

**I. Klymenko, A. Gaidai, S. Nikolskyi, V. Tkachenko**

**THE ARCHITECTURAL CONCEPT  
OF THE MONITORING SYSTEM  
ON THE BASIS ON A NEURON MODULE  
IoT DATA ANALYTICS**

*Abstract:* The purpose of this article is to increase the efficiency of intelligent IoT systems by improving their architecture and the use of the means of IoT analytics based on trained neural networks. An architectural concept of a sleep monitoring system based on the IoT analytics neural module is proposed. The proposed hardware-software tools based on intelligent analytics implement a way to determine the comfortable of each user's environmental parameters based on the recommendation system for configuring it. A test experimental prototype of the sleep monitoring system was created, based on research, the effectiveness of the proposed solutions is experimentally justified.

*Keywords:* IoT monitoring system, NodeMcu, neural network, random forest, distributed computing, microclimate, Zabbix, machine learning.

**Introduction**

The modern level of software development, the high technological level of hardware, in particular the ultra-high performance of modern processors and the expansion of their functionality, the increasing power and speed of network facilities, in particular the modern Internet, contribute to the widespread proliferation of sensors and sensor tools connected to the Internet, which provide observations of the surrounding world, collect and measure data. The modern Internet of Things (IoT) is billions of smart things that communicate with each other, produce and collect a huge variety of data. The analytics of this huge data becomes a critical tool, which today requires the search for new methods and means of improvement.

Today, the current reasons for such a search are being considered: IoT Analytics is based on data and anomalies, a huge amount of collected data and created information needs to be stored and used in the future, as well as providing a quasi-real-time mode for their processing; a large amount of network traffic is transmitted by the Internet and also often needs to be processed in a Time Mode close to real; significant problems are created by unexpected attacks and intrusions of IoT, which also require advanced detection solutions.

In a word, the meaning of the problems is the development of the intellectual and applied possibilities of the analysis of these IoT solutions, it is better to allow the IoT paradigm to be developed today for the people. Neural networks and other intelligent means of processing large amounts of data expand the capabilities of IoT applications and are

discussed in modern sources as the most relevant and promising way to improve the efficiency of IoT data analytics. Neural networks and deep learning are described in many projects related to IoT and data analytics in IoT and show advanced results. Thanks to its methodology based on data and anomalies, as well as the ability to detect development, the issues of intelligent data processing and intrusion detection are being solved.

The article is devoted to the development of an architectural concept of software-hardware for the implementation of a monitoring system based on intelligent IoT data analytics using trained neural networks. The monitoring system is designed to monitor a person's sleep and adjust to this comfortable environment, and is useful both in a person's daily life and in professional medical institutions for diagnosis and treatment.

### **Setting the task**

The use of well-known devices for monitoring human health indicators, from smart mattresses, smartwatches or fitness trackers, built-in environmental monitoring systems that communicate with the user through the application are separate devices that require manual control, independent analysis of the data obtained and decision-making. To create comfortable conditions in automatic mode, it is necessary to control a large number of parameters and constantly monitor the user's condition, on the basis of which it is necessary to form control actions to adjust comfortable conditions, as well as to form certain conclusions and tips. It is also advisable to store the obtained data for further analysis and study of the human condition, as well as, if necessary, transfer it to medical institutions for monitoring.

The bottleneck for solving the task is the complex processing of data from many sensors, which requires decision-making based on the data received; feedback in real-time mode; availability of data warehouses for monitoring and further processing; the need for dynamic scaling of the system to connect new sensors and increase the number of users, creating for them different scenarios; the need to protect users' personal data and a large amount of network traffic, which reduces the efficiency of using cloud data analytics services; the cost of software-hardware; the cost of computing on cloud services. We propose an architectural concept of a monitoring system, which, through the use of a trained neural network for IoT data analytics and the IoT boundary computing paradigm, allows overcoming the above-mentioned limitations of data processing efficiency and personal information security, a combination of inexpensive hardware and affordable wearable electronics aimed at reducing the cost of accessibility of the system for a wide range of users.

The article examines the processes of data collection and analysis in intelligent systems, which cause the problem of increasing the efficiency of their processing in real-time, solved by using artificial intelligence in the form of a neural network and machine algorithms for training networks. The subject of the study is the architectural concept of intelligent systems based on boundary computing and means to improve the efficiency of sleep monitoring systems using machine learning algorithms and neural networks.

### **Overview of Eastern solutions**

There is a large number of works in which various methods of sleep quality research are analyzed, which include the development of either specific systems or special devices that will be engaged in monitoring. The work [1] is based on data analysis using neural networks representing the stages of human sleep to detect narcolepsy. With manual data processing takes a lot of time, there is an error probability of up to 18%, the procedure is expensive and requires spending a significant amount of time in a medical facility. The proposed method based on the use of neural networks makes it possible to diagnose the disease much faster and at a lower cost, reduces the time for initial diagnosis, while maintaining high accuracy.

The authors of the article [2] propose solutions for the detection of apnea in humans, based on IoT. The value of this solution for humanity is that - it is a disorder that is expressed in problems related to human breathing and people of different ages suffer from it, so effective diagnosis is quite relevant. The main parameters that the system analyzes include blood oxygen saturation, body mass index, sleep time, blood glucose level, age, blood cholesterol level, and others. The authors of the article compare different solutions in the eastern region, but the use of neural networks is new, which made it possible to implement data processing in real-time. This allows to notify medical staff or family that a person needs emergency care. The purpose of the article is a hardware and software module based on Arduino UNO, an analog sensor system and a mobile application.

Article [3] describes a system for monitoring sleep quality, a system that consists of three sensors: a microphone, an accelerometer, and a pulse oximeter. A method of diagnosis using machine learning is proposed. The authors used a controlled classification algorithm – a random forest. The collected data after processing can be transferred to doctors so that they take measures in accordance with the patient's condition. Using the random forest method, the authors were able to achieve an accuracy of 95.67%. According to the authors, their development is inexpensive due to the available sensors, which allow it to be used at home or in a hospital, and this system has high accuracy. The system consists of a sensor system, an Arduino board and a server on which data is processed.

Here, we consider developments related to human sleep monitoring, which analyze data using neural networks. A review and analysis of the studied tools presented by the authors of the articles substantiating the conclusions about improving the efficiency of data processing by increasing the speed, accuracy of calculations, reducing the cost of diagnostics and the cost of smart devices in the field of IoT through the use of intelligent analytics.

### **Methods and technologies used**

During the research and the creation of an experimental sample of the system, the following data were collected: to assess the human condition – pulse, sleep stage; to assess the

state of the environment – air quality, temperature, light level, humidity. To measure a person's condition, it is advisable to use fitness bracelets or watches, they are convenient for human use and do not require the addition of additional sensors to the system. To assess the quality of the environment, special sensors are used, from which the module aggregates and analyzes information. The module is designed to collect data from different users and IoT sensors, so the sensor complex is not integrated directly into the processing module, but is implemented in separate modules. In the future, the set of sensors can be changed or expanded. The processed data is transferred to data storage centers, which are located, for example, in medical institutions and are available for use.

For data processing at the level of the analytical module, an artificial neural network is implemented whose task is to analyze input data, as a result of which the user should receive forecasts and advice on further actions, for example, changes in temperature, light level, and the like.

The work of a complete neural network is reduced to entering variable input data and obtaining the initial result after calculations. The chosen architecture of an artificial neural network is when neurons are placed on columns, so that a neuron of column  $n$  can be connected to neurons only from columns  $(n-1)$  and  $(n + 1)$ , thus an artificial neural network is an architecture of interaction of various neurons. Each neuron performs a classical data processing algorithm [4, 5]. The neuron adds the value of each neuron from the previous column to which it is connected. Before adding, this value is multiplied by the variable «weight», which determines the connection between two neurons. Each connection of neurons has its own weight, this value changes during the learning process. To total calculated value is added to the offset value. This value is selected before the network training stage. After that, the neuron applies the "activation function" to the resulting value. The activation function is performed using the sigmoid function [4]. Next, the neuron transmits the received value to the following columns and eventually data is generated that are the results. The neural network is being trained. For training, it is necessary to prepare a large amount of data that needs to be transmitted to the neural network. A controlled machine learning algorithm has been selected for neural network training *Random Forest*. The *Random Forest* algorithm is designed to process a data set containing continuous variables, both in the case of regression and categorical variables, and in the case of classification. He gives the best results for classification problems [6, 7].

To create a prototype of the software-hardware of the sleep monitoring system, the following sensor complexes were used [8, 9]: the *MQ2* gas sensor, which is mainly used to detect gases such as methane, butane, smoke, and others. Humidity and temperature sensor *AM2320*. This is a small sensor, but unlike the classic *DHT* sensors, it has an interface *I2C*. Temperature sensor *DS18B20* with data output in digital form. Due to the presence of a unique serial code for each *DS18B20*, it is possible to control many such sensors that will be

processed at one output of the board, which ensures the measurement of temperature values over a large area. Light sensor module with threshold comparator.

For data aggregation, a *NodeMcu* board based on *ESP8266*. This device can be used to host a small website with minimal functionality, or to transmit/receive data via a local network or via the Internet using *Wi-Fi*. The neural module of IoT analytics is implemented on a computer *Raspberry Pi*.

The developed architectural concept is based on the current paradigm of marginal computing, this will reduce feedback time and will save bandwidth. We have followed the path of creating modular neural networks and in the article, we describe the architectural concept of the IoT system, which is based on the use of a neural module for IoT data analytics. In the future, the modular neural network can be supplemented with modules for monitoring performance, security, and data protection, depending on the requirements of the system being developed

### **The architectural concept of software-hardware**

The developed architectural concept describes solutions, algorithms, models, and software to be used in each component of the system.

***Initializing the system and collecting input data.*** When registering in the system, the user fills out an electronic form containing questions to the user about the state of his health and comfortable conditions, in his opinion and preferences. The personal data is used as input parameters for the initialization of the system and is used in combination with the initial sensor data. The sensor system is connected to a data aggregation device in a local network. The software on this device consists of a module for aggregating indicators from sensors and sending them to the neural module of IoT analytics, a web interface module that performs the function of registering a client and giving him personal identifiers. Personal identifiers are used to protect user data. Personal information is encoded by a unique personal identifier, which is available only at the level of the data aggregation device in the user's local network. On the upper layers of the system, all data is accompanied by a personal identifier, which can be decrypted only at the request of the user, for example, during communication with a doctor in a medical institution. The data aggregation module is available for all devices in the local network and is connected to the IoT neural analytics module, which can be placed both in the local network and outside it. Thus, according to the proposed concept, several data aggregators can be connected to the same neural module of IoT analytics.

Thus, the components of the system are data aggregators, a mobile application for the data aggregator module that performs the function of registering in the system and issues a unique user ID, a neural IoT analytics module that processes data from aggregators, makes decisions and creates tips for system users, the main server for general data management and storage. The general structure of the monitoring system is shown in Fig. 1.

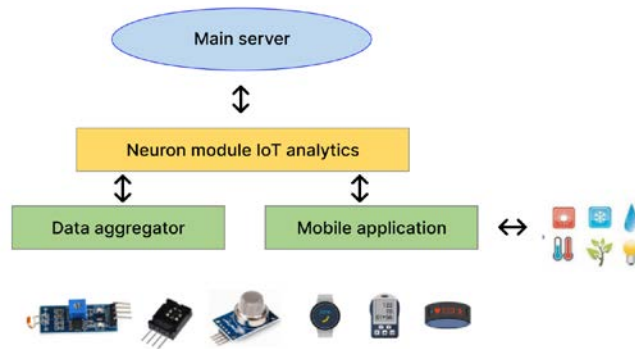


Figure 1. Block diagram of the monitoring system

Data aggregators, mobile applications, IoT analytics modules, and the main server are connected in the form of a tree. Neural analytics modules exchange data with the main server, and all data aggregators and mobile applications exchange data - with analytics modules. The mobile application expands the functions of data aggregators, it is used to collect information from fitness trackers, smart-watches, and the like, and also collects user reviews about the current state of the microclimate. Through the mobile application, the user can transfer additional parameters and settings to the neural module, for example, the user can transmit information about the ideal temperature for him and the data is recorded as a reference. The algorithm of the interaction of the data aggregator with sensors is shown in Fig. 2.

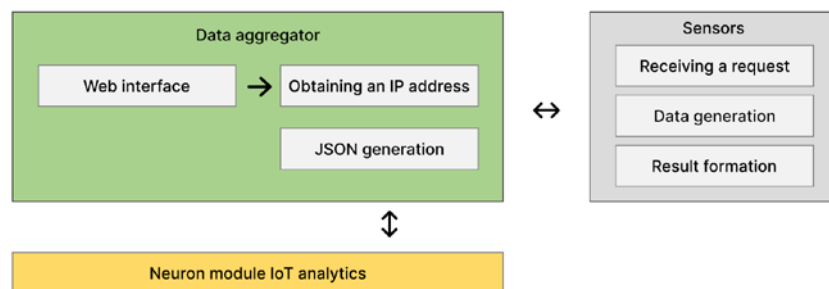


Figure 2. Component interaction diagram

### **Mobile application**

To get data from fitness trackers or smart-watches, a *Bluetooth* module is used, and since it is still necessary to provide data regarding the current state of the user, this can be combined in a mobile application. So the application should include several parts, each of which performs certain functions.

The mobile application in the monitoring system performs the following functions:

- user identification in the system - login and password are sent to the mailbox specified in the questionnaire;
- data exchange with a fitness tracker (or other wearable electronics); the app runs in the background;
- transmitting information about errors and incorrect measurements to the user;

– support of the questionnaire in Google forms for collecting data on the current state of the user and his preferences or doctor's recommendations; the data specified in the questionnaire are received by the neural IoT analytics module, which is combined with data that comes from sensors; the user has the ability to update and transmit the questionnaire data to the analytics module; the parameters in the questionnaire for setting the appropriate settings can be used by a neural network as a reference.

– formation of a data package in *JSON* format that will be transmitted for further processing; visualization of the results of analytics and decision-making on environmental parameters to improve the user's comfort; feedback is implemented in the form of notifications to a mobile phone and in the interface of a mobile application;

**The neural module of IoT analytics.** The neural module of *IoT* analytics is implemented on a computer *Raspberry PI*. The *Linux* operating system is deployed on the device, a storage for storing data of weights and network settings, and a software module for analyzing and generating data packets of each user. An agent *Zabbix* is used to monitor device resources and collect data for the main server. The neural data analytics module can be placed as a local network with a data collection system or in an external network, in which case we can receive input data from various users. *Zabbix* server tools are used to determine the current location of the neural module of *IoT* analytics using the map widget. The processing of questionnaires takes place using a script on *Python*. The results of the processing device are transmitted by the *Zabbix* agent to the main server, where they are stored in a database. The neural module of *IoT* data analytics is a key element of the system that implements intelligent data processing, decision-making based on the work of a neural network. In accordance with the scheme-algorithm of functioning (Fig. 3), the analytics module performs the following functions: accepts file names, loads previous weights for the neural network, generates a decision tree with optimal parameters, collects data and analyzes it, saves the analytics results and transmits a copy of the received data to the main server.

**The main server.** The main server monitors all components of the system, in particular the neural analytics module, processes data from registration forms, provides information to users, stores user data, stores the results of processing user data, works with questionnaire data. The main server is implemented as a physical machine, or as a cloud server.

To retrieve data from *Google Forms*, the main server runs a program that accesses the *Google Tables* service via *API*, compares new form data with existing data, and updates it as needed. The server generates file names according to the following scheme: get username, add last name, add first name, add age, add the number of the analytics module that the user received during registration. Data is sent to the analytics module and captured for agent-level monitoring *Zabbix*. The main server generates a random password to log in, the filename is used as a login. The login data is sent to the user's mailbox. In an experimental prototype of

the system, a mail server has been created at the main server level for communication, with the use of cloud technologies, it is possible to use cloud mail services.

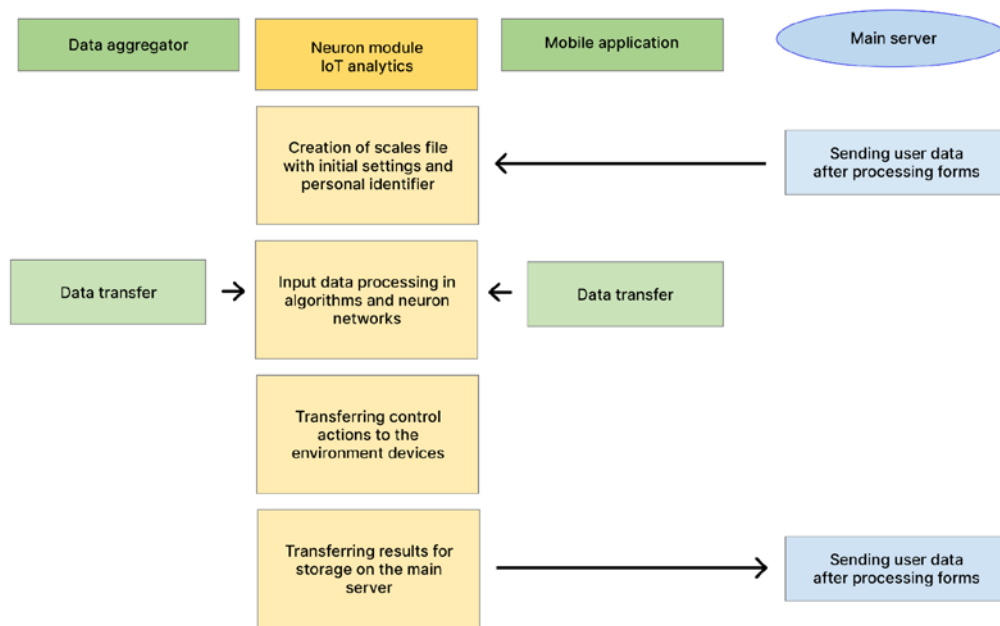


Figure 3. Interaction diagram of the processing device and other components

### Discussion of results

Data from the *Google Forms* questionnaire can be obtained in several ways using the developer's company *API*. To minimize calls to *Google* services, reduce the amount of redundant information transfer and deprive manual management, we went the way of getting data from *Google Tables*, instead of transferring all *Google* form responses. Although, this led to the need to support a separate module of the system. We use the information obtained to form a unique file name (full name, age, gender, unique identifier that the user sends to the form). After processing the form, the master server passes the file name to the analytics module according to the user ID.

We used standard libraries to work with sensors and the web interface. Entering the *IP* address of the analytics module is implemented through a web server in the local network at the level of the data aggregator (Fig. 4, 5).

The data is sent once a minute. To do this, the data string is formed in the form of *JSON* (fig. 5). In each measurement transmission session, the ID number of the analytics module is transmitted.

The mobile app sends data to the analytics module. On the first page of the application (Fig. 6.) you must enter the registration data, on the next page (Fig. 7.) the user fills out the questionnaire data to configure or adjust the monitoring system.



## Enter IP addresses of neural network server

IP:

Figure 4. Data collector web page

```
20:14:07.973 -> SN:D000023134, Temperature:23.81, Humidity:44.40, Gas:42.00, Light:102  
20:15:01.459 -> SN:D000023134, Temperature:23.88, Humidity:44.10, Gas:42.00, Light:102
```

Figure 5. An example of a message that is sent to a processing device

device IP

---

login

---

password

---

Figure 6. The first tab of the application

Is the current temperature comfortable?

Yes  No

Is the current temperature comfortable?

Yes  No

Is the light level acceptable?

Yes  No

Is there a problem with breathing?

Yes  No

Feeling sleepy?

Yes  No

I'm going to sleep

Figure 7. Page of the questionnaire for data correction

In the test mode of the study of the experimental model of the monitoring system and the training mode of the neural network is recommended to the user and can be performed at any time at will, while the monitoring system corrects the data and predictions. The item "I'm going to sleep" helps to determine the time of the beginning of sleep, to reject false positives fitness tracker, because there is a problem when a person is lying down and for example reading an electronic book bracelet can perceive this as a state of sleep. In the background, the

mobile app polls the fitness tracker and generates data in *JSON* format (see Fig. 5), which it sends for averaging and adjusting predictions. Once a satisfactory level of prediction is established, the mobile app switches to automatic operation without questioning or user participation.

We implemented an analytics module on a *Raspberry Pi* computer, the board accepts data in *JSON* format from the mobile app and from a data aggregator. The input data processing module is an input stream parser that passes the input data to the neural network for processing. The data processing technique and the network learning algorithm are beyond the scope of this article. We only want to generalize the processing algorithm with the following steps: loading the weights of the user network and the scaffolding array; feeding the algorithms with the corresponding data that were received from the collection devices; data processing and saving the results in the appropriate files. The original query results are read by the main server. At the stage of training the neural network feedback to the mobile application is implemented through the main server, but after a long training the functional ability to transmit predictions and advice to the user directly from the level of the analytics module is implemented, which corresponds to the paradigm of boundary computing.

A prototype of the analytics module with connected sensors is shown in Fig. 8. The disadvantage of the *NodeMCU* board is only one analog output, so we implemented a serial connection of sensors to the output by sequentially supplying power to the sensors and sequentially reading information (Fig. 9). If the number of sensors is increased, it is necessary to use sensors with digital data transmission channels.

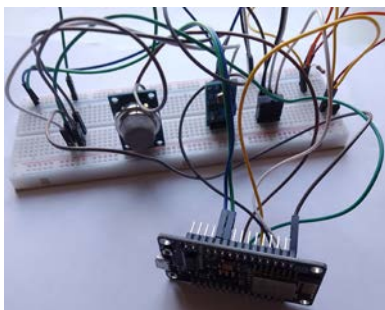


Figure 8. Test sample

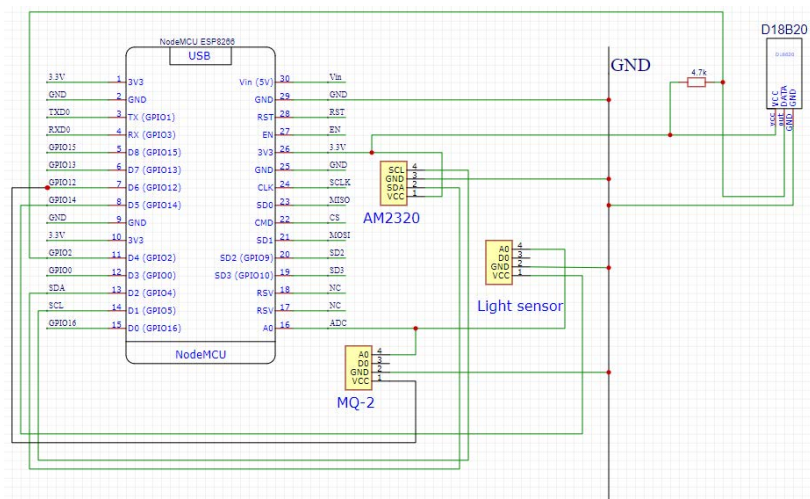


Figure 9. Functional diagram of the data aggregator

If you generally pay attention to the components, they are not expensive sensors that can be obtained in large quantities. The *NodeMcu* board can also be obtained in large enough quantities and its price is not great, while the *Raspberry Pi* is relatively expensive equipment, but it can be replaced with a personal computer, or another device with enough power for computing

and data storage. The sensor *AM2320*, although it has the ability to measure temperature, but taking into account the research carried out revealed that the temperature readings fluctuate quite a lot from the current, *D18B20* sensor, which is more reliable temperature measurement. There is a problem with selecting a light sensor, because the selected sensor cannot transmit data other than zero and one via a digital pin, this creates difficulties in its use.

As a development of the proposed architectural concept, we consider an architectural solution where the analytics module is placed in a local network with a sensor system and data aggregator. Such solutions are designed to implement additional functionality, which is associated with the automatic feedback system, and implement on its basis the control actions on the equipment that maintains the state of the environment. Thus, using the infrared transmitter on the board can be implemented control of air conditioners, lighting and other devices. The analytics module can be built into the "smart home" system and means the *Zabbix* agent can monitor the status of existing equipment in the home. This architecture reduces the amount of data that are transmitted to the main server, it is possible to connect many sensors, there may be other solutions to improve the efficiency of the system, which are based on the use of local network technology that will rid the system of redundant software on the main server, in particular data parsers.

### Conclusions

In this paper we proposed an architectural concept of the monitoring system, which through the use of a trained neural network for *IoT* data analytics and the paradigm of edge computing *IoT* allowed us to overcome the limitations of data processing efficiency associated with the use of cloud services *IoT*, and the combination of inexpensive hardware and affordable wearable electronics aimed at reducing the cost of accessibility of the system to a wide range of users.

Implementation of the prototype allowed to confirm of the performance of the proposed architectural solution. Edge computing technology made it possible to use virtualization technology and run services on peripheral nodes. The proposed architectural concept allowed to move the time-critical calculations from the traditional level of cloud services to the end-user network level, which provided a guaranteed speed of decision-making, the implemented solutions allowed to cache data, which also resulted in increased data processing speed and reduced response time.

At the same time, experiments have shown that the distribution of logic across different nodes in the network creates new problems related to data security, which now does not depend on cloud services, and limited device resources, which also prevents the deployment of data encryption mechanisms to ensure a high level of security. The proposed architectural concept offers solutions for security at the level of conventional means, which are used at the local network level. But this question requires further research.

The monitoring system is designed to monitor a person's sleep and adjust for this comfortable environment, and is useful both in a person's daily life and in professional medical institutions for diagnosis and treatment. The operation of the prototype showed that the two ways of placing the analytics module offer advantages and disadvantages to the end users. Implementation of the system to connect many users requires the implementation of additional software on the main server during the training of the system, otherwise the analytics module is able to implement control actions on the external environment, affecting the saving of hardware and energy resources for the implementation of «smart homes».

The system of neural networks, conceptually supported by the proposed solution, it will enable future scaling problems in a distributed network and the management of equipment failures.

The proposed architectural concept of bringing analytical computing resources closer to end users was able to implement complex analytical tools of artificial intelligence based on a neural network on the periphery of the system, which theoretically causes an increase in the response speed and throughput of the implemented monitoring system, compared with the time of cloud computing. This architecture aims to increase the operational efficiency of the system.

## REFERENCES

1. Neural network analysis of sleep stages enables efficient diagnosis of narcolepsy / Stephansen J.B., Olesen A.N., Olsen M. et al. // Nat Commun. – No.9, Vol. 5229. 2018. <https://doi.org/10.1038/s41467-018-07229-3>
2. Development of an IoT-Based Sleep Apnea Monitoring System for Healthcare Applications / Abdur Rab Dhruba, Kazi Nabiul Alam, Md Shakib Khan, Sami Bourouis, Mohammad Monirujjaman Khan // Comput Math Methods Med. – 2021. <https://doi.org/10.1155/2021/7152576>
3. IoT Healthcare: Design of Smart and Cost-Effective Sleep Quality Monitoring System / Khizra Saleem, Imran Sarwar Bajwa, Nadeem Sarwar, Waheed Anwar and Amna Ashraf // Journal of Sensors. – Vol. 2020, Article ID 8882378. <https://doi.org/10.1155/2020/8882378>
4. Nielsen M.A. Neural Networks and Deep Learning. – Determination Press, 2015. – p. 211. [Electronic resource] <https://academia.edu>
5. Ciresan D., Giusti A., Gambardella L., Schmidhuber J. Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images: Part of Advances in Neural Information Processing Systems (NIPS 2012) // Edited by: F. Pereira and C.J. Burges, L. Bottou, K.Q. Weinberger. – 2012. [Electronic resource] <https://papers.nips.cc/paper/2012>
6. Biau G., Scornet E. A random forest guided tour // TEST. – Springer Link, 2016. – Vol. 25. – p. 197–227. <https://doi.org/10.1007/s11749-016-0481-7>

7. Yiu T. Understanding Random Forest. – 2019. – [Electronic resource] - <https://towardsdatascience.com/understanding-random-forest-58381e0602d2>

8. A Review of IoT Sensing Applications and Challenges Using RFID and Wireless Sensor Networks / H. Landaluce, L. Arjona, Perallos A., F. Falcone et al. // Sensor. – 2020. – No. 20, Vol. 2495. - 18 p. <https://doi.org/10.3390/s20092495>

9. Sensing, Controlling, and IoT Infrastructure in Smart Building: A Review / Verma A., Prakash S., Srivastava V., Kumar A., Muk S.C.// IEEE Sensors Journal. – Vol. 19 Issue: 20 – 2019. – p. 9036 – 9046. <https://doi.org/10.1109/JSEN.2019.2922409>