SportE^2 - Intelligent Management System to integrate and control energy generation, consumption and exchange for European Sport and Recreation Buildings
Start: 1 September 2010
Duration: 48 months
FP7 ICT-EEB Grant Agreement 260124

D5.3 SportE^2 Where available as a Product Module

Date	Delivery date: M48
Deliverable Number	D5.3
WP Number	WP5 - Integration & Testing
Version	3.0
Status	Final
Dissemination level (*)	PP

Abstract

SportE^2 Where available as a Product Module: Report and product data sheets for SportE^2 Where module

Versions

Date	*Partner*	*Content*	*Ver.*
20-01-2014	DAPP	First version of table of content	0.0
12-09-2014	DAPP	First draft complete	1.0
19-09-2014	UNIVPM	Quality check	1.1
30-09-2014	DAPP	Final version	2.0
27-01-2014	DAPP	Addition of the product data-sheet and introduction reviewed	3.0

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

The report constitutes the data sheet of SportE^2 WHERE Module, which is used by the facility managers for performing comparison of energy performances of different sport facilities they manages. As a data sheet, the document includes the technical information needed to design, develop, install and use the module.

The module is composed of various components and its primary focus is to enable benchmarking of energy behaviour between multiple sport facilities.

With the SportE^2 WHERE Module the sport facility manager has the possibility to:

- compare energy performances of different sport facilities that he manages;
- analyse primary and secondary Key Performance Indicators (KPI) (e.g. total energy consumption, total gas consumption, etc.) for the different facilities;
- monitor the actual situation of energy demands, consumptions, production, and grid exchange across multiple facilities (e.g. [kWh/m^2] or [TEP/m^2], [€/m^2], [kg of CO_2/m^2]) from a remote web browser. Data is acquired from the SportE^2 How Module, elaborated and provided to the users and then *re-published* through a web service interface for other usages (e.g., elaboration by external application such as legacy systems).

The SportE^2 WHERE Module is currently deployed in D'Appolonia Server Farm and accessible at the URL http://sporte2.dappolonia-innovation.com/. For the future exploitation of project results, to be implemented after its conclusion, the platform could be easily migrated to a Cloud system.

1. INTRODUCTION

This document describes the SPORTE^2 Where Module and presents in annex the data sheet of the module as a product ready for market deployment.

The SportE^2 WHERE Module comprises a single component called benchmarking unit. This JavaEE software module deployed on Oracle Glassfish application server represents a web application accessing the information of each managed sport facility to monitor the actual situation of energy demands, consumptions, and production. The developed Graphical User Interface (GUI) offers benchmarking capabilities by means of comparison energy situations and measures from different monitored and controlled facilities.

The main objectives and functionalities illustrated in Deliverable D5.2 are here consolidated and as a final product after a pathway starting from the conceptualization to ending to the final release deployed and remotely accessible from D'Appolonia server farm. These consolidated functionalities are:

- **Benchmark Monitor**: permit to monitor one benchmark at a time
- **Dashboard**: permit to monitor different type of benchmark and indicator at the same time
- **Tariff Management**: permit to edit and maintain tariffs for the facility in order to calculate energy consumption and costs
- **Formula Management**: permit to edit and maintain the formulas used to retrieve and calculate benchmark indicators
- **Background Synchronization**: synchronize the database and the benchmark values with sources available from the How Module
- **Administrative Interfaces**: multiple interfaces that permit to access edit and configure all the information used by the Where Module.

This deliverable contains in Chapter 2 a description of the module architecture and the integration with the other SportE^2 modules. The communication interface and the protocols are described in detail, thus providing a ground for comprehension of the features implemented.

Additionally, the functional analysis and the use cases are described and detailed in UML notation. Users and main actions are described and listed.

The development and deployment architecture is described as well, providing an indication of the programming language used and the main libraries and standards adopted.

In Chapter 3 the main characteristics and functionalities of the SportE^2 Where Module as a product are described, altogether with user interfaces with pictures and descriptions of the functionalities implemented the login interface, the main page and the link to the various components of the module. Furthermore, Section 3.3 contains the SportE^2 Where Module User Manual.

The final assessment of the SportE^2 Where Module is presented in Chapter 4, while in Annex 1 is provided the product data sheet.

2. SPORTE² WHERE MODULE TECHNOLOGICAL OVERVIEW

2.1 SportE² Where Module integrated in the SportE² system

The SportE^2 Where Module comprises a single component called *Benchmarking Unit*. This software module (deployed externally to the facility) represents a web application accessing to the information of each sport facility managed, to monitor the actual situation of energy demands, consumptions, and eventually production. The developed GUI offers benchmarking capabilities by means of the comparison of energy situations and measures from the different monitored and controlled facilities. Software interface provides information to human operators in the remote Data Centre (Administrative Unit of the customer) in the form of:

- Charts and Tables enabling historical data evaluation;
- Comparison metrics;
- Energy Reports.

Input data (HQDS, High Quality Data Set) are retrieved by using the Web Service Interface of the Energy Data Acquisition and Processing Unit (*How* module) for energy consumption and generation benchmark, and by using the Web Service Interface of Why Module (Optimization module) for the forecast benchmark.

Weather and Energy Price forecasts are computed internally into the SportE^2 Where Module.

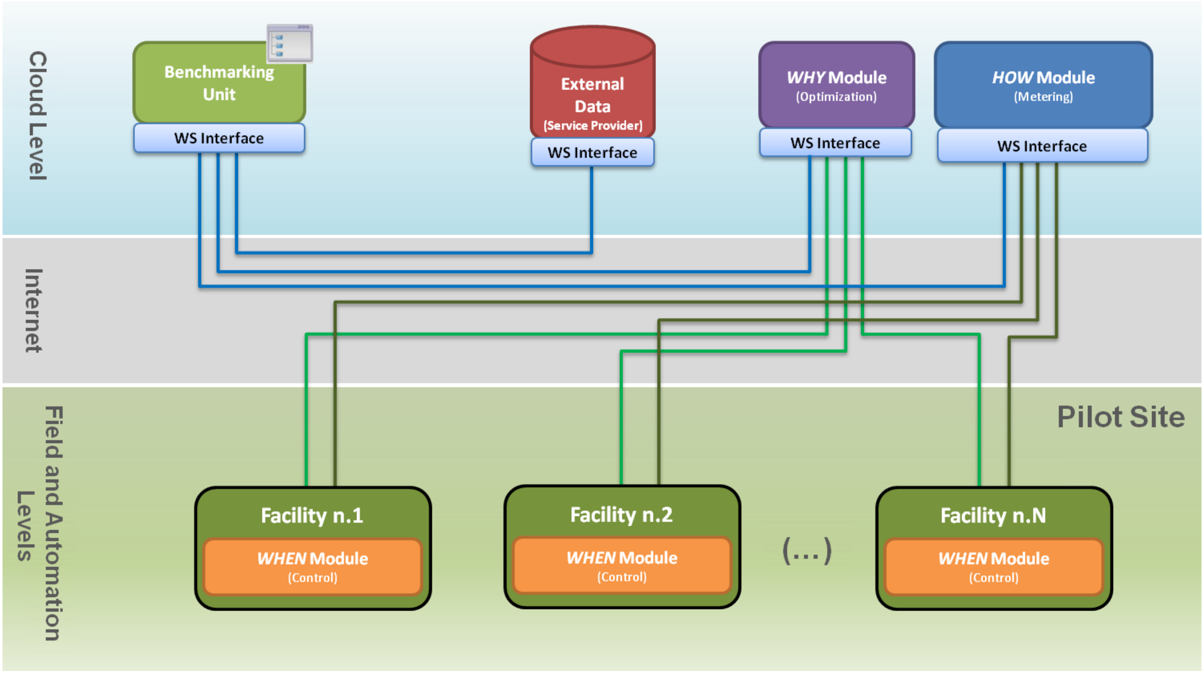


Figure 1: SportE^2 Where Benchmarking Unit and communication with the SportE^2 modules

Figure 1 provides the framework where the *Benchmarking Unit* is deployed: this unit does not communicate directly with the IT systems installed in the monitored facilities but only with the SportE^2 How and Why modules (Metering and Optimization, respectively). The communication is enabled by standard Web Service interfaces that ensure loose-coupling interaction and then high flexibility to the SportE^2 system as a whole. Finally, the SportE^2 Where Module retrieves information from external providers (external data) in order to provide Weather forecasts of each facility.

2.1.1 System Architecture

The SportE^2 Where Module comprises different components as for Figure 2:

- **Web Service (WS) Layer**
  This component has two sub-modules: WsInput and WsOutput.
  WsInput contains client code to access HOW and WHY web services. Through this layer, the *Data Unit* can collect data for each configured facility to produce benchmark index data.
  WsOutput contains web service code to deliver benchmark index data to external client.

- **Data Unit**
  This component is focused on data import and synchronization between HOW and WHY web service with local database. It contains a scheduled process that executes the synchronization work.

- **Controller Logic**
  This component is a bridge between WS Layer, Data Unit and User Interface. It takes care of the interaction with Data Unit to obtain benchmark data. It also takes care of data normalization.

- **User Interface**
  This component represents the interface (GUI, Graphical User Interface) enabling the facility operator to query and analyze benchmark index. The user interface is accessed by Web browser, since the *Where* module interface unit is represented by a Web application running on a machine acting as Web server.

- **Configuration Database**
  This component is a support element. The DB is a SQL database (Relational Database) containing several tables aiming to store configuration about facility, benchmark index and user permissions.

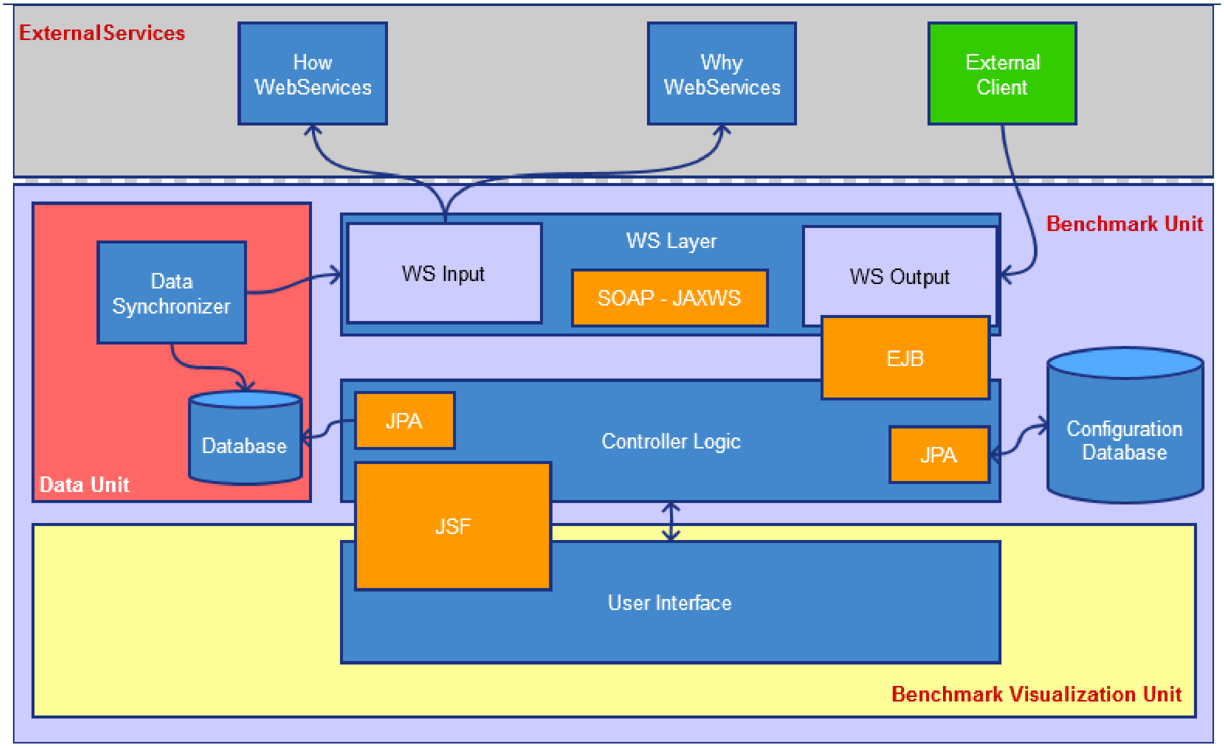


Figure 2: SportE^2 Where Module Application Layers

2.1.2 Communication Architecture

The SportE^2 Where Module interface (UI) can be accessed by common Internet browser software (i.e., Mozilla Firefox, MS Internet Explorer, etc.), thought HTTP connection. the WSLayer of the Benchmark Unit uses SOAP protocol to communicate with web services of the other SPORTE2 modules.

In the architecture as described, the following communication protocols and standards have been considered:

- ***HTTP*** – HTTP (HyperText Transfer Protocol) is an application-layer protocol whose specifications are managed by the W3C consortium. The HTTP protocol operates over TCP connections allowing the communication between a client and a server. It is a stateless protocol since each command message is executed independently by the server without knowledge of the previous commands.

- ***SOAP*** – SOAP (Simple Object Access Protocol) is a W3C standard for structured information interchange (based on XML standard for the message format) relying on HTTP (Hypertext Transfer Protocol) for message negotiation and transmission. A SOAP method (the exposed function, remotely called) is an HTTP request/response that complies with the SOAP encoding rules (HTTP+XML).

2.2 SportE^2 Where Module initial scope and objective

The SportE^2 Where Module has been developed taking into account the assessments made in the deliverable D1.1, in particular analysing the Performance Metrics (3.3 Performance metrics page 54).

The functionalities offered by the module regards the benchmarking of the performance metrics, useful to compare and catalogue the energy behaviours of the facilities.

Two types of indicators have been implemented for the benchmarking of the energy performances: primary and secondary.

Primary indicators are commonly used for buildings energy benchmarking. Secondary indicators are specific indicators defined for sport facilities. These represent the measure of energy consumption per functional area and of local energy production/RES performance that will be used as intelligent control parameters.

In the next paragraphs is provided a description of the functionalities developed into the Where module with a UML representation using Use Cases.

2.2.1 Introduction of the UML Use Cases

The Unified Modeling Language (UML) is a general-purpose modeling language in the field of software engineering, which is designed to provide a standard way to visualize the design of a system.

The Unified Modeling Language (UML) offers a way to visualize a system's architectural blueprints in a diagram (see image), including elements such as:

- Any activities (jobs);
- Individual components of the system;
- And how they can interact with other software components;
- How the system will run;
- How entities interact with others (components and interfaces);
- External user interface.

Although originally intended solely for object-oriented design documentation, the Unified Modeling Language (UML) has been extended to cover a larger set of design documentation (as listed above), and been found useful in many contexts.

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the various ways that they interact with the system. This type of diagram is typically used in conjunction with the textual use case and will often be accompanied by other types of diagrams as well.

A use case is a set of scenarios tied together by a common user goal. Each defined scenario is a sequence of steps describing an interaction between a user and a system. Current section defines a set of Use Case diagrams of SportE^2 Where Module, in which a use case diagram describes the functional behaviour of the system as seen by the user.

For each defined use case with related diagram(s), the involved actors are identified and a high level description is provided. An actor is any system, person or organization, which has an interest in, provides resources for, or is affected by the use case. An actor can take multiple roles.

A role is a set of activities and technical components, the responsibility of which is not divided between separate actors. A role can contain multiple technical components of technical architecture (technical implementation of a use case). A business interface presents a business relationship (e.g., contracts and transactions) between actors.

2.2.2 Actors in Use Case

The actors involved in the use cases were added to architectural domains. The table below provides the list of identified actors and their type.

Actor Code	Actor Name	Description
AC01	System	The scheduled job managing the process of data synchronization from sources
AC02	Facility Manager	User logged into the system that can monitor each configured facility and compute benchmark analysis
AC03	Web Service	Generic Source of data
AC04	Database	Database
AC05	Module How	Source of data
AC06	Module Why	Source of data
AC07	Google Maps	Map Rendering Service
AC08	Administrator	Administrator
AC09	Configuration Manager	User that can manager Tariff, Formula and other configurations

Table 1 - Actors

2.2.3 Use Cases

Use cases, describing the functional behaviour of the system, as seen by the user, are listed in the table below.

Use Case Code	Use Case Name
UC1	Data Synchronization
UC2	Benchmark management
UC3	Dashboard management
UC4	Administration
UC5	Formula Management
UC6	Tariff Management

Table 2 – Use Cases

In the following paragraphs, details of the single single use cases are described.

2.2.3.1 UC1 - Data Synchronization

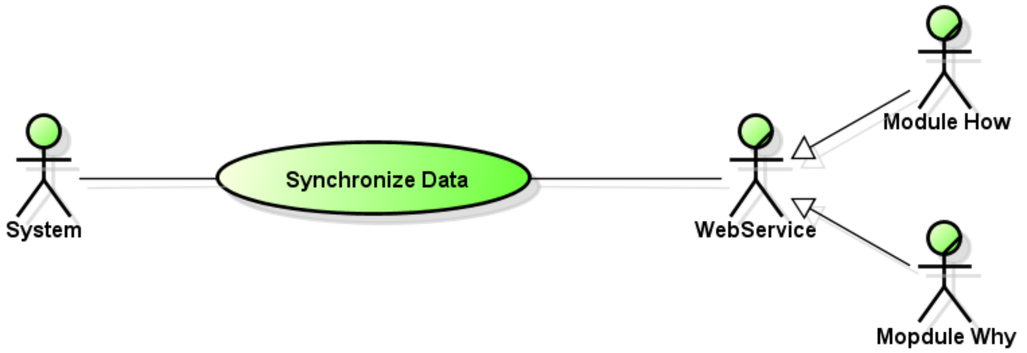


Figure 3 - UC1 Data Synchronization

The System runs every hour to synchronize local database with data sources provided by the other (WHY and HOW) Sporte2 modules.

It downloads data at the hourly granularity, evaluating each benchmark indicator and storing results into the local database, as per the schematization in the diagram.

2.2.3.2 UC2 – Benchmark management

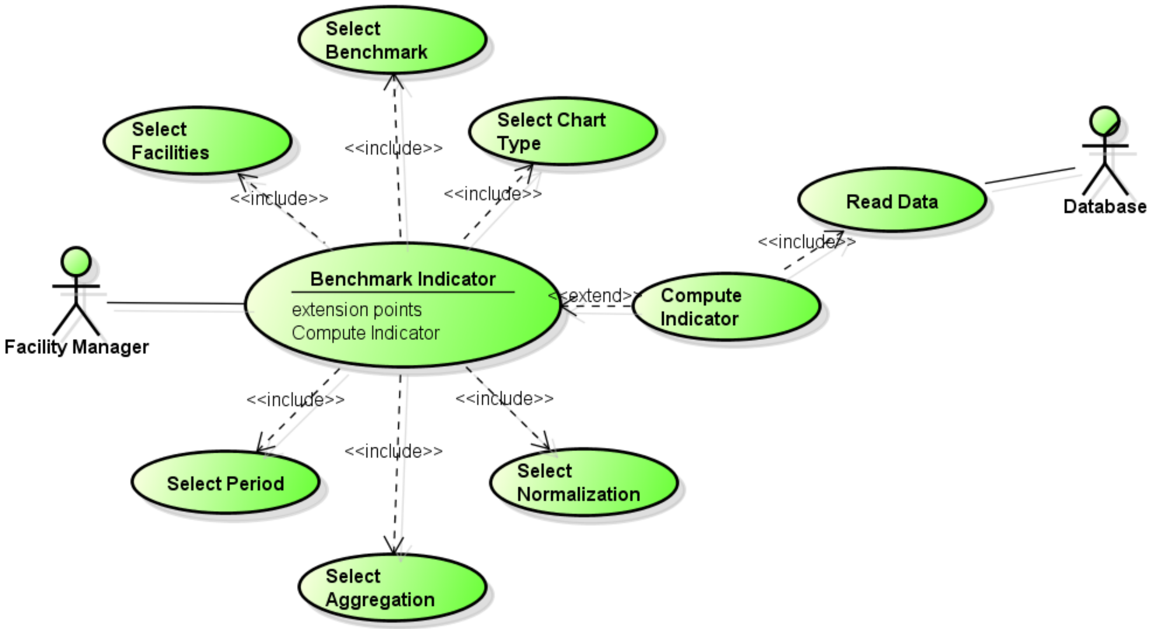


Figure 4 - UC2 Benchmark management

Benchmark indicator use case is composed and uses the following use cases to compute benchmark indicator and rendering chart.

Computation of benchmark indicator requires the selection of facility, chart type, normalization, aggregation and period. Sub Use Cases for the UC2 are detailed in the table below, and visualized as per the schematic representation.

Use Case Code	Use Case Name
UC2.1	Benchmark Indicator
UC2.2	Select Facilities
UC2.3	Select Benchmark
UC2.4	Select Aggregation
UC2.5	Select Period
UC2.6	Select Chart Type
UC2.7	Select Normalization
UC2.8	Compute Indicator
UC2.9	Read Data

Table 3 - Benchmark management

2.2.3.3 UC3 – Dashboard management

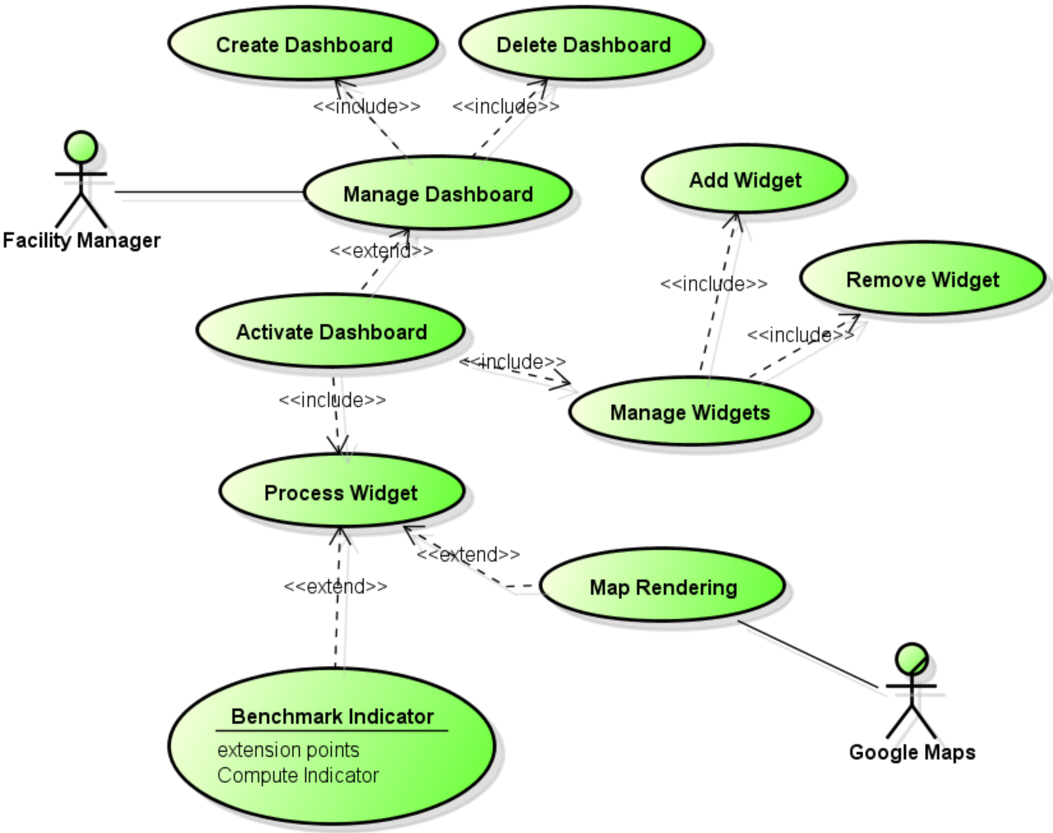


Figure 5 - UC3 Dashboard management

Dashboard management involves the management of dashboards for each user and the widget management per dashboard.

System offers the ability to create an infinite number of dashboards per user.

Each dashboard can contain different widget types.

On the active dashboard system perform widget processing to evaluate benchmark, indicators and tariffs and so on. Sub Use Cases for the UC3 are detailed in the table below, and visualized as per the schematic representation.

Use Case Code	Use Case Name
UC3.1	Create Dashboard
UC3.2	Delete Dashboard
UC3.3	Activate Dashboard
UC3.4	Manage Widgets
UC3.5	Remove Widget
UC3.6	Add Widget
UC3.7	Process Widget
UC3.8	Map Rendering
UC3.9	Benchmarks Indicator

Table 4 - Dashboard management

2.2.3.4 UC4 – Administration

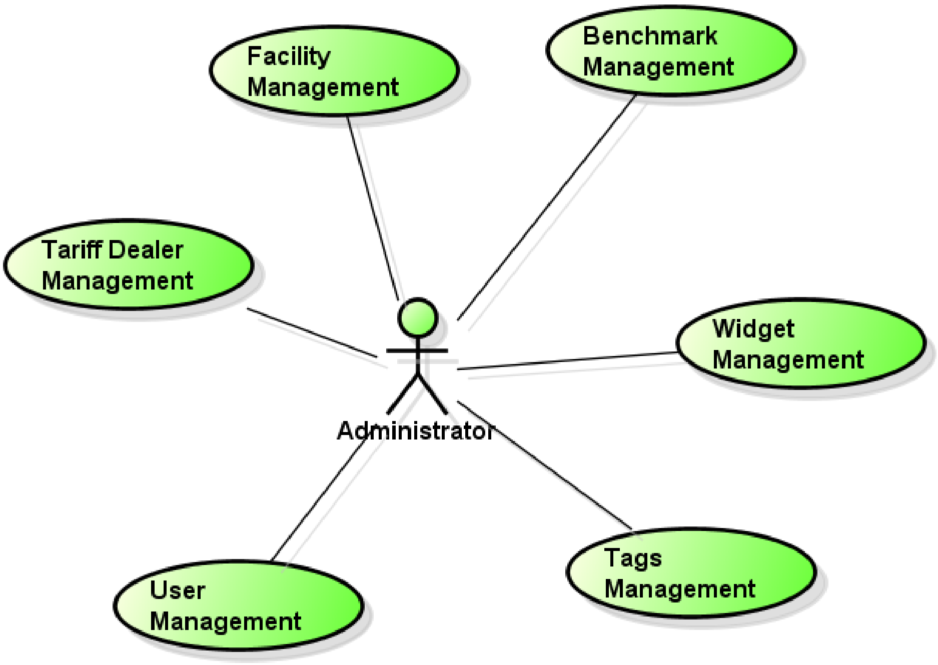


Figure 6 - UC4 Administration

Administration includes the management of all the aspect and configuration of the application. In addiction it enables the execution of CRUD operation on each database table available in the module.

Sub Use Cases for the UC4 are detailed in the table below, and visualized as per the schematic representation.

Use Case Code	Use Case Name
UC4.1	Facility Management
UC4.2	Benchmark Management
UC4.3	Widget Management
UC4.4	Tags Management
UC4.5	Tariff Dealer Management
UC4.6	User Management

Table 5 - Administration

2.2.3.5 UC5 – Formula Management

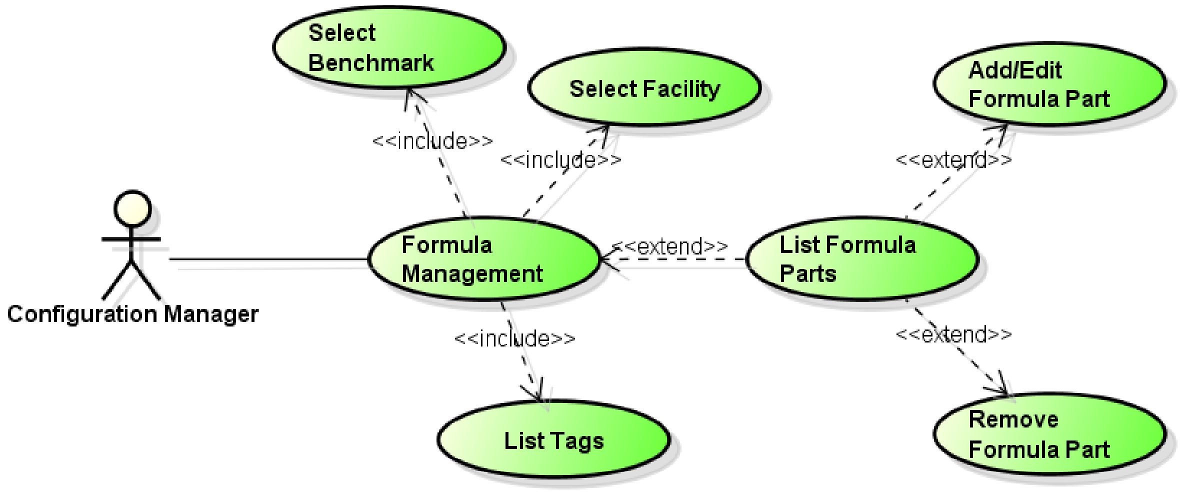


Figure 7 - Formula Management

Formula management enables the authorized user to create or edit formulas needed by synchronizer service to retrieve and compute benchmark indicators.

User can build formula by adding blocks (formula parts).

Each block is composed by:

- Sign
- Tag id
- Multiplier

Each formula requires the selection of a facility and a benchmark indicator.

Sub Use Cases for the UC5 are detailed in the table below, and visualized as per the schematic representation.

Use Case Code	Use Case Name
UC5.1	Formula Management
UC5.2	Select Benchmark
UC5.3	Select Facility
UC5.4	List/Select Tags
UC5.5	List Formula Parts
UC5.6	Add/Edit Formula Parts
UC5.7	Remove Formula Parts

Table 6 - Formula Management

2.2.3.6 UC6 – Tariff Management

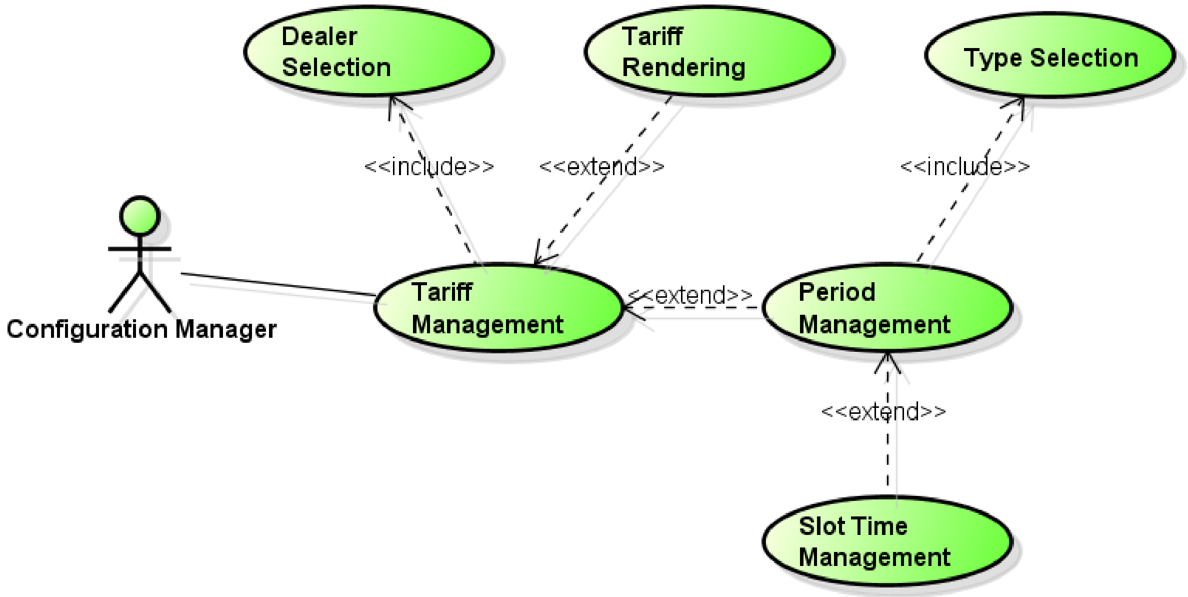


Figure 8 - Tariff Management

Tariff management regards management of tariffs, periods and slot times.
A tariff can have one or more periods.
A period has a given range of applicability, a name and one or more price values (with a validity date).

Period has one or more slot time defined that describes applicability of the period inside a time range (usually the time range is defined with start hour and end hour of a day).

Sub Use Cases for the UC6 are detailed in the table below, and visualized as per the schematic representation.

Use Case Code	Use Case Name
UC6.1	Tariff Management
UC6.2	Dealer Selection
UC6.3	Tariff Rendering
UC6.4	Period Management
UC6.5	Type Selection
UC6.6	Slot Time Management

Table 7 - Tariff management

2.3 SportE^2 Where Module open technologies

In this chapter the technologies used to develop the SportE^2 Where Module, the sources of data and the activities used to update the local database from the data source are described.
In particular the technologies and the solutions adopted for the components building the SportE^2 Where Module will be described.

2.3.1 Database

The database is used to store the data about the configuration and user profiles, information about facilities and their characterization and benchmark indicator values computed with the data retrieved from web services.
The database server used for development and test is MySql Server 5 [1]. It is an open source and widely used SQL server from Oracle.

2.3.2 Data Synchronizer

Data Synchronizer is a scheduled process that works in background to retrieve data from How and Why web services and compute benchmark indicator evaluation and store them into the database.
It uses Quartz Scheduler to schedule the process into an application server.
Quartz Scheduler is an open source scheduler engine widely used and fully integrated into the most used Java application Server.

2.3.3 Scheduler

A job scheduler is a computer application for controlling unattended background program execution (commonly called batch processing).

Batch system, Distributed Resource Management System (DRMS), and Distributed Resource Manager (DRM) can be taken as Synonyms. Today's job schedulers, often termed workload automation, typically provide a graphical user interface and a single point of control for definition and monitoring of background executions in a distributed network of computers.

Increasingly, job schedulers are required to orchestrate the integration of real-time business activities with traditional background IT processing across different operating system platforms and business application environments.

2.3.3.1 Quartz [2]

Quartz is a job scheduling service that can be integrated into a wide variety of Java applications. Quartz is generally used for Enterprise class applications to support process workflow, system maintenance actions and to provide timely services within the applications.

Quartz is a full-featured, open source job scheduling service that can be integrated with, or used alongside virtually any Java application - from the smallest stand-alone application to the largest e-commerce system. Quartz can be used to create simple or complex schedules for executing tens, hundreds, or even tens-of-thousands of jobs; jobs whose tasks are defined as standard Java components that may execute virtually anything you may program them to do. The Quartz Scheduler includes many enterprise-class features, such as support for JTA transactions and clustering.

2.3.4 Where Web Application

The SportE^2 WHERE Module is a web application built using JavaEE [3] technologies. In particular the application has been developed using JavaEE7 and its features like EJB, JSF and JPA.

2.3.4.1 Java

Java Platform, Enterprise Edition or Java EE is Oracle's enterprise Java computing platform. The platform provides an API and runtime environment for developing and running enterprise software, including network and web services, and other large-scale, multi-tiered, scalable, reliable, and secure network applications. Java EE extends the Java Platform, Standard Edition (Java SE), providing an API for object-relational mapping, distributed and multi-tier architectures, and web services. The platform incorporates a design based largely on modular components running on an application server. Software for Java EE is primarily developed in the Java programming language. The platform emphasizes Convention over configuration and annotations for configuration.

JavaEE has been used in consideration that it represents a standard for the web development, inside the open source world.

2.3.4.2 EJB [4]

Enterprise JavaBeans (EJB) is a managed, server-side component architecture for modular construction of enterprise applications.

The EJB specification is one of several Java APIs in the Java EE specification. EJB is a server-side model that encapsulates the business logic of an application. The EJB specification intends to provide a standard way to implement the back-end 'business' code typically found in enterprise applications (as opposed to 'front-end' interface code). Such code addresses the same types of problems, and solutions to these problems are often repeatedly re-implemented by programmers. Enterprise JavaBeans are intended to handle such common concerns as persistence, transactional integrity, and security in a standard way, leaving programmers free to concentrate on the particular problem at hand.

To deploy and run Enterprise JavaBeans, a Java EE Application server can be used, as these include an EJB container by default.

EJB simplifies the development of piece of code that runs as service and are able to decouple logic from presentation inside a process of code development.

2.3.4.3 JSF [5]

JavaServer Faces (JSF) is a Java specification for building component-based user interfaces for web applications. It was formalized as a standard through the Java Community Process and is part of the Java Platform, Enterprise Edition.

JSF 2 uses Facelets as its default templating system.

PrimeFaces [6]

PrimeFaces is an open source component suite for Java Server Faces featuring 100+ Ajax powered rich set of JSF components. Main goal of PrimeFaces is to create the ultimate component suite for JSF.

- 100+ rich set of components (HtmlEditor, Dialog, AutoComplete, Charts and more).
- Built-in Ajax with Lightweight Partial Page Rendering.
- Native Ajax Push/Comet support.
- One jar, zero-configuration and no required dependencies.
- Skinning Framework with 30 pre-designed themes.
- Extensive documentation.

2.3.4.4 Web Services

The SportE^2 Where Module uses web services both to consume how and why data and to produce benchmark index data.
The technology used in web service layer is Jax Ws, a standard java implementation to consume and produce SOAP service.

A web service is a method of communication between two electronic devices over the World Wide Web. A web service is a software function provided at a network address over the web or the cloud, it is a service that is "always on" as in the concept of utility computing.

The W3C defines a "Web service" as:

[...] a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using SOAP messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

The W3C also states:

We can identify two major classes of Web services:

- REST-compliant Web services, in which the primary purpose of the service is to manipulate XML representations of Web resources using a uniform set of "stateless" operations; and
- arbitrary Web services, in which the service may expose an arbitrary set of operations.

2.3.4.5 Web and Application Server

The web server is mainly for receiving document requests and data submission from web browsers through the HTTP protocol on top of the Internet's TCP/IP layer. The main function of the web server is to feed HTML files to the web browsers. If the client is requesting a static existing file, it will be retrieved on a server hard disk and sent back to the web browser right away.

The application server is responsible for computing the business logics of the web application, like carrying out a bank account fund transfer and computing the shortest route to drive from one city to another. If the business logic is simple or the web application is only used by a small group of clients, the application server is usually missing and business logics are computed in the web server extensions. But for a popular web application that generates significant computation load for serving each client, the application server will take advantage of a separate hardware server machine to run business logics more efficiently. This is a good application of the divide-and-conquer problem solving methodology.

2.3.4.5.1 Oracle Glassfish ***[7]***

GlassFish is an open-source application server project started by Sun Microsystems for the Java EE platform and now sponsored by Oracle Corporation.

GlassFish is the reference implementation of Java EE and as such supports Enterprise JavaBeans, JPA, JavaServer Faces, JMS, RMI, JavaServer Pages, servlets, etc. This allows developers to create enterprise applications that are portable and scalable, and that integrate with legacy technologies. Optional components can also be installed for additional services.

3. SportE² Where Module as a Product

3.1 SportE² Where Module generalities

3.1.1 Database

The database schema illustrates the entities involved into the management of the information about the benchmark, facility, formula, tariffs, dashboard, widgets and users of the Where Module. The schema also shows the relationship between involved entities.

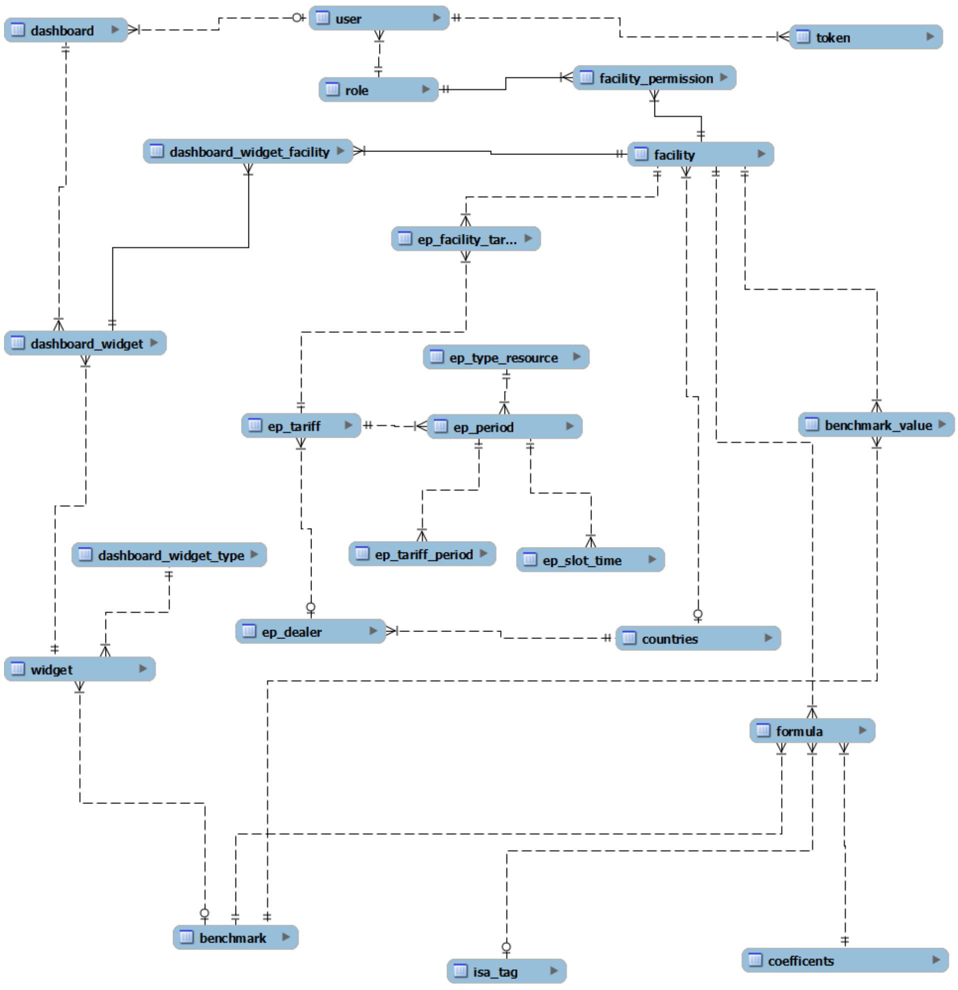


Figure 9 - database schema

3.1.2 Dashboard

The Dashboard is a configurable area where every user can define its own group of widget.

Each user can define multiple Dashboards and configure multiple widgets and switch between them.

When a Dashboard is activated, the relevant widgets appears; they can be moved, rearranged, deleted, and recreated (so that more than one of the same Widget can be open at the same time, possibly with different settings). The widgets are arranged into a 3 x n grid.

Dashboard representation with custom widget enables the user to check and monitor different parameter at the same time with an immediate view of the systems status. The possibility to add and organize different widget permits the user to create a personal and simple to read status of the system.

Digital dashboards allow managers to monitor the contribution of the various indicators together. Benefits of using digital dashboards include:

- Visual presentation of performance measures
- Ability to identify and correct negative trends
- Measure efficiencies/inefficiencies
- Ability to generate detailed reports showing new trends
- Ability to make more informed decisions based on collected business intelligence
- Align strategies and organizational goals
- Saves time compared to running multiple reports
- Gain total visibility of all systems instantly
- Quick identification of data outliers and correlations

3.1.3 Widgets

In the application are defined many widgets, there are different types of widgets:

- Chart
- Maps
- Weather
- Indicator
- Tariff
- Consumption

Chart Widget

Chart widget displays the results of one indicator with charts. There is a widget for each defined indicator.
Each widget configuration permits to configure the facilities, the period and the aggregation level to be monitored, the type of chart (Line or Bar) and the normalization type (None, Area, Occupancy)

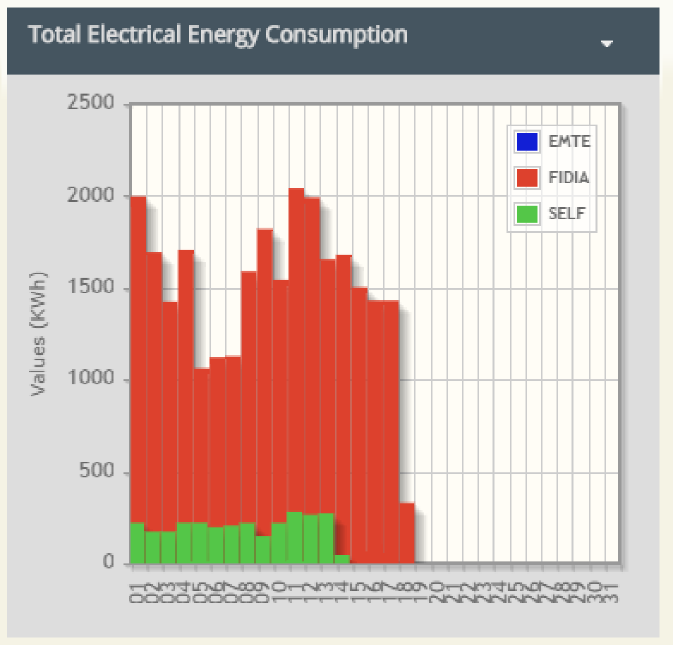


Figure 10 - Chart Widget

Map Widget

Map widget displays the selected facilities into a Google map interface. Each facility is shown with a graphical marker positioned on the map at the defined longitude and latitude. By clicking on a marker, a popup is opened to show details about the selected facility.
The configuration of the widgets permits to define the list of facilities, which the user wants to locate into the map, from the list of all available facilities configured into the application.

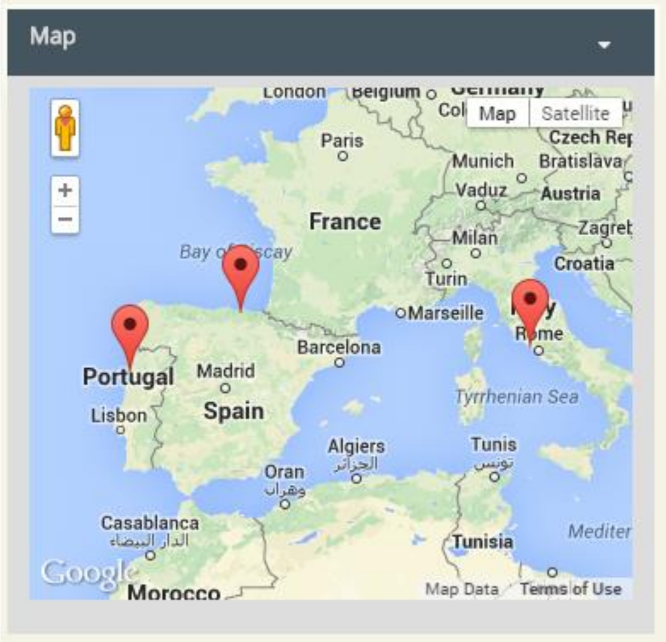
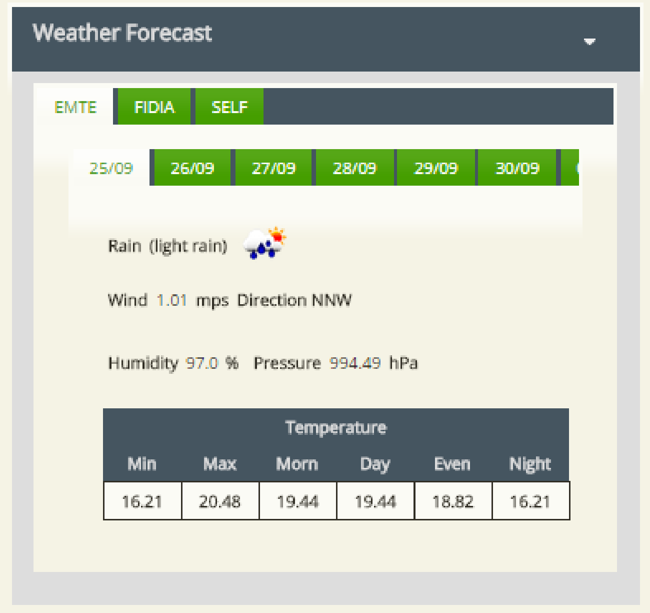


Figure 11 - Map Widget

Weather Widget

Weather widget displays forecast information on weather for the selected facilities.
It displays 7 days tab forecast data for each facility with indication of temperature (max, min, day, night, morning, evening), wind direction and speed, humidity and pressure.
It uses the OpenWeatherMap service available online.



Weather Forecast

EMTE FIDIA SELF

25/09 26/09 27/09 28/09 29/09 30/09

Rain (light rain)

Wind 1.01 mps Direction NNW

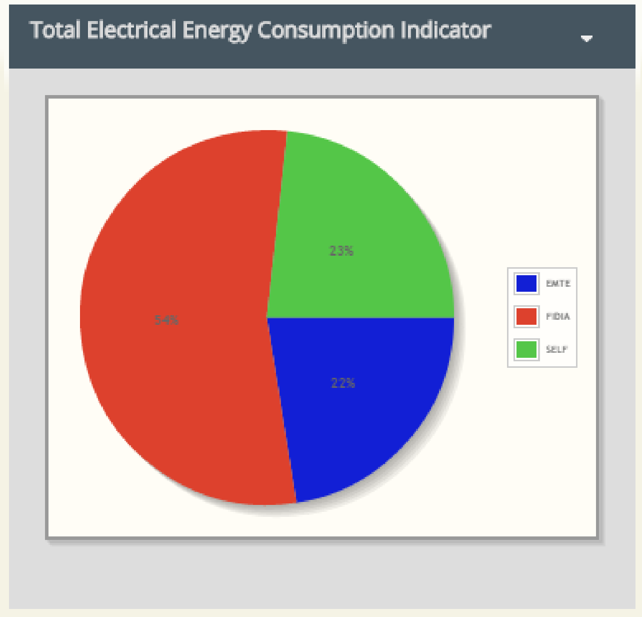
Humidity 97.0 % Pressure 994.49 hPa

Temperature					
Min	Max	Morn	Day	Even	Night
16.21	20.48	19.44	19.44	18.82	16.21

Figure 12 - Weather Widget

Indicator Widget

Indicator widget displays the result of one indicator with pie chart. There is a widget for each defined indicator.
Each widget configuration permits to configure the facilities, the period and the aggregation level to be monitored and the normalization type (None, Area, Occupancy)

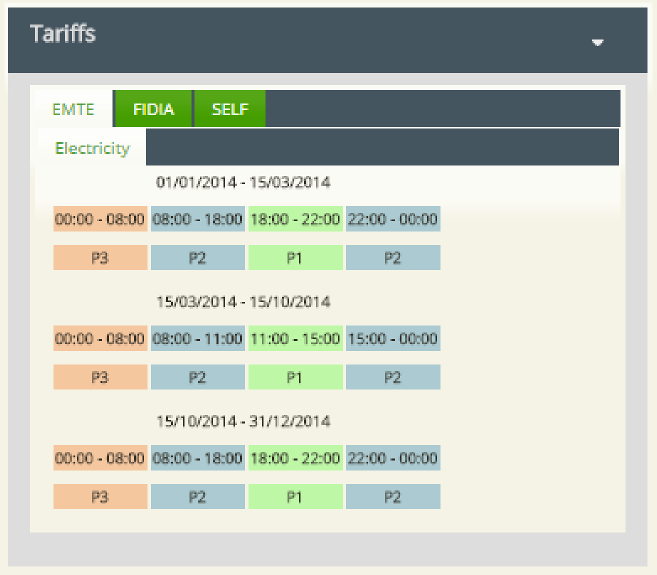


Total Electrical Energy Consumption Indicator

Figure 13 - Indicator Widget

Tariff Widget

The Tariff Widget displays time schedule of the tariffs applied to the selected facilities.
It displays a table with each configured period, the relative name and the time range of applicability for electricity, gas and water tariffs.



01/01/2014 - 15/03/2014

00:00 - 08:00	08:00 - 18:00	18:00 - 22:00	22:00 - 00:00
P3	P2	P1	P2

15/03/2014 - 15/10/2014

00:00 - 08:00	08:00 - 11:00	11:00 - 15:00	15:00 - 00:00
P3	P2	P1	P2

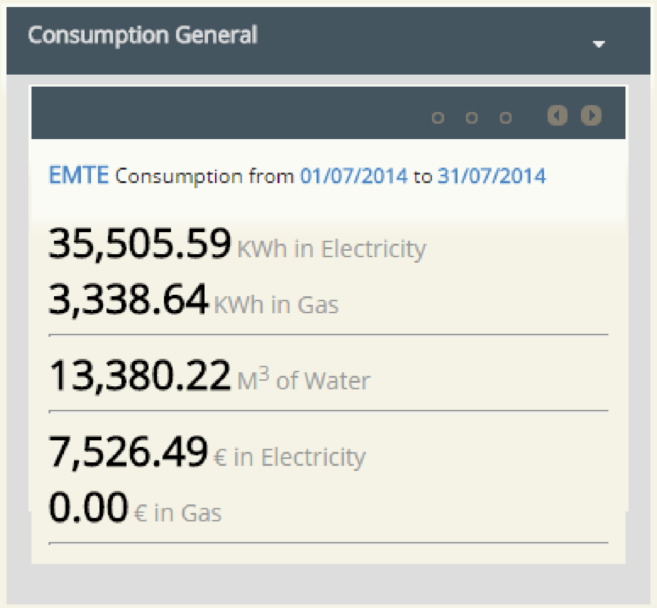
15/10/2014 - 31/12/2014

00:00 - 08:00	08:00 - 18:00	18:00 - 22:00	22:00 - 00:00
P3	P2	P1	P2

Figure 14 - Tariff Widget

Consumption Widget

Consumption widget displays information on total consumption, expressed in electricity, gas and water and the relative costs calculated using the defined tariffs for each facility.



EMTE Consumption from 01/07/2014 to 31/07/2014

35,505.59 KWh in Electricity

3,338.64 KWh in Gas

13,380.22 M^3 of Water

7,526.49 € in Electricity

0.00 € in Gas

Figure 15 - Consumption Widget

3.1.4 Configuration

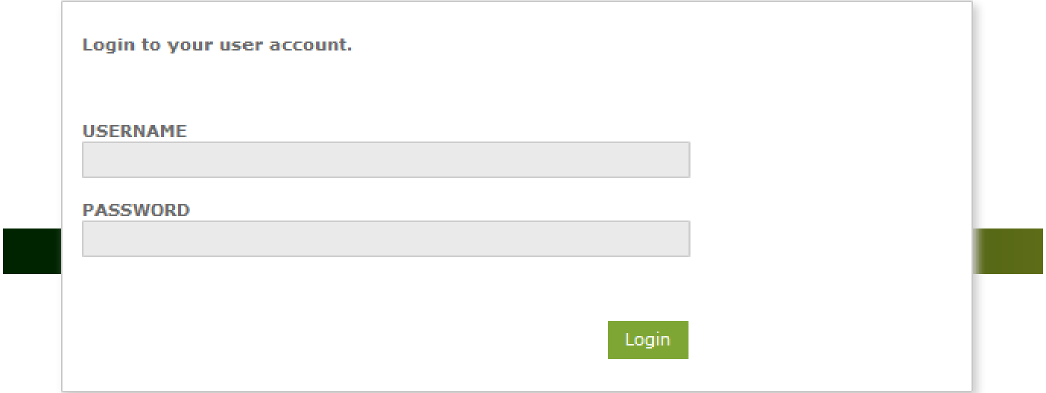
Each user can create an indefinite number of dashboards. For each dashboard, the user can add and delete widgets and configure them.
The dashboard and widget configuration are stored in a database.

3.2 SportE^2 Where Module structure and functional pictures

In this section the user interface of the SportE^2 Where Module is described.

3.2.1 Login Interface

As this interface is deployed a login screen was implemented in order to protect the interface from unauthorized access and for identification of the accessing user. The user authentication is made by using a valid username and password pair. Figure 16shows the login page of the SportE^2 WHERE module.



Login to your user account.

USERNAME

PASSWORD

Figure 16: SportE^2 Where Module Login Page

3.2.2 Main Page

The main page presents some clickable icons (activated on the basis of user permission and administration level) to access to the different module functionalities.
In the next screen shot are visible buttons for benchmark page, dashboard page, manual updater service, formula editor and administration.

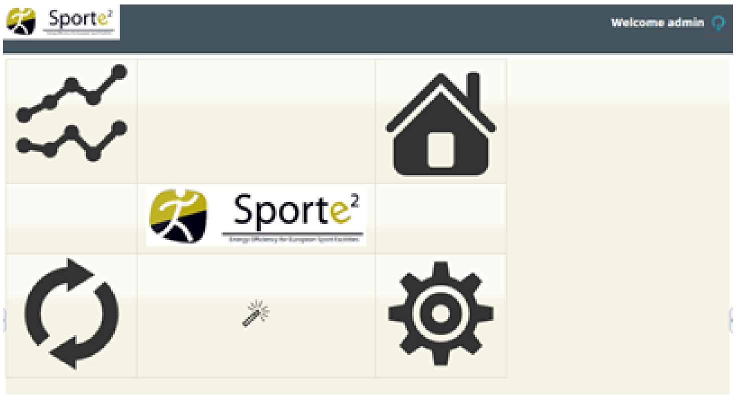


Figure 17 - SPORTE2 WHERE Module Main Page

3.2.3 Benchmark Interface

After user authentication, the benchmark interface is shown.

In the top of this interface, a panel is available to select the main category of benchmark among *Forecast*, *Primary* and *Secondary*.

After benchmark category selection, the submenu (above the main menu) loads the available benchmark indicators.

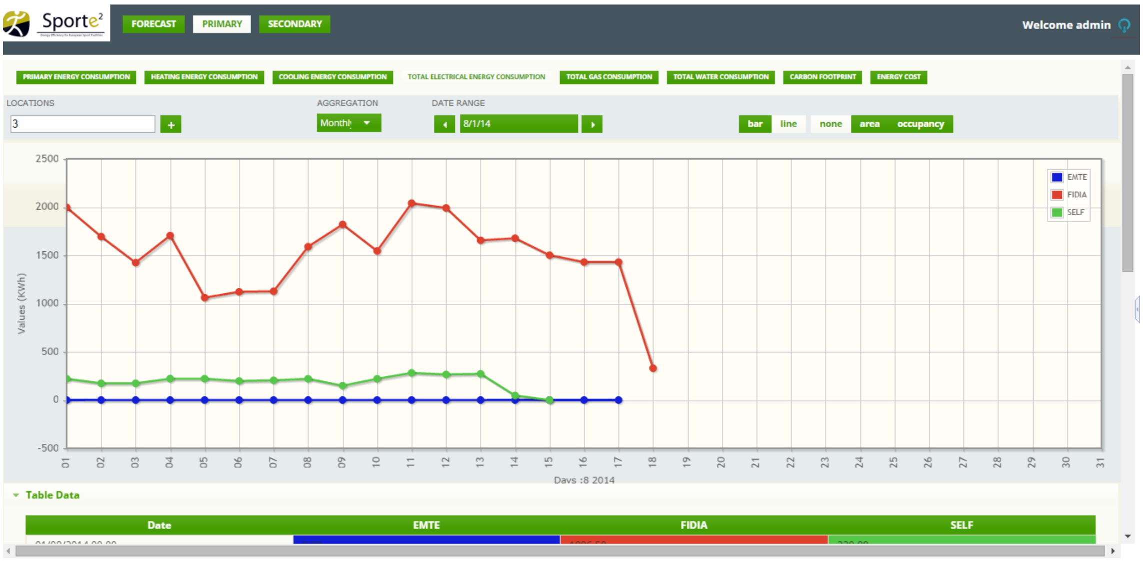
Below the submenu, the next panels are present:

- Location selection: let user to select the facility on which calculate the selected benchmark indicator.
- Aggregation type selection: user can select four aggregation level (daily, weekly, monthly and yearly) depending on the selected indicator.
- Time selection: user can modify time interval for data retrieval.
- Chart type selection: user can select between Bar or Line chart type to displays indicator values
- Normalization selection: user can select between None, Area or Occupancy data normalization.

The central area of the page contains the selection of the available normalization parameter to apply on the raw data retrieved from the web services.

Available normalization are described in deliverable D1.1 in the paragraph 3.3.3.3 "Normalization of measured data".

Finally, the central area of the page contains the graph area Where the charts are drawn (bar charts, trend lines, etc.).



LOCATIONS 3

AGGREGATION Monthly

DATE RANGE 8/1/14

Date	EMTE	FIDIA	SELF

Figure 18: SportE^2 Where Module Benchmark Page

3.2.4 Facility selection panel

The following screen is activated by user clicking on the facility selection button located on the Benchmark page: through this dialog box, user can select the facilities to benchmark.

Selected facilities are "moved" from the panel on the left, named *Available Facilities*, to the panel on the right, named *Selected Facilities*.

The four stacked buttons in the middle permit to:

- Add the highlighted facilities from the *Available Facilities* box to the *Selected Facilities* box;
- Add all the facilities from the *Available Facilities* to the *Selected Facilities* box;

- Remove the highlighted facilities from the *Selected Facilities* to the *Available Facilities* box;
- Remove all the facilities from the *Selected Facilities* to the *Available Facilities* box.

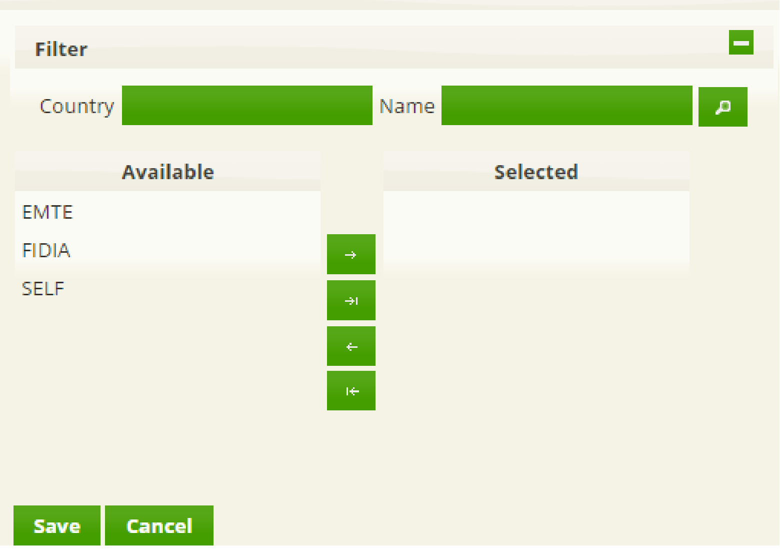


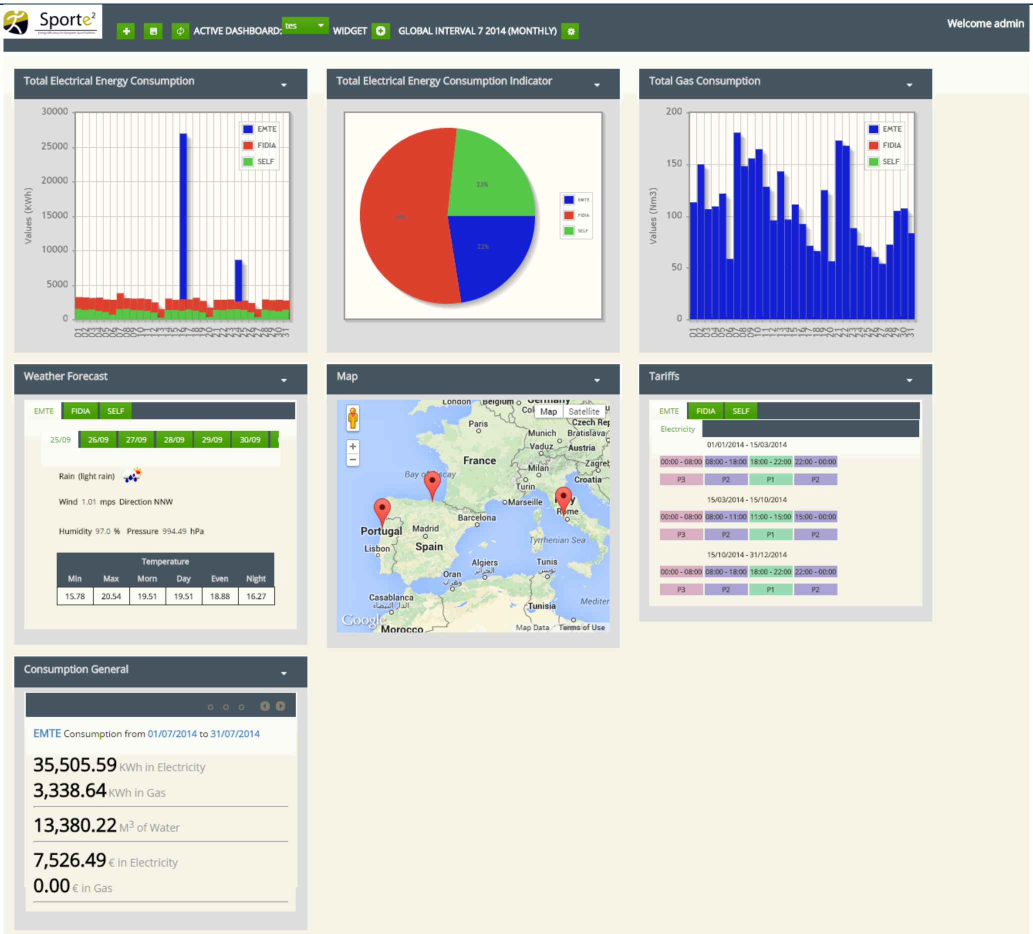
Figure 19: SportE^2 Where Module Facility Selection Dialog

3.2.5 Dashboard interface

Dashboard screen presents a toolbar with the following functionalities:

- Create dashboard
- Save current dashboard
- Reload dashboard configuration from database
- Menu to select current dashboard from a list of available dashboard for the user.
- Add new widget to the dashboard
- Change global aggregation and period settings for all the widget that need this information.

In the central area are rendered the widget activated in the current dashboard.



EMTE FIDIA SELF

25/09 26/09 27/09 28/09 29/09 30/09

Rain (light rain)

Wind 1.01 mps Direction NNW

Humidity 97.0 % Pressure 994.49 hPa

Temperature					
Min	Max	Morn	Day	Even	Night
15.78	20.54	19.51	19.51	18.88	16.27

EMTE FIDIA SELF

Electricity

01/01/2014 - 15/03/2014

00:00 - 08:00	08:00 - 18:00	18:00 - 22:00	22:00 - 00:00
P3	P2	P1	P2

15/03/2014 - 15/10/2014

00:00 - 08:00	08:00 - 11:00	11:00 - 15:00	15:00 - 00:00
P3	P2	P1	P2

15/10/2014 - 31/12/2014

00:00 - 08:00	08:00 - 18:00	18:00 - 22:00	22:00 - 00:00
P3	P2	P1	P2

EMTE Consumption from 01/07/2014 to 31/07/2014

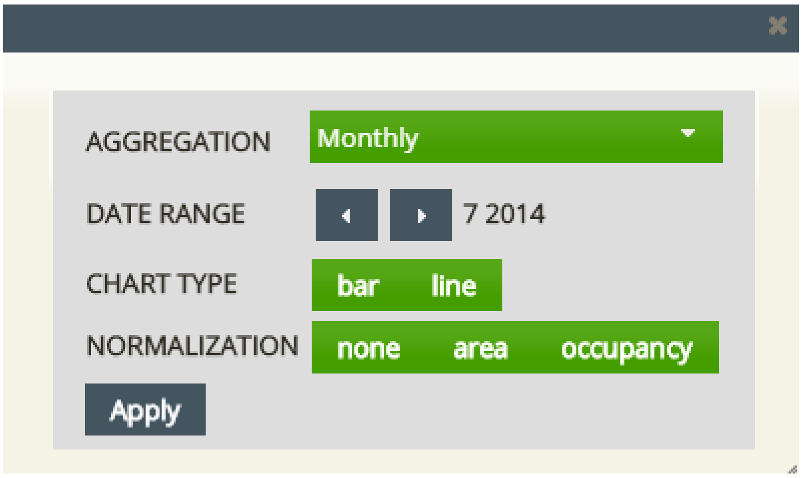
35,505.59 KWh in Electricity
3,338.64 KWh in Gas

13,380.22 M^3 of Water

7,526.49 € in Electricity
0.00 € in Gas

Figure 20 - SportE^2 Where Module Dashboard Page

The following screen show the interface for aggregation and period selection, chart type and normalization settings.


AGGREGATION Monthly

DATE RANGE 7 2014

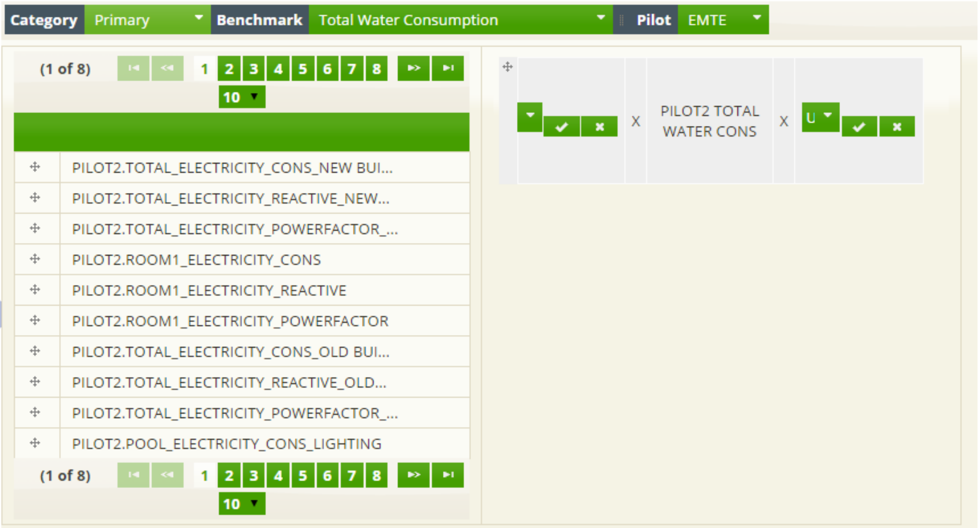
CHART TYPE bar line

NORMALIZATION none area occupancy

Figure 21 - Chart Base Widget Configuration Dialog

3.2.6 Formula Editor Interface

This screen show formula editor page. In the top area there are the menus to select category of benchmark, benchmark and facility. By selecting these values the central area is loaded with the list of available tags for the facility on the left and the formula on the right.



Category: Primary
Benchmark: Total Water Consumption
Pilot: EMTE

(1 of 8)

- PILOT2.TOTAL_ELECTRICITY_CONS_NEW BUI...
- PILOT2.TOTAL_ELECTRICITY_REACTIVE_NEW...
- PILOT2.TOTAL_ELECTRICITY_POWERFACTOR_...
- PILOT2.ROOM1_ELECTRICITY_CONS
- PILOT2.ROOM1_ELECTRICITY_REACTIVE
- PILOT2.ROOM1_ELECTRICITY_POWERFACTOR
- PILOT2.TOTAL_ELECTRICITY_CONS_OLD BUI...
- PILOT2.TOTAL_ELECTRICITY_REACTIVE_OLD...
- PILOT2.TOTAL_ELECTRICITY_POWERFACTOR_...
- PILOT2.POOL_ELECTRICITY_CONS_LIGHTING

(1 of 8)

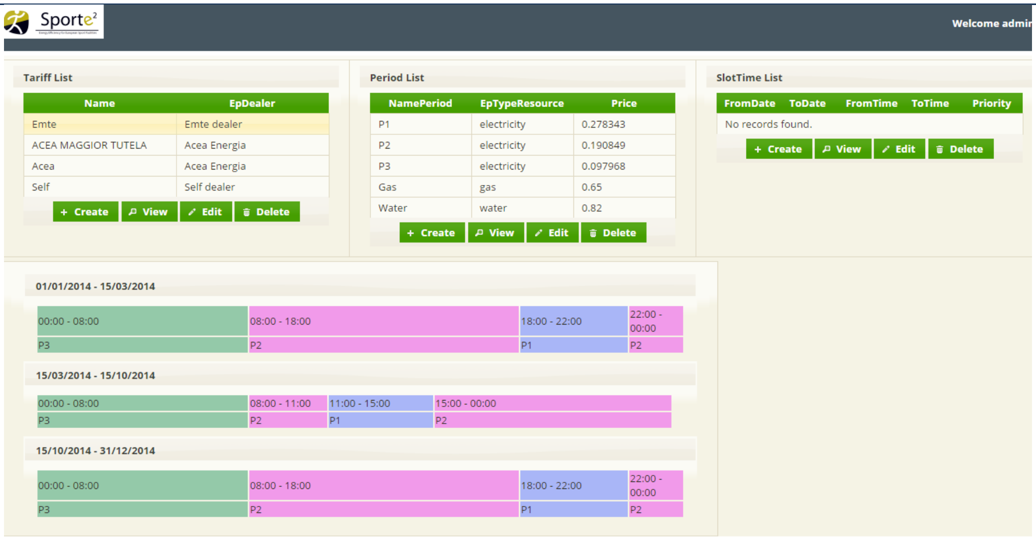
Figure 22 - SportE^2 Where Module Formula Editor Page

3.2.7 Tariff Editor Interface

The tariff editor interfaces is divided into 4 parts.

1. Tariff list
2. Period list
3. Slot time list
4. Tariff rendering area

In the first area will be listed the available tariffs and the user can create new tariff.
In the second area will be listed periods available for the selected tariff. User can create new periods.
In the third area will be listed the slot times defined for the selected period. User can create new slot times.
In the fourth area is displayed a graphical representation of the tariff with the relative periods.



Sporte^2

Welcome admi

Tariff List

Name	EpDealer
Emte	Emte dealer
ACEA MAGGIOR TUTELA	Acea Energia
Acea	Acea Energia
Self	Self dealer

Period List

NamePeriod	EpTypeResource	Price
P1	electricity	0.278343
P2	electricity	0.190849
P3	electricity	0.097968
Gas	gas	0.65
Water	water	0.82

SlotTime List

FromDate	ToDate	FromTime	ToTime	Priority
No records found.				

01/01/2014 - 15/03/2014

00:00 - 08:00	08:00 - 18:00	18:00 - 22:00	22:00 - 00:00
P3	P2	P1	P2

15/03/2014 - 15/10/2014

00:00 - 08:00	08:00 - 11:00	11:00 - 15:00	15:00 - 00:00
P3	P2	P1	P2

15/10/2014 - 31/12/2014

00:00 - 08:00	08:00 - 18:00	18:00 - 22:00	22:00 - 00:00
P3	P2	P1	P2

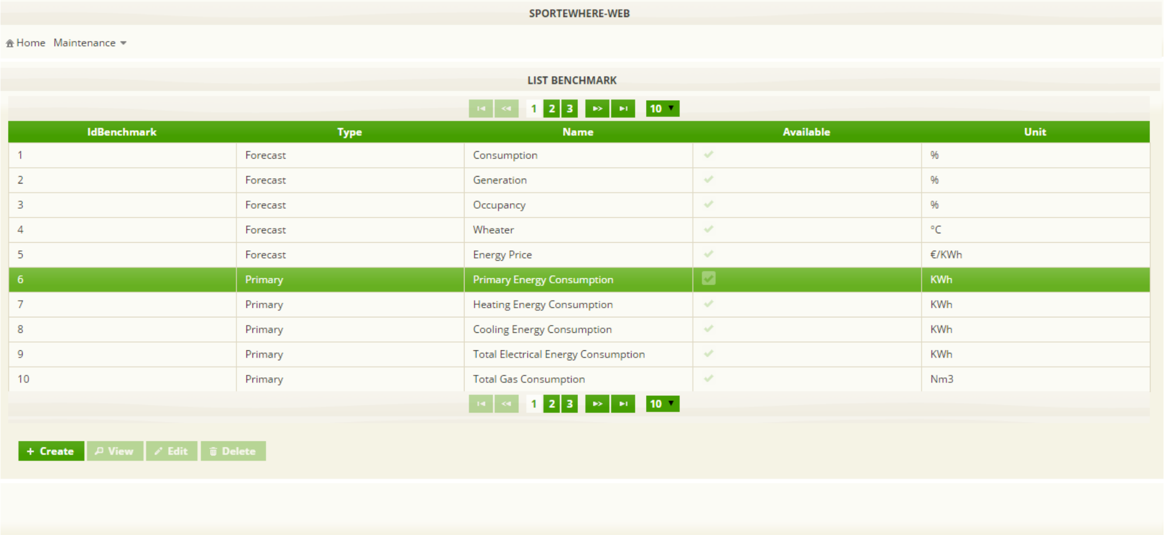
Figure 23 - SportE^2 Where Module Tariff Editor Page

3.2.8 Administration Interface

The following screen shows an example of administration interfaces.

In the top there is a menu, under maintenance voice are listed all the available tables to manage.

In the bottom area there are four buttons to create, view, edit and delete record from the table.



SPORTEWHERE-WEB

LIST BENCHMARK

IdBenchmark	Type	Name	Available	Unit
1	Forecast	Consumption	✓	%
2	Forecast	Generation	✓	%
3	Forecast	Occupancy	✓	%
4	Forecast	Wheater	✓	°C
5	Forecast	Energy Price	✓	€/KWh
6	Primary	Primary Energy Consumption	✓	KWh
7	Primary	Heating Energy Consumption	✓	KWh
8	Primary	Cooling Energy Consumption	✓	KWh
9	Primary	Total Electrical Energy Consumption	✓	KWh
10	Primary	Total Gas Consumption	✓	Nm3

Figure 24 - SportE^2 Where Module Administrative Page

3.3 SportE^2 Where Module User Manual

This chapter presents the reference steps to be followed in the use of SportE^2 Where Module.

3.3.1 Monitor page

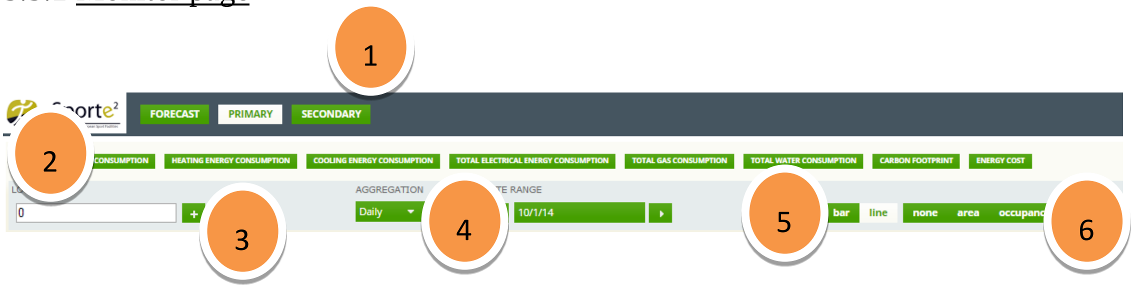


Figure 25 - Monitor Page toolbar

The following Steps are enabled, to permit to the end user to interact with the SportE^2 Where Module:

1. Select one of the main category of indicators
2. Select one indicator from the list that will be loaded after category selection
3. Select the facilities to query by clicking + button
4. Adjust aggregation type and data range
5. Switch chart type between bar or line chart
6. Switch normalization type between none, area or occupancy

3.3.2 Dashboard

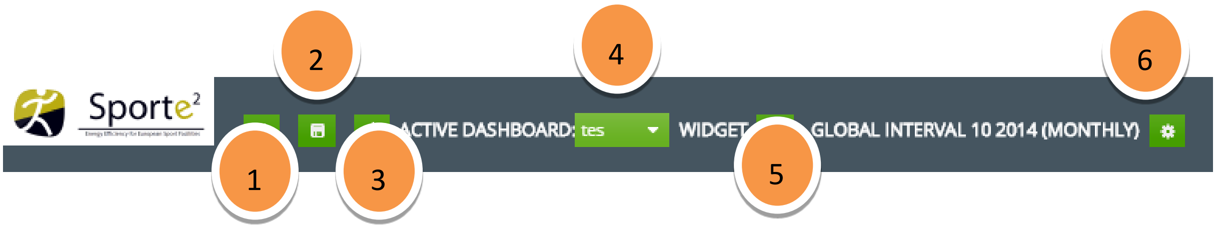
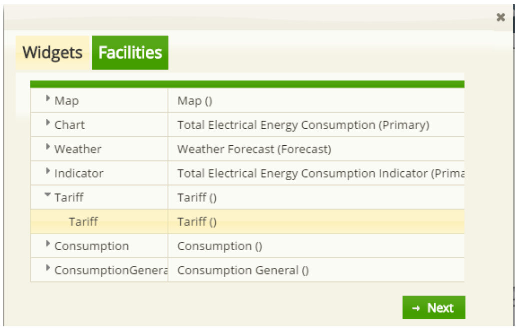


Figure 26 - Dashboard Page toolbar

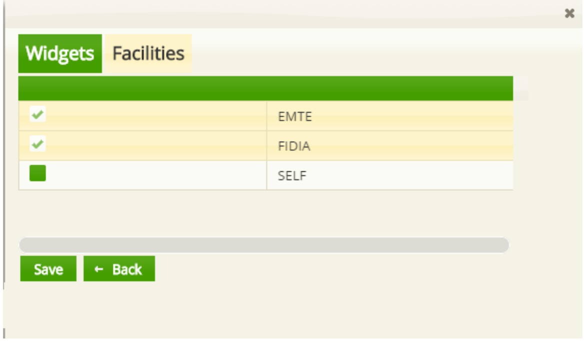
Dashboard toolbar buttons allow the user to execute the following actions:

1. Create new dashboard, a popup window will be opened to insert the name of the dashboard
2. Save dashboard changes
3. Reload dashboard configuration
4.
5. Select active dashboard from the available list.
6. Open the wizard to add a new widget



Map	Map ()
Chart	Total Electrical Energy Consumption (Primary)
Weather	Weather Forecast (Forecast)
Indicator	Total Electrical Energy Consumption Indicator (Prima
Tariff	Tariff ()
Tariff	Tariff ()
Consumption	Consumption ()
ConsumptionGenera	Consumption General ()

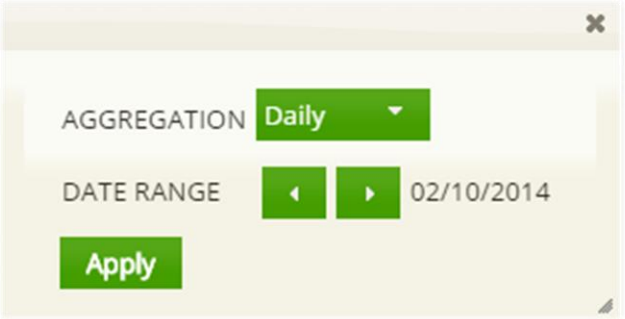
Figure 27 - Add Widget Wizard Step 1



✓	EMTE
✓	FIDIA
🟩	SELF

Figure 28 - Add Widget Wizard Step 2

7. Configure global time range, by clicking this button a popup will permit to configure aggregation and range globally for all the widget that support this configuration.

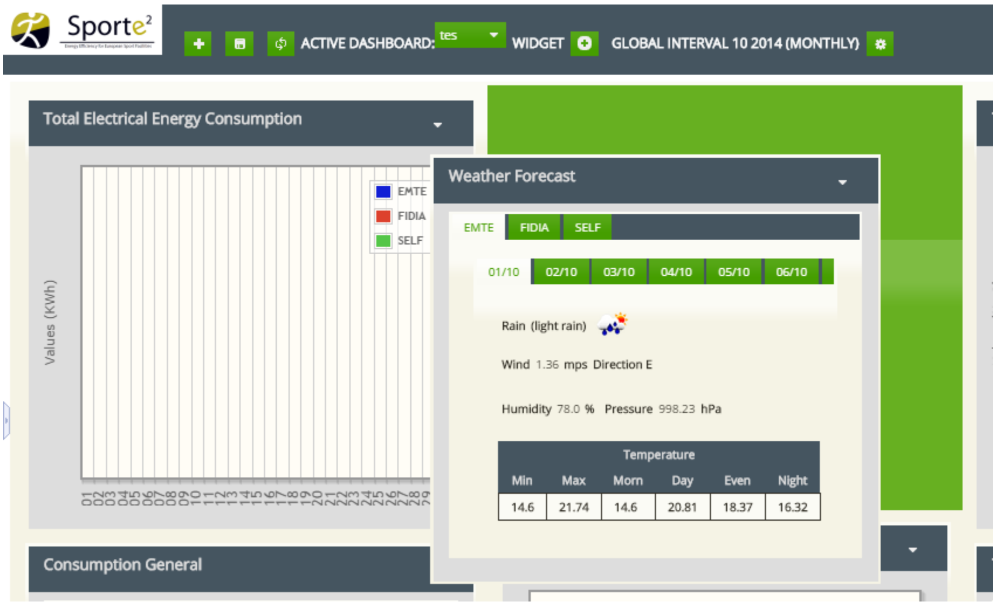


AGGREGATION Daily

DATE RANGE 02/10/2014

Figure 29 - Global time range configuration dialog

Reordering widget is a functionality offered by dashboard. By drag and drop the title bar of the widget user can change the position of the widget. A dynamic green background shows the next position where the widget will be released.



Rain (light rain)

Wind 1.36 mps Direction E

Humidity 78.0 % Pressure 998.23 hPa

Temperature					
Min	Max	Morn	Day	Even	Night
14.6	21.74	14.6	20.81	18.37	16.32

Figure 30 - Widget Reordering example

Each widget can be configured by selecting the button with down arrow. It will open a submenu with:

- Info button to show information about widget and configuration
- Facility button to add or remove facility to the widget
- Settings button to edit aggregation, time interval, chart type and normalization type
- Close button to remove the widget from the dashboard

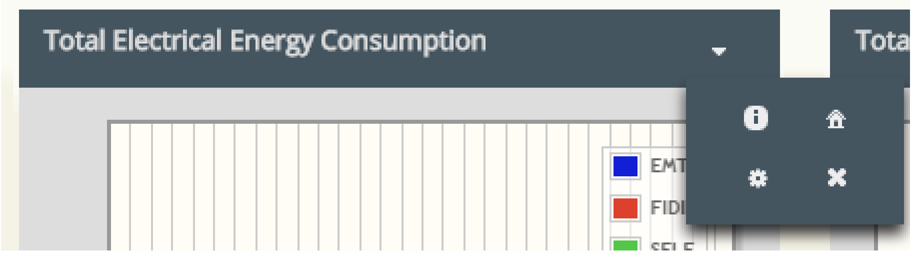
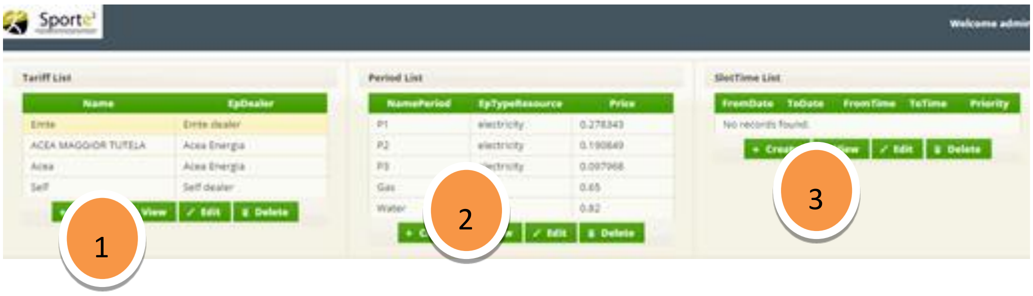


Figure 31 - Widget Configuration menu

3.3.3 Tariff editor



Tariff List

Name	EpDealer
Ernte	Ernte dealer
ACEA MAGGIOR TUTELA	Acea Energia
Acea	Acea Energia
Self	Self dealer

Period List

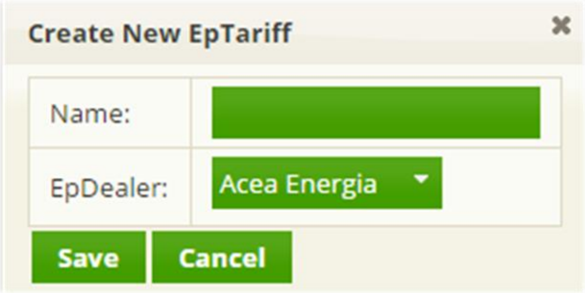
NamePeriod	EpTypeResource	Price
P1	electricity	0.278343
P2	electricity	0.190649
P3	electricity	0.097968
Gas		0.65
Water		0.82

SlotTime List

FromDate	ToDate	FromTime	ToTime	Priority
No records found.				

Figure 32 - Tariff Editor Steps

1. Create new tariff by clicking create button

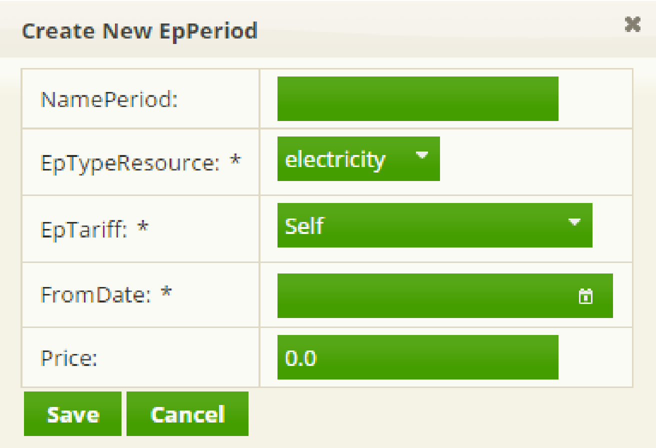


Create New EpTariff

Name:	
EpDealer:	Acea Energia

Figure 33 - New Tariff dialog

2. Create new period for the selected tariff by clicking create button

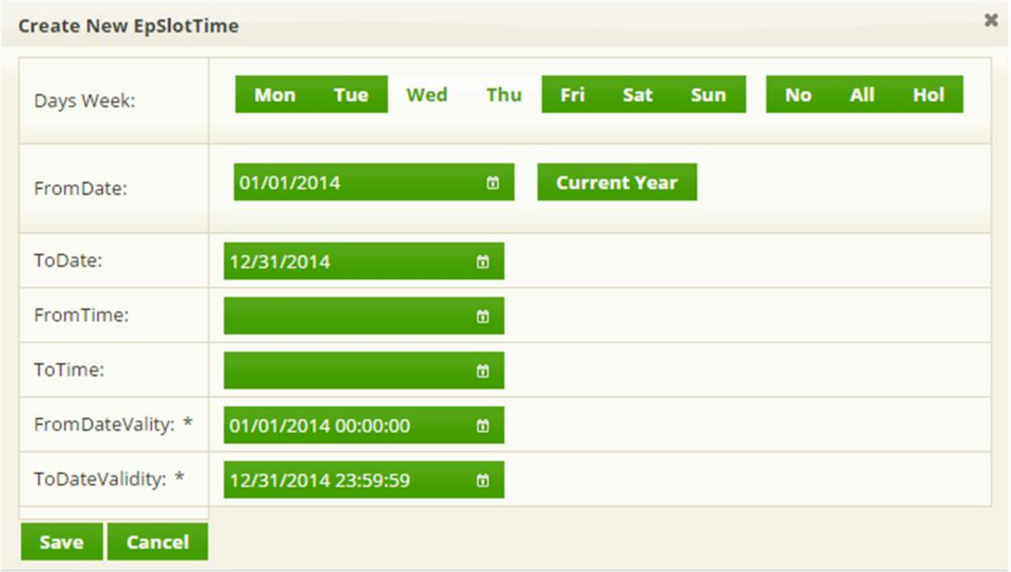


Create New EpPeriod

NamePeriod:	
EpTypeResource: *	electricity
EpTariff: *	Self
FromDate: *	
Price:	0.0

Figure 34 - New Period dialog

3. Create new time slot for the selected period
   The user had to select the days of the week for the period validity or select ALL for all the days of the week or HOL to indicate that the time slot refers to an holiday period.



Create New EpSlotTime ✖

Days Week: Mon Tue Wed Thu Fri Sat Sun No All Hol

FromDate: 01/01/2014 Current Year

ToDate: 12/31/2014

FromTime:

ToTime:

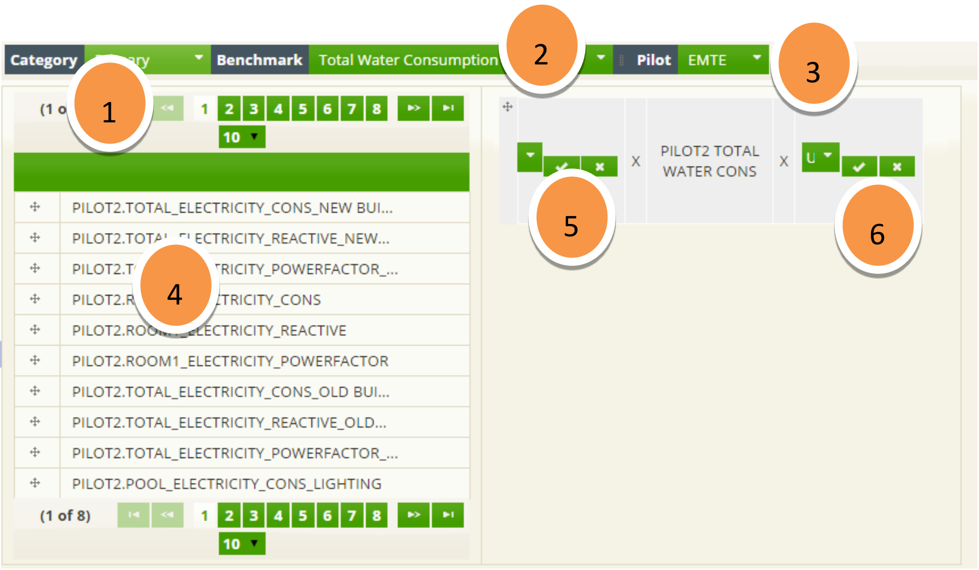
FromDateVality: * 01/01/2014 00:00:00

ToDateValidity: * 12/31/2014 23:59:59

Save Cancel

Figure 35 - New Time Slot dialog

3.3.4 Formula editor



Category …ry | Benchmark Total Water Consumption | Pilot EMTE

(1 o… 1 2 3 4 5 6 7 8 | 10

PILOT2.TOTAL_ELECTRICITY_CONS_NEW BUI...

PILOT2.TOTA…ELECTRICITY_REACTIVE_NEW...

PILOT2.T…TRICITY_POWERFACTOR_...

PILOT2.R…TRICITY_CONS

PILOT2.ROO…ELECTRICITY_REACTIVE

PILOT2.ROOM1_ELECTRICITY_POWERFACTOR

PILOT2.TOTAL_ELECTRICITY_CONS_OLD BUI...

PILOT2.TOTAL_ELECTRICITY_REACTIVE_OLD...

PILOT2.TOTAL_ELECTRICITY_POWERFACTOR_...

PILOT2.POOL_ELECTRICITY_CONS_LIGHTING

(1 of 8) 1 2 3 4 5 6 7 8 | 10

X PILOT2 TOTAL WATER CONS X U

Figure 36 - Formula Editor Steps

1. Select the category of the indicator to edit
2. Select the benchmark
3. Select the facility
4. Tag list will be reloaded to fetch the facility tag list, drag the desired tag into the formula area and a box will appear
5. Customize the formula by selecting the sign (+ o -) contribute for the block
6. Customize the formula by selecting the multiplier applied to the block from a list of available multiplier of by entering it manually.

4. Conclusions

This Deliverable closes the pathway to develop a multi-facility benchmarking system for sports facilities performed during the SportE^2 project. The aim of this document is to presents the final version of the SportE^2 Where Module as a product ready to the market.

The prototype designed in Deliverable D5.2 was further developed and consolidated and reaching a level of maturity at the end of testing phase when it was refined in order to encounter the needs of facility managers (acting as tester of the platform).

Thank to multi-facility benchmarking system released in the context of SportE2 project, the sport facility manager has the possibility to exploit the functionalities of a system capable to:

- Compare energy performances of different sport facilities located in different sites all-over the Europe;
- Analyse primary and secondary Key Performance Indicators (KPI) (e.g. total energy consumption, total gas consumption, etc.) for the different facilities (all and singularly);
- Monitor the actual situation of energy demands, consumptions, production, and grid exchange across multiple facilities (e.g. [kWh/m^2] or [TEP/m^2], [€/m^2], [kg of CO_2/m^2]) from a remote web browser. This data is acquired from the SportE^2 How Module, elaborated and provided to the users and then *re-published* through a web service interface for other usages.

The SportE2 Where Module is currently deployed at D'Appolonia Server Farm and can be accessed by authorized users at the following URL http://sporte2.dappolonia-innovation.com/. For support

Finally, in order to support the exploitation and the replication of the developed technology in the market a product data sheet of the SportE^2 Where Module is provided in Annex 1.

5. References

[1] «Mysql Server 5,» [Online]. Available: http://www.mysql.com/products/community/.

[2] «Quartz Scheduler,» [Online]. Available: http://quartz-scheduler.org/.

[3] «Java EE,» [Online]. Available: http://www.oracle.com/technetwork/java/javaee/overview/index.html.

[4] «EJB,» [Online]. Available: http://www.oracle.com/technetwork/java/javaee/ejb/index.html.

[5] «JSF,» [Online]. Available: http://www.oracle.com/technetwork/java/javaee/javaserverfaces-139869.html.

[6] «Primefaces,» [Online]. Available: http://www.primefaces.org/.

[7] «Oracle Glassfish,» [Online]. Available: https://glassfish.java.net/.

ANNEX 1 - SPORTE^2 WHERE MODULE DATA SHEET

Getting Smart about Where Module Multi-facility Benchmarking System

RINA 1861 D'APPOLONIA

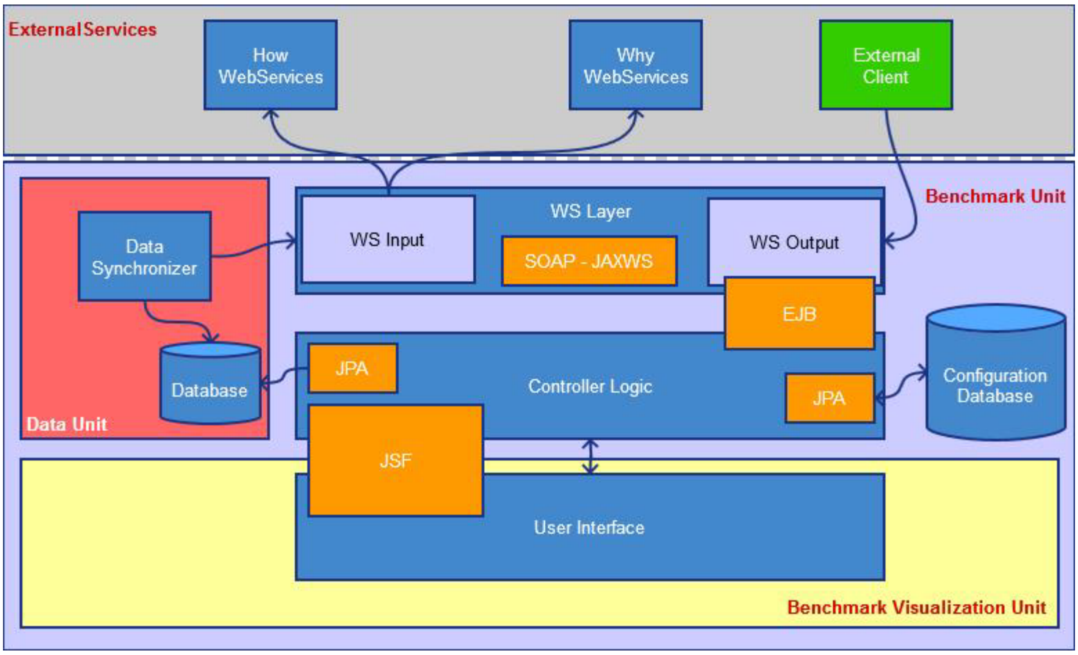
consulting, design, operation & maintenance engineering

Technical Specifications

System Architecture

Overview

The multi-facility benchmarking system is a Web Application built on ORACLE Java technologies and runs under ORACLE Glassfish Application Server. In particular the application has been developed using JavaEE7 and its features like EJB, JSF and JPA.



Data Unit - This component is focused on data import, storage and synchronization between How and Why web service with local database. It contains a scheduled process that executes the synchronization work.

Web Services - They represent a method to access to the data stored into the data storage module. These webservices are used by the multi-facility benchmarking system to intereact with the other SportE2 Modules (How and Why) from the SportE2 Ecosystem.

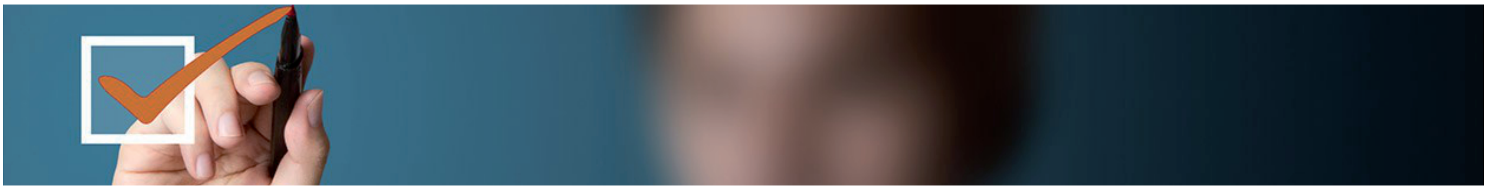
Controller Logic - This component is the bridge between Web Services Layer, Data Unit and User Interface. It takes care of the interaction with Data Unit to obtain benchmark data. Pre-processing and data normalization operations are performed for providing relieable energy benchmarking data to the end users.

Graphical User Interface - The multi-facility benchmark system Graphical User Interface (GUI) has been designed in collaboration with the technical partners and the sport facility managers/owners in order to understand their needs and the goal of the benchmarking system.

SportE2 Where Module as a multi-facility benchmarking system has been developed by using the up-to-date and open ORACLE Java Technologies in order to be easily accessible through laptop and tablet.

D'APPOLONIA

consulting, design, operation & maintenance engineering



System Functionalities

Sport is a culture and a way of life. It inspires us. Millions of spectators follow sports and just as many conduct sport activities of their own in facilities across Europe. If we can embed energy efficient thinking into this culture, it will make a difference. Sport facilities present an excellent sector for energy savings. There are many sport facilities around Europe and they consume an enormous amount of energy.

Why Where Module?

The Where module is the multi-facility benchmark system of the SportE2 suite that provides to sport facility managers or owners the remote access to the information of each managed sport facility, for monitoring the actual situation of energy demands, consumptions and production. It offers benchmarking capabilities by means of the comparison of energy situations and measures from the different monitored and controlled facilities.

Where Module key features

- Input data (HQDS, High Quality Data Set) are retrieved by using the Web Service Interface of the Energy Data Acquisition and Processing Unit (How Module) for energy consumption and benchmark generation, and by using the Web Service Interface of Why Module for the forecast benchmark.
- Weather and Energy Price forecasts are computed internally to the Where Module by using external and open data sources.
- Generates detailed reports showing real-time energy trends and offers Charts and Tables enabling historical data evaluation.
- Provides a set of energy benchmarking indicators tailored on sport facilities

Where Module key benefits

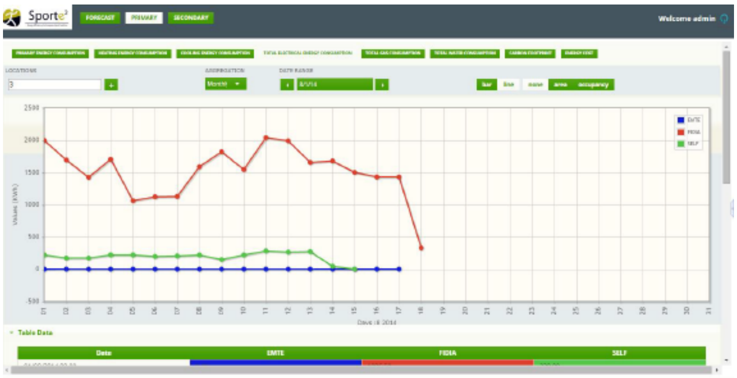
- Supports sport facilities owner/manager in finding similarities among different sport facilites through the selection of primary and secondary energy indicators.
- Provides a visual presentation of performance measures among different sport facilites through an easy to use Graphical User Interface (GUI).
- Measures efficiencies/inefficiencies among different sport facilites.
- Provides an overall view of all the plants localted in different geographical places.
- Support decision making thanks to the dedicated business intelligence.
- Facility owner or sport facilities manager can verify energy savings improvement outcomes from a building upgrade.
- Aligns strategies and organizational goal.
- To identify equipment malfunction or running unnecessarily or needed of service among the sport facilities.

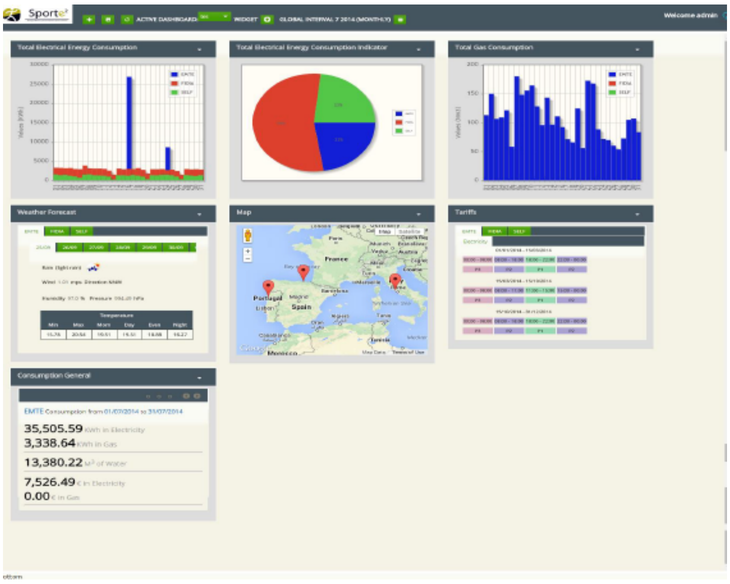
RINA 1861 D'APPOLONIA
consulting, design, operation & maintenance engineering

Graphical User Interface

The Graphical User Interface offers benchmarking capabilities by means of comparison energy situations and measures from different monitored and controlled facilities. The energy data presented by the SportE2 Where Module user interface is gathered from the SportE2 How Module and then elababorated in real-time for better supporting the benchmarking and decision making regarding energetic efficiency of the sport facilities.
Requirements:
- Computer or tablet with internet connection;
- Graphical Web Browser, supporting HTML5.

User Interface Functionalities





EMTE Consumption from 01/07/2014 to 31/07/2014

35,505.59 KWh in Electricity
3,338.64 KWh in Gas
13,380.22 M³ of Water
7,526.49 € in Electricity
0.00 € in Gas

Benchmark Monitor
Permit to monitor one benchmark at a time.

Dashboard
Permit to monitor different type of benchmark and indicator at the same time.

Tariff Management
Permit to edit and maintain tariffs for the facility in order to calculate energy consumption and costs.

Formula Management
Permit to edit and maintain the formulas used to retrieve and calculate benchmark indicators.

Background Synchronization
Synchronize the SportE2 Where Module database and the benchmark values with data sources available from the SportE2 How Module.

Administrative Inteface
Set of administrative pages that permit to access, edit, modify and configure the SportE2 Where Module.

Contacts

D’Appolonia S.p.a.
Via San Nazaro 19
16145 Genova | Italy

Donato Zangani
Email: donato.zangani@dappolonia.it
Tel: (+39) 010 36 28 148

RINA 1861 D'APPOLONIA
consulting, design, operation & maintenance engineering

This project has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 260124.