



# Development of Energy Planning & Transition Decision Support Toolset\_v2

## **AUTHORS:**

Anastasis Tzoumpas, Athanasios Bachourmis, Katerina Drivakou, Ermis Vasileiou, Magda Foti, Kostas

Mylonas (UBE)

Petros Iliadis, Charalampos Papadopoulos, Panos Zestanakis, Niki Skopetou, Andreas Seitaridis, Nikolas

Tagkoulis, Konstantinos Kourtzanidis (CERTH) / Selma Causevic (TNO)



**H2020-LC-SC3-2018-2019-2020 / H2020-LC-SC3-2020-EC-ES-SCC**  
**EUROPEAN COMMISSION**

Innovation and Networks Executive Agency  
Grant agreement no. 957810

## PROJECT CONTRACTUAL DETAILS

<b>Project title</b>	IntegrAted SolutioNs for the DecarbOnization and Smartification of Islands
<b>Project acronym</b>	IANOS
<b>Grant agreement no.</b>	957810
<b>Project start date</b>	01-10-2020
<b>Project end date</b>	30-09-2024
<b>Duration</b>	48 months
<b>Project Coordinator</b>	João Gonalo Maciel (EDP) - JoaoGoncalo.Maciel@edp.com

## DOCUMENT DETAILS

<b>Deliverable no.</b>	D3.6
<b>Dissemination level</b>	Public
<b>Work package</b>	WP3
<b>Task</b>	T3.3
<b>Due date</b>	31-03-2023
<b>Actual submission date</b>	31-03-2023
<b>Lead beneficiary</b>	UBITECH ENERGY



<b>Version</b>	<i>Date</i>	<i>Beneficiary</i>	<i>Changes</i>
0.1	09/02/2023	UBE	First draft of ToC
0.2	15/02/2023	UBE	Refinement of ToC and allocation of work to the involved partners
0.3	08/03/2023	TNO	Contribution to chapter 4
0.4	13/03/2023	UBE	Elaboration in all chapters
0.5	17/03/2023	CERTH	Contribution to chapters 5 and 6
1.0	21/03/2023	UBE	Document finalization after quality review
2.0	28/03/2023	UBE	Final version after incorporation of reviewers' comments

“This publication reflects the author’s view only and the European Commission is not responsible for any use that may be made of the information it contains.”



# Executive Summary

The assessment of investment activities is a pillar in strategic management at any level. When investment activities involve the deployment of projects, project evaluation aims at determining the effects caused to foster the best allocation of scarce resources. In project evaluation, laws of economics are pivotal to determine the profitability of an initiative. However, projects that also have a social interest must be assessed considering a broader range of impacts. Assessment shall be performed before project deployment to forecast the effects expected (ex-ante), during project deployment to check the performance level and to identify corrective measures (in medias res) and after the time horizon of the project to verify the actual impacts generated (ex-post). Cost-benefit analysis (CBA) is the most acknowledged tool for financial viability assessment. CBA provides a wider project assessment by considering soft effects and intangible and non-monetizable impacts.

Therefore, the decision-makers across the energy value chain need a tool that can provide quantifiable insights supporting their potential investment decisions in clean and smart energy interventions. Toward that direction, the activities of WP3 aim at proposing an IANOS Energy Planning and Transition (IEPT) suite that supports the investments of the different stakeholders, providing a holistic approach that quantifies both the costs and the benefits of the IANOS interventions in the demonstration sites, i.e., Lighthouse and Fellow Islands of IANOS, as well as providing a tool that facilitates the fundraising campaigns.

To reach the objectives of WP3, D3.6 (which is the updated version of deliverable D3.5 related to Task 3.3) elaborates on the architecture and the main components of the IEPT suite. Through an API Gateway, the different modules which are deployed as dockerized microservices communicate and exchange the calculated values of Key Performance Indicators (KPIs). Those KPIs quantify the benefits of the smart grid interventions from a technical, economic and environmental perspective.



In addition, the overview, functionalities, technology stack and user interface (UI) of each internal module of the IEPT suite is thoroughly described. Briefly, those deployed microservices are:

- a) **Cost-Benefit analysis (CBA) tool:** The CBA tool was based on the JRC's and ENTSO-E's CBA methodologies and offers an analytical approach for the stakeholders/investors that provides a quantifiable insight regarding whether a smart grid intervention exceeds the existing baseline scenario in terms of costs and benefits.
- b) **Energy System Simulator (ESSIM) system modeler:** This tool simulates network balancing and the effects, thereof, in an interconnected hybrid energy system over a period of time. It takes as input the energy system defined in ESDL and calculates the optimal schedule of flexible producers and the effect of this schedule in terms of emissions, costs, load on the network, etc.
- c) **INTEgrated Energy Management (INTEMA.grid) Simulator:** It is an energy system modelling and simulation platform based on open-source non-proprietary tools. It is composed of the following main components: (a) dynamic power system simulations – based on the acausal, object-oriented, equation based Modelica language to conveniently model complex physical systems, (b) power optimization – ability to construct optimization problems using the Python optimization tool PyPSA and (c) power forecasting – utilizing open-source Machine Learning (ML) frameworks in Python, i.e., scikit-learn and Keras.
- d) **VERIFY District Platform (VERIFY-D):** A dedicated web platform for Life Cycle Analysis (LCA)/Life Cost Cycle (LCC) studies. Environmental and economic analyses are conducted through the computation of emissions extracted (life cycle environmental footprint), which depend on the technology applied in the grid, as well as the type of fuel mixture used to generate energy in the LH islands. Also, this tool computes the life cycle

costing based on the energy components that appear in the energy grid. Finally, the comparison between baseline and target scenarios in terms of the environmental gains and economic profits is analysed.

- e) **Equity crowdfunding tool:** This tool created a crowd-equity (or crowdfunding) platform, where actors (i.e., project investors, islanders and other key stakeholders) are able to register their foreseen projects and set a funding goal in return for equity, creating a fundraising campaign. The platform will ultimately enable fractional ownership of RE assets, supporting projects' fundraising and transactions either via normal FIAT currency or tokenized energy over Distributed Ledger Technology (DLT).

Finally, this deliverable elaborates on the way the IEPT suite is leveraged and employed for the evaluation of the Lighthouse Islands, as well as for the purposes of the CBA and the scalability and replicability activities for the Fellow Islands.

# Table of Contents

<b>List of Abbreviations and Acronyms</b>	<b>12</b>
<b>1. Introduction</b>	<b>14</b>
1.1 <i>Objectives and Scope</i>	14
1.2 <i>Relation to other IANOS activities</i>	14
1.3 <i>Deliverable outline</i>	16
1.4 <i>Extension to D3.5</i>	17
<b>2. IEPT suite</b>	<b>18</b>
2.1 <i>Description and Functionalities</i>	18
2.2 <i>Architecture</i>	20
Overview-Context view	21
Functional view	22
Information view	26
UML view	37
Open-API specifications view	38
2.3 <i>Tools integration process</i>	40
2.4 <i>User interface</i>	40
<b>3. CBA module</b>	<b>43</b>
3.1 <i>Overview</i>	43
3.2 <i>Functionalities and Methodological Approach</i>	44



3.3	<i>Implementation and technology stack</i>	50
<b>4.</b>	<b>ESSIM module</b>	<b>52</b>
4.1	<i>Overview</i>	52
	ESDL	53
	ESDL MapEditor	54
4.2	<i>Functionalities</i>	54
4.3	<i>Implementation and technology stack</i>	58
4.4	<i>User interface</i>	59
<b>5.</b>	<b>INTEMA.grid module</b>	<b>62</b>
5.1	<i>Overview</i>	62
5.2	<i>Functionalities</i>	63
5.3	<i>Implementation and technology stack</i>	65
	Data exchange model and synergy with the rest of IEPT tools	68
	Integration: exposed services/ API	69
5.4	<i>User interface</i>	70
<b>6.</b>	<b>VERIFY-D module</b>	<b>81</b>
6.1	<i>Overview</i>	81
6.2	<i>Functionalities</i>	81
6.3	<i>Implementation and technology stack</i>	82
6.4	<i>User interface</i>	82
<b>7.</b>	<b>Conclusions</b>	<b>83</b>



<b>8. Appendix A – Description of the INTEMA.grid dynamic component models</b>	<b>84</b>
8.1 <i>Passive grid components</i>	84
8.2 <i>Conventional power plants</i>	86
8.3 <i>Renewable power plants</i>	90
8.4 <i>Energy storage</i>	92
8.5 <i>IANOS technologies</i>	94
<b>9. References</b>	<b>112</b>

# List of Tables

Table 1: INTEMA.grid Microservice. ....	23
Table 2: ESSIM Microservice. ....	24
Table 3: CBA Microservice.....	24
Table 4: VERIFY-D Microservice.....	24
Table 5: CrowdEquity Microservice.....	25
Table 6: API Gateway.....	25
Table 7: Data interfaces of the components of IEPT suite.....	26
Table 8: Table of Class and Attributes for the JSON files contain the KPIs requested to be computed by the CBA Microservice. ....	27
Table 9: Table of Class and Attributes for the JSON files of the calculated KPIs data. ....	27
Table 10: The top-level class and attributes description for the JSON-request timeseries data, sent by VERIFY-D tool to the INTEMA.grid and/or ESSIM, respectively. ....	29
Table 11: The nested class and attributes description for the JSON file containing the timeseries data requested by the VERIFY-D tool from INTEMA.grid and/or ESSIM tools.....	30
Table 12: Table of class and attributes for the JSON file that contains the timeseries data provided back to VERIFY-D tool from INTEMA.GRID and/or ESSIM tools. ....	35
Table 13: The IEPT suite's API based on the Swagger documentation.....	38
Table 14: Recommendation for investments based on BCR values.....	47
Table 15: Mapping of KPIs to the specific components of the IEPT suite.....	49
Table 16: Functionalities attributes and development timeframe for IANOS.....	63
Table 17: Dynamic models development timeframe. ....	67
Table 18: INTEMA.grid frontend – components and required parameters.....	78
Table 19: INTEMA.grid frontend – IANOS technologies required parameters. ....	79
Table 20: Suwotec battery specifications as provided by the manufacturer .....	95
Table 21: Number of installed HHP units in 135 Ameland apartments per manufacturer model.....	96
Table 22: Technical specifications of Elga Ace HHP.....	97
Table 23: Elga Ace HHP nominal COP and heating power values for varying outdoor air temperature.....	97
Table 24: Elga Ace HP COP results comparison against available manufacturer data .....	99
Table 25: Elga Ace HP heating power results comparison against available manufacturer data .....	100
Table 26: Teraloop flywheel system parameters. ....	101



<i>Table 27: Parameters of the hybrid transformer according to IEC 60076.....</i>	<i>102</i>
<i>Table 28: Operating conditions of the geothermal plant [46].....</i>	<i>106</i>
<i>Table 29: Feedstock composition .....</i>	<i>109</i>
<i>Table 30: Simulation results for the Autogenerative High Pressure Digestion .....</i>	<i>110</i>



# List of Figures

Figure 1: PERT diagram showcasing the relation of T3.3 with the rest IANOS tasks. ....	16
Figure 2: Graphical outline of the deliverable. ....	17
Figure 3: Context view of IEPT suite. ....	22
Figure 4: Component Diagram of IEPT suite. ....	23
Figure 5: UML diagram of the IEPT Suite. ....	38
Figure 6: Registration page of the IEPT Suite dashboard. ....	41
Figure 7: Login page of the IEPT Suite dashboard. ....	41
Figure 8: "At a glance" page of the IEPT Suite dashboard. ....	42
Figure 9: The CBA methodology and its materialization in the IANOS project interventions. ....	45
Figure 10: Position of the CBA component in the IEPT suite and its interconnection with the rest components. ....	48
Figure 11: An overview of the functionalities of ESSIM. ....	52
Figure 12: ESSIM output: Hourly data showing imbalance. ....	55
Figure 13: ESSIM used in the planning-design phase of Energy Transition. ....	55
Figure 14: All implemented asset models in ESSIM. ....	56
Figure 15: Splitting of an energy system into constituent transport (transmission) networks. ....	57
Figure 16: ESSIM Technology stack. ....	58
Figure 17: Running ESSIM from MapEditor. ....	60
Figure 18: ESSIM simulation in progress. ....	60
Figure 19: Running ESSIM using Postman API platform. ....	61
Figure 20: INTEMA.grid technology stack. ....	66
Figure 21: INTEMA platform landing page. ....	70
Figure 22: INTEMA.grid case set-up process. ....	71
Figure 23: INTEMA.grid GUI - available systems and new system creation view. ....	72
Figure 24: INTEMA.grid GUI – location definition. ....	73
Figure 25: INTEMA.grid GUI – automatic weather data retrieve. ....	73
Figure 26: INTEMA.grid GUI – buses definition. ....	74
Figure 27: INTEMA.grid GUI – branches (lines and transformers) definition. ....	74
Figure 28: INTEMA.grid GUI - loads definition. ....	75
Figure 29: INTEMA.grid GUI - generators definition. ....	75



Figure 30: INTEMA.grid GUI – innovative IANOS technologies definition. ....	76
Figure 31: INTEMA.grid GUI – power flow additional parameters for the power plants. ....	77
Figure 32: INTEMA.grid GUI – power flow time-series definition view. ....	77
Figure 33: INTEMA.grid GUI – power flow simulation and results view. ....	78
Figure 34: CERTH developed steam power plant model. ....	87
Figure 35: Steam turbine governor IEEE-TGOV [30]. ....	87
Figure 36: Automatic voltage regulator (AVR) IEEE T1 [31]. ....	88
Figure 37: Simple PSS model [32]. ....	88
Figure 38: Gas turbine model type GAST [32]. ....	89
Figure 39: Diesel turbine governor DEGOV [30]. ....	90
Figure 40: Doubly-fed induction generator with cycloconverter of the experimental Growian wind turbine. .....	91
Figure 41: Direct drive permanent magnet synchronous generator wind turbine energy conversion principle scheme. ....	91
Figure 42: HYGOV hydro turbine-governor model [30] ....	92
Figure 43: The generalized Equivalent Circuit Model (ECM) [26]. ....	93
Figure 44: CERTH battery cell model with 1 RC branch and thermal dependence ....	94
Figure 45: Saline battery model ....	95
Figure 46: CERTH model of hybrid heat pump. ....	96
Figure 47: COP simulation results against data provided by manufacturer ....	98
Figure 48: Heating power simulation results against data provided by manufacturer ....	99
Figure 49: Developed flywheel energy storage model. ....	100
Figure 50: Modelica example of a hybrid power electronic transformer driving a dynamically changing asymmetric load. ....	102
Figure 51: Modelica example featuring a SOFC model operating in conjunction with components of the PowerSystems library. ....	104
Figure 52: Comparison of the Modelica efficiencies with the experimental data. Left: BG efficiencies as a function of the generated power in W (adapted from [42]). Right: As in left, but for the efficiencies calculated for the Modelica model. ....	104
Figure 53: Two-phase recuperated ORC flow diagram [46]. ....	106
Figure 54: Developed dynamic model for the geothermal power plant in Pico Alto. ....	107
Figure 55: CERTH PCM model ....	108
Figure 56: Flow diagram of the Autogenerative High Pressure Digestion process. ....	109



# List of Abbreviations and Acronyms

<i>Acronym</i>	<i>Meaning</i>
API	Application Programming Interface
BaU	Business as Usual
BCR	Benefit Cost Ratio
CBA	Cost Benefit Analysis
DCF	Discounted Cash Flow
DLT	Distributed Ledger Technology
DSO	Distribution System Operator
ENTSO-E	European Network of Transmission System Operators
ESDL	Energy System Description Language
ESSIM	Energy System Simulator
FI	Fellow Islands
HTTP	HyperText Transfer Protocol
IEPT	IANOS Energy Planning and Transition
INTEMA	INTEgrated Energy Management Simulator
IRR	Internal Rate of Return
KPI	Key Performance Indicators
LCA	Life Cycle Analysis
LCC	Life Cycle Cost
LH	Lighthouse Islands
ML	Machine Learning
MQTT	MQ Telemetry Transport
NPV	Net Present Value
OAS	OpenAPI Specification
PERT	Program Evaluation and Review Technique
RE	Renewable
RES	Renewable Energy Sources
REST (RESTful)	Representational State Transfer
ROI	Return of Investment

TSO	Transmission System Operator
UC	Use Case
UI	User Interface
UML	Unified Modelling Language
WP	Work Package





# 1. Introduction

## 1.1 Objectives and Scope

Work Package 3 (WP3) of IANOS is entitled as “Transition and Investment Decision Support Framework”. During the WP3 activities, a concrete IANOS Energy Planning and Transition decision-making (IEPT) suite was developed to assist the energy transition of the Lighthouse (LH) and Fellow Islands (FI). IEPT constitutes a holistic tool able to evaluate the overall benefits expected from clean energy/smart grid interventions, from various perspectives, based on the viewpoint of each stakeholder, i.e., Municipality, Distribution System Operator (DSO), community representatives, etc. The tool was pre-validated (reported in deliverable D3.8), by evaluating the LH and FI Use Cases (UCs) holistically for their sustainability, scalability and replicability potential.

Within WP3, Task 3.3 entitled as “Energy Planning & Transition Decision Support Toolset” aimed at providing information about the architecture and the communication of the different components under the umbrella of the IEPT suite, as well as presenting extensively the methodology for the Cost Benefit Analysis (CBA) tool, which is considered as the cornerstone of the IEPT suite. The different components that are integrated in the IEPT suite are also thoroughly described in this deliverable. The information included in both D3.2 “Development of LCA/LCC tool for Transition Support\_v2” and D3.4 “Equity crowdfunding tool for Community-driven Investments\_v2” is complementary to the work documented here and duplications have been omitted. Hence, we advise the reader to also read those deliverables, in order to have a complete view of the functionalities of the IEPT suite and its internal modules.

## 1.2 Relation to other IANOS activities

As can be seen in Figure 1, T3.3 needs to consider the interactions with several tasks in multiple WPs of the IANOS project. There is a strong interconnection with those tasks as concisely described below:



a) T3.3 builds on the findings of:

- T2.3: The activities and vision of the various stakeholders, as well as the factors of interest have been selected and respective KPIs have been introduced and were used as an input for the CBA approach.
- T2.5: The architecture of the IANOS solutions that have been introduced was used as a cornerstone for the definition of IEPT architecture.
- T3.1: The extensive description of the VERIFY-D tool, i.e., a Life Cycle Analysis (LCA)/ Life Cycle Cost (LCC) tool, is given in this task. As described later in this deliverable, this component is an important element of the IEPT suite.
- T3.2: The equity crowdfunding tool, as thoroughly described in this task, is an internal component of the IEPT suite and along with the outcome of the CBA provides a tool for stakeholders to invest in smart grid interventions.

b) T.3.3 constitutes an important input for:

- T3.4: The IEPT suite was tested and fine-tuned by the pre-validation/assessment of the selected UCs in LH and FI.
- T5.3: For the evaluation of the demonstration campaigns in the LH of Ameland, the functionalities of the IEPT suite are leveraged.
- T6.3: For the evaluation of the demonstration campaigns in the LH of Terceira, the functionalities of the IEPT suite are leveraged.
- T7.2: The environmental impact assessment that will be conducted in the context of this task will use the capabilities of the VERIFY-D component, which is an integral element of the IEPT suite.
- T7.3: The CBA that will be conducted in the context of this task will leverage the functionalities of the IEPT suite.
- T9.2: The scalability and replicability activities of this task will use the functionalities of the component of the IEPT platform.

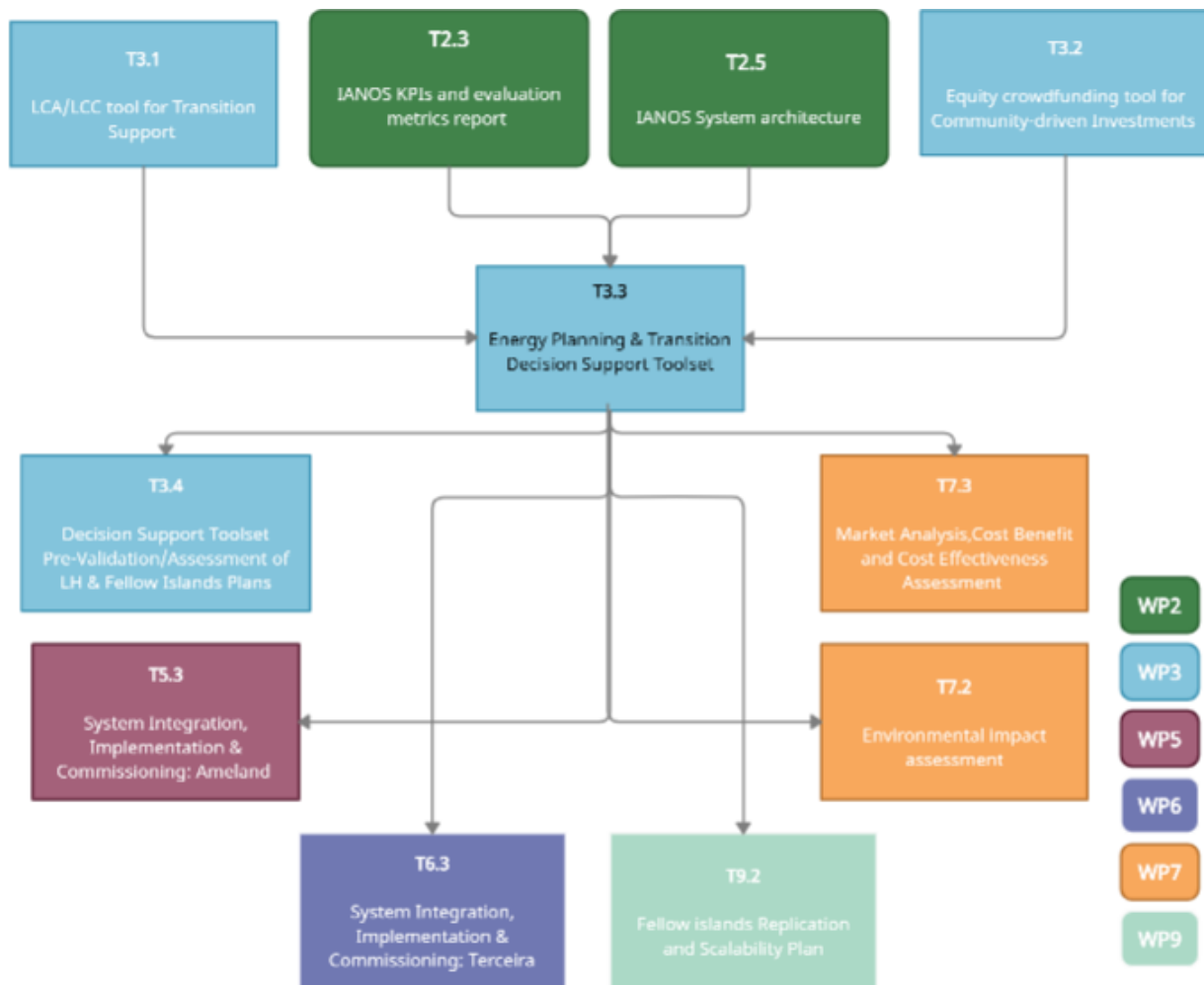


Figure 1: PERT diagram showcasing the relation of T3.3 with the rest IANOS tasks.

### 1.3 Deliverable outline

Figure 2 provides a graphical outline of D3.6. This deliverable aims at describing the IEPT suite, its architecture, functionalities and internal components. Specifically, chapter 2 provides an extensive description of the first two points, i.e., the architecture and the functionalities of the IEPT suite, along with its user interface. Chapters 3 to 6 provide an overview of the internal components of the IEPT suite, along with the specific functionalities of each tool, the description of the used technology stack and the user interfaces. Finally, chapter 7 concludes the presentation of the IEPT suite briefly providing its main characteristics and functionalities, as well as the connection with the rest of the IANOS tasks.

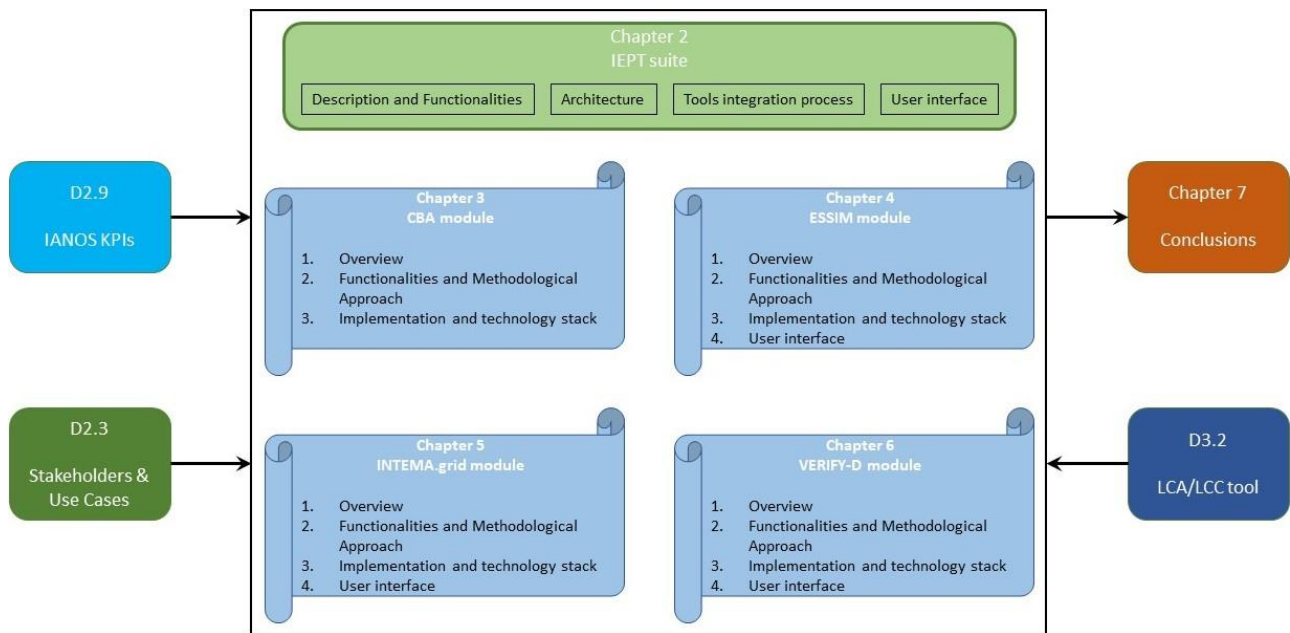


Figure 2: Graphical outline of the deliverable.

## 1.4 Extension to D3.5

The current deliverable (D3.6) is an updated version of D3.5. New sections were added and some of the content was modified to accurately reflect the status of the project up to M30. Specifically, the changes from the previous version include:

- New subsections “UML view” and “Open-API specifications view” under section 2.2, presenting additional viewpoints and a better visualization and understanding of the IEPT suite’s architecture.
- New section 2.3, describing the integration process of IEPT suite’s different components.
- New sections 2.4, 4.4, 5.4 and 6.4, where the UIs of the different tools are presented.
- The rest of the content was updated accordingly, based on all available information and progress of the project up to M30.

## 2. IEPT suite

### 2.1 Description and Functionalities

During the WP3 activities, a concrete IANOS Energy Planning and Transition decision-making (IEPT) suite was developed to assist the energy transition of the Lighthouse (LH) and Fellow Islands (FI). IEPT constitutes a holistic tool able to evaluate the overall benefits expected from clean energy/smart grid interventions, from various perspectives, based on the viewpoint of each stakeholder (Building owner, RES owner, Municipality, DSO, community representatives, etc.). The tool was pre-validated (reported in deliverable D3.8), by evaluating the LH and FI Use Cases (UCs) holistically for their sustainability, scalability and replicability potential.

The different components that have been integrated under a unified concept are the following:

- a) **VERIFY District Platform (VERIFY-D)**: A dedicated web platform for Life Cycle Analysis (LCA)/Life Cost Cycle (LCC) studies. Environmental and economic analyses are conducted through the computation of emissions extracted (life cycle environmental footprint), which depend on the technology applied in the grid, as well as the type of fuel mixture used to generate energy in the LH islands. Also, this tool computes the life cycle costing based on the energy components that appear in the energy grid. Finally, the comparison between baseline and target scenarios in terms of the environmental gains and economic profits is analysed.
- b) **An equity crowdfunding tool (CrowdEq)<sup>1</sup>**: This tool will create a crowd-equity (or crowdfunding) platform, where actors (i.e., project investors, islanders and other key stakeholders) will be able to register their foreseen projects and set a funding goal in return for equity, creating a fundraising campaign. The platform will ultimately enable fractional

---

<sup>1</sup> Extensive description of the CrowdEq tool can be found in D3.4.

ownership of RE assets, supporting projects' fundraising and transactions either via normal FIAT currency or tokenized energy over Distributed Ledger Technology (DLT).

- c) **Energy System Simulator (ESSIM):** This tool simulates network balancing and the effects, thereof, in an interconnected hybrid energy system over a period of time. It takes as input the energy system defined in ESDL and calculates the optimal schedule of flexible producers and the effect of this schedule in terms of emissions, costs, load on the network, etc.
- d) **INTEgrated Energy Management Simulator (INTEMA.grid):** It is an energy system modelling and simulation platform based on open-source non-proprietary tools. It is composed of the following main components: (a) dynamic power system simulations – based on the acausal, object-oriented, equation based Modelica language to conveniently model complex physical systems, (b) power optimization – ability to construct optimization problems using the Python optimization tool PyPSA and (c) power forecasting – utilizing open-source Machine Learning (ML) frameworks in Python, i.e., scikit-learn and Keras.
- e) **A Cost-Benefit analysis tool (CBA):** The CBA tool was based on the JRC's and ENTSO-E's CBA methodologies. For this reason and depending on the activities and vision of the various stakeholders, the factors of interest will be selected and respective KPIs will be calculated to assist them in setting strategic priorities, aligning horizontally in all cases with the priorities set by the EU Green Deal. The CBA tool offers an analytical approach for the stakeholders/investors and provides a quantifiable insight regarding whether a smart grid intervention exceeds the existing baseline scenario in terms of costs and benefits. This tool is considered as the cornerstone of the IEPT suite and will assist the activities in several tasks across the WPs of IANOS, such as the CBA of

WP7 and the scalability and replicability studies activities that will be conducted in WP9.

## 2.2 Architecture

The system design is a conceptualization depicting its fundamental characteristics, while demonstrating the means to fulfil its individual requirements. The methodology adopted to report the design is based on the Viewpoints and Perspectives framework [1] and ISO/IEC/IEEE 42010:2011 “Systems and software engineering-Architecture description” [2]. According to that concept, an architecture description may be organised in different views using viewpoints, providing a mechanism for the separation of concerns amongst the stakeholders, while at the same time providing the description of the whole system that is fundamental to the notion of architecture. A viewpoint describes how certain stakeholders - with a particular concern - interpret the system. An architecture viewpoint frames one or more concerns, providing the conventions for constructing, presenting and analysing views. A view is a representation of an entire system from the perspective of a related set of concerns. It expresses the architecture of the system-of-interest in accordance with an architecture viewpoint.

For the proper reporting of the final version of the IEPT suite, the following viewpoints have been selected:

- **Overview-Context:** Describes the main interactions of the system with its environment, providing a high-level concise overview.
- **Functional:** Describes the system's functional elements, their responsibilities, interfaces and primary interactions. A Functional view is the cornerstone of most architectural designs and is often the first part of the description that stakeholders try to read.
- **Information:** Describes the way that the architecture stores, manipulates, manages and distributes information. The ultimate purpose of virtually any computer system is to manipulate information in some form and this viewpoint develops a complete, but high-level, view of static data structure and information flow.

- **Unified Modeling Language (UML):** Through a UML diagram, a visual representation of the system's architecture and the interactions between its internal components is achieved.
- **Open-API specifications:** Describes the way that the IEPT suite's different components exchange information with each other through their respective API endpoints that constitute the suite's main communication gateway.

OpenAPI Specification (OAS) is a standard, language-agnostic interface to RESTful APIs that allows both humans and computers to discover and understand the capabilities of the service without access to source code and is utilized to document REST interfaces of the suite. More specifically the Swagger tool was used to adequately report the interfaces both in file format, as well as in a more user-friendly graphical environment [4].

In order to facilitate the reader and provide a well-established approach across software engineering industry about the modelling of the system's views, the Unified Modeling Language (UML) is used [3]. Specifically, component diagrams are used to model the aspects concerning the functional and UML viewpoints.

## ***Overview-Context view***

This subsection aims at providing a high-level view of the IEPT suite, exposing its scope, responsibilities and interfaces with its environment, i.e., people, systems, external components. The IEPT suite acts as an integration solution of existing and under-development components and provides interfaces to stakeholders to conduct planning and funding processes for smart interventions. Figure 3 illustrates the context view of the IEPT suite. The IEPT suite provides the following:

- Integration of the components developed by different providers.
- Access to data of the different Lighthouse (LH) islands of the IANOS project.



- Access analysis data from the different components and data flow coordination amongst them.
- Provision of analytics to the engaged stakeholders.

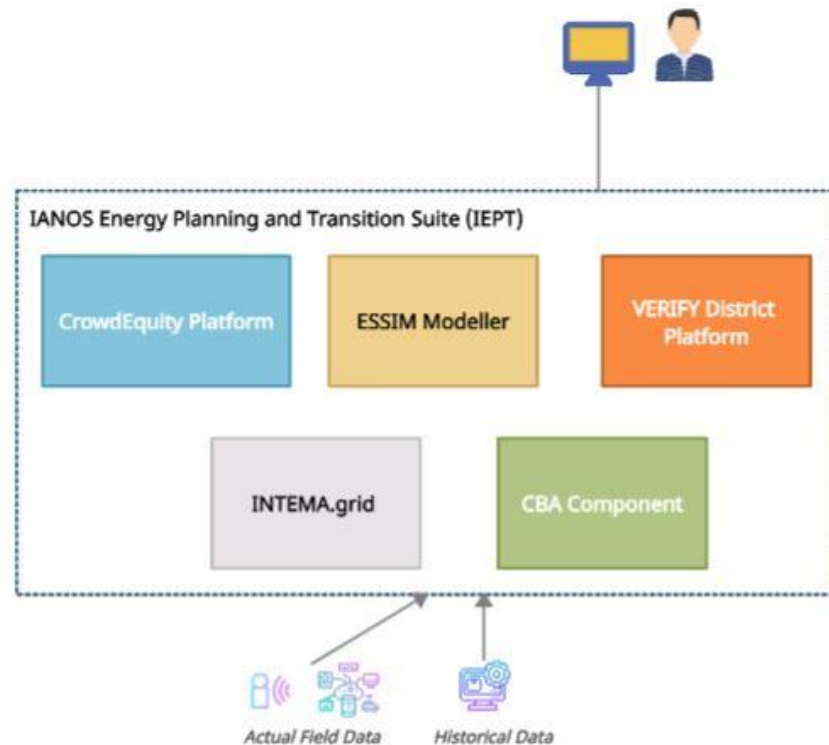


Figure 3: Context view of IEPT suite.

## Functional view

The functional viewpoint describes the system's functional elements, their responsibilities, interfaces and primary interactions.

The Microservice architectural paradigm is applied for the design of the solution [5]. Following this paradigm, a collection of lightweight, loosely-coupled services is utilized - each one offering a single business capability. Those services are the following ones:

- CBA Component
- INTEMA.grid
- ESSIM System Modeler
- VERIFY District Platform
- CrowdEquity platform

This architecture offers great independence in the scalability of the solution, the development and realisation of the various architectural components, as well as the capability of decentralized governance (implementation decisions) offering the relevant autonomy of each component. The microservices are containerized in Dockers, simplifying the delivery and management of those microservices [6]. The process of containerization provides individual microservices with their own isolated workload environments, making them independently deployable and scalable. The main elements of the design are illustrated in the diagram of Figure 4 and are elaborated on the list of Tables below.

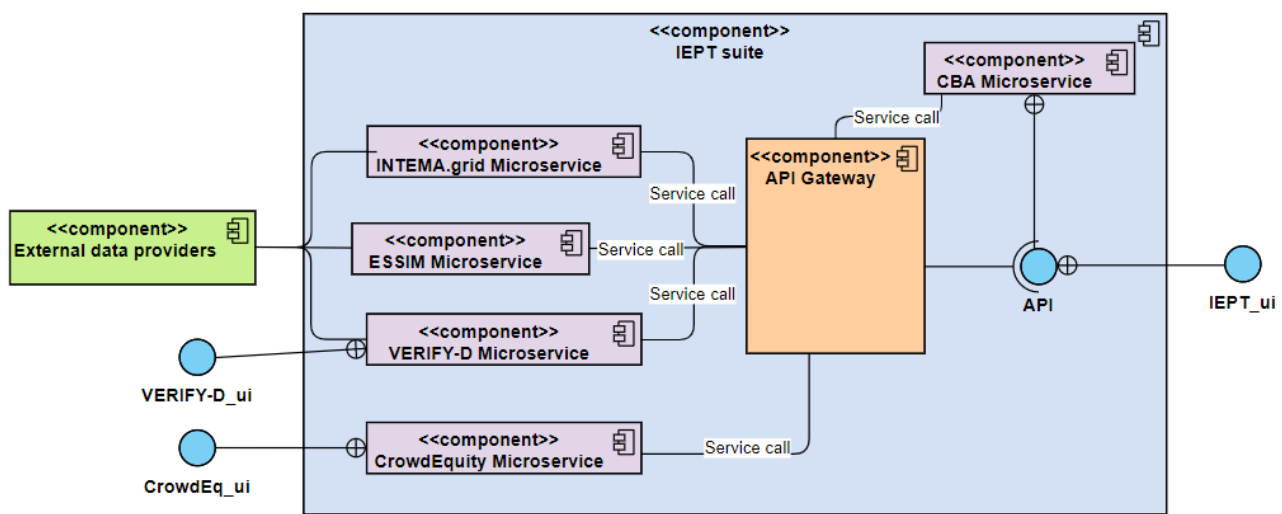


Figure 4: Component Diagram of IEPT suite.

Table 1: INTEMA.grid Microservice.

Element Name	INTEMA.grid Microservice
Responsibilities	An application offering an autonomous business capability exposed through a REST interface, via the API Gateway. This dockerized application provides system modelling capabilities for the Terceira LH island. Specific KPIs are computed and provided as an input into the CBA Microservice. Extensive description of this component is given in Chapter 5.

<b>Interfaces</b>	<b>Service_call:</b> Interface to API Gateway providing calculated KPI values and exposing services to the IEPT_ui.
-------------------	---

Table 2: ESSIM Microservice.

<b>Element Name</b>	<b>ESSIM Microservice</b>
<b>Responsibilities</b>	An application offering an autonomous business capability exposed through a REST interface, via the API Gateway. This dockerized application provides system modelling capabilities for the Ameland LH island. Specific KPIs are computed and provided as an input into the CBA Microservice. Extensive description of this component is given in Chapter 4.
<b>Interfaces</b>	<b>Service_call:</b> Interface to API Gateway providing calculated KPI values and exposing services to the IEPT_ui.

Table 3: CBA Microservice.

<b>Element Name</b>	<b>CBA Microservice</b>
<b>Responsibilities</b>	An application offering an autonomous business capability exposed through a REST interface, via the API Gateway. This component conducts a CBA analysis using the KPIs values, calculated by the rest microservices. Extensive description of this component is given in Chapter 3.
<b>Interfaces</b>	<b>API:</b> Service requests for KPIs calculation. <b>Service_call:</b> Interface to API Gateway providing calculated KPI values and exposing services to the IEPT_ui.

Table 4: VERIFY-D Microservice.

<b>Element Name</b>	<b>VERIFY-D Microservice</b>
<b>Responsibilities</b>	An application offering an autonomous business capability exposed through a REST interface, via the API Gateway. This dockerized application provides LCA/LCC capabilities for

	both LH islands. Extensive description of this component is given in Chapter 6.
<b>Interfaces</b>	<p><b>VERIFY-D_ui:</b> User interface for managing and monitoring the VERIFY component directly.</p> <p><b>Service_call:</b> Interface to API Gateway providing calculated KPI values.</p>

Table 5: CrowdEquity Microservice.

<b>Element Name</b>	<b>CrowdEquity Microservice</b>
<b>Responsibilities</b>	An application offering an autonomous business capability exposed through a REST interface, via the API Gateway. Due to the non-direct communication of this component with the REST microservices, the extensive description of this component is not provided in this deliverable but in D3.4.
<b>Interfaces</b>	<p><b>CrowdEq_ui:</b> User interface for managing and monitoring the CrowdEquity component directly.</p> <p><b>Service_call:</b> Interface to API Gateway for exposing services to the IEPT_ui.</p>

Table 6: API Gateway

<b>Element Name</b>	<b>API Gateway</b>
<b>Responsibilities</b>	Acts as a single-entry point for all requests as well as an edge service for exposing the different microservices to the CBA component. It acts as an intermediate in all the incoming requests and routes them towards the respective components in the IEPT suite. A detailed description of the APIs is presented in Table 7.
<b>Interfaces</b>	<p><b>API:</b> Service requests from CBA component and the IEPT_ui.</p> <p><b>Service_call:</b> Interface to microservices.</p>

## Information view

A detailed analysis of the information flows through the IEPT suite was conducted. Table 7 contains the information regarding the data interfaces in the IEPT suite.

Table 7: Data interfaces of the components of IEPT suite.

Component	Data Item	Related Data Model	Data format	Communication Information
<b>INTEMA.grid</b>	KPIs relevant to the Terceira LH island	HTTP (REST)	JSON	HTTP REST services for data exchange
<b>ESSIM</b>	KPIs relevant to the Ameland LH island	HTTP (REST)	JSON	HTTP REST services for data exchange
<b>CBA</b>	Requests for data exchange with the rest components	HTTP (REST)	JSON	HTTP REST services for data exchange
<b>VERIFY-D</b>	KPIs relevant to both of the LH islands	HTTP (REST)	JSON	HTTP REST services for data exchange
<b>CrowdEquity</b>	Outcome of CrowdEquity process	HTTP (REST)	JSON	HTTP REST services for data exchange

The specific format of the exchanged data has been defined as follows:

### a) KPIs Data Exchange

Table 8: Table of Class and Attributes for the JSON files contain the KPIs requested to be computed by the CBA Microservice.

Class	Attribute <sup>2</sup>
KPI_ID	<p><b>ID (String):</b> The specific ID for a KPI.</p> <p><b>Source (String):</b> The component that originally calculated the specific KPI.</p> <p><b>Location (String):</b> The demonstrator for which the specific KPI was calculated.</p> <p><b>UseCase (String):</b> The specific scenario for a particular Use Case for which the KPI was calculated.</p>

Here, we provide an indicative example of the output JSON files that the CBA microservice requests from INTEMA, VERIFY and ESSIM.

```
{
  "KPI_ID": [
    {
      "ID": "T1",
      "Source": "ESSIM",
      "Location": "Ameland",
      "UseCase": "1-Baseline"
    }
  ]
}
```

Table 9: Table of Class and Attributes for the JSON files of the calculated KPIs data.

Class	Attribute <sup>3</sup>
KPI_ID	<p><b>ID (String):</b> The specific ID for a KPI.</p> <p><b>Source (String):</b> The component that originally calculated the specific KPI.</p> <p><b>Location (String):</b> The demonstrator for which the specific KPI was calculated.</p>

<sup>2</sup> The ID of the specific KPIs along with the Use Cases nomenclature and the components that compute each KPI are thoroughly presented in chapter 3 of this deliverable.

<sup>3</sup> The ID of the specific KPIs along with the Use Cases nomenclature and the components that compute each KPI are thoroughly presented in chapter 3 of this deliverable.

	<b>UseCase (Integer):</b> The specific Use Case for which the KPI was calculated. <b>Timestamp:</b> Timestamp of the KPI calculation process.
KPI_value	<b>Value (Float):</b> The value that corresponds to the specific KPI. <b>Units (String):</b> Units. <b>eventStartTime:</b> The starting timestamp of the calculation period for the specific KPI (optional). <b>eventEndTime:</b> The ending timestamp of the calculation period for the specific KPI (optional).
KPI_Valid	<b>CalculateActions (Binary):</b> 1 whether the KPI was calculated and 0 whether the calculation was not conducted. <b>DataUsed (String):</b> Synthetic or real data stemming from demonstrator campaigns.

Here, we provide an indicative example of the output JSON files that the INTEMA, VERIFY and ESSIM Microservices sent to the CBA microservice after its request.

```
{
  "KPI_ID":[
    {
      "ID": "T1",
      "Source": "ESSIM",
      "Location": "Ameland",
      "UseCase": "1",
      "Timestamp": "23-02-2022 16:02:00"
    }
  ],
  "KPI_value":[
    {
      "Value": "20.2",
      "Units": "kWh",
      "eventStartTime": "22-02-2022 16:00:00",
      "eventEndTime": "22-02-2022 18:00:00"
    }
  ],
  "KPI_Valid":[
    {
      "CalculateActions": "1",
      "DataUsed": "Synthetic"
    }
  ]
}
```

b) Timeseries Data Exchange

Table 10: The top-level class and attributes description for the JSON-request timeseries data, sent by VERIFY-D tool to the INTEMA.grid and/or ESSIM, respectively.

Class	Attribute
GeneralInfo	<p><b>Name (String):</b> The project's name assigned by the user.</p> <p><b>Country (String):</b> The demonstrator's country.</p> <p><b>Coordinates (List&lt;Float&gt;):</b> The demonstrator's exact latitude and longitude.</p> <p><b>Altitude (Float):</b> The demonstrator's altitude measured from the sea level.</p> <p><b>Lifespan (Integer):</b> The considered lifespan of the project.</p>
CurrentScenario	<p><b>Buildings (Hash):</b> Details about the buildings of the demonstrator's current state.</p> <p><b>Plants (Hash):</b> Details about the energy production plants of the demonstrator's current state.</p> <p><b>EnergyStorage (Hash):</b> Details about the energy storage systems of the demonstrator's current state.</p> <p><b>PublicInfrastructure (Hash):</b> Details about the public infrastructures of the demonstrator's current state.</p> <p><b>Transport (Hash):</b> Details about the vehicles of the demonstrator's current state.</p>
PlannedScenario	<p><i>Similar attributes to CurrentScenario. Details about the demonstrator's planned state.</i></p>
TimeSeriesConfig	<p><b>Timezone (String):</b> Time zone of the country in which the scenario is applied.</p> <p><b>Timestep (Integer):</b> Time step of the time series in minutes.</p> <p><b>StartDatetimeCurrent (Timestamp):</b> Starting datetime for time series of the current scenario.</p> <p><b>EndDatetimeCurrent (Timestamp):</b> Ending datetime for time series of the current scenario.</p>



	<p><b>StartDatetimePlanned (Timestamp):</b> Starting datetime for time series of the planned scenario.</p> <p><b>EndDatetimePlanned (Timestamp):</b> Ending datetime for time series of the planned scenario.</p> <p><b>RequestFrom (String):</b> The name of the service which will generate the requested time series.</p>
FuelPrices	<p><b>BuildingExport (Float):</b> Electricity export price for buildings (€/kWh).</p> <p><b>BuildingImport (Float):</b> Electricity import price for buildings (€/kWh).</p> <p><b>PlantExport (Float):</b> Electricity export price for power plants (€/kWh).</p> <p><b>NaturalGas (Float):</b> Natural gas price (€/kWh).</p> <p><b>Diesel (Float):</b> Diesel price (€/liter).</p> <p><b>Biomass (Float):</b> Biomass price (€/kg).</p>

Details of the nested attributes, included in the JSON-request file are presented in Table 11.

*Table 11: The nested class and attributes description for the JSON file containing the timeseries data requested by the VERIFY-D tool from INTEMA.grid and/or ESSIM tools.*

Class	Attribute	
Building	<p><b>Type (String):</b> The purpose of the building (e.g., single household).</p> <p><b>Number (Integer):</b> The number of the buildings with the same typology and infrastructure existing on the island.</p>	
	Envelope	<p><b>TargetTemperature (Hash):</b> The indoor temperature preferred by the platform user and must be achieved by thermal components.</p> <p>Winter and summer preference.</p>

		<b>FloorSurface (Float):</b> The area of the building's floor measured in square meters. <b>ExternalWallMaterial (String):</b> The material used for the building's external wall construction. <b>Glazing (Hash):</b> The specifications of the building's glazing infrastructure. <b>Insulation (Hash):</b> The specifications of the building's insulation infrastructure.	
	Electrical	<b>AnnualConsumption (Float):</b> The building's annual electricity consumption for lighting and appliances.	
		Generation	<b>PV (List&lt;Hash&gt;):</b> The building's installed photovoltaics specifications. <b>Wind (List&lt;Hash&gt;):</b> The building's installed wind turbines specifications.
		<b>Storage (Hash):</b> The building's installed battery's specifications.	
	Thermal	<b>HeatSource (Hash):</b> The building's installed cool source (e.g., air-conditioner) specifications. <b>CoolSource (Hash):</b> The building's installed cool source (e.g., boiler) specifications.	
	<b>TimeSeriesNames (List&lt;String&gt;):</b> The names of the energy time series to be requested by external tools for the building.		
Plants	Renewable	<b>Wind (List&lt;Hash&gt;):</b> The specifications of wind installations. <b>Solar (List&lt;Hash&gt;):</b> The specifications of solar installations. <b>Tidal (List&lt;Hash&gt;):</b> The specifications of tidal kites. <b>Geothermal (List&lt;Hash&gt;):</b> The specifications of geothermal plants.	

		<p><b>Hydro (List&lt;Hash&gt;):</b> The specifications of hydro plants.</p> <p><b>Incineration (List&lt;Hash&gt;):</b> The specifications of fuel cells.</p>
	Convention al	<p><b>PowerPlants (List&lt;Hash&gt;):</b> The specifications of power plants.</p> <p><b>FuelCellPlants (List&lt;Hash&gt;):</b> The specifications of fuel cells.</p>
		<b>TimeSeriesNames (List&lt;String&gt;):</b> The names of the energy production time series to be requested by external tools each energy source.
EnergyStorage		<p><b>Batteries (List&lt;Hash&gt;):</b> The specifications of batteries.</p> <p><b>Flywheels (List&lt;Hash&gt;):</b> The specifications of flywheels.</p>
PublicInfrastructure		<p><b>ChargingStations (List&lt;Hash&gt;):</b> The specifications of electric vehicle charging stations.</p> <p><b>Transformers (List&lt;Hash&gt;):</b> The specifications of transformers.</p> <p><b>Lighting (List&lt;Hash&gt;):</b> The specifications of lighting infrastructure.</p> <p><b>TimeSeriesNames (List&lt;String&gt;):</b> The names of the time series for the energy consumption of the components.</p>
Transport		<b>EVs (List&lt;Hash&gt;):</b> The specifications of electric vehicles.

An indicative example of the output JSON-response files that the VERIFY-D requests from INTEMA.grid and/or ESSIM is presented. The extended synthetic/estimated timeseries data are received through the JSON response file.

```
{
  "Project": {
    "Name": "Ameland",
    "GeneralInfo": {
      "Country": "Netherlands",
      "coordinates": [
        53.448,
        5.73
      ],
      "altitude": 4,

```

```

    "lifespan": 25
  },
  "CurrentScenario": {
    "Buildings": {
      "Building_1": {
        "id": 1,
        "type": "SingleHousehold",
        "envelope": {
          "target_temperature": {
            "winter": 22,
            "summer": 27
          },
        },
        "FloorSurface": 120,
        "ExternalWallMaterial": "Brick",
        "Glazing": {
          "Opening": 50,
          "Layers": 2,
          "Thickness": 10,
          "GlassType": "NormalGlass",
          "Frame": "Aluminum"
        },
        "Insulation": {
          "Surface": 100,
          "Material": "StoneWool",
          "Thickness": 75
        }
      },
    },
    "electrical": {
      "AnnualConsumption": 2500,
      "Generation": {
        "PVs": [
          {
            "Type": "RoofFlat",
            "InstalledPower": 10
          }
        ],
        "Wind": [
          {
            "Type": "Geared",
            "InstalledPower": 25
          }
        ]
      },
    },
    "Storage": {
      "Type": "Lilon",
      "Capacity": 80
    },
  },
  "thermal": {

```

```

    "HeatSource": {
      "Type": "BoilerNGas",
      "InstalledPower": 25
    },
    "CoolSource": {
      "Type": "AirconditionBasic",
      "InstalledPower": 3.5
    }
  },
  "TimeSeriesNames": [
    "electricity_consumption",
    "boiler_ngas",
    "aircondition_basic",
    "temperature_indoor"
  ]
},
"Plants": {
  "Renewable": {
    "Wind": [
      {
        "Pinst": "300.0",
        "type": "geared",
        "number": "2",
        "column_name": "wind_farm0"
      }
    ],
    "Solar": [
      {
        "Pinst": "50.0",
        "type": "poly",
        "number": "1",
        "column_name": "solar_farm0"
      }
    ],
    "Tidal": [],
    "Geothermal": [],
    "Hydro": [],
    "Incineration": []
  },
  "Conventional": {
    "PowerPlants": [],
    "FuelCellPlants": []
  },
  "TimeSeriesNames": [
    "electricity_production"
  ]
},
"EnergyStorage": {

```

```

    "Batteries": [],
    "Flywheels": []
  },
  "PublicInfrastructure": {
    "ChargingStations": [],
    "Transformers": [],
    "Lighting": [],
    "TimeSeriesNames": [
      "electricity_consumption"
    ]
  },
  "Transport": {
    "EVs": []
  }
},
"PlannedScenario": {},
"FuelPrices": {
  "BuildingExport": 0.05,
  "BuildingImport": 0.12,
  "PlantExport": 0.1,
  "NaturalGas": 0.04,
  "Diesel": 0.2,
  "Biomass": 0.07
},
"TimeSeriesConfig": {
  "Timezone": "UTC",
  "Timestep": 60,
  "StartDatetimeCurrent": "2021-02-28T00:00:00.000+00:00",
  "EndDatetimeCurrent": "2022-02-28T00:00:00.000+00:00",
  "StartDatetimePlanned": "2022-02-28T00:00:00.000+00:00",
  "EndDatetimePlanned": "2023-02-28T00:00:00.000+00:00",
  "RequestFrom": "INTEMA.grid"
}
}
}

```

Table 12: Table of class and attributes for the JSON file that contains the timeseries data provided back to VERIFY-D tool from INTEMA.GRID and/or ESSIM tools.

Class	Attribute
Timeseries	<b>SentBy:</b> (String): The platform that generated the requested time series.
	<b>CompletedAt:</b> (Timestamp): The timestamp that the time series generation was completed.

	<b>CurrentScenario (Hash):</b> The requested time series for the current state of the demonstrator. <b>PlannedScenario (Hash):</b> The requested time series for the planned scenario.	
Buildings	Building_ID	<b>Unit (String):</b> The unit in which the time series are returned. <b>Values: (List&lt;Float&gt;):</b> The requested time series for the building with the specific ID.
Plants	Component_ID	<b>Unit (String):</b> The unit in which the time series are returned. <b>Values: (List&lt;Float&gt;):</b> The requested time series for the plant with the specific ID.
Public Infrastructure	Plant_ID	<b>Unit (String):</b> The unit in which the time series are returned. <b>Values: (List&lt;Float&gt;):</b> The requested time series for the component with the specific ID.

Here, we provide an indicative example of the output JSON file that the INTEMA.grid and/or ESSIM tool respond to VERIFY-D containing all the time series data.

```
{
  "TimeSeries": {
    "CurrentScenario": {
      "Buildings": {
        "1": {
          "electricity_consumption": {
            "unit": "kWh",
            "values": []
          },
          "boiler_ngas": {
            "unit": "kWh",
            "values": []
          },
          "aircondition_basic": {
            "unit": "kWh",
            "values": []
          },
          "temperature_indoor": {
```

```

        "unit": "kWh",
        "values": []
      }
    },
    "Plants": {
      "1": {
        "electricity_production": {
          "unit": "kWh",
          "values": []
        }
      }
    },
    "PublicInfrastructure": {
      "2": {
        "electricity_consumption": {
          "unit": "kWh",
          "values": []
        }
      }
    }
  },
  "PlannedScenario": {},
  "SentBy": "INTEMA.grid",
  "CompletedAt": "2021-02-28T14:25:35.584+00:00"
}

```

## UML view

Figure 5 shows a UML diagram of the IEPT suite focused on the interactions between the different components. Initially, the pre-defined use cases are communicated to the modules VERIFY-D, INTEMA.grid and ESSIM to run the simulations based on the scenarios of the interventions to be studied per use case. After that, the results are stored to the central database through the API gateway. As a next step, the CBA module gets the calculated results to conduct the analysis and store the KPIs into the central database again through the API gateway. At this moment, all the results and KPIs for a specific use case are stored in the local database and the IEPT UI can access and visualize them.



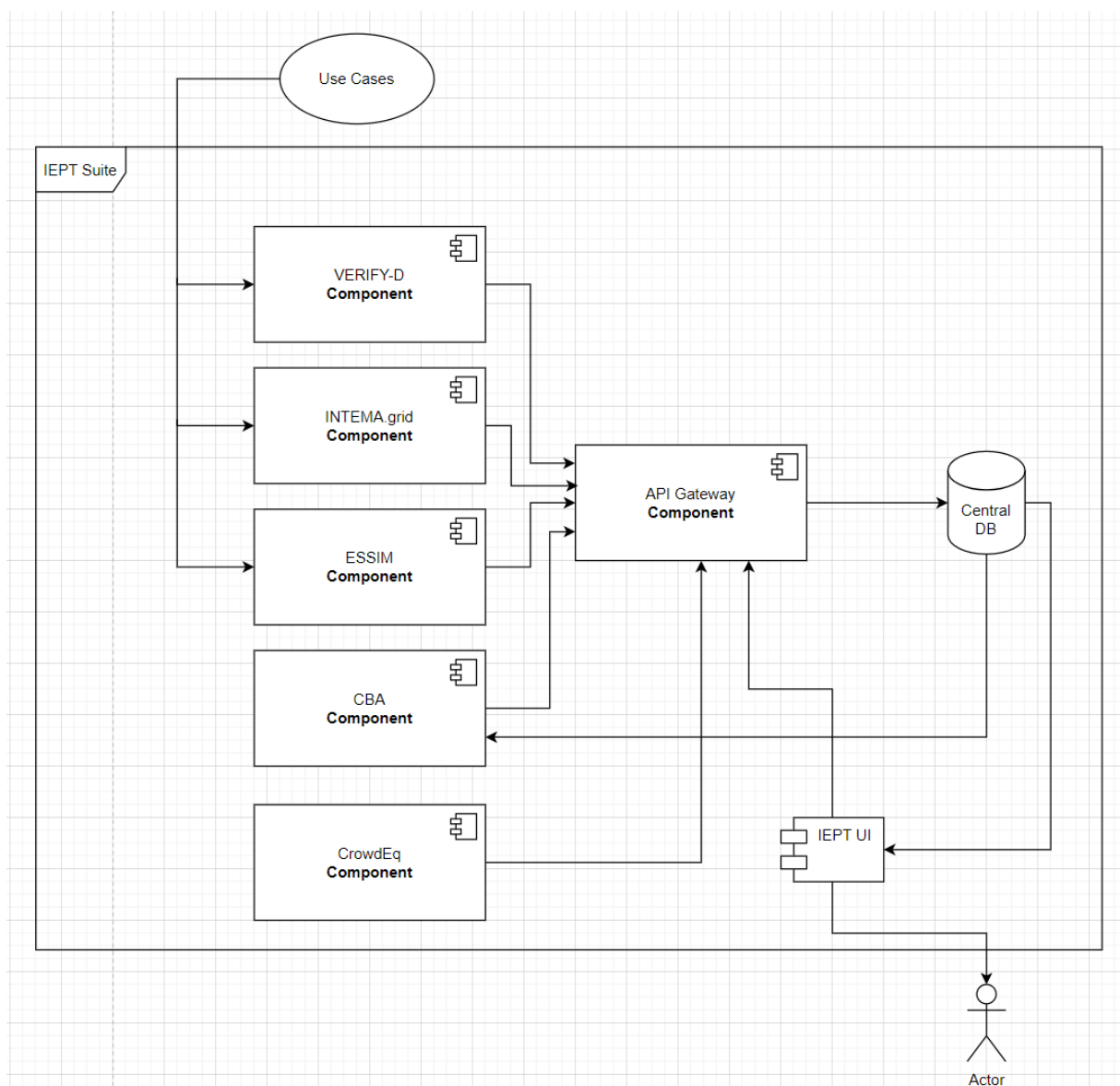


Figure 5: UML diagram of the IEPT Suite.

## Open-API specifications view

Based on the Swagger tool the collection of the API endpoints that constitute the API gateway are summarized in Table 13.

Table 13: The IEPT suite's API based on the Swagger documentation.

cba-controller CBA Controller	
POST	/dataService/CBA_Auth Endpoint for authorization to access the IEPT_Suite

POST	/dataService/CBA_KPI_Store Endpoint to store the CBA specific KPIs to the central database
POST	/dataService/CBA_KPI Endpoint to request the CBA specific KPIs from the microservices
GET	/dataService/CBA_KPI Endpoint to request the CBA specific KPIs from the central database
<b>VERIFY-D-controller VERIFY-D Controller</b>	
POST	/dataService/VERIFY-D_Auth Endpoint for authorization to access the IEPT_Suite
POST	/dataService/VERIFY-D_KPI Endpoint to store the specific KPIs to the central database
POST	/dataService/VERIFY-D_TimeSeries Endpoint to request the timeseries data from ESSIM and/or INTEMA.grid
<b>INTEMAgrid-controller INTEMAgrid Controller</b>	
POST	/dataService/INTEMAgrid_Auth Endpoint for authorization to access the IEPT_Suite
POST	/dataService/INTEMAgrid_KPI Endpoint to store the specific KPIs to the central database
POST	/dataService/INTEMA.grid_TimeSeries Endpoint to request the timeseries data from VERIFY-D and/or ESSIM
<b>ESSIM-controller ESSIM Controller</b>	
POST	/dataService/ESSIM_Auth Endpoint for authorization to access the IEPT_Suite
POST	/dataService/ESSIM_KPI Endpoint to store the specific KPIs to the central database
POST	/dataService/ESSIM_TimeSeries Endpoint to request the timeseries data from VERIFY-D and/or INTEMA.grid
<b>CrowdEq-controller CrowdEq Controller</b>	
POST	/dataService/CrowdEq_Auth Endpoint for authorization to access the IEPT_Suite
POST	/dataService/CrowdEq Endpoint to store the outcome to the central database
<b>IEPT_ui-controller IEPT_ui Controller</b>	
GET	/dataService/Visual Endpoint to request specific KPIs from the central database for visualization to the IEPT Suite's dashboard

## 2.3 Tools integration process

The integration of the different modules has been done with REST APIs through the API gateway. The procedure includes two steps:

- Step 1: Access token retrieval
- Step 2: KPIs storage/retrieval

Specific credentials have been generated for each module representative in order to establish a secure connection to the API gateway. Each time a module needs to use the API gateway the access token retrieval is being done based on these credentials (username and password). After the successful authentication, the module can do one of the following:

- Store calculated results/KPIs to the central database.
- Retrieve calculated results/KPIs from the other modules from the central database.
- Request timeseries from the other modules that can be used in the simulations.

## 2.4 User interface

In the user interface, users are required to enter their personal information as well as their role and the demonstration site they are interested in, as seen in Figure 6. At the time of submission of this deliverable, the available roles are "DSO", "Municipality" and "Community Representative" and the available demonstration sites are "Ameland", "Terceira", "Lampedusa", "Bora Bora" and "Nisyros".

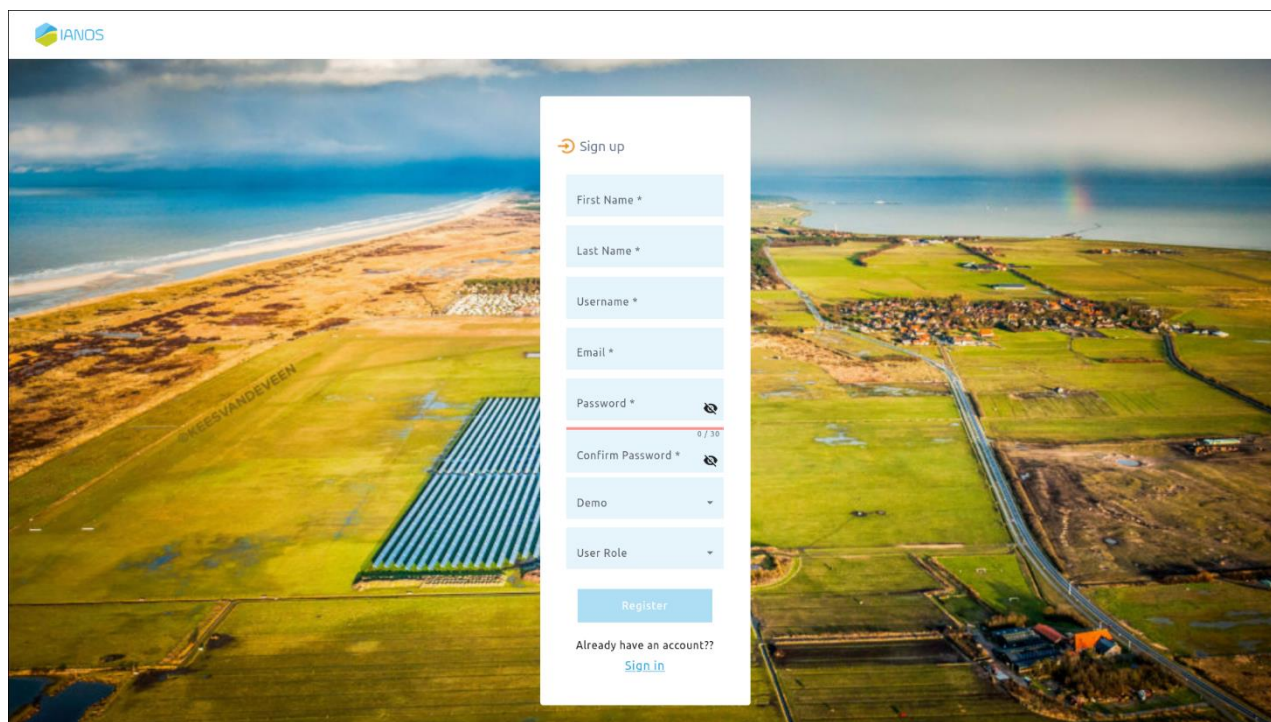


Figure 6: Registration page of the IEPT Suite dashboard.

Upon successful registration, users are able to log in, as seen in Figure 7 and access the IEPT Suite dashboard.

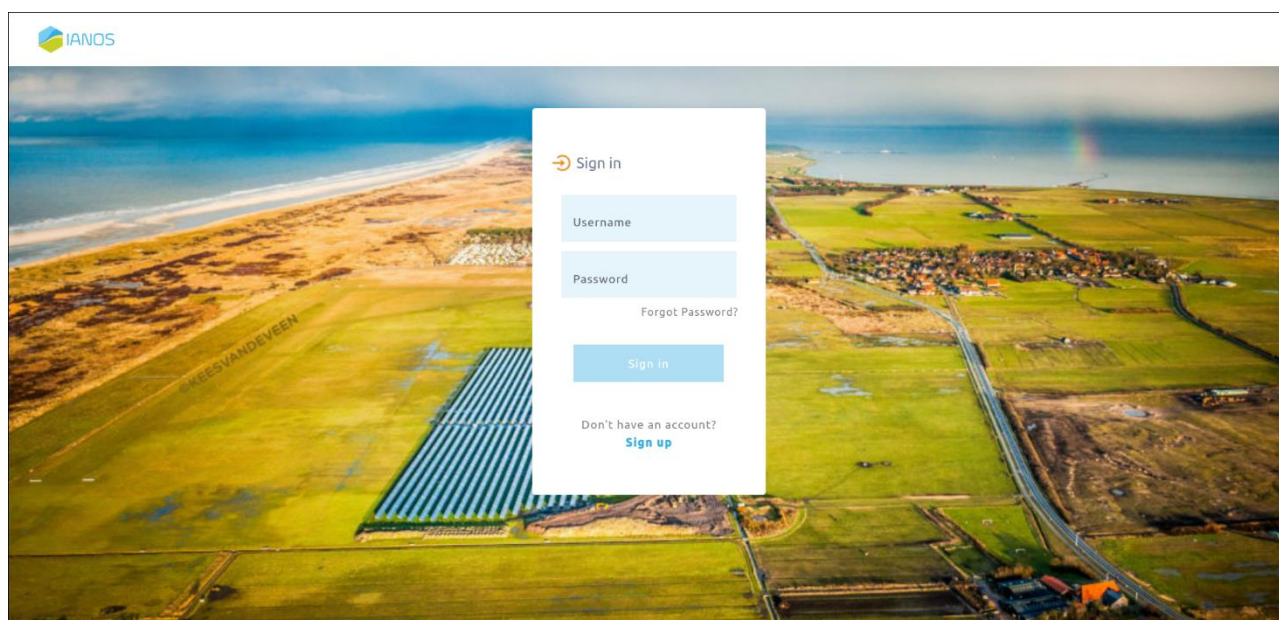


Figure 7: Login page of the IEPT Suite dashboard.

Upon successful login, the user can have an overview of the CBA module results, as seen in Figure 8.

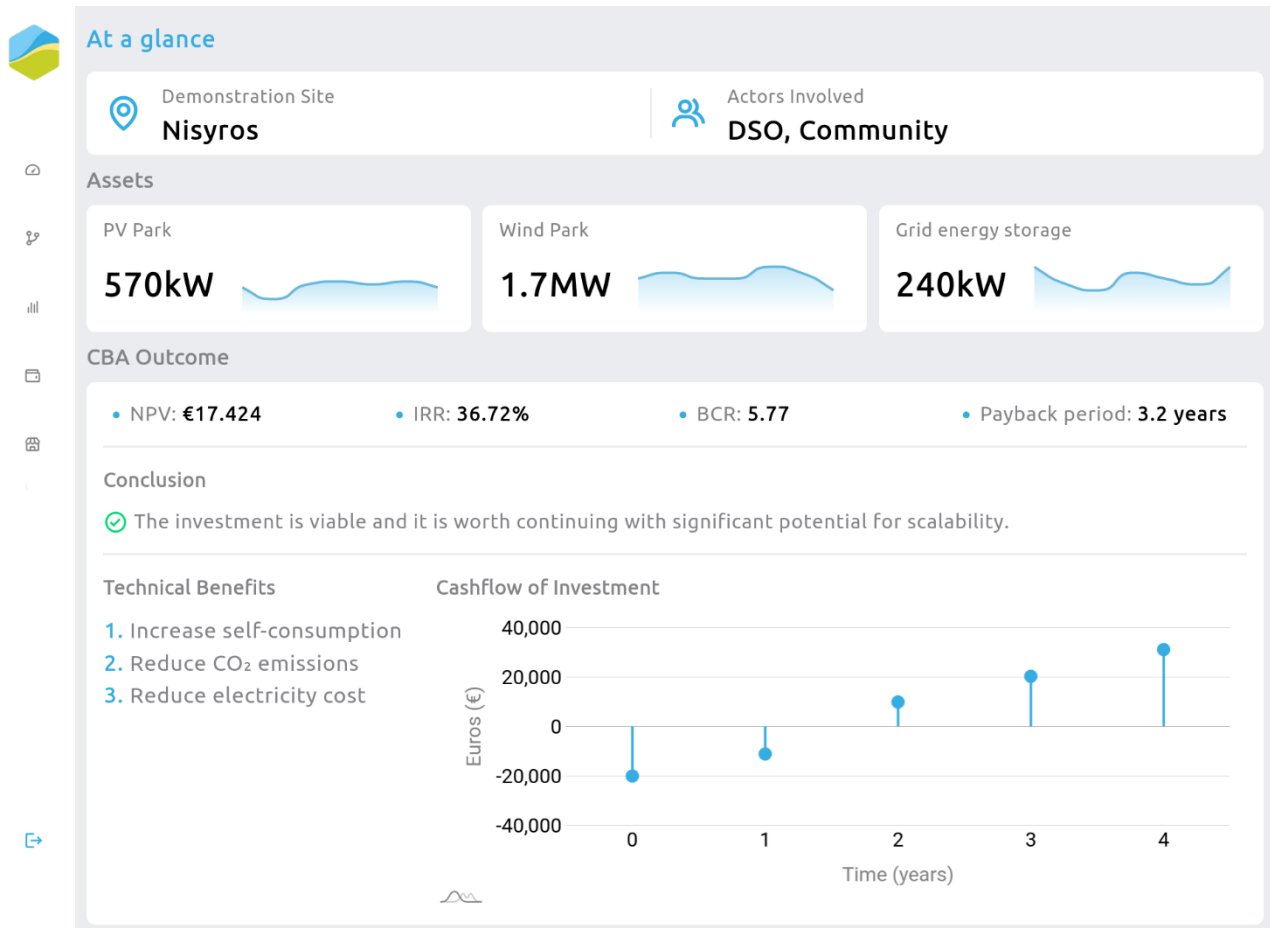


Figure 8: "At a glance" page of the IEPT Suite dashboard.

Specifically, the users can see information about the demonstration site, the actors involved in the investments and the specific assets that were added into the CBA. The CBA outcome includes values, such as NPV, IRR, etc., a graph with the cashflow of the investment and the final conclusion of the CBA. There is also a list of the technical benefits that the actors will achieve with the addition of the selected assets.

## 3. CBA module

### 3.1 Overview

Smart grid technologies can enable higher levels of renewables in electricity systems by making the system more flexible, responsive and intelligent. As more and more countries, particularly in the developing world, plan to increase their use of renewables, smart grid technologies provide the means to integrate these renewables in a cost-efficient and effective way. Smart grid projects are often evaluated and justified on an economic basis. The challenge for decision-makers (which can be utilities, policymakers or building-level consumers) is to evaluate smart grid proposals rigorously, objectively and with a well-defined and consistent methodology. Such analyses are critical for ensuring that scarce capital is invested wisely.

CBA is an ideal first tool for evaluating a smart grid investment. Its value lies not just in the provided outcome, but also in how it requires one to define and quantify the expected costs and benefits. Often it is this analytical discipline, rather than the result itself, that is most informative and offers the various stakeholders a clear path towards the integration of smart green solutions in the Business-as-Usual (BaU) operations. Before conducting a holistic CBA, one needs to consider several issues, such as the follows:

- Firstly, different stakeholders will value the benefits of a smart grid differently. A societal perspective will account for all benefits, however, one may want to take a narrower perspective, such as that of the utility or the electricity users.
- Secondly, undertaking a CBA requires careful definition of a baseline, documenting what would happen in the absence of the smart grid project. This is a very tricky part, sometimes ignored completely during the first stages of the analysis, rendering inadequate the proper evaluation of the interventions.



- Thirdly, CBA requires considerable judgment on the part of the analyst, particularly in estimating uncertain inputs and assessing qualitative benefits. CBA can support better decisions, but it should not be used to make decisions on its own.

The CBA methodology was first introduced for Transmission System Operators (TSOs). Transmission system development focuses on the long-term preparation and scheduling of reinforcements and extensions to the existing transmission grid. The identification of an investment need is followed by a project promoter(s) defining a project that addresses this need. Following Regulation (EU) 347/2013, these projects must be assessed under different planning scenarios, each of which represents a possible future development of the energy system [7].

Whilst their costs mostly depend on scenario independent factors like routeing, technology, material, etc., benefits strongly correlate with scenario-specific assumptions. Therefore, scenarios that define potential future developments of the energy system are used to gain an insight in the future benefits of transmission projects. The essence of scenario analysis is to come up with plausible pictures of the future.

### **3.2 Functionalities and Methodological Approach**

The proposed CBA methodology, presented in Figure 9, is based on both JRC's and ENTSO-E's methodologies [8], [9]. These are comprehensive frameworks to assess costs and benefits of Smart Grid Interventions inserted into the electricity infrastructure from the system planning perspective. The analysis accounts at the same time for technical, economic, reliability, environmental and security benefits. This framework aims at evaluating its effectiveness in the IANOS project demonstrators, i.e., LH islands of Terceira and Ameland and the defined UCs.

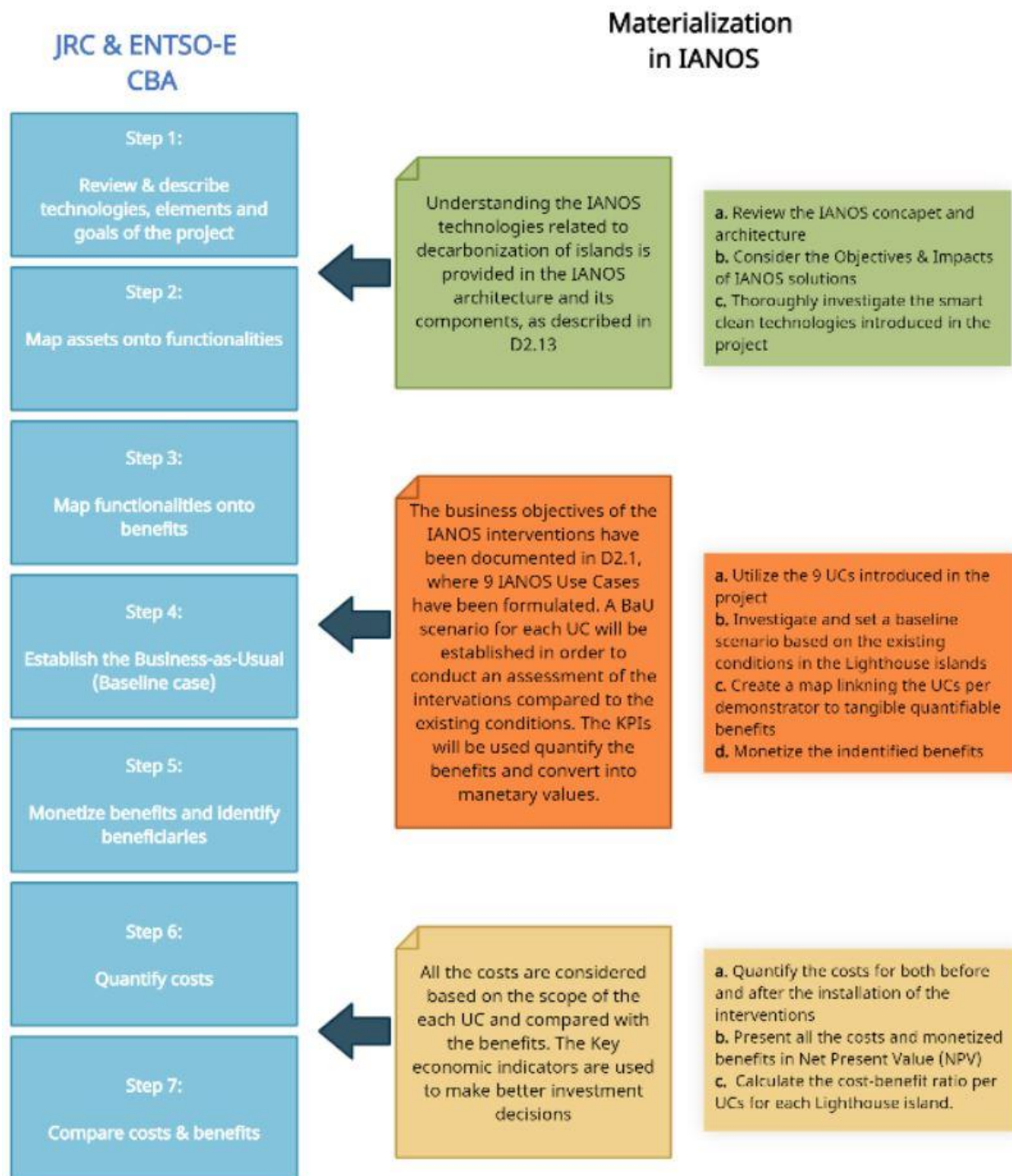


Figure 9: The CBA methodology and its materialization in the IANOS project interventions.

The economic indicators that are leveraged to perform the CBA are introduced. Firstly, the Discounted Cash Flow (DCF) model is introduced. This model is used to compute the present value of future cash flows, resulting from Smart Grid Interventions for each year of the analysed period. The DCF model is a basic evaluation model for an asset that is expected to generate cash payment in the form of cash earnings, interest and principal payment or dividends. The mathematical formulation of the DCF models is:



$$V_o = \frac{CF_1}{(1+r)^1} + \frac{CF_2}{(1+r)^2} + \dots + \frac{CF_n}{(1+r)^n} \quad (1)$$

where  $V_o$  is the present value of the anticipated cash flows from the asset,  $CF_{1,2,..,n}$  are the cash flows expected to be received up to  $n$  periods in the future and  $r$  the discount rate, which is the required rate of return per period.

After the computation of the DCF for each year, the Net Present Value (NPV) for each benefit is computed. The NPV of a capital budgeting project is the monetary amount of the change in the value of the firms because of undertaking the project. In the case of the IANOS project, the NPV depicts the monetary amount of the change in the value of the energy infrastructure because of selecting and applying one or more Smart Grid Interventions, according to the business objectives of each introduced IANOS UC. A positive NPV value showcases that the value of the infrastructure will increase if the new interventions are adopted. On the other side, a negative NPV means that the infrastructure's value will decrease if the new interventions are adopted. The specific formula for the NPV calculation is given by:

$$NPV = \frac{CF_1}{(1+r)^1} + \frac{CF_2}{(1+r)^2} + \dots + \frac{CF_n}{(1+r)^n} - \text{Initial Investment} \quad (2)$$

where,  $CF_{1,2,..,n}$  is the cash flow at the indicated times,  $r$  is the discount rate and  $n$  is the life of the technology measured in number of years.

The discount rate  $r$  is the rate of interest that determines the present value of future cash flows. A low interest rate indicates promoting trend of development created by central banks. A high interest rate shows inflationary, riskier and uncertain trends.

The selection of time horizon is of vital importance and has to be defined within the benefits and the costs of each intervention. The time horizon depends on the asset type of each installation, the investment scheme and the benefits for the

overall electricity infrastructure (and in general energy infrastructure if benefits can be found in the rest energy carriers).

In order to conduct the assessment between the existing BaU condition and the benefits that the IANOS interventions will bring in the buildings, communities, regions and islands in total, the following indicator is introduced:

$$\text{Benefit – cost ratio (BCR)} = \frac{NPV(\text{Quantified Benefits})}{NPV(\text{Costs})} \quad (3)$$

Table 14 provides a roadmap for the investors to assess whether a smart grid intervention for the specific setup that is described in the demonstration campaigns of the project, is recommended.

Table 14: Recommendation for investments based on BCR values.

BCR value	Investment recommendation
$<<1$	Not worth continuing the investment and potentially increased scalability will not benefit either.
$\sim \leq 1$	Not worth continuing the investment with the existing setup.
$\sim \geq 1$	Risky to continue the investment (this implies that relatively low IRR exists).
$>>1$	It is worth continuing the investment.

According to ENTSO-E's CBA methodology, the KPIs have been grouped in three distinctive categories, i.e., Benefits, Costs and Residuals, based on which type of impact each KPI can quantify. Each UC of the IANOS project is linked to specific KPIs that quantify the business impact of the interventions. The KPIs that will be used as the outcome of the Energy Planning & Transition Decision Support suite, will be computed using the functionalities of the components that constitute the suite. Specific description is provided in Chapter 2 of this deliverable. Figure 10 provides the position of the CBA component in the suite and the way it uses the outcome of the rest components, i.e., the calculated KPIs values, in order to conduct a CBA for the IANOS solutions. The benefits, costs and residual impacts

columns include the specific categories that were identified during the pre-validation and refinement activities of the IEPT tool (related to task 3.4), where a cross-checking process was conducted to provide a list of specific monetizable KPIs for each UC.

The following chapters of this deliverable contain an extended description of the functionalities of the modules that are responsible for the KPIs calculation. Table 15 includes the KPIs list that contains the ones that will be used in the context of IEPT suite. The relation of those KPIs with the CBA methodology is presented in the last column of the Table. The role of the CBA component is to convert into monetary gains (if possible) all the impacts of those benefits for each UC and to provide a holistic tool for investment planning for the different stakeholders. In case a benefit/residual impact cannot be directly converted into monetary gains, a scaler is introduced in order to provide an auxiliary tool to the decision-makers by showcasing the improvements/positive impacts that a specific IANOS intervention will make.

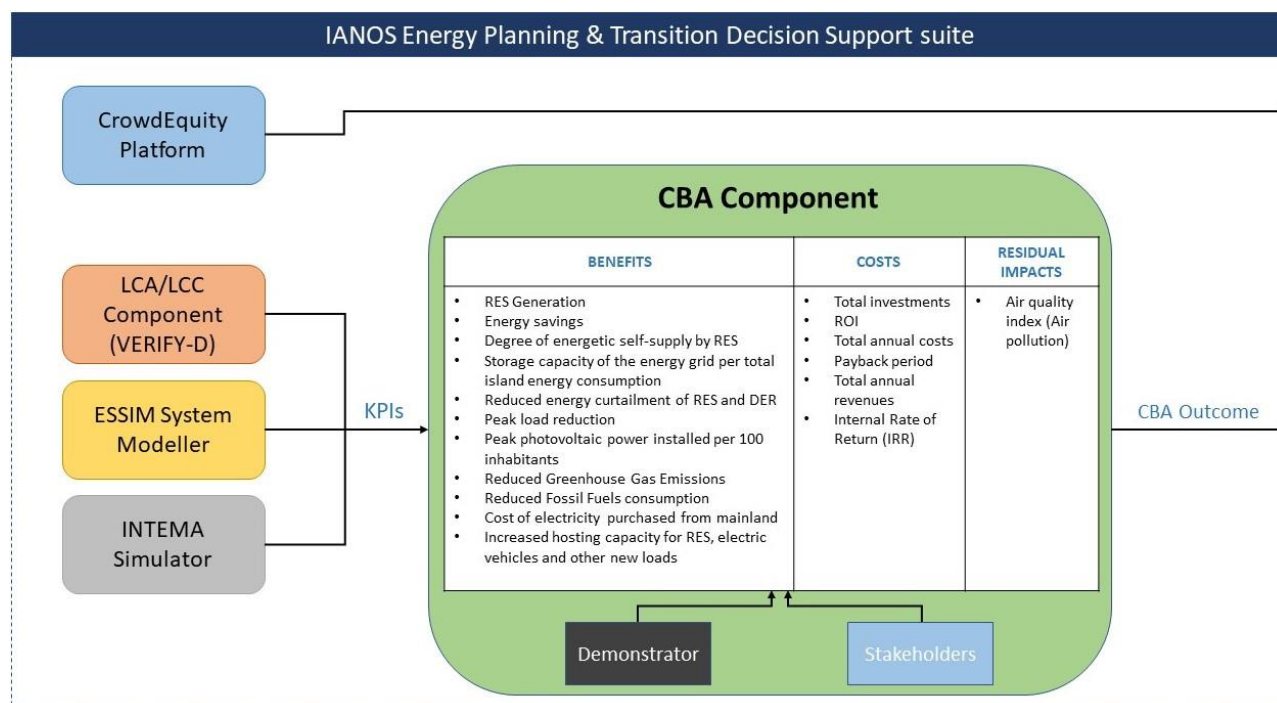





























Figure 10: Position of the CBA component in the IEPT suite and its interconnection with the rest components.

Table 15: Mapping of KPIs to the specific components of the IEPT suite.

KPIs Information		Calculated <sup>4</sup>			Mapping to CBA
Categories in D2.9	KPI NAME	INTEMA	VERIFY	ESSIM	Categories
T1	RES Generation				<i>Benefits</i>
T2	Energy savings				<i>Benefits</i>
T5	Degree of energetic self-supply by RES				<i>Benefits</i>
T7	Storage capacity of the energy grid per total island energy consumption				<i>Benefits</i>
T8	Reduced energy curtailment of RES and DER				<i>Benefits</i>
T9	Peak load reduction				<i>Benefits</i>
T12	Peak photovoltaic power installed per 100 inhabitants				<i>Benefits</i>
EN-1	Reduced Greenhouse Gas Emissions				<i>Benefits</i>
EN-2	Reduced Fossil Fuels consumption				<i>Benefits</i>

<sup>4</sup> VERIFY will conduct the KPIs calculation for both LH islands of the IANOS project. INTEMA will conduct the KPIs calculation for the LH island TERCEIRA and the ESSIM modeler for the LH island AMELAND. More information is given in D3.8 (updated version of D3.7).

EN-4	Air quality index (Air pollution)				<i>Residual</i>
EN-6	Primary Energy Demand and Consumption				
EC-1	Total investments				<i>Costs</i>
EC-2	ROI				<i>Costs</i>
EC-3	Total annual costs				<i>Costs</i>
EC-4	Payback period				<i>Costs</i>
EC-5	Total annual revenues				<i>Costs</i>
EC-8	Internal Rate of Return (IRR)				<i>Costs</i>
EC-10	Cost of electricity purchased from mainland				<i>Benefits</i>
I-4	Increased hosting capacity for RES, electric vehicles and other new loads				<i>Benefits</i>

### 3.3 Implementation and technology stack

The backend of the CBA module was developed as a web app using the Django Framework in Python. This is responsible for sending API requests to other microservices-modules, processing and analysing the received information in json format and computing the necessary values for the CBA. The dashboard was

developed in Typescript using the Angular framework in conjunction with the Angular Material UI Component library and the amCharts5 library for the required figures and diagrams. The user authentication was implemented with the integrated user authentication system offered by the Django framework. All the KPIs received from the other modules and the results of the CBA are stored in a PostgreSQL database.



## 4. ESSIM module

### 4.1 Overview

The Energy System Simulator (ESSIM), as illustrated in Figure 11, is a discrete time simulation tool and collection of models that calculates energy flows in assets and the effects thereof, in an interconnected hybrid energy system over a period of time. With the help of the energy flows ESSIM calculates, one can get insights into how well the assets in a network are dimensioned, if there is overloading in any given transport asset (like pipe, cables, etc.) and what is the effect of storage in any part of the network.

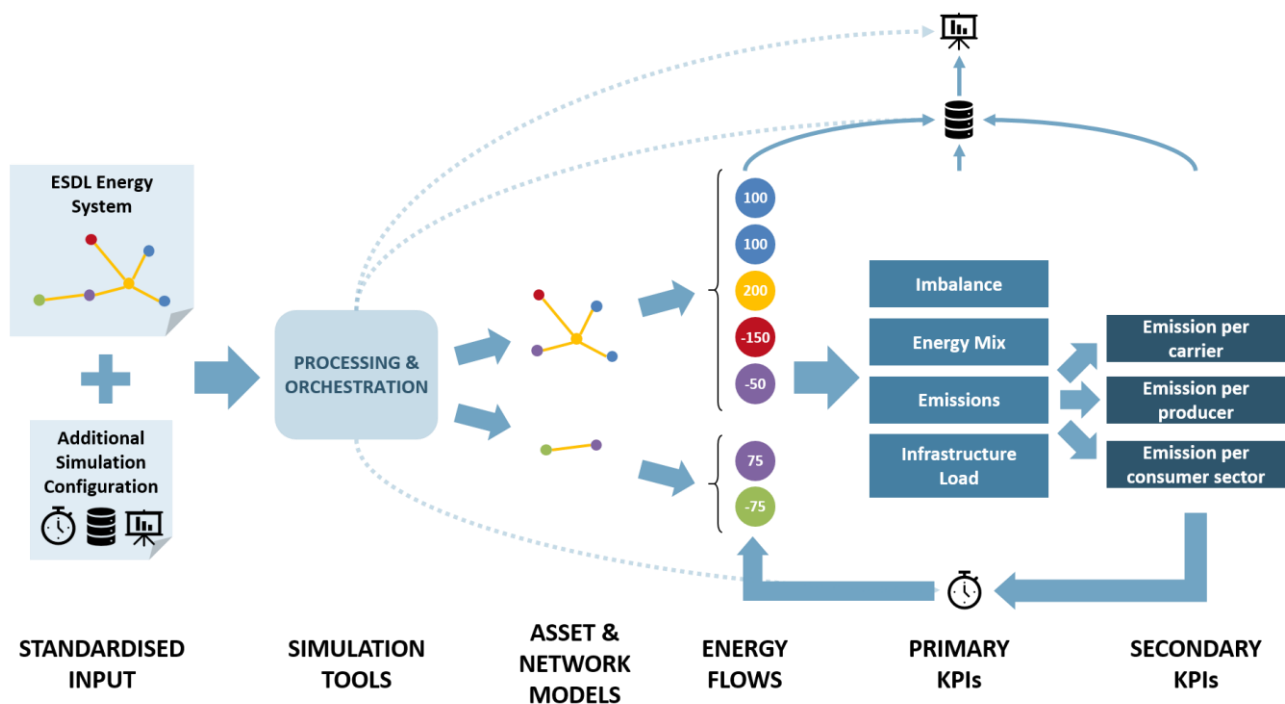


Figure 11: An overview of the functionalities of ESSIM.

As an input, ESSIM uses an energy system modelled using Energy System Description Language (ESDL) to get information about energy assets, carriers and commodities, infrastructure etc. An energy system is modelled in a map-based UI tool called MapEditor, that ultimately generates an ESDL. The following two subsections describe ESDL and MapEditor.

## ESDL

ESDL is an open-source modelling language created for modelling the components in an energy system and their relations towards each other. Technically it is an XML schema definition (XSD) for formally describing energy assets in an XML format. This allows ESDL to be used as a formal specification of an energy grid for unambiguous interpretation by experts and tools. The language is machine readable so makers of energy transition calculation tools, simulations and GIS applications can support ESDL in order to enforce the interoperability of their products.

ESDL is capable of expressing the dynamic behaviour of components in the energy system. For instance, the power consumption of a neighbourhood. ESDL describes components by their basic functionality (Energy Capabilities), these are modelled in 5 abstract categories: Production, Consumption, Storage, Transmission and Conversion. Aggregation and composition principles of ESDL enables energy modelers to model a complex energy system in a generic way.

ESDL can be used in various ways, for example:

- By energy transition calculation tools. ESDL acts as a common language for different energy transition calculation tools, specifying the format of the inputs and outputs of such tools. This allows for integration of multiple tools.
- In an Energy Information System. ESDL can be used as a basis for a central energy information system where the energy system of a certain region is registered.
- As a language for (local) governments to model, reason about and share their (local) energy system.
- To monitor the evolution of an energy system: multiple ESDL snapshots of a certain area over time provide insight in the evolution of an energy system.
- As a format to share data relevant to energy systems or the energy transition. Examples:
  - CO<sub>2</sub> emissions per energy carrier.



- Technology factsheets for specific components, brands, types (e.g., a heat pump factsheet that describes its typical parameters).
- Cost information of assets or expected cost developments in the future.
- Standard configurations or templates of typical parts of the energy system (e.g., a house with a heat pump, solar panels and an EV charging station).

A fully detailed description of ESDL, both technical and pragmatic, can be found as an on-line gitbook on [8].

### ***ESDL MapEditor***

ESDL MapEditor is a map-based energy system editor. With the MapEditor ESDL based energy system descriptions are defined, that can be simulated with a growing number of ESDL capable simulators, among which is ESSIM (section 4.3). The editor allows to define the system on a map, thus placing assets in their real-life location. These assets can be anything from production, consumption, infrastructure (transmission), conversion and storage for every imaginable energy carrier. The editor provides features to intuitively construct the system, work with Geographic Information System (GIS) components (WMS layers, shapefiles and more) to assist in visualization of the system. Also, the editor is connected to external data sources (on buildings for example) and to external models (e.g., spatial analysis models). The editor further supports configuring the assets in the system with energy profiles of energy demand, consumption for every applicable asset or any other data that is relevant for further processing/simulating the ESDL energy system. When ESSIM is used for simulation, the results can be visualized in the MapEditor by selecting specific assets and viewing the simulation results in the form of charts and graphs which makes analysis of the system possible. An open-source version of the ESDL MapEditor can be found on [9].

## **4.2 Functionalities**

ESSIM calculates energy flows per time step for a simulation period. It takes as inputs the energy system defined in ESDL and calculates optimal schedule of



flexible producers and the effect of this schedule in terms of emissions, costs, load on the network, etc. (Figure 12 shows an ESSIM output example). Thanks to the easy configurability of the energy system with the help of the ESDL MapEditor, the user can use ESSIM to perform “what if?” scenario analyses on current and future energy systems.

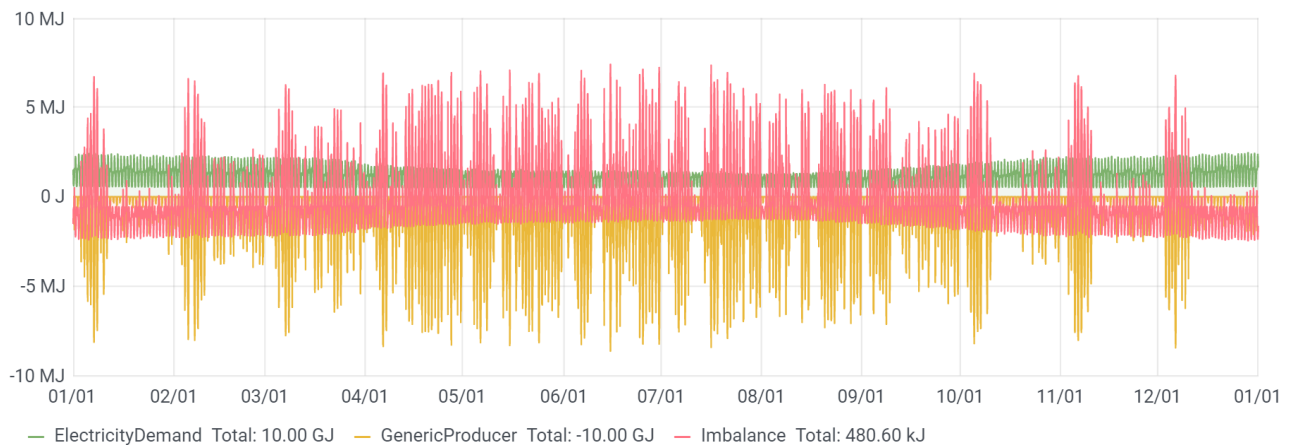


Figure 12: ESSIM output: Hourly data showing imbalance.

Along with the primary KPIs that ESSIM generates (Energy mix, network imbalance, emissions, etc.), external KPI modules connected to ESSIM also allow for post-processing ESSIM data to get measurable insights into the energy system variations (Figure 13 shows the design phase workflow). Results of KPI modules are in JSON format and as such can be communicated with external tools.

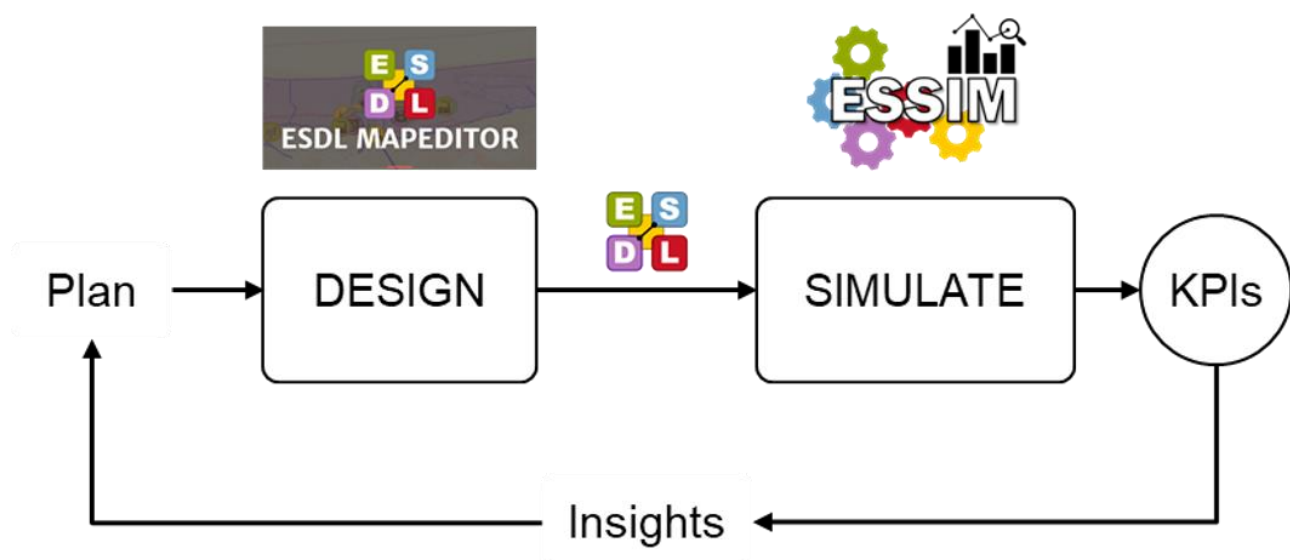


Figure 13: ESSIM used in the planning-design phase of Energy Transition.

## Energy Assets

ESSIM follows the ESDL principle of classifying energy system assets into Producers, Consumers, Storages, Conversions and Transport (see Figure 14). Each asset (except for Transport assets) can either have a pre-determined behaviour (inflexible) programmed into it with the help of a profile or let ESSIM calculate its energy allocation (flexible) based on its flexibility, some control strategy and the behaviour of other assets in its network. ESSIM contains models for each of these five classes of assets and thanks to ESDL's hierarchical data structure, model behaviour can also be inherited from parent models. E.g., with a generic Consumer behaviour implementation, all existing ESDL Consumers such as ElectricityDemand, HeatingDemand, etc., and any ESDL Consumers added in the future can be supported within ESSIM. Additionally, implemented within ESSIM are also some specific asset models that have certain peculiarities (see Figure 14). E.g., a co-generation plant is a conversion asset with two output ports with one output effecting the other.



Figure 14: All implemented asset models in ESSIM.

## Transmission Network

In ESSIM, a transmission network is an atomic network of connected energy assets using the same energy carrier. This means ESSIM treats isolated assets physically connected only to each other but using the same energy carrier as separate transport networks. Internally ESSIM creates a tree of such a network where the Producer/Consumer/Conversion/Storage assets are the leaf nodes, and the Transport assets are the branches. Conversion assets appear in multiple transport networks as consumers of some energy carriers and producers of some others. Based on the provided ESDL description and configured behaviour, each transport network attempts to balance itself in each time step.

The transport network trees are created in the pre-processing step in ESSIM and cannot be changed during the simulation. In addition to creating the trees, ESSIM also pre-determines the order and parallelisation in computing the network tree balances, as represented in Figure 15.

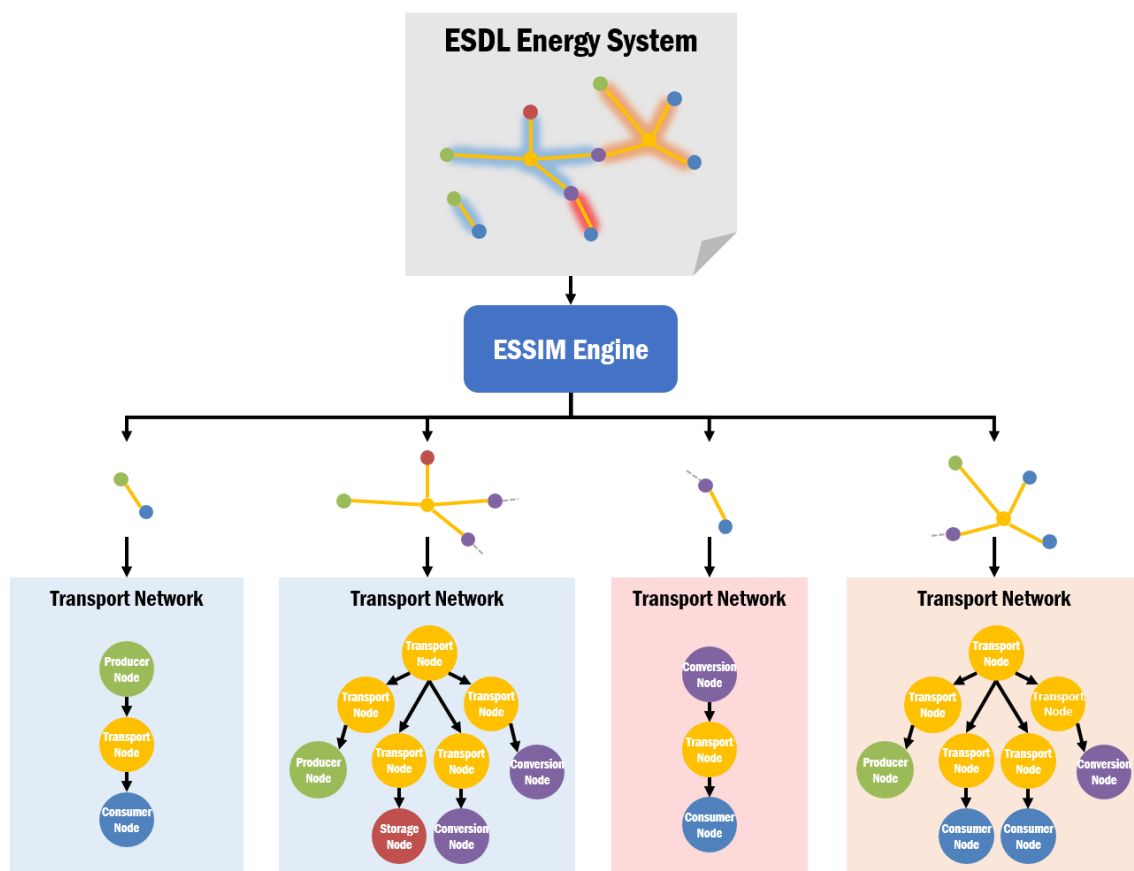


Figure 15: Splitting of an energy system into constituent transport (transmission) networks.

### 4.3 Implementation and technology stack

ESSIM is implemented in Java. At the heart of the ESSIM tool are:

- A simulation engine that orchestrates a fixed discrete timestep evaluation of each commodity network's balance in a pre-determined order,
- A flexibility-based demand-supply matching algorithm that uses marginal costs of energy production as a means to grade desirability of producers,
- A tree-based transport network solver that calculates the load on various transport elements based on the demand-supply solution determined above.

The tool outputs the data provided and generated during the course of the simulation into a timeseries database (InfluxDB) and generates a dashboard (Grafana) to visualise the same. ESSIM allows for connecting external asset models (via an MQTT interface) and external network (solver) models (via a REST interface), alongside its internal models and while using its own orchestration mechanism. This makes ESSIM a viable candidate for a co-simulation orchestrator.

Figure 16 shows the technology stack of ESSIM, including KPI Modules that calculate and communicate KPI results with ESSIM and the underlying database. As seen in Figure 16, each interacting component is Dockerized.

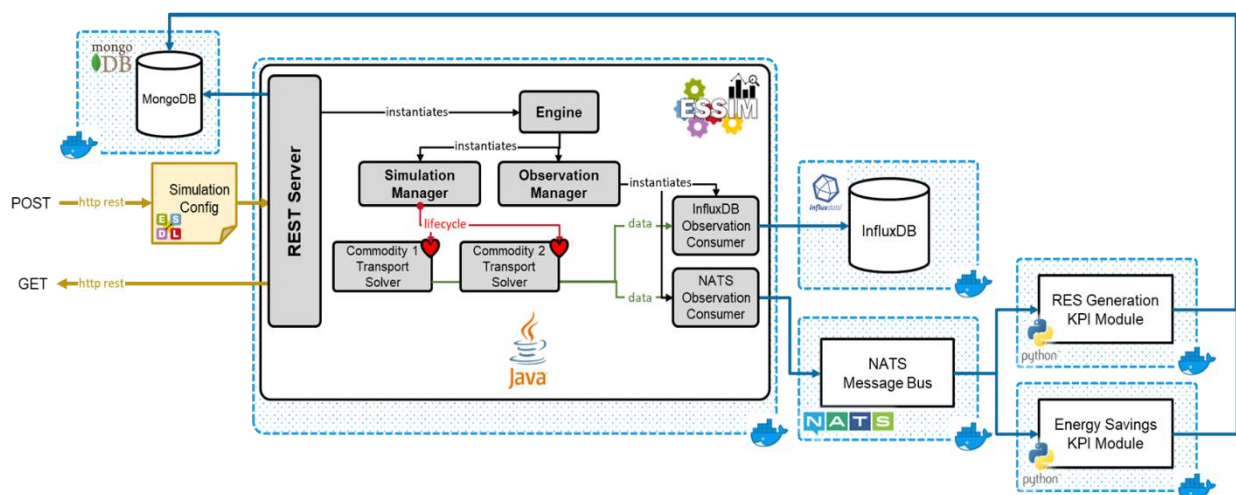


Figure 16: ESSIM Technology stack.

The KPI Modules subscribe to ESSIM to receive simulation results at each time step. This subscription and message communication is performed via NATS message bus. Once the simulation is finished and all the messages (results at each timestep) are received, KPI Modules perform their calculation and write the result in MongoDB. Using a KPI ID and a simulation ID, an application can send a GET request to ESSIM to retrieve KPI results for a specific simulation. ESSIM fetches these results from MongoDB and sends it back to the requesting application. The format of the GET request is:

*essim/simulation/<simulation-id>/kpi/<kpi-id>*,

where *<simulation-id>* and *<kpi-id>* are the ID of an ESSIM simulation and the desired KPI ID, respectively.

With respect to the message request specification presented in Table 9, a lookup Table should be used to perform the mapping between the desired IANOS KPI IDs (e.g., TI) and the ESSIM KPIs (e.g., RES Generation), as well as the Use Case specification and ESSIM simulation IDs.

## 4.4 User interface

ESSIM is a simulation tool that can communicate via a REST interface with various other tools to perform energy flow calculations. It starts its calculations once it receives a POST request from another tool. As such, ESSIM does not have its own dedicated user interface. The user interface most frequently used is MapEditor, as explained in Section 4.1, ESDL MapEditor. Using MapEditor, a user can model an energy system and run an ESSIM simulation. Figure 17 shows how an ESSIM simulation is run using MapEditor. A user can edit a number of parameters, such as the description of a simulation and the time range for which it is ran. Once the simulation is configured, the simulation progress can be seen (see Figure 18).



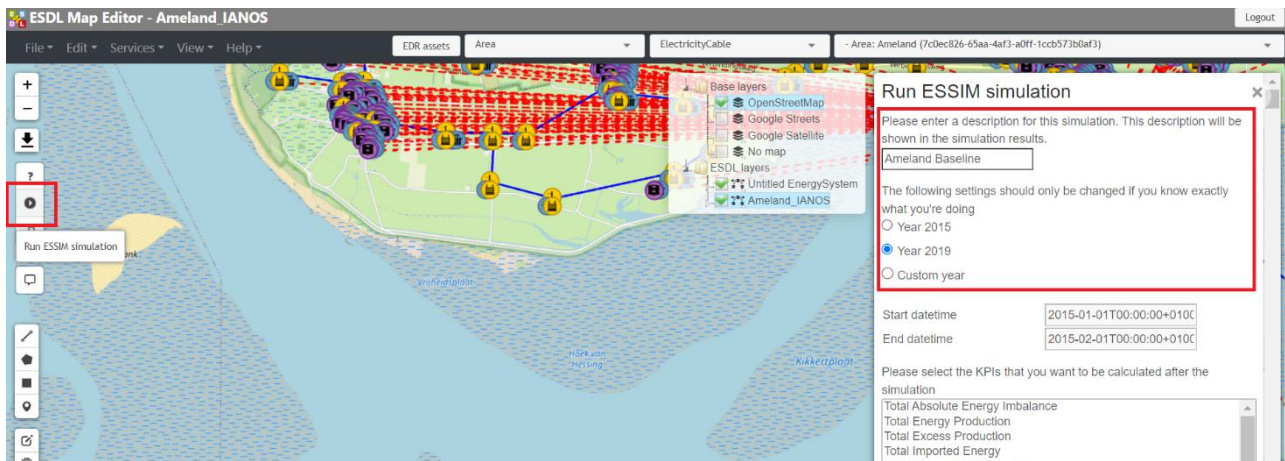


Figure 17: Running ESSIM from MapEditor.

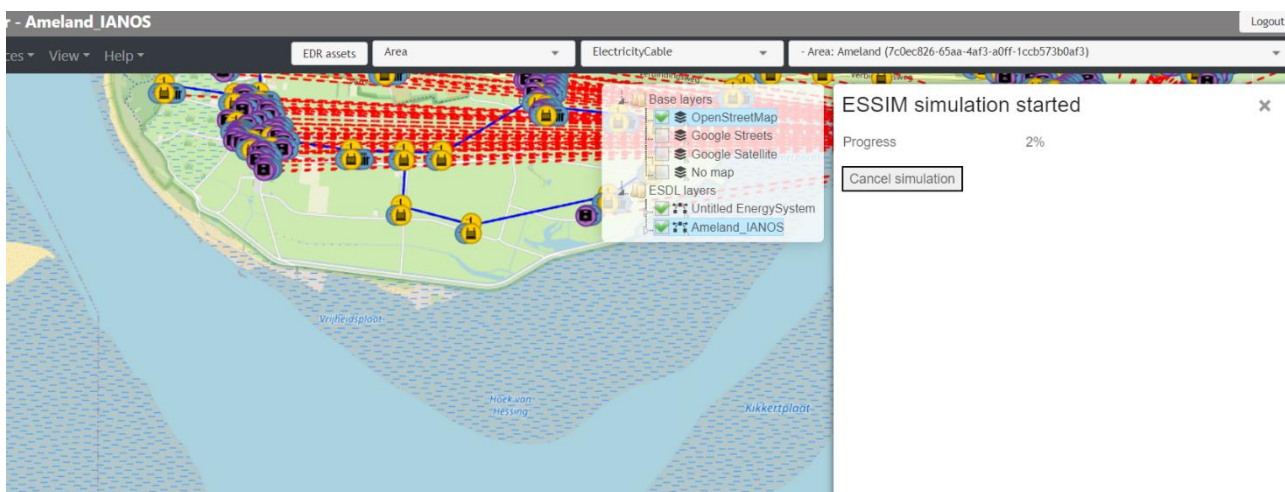


Figure 18: ESSIM simulation in progress.

However, running an ESSIM simulation using MapEditor is only one of the options. In principle, any tool that can send an ESDL and a GET request containing ESSIM configuration parameters (including an ESDL file) can start an ESSIM simulation. Figure 19 shows a GET request example sent through Postman API platform to start an ESSIM simulation.

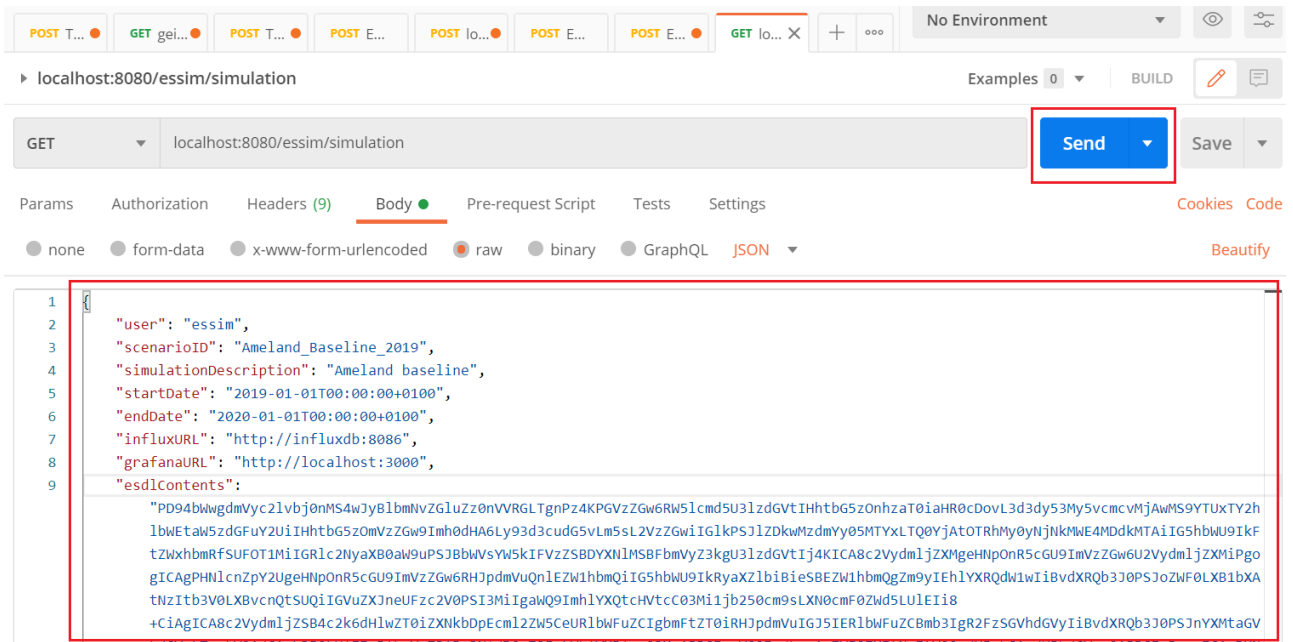


Figure 19: Running ESSIM using Postman API platform.



## 5. INTEMA.grid module

### 5.1 Overview

CERTH has been developing its own Suite of software Tools for the energy sector under the name INTegrated Energy Management (INTEMA). INTEMA.grid is part of the INTEMA Suite, with specific implementations and tools for grid-level simulations, able to simulate multi-vector integrated energy systems, accounting also for the grid topologies and interconnections among variable systems. INTEMA.grid aims at helping engineers to address challenges towards a sustainable energy transition, while being developed to be fully customizable by energy experts, rendering it in a position to include any type of system simulation (mature or highly innovative), on multiple spatial and temporal scales (both steady and transient simulations, with a user-defined timestep). The tool offers a set of functionalities for an energetic grid assessment, each one covering different needs of the decision-making process.

Since the launch of IANOS project, CERTH has concentrated its efforts in two main directions:

- i) To be capable of fully supporting the dynamic simulation of Terceira lighthouse along with the three fellow islands, i.e., Bora-Bora, Nisyros and Lampedusa, by developing the corresponding component models for the technologies (mature and innovative) that are present.
- ii) The development of a web-based GUI that will be supporting the scenario setup, simulation and assessment. Under this view, a back-end automated engine is still under development, able to automatically generate the corresponding system model to be simulated, accounting also for user-friendliness and customizability, covering all typical grid cases.

The following sections report in detail the developments conducted and highlight the specific advancements occurred in the context of Task 3.3 of IANOS.

This deliverable (D3.6), which is the updated version of D3.5, also includes the completion of the web application development (both front- and back-end) that assists the automation of the simulation procedure and enables rapid scenario definitions and assessment, primarily for the case of the Fellow islands. Although several changes and corrections have been applied throughout the entire document, major revisions and additions are present in Sections 5.3 and 5.4, where the technology stack and the user interface are discussed, respectively.

## 5.2 Functionalities

The present section lists the main functionalities of INTEMA.grid along with a short description and some implementation info. Although specific functionalities can be used in sequence (e.g., optimal power flow, followed by a dynamic simulation), the functionalities are independent of each other. Table 16 lists all the available functionalities of the INTEMA.grid tool. Furthermore, it describes the functionalities that are supported through the web interface. Features annotated with an asterisk are also supported via the web-interface despite not being considered in the initial plan.

Table 16: Functionalities attributes and development timeframe for IANOS.

Functionality	Free and open-source tools utilization	Developed by CERTH		Automation through the web-interface
		Developed in previous projects	Developed during IANOS	
Power Flow	✓	-	-	✓*
Optimal Power Flow	✓	-	-	✓*
Dynamic Grid Analysis	-	-	✓	✓
Forecasting	-	✓	-	-
Web Interface	-	-	✓	✓

IEEE Common Data Format	✓	-	-	✓*
-------------------------	---	---	---	----

- Power Flow: A Power Flow (PF) study, also known as load flow study, is a fundamental analysis performed in electrical power systems to determine the steady-state behaviour of the system under normal operating conditions. It provides an analysis of the most important and essential variables in the power system operating phase by calculating all major quantities (e.g., active and reactive power, voltage angle etc.) at each node (bus) of the system. INTEMA.grid offers PF functionality with the integration of the popular open-source Python module Python for Power System Analysis (PyPSA) [12].
- Optimal Power Flow: While energy forecasts are the first step to minimizing the cost, particular decisions must be made regarding the coordination of the utilized power plants. Such decisions involve solving an optimal resource allocation problem, while considering the power plants' operating costs. The control and coordination of the grid assets can either be centralized into a single decision-maker or distributed [14]. INTEMA.grid offers an efficient and reliable solution to the Optimal Power Flow (OPF) problem. Taking advantage of modern convex optimization methods [15], the tool is able to address the problems of unit commitment, i.e., generator start-up and shut-down, and economic dispatch, i.e., determining the power output of each generator, in order to meet the power requirements at minimum cost, within the constraints of the network [16]. Optimization can be carried out for both pure or mixed DC and AC networks on the basis of linearized power flow equations. The optimization method assumes that the loads are non-dispatchable and load requirements must be met at all time snapshots.
- Dynamic Grid Analysis: INTEMA.grid's power grid simulation engine leverages Modelica language and offers detailed dynamic simulations considering variables such as: a) grid assets (lines, transformers, generators, capacitor banks, loads etc.), b) conventional and renewable energy generation and c) energy storage units. The tool is capable of estimating

with high accuracy all relevant quantities regarding energy management, voltage/frequency stability and power quality. At the same time, the high order dynamic representation enables stability assessment.

- Forecasting: The stability of the electricity grid requires the electricity supply to constantly meet demand. Thus, forecasting energy production and consumption have always been of great interest. INTEMA.grid offers in-house developed power forecasting modules for load and RES generation (wind and solar energy) based on machine learning algorithms [17]. The models have been developed in the context of the Horizon projects SMILE [18] and RENAISSANCE [19].
- Web Interface: A convenient web-based graphical user interface has been developed to support the case definition and results visualization. Its features and functionalities will be described in detail in Section 5.4.
- IEEE Common Data Format: The scenario definition for a grid scale simulation involves the import of a large set of information. An automated data import is supported based on the IEEE Common Data Format (CDF) [21], which is a well-established format and is frequently used by the industry and academia. The user can upload a .cdf file containing all buses, lines, transformers and generation/load definitions. An important note is that its standard format regards mainly steady state information; thus, if the user aims to perform dynamic simulation, additional data should be defined.

## 5.3 Implementation and technology stack

An important requirement in the development of INTEMA.grid is the utilization of open-source software wherever possible. This requirement offers a set of valuable attributes in the development process. First, it allows the access and modification of the source code, giving greater control and flexibility over how the software operates. This can lead to faster development and customization of the software to meet specific needs. At the same time, open-source software also tends to have a strong community of developers and users who contribute to its development and maintenance, ensuring that it remains valid, up-to-date, and secure over time. Finally, open-source software promotes transparency and collaboration,

allowing users to share their knowledge and expertise with others, and fostering a culture of innovation and creativity.

INTEMA.grid architecture follows the common practise of separation of concerns between the presentation layer (front-end) and the data access layer (back-end) and inherits all advantages of this design. The two layers communicate asynchronously via RESTful API. A schematic high-level overview of the architecture is presented in Figure 20.

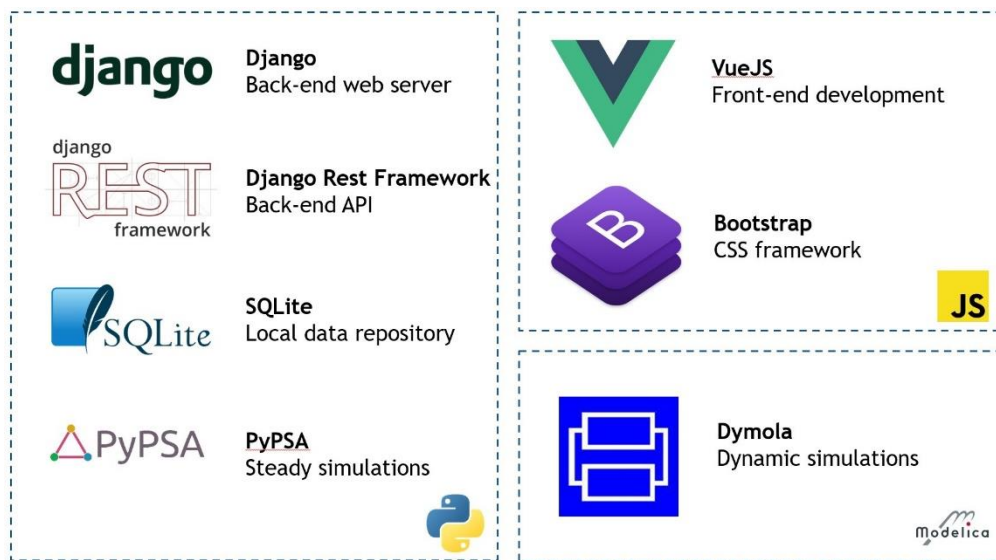


Figure 20: INTEMA.grid technology stack.

The back-end is developed in Python utilizing the popular Django web framework in addition to Django Rest Framework toolkit which enables the API functionality. The underlying database engine is SQLite, enabling the rapid prototyping, experimentation and development. Still, if and when storage or performance limitations occur the mitigation to another database engine (e.g., PostgreSQL, MySQL, MongoDB etc.) is straightforward, supported by the Django framework. Several Python modules are also utilized in the backend processes. A noteworthy mention is the PyPSA module, which is a toolbox that offers power flow simulations and optimization functionalities.

A major feature of INTEMA.grid is the integration of the Modelica dynamic modelling language. Thus, INTEMA.grid's back-end takes advantage of the

extensive Modelica code repository and builds upon well-validated open-source libraries in the domain of energy generation, distribution and storage. The tool uses the PowerSystems (PS) library [22], as well as in-house developed models. Despite its highly technical nature, the present section aims to give an overview of the most important models utilised for the dynamic simulations. Table 17 lists all the main components utilized in INTEMA.grid dynamic simulation engine, categorizing them on whether they were developed by CERTH or not. Appendix A provides an extensive description for each component.

*Table 17: Dynamic models development timeframe.*

Category	Component	Open-source component	Developed by CERTH	
			Developed in previous projects	Developed during IANOS
Passive grid components	Lines	✓	-	-
	Transformers	✓	-	-
	Loads	✓	-	-
Conventional power plants	Steam power plant	✓	✓	-
	Gas power plant	✓	✓	-
	Diesel power plant	✓	✓	-
Renewable power plants	Photovoltaics	✓	-	-
	Wind turbines	-	✓	-
	Hydroelectric power plants	✓	✓	-
Energy storage	Battery systems	-	✓	-
IANOS technologies	Biobased Saline	-	-	✓
	Hybrid heat pump	-	-	✓
	Flywheel	-	-	✓

	Hybrid Power Electronic Transformer	-	-	✓
	Solid Oxide Fuel Cell	✓	-	✓
	Proton Exchange Membrane Electrolyzer Cell	-	✓	-
	Geothermal Power Plant	-	-	✓
	MicroCHP	✓	-	-
	PCM storage	-	✓	-
	Autogenerative High Pressure Digestion	-	-	✓

The front-end is based on the Vue.js framework which simplifies the web interface development without diminishing performance and functionality. The Bootstrap CSS framework is also utilized for the visual design. Additionally, several other JavaScript packages are used, including Leaflet for interactive maps, Dygraphs for high performance plots, and v-network-graph for the single-line diagram visualisation.

### ***Data exchange model and synergy with the rest of IEPT tools***

INTEMA.grid requires a large set of input data including main grid assets parameters and design points, their topology, and the corresponding loads. The tool utilizes mainly static/historical data for its simulations; thus, there is no need for any real-time connection, minimizing the risks for any security issues. Output of the tool, includes the list of calculated KPIs listed in Table 15 and generally regard:

- production of conventional and renewable generation units,
- energy exchange with the grid,

- storage assets integration (batteries, heating batteries, pumped hydro storage, etc.), and
- aggregated values for the performance assessment of the grid.

The calculated indicators will be considered by the CBA tool for the holistic scenario assessment. CERTH generates and stores in its own storage disks the detailed simulations results. Since these files can become large (several GB), the most important variables calculated are stored in a JSON formatted file. The following naming convention is used “INTEMA\_’case\_id’\_run’run\_id’.json”.

An additional data exchange model is under development for the integration of INTEMA.grid and VERIFY. The purpose of this integration is the provision of simulated operation data to the VERIFY platform in the form of hourly-based timeseries. This enables higher granularity of the LCA and LCC calculations performed by VERIFY. The data exchange format is described in detail in subsection 2.2.

### ***Integration: exposed services/ API***

INTEMA.grid requires a large set of input data including main grid assets parameters and design points, their topology, and the corresponding loads. The tool utilizes mainly static/historical data for its simulations; thus, there is no need for any real-time connection, minimizing the risks for any security issues. The output of the tool includes the list of calculated KPIs listed in Table 15 and generally regards:

- production of conventional and renewable generation units,
- energy exchange with the grid,
- storage assets integration (batteries, heating batteries, pumped hydro storage, etc.), and
- aggregated values for the performance assessment of the grid.

The calculated indicators will be considered by the CBA tool for the holistic scenario assessment. CERTH generates and stores in its own storage disks the



detailed simulations results. Since these files can become large (several GB), the most important variables calculated are stored in a JSON formatted file. The following naming convention is used “INTEMA\_’case\_id’\_run’run\_id’.json”.

An additional data exchange model has been development for the integration of INTEMA.grid and VERIFY-D. The purpose of this integration is the provision of simulated operation data to the VERIFY platform in the form of hourly-based timeseries. This enables higher granularity of the LCA and LCC calculations performed by VERIFY-D. The data exchange format is described in detail in subsection 2.2.

## 5.4 User interface

A convenient web-based graphical user interface has been developed to support the case definition and results visualization (Figure 21). The interface is based on the Single Page Application (SPA) [20] approach, which eliminates page reloads and enhances the usability in more complex applications where multiple interactions between the user and the system must take place. The implementation and technology stack that has been utilized is discussed in Section 5.3.

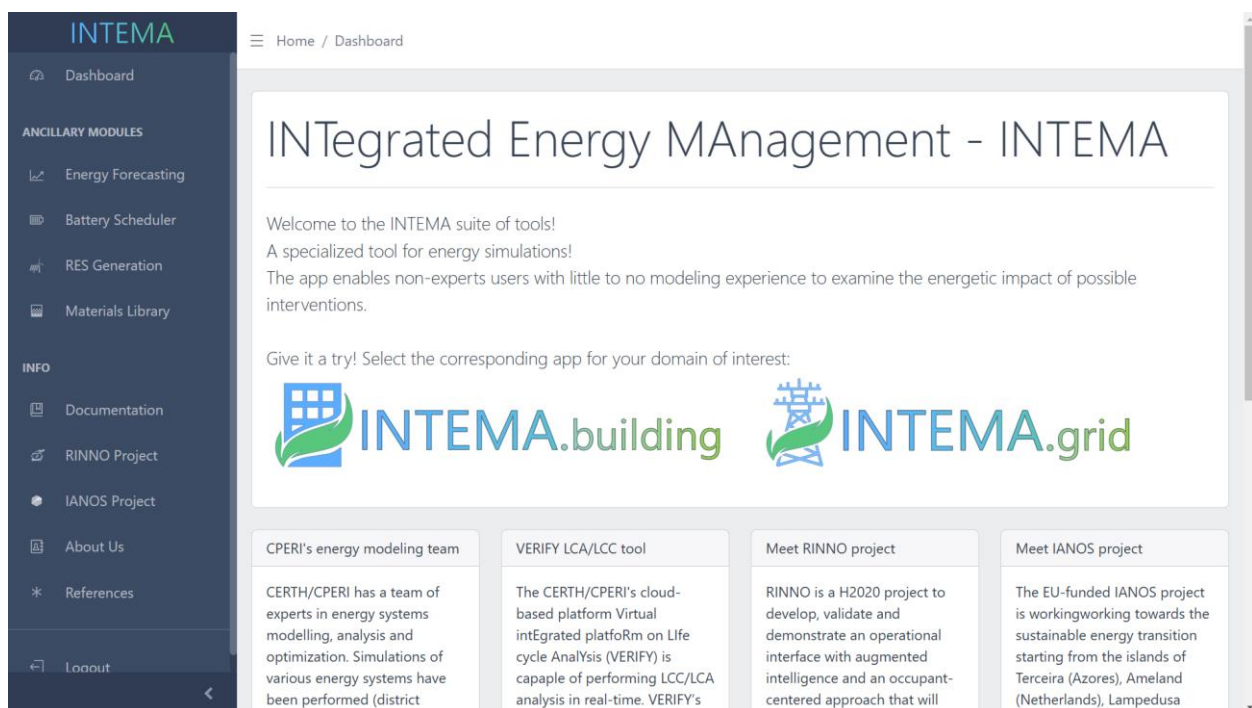


Figure 21: INTEMA platform landing page.

Utilizing the web interface, the user is able to define, for the network topologies of interest, its key components and interconnections among them, and perform steady or transient simulations, with a user defined timestep, according to the objectives of the simulation.

A significant feature of the tool is that **each grid topology should be defined only once and can be then reused in different studies**, i.e., power flow, optimization or dynamic. This feature eliminates the tedious data re-definition that typically occurs during the various power system studies, thus enhancing user experience and usability. To achieve this functionality, the user is navigated to the “Topology Definition” process first, and then the various studies become available. The overall case set-up procedure is schematically depicted in Figure 22. In addition, the network can be defined either manually or by uploading a properly formatted file (.cdf) file, describing it, enhancing further the available functionality.

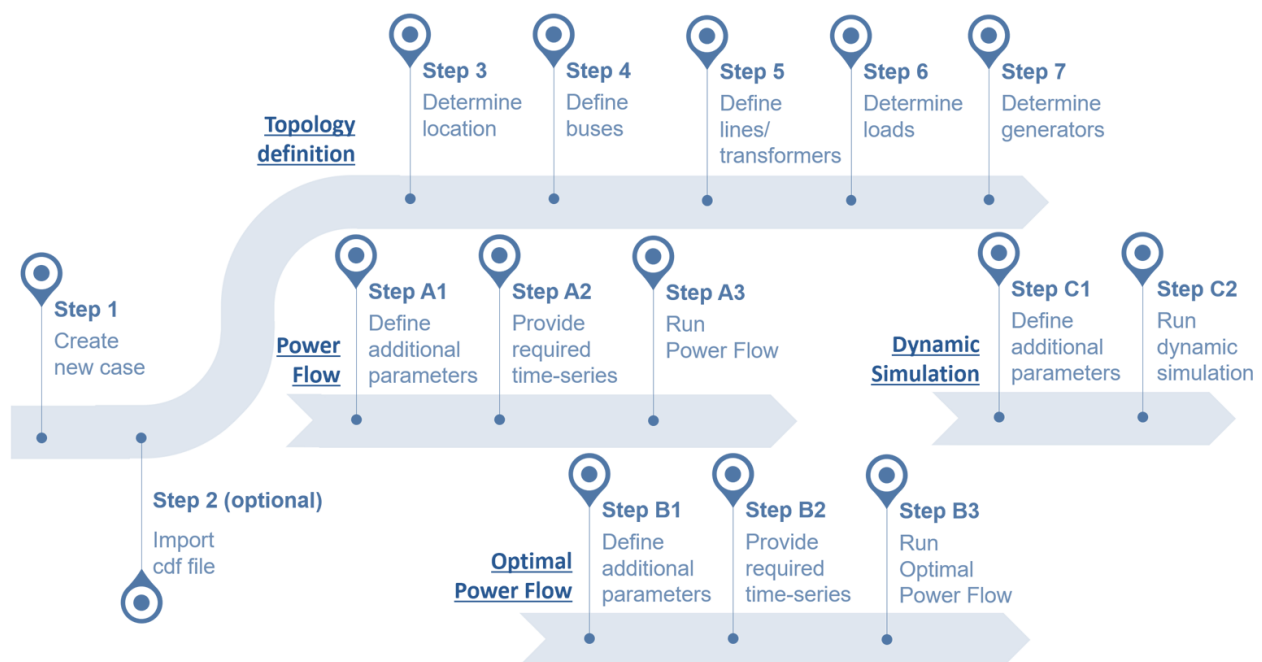


Figure 22: INTEMA.grid case set-up process.

In Figure 23, the system definition view is depicted. Here, all already defined systems are available for revisit while the user can define a new system. The several options regarding topology definition, power flow and optimal power flow

analysis, dynamic simulation, as well as .cdf file upload, can be viewed as buttons in the table.

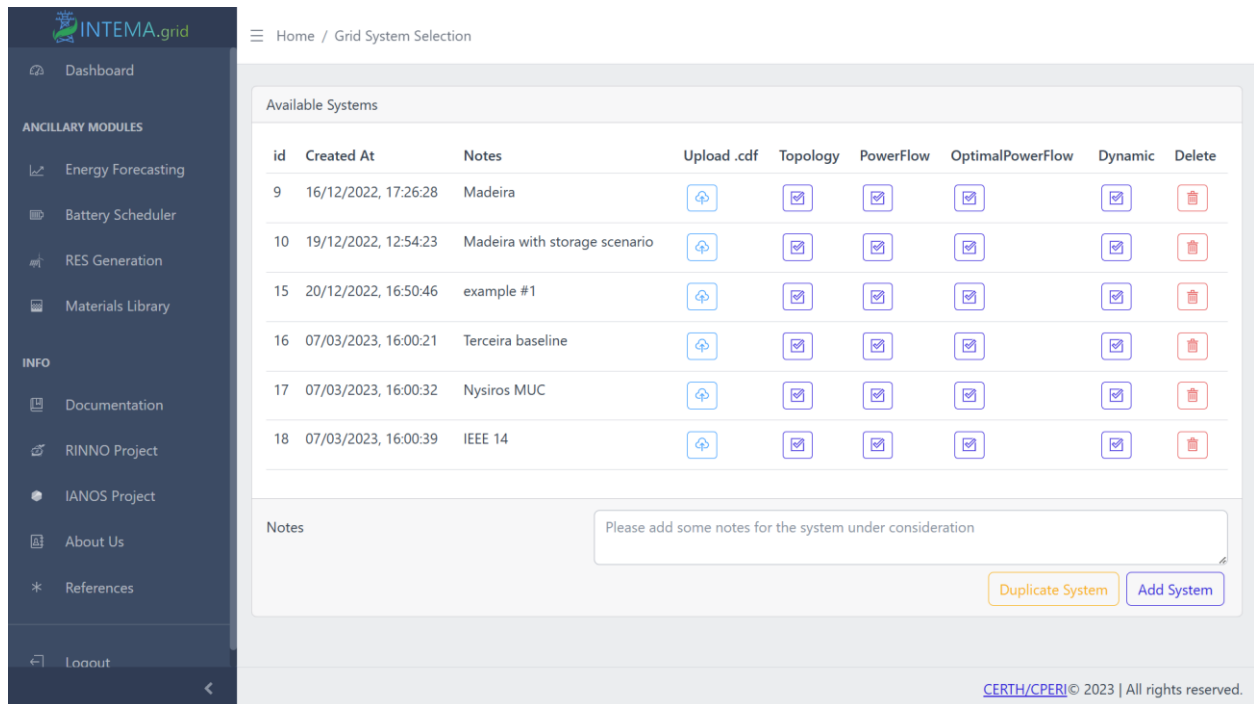


Figure 23: INTEMA.grid GUI - available systems and new system creation view.

As previously mentioned, for the case where no .cdf file is available, the user has to manually define the grid following topology definition procedure. In that case, the user successively defines i) the location of the grid (Figure 24), something that automatically fetches the corresponding weather data (Figure 25) required for the case of RES installations, ii) lines and transformers (Figure 27), iii) loads (Figure 28), iv) generators (Figure 29) and lastly iv) any IANOS innovative technologies (Figure 30) present in the system under consideration. At this stage, only the general information regarding the previously mentioned assets is defined, that are common between studies (power flow, optimal power flow, dynamic), while any additional study-specific information required will be supplemented at later time. The aim is to form the basis upon which the various studies will be built.

An interactive single-line-diagram can be viewed in the figures, which helps the user to easily identify the asset they want to modify or delete. The assets in the graph can be dragged into position, which enables the user to properly represent the grid topology “as on paper”. Furthermore, an automatic arrange of

nodes has been implemented for the case where no specific node positions are inserted.

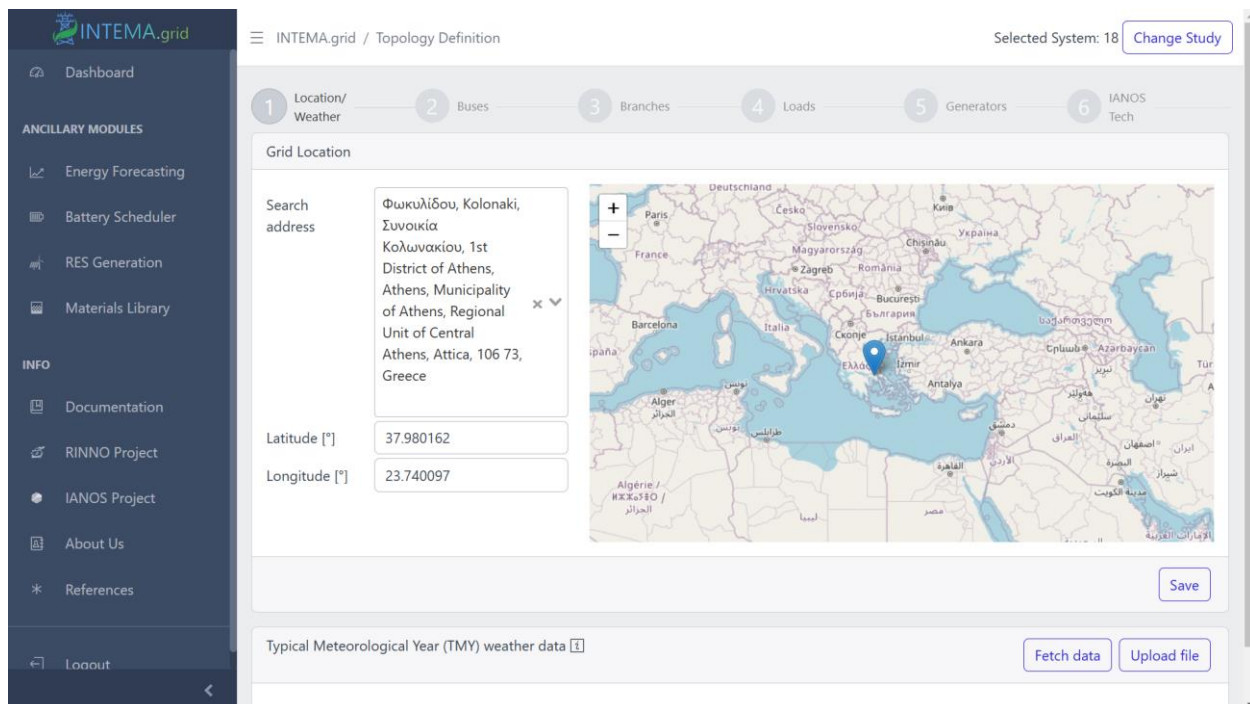


Figure 24: INTEMA.grid GUI – location definition.

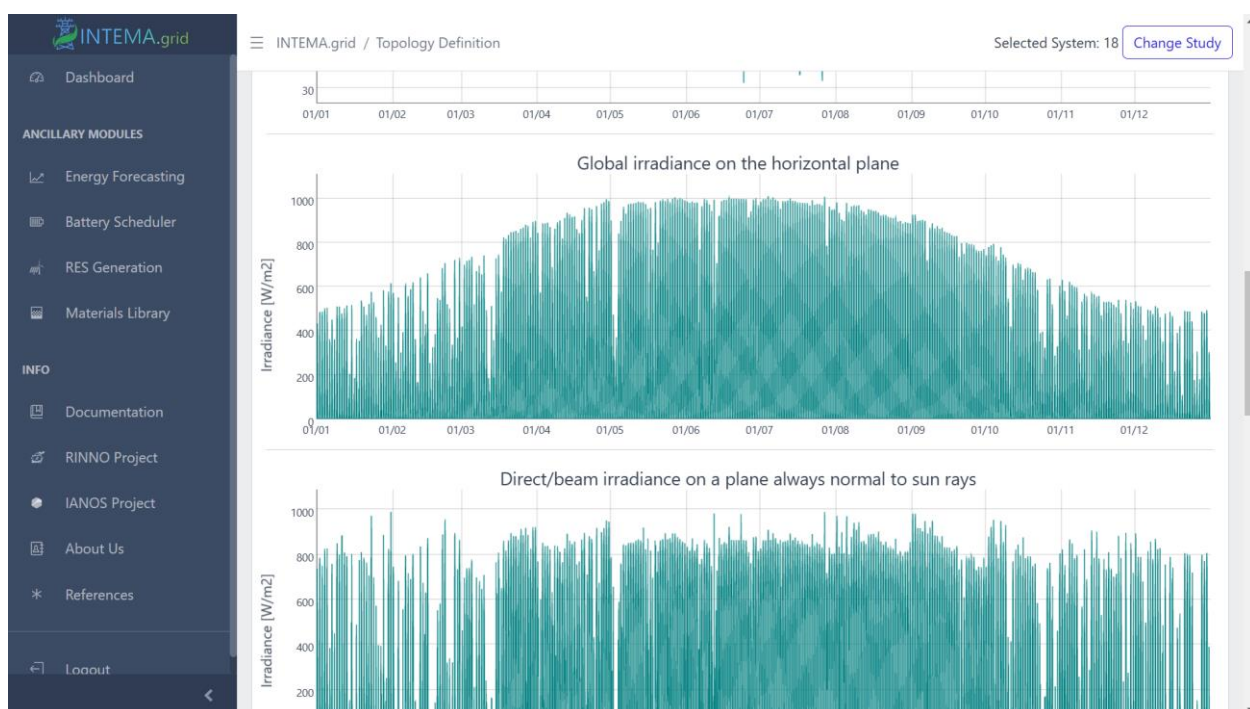



Figure 25: INTEMA.grid GUI – automatic weather data retrieve.



INTEMA.grid / Topology Definition


Selected System: 18 [Change Study](#)

1 Location/Weather 2 Buses 3 Branches 4 Loads 5 Generators 6 IANOS Tech

**Buses Definition**

ID	Name	Nom.Voltage [kV]	Edit	Delete
1	bus_1	69		
2	bus_2	69		
3	bus_3	69		
4	bus_4	69		
5	bus_5	69		
6	bus_6	13.8		
7	bus_7	13.8		
8	bus_8	18		
9	bus_9	13.8		
10	bus_10	13.8		

Figure 26: INTEMA.grid GUI – buses definition.



INTEMA.grid / Topology Definition

Selected System: 18 [Change Study](#)

1 Location/Weather 2 Buses 3 Branches 4 Loads 5 Generators 6 IANOS Tech

**Lines Definition**

ID	Name	Bus from	Bus to	Resistance [pu]	Reactance [pu]	Nom.Power [MVA]
1	line_1	bus_1	bus_2	0.01938	0.05917	120
2	line_2	bus_1	bus_5	0.05403	0.22304	65
3	line_3	bus_2	bus_3	0.04699	0.19797	36
4	line_4	bus_2	bus_4	0.05811	0.17632	65
5	line_5	bus_2	bus_5	0.05695	0.17388	50
6	line_6	bus_3	bus_4	0.06701	0.17103	65
7	line_7	bus_4	bus_5	0.01335	0.04211	45
11	line_11	bus_6	bus_11	0.09498	0.1989	18
12	line_12	bus_6	bus_12	0.12291	0.25581	32
13	line_13	bus_6	bus_13	0.06615	0.13027	32

Figure 27: INTEMA.grid GUI – branches (lines and transformers) definition.





INTEMA.grid / Topology Definition

Selected System: 18 [Change Study](#)

1 Location/Weather 2 Buses 3 Branches 4 Loads 5 Generators 6 IANOS Tech

**Loads Definition**

ID	Name	Bus	Edit	Delete
1	load_1	bus_2		
2	load_2	bus_3		
3	load_3	bus_4		
4	load_4	bus_5		
5	load_5	bus_6		
6	load_6	bus_9		
7	load_7	bus_10		
8	load_8	bus_11		
9	load_9	bus_12		
10	load_10	bus_13		

Figure 28: INTEMA.grid GUI - loads definition.



INTEMA.grid / Topology Definition

Selected System: 18 [Change Study](#)

1 Location/Weather 2 Buses 3 Branches 4 Loads 5 Generators 6 IANOS Tech

**Generation Definition**

Name	Type	Bus	Edit	Delete
steam_1	steam	bus_1		
steam_2	steam	bus_2		
steam_3	steam	bus_3		
steam_4	steam	bus_6		
steam_5	steam	bus_8		

Type:

Name:

Bus:

[Add Generator](#)

Figure 29: INTEMA.grid GUI - generators definition.

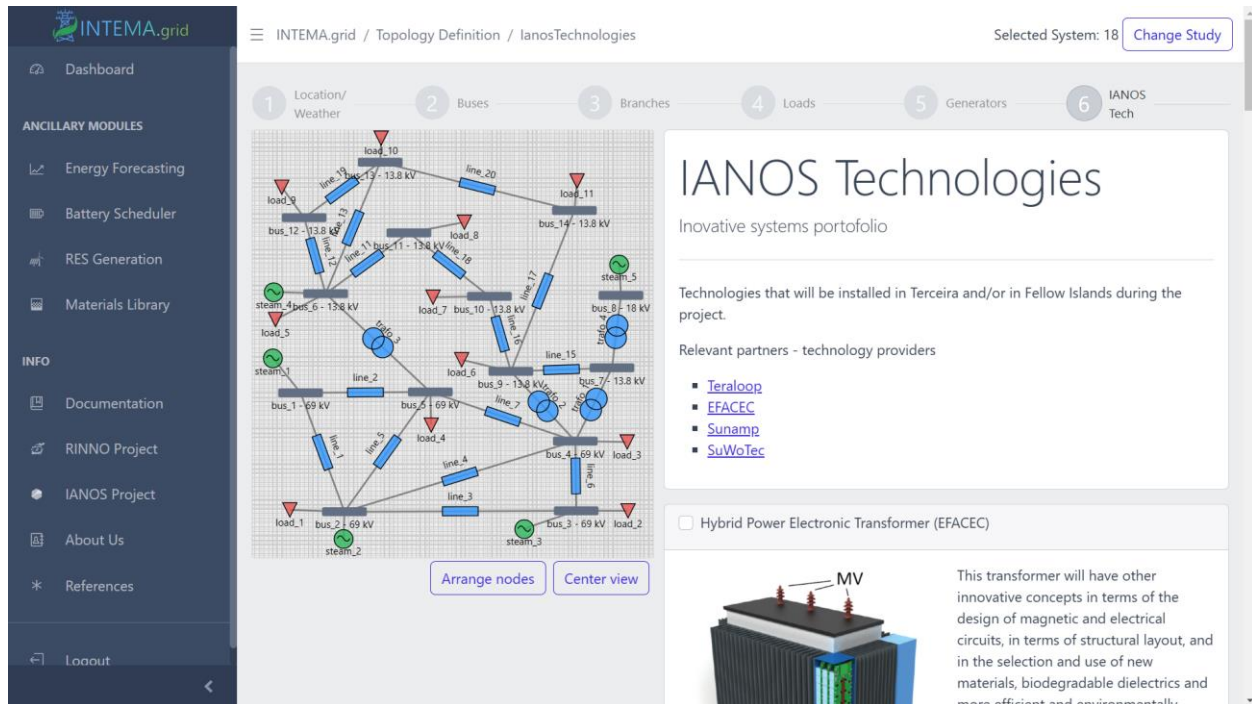


Figure 30: INTEMA.grid GUI – innovative IANOS technologies definition.

The next step depends on the particular study that the user will choose. In any case, there are three main steps involved: i) additional parameters definition (Figure 31), ii) time-series data definition (Figure 32), iii) simulation and results visualization (Figure 33). The following figures present the required steps for a power flow simulation, serving as an example; implementation differences in the data input and functionality are present among the three studies.

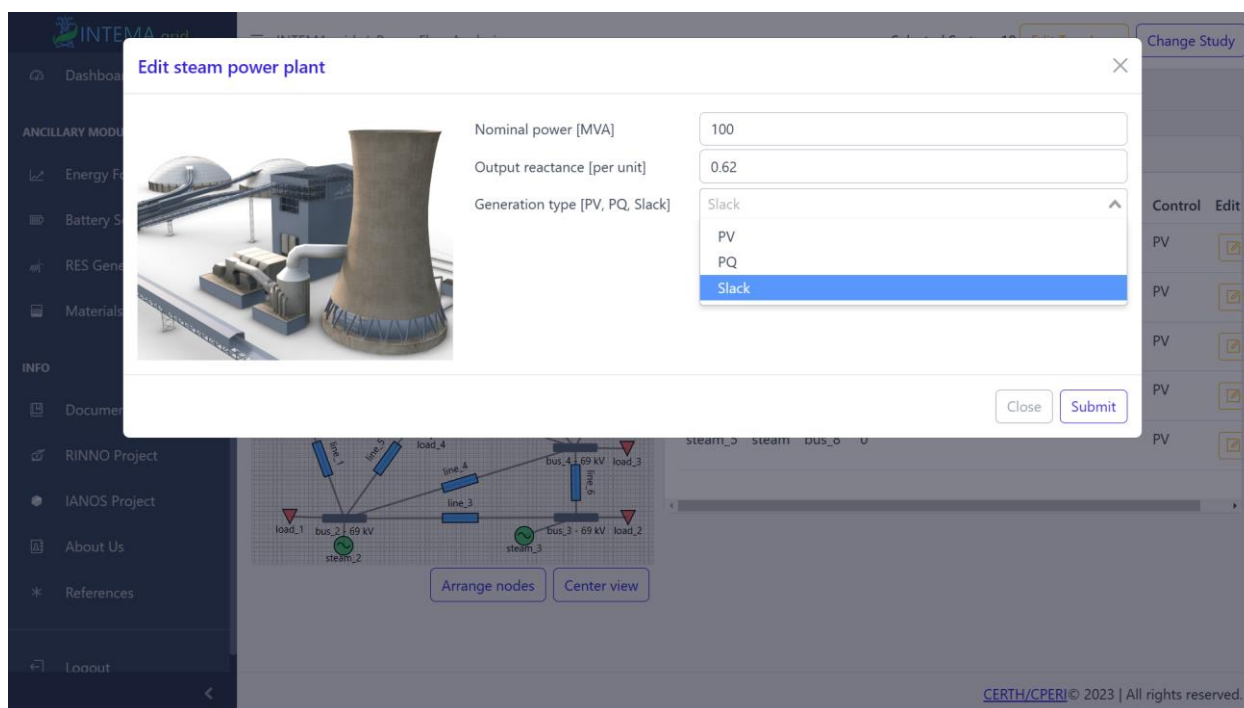


Figure 31: INTEMA.grid GUI – power flow additional parameters for the power plants.

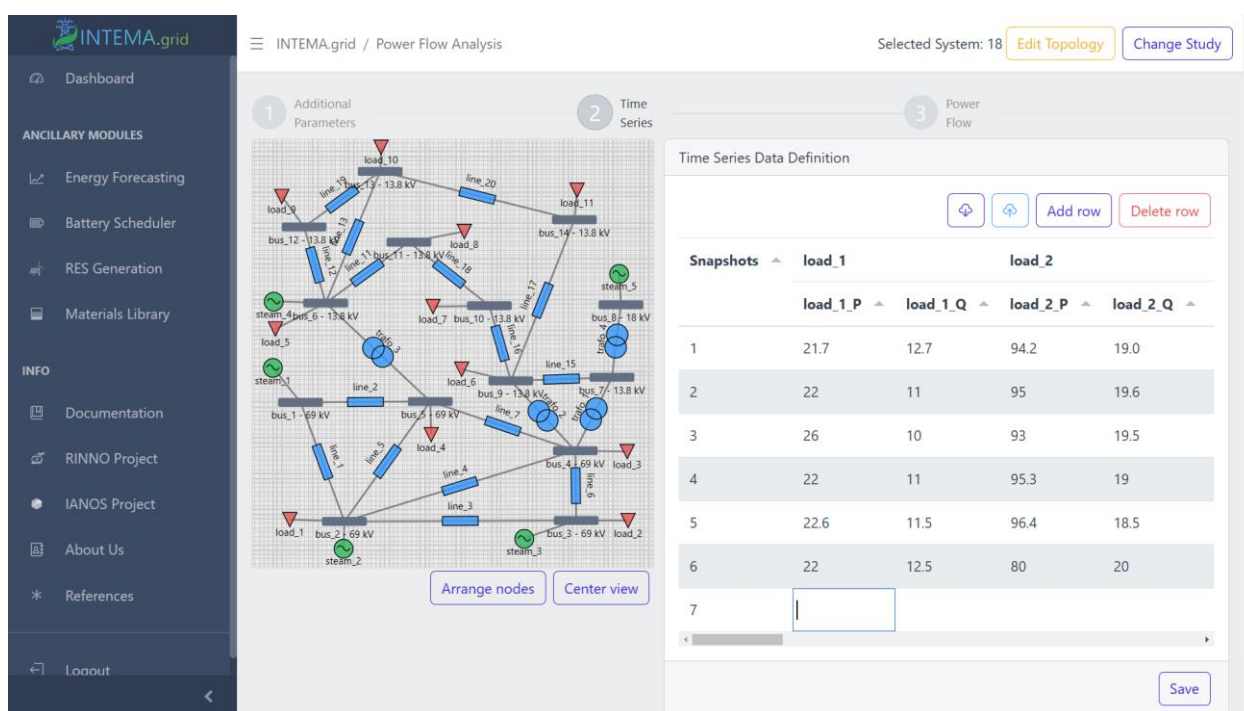


Figure 32: INTEMA.grid GUI – power flow time-series definition view.



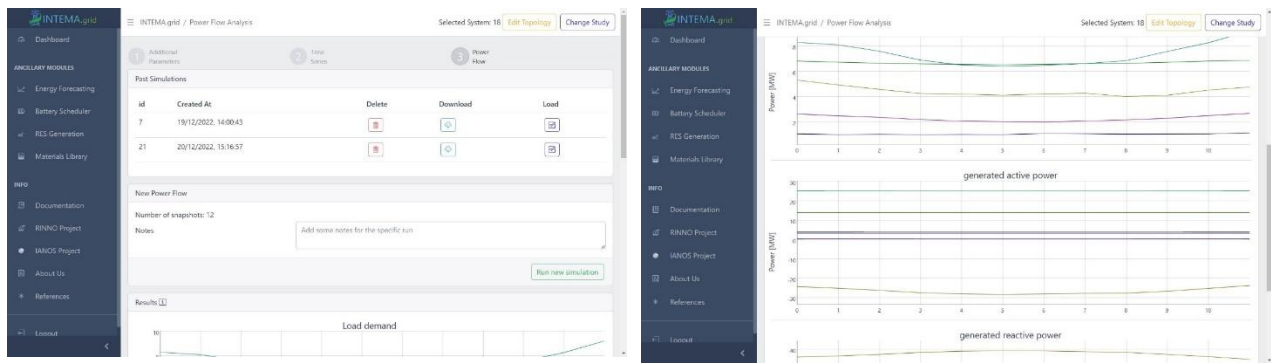


Figure 33: INTEMA.grid GUI – power flow simulation and results view.

A design choice that has been followed during the development is to limit the required input data to the minimum without compromising the quality of the study. Table 18 includes some indicative inputs for the case of power flow. Complementary inputs are required for the other two studies, for instance marginal generation costs for the optimal power flow or response time constants for the dynamic analysis.

Table 18: INTEMA.grid frontend – components and required parameters.

Category	Components	Parameters required
Buses	-	name, nominal voltage
Branches	Lines, Transformers	name, resistance, reactance, nominal apparent power, bus from, bus to
Loads	-	name, active and reactive power consumption, nominal apparent power, bus
Synchronous Plants	Gas, Steam, Diesel, Hydro	name, active and reactive power production, nominal apparent power, voltage magnitude setpoint, control type (PV, PQ or Slack), bus
Variable RES	Photovoltaics, Wind Turbines	name, active and reactive power production, nominal apparent power, bus
Storage	Batteries, Flywheel	name, active and reactive power production, nominal apparent power, voltage magnitude setpoint, bus

In a similar manner, Table 19 presents indicative inputs for the components of IANOS novel technologies for the case of dynamic analysis.

*Table 19: INTEMA.grid frontend – IANOS technologies required parameters.*

Component	Parameters required
Biobased Saline	name, nominal energy capacity, nominal voltage, fully charged voltage, min state-of-charge, maximum discharge power, maximum charge power, discharge efficiency, charge efficiency, life cycles
Hybrid heat pump	name, nominal boiler capacity, nominal boiler efficiency, nominal heat pump capacity, nominal heat pump coefficient of performance (COP), nominal heat pump energy efficiency ratio (EER), heating temperature setpoint, cooling temperature setpoint
Flywheel	name, motor type, number of poles, energy capacity, power capacity, self-discharge time, response time, speed range, moment of inertia, droop, bus
Hybrid Power Electronic Transformer	name, primary and secondary voltage levels, transformer capacity, power electronics capacity, voltage magnitude setpoint, voltage regulation range, bus from, bus to
Solid Oxide Fuel Cell	name, active power production, voltage magnitude setpoint, bus
Proton Exchange Membrane Electrolyzer Cell	name, active and reactive power production, nominal apparent power, voltage magnitude setpoint, bus
Geothermal Power Plant	name, capacity, working medium (e.g., cyclopentane), active and reactive power production, nominal apparent power, voltage magnitude setpoint, bus
MicroCHP	name, nominal mass flow rate, water volume, maximum flow temperature, maximum return temperature, nominal electric power, fuel type, nominal thermal

	power, nominal fuel input (kW), nominal fuel consumption (m <sup>3</sup> /h), outlet pipe diameter
PCM storage	name, capacity, nominal mass flow rate, heat exchange area, PCM volume, material properties (density, solidus and liquidus temperatures, latent heat of fusion, specific heat capacity)
Autogenerative High Pressure Digestion	name, reactor volume, feedstock composition, feedstock volume rate, vapor/liquid volume ratio

## 6. VERIFY-D module

### 6.1 Overview

VERIFY-D is a web-based platform that provides a holistic methodology and evaluation for the LCA and LCC analysis. The platform provides extensive analysis moving beyond the inventory level computations, by assessing the flows in terms of environmental and costing impacts of the city infrastructure elements.

The environmental and costing analysis covers multiple time periods (25 to 45 years ahead). The VERIFY-D methodology can be further applied at a district/island level to model and perform multi-domain LCA analysis, considering the impacts of 1) private and public buildings, 2) transportation infrastructure elements, 3) produced energy of RES and non-RES technologies, 4) energy storage systems.

### 6.2 Functionalities

The main objective of the platform is to assist in achieving the desired energy transition targets, as they are proposed by the EC [23]. For obtaining more accurate results almost all stages of the value chain are considered (e.g., production, use phase). VERIFY-D as a software tool combines the static LCA-LCC analysis with the dynamic use phase of system components set during the specified lifetime. Input data (either real-time, near real-time or synthetic) from multiple external sources or tools (specifically for synthetic type of data), are supported through a custom API implementation. A large variety of innovative technologies is included and demonstrated as the main scope of IANOS. The operation and environmental assessment of which, in terms of LCA and LCC terms, will be analysed using VERIFY-D. In addition, data repositories assist the platform functionality, by storing large amount of data in specific ontology frames (e.g., SAREF [24]).

## 6.3 Implementation and technology stack

In contrast with other LCA-LCC applications, VERIFY-D is based on open-source libraries, frameworks and databases (e.g., Python, Ruby on Rails, PostgreSQL etc.) eliminating the dependencies with closed code tools. Extended description for the VERIFY-D platform architecture, functionalities and capabilities is available in IANOS Deliverable D3.2.

## 6.4 User interface

VERIFY-D has an intuitive web interface based on the Ruby on Rails framework. The user is navigated through the various forms for the data input and can explore and analyse the results via interactive plots. The reader is referred to IANOS Deliverable D3.2 for a detailed presentation of these functionalities.



## 7. Conclusions

This report presents the overview, architecture and functionalities of the IEPT suite. The IEPT suite is a holistic tool for the decision-makers in the energy value chain to assess quantitatively from an economic, technical and environmental perspective whether the smart grid interventions provide benefits that exceed the existing BaU operational scenarios. This will diminish the barriers for RES acceptance of the stakeholders, unleashing the decarbonization potential for the islands in the European region.

The backbone of this suite is the CBA component, which receives the input from the LCA/LCC and the energy and power system modellers, offering quantifiable investment insights. On top of that, the IEPT suite contains a platform that offers crowdfunding services to the investors, thus providing an end-to-end experience for the stakeholders that are willing to upgrade their existing energy assets and power system infrastructure, towards a sustainable future. The integral components of the IEPT suite are:

- CBA Component
- VERIFY-D LCA/LCC Platform
- INTEMA.grid System Modeller
- ESSIM Modeller
- CrowdEquity Platform

The IEPT suite will be employed by the LH in order to evaluate the demonstration activities, through the calculation of a set of KPIs, as well as for the CBA of the smart grid interventions in both LH and FI. Finally, the IEPT suite will be used for the scalability and replicability studies of the IANOS solutions in the three FI.

## 8. Appendix A – Description of the INTEMA.grid dynamic component models

The dynamic module of INTEMA.grid is based on the Modelica dynamic modelling language and developed in Dymola environment. This entails the requirement of high-order transient models development that are capable of revealing the dynamic response of the systems under consideration. The present Appendix section describes the dynamic models that are utilized in the transient simulations, while also presents the development of the in-house models for specific technologies that were needed in the IANOS project context.

### 8.1 Passive grid components

#### Transmission/distribution lines

The “ $\pi$ -equivalent” or “T-equivalent” circuit [25] is used to represent the distribution and transmission line with the discretized telegrapher equation. The models are available in PS.

#### Transformers

The PS library contains models for transformers in different abstraction levels, with the choice of Y-topology or  $\Delta$ -topology at primary and secondary windings. The impedance values are determined ‘as seen from the terminals’, so that the same parameter values lead to equivalent networks, independently of the choice of the primary and secondary topology. A phase-shift parameter may be used in order to account for the phase difference between the primary and secondary sides, due to the applied winding and polarity scheme. The models provided by the PS library include:

- Ideal magnetic coupling, i.e., zero magnetization current, no stray-impedance.
- Ideal magnetic coupling, but with stray-impedance
- Ideal magnetic coupling, with stray-impedance, but with two secondary windings.



- Stray-impedance and resistance, with non-ideal magnetic coupling, i.e., non-zero magnetisation current and magnetic losses.
- Stray-impedance and resistance, with non-ideal magnetic coupling, but including effective “time-averaged” saturation.
- Stray-impedance and resistance, with non-ideal magnetic coupling, but using a more realistic and computation-intensive model for the magnetic saturation.

All aforementioned models are valuable for the grid representations. Their choice depends on several factors, such as level of detail, grid scale (distribution or transmission), transient phenomenon under consideration etc.

### Loads

PS library offers a set of load models, each one corresponding to well documented models in standard literature [26],[27] providing different levels of control on the consumed active and reactive power. In all cases, the user may control the load response by means of an input signal, which may be constant or vary dynamically. Depending on the model, the input signal has the meaning of either the “consumed power” or the “consumed power at nominal voltage”. The response of the loads to varying input signals falls into two categories: either “instantaneous”, where the loads instantaneously change their internal state to meet the specified power requirements, or “asymptotic”, where the loads meet the specified power requirements asymptotically at steady state. Loads with asymptotic response accept as parameter a characteristic response time constant. The following list gathers the most useful load models:

- The dynamic impedance load models a load with impedance and instantaneous response that consumes the specified active and reactive power at nominal voltage. It may induce transient effects.
- The steady-state PQ load adjusts instantaneously its internal impedance so that it consumes the specified active and reactive power, independently of the input voltage. The specified power can vary dynamically. Since it requires the solution of a nonlinear problem at each simulation step, it should only be used for small to medium-scale simulations.



- The PQ inductive or capacitive load dynamically change their internal state to asymptotically meet the power requirements specified by the input signal at all voltages.
- The ZIP load model is an inductive nonlinear load with a characteristic that is a second-degree polynomial of voltage and current. The effective load asymptotically relaxes to the input signal at steady state and nominal voltage.

## 8.2 Conventional power plants

### Steam power plant

Steam turbines of fossil-fuel power plants make up the most of generation capacities of the industrially developed countries [28]. Such plants use coal, lignite, peat, natural gas or diesel as their fuel and produce superheated (approximately 600 °C) and high pressurized steam (approximately 28 MPa) supplying the steam turbine. High-capacity power plants often use multiple stages of steam turbines, i.e., high, intermediate and low-pressure turbines.

A steam power plant model is implemented by CERTH using PS and Modelica standard libraries and is presented in *Figure 34*. The model receives the active and reactive power setpoints and injects the power to the grid accordingly. In *Figure 34*, the main components can be identified, namely the turbine-governor subsystem, the automatic voltage regulator and exciter subsystem, the power system stabilized and the synchronous generator. In general, standardised models for the synchronous generation components have been favoured to ensure the validity of the responses.

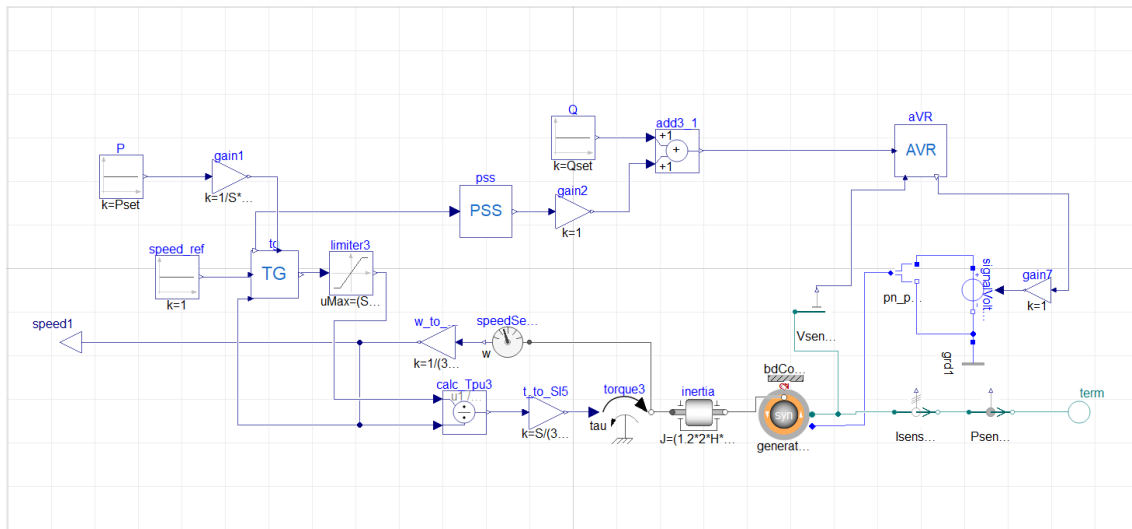


Figure 34: CERTH developed steam power plant model.

The required torque is set and controlled by the turbine's governor. The model of the standardized steam turbine-governor system IEEE-TGOV1 and its parameters are depicted in Figure 35. The model receives as an input the angular velocity of the turbine and outputs the torque that is needed to remain synchronised. Firstly, the speed control ( $1/R$ ) regulates the torque output as  $\omega$  (grid frequency) oscillates slightly due to load changes. Therefore, the steam flow varies depending on the position of the steam (control valves) introducing time constants/delays to the model. Lastly, a damping coefficient (DT) must be considered as part of the mechanical power dissipated, turning into thermal energy, mainly due to friction. The important parameters that define system's response include the permanent droop  $R$ , the time constants  $T_1$ ,  $T_2$ , and  $T_3$ , as well as the turbine's damping coefficient  $DT$ .

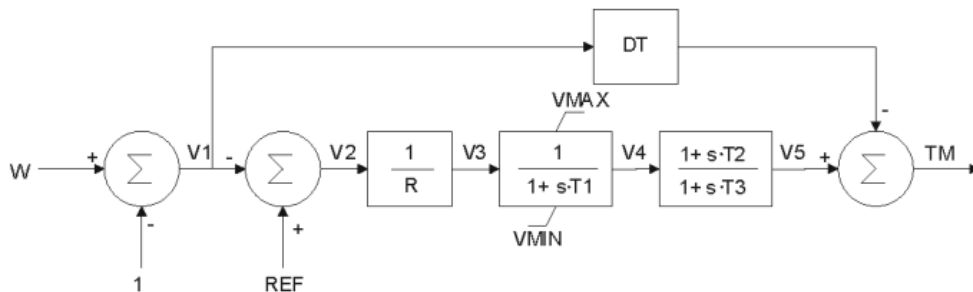


Figure 35: Steam turbine governor IEEE-TGOV [30].

In the generator's side, the terminal voltage is regulated by the automatic voltage regulator, as shown in Figure 36. The model receives as input the generator

terminal voltage and outputs the required field voltage. The basic components of an exciter control system comprise three main components, namely the amplifier, the sensor and the exciter. Parameters  $K_A$ ,  $K_E$ ,  $K_F$  and  $T_A$ ,  $T_E$ ,  $T_F$  are gains and time constants of the amplifier, the exciter and the sensor, respectively.

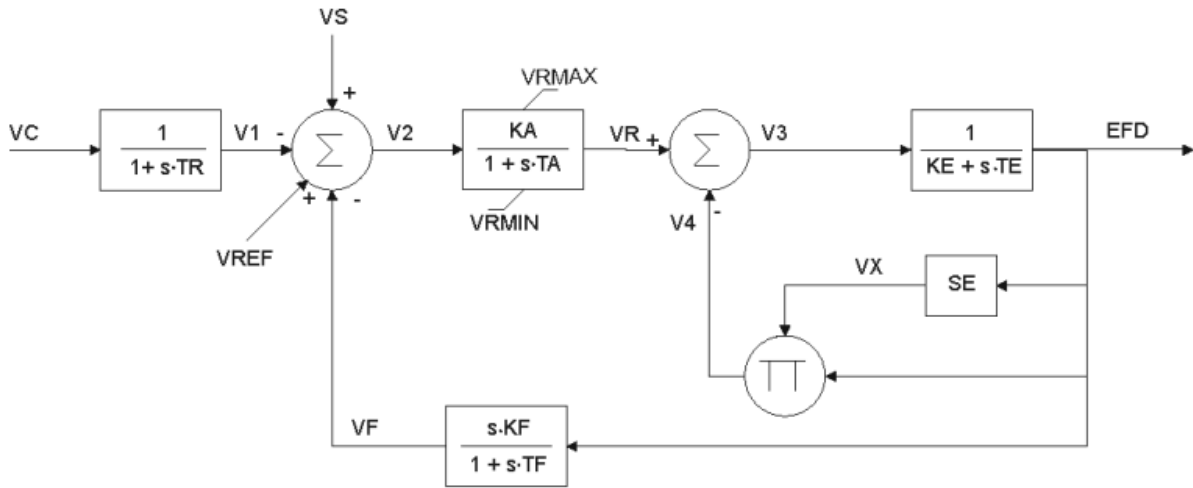


Figure 36: Automatic voltage regulator (AVR) IEEE TI [31].

Power system stabilizers are responsible for the damping of electromechanical oscillations caused by small disturbances due to load changes. The utilized stabilizer model is depicted in Figure 37. The model consists of the KPSS gain, the wash-out filter (TW time constant), provided to eliminate steady-state bias in the output and a phase compensation block/lead-lag filter (T1 and T2 time constants). The output is limited by VPSSMAX and VPSSMIN voltage limits.

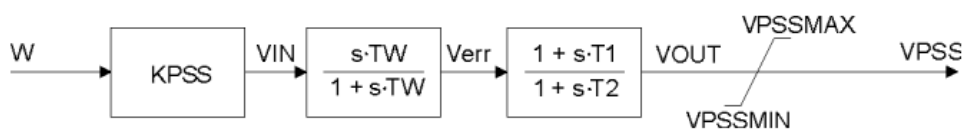


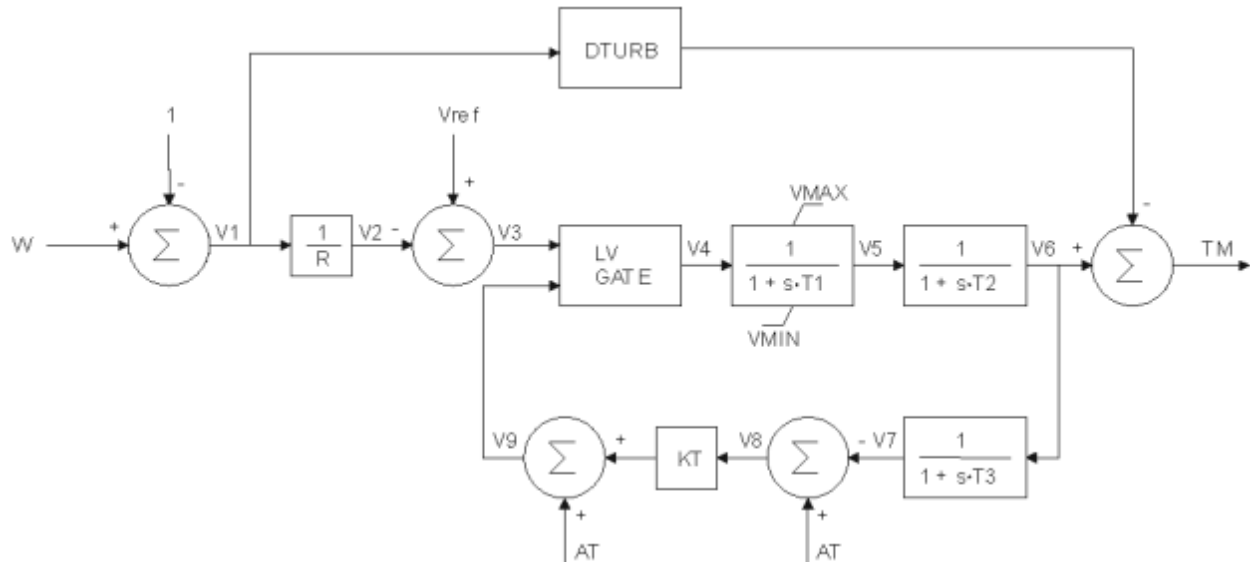
Figure 37: Simple PSS model [32].

Important parameter of this model is TW, known as washout time constant.

### Gas power plant

A gas turbine power plant produces a great amount of energy for its size and weight. It comprises of three primary sections, i.e.: the compressor, the

combustion chamber and the turbine. The new gas turbines have inlet temperatures close to 1500 °C, pressure ratios of approximately 40:1 and efficiencies that surpass 45% [33]. A gas turbine model with type GAST governor is depicted in *Figure 38*.



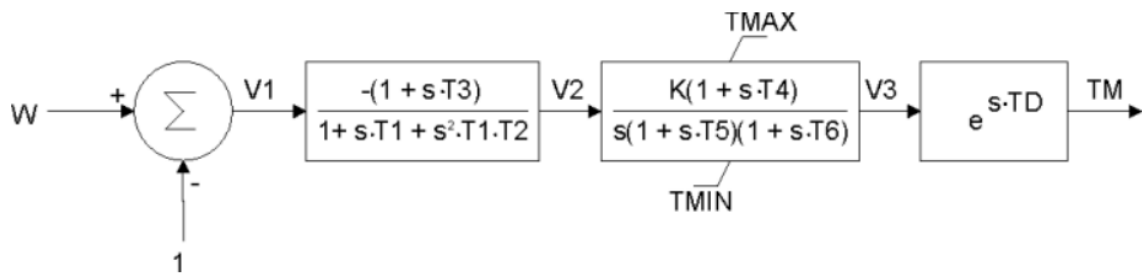
*Figure 38: Gas turbine model type GAST [32].*

As in all turbine-governors, the permanent droop control ( $1/R$ ) automatically adjusts the produced power ( $TM$ ) on synchronous units as the grid frequency changes. The lowest value gate ( $LV$ ) determines how much fuel is to be admitted to the turbine and introduces the time constant  $T1$  representing the natural valve positioning time constant of the governor for small disturbances.  $T2$  represents the delay due to internal energy storage of the gas turbine engine. Furthermore, there are limits to the rise of exhaust temperatures caused by increasing loads. Thus, the loop with the ambient temperature load limits ( $AT$ ), the turbine exhaust temperature time constant  $T3$  and the temperature limiter gain  $KT$  is included in the model. The  $DTRUB$  factor must be introduced as well, as the energy extracted from the high temperature, high pressure gas produced by the combustor on the turbine blades is limited.

### Diesel power plant

Power plants with reciprocating generators run on cheap heavy fuel oil and have proven to be a reliable and economical means of power generation worldwide. A

Modelica diesel power plant model is integrated by CERTH with the synchronous generator of PS library. The modelling of the diesel power plant follows the one shown in *Figure 34*, but with a different turbine governor, presented in *Figure 39*.



*Figure 39: Diesel turbine governor DEGOV [30].*

The DEGOV model is employed to simulate torque produced by the diesel engine based on the speed signal from the governor. This model consists of an electric control box with time constants  $T_1$ ,  $T_2$ ,  $T_3$ , an actuator with  $T_4$ ,  $T_5$ ,  $T_6$ , and a diesel engine with time delay ( $TD$ ).

## 8.3 Renewable power plants

### Photovoltaics

Constituting a popular generation technology, a set of well-validated photovoltaic models have already been developed. The most suitable model for grid level simulations is included in BuildingSystems library. The model applies the Lagrange multipliers method for the calculation of the Maximum Power Point (MPP) at each time. This approach avoids the need for a MPP tracking algorithm (e.g., perturbation and observation, incremental conductance) which introduce discrete (clocked) decision making and burden simulation time.

### Wind turbines

Wind turbines use generators to transform kinetic energy into electricity. INTEMA.grid library includes two different types of wind turbine models. One is modeled with a doubly fed induction generator (slip ring), as shown in *Figure 40*, while the other model presented in *Figure 41* is of a direct-driven generator with permanent magnet excitation [34].

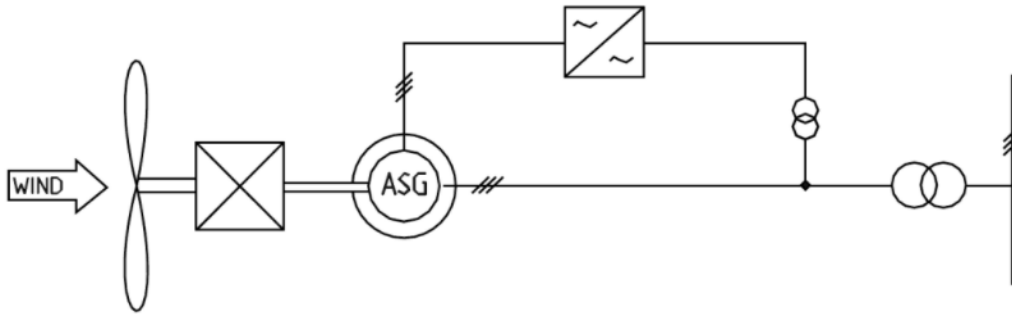


Figure 40: Doubly-fed induction generator with cycloconverter of the experimental Growian wind turbine.

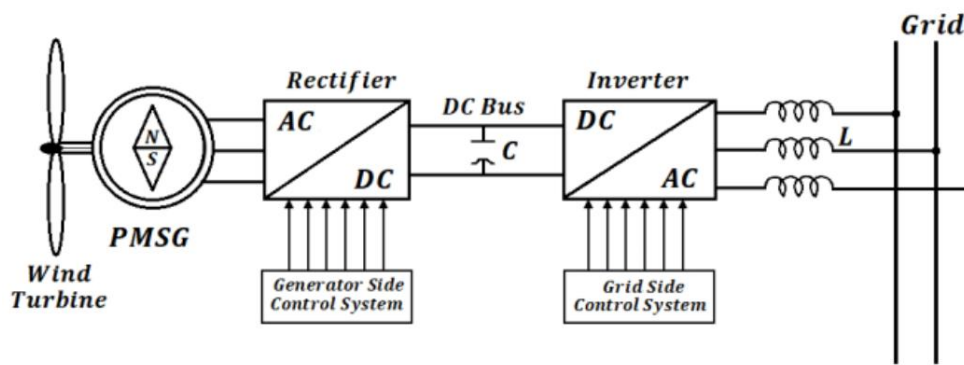


Figure 41: Direct drive permanent magnet synchronous generator wind turbine energy conversion principle scheme.

The models consider for the ambient air velocity profile and calculates the injected power to the grid, according to the following arithmetical expression:

$$P = \frac{1}{2} \times \rho \times A \times v^3 \times C_p$$

where  $\rho$  is the density of air,  $A$  the area swept by the rotor,  $C_p$  is the power coefficient and  $v$  is the windspeed at hub height. The power coefficient is expressed by the equation:

$$C_p = c_1 \times (c_2 \times \lambda_{dai} - c_3 \times \beta - c_4) \times e^{-c_5 \times \lambda_{dai}} + c_6 \times \lambda$$

$$\lambda_{dai} = \frac{1}{\lambda + 0,008 \times \beta} - \frac{0,035}{\beta^3 + 1}$$

$$\lambda = \frac{\omega \times R}{v}$$

, where  $\beta$  is the pitch angle,  $\lambda$  is the tip speed ratio,  $\omega$  the angular velocity,  $R$  the radius,  $v$  the velocity of the air, and  $\lambda_{dai}$  an auxiliary variable.

## Hydroelectric power plants

Hydropower is considered to be the best option for electric power generation, because of its economic and environmental merits. Hydroelectric stations can either be pumped-storage plants, an impoundment facility with a reservoir in upland or mountainous region, or a run-of-river plant [35]. The general model of a hydroelectric turbine-governor system is presented in *Figure 42*.

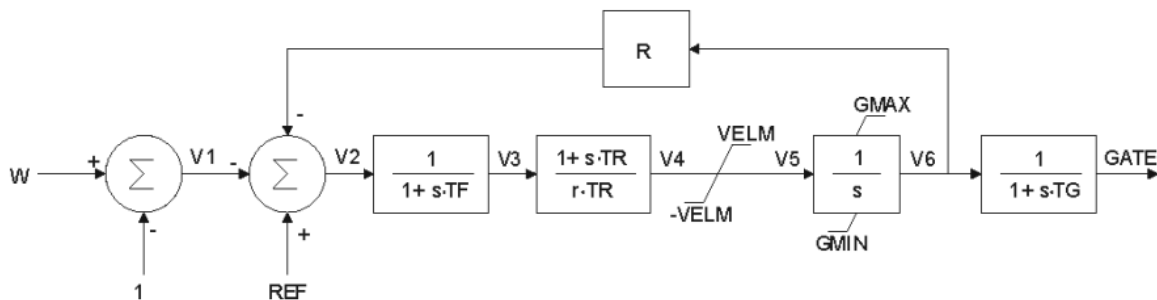


Figure 42: HYGOV hydro turbine-governor model [30]

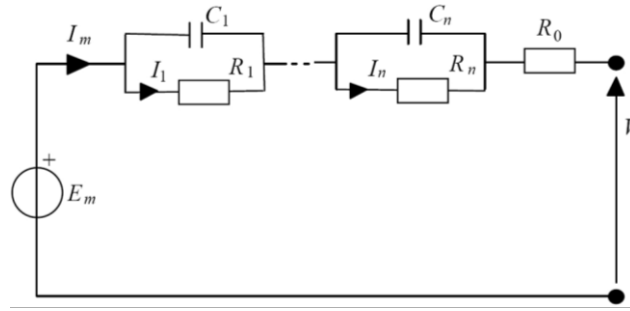
The input and output of the model are the reference rotor speed ( $w$ ) and the gate opening (GATE) in p.u. Parameters  $R$  and  $r$  represent the permanent and temporary droop, respectively. The model includes a deadband and a lead/lag filter (TF) on the primary speed input. The velocity limit (VELM) is the reciprocal of the time taken for the wicket gates to move from fully open to fully closed. The parameter TG is the representation of the time lags within the governor hydraulic servo system. GMAX is the maximum gate limit setting as established by the operator at the governor console. The minimum gate position (GMIN) is normally zero.

## 8.4 Energy storage

### Battery systems (Lithium-ion, Lead-acid)

Dynamic models for two types of battery cells have been developed to support the representation of small, medium and large-scale Battery Energy Storage Systems (BESSs). The two types of batteries are lithium-ion (Li-ion) and lead-acid, which are the most widely used in energy applications. Both have been modelled following the widely approved modelling technique of the Equivalent Circuit

Method (ECM), presented in the literature for both Li-ion and Lead-acid batteries [36], [37], [38]. The generalised topology of ECM is depicted in *Figure 43*.



*Figure 43: The generalized Equivalent Circuit Model (ECM) [26].*

The model focuses on calculating accurately the terminal voltage and state-of-charge (SoC) of each cell of the BESS configuration, based on the instantaneous power extracted to (discharging) or received by (charging) the connected electrical grid. Battery voltage response evolves with a nonlinear behaviour when electric charge is transferred through battery terminals, and this is due to complex electrochemical phenomena (polarisation – diffusion, double layer). These transient phenomena, related to the nonlinearity of voltage response, are captured by a number  $n$  of parallel resistor-capacitor (RC) branches. This number is decided by the specific battery application, required accuracy and available computational resources. A resistance connected in series ( $R_0$ ) is equivalent to instantaneous voltage drop, while open-circuit voltage (OCV or  $E_m$ ) reflects the available internal energy. All of the above model parameters are dependent on SoC, temperature, current and therefore their values are extracted for varying external conditions from look-up Tables made available from CERTH datasets for many types of battery chemistries. The set of equations solved by the model:

$$\text{SoC}(t) = \text{SoC}(t_0) - \frac{1}{C_{\text{bat}}} \int I_m(t) dt$$

$$I_{C,i} = C_i \frac{dV_i}{dt}$$

$$I_m = I_{C,i} + I_i = C_i \frac{dV_i}{dt} + \frac{V_i}{R_i}$$



$$V_t = E_m - \sum_{i=1}^n V_i - I_m R_0$$

, where  $C_{bat}$  denotes battery cell capacity and  $V_t$  is the cell terminal voltage. The battery cell model developed in Dymola with 1 RC branch and thermal dependence is presented in *Figure 44*. The model receives the charge/discharge power setpoint based on specific operation strategy. Then, based on the battery's power balance, the SoC is estimated, and the terminal voltage is given as an output.

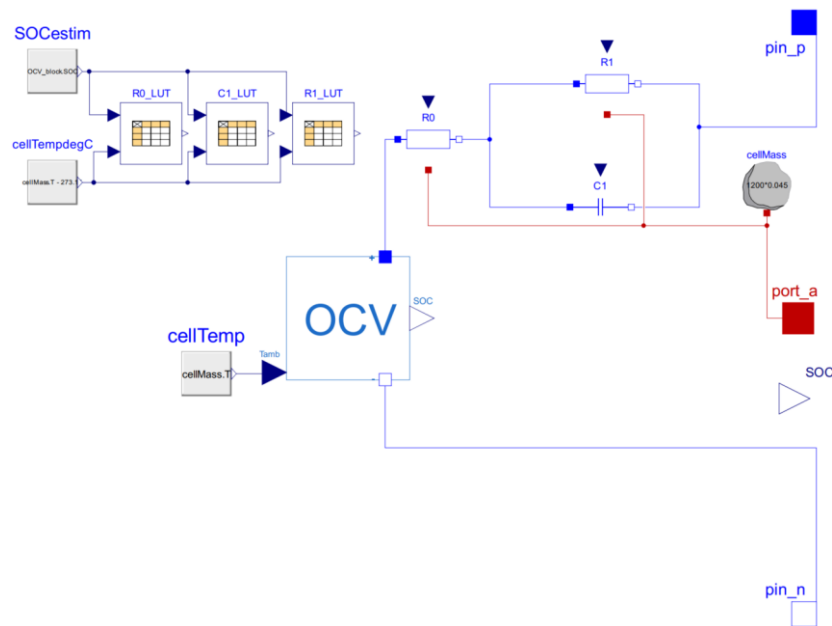


Figure 44: CERTH battery cell model with 1 RC branch and thermal dependence

## 8.5 IANOS technologies

The technologies presented up until now can be considered common in the context of grid modelling, covering the majority of simulation cases. The present section presents the development of specialized dynamic models for the novel IANOS technologies that are installed in the lighthouse islands, i.e., Terceira and Ameland. In this manner, special effort has been made to properly represent these assets in the INTEMA.grid tool. Focus will be placed on the models developed by CERTH, while adopted open-source models from will be described in brief.

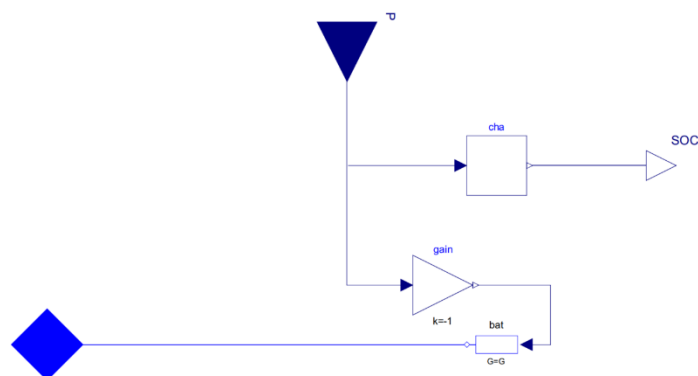
## Biobased Saline

SuWoTec energy storage system consists of a sustainable recyclable saltwater gel battery (chemical compound without any lithium or cobalt). Among other novel features, it supports deep discharge without any effect on aging mechanisms, both AC and DC charging/discharging options, as well as guaranteed safety. Technical specifications are available in Table 20.

*Table 20: Suwotec battery specifications as provided by the manufacturer*

Parameter	Value
Nominal voltage	552 V
Nominal capacity	120 Ah
Maximum charging power capacity	12 kW
Maximum discharging power capacity	15 kW
Charge efficiency	>97% at 20 °C
Discharge efficiency	>96% at 20 °C
Weight	3600 kg

The model used in Dymola software, is based on the battery model of Buildings open-source library, is depicted in Figure 45. As it can be also noticed graphically, the model receives the battery power exchange (P) as an input for each time step and integrates an electrical conductor with the equivalent electrical behaviour in the broader system model, through the positive terminal (at the left of the figure). SoC is also calculated and provided as an output.



*Figure 45: Saline battery model*

## Hybrid heat pump

This type of heat pump (HP) is supported by a complementary boiler combining the advantages of both technologies to provide the necessary heat to the heating system supply water in the most profitable way. For example, at extremely low ambient temperature the coefficient of performance (COP) of the HP is of really low values and therefore boiler contributes exclusively for load coverage. The model developed for the Hybrid Heat Pump (HHP) is presented in *Figure 48*. As can be seen, the model receives as input values the temperature setpoint of the heating system and weather conditions. The opportunity to switch HP mode from heating to cooling is also given. Through its terminal ports, integration within other water and air circuits is made possible. On top of the above, a control system adjusts the heat pump and boiler heat contribution depending on the ambient temperature value.

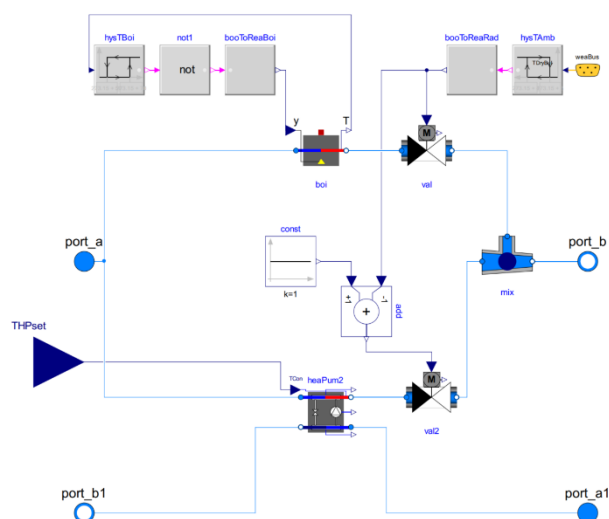


Figure 46: CERTH model of hybrid heat pump

The 135 HHPs installed in Ameland involve 7 different manufacturer models, which are listed in *Table 21*.

Table 21: Number of installed HHP units in 135 Ameland apartments per manufacturer model

HHP model	Number of installed units
Remeha Elga Ace	82
Remeha Tzerra HP 40-5c	37

DAIKIN EHYHBH05AV32	8
DAIKIN Altherma	6
Remeha Tzerra HP 390-5c	1
Mitsubishi Ecodan	1
Vaillant	1

Parameters of the developed model are modified for each application to meet the design specifications of each model. However, since most of the installed units are Elga Ace HHPs, a verification process has been setup for the particular machine. Available technical data from the manufacturer are given in *Table 22*.

*Table 22: Technical specifications of Elga Ace HHP*

Parameter	Value
HP heating capacity	4.14 kW <sub>th</sub>
HP cooling capacity	2.78 kW <sub>c</sub>
Nominal COP	4.5
Nominal EER	4.28
Boiler capacity	24 kW <sub>th</sub>
Boiler fuel type	Natural gas
External unit water operating temperature range (heating)	18 °C/ 55 °C
External unit water operating temperature range (cooling)	17 °C/ 25 °C
External unit air operating temperature range (heating)	-15 °C/ 20 °C
External unit air operating temperature range (cooling)	10 °C/ 45 °C
Nominal water volume flow rate	0.71 m <sup>3</sup> /h
Nominal air volume flow rate	1800 m <sup>3</sup> /h

To move forward with HHP verification the Elga Ace HP performance in heating mode has been compared against values given by the manufacturer datasheet under varying operating conditions [39]. These conditions are listed in *Table 23*.

*Table 23: Elga Ace HHP nominal COP and heating power values for varying outdoor air temperature*

Outdoor air temperature (°C)	Nominal COP	Nominal Power (kW <sub>th</sub> )
-7	2.77	2.8
2	4.04	2.7
7	6.48	1.7
12	9.09	2

Through this procedure the HP model results have been verified since the output thermal power and COP values have been compared against specified data at multiple points throughout the entire allowed operating range.

Figure 47 shows the obtained simulation results of COP value against all possible ambient temperatures. To evaluate the model accuracy, the sum squared error between simulated and measured values expressed as a percentage of nominal COP (4.50) has been calculated and is equal to 1.96 %. Detailed results are listed in Table 24.

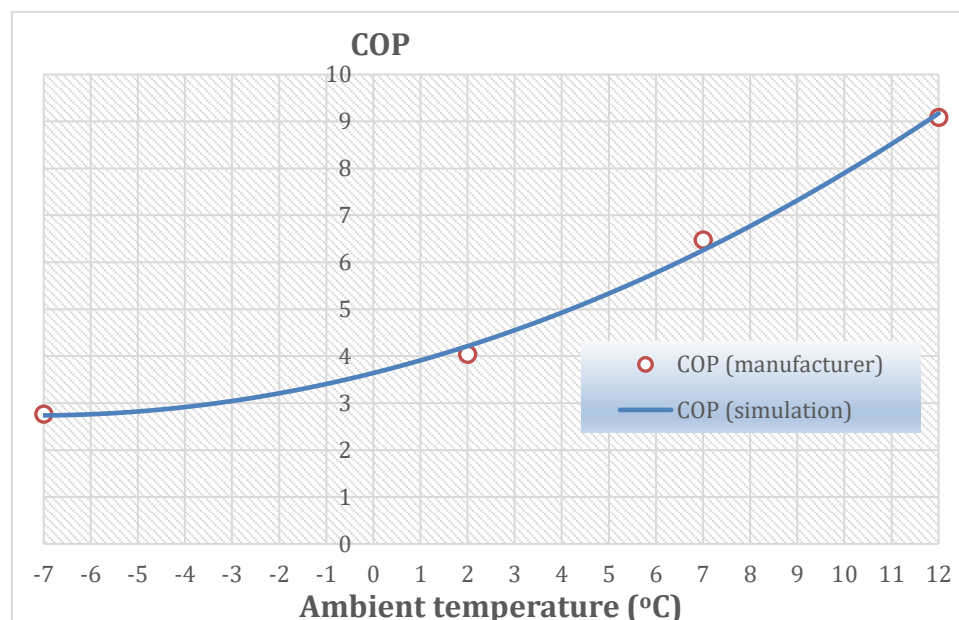


Figure 47: COP simulation results against data provided by manufacturer

Table 24: Elga Ace HP COP results comparison against available manufacturer data

Ambient air temperature in °C	COP (measured)	COP (simulated)	Error	Squared Error
-7	2.770	2.737	0.033	0.001
2	4.040	4.214	-0.174	0.030
7	6.480	6.257	0.223	0.050
12	9.090	9.172	-0.082	0.007
Sum squared error - SSE				0.088
SSE/(Nominal Power) %				1.96%

Figure 48 shows the extracted simulation results of heating power value against all possible ambient temperatures. The sum squared error as a percentage of nominal heating power ( $4.14 \text{ kW}_{\text{th}}$ ) has also been calculated and is equal to 7.25 %. Detailed results are listed in Table 25.

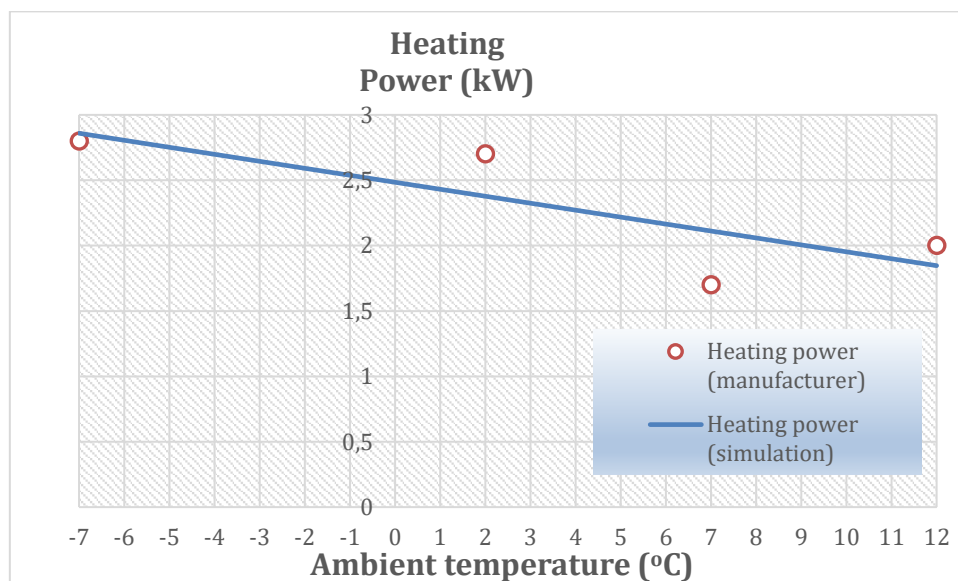


Figure 48: Heating power simulation results against data provided by manufacturer

Table 25: Elga Ace HP heating power results comparison against available manufacturer data

Ambient air temperature in °C	Heating power in kW (measured)	Heating power in kW (simulated)	Error	Squared Error
-7	2.800	2.859	-0.059	0.004
2	2.700	2.378	0.322	0.104
7	1.700	2.112	-0.412	0.170
12	2.000	1.847	0.153	0.023
Sum squared error - SSE				0.300
SSE/(Nominal Power) %				7.25%

## Flywheel

Flywheel Energy Storage Systems (FESS) store energy in rotating masses. The inertia and speed of the rotating mass, dictate the amount of kinetic energy that is stored as rotational energy [40]. The stored rotational energy is given by:

$$E = \frac{1}{2} J \omega^2,$$

where  $J$  is the moment of inertia and  $\omega$  is the angular velocity of the flywheel. Flywheel models can be integrated with asynchronous machines, for large and relatively slow flywheels, whereas synchronous permanent magnet motors are more suitable for small and fast-turning flywheels. The implemented FESS model is presented in Figure 49 and includes the rotor, a permanent magnet generator and two inverters in a back-to-back configuration.

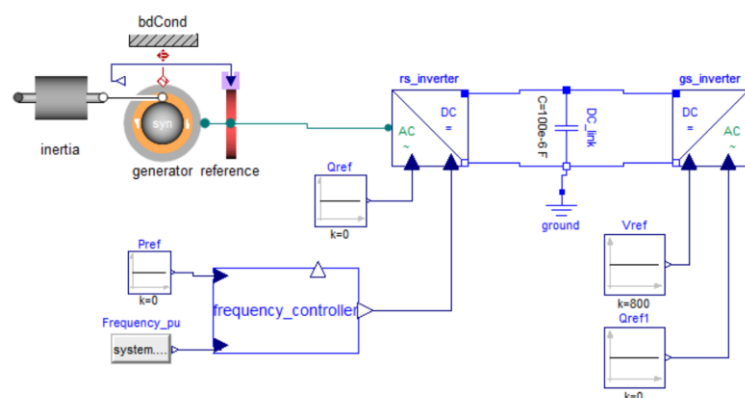


Figure 49. Developed flywheel energy storage model.

FESSs are used in grid stability applications, as they can provide large power for a short duration, usually during a contingency event (e.g., loss of generation, short-circuit). The novel FESS of Teraloop will be installed in Terceira. *Table 26* lists the main parameters of the particular system. Currently, a verification process is undergoing to ensure that the developed model represent accurately Teraloop system.

*Table 26: Teraloop flywheel system parameters.*

Parameter	Value
Capacity (kWh)	3 kWh
Power rating (kW)	100 kW
Roundtrip efficiency (%)	97 %
Depth of discharge (%)	100 %
Discharge duration (minutes)	1,8 minutes
Installation type	Array - containerized or other.
Capital cost / unit	€100,000 + installation fees.

### Hybrid Power Electronic Transformer

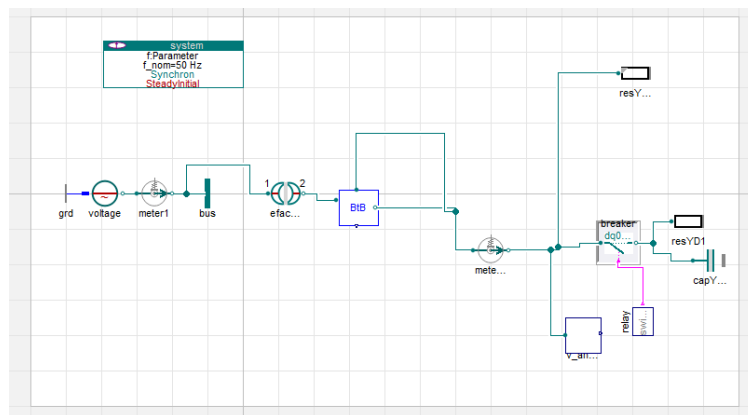
The Hybrid PET (HPET) is a special type of transformer that combines a conventional Low-Frequency Transformer (LFT) with a Back-to-Back (BtB) AC to AC electronic converter. The BtB converter is able to modulate the voltage on the low field side (LFS), acting on each phase independently in order to improve the quality of the voltage at the LFS. Although the efficiency of a typical BtB converter is considerably lower than that of the LFT transformer, the overall efficiency of the HPET transformer is minimally affected since the electronic converter is designed to operate at a fraction of the LFT rated power.

The HPET Modelica model, developed by CERTH, uses the PS Library and is mainly kept at a medium to high level of abstraction, in order to mitigate the computational cost of simulations, in line with the assumptions adopted in [41]. The internal controller acts towards minimizing the error between the voltage at an external setpoint, defined by the user and an ideally balanced signal (in dq0 representation):

$$V_{ref} = \begin{bmatrix} \sqrt{3/2}V_0\cos(\phi(t)) \\ \sqrt{3/2}V_0\sin(\phi(t)) \\ 0 \end{bmatrix}$$



Phase locking between the controller internal phase and the phase at the reference point guarantees that overcorrection by the BtB converter is avoided, in order to maximize the HPET efficiency. *Figure 50* presents an example of HPET usage. A three-phase generator is used to drive a three-phase asymmetric load. A relay controls the dispatch of an additional asymmetric load. The BtB converter located in the middle of the figure regularizes the voltage at the load terminal by controlling each phase independently. Table 27 presents the parameters for the HPET provided by EFACEC following IEC 60076 standard. A verification procedure has been initiated, the results of which will be included in the next reporting period.



*Figure 50: Modelica example of a hybrid power electronic transformer driving a dynamically changing asymmetric load.*

*Table 27: Parameters of the hybrid transformer according to IEC 60076.*

Parameter		Value
Nominal power		400 kVA
# Phases		3
Frequency		50 Hz
Connection		Dyn5
Short Circuit Impedance		0.04
Cooling type		KNAN
Fluid		ESTER
Transformer losses		TIER 2 ECODESIGN
	High Voltage Side	Low Voltage Side
(Off circuit tap changer)	15750 V	400 ±12 % V on load continuous voltage regulation (independently for each phase) with power electronics. The voltage is adjusted by an
	15375 V	
Nominal Voltage	15000 V	
	14625 V	
	14250 V	

		external setpoint provided by the DSO.
Nominal Current	15.4 A	550 A
Insulation Level:	LI 95 AC 38	LI 30 AC 10

### Solid Oxide Fuel Cell

A Modelica model of a domestic Solid Oxide Fuel Cell (SOFC) with heat recovery has been developed based on the TransiENT and ClaRa libraries.

Typical SOFCs are based on the electrochemical reaction of hydrogen (H<sub>2</sub>) oxidation on the anode with oxygen ions (O<sub>2</sub><sup>-</sup>). A minimal operational temperature of approximately 600 °C is required for the O<sub>2</sub><sup>-</sup> transport through the electrolyte. Instead of using H<sub>2</sub> as fuel, CH<sub>4</sub> can also be used after being reformed with steam to produce H<sub>2</sub> and CO:



Reforming can take place in an external component or internally on the anode [42].

CERTH has developed a Modelica model for the “BlueGen” (BG) residential SOFC system generator, with a maximum of 2 kW electrical output which is modulated by the residential load demands [43]. Any load demands beyond 2 kW are assumed to be met by the grid. The system is fuelled by a mixture of steam and methane, which is internally reformed on the anode, utilizing the heat generated by the electrochemical reaction to produce the required hydrogen. Waste heat from the SOFC operation can be recovered as hot water, while the heat generated from the anode exhaust gas afterburner is used to preheat the fuel and generate steam.

The developed model has been based on components provided by the TransiENT and ClaRa libraries, while special care has been taken to guarantee that the model is compatible with components of the PS library (*Figure 51*). Internal reforming is modeled by bringing the “SteamReformer” model provided by the TransiENT library in thermal contact with the anode of the FC component, which is used to model the SOFC electrochemical reaction. The SOFC is represented by the block of components on the left and the load driven by the SOFC by the block of

components on the right. Internal reforming is modelled by bringing the SteamReformer model provided by the TransiEnt library in thermal contact with the anode of the FC component, which is used to model the SOFC electrochemical reaction. The SOFC is used to drive a dynamically changing load (middle right). A power controller (bottom left) drives the SOFC in order to minimize the power flow from the external grid to the load.

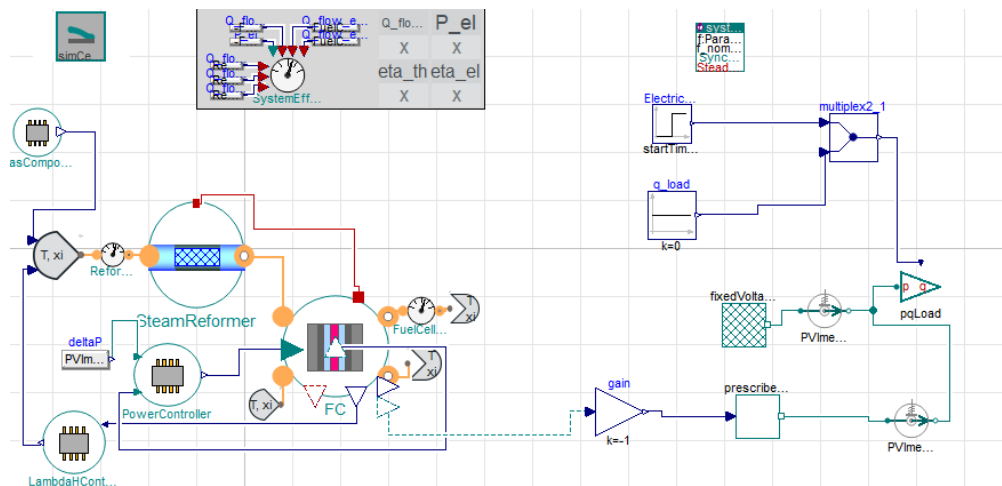


Figure 51: Modelica example featuring a SOFC model operating in conjunction with components of the PowerSystems library.

At the moment, all major modules of the BG system have been accounted for in the Modelica model, with the exception of the exhaust gas afterburner. With this module missing, the calculated efficiencies for the model are lower than the efficiencies reported for the BG system (Figure 52).

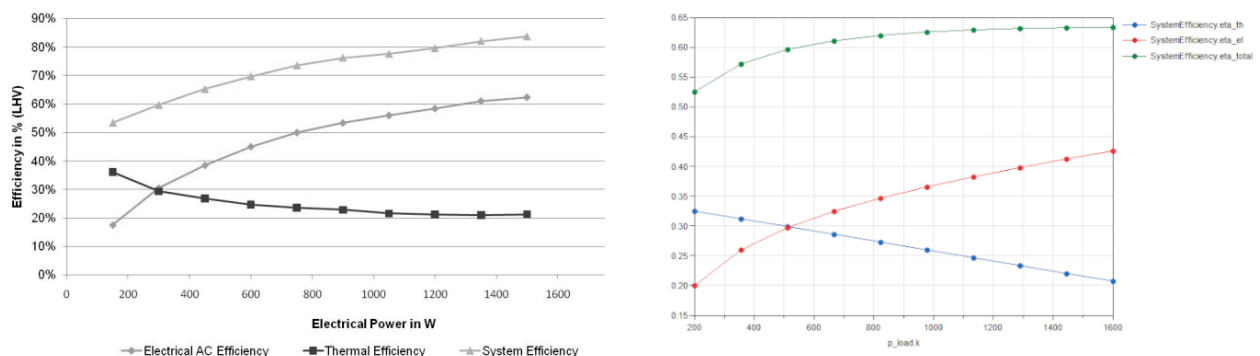



Figure 52: Comparison of the Modelica efficiencies with the experimental data. Left: BG efficiencies as a function of the generated power in W (adapted from [42]). Right: As in left, but for the efficiencies calculated for the Modelica model.

### Proton Exchange Membrane Electrolyzer Cell

Electrolyzers are devices able to produce almost pure hydrogen and oxygen with very small traces of contaminants that can be used to serve the requirements of renewable energy systems. Among the other alternatives PEMs have the advantage of being compact and simple in operation, free of hazardous chemicals, and able to produce hydrogen with high levels of purity [44].

A standalone Modelica model of a Proton Exchange Membrane (PEM) electrolyzer cell has been developed by CETH. The model depends only on the Modelica standard library. The developed model is based on a one-dimensional, semi-static description, where the transient electrical response is ignored, following the modelling approach in [44], [45]. This approach is justified by the experimentally established fast transient electrical response of existing PEM cells.

### Geothermal Power Plant

A thermodynamic model for the Pico Alto geothermal binary cycle power plant located in Terceira [46] has been developed. The plant's binary cycle consists of high enthalpy (over 150 °C) geothermal heat source in steam and brine phases. The secondary cycle is an Organic Rankine Cycle (ORC) with a capacity of 3.5 MW. The plant operates in a binary cycle for the utilization of the geothermal heat source. The first cycle refers to the geothermal fluid (steam & brine) and begins with the fluid entering the system from the production wells and ends with the fluid entering the reinjection well. The second cycle is the Organic Rankine Cycle (ORC) with cyclopentane ( $C_5H_{10}$ -) as the working fluid.

The flow diagram of the binary cycle is shown in Figure 53. In each hot well, the geothermal fluid mixture is separated into two discrete phases (steam & brine) and enters the power plant shell and tubes heat exchangers (two brine preheaters and one steam preheater in parallel and one evaporator in series with the preheaters). The nominal operating conditions of the geothermal cycle are presented in Table 28. Furthermore, both steam and brine enter the reinjection well with a temperature value over 95 °C managed by the plant's control system.

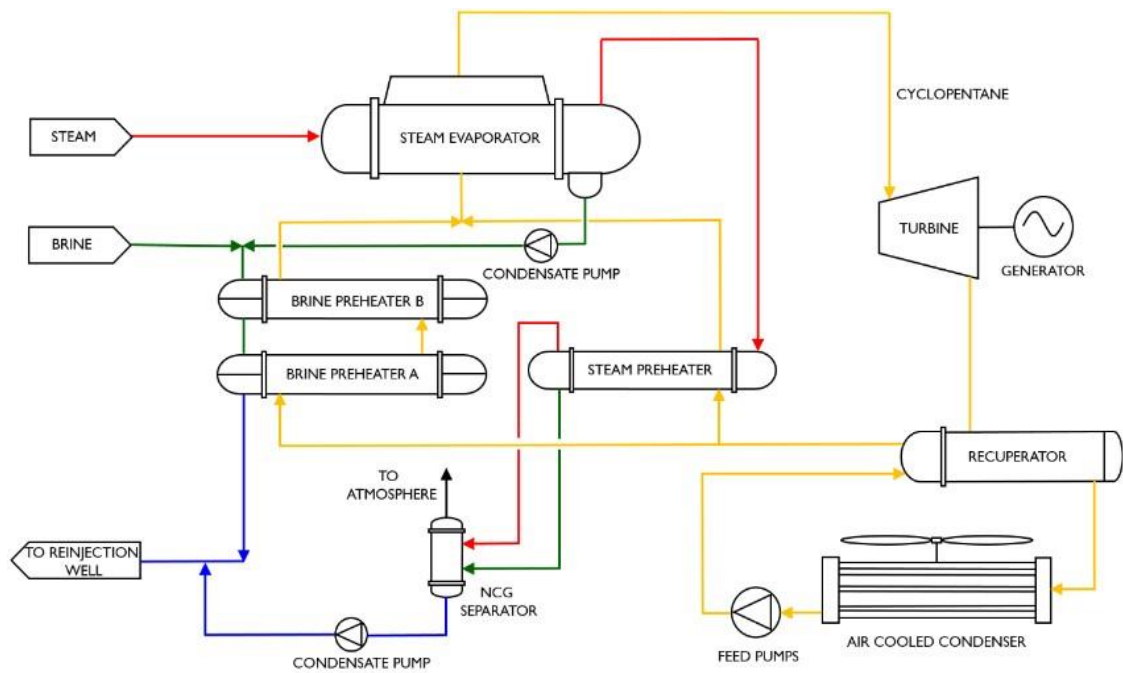


Figure 53: Two-phase recuperated ORC flow diagram [46].

Table 28: Operating conditions of the geothermal plant [46].

	Mass flow rate (kg/s)	Inlet Temperature (°C)	Absolute Pressure (bar)
Steam	9.6	160	5.5
Brine	14	175	10

The liquid cyclopentane is pressurized and driven from the air-cooled condenser to the recuperator tube side, where it will be initially heated by the cyclopentane vapour stream from the turbine outlet. Then it will flow through steam and brine preheaters, where it will be heated until it reaches saturation conditions and then it will enter the steam evaporator where vapour cyclopentane will be produced. Then the saturated vapour will be expanded in the axial inflow - radial outflow turbine. Afterwards, the cyclopentane vapour will flow into the recuperator shell side and finally to the condenser where it will be air-cooled and condensed, reinitiating the cycle. As operation parameters and the thermodynamic states throughout the ORC cycle were not available, the dynamic model was tuned in accordance to the operating data available in the literature (for the specific plant) [47]–[51], changed parameters iteratively until numerical and physical stability was reached.

The model of the plant is shown in Figure 54. The bottom half of the model represents the ORC with cyclopentane as the working medium. At the upper side of the figure, the geothermal cycle is shown, consisting of the brine and steam flow circuits.

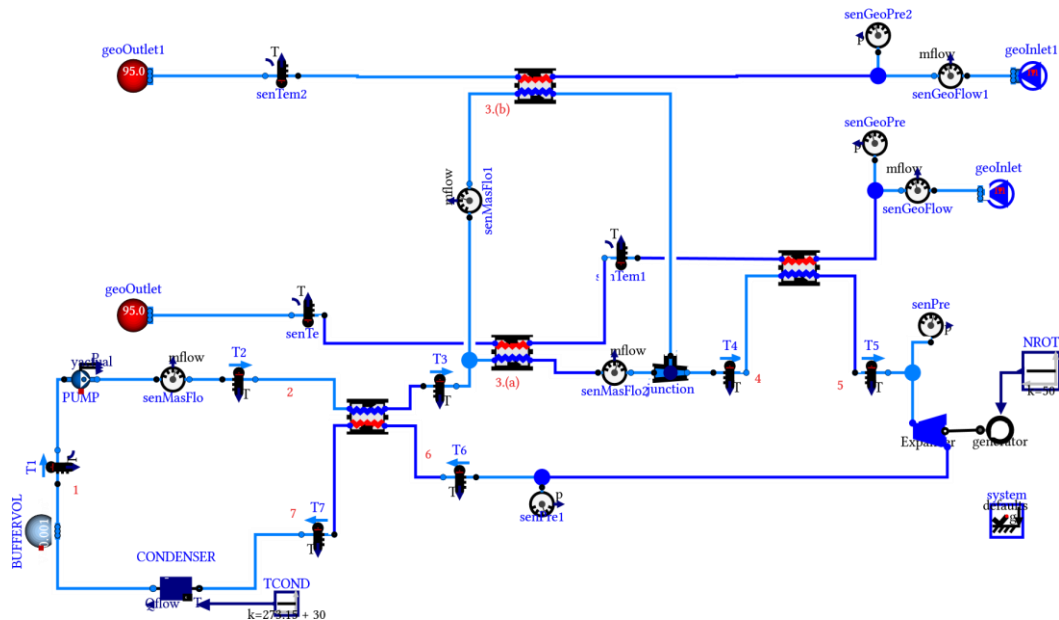


Figure 54: Developed dynamic model for the geothermal power plant in Pico Alto.

The detailed thermodynamic model enables the transient analysis of the plant under consideration. A low order model will be developed based on the insights gained for the integration in the overall island power system model to avoid computational barriers.

### MicroCHP

To proceed with micro – Combined Heat and Power (micro-CHP) representation the CHP model from the open-source library AixLib has been used. This model supports the calculation of both thermal and electrical output power of the unit based on two possible strategies. The first is to satisfy the thermal comfort constraint of a specific temperature setpoint and the second is to cover an electrical load demand profile provided as model input. For both control strategies, the necessary unit capacity is calculated and through Table-based analysis, all CHP variables are given as outputs. These include electrical power, thermal power and fuel consumption. Advanced control algorithms make sure

that operation takes place considering the design and operating constraints of the unit, such as maximum available or minimum allowed capacity.

### PCM storage

Phase-change Material (PCM) storage technology has been modelled with the utilisation of CERTH in-house custom models. These models have undergone evaluation in three EU projects (AMADEUS [52], Plug-N-Harvest [53], and SMILE [18]) and can therefore be considered validated and reliable. A graphical representation of the model is available in Figure 55.

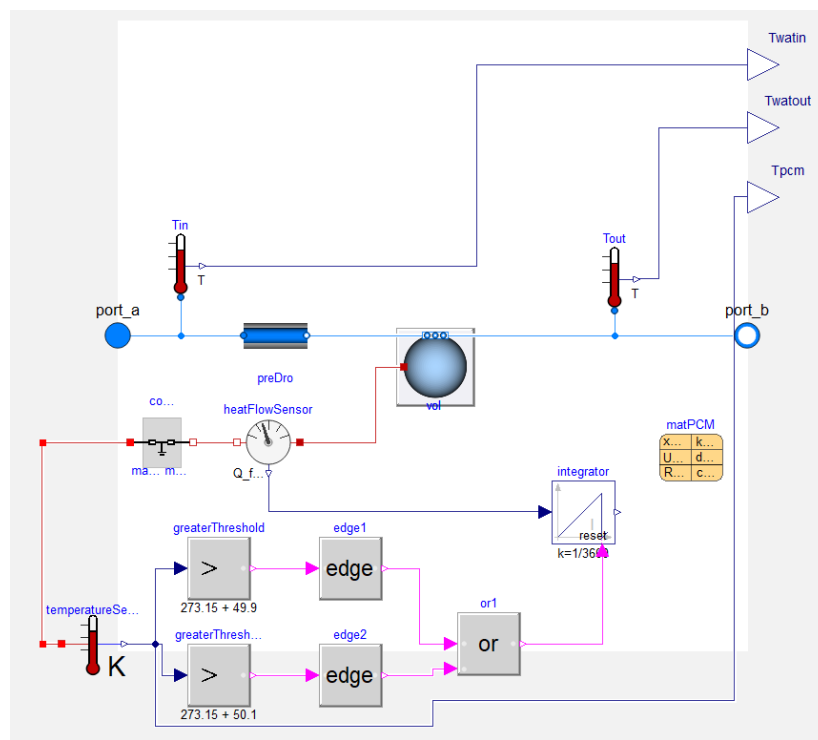


Figure 55: CERTH PCM model

### Autogenerative High Pressure Digestion

An Autogenerative High Pressure Digestion (AHPD) model has been developed by CERTH in Aspen Plus. The AHPD model is an adaptation of Rajendran et al. (2013) process model for anaerobic digestion [54]. Figure 56 presents the nominal operation point of the developed model in a diagram form.

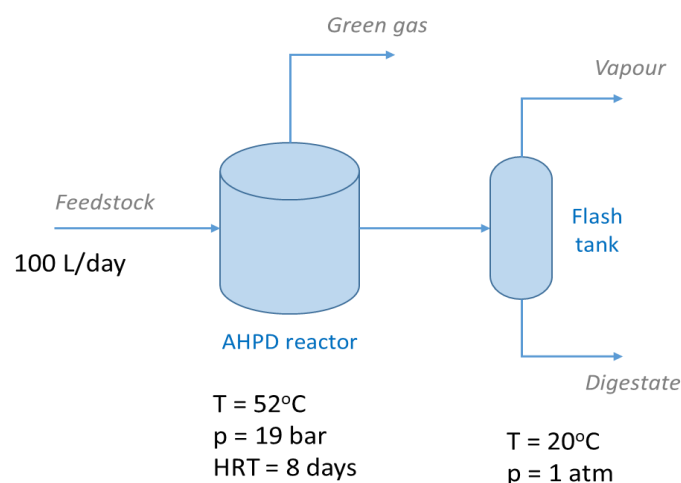


Figure 56: Flow diagram of the Autogenerative High Pressure Digestion process

The feedstock composition was extracted from literature [55] (secondary sludge composition) adapted accordingly and presented in Table 29.

Table 29: Feedstock composition

Substance	Composition (wt%)
Water	99.2
Total solids	0.08
Cellulose	23.59
Hemicellulose	10.72
Soluble proteins	16.22
Insoluble proteins	9.73
Triolein	0.48
Tripalmitin	0.48
Sn-1-Palmito-2-Linolein ( $\text{C}_{37}\text{H}_{68}\text{O}_5$ )	0.48
Dextrose	10.72
$\text{NH}_3$	1.07
Inerts	26.50

The reactor volume was considered 858.3 L (Vapor/Liquid volume ratio= 1/16).

Table 30 presents the stream results in mass fractions. Vapor streams are also presented in mol fractions.



Table 30: Simulation results for the Autogenerative High Pressure Digestion

	Feedstock	Green Gas		Vapour		Digestate
	Mass fraction	Mass fraction	Mole fraction	Mass fraction	Mole fraction	Mass fraction
WATER	0.992	0.00812	0.008368	0.015244	0.024004	0.992784
CO <sub>2</sub>	0	0.208051	0.087765	0.661297	0.426248	0.000761
CH <sub>4</sub>	0	0.778651	0.901081	0.300306	0.531007	1.37E-05
GLYCEROL	0	0	0	0	0	7.63E-06
DEXTROSE	0.000858	0	0	0	0	7.19E-07
PROPIONIC ACID	0	4.06E-06	1.02E-06	5.42E-06	2.08E-06	0.000332
ISOBUTYRIC ACID	0	1.22E-06	2.57E-07	4.44E-06	1.43E-06	0.000351
ISOVALERIC ACID	0	1.24E-06	2.25E-07	1.77E-06	4.91E-07	6.20E-05
NH <sub>3</sub>	8.58E-05	2.88E-07	3.14E-07	1.76E-06	2.93E-06	5.83E-07
NH <sub>4</sub> <sup>+</sup>	0	0	0	0	0	0.00031
ETHYL-CYANOACETATE	0	0	0	0	0	0.000165
HYDROGEN	0	1.35E-05	0.000124	4.36E-06	6.14E-05	0
H <sub>2</sub> S	0	0.004643	0.002529	0.021832	0.018171	7.51E-05
BENZENE	0	0.000459	0.000109	0.001187	0.000431	8.02E-06
HCO <sub>3</sub> <sup>-</sup>	0	0	0	0	0	0.000988
HS <sup>-</sup>	0	0	0	0	0	1.90E-05
CELLULOSE	0.001887	0	0	0	0	0.000567
HEMICELLULOSE	0.000858	0	0	0	0	0.000344
TRIOLEIN	3.84E-05	0	0	0	0	3.85E-06
TRIPALMITIN	3.84E-05	0	0	0	0	3.85E-06
SN-1-PALMITO-2-LINOLEIN (C <sub>37</sub> H <sub>68</sub> O <sub>5</sub> )	3.84E-05	0	0	0	0	3.85E-06
XYLOSE	0	0	0	0	0	9.76E-05
ETHANOL	0	5.54E-05	2.23E-05	0.000115	7.10E-05	0.00043
SOLUBLE PROTEIN (C <sub>13</sub> H <sub>25</sub> O <sub>7</sub> N <sub>3</sub> S)	0.001298	0	0	0	0	0.00013
INSOLUBLE PROTEIN (C <sub>4.39</sub> H <sub>8</sub> NO <sub>2.1</sub> )	0.000779	0	0	0	0	0.00039
ACETATE	0	0	0	0	0	2.45E-05
INERT	0.00212	0	0	0	0	0.002123
Total Mass flow (kg/day)	99.88	0.0799		0.0807		99.72

The green gas Yield is 109.1 Nm<sup>3</sup> green gas / tons dry input and the CH<sub>4</sub> content in green gas is 70.8% of the total CH<sub>4</sub> produced. A parametric investigation has been performed on the developed model in ASPEN plus, sweeping key parameters across the range of values the plant is designed to operate. The resulted operation “maps” were imported in Modelica, adequately encapsulating the system’s behaviour while also avoiding computational burdens.



## 9. References

- [1] “(5) Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives | Request PDF.”  
[https://www.researchgate.net/publication/233747780\\_Software\\_Systems\\_Architecture\\_Working\\_With\\_Stakeholders\\_Using\\_Viewpoints\\_and\\_Perspectives](https://www.researchgate.net/publication/233747780_Software_Systems_Architecture_Working_With_Stakeholders_Using_Viewpoints_and_Perspectives) (accessed Feb. 22, 2022).
- [2] “ISO - ISO/IEC/IEEE 42010:2011 - Systems and software engineering — Architecture description.”  
<https://www.iso.org/standard/50508.html> (accessed Feb. 22, 2022).
- [3] “Welcome To UML Web Site!” <https://www.uml.org/> (accessed Feb. 22, 2022).
- [4] “API Documentation & Design Tools for Teams | Swagger.” <https://swagger.io/> (accessed Feb. 22, 2022).
- [5] “What are microservices?” <https://microservices.io/> (accessed Feb. 23, 2022).
- [6] “Use Cases | Docker.” <https://www.docker.com/use-cases> (accessed Feb. 23, 2022).
- [7] “EUR-Lex - 32013R0347 - EN - EUR-Lex.” <https://eur-lex.europa.eu/legal-content/en/TXT/?uri=celex%3A32013R0347> (accessed Feb. 25, 2022).
- [8] “Cost Benefit Analysis.” <https://tyndp.entsoe.eu/cba> (accessed Mar. 01, 2022).
- [9] “Smart grid cost-benefit analysis | JRC Smart Electricity Systems and Interoperability.”  
<https://ses.jrc.ec.europa.eu/smart-grid-cost-benefit-analysis> (accessed Mar. 01, 2022).
- [10] “ESDL Documentation - ESDL.” <https://energytransition.gitbook.io/esdl> (accessed Feb. 28, 2022).
- [11] “ESDLMapEditorESSIM/docker-toolsuite: All information to install, setup and run the ESDL MapEditor and ESSIM toolsuite on your local machine.”  
<https://github.com/ESDLMapEditorESSIM/docker-toolsuite> (accessed Feb. 28, 2022).
- [12] T. Brown, J. Hörsch, and D. Schlachtberger, “{PyPSA}: {Python} for {Power} {System} {Analysis},” *Journal of Open Research Software*, vol. 6, Jan. 2018, doi: 10.5334/jors.188.
- [13] L. Thurner et al., “pandapower - an {Open} {Source} {Python} {Tool} for {Convenient} {Modeling}, {Analysis} and {Optimization} of {Electric} {Power} {Systems},” *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 6510–6521, Nov. 2018, doi: 10.1109/TPWRS.2018.2829021.
- [14] A. D. Dominguez-Garcia, S. T. Cady, and C. N. Hadjicostis, “Decentralized optimal dispatch of distributed energy resources,” in *2012 {IEEE} 51st {IEEE} {Conference} on {Decision} and {Control} ({CDC})*, Dec. 2012, pp. 3688–3693. doi: 10.1109/CDC.2012.6426665.
- [15] J. A. Taylor, *Convex {Optimization} of {Power} {Systems}*; 1st ed. Cambridge University Press, 2015. doi: 10.1017/CBO9781139924672.



- [16] A. J. Conejo and L. Baringo, "Unit {Commitment} and {Economic} {Dispatch}," in *Power {System} {Operations}*, Cham: Springer International Publishing, 2018, pp. 197–232. doi: 10.1007/978-3-319-69407-8\_7.
- [17] P. Iliadis, S. Ntomalis, K. Atsonios, A. Nesiadis, N. Nikolopoulos, and P. Grammelis, "Energy management and techno-economic assessment of a predictive battery storage system applying a load levelling operational strategy in island systems," *International Journal of Energy Research*, vol. 45, no. 2, pp. 2709–2727, Oct. 2020, doi: 10.1002/er.5963.
- [18] "{SMILE} {H2020}." Accessed: Feb. 15, 2022. [Online]. Available: <https://www.h2020smile.eu/>
- [19] "Renaissance {H2020} project towards sustainable energy communities." Accessed: Oct. 02, 2022. [Online]. Available: <https://www.renaissance-h2020.eu/>
- [20] E. Molin, "Comparison of {Single}-{Page} {Application} {Frameworks} - {A} method of how to compare {Single}-{Page} {Application} frameworks written in {JavaScript}," Stockholm, Sweden, 2016.
- [21] W. Group, "Common {Format} {For} {Exchange} of {Solved} {Load} {Flow} {Data}," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-92, no. 6, pp. 1916–1925, Nov. 1973, doi: 10.1109/TPAS.1973.293571.
- [22] R. Franke and H. Wiesmann, "Flexible modeling of electrical power systems -- the {Modelica} {PowerSystems} library," Mar. 2014, pp. 515–522. doi: 10.3384/ecp14096515.
- [23] "A European Green Deal | European Commission." [https://ec.europa.eu/info/strategy/priorities-2019-2024/european-green-deal\\_en](https://ec.europa.eu/info/strategy/priorities-2019-2024/european-green-deal_en) (accessed Mar. 08, 2022).
- [24] "SAREF: the Smart Applications REference ontology." <https://saref.etsi.org/core/v3.1.1/> (accessed Mar. 08, 2022).
- [25] M. E. El-Hawary, *Electrical power systems: design and analysis*, Rev. print. New York: Institute of Electrical and Electronics Engineers, 1995.
- [26] P. Kundur, N. J. Balu, and M. G. Lauby, *Power system stability and control*. New York: McGraw-Hill, 1994.
- [27] A. Arif, Z. Wang, J. Wang, B. Mather, H. Bashualdo, and D. Zhao, "Load {Modeling} – {A} {Review}," *IEEE Transactions on Smart Grid*, p. 1, 2017, doi: 10.1109/TSG.2017.2700436.
- [28] A. S. Leizerovich, *Steam turbines for modern fossil-fuel power plants*. Lilburn, GA : Boca Raton, FL: Fairmont Press ; Distributed by Taylor & Francis/CRC Press, 2008.
- [29] X. Wu, J. Shen, Y. Li, and K. Y. Lee, "Steam power plant configuration, design, and control," *WIREs Energy and Environment*, vol. 4, no. 6, pp. 537–563, Nov. 2015, doi: 10.1002/wene.161.



- [30] NEPLAN, "Standard {Dynamic} {Turbine}-{Governor} {Systems} in {NEPLAN} {Power} {System} {Analysis} {Tool}." [Online]. Available: [https://www.neplan.ch/wp-content/uploads/2015/08/Nep\\_TURBINES\\_GOV.pdf](https://www.neplan.ch/wp-content/uploads/2015/08/Nep_TURBINES_GOV.pdf)
- [31] NEPLAN, "Standard {Dynamic} {Excitation} {Systems} in {NEPLAN} {Power} {System} {Analysis} {Tool}."
- [32] NEPLAN, "Standard {Dynamic} {Power} {System} {Stabilizers} in {NEPLAN} {Power} {System} {Analysis} {Tool}."
- [33] M. P. Boyce, *Gas turbine engineering handbook*, 4th ed. Amsterdam ; Boston: Elsevier/Butterworth-Heinemann, 2012.
- [34] E. Hau, *Wind {Turbines}*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. doi: 10.1007/978-3-642-27151-9.
- [35] G. A. Munoz-Hernandez, S. P. Mansoor, and D. I. Jones, *Modelling and {Controlling} {Hydropower} {Plants}*. London: Springer London, 2013. doi: 10.1007/978-1-4471-2291-3.
- [36] M. Chen and G. A. Rincon-Mora, "Accurate {Electrical} {Battery} {Model} {Capable} of {Predicting} {Runtime} and {I}-[V] {Performance}," *IEEE Transactions on Energy Conversion*, vol. 21, no. 2, pp. 504–511, Jun. 2006, doi: 10.1109/TEC.2006.874229.
- [37] M. Ceraolo, "New dynamical models of lead-acid batteries," *IEEE Transactions on Power Systems*, vol. 15, no. 4, pp. 1184–1190, Nov. 2000, doi: 10.1109/59.898088.
- [38] T. Huria, M. Ceraolo, J. Gazzarri, and R. Jackey, "High fidelity electrical model with thermal dependence for characterization and simulation of high power lithium battery cells," in *2012 {IEEE} {International} {Electric} {Vehicle} {Conference}*, Mar. 2012, pp. 1–8. doi: 10.1109/IEVC.2012.6183271.
- [39] Elga Ace, "Installatie-, gebruikers- en servicehandleiding {Hybride} 'split' warmtepomp lucht / water." Dec. 2020. [Online]. Available: <https://tools.remeha.nl/wp-content/uploads/sites/3/2020/06/IOM-Elga-Ace-NOT-7745647-05.pdf>
- [40] B. Bolund, H. Bernhoff, and M. Leijon, "Flywheel energy and power storage systems," *Renewable and Sustainable Energy Reviews*, vol. 11, no. 2, pp. 235–258, Feb. 2007, doi: 10.1016/j.rser.2005.01.004.
- [41] F. Prystupczuk, V. Rigoni, A. Nouri, R. Ali, A. Keane, and T. O'Donnell, "Hybrid {Power} {Electronic} {Transformer} {Model} for {System}-{Level} {Benefits} {Quantification} in {Energy} {Distribution} {Systems}," *Frontiers in Electronics*, vol. 2, p. 716448, Sep. 2021, doi: 10.3389/felec.2021.716448.
- [42] S. A. Saadabadi, N. van Linden, A. Heinsbroek, and P. v Aravind, "A solid oxide fuel cell fuelled by methane recovered from groundwater," *Journal of Cleaner Production*, vol. 291, p. 125877, Apr. 2021, doi: 10.1016/j.jclepro.2021.125877.
- [43] K. Föger and T. Rowe, "Ceramic fuel cells ltd residential generator {BlueGen}, ultra-efficient distributed power generation in smart grid," 2011, p. 12.



- [44] Z. Abdin, C. J. Webb, and E. MacA. Gray, "Modelling and simulation of a proton exchange membrane ({PEM}) electrolyser cell," *International Journal of Hydrogen Energy*, vol. 40, no. 39, pp. 13243–13257, Oct. 2015, doi: 10.1016/j.ijhydene.2015.07.129.
- [45] H. Gorgun, "Dynamic modelling of a proton exchange membrane ({PEM}) electrolyzer," *International Journal of Hydrogen Energy*, vol. 31, no. 1, pp. 29–38, Jan. 2006, doi: 10.1016/j.ijhydene.2005.04.001.
- [46] A. Franco, N. Vieira, C. Ponte, and E. Renováveis, "Geothermal {Developments} in {Pico} {Alto}, {Terceira} {Island}, {Portugal}," in *Geothermal {Resources} {Council} {Transactions}*, 2017, vol. 41, p. 13.
- [47] S. Frick, S. Kranz, G. Kupfermann, A. Saadat, and E. Huenges, "Making use of geothermal brine in {Indonesia}: binary demonstration power plant {Lahendong}/{Pangolombian}," *Geothermal Energy*, vol. 7, no. 1, p. 30, Dec. 2019, doi: 10.1186/s40517-019-0147-2.
- [48] G. Andritsos, A. Desideri, C. Gantiez, and V. Lemort, "Steady state and dynamic modelling of a 1 {MWel} commercial waste heat recovery {ORC} power plant," 2016, p. 12.
- [49] M. Imran, R. Pili, M. Usman, and F. Haglind, "Dynamic modeling and control strategies of organic {Rankine} cycle systems: {Methods} and challenges," *Applied Energy*, vol. 276, p. 115537, Oct. 2020, doi: 10.1016/j.apenergy.2020.115537.
- [50] H. Zhai, L. Shi, and Q. An, "Influence of working fluid properties on system performance and screen evaluation indicators for geothermal {ORC} (organic {Rankine} cycle) system," *Energy*, vol. 74, pp. 2–11, Sep. 2014, doi: 10.1016/j.energy.2013.12.030.
- [51] A. F. Altun and M. Kilic, "Thermodynamic performance evaluation of a geothermal {ORC} power plant," *Renewable Energy*, vol. 148, pp. 261–274, Apr. 2020, doi: 10.1016/j.renene.2019.12.034.
- [52] "{AMADEUS} {Project} - {Electric} {Thermal} {Energy} {Storage}." Accessed: Feb. 15, 2022. [Online]. Available: <https://amadeus-project.eu/>
- [53] "Plug-{N}-{Harvest} {H2020} {\textbar} {Plug}-{N}-{Harvest} {H2020}." Accessed: Feb. 15, 2022. [Online]. Available: <https://www.plug-n-harvest.eu>
- [54] K. Rajendran, H. R. Kankanala, M. Lundin, and M. J. Taherzadeh, "A novel process simulation model ({PSM}) for anaerobic digestion using {Aspen} {Plus}," *Bioresource Technology*, vol. 168, pp. 7–13, Sep. 2014, doi: 10.1016/j.biortech.2014.01.051.
- [55] T. A. Marcinkowski, "Effect of alkalization process on changes in the chemical composition of secondary sludge," *Environment Protection Engineering*, vol. 36, no. 2, pp. 153–160, 2010.