. Revision of the features and requirements to be probed, using the system architecture described in D4.4 as the starting point. These features are related to integration, since the working of each component as an isolated item has been probed during the phase of development.
2. Classification of the features and requirements into groups in order to effectively determine the appropriate type of testing in each case. In this step we define the group of testing which each requirement is belonging to, such as control, functionality, etc.
3. Definition of the set of test cases that validate each of requirements and each responsible of a requirement is validated.
4. Execution of the defined test cases, so each responsible is able to detect if the requirement is validated or if a problem has appeared.
5. Documentation of execution results and determination of the next process steps and the test protocols. In this point we have validated that almost test cases have finished in a sucesfull status. However, six test cases were failed. For them we have exposed the causes and the actions to be proceeded in order to fix the problems.

As a final conclusion, we can say that the testing performed in this task is essential in order to prevent future difficulties during the upcoming deployment and implementation operation of TwinERGY components in real pilot environment, in terms of possible malfunction, delays, or needs to rework. The results have demonstrated the solidity of the solutions within the TwinERGY solution for the scope that provides confidence that no serious problems will be faced during the final deployment of the TwinERGY system.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

References

[1] TwinERGY project, «TwinERGY - Deliverables,» [En línea]. Available: https://www.twinergy.eu/deliverables. [Último acceso: December 2022].

[2] E. Commision, «NOBEL GRID (New Cost Efficient Business Models for Flexible Smart Grids),» [En línea]. Available: https://ec.europa.eu/inea/en/horizon-2020/projects/h2020-energy/grids/nobel-grid. [Último acceso: 12 2022].

[3] IEEE Standard for Software and System Test Documentation, «IEEE 829-2008,» [En línea]. Available: https://standards.ieee.org/ieee/829/3787/. [Último acceso: 12 2022].

[4] Network Working Group , « Key words for use in RFCs to Indicate Requirement Levels,» [En línea]. Available: https://datatracker.ietf.org/doc/html/rfc2119.

[5] Keycloak, «Keycloak features and concepts,» [En línea]. Available: https://www.keycloak.org/docs/latest/server_admin/.

[6] USEF, «USEF: The Framework explained,» 2015.

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Annex 1. Complete description of Test Cases

Test cases for UC01

Name	TC01.001. Retrieve telemetry data from sensors (MQTT)		
Module under test	H&TEM	**Resp.**	STAM
Module requirement	R01.001		
Test environment	H&TEM backend platform, monitoring devices		
Features to be tested	Remote connection with the sensors		
Features not to be tested			
Preparation	- Sensors installed in building - Sensors sending data to a gateway via MQTT - Gateway installed in building - Gateway connected to a LAN - Gateway sending data to backend platform - Backend platform up and running		
Dependencies			
Steps	1. Check logs in backend platform (as module admin / developer)		
Pass criteria	The logs contain data not older than 30 minutes		
Suspension criteria	- Logs do not contain telemetry data - Data is older than 30 minutes		
Results	Telemetry payload correctly received and containing sensor data		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Name	TC01.002. Retrieve telemetry data from sensors (HTTP)		
Module under test	H&TEM	**Resp.**	STAM
Module requirement	R01.002		
Test environment	H&TEM backend platform, monitoring devices		
Features to be tested	Remote connection with the sensors		
Features not to be tested			
Preparation	- Sensors installed in building - Sensors connected to a LAN - Sensors sending data to the backend platform via HTTP - Backend platform up and running		
Dependencies			
Steps	1. Check logs in backend platform (as module admin / developer)		
Pass criteria	The logs contain data not older than 30 minutes		
Suspension criteria	- Logs do not contain telemetry data - Data is older than 30 minutes		
Results	Telemetry payload correctly received and containing sensor data		

Name	TC01.003. View monitored energy data last timeseries		
Module under test	H&TEM	**Resp.**	STAM
Module requirement	R01.003		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Test environment	H&TEMM platform, monitoring devices, data gathering process
Features to be tested	Energy data displayed on the platform
Features not to be tested	
Preparation	1. Backend platform up and running 2. Backend contains previously registered telemetry data
Dependencies	TC01.001 or TC01.002
Steps	1. Log into the platform 2. Go to the Dashboard page
Pass criteria	Energy data related to the connected user is displayed in the chart, showing timeseries for the last 7 days
Suspension criteria	- The displayed data is not the one the user is expecting - The data does not update itself after a short time
Results	Energy data related to the connected user is displayed in the chart

Name	TC01.004. Consult list of anomalies		
Module under test	H&TEM	**Resp.**	STAM
Module requirement	R01.004		
Test environment	H&TEMM platform, monitoring devices, fault anomaly detection process		
Features to be tested	Anomalies detected related to the user's appliances		
Features not to be tested			
Preparation	1. Monitored data correctly registered 2. Anomaly detection process running 3. Anomalies detected and stored in a DB		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Dependencies	
Steps	1. Log into the platform 2. Go to the Anomalies Detection page
Pass criteria	List of anomalies displayed in the table
Suspension criteria	No anomalies displayed
Results	Table in the page is filled with a list of anomalies data and their dates

Name	TC01.005. Browse building risk evaluations		
Module under test	Risk Module	**Resp.**	STAM
Module requirement	R01.005		
Test environment	H&TEM platform, risk evaluation process		
Features to be tested	Risk related to the connected user's building		
Features not to be tested			
Preparation	Building data correctly retrieved from DT		
Dependencies			
Steps	1. Log into the platform 2. Go to the Risk Management page 3. Select a target and a threat from the dropdown menus 4. Press the button Run Simulation		
Pass criteria	Risk data correctly appeared in the page		
Suspension criteria	- Simulation fails to start - Dropdown menus do not have threats or targets		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Results	- Economic impact displayed on the left side - Possible countermeasures displayed on the left side - List of assets and related threats displayed on the right side - Risk level and probability of threats displayed for each asset

Name	TC01.006. Control user assets remotely		
Module under test	H&TEM	**Resp.**	STAM
Module requirement	R01.006		
Test environment	H&TEM platform, user appliances		
Features to be tested	Asset control via RPC		
Features not to be tested			
Preparation	- Metering devices correctly installed - Metering devices sending data to the backend platform		
Dependencies			
Steps	1. Log into the platform 2. Go to the Dashboard page 3. Use the sliders to manage the assets on the bottom of the page		
Pass criteria	The selected appliance turns on or off		
Suspension criteria	- No appliance showed in the Dashboard page - The asset does not turn on / off		
Results	User can easily manage their assets from the platform		

Name	TC01.007. Custom query build on DCMP for Beneutti pilot

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module under test	H&TEM	**Resp.**	STAM
Module requirement	R01.007		
Test environment	H&TEM platform, user appliances		
Features to be tested	Monitoring dashboard		
Features not to be tested			
Preparation	- Metering devices correctly installed - Metering devices sending data to CDMP		
Dependencies			
Steps	1. Log into the platform 2. Go to the Dashboard page 3. Visualize real time data from the Beneutti Pilot		
Pass criteria	Data is correctly retrieved		
Suspension criteria			
Results	Data is available on the H&TEM module		

Name	TC01.008. API for retrieving the digital twin information from the interoperability platform		
Module under test	H&TEM,	**Resp.**	STAM
Module requirement	R01.008 –		
Test environment	H&TEM platform, user appliances		
Features to be tested	All features		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Features not to be tested	
Preparation	- Asset modelled on iSCAN - Integration pipelines from iSCAN to interoperability platform - Interoperability platform up and running
Dependencies	
Steps	1. Log into the platform 2. Go to the Dashboard page 3. Use the sliders to manage the assets on the bottom of the page
Pass criteria	The correct appliances are shown for the user
Suspension criteria	- No appliance showed in the Dashboard page
Results	User can easily monitor and manage their assets from the platform

Test cases for UC02

Name	TC02.001: Creation of the iSCAN Projects		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R02.001		
Test environment	Azure on the cloud, python, iSCAN		
Features to be tested	Creation of environment		
Features not to be tested	Population of the iSCAN channels with Data, linking the data sources within the Core Data Management Platform to the channels		
Preparation	The Core Data Management Platform is active and available. The user has access to the predefined JSON file and has the required information for the building description and data channels relevant to the building.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Dependencies	NA
Steps	1. The user accesses the predefined JSON file 2. The user inputs the required fields (building description, location, list of data inputs required, etc) 3. The JSON file is saved and the user inputs the file path/hyperlink as an input to the algorithm 4. The algorithm accesses the file and creates the iSCAN project for the building using the building description and location from the JSON file. 5. The algorithm creates the individual data channels as specified by the user in the JSON file and is saved to define the iSCAN project for the building 6. The algorithm accesses the building location in the JSON to define the project location and establish the relevant weather file 7. Once the building level iSCAN project is defined, the algorithm creates the community level project 8. The Energy Tariff Channel is created 9. The community level project creates dedicated channels for aggregated demand for each individual building, optimised rescheduling of demand, cost analysis, renewable generation, flexibility assessment and end-user acceptance ratio. The iSCAN project is now set up.
Pass criteria	The iSCAN project is set up at both building and community level with all specified channels created and linked to the relevant channels in the Core Data Management Platform.
Suspension criteria	• The algorithm can't access the JSON file • The algorithm can't read the JSON file • The algorithm fails to initiate the creation of the iSCAN project • The algorithm is unable to create the specified iSCAN channels • The creation of the building level iSCAN project does not trigger the creation of the Community level iSCAN project • The required Community level iSCAN channels are not created The channels are not linked to the relevant data source within the Core Data Management Platform.
Results	The system has validated the creation of the environment

Name	TC02.002: Data Channels linked to Core Data Management Platform		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	R02.002
Test environment	Azure on the cloud, python, iSCAN, Core Data Management Platform (DMP)
Features to be tested	The user can link the iSCAN channels to the relevant data source within the Core Data Management Platform
Features not to be tested	Automated Population of the iSCAN Channels with Data
Preparation	The iSCAN project has been created and the relevant channels in place. The DMP is available and contains data for each data source.
Dependencies	TC02.001
Steps	1. The user identifies the individual iSCAN channel and its associated API 2. The user accesses the Core Data Management Platform and identifies the relevant data source within the DMP 3. The user provides the data source with the API link to push data to the iSCAN channel 4. The user then initiates the transfer to verify the connection The iSCAN channel is populated with the data required.
Pass criteria	The user is able to link the iSCAN channel to the relevant data source via API and can initiate the population of each iSCAN data with data.
Suspension criteria	• The user is unable to link the relevant channel with the data source The data transfer fails
Results	The system has validated the data channels connections

Name	TC02.003: Transactive Energy Market (TEM) Connected		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R02.003		
Test environment	Azure on the cloud, python, iSCAN, TEM		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Features to be tested	The user can link the iSCAN channels to the TEM
Features not to be tested	Automated Population of the iSCAN Energy Tariff Channel with Data
Preparation	The iSCAN project has been created and the relevant channels in place. The TEM is active.
Dependencies	TC02.001
Steps	1. The user identifies the individual iSCAN Energy Tariff and its associated API 2. The user provides the iSCAN API to the TEM to allow the module to push updated tariff data to the iSCAN channel 3. The user then initiates the transfer to verify the connection 4. The iSCAN channel is populated with the data required.
Pass criteria	The user is able to link the iSCAN channel to the relevant data source via API and can initiate the population of each iSCAN data with data.
Suspension criteria	• The user is unable to link the relevant channel with the TEM • The data transfer fails The TEM fails to initiate when called by the user during verification.
Results	The system has validated the Transactive Energy Market (TEM) connections

Name	TC02.004: Automated Algorithm Initiation & Data Collection		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R02.004		
Test environment	Azure on the cloud, python, iSCAN, DMP, TEM, VE, StRoBe Library, iVN		
Features to be tested	The algorithm is automatically initiated and the data transferred to the relevant iSCAN channels		
Features not to be tested	Optimisation of the Demand Profiles		
Preparation	The iSCAN project has been set up and all relevant channels created. The API for each of the channels have been mapped to		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	the data sources within the DMP and TEM. The StRoBe library and Building Digital Twin are available.
Dependencies	TC02.001, TC02.002, TC02.003
Steps	1. At a predefined time (18:00) the algorithm is initiated and begins to run 2. The algorithm accesses the predefined JSON file 3. The algorithm verifies that the iSCAN project and all relevant channels are created 4. The algorithm calls on the DMP and data for the previous 24hours is pushed form the DMP to the iSCAN channel via API 5. The algorithm calls on the TEM and forecasted tariff data for the next 24 hours are pushed to the iSCAN channel 6. The iVN engine is called and generates a RES profile for the relevant time period (one day ahead) 7. The algorithm validates that all necessary data has been transferred to the iSCAN project 8. Where data is missing, the Building Digital Twin and StRoBe library are initiated and used to populate the missing data channels 9. All iSCAN channels are now populated with the most current data available.
Pass criteria	All iSCAN channels are now populated with the most up to date data
Suspension criteria	• Algorithm fails to initiated at the defined time • Communication error with the DMP or the TEM • Missing data is not identified and sourced from alternate sources iSCAN channels fail to populate
Results	The system has validated the Algorithm Initiation automation & Data Collection

Name	TC02.005: Demand Profile Optimisation		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R02.005		
Test environment	Azure on the cloud, python, iSCAN,		
Features to be tested	The algorithm is automatically initiated and the data transferred to the relevant iSCAN channels		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Features not to be tested	Optimisation of the Demand Profiles
Preparation	The iSCAN project has been set up and all relevant channels created. The API for each of the channels have been mapped to the data sources within the DMP and TEM. The StRoBe library and Building Digital Twin are available.
Dependencies	TC02.001, TC02.002, TC02.003
Steps	1. The iSCAN API is called to retrieve building level time-series data for the last 30 days 2. This time-series data is then analysed through integrated Python libraries to identify the pattern of use of smart appliances over the defined time-period 3. The process generates the standard profile of use for the next day and determines the shiftable loads based on user input 4. If the appliance use is shiftable, a preferred time period of use is defined based on previous usage patterns or user preferences 5. The forecast profiles are resampled at the same time resolution 6. The optimisation algorithm assesses and inputs the resampled forecast profiles and schedules the loads based on RES production and cost 7. The algorithm optimises operational cost at the building level individually 8. The algorithm identifies the list of starting point in minutes over the course of the day for each individual smart appliance 9. The demand profile is reconstructed based on the starting point for each appliance. This is the optimised demand profile 10. The optimised demand profile for each building is then returned to the corresponding aggregated iSCAN data channel 11. The individual building/customer data channel is then aggregated to derive the community based optimised profile.
Pass criteria	The building and community level optimised demand profiles are defined
Suspension criteria	• The python libraries aren't initiated • The profile is not optimised based on cost and RES production The optimised profiles are not returned.
Results	The system has validated the Demand Profile Optimisation

Test cases for UC03

Name	TC03.001. Reserve a charge point

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module under test	TwinEV, CPO, CP	**Resp.**	ETRAID
Module requirement	R03.001		
Test environment	TwinEV for drivers, CPO and CP		
Features to be tested	Communication TwinEV-CPO and CPO-CP. Control of CP by TwinEV		
Features not to be tested			
Preparation			
Dependencies			
Steps	1. Log into TwinEV for drivers 2. Go to screen "Search and book" 3. Search for available stations 4. Select one available station 5. Click on button "Reserve" 6. Log into TwinEV Dashboard 7. Go to screen commands 8. Check commands table		
Pass criteria	A new arrow appears in the table with: command "Reserved", current timestamp, location ID corresponding to the charge station.		
Suspension criteria	The new line does not appear or the data in the line are not correct		
Results	1. A message with the confirmation appears in the application. 2. The charge point is locked		

Name	TC03.002. Cancel a reservation		
Module under test	TwinEV, CPO, CP	**Resp.**	ETRAID

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	R03.002
Test environment	TwinEV for drivers, CPO and CP
Features to be tested	Communication TwinEV-CPO and CPO-CP. Control of CP by TwinEV
Features not to be tested	
Preparation	• It is needed to have a reservation in one charge point
Dependencies	TC03.001. Reserve a charge point
Steps	1. Log into TwinEV for drivers 2. Go to screen "Charge" 3. Click on button "Cancel" 4. Log into TwinEV Dashboard 5. Go to screen commands 6. Check commands table
Pass criteria	The Charge Point is unlocked
Suspension criteria	A new arrow appears in the table with: command "Cancel reservation", current timestamp, corresponding to the charge station.
Results	1. The screen in TwinEV for drivers is in its initial state. 2. The change point is unlocked.

Name	TC03.003. Reservation of available charge points (1)		
Module under test	TwinEV	**Resp.**	ETRAID
Module requirement	R03.003		
Test environment	TwinEV for drivers, CPO and CP		
Features to be tested	Communication TwinEV-CPO and CPO-CP.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Features not to be tested	
Preparation	Reserve a charge point following TC03.001
Dependencies	TC03.001. Reserve a charge point
Steps	1. Log into TwinEV for drivers 2. Go to screen "Search and book" 3. Search for available stations 4. Select one available station 5. Click on button "Reserve" 6. Return again to "Search and book" 7. Search for available stations with same criteria as step 3
Pass criteria	The list of available stations does not include the charge point reserved in step 5.
Suspension criteria	The charge point is in the list
Results	A list with available stations, not including the charge point previously reserved, appears

Name	TC03.004. Reserve of available charge points (2)		
Module under test	TwinEV	**Resp.**	ETRAID
Module requirement	R03.003		
Test environment	TwinEV for drivers, CPO and CP		
Features to be tested	Communication TwinEV-CPO and CPO-CP.		
Features not to be tested			
Preparation	Cancel an active reservation following TC03.002		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Dependencies	TC03.001. Reserve a charge point TC03.002. Cancel a reservation
Steps	1. Log into TwinEV for drivers 2. Go to screen "Search and book" 3. Search for available stations 4. Select one available station 5. Click on button "Reserve" 6. Cancel the reservation 7. Return again to "Search and book" 8. Search for available stations with same criteria as step 3.
Pass criteria	The Charge Point appears in the list of available charge points
Suspension criteria	The Charge point is not in the list
Results	A list with available stations appears

Name	TC03.005. Send RES production forecasting		
Module under test	RES Integration & DER Management module	**Resp.**	TH-OWL
Module requirement	*R03.004*		
Test environment	RES Integration & DER Management module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	• RES Integration & DER Management module has a client connected to Interoperability platform		
Dependencies			

Steps	1. Log into the Interoperability platform console website.
	2. Navigate to screen "Messages"
	3. Subscribe to subject 'M5.renewableGeneration.Building.Pilot'
	4. Execute RES Integration & DER Management module, ensuring that the module executes a loop:
	• The module calculates the RES production forecasting
	• The module transforms the RES production forecasting into a json.
	• The module module connects to interoperability platform through a NATS client
	• The module publishes a message with structure:
	{
	"generationID": "string",
	"generationName":"string",
	"buildingName": "string",
	"pilotName": "string",
	"startDate: "YYYY-MM-DD",
	"endDate: "YYYY-MM-DD",
	"timeZoneOffset: float,
	"samplePeriod: int,
	"profile": [floats]
	}
	and subject: 'M5.renewableGeneration.Building.Pilot'

Pass criteria	In the table of screen "Messages" in Interoperability platform console a message appears, containing the same structure and values as those sent by the RES Integration & DER Management module
Suspension criteria	None message in the table of screen "Messages" appears
Results	The message with the information has been sent to the Interoperability Platform

Name	TC03.006. Send neighbourhood flexibility profile

Module under test	Neighbourhood demand flexibility profiling module	**Resp.**	*IES*
Module requirement	*R03.005*		
Test environment	Neighbourhood demand flexibility profiling module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	• Neighbourhood demand flexibility profiling module has a client connected to Interoperability platform		
Dependencies			

Steps	1. Log into the Interoperability platform console website.
	2. Navigate to screen "Messages"
	3. Subscribe to subject ' 'M2.neighbourhoodFlexibility.Pilot'
	4. Execute Neighbourhood demand flexibility profiling module, ensuring that the module executes a loop:
	• The module calculates the calculates the neighbourhood demand flexibility profile
	• The module transforms the profile into a json.
	• The module connects to interoperability platform through a NATS client
	• The module publishes a message with structure:
	{
	"pilotName":"string",
	"startDate: "YYYY-MM-DD",
	"endDate: "YYYY-MM-DD",
	"timeZoneOffset: float,
	"samplePeriod: int,
	"profile": ArrayOf(floats),
	}
	and subject: 'M2.neighbourhoodFlexibility.Pilot'		

Pass criteria	In the table of screen "Messages" in Interoperability platform console a message appears, containing the same structure and values as those sent by the Neighbourhood demand flexibility profiling module.		

Suspension criteria	None message in the table of screen "Messages" appears
Results	The message with the information has been sent to the Interoperability Platform

Name	TC03.007. Send Grid status		
Module under test	RES Integration & DER Management module	**Resp.**	*TH-OWL*
Module requirement	*R03.006*		
Test environment	RES Integration & DER Management module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	• RES Integration & DER Management module has a client connected to Interoperability platform		
Dependencies			

Steps	1. RES Integration & DER Management module receives the grid status from substations
	2. RES Integration & DER Management transform the Grid status in a json called GridData
	3. RES Integration & DER Management module connects to interoperability platform through a NATS client
	4. RES Integration & DER Management module publishes a message {"code": 200, "data": GridData} with subject 'RESMODULE.notif.GridInfo', including a callback, like this:
	
	Function (err, guid){
	if(err) {		

	console.log('Error publishing Grid data: ' + err);
	} else {
	console.log('Published message Grid data with guid: ' + guid);
	}

Pass criteria	The callback shows a "Published message Grid data"
Suspension criteria	The callback doesn't show any message or shows an error message
Results	After the publish, RES Integration & DER Management module knows if the message was sent or something was wrong

Name	TC03.008. Send user appliances flexibility		
Module under test	Consumer Demand Flexibility Profiling module	**Resp.**	*IES*
Module requirement	*R03.007*		
Test environment	Consumer Demand Flexibility Profiling module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	Consumer Demand Flexibility Profiling module has a client connected to Interoperability platform		
Dependencies			
Steps	1. Log into the Interoperability platform console website. 2. Navigate to screen "Messages" 3. Subscribe to subject 'M2.applianceFlexibility.Building.Pilot' 4. Execute Consumer demand flexibility profiling module, ensuring that the module executes a loop:		

	• The module calculates the consumer demand flexibility profile
	• The module transforms the profile into a json.
	• The module connects to interoperability platform through a NATS client
	• The module publishes a message with structure:
	{
	"applianceID":"string",
	"applianceName":"string",
	"buildingName":"string",
	"pilotName":"string",
	“startDate: "YYYY-MM-DD",
	“endDate: "YYYY-MM-DD",
	“timeZoneOffset: float,
	“samplePeriod: int,
	“profile”: ArrayOf(floats),
	}
	and subject: 'M2.applianceFlexibility.Building.Pilot'

Pass criteria	In the table of screen “Messages” in Interoperability platform console a message appears, containing the same structure and values as those sent by the Neighbourhood demand flexibility profiling module.
Suspension criteria	None message in the table of screen “Messages” appears
Results	The message with the information has been sent to the Interoperability Platform

Name	TC03.009. Send LEM pricing		
Module under test	Transactive Energy module	**Resp.**	*WEC*
Module requirement	*R03.008*		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Test environment	Transactive Energy module, Interoperability platform
Features to be tested	Communication
Features not to be tested	
Preparation	Transactive Energy module has a client connected to Interoperability platform
Dependencies	

Steps	1. Transactive Energy module calculates the LEM pricing
	2. Transactive Energy module transform the LEM pricing in a json called LEMpricing
	3. Transactive Energy module connects to interoperability platform through a NATS client
	4. Transactive Energy module publishes a message {"code": 200, "data": LEMPricing } with subject 'TEMODULE.notif.lem, including a callback, like this:
	
	Function (err, guid){
	if(err) {
	console.log('Error publishing LEM pricing: ' + err);
	} else {
	console.log('Published LEM pricing with guid: ' + guid);
	}

Pass criteria	The callback shows a "Published message LEM pricing"
Suspension criteria	The callback doesn't show any message or shows an error message
Results	After the publish, Transactive Energy module knows if the message was sent or something was wrong

Name	TC03.010. Load RES production forecast

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module under test	TwinEV module	**Resp.**	*ETRAID*
Module requirement	*R03.009*		
Test environment	TwinEV module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	• TwinEV module has a client connected to Interoperability platform. • TwinEV generates logs when it receives messages from Interoperability platform		
Dependencies	*TC03.005*		

Steps	1. Execute TwinEV.
	2. Log into the Interoperability platform console website.
	3. Navigate to screen "Messages"
	4. Send a message with subject 'M5.renewableGeneration.Building.Pilot' and structure:
	{
	"generationID": "string",
	"generationName":"string",
	"buildingName": "string",
	"pilotName": "string",
	"startDate: "YYYY-MM-DD",
	"endDate: "YYYY-MM-DD",
	"timeZoneOffset: float,
	"samplePeriod: int,
	"profile": [floats]
	}		

Pass criteria	TwinEV has written in its logs a message with structure		

	{
	"generationID": "string",
	"generationName":"string",
	"buildingName": "string",
	"pilotName": "string",
	“startDate: "YYYY-MM-DD",
	“endDate: "YYYY-MM-DD",
	“timeZoneOffset: float,
	“samplePeriod: int,
	“profile”: [floats]
	}
	And same values as those in the message

Suspension criteria	TwinEV has not written any log message
Results	TwinEV updates calculations according to the message

Name	TC03.011. Load neighbourhood flexibility profile		
Module under test	TwinEV module	**Resp.**	*ETRAID*
Module requirement	*R03.009*		
Test environment	TwinEV module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	TwinEV module has a client connected to Interoperability platform		

Dependencies	*TC03.006*
Steps	• TwinEV module has a client connected to Interoperability platform. • TwinEV generates logs when it receives messages from Interoperability platform
Pass criteria	*TC03.005*

Suspension criteria	1. Execute TwinEV.
	2. Log into the Interoperability platform console website.
	3. Navigate to screen "Messages"
	4. Send a message with subject 'M2.neighbourhoodFlexibility.Pilot' and structure
	{
	"pilotName":"string",
	"startDate: "YYYY-MM-DD",
	"endDate: "YYYY-MM-DD",
	"timeZoneOffset: float,
	"samplePeriod: int,
	"profile": ArrayOf(floats),
	}

Results	TwinEV has written in its logs a message with structure
	{
	"pilotName":"string",
	"startDate: "YYYY-MM-DD",
	"endDate: "YYYY-MM-DD",
	"timeZoneOffset: float,
	"samplePeriod: int,
	"profile": ArrayOf(floats),
	}
	And same values as those in the message

Name	TC03.012. Load user appliances flexibility		
Module under test	TwinEV module	**Resp.**	*ETRAID*

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	*R03.009*
Test environment	TwinEV module, Interoperability platform
Features to be tested	Communication
Features not to be tested	
Preparation	TwinEV module has a client connected to Interoperability platform
Dependencies	*TC03.008*
Steps	• TwinEV module has a client connected to Interoperability platform. • TwinEV generates logs when it receives messages from Interoperability platform
Pass criteria	*TC03.005*

Suspension criteria	1. Execute TwinEV.
	2. Log into the Interoperability platform console website.
	3. Navigate to screen "Messages"
	4. Send a message with structure
	{
	"applianceID":"string",
	"applianceName":"string",
	"buildingName":"string",
	"pilotName":"string",
	"startDate: "YYYY-MM-DD",
	"endDate: "YYYY-MM-DD",
	"timeZoneOffset: float,
	"samplePeriod: int,
	"profile": ArrayOf(floats),
	}
	and subject: 'M2.applianceFlexibility.Building.Pilot'

Results	TwinEV has written in its logs a message with structure

	{
	"applianceID":"string",
	"applianceName":"string",
	"buildingName":"string",
	"pilotName":"string",
	“startDate: "YYYY-MM-DD",
	“endDate: "YYYY-MM-DD",
	“timeZoneOffset: float,
	“samplePeriod: int,
	“profile”: ArrayOf(floats),
	}
	And same values as those in the message

Name	TC03.013. Load LEM pricing		
Module under test	TwinEV module	**Resp.**	*ETRAID*
Module requirement	*R03.009*		
Test environment	TwinEV module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	TwinEV module has a client connected to Interoperability platform		
Dependencies	*TC03.009*		
Steps	1. TwinEV module connects to interoperability platform through a NATS client		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	2. TwinEV subscribe to subject 'TEMODULE.notif.lem'
Pass criteria	TwinEV module receives a message with subject 'TEMODULE.notif.lem' and content {"code": 200, "data": LEMPricing}
Suspension criteria	The message has not been received
Results	

Name	TC03.014. Load actual consumption/production		
Module under test	TwinEV module	**Resp.**	*ETRAID*
Module requirement	*R03.010*		
Test environment	TwinEV module, CDMP		
Features to be tested	Communication		
Features not to be tested			
Preparation			
Dependencies			
Steps	1. TwinEV selects the list of assets to be asked for. 2. TwinEV module ask for assets' actual consumption/production to CDMP 3. CDMP returns actual consumption/production		
Pass criteria	TwinEV module has the actual consumption / production		
Suspension criteria	TwinEV does not get the actual consumption/production or the info is related to other assets		
Results	TwinEV gets the actual consumption/production		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Name	TC03.015. Sending of charge curve		
Module under test	TwinEV module	**Resp.**	*ETRAID*
Module requirement	*R03.011*		
Test environment	TwinEV module, CPO, Charge points		
Features to be tested	Communication		
Features not to be tested			
Preparation	The Charge point must be locked by the user, the user is able to check the charged battery during the session.		
Dependencies	TC03.001		
Steps	1. The user selects one mode of charge 2. The user fills the rest of elements related to the session 3. The user starts the charge 4. TwinEV modules calculates the charge curve 5. TwinEV sends the curve to the charge point through the CPO 6. Charging point charges the EV battery, following the charge curve		
Pass criteria	The charge point follows the charge curve during the session		
Suspension criteria	The charge point is not charging, the charge point doesn't follow the charge curve		
Results	When the session ends, the vehicle is charged at the desired battery level		

Name	TC03.016. Load grid status

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module under test	TwinEV module	**Resp.**	*ETRAID*
Module requirement	*RU03.012*		
Test environment	TwinEV module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	TwinEV module has to be connected		
Dependencies	TC03.007		
Steps	1. User TwinEV module connects to interoperability platform through a NATS client 2. TwinEV subscribe to subject 'RESMODULE.notif.GridInfo'		
Pass criteria	TwinEV module receives a message with subject 'RESMODULE.notif.GridInfo'and content {"code": 200, "data": GridData}		
Suspension criteria	TwinEV module doesn't receive the message or it gets an error		
Results	TwinEV module can considerate the grid information to operate		

Name	TC03.017. Send minimal and maximal power to charge points		
Module under test	TwinEV module	**Resp.**	*ETRAID*
Module requirement	*R03.013*		
Test environment	TwinEV module, CPO, Charge points		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Features to be tested	Control
Features not to be tested	
Preparation	
Dependencies	TC03.007, TC03.001
Steps	1. Log into TwinEV for grid operators 2. Select a charge point where user can charge his/her vehicle 3. Fill the form with some valid values and click on "Save" 4. Log into TwinEV for drivers 5. Reserve the charge point following test TC03.001. Reserve a charge point 6. Go to screen "Charge" 7. Start the charge session 8. Check the charge curve
Pass criteria	• The charge curve does not exceed the limits imposed • The vehicle is charged according to the charge curve
Suspension criteria	• The charge curve exceeds the limits imposed in some points • The vehicle is not charged according to the charge curve
Results	The vehicle is charged at the desired level

Name	TC03.018. Send charging session costs		
Module under test	TwinEV module	**Resp.**	*ETRAID*
Module requirement	*R03.014*		
Test environment	TwinEV module, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Preparation	
Dependencies	TC03.001
Steps	1. Log into the Interoperability platform console website. 2. Navigate to screen "Messages" 3. Subscribe to subject 'M7.notification.sessionEnd' 4. Log into TwinEV for drivers 5. Reserve the charge point following test TC03.001. Reserve a charge point 6. Go to screen "Charge" 7. Start the charge session 8. Wait for 5 minutes 9. Stop the session 10. Pick up the vehicle 11. Check Interoperability platform console

Pass criteria	In the Interoperability platform console, a message with the following structure has been registered:
	{
	"stationId": String,
	"chargePoint": String,
	"timestamp": datetime('YYYY-MM-DDThh:mm:ss.sssZ'),
	"totalCost": float,
	"totalEnergySupplied": float
	"totalEnergyToGrid": float
	}

Suspension criteria	None message has been registered.
Results	The vehicle is charged at the desired level

Name	TC03.019. Load charging session costs		
Module under test	Transactive Energy module	**Resp.**	*WEC*

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	*R03.015*
Test environment	Transactive Energy module, Interoperability platform
Features to be tested	Communication
Features not to be tested	
Preparation	• Transactive Energy module has a client connected to Interoperability platform. • Transactive Energy module generates logs when it receives messages from Interoperability platform
Dependencies	*TC03.018*

Steps	1. Execute Transactive Energy module.
	2. Log into the Interoperability platform console website.
	3. Navigate to screen "Messages"
	4. Send a message with subject 'M7.notification.sessionEnd' and structure:
	{
	"stationId": String,
	"chargePoint": String,
	"timestamp": datetime('YYYY-MM-DDThh:mm:ss.sssZ'),
	"totalCost": float,
	"totalEnergySupplied": float
	"totalEnergyToGrid": float
	}

Pass criteria	Transactive Energy module has written in its logs a message with structure
	{
	"stationId": String,
	"chargePoint": String,
	"timestamp": datetime('YYYY-MM-DDThh:mm:ss.sssZ'),
	"totalCost": float,
	"totalEnergySupplied": float
	"totalEnergyToGrid": float
	}
	and same values as those in the message

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Suspension criteria	Transactive Energy module has not written any log message
Results	Transactive Energy module updates calculations according to the message

Test cases for UC04

Name	TC04.001 - Check readings availability		
Module under test	Smart Meter Oracle	**Resp.**	WEC
Module requirement	R04.001		
Test environment	Raspberry PI module,		
Features to be tested	Functionality for Smart meters to be exporting readings to Raspberry Pi devices		
Features not to be tested	-		
Preparation	A smart meter (EmonPI) from the same model as used across the pilot sites must be connected to a circuit.		
Dependencies	Smartmeter EmonPI		
Steps	9. Establish connection between the smart meter and the circuit 10. Confirm meter data is recorded and formatted correctly in the local database		
Pass criteria	The data are successfully transmitted, retrieved and formatted by the Oracle application		
Suspension criteria	1. The data are not successfully transmitted 2. The data are not successfully validated		
Results	Data formatted and ready to be transmitted to the Transactive Energy Blockchain platform.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Name	TC04.002 – Transmitting meter data to the Transactive Energy Blockchain platform.		
Module under test	Transactive Energy Blockchain platform.	**Resp.**	WEC
Module requirement	R04.002		
Test environment	Back-end environment of the Transactive Energy Blockchain platform.		
Features to be tested	Functionality of the Oracle application and Smart meter Smart Contracts		
Features not to be tested	-		
Preparation	Blockchain network deployed consisting of 3 nodes. Smart Meter Smart Contract deployed		
Dependencies	-		
Steps	1. Oracle application post meter data to Smart Contract on set interval of 5 minutes 2. Smart Contract post meter data on Blockchain ledger 3. Query the ledger to validate synchronisation with smart meter		
Pass criteria	The data are successfully transmitted, posted, retrieved and validated from the Blockchain network		
Suspension criteria	1. The data are not successfully transmitted to the Smart Contract 2. Ledger data is not synchronized with the Smart meter		
Results	Data recorded successfully on the Blockchain ledger		

Name	TC04.003 – Storage of energy or consumption of stored energy must burn, transfer or mint kWh tokens		
Module under test	Transactive Energy Blockchain platform.	**Resp.**	WEC
Module requirement	R04.003		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Test environment	Back-end environment of the Transactive Energy Blockchain platform.
Features to be tested	Function of Smart Contracts for kWh tokens
Features not to be tested	-
Preparation	kwH Smart contracts deployed. Simulation of energy storage and consumption set-up by running a model through the Smart Meter Oracle application
Dependencies	-
Steps	1. Create data model to feed to though the Oracle application 2. Deploy kwH smart contracts 3. Configure wallets to receive and send tokens 4. Feed data through the Oracle application 5. Validate the correct execution of workflows
Pass criteria	The data are successfully transmitted, posted, retrieved and validated from the Blockchain network. Tokens are minted, burned or transferred between wallets by the Smart Contracts
Suspension criteria	Tokens are not minted, burned or transferred correctly
Results	1. The data model simulates a storage of energy event, 2. The meter data is recorded on the Blockchain ledger and triggers the minting of tokens respective to the volume of kWh stored 3. The tokens are transferred to the recipient wallet 4. The data model simulates a usage of stored energy event 5. The meter data is recorded on the Blockchain ledger and triggers the burning of tokens respective to the volume of kWh stored 6. The data model simulates a transfer of kWh 7. The meter data is recorded on the Blockchain ledger. The Smart contract transfers kWh tokens from the sending wallet to the receiving wallet

Name	TC04.004. Settled of RPOA

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module under test	Transactive Energy Blockchain platform.	**Resp.**	WEC
Module requirement	R04.005		
Test environment	Back-end environment of the Transactive Energy Blockchain platform.		
Features to be tested	Settle meter data		
Features not to be tested	-		
Preparation	Deploy the validation model in the Oracle application to query the ledger and validate with the meter readings and battery State of Charge readings.		
Dependencies	-		
Steps	1. Deploy the settle Smart Contracts 2. Configure the Oracle application to perform validation tasks on interval 3. Configure the Oracle application to execute correction posts to the Blockchain ledger		
Pass criteria	The data on the meter and the ledger is validated and the result is negative, i.e. not in synchronization. The Application applies a correction.		
Suspension criteria	1. The correction is executed incorrectly 2. The correction is not executed 3. The correction results in non-synchronization		
Results	Data across smart meters is settled correctly and corrected when running out of synch due to slippage.		

Name	TC04.005. Submission of sell orders		
Module under test	Transactive Energy Blockchain platform.	**Resp.**	WEC
Module requirement	R04.006, R04.007, R04.008		
Test environment	Back-end environment of the Transactive Energy Blockchain platform.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	HEMS Module Direct battery instruction by transactive energy module
Features to be tested	Execute a P2P sale of kwH
Features not to be tested	-
Preparation	1. Set-up data model and environment to simulate a test of energy transfer between 2 batteries 2. Deploy P2P sale Smart Contract
Dependencies	Ability to integrate with the HEMS or Interface of Smart batteries
Steps	1. Deploy a reference front-end to instruct buy and sell orders 2. Integrate with the HEMS to instruct on scheduling of energy transfer events 3. Integrate with battery interface to instruct on charge and discharge of energy 4. Place a sell order 5. Accept the sell order 6. Execute the order 7. Deliver the energy 8. Transfer kWh token and TwinERGY token between buyer and seller
Pass criteria	1. The sell order is placed successfully and contract posted on the Blockchain. KwH tokens are deposited into the Smart Contract. 2. The Buy order is placed successfully and TwinERGY tokens deposited into the Smart Contract 3. The transfer of Energy is scheduled in the HEMS or with the battery interface directly 4. The transfer of energy is executed and Smart meter data is recognized by the P2P Smart Contract 5. Once the transfer is completed the kWh and TwinERGY tokens are released by the P2P Smart Contract and deposited in the respective wallets.
Suspension criteria	1. The chain of Smart Contracts are not able to fully execute and the transfer becomes stuck 2. The transfer of energy cannot be scheduled correctly 3. The transfer of energy is not executed correctly.
Results	Transactive energy platform submits sell orders to the P2P sale logic smart contract contracts. When a sale is completed the TEP instruct the HEMS to schedule an energy transfer between Smart Batteries. When the transfer

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	is completed the P2P sale logic smart contract transfers kWh and TwinERGY tokens from and to the buyer and seller respectively.

Name	TC04.006. Instruction on the Smart Contract to mint and distribute DER tokens		
Module under test	Transactive Energy Blockchain platform.	**Resp.**	WEC
Module requirement	R04.009		
Test environment	Back-end environment of the Transactive Energy Blockchain platform. Social module		
Features to be tested	Reward Prosumers with DER tokens		
Features not to be tested	-		
Preparation	1. Integrate with the Social module to receive messages instructing the awarding of DER tokens to prosumers 2. Deploy the DER Smart Contract		
Dependencies			
Steps	1. Listen to messages from the Social module 2. On receiving a message post an instruction to the DER Smart Contract 3. DER Smart contract mint DER tokens and sends to the wallet address of the Prosumer		
Pass criteria	1. The messages from the Social module are consumed correctly 2. The instruction to the DER Smart Contract is successful 3. The Smart contract mints DER tokens and sends to the recipient		
Suspension criteria	1. The message is not received 2. Smart contract is not able to execute		
Results	Prosumers are rewarded with DER tokens on instruction of the Social module		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Name	TC04.007. Transmission of LEM price to the LEM price – logic		
Module under test	Transactive Energy Blockchain platform.	**Resp.**	WEC
Module requirement	R04.007		
Test environment	Back-end environment of the Transactive Energy Blockchain platform. LEM price application		
Features to be tested	Broadcast and record LEM price		
Features not to be tested	-		
Preparation	1. Integrate the LEM price algorithm with the TEM		
Dependencies	-		
Steps	1. Consume data input from LEM price application 2. Post LEM price data to LEM price Smart Contract 3. Record LEM price on Blockchain ledger 4. Post LEM price to NATS server		
Pass criteria	LEM pricing is consumed correctly, posted on the Blockchain and a message is sent to the NATS server		
Suspension criteria	LEM price is not posted on the Blockchain LEM price is not sent to the NATS server		
Results	LEM price is posted on the Blockchain ledger and broadcasted to the NATS server.		

Test cases for UC05

Name	TC05.001 Initialise the module

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module under test	RES & DER Management Module	**Resp.**	TH OWL
Module requirement	R05.001		
Test environment	RES & DER Module Server		
Features to be tested	Initialisation of the software		
Features not to be tested			
Preparation	RES & DER module must be ready on the computer for operation		
Dependencies			
Steps	1. Set a proper brake mark 2. Start module in debugging mode 3. When the program stops, check the initialised data		
Pass criteria	- The correct pilot-site data is loaded - The module can continue to run without errors		
Suspension criteria	No wrong data about PV or wind is loaded		
Results	The module has loaded the pilot site data and is ready to download weather information		

Name	TC05.002 Download weather information		
Module under test	RES & DER Management Module	**Resp.**	TH OWL
Module requirement	R05.002		
Test environment	RES & DER Module Server		
Features to be tested	Correct download of the weather forecast		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Features not to be tested	
Preparation	1. RES & DER module must be ready on the computer for operation 2. A suitable source for weather forecast must be available (SUITE5 DB, openweather API as fall-back)
Dependencies	SUITE5 DB, openweather API as fall-back
Steps	1. Set a proper brake mark 2. Start module in debugging mode 3. When the program stops, check the initialised weather-data
Pass criteria	The correct weather forecast data is loaded
Suspension criteria	No wrong data is loaded
Results	The module has loaded the correct weather information and is ready to calculate the estimated RES power generation

Name	TC05.003 Test the ICC implementation		
Module under test	RES & DER Management Module	**Resp.**	TH OWL
Module requirement	R05.003		
Test environment	RES & DER Module Server		
Features to be tested	Incentive Curve calculation		
Features not to be tested			
Preparation	RES & DER Module must be ready on the computer for operation		
Dependencies	Previous tests have to be successful		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Steps	1. Set a proper brake mark 2. Start module in debugging mode 3. When the program stops, check the calculated ICC printed in the terminal
Pass criteria	ICC values are correct
Suspension criteria	Error messages, implausible results
Results	The module has calculated the correct ICC

Name	TC05.004 Test the NATS communication		
Module under test	RES & DER Management Module	**Resp.**	TH OWL
Module requirement	R05.004		
Test environment	RES & DER Module Server		
Features to be tested	Initialisation of the software		
Features not to be tested	NATS server itself		
Preparation	DER module must be ready on the computer for operation		
Dependencies	NATS server		
Steps	1. Start module 2. Check, if the ICC is uploaded properly		
Pass criteria	ICC is uploaded properly		
Suspension criteria	No wrong data is loaded		
Results	1. The module has successfully sent the data to the NATS server, where it is available for other modules		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Test cases for UC06

Name	TC06.001. Test DR and EL (Energy Light) signal calculation		
Module under test	RES & DER Management Module	**Resp.**	TH OWL
Module requirement	R06.001		
Test environment	RES & DER Management Module server		
Features to be tested	DR-signal and EL-signal calculation		
Features not to be tested			
Preparation	Running RES & DER Management Module		
Dependencies			
Steps	1. Start the program 2. In the terminal output, check if energy light is set correctly (It should say "Energy Light set to green" or similar messages)		
Pass criteria	Correct message is displayed		
Suspension criteria	Error, wrong output		
Results	• Energy Light switches state • Energy Light keeps state		

Name	TC06.002. Send DR signal via NATS		
Module under test	DER Module	**Resp.**	TH OWL

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	R06.002
Test environment	RES & DER Management Module Server, NATS Server
Features to be tested	NATS implementation of the module
Features not to be tested	NATS server itself
Preparation	Running module
Dependencies	
Steps	1. Start module 2. Check, if data is sent to the NATS server correctly
Pass criteria	Successful data transfer
Suspension criteria	Errors
Results	The Incentive Curve and the DR-Signal is sent to the Interoperability Platform and is now accessible to other modules for further usage

Name	TC06.003. Access EL-signal via NATS		
Module under test	DER Module	**Resp.**	TH OWL
Module requirement	R06.003		
Test environment	Energy Light (hard- or software)		
Features to be tested	Download of the EL-Signal		
Features not to be tested			

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Preparation	RES & DER Management module up and running
Dependencies	
Steps	1. Start Energy Light 2. Check, if the correct state is assumed
Pass criteria	Energy Light switches correctly
Suspension criteria	Energy light does not switch
Results	Successful operation

Name	TC06.004. Test refreshing of the EL-signal		
Module under test	DER Management Module	**Resp.**	TH OWL
Module requirement	R06.004		
Test environment	Energy Light (hard- or software)		
Features to be tested	Communication between Energy Light and NATS server		
Features not to be tested			
Preparation	DER Module running		
Dependencies			
Steps	1. Start the Energy Light 2. Wait for the next state switch 3. Check, if the colour changes		
Pass criteria	Correct colour change		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Suspension criteria	No colour change
Results	Correct change of state

Test cases for UC07

Name	TC07.001. iSCAN retrieving data from Digital Twin Platform		
Module under test	iSCAN – Social Network Module	**Resp.**	IES
Module requirement	R07.001		
Test environment	iSCAN, Digital Twin Platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	iSCAN and Digital Twin Platform seamless connectivity, proper algorithms to calculate forecasts and disaggregation of profiles		
Dependencies			
Steps	1. Digital Twin Platform collects raw data from CDMP 2. Digital Twin Platform creates processed data (forecasts etc.) 3. Digital Twin Platform communicates processed data to iSCAN		
Pass criteria	Successful API Call obtaining requested data		
Suspension criteria	-		
Results	After obtaining the requested input, SN module knows if the message was sent or something was wrong		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	Consumption Data Accessible through Social Network Module UI

Name	TC07.002. iSCAN publishing dataset through NATS		
Module under test	iSCAN	**Resp.**	IES
Module requirement	R07.002		
Test environment	iSCAN, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	iSCAN has a client connected to Interoperability platform		
Dependencies			
Steps	1. iSCAN prepares datasets (disaggregated profiles, optimized net profiles, DT model of community) 2. iSCAN publishes new occurrences of the dataset on NATS server		
Pass criteria	Successful API Call		
Suspension criteria	NATS server is down.		
Results	iSCAN dataset are made available on NATS for multiple listeners.		

Name	TC07.003. Social Network Module subscribes (via NATS) and retrieves disaggregated profiles, optimized net profiles, DT model of community,		
Module under test	Social Network Module	**Resp.**	ED

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	R07.003
Test environment	Social Network Module, iSCAN, Interoperability platform
Features to be tested	Communication
Features not to be tested	
Preparation	SN module has a client connected to Interoperability platform (or directly to iSCAN)
Dependencies	
Steps	1. SN module connects to interoperability platform through SCAN API or NATS client 2. Social Network Module subscribes to iSCAN as a listener to iSCAN messages
Pass criteria	Successful API Call
Suspension criteria	NATS server down.
Results	Successful subscription

Name	TC07.004. Social Network Module retrieves (processed data) information profiles from NATS		
Module under test	iSCAN – Social Network Module	**Resp.**	ED
Module requirement	R07.004		
Test environment	Social Network Module, iSCAN, Interoperability platform		
Features to be tested	Communication		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Features not to be tested	
Preparation	SN module has a client connected to Interoperability platform (or directly to iSCAN)
Dependencies	
Steps	1. SN module connects to interoperability platform through SCAN API or NATS client 2. Social Network Module performs an API call to iSCAN
Pass criteria	Successful API Call
Suspension criteria	The API Call doesn't show any message or shows an error message, Social Network Module will continue to present obsolete analytics/results (consecutive calls will be executed till data are retrieved)
Results	After obtaining the requested input, SN module knows if the message was sent or something was wrong Forecasting Data Accessible through Social Network Module UI

Name	TC07.005. Comfort well-being publishes at NATS data analytics for individuals		
Module under test	Comfort well being module	**Resp.**	UoP
Module requirement	R07.005		
Test environment	Comfort well-being, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	Comfort well-being module has a client connected.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Dependencies	
Steps	1. Comfort well-being module connects to interoperability platform through a NATS client, 2. Makes available data analytics for households
Pass criteria	Successful API Call
Suspension criteria	NATS server is down.
Results	Successful publishing of data analytics

Name	TC07.006. Interoperation of assets with controllers		
Module under test	- (assets only)	**Resp.**	Pilot leaders
Module requirement	R07.006		
Test environment	Assets controllers/Pis		
Features to be tested	Communication		
Features not to be tested			
Preparation	Assets controllers/Pis proper installation		
Dependencies	Availability of asset to expose APIs for controllers' communication.		
Steps	Expose APIs in both sides (asset and the Controller)		
Pass criteria	Successful API Call		
Suspension criteria	The API Call doesn't show any message or shows an error message.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Results	Successful exchange of status level HTTP 200 OK

Name	TC07.007. Asset Controllers/Pi's assumes connectivity with HEMS		
Module under test	- Assets and their controllers	**Resp.**	Pilot leaders
Module requirement	R07.007		
Test environment	HEMS/Controllers		
Features to be tested	Communication		
Features not to be tested			
Preparation	HEMS and controllers' successful installation.		
Dependencies			
Steps	Expose APIs in both sides HEMS/Controllers		
Pass criteria	Successful API Call		
Suspension criteria	The API Call doesn't show any message or shows an error message		
Results	Successful exchange of status level HTTP 200 OK		

Name	TC07.008. HEMS to HEMS Gateway connectivity
Module under test	HEMS

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	R07.008	**Resp.**	Pilot leaders
Test environment	Social Network Module, iSCAN, Interoperability platform		
Features to be tested	Communication		
Features not to be tested			
Preparation	Successful installation of both components (HEMS/HEMS Gateway)		
Dependencies			
Steps	Expose APIs in both sides HEMS/HEMS Gateway		
Pass criteria	Successful API Call		
Suspension criteria	The API Call doesn't show any message or shows an error message.		
Results	Successful exchange of status level HTTP 200 OK		

Name	TC07.009. HEMS Gateway publishes to CDMP		
Module under test	CDMP/ (HEMS)	**Resp.**	Pilot leaders
Module requirement	R07.009		
Test environment	CDMP/HEMS		
Features to be tested	Communication		
Features not to be tested			

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Preparation	Pilot leader have created proper accounts for CDMP
Dependencies	
Steps	1. Create ingest jobs 2. Expose data with proper access policies to modules 3. Make available the restAPI endopoint
Pass criteria	Successful API Call
Suspension criteria	CDMP is down.
Results	Raw data from the field are made available on CDMP for modules exploitation.

Name	TC07.010. Social Network Module GUI requests for competitions badges/coins, and monitoring features.		
Module under test	Social Network Module GUI– Social Network Module	**Resp.**	ED
Module requirement	R07.010		
Test environment	Social Network Module – Social Network Module GUI		
Features to be tested	Communication		
Features not to be tested			
Preparation	Social Network Module GUI invokes Social Network Module back-end to fetch data related to household performance (monitoring functionalities, recommendations etc)		
Dependencies	With R07.001-R07.009, since the data obtained there are necessary for the analytics calculations and representation		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Steps	Social Network Module GUI invokes Social Network Module back-end to fetch data related to household performance (monitoring functionalities, recommendations etc)
Pass criteria	Successful API Call
Suspension criteria	-
Results	After obtaining the requested input, visualisation on the GUI.

Name	TC07.0011. Sync and preview information in regard to the balance of the consumers digital wallet and the TwinERGY point earned		
Module under test	TEM/Social Network Module	**Resp.**	ED-WEC
Module requirement	R07.011		
Test environment	Social Network Module, TEM, NATS server		
Features to be tested	Communication		
Features not to be tested			
Preparation	SN module has a client connected to NATS Interoperability platform		
Dependencies			
Steps	1. Social Network Module module connects to interoperability platform through NATS client 2. Social Network Module performs an API call to TEM (and vice versa)		
Pass criteria	Successful API Call		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Suspension criteria	The API Call doesn't show any message or shows an error message, Social Network Module will continue to present obsolete results (consecutive calls will be executed till data are retrieved)
Results	After obtaining the requested input, SN module knows if the message was sent or something was wrong Send Equipment Data Accessible through Social Network Module UI

Name	TC07.0012. Social Network Module retrieving raw data from CDMP		
Module under test	TEM/Social Network Module	**Resp.**	ED-WEC
Module requirement	R07.012		
Test environment	Social Network Module, CDMP		
Features to be tested	Communication		
Features not to be tested			
Preparation	Create an endpoint for each user.		
Dependencies			
Steps	Social Network Module makes a get request to the corresponding API endpoint to retrieve raw data.		
Pass criteria	Successful API Call		
Suspension criteria	The API Call doesn't show any message or shows an error message, Social Network Module will continue to present obsolete results (consecutive calls will be executed till data are retrieved)		
Results	Raw data collected		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Name	TC07.0013. Social Network Module retrieving data from iSCAN		
Module under test	TEM/Social Network Module	**Resp.**	ED-WEC
Module requirement	R07.012		
Test environment	Social Network Module, iSCAN		
Features to be tested	Communication		
Features not to be tested			
Preparation	Social Network Module routine for requesting data from iSCAN		
Dependencies	New occurring data are timely uploaded on iSCAN		
Steps	Social Network Module makes a get request to the corresponding API endpoint to retrieve raw data.		
Pass criteria	Successful API Call		
Suspension criteria	The API Call doesn't show any message or shows an error message, Social Network Module will continue to present obsolete results (consecutive calls will be executed till data are retrieved)		
Results	Digital Twin Platform data collected and assessed at Social Network Module level.		

Test cases for UC08

Name	TC08.001 - Check readings availability		
Module under test	Consumer Wellbeing Module	**Resp.**	UoP

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	R08.001
Test environment	Back-end environment of Consumer Digital Twin (CDT)
Features to be tested	Communication, Functionality
Features not to be tested	-
Preparation	It is needed to have the wrist-wearable device operational & connected to the network
Dependencies	-
Steps	11. Establish connection between the local network and the wrist-wearable device. 12. Transmit the payload with corresponding data from the wrist-wearable device to the CDT's back-end infrastructure. 13. Retrieve each physiological and activity parameter from the transmitted payload.
Pass criteria	The data are successfully transmitted, retrieved and validated from the CDT.
Suspension criteria	3. The data are not successfully transmitted to the CDT. 4. The data are not successfully validated from the CDT.
Results	Data to be applied on the algorithm for Thermal Comfort assessment.

Name	TC08.002 - Check readings availability		
Module under test	Consumer Wellbeing Module -	**Resp.**	UoP
Module requirement	R08.002		
Test environment	Back-end environment of Consumer Digital Twin (CDT)		
Features to be tested	Communication, Functionality		
Features not to be tested	-		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Preparation	It is needed to have the wrist-wearable device operational & connected to the network
Dependencies	-
Steps	1. Establish connection between the local network and the wrist-wearable device. 2. Transmit the payload with corresponding data from the wrist-wearable device to the CDT's back-end infrastructure. 3. Transmit the payload with corresponding data from the wrist-wearable device to the CDT's back-end infrastructure. 4. Retrieve each well-being parameter from the transmitted payload.
Pass criteria	The data are successfully transmitted, retrieved and validated from the CDT.
Suspension criteria	5. The data are not successfully transmitted to the CDT. 6. The data are not successfully validated from the CDT.
Results	Data to be applied on the algorithm for Well-being status assessment.

Name	TC08.003 - Check input availability		
Module under test	Consumer Wellbeing Module	**Resp.**	UoP
Module requirement	R08.003		
Test environment	Back-end environment of Consumer Digital Twin (CDT)		
Features to be tested	Visualization and analysis		
Features not to be tested	-		
Preparation	Access on the Consumer Digital Twin front-end and back-end.		
Dependencies	-		
Steps	1. Insert the values that reflect the clothing insulation for each season through the drop-down UI lists.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	2. Click "Update" to save the defined values of clothing insulation for each season.
Pass criteria	The consumer's clothing insulation is successfully determined from the end-user through the front-end of the CDT and stored on the back-end.
Suspension criteria	1. The data are not successfully displayed from the CDT. 2. The data are not successfully saved by the CDT.
Results	Data to be applied on the algorithm for Thermal Comfort assessment.

Name	TC08.004 - Check execution of algorithms		
Module under test	Consumer Wellbeing Module	**Resp.**	UoP
Module requirement	R08.004		
Test environment	Back-end environment of Consumer Digital Twin (CDT)		
Features to be tested	Visualization and analysis		
Features not to be tested	-		
Preparation	Access on the Consumer Digital Twin back-end.		
Dependencies	-		
Steps	No steps required, since the algorithms are scheduled to be automatically executed and their outputs are displayed on the front-end of the Consumer Digital Twin		
Pass criteria	The algorithms for the assessment of Thermal Comfort and Well-being status are successfully executed.		
Suspension criteria	The execution of the algorithms fails or empty data are displayed.		
Results	Thermal Comfort level and well-being status of the individual.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Name	TC08.005 - Check execution of algorithms		
Module under test	Consumer Wellbeing Module	**Resp.**	UoP
Module requirement	R08.005		
Test environment	Back-end environment of Consumer Digital Twin (CDT)		
Features to be tested	Visualization and analysis, Functionality		
Features not to be tested	-		
Preparation	It is needed to have the wrist-wearable device operational & connected to the network and available datasets for the corresponding individual.		
Dependencies	-		
Steps	No steps required, since the algorithms are scheduled to be automatically executed and their outputs are displayed on the front-end of the Consumer Digital Twin		
Pass criteria	Data related to the consumer's preferences, thermal comfort, and well-being status are successfully parsed by the Demand Response algorithm.		
Suspension criteria	1. Data related to the consumer's preferences are not available. 2. Data related to the consumer's thermal comfort are not available. 3. Data related to the consumer's well-being status are not available.		
Results	Data related to the consumer's preferences, thermal comfort, and well-being status.		

Test cases for UC09

Name	TC09.001: Creation of the iSCAN Projects

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R09.001		
Test environment	Azure on the cloud, python, iSCAN		
Features to be tested	Creation of environment		
Features not to be tested	Population of the iSCAN channels with Data, linking the data sources within the Core Data Management Platform to the channels		
Preparation	The Core Data Management Platform is active and available. The user has access to the predefined JSON file and has the required information for the building description and data channels relevant to the building.		
Dependencies	NA		
Steps	1. The user accesses the predefined JSON file 2. The user inputs the required fields (building description, location, list of data inputs required, etc) 3. The JSON file is saved and the user inputs the file path/hyperlink as an input to the algorithm 4. The algorithm accesses the file and creates the iSCAN project for the building using the building description and location from the JSON file. 5. The algorithm creates the individual data channels as specified by the user in the JSON file and is saved to define the iSCAN project for the building 6. The algorithm accesses the building location in the JSON to define the project location and establish the relevant weather file 7. Once the building level iSCAN project is defined, the algorithm creates the community level project 8. The Energy Tariff Channel is created 9. The community level project creates dedicated channels for aggregated demand for each individual building, optimised rescheduling of demand, cost analysis, renewable generation, flexibility assessment and end-user acceptance ratio. 10. The iSCAN project is now set up.		
Pass criteria	The iSCAN project is set up at both building and community level with all specified channels created and linked to the relevant channels in the Core Data Management Platform.		
Suspension criteria	• The algorithm can't access the JSON file • The algorithm can't read the JSON file • The algorithm fails to initiate the creation of the iSCAN project		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	• The algorithm is unable to create the specified iSCAN channels • The creation of the building level iSCAN project does not trigger the creation of the Community level iSCAN project • The required Community level iSCAN channels are not created • The channels are not linked to the relevant data source within the Core Data Management Platform.
Results	The system has validated the creation of the environment

Name	TC09.002: Data Channels linked to Core Data Management Platform		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R09.002		
Test environment	Azure on the cloud, python, iSCAN, Core Data Management Platform (DMP)		
Features to be tested	The user can link the iSCAN channels to the relevant data source within the Core Data Management Platform		
Features not to be tested	Automated Population of the iSCAN Channels with Data		
Preparation	The iSCAN project has been created and the relevant channels in place. The DMP is available and contains data for each data source.		
Dependencies	TC09.001		
Steps	1. The user identifies the individual iSCAN channel and its associated API 2. The user accesses the Core Data Management Platform and identifies the relevant data source within the DMP 3. The user provides the data source with the API link to push data to the iSCAN channel 4. The user then initiates the transfer to verify the connection 5. The iSCAN channel is populated with the data required.		
Pass criteria	The user is able to link the iSCAN channel to the relevant data source via API and can initiate the population of each iSCAN data with data.		
Suspension criteria	• The user is unable to link the relevant channel with the data source • The data transfer fails		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Results	The system has validated the data channels connections

Name	TC09.003: Transactive Energy Market (TEM) Connected		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R09.003		
Test environment	Azure on the cloud, python, iSCAN, TEM		
Features to be tested	The user can link the iSCAN channels to the TEM		
Features not to be tested	Automated Population of the iSCAN Energy Tariff Channel with Data		
Preparation	The iSCAN project has been created and the relevant channels in place. The TEM is active.		
Dependencies	TC09.001		
Steps	1. The user identifies the individual iSCAN Energy Tariff and its associated API 2. The user provides the iSCAN API to the TEM to allow the module to push updated tariff data to the iSCAN channel 3. The user then initiates the transfer to verify the connection 4. The iSCAN channel is populated with the data required.		
Pass criteria	The user is able to link the iSCAN channel to the relevant data source via API and can initiate the population of each iSCAN data with data.		
Suspension criteria	• The user is unable to link the relevant channel with the TEM • The data transfer fails • The TEM fails to initiate when called by the user during verification.		
Results	The system has validated the Transactive Energy Market (TEM) connections		

Name	TC09.004: Automated Algorithm Initiation & Data Collection		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module requirement	R02.004
Test environment	Azure on the cloud, python, iSCAN, DMP, TEM, VE, StRoBe Library, iVN
Features to be tested	The algorithm is automatically initiated and the data transferred to the relevant iSCAN channels
Features not to be tested	Optimisation of the Demand Profiles
Preparation	The iSCAN project has been set up and all relevant channels created. The API for each of the channels have been mapped to the data sources within the DMP and TEM. The StRoBe library and Building Digital Twin are available.
Dependencies	TC09.001, TC09.002, TC09.003
Steps	1. At a predefined time (18:00) the algorithm is initiated and begins to run 2. The algorithm accesses the predefined JSON file 3. The algorithm verifies that the iSCAN project and all relevant channels are created 4. The algorithm calls on the DMP and data for the previous 24hours is pushed form the DMP to the iSCAN channel via API 5. The algorithm calls on the TEM and forecasted tariff data for the next 24 hours are pushed to the iSCAN channel 6. The iVN engine is called and generates a RES profile for the relevant time period (one day ahead) 7. The algorithm validates that all necessary data has been transferred to the iSCAN project 8. Where data is missing, the Building Digital Twin and StRoBe library are initiated and used to populate the missing data channels
Pass criteria	All iSCAN channels are now populated with the most up to date data
Suspension criteria	• Algorithm fails to initiated at the defined time • Communication error with the DMP or the TEM • Missing data is not identified and sourced from alternate sources • iSCAN channels fail to populate
Results	The system has validated the Algorithm Initiation automation & Data Collection

Name	TC09.005: Demand Profile Optimisation

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R09.005		
Test environment	Azure on the cloud, python, iSCAN,		
Features to be tested	The algorithm is automatically initiated and the data transferred to the relevant iSCAN channels		
Features not to be tested	Optimisation of the Demand Profiles		
Preparation	The iSCAN project has been set up and all relevant channels created. The API for each of the channels have been mapped to the data sources within the DMP and TEM. The StRoBe library and Building Digital Twin are available.		
Dependencies	TC09.001, TC09.002, TC09.003		
Steps	1. The iSCAN API is called to retrieve building level time-series data for the last 30 days 2. This time-series data is then analysed through integrated Python libraries to identify the pattern of use of smart appliances over the defined time-period 3. The process generates the standard profile of use for the next day and determines the shiftable loads based on user input 4. If the appliance use is shiftable, a preferred time period of use is defined based on previous usage patterns or user preferences 5. The forecast profiles are resampled at the same time resolution 6. The optimisation algorithm assesses and inputs the resampled forecast profiles and schedules the loads based on RES production and cost 7. The algorithm optimises operational cost at the building level individually 8. The algorithm identifies the list of starting point in minutes over the course of the day for each individual smart appliance 9. The demand profile is reconstructed based on the starting point for each appliance. This is the optimised demand profile 10. The optimised demand profile for each building is then returned to the corresponding aggregated iSCAN data channel The individual building/customer data channel is then aggregated to derive the community based optimised profile.		
Pass criteria	• The building and community level optimised demand profiles are defined.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	Demand response (implicit & explicit) are calculates
Suspension criteria	• The python libraries aren't initiated • The profile is not optimised based on cost and RES production The optimised profiles are not returned.
Results	The system has validated the Demand Profile Optimisation

Name	TC09.006: Explicit & Implicit Demand Response Results		
Module under test	Community Demand Flexibility Profiling Module	**Resp.**	IES
Module requirement	R09.006		
Test environment	Azure on the cloud, python, iSCAN, Social Network Module		
Features to be tested	The data transferred from the relevant iSCAN channels to the Social Network Module		
Features not to be tested	Optimisation of the Demand Profiles		
Preparation	The iSCAN project has been set up and all relevant channels created. The API for each of the channels have been mapped to the data sources within the DMP and Social Network Module. The StRoBe library and Building and Community Digital Twin are available.		
Dependencies	TC09.001, TC09.002, TC09.003, TC09.004, TC09.005		
Steps	1. The user identifies the individual iSCAN channels and its associated API 2. The user provides the iSCAN API to the Social module to allow the module to push updated results of Explicit and Implicit Demand Response from the iSCAN channel 3. The user then initiates the transfer to verify the connection 4. The data required is retrieved from scan		
Pass criteria	• The building and community level optimised demand profiles are defined. • Demand response (implicit & explicit) are calculated		

Suspension criteria	• The python libraries aren't initiated • The profile is not optimised based on cost and RES production
Results	The system has displayed the Explicit & Implicit Demand Response Results

Test cases for Core Data Management Platform

Name	TCCDMP01. File ingestion		
Module under test	Core Data Management Platform	**Resp.**	Suite5
CDMP requirement	RCDMP.001, RCDMP.007, RCDMP.008, RCDMP.009		
Test environment	Core Data Management Platform		
Features to be tested	Ingestion of files		
Features not to be tested	The rest of the platform's functionality, such as mapping, curation, metadata definition, search, retrieval, etc.		
Preparation	• A sample of the data to be imported in the form of a locally stored file is available • The actual data to be imported in the form of a locally stored file is available		
Dependencies	User has logged in in the platform successfully.		
Steps	1. Click on the Data Collection, click on + Create 2. Enter Name and Description, DATA INGESTER and DATA MAPPER are pre-selected 3. Click create 4. On the job created, click on the Data Ingester. 5. Select "File Upload" 6. Select one of the available formats (CSV or JSON 7. Click "Browse" next to the "Sample" and select the file you want to use as sample from the file browser window that opens.		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	8. Click "Browse" next to the "Upload File(s)" and select the file that contains the data, from the file browser window that opens. 9. Perform the rest of the steps.
Pass criteria	The data is available in the platform
Suspension criteria	The file is not ingested, and it cannot be retrieved.
Results	The data of the file ingested in the platform is available for the platform users that have access to it, to be located via the search functionality and retrieved via configurable APIs

Name	TCCDMP02. Ingestion via data provider's available API		
Module under test	Core Data Management Platform	**Resp.**	Suite5
CDMP requirement	RCDMP.001, RCDMP.007, RCDMP.008, RCDMP.009		
Test environment	Core Data Management Platform		
Features to be tested	Ingestion of data via external APIs		
Features not to be tested	The rest of the platform's functionality, such as mapping, curation, metadata definition, search, retrieval, etc.		
Preparation	• A 3rd party API, with all parameters such an authentication and parameters, is available		
Dependencies	User has logged in in the platform successfully.		
Steps	1. Click on the Data Collection, click on + Create 2. Enter Name and Description, DATA INGESTER and DATA MAPPER are pre-selected 3. Click create 4. On the job created, click on the Data Ingester. 5. Select "DATA PROVIDER'S AVAILABLE API" 6. Configure the authentication details 7. Configure the method and the URL to be used		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	8. Define request parameters and extra headers as needed 9. Configure the retrieval settings and setup the schedule 10. Click next 11. Select the fields to be used from the sample made available 12. Click finalize 13. Perform the rest of the steps.
Pass criteria	The data ingested via the API based on the schedule set, is available for the platform users that have access to it, to be located via the search functionality and retrieved via configurable APIs
Suspension criteria	The data is not ingested and cannot be retrieved.
Results	The data is available in the platform

Name	TCCDMP03. Retrieve data via APIs		
Module under test	Core Data Management Platform	**Resp.**	Suite5
Module requirement	RCDMP.002, RCDMP.003, RCDMP.004, RCDMP.005, RCDMP.006, RCDMP.010,R CDMP.011, RCDMP.012, RCDMP.013, RCDMP.014, RCDMP.015		
Test environment	Core Data Management Platform		
Features to be tested	Retrieval of data via APIs		
Features not to be tested	The rest of the platform's functionality, such as ingestion, mapping, curation, metadata definition, etc.		
Preparation	• A data asset is available in the platform and the user that performs the test has access to find and retrieve it based on the access policies set.		
Dependencies	User has logged in in the platform successfully.		
Steps	1. Click on Search 2. Utilizing the appropriate filters and free text search, identify the data asset of interest 3. Click "ADD TO QUERY RESULTS", click on Save and Proceed		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

	4. Set a Title and Description to save a query. 5. Select the fields of interest and define the query parameters, click on Save and Proceed 6. Check the results via the test run query, click on Save and Proceed 7. Copy the URL provided in the "Endpoint for using GET" 8. Utilize this URL in the module or in the DT, combined with a user created access token, to verify that the desired data is being retrieved correctly.
Pass criteria	The data is made available in the platform via an API
Suspension criteria	The data is available, or the data is wrong.
Results	The data that the platform user wants to retrieve is made available in the platform via an API and the modules/DTs can access it successfully.

Test cases for Identity Management Platform & Citizens Platform

Name	TCIdM.001. Login with existing user		
Module under test	IdM platform	**Resp.**	ETRAID
Requirement	RIdM.001, RIdM.005, RIdM.006, RIdM.007, RIdM.008, RIdM.009		
Test environment	Keycloak		
Features to be tested	User authentication		
Features not to be tested			
Preparation	A client application registered in the realm for TwinERGY includes a screen to login. There is a user registered in the realm for TwinERGY, with known credentials		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Dependencies	
Steps	1. The user goes to login in the client application 2. The user clicks on login 3. A dialog for inserting the credentials appears 4. The user inserts the credentials 5. The application validates the user credentials through IdM platform
Pass criteria	The application redirects to the main screen
Suspension criteria	The application doesn't redirect to the main screen and/or shows an error message
Results	The system has validated the user

Name	TCIdM.002. Login with non-existing user		
Module under test	Identity Manager management platform	**Resp.**	ETRAID
Requirement	RIdM.002		
Test environment	Keycloak		
Features to be tested	User authentication		
Features not to be tested			
Preparation	A client application registered in the realm for TwinERGY includes a screen to login.		
Dependencies			
Steps	1. The user goes to login in the client application 2. The user clicks on login 3. A dialog for inserting the credentials appears 4. The user inserts some invented credentials 5. The application validates the user credentials through IdM platform		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Pass criteria	The application shows an error message and stays in the login screen
Suspension criteria	The application redirects to the main screen and/or doesn't show an error message
Results	The system has validated the user

Name	TCIdM.003. Login with existing user with wrong password		
Module under test	Identity Manager management platform	**Resp.**	ETRAID
Requirement	RIdM.002		
Test environment	Keycloak		
Features to be tested	User authentication		
Features not to be tested			
Preparation	A client application registered in the realm for TwinERGY includes a screen to login. There is a user registered in the realm for TwinERGY, with known credentials		
Dependencies			
Steps	1. The user goes to login in the client application 2. The user clicks on login 3. A dialog for inserting the credentials appears 4. The user inserts the real username but wrong password 5. The application validates the user credentials through IdM platform		
Pass criteria	The application shows an error message and stays in the login screen		
Suspension criteria	The application redirects to the main screen and/or doesn't show an error message		
Results	The system has validated the user		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Name	TCIdM.004. Session token attributes		
Module under test	Identity Manager management platform	**Resp.**	ETRAID
Requirement	RIdM.003		
Test environment	Keycloak		
Features to be tested	User authentication		
Features not to be tested			
Preparation	A client application registered in the realm for TwinERGY includes a screen to login. There is a user registered in the realm for TwinERGY, with known credentials		
Dependencies	TCIdM.001		
Steps	1. The user goes to login in the client application 2. The user clicks on login 3. A dialog for inserting the credentials appears 4. The user inserts some the real username but wrong password 5. The application validates the user credentials through IdM platform 6. The application redirects to the main screen.		
Pass criteria	The application shows information in the token session about user		
Suspension criteria	The application doesn't show retrieved information about the user		
Results	The system has validated the user		

Name	TCIdM.005. Validation of existing users in another Realm		
Module under test	Identity Manager management platform	**Resp.**	ETRAID

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Requirement	RIdM.004
Test environment	Keycloak
Features to be tested	User authentication
Features not to be tested	
Preparation	A client application registered in the realm for TwinERGY includes a screen to login. There is a user registered in the realm for TwinERGY, with known credentials There is a second user registered in another realm to TwinERGY, with known credentials
Dependencies	TCIdM.001
Steps	1. The user goes to login in the client application 2. The user clicks on login 3. A dialog for inserting the credentials appears 4. The user inserts the credentials corresponding to the second user (belonging to the different realm) 5. The application validates the user credentials through IdM platform
Pass criteria	The application shows message of unknown user
Suspension criteria	The application navigates to the main screen or doesn't show an error message
Results	The system has validated the user

Test cases for Interoperability Platform

Name	TCIP.001. Connection with correct credentials		
Module under test	Interoperability platform	**Resp.**	ETRAID

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Requirement	RIP.001
Test environment	NATS
Features to be tested	Connection to NATS
Features not to be tested	Sending of data
Preparation	• The Interoperability Platform is online and running
Dependencies	

Steps	1. A module including the next piece of code in its start is executed:
	`const nc = NATS.connect({url: "accessURL", token: "accessToken", json:true});`
	`nc.on('connect', (c) => {`
	`console.log('Connected to Interoperability Platform');`
	`nc.close();`
	`});`
	`nc.on('error', (err) => {`
	`console.log('Error in connection to Interoperability Platform:');`
	`console.log(err);`
	`});`
	Being "accessURL" the url where the server is published and "accessToken" a valid and recognized token for connection.

Pass criteria	The module shows the message "Connected to Interoperability Platform"
Suspension criteria	The application doesn't show the success message or shows "Error in connection to NATS"
Results	The module has tried the connection

Name	TCIP.002. Connection with wrong credentials

Module under test	Interoperability platform	**Resp.**	ETRAID
Requirement	RIP.002		
Test environment	NATS		
Features to be tested	Connection to the Interoperability Platform		
Features not to be tested	Transmission of messages		
Preparation	The Interoperability Platform is online and running		
Dependencies			

Steps	1. A module including the next piece of code in its start is executed:
	`const nc = NATS.connect({url: "accessURL", token: "incorrectToken", json:true});`
	`nc.on('connect', (c) => {`
	`console.log('Connected to Interoperability Platform');`
	`nc.close();`
	`});`
	`nc.on('error', (err) => {`
	`console.log('Error in connection to Interoperability Platform:');`
	`console.log(err);`
	`});`
	Being "accessURL" the url where the server is published and "incorrectToken" an invalid token for the connection.		

Pass criteria	The module shows the message "Error in connection to NATS"		
Suspension criteria	The application doesn't show the error message or shows "Error in connection to Interoperability Platform"		
Results	The module has tried the connection		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

Name	TCIP.003. Reception of messages		
Module under test	Interoperability platform	**Resp.**	ETRAID
Requirement	RIP.003		
Test environment	NATS, NATS console		
Features to be tested	Transmission of messages		
Features not to be tested	Connection to the Interoperability Platform		
Preparation	The Interoperability Platform is online and running User is able to log into the NATS console		
Dependencies	RPI.001		

Steps	1. User logs into the NATS console
	2. User navigates to screen "Messages"
	3. User fills field "Subject to listen" with text "testTransmission"
	4. User clicks on button play.
	5. A module including the next piece of code in its start is executed:
	`const nc = NATS.connect({url: "accessURL", token: "accessToken", json:true});`
	`nc.on('connect', (c) => {`
	`console.log('Connected to Interoperability Platform');`
	`nc.publish("testTransmission",{"message": "This is a test message"});`
	`nc.close();`
	`});`
	`nc.on('error', (err) => {`
	`console.log('Error in connection to Interoperability Platform:');`
	`console.log(err);`
	`});`
	Being "accessURL" the url where the server is published and "accessToken" a valid token for the connection.		

Pass criteria	The NATS console shows a new line in its table with timestamp this moment, subject "testTransmission" and contain of the message "`{"message": "This is a test message"}`".
Suspension criteria	The table doesn't contain any of the message with that subject of it includes messages related to other subjects
Results	The module closes its connection after sending the message.

Name	TCIP.004. Subscription to subject (1)		
Module under test	Interoperability platform	**Resp.**	ETRAID
Requirement	RIP.004		
Test environment	NATS, NATS console		
Features to be tested	Transmission of messages		
Features not to be tested	Connection to the Interoperability Platform		
Preparation	The Interoperability Platform is online and running User is able to log into the NATS console		
Dependencies	RPI.001		

Steps	1. A module including the next piece of code in its start is executed:
	`const nc = NATS.connect({url: "accessURL", token: "accessToken", json:true});`
	`nc.on('connect', (c) => {`
	`console.log('Connected to Interoperability Platform');`
	`nc.subscribe("testTransmission",(msg) => {`
	`console.log(msg.message);`
	`nc.close();`		

	<pre>});
	});
	nc.on('error', (err) => {
	console.log('Error in connection to Interoperability Platform:');
	console.log(err);
	});</pre> Being "accessURL" the url where the server is published and "accessToken" a valid token for the connection.
	2. User logs into the NATS console
	3. User navigates to screen "Messages"
	4. User clicks on button "Send message"
	5. In the new dialog, users fills "Subject" with text "`testTransmission`" and "Message" with "`{"message": "This is a test message"}`"
	6. User clicks on button "Send".

Pass criteria	The module prints in this order the next messages: • "`Connected to Interoperability Platform`" • "`{"message": "This is a test message"}`"
Suspension criteria	The module does not print the second message or prints another message.
Results	The module closes its connection after receiving a message.

Name	TCIP.005. Subscription to subject (2)		
Module under test	Interoperability platform	**Resp.**	ETRAID
Requirement	RIP.004		
Test environment	NATS, NATS console		
Features to be tested	Transmission of messages		
Features not to be tested	Connection to the Interoperability Platform		

Twinergy has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 957736

<table>
<tr><td>Preparation</td><td>The Interoperability Platform is online and running
User is able to log into the NATS console</td></tr>
<tr><td>Dependencies</td><td>RPI.001</td></tr>
<tr><td>Steps</td><td>1. A module including the next piece of code in its start is executed:
<pre>const nc = NATS.connect({url: "accessURL", token: "accessToken", json:true});

nc.on('connect', (c) => {
  console.log('Connected to Interoperability Platform');
  nc.subscribe("testTransmission",(msg) => {
    console.log(msg.message);
    nc.close();
  });
});
nc.on('error', (err) => {
  console.log('Error in connection to Interoperability Platform:');
  console.log(err);
});</pre>
Being "accessURL" the url where the server is published and "accessToken" a valid token for the connection.
2. User logs into the NATS console
3. User navigates to screen "Messages"
4. User clicks on button "Send message"
5. In the new dialog, users fills "Subject" with text "<code>testTransmission</code>" and "Message" with "<code>{"message": "This is a test message"}</code>"
6. User clicks on button "Send".
7. User repeats steps 4 and 5 with "Subject" set to "<code>testTransmission2</code>" and "Message" with "<code>{"message": "This is a second test message"}</code>"</td></tr>
<tr><td>Pass criteria</td><td>The module prints in this order the next messages:
"<code>Connected to Interoperability Platform</code>"
"<code>{"message": "This is a test message"}</code>"</td></tr>
<tr><td>Suspension criteria</td><td>The module does not print the message with subject "testTransmission" or prints the message with subject "testTransmission"</td></tr>
<tr><td>Results</td><td>The module closes its connection after receiving a message.</td></tr>
</table>