

О. О. Іванюк, Г. І. Влах-Вигриновська, А. М. Близнюк, І. В. Сапіга
Національний університет “Львівська політехніка”,
кафедра комп’ютеризованих систем автоматики,

СИСТЕМА ВИЯВЛЕННЯ ТА ОЧИЩЕННЯ ПРИМІЩЕННЯ ВІД ШКІДЛИВИХ ГАЗІВ НА БАЗІ ТЕХНОЛОГІЇ ІНТЕРНЕТУ РЕЧЕЙ

<https://doi.org/10.23939/amm2020.01.040>

© Іванюк О. О., Влах-Вигриновська Г. І., Близнюк А. М., Сапіга І. В., 2020

Проаналізовано основні аспекти проектування системи виявлення та очищення приміщення від шкідливих газів. Запропоновано функціональну та принципову схеми системи, проаналізовано алгоритм функціонування пристрою. Розглянуто особливості платформи Інтернету речей ThingSpeak та взаємодію платформи з проєктованим пристроєм. Наведено опис мобільного iOS-додатка, який комунікує із проєктованим пристроєм через платформу ThingSpeak.

Ключові слова: система виявлення та очищення повітря від шкідливих газів, платформа Інтернету речей ThingSpeak, Arduino Uno, мобільний iOS-додаток.

The article is devoted to the analysis of the main aspects of designing a system for detecting and cleaning a room from harmful gases. The functional and schematic diagram of the system is proposed, the algorithm of the device functioning is analyzed. The features of the ThingSpeak Internet of Things platform and the interaction of the platform with the designed device are considered. The description of the mobile iOS application that communicates with the designed device through the ThingSpeak platform is given.

Key words: system of detection and cleaning of the premises from harmful gases, ThingSpeak Internet of Things platform, Arduino Uno, mobile iOS application.

Постановка проблеми

Сьогодні одним із головних видів забруднення, пов’язаних з діяльністю людини, є забруднення повітря. Внаслідок цього в атмосферу навколишнього середовища, зокрема у закриті приміщення, потрапляють викиди хімічних сполук, біологічних речовин, небезпечних як для людини, так і для інших живих організмів. Додаткова небезпека полягає в тому, що результат дії забруднення виявляється з часом, до прикладу, деякі речовини можуть завдавати шкоди озоновому шарові, діючи негативно на значну частину земних екосистем [1].

Враховуючи те, що в різноманітних приміщеннях закритого типу людина проводить велику частину свого життя, стан повітря в них мусить бути прийнятним відповідно до встановлених норм атмосферного повітря. Зокрема, згідно з державними стандартами, в закритих приміщеннях характеристики повітря визначають на основі норм атмосферного повітря.

Не викликає сумнівів, що питання керування вмістом шкідливих газів у повітрі, особливо враховуючи тенденції активного розвитку промисловості, залишатиметься і надалі актуальним. Необхідно регулювати концентрацію небезпечних газів в атмосферному повітрі, оскільки вона не може самостійно підтримуватися на прийнятному рівні.

До факторів, що впливають на рівень забрудненості повітря, можна зарахувати швидкість руху, температуру і відносну вологість повітря [2].

Щоб забезпечити високу продуктивність та хороше самопочуття працівників, необхідно ефективно керувати зазначеними вище фізичними характеристиками.

Аналіз останніх досліджень та публікацій

У роботі [3] наведено платформу моніторингу якості повітря в приміщенні на основі Інтернету речей, що складається з пристрою вимірювання якості повітря за назвою “Smart-Air” і веб-сервера. Платформа використовує Інтернет речей і технологію хмарних обчислень для моніторингу якості повітря в приміщенні в довільному місці без жодних часових обмежень. Smart-Air був розроблений на основі технології IoT для ефективного моніторингу якості повітря і передавання даних на веб-сервер через LTE в режимі реального часу. Пристрій складається з мікроконтролера, датчиків виявлення забруднюючих речовин і модему LTE. Розроблений пристрій забезпечує вимірювання концентрації аерозолі, летких органічних сполук, CO, CO₂ і температури-вологості для контролю якості повітря. Передбачено, що до платформи можна підключити систему вентиляції. Отже, система може працювати автоматично для поліпшення якості повітря щоразу, коли якість повітря невисока.

Проект [4] пропонує систему моніторингу забруднення повітря. Систему було реалізовано з використанням мікроконтролера Arduino. Запропоновану авторами систему розроблено для моніторингу та аналізу якості повітря в режимі реального часу і реєстрації даних на віддаленому сервері, підтримуючи оновлення даних через Інтернет. Вимірювання якості повітря проводили на основі показників “частин на мільйон” (PPM) і аналізували за допомогою Microsoft Excel. Результат відображається на інтерфейсі дисплея розробленого пристрою, і до нього можна отримати доступ через хмару на будь-якому розумному мобільному пристрої.

У роботі [5] розглядається розроблений прототип системи моніторингу забруднення повітря в навколишньому середовищі для визначення концентрацій основних газів, що забруднюють повітря. В системі використовуються недорогі вузли контролю якості повітря, що складаються з недорогого напівпровідникового датчика газу з модулями Wi-Fi. Система вимірює концентрацію газів, таких як CO, CO₂, SO₂ і NO₂, за допомогою напівпровідникових датчиків. Датчики збирають дані про різні параметри навколишнього середовища і передають їх на raspberry pi, який діє як базова станція. Реалізація даних, зібраних датчиками, відображається на веб-сервері на базі Raspberry pi 3. Відображення даних на веб-сайті реалізоване за допомогою програмного стеку MEAN. Основним аспектом пропонованої роботи є забезпечення недорогої інфраструктури для збирання і поширення даних серед усіх зацікавлених сторін.

Мета роботи

Мета роботи – розроблення системи виявлення та очищення приміщення від шкідливих газів із використанням платформи Інтернету речей ThingSpeak та мобільного iOS-додатка.

Виклад основного матеріалу

Розроблення функціональної схеми та алгоритму роботи системи виявлення та очищення приміщення від шкідливих газів

Враховуючи високу затребуваність та ефективність систем виявлення та очищення приміщення від шкідливих газів, нами запропоновано функціональну схему цієї системи з використанням мікроконтролера на базі Arduino, давачів сімейства MQ, платформи Інтернету речей ThingSpeak та мобільного iOS-додатка.

Функціональну схему проєктованої системи зображено на рис. 1. До складу системи входить:

- мікроконтролер Arduino Uno (збирає покази вмісту шкідливих газів в повітрі, опрацьовує їх, надсилає на Wi-Fi модуль ESP8266 та передає на LCD-дисплей);
- модуль давача вуглекислого газу MQ-135 (аналізує вміст CO₂ в повітрі);

- модуль давача метану MQ-9 (аналізує вміст CO в повітрі);
- модуль давача чадного газу MQ-7 (аналізує вміст CH₄ в повітрі);
- модуль давача температури та вологості DHT11;
- LCD-дисплей (відображає покази давачів);
- вентилятор (об'єкт керування);
- Wi-Fi модуль ESP8266 (здійснює комунікацію між мікроконтролером та ThingSpeak);
- платформа Інтернету речей ThingSpeak (працює з показниками давачів та командами включення/виключення вентилятора);
- мобільний iOS-додаток (відображає покази давачів, керує вентилятором).

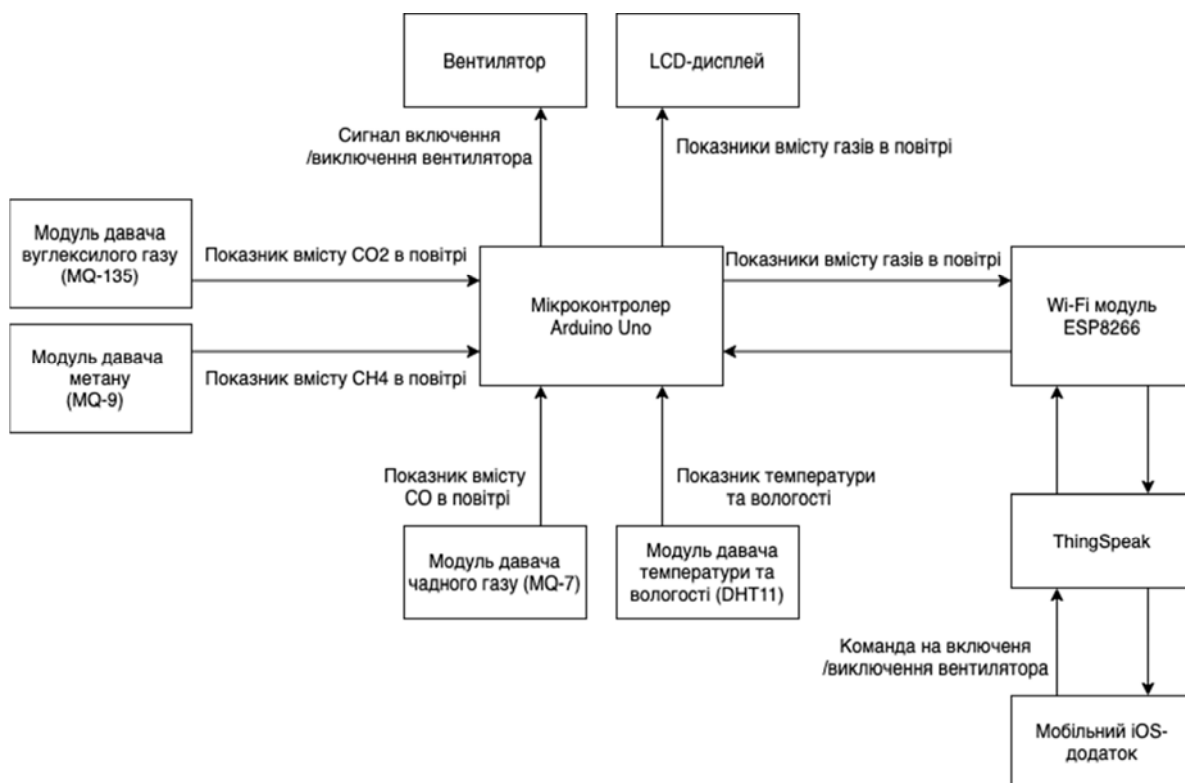


Рис. 1. Функціональна схема системи виявлення та очищення приміщення від шкідливих газів

Розглянемо детальніше призначення та особливості роботи блоків системи.

Програму керування пристроєм написано з використанням мови програмування C++.

В основу розробленої системи покладено мікроконтролер на базі платформи Arduino Uno. Він виконує такі завдання системи: збирає, аналізує та опрацьовує покази давачів вмісту метану, вуглекислого та чадного газів в повітрі. Крім того, контролер передає покази давачів на LCD-дисплей, використовуючи I2C-модуль. Мікроконтролер порівнює отримані покази давачів із нормованими. У разі знаходження високого показника вмісту будь-якого з газів у повітрі вмикається вентилятор. Також є можливість включення/виключення вентилятора з мобільного додатка.

LCD-дисплей працює на відображення показів давачів. Алгоритм відображення:

1. Температура та вологість;
2. Вміст чадного газу в повітрі та його норми;
3. Вміст метану в повітрі та його норми;
4. Вміст вуглекислого газу в повітрі та його норми.

Wi-Fi модуль ESP8266 комунікує з мікроконтролером, який передає йому покази давачів. Завданням ESP8266 є підключення до Wi-Fi, встановлення з'єднання з платформою Інтернету речей ThingSpeak для подальшого передавання отриманих показів давачів мікроконтролером у відповідний канал платформи [6]. Також через ESP8266 здійснюється отримання команди з внутрішньої аплікації платформи TalkBack на вмикання/вимкнення вентилятора.

Платформа Інтернету речей ThingSpeak слугує для зберігання та відображення отриманих показів давачів. Вона містить канал із п'ятьма графіками: графік температури, графік вологості, графік вмісту чадного газу в повітрі, графік вмісту метану в повітрі, графік вмісту вуглекислого газу в повітрі. Також велику роль відіграє її внутрішня аплікація TalkBack, яка являє собою чергу команд вмикання та вимкнення вентилятора, які створюються в мобільному додатку.

Мобільний додаток працює на операційній системі iOS. Написаний на мові програмування Swift [7]. Основною його задачею є відображення показів давачів на екрані мобільного телефону в реальному часі. Також в додатку присутня можливість перегляду відповідних графіків з платформи Інтернету речей ThingSpeak та, в разі потреби, ручного керування вентилятором. Ручне керування вентилятором здійснюється через маніпуляції з тумблером на головному екрані додатку і працює на створення команд TalkBack.

Розроблення принципової схеми пристрою

Для живлення мікроконтролера буде використаний мережевий AC/DC-адаптер. Усі компоненти системи живляться від 5В піна на мікроконтролерній платі. Винятком є Wi-Fi модуль ESP8266 – енергоживлення цього компонента здійснюється 3.3В піном.

Давачі чадного, вуглекислого газів та метану (MQ-7, MQ-135, MQ-9) передають дані через аналогові піни. MOSFET IRF520 та DHT11 підключаються до цифрових пінів. Wi-Fi модуль ESP8266 забезпечує комунікацію з Arduino Uno за протоколом UART, LCD-дисплей – за протоколом I2C.

Принципову схему наведено на рис. 2.

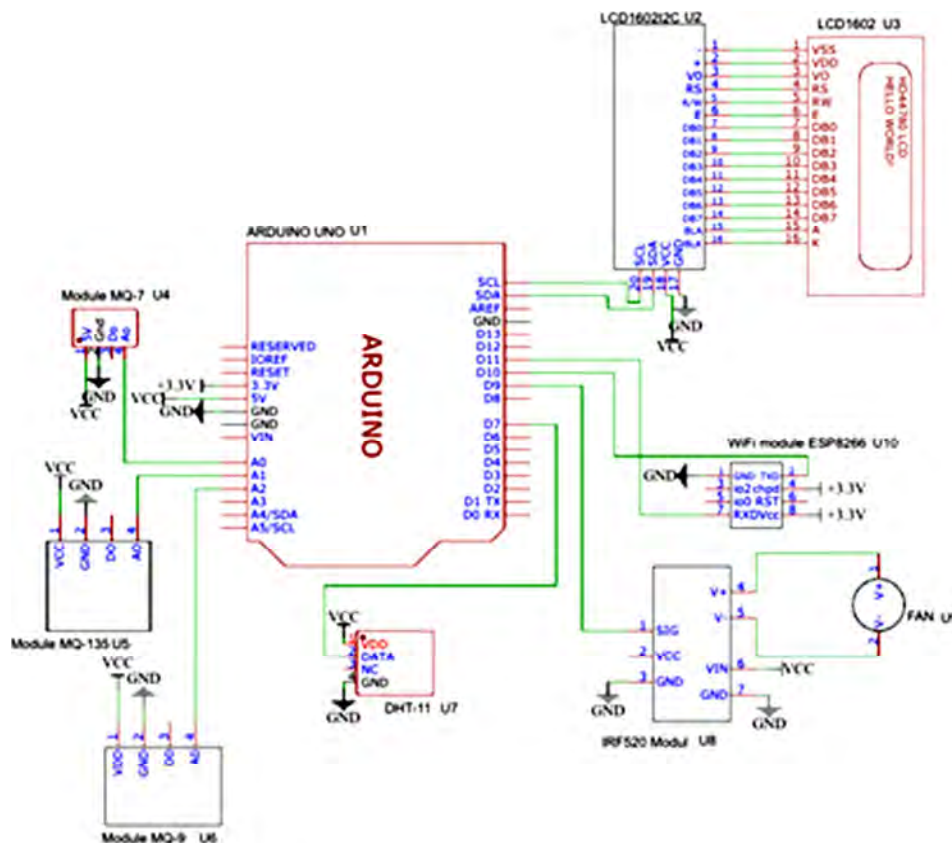


Рис. 2. Принципова схема пристрою

Компоненти для інтеграції, надані платформою, дозволяють передавати зібрані дані в деякі системи візуалізації / аналізу даних / зберігання даних, а також передавати дані на підключені пристрої або між ними на різні види призначених для користувача додатків.

Особливості платформи Інтернету речей ThingSpeak

Технологія Інтернет речей (IP) забезпечує підключення до мережі різних пристроїв та систем і розроблення програм для управління цими пристроями та системами. Завданням платформ IP є усунення всіх складностей з підключенням сервісів та хмари до таких систем.

У цій роботі використано платформу Інтернету речей ThingSpeak.

Канал є основною складовою моделі даних ThingSpeak, до якого входять поле ідентифікатора каналу, поле назви, поля розташувань, поля назв графіків, поля дати створення каналу та дати останньої зміни, поле кількості записів. Дані можливо записати в тому випадку, коли створено канал та відповідні графіки в каналі [8].

Для запису даних на канал ThingSpeak програмна реалізація алгоритму роботи пристрою з використанням Wi-Fi модуля надсилає запити на сервер ThingSpeak, ініціюючи виклик HTTP-запитів. Своєю чергою, для читання каналу мобільний додаток також використовує HTTP-запити для спілкування з сервером. До кожного каналу ThingSpeak може входити до восьми полів даних у різних форматах. До каналу також входить інформація про місцезнаходження. Кожному запису каналу відповідає унікальний ідентифікатор запису (entry_id), якому присвоюють дату із зазначенням часу. Вигляд каналу зображено на рис. 3.



Рис. 3. Вигляд каналу та його характеристик

Для організації процесу запису та читання даних з каналу ThingSpeak потрібно використовувати ThingSpeak API. Дозволом на читання та запис слугує 16-значний API ключ та 7-значний ідентифікатор каналу, які дають можливість клієнту читати та записувати дані на публічні та приватні канали [8]

Аплікація TalkBack слугує для створення команд, які повинен виконувати пристрій або система, яка під'єднана до платформи Інтернету речей ThingSpeak.

Основою аплікації є команда. До команди входять поле ідентифікатора доступу, поле назви команди, поле позиції, поле часу виконання команди, поле часу створення команди. Сукупність команд являє собою чергу команд, які повинен виконувати пристрій за послідовністю їх створення. Аплікація дозволяє створити команди в тому випадку, коли створено канал ThingSpeak.

Для створення і виконання команди мобільний додаток ініціює HTTP POST запити, які надсилаються на сервер ThingSpeak. Своєю чергою, програмна реалізація алгоритму роботи пристрою відслідковує останню виконану команду за допомогою HTTP GET запитів.

Кожній команді присвоюється унікальний ідентифікатор доступу із зазначенням часу її створення.

Для організації процесу створення та виконання команд використовують 5-значний ідентифікатор доступу TalkBack та 16-значний API ключ.

Виконання команди супроводжується її видаленням з черги. Список команд відображається на сторінці аплікації, зображеній на рис. 4.

The screenshot displays the 'Cooler TalkBack' configuration page in the TalkBack application. At the top, there is a breadcrumb trail: 'Apps / TalkBack / Cooler TalkBack'. Below this is a green 'Edit TalkBack' button. The main content area shows the following details:

- Name:** Cooler TalkBack
- TalkBack ID:** 38519
- API Key:** YCONCF8EYPCU5VMS (with a red 'Regenerate API Key' button next to it)
- Created:** 2020-05-04 4:28 pm
- Logged to Channel:** Air Pollution System

Below the configuration details is a section titled 'Commands' containing a table with the following data:

Position	Command ID	Command string
1	18869507	cooler_off
1	18869508	cooler_on

At the bottom of the 'Commands' section is a green 'Add a new command' button.

Рис. 4. Візуалізація команд в аплікації TalkBack

Опис мобільного iOS-додатка

Мобільний iOS-додаток містить 7 сцен:

- сцена підключення додатка до засобів платформи ThingSpeak;
- сцена відображення показів давачів температури та вологості, чадного газу, метану та вуглекислого газу. Також ця сцена містить тумблер, який працює на ввімкнення/вимкнення вентилятора та відповідні кнопки переходу на сцени графіків;
- 5 сцен побудови та відображення графіків температури, вологості повітря, вмісту чадного газу в повітрі, вмісту метану в повітрі, вмісту вуглекислого газу в повітрі відповідно.

Під час запуску додатка відкривається сцена підключення до засобів платформи Інтернету речей (рис. 5, а). Вона містить поля API-ключа каналу та ідентифікатора доступу до каналу ThingSpeak, поля API-ключа та ідентифікатора доступу до черги команд TalkBack. При коректному введенні даних та натиску на кнопку переходу відкриється наступна сцена, в іншому випадку – на екрані з'явиться повідомлення про помилку підключення (рис. 5, б).

Наступна сцена відповідає за відображення показів давачів та ввімкнення/вимкнення вентилятора (рис. 5, в). Механізм відображення показів реалізовано за допомогою формування запиту на отримання показів давачів з каналу ThingSpeak, із застосуванням HTTP GET-методів [8]. Своєю чергою, механізм керування вентилятором також використовує HTTP POST-методи для формування запиту на створення та виконання команд в черзі команд TalkBack.

При натиску на будь-яку кнопку-зображення графіка, відкриється відповідна сцена графіка (рис. 5, г).

Сцени графіків використовують посилання на канал платформи ThingSpeak для побудови перехідних процесів показів вимірюваних величин. Перехідні процеси змінюються динамічно, коли пристрій ввімкнено.

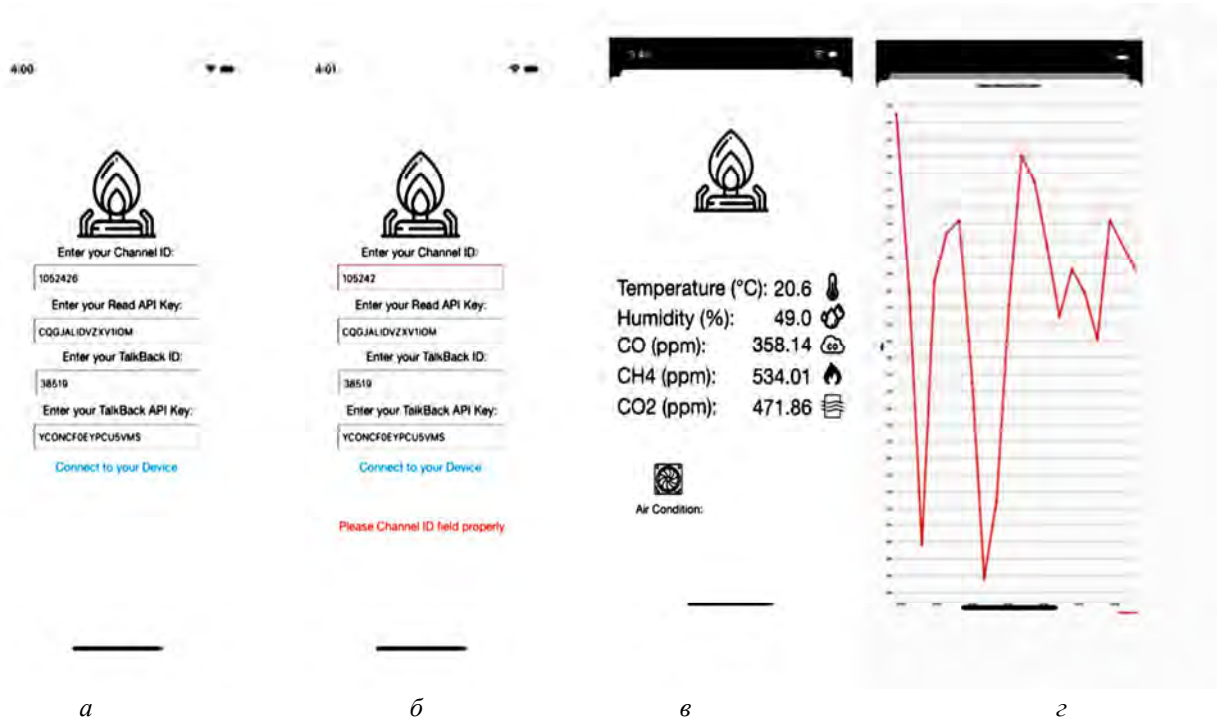


Рис. 5. Сцени додатка

Висновки

Розглянуто особливості розроблення системи виявлення та очищення приміщення від шкідливих газів.

Запропоновано функціональну схему цієї системи на основі сучасної апаратної бази, до якої входять мікроконтролер Arduino Uno, датчі вуглекислого, чадного газів, метану, температури та вологості (MQ-135, MQ-7, MQ-9, DHT11), Wi-Fi модуль ESP8266, LCD-дисплей. Для збереження, візуалізації, агрегації показів датчів використано платформу Інтернету речей ThingSpeak, а для користувача цієї системи розроблено інтерфейс мобільного додатка, який дозволяє відобразити поточні покази датчів пристрою, побудувати графіки вмісту будь-якого з газів у повітрі, температури та вологості, керувати вентилятором.

Розроблене програмне забезпечення для платформи Arduino Uno у разі дотримання певних обмежень фізичної платформи може працювати на всіх подібних платформах Arduino. Мобільний додаток може працювати тільки на операційній системі iOS.

Отже, запропонована авторами система виявлення та очищення приміщення від шкідливих газів з використанням платформи Інтернету речей ThingSpeak та мобільного iOS-дodatка дасть змогу забезпечити зручність та високу якість керування вмістом шкідливих газів у повітрі приміщення, а також може стати основою для розроблення автоматичних систем керування аналогічного класу.

Список літератури

1. *Забруднення повітря* [Електронний ресурс]. Режим доступу: https://uk.wikipedia.org/wiki/Забруднення_повітря.
2. Ramasamy Jayaturugan, B. Kumaravel, S. Palanivelraja, M. P. Chockalingam, "Influence of Temperature, Relative Humidity and Seasonal Variability on Ambient Air Quality in a Coastal Urban Area" [Електронний ресурс], *International Journal of Atmospheric Sciences*, vol. 2013, Article ID 264046, 7 pages, 2013. Режим доступу: <https://doi.org/10.1155/2013/264046>
3. JunHo Jo, ByungWan Jo, JungHoon Kim, SungJun Kim, WoonYong Han, "Development of an IoT-Based Indoor Air Quality Monitoring Platform" [Електронний ресурс], *Journal of Sensors*, vol. 2020, Article ID 8749764, 14 pages, 2020. Режим доступу: <https://doi.org/10.1155/2020/8749764>
4. Okokpuije K., Noma-Osaghae E., Modupe O., John S. and Oluwatosin O. "A smart air pollution monitoring system", *International Journal of Civil Engineering and Technology*, vol. 9, pp. 799–809, 2018.
5. Parmar G., Lakhani S. and Chattopadhyay M. "An IoT based low cost air pollution monitoring system", in *2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE), Bhopal, India, October 2017*.
6. *Building the Internet of Things with the ESP8266 Wi-Fi Module and ThingSpeak* [Електронний ресурс]/ Режим доступу: <https://community.thingspeak.com/tutorials/esp8266/building-the-internet-of-things-with-the-esp8266-wi-fi-module-and-thingspeak/>.
7. Swift. *A powerful open language that lets everyone build amazing apps.* [Електронний ресурс] - Режим доступу: <https://www.apple.com/swift/>
8. *Collect Data in a New Channel* Режим доступу: <https://www.mathworks.com/help/thingspeak/collect-data-in-a-new-channel.html>

Reference

1. *Zabrudnenia povitria* [Elektronnyi resurs]. Rezhym dostupu: <https://uk.wikipedia.org/wiki>
2. Ramasamy Jayaturugan, B. Kumaravel, S. Palanivelraja, M. P. Chockalingam, "Influence of Temperature, Relative Humidity and Seasonal Variability on Ambient Air Quality in a Coastal Urban Area" [Electronic resource], *International Journal of Atmospheric Sciences*, vol. 2013, Article ID 264046, 7 pages, 2013. Access mode: <https://doi.org/10.1155/2013/264046>.

3. JunHo Jo, ByungWan Jo, JungHoon Kim, SungJun Kim, WoonYong Han, "Development of an IoT-Based Indoor Air Quality Monitoring Platform" [Electronic resource], *Journal of Sensors*, vol. 2020, Article ID 8749764, 14 pages, 2020. Access mode: <https://doi.org/10.1155/2020/8749764>
4. Okokpujie K., Noma-Osaghae E., Modupe O., John S. and Oluwatosin O. "A smart air pollution monitoring system," *International Journal of Civil Engineering and Technology*, vol. 9, pp. 799–809, 2018.
5. Parmar G., Lakhani S. and Chattopadhyay M. "An IoT based low cost air pollution monitoring system", in *2017 International Conference on Recent Innovations in Signal processing and Embedded Systems (RISE)*, Bhopal, India, October 2017. <https://doi.org/10.1109/RISE.2017.8378212>
6. *Building the Internet of Things with the ESP8266 Wi-Fi Module and ThingSpeak* [Electronic resource]. Access mode: <https://community.thingspeak.com/tutorials/esp8266/building-the-internet-of-things-with-the-esp8266-wi-fi-module-and-thingspeak/>.
7. Swift. A powerful open language that lets everyone build amazing apps. [Electronic resource]. Access mode: <https://www.apple.com/swift/>
8. *Collect Data in a New Channel* Access mode: <https://www.mathworks.com/help/thingspeak/collect-data-in-a-new-channel.html>