

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

КВАЛІФІКАЦІЙНА РОБОТА
на тему: «АВТОНОМНА СИСТЕМА РОЗУМНОГО БУДИНКУ З
ЛОКАЛЬНОЮ ОБРОБКОЮ ДАНИХ»

на здобуття освітнього ступеня магістра
зі спеціальності 123 Комп'ютерної інженерії
(код, найменування спеціальності)
освітньо-професійної програми Комп'ютерні системи та мережі
(назва)

*Кваліфікаційна робота містить результати власних досліджень. Використання
ідей, результатів і текстів інших авторів мають посилання на відповідне
джерело*

(підпис)	Денис ПИСАРЕНКО
	Ім'я, ПРІЗВИЩЕ здобувача

Виконав:	здобувач(ка) вищої освіти гр. <u>КСДМ-61</u> <u>Денис ПИСАРЕНКО</u> Ім'я, ПРІЗВИЩЕ
Керівник: науковий ступінь, вчене звання	<u>Наталія ЛАЩЕВСЬКА</u> Ім'я, ПРІЗВИЩЕ
Рецензент: науковий ступінь, вчене звання	Ім'я, ПРІЗВИЩЕ

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**
Навчально-науковий інститут Інформаційних технологій

Кафедра Комп'ютерної інженерії

Ступінь вищої освіти Магістр

Спеціальність 123 «Комп'ютерна інженерія»

Освітньо-професійна програма Комп'ютерні системи та мережі

ЗАТВЕРДЖУЮ

Завідувач кафедри Наталія ЛАЩЕВСЬКА

_____ Ім'я, ПРІЗВИЩЕ
«_____» _____ 2024р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Писаренко Денис Сергійович

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: «Автономна система розумного будинку з локальною обробкою даних»

керівник кваліфікаційної роботи: Наталія ЛАЩЕВСЬКА, канд.техн.наук, доцент

(Ім'я, ПРІЗВИЩЕ, науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «15» жовтня 2024р. №320

2. Строк подання кваліфікаційної роботи «29» грудня 2024р.

3. Вихідні дані до кваліфікаційної роботи: Науково-технічна література, технологічні стандарти та протоколи. Дизайн і архітектура системи. Технічні характеристики пристроїв і вимоги. Інтеграція компонентів і функціональність системи. Оцінка ефективності, порівняльний аналіз та бенчмаркінг.

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

1. Теоретичні основи автономних систем розумного будинку
2. Порівняння локальної обробки даних із хмарними рішеннями: переваги та недоліки
3. Розробка та реалізація архітектури автономної системи розумного будинку
4. Інтеграція сенсорів, контролерів і виконавчих пристроїв у локальну мережу

5. Перелік ілюстративного матеріалу: презентація

1. Мета, об'єкт, предмет дослідження
2. Актуальність роботи
Проблеми традиційних систем розумного будинку та необхідність автономних рішень
3. Порівняння хмарних і локальних обчислень у контексті IoT
4. Архітектура автономної системи розумного будинку
5. Демонстрація функціональних можливостей: використання власного локального сервера для моніторингу і управління
6. Інтеграція з Proxmox і Home Assistant для управління системою розумного будинку
7. Висновки

8. Публікації та апробація роботи

6. Дата видачі завдання «01» вересня 2024р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Збір даних	10.10-12.10.2024	
2	Обробка матеріалу	12.10-15.10.2024	
3	Розробка теоретичної частини	15.10-18.10.2024	
4	Розробка архітектури системи	18.10-22.10.2024	
5	Інтеграція обладнання та програмного забезпечення	22.10-26.10.2024	
6	Оптимізація системи	26.10-30.11.2024	
7	Висновки за результатами аналізу	30.11-02.12.2024	
8	Розробка презентації	02.12-08.12.2024	
9	Передзахист	09.12-11.12.2024	
10	Здача в деканат	20.12-29.12.2024	

Здобувач вищої освіти

(підпис)

Денис Писаренко

(Ім'я, ПРІЗВИЩЕ)

Керівник кваліфікаційної роботи

(підпис)

Наталія ЛАЩЕВСЬКА

(Ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина бакалаврської роботи 80 с., 18 рис., 50 джерел.

АВТОНОМНА СИСТЕМА РОЗУМНОГО БУДИНКУ, ЛОКАЛЬНА ОБРОБКА ДАНИХ, IoT, СЕНСОРИ, БЕЗПЕКА, ЕНЕРГОЕФЕКТИВНІСТЬ, МАСШТАБУВАННЯ, НАДІЙНІСТЬ.

Актуальність теми – У сучасному світі зростає попит на розумні будинки, які здатні забезпечити комфорт, енергоефективність, безпеку та незалежність від зовнішніх серверів. Розробка автономної системи розумного будинку з локальною обробкою даних дозволяє підвищити рівень конфіденційності, швидкості обробки інформації та надійності роботи, що є надзвичайно актуальним у наш час.

Об'єктом дослідження є процес функціонування розумного будинку з локальною обробкою даних, який характеризується необхідністю забезпечення безперебійної роботи пристроїв, ефективного використання енергоресурсів, високої швидкості обробки даних, а також захисту інформації в умовах автономної роботи. Цей процес породжує проблемні ситуації, пов'язані з інтеграцією нових пристроїв, оптимізацією обробки даних і підвищенням надійності системи.

Предмет дослідження – функціональні можливості та переваги автономних систем розумного будинку, що базуються на локальній обробці даних.

Мета роботи – створення автономної системи розумного будинку, яка забезпечує функціональність, безпеку, конфіденційність даних та зручність у використанні завдяки локальній обробці інформації.

Методи дослідження – аналіз наявних комерційних та відкритих платформ, порівняння рішень локальної та хмарної обробки даних, розробка та тестування запропонованої системи.

У роботі висвітлено переваги локальної обробки даних у системах розумного будинку, що дозволяє забезпечити їхню роботу навіть за

відсутності інтернет-з'єднання, підвищує безпеку даних користувачів та зменшує залежність від зовнішніх серверів.

Результатом роботи є розробка автономної системи розумного будинку з використанням сучасних сенсорів, алгоритмів обробки даних та енергоефективних рішень. Система демонструє простоту інтеграції, високий рівень надійності та масштабованість для використання в різних типах будинків.

ABSTRACT

The text part of the bachelor's thesis consists of 80 pages, 18 figures, and 50 references.

AUTONOMOUS SMART HOME SYSTEM, LOCAL DATA PROCESSING, IoT, SENSORS, SECURITY, ENERGY EFFICIENCY, SCALABILITY, RELIABILITY.

Relevance of the topic – In the modern world, the demand for smart homes capable of providing comfort, energy efficiency, security, and independence from external servers is growing. The development of an autonomous smart home system with local data processing enhances privacy, increases the speed of data processing, and improves system reliability, which is highly relevant today.

The object of research is the operation process of a smart home system with local data processing, characterized by the need to ensure uninterrupted device performance, efficient use of energy resources, high-speed data processing, and secure information handling in autonomous conditions. This process generates challenges related to integrating new devices, optimizing data processing, and enhancing system reliability.

The subject of research is the functional capabilities and advantages of autonomous smart home systems based on local data processing.

The goal of the work is to develop an autonomous smart home system that ensures functionality, security, data privacy, and user convenience through local information processing.

The methods of research include the analysis of existing commercial and open platforms, comparison of local and cloud data processing solutions, and the design and testing of the proposed system.

The thesis highlights the advantages of local data processing in smart home systems, enabling them to operate even without an Internet connection, enhancing user data security, and reducing dependency on external servers.

The result of the work is the development of an autonomous smart home system using modern sensors, data processing algorithms, and energy-efficient solutions. The system demonstrates ease of integration, high reliability, and scalability for use in various types of homes.

ЗМІСТ

ВСТУП	8
1 ТЕОРЕТИЧНІ ОСНОВИ АВТОНОМНИХ СИСТЕМ РОЗУМНОГО БУДИНКУ	10
1.1 Поняття та еволюція розумного будинку	10
1.2 Функціональні можливості сучасних розумних будинків	17
1.3 Основні технології та протоколи IoT для розумних будинків	21
1.4 Переваги локальної обробки даних у системах розумного будинку	28
2 АНАЛІЗ ІСНУЮЧИХ ТЕХНОЛОГІЙ ТА РІШЕНЬ	31
2.1 Огляд сучасних систем розумного будинку	31
2.2 Порівняльний аналіз локальної та хмарної обробки даних	37
2.3 Проблеми та виклики в реалізації автономних систем	41
3 РОЗРОБКА АРХІТЕКТУРИ АВТОНОМНОЇ СИСТЕМИ РОЗУМНОГО БУДИНКУ	45
3.1 Вибір апаратних компонентів та сенсорів	45
3.2. Проектування програмного забезпечення для локальної обробки	55
3.3 Інтеграція компонентів та протоколів зв'язку	66
3.4 Безпека та конфіденційність даних у локальних системах	70
3.5 Можливості масштабування та вдосконалення системи	75
ВИСНОВКИ	80
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	81
ДОДАТОК А	87
ДОДАТОК Б	88
ДОДАТОК В	89
ДОДАТОК Г	90
ДОДАТОК Д ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)	91

ВСТУП

Забезпечення комфорту, безпеки та ефективного управління ресурсами стає все більш актуальним завданням у сфері сучасного житла. Інноваційні рішення, що використовуються у «розумних» будинках, дозволяють автоматизувати виконання багатьох рутинних дій, сприяють зниженню енергоспоживання та підвищують загальну якість життя. Залежність таких систем від зручності керування та доступності даних визначає потребу в оптимальному підході до обробки інформації.

Одним із ключових аспектів побудови сучасних розумних будинків є вибір між локальною та хмарною обробкою даних. Цей вибір є важливим, оскільки він впливає на безпеку, швидкість доступу до даних та загальну надійність системи.

В рамках дослідження були поставлені та вирішені наступні завдання: вивчення, аналіз та порівняння комерційних та відкритих платформ для розумних будинків; оцінка вимог до обладнання та сенсорів для забезпечення автономної роботи системи; проведення порівняльного аналізу локальної та хмарної обробки даних за критеріями безпеки, швидкості, надійності, вартості та масштабованості.

Метою роботи є розробка автономної системи розумного будинку з локальною обробкою даних, яка забезпечує високу ефективність управління ресурсами та підвищує рівень безпеки і конфіденційності. Для досягнення цієї мети було поставлено декілька завдань: аналіз існуючих рішень для систем розумних будинків та виявлення їх переваг і недоліків; вибір оптимальних компонентів для реалізації локальної обробки даних у розумному будинку; розробка концепції архітектури автономної системи, яка базується на локальній обробці даних

Об'єктом дослідження є процес автоматизації управління ресурсами в розумному будинку, що базується на локальній обробці даних. Предметом дослідження є підходи до обробки даних у системах розумного будинку,

зокрема порівняння локальної та хмарної обробки за основними критеріями ефективності.

Наукова новизна полягає у розробці підходу до автономного управління розумним будинком, який базується на локальній обробці даних, що дозволяє забезпечити більший рівень конфіденційності і зменшити залежність від інтернет-з'єднання. Практична значущість полягає у тому, що результати дослідження можуть бути використані для розробки комерційних систем розумного будинку, що потребують високого рівня надійності та безпеки. Зокрема, ці результати можуть знайти застосування у галузі охоронних систем для житлових приміщень, автономних енергетичних систем, а також у розробці платформ для автоматизації процесів управління ресурсами у розумних будинках, таких як освітлення і клімат-контроль.

Теоретична значущість роботи полягає у розробці моделі автономної обробки даних для систем розумного будинку, яка може бути застосована для подальших досліджень у галузі автоматизації житлових приміщень. Методична значущість полягає у запропонованих підходах до проектування таких систем, що можуть бути використані для розробки інших подібних проектів. Практична значущість включає можливість застосування результатів дослідження на конкретних підприємствах або для розробки автономних систем для комерційного використання.

1 ТЕОРЕТИЧНІ ОСНОВИ АВТОНОМНИХ СИСТЕМ РОЗУМНОГО БУДИНКУ

1.1. Поняття та еволюція розумного будинку

Розумний будинок являє собою технологічно інтегровану житлову систему, яка використовує автоматизовані рішення для забезпечення комфорту, безпеки та енергоефективності [1]. Сучасні підходи до визначення розумного будинку варіюються від вузького розуміння як набору окремих автоматизованих пристроїв до широкого трактування, що передбачає інтегроване середовище, яке адаптивно реагує на поведінку користувачів[2].

Одним із підходів є розгляд розумного будинку як системи з високим рівнем автоматизації, що забезпечує контроль над різними аспектами житла, такими як освітлення, кліматичні умови, безпека, а також управління побутовими приладами за допомогою централізованих або віддалених механізмів [3]. Інший підхід зосереджується на розумному будинку як на динамічній екосистемі, яка не тільки автоматизує побутові функції, але й аналізує поведінкові патерни мешканців, адаптується до їхніх індивідуальних потреб, оптимізує використання ресурсів і забезпечує зворотний зв'язок для вдосконалення функціональності системи [4].

Розумний будинок характеризується рядом ключових особливостей, що відрізняють його від традиційного житла. Першою характерною ознакою є високий рівень інтеграції між різними підсистемами. Наприклад, система освітлення може бути інтегрована з системами безпеки та клімат-контролю для створення комплексних сценаріїв автоматизації, що забезпечують оптимальні умови для мешканців. Другою важливою ознакою є підвищений комфорт, який досягається завдяки персоналізованим налаштуванням, орієнтованим на індивідуальні потреби кожного користувача. Третім важливим аспектом є підвищена ефективність використання енергоресурсів, що забезпечується через можливості аналізу споживання та автоматичного регулювання параметрів системи [5].

Безпека є однією з ключових функцій розумного будинку. Вона досягається за рахунок інтеграції різноманітних сенсорів, камер спостереження та інших засобів моніторингу, що забезпечують постійний контроль як внутрішніх приміщень, так і прилеглої території. Системи безпеки можуть автоматично виявляти підозрілу активність, оперативно інформувати мешканців про можливі загрози і навіть здійснювати захисні дії для збереження майна та життя користувачів.

Зручність використання також є важливою складовою концепції розумного будинку. Системи управління, інтегровані з мобільними додатками, дозволяють користувачам контролювати основні функції будинку за допомогою смартфонів чи інших пристроїв незалежно від їхнього місцезнаходження. Це включає як базові функції, такі як управління температурою або освітленням, так і можливість отримувати інформацію про стан системи в реальному часі.

Іншою важливою рисою розумного будинку є використання алгоритмів штучного інтелекту для адаптації до звичок мешканців. Наприклад, система здатна навчатися розпізнавати повторювані патерни в поведінці мешканців, такі як ввімкнення певного освітлення в конкретний час доби, і автоматично здійснювати ці дії, що зменшує необхідність виконання рутинних завдань і підвищує рівень комфорту.

1. Початкові етапи: Перші системи автоматизації

Історія розвитку концепції розумного будинку бере свій початок з середини XX століття, коли з'явилися перші спроби автоматизувати повсякденні процеси в побуті. Перші системи автоматизації були засновані на механічних і електричних пристроях, що виконували базові функції, такі як контроль освітлення та опалення [6]. Ці системи мали обмежений функціонал і орієнтувалися виключно на вирішення вузького кола завдань, не маючи можливості взаємодії між різними компонентами. Ранні дослідження та розробки у галузі автоматизації, зокрема використання електромеханічного управління, проклали шлях для подальшого розвитку технологій комфортного

проживання. У цей період основні зусилля спрямовувалися на створення пристроїв, що полегшували виконання рутинних завдань і сприяли підвищенню зручності для користувачів. Перші системи автоматизації, такі як електричні вимикачі світла та прості термостати, хоча й мали обмежену функціональність, стали важливим кроком на шляху до подальших технологічних інновацій.

Системи автоматизації середини XX століття залишалися ізольованими одна від одної, що значно обмежувало їх можливості. Проте, ці початкові розробки мали значний вплив на майбутнє, заклавши основу для формування концепції централізованого управління. Перші експерименти з електромеханічними пристроями продемонстрували потенціал автоматизації для підвищення якості життя, хоча в ті часи ця технологія була доступна тільки вузькому колу користувачів через високу вартість та технічні обмеження.

2. Розвиток технологій: Впровадження комп'ютерних технологій та поява Інтернету речей (IoT)

Наступним важливим етапом в еволюції розумного будинку стало впровадження комп'ютерних технологій у 1970-1980-х роках. Поява мікропроцесорів та напівпровідникових пристроїв дозволила значно підвищити рівень автоматизації, зробивши можливим створення програмованих систем управління [7]. Ці системи не лише виконували базові функції, а й дозволяли створювати гнучкі сценарії автоматизації, що адаптувалися до конкретних умов, наприклад, автоматичне вмикання освітлення при виявленні присутності. Впровадження комп'ютерних технологій не тільки полегшило реалізацію автоматизованих процесів, але й відкрило можливості для їх інтеграції у більш складні системи управління житлом.

Важливим проривом стало підключення цих систем до локальних мереж, що дало змогу централізовано контролювати різні аспекти функціонування житла. Локальні мережі забезпечили новий рівень управління, коли різні компоненти розумного будинку могли обмінюватися даними та працювати у взаємозв'язку, створюючи комплексні сценарії автоматизації. Наприклад, система безпеки могла автоматично сповіщати про рух у приміщенні, водночас

активуючи систему освітлення та відеоспостереження. Такий підхід дозволив значно підвищити рівень безпеки і зручності для користувачів.

Подальший етап розвитку пов'язаний з виникненням концепції Інтернету речей (IoT). Ця концепція передбачала інтеграцію фізичних пристроїв у глобальну мережу Інтернет, що забезпечило можливість віддаленого управління системами розумного будинку та дозволило інтегрувати різні пристрої в єдину екосистему. Впровадження IoT забезпечило двосторонній зв'язок між користувачем та системою, а також між компонентами розумного будинку, що значно розширило функціональні можливості цих систем. Завдяки IoT розумні будинки стали більш гнучкими та адаптивними, а їхній функціонал розширився до таких аспектів, як безпека, моніторинг ресурсів, контроль клімату, освітлення та багато іншого.

Інтернет речей також дав можливість створювати нові сервіси на основі зібраних даних, що відкривало додаткові можливості для покращення роботи системи [8]. Наприклад, за допомогою аналізу даних про споживання енергії можна було автоматично оптимізувати режими роботи електроприладів, тим самим знижуючи витрати та підвищуючи енергоефективність. Віддалене управління дозволило користувачам контролювати стан свого будинку незалежно від місця перебування, що значно підвищило рівень зручності та безпеки.

3. Сучасний етап: Інтеграція з хмарними сервісами та розвиток штучного інтелекту

Сучасний етап розвитку розумних будинків характеризується активною інтеграцією з хмарними сервісами. Використання хмарних обчислень забезпечило можливість зберігання та обробки великих обсягів даних, що генеруються системами розумного будинку, підвищуючи ефективність їхньої роботи. Завдяки хмарним сервісам з'явилася можливість віддаленого управління системами за допомогою мобільних пристроїв, що зробило розумний будинок доступним і зручним для користувачів незалежно від їхнього місцезнаходження. Інтеграція з хмарними сервісами також відкрила можливість

для постійного оновлення програмного забезпечення, що забезпечувало впровадження нових функцій і підвищення надійності систем.

Хмарна інтеграція також дозволяє використовувати алгоритми машинного навчання для аналізу поведінки мешканців та автоматичного вдосконалення функцій будинку, підвищуючи його адаптивність. Машинне навчання забезпечує аналіз великих обсягів даних для визначення закономірностей у поведінці мешканців, що дозволяє оптимізувати використання ресурсів. Наприклад, система може автоматично навчатися оптимальному графіку освітлення або опалення на основі спостережень за поведінкою мешканців, що зменшує енергоспоживання і забезпечує більший комфорт.

Ключовим елементом сучасного розумного будинку є штучний інтелект. Інтеграція алгоритмів штучного інтелекту дозволяє створювати системи, що здатні навчатися на основі зібраних даних про поведінку користувачів і автоматично приймати рішення для підвищення комфорту та енергоефективності. Системи штучного інтелекту можуть аналізувати не лише поточні дії мешканців, але й робити прогнози щодо майбутніх потреб. Наприклад, аналізуючи звички користувачів, система може передбачити, коли їм знадобиться гаряча вода або опалення, і підготувати ресурси заздалегідь, що сприяє підвищенню ефективності роботи [9].

Штучний інтелект також дозволяє розумним будинкам передбачати потреби користувачів і адаптуватися до змінних умов, роблячи житло не лише комфортнішим, але й більш ефективним з точки зору використання ресурсів. Адаптивність розумного будинку забезпечується через безперервний процес збору даних, аналізу та оновлення алгоритмів управління. Це дозволяє враховувати зміни у звичках мешканців, сезонні коливання та інші фактори, що впливають на функціонування систем.

Однією з ключових тенденцій у розвитку технологій розумного будинку є посилення ролі локальної обробки даних, що стає дедалі актуальнішим в умовах сучасної цифрової екосистеми. Локальна обробка даних передбачає

виконання обчислень і обробку інформації безпосередньо на пристроях, розташованих у межах будинку, а не на віддалених хмарних серверах. Такий підхід має численні переваги, зокрема зменшення залежності від інфраструктури хмарних обчислень, що дозволяє підвищити автономність системи і знизити ризики, пов'язані з перебоями в роботі мережі Інтернет [10].

Зростання ролі локальної обробки зумовлено потребою у забезпеченні низької затримки при виконанні важливих операцій, що критично необхідні для багатьох функцій розумного будинку. Наприклад, для систем управління безпекою або кліматичними умовами швидкість реакції системи є вирішальним фактором, від якого залежить рівень комфорту та безпеки мешканців. Локальна обробка дозволяє суттєво знизити затримку під час виконання команд, забезпечуючи майже миттєвий зворотний зв'язок.

Крім того, локальна обробка даних підвищує рівень безпеки та конфіденційності. Зберігання чутливих даних безпосередньо на локальних пристроях зменшує ризик витоку інформації під час її передачі через мережу. Це особливо важливо в умовах зростання кількості кіберзагроз, які можуть бути спрямовані на отримання доступу до персональної інформації мешканців. Рішення щодо локальної обробки забезпечують більший контроль над даними, дозволяючи уникати потенційних загроз, пов'язаних із використанням хмарних сервісів сторонніх компаній.

Окремої уваги заслуговує здатність локальних обчислень сприяти підвищенню енергоефективності систем розумного будинку. Використання локальних ресурсів для обробки інформації дозволяє мінімізувати потребу у передачі великих обсягів даних до хмарних серверів, що сприяє зниженню енерговитрат і оптимізації використання доступних ресурсів [11]. Це особливо важливо в умовах, коли екологічна стійкість стає одним з пріоритетів розвитку сучасних технологій.

З розвитком технологій розумного будинку питання безпеки та конфіденційності даних набувають усе більшої важливості, оскільки розумні системи обробляють значні обсяги персональної інформації, яка може бути

цінною ціллю для зловмисників. Підвищення рівня безпеки стало одним із основних напрямків розвитку, що включає комплексні заходи на рівні апаратного забезпечення, програмного забезпечення та мережових протоколів.

Для захисту даних використовуються новітні методи шифрування, що забезпечують конфіденційність під час передачі даних між пристроями. Шифрування забезпечує захист даних як під час їхнього зберігання, так і під час передачі, що зменшує ризик несанкціонованого доступу. Крім того, активно розробляються та впроваджуються нові методи аутентифікації, такі як багатофакторна аутентифікація (MFA), біометричні дані та унікальні криптографічні ключі, що унеможливають доступ до системи для неавторизованих користувачів [12].

Сучасні системи моніторингу, які використовуються в розумних будинках, мають здатність не тільки виявляти потенційні загрози, але й аналізувати типові поведінкові патерни для ідентифікації аномалій. Це дозволяє виявляти спроби несанкціонованого доступу на ранніх етапах, реагувати на загрози в режимі реального часу і забезпечувати більш високий рівень захисту для мешканців. Інтелектуальні системи моніторингу також здатні навчатися, що підвищує їхню ефективність у виявленні загроз завдяки безперервному вдосконаленню алгоритмів на основі зібраних даних.

Крім того, питання конфіденційності пов'язане з необхідністю обмеження доступу до персональної інформації сторонніх компаній. Використання закритих мереж і локальних серверів дозволяє уникнути передачі персональних даних за межі будинку, що мінімізує ризики, пов'язані з можливим зловживанням інформацією. Забезпечення конфіденційності стає не просто вимогою користувачів, а й важливим критерієм конкурентоспроможності на ринку розумних будинків, де довіра до безпеки рішень є вирішальним фактором для споживачів.

На основі проведеного аналізу концепції розумного будинку, можна узагальнити ключові аспекти його еволюції, а також визначити місце цього дослідження у загальному контексті розвитку технологій. Розумний будинок

пройшов кілька важливих етапів розвитку, починаючи від перших спроб автоматизації побутових функцій до сучасних складних інтегрованих систем, що використовують штучний інтелект і хмарні обчислення для забезпечення максимального комфорту, безпеки та енергоефективності.

Дане дослідження спрямоване на аналіз сучасного стану розвитку розумних будинків та визначення основних тенденцій, що впливають на їхнє майбутнє. Визначення ключових етапів еволюції дозволяє краще зрозуміти поточні виклики і можливості, а також напрямки подальших досліджень у сфері локальної обробки даних, підвищення безпеки та персоналізації. Розумний будинок, як концепція, є важливим елементом сучасного житлового середовища, що демонструє високий потенціал для подальшого розвитку завдяки прогресу в технологіях автоматизації, штучного інтелекту та інтернету речей.

Цей аналіз підкреслює значущість індивідуального підходу до кожного аспекту розумного будинку, що є критичним для забезпечення ефективності, безпеки та зручності використання. Технології розумного будинку не тільки полегшують повсякденне життя мешканців, але й забезпечують стійке використання енергоресурсів, сприяючи глобальним зусиллям із збереження енергії та зменшення екологічного впливу. Відповідно, розвиток розумних будинків стає однією з ключових тем у сфері сучасних технологій, яка має на меті зробити життя людей більш комфортним, безпечним та сталим.

1.2 Функціональні можливості сучасних розумних будинків

Функціональні можливості сучасних розумних будинків є основоположними елементами їхньої інтеграції в якості технологічно просунутих і автоматизованих житлових середовищ. Під "функціональністю" в контексті розумного будинку розуміють здатність системи забезпечувати ефективне виконання різноманітних завдань, спрямованих на підвищення комфорту, безпеки, енергоефективності та зручності для мешканців [13]. Це

охоплює не лише автоматизацію рутинних побутових процесів, але й інтеграцію управління житловими системами, що істотно підвищує якість життя мешканців та оптимізує використання ресурсів.

Функціональні можливості розумного будинку важливі для користувачів з багатьох причин. По-перше, вони мінімізують зусилля, необхідні для виконання повсякденних побутових завдань, що особливо актуально в умовах інтенсивного сучасного ритму життя. Системи автоматизації сприяють автономності, даючи змогу користувачам зосередитися на більш важливих справах, тоді як автоматизовані рішення беруть на себе контроль над рутинними процесами. Важливим аспектом також є підвищення рівня безпеки через можливість автоматичного реагування на потенційні загрози, такі як витік газу, затоплення або проникнення сторонніх осіб. Персоналізоване середовище, яке динамічно адаптується до індивідуальних потреб мешканців, створює максимальний рівень комфорту і задоволення від проживання.

Сучасні системи розумного будинку орієнтовані на створення інтуїтивного управління, що забезпечує оптимальну зручність користування. Наприклад, розумні термостати можуть автоматично регулювати температуру в приміщенні, ґрунтуючись на поведінкових шаблонах мешканців і зовнішніх умовах, що дозволяє знижувати необхідність у ручному налаштуванні системи опалення або кондиціонування. Також системи управління освітленням і безпекою можуть автоматизувати налаштування світлових режимів, перевіряти стан дверей та вікон, а також своєчасно інформувати користувачів про будь-які підозрілі активності в будинку.

1. Автоматизація освітлення

Автоматизація освітлення є важливою складовою сучасних розумних будинків, що підвищує комфорт та енергоефективність. Інтелектуальні системи освітлення дозволяють регулювати яскравість, колір та інтенсивність світла залежно від контексту — для відпочинку, роботи чи інших видів активності. Освітлення може автоматично налаштовуватися на основі часу доби, рівня природного освітлення або попередньо визначених сценаріїв. Інтеграція з

датчиками руху знижує енергоспоживання, вмикаючи світло лише тоді, коли це необхідно [14].

2. Системи безпеки

Системи безпеки є критичною складовою розумного будинку. Камери відеоспостереження, оснащені інфрачервоними датчиками та сенсорами руху, забезпечують цілодобовий моніторинг внутрішніх і зовнішніх зон будинку. Доступ до відеопотоків через мобільні додатки дає змогу власникам спостерігати за об'єктом у реальному часі. Сигналізаційні системи автоматично реагують на загрози, такі як пожежа чи спроба проникнення, сповіщаючи мешканців та служби безпеки, а також активуючи запобіжні заходи, як-от блокування дверей чи увімкнення освітлення [15].

3. Контроль доступу

Контроль доступу забезпечується за допомогою електронних замків, біометричних систем та систем розпізнавання обличчя. Електронні замки дозволяють керувати доступом через мобільні додатки, а біометричні системи підвищують безпеку, усуваючи необхідність у фізичних ключах. Також можливо налаштувати тимчасові права доступу для гостей або обслуговуючого персоналу. Всі події доступу журналізуються, що дозволяє здійснювати аудит безпеки [16].

4. Клімат-контроль

Клімат-контроль забезпечує автоматизоване регулювання параметрів мікроклімату для комфорту мешканців. Системи управління опаленням, вентиляцією та кондиціонуванням (HVAC) інтегруються з датчиками температури та вологості для адаптації режимів роботи. Наприклад, система може підігрівати приміщення до прибуття мешканців або охолоджувати кімнати в спекотний день. Інтелектуальні алгоритми враховують час доби, активність мешканців і тарифи на енергію для мінімізації витрат [17].

5. Енергозбереження

Енергозбереження є ключовою функцією розумного будинку, спрямованою на раціональне використання ресурсів. Інтелектуальні системи

управління енергоспоживанням моніторять роботу електроприладів та автоматично вимикають або переводять їх у режим низького енергоспоживання, якщо вони не використовуються. Інтеграція з відновлюваними джерелами енергії, такими як сонячні панелі, додатково знижує залежність від зовнішніх джерел та сприяє економії витрат [18].

6.Комфорт

Розумні будинки підвищують комфорт мешканців завдяки автоматизації побутових процесів. Інтелектуальні системи освітлення, клімат-контролю та мультимедійного управління створюють адаптивне середовище, що відповідає вподобанням користувачів. Наприклад, освітлення може змінювати колірну температуру залежно від часу доби, а клімат-контроль підтримує оптимальну температуру.

7.Економія енергії

Розумні будинки ефективно управляють енергоспоживанням. Інтелектуальні системи автоматично вимикають невикористовувані пристрої та переводять їх у режим низького енергоспоживання, аналізуючи поведінкові шаблони мешканців. Інтеграція з відновлюваними джерелами енергії, такими як сонячні панелі, знижує витрати та підвищує енергоефективність [19].

8.Безпека

Системи безпеки розумних будинків включають відеоспостереження, сигналізації та контроль доступу, забезпечуючи комплексний захист. Камери інтегруються з мобільними додатками для моніторингу, а сигналізації автоматично реагують на загрози, такі як пожежа або проникнення, сповіщаючи власників та служби безпеки [20].

9.Персоналізація

Персоналізація дозволяє створити індивідуалізоване середовище, адаптоване до потреб кожного мешканця. Технології штучного інтелекту аналізують звички користувачів і налаштовують параметри освітлення, температури та мультимедійного супроводу. Це забезпечує комфорт і

інтуїтивність взаємодії — система самостійно оптимізує середовище під потреби користувачів [21].

Таким чином, функціональні можливості сучасних розумних будинків охоплюють широкий спектр рішень для автоматизації освітлення, безпеки, клімат-контролю та енергозбереження, що сприяє підвищенню рівня комфорту, безпеки та енергоефективності. Інтеграція цих технологій дозволяє створити житлове середовище, що динамічно адаптується до потреб мешканців, забезпечуючи оптимальні умови для життя і стійке використання ресурсів.

1.3 Основні технології та протоколи IoT для розумних будинків

Метою цього підрозділу є детальний аналіз ключових технологій та протоколів, які забезпечують функціонування розумних будинків. У центрі уваги знаходяться технічні характеристики та стандарти, що застосовуються в індустрії, для забезпечення ефективної, надійної та безпечної інтеграції різноманітних компонентів інтелектуальних житлових систем.

Інтернет речей (IoT) відіграє фундаментальну роль у формуванні сучасних розумних будинків, забезпечуючи можливість об'єднання різноманітних пристроїв і систем в єдину інтегровану мережу. За допомогою IoT відбувається інтеграція сенсорів, контролерів та виконавчих пристроїв, що дозволяє реалізувати дистанційне керування, моніторинг та автоматизацію різних аспектів побутових процесів.

Надійна та ефективна робота IoT у розумних будинках потребує чіткої стандартизації та застосування узгоджених протоколів. Стандарти та протоколи забезпечують сумісність між пристроями різних виробників, що є критичним для побудови інтегрованих, масштабованих і безпечних систем [22]. Використання стандартизованих протоколів передачі даних дозволяє забезпечити взаємодію між пристроями, що працюють на різних платформах і в різних технологічних середовищах. Це також сприяє зменшенню складності

інтеграції різнорідних компонентів, що особливо важливо для гнучкого розширення та модернізації розумних будинків.

Бездротові протоколи є важливим елементом технології IoT для розумних будинків, забезпечуючи зв'язок між численними сенсорами, контролерами та іншими пристроями. Далі розглянуто основні протоколи, які використовуються для передачі даних у контексті інтегрованих IoT-систем у розумних будинках.

Wi-Fi є одним із найпоширеніших бездротових протоколів для розумних будинків завдяки своїй доступності та широкому впровадженню у побутових мережах [23]. Ця технологія дозволяє об'єднувати численні пристрої в одну мережу, забезпечуючи високу швидкість і надійність підключення як до інтернету, так і до внутрішніх ресурсів будинку.

Переваги Wi-Fi:

– Широка доступність: Wi-Fi є стандартом, який підтримується переважною більшістю побутових пристроїв, що дозволяє легко інтегрувати різні компоненти в систему розумного будинку.

– Висока пропускна здатність: Wi-Fi забезпечує високу швидкість передачі даних, що робить його придатним для завдань, де потрібна передача великих обсягів інформації, таких як відеоспостереження або потокове аудіо і відео.

Недоліки Wi-Fi:

– Високе енергоспоживання: Пристрої, що працюють на Wi-Fi, характеризуються високим енергоспоживанням порівняно з іншими IoT-протоколами, що може бути критичним для сенсорів, які живляться від батарейок.

– Залежність від мережевої інфраструктури: Робота пристроїв Wi-Fi потребує стабільної мережі та доступу до маршрутизаторів, що обмежує їх використання в умовах недостатньої якості зв'язку або у випадках нестабільного доступу до мережі.

Wi-Fi широко використовується для підключення інтелектуальних камер, аудіосистем, побутових приладів та інших високопродуктивних пристроїв. Його

висока пропускна здатність робить його оптимальним для завдань, які потребують постійного і швидкого з'єднання. Проте, для низькоенергетичних пристроїв, таких як сенсори температури або дверні контакти, більш ефективними можуть бути протоколи Zigbee або Z-Wave, оскільки вони краще оптимізовані з точки зору енергоспоживання.

Bluetooth є популярним протоколом для з'єднання пристроїв на короткій відстані, включно з пристроями для розумних будинків. Bluetooth Low Energy (BLE) є спеціалізованою версією Bluetooth, яка призначена для мінімізації енергоспоживання, що робить її оптимальним варіантом для IoT-пристроїв [24].

Bluetooth та BLE використовуються для підключення периферійних пристроїв, таких як датчики руху, дверні замки, а також носимі пристрої. BLE є особливо придатним для пристроїв, які не потребують постійного з'єднання, а лише короточасних передач невеликих обсягів даних.

Основною перевагою BLE є його низьке енергоспоживання, що дозволяє використовувати його у пристроях із живленням від батарей протягом тривалого часу без потреби у частому заряджанні або заміні.

Zigbee є одним із провідних бездротових протоколів, що спеціалізується на низькому енергоспоживанні та застосуванні у розумних будинках. Він призначений для створення сітчастих мереж, здатних забезпечувати зв'язок між численними пристроями без значних витрат на ресурси, що робить його ідеальним для сенсорів і контролерів, де довговічність роботи від батарей є критично важливою.

Однією з ключових переваг Zigbee є його здатність підтримувати мережі з наднизьким енергоспоживанням. Пристрої, що використовують цей протокол, можуть функціонувати від батарейок протягом кількох років без необхідності їх заміни, що є особливо привабливим для сенсорів, які встановлюються у важкодоступних місцях [25]. Висока енергоефективність є основним чинником, який сприяє широкому впровадженню Zigbee у розумних будинках, де ефективність використання енергії є пріоритетом.

Zigbee забезпечує високий рівень сумісності між пристроями різних виробників завдяки використанню відкритого стандарту. Це дозволяє створювати екосистему, у якій пристрої можуть взаємодіяти незалежно від бренду, що суттєво спрощує розширення мережі розумного будинку та інтеграцію нових компонентів. Завдяки високій сумісності, Zigbee стає ключовою технологією для побудови гнучких і масштабованих IoT-систем, які легко адаптуються до зростаючих вимог користувачів.

Z-Wave є спеціалізованим протоколом, розробленим для забезпечення надійної автоматизації розумних будинків. Його основні характеристики включають стабільність з'єднання, низьке енергоспоживання та здатність працювати у менш завантажених частотних діапазонах, що допомагає уникнути інтерференцій і забезпечити високу якість зв'язку.

Z-Wave працює на частотах 800-900 MHz, що надає йому унікальні переваги порівняно з іншими протоколами, які використовують більш завантажені частотні діапазони, такі як Wi-Fi або Bluetooth [26]. Це дозволяє Z-Wave забезпечувати стабільну роботу навіть у середовищах з великою кількістю інших бездротових мереж, що робить його оптимальним для домашнього застосування, де існує велика кількість конкуруючих сигналів.

Z-Wave вирізняється своєю високою надійністю та здатністю створювати сітчасті мережі, де кожен пристрій може виступати ретранслятором сигналу, що суттєво збільшує радіус дії мережі та підвищує її надійність. Окрім того, низьке енергоспоживання робить Z-Wave оптимальним для пристроїв, які працюють на батарейках, забезпечуючи довготривалу автономну роботу і мінімізуючи необхідність частого обслуговування.

Окрім традиційних протоколів, таких як Zigbee та Z-Wave, останніми роками активно розвиваються нові стандарти, зокрема Thread та Matter, спрямовані на підвищення сумісності між різними пристроями та забезпечення нового рівня ефективності для розумних будинків.

Thread є сучасним протоколом, що підтримує створення сітчастих мереж із вбудованою підтримкою IPv6, що дозволяє кожному пристрою отримувати

унікальну IP-адресу та легко інтегруватися у глобальну мережу [27]. Це забезпечує новий рівень масштабованості та підвищує можливості інтеграції пристроїв від різних виробників, роблячи мережі більш стійкими та динамічними.

Matter, зі свого боку, спрямований на стандартизацію комунікації між пристроями розумного будинку від різних виробників [28]. Це спільний проєкт провідних технологічних компаній, що має на меті створення уніфікованого стандарту для забезпечення сумісності, що суттєво спростить інтеграцію, забезпечить стабільний зв'язок та підвищить споживацький комфорт.

Ethernet залишається одним із найнадійніших способів забезпечення зв'язку для пристроїв IoT завдяки високій швидкості передачі даних та стабільності з'єднання.

Ethernet забезпечує винятково високу швидкість передачі даних і стабільність з'єднання, що робить його ідеальним для застосування в критично важливих системах, таких як охоронні сервери або централізовані системи управління [29]. Завдяки стабільності та відсутності затримок, Ethernet стає оптимальним вибором для обробки даних у режимі реального часу, де будь-які затримки можуть призвести до небажаних наслідків.

Основним недоліком Ethernet є необхідність кабельного підключення, що обмежує його використання для мобільних або важкодоступних пристроїв. Це робить Ethernet менш придатним для сценаріїв IoT, де потрібна гнучкість і мобільність, особливо в умовах, коли встановлення додаткової інфраструктури не є практичним.

PLC (Power Line Communication) є технологією, що використовує наявну електричну проводку для передачі даних, дозволяючи уникнути необхідності прокладання окремих комунікаційних кабелів. Це робить PLC привабливим варіантом для розумних будинків, які бажають використовувати існуючу інфраструктуру для забезпечення зв'язку.

PLC забезпечує передачу даних через звичайну електричну мережу, що спрощує інтеграцію нових пристроїв у існуючу інфраструктуру без значних

витрат на прокладання додаткових кабелів [30]. Це рішення особливо вигідне у великих будинках або будівлях, де прокладання нових ліній може бути фінансово і технічно непростим завданням.

Основною перевагою PLC є його здатність використовувати існуючу інфраструктуру електропроводки, що значно знижує витрати на встановлення і інтеграцію нових компонентів. Проте якість передачі сигналу може бути залежною від стану електричної мережі і бути під впливом перешкод, створених іншими електропристроями, що обмежує ефективність цієї технології у певних сценаріях використання. Деякі електричні системи можуть бути схильні до шумів та інтерференції, що може призвести до зниження пропускної здатності і стабільності передачі даних.

Платформи, такі як OpenHAB та Home Assistant, є одними з провідних рішень для управління розумними будинками, завдяки їхній гнучкості, розширюваності та підтримці інтеграції з широким спектром пристроїв. Обидві платформи пропонують централізоване управління пристроями розумного будинку через єдиний інтегрований інтерфейс, що дозволяє реалізувати складні сценарії автоматизації, контролювати освітлення, клімат-контроль, безпеку та інші важливі аспекти інтелектуального житла.

OpenHAB — це платформа з відкритим вихідним кодом, яка забезпечує високу сумісність із різноманітними пристроями та протоколами завдяки модульній архітектурі. Вона відзначається винятковою гнучкістю, що дозволяє користувачам створювати складні інтеграції та логіки на основі правил і сценаріїв, використовуючи її можливості для кастомізації. OpenHAB працює на різних платформах, таких як Windows, Linux, та Raspberry Pi, що робить її доступною для широкого кола користувачів, від початківців до просунутих ентузіастів [31].

Home Assistant також є платформою з відкритим вихідним кодом і користується активною підтримкою великої спільноти розробників і користувачів. Вона підтримує інтеграцію з численними протоколами, такими як Zigbee, Z-Wave, MQTT та іншими, що робить її одним з найбільш

універсальних рішень для розумних будинків [32]. Основна перевага Home Assistant полягає у її інтуїтивному веб-інтерфейсі та можливості створення автоматизацій без необхідності глибоких знань у програмуванні. Крім того, Home Assistant надає можливість інтеграції з хмарними сервісами, такими як Google Assistant і Amazon Alexa, що значно підвищує її функціональність та зручність у використанні.

Інші платформи, такі як Domoticz, Jeedom та SmartThings, також мають значний вплив на розвиток систем управління розумними будинками. Кожна з цих платформ має свої унікальні особливості та переваги, які можуть варіюватися від енергоефективності до специфічних регіональних інтеграцій або спрощеного налаштування для користувачів з меншими технічними знаннями [33]. Вибір оптимальної платформи значною мірою залежить від специфічних вимог користувача, наявного обладнання, а також бажаного рівня кастомізації та контролю над екосистемою розумного будинку.

API (Application Programming Interface) та SDK (Software Development Kit) є ключовими інструментами для розробки додатків і розширень для платформ розумного будинку. API дозволяє інтегрувати сторонні сервіси, додатки або пристрої з існуючими платформами, що надає широкі можливості для персоналізації, автоматизації та створення інноваційних сценаріїв управління [34]. Використання API дозволяє розробникам взаємодіяти з різними компонентами екосистеми розумного будинку, такими як сенсори, контролери та інші пристрої, відкриваючи можливості для налаштування і контролю всіх аспектів системи.

SDK, своєю чергою, забезпечує розробників усім необхідним для створення нових додатків, інтеграцій або пристроїв, що можуть працювати в межах обраної платформи. З використанням SDK розробники мають змогу створювати нові пристрої або розширювати функціональність уже наявних компонентів екосистеми. Наприклад, SDK для Home Assistant або OpenHAB надає детальну документацію, готові бібліотеки та приклади коду, що значно

спрощують процес розробки та прискорюють інтеграцію нових рішень у вже існуючу інфраструктуру [35].

API та SDK є ключовими інструментами, які дозволяють розробникам створювати додаткові функції та інтеграції, що адаптуються до специфічних потреб користувачів, і забезпечують можливість активного розвитку платформи. Їх використання сприяє побудові більш гнучкої та масштабованої екосистеми розумного будинку, яка відповідає вимогам сучасних користувачів, дозволяючи інтегрувати різноманітні пристрої та сервіси в єдину, добре керовану інфраструктуру. Це забезпечує не лише зручність користування, але й можливість створення висококастомізованих рішень, що відповідають вимогам навіть найбільш вимогливих користувачів.

1.4. Переваги локальної обробки даних у системах розумного будинку

Метою цього підрозділу є визначення концепції локальної обробки даних та обґрунтування її переваг порівняно з хмарними рішеннями в контексті розумних будинків. Розглядаються ключові характеристики локальної обробки, її відмінності від хмарної обробки, а також переваги, виклики та порівняння цих двох підходів.

Локальна обробка даних передбачає виконання всіх обчислювальних процесів безпосередньо на місці, тобто в межах інфраструктури розумного будинку, використовуючи локальні ресурси, як-от контролери, сенсори та інші пристрої. Цей підхід відрізняється від хмарної обробки, при якій дані передаються на віддалені сервери для обробки і зберігання.

Локальна обробка забезпечує безпосереднє виконання операцій, що сприяє високій швидкодії та покращеній безпеці [36]. Хмарна обробка, з іншого боку, пропонує перевагу масштабованості, проте має більшу залежність від стабільного інтернет-з'єднання, що може негативно впливати на конфіденційність та безпеку даних.

Переваги локальної обробки:

—Ключовою перевагою локальної обробки даних є підвищений рівень

конфіденційності та безпеки, а також можливість повного контролю над даними. Дані залишаються в межах будинку, що значно зменшує ризики компрометації, пов'язані з передачею інформації через публічні мережі, і мінімізує ймовірність несанкціонованого доступу. Такий підхід дозволяє користувачам реалізувати власні політики безпеки та доступу, контролюючи всі аспекти обробки та зберігання даних, і виключає залежність від сторонніх хмарних провайдерів, що знижує ризик порушення конфіденційності.

—Локальна обробка даних забезпечує мінімальні затримки та високу швидкодію, що є критично важливим для систем, які вимагають миттєвої реакції, таких як системи безпеки, клімат-контролю та управління освітленням [37]. Виконання операцій безпосередньо на місці дозволяє уникнути затримок, пов'язаних із передачею даних до віддалених серверів, що забезпечує стабільну роботу системи в реальному часі. Така оперативність є ключовою для ситуацій, де навіть коротка затримка може мати значні наслідки, як, наприклад, у випадках аварійних ситуацій або у випадках необхідності негайної зміни кліматичних параметрів.

—Ще однією важливою перевагою локальної обробки є її незалежність від стабільного інтернет-з'єднання. Хмарні системи цілком залежать від наявності та якості мережевого підключення, що може стати проблемою під час перебоїв зв'язку або в умовах обмеженого доступу. Локальна обробка дозволяє забезпечити функціонування критичних систем розумного будинку навіть за відсутності інтернет-з'єднання, що підвищує надійність і стабільність роботи. Особливо це важливо для систем безпеки та клімат-контролю, де кожна затримка може призвести до серйозних наслідків.

—Надійність є ще однією суттєвою перевагою локальної обробки даних. Системи, що використовують локальні обчислення, не залежать від стабільності зовнішніх серверів чи хмарних платформ. Це означає, що вони зберігають свою функціональність навіть у випадках втрати інтернет-з'єднання, забезпечуючи безперервність роботи критичних компонентів житлової інфраструктури. Для

систем безпеки, освітлення та управління кліматом ця автономія має надзвичайно важливе значення, адже гарантує користувачам надійність функціонування незалежно від зовнішніх умов.

Недоліки та виклики:

—Локальна обробка даних потребує наявності потужного обладнання, здатного забезпечити виконання всіх необхідних обчислювальних задач [38]. У багатьох випадках це може вимагати модернізації існуючої інфраструктури будинку, що може стати значним фінансовим навантаженням.

—Ще одним викликом є складність налаштування локальної обробки. Вона часто потребує технічних знань для встановлення, конфігурації та підтримки. Багатьом користувачам, які не володіють такими навичками, може знадобитися допомога фахівців, що збільшує час і витрати на впровадження таких систем.

—Локальні системи не мають автоматичних оновлень, що характерно для хмарних рішень. Користувачам необхідно самостійно слідкувати за оновленнями програмного забезпечення, що створює додаткові виклики щодо підтримки безпеки та функціональності системи на належному рівні.

Локальна обробка даних у системах розумного будинку пропонує низку суттєвих переваг, таких як конфіденційність, надійність, низька затримка та незалежність від інтернет-з'єднання. Попри певні виклики, пов'язані з технічними знаннями, підтримкою та початковими витратами, локальна обробка забезпечує користувачам максимальний контроль над їхніми даними, а також високий рівень безпеки та автономності. У світлі цих аспектів локальна обробка залишається важливим і перспективним підходом для тих, хто надає перевагу конфіденційності та незалежності у використанні розумних технологій для житла.

2 АНАЛІЗ ІСНУЮЧИХ ТЕХНОЛОГІЙ ТА РІШЕНЬ

2.1 Огляд сучасних систем розумного будинку

Системи розумного будинку набули широкого поширення завдяки зростаючій доступності та зручності технологій для автоматизації повсякденних завдань. Вони забезпечують інтеграцію різноманітних пристроїв, таких як системи освітлення, клімат-контроль, безпека та мультимедійні системи, у єдину, зручну для користувача мережу. Ці рішення дозволяють підвищити комфорт, безпеку та ефективність управління домом, одночасно скорочуючи витрати на енергію та мінімізуючи вплив на довкілля.

У цьому розділі будуть розглянуті основні типи сучасних систем розумного будинку, починаючи з популярних комерційних рішень, що вирізняються готовими рішеннями та широкою інтеграцією з іншими комерційними продуктами, і завершуючи відкритими платформами, які пропонують більш високий рівень персоналізації та контроль над даними. Комерційні рішення, такі як Google Nest, Amazon Alexa та Apple HomeKit, орієнтовані на простоту використання, пропонуючи користувачам інтуїтивний інтерфейс і широкий вибір підтримуваних пристроїв. Однак ці системи часто передбачають певну залежність від екосистеми обраного бренду, що може обмежити можливості для користувачів, які прагнуть інтегрувати пристрої інших виробників.

2.1.1 Google Nest

Google Nest представляє собою одну з найбільш розвинених платформ для автоматизації управління розумним будинком, яка надає користувачам комплексні можливості для контролю різноманітних параметрів житлового середовища. Ця система підтримує автоматизацію освітлення через інтеграцію з розумними лампами та світильниками, що дозволяє регулювати яскравість і кольорову температуру світла відповідно до потреб та обставин, таких як час доби чи настрої користувача. Що стосується клімату, Google Nest забезпечує зручний і гнучкий контроль температури завдяки розумним термостатам, які

можуть адаптуватися до часу доби та присутності користувачів, створюючи комфортні умови в приміщенні та знижуючи витрати на енергію.

У сфері безпеки Google Nest інтегрує розумні камери, датчики руху та електронні замки, що дозволяє здійснювати повний моніторинг і контроль безпеки приміщень [39]. Ці пристрої взаємодіють для запобігання проникненню сторонніх осіб та автоматичного сповіщення користувача у разі загрози.

Крім цього, інтеграція з мобільними додатками для управління і моніторингу дозволяє користувачам керувати системою Google Nest дистанційно через смартфон або планшет, забезпечуючи мобільний контроль за всіма аспектами системи. Завдяки алгоритмам штучного інтелекту, Google Nest також вміє адаптуватися до звичок і режиму користувачів, автоматично налаштовуючи режими роботи для зниження енергоспоживання та підвищення рівня комфорту.

Google Nest розроблена з урахуванням підтримки різноманітних стандартів зв'язку, таких як Wi-Fi, Zigbee та Z-Wave, що дозволяє інтегрувати численні пристрої різних виробників. Завдяки цьому, система є універсальною і може працювати з багатьма популярними пристроями, включаючи освітлювальні прилади, термостати, камери безпеки та сенсори різного призначення. Крім того, Google Nest надає розробникам відкритий API, який дозволяє інтегрувати нові пристрої та функції, тим самим розширюючи екосистему і можливості системи. Це робить Google Nest гнучкою та адаптивною платформою, здатною задовольнити широкий спектр потреб користувачів.

Завдяки додатку Google Home користувачі отримують гнучкий інструмент для налаштування і управління системою Google Nest. Через додаток можна створювати індивідуальні сценарії автоматизації, які запускаються за розкладом або при виконанні певних умов, таких як повернення користувача додому або зниження рівня освітленості. Система також використовує штучний інтелект для адаптації до потреб і звичок користувача, що дозволяє підвищити ефективність і зручність використання.

Таким чином, користувачі можуть створювати персоналізовані умови автоматизації, що робить Google Nest не тільки універсальною, але й індивідуально адаптованою системою.

Google Nest переважно функціонує на основі хмарної архітектури, що забезпечує централізовану обробку даних та управління пристроями через хмарні сервіси Google. Це рішення підвищує гнучкість у налаштуванні і управлінні системою та дозволяє користувачам отримувати доступ до пристроїв з будь-якого місця, де є Інтернет-з'єднання. Однак така архітектура також вимагає стабільного інтернет-з'єднання для повноцінного функціонування системи, оскільки значна частина обробки даних і прийняття рішень відбувається на серверах Google. З іншого боку, хмарна архітектура дозволяє швидко оновлювати програмне забезпечення і вдосконалювати функціональні можливості, що робить систему більш адаптивною та захищеною від загроз і вразливостей.

2.1.2 Amazon Alexa

Amazon Alexa є однією з найбільш розвинених і популярних платформ для розумного будинку, яка відома своєю зручністю завдяки інтеграції голосового управління та розширеним можливостям автоматизації. Одна з ключових функцій Alexa — голосове управління, що дозволяє користувачам здійснювати контроль за розумними пристроями за допомогою голосових команд. Це значно спрощує взаємодію з системою, надаючи інтуїтивний та швидкий спосіб керування освітленням, електроприладами та іншими компонентами розумного будинку.

Amazon Alexa також пропонує зручний контроль освітлення та електроприладів, забезпечуючи можливість автоматизувати роботу ламп, розеток та інших пристроїв [40]. Ця функція дозволяє налаштовувати розклад вмикання та вимикання пристроїв або використовувати команду для одночасного керування групою приладів.

У сфері безпеки Alexa інтегрується з розумними камерами, датчиками руху та електронними замками, що забезпечує моніторинг і підвищений рівень захисту приміщення. Ці пристрої можуть взаємодіяти один з одним, створюючи комплексну систему безпеки, яка сповіщає користувача про можливі загрози.

Для мультимедійних можливостей Alexa підтримує відтворення музики, новин та іншого контенту за допомогою голосових команд, а також інтеграцію з популярними стрімінговими сервісами, що робить її зручною для повсякденного використання у домогосподарстві.

Alexa також дозволяє створювати розширені сценарії автоматизації, відомі як "Routine". Ці сценарії можуть бути налаштовані на різноманітні тригери, включаючи час доби, погодні умови чи наявність людей у приміщенні, що забезпечує високий рівень персоналізації та зручності для користувачів.

Amazon Alexa підтримує широкий спектр розумних пристроїв, завдяки чому система має значну популярність серед користувачів. Відкрита платформа Alexa Skills дозволяє інтегрувати десятки тисяч пристроїв від різних виробників, включаючи освітлювальні прилади, системи безпеки, розумні розетки, термостати та інші сенсори. Alexa підтримує популярні протоколи зв'язку, такі як Wi-Fi, Zigbee та Bluetooth, що підвищує її сумісність із різноманітними пристроями. Amazon постійно розширює можливості Alexa, додаючи нові Skills, які забезпечують сумісність з новими пристроями та надають додаткові функції.

Amazon Alexa надає користувачам можливість гнучкого налаштування за допомогою функції "Routine". Користувачі можуть створювати комплексні сценарії, які виконуються автоматично на основі певних умов або тригерів, таких як час доби, зміна погодних умов, наявність людей у кімнаті тощо. Це дозволяє оптимізувати роботу системи під індивідуальні потреби користувачів, надаючи можливість персоналізації як для окремих пристроїв, так і для групи пристроїв. Функція "Routine" дозволяє налаштувати Alexa так, щоб вона відповідала конкретним звичкам та розпорядку дня користувачів, підвищуючи рівень комфорту та ефективності.

Amazon Alexa базується на хмарній архітектурі, де обробка голосових команд та управління пристроями відбувається через хмарні сервіси Amazon. Це дозволяє швидко та ефективно обробляти команди, а також надає користувачам доступ до багатьох функцій завдяки інтеграції з Alexa Skills. Проте, як і у випадку з іншими платформами розумного будинку, Alexa залежить від наявності стабільного інтернет-з'єднання для оптимального функціонування. Хмарна архітектура також забезпечує постійні оновлення системи, що дозволяє розширювати функціональні можливості без необхідності оновлення програмного забезпечення на пристроях. Такий підхід забезпечує гнучкість і актуальність функцій, що пропонуються Amazon Alexa, а також можливість безперервного розвитку системи відповідно до потреб ринку та користувачів.

2.1.3 Apple HomeKit

Apple HomeKit — це комплексна екосистема для розумного будинку, що дозволяє користувачам інтегрувати різноманітні пристрої в єдину систему управління через екосистему Apple. Однією з головних функцій HomeKit є автоматизація освітлення та регулювання клімату за допомогою розумних ламп, термостатів та інших пристроїв. Це дозволяє не лише створювати комфортні умови у приміщенні, але й оптимізувати витрати на енергію завдяки можливості налаштовувати графіки та режими роботи відповідно до щоденних потреб.

Apple HomeKit також надає комплексний контроль за безпекою житлових приміщень через інтеграцію з камерами, датчиками руху та електронними замками [41]. Завдяки цьому користувачі можуть контролювати доступ до свого будинку та отримувати сповіщення про будь-яку підозрілу активність.

Функція автоматизації в HomeKit дозволяє створювати так звані сцени, які активують кілька пристроїв одночасно, базуючись на певних умовах, таких як прихід чи вихід користувача з дому, час доби або зміни навколишніх умов (наприклад, температура чи рівень освітлення). Сцени дозволяють

користувачам адаптувати систему до власних звичок, роблячи її більш інтуїтивно зрозумілою та комфортною у повсякденному використанні.

Додатково HomeKit підтримує інтеграцію з голосовим помічником Siri, що забезпечує голосове управління системою. Це дозволяє користувачам керувати освітленням, кліматом, безпекою та іншими аспектами розумного будинку лише за допомогою голосових команд.

Apple HomeKit сумісний з великою кількістю пристроїв, сертифікованих за стандартом "Works with Apple HomeKit". Це включає продукти таких відомих брендів, як Philips Hue, Ecobee, August, Logitech та інших. HomeKit підтримує протоколи зв'язку Wi-Fi і Bluetooth, а також сучасний стандарт Thread, що покращує сумісність і підвищує продуктивність мережі розумного будинку.

Однак, оскільки для інтеграції в HomeKit пристрої мають відповідати стандартам Apple, це може обмежити вибір деяких пристроїв, порівняно з більш відкритими платформами, такими як Google Nest та Amazon Alexa. Перевага такої сертифікації полягає у високих стандартах безпеки та надійності, які Apple застосовує до сертифікованих пристроїв, що підвищує довіру користувачів до системи.

Apple HomeKit надає користувачам широкі можливості для налаштування через додаток Home на пристроях iOS. Завдяки цьому додатку можна створювати сцени для одночасної активації кількох пристроїв, а також налаштовувати автоматизацію на основі різних параметрів: місцезнаходження користувача, час доби або зміни навколишніх умов. Це дозволяє створити персоналізовану екосистему, яка адаптується під ритм життя користувача.

Інтеграція з Siri надає можливість голосового управління, що спрощує взаємодію з системою, оскільки команди можна задавати просто за допомогою голосу. Завдяки інтуїтивному інтерфейсу та підтримці голосових команд Apple HomeKit пропонує зручний спосіб налаштування та управління розумним будинком відповідно до індивідуальних потреб.

Архітектура Apple HomeKit має одну з ключових переваг — підтримку локальної обробки даних, що значно відрізняє її від інших платформ, таких як

Google Nest та Amazon Alexa. В той час як певні функції все ж використовують хмарні сервіси, велика частина обробки даних виконується локально на пристроях користувача, наприклад, на iPhone або HomePod. Це знижує залежність від стабільного інтернет-з'єднання, підвищує швидкість реакції системи на події та зменшує затримки у виконанні команд.

Локальна обробка також підвищує рівень конфіденційності та безпеки даних, оскільки інформація не завжди надсилається на сервери Apple. Це важливий аспект для користувачів, які прагнуть зберегти приватність своїх даних і мінімізувати ризики зовнішнього втручання в роботу системи. Хоча хмарна архітектура надає додаткові можливості для віддаленого управління та зберігання налаштувань, саме локальна обробка в HomeKit робить систему більш автономною і надійною у використанні.

2.2 Порівняльний аналіз локальної та хмарної обробки даних

Метою цього підрозділу є здійснення глибокого порівняльного аналізу локальної та хмарної обробки даних. Порівняння охоплює ключові аспекти: безпеку, швидкодію, надійність, вартість і масштабованість. Такий аналіз дозволяє визначити оптимальні рішення для розумних будинків залежно від конкретних вимог та обмежень користувачів.

З розвитком технологій Інтернету речей (IoT) вибір між локальною та хмарною обробкою даних став важливою дилемою для власників розумних будинків і розробників таких систем. Сучасні системи розумного будинку все більше покладаються на інтелектуальні сенсори та пристрої, що генерують значні обсяги даних, які потребують оперативної обробки та надійного зберігання. І локальна, і хмарна обробка мають свої переваги й обмеження, і ефективність кожного підходу залежить від конкретних характеристик та умов використання. Метою цього аналізу є визначення того, як різні підходи до обробки даних можуть задовольнити потреби користувачів стосовно безпеки, швидкодії, надійності, вартості та гнучкості систем.

Критерії порівняння:

1. Безпека та конфіденційність

Безпека та конфіденційність є критичними аспектами при виборі способу обробки даних у розумних будинках. Локальна обробка надає можливість зберігати дані всередині будинку, що значно знижує ризики несанкціонованого доступу чи витоку інформації під час передачі. Таким чином, користувач має максимальний контроль над своїми даними. Проте локальні рішення вимагають від користувачів підтримувати фізичну та програмну безпеку обладнання самостійно, що може бути викликом для людей без спеціальних технічних знань.

Хмарна обробка забезпечує централізовані механізми захисту, включаючи шифрування та багаторівневу автентифікацію. Однак зберігання даних на віддалених серверах підвищує ймовірність компрометації через кібератаки. Також конфіденційність даних залежить від політики компанії-провайдера, і правові аспекти можуть стати додатковим фактором ризику, особливо якщо провайдер зберігає дані в іншій юрисдикції.

2. Швидкодія та затримка

Швидкодія та затримка є вирішальними критеріями для систем розумного будинку, які потребують негайної реакції, наприклад, систем безпеки чи управління кліматом. Локальна обробка забезпечує обчислення даних безпосередньо на місці, що мінімізує затримки та забезпечує швидку реакцію. Це дозволяє системі працювати майже в режимі реального часу, що особливо важливо для критичних сценаріїв.

Хмарна обробка, навпаки, передбачає передачу даних на віддалені сервери, що може спричиняти затримки, особливо у випадках нестабільного інтернет-з'єднання або надмірного навантаження на мережу. Такі затримки можуть значно знижувати ефективність критичних систем, що робить локальну обробку переважним варіантом для застосувань, де важлива швидка реакція.

3. Надійність та доступність

Локальна обробка характеризується високою надійністю, оскільки система не залежить від інтернет-з'єднання. Це дозволяє розумним будинкам функціонувати безперервно навіть у випадку втрати зв'язку з мережею. Такий підхід особливо цінний для життєво важливих систем, таких як охоронні механізми, освітлення чи клімат-контроль, які повинні функціонувати незалежно від зовнішніх умов.

Хмарні рішення, зі свого боку, пропонують перевагу у вигляді автоматичного резервного копіювання даних і можливості швидкого відновлення після збою, що підвищує загальну стійкість системи до втрат даних. Однак хмарні сервіси залежать від надійного інтернет-з'єднання, і будь-які збої в мережі чи на стороні провайдера можуть впливати на доступність сервісу, що створює ризики для безперебійної роботи системи.

4. Вартість впровадження та експлуатації

Локальна обробка потребує значних початкових витрат на обладнання, зокрема на контролери, сервери та відповідні сенсори. Однак операційні витрати є відносно низькими, оскільки після початкового впровадження користувач не несе регулярних абонентських платежів. Локальні системи стають більш економічно вигідними у довгостроковій перспективі, якщо не враховувати потенційні витрати на обслуговування та модернізацію.

Хмарна обробка має низькі початкові витрати, оскільки не вимагає закупівлі дорогого обладнання, але наявні постійні витрати на абонентське обслуговування та використання ресурсів. З часом ці витрати можуть зрости, особливо при зростанні обсягу оброблюваних даних чи кількості підключених пристроїв. Отже, фінансова ефективність хмарної обробки значною мірою залежить від характеру та масштабу використання.

5. Масштабованість та гнучкість

Хмарна обробка має перевагу в масштабованості, оскільки дозволяє легко збільшувати обчислювальні ресурси у відповідь на зростання кількості даних або пристроїв, що підключаються. Це особливо важливо для динамічних систем, де потреби змінюються з часом. Хмарні провайдери забезпечують

доступ до майже необмежених ресурсів, що дозволяє адаптувати систему до будь-яких змінних вимог користувача.

Локальна обробка обмежена потужністю наявного обладнання, і для її масштабування можуть знадобитися значні інвестиції, включаючи придбання нових контролерів або серверів. Проте локальні рішення забезпечують більший контроль над інфраструктурою, що робить їх привабливими для користувачів, які прагнуть зберегти автономію та знизити залежність від сторонніх постачальників послуг.

Вибір між локальною та хмарною обробкою даних у системах розумного будинку залежить від багатьох факторів, таких як вимоги до безпеки, затримки, надійності, масштабованості та економічної ефективності. У випадках, коли критичною є низька затримка, локальна обробка даних виступає оптимальним рішенням, забезпечуючи мінімальні часові затримки та високий рівень автономності системи. Цей підхід ідеально підходить для застосувань, де швидкість реакції є вирішальним фактором, наприклад, для безпекових систем або управління критичними функціями житлового середовища.

Навпаки, у випадках, коли ключовими вимогами є гнучкість і масштабованість, хмарні технології надають суттєві переваги завдяки можливості обробляти значні обсяги даних, забезпечуючи централізоване управління та спрощуючи доступ до ресурсів. Хмарні рішення є також більш економічно ефективними для систем, які потребують зменшення початкових інвестицій, оскільки вони не вимагають придбання та підтримки складної локальної інфраструктури.

Особливу увагу слід приділити конфіденційності даних: для систем, які працюють із чутливими або персональними даними, локальна обробка забезпечує більший контроль і захист інформації, оскільки дані не передаються через мережу до віддалених серверів [42]. Водночас хмарні рішення можуть мати перевагу для зниження операційних витрат і легкості управління, особливо коли йдеться про неглибоку автоматизацію або додаткові сервіси, такі як аналітика або прогнозування.

Оптимальний підхід, таким чином, визначається через баланс специфічних вимог користувачів та особливостей середовища, де система буде застосовуватися. Іноді найбільш ефективним рішенням може стати гібридна архітектура, яка комбінує переваги обох підходів — локальної обробки для швидких реакцій і хмарної інфраструктури для резервного копіювання та аналізу великих даних.

Результати проведеного порівняльного аналізу локальної та хмарної обробки даних створюють основу для подальших досліджень, спрямованих на оптимізацію конкретних застосувань розумних будинків. Наприклад, висновки можуть вказувати на необхідність впровадження комбінованого підходу, який поєднує локальні обчислення для забезпечення миттєвої реакції системи, а також хмарні сервіси для довготривалого зберігання даних та аналітики.

Крім того, аналіз може виявити необхідність інвестування в конкретні компоненти системи для підвищення їхньої надійності та продуктивності. Наприклад, у випадках, коли локальні ресурси є обмеженими, впровадження інтелектуальних алгоритмів оптимізації може забезпечити баланс між ефективністю та витратами на енергію. Дослідження подальшої інтеграції алгоритмів машинного навчання дозволить визначити, які завдання найкраще залишати на локальному рівні, а які ефективніше передавати для хмарної обробки. Це сприятиме підвищенню гнучкості систем, їхньої адаптивності до умов середовища та зниженню загальних витрат на інфраструктуру.

2.3 Проблеми та виклики в реалізації автономних систем

Реалізація автономних систем передбачає подолання низки технічних викликів, серед яких особливе місце займають забезпечення сумісності пристроїв різних виробників та ефективне управління ресурсами. Ці аспекти є критично важливими для створення інтегрованих, надійних та високопродуктивних систем, здатних виконувати складні завдання в умовах динамічного середовища.

Одним із основних викликів у забезпеченні сумісності є велика різноманітність протоколів зв'язку та стандартів, що використовуються різними виробниками пристроїв [43]. Ця різноманітність ускладнює інтеграцію пристроїв у єдину автономну систему, оскільки кожен виробник може використовувати власні специфікації для передачі даних, управління пристроями та обробки інформації. Наприклад, в середовищі Інтернету речей (IoT) існують різні протоколи, такі як MQTT, CoAP, Zigbee, що потребують спеціальних шлюзів або адаптерів для забезпечення взаємодії між пристроями.

Відсутність уніфікованих стандартів у галузі автономних систем призводить до проблем зі сумісністю та обміну даними між різними компонентами системи. Це особливо актуально для великих систем, де інтеграція численних пристроїв від різних виробників стає складною задачею. Для подолання цього бар'єру необхідно розробляти та впроваджувати загальноприйняті стандарти, які забезпечують узгодженість протоколів зв'язку, форматів даних та методів управління пристроями.

Іншим важливим аспектом забезпечення сумісності є розробка та впровадження API (Application Programming Interface) та SDK (Software Development Kit), які дозволяють інтегрувати пристрої різних виробників у єдину систему [44]. API забезпечують стандартизований інтерфейс для взаємодії між компонентами системи, спрощуючи процес інтеграції та знижуючи залежність від конкретних виробників. SDK, в свою чергу, надають набір інструментів для розробників, що дозволяють створювати додатки та сервіси, які можуть безперешкодно працювати з різними пристроями.

Реальні приклади інтеграції пристроїв різних виробників демонструють важливість вирішення проблем сумісності. Наприклад, у сфері розумного будинку інтеграція систем безпеки, освітлення, опалення та інших пристроїв від різних виробників вимагає узгодженості протоколів та стандартів. Використання платформи, що підтримує мультипротокольную взаємодію, дозволяє створити єдину систему управління, яка забезпечує взаємодію між різними компонентами без необхідності використання численних адаптерів.

Автономні системи повинні ефективно використовувати обчислювальні ресурси для забезпечення швидкої обробки даних та виконання завдань в реальному часі. Це вимагає впровадження оптимізованих алгоритмів обробки даних, розподілу навантаження та управління обчислювальними потужностями. Наприклад, використання розподілених обчислень та паралельних алгоритмів дозволяє підвищити продуктивність і знизити затримки у виконанні завдань.

Ці системи часто залежать від безперебійного та швидкого обміну даними між різними компонентами. Ефективне управління комунікаційними ресурсами включає оптимізацію пропускну здатності мережі, мінімізацію затримок та забезпечення надійності зв'язку. Це може бути досягнуто за допомогою використання пріоритетних каналів зв'язку, адаптивних протоколів передачі даних та механізмів забезпечення якості обслуговування.

В багатьох випадках застосовуються комбінації локальних обчислювальних ресурсів та хмарних сервісів для забезпечення гнучкості та масштабованості. Баланс між цими ресурсами визначає продуктивність, здатність до масштабування та витрати на обслуговування. Використання хмарних обчислень дозволяє динамічно збільшувати або зменшувати обчислювальні потужності в залежності від поточних потреб, забезпечуючи оптимальне використання ресурсів та зниження витрат [45].

Ефективне управління ресурсами вимагає прогнозування майбутніх потреб системи та планування їхнього використання. Це включає аналіз історичних даних, моделювання майбутніх сценаріїв та впровадження механізмів автоматизованого управління ресурсами. Наприклад, використання машинного навчання для прогнозування навантаження на систему дозволяє динамічно коригувати розподіл ресурсів, забезпечуючи стабільну роботу системи навіть у випадку пікових навантажень.

Технічні виклики, пов'язані із забезпеченням сумісності пристроїв різних виробників та управлінням ресурсами, є суттєвими бар'єрами на шляху до успішної реалізації автономних систем. Вирішення цих проблем вимагає комплексного підходу, що включає стандартизацію протоколів зв'язку,

використання проміжного програмного забезпечення, оптимізацію обчислювальних та енергетичних ресурсів, а також впровадження сучасних інструментів та технологій управління ресурсами. Подолання технічних викликів дозволяє створити інтегровані, надійні та високопродуктивні автономні системи, здатні ефективно функціонувати в різних сферах діяльності та адаптуватися до змінних умов середовища.

3. РОЗРОБКА АРХІТЕКТУРИ АВТОНОМНОЇ СИСТЕМИ РОЗУМНОГО БУДИНКУ

3.1 Вибір апаратних компонентів та сенсорів

У цьому розділі детально описано розробку архітектури автономної системи розумного будинку, яка забезпечує безпеку та комфорт мешканців. Розглянуто вибір апаратних компонентів та сенсорів, проектування програмного забезпечення для локальної обробки даних, а також інтеграцію компонентів та протоколів зв'язку.

Вибір апаратних компонентів є критичним етапом у розробці системи, оскільки від цього залежить її функціональність, надійність та ефективність [46]. Правильний підбір компонентів забезпечує не лише виконання основних завдань системи, але й її здатність адаптуватися до змінних умов експлуатації та вимог користувачів. У процесі вибору було враховано низку ключових факторів, серед яких сумісність компонентів, енергоефективність, масштабованість, вартість та можливість подальшого розширення системи.

Перед початком вибору конкретних компонентів було проведено аналіз вимог до системи, визначено її основні функціональні блоки та взаємозв'язки між ними. Особливу увагу приділено питанням інтеграції різних сенсорів та виконавчих пристроїв, а також забезпеченню надійного та безпечного зв'язку між ними. Крім того, було враховано умови експлуатації системи, такі як можливі коливання температури, потреба у резервному живленні та необхідність захисту даних від несанкціонованого доступу.

Основними критеріями вибору були:

–Сумісність компонентів: Забезпечення безперебійної роботи всіх елементів системи без необхідності додаткової адаптації чи використання перехідників.

–Енергоефективність: Мінімізація споживання енергії для подовження терміну служби автономних систем та зменшення експлуатаційних витрат.

–Масштабованість: Можливість легкої розширюваності системи шляхом додавання нових сенсорів або виконавчих пристроїв без значних змін у її

архітектурі.

–Вартість: Оптимізація витрат на компоненти без компрометації якості та функціональності системи.

Наступним кроком стало детальне розглядання окремих компонентів, включаючи мікроконтролери, сенсори, виконавчі пристрої та комунікаційні модулі, з урахуванням зазначених критеріїв. Кожен компонент був проаналізований щодо його технічних характеристик, сумісності з іншими елементами системи, а також можливості інтеграції у загальну архітектуру.

Також розглянуто аспекти живлення та енергоефективності системи, підтримку сучасних IoT протоколів зв'язку, безпеку даних, точність сенсорів та можливості оновлення прошивки, що забезпечують надійність та довговічність роботи системи в умовах реального використання.



Рисунок 3.1 – Блок схема розумного будинку

Основою системи є мікроконтролер, який координує роботу всіх підключених пристроїв та забезпечує обробку даних. Для нашої системи було обрано Arduino Portenta H7 з наступних причин:

- Доступність та простота використання: Arduino широко використовується як у навчальних, так і в професійних проектах завдяки зручному середовищу програмування та великій кількості бібліотек.

- Розширення через модулі: Arduino Portenta H7 підтримує Wi-Fi та Ethernet для створення надійного та стабільного веб-сервера, що дозволяє реалізувати повноцінну інтеграцію розумного будинку в мережу.

- Оптимальне енергоспоживання: Arduino Portenta H7 підтримує ефективне управління енергоспоживанням, що дозволяє стабільно працювати в інтенсивному режимі без перегріву.

- Велика спільнота та документація: Завдяки великій спільноті розробників легко знайти приклади, вирішення проблем та рекомендації щодо розширення функціональності системи.

- Підтримка різних інтерфейсів: Arduino має вбудовану підтримку I2C, SPI, UART, що дозволяє підключати різноманітні сенсори та пристрої, забезпечуючи необхідну гнучкість.



Рисунок 3.2 – Arduino Portenta H7

Для забезпечення функціональності системи обрано такі сенсори:

–Датчик руху PIR (HC-SR501): Виявляє інфрачервоне випромінювання від теплових об'єктів, що дозволяє виявити присутність людини. Цей датчик є енергоефективним та має високу чутливість, що робить його ідеальним для систем безпеки.



Рисунок 3.3 – Датчик руху PIR (HC-SR501)

–Магнітоконттактний датчик (Reed Switch): Контролює відкриття та закриття дверей і вікон, забезпечуючи моніторинг доступу до приміщень. Вибір цього датчика обумовлений його надійністю та довговічністю при мінімальній потребі в обслуговуванні.



Рисунок 3.4 – Магнітоконттактний датчик

–Датчик диму та газу (MQ-2): Забезпечує раннє виявлення пожежі або витоку газу, що є критично важливим для забезпечення безпеки житлових і комерційних приміщень.



Рисунок 3.5 – Датчик диму та газу (MQ-2)

–Датчик температури та вологості (DHT22): Моніторить кліматичні умови в приміщенні, забезпечуючи можливість автоматичного регулювання систем опалення, вентиляції та кондиціонування повітря.

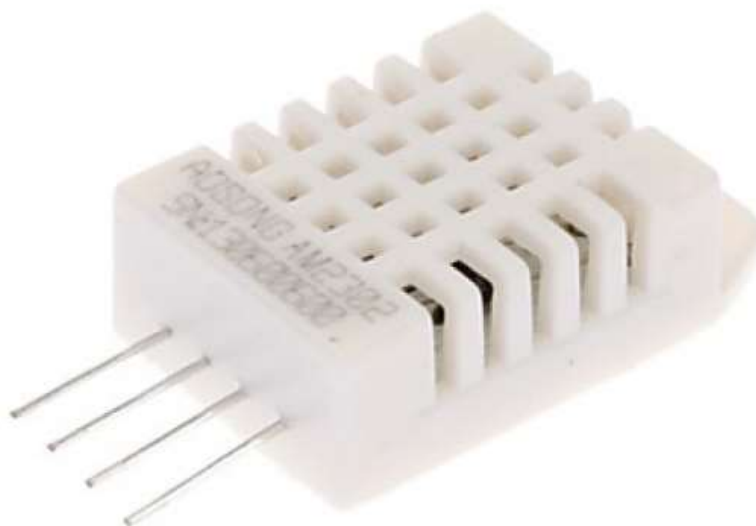


Рисунок 3.6 – Датчик температури та вологості (DHT22)

–Датчик освітленості (BH1750): Вимірює рівень освітлення для автоматичного керування освітленням, що сприяє підвищенню енергоефективності та комфорту користувачів.

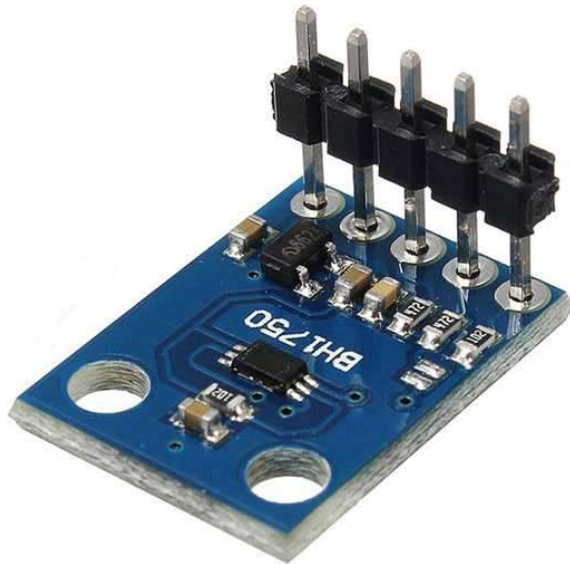


Рисунок 3.7 – Датчик освітленості (BH1750)

–Камера (ESP32-CAM): Забезпечує відеоспостереження та може використовуватися для розпізнавання облич, що підвищує рівень безпеки та дозволяє здійснювати віддалений моніторинг.

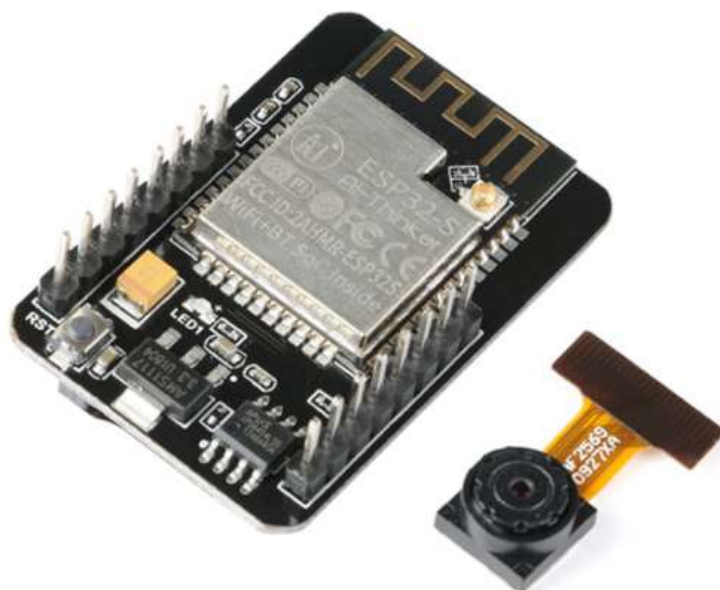


Рисунок 3.8 – Камера (ESP32-CAM)

Для реалізації автоматизованих функцій системи були обрані наступні виконавчі пристрої:

–Реле (Relay Module): Дозволяє керувати живленням побутових приладів та освітлення, забезпечуючи можливість автоматичного вмикання та вимикання електроприладів відповідно до заданих умов.



Рисунок 3.9 – Реле 1 канал 12 В

–Сирена: Активується у випадку тривоги для привернення уваги та попередження про небезпеку, сприяючи швидкому реагуванню користувачів.



Рисунок 3.10 – Сирена

–Сервоприводи: Використовуються для автоматизації механічних процесів, таких як відкриття штор або дверей, що підвищує комфорт та зручність використання системи.



Рисунок 3.11 – Сервопривод

Для забезпечення надійного та ефективного зв'язку між компонентами системи були вибрані наступні комунікаційні модулі:

–NRF24L01: Забезпечує бездротовий зв'язок між сенсорами та мікроконтролером на частоті 2.4 ГГц з низьким енергоспоживанням. Цей модуль є ідеальним для створення мережі сенсорів завдяки своїй високій швидкості передачі даних та стабільності з'єднання.

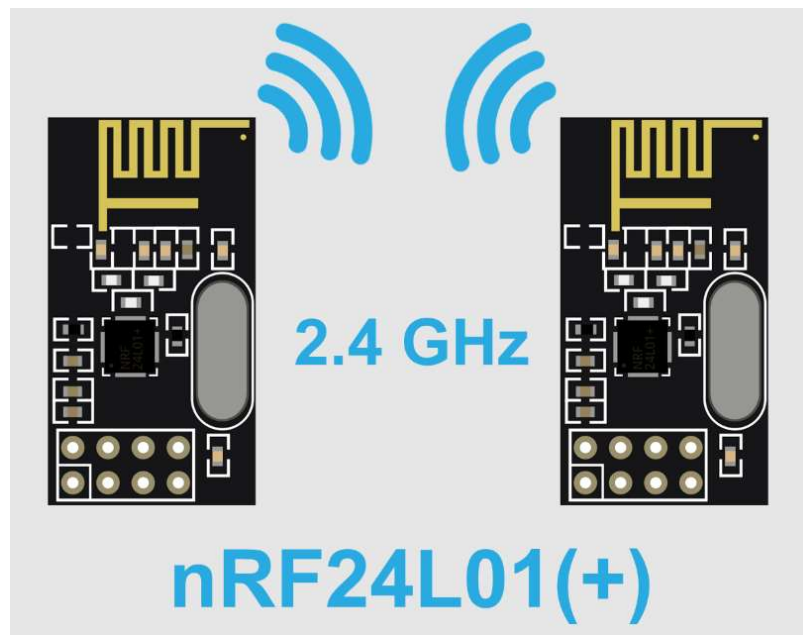


Рисунок 3.12 – Бездротовий модуль NRF24L01

–GSM-модуль (SIM800L): Використовується для відправки SMS-повідомлень у випадку відсутності Wi-Fi з'єднання, що забезпечує додатковий рівень надійності та можливість отримання повідомлень про події незалежно від наявності локальної мережі.

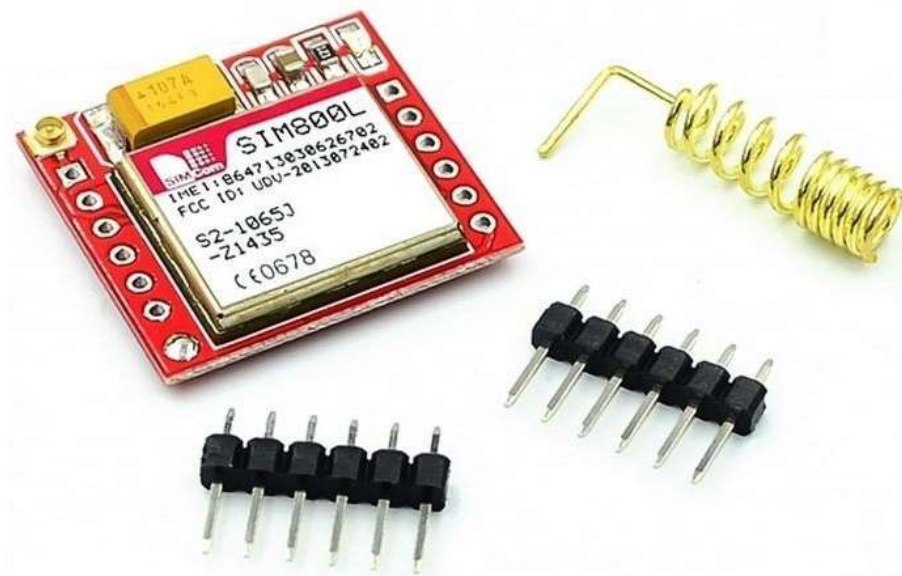


Рисунок 3.13 – GSM-модуль (SIM800L)

Перехідячи до аспектів живлення та енергоефективності, важливо забезпечити стабільну роботу системи навіть у випадках відсутності основного джерела живлення. Енергоефективність є ключовим фактором, особливо для автономних рішень.

Джерела живлення: Основним джерелом живлення для системи є стабільне джерело постійного струму (DC) з необхідними характеристиками напруги та струму для забезпечення роботи всіх компонентів. Для автономних систем використовуються акумулятори, які дозволяють системі працювати без підключення до мережі електропостачання.

Системи резервного живлення: Використання акумуляторів або безперебійних блоків живлення (UPS) забезпечує стабільну роботу системи під час відсутності основного живлення. Це особливо важливо для систем безпеки та моніторингу, де перебої в живленні можуть призвести до втрати важливої інформації або порушення роботи системи.

Система охолодження: У випадку роботи системи в умовах підвищеної температури необхідно забезпечити ефективне охолодження компонентів. Для цього можуть використовуватися активні системи охолодження, такі як вентилятори, або пасивні системи, наприклад, радіатори. Вибір системи охолодження залежить від конкретних умов експлуатації та вимог до температурного режиму компонентів.

Після забезпечення стабільного живлення, важливо врахувати можливість комунікації між компонентами системи та зовнішніми пристроями. Підтримка сучасних IoT протоколів зв'язку є ключовою для забезпечення інтеграції системи з іншими IoT-пристроями та її масштабованості.

Наступним важливим аспектом є забезпечення безпеки даних та захисту від несанкціонованого доступу. У IoT системах безпека інформації є пріоритетом.

Протоколи шифрування: Для захисту передачі даних через Wi-Fi або Bluetooth використовуються сучасні протоколи шифрування, такі як WPA3 для

бездротових мереж та AES для шифрування даних на рівні додатків. Це забезпечує конфіденційність та цілісність переданої інформації.

Аутентифікація та авторизація: Впровадження механізмів аутентифікації та авторизації користувачів та пристроїв дозволяє забезпечити доступ до системи лише авторизованим користувачам, що знижує ризик несанкціонованого доступу та потенційних атак.

Для забезпечення надійності та ефективності роботи системи критично важливою є точність вимірювань сенсорів. Цей аспект тісно пов'язаний з налаштуванням та підтримкою сенсорних компонентів у робочому стані.

Калібрування сенсорів: Під час початкової налаштування системи проводиться калібрування сенсорів для забезпечення точності вимірювань. Це включає встановлення базових параметрів та налаштування сенсорів відповідно до специфічних умов експлуатації.

Забезпечення актуальності програмного забезпечення є важливим аспектом для підтримки функціональності та безпеки системи. Регулярні оновлення дозволяють впроваджувати нові функції, виправляти помилки та закривати вразливості.

Оновлення прошивки через OTA (Over-the-Air): Можливість оновлення прошивки через бездротове з'єднання дозволяє легко та швидко впроваджувати нові функції, виправляти помилки та закривати вразливості без необхідності фізичного доступу до пристроїв. Це значно спрощує процес обслуговування та підтримки системи.

Таким чином, ретельний вибір апаратних компонентів та сенсорів, а також врахування аспектів живлення, енергоефективності, підтримки сучасних IoT протоколів, безпеки даних, точності сенсорів та можливостей оновлення прошивки забезпечують створення надійної, ефективної та масштабованої системи. Такий підхід дозволяє системі відповідати сучасним вимогам користувачів та умовам експлуатації, забезпечуючи високу якість та довговічність її роботи.

3.2 Проектування програмного забезпечення для локальної обробки

Проектування програмного забезпечення є ключовим етапом у розробці системи, оскільки воно визначає ефективність, надійність та можливість подальшого розширення системи. Локальна обробка даних дозволяє зменшити затримки у реагуванні на події, забезпечити більшу автономність системи та покращити безпеку даних. У цьому розділі детально розглянуто основні аспекти проектування програмного забезпечення, включаючи модульність і масштабованість, обробку даних у режимі реального часу, розширене керування виконавчими пристроями, обробку помилок та відновлення, інтерфейс для користувача та локальну взаємодію, а також моніторинг продуктивності та діагностику.

Одним із основних принципів проектування програмного забезпечення є модульність, яка забезпечує розбиття системи на окремі, взаємодіючі між собою компоненти. Це дозволяє не лише полегшити процес розробки, тестування та обслуговування системи, але й забезпечує гнучкість у її адаптації до нових вимог і умов експлуатації. Завдяки модульному підходу до структурування програмного забезпечення, можна ефективніше керувати ресурсами, уникати дублювання коду, а також створювати більш зрозумілу та логічно організовану систему. Крім того, модульність спрощує інтеграцію нових функціональних можливостей без значного впливу на вже існуючі частини системи, що є критично важливим у середовищах із динамічними вимогами.

Програмне забезпечення системи було розроблено з використанням модульної архітектури, де кожен модуль відповідає за виконання певних функціональних завдань. Основні модулі включають:

–Збір даних: Відповідає за отримання інформації від сенсорів та їх первинну обробку. Цей модуль забезпечує збирання максимально точних і достовірних даних із різних джерел, таких як сенсори температури, вологості, руху та інші, що є основою для подальшої обробки та аналізу. Його робота включає також базову

фільтрацію шумів, валідацію даних та їх підготовку до передачі в модуль аналізу.
(Додаток А)

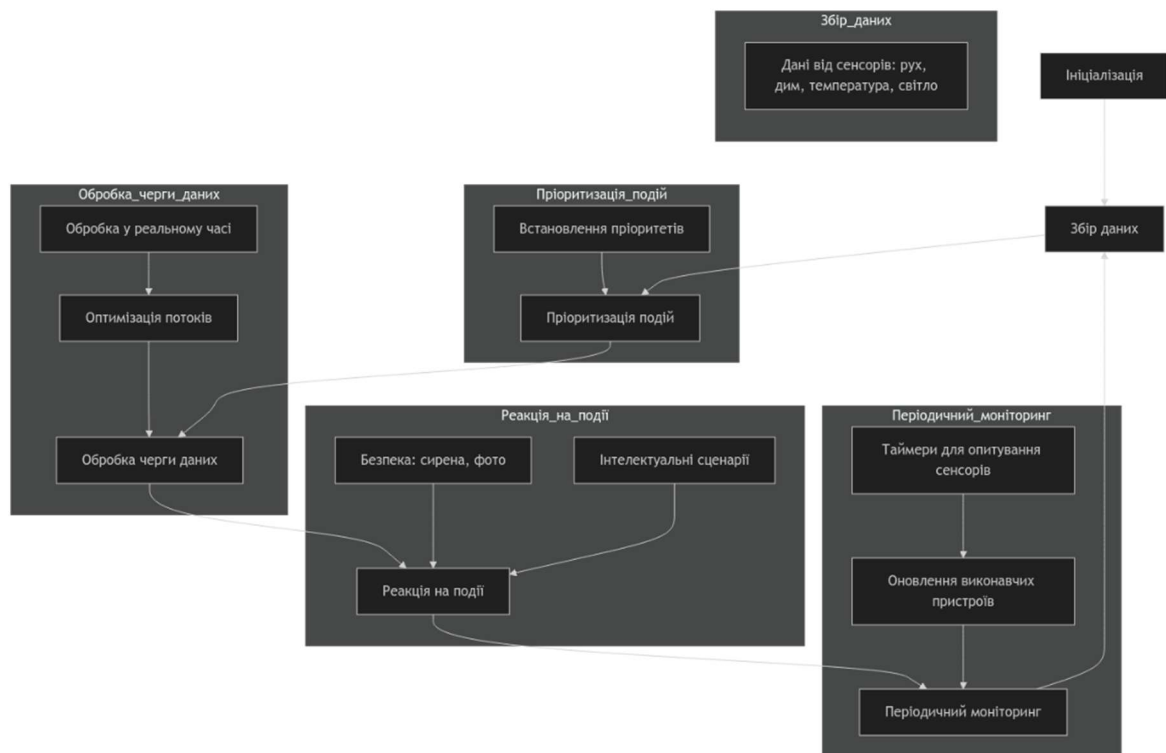


Рисунок 3.14 – Блок-схема модуля збору даних

–Аналіз: Здійснює обробку та інтерпретацію зібраних даних для виявлення подій або станів. Цей модуль реалізує алгоритми для ідентифікації ключових змін у середовищі, таких як перевищення заданих параметрів чи поява визначених сценаріїв. Завдяки цьому система може швидко реагувати на події, забезпечуючи її ефективність і точність. (Додаток Б)

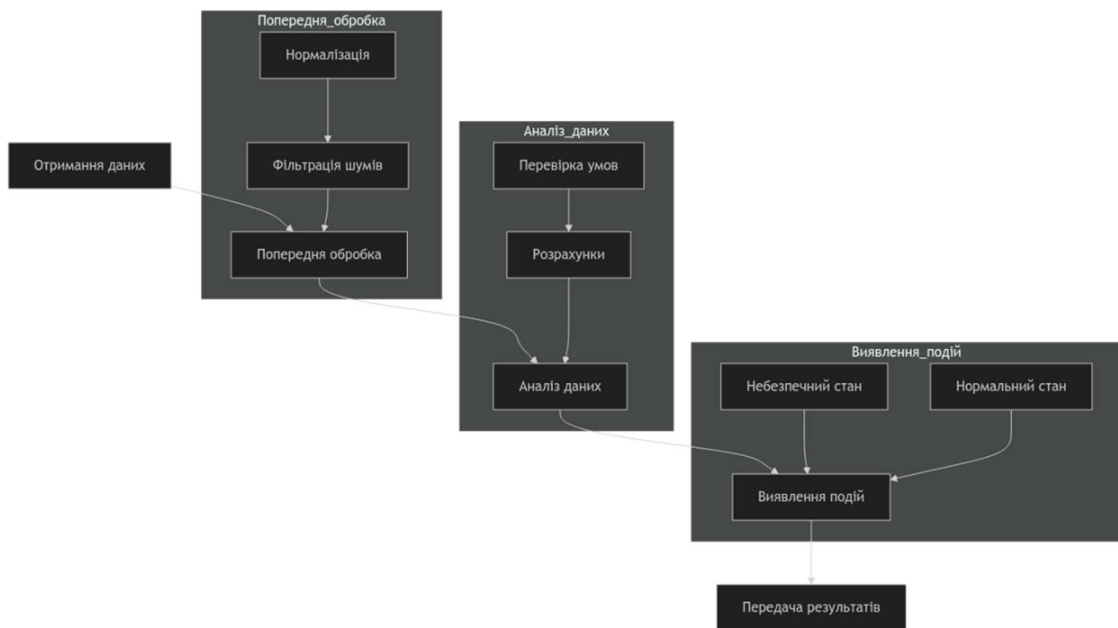


Рисунок 3.15 – Блок схема модуля аналізу

–Керування: Управляє виконавчими пристроями на основі результатів аналізу даних. Цей модуль реалізує функціональність, яка дозволяє автоматизувати процеси, включаючи активацію або деактивацію виконавчих пристроїв, таких як освітлення, опалення або сигналізація. Завдяки адаптивності цього модуля, можна забезпечити оптимальне управління ресурсами та мінімізувати людське втручання. (Додаток В)

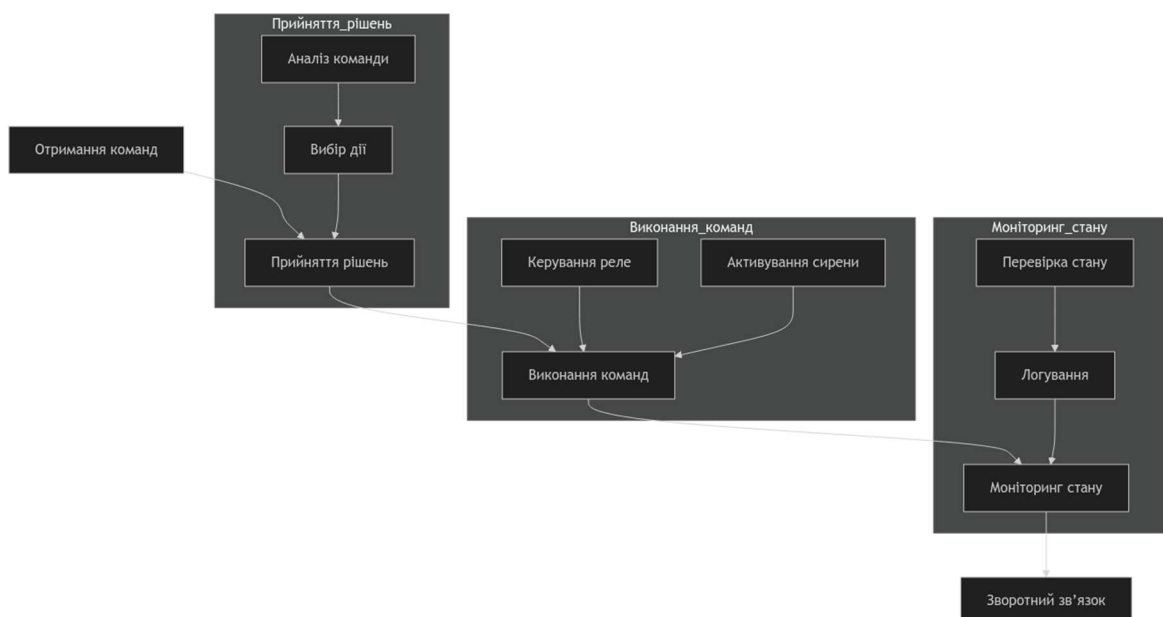


Рисунок 3.16 – Блок схема модуля керування

–Комунікація: Забезпечує обмін інформацією між різними частинами системи та зовнішніми мережами. Основна роль цього модуля полягає у створенні ефективної і безпечної мережі передачі даних, яка підтримує взаємодію між компонентами системи, а також дозволяє інтеграцію з хмарними платформами чи іншими зовнішніми сервісами. Це забезпечує масштабованість та зручність у використанні системи. (Додаток Г)

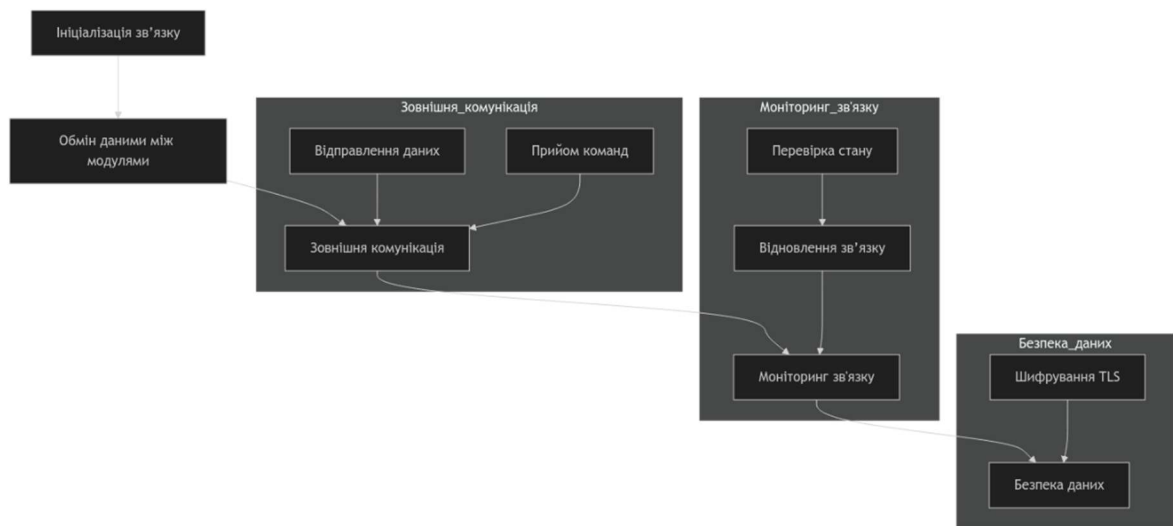


Рисунок 3.17 – Блок схема модуля комунікації

Таке розбиття на модулі дозволяє легко оновлювати окремі компоненти системи без необхідності внесення змін у всю архітектуру. Наприклад, додавання нового сенсора може бути реалізовано шляхом додавання відповідного підмодуля без впливу на інші частини програмного забезпечення.

Масштабованість програмного забезпечення передбачає можливість розширення системи шляхом додавання нових сенсорів або виконавчих пристроїв без значних змін у базовій програмній архітектурі. Це досягається завдяки використанню стандартних інтерфейсів та протоколів зв'язку, що дозволяє інтегрувати нові компоненти з мінімальними зусиллями. Наприклад, додавання нового типу сенсора потребує лише розробки відповідного драйвера та його інтеграції у модуль збору даних, без необхідності змін у модулях аналізу чи керування.

Для забезпечення своєчасної реакції на події система повинна мати можливість обробляти дані у режимі реального часу. Це досягається за рахунок ефективного управління потоками даних та використання механізмів оптимізації обробки. Реалізація такої функціональності досягається завдяки ефективному управлінню потоками даних, яке передбачає багатопотоковість та асинхронність. Це дозволяє зменшити затримки обробки запитів, особливо у сценаріях із високою частотою оновлення даних, наприклад, від датчиків руху або температури. Крім того, використання механізмів оптимізації обробки даних, таких як кешування, попереднє обчислення або застосування спеціалізованих алгоритмів стиснення, дозволяє значно зменшити навантаження на апаратні компоненти. Зокрема, розробка програмного забезпечення передбачає інтеграцію інструментів, які автоматично розподіляють ресурси залежно від пріоритетності задач, таких як обробка сигналів аварійних датчиків у порівнянні з менш важливими запитами, наприклад, зміна кольору освітлення.

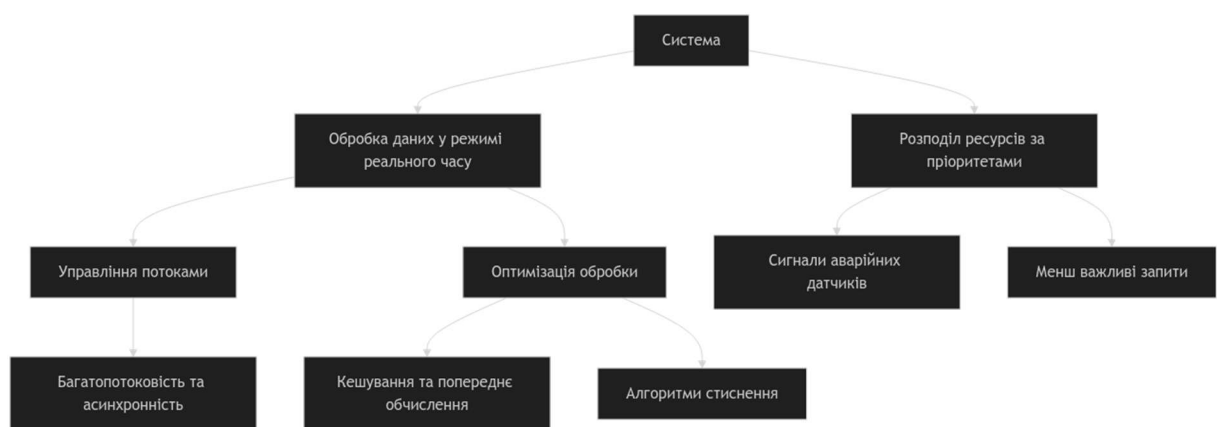


Рисунок 3.18 – Схема оптимізації потоків даних і ресурсів у системі

Для обробки потоків даних від різних сенсорів у реальному часі була впроваджена система черг обробки даних. Ця система є ключовим компонентом архітектури, що забезпечує структурований підхід до управління інформаційними потоками. Основна функція системи черг — забезпечення пріоритизації подій на основі їхньої важливості, що критично для мінімізації часу реакції на події, пов’язані із безпекою та комфортом.

Пріоритизація подій виконується за допомогою механізму тегів або категорій подій. Наприклад, події, які потенційно загрожують безпеці, такі як виявлення диму, витоку газу або несанкціонованого руху в приміщенні, автоматично отримують найвищий пріоритет у черзі. Це дозволяє системі негайно передавати дані на обробку відповідним модулям та ініціювати необхідні дії, наприклад, увімкнення сигналізації чи сповіщення користувачів. Менш критичні події, такі як зміна температури у приміщенні чи налаштування рівня освітлення, обробляються у другорядному порядку, що знижує навантаження на апаратну частину системи без шкоди для користувацького досвіду. Такий підхід забезпечує оптимальний баланс між швидкістю обробки даних та ефективністю використання ресурсів.



Рисунок 3.19 – Архітектура пріоритизації подій у реальному часі

Використання програмних таймерів є важливою частиною програмного забезпечення для локальної обробки даних, що забезпечує періодичне опитування сенсорів та виконання інших завдань з певним інтервалом часу. Цей механізм дозволяє системі здійснювати регулярний моніторинг стану різних компонентів і гарантує виконання необхідних дій без значних затримок, незалежно від загального навантаження на систему.

Програмні таймери є основою для автоматизації таких завдань, як збір даних з сенсорів, перевірка стану виконавчих пристроїв, оновлення внутрішніх параметрів системи та контроль навколишнього середовища. Наприклад, для сенсорів температури та вологості таймери можуть ініціювати періодичне зчитування даних кожні 10 секунд. Це дозволяє підтримувати актуальну

інформацію для аналізу стану середовища та прийняття відповідних рішень. Крім цього, таймери можуть використовуватися для забезпечення синхронізації між різними підсистемами. Наприклад, періодичний запуск завдання оновлення стану виконавчих пристроїв, таких як термостати або електричні замки, гарантує, що ці пристрої працюють у відповідності до останніх команд або налаштувань, навіть якщо деякі події пропущені через інтенсивне навантаження на систему. Для підвищення надійності програмного забезпечення передбачено адаптивний підхід до налаштування інтервалів таймерів залежно від поточного стану системи. Наприклад, якщо сенсори виявляють стабільні умови, система може автоматично збільшити інтервал опитування, зменшуючи навантаження на процесор. У критичних ситуаціях, таких як виявлення швидких змін параметрів середовища, інтервал опитування може бути зменшений для забезпечення точнішого моніторингу. Також важливим аспектом є інтеграція таймерів із системою обробки подій. Це дозволяє запускати додаткові завдання у визначені моменти часу, наприклад, регулярне резервне копіювання конфігураційних даних або оновлення програмного забезпечення.

Використання програмних таймерів у системі розумного будинку сприяє оптимізації ресурсів та забезпечує плавну і стабільну роботу всіх компонентів. Завдяки цьому система може ефективно реагувати на змінні умови середовища та виконувати завдання у встановлений час, що є критично важливим для автономної роботи.



Рисунок 3.20 – Архітектура програмних таймерів для розумного будинку

Для забезпечення високої гнучкості та адаптивності системи було реалізовано розширене керування виконавчими пристроями, яке включає інтелектуальні сценарії та прогнозування подій. Цей підхід дозволяє значно розширити функціональність системи та підвищити її здатність адаптуватися до змін у середовищі в режимі реального часу.

Інтелектуальні сценарії підтримують створення складних умовних логік, які базуються на аналізі даних з кількох сенсорів одночасно. Наприклад, система може приймати рішення на основі комбінованого аналізу даних з датчиків руху, освітленості та температури. У випадку виявлення присутності людини у приміщенні з недостатнім освітленням, система автоматично вмикає світло. Однак, якщо виявлено, що приміщення вже має оптимальну температуру, система утримується від увімкнення клімат-контролю, зберігаючи енергію.

Додатково система підтримує адаптивні сценарії, які можуть змінювати свою поведінку залежно від історичних даних або поточного контексту. Наприклад, аналіз попередніх шаблонів поведінки мешканців може допомогти системі прогнозувати дії. Якщо мешканець зазвичай перебуває вдома у певний час, система може автоматично регулювати температуру до комфортного рівня перед його приходом, а також вмикати освітлення у відповідних зонах будинку.

Для реалізації цих сценаріїв у системі передбачено інтуїтивно зрозумілий інтерфейс для користувача, який дозволяє налаштовувати складні логічні зв'язки між сенсорами та виконавчими пристроями без необхідності в програмуванні. Наприклад, користувач може створити сценарій, який активується при одночасному виконанні кількох умов, таких як підвищення рівня вологості та відсутність людей у приміщенні, для автоматичного запуску вентиляційної системи.

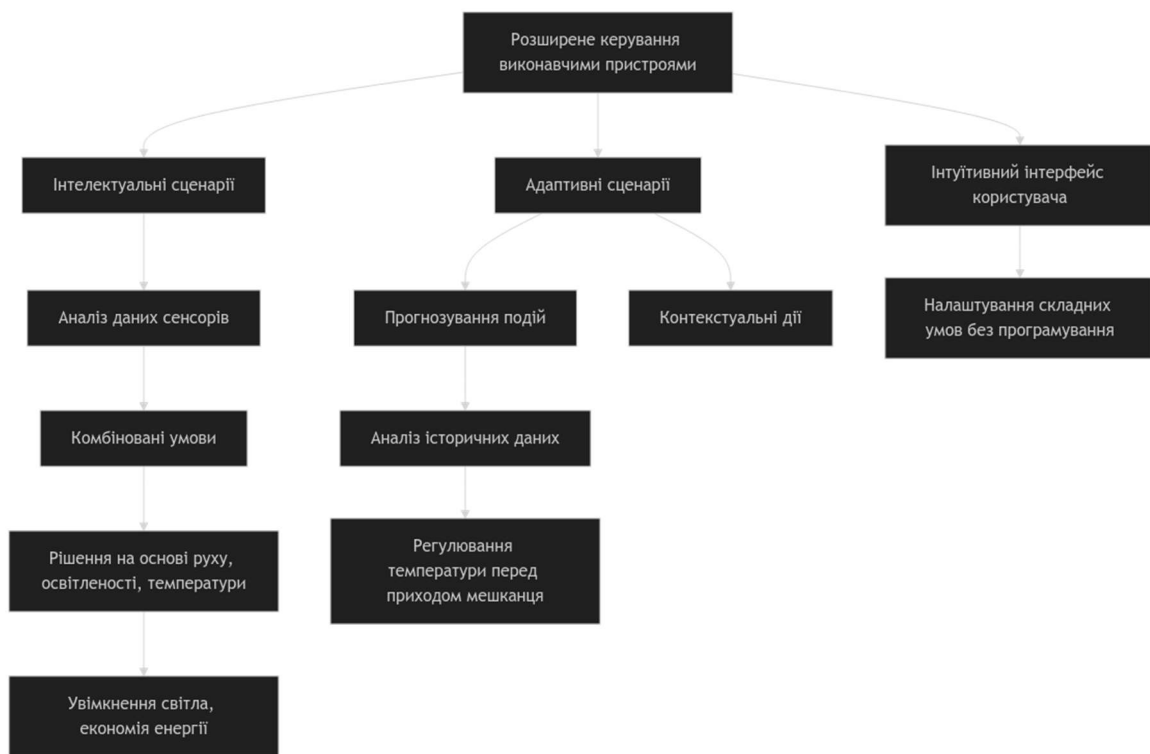


Рисунок 3.21 – Модель адаптивного керування розумного будинку

Система реалізує механізми перевірки стану сенсорів та виконавчих пристроїв для виявлення можливих помилок, таких як несправність сенсора, втрати зв'язку з пристроєм або відхилення в роботі виконавчих механізмів. Це критично важливо для забезпечення надійності та безпеки роботи системи розумного будинку, адже своєчасне виявлення аномалій дозволяє уникнути серйозних наслідків, таких як неспрацювання сигналізації або відсутність регулювання температури.

Для цього система здійснює постійний моніторинг статусу кожного компонента через регулярні сигнали "heartbeat" (сигнали життєздатності) від сенсорів та пристроїв. Ці сигнали дозволяють перевіряти зв'язок між центральним контролером і підключеними модулями. Якщо сигнал не надходить у встановлений час, система позначає пристрій як потенційно несправний. Крім перевірки зв'язку, система порівнює поточний стан кожного компонента з очікуваним. Наприклад, якщо система отримує показання, що значно відрізняються від попередніх або вказують на некоректну поведінку

(наприклад, сенсор температури передає однакове значення протягом тривалого часу, що нетипово для змінних умов), це розцінюється як аномалія.

У випадку виявлення несправності система автоматично генерує сповіщення, які можуть надсилатися користувачеві через різні канали, такі як мобільний додаток, електронна пошта або SMS. Повідомлення містить деталі про тип проблеми, ідентифікатор пристрою та рекомендації щодо подальшої діагностики чи ремонту. Для підвищення точності виявлення помилок система також може використовувати алгоритми аналізу історичних даних для ідентифікації патернів, що свідчать про можливі несправності. Наприклад, постійне зниження рівня сигналу бездротового зв'язку від сенсора може свідчити про деградацію батареї, а різкі стрибки у даних — про фізичне пошкодження сенсора.



Рисунок 3.22 – Алгоритми забезпечення надійності та безпеки системи

Таким чином, впровадження механізмів моніторингу стану компонентів сприяє підвищенню надійності, безпеки та ефективності роботи системи розумного будинку, роблячи її більш стійкою до непередбачуваних ситуацій.

Для забезпечення безперебійної роботи системи було впроваджено алгоритми автоматичного відновлення після виявлення помилок. Наприклад, у

випадку втрати зв'язку з сенсором, система може автоматично спробувати перепідключитися до нього або перезавантажити мікроконтролер. Такі заходи дозволяють мінімізувати час простою та забезпечити стабільність роботи системи навіть у випадках непередбачених збоїв.

Для забезпечення зручності використання системи важливо надати користувачам інтуїтивно зрозумілий інтерфейс для управління та моніторингу. У цьому контексті було розроблено локальний інтерфейс та реалізовано підтримку голосових команд.

У випадку відсутності зв'язку з Інтернетом система забезпечує можливість локального управління через веб-інтерфейс або мобільний додаток через локальну мережу (Wi-Fi). Це дозволяє користувачам здійснювати контроль над системою, переглядати стан сенсорів та керувати виконавчими пристроями навіть без доступу до зовнішніх мереж. Локальний інтерфейс реалізований з урахуванням принципів зручності використання та безпеки, забезпечуючи доступ лише авторизованим користувачам.

Для забезпечення ефективної роботи та своєчасного виявлення проблем у системі було впроваджено механізми моніторингу продуктивності та діагностики.

Система збору журналів (логування) дозволяє відстежувати події, стан сенсорів, помилки та енерговитрати. Це забезпечує можливість здійснювати діагностику системи як у режимі реального часу, так і при аналізі минулих подій. Логування допомагає виявляти патерни поведінки системи, а також швидко реагувати на виникаючі проблеми.

Таким чином, проектування програмного забезпечення для локальної обробки включає в себе ретельне планування модульної архітектури, забезпечення масштабованості, ефективної обробки даних у режимі реального часу, розширеного керування виконавчими пристроями, надійної обробки помилок та відновлення, зручного інтерфейсу для користувача, а також систем моніторингу та діагностики. Такий комплексний підхід дозволяє створити програмне забезпечення, яке відповідає сучасним вимогам до

функціональності, надійності та ефективності, забезпечуючи стабільну та довговічну роботу системи в умовах реального використання.

3.3 Інтеграція компонентів та протоколів зв'язку

Інтеграція апаратних та програмних компонентів, а також вибір відповідних протоколів зв'язку є вирішальними факторами для успішної роботи системи. Цей етап включає в себе не лише фізичне з'єднання різних компонентів, але й оптимізацію обміну даними, забезпечення безпеки та відмовостійкості, а також інтеграцію з іншими системами, такими як екосистема "розумного будинку". У цьому розділі детально розглянуто ключові аспекти інтеграції, включаючи обробку та оптимізацію пакетів даних, механізми відновлення після збоїв, забезпечення відмовостійкості, розширені методи безпеки, оптимізацію енергоспоживання, розширене логування та діагностику, а також інтеграцію з екосистемою "розумного будинку".

Інтеграція компонентів у системах розумного будинку базується на багаторівневій архітектурі, яка охоплює сенсори, виконавчі пристрої, локальні контролери, хмарні сервіси та користувацькі інтерфейси [47]. Сенсори отримують дані про стан середовища, передаючи їх локальним контролерам для обробки. Далі інформація може бути передана на хмарні платформи для зберігання, аналізу або взаємодії з іншими сервісами.

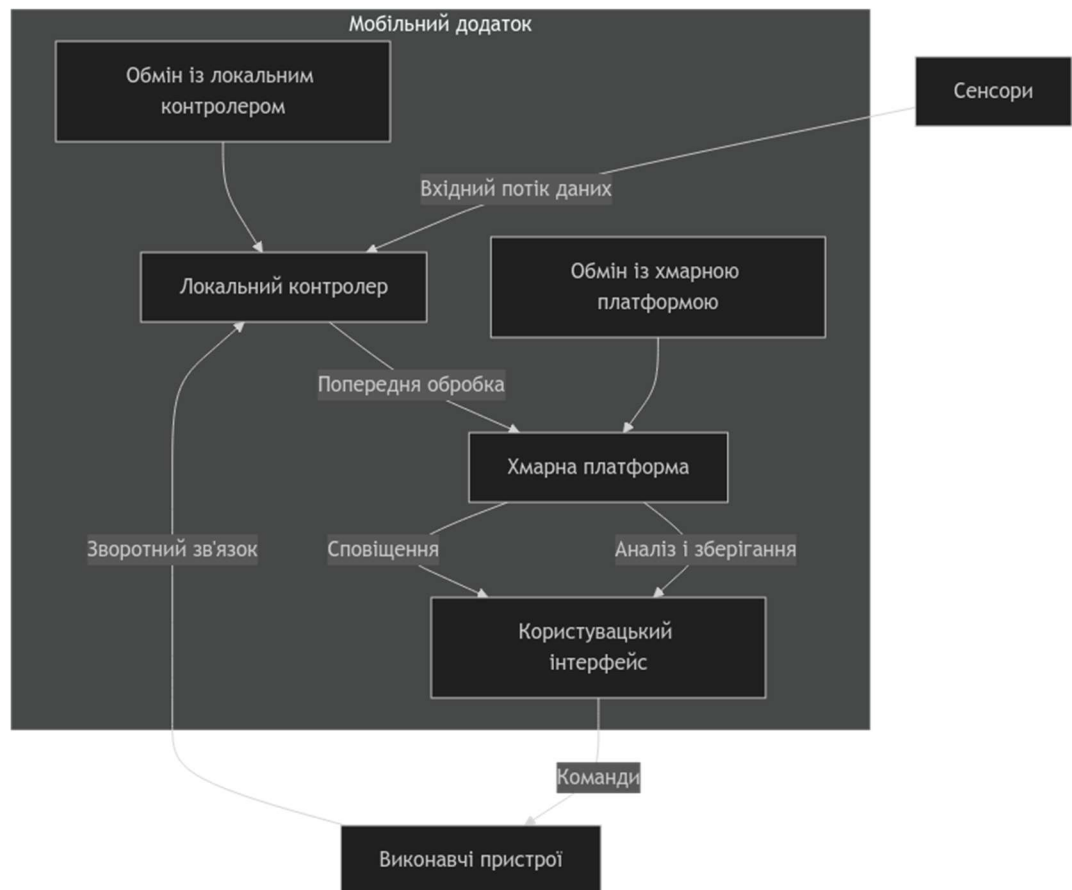


Рисунок 3.18 – Діаграма взаємодії компонентів

Діаграма взаємодії ілюструє основні потоки даних між компонентами, включаючи:

- Зв'язок сенсорів із локальними контролерами.
- Обмін інформацією між локальними системами та хмарною платформою.
- Інтерацію між мобільними додатками та іншими інтерфейсами користувача.

Локальні контролери забезпечують низьку затримку та оперативність виконання команд, тоді як хмарні сервіси дозволяють виконувати складні аналітичні задачі та забезпечують довготривале зберігання великих обсягів даних.

Для забезпечення ефективного обміну даними між компонентами використовуються кілька ключових протоколів. Кожен із них вибрано з урахуванням специфічних вимог системи:

–MQTT: Полегшений протокол, оптимізований для пристроїв із низьким енергоспоживанням. Він ідеально підходить для передачі даних між сенсорами та локальними контролерами, забезпечуючи енергоефективність і мінімальну затримку. Його низьке навантаження на мережу робить його популярним вибором для IoT-систем.

–HTTP: Використовується для інтеграції з хмарними платформами завдяки його сумісності з веб-додатками та сервісами. Простота використання та підтримка широкого спектра клієнтів роблять цей протокол універсальним рішенням для обміну даними.

–WebSocket: Забезпечує двосторонню комунікацію в реальному часі, що необхідно для інтерактивних функцій і швидкого реагування системи. Це ідеальний вибір для застосунків, де потрібна миттєва синхронізація між компонентами.

Інтеграція компонентів у системах розумного будинку базується на ретельно розроблених і стандартизованих інтерфейсах. API (Application Programming Interface) виступають посередниками, що забезпечують взаємодію між сенсорами, виконавчими пристроями, локальними контролерами, хмарними платформами та користувацькими додатками, забезпечуючи прозорий і ефективний обмін даними.

REST (Representational State Transfer) є найбільш поширеним підходом до побудови API завдяки своїй гнучкості, масштабованості та простоті використання. Основні переваги:

–Універсальність: REST API використовує стандартні HTTP-запити (GET, POST, PUT, DELETE), що робить його сумісним із більшістю веб-додатків та мобільних платформ.

–Масштабованість: Стандартизація запитів дозволяє легко інтегрувати нові функції системи, такі як управління пристроями або доступ до історичних даних.

–Сумісність із JSON: Підтримка JSON як основного формату передачі даних забезпечує простоту інтеграції з іншими сервісами.

REST API є ідеальним вибором для інтеграції з хмарними платформами, оскільки забезпечує надійний і безпечний зв'язок між додатками та серверами.

Локальні API відповідають за зв'язок між компонентами всередині локальної мережі [48]. Їх ключові особливості:

- Швидкодія: Завдяки прямій комунікації в межах локальної мережі локальні API забезпечують мінімальні затримки, що критично важливо для систем безпеки та управління кліматом.

- Оптимізація ресурсів: Використання ефективних протоколів, таких як MQTT чи CoAP (Constrained Application Protocol), дозволяє зменшити обсяг переданих даних та енергоспоживання пристроїв.

- Локальна автономність: Система залишається функціональною навіть у разі втрати інтернет-зв'язку.

Формати повідомлень є фундаментальним аспектом забезпечення сумісності та інтеграції компонентів у єдину систему. Використання стандартизованих форматів дозволяє зменшити час на розробку та впровадження нових пристроїв.

JSON є основним форматом для обміну даними завдяки своїй простоті та гнучкості. Його основні переваги:

- Легкість інтеграції: JSON підтримується всіма сучасними мовами програмування.

- Прозорість: Формат легко читається людьми, що спрощує налагодження та тестування.

- Гнучкість: JSON дозволяє динамічно змінювати структуру даних без впливу на існуючі функції.

Приклад JSON-повідомлення:

```
{"device_id": "sensor-001",  
  "status": "active",  
  "temperature": 23.4,  
  "humidity": 55}
```

Для передачі великих обсягів даних, таких як потокове відео чи аудіо, використовується двійковий формат. Його переваги:

- Висока швидкість: Зменшує навантаження на мережу, забезпечуючи швидку передачу даних.

- Компактність: Економить пропускну здатність мережі порівняно з текстовими форматами.

- Сумісність із мультимедійними даними: Ідеально підходить для обробки відеопотоків або графічних матеріалів.

Ефективність роботи інтегрованих систем залежить від якості та надійності інтерфейсів. Основні аспекти забезпечення інтеграції:

- Документованість: Ретельна документація API дозволяє розробникам швидко адаптуватися до вимог системи.

- Тестування сумісності: Регулярна перевірка взаємодії між компонентами гарантує стабільність системи.

- Версійність API: Забезпечення сумісності між новими та старими версіями інтерфейсів мінімізує ризики втрати функціональності.

Чітко розроблені API та стандартизовані формати повідомлень є критичними елементами успішної інтеграції компонентів, що забезпечують стабільну роботу систем розумного будинку.

3.4 Безпека та конфіденційність даних у локальних системах

З розвитком IoT-технологій питання забезпечення конфіденційності та захисту даних стає критично важливим. Зростаюча кількість вразливостей у локальних системах, таких як злом мереж, несанкціонований доступ та інші кіберзагрози, вимагає впровадження нових підходів до управління безпекою. Ця робота спрямована на створення рішень, які враховують сучасні виклики, забезпечуючи баланс між ефективністю системи та рівнем захисту [49].

Перелік загроз:

- Злом мережі: Використання вразливостей мережевої інфраструктури для

отримання несанкціонованого доступу.

- Несанкціонований доступ: Компрометація облікових даних або атаки на паролі.

- Шкідливе програмне забезпечення: Розповсюдження вірусів та інших шкідливих компонентів, що можуть впливати на стабільність системи.

- Перехоплення даних: Використання незахищених каналів зв'язку для несанкціонованого доступу до інформації.

- Внутрішні помилки: Ненавмисні дії користувачів або неправильне налаштування системи, що може призвести до компрометації даних.

Після виявлення основних загроз важливо оцінити рівень ризиків, пов'язаних із кожною із них, а також зрозуміти їхній потенційний вплив на функціонування системи. Детальний аналіз дозволить виявити найбільш критичні аспекти, які потребують першочергового вирішення.

1.Злом мережі:

- Високий ризик за умов недостатнього використання шифрування або застарілих протоколів зв'язку.

- Потенційний наслідок: повний контроль над системою.

2.Несанкціонований доступ:

- Середній ризик, що може бути зменшений через впровадження багатфакторної автентифікації.

- Наслідки: втрата конфіденційності та можливість зловживання функціями системи.

3.Шкідливе програмне забезпечення:

- Середній рівень ризику через можливість впровадження через незахищені пристрої.

- Наслідки: збої в роботі системи та порушення цілісності даних.

4.Перехоплення даних:

- Підвищений ризик у разі використання незашифрованих каналів передачі.

- Наслідки: компрометація особистої та службової інформації.

5. Внутрішні помилки:

- Низький ризик за умови дотримання протоколів доступу.
- Наслідки: тимчасова втрата працездатності системи.

Після аналізу ризиків необхідно визначити найбільш ефективні стратегії, спрямовані на покращення рівня захищеності даних. Нижче подано перелік заходів, які є ключовими для мінімізації загроз.

1. Впровадження шифрування:

- Використання сучасних алгоритмів, таких як AES-256, для забезпечення захисту інформації на всіх етапах обробки.

2. Двофакторна автентифікація:

- Застосування біометричних даних або динамічних паролів як додаткового рівня захисту.

3. Моніторинг активності в реальному часі:

- Використання інструментів для виявлення аномалій, які можуть сигналізувати про загрози.

4. Своєчасні оновлення:

- Забезпечення регулярного оновлення прошивки та програмного забезпечення для усунення вразливостей.

5. Обмеження доступу:

- Впровадження політик на основі принципу мінімальних привілеїв.

6. Резервне копіювання:

- Налаштування автоматичного створення резервних копій критичних даних із перевіркою їхньої цілісності.

Дослідження зосереджене на локальних системах розумного будинку, а саме на методах забезпечення конфіденційності, безпеки даних і відповідності сучасним стандартам кібербезпеки. Особливу увагу приділено розробці політик управління даними, інтеграції автоматизованих засобів виявлення загроз і впровадженню стандартів для підтримки високого рівня довіри користувачів.

Ефективне забезпечення конфіденційності даних у локальних системах розумного будинку потребує розробки чітко визначених політик зберігання та обробки інформації. Основними елементами цих політик є:

1.Обмеження доступу до даних:

–Дані повинні бути доступні виключно авторизованим користувачам та компонентам системи.

–Використання багатофакторної автентифікації та сучасних криптографічних алгоритмів для захисту даних.

2.Політики видалення даних:

–Забезпечення автоматичного видалення даних після завершення строку зберігання чи досягнення мети обробки.

–Використання методів безпечного видалення, таких як криптографічне стирання або багаторазове перезаписування.

3.Логування доступу до даних:

–Ведення журналів усіх операцій із даними, що дає змогу ідентифікувати несанкціоновані дії або аномалії.

Принцип мінімізації збору даних є фундаментальним для зменшення ризиків, пов'язаних із витоком або компрометацією інформації. Основні аспекти:

1.Обмеження обсягу даних:

–Збір лише тих даних, які є необхідними для забезпечення функціональності системи.

–Уникнення зберігання дубльованих або надлишкових даних.

2.Анонімізація та псевдонімізація:

–Застосування технік, що унеможливають ідентифікацію конкретного користувача на основі зібраної інформації.

3.Прозорість процесів для користувачів:

–Надання чіткої та зрозумілої інформації про мету збору даних, обсяг і способи їх обробки.

Інтеграція систем виявлення вторгнень (IDS) є ключовою складовою забезпечення безпеки локальних систем. IDS виконують наступні функції:

1. Аналіз мережевого трафіку:

– Постійне сканування трафіку для виявлення аномалій, що можуть сигналізувати про загрозу або спробу несанкціонованого доступу.

2. Кореляція з журналами подій:

– Використання даних із логів для точнішої ідентифікації підозрілих активностей у системі.

3. Автоматичні сповіщення та превентивні заходи:

– У разі виявлення загрози система негайно повідомляє адміністратора та активує заздалегідь визначені заходи безпеки, такі як блокування підозрілих IP-адрес.

На основі проведеного аналізу предмету дослідження виявлено, що сучасні локальні системи розумного будинку потребують інтеграції багаторівневих механізмів захисту для забезпечення конфіденційності даних. Запропоновані рішення, зокрема адаптивні системи моніторингу та алгоритми автоматичного реагування на інциденти, сприяють зниженню ризиків втрати даних і підвищенню загальної ефективності роботи систем.

Наукова новизна підходів до аналізу ризиків і впровадження політик конфіденційності дозволяє використовувати отримані результати в розробці нових локальних систем IoT, орієнтованих на безпеку. Практична значимість роботи підтверджується її придатністю до впровадження у реальні інфраструктури та підвищенням рівня довіри користувачів.

3.5 Можливості масштабування та вдосконалення системи

Метою цього підрозділу є обговорення перспектив розширення системи розумного будинку, визначення напрямків її вдосконалення та інтеграції нових технологій для забезпечення довготривалої актуальності та функціональності.

Масштабування системи розумного будинку є однією з найбільш значущих задач, яка визначає її здатність адаптуватися до зростаючих потреб користувачів і технологічного прогресу. У контексті постійного розширення екосистем IoT забезпечення безшовної інтеграції нових компонентів стає пріоритетним завданням.

Система розумного будинку має бути побудована з урахуванням можливостей інтеграції нових компонентів без зниження її продуктивності. Це забезпечується:

1. Використанням нових підходів до архітектури:

– Перехід до мікросервісної архітектури для підвищення гнучкості та масштабованості системи.

– Застосування контейнеризації (наприклад, Docker) для легшого управління компонентами.

2. Розширенням протоколів зв'язку:

– Впровадження підтримки сучасних стандартів, таких як Thread, для покращення інтеграції пристроїв у мережу.

3. Автоматизацією управління мережею:

– Використання інструментів для автоматичного балансування навантаження між пристроями з метою підвищення ефективності.

– Використання локальних шлюзів і серверів для обробки даних дозволяє зменшити навантаження на мережу та зберігати швидкодію навіть при значному збільшенні кількості підключених пристроїв.

Система повинна не лише масштабуватися, але й адаптуватися до змін у потребах користувачів і характеристиках пристроїв [50]. Це особливо важливо у контексті сучасного розвитку технологій, де швидкі зміни стають нормою. Для забезпечення довговічності та актуальності системи, необхідно впроваджувати рішення, які дозволяють їй залишатися ефективною навіть в умовах стрімкого технологічного прогресу. Гнучкість системи відіграє ключову роль у її успішному функціонуванні. Вона має включати параметри конфігурації, які

можуть адаптуватися до індивідуальних потреб користувачів. Основні напрямки розвитку в цьому контексті включають:

1.Індивідуалізація функцій:

–Налаштування сценаріїв автоматизації залежно від поведінкових патернів користувачів.

2.Інтеграція персоналізованого управління:

–Використання мобільних додатків або голосових асистентів для зміни параметрів системи у реальному часі.

3.Машинне навчання для адаптації:

–Впровадження алгоритмів, що аналізують звички користувачів і автоматично налаштовують функції системи.

Інтеграція систем розумного будинку з елементами міської інфраструктури є одним із ключових напрямків розвитку IoT-екосистем. Така інтеграція дозволяє створити гармонійне середовище, де приватні домогосподарства, транспортні мережі, енергетичні системи та інші компоненти міського життя функціонують як єдиний механізм.

Реалізація інтеграції систем розумного будинку з міськими інфраструктурами спрямована на забезпечення безперервності функціонування міських процесів та підвищення якості життя мешканців. Завдяки такій співпраці можна досягти значної ефективності у використанні міських ресурсів, зниженні екологічного навантаження та покращенні управління міськими службами.

1.Оптимізація транспортних потоків:

–Системи розумного будинку можуть взаємодіяти з міськими службами для отримання інформації про транспортну ситуацію, сприяючи плануванню переміщень мешканців.

–Інтеграція з інтелектуальними парковками забезпечує автоматичне резервування місць для автомобілів.

2.Енергетична ефективність:

–Співпраця з міськими енергомережами дозволяє адаптувати споживання енергії будинком до пікових навантажень мережі.

–Використання відновлюваних джерел енергії, таких як сонячні панелі, інтегровані в міські енергетичні системи.

3.Екологічний моніторинг:

–Розумні будинки можуть забезпечувати збір даних про стан навколишнього середовища (рівень забруднення, вологість тощо) та передавати їх до міських систем моніторингу.

Ефективна взаємодія систем розумного будинку з енергомережами є критичною для оптимізації споживання ресурсів і забезпечення сталого розвитку. Використання інтелектуальних технологій у цій галузі сприяє раціоналізації енергетичних витрат, підвищенню автономності будинків та мінімізації впливу на довкілля.

1.Гнучке управління споживанням:

–Система розумного будинку може отримувати дані про тарифи на електроенергію в реальному часі та адаптувати споживання для мінімізації витрат.

–Інтеграція з розумними лічильниками дозволяє мешканцям контролювати споживання енергії через мобільні додатки.

2.Використання децентралізованих джерел енергії:

–Інтеграція з мікромережами сприяє розподілу надлишків енергії, виробленої домогосподарством, серед інших споживачів у мережі.

–Забезпечення автономної роботи будинку в разі відключення основного енергопостачання.

3.Підтримка відновлюваних джерел енергії:

–Використання систем накопичення енергії (акумуляторів) для оптимізації її використання в години максимального споживання.

Інтеграція з системами безпеки дозволяє забезпечити більш високий рівень захисту мешканців та об'єктів нерухомості. Завдяки поєднанню можливостей розумного будинку та міських систем реагування створюється

скоординована інфраструктура, здатна оперативно вирішувати кризові ситуації та зменшувати ризики.

1.Централізований контроль безпеки:

–Система розумного будинку може передавати дані про стан об'єкта до міських систем безпеки для оперативного реагування на загрози.

–Інтеграція з системами відеоспостереження дозволяє органам правопорядку отримувати доступ до камер у разі надзвичайної ситуації.

2.Моніторинг надзвичайних ситуацій:

–Взаємодія з міськими службами надзвичайних ситуацій (пожежна безпека, швидка допомога) забезпечує автоматичне сповіщення у разі виявлення загроз.

3.Синхронізація з громадськими системами оповіщення:

–Інтеграція з міськими системами дозволяє розумним будинкам передавати інформацію про потенційні загрози мешканцям у реальному часі.

Інтеграція систем розумного будинку з міськими інфраструктурами, енергомережами та системами безпеки створює синергетичний ефект, що дозволяє оптимізувати використання ресурсів, підвищити рівень безпеки та забезпечити екологічну стійкість. Такий підхід сприяє створенню гармонійного середовища, орієнтованого на комфорт і ефективність у рамках концепції "розумного міста".

ВИСНОВОК

У процесі виконання роботи було розглянуто та проаналізовано основні аспекти створення автономної системи розумного будинку з локальною обробкою даних. Проведені дослідження дозволили встановити, що автономна обробка даних забезпечує ряд значних переваг, таких як підвищена безпека, стабільність роботи незалежно від стану Інтернет-з'єднання та конфіденційність персональних даних.

Порівняння комерційних і відкритих платформ для розумних будинків дало можливість визначити оптимальні підходи та необхідні технічні вимоги для реалізації автономних функцій. В результаті було підібрано найбільш відповідне обладнання та сенсори, що забезпечують ефективну роботу системи в умовах локальної обробки.

Проведений порівняльний аналіз хмарної та локальної обробки показав, що хоча хмарні рішення можуть забезпечити більшу масштабованість і зручність для користувачів, локальна обробка є більш оптимальним вибором з точки зору безпеки, швидкості доступу до даних і стабільності роботи. Розроблена методологія розгортання такої системи враховує ці переваги та дозволяє створити стабільне, безпечне та ефективне рішення для керування розумним будинком.

Загалом, виконана робота підтверджує доцільність використання автономних систем у розумних будинках для підвищення безпеки, надійності та незалежності від зовнішніх сервісів. Отримані результати можуть бути використані для подальшої оптимізації рішень у сфері розумного житла, зокрема для створення нових систем, що ще більше підвищують автономність і функціональність сучасних будинків.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Smart Home Automation with IoT [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://smarthomeautomation.org> (дата звернення 13.11.2024) –
2. Schneider Electric. Smart Home Solutions [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://www.se.com/ua> (дата звернення 13.10.2024) –
3. Рябцев М. Розумний будинок: сучасні технології автоматизації житла / М. Рябцев. – Київ: Технопрес, 2021. – 240 с.
4. Mullen C. The Future of Smart Home Technology / C. Mullen. – New York: TechVision, 2022. – 180 p.
5. Philips Hue [Електронний ресурс]: [Веб-сайт]. – Ейндговен, Нідерланди: Signify. – Режим доступу: <https://www.philips-hue.com> (дата звернення 08.11.2024)
6. Smart Home History and Evolution [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://www.smarthome.ua/ua/istoriya-umnogo-doma.html> (дата звернення 23.10.2024)
7. Internet of Things in Smart Homes [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://iot-smarthomes.com> (дата звернення 06.11.2024)
8. IoT Solutions for Smart Living [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://iot-solutions.org> (дата звернення 15.10.2024)
9. Сучасні рішення для розумного будинку: технології, переваги та інновації [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://vash-master.com.ua/suchasni-rishennya-dlya-rozumnogo-budinku-texnologii-perevagi-ta-innovatsii/> (дата звернення 29.11.2024)
10. Обробка даних в реальному часі в смарт-домах [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://peerdh.com/uk/blogs/programming-insights/real-time-data-processing-in-smart-home-systems> (дата звернення 28.11.2024)
11. Система енергоефективної автоматизації «розумного будинку» [Електронний ресурс]: [Веб-сайт]. – Режим доступу: https://stud.knutd.edu.ua/bitstream/123456789/24369/1/Dyplom151_Saveliev_Skidan.pdf (дата звернення 01.11.2024)
12. Методи забезпечення безпеки розумного будинку [Електронний ресурс]:

[Веб-сайт]. – Режим доступу: <https://csecurity.kubg.edu.ua/index.php/journal/article/view/119> (дата звернення 07.11.2024)

13.Можливості розумного будинку - огляд систем [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://vencon.ua/ua/articles/vozmozhnosti-umnogo-doma-obzor-sistem> (дата звернення 14.11.2024)

14.Автоматизація освітлення в «Розумному домі»: зручність, ефективність та безпека [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://smartsys.team/uk/our-blog/light-smarthome/> (дата звернення 07.11.2024)

15.Системи безпеки в Розумному домі [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://www.larnitech.com/ua/blog/security-systems-in-a-smart-house/> (дата звернення 14.11.2024)

16.Контроль доступу в розумному будинку, максимальний захист від проникнення [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://smartsite.com.ua/blog/kontrol-dostupu-v-rozumnomu-budynku/> (дата звернення 23.10.2024)

17.Клімат-контроль та розумний будинок [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://www.larnitech.com/ua/blog/climate-control-and-a-smart-house/> (дата звернення 13.10.2024)

18.Розумний будинок і енергонезалежність: технології майбутнього [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://kvolts.com.ua/rozumnyi-budynok-ta-enerhonezalezhnist-idealnyi-duet/> (дата звернення 29.11.2024)

19.Енергоефективні рішення для розумного дому: як знизити витрати на енергію [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://umniydom.com.ua/energoefektivni-rishennya-dlya-rozumnogo-domu-yak-zniziti-vitrati-na-energiyu/> (дата звернення 03.11.2024)

20. Системи безпеки в Розумному домі [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://www.larnitech.com/ua/blog/security-systems-in-a-smart-house/> (дата звернення 10.11.2024)

21. Керування розумним будинком за допомогою III: як це працює і які можливості [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://smartsite.com.ua/blog/keruvannya-rozumnym-budynkom-za-dopomogoyu-shi/> (дата звернення 15.11.2024)

22. Огляд протоколів управління «розумним будинком» [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://conf.ztu.edu.ua/wp-content/uploads/2021/01/39-1.pdf> (дата звернення 13.10.2024)

23. Zigbee проти Bluetooth проти Wi-Fi для розумних будинків [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://www.dusuniot.com/uk/blog/zigbee-vs-bluetooth-vs-wi-fi-which-one-is-the-best-for-smart-home/> (дата звернення 08.12.2024)

24. Технологія Bluetooth Low Energy: можливості використання [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://rossinno.net/bluetooth-low-energy> (дата звернення 11.10.2024)

25. Zigbee: що це таке, пристрої, дальність зв'язку [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://bezprovodoff.com/seti/vidy/zigbee-hto-eto-takoe.html> (дата звернення 06.12.2024)

26. Z-Wave Protocol [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://www.geeksforgeeks.org/z-wave-protocol/> (дата звернення 10.11.2024)

27. Протоколи зв'язку для розумного дому [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://wisehome.ua/ua/scho-take-protokol-dlya-rozumnogo-budinku-6-populyarnih-vidiv/> (дата звернення 23.11.2024)

28. Стандарт розумного дому Matter офіційно затверджено, його підтримка має з'явитися в HomeKit, Google Home та інших системах [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://mezha.media/2022/10/05/standart-rozumnoho-domu-matter/> (дата звернення 16.10.2024)

29. Ethernet чи Wi-Fi: Що краще обрати для дому та офісу - порівняння [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://mfix.ua/ethernet-vs-wifi/> (дата звернення 12.11.2024)

30. Що таке Powerline адаптер і як він працює [Електронний ресурс]: [Веб-

сайт]. – Режим доступу: <https://root-nation.com/ua/articles-ua/tech-ua/ua-powerline-adapter/> (дата звернення 18.11.2024)

31. Установка openHAB в Windows [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://sprut.ai/article/ustanovka-openhab-v-windows> (дата звернення 22.11.2024)

32. Home Assistant 101. Посібник для початківців [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://dou.ua/forums/topic/38947/> (дата звернення 02.12.2024)

33. SmartThings порівняно з Fibaro, Vera, Domoticz тощо — кілька конкретних питань інтеграції [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://community.smarthings.com/t/smarthings-compared-to-fibaro-vera-domoticz-etc-a-few-specific-integration-questions/61672> (дата звернення 04.11.2024)

34. Розробка API додатків: створення API для мобільних та веб додатків [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://pnn.com.ua/ua/it-solutions/custom-api-development> (дата звернення 06.11.2024)

35. Home Assistant Developer Docs [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://developers.home-assistant.io/> (дата звернення 31.10.2024)

36. Впровадження крайових обчислень для обробки даних у реальному часі в розумних домівках з Raspberry Pi [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://peerdh.com/uk/blogs/programming-insights/implementing-edge-computing-for-real-time-data-processing-in-smart-homes-with-raspberry-pi> (дата звернення 03.12.2024)

37. Впровадження крайових обчислень для обробки даних у реальному часі в розумних домівках з Raspberry Pi [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://peerdh.com/uk/blogs/programming-insights/implementing-edge-computing-for-real-time-data-processing-in-smart-homes-with-raspberry-pi> (дата звернення 04.12.2024)

38. Опрацювання даних у системах "розумного" будинку з використанням моделей на підставі мереж Петрі [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://nv.nltu.edu.ua/index.php/journal/article/view/2264> (дата звернення

01.11.2024)

39. Розумні системи управління домом: освітлення, безпека, опалення [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://resheniya.com.ua/uk/rozumni-sistemi-upravlinnya-domom-osvitlennya-bezpeka-opalennya/> (дата звернення 12.10.2024)

40. Як користуватися Alexa: функції, поради та підказки в нашому повному посібнику [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://hitech.mediadoma.com/uk/jak-koristuvatisja-alexa-funkcii-poradi-ta-pidkazki-v-nashomu-povnomu-posibniku/> (дата звернення 13.10.2024)

41. Що може «розумний» будинок Apple HomeKit [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://journal.ilounge.ua/ua/review/apple-homekit-chto-mozhet-umnyj-dom> (дата звернення 15.10.2024)

42. Опрацювання даних у системах "розумного" будинку з використанням моделей на підставі мереж Петрі [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://nv.nltu.edu.ua/index.php/journal/article/view/2264> (дата звернення 29.11.2024)

43. Дослідження протоколів передачі даних в умовах Інтернету речей [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://journals.nupp.edu.ua/sunz/article/view/2446> (дата звернення 05.12.2024)

44. Що таке SDK і чому він важливий для розробників [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://wezom.com.ua/ua/blog/scho-take-sdk-i-chomu-vin-vazhliiy-dlya-rozrobnikiv> (дата звернення 01.11.2024)

45. Хмарні обчислення: ключові аспекти та переваги для бізнесу та користувачів [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://saascompany.com.ua/hmarni-obchyslennya-klyuchovi-aspekty-ta-perevagy-dlya-biznesu-ta-korystuvachiv-2/> (дата звернення 06.12.2024)

46. Виробничі процеси та обладнання об'єктів автоматизації [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://openarchive.nure.ua/bitstreams/50959360-2aaf-4b25-9e6c-dd7437bbc3af/download> (дата звернення 14.11.2024)

47. Nanoleaf [Електронний ресурс]: [Веб-сайт]. – Електронні дані. – Канада: Nanoleaf. – Режим доступу: www.nanoleaf.me (дата звернення 28.11.2024) – Інтерактивні освітлювальні системи.

48. Розумний будинок: що це таке, як працює, що вміє система розумного будинку [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://alterair.ua/stati/chto-takoe-umnyy-dom/> (дата звернення 15.10.2024)

49. Кібератаки на IoT пристрої: мета, наслідки та рішення [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://wezom.com.ua/ua/blog/kiberzagrozi-dlya-internetu-rechey-iot-zahist-smart-pristroyiv> (дата звернення 16.10.2024)

50. Розумні будинки: функції, пристрої та принцип роботи [Електронний ресурс]: [Веб-сайт]. – Режим доступу: <https://bizmag.com.ua/rynok-rozumnyh-budynkiv/> (дата звернення 26.11.2024)

ДОДАТОК А

```
1  #include <SD.h>
2  #include <SPI.h>
3  #include <Wire.h>
4
5  const int chipSelect = BUILTIN_SDCARD;
6
7  float temperature = 0.0;
8  float humidity = 0.0;
9
10 void setup() {
11     Serial.begin(9600);
12     while (!Serial);
13     Serial.println("Система збору даних запускається...");
14
15     if (!SD.begin(chipSelect)) {
16         Serial.println("Помилка SD-карти!");
17         while (1);
18     }
19     Serial.println("SD-карта успішно ініціалізована.");
20
21     Wire.begin();
22 }
23
24 void loop() {
25     temperature = readTemperature();
26     humidity = readHumidity();
27     String dataString = "Температура: " + String(temperature) + " C, Вологість: " + String(humidity) + " %";
28     File dataFile = SD.open("datalog.txt", FILE_WRITE);
29     if (dataFile) {
30         dataFile.println(dataString);
31         dataFile.close();
32         Serial.println("Дані записано: " + dataString);
33     } else {
34         Serial.println("Помилка запису у файл.");
35     }
36     delay(1000);
37 }
38
39 float readTemperature() {return random(20, 30) + random(0, 100) / 100.0;}
40
41 float readHumidity() {return random(40, 60) + random(0, 100) / 100.0;}
```

ДОДАТОК Б

```
1  #include <Arduino.h>
2  #include <SD.h>
3  #include <Wire.h>
4
5  const float TEMP_THRESHOLD = 28.0;
6  const float HUMIDITY_THRESHOLD = 50.0;
7  void analyzeData(float temperature, float humidity);
8  void triggerEvent(const char* eventMessage);
9
10 void setup() {
11     Serial.begin(9600);
12     while (!Serial);
13     Serial.println("Модуль аналізу запущено...");
14     if (!SD.begin(BUILTIN_SDCARD)) {
15         Serial.println("Помилка SD-карти!");
16         while (1);
17     }
18     Serial.println("SD-карта успішно ініціалізована.");
19
20 void loop() { float temperature = readTemperature();
21 float humidity = readHumidity();
22 analyzeData(temperature, humidity);
23 delay(1000);}
24
25 void analyzeData(float temperature, float humidity) {if (temperature > TEMP_THRESHOLD) {triggerEvent("Перевищено поріг температури");}
26 if (humidity > HUMIDITY_THRESHOLD) {triggerEvent("Перевищено поріг вологості");}
27 if (temperature > 30.0 && humidity < 40.0) {triggerEvent("Сценарій: Сухе гаряче середовище");}}
28
29 void triggerEvent(const char* eventMessage) {Serial.println(eventMessage);
30 File eventFile = SD.open("events.txt", FILE_WRITE);
31 if (eventFile) {eventFile.println(eventMessage);
32 eventFile.close();
33 Serial.println("Подія записана на SD-карту: " + String(eventMessage));
34 } else {Serial.println("Помилка запису події на SD-карту.");}}
35
36 float readTemperature() {return random(20, 35) + random(0, 100) / 100.0;}
37
38 float readHumidity() { return random(30, 60) + random(0, 100) / 100.0;}
39 |
```

ДОДАТОК В

```
1  #include <Arduino.h>
2
3  const int lightPin = 5;
4  const int heaterPin = 6;
5  const int alarmPin = 7;
6
7  void activateDevice(int pin);
8  void deactivateDevice(int pin);
9  void handleControl(float temperature, float humidity);
10
11 void setup() {pinMode(lightPin, OUTPUT);
12   pinMode(heaterPin, OUTPUT);
13   pinMode(alarmPin, OUTPUT);
14   Serial.begin(9600);
15   while (!Serial);
16   Serial.println("Модуль керування запущено...");}
17
18 void loop() {float temperature = readTemperature();
19   float humidity = readHumidity();
20   handleControl(temperature, humidity);
21   delay(1000);}
22
23 void activateDevice(int pin) {digitalWrite(pin, HIGH);}
24
25 void deactivateDevice(int pin) {digitalWrite(pin, LOW);}
26
27 void handleControl(float temperature, float humidity) {
28   if (temperature > 30.0) {activateDevice(heaterPin);}
29   else {deactivateDevice(heaterPin);}
30
31   if (humidity < 40.0) {activateDevice(lightPin);} else {deactivateDevice(lightPin);}
32
33   if (temperature > 35.0 || humidity > 70.0) {
34     activateDevice(alarmPin);
35   } else {
36     deactivateDevice(alarmPin);
37   }
38 }
39
40 float readTemperature() {return random(20, 40) + random(0, 100) / 100.0;}
41
42 float readHumidity() {return random(30, 80) + random(0, 100) / 100.0;}
43
```

ДОДАТОК Г

```
1  #include <WiFi.h>
2  #include <HTTPClient.h>
3
4  const char* ssid = "Your_SSID";
5  const char* password = "Your_PASSWORD";
6  const char* serverUrl = "http://example.com/api/data";
7
8  void connectToWiFi();
9  void sendData(float temperature, float humidity);
10
11 ∨ void setup() {Serial.begin(9600);
12   while (!Serial);
13   connectToWiFi();}
14
15 ∨ void loop() {float temperature = readTemperature();
16   float humidity = readHumidity();
17   sendData(temperature, humidity);
18   delay(5000);}
19
20 ∨ void connectToWiFi() {WiFi.begin(ssid, password);
21   while (WiFi.status() != WL_CONNECTED) {
22     delay(500);}
23 }
24 ∨ void sendData(float temperature, float humidity) {
25   if (WiFi.status() == WL_CONNECTED) {
26     HTTPClient http;
27     http.begin(serverUrl);
28     http.addHeader("Content-Type", "application/json");
29     String jsonPayload = "{\"temperature\": " + String(temperature) + ", \"humidity\": " + String(humidity) + "}";
30     http.POST(jsonPayload);
31     http.end();
32   }
33 }
34
35 ∨ float readTemperature() {
36   return random(20, 40) + random(0, 100) / 100.0;
37 }
38
39 ∨ float readHumidity() {
40   return random(30, 80) + random(0, 100) / 100.0;
41 }
42
```

ДОДАТОК Д. ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)



ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ
ТЕХНОЛОГІЙ

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра Комп'ютерної інженерії



КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня магістра
На тему: «Автономна система розумного будинку з локальною
обробкою даних»

Розробив: студент групи КСМД-61
Писаренко Денис Сергійович
Керівник : Лащевська Наталія Олександрівна
кандидат технічних наук, доцент

Мета роботи – створення автономної системи розумного будинку, яка забезпечує функціональність, безпеку, конфіденційність даних та зручність у використанні завдяки локальній обробці інформації.

Об'єктом дослідження є процес функціонування розумного будинку з локальною обробкою даних. Цей процес породжує проблемні ситуації, пов'язані з інтеграцією нових пристроїв, оптимізацією обробки даних і підвищенням надійності системи.

Предмет дослідження – функціональні можливості та переваги автономних систем розумного будинку, що базуються на локальній обробці даних.

Актуальність обраної теми

Забезпечення комфорту, безпеки та ефективного управління ресурсами стає все більш актуальним завданням у сфері сучасного житла. Інноваційні рішення, що використовуються у «розумних» будинках, дозволяють автоматизувати виконання багатьох рутинних дій та підвищують загальну якість життя.

Одним із ключових аспектів побудови сучасних розумних будинків є вибір між локальною та хмарною обробкою даних.

3

Аналіз потреб

Аналіз потреб є важливим етапом розробки автономної системи розумного будинку, оскільки він дозволяє визначити, які функції та можливості потрібні для забезпечення комфортного, безпечного та енергоефективного середовища.

Цей процес передбачає вивчення специфічних вимог користувача, особливостей житлового простору та існуючої інфраструктури.

4

Архітектурні відмінності систем розумного будинку



Рисунок 1 - Автономна система

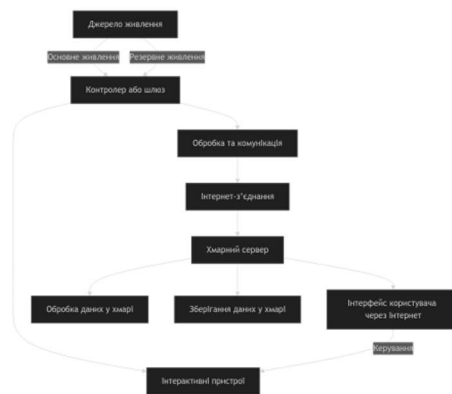
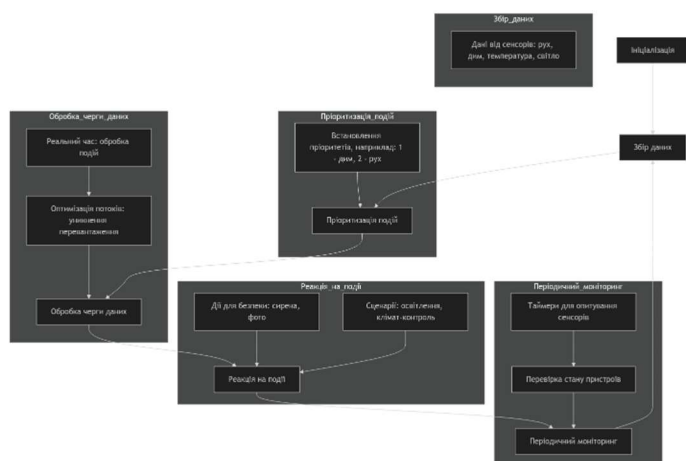


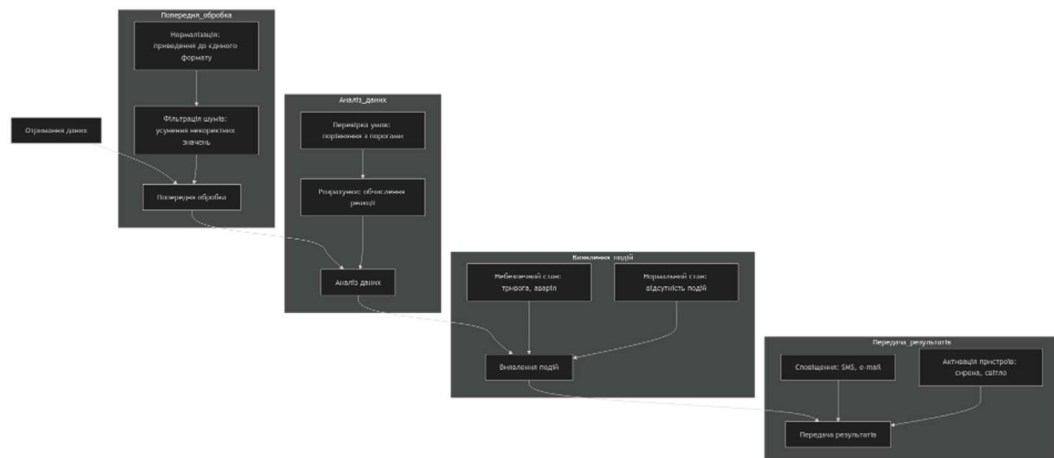
Рисунок 2 – Традиційна система

Програмне забезпечення системи побудоване на основі модульної архітектури. Ключові компоненти:

- Збір даних:

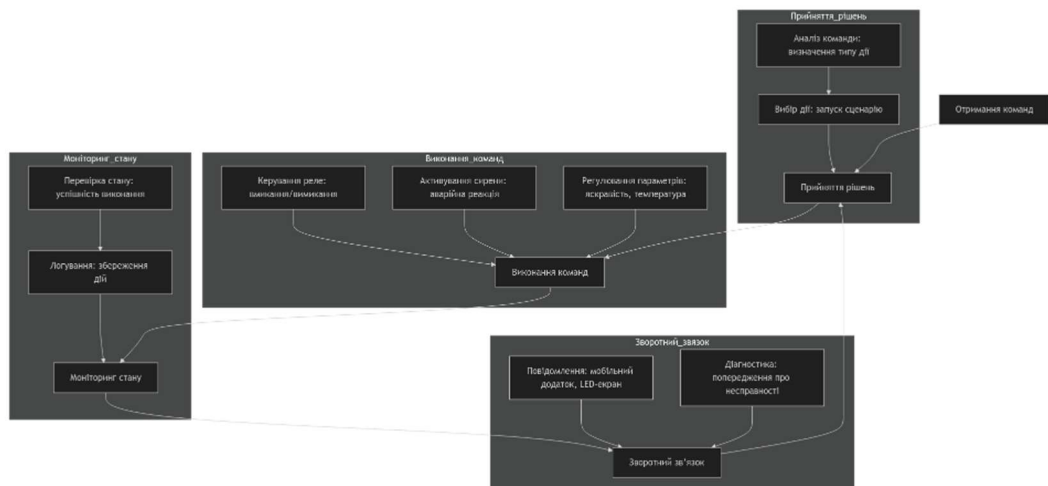


- Аналіз даних :



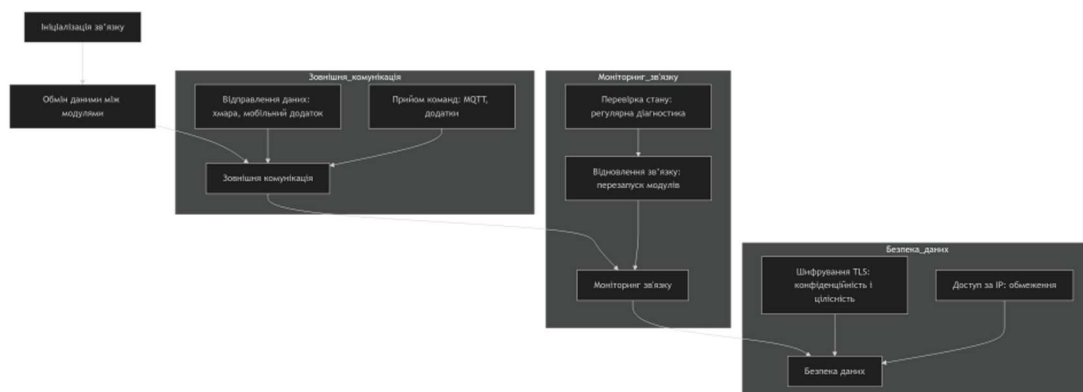
7

- Керування:



8

- Комунікація:



Елементи систем розумного будинку



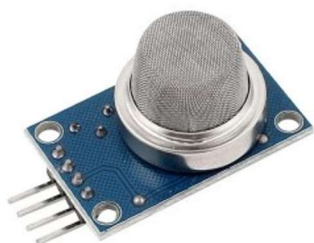
Arduino Portenta H7



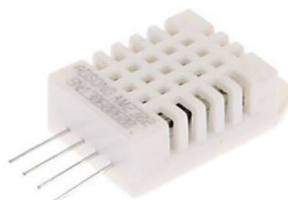
Датчик руху PIR (HC-SR501)



Магнітоконтактний датчик



Датчик диму та газу (MQ-2)



Датчик температури та вологості (DHT22)



Датчик освітленості (BH1750)

Елементи систем розумного будинку



Камера (ESP32-CAM)



Реле 1 канал 12 В



Сирена

11

Висновки

1. Використання енергоефективних пристроїв.
2. Система підтримує модульну архітектуру, що дозволяє легко інтегрувати нові сенсори чи пристрої.
3. Можливість підключення до інших IoT-систем для розширення функціональності (наприклад, систем енергоменеджменту).
4. Мінімізація залежності від хмарних сервісів та швидкість реакції системи.

12