Smart energy tools for buildings energy management

George Suciu, Teodora Ușurelu, Carlos Jiménez

Abstract— Nowadays, due to population growth, the demand for energy resources has increased. In this respect, the consortium of the European project CitiSim has developed a platform for the smart city ecosystem. It will provide a powerful monitoring and control infrastructure for planners to make critical management decisions on tactical and strategic levels. Smart services included are focused on energy consumption and building management. We focused our work on developing two innovative solutions: Smart Energy for building management and Business Intelligence Tool. The Smart Energy for building service is a service designed to deal with energy-related data, using analytic and forecasting algorithms to provide KPIs useful for the strategic decision making within a building. The second service, Business Intelligence Tool, relies on energy production and consumption monitoring, energy production potential simulation, and simulation of the economic performance of energy efficiency projects for various investment scenarios.

Index Terms—energy consumption, energy production, smart tools.

I. INTRODUCTION

In recent years, several big countries have increased their energy consumption with 20% - 40%. From year to year in the UK, energy consumption increases by 0.5%, in Spain it rises by 4.2% annually, and in Europe, it exceeds 40%. The service sector includes public buildings (schools, restaurants, hospitals, etc.) and commercial buildings. They have a wide variety of uses of energy services (HVAC, lighting, powering electronics, computers, etc.). In the residential sector and beyond, energy consumption has been intensified mainly due to HVAC systems, which have become essential and are mostly used in the hot season or throughout the year in areas with very high temperatures or are used for heating or ventilation during the rest of the periods. Offices and retail buildings generate the most significant energy consumption. In Spain, energy consumption in buildings accounts for one-third of the energy consumption in the commercial sector and 3% of the total. In the UK it represents 17% of the energy consumption in the business sector and 2% of the total. In this sense, to help citizens and to contribute to increase the quality of life and reduce the energy consumption, within CitiSim project we have been developed two smart services related to energy: Smart Energy for Building and Business Intelligence Tool and Business Intelligence Tool.

II. RELATED WORK

Internet of Things technologies have grown more and more in the last few years, providing useful monitoring and management solutions especially for management over the utility operations. The use of sensors and actuators can reduce the amount of energy consumption at the building level only through quality management. The great advantages of using IoT technologies are highlighted in zero-net buildings (ZNEs) or zero energy buildings [1]. These are considered high-performance buildings, where the amount of the sustainable energy produced in the building is equal to the amount of energy consumed every year, with the help of Artificial Intelligence systems or solar systems [2]. Businesses, regardless of their size or field of activity, are looking for the most efficient solutions for reducing energy consumption.

Cisco Energy Management Suite is a tool that facilitates the use of energy by knowing the energy consumption, savings and prices. These data are collected using IoT devices and are displayed and classified by time and cost center (device, device location, group). Therefore, its applicability in terms of energy efficiency improvement is based on raising awareness of inefficient equipment and practices within the client company. This tool does not require the use of smart meters for data collection [3].

DEXCell Energy Manager [4] is a Cloud energy management platform focused on a time observance and energy analytics. This solution is based on periodic observation of energy consumption, consumption bills, widget-based dashboards, the invention of energy patterns, comparative analysis, analysis, reporting and notification. Using the API provided by DEXCell, users can make dashboards, alarms and reports made according to their preferences [5].

Teodora Ușurelu, Research & Development Department, Beia Consult International, Bucharest, Romania, teodora.usurelu@beia.ro
Carlos Jiménez, Research & Development Department, Abalia, Madrid, Spain, carlos.jimenez@abalia.com

This work has been supported in part by UEFISCDI Romania and MCI through projects CitiSim, ESTABLISH and PARFAIT, funded in part by European Union’s Horizon 2020 research and innovation program under grant agreement No. 777996 (SealedGRID project) and No. 787002 (SAFECARE project).

George Suciu, Research & Development Department, Beia Consult International, Bucharest, Romania, george@beia.ro, IEEE member

978-1-5386-8218-0/19/$31.00 ©2019 IEEE
Engage [6] is a software system that facilitates the observation and management of electricity. It is characterized by monitoring energy consumption, defining the budget, calculating prices, analyzing the energy demand and recording the energy consumption history. The platform uses proprietary sensors from Engage. The energy consumption forecast is based on the entries regarding the climatic conditions, the history of the previous consumption, the operating hours etc. [7].

DesignBuilder is an easy-to-use software that consists of a virtual building modeling environment. For the simulation of the energy consumption inside the buildings and for the forecasting of the production potential DesignBuilder is based on EnergyPlus. Moreover, the solution facilitates the modeling of the control systems by choosing the lighting system, solar shading, environmental conditions, thermal conditions inside buildings, carbon emissions, energy consumption, daylight illumination and maximum and minimum temperatures depending on the season [8].

III. SMART ENERGY FOR BUILDING MANAGEMENT

The energy consumption forecasting service is used to allow users and other systems to obtain predictions of the energy consumption of a whole building. To achieve this, the energy consumption forecasting service builds different behavior-predicting data models using machine learning algorithms and statistical processes.

The goal of CitiSim’s energy consumption prediction service is to allow for the prediction - with an acceptable degree of accuracy - the amount of energy that a building will consume in a specific date range. To accomplish this, we must first take a look at the data that can be used to feed the model. This existing historical data is crucial, as it allows for the identification of patterns that may be useful when predicting consumption.

One of the datasets used for the development of this prediction service is a log that has been collecting information from different sensors since the year 2011. This dataset has over 4 million data points containing information regarding the specific sensor ID, the timestamp when the measurement was taken, and the actual value of the amount of energy that was consumed.

First of all, we can rule out the family of classification algorithms, given that we do not want to classify information. Once we have chosen the family of algorithms, we can go ahead and choose the adequate one. In this specific case, we can observe a direct relationship between the time of the year and the amount of energy being consumed. Taking a look at the month, day of week and day of year variables that can be obtained from the sample timestamps, we can say that depending on the time of the year, and energy consumption will either increase or decrease. In winter months, consumption will probably increase due to the use of heating elements within the building. Additionally, taking a look at weekdays, we can assume that the consumption during the weekends will be much lower than consumption during the weekday depending on the nature of the building (office building, university building, etc.). These insights will prove to be very valuable when using the model.

The steps taken to create the model are: analyze a dataset, data preprocessing, identify training method and algorithms to use, implement, and use the model.

IV. ENERGY CONSUMPTION PREDICTION MODEL

We can say it is a direct relation between the weekday as well as the month of the year and the energy consumption of the building. We can generalize this concept into one that is more statistically suited and say that there is a dependent variable (energy consumption) which depends on multiple factors, also known as independent variables. In this case, the independent variables are the month of the year, day of the month, and weekday. With this generalization, we can conclude that we are working with multivariate linear regression.

Multivariate linear regression in machine learning will usually be trained by a vector composed of different “features”. In our specific case, the features are the columns of the dataset, so for instance, a single data point will have features detailing the month, day and weekday it was taken, as well as information regarding the sensor that made the sample. On the other hand, we will have the actual output that we want to predict, which is the consumption. To get a prediction out of our model, we will have to provide a vector in the same format as it was initially trained. So our prediction request must also be composed of a month, day, and weekday values in order to tell the model those values are what we are testing against.

The mathematical representation for a multivariate linear regression looks just like in the following equation:

\[ y = X_1\beta_1 + X_2\beta_2 + ⋯ + X_k\beta_k + \epsilon \]  

Where \(\beta_i\) are the coefficients associated with \(X_i\) and \(\epsilon\) is the error value between the observed data and the linear relationship. To better understand the previous formula, we can note that the \(k^{th}\) regression coefficient \(\beta_k\) represents the expected change in the dependent variable \(y\) for each unit change in the \(k^{th}\) independent variable \(X_k\).

These coefficients are determined during the training of the model, where values are adjusted to minimize the overall model error. However, the actual values of the independent variables first need to be transformed. Here, the concept of dummy regressors plays a significant role. The issue is if we translate the days of the week into values from 0 to 6, and the months of the year to an amount from 0 to 11 and days of the month to a value from 0 to 30, we will have bias issues in our formula. For instance, if we have a sample taken at day 5 and another taken at day 30, our model is going to put more weight on the sample taken at day 30 simply because it is larger than the value of 5, and this will affect our results negatively. What must be done in this case is to employ dummy regressors and formulate our
model with dichotomous variables. Dichotomous variables may only take one of two predefined values. In this case, our predefined values will be 0 and 1, and our feature vector will grow to accommodate features for every month, day of the month, and weekday, and our dichotomous variables will be used as Boolean values to determine the date of the sample. By using this approach, our model will not be skewed at the data points that have a naturally higher value due to their categorical variables (month, day, weekday).

V. HOW IT WORKS & DATA VISUALISATION

This section details the pipeline of different services involved in data distribution, how a transducer is announced, and how its data is consumed by specific service (Energy service). For better understanding, it is broken down into node configuration, creation of a virtual sensor, and information consumption/representation.

![Sensor node setup](image)

Fig. 1. Sensor node setup

The consumption sensors are advertised to the “Welcome Service” by providing their address (sensor-address). This “Welcome Service” consults the common-topic property of the sensor (this is a channel that will be common to all sensors that capture the same magnitude) in the Property Service property server providing the newly acquired sensor-address. In this case, the common-topic is Active Power. After this, the Welcome Service is in charge of obtaining the publisher associated to the common-topic sensor and asks the Wiring Service to add an Observer for this sensor providing its sensor-address and the topic-publisher for the common channel.

Once the observer is created, the Wiring Service assigns it to the sensor through setObserver. In this way, the sensor can know where to publish its events. From here, the consumption sensors measure the instantaneous power of the associated load and send messages every 15 seconds, including the measured instantaneous power value, the timestamp, and the sensor identifier.

The procedure is designed for a place & play philosophy in a CitiSim instance directly oriented to reduce time configuration & setup. After network join specific method (WiFi, Bluetooth, wired network), only the Welcome Server service needs to be configured (is specific and unique for a CitiSim instance). The Welcome Server will have a well-known address.

The final step is the power sensor sending its instantaneous power readings to a logical channel (“ActivePower” orange box in Fig. 1).

Energy Server, subscribed to the common-topic, receives notifications from the different consumption sensors and creates an associated virtual sensor for each of them. These virtual sensors will, in turn, go through the configuration procedure mentioned in the previous section, and after this, they will know where to publish their messages. In this case, the virtual sensors will post a measure derived from the events originated by the physical devices, notifying the energy consumed, for example, in the last hour instead of the instantaneous power.

The creation of virtual sensors, in this case, is for creating processed information from raw data coming from physical sensors. This procedure is the same for creating virtual sensors oriented for simulations/sandboxes for smart service testing, etc.

The Transducer Directory, which is an instance of the Persistence Service, is subscribed to the common-topics of the physical and virtual transducers so that it receives all their notifications and allows maintaining an updated list of all the sensors and actuators present in the domain as well as the latest information available about them.

On the other hand, if a visualizer service (3D, dashboard, etc.) pretends to represent these events visually, any visualizer should follow the next two steps:
- It needs to know which transducers are located in the zone that is going to be visualized to subscribe to them. First, it requests to geoHash Service a list with the identifiers of transducers associated with a place, providing the placeID. After receiving this list, it consults again, which is the exact location of each one of the transducers, providing for each request the corresponding transducerID. Also, for each transducer, it consults in the Property Service which is the associated asset for its representation;
- Already knowing the list of transducerID, the Visualizer consults the Transducer Directory which is the last known state of each transducer, in order to have a data to represent instead of having to wait for the next notification, and in addition, it makes a subscription request to the Wiring Service to subscribe to the upcoming notifications generated by each transducer.

In this way, the Visualizer already has all the necessary data to represent the information notified by the transducers in the specific location of each one of them.

The data visualization layer of the Smart Service uses a 2D Dashboard through a web page. This website uses the data collected in the database and formats it to provide the KPIs displayed using several widgets of different type (charts, maps, etc.). Filters can be used with those widgets to adjust the data visualization to our requirements.

The dashboard is implemented in Angular 7, and the charts are developed with the highcharts,js library. The backend of the application is built with node.js, using MongoDB as the database. The connection between the backend and the database is made with the Mongoose.js library. Authentication is required in the web interface to be able to visualize the information. Once logged in, a screen with the main dashboard appears, as well as a set of filters. There are additional views with specific widgets focused on Active Power, Energy, Power Factor, and Voltage.

Define less standard abbreviations and acronyms the first time they are used in the text, even after they have been defined.
In the abstract. Abbreviations such as IEEE, SI, MKS, CGS, ac, dc, and rms do not have to be determined. Do not use abbreviations in the title unless they are unavoidable.

See Appendix A of the Author’s Kit for additional information and standard abbreviations.

Use either the Microsoft Equation Editor or the MathType commercial add-on for MS Word for all math objects in your paper (Insert | Equation or MathType Equation).

To make your equations more compact, you may use the solidus (\(\div\)), the exp function, or appropriate exponents. Italicize Roman symbols for quantities and variables, but not Greek symbols use a long dash rather than a hyphen for a minus sign. Use parentheses to avoid ambiguities in denominators.

Number equations consecutively with equation numbers in parentheses flush with the right margin, as in (1). Be sure that the symbols in your equation have been defined before the equation appears or immediately following.

VI. BUSINESS INTELLIGENCE TOOL

In the past decade, energy efficiency has grown substantially. To deal with this problem, we created an intelligent energy consumption tool based on the historical data collected by Verbund sensors and stored in Grafana [9] platform for data analysis and monitoring.

The CitiSim Smart Energy BI platform relies on three functionalities:

- energy production and consumption monitoring, based on IoT devices, MQTT (Message Queuing Telemetry Transport) and Grafana [9];
- energy production potential simulation, based on EnergyPlus and weather data;
- simulation of the economic performance of energy efficiency projects for various investment scenarios.

Fig. 2 presents the IoT devices used for collecting data necessary for simulating the consumption of energy inside the buildings. For testing, we have used Verbund [10] energy sensors and Libelium [11] environmental sensors.

![Fig. 2. Energy monitoring tool architecture](image)

Verbund [4] IoT device represents an essential element for measuring energy efficiency. It records the desired parameters and transmits them further by accessing the JSON link address REST/JSON. However, this request can only be made if it is in the same wireless network. With the help of the Siemens [12] IoT 2020, the information received from Verbund [4] can be transferred via MQTT protocol to the Beia server with a precise sampling period. Using Grafana platform, the necessary parameters are displayed as a graph to facilitate understanding the events that take place at the inverter level.

Libelium [11] devices are used for monitoring the temperature, solar radiation, and luminosity. Data is collected and sent by the nodes (Waspemotes) to the gateway (Meshium) using various protocols (e.g., XBee, LoRa, GPRS, 3G, or WiFi), where it is stored. Meshium sends data to the Local MQTT Broker using MQTT protocol, and then to the Central MQTT Broker.

Through the MQTT Client, the information is being sent to the POSTGRESQL CitiSim Database. It retrieves data stored in the database and displays it in Grafana [9] dashboards as a GUI (Graphical User Interface).

For energy monitoring, the goal is to extract the output parameters of an energy sensor, in the current case, an inverter and send this information to a data visualization platform.

The data are used to determine the specific KPIs included in the platform. The KPIs are related to energy efficiencies inside a building, such as Internal Rate of Return (IRR), Net Present Value (NPV), and Return Of Investment (ROI).

The Internal Rate of Return (IRR) is a metric used in capital budgeting to evaluate the profitability of possible investments. The IRR is a discount rate, which makes the Present Net Value (NPV) of all cash flows from a particular project equal to zero. IRR predictions rely on the same formula as NPV does. The following is the equation for calculating NPV:

\[
NPV = \sum_{t=1}^{T} \frac{C_t}{(1+r)^t} C_0
\]

where: \(C_t\) is net cash inflow during the period \(t\); \(C_0\) is total initial investment costs; \(r\) is discount rate, and \(t\) is several periods.

Return on Investment (ROI) is a KPI, used to estimate the efficiency of an investment or to compare the effectiveness of different investments. ROI determines the amount of return on investment, relative to the investment's cost. For calculating ROI, the benefit (or return) of an investment is divided by the cost of the investment. The result is displayed as a percentage or a ratio. The return of the investment equation:

\[
ROI = \frac{\text{Gain from Investment} - \text{Cost of Investment}}{\text{Cost of Investment}}
\]

VII. HOW IT WORKS & DATA VISUALISATION

To use the BI tool, the first step is to create a user-defined scenario. For this, the user needs to click on the green plus sign in the bottom right corner of the screen.

A new window will be displayed, where the user adds data regarding the offer received from the ESCo (the value of the investment and the duration of the return of the investment).

By clicking the Plus sign in the top right corner of the scenario, the user can create a set of rules for his defined scenario.

To add the current bill (EnergyPLus simulation) and the baseline bill (consumption history mean), the user needs to click on the scenario so that he can add the values for each month of the contract. After this, the results and KPIs will be automatically calculated (see Fig. 3).
Entries can be edited by clicking on the Edit button. KPIs can be visualized in various forms, such as diagrams and graphs. For KPI forecasting, additional user input is required, such as the value of the investment, rules for the sharing of savings between involved parties and duration of the contract, etc. Each parameter is displayed as a chart.

CONCLUSIONS

The IoT and Cloud-based energy services aim to facilitate the management of energy resource within buildings and suggests some solutions and decisions that should be taken to decrease energy consumption. Also, it provides economic KPIs for the substantiation of investment decisions, thus enhancing the success rate of energy efficiency projects. As future work, the services will continuously be updated to provide more features for the users.

VIII. REFERENCES

[1] M.A. Hannan, M. Faisal, P.J. Ker, L.H. Mun, K. Parvin, T.M. Mahlia, F. Blaabjerg, “A review of internet of energy based building energy management systems: Issues and recommendations”, IEEE Access, Vol. 6, 2018, pp. 38997-39014.
[2] S. Attia, “Net Zero Energy Buildings (NZEB): Concepts, Frameworks, and Roadmap for Project Analysis and Implementation”, Butterworth-Heinemann, March 2018.
[3] Cisco. (2019). Cisco Energy Management Suite. Retrieved on March 2019 from https://www.cisco.com/c/en/us/products/switches/energy-management-technology/
[4] Dextmatech. (2019). DEXCell Energy Manager software. Retrieved on March 2019 from http://www.dexmatech.com/software/.
[5] K.M. Fowler, C.J. Anderson, and B.E. Ford, “Energy data management system commercial product summary”, No. PNNL-26693, Pacific Northwest National Lab (PNNL), Richland, WA, United States, September 2017.
[6] Engage. (2019). Engage platform. March on January 2019 from https://engage.efergy.com/content/about-engage
[7] S.O. Masebinu, E.T. Akinlabi, E. Muzenda, and A.O. Aboyade, “Techno-economic analysis of grid-tied energy storage.”, International Journal of Environmental Science and Technology, 2017, pp. 1-12.
[8] DesignBuilderSoftware. (2019). DesignBuilder software. Retrieved on March 2019 from https://www.designbuilder.co.uk/helpv4.7/.
[9] Grafana Labs. (2019). Grafana Dashboards. Retrieved in June 2019 from https://grafana.com/grafana.
[10] Verbund website. (2019). Retrieved in June 2019 from https://www.verbund.com.
[11] Libelium website. (2019) Retrieved on March 2019 from http://www.libelium.com/.
[12] Siemens website. IoT 2020 (2019). Retrieved on March 2019 from https://w3.siemens.com