Digicities

DATA PROCESSING FOR DEMAND PROJECTION

Francesca Mangili, Matteo Salani, Lorenzo Zambon, Marco Derboni, Vincenzo Giuffrida, Andrea Rizzoli, Ali Hainoun

Project End

ERA-Net Smart Energy Systems

This project has received funding in the framework of the joint programming initiative ERA-Net Smart Energy Systems, with support from the European Union's Horizon 2020 research and innovation programme.

Smart Energy Systems ERA-Net

INTERNAL REFERENCE

Deliverable No.:	D 3.1
Deliverable Name:	Peer reviewed WP report on data processing for demand projection.
Lead Participant:	Francesca Mangili
Work Package No.:	3
Task No. & Name:	T3.1, T3.2, T3.2, T3.3, T3.4, T3.5
Document (File):	D3 Data processing for demand projection_v0.2.docx
Issue (Save) Date:	2026-04-10

DOCUMENT STATUS

Version	Date	Author(s),	Description
0.1	2026-03-29	**Francesca Mangili, Matteo Salani, Lorenzo Zambon, Marco Derboni, Vincenzo Giuffrida, Andrea Rizzoli**	
0.2	2026-04.10	Ali Hainoun	Adding the new chapter 3 on AUC2: Prediction of room comfort (temperature and humidity)

DOCUMENT SENSITIVITY

☒	**Not Sensitive**	Contains only factual or background information; contains no new or additional analysis, recommendations or policy-relevant
☐	**Moderately Sensitive**	Contains some analysis or interpretation of results; contains no recommendations or policy-relevant statements
☐	**Sensitive**	Contains analysis or interpretation of results with policy-relevance and/or recommendations or policy-relevant statements
☐	**Highly Sensitive Confidential**	Contains significant analysis or interpretation of results with major policy-relevance or implications, contains extensive recommendations or policy-relevant statements, and/or contain policy-prescriptive statements. This sensitivity requires SB decisions.

☒ [copy and delete]

TABLE OF CONTENT

Disclaimer

The content and views expressed in this material are those of the authors and do not necessarily reflect the views or opinion of the ERA-Net SES initiative. Any reference given does not necessarily imply the endorsement by ERA-Net SES.

About ERA-Net Smart Energy Systems

ERA-Net Smart Energy Systems (ERA-Net SES) is a transnational joint programming platform of 30 national and regional funding partners for initiating co-creation and promoting energy system innovation. The network of owners and managers of national and regional public funding programs along the innovation chain provides a sustainable and service oriented joint programming platform to finance projects in thematic areas like Smart Power Grids, Regional and Local Energy Systems, Heating and Cooling Networks, Digital Energy and Smart Services, etc.

Co-creating with partners that help to understand the needs of relevant stakeholders, we team up with intermediaries to provide an innovation eco-system supporting consortia for research, innovation, technical development, piloting and

demonstration activities. These co-operations pave the way towards implementation in real-life environments and market introduction.

Beyond that, ERA-Net SES provides a Knowledge Community, involving key demo projects and experts from all over Europe, to facilitate learning between projects and programs from the local level up to the European level.

www.eranet-smartenergysystems.eu

1 EXECUTIVE SUMMARY

This deliverable provides a summary of the data processing, ML-based forecasting methods, and modelling activities carried out in the context of the Swiss use case on Model Predictive Control (MPC) for Renewable Energy Resource Optimization.

Time series about energy demand and resources availability were collected, pre-processed and modelled to produce forecasts that could be used within an MPC framework to optimize resource exploitation for improved energy efficiency.

In this report we briefly describe the data pre-processing and modelling steps and summarize the results obtained in two scenarios:

- Peak-shaving in small hydropower plants (AEM, SES), where MPC driven by ML forecasts can improve scheduling performance and reduced peak demand compared to operator decisions.
- Battery sizing and operation for an energy community (LIC), where combined ML-forecasting and MPC simulations provided insights into optimal PV-battery configurations and their impact on self-consumption and self-sufficiency.

Overall, the results show that accurate short-term forecasting offers a robust basis for MPC-based control and decision-making.

Full details on the data processing and modelling approach are presented in the recently published open-access paper:

Mangili, F., Derboni, M., Zambon, L., Giuffrida, V., & Salani, M. (2026). Enhancing Peak Shaving Efficiency in Small Hydro Power Plants Through Machine Learning-Based Predictive Control. Energies, 19(4), 985. https://doi.org/10.3390/en19040985

2 MODEL PREDICTIVE CONTROL FOR RENEWABLE ENERGY RESOURCE OPTIMIZATION.

The use case which has driven the development of data processing methods for demand projection, investigates how machine learning (ML) and Model Predictive Control (MPC) can support the operational management of a small hydropower plant (HPP) or of a small self-consumption community with the goal of improving peak shaving performance and optimizing water or battery storage usage. The test benches are the HPP operated by Azienda Elettrica di Massagno (AEM), which relies on a small reservoir fed by two rivers; the HPP operated by Società Elettrica Sopracenerina SA (SES) in Ticino, which relies on the the Vasasca dam reservoir; the Lugano Innovation Community (LIC), which is connected to PV production and battery storage.

Currently, production scheduling for the HPPs is based on a combination of real-time corrective control (that adjusts system operations in response to observed variations in demand) and operator expertise. Operators aim to minimize daily and monthly electricity peaks—which strongly influence the power transfer price charged by AET—while ensuring sufficient water availability for future needs. When the reservoir is full, production is maximized during peak demand to reduce grid costs; when it is not, operators balance current peak reduction with future resource availability. They also adjust decisions manually based on weather conditions (e.g., increasing production before forecasted temperature drops). Although experienced operators handle this well, the growing complexity of grid conditions makes decision support increasingly valuable.

For batteries, the typical management is based on charging them when PV production exceeds community demand and discharging when demand exceeds PV production.

To enhance decision making, the project introduces an MPC based scheduling system that optimizes HPP production over a 48 hour horizon with variable resolution. MPC uses updated forecasts at each time step and adjusts only the first control action, recalculating the rest when new information becomes available. MPC thus determines the discharge power sequence that minimizes the maximum power drawn from the grid across the horizon for HPPs and maximizes self-consumption for battery management, using predicted demand values.

Accurate short term energy demand forecasting is essential for MPC performance. Earlier work by Derboni et al. (2021) used a baseline predictor combining seasonal averages with weather-based classification to estimate future demand. However,

This project has received funding in the framework of the joint programming initiative ERA-Net Smart Energy Systems, with support from the European Union's Horizon 2020 research and innovation programme.

this method is limited in flexibility and accuracy. The current work replaces it with a LightGBM based forecasting model trained on historical utility or community data. LightGBM is chosen for its strong performance in time series prediction, ease of training, and efficient handling of large datasets. The forecasting task is challenging because the model must generate multiple predictions at each time step. It is formulated as a multi-output regression problem spanning a two-day horizon, addressed through either training several independent models, one for each time horizon (for inflow prediction, which has lower resolution) or, for the demand forecasts which are requested at higher resolution, a global model exploiting LightGBM ability to handle missing data (Mangili et al. 2026).

2.1 METHODS

2.1.1 Cleaning and Pre-processing of Data

To ensure the quality and usability of the data for analysis and modeling, we applied a series of cleaning and processing steps. These included:

- *Data distribution and consistency checks.* We examined the characteristics, time evolution, and statistical distribution of each variable and verified consistency with prior knowledge and mechanistic models, identifying unexpected patterns or anomalies.

- *Outlier removal and smoothing.* Outliers were detected and removed. For time series data, smoothing techniques such as moving averages were applied to highlight underlying trends.

- *Missing data analysis and imputation.* We assessed the amount and distribution of missing data to identify non-random patterns. For tabular data, we found Multiple Imputation by Chained Equations (MICE) very effective. For time series, classical methods such as rolling mean and interpolation were preferred. When sufficient training data was available, we found that using machine learning models (LightGBM) to predict and impute missing value can capture complex patterns, enabling more accurate imputations. Finally, we also found that for modeling, using machine learning methods that can handle missingness without imputation, such as LighGBM, is a valid alternative.

- *Units' conversion.* All measurements must be standardized to consistent units across datasets to ensure comparability and avoid misinterpretation.

- *Feature engineering and derivation.* New features can be derived both from prior mechanistic knowledge and from transformation of the raw signals to enrich the dataset and improve model performance. For time-series data, this includes aggregations and transformations such as lagged variables, rolling-window statistics (e.g., local maxima, minima, means, and other window-based summaries).

- *Temporal alignment and normalization.* Time series with different time steps, offsets, or scales must be aligned and normalized to enable integration and comparison across sources, which is particularly important in energy-related applications.

- *Predictive Modelling Preparation.* The cleaned and processed data must be structured for use in time series predictive modelling, including formatting, feature construction and selection.

2.1.2 Energy demand forecasting

The time series of energy demand and water inflow, collected and pre-processed, were used to train a machine learning model for energy demand forecasting. The approach leverages LightGBM as the primary forecasting method for both energy demand and inflow prediction, selected for its ability to combine competitive accuracy with low computational requirements—an essential aspect for deployment in small utilities with limited technical resources. Unlike deep learning approaches—which require large training datasets, significant computational resources, and complex tuning—LightGBM offers a lightweight solution that is well aligned with the practical constraints of local hydropower operators.

The forecasting task is formulated as a multi-output regression problem spanning a two-day horizon, addressed through either independent model (for inflow), one for each time horizon, or a two-models approach (Mangili et al. 2026):

Model 1: predicts the immediate next step, using observed weather variables for maximum accuracy.

Model 2: generates forecasts for all longer horizons (2–576 steps), relying on past observations, calendar features, and the most recent available weather forecast. Missing short term lag features naturally occur for far future horizons and are handled using LightGBM's capability to process missing values.

LightGBM-based modeling was used to train a demand and an inflow forecaster producing accurate short-term predictions across a 48-hours horizon with variable resolution for the demand and 1-hour inflow resolution. The effectiveness of these forecasters was tested in two application involving deterministic Model Predictive Controller (MPC): optimal scheduling of hydropower plant (HPP) production for peak shaving; control of small self-consumption community battery charge-discharge cycles.

The models incorporate:

- Lagged demand or inflow variables, including short-term (5–15 minutes) and longer-term (daily and weekly) dependencies.

- Rolling aggregates, such as local minima, maxima, and averages to capture intra-day variability.

- Weather forecasts, including temperature, irradiance, and precipitation, all crucial for modeling both electricity demand patterns and inflow behavior.

Proper temporal alignment is crucial to avoid information leakage and ensure fair performance estimation. Production scheduling for the hydropower plant relies on forecasts available at the decision time. This is not always possible in training and testing phases, as discussed above, which may lead to overestimation of the predictive performance.

2.1.3 Model predictive control

In the HPP applications, the forecasting models feed a deterministic MPC tasked with smoothing electricity demand peaks through the optimal management of the available water. The MPC incorporates operational constraints (reservoir levels, turbine limits, inflow forecasts) and produces a production schedule designed to minimize daily peak load. We assessed the approach using data from AEM and SES power plants and compared the monthly demand peaks thus achieved by Operator decisions, MPC using perfect information ("oracle"), MPC using LightGBM-based forecasts.

The forecasting models were further applied to a second scenario involving the design and operational evaluation of a community battery for the Lugaggia Innovation Community (LIC).

Demand and PV forecasts were generated using the same LightGBM framework adapted to 5-minute data resolution (576 steps over 48 hours). Weather forecasts included temperature, radiation, and precipitation. Since historical PV forecasts were unavailable, PV generation was assumed known—a reasonable approximation for sizing studies. Forecasts were fed into an MPC planning the charge discharge cycles of the battery so to maximize the community self-consumption.

The goal was to simulate the battery and energy demand behaviours of the LIC community under MPC, and determine how different combinations of photovoltaic (PV) capacity and battery size affect:

- Self-consumption (SC): fraction of PV energy consumed locally
- Self-sufficiency (SS): fraction of total demand supplied by local PV + storage

2.2 RESULTS

2.2.1 Data transformations and pre-processing

For the signals used in Use Case 1 – HPP Scheduling, we manually supervised the data and its distribution, removing only a few outliers. During this process, we identified a frequent inconsistency between the measured flow rate and the reconstructed flow rate, which was derived from production data and reservoir level. We therefore chose to use the reconstructed flow rate, defined as:

(Change in reservoir volume per unit time) – (Water consumption at the given production level per unit time)

Reservoir volume estimates were derived from observed water levels using an interpolated function based on the level–volume characteristic curve supplied by the utilities.

We also analyzed missing data, which in some cases corresponded to extended periods. Imputation was performed using traditional methods (e.g., linear interpolation) only when a small number of consecutive values were missing, and exclusively for simulation purposes in scheduling. No imputed data were used for model training, as we employed LightGBM, which natively handles missing values.

To enrich the dataset, we integrated additional sources of information, specifically weather forecasts from OpenMeteo. These included forecasts for the utility service area which were used in demand modeling and forecasts for the basin catchment area which were used in flow rate prediction.

When using weather forecasts, it is important to align them with the demand prediction horizon. For instance, if the model predicts demand 24 hours ahead, the corresponding weather forecast must be one that was issued at least 24 hours before the target time — in other words, the forecast must reflect the information that would have been available at the moment the 24-hour-ahead prediction is made.

It is important to note that historical data on weather forecasts often only include the latest forecast produced before the target time, that is, typically 12 hours earlier. As a result, models were trained (and sometimes tested) using as inputs weather forecasts produced at most 12 hours before the target time but later applied using longer-term predictions (up to 48 hours ahead), which are inherently less accurate. This mismatch is not optimal, as it may lead the model to overestimate the importance of weather features, and the estimated performance may be overly optimistic. However, when tested with realistic forecast horizons, the models still performed well. Where possible, it is recommended to retain all available weather forecasts, to enable more accurate and horizon-consistent training and evaluation. (Change in reservoir volume per unit time) – (Water consumption at the given production level per unit time)

Reservoir volume estimates were derived from observed water levels using an interpolated function based on the level–volume characteristic curve supplied by SES.

We also analyzed missing data, which in some cases corresponded to extended periods. Imputation was performed using traditional methods (e.g., linear interpolation) only when a small number of consecutive values were missing, and exclusively for simulation purposes in scheduling. No imputed data were used for model training, as we employed LightGBM, which natively handles missing values.

To enrich the dataset, we integrated additional sources of information, specifically weather forecasts from Open Meteo. These included forecasts for the utility service area which were used in demand modeling; forecasts for the basin catchment area which were used in flow rate prediction.

Including weather forecasts requires a careful alignment of weather forecasts with demand forecasts, based on the prediction horizon. For example, to train a model that predicts demand 24 hours ahead, the weather forecast used must have been produced at least 24 hours before the target time.

It is important to note that historical data often only include the latest forecast available before the target time, typically with a maximum horizon of 12 hours. As a result, models were trained (and sometimes tested) using 12-hour ahead forecasts but later applied to longer-term predictions (up to 48 hours ahead), which are inherently less accurate. This mismatch is not optimal, as it may lead the model to overestimate the importance of weather features, and the estimated performance may be overly optimistic. However, when tested with realistic forecast horizons, the models still performed well. Where possible, it is recommended to retain all available weather forecasts, to enable more accurate and horizon-consistent training and evaluation.

2.2.2 Modeling and control

The MPC clearly improves peak shaving efficiency relative to operator decisions when perfect information is available. It reduces both average and maximum daily peaks, confirming the operational feasibility of ML-assisted predictive scheduling in real-world conditions. Improvements remain visible even when using imperfect forecasts, though with less consistency, showing the importance of feeding the MPC with reliable short-term forecasts to achieve reductions in peak demand and more efficient use of reservoir storage, as predictive accuracy directly affects the quality of scheduling decisions.

Further enhancements could be achieved by incorporating additional training data from multiple plants and by exploring the integration of pre-trained foundation models for energy-demand forecasting.

Full methodological details and results for the HPP scheduling pipeline applied to the AEM plant are provided in Mangili et al. (2026).

Concerning the application to battery sizing, the combination of ML-based forecasts and MPC enabled realistic simulations, showing how appropriately sized community batteries can substantially increase local energy autonomy and suggesting that the developed framework provides a solid analytical basis for decision-makers.

3 Prediction of room comfort (temperature and humidity)

The second Austrian use (AUC2) deals with predicting heating and cooling demand of an office building consisting of 36 rooms (see D5). To heat or cool the offices, concrete core activation is used. Several factors specify the operation modes for heating and cooling the Future Base. On the one hand, the outdoor temperature defines whether heat energy is needed or not, but also the required cooling energy. The restricted groundwater extraction does also influence the mode of operation, as permanent cooling with groundwater would exceed the maximum quantity. On the other hand, cooling appliances should be used efficiently throughout the year.

The building is equipped with a range of sensors and meters, with each zone monitored through a consistent set of measurements (e.g. zone and concrete core temperature). In addition, an IoT sensor network installed on the 1st and 3rd floors collects indoor environmental data, including temperature, humidity, VOC, occupancy, window/door status, and brightness. These data, combined with weather forecasts, are used in data-driven models to assess indoor comfort. If predicted comfort levels fall outside predefined thresholds, heating and cooling are adjusted accordingly. This use case integrates both IoT and BACnet data within the platform.

A BRICK-based model defines the required data per room. Data are collected from cloud-based sources (IoT sensors and Open-Meteo) and undergo preprocessing (e.g. 15-minute resolution, outlier removal, gap filling). For each room, variables are structured into: (i) targets (room temperature, humidity, concrete core temperature), (ii) future features (weather and system operation), and (iii) historical features (room conditions and control signals).

Based on this dataset, room-specific ML models are trained and validated to predict conditions up to three days ahead, using lagged inputs (1, 2, and 6 days). Results are visualised in a dashboard, including a comfort indicator, to support building operators in optimising system performance.

3.1 Brick data model development

An overview of the data model structure regarding the IoT sensor network is shown in Figure 1 **Error! Reference source not found.**in which the installed sensors for an office room are illustrated. Further, an HVAC zone which is a logical entity to connect several rooms together, e.g., meeting rooms and offices. In these offices are several different IoT sensors (temperature, humidity, contact, illuminance) placed. The sensors itself provide the meta data of their unit (detailed elaboration on AUC2 and Brick model are presented in deliverable D5).

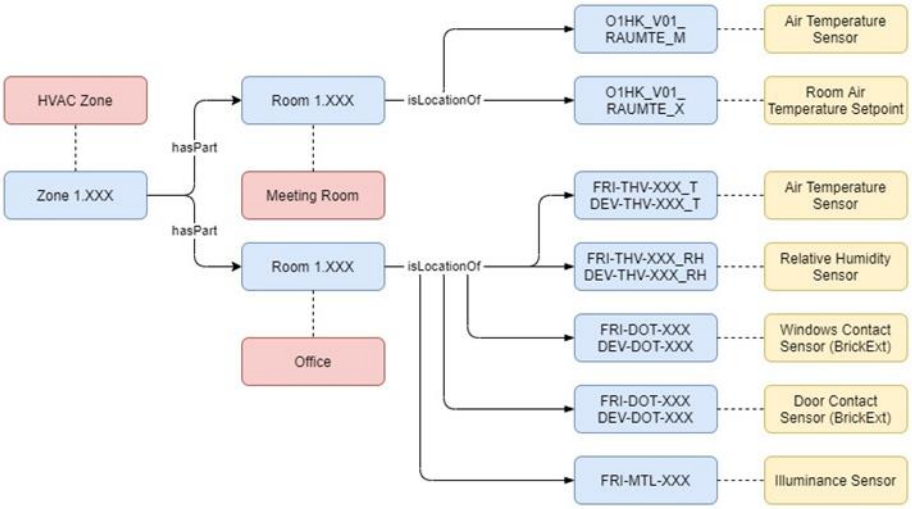

Figure 1: AUC2 – Ontology of the IoT sensors in office rooms

3.2 Dashboard and thermal comfort forecaster for AUC2

The data available for the office building investigated in AUC2 has been collected from various sources, organized using the Brick ontology, cleaned, and explored in a different project. As a useful addition to DIGICITIES, it is valuable to establish both a forecasting model and a user-facing dashboard within. The forecasting model will leverage machine learning techniques in order to predict the thermal comfort of users of the building in AUC2, an important KPI to maintain by the facility management (FM) of the building. The predictions will be made few days ahead and on the level of zones since this is the level at which control can be exerted by the FM. Since the building contains numerous zones, the dashboard has been envisioned to support the FM by rapidly painting an accurate yet detailed picture of the probable future state of the building and allowing for timely interventions.

3.3 Embedding AUC2 dashboard into a docker container

3.4 Data-preparation:

Building sensor data was collected for over a year time range. The sensor data included temperature and humidity meters, window opening sensors, as well as weather data. The sensor data was mapped to rooms with the help of a BRICKs semantic model. Based on the dataset a machine learning model was trained to predict the humidity and temperature of a single room **for a 3 days time horizon.** Two sensor data source was used for the analysis: The HVAC sensor data from the existing BACnet amounting to 2700 timeseries with 47 months of history, and IoT sensor network for the room climate that collected 1200 timeseries with a 17 months of history.

To describe the semantic relations in the building a BRICK model was built containing building topology and HVAC system components as well as elements for the IoT sensors. The connections between the components are described by triples.

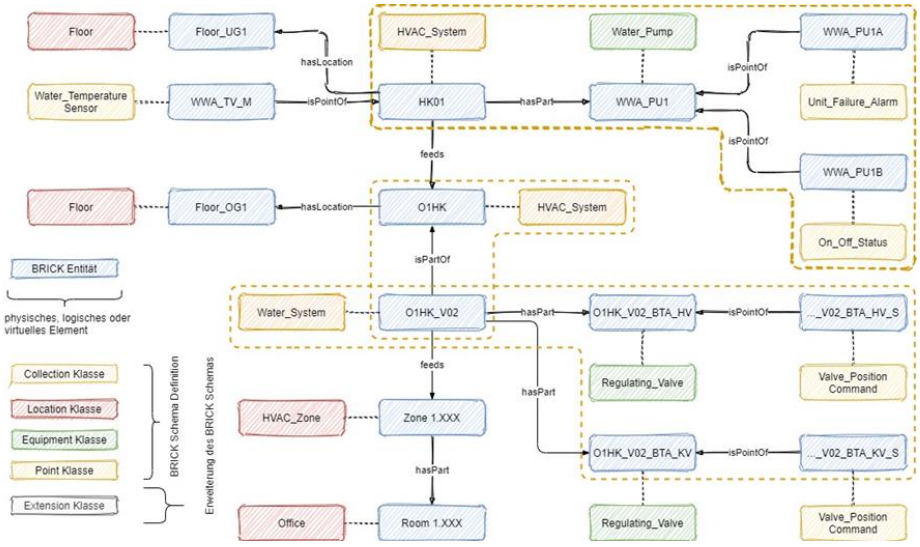

Figure 2: AUC2 – BRICK model for building topology and HVAC system

The BRICK ontology was used to find the sensors corresponding to the rooms (sensor to room mapping). All together 19 rooms from the 1-st floor, and 17 rooms from the 3. floor were extracted from the dataset. For each room 50 features were prepared including room temperatures window/door contact values, outside temperatures, temperature set values, heating return and supply temperature, heating/cooling volume flows. Based on the time-history of these features 3 quantities are predicted: temperature of the concrete core, room temperature and the humidity in the room.

For the predictions the LightGBM Model from the *darts* library was used[1]. This model is based on gradient boosted trees algorithm. It is based on gradient boosting where weak learners are added iteratively to an initial decision tree to improve prediction accuracy, but LightGBM outperforms other gradient boosted models in terms of computational speed and memory consumption. For each of the 36 rooms a separate model was trained and stored in separate file. The training time range was 2022.09.08 – 2023.12.09.

As an implementation a python flask function was written that provides prediction for one room based on the stored data. The prediction is calculated for a specified date. After the room and date is selected in the frontend the backend loads the data history for that room, loads the prediction model from the memory, and runs the LightGBM algorithm to predict the room, and concrete core temperatures as well as the humidity for the next 3 days with a 2 hours timesteps (all together 3*12*3 values). The measured values are also shown next to the predictions, so that the user can get a quick impression about the model accuracy. It should be notified that

[1] https://unit8co.github.io/darts/generated_api/darts.models.forecasting.lgbm.html

timepoints before 12.2024 were part of the training dataset, which explains the apparently higher accuracy of the models.

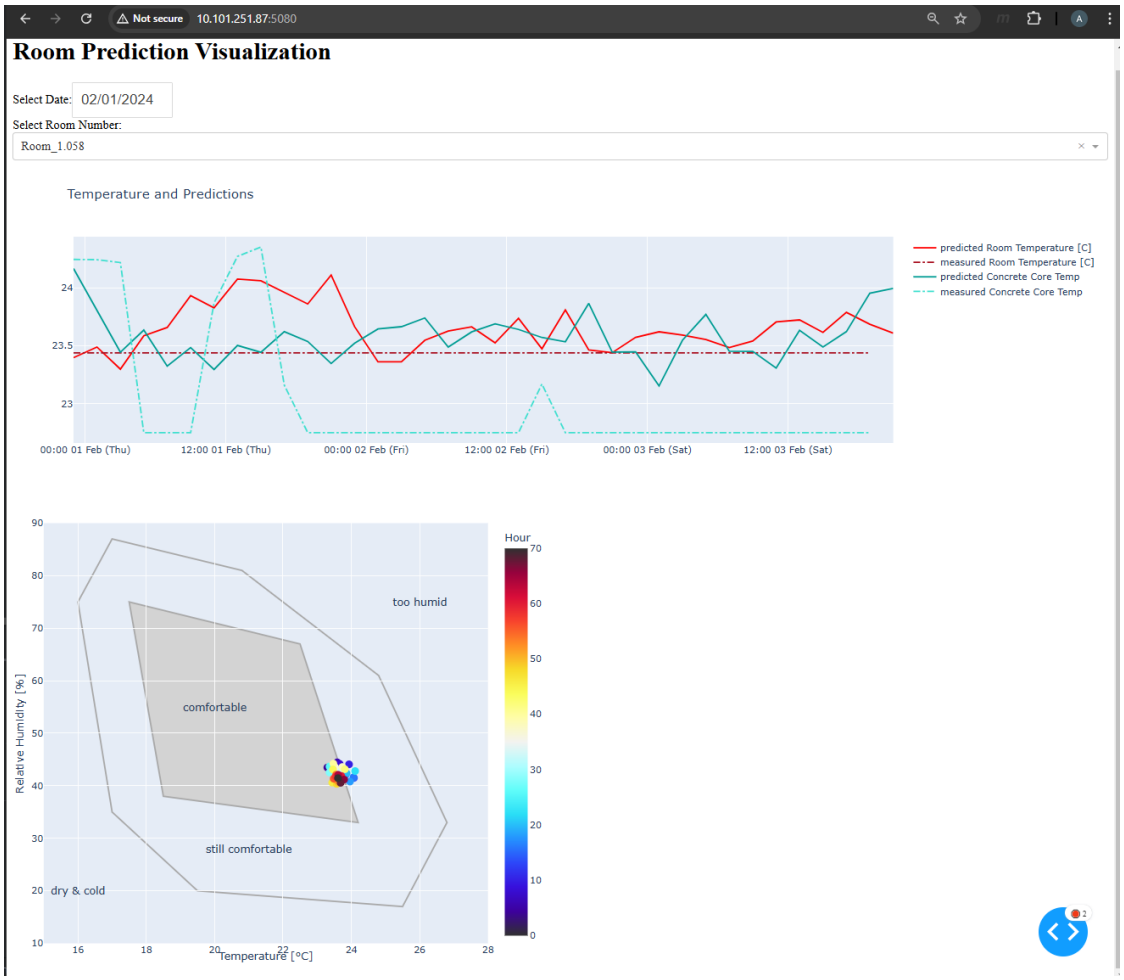

Figure 3 Screenshot of the room comfort prediction dashboard

For the frontend we have prepared a dashboard with python dash. We plot the next 3 days predictions for the temperature as well as the measured values. So that the user can visually compare the predicted values to the measured values and grasp the prediction accuracy.

3.5 Implementation

For an interactive visualization an online dashboard was developed for the use case. The dashboard, and calculation engine (API) was developed with the python flask library. For easy transferability the implementation is containerized with Docker. The dashboard GUI runs in a smaller docker container serving the user requests, and responsible for the visualization, while the backend is served by a bigger container (around 12 GB) and is responsible for fetching the data and running the machine learning model for the prediction of the comfort. The two containers can communicate with each other, if they share the same docker network (specified by the compose script). The client is ideally only communicating with the frontend. The advantage of the two containers, that the firewall could be also set such a way that

the backend would not directly be available from the client, only the frontend could access it, so that the backend would be safely hidden from the user. (For example the firewall would block the port 5085 from outside).

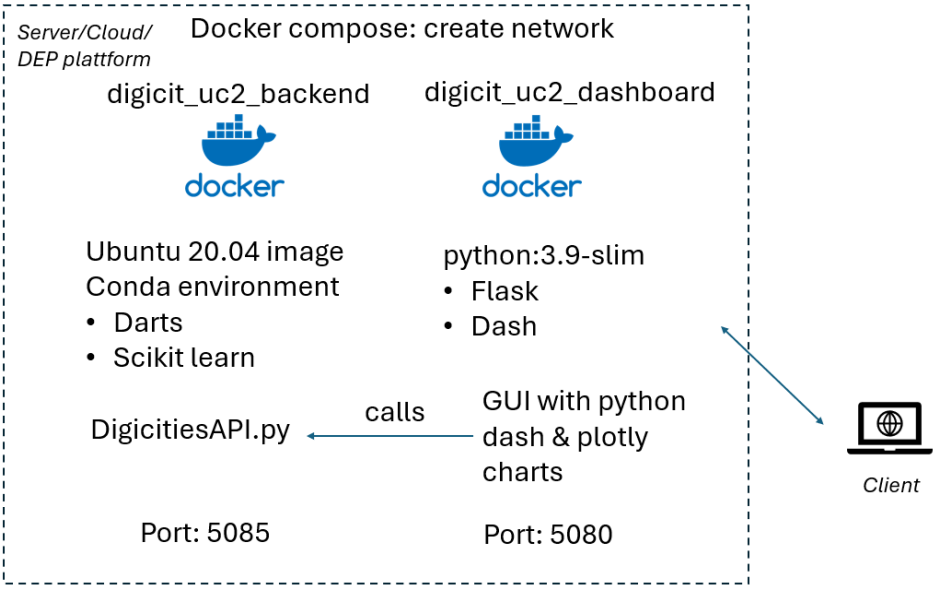

Figure 4 Structure of the docker containers running the dashboard.

4 REFERENCES

Mangili, F., Derboni, M., Zambon, L., Giuffrida, V., & Salani, M. (2026). Enhancing Peak Shaving Efficiency in Small Hydro Power Plants Through Machine Learning-Based Predictive Control. Energies, 19(4), 985. https://doi.org/10.3390/en19040985

FUNDING

This document was created as part of the ERA-Net Smart Energy Digicities project, which was funded through the through the framework of the joint programming initiative ERA-Net Smart Energy Systems' focus initiative Digital Transformation for the Energy Transition, with support from the European Union's Horizon 2020 research and innovation programme under grant agreement No 883973.

© Copyright 2019 – ERA-Net SES