ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ТЕЛЕКОМУНІКАЦІЙ ТА ІНФОРМАТИЗАЦІЇ

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра інформаційних систем та технологій

Пояснювальна записка

до магістерської роботи
на освтіній ступінь магістр

на тему: «**РОЗРОБКА СИСТЕМИ «РОЗУМНИЙ ДІМ» НА БАЗІ ТЕХНОЛОГІЇ Z-WAVE**

Виконав: студентка 6 курсу, групи ІМДМ-61
спеціальності 6.050903 Телекомунікації
(шифр і назва спеціальності)
Ковтун А.Р._____
(прізвище та ініціали)
Керівник Полоневич О.В._____
(прізвище та ініціали)
Рецензент_____
(прізвище та ініціали)
Нормоконтроль_____
(прізвище та ініціали)

Київ – 2019

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ ТА ІНФОРМАТИЗАЦІЇ

Факультет Інформаційних технологій

Кафедра Інформаційних систем та технологій

Ступінь вищої освіти - «Магістр»

Спеціальність підготовки 172 Інформаційні мережі зв'язку

ЗАТВЕРДЖУЮ

Завідувач кафедри ІСТ

_____ К.П.Сторчак

"____" _____2019 року

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ

Ковтун Анастасії Русланівні
(прізвище, ім'я, по батькові)

1.Тема роботи: «Розробка системи «Розумний дім» на базі технології Z-Wave»
Керівник роботи: Полоневич Ольга Володимирівна, к.т.н. _____,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом вищого навчального закладу від «» дата року №184
2. Строк подання студентом роботи дата_____
3. Вихідні дані до роботи :

1. Науково-технічна література
2. Методи побудови інтелектуальних систем
3. Стандарти концепції «розумний дім»

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

1. Аналіз основних принципів концепції «Інтернет речей»
2. Розгляд функціональних елементів «розумного дому»
3. Аналіз готових рішень для побудови системи «розумний дім
4. Розробка власної системи «розумний дім» на Raspberry Pi
5. Перелік графічного матеріалу
6. Перелік демонстраційного матеріалу(презентація)
7. Дата видачі завдання дата

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури		
2	Вивчення матеріалів для аналізу та порівняння роботи інтелектуальних систем		
3	Дослідження концепцій «Інтернету речей»		
4	Дослідження сфер застосування «Інтернету речей»		
5	Аналіз способів управління системою «розумний дім»		
6	Розробка власної системи «розумний дім» на Raspberry Pi.		
7	Вступ, висновки, реферат		
8	Розробка демонстраційних матеріалів		

Студент _____ Ковтун А.Р.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Полоневич О.В.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина магістерської роботи 78 с., 27 рис., 13 джерел.

Об`єкт дослідження: інтелектуальні системи для житлових приміщень.

Предмет дослідження: розробка моделі інтелектуального помічника.

Мета роботи: дослідити основні критерії роботи інтелектуальних систем для житлових приміщень та розробити власну систему «розумний дім» на Arduino.

Методи дослідження: методи теорії інформації, методи наукового моделювання, методи дослідження інформаційних систем.

У роботі проведено аналіз шляху розвитку технології інтелектуальних систем. Інтелектуальні системи позиціонуються як ядро, яку розробники будуть поступово впроваджувати в концепцію Інтернету речей. Досліджено актуальну на сьогодні концепцію «розумний дім» та визначено властивості, які дозволяють забезпечити коректну роботу технології на сучасному етапі розвитку.

Проведено аналіз рівня впровадження концепції «розумний дім» у світі та Україні. Розглянуто технічні засоби для розробки систем концепції «Інтернету речей» та рівень їх розвитку. Досліджено та запропоновано до реалізації метод впровадження концепції «розумний дім» у повсякденне життя, на основі апаратних засобів Raspberry Pi та мови програмування Python. Представлено приклад такого впровадження, та його переваги.

Галузь використання: сучасні системи IT-інфраструктур

ІНТЕЛЕКТУАЛЬНИЙ ПОМІЧНИК, ІНТЕРНЕТ РЕЧЕЙ, РОЗУМНИЙ ДІМ, RASPBERRY PI, IOT, СЕНСОРИ, ДАТЧИКИ, КОНЦЕПЦІЯ, PYTHON, Z-WAVE

ЗМІСТ

ВСТУП 8
1 ДОСЛІДЖЕННЯ КОНЦЕПЦІЇ ІНТЕРНЕТУ РЕЧЕЙ 10
1.1. Основні принципи концепції «Інтернет речей» 10
1.2 Сфери застосування Інтернету речей 21
2 **ФУНКЦІОНАЛЬНІ ЕЛЕМЕНТИ ТА ВИМОГИ ДО РЕАЛІЗАЦІЇ СИСТЕМИ «РОЗУМНИЙ ДІМ»** 37
2.1 Концепція «Розумний дім» 37
2.2 Функціональні елементи «розумного дому» 45
2.3 Аналіз готових рішень для побудови системи «розумний дім» . **51**
3 РОЗРОБКА СИСТЕМИ «РОЗУМНИЙ ДІМ» НА RASBERRY PI 58
3.1 Вибір мережевої інфраструктури 58
3.2 Опис моделі керування 62
3.3 Апаратна модель системи автоматизованого управління 68
3.4 Опис програмної частини моделі автоматизованого управління 77
ВИСНОВКИ 84
ПЕРЕЛІК ПОСИЛАНЬ 86
ДОДАТКИ 88
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація) 96

ВСТУП

На сьогоднішній день інтелектуальні системи є одними з найбільш динамічних галузей інформаційних технологій. Розширення спектру їх можливостей та швидкості обробки даних призводять до ускладнення процесу перерозподілу мережевих ресурсів та збільшення обчислювальних потужностей які необхідні для обслуговування великих систем обробки даних. Більшість сучасних інформаційних систем базуються на управлінні із людським втручанням, та топологіях які були впровадженні ще у минулому столітті. Звичайно, така технологія задовольняє усі потреби інформаційної системи, але при цьому використовуючи необґрунтовано великий обсяг часу при обробці деякої задачі. Одним з прогресивних напрямків організації роботи центрів обробки даних та інформаційних систем є впровадження в робочий процес інтелектуальних помічників та елементів «Інтернету речей».

Разом з розвитком «Інтернету речей» досвід користувача поширився і на численні «розумні», підключені до мережі пристрої. Забезпечення однакової взаємодії навіть з серією пристроїв одного виробника є нетривіальним завданням для проектувальників і дизайнерів, так як, не дивлячись на різноманітність фізичних інтерфейсів, користувач повинен відчувати єдність закладеного в послузі задуму.

На 2019 рік термін «Інтернет речей» поширюється не тільки на програмно-апаратні системи для домашнього застосування, але і на промислові об'єкти. Розвиток концепції «Інтелектуальних будівель» отримало назву «Building Internet of Things», тобто «Інтернет речей в будинку», розвиток розподіленої мережевої інфраструктури в АСУ ТП призвело до появи «Industrial Internet of Things», тобто «Індустріальний (промисловий) інтернет речей»

У віддаленій перспективі варто очікувати появи єдиного цілого, що складається з традиційних технологій для роботи з даними і з промисловим систем управління (ICS) і автоматизованих систем диспетчерського

управління та збору даних (SCADA). Можливо, в кінцевому підсумку це будуть програмно-апаратні системи або навіть соціальні програмно-апаратні системи.

Застосування інтелектуальних систем та помічників значно дозволяє знизити сукупну вартість обслуговування центрів обробки даних та інформаційних систем, оптимізувати час виконання задачі та час обробки запитів, зробити сервіси більш доступними для користувачів, підвищити рівень безпеки завдяки відсутності людського фактору, а також реалізувати нові програмні методи керування мережами і цим самим збільшити їх гнучкість та керованість.

1 ДОСЛІДЖЕННЯ КОНЦЕПЦІЇ ІНТЕРНЕТУ РЕЧЕЙ

1.1 Основні принципи концепції «Інтернет речей»

Інтернет речей або IoT - це система взаємопов'язаних обчислювальних пристроїв, механічних і цифрових машин, предметів, тварин або людей, яким надаються унікальні ідентифікатори (UID) та можливість передавати дані по мережі. Все частіше організації в різних галузях використовують IoT для більш ефективної роботи, що допомагає їм краще розуміти клієнтів для надання розширеного обслуговування, поліпшення прийняття рішень та підвищення цінності бізнесу. [1]

Визначення IoT склалося завдяки зближенню декількох технологій, таких як бездротова технологія, машинне навчання, автоматизація, мікроелектромеханічні системи та Інтернет. Ця конвергенція сприяла усуненню розриву між операційною технологією та інформаційними технологіями, дозволяючи аналізувати неструктуровані машинно-сформовані дані для розуміння, що сприятиме інноваціям. Поняття мережі розумних пристроїв існувало з 1982 року. Першим пристроєм, підключеним до Інтернету, став модифікований апарат Coke в університеті Карнегі Меллон. Машина Coke змогла повідомити про свій інвентар та про те, чи були нещодавно завантажені напої холодними. Сучасне бачення, яке ми маємо сьогодні в Інтернеті, значною мірою пояснюється зусиллями роботи Марка Вайзера "Комп'ютер 21 століття". У його доповіді було розказано про всюдисущі обчислення та про те, як з часом технології настільки переплелися з нашими живе до того моменту, коли ми навіть не могли собі уявити, яким би було життя в іншому випадку. У своїй книзі "Нові технології для навчання" Девід Бекта описує всюдисущі обчислення як "бачення обчислювальної потужності, вбудованого у навколишній світ і доступ до нього через інтелектуальні інтерфейси". Найвищий ідеал - зробити комп'ютер таким чином вбудованим, що, ми використовуємо його, навіть не замислюючись про це.

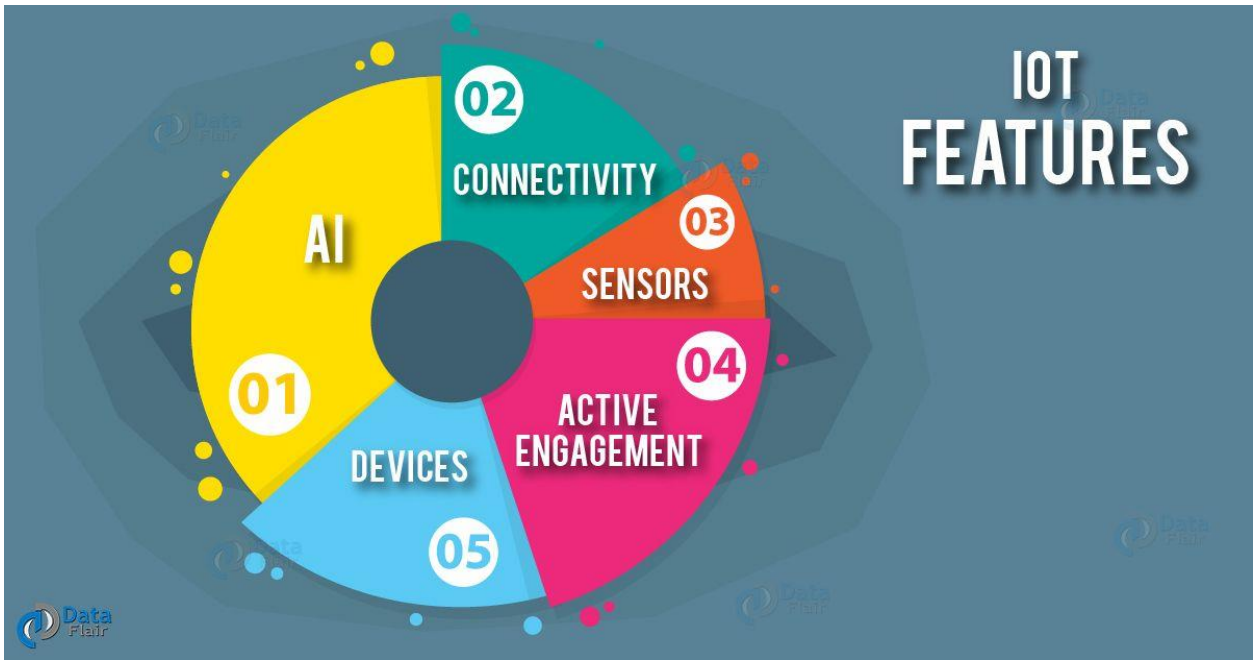

Рисунок 1.1 – Особливості IoT

AI - IoT технічно робить речі розумними, тобто покращує різні аспекти життя завдяки правильному використанню даних, мереж та алгоритмів. Це може варіюватися від чогось простого, як поліпшення або покращення вашого холодильника, вбудовуючи його в датчики, які автоматично визначають, коли молоко та яйця закінчуються, до замовлення на вибір продуктового продукту.

Conectivity - Поняття мереж не завжди має обмежуватися великими мережами, воно також може існувати в меншому та дешевшому масштабі, не заважаючи його ефективності. IoT виходить на малюнок і створює ці маленькі мережі між своїми системними пристроями.

Sensors - Справжня суть IoT не була б ефективною без датчиків. Вони в основному є причиною і сутністю того, чому ця технологія виділяється. Вони відіграють головну роль у визначенні меж IOT шляхом перетворення її з пасивної в активну мережу.

Active Engagement - Сьогоднішня взаємодія між різними технологіями, що пов'язані між собою, відбувається за допомогою пасивної взаємодії. IoT подав приклад, залучивши активний вміст, продукт чи послугу.

Devices - пристрої з часом стають більш потужні, дешевші та менші, Інтернет речей спеціально використовує невеликі пристрої, щоб забезпечити його масштабність, універсальність та точність. [2]

Історія IoT

Кевін Ештон, співзасновник Центру автоматичної ідентифікації в MIT, вперше згадав про Інтернет речей у презентації, яку він зробив в Procter & Gamble (P&G) у 1999 році. Бажаючи довести радіочастотний ідентифікатор (RFID) до відома P&G вищого керівника, Ештон назвав свою презентацію "Інтернет речей", щоб включити новий тренд 1999 року: Інтернет речей. У книзі професора MIT Ніла Гершенфельда «Коли починають думати речі», яка також з’явилася в 1999 році, не було використано точного терміна, але було чітке бачення того, куди рухається IoT. IoT розвинувся завдяки конвергенції бездротових технологій, мікроелектромеханічних систем (MEMS), мікросервісів та Інтернету. Конвергенція допомогла знищити межу між операційною технологією (OT) та інформаційною технологією (IT), що дозволило аналізувати неструктуровані машинно-сформовані дані для розуміння, щоб досягнути вдосконалень. Незважаючи на те, що Ештон вперше згадав про Інтернет-речі, ідея підключення пристроїв існує ще з 1970-х років, коли були широко поширені обчислювальні процеси.

IoT розвивався від зв'язку від машини до машини (M2M), тобто машин, що з'єднувались одна з одною через мережу без взаємодії людини. M2M відноситься до підключення пристрою до хмари, управління ним та збору даних. Піднімаючи M2M на наступний рівень, IoT - це сенсорна мережа з мільярдів розумних пристроїв, яка з'єднує людей, системи та інші програми для збору та обміну даними. В якості основи M2M пропонується підключення, що дозволяє IoT.

Інтернет речей - це також природне розширення SCADA (наглядовий контроль та збирання даних), категорії програмного забезпечення для управління процесами, збирання даних у режимі реального часу з віддалених місць для контролю обладнання. Системи SCADA включають апаратні та

програмні компоненти. Апаратне забезпечення збирає та передає дані в комп’ютер, на якому встановлено програмне забезпечення SCADA, де вони потім обробляються. Еволюція SCADA полягає в тому, що системи SCADA пізнього покоління перетворилися на системи IoT першого покоління. Однак концепція екосистеми IoT насправді не стала власною до середини 2010 року, коли, частково, уряд Китаю заявив, що зробить IoT стратегічним пріоритетом у своєму п'ятирічному плані. [3]

Екосистема IoT складається з розумних пристроїв з підтримкою Інтернету, які використовують вбудовані процесори, сенсори та комунікаційне обладнання для збору та надсилання даних, які вони отримують із свого середовища. Пристрої IoT діляться даними, які вони збирають, підключаючись до шлюзу IoT або іншого крайового пристрою, де дані надсилаються в хмару для аналізу або локального аналізу. Іноді ці пристрої спілкуються з іншими пов'язаними пристроями та діють на інформацію, яку вони отримують один від одного. Пристрої виконують більшу частину роботи без втручання людини, хоча люди можуть взаємодіяти з пристроями - наприклад, налаштувати їх, дати їм інструкції або отримати доступ до даних.

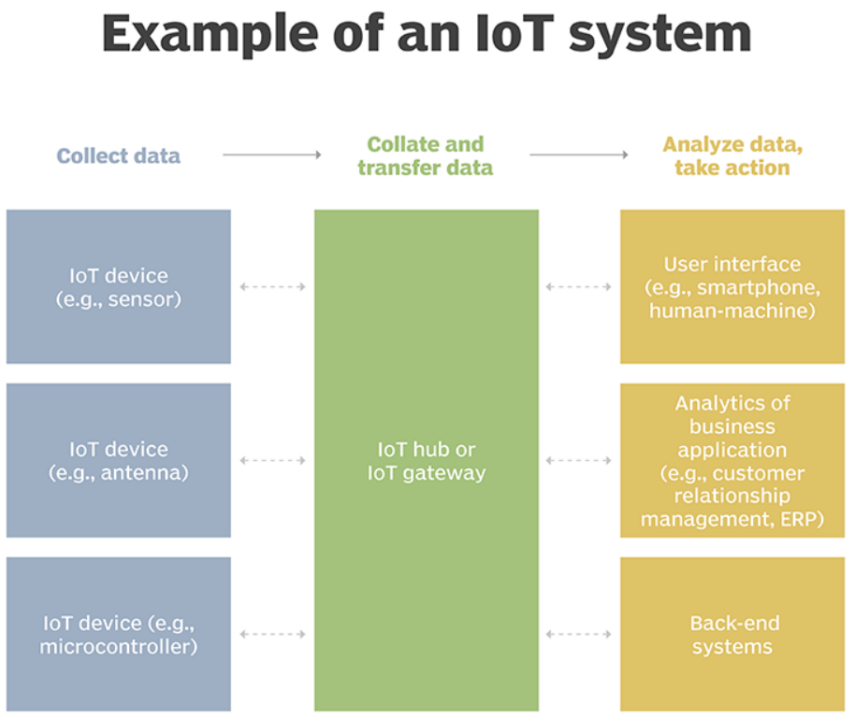

Рисунок 1.2 – Приклад використання IoT

"Речі", які складають IoT, можуть бути будь-якими - від носячих фітнес-трекерів до автономного транспортного засобу. Незалежно від того, яку функцію вони виконують для користувачів, ці пристрої повинні мати такі компоненти, щоб вони належним чином працювали як частини відповідних систем IoT.

Датчики. Дані спочатку збираються із середовища для того, щоб система IoT почала обробку. Він збирається датчиками в пристроях, які можуть вимірювати спостережувані явища або зміни в навколишньому середовищі. Вид даних, що вимірюються пристроєм, залежить від його функції: це може бути імпульс людини у випадку фітнес-трекера або відстань найближчого об'єкта до автономного транспортного засобу.

Підключення та ідентифікація. Дані повинні бути передані з пристрою іншій системі IoT, будь то на комп’ютер або на інший пристрій. І щоб ця комунікація мала будь-яке значення, пристрій повинен мати унікальну присутність в Інтернеті, що здійснюється через власну IP-адресу.

Приводи. Більшість пристроїв IoT здатні виконувати свої основні функції без фізичної взаємодії зі своїми користувачами. Пристрої IoT повинні мати можливість приймати дії на основі даних своїх датчиків та подальших зворотних зв’язків з мережі. Наприклад, розумна лампочка може вмикатися за командою свого користувача, навіть коли користувач знаходиться на відстані. Таким же чином клапан на розумній фабриці може автоматично відкриватися або закриватися відповідно до даних, зібраних його датчиками по лінії виробництва.

Навіть незважаючи на те, що пристрої зазвичай побудовані з урахуванням автоматизації, для роботи IoT-систем повинні бути створені інші технології. Завершення посилань того, як системи IoT обробляють дані, є наступними компонентами.

Шлюз IoT. Шлюз IoT діє як міст для даних різних пристроїв до хмари.

Це також допомагає в перекладі різних протоколів різних пристроїв IoT лише в один стандартний протокол і в фільтруванні непотрібних даних, зібраних пристроями.

Хмара. Хмара - це місце, де збираються всі дані з різних пристроїв і де програмне забезпечення може отримати ці дані для обробки. Оскільки більша частина обробки даних відбувається в хмарі, це зменшує навантаження на окремі пристрої.

Користувацький інтерфейс. Інтерфейс користувача повідомляє користувачам дані, зібрані пристроями, і дозволяє користувачам робити необхідні команди для виконання пристроями.

Рада з архітектури Інтернету оприлюднила керівний документ, у якому викладено чотири канали зв'язку, які використовує IoT. Чотири моделі також демонструють, як підключення пристроїв IoT допомагає розширити значення кожного пристрою та додає якість до загальної роботи користувачів:

Пристрій-до-пристрою. Ця модель відображає, як два чи більше пристроїв з'єднуються та спілкуються безпосередньо між собою. Зв'язок між пристроями зазвичай досягається за допомогою таких протоколів, як Bluetooth, Z-Wave та Zigbee. Ця модель часто зустрічається в носячих предметах і в пристроях домашньої автоматики, де невеликі пакети даних передаються від одного пристрою до іншого.

Пристрій до хмари. Багато пристроїв IoT підключаються до хмари, часто за допомогою дротового Ethernet або Wi-Fi. Підключення до хмари дозволяє користувачам та пов’язаним програмам отримувати доступ до пристроїв, завдяки чому можна віддалено переглядати команди, а також надсилати необхідні оновлення програмного забезпечення пристрою. За допомогою цього з'єднання пристрої також можуть збирати дані користувачів для вдосконалення своїх постачальників послуг.

Пристрій до шлюзу. Перш ніж підключитися до хмари, пристрої IoT можуть спочатку зв’язатися з посередницьким пристроєм шлюзу. Шлюз може перекладати протоколи та додавати додатковий рівень безпеки для всієї

системи IoT. Наприклад, у випадку розумного будинку, наприклад, усі розумні пристрої можуть бути підключені до концентратора (шлюзу), який допомагає різним пристроям працювати разом, незважаючи на наявність різних протоколів підключення.

Обмін даними заднім числом. Ця модель - це розширення моделі «від хмари до пристрою» та дає змогу користувачам отримати доступ та аналізувати колекцію даних з різних розумних пристроїв. Наприклад, компанія може використовувати цю модель для доступу до інформації з усіх пристроїв, що працюють всередині будівлі компанії, як організовано разом у хмарі. Ця модель також допомагає зменшити проблеми з переносністю даних.

Інтернет речей допомагає людям жити та працювати, а також отримувати повний контроль над своїм життям. Крім використання розумних пристроїв для автоматизації будинків, IoT є важливим для бізнесу. IoT надає бізнесу бізнесу можливість, в режимі реального часу, вивчити, як дійсно працюють системи їхніх компаній, даючи зрозуміти все: від продуктивності машин до ланцюгових операцій та логістичних операцій. IoT дозволяє компаніям автоматизувати процеси та зменшити витрати на оплату праці. Це також скорочує відходи та покращує надання послуг, роблячи дешевшим виготовлення та доставку товарів, а також пропонуючи прозорість угод із клієнтами. IoT торкається будь-якої галузі, включаючи охорону здоров'я, фінанси, роздріб та виробництво. Розумні міста допомагають громадянам зменшити споживання енергії, а підключені датчики навіть використовуються у сільському господарстві, щоб допомогти контролювати врожайність та прогнозувати тенденції зростання.

Таким чином, IoT є однією з найважливіших технологій повсякденного життя, і вона буде продовжувати набирати оберти, оскільки більше підприємств усвідомлюють потенціал підключених пристроїв, щоб підтримувати їх конкурентоспроможними.

Існує кілька нових IoT стандартів, серед яких:

Bluetooth, який став дуже важливим для обчислень та багатьох ринків

споживчих товарів. Очікується, що він буде ключовим, зокрема, для носячих продуктів, і знову підключатиметься до IoT, хоча, ймовірно, через смартфон у багатьох випадках. Новий Bluetooth з низьким рівнем енергії (BLE) - або Bluetooth Smart, як зараз він називається - є важливим протоколом для програм IoT. Що важливо, хоча він пропонує аналогічний діапазон для Bluetooth, він був розроблений, щоб запропонувати значно скоротити енергоспоживання.

6LoWPAN (IPv6 через бездротові персональні мережі з низькою потужністю), відкритий стандарт, визначений спеціальною групою IETF. Стандарт 6LoWPAN дає змогу будь-якому радіозв'язку малої потужності спілкуватися в Інтернеті, включаючи 804.15.4, Bluetooth з низькою енергією Bluetooth та Z-Wave (для домашньої автоматизації).

ZigBee0 - бездротова мережа з низькою швидкістю передачі даних, що використовується переважно в промислових умовах. ZigBee заснований на базі стандарту IEEE 802.15.4. Альянс ZigBee створив Dotdot, універсальну мову для IoT, яка дозволяє розумним об’єктам надійно працювати в будь-якій мережі та розуміти один одного.

LiteOS, Unix-подібна операційна система для бездротових сенсорних мереж. LiteOS підтримує смартфони, інтелектуальні виробничі програми, розумні будинки та Інтернет транспортних засобів (IoV). Операційна система також служить платформою для розробки розумних пристроїв.

OneM2M - рівень обслуговування від машини до машини, який можна вбудувати в програмне та апаратне забезпечення для підключення пристроїв. Організатор глобальної стандартизації, OneM2M, був створений для розробки стандартів багаторазового використання, щоб забезпечити комунікації додатків IoT в різних вертикалях.

DDS (служба розподілу даних) була розроблена групою управління об'єктами (OMG) і є стандартом IoT для зв'язку в режимі реального часу, масштабованого та високоефективного зв'язку між машинами.

AMQP (Advanced Message Queuing Protocol), відкритий вихідний код, опублікований стандарт для асинхронних повідомлень проводом. AMQP

дозволяє зашифровувати та сумісно обмінюватися повідомленнями між організаціями та програмами. Протокол використовується в обміні повідомленнями клієнт / сервер та в керуванні пристроями IoT.

LoRaWAN - одна з популярних IoT-технологій, орієнтована на широкосмугові мережі (WAN). Конструкція LoRaWAN забезпечує WAN з низькою потужністю з особливостями, необхідними для підтримки дешевого мобільного безпечного зв'язку в IoT в розумному місті та промислових додатках. Спеціально відповідає вимогам щодо низького енергоспоживання та підтримує великі мережі з мільйонами та мільйонами пристроїв, швидкість передачі даних становить від 0,3 кбіт до 50 кбіт / с.

Z-Wave - це технологія IoT радіочастотного зв'язку з низькою потужністю, яка в першу чергу призначена для домашньої автоматизації для таких виробів, як контролери ламп та датчики серед багатьох інших пристроїв. Z-Wave використовує більш простий протокол, ніж деякі інші, який може забезпечити швидший і простіший розвиток, але єдиним виробником чіпів є Sigma Designs порівняно з декількома джерелами для інших бездротових технологій, таких як ZigBee та інших. [4]

Платформи для використання IoT:

AWS IoT, хмарна платформа для IoT, випущена Amazon. Цей фреймворк створений для того, щоб розумні пристрої могли легко підключатися та безпечно взаємодіяти з хмарою AWS та іншими підключеними пристроями.

ARM Mbed IoT, платформа для розробки додатків для IoT на базі мікроконтролерів ARM. Метою платформи ARM Mbed IoT є створення масштабованого, підключеного та безпечного середовища для пристроїв IoT шляхом інтеграції інструментів та служб Mbed.

Майкрософт Azure IoT Suite, платформа, що складається з набору сервісів, що дозволяє користувачам взаємодіяти та отримувати дані зі своїх пристроїв IoT, а також виконувати різні операції над даними, такі як багатовимірний аналіз, перетворення та агрегація та візуалізувати ці операції в спосіб, який підходить для бізнесу.

Brillo / Weave від Google - платформа для швидкого впровадження програм IoT. Платформа складається з двох основних опорних систем: Brillo, операційної системи на основі Android для розробки вбудованих пристроїв малої потужності; і Weave, орієнтований на IoT протокол зв’язку, який служить мовою зв'язку між пристроєм та хмарою.

Calvin, платформа IoT з відкритим кодом, випущена Ericsson, призначена для створення та управління розподіленими програмами, що дозволяють пристроям спілкуватися один з одним. Calvin включає структуру розробки для розробників додатків, а також середовище виконання для роботи з запущеною програмою.

1.2 Сфери застосування Інтернету речей

Існують численні додатки Інтернет-речей у реальному світі, починаючи від споживчого IoT та корпоративного IoT до виробничого та промислового. Програми IoT охоплюють численні вертикалі, включаючи автомобільну, телекомунікаційну та енергетичну галузі.

Статистика показує, що кількість підключених пристроїв у 2018 році становить 23+ мільярдів. Це майже в 3 рази перевищує щільність населення Землі, яка на момент написання цього блогу становила 7,6 мільярда. Крім того, ця кількість збільшується і прогнозується, що сягне 75+ мільярдів у наступному десятилітті:

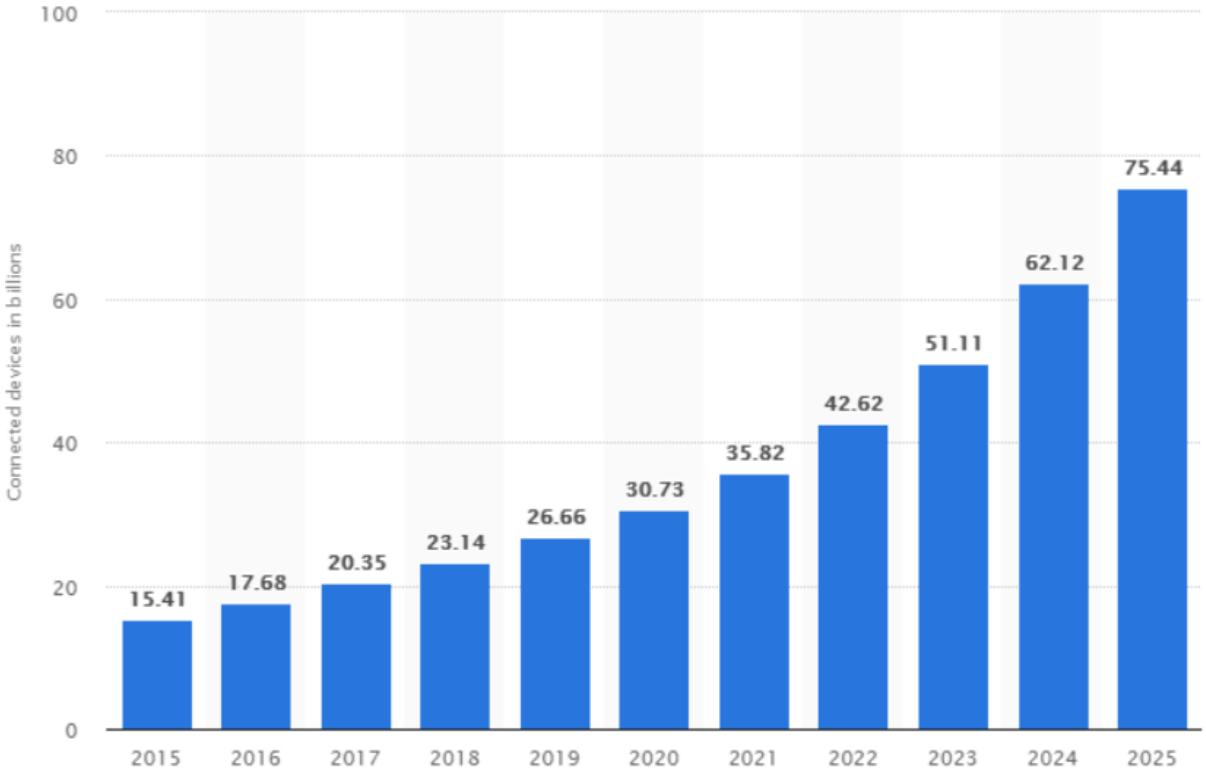

Рисунок 1.3 – Прогноз підключених пристоїв

Accenture вважає, що IoT ринок до 2020 року складе $ 14,2 трлн. IDC прогнозує витрату 1,2 мільйона доларів на бізнес в Інтернет речі. Bain прогнозує, що ринок зросте до приблизно 520 доларів США в 2021 році. [5]

Застосування IoT у транспорті

а. Автомобіль

Більшість компаній в автомобільному секторі почали передбачати майбутнє для двигунів, в яких епоха IoT робить транспортні засоби «розумними», привабливі альтернативи, ідентичні MRT, IoT дає незначні вдосконалення приватних автомобілів. Максимальні переваги отримують від кращого контролю за пов'язаною інфраструктурою та властивими вадами транспорту транспортних засобів. Однак IoT робить покращення приватних двигунів як непублічних просторів. IoT приносить ідентичні удосконалення та налаштування автомобіля, як і всередині будинку.

б. Автономні транспортні засоби

Автономний транспортний засіб, який іноді називають автомобілями, що керують автотранспортом, або автомобілями без водіїв - це транспортний засіб, який використовує поєднання датчиків, камери, радара та штучного інтелекту (AI) для подорожі між пунктами призначення без оператора людини. Для того, щоб вважати себе повністю автономним, транспортний засіб повинен мати можливість переміщатися, не потребуючи втручання людини до заздалегідь визначеного пункту призначення через дороги, які не були пристосовані для його використання.

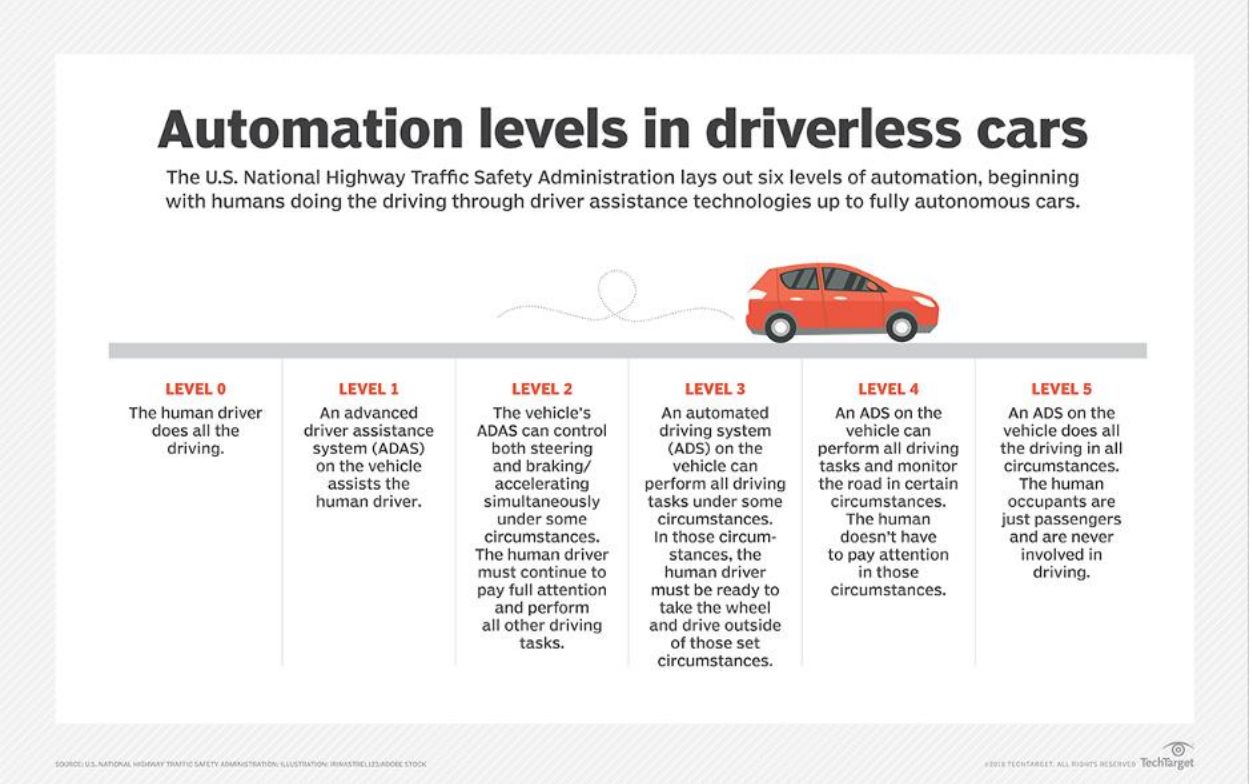

Рисунок 1.4 – Процес автоматизації в автономних транспортних засобів

AI технології живлення автомобільних систем, що керують автомобілем. Розробники автономних транспортних засобів використовують величезну кількість даних із систем розпізнавання зображень разом з нейронними мережами та машинним навчанням. Перетин цих різних полів призводить до того, що можна будувати системи, які можуть керувати автономно. Нейронні мережі ідентифікують закономірності даних, отриманих за допомогою

датчиків та різного іншого обладнання, яке передається алгоритмам машинного навчання. Одним з відомих прикладів є автомобіль Google, що управляє авто, що використовує поєднання датчиків, а потім поєднує всі дані, створені цими системами, щоб ідентифікувати об'єкти навколо транспортного засобу.

в. Затори

Проблема перевезень номер один - це перевантаження заторів, зменшення аварій та паркування. IoT дозволяє нам краще оглянути та проаналізувати рух по потоку відвідувачів через гаджети в будь-якому відношенні факторів коментування трафіку. Це допомагає при парковці за допомогою зробити очевидне ковзання під час зберігання, тоді як сучасні методи пропонують мало, якщо є якась статистика. Аварії зазвичай закінчуються наслідком більш ніж декількох факторів, але управління відвідувачами впливає на їх частоту. Виробничі майданчики, погана маршрутка та відсутність фактів про репутацію дорожнього руху - все це проблеми, які призводять до інцидентів. IoT пропонує рішення у формі вищого обміну фактами з широкою громадськістю та серед різноманітних подій, що безпосередньо впливають на відвідувачів дорожнього руху.

г. Промисловий транспорт

Переваги на транспорті поширюються на комерційне підприємство та виробництво за допомогою оптимізації транспортної галузі бізнесу. Це зменшує та усуває неприємності, пов’язані з поганим контролем автопарку, завдяки кращій аналітиці та контролю разом із моніторингом холостого ходу, споживанням бензину, ситуаціями з подорожами та часом подорожі серед факторів. Це призводить до того, що транспортування товару працює більше, як узгоджений постачальник, і менше, як група пропозицій, що укладаються за контрактом.

Рисунок 1.5 – Використання IoT в промисловому транспорті

Застосування IoT в медицині

У галузі охорони здоров’я IoT пропонує безліч переваг, включаючи можливість більш ретельного моніторингу пацієнтів за використанням даних, що генеруються, а також їх аналізу. Лікарні часто використовують системи IoT для виконання завдань, таких як управління запасами, як для фармацевтичних препаратів, так і для медичних інструментів.

Технологія Internet of Things (IoT) стає все більш поширеною в галузі охорони здоров'я. Основне застосування IoT в галузі інтелектуальної медицини включає візуалізацію управління матеріалами, оцифрування медичної інформації та оцифрування медичних процесів. За допомогою RFID, IoT почав знаходити більш широке застосування у галузі візуалізації управління медичними матеріалами. IoT з RFID може допомогти уникнути проблем із охороною здоров'я, допомагаючи у виробництві, розповсюдженні та відстеженні медичних виробів та ліків. Це підвищує якість медикаментозного лікування, зменшуючи при цьому витрати на управління.

За даними Всесвітньої організації охорони здоров'я (ВООЗ), кількість підроблених лікарських засобів у світі становить понад 10 відсотків продажу наркотиків у всьому світі. Дані Китайської фармацевтичної асоціації показують, що лише в Китаї щорічно помирає щонайменше 200 000 людей внаслідок неправильних або неправильно використовуваних ліків. Від 11 до 26 відсотків пацієнтів вживають ліки неправильно. Сюди входить близько 10 відсотків неправильно призначених ліків. Тому технологія RFID відіграватиме важливу роль у відстеженні та моніторингу лікарських засобів та обладнання та регулюванні ринку медичних виробів.

Етикетка, прикріплена до товару, матиме унікальну ідентичність, яку надзвичайно важко підробити. Це відіграватиме важливу роль у перевірці інформації та боротьбі з підробкою, підтверджуючи ефективний контрзахід проти медичних шахрайств. Наприклад, можна буде передати інформацію про наркотики до загальнодоступної бази даних, з якої пацієнт або лікарня можуть перевірити вміст етикетки на предмет записів у базі даних, щоб легко визначити потенційні підробки.

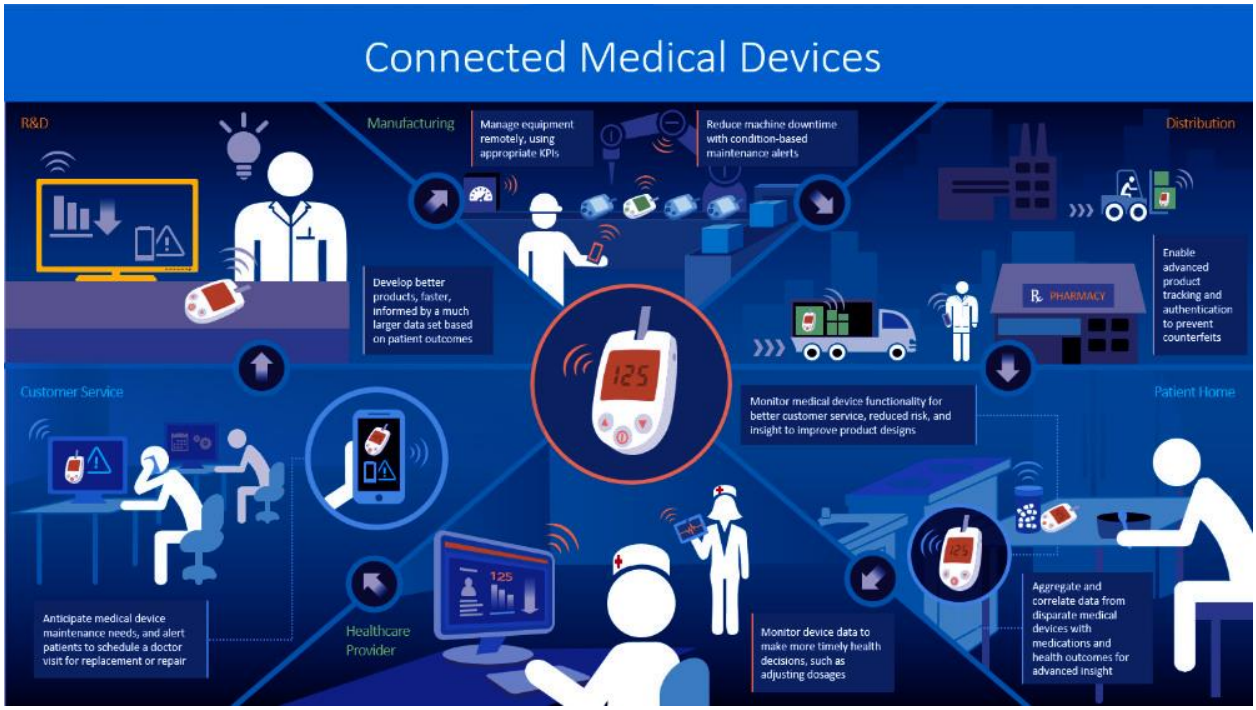

Рисунок 1.6 – Використання IoT в медицині

Інтернет речей має широкі перспективи застосування в галузі управління медичною інформацією. В даний час попит на управління медичною інформацією в лікарнях полягає переважно в таких аспектах: ідентифікація, розпізнавання зразків та ідентифікація медичних записів. Ідентифікація включає ідентифікацію пацієнта, ідентифікацію лікаря, ідентифікацію проби (включаючи ідентифікацію лікарського засобу), ідентифікацію медичного обладнання, лабораторну ідентифікацію та ідентифікацію медичної документації (включаючи симптоми та ідентифікацію захворювання). У ресурсах, якими використовуються сучасні медичні дослідження, бракує критичної інформації в реальному світі. Він здебільшого використовує залишки, контрольоване середовище та добровольців для медичного обстеження.

Служби екстреної допомоги завжди мали проблеми через обмежені ресурси. Сучасна автоматизація та аналітика IoT вирішує цю проблему в галузі охорони здоров’я. Надзвичайну ситуацію можна проаналізувати з великої відстані, а точніше за кілометри. Служби отримують доступ до профілів пацієнтів до їх приїзду, завдяки чому вони можуть вчасно надати основну допомогу пацієнтам. Таким чином зменшуються пов'язані з цим втрати, і покращується швидка медична допомога.

Застосування IoT в освіті

Одним із дуже розумних компонентів сучасних коледжів та аудиторій є те, що IoT покращує навчання в школі та приносить додаткову плату фізичному оточенню та системам. Розумний коледж легко працює та забезпечує кращий етап особистого знайомства. Розумні гаджети, які використовуються в кампусі, використовують спільноту wi-fi для передачі фактів та отримання команд. Обчислювальний Інтернет-гаджет із речей для факультетів та навчальних закладів дає змогу створювати розумніші плани уроків, підтримувати мелодію критичних ресурсів, покращує запис про вступ, проектувати безпечніші кампуси та багато іншого.

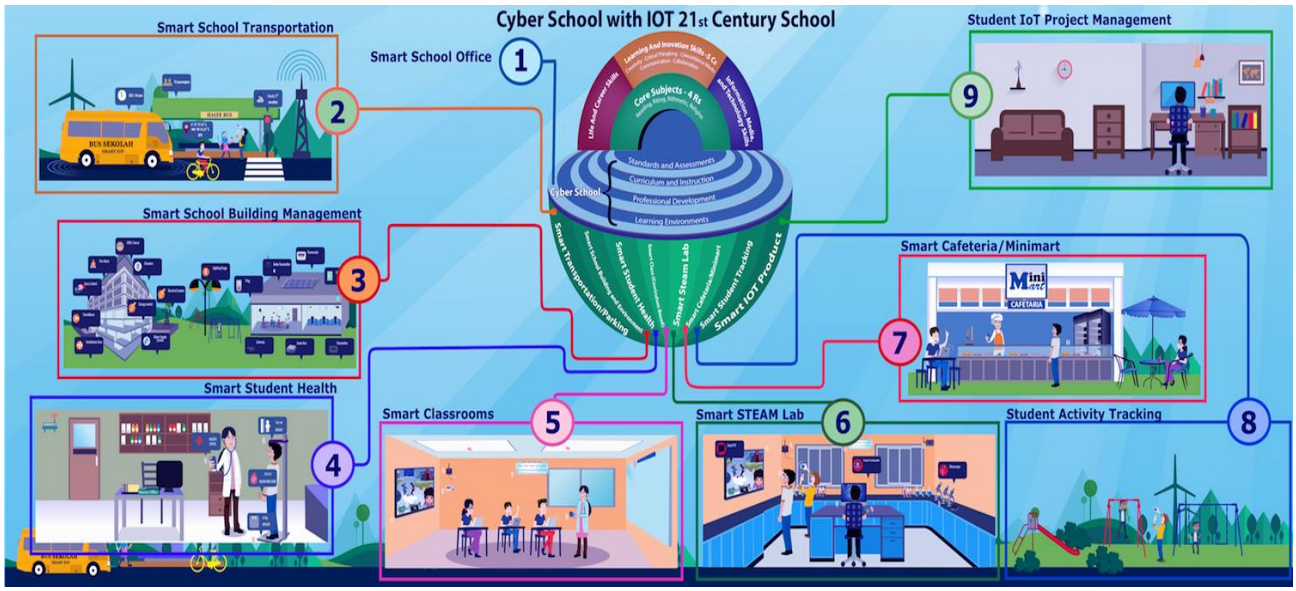

Рисунок 1.7 – Використання IoT в навчальному процесі

а. Дошки з підтримкою IoT

Справді дуже важко порівнювати презентаційні дошки старшої епохи з сучасними дошками для мультимедійних плакатів. Інтернетна передача на зразок Glogster змінила цю простоту і дозволяє нам без проблем створювати цифрові плакати, поєднуючи фотографії, аудіо, відео, текст та гіперпосилання. Це дозволяє нам їх електронне співвідносити в електронному вигляді з іншими та без проблем розкривати активність науковця. Ці віртуальні плакати потім можуть бути надіслані однокласникам та викладачам електронною поштою, надіслати доступ до них через URL-адресу плаката та розмістити їх у блогах елегантності. Останнім часом студенти використовують дуже потужну платформу, яка включає розумні дошки. Це полегшує лекторів без проблем пояснювати лекції за допомогою онлайн-показів та фільмів.

б. Інтерактивне здобуття знань

Знайомство в ці дні не обмежується сумішшю текстів та картинок, а поза цим. Більшість підручників у поєднанні з веб-сайтами, заснованими на мережі, які складаються із зайвих речовин, фільмів, іспитів, анімацій та різних речовин для підтримки освоєння. Це дає студентам більш широкий кругозір для аналізу нових речей з кращим розумінням та взаємодією з викладачами та їхніми

друзями. Професіонали-інструктори доставляють справжні світові неприємності всередині навчального кабінету і дозволяють студентам знайти власні відповіді.

в. Навчання в будь-який час і в будь-якому місці

IoT відіграє важливу позицію в побудові мережі за допомогою спеціальних Інтернет-систем. Передові технології дозволяють науковцям відображати розвиток науковців. Едмодо - це чудовий спосіб словесного обміну тренера-учня. Edmodo дає можливість новачкам в будь-який час користуватися інформацією з будь-якого місця. IoT дозволяє учням та викладачам спілкуватися за допомогою надзвичайного методу, перевіряючи повідомлення та майбутні події одночасно, коли вони перебувають далеко від аудиторії чи навіть відповідають на повідомлення. Це далеко не дуже ефективний додаток, який забезпечує безпечну мережу та повну конфіденційність. Це також дозволяє користувачеві зберегти ваші конкретні думки та заняття, не турбуючись, та запевнить вас у повній конфіденційності.

г. Функції безпеки

Це застосування IOT в освіті є важливим, оскільки використання найкращих технологій відповідей у шкільних кімнатах та навчальній зоні може бути дуже корисним. Він включає в себе індикатори надзвичайних ситуацій, покращення звуку, годинник Wi-Fi та сповіщення з порушеннями слуху, які пропонують науковцям та тілу працівників почуття безпеки. Він також здатний зменшити спустошення і зберігати життя, що може спричинити за собою катастрофічний стан справ. Коледжі та навчальні центри приймають конкретні заходи безпеки, які сприяють розслабленню містечок. Система зв’язку, що підтримує IoT, також може використовуватися для різних випадків, таких як спеціальні тони аварійних ситуацій, живі бюлетені, пара розкладу дзвонів та попередньо записані інструкційні повідомлення, щоб направити групу працівників та студентів у певний момент надзвичайної ситуації. [6]

Застосування IoT в урядовому сегменті

IoT допомагає покращити розумні міжнародні місця та розумні міста. Сюди входить розширення інфраструктури, про яку вже йшлося раніше (наприклад, охорона здоров'я, транспорт тощо), захист, а також інженерія та підтримка громад.

а. Національна оборона

Національні загрози, з якими стикається країна можуть бути дуже різного ступеня і складності. IoT вдосконалює та підтримує системи та служби міліції та надає важливу технологію для контролю панорами національної оборони. Це допомагає підвищити безпеку кордонів за рахунок дешевших, кращих продуктивних пристосувань, керованих та чудових. IoT автоматизує обов'язки з безпеки, які, як правило, розгортаються у численних підрозділах та у багатьох людей. Це досягає, підвищуючи точність та швидкість.

б. Розумні міста

Розумні міста - це групи, які використовують територію для перетворення тілесних структур та служб таким чином, що доповнює існування її мешканців та бізнесу одночасно, а також робить владу надзвичайно ефективною. Це набагато більше, ніж проста автоматизація процесів; також гіперпосилання розрізнені структури та мережі для збору та аналізу фактів. Ці факти потім використовуються для перетворення цілих систем. Незважаючи на те, що концепція розумних міст існує протягом багатьох років, вона набула нової актуальності, оскільки зайві люди поширюються в міста. Сталий розвиток перетворюється на суттєве та імперативне. Технологія певною мірою просунулася, коли може спостерігатися взаємодія між містами, громадянами та агенціями в режимі реального часу.

в. Планування та контроль міста

Органи управління та інженери можуть використовувати IoT для аналізу часто складних факторів планування та контролю містобудування. IoT спрощує це, вивчаючи різні фактори, які включають збільшення чисельності населення, зонування, картографування, доставку води, схеми

транспортування, доставку їжі, соціальні пропозиції та землекористування. Він збирає позначені факти в цих регіонах і виробляє більш дорогоцінні та точні записи, ніж сучасна аналітика, враховуючи її здатність по-справжньому «жити» з людьми в мегаполісі. Всередині регіону управління IoT підтримує міста завдяки впровадженню основних послуг та інфраструктури, що включають транспорт та охорону здоров'я. Він також допомагає в інших ключових сферах, таких як контроль води, поводження з відходами та поводження з надзвичайними ситуаціями.

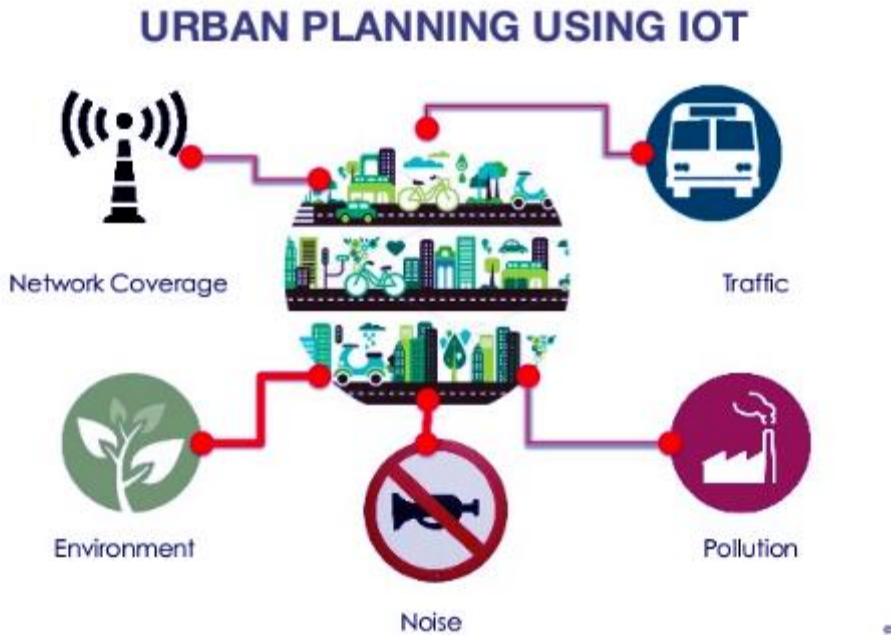

Рисунок 1.8 – Контоль міста за допомогою IoT

г. Створення робочих місць

Це одне з найважливіших урядових додатків в IoT. IoT пропонує ретельну економічну оцінку. Це робить попередні незрозумілі плями та сприяє кращому відстеженню та моделюванню грошей. Він аналізує галузь та ринок, щоб виявити можливості для збільшення та перешкод.

д. Побудова екосистеми для безпеки води

Епоха IoT може вирішити складні виклики, пов'язані з безпекою води, дозволяючи урядам краще визначити пріоритети щодо водопостачання, закликів споживачів та управління. Як і інші проблеми, зумовлені

множинними та різними факторами, для покращення результатів управління водними ресурсами потрібні внески оточуючих супутників, багато з яких навіть не знають, яку роль вони відіграють у збереженні води.

е. Швидке реагування на надзвичайні ситуації

Програми IoT можуть дуже швидко вивчати дані про такий випадок, допомагаючи респондентам краще усвідомлювати інциденти, визначати спосіб реагування та приймати рішення (та критичні кроки) для тих, хто бере участь. Екологічні датчики, як приклад, можуть входити та повідомляти про ранні ознаки надзвичайної ситуації чи злочину. Вже зараз пристрої, до яких належить ShotSpotter, можуть наткнутися на звук вогнепального пострілу та визначити його місце. За допомогою автоматичного оповіщення про відправлення поліції інструмент може визначити час реакції на швидкість, крім того, зменшити залежність від свідків щодо документування злочину, допомагаючи виявити злочини. [7]

Застосування IoT в промисловості

Індустріальний Інтернет речей - це постійно зростаючий і швидко зростаючий сектор, на який припадає більшість частки витрат IoT на світовому ринку. Промисловці та виробники майже в кожному секторі мають надзвичайну можливість не лише контролювати. Але також автоматизувати багато складних процесів, що беруть участь у виробництві. Протягом тривалого часу у галузях і заводах були сенсори та системи для відстеження прогресу, але IoT робить крок вперед і надає тонкощі навіть найменших проблем.

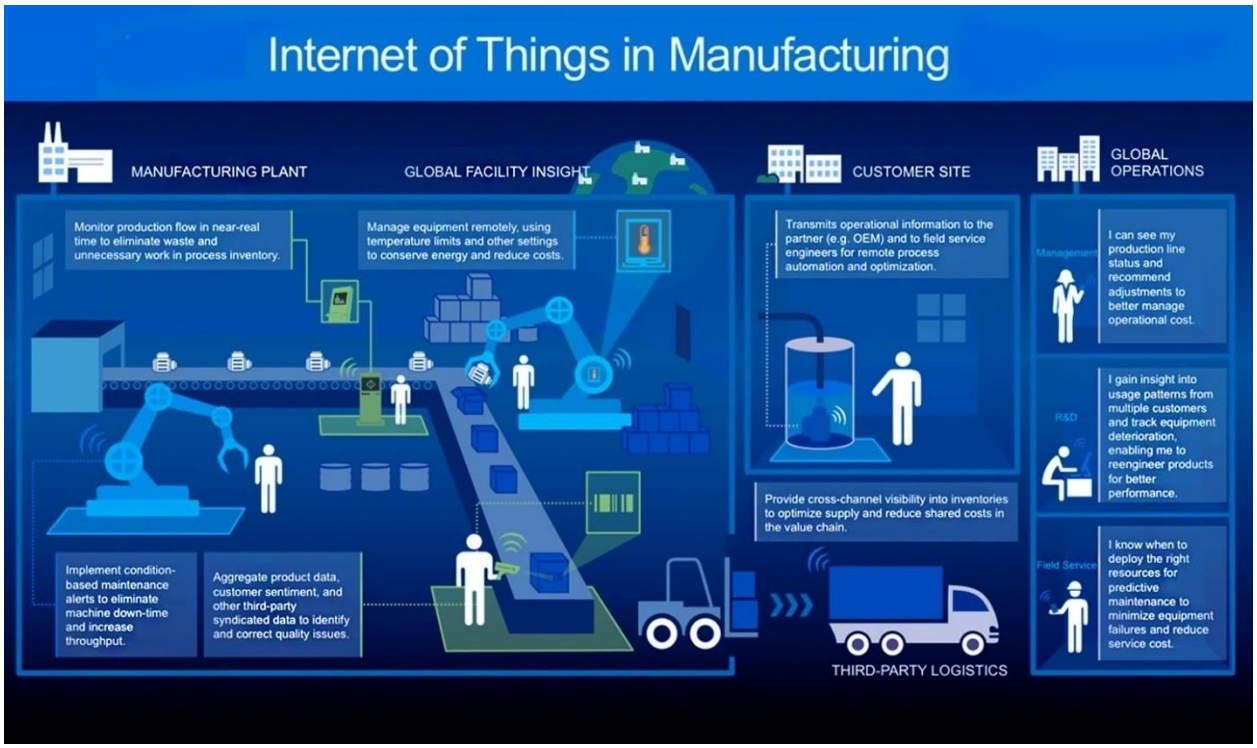

Рисунок 1.9 - Застосування IoT в промисловості

а. Цифрова фабрика

Машини, вбудовані в систему IoT, можуть передавати інформацію, пов'язану з операціями, людям, таким як виробники оригінального обладнання, та інженерам. Таким чином, автоматизація та оптимізація технологічних процесів є вигідною завдяки можливості керівників операцій та керівників заводів віддалено керувати заводськими підрозділами. Поряд з цим, підрозділ, який пов'язаний цифровим шляхом, допомагає встановити кращу командну команду, а також допомагає визначити області з ключовими результатами та сфери, які можуть мати потенційні проблеми для керівників.

б. Об'єкт управління

Датчики IoT, розміщені всередині виробничого обладнання, спрацьовують на основі гарантії технічного обслуговування. Більшість верстатів є критичними і призначені для функціонування між певними температурними та вібраційними діапазонами. Щоразу, коли обладнання відхиляється від встановлених параметрів, датчики IoT можуть активно

контролювати машини та надсилати попередження. Таким чином виробники можуть заощадити енергію, зменшити витрати, усунути час простою машини та підвищити експлуатаційну ефективність, забезпечуючи встановлене робоче середовище для машин.

в. Моніторинг виробничих потоків

г. Управління запасами

Це найкращий промисловий додаток IoT, завдяки системам IoT проводиться моніторинг подій по ланцюгу поставок. Ці системи дозволяють відстежувати інвентар та відстежувати його в усьому світі на рівні позиції. Таким чином користувачі отримують сповіщення, якщо є якісь значні відхилення від плану дій. Як результат, це забезпечує далекосяжну та багатоканальну видимість запасів, що допомагає керівникам отримувати реальні оцінки наявного матеріалу, незавершеного виробництва та передбачуваного часу прибуття нових матеріалів. Зрештою, це забезпечує постачання більш оптимальним та зменшує додаткові та спільні витрати, що виникають у ланцюжку вартості.

д. Безпека

IoT в безпеці працівників на підприємстві покращуються завдяки IoT у поєднанні з аналізом великих даних. Система IoT відстежує деякі ключові показники ефективності здоров'я та безпеки, такі як кількість травм, часті випадки захворювань, ДТП та майнові збитки або будь-які збитки, понесені під час щоденних операцій. Таким чином, ефективна система моніторингу забезпечує кращу та ефективну безпеку. Якщо є деякі показники, які відстають, вони вирішуються, таким чином забезпечуючи кращі проблеми в галузі охорони здоров’я, безпеки та навколишнього середовища (HSE). Це не може ігнорувати промислові програми IoT.

е. Контроль якості

Цикл продукції має різні етапи, датчики IoT збирають суміш даних про продукцію та інші сторонні синхронізовані дані на етапах продуктового циклу. Ці дані містять інформацію про склад сировини, що використовується для

виготовлення виробу, температуру та робоче середовище, різні відходи, важливість транспортування тощо. Про кінцеве виготовлення продукції. Крім того, пристрій IoT також може надавати дані про настрої клієнтів, коли він / вона використовує продукт. Усі ці дані з різних джерел та через системи IoT можуть аналізувати, щоб виявити та виправити потенційні проблеми якості.

є. Оптимізація упаковки

Виробники можуть отримати уявлення про схеми використання та поводження з продуктом у різних клієнтів, використовуючи датчики IoT, вбудовані в продукцію та / або упаковку. Існують розумні механізми відстеження, які дозволяють простежити погіршення продукту під час транзиту товару. Інші фактори, такі як вплив погоди, стан доріг та інші змінні середовища на продукт. Завдяки цим розумінням можна перепрофілювати продукти та їх упаковку, щоб забезпечити кращі показники як витрат на упаковку, так і досвіду клієнтів.

ж. Логістика та оптимізація ланцюгів поставок

У цьому промисловому додатку IoT він забезпечує доступ до інформації про ланцюги постачань у реальному часі, відстежуючи матеріали в дорозі, продукти та обладнання під час їх переміщення по ланцюгу поставок. Завдяки ефективному звіту виробники мають змогу збирати та передавати інформацію про доставку в такі системи, як ERP, PLM тощо. Якщо заводи підключаються до постачальників, усі зацікавлені сторони в ланцюзі поставок можуть відстежувати взаємозалежності, тривалість виробничого циклу та матеріальний потік. Як результат, ці дані допоможуть виробникам скоротити запаси, передбачити потенційні проблеми, а також знизять вимоги до капіталу.

2 ФУНКЦІОНАЛЬНІ ЕЛЕМЕНТИ ТА ВИМОГИ ДО РЕАЛІЗАЦІЇ СИСТЕМИ «РОЗУМНИЙ ДІМ»

2.1 Концепція «розумний дім»

«Розумний дім» поєднують в собі найсучасніші технологічні досягнення в області хмарності, з'єднання, аналітики та фізичний дизайн, що забезпечує кращу видимість, більшу ефективність, контроль та більш задоволених його мешканців. У певному сенсі ідея інтелектуальних будівель не є новим. Багато будинків десятиліттями мали «розумні» елементи, такі як освітлення, що рухається, та програмовані HVAC системи. Сьогодні «розумний дім» - це той, який поєднує в собі найсучасніші технологічні досягнення в IoT хмару, з'єднання, аналітику та фізичний дизайн, що забезпечує кращу видимість, більшу ефективність та контроль, а також більш задоволених, продуктивних людей. Це динамічний, чутливий об'єкт, який адаптується до змінених умов через пов'язані системи будівництва та мережу датчиків і пристроїв.

Основою розумного будівництва є ця мережа пов'язаних речей, яка автоматично визначає та захоплює величезну кількість цінної інформації про фізичне середовище та його мешканців, таких як температура, світло, заселеність і навіть біометрія. Ці дані підключаються до інших систем, аналізуються та використовуються для більш обґрунтованих дій та рішень. До них відносяться автоматичні дії, як-от проактивні сповіщення, коли обладнання ризикує зазнати невдачі, а також кращі рішення щодо використання ресурсів, простору тощо.

Завдяки зближенню технологічних інновацій та попиту на ринку, площа «розумних будівель» швидко зростає. За оцінками Gartner, до 2020 року у комерційних будівлях буде більше одного мільярда пов'язаних рішень. Завдяки рішенням IoT, будинки можуть витягувати дані з широкого кола пристроїв, щоб реалізувати різні переваги від рівня інфраструктури до мешканців. Зараз ці технології є економічно вигідними за масштабами.

Вартість апаратної техніки різко скоротилася, що робить більш економічно ефективним встановлення інтелектуальних датчиків і пристроїв як у нових, так і в існуючих будівлях. Підключення є повсюдним, і стандартні протоколи побудови знаходяться на місці, що дозволяє швидко і легко їх об'єднати один з одним та з іншими системами. Хмарне зберігання даних, обробка та аналітика дають змогу швидко отримувати великі значення від величезної кількості будівельної інформації. Ці досягнення не тільки стимулюють ріст інтелектуальних будівельних рішень, але також стимулюють зростання IoT у більш широкому сенсі.

У той же час, зміна культурних та демографічних показників змінює ті види досвіду, які очікують мешканці. Нові тенденції на робочому місці, такі як готельний сервіс та гаряча обробка документів, змінюють пріоритети компаній у співпраці та гнучкості. Звертаючись до цих зростаючих очікувань, може знадобитися адаптовані можливості, які пропонують розумні рішення для будівництва. Нормативи стабільності та нові стандарти бухгалтерського обліку для орендованого простору також створюють попит на можливості розумного будівництва. По суті, створення будь-яких зацікавлених сторін має щось отримати від впровадження розумних будівельних рішень.

Значною мірою, «розумні будівлі» стають все більш помітними, оскільки вони відповідають найважливішим потребам будівельників, операторів та мешканців. Багато власників будинків зацікавлені в «розумних будівлях», оскільки вони забезпечують більшу увагу до будівельних операцій та охорони здоров'я. Це дозволяє власникам більш точно управляти витратами та ресурсами, вирішувати неефективність та покращити використання космічного простору. Для більшості власників складно отримати цілісний погляд на операції. Їм потрібно рішення, яке дає змогу існуючим системам будівництва легко спілкуватися один з одним. Розумні рішення для будівництва можуть об'єднувати дані через різні мережі та обладнання, щоб забезпечити єдине уявлення про те, що відбувається, на рівні окремих будівель або у всьому кампусі [8].

Зрештою, власники шукають шляхи підвищення вартості своїх властивостей, зберігаючи високий рівень занять. Вони в основному зосереджені на створенні цікавого та бажаного досвіду мешканцям. Для корпоративних власників це часто означає можливість збільшення продуктивності працівників. Для підприємств комерційної нерухомості це означає створення диференційованого досвіду орендаря та потенційно нових джерел надходжень від нових послуг. Оператори будівництва поділяють багато цілей, подібних власникам. Це включає в себе бажання ефективного розуміння, що дає їм змогу зменшити споживання ресурсів, дотримуватися стандартів стабільності, знайти можливості для більшої ефективності та оптимізувати бізнес-процеси. Оператори особливо шукають інформацію в режимі реального часу, щоб вони могли вирішувати проблеми швидше, керувати продуктивністю більш активно та забезпечувати задоволення мешканцям. Відмінною рисою провідних інтелектуальних будівельних рішень є їх здатність пропонувати таку інформацію в реальному часі, здійснену завдяки інтуїтивно зрозумілим інформаційним панелям та іншим візуалізаціям даних.

З випуском X10 1975 року, протокол зв'язку для домашньої автоматизації, «розумний дім», колись мріяв а-ля The Jetsons, ожив. X10 посилає частоти цифрової інформації на 120 кГц (RF) на існуючу в будинку електропроводку до програмованих розеток або комутаторів. Ці сигнали передають команди відповідним пристроям, контролюючи, як і коли працюють пристрої. Наприклад, передавач може передавати сигнал по електропроводці будинку, кажучи пристрою увімкнутись у певний час. Однак, оскільки електропроводка не розрахована на особливу свободу від "шуму" радіодіапазону, X10 не завжди був повністю надійним. Сигнали будуть втрачені, а в деяких випадках сигнали не перетинатимуть ланцюги, що їх проводять на різних полярностях, створюваних, коли служба 220-вольт розбивається на пару 100-вольтових каналів, як це прийнято в США. Крім того, X10 був спочатку це одностороння технологія, тому, хоча розумні

пристрої можуть приймати команди, вони не можуть надсилати дані в центральну мережу. Пізніше, однак, двосторонні пристрої X10 стали доступними, хоча і з більш високою ціною.

«Розумний дім», який також можна назвати будинком з цифровим зв'язком, використовує новітні системи домашньої автоматизації для забезпечення зручного способу моніторингу та контролю за рядом електронних пристроїв у всьому будинку. Прийняття смартфона сприяло зростанню та розвитку розумного будинку. Смартфон із великим набором додатків став багатофункціональним контролером для пристроїв розумного дому, який може запланувати та контролювати автоматичну роботу опалення, охолодження, освітлення, електричних пристроїв та охоронних систем. Типовий сучасний будинок має ряд незалежних функцій, які зазвичай потребують участі людини для роботи. Двері потрібно розблокувати, фізично повернувши ключ у замку, сигналізацію потрібно встановити натисканням клавішної клавіатури для введення коду, системи опалення та охолодження потрібно перекрити, повернувши циферблат на кімнатний термостат, вогні включаються та вимикаються за допомогою перемикача, а розважальними системами зазвичай керують натисканням кнопок на пристрої дистанційного керування.

«Розумний дім» може автоматизувати всі ці операції, створивши єдину систему, яку можна керувати одним пристроєм – смартфоном. З «розумним будинком», приїжджаючи додому з роботи, гаражні двері автоматично відкриються, вхідні двері відчиняться, увімкнеться світло, система опалення та охолодження спалахне, і ви навіть можете грати улюблену музику вітаємо вас додому - все, не торкаючись кнопки. Цей рівень зручності не відбувається просто магією, він потребує планування, розгляду при придбанні електричних виробів та пристроїв, надійної та надійної домашньої мережі та деяких знань про те, як керувати цифровою системою.

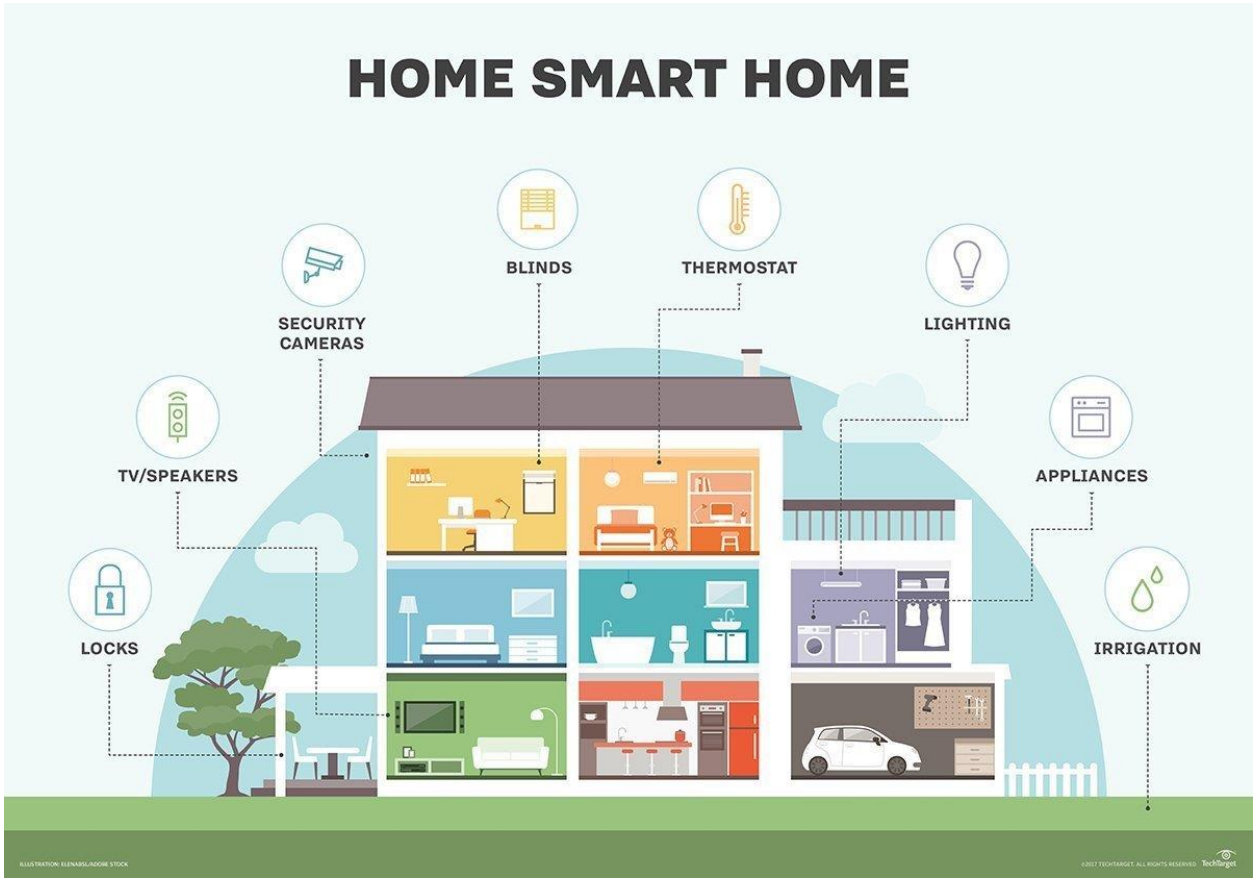

Рисунок 2.1 – Приклад використання «розумного дому»

Для багатьох людей ключовими мотивуючими чинниками створення розумного будинку є економія енергії та економія грошей. Більшість власників розумних будинків починають із встановлення розумних термостатів та розумного освітлення з метою зменшення споживання електроенергії та, таким чином, економії коштів. Незалежні дослідження показали, що встановивши розумний термостат, який оптимізує використання обігріву та / або охолодження, домогосподарства заощадили до 12% на витратах на опалення та 15% на витрачених витратах на охолодження. Після нагрівання освітлення ще однією сильною причиною, чому люди хочуть розумного будинку, є безпека. За допомогою веб-камер, розумних замків дверей, датчиків вікон та дверей та розумних систем безпеки можна стежити за своєю власністю під час роботи чи з будь-якої точки світу, використовуючи підключений до Інтернету пристрій, як смартфон. [9]

Переваги та недоліки розумного будинку

Однією з найбільш розрекламованих переваг домашньої автоматизації є забезпечення спокою господарям будинків, дозволяючи їм дистанційно стежити за своїм будинком, протидіючи небезпекам, таким як забута кавоварка, залишена на вході або вхідні двері не розблоковані. «Розумний дім» також корисний для людей похилого віку, забезпечуючи моніторинг, який може допомогти людям похилого віку комфортно та безпечно залишатися вдома. Домашня автоматизація також допомагає споживачам підвищити ефективність. Замість того, щоб залишати кондиціонер на цілий день, розумна домашня система може навчитися адаптуватись до вашої поведінки та переконатися, що будинок охолоне до моменту, коли ви приїдете додому з роботи. Те саме стосується приладів. А за допомогою розумної системи поливу ваш газон буде поливатись лише тоді, коли це потрібно, і потрібною кількістю води. Завдяки домашній автоматизації енергія, вода та інші ресурси використовуються більш ефективно, що допомагає економити як природні, так і гроші для споживача.

Однак системи домашньої автоматизації намагаються стати мейнстрімом, частково через їх технічну природу. Недоліком розумних будинків є їх сприйнята складність; деякі люди мають труднощі з технологією або відмовляться від неї з першого роздратування. Виробники розумних будинків та альянси працюють над зменшенням складності та вдосконаленням роботи користувачів, щоб зробити його приємним та корисним для користувачів усіх типів та технічних рівнів. Щоб системи домашньої автоматизації були справді ефективними, пристрої повинні бути сумісними, незалежно від того, хто їх виготовив, використовуючи той самий протокол або, принаймні, додаткові. Оскільки це такий ринок, ще не існує золотого стандарту для домашньої автоматизації. Однак стандартні альянси співпрацюють з виробниками та протоколами для забезпечення сумісності та безперебійного користування.

Ще одне важливе питання - безпека розумного будинку. У звіті NTT Data Corp. за 2016 рік було встановлено, що 80% американських споживачів

стурбовані безпекою даних про їх розумні будинки. Якщо хакерам вдасться проникнути в розумний пристрій, вони можуть потенційно вимкнути світло та тривогу та розблокувати двері, залишивши будинок беззахисним до вторгнення. Крім того, хакери можуть потенційно отримати доступ до мережі власників дому, що призведе до гірших атак або розшифровки даних. У жовтні 2016 року ботнет Mirai IoT зміг знищити частини Інтернету в серії розповсюджених атак відмови в обслуговуванні (DDoS), використовуючи погано захищені камери, відеореєстратори та маршрутизатори як точки входу. Окрім безпеки, багато противників розумного будинку турбуються про конфіденційність даних. У звіті про дані NTT встановлено, що 73% споживачів стурбовані конфіденційністю даних, якими надаються їхні смарт-домашні пристрої. У той час як виробники розумних домашніх пристроїв та платформ можуть збирати дані споживачів для кращого адаптації своєї продукції або пропонування нових та вдосконалених послуг для клієнтів, довіра та прозорість мають вирішальне значення для виробників, що вибудовують довіру до користувачів своїх смарт-продуктів.

2.2 Функціональні елементи «розумного дому»

Через складність використання програмних засобів, апаратних засобів, вбудованих систем та мережевих екосистем стає дуже важливим зрозуміти, вивчити та використовувати належну технологію домашньої автоматизації. Вітчизняна автоматизація має три переважаючі компоненти:

1. Обладнання
2. Програмне забезпечення / програми
3. Вербальні протоколи обміну

Кожна з цих частин не менш важлива для створення розумного будинку, який може сподобатися клієнтові. Наявність належного обладнання має можливість розширити прототип IoT. Розумні будівлі можуть, наприклад, знизити витрати енергії за допомогою датчиків, які визначають, скільки мешканців знаходиться

в приміщенні. Температура може регулюватися автоматично - наприклад, увімкнути кондиціонер, якщо датчики виявляють, що конференц-зал переповнений, або знизить тепло, якщо всі в офісі пішли додому. Протокол, обраний з правильним виваженим та обережним розглядом, дозволяє уникнути будь-яких вузьких місць, які в іншому випадку можуть обмежувати можливості інтеграції області та пристроїв за допомогою датчиків та IoT-шлюзів. Побудувавши очікувані потреби споживачів, автоматизація дому, як передбачається, орієнтується на великі діапазони застосувань для абсолютно нового цифрового замовника. Деякі сфери, в яких клієнти можуть передбачити підключення IoT, що підтримується вітчизняною автоматизацією.

а. Відеокамери для спостереження та аналітики

Набір веб-камер та камер, характерних для наборів обладнання для вдосконалення обладнання, зазвичай використовується у таких сценаріях, як спостереження. Існує обладнання з портами USB, які пропонують інтегрувати цифрові модулі для побудови можливостей. Але зазвичай використання портів USB не є дуже ефективним, особливо у випадку перемикання відео в режимі реального часу або будь-якої форми обробки відео.

Візьмемо, наприклад, Raspberry Pi. Він оснащений модулем digicam (Pi cam), який підключається за допомогою гніздового роз’єму безпосередньо до плати без використання USB-порту. Це робить камеру Pi надзвичайно ефективною.

Рисунок 2.2 – Модуль камери digicam для Raspberry Pi

Камера для створення системи відеоспостереження, Time-lapse камери або просто додання зору проекту на основі Raspberry Pi. Підключається до CSI роз'єму (розробленим спеціально для камери), що мінімізує використання ресурсів центрального процесора (великий недолік звичайних веб-камер). Роздільна здатність матриці: 5 Мп. Максимальний розмір зображення: 2592 x 1944.

б. Виявлення звуку

Виявлення звуку відіграє важливу позицію в цілому партії, відстежуючи немовлят, регулярно включаючи і вимикаючи, щоб технічно виявляти звук вашої собаки у дверях і відкривати його своєму вихованцеві. Деякі часто використовувані датчики для виявлення звуку включають SEN-12462 та захист EasyVR для швидкого прототипування. Однак ці датчики не такі бажані, як датчики промислового класу, такі як 3DSignals, які могли б знайти навіть надзвичайно низький рівень шуму та першокласну музику між численними ступенями шуму, щоб створити рівномірні схеми розбиття системи.

Рисунок 2.3 – Датчик звуку для Raspberry Pi

Даний датчик може розпізнати наявність звуку (по принципу вібрації), але не може розпізнати розмір звуку чи конкретну частоту звуку. Можливе регулювання чутливості звуку, робоча напруга 3,3-5 В.

в. Датчики вологості

Ці датчики несуть функціональність зондування рівня вологості / вологості повітря у повітрі всередині розумних будинків. Точність та точність чутливості залежать від безлічі різноманітних елементів, а також загальної конструкції та місця сенсора. Зазвичай використовують датчики, такі як DHT11 і DHT22.

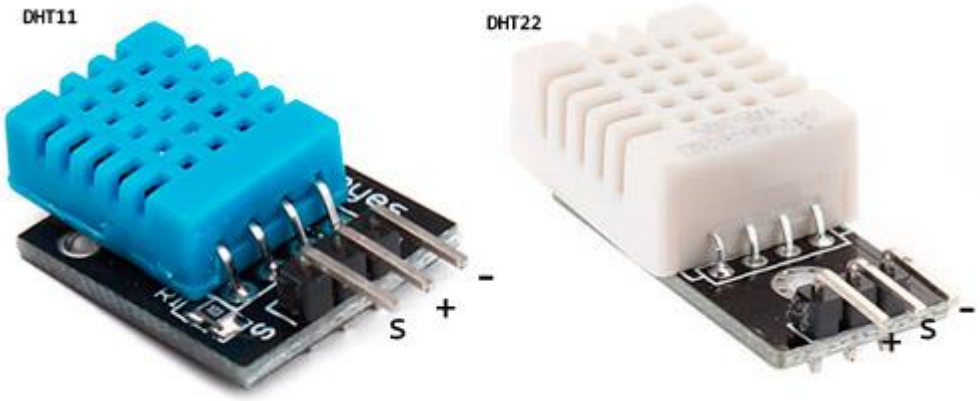

Рисунок 2.4 – Датчики вологості DHT11 і DHT22

Датчик DHT11 є цифровим датчиком температури і вологості. Працює датчик DHT11 тільки при позитивних температурах від 0 до 50 ° С. Відносну вологість повітря вимірює від 20% до 90%. Датчик DHT22 складається з ємнісного датчика вологості і термістора, а також містить в собі АЦП для перетворення аналогових значень в цифрові. Крім вбудованого АЦП основна відмінність від попередника DHT11 - поліпшена точність.

г. Навколишнє середовище та кондиціонування

Однією з найкращих вимогливих ситуацій в інженерії будинків є контроль за навколишнім середовищем та умовами завдяки багатьом елементам. Ці фактори можуть бути будівельними речовинами, погодою, спорудою та ін. Справа з оплатою за електроенергію отримує максимальний інтерес, але кондиціонування додатково впливає на довговічність та стан конструкції. IoT допомагає вдосконалювати конструкцію конструкції та справлятись із сучасними системами за рахунок надто правильних та повних даних про будівлі. Він надає важливу технічну статистику, яка включає, як гарно виконується тканина в якості ізоляції в конкретному дизайні та оточенні.

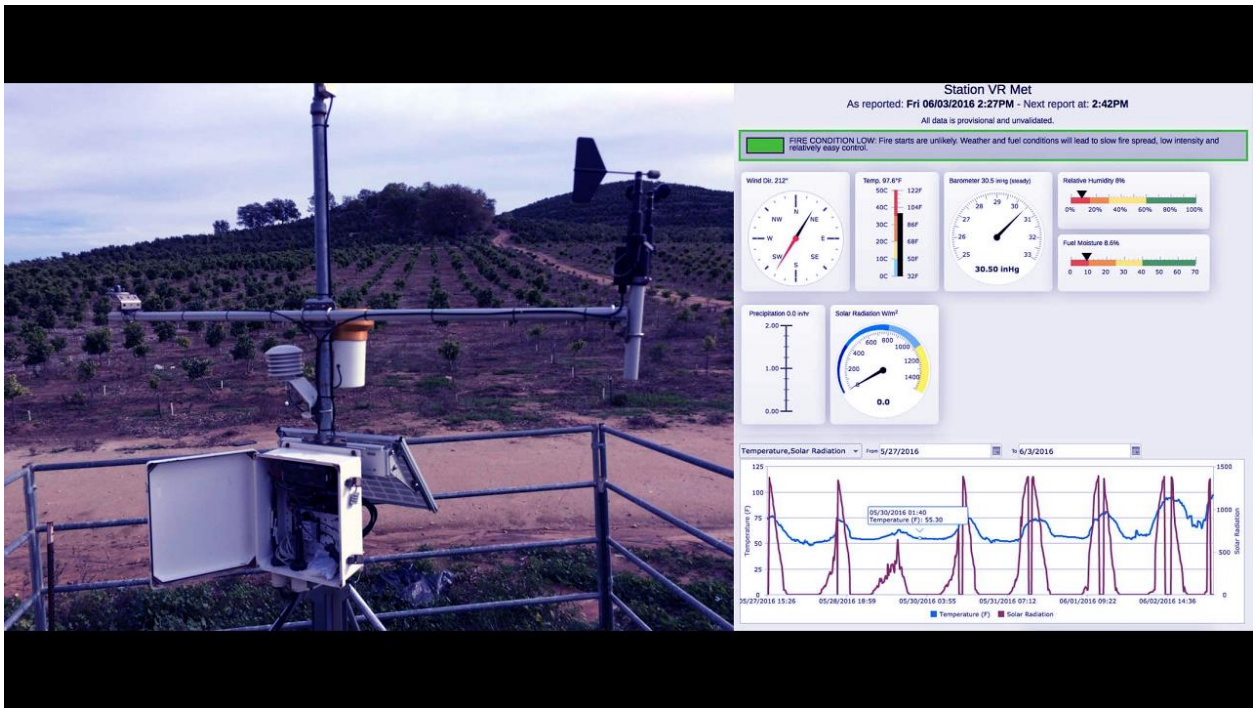

Рисунок 2.5 – Використання особливостей системи «розумного дому»

д. Здоров'я та безпека

Іноді будинки, навіть якщо будуватися з обережністю, можуть страждати від впевнених проблем зі здоров’ям та захистом. Ці неприємності складаються з поганих речовин, що виникають, недоліків, які відходять від будівлі, схильної до надмірного клімату, поганих фундаментів тощо.

Поточні рішення, які пропонуються, не мають вишуканості, необхідної для виявлення незначних проблем, перш ніж вони стануть головними проблемами або надзвичайними ситуаціями. IoT пропонує більш надійну і повну відповідь через дивлячись на проблеми приємно-детально, щоб маніпулювати ризиками та корисним ресурсом для їх зупинки; наприклад, він може вимірювати коригування в царстві гаджета, що впливає на захист від пожежі, на відміну від простого виявлення диму.

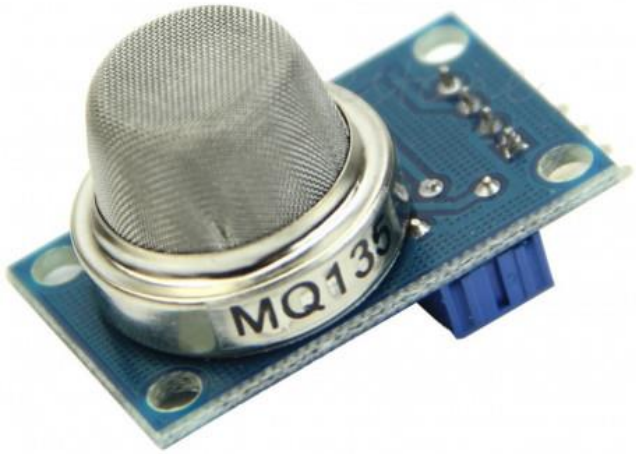

Рисунок 2.6 – Датчик якості повітря MQ-135

Модуль датчика якості повітря MQ-135 може бути використний для визначення вмісту та кількості шкідливих газів в повітрі. Чутливість датчика можна регулювати за допомогою потенціометра.

е. Продуктивність та якість життя

Крім захисту або енергетичних турбот, більшість виборів позитивних зручностей з житлових чи промислових районів, як точне освітлення та температура. IoT покращує ці переваги за допомогою швидшого та простішого налаштування. Зміни додатково дотримуються близько до продуктивності. Вони налаштовують зони для створення оптимізованого оточення разом з розумним робочим місцем або кухнею, організованою для конкретного персонажа. [10]

2.3 Аналіз готових рішень для побудови системи «розумний дім»

Типовий «розумний дім»» оснащений набором датчиків для вимірювання домашніх умов, таких як: температура, вологість та світло. Кожен датчик призначений для зйомки одного або декількох вимірювань. Температуру і вологість можна виміряти одним датчиком, інші датчики обчислюють коефіцієнт світла для даної ділянки та відстань від нього до кожного предмета, що піддається впливу. Усі датчики дозволяють зберігати

дані та візуалізувати їх, щоб користувач міг їх переглядати в будь-якому місці та в будь-який час. Для цього він включає процесор обробки сигналів, інтерфейс зв'язку та хост на хмарній інфраструктурі. Для керування побутовою технікою створюється хмарний сервіс, що буде управляти побутовою технікою, який розміщуватиметься у хмарній інфраструктурі. Служба керування дозволяє користувачеві керувати виходами розумних приводів, пов'язаних з побутовою технікою, наприклад, лампами та вентиляторами. Розумні приводи - це пристрої, такі як клапани та вимикачі, які виконують такі дії, як включення чи вимкнення речей або налаштування операційної системи. Приводи забезпечують різноманітні функції, такі як обслуговування клапанів увімкнення / вимкнення, позиціонування до відкритого відсотка, модуляція для управління змінами в умовах потоку, аварійне відключення (ESD). Щоб активувати привід, приводу видається команда цифрового запису.

Технології доступу до дому зазвичай використовуються для дверей загального доступу. Загальна система використовує базу даних з ідентифікаційними атрибутами уповноважених людей. Коли людина наближається до системи контролю доступу, її ідентифікаційні атрибути збираються миттєво та порівнюються з базою даних. Якщо вони відповідають даним бази даних, доступ дозволений, інакше доступ заборонено. Для широкого розповсюдженого інституту ми можемо використовувати хмарні сервіси для централізованого збору даних осіб та їх обробки. Одні використовують магнітні картки або картки ідентифікації близькості, інші використовують системи розпізнавання облич, друк пальців та RFID.

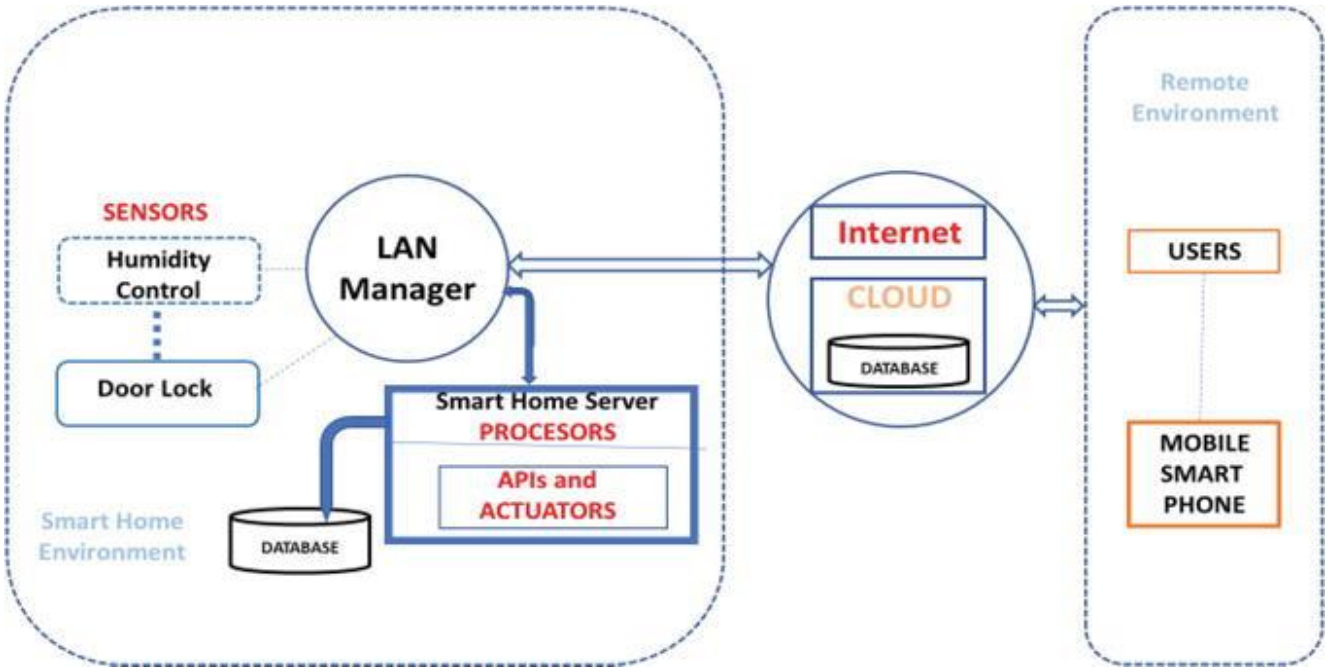

Рисунок 2.7 – «Розумний дім» з додатковим хмарним підключенням

Датчики для збору внутрішніх і зовнішніх домашніх даних та вимірювання домашніх умов. Ці датчики підключені до самого будинку та до приєднаних до нього пристроїв. Ці датчики не є датчиками інтернету речей, які прикріплені до побутової техніки. Дані датчиків збираються та постійно передаються через локальну мережу на сервер розумного дому. Процесори для виконання локальних та інтегрованих дій. Він також може бути підключений до хмари для додатків, що вимагають розширених ресурсів. Дані датчиків потім обробляються локальними серверними процесами. Колекція програмних компонентів, завершена як API, дозволяє зовнішнім програмам виконувати її, якщо вона відповідає заданому формату параметрів. Такий API може обробляти дані датчиків або керувати необхідними діями. Приводи забезпечення та виконання команд на сервері чи інших пристроях управління. Він переводить необхідну активність у синтаксис команд; пристрій може виконати. Під час обробки даних отриманих датчиків завдання перевіряє, чи якесь правило стало істинним. У такому випадку система може запустити команду на належний процесор пристрою. База даних для зберігання оброблених даних, зібраних з датчиків [та хмарних сервісів]. Він також буде

використовуватися для аналізу даних, представлення даних та візуалізації. Оброблені дані зберігаються у доданій базі даних для подальшого використання.

Хмарні обчислення - це спільний простір обчислювальних ресурсів, готових надати різноманітність обчислювальних послуг на різних рівнях, від базової інфраструктури до більш складних послуг додатків. На практиці хмарні обчислення керуються обчисленнями, сховищами та комунікаційними ресурсами, якими ділиться кілька користувачів у віртуалізованому та ізольованому середовищі.

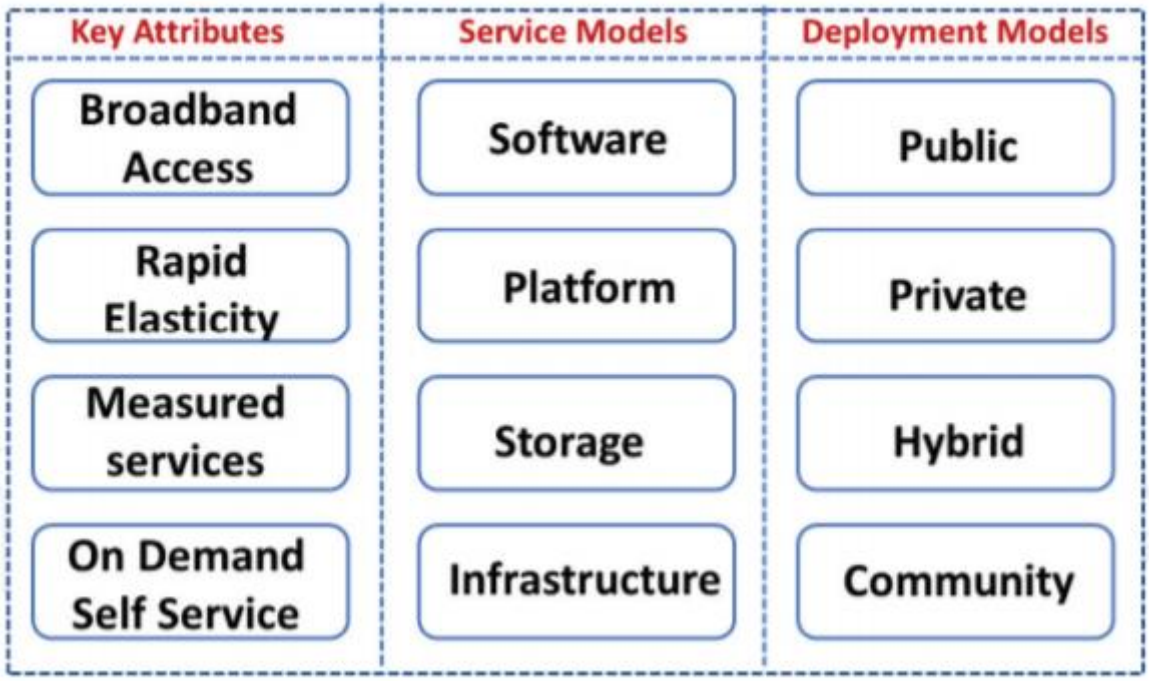

Рисунок 2.8 – Парадигма хмарних обчислень

IoT та «розумний дім» можуть скористатися широкими ресурсами та функціоналами хмар для компенсації свого обмеження в зберіганні, обробці, спілкуванні, підтримці у виборі попиту, резервного копіювання та відновлення. Наприклад, хмара може підтримувати послугу IoT управління та виконання додаткових додатків за допомогою даних, отриманих нею. «Розумний дім» та IoT будуть зосереджені на зборі даних, базовій обробці та передачі в хмару для подальшої обробки. Щоб вирішити проблеми безпеки, хмара може бути приватною для високозахищених даних та

загальнодоступною для решти.

IoT, «розумний дім» та хмарні обчислення - це не просто злиття технологій, але

і баланс між локальними та центральними обчисленнями разом з оптимізацією споживання ресурсів. Обчислювальне завдання може бути виконано на IoT і розумні домашні пристрої або передані в хмару. Де проводити обчислення, залежить від доступності даних, залежність даних, кількість передачі даних, залежність від зв'язку та міркування безпеки. З одного боку, потрібна обчислювальна модель, що включає хмару, IoT та «розумний дім», а також вона повинна мінімізувати всю вартість системи, зазвичай, приділяючи більше уваги скороченню ресурсів споживання в домашніх умовах. З іншого боку, потрібна модель, що повинна вдосконалити IoT користувачів, щоб задовольнити їх попит під час використання хмари та вирішення складних проблем, що виникають.

Зараз ми розглянемо інтеграцію «розумного будинку», IoT та хмарних обчислень, щоб визначити нову обчислювальну парадигму.

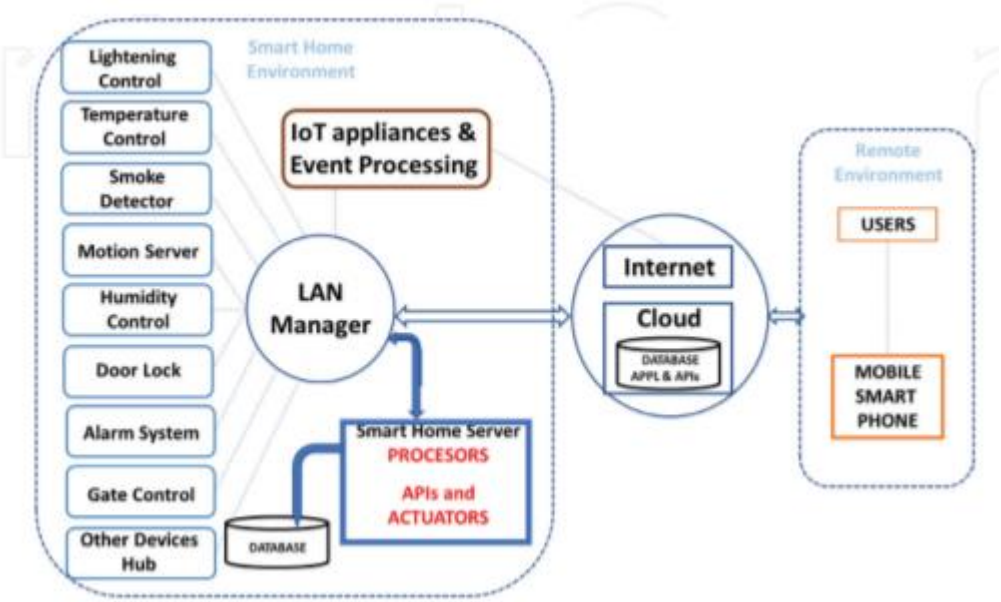

Рисунок 2.9 - «Розумний дім» з інтеграцією IoT та хмарним обчисленням

На рисунку 2.9 зображено вдосконалені основні компоненти «розумного дому» та їх взаємопов'язаність. На лівому блоці, середовищі розумного будинку, ми бачимо типові пристрої, підключені до локальної мережі [LAN].

Це дає змогу спілкуватися між пристроями та поза ними. Підключений до локальної мережі - це сервер та його база даних. Сервер контролює пристрої, реєструє його діяльність, надає звіти, відповідає на запити та виконує відповідні команди. Для більш всебічних або звичайних завдань сервер розумного дому передає дані в хмару і віддалено активує завдання в ньому за допомогою API, процесів інтерфейсу програмування програм. Крім того, побутова техніка IoT підключена до Інтернету та до локальної мережі, і таким чином розширює розумний будинок, включаючи IoT. Підключення до Інтернету дозволяє кінцевому користувачеві, резиденту, спілкуватися з розумним будинком, щоб отримувати поточну інформацію та віддалено активувати завдання. Щоб продемонструвати переваги сучасного розумного будинку, було використано RSA, надійний асиметричний алгоритм криптографії, який генерує відкритий та приватний ключ та шифрує / розшифровує повідомлення. За допомогою відкритого ключа кожен може зашифрувати повідомлення, але лише ті, хто має приватний ключ, можуть розшифрувати надіслане повідомлення. Генерування ключів та шифрування / розшифрування повідомлень передбачає великі обчислення, що потребують значного простору пам'яті та потужності для обробки. Тому його зазвичай обробляють на потужних комп'ютерах, побудованих для впорядкування необхідних ресурсів. Однак, через обмежені ресурси, запускати RSA на пристрої IoT майже неможливо, і, таким чином, це відкриває розрив у безпеці в Інтернеті, де зловмисники можуть легко використовувати. Щоб впоратися з цим, можна по’єднати в собі потужність локальних процесорів «розумного дому» для обчислення деяких обчислень RSA та передачі складніших обчислювальних завдань для обробки в хмарі. Потім результати будуть передані назад на IoT-датчик, який буде складено і зібрано разом, щоб генерувати код шифрування / дешифрування RSA і так закрити згаданий проміжок безпеки IoT. Цей приклад демонструє потік даних серед передових компонентів «розумного дому», де кожен компонент виконує свій власний стек операцій для отримання унікального результату. Однак у випадку

складних і тривалих завдань воно розділить завдання на підзавдання, які виконуються більш потужними компонентами. Посилаючись на приклад RSA, пристрій IoT ініціює необхідність генерування ключа шифрування і, таким чином, надсилає повідомлення із запитом до програми RSA, запущеного в смарт-домашньому комп'ютері. Потім розумний домашній комп'ютер запитує програму «генерація простих чисел», що працює на хмарі, надати p і q прості числа. Як тільки p і q приймаються, формується код шифрування. На пізньому етапі пристрій IoT надсилає запит на розумний домашній комп'ютер для шифрування повідомлення, використовуючи недавно створений ключ шифрування RSA. Потім зашифроване повідомлення передається назад на пристрій IoT для подальшого виконання. Аналогічний сценарій може бути в протилежному напрямку, коли пристрій IoT отримує повідомлення, він може попросити «розумний дім» розшифрувати його. Підводячи підсумок, сценарії RSA зображують використання сили хмарних обчислень, «розумний будинок» захищений обчислювальними можливостями, і в кінці обмеженою потужністю пристроїв IoT. Це доводить, що без цього автоматичного співробітництва RSA не було б можливо виконати на рівні IoT. [11]

3 РОЗРОБКА СИСТЕМИ «РОЗУМНИЙ ДІМ» НА RASBERRY PI

3.1 Вибір мережевої інфраструктури

В якості нашої мережевої моделі було вирішено розглянути одну з провідних технологій домашньої автоматизації минулого десятиліття. Z-Wave був розроблений спеціально для додатків домашньої автоматизації. Це протокол бездротового зв'язку з низькою потужністю, розроблений, щоб забезпечити кінцевим користувачам ефективний та надійний метод дистанційного керування широким спектром пристроїв та систем. Для виробників Z-Wave пропонує економічно вигідне та просте у виконанні рішення для того, щоб зробити свою продукцію розумною та підключеною.

Z-Wave була представлена на ринок у 2003 році компанією Zensys, придбаною через п'ять років компанією Sigma Designs, яка зараз ліцензує технологію і залишається основним постачальником мікросхем Z-Wave. Вирішуючи всі найважливіші потреби сегменту розумного будинку, що розвивається, він став провідним міжнародним бездротовим стандартом для управління та автоматизації в житлових умовах. На сьогоднішній день на ринку представлено понад 1300 сертифікованих пристроїв і близько 35 мільйонів сумісних одиниць, що знаходяться в обігу, це зріла і перевірена технологія.

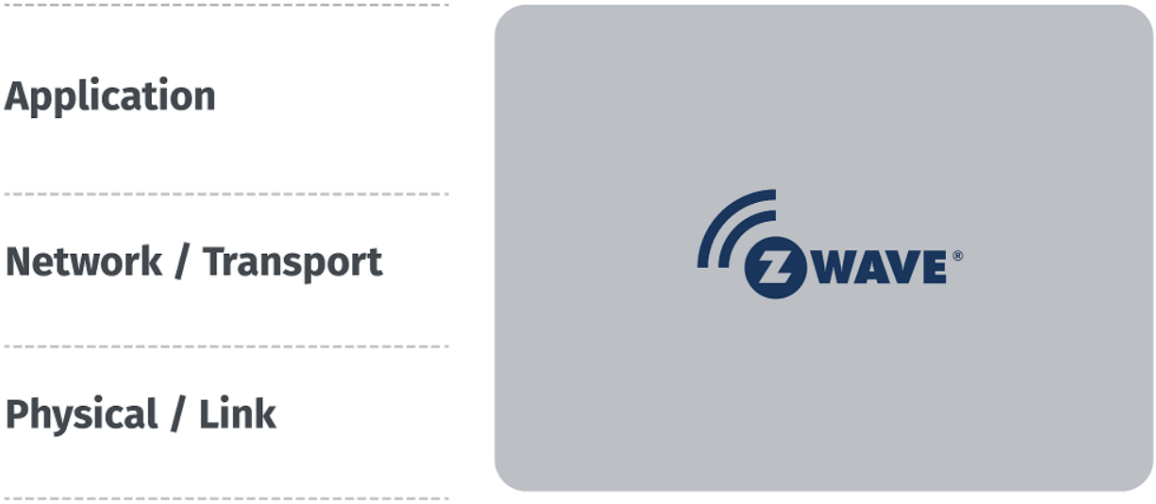

Рисунок 3.1 – Рівні моделі OSI, де використовується Z-Wave

Простий Z-Wave охоплює всі шари основної моделі OSI для мережевих комунікацій, від фізичного рівня до рівня додатків. На відміну від переважної більшості провідних рішень для бездротового зв'язку, протокол забезпечує повну сумісність між різними брендовими продуктами, заснованими на ньому. Це було б неможливо без визначення рівня додатків, але також без чітких стандартних та сертифікаційних програм, встановлених Z-Wave Alliance. Z-Wave був розроблений таким чином, щоб дозволити окремим вузлам пересилати повідомлення, поки вони не досягнуть свого кінцевого пункту призначення. Значно розширюючи діапазон бездротової мережі, навіть сьогодні мережі розглядаються як важливі для забезпечення надійного покриття в автоматизації будівель. Технологія домашньої автоматизації Z-Wave складається з чотирьох шарів.

Фізичний: визначає спосіб обміну сигналом між мережею та фізичним обладнанням. Сюди входить частота, кодування, апаратний доступ тощо.

Мережевий: визначає спосіб обміну даними управління між двома пристроями або вузлами. Сюди входить адресація, організація мережі, маршрутизація тощо. Мережевий рівень Z-Wave контролює обмін даними між різними пристроями (вузлами) в мережі, він складається з трьох підрівнів.

Рівень доступу до медіа (MAC): керує основним використанням бездротового обладнання - ці функції невидимі для кінцевого користувача. Z-Wave - одна з найнадійніших бездротових технологій, кожна відправлена команда підтверджується одержувачем, який надсилає зворотну квитанцію відправнику. Це не гарантує правильність доставки повідомлення, проте відправник отримає вказівку на те, що ситуація змінилася або сталася помилка.

Транспортний рівень: керує передачею повідомлень, забезпечуючи безперешкодне спілкування між двома бездротовими вузлами. Кінцевий користувач не може впливати на функції цього шару, але результати цього шару помітні. Рівень маршрутизації: керує можливостями «Mesh» Z-Wave, щоб максимально розширити мережевий діапазон і забезпечити потрапляння повідомлень до вузла призначення. Цей рівень використовуватиме додаткові вузли для повторного надсилання повідомлення, якщо призначення знаходиться поза "прямого" діапазону передаючого вузла.

Прикладний: визначає, з якими повідомленнями потрібно обробляти конкретні програми для виконання конкретних завдань, таких як перемикання світла або зміна температури нагрівального приладу.

Перш за все, мережа Z-Wave здатна направляти повідомлення через до 4 повторюваних вузлів. Це одне лише обмежує потенційні програми в середовищі розумного будинку, оскільки комерційні приміщення та офісні приміщення часто потребують більш широкого охоплення мережі. З типовим внутрішнім діапазоном модулів Z-Wave, що наближаються до 40 м, сітчаста мережа з обмеженням 4-х перешкод дозволяє передавати дані на розумні відстані, забезпечуючи покриття, достатнє лише для середньої квартири, але набагато нижче потреб цілої офісної будівлі або величезний промисловий об’єкт. Крім того, мережа Z-Wave може мати до 232 вузлів, хоча для запобігання насичення рекомендується менша кількість. Хоча це здається більш ніж достатньо для середнього любителя розумного будинку, це, знову ж таки, серйозний бар'єр для зв’язаного офісного середовища, де незліченна кількість пристроїв, датчиків та контролерів повинні співпрацювати, щоб

забезпечити реальні переваги, особливо що стосується управління енергією. Але навіть у житлових умовах зазначена межа може виявитись недостатньою. Gartner прогнозує, що протягом декількох років у типовому будинку може бути більше 500 розумних пристроїв - число, яке Zensys не міг передбачити при розробці архітектури Z-Wave більше десяти років тому. Z-Wave використовує маршрутизацію на основі джерела, що означає, що пристрій, який ініціює повідомлення, генерує повний маршрут через сітчасту мережу до одержувача. Перш ніж це може статися, потрібно створити таблиці маршрутизації. Це робиться за допомогою пристрою під назвою первинний контролер, який аналізує всю мережу, щоб створити оптимальні маршрути між її вузлами. Ця топологія, орієнтована на джерела, є ефективною та надійною, доки мережеве розташування залишається незмінним. Але якщо один з вузлів виходить з ладу, напр. лампа вигорає або мобільний пристрій виходить за межі діапазону, топологію мережі потрібно повторно розкрити, а таблиці маршрутів потрібно оновити. Процедура зцілення займає деякий час (до 1-2 годин), і вся мережа залишається вниз, поки не закінчиться. Рекомендується регулярно його виконувати протягом нічних годин. Для деяких клієнтів це не буде великою проблемою, але те, що ваша розумна екосистема не може працювати 24/7, може бути проблемою у багатьох сценаріях та середовищах. В останніх версіях Z-Wave були введені так звані кадри провідника, які можна використовувати для вирішення проблеми відсутніх вузлів без ініціювання тривалої процедури зцілення мережі. Однак повідомлялося, що вони часто змушують замерзнути всю мережу приблизно на хвилину. З точки зору кінцевого споживача, який не очікує таких величезних затримок, це може бути дуже дратівливим досвідом. У перші дні його хвиля Z вважалася небезпечною. Ряд успішних спроб зламати протокол сильно сприяв цій репутації, хоча слід зазначити, що деякі з цих вразливих ситуацій були спричинені поганими реалізаціями, а не архітектором Z-Wave. Важливо усвідомити, що всі проблеми затримки випливають безпосередньо з архітектури протоколу Z-Wave. Загалом, всі провідні рішення для бездротового підключення для IoT

забезпечують аналогічний рівень безпеки. Але Z-Wave довелося зробити дуже великі жертви щодо ефективності та досвіду користувачів, щоб досягти цього рівня. Що має Z-Wave спільного з переважною більшістю технологій бездротового зв'язку, це те, що центральний концентратор повинен бути розгорнутий в мережі Z-Wave, щоб кінцевий користувач міг керувати ним за допомогою смартфона. Це відбувається тому, що протокол не підтримується безпосередньо смартфонами або планшетами, функція яких залишається ексклюзивним привілеєм Bluetooth та Wi-Fi. Z-Wave – це протокол, що протокол використовує 8-бітну контрольну суму із відносно простим алгоритмом для перевірки цілісності даних, що обмінюються між окремими вузлами. Це часто розглядають як одну з основних слабких місць цього стандарту зв'язку. Оскільки величезна кількість пакетів даних працює над більш мережевими мережами, слабка контрольна сума може не змогти запобігти випадковій помилковій передачі, що призводить до абсолютно несподіваної поведінки розумного пристрою. На практиці це означає, що лампочка може вмикатися, коли її не вимагають. Або клапан води може відкритися просто так. Для вирішення цього питання нещодавно була доступна більш сильна контрольна сума для підвищення надійності мережі в деяких більш чутливих додатках, але ринок все ще залишається переповненим пристроями, які мають не надто захоплюючу щілину непередбачуваності. [12]

Розглянемо мережу зв’язку S (N) з N вузлів, безпосередньо з’єднаних між собою пучками з n каналів. Усі канали лише односторонні чи лише двусторонні. Позначимо через λ незалежну від направлення інтенсивність потоку викликів для будь якої пари вузлів.

Допустимим обхідним маршрутом назвемо маршрут, який має тільки одне транзитне сполучення. Транзитне вимога, прийняте на обслуговування, займає одночасно по одному вільному каналу в кожному пучку обхідного маршруту і звільняє одночасно всі ці канали після закінчення обслуговування. Час обслуговування викликів для будь-якої пари вузлів однаково для прямого і обхідного маршрутів і має математичне очікування $1/\mu$.

Приймемо, всі процеси надходження викликів і їх обслуговування статистично незалежними в сукупності.

Нехай CIs, CIIs, CIIIs – способи встановлення з'єднання у мережі, при яких вимога приймається на транзитне обслуговування, якщо на обхідний маршрут CIs - вільно не менше s каналів; CIIs - в кожному пучку вільно ні. Проте s-каналів; CIIIs - в мінімально завантаженому пучку вільно не менше s каналів, а в максимально завантаженому - хоча б один.

Нехай Ω- множина всіх пучків, що входять в кожен обхідний і прямий маршрути для фіксованої пари вузлів: p - число зайнятих каналів в пучку з безлічі Ω, $\boldsymbol{\alpha}$ $\boldsymbol{\beta}$ (Ω) - сума зайнятих каналів по всьому пучків з безлічі Ω.

Приймемо за стаціонарне розподілення ймовірностей для $\boldsymbol{\beta}$ (Ω) розподілення:

$$P(\boldsymbol{\beta}(\Omega))=i)=E_i(Y_{ks}(\Omega), n(\Omega)),\ i=0, 1, \ldots, n(\Omega) \quad (3.1)$$

$$E_i(Y_{ks},n)=(E_{ks}^i)/i!)/\sum_{i=0}^{n} Y_{ks}^i/i!,\ n(\Omega)=m; \quad (3.2)$$

$$Y_{ks}\ \Omega=\frac{mp(1+H1-2\boldsymbol{\pi}_{ks})}{1-Hm},\ p=\ \lambda/\mu,\ k=1, 2, 3; \quad (3.3)$$

m – число пучків, що входять Ω; $H_r, \boldsymbol{\pi}_{ks}(k=1,2,3)-$ *стаціонарні ймовірності, причому* H_r-

того, що в Ω буде зайнято r *пучків,* $\boldsymbol{\pi}_{ks}(k=1,2,3)-$

*того , що відбудиться відмова в зв'язку з*способом встановлення з'єднання CIs, CIIs, CIIIs відповідно.

$$H_m=E_{nm}\ (\frac{mp(1+H1-2\boldsymbol{\pi}_{ks})}{1-Hm}, nm),\ ,=1,2,\ldots, 2N-3. \quad (3.4)$$

При N ≥ 3 має місце наступне відношення:

$$\boldsymbol{\pi}_{ks}=\sum_{r=N-1}^{2N-3}\boldsymbol{\alpha}_{ks}^{r-N+1}\mathbf{C}_{N-2}^{r-N+1}\mathbf{2}^{2N-3-r}\text{-}\boldsymbol{\alpha}_{r},\ k=1, 2, 3 \quad (3.5)$$

$$\boldsymbol{\alpha}_{1s} = \begin{cases} -1, s = 2 \\ -1 + \boldsymbol{\Sigma}_{i=2}^{s-1} \mathbf{Y}_{1s}^{-i} \boldsymbol{\Sigma}_{i=2}^{i-1} \mathbf{A}_{n=2}^{j} \mathbf{A}_{n}^{i-j}, \mathrm{S} \geq 3, \end{cases} \quad (3.6)$$

$$\boldsymbol{\alpha}_{2s} =$$

$$\begin{cases} (\boldsymbol{\Sigma}_{i=2}^{s-1} \mathbf{A}_{n}^{j} \mathbf{Y}_{2s}^{-i}) \quad \frac{2N-3-r}{r-N+1} \left(-\boldsymbol{\Sigma}_{i=0}^{s-1} \mathbf{Y}_{2s}^{-i} \boldsymbol{\Sigma}_{j=0}^{i} \mathbf{A}_{n}^{j} \mathbf{A}_{n}^{i-j} - \boldsymbol{\Sigma}_{i=s}^{2s-2} \mathbf{Y}_{2s}^{-i} \boldsymbol{\Sigma}_{j=i-s+1}^{s-1} \mathbf{A}_{n}^{j} \mathbf{A}_{n}^{i-j} \right), \mathrm{S} \geq 2 \end{cases}$$

(3.7)

$$\boldsymbol{\alpha}_{3s} = \begin{cases} -1, s = 1 \\ -1 + \boldsymbol{\Sigma}_{i=2}^{s} \mathbf{Y}_{3s}^{-i} \boldsymbol{\Sigma}_{j=1}^{i-1} \mathbf{A}_{n}^{j} \mathbf{A}_{n}^{i-j} + \boldsymbol{\Sigma}_{i=s+1}^{2s-2} \mathbf{Y}_{3s}^{-i} \boldsymbol{\Sigma}_{j=i-s+1}^{s-1} \mathbf{A}_{n}^{j} \mathbf{A}_{n}^{i-j}, \mathrm{S} \geq 2 \end{cases} \quad (3.8)$$

Кожній парі вузлів мережі S(N) відноситься N-2 допустимих відступів маршруту. Відмова в зв'язку відбудеться тоді, коли буде зайнятий прямий маршрут і одночасно на усіх N-2 допустимих відступів маршруту завантаження пучків буде віща порогового значення. Позначимо через T_k, $T_{\boldsymbol{\Sigma}_k}$ (k=1, 2, 3) події, які складаються в тому, що при способі встановлення даних CIs, CIIs, CIIIs відповідно: T_k – відбудеться відмова транзитного зв'язку на фіксованому обхідному маршруті: $T_{\boldsymbol{\Sigma}_k}$ (k=1, 2, 3) – відбудеться відмова в транзитному зв'язку.

В силу припущення про статистичну незалежність у сукупності усіх процесів надходження викликів та їх обслуговування маємо:

$$\mathrm{P}(T_{\boldsymbol{\Sigma}_k})\, H_1, \mathrm{k}=1, 2, 3 \quad (3.9)$$

$$\boldsymbol{\pi}_{ks} = \mathrm{P}(T_{\boldsymbol{\Sigma}_k})\, H_1, \mathrm{k}=1, 2, 3 \quad (3.10)$$

Звідси видно, що за способом встановлення з’єднання CIs, CIIs, CIIIs відношення очевидне: P(T_k)=2H_1-H_2 (k=1, 2, 3)

У даному випадку, розглядаються повнодоступні багато-групові схеми, в яких допускаються з'єднання між будь-якими групами. [13]

Рисунок 3.2 – Z-Wave мережа

Все це показує, що Z-Wave, безумовно, не є ідеальною комунікаційною технологією для додатків, які потребують 100% надійності та ефективності. Він також не здається готовим вирішувати виклики майбутнього. З моменту розробки справи пройшли довгий шлях, і тепер він демонструє деякі чітко помітні ознаки старіння. Однак, це все ще залишається ідеально пристойним рішенням для любителів розумного будинку, які часто будуть задоволені своїм досвідом розумного дому, незважаючи на деякі недоліки, згадані вище, головним чином через те, що їхнє середовище пов'язане порівняно мало.

Z-Wave має два основні типи пристроїв:

Контролери - пристрої, що керують іншими пристроями Z-Wave; Раби -

пристрої, якими керують інші пристрої Z-Wave.

Контролери запрограмовані з домашнім ідентифікатором (Home ID), користувач це не може змінити. Раби не мають заздалегідь запрограмованого домашнього ідентифікатора, оскільки вони приймають домашній ідентифікатор, призначений їм мережею.

Основний контролер включає інші вузли в мережу, присвоюючи їм свій домашній ідентифікатор. Якщо вузол приймає домашній ідентифікатор основного контролера, цей вузол стає частиною мережі. Первинний контролер також призначає індивідуальний ідентифікатор вузла (Node ID) кожному новому пристрою, який додається до мережі. Цей процес відомий як включення (Inclusion).

3.2 Опис моделі керування

Сьогодні, коли інтелектуальні системи управління мережами отримують в світі все більшого поширення, багато виробників мережевого та комп'ютерного обладнання активізуються на цьому ринку, а постачальники оригінальних продуктів розширюють їх можливості. Якщо врахувати, що основні переваги цих систем орієнтовані головним чином на менеджерів з інформаційних технологій та мережевих адміністраторів, то виробникам доводиться розриватися між цими кінцевими користувачами і сформованим спільнотою інсталяторів, збільшення обізнаності та розширення підтримки яких вимагають постійної уваги.

Для управлінням такою системою потрібно використовувати комп'ютер з спеціалізованим програмним забезпеченням, але звичайний персональний комп'ютер занадто великий у розмірах і його робота у режимі 24/7 для деяких користувачів здається дещо дивною.

Розроблена нами модель реалізується на апаратній платформі Raspberry Pi. Raspberry Pi - це назва серії одно-платних комп'ютерів, створених Фондом Raspberry Pi, британською благодійною організацією, яка спрямована на те, щоб навчити людей обчислюватися та створити простіший доступ до

обчислювальної освіти. Raspberry Pi - це дуже дешевий комп'ютер, який працює під управлінням Linux, але він також пропонує набір штифтів GPIO (загального призначення введення/виведення), які дозволяють керувати електронними компонентами для фізичних обчислень та досліджувати Інтернет речей (IoT). У якості операційної системи ми будемо використовувати Raspbian, що представляє собою безкоштовну операційна система на базі Debian, оптимізована для апаратних засобів Raspberry Pi. Raspbian постачається з більш ніж 35000 пакунками: попередньо складене програмне забезпечення, яке постачається у приємному форматі для легкої установки на ваш Raspberry Pi.

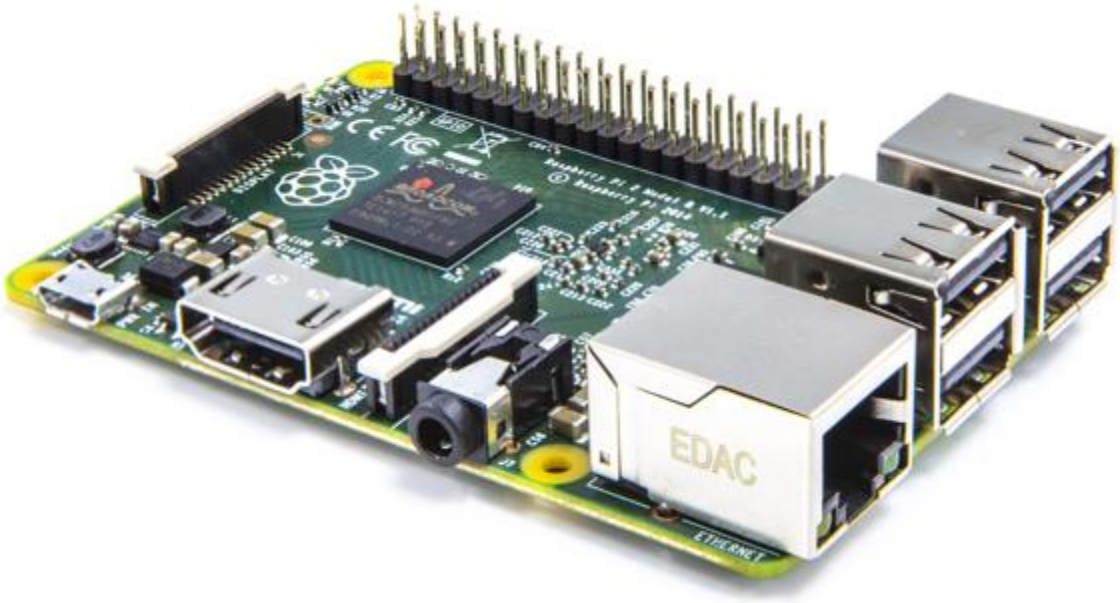

Рисунок 3.3 - Raspberry Pi, використаний в нашій розробці

Практичну реалізацію платформи було виконано мовами Python і Bash. Програмний код основних компонентів наведено в додатку. Були використані такі бібліотеки:

- Telebot – Відкрита реалізація API Telegram для мови програмування Python
- Threading - Бібліотека Python для реалізації багатопотокових обчислень
- RPi.GPIO – Модуль контролю портів вводу-виводу системи Raspberry Pi
- OS – Модуль для роботи з операційною системою
- sshpass – Утиліта для виконання команд на віддаленому сервері
- sshfs – Утиліта для монтування дискового простору віддаленого серверу на сервері локальному

Ядром автоматизованої системи управління мережею є програмний продукт написаний мовою Python, модулі отримання та обробки даних з датчиків та віддалених серверів реалізовані за допомогою мови Bash.

Як вже було відмічено, система «Інтернету речей» повинна мати здатність обслуговувати велику кількість даних, а тому в свою чергу вона повинна бути високоефективною. Ефективним рішенням в досягненні цієї мети є використання асинхронного I/O, тобто такого, який не блокує інші операції. Наприклад, в Python-бібліотеці Threading можна використовувати asyncio, що реалізує асинхронний ввід/вивід. Asyncio використовується як основа для безлічі асинхронних кадрів Python, які забезпечують високопродуктивні мережеві та веб-сервери, бібліотеки підключення до бази даних, розподілені черги завдань тощо. При застосуванні цього методу, якщо очікуються дані із мережі, періодично перевіряється стан мережевого сокету на предмет їх надходження; при цьому потік не блокується, інші обчислення та запити продовжують виконуватися. Потік, що виконує асинхронний файл

вводу/виводу, надсилає запит вводу/виводу до ядра, викликаючи відповідну функцію. Якщо запит прийнято ядром, викликає потік продовжує обробляти інше завдання, поки ядро не подасть сигнал до потоку, що операція вводу/виводу завершена. Потім він перериває своє поточне завдання та обробляє дані з операції вводу/виводу за необхідності.

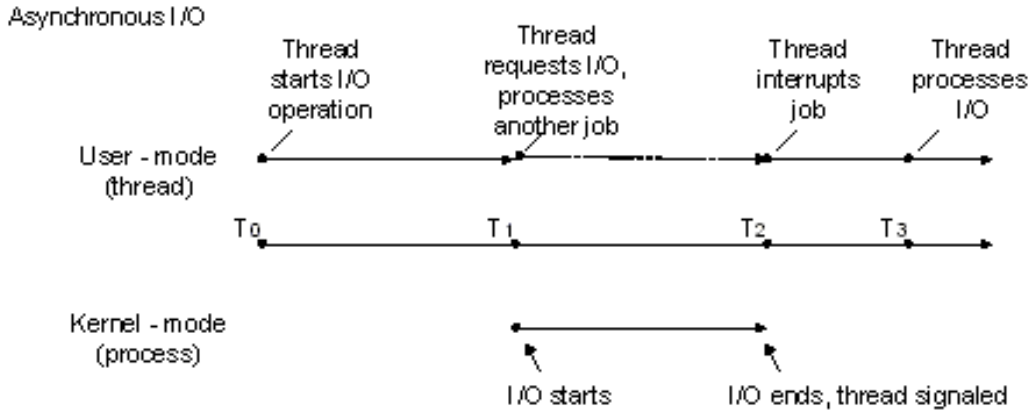

Рисунок 3.4 – Асинхронний ввід/вивід

У ситуаціях, коли очікується, що запит вводу/виводу потребує значної кількості часу, наприклад, оновлення або резервне копіювання великої бази даних або повільний зв'язок зв'язку, асинхронний введення/виведення, як правило, є хорошим способом оптимізації ефективності обробки. Однак для відносно швидких операцій вводу/виводу накладні витрати на обробку запитів вводу/виводу ядра та сигнали ядра можуть зробити асинхронні введення/виведення менш вигідними, особливо якщо потрібно зробити багато швидких операцій вводу/виводу. У цьому випадку синхронні введення/виведення було б краще. Механізми та деталі реалізації способів виконання цих завдань відрізняються залежно від типу ручки пристрою, що використовується, та конкретних потреб програми. Іншими словами, зазвичай існує кілька способів вирішення проблеми.

Іншим засобом підвищення продуктивності є організація stateless архітектури, тобто такої, яка не потребує зберігання на сервері стану між запитами. Наприклад, використання такої моделі авторизації, яка не вимагає перевірки деякого збереженого стану чи звертання до бази даних. У сучасних

розподілених системах часто вузьким місцем є пропускна спроможність мережевого обладнання, а не обчислювальні здатності процесорів. Переваги серверів без стану (stateless) очевидні. Відновлення після відмови сервера вимагає значних витрат на сервер, який підтримує стан усіх його з'єднань, тоді як у випадку з сервером без стану на клієнта це не впливає, поки сервер не працює, а потім повертається між двома послідовними запитами. Система stateless є більш простою, надійною та масштабованою. Клієнт не повинен перейматися станом сервера. Якщо клієнт отримує відповідь на запит, це означає, що сервер працює і працює; якщо ні, він повинен повторно надіслати запит пізніше. Служба на основі з'єднання повинна резервувати пробіли, щоб підтримувати стан кожного з'єднання з клієнтом; отже, така система не є масштабованою, і кількість клієнтів, з якими сервер міг би взаємодіяти в будь-який момент часу, обмежена наявним для сервера місцем зберігання. Наприклад, базовий веб-сервер не має статусу; він відповідає на запит HTTP, не підтримуючи історію минулих взаємодій з клієнтом. Клієнт, браузер, також є stateless, оскільки він надсилає запити та чекає відповідей. [13]

3.3 Апаратна модель системи автоматизованого управління

Апаратним центром системи є одноплатний комп’ютер Raspberry Pi 3 під керуванням серверної версії операційної системи Debian Linux.
Контролер може бути простим пультом дистанційного керування, який працює від батарей, але це звичайно виділений блок, підключений десь у будинку від мережі живлення, до якого зазвичай можна підключитися через Wi-Fi через веб-інтерфейс. Доступно безліч комерційних варіантів, і це правильний шлях для того, щоб налагодити мережу Z-Wave та працювати з нульовими зусиллями. За допомогою USB-Sata контроллеру, до комп’ютера також підключений жорсткий диск, на який виконується запис резервних копій даних з віддалених серверів, з якими встановлено VPN з'єднання та виконується само копіювання програмного комплексу системи. Саме на цій

платформі, у даній системі, реалізований центр керування мережею, серверами та елементами розумного дому.

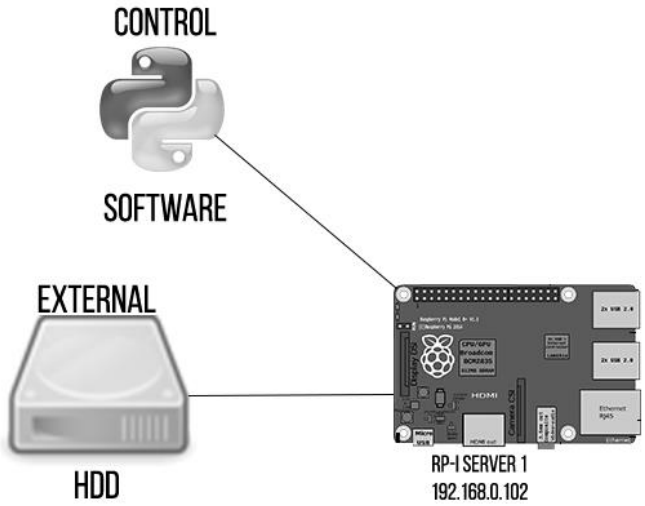

Рисунок 3.5 – Ядро системи керування мережею та елементами розумного дому

До системи вводу-виводу GPIO по протоколу Z-Wave під’єднані датчик температури та вологості, датчик якості диму, а також реле яке за допомогою інтерфейсу керування, або заданим параметрам вимикає та вмикає світло у приміщенні.

Було вирішено використовувати датчик температури і вологості DHT-11. Це недорогий цифровий датчик температури і вологості. Він використовує ємнісний датчик вологості і термістор для вимірювання навколишнього повітря, і видає цифровий сигнал на пін даних. Він досить простий у використанні, але має мінус при отриманні даних: недоліком цього датчика є час отримання даних - не частіше ніж один раз в дві секунди.

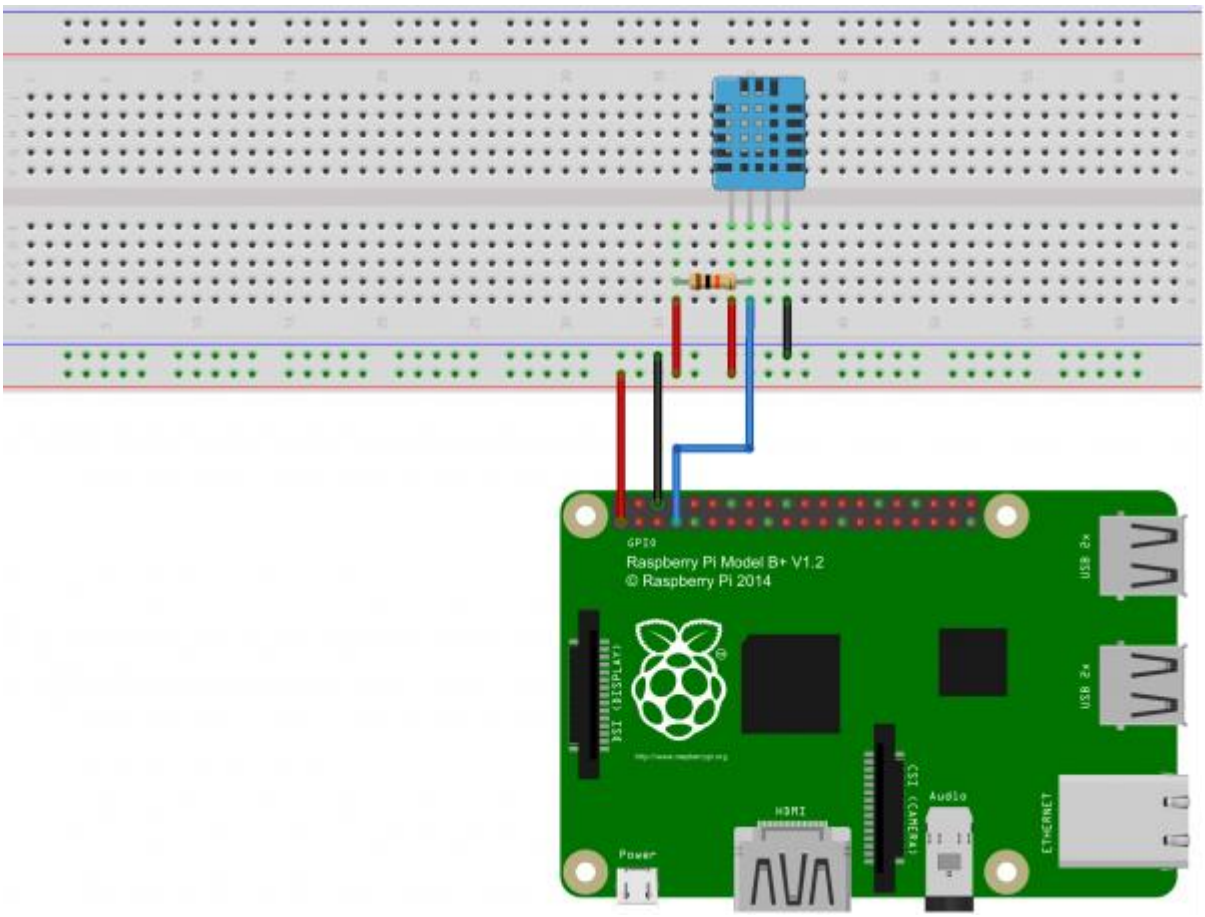

Рисунок 3.6 – Схема підключення датчика вологості та температури DHT-11

Технічні характеристики датчика вологи:

- Вимірювання температури і відносної вологості
- низький рівень споживання енергії
- Вимірювання температури в діапазоні 0 ° C - 50 ° C (похибка ± 2 ° C)
- Вимірювання вологості в діапазоні 20% - 90% (похибка ± 5%)
- Крок вимірювання температури 1 ° C
- Напруга живлення 3 - 5,5В
- Максимальний споживаний струм 2,5мА
- Струм в режимі очікування: 150нА

Плюси використання:

- Поєднує в собі два датчика: датчик температури (в нашому випадку завдяки великій похибці не використовується) і датчик вологості
- Недороге рішення для отримання температури і вологості
- Прості у використанні
- Прості в підключенні

Мінуси використання:

- Велика похибка вимірювань
- Чи не вимірюють температуру нижче 0 ° C
- Довгий час відгуку: раз в дві секунди

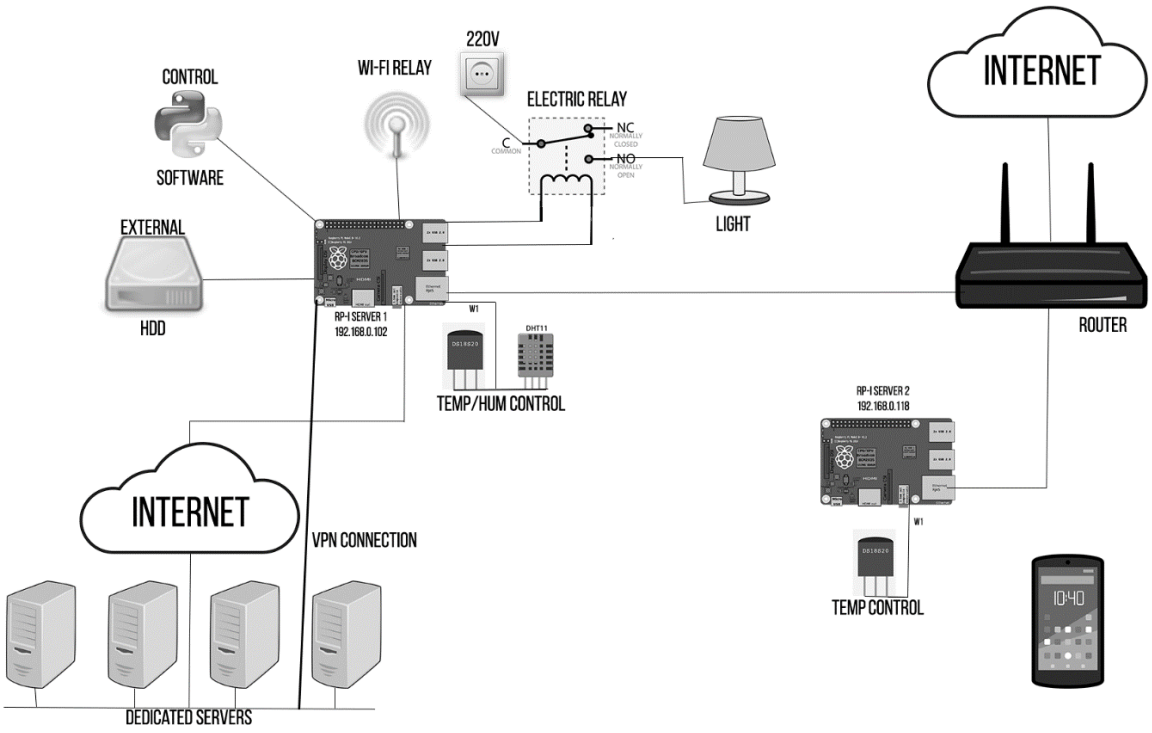

Рисунок 3.7 – Тестова модель системи керування мережею

Ще одним елементом системи є другий комп'ютер Raspberry Pi під керуванням серверної версії операційної системи Debian Linux. Цей елемент отримує, та оброблює перед відправкою на основний комп'ютер проміжні дані з під'єднаних до нього датчиків. Традиційно смартфон служить у якості пульта керування системою з можливістю управління навіть при знаходженні поза

локальною мережею.

Завдяки модульній системі як апаратного так і програмного комплексу, модель відрізняється високим рівнем масштабованості, що дозволяє додавати нові компоненти без критичних апаратних та програмних змін.

Автентифікація користувача здійснюється за допомогою API Telegram. Web-інтерфейс не передбачений задля забезпечення високого рівня захисту системи – усі дані обробляє програмно конфігурований робот який і являє собою ядро системи.

В якості датчика якості диму ми використали MQ-3 датчик, що може виявити різні гази при кімнатній температурі. MQ-3 може розпізнавати метан, бутан, скраплений газ, алкоголь, етанол і дим і т.д. Ми використовуємо напругу 5В як вихід. Це занадто багато для GPIO, тому ми ще використовуємо логічний перетворювач рівня (TTL), який знижує напругу.

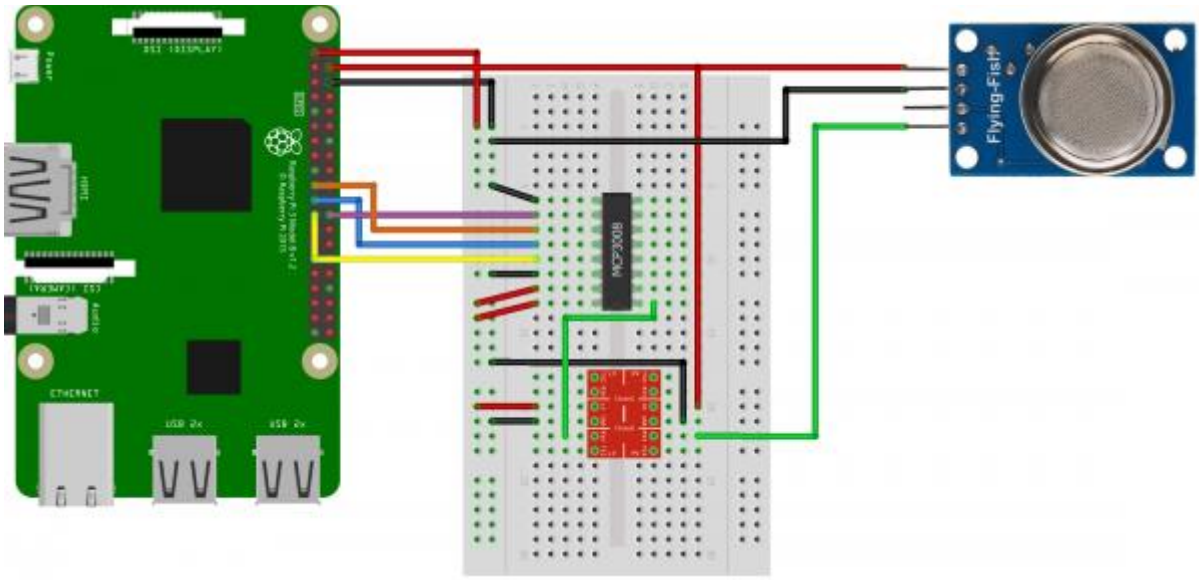

Рисунок 3.8 – Схема підключення датчика якості газу MQ-3

3.4 Опис програмної частини моделі автоматизованого управління

Зв’язок між пристроями та ядром системи відбувається по різним протоколах та за допомогою різних модулів.

Для управління модулем Rasberry основним варіантом є встановлення фреймворк Z-Way (від виробника Rasberry) на Rasberry Pi. Використання є досить простим, а інтерфейсу більш ніж достатньо для ручного управління /

управління будь-яким пристроєм у мережі Z-Wave.

Наприклад виклик даних з датчика температури виконується за допомогою протоколу Z-Wave та модулю rtemp.sh реалізованого на мові Bash. Датчик відправляє «сирі» дані, тому програма обробляє їх через математичні операції. Оброблену інформацію с датчика, яка була отримана напряму з пристрою за його адресом в файловій системі операційної системи Linux (у випадку тестової моделі, що розглядається у роботі це - /sys/bus/w1/devices/10-000802e7468d/w1_slave). Далі наведено код програми:

```
#!/bin/bash
while true; do
t1=$(($(cat /sys/bus/w1/devices/10-000802e7468d/w1_slave | grep t | cut -b 30-)/1000));
if ping -c 1 192.168.0.118 &> /dev/null
then
t2=$(($(sshpass -p "erjopsw3j9s" ssh -o StrictHostKeyChecking=no obr@192.168.0.118 'cat /sys/bus/w1/devices/28-0000075936f6/w1_slave | grep t | cut -b 30-')/1000))
echo "Outdoor temp = $t2 C" > var/outdoor
else
echo "Outdoor temp = N/A C" > var/outdoor
fi
#t=$((t/1000));
now=$(date);
echo -e "$now \nRoom temp = $t1 C" > /home/pi/Scripts/var/temp
sleep 2m
done
```

Після перевірки доступності другого комп'ютера Raspberry Pi за адресом 192.168.0.118 програма за допомогою утиліти sshpass виконає на ньому

аналогічні дії, що і на першій системі. Після обробки усіх даних, та приведення їх в вид який буде розбірливим для людини, програма записує їх у файл який далі, по запиту авторизованого користувача, буде прочитаний ядром керування, та оброблений ботом для виведення його на інтерфейс людино-машинної взаємодії. Частина коду обробки даних роботом наведена нижче:

```
with open ('/home/pi/Scripts/var/temp') as temp:
t1=temp.readlines()
temp.close()
with open ('/home/pi/Scripts/var/outdoor') as temp:
t2=temp.readlines()[0]
temp.close()
```

Іншим під’єднаним до GPIO системи приладом є реле, яке перемикає вже під'єднаний пристрій по мережі 220 вольт. Для обробки статусу приладу, тобто позиції віртуального вимикача, використовується обробник позицій вимикача який запрограмований напряму в ядрі, для кращої взаємодії з приладом. Частина коду обробки позицій наведена нижче:

```
def light(message):
if str(message.chat.id) == "163022030" or str(message.chat.id) == "375195105":
#Файл з статусом
fn='/var/tele/telepi/misc/light.log'
#Перевірка статусу
if os.path.exists(fn):
filelas=open(fn, 'r')
state=filelas.readlines()[0]
elif not os.path.exists(fn):
filelas=open(fn, 'w+')
filelas.write("1")
state="1"
```

```
filelas.seek(0)
filelas.close

#Вмикання-вимикання
if state == "1": #Якщо вимкнено, то вмикаємо
bot.send_message(message.chat.id, "Light is on")
file1= open(fn,"w+")
file1.write("0")#Запис статусу у файл
file1.seek(0)
file1.close()
GPIO.setup(pin, GPIO.OUT)#ініціалізація піна
GPIO.output(pin, 0)
#GPIO.cleanup()
elif state =="0": #Якщо включено, то вимикаємо
bot.send_message(message.chat.id, "Light is off")
file1= open(fn,"w+")
file1.write("1") #Запис статусу у файл
file1.seek(0)
file1.close()
GPIO.setup(pin, GPIO.OUT)#ініціалізація піна
GPIO.output(pin, 1)
#GPIO.cleanup()
```

У кожному модулі взаємодії контроллера з периферією прописані коди, та алгоритми виявлення помилок, що додає системі можливостей відмовостійкості

Схема підключення периферії до GPIO приведена нижче:

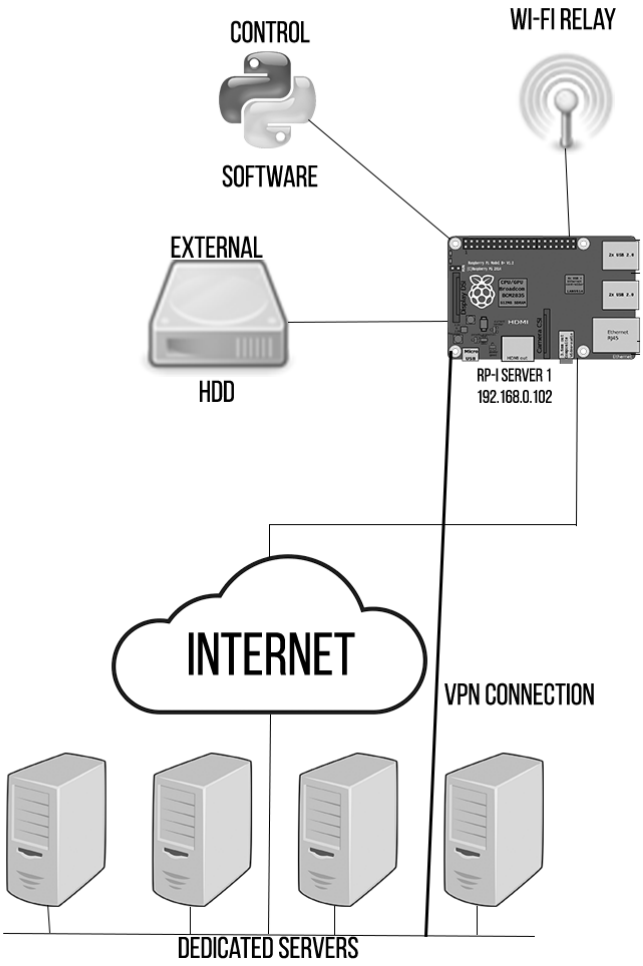

Рисунок 3.9 - З’єднання серверів з центральною машиною

Задля забезпечення відмово-стійкості системи був розроблений багато-потоковий алгоритм самовідновлення при відмові з’єднання та інших негараздів:

```
def polling():
while True:
try:
bot.polling(none_stop=True)
except:
bot.send_message("163022030", "Lost connection from Telegram Api! Bot was reconnected 2 seconds ago...")
```

```
logger.error('\nLost connection. Reconnect in 2 sec')
```

Також для менеджменту серверів використовується протокол SNMP. При використанні SNMP один або більше адміністративних комп'ютерів (де функціонують програмні засоби, звані менеджерами) виконують відстеження або управління групою хостів або пристроїв в комп'ютерній мережі. На кожній керованій системі є постійно запущена програма, яка називається агент, яка через SNMP передає інформацію менеджеру. Для ефективного моніторингу мережевої активності SNMP спирається на архітектуру, що складається з наступного:

Керовані пристрої: від принтерів та робочих станцій до таких ресурсів, як маршрутизатори та комутатори, у мережі організації є багато пристроїв, якими потрібно керувати та контролювати їх. Керовані пристрої можна налаштувати за допомогою вузлів SNMP, які дозволяють їм взаємодіяти з іншими мережевими компонентами.

Агент: Загальне управління SNMP покладається на систему локальної інформації, яка збирається та передається. Це відбувається за допомогою агентів, програм, прив’язаних до локальних пристроїв з метою збору, зберігання та сигналізації про наявність даних із цих середовищ.

Станція управління мережею: Це база, яка ділиться між агентами та менеджерами SNMP, і вона забезпечує функцію пам'яті та обробки для управління мережею.

Завдяки гнучкості, універсальності та масштабованості системи, такий програмно-апаратний комплекс можна впровадити у будь-якій мережі, великій, чи малій-домашній, для інтелектуального контролю за приладами та мережевими пристроями, адмініструванню локальних мереж, серверів, великих районних мереж тощо.

Порівнявши завдання, які ставляться перед системами розумного будинку і розумного будинку, можна прийти до висновку, що для будинку

можуть бути вирішені завдання систем все-таки будуть складнішими, а функціонал - більш розширено. Можливо, в цьому і криється те, що до теперішнього часу розумний будинок - це скоріше певна недосяжна або, більш коректно - складно реалізовується мета, а проявляються лише рідкісні її прояви. Всі причини цього, якщо розібратися, криються в завдання та цілі такого будинку, а також його ціною.

У порівнянні з комерційним будівлею або приміщенням, де перед початком експлуатації можна провести інструктаж з техніки безпеки і особливостям автоматизованих і автоматичних систем споруди, то для розумного будинку це вже зробити буде значно складніше. Зазвичай споживачі не люблять розбиратися з деталями і вчяться використовувати і застосовувати свою розумну електроніку інтуїтивно. З точки зору споживачів, а так і повинно бути, такий підхід природний, але варто зауважити, що тільки зовсім недавно стали з'являтися системи, які є безпечними для оточуючих і при цьому мають зручний і інтуїтивно зрозумілий інтерфейс. Тим більше, що жителями розумного будинку, а точніше його кімнат, цілком можуть стати діти, домашні тварини і нема про що не підозрюють гості і так далі.

Розроблена система піддається швидкій модифікації завдяки модульній системі, тому при переносі системи-керівника на іншу мережу чи додаванні нових модулів, контролер швидко зможе почати керування новими елементами мережі.

Якщо фізичні мережі зв'язку добре задокументовані і стандартизовані, то внутрішні протоколи тих же смарт-ламп або інших пристроїв де-факто реалізуються і документуються всередині компанії-виробника або навіть відокремлені на рівні її однієї лінійки пристроїв. Але не все так погано для кінцевого користувача. Останнім часом можна спостерігати чудову тенденцію відкриття внутрішніх протоколів обміну даними пристроїв у вигляді програмних API. На основі таких відкритих систем вже зараз можна інтегрувати пристрою різних постачальників в єдину інфраструктуру розумного будинку.

ВИСНОВКИ

В магістерській роботі було досліджено роботу інтелектуальних систем, персональних помічників, архітектуру та принципи побудови «розумного дому». Принципи роботи та впровадження технології «Інтернету речей». У ході роботи були виконані наступні завдання:

1. Дослідити поняття та теорію побудови інтелектуальних систем на основі яких будуються та розробляються персональні помічники. В ході досліджень було виявлено, що в сучасних технологіях прийняття рішень, інтелектуальною системою називають інформаційно-обчислювальну середу з інтелектуальною підтримкою, яка може вирішувати задачі без участі людини, тобто особи, котра приймає рішення.
2. Порівняти характеристики концепцій «розумний дім». Сьогодні «розумний дім» - це той, який поєднує в собі найсучасніші технологічні досягнення в IoT хмару, з’єднання, аналітику та фізичний дизайн, що забезпечує кращу видимість, більшу ефективність та контроль, а також більш задоволених, продуктивних людей. Це динамічний, чутливий об'єкт, який адаптується до змінених умов через пов'язані системи будівництва та мережу датчиків і пристроїв.Основою розумного будівництва є мережа пов'язаних речей, яка автоматично визначає та захоплює величезну кількість цінної інформації про фізичне середовище та його мешканців, таких як температура, світло, заселеність і навіть біометрія. Ці дані підключаються до інших систем, аналізуються та використовуються для більш обґрунтованих дій та рішень. До них відносяться автоматичні дії, як-от проактивні сповіщення, коли обладнання ризикує зазнати невдачі, а також кращі рішення щодо використання ресурсів, простору тощо.
3. Проаналізувати методи розробки інтелектуальних персональних помічників. В ході роботи був проведений аналіз в результаті якого було виявлено, що для побудови системи інтелектуального помічника найкраще

використовувати Embedded системи у якості апаратної частини та скриптові мови у якості програмного.

4. Розробити універсального персонального помічника з можливістю управління мережевою інфраструктурою та периферійними пристроями. У ході виконання завдання був розроблений персональний помічник на основі Embedded системи з широкими можливостями масштабування та елементами інтелектуального керування.

Дослідження та розробки, проведені в даній роботі, представляють технологічні рішення, які вже готові до впровадження в сучасних IT-системах.

ПЕРЕЛІК ПОСИЛАНЬ

1. Інтернет речей (Internet of Things, IoT) [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://www.it.ua/knowledge-base/technology-innovation/internet-veschej-internet-of-things-iot (дата звернення 10.10.2019)
2. Коротка історія інтернету речей [Електронний ресурс]: [Веб-сайт]. - Режим доступу: httpshttps://studway.com.ua/internet-rechey/ (дата звернення 15.10.2019)
3. IoT Tutorial for Beginners [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://data-flair.training/blogs/iot-tutorial (дата звернення 22.10.2019)
4. IoT Technology & Protocols [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://data-flair.training/blogs/iot-technology (дата звернення 22.10.2019)
5. IoT [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://1sheeld.com/iot-understanding-concepts (дата звернення 17.11.2019)
6. Important Roles & Application of IoT in Education [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://data-flair.training/blogs/iot-in-education (дата звернення 27.10.2019)
7. Government Applications in IoT [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://data-flair.training/blogs/government-applications-in-iot (дата звернення 02.11.2019)
8. Гололобов В. «Умный дом» своими руками – 2007 – 416с.
9. Smart Home Concepts [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://mydigitalhomesolutions.com/smart-home-concepts-the-whys-hows-and-whats/ (дата звернення 20.11.2019)
10. Internet of Things Applications in Building & Houses - [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://data-flair.training/blogs/internet-of-things-applications-in-building/ (дата звернення 23.11.2019)
11. M. Domb «Smart Home Systems Based on Internet of Things» - 2018 – 117с.
12. Wireless protocols showdown: Riding the Z-Wave - [Електронний ресурс]: [Веб-сайт]. - Режим доступу: https://silvair.com/blog/wireless-protocols-showdown-riding-z-wave/
13. Виктор Петин - Микрокомпьютеры Raspberry Pi. Практическое руководство – 2015 – 511 с.;

ДОДАТКИ

```
import config # конфіг
import telebot # бот-апі
import time # час (використовуємо для затримки)
import re # regxp пошук збіжності
import threading # багатопоточність
import pycurl # курл
import os # Бібліотека для системи
import RPi.GPIO as GPIO #gpio lib
import logging as logger
from io import BytesIO

bot = telebot.TeleBot(config.token)
a=0 #прапор для старого моніторингу PC
n=0 #Безкінечний цикл
pin = 27 #GPIO27
buffer = BytesIO()
#c = pycurl.Curl()
#GPIO.cleanup()
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)

@bot.message_handler(commands=["start"])
def start(message):
    bot.send_message(message.chat.id, "Hello. It's Nastia's bot v0.31.")

@bot.message_handler(commands=["help"])
def help(message):
    bot.send_message(message.chat.id, "Nastia's Dev Bot v0.31\nCurrent
commands:\n1./temp - Show Room Temp\n2./hum - Show Room
Humidity\n3./servstat - Server monitoring\n4./ip - Show home ip\n5./light - light
On-Off\nAlso shows PC status\n6./id - get your Telegram ID")

@bot.message_handler(commands=["temp"])
def temp(message):

  if str(message.chat.id) == "163022030" or str(message.chat.id) == "375195105":

    with open ('/home/pi/Scripts/var/temp') as temp:
     t1=temp.readlines()
    temp.close()
    with open ('/home/pi/Scripts/var/outdoor') as temp:
     t2=temp.readlines()[0]
```

```
    temp.close()

    bot.send_message(message.chat.id, t1[0]+t1[1]+t2)

  else:
    bot.send_message(message.chat.id, "Access denied")
    bot.send_message("163022030", "ID: "+str(message.chat.id)+"\nName:
"+str(message.chat.first_name)+" "+str(message.chat.last_name)+"\nText:
"+message.text)
#датчик вологості
@bot.message_handler(commands=["hum"])
def hum(message):
  if str(message.chat.id) == "163022030" or str(message.chat.id) == "375195105":
    with open ('/home/pi/Scripts/var/hum') as hum:
     bot.send_message(message.chat.id, "Humidity:"+hum.readlines()[0])
    hum.close()
  else:
    bot.send_message(message.chat.id, "Access denied")
    bot.send_message("163022030", "ID: "+str(message.chat.id)+"\nName:
"+str(message.chat.first_name)+" "+str(message.chat.last_name)+"\nText:
"+message.text)
@bot.message_handler(commands=["id"])
def id(message):
    bot.send_message("163022030", "ID: "+str(message.chat.id)+"\nName:
"+str(message.chat.first_name)+" "+str(message.chat.last_name)+"\nText:
"+message.text)
    if not str(message.chat.id) == "163022030":
     bot.send_message(message.chat.id, "ID: "+str(message.chat.id)+"\nName:
"+str(message.chat.first_name)+" "+str(message.chat.last_name)+"\nText:
"+message.text)

@bot.message_handler(commands=["light"])
def light(message):
 if str(message.chat.id) == "163022030" or str(message.chat.id) == "375195105":
   fn='/var/tele/telepi/misc/light.log'
   #Проверка файла
   if os.path.exists(fn):
      filelas=open(fn, 'r')
      state=filelas.readlines()[0]
   elif not os.path.exists(fn):
      filelas=open(fn, 'w+')
      filelas.write("1")
      state="1"

   filelas.seek(0)
```

```
   filelas.close

   if state == "1": #Якщо останній статус був Викл то включаємо
      bot.send_message(message.chat.id, "Light is on")
      file1= open(fn,"w+")
      file1.write("0")
      file1.seek(0)
      file1.close()
      GPIO.setup(pin, GPIO.OUT)#ініціалізація піна
      GPIO.output(pin, 0)
      #GPIO.cleanup()
   elif state =="0": # Якщо останній статус був Вкл то виключаємо
      bot.send_message(message.chat.id, "Light is off")
      file1= open(fn,"w+")
      file1.write("1")
      file1.seek(0)
      file1.close()
      GPIO.setup(pin, GPIO.OUT)# ініціалізація піна
      GPIO.output(pin, 1)
      #GPIO.cleanup()
 else:
   bot.send_message(message.chat.id, "Access denied")
   bot.send_message("163022030", "ID: "+str(message.chat.id)+"\nName:
"+str(message.chat.first_name)+" "+str(message.chat.last_name)+"\nText:
"+message.text)

@bot.message_handler(commands=["ip"])
def ip(message):
 if str(message.chat.id) == "163022030":
    c = pycurl.Curl()
    c.setopt(c.URL, 'http://icanhazip.com/')
    c.setopt(c.WRITEDATA, buffer)
    c.perform()
    c.close()

    body = buffer.getvalue()
    bot.send_message("163022030", body.decode('iso-8859-1'))
    body = None
    buffer.seek(0)
 else:
    bot.send_message(message.chat.id, "Access denied")
    bot.send_message("163022030", "ID: "+str(message.chat.id)+"\nName:
"+str(message.chat.first_name)+" "+str(message.chat.last_name)+"\nText:
"+message.text)
```

```
@bot.message_handler(commands=["servstat"])
def servstat(message):
 with open ('/home/pi/Scripts/var/relay') as f1:
       relay=f1.readlines()[0]
 f1.close()

 with open ('/home/pi/Scripts/var/relay2') as f2:
       relay2=f2.readlines()[0]
 f2.close()

 with open ('/home/pi/Scripts/var/nastia') as f3:
       nastiach=f3.readlines()[0]
 f3.close()

 with open ('/home/pi/Scripts/var/kovtun) as f4:
       kovrosol=f4.readlines()[0]
 f4.close()

 with open ('/home/pi/Scripts/var/pc') as f5:
       pc=f5.readlines()[0]
 f5.close()

 if str(message.chat.id) == "163022030":
  bot.send_message("163022030","VPS relay.kovtun.info (VPN): "+relay+"\nVPS
relay2.obrest.info (1stm,proxy): "+relay2+"\nServer nastia.kovtun.info
(nastia.kovtun): "+nastia+"\nVPS nastia.kovtun.info (test VPS): "+kovtun+"\nPC:
"+pc)
 elif str(message.chat.id) == "331935388":
  bot.send_message("331935388","Server nastia.kovtun: "+nastia)
else:
    bot.send_message(message.chat.id, "Access denied")
    bot.send_message("163022030", "ID: "+str(message.chat.id)+"\nName:
"+str(message.chat.first_name)+" "+str(message.chat.last_name)+"\nText:
"+message.text)

def monit(addr, name):
   #файл зі статусом
   fn='/var/tele/telepi/misc/'+addr+'.log'
   fh='/home/pi/Scripts/var/'+addr

   #Перевірка файлу
   if os.path.exists(fn):
      filelas=open(fn, 'r')
      state=filelas.readlines()[0]
   elif not os.path.exists(fn):
```

```
      filelas=open(fn, 'w+')
      filelas.write("0")
      state="0"

  # if not os.stat(fn).st_size==0:
  #  state = file.readlines()[0]
  # else:
  #  state = "0"
   filelas.seek(0)
   filelas.close

   global string
   fileres=open(fh, 'r')
   for line in fileres:
     string=line
   fileres.seek(0)
   fileres.close
   result = re.match('online',string,re.IGNORECASE)
   resoff = re.match('offline',string,re.IGNORECASE)
   if result is not None:
       if state=="0":
            bot.send_message("163022030", name+" is Online")
            if addr=="nastia":
              bot.send_message("331935388", name+" is Online")
            file1= open(fn,"w+")
            file1.write("1")
            file1.seek(0)
            #print(file1.readlines()[0])
            file1.close()
def pc():
   global a
   global pcs

   pcf=open(r'/home/pi/Scripts/var/pc')
   for line in pcf:
     pcs=line
   pcf.close

   result = re.search('online',pcs,re.IGNORECASE)

   if result is not None:
       if a==0:
            bot.send_message("163022030", "PC is Online")
            a=1
   elif result is None:
```

```
        if a==1:
            bot.send_message("163022030", "WARNING! PC is Offline")
            a=0
    return()

def cycle():
    bot.send_message("163022030", "Bot is Online")
    while n<10:
     pc()
     monit("relay","VPS relay.kovtun.info (VPN, squid)")
     monit("nastia","Server nastia.kovtun.info (nastia.kovtun)")
     monit("relay2","VPS relay2.kovtun.info (1stm, squid)")

def polling():
   while True:
     try:
        bot.polling(none_stop=True)
     except:
        bot.send_message("163022030", "Lost connection from Telegram Api! Bot
was reconnected 2 seconds ago...")
        logger.error('\nLost connection. Reconnect in 2 sec')
        time.sleep(2)


threading.Thread(target=polling).start()
threading.Thread(target=cycle).start()
#threading.Thread(target=bot.polling(none_stop=True)).start()
```