

Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

Навчально-науковий інститут комп'ютерних наук
та інформаційних технологій
(повне найменування інституту, назва факультету (відділення))

Кафедра комп'ютерних наук
(повна назва кафедри (предметної, циклової комісії))

Магістерська кваліфікаційна робота

другий (магістерський)
(рівень вищої освіти)

на тему: **РОЗРОБЛЕННЯ СИСТЕМИ МОНІТОРИНГУ ІОТ
"SMARTBUILDING"**

Виконав: студент бкурсу групи КН-63м
спеціальності

122 "Комп'ютерні науки"

(шифр і назва напрямку підготовки, спеціальності)

Князь О.І.

(прізвище та ініціали)

Керівник проф. Процак Н.П.

(прізвище та ініціали)

Рецензент Князь ЮЄ.

(прізвище та ініціали)

Львів – 2024

Національний лісотехнічний університет України
(повне найменування вищого навчального закладу)

ННІ комп'ютерних наук та інформаційних технологій

Кафедра комп'ютерних наук

Рівень вищої освіти другий (магістерський)

Спеціальність 122 "Комп'ютерні науки"

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри

 Борецька І.Б.

"05" січня 2024 року

ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ

Князь Олег Ігорович

(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення системи моніторингу IoT "SmartBuilding"
керівник роботи проф. Процах Н.П.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від "13" лютого 2023 року № С-49

2. Термін подання студентом роботи "05" січня 2024 року

3. Вихідні дані до роботи _____

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) _____

Дослідити актуальність предметної області та проаналізувати вимоги користувачів до систем SmartHome/SmartBuilding. Провести дослідження ринку наявних технологій та рішень та можливостей застосування програмного забезпечення, протоколів зв'язку апаратних пристроїв для використання у проекті, в разі відсутності або неповноти таких рішень запропонувати власне. Провести аналіз можливостей масштабування та інтеграцій. Розробити загальну структуру, бізнес модель та описати взаємодію між компонентами системи. Описати загальний процес налаштування системи та основних фізичних та програмних елементів. Розробити конфігураційні файли та в разі використання власних пристроїв розробити програмний код.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

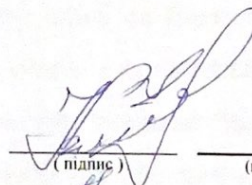
Структурна схема системи Homeassistant, Принципова схема під'єднання ESP32-WROOM

6. Дата видачі завдання "15" лютого 2023 року

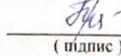
КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Строк виконання етапів роботи	Примітка
1	Дослідження актуальності предметної області	15.03.2023	виконано
2	Визначення вимог до систем SmartHome/SmartBuilding	30.03.2023	виконано
3	Дослідження наявних протоколів зв'язку та комунікації	12.04.2023	виконано
4	Дослідження ринку на наявність та відповідність потребам сенсорів та актуаторів	23.04.2023	виконано
5	Розробка архітектури системи та її основних компонентів	01.05.2023	виконано
6	Дослідження можливостей інтеграції зі сторонніми сервісами	03.05.2023	виконано
7	Розгортання та налаштування одноплатного компютера RaspberryPi 3b+	24.06.2023	виконано
8	Налаштування HomeAssistant для інтеграції з TuYa/Moes пристроями ZigBee	30.08.2023	виконано
9	Конфігурування сторонніх сервісів, баз даних та системи візуалізації	12.10.2023	виконано
10	Програмування та налаштування ESP32-WROOM	01.11.2023	виконано
11	Збір та аналіз отриманих даних	30.11.2023	виконано
12	Опис розділів «Стан проблемної області», «Інформаційне та математичне забезпечення»	01.12.2023	виконано
13	Опис розділів «Технічне та програмне забезпечення» та «Розроблення стартап-проекту»	15.12.2023	виконано
14	Формування висновків та результатів дослідження, фіналізація роботи над пояснювальною запискою	30.12.2023	виконано

Студент


(підпис)

Керівник роботи


(підпис)

Князь О.І.

(прізвище та ініціали)

проф. Процак Н.П.

(прізвище та ініціали)

АНОТАЦІЯ

Дипломна робота містить 63 сторінки пояснювальної записки, 13 рисунків, 3 таблиці, 2 додатки, 22 джерела.

Дипломна робота присвячена розробці та впровадженню інтелектуальної системи керування будинком SmartBuilding, яка базується на технологіях Інтернету речей (IoT). Метою цієї роботи є створення комплексного рішення для моніторингу та автоматизації технологічних процесів у розумному будинку.

Результати даної роботи можуть бути корисними для подальшого розвитку систем управління будинками з використанням технологій IoT та надають можливість власникам розумних будинків оптимізовано та ефективно керувати різними аспектами їхнього проживання.

Ключові слова: IoT, Docker, Grafana, Raspberry Pi, NodeMCU, ESP32, ESP8266, ZigBee, Home Assistant, Bluetooth, BLE, RFID, NFC

ABSTRACT

The diploma thesis comprises 63 pages of explanatory notes, 13 figures, 3 tables, 2 appendices, and 22 sources.

The diploma thesis is dedicated to the development and implementation of an intelligent SmartBuilding management system based on Internet of Things (IoT) technologies. The goal of this work is to create a comprehensive solution for monitoring and automating technological processes in a smart home.

The results of this work can be valuable for the further development of building management systems using IoT technologies, offering homeowners the ability to efficiently and effectively control various aspects of their living environment.

Key words: IoT, Docker, Grafana, Raspberry Pi, NodeMCU, ESP32, ESP8266, ZigBee, Home Assistant, Bluetooth, BLE, RFID, NFC.

Технічне завдання

Розробити систему моніторингу IoT "SmartBuilding" з використанням комплексного підходу до організації робочого процесу та взаємодії з елементами та сторонніх виробників. Розділити реалізацію на серверну та клієнтську частини. Клієнтська частина може реалізовуватись на мікроконтролерах та одноплатних комп'ютерах. Забезпечити гарантування доставки повідомлень між елементами системи. Використати найменш економічно затратні технології та протоколи зв'язку. (HomeAssistantта ZigBee).

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ.....	6
ВСТУП.....	7
РОЗДІЛ 1 СТАН ПРОБЛЕМНОЇ ОБЛАСТІ.....	10
1.1 Актуальність предметної області.....	10
1.2 Аналіз предметної області.....	11
1.2.1 Визначення вимог користувачів до системи SmartHome.....	13
1.2.2 Вибір пристроїв та датчиків SmartHome.....	16
1.2.3 Зв'язок та протоколи.....	17
1.2.4 Управління та інтерфейс.....	21
1.2.5 Інтеграція зі сторонніми сервісами.....	22
1.2.6 Масштабування системи.....	23
1.3 Висновки до розділу.....	24
РОЗДІЛ 2 ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ.....	25
2.1 Концептуальна модель.....	25
2.2 Бізнес-логіка.....	26
2.3 Користувацький інтерфейс.....	27
2.4 Проектування і розробка бази даних.....	31
2.5 Інтеграція з зовнішніми сервісами.....	31
2.6 Висновки до розділу.....	33
РОЗДІЛ 3. МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ.....	34
3.1 Алгоритми нечіткої логіки.....	34
3.2 Застосування алгоритмів нечіткої логіки в системі SmartHome.....	35
3.2 Практичне застосування.....	36
3.3 Алгоритм Мамдані та математичне трактування.....	38
3.4 Висновки до розділу.....	39
РОЗДІЛ 4. ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ.....	40
4.1 Побудова архітектури системи.....	40
4.2 Огляд апаратних засобів.....	42
4.3 Налаштування баз даних.....	48

4.4	Опис користувацького інтерфейсу.....	50
4.5	Розробка власного сенсора температури.....	51
4.6	Висновки до розділу.....	52
РОЗДІЛ 5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЄКТУ.....		53
5.1	Опис ідеї проєкту, визначення загальних напрямів використання потенційного товару чи послуги, а також їх відмінність від конкурентів...	53
5.2	Технологічна здійсненність ідеї проєкту.....	56
5.3	Аналіз ринкових можливостей запуску стартап-проєкту.....	57
5.4	Висновки до розділу.....	60
ВИСНОВКИ.....		61
ДОДАТКИ.....		65
Додаток А Код ESP32 з DS18B20.....		65
Додаток Б Код часткової конфігурації HomeAssistant.....		67
Додаток В Конфігураційний блок HomeAssistant для використання нечіткої логіки для керування температурою в кімнаті.....		68

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ

Кбіт/с Мбіт/с Гбіт/с – Скорочення похідних одиниць виміру великих швидкостей передачі інформації. Відповідно: Кілобіти за секунду, Мегабіти за секунду, Гігабіти за секунду.

2G/3G/4G/5G – Найменування поколінь стільникового зв'язку, де G, з англ. Generation – покоління.

BLE – Бездротовий протокол зв'язку з пониженим споживанням енергії (англ. Bluetooth Low Energy).

Bluetooth - Бездротовий протокол зв'язку на короткі відстані.

IT - Інформаційні технології.

IoT - Інтернет речей (англ. Internet of Things) .

IIoT - Промисловий Інтернет речей (англ. Industrial Internet of Things).

KNX - комунікаційна шина, що використовується для атоматизації будівель.

LAN - локальна мережа (англ. Local Area Network).

MQTT – високошвидкісний протокол обміну повідомленнями на рівні машина-машина (англ. Message Queue Telemetry Transport).

NFC – технологія бездротового зв'язку малого радіусу дії (англ. Near field communication).

RFID - спосіб автоматичної ідентифікації об'єктів (англ. Radio Frequency Identification).

SmartHome – дослівно з англ. Розумний Дім.

SmartBuilding – дослівно з англ. Розумна Будівля.

Wi-Fi – технологія бездротової локальної мережі невеликого радіусу дії.

Zigbee – відкритий стандарт мереж бездротових пристроїв з низьким споживанням енергії.

Z-Wave – пропрієтарний стандарт мереж бездротових пристроїв з низьким споживанням енергії.

ВСТУП

IoT або ж Інтернет речей – система фізичних об'єктів («речей»), взаємопов'язаних між собою за допомогою вбудованих датчиків, програмного забезпечення та/або інших технологій. Цей зв'язок потрібний для того, щоб передавати дані на інші пристрої в системі або в інші системи через Інтернет. Простіше кажучи, фізичні об'єкти виходять в Інтернет, щоб відправити інформацію чи прийняти її. Такі системи за останні кілька років набувають все більше актуальності та поширення, оскільки дозволяють задовольнити попит що зростає як бізнесу, так і окремо взятих індивідуальних користувачів. Складність та масштабність подібних систем зростає щороку, що своєю чергою вимагає більш складних систем для моніторингу, аналізу та прогнозування, чим свою чергу зміщують акцент на використання готових або підготовлених до інтеграції систем.

Актуальність проблеми. Контроль та оптимізація використання природних ресурсів, налаштування комфортних кліматичних умов приміщень набуває все більше і більше значення в умовах здорожчання енергоресурсів та зменшення логістичної доступності в силу цього. Потреба в таких системах з'являється не лише для великих підприємств, а й для не великих житлових і виробничих будівель, де вартість впровадження подібних систем дозволяє отримати нижчий поріг входження та менші капітальні затрати. При чому доступ до систем такого рівня можна здійснювати як безпосередньо з локальних терміналів так віддалено.

Об'єктом досліджень в даній роботі є система взаємодії розподілених пристроїв в просторі, що імплементовані на базі фізичних локальних пристроїв, а також програмних додатках, що створюють віртуальну інфраструктуру для взаємодії вказаних пристрої та аналізу температурних(та інших) залежностей в процесі дії джерел теплового або конвекційного випромінювання на замкнуті приміщення.

Предметом дослідження є інформаційна система аналізу температурних залежностей в закритих приміщеннях з використанням джерел теплового випромінювання що використовують природні ресурси в якості джерел енергії.

Метою даної роботи була розробити просту та ефективну систему температурного моніторингу замкнутих середовищ з джерелами теплового випромінювання. Отримана інформація дозволяє легко відлаштувати кліматичну систему для отримання оптимального співвідношення комфорт-енерго затрати.

Основні завдання що були розроблені:

1. охарактеризовано об'єкт дослідження і сформульовані задачі для розробки;
2. проведено огляд доступних сфер застосування і можливостей наявних систем;
3. досліджено способи комунікації та протоколи обміну даними;
4. спроектовано архітектуру системи
5. спроектовано структуру бази даних системи;
6. імплементовано апаратну систему що дозволяє моніторити температуру з масиву сенсорів
7. розроблено програмний продукт, що реалізовує поставлені задачі, використавши відповідний інструментарій та зовнішні рішення і бібліотеки;
8. проведено тестування розробленого програмного рішення з метою перевірки працездатності системи моніторингу температури замкнутого приміщення

Новизна роботи полягає в тому що це єдина система що дозволяє відслідковувати масиви температурних значень для заданого приміщення та дозволяє отримати цілісну картину поширення градієнта температур в приміщенні.

Практичне значення отриманих результатів. Система що розроблена дозволяє легко вирішувати задачі налаштування використання теплоносіїв та теплових ресурсів за допомогою аналізу температурних змін в замкнутих приміщеннях, квартирах, виробничих ангарах та запобігати перевитраті ресурсів.

РОЗДІЛ 1 СТАН ПРОБЛЕМНОЇ ОБЛАСТІ

1.1 Актуальність предметної області.

Актуальність предметної області, пов'язаної з системами візуалізації та керування пристроями Інтернету речей (IoT), полягає у низці факторів:

- Розширені мережі LPWA, особливо ті, що використовують технологію NB-IoT. Впровадження NB-IoT почалося, і кількість підключень зростає на 61% р/р, завдяки широкому спектру реалізацій, особливо в області обліку води та газу – дозволило розширення набір технологічних процесів які можна відслідковувати. [1]
- Користувачі відходять від застарілих 2G/3G до 4G/5G IoT. З'єднання 4G IoT зросло на 24% завдяки більш широкому поширенню чіпсетів на основі LTE Cat 1, Cat 4 і Cat 6. Для багатьох реалізацій LTE Cat 1b стає альтернативою вищезгаданим технологіям LPWA – збільшилися транспортні канали і об'єми даних що надсилаються між пристроями, виникла можливість аналізувати отримані дані з рухомих об'єктів та надсилати в зворотньому порядку команди контролю.
- Дефіцит мікросхем продовжував сповільнювати відновлення ринку.
- COVID-19 продовжував впливати на продукти та ланцюги поставок.

З 2022 року з'явилися нові тренди, що додатково підвищують актуальність вказаної проблеми та притаманні, в більшості, умовам України [1]:

- Інфляція, як на світовому ринку так і в банківській системі України - через значне зниження ВВП та залученість іноземних коштів.
- Війна, що триває, є безпрецедентним каталізатором економічних процесів та впливу на ланцюги постачання.
- Боротьба за цифрові таланти. В умовах активної війни, частина населення були змушені залишити державу, що в свою чергу створює додатковий незадоволений попит на галузевих інженерів.

- Збільшення вартості енергоресурсів та тимчасова недоступність частини з них.

Додаткові фактори, що значно впливають на актуальність проблеми, виникають з самої природи технології:

- Значна кількість пристроїв, що входять в мережу;
- Варіативність технологій та реалізацій, поколінь, версій, тощо;
- Значна кількість виробників з пропріетарними технологіями;
- Різномформатна природа даних;
- Кількість даних, що обробляється та накопичується для подальшого історичного аналізу значень;
- Безпека даних і можливість застосування їх для неправомірних дій.

Відповідно до зазначених факторів, системи, що вимагаються, повинні володіти рядом засобів та функціональністю, що дозволять легко поєднувати пристрої в більш складні системи та конфігурувати їх, швидко надавати безпечний доступ до даних і візуалізацій, дозволяти ефективно керувати активними пристроями, швидко отримувати необхідну аналітичну інформацію, гарантувати цілісність і збереження даних [2].

Переважає більшість наявних на ринку подібних систем або мають надлишкову складність, або не володіють повним набором інструментів для виконання поставлених задач.[8]

1.2 Аналіз предметної області

Аналіз предметної області системи IoT для SmartHome дозволяє з'ясувати особливості цього сегмента та виявити ключові фактори, які використанні для успішного проектування. Деякі аспекти, які враховано:

Визначення вимог користувачів: Розуміння потреб та очікувань користувачів SmartHome є важливою вимогою. Визначення, які функції та можливості очікуються від системи SmartHome. Варіанти можуть бути різні: автоматизація освітлення, кондиціонуванням повітря, безпека,

електроприладами, управління опаленням. *Для поточної системи основні вимоги: Моніторинг температури, керування опалювальними приладами.*

Вибір пристроїв та датчиків: Визначення, які пристрої та датчики потребують використання в поточній системі SmartHome. Це можуть бути елементи освітлення, термостати, датчики руху, камери спостереження, розетки з можливістю дистанційного управління тощо. *Для поточної системи основні типи: термостати, термосенсори, комутатори та комунікатори, з низьким рівнем споживання та високим рівнем надійності.*

Зв'язок та протоколи: Визначення, які засоби зв'язку використовуватимуться для підключення пристроїв IoT у системі SmartHome. Це можуть бути дротові та бездротові протоколи Wi-Fi, Bluetooth, Zigbee, Z-Wave або інші протоколи. *Для поточної систем: для ядра системи використано дротові протоколи які здатні забезпечити високі швидкості обміну повідомленнями, ядро розміщено централізовано. Для кінцевих пристроїв надано перевагу бездротовим енерго-ощадливим або енерго-незалежним пристроям для зменшення капіталовкладень.*

Управління та інтерфейс: Визначення способу взаємодії кінцевого користувача з системою SmartHome. Це може бути мобільний додаток, веб-інтерфейс, апаратні стаціонарні консолі або панелі. Головна задача - забезпечити зручний та інтуїтивно зрозумілий інтерфейс для керування та моніторингу різних функцій SmartHome. *Для поточної системи передбачено використання в якості основног інтерфейса це веб-інтерфейс, в якості допоміжного використано апаратних пристроїв, консолей для візуалізації даних.*

Безпека: Забезпечення безпеки є важливим аспектом систем SmartHome. Тому шифрування даних, двофакторна аутентифікація, віддалений доступ з використанням захищених протоколів та системи контролю доступу до пристроїв необхідно використати по максимуму, щоб унеможливити втрату даних та прив'язку даних до конкретних об'єктів та

суб'єктів системи. Для поточної системи використано вбудованих систем захисту що надаються в межах використаних протоколів або пристроїв.

Інтеграція зі сторонніми сервісами: Можливість інтеграції системи SmartHome зі сторонніми сервісами, такими як погодні додатки, сервіси безпеки, енергопостачання або інші розумні пристрої. Це дозволить розширити функціональність та забезпечити більш гнучкий досвід користувачів. Для поточної системи передбачено інтеграція із сторонніми хмарними сервісами і системати різних протоколів зв'язку.

Масштабування системи: Система функціонує, на разі зі значною кількістю пристроїв (близько 20).

1.2.1 Визначення вимог користувачів до системи SmartHome

Сфера застосування IoT, досить широка, та включає майже всі сфери життя де присутні інформаційні технології. В свою чергу це може покладати додаткові вимоги що стосуються конкретної системи. Розглянуто загальні сфери застосування та більш детально проаналізовано вимоги та можливості, які релевантні до поточної системи:

IoT-пристрої в агрокультурі. IoT-пристрої в агрокультурі використовуються для покращення ефективності та продуктивності сільськогосподарських операцій:

- Моніторинг умов зростання
- Полив та дозування речовин;
- Моніторинг урожаю;
- Управління стадами тварин
- Прогнозування та аналітика.

Інтернет речі у промисловості. IoT-пристрої в промисловості використовуються для створення "розумних" фабрик і оптимізації виробничих процесів:

- Моніторинг та діагностика обладнання

- Оптимізація виробничих процесів
- Безпека та охорона
- Прогнозування та аналітика
- IoT-пристрої допомагають підвищити продуктивність, ефективність та безпеку в промисловості. [15,16]

IoT в роздрібній торгівлі. Internet of Things (IoT) технології широко використовуються в роздрібній торгівлі для оптимізації роботи та покращення взаємодії з клієнтами:

- Управління запасами;
- Моніторинг якості продуктів;
- Персоналізований досвід покупок;
- Безкоштовні каси;
- Смарт-теги: RFID-теги ;
- IoT в логістиці.

Інтернет речей у сфері охорони здоров'я. Інтернет речей (Internet of Things, IoT) також активно використовується в галузі охорони здоров'я, що сприяє покращенню догляду за пацієнтами, забезпеченню їх безпеки та підвищенню ефективності медичних служб:

- Моніторинг пацієнтів в реальному часі;
- Телемедицина;
- Системи нагадувань;
- Моніторинг медичного обладнання;
- Дослідження та розвиток.

Пристрої, які носять на собі. Носимі IoT-пристрої дозволяють змінити спосіб стеження за здоров'ям, фітнесом та спортом, та змінити способи використання звичних повсякденних речей у незвичний спосіб. Деякі варіанти використання:

- Фітнес-трекери;
- Смарт-годинники;

- Вбудовані пристрої;
- AR/VR гарнітури.

Наступна сфера застосування пристроїв напряду пов'язана з цілями системи що запроектована, тому розглянем які щезавдання, можуть вирішуватися в галузі **SmartHome/SmartBuilding**, більш детально.

IoT використовується в розумних будинках та квартирах для автоматизації та контролю різноманітних систем та пристроїв. Типові сценарії використання, які можна взяти за основу для системи **SmartHome**:

- *Освітлення*: Системи освітлення IoT дозволяють вмикати, вимикати або регулювати освітлення у будинку через мобільний додаток або голосовий помічник. Можна налаштувати розклад освітлення, створювати схеми підсвідки на основі присутності людей в межах того чи іншого приміщення. [9,10]

- *Безпека*: Системи безпеки IoT включають в себе відеокамери, датчики руху, датчики диму та інші пристрої, які можуть сповіщати про небезпеку або небажану активність у будинку через мобільний додаток. Можна спостерігати за будинком у режимі реального часу або отримувати сповіщення про небезпеку, навіть коли господар знаходиться поза межами дому.

- *Управління енергією*: IoT може допомогти ефективно використовувати енергію в будинку. Наприклад, встановлені розумні розетки або розумні заслонки, або перемикачі, дозволяють автоматично вмикати електроприлади ,або регулювати рівень проникнення сонячного світла через жалюзі, щоб знизити витрати на електроенергію.

- *Управління побутовими пристроями*: Завдяки IoT можна керувати різними побутовими пристроями, такими як холодильник, пральна машина, праска тощо. Вони можуть бути підключені до мережі Інтернет, щоб віддалено управляти ними, отримувати сповіщення про завершення циклу прання або перевіряти запаси продуктів у холодильнику.[11]

- *Терморегулювання:* Системи IoT дозволяють керувати температурою та кліматом у будинку через термостати, якими можна керувати за допомогою мобільного пристрою. Можна налаштувати графік опалення та охолодження або віддалено керувати ними, щоб ефективно використовувати енергію та знизити витрати.

1.2.2 Вибір пристроїв та датчиків SmartHome

Вибір пристроїв та датчиків для системи SmartHome визначається конкретними потребами та функціональними вимогами нашого проекту. Однак, ось деякі загальні типи пристроїв та датчиків, які ще часто використовуються у системах SmartHome:

Елементи освітлення: Це можуть бути розумні лампи, світильники або розетки з можливістю дистанційного управління. Вони дозволяють регулювати яскравість освітлення, включати і виключати світло за допомогою мобільного додатку або голосових помічників.

Термостати та системи опалення: Ці пристрої дозволяють контролювати температуру в приміщенні. Вони можуть бути підключені до центральної системи опалення та кондиціонування повітря та дозволяють регулювати температуру з мобільного додатку або автоматично налаштовувати режими в залежності від присутності людей або погодних умов.

Датчики руху та присутності: Вони виявляють рух або присутність в приміщенні та активують відповідні дії, наприклад, увімкнення освітлення або сповіщення про незвичайну активність.

Датчики вологості та температури: Вони моніторять рівень вологості та температуру у приміщенні і можуть активувати вентиляцію або ввімкнути витяжку при перевищенні заданих значень.

Датчики диму та вуглекислого газу: Вони виявляють наявність диму або високої концентрації вуглекислого газу і вмикають сигнали тривоги або сповіщення для захисту від пожеж або витоку газу.

Камери спостереження: Вони дозволяють вам моніторити приміщення або зовнішній простір за допомогою відеозапису та можуть підключатися до системи сповіщення або зберігання записів.

Розумні розетки та роз'єми: Вони дозволяють включати або виключати електричні пристрої з дистанції та контролювати їх споживання енергії.

WiFi мережа розподілена по території об'єкта для надання можливості підключення. Дальність зв'язку: Залежить від типу кабелю та технології що використовується. Зазвичай декілька сотень метрів по кабелю Cat5e або Cat6, але можна використовувати підсилювачі сигналу для більшої дальності.

1.2.3 Зв'язок та протоколи

Протоколи комунікації та обміну даними, що використовуються в IoT, можна класифікувати на дротові (провідні) та бездротові (безпроводні). Розглянемо детальніше ті протоколи які здатні виконати поставлені задачі:

- *ядро системи використовує дротові протоколи які здатні забезпечити високі швидкості обміну повідомленнями, ядро розміщено централізовано, максимальна відстань між елементами не більша 100м, в межах одного технологічного приміщення – для поточної системи.*
- *кінцеві пристрої переважно взаємодіють за допомогою бездротових енерго-ощадливих протоколів, розташовані по будинку в різних кімнатах.*

Дротові протоколи комунікації:

Ethernet: Це найпоширеніший протокол провідної мережі, який використовується для передачі даних на великі відстані з високою швидкістю. Використовується для побудови локальних мереж (LAN) і підключення до Інтернету.

Швидкість передачі даних: Від 10 Мбіт/с (Ethernet) до 100 Гбіт/с (Ethernet-100G).

Дальність зв'язку: Залежить від типу кабелю та технології що використовується. Зазвичай декілька сотень метрів по кабелю Cat5e або Cat6, але можна використовувати підсилювачі сигналу для більшої дальності.

Споживання енергії: Залежить від активності мережі та підключених пристроїв. Комутатори та маршрутизатори можуть мати різні рівні споживання енергії.

Практичне використання: використано для взаємодії елементів системи на мережевому рівні для побудови локальної розподіленої мережі на об'єкті, високорівне під'єднання елементів таких як **RaspberryPi3+**, **WiFi MikroTik CapsMAN AC 2+5G**, **Router MikroTik Rbn951G-2HnD**

USB (Universal Serial Bus): Цей протокол використовується для підключення пристроїв до комп'ютера або інших електронних пристроїв. Він дозволяє передавати дані, а також забезпечує живлення пристроїв.

Швидкість передачі даних: Залежить від версії USB. Від 1,5 Мбіт/с (USB 1.0) до 20 Гбіт/с (USB 3.2 Gen 2x2).

Дальність зв'язку: Зазвичай до 5 метрів для USB 2.0 та до 3 метрів для USB 3.0/3.1/3.2.

Споживання енергії: Високе споживання енергії, оскільки USB надає живлення пристроям.

Практичне використання: в переважній більшості живлення розподілених пристроїв **ZigBee Universal Router**, **ZigBee NetworkExtender**

I²C (Inter-Integrated Circuit): Цей протокол дозволяє зв'язати кілька електронних пристроїв на короткій відстані, використовуючи два проводи для передачі даних.

Швидкість передачі даних: Залежить від швидкості I²C та режиму передачі. Зазвичай до 400 Кбіт/с (стандартний режим) або до 3,4 Мбіт/с (High-Speed Mode).

Дальність зв'язку: Залежить від опору кабелю та кількості підключених пристроїв. Зазвичай до кількох метрів.

Споживання енергії: Низьке споживання енергії, оскільки I²C використовує два проводи для передачі даних.

Практичне використання: комунікація з сенсорами температури розміщеними на шині **Датчик температури DS18B20**.

Powerline Communication (PLC): Цей протокол використовує електричну мережу будинку для передачі даних. Він дозволяє використовувати наявну електричну інфраструктуру для створення мережі з великим покриттям.

Швидкість передачі даних: Залежить від стандарту та умов електричної мережі. Зазвичай до 200 Мбіт/с для стандарту HomePlug AV2.

Дальність зв'язку: Залежить від якості електричної мережі та розташування пристроїв. Зазвичай до декількох сотень метрів.

Споживання енергії: Споживання енергії залежить від конкретного пристрою і стандарту, але зазвичай помірне.

Практичне використання: комунікація розрізних приміщень об'єкта з використанням **адаптерів Powerline TP-LINK TL-PA7017P KIT**

Бездротові протоколи комунікації.

Кількість бездротових протоколів комунікації досить значна, але для ефективного вирішення поставлених цілей перед проектом достатньо розглянути більш детально лише ті які активно використовуються та задовільняють критеріям надійності та достатності об'єму передачі даних.

Таблиця 1 Протоколи комунікації

Протокол	Швидкість передачі даних	Дальність зв'язку	Споживання енергії	Використання
Wi-Fi	Від 11 Мбіт/с до декількох Гбіт/с	Декілька десятків метрів	Середнє до високе	Локальні мережі, Інтернет
Bluetooth	Від 1 Мбіт/с до 3 Мбіт/с	До 10 метрів	Низьке до середнього	Сенсори, розумні годинники, аудіо-

				та інші пристрої
Zigbee	До 250 Кбіт/с	Декілька десятків метрів	Низьке до середнього	Розумний будинок, освітлення, контроль доступу
BLE	Від 125 Кбіт/с до 2 Мбіт/с	До 100 метрів	Дуже низьке	Датчики, фітнес-трекери
MQTT	Залежить від умов мережі та реалізації, може досягати кількох Мбіт/с	Залежить від використовуваних протоколів транспорту	Залежить від конкретної реалізації та пристрою	Обмін повідомленнями в мережах IoT

Наведені вище бездротові протоколи [20,21] комунікації покривають значний діапазон відстаней та швидкостей передачі даних та мають різні характеристики енергоефективності

Практичне використання:

Wi-Fi. – бездротова взаємодія компонентів системи в межах локальної мережі та ядра системи, використано в будинку

ZigBee – взаємодія кінцевих бездротових компонентів системи, термодатчики та термоголовки, зняття показників з масивів сенсорів що розміщені розподілено по приміщенням.

Bluetooth – локальна взаємодія з компонентами системи для початкової ініціалізації та налаштування ZigBee компонентів

MQTT - міжкомпонента взаємоді в межах ядра мережі з системою моніторингу та збору даних Home Assistant

1.2.4 Управління та інтерфейс

Мобільний додаток: Більшість систем SmartHome надають мобільний додаток, який дозволяє керувати пристроями та функціями вашої системи зі смартфона або планшета. Цей інтерфейс зазвичай надає зручний спосіб вмикати/вимикати пристрої, регулювати освітлення, температуру, переглядати статуси та отримувати сповіщення. Взаємодія з системою на поточному рині реалізовано через мобільний додаток.

Голосові помічники: За допомогою голосових помічників, таких як Amazon Alexa, Google Assistant або Apple Siri, ви можете керувати вашою системою SmartHome за допомогою голосових команд. Ви можете говорити з голосовим помічником і вказувати, що саме ви хочете зробити, наприклад, вмикати світло, регулювати температуру або запускати автоматичні сценарії. Для поточного проекту інтеграція не налаштована, але HomeAssistant дозволяє це зробити в майбутньому.

Веб-інтерфейс: Деякі системи SmartHome надають веб-інтерфейс, який дозволяє керувати пристроями та налаштуваннями вашої системи через веб-браузер. Це може бути зручно, якщо ви хочете мати доступ до вашої системи з комп'ютера або великого екрану. Веб інтерфейс наразі відсутній, може бути налаштований додатково, основна причина – відсутність зовнішнього доступу ззовні мережі.

Автоматичні сценарії: Багато систем SmartHome дозволяють створювати автоматичні сценарії або розклади дій. Ви можете налаштувати специфічні умови або часові розклади, коли певні пристрої або функції повинні виконувати певні дії. Наприклад, ви можете налаштувати автоматичне вмикання освітлення та вимикання опалення певного часу дня або в реакції на датчик руху. В системі налаштовано автоматичний сценарій відкриття-закриття жалюзів-ролетів на сході-заході сонця.

Дистанційний доступ: Багато систем SmartHome надають можливість дистанційного доступу до системи через Інтернет. Через яку можна керувати

пристроями та переглядати статуси з будь-якого місця з підключенням до Інтернету, використовуючи мобільний додаток або веб-інтерфейс. Обов'язковий критерій роботи – доступність з зовнішньої мережі, що вирішується переналаштуванням роутера і купівлею додаткових послуг у провайдера.

1.2.5 Інтеграція зі сторонніми сервісами

Інтеграція системи SmartHome зі сторонніми сервісами дозволяє розширити функціональність та забезпечити більш гнучкий досвід для користувачів. Ось деякі можливості інтеграції SmartHome зі сторонніми сервісами:

Погодні сервіси: Інтеграція з погодними сервісами дозволяє отримувати актуальні погодні дані і використовувати їх для автоматичного налаштування системи SmartHome. Наприклад, система може автоматично реагувати на зміни погоди, включати/вимикати систему кондиціонування повітря, адаптувати освітлення тощо.

Системи безпеки: Інтеграція з системами безпеки, такими як системи відеоспостереження, системи сповіщення про вторгнення або системи контролю доступу, дозволяє вам моніторити та керувати безпекою вашого домашнього середовища з одного центрального інтерфейсу SmartHome.

Розумні асистенти: Інтеграція з голосовими помічниками, такими як Amazon Alexa, Google Assistant або Apple Siri, дозволяє вам керувати вашою системою SmartHome за допомогою голосових команд. Ви можете вмикати/вимикати світло, регулювати температуру, управляти розетками та багато іншого, просто говорячи зі своїм голосовим помічником.

Системи енергозбереження: Інтеграція зі системами енергозбереження дозволяє вам моніторити та керувати споживанням енергії у вашому домі. Ви можете встановлювати режими енергозбереження, автоматично вимикати електроприлади відповідно до графіку або налаштовувати розклад роботи електроприладів.

Системи розумного голосового домашнього кінотеатру: Інтеграція з системами розумного голосового домашнього кінотеатру дозволяє вам керувати аудіо та відео компонентами вашої розважальної системи з центрального інтерфейсу SmartHome. Ви можете вмикати/вимикати телевізор, регулювати гучність, вибирати контент і багато іншого за допомогою голосових команд або мобільного додатку.

Системи розумних пристроїв: Інтеграція з розумними пристроями, такими як розумні холодильники, розумні пральні машини, розумні пристрої для кухні та інші, дозволяє вам отримувати статус та керувати цими пристроями зі свого інтерфейсу SmartHome. Наприклад, ви можете отримувати сповіщення про стан пральної машини або керувати режимами холодильника.

1.2.6 Масштабування системи

Масштабування системи SmartHome важливо, оскільки з часом виникне необхідність додати нові пристрої, функції та розширити покриття системи. Ось кілька аспектів, які варто врахувати для ефективного масштабування системи SmartHome:

Гнучка архітектура: При проектуванні системи SmartHome враховується гнучкість та масштабованість її архітектури. Використовується модульний підхід, щоб можна було легко додавати та інтегрувати нові пристрої та функції без необхідності переробки всієї системи.

Мережева інфраструктура: Використання маршрутизаторів та комутаторів з достатньою пропускну здатністю та можливістю підключення багатьох пристроїв.

Системи керування: Використання систем керування SmartHome, які можуть централізовано керувати всіма пристроями і функціями системи. Це дозволить легко додавати, налаштовувати та керувати новими пристроями без необхідності окремо налаштовувати кожен з них.

Сумісність та стандарти: Необхідність вибору пристроїв та технологій, які підтримують відкриті стандарти та інтероперабельність. Це дозволить

додавати різні пристрої від різних виробників до системи без обмежень та проблем інтеграції.

Масштабування хмарних служб: При використанні віртуальних сервісів для зберігання даних та керування системою, необхідно врахувати, що служба може масштабуватися з ростом кількості пристроїв та обсягу даних.

Планування майбутнього: Передбачення подальшого зростання кількості пристроїв, обсягу даних та функціональних вимог, потенційних потреб та запасних резервів в системі.

Тестування та зворотний зв'язок: Після масштабування системи SmartHome важливо провести відповідні тестування, щоб переконатися, що всі пристрої та функції працюють належним чином. Зверніть увагу на зворотній зв'язок користувачів та вчасно вносьте вдосконалення та зміни для оптимізації системи.

1.3 Висновки до розділу

У цьому розділі проведено аналіз предметної області, проведено порівняння, існуючих на даний момент технологій та протоколів зв'язку. Проаналізовано актуальність та важливість розвитку інтелектуальних систем "Розумний Будинок", що є відповіддю на зростаючу потребу в ефективному управлінні ресурсами, підвищенні комфорту та безпеки у побуті. Особлива увага приділено потенціалу IoT технологій, які стають все більш інтегрованими в повсякденне життя, підкреслюючи необхідність розвитку адаптивних та інтелектуальних систем керування домогосподарствами.

В розділі продемонстровано важливість комплексного підходу до проектування систем SmartHome, включаючи аналіз сучасних тенденцій, технологій та вимог до інтелектуальних систем. Наголошено на необхідності інтеграції різних технологічних рішень, масштабування систем та забезпечення їх безпеки і надійності. Описано аспекти які необхідні для глибокого розуміння широкого спектра факторів, що впливають на ефективність та функціональність систем "Розумний Будинок".

РОЗДІЛ 2 ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

2.1 Концептуальна модель

Концептуальна модель системи SmartHome з веб-інтерфейсом для моніторингу системи опалення та кліматичних показників в приміщенні включає такі компоненти та їх функції:

1. Центральний контролер: Це основний компонент системи SmartHome, який відповідає за керування всіма пристроями та функціями. Веб-інтерфейс дозволяє користувачу взаємодіяти з центральним контролером за допомогою веб-браузера.

2. Веб-інтерфейс: Веб-інтерфейс є інтерфейсом для користувача, який доступний через веб-браузер. Він дозволяє моніторити та керувати системою опалення та кліматичними показниками в приміщенні. Користувач може переглядати температуру, вологість, налаштування системи опалення та здійснювати відповідні зміни.

3. Датчики температури та вологості: Датчики розміщуються в різних частинах приміщення для моніторингу температури та вологості. Вони передають дані про кліматичні показники до центрального контролера, які потім відображаються в веб-інтерфейсі для моніторингу.

4. Система опалення: Цей компонент включає опалювальні пристрої, наприклад котли, радіатори або системи кондиціонування повітря. Веб-інтерфейс дозволяє користувачу керувати режимами опалення, налаштовувати температуру, запускати або зупиняти систему опалення.

5. Графіки та звіти: Веб-інтерфейс може включати можливість перегляду графіків та звітів про кліматичні показники. Користувач може аналізувати історичні дані, визначати тренди та приймати рішення щодо налаштування системи опалення для досягнення оптимального комфорту та енергоефективності.

6. Налаштування та розклади: Веб-інтерфейс дозволяє користувачу налаштовувати параметри системи опалення та створювати розклади роботи.

Користувач може встановлювати режими опалення для різних часових періодів або відповідно до власних потреб.

7. Сповіщення та автоматичні сценарії: Веб-інтерфейс може підтримувати сповіщення користувача про надмірну температуру, низьку вологість або інші події, пов'язані з кліматичними показниками. Крім того, він може дозволяти налаштовувати автоматичні сценарії, які включають автоматичну зміну параметрів системи опалення відповідно до зазначених умов.

2.2 Бізнес-логіка

Бізнес-логіка для застосування системи SmartHome з веб-інтерфейсом для моніторингу системи опалення та кліматичних показників включає наступні аспекти:

- Керування опаленням та кліматичними параметрами: Основною функцією системи буде керування опаленням та налаштуванням кліматичних параметрів в приміщенні. Бізнес-логіка буде включати здатність встановлювати температуру, режими опалення, рівень вологості і контролювати їх зміну залежно від вимог.
- Графіки та аналітика: Система може збирати дані про кліматичні показники та забезпечувати графіки та аналітику. Це дозволяє аналізувати тренди, спостерігати за змінами клімату в приміщенні та визначати оптимальні налаштування для комфорту та енергоефективності.
- Розклади роботи та автоматизація: Система підтримує функцію розкладів роботи, що дозволяє заздалегідь встановлювати режими опалення для різних часових періодів. Крім того, можлива автоматизація, яка дає змогу системі самостійно реагувати на зміни температури, вологості або наявності людей у приміщенні.
- Сповіщення та повідомлення: Система надсилає сповіщення та повідомлення користувачеві щодо важливих подій, таких як перевищення

заданої температури або низький рівень вологості. Це допомагає забезпечити контроль та реагування на проблеми або незвичайні ситуації.

- **Управління доступом та безпекою:** Важливим аспектом бізнес-логіки є забезпечення безпеки системи та захисту від несанкціонованого доступу. Це включає автентифікацію користувачів, рівні доступу та шифрування комунікації для забезпечення конфіденційності та цілісності даних.

- **Інтеграція з іншими системами:** В деяких випадках можлива інтеграція з іншими системами SmartHome або зовнішніми сервісами. Наприклад, інтеграція з системою метеорологічних прогнозів може допомогти у встановленні оптимальних налаштувань опалення на основі передбачуваних погодних умов.

2.3 Користувацький інтерфейс

Користувацький інтерфейс системи візуалізації даних з пристроїв IoT SmartHome є важливим компонентом, який дозволяє користувачеві моніторити та керувати різними аспектами їхнього домашнього середовища. Ось кілька ключових елементів, які можуть бути включені в користувацький інтерфейс для візуалізації даних:

Дашборд: Дашборд є головною сторінкою інтерфейсу, де користувач може отримати огляд інформації про стан системи SmartHome. Він може містити різні віджети, такі як графіки температури, вологості, споживання енергії, статуси пристроїв та інші ключові показники.

Графіки та діаграми: Графіки та діаграми дозволяють візуально представити дані, отримані від пристроїв IoT. Наприклад, графіки температури можуть відображати зміну температури протягом певного періоду часу, а діаграми споживання енергії - розподіл споживання енергії за певні пристрої або часові інтервали.

Списки пристроїв та їх стан: Користувач може бачити список підключених пристроїв IoT SmartHome та їх поточний стан. Це може включати інформацію про вмикання/вимикання, режими роботи та інші параметри пристроїв.

Сповіщення та повідомлення: Інтерфейс може надавати можливість відправки сповіщень та повідомлень користувачеві. Наприклад, повідомлення про виявлення руху в зоні, сповіщення про незвичайну температуру або низький рівень батареї пристрою.

Керування пристроями та налаштування: Користувач може мати можливість керувати пристроями SmartHome через інтерфейс візуалізації даних. Це може включати вмикання/вимикання пристроїв, зміну налаштувань, налаштування розкладів роботи та інші дії, які дозволяють користувачеві контролювати їх домашнє середовище.

Дистанційний доступ: Користувач може мати можливість отримати дистанційний доступ до системи SmartHome через веб-інтерфейс. Це дозволяє контролювати та моніторити систему з будь-якого пристрою з підключенням до Інтернету, надаючи зручність та гнучкість управління.

Важливо, щоб користувацький інтерфейс був інтуїтивно зрозумілим, привабливим та легким у використанні, забезпечуючи зручний доступ до важливих даних та функцій системи SmartHome.

Деякі з прикладів таких візуалізацій:

Home Assistant: Home Assistant є відкритою системою автоматизації SmartHome, яка має вбудований веб-інтерфейс для візуалізації даних. Вона надає гнучкість налаштувань та можливість створення персоналізованих дашбордів з графіками, картами, списками пристроїв та іншими елементами.

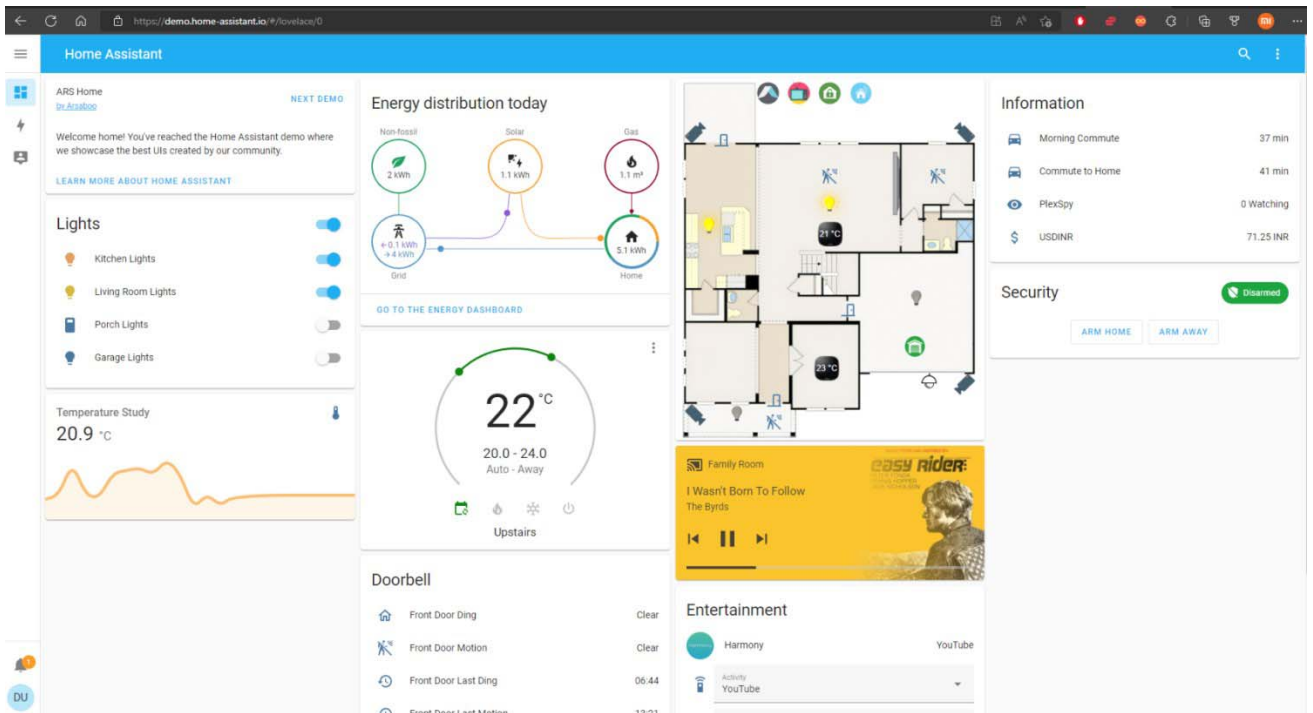


Рисунок 1 Приклад інтерфейсу

OpenHAB: OpenHAB є ще однією відкритою платформою для автоматизації SmartHome, яка має вбудований інтерфейс візуалізації даних. Вона надає можливість створення гнучких графіків, діаграм та інших елементів, що відображають стан пристроїв та параметри клімату.

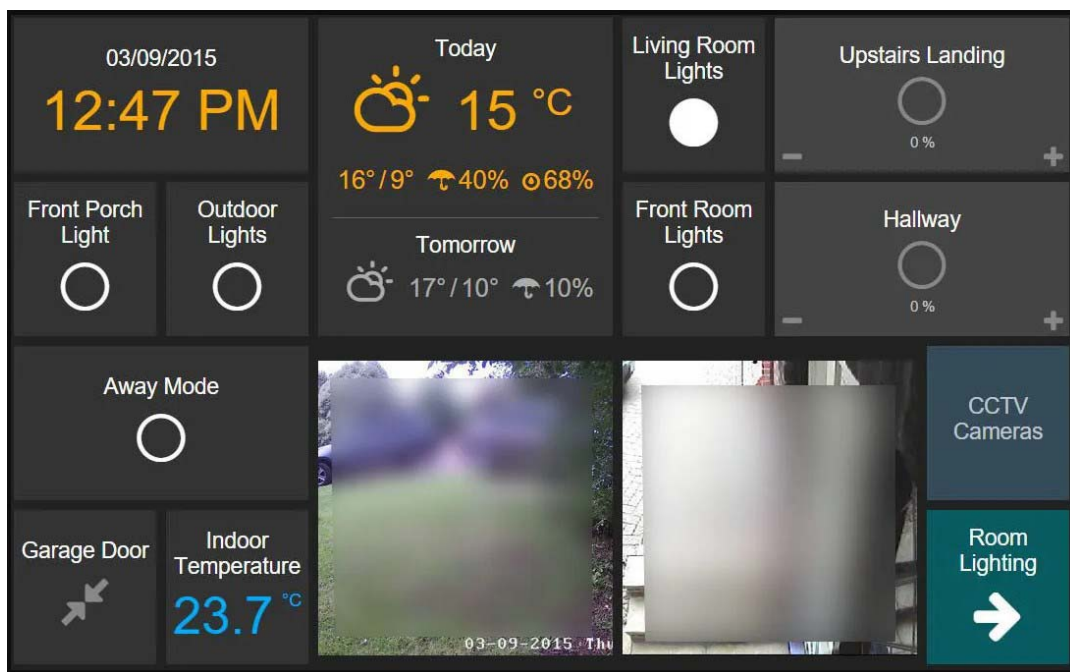


Рисунок 2 Приклад інтерфейсу

Domoticz: Domoticz - це інша популярна система автоматизації SmartHome з вбудованим інтерфейсом візуалізації даних. Вона надає графіки та діаграми для відображення параметрів, таких як температура, вологість, енергоспоживання, а також можливості налаштування дашбордів та списків пристроїв.

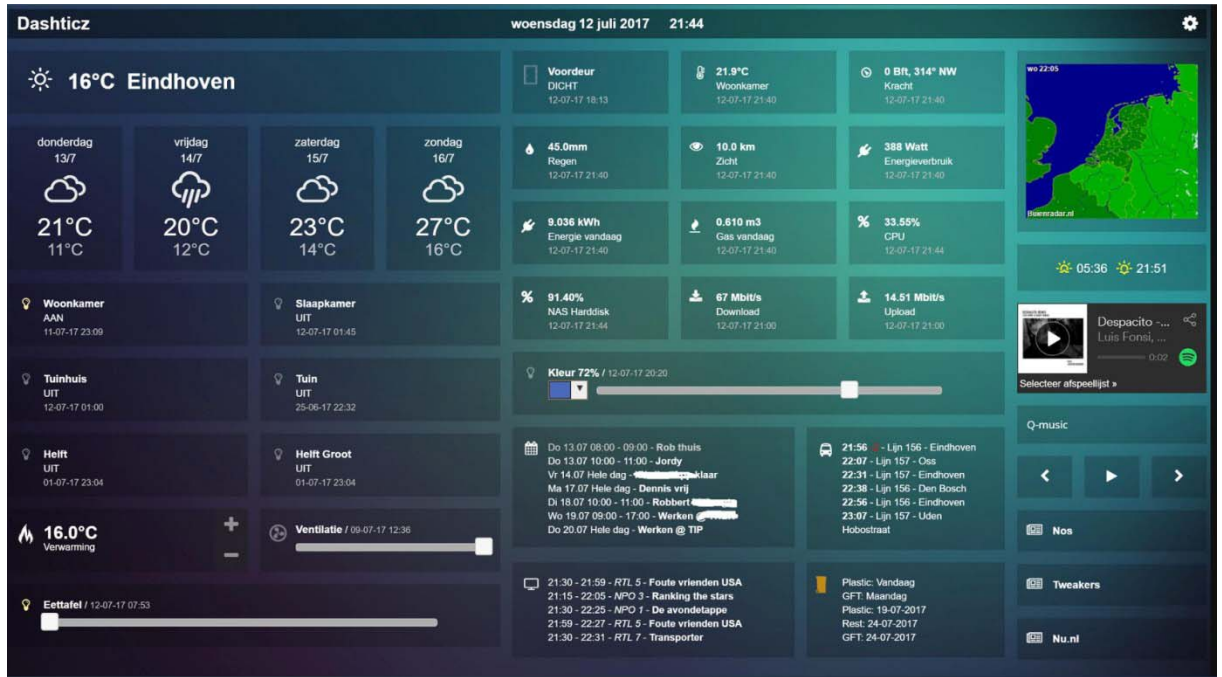


Рисунок 3 Приклад інтерфейсу

ThingsBoard: ThingsBoard - це відкрита платформа IoT, яка дозволяє візуалізувати дані з пристроїв SmartHome. Вона надає можливість створювати інтерактивні графіки, карти, віджети та інші елементи, які відображають стан пристроїв та їх дані.



Рисунок 4 Приклад інтерфейсу

2.4 Проектування і розробка бази даних

Для зберігання даних IoT SmartHome можна використовувати різні типи баз даних, залежно від вимог проекту та відповідного використання. Ось кілька прикладів баз даних, які часто використовуються в IoT SmartHome:

Реляційні бази даних (наприклад, MySQL, PostgreSQL): Реляційні бази даних є популярним вибором для зберігання даних IoT SmartHome. Вони відомі своєю структурованістю, можливістю забезпечити цілісність даних, зручність у роботі зі складними запитами та розширення. Однак, вони можуть бути менш ефективними для обробки великого обсягу швидкозмінюваних даних, що характерні для IoT.

NoSQL бази даних (наприклад, MongoDB, Cassandra): NoSQL бази даних набувають популярності в IoT-системах, оскільки вони можуть забезпечити гнучкість, горизонтальне масштабування та швидку обробку великого обсягу даних. Вони дозволяють зберігати неструктуровані або напівструктуровані дані, такі як JSON-документи, що часто використовуються в IoT. Однак, вони можуть бути складнішими у використанні і вимагати більше зусиль для забезпечення цілісності даних.

Часові ряди баз даних (наприклад, InfluxDB, TimescaleDB): Часові ряди баз даних спеціалізуються на зберіганні та обробці часових серій даних, що є типовим для даних IoT SmartHome. Вони забезпечують оптимізовані операції запису, запиту та аналізу даних за часовою міткою. Це дозволяє швидше зберігання та доступ до історичних даних. Однак, часові ряди баз даних можуть бути менш гнучкими для зберігання інших типів даних, крім часових серій.

2.5 Інтеграція з зовнішніми сервісами

Система SmartHome для моніторингу системи опалення та кліматичних показників може бути інтегрована з різними сторонніми сервісами для

розширення її функціональності та забезпечення більш широкого спектру можливостей. Ось декілька прикладів сторонніх сервісів, з якими можлива інтеграція:

Сервіси погоди: Інтеграція з сервісами погоди, наприклад, AccuWeather або OpenWeatherMap, дозволяє отримувати актуальні дані про погоду та передбачення. Це може допомогти системі SmartHome враховувати зміни погодних умов при керуванні опаленням та кліматичними параметрами.



Рисунок 5 Графічний інтерфейс інтеграційного модуля AccuWeather

Голосові помічники: Інтеграція з голосовими помічниками, такими як Amazon Alexa, Google Assistant або Apple Siri, дозволяє керувати системою SmartHome голосом. Користувач може дати команди для встановлення температури, включення/вимикання опалення та інших дій, спілкуючись з голосовим помічником.

Енергетичні сервіси: Інтеграція з енергетичними сервісами, такими як UtilityAPI або Tiko, дозволяє отримувати дані про споживання енергії та ціни. Це може допомогти оптимізувати режими опалення та керування енергоефективністю.

Системи безпеки: Інтеграція з системами безпеки, такими як Ring або Nest Secure, дозволяє об'єднати функціональність системи SmartHome з моніторингом безпеки. Наприклад, в разі виявлення пожежі або витоку газу

система безпеки може автоматично вимкнути опалення та надіслати сповіщення користувачеві.

Мобільні додатки та платформи IoT: Інтеграція з мобільними додатками або платформами IoT, такими як Samsung SmartThings або Apple HomeKit, дозволяє розширити можливості керування та моніторингу системи SmartHome. Користувач може використовувати мобільний додаток для керування опаленням та кліматичними параметрами з будь-якого місця.

2.6 Висновки до розділу

Вданому розділі було детально розглянуто та вибрано компоненти системи що дозволяють оптимально вирішити поставлені завдання. Аналіз та порівняння компонентів дозволив зупинити свій вибір на системі HomeAssist в якості ядра проекту. Розглянуто можливості по інтеграції зі сторонніми ресурсами та вибрано бізнес модель проекту.

Окрім цього, розглянуто можливості використання різноманітних графічних інтерфейсів для досягнення більш оптимальної користувацької взаємодії. В результаті дослідження було вибрано централізований підхід з використанням функціональних блоків і графічного дизайнера. Цей підхід надає користувачеві можливість самостійно створювати сцени та звіти по отриманих даних.

Додатково, був розроблений робочий прототип в якості тестового стенду де функціонує зазначене ядро.

РОЗДІЛ 3. МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1 Алгоритми нечіткої логіки

Алгоритми нечіткої логіки - це математичні методи та інструменти, які використовуються для обробки та моделювання нечіткої інформації або нечітких змінних. Нечітка логіка виникла як спроба математично описати недосконалість та неоднозначність в людському мисленні та мові. Вона відрізняється від традиційної булевої логіки, де значення змінних може бути "правда" або "хибно", і дозволяє виразити ступінь істинності або належності до певного поняття від 0 до 1.

Основні поняття та ідеї нечіткої логіки включають:

Лінгвістичні змінні: Замість точних числових значень, нечітка логіка оперує лінгвістичними змінними, які виражаються мовними термінами, такими як "низький", "середній", "високий".

Терм-множини: Для кожної лінгвістичної змінної створюються терм-множини, які визначають, яким чином значення може відповідати різним лінгвістичним термінам.

Функції належності: Кожній терм-множині присвоюється функція належності, яка визначає ступінь належності конкретного значення цій множині.

Правила нечітких висновків: Формулюються правила, що використовують лінгвістичні змінні, їх терм-множини та функції належності для отримання висновків на основі вхідних даних.

Дефазифікація: Процес перетворення нечітких висновків у конкретні числові значення.

Алгоритми нечіткої логіки застосовуються у різних галузях, включаючи системи керування, штучний інтелект, обробку сигналів, медицину, фінанси та багато інших. Вони дозволяють ефективно моделювати та управляти складними системами, де інформація може бути нечіткою чи неоднозначною.

3.2 Застосування алгоритмів нечіткої логіки в системі SmartHome

Нечітка логіка має широкий спектр застосувань в системах "розумного будинку" (SmartHome), оскільки вона дозволяє ефективно моделювати та керувати різноманітними аспектами життя в будинку, де велика кількість параметрів може бути нечіткою та залежить від індивідуальних вподобань користувачів. Ось декілька способів застосування нечіткої логіки в системі Smart Home:

Керування температурою: Нечітка логіка може використовуватися для оптимального управління системою опалення та кондиціонування повітря в будинку, враховуючи внутрішній і зовнішній температури, вологість, зони активності та вподобання користувачів.

Освітлення: Нечітка логіка дозволяє налаштувати освітлення в будинку відповідно до потреб користувачів, враховуючи час доби, погодні умови, рівень освітленості та активність в певних приміщеннях.

Безпека: В системах Smart Home нечітка логіка може використовуватися для визначення рівня загрози та прийняття рішень щодо безпеки, такі як виявлення вторгнень, контроль доступу або пожежна безпека.

Управління енергоспоживанням: Нечітка логіка дозволяє оптимізувати енергоспоживання в будинку, враховуючи різні фактори, такі як добова година, звички мешканців та стан приладів.

Голосове керування: Системи голосового керування можуть використовувати нечітку логіку для розпізнавання інтенцій користувачів та виконання відповідних дій.

Комфорт та настрої: Нечітка логіка може допомагати враховувати особисті вподобання користувачів щодо атмосфери в будинку, наприклад, настрої (романтичний, робочий, релаксуючий) та музичний супровід.

Застосування нечіткої логіки в системі Smart Home дозволяє створити більш гнучкі та інтелектуальні системи, які можуть адаптуватися до змінних

потреб користувачів та забезпечувати комфорт та ефективність управління різними аспектами життя в будинку.

3.2 Практичне застосування

Реалізація системи оцінювання в контексті використання нечіткої логіки включає такі основні етапи:

1. Визначення лінгвістичних змінних, базових терм-множин і функцій належності:

- Визначення параметрів, які потрібно оцінювати в системі (наприклад, температура, вологість, освітленість).
- Створення лінгвістичних змінних для цих параметрів та визначення базових терм-множин, які описують діапазони значень (наприклад, "холодно", "тепло", "гаряче" для температури).

2. Визначення правил нечітких висновків:

- Сформулювання правил, які вказують, які дії потрібно виконати в залежності від вхідних значень лінгвістичних змінних. Правила мають вигляд "Якщо... то...".

3. Процес дефазифікації:

- Після використання нечітких правил для визначення вихідних результатів, проводиться процес дефазифікації, який перетворює нечіткі висновки в конкретні числові значення або дії.

4. Реалізація алгоритмів та програмного забезпечення:

- Розробка програмного забезпечення або алгоритмів, які реалізують всі вищезазначені етапи системи оцінювання на практиці.

5. Тестування та налаштування:

- Після реалізації системи проводиться тестування для перевірки її працездатності та налаштування параметрів для досягнення оптимальних результатів.

Ці етапи допомагають створити систему оцінювання, яка може приймати рішення на основі нечітких вхідних даних та забезпечувати більш гнучкий та інтелектуальний підхід до управління різними аспектами системи.

Ці етапи зображено на Рисунку 6.



Рисунок 6. Модель нечіткої логіки в системі оцінювання

Застосування вказаної системи оцінювання подано для визначення комфортних умов (температури) перебування у приміщенні (кімнаті) в системі HomeAssistant.

При розгляді змінної "температура повітря атмосфери", для неї формується набір термінів, який включає чотири категорії: "холодно", "нейтрально", "тепло" та "гаряче". Після визначення цих лінгвістичних змінних і створення відповідних терм-множин з функціями належності, наступним етапом є розробка продукційних правил для моделі. Продукційні правила мають такий формат: "Якщо ... , то ...". У таблиці наведено приклади таких правил. Деякі значення можуть бути відсутні в таблиці, що свідчить про те, що в відповідних правилах вихідні результати не залежать від цієї конкретної лінгвістичної змінної.

Приклад інтепретації прав з таблиці:

- Якщо T_{in} є «холодно», і T_{out} є «мороз», і S_{inf} є «середній», тоді T_{surf} є «низька». 147
- Якщо T_{in} є «холодно» і T_{out} є «холодно», і V_w є «помірний», і S_{inf} є «середній», і R є «високий» тоді T_{surf} є «низька».

Правила								
№ з/п		T_{in}	T_{out}	V_w	S_{ind}	T		T_{surf}
1	Якщо	холодно	мороз		середній		Тоді	низька
2	Якщо	холодно	холодно	помірний	середній	високий	Тоді	низька
3	Якщо	холодно	тепло	сильний		низький	Тоді	низька
4	Якщо	комфортно	тепло	свіжий		низький	Тоді	середня
5	Якщо	комфортно	мороз	свіжий	високий	низький	Тоді	низька
6	Якщо	комфортно	холодно	помірний	високий	низький	Тоді	середня
7	Якщо	комфортно	спека	сильний	середній	низький	Тоді	середня
8	Якщо	комфортно	тепло	помірний	високий	високий	Тоді	висока
...								
14	Якщо	тепло	тепло	помірний			тоді	дуже висока
...								

Рисунок 7. Правила побудови нечіткої логіки

Отриманий результат після оцінки нечітких правил зазвичай являти собою нечітке значення. Для перетворення цього нечіткого виходу в чітке значення використовується функція належності, а цей процес називається дефазифікацією. У цій роботі використовується метод середнього центру, що також відомий як центроїдний метод, для цього. Отримані обрізані функції (за алгоритмом Мамдані) об'єднуються за допомогою максимізації.

3.3 Алгоритм Мамдані та математичне трактування

Визначення: У предметній області формуються нечіткі предикатні правила у такому форматі:

П1: Якщо x належить до A_1 , то z належить до B_1 ,

П2: Якщо x належить до A_2 , то z належить до B_2 ,

...

П n : Якщо x належить до A_n , то z належить до B_n ,

Дех - вхідна змінна (ім'я для відомих значень даних), z - змінна виводу (ім'я для значення даних, яке буде обчислене). A_i та B_i - нечіткі множини, які визначаються на X та Z відповідно за допомогою функції належності (z).

Математичне трактування: У представленій ситуації даних вивід у формі алгоритму Мамдані математично можна представити наступним чином.

1. *Введення нечіткості (fuzzification)*: для заданого(чіткого) значення аргументу $x = x_0$ знаходяться ступені істинності для заданого кожного правила $a_i = (x_0)$.

2. *Нечіткий вивід*: знаходяться рівні відсічення для кожного з правил (з використання правила мінімуму):

$$\alpha_1 = A_1(x_0) \wedge B_1(y_0)$$

$$\alpha_2 = A_2(x_0) \wedge B_2(y_0)$$

де через \wedge позначено операцію логічного мінімуму. Потім знаходяться усічені функції належності:

$$C'_1(z) = (\alpha_1 \wedge C_1(z))$$

$$C'_2(z) = (\alpha_2 \wedge C_2(z))$$

3. *Композиція*: з використанням операції *max* (позначеної як \vee) виконується об'єднання знайдених усічених функцій, що приводить до отримання підсумкової нечіткої підмножини для змінної виходу з функцією належності.

$$M(z) = C(z) = C'_1(z) \vee C'_2(z) = (\alpha_1 \wedge C_1(z)) \vee (\alpha_2 \wedge C_2(z))$$

3.4 Висновки до розділу

З використанням нечіткої логіки для керування температурою в системі "розумний будинок" отримано наступні покращення:

- Підвищення комфорт мешканців.
- Зменшення енергоспоживання та витрат на опалення.
- Адаптування до змінних умов навколишнього середовища та активності членів сім'ї.
- Уникнення небезпеки перегріву.
- Більш зручний спосіб керування для користувачів.

Приклад конфігураційного файлу для модуля нечіткої логіки HomeAssistant надано в Додатку 3

РОЗДІЛ 4. ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

4.1 Побудова архітектури системи

Архітектура програмної системи - це опис високорівневої структури та організації програмного продукту. Вона визначає, як компоненти системи взаємодіють між собою, як дані та функції організовані, та як система відповідає на вимоги та виконує свої функції.

З огляду на те, що є необхідність використовувати сторонні сервіси для збереження, обробки та візуалізації даних, а також необхідність налаштування автоматичного розширення системи – оптимальним підходом, до побудови архітектури системи, буде розподілена система з використанням сторонніх та хмарних сервісів.

Розподілена архітектура має декілька переваг, особливо в контексті системи SmartHome. Ось деякі з них:

1. Масштабованість: Розподілена архітектура дозволяє гнучко масштабувати систему. Компоненти системи можуть бути розгорнуті на різних фізичних серверах або вузлах, що дозволяє легко збільшувати обсяг обробки даних або навантаження системи.

2. Висока доступність: Розподілена архітектура може забезпечити високу доступність системи. Якщо один компонент несправний або недоступний, інші компоненти можуть продовжувати працювати, забезпечуючи неперервну роботу системи SmartHome.

3. Відмовостійкість: Розподілена архітектура може бути більш стійкою до відмов. Якщо один компонент виходить з ладу, інші компоненти можуть продовжувати працювати, уникнувши збоїв у системі.

4. Гнучкість і зручність розробки: Розподілена архітектура дозволяє розробникам працювати над окремими компонентами системи незалежно один від одного. Це дозволяє ефективніше розподілити завдання між

командами розробників і сприяє швидкому розвитку та впровадженню нових функцій.

5. Зменшення навантаження на мережу: Розподілена архітектура дозволяє розподіляти навантаження на мережу між різними компонентами системи. Це може знизити пропускну здатність та навантаження на окремі вузли, покращуючи продуктивність системи.

6. Легша інтеграція зі сторонніми сервісами: Розподілена архітектура дозволяє легше інтегрувати систему SmartHome зі сторонніми сервісами або іншими системами. Компоненти можуть взаємодіяти за допомогою стандартизованих інтерфейсів або протоколів, спрощуючи обмін даними та функціональність між різними системами.

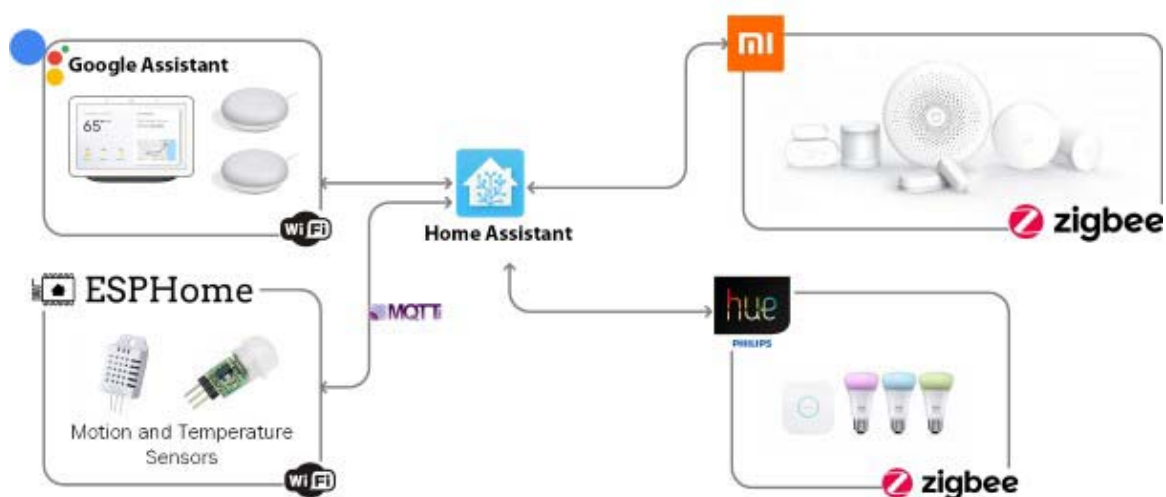


Рисунок 8 Архітектурна схема системи SmartHome

З архітектурної схеми видно, що центральним вузлом системи, є HomeAssistant, що вибраний через швидке розгортання та безоплатну систему поширення, окрім того система має дуже велику підтримку в колах розробників та користувачі.

В якості кінцевих пристроїв, використовуються скенсори та керучі пристрої на базі протокола ZigBee системи SmartLife/TUYA.

Система також включає кінцеві пристрої на базі ESP32 в якості моніторів температури що були розроблені та запрограмовані власноруч.

4.2 Огляд апаратних засобів

Система HomeAssistant встановлено на базі **Raspberry Pi 3 Model B+** що підтримує функції контейнеризації, де водночас розгорнуто базу даних.

Raspberry Pi 3 Модель B+ (RPI3-MODBP) це повноцінний одноплатний комп'ютер розміром з банківську карту. Raspberry Pi 3 Модель B+ це поліпшена версія Raspberry Pi 3 Модель B. Модель базується на SoC (системі-на-чипі) BCM2837B0, яка включає в себе 4-ядерний ARMv8 64-бітний процесор з робочою частотою 1,4 ГГц і потужний відеопроцесор VideoCore IV.

Raspberry Pi 3 Модель B був першим Raspberry Pi, який мав функціонал бездротового і Bluetooth зв'язку. **Raspberry Pi Модель B+** має такі ж можливості, але на відміну від попередника, він має Bluetooth версії 4.2.



Рисунок 9 RaspberryPi

Raspberry Pi був розроблений в якості доступної платформи для експериментів і освіти в області комп'ютерного програмування. Raspberry Pi може використовуватися для багатьох речей, аналогічних тим, що робить

звичайний настільний ПК, в тому числі обробки текстів, електронних таблиць, відео високої чіткості, ігор і програм. USB-пристрої, такі як клавіатури і миші можуть бути підключені через наявні на платі чотири порти USB.

Відмінною особливістю мікрокомп'ютерів Raspberry Pi є наявність роз'єму 40-pin GPIO для управління різними пристроями і його малий розмір, завдяки чому Raspberry Pi може також працювати як програмований контролер в найрізноманітніших додатках робототехніки та електроніки. Як варіант, він може бути об'єднаний з Pololu A-Star 32U4 Robot Controller LV з Raspberry Pi Bridge, для створення контролера для цього маленького робота.

Завдяки стандартному роз'єму 40-pin GPIO до мікрокомп'ютеру можна приєднувати всілякі плати розширення для додавання необхідного функціоналу.

Raspberry Pi може працювати з широким спектром дистрибутивів ARM GNU / Linux, включаючи Ubuntu Snappy Core, Debian, Fedora і Arch Linux, а також Microsoft Windows 10 IoT Core.

Характеристики:

Чіп: Broadcom BCM2837B0 (CPU, GPU, DSP, and RAM)

Процесор (CPU): 64-бітний 4-ядерний ARMv8 Cortex-A53 з тактовою частотою 1,4 ГГц

Відеопроцесор (GPU): VideoCore IV 3D

Пам'ять (RAM): 1 Гб LPDDR2 (900 МГц)

Роз'єми:

аудіо / відео цифровий: HDMI

аудіо / відео аналоговий: 3,5 мм jack (4 pin)

USB порт: USB 2.0 - 4 шт

Накопичувач: microSD

Мережа: 10/100 Ethernet

Порт дисплея: Display Serial Interface (DSI)

Порт камери: MIPI Camera Serial Interface (CSI-2)

Роз'єм ведення-виведення: GPIO 40 pin

Бездротові інтерфейси:

WiFi: двосмуговий 2,4 ГГц and 5 ГГц IEEE 802.11.b / g / n / ac 2,4 ГГц

Bluetooth 4.2 Classic і Low Energy (BLE)

В якості ядра локальної мережі використується **RouterBOARD 951G-**

2HnD:

Таблиця 2 Технічні характеристики RouterBOARD 951G-2HnD

Система	
Процесор:	Atheros AR9344 600 МГц
Оперативна пам'ять:	128 MB DDR SDRAM
Постійна пам'ять:	64 MB
Роз'єми	5×10/100/1000 Мбит/с LAN 1×USB 2.0
Операційна система:	MikroTik RouterOS Level4
Точка доступу	
Стандарти:	IEEE 802.11 b/g/n
Канальна швидкість:	300 Мбит/с
Передавачі:	802.11g: 30dBm @ 6Mbps ⇒ 25dBm @ 54Mbps 802.11n: 30dBm @ MCS0 ⇒ 23dBm @ MCS7
Приймачі:	802.11g: -96dBm @ 6Mbps ⇒ -80dBm @ 54Mbps 802.11n: -96dBm @ MCS0 ⇒ -78dBm @ MCS7
Антенна:	2×всенаправлені антени, MIMO2×2
Підсилення антени:	2,5 дБи
Частоти:	2,4 ГГц
Додаткові функції	
Керування:	winbox, telnet, WEB інтерфейс
Інше	
Живлення:	PoE: 8..30 V DC on Ethernet port1 jack: 8..30 V DC
Споживання:	до 13 Вт



Рисунок 10 Роутер

В якості розподілених WiFi антен використано **MikroTik cAP ac (RBcAPGi-5acD2nD)**

Таблиця 3 Технічні характеристики MikroTik cAP ac (RBcAPGi-5acD2nD)

Система	
Процесор:	IPQ-4018 716 МГц 4 ядра
Оперативна пам'ять	128 МВ
Постійна пам'ять	16 МВ
Роз'єми	2 × 10/100/1000 Base-TX (Cat. 5, RJ-45) Ethernet
Операційна система	MikroTik RouterOS Level4 license
Радіомодуль	
Стандарти:	802.11 a/b/g/n/ac
Режими роботи:	Access Point Station Point-to-point Repeater
Швидкість Wi-Fi:	300 Мбит/с на 2.4ГГц / 866.7 Мбит/с на 5ГГц
Частота:	2,4 ГГц / 5 ГГц
Антенна	
Вид:	Вбудована всепрямована
Підсилення:	2 × 2 dBi антени на 2,4ГГц 2 × 2.5 dBi антени на 5ГГц
Другое	
Споживання	PoE-in Ethernet port 1: 17–57 В (802.3af/at or Passive PoE) PoE-out Ethernet port 2: 17–57 В (Passive PoE)



Рисунок 11 WiFi антена

В якості пристрою поєднання WiFi/Ethernet + ZigBee мережі використано шлюз **Zigbee 3.0 шлюз Moes Gateway Wireless** що дозволяє інтегрувати обидві системи з Raspberry Pi 3+



Рисунок 12 Шлюз

Кінцеві пристрої працюють на базі протоколу Zigbee та являють собою сенсори температури Zigbee датчик температури і вологості Aqara

(WSDCGQ11LM) та термостатичні головки радіаторів Zigbee, Moes BRT-100 і сенсори на базі ESP32



Рисунок 13 Термосенсор



Рисунок 14 Термостатична головка

4.3 Налаштування баз даних

Для запуску Home Assistant, MySQL та Grafana в контейнерах Docker на Raspberry Pi 3+, було встановлено і виконано наступне:

1. Docker: Для встановлення Docker на Raspberry Pi 3:

```
...  
curl -sSL https://get.docker.com | sh  
sudo usermod -aG docker pi  
...
```

- Перезавантажте Raspberry Pi, щоб зміни вступили в силу:

```
...  
sudo reboot  
...
```

2. Home Assistant: Для запуску Home Assistant в контейнері Docker, використав офіційний образ `homeassistant/raspberrypi3-homeassistant`. Для цього створили `docker-compose.yml` файл з наступним вмістом:

```
``yaml  
version: '3'  
services:  
  homeassistant:  
    container_name: homeassistant  
    image: homeassistant/raspberrypi3-homeassistant:stable  
    volumes:  
      - ./config:/config  
    restart: always  
    network_mode: host  
...``
```

Цей `docker-compose.yml` файл налаштовує контейнер Home Assistant на використання офіційного образу `homeassistant/raspberrypi3-homeassistant:stable`. Ваша конфігурація Home Assistant буде зберігатися в локальній папці `./config` (створіть її перед запуском `docker-compose up`). Контейнер використовуватиме мережу хоста Raspberry Pi.

3. MySQL: Для запуску MySQL в контейнері Docker, ви можете використав ти офіційний образ `mysql`. Ось приклад `docker-compose.yml` файлу для MySQL:

```
``yaml
version: '3'
services:
  mysql:
    container_name: mysql
    image: mysql
    restart: always
    environment:
      MYSQL_ROOT_PASSWORD: your_root_password
      MYSQL_DATABASE: your_database_name
      MYSQL_USER: your_username
      MYSQL_PASSWORD: your_password
    volumes:
      - ./data:/var/lib/mysql
...

```

4. Grafana: Для запуску Grafana в контейнері Docker, ви можете використав офіційний образ `grafana/grafana`. Ось приклад `docker-compose.yml` файлу для Grafana:

```
``yaml
version: '3'
services:
  grafana:
    container_name: grafana
    image: grafana/grafana
    restart: always
    ports:
      - '3000:3000'
    volumes:
      - ./data:/var/lib/grafana
...

```

Цей `docker-compose.yml` файл налаштовує контейнер Grafana на використання офіційного образу `grafana/grafana`. Контейнер мапує порт 3000 в контейнері на порт 3000 на вашій машині, щоб ви могли отримати доступ до веб-інтерфейсу Grafana. Дані Grafana зберігаються в папці `./data`.

Після створення `docker-compose.yml` файлу, запуск контейнерів здійснюється за допомогою команди `docker-compose up`. Home Assistant, MySQL та Grafana будуть розгорнутися в окремих контейнерах Docker з вказаними налаштуваннями.

4.4 Опис користувацького інтерфейсу

Основні вимоги до користувацького інтерфейсу веб-додатку зазвичай включають наступні аспекти:

Інтуїтивно зрозумілий та простий у використанні: Інтерфейс повинен бути легким у навігації та розумінні, зрозумілим для користувача без необхідності великого навчання чи інструкцій. Елементи керування повинні бути розміщені логічно та мати зрозумілі підказки.

Адаптивний дизайн: Інтерфейс повинен бути адаптивним до різних пристроїв і розмірів екранів, щоб користувачі могли зручно взаємодіяти з додатком на різних пристроях, таких як комп'ютери, планшети або смартфони.

Приємний зовнішній вигляд та естетика: Інтерфейс повинен мати привабливий та професійний дизайн, який відповідає бренду і сприяє позитивному враженню користувачів. Використання візуальних елементів, таких як кольори, шрифти, ілюстрації тощо, допоможе створити привабливий вигляд.

Ефективність та продуктивність: Інтерфейс повинен бути швидким та ефективним, з мінімальним часом відгуку на дії користувача. Це можна досягти шляхом оптимізації завантаження сторінок, кешування даних, асинхронної передачі даних тощо.

Цим вимогам повністю відповідає вбудований інтерфейс HomeAssistant що дозволяє отримати дуже гнучкий та зрозумілий набір маркерів, графіків та представлень для зчитування даних:

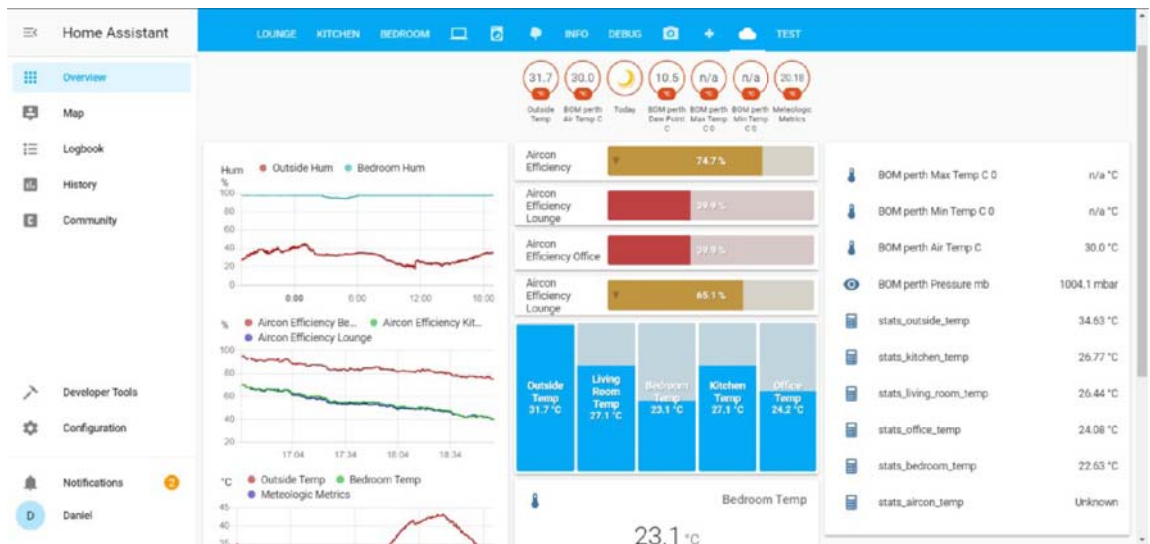


Рисунок 15 Інтерфейс сенсорів температури

4.5 Розробка власного сенсора температури

Для розробки власного сенсора температури взято типову схему підключення ESP32 та DS18B20. 1й є програмованим мікроконтроллером, Ю інший є високоточним датчиком температури що працює на шині 1-Wire ,що дозволяє водночас підключити до 127 сенсорів по одному дроту та отримувати цифрові дані температури.

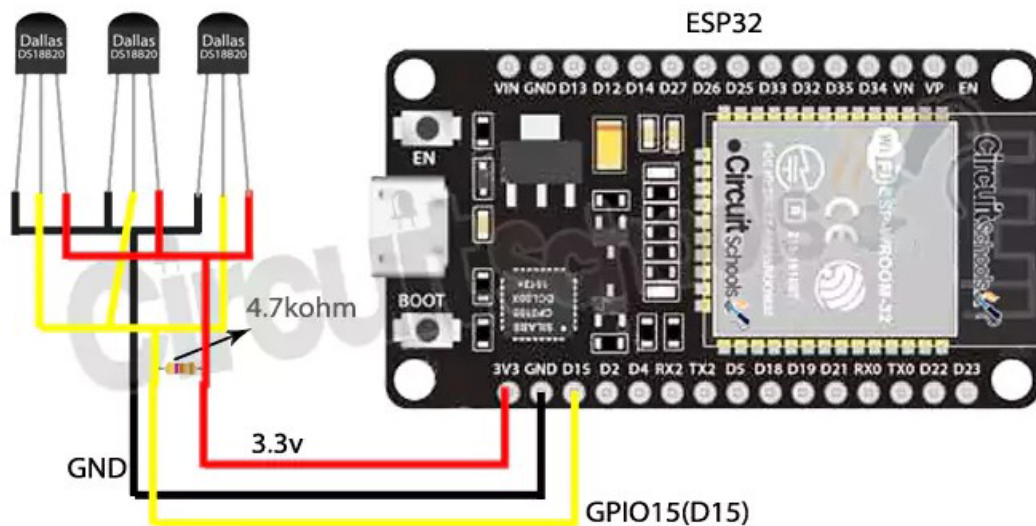


Рисунок 16 Принципова схема підключення ESP32-WROOM та DS18B20

4.6 Висновки до розділу

В цьому розділі проведено детальний аналіз матеріало-технічної бази для проекту. Надано та реалізовано, спираючись на конфігурації схеми міжкомпонентної взаємодії використовуючи розглянуті протоколи та файли конфігурації.

Застосовано та налаштовано бібліотеку Graphana для візуалізації даних які надходять від сенсорів даних, по температурі та вологості. Це дозволило створювати графіки та аналізувати залежності від кількох джерел водночас.

В цьому розділі визначено ключові компоненти та їхні конфігурації які застосовані на проекті. Також надано вичерпну картину можливостям розширення та інтеграції додаткових сенсорів та актуаторів для вирішення ширшого кола задач (керування та досягнення економічної вигоди) а також розроблених власноруч.

РОЗДІЛ 5. РОЗРОБЛЕННЯ СТАРТАП-ПРОЄКТУ

5.1 Опис ідеї проєкту, визначення загальних напрямів використання потенційного товару чи послуги, а також їх відмінність від конкурентів

Зміст ідеї розробки інтелектуальної системи керування будинком SmartBuilding за допомогою пристроїв IoT полягає в створенні розумного та ефективного середовища для життя та роботи в будинках і офісах. Основна мета такої системи - полегшити керування будинком, знизити споживання енергії, покращити безпеку та комфорт користувачів.

Можливі напрямки застосування системи керування будинком SmartBuilding за допомогою пристроїв IoT досить різноманітні і можуть включати не лише окремі напрямки, а й цілі галузі та сфери економіки :

Житлові будинки: У житлових будинках система може оптимізувати споживання енергії, полегшити керування опаленням, освітленням та кондиціонуванням повітря, а також забезпечити комфорт та безпеку мешканців.

Офісні приміщення: В офісних будівлях система може допомагати управляти освітленням та кліматом для підвищення продуктивності праці та ефективності використання енергії.

Готелі та гостьові номери: У готелях і гостьових номерах система може забезпечувати зручність для гостей, дозволяючи керувати усіма аспектами їхнього перебування, включаючи освітлення, температуру та розваги.

Спортивні заклади: У спортивних закладах система може допомагати забезпечити комфортні умови для спортсменів та глядачів, а також оптимізувати використання ресурсів, наприклад, ефективно керувати освітленням на стадіонах або у спортивних залах.

Освітні установи: В школах та університетах система може допомагати знижувати споживання енергії та забезпечувати зручні кліматичні умови в навчальних приміщеннях.

Медичні заклади: У лікарнях та медичних центрах система може бути корисною для забезпечення стабільних умов зберігання медичних препаратів та обладнання.

Промислові об'єкти: У промислових комплексах система може допомагати управляти та моніторити устаткування та споживання енергії для підвищення продуктивності та зниження витрат.

Будинки для літніх людей та інвалідів: У таких будинках система може забезпечувати контроль над навколишнім середовищем, включаючи моніторинг датчиків руху та валідність ванн та душів.

Ресторани і готелі: В галузі гостинності система може автоматизувати процеси освітлення та клімату в об'єктах громадського харчування, підвищуючи комфорт клієнтів і знижуючи енергоспоживання.

Сільське господарство: У сільському господарстві система може служити для моніторингу умов вирощування рослин та тварин і оптимізації процесів.

Архітектурні об'єкти: Система може бути використана в архітектурних об'єктах для демонстрації сучасних технологій та створення інтерактивних виставкових приміщень.

Місця відпочинку та парки: В парках та рекреаційних зонах система може допомагати в управлінні освітленням, фонтанами, музикою та іншими атракціями.

Не залежно від способу та сфери застосування **основні вигоди** від подібної системи можуть бути згруповані в наступні категорії:

Економія енергії: Система дозволяє оптимізувати використання енергії в будинку. Ви можете автоматично вимикати світло, кондиціонери та інші пристрої, коли вони не потрібні, що допомагає знижувати рахунки за комунальні послуги.

Комфорт та зручність: Ви можете керувати освітленням, температурою та іншими аспектами вашого оточення з використанням

смартфона або голосових команд. Це дозволяє створити комфортні умови в будинку за вашими перевагами.

Безпека: Система відеоспостереження та сигналізації дозволяє вам моніторити свій будинок здалеку та отримувати сповіщення про події в реальному часі. Це покращує безпеку майна та мешканців.

Зменшення ризику аварій: Система може виявляти потенційні проблеми, такі як витік води або несправність системи опалення, і відразу ж повідомляти про них. Це допомагає зменшити ризик аварій та ремонтних робіт.

Розумна автоматизація: Ви можете створювати різні сценарії автоматизації для спрощення щоденних рутинних завдань. Наприклад, автоматично вмикати та вимикати освітлення певних приміщень або налаштовувати режими опалення та охолодження відповідно до ваших потреб та графіка.

Доступність: Система може бути корисною для людей з обмеженими можливостями, надаючи можливість керувати будинком та взаємодіяти з ним за допомогою голосових команд або мобільного додатка.

Зручний моніторинг: Ви можете в реальному часі моніторити стан вашого будинку, навіть якщо ви далеко від нього. Це дозволяє вам залишати будинок з впевненістю та контролювати його, навіть коли ви відпочиваєте або подорожуєте.

Підвищена вартість нерухомості: У випадку продажу будинку система SmartBuilding може зробити його більш привабливим для покупців, оскільки вона додає значну вартість та покращує якість життя.

Оптимізація витрат на обслуговування: Система допомагає вчасно виявляти та усувати проблеми, що дозволяє знизити витрати на ремонт та обслуговування.

Екологічна підтримка: Зниження споживання енергії і ресурсів допомагає зменшити викиди в атмосферу і сприяє збереженню природних ресурсів.

Загалом, система керування будинком SmartBuilding з використанням IoT покращує якість життя користувачів, забезпечує економію енергії та ресурсів і забезпечує безпеку та зручність в будинку або офісі.

Відмінності запропонованого проекту від існуючих аналогів:

- *Низька вартість і невелика складність:* Система може інтегрувати існуючі протоколи з відкритим кодом які не вимагають оплати та використовувати пристрої які мають високу доступність на ринку , а отже і відносно низьку ціну.
- *Відкритість та гнучкість:* Система використовує ядро з відкритим кодом, що можна самостійно модифікувати, залежно від потреб. Крім того всі моду використовують файли конфігурацій, які можна редагувати та підлаштовувати для кожного окремого випадку.
- *Доступність оновлень і підтримки:* Колектив розробників ядра досить великий та легко йде на контакт в разі наявних труднощів та проблем з оновленнями та помилками виконання, які можуть траплятися в період експлуатації системи

5.2 Технологічна здійсненність ідеї проекту

Технологічна здійсненність ідеї проекту перебуває на досить високому рівні, в силу високої доступності матеріально-технічних компонентів системи:

- Готові сенсори та актуатори доступні на ринку України за помірними цінами, в разі пошуку шляхів здешевлення можливе замовлення компонентів з-за кордону (Китаю). Економія може складати 50% від вартості в Україні
- В разі застосування унікальних сенсорів та їх комбінацій можливе виготовлення пристроїв на основі макетів та замовлення готових рішень на існуючих потужностях промисловості України (Харків, Київ, Одеса)

- Програмна частина системи включає протоколи з відкритим кодом, програмні додатки та модулі та мови програмування що які доступні для скачування та використання на безоплатних правах. В більшості випадків навіть для комерційного застосування.

5.3 Аналіз ринкових можливостей запуску стартап-проекту

Аналіз ринкових можливостей для стартап-проекту системи керування будинком SmartBuilding за допомогою пристроїв IoT важливий для розуміння потенційного попиту та конкуренції в цій галузі. Кроки та фактори важливі для врахування:

Визначення цільової аудиторії: Це можуть бути власники житлових будинків, офісів, готелів, господарства та інші. Основні потреби та вимоги:

- Висока доступність та швидкість розгортання
- Висока економічна доцільність
- Надійність та безпечність використання як фізичних пристроїв так і протоколів передачі даних та зв'язку
- Низький рівень затрат на підтримку та можливість адміністрування системи власними силами або навченим персоналом

Аналіз попиту: Системи схожі на поточні не є новими на ринку, але даних для створення статистики в розрізі нашої країни впродовж останніх років вкрай мало, тим паче ґрунтовних наукових досліджень, які б могли б надати кількісно-якісні характеристики ринку. Попит, здебільшого базується на потребах індивідуальних споживачів, а не на масовості потреб якоїсь галузі.

Конкурентне середовище: Схожі системи існують та вже успішно продаються у готельній сфері та аграрній. Але, оскільки, вони пропонуються як готові рішення, та здебільшого використовують пропрієтарні протоколи комунікації, не дозволяють бути використаними для менших об'єктів та є менш гнучкими для налаштування.

Унікальність пропозиції: Система принаймні унікальній конфігурації є специфічним та вузько застосовним проектом. В той же час вона може надавати кінцевому користувачу можливості для розширення, без необхідності заміни ядра, або основних функціональних блоків.

Ринковий ріст: Тенденції ринку та збільшення пропозиції на складові подібних систем дозволяють припустити, що ринок перебуває на зародковому етапі та лише починає розвиватися в цьому напрямку. Найбільший бум подібних систем очікується в найближчі 1-3 (в ідеалі після завершення війни, в період активного здорожчання енергоресурсів)

Законодавче середовище: На даний момент ринок не регулюється законодавчою базою та не обмежується певними стандартами, окрім тих що запропоновані в галузі розробки програмного забезпечення та галузевими нормативами охорони праці, протипожежними заходами та нормами, а також правилами експлуатації електро установок. Але оскільки споживання пристроїв досить низьке та радіочастотний діапазон попадає в смугу 2.4G що не вимагають додаткового ліцензування та перевірки на відповідність.

Бізнес-модель і монетизація: Систему з відкритим кодом досить складно монетизувати. Єдиним очевидним рішенням на даний момент продаж послуг з розгортання та підтримки системи. В першому випадку – це разова послуга, в другому – потенційно можлива реалізація принципу «підписки» або pre-paid на певний період часу.

Маркетингова стратегія: На даний момент нашвидша стратегія просування з використанням соціальних мереж та ресурсів цільового призначення. Вказана стратегія також дозволяє таргетувати модель для певної категорії користувачів.

Співпраця та партнери: На поточному етапі розвитку проекту співпраця з більшими компаніями/проектками неможлива оскільки може призвести до повного опоглинання та захоплення ідеї.

Фінансування: Виключно з власних фінансових надходжень в необхідності отримання фінансової незалежності та можливості розвитку ідеї в готове рішення.

Аналіз ризиків: Ризики – низьки, оскільки відсутня залежність від зовнішнього фінансування та використання пропріетарних технологій.

Технічна реалізація і розробка: Технічна реалізація проекту для галузевого спеціаліста не є складною, однак, вимагає специфічних знань в суміжних галузях, як то: мережеві технології, мікроелектроніка, програмування та основи роботи з модульним програмним забезпеченням.

Масштабування: На даний момент не планується, до повного налагодження технологічних процесів та створення стійких конфігурації під окремі завдання.

Постійний моніторинг і адаптація: Оскільки дані що збираються є конфіденційними для окремо взятого користувача, моніторинг може здійснюватися лише на основі моделей та сценаріїв використання продукту а не на основі кінцевих даних.

5.4 Висновки до розділу

Після аналізу вказаних пунктів можна зробити кілька ключових висновків, які допоможуть усвідомити можливості та ризики для стартапу системи керування будинком SmartBuilding з використанням IoT:

- Ринок розумних систем керування будинком і IoT є перспективним і має потенціал для зростання, оскільки багато власників будинків та бізнесів шукають способи покращити комфорт, ефективність та безпеку.
- Конкурентне середовище: Ринок вже насичений гравцями, включаючи великі технологічні компанії і стартапи в Америці та Європі, країни східної Європи перебувають в процесі розвитку
- Технічні вимоги і можливості: Розробка та впровадження системи IoT вимагають технічної експертизи та інвестицій у розробку

Загальний висновок може бути таким: ринок для систем керування будинком SmartBuilding за допомогою IoT є обіцяючим, але конкурентним. Для успіху стартапу важливо ретельно розробити стратегію, враховуючи технічні, бізнес- і маркетингові аспекти, а також швидко адаптуватися до змін на ринку та спостерігати за новими можливостями.

ВИСНОВКИ

У процесі створення системи Smart Home дійшов до висновку, що вона відкриває безліч можливостей для автоматизації та управління різними пристроями та функціями в домашньому середовищі. Деякі з них:

1. Зручність і комфорт: Система Smart Home дозволяє керувати освітленням, температурою, безпекою, аудіо-відео обладнанням та іншими пристроями за допомогою мобільного пристрою або голосових команд. Це забезпечує вам зручність та комфорт у повсякденному житті.

2. Енергоефективність: Завдяки автоматизації та розумному керуванню енергопотребною, система Smart Home допомагає знизити споживання енергії та оптимізувати роботу пристроїв. Наприклад, автоматичне вимикання світла, якщо всі члени сім'ї залишають кімнату.

3. Підвищена безпека: Система Smart Home в подальшому інтегрує рішення для безпеки, такі як відеоспостереження, сигналізацію, контроль доступу та датчики пожежі. Зможу отримувати сповіщення про небажану активність або загрози та зможу контролювати ці аспекти будинку в режимі реального часу.

4. Збільшення ефективності: Система Smart Home сприяє автоматизації рутинних завдань і процесів, звільняючи від витрати часу і зусиль. Наприклад, автоматичне вмикання світла при вході в кімнату або відкривання воріт.

5. Розширені можливості: Система Smart Home може бути розширена шляхом додавання нових пристроїв та інтеграції з іншими системами. Ця система може бути розширена системою автоматизації поливу газону, управління звуком, інтелектуальні замками, системою збору даних про споживання енергії.

6. Покращений контроль: Завдяки системі Smart Home можу віддалено контролювати будинок з будь-якого місця через мобільний пристрій. Завдяки

їй я вже можу включати та вимикати пристрої, налаштовувати параметри, отримувати сповіщення та збирати дані для аналізу.

ПЕРЕЛІК ПОСИЛАНЬ

1. Internet.ua [Електронний ресурс] - режим доступу до ресурсу: <https://internetua.com/zvit-kilkist-pidkluacsenih-pristroyiv-iot-zrosla-na-18-do-14-4-milyardiv-po-vsomu-svitu>;
2. codit [Електронний ресурс] - режим доступу до ресурсу: https://www.codit.eu/blog/four-basic-requirements-for-any-iot-platform/?country_sel=be;
3. Internet of things (IoT) Sensors and Actuators [Електронний ресурс] - режим доступу до ресурсу: <https://dgtlinfra.com/internet-of-things-iot-sensors/>;
4. IT Enterprise [Електронний ресурс] - режим доступу до ресурсу: <https://it-enterprise.com/knowledge-base/technology-innovation/promyshlennyj-internet-veschej>;
5. Связь Комплект [Електронний ресурс] - режим доступу до ресурсу: <https://skomplekt.com/kak-najti-i-ubrat-pomekhi-meshayushchie-rabote-wifi/>;
6. Sumatosoft. IoT(Internet of Things) Device Monitoring Best Practices & Top Tools [Електронний ресурс] - режим доступу до ресурсу: <https://sumatosoft.com/blog/iot-device-monitoring-top-tools-and-best-practices>
7. IAIMultiple. A Guide to IoT Monitoring in 2023: Pros, Cons & Importance, Written by Bardia Eshghi [Електронний ресурс] - режим доступу до ресурсу: <https://research.aimultiple.com/iot-monitoring/>
8. Comparitech. The Best IoT Device Monitoring Tools – Updated in 2023 [Електронний ресурс] - режим доступу до ресурсу: <https://www.comparitech.com/net-admin/iot-device-monitoring/>
9. Lightning India. Smart Lighting And Internet Of Things (IoT) [Електронний ресурс] - режим доступу до ресурсу: <https://www.lightningindia.in/smart-lighting-and-internet-of-things-iot/>
10. Philips, official site [Електронний ресурс] - режим доступу до ресурсу: <https://www.philips-hue.com/en-us>
11. DelightedCooking. What is a Smart Refrigerator? Ken Black, Last Modified Date: May 10, 2023 [Електронний ресурс] - режим доступу до ресурсу: <https://www.delightedcooking.com/what-is-a-smart-refrigerator.htm>
12. BizIntellia. An End-to-end solutions. 5 Applications of IoT in Agriculture - Making Agriculture Smarter [Електронний ресурс] - режим доступу до ресурсу: <https://www.biz4intellia.com/blog/5-applications-of-iot-in-agriculture/>
13. IOT solutions congress. IOT TRANSFORMING THE FUTURE OF AGRICULTURE [Електронний ресурс] - режим доступу до ресурсу: <https://www.iotsworldcongress.com/iot-transforming-the-future-of-agriculture/>
14. ScienceDirect. Journals&Books. Review of agricultural IoT technology by Jinyuan Xu, Baoxing Gu, Guangzhao Tian [Електронний ресурс] - режим

- доступу до ресурсу:
<https://www.sciencedirect.com/science/article/pii/S2589721722000010>
15. TechTargetNetwork. industrial internet of things (IIoT) By Brien Posey,Linda Rosencrance,Sharon Shea [Электронный ресурс] - режим доступа до ресурсу: <https://www.techtarget.com/iotagenda/definition/Industrial-Internet-of-Things-IIoT>
 16. IOT WORLD TODAY. solutions congress. The Top 20 Industrial IoT Applications by Brian Buntz [Электронный ресурс] - режим доступа до ресурсу: <https://www.iotworldtoday.com/iiot/the-top-20-industrial-iot-applications>
 17. Builtin. 14 IoT in Retail Examples Improving Your Shopping Experience, Written by Mike Thomas [Электронный ресурс] - режим доступа до ресурсу: <https://builtin.com/internet-things/iot-in-retail-tech-applications>
 18. Digiteum. How to Use Internet of Things (IoT) in the Retail Industry: Examples, Use Cases and Applications [Электронный ресурс] - режим доступа до ресурсу: <https://www.digiteum.com/internet-of-things-retail-industry/>
 19. Itransition. IoT in retail: 10 use cases, benefits, and challenges by Max Pliats, IoT Solution Architect [Электронный ресурс] - режим доступа до ресурсу: <https://www.itransition.com/iot/retail>
 20. IoT Communication Protocols—IoT Data Protocols [Электронный ресурс] – режим доступа до ресурсу: <https://www.allaboutcircuits.com/technical-articles/internet-of-things-communication-protocols-iot-data-protocols/>
 21. Link Labs. The Complete List Of Wireless IoT Network Protocols [Электронный ресурс] – режим доступа до ресурсу: <https://www.link-labs.com/blog/complete-list-iot-network-protocols>
 22. iWave: Embracing the IoT Future with BLE and ZigBee Embedded Technology [Электронный ресурс] – режим доступа до ресурсу: <https://www.iwavesystems.com/news/iot-future-ble-zigbee-embeddedtechnology/>

ДОДАТКИ

Додаток А Код ESP32 з DS18B20

```
#include <WiFi.h>
#include <PubSubClient.h>
#include <OneWire.h>
#include <DallasTemperature.h>

// Налаштування Wi-Fi
const char* ssid = "назва_мережі";
const char* password = "пароль_мережі";

// Налаштування MQTT
const char* mqttServer = "IP_адреса_HomeAssistant";
const int mqttPort = 1883;
const char* mqttUser = "користувач_MQTT";
const char* mqttPassword = "пароль_MQTT";
const char* mqttTopic = "home/temperature";

// Пини датчиків температури DS18B20
const int oneWireBus = 4;

OneWire oneWire(oneWireBus);
DallasTemperature sensors(&oneWire);
WiFiClient espClient;
PubSubClient client(espClient);

void setup() {
  // Підключення до Wi-Fi мережі
  Serial.begin(115200);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(1000);
    Serial.println("Connecting to WiFi...");
  }
  Serial.println("Connected to WiFi");

  // Підключення до MQTT брокера
  client.setServer(mqttServer, mqttPort);
  while (!client.connected()) {
    Serial.println("Connecting to MQTT...");
    if (client.connect("ESP32Client", mqttUser, mqttPassword)) {
      Serial.println("Connected to MQTT");
    } else {
      Serial.print("Failed with state ");
      Serial.print(client.state());
      delay(2000);
    }
  }
}

// Ініціалізація датчиків температури
```

```

sensors.begin();
sensors.setResolution(12); // Налаштування роздільності датчиків (9 - 12 біт)
}

void loop() {
  sensors.requestTemperatures();

  // Зчитування значень температури з кожного датчика
  for (int i = 0; i < 10; i++) {
    float temperatureC = sensors.getTempCByIndex(i);

    // Відправка значення температури до Home Assistant через MQTT
    char mqttTopicWithIndex[30];
    snprintf(mqttTopicWithIndex, sizeof(mqttTopicWithIndex), "%s/%d", mqttTopic, i);
    char tempString[6];
    dtostrf(temperatureC, 4, 2, tempString);
    client.publish(mqttTopicWithIndex, tempString);

    Serial.print("Temperature Sensor ");
    Serial.print(i);
    Serial.print(": ");
    Serial.println(temperatureC);
  }

  delay(5000); // Затримка 5 секунд перед повторним зчитуванням
}

```

Додаток Б Кодчасткової конфігурації HomeAssistant

automation:

- id: enable_zigbee_join
alias: Enable Zigbee joining
hide_entity: true
trigger:
 - platform: state
entity_id: input_boolean.zigbee_permit_join
to: 'on'action:
 - service: mqtt.publish
data:
 - topic: zigbee2mqtt/bridge/config/permit_join
payload: 'true'
 - service: timer.start
data:
 - entity_id: timer.zigbee_permit_join
- id: disable_zigbee_join
alias: Disable Zigbee joining
trigger:
 - entity_id: input_boolean.zigbee_permit_join
platform: state
to: 'off'action:
 - service: mqtt.publish
data:
 - payload: 'false'
topic: zigbee2mqtt/bridge/config/permit_join
 - service: timer.cancel
data:
 - entity_id: timer.zigbee_permit_joinhide_entity: true
- id: disable_zigbee_join_timer
alias: Disable Zigbee joining by timer
hide_entity: true
trigger:
 - platform: event
event_type: timer.finished
event_data:
 - entity_id: timer.zigbee_permit_joinaction:
 - service: mqtt.publish
data:
 - topic: zigbee2mqtt/bridge/config/permit_join
payload: 'false'
 - service: input_boolean.turn_off
data:
 - entity_id: input_boolean.zigbee_permit_join

Додаток В Конфігураційний блок HomeAssistant для використання нечіткої логіки для керування температурою в кімнаті

```
- id: heating_room_start
  alias: 'Heating room Start'
  trigger:
    - platform: state
  entity_id: input_boolean.heating_state_room_1
    from: 'off'
    to: 'on'
    - platform: state
  entity_id: input_boolean.heating_state_room_2
    from: 'off'
    to: 'on'
    - platform: state
  entity_id: input_boolean.heating_thermostat_request
    from: 'off'
    to: 'on'
  condition:
    condition: and
    conditions:
      - condition: state
  entity_id: input_boolean.heating_state_room_1
    state: 'on'
      - condition: state
  entity_id: input_boolean.heating_state_room_2
    state: 'on'
      - condition: state
  entity_id: input_boolean.heating_thermostat_request
    state: 'on'
  action:
    - service: switch.turn_on
  entity_id: switch.radiator

- id: heating_room_stop
  alias: 'Heating room Stop'
  trigger:
    - platform: state
  entity_id: input_boolean.heating_state_room_1
    from: 'on'
    to: 'off'
    - platform: state
  entity_id: input_boolean.heating_state_room_2
    from: 'on'
    to: 'off'
    - platform: state
  entity_id: input_boolean.heating_thermostat_request
    from: 'on'
    to: 'off'
  condition:
```

```
condition: and
conditions:
  - condition: state
entity_id: input_boolean.heating_state_room_1
  state: 'off'
  - condition: state
entity_id: input_boolean.heating_state_room_2
  state: 'off'
  - condition: state
entity_id: input_boolean.heating_thermostat_request
  state: 'off'
action:
  - service: switch.turn_off
entity_id: switch.radiator
```