UDC 004.62

V. Storchun, K. Mamedova

ENERGY AND WATER CONSUMPTION MONITORING SYSTEM FOR CO-OWNERS OF AN APARTMENT BUILDING

Annotation: The problem of collecting, analyzing and distributing data collected from energy and water sensors between co-owners of an apartment building is considered. A comprehensive solution is proposed, including automated monitoring and notification system, which allows to increase the efficiency of expenses management and improve reaction time in an emergency.

Keywords: data collection, data analysis, monitoring system, energy resource consumption, expenses management.

Introduction

The administrations of housing and communal services often do not provide apartment owners with sufficient awareness of the expenses for maintaining the building. As a result, we get: unclear reports on expenses, insufficient transparency of management processes, and a lack of mechanisms for involving apartment owners in building management, which in turn leads to inefficient use of funds for building maintenance and conflict situations between the administration and apartment owners.

To address this problem, it is proposed to establish more transparent management processes, ensure regular reports on expenses, and involve apartment owners in decision-making processes regarding building management by creating an automated system that provides user cabinets with distributed access rights according to a defined role model.

Statement of the problem research

The results of the study of existing analogs have shown that, despite what seemed to be a saturated market with Internet of Things (IoT) solutions, there were no systems found that would allow for automated collection of data on the consumption of already mentioned indicators in multi-apartment buildings.

Most of them are designed for individual apartments, large industrial enterprises, or cities, so using such systems is impractical because they do not address the specific issues of an average multi-apartment building. Some of them only work with electricity or heating. Some limit themselves to the idea of smart sensors and ignore the automation of further work with their readings. The results of the study are summarized in the table (tabl. 1).

As a result of analyzing the existing solutions available on the internet, the absence of consumer-oriented proposals for consumption monitoring systems in buildings was identified. Most of them operate on the scale of either a single apartment or an entire city or large enterprise.

Table 1.

Description of existing resources usage monitoring systems

System name	System description	Source
Schneider Home	Monitoring of primary and backup power sources, control of light switches and sockets, is still in the development stage.	[1]
Azure IoT Central	The software product, which requires customization for each user (installation of smart sensors, etc.), utilizes potentially vulnerable cloud services.	[2]
Thermostats		
Google Nest	The products allow for tracking and optimizing heating expenses. However, automation of monitoring electricity and water usage is not implemented in them.	[3]
Hive Thermostat		[4]
OpenWay Riva	The system is designed to operate on a smart city scale, which, in the context of a multi-apartment building, includes many rudimentary tools as well as lacking instruments	[5]
numerous SCADA systems	None of them are oriented towards use by associations and ensuring transparency towards apartment owners.	[6][7][8]

The intermediate link of a multi-apartment building currently remains unnoticed and underserved by IoT solutions. Therefore, none of the available products:

– offers systematic data collection and processing at the levels of apartment-floor-building;

– provides detailed information to apartment owners about the efficiency of their expenditure;

– includes an intelligent emergency notification system for emergency situations.

Problem solution

This section describes the proposed system for monitoring energy and water consumption for co-owners of a multi-apartment building.

The hardware part is implemented using a network of digital (or, if necessary, advanced analog) sensors and mini PCs like Raspberry Pi (RPI).

Unlike other controllers, RPI can be quickly and conveniently set up, it allows for efficient initial data analysis, enabling timely detection of abnormal situations [9].

ISSN 1560-8956

RPI controllers send requests to sensors for readings, perform initial analysis of the obtained results, and send the data to the server application.

Communication between RPI and the server is carried out via a local Ethernet network. The server application informs users about sensor and meter readings. In case of emergencies and abnormal situations, the application notifies the relevant users [10].

If analog sensors are used in the building, digitization modules are employed, such as cameras with illumination, whose images are analyzed by the server application. The digitization module is powered only at the moment of reading the readings. Digit recognition on photos is carried out using existing Computer Vision [11] software tools.

Fig. 1 depicts the sequence of interactions among system participants in a typical apartment. Here, the term "Sensors" refers to heat consumption meters, cold water consumption meters, hot water consumption meters, electricity consumption meters, smoke detectors, heat flow sensors, and security sensors (doors, windows, volume).

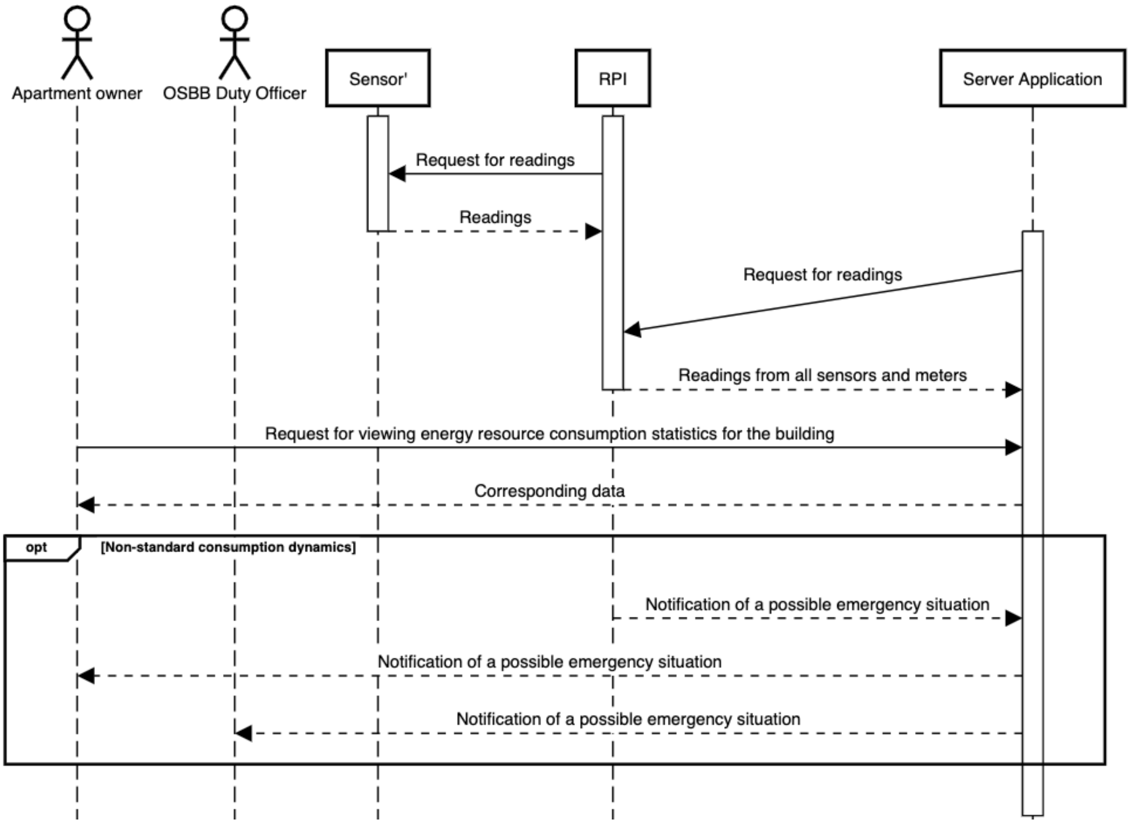

Figure 1. The sequence of interactions of the system at level of a single apartment

Fig. 2 depicts the sequence of interactions among system participants in the entire building.

Fig. 3 depicts main nodes of the monitoring system.

At each nodal point (each apartment, floor), there is an RPI designed for data collection, initial analysis, and sending data to the server.

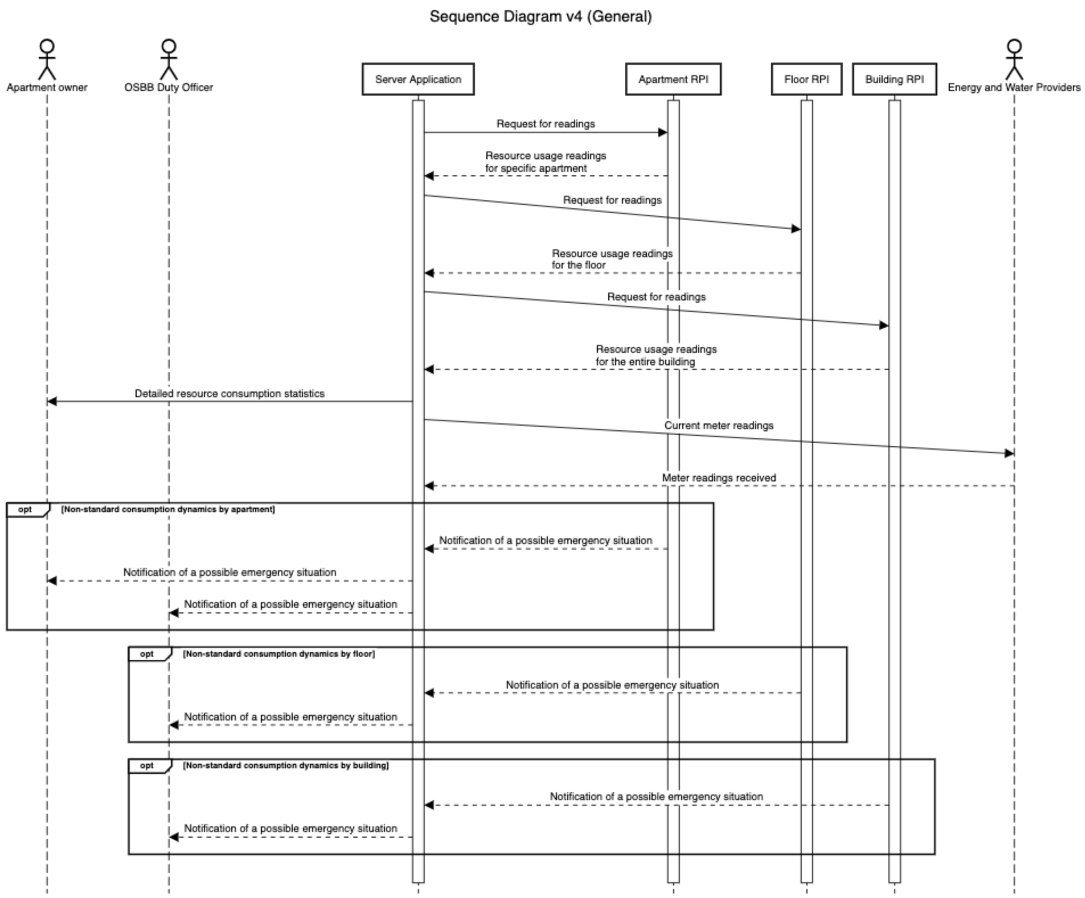

Figure 2. The sequence of interactions of the system at level of entire building

The sensors of all apartments, floors, and the building as a whole are interconnected into one system, allowing for prompt access to comprehensive data on water and energy resource usage, as well as for identifying and collecting information about emergencies.

The server application is unified for all users. Using an external SMTP service, it sends out emails with reports and alerts about emergency situations.

The administrator can control the network using either a device (laptop, desktop computer, tablet, etc.) connected to the local network or using pfSense from any other device.

The equipment proposed for use in the monitoring system includes:

1. Power consumption monitor 80-260V 100A PZEM-004T UART that:

- measures:
  - current intensity(0-100 A),
  - voltage (80-260 V),
  - frequency (45-65 Hz);
- calculates power based on their product (0-22 kW);
- accumulates data on power consumption over time;
- uses serial UART interface with a speed of 9600 (5 V DC).

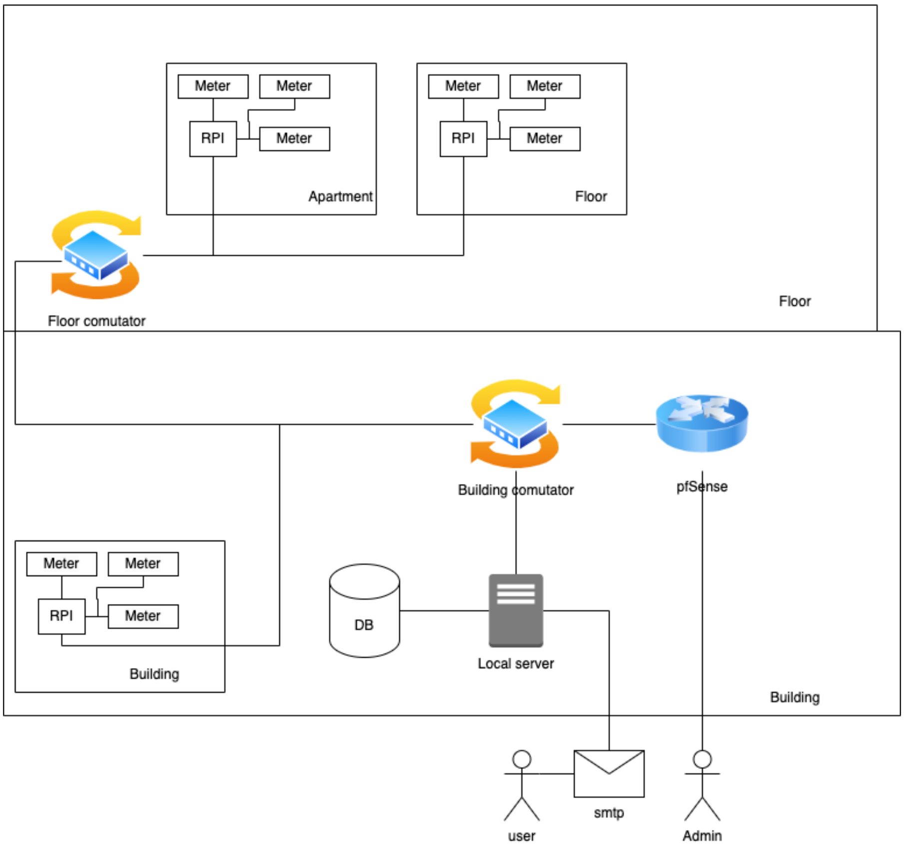

Figure 3. Structural scheme of monitoring system main nodes

2. Temperature sensor DS18B20 that:
   – measures temperature in the range from -55 to +125 °C;
   – has accuracy as 0.5%;
   – uses 1-Wire interface.
3. Gravity: Analog Water Pressure Sensor that:
   – has analog water pressure sensor (0-1.6 MPa);
   – has accuracy as 0.5-1%;
   – supports Gravity 3-pin interface;
   – outputs voltage (up to 5 V) and requires external power;
   – uses an analog-to-digital converter (ADC) to obtain digital data.
4. Smoke sensor Arton SPD-3.10.

Connection of the SPD-3.10 sensor to the control and measurement equipment (CME) with two-wire CS is carried out using the B01, B1 bases.

5. Ultrasonic heat meter CEM CM-HR DN 15 0.6 m3/h which has:

– ultrasonic heat energy meter with flow ultrasonic sensor for heating and/or cooling systems.

– data transmission interface: M-Bus.

– nominal size: DN 15 qp=0.6 m³/h.
– temperature of the heat carrier: From 4 °C to 95 °C.
– accuracy class according to DSTU EN 1434: 2.
– M-Bus interface.

6. IP camera D-Link DCS-2132L.

Local Network

The system uses huge variety of sensors and meters produced by different manufacturers. Therefore RPIs collect data utilizing diverse protocols. On the other hand data transition from RPIs to the server is conducted using Ethernet.

User notification is done implementing third party SMTP service, which receives data applying HTTPS protocol. Figure 4 depicts network diagram of the monitoring system.

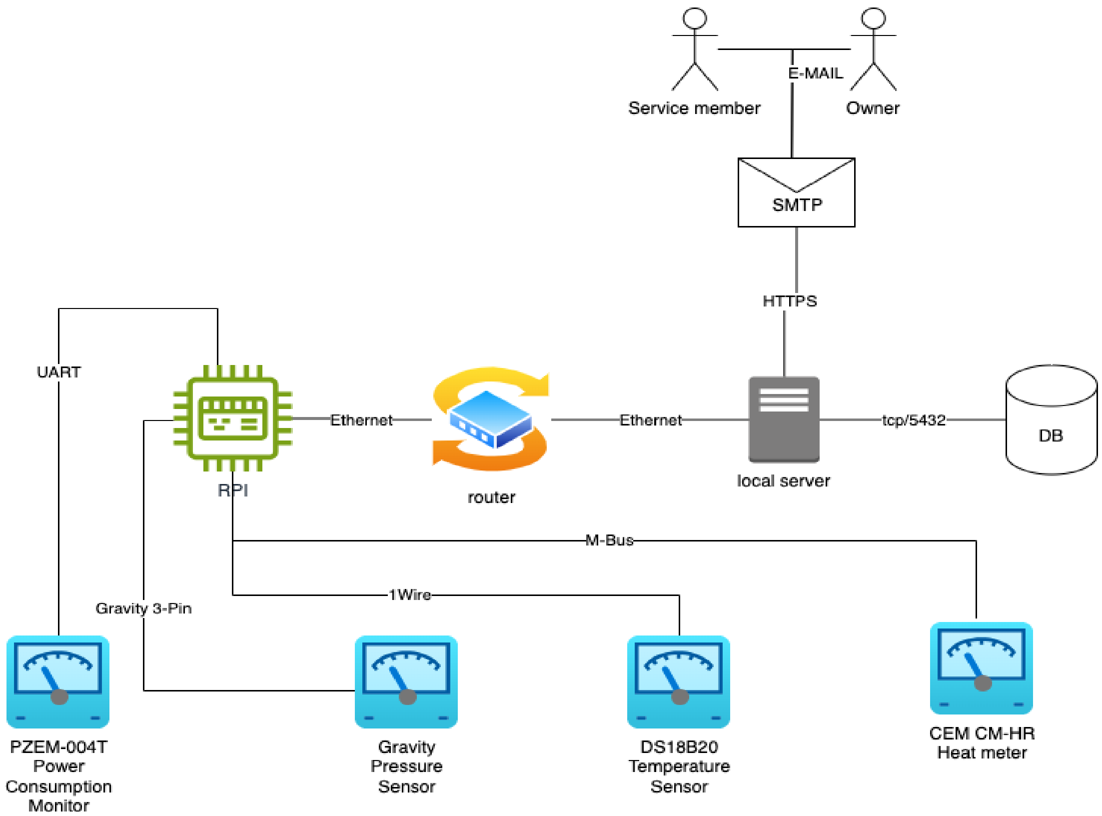

Figure 4. Networking diagram

Conclusion

The issue of monitoring resource usage in multi-apartment buildings is highly relevant. Automation of consumption monitoring processes contributes to the transparency of operations processed by housing and communal services administration and efficient cost management.

 ISSN 1560-8956

The article describes a proposed solution for implementing a hardware and software monitoring complex that automates the collection, analysis, and dissemination of data read from sensors, ensuring apartment owners have comprehensive access to consumption data and confidence in the usage efficiency of their funds. As a result of using the monitoring system, automatic analysis of indicators occurs, ensuring timely detection of potentially hazardous situations and subsequent notification of the relevant users.

REFERENCES

1. Schneider Home. URL: https://shop.se.com/us/en

2. Azure IoT Central. URL: https://azure.microsoft.com/en-us/products/iot-central#overview

3. Google Nest. URL: https://store.google.com/gb/product/nest_learning_thermostat_3rd_gen?hl=en-GB

4. Hive Thermostat. URL: https://www.hivehome.com/shop/smart-heating/hive-thermostat

5. OpenWay Riva. URL: https://vallin.lv/wp-content/uploads/2017/10/openway-riva-generic-brochureweb.pdf

6. OneView SCADA. URL: https://scada-international.com/oneview-park-scada/

7. Microgrid Controls. URL: https://process.honeywell.com/us/en/industries/renewable-and-energy-storage-solutions/microgrid-controls

8. Home SCADA. URL: https://homescada.com/

9. Rui Santos, Sara Santos - SMART HOME with Raspberry Pi, ESP32, and ESP8266. URL: https://randomnerdtutorials.com/smart-home-ebook/

10. Rui Santos, Sara Santos - ESP32 Web Server with Arduino IDE. URL: https://randomnerdtutorials.com/learn-esp32-with-arduino-ide/