Ministry of Education and Science of Ukraine

Department of Science and Education of Kharkiv Regional State Administration

Municipal Institution “Kharkiv Regional Junior Academy of Sciences of

Kharkiv Regional Council”

Computer Science Department

Section: Internet-Technologies and WEB-Design

**indoor air quality monitoring system**

The paper was prepared by:

**Mitin Valentyn Vitaliyovych**, student of the 11th grade of Kharkiv Educational Complex No.45 “Academic Gymnasium” of Kharkiv City Council, Kharkiv Region

Research Advisor:

**Rukkas Kyrylo Markovych**, Professor of the Department of Theoretical and Applied Informatics, Faculty of Mechanics and Mathematics, V.N. Karazin Kharkiv National University, Associate Professor, Doctor of Engineering

Kharkiv - 2021

**abstract**

**indoor AIR QUALITY MONITORING SYSTEM**

**Mitin Valentyn Vitaliyovych**; Kharkiv Territorial Branch of the Academy of Sciences of Ukraine; Municipal Institution “Kharkiv Regional Junior Academy of Sciences of Kharkiv Regional Council”; Kharkiv Educational Complex No.45 “Academic Gymnasium” of Kharkiv City Council of Kharkiv Region; Grade 11; Kharkiv;

**Rukkas Kyrylo Markovych**, Professor of the Department of Theoretical and Applied Informatics, Faculty of Mechanics and Mathematics, V.N. Karazin Kharkiv National University, Associate Professor, Doctor of Engineering.

The state-of-the-art of atmospheric air is accompanied by a change of its natural composition, in particular, increased amount of carbon dioxide (CO2). The growing number of air pollutants poses a threat to the environment, as well as human health. Maximum permissible concentrations of harmful substances or approximate safe levels of exposure are the criterion for air quality. In this regard, more relevant is development of air quality measuring devices that allow monitoring data for transmission to the end user.

The purpose of the study is to ensure collection of data on indoor air quality. For this purpose, an air quality measuring device and program reading parameters from the sensor, and 2 programs for communication with the chatbot user in the Telegram and Web-page have been developed. All of these programs make up the air quality monitoring system described in this explanatory note.

A thorough analysis of the most well-known existing monitoring systems was performed to implement a fundamentally new system that has more advantages rather than disadvantages compared to older counterparts.

In practice, the system can be used to warn the user of the device in advance about polluted air in the room where he/she has installed the device, to prevent serious health problems.

**Keywords:** chat bot, monitoring system, volatile organic compounds, VOC, CO2, and air quality.

**CONTENT**

INTRODUCTION 4

SECTION 1. ANALYSIS OF EXISTING MONITORING SYSTEMS 7

SECTION 2. SELECTION OF HARDWARE FOR PROJECT IMPLEMENTATION AND SYSTEM OPERATIng procedure 11

2.1. Scheme of system operation 11

2.2. Rationale for the selection of the sensor 12

2.3. Justification for the selection of board 15

2.4. Raspberry pi settings for indoor air analysis system 18

2.5. Software on Raspberry pi 21 21

2.6. Software on the main server 23

2.7. Telegram-bot development of 23

2.8. Development of a WEB-page for interaction with the user 24

2.9. Strategy of frequency of data collection from the sensor and their further delivery to the user 27

SECTION 3. EXAMPLES OF THE SYSTEM use 29

3.1. Example of Telegram bot operation 29

3.2. Example of WEB-page operation 30

CONCLUSIONS 35

References 36

**INTRODUCTION**

The state-of-the-art of atmospheric air is accompanied by a change of its natural composition, in particular, increased amount of carbon dioxide (CO2). According to the Mauna Loa Observatory on May 11, 2019, the level of CO2 in the atmosphere reached an unprecedented mark in the history of its measurements - 415.26 ppm [1]. Such an increase was caused by fossil burning and deforestation of large forests.

On October 21, 2019, Ukraine was in third place in terms of air pollution on the map of air pollution around the world, which collects data from 11 thousand sensors in 90 countries, outriding India and China (US Environmental Protection Agency Project) [2].

The concentration of air pollutants is above the maximum permissible level poses a threat to the environment, as well as human health. It should be understood that the concentration of harmful substances in the air and in the indoor air may be different.

In all the premises, where people stay temporarily or permanently, carbon dioxide (CO2) is the main air pollutant. Normally, this gas is contained in the outdoor air at concentration of 300-400 ppm (0.03-0.04%), however, with each exhalation a person fills the surrounding air with a new portion of CO2 (18-25 liters per hour). Given that the concentration of carbon dioxide in the exhaled air is 100 times higher than in clean air, a room quickly becomes potentially dangerous to health.

Increasing level of CO2 can cause symptoms of lack of oxygen, deterioration of cognitive abilities, directly affect one’s ability to work, lead to its complete loss, which ultimately affects the educational process in schools and the results of companies.

One should take into account many products that emit volatile organic compounds into the air. The concentration of these substances indoors can be even 100 times higher than outside.

Volatile organic compounds or VOCs are chemicals that are released in the form of gases from solids or liquids, easily evaporate into the air, even at room temperature. Of course, formaldehyde, benzene and phenol are among the most dangerous according to the classification of the US Environmental Protection Agency (EPA), US Green Building Council (USGBC) and European Union (EU).

Studies claim that values above 500 ng/l (nanograms per liter) of volatile organic compounds can pose a health hazard to homeowners. However, data from thousands of tested houses show that average value is 1200 ng/l - more than twice the allowable level [3]. Even moderately elevated levels of these chemicals in the air can cause health problems for people, especially those suffering from allergies and asthma.

Because of this monitoring of air composition to preserve health and life, which becomes possible and convenient thanks to the developed system of air quality monitoring in any place chosen by the user, with the ability to obtain data on the result directly by the user, becomes especially **relevant**.

**The purpose** of the study is to ensure collection of data on indoor air quality.

For this purpose, the author was solving the following **tasks:**

1. Selection of optimal sensor and board;

2. Development of the program of reading of indications from the sensor and their transfer to the main server;

3. Development of a program for data analysis on the server and transmission of the final result to the user.

As a result of this study, a software and hardware complex that allows obtaining and processing air quality data at any point of geolocation with subsequent transmission of the result to the end user using a bot in the messenger - Telegram or on the user’s web page, has been developed.

**The object of study -** processes of collecting and displaying indoor air quality data.

**The subject of study -** algorithms for collecting and displaying indoor air quality data.

**Practical importance**. The resulting device can be used to measure indoor air quality.

**The methods of study** - collection and display of indoor air quality data using the developed programs.

To achieve **scientific novelty**, a thorough analysis of the most well-known existing monitoring systems described in the paper was performed to implement a fundamentally new system that has more advantages rather than disadvantages compared to old counterparts.

**Personal contribution:** this paper describes my development of a system for monitoring condition of the indoor air. For this, I have developed 2 programs using the python programming language, namely a Telegram bot and program for a measuring device. Also, for easy interaction, I have developed a web page, using MERN technology, to send settings to the user.

**SECTION 1**

**ANALYSIS OF EXISTING MONITORING SYSTEMS**

There are many types of measuring devices designed to monitor air quality, but normally for ordinary users they can be divided into 2 main types: for indoor use and portable ones that users can take with them.

According to one article in The New York Times on November 30, 2018, “a small gadget that measures ambient air pollution is a new version of preparation for the apocalypse as of 2019”. The article states that when confidence in air quality control at the government level fades, thousands of people across the country can take air measurements into their own hands.

As a result of a detailed review of the existing individual air quality monitoring systems, it was found that Atmotube devices are the most widespread in world practice.

Atmotube PLUS, that detects concentration of a wide range of Volatile Organic Substances (VOCs) in the atmosphere and indoor air and reports them in real time using the Atmotube program is one of the most common devices. The Atmotube PLUS also measures atmospheric pressure, temperature and humidity [10].

An example of the Atmotube PLUS device in Fig.1.1.



Fig. 1.1. Atmotube Atmospheric Tube [19]

Having studied its characteristics, it can be noted that its advantages are as follows:

- portability;

- ability to view air quality on the map;

- convenient application with graphics;

- ability to work for enough time without charging.

It also has some disadvantages:

- use of bluetooth transfer technology (4.0 LE), which when constantly connected to Atmotube affects the discharge rate of the user’s battery [11];

- use of this device is designed only to work on Android and IOS operating systems [10];

- relatively high price of the device.

In the comparative analysis of the device described above and system covered by this paper, it can be noted that the advantages of the developed system are as follows:

- no requirements to work with any particular operating system;

- ability to work with the device, if you have access only to the search engine, use Telegram Web, or if you have a Telegram messenger;

- more or less affordable.

Distributed in Ukraine, Xiaomi has developed a device called “Xiaomi PM 2.5 Air Detector”, as shown in Fig.1.2 [18]. The abbreviation PM 2.5 in the name of the gadget is a designation of a class of small harmful solid particles that are dangerous for human lungs. Thanks to PM 2.5 Air Detector laser sensor, it is easy to identify these harmful elements, which can be as small as 0.3 μm.

The OLED display of the air analyzer is provided with a special LED, which changes its color depending on the degree of purity of the air. Its parameters are as follows:

1) at concentration of harmful solid particles in the range of 0-75 μg/m3 - the color of the indicator is green, which means that air purity is normal;

2) at concentration in the range of 75-150 μg/m3 - it is already orange, which means that air condition is not satisfactory. At concentrations above 150 μg/m3, a red color is activated, which warns the owner that the ambient air is seriously polluted;

3) the device is also provided with a battery with capacity of 750 mAh, which provides up to 3 hours of continuous battery life of the gadget. PM 2.5 Air Detector is provided with a Wi-Fi module, thanks to which it can be controlled from a smartphone.

Fig.1.2. Measuring device from Xiaomi [20]

Having studied its characteristics, it can be noted that its advantages are as follows:

- portability;

- convenience of the application with graphics for the phone;

- ability to work for enough time without charging.

Whereas disadvantages refer to:

- measurement of only one parameter that affects the indoor air quality;

- high price of the device;

- inability to use the website or bot in the messenger if the user cannot or does not want to install an official application on the phone.

In the comparative analysis of the device described above and the system covered by this paper, it can be noted that the advantages of the developed system are:

- no requirements to work with a certain operating system;

- ability to work with the device, if you have access only to the search engine, use Telegram Web, or if you have a Telegram messenger;

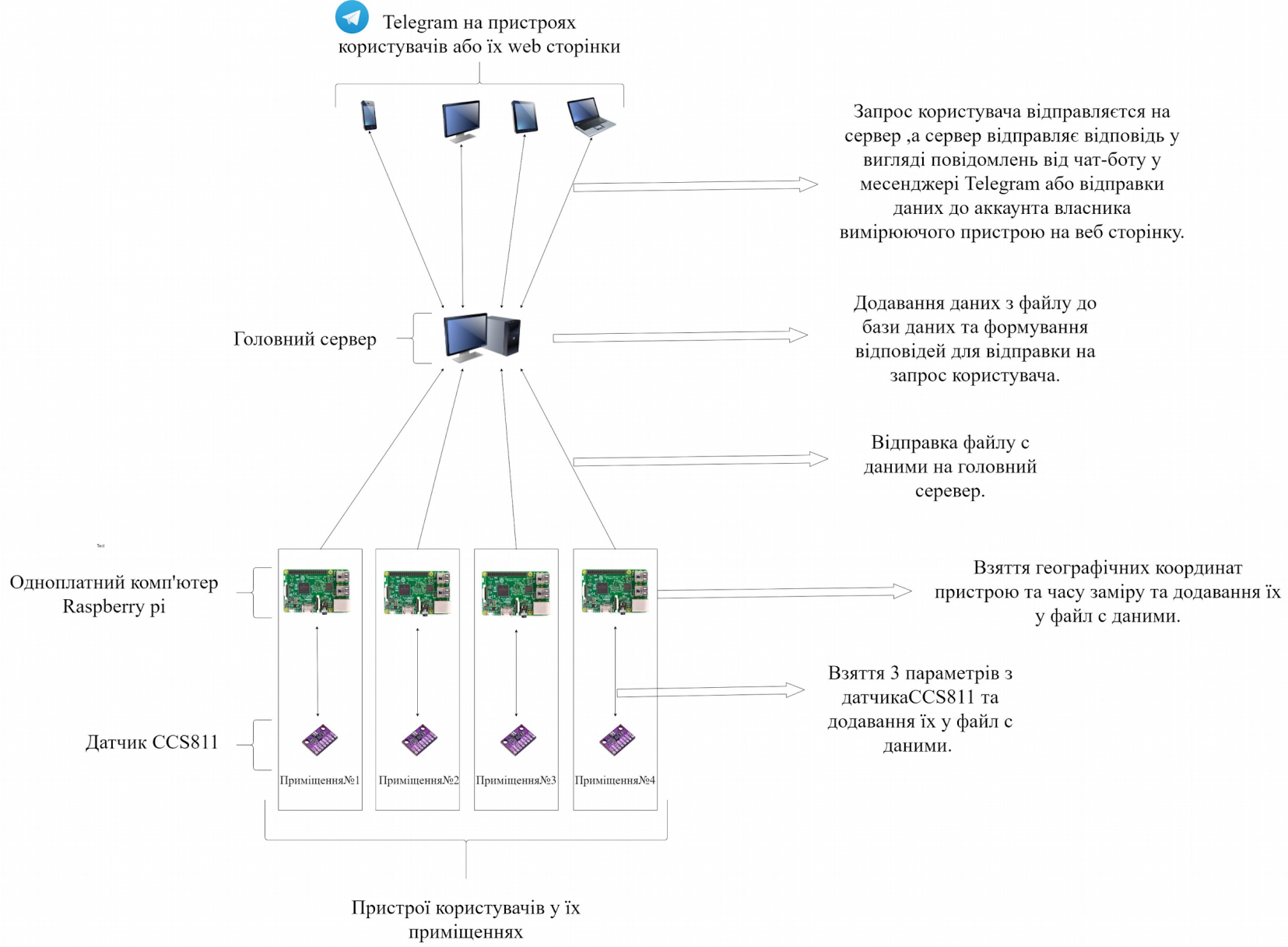
- more or less affordable.

**SECTION 2**

**SELECTION OF HARDWARE FOR PROJECT IMPLEMENTATION AND SYSTEM OPERATIng procedure**

**2.1. Scheme of system operation**

The system of analysis of the state of indoor air is comprised of many components that continuously perform their functions according to the algorithms developed by me:

Fig.2.1. Monitoring system (picture made by the author)

|  |  |
| --- | --- |
| *Запрос користувача відправляється на сервер, а сервер відправляє відповідь у вигляді повідомлень від чат-боту у месенджері Telegram або відправки даних до аккаунта власника вимірюючого пристрою на веб сторінку* | *The user’s request is sent to the server, and the server sends a response in the form of messages from the chatbot in the Telegram messenger or sending data to the account of the owner of the measuring device on the web page* |
| *Додавання даних з файлу до бази даних та формування відповідей для відправки на запрос користувача* | *Adding data from a file to the database and generating responses to send to the user’s request* |
| *Відправка файлу з даними на головний сервер* | *Sending a data file to the main server* |
| *Взяття географічних координат пристрою та часу заміру та додавання їх у файл з даними* | *Taking geographical coordinates of the device and measurement time and adding them to the data file* |
| *Взяття 3 параметрів з датчика CCS811 та додавання їх у файл з даними* | *Taking 3 parameters from the CCS811 sensor and adding them to the data file* |
| *Telegram на пристроях користувачів або їх web сторінки* | *Telegram on users’ devices or their web pages* |
| *Головний сервер* | *Main server* |
| *Одноплатний комп’ютер Raspberry pi* | *Raspberry pi single board computer* |
| *Датчик CCS811* | *CCS811 sensor* |
| *Пристрої користувачів у їх приміщеннях* | *Devices of users in their premises* |

**2.2. Justification for the selection of the sensor**

To develop the project, I had to select a sensor that could monitor air quality with some parameters. It turned out that in most cases people use a series of MQ sensors, which is very well distributed due to its affordable price. These sensors are made by the same manufacturer - the Chinese company HANWEI. Having studied HANWEI’s specifications, I have summarized all available MQ series sensors, substrate material and detection type in one table 2.1.

You can see an example of a sensor from MQ series in Figure 2.2.



Fig. 2.2. MQ series sensor [21]

The electrochemical sensor is built on the principle of changing resistance of one element when interacting with another element. In other words, there is a chemical reaction between these two elements, resulting in changes in the resistance of the substrate. But to ensure normal course of the reaction and not to make the sensor disposable, the sensitive part of the sensor shall be kept warm.

Advantages of MQ sensors:

* affordable price.

Disadvantages of MQ sensors:

* virtually identical sensors use the same sensing element and differ in the one used face value customizable resistors;
* no chance to select the measured gases, reacting to anything with carbon (and, quite possibly, to other elements that react with the substrate);
* lack of practical possibility to obtain meaningful values in the form of ppm or % [4].

*Table 2.1*

**Comparison of different models of MQ series sensors**

|  |  |  |
| --- | --- | --- |
| **Sensor** | **Gas causing reaction** | **Substrate material** |
| MQ-2 | Liquefied hydrocarbon gases | SnO2 |
| MQ-3 | Alcohol | SnO2 |
| MQ-4 | CH4 | SnO2 |
| MQ-5 | Liquefied hydrocarbon gases, natural gas | SnO2 |
| MQ-6 | Liquefied hydrocarbon gases, propane | SnO2 |
| MQ-7 | CO | SnO2 |
| MQ-9 | CH4, liquefied hydrocarbon gases | SnO2 |
| MQ-131 | O3 | SnO2 |
| MQ-135 | Multipurpose | SnO2 |
| MQ-136 | Multipurpose | SnO2 |
| MQ-137 | Multipurpose | SnO2 |
| MQ-138 | Multipurpose | SnO2 |
| MQ-303A | Alcohol | - |
| MQ-306 | Liquefied natural gas | - |

Because of these disadvantages had to look for other options for sensors.

CCS811 is a low-power digital gas sensor that integrates a metal oxide gas (MOX) sensor to detect a wide range of volatile organic compounds for indoor use. It monitors air quality using a microcontroller (MCU), which includes an analog-to-digital converter (ADC) and I²C interface. CCS811 has been designed for military cartographic service.

The integrated microcontroller controls the sensor driver modes and measurements.

The I²C digital interface simplifies hardware and software design greatly.

CCS811 supports intelligent algorithms for processing CO2 and VOC measurements.

CCS811 supports several measurement modes that have been optimized for low power consumption during active sensor measurement, and idle mode, increasing battery service life in portable devices.

Output indicators:

CO2: range for CCS811 is from 400 ppm to 8192 ppm.

VOC: range from 0 ppb to 1187 ppb [5, p.1].

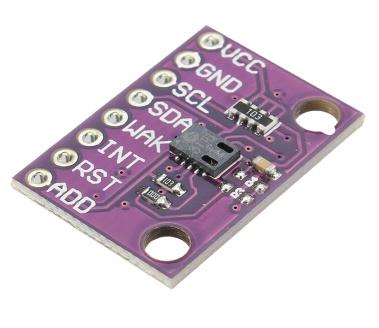
You can see an example of CCS811 sensor in Fig. 2.3.

Fig.2.3 CCS811 sensor [22]

**2.3. Justification of the selection of board**

Arduino is a company [with software](https://en.wikipedia.org/wiki/Open-source_hardware) and [software](https://en.wikipedia.org/wiki/Open-source_software) with [open source](https://en.wikipedia.org/wiki/Open-source_hardware), community of projects and users that develops and manufactures unidirectional microcontrollers and kits [microcontrollers](https://en.wikipedia.org/wiki/Microcontroller) to build digital devices. Its products are licensed for [GNU General Public License](https://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License) (LGPL) or GNU General Public License (GPL)[, what](https://en.wikipedia.org/wiki/Arduino#cite_note-1) allows producing Arduino boards and selling its software to anyone. Arduino boards are available for sale as pre-assembled items or “[do](https://en.wikipedia.org/wiki/Do-it-yourself) yourself” kits.

Different microprocessors and controllers are used in the design of Arduino boards. The boards are equipped with sets of digital and analog input pins/output, which can be combined with various extension boards or mock-ups (for prototyping) and other circuits. The boards have a serial communication interface, including USB on some models, which are also used to download programs from personal computers. Microcontrollers can be programmed using languages programming C and C++. You can see an example of Arduino microcontroller in Fig.2.4.

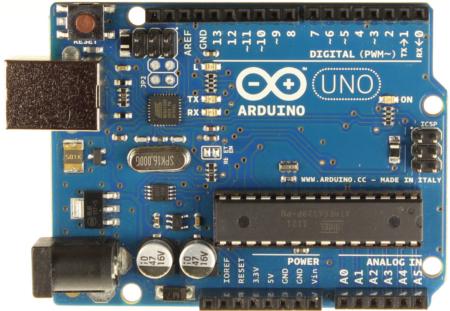


Fig.2.4. Arduino microcontroller [23]

Raspberry Pi is a series of small, single-board computers developed in the United Kingdom by Pi Foundation, to promote teaching of basic computer science in schools. The original model has become much more popular than expected, for sale outside its target market for uses such as robotics. The Raspberry Pi Foundation offers Raspbian, a Debian-based Linux distribution for download [6]. You can see an example of a Raspberry Pi single board computer in Figure 2.5.

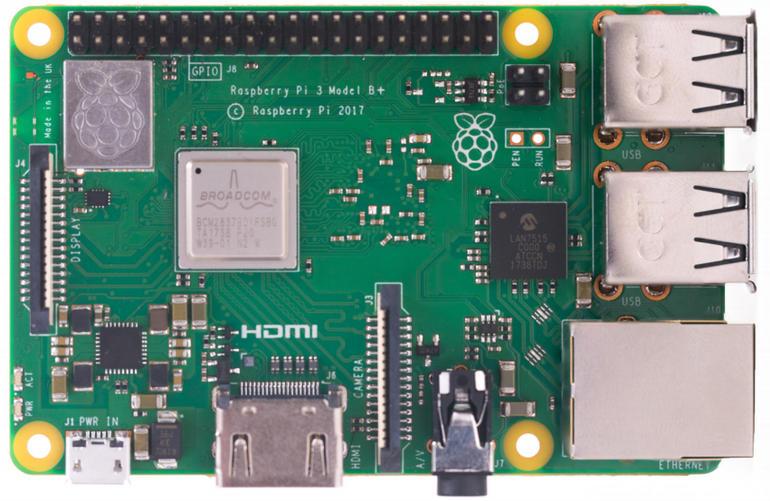


Fig.2.5. Raspberry Pi Single-Board Computer [24]

To select a board, I have selected the most important criteria and compared them in table 2.2.

*Table 2.2*

**Comparison of Arduino with Raspberry pi**

|  |  |  |
| --- | --- | --- |
|  | Microcontroller | Single-board computer |
| Performance | 1 core,  tens to hundreds of MHz,  dozens of KB of RAM,  tens to hundreds of KB of permanent memory. | 1 or more cores,  hundreds of thousands of MHz,  hundreds of MB of RAM,  gigabytes of non-volatile memory. |
| Multitasking | No. | Yes.  Managed by Raspbian OS. |
| Ease of working with the Internet | Additional modules and in-depth knowledge of protocols are usually required. | It is easily connected from the box, network module is usually already installed on the board. |
| Battery life | Consumes units - tens of mA. Possible weeks of battery life. | It consumes hundreds of thousands of mA. The charge of a large battery will last for ten hours. |
| Response rate in time-critical projects | 100% control over the time and duration of signals. | Because of multitasking, a critical process can fall asleep in time. |
| Choice of programming languages | Limited. Most often C/C++. | Python, JavaScript, Bash and dozens of others: any available in OS. |

For this project, I have selected Raspberry pi, because in the algorithmic part I use sending a file with data from the sensor to the main server using SCP. SCP refers to files transfer via SSH, which would be very difficult to do on the Arduino, buying additional modules. Yet, on Raspberry pi one can program on python. The library for programming CCS811 sensor from Adafruit Industries is written under this programming language.

Adafruit Industries is an open-source company what operates in New York, and was founded by Limor Frieda in 2005. The company develops, manufactures and sells a range of electronics, electronic components, tools and accessories. It also produces a number of training resources, including live and recorded videos related to electronics, technology and programming [7].

**2.4. Raspberry pi settings for indoor air analysis system**

To make a sensor work, one will need to properly connect it to the board and configure it to work with it. for this, follow these step-by-step instructions:

1) You need to turn on the I2C bus so that Raspberry pi could communicate with the sensor. For this, open the console and enter the command: sudo raspi-config.

2) After that, follow the step-by-step instructions in the order shown in Fig.2.6, 2.7, 2.8, and 2.9.

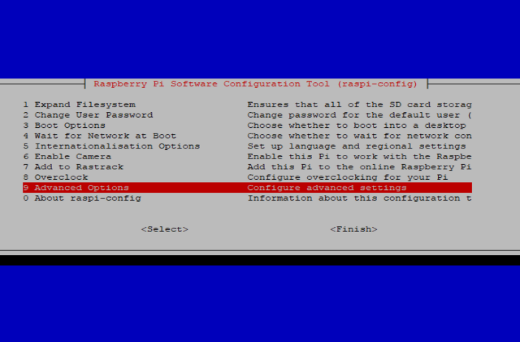


Fig.2.6. Step-by-step instructions for setting up Raspberry pi [25]

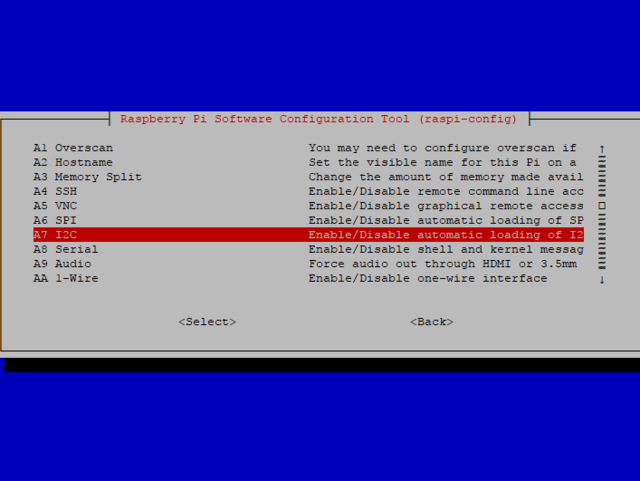


Fig.2.7. Step-by-step instructions for setting up Raspberry pi [26]



Fig.2.8. Step-by-step instructions for setting up Raspberry pi [27]

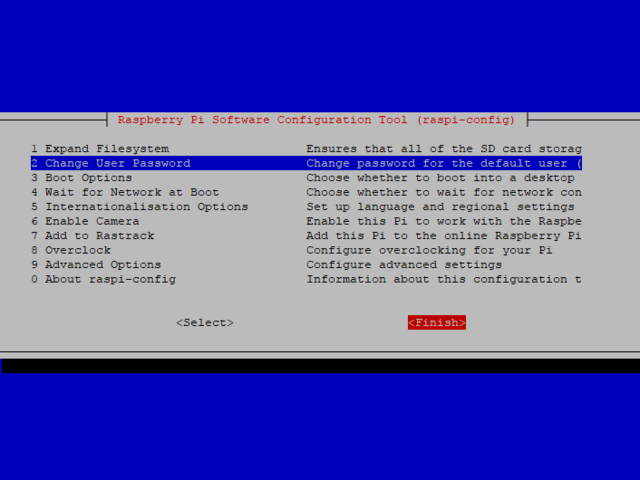


Fig.2.9. Step-by-step instructions for setting up Raspberry pi [28]

To make a sensor work properly, reduce the flow rate. For this, enter the following at the command prompt:

**sudo nano/boot/config.txt**

After that, a file will open in the command line, at the end thereof you need to add a line:

**dtparam = i2c\_baudrate = 10000**

Press Ctrl+X, then Y to save any changes to the file.

Now, we need to disconnect the Raspberry pi from the power supply to connect the sensor to it.

When Pi is disconnected, we can connect the sensor to the Raspberry Pi as follows:

* connect Vin to a 3V or 5V power supply on the board;
* connect GND to the ground pin on the board;
* connect SDA to the sensor to SDA pin on the board;
* connect SCL on the sensor to SCL pin on the board;
* connect Wake to the ground contact on the board [8].

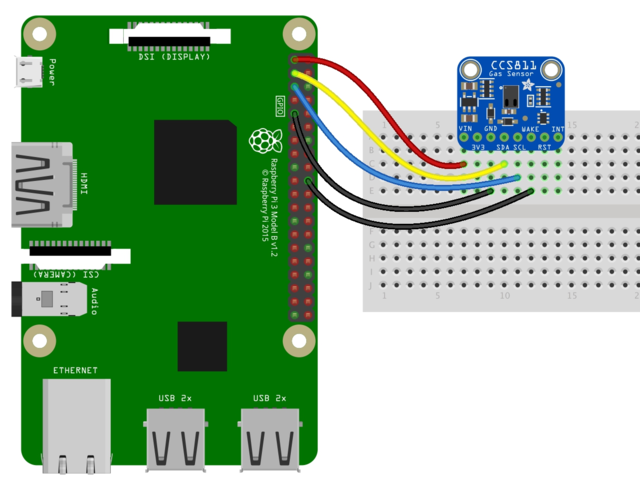
This connection is shown in detail in Fig.2.10.

Fig.2.10. Sensor connection to Raspberry Pi [29]

But to program the sensor in the python programming language, one needs to install a special library from Adafruit Industries. For this, write 2 commands on the command line:

**sudo apt-get update**

**sudo pip install Adafruit\_CCS811**

**2.5 Software for Raspberry pi**

The program on Raspberry pi can be divided into 4 conditional parts:

1) taking parameters from CCS811sensor using Adafruit\_CCS811 library;

2) taking geographical coordinates of the device using the library using the geocoder library for the python programming language;

3) encoding them in one line and write to a file on Raspberry pi;

4) transferring the data file to the server for further analysis.

Description of every part:

1) To write a program for interaction with this sensor, one needs to install the program library Adafruit\_CCS811. This is done using the command entered on the command line:

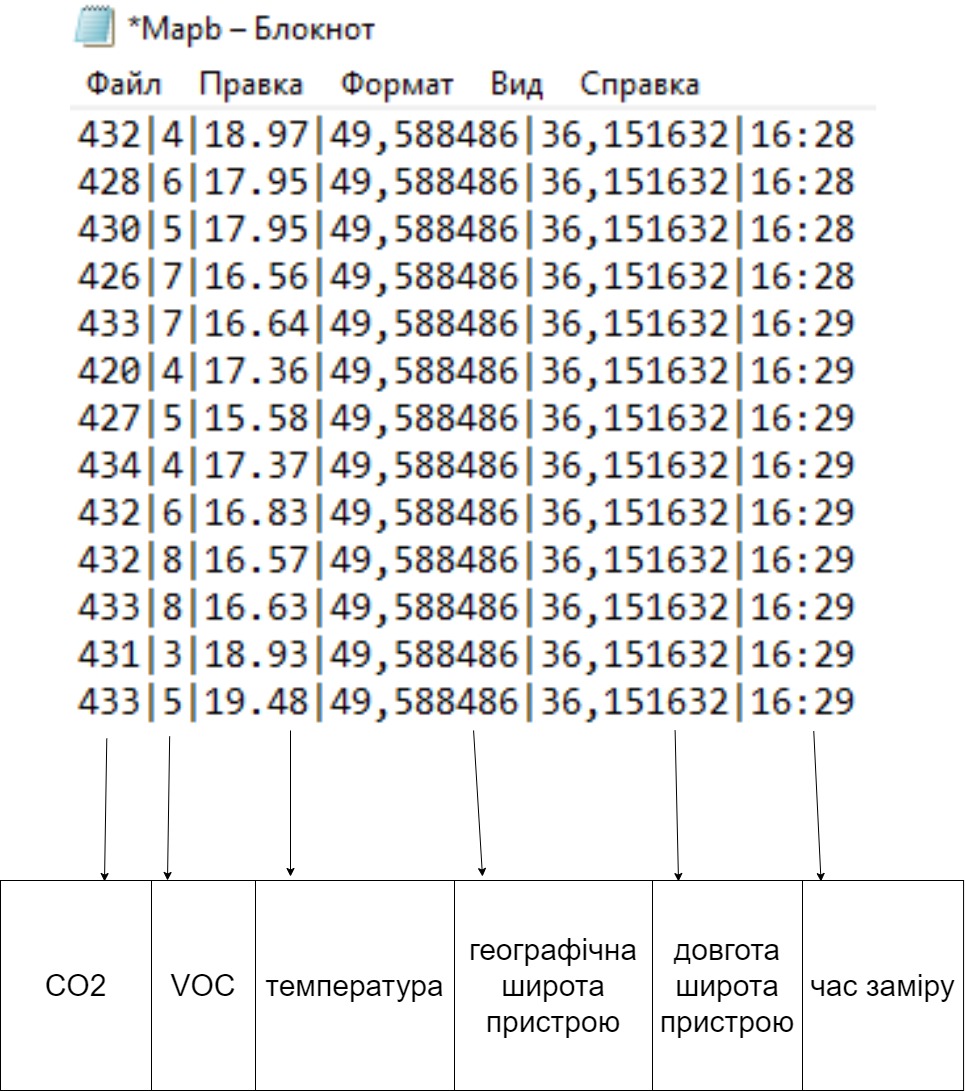
**pip install Adafruit\_CCS811**

After that, just connect it to the program.

2) Obtaining geographical coordinates for further establishment of distance between devices in the program on the server is ensured by means of the geocoder library which is established by the command entered in a command line:

**pip install geocoder**

3) After receiving 3 parameters from the sensor: VOC, CO2, temperature, geodata and time of measurement from the device for convenience of processing of these data on the server, they need to be written down in a file through the given symbol “|”. Together in the file there are lines in which the parameters of measurements are written in the following order: CO2 | VOC | temperature | latitude | longitude | time. You can see an example of such a file in Fig.2.11.

Fig.2.11. Example of a file with recorded parameters (picture made by the author)

|  |  |
| --- | --- |
| *Блокнот* | *Notepad* |
| *Температура* | *Temperature* |
| *Географічна широта пристрою* | *Latitude of the device* |
| *Довгота широта пристрою* | *Longitude, latitude of the device* |
| *Час заміру* | *Time of measurement* |
| *Файл* | *File* |
| *Правка* | *Edit* |
| *Формат* | *Format* |
| *Вид* | *View* |
| *Справка* | *Help* |

4) I have selected SCP to transfer the files.

SCP (from the English. Secure copy) - a utility and copy protocol, in contrast to the utility RCP, as a transport is not RSH, and encrypted SSH. A functionally similar utility is sftp.

On UNIX-like operating systems, the eponymous (scp) remote file copying utility is often included in the openssh package [9].

Raspberry pi sends the file as in Fig.2.12, to the server every minute.

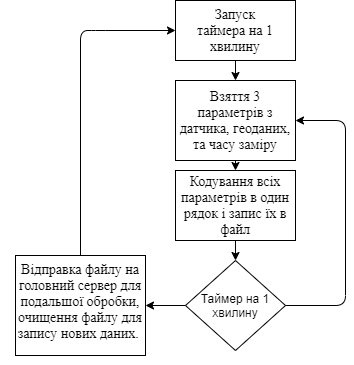


Fig.2.12. Flowchart of the Raspberry Pi (picture made by the author)

|  |  |
| --- | --- |
| *Запуск таймера на 1 хвилину* | *Start the timer for 1 minute* |
| *Взяття 3 параметрів з датчика, геоданих та часу виміру* | *Taking 3 parameters from the sensor, geodata and measurement time* |
| *Кодування всіх параметрів в один рядок і запис їх в файл* | *Encoding all parameters into one line and saving them to a file* |
| *Таймер на 1 хвилину* | *Timer for 1 minute* |
| *Відправка файлу на головний сервер для подальшої обробки, очищення файлу для запису нових даних* | *Sending the file to the main server for further processing, cleaning the file to record new data* |

**2.6. Software on the main server**

The program on the main server is much more comprehensive than the one on the Raspberry pi and is comprised of 3 large blocks:

1) Algorithm for decoding a file to extract parameters from it, as well as their entry in a common database. The device id, 3 measurement parameters and geographical coordinates of the device are added to the database.

As the sensor is too sensitive, I have decided to average all the values obtained from the file. After that, these values are analyzed to compile a report and then send to the user, as in Fig.3.1.

2) A bot in the Telegram messenger called “Air Pollution Monitoring Bot” (APMB) with which the user communicates.

3) A website where a user can register to communicate with the device, if he/she is not able to use the bot in Telegram.

**2.7. Telegram bot development**

To start working with the bot you need to send a command to it:

**/start**

After that, 6 buttons appear: “information”, “monitoring”, “latest data”, “complete testing”, “statistics”, “nearest devices”. An example of a panel with buttons is shown in Fig.3.3.

Functions of every button are as follows:

1) when pressing the “information” button, the bot sends a response about the impact on the human body of CO2 and VOC at their concentration above the permissible level;

2) “monitoring” button activates the process of notification from the 1st block of the user about the state of the air with the set frequency every 10 minutes;

3) “latest data” button sends from the 1st block to the user the latest report on a condition of air. An example of such a message is shown in Figure 3.1;

4) the "stop monitoring" button interrupts the process of sending notifications to the user;

5) “statistics” button sends the data issued by the algorithm from the point for the period of time specified by the user after pressing the button, as in Fig.3.2;

The flowchart of the program on the main server is shown in Fig.2.13.

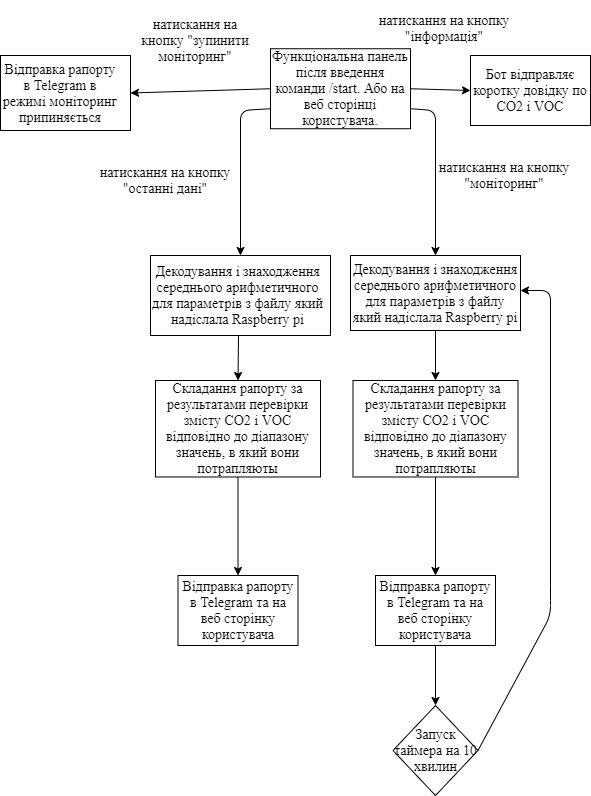
6) “nearest devices” button sends to the user the last measurements from devices of other persons which are in the range specified by the user after pressing this button.

Fig. 2.13. Flowchart of the Telegram bot program (picture made by the author)

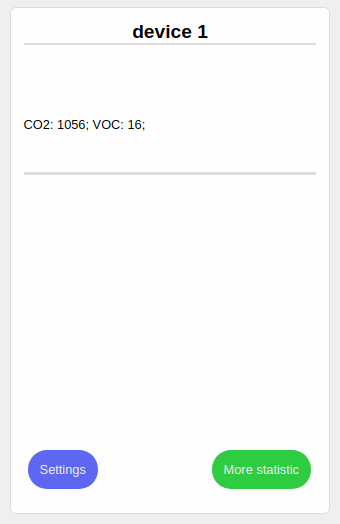
|  |  |
| --- | --- |
| *Відправка рапорту в Telegram в режимі моніторинг припиняється* | *Sending a report to the Telegram in monitoring mode is stopped* |
| *Натискання на кнопку «останні дані»* | *Clicking on “latest data” button* |
| *Натискання на кнопку «запинити моніторинг»* | *Clicking on “stop monitoring” button* |
| *Натискання на кнопку «інформація»* | *Clicking on “information” button* |
| *Бот відправляє коротку довідку по СО2 і VOC* | *The bot sends short information on CO2 and VOC* |
| *Натискання на кнопку «моніторинг»* | *Clicking on “monitoring” button* |
| *Функціональна панель після введення команди /start або на веб сторінці користувача* | *Function panel after entering the command /start or on the user’s web page* |
| *Декодування та знаходження середнього арифметичного для параметрів з файлу, який надіслала Raspberry pi* | *Decoding and finding the arithmetic mean for parameters from a file sent by Raspberry pi* |
| *Складання рапорту за результатами перевірки змісту СО2 і VOC відповідно до діапазону значень, в який вони потрапляють* | *Compilation of a report on the results of checking the content of CO2 and VOC in accordance with the range of values they refer to* |
| *Відправка рапорту в Telegram або на веб сторінку користувача* | *Sending a report to the Telegram or user’s web page* |
| *Запуск таймера на 10 хвилин* | *Start the timer for 10 minutes* |

**2.8. Development of a WEB-page for interaction with the user**

The site is comprised of 4 pages: “Devices”, “Info”, “Login”, “Register”.

Functions of every page are as follows:

1) “Devices” page is the main page of the site, where convenient cards are located, an example of one of them is shown in Fig.2.15 with the number of the measuring device and 2 main parameters VOC and CO2. In order to see other parameters and statistics, you need to click on “More statistic” button. You can also turn off the device or change the frequency of updates from standard to your room using “Settings” button;

Fig. 2.15. Example of a card of one of the devices (picture made by the author)

2) “Info” page is a page where the user can read about the project;

3) “Login” page allows the user logging in to see data from their devices;

4) “Register” page is a page for registering an account in the system. You can only use the web page when registering in the system.

To develop the site, I have used a technology called MERN [15].

It stands for MongoDB, Express.js, React.js, Node.js, where:

1) MongoDB: JSON documents created in your React.js interface can be sent to the Express.js server for processing and (provided they are valid) stored directly in MongoDB for further retrieval.

2) Express.js: works inside the Node.js server. Express.js calls itself “a fast, impenetrable, minimalist web environment for Node.js”, and that’s really what it is. Express.js has powerful models for routing URLs (matching the inbound URL with the server function) and processing HTTP requests and responses.

You can connect to the Express.js features that power your program by making HTTP (XHR) or GET or POST XML requests from your React.js interface. These features, in turn, use the Node.js, MongoDB drivers, or through callbacks to use Promises, to access and update data in your MongoDB database.

3) React.js: React.js is a stack component of MERN, a declarative JavaScript framework for creating dynamic client-side applications in HTML. React allows creating complex interfaces with simple components, connect them to data on your server and display them as HTML.

The strength of React refers to processing of data-driven interfaces with minimal code and minimal pain, and it has all the impressive features you would expect from a modern web structure: excellent form support, error handling, events, lists, and more.

4) Node.js: it an open-source platform for high-performance networking applications written in the language JavaScript. The founder of the platform is Ryan Dahl. If before JavaScript was used to process data in the user’s browser, then Node.js provided the ability to run JavaScript scripts on the server and send the user the result of their execution. The Node.js platform has turned JavaScript into a common language with a large community of developers.

The MERN architecture makes it easy to build a three-tier architecture (interface, server, database) using JavaScript and JSON. The interaction of the stack components is shown in Fig.2.14 [15].

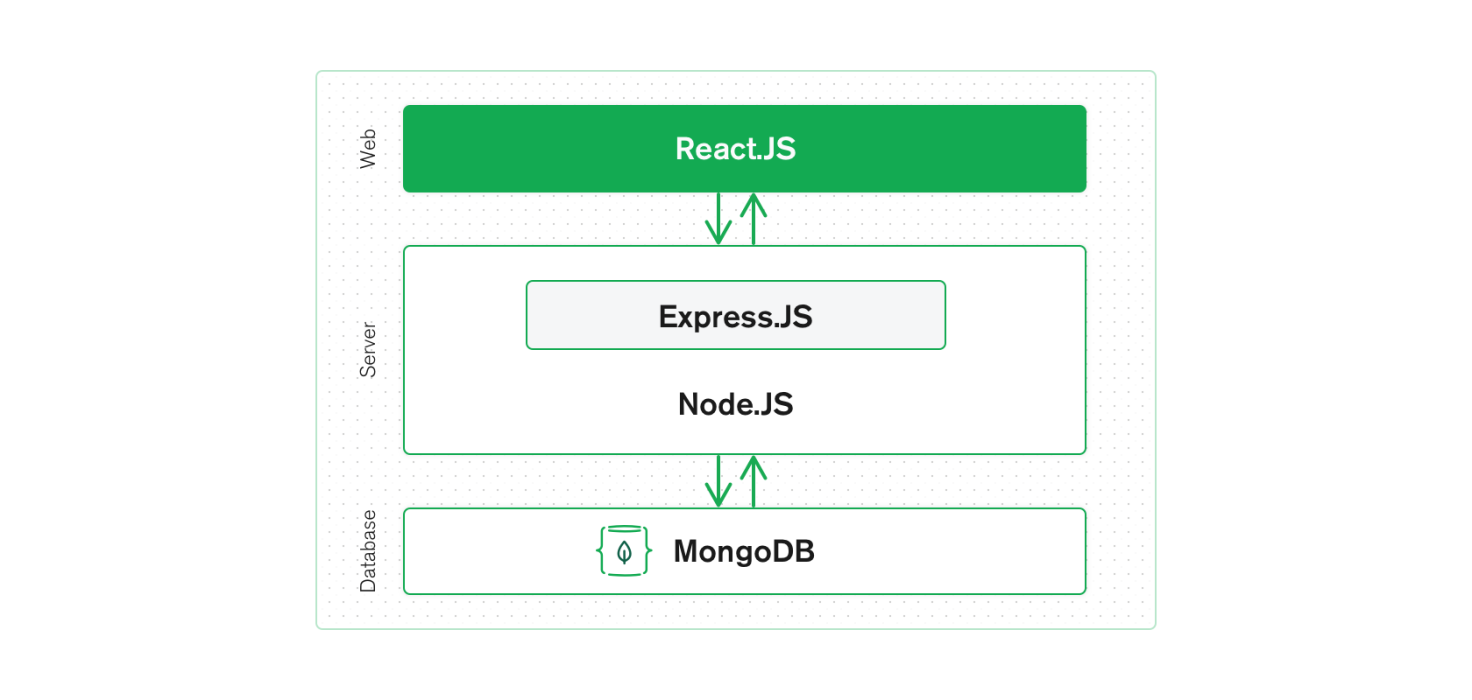


Fig.2.14. MERN Stack Interaction Scheme [30]

**2.9. Strategy of frequency of data collection from the sensor and their further delivery to the user**

To select frequency of data collection from the sensor and send them to the user, the following parameters should be considered:

1. the amount of CO2 produced by one person per unit time;
2. the number of persons, who can stay in the room.

This is necessary in order not to consume excess electricity, for example, if the device is connected to an uninterruptible power supply.

One person exhales from 18 to 25 liters of carbon dioxide within an hour, consuming 20-30 liters of oxygen.

If a person is in a room of 20 m2 with a ceiling height of 2.5 m and poor ventilation, the level of CO2 will be increased by 580 ppm every hour. Therefore, even a perfectly ventilated room will become a source of headache in an hour, and after 8 hours the concentration of CO2 in it will approach a critical level. If there is more than one person in the room, the speed of production СО2 will increase accordingly [16].

As an example, consider a closed classroom with 25 students and a teacher. In an hour and a half, 26 persons exhale 0.78 m3 of carbon dioxide into the auditorium (0.02 m3/h per 1 person). The volume of the classroom will be taken, for example, equal to 280 m3. As a percentage, calculate the volume of air replaced by CO2 gas- 0.2786%, which is the number of parts per million 2786 ppm. At this level there is already a decline in strength and poor performance [17].

On the basis of calculations, it is optimal to measure data with a sensor with an interval of one minute for a more accurate average value sent to the user. A more frequent number of measurements is not necessary and is excluded due to energy consumption. The frequency of data transmission is selected with an interval of 15 minutes, which allows not missing the moment of informing the user with a possible increase in CO2 and VOC above allowable values.

**SECTION 3**

**EXAMPLES OF THE SYSTEM USE**

**3.1. Example of Telegram-bot operation**

In this section, I show screenshots of the system and user interface:

1) Figure 3.1 shows menu of commands in Telegram, with which the user commands the bot.

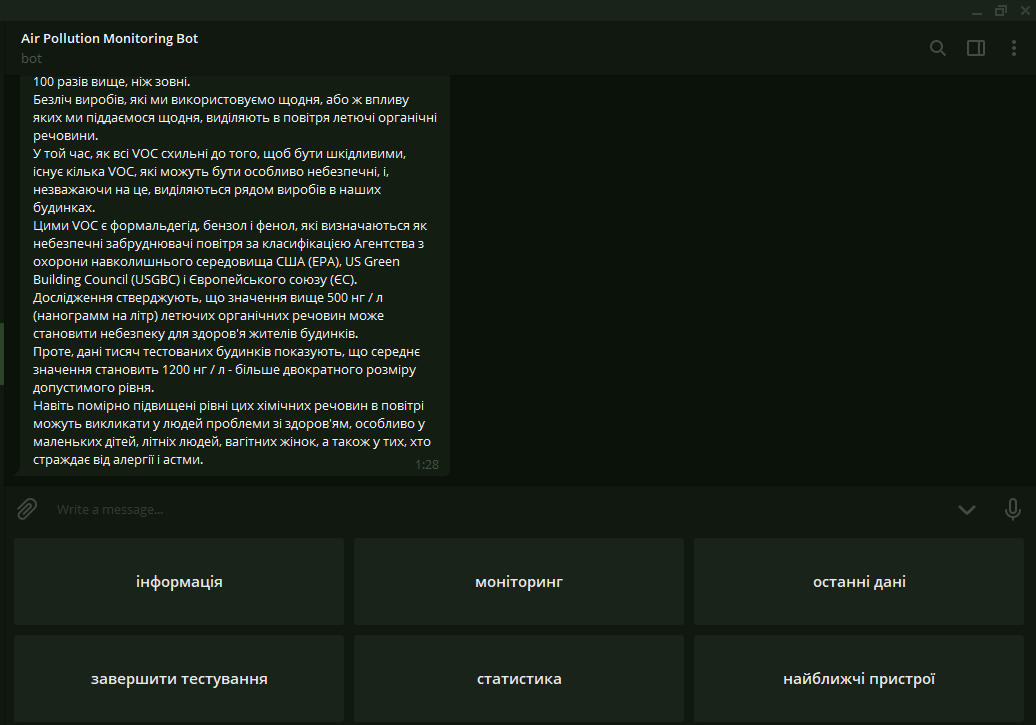


Fig.3.1. Example of the main control panel (picture made by the author)

2) Fig.3.2 shows response of the bot to the command “latest data” in the format. This message contains 3 measurements of VOC, CO2 and temperatures with textual explanations of the impact of such indicators.

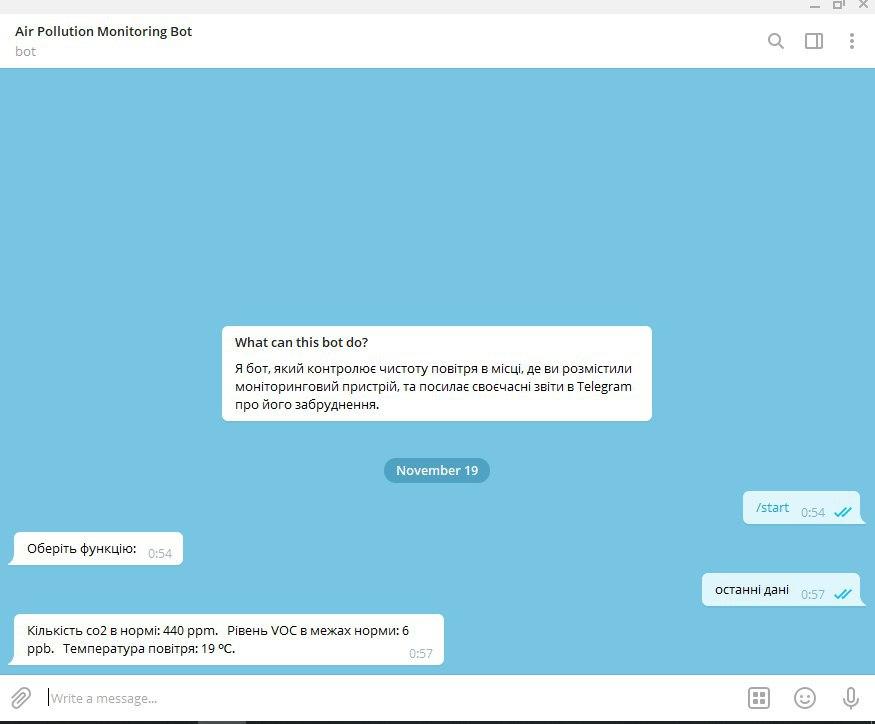


Fig.3.2. Example of an air condition report (picture made by the author)

3) Fig. 3.3 shows response of the bot to the command “statistics”, which indicates maximum and minimum values of all parameters: CO2, VOC and room temperature for a period of time specified by the user.



Fig. 3.3 Example of a statistical data collection algorithm report (picture made by the author)

**3.2. Example of WEB-page operation**

1) Fig.3.4 shows the main page of the website with device cards, which can be used to control measuring devices using “Settings” button, and view data using the “More statistic” button.

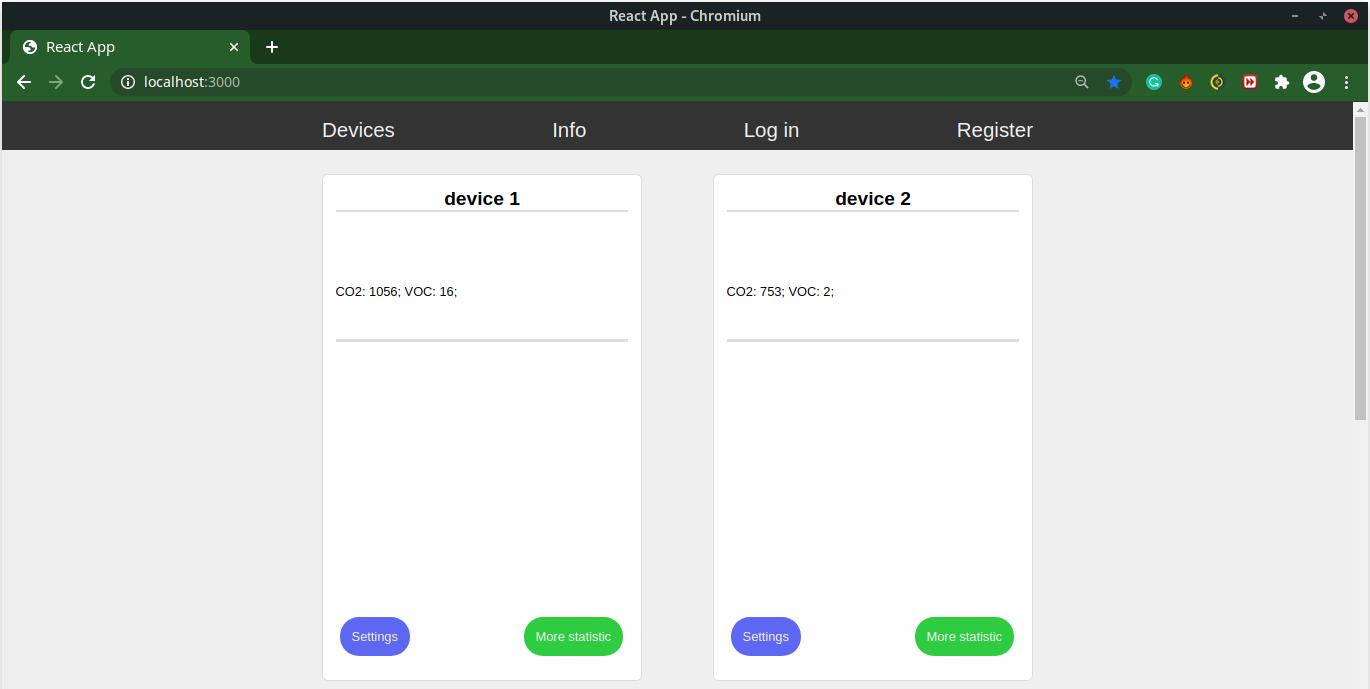


Fig.3.4. View of the main page of the WEB-site (picture made by the author)

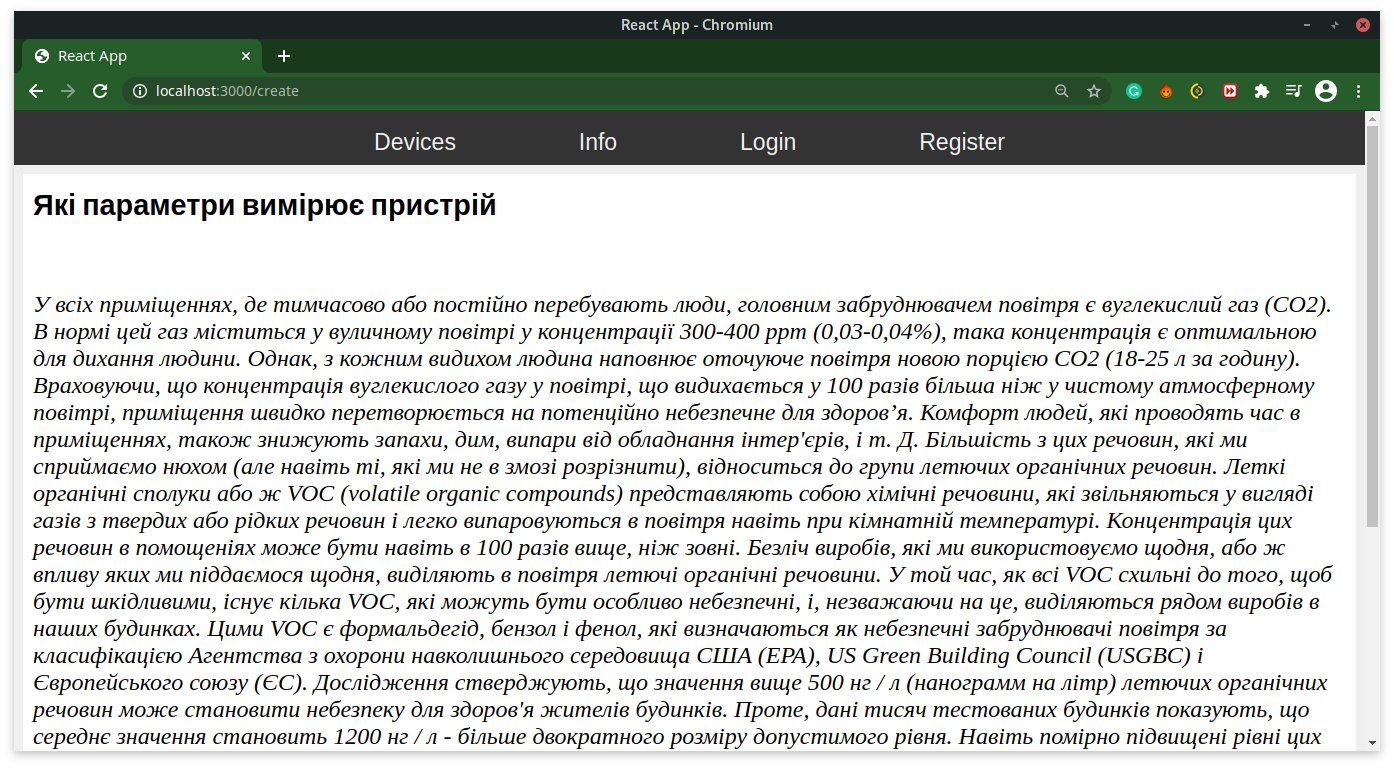
2) Fig.3.5 shows appearance of “Info” page, a link to which is on the top control panel. This page contains information about the usefulness of the developed system and the impact of the substances it measures on human health.

Fig. 3.5. Info page view (picture made by the author)

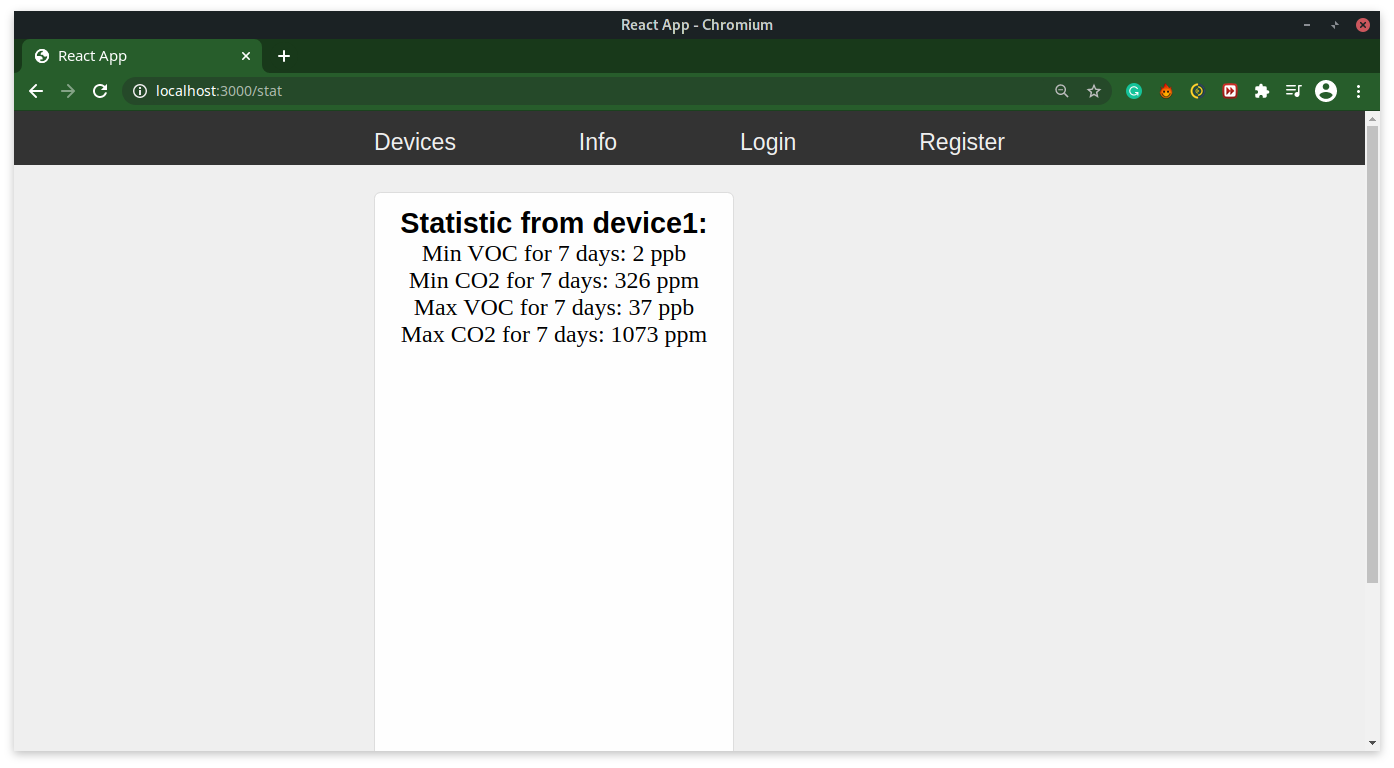
3) Fig.3.6 shows appearance of “More statistic” page, a link to which is located on the card of the device. It indicates the maximum and minimum CO2 and VOC values for the period of time specified by the user on the other “Settings” page (the default interval is 1 week).

Fig.3.6. Page “More statistic” (picture made by the author)

4) When you click on ‘Settings” button on the device card, the site will forward you to the page with the settings of this device (Fig.3.7). This page is comprised of the following components:

* texts box “Interval for collecting statistics” is provided for the user to enter a number in minutes, which will affect the interval of analysis of statistical data, ie if the user specified, for example, 2 weeks, “More Statistic” section of this device will indicate the maximum and minimum CO2 and VOC in exactly 2 weeks;
* text box “Data refresh rate from server to user” is used by the user to write the time interval in minutes, after which the air quality parameters on the site will be updated;
* text box “Data refresh rate from device to server” is designed to set the time in minutes, which affects the frequency of sending new data from the device to the server;
* “Reboot” button is designed to restart the measuring device, for example if it has stopped updating data;
* Shutdown button (device name) is designed to turn off the measuring device if the user wants to turn off the device;
* “Apply” button is designed to save new settings. Also, after clicking on it, the user gets to the main page.

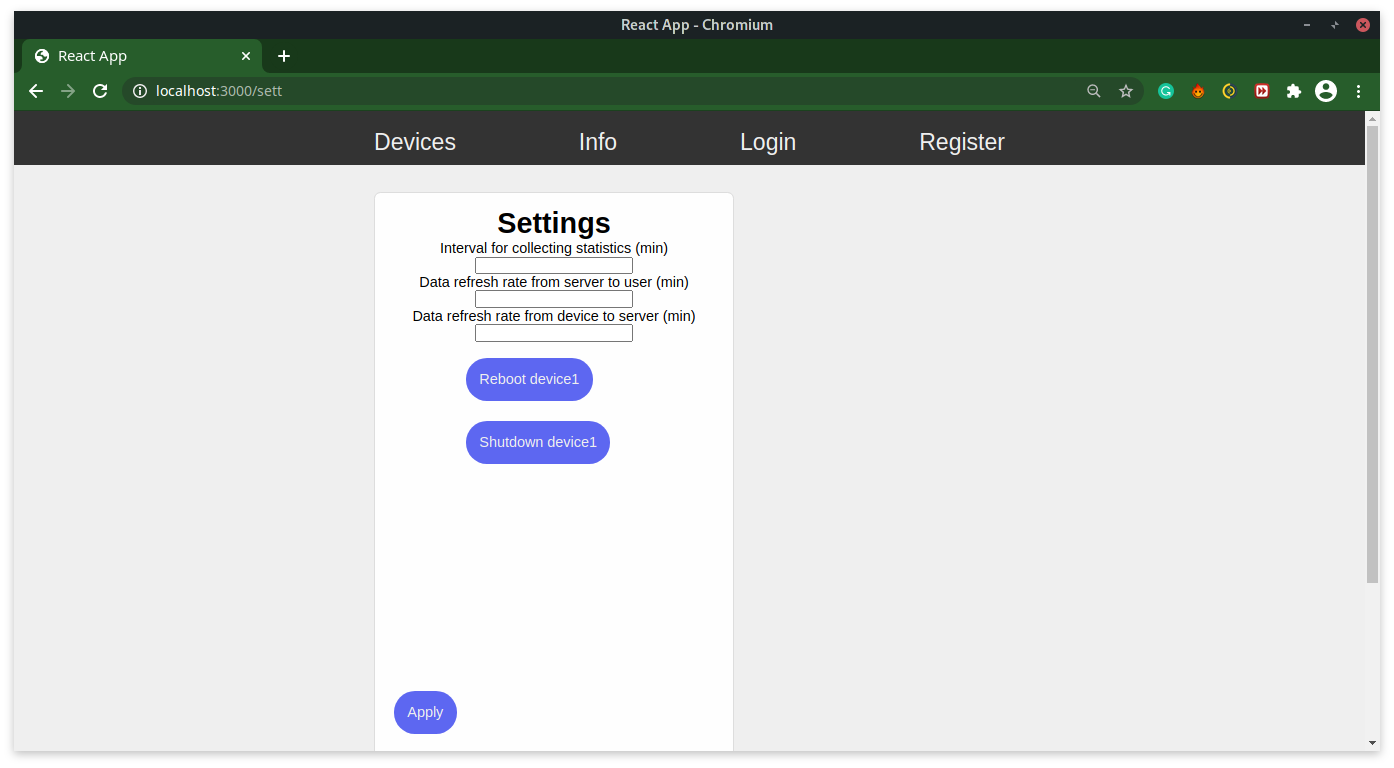


Fig.3.7. Example of Settings page (picture made by the author)

**CONCLUSIONS**

1. Currently, development of air quality measuring devices, which would allow users getting information at any time convenient for them, is becoming increasingly important.

2. A detailed review of existing projects and individual air quality monitoring systems revealed that they have not yet become widespread in Ukraine and that the need for information on air quality, especially indoors, is not satisfactory.

3. To develop an air quality monitoring system, the optimal CCS811 sensor, which supports intelligent algorithms for processing CO2 and VOC measurements and Raspberry pi board have been analyzed and selected.

4. The software was written in the Python programming language.

5. As a result of this paper, a software and hardware complex that allows getting and processing air quality data at any point of geolocation, followed by transmission of the result to the end user using a bot in the messenger - Telegram or user web pages, has been developed.

**REFERENCES**

**INTERNET RESOURCES**

1. TSN. The article “CO2 Levels in the Atmosphere Have Reached Unprecedented Levels. What Does This Mean for the Planet and Human Health”. URL:

https://tsn.ua/ru/svit/uroven-co2-v-atmosfere-dostig-nevidannoy-v-istorii-otmetki-chto-eto-oznachaet-dlya-planety-i-zdorovya-lyudey-1345605.html

2. UA League News. Article “Ukraine in a Cloud of Black Air”. How to Monitor CO2 and Follow up the Situation”. URL:

https://ua-news.liga.net/society/articles/ukraina-v-hmari-chornogo-povitrya-yak-monitoriti-pokazniki-so2-i-stejiti-za-situatsieyu

3. Official site of the Czech manufacturer of sensors Protronix. Article “Measuring CO2 is Often Insufficient”. URL:

https://www.careforair.eu/ru/merit-co2-casto-nestaci-ru/

4. Sparkfan. Documentation for CCS811 sensor. URL:

https://cdn.sparkfun.com/assets/learn\_tutorials/1/4/3/CCS811\_Datasheet-DS000459.pdf

5. [Mogobukfff](https://blog.kvv213.com/). Article “Dealing with CO and Methane Sensors MQ-4 and MQ-7”. URL:

https://blog.kvv213.com/2016/09/razbiraemsya-s-datchikami-co-i-metana-mq-4-i-mq-7/

6. Wikipedia. Article “Raspberry Pi”.URL:

https://en.wikipedia.org/wiki/Raspberry\_Pi

7. Wikipedia. Article “Adafruit Industries”. URL:

https://en.wikipedia.org/wiki/Adafruit\_Industries

8. Adafruit. Article “Adafruit CCS811 Air Quality Sensor”. URL:

https://learn.adafruit.com/adafruit-ccs811-air-quality-sensor/raspberry-pi-wiring-test

9. Wikipedia. Article “SCP”. URL:

https://ru.wikipedia.org/wiki/SCP

10. Official site of the manufacturer Atmotube. URL:<https://atmotube.com/?view=ru>

11. Habr. Article “Atmotube is a Portable Air Quality Sensor”. URL:

<https://habr.com/ru/company/dadget/blog/404389/>

12. New York Times. Article “Do You Know What You're Breathing?” URL: https://www.nytimes.com/2018/11/30/style/air-quality-pollution-monitors.html

13. UA League News. Article “Ukraine in a Cloud of Black air”. How to Monitor CO2 and Follow up the Situation”. URL:

https://ua-news.liga.net/society/articles/ukraina-v-hmari-chornogo-povitrya-yak-monitoriti-pokazniki-so2-i-stejiti-za-situatsieyu

14. Wikipedia. Article “Arduino”. URL:

https://en.wikipedia.org/wiki/Arduino

15. Mongo DB. Article “What is the MERN Stack?”. URL:

https://www.mongodb.com/mern-stack

16. Official site of the manufacturer of Vairgeo sensors. Article “Control of Ventilation by CO2 Level”. URL:

[http: //www.wiregeo.com/solutions/25/](http://www.wiregeo.com/solutions/25/)

17. Storeint. Article “The Content of Carbon Dioxide in a Room is Normal”. URL:

http://storeint.ru/soderzhanie-uglekislogo-gaza-v-pomeschenii-norma/

18. Myot. Article “Review: Xiaomi PM 2.5 Air Detector - a Sensor That Will not Leave Any Chance for the Dirty Air!”. URL:

https://miot.ua/news-and-actions/oglyad-xiaomi-pm-25-air-detector-sensor-yakiy-ne-zalishit-brudnomu-povitru-jodnogo-shansu-vtekti/

19. Atmospheric tube Atmotube. URL:

https://images.satu.kz/156219705\_w640\_h640\_portativnyj-monitor-kachestva.jpg

20. Measuring device from Xiaomi. URL:

https://wo.ua/upload/iblock/4a2/4a21133e11d77794159c76857374b3ed.jpg

21. MQ series sensor. URL:

https://buddyelectronics.com/wp-content/uploads/2020/11/cdds-300x269.png

22. CCS811 sensor. URL: https://opencircuit.shop/resources/content/ZJN5Y3K4M5DXRKHK6ASW/crop/900-600/Luchtkwaliteit.-temperatuur-en-luchtvochtigheids-sensor-CCS811-HDC1080.jpg

23. Arduino microcontroller. URL: https://media.springernature.com/original/springer-static/image/chp%3A10.1007%2F978-981-15-0199-9\_39/MediaObjects/481382\_1\_En\_39\_Fige\_HTML.png

24. Оraspberry pi computer. URL: https://pisces.bbystatic.com/image2/BestBuy\_US/images/products/6339/6339655\_sd.jpg

25. Step-by-step instructions for setting up Raspberry pi. URL:

https://cdn-learn.adafruit.com/assets/assets/000/047/803/thumb160/sensors\_Screenshot\_%2821%29.png?1509568345

26. Step-by-step instructions for setting up Raspberry pi. URL:

https://cdn-learn.adafruit.com/assets/assets/000/047/804/thumb160/sensors\_Screenshot\_%2822%29.png?1509568357

27. Step-by-step instructions for setting up Raspberry pi. URL:

https://cdn-learn.adafruit.com/assets/assets/000/047/806/thumb160/sensors\_Screenshot\_%2823%29.png?1509568411

28. Step-by-step instructions for setting up Raspberry pi. URL:

https://cdn-learn.adafruit.com/assets/assets/000/047/807/thumb160/sensors\_Screenshot\_%2824%29.png?1509568423

29. Connecting the sensor to the Raspberry Pi. URL:

https://cdn-learn.adafruit.com/assets/assets/000/098/293/medium640/adafruit\_products\_raspi\_ccs811\_i2c\_bb.jpg?1609367757

30. Scheme of interaction of MERN stack components. URL: https://webassets.mongodb.com/\_com\_assets/cms/mern-stack-b9q1kbudz0.png