STREAMING FLEXIBILITY TO THE POWER SYSTEM

D3.3: sGRID AND sPLAN GRID PLANNING AND SERVICE MANAGEMENT TOOL

Delivery Date: March 2025

STREAMING FLEXIBILITY TO THE POWER SYSTEM

DISCLAIMER

COPYRIGHT STATEMENT

This project has received funding from the European Union's Horizon Europe research and innovation programme under the grant agreement N° 101075654.

More information available at https://stream-he-project.eu/

Title	Document Version
D3.3: sGRID and sPLAN grid planning and service management tool	1.0

Project Number	Project Acronym	Project Title
Grant agreement N°: 101075654	STREAM	STREAM: Streaming flexibility to the power system

Contractual Delivery Date	Actual Delivery Date	Deliverable Type*-Security**
March 2025 (M30)	March 2025 (M30)	O-PU

*Type: P: Prototype; R: Report; D: Demonstrator; O: Other.

**Security Class: Public - fully open (automatically posted online); Sensitive - limited under the conditions of the Grant Agreement; EU classified - RESTREINT-UE/EU-RESTRICTED, CONFIDENTIEL-UE/EU-CONFIDENTIAL, SECRET-UE/EU-SECRET under Decision 2015/444

Responsible	Organisation	Contributing WP
Daniel Pastor Pascual	ETRA	WP3

Authors (organisation)

Daniel Pastor Pascual (ETRA), Diego García-Casarrubios Gálvez (ETRA), Klemen Peter Kosovinc (KOL), Gašper Artač (KOL), Denis Sodin (COMS), Alessio Cavadenti (ASM), Prashanth Kumar Pedholla (ASM), Janez Gregor Golja (IRI UL), Matej Pecjak (IRI UL), Bernardo Di Chiara (OPT), Mikko Tuohimaa (OPT), Efren Gullio (ENER), Bogomir Zidaric (EPR)

Abstract

This deliverable reports on the work and contributions of the partners related to Task 3.4: Grid Planning and Service Management Tool. This task includes the development and documentation of sGRID, a tool for managing distribution networks, and sPLAN, a tool for distribution planning and analysis. Following the two subtasks of this task, the data analytics used across the different tools in the pilot sites are examined, along with the user profiling approach at each pilot site, focusing on one asset per site.

Keywords

Distribution System Operator (DSO), data analytics, user profiling, energy generation forecast, energy consumption forecast, energy flexibility forecast, baseline, congestion management, distribution network planning, power flow, time series, network topology.

Revision	Date	Description	Author (Organisation)
V0.1	30.04.2024	New document & first version of the Table of Contents	Daniel Pastor Pascual (ETRA), Diego García-Casarrubios Gálvez (ETRA)
V0.2	18.03.205	Contributions prepared by the relevant partners.	Daniel Pastor Pascual (ETRA), Diego García-Casarrubios Gálvez (ETRA), Klemen Peter Kosovinc (KOL), Denis Sodin (COMS), Alessio Cavadenti (ASM), Prashanth Kumar Pedholla (ASM), Janez Gregor Golja (IRI UL), Matej Pecjak (IRI UL), Bernardo Di Chiara (OPT), Mikko Tuohimaa (OPT), Efren Gullio (ENER), Bogomir Zidaric (EPR)
V0.3	18.03.2025	Merge of the contributions from all the partners. Executive summary, introduction and conclusions added. Document ready for peer review.	Daniel Pastor Pascual (ETRA), Diego García-Casarrubios Gálvez (ETRA)
V0.4	24.03.2025	Peer review provided by IND, KOL, VTT and CYG.	Valerija Jordan Trsinar (IND), Klemen Peter Kosovinc (KOL), Mäki Kari (VTT), Darja Skrt Georgiev (CYG)
V0.5	31.03.2025	Modifications provided by the relevant partners addressing the peer review's feedback.	Daniel Pastor Pascual (ETRA), Diego García-Casarrubios Gálvez (ETRA), Klemen Peter Kosovinc (KOL), Denis Sodin (COMS), Alessio Cavadenti (ASM), Prashanth Kumar Pedholla (ASM), Janez Gregor Golja (IRI UL), Matej Pecjak (IRI UL), Bernardo Di Chiara (OPT), Mikko Tuohimaa (OPT), Efren Gullio (ENER), Bogomir Zidaric (EPR)
V1.0	31.03.2025	Document merged and cleaned. Final version of the deliverable ready for submission.	Daniel Pastor Pascual (ETRA), Diego García-Casarrubios Gálvez (ETRA)

EXECUTIVE SUMMARY

This document reports the **development and implementation** of the **sGRID** and **sPLAN** tools within the **STREAM pilot sites,** which constitutes T3.4 Grid planning and service management tool. **sGRID** is an operation and management tool designed to help **Distribution System Operators (DSOs)** forecast network issues (e.g., **congestion** and/or **voltage violations**) within their grids and, allowing DSOs to participate in the **flexibility market** to address these issues. **sPLAN** is a tool for planning and analysing used to identify future **reinforcement measures**, both with and without flexibility provision.

Following the subtasks of this task, the **data analytics** used across the different tools at the pilot sites—including not only **sGRID** and **sPLAN**, but also **sFLEX**, **sENC**, and **sDATA**—are examined. Additionally, the **methodologies for user profiling** of the assets at the pilot sites are reviewed, with **baseline** and **flexibility forecasts** of one asset from each pilot site used as examples.

The **sGRID** tool has been **tailor-developed** for the Slovenian, Spanish, and Italian pilot sites, focusing on calculating the flexibility for **DSOs' network need for congestion management**. The architecture and functionalities of the different versions of the tool are detailed in the document.

The **sPLAN** tool has been developed exclusively for the Slovenian pilot site. The **sPLAN section** also outlines the different functionalities implemented in the tool.

Data analytics is essential across all tools developed in the **STREAM project**, as data must be processed to generate the necessary outputs for DSOs, aggregators, and energy communities to participate in flexibility markets. The **data analytics models** used at the pilot sites are listed in the document, with an explanation for their use, the required inputs, and the analysis process to obtain the desired outputs. In total, **15 data analytics models** are used across the pilot sites.

The **baseline** and **flexibility forecast calculations**, which together form the **user profiling** of flexibility assets, are particularly detailed in separate subsections for each pilot site due to the significance of this data. **User profiling** optimizes the use of pilot site assets in flexibility markets. To keep the document **concise yet comprehensive**, each pilot site details its user profiling approach, using one asset from the site as an example. The Slovenian pilot site focuses on the **cold storage room of Incom Leone**, the Spanish pilot site on the **energy community battery**, the Italian pilot site on the **water pump**, and the Finnish pilot site on the **aggregation of the heating systems of their users**.

The development of these tools enables **DSOs** to identify network issues and participate in the flexibility market. The **data analytics models** and **user profiling algorithms** used in the tools are critical for both sides of the flexibility market: for **DSOs** to monitor and estimate the status of their grids and send flexibility requirements to the market, and for **aggregators** to understand the flexibility their users can provide during a specific market session.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 PURPOSE OF THE DOCUMENT

The purpose of this document is to report on the activities undertaken in **Task 3.4: Grid Planning and Service Management Tool**, and its two subtasks: **Subtask 3.4.1: Data Analytics** and **Subtask 3.4.2: User Profiling**.

T3.4 entails the development of the **sGRID** and **sPLAN** tools. Both tools are targeted at **DSOs**, with **sGRID** serving as a platform to **monitor and operate the grid**, enabling the **detection and forecasting of congestions** to send requirements to **flexibility markets**, and **sPLAN** supporting **DSOs** in **analyzing and planning grid reinforcement**. **sGRID** is being deployed in the **Slovenian, Spanish, and Italian pilot sites**, whereas **sPLAN** is being deployed only in the **Slovenian pilot site**. Each pilot site has developed its own version of the tools to adapt them to the needs of their respective **DSOs**. The **architecture and functionalities** of each tool are documented in this deliverable.

ST3.4.1 examines the **data analytics models** that have been integrated into the different tools across the four pilot sites of the **STREAM project**. A **comprehensive analysis** of each model per pilot site is provided in their respective sections, identifying the **tools that use each model**, the **input data required** to run the algorithms, as well as the **processing steps and output data** generated by the models.

ST3.4.2 focuses on developing **user profiling algorithms** to establish **baseline and flexibility forecasts** for the different assets involved in the four pilot sites. Each pilot site has selected **one asset** to exemplify the algorithm used for **user profiling**.

The deployment of **sGRID** and **sPLAN** will continue through **WP5**, where progress toward the **deployment and demonstration** of the tools will be monitored and documented in **Deliverable 5.1: Demonstration Activities Report**.

1.2 SCOPE OF THE DOCUMENT

This deliverable documents the development of the sGRID and sPLAN tools under T3.4, providing an overview of the tools for each pilot site, their architecture, and a detailed analysis of their functionalities. The data analytics models used in the different tools across the pilot sites are examined in ST3.4.1, along with the methodologies and algorithms applied for user profiling of the flexibility assets in ST3.4.2.

1.3 STRUCTURE OF THE DOCUMENT

The document is structured as follows:

- **Section 2** presents the framework of the developments undertaken in task T3.4. First, it provides an analysis of the needs and responsibilities of the DSOs in the STREAM project, explaining how sGRID and sPLAN help DSOs address these needs. Second, it introduces data analytics, outlining the typical steps involved in data processing and analysis. Finally, the user profiling introduction subsection offers a comprehensive overview of the generic user profiling methodology, including baseline and flexibility forecasting.
- **Sections 3, 4, 5 and 6** include, for each pilot site, the following contents:
    - **Sections X.1** present and analyse the data analytics models of the Slovenian, Spanish, Italian, and Finnish pilot sites. They include a list of the data analytics models implemented at each site, an introduction to these models, the input data required for proper operation, and the data processing and analysis procedures. This includes

the steps followed by the model, a description of the algorithm, and the resulting output data.

- **Sections X.2** examine the user profiling methodologies and algorithms used for baseline and flexibility forecasting for individual assets or aggregations across the pilot sites.

- **Sections X.3** provide an overview and defines the functionalities of the sGRID tools developed in the Slovenian, Spanish, and Italian pilot sites.
    - **Sections X.4** provide an overview and defines the functionalities of the sPLAN tool developed in the Slovenian pilot site.
- Finally, **section 7** provides the conclusions of the document, summarising the main findings and results of the task.

2 DEVELOPMENTS FRAMEWORK

2.1 DSO NEEDS AND RESPONSIBILITIES IN THE STREAM ECOSYSTEM

This section will provide an overview of the challenges DSOs are facing in the EU, starting with the traditional role of DSOs and their rapid evolution in the recent years and those to come, with a special focus on challenges related to the STREAM project. The outcomes of this section will serve as a basis for introducing the sPLAN and sGRID tools, their significance for DSOs and their objectives within the STREAM ecosystem.

2.1.1 Evolution of DSO's role and responsibilities in the European power systems

When DSOs were first created, their role in Europe was to manage the Medium Voltage (MV) and Low Voltage (LV) grids in the European power systems. This role involves ensuring the safe and secure energy supply for the energy consumers. In the traditional power systems, in which a few great generators (mainly thermal power plants fuelled by coal and large hydroelectric power plants) were connected to the high voltage grid, DSOs were only responsible for the supply of electricity for the end-users, who were traditional consumers without generation technologies. As such, DSOs were primarily concerned with estimating the medium-term new supply connections within their grids to invest in and install new equipment (transformers, substations, and new feeders) to meet future capacity requirements in order to avoid grid congestions, which jeopardizes the security of the energy supply.

However, with the appearance of small generators – mainly Renewable Energy Sources (RES) – connected to the MV and LV grid, the planning of the grids also required considering the installation of such technologies. The direct implication of this paradigm shift implies that the power flows are not only unidirectional (from the interconnection with the High Voltage (HV) grid downstream to the consumption points), but bidirectional, which increases the complexity of managing both the MV and LV distribution grids. This phenomenon has been exacerbated by the boom in self-consumption, EVs, which increase the needs of energy for their charging and, in the case of bidirectional charger capabilities, increases the complexity of the power flows, and Battery Energy Storage Systems (BESS), which altogether constitute the Distributed Energy Resources (DERs). These installations are most commonly connected to the LV grid, increasing the bidirectionality of the power flows and, consequently, further complicating the management of the grid. The evolution from traditional power systems to the emerging scenarios is illustrated in Figure 1 Moreover, the intermittent nature of the renewable energy production also involves grid stability issues, with increasing voltage deviations that can cause problems in the consumption assets, and less predictable power flows.

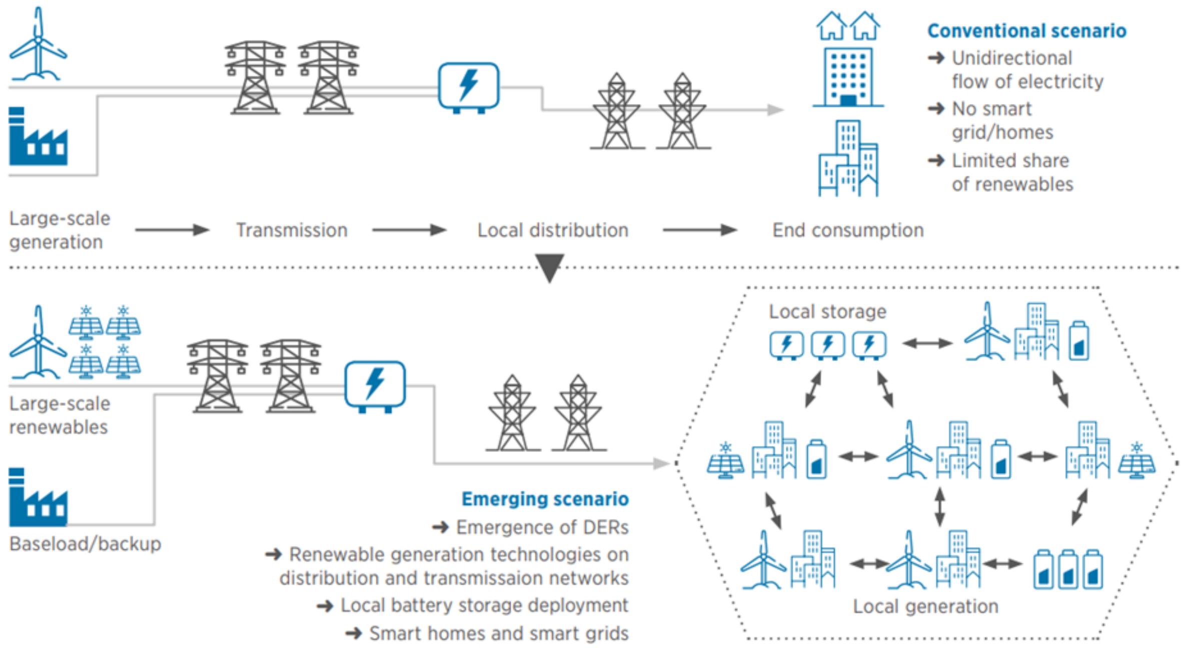


Figure 1: Conventional scenario versus emerging scenario in the power system. Source [1].

The International Renewable Energy Agency (IRENA) sees all these challenges as opportunities for DSOs to bet on innovation [1], with the creation of new business models (as already explored in this project in [2]). This flexibility can also be obtained by introducing prosumers as market players. With all these players, DSOs can obtain and procure flexibility services to manage network congestions and control voltage deviations. In turn, when considering this flexibility provision in the network planning, the investment in new equipment can be significantly reduced, since the optimal operation management of flexibility resources leads to a reduction of peak loads in the grid. Moreover, this optimization also helps DSOs to easily integrate more RES within their grids, since the BESS of prosumers and larger BESS installed directly in the grid can, for example, store PV generation during peak renewable energy generation to inject it during night hours, consequently reducing curtailments.

However, achieving these new roles is not easy. The digitalization of the grids and the integration of smart grids technologies are essential. These deployments ensure the appropriate level of management to establish the required communications with the assets that can provide this flexibility by enabling automatic control capabilities. Near-real-time monitoring is also crucial to estimate the live state of the grid to detect potential issues in the grid (mainly congestions and voltage violations). At the same, reliability of distributed flexibility volumes becomes a very important aspect to be considered by DSO.

2.1.2 DSOs' needs: the STREAM tools to unlock the new roles

Landing in the STREAM project, the pilot sites participating in the project have already integrated DERs within their grids and have installed smart grid technologies (e.g., SCADA systems, smart meters, actuators). Thus, they are facing these challenges and want to take advantage of the opportunities. To do so, two tools are being developed and deployed:

- **sPLAN:** the objective of sPLAN is to provide DSOs with a tool to help them with the decision-making process of grid planning. With this tool, DSOs are able to compare results considering just the traditional solutions (investing in new grid equipment) with the innovative solutions, such as considering the flexibility available in the grid. This tool has been deployed in the Slovenian pilot site.

- **sGRID:** by using standard communication protocols (e.g., MQTT, REST API, and OCPP), sGRID is a tool that will be connected to the sGRID ICT infrastructure (e.g., Advanced Metering Infrastructure (AMI) system, SCADA system) already integrated in the distribution grids of the projects to monitor the grids in real-time. By using these measurements, and integrating the grid topology, the tool is able to produce forecasts that help DSOs to detect potential issues in their networks, such as network congestions and voltage violations. With this information, sGRID is connected to sMART, the Local Market platform, (see more details in [3]) allowing DSOs to send flexibility requirements so the flexibility assets are able to negotiate the provision of flexibility services in order to tackle the issue. sGRID has been deployed in the Slovenian, Spanish, and Italian pilot sites.

Each pilot site has implemented their own version of the tools. The details of the functionalities deployed in the pilot sites are provided in sections 3.3, 3.4 (sGRID and sPLAN in the Slovenian pilot site respectively), 4.3 (sGRID in the Spanish pilot site) and 5.3 (sGRID in the Italian pilot site).

2.2 DATA ANALYTICS INTRODUCTION

Data Analytics is crucial in the development of innovative solutions for the management and optimization of complex systems. In the context of the STREAM project, the complexity of the ecosystem that is being built across the four pilot sites entails the collection of a huge amount of data. This raw data requires to be transformed into useful information, that will subsequently serve as inputs for the processes that are required in the flexibility markets and ancillary services as described in WP4 deliverables. To achieve this, the tools that have been, or are being, deployed at the different pilot sites incorporate a series of data analytics models designed to process data coming from both the assets of flexibility providers and network operators.

Within the *Data analytics* section of each pilot site, the models implemented in each service deployed at the pilot site are detailed. In these subsections, the pilot sites identify the data analytics models utilized, along with the tool(s) where these models are integrated. After this identification, a subsection is dedicated to each data analytics model, providing the background of the model and the identification of the problem the model is addressing, the raw input data required for the model, and the description of the data processing and analysis. Typically, a data analytics process encompasses several steps that will be covered in each model, including:

- **Data ingestion**: Collection of the input data identified in the previous subsection, identifying the source of the data (API, database, sDATA, etc.).
- **Data cleaning**: Identification and removal of abnormal values in the collected dataset, including errors in the format, duplicated parameters, and values out of defined boundaries.
- **Data exploration**: Analysis of the resulting cleaned dataset to identify patterns and relationships. This step helps choosing the more suitable type of model.
- **Feature extraction**: Creation and selection of additional characteristics that are identified as relevant to enhance the performance of the model.
- **Data processing**: Preparation of the data for the analysis, splitting the data into training and test sets.
- **Selection of model**: Once the data is cleaned, explored and processed, the more suitable model for the specific problem is selected, considering different algorithms and techniques. This step can be an iterative process if, after the validation step, the model is not performing adequately. The steps included in each iteration are:
    - **Model training**: Training of the selected model using the training set.

- **Model validation and tuning**: Evaluation of the performance of the model with the outputs obtained during the training phase using the test set. At this step, the developer decides whether the results are appropriate and can tune the model's parameters or test a different model until the optimal model is found.

- **Final model evaluation**: Evaluation of the final model using performance metrics. A wide variety of metrics can be used, depending on the type of model utilized.
- **Obtained outputs**: description of the datasets obtained as a result of the data analytics process.

It is important to note that, since most of the tools deployed in this project are already at a mature level, some of these steps may not be necessary and can be omitted for certain models.

2.3 USER PROFILING

Within the framework of the STREAM project, project partners have strategically acquired a multitude of flexibility assets situated across the four designated pilot sites (ES, FI, IT, and SI). The purpose of these acquisitions is to showcase diverse ancillary services that serve the needs of both Transmission System Operators (TSO) and DSO. To harness the full potential of these flexibility assets for TSO/DSO ancillary services within each pilot site, a thorough analysis involving baseline and flexibility forecasting is imperative. This analysis considers several crucial factors such as the asset's business-as-usual operation, security and technical constraints, availability timeframes, market opportunities, and other considerations.

As part of the project's scope, the ST3.4.2 user profiling process is dedicated to creating a comprehensive inventory of flexibility assets. This involves the calculation of baseline and flexibility forecasts, allowing for a nuanced understanding of the range of flexibility services that can be effectively executed. The objective is to determine the optimal utilization of these assets in alignment with the broader goals of the STREAM project.

2.3.1 Baseline forecast methodology

Conducting a baseline forecast for flexibility assets involves anticipating future electricity demand or production by leveraging historical data, related variables such as weather data and a suitable predictive model. While formulating a baseline forecast for a specific flexibility asset involves a customized approach, the following generalized steps are proposed:

1. **Data collection**:

The process is initiated by collecting historical measurements, typically represented in a time series format. The chosen dataset should span a significant period, if available at least a year, encompassing all seasons, holidays, and distinctive patterns in electricity consumption or generation.

2. **Data processing**:

In the second step, missing measurements are addressed to ensure data consistency. Furthermore, the dataset is analysed to determine patterns, trends, outliers, and influential factors such as weather conditions, holidays, or periods of low electricity prices.

3. **Baseline forecasting model**:

Based on the collected data and the identified patterns, trends, and outliers, the forecasting model (e.g., transformer neural network), that aligns with the specific requirements of forecasting the targeted flexibility asset, is chosen.

4. **Model training**:

During the model training phase, the selected forecasting model is trained using the training dataset and fine-tuned to achieve the optimal fit.

5. **Model evaluation**:

Once the forecasting model is trained, the performance of the model is evaluated in predicting future consumption or generation patterns. This evaluation ensures the model's accuracy and reliability in practical applications.

These structured steps provide a framework for developing a baseline forecast, offering a systematic approach to understanding and predicting electricity demand or generation for flexibility assets. The adaptability of these steps allows for tailoring the process to the unique characteristics and requirements of each flexibility asset under consideration. In user profiling, different pilots may follow all the steps above, select only certain steps, or even introduce additional ones.

2.3.2 Flexibility forecast methodology

Building on the baseline forecast, the crucial part of participating in the ancillary services market is to assess the available flexibility forecast of individual energy assets. The asset owner typically determines the availability of flexibility based on operational, technical, security, and economic constraints. However, the optimal flexibility forecast can also be influenced by emerging market opportunities, which might encourage asset owners to participate despite potentially exceeding, for example, operational limits. To forecast flexibility effectively, the following series of steps is proposed:

1. **Defining asset owner's parameters**:

A comprehensive workshop with the asset owner is conducted to get insights into economic conditions, operational and technical specifications, and security considerations when managing the flexibility asset. Establish a consistent communication channel between the asset owner and the aggregator to promptly exchange information on unforeseen events and their potential impact on the flexibility forecast.

2. **Data analysis**:

Utilize the information obtained from the asset owner to conduct an in-depth analysis of specific patterns. Identify time intervals during which the flexibility asset can contribute to either TSO or DSO. Generate a training set that reflects the asset's flexibility potential when specific patterns are recognized, facilitating the training set for the forecasting model.

3. **Generate flexibility availability**:

Based on the defined parameters from the asset owner and data analysis, the flexibility forecast of individual flexibility assets is generated. The available capacity or power can be offered in the various markets based on the system operator's needs.

The integration of baseline and flexibility forecasts constitutes a comprehensive user profiling process, which will be performed for different flexibility assets across the four designated pilot sites within the scope of this report. This holistic approach ensures a thorough understanding of the flexibility landscape and sets the stage for effective utilization in real-world scenarios.

3 SLOVENIAN PILOT SITE

3.1 DATA ANALYTICS

In SI pilot the STREAM tools will utilize various data analytics models, which are outlined in Table 1.

Table 1: Data analytics models in the Slovenian pilot site.

Data analytics model	Involved tool(s)
Time series forecasting	sGRID, sFLEX, sPLAN
Grid upgrade optimization	sPLAN
Power flow analysis	sGRID, sPLAN

3.1.1 Time series forecasting

3.1.1.1 Introduction

Time series data is one of the most widely used types of data, particularly in power systems. This data is collected through various devices such as smart meters and other metering equipment installed at network elements and is used by a range of power system actors, including system operators (both TSO and DSO), aggregators, consumers, and more.

In the Slovenian pilot, the emphasis is on the DSO’s perspective, with the sPLAN and sGRID tools addressing critical requirements for grid planning and monitoring. Both tools rely on time series data for power flow calculations, although their approaches to gathering and utilizing this data differ.

sPLAN is a grid planning tool designed to support the DSO's decision-making process by providing critical information needed for grid planning. It also incorporates the flexibility potential of emerging technologies such as EVs, PVs, and HPs. Its time series forecasting module generates typical consumer profiles for each season, accounting for annual load growth and newly added technologies in the grid.

Meanwhile, sGRID focuses on grid monitoring and management. It leverages time series data to forecast grid conditions for the day ahead, which serves as the basis for power flow simulations. These simulations assess grid stability and identify any potential issues, enabling proactive management and operational optimization.

In sFLEX time series forecasting model will be used to forecast the baseline operation of the flexible units included in the KOL aggregators portfolio. These models also help define the available flexibility schedules for each asset, supporting efficient participation in flexibility markets. This time series forecasting model is detailed in Section 2.3.

3.1.1.2 Input datasets

The datasets that are required as inputs for the model are specified in Table 2:

Table 2: Input datasets for time series forecasting in SI pilot.

Data element	Units	Tool(s)	Comments
Active and reactive power (consumer smart meters)	kW, kVAr	sGRID	Historical data (~2 years)
Active and reactive power (control measurements from MV/LV transformers)	kW, kVAr	sGRID	Historical data (~2 years)
Weather forecast data (temperature and global irradiance)	°C, W/m^2	sGRID	Gathered from external weather data provider
Active power of EV chargers	kW	sFLEX	Historical data for training
Active power of pilot site flexibility assets	kW	sFLEX	

3.1.1.3 Data processing and analysis

The methodologies applied in sGRID and sPLAN differ substantially, reflecting their specific objectives.

sGRID

sGRID uses XGBoost [4] to generate advanced load forecasts for all consumer loads in the grid. The forecasting models are trained using historical time series data and leverage both time-based and weather-based features, such as:

Time-based features

Time-based features help the model recognize recurring temporal patterns within the load data, significantly improving prediction accuracy. The primary time-based features employed are:

- **Time of day**: captures daily load variations
- **Day of week**: represents weekly load patterns, account for variations between different days of the week and weekend
    - **Type of day**: binary indicator distinguishing between work or non-work days (weekends, holidays)
- **Month of year**: accounts for seasonal variability and yearly cycles of load patterns

Weather-based features

Weather conditions directly influence power consumption patterns. sGRID incorporates the following key weather parameters:

- **Temperature** (°C): very strongly connected with heating and cooling loads
- **Global irradiation** ($^{W}/_{m^2}$): key for forecasting generation from PV installations

Lagged features (short-term forecasts)

For shorter-term forecasts (1-hour and 15-minute horizons), sGRID includes lagged load features, which reflect recent historical data:

- **1-hour lag**: load measurements from one hour prior, capturing broader short-term trends

- **15-minute lag**: load measurements from the previous interval (15 minutes before)

Once trained, these models and their associated artifacts (e.g., scalers) are stored in a database using mlflow [5]. Each time sGRID runs, it retrieves the necessary model parameters from the database to generate new forecasts.

sPLAN

In contrast, sPLAN does not rely on complex forecasting techniques for generating input load profiles. Instead, it creates typical seasonal profiles for each consumer load, which are stored in a database for one-week periods. These profiles are calculated from historical smart meter and control measurements, averaging the weekly load patterns for each season. When new technologies (such as PVs) are added to a consumer load, the corresponding load profiles are modified to incorporate their impact (for example, by including PV production). The adjusted profiles are then used as inputs for the load flow calculations.

The outputs of the time series forecasting are presented in table below.

Table 3: Output datasets for time series forecasting in SI pilot.

Data element	Units	Tool(s)	Comments
Forecasted active and reactive power of grid loads	kW, kVAr	sGRID, sPLAN	
Forecasted charging power of EV charging stations	kW	sFLEX	
Forecasted power of flexibility assets	kW	sFLEX	

3.1.2 Grid upgrade optimization

3.1.2.1 Introduction

Grid upgrade optimization is used within the sPLAN tool, which is a grid planning tool targeted to the DSO. The tool calculates the state of the grid based on inputted parameters (number new devices such as PVs, HPs, EVs and BESS, as well as the expected yearly load increase) and checks for limit violations (congestion or voltage violation) in any grid elements. In case an issue is detected, the tool is able to calculate the optimal grid upgrade, which will resolve the issue with the lowest costs.

3.1.2.2 Input datasets

Table 4: Input datasets for grid upgrade optimization in SI pilot.

Data element	Units	Tool(s)	Comments
Grid upgrade costs (per type of grid element)	€, €/km	sPLAN	Used approximate costs from EPR.
Flexibility price	€	sPLAN	
Calculated grid state	kW, kVAr, A, V	sPLAN	Results from power flow analysis performed in the tool.

3.1.2.3 Data processing and analysis

The grid upgrade is optimised as follows. First, the part of the network and elements that are overloaded according to the initial simulation results are identified. Once these elements have been identified, the type of line or transformer used is defined. These are then in the network model replaced by the element type from the library which, regarding the technical parameters, represents the first step of the upgrade. Once the replacement has been done, the simulation is re-run and if the results are sufficient, the investment cost of the upgrade is determined. If the results still indicate that the element is overloaded, the steps shall be repeated until a normal grid operation is reached. In the latter case, the grid upgrade costs are defined only for the final step.

The alternative solution is to improve the network conditions using flexibility. In this case, the power profiles of users with flexibility resources are modified (reduced) and the impact on the network is analysed. In the case of flexibility usage, the price of flexibility is also an important factor that is also considered.

The output datasets of the Grid upgrade optimisation are presented in the table below.

Table 5: Output datasets for grid upgrade optimization in SI pilot.

Data element	Units	Tool(s)	Comments
Optimal location for grid reinforcement	/	sPLAN	
Total reinforcement cost	€	sPLAN	

3.1.3 Power flow analysis

3.1.3.1 Introduction

Power flow analysis is a method used to determine the steady-state operating conditions of an electrical grid. It calculates key electrical parameters such as voltages, currents, power flows, and losses throughout the network.

It is a key functionality of both the sPLAN and sGRID tools, supporting grid planning and grid operational management respectively.

In the sPLAN tool, power flow analysis supports the evaluation of long-term planning scenarios. The inputs include typical seasonal load profiles and expected integration of new grid technologies, such as EVs, PVs, HPs, and BESS. These inputs are used to generate adjusted load profiles reflecting the anticipated future state of the grid. The resulting load profiles are subsequently processed through a power flow simulation, enabling the identification of potential limit violations, such as congestion or voltage issues. If violations are detected, the analysis helps pinpoint areas requiring grid reinforcement or flexibility interventions.

In contrast, sGRID employs power flow analysis for short-term operational management. It uses forecasted load profiles as inputs to the power flow model, where simulation results are evaluated against predefined planning and operational limits. Grid elements approaching or exceeding these limits signal potential congestion. When such violations occur, sGRID determines the required flexibility to alleviate congestion and communicates this information to sMART, enabling proactive measures to maintain grid stability.

Both sPLAN and sGRID utilize the PandaPower library [6], an open-source Python library used for power system analyses, including detailed load flow calculations, short-circuit analyses, and optimal power flow studies.

3.1.3.2 Input datasets

The input parameters for the power flow analysis within sPLAN and sGRID are detailed in the table below. Although both tools use consumer load profiles as inputs, they differ in the method of profile creation.

Table 6: Input datasets for power flow analysis in SI pilot.

Data element	Units	Tool(s)	Comments
Active and reactive power consumer profiles (forecasted or generated)	kW, kVAr	sGRID, sPLAN	Forecasted for sGRID and generated for sPLAN.
Active and reactive power (control measurements from MV/LV transformers)	kW, kVAr	sGRID	Mainly for validation purposes.

3.1.3.3 Data processing and analysis

The output datasets from the power flow analysis include several grid parameters, categorized as follows:

- **Bus/node results**
  - Voltage magnitude (V or p.u.)
  - Voltage phase angle (degrees)
- **Branch** (lines and transformers) **results**
  - Active power flow (kW)
  - Reactive power flow (kVAr)
  - Loading factor (% of operational limit)
- **Slack bus and external grid results**
  - Slack bus generation (kW, kVAr)
- **Grid losses**
  - Active losses (kW)
  - Reactive losses (kVAr)

While both sPLAN and sGRID tools produce these results, certain parameters hold greater importance depending on the tool's specific purpose. The loading factor, which indicates potential congestion, is key for both tools. In contrast, voltage magnitude results are more important for sPLAN, given its focus on long-term grid reinforcement planning and voltage stability. sGRID is primarily focused on congestion management, relying less on voltage magnitude as it interacts with sMART to address congestions in the grid. The output datasets resulting from this model are listed in Table 7.

Table 7: Output datasets for power flow analysis in SI pilot.

Data element	Units	Tool(s)	Comments
Line loading	%	sGRID, sPLAN	Identifies congestion on lines.
Transformer loading	%	sGRID, sPLAN	Identifies congestion on transformers.
Node voltages	p.u. (can be transformed to V)	sPLAN	

3.2 USER PROFILING

3.2.1 Flexibility Assets in the STREAM environment

The Slovenian pilot site is situated within an industrial park in the town of Ajdovščina, located in the southwestern region of Slovenia. This location was selected before the official launch of the STREAM project due to its suitable power consumption characteristics. Preliminary analysis revealed a peak power consumption of approximately 9.8 MW in this area, with an estimated 20 % of this capacity available for ancillary flexibility services to DSO or TSO.

Through the classification of data obtained from metering points within the Ajdovščina industrial park, project partners identified thirteen potential end-users. These entities were identified based on their potential contributions (e.g., offering consumption/production units for flexibility services) and potential benefits (e.g., creating monetization streams with their flexibility units) to the STREAM project. Following contact with these end-users and the presentation of the project details, letters of intent were signed by eleven companies even before the official project kick-off. This early commitment proves crucial for project continuity, simplifying interactions with these companies as a point of contact has already been established, and they are aware of their future involvement in the project.

Once the project commenced, all end-users with signed letters of intent were approached and presented with various flexibility service opportunities. Table 8 provides detailed information on companies that chose to participate in the project actively, offering an overview of the electricity consumption for one or more of their flexibility devices for project demonstration purposes. All of the listed energy assets were or are in the process of integration with the Kolektor sETup aggregator platform (sFLEX) – the integration with the aggregator encompasses monitoring and managing the energy assets when possible. Some companies formalized contracts with the aggregator, allowing the aggregator to provide their energy assets in the already established TSO ancillary services market. The cumulative power of assets under contract surpassed the initial estimated 2 MW at the project's outset.

Table 8: Overview of the energy assets in the Slovenian pilot.

Asset	Company	Nominal power	Contractual agreement for mFRR ancillary services
Cold storage rooms	Incom Leone	1.500 MW	YES
Mill	Mlinotest	0.450 MW	YES
PV	Petrič	0.250 MW	NO
PV	Tosla	0.250 MW	NO
Cold storage rooms	Mlinotest	0.215 MW	YES
PV	Metal design	0.100 MW	NO
BESS	Tosla	0.08 MW/0.2 MWh	NO
E-charging stations	Tosla	0.044 MW	NO
E-charging stations	Avantcar	0.088 MW	NO

Table 8 highlights a substantial pool of flexibility potential within the industrial zone of Ajdovščina, showcasing a collective flexibility power of assets that surpasses 3.5 MW. The table encompasses diverse energy asset types, incorporating various consumption asset types (e.g., mill, e-charging stations), one representative of a production asset type (PV power plant), and representatives for energy storage type (cold storage rooms, BESS). This diverse mix lays the groundwork for multiple demonstrations of distinct flexibility services, both on the DSO and TSO levels. However, the initial phase of the analysis entails a thorough examination of the measurement data, with a primary focus on establishing a baseline and flexibility forecast. The principle of user profiling will be demonstrated in the Slovenian pilot using the cold storage rooms of Incom Leone. This is due to the availability of measurement data from the start of 2024, unlike other assets, which were integrated or are being integrated with the sFLEX platform at a later time. Profiling for the remaining flexibility assets of the Slovenian pilot site will be conducted in the scope of WP5 and presented at an aggregated level.

3.2.2 Baseline Forecast

In December 2023, during M16 of the STREAM project, Incom Leone's cold storage rooms were integrated with the sFLEX tool. Figure 2 illustrates the total energy consumption of this asset over the first five months of data collection. As shown, the maximum daily power usage in January and February is about 0.5 MW. This increases to approximately 0.6 MW in March and April. A significant rise is observed in May and June, with the maximum daily power reaching around 1.0 MW. This upward trend in daily power consumption closely correlates with rising ambient temperatures, indicating that higher temperatures necessitate greater cooling power. The increase in electricity consumption is not solely temperature-related; part of it comes from the operation of additional compressors installed during the early months of 2024. Furthermore, the minimal daily power consumption increase can be detected, from approximately 50 kW in winter months to almost 200 kW at the start of June.

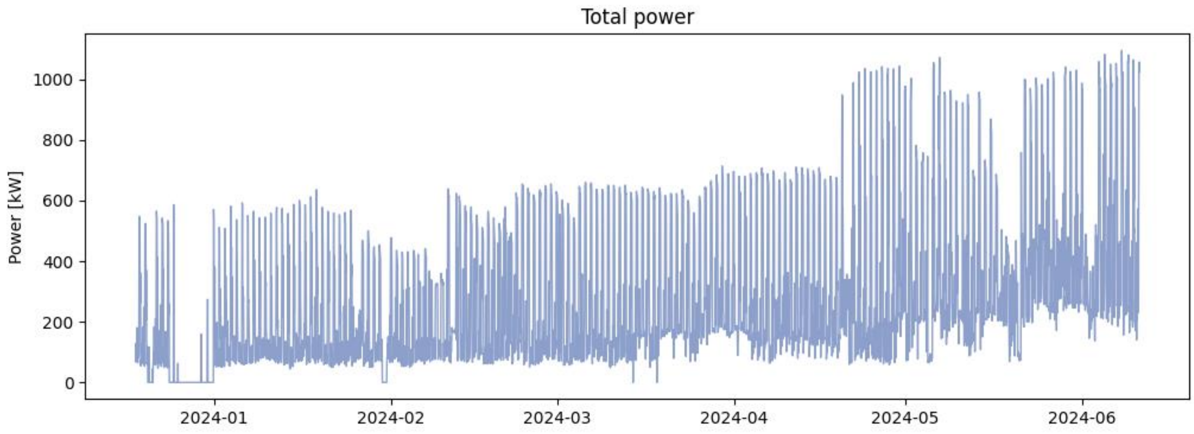


Figure 2: Total power of cold storage rooms of Incom Leone.

The collected data spans over three seasons: winter, spring, and the beginning of summer. Autumn is not included (yet), but the spring data suggests how the cold storage rooms might respond to decreasing ambient temperatures in autumn. Figure 3 shows the average daily profile for each weekday, analysing data from the first five months of 2024 (January 2024-01 to June 2024-06). The figure indicates minimal variation between weekdays. However, there is a notable difference between daytime and nighttime consumption, implying that Incom is leveraging low electricity tariffs from 22:00 to 06:00 and avoiding high electricity tariffs from 06:00 to 22:00 throughout the whole week.

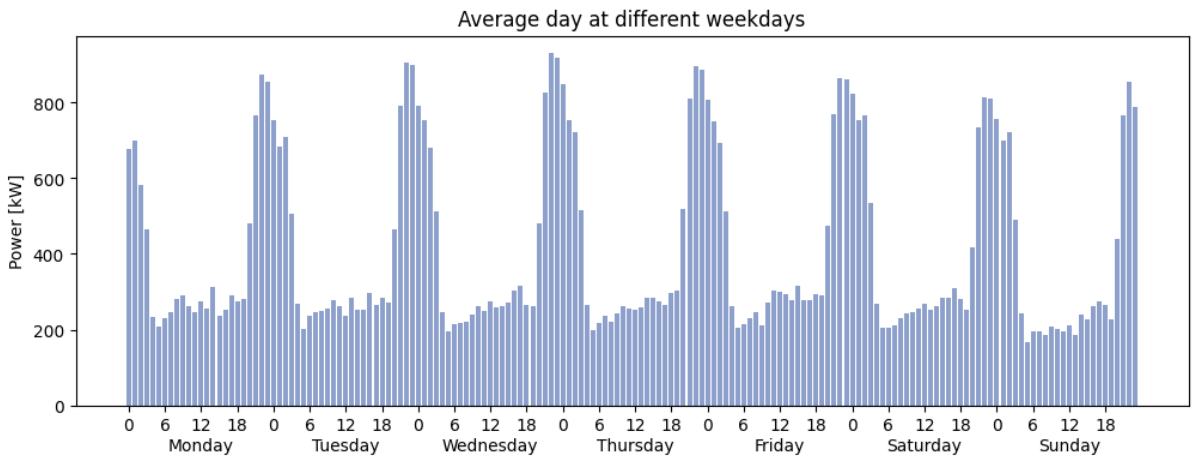


Figure 3: Average daily profile for each day in a week.

Additionally, we examined the differences between workdays and non-workdays to understand the consumption patterns when the company is inactive. Figure 4 reveals a slight decrease in consumption on non-workdays. This is likely because the measurements are collected from transformer feeders supplying cold storage rooms and a small part of the company building. As the described figure suggests, the consumption of this part of the building is minimal compared to that of the cold storage rooms.

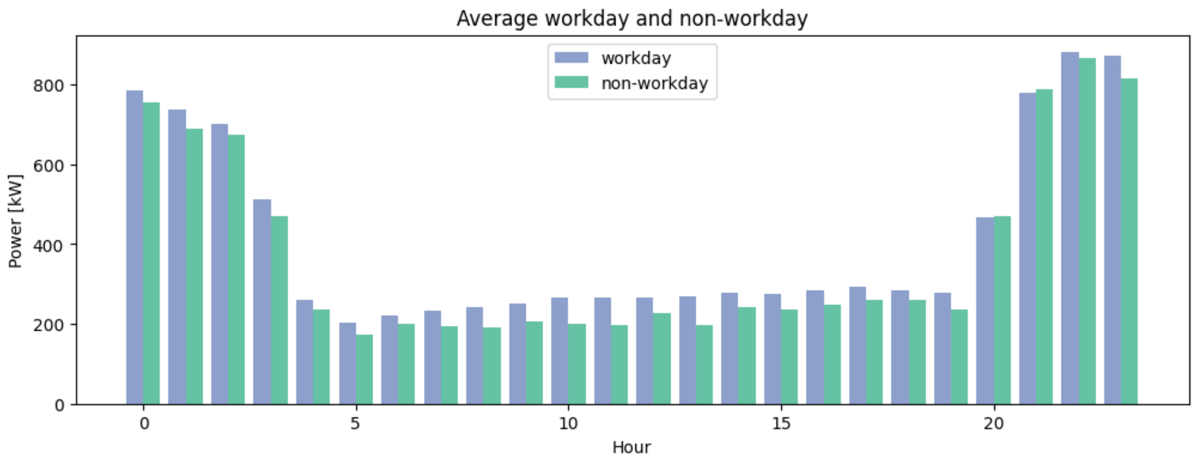


Figure 4: Average daily profile of a work and a non-workday.

In terms of data processing, no significant effort was necessary, as the measurements remained extremely stable after the initial few weeks (after the integration period). There were no data fallouts, eliminating the need to replace missing values with NaN. Additionally, the collected measurements were realistic and consistent, so there was no need to exclude any outliers that might have been excessively high or low.

For forecasting, we experimented with several algorithms, including:

- **Random Forest Regressor (RFG)** [7]: Available in scikit-learn library.
- **Light GBM (LGBM)** [8]: A framework developed by Microsoft and available as a Python library.
- **Histogram-based Gradient Boosting Regression Tree (HBGBRT)** [9]: The equivalent of the Light GBM from the scikit-learn library.

Based on our findings, the HBGBRT or LGBM (equivalents in different libraries) demonstrated the best overall performance in terms of consistency, speed, and accuracy. Here is a detailed breakdown of the HGBRT algorithm:

- The base learners in this algorithm are decision trees. Decision trees partition the feature space into regions and make predictions for each region.
- It employs a technique of ensemble learning, where multiple models are combined to solve a particular problem. Instead of relying on a single model's predictions, ensemble methods leverage the collection of models to create more accurate predictions.
- The algorithm uses gradient boosting to ensemble multiple decision trees. The trees are built sequentially, with each tree attempting to correct the errors made by the previous ones. In practice, this means that the models can be slow to train, unlike for example random forest regression where trees can be trained in parallel.
- Training is accelerated by reducing the number of values for continuous input features achieved by binning. This reduces the number of unique values and speeds up decision tree construction. Histogram-based algorithms further improve efficiency by binning feature values and constructing histograms during training.
- Gradient optimization refers to the optimization of the parameters of the decision trees to minimize a loss function. It does this by iteratively fitting new trees to the negative gradient of the loss function.

To prepare the data for the forecast model, we selected the following features: seasonal information such as month, weekday, hour, and workday/non-workday indicators. Additionally, we included temperature and precipitation data sourced from the Meteo ARSO [10] database. The data was then split into training and testing datasets. The metric used to evaluate accuracy was the normalized mean absolute error (wMAPE):

$$\text{wMAPE} = \frac{\sum_i |A_i - F_i|}{\sum_i A_i}$$

where A_i is the actual value and F_i is the forecasted value in each time interval.

During model training, we recorded the accuracies achieved with each algorithm. We trained models using three different algorithms: RFG, LGBM, and HBGBRT. The accuracies recorded for each model, based on the selected metric, can be seen in Table 9:

Table 9: Comparing metrics on train dataset.

Algorithm	HBGBRT	RFG	LGBM
wMAPE [%]	16.3	16.3	16.4

Figure 5 provides an example of the fitted data while training the LGBM model.

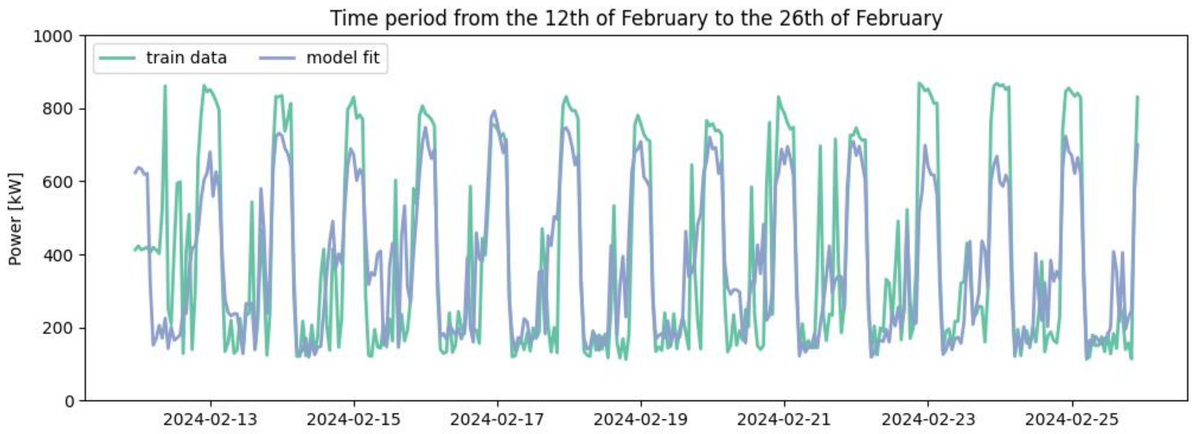


Figure 5: Achieved accuracy during the training of the LGBM model.

Besides using the wMAPE as the selected metric for the accuracy of the model, we can also evaluate the cost of deviation between the forecasted and the actual consumption. This can be written as:

$$c_1(t) = E_{\text{pred}}(t) \cdot p_{\text{DA}}(t) + \left(E_{\text{actual}}(t) - E_{\text{pred}}(t)\right) \cdot p_{\text{imb}}(t),$$

$$c_2(t) = E_{\text{actual}}(t) \cdot p_{\text{DA}}(t),$$

where the total supplier's cost is $C_1 = \int c_1(t)\ \mathrm{d}t$ while the apparent user's cost is $C_2 = \int c_2(t)\ \mathrm{d}t$, which in our case equals the summation of all $c_1(t)$, $c_2(t)$; calculated for each data point. E_{pred} is the prediction of energy consumption, E_{actual} is the actual realization, p_{DA} is the electricity price on the day-ahead (DA) market and p_{imb} is the price for imbalances. We calculate the difference between the supplier's and apparent user's cost $\Delta = C_1 - C_2$, so we can output two ratios:

$$\sigma_C = \frac{\Delta}{C_1}, \qquad \sigma_E = \frac{\Delta}{\sum_i E^i_{\text{actual}}}$$

σ_C is the relative difference in percent, $\sigma_E$is the surcharge on the electricity price based on the inaccuracy of the forecasting and i represents data points. We calculated the proposed costs, evaluated the accuracies of chosen models on the test dataset and compared the results in Table 10:

Table 10: Comparing metrics and cost of deviation on test dataset.

Algorithm	HBGBRT	RFG	LGBM
wMAPE [%]	10.6	11.8	11.2
Consumer's cost [EUR]	4095		
Supplier's cost [EUR]	4576	4539	4561
Cost diff [EUR]	481	444	466
Relative diff [%]	11.8	10.8	11.4
Relative diff [EUR/MWh]	6.6	6.1	6.4

Table 10 presents the accuracy results of the proposed models on the test dataset, showing that all three models perform similarly. The HBGBRT achieves the lowest wMAPE with 10.6 %, while the RFG has the highest with 11.2 %, and LGBM is in the middle with 11.2 %. Although the RFG has the highest wMAPE, it has the smallest cost difference and therefore the lowest surcharge on the electricity price. According to the table, the electricity supply surcharge for the cold storage rooms would be between 6.1 €/MWh and 6.6 €/MWh. Despite not having the best accuracy, we chose to proceed with the LGBM model due to its optimal balance of accuracy, speed (being integrated into Python), and consistency. Figure 6 illustrates an example of the achieved accuracy on the test dataset.

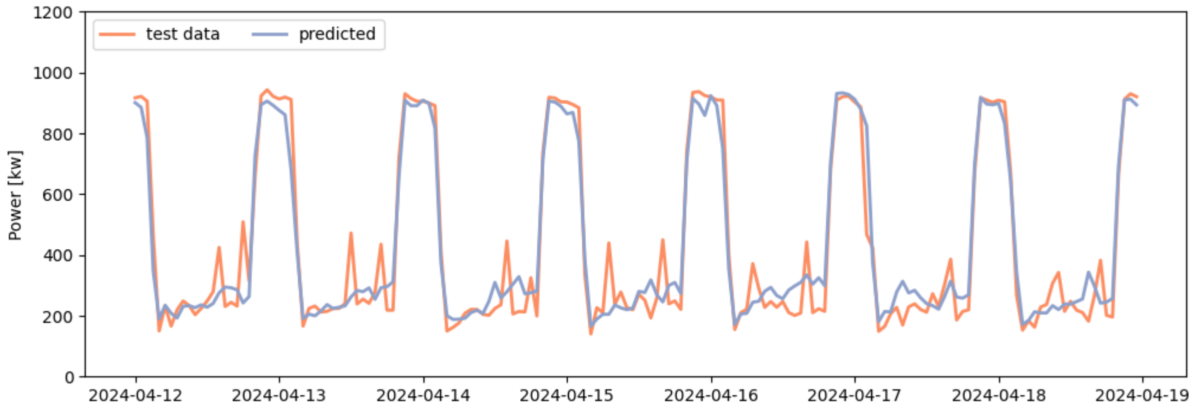


Figure 6: Achieved accuracy on the test dataset.

3.2.3 Flexibility Forecast

KOL held several meetings with Incom Leone to identify operational, technical, security, and market limitations and considerations. Based on input from Incom Leone, we can summarize the key constraints as follows:

- **Technical Limitation**: The ramp-up time for five compressors, with a combined power of 1.5 MW, is approximately 30 minutes. This is only partially compatible with opportunities in the DA, the Intraday (ID), and the ancillary services market, specifically the manual Frequency Restoration Reserve (mFRR). The proportion of response of the five compressors in 15 minutes, the response time for mFRR, will need to be tested.
- **Technical Limitation**: The step-down time for five compressors is approximately 20 minutes. This is only partially compatible with opportunities in the DA, the Intraday (ID), and the

ancillary services market, specifically the mFRR. The proportion of response of the five compressors in 15 minutes, the response time for mFRR, will need to be tested.

- **Technical Limitation**: The operation of cold storage rooms can be stopped at night if the opportunity in the market arises. The power can be decreased to approximately 0 MW.
- **Technical Limitation**: The operation of cold storage rooms can be increased during the day, but only up to the maximum power used during the previous night's operation, as this maximum power is constrained by ambient temperature limits.
- **Operational/Security Limitation**: The operation of the compressors in Incom Leone's cold storage rooms allows for stoppages of a few hours during the day without significant issues due to sufficient system inertia. This does not hinder mFRR, DA, or ID market participation.
- **Market Limitation**: Incom Leone established the marginal price for both mFRR- and mFRR+, which was effectively integrated into a contractual agreement with Kolektro sETUp.

Since STREAM mostly concentrates on providing flexibility to system operators, Incom Leone and Kolektor sETup provided an analysis of the suitability of cold storage rooms for ancillary services. We analysed the available daily capacity and available energy on an hourly basis.

Although the analysis mainly targeted mFRR services, the findings could also be applied to the local flexibility market for DSOs developed in STREAM. Positive reserve refers to a decrease in consumption, while negative refers to an increase in consumption. To make a rough estimation of the potential availability in the mFRR we took the actual and calculated the daily maximum and minimum values. Theoretically, the range from the measured maximum daily power to asset's apparent power $AP = 1.5\ MW$ is the negative energy band available for the negative balancing reserve. Similarly, the range from 0 to the daily minimum power forms the positive band, that can be used for the positive balancing reserve. Assuming a business-as-usual price of $p = 3\ \mathrm{EUR/MWh}$ per 1 MWh block, success rate of 80%, $sr = 0.8$ for our bids and daily hours $dh = 24\ h$, the following profit for mFRR capacity can be calculated for the observed period:

$$\mathrm{C}_{+} = \sum_{i}^{N_{\mathrm{days}}} \min\left(P_{\mathrm{actual}}^{i}\right) * p * sr * dh = 1210.15\ \mathrm{EUR}$$

$$\mathrm{C}_{-} = \sum_{i}^{N_{\mathrm{days}}} \left(AP - \max\left(P_{\mathrm{actual}}^{i}\right)\right) * p * sr * dh = 5875.15\ \mathrm{EUR}$$

The theoretical profit for the negative capacity reserve is significantly higher than for the positive capacity reserve. This is evident in Figure 7, where the maximum available daily power (shown in blue) exceeds that of the positive reserve (shown in orange). However, this profit remains theoretical, as the mFRR capacity bidding requires a minimum block of 1 MWh to be available throughout the entire day, which is not achievable with the current consumption levels. Based on the figure, cold storage rooms could potentially participate in the mFRR capacity market when aggregated with other flexibility units, therefore limiting the theoretical profit.

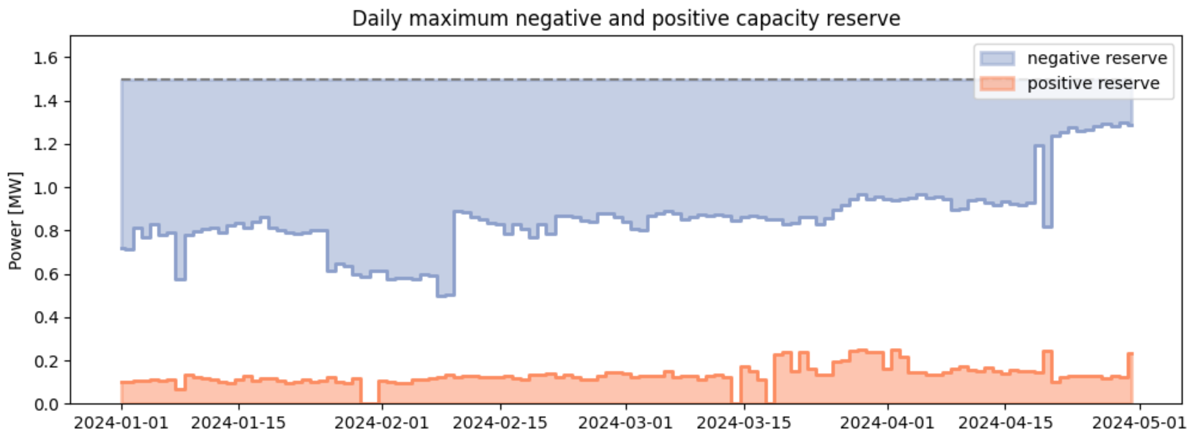


Figure 7: Negative and positive capacity reserve.

Furthermore, we utilized the actual realized mFRR capacity prices during the observed period, given the increasingly constrained market conditions due to a growing number of participants, especially due to the surplus of offered power in the negative capacity market. The average price during the observed period was $p_{-} = 0.71\ \mathrm{EUR/MWh}$ for the negative capacity reserve and $p_{+} = 3.98\ \mathrm{EUR/MWh}$ for the positive reserve. The profit in the case of the actual daily prices is:

$$\mathrm{C}_{+} = 973.77\ \mathrm{EUR}$$

$$\mathrm{C}_{-} = 779.30\ \mathrm{EUR}$$

When utilizing the realized capacity reserve prices during the observed period, the potential profit for negative capacity decreases substantially and is even lower than the one for positive capacity reserve even though the cold storage rooms could offer more power in the negative capacity market. The results suggest that cold storage rooms of Incom Leone could be utilised in the capacity market, but it would need to be done in combination with other assets to build 1 MWh blocks. Based on the current prices the maximal profit is approximately 2,000.00 EUR per year considering the 80 % success in bidding.

In the energy market, Incom Leone could offer approximately 0.6 MWh in the negative direction (based on the constraints) during daytime hours, and 0.8 MWh in the positive direction during nighttime, Figure 8. Since the minimum block in the energy mFRR market is 1 MWh, Incom Leone would need to combine bids with other assets during the night and day to participate in the market. Based on the previous calculation, the energy market appears favourable in comparison to the capacity market. For instance, considering 10 hours of activation, with a price of up to 3,500.00 EUR per MWh and offering of:

- 0.6 MWh during the day, the asset could potentially earn up to 21,000.00 EUR annually in the negative energy market.
- 0.8 MWh during the night, the asset could potentially earn up to 28,000.00 EUR annually in the positive energy market.

The calculated values are theoretical because the power offered in the market is typically lower than the maximum available to ensure reliability. Additionally, the asset's bid must be accepted to achieve the estimated hours of annual activation, and higher offered prices reduce the chances of acceptance. Furthermore, successful activation is necessary to avoid potential penalties.

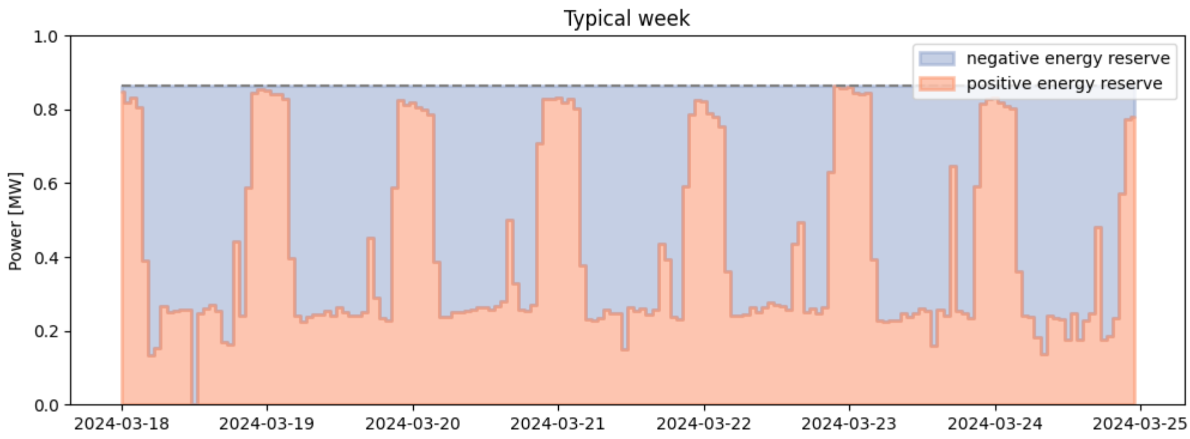


Figure 8: Negative and positive balancing energy reserve.

3.3 SGRID IN THE SLOVENIAN PILOT SITE

3.3.1 Overview and architecture

sGRID is a tool designed to support DSOs by providing three key functionalities:

- Day-ahead forecasting of the grid state by utilizing advanced forecasting techniques.
- Identification of congestions ahead of time.
- Communication with sMART to mitigate identified congestions.

As shown in Figure 9, sGRID's architecture consists of three main components: the GUI, Database, and Engine. It also connects to external services and tools, including sMART, SCADA/AMI systems (via sDATA), and weather forecast services.

The database serves as the backbone of sGRID, holding data required for its operation, including historical grid measurements, grid topology, and predefined grid limits, as well as the forecasting models. It also stores results from previous runs, enabling DSOs to conduct analyses and enhance their decision-making processes.

The core functionality of sGRID lies within its engine, which is developed in Python and leverages a variety of specialized libraries to deliver precise results. After completing each run, the tool initiates communication with sMART, which subsequently opens auctions to procure flexibility and resolve the forecasted grid congestions effectively.

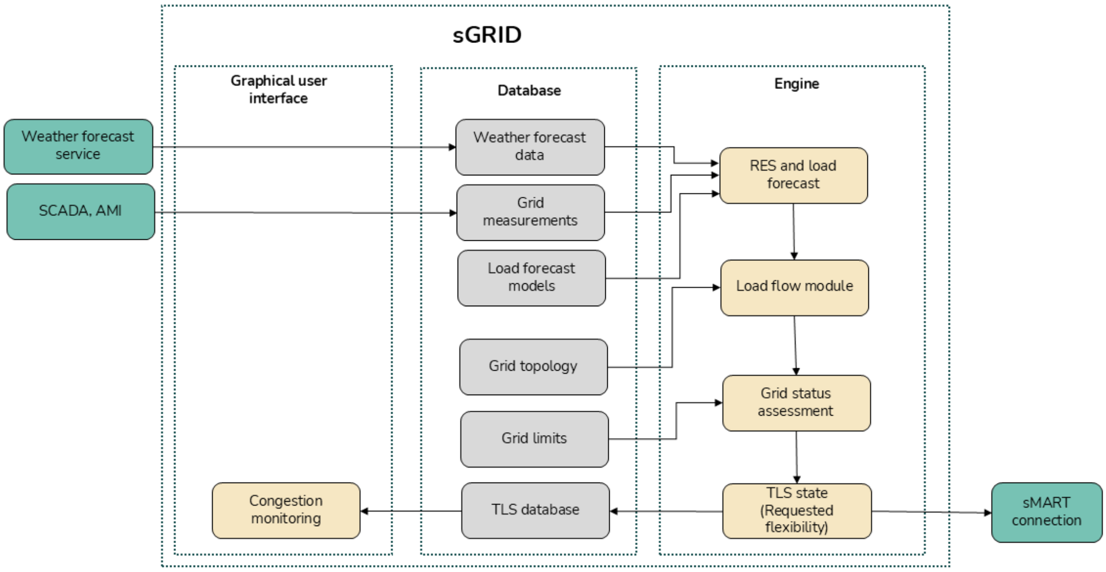


Figure 9: sGRID architecture (SI pilot site).

3.3.2 sGRID functionalities

3.3.2.1 Congestion forecasting

The primary functionality of sGRID is to accurately forecast grid conditions and detect potential congestions across multiple forecasting horizons: day-ahead, 1-hour ahead, and 15-minute ahead. To achieve this, sGRID utilizes advanced forecasting methods based on XGBoost models, previously trained and stored within the tool's database. These models incorporate historical load measurements, recent load data (from the previous hour or 15-minute intervals), weather conditions, and various time-based features, such as time-of-day, day-of-week, and type-of-day indicators.

- **Day-ahead forecasts:** leverage external weather forecast data combined with historical load patterns and time-based features to generate load predictions for the next day. These forecasts provide DSOs with early identification of potential congestion points, facilitating proactive grid management and operational planning.
- **Short-term forecasts (1-hour and 15-minute ahead):** unlike day-ahead forecasts, short-term predictions integrate recent historical load measurements (lagged features) to improve accuracy. Lag features comprise immediate historical load values from the previous intervals, allowing these forecasts to reflect rapid load changes more accurately. By updating these forecasts at regular intervals, sGRID continuously refines the predictions, closely aligning forecasts with actual real-time conditions. This approach enhances the precision of congestion detection.

All forecasted load profiles generated by sGRID serve as inputs for subsequent power flow analyses conducted using the PandaPower library in Python. The simulation combines these forecasted loads with the existing grid topology and operational parameters—both taken from the database—to calculate power flows and loading conditions throughout the grid. This allows the tool to identify the network elements experiencing congestion risks.

To demonstrate the impact of incorporating lagged features and shortening the forecasting horizon, we present results for two different cases: an aggregated load at the transformer level and a single consumer's load, both across all three forecasting horizons. The figures show a one-week period during the test phase, while the overall model performance is evaluated on the full test dataset.

Forecasting Performance for Aggregated Load (Transformer Level)

Day-ahead forecasting results (Figure 10): The predicted curve generally follows the actual load profile well but struggles to accurately capture peak values, which are critical for congestion detection. Additionally, some deviation is observed in forecasting valley points, though this is less critical for operational decision-making.

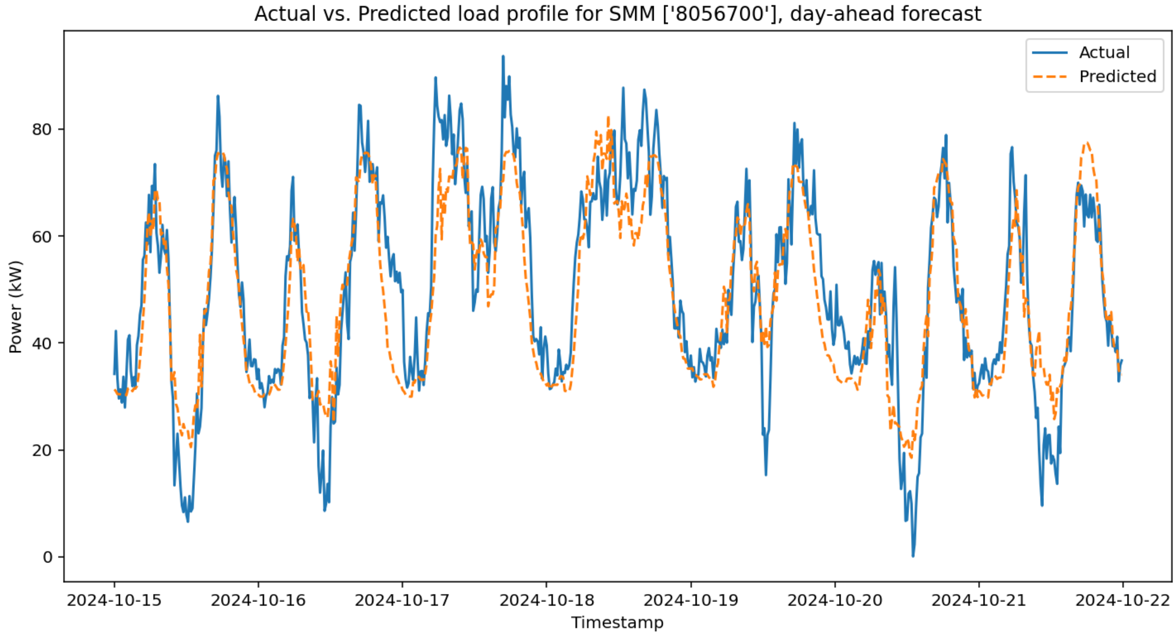


Figure 10: Forecasting model results evaluation, day-ahead forecast.

1-hour ahead forecasting results (Figure 11): With the integration of up-to-date measured values, the accuracy of the forecast improves. The predicted peak values are closer to the actual ones, though some discrepancies remain.

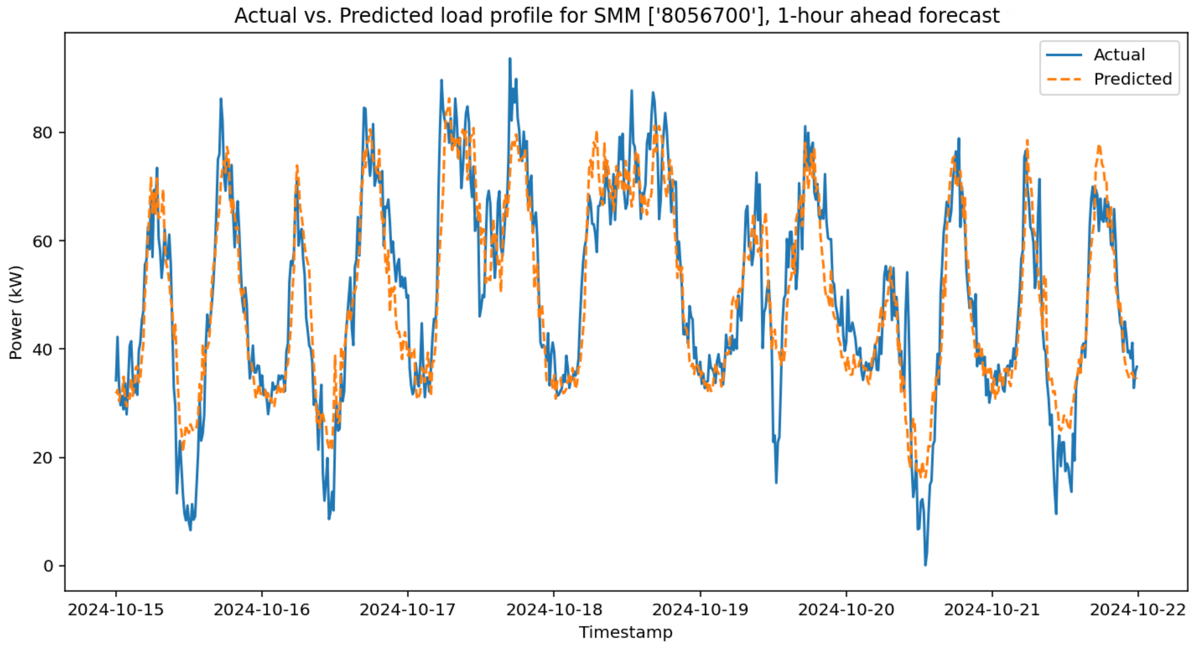


Figure 11: Forecasting model results evaluation, 1h ahead forecast.

15-minute ahead forecasting results (Figure 12): The forecast closest to real-time benefits from access to the latest measurement interval, allowing it to make highly accurate short-term predictions. The model consistently captures peak values with improved precision, reducing the error observed in the previous forecasting horizons.

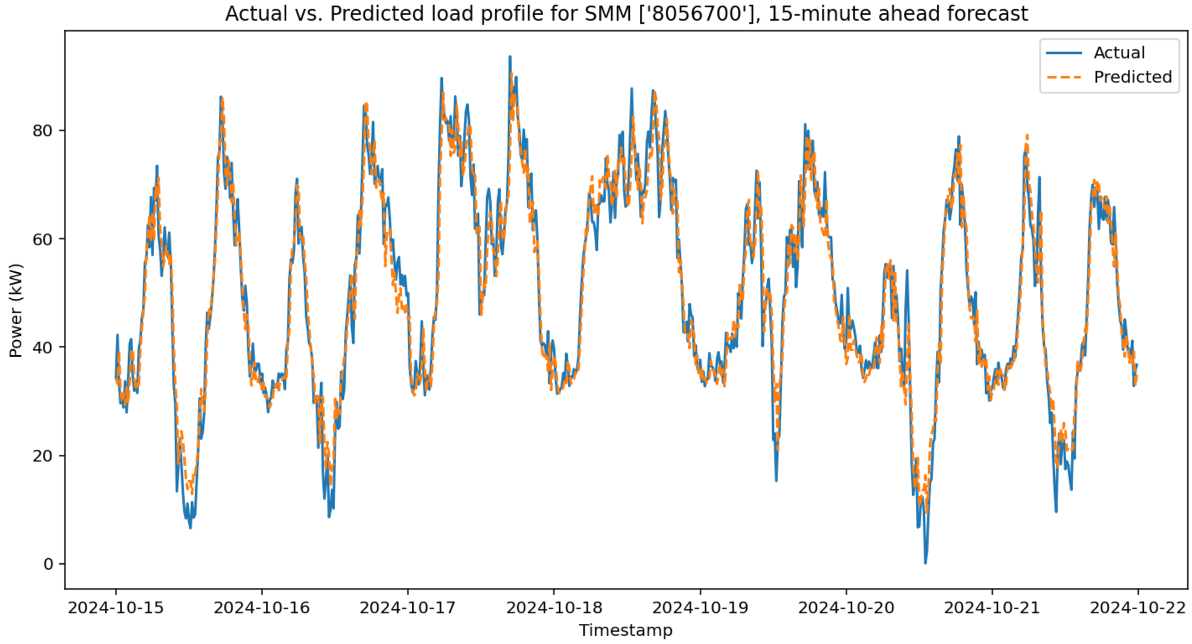


Figure 12: Forecasting model results evaluation, 15min ahead forecast.

The improvement in forecasting accuracy is also reflected in key performance indicators (KPIs) calculated over the test dataset (20% of total data). We used the following KPIs:

- **nMAPE** (normalized Mean Absolute Percentage Error), which provides a scale-independent measure of forecast accuracy.

$$nMAPE = \frac{100}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{\max(y) - \min(y)}$$

  Where y_i is the actual value, $\hat{y}_i$ is the predicted value and n is the number of observations.

- **MAE** (Mean Absolute Error) measures the average absolute difference between the predicted and actual values.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

- R^2 (coefficient of determination) indicates how well the model explains the variability of the target variable and is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

  Where $\bar{y}$ is the mean of the actual values.

Table 11 presents the comparative performance of the models:

Table 11: Forecast models performance comparison, aggregated load.

	Day-ahead	1-hour ahead	15-minute ahead
nMAPE (%)	15.7	12.14	8.68
MAE (kW)	7.58	5.85	4.18
R^2	0.74	0.84	0.91

From the KPIs, we observe a clear trend: as the forecasting horizon shortens, accuracy improves across all metrics. The nMAPE decreases significantly from 15.7% in the day-ahead model to 8.68% in the 15-minute ahead model, indicating a notable reduction in relative error. Similarly, the MAE drops from 7.58 kW in the day-ahead forecast to 4.18 kW in the 15-minute ahead forecast, demonstrating that the average deviation from actual values is almost halved.

Additionally, the R^2 improves from 0.74 for the day-ahead forecast to 0.91 for the 15-minute ahead forecast, indicating that the model's ability to explain variance in the data significantly increases with shorter horizons.

Forecasting Performance for Individual Consumer Load

While transformer-level load profiles are relatively stable, individual consumer loads are more unpredictable due to higher stochastic variations. To assess how forecasting accuracy changes for these smaller, more volatile loads, the second example focuses on one such consumer.

Day-Ahead Forecasting Performance (Figure 13): The figure presents the result of the day-ahead forecast model. The model effectively captures the daytime production patterns, particularly when generation dominates consumption. However, it struggles with the nighttime profile, where the load is more erratic. The model tends to predict an average load, which is sometimes accurate (e.g., the night between October 17 and 18) but is not able to capture the rapid changes present in other periods.

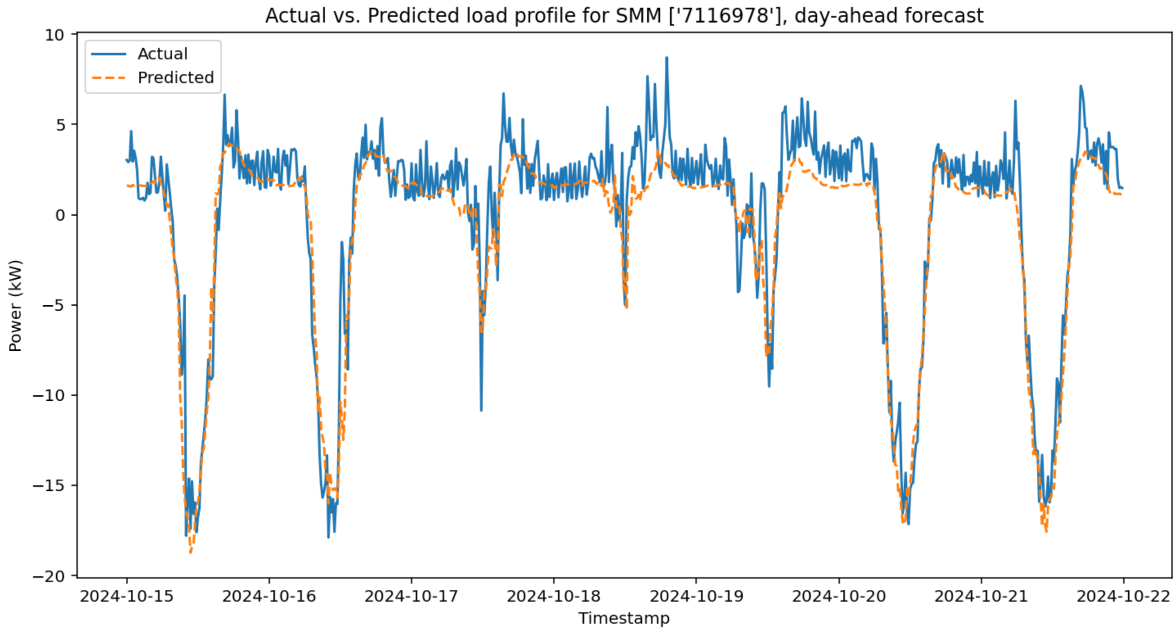


Figure 13: Forecasting model results evaluation (individual consumer level).

1-Hour Ahead Forecasting Performance (Figure 14): The figure shows the performance of the 1-hour ahead forecast model. Compared to the day-ahead forecast, this model better captures nighttime consumption patterns, reducing some of the error seen in the previous model.

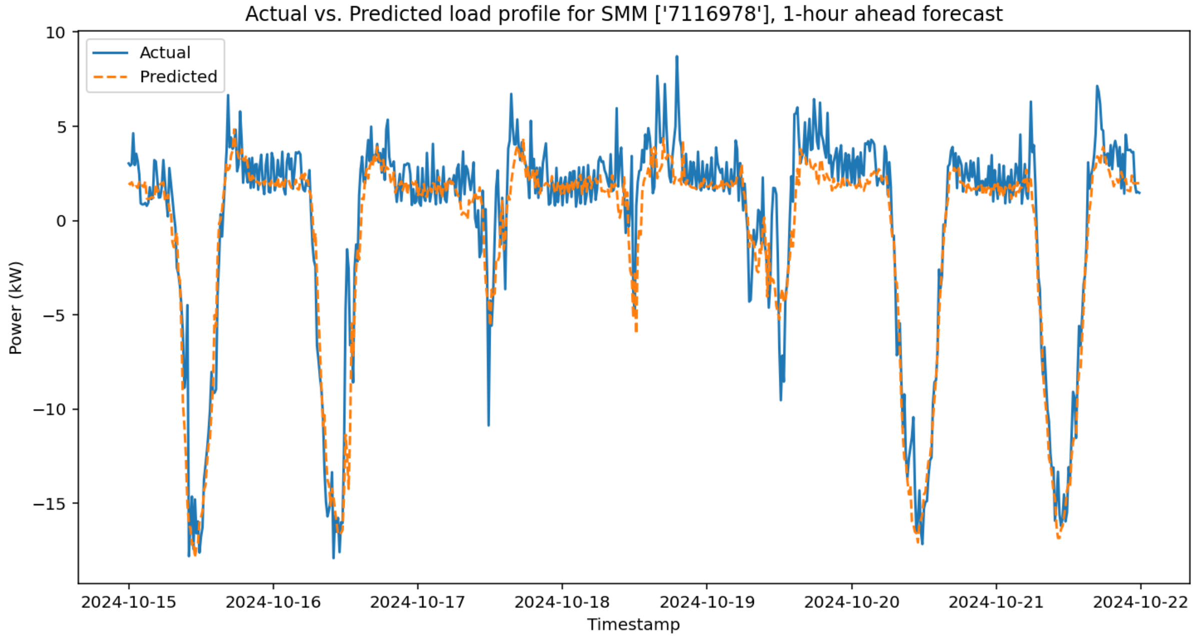


Figure 14: Forecasting model results evaluation, 1h ahead forecast (individual consumer level).

15-Minute Ahead Forecasting Performance (Figure 15): The figure shows the 15-minute ahead forecast, where we can see a significant improvement in accuracy. The model effectively tracks both the peaks and valleys in consumption, aligning much more closely with the actual load profile.

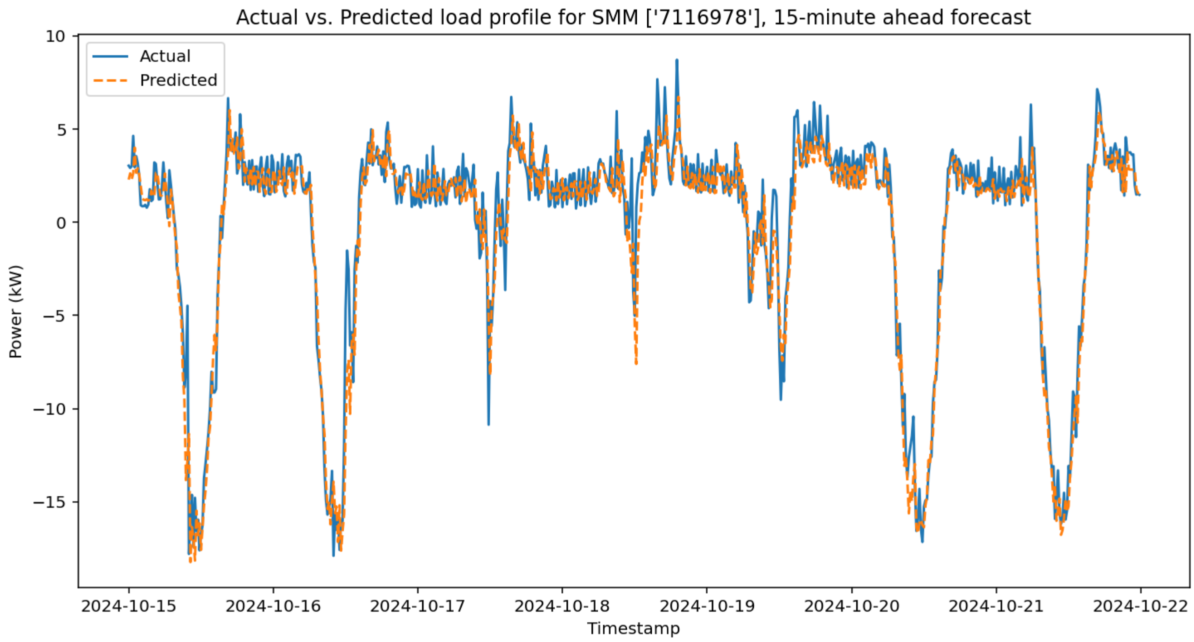


Figure 15: Forecasting model results evaluation, 15min ahead forecast (individual consumer level).

A quantitative comparison of the models with KPIs is shown in Table 12.

Table 12: Forecast models performance comparison, individual load.

	Day-ahead	1 hour ahead	15 minute ahead
nMAPE (%)	122.43	105.22	80.67
MAE (kW)	1.72	1.48	1.14
R2	0.87	0.90	0.94

For this individual consumer load, the nMAPE values are high due to the small overall magnitude of the load profile, making relative errors appear large. However, the MAE values are low and improve with the inclusion of lag features. The R^2 values also indicate that despite the complexity of individual consumer behavior, sGRID is capable of providing meaningful forecasts.

3.3.2.2 TLS stage calculation

The Traffic Light System (TLS) stage calculation within sGRID establishes a structured framework for assessing grid conditions and determining interactions between the distribution grid and the local flexibility market (sMART). The TLS categorizes the grid's operational state into four stages—green, yellow, orange, and red—each representing a different level of congestion severity and the corresponding need for intervention.

At each simulation timestep (every 15 minutes), sGRID executes a power flow analysis using the PandaPower library to assess the loading factors of grid elements, including MV feeders, MV/LV transformers, and LV feeders. These loading factors, expressed as a percentage of the rated operational capacity, provide the information of the operational state of the grid and indicating any potential congestions.

To determine the TLS stage for each grid element, sGRID uses two predefined sets of grid limits, which are stored in the tool's database:

- **Operational limits**: Defined by the technical parameters of the grid elements, these limits represent the maximum allowable loading before critical constraints are exceeded. Surpassing these limits indicates congestion, requiring immediate flexibility measures to prevent service disruptions or equipment overloading.
- **Planning limits**: Set below the operational limits, these limits enable the utilization of flexibility for investment deferral. Since grid infrastructure upgrades require significant time for planning and implementation, DSOs must begin procuring flexibility for congested areas before operational limits are reached.

The TLS stage for each element is determined as follows:

- **Green**: Loading is below the planning limit, indicating normal operation.
- **Yellow**: Loading exceeds the planning limit but remains below the operational limit.
- **Orange**: Loading surpasses the operational limit but remains below the critical threshold.
- **Red**: Loading significantly surpasses operational limits (by more than 100%) – this stage is expected to occur only in major outages or critical conditions and does not represent normal operating scenarios.

When a grid element enters a yellow or orange stage, sGRID calculates the required flexibility to mitigate the forecasted congestion, ensuring that the loading is reduced back within planning limits. The required flexibility is determined using the following formula:

$$F_{req} = P_{forecasted} - (P_{operational} \times P_{planning}^{lim})$$

Where:

- F_{req} is the required flexibility in kW,
- $P_{forecasted}$ is the forecasted loading for the element in kW
- $P_{operational}$ is the rated operational limit (capacity) of the grid element in kW,
- $P_{planning}^{lim}$ is the planning limit expressed as a percentage of the operational limit

These calculated TLS results for each timestep are stored within sGRID's database and simultaneously communicated to the sMART flexibility platform. Upon receiving notifications of yellow or orange stages, sMART initiates auctions to procure the necessary flexibility.

3.3.2.3 User interface

While sGRID primarily functions as a backend tool designed to enable DSOs to detect congestion ahead of time, it also includes a simple GUI that provides access to archived forecasting results. The GUI allows users to view historical and current day-ahead forecasts, offering a clear overview of past and upcoming grid conditions. This interface is not intended for active, real-time operation but rather serves as an informational tool, enabling DSOs to review results, conduct analyses, and support decision-making based on past congestion forecasts.

The sGRID UI is developed using the Taipy Python library, offering a simple and intuitive way to interact with the tool's stored data.

The UI includes the following functionalities:

- Select a date to retrieve the forecasting results, with future forecasts being limited to the next day.
- Browse results for key grid elements (transformers and lines), focusing on TLS stage assignments for each 15-minute timestep.
- Analyse grid status through tables and visualizations displaying loading factors, TLS stages, and required flexibility levels.

The initial screen (shown in Figure 16) of the UI presents the user with a date selection panel, allowing them to choose a specific day for which they wish to retrieve forecasting results. The tool restricts future selection to the next day, ensuring the user only accesses relevant, up-to-date predictions.

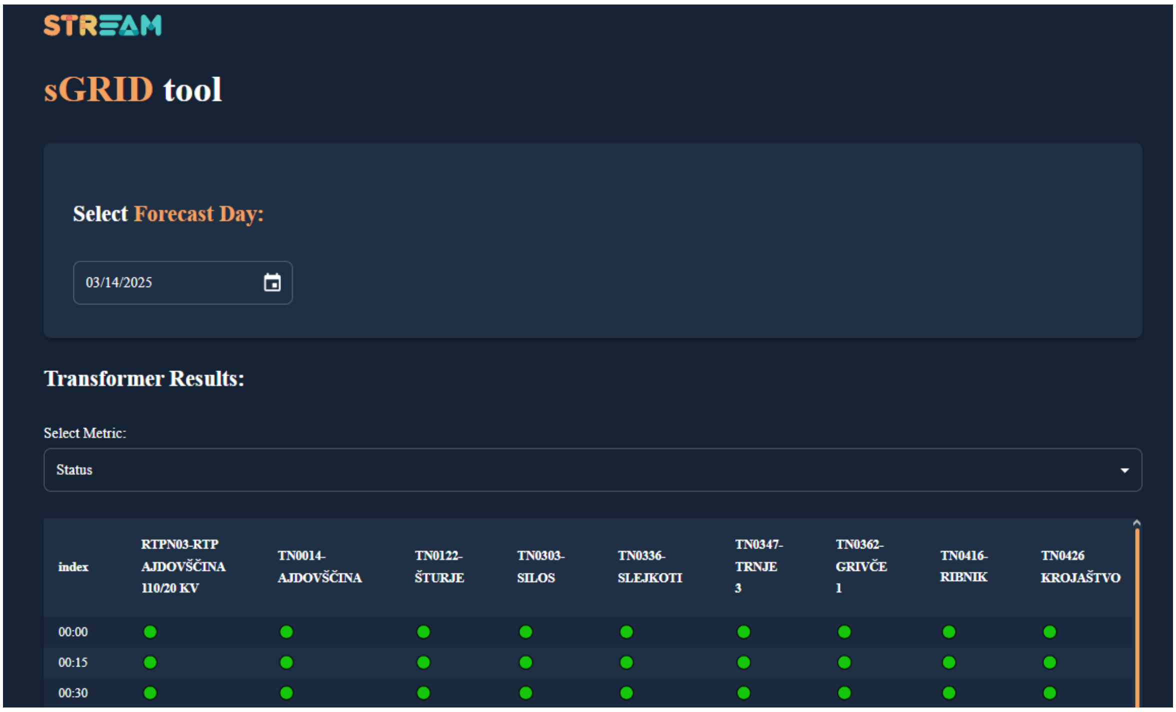


Figure 16: sGRID date selection screen.

Upon selecting a date, the tool displays results in tabular format, categorizing data for transformers and lines, which are the primary elements where TLS stages are assigned. The tables include key metrics such as:

- **Loading factor (%)**: Represents how much of the operational capacity a grid element is utilizing
- **TLS stage**: Displays the assigned congestion level (Green, Yellow, Orange, or Red) for each timestep.
- **Required flexibility (kW)**: Indicates how much flexibility would be needed to bring the element's loading back within planning limits.

To demonstrate the UI's functionality, an example case is presented where the parameters of a transformer (TN0336-SLEJKOTI) have been manually adjusted to showcase a congestion event. The UI provides two tables to track congestion:

Loading Factor table (shown in Figure 17): This table illustrates how the transformer's loading factor fluctuates over time. Between 11:15 and 15:00, the loading factor exceeds 100%, indicating an overload situation, before gradually decreasing below 80%.

Transformer Results:

Select Metric:

Loading Factor

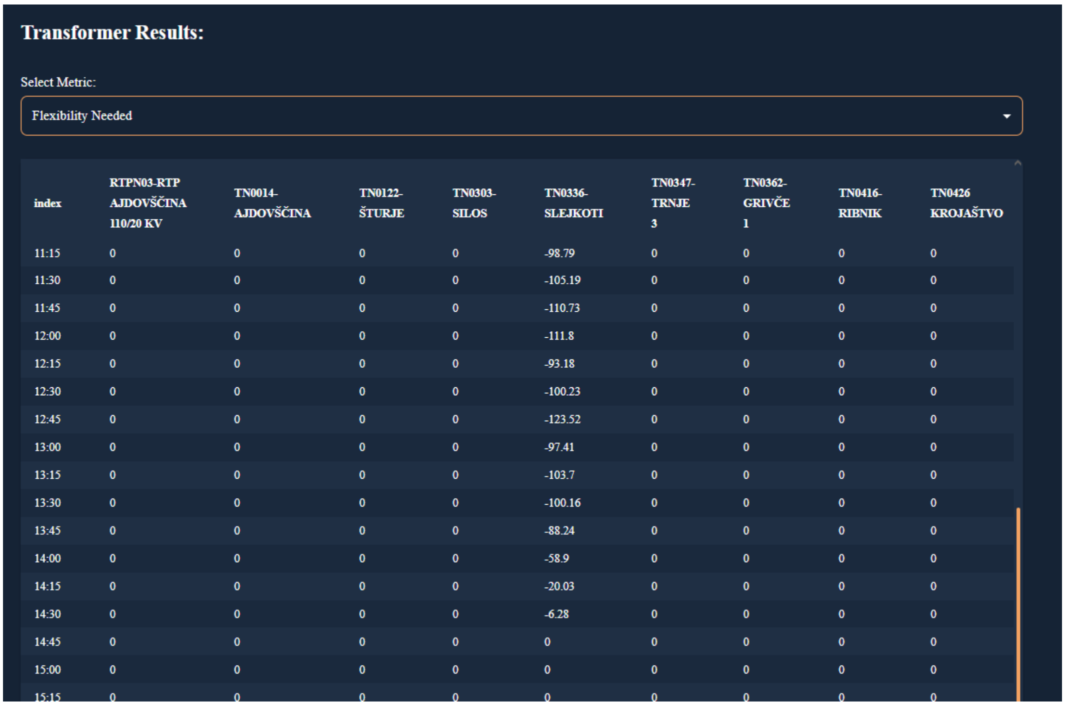
index	RTPN03-RTP AJDOVŠČINA 110/20 KV	TN0014-AJDOVŠČINA	TN0122-ŠTURJE	TN0303-SILOS	TN0336-SLEJKOTI	TN0347-TRNJE 3	TN0362-GRIVČE 1	TN0416-RIBNIK	TN0426 KROJAŠTVO
11:15	28.97	35.62	21.88	0.2	119.52	35.98	8.9	27.54	21.35
11:30	29.17	35.61	21.89	0.2	122.08	37.06	9.09	28.93	20.97
11:45	29.49	35.61	21.68	0.2	124.29	37.48	9.52	28.77	20.86
12:00	29.94	35.5	21.44	0.2	124.72	37.06	10.38	28	20.21
12:15	29.25	35.36	21.54	0.2	117.27	37.39	9.62	27.12	19.24
12:30	29.75	35.39	22.09	0.2	120.09	38.79	9.94	27.62	19.45
12:45	29.17	36.14	21.74	0.2	129.41	39.61	10.61	30.28	20.61
13:00	28.17	35.44	21.5	0.2	118.96	40.26	11	26.96	21.35
13:15	27.07	35.24	21.24	0.2	121.48	38.07	11.28	27.85	20.65
13:30	26.57	35.31	20.98	0.2	120.07	39.35	11.86	28.22	20.49
13:45	26.21	35.07	21.36	0.2	115.3	38.61	11.15	27.7	20.82
14:00	25.48	34.92	20.91	0.2	103.56	35.64	11.95	23.73	20.28
14:15	24.04	34.73	20.29	0.2	88.01	28.31	11.65	17.91	20.1
14:30	24.11	34.62	20.42	0.2	82.51	27.28	13.36	17.53	19.24
14:45	24.13	34.59	21.34	0.2	78.52	25.93	15.13	15.43	19.91
15:00	24.14	34.49	20.83	0.2	75.8	25.6	15.69	16.13	19.52
15:15	23.87	34.23	20.53	0.2	66.7	24.58	15.24	16.94	19.12
15:30	23.88	34.36	20.25	0.2	66.09	25.28	15.33	15.24	18.92

Figure 17: sGRID transformer results example - loading factor.

TLS Stage Visualization (shown in Figure 18): The next table displays the TLS stage assigned to the transformer for each timestep. As the forecasted load surpasses operational limits between 11:15 and 14:00, the TLS stage transitions from Orange (severe congestion) to Yellow (moderate congestion) before finally dropping to Green at 14:45, signifying a return to normal operation.

Figure 18: sGRID transformer results example - TLS status.

In addition to congestion identification, the UI displays the calculated required flexibility necessary to alleviate overloading (shown in Figure 19). This table highlights this parameter, where negative values represent the need to reduce consumption in the congested area under the transformer.



index	RTPN03-RTP AJDOVŠČINA 110/20 KV	TN0014-AJDOVŠČINA	TN0122-ŠTURJE	TN0303-SILOS	TN0336-SLEJKOTI	TN0347-TRNJE 3	TN0362-GRIVČE 1	TN0416-RIBNIK	TN0426 KROJAŠTVO
11:15	0	0	0	0	-98.79	0	0	0	0
11:30	0	0	0	0	-105.19	0	0	0	0
11:45	0	0	0	0	-110.73	0	0	0	0
12:00	0	0	0	0	-111.8	0	0	0	0
12:15	0	0	0	0	-93.18	0	0	0	0
12:30	0	0	0	0	-100.23	0	0	0	0
12:45	0	0	0	0	-123.52	0	0	0	0
13:00	0	0	0	0	-97.41	0	0	0	0
13:15	0	0	0	0	-103.7	0	0	0	0
13:30	0	0	0	0	-100.16	0	0	0	0
13:45	0	0	0	0	-88.24	0	0	0	0
14:00	0	0	0	0	-58.9	0	0	0	0
14:15	0	0	0	0	-20.03	0	0	0	0
14:30	0	0	0	0	-6.28	0	0	0	0
14:45	0	0	0	0	0	0	0	0	0
15:00	0	0	0	0	0	0	0	0	0
15:15	0	0	0	0	0	0	0	0	0

Figure 19: sGRID transformer results example - required flexibility.

3.3.2.4 Connection to sMART

To enable proactive congestion mitigation, sGRID establishes direct communication with the sMART flexibility platform. After determining the TLS stage and calculating the required flexibility at each simulation timestep, sGRID generates structured JSON messages conveying congestion details and the requested flexibility to sMART.

An example message structure is shown below:


```
{
    "CongestionDetails": {
        "CongestedElementID": "7088670",
        "OrderAmount": 30000,
        "OrderDirection": "Downward",
        "ISPNumber": 1,
        "AuctionDate": "2024-12-21",
        "AuctionID": ""
    },
    "TLS": {
        "Stage": "Yellow"
    }
}
```

In this JSON structure:

- **CongestedElementID** identifies the grid element experiencing congestion (e.g., transformer or feeder ID).
- **OrderAmount** specifies the required flexibility volume (in watts) necessary to alleviate congestion
- **OrderDirection** indicates the type of flexibility needed.
- **ISPNumber** (Imbalance Settlement Period number) indicates the specific 15-minute timestep when congestion occurs.
- **AuctionDate** represents the day on which flexibility is required

- **AuctionID** is an unique ID assigned by sGRID and sent to sMART for auction creation.
  - **TLS Stage** communicates the severity of congestion (TLS stage) to sMART

3.4 SPLAN IN THE SLOVENIAN PILOT SITE

3.4.1 Overview and architecture

sPLAN is a tool designed for distribution grid planning and analysis. Its core functions include:

- **Grid state simulation and analysis**: For various scenarios with different numbers of new devices (EVs, HPs, PVs, and BESS) and annual load increases.
- **Identification of grid vulnerabilities:** The tool identifies weak parts of the grid that may require reinforcement.
- **Flexibility measures simulation:** sPLAN enables the flexibility measures as alternative or complementary solution to grid reinforcement strategies.

Figure 20 shows the architecture of the sPLAN tool. The architecture is divided into three main components: the database, the graphical user interface (GUI), and the engine. The database stores different types of data, including the grid topology of the Slovenian pilot site, all electrical parameters of grid elements, and real historic measurements that are used as inputs for typical weekly profiles for each season. The database also stores all the relevant data related to the costs of reinforcement measures. Based on the availability of the data from the real network the database is regularly updated.

GUI allows interactive user interaction with the tool, enabling the user to set all the simulation parameters and analyse the tool outputs. The parameters that can be configured on the input mask that is part of the GUI are the following:

Through the GUI, users can configure several simulation parameters, such as:

- **Simulation Period**: Due to the long simulation time as a result of the complexity of the Slovenian pilot site grid model, the simulation period in the sPLAN is limited to one week. Typical seasonal profiles are generated for each consumer in the grid based on the measurements. User can simulate one week in each of the four seasons.
- **Photovoltaics** (PVs):
  - Number of new devices.
  - Installed power (kW), which is the same for all new devices placed in the grid.
  - Option to include a BESS alongside the PV installation (Yes/No).
  - If BESS is included: maximum power (kW) and capacity (kWh).
- **Electric Vehicles** (EVs): Number of new charging stations.

- **Heat Pumps** (HPs):
    - Number of devices.
    - Area (m^2).
    - Efficiency (kWh/m^2).
- **Uncontrollable Loads**: Annual change (increase or decrease) can be applied.

The simulation results are presented and visualized through various interactive charts within the GUI, as detailed in later sections.

The sPLAN engine includes all the scripts and algorithms that are running the tool. The process begins with the data preparation stage, in which the tool extracts the relevant seasonal profiles based on user inputs. Next, new technologies are placed into the grid model and their impacts on load profiles are incorporated. The updated load profiles are used as inputs in a load flow simulation, with the results analysed to estimate grid conditions and identify potential issues. To tackle the detected problems two modules are available within the tool: the Reinforcement Measures module, which upgrades the overloaded elements in the grid, and the Flexibility module, which adjusts the load profiles of consumers with new technologies. A subsequent load flow simulation with deployed measures then verifies the success of these interventions.

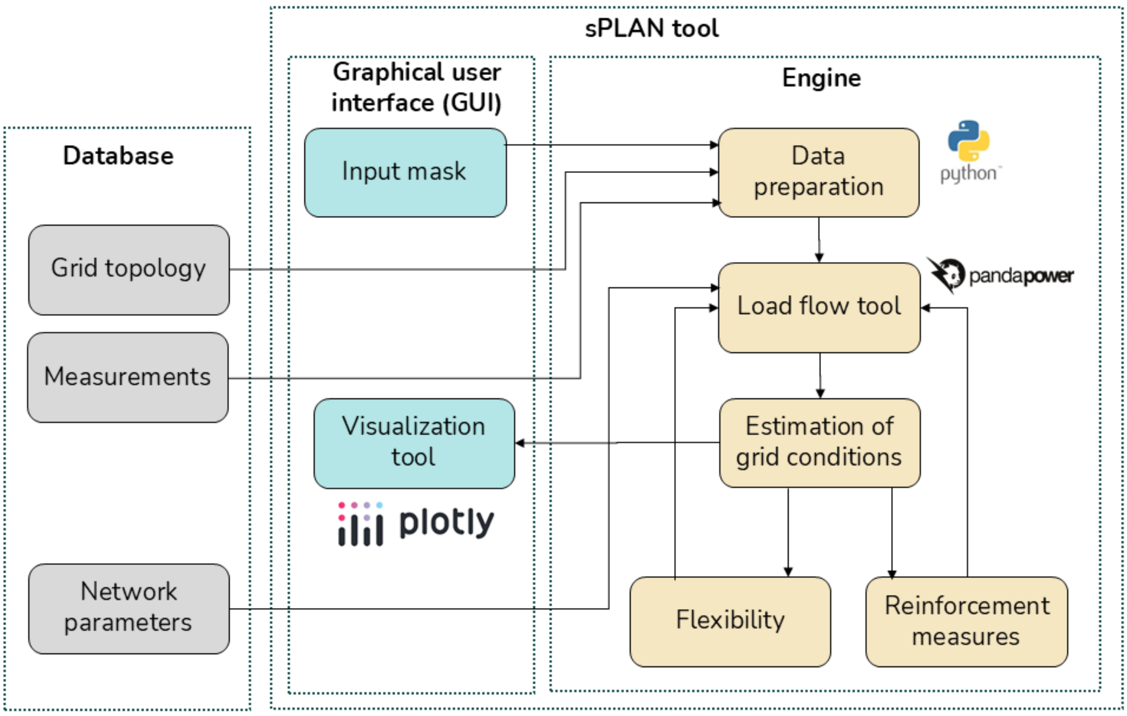


Figure 20: sPLAN architecture (SI pilot).

3.4.2 sPLAN functionalities

In general, the main functionality of sPLAN is to run load flow simulation on the network model which mimics the operation of a real network. As the main purpose of the tool is to support decision-making on network planning and development, the additional functions of the tool ensure that all aspects are addressed. The main window of the tool, which represents the basic interaction of the user with the tool, is shown in the picture below.

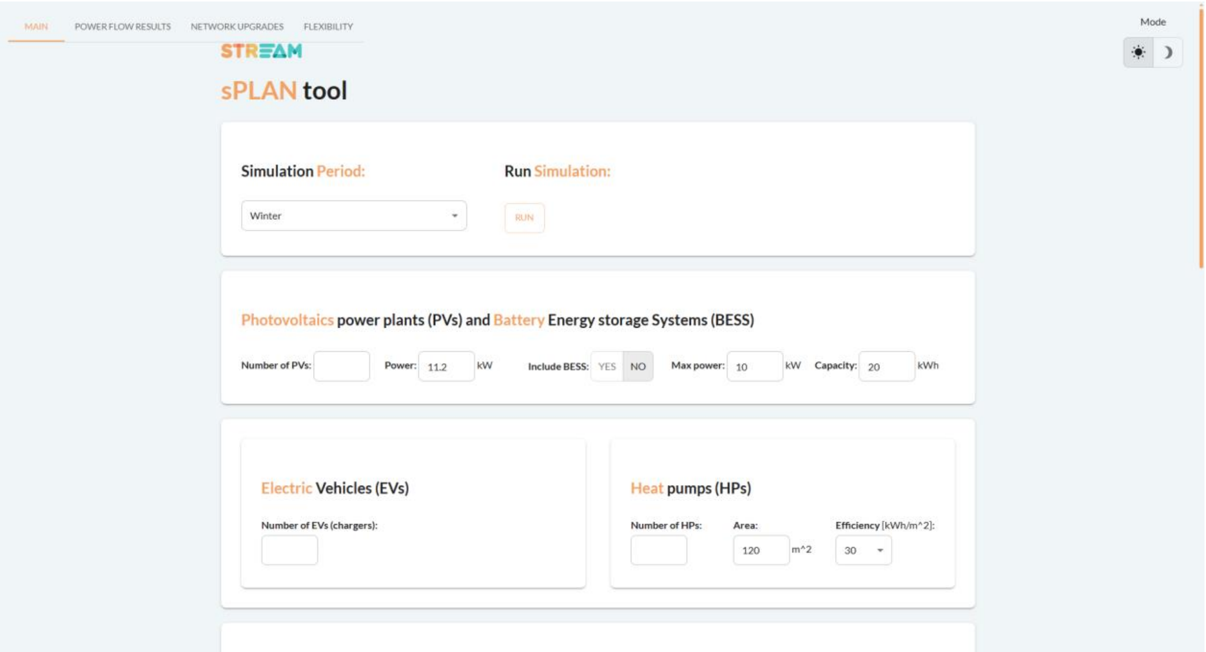


Figure 21: sPLAN main window.

The main page of the tool also includes the basic visualization of the network as shown in Figure 22. In addition to the main tab, the tool has three other tabs (for results visualization, network upgrades and flexibility).

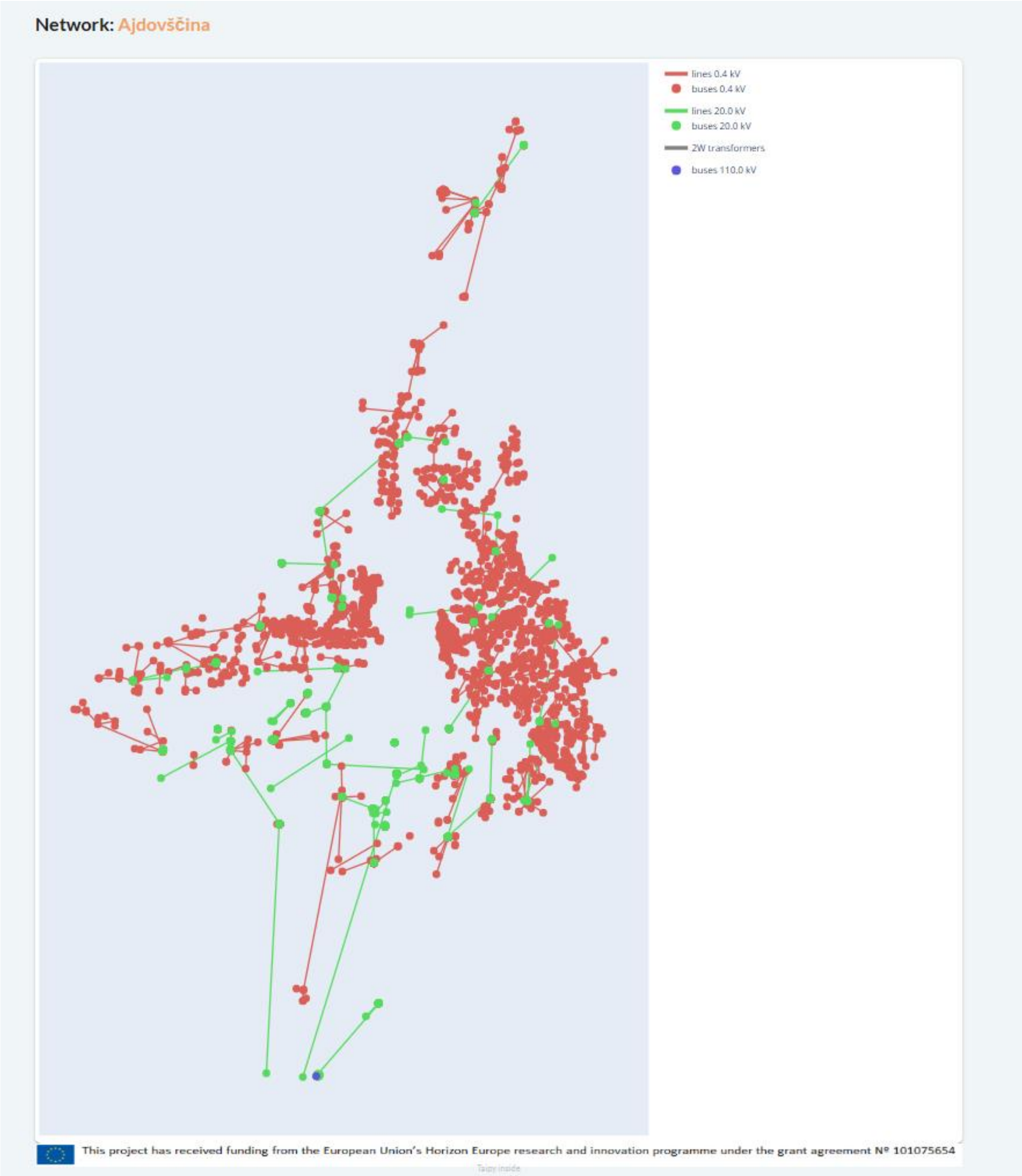


Figure 22: sPLAN basic network visualization.

The simulation results are displayed in a separate tab (power flow results), where the user can select to display the results for a specific element or part of the network. Figure 23 shows the loading factor for two selected lines over one week at a 15-minute time resolution.

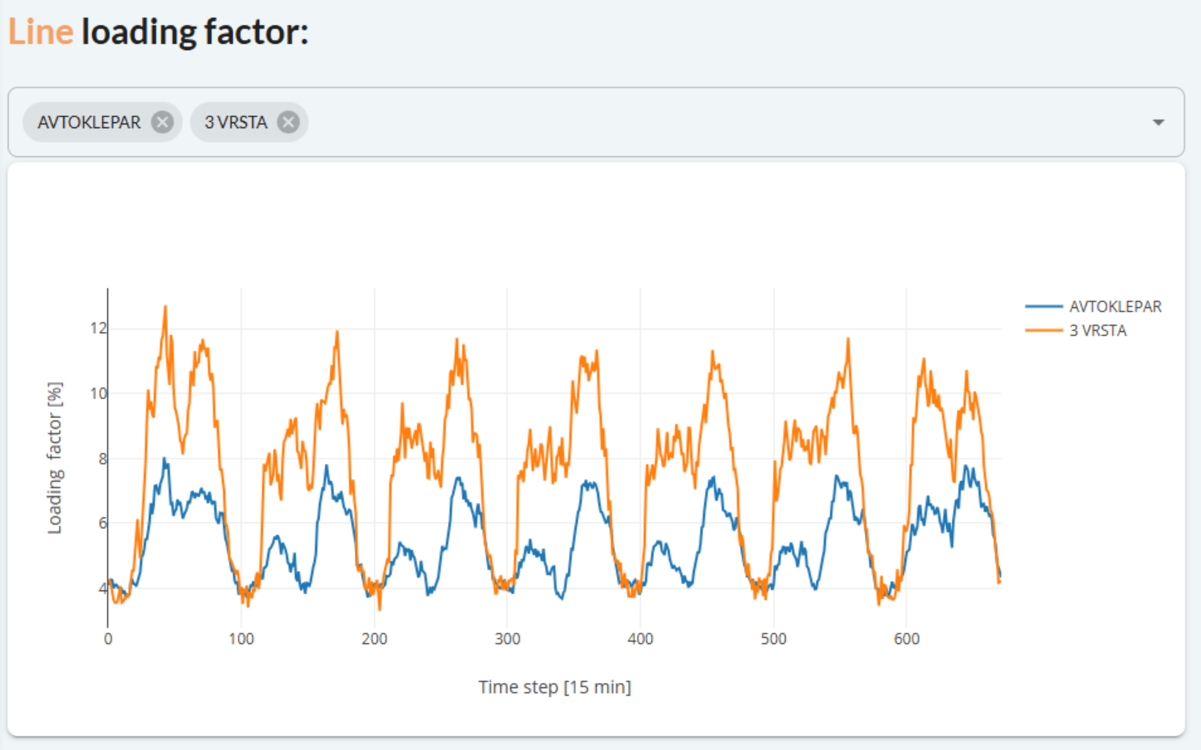


Figure 23: Loading factor for two selected lines.

The individual basic modules that support the sPLAN tool and its functionalities are described in separated sections.

3.4.2.1 Technology placement module

Placing new devices in the network can be done either manually by the user or through an algorithm. In the first case, the user has to manually select the consumer and define the type of device to be added. In the second case, the user only defines the number of each new device and devices are placed following some predefined rules. The algorithm aims to create a technology diffusion scheme in which more chances are given to PV owners to adopt other new technologies. In general, depending on the technology, different rules are given. However, the basic rule is that consumers with already installed PV and 3-phase connection have more chances to adopt new technologies. One of the rules also gives priority to the users that are further away from their main supply point. The aim of the latter is to simulate the worst-case scenarios. An example of a table giving all the key data on the connection of consumers and the list of installed devices is given in Figure 24.

name	SMM	TP	Feeder	P+	P-	un	distance	PV	EV
224685	7054194	TN0347-TRNJE 3	3 VRSTA	6		0.4	2.455	0	0
162887	7126628	TN0347-TRNJE 3	R.O.CANKARJEV TRG	6		0.4	2.457	0	0
121816	7050726	TN0969-PUTRHI	2. R.O.ŽUPANČIČEVA	4		0.4	2.41	0	0
121816	7050727	TN0969-PUTRHI	2. R.O.ŽUPANČIČEVA	6		0.4	2.41	0	0
121816	7050728	TN0969-PUTRHI	2. R.O.ŽUPANČIČEVA	6		0.4	2.41	0	1

Figure 24: Key data on the connection of consumers and the list of installed devices.

3.4.2.2 New profiles generation

- **EV charging profiles**: The algorithm for simulating the charging profiles of EVs is run for a typical weekday and a typical weekend day. Results are then combined in a weekly profile. The idea behind EV charging is to simulate a daily trip for each vehicle, thus creating daily schedules for being parked at home and charging to be fully charged until the next departure. The necessary inputs to generate charging profiles of EVs are the number of EVs placed in the grid and EV characterisation including average consumption, battery usable and max charging power. The average consumption is determined based on the technical data of the EVs on the market. The battery's useful capacity is chosen at random between 40 and 100 kWh. The charging power is determined by the connection type and power of the user to which the EV is placed. If the connection is three-phase and the contractual power is greater than 23 kW, the charging power is 22 kW. If the connection is three-phase and the contractual power is less than or equal to 23 kW, the charging power is set to 11 kW. In the case of a single-phase connection, the charging power is calculated as the minimum between 7.4 kW and the contractual power reduced by 1 kW. It is also possible to consider a charging power of 11 kW for all users with a three-phase connection, as this is becoming the standard for electric vehicles.
- **Heat pumps**: The HP's power demand is determined by its coefficient of performance (COP) and the amount of heat required. Based on the area and the efficiency defined by the user annual heat demand is defined. The COP of the heat pump is a function of different parameters and varies during the year. The manufacturers usually provide the max COP value, which also depends on the heat pump type. The HP power profiles within sPLAN are generated considering the national time series of the heat demand and the COP, scaled to the annual heat demand. Average weekly seasonal profiles are extracted from the annual profile.
- **PVs and BESS**: The PV profiles are based on the solar irradiance data for the selected location and the technical parameters of the PV modules. In the case of BESS, their operation is optimised in order to increase own self-supply of PV produces electricity.

3.4.2.3 Grid conditions estimation

In sPLAN the simulations (grid calculations) are done through the open source PandaPower library aimed at the automation of analysis and optimization in power systems. Through this library, the simulation results for each iteration (time step) are stored directly in the network model. Key network parameters such as power, voltage and loading factor are stored in separate tables within the tool for the entire simulation period. The PandaPower library itself contains certain options for displaying results, such as heat map which is integrated in sPLAN tool as shown in Figure 25.

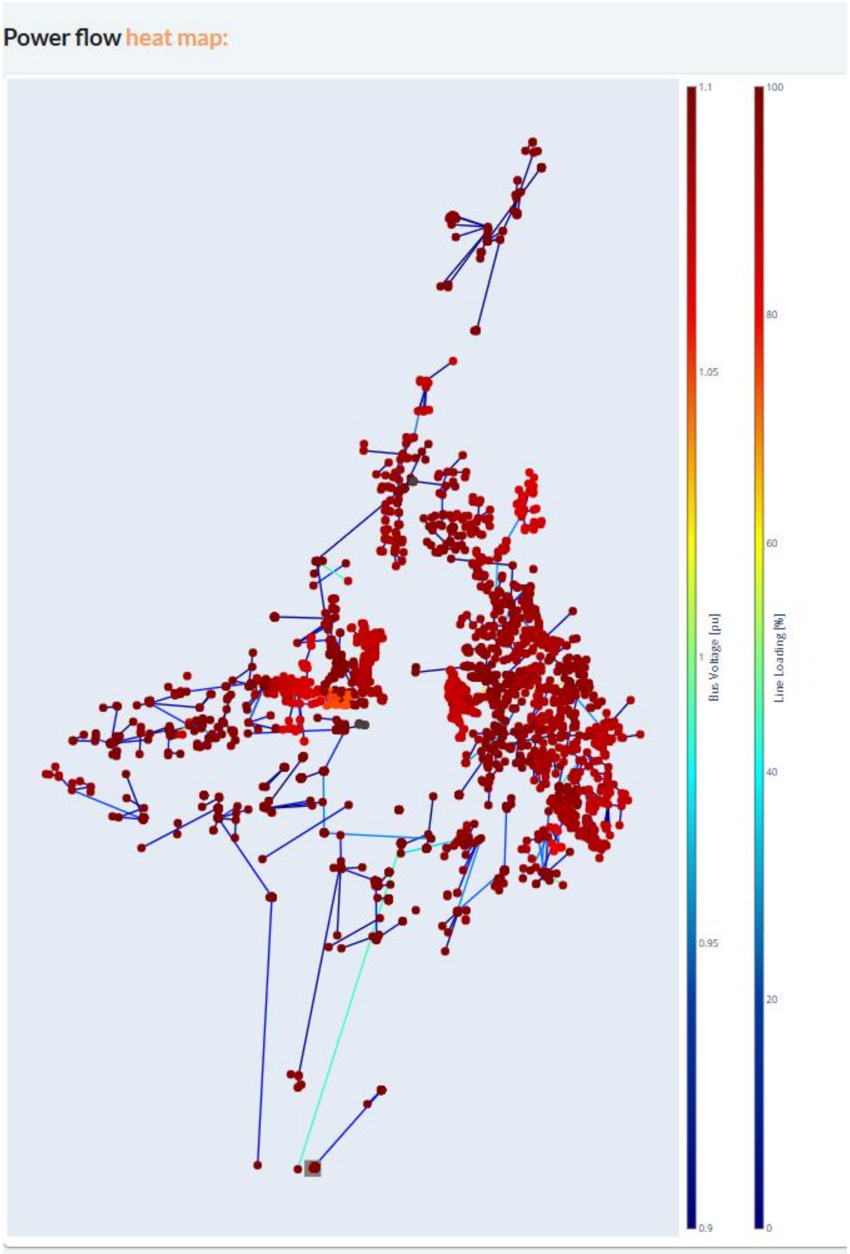


Figure 25: Buses voltages and lines loadings heatmap.

3.4.2.4 Flexibility module

As regards the use of flexibility in sPLAN, the tool allows the adaptation of profiles of flexibility resources during periods of increased load, i.e. when the loading factor of network elements exceeds a certain value, or when voltages are outside the allowed limits. The primary flexibility resources are devices added to the grid in the context of the definition of the initial parameters of the simulation, i.e. PVs and BESS, HPs, EVs. The module allows the adjustment of power profiles during periods of increased load, thus simulating the flexibility services that could be offered to the grid operator as an alternative to grid upgrades. The purpose of the flexibility module is not to determine the activation of the most appropriate, cost-effective flexibility resources and to set the price of flexibility but to determine the potential performance of flexibility deployment. However, as price is key to evaluating flexibility-based services, the latter is available within the STREAM ecosystem and allows sPLAN to perform an economic evaluation of flexibility compared to grid upgrades.

3.4.2.5 Optimal reinforcement module

Based on the simulation results and the calculated loading factor of grid elements, the Optimal reinforcement module determines which elements in the grid model need to be replaced with more powerful ones. The threshold of the loading factor is set by the user. For all upgrades made, the tool provides an indicative calculation of the cost of the investment. An example of a comparison of the loading factor of an existing and an upgraded element for the same set of input data, and an estimate of the investment cost is shown in Figure 26.

Figure 26: Loading comparison and upgrade costs results.

4 SPANISH PILOT SITE

4.1 DATA ANALYTICS

Due to the complexity of energy systems in general, and especially of local flexibility markets, data analytics are essential for improving the knowledge acquisition from the data coming from the pilot site, and allowing stakeholders to take the correct actions and decisions to properly act in the local flexibility market (e.g. forecasting congestions, estimating the flexibility that a prosumer can provide, etc.) [11]. For such purposes, the tools deployed in the Spanish pilot site integrate a range of data analytics. The following table summarizes the models that are used in the tools, identifying the tool in which the analytics are used:

Table 13: Data analytics models in the Spanish pilot site.

Data analytics model	Involved tool(s)
Time series forecasting	sGRID, sFLEX, sENC
Power flow analysis	sGRID
Sensitivity analysis	sGRID
Baseline calculation	sENC, sFLEX
Energy performance-based clustering	sENC
Fault detection and diagnostics	sFLEX

While the baseline calculation is extensively described in Section 4.2 as part of the user profiling process, a comprehensive explanation of the other models is provided in the following sections. This includes the rationale for each model in the context of the pilot site, the input data, the data processing, including the steps of the process, libraries used and algorithms, and the output data resulting from the analytics.

4.1.1 Time series forecasting

4.1.1.1 Introduction

Advancing the behaviour of the elements that compose the system being monitored is always a useful tool, which serves multiple purposes. Using historical records of time series, each element is modelled to predict its patterns and dependency with other external variables.

For this purpose, a generalist time series forecasting service is used by sGRID, sFLEX, and sENC in the ES pilot, each of them making use of the algorithm to predict the status of their key managed assets for the next 24 hours. This information will then be used both as-is and as input for further data analytics services.

4.1.1.2 Input datasets

The following datasets will be used in each tool:

Table 14: Input datasets for time series forecasting in ES pilot.

Data element	Units	Tool(s)	Comments
Active and reactive power of loads, generators, grid connection points	kW, kvar	sGRID	Historical data (1 year)
Active power of PV and community battery	kW	sGRID, sENC, sFLEX	Historical data (1 year)
Prosumers consumption and generation	kWh	sENC	Historical data (1 year)
EnC collective and domestic self-consumptions	kWh	sENC, sFLEX	Historical data (1 year)
Active power of EV chargers	kW	sENC, sFLEX	Historical data (1 year)

Additionally, the following parameters will be gathered for seasonality characterisation:

Table 15: Seasonality data for time series forecasting in ES pilot.

Data element	Units	Comments
Temperature	°C	
Solar irradiation	W/m^2	
Day typology	-	Day and month, day of the week, working days, weekends, holidays

4.1.1.3 Data processing and analysis

The generalist time series forecasting service is implemented in Python using XGBoost library [12]. It allows the configuration of queries to different repositories for retrieving training and regressor data.

A forecasting model is basically defined by a set of input stages in the form of database queries. The results of these queries will be used to train the model. The results of each input stage are homogenised and combined as follows:

- ds: timestamp associated to the value(s)
- y: forecast variable (typically returned by the first stage of the model)
- x_1, x_2, …, x_n: regressors (i.e. additional time series that have a linear relationship with the forecast variable)

The service is prepared to receive definitions of input stages from MongoDB, InfluxDB, Elasticsearch, NATS, REST APIs, as well as custom outputs from other services. JSONBender [13] is used to homogenise the results of each input stage. Additionally, the model can incorporate country- and state/province-specific holidays from a dedicated library in Python [14], as well as considering the day of the week as regressor.

The input stages are used to create a model using the scikit-learn API implementation for XGBoost regression [15]. For each stage, data are cleaned to discard anomalous values and fill possible gaps. With these data, the models are initially trained and forecasted values (same as described inputs in Table 14) are periodically calculated by request of the corresponding STREAM tool. The results are provided by the service via NATS and subsequently stored in the long-term repository of each application.

The calculated wMAPE for an average week taken into study are:

- wMAPE (demand): 5,63%
- wMAPE (generation): 23,24%

4.1.2 Power flow analysis

4.1.2.1 Introduction

The analysis of power flows in an electrical network study the electrical conditions and distribution of power in the grid to assess its status and identify potential deviations and hazards, such as congestions and overvoltages. This analysis calculates additional parameters that are not provided by the measuring equipment deployed in a subset of elements of the network, thus enhancing and expanding its visibility.

In the ES pilot, sGRID integrates a model of the MV and LV network of Crevillent, which is used to provide the system operator with the results of the power flow calculation in different scenarios. Depending on the input data, the calculation can be performed for the current time, past situations, foreseen circumstances, or simulated scenarios.

4.1.2.2 Input datasets

The following datasets are used as inputs of the power flow analysis of sGRID:

Table 16: Input datasets for power flow analysis in ES pilot.

Data element	Units	Comments
LV and MV GIS model	N/A	Including technical limits of lines, nominal voltage of buses, installed power of loads and generators, status of switches
Active and reactive power of loads and generators	kW, kvar	
Active and reactive power of grid connection points	kW, kvar	Optional

4.1.2.3 Data processing and analysis

The model of the MV and LV network in the Spanish pilot is gathered from Enercoop's GIS through a dedicated API. Each element type is transformed into a MongoDB document associated to a specific network (i.e. MV grid or LV feeder). The types of elements and their main features are:

- **Substation:** Name, location.
- **Bus:** ID, substation, nominal voltage.
- **Line:** ID, bus 1, bus 2, length, resistance, reactance, capacitance, current limit.
- **Transformer:** ID, MV bus, LV bus, nominal power, rated voltage per side, resistance, reactance.

- **3-winding transformer:** ID, HV bus, MV bus, LV bus, rated voltage per side, rated power per side, X per side, R per side.
- **Switch:** ID, bus 1, bus 2, type, default state (open/closed).
- **Dangling line (grid connection points):** ID, bus, controllable.
- **Load:** ID, bus, nominal active and reactive power.
- **Generator:** ID, bus, installed power, voltage per unit, controllable.
- **Usage point:** ID, node, nominal voltage, nominal active and reactive power.
- **Meter:** ID, usage point.

The use of this model in sGRID is twofold:

- The power flow analysis service transforms the elements of each network into a PandaPower network model [16] to be used in its calculations, and
- The GUI of the tool represents each network as a dynamic Cytoscape diagram [17] where the results of the calculations can be shown.

Besides the network model, the power flow service receives a picture of the status of the network in terms of active and reactive power of all loads, generators, and dangling lines. These values can represent the current status of the grid, forecasted situations, historical records, or a simulated scenario.

The power flow service of sGRID is implemented in Python and performed using the *runpp* method of PandaPower [18]. The Newton-Raphson algorithm is used to solve the balanced AC power flow problem. If the power flow calculation converges, the results detailed in Table 17 are stored in the corresponding repository of sGRID for subsequent access by other services and the network operator. In case of error, a PandaPower *diagnostic* [19] is run in case some parameter in incorrect and/or can be adjusted to obtain proper results.

Table 17: Output datasets for power flow analysis in ES pilot.

Data element	Units	Comments
Active and reactive power flow at grid lines, busbars, transformers	kW, kvar	
Active and reactive power at loads, generators, dangling lines	kW, kvar	May differ from inputs if they are controllable
Current at grid lines, transformers, switches	A	
Power losses at grid lines, transformers	kW, kvar	
Voltage and angle at nodes	V, deg	

4.1.3 Sensitivity analysis

4.1.3.1 Introduction

A sensitivity analysis calculates the impact of the variation of flexible assets in the grid (with modulation of consumption and/or generation) on the power flows and the values of the current in specific lines.

This algorithm is used in sGRID in the ES pilot upon a detection or prevision of a congestion, in order to identify those flexible elements that will have higher influence on the resolution of the contingency.

4.1.3.2 Input datasets

The sensitivity analysis takes model of the grid, the results of power flow calculation as inputs, i.e.

Table 18: Input datasets for sensitivity analysis in ES pilot.

Data element	Units	Comments
LV and MV GIS model	N/A	Including technical limits of lines, nominal voltage of buses, installed power of loads and generators, status of switches
Active and reactive power flow at grid lines, busbars, transformers	kW, kvar	From power flow analysis
Active and reactive power of Identified flexible assets	kW, kvar	

4.1.3.3 Data processing and analysis

As PandaPower does not implement a sensitivity analysis *per se*, a manual approximation for this purpose has been implemented as a Python algorithm using the network model described in section 4.1.2.

When a congestion in a line is identified by sGRID, each flexible asset in that network is evaluated in parallel to calculate its potential effect to alleviate the situation. The sensitivity factor for each asset is calculated by running a power flow calculation for the detected situation but varying the value of the setpoint (i.e. the active power value) for the specific asset.

Having the results of the power flow calculation where the congestion was identified, the algorithm is implemented as follows for each flexible asset in the same network/feeder as the congested line:

1. Based on the technical limits of the line and the current status of the asset, define two possible setpoints for generation increase/demand reduction (‘S_{up}’ and ‘S_{down}’).
   a. ‘S_{up}’: An increase in generation/reduction in demand of 1 kW (or less if not possible).
   b. ‘S_{down}’: A reduction in generation/increase in demand of 1 kW (or less if not possible).
2. Run a power flow calculation for each of the defined setpoints.
3. Based on the output of each analysis, calculate the sensitivity factor of the flexible asset as:
   a. $SF_{down} = \frac{Pfa_{down} - Pfa_{orig}}{Pline_{down} - Pline_{orig}}$
   b. $SF_{up} = \frac{Pfa_{up} - Pfa_{orig}}{Pline_{up} - Pfa_{orig}}$
   c. $SF_{fa} = \frac{SF_{down} + SF_{up}}{2}$

   Where:
   - SF_{down}: Sensitivity factor when applying the ‘S_{down}’ setpoint (%).
   - Pfa_{down}: Active power of the flexible asset when applying the ‘S_{down}’ setpoint (MW).

- Pfa_{orig}: Active power of the flexible asset in the original power flow calculation (MW).
- $Pline_{down}$: Active power flowing on the congested line when applying the 'S_{down}' setpoint (MW).
- $Pline_{orig}$: Active power flowing on the congested line in the original power flow calculation (MW).
- SF_{up}: Sensitivity factor when applying the 'S_{up}' setpoint (%).
- Pfa_{up}: Active power of the flexible asset when applying the 'S_{up}' setpoint (MW).
- $Pline_{up}$: Active power flowing on the congested when applying the 'S_{up}' setpoint (MW).
- SF_{fa}: Sensitivity factor of the flexible asset (%).

The calculated flexible assets are stored in sGRID and presented to the operator to be used in the decision process when accessing the flexibility markets:

Table 19: Output datasets for sensitivity analysis in ES pilot.

Data element	Units	Comments
Sensitivity factor for each flexible asset	%	Ratio of power increase in line per generation increase (or demand reduction) unit

4.1.4 Energy performance-based clustering

4.1.4.1 Introduction

The clustering of the portfolio of the energy community based on consumption profile is performed in sENC with different purposes:

- Identify members that present a similar behaviour in terms of consumption, in order to optimise the cost of the EnC by means of tariff recommendation or achieving better deals with the utility, and
- Group the flexibility of members with common patterns in consumption to maximize their suitability to participate in the flexibility market.

4.1.4.2 Input datasets

The energy performance-based clustering algorithm takes the following datasets as inputs:

Table 20: Input datasets for energy performance-based clustering in ES pilot.

Data element	Units	Comments
Prosumers consumption and generation	kWh	Baseline calculated from historical records
Prosumer characterisation	-	Installed power, production & storage capabilities
Day typology	-	Day and month, day of the week, working days, weekends, holidays

4.1.4.3 Data processing and analysis

Previous to the clustering process, the historical data of the Energy Community members is analysed in order to calculate their baselines. The available consumption and generation records are characterised in hourly curves based on the type of day (day of the week, holiday) season of the year (winter/summer). Each 24-value curve is stored in the sENC repository for subsequent use as input for the clustering analysis.

The clustering of members of the Energy Community is implemented as a Python script. The calculated baselines are imported as Pandas DataFrames [20] and each column properly labelled. Then, the optimal number of clusters is analysed using the “elbow” method as provided by Yellowbrick [21], a Machine Learning visualization extension of the scikit-learn library. The method runs k-means clustering of a sample of the dataset for a range of values (3-12), then computes the average score for all clusters in each value. When represented, the point of inflection on the plotted curve is the optimal number of clusters (k=4 in Figure 27). It is however not mandatory to plot the results, as the library provides this value as part of its results.

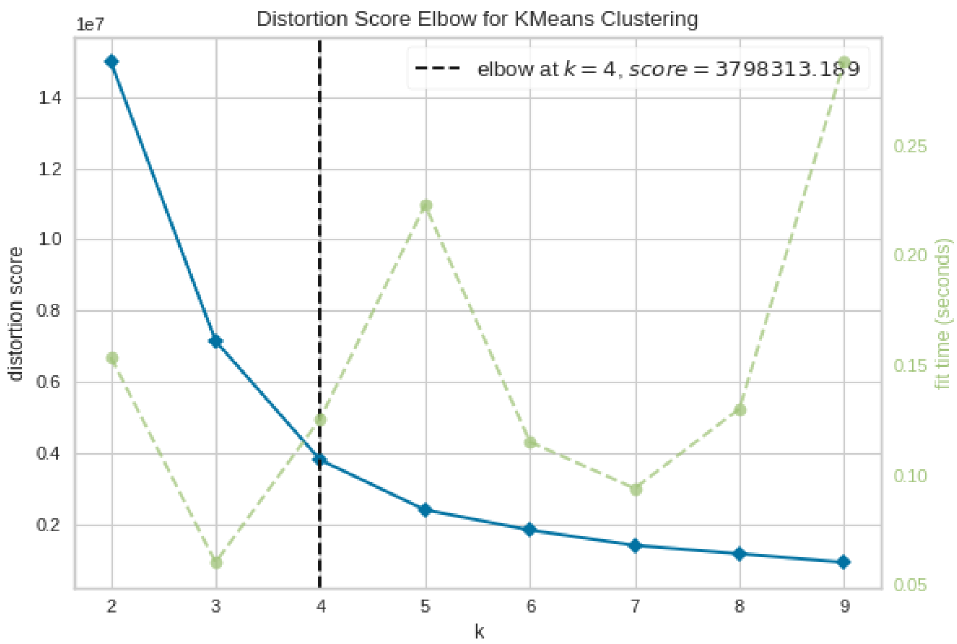


Figure 27: Distortion score elbow for k-means clustering.

Once the number of clusters is decided, the full dataset is used to run the k-means algorithm from the scikit-learn library [22]. The results of this calculation (including the new clusters generated) are stored in the sENC repository for subsequent use in the everyday management of the EnC, including market access.

4.1.5 Fault detection and diagnostics

4.1.5.1 Introduction

The maintenance of the assets involved in the operation of a system is an essential to maximize their performance and profitability. While corrective maintenance acts after a failure have occurred and preventive maintenance schedules periodical revisions, predictive maintenance aims at anticipating potential malfunctions or degradations of the devices based on the analysis of their historical records.

In the ES pilot, sFLEX implements predictive maintenance with its fault detection and diagnostics service, which aims at creating models of the flexible assets managed by the tool and identifying the discrepancies in tendencies of key parameters over time, alerting the operator when these variations exceed a specific threshold.

4.1.5.2 Input datasets

The service will focus on the community battery + PV installation in the ES pilot, with the following datasets being considered for the modelling:

Table 21: Input datasets for energy performance-based clustering in ES pilot.

Data element	Units	Comments
Temperature of batteries charger, battery management system (BMS), and PV inverter	°C	Historical data (1 year) and real time
Other metrics provided by the system (active and reactive power, voltage, current, frequency, state of charge, number of cycles)	Various	Endogenous variables
Weather data and day typology	Various	Exogenous variables, optional

4.1.5.3 Data processing and analysis

The service aims at creating a thermal model based on the historical records from the community battery + PV installation. The aim is that this Machine Learning model creates a forecast of the temperature in real time, i.e. the next value expected to come from the system.

Models from three different elements of the installations are created: the batteries charger, the battery management system (BMS), and the PV inverter. All three of them provide their internal temperature as a metric, as shown in Figure 28.

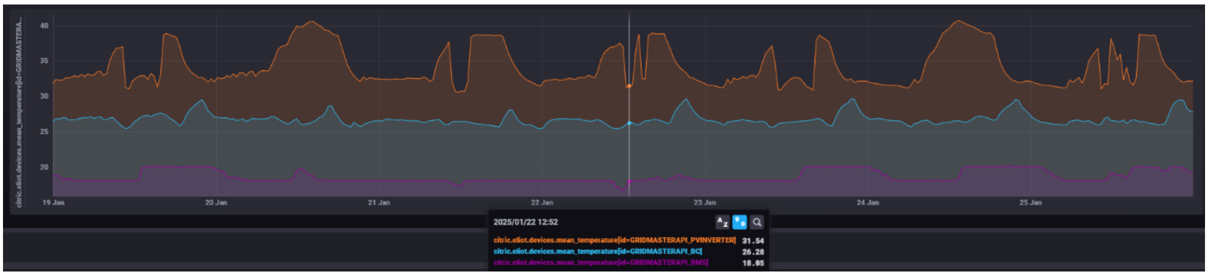


Figure 28: Temperature of different elements in the community battery + PV installation in Crevillent.

The steps to create each of this model are:

1. **Data cleaning and filtering**: Removing any anomalous or corrupted values from the historical records, both of the target and the endogenous variables. Values outside the expected ranges are marked as corrupted and discarded, as well as those records that have been identified as a non-operational state of the system or a malfunction. This is performed only over the training data set, in order to create a model that represents the optimal functioning of the installation, thus allowing the identification of deviations by comparison.
2. **Data transformation and annotation**: The target and the endogenous variables are formatted using Pandas [20] for its use as input in the model. Moreover, each record is classified based on its time stamp, characterizing it by month, day, hour, and day of the week.
3. **Model training and testing**: With the full data set (1 year of historical data) split into training and testing, an XGBoost regression model [15] is created and fed with the training data, which

represent a correct functioning of each element under study. The parameters of the model (number of gradient boosting trees, maximum tree depth for base learners, boosting learning rate) are tweaked with different values in order to identify those that provide a more refined model. The testing data set is used in each case to validate the data and residuals (differences between the predicted temperature data and the real temperature values) are calculated in order to evaluate each case.

4. **Alarm classification and generation**: Once the model is defined and trained, it is compared with real-time measurements. Thresholds are established to raise alarms when the difference trespasses them. Different alarm levels are defined depending on the magnitude of the discrepancy, and they are used to build a health index to diagnose the status of the installation.

4.2 USER PROFILING

4.2.1 Flexibility assets in the STREAM environment

The Spanish pilot is located in Crevillent, a municipality of 29,000 inhabitants located in the south of the province of Alicante in the Valencia Region. The Electric Cooperative of Crevillent is a DSO, energy producer and retailer that provides electricity to 14,000 consumers (13,047 households and 1,268 companies) in low voltage network and 30 consumers in medium voltage network (mainly industrial and service sector companies), with an annual total demand of 90 GWh. The entire energy production of the entity has zero emissions and is working hard to get all the generated and distributed energy 100% clean. The company is the first Renewable Energy Community (REC) in Spain.

In the framework of STREAM, the main flexible assets have been incorporated into the different tools implemented in the pilot (mainly sFLEX, with communication with the rest) for Congestion Management purposes are listed in Table 22 below.

Table 22: Overview of the energy assets in the Spanish pilot.

Asset	Nominal power	Capacity	DR type
Community PV and battery	120 kWp	240 kWh	Congestion Management
Community buildings PV installations	465 kWp	-	Congestion Management
EV chargers	4 x 7 kW 1 x 11 kW	-	Congestion Management
10 residential users with rooftop PV and batteries	43 kWp (total)	90 kWh	Congestion Management

From all flexible assets listed above, the community installation of PV production and storage in the low voltage network is the main industrial flexibility asset and will be used to describe the profiling performed by sFLEX.

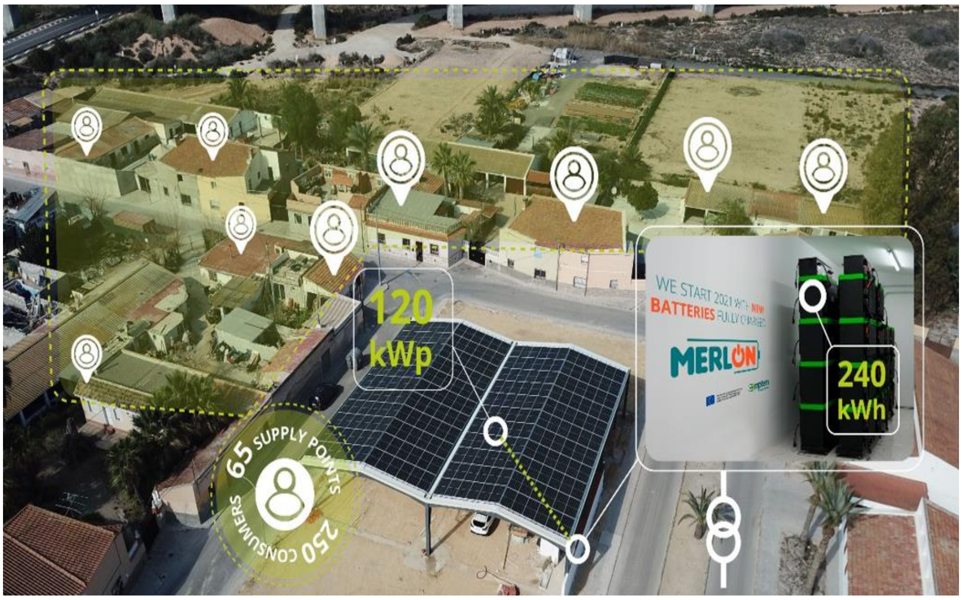


Figure 29: PV plant and community Battery in the Spanish pilot.

4.2.2 Baseline forecast

Figure 30 shows the interconnection schema of the different elements of the installation, including:

- Microgrid controller (Power Management System)
- Batteries charger (Norvento Gridmaster Converter [23])
- Battery Management System (Cegasa eBick [24])
- PV inverter (Huawei Sun2000 [25])
- Network connection point

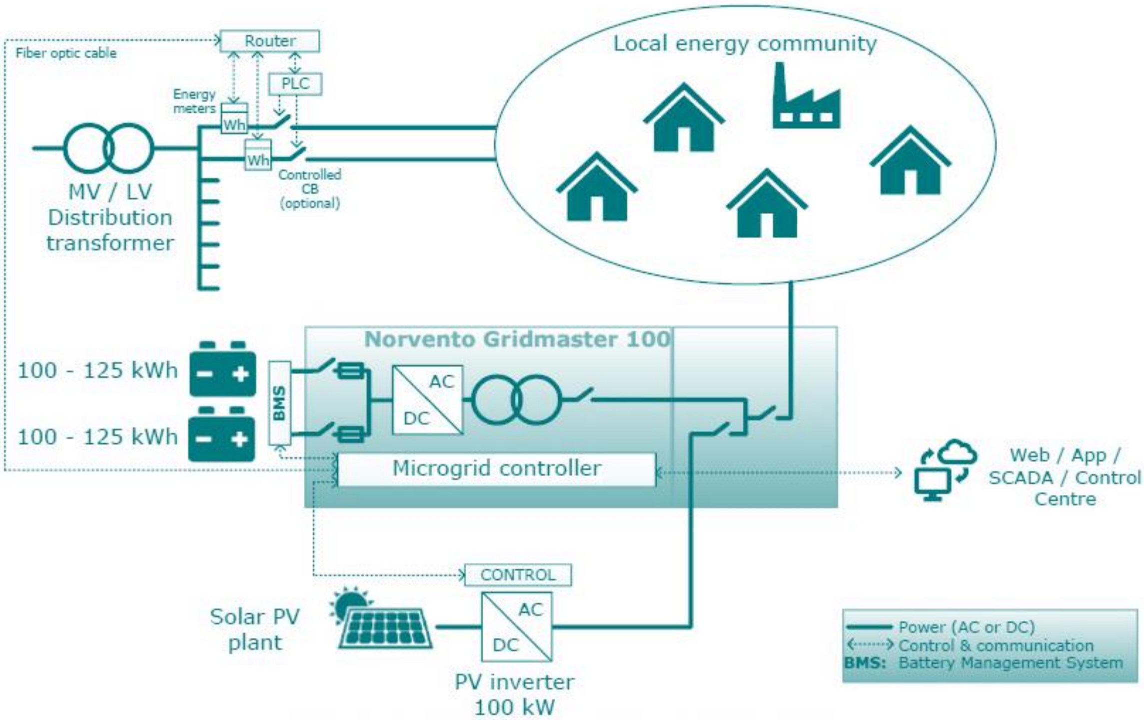


Figure 30: Interconnection schema for the community battery installation in Crevillent.

All the elements are managed by the microgrid controller, which operates the installation connected to the grid and provides a control system to define the operation of the plant. This control system was initially integrated using Modbus TCP, but this method resulted in collisions with another service used by ENERCOOP to periodically read and store measurements in a local database. An agreement was reached to replace the initial Modbus integration with a RESTful API that serves the records stored in their own repository, which is where the data that eventually arrives to sFLEX comes from. The API is prepared to receive active power setpoints for control of the battery.

From all measurements gathered, sFLEX takes into account the active power from three key elements: grid connection point, PV inverter, and Battery Management System (BMS). The total demand is calculated as:

$$Demand(t) = Grid(t) + PV(t) + BMS(t)$$

Where:

- *Demand(t)* = Total demand at time *t*
- *Grid(t)* = Grid consumption (*Grid(t) > 0*) or export (*Grid(t) < 0*) at time *t*
- *PV(t)* = PV production at time *t*
- *BMS(t)* = BMS charge (*BMS(t)* < 0) or discharge (*BMS(t)* > 0) at time *t*

An example of daily measures, both read and calculated, is provided in Figure 31.

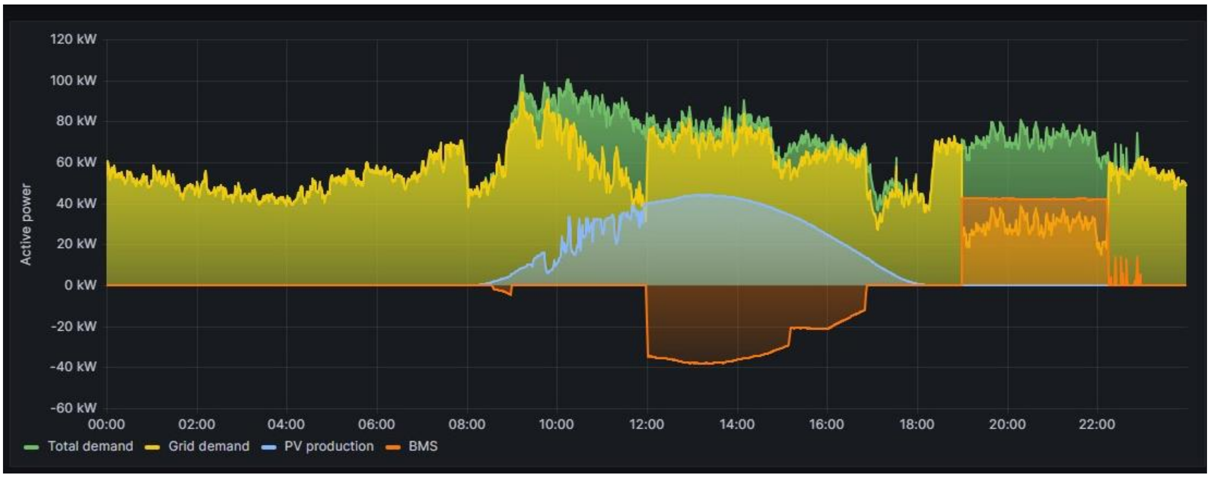


Figure 31: Daily active power of the key elements of the community battery installation in Crevillent.

A forecast of demand and production for the next 24 to 48 hours is calculated next. The forecasting is generated using the eXtreme Gradient Boosting (XGBoost [12]) library, and the models take into account:

- The historical records of the series to forecast (demand, production),
- The forecasted temperature and solar radiation (taken from Open-Meteo [26]), and
- The day of the week and type of day (for demand only).

The models are trained every day with new data and forecasts generated every 15 minutes.

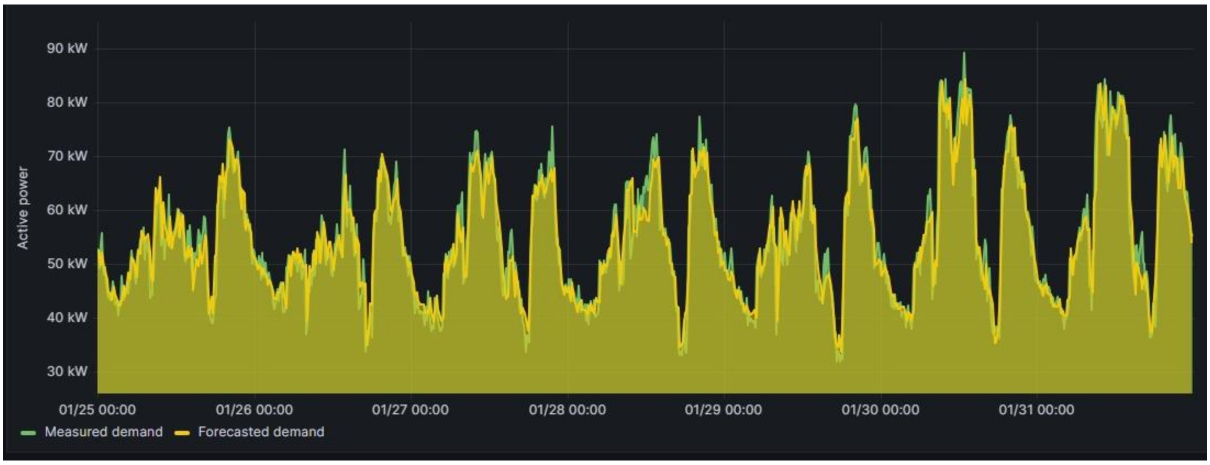


Figure 32: Measured vs. forecasted demand of the community battery installation in Crevillent.

Figure 33: Measured vs. forecasted generation of the community battery installation in Crevillent.

The calculated wMAPE for an average week taken into study (Figure 32 and Figure 33) are:

- wMAPE (demand): 5,63%
- wMAPE (generation): 23,24%

4.2.3 Flexibility forecast

An internal optimization is performed before calculating the available flexibility of the installation. This optimization process considers:

- The duration of each slot of time (minutes),
- The time horizon for the optimization (minutes),
- The battery capacity and state of charge (kWh),
- The maximum charge and discharge power of the battery (kW),
- The maximum and minimum state of charge allowed (%),
- The battery schedule (planned setpoint) per time slot (kW),
- The forecasted demand and production per time slot (kWh),
- The maximum and minimum power at supply point (kW),
- The price of the energy per time slot (in currency units), and
- The flexibility already allocated per time slot (kWh).

These variables are considered in a linear optimization problem, which is modelled using the PuLP library [27] and solved with Cbc (*C*oin-or *b*ranch and *c*ut) [28]. The optimization calculates the upper and lower flexibility (kWh) for the considered time horizon.

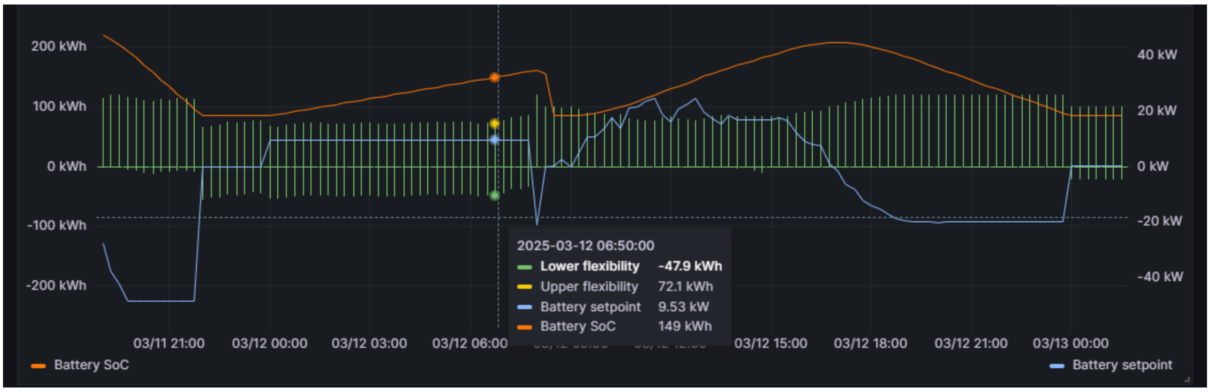


Figure 34: Calculated upper and lower flexibility of the community battery installation in Crevillent.

4.3 SGRID IN THE SPANISH PILOT SITE

4.3.1 Overview and architecture

sGRID has been developed at the Spanish pilot site to equip the DSO with the necessary tools to:

- Provide real time and historical graphical visualizations of the grid's state, including consumption, RES production, imported power at the header substation, and power flows in the feeders,
- Deliver analytics, such as power flow analysis and grid parameter forecasting, to detect and predict congestions and voltage deviations,
- Offer a predictive maintenance and fault detection module to identify and log grid incidents, and
- Supply the necessary tools to address or mitigate detected congestions or voltage deviations by sending requirements to the LFM through sMART.

The architecture of the sGRID tool at the Spanish pilot site is shown below:

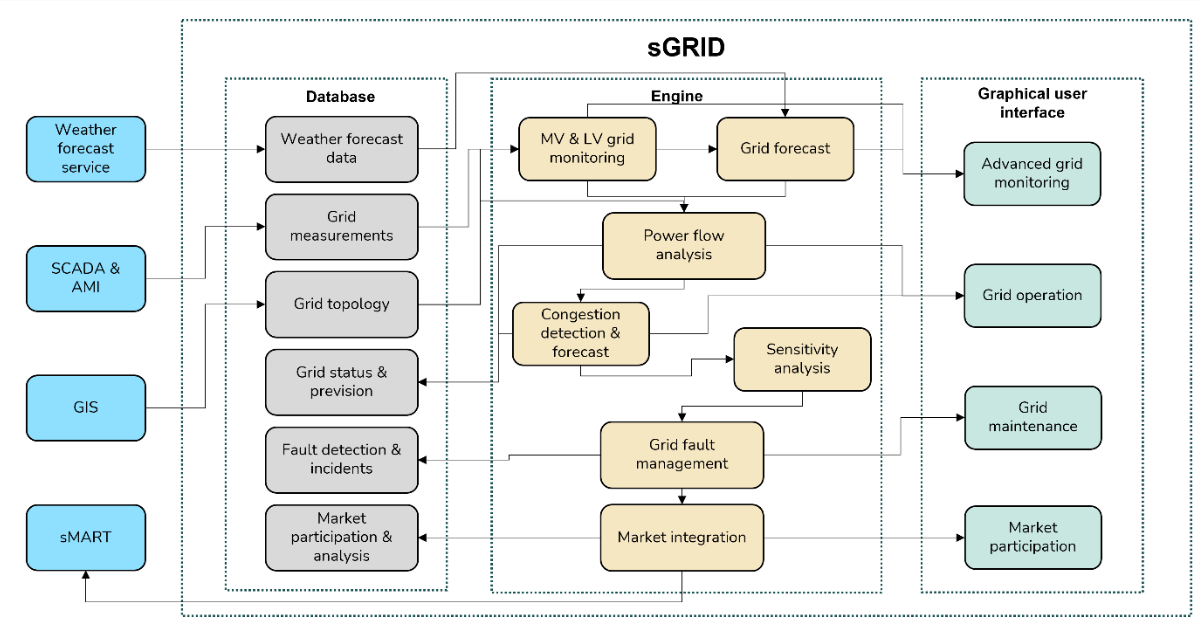


Figure 35: sGRID architecture in the Spanish pilot site.

The tool builds upon the GRIDFLEX tool developed within the X-FLEX project [29], evolving it to include more advanced features such as:

- Grid forecast service updated from Prophet library to XGBoost.
- Power flow analysis updated from PowSyBl to PandaPower library.
- Market integration with OMIE platform (sMART).

The different functionalities of sGRID are detailed in the following sections.

4.3.2 sGRID functionalities

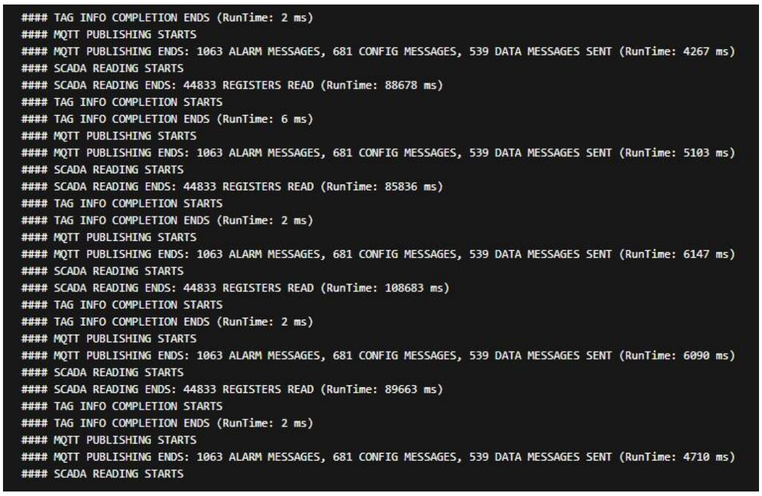
4.3.2.1 MV & LV grid monitoring

The following system have been integrated into sGRID in the Spanish pilot for grid monitoring:

- GIS: Provides the topological information of the MV and LV network. A dedicated RESTful API has been created by Enercoop for ETRA to gather the updated definition of the grid, which includes:
    - ~25,000 buses
    - ~25,000 line segments
- 230 transformers
    - ~5,300 switches
- ~15,000 loads
- 166 generators

  By the time of submission of this deliverable, the service that updates the information is run manually and generates the whole MV/LV grid model. It is planned to automatize the process in order to read the new information daily and update just the changes from the previous version.

  SCADA: ~45,000 registers from the MV and LV network are read every 5 minutes using a SOAP web service and sent to sGRID through MQTT for grid monitoring purposes. The information includes measurements, statuses, and alarms from key monitoring points of the grid.



```
#### TAG INFO COMPLETION ENDS (RunTime: 2 ms)
#### MQTT PUBLISHING STARTS
#### MQTT PUBLISHING ENDS: 1063 ALARM MESSAGES, 681 CONFIG MESSAGES, 539 DATA MESSAGES SENT (RunTime: 4267 ms)
#### SCADA READING STARTS
#### SCADA READING ENDS: 44833 REGISTERS READ (RunTime: 88678 ms)
#### TAG INFO COMPLETION STARTS
#### TAG INFO COMPLETION ENDS (RunTime: 6 ms)
#### MQTT PUBLISHING STARTS
#### MQTT PUBLISHING ENDS: 1063 ALARM MESSAGES, 681 CONFIG MESSAGES, 539 DATA MESSAGES SENT (RunTime: 5103 ms)
#### SCADA READING STARTS
#### SCADA READING ENDS: 44833 REGISTERS READ (RunTime: 85836 ms)
#### TAG INFO COMPLETION STARTS
#### TAG INFO COMPLETION ENDS (RunTime: 2 ms)
#### MQTT PUBLISHING STARTS
#### MQTT PUBLISHING ENDS: 1063 ALARM MESSAGES, 681 CONFIG MESSAGES, 539 DATA MESSAGES SENT (RunTime: 6147 ms)
#### SCADA READING STARTS
#### SCADA READING ENDS: 44833 REGISTERS READ (RunTime: 108683 ms)
#### TAG INFO COMPLETION STARTS
#### TAG INFO COMPLETION ENDS (RunTime: 2 ms)
#### MQTT PUBLISHING STARTS
#### MQTT PUBLISHING ENDS: 1063 ALARM MESSAGES, 681 CONFIG MESSAGES, 539 DATA MESSAGES SENT (RunTime: 6090 ms)
#### SCADA READING STARTS
#### SCADA READING ENDS: 44833 REGISTERS READ (RunTime: 89663 ms)
#### TAG INFO COMPLETION STARTS
#### TAG INFO COMPLETION ENDS (RunTime: 2 ms)
#### MQTT PUBLISHING STARTS
#### MQTT PUBLISHING ENDS: 1063 ALARM MESSAGES, 681 CONFIG MESSAGES, 539 DATA MESSAGES SENT (RunTime: 4710 ms)
#### SCADA READING STARTS
```

Figure 36: Excerpt of the logs from the service reading Enercoop's SCADA.

- Advanced Metering System (AMI): ~15,000 metering points from Enercoop's AMI are read once a day from a dedicated FTP and sent to sGRID via AMQP. Hourly imported and exported data from each meter is sent to sGRID in order to have the most up-to-date status of the loads in each LV feeder. This allows sGRID to estimate the demand when e.g. calculating the power flows in each branch of the network.

```
> [ct.071.alarico] Processing file ZIV0006842264_2FE9CD44_S02_0_20250204115030
Processing CT ct.073.la.ceramica
> [ct.015.paseo] Processing file ZIV0006824605_2FE9CD96_S02_0_20250204100412
> [ct.015.paseo] Processing file ZIV0006824605_2FE9CDB5_S02_0_20250204101433
> [ct.015.paseo] Processing file ZIV0006824605_2FE9CDDA_S02_0_20250204110047
> [ct.015.paseo] Processing file ZIV0006824605_2FE9CDDB_S02_0_20250204112243
> [ct.015.paseo] Processing file ZIV0006824605_2FE9CE2F_S02_0_20250204120117
Processing CT ct.137.ilimar
Processing CT ct.016.balmes.conc1
> [ct.037.pavos] Processing file SAG0179001015_2FE9CD94_S02_0_20250204100334
> [ct.037.pavos] Processing file SAG0179001015_2FE9CE04_S02_0_20250204110545
> [ct.037.pavos] Processing file SAG0179001015_2FE9CE66_S02_0_20250204123007
Processing CT ct.099.guilabert
Processing CT ct.120.alfombras.imp
> [ct.137.ilimar] Processing file ZIV0004437782_2FE9CDB4_S02_0_20250204101319
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CC75_S02_0_20250204083447
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CCEC_S02_0_20250204090504
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CCEC_S02_0_20250204094720
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CDAC_S02_0_20250204104606
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CDAD_S02_0_20250204100845
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CDB0_S02_0_20250204113205
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CDB9_S02_0_20250204102038
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CE01_S02_0_20250204114717
> [ct.073.la.ceramica] Processing file SAG0189001334_2FE9CE62_S02_0_20250204121640
Processing CT ct.074.sanfran
> [ct.176.limonero] Processing file SAG0179001169_2FE9CCA3_S02_0_20250204073105
> [ct.176.limonero] Processing file SAG0179001169_2FE9CCF2_S02_0_20250204082201
> [ct.176.limonero] Processing file SAG0179001169_2FE9CD5B_S02_0_20250204092214
Processing CT ct.142.medida.tren
> [ct.176.limonero] Processing file SAG0179001169_2FE9CD5C_S02_0_20250204092230
> [ct.176.limonero] Processing file SAG0179001169_2FE9CD73_S02_0_20250204095919
> [ct.176.limonero] Processing file SAG0179001169_2FE9CDBB_S02_0_20250204102648
```

Figure 37: Excerpt of the logs from the service reading Enercoop's AMI.

The grid monitoring functionality is the cornerstone of sGRID, and thus, it is visible in several parts of the tool, such as:

- **Dashboard**: Provides an overview of the last 24 hours of key metrics of the grid, including the power flow at HV/MV connection point, at each main MV line, as well as the total production of the network.

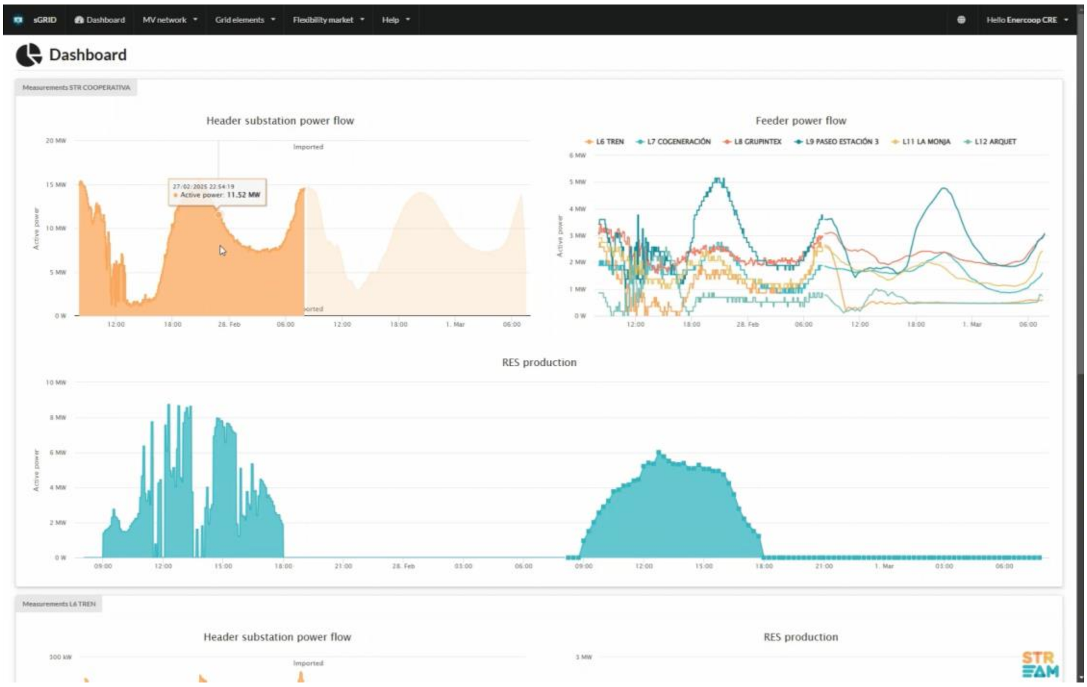


Figure 38: sGRID (ES) Dashboard.

- **Grid overview**: Allows the operator to query the information from the points included in the dashboard in historical periods. Additionally, the section includes information on national and local energy mix, as well as estimation of CO_2 emissions.

Figure 39: sGRID (ES) Grid overview.

- **Signals**: List of all signals incorporated into sGRID from external sources (SCADA, AMI) or calculated internally. The list can be queried and filtered and provides an overview of the status of each element (type, online/offline, date of last reported measurement, etc.). The details of each signal can then be accessed, including their current and historical measurements.

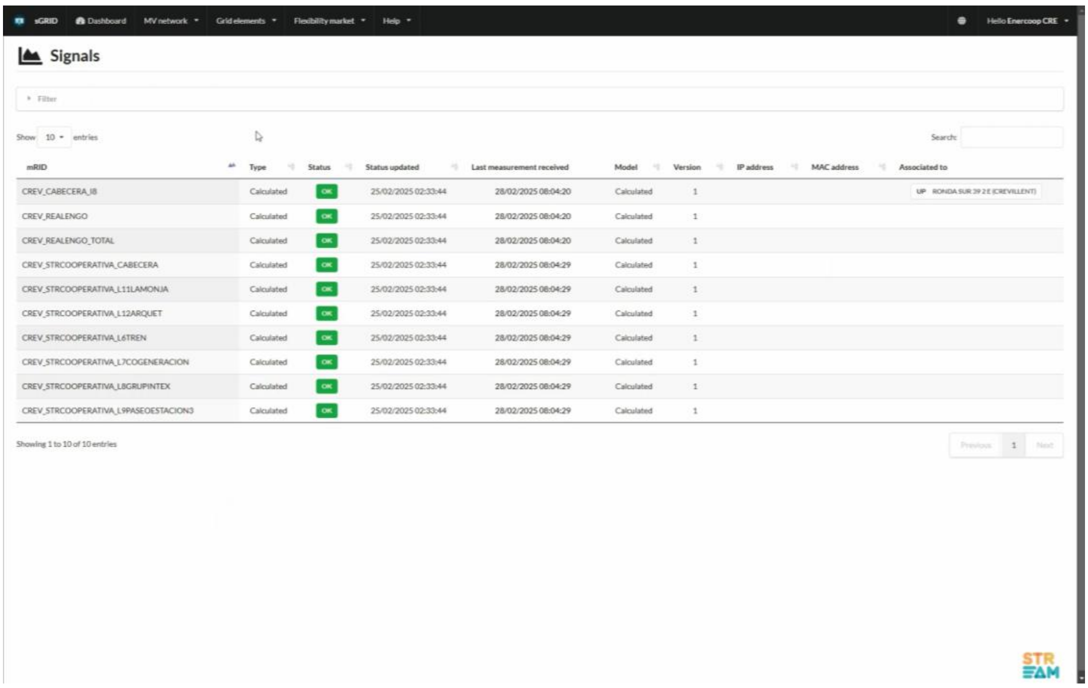


Figure 40: sGRID (ES) Signals list.

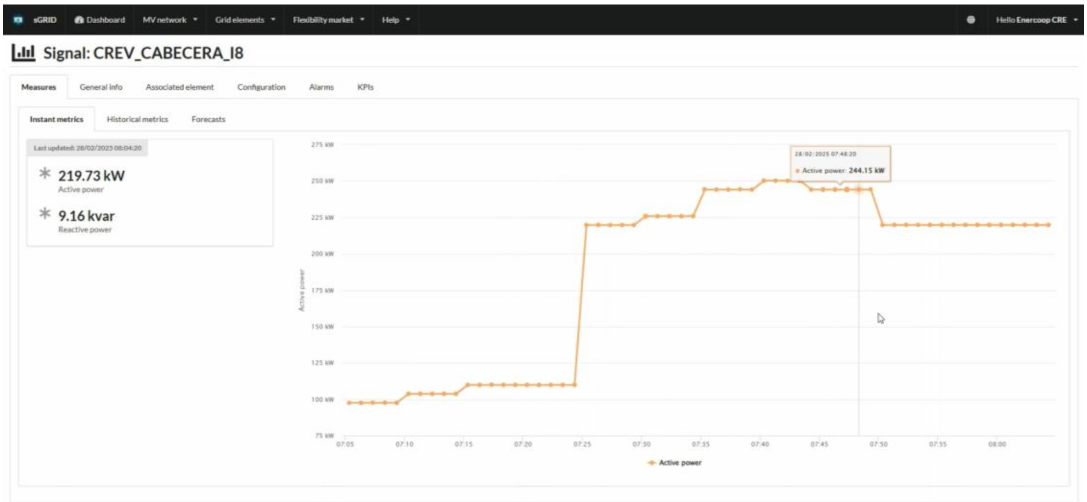


Figure 41: sGRID (ES) Signal details.

4.3.2.2 Grid forecast

A subset of the measured points of the grid is fed into the time series forecasting service (Section 4.1.1) to advance the status of the grid in the next 24 hours. The results of the forecasting service are included in different parts of the tool:

- **Dashboard**: All the graphs that conform to the dashboard of sGRID are divided into two parts: the left side shows the progression of the received measurements in the last 24 hours, while the right side shows the forecasted evolution of each specific element for the next day. Both sets of values are updated as soon as new measurements or results are available.

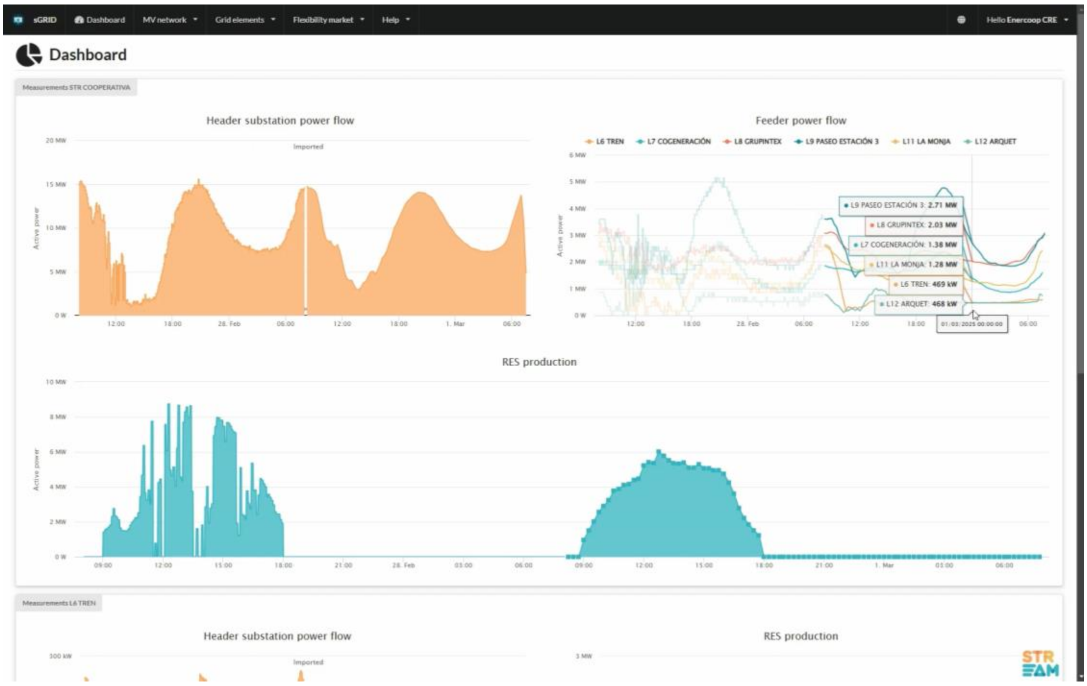


Figure 42: sGRID (ES) Dashboard, including forecasts.

- **Signals**: Besides current and historical measurements, the detailed visualization of the sGRID signals also allows the operator to query the calculated forecasts for those elements that are considered by the service.

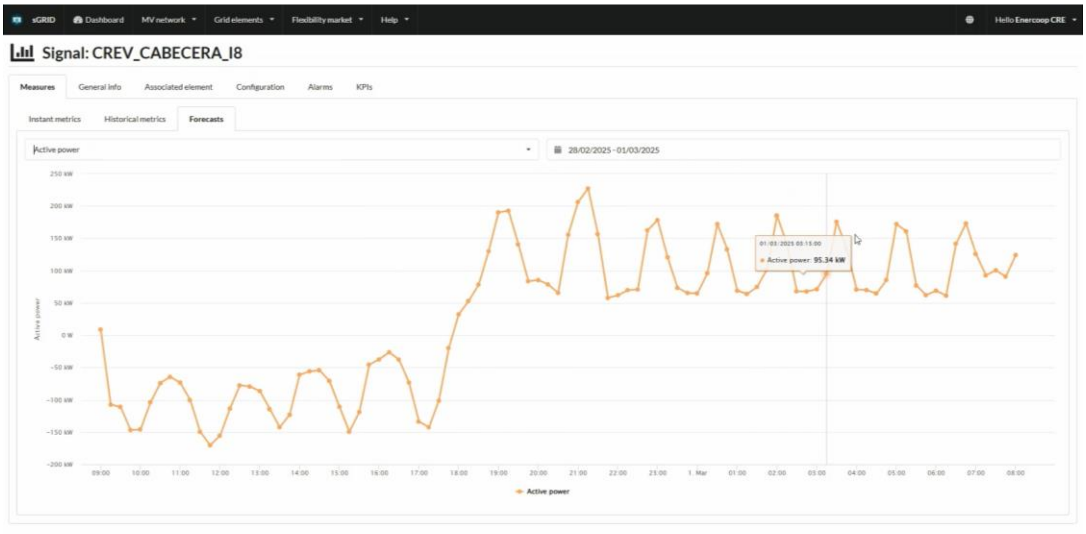


Figure 43: sGRID (ES) Signals detail, including forecasts.

4.3.2.3 Power flow analysis

The results of the different power flow calculations performed by the specific data analytic service (Section 4.1.2) are shown in a dedicated section of sGRID. The dynamic representation of the grid is created using the information from Enercoop's GIS as shown in Figure 44 and Figure 45.

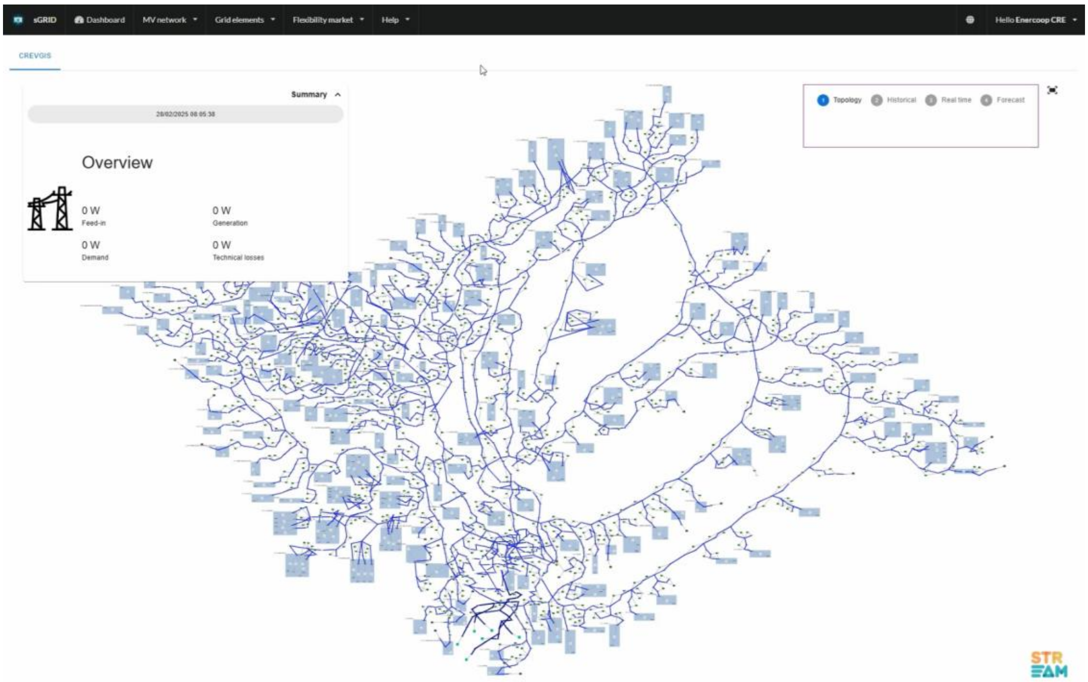


Figure 44: sGRID (ES) Crevillent MV grid representation.

Figure 45: sGRID (ES) Detail of MV topology of Crevillent.

The results of the power flow calculation generated every 15 minutes using the real-time measurements of the grid can be accessed by selecting the "Real-time" option. The colour of the lines represents their capacity status—green when there is no congestion (less than 60% of capacity), orange for high load (between 60% and 80%), and red when the line is congested (more than 80% of capacity)—, the width represents their proportional load, and the direction in which the dots move represents the flow of the power running through it.

When clicking a specific element, their static details and values (either measured or calculated) are shown in a window on the left as depicted in Figure 46 and Figure 47.

Figure 46: sGRID (ES) Dynamic representation of power flow calculation in Crevillent MV grid.

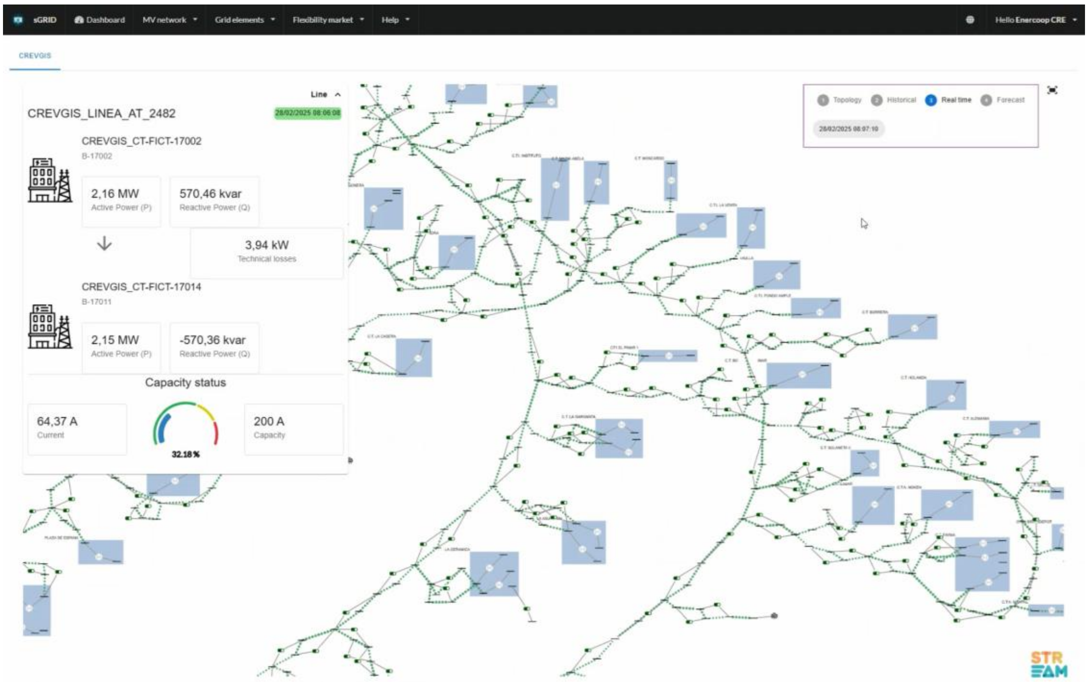


Figure 47: sGRID (ES) Power flow results of a specific MV line in Crevillent.

The results of past calculations can be queried by clicking on "Historical" and selecting the corresponding day and time. The same mechanism is used to query the power flows analyses that use the forecasted values of the grid as inputs (Figure 48).

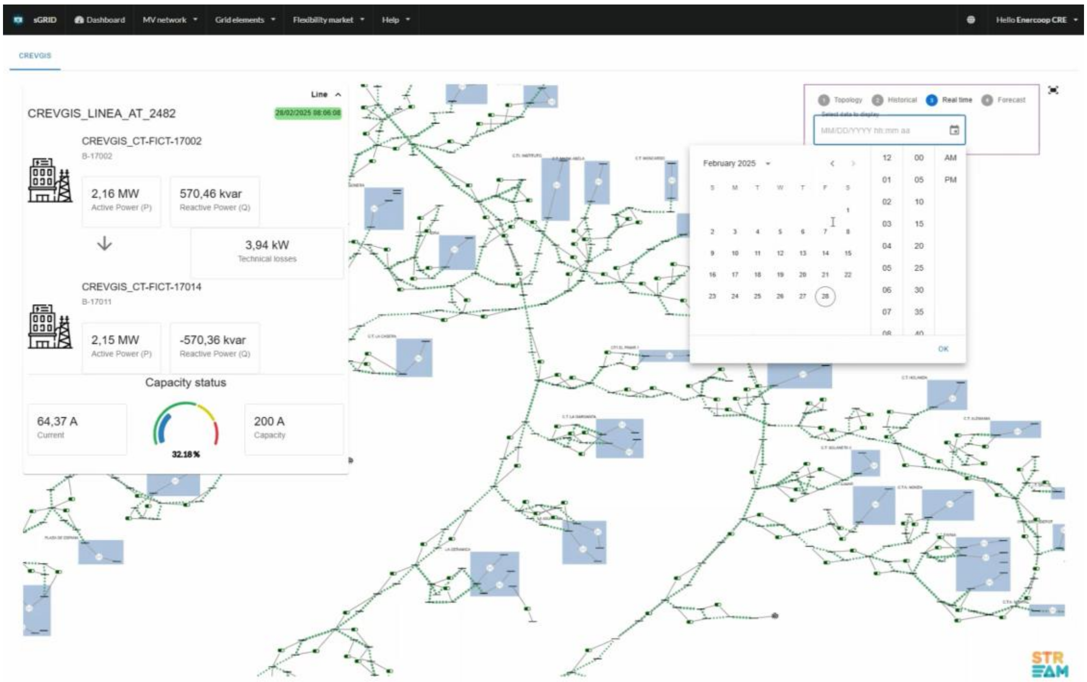


Figure 48: sGRID (ES) Historical power flow results.

Substations are represented as blue squares with one or more MV/LV transformers inside them. When clicking on the LV bus associated to a transformer, if the topological information is available, an additional button with the text "View feeder details" will appear in the detail window. When it is clicked, the representation of the topology for the LV feeder will be opened in an additional tab as shown in Figure 49 and Figure 50.

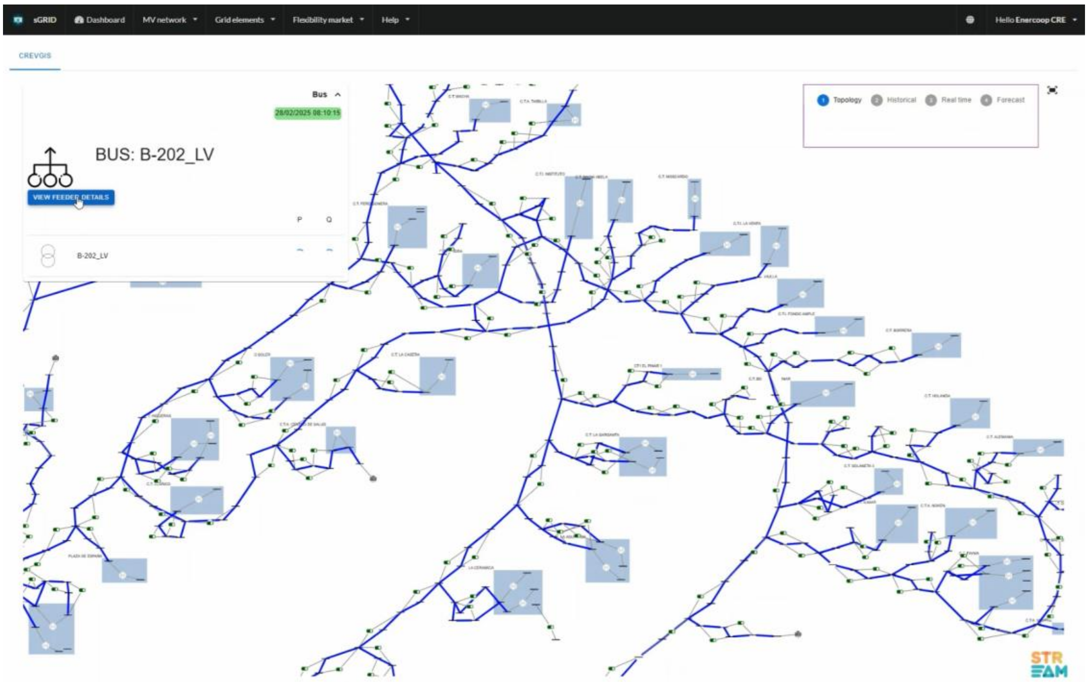


Figure 49: sGRID (ES) LV feeder topology button.

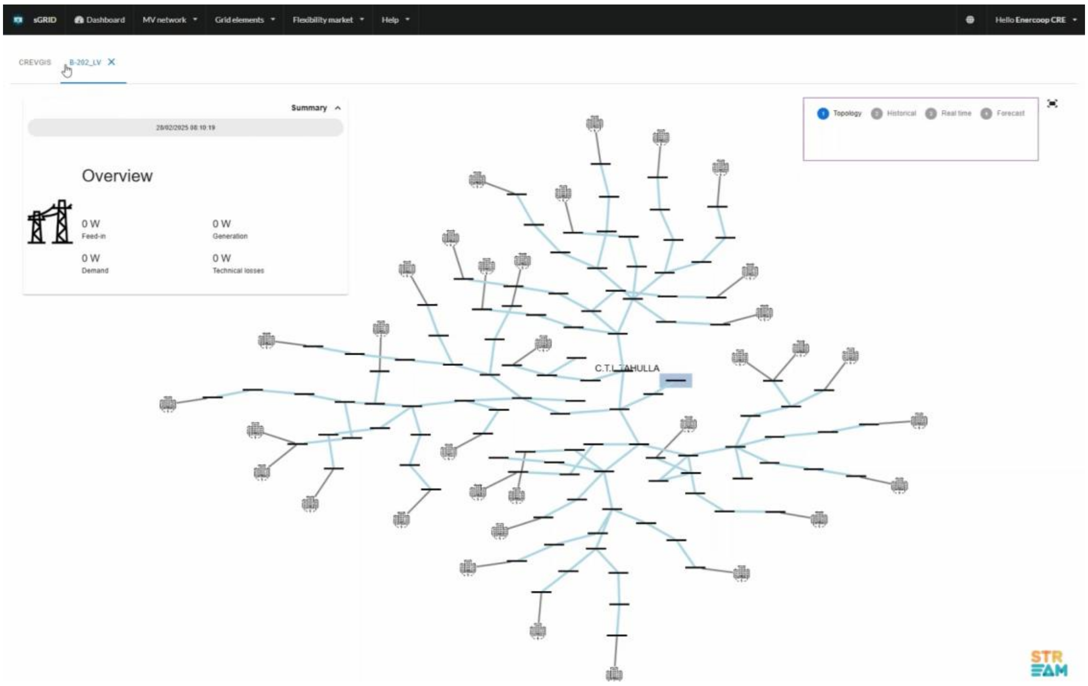


Figure 50: sGRID (ES) Example of a LV feeder in Crevillent's topology.

The same visualization options available in the MV representation of the grid (real time, historical, and forecasts) are available for each LV feeder integrated in sGRID (Figure 51).

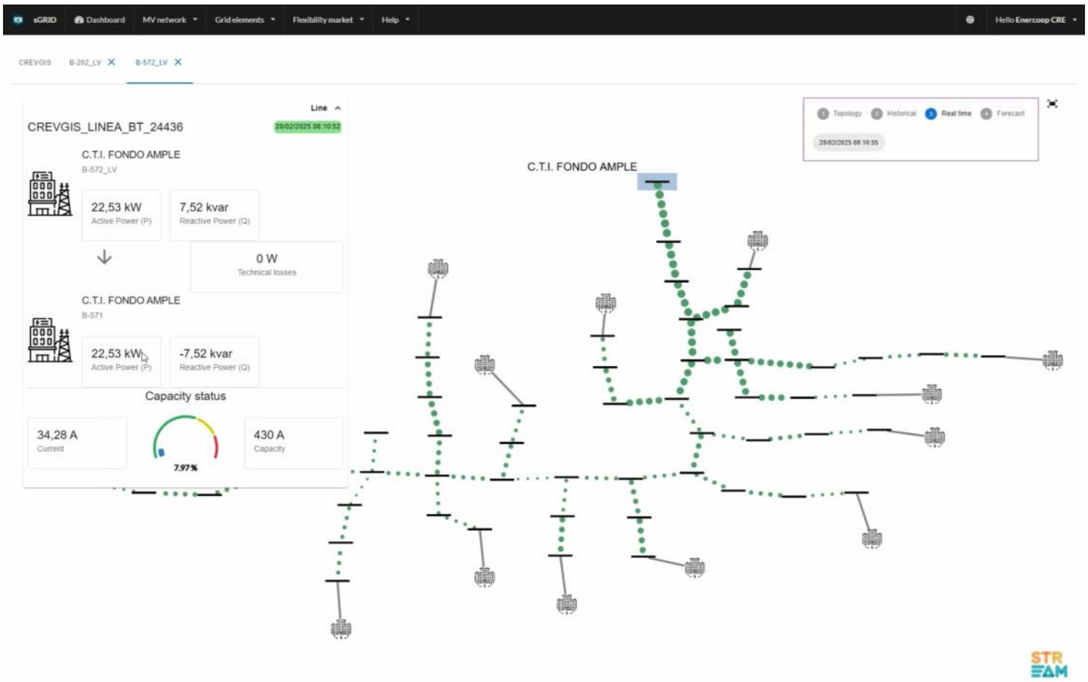


Figure 51: sGRID (ES) Example of power flow representation in a LV feeder.

4.3.2.4 Sensitivity analysis

One of the tools of sGRID upon detection of a congestion in a line is to run a sensitivity analysis to identify the flexible assets that will have greater impact in its resolution. The results of the service (as explained in Section 4.1.3) are shown in the details of the line, ranked by their sensitivity factor.

Figure 52: sGRID (ES) Detail of congested line and sensitivity analysis calculation request.

Figure 53: sGRID (ES) Detail of results from sensitivity analysis.

4.3.2.5 Grid fault management

The management of incidents and contingencies is part of the everyday operation of the grid. These types of events include:

- Detected or forecasted congestions as result of a power flow analysis,
- Reported alarms from another system (e.g. SCADA),
- Communication failures of any integrated system, and
- Any other potential type of reported event that may potentially affect the quality of service (e.g. reported by a field worker or a client, as a result of the analysis of a specific asset, etc.).

All these types of events are compiled in a dedicated section of sGRID, where they are listed and can be queried and filtered. Moreover, specific details are provided for each of them, including:

- Type of event,
- Location,
- Start and end times,
- Current status,
- Severity,
- Affected element(s) of the grid,
- Agent that reported the issue, and
- Any other specifics of the event (depending on its type).

sGRID allows the operator to acknowledge the issue and modify its progress as it is being worked on.

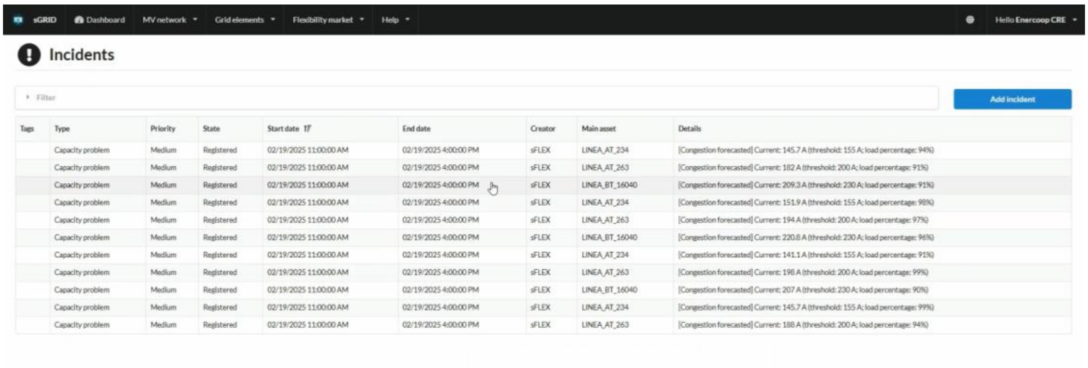


Figure 54: sGRID (ES) List of detected incidents.

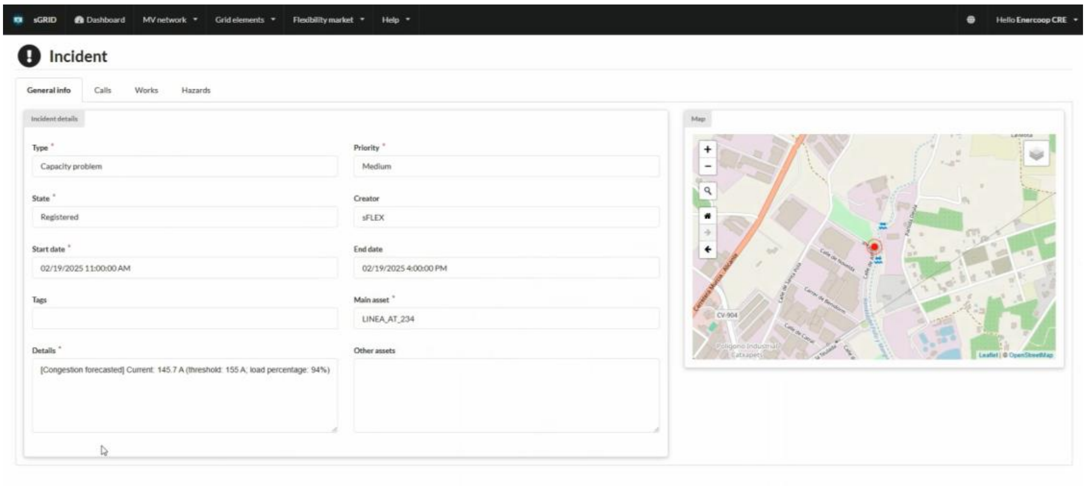


Figure 55: sGRID (ES) Detail of a detected incident in the grid.

4.3.2.6 Market integration

The detection of a contingency in the grid might be solved by means of accessing the sMART to request flexibility from the available agents and flexibility sources. sGRID integrates the information from sMART in a dedicated section, with split subsections according to the specific market.

A section for the long-term flexibility market lists the information of the sessions currently open in sMART. Information includes the session identifier, the zone, the type of product (availability or activation), the required capacity and direction, the start and end dates of the service, and the days of the week the requirements may take place.

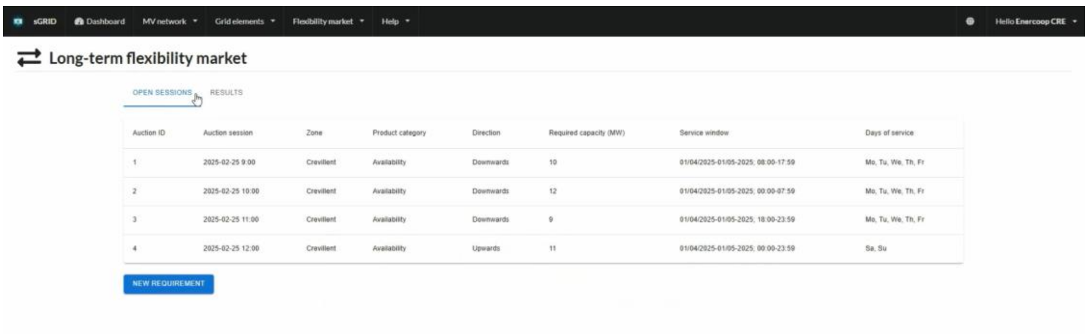


Figure 56: sGRID (ES) Long-term flexibility market – List of open sessions.

As currently the information from the long-term market in sMART cannot be retrieved in an automated way, a form is included to incorporate this information into sGRID for the operator to be aware. Further iterations of this functionality will attempt to at least being able to export the information from sMART in CSV format and import it into sGRID with a semi-automatic process.

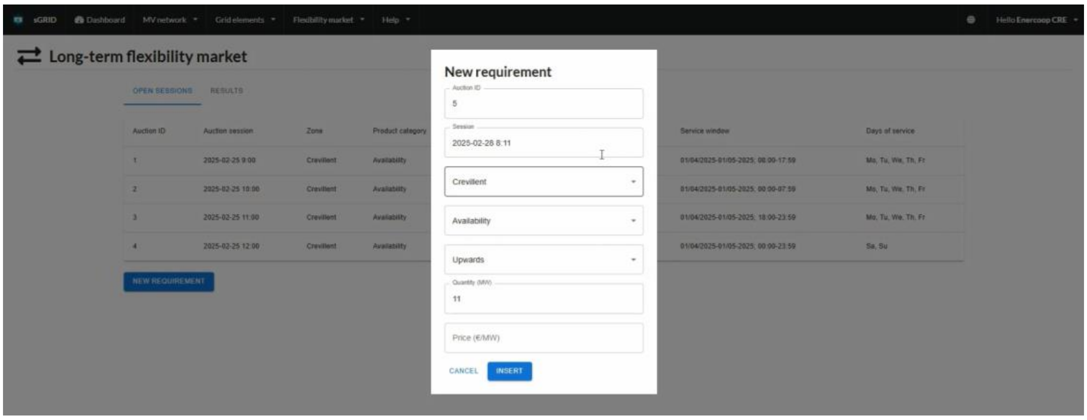


Figure 57: sGRID (ES) Long-term flexibility market – New requirement form.

A second tab in the section lists the results of the sessions already closed. Included the required vs. matched flexibility. Furthers details for each session can be queried, including the agent, units, and prices that compose the market results.

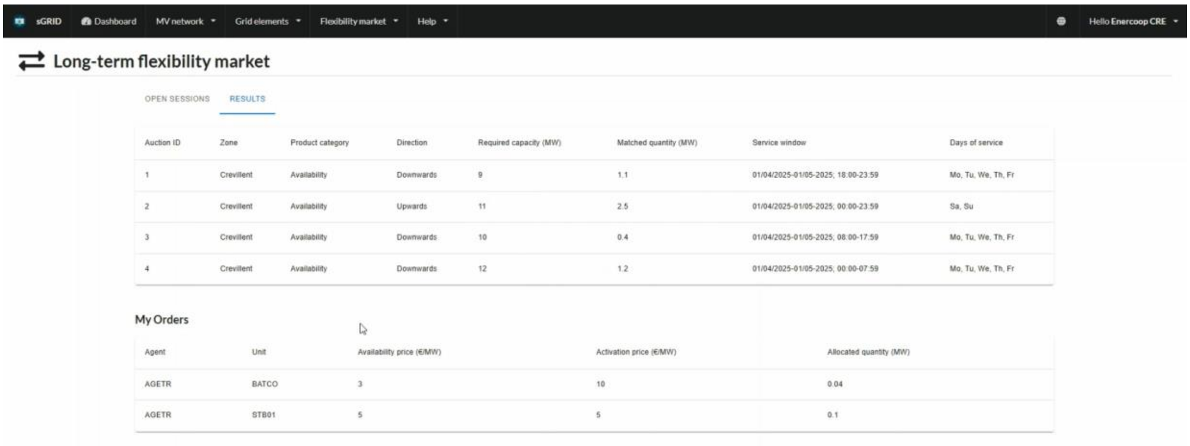


Figure 58: sGRID (ES) Long-term flexibility market – List of results and details.

An analogous section is available for sMART's short-term flexibility market. In this case, the information is provided by the sMART connector, which integrates the APIs to communicate with sMART automatically (as explained in D4.2 "Local Markets structure development", Section 4.2 [3]).

For each session, the date of generation, zone and period of application, quantity and direction of the flexibility, closure date, and status of the requirement are provided. Taking into account this information and the status of the grid as provided by sGRID, the grid operator can validate or reject each requirement in this section. This information in shared with sMART for coordination between market and grid.

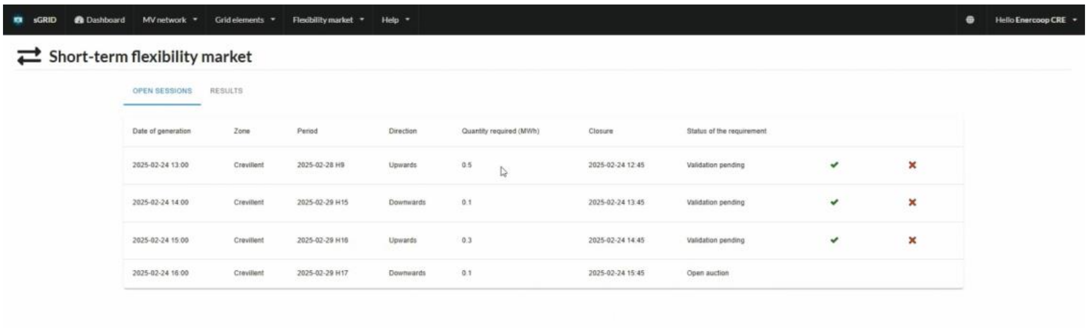


Figure 59: sGRID (ES) Short-term flexibility market – List of open sessions.

In a second tab, the results of the short-term market are shown, including the quantity required, matched, and the average price of the energy for each session. Additional details for each unit can be queried, including their allocated quantity and the prices of availability and activation.

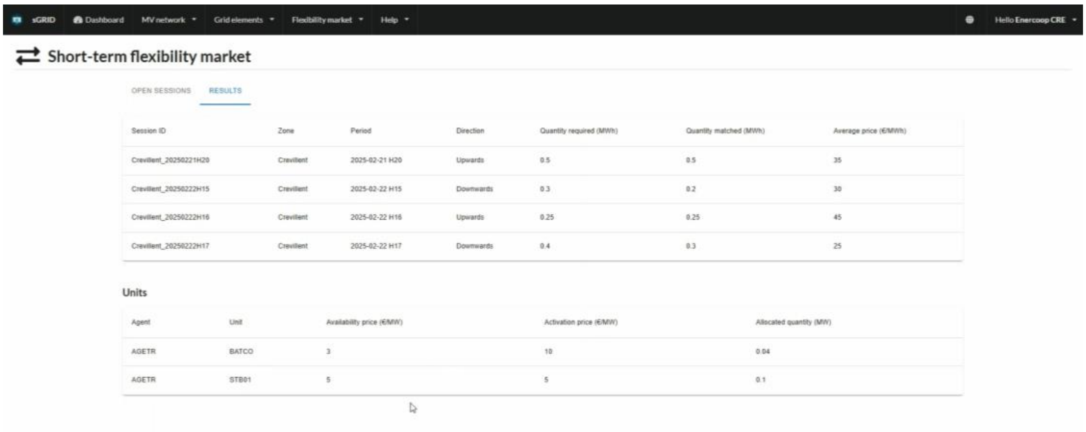


Figure 60: sGRID (ES) Short-term flexibility market – List of results and details.

4.3.2.7 Graphical user interface

The user interface of sGRID in the Spanish pilot is a web application built in Meteor [30], the same JavaScript framework used for sENC and sFLEX. The front-end uses a mixture of BlazeJS [31] and React [32] to develop each component, as well as a set of NPM libraries [33] for specific functionalities. Real-time data is stored a MongoDB repository [34], while time series and historical records are managed by InfluxDB [35]. Messages from external sources and systems come mainly through RabbitMQ [36], while communication with back-end services and other STREAM tools is typically performed using a NATS broker [37].

Access to the tool is provided with a username and password. sGRID is integrated with Keycloak [38], which manages the users, permissions, and access rights, is compliant with OpenID [39] and OAuth 2.0 [40], and additional functionalities such as single sign-on over the STREAM ecosystem of tools in the pilots. sGRID is a multitenant application that support several installations and domains of data within a single deployment, allowing its commercialization as Software-as-a-Service (SaaS).

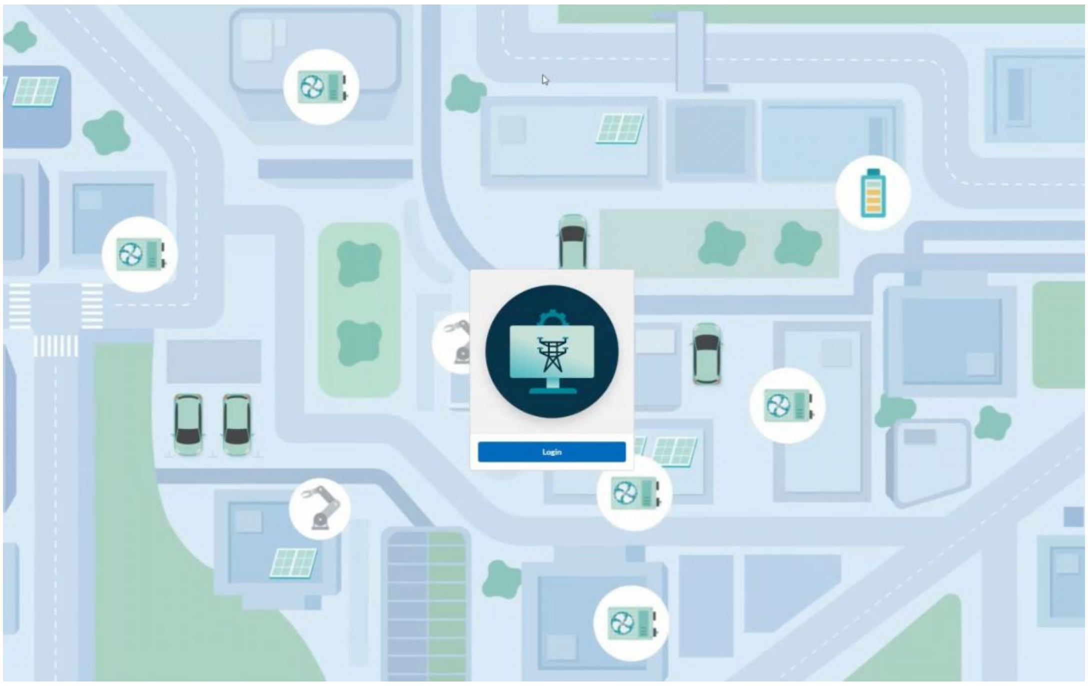


Figure 61: sGRID (ES) Log in.

Although most of the GUI of sGRID has been presented in previous chapters, additional views and sections of the tools are listed below:

- **Map**: Geographical visualization of the grid, including substations, MV & LV lines, usage points and generators. Additional details of each element are shown when the user moves over or click them.

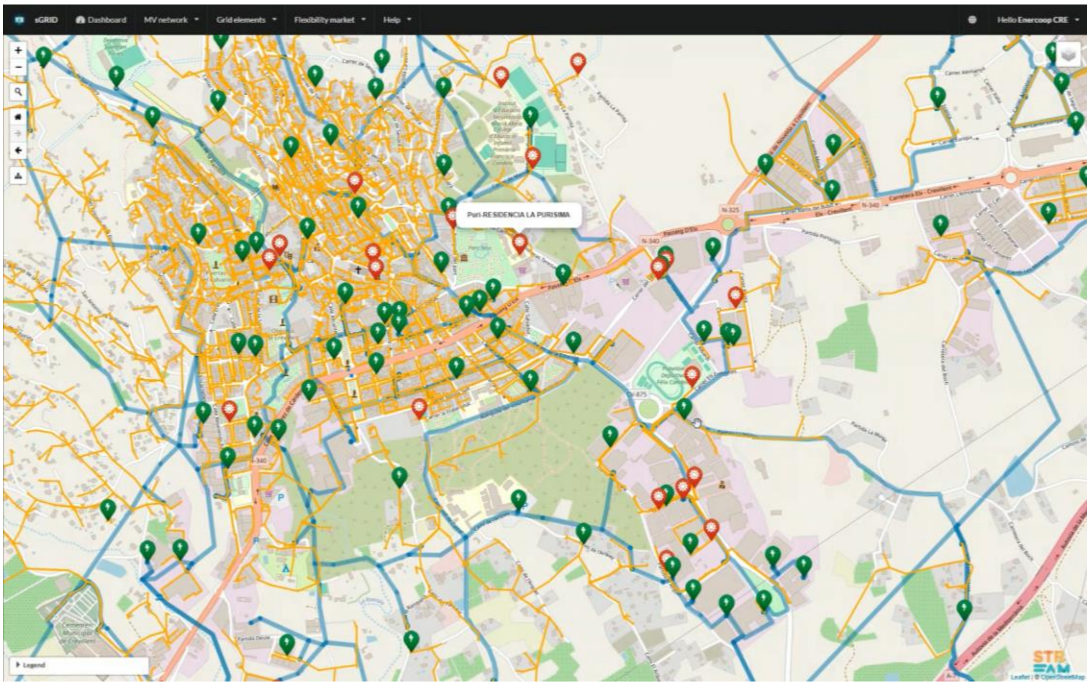


Figure 62: sGRID (ES) Map.

- **Inventory**: The different types of elements that conform the topology of the grid can be queried and their details accessed in dedicated sections of the tool. For each element, their related assets are listed and linked to foster navigation.

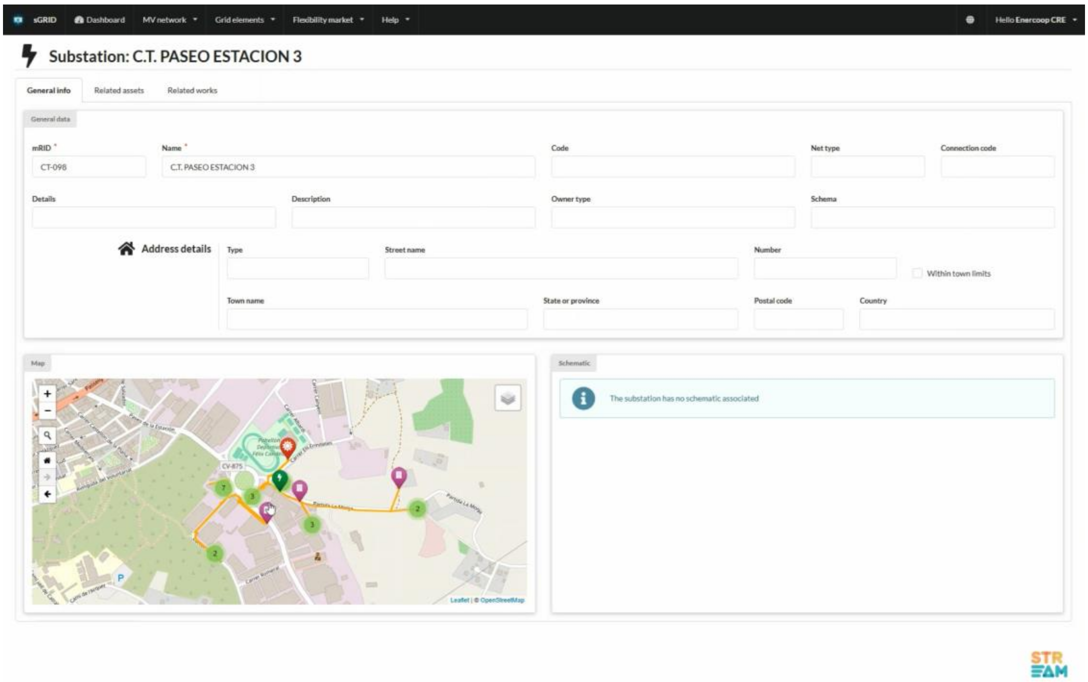


Figure 63: sGRID (ES) Substation details.

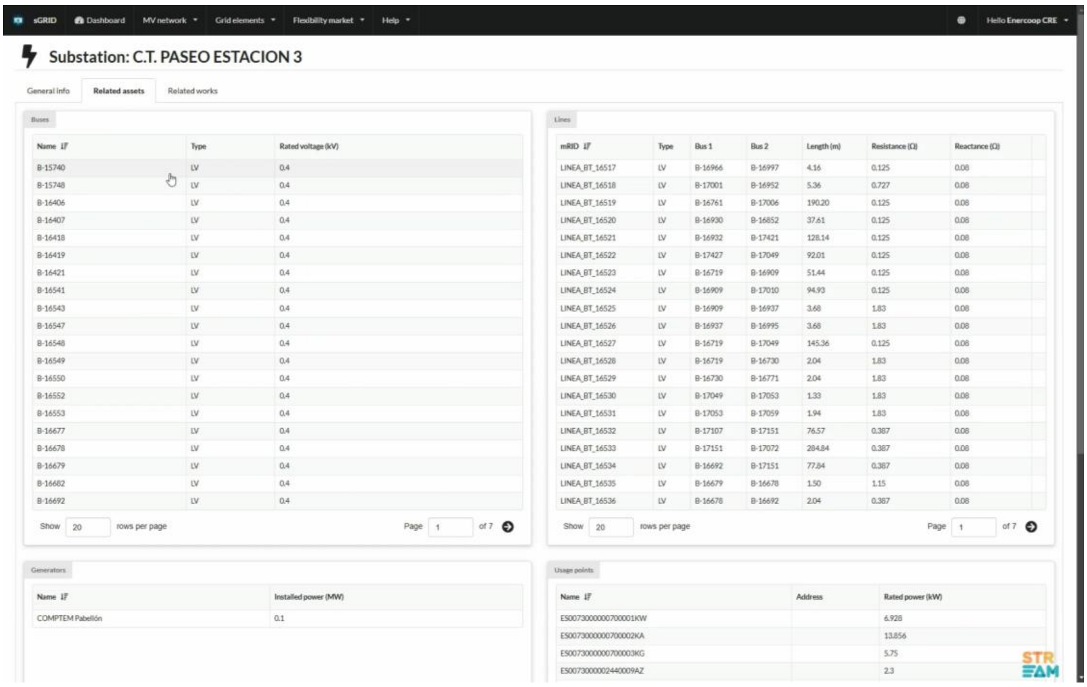


Figure 64: sGRID (ES) Lists of assets associated to the selected substation.

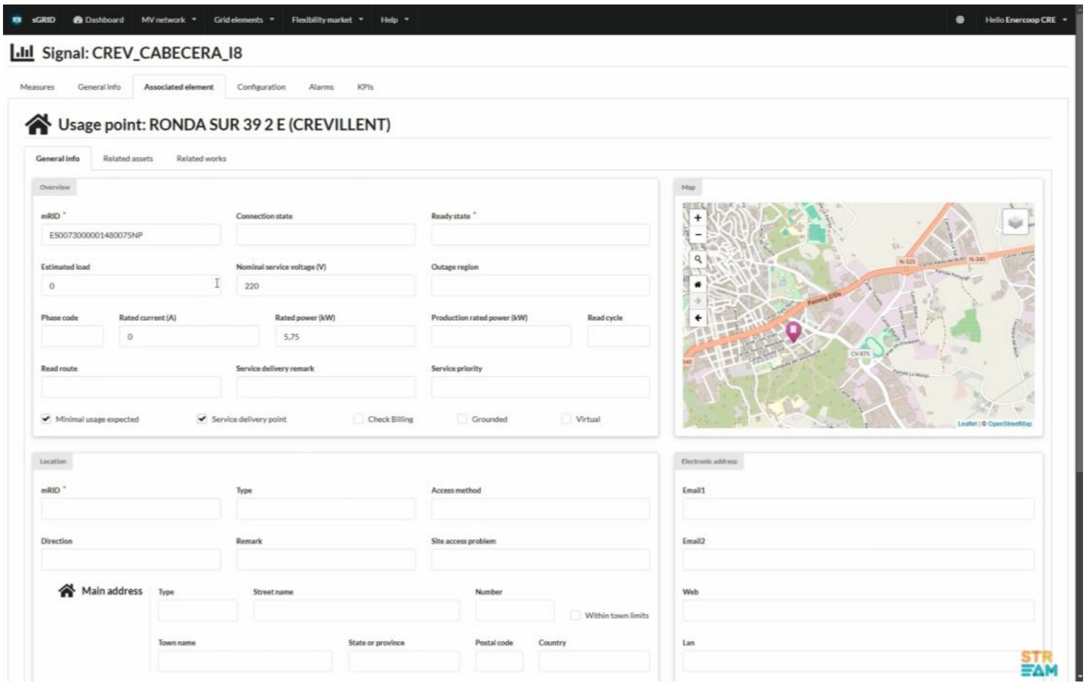


Figure 65: sGRID (ES) Usage point details.

5 ITALIAN PILOT SITE

5.1 DATA ANALYTICS

Data analytics is becoming vital in power systems, enabling enhanced operational efficiency, increased reliability and drive the advancement of smart grid technologies. As power grids face growing complexities, due to factors such as renewable energy integration, fluctuating demand and changing consumption patterns, utilities must leverage advanced analytical techniques to overcome these challenges. Key applications of data analytics in the Italian pilot include time series forecasting, baseline calculation, grid reconstruction, load disaggregation and multi-objective optimization.

The following table summarizes the models that are used in the tools, identifying the tool in which the analytics are used:

Table 23: Data analytics models in the Italian pilot site.

Data analytics model	Involved tool(s)
Time series forecasting	sGRID, sFLEX
Grid reconstruction	sGRID
Load disaggregation	sFLEX
Baseline calculation	sFLEX
Multi-objective optimization	sENC

5.1.1 Time series forecasting

5.1.1.1 Introduction

Time series forecasting plays a crucial role in optimizing the performance of IT systems by enabling accurate predictions of future behaviour. In the context of our pilot, it is used to forecast medium voltage transformer congestion, PV generation and water pump consumption for the day ahead. By analysing historical data, these forecasts help to understand patterns and dependencies, enabling more efficient management of resources. The predicted results support decision-making, help with capacity planning, and provide valuable insights for smooth and reliable operation of the system.

5.1.1.2 Input datasets

The following datasets will be used in each tool:

Table 24: Input datasets for time series forecasting in IT pilot.

Data element	Units	Tool(s)	Comments
Active and reactive power, current of transformer	kW, kVAr, A	sGRID	Historical data (1 year)
Active power of PV	kW	sENC	Historical data (1 year)
Water pump active and reactive power consumption	kW, kVAr	sFLEX	Historical data (1 year)

Additionally, the following parameters will be gathered for seasonality characterisation:

Table 25: Seasonality data for time series forecasting in IT pilot.

Data element	Units	Comments
Temperature	°C	
Precipitation	mm	
Solar irradiation	W/m^2	
Type of day	-	Day and month, day of the week, working days, weekends, holidays

5.1.1.3 Data processing and analysis

The time series forecasting tool is developed in Python using the XGBoost library. It is designed to retrieve and process training and regressor data from various sources to improve forecast accuracy. A forecasting model is structured through a series of data retrieval steps, where the retrieved data is pre-processed and organized as follows:

- y: Target variable representing the forecasted metric (e.g. transformer power consumption, PV generation or water consumption).
- x1, x2, …, xn: Regressors consisting of additional time series that provide a relevant forecasting context, such as temperature, precipitation, solar radiation, day type (weekend, weekday, vacation), time of day and historical lag characteristics (previous day and previous week values).

The tool integrates data from various sources, including historical weather data retrieved via Open-Meteo, while electrical and non-electrical water pump consumption, power flow from transformers and power generation from renewables are taken from sDATA. To ensure consistency, pre-processing techniques such as resampling to uniform time intervals, handling missing values and correcting timestamp inconsistencies are already applied in sDATA.

An important aspect of the model is feature engineering, where derived features such as lagged consumption values, weather forecasts and categorical encodings of time patterns are included. These processed datasets are then used to train an XGBoost regression model via the Scikit-Learn API. Predictions are made on a rolling basis for each day to produce short-term energy and water forecasts. The results are visualized through interactive graphs comparing actual to predicted values and are stored for further evaluation and refinement. Model performance is assessed using Mean Absolute

Percentage Error (MAPE) and Weighted Mean Absolute Percentage Error (wMAPE), ensuring accurate and reliable forecasts. In addition, changes in water levels in the reservoirs are estimated based on the predicted and actual water flow values, providing insight into system performance and deviations from expected behavior.

Table 26: Output datasets for time series forecasting in IT pilot.

Data element	Units	Tool(s)	Comments
Forecasted current and power consumption,	A, kVA	sGRID, sFLEX	Day ahead forecast
Forecasted RES generation	kWh	sENC	Day ahead forecast
Forecasted flexibility of water pumps	kWh	sFLEX	Day ahead forecast
Forecasted water consumption	l	sFLEX	Day ahead forecast

5.1.2 Grid reconstruction

5.1.2.1 Introduction

Grid reconstruction is a key component of sGRID, which is responsible for deriving the impedance matrix of the power grid from the measurements of the power flows and the voltages at the MV/LV transformers. This matrix is then used to compare the reconstructed grid parameters with the expected line parameters, which helps to identify potential anomalies in the line calculations or to detect incorrect measurements of transformers. The model relies on time-aligned data and only considers time stamps when all required measurements are available. This ensures the consistency of the analysis and supports the validation of grid parameters for more reliable operation.

5.1.2.2 Input datasets

The following datasets will be used:

Table 27: Input datasets for grid reconstruction in IT pilot.

Data element	Units	Tool(s)	Comments
Power and voltage of transformers	kVA, kV	sGRID	Historical data (1 year)
Voltage of substation	kV	sGRID	Historical data (1 year)
Grid data	N/A	sGRID	Electrical parameters of the lines and the topology of the grid

5.1.2.3 Data processing and analysis

The first step in data processing involves loading and structuring the measured values. Voltage and power measurements are extracted from the dataset and organized for further analysis. Given the large volume of data, all measurements are checked and reduced to intervals where all required values are available simultaneously, ensuring consistency and eliminating discrepancies due to missing data. To achieve this, an index list is created to filter out invalid measurements. The primary criterion for filtering is the availability of all required measurements at a given time step. Additionally, measurements where the voltage drops below 80% of the nominal voltage are discarded, as such drops indicate potential measurement errors. This ensures that only valid and reliable data points are used in the impedance calculation, significantly improving accuracy.

The impedance of the transmission lines is determined by minimizing the difference between the actual measured voltage drop and the assumed voltage drop, which is computed using an estimated impedance value. The actual voltage drop is calculated as the difference between the substation voltage and the measured voltage at the loads. Current values are derived from the measured power and voltage, and the assumed voltage drop is then computed as the product of the real measured current and the assumed impedance:

$$\Delta U_{calculated} = I * Z_{assumed}$$

where $Z_{assumed}$ is the variable being optimized. The calculated voltage drop is directly compared to the actual measured voltage drop, and the difference between them is minimized through an optimization process.

To ensure that a diverse range of operational conditions is considered, clustering techniques such as K-Means or Gaussian Mixture Models (GMM) are applied to categorize the data into representative clusters instead of simply sorting based on peak power. This allows for a more comprehensive estimation of impedance across different operating scenarios.

The impedance matrix $Z_{assumed}$ is structured as a symmetric matrix, and an optimization function is defined to minimize the squared difference between the measured and assumed voltage drops. A constraint is imposed to ensure that off-diagonal elements do not exceed the diagonal elements, maintaining physical feasibility. The Powell optimization method is used to iteratively refine the impedance values until convergence is achieved. By explicitly linking the assumed voltage drop to real measured current and systematically refining the impedance estimates, the approach ensures an accurate representation of the distribution network.

Table 28: Output datasets for grid reconstruction in IT pilot.

Data element	Units	Tool(s)	Comments
Impedance matrix of the grid	Ω	sGRID	Matrix reflecting the self and mutual impedances of the lines and corresponding topology

5.1.3 Load disaggregation

5.1.3.1 Introduction

The load disaggregation tool is a key component of sFLEX designed to estimate daily heating, ventilation and air conditioning (HVAC)-related energy consumption in residential apartments. It uses historical energy consumption data from sDATA and weather variables from Open-Meteo to analyze the relationship between outdoor temperature and energy consumption.

By identifying a baseline temperature range where HVAC systems are not normally required, the tool isolates heating and cooling loads from overall energy consumption. This enables the estimation of potential flexibility by determining how much HVAC energy consumption could be shifted or reduced based on temperature thresholds without compromising indoor comfort.

5.1.3.2 Input datasets

The following dataset will be used:

Table 29: Input datasets for load disaggregation in IT pilot.

Data element	Units	Tool(s)	Comments
Power and energy of end user	kW, kWh	sFLEX	Historical data (1 year)

5.1.3.3 Data processing and analysis

Data processing and analysis to assess the relationship between temperature and energy consumption follows a structured approach. First, the temperature is loaded from Open-Meteo and the power profiles of the end users from sDATA and merged by date. The energy consumption data is calculated as the integral of the power values, which are pre-cleaned by replacing missing values with zeros and filtering out inappropriately high values. Once the energy data is structured, it is adjusted to the required weather granularity (either hourly or daily), with appropriate aggregation for each period. For hourly data, the timestamps are rounded to the nearest hour; for daily data, the consumption values are summed for each day.

Data validation ensures that only valid measurements are included in the analysis. Consumption data with zero values is removed and the temperature data is checked for extreme or out-of-range values. The data is then divided into temperature categories: low (<14°C), mid (14-22°C), and high (>22°C), which allows a more detailed analysis of energy consumption at different temperature conditions. Within each temperature segment, a correlation analysis is performed to assess the relationship between temperature and energy consumption. These correlations can be used to determine how temperature affects energy consumption in different operating contexts.

The flexibility analysis calculates the difference in energy consumption between the low, medium, and high temperature segments. This quantifies the additional energy required for heating under cold conditions and cooling under hot conditions and provides information about the flexibility of the system. In addition, the energy consumption per degree Celsius is calculated by determining the slope of the relationship between temperature and energy consumption, which illustrates the sensitivity of the energy demand to temperature changes.

Several Python libraries with their specific functions were used to perform the data processing and analysis. Pandas was essential for data manipulation and cleaning, including merging datasets, handling missing values and performing aggregations based on different time granularities. NumPy has been used for numerical operations such as calculating energy consumption (integration of power values) and efficiently handling array operations. For statistical analysis and correlation assessment, SciPy and statsmodels provided functions for performing correlation analysis and calculating regression slopes. To divide the data into temperature categories and facilitate flexibility analysis, Pandas was again used for conditional filtering and grouping.

Table 30: Output datasets load disaggregation in IT pilot.

Data element	Units	Tool(s)	Comments
Potential flexibility of HVAC	kWh	sGRID	

5.1.4 Multi-objective optimization

5.1.4.1 Introduction

The multi-objective optimization tool as part of sENC aims to improve self-consumption and the predictability of energy use within the grid. Based on predictions of water consumption, energy production and congestion during the day from other STREAM tools, this tool optimizes activation schedules for flexible assets such as water pumps and EV charging stations to ensure more efficient energy use and less dependence on the external grid supply.

For water pumps, the tool suggests day-ahead activation schedules that ensure the pumps operate during periods of surplus renewable energy generation. For the EV charge point operator (CPO), the tool adjusts prices based on forecasted energy availability and demand and sends flexible price notifications to consumers via SMS. In addition, end users are encouraged via SMS to adjust their energy consumption during specific time windows to align their consumption with periods of energy surplus and contribute to overall energy efficiency.

Although flexibility is currently limited to price signals and scheduled activations, the ability to integrate battery storage in the future could further enhance the system's ability to store excess energy and provide more dynamic flexibility.

5.1.4.2 Input datasets

The following datasets are used in each tool:

Table 31: Input datasets for multi-objective optimization in IT pilot.

Data element	Units	Tool(s)	Comments
Water consumption forecast	l	sFLEX	Day ahead forecast
Congestion forecast	kWh	sGRID	Day ahead forecast
RES generation forecast	kWh	sENC	Day ahead forecast
Community level flexibility potential	kWh	sFLEX	Temperature dependant flexibility potential of community
Price of energy	€/kWh	sENC	Will be used for optimization in case if there are no congestion issues and all local RES energy is already utilized

Additionally, the following parameters will be gathered for seasonality characterisation:

Table 32: Seasonality data for time series forecasting in IT pilot.

Data element	Units	Comments
Temperature	°C	
Precipitation	mm	
Solar irradiation	W/m^2	

5.1.4.3 Data processing and analysis

The multi-objective optimization tool in sENC processes forecasted data from sFLEX (water consumption), sGRID (grid congestion) and sENC (renewable energy) to calculate optimal energy consumption plans for flexible assets. The optimization is based on maximizing the local use of renewable energy, minimizing grid congestion at the transformer level, or a combination of both. The tool applies mathematical optimization techniques, including linear programming (LP), mixed integer programming (MIP) and evolutionary algorithms, to create day-ahead schedules. These schedules ensure that water pumps receive explicit activation signals, that electric vehicle owners are incentivized through dynamic price adjustments via SMS, and that other consumers receive recommendation messages indicating optimal energy usage times. The system processes time series forecast data, formulates an optimization problem with relevant constraints (e.g. energy availability, congestion thresholds and system flexibility) and calculates an optimal dispatch strategy. Future enhancements could also integrate machine learning models for demand forecasting and battery storage optimization to further improve the flexibility and efficiency of the system. The outputs resulting from the model are listed in Table 33.

Table 33: Output datasets for time series forecasting in IT pilot.

Data element	Units	Tool(s)	Comments
Day-ahead schedule of water pumps switching	ON/OFF signal with timestamp	sENC	Day ahead schedule of when particular pump should be turned on or off
SMS incentive to end-user		sENC	Suggested time intervals to end-users when to reduce consumption

5.2 USER PROFILING

5.2.1 Flexibility assets in the STREAM environment

The Italian pilot site is located in Terni, a city in central Italy. The distribution network of Terni is managed by ASM, the local DSO, which plays a dual role in the STREAM project:

- leveraging flexibility for congestion management within the electrical grid, and
- facilitating cross-sector synergies by exploring the potential of the water distribution system in providing ancillary services to the electric grid.

Within STREAM, ASM is overseeing the implementation of the STREAM ecosystem, integrating advanced energy management solutions to enhance grid stability and reliability.

Additionally, ASM actively engages with end users from local energy communities (LECs) through a dedicated app, empowering community members to participate in grid-friendly operations and respond to demand-response signals.

EMOTION, an EV charging operator, plays a crucial role in the pilot by organizing a “citizens-on-the-move” virtual energy community. Through an innovative flexibility marketplace, EMOTION provides grid services by optimizing EV charging, ensuring consumer preferences are taken into account via a user engagement app. An overview of the energy assets that actively participate in the Italian pilot site is listed in Table 34.

Table 34: Overview of the energy assets in the Italian pilot.

Flexibility asset	Nominal power
Decentralized PV generation	240 kW
Controllable loads through Building Energy Management System (including HVAC)	~100-120 kW
EV chargers	2 × 22 kW, 1 × 50 kW
Water pumps	12 × 30 kW (total 360 kW)

5.2.2 Baseline forecast

In January 2023, monitoring of the water pumps' consumption began. Measurements of active and reactive energy consumption were recorded every 15 minutes over a period of almost two years. The total energy consumption for this period is shown in Figure 66, with the daily consumption values aggregated. It can be seen from this data that energy consumption increases significantly in the months of July to October. Further investigations revealed that this increase is directly related to the drought period, during which more water has to be pumped to meet the higher demand.

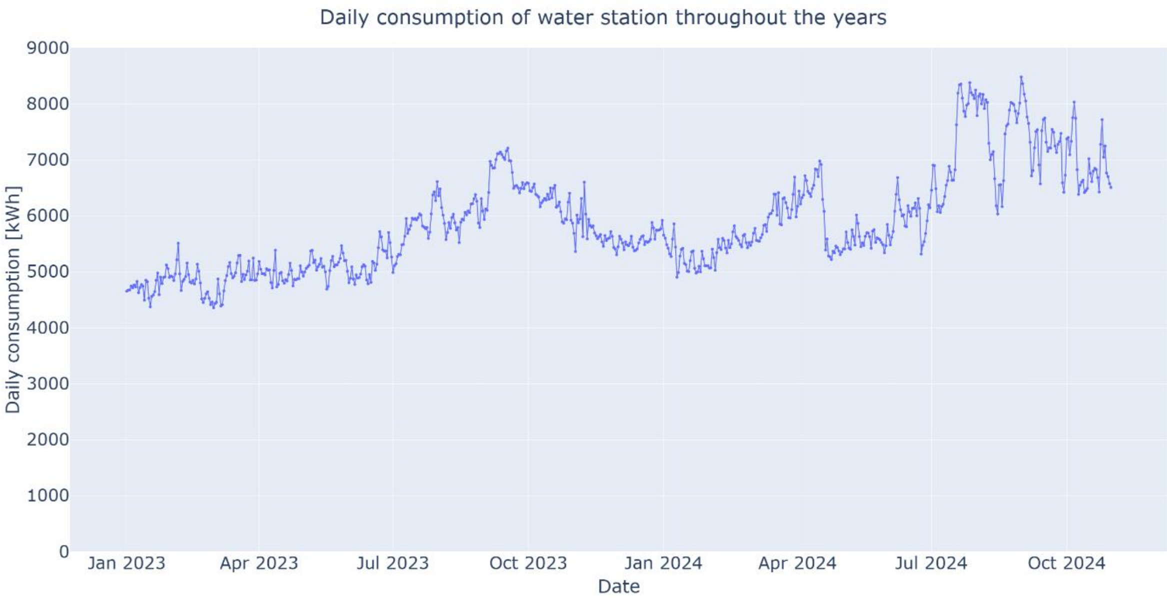


Figure 66: Daily consumption of water pumps since January 2023.

Analysis of the average daily power profile in Figure 67, shows a consistent pattern of energy consumption throughout the day. During the day, from 8:00 to 23:00, the average power consumption is over 250 kW, while at night the consumption drops to as low as 170 kW. This difference between daytime and night-time consumption indicates a potential for operational flexibility. If the capacity of the water reservoir is sufficient to act as a storage buffer, some of the energy-intensive operations could be shifted to the night hours. This strategy would benefit from the typically lower energy prices at night and thus enable cost savings.

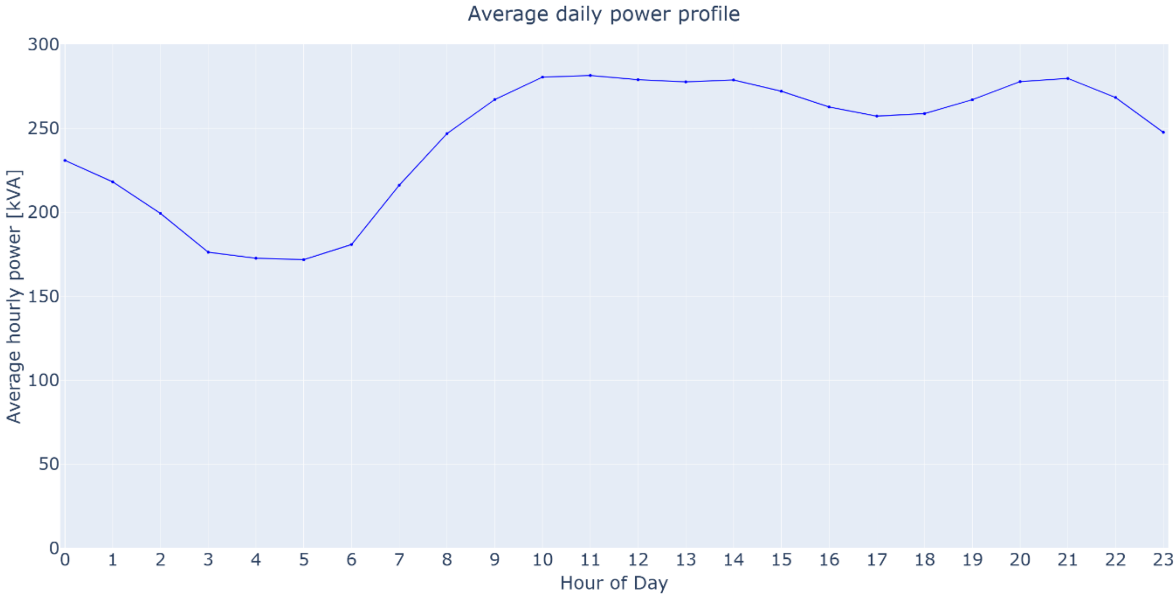


Figure 67: Average daily power profile.

A closer look at the daily electricity profiles by month presented in Figure 68 shows the consistency of patterns within each month, with the highest consumption observed in the drought months of July to October. Outside of these peak periods, the daily profiles remain even and consistent, indicating predictable energy usage patterns.

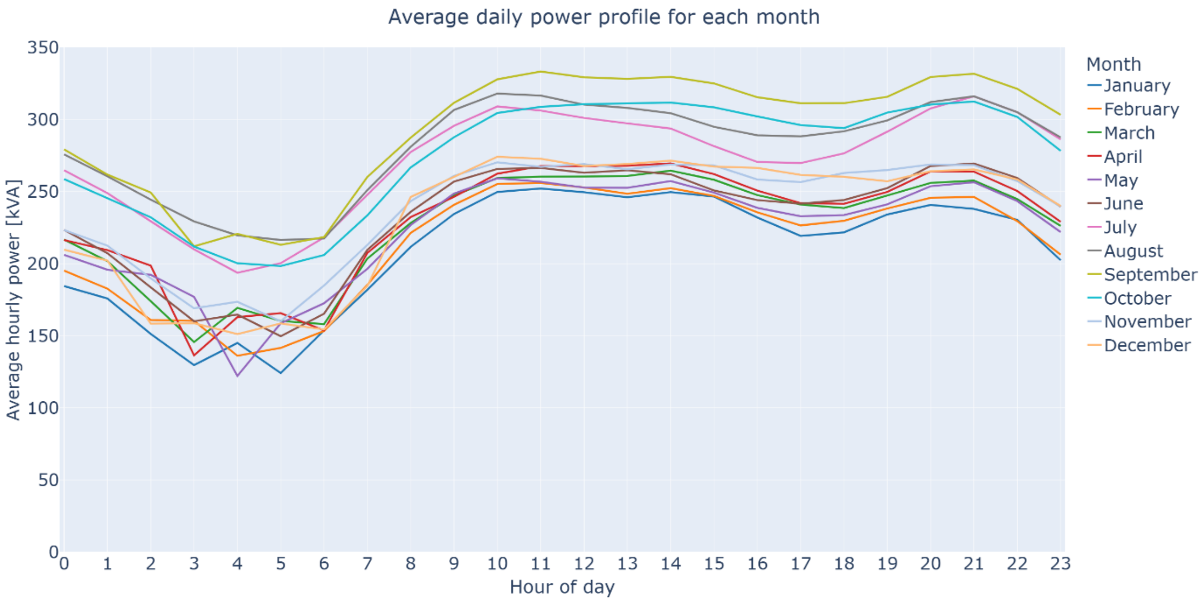


Figure 68: Average daily power profile for each month.

However, while the seasonal patterns are repetitive, a trend of increasing energy consumption over the years is detected. This is evident from the bar plot in Figure 69, where the average monthly energy consumption for each year is represented side by side. The increase reflects the growing demands on the water supply system.

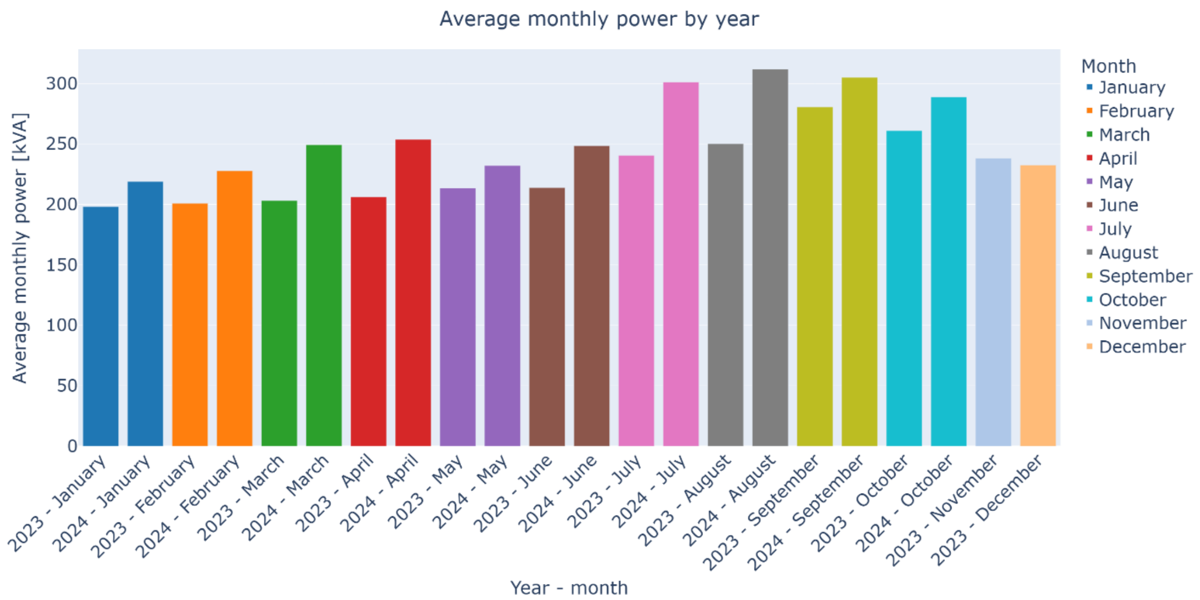


Figure 69: Average monthly power by year.

When comparing power consumption over hours and days, there is no significant difference between weekdays and weekends. This consistency is to be expected as water supply is a critical service for the city that requires continuous operation regardless of the day of the week. The hourly profiles shown in Figure 70 further confirm this, as similar energy consumption patterns were observed across all hours of the day on both weekdays and weekends.

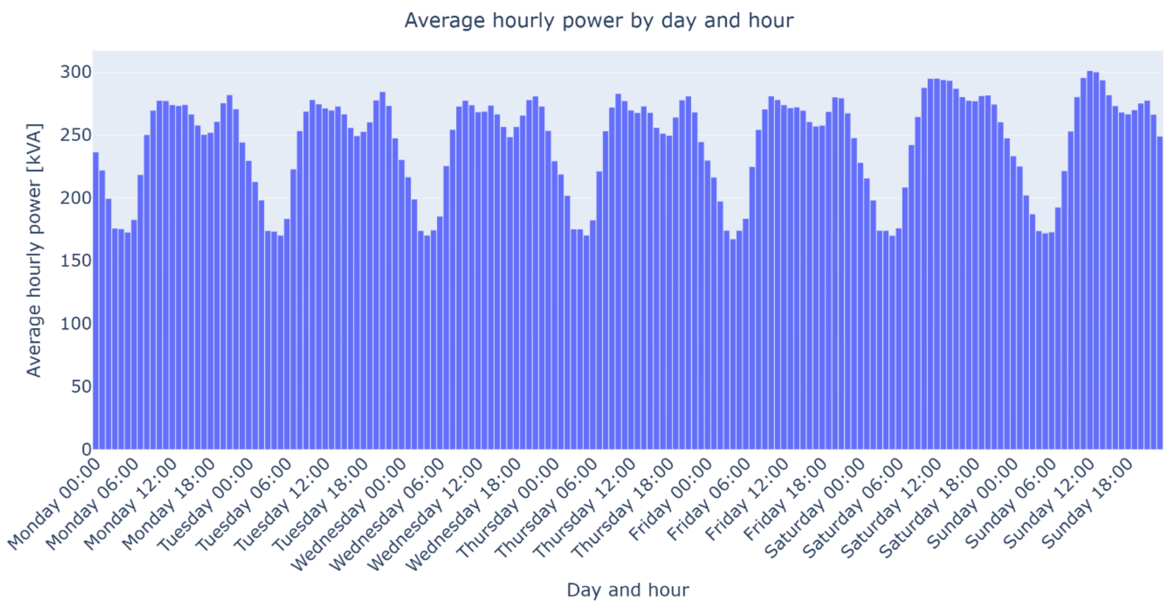


Figure 70: Average hourly profile by day and hour.

The lack of differentiation is also evident when analysing monthly power consumption over the two-year period, shown in Figure 71. Although a reliable seasonal pattern can be observed, with consumption peaking in the drought months of July to October, there is no significant difference between weekends and weekdays on a monthly basis. This supports the conclusion that the water supply system operates evenly to meet the city's demand, regardless of the day.

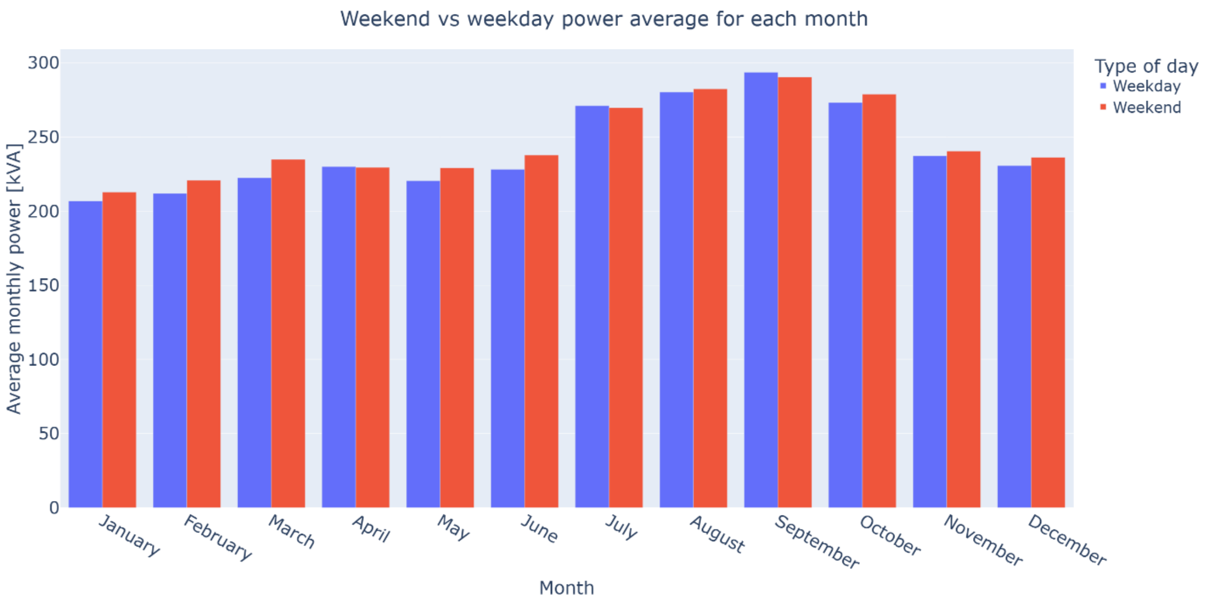


Figure 71: Weekend vs weekday average power for each month.

In terms of data processing, no significant effort was required, as there were no data gaps, so there was no need to fill in missing values with NaN. Furthermore, the collected data was both realistic and consistent, eliminating the need to exclude any outliers with unusually high or low values.

The consumption forecast for the water pumping station was developed using the Prophet time series forecasting model, a robust tool developed by Facebook for processing complex seasonal patterns and trends. The model included several external regressors to capture key factors that influence water consumption. Meteorological variables, including temperature, solar radiation and precipitation data from Open Meteo, were included to account for environmental influences. Additionally, domain-specific variables such as peak and off-peak load hours, vacation schedules and monthly indicators were included to account for operational and seasonal variations in water consumption. The forecast provided a detailed prediction of apparent power consumption (in kVA) and included confidence intervals to highlight the uncertainties in the projections. The results presented in Figure 72 show strong agreement between actual and forecast values, effectively capturing seasonal trends and short-term fluctuations. However, as expected, it struggled to accurately predict very abrupt changes, where the sudden variability was beyond the scope of the model's typical trend and seasonality framework. The model's accuracy was evaluated using standard error metrics, with a **MAPE of 8.20%** and a **wMAPE of 6.83%**. These findings demonstrate the model's reliability in forecasting power consumption while acknowledging its limitations in handling sudden anomalies.

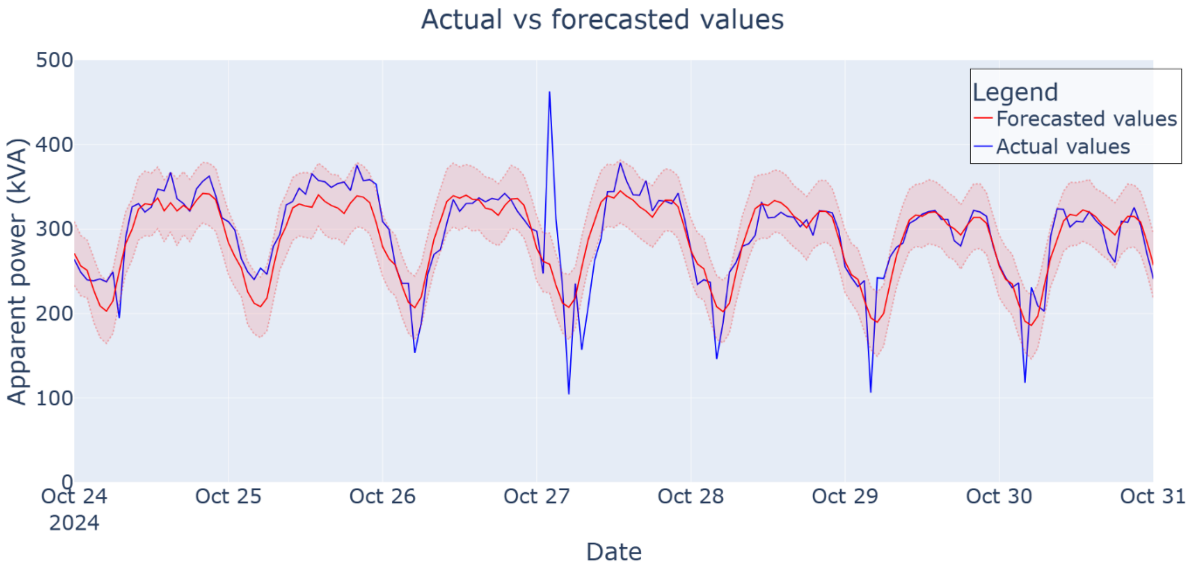


Figure 72: Predicted and actual consumption for 1 week period.

5.2.3 Flexibility forecast

This section outlines the flexibility potential of the water reservoir system, focusing on its ability to support power grid congestion management through adaptive pump operation. By adjusting pump activation thresholds and utilizing day-ahead scheduling, the system can shift electricity consumption away from peak periods of grid demand, contributing to grid stability. The water storage system has the following main features:

- Surface area: 800 m²
- Maximum height of the water: 3.4 m
- Minimum operating water level: 2.0 m
- Usable water level range for flexibility: 1.4 m (from 2.0 m to 3.4 m)
- Pumping system: 10 pumps, each with a nominal power of 30 kW

The current pump activation strategy is based on water level thresholds, Table 35.

Table 35: Pump activation strategy.

Water level [m]	Number of pumps active
3.1 – 3.4	5
2.9 – 3.1	6
2.7 – 2.9	7
2.5 – 2.7	8
2 – 2.5	9
< 2	10

The reservoir can act as a controllable load, allowing the pumping system to be switched off during periods of high grid demand while maintaining water availability. The duration for which the pumps can remain off depends on water consumption rates. Under normal conditions, the reservoir maintains a water level of over 3.0 m most of the time, ensuring a buffer for the provision of flexibility.

Maximum flexibility window (worst-case scenario)

The highest recorded water consumption in 1 year of historical data was 150 l/s (0.15 m^3/s) and lasted 15 minutes before consumption dropped significantly. As already mentioned, the water level is normally kept above 3.0 m therefore the flexibility window for the worst-case scenario can be determined by the time required to drain the reservoir from 3.0 m to 2.0 m (1 m usable height) at this peak consumption:

- Total usable volume of water in the 1-meter height range:

$$V = 800\ m^2 * 1\ m = 800\ m^3$$

- Water depletion rate (worst-case)

$$Q = 0.15\ m^3/s$$

- Flexibility time window:

$$t = \frac{800\ m^3}{0.15\ m^3/s} = 5{,}333.33\ s \cong 1.48\ h$$

In the worst-case scenario, where the consumption rate would remain constant at 150 l/s and no water pump would be returning the water into the reservoir, the system can remain above the water threshold of 2.0 m for 1.48 hours (approximately 1h 29 minutes).

Assuming that only 5 pumps (150 kW in total) are in operation before they are switched off (which again is the worst-case scenario in terms of energy saving), an immediate 150 kW demand reduction can be achieved, helping to mitigate grid congestion at critical times.

This means that the potential energy savings during the 1.48-hour flexibility window are:

$$W = 150\ kW * 1.48\ h = 222\ kWh$$

Note that the peak consumption of 150 l/s occurred only once and lasted 15 minutes, followed by significantly lower consumption rates, so the actual flexibility window under normal operating conditions can extend well beyond 1.48 hours. In addition, with day-ahead scheduling, the reservoir can be filled to a higher level before anticipated congestion, further extending the flexibility period and ensuring a more reliable response to grid demand.

Flexibility based on real consumption patterns

While the worst-case scenario represents a clear upper limit for the flexibility of the system, the actual forecasted flexibility will be based on actual water level, historical consumption patterns and day-ahead forecasts of grid demand and water consumption. By integrating day-ahead scheduling and forecasting tools, the system can optimize its response and adjust water levels in advance to provide more flexibility without depleting the available water buffer.

In reality, consumption peaks of 150 l/s are short-lived and do not last long. As mentioned above, the worst peak in the historical data lasted only 15 minutes before consumption dropped significantly. Therefore, the system can provide longer periods of flexibility than the 1.48 hours calculated in the worst case, especially under normal operating conditions. For example, an average consumption of 60 l/s (0.06 m^3/s) would empty the 1-meter water buffer in about 3.7 hours, while even higher consumption of 100 l/s (0.10 m^3/s) would result in a flexibility window of 2.22 hours. These figures show that under normal consumption conditions, the flexibility period can extend far beyond the worst-case scenario.

5.3 SGRID IN THE ITALIAN PILOT SITE

5.3.1 Overview and architecture

sGRID is an advanced analytical tool that supports DSO in monitoring and managing the distribution grid more efficiently. It fulfills two key functions:

- **Grid impedance matrix calculation**: sGRID calculates the grid impedance matrix based on voltage and power measurements of MV/LV transformers.
- **Prediction of congestion**: By analyzing historical transformer data, sGRID predicts potential congestion issues in the distribution network.

The architecture of the sGRID tool in the Italian pilot site is shown below:

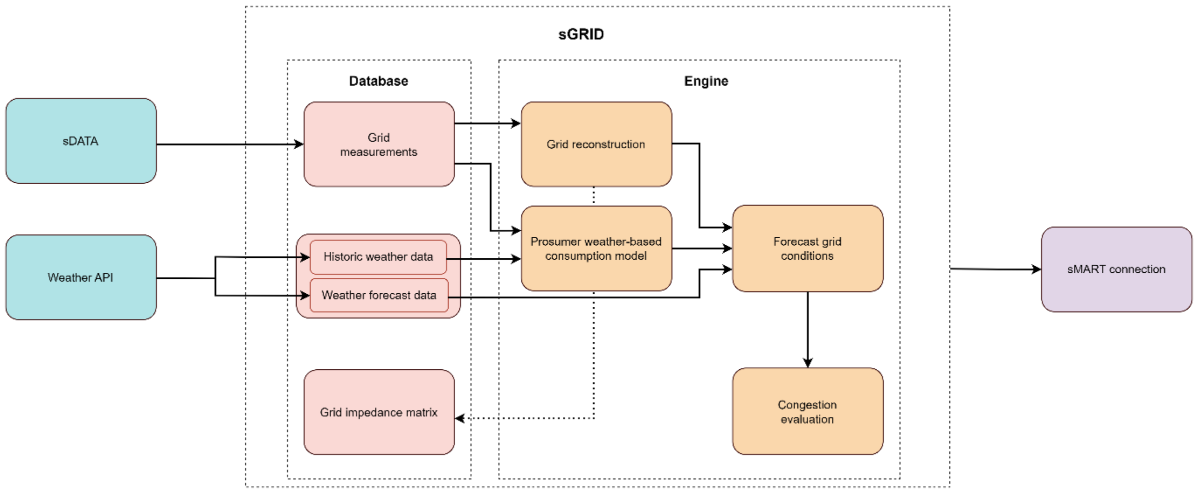


Figure 73: Italian pilot site sGRID tool architecture.

The results generated by sGRID are seamlessly integrated into sMART, a dedicated graphical user interface (GUI) for DSO. Within sMART, congestion points are visualized and operator can take proactive corrective actions, such as submitting flexibility requests, to reduce the load on the grid. Together, sGRID and sMART provide a comprehensive solution to improve grid resilience and optimize grid operations.

5.3.2 sGRID functionalities

sGRID is an advanced analytical tool that supports DSO in monitoring, analyzing and forecasting grid conditions. It plays a crucial role in the calculation of the grid impedance matrix and the prediction of congestion in MV/LV transformers. To achieve this, sGRID processes data from various sources, including electrical measurements from sDATA, weather data from an external API and metadata such as the type of day (weekend, working day or public vacation). The results generated by sGRID are then integrated into sMART, a GUI (shown in Figure 74 below) that enables DSOs to visualize grid conditions and take the necessary actions to maintain stability. Although sGRID itself is not directly integrated into the ICT infrastructure of the DSO, its results serve as essential inputs for sMART, which provides the DSO with the necessary tools for operational decision-making.

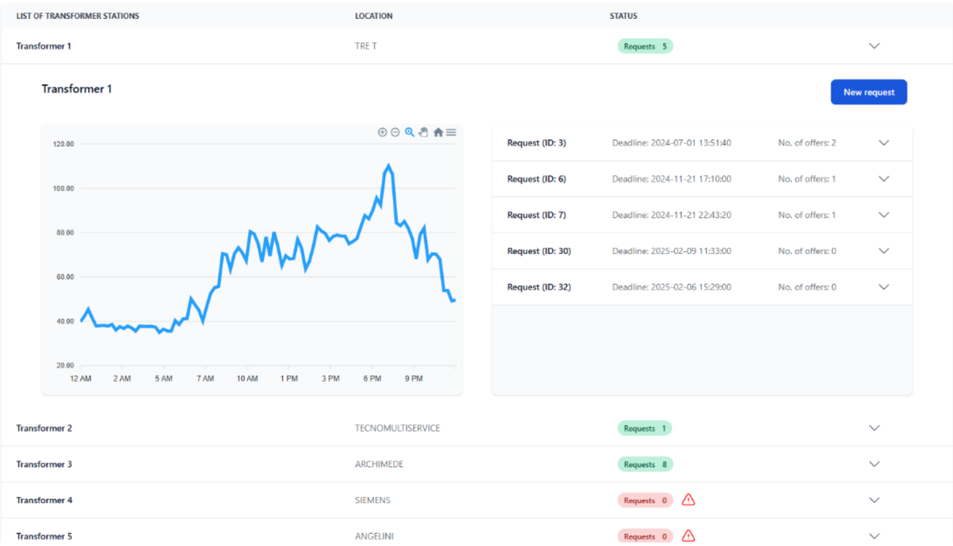


Figure 74: Transformers status overview in sMART.

5.3.2.1 Grid model reconstruction

One of the core functions of sGRID is the calculation of the grid impedance matrix, which provides a detailed representation of the electrical properties of the grid. This matrix is derived from voltage and power measurements collected from MV/LV transformers. Before these measurements are processed, they are validated and pre-processed in sDATA, handling missing values and filtering out unrealistic measurements. This ensures that the impedance matrix is calculated with high-quality, reliable data, which improves the accuracy of grid modelling and analysis.

To verify the quality of the reconstructed grid model, sGRID incorporates network topology data. By comparing the calculated impedance matrix with expected values derived from the known topology, inconsistencies can be identified, helping to detect errors in measurement data or incorrect assumptions about network configuration. This validation step enhances the reliability of the reconstructed grid model, ensuring that it accurately reflects the real electrical characteristics of the distribution network.

Moreover, sGRID supports dynamic model updates by continuously refining the impedance matrix as new measurement data becomes available. This real-time adaptation allows DSO to monitor changes in the electrical properties of the grid due to seasonal demand fluctuations, grid reconfigurations, or the integration of new distributed energy resources (DERs). Maintaining an up-to-date grid model ensures that network planning and operational strategies remain effective, reducing the risk of voltage instabilities and power quality issues.

5.3.2.2 Day-ahead congestion forecast

In addition to impedance calculation, sGRID provides a congestion forecasting feature, allowing DSO to predict transformer overloads before they occur. This prediction process relies on multiple sources, including historical electrical measurements, weather conditions and metadata related to the type of day. Since power consumption patterns vary significantly between weekdays, weekends and holidays, incorporating this information increases the accuracy of congestion predictions. By recognizing these fluctuations, sGRID creates more accurate forecasts that enable DSOs to make informed operational decisions. To further refine the congestion forecasts, sGRID integrates weather data from an external API that provides historical trends and forecasts for temperature, precipitation and solar radiation. These environmental factors have a direct impact on electricity demand and distributed generation — extreme temperatures affect heating and cooling loads, while solar radiation affects photovoltaic

(PV) power generation. By correlating these weather conditions with grid measurements, sGRID improves its ability to predict congestion and detect potential grid constraints before they escalate.

5.3.2.3 sMART connection

Once sGRID has produced its congestion forecasts, the results are transferred to sMART GUI used by the DSO. Within sMART, an explicit function of the GUI is the visualization of congestion, where predicted transformer overloads are displayed on an interactive grid map as shown in Figure 75. This enables DSO to assess congestion risks at a glance and take appropriate measures to maintain grid stability. In addition, the GUI provides access to historical congestion trends, allowing operators to analyze recurring problems and optimize future grid operations.

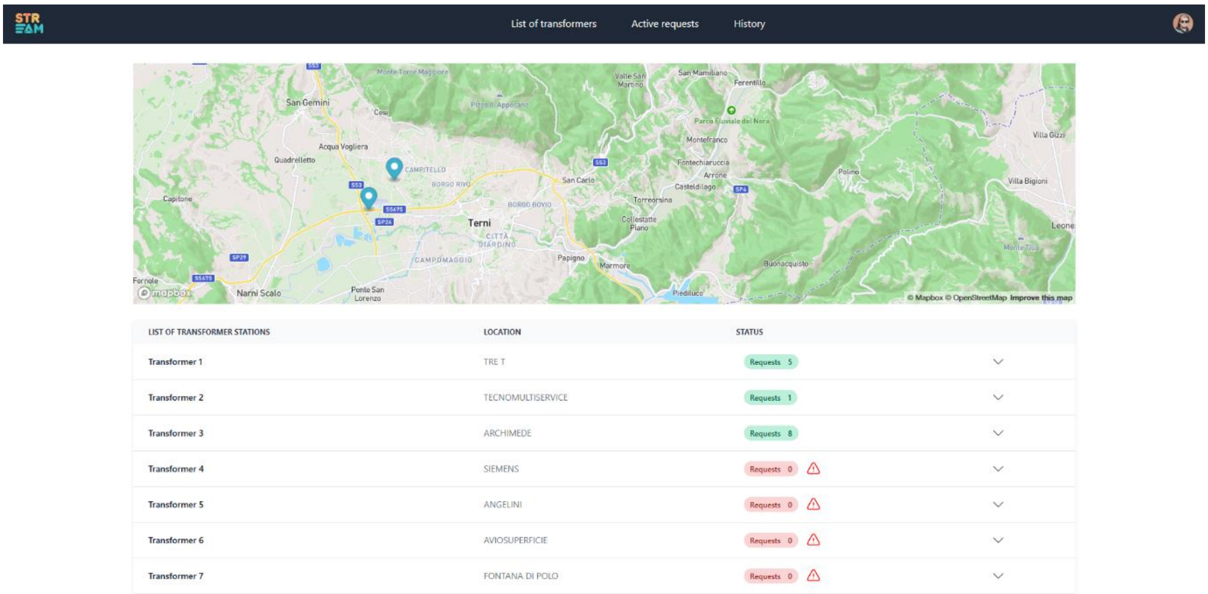


Figure 75: Predicted transformers overloads.

5.3.2.4 Summary

To summarize, the sGRID serves as an advanced analytics engine and plays a supporting role in proactive grid management. It has no direct interface to the ICT systems of the DSO, but instead provides essential congestion forecasts and impedance calculations to sMART, where the DSO can visualize, analyze and act on the information. By integrating with sDATA for validated grid measurements and the weather API for environmental forecasts, sGRID ensures that DSO receives accurate, data-driven insights to improve grid reliability and efficiency.

6 FINNISH PILOT SITE

6.1 DATA ANALYTICS

Due to the complexity of the demand-response functionality, data analytics are needed for improving the understanding of certain key aspects of the pilot projects such as what is the impact on the customers-costs of being part of a Frequency Containment Reserve (FCR) group or what kind of buildings and what kind of heaters are best suitable for being used in the demand-response market.

In addition to user profiling, there are three major data analytics models related to three separate use cases:

- Calculating the difference of heating costs by using explicit flexibility versus not using explicit flexibility (part of use case FI05 “Evaluating the profitability for the customer”).
- Calculating the differences in room temperatures by using explicit flexibility versus not using explicit flexibility (this helps for evaluating the need for additional settings that are meant to prevent room temperature to deviate too much from the set values when using flexibility and therefore it helps in use case FI02 “Maximizing flexibility potential and minimizing customers’ discomfort”).
- Calculating building-types’ and heating-types’ potential (use case FI04 “Defining the buildings potential on flexibility”).

Both user profiling and the other analytics models affect the following tools: sFLEX and sDATA. sFLEX contains the logic, sDATA contains the data.

Table 36 summarizes those analytics models.

Table 36: Data analytics models of the Finnish pilot.

Data analytics model	Involved tool(s)
Costs savings with explicit flexibility	sFLEX, sDATA
Room temperature changes with explicit flexibility	sFLEX, sDATA
Building and heating-types’ flexibility potential	sFLEX, sDATA

User profiling is described in a dedicated chapter.

6.1.1 Calculating the difference of heating costs by using explicit flexibility versus not using explicit flexibility

6.1.1.1 Introduction

One of the main goals of the Finnish pilot project is studying the business feasibility for the customer of being part of FCR services. In order to do this, some calculations are needed to compare the costs of using explicit flexibility versus the costs of not using it. This is part of use case FI05 “Evaluating the profitability for the customer”.

The sFLEX Analyzer is the module which is responsible for those calculations.

Note that it is hard to use any machine learning model in this case due to the unavailability of historical data related to explicit flexibility that would be needed to train the new model.

Later on, after enough data has been collected about the demand-response functionality, this data could be used to fit prediction models.

6.1.1.2 Input datasets

In order to calculate the effects of participating in a FCR, the actualized power consumption from the period under scrutiny is needed along with the electricity contracts participants had at the time. Contracts are required whether or not hour-based pricing is used. Depending on how to calculate the effects, either the consumption simulation results or details about when and which FCR operation was carried out is relied on as precisely as feasible.

Table 37: Input datasets of the different of heating costs analytics.

Data element	Units	Tool(s)	Comments
Used energy when being part of a FCR group	Wh	sFLEX, sDATA	
Used energy without being part of a FCR group	Wh	sFLEX, sDATA	
Baseline forecast	W	sFLEX, sDATA	
Room temperature	°C	sFLEX, sDATA	
Outdoor temperature	°C	sFLEX, sDATA	
Customer contract	c/kWh	sFLEX, sDATA	
Energy spot price	c/kWh	sFLEX, sDATA	
FCR activation and deactivation events	timestamp	sFLEX, sDATA	

Note that they might also be extra HW installation costs (€) to be considered when calculating customer profitability. Also, the FCR market size (€/year) is used to calculate the profitability for the aggregator.

6.1.1.3 Data processing and analysis

There are (at least) two alternatives to calculate the effects of participating in an FCR.

The first alternative is to calculate the difference between actualized and simulated power consumptions. This approach obviously relies heavily on the accuracy of the simulation. Such difference would point out the effects of FCR operations (when power was cut off, when backlashes occurred, etc.). The simulation results are the baseline forecasts referred in chapter 6.2.2. and thus, serve as an input in this analytical model.

The second alternative does not rely on consumption simulations at all. Instead, it uses the detailed knowledge of which FCR operation was carried out at any given time during the period under scrutiny (cutting power or backlash). Therefore, it is possible to mask the effects of such operations from the actualized power consumption. FCR activation and deactivation events (listed as input) are the mask. With this information it is possible to recognize from the actualized power consumption time series the periods when power was consumed or cut due to FCR operations. The resulting time series tells us how much power was consumed/cut due to FCR.

In either case detailed knowledge about electricity contracts and their relevance in the measurement period are required to calculate the (actual or potential) cost in euros from the (actual or potential) energy consumption. This is especially true when hourly-based contracts are used. Hourly-based contracts, referred also as spot-price contracts, charge the customers based on the used energy for each hour. In the future, energy price might be calculated with a granularity of 15 minutes.

Both algorithms will be tested. The results will be reported in the deliverable D5.1 describing the data sets.

Table 38: Output datasets resulting from the analytics model.

Data element	Units	Tool(s)	Comments
Customer's savings	€	sFLEX, sDATA	
Profit for the aggregator	€	sDATA	CONFIDENTIAL
Defined customer compensation	€	sDATA	Part of the aggregator's business model - CONFIDENTIAL

6.1.2 Calculating the difference of temperature by using explicit flexibility versus not using explicit flexibility

6.1.2.1 Introduction

One of the potential negative effects of adding a certain room in a flexibility group is that the room temperature might differ from the set temperature due to either extra heating or temporary switching-off of the heating device. While based on the statistics collected by the TSO so far it is foreseen that such temperature deviations will not be big, it might be useful to have additional customers settings to prevent the room temperature deviating too much from the set temperature.

This is part of the use case FI02 “Maximizing flexibility potential and minimizing customers' discomfort”.

Simulating the difference in room temperature when using flexibility could help in evaluating the need for such functionality and in fine tuning its specification.

The sFLEX Analyzer is the module which is responsible for those calculations.

The working assumption is that the energy consumption (or lack of it), and thus the accrued costs/savings, are far more noticeable and relevant for customers than room temperature. The rationale behind this is that disturbances leading to power cuts (and potentially subsequent backlashes) are most often very short (lasting just few seconds) and in such time there's no noticeable difference in room temperature whatsoever. This assumption is based on the statistics the TSO has provided.

For longer disturbances, the effects on the temperature can be estimated if the room's behaviour is known in advance. Delay is a measure of the time it takes for the room temperature to change of a predefined step such as 0.2 centigrade. Room's delay can be (and currently is) calculated from room temperature measurements. There are two separate values for delay: cool-down (a drop in room temperature) and heat-up (a hike in room temperature).

6.1.2.2 Input datasets

The following variables can be used as input data set:

Table 39: Input datasets required for the difference of temperatures by using and not using explicit flexibility.

Data element	Units	Tool(s)	Comments
Actualized electric power usage for each room in an aggregate (Flex Group)	W	sFLEX, sDATA	
Electricity prices for each apartment	c/kWh	sFLEX, sDATA	Contracts do vary from one apartment to another
Flexibility operations (power cuts, backlash) which have been done	time stamp	sFLEX, sDATA	
Temperature measurements for each room in an aggregate	°C	sFLEX, sDATA	
Delay values for each room in an aggregate	seconds	sFLEX, sDATA	

6.1.2.3 Data processing and analysis

Data processing depends on the level of granularity that is used in the activation of flexibility services. The final result of this analytical model will be reported in the deliverable D5.1 describing the data sets.

Assuming room level granularity in flexibility activations

The first step is to create a time series of the flexibility operations that have been done. The series should indicate periods when flexibility operation was active as opposed to idle periods when rooms were under regular control (including implicit flexibility). The resulting series could simply be a discrete binary signal which can then be used to mask the effects of activations from electric power usage which is measured/calculated independently from flexibility activations for each room. As mentioned, activation periods are expected to be generally very short and therefore the effects of electricity (energy) consumption are expected to be moderately low.

Once the effects in power usage have been established, it is straightforward to calculate the individual costs (due to backlash) and savings (due to power cuts) for each activation separately and in total since used contracts and electricity prices are readily available.

Assuming Flex Client (typically one apartment) level granularity

In the case of apartment-level activation there are not many changes in constructing the masking signal compared to the room-level activation. Power usage, however, is now the total of all loads under the Client rather than the load of each individual room and this affects costs/savings analysis to an extent. If room level accuracy cannot be established, then the Client level consumptions and costs/savings is the only option. Of course, it is still possible to compute a crude average of activation effects per room under the Client.

Analysing the effects on room temperature

As mentioned in the introduction, delay values can be used to estimate the effects of frequency disturbances on rooms' temperature. If disturbances are short, the effects on the temperature are insignificant. For longer disturbance lasting several minutes (ought to be a rather rare occurrence), delay can be used to either interpolate (within delay) or extrapolate (longer than delay) an estimate on the change in temperature for each longer activation. Of course, there's no need to discriminate between short and long activations but if the disturbance lasts, let's say one second, it is pointless to derive any effect on temperature for such a short period. For really long disturbances that outlast delay, it can either be assumed that the rate of change is linear (which it isn't, but likely it is still a sufficient approximation) and an extrapolation can be done or the actual change in temperature can be calculated by using Newton's formula for cooling (or heating). This is possible since the required parameters are computed by OptiWatti's system anyway. This analysis presumes room-level accuracy in flexibility activations.

6.1.3 Calculating building-types' and heating-types' potential

6.1.3.1 Introduction

It is interesting to study if there are any differences in terms of benefits of applying explicit flexibility based on different types of building and different heating types. This would also help once the feature is commercialized since it would allow to target the most suitable customer segments.

This is part of the use case FI04 "Defining the buildings potential on flexibility".

In the pilot project it is hard to use any machine learning model for this analytical model due to the unavailability of historical data related to explicit flexibility that would be needed to train the new model. However, during the pilot project, data can be collected and stored in order to be able in the future to fit prediction models to fine tune the calculations.

Part of the data extraction and collection process shall be run manually as described in the table below.

Part of the calculations are run automatically, and part of the calculations are done by running manually certain scripts as defined in more details in the following chapters.

The sFLEX Analyzer is the module which is responsible for those calculations.

6.1.3.2 Input datasets

Table 40 shows a list of input variables for the calculation and their location. The data is contained in sDATA and the calculations are done in sFLEX.

Table 40: Input data set for calculating building-types' potential.

Variable	Location	Notes
Available fast power of the apartment	Optiwatti measurements database	"Fast power" means power that can be cut or added quickly and therefore power that can be used for FCR
Available slow power of the apartment	Optiwatti measurements database	"Slow power" means power that cannot be cut or added quickly and therefore that cannot be used for FCR
Apartment's customer type	Optiwatti control database	Possible values: B2C/B2B
Heater type	Optiwatti control database	Possible values: electric radiator / water-based radiator / deep electric floor heating / surface electric floor heating / …

Variable	Location	Notes
Apartment type	WebCRM	Possible values: detached house, semi-detached house, cottage, cottage village (holiday resort), camping site, parish hall, … The information might have to be collected manually
Apartment's age	WebCRM	The information might have to be collected manually
Building material	WebCRM	Possible values: concrete, wood, … The information might have to be collected manually

Fast power is the power consumed by devices, which "immediately" switch ON or OFF when the controlling relay toggles to ON or OFF state. For such devices the output power must change in less than 1 second after the relay toggles to a different state. The requirement comes from TSO FCR specifications.

6.1.3.3 Data processing and analysis

Table 41 shows the output dataset. The data is contained in sDATA and the calculations are done in sFLEX.

Table 41: Output dataset for calculating building-types' potential.

Variable	Notes
Apartment id	An integer identifying the apartment
Market	Possible values: FCR, other, …
Apartment's customer type	Possible values: B2C/B2B
Apartment type	Possible values: detached house, semi-detached house, cottage, cottage village (holiday resort), camping site, parish hall,
Apartment's age	
Building material	Possible values: concrete, wood, …
Fast_capacity_max	"Fast power" means power that can be used for FCR
Fast_avg_capacity_<month-name>	The average power that can be used as part of a FCR group in a certain month
Slow_capacity_max	"Slow power" means power that cannot be used for FCR
Slow_avg_capacity_<month-name>	

The result of this analytical model will be reported in the deliverable D5.1 describing the data sets.

6.2 USER PROFILING

6.2.1 Flexibility assets in the STREAM environment

The flexibility assets in Finnish pilot site consist of heating systems within 6 residential apartments and 3 cottages in a holiday resort. Those buildings are equipped with OPT's management system, which optimizes the use of assets while maintaining proper indoor conditions.

A wide range of loads can be controlled through OPT energy management system, including:

- Radiators
- Floor heating
- Heat pumps
- Hot water boilers
- EV charging units

Only electric radiators and electric floor heating are used as flexible loads. Heat pumps are not used as flexible load in this pilot since their switch on/off cannot happen quickly enough for demand-response and since OPT's system is not aware of the exact power such devices are using in a defined interval of time. Hot water boilers are not included in the flexible load either since OPT does not have yet full control of those loads because they can be turned off by their local thermostat. EV charging units are the most challenging type of load to be used for flexibility since the aggregator has no information on the presence of the EV and its connection to the grid. Therefore, the overall controllable load that can be used as a flexible load from the apartments is 54 kW. Table 42 contains more details for each building.

Table 42: Flexibility assets, (') Calendar integration feature enabled.

Location	Building type and size	Controlled flexible power (kW)	Floor heating	Heat pump	Hot water boiler	EV	PV power (kWp)
Järvenpää	Residential-detached house, 142 m^2	7	Yes	Yes	-	2	6.84
Veiokkkola	Residential row house, 104 m^2	7.2	Yes	Yes	Yes	0	-
Hämenlinna	Residential-detached wooden house, 180 m^2	12.2	Yes	Yes	-	1	5.7
Lappeenranta	Residential-detached house	3.2	Yes	No	Yes	0	-
Espoo Viherlaakso	Residential row house, 135 m^2	3.6	Yes	Yes	-	1	4.44
Espoo Vaasanrinne	Residential-detached house, 140 m^2	7.2	Yes	Yes	-	0	3.3

Location	Building type and size	Controlled flexible power (kW)	Floor heating	Heat pump	Hot water boiler	EV	PV power (kWp)
Ivalo (‘)	3 cottages with glazed roof igloo, 28 m^2	13	Yes	Yes	-	0	-

In addition to these nine buildings, data from over 2,000 other buildings is available for specific calculations such as estimating how much flexible capacity overall the aggregator (OPT) can offer to the TSO. However, the data aggregation interval increases with age according to the defined purging policy: for data older than one year, only hourly aggregates are retained and, for data between three and five years old, only daily aggregates are retained.

Before sending any bid to the TSO, sFLEX needs to estimate the available power that can be used for the demand-response functionality (explicit flexibility). This shall be done by considering also consumer’s reaction to energy’s price changes (implicit flexibility). Implicit flexibility allows consumers to set their control system to use less energy when energy prices are above a pre-defined limit. This may affect the available power to be used for the demand-response functionality. This is part of the basic use case FI01 “Using heating for flexibility”. This use case describes how to use heating load as a source of flexibility to offer FCR-D and FCR-N services to the TSO. This functionality is achieved by the sFLEX Simulator component of the sFLEX tool.

6.2.2 Baseline forecast

The sFLEX Simulator performs a baseline forecast by leveraging historical data and a suitable predictive model. The sFLEX Simulator uses measurements and control data (user’s settings, weather forecasts, spot prices, room temperature, outdoor temperature, etc.) to simulate the heating of rooms.

For the first pilot phase, a simple and tested method is used (explained in the following subsection). This method is currently used by the basic functionality of OPT’s system. It uses the data set FI01_05 “Baseline flex power forecast input”, which provides the input for the flexible power baseline forecast. In a second phase, a more sophisticated method which uses machine learning might be investigated.

6.2.2.1 Calculation used in the first prototype

An iteration is performed in a 5-minute interval. The iteration uses the OPT software which OPT controllers use to predict when to start or stop heating (the details are IPR-protected). This algorithm uses the following variables as input:

- outdoor temperature
- weather forecast
- electric energy spot prices
- users’ preferences in OPT User Interface (UI) settings
- room temperature
- calculated used energy in the room

Those are the start conditions. The system calculates how much the room temperature changes during the 5 minutes and the used energy in this time interval. The simulated temperature of the room is then used as an input variable for the next iteration. The used power during the 5-minute period is stored. Then, the next iteration starts.

Figure 76 shows the average weekly power used for heating by OPT customers in the year 2023.

Figure 76: Average weekly power used for heating by OPT customers in year 2023.

Figure 77 shows the average weekly energy price per MWh in Finland during year 2023.

Figure 77: Average weekly energy price per MWh in Finland during year 2023.

Figure 78 shows the average weekly outdoor temperature in all the apartments controlled by OPT.

Figure 78: Average outdoor temperature in all the apartments controlled by OPT during year 2023.

Figure 79 shows the average weekly temperature of a room which belongs to an apartment that is part of the Finnish pilot project. The same plot shows also the average weekly power used for heating that room for the full year.

Figure 79: Average weekly temperature of a room and average weekly power used for heating that room.

6.2.3 Flexibility forecast

Currently, there is no historical data of the used flexible power so currently no machine-learning-based prediction model can be created. In the first iterations, a safety margin can be applied to the baseline, like for example 70% of the baseline value.

This safety margin is needed for complying with customer settings which might partially limit the usage of the apartments' assets as a source of flexibility. Such settings have the scope of mitigating customers' discomfort caused by the demand-response functionality. Also, in order to minimize the impact on customers' perception, load is rotated within a single FCR activation. In other words, the very same amount of flexible load might be provided by different apartments during a single FCR activation-deactivation cycle. Finally, there is the possibility of some apartments' controllers not responding to the FCR activation request, even if this is a rare event.

The safety margin is not a hard-coded value. Once the reliability of the estimation is tested, a smaller safety margin (that is, a higher value of the percentage), can be used.

Data on explicit flexibility usage will be gathered once the system is operational. This data will enable future refinement of the model through machine learning.

The description below outlines the approach that will be followed in the pilot:

- The **power baseline** is calculated as the sum of the estimated power consumption of all the rooms in a flexibility group at a given hour.
- The delta to zero with safety margin is given by 0.7 times the baseline. This is the flexible load that can be cut and that can be used in the bid to the TSO for the FCR-D-up service. In other words, this is the **FCR-D-up-capacity.**
- The maximum power (or **max_power**) is calculated as the sum of the power which is installed in the heaters in all the rooms in a flexibility group.
- The delta to max power with safety margin is given by: (max_power - baseline) * 0.7. This is the flexible load that can be added and that can be used in the bid to the TSO for the FCR-D-down service. This is the **FCR-D-down-capacity**.

Figure 80 shows the different variables. The power values are negative since they are absorbed power.

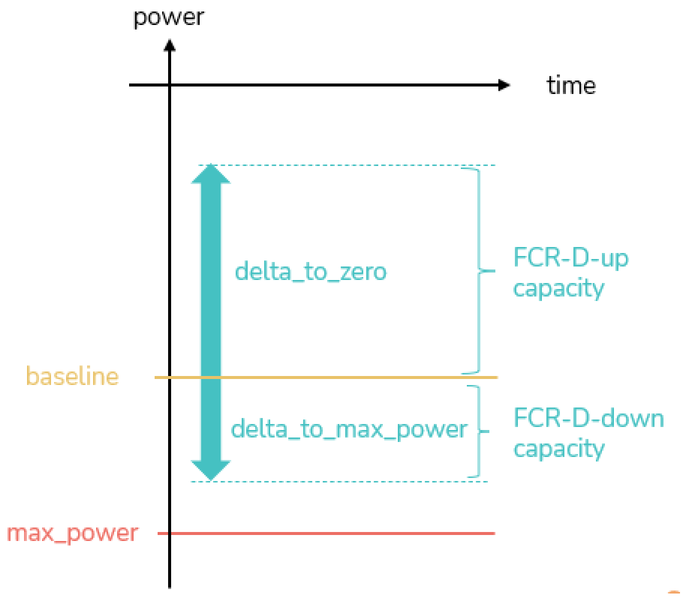


Figure 80: Flexibility forecast.

7 CONCLUSIONS

This deliverable has detailed the activities carried out in **T3.4, ST3.4.1, and ST3.4.2** across the **Slovenian, Spanish, Italian and Finnish pilot sites**. A key achievement under **T3.4** has been the development of the **sGRID and sPLAN tools**, tailored for **DSOs**. **sGRID**, designed for real-time **grid monitoring and congestion forecasting,** directly **enabling DSOs to send flexibility requirements to the flexibility market**, is being deployed in **Slovenia, Spain, and Italy**, while **sPLAN**, aimed at **grid planning**, is currently implemented in **Slovenia**. Each of these pilot sites has adapted these tools to their specific needs, and this deliverable documents their architectures and functionalities.

Under **ST3.4.1**, a detailed analysis of the **data analytics models** used in of the four pilot sites tool has been conducted. This includes an evaluation of the **models applied in each pilot site**, their **input data requirements**, **processing workflows**, and **output data**. In total, 15 different models have been integrated into the four pilot sites.

Furthermore, in **ST3.4.2**, the four pilot sites have developed **user profiling methodologies** to enhance forecasting accuracy. Given the complexity of these methodologies, each pilot site has exemplified the approach using **a representative asset**, ensuring clarity while avoiding excessive document length.

The **deployment and demonstration of sGRID and sPLAN will continue under WP5**, with further validation and refinements expected. Progress will be documented in **Deliverable 5.1: Demonstration Activities Report**, providing insights into the tools' **performance, scalability, and impact on grid operations**.

8 REFERENCES AND ACRONYMS

8.1 REFERENCES

[1] IRENA, "Future role of Distribution System Operators - innovation landscape brief," 2019. [Online]. Available: https://www.irena.org/-/media/Files/IRENA/Agency/Publication/2019/Feb/IRENA_Landscape_Future_DSOs_2019.PDF?la=en&hash=EDEBEDD537DE4ED1D716F4342F2D55D890EA5B9A.

[2] STREAM Project, *Deliverable 2.3 - Business models structuring*, 2024.

[3] STREAM Project, *Deliverable 4.2 - sMART local market structure development,* 2025.

[4] [Online]. Available: https://xgboost.readthedocs.io/en/stable/python/python_intro.html.

[5] [Online]. Available: https://mlflow.org.

[6] pandapower, "pandapower documentation," [Online]. Available: https://pandapower.readthedocs.io/en/v2.2.1/index.html.

[7] "scikit learn," [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html.

[8] "github," [Online]. Available: https://github.com/Microsoft/LightGBM.

[9] "scikit learn," [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.HistGradientBoostingRegressor.html.

[10] [Online]. Available: https://www.meteo.si/met/sl/app/webmet/#webmet=vUHcs9WYkN3LtVGdl92LhBHcvcXZi1WZ09Cc1p2cvAncvd2LyVWYs12L3VWY0hWZy9SaulGdugXbsx3cs9mdl5WahxHf;.

[11] STREAM Project, *Deliverable 3.1 - sDATA open platform developed*, 2024.

[12] "XGBoost," [Online]. Available: https://xgboost.ai/.

[13] "eduardostalinho/jsonbender: Library for transforming JSONs," [Online]. Available: https://github.com/eduardostalinho/jsonbender.

[14] "dr-prodigy/python-holidays: Generate and work with holidays in Python," [Online]. Available: https://github.com/dr-prodigy/python-holidays.

[15] "Python API Reference — xgboost 2.1.3 documentation," [Online]. Available: https://xgboost.readthedocs.io/en/stable/python/python_api.html#module-xgboost.sklearn.

[16] “Datastructure and Elements — pandapower 2.14.10 documentation,” [Online]. Available: https://pandapower.readthedocs.io/en/stable/elements.html.

[17] “Cytoscape: An Open Source Platform for Complex Network Analysis and Visualization,” [Online]. Available: https://cytoscape.org/.

[18] “Balanced AC Power Flow — pandapower 2.14.10 documentation,” [Online]. Available: https://pandapower.readthedocs.io/en/stable/powerflow/ac.html#.

[19] “Diagnostic Function — pandapower 2.14.10 documentation,” [Online]. Available: https://pandapower.readthedocs.io/en/stable/powerflow/diagnostic.html#.

[20] “Intro to data structures — pandas 2.2.3 documentation,” [Online]. Available: https://pandas.pydata.org/docs/user_guide/dsintro.html#dataframe.

[21] “Yellowbrick: Machine Learning Visualization — Yellowbrick v1.5 documentation,” [Online]. Available: https://www.scikit-yb.org/en/latest/index.html.

[22] “KMeans — scikit-learn 1.6.1 documentation,” [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html.

[23] “Norvento Enerxia,” february 2024. [Online]. Available: https://www.norvento.com/en/.

[24] “CEGASA,” [Online]. Available: https://www.cegasa.com/en/lithium. [Accessed february 2024].

[25] “Huawei,” february 2024. [Online]. Available: https://solar.huawei.com/es.

[26] “Free Open-Source Weather API | Open-Meteo.com,” [Online]. Available: https://open-meteo.com/.

[27] “PuLP · PyPI,” [Online]. Available: https://pypi.org/project/PuLP/.

[28] “coin-or/Cbc: COIN-OR Branch-and-Cut solver,” [Online]. Available: https://github.com/coin-or/Cbc.

[29] “X-FLEX project,” [Online]. Available: https://xflexproject.eu/.

[30] Meteor, “Meteor.js an open source platform for building Web, Mobile, and Desktop applications.,” [Online]. Available: https://www.meteor.com.

[31] “Overview | BlazeJS,” [Online]. Available: https://www.blazejs.org/.

[32] REACT, “REACT,” february 2024. [Online]. Available: https://react.dev/.

[33] “npm | Home,” [Online]. Available: https://www.npmjs.com/.

[34] MongoDB, [Online]. Available: https://www.mongodb.com.

[35] InfluxData, “InfluxDB,” [Online]. Available: https://www.influxdata.com.

[36] RabbitMQ, “RabbitMQ,” [Online]. Available: https://www.rabbitmq.com.

[37] "NATS.io – Cloud Native, Open Source, High-performance Messaging," [Online]. Available: https://nats.io/.

[38] "Keycloak," [Online]. Available: https://www.keycloak.org/.

[39] "OpenID - OpenID Foundation," [Online]. Available: https://openid.net/.

[40] "OAuth 2.0 — OAuth," [Online]. Available: https://oauth.net/2/.

8.2 ACRONYMS

Acronyms list	
AMI	Advanced Metering Infrastructure
API	Application Programming Interface
BESS	Battery Energy Storage Systems
CPO	Charging Point Operator
DER	Distributed Energy Resource
DSO	Distribution System Operator
EC	European Commission
EU	European Union
EV	Electric Vehicle
FCR	Frequency Contingency Reserve
GMM	Gaussian Mixture Model
GUI	Graphical User Interface
HBGBRT	Histogram-Based Gradient Boosting Regression Tree
HP	Heat Pump
HV	High Voltage
HVAC	Heating, Ventilation and Air Conditioning
HW	Hardware
ICT	Information and Communication Technology

IRENA	International Renewable Energy Agency
LFM	Local Flexibility Market
LGBM	Light Gradient Boosting Machines
LP	Linear Programming
LV	Low Voltage
MAPE	Mean Absolute Percentage Error
mFRR	Manual Frequency Restoration Reserve
MIP	Mixed Integer Programming
MV	Medium Voltage
NaN	Not a Number
PV	PhotoVoltaic
RES	Renewable Energy Sources
RFG	Random Forest Regressor
SCADA	Supervisory Control and Data Acquisition
TLS	Traffic Light System
TSO	Transmission System Operator
wMAPE	Weighted Mean Absolute Percentage Error