Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“____” _____ 2025 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти

на тему

Дослідження та програмна реалізація системи хмарного керування пристроями комплексу рішень «розумний дім»

Виконав здобувач вищої освіти
ІІ курсу, групи КІ-24М
ОПП «Комп’ютерна інженерія»
спеціальності 123 «Комп’ютерна інженерія»
_____ Герасімов О.І.
«____» _____ 2025 р.

Керівник проекту
кандидат технічних наук, доцент
_____ Марченко К.М.
«____» _____ 2025 р.
Рецензент _____
_____

м. Кропивницький

АНОТАЦІЯ

Герасімов О.І. Дослідження та програмна реалізація системи хмарного керування пристроями комплексу рішень «розумний дім». 123 Комп'ютерна інженерія. Центральноукраїнський національний технічний університет. Кропивницький. 2025.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для хмарного керування розумним будинком.

Метою розробки є дослідження та програмна реалізація системи хмарного керування розумним будинком.

Об'єктом дослідження є процес забезпечення хмарного керування «розумним будинком».

Предметом дослідження є методи забезпечення хмарного керування системами розумного будинку.

Методи дослідження базуються на методах теорії кодування, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи хмарного керування розумним будинком.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ архітектури IBM PC з ОС Windows XP/Vista/7/8/10 та системи Android.

Програму розроблено в середовищі C++ та PHP.

Ключові слова: комп'ютерна інженерія, IoT, MQTT, smart house.

ABSTRACT

Herasimov O.I. Research and software implementation of the cloud-based device management system of the "smart home" solution complex. 123 Computer Engineering. Central Ukrainian National Technical University. Kropyvnytskyi. 2025.

In this final qualification work for the second (master's) level of higher education, software is developed that is intended for cloud management of a smart home.

The purpose of the development is the research and software implementation of the smart home cloud management system.

The object of the study is the process of providing cloud-based management of a smart home.

The subject of the study is methods for providing cloud-based management of smart home systems.

Research methods are based on methods of coding theory, methods of mathematical statistics, methods of software development.

The result is the software implementation of a smart home cloud management system.

In the process of working on the software model, an analysis of existing hardware and software was performed. All components of the developed software are fully described.

User friendly interface developed. Instructions for working with software are given.

The program can be used on the PC PC architecture of IBM PC with Windows XP / Vista / 7/8/10 and Android.

The program is developed in C ++ and PHP environment.

Keywords: computer engineering, IoT, MQTT, smart house.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

GPS (Global Positioning System) - система для отримання місцезнаходження об’єкта.

PKI (Public Key Infrastructure) - інфраструктура відкритих ключів.

GSM (Group Special Mobile) - глобальна система мобільного зв’язку.

SOAP (Simple Object Access Protocol).

MQTT (Message Queue Telemetry Transport).

IDE (Integrated Development Environment) - вбудована система розробки.

API (Application Programming Interface) - прикладний програмний інтерфейс.

IaaS - інфраструктура як послуга.

Pub/Sub (Publish–Subscribe).

SDK (Software Development Kit).

XML (Extensible Markup Language).

IoT (Internet of Things).

ВСТУП

Актуальність теми. Протягом останніх десяти років ми дедалі більше звикаємо керувати різними сферами нашого життя за допомогою технологій. Онлайн-банкінг, інтернет-магазини та інші цифрові сервіси суттєво полегшили наше повсякдення. Нині подібна технологічна революція відбувається й у наших оселях. «Розумний дім» - це житло, у якому майже кожну систему можна контролювати цифровим шляхом. Дверний дзвінок, освітлення, безпека, опалення, водопостачання, а навіть керування шторами - усе це може працювати дистанційно завдяки сучасним технологіям.

Такі рішення роблять побут значно простішим та комфортнішим, даруючи впевненість у тому, що дім функціонує належним чином навіть тоді, коли ви надовго від'їжджаєте. Смарт-технології дозволяють скоротити кількість кабелів і пультів, якими раніше доводилося користуватися для управління музикою чи телевізором. Частина розумних систем орієнтована насамперед на безпеку: наприклад, комплексні системи відеоспостереження дають змогу контролювати ситуацію вдома з будь-якої точки світу. Доступні також рішення, які захищають дім від пожеж чи затоплень. Розумний будинок надає більше контролю та інформації про все, що відбувається у вашій оселі.

Технологію розумного будинку можна впроваджувати поступово, залежно від ваших потреб і пріоритетів. Якщо для вас у першу чергу важливо відкривати штори голосовою командою - почніть саме з цього! Проте сучасні тенденції спрямовані на створення єдиної інтегрованої системи, що об'єднує всі елементи житла в одному центрі керування.

Ідея smart home формувалася у США ще з 1950-х років XX століття. Тодішньою метою було створення системи, яка могла б самостійно контролювати життєві процеси в будинку - те, що раніше існувало лише у фантастичних романах. Передусім це була «система комфорту» для

забезпечених американців. У 1970-х роках Вашингтонський інститут інтелектуального будинку дав офіційне визначення поняття smart home. Під ним розуміли здатність системи класифікувати різні ситуації в оселі, адаптуватися до них і автономно приймати рішення щодо реагування. Крім того, система мала повністю контролювати всю екосистему будинку та керувати іншими незалежними підсистемами. Іншими словами, «система комфорту» мала стати центральною та єдиною.

Концепцію активно підтримали як на соціальному, так і на фінансовому рівнях. Багато компаній з ентузіазмом почали експериментувати з кабельними мережами та електронікою, намагаючись створити пристрої, здатні, наприклад, автоматично ввімкнути чи вимкнути світло залежно від присутності людини. У результаті численних спроб і розробок у 1975 році з'явився перший універсальний стандарт для створення простої домашньої автоматизованої системи - X10. Його створили інженери шотландської компанії Pico Electronics. Варто зазначити, що стандарт X10 виявився надзвичайно успішним і досі використовується.

Шотландські фахівці не зволікали: вони заснували компанію X10 USA і вийшли з новою технологією на американський ринок. А вже у 1978 році, співпрацюючи з компанією Leviton, вони створили повноцінну систему автоматизованого управління побутовою технікою через електромережу. Це стало справжнім проривом, і 1978 рік увійшов в історію як ключовий етап розвитку концепції «розумного будинку». Саме тоді почалося стрімке зростання smart-технологій. З'явилися альтернативні рішення, такі як EIB (згодом KNX), протоколи IEC61158, LonTalk та інші.

За майже сорок років технології розумного дому здійснили величезний стрибок у розвитку, ставши доступними широкому колу інженерів і користувачів. Було створено безліч автоматизованих систем контролю за кліматом, електрикою, опаленням, безпекою тощо. Не менш важливу роль

відіграють і численні smart-ґаджети, що також входять до сучасної інноваційної екосистеми.

Мета й завдання дослідження. Метою даної роботи є вивчення та програмна розробка системи хмарного керування пристроями в комплексі рішень «розумний дім».

Для досягнення цієї мети сформовано план дослідження, що включає такі завдання:

- Провести огляд існуючих систем хмарного управління комплексами «розумний дім»;

- Дослідити принципи роботи систем хмарного керування розумним будинком;

- Виконати програмну реалізацію системи хмарного керування розумним будинком.

Об'єктом дослідження є процес забезпечення хмарного керування «розумним будинком».

Предметом дослідження є методи реалізації систем хмарного керування розумним будинком.

Методи дослідження базуються на методах теорії кодування,методах захисту інформації, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено систему ухмарного керування розумним будинком.

– Проведено огляд технологій зв'язку в системах «Розумний дім».

– Розроблено вітчизняний продукт хмарного керування розумним будинком, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми забезпечують ефективне розв’язання завдань, пов’язаних із реалізацією систем хмарного керування технологіями «розумного будинку».

Достовірність наукових результатів підтверджується теоретичними обґрунтуваннями, даними комп’ютерного моделювання, коректними дослідженнями параметрів у діючій обчислювальній мережі, а також відповідністю отриманих висновків результатам, наведеним у наукових джерелах.

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

Smart Home (домашня автоматизація, розумний будинок) є одним із найбільш перспективних напрямів розвитку сучасних інформаційних та комунікаційних технологій. Завдяки таким системам усі електроприлади в будівлі можуть бути об’єднані в єдину функціональну мережу, якою можна керувати централізовано - за допомогою дисплея користувача або автоматично, відповідно до заданих алгоритмів.

Метою роботи є створення системи оптимізації енергоспоживання для розумного будинку, яка складається з двох основних компонентів:

- пристрою моніторингу та керування (Smart Switch), що передає дані про споживання чи генерацію енергії конкретним приладом домогосподарства та забезпечує можливість його комутації;

- сервера (Smart Dispatcher), який збирає інформацію від усіх пристроїв локальної мережі та приймає рішення щодо зміни її конфігурації, підключення або відключення обладнання залежно від поточного стану системи.

1.2 Область застосування

Система хмарного керування розумним будинком:

- реалізує ключові концепції технологій «розумного дому»;

- містить унікальні функціональні можливості, яких немає в інших подібних системах;

- є простою, недорогою та надійною завдяки мінімалістичному інтерфейсу;

- легко встановлюється та налаштовується будь-яким користувачем;

- за замовчуванням працює в автономному режимі, проте може бути налаштована для взаємодії з хмарними сервісами.

Отже, враховуючи зазначені переваги та особливості, дослідження й програмна реалізація системи хмарного керування розумним будинком із підсистемою захищеної передачі даних є актуальним завданням, яке потребує вирішення в межах цієї магістерської роботи.

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Одним із найпоширеніших сучасних світових трендів у сфері домашніх технологій є система «Розумний будинок», про яку сьогодні чув майже кожен. Ідея про те, що житло може функціонувати самостійно, допомагаючи власнику в побуті - від підігріву їжі до забезпечення безпеки - стала близькою і зрозумілою всім українцям. Проте, тоді як у Європі такими системами користуються близько 10% населення, в Україні це поки що рідкість, адже впровадження технології все ще є дорогим. Хоча рівень «розумності» будинку можна обирати самостійно, адаптуючи його під власні потреби та фінансові можливості.

В Україні ці системи лише починають привертати увагу власників житла, переважно приватних будинків, які зацікавлені насамперед в економії ресурсів. Це стосується і «розумних» квартир, особливо великих, де високі витрати на опалення та електроенергію роблять оптимізацію споживання дуже актуальною. Економія - один із головних аргументів на користь домашніх інтелектуальних систем.

«Розумний будинок» - це мережа датчиків, розміщених по всій площі житла. Вони чутливі до різних параметрів: кліматичні реагують на зміну температури та вологості, світлові - на освітленість і час доби, датчики руху та об’єму визначають наявність сторонніх у приміщенні тощо. Їх можна запрограмувати на індивідуальні комфортні режими: підтримання певної температури в кожній кімнаті, регулювання нагріву води, вимкнення світла в дитячій у визначений час, увімкнення будильника чи навіть тостера. Усі

датчики об’єднані в єдину систему керування - пульт, дані з якого виводяться на планшет або настінний монітор у будинку та синхронізуються зі смартфонами власників.

Таким чином, навіть перебуваючи поза домом, можна дистанційно контролювати стан житла й змінювати параметри роботи систем. Наприклад, якщо власник повертається о 21:00, він може встановити запуск опалення о 20:00, щоб до його приходу в будинку вже було тепло.

Звичайно, як будь-яка технологія, система має як переваги, так і недоліки.

Де знайти якісний «мозок» для вашого будинку

Найкраще встановлювати систему «Розумний будинок» під час масштабного ремонту, щоб мати можливість приховати всі кабелі. Проєктування такої системи повинні виконувати лише кваліфіковані фахівці, які пройшли відповідне навчання та можуть надати сертифікат, що підтверджує їхній професійний рівень. Звичайний електрик, навіть дуже досвідчений, не впорається з подібною задачею. Купувати коробкове рішення і намагатися встановити його самостійно теж не варто - це може призвести до пошкодження побутової техніки. Крім того, компанія-інсталятор має забезпечувати гарантійне обслуговування й ремонт, а її технічні спеціалісти - мати підготовку для підтримки таких систем.

Кожен проєкт «Розумного будинку» створюється індивідуально, відповідно до способу життя та побажань власників. Фірма-інсталятор не лише розробляє проєкт, а й займається прокладанням усіх необхідних комунікацій.

Оскільки монтаж такої системи є дорогим, обирати компанію слід дуже ретельно. Хоча знайти інсталяторів можна в інтернеті, краще звертатися за порадами знайомих. Пропозицій багато, але важливо враховувати досвід: якщо компанія працює менше п’яти років - краще не ризикувати. Якщо ж досвід становить понад 5 років, варто попросити контакти клієнтів для отримання відгуків і не соромитися зв’язатися з ними. Рекомендації в цьому випадку

забезпечують половину успіху. Бажано, щоб обладнання було від європейського або американського виробника з більш ніж 20-річною історією. Наявність сертифікації KNX є додатковим показником якості та надійності, оскільки її отримують лише перевірені виробники.

Інтернет речей та хмарні обчислення

На сьогодні більшість систем «Розумного будинку» не підтримують віддалене керування через Інтернет. Водночас мобільні пристрої з постійним доступом до мережі стали звичайним явищем і практично у кожного користувача.

У 1999 році Кевін Ештон, засновник дослідницького центру Auto-ID Center Массачусетського технологічного інституту, запропонував термін Internet of Things (Інтернет речей). Суть концепції полягає в тому, що сучасні «розумні» пристрої будуть об'єднані в мережу, утворюючи Інтернет речей. Це дозволяє смартфонам, планшетам, телевізорам, різним датчикам та керованим приладам із бездротовими модулями Wi-Fi та Bluetooth взаємодіяти між собою та користувачами.

З поширенням мобільних пристроїв, сумісних із концепцією Інтернету речей, стало можливим віддалене управління системами «Розумного будинку». Це надає ряд переваг:

- безпека: мешканці можуть контролювати будинок або квартиру на відстані, спостерігаючи через камери або моніторячи стан датчиків безпеки (пожежні датчики, датчики відкриття дверей тощо). Функція корисна також для тих, хто часто забуває вимкнути світло чи побутові прилади;

- комфорт: системи «Розумного будинку» часто використовують сценарії автоматичного керування світлом і опаленням. Деякі користувачі віддають перевагу ручному керуванню, і віддалений доступ дозволяє, наприклад, увімкнути освітлення, побутові прилади або опалення ще до приходу додому.

Віддалене управління реалізується за допомогою хмарних обчислень, що забезпечують користувачам доступ до мережевих ресурсів, сервісів і додатків

незалежно від місцезнаходження. Існує кілька моделей хмарних обчислень, і для систем «Розумного будинку» найбільш підходить модель SaaS (Software as a Service, програмне забезпечення як послуга). Ця модель передбачає доступ користувача до програмного забезпечення через Інтернет без потреби його встановлення або оновлення та без турбот про працездатність обладнання.

При впровадженні хмарних обчислень у «Розумний будинок» можливі два варіанти:

- контролер (сервер) знаходиться у хмарі, а не в будинку. Це дозволяє керувати системою з будь-якої точки світу при наявності Інтернету.

- контролер розташований у будинку (рисунок. 2.1), але віддалене керування здійснюється через хмарний сервер, де встановлено все програмне забезпечення. У цьому випадку домашній контролер відповідає лише за забезпечення доступу до Інтернету, що зменшує вимоги до його технічних характеристик.

Також у другому варіанті не потрібна заміна існуючого обладнання для впровадження віддаленого управління - достатньо забезпечити підключення контролера до хмарного сервера.

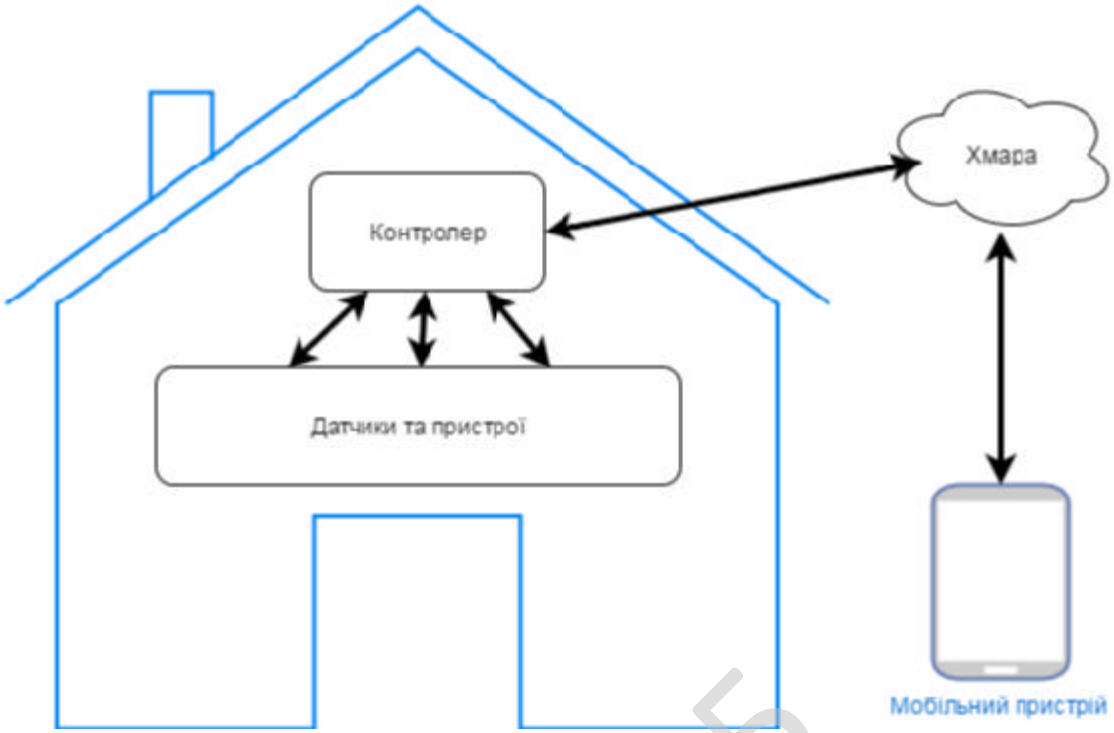

Рисунок 2.1 - Контролер розташовуваний в будинку

Безпосередньо віддалене управління системами розумного будинку можливо здійснювати або через веб-браузер, або через спеціальне мобільний додаток. Варто відзначити ще одну важливу деталь. Багато сучасні пристрої, що використовуються в розумному будинку, як кінцеве обладнання, так і керуючі пристрої, працюють по своїх власних протоколах передачі даних і, крім того, можуть взаємодіяти з Інтернет-сервісами тільки через свої API. Тому часто немає можливості розширити систему розумного будинку, наприклад, яким-небудь розумним холодильником, або ж додати до неї пристрої, що працюють за іншими протоколами передачі даних. Однак за допомогою хмарного сервісу, який буде надавати загальний інтерфейс управління всіма системами, а різні пристрої будуть взаємодіяти між собою через хмару, з'являється можливість використання пристроїв від різних виробників з різними протоколами передачі даних. В результаті застосування хмарних технологій в системах розумного будинку дозволить зробити їх набагато більш гнучкими, а також дозволить скоротити витрати на обслуговування і розширення системи.

Протокол взаємодії хмарного сервера з пристроями розумного будинку: XML/SOAP

Щоб хмарний сервер та пристрої «Розумного будинку» ефективно взаємодіяли, вони повинні «спілкуватися» однією мовою. Одним із способів забезпечити це є обмін даними у форматі XML.

Протокол SOAP (Simple Object Access Protocol - простий протокол доступу до об'єктів) використовує XML для передачі повідомлень і є одним із популярних рішень для організації взаємодії хмарного сервера з різними пристроями. Основна перевага SOAP полягає у можливості забезпечувати безперервну взаємодію веб-сервісу з пристроями, які працюють за різними протоколами передачі даних.

Додаткові переваги використання SOAP у порівнянні з іншими форматами передачі даних:

- XML-структури даних у SOAP легко кодувати так само, як і прості скалярні типи;
- SOAP надає додаткові інструменти для реалізації безпеки, трасування та інших сервісних функцій;
- Існують готові набори інструментів SOAP для різних мов програмування, що спрощує розробку.

Спрощений мережевий протокол MQTT

MQTT (Message Queue Telemetry Transport) - це легкий мережевий протокол, який працює поверх TCP/IP і використовується для обміну даними між пристроями за принципом publish-subscribe.

Протокол був вперше розроблений у 1999 році доктором Енді Станфорд-Кларком (IBM) та Арленом Ніппером (Arcom) і опублікований під ліцензією royalty-free. Версія MQTT 3.1.1 була стандартизована консорціумом OASIS у 2014 році.

Основні переваги протоколу MQTT:

- модель publish-subscribe зручна для роботи з датчиками та IoT-пристроями, дозволяє пристроям публікувати повідомлення і підписуватися на необхідні теми;

- простота використання: легкий програмний блок, що легко інтегрується у складні системи;

- гнучкість контенту: повідомлення можуть містити будь-які дані, заздалегідь не визначені;

- легкість адміністрування;

- зменшення навантаження на канал зв'язку;

- стійкість до нестабільного зв'язку, втрати пакетів чи інших проблем на лінії;

- відсутність обмежень на формат даних, що передаються.

Для роботи MQTT необхідний брокер повідомлень, який відповідає за доставку повідомлень клієнтам, підписаним на певну тему. MQTT використовує методи (дієслова) для вказівки дії над конкретним ресурсом, який може повертати статичні або динамічні дані. Часто ресурс відповідає імені файлу або виходу виконуваного файлу на сервері.

Особливість протоколу полягає у можливості повторного використання даних за допомогою прапора Retain, що дозволяє клієнту отримати останнє повідомлення навіть при нестабільному зв'язку або затримках у мережі.

Висновок

Аналіз розвитку концепції розумного будинку та методів передачі даних показує, що технології пройшли довгий шлях еволюції, віднайшовши оптимальні рішення для роботи систем. Проте розвиток продовжується: великі постачальники хмарних платформ дедалі активніше інтегрують Internet of Things (IoT) у свої сервіси, і про ці нововведення та вектори розвитку буде розказано у наступному розділі.

Аналіз існуючих API хмарного керування розумним будинком

Технології розумного дому та домашньої автоматизації стають дедалі популярнішими. Великі гравці ринку хмарних обчислень активно інтегрують Інтернет речей (IoT) у свої платформи та пропонують широкий спектр рішень для зручного впровадження автоматизації в будь-якому середовищі.

AWS IoT (Amazon)

Платформа AWS IoT дозволяє підключати пристрої до сервісів AWS і інших пристроїв, забезпечує захист даних та безпечну взаємодію між компонентами. Вона підтримує обробку даних від пристроїв, управління ними, а також інтеграцію додатків із пристроями навіть у випадку відсутності постійного підключення до Інтернету.

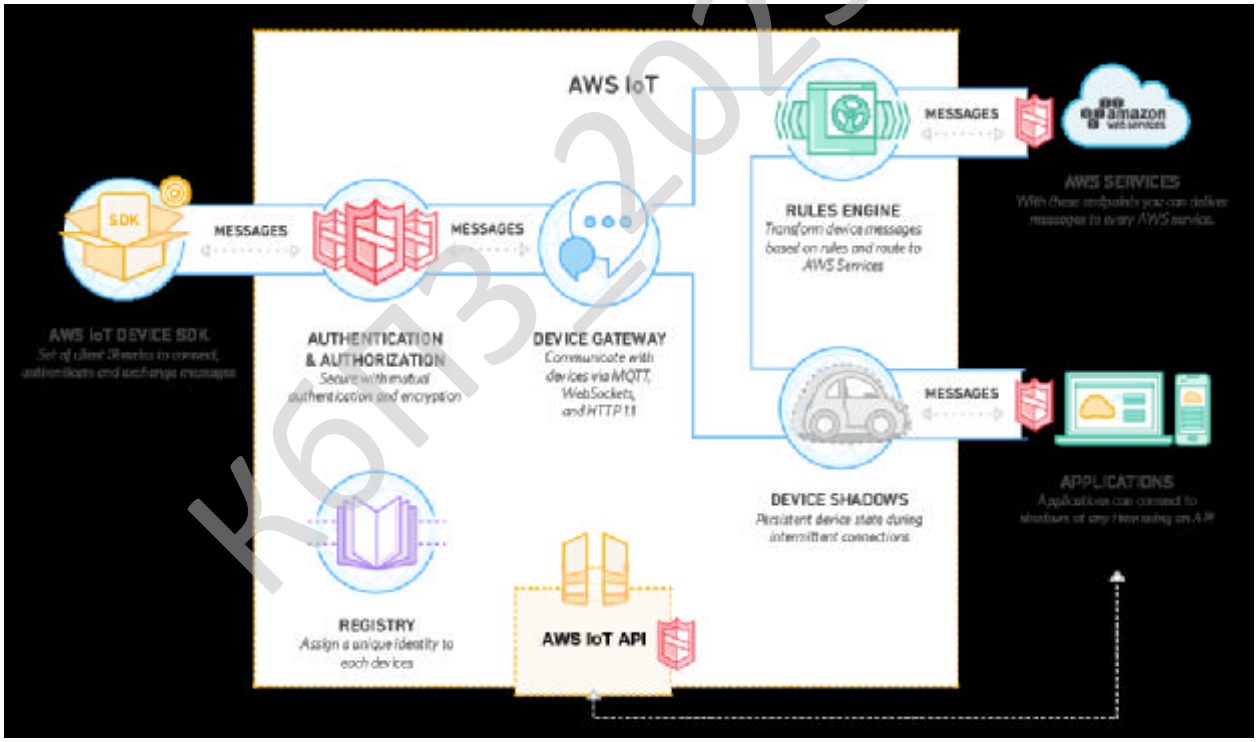

Рисунок 2.2 - Схема роботи AWS IoT

AWS IoT SDK для пристроїв

Платформа AWS IoT надає SDK, що дозволяє швидко та легко підключати апаратні пристрої та мобільні додатки. SDK забезпечує аутентифікацію пристроїв та обмін повідомленнями з платформою через

протоколи MQTT, HTTP або WebSockets. Пакет підтримує мови C, JavaScript та Arduino, містить клієнтські бібліотеки, документацію для розробників та інструкції для виробників щодо перенесення програмного забезпечення. Крім того, можна використовувати альтернативні відкриті SDK або створювати власні рішення.

Шлюз пристроїв

Шлюз пристроїв AWS IoT забезпечує безпечну та ефективну взаємодію між підключеними пристроями та платформою AWS IoT. Він використовує модель Publish-subscribe, що дозволяє реалізувати комунікацію за схемами «один-до-одного» та «один-до-багатьох». У рамках схеми «один-до-багатьох» підключений пристрій може надсилати дані одночасно кільком підписникам певної теми. Шлюз підтримує протоколи MQTT, WebSocket та HTTP 1.1, а також може інтегруватися з пропрієтарними або застарілими протоколами. Крім того, шлюз автоматично масштабується, здатний обслуговувати понад мільярд пристроїв без необхідності додаткової інфраструктури.

Аутентифікація та авторизація

Платформа AWS IoT гарантує взаємну аутентифікацію та шифрування для всіх точок підключення, що забезпечує безпечний обмін даними між пристроями та сервісом лише після підтвердження їхньої ідентифікації. AWS IoT підтримує метод SigV4, а також аутентифікацію за сертифікатами стандарту X.509. Підключення через HTTP можуть використовувати будь-який із цих методів, у той час як MQTT використовує сертифікати, а WebSockets - SigV4.

Сервіс дозволяє застосовувати сертифікати, створені самим AWS IoT, або сертифікати, підписані обраним центром сертифікації. До кожного сертифікату можна прив’язати роль або політику для контролю авторизованого доступу пристроїв і додатків з можливістю відкликання прав без прямої роботи з пристроєм. Керувати сертифікатами та політиками можна через Консоль AWS або за допомогою API.

Крім того, AWS IoT підтримує підключення мобільних додатків користувачів через сервіс Amazon Cognito, який автоматично створює унікальні ідентифікатори для користувачів та надає тимчасові обмежені права доступу до ресурсів AWS.

Реєстр

Реєстр у AWS IoT забезпечує унікальну ідентифікацію пристроїв та дозволяє відстежувати їх метадані, такі як характеристики або функціональні можливості. Кожен пристрій можна зареєструвати у єдиному форматі незалежно від його типу чи способу підключення. Реєстр також підтримує метадані, що описують специфічні можливості пристрою, наприклад, відображення температури датчика у градусах Цельсія або Фаренгейта. Збереження метаданих у реєстрі не потребує додаткових витрат. Щоб дані залишалися актуальними, достатньо хоча б раз на сім років звертатися до запису або оновлювати його.

Фреймворк правил

Движок правил AWS IoT дає змогу створювати IoT-додатки для збору, обробки та аналізу даних, що надходять від підключених пристроїв, а також для виконання дій з цими даними у масштабі без необхідності керувати інфраструктурою. Він аналізує вхідні повідомлення, що публікуються на платформі, перетворює їх і направляє іншим пристроям або хмарним сервісам відповідно до визначених бізнес-правил.

Правила можуть застосовуватися до даних одного або кількох пристроїв і виконувати одну дію або кілька паралельних дій. Движок правил підтримує передачу повідомлень до різних кінцевих точок AWS, включно з AWS Lambda, Amazon Kinesis, Amazon S3, Amazon Machine Learning, Amazon DynamoDB, Amazon CloudWatch та Amazon Elasticsearch Service, з можливістю інтеграції з Kibana. Підключення до зовнішніх сервісів також можливе через AWS Lambda, Amazon Kinesis або Amazon Simple Notification Service (SNS).

Створювати правила можна як через Консоль управління, так і за допомогою SQL-подібного синтаксису. Правила можна налаштувати так, щоб

вони виконували різні дії залежно від змісту повідомлення. Наприклад, якщо показники температурного датчика перевищують певний поріг, правило може активувати передачу даних до AWS Lambda. Також можна враховувати дані від інших пристроїв у хмарі, наприклад виконати дію, якщо температура перевищує середнє значення п’яти інших датчиків на 15%.

У движку правил доступні десятки вбудованих функцій для обробки та трансформації даних, а за допомогою сервісу AWS Lambda їх можна розширювати практично без обмежень. Наприклад, при роботі з великим діапазоном значень можна обчислювати середнє значення вхідних даних. Крім того, правила дозволяють запускати виконання коду на Java, Node.js або Python у AWS Lambda, що надає надзвичайно гнучкі можливості для потужної обробки інформації, отриманої від пристроїв.

Рішення Google Cloud Platform для IoT

Хмарна платформа GCP надає низку інструментів, які можна ефективно використовувати для щоденної роботи з IoT-пристроями:

- Google Cloud Monitoring - забезпечує панелі для моніторингу та налаштування оповіщень для хмарних додатків. На Linux-пристроях можна встановити Cloud Monitoring agent, який базується на Stackdriver, а також користуватися API Monitoring Cloud для отримання необхідних метрик.

- Google Cloud Logging - дозволяє збирати та зберігати журнали подій, переглядати, шукати, фільтрувати та експортувати дані. Використання Cloud Logging значно економить час та ресурси у порівнянні з розробкою індивідуального рішення.

- Google Cloud Audit Logs - охоплює адміністративну діяльність і доступ до даних у Cloud Platform, зберігаючи їх у незмінних журналах, що можуть застосовуватися для аудиту.

Cloud Pub/Sub

У Google Cloud Pub/Sub можна створювати теми для потоків або каналів, що дозволяє різним компонентам застосунків підписуватися на потрібні потоки даних без необхідності налаштовувати окремі канали для кожного пристрою. Платформа легко інтегрується з іншими сервісами Google Cloud, забезпечуючи підключення прийому даних, шлюзів та систем зберігання інформації.

Cloud Pub/Sub також може виконувати функцію буфера та балансувальника швидкості для вхідних потоків даних, що особливо корисно, коли багато пристроїв одночасно передають телеметрію або реагують на події в реальному світі. Це допомагає ізолювати піки навантаження та запобігти перевантаженню додатків моніторингу.

Крім стандартного API та HTTPS REST, Cloud Pub/Sub підтримує gRPC - відкритий високопродуктивний протокол, який забезпечує більшу пропускну здатність для бінарних повідомлень. Наприклад, тестування з використанням Java-клієнта показало, що при публікації повідомлень розміром 50 КБ за допомогою 9 gRPC-каналів досягається максимальна пропускна здатність одного комп'ютера (рисунок 2.3).

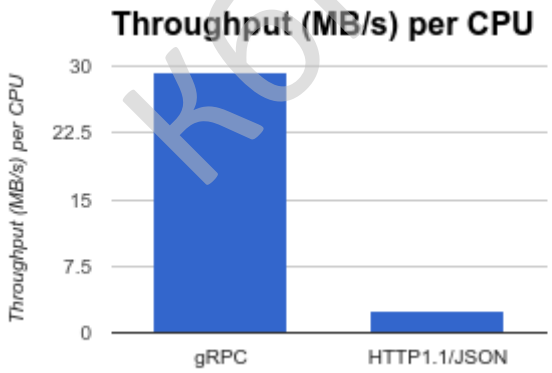

Рисунок. 2.3 - Порівняльна характеристика gRPC та HTTP

Зберігання даних

Дані з фізичного світу надходять у різних форматах та обсягах. Хмарні платформи пропонують різноманітні рішення для зберігання як неструктурованих даних, наприклад, зображень або відеопотоків, так і структурованих даних, таких як показники роботи пристроїв або транзакції.

Стан пристрою зазвичай моделюється як набір пар «ключ-значення». Деякі пристрої підключені безпосередньо до апаратного забезпечення, інші - зберігають стан на рівні програмного забезпечення. Іноді важливо, щоб інші сервіси, наприклад мобільний додаток або веб-сайт, могли читати або змінювати останній стан пристрою.

Оскільки IoT-пристрої можуть працювати у низькому енергоспоживанні або перебувати у нестабільних мережах, корисно синхронізувати певні стани з хмарою. Це дозволяє забезпечити доступ до актуальної інформації навіть тоді, коли самі пристрої тимчасово недоступні.

Smart Home Cloud API

Smart Home Cloud API надає інструменти для керування та моніторингу пристроїв Samsung Smart Home. За допомогою Smart Home Service Control додатки можуть підключатися до різних пристроїв і надавати користувачам додаткові послуги. Цей сервіс працює через інтеграцію «хмара-хмара», забезпечуючи взаємодію між клієнтською хмарою та Smart Home Cloud.

Samsung пропонує кілька REST API для партнерів, що дозволяє інтегрувати сторонні системи з Smart Home Cloud. Дані REST API передаються у стандартному форматі JSON. Тому розробники-партнери повинні ознайомитися зі структурою документації JSON, відомою як Smart Home Data Model.

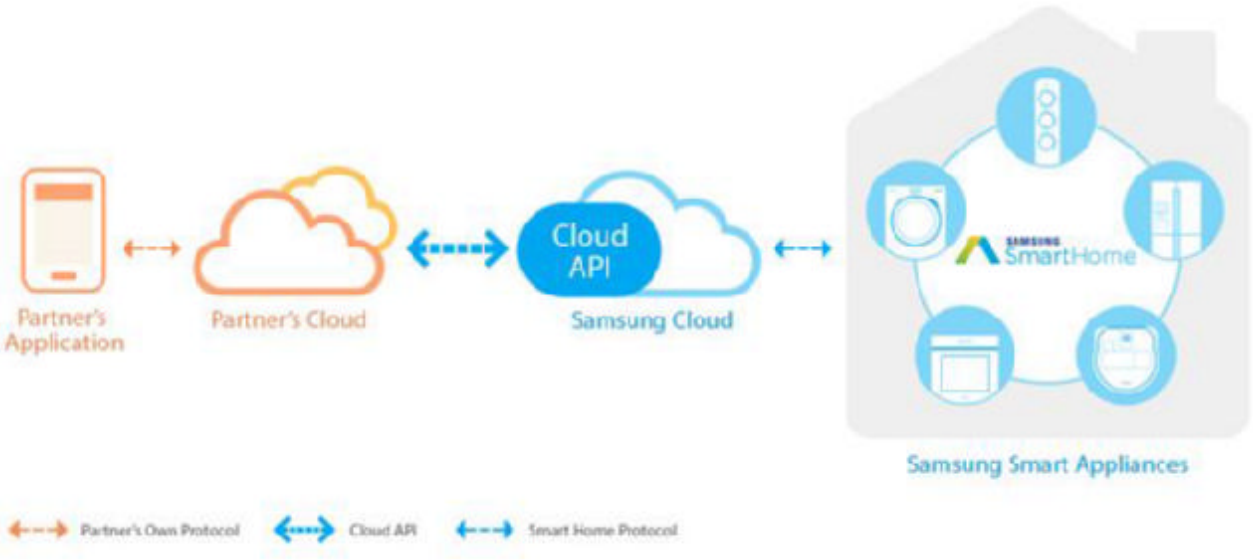

Рисунок 2.4 - Специфікації Smart Home Cloud API

Модель інтеграції

Smart Home Data Model - це структура даних у форматі JSON, яка описує пристрої Samsung Smart Home, такі як холодильники, пральні машини, кондиціонери, очищувачі повітря, роботи-пилососи, сушарки та печі.

Процес інтеграції користувачів (партнерів) зі Smart Home Cloud включає такі етапи:

- Authentication (Аутентифікація). Для підключення до пристроїв Samsung система партнера повинна пройти авторизацію через обліковий запис Samsung. Підключення можливе лише при наявності дійсного токена, отриманого через OAuth.

- Discovery (Виявлення). Партнер отримує список усіх розумних домашніх пристроїв, зареєстрованих для конкретного користувача, разом із докладною інформацією: тип, назва, модель, версія та доступні ресурси.

- Sensing (Моніторинг стану). Партнер може отримувати інформацію про стан конкретного пристрою. Sensing API забезпечує доступ до найнижчого рівня статусу ресурсів пристрою.

- Subscription (Підписка). Партнер реєструє підписку на повідомлення про зміни стану пристрою та отримує дані в реальному часі.

- Notification (Повідомлення). Якщо Smart Home Cloud виявляє зміну стану пристрою, він надсилає відповідні повідомлення та оновлений статус партнеру.

- Control (Керування). Партнер може управляти пристроєм через API керування Smart Home Cloud.

- Unsubscription (Відмова від підписки). Партнер може відмовитися від отримання повідомлень про зміни стану пристрою. Після цього дані повідомлень від пристрою більше не надходять.

Не розглянуті хмарні платформи

Під час аналізу хмарних платформ для керування розумним будинком були виявлені деякі сервіси, які не включені до дослідження у дипломній роботі. Причиною цього стало обмежена або відсутня інформація на офіційних сайтах, незрозумілі типи інтеграції, а також недостатня або неповна документація для розробників.

Таблиця 2.1 - Не розлянуті хмарні платформи

Назва	Посилання	Причина
EasyIoT	http://iot-playground.com/	Недостатня документація, Бета-версія
Onion.io	https://onion.io/cloud/	Незрозумілий тип інтеграції, суб'єктивна негативна оцінка
Salesforce	http://www.salesforce.com/iot-cloud/	Незрозумілий тип інтеграції, недостатня документація, суб'єктивна негативна оцінка
Oracle	https://cloud.oracle.com/iot	Незрозумілий тип інтеграції, недостатня документація, суб'єктивна негативна оцінка

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Порівняльна характеристика

Порівняння розглянутих хмарних платформ буде здійснюватися за такими критеріями:

- безпека передачі даних;
- можливості масштабування та розширення;
- різноманіття підтримуваних пристроїв;
- засоби моніторингу та аналізу даних.

Ці критерії наведені у довільному порядку. Важливість кожного з них (вага критерію) буде визначена пізніше.

Для зручності у подальшому тексті три хмарні платформи будуть позначатися скорочено за назвами компаній, що надають послуги: Amazon IoT, Google IoT та Samsung IoT.

Безпека передачі даних

Основні методи забезпечення захисту інформації, що передається від датчиків та клієнтських програм управління чи адміністрування, включають:

- Використання SSL-з'єднань для шифрування переданих даних;
- Аутентифікацію сеансів через логін та пароль;
- Присвоєння унікального ідентифікатора кожному пристрою (див. таблицю 2.2).

Таблиця 2.2 - Оцінка безпеки передачі даних

Назва	SSL	Аутентифікація	Унікальний ідентифікатор кожного пристрою	Додаткові засоби
Amazon IoT	+	+	+	-
Google IoT	+	+	+/-	-
Samsung IoT	+	+	+	Samsung OAuth

Google IoT не призначає унікальний ідентифікатор окремому датчику. Натомість, ідентифікатор присвоюється групі пристроїв, об'єднаних одним маршрутизатором.

У випадку Samsung IoT використовується єдина система аутентифікації Samsung OAuth, яка охоплює всі пристрої Samsung під одним аккаунтом.

Детальніші дані про окремі пристрої можна отримати лише маючи доступ до цього аккаунту та фізичний пристрій, тому додаткової інформації отримати не вдалося.

Можливості розширення

Amazon IoT

Стартовий безкоштовний пакет Amazon IoT включає всі необхідні компоненти: базу даних, обчислювальні ресурси та канали зв'язку. Цінова модель «оплата за фактом використання» надає можливість оптимізувати витрати, планувати масштабування або підвищення ресурсів залежно від сезонних потреб, а також коригувати бюджет відповідно до вимог бізнесу. Крім того, користувач може окремо розширювати обчислювальні потужності, обсяг пам'яті та пропускну здатність каналу передачі даних у відповідності до власних потреб.

Google IoT

Безкоштовний стартовий пакет Google IoT надає доступ до базових компонентів платформи: бази даних, обчислювальних ресурсів та каналів зв'язку. Додатково користувач може підключати різні API для виконання специфічних завдань, таких як:

- обробка великих даних (Big Data);
- ведення логів;
- застосування алгоритмів машинного навчання;
- управління мережею (DNS, балансування навантаження).

Варто зазначити, що ці додаткові сервіси є дорогими і потребують спеціальних знань у відповідних галузях. З одного боку, звичайному користувачу знадобиться допомога фахівця для налаштування цих сервісів; з іншого боку, їх використання може значно скоротити час розробки та впровадження системи. Оплата за додаткові сервіси здійснюється на момент їх підключення.

Samsung IoT

У Samsung IoT пакет послуг є платним з моменту підключення. Всі сервіси надаються відразу і автоматично оновлюються при виході нових версій платформи або оновлень. Безкоштовної версії платформи для користувачів не передбачено.

Таблиця 2.3 - Оцінка можливості розширення

Назва	Безкоштовний стартовий пакет послуг	Оплата по факту використання	Можливість підключення готових додаткових API
Amazon IoT	+	+	+
Google IoT	+	-	+
Samsung IoT	-	-	-

Різноманітність пристроїв, які можуть бути підключені
Amazon IoT

Платформа надає SDK для Embedded C, JavaScript та спеціалізовану підтримку для Arduino Yún. Багато мікроконтролерів Arduino сумісні з прошивкою на Embedded C, що дозволяє інтегрувати широкий спектр пристроїв. Користувачі можуть самостійно прошивати та налаштовувати пристрої відповідно до власних потреб, що забезпечує гнучкість і масштабованість системи розумного будинку.

Google IoT

Технологія Google Cloud Pub/Sub дозволяє підключати датчики та мікроконтролери за допомогою протоколу gRPC, який сумісний з великою кількістю мікроконтролерів, включно з Arduino NodeMCU. Це забезпечує гнучку інтеграцію різних пристроїв у систему та дозволяє організовувати надійний обмін даними між пристроями та хмарною платформою.

Samsung IoT

Платформа Платформа Samsung IoT підтримує підключення виключно до пристроїв власного виробництва, що значно обмежує можливості для бюджетного тестування та експериментальної інтеграції з іншими пристроями.

Моніторинг та аналіз даних

Amazon IoT

Amazon IoT дозволяє створювати «тіні» пристроїв у хмарі - віртуальні моделі, куди записуються дані датчиків та надсилаються команди. Платформа підтримує генерацію приватних і публічних ключів для безпечної передачі інформації, а також створення «ролей» - тригерів на певні події, такі як критичні показники датчиків, сигнали тривоги чи команди з клієнтської частини. Через відкритий API та SDK можна отримувати дані на клієнтські пристрої (телефон, сайт, десктоп) за допомогою протоколу MQTT, що забезпечує зручний моніторинг і управління пристроями.

Google IoT

Google IoT надає широкий набір інструментів для аналізу даних, включно з побудовою графіків, відстеженням показників у часі та обробкою великих обсягів даних (Big Data). Платформа також інтегрує нейронну мережу, яку можна використовувати для автоматизованого управління пристроями за допомогою штучного інтелекту. Однак ця функція наразі перебуває у бета-версії та ефективна лише при наявності великого обсягу навчальних даних.

Samsung IoT

Samsung IoT вирізняється наявністю спеціалізованого програмного забезпечення на базі Android, яке можна легко встановити на пристрої Samsung та отримувати дані з хмари без написання додаткового коду. Платформа підтримує систему нотифікацій із налаштуванням пріоритетів, можливістю підписки та відписки від певних подій, а також відображення показників за великий проміжок часу. Проте інші інструменти моніторингу залишаються недоступними для дослідження через обмежену інформацію для неліцензійованих користувачів.

У ході дослідження було проаналізовано три платформи хмарного керування розумним будинком: AWS IoT (Amazon), Google Cloud Platform IoT Solutions та Smart Home Cloud API (Samsung Smart Home) за чотирма критеріями:

- Безпека передачі даних.
- Можливості розширення.
- Різноманітність підключених пристроїв.
- Моніторинг та аналіз даних.

У аспекті безпеки передачі даних всі три платформи виявилися надійними, використовуючи сучасні технології шифрування та аутентифікації користувачів і пристроїв.

Що стосується можливостей розширення, Google IoT виділяється завдяки широкому набору додаткових інструментів для аналізу та моніторингу даних. Однак, оскільки більшість технологій цієї платформи є платними, а для Amazon IoT доступні безкоштовні альтернативи, вибір лідера залежить від навичок розробника, наявного часу та бюджету. Samsung IoT пропонує готовий набір інструментів, який складно кастомізувати під специфічні потреби.

За критерієм різноманітності підключених пристроїв Samsung IoT значно поступається, оскільки підтримує лише пристрої власного виробництва. Amazon

IoT та Google IoT забезпечують більшу свободу вибору датчиків, протоколів і контролерів.

Що стосується моніторингу та аналізу даних, усі платформи надають необхідні інструменти для адміністрування та налаштування пристроїв і датчиків. Google IoT вирізняється потужними засобами візуалізації даних, які при бажанні можна інтегрувати до Amazon IoT. Samsung IoT лише описує наявні інструменти, без можливості протестувати їх на тестових даних, що ускладнює оцінку платформи при виборі рішення для проекту.

Таблиця 2.4 - Порівняння трьох платформ за чотирма критеріями

Критерій	**Amazon IoT**	**Google IoT**	**Samsung IoT**
Безпека передачі даних	Високий рівень: SSL, аутентифікація сертифікатами, приватні та публічні ключі	Високий рівень: SSL, аутентифікація користувачів і груп пристроїв	Високий рівень: Samsung OAuth, єдиний акаунт для всіх пристроїв
Можливості розширення	SDK для C, JavaScript, Arduino; відкритий API; «тіні» пристроїв; ролі для тригерів	Великий набір інструментів для аналізу та моніторингу; інтеграція з Big Data, ML; гнучкі API	Готовий набір інструментів; важко кастомізувати; автоматичне оновлення
Різноманітність пристроїв	Підтримка великої кількості датчиків та контролерів; можливість кастомної прошивки	Підтримка різних мікроконтролерів через gRPC (Arduino, NodeMCU)	Працює тільки з пристроями Samsung; обмежена можливість бюджетного тестування
Моніторинг та аналіз даних	Вбудовані інструменти моніторингу; дані можна виводити на клієнтські пристрої; інтеграція через MQTT	Потужні інструменти візуалізації та аналізу; AI/ML для автоматизації; інтеграція з Amazon IoT можлива	Система нотифікацій та відображення даних; інші інструменти недоступні для тестування

Висновок

Було проведено аналіз найпопулярніших та потенційно надійних хмарних платформ для керування розумним будинком. Кожна з них має свої сильні та слабкі сторони, тому було прийнято рішення розробити власну систему керування розумним будинком з урахуванням найкращих практик, запозичених із досліджених хмарних сервісів.

Для дослідження та програмної реалізації хмарного керування розумним будинком було обрано такі засоби та мову програмування:

Розроблювана система хмарного керування «Розумний дім» складатиметься з наступних компонентів:

- пристрої - безпосередньо всі електронні елементи, контроль над якими необхідно автоматизувати;
- датчики - пристрої для збору інформації та керування системою розумного будинку; вони виконують роль базових одиниць у системі;
- мікроконтролери - апаратні системи, які об’єднують датчики у групи та виконують роль центрального процесора управління, передаючи інформацію з датчиків на сервер та керуючи кінцевими вузлами;
- сервер - комп’ютер, який забезпечує інтерфейс між користувачем та системою розумного будинку; відповідає за надійність та функціональність системи;
- канали передачі даних - фізичні та логічні канали, через які передаються дані з урахуванням вимог до безпеки та швидкості обміну;
- хмара - зовнішня служба, яка виконує роль бази даних для статистики та іншої службової інформації;
- мобільні пристрої - пристрої, через які користувач може управляти системою розумного будинку через сервер;
- мова програмування - c, що дозволяє створювати програми з високим рівнем зручності та ефективності для розробника;

- аппаратна база - система буде побудована на мікроконтролері NodeMCU v3 LoLin;

Таким чином, запропонована архітектура забезпечує надійність, масштабованість та гнучкість системи керування розумним будинком, поєднуючи локальні пристрої, хмарні сервіси та мобільні інтерфейси.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на магістерську роботу, об’єктом розробки є програмне забезпечення для хмарного керування розумним будинком.

У процесі розробки магістерської роботи передбачається виконати наступний обсяг робіт:

а) Аналіз існуючих систем-аналогів:

- дослідити наявні платформи та системи для керування розумним будинком;
- виявити їхні позитивні та негативні характеристики;
- використати результати аналізу для оптимізації подальшої розробки.

б) Вибір методики побудови системи:

- обґрунтувати обрану методику створення системи контролю роботи технологічного обладнання в автоматизованому режимі;
- розробити функціональну та структурну схеми системи.

в) Розробка програмного забезпечення:

-створити програмне забезпечення, яке реалізує завдання, визначені технічним завданням;

-побудувати блок-схеми алгоритмів програм та підпрограм.

г) Організація інтерфейсу користувача:

-розробити інтерфейс для формування та виводу повідомлень на екран ЕОМ про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання.

д) Розробка рекомендацій з впровадження:

-підготувати методичні та організаційні рекомендації щодо впровадження системи в промислову експлуатацію та забезпечення її подальшої успішної роботи.

е) Розрахунок економічної ефективності:

-виконати розрахунки для визначення економічної ефективності розробленої системи.

ж) Заходи безпеки:

-розробити заходи з охорони праці під час впровадження та експлуатації системи;

-розробити заходи цивільного захисту при роботі з системою.

з) Формування висновків:

-узагальнити результати виконаних робіт;

-підвести підсумки щодо досягнення поставлених завдань та отриманих результатів.

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Виходячи з теми магістерської роботи, необхідно розробити програмне забезпечення системи хмарного керування «Розумний дім», а також створити автоматизовану систему управління із встановленням відповідних датчиків та допоміжних підсистем.

Устаткування для реалізації системи «Розумний дім»

Перед початком розробки слід окреслити базовий набір обладнання, необхідний для перетворення звичайного житлового приміщення - квартири, приватного будинку чи дачі - на систему «розумного дому». До основних компонентів належать:

– датчики, призначені для вимірювання параметрів навколишнього середовища (температури, освітленості, вологості, наявності протікання тощо);

– виконавчі пристрої, які здійснюють безпосередній вплив на об’єкти управління (вмикання/вимикання освітлення, керування розетками, клапанами, електроприладами);

– контролер, який обробляє інформацію від датчиків, виконує логіку роботи системи та передає команди виконавчим елементам.

На наступній схемі зображено приклад розміщення елементів у типовому «розумному домі»: датчики протікання води (1) у ванній кімнаті, датчики температури (2) та освітленості (3) у спальні, «розумна» розетка (4) на кухні, а також камера відеоспостереження (5) у передпокої.

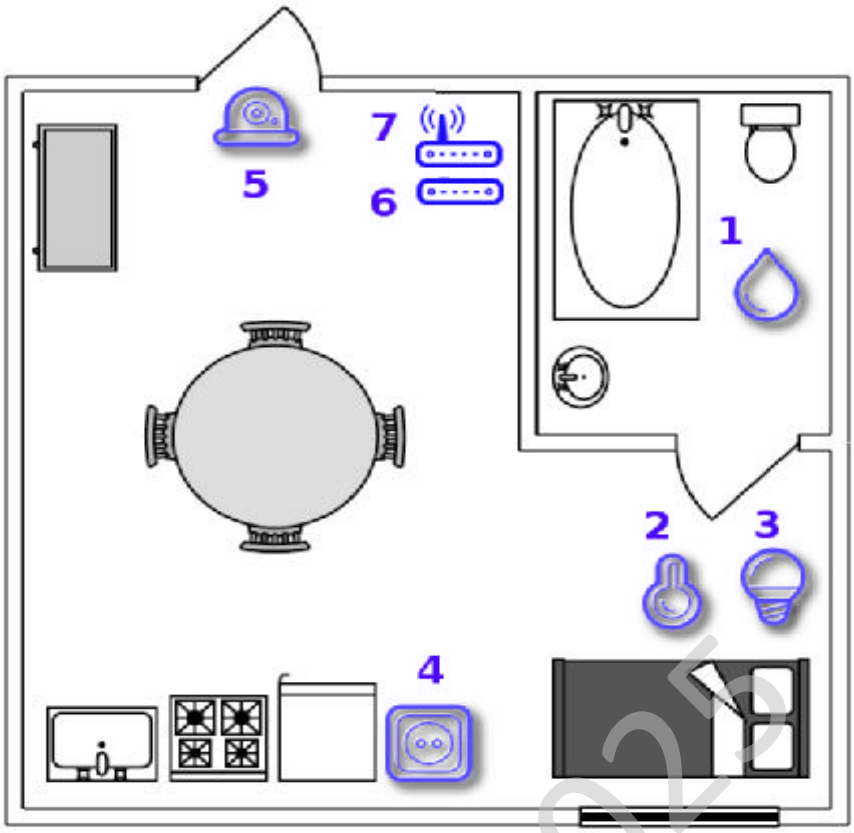

Рисунок 3.1 - Обладнання «розумного дому»

Датчики

Сьогодні значного поширення набули бездротові датчики, що працюють за протоколами RF433, Z-Wave, ZigBee, Bluetooth та WiFi**.** Їхня популярність пояснюється простотою монтажу й експлуатації, а також доступною вартістю та високою надійністю. Виробники активно орієнтуються на масовий ринок, роблячи такі пристрої максимально простими у встановленні та використанні для пересічного користувача.

Зазвичай датчики та виконавчі пристрої під’єднуються до контролера розумного дому (6) - спеціалізованого мікрокомп’ютера, що об’єднує всі елементи системи в єдину мережу та забезпечує їх узгоджену роботу. У деяких рішеннях один пристрій може одночасно виконувати функції датчика, виконавчого елемента та контролера. Наприклад:

– «розумна» розетка може самостійно вмикатися/вимикатися за заданим розкладом;

– хмарна камера відеоспостереження здатна записувати відео після спрацювання детектора руху.

У найпростіших конфігураціях систему можна побудувати й без окремого контролера, проте для створення повноцінної, гнучкої та сценарно орієнтованої системи контролер є необхідним компонентом.

Для підключення контролера до Інтернету зазвичай застосовують звичайний Wi-Fi-роутер (7), який є в більшості домогосподарств. Це забезпечує ще одну важливу перевагу: у разі втрати доступу до мережі Інтернет система «розумного дому» продовжуватиме працювати у штатному режимі, оскільки її логіка зберігається локально в контролері, а не в хмарній інфраструктурі.

Контролер розумного будинку

Процес складання контролера є досить простим. Основним елементом виступає мікрокомп’ютер (1), який розміщується в пластиковому корпусі (2). У призначені слоти встановлюється карта пам’яті microSD об’ємом 8 ГБ із заздалегідь підготовленим програмним забезпеченням (3), а також USB-модуль Z-Wave (4), що забезпечує взаємодію з бездротовими пристроями.

Для живлення контролера використовується адаптер 5В, 2.1А (5) разом із кабелем USB-microUSB (6).

Кожен контролер отримує унікальний ідентифікаційний номер, який автоматично записується у конфігураційний файл під час першого запуску. Цей ідентифікатор необхідний для подальшої роботи пристрою із сервісами хмарної платформи розумного будинку.

Програмне забезпечення контролера

Програмне забезпечення контролера розумного будинку, створене автором на основі операційної системи Linux, являє собою комплекс взаємопов’язаних програмних модулів. Його структура включає такі основні складові:

– Серверна підсистема, що забезпечує обмін даними між контролером, периферійним обладнанням розумного будинку та хмарною інфраструктурою. Вона відповідає за приймання, обробку та маршрутизацію команд, а також за підтримку стійкого каналу зв’язку з датчиками та виконавчими пристроями.

– Графічний інтерфейс користувача, призначений для конфігурування системи, зміни робочих параметрів і керування режимами роботи контролера. Завдяки інтуїтивному інтерфейсу користувач може виконувати налаштування без залучення додаткових спеціалізованих засобів.

– База даних, яка використовується для зберігання конфігураційних параметрів, інформації про підключені пристрої та службових даних. Її структура оптимізована для швидкого доступу та надійного зберігання у випадку некоректного завершення роботи системи.

У сукупності ці компоненти забезпечують стабільну, безпечну та гнучку роботу контролера розумного будинку, дозволяючи реалізувати широкий спектр сценаріїв автоматизації та інтеграції з хмарними сервісами.

База даних контролера розумного будинку

База даних контролера розумного будинку реалізована з використанням вбудованої СУБД PostgreSQL та зберігається у вигляді окремого файлу на SD-карті разом із системним програмним забезпеченням. Вона виконує роль основного сховища конфігураційних даних контролера, включаючи інформацію про під’єднане обладнання, його поточний стан та набір логічних продукційних правил, що визначають роботу системи автоматизації.

Крім того, у базі даних зберігаються відомості, які потребують індексації, наприклад, перелік файлів локального відеоархіву або службові параметри пристроїв. Завдяки цьому забезпечується швидкий доступ до необхідних даних і ефективна робота програмних модулів контролера.

Однією з ключових переваг такої організації є збереження даних після перезавантаження або відновлення живлення. Це дозволяє контролеру

автоматично повернутися до стабільного робочого стану після збоїв та гарантує безперервність роботи системи розумного будинку.

Графічний інтерфейс контролера

Графічний інтерфейс контролера розумного будинку створений мовою PHP 7 з використанням легковагого мікрофреймворка Slim, що забезпечує швидку розробку та високу гнучкість у налаштуванні веб-додатка. За обробку запитів і роботу інтерфейсу відповідає веб-сервер lighttpd, який широко застосовується у вбудованих системах завдяки своїй високій продуктивності, малій вимогливості до ресурсів та стабільності в умовах обмеженого апаратного забезпечення.

Хмарний сервіс розумного будинку

Хмарний сервіс розумного будинку забезпечує зручний, гнучкий та економічний спосіб зберігання й доступу до даних, що надходять від підключених пристроїв. Завдяки використанню хмари користувачу не потрібно турбуватися про надійність та довготривале збереження інформації.

Потужності багатопроцесорного медіасервера, оснащеного дисковим кошиком з RAID-масивом, об’єднаним з кількох дисків об’ємом 10-12 ТБ, значно перевищують ємність внутрішніх SD- або Flash-карт у контролері розумного будинку. Крім того, карти пам’яті мають обмежений ресурс циклів перезапису та є менш надійними, що підвищує ризик втрати даних.

Глибина зберігання даних у хмарі залежить від тарифного плану користувача та легко змінюється через особистий кабінет. Для доступу до інформації не потрібно виконувати «проброс портів» на маршрутизаторі, оскільки пристрої розумного будинку зазвичай приховані за NAT і недоступні ззовні.

У особистому кабінеті, доступному з комп’ютера або мобільного пристрою, користувач може:

- налаштовувати конфігурацію системи;
- задавати логіку роботи розумного будинку;

– переглядати дані та події;

– формувати статистику за будь-які періоди.

Хмарний сервіс дозволяє не лише зберігати дані, а й виконувати їх обробку. Наприклад, можна легко обчислити середню температуру в приміщенні за тиждень, використовуючи вимірювання мультисенсора.

3.2 Розробка структурної схеми

Структурна схема системи - це сукупність основних компонентів та елементів, а також зв’язків між ними. Вона призначена для наочного відображення складових частин розроблюваної системи, її ключових блоків, модулів і взаємодій.

На рисунку 3.2 представлено структурну схему створеної системи, на якій показано її будову та логічні зв’язки між основними елементами.

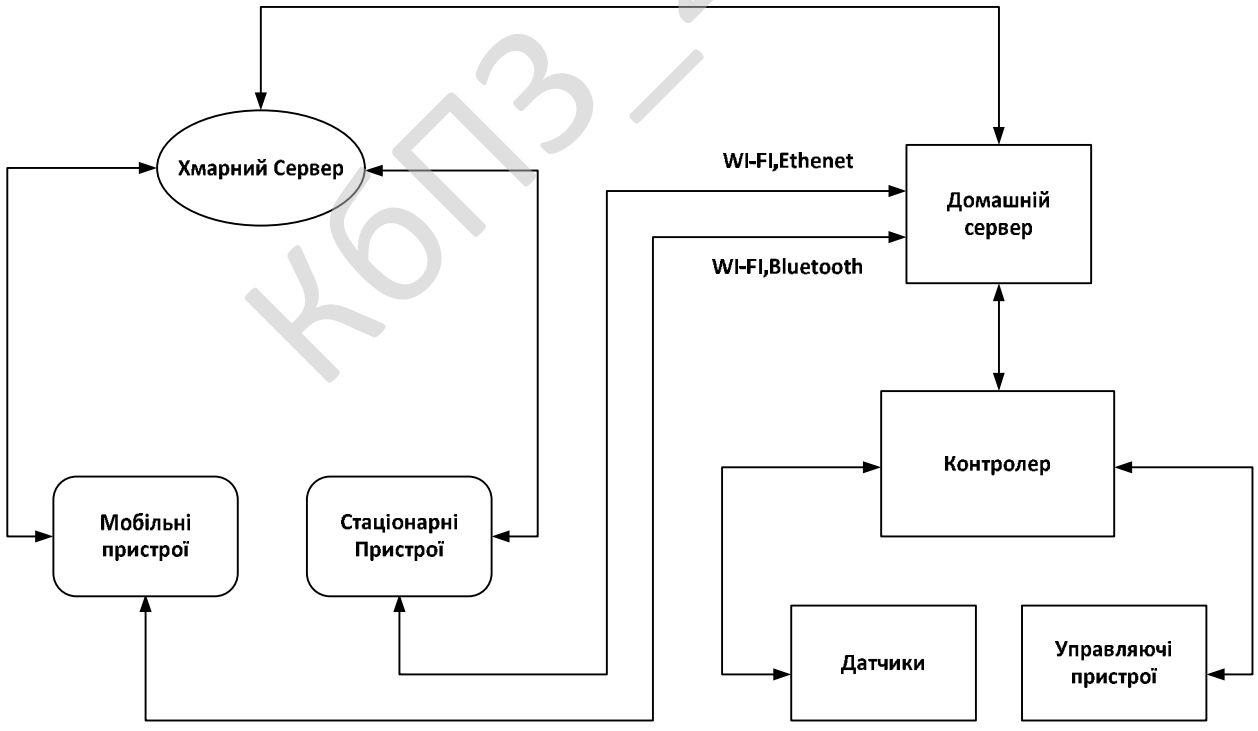

Рисунок 3.2 - Структурна схема системи «Розумний дім»

Пристрій, побудований за наведеною структурною схемою, живиться від електромережі змінного струму з параметрами 220 В, 50 Гц. Змінна напруга на вході пристрою через бустерний конвертер перетворюється на стабілізовану постійну напругу 3,5 В, яка забезпечує живлення мікроконтролера.

Мікроконтролер формує сигнали керування для виконавчого елемента, що відповідає за комутацію електроприладу. У самому електричному приладі встановлені датчики напруги та струму, вихідні сигнали яких у вигляді аналогових величин (0-3,5 В) подаються на входи АЦП мікроконтролера для подальшої обробки.

3.3 Розробка функціональної схеми

Функціональна схема розробленої системи представлена на рисунку 3.3. Як видно з рисунку, система складається з таких основних блоків:

– Хмара - зовнішній сервіс, який виконує функції бази даних для зберігання статистики та іншої службової інформації;

– WEB-сервер;

– Контролер з підключеним обладнанням;

– Модуль безпеки та шифрування;

– Web-інтерфейс;

– Мобільний застосунок;

– Виконавчі механізми.

Веб-сервер

Веб-сервер зберігає дані, отримані від сервера Bluetooth LE, та на їх основі за допомогою алгоритму оптимізації приймає рішення щодо підключення або відключення побутових пристроїв. Це дозволяє зменшувати навантаження на електромережу та запобігати аварійним ситуаціям. Крім того, сервер надає API для ручного керування системою через графічний інтерфейс у браузері.

У якості основи веб-сервера використано JavaScript-фреймворк ExpressJS, який є стандартом для більшості сучасних NodeJS-додатків. Його архітектура наслідує принципи фреймворка Ruby Sinatra, головними перевагами якого є простота та висока швидкість обробки запитів.

Серверний процес

Серверний процес є ключовим компонентом системи, який забезпечує автоматизацію всіх основних інформаційних процесів розумного будинку: прийом та обробку даних від сенсорів, формування керуючих сигналів відповідно до закладеної логіки. Основна функція серверного процесу - взаємодія з обладнанням, виконання продукційних логічних правил, а також обробка команд, отриманих від графічного інтерфейсу та хмарного сервісу.

У розглянутому контролері серверний процес реалізований як багатопоточний додаток на мові C++ і запускається як окремий системний сервіс.

Основні блоки серверного процесу:

- Диспетчер повідомлень;
- Сервер IP-камери;
- Сервер пристроїв Z-Wave;
- Сервер продукційних логічних правил;
- База даних конфігурації контролера та логічних правил;
- RESTful API сервер для взаємодії з графічним інтерфейсом;
- MQTT клієнт для обміну даними з хмарою.

Кожен блок реалізований як окремий потік, а обмін інформацією між потоками здійснюється у вигляді JSON-повідомлень (або структур даних, що представляють цей формат у пам’яті процесу).

Диспетчер повідомлень є центральним компонентом серверного процесу - він маршрутизує JSON-повідомлення між усіма блоками. Типи полів JSON-повідомлень та допустимі значення наведені у відповідній таблиці.

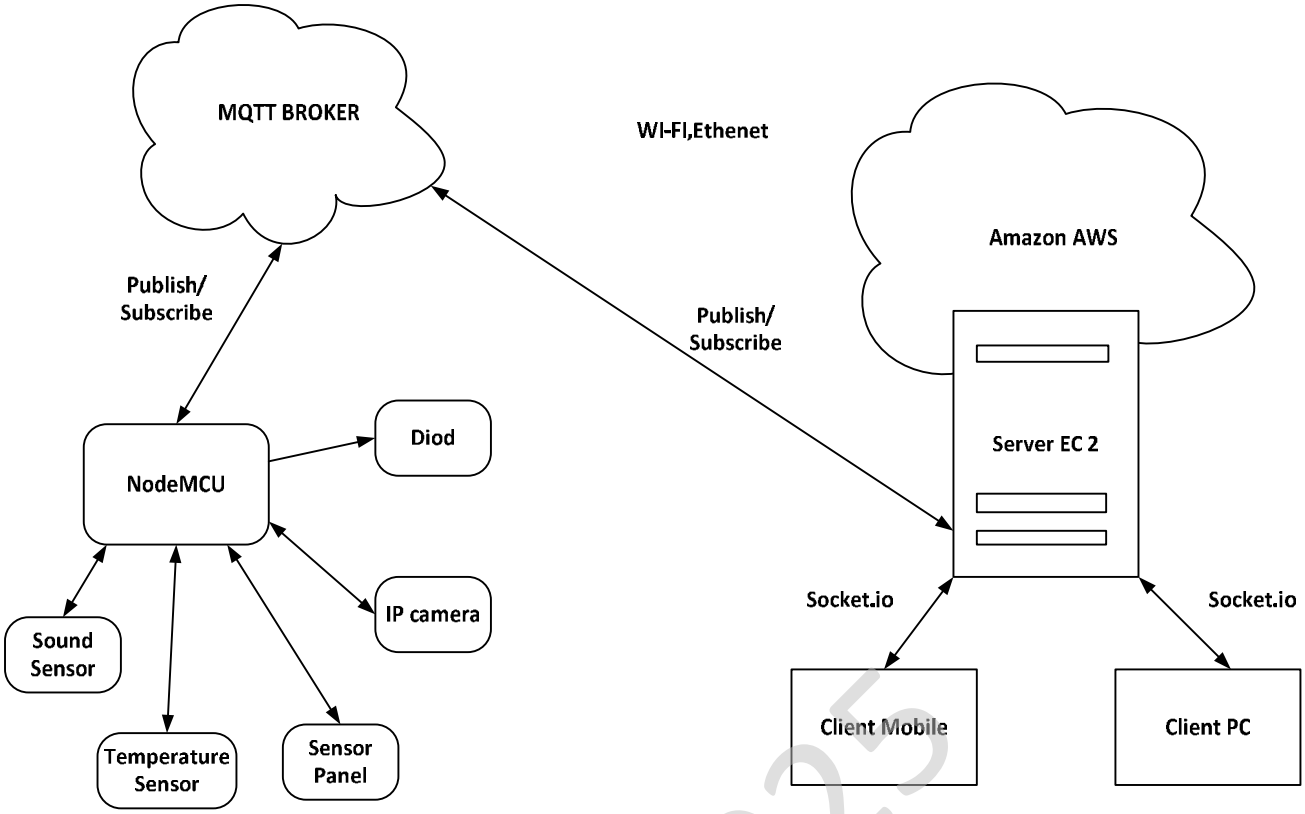

Рисунок 3.3 - Фукціональна схема системи «Розумний дім»

3.4 Розробка діаграми процесів

Діаграма процесів розробленої системи наведена на рисунку 3.4. Вона демонструє послідовність і механізм взаємодії компонентів у системі. Для моделювання використовується підхід «потоків даних» (Data Flow Diagram), який надає графічне представлення процесів обробки інформації в системі.

Діаграма взаємодії процесів застосовується для візуалізації обробки даних на рівні структурного проектування. Розробник зазвичай починає з побудови діаграми рівня контексту, що відображає загальну взаємодію системи з зовнішніми користувачами та підсистемами. Надалі ця діаграма деталізується шляхом уточнення окремих процесів і потоків даних, що дозволяє створити повне уявлення про систему.

На основі існуючих методів була реалізована система керування розумним будинком із підсистемою безпеки, яка гарантує захищену передачу керуючих сигналів від смартфона до мікроконтролера через мережу Internet.

Таким чином, після ознайомлення з описом системи, її структурною та функціональною схемами, а також діаграмою взаємодії процесів, переходять до побудови блок-схем основної програми та підпрограм, які безпосередньо реалізують функціонал системи.

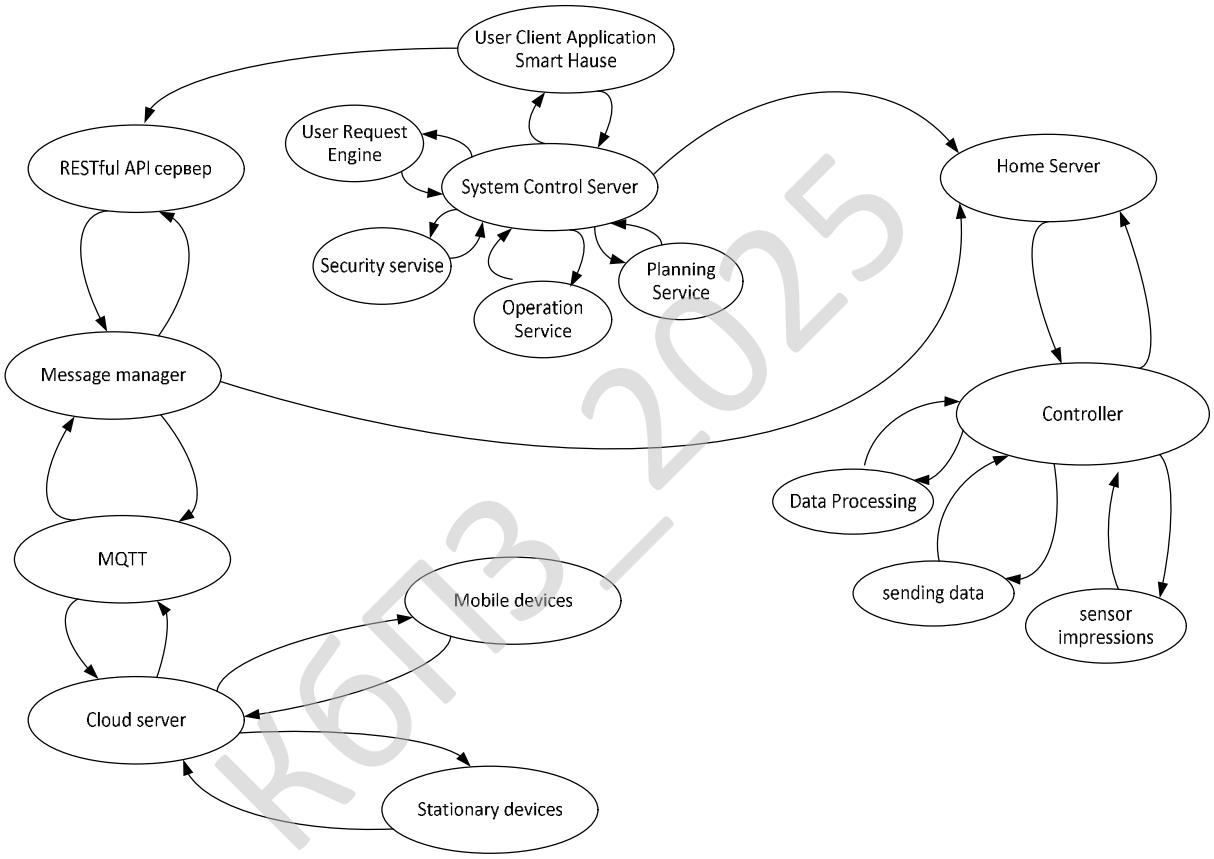

Рисунок 3.4 - Діаграма процесів системи «Розумний дім»

4 РЕАЛІЗАЦІЯ РОБОТИ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

Було проведено аналіз найпоширеніших і, ймовірно, найнадійніших хмарних платформ для керування розумним будинком. Кожна з них має свої переваги та недоліки, тому було прийнято рішення створити власну систему управління розумним будинком із врахуванням кращих практик, запозичених із результатів цього аналізу.

Схема підключення робочої моделі розумного будинку

Схема побудована на основі мікроконтролера NodeMCU v3 LoLin із вбудованим Wi-Fi модулем ESP8266 (рисунок 4.1). Незважаючи на невисоку вартість, цей контролер є дуже зручним інструментом для розробки цифрових схем розумного будинку.

Основні переваги NodeMCU:

– Простота програмування;

– Використання Lua 5.1.4 (без підтримки дебагу, лише ОС модулі);

– Асинхронна івент-орієнтована модель програмування;

– Понад 40 вбудованих модулів;

– Прошивка доступна як із підтримкою плаваючої точки, так і без неї (цілочисельна прошивка економить пам’ять);

– Постійне оновлення проекту та документації;

– Підтримка програмування на декількох мовах, включно з C.

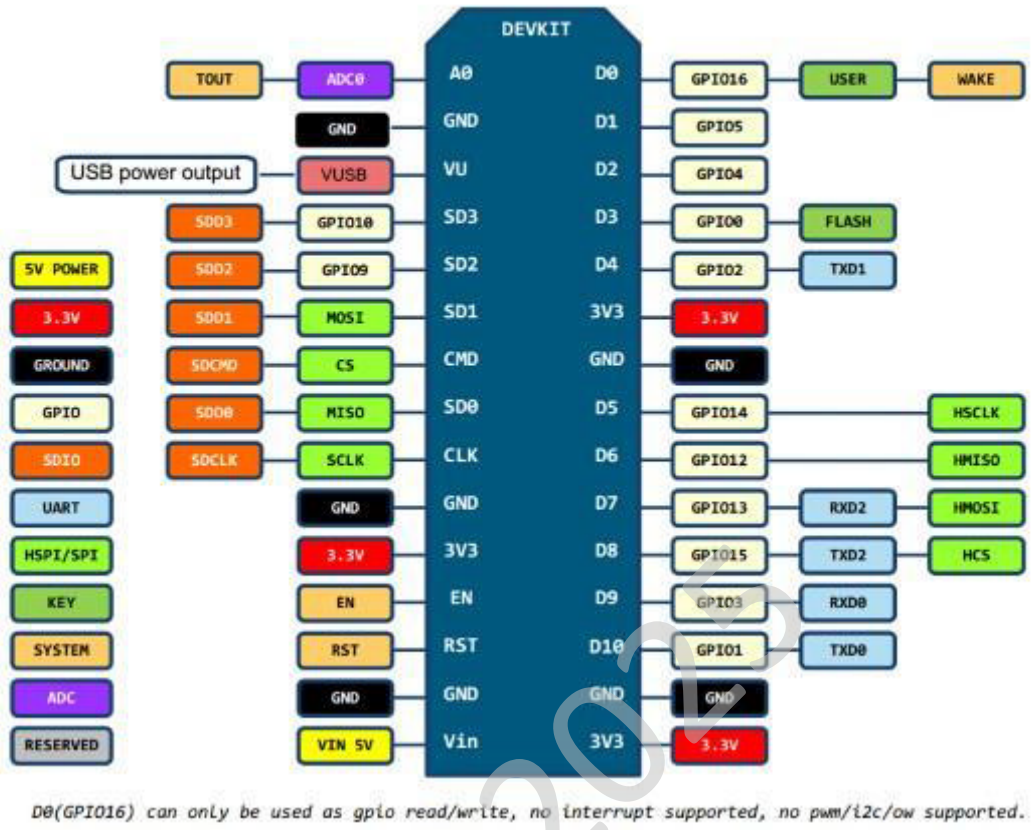

Рисунок 4.1 - Схема мікроконтролера NodeMCU v3 LoLin

Програмування контролера здійснюється в середовищі Arduino IDE мовою С. Мікроконтролер підключається до комп'ютера через кабель microUSB, який одночасно забезпечує його живлення та програмування.

До тестової робочої схеми підключені такі датчики та виконавчі елементи:

- Датчик сили/моменту Hex C227986;
- Датчик температури та вологості повітря Aosong AM2302;
- Датчик атмосферного тиску;
- Сенсорна панель із чотирма сенсорними кнопками;
- Світлодіодна лампочка.

Схема підключення показана на рисунку 4.2

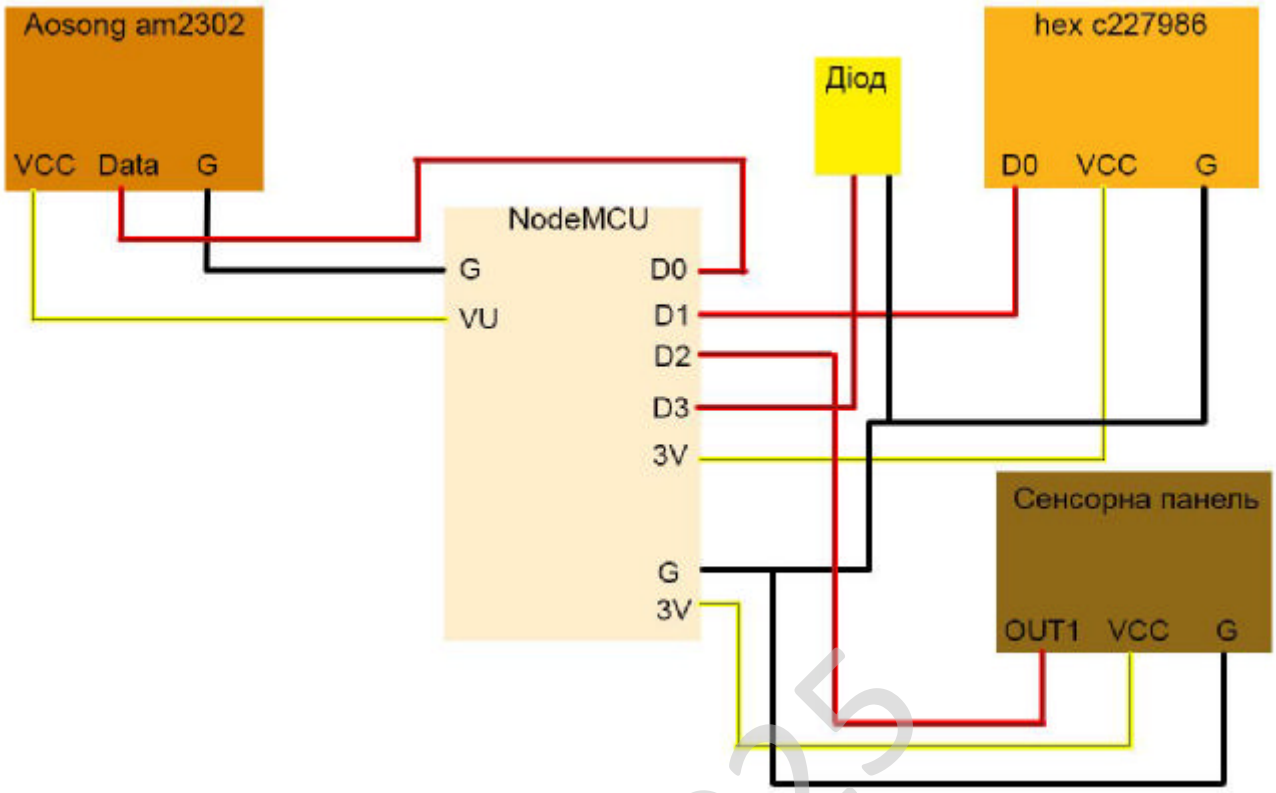

Рисунок 4.2 - Схема підключення робочої моделі

Для передачі даних з датчиків було обрано протокол MQTT. MQTT - це простий відкритий протокол, спеціально розроблений для застосування в IoT і обміну даними між пристроями. Мережа MQTT складається з MQTT-брокера, який виконує роль посередника між MQTT-агентами: видавцями та підписниками. Видавці публікують інформацію, яка призначена для підписників.

На рисунку 4.3 наведена схема роботи MQTT.

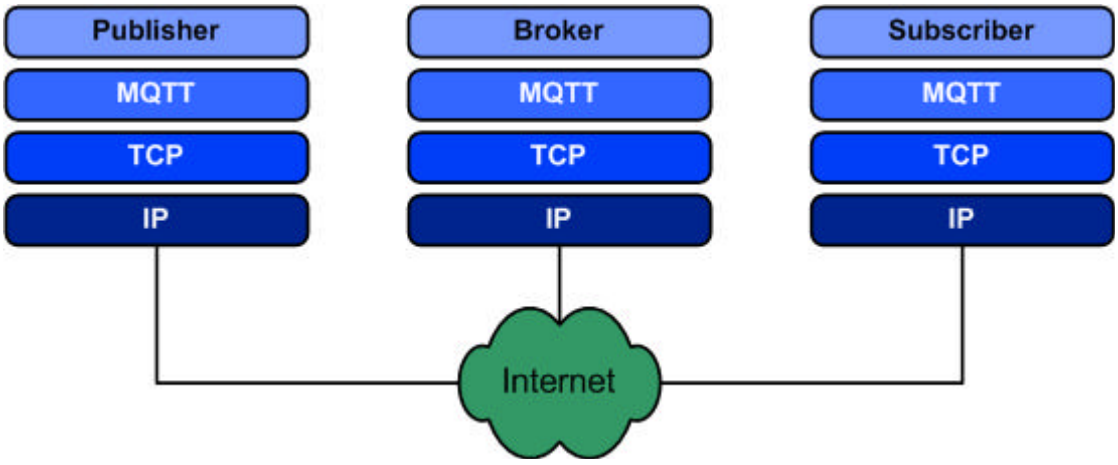

Рисунок 4.3 - Схема MQTT

MQTT працює за моделлю «видавець - передплатник» і використовує мінімальний набір методів, що визначають необхідні дії для комунікації з брокером та роботи з темами й повідомленнями. Агенти підключаються до брокера і виконують одну з двох основних функцій:

– публікують повідомлення в певні теми (видавці);

– підписуються на теми та отримують повідомлення, опубліковані в цих темах (передплатники).

Після завершення роботи агент відключається від брокера. Основні методи MQTT:

– Connect - встановлення з’єднання з брокером;

– Disconnect - розірвання з’єднання з брокером;

– Publish - публікація повідомлення в тему;

– Subscribe - підписка на тему;

– Unsubscribe - відписка від теми.

Спрощена схема взаємодії видавця та передплатника через MQTT-брокера наведена на рисунку 4.4.

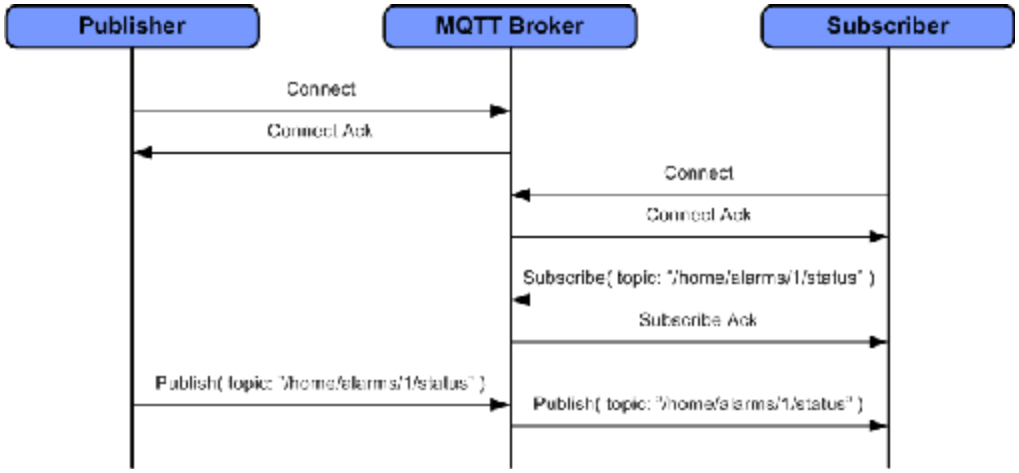

Рисунок 4.4 - Схема роботи MQTT

MQTT підтримує налаштування рівня якості обслуговування (QoS - Quality of Service), що визначає, як саме доставляються повідомлення. Існують три рівні QoS:

– QoS 0 - «максимум одноразова доставка». Повідомлення надсилається лише один раз без підтвердження від отримувача. Це метод «відправив і забув».

– QoS 1 - «мінімум одноразова доставка». Гарантує, що повідомлення буде отримане принаймні один раз. Передплатник може отримати його кілька разів, а відправник повторює спроби до отримання підтвердження успішної доставки.

– QoS 2 - «одноразова доставка». Найнадійніший, але найповільніший рівень. Використовується чотириступінчастий процес підтвердження доставки, що забезпечує гарантовану доставку повідомлення лише один раз.

Вибір рівня QoS залежить від критичності переданих даних і необхідності їх точної доставки.

Для обміну інформацією було обрано хмарний сервіс CloudMQTT, пакет послуг Cute Cat (рисунок 4.5). Сервіс надає до 10 безкоштовних з'єднань зі швидкістю 10 Kbit/s, що цілком достатньо для тестової моделі.

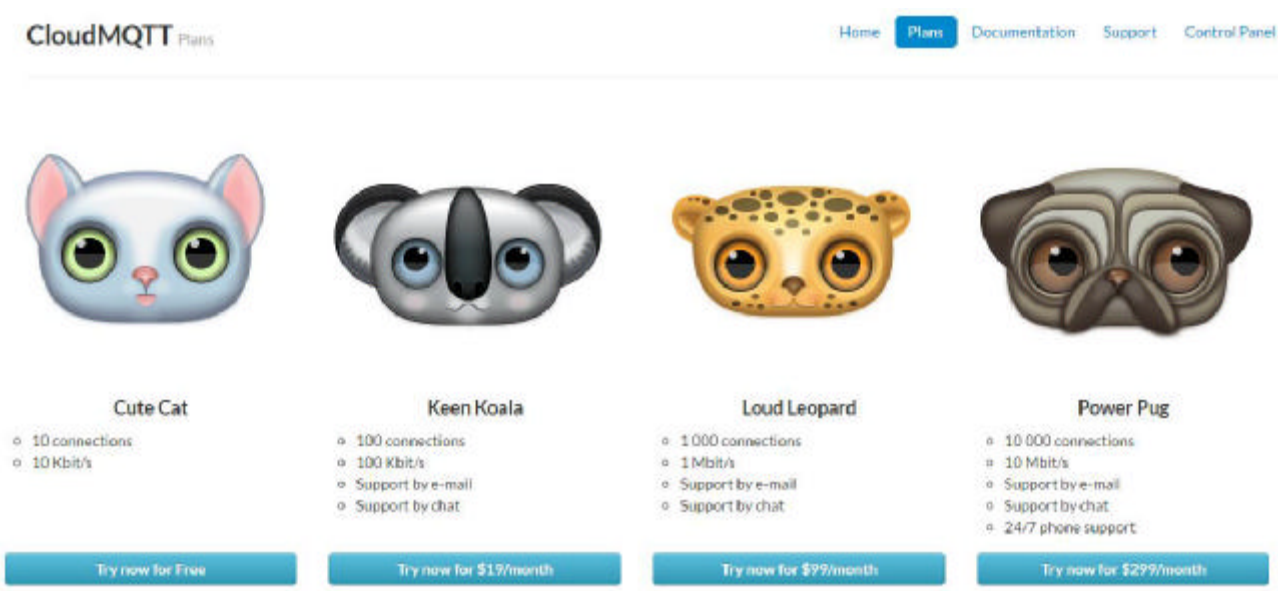

Рисунок 4.5 - Бізнес-плани CloudMqtt

Передача даних від датчиків здійснюється за наступним графіком: датчик температури та вологості передає дані раз на 3 секунди, а датчик шуму та сенсорна панель - раз на секунду, але лише при зміні значень.

Веб-застосунок для керування робочою моделлю

Для спрощення архітектури та через невелике навантаження робочої моделі було прийнято рішення об’єднати бізнес-логіку та серверну частину сайту на одному сервері. Використано NodeJS з бібліотекою Express. Оскільки фронтенд реалізований на JavaScript, цей вибір дозволяє швидко та ефективно інтегрувати серверну та клієнтську частини, забезпечуючи достатню продуктивність для обробки даних від датчиків та передачі через доступний канал зв’язку.

Для забезпечення швидкого та безперервного обміну даними між веб-сторінкою та сервером застосовано протокол Socket.IO, який дозволяє миттєво реагувати на зміни показників датчиків.

У хмарному сервісі Amazon було виділено віртуальний сервер EC2 з ОС Ubuntu, на який встановлено веб-сервер NodeJS+Express та розгорнуто базу даних PostgreSQL. Amazon надає рік безкоштовного використання у моделі IaaS,

що дозволяє самостійно налаштовувати середовище під потреби користувача. PostgreSQL є потужною базою даних, яка повністю відповідає вимогам робочої моделі для зберігання показників датчиків та інформації про користувачів системи.

Схема з‘єднання показана на рисунку 4.6

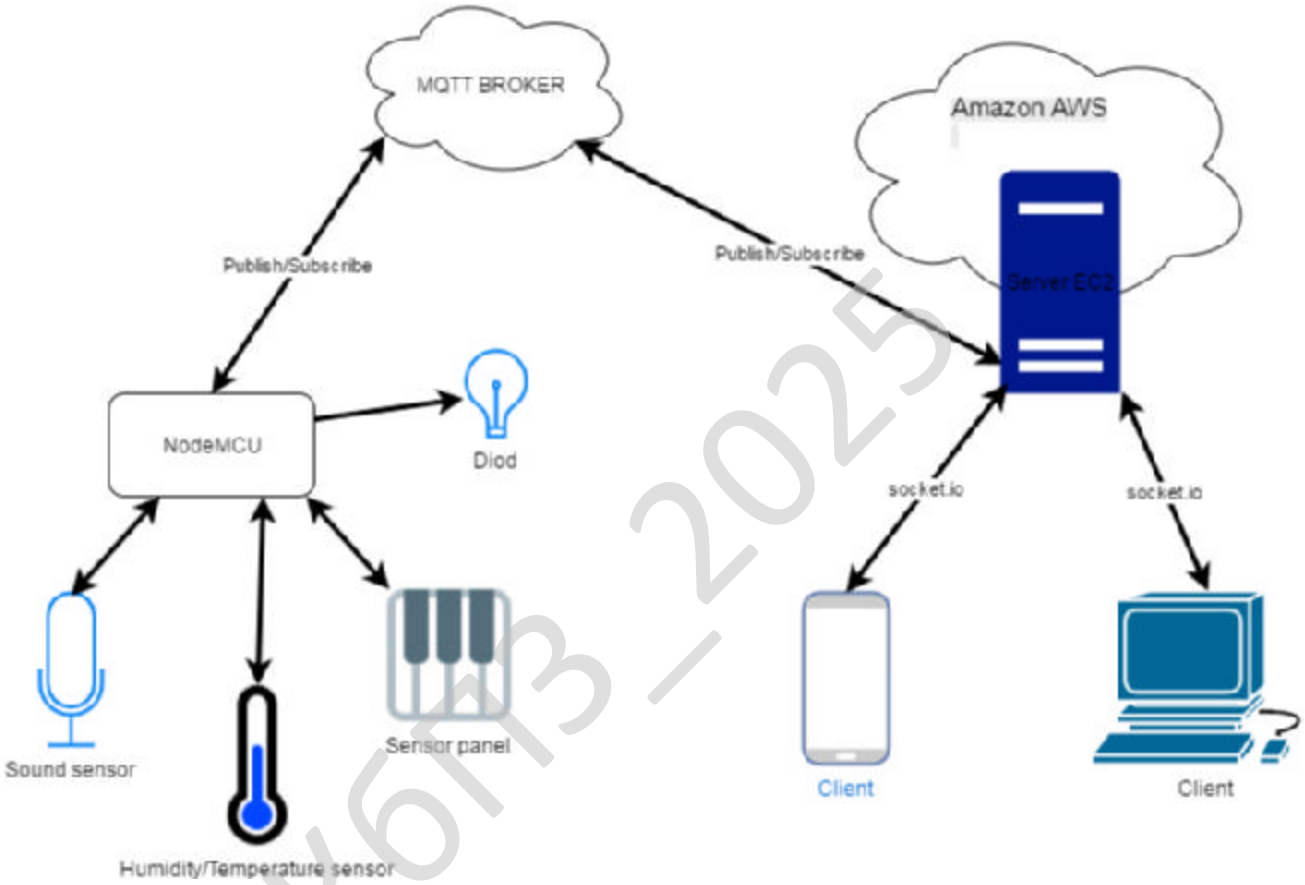

Рисунок 4.6 - Схема зв'язку робочої моделі з хмарним керуванням

Було створено веб-інтерфейс, представлений на рисунку 4.7, що дозволяє моніторити показники температури та вологості з підключених датчиків. Крім цього, інтерфейс відображає рівень звукового шуму, який перевищує встановлений пороговий рівень. Також передбачена функція дистанційного керування світлодіодом, підключеним до мікроконтролера, що дозволяє його вмикати або вимикати віддалено.

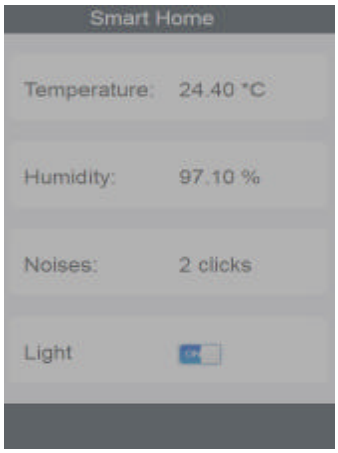

Рисунок 4.7 - Веб-інтерфейс моніторингу даних

Стартова сторінка веб-сайту представлена у вигляді вікна входу (рисунок 4.8). Доступ до даних з датчиків та можливість керування пристроями розумного будинку сторонніми особами обмежено. Тільки авторизовані користувачі можуть здійснювати моніторинг та управляти підключеними пристроями.

Рисунок 4.8 - Стартове вікно логіну сайта

Коли сервер отримує дані від MQTT-брокера, то після кожних десяти отриманих показників вологості та температури обчислюється їх середнє значення. Це середнє значення заноситься до відповідної таблиці бази даних з часовим штампом, що дозволяє надалі проводити аналіз даних та формувати інфографічні моделі.

Висновок

Хоча тестова схема відображає лише невеликий набір даних, вона демонструє модель із великим потенціалом для масштабування. Використані датчики є абстрактним представленням більш складних пристроїв, які легко можуть бути інтегровані у систему, розширюючи її функціонал відповідно до потреб користувача. Було продемонстровано двосторонню роботу хмарного керування: від датчиків до веб-інтерфейсу та навпаки.

Важливим є те, що система хмарного керування розумним будинком, навіть з невеликою кількістю змінних, може бути адаптована до інших хмарних платформ. Це можливо завдяки використанню сучасних та загальноприйнятих технологій: протоколу MQTT та socket.io для передачі даних, JavaScript та HTML/CSS для веб-інтерфейсів, бази даних PostgreSQL для збереження інформації та мови програмування C для налаштування мікроконтролера. Така архітектура забезпечує гнучкість, масштабованість та сумісність із різними середовищами розумного будинку.

4.1 Блок-схеми та опис алгоритмів функціонування системи

Щойно користувач переходить на головну сторінку сайту для онлайн-керування параметрами, йому необхідно авторизуватися або, у разі відсутності облікового запису, пройти процедуру реєстрації. На рисунку 4.9 представлена блок-схема алгоритму роботи сторінки реєстрації користувача.

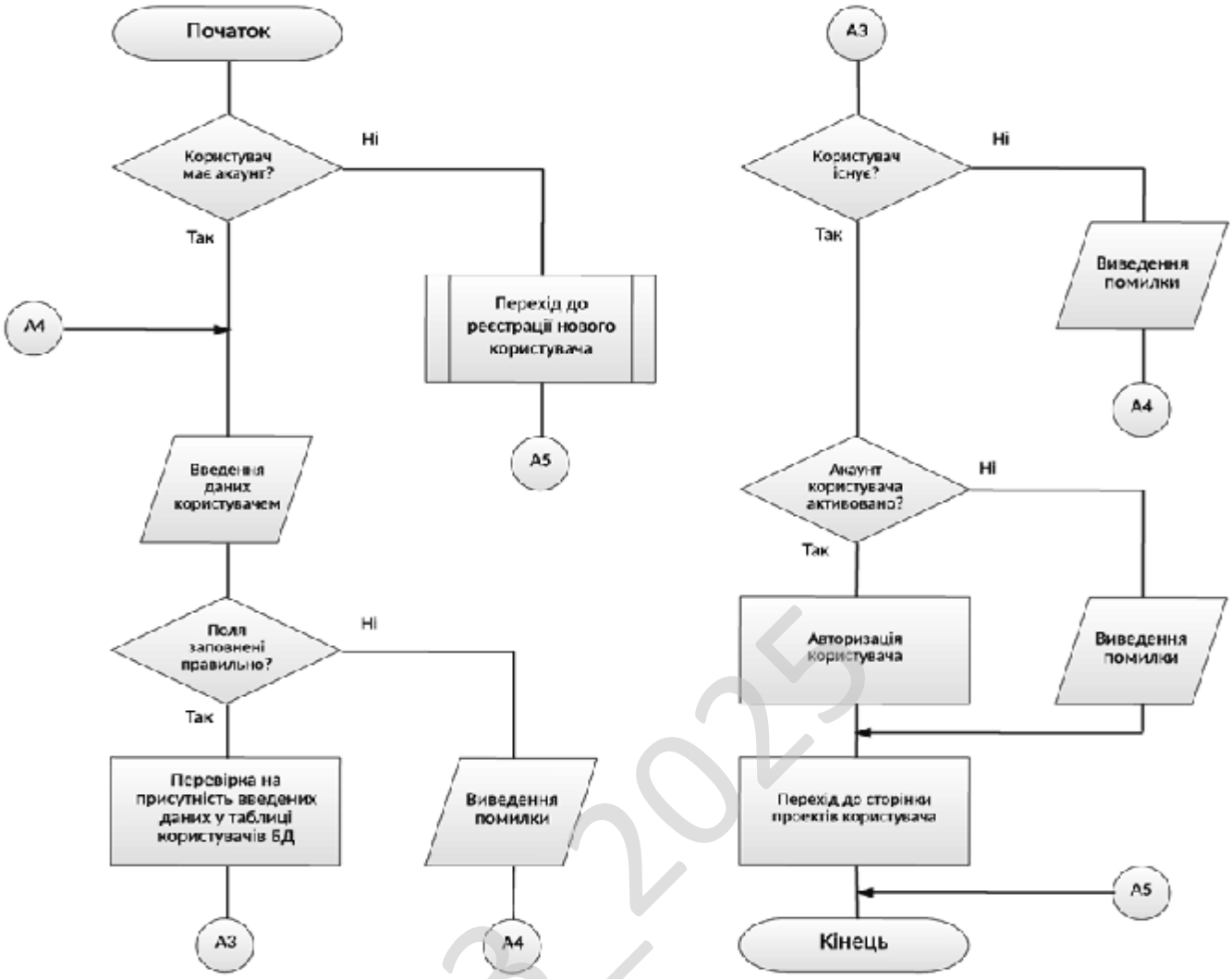

Рисунок 4.9 - блок-схема алгоритму роботи сторінки реєстрації користувача.

Розглянемо алгоритм роботи програмної частини системи захисту розумного будинку. Його блок-схема зображена на рисунку 4.10.

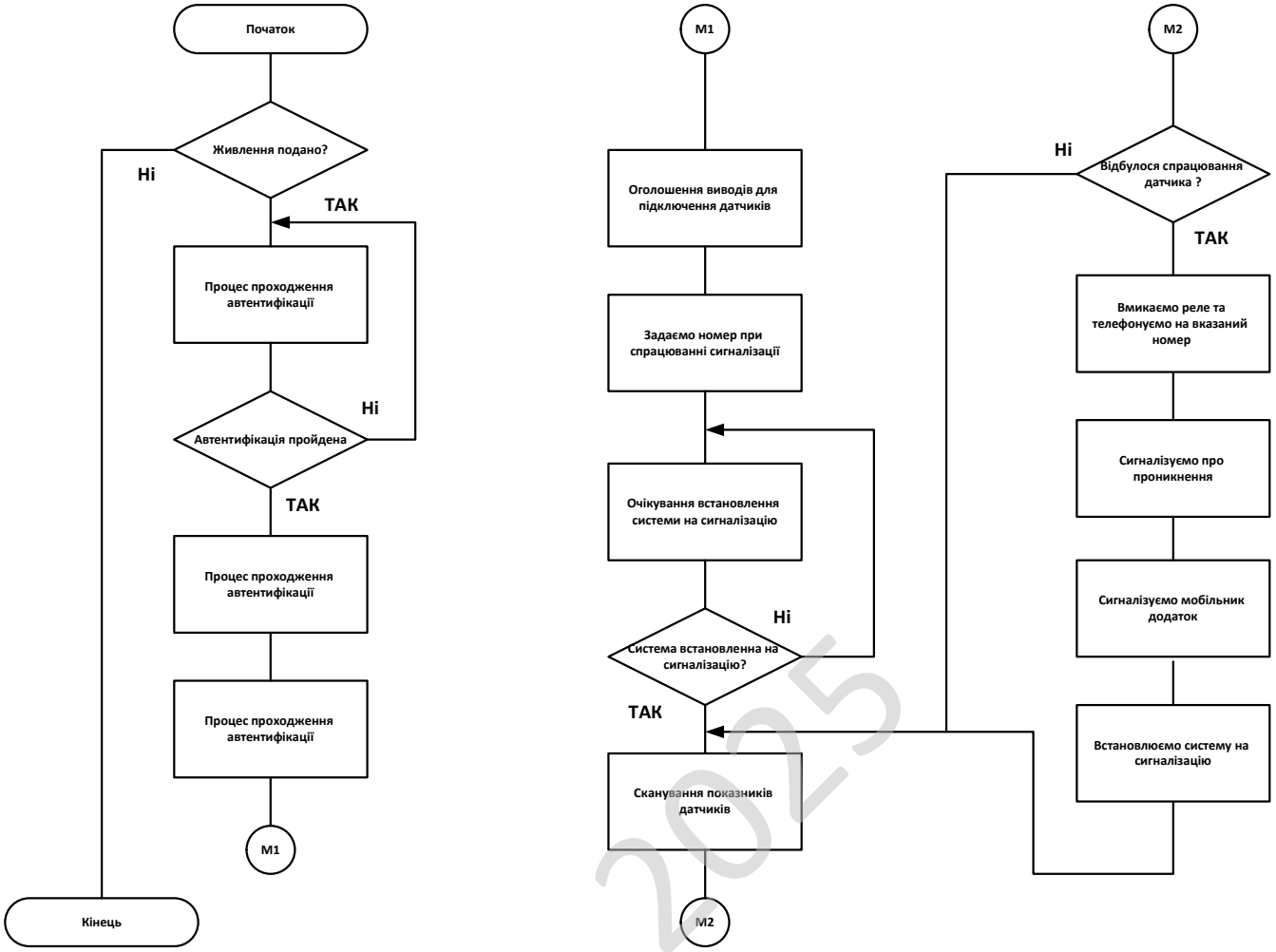

Рисунок 4.10 - Алгоритм роботи програмної частини системи захисту розумного будинку

З рисунку видно, що після запуску програми спочатку відображається основне вікно програми. Далі відбувається ініціалізація бази даних та підключення необхідних бібліотек. На наступному етапі здійснюється автоматичне налаштування зв’язку між мікроконтролером та пристроями системи. Якщо перевірка проходить успішно і надходить відповідний сигнал, система починає сканування всіх пристроїв із виводом отриманих даних на монітор. Потім встановлюються параметри системи за замовчуванням та передаються керуючі сигнали до мікропроцесора. Після цього система аналізує отримані дані і, у разі перевищення заданої споживаної потужності, відключає

надлишкові пристрої відповідно до алгоритму, збереженого в базі даних. Таким чином реалізується один із кроків управління системою «Розумний дім». Інші підсистеми функціонують за аналогічним алгоритмом.

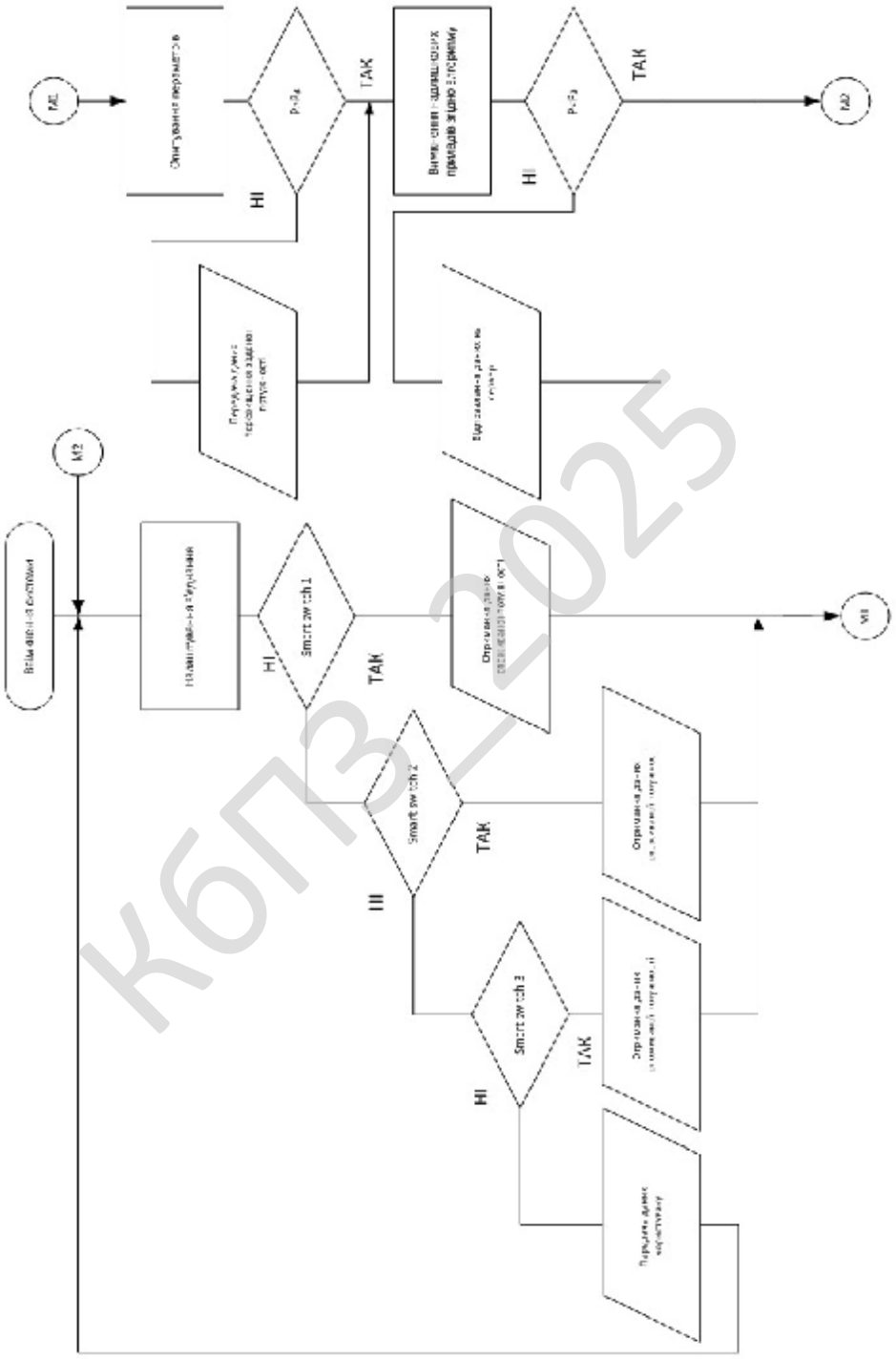

Рисунок 4.11 - блок-схема алгоритму запуску системи.

Алгоритм операції виконання дії

Алгоритм виконання дії в системі «Розумний дім» виглядає наступним чином:

Спочатку перевіряється доступність обраного фізичного пристрою. Якщо пристрій недоступний, користувач отримує повідомлення про його відсутність у мережі. У разі доступності пристрою надсилається запит із параметрами, що змінюють його стан відповідно до вибору користувача.

Якщо зміна стану пройшла успішно, система виводить повідомлення про успішну операцію; у разі виникнення помилки - відображається повідомлення про збій. Алгоритм цієї операції наведено на рисунку 4.12.

Рисунок 4.12 - Алгоритм операції виконання дії

Алгоритм роботи планувальника дій

Алгоритм роботи планувальника дій у системі «Розумний дім» реалізований у вигляді циклу з перевіркою настання заданого часу. Коли встановлений час досягається, виконується запрограмована дія; якщо час ще не настав - система перебуває в режимі очікування до настання моменту виконання.

Схема алгоритму наведена на рисунку 4.13.

Рисунок 4.13 - Алгоритм роботи планувальника задач

Алгоритм роботи сервісу моніторингу системи

Алгоритм моніторингу стану системи «Розумний дім» працює наступним чином: спочатку перевіряється доступність обраного фізичного пристрою. Якщо пристрій недоступний, користувач отримує відповідне повідомлення; у разі доступності пристрою надсилається запит для отримання його поточного стану згідно з вибором користувача.

Схема алгоритму наведена на рисунку 4.14.

Рисунок 4.14 - Алгоритм операції виконання дії

Блок схема основного алгоритма представлено на рисунку 4.15

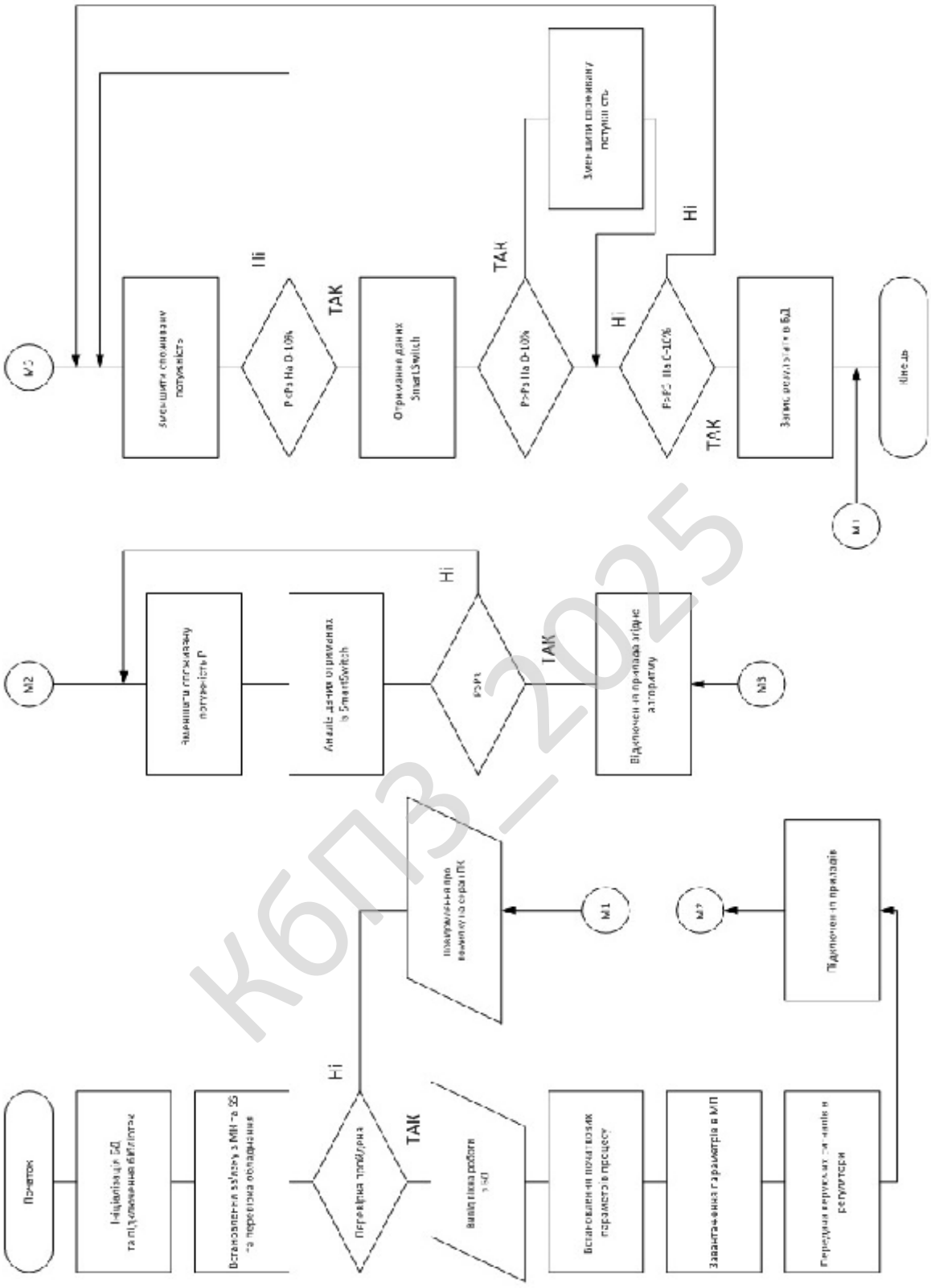

Рисунок 4.15 - блок-схема основного алгоритму роботи системи

4.2 Захист розробленого програмного забезпечення

Захист інформації можна реалізувати різними способами. Один із варіантів - шифрування та дешифрування повідомлень безпосередньо на пристроях IoT. Однак тут може виникнути проблема недостатньої продуктивності, оскільки більшість таких пристроїв є мікроконтролерами з обмеженими ресурсами.

Інший підхід полягає у покладанні відповідальності за надійність і безпеку передачі даних на комп'ютерну мережу. У цьому випадку передбачається, що мережа, до якої підключені пристрої, достатньо надійна та ізольована. Для доступу до віддалених сегментів мережі можна використовувати VPN або SSH-тунелі.

Захист Wi-Fi

Wi-Fi є одним із найпоширеніших способів бездротового підключення до локальних мереж. Нині ця технологія використовується практично повсюдно, що робить її одним із найзручніших варіантів для побудови IoT-систем. Загалом Wi-Fi вважається достатньо безпечним за умови вимкнення потенційно вразливих режимів роботи. До таких належать шифрування WEP та технологія QSS, оскільки вони дозволяють зловмисникам відносно легко отримати доступ до мережі, тому їх використання не рекомендується.

Найбільш поширеним стандартом захисту на сьогодні є WPA2-PSK, що базується на алгоритмі шифрування AES. Питання злому точок доступу детально розглянуто у статті [26]. Автор зазначає, що головним методом атаки залишається повний перебір паролів. Так, для перевірки словника обсягом близько 250 мільйонів паролів (приблизно 2 ГБ) на звичайному ноутбуку потрібно орієнтовно 66 годин. У свою чергу, словник, що міститиме всі можливі комбінації дев'ятисимвольного пароля, налічуватиме приблизно 900 мільйонів варіантів, і для його повного перебору знадобиться вже кілька тижнів.

Обмеження доступу у протоколах передачі

Більшість протоколів передачі даних підтримують механізми аутентифікації. Хоча інколи їх ігнорують для спрощення процесу обміну інформацією, це зменшує рівень безпеки, адже навіть базові заходи аутентифікації значно ускладнюють роботу потенційних зловмисників.

Зокрема, протоколи MQTT, REST та SNMP надають можливості аутентифікації, що дозволяє ефективно забезпечувати захист переданих даних та контроль за керуванням пристроями.

Модель підключення через VPN/SSH

VPN та SSH є найпоширенішими засобами побудови безпечних тунелів для передачі даних. При використанні VPN необхідно налаштувати мережеве обладнання для коректної маршрутизації трафіку, або ж цю функцію може виконувати сам міні-ПК. У випадку застосування SSH-тунелів дані передаються через відповідний мережевий протокол, для чого потрібен пристрій із запущеним SSH-сервером та налаштованою переадресацією портів.

Обидва підходи ефективно застосовуються при роботі з міні-комп'ютерами на ОС Linux. Для підвищення безпеки SSH існує безліч методів захисту, зокрема обмеження кількості невдалих спроб підключення, що дозволяє значно ускладнити проведення атак методом перебору паролів [27].

У типовій схемі підключення міні-ПК налаштовує тунель через VPN або SSH до централізованої системи, а датчики на базі мікроконтролерів підключаються через нього до центрального сервера або безпосередньо через MQTT, або з використанням MQTT-брокера. У другому випадку брокер відповідає за отримання повідомлень у своїй мережі та їх подальшу ретрансляцію до централізованого вузла.

Таким чином, доцільним є використання стандартних механізмів безпеки, передбачених протоколами взаємодії та мережевої передачі даних. Крім того, слід дотримуватися принципу, що безпеки забагато не буває: не використовувати прості паролі та увімкнути аутентифікацію там, де це можливо.

Захист програмного забезпечення

Для забезпечення захисту даних у розробленій веб-системі командного керування проектами були використані стандартні функції мови PHP: `password_hash()` та `password_verify()`.

Функція `password_hash()` застосовується для хешування паролю користувача, а `password_verify()` - для перевірки відповідності введеного паролю збереженому хешу.

Перший параметр функції `password_hash()` задає пароль, що підлягає захисту, а другий параметр визначає алгоритм хешування. На даний момент підтримуються два алгоритми:

– `PASSWORD_DEFAULT` - використовує алгоритм BCrypt, що генерує хеш довжиною від 60 символів;

– `PASSWORD_BCRYPT` - застосовує алгоритм CRYPT_BLOWFISH і завжди повертає рядок довжиною 60 символів.

Головною перевагою цих функцій є автоматичне управління значенням «солі» та складністю обчислення хешу, що значно спрощує роботу програміста. У разі потреби можна передати власне значення «солі» через третій параметр функції.

Приклад використання функції без власної «солі»:

```
<?php echo password_hash("rasmuslerdorf", PASSWORD_DEFAULT); ?>
```

Результатом виконання може бути:

```
$2y$10$.vGA1O9wmRjrwAVXD98HNOgsNpDczlqm3Jq7KnEd1rVAGv3Fykk1a
```

Приклад з передачею власних параметрів «солі» та вартості обчислення:

```
<?php
$options = [ 'cost' => 12 ];
echo password_hash("rasmuslerdorf", PASSWORD_BCRYPT, $options);
?>
```

Результатом буде:

```
$2y$12$QjSH496pcT5CEbzjD/vtVeH03tfHKFy36d4J0Ltp3lRtee9HDxY3K
```

Як видно, два хеші відрізняються один від одного, що демонструє роботу алгоритму з різними параметрами.

Крім цього, сайт був захищений від XSS-атак та SQL-ін’єкцій.

Для запобігання XSS-атакам використовувались функції `htmlspecialchars()` та `strip_tags()`, які замінюють спеціальні символи, такі як < та >, на безпечні еквіваленти (`<` і `>`). Додатково можлива конфігурація у файлі `.htaccess`:

```
Options +FollowSymLinks
RewriteEngine On
RewriteCond %{QUERY_STRING} (\<|%3C).*script.*(\>|%3E) [NC,OR]
RewriteCond %{QUERY_STRING} GLOBALS(=|\[|\%[0-9A-Z]{0,2}) [OR]
RewriteCond %{QUERY_STRING} _REQUEST(=|\[|\%[0-9A-Z]{0,2})
RewriteRule ^(.*)$ index.php [F,L]
```

Для захисту від SQL-ін’єкцій застосовано функцію `mysql_real_escape_string()`, яка екранує спеціальні символи у рядках SQL-запитів з урахуванням кодування з’єднання.

5 ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Реєстрація домена

Реєстрація доменів - це процес внесення запису про нове доменне ім’я до реєстру зони першого рівня. Процедура реєстрації є відносно простою: необхідно створити акаунт у реєстратора доменних імен, поповнити рахунок, перевірити доступність обраного доменного імені та, у разі його вільності, подати заявку на реєстрацію.

Після внесення запису до реєстру, який містить інформацію про адміністратора, реєстратора, дати реєстрації та її закінчення, а також стан делегування, доменне ім’я стає готовим для використання, зазвичай протягом 5-10 хвилин. Для функціонування домену необхідно делегувати його на DNS-сервери через інтерфейс реєстратора або хостинг-провайдера.

Реєстратор доменних імен - це організація, уповноважена створювати нові доменні імена та подовжувати термін дії вже існуючих у доменах, для яких встановлено обов’язкову реєстрацію. До таких доменів належать:

- кореневий домен (домен нульового рівня);
- всі домени верхнього рівня (першого рівня);
- окремі домени другого рівня (наприклад, com.ru або co.uk).

У решті доменів для створення субдоменів спеціальні повноваження не потрібні.

Хостинг

Хо́стинг (англ. hosting) - це послуга, що передбачає надання дискового простору, підключення до мережі та інших ресурсів для розміщення інформації на сервері, який постійно підключений до мережі (наприклад, Інтернет).

Поняття хостингу охоплює широкий спектр сервісів із використанням різного апаратного та програмного забезпечення. Зазвичай під хостингом

розуміють, як мінімум, розміщення файлів веб-сайту на сервері з відповідним програмним забезпеченням для обробки запитів (веб-сервер). Крім того, послуги хостингу часто включають надання місця для поштових скриньок, баз даних, DNS, файлового сховища та підтримку роботи цих сервісів. Водночас деякі з цих послуг можуть надаватися окремо, наприклад:

– поштовий хостинг - розміщення електронної кореспонденції та відповідного ПЗ;

– файловий хостинг - розміщення клієнтських файлів;

– відео-хостинг - розміщення виключно відеофайлів;

– інші спеціалізовані види хостингу за певними умовами.

Послуги хостингу часто пропонуються у складі пакетів із додатковими інформаційними сервісами, такими як реєстрація доменного імені, створення веб-сайтів, надання додаткового програмного забезпечення тощо.

Провайдерами хостингу можуть бути як спеціалізовані компанії («хостери»), так і великі інформаційні провайдери, що надають інші послуги (наприклад, Google, Microsoft, Yahoo та інші).

Хостинг поділяють на безкоштовний та платний. Безкоштовні провайдери, як правило, отримують прибуток від розміщення реклами на своїх ресурсах або через надання додаткових платних сервісів, включаючи їх у пакет із безкоштовними послугами.

Завантаження вихідного коду

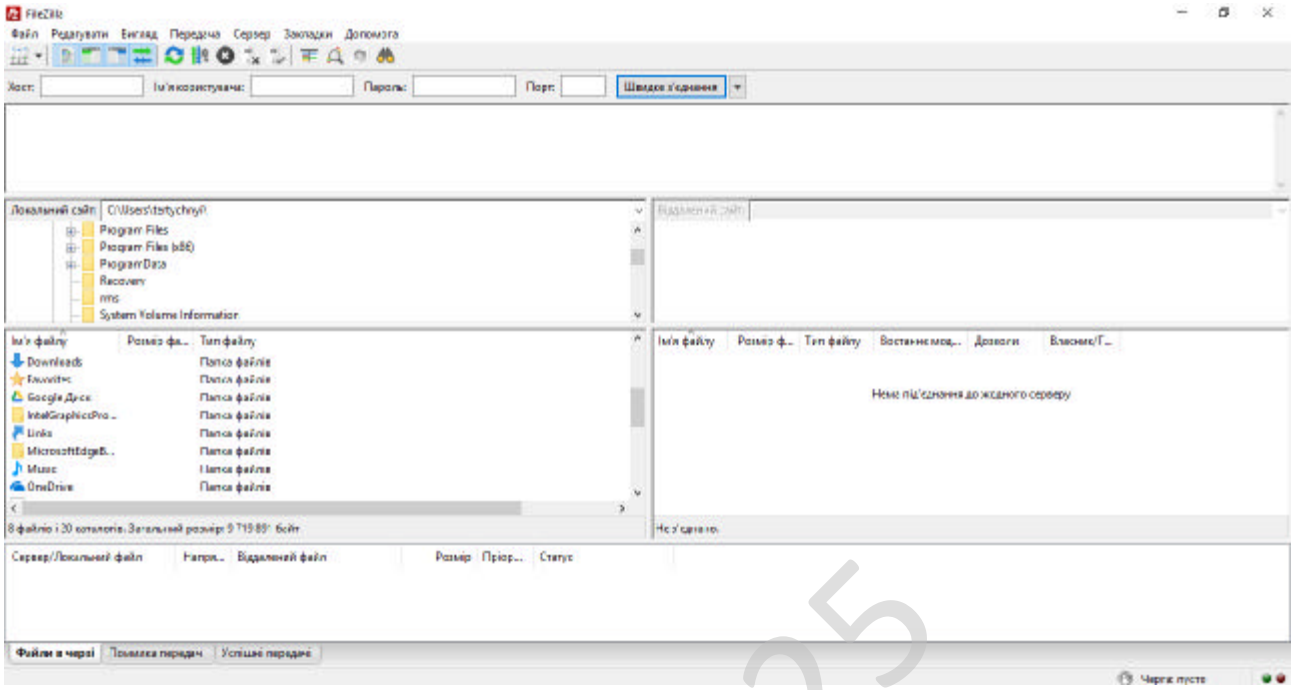

Рисунок 5.3 - Головне вікно програми для завантаження файлів на сервер

Після придбання доменного імені та підключення послуг хостингу необхідно завантажити на сервер вихідний код розробленого програмного забезпечення за допомогою програм, таких як FileZilla, та перевірити коректність роботи веб-сайту.

Управління базою даних сайту здійснюється через спеціалізовані інструменти. Одним із найпоширеніших є веб-додаток phpMyAdmin - програмне забезпечення з відкритим кодом, розроблене на мові PHP, що надає графічний веб-інтерфейс для адміністрування баз даних MySQL або MariaDB.

За допомогою phpMyAdmin можна адмініструвати сервер MySQL через браузер, виконувати SQL-запити, переглядати та редагувати вміст таблиць. Додаток є популярним серед веб-розробників, оскільки дозволяє керувати базами даних без необхідності введення команд SQL, використовуючи зручний інтерфейс і доступ з будь-якого комп'ютера, підключеного до Інтернету, без встановлення додаткового програмного забезпечення.

На сьогодні phpMyAdmin широко застосовується на практиці. Це пояснюється активним розвитком додатку розробниками з урахуванням усіх нововведень у СКБД MySQL. Більшість українських провайдерів використовують phpMyAdmin як панель управління, надаючи своїм клієнтам можливість зручно адмініструвати виділені бази даних.

Реалізація запуску системи управління

Система «Розумний дім» побудована на принципі модульної архітектури, де функціональні блоки реалізовані як незалежні модулі. Така структура дозволяє зменшити взаємозалежність компонентів та підвищити повторне використання реалізованого функціоналу.

Користувацький інтерфейс системи має зручний та інтуїтивно зрозумілий дизайн, що забезпечує низький поріг входу для користувачів. Інтерфейс тестування системи також спрощений і має графічне представлення, що полегшує проведення тестів. У документації надано опис основних компонентів програмного забезпечення та детальний опис роботи додатку, а також створено інструкцію користувача, яка демонструє послідовність дій при роботі з програмою.

Після запуску програми відображається вікно завантаження. Після завершення завантаження з'являється вікно авторизації, де користувач повинен ввести логін та пароль. У разі некоректного введення даних система повідомляє про помилку, і для повторного введення користувачеві необхідно натиснути кнопку «TRY AGAIN». Після коректної авторизації відображається вікно вибору функціоналу системи.

Користувач може обрати один із доступних варіантів:

- Моніторинг системи - відкривається відповідне вікно функціоналу, де відображається стан вибраного пристрою у вигляді діалогового вікна.

- Операції з системою - відкривається вікно функціоналу для управління пристроями. Тут відображається поточний стан вибраного пристрою та доступні дві кнопки: для встановлення нового стану або повернення до попереднього

меню. Для зміни параметрів, наприклад температури, користувач обирає нове значення за допомогою спіннера та натискає «ACCEPT». Після цього він повертається до меню «Операції з системою», а для повернення до головного меню використовується свайп вліво.

- Планувальник завдань - відкриває вікно для встановлення часу виконання операцій. Користувач задає час та підтверджує натисканням кнопки «ОК» або відміняє операцію кнопкою «Cancel». Для перегляду списку запланованих завдань застосовується свайп вправо. Видалення завдань здійснюється свайпом по елементу, редагування - затримуванням пальця на елементі та натисканням кнопки редагування.

Таким чином, у проекті системи «Розумний дім» реалізовані всі визначені раніше вимоги, що забезпечує повноцінне функціонування та зручність користування системою.

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення для системи хмарного керування пристроями комплексу рішень «Розумний дім».

Об'єктом дослідження є процес забезпечення хмарного керування «розумним будинком».

Предметом дослідження є методи реалізації систем хмарного керування розумним будинком.

Методи дослідження базуються на методах теорії кодування,методах захисту інформації, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено систему ухмарного керування розумним будинком.

– Проведено огляд технологій зв'язку в системах «Розумний дім».

– Розроблено вітчизняний продукт хмарного керування розумним будинком, який має більш широкі можливості, на відміну від існуючих аналогів.

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати цього дослідження можуть бути цікавими насамперед звичайним користувачам - власникам квартир і приватних будинків, які хочуть мати більше контролю над своєю безпекою та комфортом. Люди вже звикли керувати освітленням, температурою або камерами зі смартфона, але їм часто бракує єдиного центру керування, який би об’єднав усе це в одну систему. Саме тому для мешканців, які прагнуть спростити побут і почуватися захищеними, ця розробка має практичну цінність.

Також проєкт може зацікавити забудовників і керуючі компанії. Для них важливо мати інноваційні рішення, які підвищують цінність житлових комплексів на ринку. Хмарна система керування може стати додатковою конкурентною перевагою, адже сучасні покупці нерухомості вже очікують на наявність «розумних» технологій як на норму, а не як на розкіш. Не менш важливими зацікавленими сторонами є компанії, що спеціалізуються на встановленні сигналізацій, відеоспостереження чи охоронних систем. Вони можуть інтегрувати хмарне рішення у свої сервіси, зменшуючи навантаження на фізичну охорону та підвищуючи ефективність реагування на події. Це відкриває для них можливості масштабування та підвищення рентабельності.

Окремо варто згадати ІТ-компанії та стартапи, що працюють у сфері інтернету речей. Для них результати дослідження можуть стати основою для подальшої розробки нових функцій, модулів або навіть партнерських рішень. Хмарна архітектура - це гнучкий фундамент, який дозволяє швидко розширювати можливості та адаптувати продукт під різні ринки. На рівні громади або муніципалітету цей проєкт теж може мати значення. Хмарне централізоване управління може використовуватися для створення безпечних районів, організації колективного моніторингу або підвищення рівня взаємодії між мешканцями в

межах одного району. Це робить проєкт важливим не лише для окремих користувачів, але й для ширших соціальних змін.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Оцінку привабливості цього проєкту можна здійснити через залучення групи експертів, які мають практичний досвід у сферах безпеки, ІТ-розробки, житлової інфраструктури та інтернету речей. Зазвичай кожен експерт формує власне бачення перспективності проєкту, виходячи зі свого професійного досвіду. Саме так можна забезпечити реалістичність і незалежність оцінки.

Наприклад, експерти з охоронних компаній можуть оцінити, наскільки хмарна система дійсно здатна зменшити кількість інцидентів та витрати на фізичну охорону. Їхня думка є важливою, оскільки вони безпосередньо стикаються з проблемами оперативного реагування та фіксують статистику ризиків у реальних умовах.

Інженери та ІТ-фахівці здатні дати оцінку технічній життєздатності проєкту. Вони можуть визначити, наскільки масштабованою є система, чи може вона витримати навантаження від сотень будинків та тисяч пристроїв, і як складно буде інтегрувати нові модулі штучного інтелекту. Їхні оцінки допоможуть зрозуміти, чи відповідає проєкт тенденціям ринку та актуальним технологічним стандартам.

Представники забудовників або управляючих компаній оцінять проєкт з точки зору ринкової привабливості та комерційного потенціалу. Для них важливо зрозуміти, чи підвищить система цінність нерухомості та чи стане вона додатковим аргументом для покупців житла. Думка таких експертів дозволяє оцінити бізнес-перспективу розробки.

У результаті, застосування методу експертних оцінок дає можливість сформувати середній інтегральний показник привабливості, який буде

враховувати технічні, економічні та ринкові аспекти. Такий підхід допомагає мінімізувати суб'єктивність і отримати реалістичну картину майбутнього проєкту.

7.3 Вибір методу оцінки вартості ПЗ

Для оцінки вартості проєкту «Розумний дім» найкраще застосувати метод, який враховує не лише поточні витрати, а й майбутні вигоди та економію. У цьому випадку найбільш доречним є дисконтований грошовий потік, адже він дозволяє оцінити реальну вартість проєкту з урахуванням майбутніх економічних ефектів у часі. Це особливо важливо для технологічних рішень, де економія формується поступово.

Цей підхід дає змогу більш точно оцінити окупність, оскільки проєкт впливає на зменшення витрат на охорону та ліквідацію інцидентів упродовж багатьох років. Враховуючи швидкість розвитку технологій та можливість розширення системи, важливо розуміти, як ці зміни впливатимуть на її довгострокову ефективність.

Метод DCF підходить і тому, що дозволяє врахувати різні сценарії розвитку. Наприклад, активне розширення системи на більшу кількість будинків, удосконалення алгоритмів машинного навчання або підвищення цін на традиційні послуги охорони. Такі сценарії можуть суттєво змінити показники проєкту, і дисконтований підхід дає змогу це точно врахувати.

Крім того, метод DCF краще відображає інноваційну природу проєкту. У технологічній сфері цінність рішення часто полягає не в матеріальних компонентах, а в майбутньому потенціалі, масштабуванні та підвищенні вартості сервісу. Саме цей метод дозволяє включити вартість інтелектуальної складової, що є критично важливим у випадку хмарної платформи.

Таким чином, використання дисконтованого грошового потоку дозволяє максимально точно й об'єктивно оцінити вартість та економічну перспективу проєкту, враховуючи всі його фінансові й стратегічні особливості.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Оцінка економічної ефективності впровадження системи хмарного керування пристроями комплексу рішень «Розумний дім»

Розглянемо типовий житловий комплекс (або масив приватної забудови), у якому встановлено систему «розумного дому», що включає десятки пристроїв: відеокамери, датчики руху, диму, температури, «розумні» замки, освітлення тощо. За відсутності централізованого хмарного керування безпекою система залишається фрагментованою: дані розпорошені між окремими пристроями, оновлення програмного забезпечення затримуються, а реагування на інциденти не завжди є своєчасним.

Впровадження **хмарного сервісу керування та безпеки комплексу «Розумний дім»** дає змогу централізувати контроль за всіма пристроями, інтегрувати алгоритми машинного навчання для виявлення підозрілої активності, забезпечити цілодобовий моніторинг 24/7 з будь-якої точки світу. Це підвищує рівень захисту мешканців, скорочує витрати на фізичну охорону та зменшує кількість хибних сповіщень. Вихідні дані для розрахунку наведено в таблиці 7.1.

Таблиця 7.1 – Вихідні дані для розрахунку економічної ефективності системи хмарного керування

Показник	До впровадження	Після впровадження	Економічний ефект
Кількість будинків у системі	500	500	-
Середні річні витрати на фізичну охорону одного будинку	36 000 грн	18 000 грн	–18 000 грн на будинок
Кількість інцидентів на рік (крадіжки, вторгнення, помилкові тривоги)	120	30	–75 %
Витрати на усунення наслідків інцидентів	600 000 грн	150 000 грн	–450 000 грн
Річне обслуговування ІТ-системи безпеки	1 200 000 грн	800 000 грн	–400 000 грн
Початкові інвестиції у створення хмарного сервісу (розробка, сервери, тестування, навчання)	-	-	2 500 000 грн

На основі даних таблиці виконаємо поетапний розрахунок.

1. Сукупні витрати на систему безпеки до впровадження хмарного рішення. Річні витрати на фізичну охорону всіх будинків - 18 000 000 грн.

Витрати на усунення наслідків інцидентів - 600 000 грн

Річне обслуговування ІТ-системи безпеки - 1 200 000 грн

Загальні річні витрати до впровадження =19 800 000 грн/рік.

2. Сукупні витрати на систему безпеки після впровадження хмарного рішення.

Річні витрати на фізичну охорону всіх будинків =9 000 000 грн.

Витрати на усунення наслідків інцидентів =150 000 грн.

Річне обслуговування оновленої ІТ-системи безпеки =800 000 грн.

Загальні річні витрати після впровадження = 9 950 000 грн/рік.

3. Річна економія (зменшення витрат). Річна економія від впровадження хмарного сервісу становитиме = 9 850 000 грн/рік.

4. Чистий економічний ефект у перший рік з урахуванням початкових інвестицій=7 350 000 грн.

Отже, чистий економічний ефект у перший рік експлуатації системи становить 7,35 млн грн.

5. Термін окупності проєкту. Термін окупності (Ток) визначимо за формулою≈0,25 року, що відповідає приблизно 3 місяцям. Таким чином, інвестиції у створення хмарної системи керування комплексом «Розумний дім» окупаються менш ніж за квартал.

6. Рентабельність інвестицій (ROI). Рентабельність інвестицій можна оцінити за формулою≈394%. Отже, рентабельність інвестицій становить близько 394 %, що свідчить про високу економічну доцільність проєкту.

Окрім прямого економічного ефекту, впровадження хмарної системи керування пристроями комплексу «Розумний дім» забезпечує низку важливих нефінансових результатів:

- зменшення кількості інцидентів (використання алгоритмів машинного навчання для аналізу поведінкових патернів дозволяє виявляти вторгнення та підозрілу активність на ранніх стадіях);

- комфорт і дистанційний доступ (користувачі отримують можливість повного контролю системи безпеки й «розумних» пристроїв через мобільний застосунок або веб-інтерфейс у режимі 24/7),

- підвищення ринкової вартості нерухомості (об’єкти з інтегрованими системами «розумного дому» та хмарним сервісом безпеки мають вищу інвестиційну й споживчу привабливість),

- екологічний ефект (хмарна обробка даних зменшує потребу у великій кількості локальних серверів та обладнання, що сприяє зниженню енергоспоживання),

- соціальний ефект (зростає суб’єктивне відчуття безпеки мешканців, формується довіра до інноваційних рішень та з’являється можливість колективного моніторингу у межах житлового комплексу).

Отримані результати підтверджують високу економічну ефективність запровадження системи хмарного керування пристроями комплексу рішень «Розумний дім». Проєкт забезпечує суттєве скорочення експлуатаційних витрат, швидкий термін окупності інвестицій, високу рентабельність та значний нефінансовий ефект, пов’язаний із підвищенням рівня безпеки, комфорту та якості життя мешканців.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Просування проєкту хмарної системи «Розумний дім» доцільно почати з формування чіткого позиціонування. Люди мають одразу зрозуміти, яку саме проблему вирішує система, у чому її унікальність та чому вона краща за класичні охоронні підходи. Якщо зробити фокус на безпеці, зручності та економії, це створить чіткий образ продукту в очах потенційних користувачів.

Наступним кроком може стати пілотний запуск у реальному житловому комплексі. Показ реальної роботи системи завжди має більший ефект, ніж рекламні обіцянки. Коли люди бачать, що система справді знижує ризики та економить кошти на охороні, їхня довіра до продукту суттєво зростає. До того ж, результати пілоту можна використати як кейс у подальшому просуванні.

Паралельно важливо працювати з цифровим маркетингом. Цільові аудиторії часто шукають інформацію онлайн, тому потрібно створити зрозумілий сайт, серію демонстраційних відео та детальні інструкції, які покажуть переваги системи. Це допоможе користувачам прийняти рішення, навіть якщо вони не глибоко розуміються на технологіях.

Суттєву роль відіграє співпраця з забудовниками та керуючими компаніями. Для них система може стати частиною доданої вартості кожного житлового комплексу. Якщо інтегрувати послугу ще на етапі будівництва або ремонту, це значно здешевить встановлення для кінцевого користувача і збільшить попит.

Завершальним етапом може бути участь у виставках, форумах і технічних заходах. Демонстрація можливостей системи на професійних майданчиках допоможе підвищити довіру з боку експертів та партнерів, а також розширить коло потенційних клієнтів у B2B-сегменті.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для оптимізації каналів збуту важливо розуміти, як саме потенційні користувачі приймають рішення про покупку. У випадку «Розумного дому» багато хто хоче отримати готове рішення «під ключ», тому доцільно розвивати канали продажів через інсталяційні компанії та інтеграторів систем безпеки. Вони вже мають довіру клієнтів і розуміють їхні потреби, тож можуть якісно представити і встановити продукт.

Також варто посилити онлайн-продажі. Частина аудиторії вже звикла замовляти технологічні продукти через інтернет, тому доступність системи на офіційному сайті, на маркетплейсах або у вигляді підписки може значно розширити ринок. Онлайн-продажі дозволяють швидко реагувати на запити користувачів та надавати технічну підтримку у режимі 24/7.

Партнерство із забудовниками може стати найефективнішим каналом. Якщо система інтегрується на етапі будівництва, то мешканці отримують готовий продукт без додаткових витрат на монтаж. Це також відкриває можливість укладення довгострокових контрактів на обслуговування, що створює стабільний дохід для компанії.

Для бізнес-клієнтів варто розглянути варіант корпоративного сервісу. Торгові центри, бізнес-парки або готелі теж потребують централізованого хмарного керування безпекою. У цьому сегменті ключову роль відіграють персоналізація та масштабованість, тому важливо пропонувати індивідуальні пакети та професійний супровід.

Окремим напрямом може бути франчайзинг або партнерська програма для регіонів. Це дозволить швидко розширити покриття ринку, зберігши при цьому контроль за якістю сервісу та інтеграції. Чим більше професійних партнерів зможуть встановлювати й обслуговувати систему, тим швидше буде рости її популярність.

7.7 Визначення ключових факторів успіху конкретного проєкту

Успіх цього проєкту значною мірою залежить від надійності технології. Користувачі довіряють системі свою безпеку, тому будь-які збої можуть суттєво підірвати їхню довіру. Чим стабільніша, швидша та безпечніша працює хмарна платформа, тим більша ймовірність, що вона стане стандартом для всього комплексу «Розумний дім».

Не менш важливою є простота використання. Більшість людей не хочуть розбиратися в складних технологічних налаштуваннях, тому інтерфейс має бути інтуїтивним. Якщо керування будинком виглядатиме зрозуміло, а встановлення займатиме мінімум часу, користувачі швидко звикнуть до системи і почнуть рекомендувати її іншим.

Одним із ключових факторів можна назвати масштабованість. Житлові комплекси ростуть, а кількість пристроїв збільшується з кожним роком. Система має підтримувати можливість розширення без зниження якості роботи. Це забезпечить довговічність продукту та дозволить адаптувати його під нові виклики та технології.

Важливу роль відіграє і підтримка. Користувачі очікують, що у разі поломки чи запитання вони зможуть швидко отримати допомогу. Сервісна служба, що оперативно реагує на звернення, формує довіру й гарантує, що система не стане джерелом проблем.

І нарешті, важливою умовою успіху є вміння компанії працювати з партнерами. Забудовники, інсталятори, ІТ-компанії та охоронні служби можуть

значно прискорити розвиток проєкту. Якщо налагодити з ними тісну співпрацю, система «Розумний дім» може стати масштабним та впізнаваним рішенням на ринку.

8 ЗАХОДИ ЩОДО ОХОРОНИ ПРАЦІ І ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

Характерною ознакою сучасного науково-технічного прогресу практично у всіх сферах діяльності людини є широке застосування комп'ютерних технологій, заснованих на використанні електронно-обчислювальних машин (ЕОМ). Сьогодення, а тим більше, майбутнє, вже важко уявити без комп'ютерів та іншої електронної техніки. Адже саме завдяки їм стала можливою швидка переробка величезних обсягів інформації, проведення необхідних розрахунків, виконання різних видів робіт, пов'язаних обробкою текстових та ілюстраційних зображень, організація оперативного отримання та передачі інформації, збереження її значних обсягів електронним способом.

Стрімке впровадження комп'ютерів не тільки в сфері управління виробництвом, в банківській систему, бізнесі, системі освіти, але також на транспорті, сфері обслуговування призвело до того, що десятки мільйонів людей у всьому світі виявились втягнутими у взаємодію людини з комп'ютером. Природно виникає запитання: настільки безпечною є ця взаємодія для людини? Адже відома аксіома про те, що будь-яка взаємодія людини та засобів праці двостороння.

Людина впливає на удосконалення засобів праці, а останні –на працюючу людину. Отже, навіть сучасні технології та техніка, до яких безперечно, залежать комп'ютерні технології та ЕОМ несуть у собі певні потенційні небезпеки. У зв'язку з цим набуває актуальності адекватна оцінка конкретних умов і характеру праці, яка сприяє обґрунтованому розробленню та впровадженню комплексу заходів і засобів, спрямованих на збереження здоров’я і працездатності людини в процесі праці за рахунок поліпшення параметрів виробничого середовища, зменшення важкості, напруженості

трудового процесу та збереження здоров'я працівників на комп'ютеризованих робочих місцях.

Законодавством України чітко врегульовано норми та вимоги до використання комп’ютерної техніки на підприємстві, безпосередньо й охорона праці на підприємстві при роботі за комп’ютером., зокрема «Вимоги щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями», затверджені наказом Мінсоцполітики від 14.02.2018 № 207 [1], «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98. [2].

Загальні вимоги пожежної безпеки під час експлуатації комп’ютерної техніки визначають «Правила пожежної безпеки в Україні» (затверджені наказом МВС від 30.12.2014 № 1417) [3], комп’ютерних класів — пункт 3 розділу VIII «Правил пожежної безпеки для навчальних закладів та установ системи освіти України» (затверджені наказом МОН від 15.08.2016 № 974). [4] та інші державні стандарти, що регламентують експлуатування комп’ютерної техніки як радіоелектронної апаратури.

8.2 Шкідливі і небезпечні фактори при роботі з комп’ютером

Можна виділити наступні основні фактори, що впливають на стан здоров'я людей, які працюють за комп'ютером:

- сидяче положення на протязі тривалого періоду;
- вплив електромагнітного випромінювання монітора;
- втома очей, навантаження на зір;
- перевантаження суглобів кистей;
- стрес при втраті інформації.

У кожному з цих випадків ступінь ризику прямо пропорційний часу, що проводиться за комп’ютером і поблизу нього.В сучасних умовах взаємодія людини з технікою значно ускладнилась, що вимагає комплексного підходу,

який передбачає розгляд людини, технічних засобів праці та виробничого середовища, як взаємозв'язаних елементів єдиної системи. Все вищесказане в повній мірі відноситься й до системи «людина–комп'ютер–середовище».

Вагомий вплив на працездатність та здоров’я користувачів комп’ютерів

здійснює виробниче середовище. Це середовище у виробничих приміщеннях

(офісах), в основному, визначається мікрокліматом, освітленням, наявністю

шкідливих речовин у повітрі, рівнем шуму, випромінювання.

Для того, щоб об’єктивно проаналізувати відповідність умов праці діючим нормативно-правовим актам та запропонувати заходи щодо зменшення негативного впливу комп’ютера на організм людини необхідно здійснити санітарно-гігієнічну характеристику умов працівника, який працює з програмним продуктом.

8.3 Аналіз санітарно-гігієнічних умов праці на робочому місці користувача ПК

Розглянемо приміщення в якому працює користувач ПК з даним програмним продуктом.

Приміщення має одностороннє природне освітлення і загальне штучне освітлення. Стіни і стеля обклеєні світлими шпалерами, підлога вкрита темним ламінатом. У приміщенні відсутні сильні вібрації та шкідливі речовини. Склад повітря в нормі. У кімнаті знаходиться ПК з 4-ядерним процесором і 23-дюймовим IPS монітором, а також меблі.

Приміщення має довжину 4м, ширину 3,5м, висоту стелі 2,7м. Кількість робочих місць–одне. Площа–14м^2, об'єм–37,8м^3. Виходячи з цього, отримано дані, наведені в таблиці 8.1.

Таблиця 8.1–Фактичні та нормативні значення параметрів приміщення

Параметр	Норма *	Реальні параметри
Площа, S	не менше 6 м^2	14 м^2
Об'єм, V	не менше 20 м^3	37,8 м^3

*Згідно ДСанПіН 3.3.2.007-98 (Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин).

За даними, які наведено у табл. 8.1, можна зробити висновок, що отримані показники, площа та об'єм приміщення у розрахунку на одно робоче місце користувача ПК відповідає чинним нормам і вимогам.

Щодо мікроклімату, то згідно з ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» [5] роботу з ПК можна віднести до категорії легка 1а. Джерелами тепла в цьому приміщенні є люди, електроустаткування, освітлювальні прилади в темний час доби і система опалювання взимку. Оператором виділяється до 120ккал теплової енергії за годину. Оптимальні та фактичні значення параметрівмікроклімату приведені в таблиці 8.2.

Таблиця 8.2 – Значення параметрів мікроклімату

Період року	Параметр	Оптимальний*	Фактичний
Теплий	Температура	23 – 25 ^{0} С	24^0 С
	Вологість	40 – 60%	50%
	Швидкість повітря	< 0,1м/с	
Холодний	Температура	22 - 24^0 С	23^0 С
	Вологість	40 – 60%	55%
	Швидкість повітря	< 0,1м/с	

*ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень»

По отриманим замірам параметрів мікроклімату можна зробити висновок, що усі показники задовольняють вимогам зазначеним для робіт категорії легка 1а і є задовільними для здоров'я людини.

Щодо освітлення , то згідно з ДБН В.2.5-28:2006 «Природне і штучне освітлення» [6] ця робота відноситься до Va розряду зорових робіт. Передбачається використання природного, штучного і змішаного освітлення.

Природне освітлення здійснюється за допомогою вікна, площа якого складає $S' = 1,8*1,5 = 2,7м^2$ і є боковим освітленням. У світильниках місцевого і загального освітлення використовуються світлодіодні лампи потужністю 20Вт із світловим потоком лампи 900 лм. Згідно замірів рівень освітлення в даному приміщенні і на робочому місці складає в межах 350 -500 лк, що відповідає нормованому значенню

Джерелом шуму в приміщенні є комп'ютер. Вентилятори (кулери) системного блоку, процесора, відеокарти і блоку живлення є сучасними і мають низький рівень шуму. Згідно з технічною документацією шум, зумовлений кулером в блоці живлення складає 25 дБ, кулером процесора - 30 дБ, загальний -34 дБ. Враховуючи незначний рівень шуму від персонального комп'ютера і незначний рівень фонового шуму від іншого устаткування, можна стверджувати, що сумарний рівень шумового забруднення приміщення не перевищує максимально допустимий рівень коригованої звукової потужності і складає не більше 50 дБА, що відповідає рівню шуму для приміщень з комп’ютерною технікою згідно Державних санітарних правил і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98.

У приміщенні відсутні джерела інфрачервоного, ультрафіолетового і електромагнітного випромінювання, бо монітор ПК вироблений на основі

рідкокристалічної матриці, підсвітка якої здійснюється неоновою лампою, що не має сильного електромагнітного випромінювання і сертифіковані в Україні.

Блок живлення є екранованим і не випускає вищезазначених видів випромінювання.

8.4 Розробка заходів з умов поліпшення охорони праці

Перерахуємо проведені заходи щодо забезпечення умов праці на робочому місці користувача ПК.

З точки зору забезпечення електробезпеки до цих заходів можна віднести: устаткування розподільних щитів спеціальними розетками з заземлюючими контактами; організація заземлення всіх приладів і пристроїв; періодична перевірка всіх приладів і пристроїв; щорічна здача іспитів з охорони праці.

З точки зору забезпечення оптимальних умов мікроклімату, рівня звуку і освітленості до цих заходів можна віднести: організацію природної вентиляції, за допомогою дефлектора, для забезпечення необхідного повітрообміну в приміщенні вузла; організацію системи центрального опалювання, для підтримки оптимальної температури в холодний період року; організацію штучного загального освітлення, для забезпечення необхідних умов зорової роботи, що відповідають, оформлення паспорта на приміщення вузла, з занесенням в нього вимірювань освітленості і рівня звуку, проведених відділом охорони праці.

Крім рекомендацій щодо конкретного приміщення, де було проведено дослідження умов праці, існують загальні вимоги, які зарекомендовані відповідними нормативними документами.

Правильна організація робочих місць запобігає передчасній втомлюваності користувача і сприяє збереженню здоров'я. Організація робочого місця передбачає:

- правильне розміщення робочого місця у виробничому приміщенні;

- вибір ергономічного обґрунтованого робочого положення, виробничих меблів з урахуванням характеристик людини;.

- раціональне компонування обладнання на робочих місцях;

- урахування характеру й особливостей трудової діяльності. Стосовно робочих місць користувача ВДТ, то організація робочого має забезпечуватися відповідно до ДСанПіН 3.3.2-007-98. Для запобігання перевтомленню необхідно виконувати вправи для очей та дотримуватись розпорядку роботи та відпочинку. На робочому місці реалізовувався режим відпочинку: кожні дві години – перерва для виконання фізичних вправ для м‘язів очей.

8.5 Протипожежний захист

Пожежі в приміщеннях з оргтехнікою становлять особливу небезпеку, бо поєднані з великими матеріальними збитками. Пожежа може виникнути при взаємодії горючих речовин і джерел запалювання. Горючими речовинами є будівельні та опоряджувальні матеріали, пластмасові корпуси техніки, шнури тощо. Джерелами запалювання можуть бути електронні схеми комп'ютерів, принтерів, пристроїв електроживлення, де внаслідок різних порушень виникає перегрівання елементів, утворюються електричні іскри та дуги, здатні спричинити займання горючих матеріалів.

При обслуговуванні, ремонтних та профілактичних роботах використовуються різні легкозаймисті рідини, прокладаються тимчасові електропровідники, здійснюється паяння. Виникає додаткова пожежна небезпека, яка потребує відповідних заходів пожежного захисту. До засобів гасіння пожежі, призначених для локалізації невеликих займань, належать вогнегасники, сухий пісок, азбестові ковдри. Приміщення, в який встановлено комп'ютери і де немає необхідності влаштування систем автоматичного пожежогасіння, необхідно оснащувати переносними вуглекислотними з

розрахунку 2 шт. на кожні 20 м^2 в приміщеннях. Звуковбирне облицювання стін, стель приміщень треба виконувати з негорючих та важко горючих матеріалів.

З метою виявлення початкової стадії займання необхідно використовувати пристрої систем автоматичного пожежогасіння там, де цього вимагають Правила пожежної безпеки.

З точки зору забезпечення пожежної безпеки до цих заходів можна віднести наявність схеми евакуації з приміщення вузла, у випадку пожежі, повішену на вхідні двері.

8.6 Розрахункова частина

В приміщенні (де відсутні джерела виділення шкідливих речовин) працює одна людина. Робота пов'язана з використанням ПЕОМ. Розміри приміщення: А = 4 м, В = 3,5 м, Н = 2,8 м, устаткування займає 15% об'єму. Визначити найменшу необхідну кількість повітря для вентиляції.

Для приміщень, в яких відсутні виділення шкідливих речовин у повітрі, розрахунок вентиляції здійснюється залежно від кількості працюючих.

Необхідна кількість повітря (м^3 /год.), яка забезпечує відповідність параметрів повітря робочої зони нормованим значенням, визначається за наступною формулою:

$$L = L' \cdot N, \quad (8.1)$$

де L' - нормативна кількість повітря на одного працюючого, яка залежить від питомого об'єму приміщення, м^3 /(год.-люд.);

N - кількість працюючих.

Питомий об'єм приміщення $V_п$, (м^3 /люд.), визначається за формулою:

$$V_п = V/N, \quad (8.2)$$

де V - об'єм приміщення, м^3.

Визначаємо вільний об'єм приміщення

$V = A \cdot B \cdot H \cdot 0,85 = 4 \cdot 3,5 \cdot 2,8 \cdot 0,85 = 33,3$ м^3.

Питомий вільний об’єм складає

$V' = V / N = 33,2 / 1 = 33,2$ м^3 /люд. < 20 м^3/люд.

Нормована кількість повітря на одну людину при $V' < 20$ м^3 /люд. становить 30 м^3 /(год. • люд.).

Висновки до розділу

У даному розділі магістерської роботи проведено аналіз умов працівника робота якого пов’язана з комп’ютерною технікою. Проведено аналіз основних санітарно – гігієнічних показників в заданому приміщені , де працівник зайнятий постійною роботою за комп’ютером.. Створені умови повинні забезпечувати комфортну роботу. На підставі вивченої літератури з даної проблеми, були зазначені оптимальні параметри мікроклімату, освітлення, допустимі рівні шуму та іонізуючого випромінювання при роботі з ПЕОМ, а також розраховано найменшу необхідну кількість повітря для вентиляції.

Дотримання умов, що визначають оптимальну організацію робочих місць працівників, дозволить зберегти гарну працездатність протягом усього робочого дня, підвищить як в кількісному, так і в якісному відносинах продуктивність їх праці.

Список використаних джерел інформації

1. Наказ Міністерства соціальної політики України 14.02.2018 № 207 «Про затвердження Вимог щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями». - *Режим доступу до ресурсу:* https://zakon.rada.gov.ua/laws/show/z0508 (дата звернення 19.10.22).

2. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин: ДСанПІН

3.3.2-007-98. - *Режим доступу до ресурсу:* https://zakon.rada.gov.ua/rada/show/v0007282-98.

3. Наказ Міністерства внутрішніх справ України 30.12.2014 №1417 «Про затвердження Правил пожежної безпеки України» - *Режим доступу до ресурсу* https://zakon.rada.gov.ua/laws/show/z0252-15#Text (дата звернення 19.10.22).

4. Наказ Міністерства освіти і науки України 15.08.2016 № 974 «Про затвердження Правил пожежної безпеки для навчальних закладів та установ системи освіти України» - Режим доступу до ресурсу https://zakon.rada.gov.ua/laws/show/z1229-16#Text (дата звернення 19.10.22).

5. Постанова № 42 від 01.12.1999 Головного державного санітарного лікаря України «Санітарні норми мікроклімату виробничих приміщень ДСН 3.3.6.042-99 - Режим доступу до ресурсу: https://zakon.rada.gov.ua/rada/show/va042282-99 (дата звернення 19.10.22).

6. Державні будівельні норми України: ДБН В.2.5-28:2018. - *Режим доступу до ресурсу: https://goo.su/9AkQ* (дата звернення 19.10.22).

7. Методичні рекомендації до виконання розділу "Заходи з охорони праці та техніки безпеки" випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти для здобувачів вищої освіти спеціальностей 123 "Комп'ютерна інженерія" та 122 "Комп'ютерні науки" / М-во освіти і науки України, Центральноукраїн. нац. техн. ун-т, каф. кібербезпеки та програм. забезпечення; [укл. О.В. Оришака, К.М. Марченко]. - Кропивницький: ЦНТУ, 2022. — 19 с. [Електронний ресурс]. – Режим доступу : http://dspace.kntu.kr.ua/jspui/handle/123456789/12240 (дата звернення 19.10.22).

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи хмарного керування пристроями комплексу рішень «розумний дім».

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів управління розумним будинком з підсистемою безпеки передачі даних.

Рішення даного завдання полягало у вирішенні наступних задач:

- Був проведений огляд існуючих систем хмарного керування управління розумним будинком.

- Досліджена система хмарного керування розумним будинком.

- На основі отриманих результатів досліджень створена програмна реалізація системи хмарного керування розумним будинком.

Розроблені під час виконання магістерської роботи алгоритми дозволяють успішно вирішувати завдання хмарного керування системою «Розумний дім».

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудовано алгоритм і обрано середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створені програмного забезпечення було використано об’єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня C++ та PHP. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 8/10 та системи Android.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати змішану систему шифрування що використовує Base64, AES, RSA.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Alex Matrosov, Eugene Rodionov, Sergey Bratus. Rootkits and Bootkits. No Starch Press. 2019. 450 p.

2. Петрик В.М., Присяжнюк М.М., Аль-Файюмі Халед та ін. «Системи інформаційної зброї та технології інформаційної війни»: підручник / Петрик В.М., Присяжнюк М.М., Аль-Файюмі Халед, Жарков Я.М., Смірнов О.А, Буравченко К.О., Давидюк А.В., Кононович В.Г., Корчинский В.В., Кудирко В.М., Фесенко А.О.; за заг. ред. В.М. Петрика, М.М. Присяжнюка.- К.: Видавничий центр “Кафедра”, 2025.- 320 с.

3. Усік, П.С., Смірнова, Т.В., Буравченко, К.О., Смірнов, О.А., Улічев, О.С., Смірнов, С.А. «Дослідження технологій забезпечення кібербезпеки банківських систем з використанням штучного інтелекту». *Кібербезпека: освіта, наука, техніка.* 2025. Том 1 № 29. С.704-716, 2025

4. Kuznetsov, O., Frontoni, E., Kuznetsova, K., Arnesano, M., Smirnov, O. «A secure biometric authentication architecture for blockchain-driven cyber-physical systems». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 193-224.

5. Kuznetsov, O., Atzeni, G., Arnesano, M., Randieri, C., Smirnov, O. «Secure IoT-based smart wheelchair system: From implementation to security enhancement strategy». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 225-257.

6. Kuznetsov, O., Smirnov, O., Kuznetsova, T., Shaikhanova, A., Svatowsky, I. «Privacy-utility trade-offs in IoT networks: A comparative analysis of differential privacy mechanisms for sensor data aggregation». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 589-622.

7. Lakhno, V., Malyukov, V., Smirnov, O., Bebeshko, B., Chubaievskiy, V.,

Zhumadilova, M., Malyukova, I., Smirnov, S. «Multifactorial Model for Targeted Attacks Counteracting Within the Framework of a Multi-Step Quality Game with Fuzzy Information». *8th International Symposium on Intelligent Informatics, ISI 2023*, 2025. vol 389. pp 377-389. Springer, Singapore.

8. Kuznetsov O., Frontoni E., Kuznetsova Y., Smirnov O., Moskovchenko I. «Trust-Based Security Architecture for Edge Computing: A Simulation Study of Dynamic Trust Evolution and Attack Detection». *CEUR Workshop Proceedings,* 2024, 3909, pp. 227-241.

9. Kuznetsov, O., Frontoni, E., Kryvinska, N., Chevardin, V., Smirnov, O. «Wireless Network Encryption Stream Ciphers, Computational Modeling, and Security Analysis». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 379-402.

10. Kuznetsov, O., Frontoni, E., Kryvinska, N., Smirnov, O., Imoize, G.L. «Computational Modeling of Enhanced Spread Spectrum Codes for Asynchronous Wireless Communication». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 403-447.

11. Ткаченко, О., Ільєнко, А., Улічев, О., Мелешко, Є., Смірнов, О. «Правові засади поширення інформаційних впливів в соціальних мережах». *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»,* 2024. № 2(26), С. 170-188.

12. Смірнова Т.В., Конопліцька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка.* 2024. №4(24), С. 6-27.

13. Вінтенко, Б., Миронець, І., Смірнов, О., Кравчук, О., Козірова, Н., Савеленко, Г., Коваленко, А. «Дослідження вимог та аналіз кібербезпеки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Кібербезпека: освіта, наука, техніка.* 2024. №3(23), С. 111-131.

14. Батрак О., Смірнова Т., Гнатюк В., Одарченко Р., Смірнов О.

«Дослідження показників ефективності функціонування та перспектив розвитку систем IP-телефонії». *Підводні технології,* 2024, № 13, с. 28-35.

15. Kuznetsov, O., Kryvinska, N., Ilchenko, O., Smirnova, T., Ulianovska, Y. «Comparative Analysis of Cryptocurrency Trading Platforms Using the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, 2023, 3628, pp. 106-115.

16. Akhalaia, G., Iavich, M., Iashvili, G., Prysiazhnyy, D., Smirnova, T. «Secure Encrypted Connection on Georgian Website». *CEUR Workshop Proceedings,* 2023, 3550, pp. 313-320.

17. Al-Mudhafar Aqeel, A.M., Smirnova, T., Buravchenko, K., Smirnov, O. «The method of assessing and improving the user experience of subscribers in software-configured networks based on the use of machine learning». *Advanced Information Systems*, 2023, 7(2), pp. 49-56

18. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings,* Volume 3530, 2023, pp. 256-265.

19. Kuznetsov, O., Kandiy, S., Frontoni, E., Smirnov, O. «Trade-offs in Post-Quantum Cryptography: A Comparative Assessment of BIKE, HQC, and Classic McEliece». *CEUR Workshop Proceedings,* Volume 3504, 2023, pp. 1-11.

20. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskyi, V., Khorolska, K., Bebeshko, B. «Selection of a Rational Composition of İnformation Protection Means Using a Genetic Algorithm». *In: Rajakumar, G., Du, KL., Vuppalapati, C., Beligiannis, G.N. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. Lecture Notes on Data Engineering and Communications Technologies*, vol 131. 2023. Springer, Singapore. pp. 21-34.

21. Смірнова Т.В., Гнатюк С.О., Бердибаєв Р.Ш., Сидоренко В.М., Жигаревич О.К., «Система корелювання подій та управління інцидентами кібербезпеки на об’єктах критичної інфраструктури». *Кібербезпека: освіта, наука, техніка*, №*3*(19), 2023, С. 176-196.

22. Смірнов О.А. Козлов Я.О., Смірнова Т.В. «Дослідження застосування SIEM-систем для забезпечення кібербезпеки та захисту інформації». *II Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи hозвитку інформаційних технологій (ІПШРІТ-2023)»* м.Черкаси 6 грудня 2023 року - Черкаси: ЧДТУ.- 2023. - С.251-252.

23. Козлов Я.О., Смірнова Т.В., Смірнов О.А. «Дослідження SIEM-систем для забезпечення кібербезпеки». *VII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології" до 30-ти річчя кафедри кібербезпеки та програмного забезпечення,* м. Кропивницький. 1 листопада 2023 р. - Кропивницький: ЦНТУ. - 2023. - С. 26.

24. Козлов Я.О., Козірова Н.Л., Смірнов О.А. «Дослідження структури та принципу роботи SIEM-системи». *VII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології" до 30-ти річчя кафедри кібербезпеки та програмного забезпечення,* м. Кропивницький. 1 листопада 2023 р. - Кропивницький: ЦНТУ. - 2023. - С. 59.

25. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А., Коваленко А.С. «Дослідження нормативних документів та галузевих стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС, важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 2(72), С. 170-178.

26. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» *CEUR Workshop Proceedings,* Volume 3187, 2022,

27. Smirnov O.A., Al-Oraiqat A.M., Ulichev O.S., Meleshko Ye.V., Al-Rawashdeh H.S., Polishchuk L.I. «Modeling strategies for information influence dissemination in social networks». *Journal of Ambient Intelligence and Humanized Computing* Volume 13, Issue 5. Springer, Cham. 2022, pp. 2463-2477.

28. Смірнова Т.В., Гнатюк С.О., Сидоренко В.М., Юдін О.Ю., Сидоренко С.Ю., «Модель визначення критичності галузевих інформаційно-телекомунікаційних систем». *Проблеми інформатизації та управління,* № 2(70). 2022. С. 28-37.

29. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку,* 2022, № 3(69). С. 93-98.

30. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»,* № 2 (307). С. 46-52. 2022.

31. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.

32. Smirnov O., Kuznetsov A., Zhora V., Onikiychuk A., Pieshkova O. «Hiding Messages in Audio Files Using Direct Spread Spectrum». *11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021,* Cracow, Poland, 22-25 September 2021. P. 414-418

33. Smirnov O., Kuznetsov A., Lokotkova I., Kuznetsova T., Florov S., Lebid O. «Using Orthogonal Signals to Hide Information in Images». *4 IEEE International Conference on Advanced Information and Communication Technologies (AICT) - 2021*, Lviv, Ukraine, September 21-25, 2021. P. 255-260.

34. Smirnov O., Kuznetsov A., Girzheva O., Kiian A., Nakisko O., Kuznetsova T. «Advanced Code-Based Electronic Digital Signature Scheme». *2020 IEEE International Conference on Problems of Infocommunications Science and Technology, PIC S and T 2020,* Kharkiv, 6 October 2020-9 October 2020, P. 358-362.

35. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova K. «Data hiding scheme based on spread sequence addressing». *CEUR Workshop Proceedings* Volume 2805, 2020, Pages *44-58.*

36. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». *International Journal of Computing*; 2020, Volume 19, Issue 2 - Research Institute for Intelligent Computer Systems - 2020. - P. 247-256.

37. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.

38. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.

39. Smirnov O., Kuznetsov A., Arischenko A., Chepurko I., Onikiychuk A., Kuznetsova T. «Pseudorandom sequences for spread spectrum image steganography». *CEUR Workshop Proceedings* Volume 2654, 2020, Pages 122-131.

40. Smirnov O., Kuznetsov A., Kovalchuk D., Kuznetsova T. «New technique for data hiding in cover images using adaptively generated pseudorandom sequences». *CEUR Workshop Proceedings* Volume 2654, 2020, Pages 1-14.

41. Smirnov O., Lutsenko M., Kuznetsov A., Kiian A., Kuznetsova T., «Biometric cryptosystems: overview, state-of-the-art and perspective directions». *Lecture Notes in Networks and Systems*, vol 152. Springer, Cham. 2021, pp 66-84.

42. Smirnov O., Kuznetsov A., Onikiychuk A., Makushenko T.,

Anisimova O., Arischenko A. «Adaptive pseudo-random sequence generation for spread spectrum image steganography». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT)*, Ukraine, Kyiv, May 14-18. 2020. P. 161-165.

43. Smirnov O., Kuznetsov A., Kiian A., Babenko V., Perevozova I.,Chepurko I. «New Approach to the Implementation of Post-Quantum Digital Signature Scheme». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT),* Ukraine, Kyiv, May 14-18. 2020. P. 166-171.

44. Smirnov O., Kuznetsov A., Kiian A., Cherep A., Kanabekova M., Chepurko I. «Testing of code-based pseudorandom number generators for post-quantum application». *2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT),* Ukraine, Kyiv, May 14-18. 2020. P. 172-177.

45. Smirnov O., Kuznetsov A., Pushkar'ov A., Serhiienko R., Babenko V., Kuznetsova T., «Representation of Cascade Codes in the Frequency Domain». In: Radivilova T., Ageyev D., Kryvinska N. (eds) Data-Centric Business and Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 48. Springer, Cham. 2021. pp 557-587.

46. Smirnov, O., Markovets, O. Vovk, N., Turchyn, Y., «Model of informational support for social network administrators' content creation». *CEUR Workshop Proceedings* Volume 2616, 2020, Pages 125-136.

47. Smirnov, O., Shekhanin, K., Kuznetsov, A., Krasnobayev, V. «Detecting Hidden Information in FAT». *International Journal of Computer Network and Information Security (IJCNIS)*. Vol. 12, No. 3, 2020. PP.33-43.

48. Smirnov, O., Kuznetsov, A., Gorbacheva, L., Babenko, V., «Hiding data in images using a pseudo-random sequence», *CEUR Workshop Proceedings* Volume 2608, 2020, Pages 646-660.

49. Smirnov, O., Kuznetsov, A., Kolovanova, I., Kuznetsova, T., «Noise

immunity of the algebraic geometric codes». International Journal of Computing; 2019, Volume 18, Issue 4 - Research Institute for Intelligent Computer Systems - 2019. - P. 393-407.

50. Smirnov, O., Kuznetsov, A., Reshetniak, O., Ivko, N., Katkova, T., Kuznetsova, T., «Generators of Pseudorandom Sequence with Multilevel Function of Correlation». *2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T),* Kyiv, Ukraine, 8 - 11 October 2019 . P.517-522.

51. Smirnov, O., Ulichev, O., Meleshko, Y., Khokh, V., Goncharenko, I. «Method of Choosing Objects for Informational Influence in Social Networks during Information Campaign Based on the Analytic Hierarchy Process». *CEUR Workshop Proceedings*, Vol 2588, P. 215-227, 2019.

52. Smirnov, O., Krasnobayev, V., Yanko, A., Kuznetsova, T. «Methods of nulling numbers in the system of residual classes». *CEUR Workshop Proceedings*, Vol 2588, P. 90-106, 2019.

53. Smirnov, O., Kuznetsov, A., Kiian, A., Gorbenko, Y., Cherep, O., Bexhter L. «Code-based Pseudorandom Generator for the Post-Quantum Period», *2019 IEEE International Conference on Advanced Trends in Information Theory (IEEE ATIT 2019).* 18.12.19-20.12.19 Kyiv Ukraine. P. 204 - 209.

54. Smirnov, O., Kuznetsov, A., Nariezhnii, O., Stelnyk, S., Kokhanovska, T., Kuznetsova T., «Side Channel Attack on a Quantum Random Number Generator», *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019*; Metz; France; 18 -21 September 2019. P.713-718.

55. Kuznetsova, T., «Code-Based Schemes for Post-Quantum Digital Signatures», *10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2019;* Metz; France; 18-21 September 2019. P. 707-712.