



D2.4 SportE2 How Module complete

Date	Delivery date: M44
Deliverable Number	D 2.4
WP Number	WP2 – Smart Metering Network
Version	2.0
Status	Final
Dissemination level (*)	PU

Abstract

Report and product data sheets for SPORTE2 How module. This report and these data sheets are critical to WP7 (Replication) and WP8 (Dissemination). The deliverable will describe all the capabilities available with the How Module. It will contain technical details for implementation in each possible architecture with a list of needed data and technical requirements. Guidelines explaining the integration layer of the module will be provided with the description of compatible systems. Optimization unit of the How Module will be presented as a standalone tool and functionalities will be demonstrated from required input data to output. These documents will also include guidelines for Energy Smart Metering in Sports and Recreational Facilities.

Versions

<i>Date</i>	<i>Partner</i>	<i>Content</i>	<i>Ver.</i>
03-03-2014	UNIVPM	First version of table of content	0.0
22-04-2014	ISA	Contribution to chapter 2 and chapter 3	1.0
23-04-2014	UNIVPM	First draft complete	1.1
30-04-2014	UNIVPM	Final version	2.0

TABLE OF CONTENTS

TABLE OF CONTENTS	2
LIST OF FIGURES	4
LIST OF TABLES	6
EXECUTIVE SUMMARY	7
1. INTRODUCTION	8
2. WHAT IS THE HOW MODULE?	9
2.1 SYSTEM DESCRIPTION	9
2.2 FUNCTIONALITIES	11
2.3 CONFIGURATION AND GUIDELINES FOR METERING IN SPORTS FACILITIES	12
2.3.1 <i>Energy metering</i>	12
2.3.2 <i>Comfort metering</i>	14
2.3.3 <i>Field monitoring</i>	14
2.4 INSTALLATION.....	15
2.5 INTEGRATION WITH THE SPORTE2 MODULES	17
2.5.1 <i>iEnergy2 Integration Webservices</i>	17
2.5.2 <i>iEnergy3 Integration Webservices</i>	22
3. GETTING STARTED WITH THE HOW MODULE	37
3.1 INTRODUCTION AND SYSTEM REQUIREMENT	37
3.2 USER MANUAL	38
3.2.1 <i>Login Page</i>	41
3.2.2 <i>Presentation View</i>	42
3.2.3 <i>Data Explorer</i>	43
3.3 EXAMPLES FROM REAL CASES	55
3.3.1 <i>Imported energy from Grid</i>	55
3.3.2 <i>Grid import, export and CHP production</i>	55
3.3.3 <i>Thermal consumption/production</i>	56
3.3.4 <i>Water consumption</i>	57
3.3.5 <i>Gas consumption</i>	57
3.3.6 <i>PMV</i>	58
3.3.7 <i>Air temperature and humidity monitoring</i>	58
3.3.8 <i>CO2 concentration level</i>	59
3.3.9 <i>CL2 concentration level</i>	60
3.3.10 <i>Water parameters</i>	60
3.3.11 <i>HVAC parameters</i>	61
3.3.12 <i>Light level</i>	62
3.3.13 <i>Occupancy</i>	62
4. SENSOR OPTIMIZATION UNIT USER MANUAL	64
4.1 INTRODUCTION AND SYSTEM REQUIREMENT	64
4.2 USER MANUAL.....	64
4.2.1 <i>General info</i>	65
4.2.2 <i>Construction</i>	68
4.2.3 <i>Plant</i>	79
4.2.4 <i>Internal</i>	83
4.2.5 <i>Profiles</i>	84
4.2.6 <i>Positioning</i>	86
5. CONCLUSIONS	90
6. ACKNOWLEDGEMENTS	91

7. REFERENCES 92
8. APPENDIX A1: COMPATIBLE DEVICES 93

LIST OF FIGURES

Figure 1: How Module Architecture.....	9
Figure 2 - Cloud Service Architecture.....	10
Figure 3: Possible configurations of the energy metering of the How Module	13
Figure 4: Installation procedure of the How Module.....	16
Figure 5 – Data Server’s data model	18
Figure 6 - IIS Manager - Mime Types Selected	37
Figure 7 - IIS Manager - Adding the .json MIME Type.....	38
Figure 8 - Common User Interface Layout	39
Figure 9 - Top Bar of the common view.....	39
Figure 10 - Top Bar Logotype	39
Figure 11 - Top Bar Active View Control	40
Figure 12 - View mode dropdown selector.....	40
Figure 13 - Top Bar Contents Control.....	40
Figure 14 - Top Bar Session Control	40
Figure 15 - Session control dropdown selector.....	41
Figure 16 - How Module User Interface Login Page.....	41
Figure 17 - Presentation screen example.....	43
Figure 18 - Data Explorer View.....	43
Figure 19 - Data Explorer selection of displayed measurement nodes	44
Figure 20 - Data Explorer selection of multiple variable types	45
Figure 21 - Data Explorer button to show/hide the variable selection	45
Figure 22 - Data Explorer Graph options selection	45
Figure 23 - Data Explorer Grouping selector.....	46
Figure 24 - Data Explorer accumulated variable selection.....	46
Figure 25 - Data Explorer instant variable selection	46
Figure 26 – Variable normalization selector	47
Figure 27 - Data Explorer example of a produced graph	47
Figure 28 - Data Explorer displayed aggregation of data	47
Figure 29 - Data Explorer displayed period change buttons.....	48
Figure 30 - Data Explorer graph buttons.....	48
Figure 31 - Multiple variables displayed on the graph.....	48
Figure 32 - Multiple variables displayed on the graph (stacked)	49
Figure 33 – Comparison/profiling of consumption profiles with the homologous period	49
Figure 34 - Selection of the PMV calculation functionality	50
Figure 35 - PMV Circuit type selection.....	50
Figure 36 - PMV Variables selection.....	50
Figure 37 - Data Explorer accumulated variable summary	51
Figure 38 - Data Explorer instant variable summary.....	51
Figure 39 - Advanced Data Explorer selection	52
Figure 40 - Advanced Data Explorer layout.....	52
Figure 41 – Advanced Data Explorer variable selection zone	52
Figure 42 - Advanced Data Explorer displayed time frame selector	53
Figure 43 - Advanced view calendar selector	53
Figure 44 - Advanced view time of day selection.....	53
Figure 45 - Advanced Data Explorer table data view	54
Figure 46 - Advanced View column display selection	54
Figure 47 - Advanced View data download icon.....	54
Figure 48 - FIDIA Imported electric energy from Grid over a full day.....	55
Figure 49 - SELF energy import, export and CHP production.....	56
Figure 50 - EMTE swimming pool HVAC thermal consumption	56
Figure 51 - EMTE solar system production	57
Figure 52 - FIDIA consumed water over a month	57
Figure 53 - EMTE gas consumption	58
Figure 54 - PMV calculation for FIDIA swimming pool.....	58

Figure 55 - Calculation of discomfort hours.....	58
Figure 56 - SELF Swimming pool air temperature.....	59
Figure 57 - SELF Swimming pool air humidity.....	59
Figure 58 - EMTE Room 1 CO2 concentration.....	60
Figure 59 - SELF Swimming pool CL2 concentration level.....	60
Figure 60 - SELF pilot water parameters (advanced view).....	61
Figure 61 - FIDIA pilot swimming pool water temperature.....	61
Figure 62 - FIDIA pilot HVAC duct air speed.....	62
Figure 63 - EMTE light level measurement at Room 1, Multisport Court and Corridor.....	62
Figure 64 - FIDIA occupancy for the Fitness room.....	63
Figure 65 – SOU opening window.....	65
Figure 66 – (a) Site information section – (b) Latitude and Longitude web service calculator.....	66
Figure 67 – Meteo data section.....	67
Figure 68 – Albedo most common values.....	67
Figure 69 – Shape section.....	68
Figure 70 – SOU construction section.....	69
Figure 71 – External wall section.....	69
Figure 72 – Material layers window.....	70
Figure 73 – Material selection window.....	70
Figure 74 – Material list properties.....	71
Figure 75 – Internal wall section.....	72
Figure 76 – Mixed wall section.....	73
Figure 77 – Screen shot of the construction section window.....	74
Figure 78 – “Add Shadows” window.....	74
Figure 79 – Type of shadow: Trees.....	75
Figure 80 – Type of shadow: Structures.....	76
Figure 81 – Windows section.....	77
Figure 82 – Floor section.....	78
Figure 83 – Roof section.....	78
Figure 84 – Roof section after the flat roof selection.....	79
Figure 85 – SOU plant section.....	80
Figure 86 – Heating section of the Tab “Plant”.....	80
Figure 87 – Cooling section of the Tab “Plant”.....	81
Figure 88 – Humidification section of the Tab “Plant”.....	82
Figure 89 – Dehumidification section of the Tab “Plant”.....	82
Figure 90 – Distribution section of the Tab “Plant”.....	83
Figure 91 – Tab “Internal” of the SOU interface.....	83
Figure 92 Tab “Profile” of the SOU interface.....	84
Figure 93 “Profile” panel of the “Profile” Tab of the SOU interface.....	85
Figure 94 – “Weekly Assignment” panel of the Tab “Profile” of the SOU interface.....	85
Figure 95 – Tab “Positioning” of the SOU interface.....	86
Figure 96 – Optimization objective options of the SOU.....	87
Figure 97 – Output window of the SOU.....	88
Figure 98 – Sensor location guidelines.....	89

LIST OF TABLES

Table 1: List of macro functionalities provided by the How Module 11

EXECUTIVE SUMMARY

This Deliverable constitutes the data sheet of the How Module, which is the Module devoted to the smart metering in Sports and Recreational Buildings. As data sheet, the report includes all the information needed to implement, install and use the Module.

In details, the second chapter introduces the Module functionalities with the architecture adopted to provide them. Then, the procedure to implement the Module in a real facility is illustrated together with the integration of the other Modules (When, Why and Where). Since the How Module is a multi-vendor system, the chapter introduces general information about the compatibility with the technologies present in the market.

The third chapter illustrates how to get started with the Module, which scope is to provide the user manual dedicated to the final user (building manager/owner) for the interaction with the Module functionalities and GUI. The initial part is dedicated to give information on the system requirement needed to allow the Module operation. Then, each functionality available in the GUI is described with the related commands. Finally, some examples of real case studies are reported to show the Module in practice.

Chapter four introduces the Sensor Optimization Unit (SOU) user manual. As described in the previous deliverables, this is a stand-alone software tool for the optimization of T/RH sensor location in sport facilities. After the description of the system requirement needed to use the software, the chapter illustrates in details how to interact with the user interface to apply the whole optimization procedure from data input to output.

Given the requirements defined in the Deliverable D2.1 and the development of a dedicated product to satisfy those requirements, showed in the Deliverable D2.3, the scope of this report is to connect the two deliverable and provide a general overview of how to perform the smart metering in these buildings using the How Module.

1. INTRODUCTION

This document describes the final How Module and introduces its data sheets as a product to be deployed in the market.

The How Module is a smart metering system dedicated to sports and recreational buildings able to provide aggregated information about the energy (consumption, usage, exchange and generation) and the comfort level in the facility. The main objectives and functionalities illustrated in the Deliverable D2.1 are modeled here as a final product after a pathway that passed through the first prototype (Deliverable D2.2) and the system deployed in the three pilot cases (Deliverable D2.3). The reader will find here a complete description of how to implement and use the Module in a real sports facility. This user manual aims to provide a set of information to support the design of the Module following the approach presented and to interact with the software tools developed. Two tools include all the functionalities achievable with the mentioned approach:

1. The How UI: a web-based application to read an process energy and comfort data;
2. The Sensor Optimization Unit (SOU): a stand-alone tool to define the optimal location of the T/RH sensors in large spaces.

The deliverable describes in details how to install and use these tools, the fields of application and benefits. The user manuals explain each functionality, needed input data, range of application and limits. Practical examples from real cases support those descriptions.

The How Module comprises both hardware and software tools. Since the hardware installation is strictly related to the installation of the Automation Server, its argumentation references to the Deliverable D3.4 that describes in details the procedure to install and configure all the components. For this reason, this report presents the procedure to configure the system once that the sensor network is installed together with the Automation Server. On the other side, the description includes a list of the compatible hardware to build the sensor network, to get needed data for the smart metering and for the other Modules (integration). All these information constitute the guideline to support the energy smart metering in sports facilities.

2. WHAT IS THE HOW MODULE?

This chapter provides an overview of the How Module with its functionalities, architecture and implementation. The metering Module could be deployed as stand-alone solution or integrated with other SportE2 Modules and its capability of integration is thus described.

2.1 System description

In general terms, Figure 1 describes the generic architecture of the How module which is a smart metering system. It is mainly composed of a field level, a local network, a cloud service for data processing and a user interface. The system is supported also with a stand-alone tool, named Sensor Optimization Unit, for the design of the sensor network in large spaces.

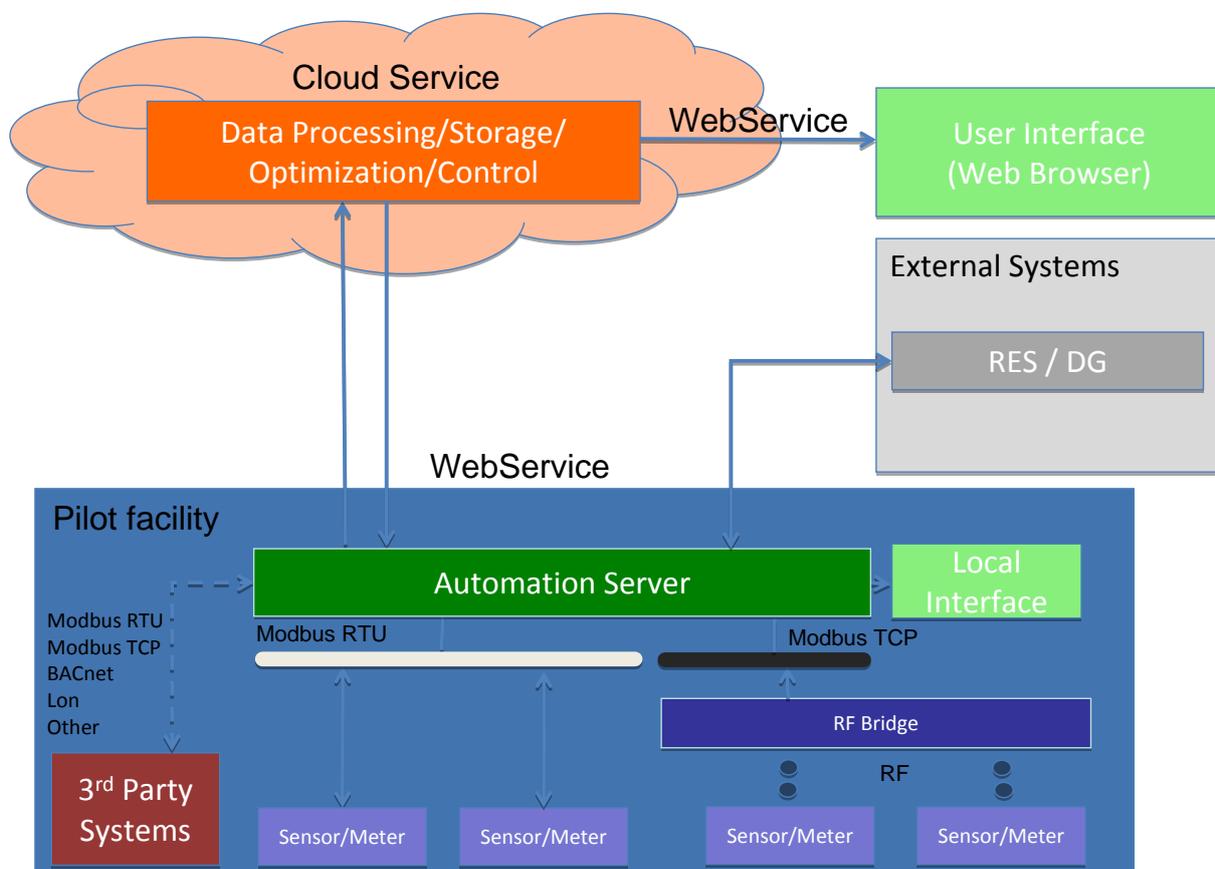


Figure 1: How Module Architecture

In the proposed architecture data flows from the sensors/meters installed in the pilots to the Automation Server using two main alternative interfaces, depending on the sensor/meter type and its communication capabilities. Wired meters, such as the iMeter or the remote I/O modules found in the pilots make use of the Modbus RTU interface provided by the Automation Server. For all the wireless sensors, such as the NetMeter, iPoint or iPointCO₂, a RF to Modbus Bridge is used to convert the RF proprietary protocol into the Modbus TCP protocol, which can be queried by the Automation Server.

Other 3rd party metering systems already available in the pilots, such as Building Management Systems, can be also easily integrated using the mentioned protocols or any of the other field bus protocols available on the Automation Server (such as BacNET, Lon network).

Furthermore, other type of sensors can be used provided an appropriate protocol bridge is available, which is always the case.

Automation Server retrieves the data from the available sensors and meters using the field bus protocols acting as a data logger. Data retrieved by the Automation Server can be used locally to feed its local control algorithms and Local Interface, or can be read by other modules, such as data processing, storage, optimization and control modules using the provided Control Network communication protocol. The remote User Interface and Optimization module (Why) uses the Integration Protocols to retrieve historical and processed data from the Data Server.

External systems, such as RES or DG systems, can be also integrated in the How Module through the Automation Server interfaces through its Control Network communication protocol. The external systems interface can be used to retrieve measurements/production data from the connected external systems and to control them.

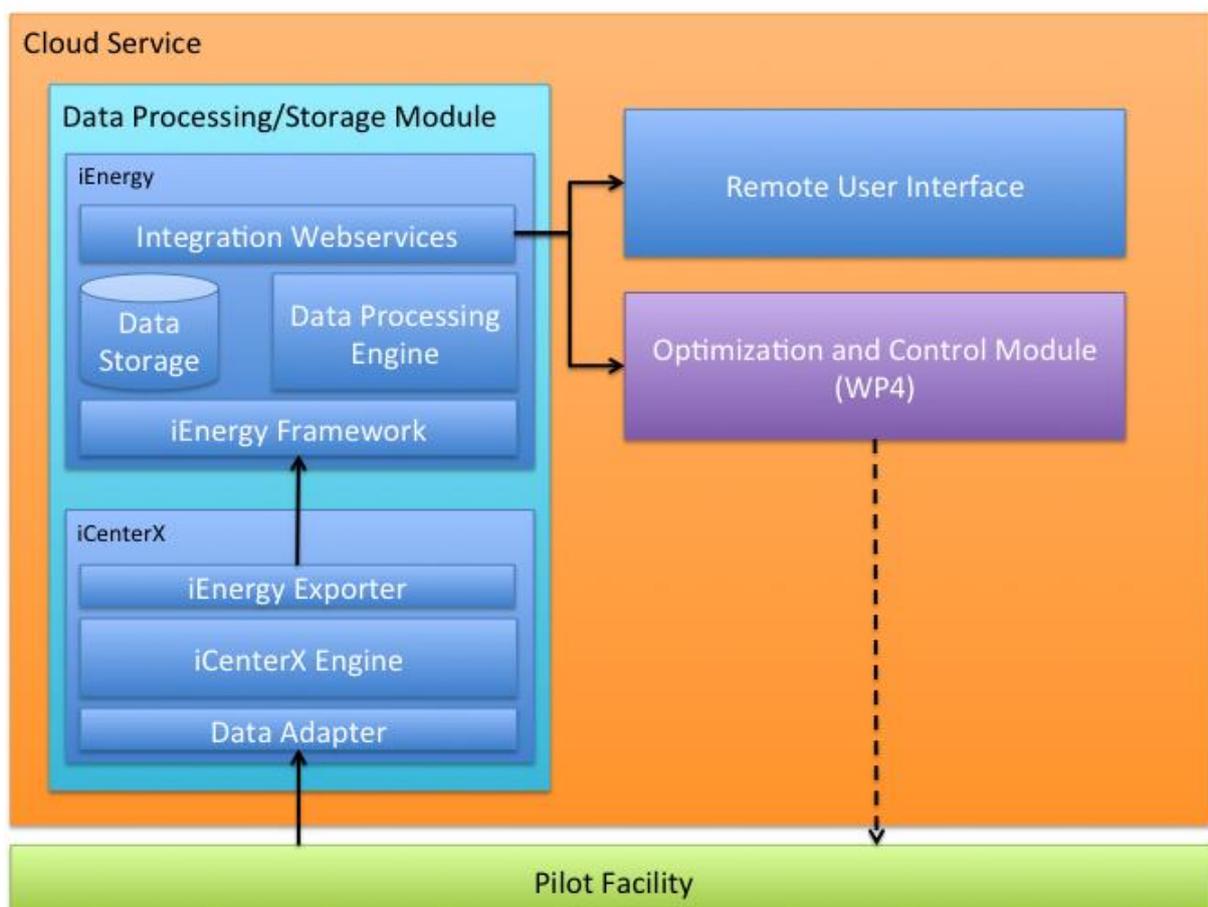


Figure 2 - Cloud Service Architecture

Figure 2 depicts in detail the Cloud Service architecture. In this architecture three main modules can be identified: the Data Processing/Storage Module which is responsible for collecting and storing processed data from the pilots; the Optimization and Control Module, in the scope of WP4.

The Data Processing/Storage Module is composed of two main components. The first component, the iCenterX, is responsible for the communication with each pilot facility (specifically with the Automation Server) in order to retrieve the data collected from the

sensors and meters installed. This component, retrieves the data, adapts its format and then delivers it to the other component. The second component, the iEnergy, is responsible for storing and processing (i.e. perform data aggregation) the data retrieved through the iCenterX component and to make the stored historical data available to the external modules or to the Remote User Interface.

2.2 Functionalities

The How Module is a smart metering system able to provide high-level data on how the energy is used within the building and with which level of comfort. The following table recaps the main features available with the Module:

Table 1: List of macro functionalities provided by the How Module

Module functionalities		Description
Knowledge of real-time energy consumption via sub-metering and proper visualisation/analysis	Total Electricity consumption	Metering of the total electricity consumption (whole building)
	Total Gas consumption	Metering of the total gas consumption (whole building)
	Total Water consumption	Metering of the total water consumption (whole building)
	PV monitoring	Metering of the electricity produced by the PV system
	Solar Thermal System monitoring	Metering of the heating energy produced by solar thermal system
	Biomass monitoring	Metering of the heating energy produced by biomass boiler
	Cogeneration monitoring	Metering of the heating and electrical energy produced by CHP unit
	Area's Thermal Consumption	Metering of the total thermal consumption (heating energy) of the selected areas (e.g. Swimming pool, gym etc.)
	Area's Electric Consumption	Metering of the total electrical consumption of the selected areas (e.g. Swimming pool, gym etc.)
Grid Exchange	PV grid exchange	Metering of the electricity exchanged with the grid (supply and taken)
	CHP exchange	Metering of the electricity supplied to the grid
	Wind system grid exchange	Metering of the electricity supplied to the grid
	Real Time pricing for load management	Retrieving of the actual electricity price for load management (in case of smart grid)
Knowledge of the comfort parameters and air quality	Indoor Temperature	Optimized sensors network for the measurement of the air temperature in the selected areas of the building
	Relative Humidity	Optimized sensors network for the measurement of the relative humidity in the selected areas of the building
	CO ₂ levels	CO ₂ sensor in the selected areas of the building
	Cl ₂ levels	Measurement of the Chlorine content in the air of the swimming pool (index of air quality and potential use for optimized ventilation control)
	Area's Thermal Comfort	Evaluation of the comfort indices for the selected areas. Comfort indices evaluated using a methodology dedicated for typical sport activities
Knowledge of the comfort parameters	Water temperature (pool)	Measurement of the pool water temperature

and water quality	pH and Cl levels (pool)	Measurement of the pH and Chlorine content in the pool water (water quality and potential use for optimized water treatment control)
Software functionalities with data storage, analysis and dedicated user interfaces	User Interfaces	Set of user interfaces available for the manager that provides data analysis results, comfort and usage information.
	Comfort evaluation	From sensors data the comfort level is evaluated and shown for the selected area.
	Consumptions Tracking	Tracking of the energy usage (or other variables such as air temperature or humidity) over time. This way, users can evaluate variables over time and account its increases/decreases or irregularities.
	Carbon Accounting	Carbon accounting functionality should show a graph showing carbon accounting for the facility. It may show total and normalized values by area or end-users over time (days, months, years).
	Longitudinal Benchmarking	User interface gives the possibility to compare/benchmark current consumptions with a pre-defined baseline period; this can be produced by displaying a detailed line graph comparing current consumptions with the previously defined baseline period or by displaying an aggregated bar graph for comparison with a baseline consumption.
	Load Profiling	Load profiling is the method used to establish a relationship between the energy use and time of day. Changes to the normal profile are indicators to abnormalities due to scheduling errors, unexpected or irregular equipment behaviours (such as malfunctions).
	Simple Baselines	Creation of the baseline of consumptions that characterize the building.
	Energy savings	Evaluation on the amount of energy saved in a determined period of time (e.g. One year, one month).
	Anomaly Detection	Using the consumption profile, it is possible to predict consumptions over time. By comparing the computed prediction with the data obtaining from the sensors, it is possible to identify abnormal energy consumptions.
Optimization of temperature and humidity measurement in large spaces	Temperature and Humidity sensors positioning in large spaces	A dedicated tool (SOU: Sensors Optimization Unit) for the system designer that provide the number and position of temperature/RH sensors.
Data exchange	Interface with the other modules	Supply of measured and processed data to the other modules.

2.3 Configuration and guidelines for metering in sports facilities

Sports buildings are complex systems and consequently a metering system for them will be complex as well. For this reason, the How Module can be deployed with different configurations to monitor the energy and comfort. This solution guarantees the optimal way to satisfy the customer requirements.

2.3.1 Energy metering

In some cases the knowledge of the merely whole building consumption is not enough to provide a clear figure of its consumption, but, on the other side, a detailed breakdown metering considering all possible areas and subsystems is not feasible from an economic point

of view. Three possible configurations are thus defined, as shown in Figure 3, to satisfy the customer requirement.

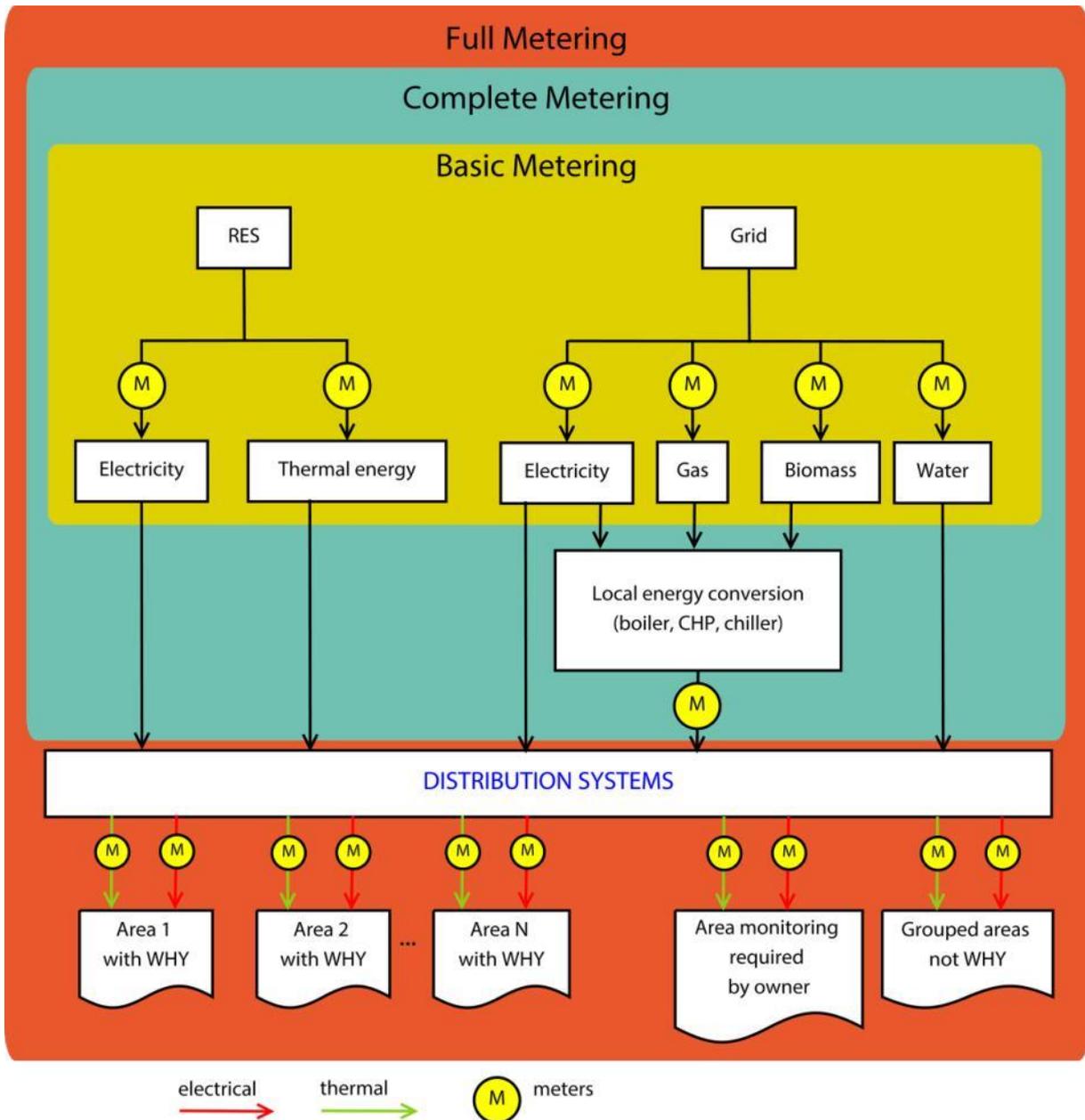


Figure 3: Possible configurations of the energy metering of the How Module

The difference between these configurations lays on the level of sub-metering reached:

1. Whole building (input energy carriers): measurement of the total building consumption. This comes from the grid and RESs through different carriers which are metered;
2. Local energy conversion systems (boiler, CHP, chiller etc.): metering of the energy conversion systems in order to measure how much energy is converted and, if required, with which efficiency;
3. Functional areas: metering of the total electrical and thermal energy consumption of an area or group of areas.

The energy metering procedure demands also the occupancy measurement. The occupancy of the whole facility is needed for the data normalization and it can be measured using two methods:

1. Integration of an existing access control system (if any);
2. Installation of occupancy sensors at the main entrance;

In some cases, the installation at the main entrance of the facility could be not feasible, so the occupancy sensors are installed at the strategic positions in order to obtain the total occupancy in any case. While in other cases the measurement system includes an integration between existing access control system and new sensors. The Full Metering configuration comprises the monitoring of the occupancy in the areas considered relevant (e.g. for the implementation of a scenario of the Why Module).

2.3.2 Comfort metering

The comfort monitoring approach applied to pilot cases follows the methodology described in the Deliverable D2.1. The How Module is the first monitoring tool in the market to make use of the Fanger's theory to assess the level of comfort in the different areas. A dedicated calibration, described in [1], allows the implementation in three different sports environments:

- Swimming pool
- Gym
- Fitness room

When applied, this functionality provides two types of information about the thermal conditions of the environment: a real-time value for day-by-day analysis and a long term indicator for long period analysis. The approach is based on the *PMV* (predicted mean vote) model, which is an index to predict the vote of a large group of persons about the thermal environment. The *PMV* is calculate as function of the four ambient parameters (air temperature, relative humidity, air velocity and mean radiant temperature) and two personal parameters (clothing thermal insulation and metabolic activity). Calculated in this way, it is a real-time quantity to assess the actual indoor thermal comfort. When averaged with a weighting factor based on predefined comfort bands, the *PMV* provides information about the long term indoor conditions, providing thus a comfort figure, that can be compared with the related energy figure. An example of the comfort figures available with the How Module is shown in the Section 3.3.6.

The How Module provides a calculation of these indicators with a specific calibration for sports facilities that considers the typical sports ensembles and activities. Concerning the measurement of environmental parameters, a simplified sensors network can be applied in the case of large areas without radiating sources and taking into account that most of the regulations for sports environments limit air velocity at values < 0.2 m/s. Once verified these conditions, the comfort monitoring of each area needs of an air temperature/humidity sensor installed inside the room and a water temperature sensor for swimming pools. It can be applied in none, some or all the areas in function of the customer requirement or of the installation of the other Modules (e.g. optimization of the HVAC). In the case that the conditions above are not verified the installation of air velocity and mean radiant sensors could be required. For this reason, a case-by-case analysis is needed to apply the comfort monitoring proposed by the How Module.

2.3.3 Field monitoring

The field monitoring consists of the measurement of all the quantities to be measured not for the energy and comfort metering but to supply data needed by the other Modules of the SportE2 BMS or by a specific requirement of the building manager/owner. They are of different nature and their definition depends case-by-case. For example, the field monitoring includes air duct sensors to measure air velocity, temperature and humidity which are typically used for the control of Air Handling Units. The design and deploy of sensors for the field monitoring is complex in sports facilities, especially in case of large spaces. In fact, the indoor air temperature and humidity measurement can be affected by the uncertainty of sensors positioning when large spaces occurs, which is reflected in a poor monitoring and control of the environment. For this reason, a dedicated tool of the How Module called Sensors Optimization Unit (SOU) supports the installation of the temperature and humidity sensors. The installation kit of the Module includes the application of this tool done by the system designer. Chapter 4 illustrates the SOU functionalities and the user manual.

The air quality is monitored as well. It consists of the measurement of the CO₂ level and the CL₂ in the air. The chlorine level in the air is specific problem to be addressed in the swimming pools. In case of low air changes rates or high pool evaporation rates, the chlorine in the air increases and can thus be used as an indicator for the better control. The water quality is also a specific issue for the swimming pools. In this case, three quantities can be monitored with the How Module: pH, Chlorine and temperature. The continuous monitoring of those quantities allows the optimal management of the pool water treatment to guarantee the water quality for the users. The control and optimization Modules make use of the water quality parameters in case they are installed with the How Module.

Appendix A1 reports the complete list with sensors employed with the How Module. The list comprises all devices compatible with the Module and already tested in real cases, but other devices from the market can be integrated.

2.4 Installation

The hardware installation of the How Module lays on the installation of the Automation Server, the field devices (such as sensors and meters) and the communication channels for connecting the field devices to the Automation Server. The Deliverable D3.4 illustrates this procedure in details.

The definition of the architecture and implementation of the Module is a complex process and must be analysed and defined in a case by case basis, requiring in some cases several iterations of technical visits to the pilot in order to better evaluate the feasibility and costs of the idealized monitoring points.

The main procedure to instantiate the How Module on a specific facility must follow the steps depicted in the diagram on Figure 4:

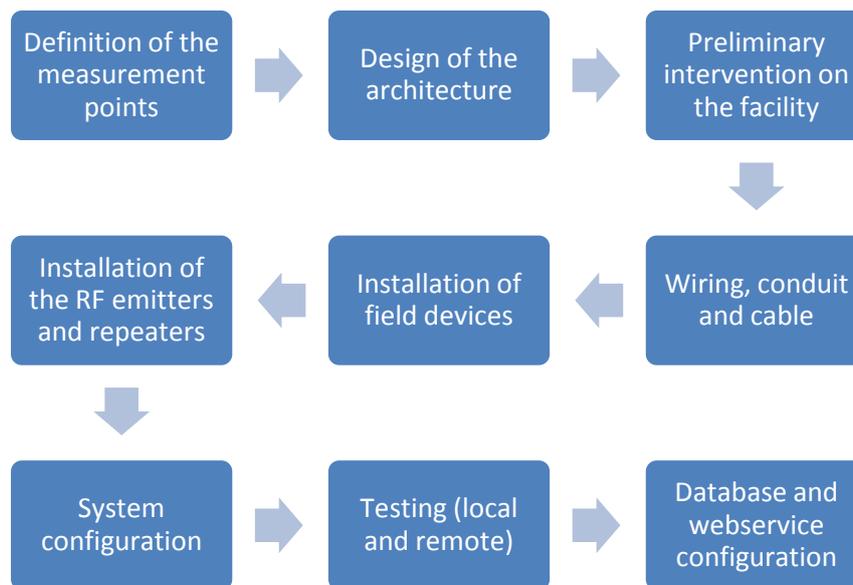


Figure 4: Installation procedure of the How Module

Definition of the measurement points: after the preliminary configuration of the Module (definition quantities needed to be measured), a technical survey allows the identification of the field constraints and the definition of how (i.e. which sensor type) and where each defined variable can be measured. The main output of this phase is the installation plan with the detailed description of all sensors required to satisfy the Module configuration.

Design of the architecture: the architecture of the Module to instantiate the planned sensor networks is designed. This phase concerns the location and specification the communications networks (wired and wireless) for assuring the data flow for all the installed devices in the facility, according to the field constraints, topology of the building and its plants.

Preliminary intervention on the facility: this activity is needed to prepare the building for the module installation in case of retrofit. The local technician makes the intervention with the supervision of the Module provider and all the needed technical operations are identified.

Wiring, conduit and cable: once that the building is ready, the installer performs the needed wiring for the physical connection of the devices, according to the defined architecture and installation plan.

Installation of the devices: this phase could coincide with the previous and consists in the installation of all the module's devices (field sensors or meters, Automation Server, RTUs etc.) deployed in the building according to the defined installation plan.

Installation of the RF emitters and repeaters: the sensor network comprises both wired and wireless sensors. Wireless solutions are fundamental especially in case of building retrofit where some zones are not easily reachable by wired communications. The design of the architecture includes also the number and position of the RF repeaters/emitters, but the effectiveness of their locations needs to be assessed in this phase. If the coverage is not guaranteed, the architecture is adjusted accordingly.

System configuration: after all the devices are installed and the communication networks established, the configuration phase needs to be performed. This phase comprises the configuration of the parameters of each device to assure the correct measurement and communication to the data logger (Automation Server). The main configuration tasks are:

- **Electrical Meters:** Configuration of the correct current transformer ratios and polarity and configuration of the Modbus address;

- I/O Modules: Configuration of the Modbus address and configuration of the correct conversion factors for the measured quantity on the Automation Server;
- RF/Modbus bridge: Configuration of the installed wireless devices IDs and configuration of correct the IP network parameters;
- Automation Server: Configuration of the correct communication parameters and correct sensor parameters (e.g. Modbus IDs and registers). It is also necessary to assure the network conditions (i.e. firewall and/or port forwarding on the facility network) to permit the incoming internet connections to the http and https ports of the Automation Server.

Testing (local and remote): each sensor is tested in order to verify the communication (both locally and remotely), the storing in the AS and the quality of the measured data.

Database and web-service configuration: the data measured by sensors are stored locally in the AS which accessed by the usage of web services. The correct mapping between the tag IDs of the data server (iEnergy) and the internal variables of the AS corresponding to the field devices is necessary. This mapping allows the correct retrieval, processing and storage of the collected data from each one of the installed sensors in the iEnergy platform which will be afterwards made available for the module's user interface or to the other SportE2 modules which need to consume the data.

2.5 Integration with the SportE2 Modules

The How Module acts as a smart metering system in case of stand-alone deployment, but comprises additional monitoring functionalities in case of integration with the other Modules of the SportE2 BMS. The integration serves two main functions:

1. To provide live, short-term and long term historical metering data for the other modules;
2. To provide additional computed variables used for the control or optimization.

The physical integration passes through the Automation Server, which operates as central unit and collector of the BMS. Whereas, the software integration interface made available by the data server allows the sharing of historical and processed data of the facility with the other Modules. The technology chosen to provide such integration is a Web Service, which is fully described in detail in the following sub-sections. During the development of the SportE2 platform, two versions of the data platform were used: the iEnergy2 in the first stage of the project and the new version iEnergy3 in the final stage of the project. In order to maintain compatibility with previously developed modules, despite of being considered obsolete and recommended to not being used in future, the iEnergy2 platform will be maintained online and updated with the data retrieved from the field.

2.5.1 iEnergy2 Integration Web services

The service revolves around the following main concepts:

- Local – represents a physical location that is being monitored. In practice this entity represents each client's building.
- Unit – corresponds to the physical equipment (Gateway) that is used to monitor energy use. Each location has an associated unit.
- Device – represents a physical sensor.

- Tag - represents a measurement end point that stores the energy use readings. A tag may measure electrical consumption, water consumption, gas consumption, temperature, etc.

The model shown in Figure 5 represents the data model that can be accessed through the integration protocols in order to retrieve processed and historical data from the pilot’s devices.

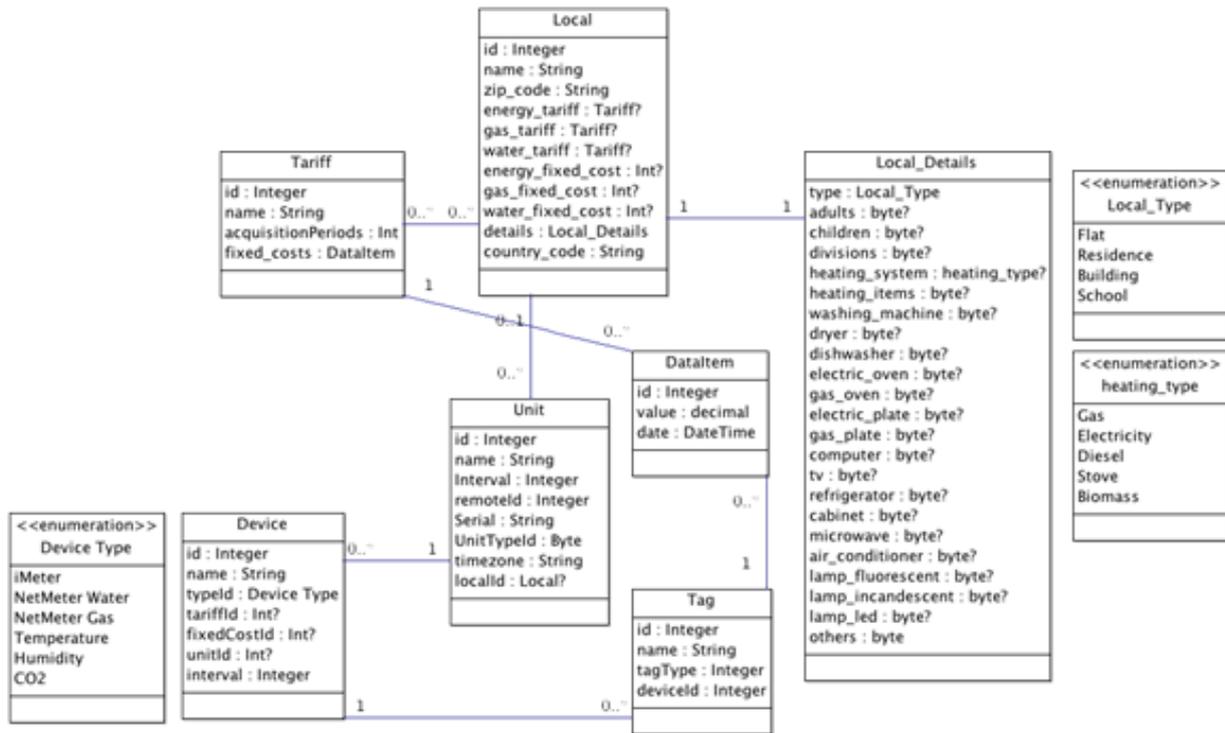


Figure 5 – Data Server’s data model

The following sub-sections describe the particularities of the Web Service that can be used to integrate data from the facility on external modules.

2.5.1.1 Authentication

Invoking the Web Service methods requires the user authentication using login and password. If the authentication is successful, the client will be returned an authentication token that should be provided in subsequent requests to the Web Service. The signature of the login method is:

```

ResultCode Login(string login, string pass, out string token)
    
```

Note that all methods provided by this Web Service have a ResultCode return type. The authentication token is returned by the output parameter that is provided when calling the function.

The returned token will allow the user to execute the Web Service methods given that his profile type allows him to do so. The provided token will expire after 60 minutes of inactivity in which case the user is required to login again and obtain a new token.

A utility method has been provided that allows the client to refresh the status of the token, the signature of the method is:

```
ResultCode RefreshToken(string token)
```

This method is useful in scenarios where the client may not have any requests for a period longer than 60 minutes but does not want to login again. In these cases the client simply needs to ensure that the token is refreshed before the 60 minute limit is reached granting another 60 minutes of inactivity.

2.5.1.2 Data Structures

In this section, all the common data structures and enumerations that are used in this service are specified. Data structures that are specific to a method or set of methods are specified along with those methods.

2.5.1.2.1 Outcome of an Operation

All method invocations return a code that indicates the success or failure of its execution. The possible values are encoded in the following enumeration:

```
public enum ResultCode : byte
{
    Success = 1,           // Operation was successful
    UnexpectedError = 2,  // Error during execution
    InvalidUser = 3,      // Invalid user
    InvalidParameter = 4, // Invalid parameter
    NoData = 5,           // Missing data for required operation
    Unauthorized = 6,     // User does not have the required permissions
    ImpossibleConversion = 7 // The measurement unit conversion is impossible
}
```

The first three values are common to all operations of the service, the remaining are specific to each method and will be detailed in each case.

2.5.1.2.2 Tag

The Tag class is used to represent a tag (measurement end-point) in the system and is serialized between clients and the platform. The structure is defined as:

```
public class Tag
{
    public int Id { get; set; }
    public string Name { get; set; }
    public int UnitId { get; set; }
    public int DeviceId { get; set; }
    public int RemoteId { get; set; }
    public int TagTypeId { get; set; }
    public int? LocalId { get; set; }
}
```

2.5.1.2.3 Device

The device class is used to represent a device in the system and is used to serialize device information between clients and the server. The device class is defined as:

```
public class Device
{
    public int Id { get; set; }
}
```

```
public string Name { get; set; }
public int UnitId { get; set; }
public int DeviceTypeId { get; set; }
public int AcquisitionPeriod { get; set; }
public int? TariffId { get; set; }
public int? FixedCostId { get; set; }
public int RemoteId { get; set; }
}
```

2.5.1.2.4 Unit

The unit class is used to represent a unit in the system and is used to serialize unit information between clients and the server. The unit class is defined as:

```
public class Unit
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string Serial { get; set; }
    public int CommunicationInterval { get; set; }
    public int RemoteId { get; set; }
    public byte UnitTypeId { get; set; }
    public string TimeZoneId { get; set; }
    public int? LocalId { get; set; }
}
```

2.5.1.2.5 Group

The group class is used to represent a group in the system and is used to serialize group information between clients and the server. The group class is defined as:

```
public class Group
{
    public int Id { get; set; }
    public string Name { get; set; }
    public int ParentId { get; set; }
}
```

2.5.1.2.6 User

The user class is used to hold the a user's information in the system. The class is defined as:

```
public class User
{
    public int Id { get; set; }
    public string Login { get; set; }
    public string Email { get; set; }
    public int Domain { get; set; }
    public string Name { get; set; }
    public string Address { get; set; }
    public string Phone { get; set; }
    public string CellPhone { get; set; }
    public string Password { get; set; }
    public byte Language { get; set; }
    public byte Currency { get; set; }
    public byte Authentication { get; set; }
    public bool IsTrusted { get; set; }
}
```

}

2.5.1.2.7 Obtaining Lists of Structures

Several methods of this service return lists of structures. In some situations these lists can be very large. Since the communication channel imposes limits on the maximum size of each message sent between client and server, there may be cases where one may not be able to obtain all data with just one invocation.

As an example consider a request for data in a given time interval. If the amount of data to be returned is large, the service may not immediately provide all the data. One way of overcoming this limitation is by breaking the original interval into smaller non-overlapping consecutive intervals that the communication channel can handle.

2.5.1.3 Methods

Below the most significant methods of the Web Service are described.

2.5.1.3.1 Get Data

The following method is used to obtain a list of consumptions or readings in a given interval:

```
ResultCode GetConsumptions(string token, int[] tagId, DateTime beginDate,
    DateTime endDate, AggType aggregationType,
    int measurementUnitId, out List<DataItemAgg> consumption);
```

The arguments are:

Argument	Meaning
token	The user's access token
tagId	<i>Ids of the tags for which consumptions values are obtained</i>
beginDate	Start date of the interval
endDate	End date of the interval
aggregationType	Id of the type of intended aggregation (see note 1 below)
measurementUnitId	Id of the intended measurement unit
consumption	<i>List of requested consumptions</i>
Return value	Outcome of operation

Note 1: The aggregation type may take values: 0 – Instant, 1 – Hourly, 2 – Daily and 3 – Monthly.

Note 2: When aggregated data is requested, data is truncated by hour/day/month and treated that way (i.e. if daily aggregation is requested and the date provided was 15-01-09, the service will return the consumptions of each day of January 2009).

The values that may be returned by the method are:

ResultCode	Meaning
InvalidParameter	tagId not found, endDate < beginDate, aggregationType not found or measurementUnitId not found
InvalidUser	Wrong token Unauthorized access to ws method
Success	Operation occurred with success
UnexpectedError	Unexpected error with the database
NoData	No consumptions found

Unauthorized	Unauthorized access to tag
---------------------	----------------------------

2.5.1.3.2 GetTags

The method to get a list of tags that the user has permissions to access:

```
ResultCode GetTags(string token, out List<Tag> tags);
```

The arguments are:

Argument	Meaning
Token	The user's access token
tags	The tags list
Return value	Outcome of the operation

The values that can be returned by the method are:

ResultCode	Meaning
InvalidUser	Wrong token unauthorized access to ws method
Success	Operation occurred with success
UnexpectedError	Unexpected error with the database
NoData	There are no tags that the user has access

2.5.2 iEnergy3 Integration Web services

This section describes the Data Server application programming interface (API). This API is provided by the iEnergy3 for the other application or components to use. The API tries to conform to the RESTful principles of the HTTP protocol thus providing a machine friendly, robust and predictable interface to the system functionalities. Except otherwise noted, all data is to be transmitted using the JSON data format.

Before proceeding to the specification of the above mentioned API, there are four main concepts that are central to the Data Server platform and that should be clearly understood. These four concepts are:

- **Local:** A physical location that has the remote metering hardware installed. Locations usually correspond to a house or a building where the metering hardware is installed.
- **Unit:** The hardware that is located at the customer's location and that receives the data readings from devices. A unit usually communicates with the server using a TCP/IP connection.
- **Device:** Another piece of hardware that is located at the site being monitored. Devices can read several parameters (current, power, etc.) and send the data that has been read to its governing unit. There can be more than one device associated to unit. A device can be, for instance, a socket plug that is used to monitor the consumption of a specific appliance, or it can be a clamp that is used to monitor the current on a specific wire. A device usually communicates with its unit using some form of wired or wireless connection.
- **Tag:** A measurement end point that stores the data readings of a specific device. If a device measures current and power then it has two tags associated, one for current and

another for power. A device may have one or more tags; in other words, it may measure one or more parameters.

All operations, except otherwise noted, must be invoked using HTTPS protocol and provide an HTTP Authorization header like “Authorization: ISA <session token>” where session token is received as a response when a session is created. See Sessions’ methods documentation for more details about authentication.

To ease reading and add meaning, this document follows these simple writing conventions.

Variables are represented as <type: description of the value>. The possible data types are string, number, date, time and Boolean.

Resources can be represented in the following formats:

- Single resource – Resource
- List of resources - [Resource]
- Paged list of resources - [Resource]+

In result to each request performed to the API, a specific response code is returned in the reply to the requester. The following table lists the possible response codes and their meaning:

Response Status Code	Description
200 OK	Successful request. Response body contains the resource representation requested
201 Created	Resource was successfully created.
204 No Content	Resource was successfully updated and its representation didn’t change compared to what was sent by the client. Or, resource was successfully deleted.
304 Not Modified	Your cached resource is still up-to-date.
400 Bad Request	The request contains invalid data, check error details.
401 Unauthorized	Insufficient credentials to access resource.
404 Not Found	Resource not found.
405 Method Not Allowed	The verb used in the request is incorrect.
500 Internal Server Error	General server side error. Please try again later.

2.5.2.1 General conventions

Operations that return lists of resources can return these as a list (represented as [Resource]) or as a paged list (represented as [Resource]+). In either case, the results can be filtered and ordered by fields defined on an operation basis.

Non-paged lists are returned as [] (arrays) of resources whereas paged lists are returned as {} (objects) with the following properties:

```

{
  "page": <number: current page number (zero based)>,
  "page_size": <number: number of maximum records per page>,
  "total": <number: total number of records>,
  "list": [Resource]
```

}

Methods which return paged lists, accept the following URL parameters to customize the list returned:

?page=<page>&page_size=<page size>

All these parameters are optional. Defaults and allowed values are:

- page - default is 0;
- page_size - default is 50, allowed values are between 1 and 200;

Search operations allowing user-defined ordering specify valid order properties as:

Order: <PropertyName1>, <PropertyName2>

Software

For example, an operation documented as:

Order: Id, Type

Allows the caller to specify none, one or both properties as ordering. If none, the bold property is used as the default (ascending) order. If a '-' (minus) is prepended to a property, the ordering is descending. Order is specified as the 'order' query parameter with a list of values. All of the following are valid order parameters:

?order=[Id]

?order=[-Id]

?order=[Type]

?order=[Type,-Id]

Search operations allowing user-defined filters, specify valid filter properties as:

Filter: <PropertyName1>, <PropertyName2>

Filters are defined using a custom language which allows use logical and relational expression to build the desired filter. A simplified language grammar is:

Logical Expression		
Relational Expression		Relational Expression
	&	

Relational Expression		
Property	=	"string"
	<	number
	<=	@timestamp
	>	
	>=	
	!=	
Property	=%	"string"
	%=	

	%	
Property	in	["string",..., "string"] [number,...,number]

Symbol	Operation
&	Logical AND
	Logical OR
=	Equals to
<	Less than
<=	Less or equal than
>	Greater than
>=	Greater or equal than
!=	Different from
=%	Starts with
%=	Ends with
%	Contains
In	Exists in list of values

The expected and returned format for the date data type, except where noticed, is the number of milliseconds between January 1, 1970, 00:00:00 UTC, and the desired date. For instance, 2012 January's first would be sent as 1325376000000.

The time data type is used for specifying a time stamp in a generic day. For this data type, the expected format, except where explicitly stated, is the number of milliseconds since the beginning of one day using the UTC standard. This means that the time stamp 20:00:00 of one day would be sent as 72000000. This also means that this data type only accepts values between 0 and 86400000, including these two values.

For numbers, the decimal separator used is '.' (dot).

2.5.2.2 API Entities

2.5.2.2.1 Actuation

This entity represents an actuation.

```
{
  "ActuationLogId": <number: actuation log identifier>,
  "Command": <string: command (0 for off, 1 for on and 3 for commuting the current state to the other one)>,
  "TagId": <number: tag identifier>
}
```

2.5.2.2.2 ActuatorState

This entity represents the actuation state of a tag

```
{
  "TagId": <number: tag identifier>,
  "State": <string: actuator state (0 for off, 1 for on or 3 for unknown, if the actuator did not reply for some reason)>
}
```

2.5.2.2.3 Consumption

This entity represents a consumption report.

```
{ "UnitId": <number: unit id>,
  "Granularity": <string: type of granularity (instant, daily, hourly, monthly or yearly)>,
  "TagId": <number: tag id>,
  "Date": <date: begin date (using the unit time zone) of the time interval to which this consumption data corresponds>,
  "Read": <number: value of the consumption in the default measurement unit of the respective tag type>,
  "ReadCurrency": <number: value of Read converted to default currency of the supplier>,
  "ReadCarbon": <number: value of Read converted to carbon dioxide values>,
  "Trees": <number: number of trees necessary to compensate the emission of ReadCarbon>,
  "Cars": <number: number of cars which emit ReadCarbon to the atmosphere>,
  "CurrencySymbol": <string: symbol that represents the currency which is used in the value contained in the field ReadCurrency>
}
```

2.5.2.2.4 Device

This entity represents a device.

```
{
  "Id": <number: id>,
  "Name": <string: name>,
  "DeviceTypeId": <number: device type id>,
  "TariffId": <number: tariff id>,
  "Tariff": <Tariff: tariff>,
  "ModuleType": <[ ModuleType]: list of module types that are associated to the device>,
  "FixedCostId": <number: id of the fixed cost associated to a tariff>,
  "UnitId": <number: unit id>,
  "LocalId": <number: local id>,
  "Interval": <number: interval between data acquisitions>,
  "IsCommunicating": <boolean: flag indicating if it is communicating>,
  "Index": <number: index of the device in the unit>,
  "Address": <string: Device ID that comes with the Device>,
  "Deleted": <boolean: flag indicating if the device is deleted or not>,
  "State": <Number: string: the device initial state. By default it is deactivated >,
}
```

2.5.2.2.5 Session

This entity represents a session in the API.

```
{
  "Token": <string: session token>,
  "Timeout": <number: session timeout in seconds>
}
```

2.5.2.2.6 Tag

This entity represents a tag.

```
{ "Id": <number: id>,
  "UnitId": <number: unit id>,
  "Name": <string: name>,
  "Index": <number: The address of the tag in device>,
  "LocationId": <number: location id>,
  "TagTypeId": <number: tag type id>,
  "Created": <date: date when it was created>,
  "Active": <boolean: flag indicating if the tag is active>,
  "Deleted": <boolean: flag indicating if the tag is deleted>,
  "Communicate": <boolean: flag indicating if the tag is communicating>,
  "Manual": <boolean: flag indicating if the tag is manual>,
  "Visible": <boolean: flag indicating if the tag is visible>,
  "Validate": <boolean: flag indicating if the tag should be validated>,
  "InternalAddress": <number: internal address>,
  "VirtualField": <string: data value over which the virtual tag is calculated (value of the tag reading or difference between two values)>,
  "DeviceId": <number: id of the device>
}
```

2.5.2.2.7 Unit

This entity represents an unit

```
{
  "Id": <number: id>,
  "Name": <string: name>,
  "State": <number: unit state id>,
  "MacAddress": <string: mac address>,
  "LocalId": <number: local id>,
  "ModuleType": <[ ModuleType]: list of module types that are associated to the unit>,
  "B64AccessKey": <string: access key for direct communication with unit. Note: currently this field has the same value for all units, but this will be changed in the future>,
  "TimeZoneId": <string: time zone id for this unit>,
  "HashedSerial": <string: public unit identifier>,
  "Interval": <number: communication interval>,
  "LastComm": <date: date of last communication>,
  "DomainId": <number: id of the domain this unit belongs to>,
  "TypeId": <number: unit's type id>,
}
```

```

"IsActive": <boolean: flag indicating whether this unit is valid or not>,
"IsComm": <boolean: flag indicating whether this unit is communicating>,
"Local": <Local: local associated to the unit>,
"Deleted": <boolean: flag indicating if the unit is deleted or not>,
"Dev
"Devices": <[Device]: list of devices that are associated to the unit>,
MonthlyObjective": <number: decimal value that corresponds to the value, in the currency
associated to the user account, of the monthly objective for the unit>,
"LastConfigDate": <date: date of the last configuration change>
}

```

2.5.2.3 API Operations

This section describes the most important API operations

2.5.2.3.1 CreateSession

This method Logs a user in the API.

Request	<p><i>POST /sessions</i></p> <p><i>User</i></p> <p><i>Required fields:</i></p> <p><i>Login</i></p> <p><i>Password</i></p>
Responses	<p><i>201 Created</i></p> <p><i>Session</i></p> <p><i>Possible error codes:</i></p> <p><i>400 Bad Request</i></p> <p><i>null_or_empty.user.password</i></p> <p><i>invalid_length.user.password</i></p> <p><i>null_or_empty.user.login</i></p> <p><i>invalid_length.user.login</i></p> <p><i>invalid.social_provider</i></p> <p><i>null_or_empty.facebook.access_token</i></p> <p><i>null_or_empty.facebook.email</i></p> <p><i>null_or_empty.user.email</i></p> <p><i>401 Unauthorized</i></p> <p><i>no_permissions_to_access</i></p>

2.5.2.3.2 FindUnits

This method returns one list with the details of one set of units, optionally filtered and ordered by lists of its ids and local ids, and optionally showing the information of the local of each unit returned.

Request	<p><i>GET /units</i></p> <p><i>Include: Local,Device,FixedCost,Tariff,Interval</i></p> <p><i>Where: Id, LocalId, HashedSerial, IsComm</i></p> <p><i>Order: Id, LocalId, HashedSerial, IsComm</i></p>
---------	--

	<p><i>Required fields:</i></p> <p>None</p> <p><i>Optional fields:</i></p> <p>Include</p> <p>Where</p> <p>Order</p>
Responses	<p>200 OK</p> <p>[Unit]+</p> <p><i>Possible error codes:</i></p> <p>400 Bad Request</p> <p><i>invalid.include_parameter</i></p> <p><i>invalid.where_parameter</i></p> <p><i>invalid.order_parameter</i></p> <p>404 Not Found</p> <p><i>not_found.unit</i></p> <p><i>not_found.local (not implemented yet)</i></p>

2.5.2.3.3 FindTags

This method returns one list with the details of a set of tags, optionally filtered and ordered by lists of its ids, unit ids, tag type ids and device ids.

Request	<p>GET /tags</p> <p>Where: <i>Id, UnitId, TagTypeId, DeviceId</i></p> <p>Order: Id, UnitId, TagTypeId, DeviceId</p> <p><i>Required fields:</i></p> <p>None</p> <p><i>Optional fields:</i></p> <p>Where</p> <p>Order</p>
Responses	<p>200 OK</p> <p>[Tag]+</p> <p><i>Possible error codes:</i></p> <p>400 Bad Request</p> <p><i>invalid.where_parameter</i></p> <p><i>invalid.order_parameter</i></p> <p>401 Unauthorized</p> <p><i>no_permissions_to_access.tag</i></p> <p><i>no_permissions_to_access.unit</i></p> <p><i>no_permissions_to_access.device</i></p>

	<p>404 Not Found</p> <p><i>not_found.unit</i></p> <p><i>not_found.tag_type</i></p> <p><i>not_found.device</i></p> <p><i>not_found.tag</i></p>
--	---

2.5.2.3.4 GetConsumptions

This method returns a set of consumptions filtered by granularity, begin date, end date and a list of tag identifiers. It is important to note that some of the tag types are instantaneous (for instance, temperature and humidity). In those cases the request to this method should include the optional field *InstantsType*, which will determine how the value of *Read* (one of the fields of the objects *Consumption* that are returned in this method) will be calculated. If the optional field *InstantsType* is passed with a null value, the method will treat the request as it was passed the value *Avg* in that field, for each tag that is noncumulative. If the tag is cumulative, the value passed in the field *InstantsType* is ignored.

Request	<p><i>GET /consumptions/<string: the type of aggregation in which we want to obtain the consumptions >?from=<string: begin date (in the unit time zone) of the time interval to which the calculation will be performed>&to=<string: end date (in the unit time zone) of the time interval to which the calculation will be performed>&tags=<string: list of comma separated tag identifiers, delimited by characters [and]>&instantsType=<string: the instants type></i></p> <p><i>Granularity: Instant, Hourly, Daily, Monthly, Yearly</i></p> <p><i>InstantsType: Avg, Max, Min, Stdev</i></p> <p><i>Avg – Average</i></p> <p><i>Max – Maximum</i></p> <p><i>Min – Minimum</i></p> <p><i>Stdev – Standard deviation</i></p> <p><i>Required fields:</i></p> <p><i>Granularity</i></p> <p><i>From</i></p> <p><i>To</i></p> <p><i>Tags</i></p> <p><i>Optional fields:</i></p> <p><i>InstantsType</i></p>
Responses	<p>200 OK</p> <p><i>[Consumption]</i></p> <p><i>Possible error codes:</i></p> <p>400 Bad Request</p> <p><i>null_or_empty.granularity</i></p> <p><i>null_or_empty.from_date</i></p> <p><i>null_or_empty.to_date</i></p> <p><i>null_or_empty.tags</i></p> <p><i>invalid.consumption_granularity – invalid_value.granularity</i></p>

	<i>invalid.generic.dates - invalid_value.from_date ; invalid_value.to_date ; invalid_date_relation.from_date_vs_to_date</i> <i>invalid.instant_type - invalid_value.instant_type</i> 401 Unauthorized <i>no_permissions_to_access.tag (not implemented yet)</i> 404 Not Found <i>not_found.tag (not implemented yet)</i>
--	---

2.5.2.3.5 Actuate

This method creates a new actuation command and sends it, ‘on the fly’, to the respective unit.

Request	<i>POST /actuations</i> <i>Actuation</i> <i>Required fields:</i> <i>TagId</i> <i>Command</i> <i>Optional fields:</i> <i>None</i>
Responses	201 Created <i>Actuation</i> <i>Possible error codes:</i> 400 Bad Request <i>non_actuation.tag</i> <i>invalid.command invalid_value.command</i> 401 Unauthorized <i>no_permissions_to_use.tag- no_permissions_to_access.tag</i> 404 Not Found <i>not_found.tag</i>

2.5.2.3.6 GetAtuationStateByTagId

This method returns the state of an actuator given an actuation tag identifier. This operation is made ‘on the fly’, which implies communicating with the respective unit.

Request	<i>GET /actuatorstate/<string: actuation tag identifier></i> <i>Required fields:</i> <i>TagId</i> <i>Optional fields:</i> <i>None</i>
---------	---

Responses	<p>200 OK</p> <p><i>ActuatorState</i></p> <p>Possible error codes:</p> <p>400 Bad Request</p> <p><i>null_or_empty.tag_id</i></p> <p><i>non_actuation.tag</i></p> <p>401 Unauthorized</p> <p><i>no_permissions_to_access.tag</i></p> <p>404 Not Found</p> <p><i>not_found.tag</i></p>
------------------	--

2.5.2.4 API Usage Scenario

In order to use the API, first an authentication has to be preformed using the `CreateSession` method. This method needs a valid username and password and, if the authentication is valid, returns a token that can be used to identify the user in the following requests. An example of one possible pair request/response to this method is shown below.

Request	<pre>POST http://innov.isa.pt:6600/api/1.0/sessions HTTP/1.1 Accept-Encoding: gzip,deflate Content-Type: application/json User-Agent: Jakarta Commons-HttpClient/3.1 Host: innov.isa.pt:6600 Content-Length: 48 Body: { "Login": "my_username", "Password": "my_password" }</pre>
Response	<pre>HTTP/1.1 201 Created Server: ASP.NET Development Server/10.0.0.0 Date: Fri, 29 Jul 2011 14:17:15 GMT X-AspNet-Version: 4.0.30319 Content-Length: 117 Cache-Control: private Content-Type: application/json; charset=utf-8 Connection: Close { "Timeout": 900, "Token": "qv6/8VpPAK9+Hon6RQVSIMZCRMZMKAZ+T9y3/0qXkJ7N+PyAsda0mdAv6YZV6zyC4+ReDlwGqQiuGMn ANqqHsw==" }</pre>

The authentication token is valid for the “timeout” value, which is in this case 900 seconds. After the timeout has expired, a new token must be generated using the same method.

After the authentication has been performed, a method to discover the user's units may be called: the FindUnits Method. An example request and response follows.

Request	<pre>GET http://innov.isa.pt:6600/api/1.0/units?Where=Id%3D94%7CId%3D2&Order=-LocalId%2CId&Include=Local HTTP/1.1 Accept-Encoding: gzip,deflate Accept: application/json Authorization: ISA qv6/8VpPAK9+Hon6RQVSI MZCRMZMKAZ+19y3/0qXkJ7N+PyAsda0mdAv6YZV6zyC4+ReDwwGqQiuGMnA NqqHsw== User-Agent: Jakarta Commons-HttpClient/3.1 Host: innov.isa.pt:6600 Body:</pre>
Response	<pre>HTTP/1.1 200 OK Server: ASP.NET Development Server/10.0.0.0 Date: Fri, 29 Jul 2011 15:18:09 GMT X-AspNet-Version: 4.0.30319 Content-Length: 508 Cache-Control: private Content-Type: application/json; charset=utf-8 Connection: Close { "List": [{ "B64AccessKey": "000111222333444555", "Id": 94, "Local": { "CountryCode": "PT", "Id": 85, "IsStorage": false, "Name": "B0605-Trofa", "UserId": 1, "ZipCodeId": 190856 }, "LocalId": 85, "MacAddress": "00-04-A3-22-81-D8", "Name": "B0605-Trofa", "State": 1 }, { "B64AccessKey": "000111222333444555", "Id": 2, "Local": { "CountryCode": "PT", "Id": 4, "IsStorage": false, "Name": "iMeter ESI Testes", "UserId": 1, </pre>

```

    "ZipCodeId": 71756
  },
  "LocalId": 4,
  "MacAddress": "00-04-A3-22-88-C7",
  "Name": "iMeter Teste ESI",
  "State": 1
}
],
"Page": 0,
"PageSize": 50,
"Total":2
}

```

This response shows the list of units in the database that match the request criteria and their respective information. The list is composed by 2 elements, the ones whose id is 94 or 2, using the fields “LocalId” (in decreasing order) and “Id” (in crescent order) for ordering. The page size used to show the results list is 50 and only the first page (page 0) of that list is shown (which is the only one, in this particular case). The information of the each unit’s local is also displayed.

The tags belonging to a unit can also be retrieved. For this task, the FindTags request should be issued. The following table illustrates an example of the retrieval of the tags from a Unit.

Request	<pre> GET http://innov.isa.pt:6600/api/1.0/tags? where=TagTypeId+in+%5B4%2C5%5D+%26+UnitId+%3D+2 HTTP/1.1 Accept-Encoding: gzip, deflate Authorization: ISA qv6/8VpPAK9+Hon6RQVSIMZCRMZMKAZ+79y3/0qXkJ7N+PyAsda0mdAv6YZV6zyC4+ReDwwGqQiuGMnA NqqHsw== User-Agent: Jakarta Commons-HttpClient/3.1 Host: innov.isa.pt:6600 Body: </pre>
Response	<pre> HTTP/1.1 200 OK Server: ASP.NET Development Server/10.0.0.0 Date: Fri, 29 Jul 2011 15:18:09 GMT X-AspNet-Version: 4.0.30319 Content-Length: 508 Cache-Control: private Content-Type: application/json; charset=utf-8 Connection: Close { "List": [{ "Active": true, "Communicate": false, "Created": 1311870060000, "Deleted": false, "DeviceId": 895, "DomainId": 2, "Id": 2684, </pre>

```

    "InternalAddress": 0,
    "LocationId": 151,
    "Manual": false,
    "Name": "Energia Activa",
    "RemoteId": null,
    "TagTypeId": 4,
    "UnitId": 2,
    "Validate": false,
    "VirtualField": null,
    "Visible": true
  },
  {
    "Active": true,
    "Communicate": false,
    "Created": 1311870060000,
    "Deleted": false,
    "DeviceId": 896,
    "DomainId": 2,
    "Id": 2687,
    "InternalAddress": 0,
    "LocationId": 151,
    "Manual": false,
    "Name": "Energia Activa",
    "RemoteId": null,
    "TagTypeId": 4,
    "UnitId": 2,
    "Validate": false,
    "VirtualField": null,
    "Visible": true
  }],
  "Page": 0,
  "PageSize": 50,
  "Total": 2
}

```

This response shows the list of tags in the database that match the request criteria and their respective information. The list is composed by 2 elements, the ones whose tag type id is 4 or 5 and unit id is 2, using the default ordering (by the field “Id”, in crescent order). The page size used to show the results list is 50 and only the first page (page 0) of that list is shown (which is the only one, in this particular case).

The values from each tag may be also retrieved from the API. For this task the GetConsumptions method should be used. The following table illustrates a GetConsumptions request.

Request	Full path: <i>http://innov.isa.pt:6600/api/1.0/consumptions/daily?to=1406105200000&from=1304982000000&tags=%5B2684%5D</i>
	HTTP method: GET

Authorization HTTP header: "ISA
qv6/8VpPAK9+Hon6RQVSIMZCRMZMKAZ+T9y3/0qXkJ7N+PyAsda0mdAv6YZV6zyC4+ReDwwGqQiuGMnA
NqqHsw=="

Media type: Application/json

Body:

Response **Media type:** Application/json

Body:

```
[  
{  
  "Cars": 329.622,  
  "Date": 1311897600000,  
  "Granularity": "daily",  
  "Read": 109.874,  
  "ReadCarbon": 38.4559,  
  "ReadCurrency": 13.3052,  
  "TagId": 2684,  
  "Trees": 219.748,  
  "UnitId": 150  
}
```

3. GETTING STARTED WITH THE HOW MODULE

3.1 Introduction and system requirement

As mentioned in the previous sections, the How Module presents a user interface, which is the front-end for the user to access the module's functionalities. This user interface is a stand-alone component that might run entirely outside the Data Server of the platform, presenting a web-based interface to the user. In terms of high-level requirements, the How Module's user interface needs to be available on a web server and to be accessed by a recent web browser.

As all of the module's user interface logic is implemented inside the Data Server and the user's web-browser, the requirements in the server side for the user interface are very simple. In order to make available the How module's user interface a simple web server, such as Microsoft IIS or Apache Web Server, capable of serving simple web pages is sufficient. The only requirement is that the web server should be capable to classify the json extension as application/json content mime type in the response headers. As example, in IIS requirement can be fulfilled in the IIS Manager by accessing the site's MIME Types (Figure 6) and adding a new MIME Type for .json as "application/json". This feature is required for the user interface to work properly, so failing to do this step will prevent the module's user interface to execute.

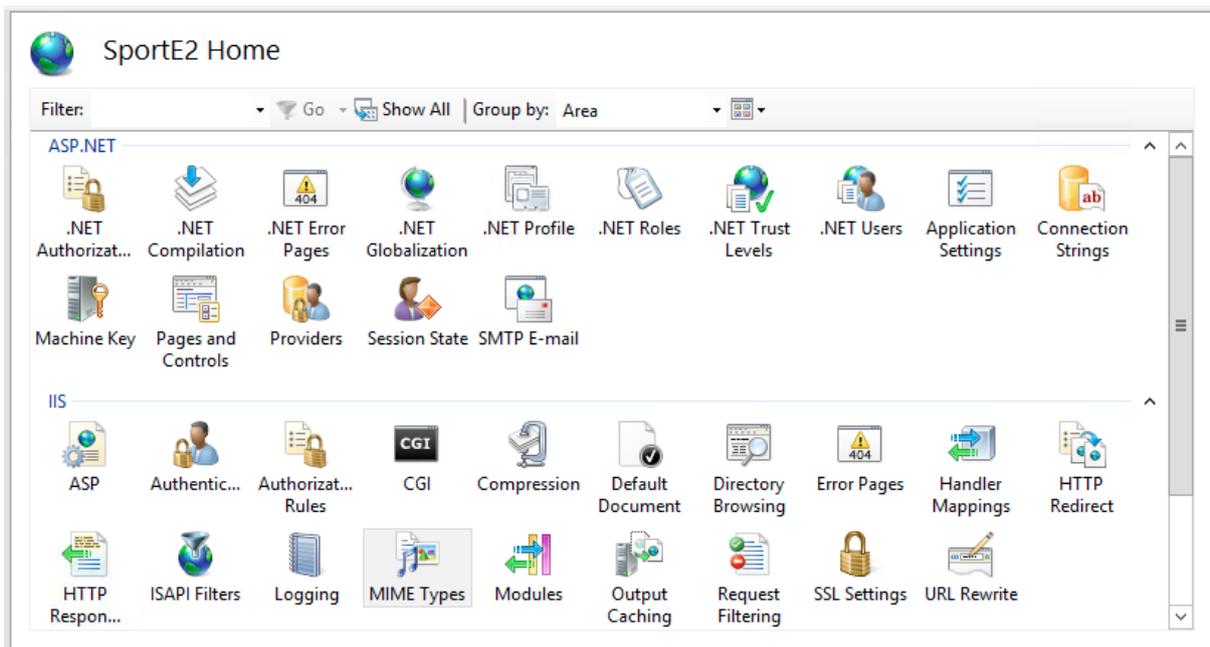


Figure 6 - IIS Manager - Mime Types Selected

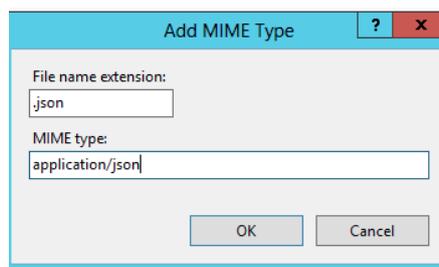


Figure 7 - IIS Manager - Adding the .json MIME Type

As mentioned before, the How Module's user interface is web-based and may be accessed from anywhere that fulfils the client-side requirements. The client-side requirements are presented in the following list:

- Internet connection to the web server hosting the user interface - This connection is necessary for accessing the user interface web site. In this project the user interface is hosted in <http://innov.isa.pt:11115/>, so a connection to the TCP port 11115 of the host innov.isa.pt must be accessible and allowed in any firewall that might exist in the internet path between the web server and the web browser.
- Internet connection to the Data Server API - This connection is necessary for the user to access the data from the platform. In this project the Data Server API is available in <http://innov.isa.pt:6600/>, so a connection to the TCP port 6600 of the host innov.isa.pt must be accessible and allowed in any firewall that might exist in the internet path between the web server and the web browser.
- Recent Graphical Web Browser, capable of interpreting HTML5 based pages - As the How Module's uses the latest HTML5 web programming technologies to implement the most richest user experience, a recent web browser is required, such as Google Chrome version 34.0.1847.116 (or above) or Mozilla Firefox version 28.0 (or above) should be used for accessing the user interface.

In order to access the module's user interface it is necessary a graphical web browser as described in the previous sub-section and point it to the following url:

<http://innov.isa.pt:11115/>

To use the interface, a user with full access to all the three pilot facilities of the SportE2 project was created. The credentials of this user are:

- Username: kisense
- Password: kisense

3.2 User Manual

This section describes the functionalities of the How Module's user interface and the procedure to be followed to access and use them.

The user interface contains the following views/pages:

- Login page – Used to identify the user and prevent unauthorized access to the user interface;
- Presentation view – Used to display general information about measured variables values or the pilot facility information in the format of a automatic slideshow;
- Data Explorer view – Used to permit the visualization of the collected and calculated values or indicators from the facility;
- Alarms view – Used to access the alarms detected by the platform.

With the login page as exception, all the views have a common base layout composed of a top bar and a contents view below the top bar, as shown on Figure 8.



Figure 8 - Common User Interface Layout

The Top Bar contains the SportE2 logotype and some controls that can be used to select the active view, control the contents of a specific view or control the current user of the interface (i.e. perform logut). Figure 9 shows an example of the Top bar.



Figure 9 - Top Bar of the common view

The first zone of the top bar is the project logotype. This zone has no actions associated with it and is informative only. Figure 10 highlights this zone inside the Top Bar.

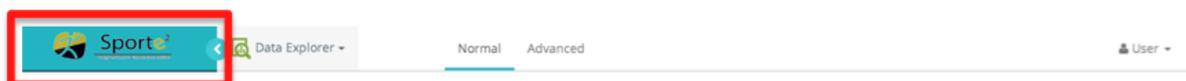


Figure 10 - Top Bar Logotype

The second zone, highlighted in Figure 11, shows the currently activated view. This zone has also the functionality of enabling the user to select the active view displayed on the user interface.



Figure 11 - Top Bar Active View Control

When the user hovers the mouse in this zone, a dropdown selector control is displayed on the interface. The dropdown selector control displays the view modes available, as shown in Figure 12. By clicking on any of the available options, the user may switch the active view mode.

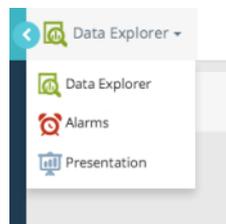


Figure 12 - View mode dropdown selector

Depending on the selected active view, the Contents Control zone of the Top Bar, highlighted in Figure 13, may display view-dependent information or controls that affect the way the specific view displays content to the users. Further information on this topic may be found in the specific view sub-section.

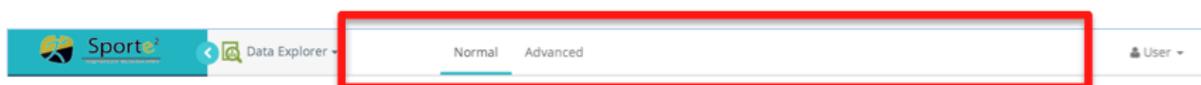


Figure 13 - Top Bar Contents Control

In the rightmost zone of the Top bar there is the session control zone. This zone, highlighted in Figure 14, displays the currently logged in username.



Figure 14 - Top Bar Session Control

By clicking with the mouse in this session control zone, a dropdown selector control is displayed, as shown on Figure 15, containing the option to perform the log out of the currently logged user. For security reasons it is recommended that the users perform the log-out action after using the module's user interface but, if the user fails to perform the log-out

action, the platform automatically performs the log-out of the user if doesn't detect any user action after 5 minutes.

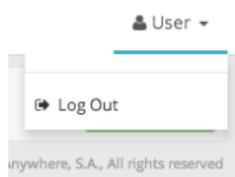


Figure 15 - Session control dropdown selector

3.2.1 Login Page

The entry page of the How Module's user interface is the Login Page. As it happens commonly on other web sites, this page is used to prevent unauthorized access to the user interface and, even if the user is allowed to admitted into the user interface, to allow access only to the data that the user is entitled to access.

The Login Page presents the interface shown in Figure 16. This interface has two text input fields: the top one is the username input, where the user inputs his username; the bottom one is the password field.

This page also allows the memorization of the session by using the "Remember me" checkbox, for avoiding the Login page and introduction of the access credentials in subsequent accesses to the How Module's user interface.

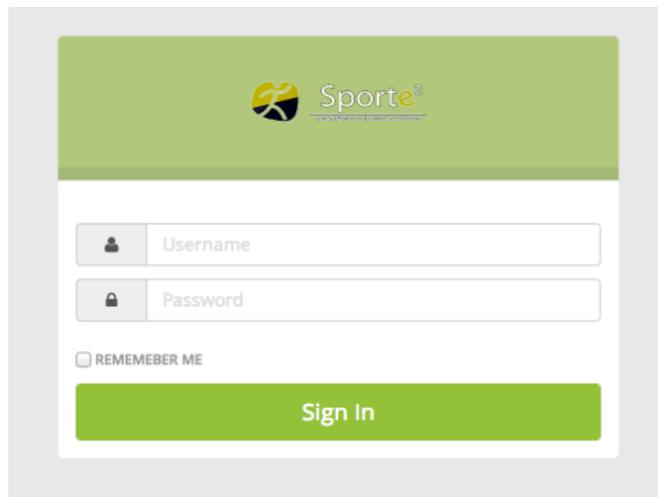


Figure 16 - How Module User Interface Login Page

The process for accessing the user interface is simple:

1. User inserts his username in the top input box
2. User inserts his password in the bottom input box
3. Optionally user chooses if he wants the user interface to "remember" the session in the "Remember Me" checkbox
4. User clicks the "Sign In" button.

3.2.2 Presentation View

The presentation view is a view that can be used to present a facility to the general public as a dissemination tool for the project. This view presents a general summary of the facility status, energy consumption or other relevant information that might be of interest to the user or public.

The user interface permits the existence of a number of independent slides showing distinct information in each one. By configuration, the slides may be automatically rotating after being displayed for some (configurable) time. Even if the presentation screen is configured for automatic rotation of the slides, the user may override the currently displayed slide by clicking on the correspondent slide “dot” on the bottom section of the screen.

Each one of the slide’s compositions may also be configured for displaying graphs containing the historic of a specific variable and cards containing statistics or current values from specific variables as well. There is also the possibility to create informational cards with miscellaneous information such as maps displaying the pilot (or pilots) locations or informative texts about the SportE2 project.

Figure 17 depicts an example of a presentation screen, containing a variable graph (containing the Electricity consumption) of an abstract facility in the top and three distinct informative cards in the bottom. The informative cards can contain a map with pinpoints (e.g. displaying the location of the facility or location of the project facilities), summary of a variable such as the total electric consumption of the current month or display the last value collected for a variable of interest such as the air temperature of a specific facility space or water parameters collected from the swimming pool.

The presentation screen can be made available in a visible space inside the facility (i.e. near the facility reception). For this scenario, a full screen button is available to make the presentation screen occupy the full display screen. In the full screen mode, the web browser menus are hidden and the presentation screen occupies the full display.

Depending on the pilot and the pilot’s stakeholders, several distinct informative screen can be configured on the platform and made available. The following list show a list of possible screens considered interesting to consider in the presentation view:

- Air temperature and humidity of the different facility spaces such as swimming pool, gyms, indoor and outdoor courts;
- Comfort indexes calculated for the different facility spaces;
- Water parameters collected from the swimming pool(s) such as water temperature, pH or CL concentration level;
- Energy consumption of the global facility of sub-zone of the facility by energy type;
- SportE2 project information;
- Information about the facility characteristics.

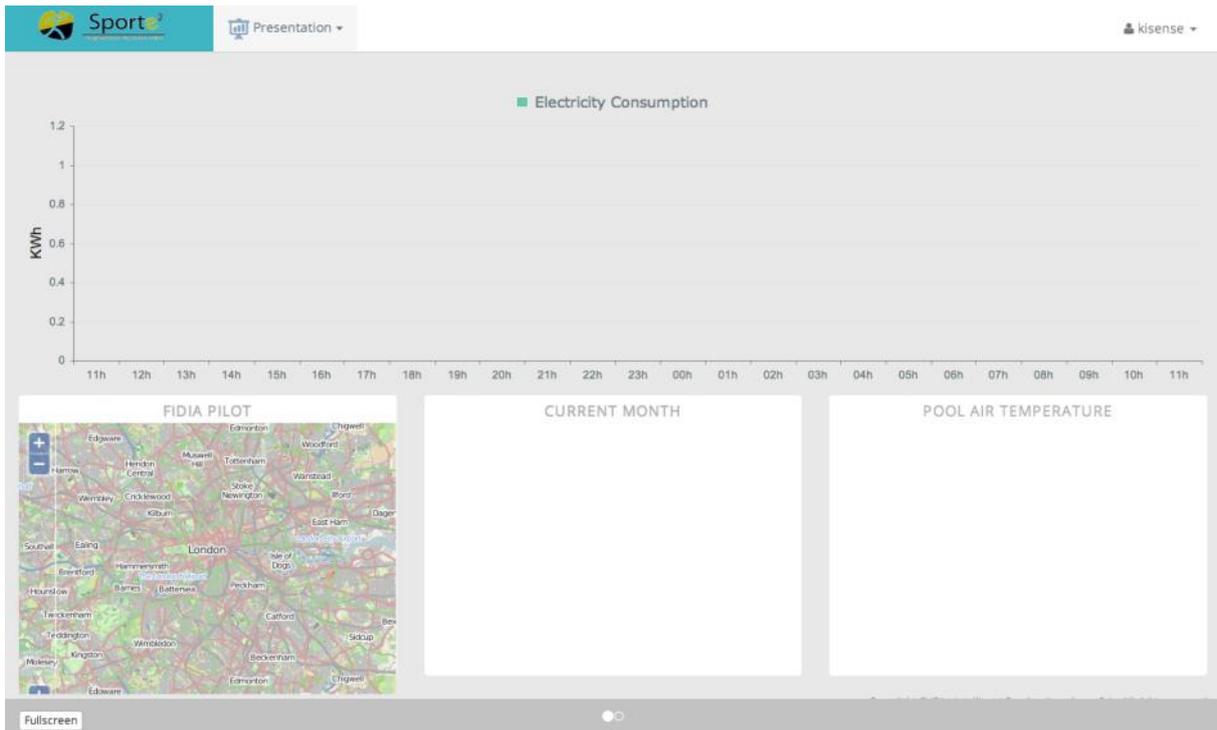


Figure 17 - Presentation screen example

3.2.3 Data Explorer

The Data Explorer view can be considered the main view of the How module’s user interface. This view can be used to explore all the data collected from the facility or facilities. Figure 18 shows the global view of the Data Explorer view.

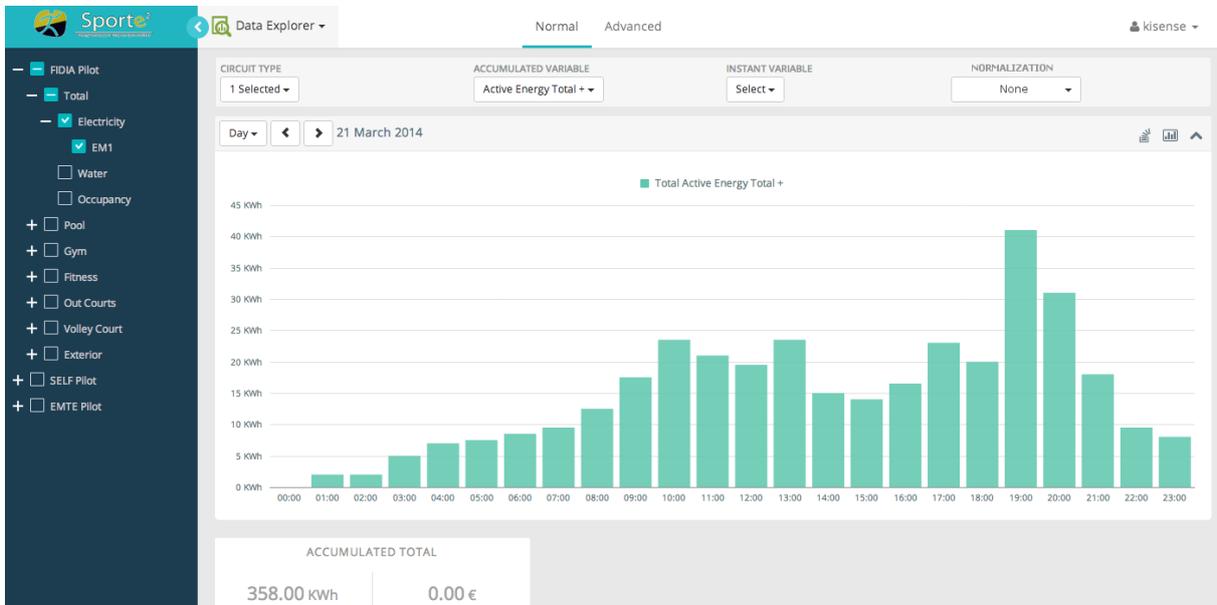


Figure 18 - Data Explorer View

3.2.3.1 Measurement device selection

On the left side of the Data Explorer view the user can find the selection of the facility or facilities if the user is allowed in more than one facility, facility zones, systems, measurement types and measurement devices, organized in an collapsible hierarchical intuitive format.

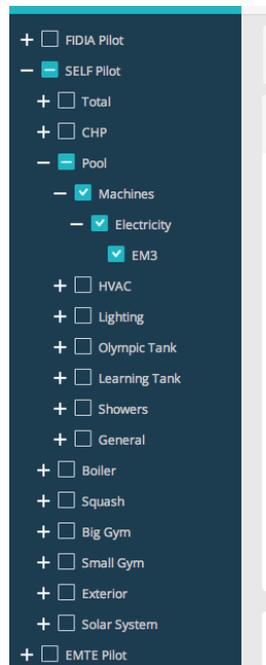


Figure 19 - Data Explorer selection of displayed measurement nodes

Figure 19 shows an example for selecting a variable to be explored in the Data Explorer View. The example shows a selection of an electrical meter and, following the selection list hierarchy, the user is aware that the selected variable is an electrical meter (EM3) measuring the electricity consumed by the Pool's machines of the SELF Pilot. This way allows the user to go to the detail of the meter/sensor installed in the field, always having knowledge of the meaning of the displayed values in the screen.

The user interface also permits the user to select the displayed variables at a higher level (i.e. not at the sensor level), by selecting the variables at a higher hierarchical level. By selecting variables at a higher hierarchical level the user interface automatically selects the variables found at the correspondent lower levels. In the example of Figure 19, the user could make the same selection at the level of the Machines item without expanding the lower levels, that the lower levels (Electricity -> EM3) would be selected automatically.

The interface also allows the selection of multiple variable types in the selection as shown on Figure 20 where the Electricity and Thermal variables provided by EM4 and HM3 are selected. This selection makes both the variables available for exploring. In the case of the selected variables having similar variable types (as in this case where both variables have Energy), the values can be displayed combined (summed) in the graph. If the variables are of incompatible types, the user has the possibility of showing the variables in an independent way in the graph.

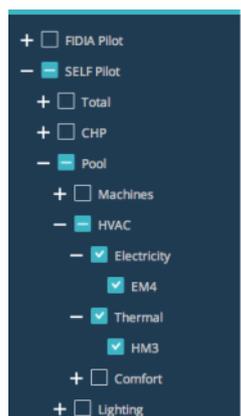


Figure 20 - Data Explorer selection of multiple variable types

After selecting the desired variables to explore, the user may optionally hide the variable selection box to maximize the graph displaying area. To perform this action, the user should click on the correspondent button on the interface's Top Bar as shown on Figure 21.



Figure 21 - Data Explorer button to show/hide the variable selection

3.2.3.2 Displayed variables selection

The next step to explore the data from the selected devices is to act on the graph options, found in the top of the graph, as shown on Figure 22. The options found on this zone are the grouping, accumulated variable and instant variable.

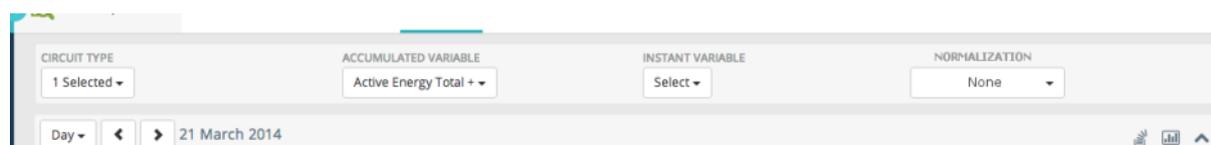


Figure 22 - Data Explorer Graph options selection

The circuit type selector enables the possibility to select the desired circuit groups. The circuit groups can be interpreted as measurement device categories, such as Electricity, Thermal, Air Velocity, among others available in the selected measurement devices. Multiple groups can be selected at once.

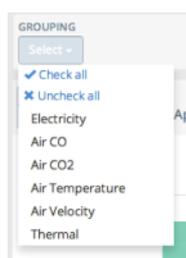


Figure 23 - Data Explorer Grouping selector

After selecting the desired groups, the user should select the variables to show in the exploring graph. The variables are divided in two types:

- Accumulated variables – Values obtained from totalized variables such as energy counters. An example is shown on Figure 24;



Figure 24 - Data Explorer accumulated variable selection

- Instant variables – Values obtained from instant measurements such as temperatures, air speed or pH level. An example is shown on Figure 25.

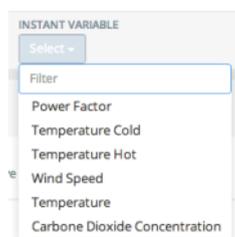


Figure 25 - Data Explorer instant variable selection

The user interface enables the possibility of multiple variables being selected at once. If multiple measurement devices are selected, the similar variables of distinct measurement devices are combined by the user interface when displaying the values to the users.

Another functionality of the user interface is the possibility to perform calculations on the displayed data such as normalizations. The normalizations performed may be the application of a specific constant factor to a variable, such as the area or volume of a facility or facility space (i.e. for the normalization of a energy by area), or the relation between two different series such as a energy variable and a occupancy variable (i.e. for normalization of a energy by the occupancy) or energy variable (i.e. the energy consumed by the HVAC system) and the HVAC effort in degree-days. For selecting the desired normalization to apply to the data, the normalization selector should be used, as shown on Figure 26.

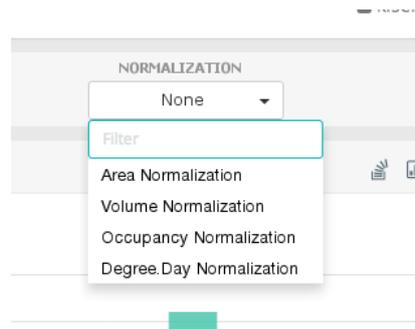


Figure 26 – Variable normalization selector

3.2.3.3 Graphing variables

In order to see the values from the device’s variables, the user needs perform the selection of the devices on the left side bar and select the desired variable on the dropdown boxes (Figure 24 and/or Figure 25). After performing the selection, the user interface will fetch the selected data from the server and display the values to the user in the form of a graph as shown on Figure 27.

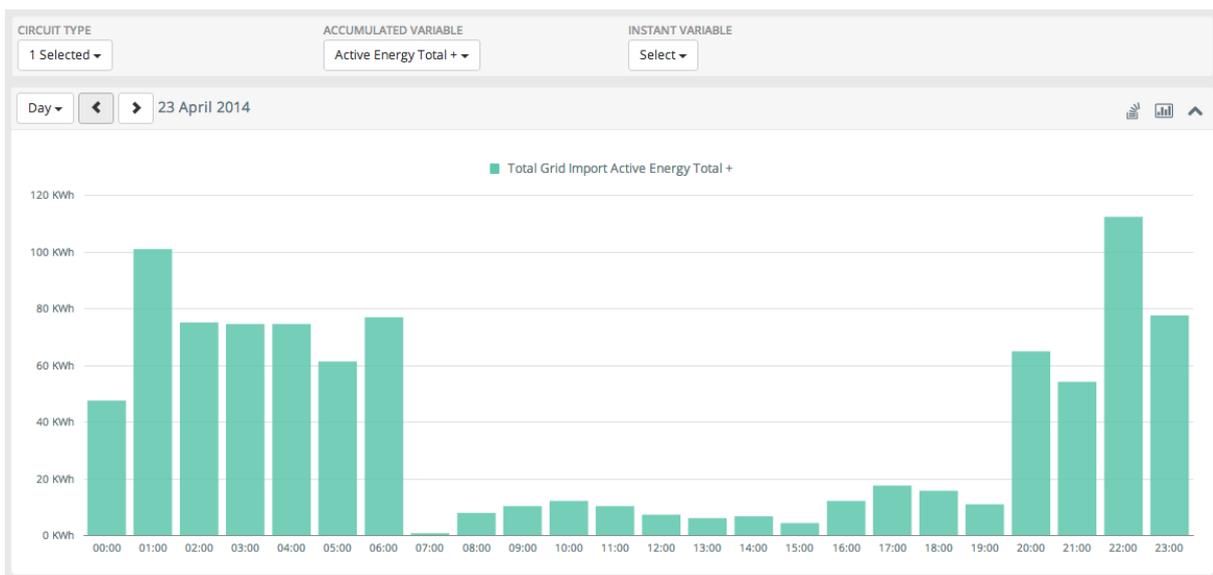


Figure 27 - Data Explorer example of a produced graph

Inside the top of the graph zone, the user is presented with some additional options. The first option is regarding the displayed aggregation of the data, as shown in Figure 28.

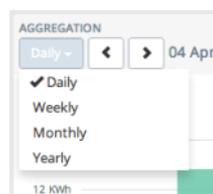


Figure 28 - Data Explorer displayed aggregation of data

The aggregation type changes the displayed data combined in each bar or graph point:

- Daily aggregation – Displays a full day, each bar/point represents a hour of the day;
- Weekly aggregation – Displays a full week, each bar/point represents a full day;
- Monthly aggregation – Displays a full month, each bar/point represents a full day;
- Yearly aggregation – Displays a full year, each bar/point represents a full month;

The controls found at the right of the aggregation selection, shown on Figure 29, enable the user to move the displayed time period to the period before or after the displayed one, depending on the aggregation type.



Figure 29 - Data Explorer displayed period change buttons

On the right side of the top of the graph zone, the user can find three buttons as shown in Figure 30, having additional functionalities.



Figure 30 - Data Explorer graph buttons

The first button enables the possibility to stack the several displayed variables on a single stacked bar, instead of an individual bar per displayed variable as displayed per default. Figure 31 displays a graph of the different electric consumption of SELF swimming pool systems as individual bars, enabling the possibility to compare the different systems among them; Figure 32 shows the same variables with the stacking functionality activated, enabling the user to have a idea of the global consumption and the weight of each one of the systems in that consumption.

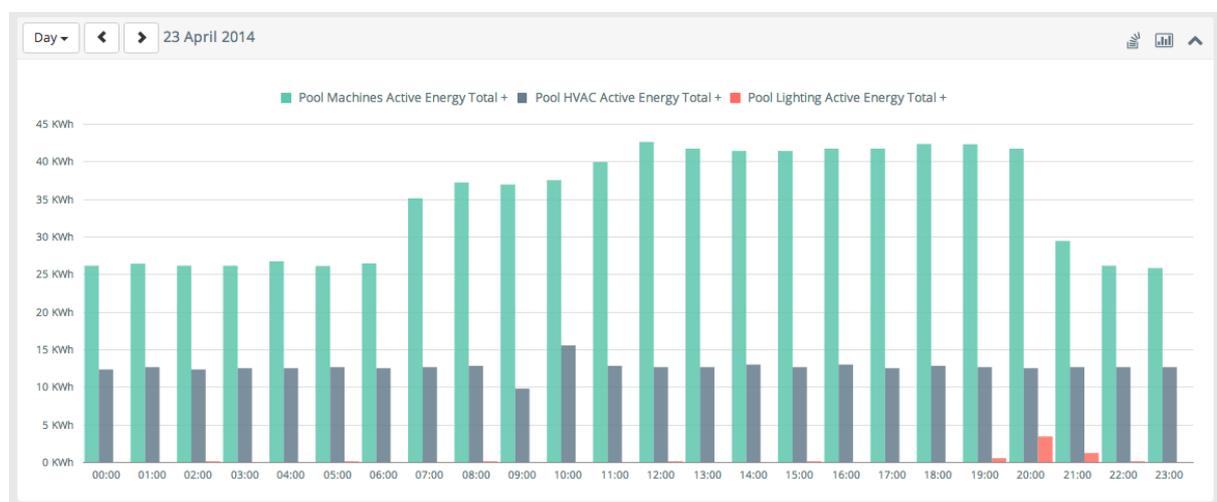


Figure 31 - Multiple variables displayed on the graph

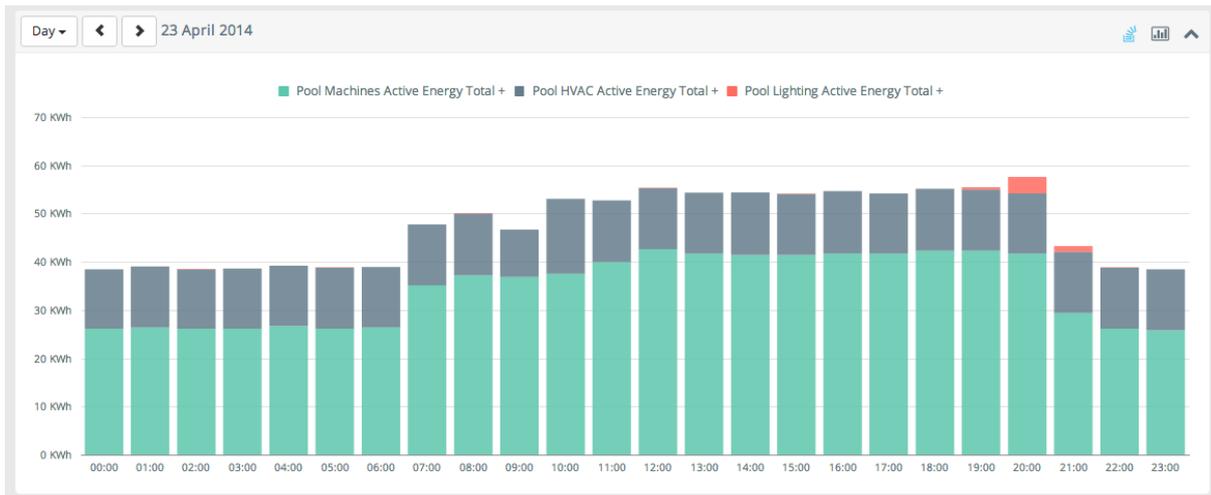


Figure 32 - Multiple variables displayed on the graph (stacked)

The second button enables the possibility to profile the displayed variable period with the homologous period. This functionality can be used to compare the consumption profiles of distinct periods and be used to spot abnormal consumptions of a specific system of facility. This functionality is depicted on Figure 33. Depending on the selected aggregation, the graph can be used to display different comparisons:

- Day aggregation – Comparison with the previous day, each bar representing a full hour of the day;
- Week aggregation – Comparison with previous week, each bar representing a full week day;
- Month aggregation – Comparison with previous month, each bar representing a full day;
- Year aggregation – Comparison with previous year, each bar representing a full month.

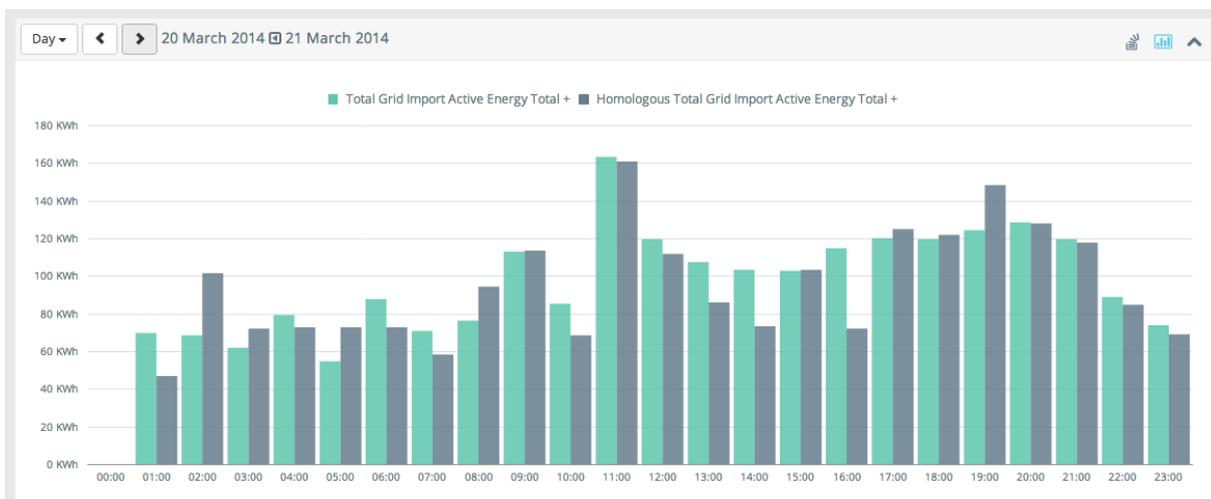


Figure 33 – Comparison/profiling of consumption profiles with the homologous period

The How module user interface also permits the displaying of comfort factor displaying, as well as the resultant hot and cold discomfort hours for the spaces where a temperature and humidity sensor is installed. The access to the comfort functionality is made seamlessly with the same process used to display data from any of the installed field sensors. The comfort calculations are present in the hierarchical tree, under the Comfort leaf, as shown on Figure 34.

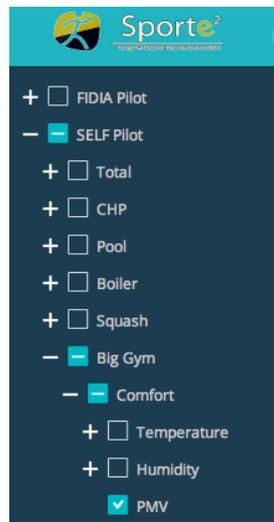


Figure 34 - Selection of the PMV calculation functionality

After selecting the PMV “virtual” device, the PMV circuit is made available on the Circuit type selector (Figure 35) and the respective variables under the Instant Variable selector (Figure 36). After selecting the respective variable, the graph is displayed to the user



Figure 35 - PMV Circuit type selection

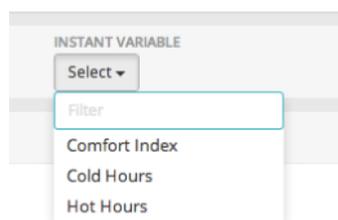


Figure 36 - PMV Variables selection

3.2.3.4 Graph summary

At the bottom of the presented graph, the user may find a summary card displaying the summary of information found on the displayed graph. For accumulated variables (i.e. energy variables), a summary card is displayed containing the total sum value displayed in the graph,

the cost (if available) and the comparison against the defined baseline. The monetary cost of the displayed graph is also computed and displayed to the user as well in this summary card.

This summary card also displays the equivalent in CO₂ Kg, trees and car travel Km of the displayed variable in the graph. An example of a summary card for accumulated variables is shown on the Figure 37.

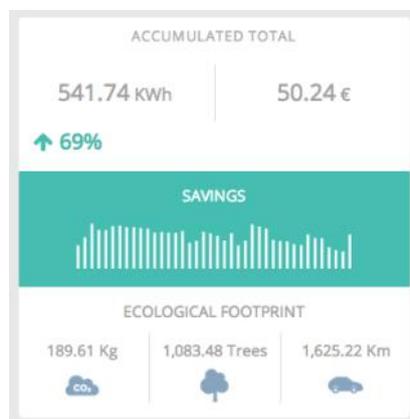


Figure 37 - Data Explorer accumulated variable summary

For instant variables, such as a temperature, air velocity or water pH, a instant summary card is displayed as shown on Figure 38. This card displays the average value of the displayed graph in the form of a gauge (with the indication of the value) as well as the minimum, maximum and standard deviation calculated values.

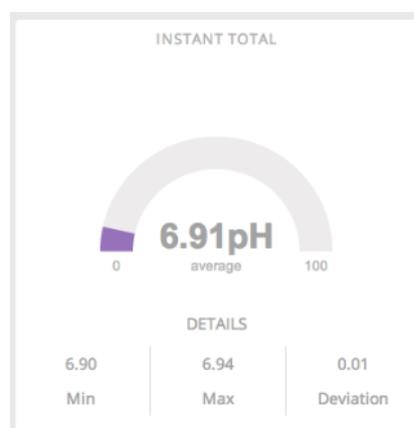


Figure 38 - Data Explorer instant variable summary

3.2.3.5 Advanced data explorer

For advanced data exploring, an advanced version of the data explorer is available for the user. The advanced data explorer exists in the user interface to enable advanced functionalities such as:

- Possibility to observe a custom defined time period – The normal explorer enables only the possibility to explore a discrete set of time spans depending on the selected aggregation (a full day, a full week, a full month or a full year) while the advanced Data Explorer permits the selection of a custom start and end date and time, with any aggregation applied to the data (hour, month, day or year).

- Possibility to view the data of the displayed graph in a table view.
- Possibility to download the displayed data from the user interface in a CSV formatted file.

In order to access the advanced data explorer, the user must switch the mode on the Top Bar as shown in Figure 39.

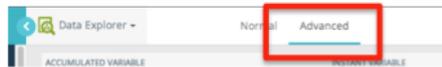


Figure 39 - Advanced Data Explorer selection

The layout of the Advanced Data Explorer is similar to the Normal Data Explorer with some differences as described in this section contents. The Advanced Data Explorer layout is shown in Figure 40.

The measurement device selection is similar to the one used in the Normal Data Explorer, in the hierarchical list view on the left side bar of the user interface, as described in the sub-section 3.2.3.1.

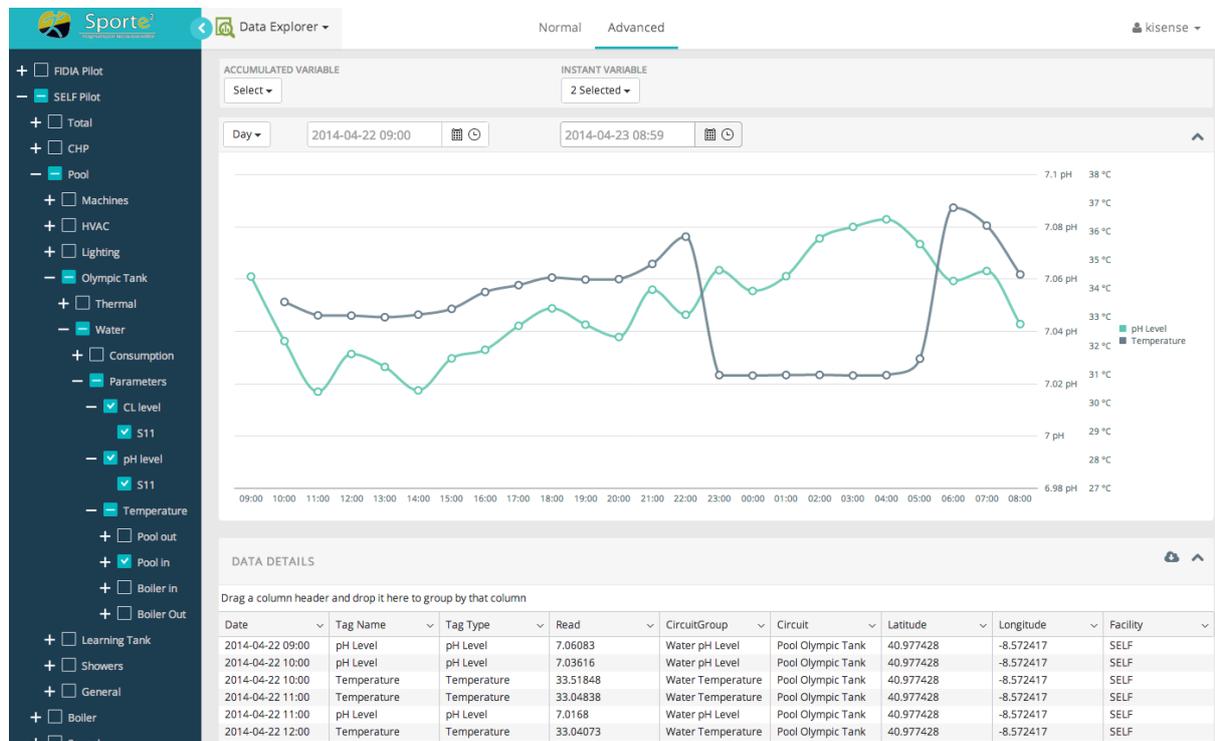


Figure 40 - Advanced Data Explorer layout

In the Advanced Data Explorer, the concept of Circuit Group is not used and variables are always treated as singular variables (i.e. no aggregation of similar variables is used). The graphing variables selector uses the same concept of Accumulated Variables (i.e. energy or water consumption) and Instant Variables (i.e. temperature, pH level). The variables are selected in the top of the graphing zone as shown in detail in Figure 41.

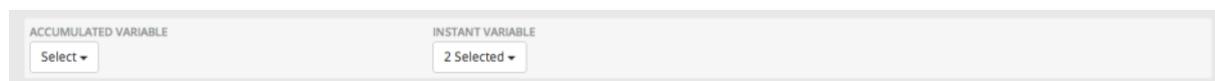


Figure 41 – Advanced Data Explorer variable selection zone

The button Get Data is used similarly to the Normal Data Explorer to fetch the selected variables from the Data Server and to display them in the graphing zone.

One of the main differences of the Advanced Data Explorer user interface regarding the Normal Data Explorer is the selection of the displayed period. The Advanced Data Explorer enables the possibility to use any arbitrary time frame as opposed to the fixed ones found on the Normal Data Explorer. In order to select the displayed time frame, the interface shown in Figure 42 should be used by the user.



Figure 42 - Advanced Data Explorer displayed time frame selector

The user may select the desired aggregation of the displayed values in the graph in the Aggregation selector control. The possible values for the aggregation are:

- Hour – One value is displayed per hour of the graphing period
- Day – One value is displayed per day of the graphing period
- Month – One value is displayed per month of the graphing period
- Year – One value is displayed per year of the graphing period

Along with the Aggregation selector, the displayed time frame selector also contains two distinct fields to choose the start and end date of the displayed data on the graph. Both fields make use of a common input interface using two distinct buttons:

- The calendar icon – For selecting the day, month and year, as shown in Figure 43;

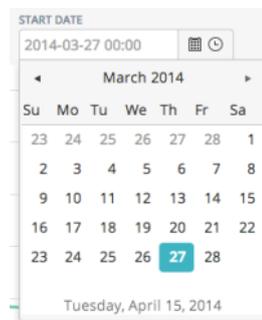


Figure 43 - Advanced view calendar selector

- The clock icon – For selecting the hour and minute, as shown in Figure 44;

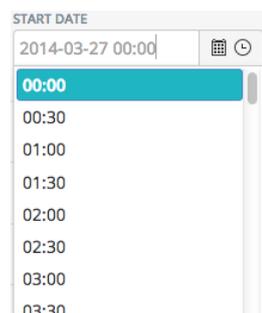


Figure 44 - Advanced view time of day selection

After setting up the desired aggregation, start time and end time, the user should click on the Get Data button to fetch the variable data from the server and display it in the graph.

Another difference regarding the Normal Data Explorer is the possibility that the Advanced Data Explorer has to display the numerical data to the user in the form of a table, as shown in Figure 45.

Date	TagId	Tag Type	Read	CircuitGroup	Circuit	Latitude	Longitude	Facility
2014-03-27 08:00	5428	pH Level	6.57599	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 09:00	5428	pH Level	6.57127	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 10:00	5428	pH Level	6.57782	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 11:00	5428	pH Level	6.57759	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 12:00	5428	pH Level	6.58537	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 13:00	5428	pH Level	6.59214	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 14:00	5428	pH Level	6.57818	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 15:00	5428	pH Level	6.57585	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 16:00	5428	pH Level	6.57974	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 17:00	5428	pH Level	6.57919	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 18:00	5428	pH Level	6.57864	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 19:00	5428	pH Level	6.58111	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 20:00	5428	pH Level	6.60693	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 21:00	5428	pH Level	6.60162	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 22:00	5428	pH Level	6.60917	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-27 23:00	5428	pH Level	6.58894	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF
2014-03-28 00:00	5428	pH Level	6.57393	Water pH Level	Pool Olympic Tank	40.977428	-8.572417	SELF

Figure 45 - Advanced Data Explorer table data view

By dragging the columns into the designated area, it is possible to group the displayed columns in the table.

By clicking in the button on the right of each one of the columns, it is also possible to hide/show columns from the table. Figure 46 shows an example of this menu where it is possible to tick on or off each one of the available columns to show or hide them.

Columns	Visible
Date	<input checked="" type="checkbox"/>
TagId	<input checked="" type="checkbox"/>
Tag Type	<input checked="" type="checkbox"/>
Read	<input checked="" type="checkbox"/>
CircuitGroup	<input checked="" type="checkbox"/>
Circuit	<input checked="" type="checkbox"/>
Latitude	<input checked="" type="checkbox"/>
Longitude	<input checked="" type="checkbox"/>
Facility	<input checked="" type="checkbox"/>

Figure 46 - Advanced View column display selection

Another functionality of the Advanced View is the possibility to download the displayed data in the form of a CSV file from the user interface. To perform this task, the user should set up the desired variables and displayed time frame and then click on the cloud icon found on the top right hand side of the data table view of the interface, as shown in Figure 47.



Figure 47 - Advanced View data download icon

3.3 Examples from real cases

This section shows some examples retrieved from real cases, using real data collected from the SportE2 pilots.

3.3.1 Imported energy from Grid

Figure 48 represents the simple case of imported electric energy from FIDIA pilot, obtained from real data acquired from the facility's meters. Each one of the bars represents the kWh consumption of the respective hour of the day.

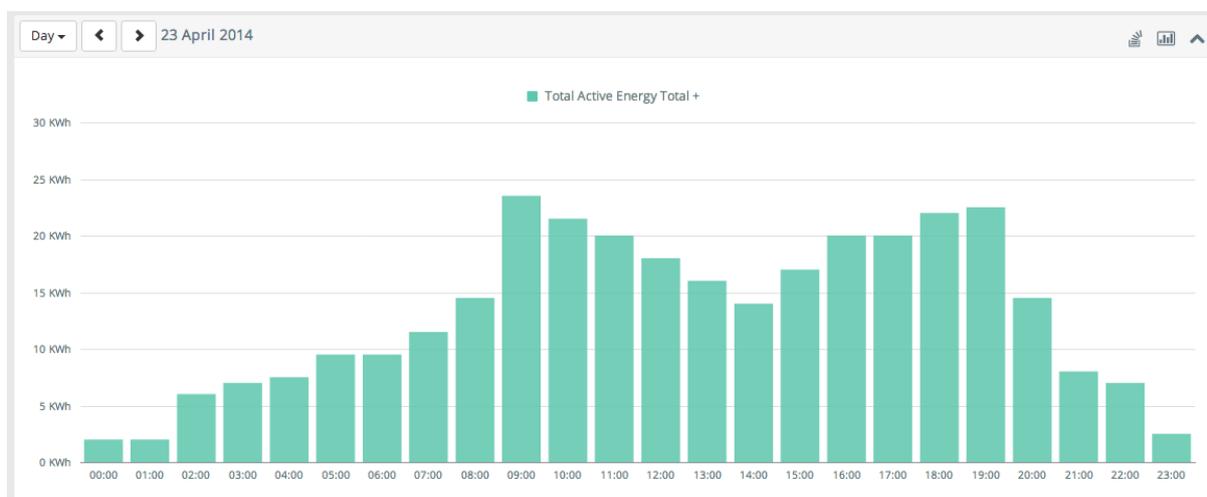


Figure 48 - FIDIA Imported electric energy from Grid over a full day

3.3.2 Grid import, export and CHP production

On Figure 49 the full energy balance for a day of SELF pilot facility is shown. The green bar represents the energy imported from the grid to the pilot. When the CHP system starts to generate electricity (red bar on the graph), the energy imported from the grid decreases and, on some periods of the day, some of the energy from the CHP which is not consumed on the facility is exported to the grid (blue bar on the graph).

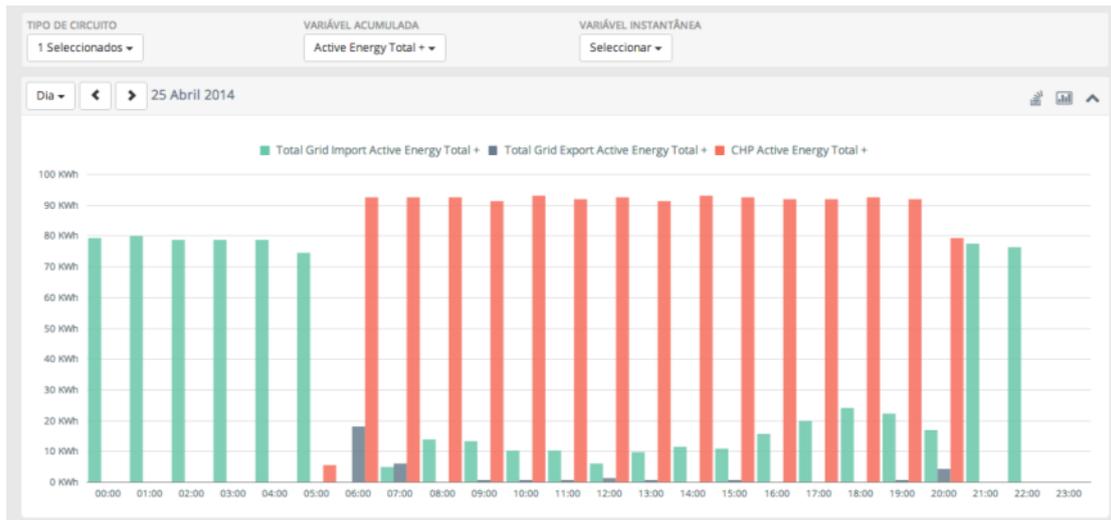


Figure 49 - SELF energy import, export and CHP production

3.3.3 Thermal consumption/production

EMTE thermal consumption for the HVAC system of the swimming pool can be found on Figure 50.

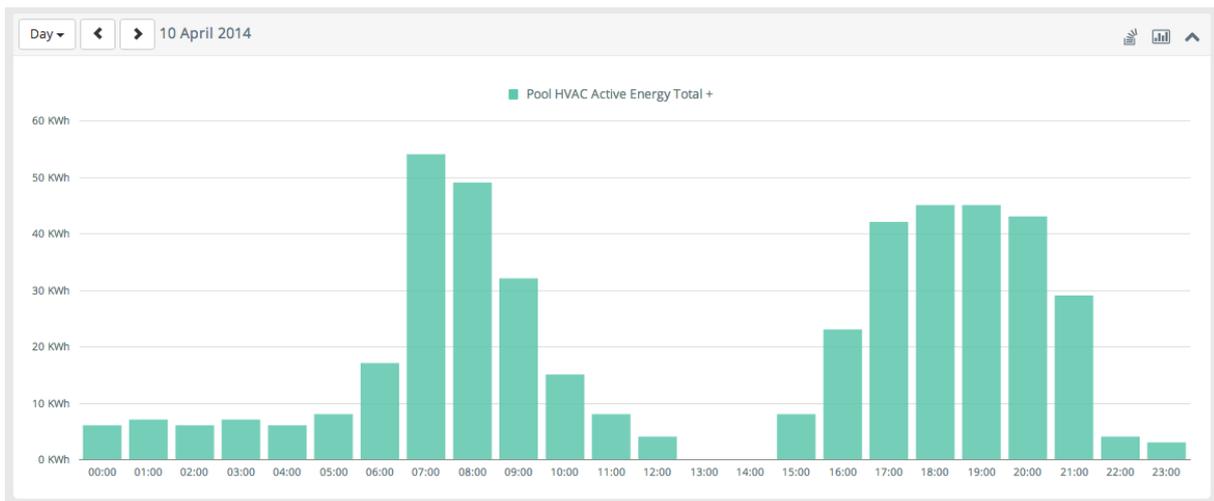


Figure 50 - EMTE swimming pool HVAC thermal consumption

The how module is also capable of displaying thermal energy provided by a solar thermal system. Figure 51 shows the example of the solar system production on the EMTE pilot.

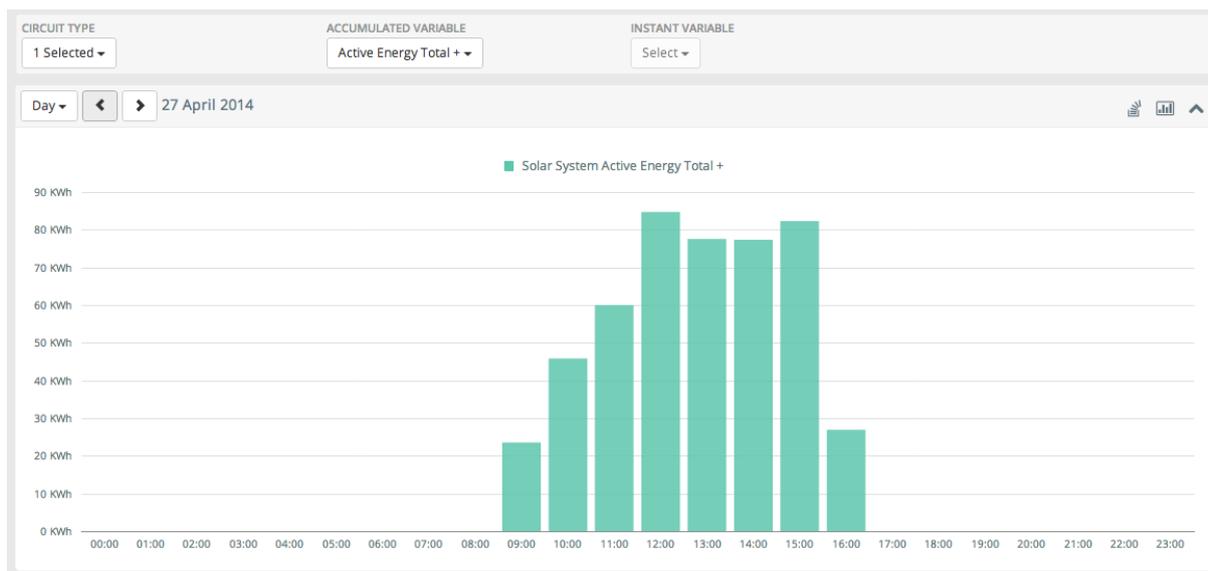


Figure 51 - EMTE solar system production

3.3.4 Water consumption

On Figure 52 we can find the example of the display of the daily water consumption profile for FIDIA over the month of January 2014.

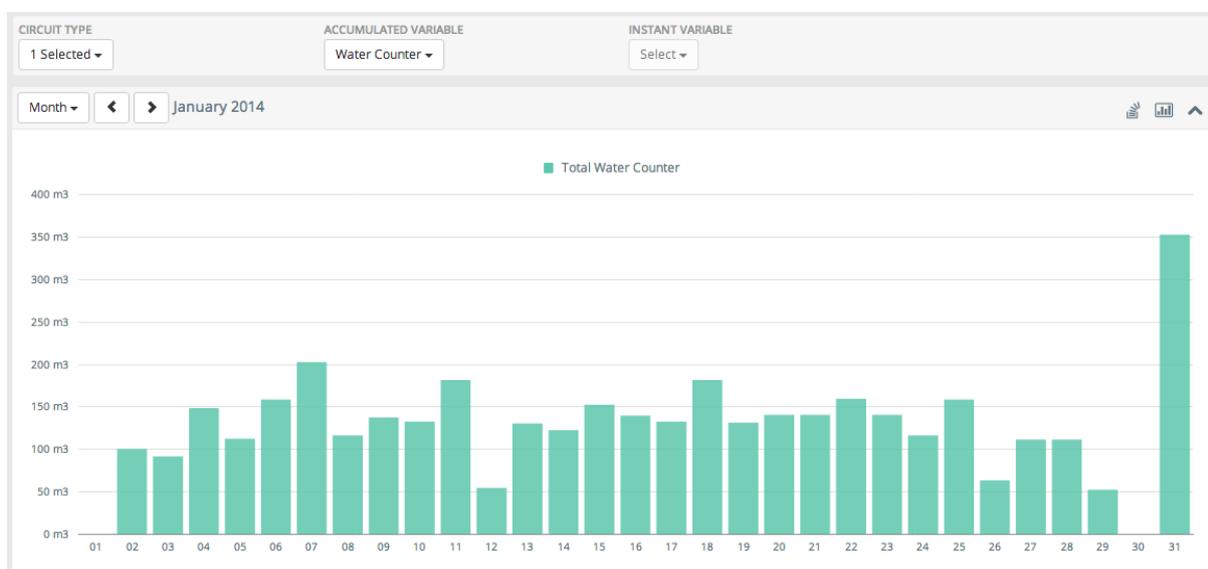


Figure 52 - FIDIA consumed water over a month

3.3.5 Gas consumption

Figure 53 shows the example of the displaying of the gas consumption for the EMTE pilot.

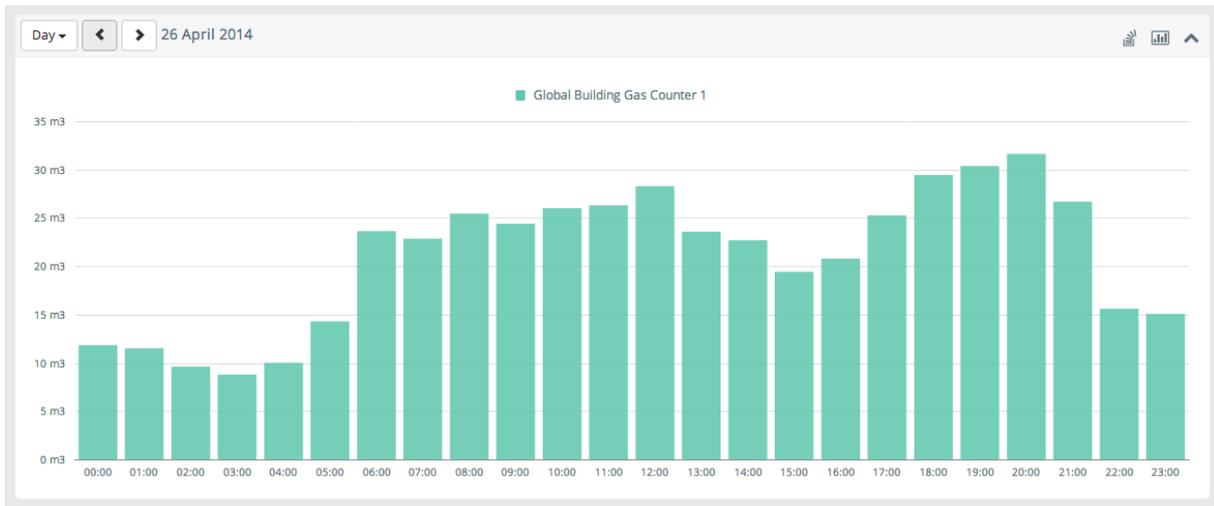


Figure 53 - EMTE gas consumption

3.3.6 PMV

Figure 54 shows the PMV values for the FIDIA swimming pool. Figure 55 shows the equivalent discomfort hours produced by the displayed PMV value.

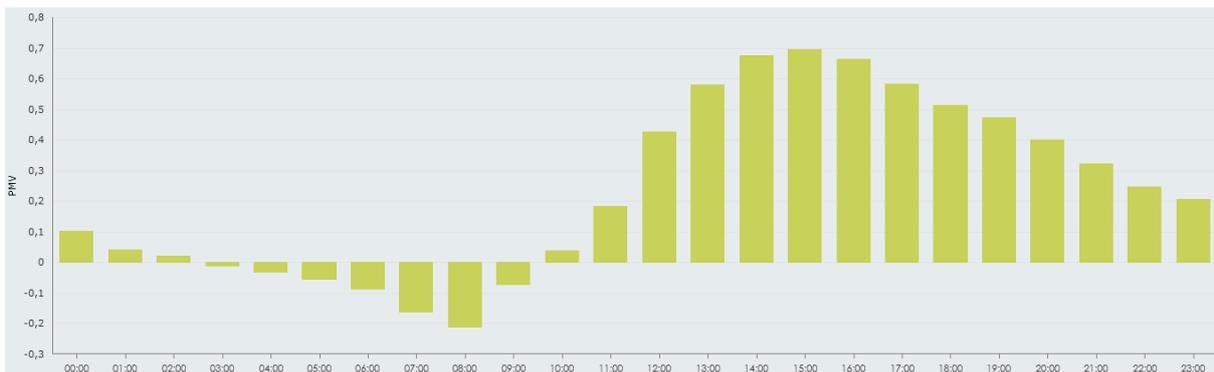


Figure 54 - PMV calculation for FIDIA swimming pool

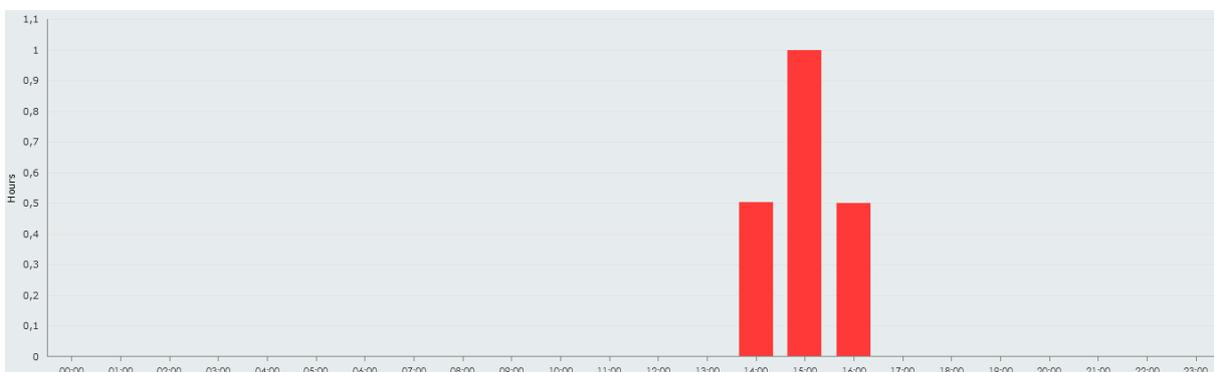


Figure 55 - Calculation of discomfort hours

3.3.7 Air temperature and humidity monitoring

Figure 56 and Figure 57 show examples of the displaying of air temperature and humidity parameters for the SELF pilot swimming pool space.

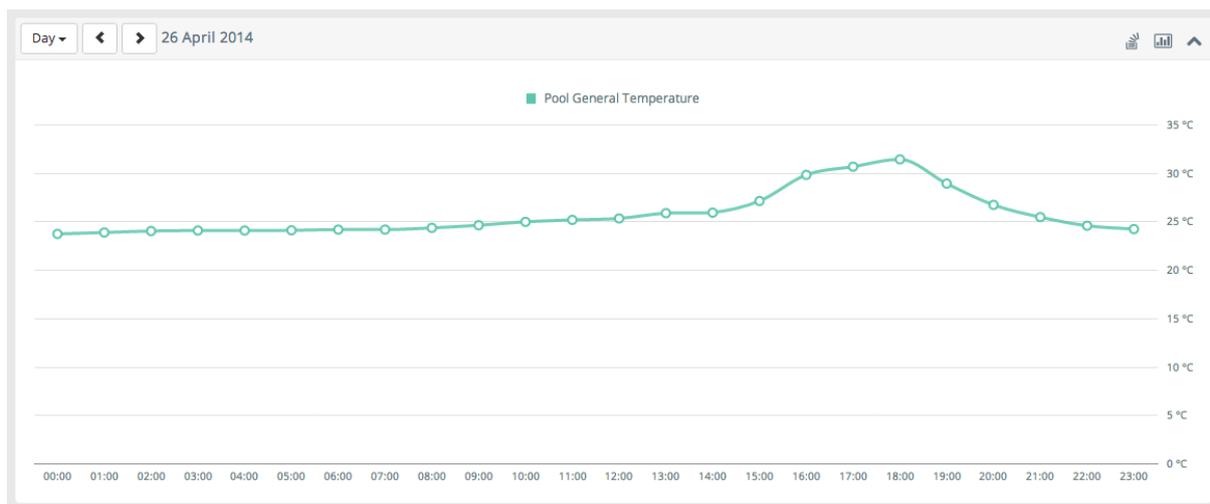


Figure 56 - SELF Swimming pool air temperature



Figure 57 - SELF Swimming pool air humidity

3.3.8 CO2 concentration level

Figure 58 shows an example for the CO2 concentration of EMTE’s room 1 (a small gym) over the different hours of the day.



Figure 58 - EMTE Room 1 CO2 concentration

3.3.9 CL2 concentration level

Figure 59 shows an example for the displaying of the concentration level of CL2 on the SELF pilot swimming pool environment.



Figure 59 - SELF Swimming pool CL2 concentration level

3.3.10 Water parameters

Figure 60 displays the advanced view of the data explorer showing the values for the pH level and CL2 concentration of the SELF swimming pool. Figure 61 presents the water temperature for FIDIA swimming pool.

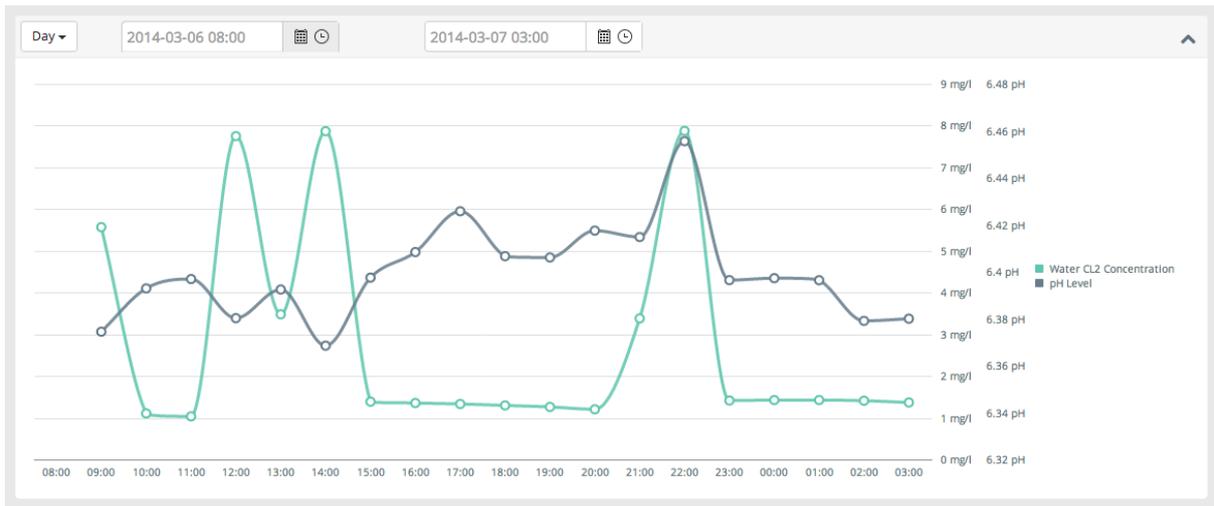


Figure 60 - SELF pilot water parameters (advanced view)

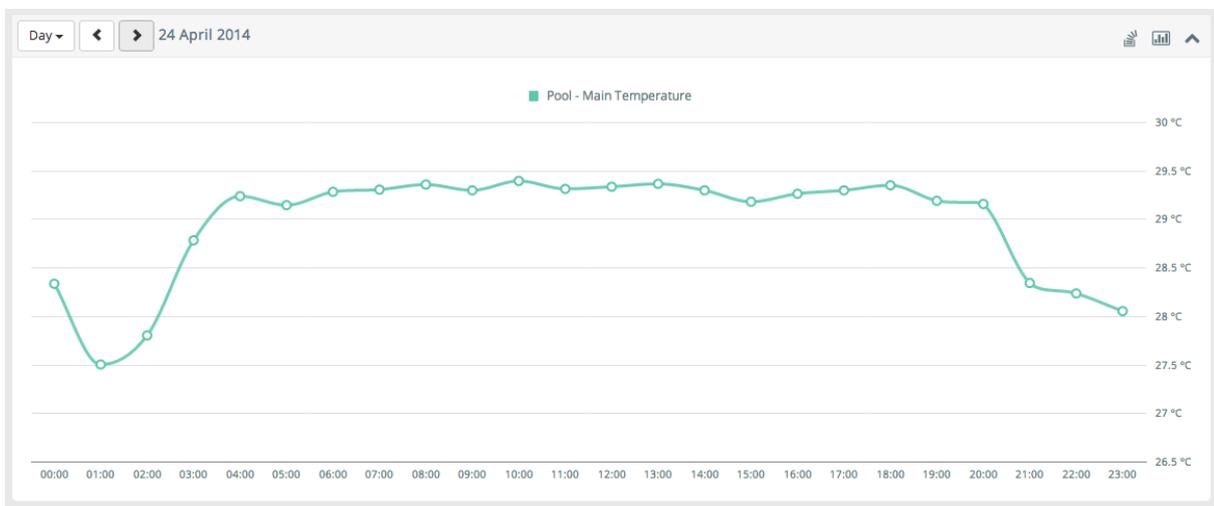


Figure 61 - FIDIA pilot swimming pool water temperature

3.3.11 HVAC parameters

Figure 62 shows the graphing of the measured air speed inside the FIDIA pilot HVAC air supply duct.

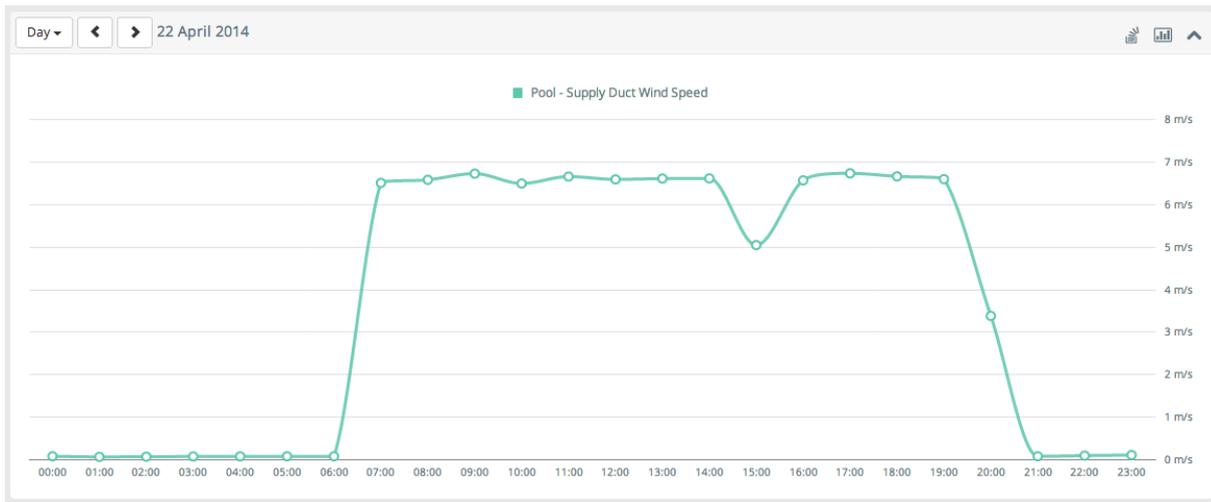


Figure 62 - FIDIA pilot HVAC duct air speed

3.3.12 Light level

Figure 63 shows the measured light levels in EMTE pilot regarding the Room 1, Multisport Court and Corridor.

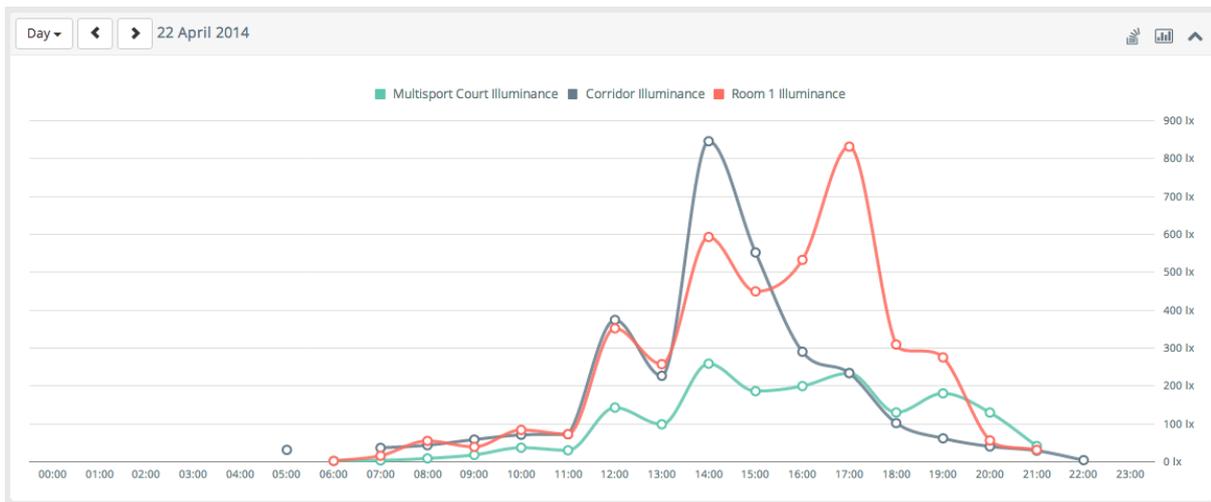


Figure 63 - EMTE light level measurement at Room 1, Multisport Court and Corridor

3.3.13 Occupancy

Figure 64 shows the graph for the registered occupancy for FIDIA’s fitness room on 11th of April 2014.



Figure 64 - FIDIA occupancy for the Fitness room

4. SENSOR OPTIMIZATION UNIT USER MANUAL

4.1 Introduction and system requirement

The Sensor Optimization Unit (SOU) is a stand-alone software to evaluate the optimal positioning of temperature and humidity sensors inside large areas, typical of sport facilities. The objective of such a tool is to support the How Module in improving the measurement accuracy of temperature and humidity in large spaces like sport areas, whilst proposing a cost-effective solution.

The targets of such a tool are sport environments as indoor swimming pool, gym/fitness rooms and multi-sport courts. The most important characteristics that these environments have in common are:

- significant height and floor area that cause air stratification and stagnation;
- HVAC control based on the return duct and not on the occupied zone thermal conditions.

The natural customer for the SOU can be:

- the designer of an HVAC system for a new sports facility or a retrofit of existing building;
- the designer and vendor of the How Module;
- the technician of the sports facility can use the SOU for refurbishment of the heating plant of the building;
- the energy manager of the sports facility when implementing strategies to improve the energy and comfort performances of the sports facility.

The tool runs on desktop or laptop machines equipped with Microsoft Windows 7, 8 and 8.1 (both x86 and x64 versions) and .Net Framework 4.5. If not already present in the machine, the Matlab Runtime Compiler is installed during the installation of the SOU. The computational requirement is low, so it can run on both low and high performing machines and the whole process can be performed in a short time; apart from the time required for the user to input data, the time needed to perform the optimization process is lower than 2 minute for a laptop with medium performances.

4.2 User manual

This chapter drives the user through the Graphical User Interface of the Sensor Optimization Unit, in order to get started using it. After the installation, launching the program it opens a window as in Figure 65.



Figure 65 – SOU opening window

The upper part of the GUI is a sort of container of the sport facility information, it is organized using a tab menu and the needed information are split under six main categories (General info, Construction, Plant, Internal, Profiles and Positioning); while the lower part of the GUI is a commands menu, it allows the user to do operations as validate the data, save the project, open an existent project and close the program.

The basic idea behind the SOU user interface is that, starting from general information of the sport facility, the user fills each section with needed input data. Two specific buttons (Figure 65, elements 10 and 11) are fundamentals to guide the user through the interface to the last tab “Positioning” (Figure 65, element 6), where the software completes and displays the calculus of the best positioning solution of sensors installation inside the ambient of study.

Using the tabs (Figure 65, elements from 1 to 6), the user can browse along the different sections of the menu, in order to have a visual check of the data inputs: the ones already implemented and the missing data.

The commands bar is illustrated in Figure 65; it is composed of five buttons in order to allow the following operations:

- Save current project (Figure 65, element 7);
- Open an existent project, already saved into the pc (Figure 65, element 8);
- Close the program (Figure 65, element 9);
- Data validation: the software checks for each tab, that all the field are filled and the data are suitable (Figure 65, elements 10 and 11);

4.2.1 General info

When the user starts the program, the main window opens with the “General info” tab ready to work. The contents of this tab are split in three main categories: site information, wheater data and shape (see Figure 65).

Starting from the site information (Figure 66(a)) the user should insert the name of the building into the text box “Building name” (Figure 66(a), element 1), then pushing the button “Latitude and Longitude Web service calculator”, an additional window opens as in Figure 66(b).

The user has to insert the country and the city of the sports facility and then press the Google icon; it connects the software to a service powered by Google. This service provides the values of Latitude and Longitude for the location, finally pressing “OK”, the values move directly to the related text boxes (Figure 66(a), elements 4 and 5).

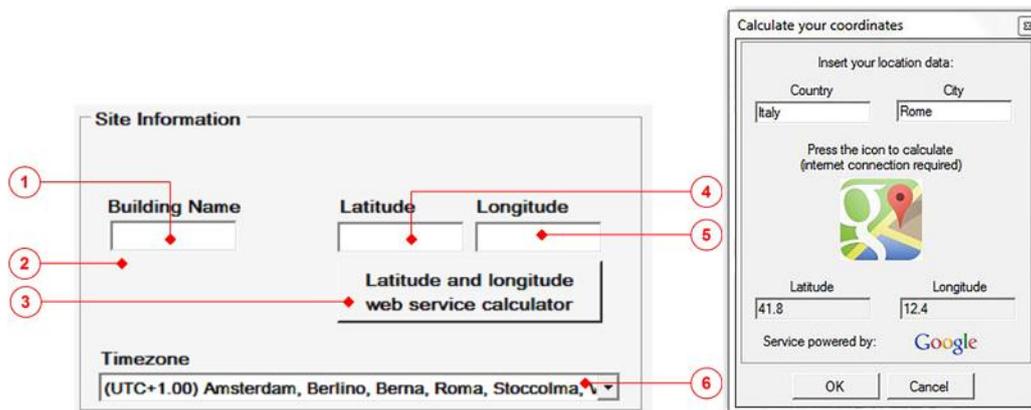


Figure 66 – (a) Site information section – (b) Latitude and Longitude web service calculator

The described web service calculator needs of an internet connection to run; in case the SOU runs offline, the user can always inserts the values of Latitude and Longitude manually inside the two specific text boxes (Figure 66(a), elements 4 and 5).

The next step is to define the time zone, the user can easily pick the right one from the list of options available on the list box “Timezone” (Figure 66(a), element 6).

After the site information, the user has to deal with wheatear information. The SOU simulates the environment of study, using external input coming from a wheatear file that contains data as: diffuse solar radiation, exterior air temperature, direct solar radiation, cloud cover, outside relative humidity, wind velocity and wind direction; all these data are different depending on the geographical location of the facility.

How to get these data? The user should move the attention to the weather Data section, in Figure 67, then should follow the step-by-step description reported below:

1. Pressing a button (Figure 67, element 1) the SOU connects directly to web site of the U.S. Department of energy, where weather data for more than 2100 locations in the world are available in EnergyPlus weather format; now inside this web page, the user selects the region of interest and download a zip file that contains three files: IDFEditor document, EPW file and a STAT file; the EPW file is the important one, that needs to be saved;
2. A second button (Figure 67, element 2) opens an additional window that requests the directory of the EPW file (available from step 1), once it has been found, just press the command OPEN on the window; if the previous steps were successfully completed, then the two boxes (Figure 67, element 6 and 7) will be automatically filled with the EPW file path and EPW file name;

3. After the second step, the button (Figure 67, element 3) is enabled, so the user can push it in order to finalize the implementation of the weather data, if the operation succeed, a new window that tells “Meteo file created” appears, then just push “OK” and the meteo process is complete.

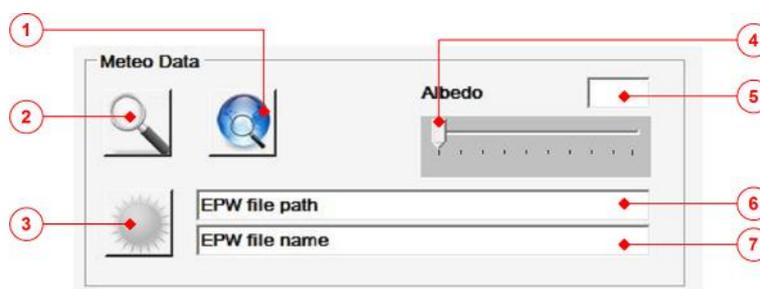


Figure 67 – Meteo data section

After created and loaded the meteo data, the user have to set the Albedo using a track bar (Figure 67, element 4), the value will appear on the box named “Albedo” (Figure 68, element 5). The Albedo is the fraction of incident radiation that it is reflected by a surface; the concrete, buildings have a lower Albedo than rural areas with trees and vegetation; the most common values of Albedo are shown below:

Natural surface types	Approximated albedo
Blackbody	0
Forest	0.05–0.2
Grassland and cropland	0.1–0.25
Dark-colored soil surfaces	0.1–0.2
Dry sandy soil	0.25–0.45
Dry clay soil	0.15–0.35
Sand	0.2–0.4
Mean albedo of the earth	0.36
Granite	0.3–0.35
Glacial ice	0.3–0.4
Light-colored soil surfaces	0.4–0.5
Dry salt cover	0.5
Fresh, deep snow	0.9
Water	0.1–1
Absolute white surface	1

Figure 68 – Albedo most common values

The last part the site information is dedicated to the shape of the facility, first of all the user have to define the type of the building (Swimming pool, Gym, Multisport Court) through a dedicated list box (Figure 69, element 1).

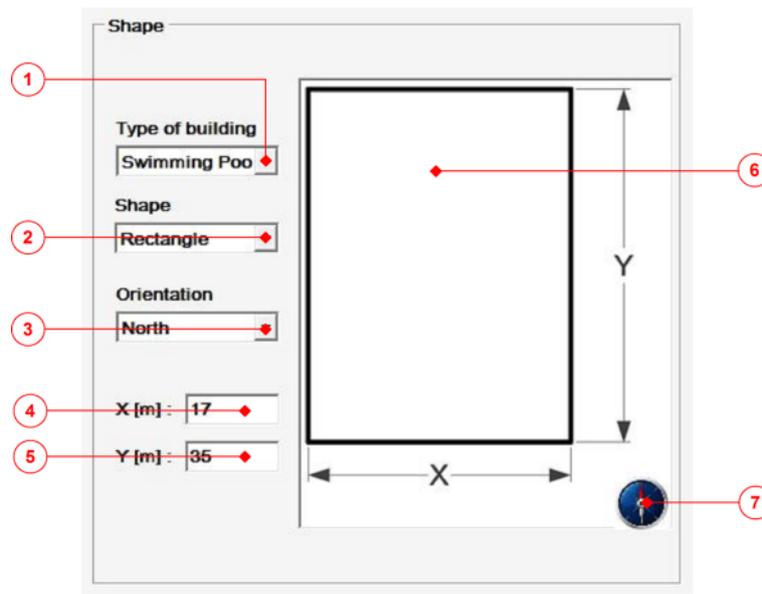


Figure 69 – Shape section

The list box named “Shape” (Figure 69, element 2) allows the definition of the shape of the environment and then the shape appears on the dedicated box (Figure 69, element 6) as a drawing in plane of the room.

The next step consists on the definition of the Orientation of the environment: the user should imagine himself positioned inside the room, facing one of the two shorts sides of the room (Figure 69, element 6, values X), then the user is actually facing the cardinal direction that needs to be selected on the list box signed as “Orientation” (Figure 69, element 3).

Finally the values of the geometry of the room expressed in [m], can be defined through the two specific text boxes (Figure 69, elements 4 and 5).

After that all the needed information have been inserted in a proper way, the user has to validate the filled data using the specific button (Figure 65, element 10): a dedicated algorithm checks the congruence of all the inputs.

For example, if the user inserts as x value a letter, instead that a number, then a message appears, it asks you to correct the value, in order to repeat the data validation process.

In case of successfully submission of the data, a small window that confirms it, will appears, then just press “Ok” and the software brings you directly to the new section “Construction”.

4.2.2 Construction

This section allows the definition of the structural characteristics of the facility, as are walls stratigraphy, floor, roof and windows and the window is organized into four main areas:

- Graphical area (Figure 70, element 1);
- The tab menu (Figure 70, element 2);
- Shadows (Figure 70, element 3);
- Windows (Figure 70, element 4 and 5);

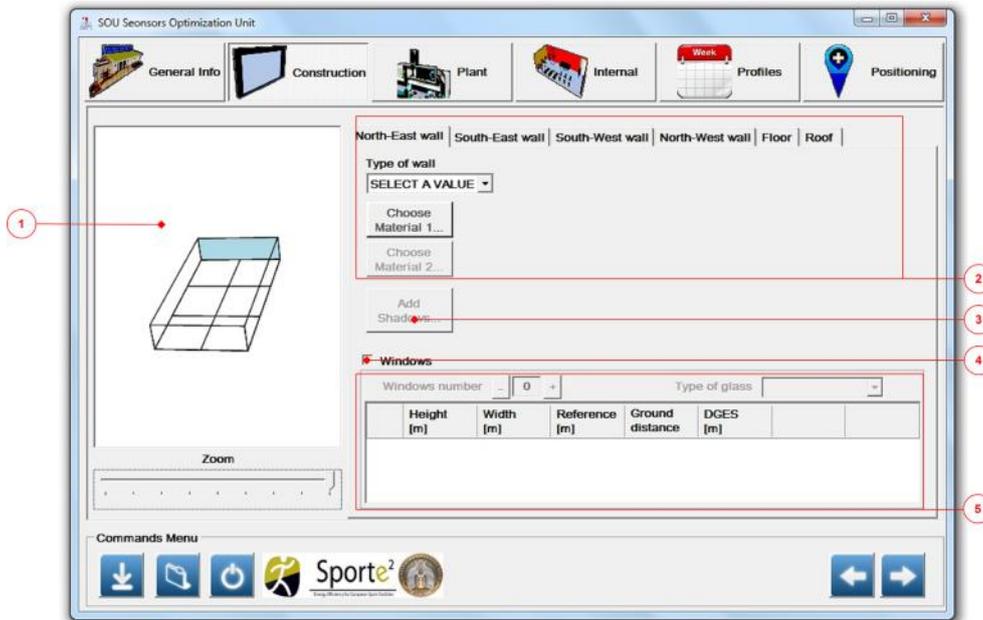


Figure 70 – SOU construction section

1. Select from the tab menu (Figure 70, element 2) one wall between the four that compose the perimetral envelope; note that the highlighted surface of the room of the drawing(Figure 70, element 1), corresponds with the wall selected from the tab menu;
2. Select from the list box the Type of wall, the available are:
 - External: a wall exposed to the exterior;
 - Internal: a wall entirely shared with an adjacent indoor zone;
 - Mixed: a wall partially external and internal.

Then the next steps depend on the type of wall selected.

External wall

3. Select from the list box named “Type of wall” (element 1 on Figure 71), the option “External”;

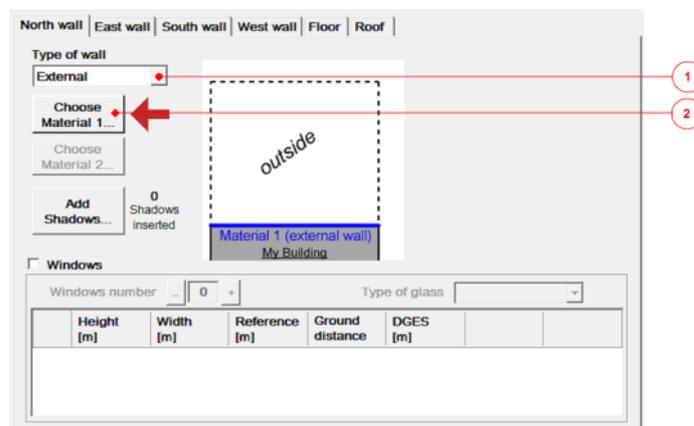


Figure 71 – External wall section

4. In order to define the wall stratigraphy, push the specific button called “Choose Material” (element 2, Figure 71), then it opens a window named “Material Layers”

(Figure 72); this window enables the possibility to create a multilayer stratigraphy (element 7);

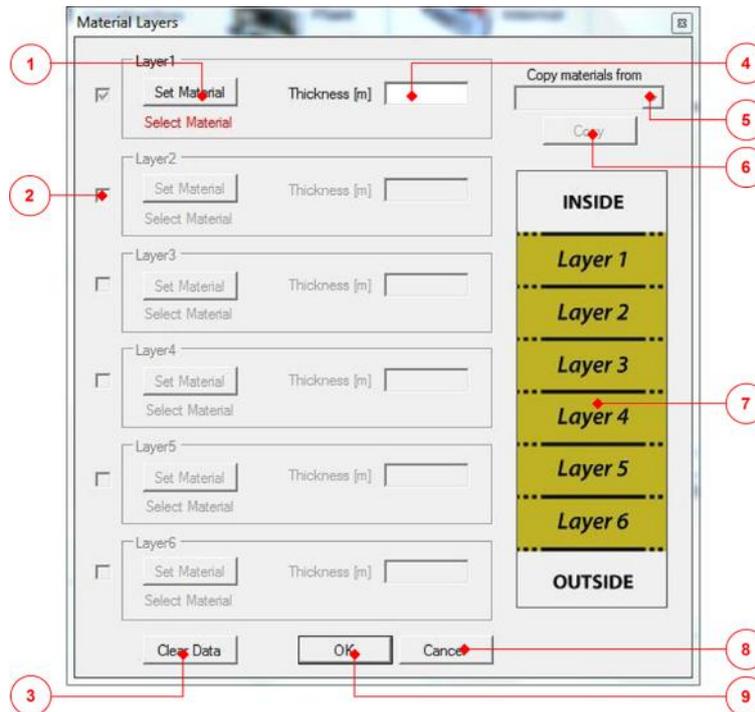


Figure 72 – Material layers window

5. Move to the layer 1 and push the button “Set Material” (element 1, Figure 72), it opens the “Material Selection” window (Figure 73);

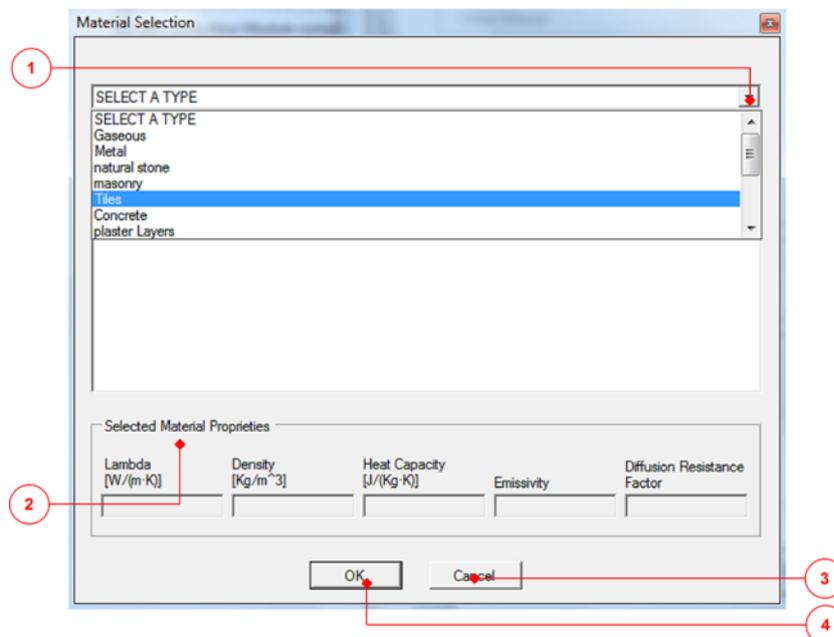


Figure 73 – Material selection window

6. Select the type of the material using the list box “SELECT A TYPE” (element 1 Figure 73); the type of materials available are: gaseous, metal, natural stone, masonry, tiles, concrete, plaster layers, other inorganic materials, inorganic insulators, organic insulators, wood materials, bound organic materials except wood and plastic, hard plastic, roof;

7. Once the type has been selected, another list appears into the same box (element 1 Figure 74), this list is composed of all the available materials from a specific type;
8. After one material has been selected, the values of its properties (as Lambda, Density, Heat Capacity, Emissivity, Diffusion Resistance Factor) fill the dedicated boxes (element 2, Figure 74); if the user is looking for a material that it is not present in the proposed list, it is suggested to take a material that shows similar properties respect the desired one;

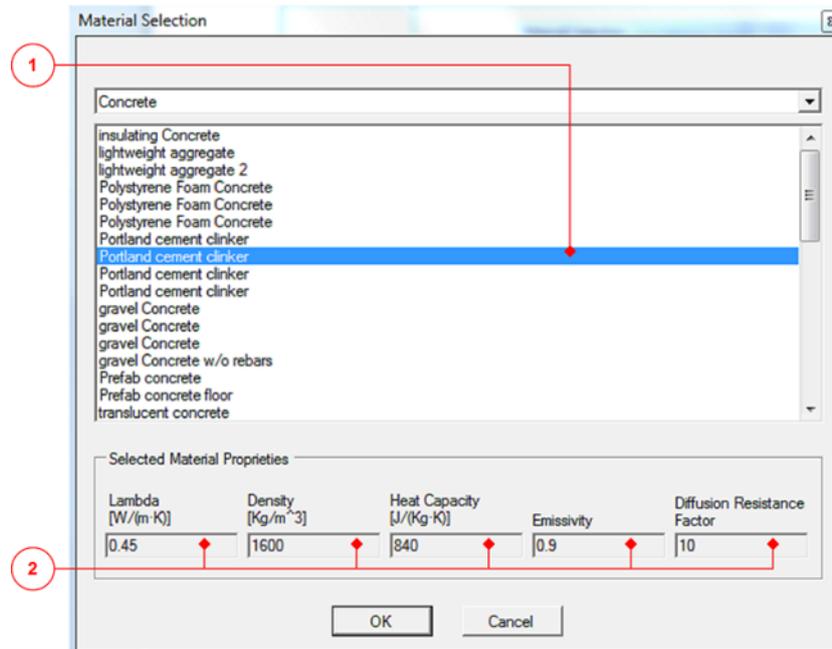


Figure 74 – Material list properties

9. Once the user has found the right material, just press “OK” (element 5), then the material will be inserted into “Layer 1” (element 1, Figure 72);
10. Define the thickness [m] of the Layer 1, using the specific text box (element 4, Figure 72);
11. If the user want to add a “Layer 2” to the wall, just click the specific check box (Figure 72, element 2) then the layer becomes active and properties can be set following the procedure from point 5 to 10; please note that maximum six layers can be added to a wall stratigraphy;
12. When the stratigraphy is complete, just press “OK” (element 9, Figure 72) and the stratigraphy will be saved and automatically assigned to the wall, which the user is working on; please note that the working wall is also recognizable through the specific label of orientation (Figure 70, Tab menu);

Other notes:

- The stratigraphy of the wall is defined starting from the Layer 1 to Layer 6, from inside to outside (element 7, Figure 72);
- During the definition of the wall stratigraphy, the user can use a functionality named “Copy materials from”: using the specific list box (element 5 in Figure 72), the user can select historical stratigraphy (only from the current project), then using the button

“Copy” (element 6, Figure 72), the user defines the characteristics of a new wall through an existent one;

- The button “Clear data” (element 3, Figure 72) can be used to clear all the layers present into the window (Figure 72) at the same time, in order to start from zero and define a new stratigraphy.

Internal wall

- Select from the list box (element 1, Figure 75), the option “Internal”;

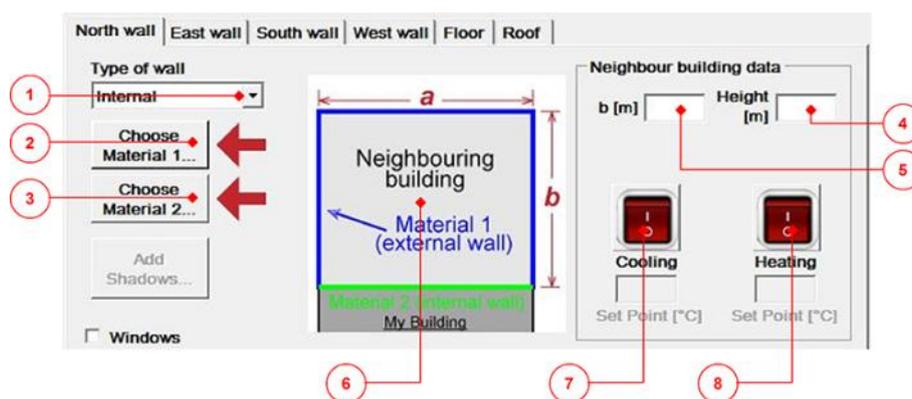


Figure 75 – Internal wall section

- The button “Choose material 1...” (element 2, Figure 75) defines the stratigraphy of the external wall of the neighboring room (element 6, Figure 75); then the stratigraphy can be defined following the procedure from point 3 to 12, fully described into the previous section “External wall”;
- The button “Choose material 2...” (element 3, Figure 75) defines the stratigraphy of the internal wall, that in this case it is a wall of the zone subject of optimization; the stratigraphy of the wall can be defined following the procedure from point 2 to 8, fully described into the previous section “External wall”;
- To better characterize the neighboring building, the user has also to insert:
 - b [m] = the length of the side of the adjacent building that is orthogonally oriented to the studied environment (Figure 75, element 5);
 - Height [m] = the height of the adjacent building (Figure 75, element 4);
 - In case of air-conditioned environment, the user should press the specific on/off switch (element 7 and/or 8, in Figure 75), in order to enable the two text boxes present under the two switches; in case that the environment has a cooling system, the user can insert the cooling set point [°C], while in case of presence of an heating system, the user can insert the heating set point [°C]; in case of presence of both the solutions, heating and cooling systems, the user can insert both the set points; in case of no heating or cooling system, the user has to leave the two switches off;

Mixed wall

- Select from the list box (element 1, Figure 76), the option “Mixed”;
- The button “Choose material 1...” (element 2,) defines the stratigraphy of the external wall of the neighboring room (element 6, Figure 76); the stratigraphy can be defined

following the procedure from point 3 to 12, fully described into the previous section “External wall”;

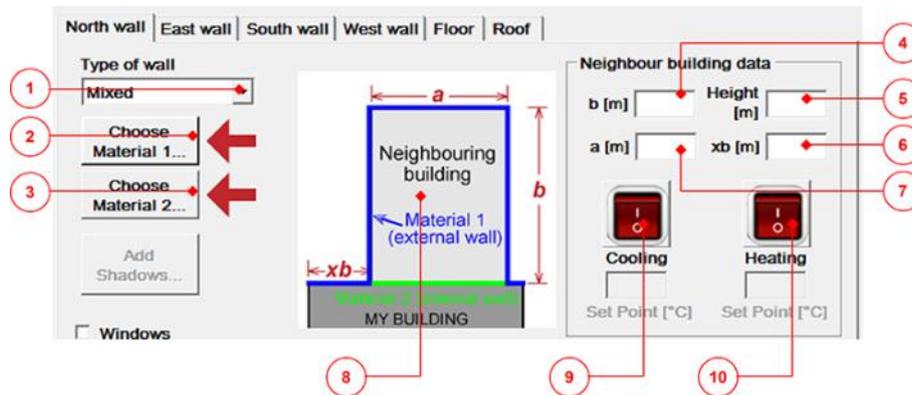


Figure 76 – Mixed wall section

5. The button “Choose material 2...” (element 3, Figure 76) defines the stratigraphy of the internal wall that is comprised by the studied zone; the stratigraphy of the wall can be defined following the procedure from point 2 to 8, fully described into the previous section “External wall”;
6. The user has also to insert:
 - b [m]= the length of the side of the adjacent building that is orthogonally oriented to the study environment (Figure 76, element 4);
 - Height [m]= the height of the adjacent building (Figure 76, element 5);
 - a [m]= the length of part of the wall defined as internal one (Figure 76, element 7);
 - xb [m] = the length of part of the wall defined as external one (Figure 76, element 6);
 - In case of air-conditioned environment the user should press the specific on-off switch (element 7 and/or 8, in Figure 76), in order to enable the two text boxes present under the two switches; in case that the environment has a cooling system, the user can insert the cooling set point [°C], while in case of presence of an heating system, the user can insert the heating set point [°C]; in case of presence of both the solutions, heating and cooling systems, the user can insert both the set points; in case of no heating or cooling systems, just live the two switches off;

Once that the stratigraphy of the first wall has been defined in a correct way, the main window appears as in Figure 77, then the symbol (element 1, in Figure 77) shows up, that means that the stratigraphy has been assigned to the wall, then the button “Add Shadows” (Figure 77, element 2) becomes enabled.

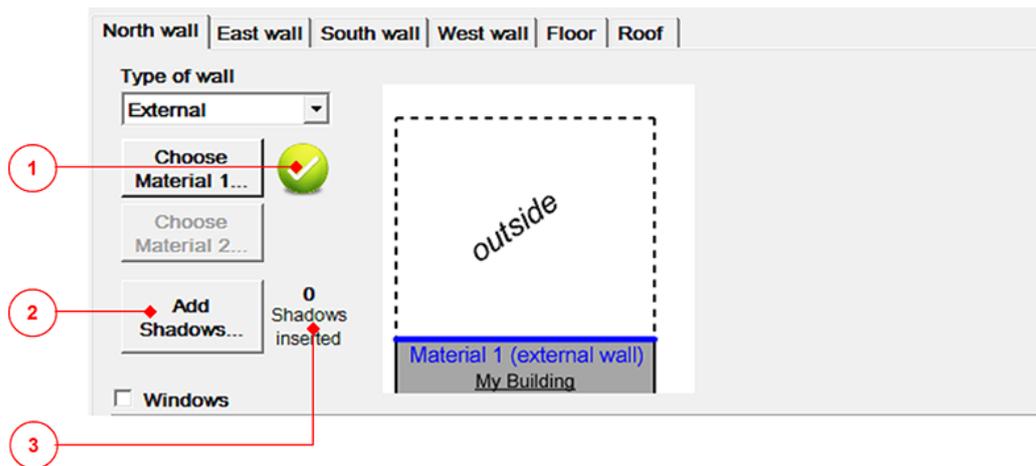


Figure 77 – Screen shot of the construction section window

This button allows the possibility to implement the properties of external shading elements. A step by step description is reported below:

1. Press the button called “Add Shadows...”;
2. A new window opens (Figure 78);

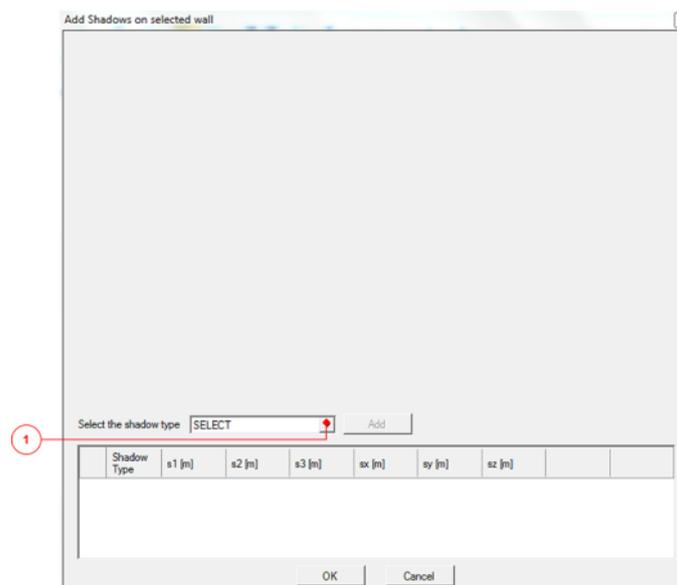


Figure 78 – “Add Shadows” window

3. Select the type of shadow using the list box (Figure 78, element 1); there are two options: tree or structures;

Tree

4. If the shadow element is a tree, the window will become as in Figure 79;

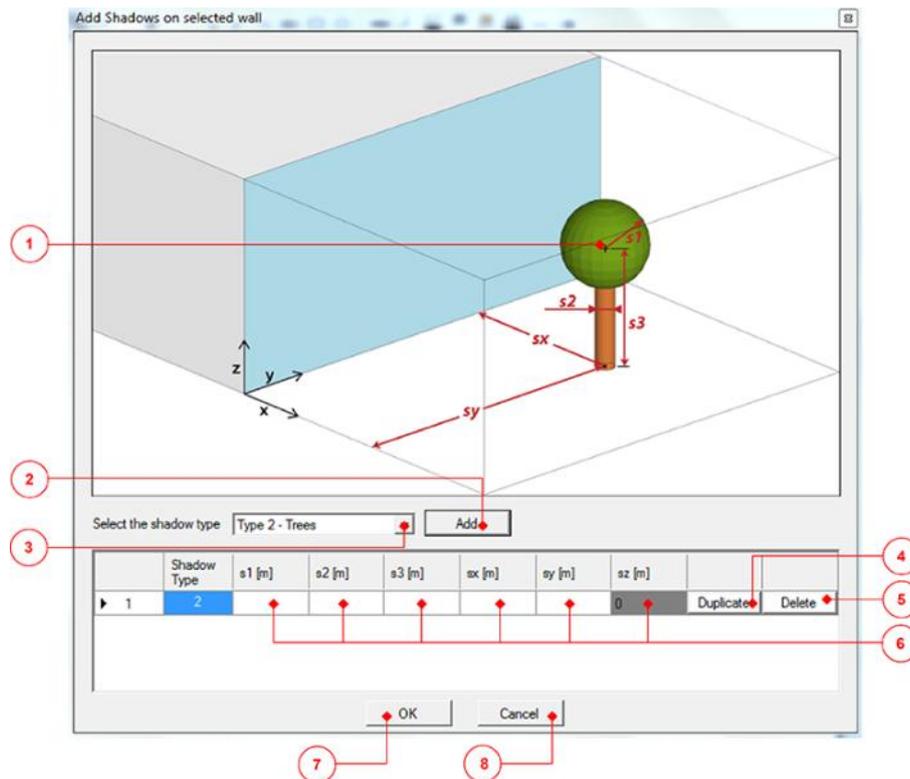


Figure 79 – Type of shadow: Trees

5. Pressing the button “Add” (Figure 79, element 2), a new row is added to the table and the row represents the shadow elements, then using the text boxes (Figure 79, element 6), the user can insert the following shadow characteristics:
 - S1 [m] = value of radius sphere (crown);
 - S2 [m] = value of radius cylinder (trunk);
 - S3 [m] = height cylinder (height to the center of the crown);
 - Sx [m], Sy [m], are the coordinates of the base of the cylinder (trunk) from the left corner of the wall, facing it from outside (as it can be seen in Figure[]);
 - Sz [m] is always zero in this case;
 - More tree are possible: just push the “Add” button (Figure 79, element 2) to get a new tree with new data to fill in;
 - Use the “Duplicate” button (Figure 79, element 4), if the user want to copy a tree already implemented;
 - Use the “Delete” button (Figure 79, element 5) to erase a tree already present into the table;
6. Press “OK” (Figure 79, element 7) to save and close the window;
7. Press “Cancel” (Figure 79, element 8) to close the window without saving the work.

Structures

4. If the shadow element is a structure, the window will become as in Figure 80;

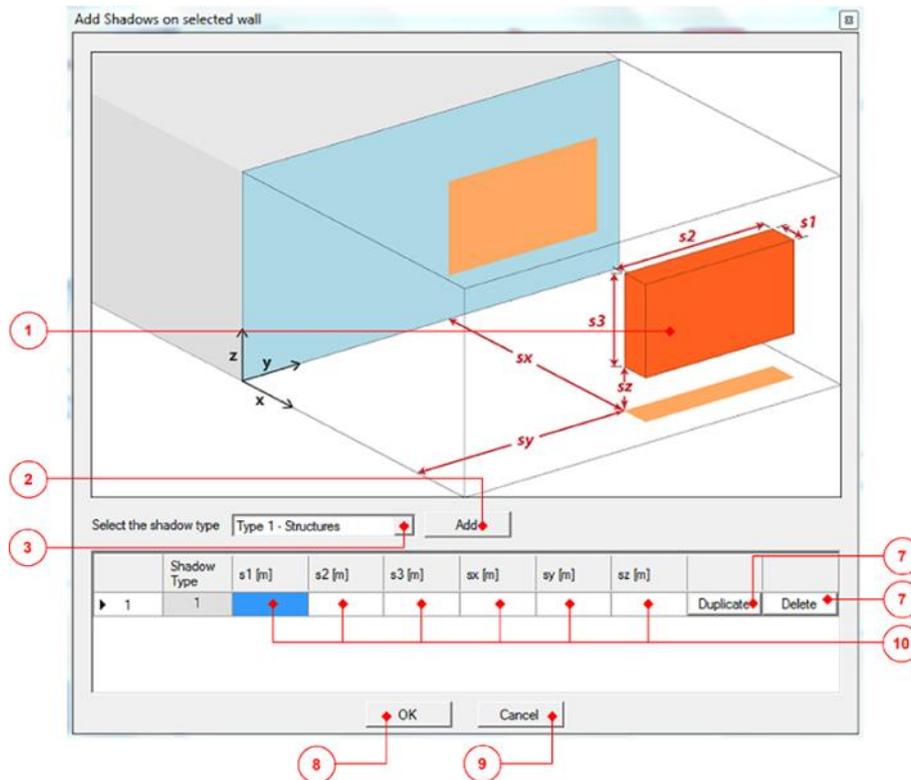


Figure 80 – Type of shadow: Structures

5. Pressing the button “Add” (Figure 80, element 2), a new row is added to the table and the row represents the shadow elements, the using the text boxes (Figure 80, element 6), the user can insert the following shadow characteristics:
 - S_1 [m] = size in x-direction (Figure 80, element 1);
 - S_2 [m] = size in y-direction (Figure 80, element 1);
 - S_3 [m] = size in z-direction (Figure 80, element 1);
 - S_x [m], S_y [m], S_z [m] are the coordinates of the left corner of the structure, from the left corner of the wall, facing it from outside (as it can be seen in Figure 80);
 - More structures are possible: just push the “Add” button (Figure 80, element 2) to get a new structure with new data to fill in;
 - Use the “Duplicate” button (Figure 80, element 4), if the user want to copy a structure already implemented;
 - Use the “Delete” button (Figure 80, element 5) to erase a structure already present into the table;
6. Press “OK” (Figure 80, element 7) to save and close the window;
7. Press “Cancel” (Figure 80, element 8) to close the window without saving the work.

Then the user should check the presence or not of one or more windows for the considered perimetral wall; the procedure to insert the windows is described below:

1. Click on the check box “windows” (Figure 81, element 1);

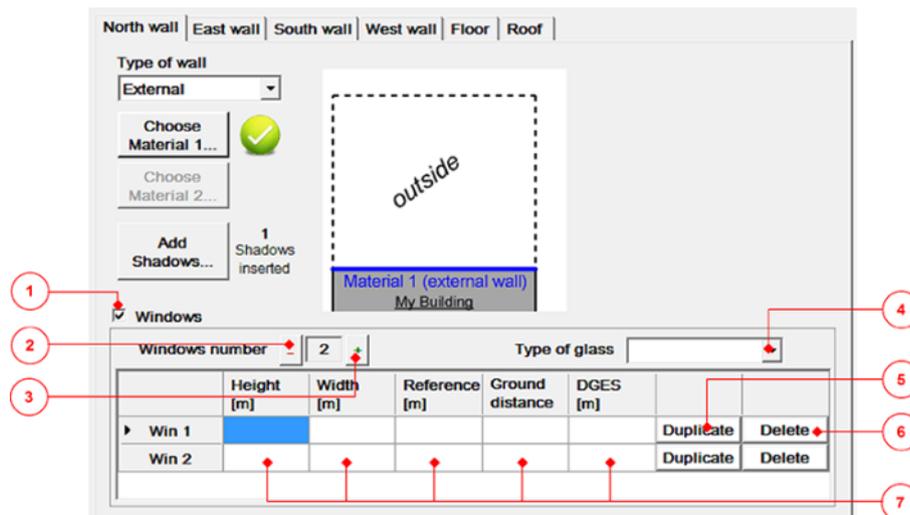


Figure 81 – Windows section

2. Press the button “+” or “-” (Figure 81, element 3 and 2 respectively) to add or delete a window element that in this case is represented by a row of the table in Figure 81;
3. Use the list box “Type of glass” (Figure 81, element 4) to select the typology of glass available on the list;
4. Using the text boxes (Figure 81, element 7), the user has to insert the following characteristics of the window:
 - Height of the window [m];
 - Width of the window [m];
 - Reference [m]: is the distance of the left corner of the window from the left corner of the wall, facing the wall from outside;
 - Distance [m] from the ground of the window ;
 - DGES [m]: is the glazing distance from the exterior surface;
5. More windows are possible: just push the “+” button (Figure 81, element 3) to get a new empty row in the table, ready to be filled with new data;
6. Use the “Duplicate” button (Figure 81, element 5), if the user want to copy a window already implemented;
7. Use the “Delete” button (Figure 81, element 5) to erase a window already present into the table;

The described procedure applied for the selected wall has to be extended to the remaining perimetral walls that compose the environment: just select from the tab menu the next wall and repeat this procedure.

Then the next step consists on the definition of the characteristics of the floor.

Floor

The user has to establish if the study environment is at a low ground or not; then selecting the tab “Floor” from the tab menu (Figure 82, element 1), the main window appears as in Figure 82.

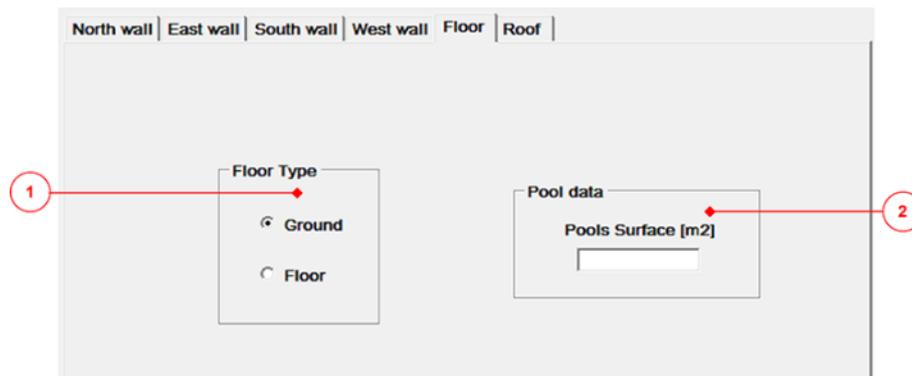


Figure 82 – Floor section

In case that the environment is situated at a low ground, the user has to select the “Ground” radio button inside the “Floor Type” panel (Figure 82, element 1); in case of the room of study is situated on an upper floor level than the ground level, the user has to select the “Floor” radio button inside the “Floor Type” panel (Figure 82, element 1); in case of the environment of study is an indoor swimming pool, the user has to insert the value in [m²] of the pool surface into the text box “Pool surface” inside the panel “Pool data” (Figure 82, element 2).

Roof

Once defined the floor properties, the user has to define the characteristics of the roof; then selecting the tab “Roof” from the tab menu (Figure 83, element 1), the main window appears as in Figure 83.

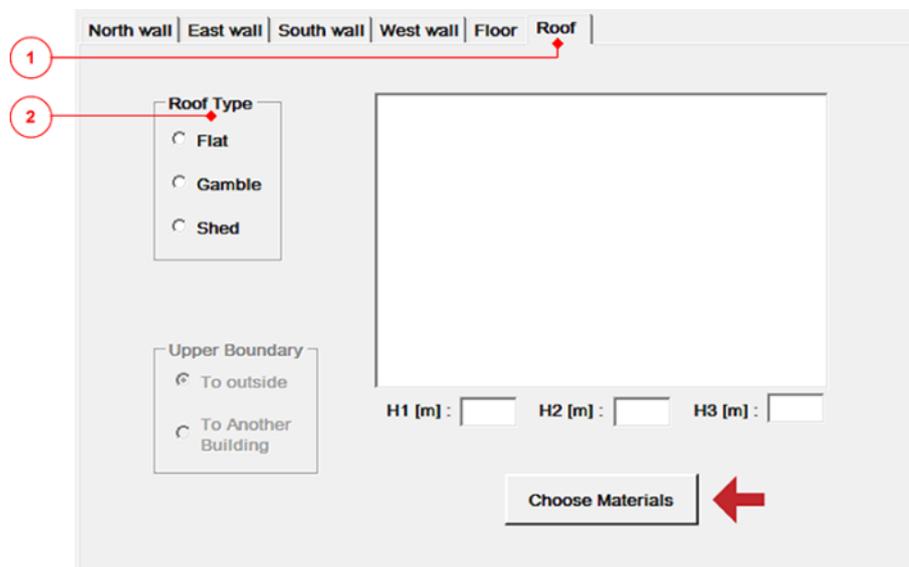


Figure 83 – Roof section

The user has to select the type of the roof of the environment of study between the available ones; the options are situated into the “Roof type” panel (Figure 83, element 2); depending on the type of the roof, the user has to insert different data; let us separate the approach based on the type of roof.

Flat roof

If the roof of the environment of study is a flat one, the user has to select the “Flat” radio button inside the “Roof Type” panel (Figure 84, element 1); the relative drawing of the roof appears inside the dedicated box (Figure 84, element 3).

Then the user has to define: if the roof is actually a roof, so it is exposed to the exterior, or it is a ceiling that divides the room from the upper floor; in case of first option, the user has to select “To outside” from the “Upper Boundary” panel (Figure 84, element 2); in case of second option, the user has to select the “To another Building” inside the “Upper Boundary” panel (Figure 84, element 2).

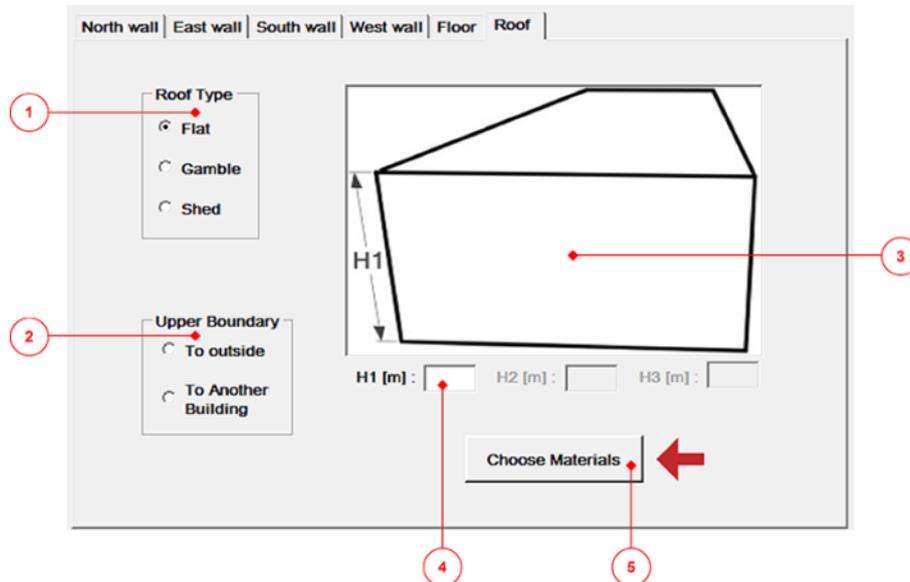


Figure 84 – Roof section after the flat roof selection

After that, using the text box “H1” (Figure 84, element 4), the user can insert the height of the environment of study; then the button “Choose Materials”(Figure 84, element 5) allows the definition of the roof stratigraphy through the same procedure previously described for the external wall; note that in case the roof is a ceiling between two floors, the user can skip the definition of the stratigraphy of it.

Gamble roof

In case the type of the roof is gamble, the procedure to insert the inputs is similar to the described on the flat section and differ only because: first the “Upper Boundary” panel (Figure 84, element 2) is disabled, second the user has also add the values ”H2”, ”H3” as features of the roof.

Shed roof

The last available option for the type of roof is Shed; the procedure to insert the inputs is the same of the one adopted for the Gamble roof, the only difference is that the user has to insert only the values of “H1” and “H3” without the “H2” value.

After the implementation of the features of the Roof, the user has complete the definition of the characteristics of the environment envelope. The next button in the command line validates the data, if the inputs are correct, the SOU will open a window to notify it and then it will move to the next section “Plant”.

4.2.3 Plant

This section is totally dedicated to the heating and cooling plant of the environment; the main window is organized as in Figure 85, the space is split into six main groups: heating, cooling, humidification, dehumidification, distribution (Figure 85, elements from 1 to 6).

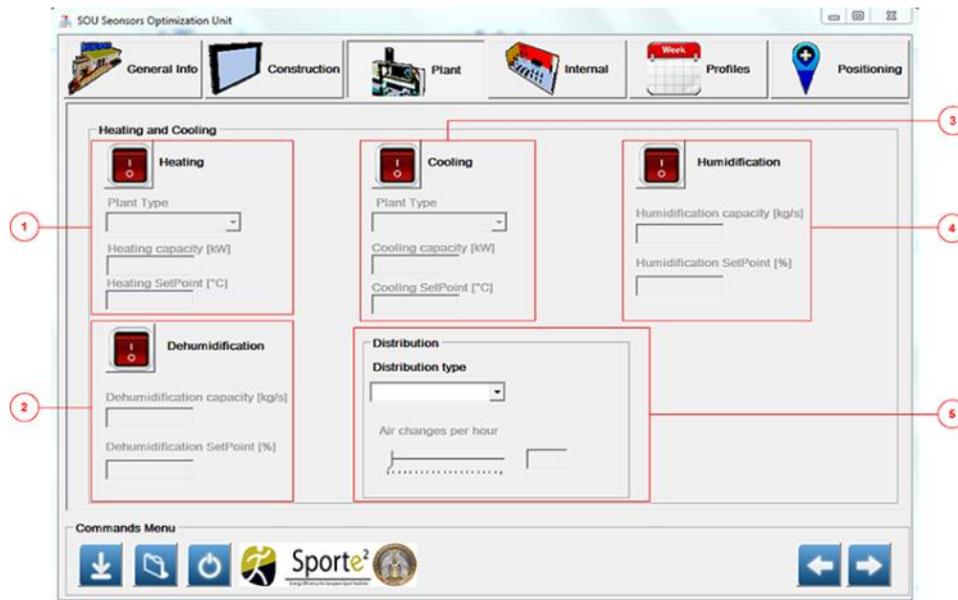


Figure 85 – SOU plant section

First of all the user should check the features of the plant, including typology (heating/cooling), capacity and distribution type; the user has also to check the capabilities of the heating/conditioning control system that serves the environment. The following part of this paragraph is dedicated to the data implementation.

Heating

1. Press the button “Heating” (Figure 86, element 1), in order to enable the possibility to insert the heating features;

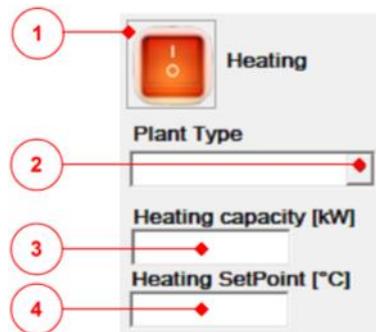


Figure 86 – Heating section of the Tab “Plant”

2. Select from the list box “Plant Type” (Figure 86, element 2) the available typologies of the heating plant for the environment:
 - Gas boiler;
 - Biomass boiler;
 - CHP (Combined Heat and Power plant);
 - Mixed (Gas boiler and Biomass boiler, Gas boiler and sun panels and so on);
3. Heating capacity [kW] (Figure 86, element 3): the maximum heating capacity that the heating system can provide to the zone considered; the user can find this value on the datasheet of the machine, in case the heating system serves only the ambient of study. If the same machine is providing heating for more zones, the user should estimate the heating capacity that is dedicated only to the zone considered; if it is unknown, the

value ‘-2’ can be used, then the SOU will estimate a reasonable value of the maximum heating capacity; the use of this last methodology should be avoided, because it is very rough and should be adopted only in case of absolute impossibility to get the information;

4. Heating SetPoint [°C] (Figure 86, element 4): is the desired air temperature value that the control system will aim to reach inside the environment.

Cooling

1. Press the button “Cooling” (Figure 87, element 1), in order to enable the possibility to insert the cooling features;

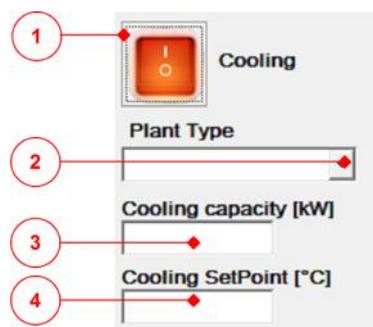


Figure 87 – Cooling section of the Tab “Plant”

2. Select from the list box “Plant Type” (Figure 87, element 2) the available typologies of the heating plant for the environment:
 - Chiller;
 - Heat pump;
 - Split;
5. Cooling capacity [kW] (Figure 87, element 3): the maximum cooling capacity that the cooling system can provide to the zone considered; the user can find this value on the datasheet of the machine, in case the cooling system serves only the zone considered. If the same machine is providing cooling for more zones, the user should estimate the cooling capacity that is dedicated only to the zone considered. If it is unknown, the value ‘-2’ can be used, then the SOU will estimate a reasonable value of the maximum cooling capacity; the use of this last methodology should be avoided, because it is very rough and should be adopted only in case of absolute impossibility to get the information;
3. Cooling SetPoint [°C] (Figure 87, element 4): is the desired the air temperature value that the control system will aim to reach inside the environment.

Humidification

1. Press the button “Humidification” (Figure 88, element 1), in order to enable the possibility to insert the humidification capabilities of the plant; please make sure that the plant has this capability of control on the air humidification of the ambient of study;

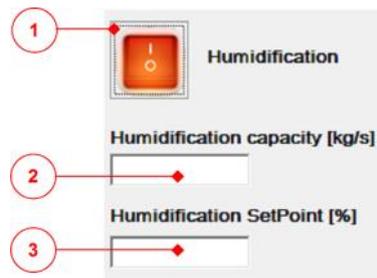


Figure 88 – Humidification section of the Tab “Plant”

- Humidification capacity [kg/s] (Figure 88, element 2): the maximum humidification capacity that the plant can provide to the zone considered; the user can find this value on the datasheet of the machine, in case of the plant is specifically dedicated to the ambient of study. If the same machine is providing humidification for more zones, the user should estimate the humidification capacity that is dedicated only to the zone considered. If it is unknown, it is strongly recommended to live this feature disable;
- Humidification SetPoint [%] (Figure 88, element 3): is the desired the percentage of air humidity that the control system will aim to reach inside the environment.

Dehumidification

- Press the button “Dehumidification” (Figure 89, element 1), in order to enable the possibility to insert the dehumidification capabilities of the plant; please make sure that the plant has this capability of control on the air dehumidification of the ambient of study;

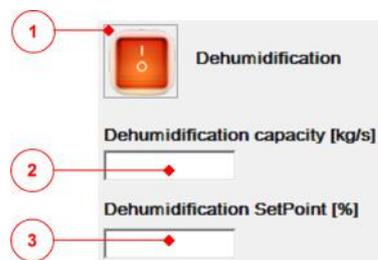


Figure 89 – Dehumidification section of the Tab “Plant”

- Dehumidification capacity [kg/s] (Figure 89, element 2): the maximum dehumidification capacity that the plant can provide to the zone considered; the user can find this value on the datasheet of the machine, in case of the plant is specifically dedicated to the ambient of study. If the same machine is providing dehumidification for more zones, the user should estimate the dehumidification capacity that is dedicated only to the zone considered. If it is unknown, it is strongly recommended to live this feature disable;
- DeHumidification SetPoint [%] (Figure 89, element 3): is the desired the percentage of air humidity that the control system will aim to reach inside the environment.

Distribution

The distribution consists on the way that the plant adopt to provide heated or cooled air inside the ambient of study; then using the specific list box (Figure 90, element), the user is able to select the distribution system between: HVAC (Heating, Ventilation and Air Conditioning), Radiators, Floor heating/cooling system, mixed (a mix between the previous ones).

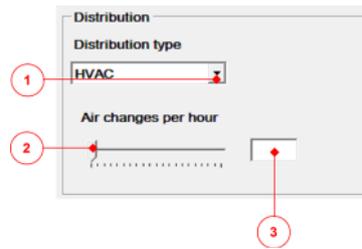


Figure 90 – Distribution section of the Tab “Plant”

Then the user can set the value of Air changes per hour (number of volumes of the environment per hour) through a specific track bar (Figure 90, element 2) that write the value inside the dedicated text box (Figure 90, element 3). If this value is unknown, the user should have a look of local building codes that stipulate the amount of ventilation required for Sports environments.

When all the data have been implemented, just move to the command line and validate the inputs, pushing the specific button (Figure 65, element 10) , if the inputs are correct, the SOU will open a window to notify it and move to the next section “Internal”.

4.2.4 Internal

This section is dedicated to the internal thermal loads (e.g. occupants doing sport, spectators, equipment). The window is organized as in Figure 91, the space is split into two main spaces: a 3D view of the room and a table to input loads properties.

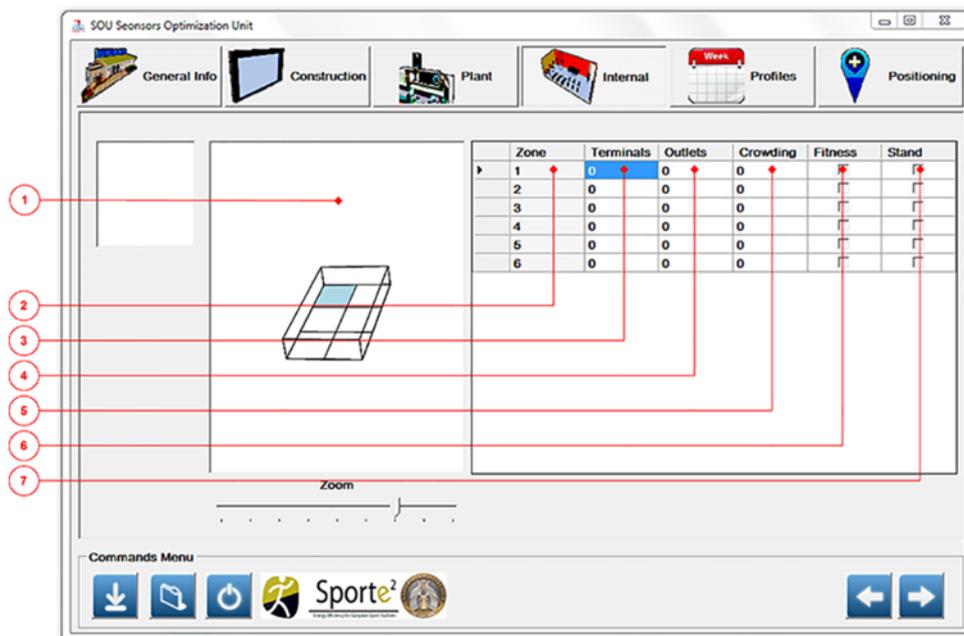


Figure 91 – Tab “Internal” of the SOU interface

The environment is split into N sub-zones (Figure 91, element 1) and the table has a number of rows equal to the number of the sub-zones; each row of the table, will define the features of the internal load inside that specific sub-zone.

The drawing has been done, keeping the orientation of the two dimensional drawing presented in Figure 91.

The table has six columns, each of them is described below:

- Zone (Figure 91, element 2) represents the current zone where the user is defining the specific internal load;

- Terminals (Figure 91, element 3) is the number of supply air inlets that is present inside a specific zone;
- Outlets (Figure 91, element 4) is the number of return air outlets that is present inside a specific zone;
- Crowding (Figure 91, element 5) is the percentage of people usually occupying the sub-zone; the sum of all the crowding values must be 100%. As an example, a gym where the space is divided in: one part where usually there are just machine and the rest of the gym is dedicated to fitness course; then inside this part of the gym usually are present more people than into the first one; so the user should try to express the crowding of each zone in percentage;
- Fitness (Figure 91, element 6): check this check box if there is sport activity in that zone;
- Stand (Figure 91, element 7): check this check box in case of terraces with spectators.

When all the data have been implemented by the user, just move to the command line and validate the inputs, pushing the specific button (Figure 91, element 10); if the inputs are correct, the SOU will open a new window to notify it, then automatically it moves to the next section “Profiles”.

4.2.5 Profiles

This section is dedicated to the definition of the profile of use. The main window (Figure 92) is divided into two main parts called “Profile information” (Figure 92, element 1) and “Weekly assignment” (Figure 92, element 2).

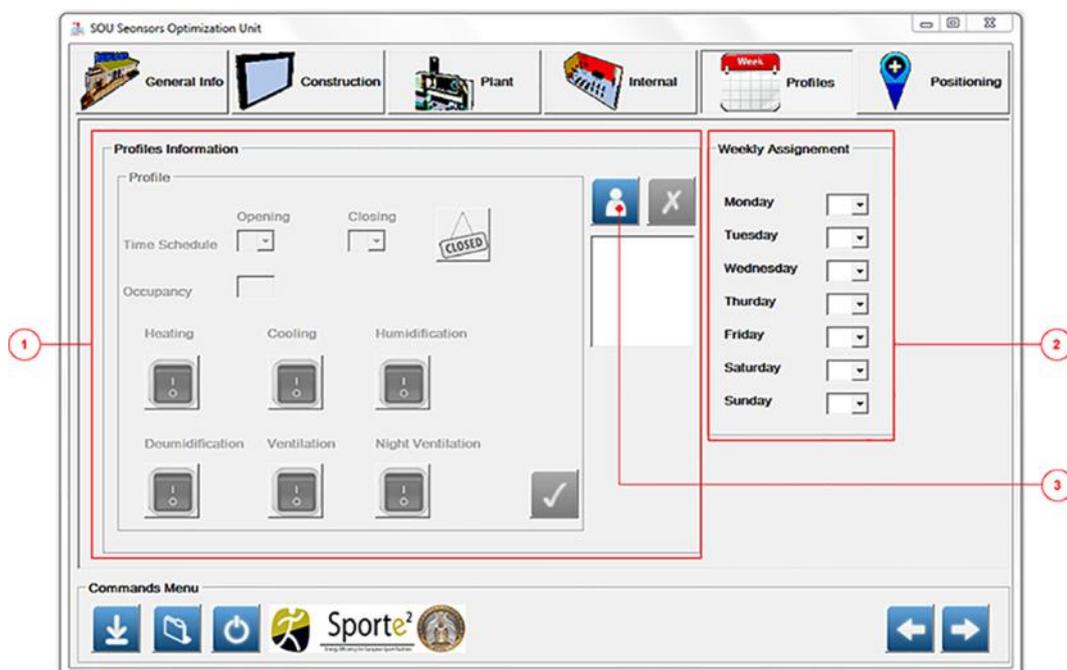


Figure 92 Tab “Profile” of the SOU interface

The user needs to insert the inputs from the panel “Profile information” and, as shown in Figure 92, the panel is disabled at this stage; then the next steps are following reported:

1. Push the button (Figure 92, element 3), it enables the panel (Figure 93), after that the SOU is ready to create the first profile of use;

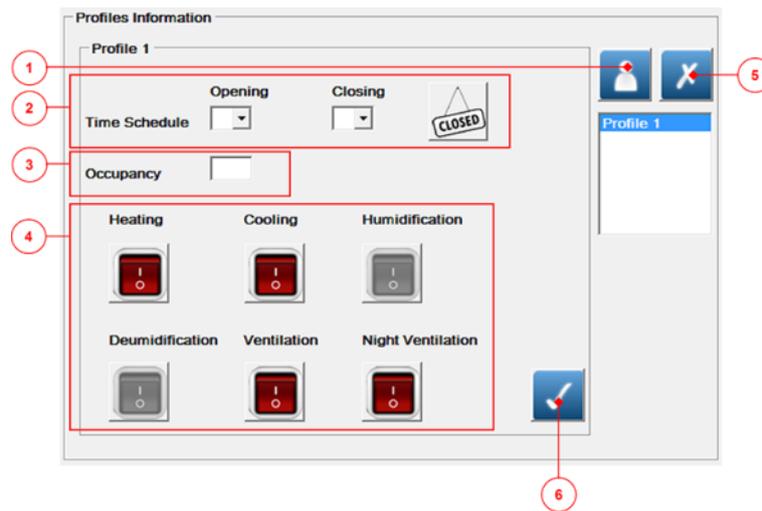


Figure 93 “Profile” panel of the “Profile” Tab of the SOU interface

2. Define the “Time Schedule” using the two list boxes (Figure 93, element 2): using the “Opening” and “Closing” list boxes, the user can define the opening and closing hours. Each profile covers one day period; the user cannot create two profile to use on the same day. If the user wants to create a day closed, just press the button “Close” (Figure 93, element 2);
3. Define a rough estimation of the possible occupancy value (number of people) for a specific profile of use;
4. Using the switches in Figure 93 element 4, the user can define the plant usage related to that profile; note as in Figure 93 not all the switches are enabled, this because they are enabled or not based on the settings made inside the “Plant” section.
5. Then press the button in Figure 93, element 6, in order to save the Profile; this operation is fundamental in order to assign the profile during the next steps;
6. Following the previous steps, the user can create maximum 7 profiles, one per each day of the week; use the button in Figure 93, element 5, in case the user needs to delete a Profile previously made;

Then the user should move to the other part of the window: “Weekly assignment” reported in Figure 94.

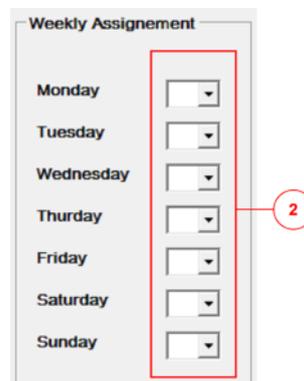


Figure 94 – “Weekly Assignment” panel of the Tab “Profile” of the SOU interface

The user has just to use the list boxes (Figure 94, element 2) and for each day of the week choose the specific profile of use for the environment.

When the user has completed the implementation of all the inputs, just press the validation button (Figure 94, element), then if all the data implementation process has been done in a proper way, the message “Data validation complete” appears; it means that SOU is ready to calculate the solution.

4.2.6 Positioning

The tab “Positioning” represents the final step that brings the user to the solution for the sensors positioning. The main window is organized into two main areas: a graphical box (Figure 95, element 1) and panels for data inputs (Figure 95, element 2).

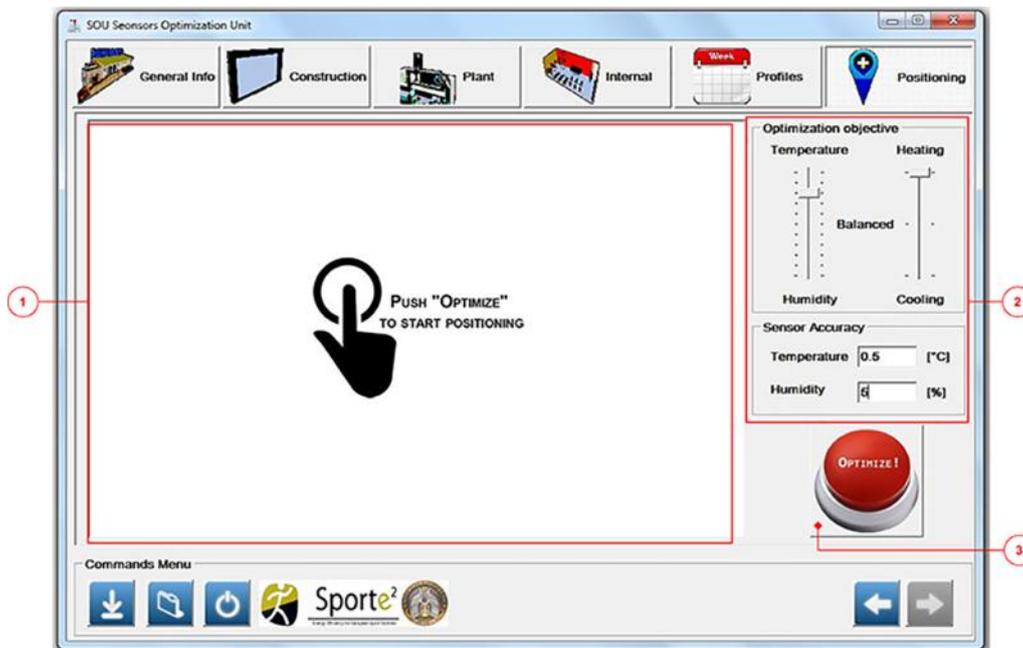


Figure 95 – Tab “Positioning” of the SOU interface

Before starting the calculus of the optimal solution, the user has to insert additional parameters using the panel “optimization objective” (Figure 95, element 2), where there are two track bars:

1. The track bar on the left side of the panel is related to the weighting factors that influence the optimization process. The track bar (Figure 96, element 1) allows 10 different positions. In the case that a greater accuracy of the air temperature measurement is required, the bar should be moved toward the “Temperature” side. The optimization process is thus configured to reach the maximum accuracy of the temperature measure. In the case that a greater accuracy of the relative humidity measurement is required, the bar should be moved toward the “Humidity” side. In the case of a balanced optimization between Temperature and Relative Humidity, the slider of the bar should be leaved in the position “Balanced”: the optimization weights are equally divided by 50% in each side. The selection of these weights defines the balance point to be used by the optimization procedure. For this reason, the selection should be performed paying attention to the end-use of the measuring system (e.g. simple comfort evaluation, feedback to HVAC control system etc.).

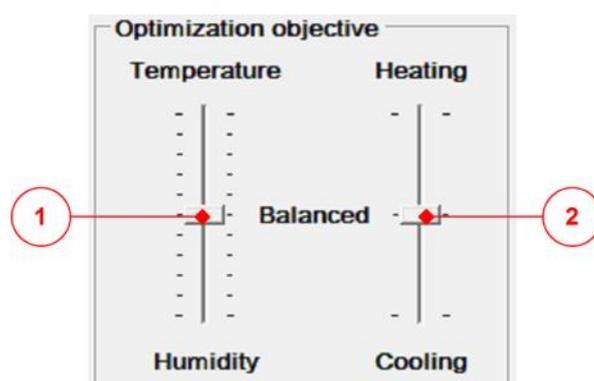


Figure 96 – Optimization objective options of the SOU

2. The track bar on the right side is related to the type of plant and the profile of use of the environment; the slider (Figure 96, element 2) has three positioning options:
 - Heating: if only heating is provided, the optimization considers only the heating period of the year;
 - Cooling: if only cooling is provided, the optimization considers only the cooling period of the year;
 - Balanced: if heating and cooling are provided, the optimization considers the period of the entire year; if the comfort monitoring for the entire year is required, the air T/RH values are fundamentals parameters for the calculus of comfort indicator as PMV value.

Based on the type of the sensors utilized inside the environment, the user has to fill the two text boxes “Temperature” and “Humidity” inside the panel “Sensor accuracy” (Figure 95, element 2). The first text box should be filled with the value of the accuracy of the temperature sensor expressed in [°C], the second text box should be filled using the value of the accuracy of the relative humidity expressed in [%]. The sensor data sheet usually provides the accuracies.

Let us make an example to better clarify the definition of the optimization parameters. The environment of study is a gym and it has only a heating system. The control system is based just on the temperature value and it works just using the temperature value coming from the thermostat that measures also the relative humidity inside the environment. The owner of the facility wants to know also the comfort level that there is inside the room during the entire year. Therefore, the optimization parameters should be set as: the track bar on the left side of the panel has to be positioned on Temperature side, the other track bar has to be set in balanced position.

After the definition of all the optimization parameters, the SOU is ready to run the optimization process, then the user should just press the red button (Figure 95, element 3) and wait until the solution will appears on the dedicated graphic box (Figure 95, element 1).

The SOU produces as output a 3D draw (Figure 97) of the optimal solution for the sensors positioning inside the environment.

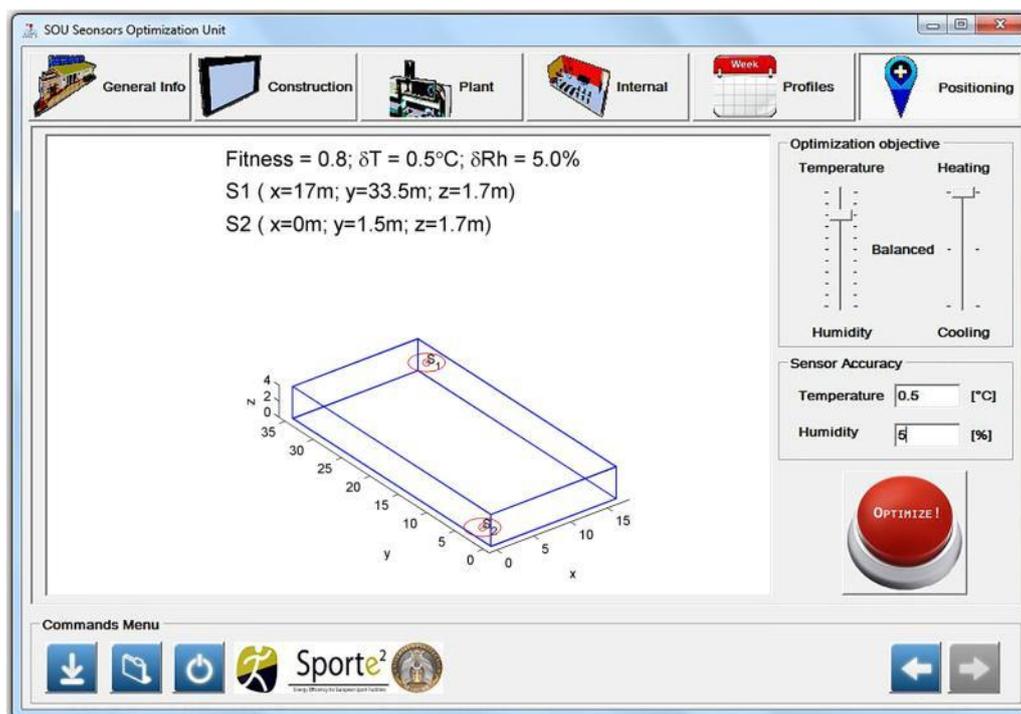


Figure 97 – Output window of the SOU

The SOU output (Figure 97) contains:

- δT e δRH that represent the values of measurement uncertainties obtained with the solution provided. These values include both the uncertainties due to the sensors and to the location;
- the coordinates of installation for the sensor/sensors, expressed in [m] using a Cartesian reference system, included into the solution drawing. The orientation of the drawing follows the one defined on the two dimensional scheme reported in Figure 69.

The sensor/sensors should be located at 1.7 [m] from the floor, considered as typical thermostat height, near the occupied zone [3].

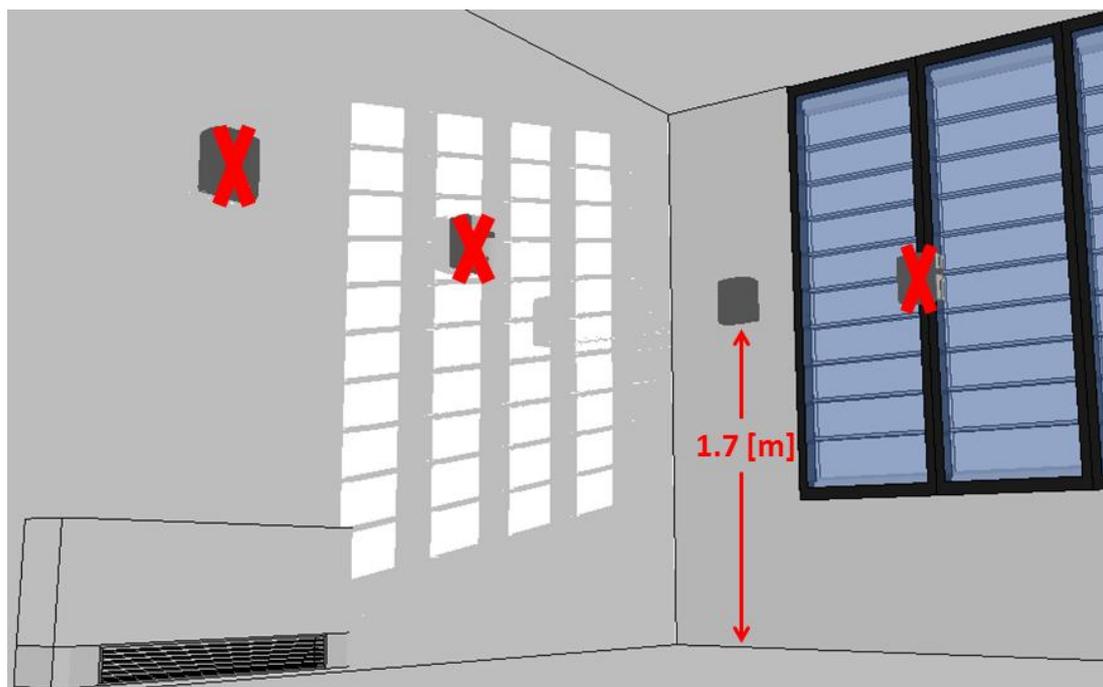


Figure 98 – Sensor location guidelines

The sensor should not be installed where:

- there are unusual heating conditions, such as: direct sunlight, near lamps, radio, TV screens, near hot water pipes in a wall;
- there are unusual cooling conditions, such as: in a draft from a stairwell, door, or window;
- air circulation is poor, such as: in a corner, an alcove, behind an open door (Figure 98).

5. CONCLUSIONS

This Deliverable closes the pathway to develop a Smart Metering system for sports facilities performed during the SportE2 project. The main scope of this document is to present the final version of the system that is the How Module. Retracing the pathway, the process began with the definition of the requirements for the smart metering of sports facilities. Moving from the state of the art, the design phase addressed the specific needs to clearly identify the behavior of these buildings and **quantify the energy profiles in relation with the user comfort**. A dedicated approach was thus defined and presented in the Deliverable D2.1. Based on this approach the first prototype was then developed in the laboratory building Kubik and presented in the Deliverable D2.2. The experience of the prototype gave the possibility to finalize the development and transfer the concept into a real application in the pilot cases. Facing the real buildings, the Module was refined in order to encounter the needs of facility managers, users and local regulations. This fine-tuning turned out to show the first version of the How Module in three sports facilities as described in the Deliverable D2.3. The implementation process provided the possibility to create a protocol for the design, install and configure the How Module.

The lessons learnt with the experience are reflected in this document, which presents a complete description of the functionalities and the approach to design a smart metering system for sports facilities. These contents are not only a mere description of the Module, but also a powerful guideline for the smart metering in sports facilities.

The main outcomes and tools from the whole research are:

- A specific approach to design the smart metering system as stand-alone or integrated tool;
- A specific methodology to assess the comfort in sports areas;
- A methodology to process metered data and produce useful information about the use of the energy in sports facilities;
- A powerful graphical tool to calculate and provide previous information to the facility manager/owner;
- An optimization tool to address the specific problem of monitoring the indoor conditions of sports areas.

This report describes how these tools work and includes the data sheet of the complete How Module to support the exploitation and the replication of the developed technology in the market.

6. ACKNOWLEDGEMENTS

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. FP7-2010-NMP-ENV-ENERGY-ICT-EeB 260124.

7. REFERENCES

- [1] G.M. Revel and M. Arnesano, *Perception of the thermal environment in sports facilities through subjective approach*, *Building and Environment* 77 (2014) 12–19
- [2] ASHRAE, *ASHRAE Handbook - Heating, Ventilating, and Air-Conditioning Applications (I-P Edition)* (2011)
- [3] D. Wang, E. Arens, T. Webster and M. Shi, *How the number and placement of sensors controlling room air distribution systems affect energy use and comfort*, *Proceedings of the 2th International Conference for Enhanced Building Operations*, Richardson, Texas (2002)

8. APPENDIX A1: COMPATIBLE DEVICES

Following there is the complete list of all the devices compatible with the How Module and their technical characteristics. These devices have been already used and tested with the How Module in the three pilot cases of the project. However, the list is not restrictive since other devices can be used to substitute the ones included in the list or to include additional measurements and functionalities.

iMeter

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Energy meter • Direct connection up to 80A • Power input: 220-240 VAC • Global counter with tariffs • LCD display • Communication in Modbus • Adjustable communication interval • IP50 protection certification • Dimensions (WxDxH): 122x90x60
---	---

Hydrocal G21

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Heat meter • Battery powered (typical autonomy of 10 yr) • Working temperature: 5 to 55 °C • IP54 protection • 2,5 m³/h of nominal flow • Admitted flow: 0,05 to 5,0 m³/h • Nominal pressure: 16 bar • Output: light pulse • Temperature sensor: PT1000
---	---

Multical 602 & ultraflow

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Heat meter • Temperature range θ: 2°C...180°C • Differential range $\Delta\theta$: 3 K...170 K • Accuracy $E_c \pm (0.5 + \Delta\theta_{\min}/\Delta\theta)\%$; • Temperature sensors: Pt500 EN 60 751, 2-wire connection; • Flow sensor type: ULTRAFLOW (it is a static flow sensor based on the ultrasonic measuring principle); • Supply voltage: 230 VAC or 24 VAC; • Battery: 3.65 VDC, D-cell lithium;
---	---

Hydrosplit

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Separated heat meter version: is an heat meter in a separate version to be installed on sites where the compact version can not be applied; it can be easily connected to any water meter equipped with pulse output. • Battery powered (typical autonomy of 10 year) • Working temperature: 5 to 55 °C • IP54 protection • 2,5 m3/h of nominal flow • Admitted flow: 0,05 to 5,0 m3/h • Nominal pressure: 16 bar • Output: light pulse • Temperature sensor: PT1000
---	--

Itron MSD

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • The water meter equipped with communicating interface modules (pulse emitted device) for data transmission; • Water meter DN 50 [mm]; • Pulse device: <ul style="list-style-type: none"> ○ Internal power supply: lithium battery;
--	--

WDE-K30-R

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • The water meters equipped with pulse emitter device can be linked to tele-reading systems, to PLC, to M-Bus networks, to impulse counters and to all those applications that need an electronic input which indicates the water consumption Battery . • DN 100 [mm]; • Nominal flow rate: 60 [m3/h]; • Length: 250 [mm];
---	---

GMB-RP-R

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • The water meters equipped with pulse emitter device can be linked to tele-reading systems, to PLC, to M-Bus networks, to impulse counters and to all those applications that need an electronic input which indicates the water consumption Battery . • DN 40 [mm]; • Nominal flow rate: 10 [m³/h]; • Length: 300[mm];
---	--

SPD310-100/1000 Pa

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • SPD 310 differential pressure transmitter is intended for use in air handling systems for the monitoring of air ducts, filters and fans. It converts the differential pressure measured into an electric 0-10 V signal. • Output: 0-10 V; • Ranges: 0-1000 Pa; • Accuracy at ambient temperature of 25 °C: ≤ ±0.4%FS • Supply: 24 Vac/ 15-32 Vdc
--	--

STD 100-200

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • The STD100 temperature sensor is intended for air duct mounting. A plastic duct mounting bracket is supplied with a securing screw; • Sensor element: NTC, 1 .8 kΩ at +25 °C (77 °F); • Accuracy: <ul style="list-style-type: none"> ○ ± 0.5 °C at 0 °C; ○ ± 0.3 °C at 25 °C; ○ ± 0.6 °C at 50 °C; ○ ± 0.9 °C at 75 °C;
---	--

STR-100

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • The STR range of room temperature sensors comprises a series of wall modules optimized for public facilities • Sensor element: NTC thermistor; • Range: 0 to 50 °C; • Accuracy: <ul style="list-style-type: none"> ○ ± 0.5 °C at 0 °C; ○ ± 0.3 °C at 25 °C; ○ ± 0.6 °C at 50 °C; ○ ± 0.9 °C at 75 °C;
---	---

SHR 100

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • The SHR100 temperature sensor is intended for air duct mounting. A plastic duct mounting bracket is supplied with a securing screw; • Sensor element: NTC, 1.8 kΩ at +25 °C (77 °F); • Accuracy: <ul style="list-style-type: none"> ○ ± 0.5 °C at 0 °C; ○ ± 0.3 °C at 25 °C; ○ ± 0.6 °C at 50 °C; ○ ± 0.9 °C at 75 °C;
---	---

SHO 100-T

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • The SHO100 is an active sensor, which measures relative humidity (%RH) and converts the measurement into an electric current 4-20 mA or a voltage level 0-10 V. It is intended for outdoor installation and for indoor areas where a more robust design is needed e.g. warehouse, swimming pool. • SHO100-T includes selectable temperature sensors NTC 1.8 kΩ and NTC 10 kΩ for I/Net products. • Output: Selectable 4-20 mA, 0-10 V; • Range: 0-95% RH; • Accuracy: ±2%; • Supply: 24 Vac / 15-36 Vdc Power;
---	---

CO&CO2

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • This device measures both carbon monoxide and carbon dioxide and therefore is ideal for measuring air quality for health purposes. • Output 0-10 V / 0-5V selectable; • Range 0-2000 ppm; • Accuracy $\pm 1\%$ of measurement range, $\pm 5\%$ of measured value; • Supply 24 Vac;
---	--

STP 100

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • The STP100 is designed for immersion mounting in pipe systems with a separate pocket (well).. • Sensor element: NTC, 1 .8 kΩ at +25 °C (77 °F); • Accuracy: <ul style="list-style-type: none"> ○ ± 0.5 °C at 0 °C; ○ ± 0.3 °C at 25 °C; ○ ± 0.6 °C at 50 °C; ○ ± 0.9 °C at 75 °C;
--	--

Net meter

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Radio Frequency: 868MHz • Transmitting Range: 30 – 300 meters • Inputs: 2 digital inputs • Maximum impulse freq.: 10 Hz • Meter interface: reed switch or pulse reader • Adjustable communication interval • IP54 protection certification • Powered by lithium batteries • Operation temperature: -10°C to 55°C
---	--

iPoint

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Temperature measurement range: -20°C to 55°C • Typical temperature resolution: 0.01°C • Typical temperature precision: ±0.4°C • Humidity measurement range: 0%RH to 100%RH • Typical humidity resolution: 0.05%RH • Typical humidity precision: ±3%RH • Communication interval: 15 minutes • Radio frequency: 868 MHz • Transmitting range: 30 – 300 m • Typical autonomy: 1 year
---	--

iPoint - insulated

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Temperature measurement range: -20°C to 55°C • Typical temperature resolution: 0.01°C • Typical temperature precision: ±0.4°C • Humidity measurement range: 0%RH to 100%RH • Typical humidity resolution: 0.05%RH • Typical humidity precision: ±3%RH • Communication interval: 15 minutes • Radio frequency: 868 MHz • Transmitting range: 30 – 300 m • Typical autonomy: 1 year • For outdoor and humid environments
---	--

iPoint CO2

	<p>General characteristics:</p> <ul style="list-style-type: none"> • Power input: 85 to 264 VAC, 50/60 Hz, 16 mA • Transmission frequency: 868MHz • Transmission range: 30 to 300 m • Operating temperature: 0°C to 50°C • Sensing Method: Non-dispersive infrared • Measurement range: 0 to 5000 ppm. • Measurement accuracy: ±30ppm, ±5% of measured value
---	---

Dulcomarin II

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Possible measured variables per circuit: pH, ORP/Redox, temperature, chlorine (free and total available chlorine), bromine, fluoride, ammonia, chlorine dioxide, chlorite, conductivity, UV intensity, flow and turbidity • Output: 4 analog outputs (4 – 20mA) • Electrical connection: 85...265 VAC, 50/60 Hz • IP65 protection
---	--

Cedes TPS 210

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Object/occupancy detector • Infrared beam based (2 infrared beams) • Direction recognition of objects <ul style="list-style-type: none"> ○ Min. 20cm wide objects ○ Max. 5m/s velocity • Max ambient light: 100.000 lux • IP65 Protection
---	--

NPK-43-20

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Level probes for liquids: wire submersible transmitters used for the hydrostatic measurement of levels in liquids; • Accuracy: on full scale <0.5%; • Operating temperature: -10~60°C; • Output: 4~20mA, 2 wire, loop powered.
---	--

Termokon

	<p>General Characteristics:</p> <ul style="list-style-type: none"> • Sensor for light measurement in rooms and offices; • Accuracy: Typ. 5% of measuring range; • Ambient temperature: -20 ... +70°C • Operating voltage: • Power consumption: Typ. 0,5W / 2VA • Measuring range : <ul style="list-style-type: none"> ○ 1: max. 2.000 Lux ○ 2: max. 20.000 Lux ○ 3: max. 65.535 Lux • Operating voltage 24V;
---	---