Хмельницький національний університет
Факультет інформаційних технологій
Кафедра кібербезпеки

КВАЛІФІКАЦІЙНА РОБОТА

Когута Артура Віталійовича

на здобуття ступеня вищої освіти Бакалавра

Система контролювання доступу на основі сервісу Blynk IoT

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека

Освітня програма Кібербезпека

КРБКБ.2102148.21.02.27 ПЗ

Виконав студент 4 курсу, група КБ-21-2 02.06.25 Артур КОГУТ
Підпис, дата Ініціали, прізвище

Керівник канд. тех. наук, доцент 04.06.25 Володимир ПЕТРУШАК
Науковий ступінь, вчене звання Підпис, дата Ініціали, прізвище

Нормоконтролер старший викладач 09.06.25 Сергій МОСТОВИЙ
Науковий ступінь, вчене звання Підпис, дата Ініціали, прізвище

До захисту допускаю:

Зав. кафедри кібербезпеки _____ Юрій КЛЬОЦ
Підпис, дата Ініціали, прізвище

09 06 2025р.

Хмельницький, 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Кібербезпеки

Рівень вищої освіти Бакалавр

Галузь знань 12 – Інформаційні технології

Спеціальність 125 – Кібербезпека

Освітня програма Кібербезпека

ЗАТВЕРДЖУЮ
Завідувач кафедри кібербезпеки
Юрій КЛЬОЦ
15 лютого 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
Когут Артур Віталійович

1 Тема роботи Система контролювання доступу на основі сервісу Blynk IoT

Керівник роботи канд. тех. наук, доц. кафедри КБ Володимир Степанович Петрушак

Затверджено наказом ректора університету від 07 лютого 2025 № 23

2 Строк подання студентом кваліфікаційної роботи на кафедру 06.06.2025

3 Вихідні дані до роботи розробити систему доступу на основі сервісу Blynk IoT

4 Зміст пояснювальної записки (перелік питань, які потрібно розробити) Вступ. Теоретичні основи побудови IoT-систем доступу. Розробка системи допуску на базі Blynk IoT. Тестування системи та результати розробки. Висновки.

5 Перелік графічного матеріалу (із зазначенням обов'язкових креслень) Схема апаратного забезпечення. Схема роботи алгоритму. Blyk Dashboard. Сценарій тестування.

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Система контролювання доступу на основі сервісу Blynk IoT».

Автор роботи: студент групи КБ-21-2 Когут Артур Віталійович.

Керівник роботи: канд. тех. наук, доцент кафедри кібербезпеки Петрушак Володимир Степанович.

Пояснювальна записка: 65 с., 16 рис., 40 джерел, 2 додатки.

Графічна частина: 4 плакати.

СИСТЕМА ДОПУСКУ, BLYNK, IOT, КОНТРОЛЕР, СЕРВІС, ПЛАТФОРМА, ДОСТУП

Кваліфікаційна робота бакалавра «Система допуску на основі сервісу Blynk IoT» присвячена розробці та реалізації системи контролю доступу з використанням платформи Інтернету речей Blynk.

У роботі розглянуті принципи функціонування сервісу Blynk, включаючи його архітектуру та можливості для взаємодії з мікроконтролерами (зокрема ESP8266). Описано практичну реалізацію системи допуску, що, ймовірно, включає апаратну частину (мікроконтролер, сенсори, виконавчі механізми, такі як реле або електрозамок) та програмне забезпечення для взаємодії з Blynk. Робота, можливо, деталізує процес налаштування Blynk-додатку, програмування мікроконтролера та алгоритми керування доступом. Результати роботи демонструють ефективність використання IoT-технологій для створення гнучких та керованих систем безпеки.

02.06.2025 _____ _____

ABSTRACT

The topic of the qualification work: «Access control system based on the Blynk IoT service».

Author of the work: student of the CS-21-2 group Artur Kogut.

Supervisor: Ph.D. tech. Sciences, Associate Professor of the Department of Cybersecurity Volodymyr Petrushak.

Explanatory note: 65 p., 16 figures, 40 sources, 2 appendices.

Graphic part: 4 posters.

ACCESS SYSTEM, BLYNK, IOT, CONTROLLER, SERVICE, PLATFORM, ACCESS

The bachelor's qualification work «Access control system based on the Blynk IoT service» is devoted to the development and implementation of an access control system using the Blynk Internet of Things platform.

The paper discusses the principles of the Blynk service, including its architecture and capabilities for interacting with microcontrollers (in particular, ESP8266). The paper describes the practical implementation of an access control system, which probably includes hardware (microcontroller, sensors, actuators such as relays or electric locks) and software for interacting with Blynk. The work probably details the process of setting up the Blynk application, programming the microcontroller, and access control algorithms. The results of the work demonstrate the effectiveness of using IoT technologies to create flexible and manageable security systems.

02.06.2025 

ЗМІСТ

ВСТУП

У сучасному світі зростаюча залежність від технологій Інтернету речей (далі - IoT) відкриває нові можливості для автоматизації різних процесів у повсякденному житті. IoT-системи дозволяють з'єднувати фізичні об'єкти з Інтернетом, забезпечуючи збирання та обробку даних у реальному часі. Однією з ключових галузей застосування IoT є забезпечення безпеки, зокрема, система контролювання доступу, яка відіграє важливу роль у захисті ресурсів, інформації та людей.

Система контролювання доступу на основі сервісу Blynk IoT є інноваційним рішенням, яке використовує можливості платформи Blynk для інтеграції різних пристроїв та сервісів у єдину екосистему. Blynk – це потужна платформа для розробки IoT-додатків, що дозволяє користувачам легко створювати та управляти своїми проектами, не вдаючись до глибоких технічних знань. Використання Blynk IoT для розробки системи контролювання доступу надає ряд переваг, таких як спрощене налаштування, доступність даних у реальному часі та можливість віддаленого управління.

Мета даної кваліфікаційної роботи полягає в розробці та впровадженні системи контролювання доступу на основі Blynk IoT, яка забезпечить надійний та зручний спосіб управління доступом до об'єктів. У рамках роботи будуть розглянуті основні компоненти системи, принципи її функціонування, а також алгоритми роботи з даними. Особлива увага буде приділена аспектам безпеки, оскільки надійний контроль доступу є критично важливим для захисту як фізичних, так і інформаційних активів.

У процесі реалізації проекту планується провести аналіз існуючих рішень у сфері контролю доступу, вивчити їх переваги та недоліки, а також визначити, як інтеграція з платформою Blynk може покращити функціональність та зручність використання. Окрім цього, буде проведено експериментальне дослідження, метою якого є тестування розробленої системи в реальних умовах, а також оцінка її ефективності та зручності для користувачів.

У рамках цієї кваліфікаційної роботи буде також розглянуто питання

інтеграції системи контролювання доступу з іншими IoT-пристроями та сервісами. Це включає можливості автоматизації, моніторингу та управління доступом через мобільні додатки, а також реалізацію сценаріїв, які дозволяють користувачам швидко реагувати на зміни в ситуації. Наприклад, система може бути налаштована на автоматичне сповіщення користувача у випадку несанкціонованого доступу або збоїв у роботі пристроїв. Такий підхід не лише підвищує рівень безпеки, але й робить систему більш гнучкою та адаптивною до потреб користувачів. В результаті, реалізація системи контролювання доступу на основі Blynk IoT забезпечить не лише зручність, а й високу ефективність у забезпеченні безпеки об'єктів. Це стане важливим кроком у розвитку технологій, що сприяють інтеграції фізичних і цифрових світів, роблячи наше життя безпечнішим та комфортнішим.

Таким чином, дана робота не лише сприятиме глибшому розумінню можливостей Blynk IoT у сфері контролювання доступу, але й надасть практичні рекомендації для подальшої розробки та впровадження IoT-рішень у цій важливій галузі.

1 ТЕОРЕТИЧНІ ОСНОВИ ПОБУДОВИ ІОТ-СИСТЕМ ДОСТУПУ

1.1 Поняття та класифікація систем контролю доступу

Системи контролю та управління доступом (далі - СКУД) є невід'ємним елементом сучасної безпеки об'єктів різного призначення — від комерційних офісів до промислових підприємств. Це комплекс апаратних та програмних засобів, які забезпечують обмеження доступу до певних зон або територій, а також фіксацію подій, пов'язаних з входом і виходом людей чи транспорту через контрольовані точки, такі як двері, турнікети, шлагбауми або ворота [1].

Основною метою впровадження таких систем є забезпечення безпеки: СКУД захищає від несанкціонованого проникнення, допомагає запобігти крадіжкам або пошкодженню майна, а також знижує ризик перебування на об'єкті сторонніх осіб. Для ефективної реалізації прохідної системи доцільно застосовувати турнікети, які здатні контролювати потік осіб, пропускаючи лише одну людину за один раз. Турнікети сучасного зразка часто мають функцію «антипаніка», що дозволяє швидко залишити приміщення у випадку надзвичайної ситуації. Не менш важливою є правильна організація робочого місця охоронця — воно має бути відгороджене прозорим бар'єром, щоб забезпечити безпечну взаємодію з відвідувачами [2].

Крім функцій фізичної безпеки, СКУД часто використовуються для ведення обліку робочого часу працівників. Це дозволяє не лише фіксувати час приходу та відходу співробітників, а й автоматизувати формування звітності. Сучасне програмне забезпечення, наприклад від компанії ZKTeco, дозволяє відстежувати відвідуваність у режимі реального часу та будувати аналітичні звіти, що є надзвичайно корисним для відділів кадрів. Ідентифікація може здійснюватися за допомогою різноманітних носіїв: безконтактних карток, брелків, відбитків пальців, сканування обличчя чи венозного малюнка долоні. Обрання того чи іншого методу залежить від вимог до точності, зручності та рівня безпеки [2].

Ефективне використання СКУД також сприяє оптимізації фінансових витрат підприємства. Автоматизовані системи замінюють ручні процеси, що дозволяє зменшити кількість персоналу, задіяного у видачі пропусків або розрахунку

заробітної плати. Наприклад, інтеграція СКУД із бухгалтерськими чи HR-системами дозволяє значно спростити розрахунок заробітної плати на основі фактично відпрацьованого часу [3].

Ще однією перевагою є можливість використання даних СКУД у бізнес-аналітиці. Завдяки інтеграції з іншими системами — документообігу, CRM, бухгалтерії чи управління персоналом — керівництво може оперативно отримувати аналітичну інформацію щодо відвідуваності, ефективності роботи персоналу, завантаженості зон об’єкта тощо. Це відкриває широкі можливості для підвищення продуктивності та управлінської ефективності [3]. На рисунку 1.1 зображено приклад системи контролю доступу та з яких компонентів вона може складатися.

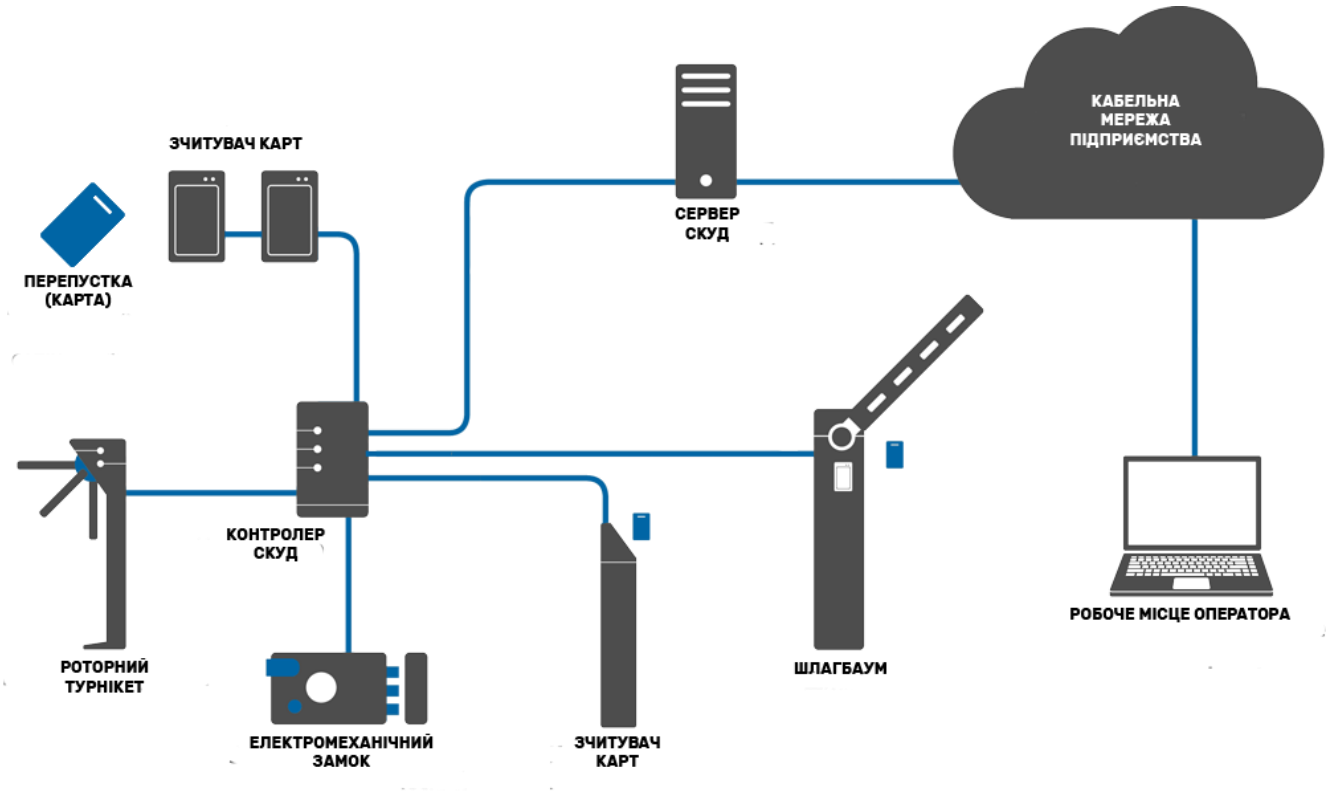


Рисунок 1.1 – Приклад системи контролю доступу

Конструктивно будь-яка СКУД складається з ряду ключових елементів. Перше — це ідентифікатори, які можуть мати вигляд RFID-карток, брелків, міток або біометричних ознак. Далі — зчитувачі, які розпізнають ці ідентифікатори та передають інформацію на контролери. Зчитувачі відрізняються за типом даних, які

вони можуть обробляти (наприклад, EM-Marine, Mifare, UHF, відбитки пальців, геометрія обличчя), і за способом підключення до системи [4].

Контролери — це центральна частина СКУД, вони приймають рішення про надання доступу, зберігають базу користувачів і реєструють події. Вони можуть бути автономними або мережевими. Автономні працюють ізольовано, налаштовуються окремо і не потребують підключення до централізованого ПЗ. Натомість мережеві контролери дозволяють будувати гнучкі та масштабовані системи з централізованим управлінням та моніторингом у реальному часі [4].

Виконавчі механізми — такі як електрозамки, турнікети, шлагбауми — забезпечують фізичне блокування доступу. Вони поділяються на електромеханічні та електромагнітні пристрої, кожен з яких має свої особливості застосування та рівень захищеності.

У сукупності всі ці компоненти формують єдину систему, яка не лише контролює доступ до об’єкта, але й виступає важливим інструментом автоматизації, аналітики й підвищення ефективності управління персоналом. З огляду на стрімкий розвиток IoT-рішень, таких як платформа Blynk, подібні системи отримують новий рівень гнучкості, зручності налаштувань і доступності для малого та середнього бізнесу [5].

Системи контролю доступу поділяються за способом управління пристроями, такими як двері, турнікети, шлюзи тощо, на три основні типи [6]:

– автономні (локальні) – призначені для управління окремими пристроями без централізованого керування або обміну інформацією з оператором. Вони функціонують незалежно та ідеально підходять для контролю доступу до окремих приміщень;

– централізовані (мережеві) – об'єднують усі пристрої в єдину систему з передачею інформації на центральний пульт, дозволяючи оператору керувати доступом у режимі реального часу;

– універсальні системи – поєднують переваги автономних і мережевих рішень, працюючи під управлінням центрального контролера, але здатні переходити в автономний режим у разі збоїв мережі або центрального вузла.

Також СКД класифікуються за масштабом [6]:

- малі – кілька точок доступу, використовуються, наприклад, в офісах;
- середні – десятки точок і тисячі користувачів (банки, готелі, підприємства);
- великі – сотні точок і десятки тисяч користувачів (аеропорти, заводи, великі корпоративні об'єкти).

Автономні СКД використовуються для контролю доступу до окремих приміщень. Основою такої системи є автономний контролер із вбудованою базою даних ідентифікаторів. Для авторизації застосовуються магнітні, proximity або штрих-кодові картки. У більшості випадків на вхід встановлюється лише один зчитувач, а вихід здійснюється за допомогою кнопки або сенсора руху. Додатково встановлюється датчик положення дверей, доводчик та резервне джерело живлення.

Автономні системи зі збором даних мають розширену функціональність — вони фіксують інформацію про кожен прохід (дата, час, ID картки) [7]. За допомогою спеціального програмного забезпечення оператор може відстежувати переміщення персоналу, вести облік робочого часу, а також здійснювати візуальний контроль ідентифікаторів. Приклад автономних СКД на рис. 1.2.

Рисунок 1.2 – Автономна СКД

Мережеві СКД використовуються на великих об'єктах і забезпечують централізований контроль доступу на різних рівнях захищеності. Вони поділяються за типами точок доступу [8]:

– прохідні – обладнуються турнікетами чи шлюзами, часто за участі охорони. Важливим є контроль пропускної здатності та наявність сигналізації на спроби несанкціонованого доступу. Для об’єктів із високим рівнем безпеки можливе використання шлюзів із металодетекторами або засобами радіаційного/вибухового контролю. Може впроваджуватися відеоідентифікація та дворівнева авторизація (наприклад, картка + код);

– офісні приміщення – зазвичай не потребують високого рівня захисту. Тут доречні замки з технологією "вільних рук", що дозволяють не витягати картку з кишені;

– приміщення з підвищеним рівнем безпеки – передбачають багаторівневу ідентифікацію (включаючи біометричну), а також можливість входу лише кількох осіб одночасно;

– зовнішні об’єкти – передбачають спеціальні загороджуючі пристрої (ворота, шлагбауми) та зчитувачі, адаптовані до вуличних умов. Часто використовуються транспортні ідентифікатори та дорожні сенсори.

Для малих та середніх об’єктів контролери об’єднуються через RS-485 або інші інтерфейси в мережу, підключену до керуючого комп’ютера. У разі збоїв зв'язку контролери продовжують працювати автономно, накопичуючи події у своїй пам’яті, а після відновлення з’єднання передають накопичені дані до централі.

На великих об’єктах система зазвичай має кілька автоматизованих робочих місць (АРМ), об’єднаних у локальну мережу. Кожен оператор має доступ до єдиної бази даних на сервері, що дозволяє керувати системою з будь-якої точки. Створюються спеціалізовані АРМ — для прохідних, бюро перепусток, охорони тощо. Це забезпечує ефективне управління СКД навіть у складній інфраструктурі з численними будівлями. Приклад мережевих СКД зображено на рис.1.3.

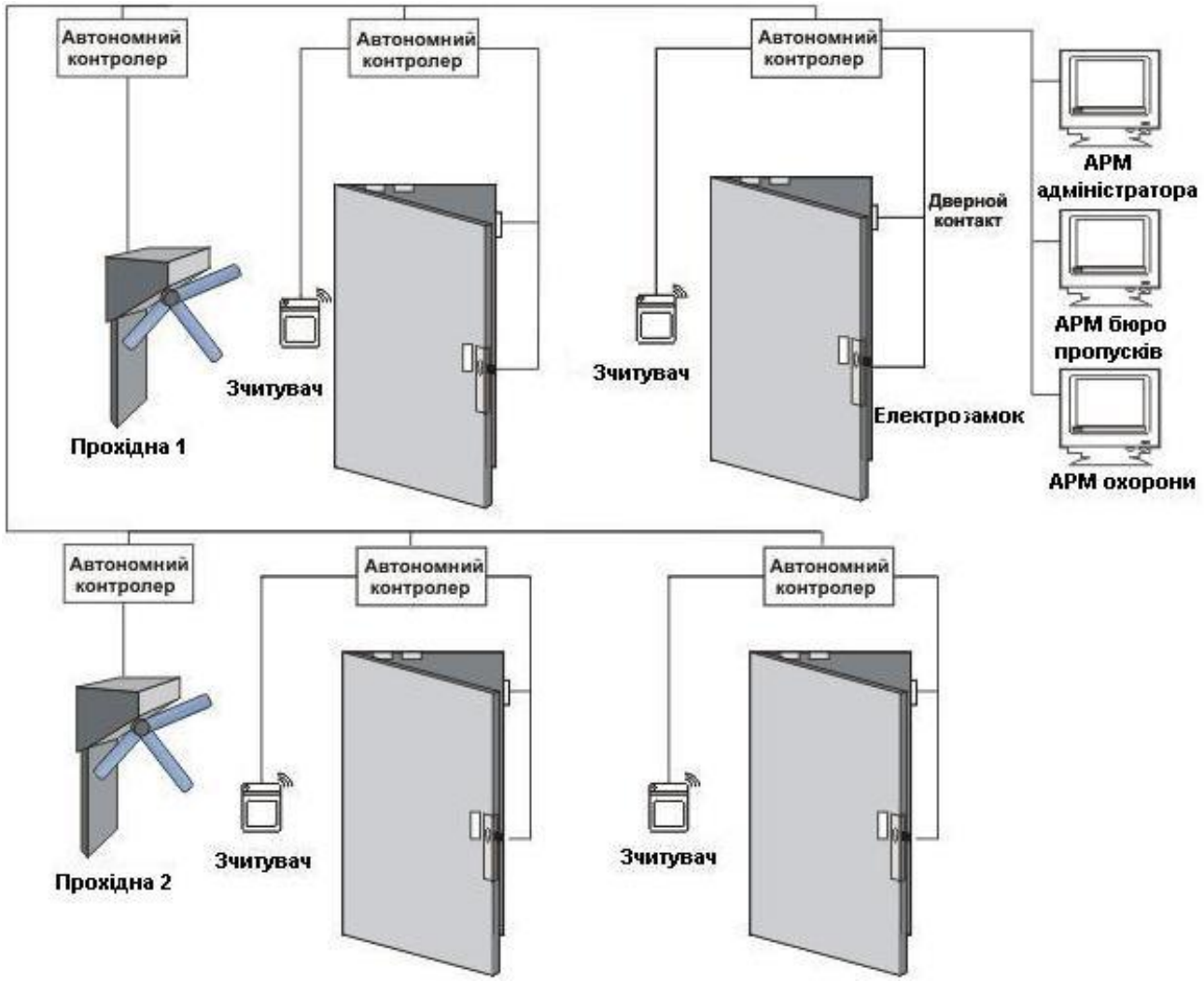


Рисунок 1.3 – Мережева СКД

Таким чином, системи контролю доступу є ключовим елементом сучасної безпекової інфраструктури, забезпечуючи регламентований допуск осіб і транспортних засобів до об’єктів різного призначення. Їх класифікація за принципом дії, масштабом та рівнем інтеграції дозволяє гнучко адаптувати СКД до конкретних умов експлуатації — від простих автономних рішень до складних централізованих систем. Розуміння цих класифікаційних ознак є основою для правильного вибору та ефективного впровадження СКД у загальну систему безпеки об'єкта.

Крім технічної реалізації, важливо також враховувати нормативні вимоги, особливості об’єкта, а також потенційні ризики і загрози. Правильно спроектована СКД не лише забезпечує контроль за доступом, а й дозволяє вести облік подій, інтегруватися з іншими безпековими підсистемами, такими як відеоспостереження,

охоронна і пожежна сигналізація, що в комплексі підвищує рівень захищеності підприємства або установи.

1.2 Основи Інтернету речей

Інтернет речей — це концепція взаємозв'язку фізичних пристроїв з можливістю обміну даними через Інтернет або інші мережі без участі людини. Вона охоплює величезну кількість об'єктів: від побутової техніки до промислових машин, від автомобілів до систем «розумного міста». Основна ідея IoT — забезпечити збір, передавання, обробку та використання інформації в автоматизованому режимі для підвищення ефективності, комфорту й безпеки [9].

Термін «Інтернет речей» був уведений у 1999 році Кевіном Ештоном — співзасновником Центру автоматичної ідентифікації при Массачусетському технологічному інституті (Auto-ID Center). Ця концепція передбачає створення глобальної мережі фізичних об'єктів, які здатні комунікувати між собою та з довкіллям за допомогою вбудованих технологій. Такі пристрої можуть самостійно здійснювати дії без участі людини, отримуючи або маючи власну IP-адресу — статичну або динамічну [10].

Ідея Інтернету речей розвинулась із поняття Machine-to-Machine (M2M) — міжмашинної комунікації, яку часто називають «Інтернетом машин». IoT включає M2M у свій склад як одну з основних складових і одночасно є наступним етапом у розвитку автоматизованих систем, що виникли завдяки технічному прогресу та переосмисленню ринку.

Парадигма M2M сформувалася завдяки трьом ключовим технологіям [11]:

- бездротовий зв'язок — дозволяє швидко розгортати рішення, легко адаптувати їх до нових завдань та змінювати конфігурацію;
- висока обчислювальна потужність пристроїв — забезпечує обробку даних безпосередньо на місці збору інформації, знижуючи навантаження на центральні вузли;
- хмарні сервіси — забезпечують масштабованість, зниження витрат та

зручний доступ до ресурсів і обчислень.

Разом ці компоненти створюють високоефективну систему, яка приносить користь як розробникам рішень, так і кінцевим користувачам.

До додаткових факторів, що сприяли поширенню IoT, належать [12]:

- низька вартість експлуатації пристроїв;
- мінімальне споживання енергії;
- готові до роботи рішення з базовою функціональністю;
- глобальна доступність апаратних засобів і платформ.

Слід відрізняти поняття «Інтернет речей» (як система) від «Інтернет-речі» (як окремий пристрій). Під інтернет-річчю мається на увазі будь-який об'єкт, який [13]:

- має доступ до Інтернету для отримання або передавання даних;
- ідентифікується у глобальній мережі (через IP або інший унікальний ідентифікатор);
- має інтерфейс для взаємодії з користувачем.

Інтернет-речі використовують єдиний протокол, що дозволяє кожному пристрою одночасно бути і споживачем, і постачальником сервісів. Проблема обмеженості адресного простору IPv4 стала на заваді масовому впровадженню IoT, але з розвитком IPv6, який дозволяє присвоїти мільйони адрес кожному жителю планети, ця перешкода фактично знята.

У системі Інтернету речей кожен вузол (пристрій) не лише передає дані, але й може приймати команди від інших пристроїв, утворюючи розподілену мережу, здатну вирішувати спільні завдання. Ці пристрої можуть організовуватись у локальні кластери, об’єднані географічно чи функціонально.

До технологій, що суттєво впливають на розвиток IoT або взаємодіють з ним, належать [14]:

- штучний інтелект (AI) та робототехніка;
- хмарні обчислення (Cloud);
- великі дані (Big Data);
- Інтернет нового покоління (Web 3.0);
- адитивне виробництво (3D-друк).

Інтернет речей не лише використовує ці інновації, а й виступає рушійною

силою їх еволюції. Багато нових можливостей і переваг IoT виникають як результат синергії цих технологій, і лише при правильному їх поєднанні можна досягти максимального ефекту.

Типова архітектура IoT включає такі основні компоненти: пристрої Інтернету речей, шлюзи, серверну інфраструктуру та клієнтський інтерфейс. Узагальнена схема представлена на рис.1.4.

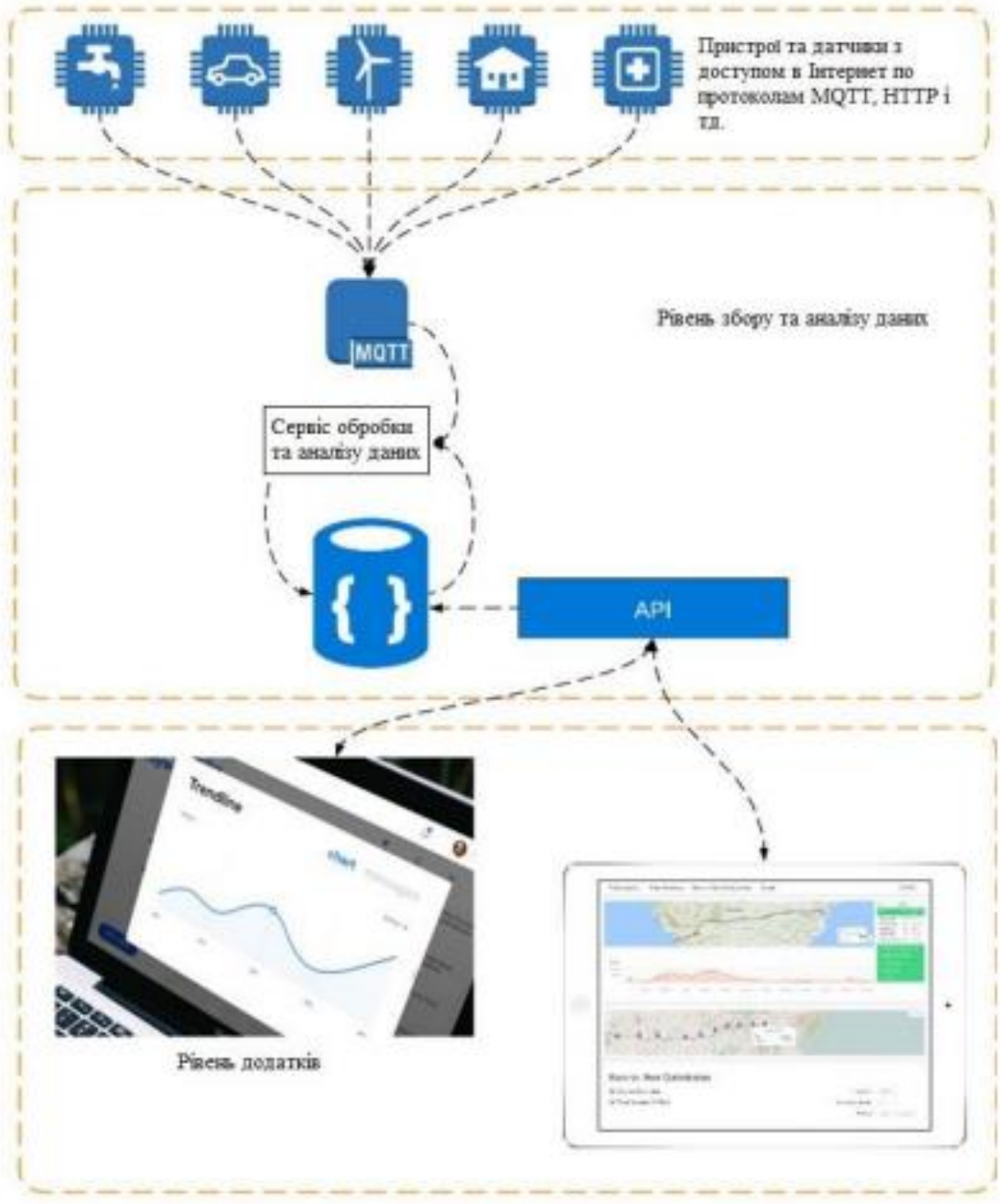


Рисунок 1.4 – Архітектура IoT

IoT-пристрої відповідають за збирання даних за допомогою сенсорів і здатні здійснювати фізичні дії. Сенсори, що інтегрують фізичний світ із цифровим, функціонують у режимі реального часу — отримують, обробляють і передають інформацію. Завдяки зменшенню розмірів сенсорів їх стало можливим вбудовувати безпосередньо в фізичні об’єкти. Вони бувають різних типів і призначені, наприклад, для вимірювання температури, тиску, швидкості, геопозиціонування тощо. Більшість сенсорів має обмежений обсяг пам’яті, що дозволяє зберігати певну кількість вимірювань. За функціональним призначенням сенсори класифікують як екологічні, медичні, побутові, автомобільні тощо [15].

Більшість сенсорів передає дані через агрегатори — шлюзи, що забезпечують комунікацію в локальних (LAN) чи персональних мережах (PAN), таких як Ethernet, Wi-Fi, Bluetooth, ZigBee або UWB. Деякі сенсори, які не потребують шлюзів, під’єднуються безпосередньо до серверів чи додатків через глобальні бездротові мережі (WAN) — GSM, GPRS, LTE [15].

Пристрої з низьким енергоспоживанням та невеликою швидкістю передачі даних зазвичай формують бездротові сенсорні мережі (WSN). Завдяки енергоефективності та можливості охоплювати великі території з численними автономними вузлами, WSN набувають дедалі більшого поширення.

Шлюзи приймають дані від сенсорів і надсилають їм керівні команди. Вони можуть бути як апаратними (наприклад, маршрутизатори), так і програмними, працюючи з різними мережевими протоколами. Уся ця інфраструктура формує конвергентне мережеве середовище, яке забезпечує об’єднання різнорідних комунікаційних систем в єдину платформу. Завдяки такій уніфікації користувачі можуть одночасно, незалежно і безпечно використовувати ресурси мережі [16].

Серверний рівень обробляє, зберігає й аналізує дані, отримані від сенсорів. Він може функціонувати як на фізичному обладнанні, так і на віртуальних машинах або в хмарному середовищі.

Клієнтська частина представлена мобільними чи веб-додатками, що забезпечують доступ до IoT-пристроїв, дозволяючи користувачам переглядати, аналізувати і взаємодіяти з отриманими даними в зручній формі.

Інтернет речей — це багаторівнева система, яка інтегрує фізичні об’єкти з

цифровим середовищем через мережу підключених пристроїв. Основу її архітектури складають сенсори, шлюзи, серверна інфраструктура та клієнтські додатки. Завдяки сенсорам IoT-пристрої можуть в режимі реального часу здійснювати збір і обробку даних, а також виконувати керовані дії без безпосередньої участі людини. Використання бездротових технологій, таких як Wi-Fi, ZigBee, LTE, дозволяє забезпечити гнучке підключення та масштабованість таких систем, а енергоефективні рішення на базі WSN сприяють впровадженню IoT у складних середовищах та на великих територіях [16].

Архітектура Інтернету речей базується на конвергенції мереж і хмарних сервісів, що забезпечують централізоване або розподілене зберігання, обробку та аналіз даних. Серверна частина відповідає за обчислення та збереження інформації, а клієнтські додатки — за візуалізацію й управління пристроями. У результаті така архітектура дозволяє створювати гнучкі, масштабовані й енергоефективні рішення, які відкривають нові можливості в автоматизації, моніторингу та управлінні у багатьох сферах — від побуту до промисловості.

1.3 Огляд сервісу Blynk

У сучасному світі, де пристрої стають усе більш «розумними» та взаємопов’язаними, платформа Blynk пропонує ефективне, гнучке та доступне рішення для розробників і компаній, які працюють у сфері Інтернету речей. Вона забезпечує повний цикл створення IoT-проєктів — від розробки інтерфейсу мобільного застосунку до керування пристроями через хмару, без необхідності писати складний код або будувати серверну інфраструктуру з нуля.

Blynk — це повністю інтегрована IoT-платформа, яка дозволяє створювати прототипи, масштабувати проєкти та дистанційно керувати пристроями будь-якого рівня складності. Це універсальне рішення, яке включає мобільні та веб-застосунки, хмарні сервіси, відкриті бібліотеки та локальні сервери. Головною перевагою платформи є її гнучкість, простота налаштування та підтримка широкого спектра апаратних платформ [17].

Однією з особливостей Blynk є доступність навіть для початківців. Платформа не вимагає від користувачів глибоких знань у сфері Інтернету речей або досвіду в програмуванні застосунків. Інтерфейс є інтуїтивно зрозумілим, налаштування віджетів виконується за кілька кліків, а приклади коду адаптовані для найпопулярніших пристроїв [17].

Це хмарна IoT-платформа, яка не прив'язана до конкретного апаратного забезпечення. Вона охоплює мобільні додатки, серверну інфраструктуру, засоби аналітики даних, автоматизацію процесів, а також базові елементи машинного навчання [9]. Blynk дозволяє розробникам інтегрувати будь-який мікроконтролер або пристрій у систему керування, надаючи уніфікований інтерфейс для роботи з ними.

Особливістю Blynk є те, що вона надає повний стек інструментів для створення «розумного» обладнання. Користувач має змогу здійснювати налаштування пристроїв (provisioning), моніторити їхній стан, керувати роботою в реальному часі, а також виконувати оновлення прошивки «по повітрю» (Over-the-Air). Додатково підтримується керування доступом користувачів, налаштування сповіщень, створення сценаріїв автоматизації та багато інших функцій [18].

Платформа успішно використовується як ентузіастами, які працюють із мікроконтролерами Arduino, ESP8266 або Raspberry Pi, так і великими компаніями, що займаються розробкою побутової електроніки, систем автоматизації, аграрного обладнання та промислових IoT-рішень. У рамках одного тарифного плану користувачі отримують доступ до повноцінної хмарної інфраструктури з можливістю створення брендованих мобільних застосунків без написання коду [18]. Загальну схему роботи сервісу Blynk зображено на рисунку 1.5.

Ключові компоненти платформи Blynk [19] включають:

- Blynk App — мобільний застосунок, що дозволяє створювати інтерфейси для керування пристроями за допомогою візуальних віджетів, таких як кнопки, слайдери, графіки та дисплеї;
- Blynk Server — сервер, що виконує роль посередника між мобільним застосунком і фізичними пристроями. Можна використовувати офіційний хмарний сервер або розгорнути локальний сервер з відкритим кодом, наприклад, на

Raspberry Pi;

– Blynk Libraries — набір бібліотек для популярних платформ, зокрема Arduino, ESP8266, ESP32, Raspberry Pi тощо, які забезпечують двосторонній зв’язок між мікроконтролером та сервером.

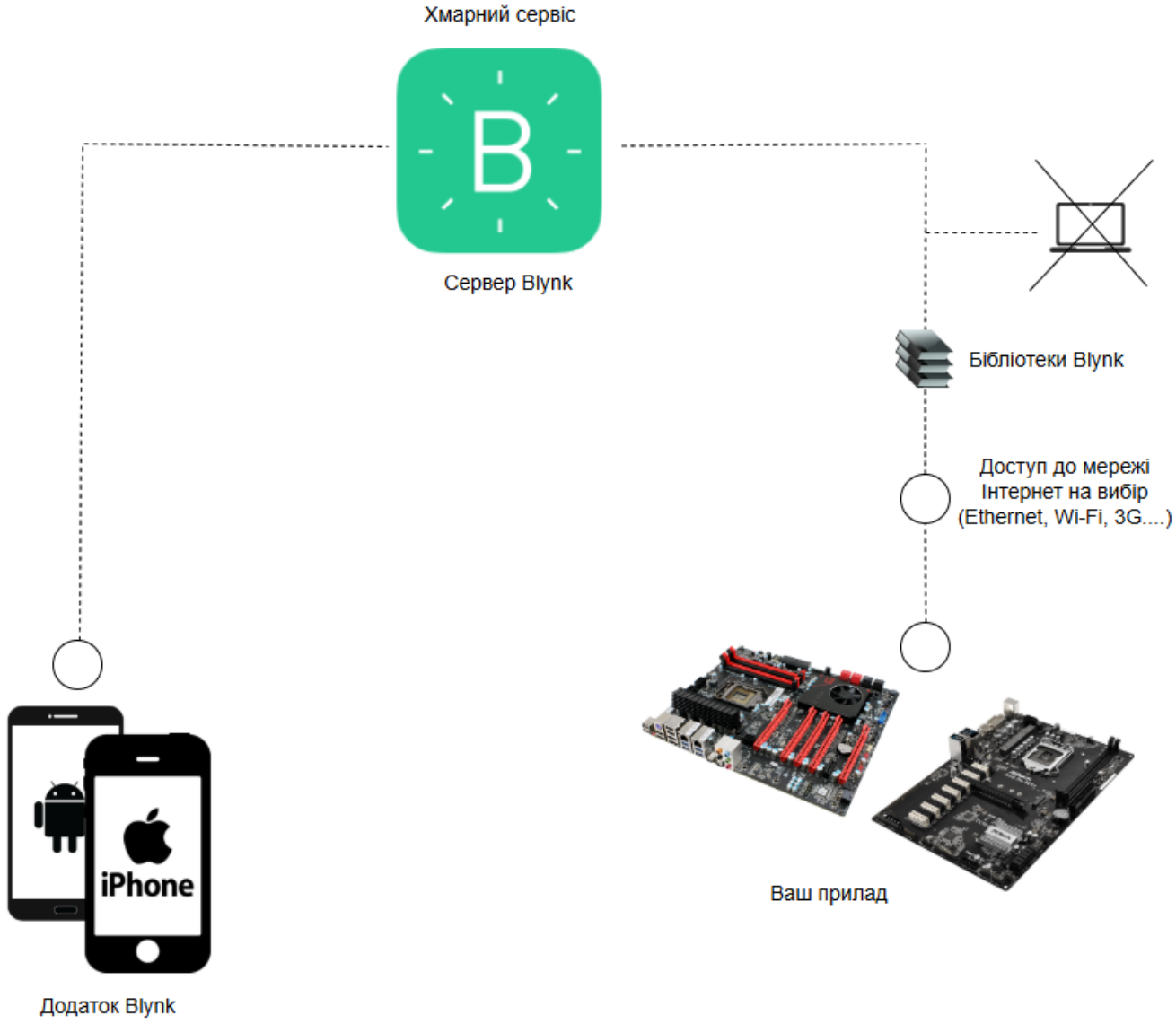


Рисунок 1.5 – Сервіс Blynk

До основних можливостей і переваг Blynk можна віднести:

– уніфіковану структуру API для всіх підтримуваних пристроїв;

– підтримку різних типів з’єднання, включаючи Wi-Fi, GSM, Bluetooth, BLE, Ethernet та інші;

– широкий набір графічних віджетів для створення зручного інтерфейсу

користувача;

– можливість прямого керування фізичними пін-контактами без додаткового програмування;

– використання віртуальних пінів, що дозволяє реалізовувати гнучку логіку взаємодії;

– можливість зберігати історію даних, надсилати push-сповіщення, email та твіти;

– використання Bridge-widget для взаємодії між декількома пристроями в межах однієї системи.

Blynk без проблем інтегрується з Arduino IDE. Після встановлення відповідної бібліотеки через вбудований менеджер бібліотек користувач отримує доступ до великої кількості прикладів у меню «Файл» → «Приклади». Ці приклади охоплюють понад 400 моделей обладнання, включаючи найпоширеніші мікроконтролери: ESP8266, ESP32, NodeMCU та інші [20]. Вони згруповані за типом з'єднання (Wi-Fi, Bluetooth, GSM тощо) і демонструють роботу з конкретними віджетами, що суттєво спрощує розробку.

Для зручності користувача передбачений веб-інтерфейс — Blynk Web Dashboard, який відкривається після створення облікового запису. Початківцям рекомендується скористатись майстром швидкого старту (Quick Start), який покроково допомагає підключити перший пристрій до хмари. Після завершення налаштування автоматично генерується шаблон проєкту з усіма необхідними параметрами. Це значно зменшує кількість помилок та прискорює процес входження в IoT-середовище [20].

Платформа Blynk підтримує реалізацію широкого спектра практичних завдань. Серед прикладів її застосування [20] можна виділити такі:

– розумний будинок — автоматизоване керування освітленням, клімат-контролем, сигналізацією та іншими системами;

– сільське господарство — моніторинг мікроклімату в теплицях, контроль вологості ґрунту та автоматизація поливу;

– промисловість — відстеження параметрів стану обладнання, дистанційне керування машинами та попередження про аварійні ситуації;

– освіта та хобі — використання в освітніх курсах, студентських проєктах та домашніх експериментах.

Таким чином, Blynk є універсальним і потужним інструментом, який суттєво спрощує реалізацію IoT-проєктів будь-якої складності. Платформа дозволяє зосередитись на логіці й функціоналі пристрою, знімаючи з розробника потребу в налаштуванні інфраструктури, створенні мобільного застосунку або складному протоколюванні. Саме завдяки цьому Blynk активно використовується як початківцями, так і професійними інженерами в комерційних рішеннях.

1.4 Аналіз існуючих рішень контролю доступу на базі IoT

Інтернет речей (IoT) відкрив нові горизонти в організації систем контролю доступу, дозволивши створювати інтелектуальні, масштабовані та мобільно керовані рішення. Сучасні IoT-системи здатні не лише відкривати або блокувати доступ, а й виконувати аналітику, обробляти події, взаємодіяти з користувачами через мобільні додатки, а також інтегруватися з системами відеоспостереження, пожежної безпеки й клімат-контролю [21].

Brivo Onair — це потужна хмарна платформа для організації системи контролю доступу, що орієнтована насамперед на комерційні, корпоративні та житлові об’єкти. Вона дозволяє віддалено керувати доступом до будівель, приміщень і ліфтів через браузер або мобільний додаток. Рішення Brivo засноване на принципах хмарної архітектури (SaaS), що дозволяє адміністраторам контролювати великі об'єми обладнання без необхідності фізичного доступу до серверів або обслуговування локальної інфраструктури [22].

Архітектура Brivo Onair включає три основні компоненти: хмарний сервер, контролери доступу, та інтерфейси користувача. Хмарний сервер відповідає за зберігання даних користувачів, журналів подій, прав доступу та забезпечує безпечну комунікацію між клієнтом і пристроями. Контролери, встановлені безпосередньо на об’єктах, зчитують RFID-мітки, PIN-коди або біометричні дані та обробляють запити на доступ. Інтерфейс адміністрування реалізований через веб-

додаток та мобільні платформи, що дозволяє керувати доступом в реальному часі, задавати розклади, групи користувачів, зони доступу тощо [23].

Перевагами Brivo Onair є, насамперед, масштабованість і простота використання. Платформа легко розширюється від кількох дверей до тисяч, не потребуючи зміни базової інфраструктури. Завдяки хмарній моделі, оновлення відбуваються автоматично, а дані надійно зберігаються й захищені за допомогою сучасних протоколів шифрування. Крім того, система дозволяє інтегрувати відеоспостереження: наприклад, відкриття дверей можна супроводити автоматичним записом з камери, що полегшує аудит подій.

Brivo також має API, що робить можливим інтеграцію з іншими корпоративними системами — наприклад, CRM, ERP, пожежними сигналізаціями, або системами моніторингу відвідуваності. Це значно розширює функціональність платформи й робить її придатною для застосування в офісах, складах, медичних закладах, житлових комплексах. Мобільний додаток підтримує створення віртуальних ключів, які можна тимчасово передати іншим особам (наприклад, гостям, орендарям або сервісному персоналу).

Серед недоліків варто згадати вартість підписки, яка може бути високою для малого бізнесу або приватних об’єктів. Також, як і всі хмарні рішення, Brivo Onair залежить від стабільного інтернет-з’єднання: у разі тривалого відключення мережі функціональність системи може бути частково обмежена. Проте контролери мають можливість працювати автономно протягом певного часу, зберігаючи базову логіку доступу на рівні локального обладнання [24].

ButterflyMX — це сучасна система домофонії та контролю доступу, побудована на основі хмарних технологій та призначена для житлових і комерційних об'єктів. Основна мета платформи — забезпечення безпечного та зручного керування входом у будівлі за допомогою смартфона. Система дозволяє мешканцям або користувачам відкривати двері віддалено, отримувати відеодзвінки від гостей, надсилати віртуальні ключі тимчасового доступу та отримувати повідомлення про відвідувачів у реальному часі. Це робить її ідеальним рішенням для багатоквартирних будинків, офісів, коворкінгів та приватних об'єктів [25].

Архітектура ButterflyMX складається з декількох ключових компонентів:

розумної панелі (smart intercom), хмарного серверу, мобільного додатку та адміністративної панелі. Панель домофону обладнана відеокамерою, мікрофоном і сенсорним дисплеєм, що дозволяє гостям здійснювати виклики за номером квартири або іменем мешканця. Коли виклик надходить, користувач отримує сповіщення на смартфон, де може переглянути відео у реальному часі та дозволити доступ натисканням однієї кнопки. Уся передача даних захищена шифруванням, а записи дзвінків можуть зберігатись для перегляду згодом.

Серед переваг ButterflyMX — простота встановлення (використовується IP-з'єднання, тому не потрібно прокладати традиційні телефонні лінії), сумісність із мобільними пристроями, можливість інтеграції з іншими системами управління будівлями, такими як управління ліфтами або контроль доступу до спільних приміщень (паркінг, спортзал, технічні кімнати). Крім того, система підтримує автоматизацію доступу для доставки (наприклад, для Amazon або UberEats), що робить її зручною в умовах сучасного способу життя [26].

До недоліків можна віднести високу початкову вартість обладнання та обслуговування, яка може бути не виправданою для малих об'єктів або приватних осель. Також для повноцінної роботи система потребує стабільного підключення до інтернету, без якого можуть бути недоступні функції відеодзвінків і віддаленого управління. Однак локальні ключові операції (наприклад, відкриття з панелі) можуть продовжити працювати навіть за відсутності зв'язку з сервером [26].

ButterflyMX дедалі частіше інтегрується з сучасними рішеннями «розумного будинку» та підтримує такі сервіси, як Google Home або Alexa. Це відкриває нові сценарії використання — наприклад, відкриття дверей голосовою командою або запуск автоматичних дій при прибутті гостей. Такий підхід підвищує зручність та адаптивність системи до потреб різних користувачів [27].

VersionX — це сучасна IoT-платформа для контролю доступу, яка поєднує хмарні технології, edge computing та багатофункціональне апаратне забезпечення. Вона розроблена для забезпечення безпеки та ефективного управління доступом у різноманітних середовищах, включаючи офіси, житлові комплекси, навчальні заклади та промислові об'єкти. Система підтримує різні методи автентифікації, такі як розпізнавання обличчя, RFID, QR-коди, мобільні додатки, клавіатури та

відбитки пальців, що забезпечує гнучкість та зручність для користувачів [28].

Архітектура VersionX базується на edge computing, що дозволяє обробляти дані безпосередньо на пристроях, зменшуючи залежність від постійного підключення до Інтернету. Це забезпечує безперервну роботу системи навіть у разі перебоїв з мережею. Крім того, система не вимагає встановлення додаткового програмного забезпечення або наявності спеціалізованих ПК, що спрощує процес впровадження та знижує витрати на обслуговування.

Однією з ключових переваг VersionX є її масштабованість. Система може керувати від одного до 48 точок доступу, що робить її придатною як для малих підприємств, так і для великих корпорацій. Інтеграція з іншими системами, такими як управління відвідувачами, паркуванням, охороною та обліком робочого часу, дозволяє створити єдину екосистему для управління всіма аспектами безпеки та доступу [28].

VersionX також пропонує хмарну платформу для моніторингу та аналітики, яка надає реальний час звіти та історію подій. Це дозволяє адміністраторам швидко реагувати на інциденти та приймати обґрунтовані рішення на основі даних. Система підтримує інтеграцію з мобільними додатками, що забезпечує зручний доступ до функцій управління з будь-якого місця [29].

Серед недоліків можна відзначити потенційні витрати на впровадження для малих підприємств, а також необхідність навчання персоналу для ефективного використання всіх функцій системи. Однак, з огляду на її гнучкість, масштабованість та широкий спектр можливостей, VersionX є потужним інструментом для організацій, які прагнуть підвищити рівень безпеки та ефективності управління доступом [30].

Отже, існуючі IoT-рішення демонструють гнучкість, інтегрованість та широкий функціонал. Проте сервіси типу Blynk роблять такі технології доступними навіть для студентських чи малобюджетних проєктів, відкриваючи нові можливості для кастомізації та автоматизації систем доступу.

1.5 Постановка задачі

Метою даної кваліфікаційної роботи є розробка функціонального прототипу системи контролю доступу на основі технологій Інтернету речей (IoT) із використанням платформи Blynk. Передбачається, що така система дозволить реалізувати надійний, гнучкий та зручний механізм контролю, моніторингу й керування доступом у режимі реального часу, що особливо актуально в умовах зростаючих вимог до безпеки житлових і комерційних об’єктів. Запропоноване рішення має забезпечити не лише базову функціональність контролю доступу, а й інтеграцію з мобільним додатком, можливість обробки подій та збирання даних із підключених пристроїв.

На основі теоретичного аналізу, проведеного в першому розділі, зроблено низку важливих висновків. По-перше, системи контролю доступу можуть класифікуватися за рядом критеріїв, зокрема рівнем складності, засобами ідентифікації (магнітні картки, RFID, біометрія, мобільні ключі тощо) та способом управління (локальне або дистанційне). По-друге, технології IoT дозволяють організувати ефективне збирання, обробку та передачу даних у реальному часі, що суттєво підвищує гнучкість та інформативність систем контролю доступу. По-третє, платформа Blynk, яка підтримує широкий спектр мікроконтролерів і дозволяє швидко створювати мобільні інтерфейси, є зручним та доступним інструментом для побудови подібних систем.

Також було проаналізовано сучасні рішення контролю доступу на базі IoT, серед яких було виявлено як значні переваги, так і певні недоліки. До переваг можна віднести можливість віддаленого доступу, масштабованість, високу адаптивність до різних типів об’єктів та простоту користування. Водночас, окремі рішення демонструють обмежену гнучкість у налаштуванні, потребують значних ресурсів на інтеграцію або не дозволяють масштабувати систему без суттєвих змін архітектури.

У зв’язку з цим основними завданнями, поставленими у кваліфікаційній роботі є:

– розробка архітектури IoT-системи контролю доступу з урахуванням

сучасних вимог до безпеки, надійності та масштабованості;

– вибір та реалізація апаратної частини системи, яка включає контролер, модуль підключення до мережі, механізм замикання/розмикання доступу, а також засоби індикації та ідентифікації користувача;

– проведення тестування системи з метою оцінки її ефективності, надійності, зручності використання та перспектив масштабування.

У результаті виконання роботи очікується отримати повністю функціональний прототип IoT-системи контролю доступу, що здатний стати основою для подальшого розвитку в повноцінний продукт — як для побутового, так і для промислового використання. Окрім практичної цінності, запропонована система також демонструє можливості поєднання сучасних цифрових технологій із прикладними рішеннями в сфері фізичної безпеки.

2 РОЗРОБКА СИСТЕМИ ДОПУСКУ НА БАЗІ BLYNK IOT

2.1 Вибір компонентів та інструментів розробки

Проєкт системи допуску передбачає створення пристрою, який забезпечуватиме можливість віддаленого керування доступом до приміщення за допомогою мобільного або веб-інтерфейсу. Для реалізації такого пристрою необхідно підібрати надійне, функціональне та недороге апаратне забезпечення, яке забезпечить бездротовий зв’язок з хмарним сервісом, керування виконавчими елементами, індикацію стану системи, а також збір і передачу сенсорних даних.

Основним елементом апаратної частини є мікроконтролер Wemos D1 mini [31], який побудований на базі чипа ESP8266, рис. 2.1. Це високофункціональний мікроконтролер з вбудованим Wi-Fi-модулем, що підтримує стандарт 802.11 b/g/n і дозволяє легко підключати пристрій до бездротової мережі.

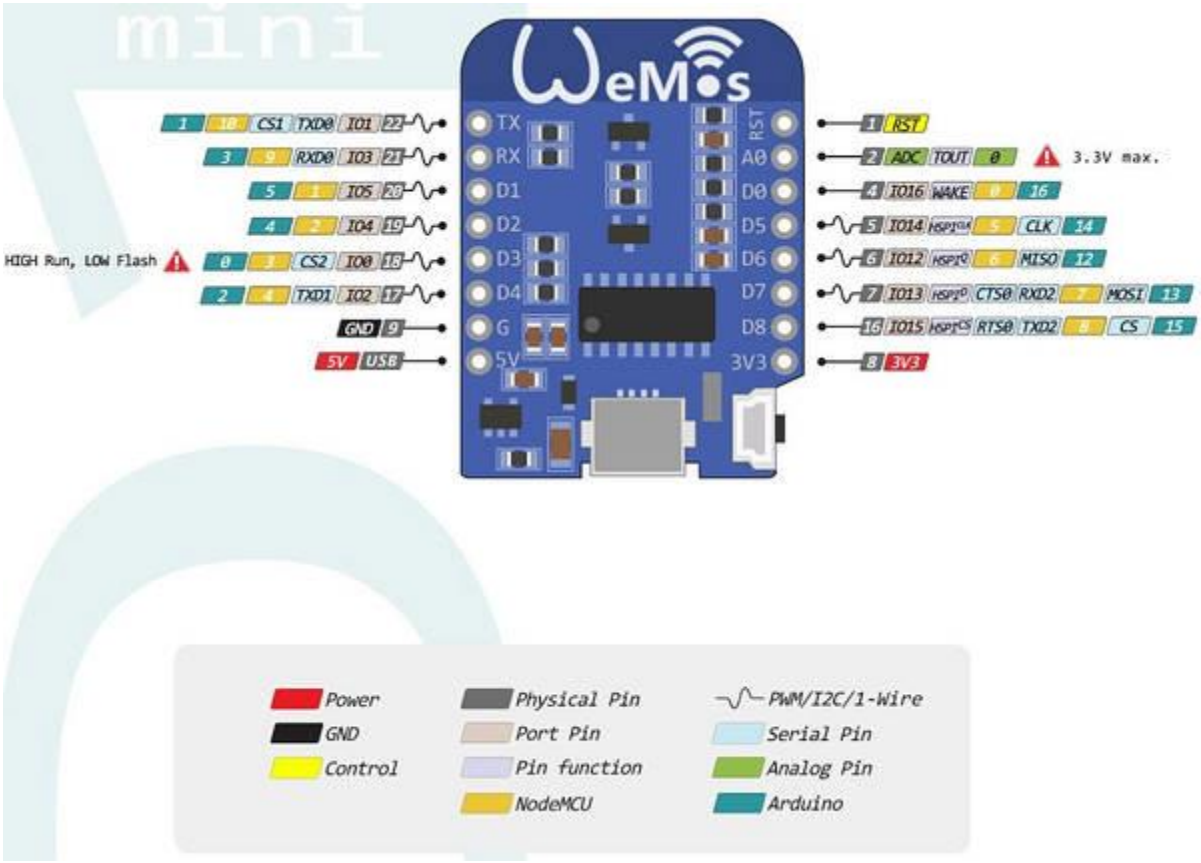


Рисунок 2.1 – Мікроконтролер Wemos D1 mini

Його головною перевагою є можливість програмування в середовищі Arduino

IDE, що значно спрощує розробку системи та інтеграцію з сервісом Blynk IoT. Плата має компактні розміри, що дозволяє використовувати її в обмеженому просторі. Крім того, вона має достатню кількість цифрових входів/виходів для підключення керуючих і сенсорних елементів, а також один аналоговий вхід для збору аналогових даних. Підтримка живлення через micro-USB або зовнішній блок 5В робить цей мікроконтролер зручним для розробки і тестування в реальних умовах експлуатації.

Для керування фізичним елементом доступу, таким як електромеханічний або електромагнітний замок, до контролера підключається модуль реле на 5В. Це електромеханічний пристрій, що дозволяє керувати силовими навантаженнями шляхом подачі низьковольтного сигналу від мікроконтролера. У даному проєкті реле використовується як електронний перемикач, який або подає напругу на замок (відкритий стан), або її відключає (закритий стан). Такий тип реле має гальванічну розв’язку завдяки вбудованому оптопарному інтерфейсу, що забезпечує безпечне розділення логічної частини системи та силового кола [32]. Використання реле забезпечує надійність і універсальність керування зовнішніми пристроями, адже воно здатне працювати як із постійним, так і зі змінним струмом.

Візуальна індикація поточного стану системи здійснюється за допомогою світлодіодів. Вони дозволяють миттєво визначити, чи перебуває система в активному або пасивному режимі, чи увімкнений доступ, або чи заблоковано вхід. Світлодіоди підключаються до цифрових виходів мікроконтролера через резистори обмеження струму [33]. Крім фізичних світлодіодів, передбачається використання віртуальних індикаторів у застосунку Blynk, що дозволяє дублювати стан системи у вигляді світлових сигналів на екрані смартфона. Така індикація суттєво підвищує зручність користування системою та дозволяє оперативно реагувати на зміну її стану.

Окрему роль у системі відіграє аналоговий сенсор, у якості якого в тестовій конфігурації використовується потенціометр. Це змінний резистор, який підключається до аналогового входу A0 мікроконтролера і дозволяє змінювати рівень вхідної напруги залежно від положення повзунка [34]. Його використання дозволяє перевірити працездатність модуля зчитування аналогових сигналів, а

також налагодити передавання аналогових значень до інтерфейсу Blynk, де вони можуть відображатися у вигляді шкали або графіка. У майбутньому потенціометр можна замінити на функціональні сенсори, наприклад, датчик температури, фоторезистор, або датчик вологості, що дозволить системі реагувати на зміни в навколишньому середовищі.

Для живлення всіх компонентів системи використовується стабілізоване джерело постійного струму з номінальною напругою 5В. Живлення може здійснюватися як через порт micro-USB, так і за допомогою зовнішнього блока живлення. Рекомендовано використовувати окремий адаптер із потужністю не менше 1А, що забезпечує стабільну роботу системи, особливо при підключенні виконавчих пристроїв з підвищеним енергоспоживанням, зокрема реле або електромеханічних замків.

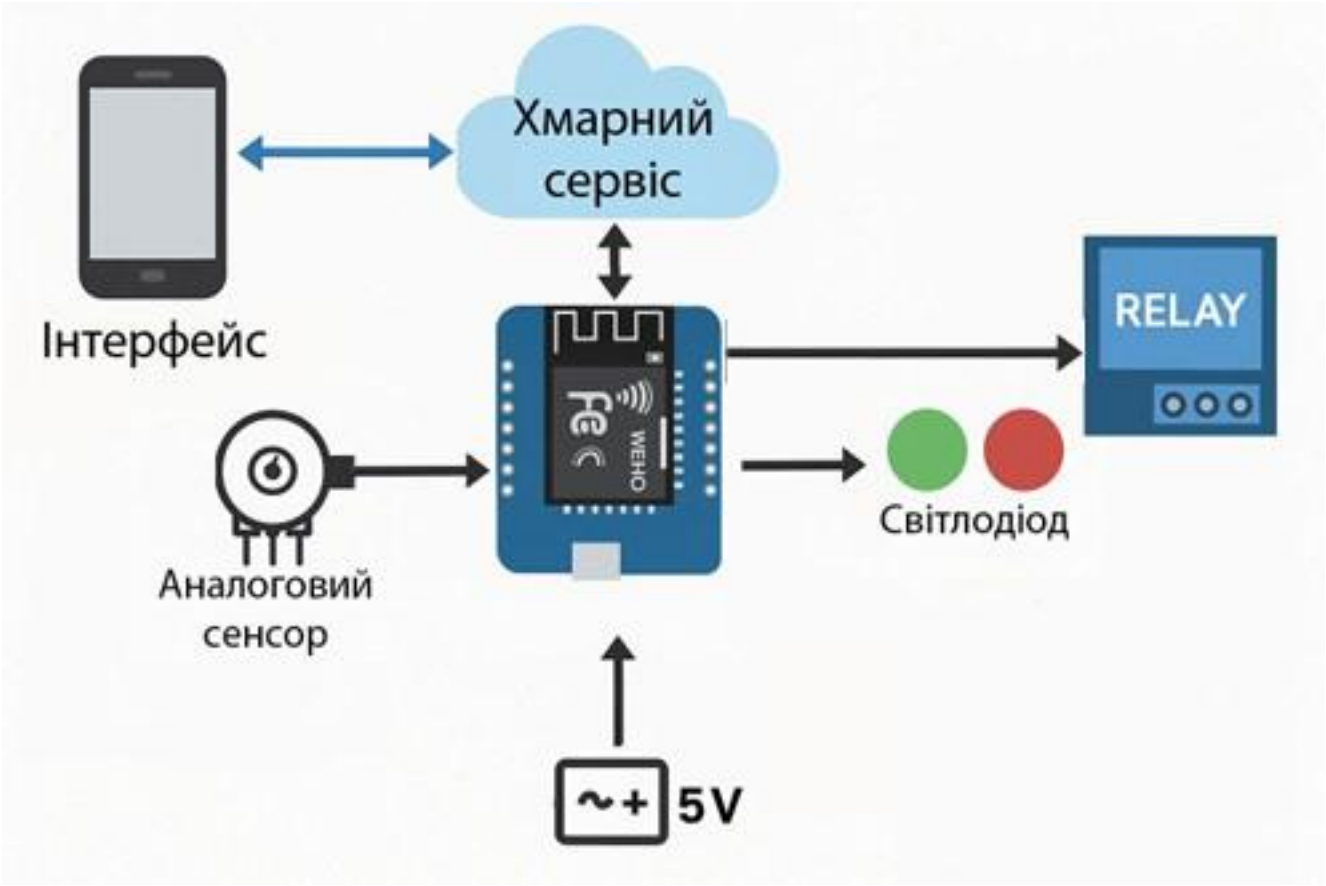


Рисунок 2.2 – Схема апаратного забезпечення

Підібране апаратне забезпечення, загальна схема рис. 2.2, забезпечує повну функціональність для побудови базової версії IoT-системи доступу. Воно є

сумісним із сучасними засобами розробки та підтримує інтеграцію з хмарними сервісами, що дозволяє реалізувати віддалене керування, візуалізацію даних і масштабування проєкту.

Програмне забезпечення відіграє ключову роль у побудові системи допуску, оскільки забезпечує реалізацію логіки керування, передачу даних, обробку сигналів, з'єднання з хмарними сервісами та взаємодію з користувачем. Для створення, компіляції, завантаження та налагодження програмного коду використовується середовище розробки Arduino IDE, яке є сумісним із платформою ESP8266. Для керування пристроєм через Інтернет застосовується хмарний сервіс Blynk IoT, що забезпечує візуальний інтерфейс, зв'язок між пристроєм і користувачем, а також збереження і передачу даних у реальному часі.

Arduino IDE є вільно доступним програмним середовищем, призначеним для розробки програмного коду (скетчів) для мікроконтролерів на базі архітектури AVR, ARM та ESP [35]. Для забезпечення підтримки плати Wemos D1 mini необхідно додатково встановити пакет плати ESP8266 через менеджер плат. Окрім того, для реалізації взаємодії з платформою Blynk потрібно встановити відповідну бібліотеку Blynk та бібліотеку SimpleTimer, що дозволяє виконувати повторювані дії з певним інтервалом. Програмний код розробляється на мові програмування C/C++ із застосуванням специфічних бібліотек Arduino, що значно спрощує створення та модифікацію логіки системи.

Blynk IoT — це хмарна платформа, розроблена для створення розумних пристроїв з можливістю керування через мобільний або веб-інтерфейс. Вона дозволяє створювати персоналізовані панелі керування, використовуючи графічні компоненти, такі як кнопки, індикатори, шкали, графіки, перемикачі тощо [36]. Платформа складається з трьох основних компонентів: мобільного застосунку, хмарного сервера та бібліотеки, яку інтегрують у код пристрою. Користувач створює візуальний шаблон (Template), в якому визначає логіку елементів інтерфейсу, тип пристрою, метод з’єднання та відповідні datastream-и. У системі також використовується унікальний токен автентифікації (Auth Token), який є ключем для зв’язку між пристроєм та обліковим записом користувача на сервері Blynk.

Для забезпечення стабільного з'єднання між пристроєм і хмарним сервером у коді необхідно ініціалізувати з'єднання Wi-Fi за допомогою введення назви мережі та пароля, після чого викликається функція Blynk.begin(), яка ініціює хмарну сесію. У циклі loop() постійно виконується обробка запитів за допомогою функції Blynk.run(), а також, за потреби, викликаються інші функції таймера. Усі дії, пов’язані з відправкою або прийманням даних, здійснюються за допомогою віртуальних пінів, які не прив’язані до фізичних входів/виходів, що дає змогу будувати гнучку логіку взаємодії з інтерфейсом [37].

Програмне забезпечення дозволяє не лише керувати виконавчими механізмами, а й зчитувати показники з датчиків, надсилати повідомлення, встановлювати порогові значення для автоматичних реакцій та вести базову аналітику. Платформа підтримує також інтеграцію з протоколом MQTT, що відкриває можливості для масштабування системи та підключення до більш розвинених IoT-систем.

Загалом програмне забезпечення, що використовується в системі, поєднує простоту налаштування з широкими функціональними можливостями, що дозволяє ефективно реалізувати основні завдання проєкту — керування доступом, індикацію стану, моніторинг показників і забезпечення віддаленого доступу до пристрою через інтернет.

2.2 Розробка та налаштування Blynk Dashboard

У межах реалізації системи допуску важливим етапом стало створення візуального інтерфейсу користувача, який забезпечує зручне й інтуїтивне керування пристроєм через мобільний або веб-додаток. Цей інтерфейс створюється у середовищі Blynk Web Dashboard, що входить до складу хмарної платформи Blynk IoT. Завдяки графічному підходу до проєктування інтерфейсу не виникає потреби у використанні складного програмного забезпечення — усі елементи додаються за допомогою перетягування (drag & drop) та налаштовуються через зручні діалогові вікна [38].

Розробка панелі керування почалася зі створення нового шаблону (Template) для пристрою, на базі якого буде працювати система доступу. Увійшовши до вебінтерфейсу Blynk через сайт [39], необхідно перейти до розділу Templates і натискає кнопку «New Template». У вікні створення шаблону необхідно задати ім’я, вибрати тип апаратного забезпечення — у даному випадку ESP8266, а також спосіб з’єднання — WiFi. Після збереження шаблону, рис 2.3, він автоматично стає доступним для подальшого налаштування, зокрема — створення візуальної панелі керування, datastream та зв’язку з мобільним застосунком.

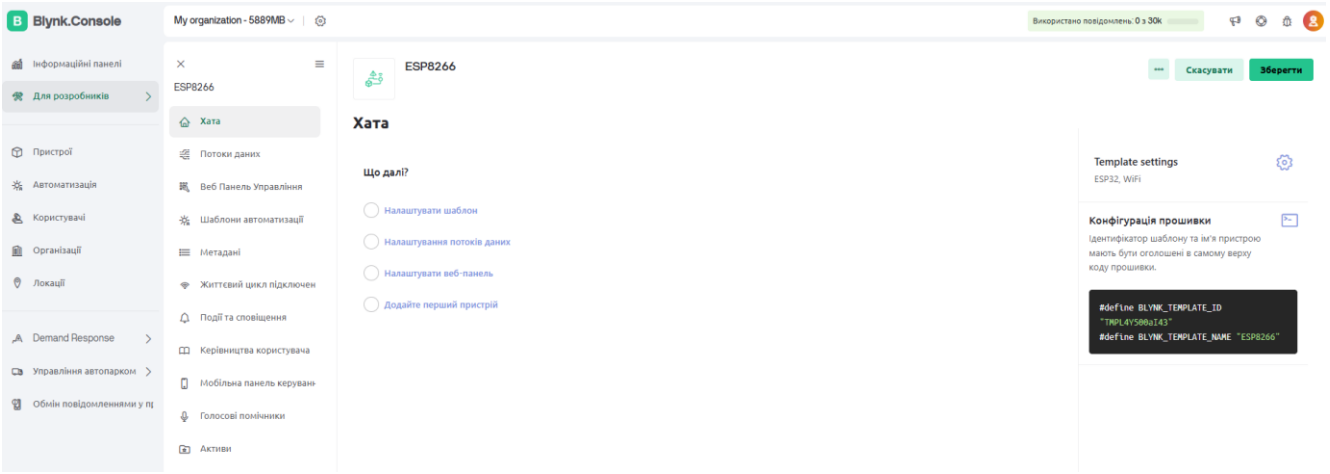


Рисунок 2.3 – Створення шаблону

Наступним кроком є налаштування datastream - потоки даних, які встановлюють зв’язок між графічними віджетами інтерфейсу та конкретними пін-контактами на мікроконтролері. Datastream визначає тип даних, віртуальний або фізичний пін, діапазон значень, одиниці вимірювання та інші параметри. Першим було налаштовано реле, рис. 2.4. Значення, які були задані під час налаштування:

– пип: Virtual Pin;
– пін: V1;
– тип даних: Integer (ціле число);
– мінімальне значення: 0;
– максимальне значення: 1;
– псевдонім: relay.

Цей потік даних буде використовуватись для передавання команди вмикання

або вимикання реле, що відповідає за доступ.

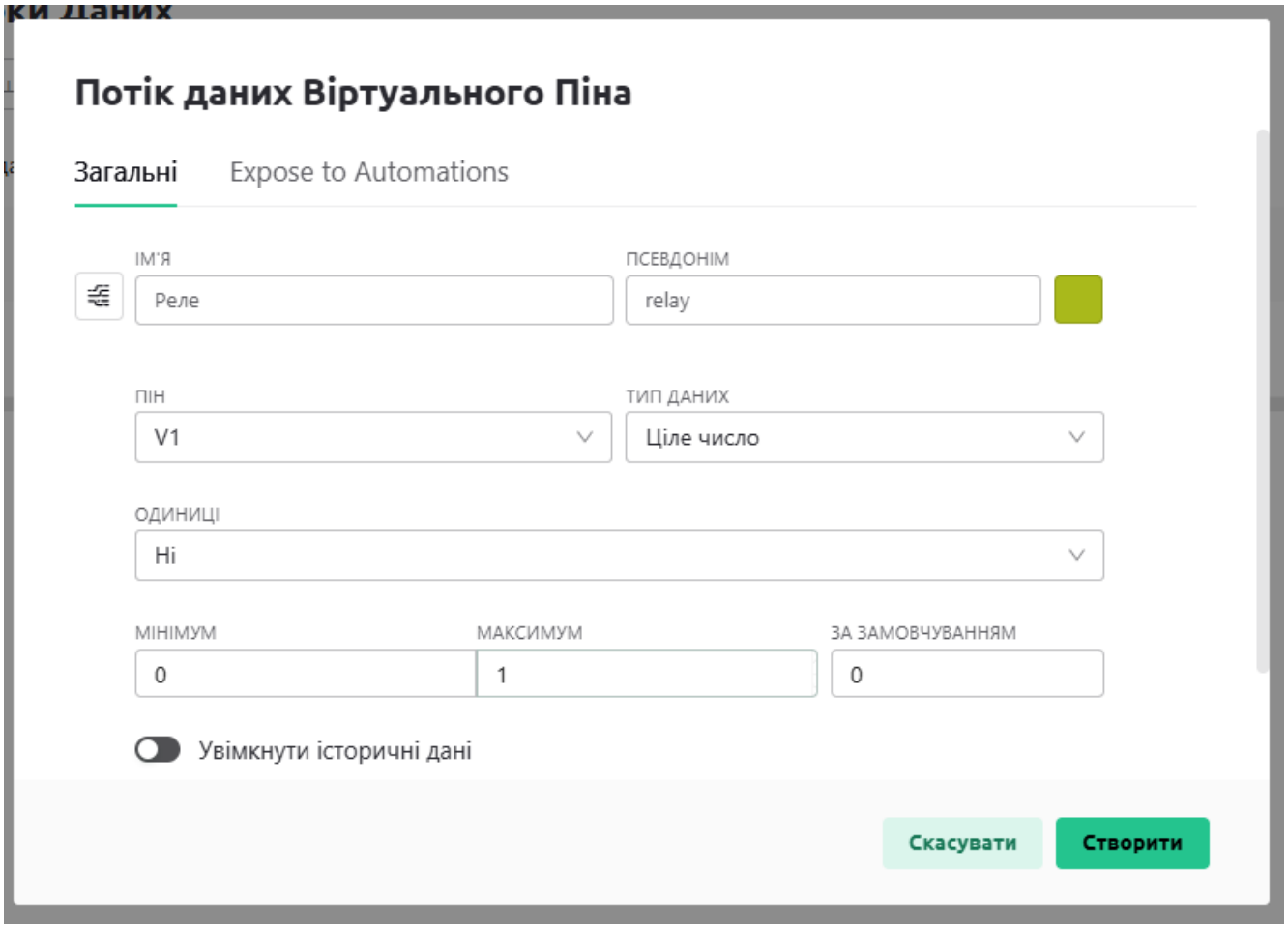


Рисунок 2.4 – Налаштування реле

Аналогічно було створено і для збору даних з аналогового сенсора, такого як потенціометр, рис. 2.5. Значення, які були встановлені:

– тип: Analog;
– пін: A0;
– тип даних: Integer;
– мінімальне значення: 0;
– максимальне значення: 1;
– псевдонім: analog.

Цей потік забезпечує зчитування значень з аналогового датчика, наприклад, потенціометра, і передавання цих даних у графічний інтерфейс.

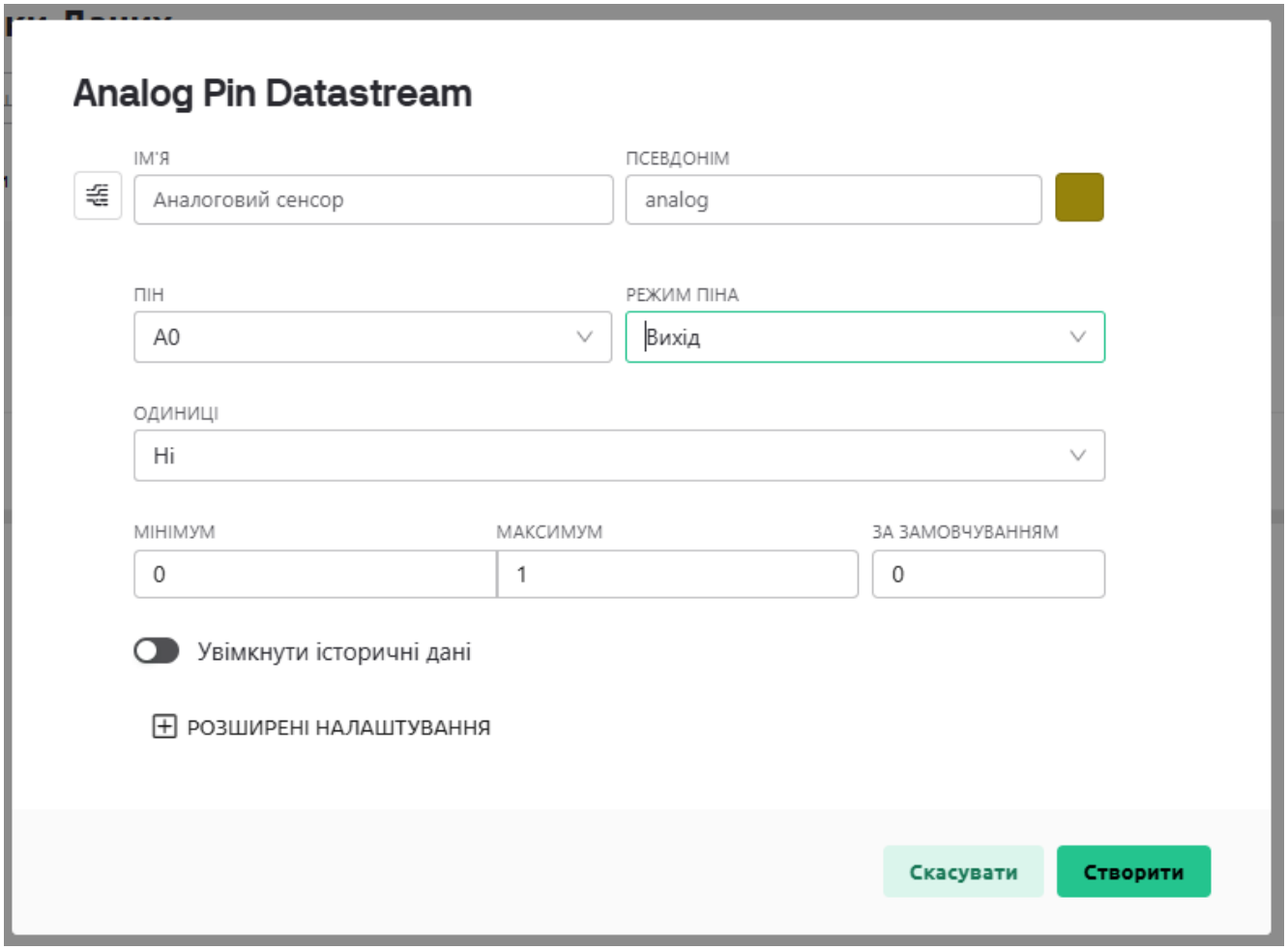


Рисунок 2.5 – Налаштування аналогового сенсора

Наступним було налаштовано потоки для віртуальних індикаторів LED, рис. 2.6, які будуть активуватись відповідно до умов доступу.

Значення, що були налаштовані:

– тип: Virtual;

– пін: V2 для сигналізації заблокованого доступу;

– пін: V4 для дозволеного доступу;

– тип даних: Integer;

– мінімальне значення: 0;

– максимальне значення: 1;

– псевдонім: LED.

Ці потоки використовуються для візуального відображення стану системи доступу як у мобільному, так і вебінтерфейсі.

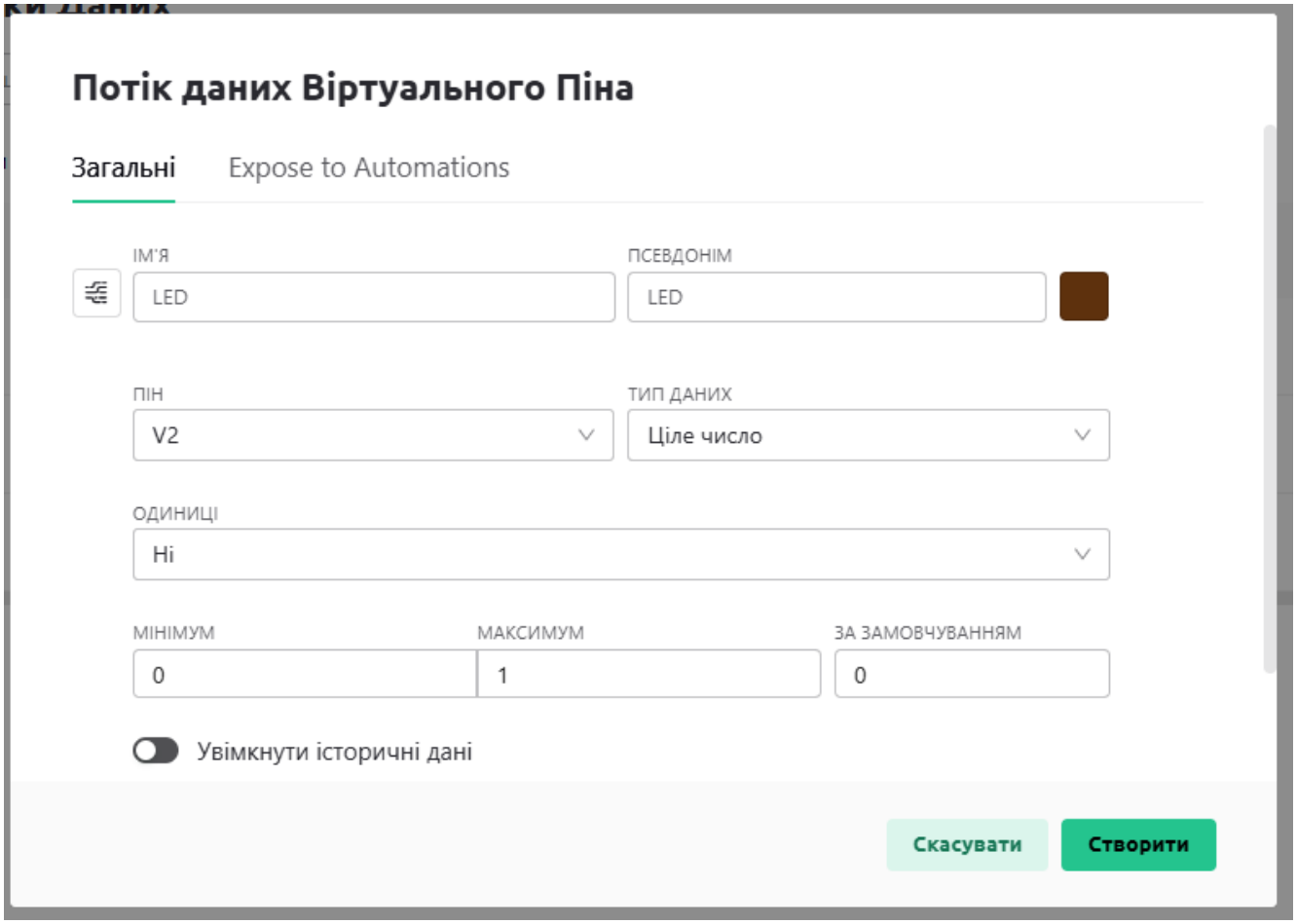


Рисунок 2.6 – Налаштування LED

Також було налаштовано перемикач, рис. 2.7., одного з основним елементів керування в системі. Він використовується для подачі команди на зміну стану виконавчого пристрою, зокрема реле, яке у свою чергу керує фізичним доступом - наприклад, електромагнітним замком. Значення, які були обрані під час налаштування:

– тип: Virtual;
– пін: V3;
– тип даних: Integer;
– мінімальне значення: 0;
– максимальне значення: 1;
– псевдонім: SwitchRelay.

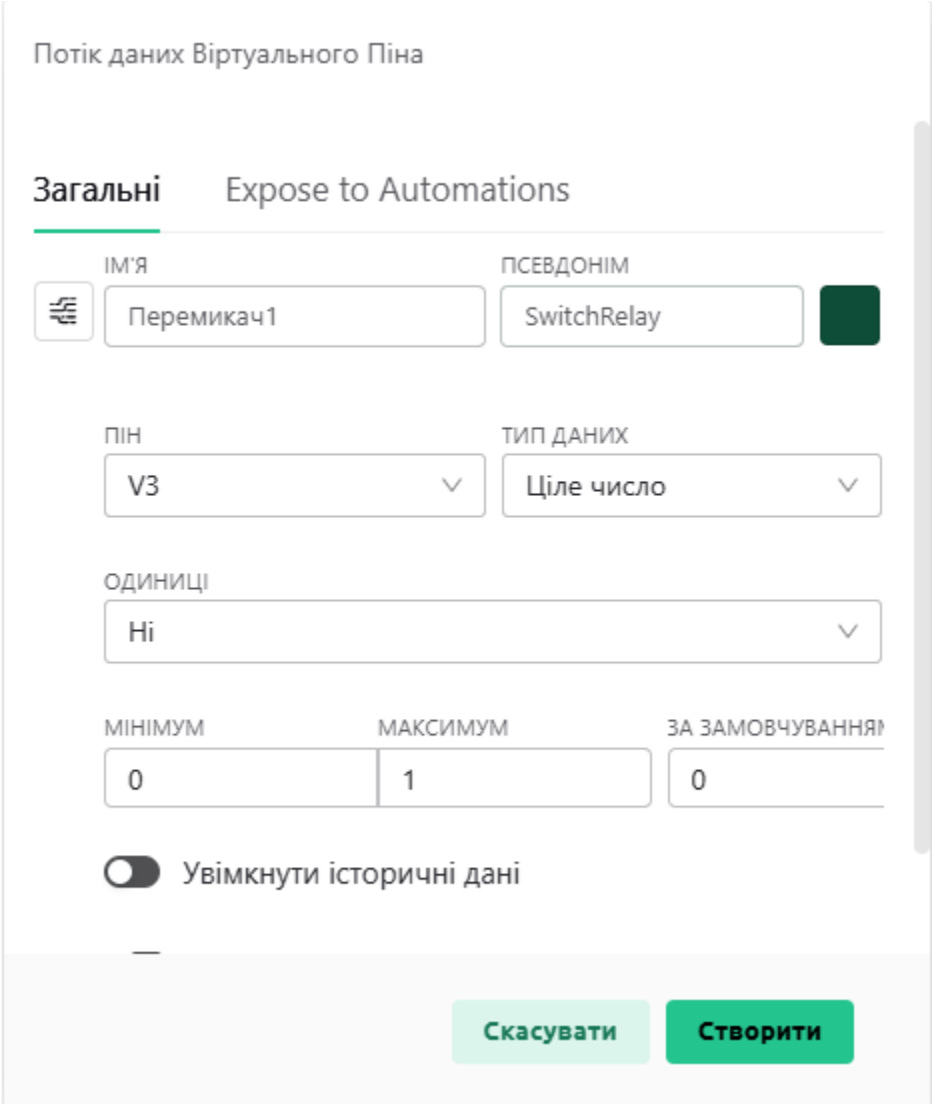


Рисунок 2.7 – Налаштування перемикача

Після налаштування потоків даних необхідно перейти до побудови графічного інтерфейсу у вкладці Web Dashboard, рис. 2.8. Тут додаються віджети, які взаємодіють із визначеними потоками. Для реалізації можливості керування доступом додається перемикач, який прив’язується до потоку даних. Цей елемент дозволяє вмикати або вимикати відповідний фізичний пін контролера, а отже, і керувати станом реле. Далі додається віджет типу Gauge, що відображає аналогові значення, які надходять із сенсора, підключеного до аналогового входу мікроконтролера. Цей віджет зв’язується з потоком analog і дозволяє візуально оцінювати зміну параметрів у режимі реального часу. Також у інтерфейс додаються світлодіодні індикатори, які прив’язуються до потоків V2 і V4 та дозволяють швидко оцінити поточний стан системи: активний, неактивний, доступ дозволено

або заборонено. Отже, формується інтуїтивно зрозумілий інтерфейс керування, який відображається як у вебзастосунку, так і синхронізується з мобільним додатком Blynk.

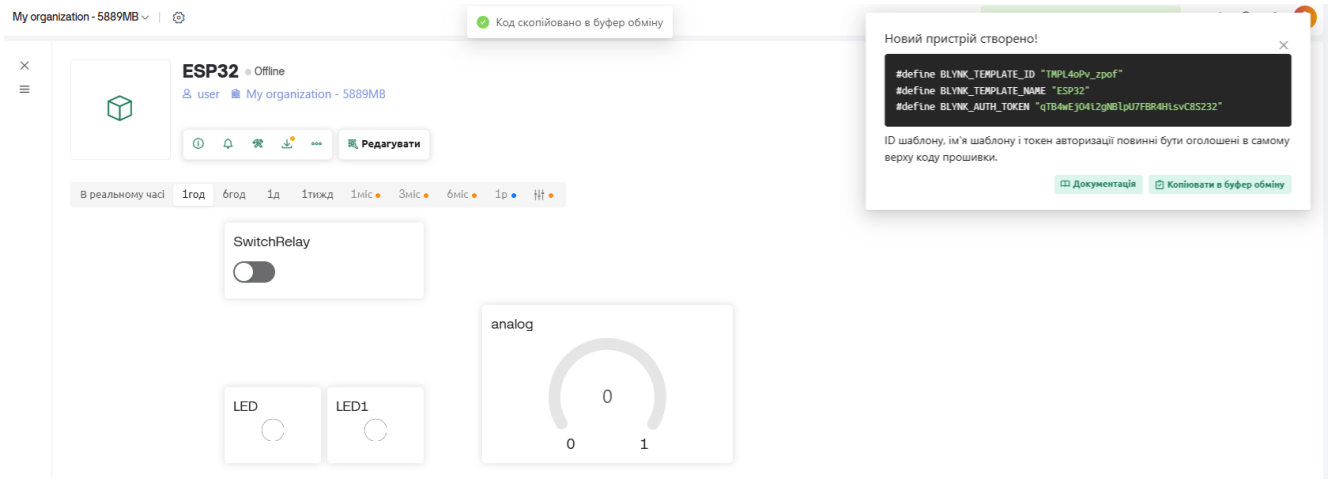


Рисунок 2.8 – Налаштування Web Dashboard

У результаті було створено повноцінна панель керування, яка дозволяє виконувати такі дії: увімкнення/вимкнення реле, спостереження за значеннями з аналогового сенсора, відстеження стану пристроїв за допомогою віртуальних індикаторів, і все це — у режимі реального часу. Такий підхід до побудови інтерфейсу дає змогу максимально адаптувати систему під конкретні потреби користувача та забезпечити високу функціональність при мінімальних витратах часу на розробку.

Також було налаштовано Dashboard у мобільному застосунку Blynk IoT. Після створення шаблону пристрою та Datastream у вебверсії платформи, його функціональність автоматично доступна і в мобільному застосунку без необхідності повторного налаштування логіки. Відповідна схема зображена на рис. 2.9. Для використання його функціоналу необхідно створити пристрій на основі вже існуючого шаблону. Це виконується через інтерфейс додатку шляхом вибору пункту створення нового пристрою та подальшого вибору шаблону зі списку доступних. Після створення пристрою користувачу відкривається можливість редагування мобільного Dashboard.

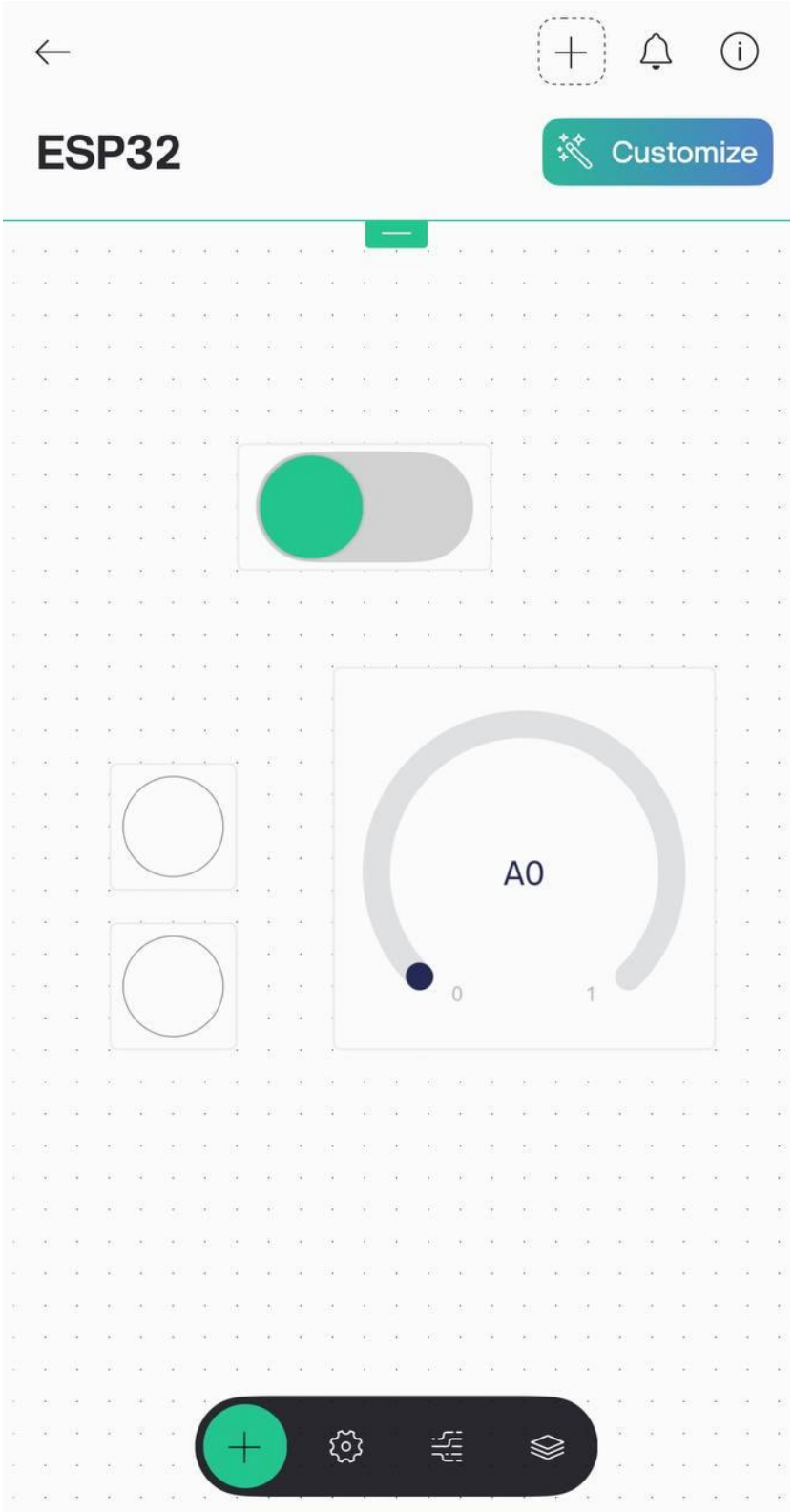


Рисунок 2.9 – Налаштування Dashboard у мобільному застосунку

У режимі редагування до інтерфейсу додається графічний компонент – перемикач, яка відповідає за керування замком. Цей компонент прив’язується до попередньо створеного в шаблоні datastream типу «Virtual Pin» з номером. У налаштуваннях компонента можна також вказати його назву, змінити іконку,

вибрати кольори для активного та неактивного стану та перевірити, чи правильно встановлено прив’язку до потрібного віртуального піну.

Після завершення налаштувань зміни зберігаються, і мобільний інтерфейс переходить у робочий режим. Тепер, при натисканні на кнопку або перемикач у мобільному застосунку, сигнал передається до мікроконтролера через хмарний сервер Blynk, і відбувається відповідне вмикання або вимикання реле, яке у свою чергу керує електрозамком.

Завдяки єдиному шаблону пристрою, який спільно використовує як вебінтерфейс, так і мобільний додаток, забезпечується зручне керування системою доступу з будь-якого пристрою, що підтримує платформу Blynk.

Загальна схема роботи логіки в даному випадку буде мати вигляд, як на рис. 2.10. Ця схема ілюструє логіку роботи прошивки, яка виконується на мікроконтролері ESP8266, що керує електрозамком через реле. Вона враховує підключення до Wi-Fi, ініціалізацію Blynk, обробку команд користувача з мобільного додатку, а також фізичне увімкнення/вимкнення реле.

Спершу відбувається запуск програми після подачі живлення на контролер. Далі система встановлює з’єднання з локальною мережею Wi-Fi, після чого відбувається ініціалізація та автентифікація через Blynk-сервер з використанням спеціального токена. Після цього здійснюється перевірка успішності з’єднання з Blynk, і в разі невдачі система повторює спробу підключення, а у випадку успіху — переходить до основного циклу роботи. У цьому циклі мікроконтролер очікує команду від користувача, яка надходить через мобільний додаток Blynk. Коли команда отримана, система зчитує її значення: якщо воно дорівнює одиниці, реле вмикається, що активує електрозамок, а якщо нулю — реле вимикається. Після обробки команди система повертається до очікування наступної команди. Теоретично програма може мати кінцеву точку завершення, проте на практиці вона працює безперервно, забезпечуючи постійний контроль над станом реле та можливість дистанційного керування замком.

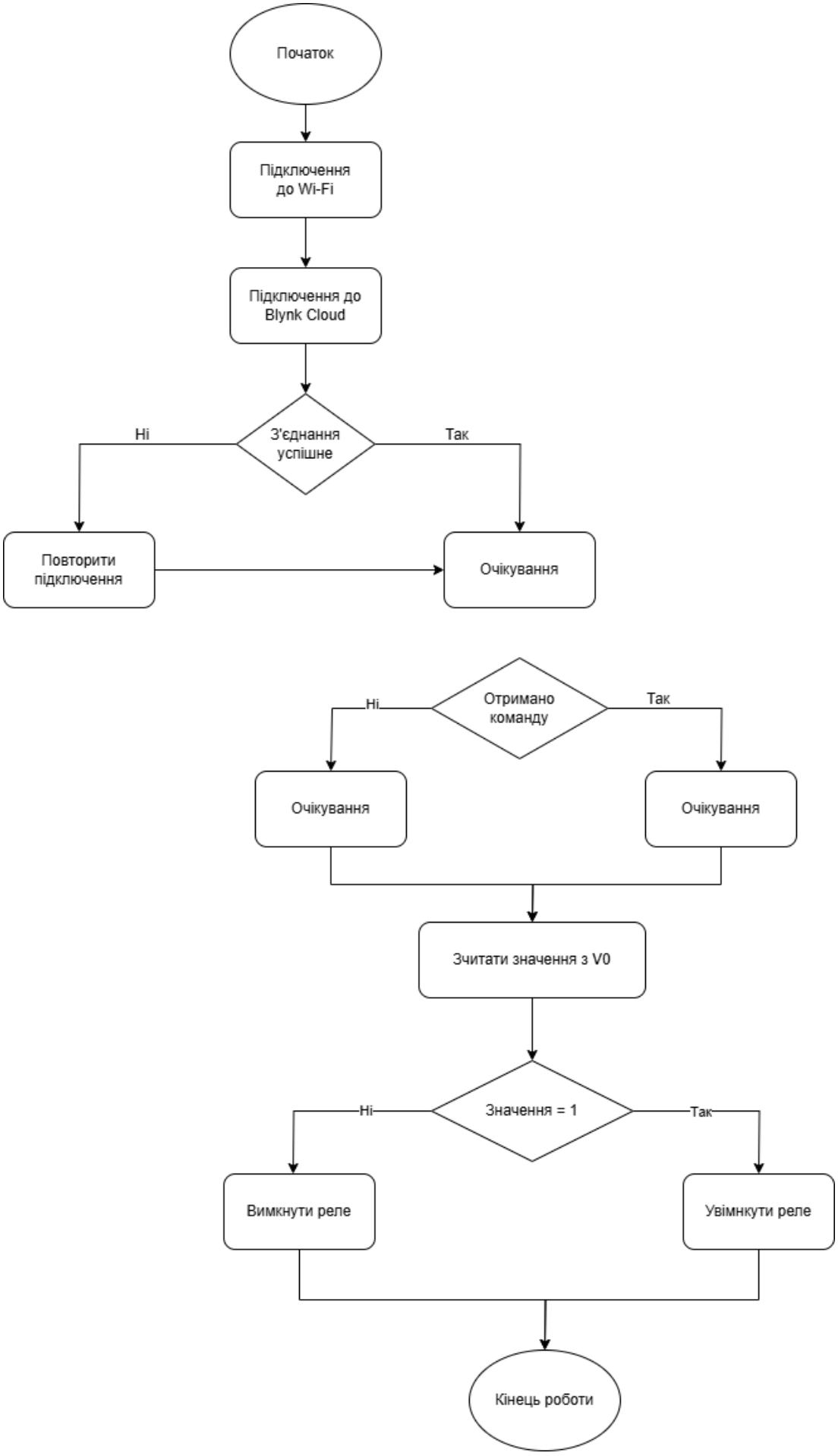


Рисунок 2.10 – Схема роботи алгоритму

2.3 Програмна частина система

Програмна частина розробленої системи допуску є основним функціональним компонентом, оскільки саме вона забезпечує реалізацію логіки роботи пристрою, його взаємодію з користувачем через віртуальний інтерфейс, обробку даних із сенсорів, керування виконавчими елементами, а також встановлення стійкого з'єднання з хмарною інфраструктурою. Реалізація програмного забезпечення виконується у середовищі Arduino IDE, що є зручним кросплатформним інструментом для розробки, компіляції та завантаження програмного коду до мікроконтролерів. У даному проєкті використовується плата Wemos D1 mini на базі модуля ESP8266, яка підтримується Arduino IDE через додатково встановлений пакет плати ESP8266 [38].

На початковому етапі у середовищі Blynk було налаштовано потоки даних (Datastreams), які встановлюють зв’язок між інтерфейсом користувача та пін-контактами мікроконтролера. Першим було створено потік для керування реле, що відповідає за фізичне відкривання або блокування доступу. Для цього було обрано тип потоку Virtual Pin, із закріпленням за віртуальним піном V1, встановлено тип даних Integer (ціле число), мінімальне значення — 0, максимальне — 1, а також призначено псевдонім: relay. Таким чином, сигнал, надісланий із віртуального інтерфейсу на пін V1, викликає відповідну дію на фізичному піні контролера, до якого підключено реле.

Аналогічно було налаштовано потік для зчитування значень з аналогового сенсора, у ролі якого виступає потенціометр. Було обрано тип потоку Analog, прив’язано до аналогового входу A0, встановлено тип даних Integer, мінімальне значення — 0, максимальне — 1, а також надано псевдонім: analog. Цей потік забезпечує постійне надсилання даних до інтерфейсу, де вони візуалізуються за допомогою графічного елемента Gauge.

Для візуальної індикації стану доступу були створені два потоки типу Virtual Pin (рис. 2.6). Один із них пов’язаний з піном V2, що використовується для сигналізації заблокованого доступу, інший — з піном V4, що сигналізує про дозволений доступ. Обидва потоки мають тип даних Integer, діапазон значень —

від 0 до 1, і спільний псевдонім: LED. У графічному інтерфейсі ці потоки пов’язані з LED-віджетами, які змінюють колір або стан залежно від поточної логіки програми.

Ключовим елементом керування у системі є перемикач (Switch), що дозволяє користувачеві вручну змінювати стан пристрою, зокрема вмикати або вимикати реле (рис. 2.7). Для перемикача був створений потік з параметрами: тип: Virtual, пін: V3, тип даних: Integer, мінімальне значення: 0, максимальне: 1, псевдонім: SwitchRelay. Саме цей пін використовується в коді програми як тригер для зміни стану виконавчого елемента на платі.

Усі зазначені потоки реалізовані у коді за допомогою відповідних конструкцій. Для підключення необхідних бібліотек застосовуються наступні директиви:

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SimpleTimer.h>
```

Ідентифікація пристрою в інфраструктурі Blynk забезпечується такими макровизначеннями:

```
#define BLYNK_TEMPLATE_ID "TMPL4oPv_zpof"
#define BLYNK_DEVICE_NAME "ESP8266"
#define BLYNK_AUTH_TOKEN " elzaQzSrABhYunRaZKpOr_FGZyQIrYE5"
```

Для з’єднання з Wi-Fi-мережею оголошуються наступні змінні:

```
char ssid[] = «khnu.km.ua»;
char pass[] = «te3s4q1»;
```

Функція setup() ініціалізує підключення:

```
void setup()
{
  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, pass);
}
```

Фізичне реле підключається до піна D1, який оголошується так:

```
#define RELAY_PIN D1
void setup()
```

```
{
  pinMode(RELAY_PIN, OUTPUT);
}
```

Функція обробки команди з перемикача виглядає наступним чином:

```
BLYNK_WRITE(V3)
{
  digitalWrite(RELAY_PIN, HIGH);
}
```

Зчитування аналогового значення виконується за допомогою:

```
int analogValue = analogRead(A0);
Blynk.virtualWrite(V5, analogValue);
```

Для керування віртуальними індикаторами у коді оголошується об’єкт типу WidgetLED:

```
WidgetLED ledBlocked(V2);
WidgetLED ledAccess(V4);
```

Подальша логіка виглядає так:

```
if (analogValue > 500)
{
  ledAccess.on();
  ledBlocked.off();
} else {
  ledAccess.off();
  ledBlocked.on();
}
```

Основний цикл програми забезпечує зв’язок з сервером та обробку подій:

```
void loop()
{
  Blynk.run();
  timer.run();
}
```

Отже, уся програмна частина побудована на принципах обробки подій,

періодичного опитування сенсорів та двосторонньої взаємодії з хмарним сервісом, що дозволяє реалізувати зручну, адаптивну і стабільну систему контролю доступу на основі Blynk IoT.

2.4 Підключення елементів

Для реалізації апаратної частини системи доступу було обрано мінімально необхідний набір елементів, які дозволяють протестувати функціональність керування пристроєм, зчитування аналогових сигналів та індикацію стану системи. У процесі розробки системи допуску було виконано логічне зіставлення між функціональними компонентами пристрою та фізичними контактами мікроконтролера ESP8266. Це дозволило визначити конкретні пін-контакти, через які здійснюється керування реле, світлодіодами та зчитування аналогових даних. У таблиці 2.1 нижче наведено відповідність між функціональними елементами та пін-контактами мікроконтролера ESP8266.

Таблиця 2.1 – Відповідність функціональним елементам та пін-контактам мікроконтролера ESP8266

Елемент	Пін на ESP8266
Реле	D1 (GPIO5)
Світлодіод 1	D0
Світлодіод 2	D2
Аналоговий датчик	A0

Керування реле здійснюється через цифровий пін D1, який активується відповідно до команд, що надходять із мобільного або вебінтерфейсу платформи Blynk. У програмному коді цей пін оголошено як вихід, і зміна його стану приводить до відповідного перемикання реле, яке подає або вимикає напругу на виконавчий пристрій (наприклад, електромагнітний замок).

Для реалізації візуального зворотного зв’язку у системі використано два світлодіоди, підключених до цифрових пінів D0 та D2. Перший (D0) сигналізує про стан блокування доступу — наприклад, коли система забороняє вхід. Другий (D2) спрацьовує при дозволеному доступі, вказуючи користувачеві на успішне проходження перевірки. Така індикація дозволяє швидко оцінити поточний стан системи навіть без доступу до інтерфейсу.

До аналогового входу A0 підключено сенсор, що дозволяє зчитувати змінне значення напруги. У тестовій конфігурації використано потенціометр як аналоговий вхідний пристрій. Він дозволяє імітувати зміну навколишнього середовища або користувацький вплив. Отримані значення передаються в систему Blynk через віртуальний пін і відображаються на інтерфейсі у вигляді шкали (Gauge), що дозволяє користувачеві бачити поточний рівень сигналу в реальному часі.

Фізичне підключення компонентів здійснюється за допомогою макетної плати (breadboard), що значно спрощує монтаж, модифікацію та експериментальні налаштування без необхідності паяння. Такий підхід особливо корисний на етапі розробки та налагодження системи. У фінальному варіанті проєкту передбачається заміна макетної плати на друковану плату (PCB), яка забезпечить вищу надійність, компактність та стійкість конструкції до механічних впливів.

Живлення всієї системи здійснюється через USB-кабель або зовнішній блок живлення зі стабілізованою напругою 5 В. Такий спосіб підключення дозволяє забезпечити надійну та безпечну роботу як самого мікроконтролера, так і підключених до нього периферійних пристроїв. У випадку використання виконавчих пристроїв із підвищеним споживанням струму — таких як електромеханічні замки або потужні реле — рекомендується використовувати зовнішнє джерело живлення, здатне видавати струм не менше 1 А. Це особливо важливо у разі тривалого активного використання системи або встановлення її в умовах промислової або побутової експлуатації.

Наявність окремого та стабільного джерела живлення є обов’язковою умовою для забезпечення безперебійної роботи системи, адже у разі нестачі напруги або просідання струму можливе «зависання» контролера або некоректна

робота периферії. Такий ризик є неприйнятним у системах безпеки, тому під час впровадження рішення в реальне середовище особливу увагу слід приділяти стабільності живлення.

З метою захисту даних і запобігання несанкціонованому доступу до системи керування було передбачено низку заходів безпеки. Насамперед, з'єднання між мобільним або вебзастосунком Blynk, сервером та фізичним пристроєм здійснюється через захищений протокол SSL/TLS, що забезпечує надійне шифрування переданих даних. Це виключає можливість перехоплення або модифікації трафіку зловмисниками.

Також реалізується використання унікальних токенів авторизації, які автоматично генеруються системою Blynk для кожного пристрою під час створення шаблону. Цей токен є обов'язковим елементом для ініціалізації з'єднання та виконує роль цифрового ключа автентифікації. Без нього встановлення зв'язку між пристроєм та сервером є неможливим.

Крім того, підключення пристрою обмежується лише захищеною локальною Wi-Fi мережею, що має встановлений пароль доступу. Це дозволяє виключити можливість підключення сторонніх пристроїв або неавторизованих користувачів. Додатковим рівнем захисту є авторизація самого користувача в мобільному застосунку Blynk, що передбачає використання облікового запису, реєстрацію, верифікацію поштової адреси та, за потреби, застосування двофакторної автентифікації.

У сукупності, всі ці заходи дозволяють реалізувати захищену інфраструктуру віддаленого доступу, в якій навіть за керування з іншого міста або мережі дані та команди передаються через захищений канал, відповідно до сучасних вимог кібербезпеки в IoT-системах.

2.5 Висновок до розділу

У другому розділі роботи було детально проаналізовано та послідовно реалізовано основні етапи проєктування апаратної та програмної частин системи

доступу на базі платформи Blynk IoT. Проведений всебічний аналіз і обґрунтування вибору ключових компонентів дозволили сформувати оптимальну архітектуру системи, що відповідає сучасним вимогам щодо надійності, функціональності та масштабованості IoT-рішень. Завдяки цьому система отримала міцний фундамент для подальшого розвитку й впровадження в реальних умовах.

В апаратній частині було зроблено свідомий вибір на користь плати Wemos D1 mini на основі мікроконтролера ESP8266. Ця плата завдяки своїм компактним розмірам, підтримці Wi-Fi-зв'язку та сумісності з широко розповсюдженим середовищем Arduino IDE стала найкращим рішенням для поставлених завдань. Для забезпечення керування виконавчими механізмами були інтегровані реле, для збору аналогових даних — відповідні сенсори, а для візуального відображення стану системи — світлодіодні індикатори. Кожен компонент було ретельно підібрано з урахуванням його надійності, простоти інтеграції, а також потенціалу для майбутнього розширення функціональних можливостей системи.

З боку програмного забезпечення було обґрунтовано та реалізовано використання платформи Blynk, яка надала не лише готову хмарну інфраструктуру, але й зручний візуальний конструктор інтерфейсу, бібліотеки для програмування мікроконтролерів, а також можливість дистанційного управління системою через мобільний застосунок. У процесі розробки було створено шаблон пристрою в середовищі Blynk, реалізовано програмні модулі для керування реле, зчитування аналогових сигналів і візуалізації актуального стану системи. Конфігурація інтерфейсу забезпечила повну інтеграцію з мобільним додатком, що гарантує зручність та оперативність контролю системи в режимі реального часу.

Підсумком другого розділу стало повне впровадження апаратної платформи, розгортання системи керування на базі Blynk Dashboard, налаштування усіх необхідних програмних компонентів та створення функціонального користувацького інтерфейсу.

3 ТЕСТУВАННЯ СИСТЕМИ ТА РЕЗУЛЬТАТИ РОЗРОБКИ

3.1 Методики тестування системи

Після завершення етапів проєктування та реалізації апаратної і програмної частин системи допуску виникла нагальна потреба в її всебічному тестуванні. Метою цього етапу стало всебічне перевірення працездатності системи у реальних або максимально наближених до реальних умовах експлуатації. Особлива увага приділялася виявленню потенційних збоїв, помилок у програмному коді, затримок у передачі команд, а також оцінці загальної ефективності розробленого рішення.

Тестування базувалося на заздалегідь розроблених сценаріях, які максимально охоплювали ключові функціональні можливості системи. Основні сценарії включали:

- керування реле через мобільний застосунок;
- візуальне відображення стану виконавчих пристроїв за допомогою світлодіодів та віджетів інтерфейсу;
- зчитування аналогових даних із сенсора та їх коректне відображення у мобільному додатку;
- синхронізацію дій і станів між апаратною платформою та хмарним середовищем Blynk у реальному часі.

Для забезпечення об’єктивності та системності оцінювання ефективності роботи системи були визначені кілька ключових критеріїв:

- час реакції системи — вимірювався як проміжок часу від моменту надсилання команди користувачем через застосунок до фактичного виконання дії виконавчим пристроєм (реле);
- стабільність зв’язку — оцінювалась за відсутністю переривань та втрати пакету даних між контролером Wemos D1 mini та сервером Blynk під час тривалих сесій тестування;
- точність зчитування сенсорних даних — перевірялася шляхом порівняння значень, отриманих мобільним додатком, з фактичними параметрами на аналоговому вході контролера;
- синхронізація інтерфейсу — аналізувалася відповідність візуального

відображення стану у віджетах мобільного застосунку до реального фізичного стану пристроїв;

– надійність керування реле — вимірювалась як кількість успішних активацій/деактивацій виконавчих пристроїв у відповідь на команди користувача.

Тестування проводилося у локальній Wi-Fi мережі з підключенням плати Wemos D1 mini до хмарного сервера Blynk, а також до периферійних пристроїв — реле, світлодіодів та аналогового сенсора. Керування системою здійснювалось за допомогою мобільного застосунку Blynk IoT, встановленого на смартфон.

Кожен сценарій тестування було повторено кілька разів з фіксацією та аналізом отриманих результатів, що дозволило оцінити стабільність роботи системи та її ефективність у різних умовах. Особливу увагу було приділено точності та швидкості відгуку реле, коректності відображення аналогових значень у віджетах типу Gauge, а також поведінці віртуальних індикаторів при зміні вхідних даних сенсора.

Крім того, у процесі тестування моделювалися типові ситуації реального використання системи, такі як:

– віддалене відкривання дверей через мобільний застосунок;

– надсилання послідовних команд з різною частотою;

– тимчасове відключення та повторне підключення Wi-Fi мережі;

– зміна параметрів сенсора (обертання потенціометра) для перевірки реакції системи на динамічні вхідні дані.

Ці випробування дали змогу не лише підтвердити функціональність системи, а й оцінити її відмовостійкість та поведінку у нестандартних умовах, що є надзвичайно важливим для впровадження IoT-рішень у реальному житті.

Отже, розроблена методика тестування охоплює як функціональні, так і технічні аспекти роботи IoT-системи доступу, надаючи об’єктивну оцінку її придатності для практичного використання. Детальний аналіз результатів тестування і висновки щодо ефективності розробленої системи будуть наведені у подальших розділах цієї роботи.

3.2 Випробування

Після розробки методики тестування було проведено серію практичних випробувань системи допуску, які повністю відповідали типовим сценаріям її майбутнього використання. Головною метою цих випробувань була перевірка функціональної адекватності розробленої системи в умовах, максимально наближених до реального середовища експлуатації, а також виявлення можливих помилок чи нестиковок у програмній логіці, апаратній взаємодії компонентів або в комунікації між різними частинами системи.

Для комплексного тестування було обрано два основних сценарії, що відображають ключові варіанти поведінки системи в реальних умовах, схематично які зображено на рис. 3.1:

– сценарій успішного доступу, коли авторизований користувач має право відкрити доступ до приміщення;

– сценарій заблокованого доступу, який активується у разі спрацювання заздалегідь заданих умов, що забороняють вхід.

У сценарії успішного доступу користувач, авторизований через мобільний застосунок Blynk, ініціює відкриття доступу шляхом натискання на відповідний перемикач в інтерфейсі додатку. Команда надходить на хмарний сервер Blynk, який забезпечує маршрутизацію даних та синхронізацію між користувачем і пристроєм. Далі команда передається на мікроконтролер Wemos D1 mini, де відбувається активація цифрового виходу D1 (GPIO5).

Цей вихід безпосередньо керує реле, підключеним до виконавчого механізму – електромагнітного замка. Реле, отримавши сигнал, замикає або розмикає електричний ланцюг, що призводить до відкриття дверей.

Одночасно з цим, у мобільному додатку змінюється стан перемикача, з’являється віртуальний світлодіод, який інформує користувача про успішне відкриття доступу. На апаратному рівні — на самій платі — загоряється фізичний світлодіод, що дублює статус системи. Такий подвійний механізм індикації забезпечує зворотній зв’язок як на відстані, так і локально.

За результатами вимірювань, час реакції системи в цьому сценарії становив

у середньому від 0,5 до 1 секунди. Це є цілком прийнятним показником для систем дистанційного керування, оскільки гарантує оперативність дії без зайвих затримок, що є важливим для безпеки і зручності користувача.

Система продемонструвала високу стабільність: не було зафіксовано жодних випадків втрати зв’язку, затримок у передачі команд або помилкових спрацьовувань. Реле надійно виконувало усі отримані команди, а інтерфейс Blynk забезпечував миттєвий зворотний зв’язок у реальному часі.

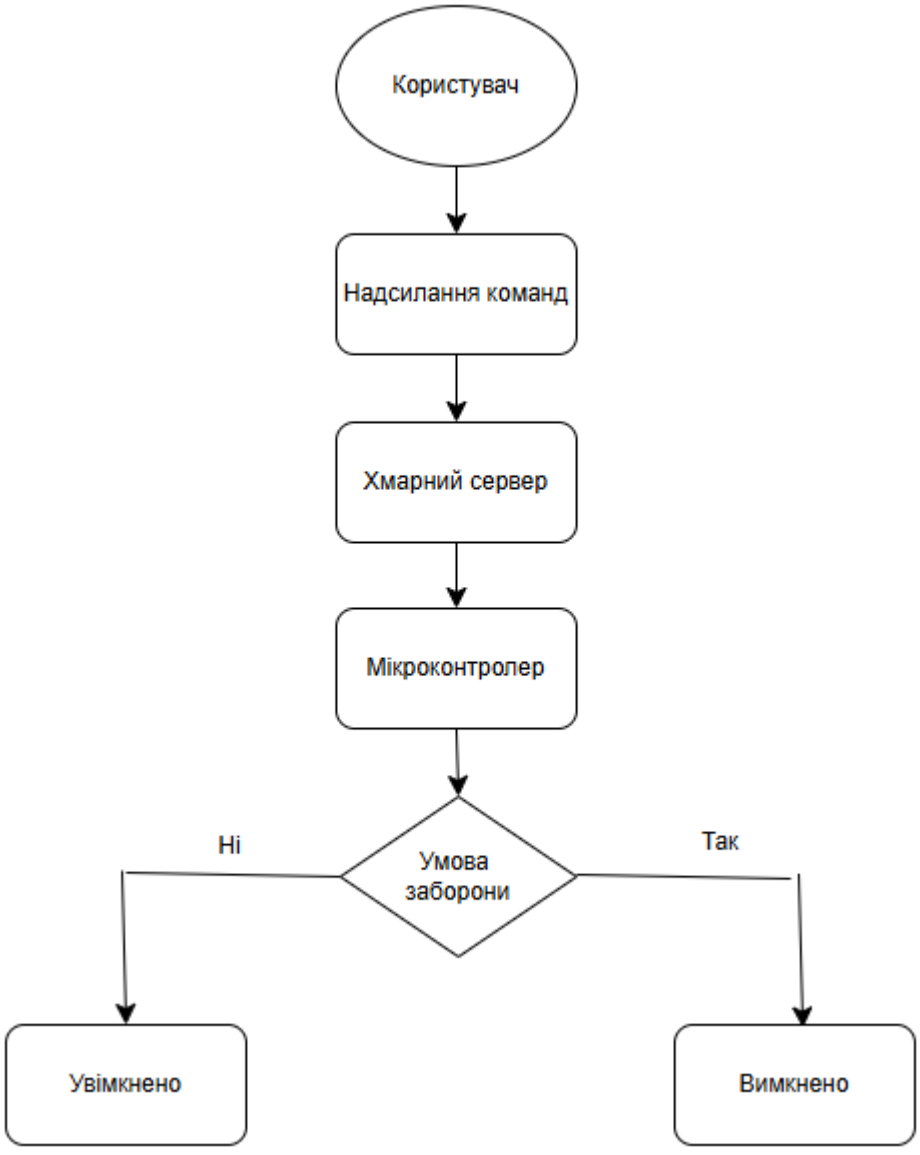


Рисунок 3.1 – Схема сценарію

Другий сценарій тестування імітує ситуацію, коли доступ до приміщення повинен бути заборонений через певні зовнішні або внутрішні умови, що

реєструються датчиками системи.

В якості тригера для блокування виступає аналоговий сенсор, підключений до аналогового входу A0 мікроконтролера. Для тестування використовувався потенціометр, який моделював змінний параметр навколишнього середовища, наприклад, рівень освітленості, температуру або інші контрольовані фактори.

У програмному коді мікроконтролера задавалося порогове значення сигналу, перевищення якого автоматично призводить до блокування подачі команди на реле. Значення, що зчитуються з аналогового входу, у режимі реального часу передавалися у додаток Blynk, де відображалися на спеціальному графічному віджеті Gauge.

Якщо показник перевищував поріг (наприклад, більше 500 одиниць), система автоматично забороняла відкриття замка: реле залишалося у вимкненому стані, а віртуальний індикатор змінював свій колір або деактивувався, сигналізуючи про відсутність доступу.

Цей сценарій імітує типову ситуацію, коли система реагує на небезпечні або критичні умови, наприклад, підвищену температуру, порушення режиму охорони, детекцію руху або інші фактори, які можуть свідчити про загрозу безпеці.

Всі виміри передавалися з мінімальною затримкою, а реакція системи була цілком автоматизованою та безпомилковою. Таким чином, забезпечувалося надійне запобігання доступу у критичних випадках.

У ході тестування обох сценаріїв система демонструвала стабільну, передбачувану та синхронізовану роботу всіх своїх компонентів. Інтерфейс мобільного додатку швидко реагував на всі зміни, показники на віджетах відповідали фактичним параметрам, а фізичні елементи (реле, світлодіоди) виконували свої функції коректно.

Взаємодія між мобільним застосунком, хмарним сервером і мікроконтролером була налагоджена ефективно і без збоїв, що свідчить про продуманість і якість реалізації всього технічного рішення.

Отже, результати випробувань підтвердили правильність та ефективність реалізованої логіки системи допуску, що успішно виконує основні функції — надання або блокування доступу відповідно до команд користувача чи визначених

умов. Це свідчить про готовність системи до подальшого впровадження у реальні умови з можливістю подальшого масштабування і вдосконалення.

3.3 Аналіз роботи системи

Після проведення тестування та випробувань, описаних у попередніх підрозділах, було здійснено аналіз роботи всієї системи допуску на основі сервісу Blynk IoT. Метою аналізу стало оцінювання ключових технічних характеристик, стабільності, точності, відгуку, інтерактивності інтерфейсу та загальної ефективності реалізованого рішення в умовах практичного застосування.

Одним із основних показників функціональності системи є швидкість реакції на команди користувача. Вимірювання показали, що час затримки між дією у мобільному застосунку (наприклад, натискання перемикача) та фактичною реакцією фізичного пристрою (спрацювання реле, зміна стану індикатора) становить у середньому менше 1 секунди. Це забезпечує комфортне керування у реальному часі та свідчить про достатню швидкодію як платформи Blynk, так і апаратного контролера ESP8266.

У процесі тривалого тестування не було зафіксовано розривів з’єднання між мікроконтролером і Blynk-сервером, що свідчить про високу стабільність з’єднання. Навіть при короткочасному вимкненні Wi-Fi система автоматично відновлювала зв’язок без необхідності перезавантаження пристрою. Це важливий показник, оскільки система доступу має працювати надійно в автономному режимі без втручання користувача.

Щодо зчитування аналогових значень, система демонструє точну передачу даних від сенсора до інтерфейсу. Значення з аналогового входу (A0) стабільно відображались на віджеті Gauge у Blynk, без ривків або спотворень. Поріг блокування доступу на основі аналогового значення працював коректно, що підтверджує правильність реалізації програмної логіки та обробки вхідних сигналів.

Віртуальні пін-контакти, до яких було прив’язано LED-індикатори, чітко

відображали стан системи. За результатами аналізу не виявлено випадків несинхронізованої індикації між фактичним станом пристрою та його відображенням у мобільному інтерфейсі. Це свідчить про правильну взаємодію програмного коду з Blynk-бібліотекою та ефективне використання віртуальних каналів.

Особливо варто відзначити гнучкість і масштабованість створеного рішення. Завдяки архітектурі Blynk, система легко адаптується до розширення: можна додавати нові сенсори, додаткові пристрої, елементи безпеки (наприклад, камери або зчитувачі RFID), при цьому не змінюючи загальної логіки або структури системи. Інтерфейс можна змінювати онлайн — без перепрошивки пристрою, що також підвищує зручність експлуатації.

Під час аналізу також було зафіксовано високий рівень зручності та доступності для кінцевого користувача. Мобільний застосунок має інтуїтивно зрозумілий інтерфейс, не потребує ручного налаштування з'єднання чи введення команд. Керування доступом здійснюється натисканням однієї кнопки, а всі зміни відображаються у реальному часі.

Аналіз результатів роботи підтверджує, що створена система допуску є стабільною, функціонально повною, швидкодіючою та зручною у використанні. Вона забезпечує базову безпеку доступу та дає змогу легко масштабувати функціонал у майбутньому. Отримані результати повністю відповідають поставленим на початку роботи вимогам до системи.

3.4 Оцінка відповідності

На етапі постановки задачі до розроблюваної системи допуску було сформульовано чіткий перелік вимог, які відображали ключові аспекти її майбутньої функціональності, надійності, зручності у використанні та здатності адаптуватися до умов реальної експлуатації. Ці вимоги слугували орієнтиром для всього процесу розробки та впровадження системи, а також стали базою для подальшого тестування та оцінки кінцевого продукту. У цьому підрозділі

здійснюється детальне порівняння очікуваних характеристик системи з фактичними результатами, отриманими під час комплексних випробувань, що дає змогу об’єктивно оцінити ефективність та практичну цінність розробленого рішення.

Однією з найважливіших функціональних вимог було забезпечення дистанційного керування доступом через хмарний сервіс. Це означало, що користувач має мати можливість віддалено відкривати або блокувати доступ за допомогою мобільного додатку, незалежно від місцезнаходження, без необхідності фізичної присутності біля пристрою. Розроблена система повністю відповідає цій вимозі: інтерфейс користувача у мобільному застосунку Blynk створено з урахуванням інтуїтивної зручності, використано візуально зрозумілі віджети, що дозволяють миттєво керувати станом реле. Практичні випробування показали, що час від надсилання команди до її виконання контролером становить у середньому менш ніж одну секунду, що підтверджує високу швидкодію та стабільність мережевого з’єднання, що є критично важливим для систем контролю доступу.

Ще однією суттєвою вимогою було наявність чіткої індикації стану системи — як у фізичній формі, так і у віртуальному середовищі. Апаратна індикація реалізована за допомогою світлодіодів, які сигналізують про основні стани пристрою: живлення, активність реле, стан доступу тощо. Програмна індикація здійснюється через віртуальні LED у Blynk, які в реальному часі відображають ті ж самі параметри. Такий двосторонній підхід до індикації дозволяє користувачу отримувати повну картину стану системи як на місці, так і дистанційно. Під час тестування підтверджено, що індикація працює коректно, даючи можливість оперативно реагувати на зміни, що є важливим для забезпечення безпеки і зручності використання.

Важливим технічним завданням була реалізація функції моніторингу аналогових сигналів, що розширює можливості системи за рахунок збору додаткових даних про навколишнє середовище або специфічні параметри об’єкта контролю. Наприклад, це можуть бути показники температури, освітленості, вологості тощо. У системі реалізовано зчитування аналогового сигналу з сенсора без помітних затримок, а результати у вигляді графічного віджета Gauge

відображаються у мобільному додатку. Крім того, реалізована логіка блокування доступу при перевищенні встановленого порогового значення, що забезпечує гнучкість і можливість налаштування системи під різні сценарії безпеки. Ця функція є важливим кроком до адаптивності системи і підвищення її корисності в реальних умовах.

Ще одним критерієм успішності проєкту стала простота налаштування та обслуговування системи. Вибір платформи Blynk повністю виправдав очікування: весь процес налаштування — від створення шаблону пристрою, розробки користувацького інтерфейсу, додавання необхідних компонентів до підключення апаратної частини — є максимально простим і не вимагає глибоких знань у програмуванні чи мережевих технологіях. Це робить систему доступною не лише для фахівців, але й для аматорів, викладачів та студентів, що сприяє її широкому використанню в навчальних та дослідницьких проєктах.

Крім того, система повністю відповідає вимогам модульності та масштабованості. Завдяки використанню відкритих протоколів і гнучкій архітектурі, до неї можна легко додавати нові пристрої — як апаратні модулі (реле, датчики), так і програмні компоненти (нові сценарії керування, розширена логіка). Це відкриває можливості для розширення функціоналу та інтеграції у більші за масштабом IoT-системи, що дозволяє застосовувати розроблене рішення в різних сферах: від контролю доступу у житлових і офісних приміщеннях до аграрного сектору, промислового моніторингу чи охоронних систем.

Отже, на основі проведених комплексних випробувань, функціонального тестування та аналітичного порівняння очікуваних та фактичних результатів можна впевнено стверджувати, що всі головні технічні та функціональні вимоги до системи допуску були виконані у повному обсязі. Розроблена система є стабільною, надійною, зручною у використанні та готовою для впровадження в реальних умовах експлуатації. Водночас вона має достатній потенціал для подальшого розвитку, що робить її перспективною платформою для широкого кола практичних застосувань у сфері Інтернету речей.

3.5 Висновок до розділу

У третьому розділі було проведено всебічне тестування розробленої системи допуску та здійснено оцінку її ефективності відповідно до поставлених функціональних і технічних вимог. На основі розробленої методики тестування були реалізовані типові сценарії використання системи — як з дозволеним, так і з обмеженим доступом. У ході випробувань система продемонструвала стабільну роботу, високу точність зчитування даних та оперативну реакцію на дії користувача.

У сценарії дозволеного доступу було підтверджено надійність передачі команд між мобільним застосунком Blynk і мікроконтролером ESP8266, а також чітке спрацювання реле, яке імітує електромеханічний замок. Сценарій блокування доступу, реалізований через контроль аналогового сигналу, продемонстрував ефективність логічної перевірки вхідних даних і здатність системи приймати рішення на основі заданих умов у реальному часі.

Аналіз роботи системи показав, що всі ключові функціональні модулі взаємодіють узгоджено: фізичні компоненти, програмна логіка та візуальний інтерфейс працюють синхронно. Час реакції системи становить менше однієї секунди, а відображення стану пристрою на інтерфейсі відбувається без затримок. Система стійка до короткочасних втрат з’єднання та не потребує втручання для відновлення роботи.

Також була проведена оцінка відповідності проєкту початковим вимогам. Встановлено, що система повністю реалізує необхідну функціональність: дистанційне керування доступом, зчитування сенсорних даних, візуальну індикацію, простоту конфігурації та можливість масштабування. Рішення є технічно ефективним і готовим до практичного використання, а завдяки використанню платформи Blynk — також зручним у налаштуванні та обслуговуванні.

Результати розробки підтверджують успішне досягнення цілей дипломної роботи, а створена система допуску може бути впроваджена як у побутових умовах, так і як прототип для подальших технічних удосконалень і досліджень у сфері IoT.

ВИСНОВКИ

У дипломній роботі на тему «Система допуску на основі сервісу Blynk IoT» було проведено комплексне дослідження, проєктування та створення прототипу інтелектуальної системи контролю доступу. Ця система поєднала сучасні апаратні засоби та програмні рішення з передовими хмарними технологіями, що дозволило реалізувати гнучке та масштабоване IoT-рішення для дистанційного керування доступом.

У першому розділі роботи було здійснено ґрунтовний аналітичний огляд ключових технологій, які лежать в основі Інтернету речей (IoT). Було детально розглянуто архітектуру IoT-систем, включно з особливостями побудови пристроїв із використанням мікроконтролерів із бездротовим зв'язком. Особлива увага приділялась протоколу MQTT, який забезпечує ефективний та надійний обмін повідомленнями між пристроями в мережі. Крім того, був проаналізований функціонал платформи Blynk IoT — хмарного сервісу, що надає інструменти для розробки, управління та моніторингу IoT-пристроїв через мобільні застосунки. В результаті проведеного аналізу було обґрунтовано вибір апаратної та програмної бази для реалізації системи дистанційного доступу: мікроконтролера ESP8266 (плата Wemos D1 mini) з підтримкою Wi-Fi та хмарного сервісу Blynk, що забезпечує зручний інтерфейс користувача і швидку інтеграцію.

Другий розділ роботи присвячений безпосередній розробці та впровадженню системи контролю доступу. На цьому етапі було здійснено детальний підбір необхідного апаратного забезпечення, що включало контролер Wemos D1 mini, модулі реле для управління виконавчими пристроями, світлодіодні індикатори для відображення стану системи, а також аналоговий сенсор для збору змінних даних. Після підключення та налаштування обладнання було розроблено програмну складову системи: створено шаблон пристрою у середовищі Blynk, реалізовано інтерфейс Web Dashboard із повним набором елементів керування та візуалізації. Завдяки цьому користувач отримав можливість у режимі реального часу контролювати доступ, отримувати інформацію з датчиків та отримувати миттєві

сповіщення про стан системи. Особливу увагу було приділено забезпеченню стабільної взаємодії між апаратною платформою і хмарною інфраструктурою, що гарантує надійність і швидкодію.

У третьому розділі було проведено всебічне тестування розробленої системи, яке включало перевірку як сценаріїв дозволеного доступу, так і випадків заборони доступу. Тестування проходило у максимально наближених до реальних умовах експлуатації, зокрема, в локальній Wi-Fi мережі, із застосуванням мобільного додатку Blynk IoT. Було перевірено коректність роботи всіх ключових функцій: керування реле, точність зчитування аналогових даних із сенсорів, швидкість реакції системи, стабільність з'єднання з хмарою та правильність відображення інформації у віджетах інтерфейсу. Проведений аналіз підтвердив стабільну роботу системи, високу швидкодію команд, точну індикацію стану пристроїв та коректну логіку реагування на різні події. Ці результати свідчать про те, що розроблене рішення повністю відповідає визначеним на початку роботи вимогам щодо функціональності, надійності, зручності використання та адаптивності.

Водночас у процесі тестування були виявлені перспективи для подальшого вдосконалення системи. Серед напрямків розвитку визначено можливість додавання RFID-модуля для безконтактної ідентифікації користувачів, інтеграції відеокамери для підвищення рівня безпеки, впровадження механізмів логування подій для аналізу доступу, розширення функціоналу захисту шляхом багаторівневих методів автентифікації, а також підключення нових типів сенсорів для більш комплексного моніторингу середовища.

У підсумку, результатом виконання дипломної роботи стало створення ефективної, надійної та зручної у використанні системи дистанційного доступу на базі платформи Blynk IoT. Розроблене рішення може бути застосоване не лише в навчальних цілях, а й служити фундаментом для впровадження в реальні об'єкти автоматизації, що потребують сучасних та гнучких систем контролю доступу. Це відкриває широкі можливості для подальших досліджень та практичної реалізації в галузі Інтернету речей.

ПЕРЕЛІК ДЖЕРЕЛ

1. What is System Access Control? precisely. URL: https://www.precisely.com/glossary/system-access-control (дата звернення: 20.02.2025).

2. What is access control and why is it vital? nedap. URL: https://www.nedapsecurity.com/insight/what-is-access-control/ (дата звернення: 20.02.2025).

3. Системи контролю і управління доступом від А до Я. deps. URL: https://deps.ua/ua/knowegable-base/reference-information/7824.html (дата звернення: 21.02.2025).

4. Що таке система контролю доступу і навіщо вона потрібна? Alarm Охоронні системи. URL: https://alarm.lviv.ua/blog/shho-take-sistema-kontrolyu-dostupu-i-navishho-vona-potribna (дата звернення: 21.02.2025).

5. Системи контролю й керування доступом. Відеокамери. URL: https://xn--80adgeboqrpy5j.com.ua/kontrol_dostypy/ (дата звернення: 21.02.2025).

6. Класифікація систем контрою доступу. Solis системи відеоспостереження. URL: http://solis.in.ua/klasyfikatsiya-system-kontrolyu-dostupu.html (дата звернення: 22.02.2025).

7. Системи контролю доступу СКД/СКУД. Укрінфосистеми. URL: https://ukrinfosystems.com.ua/uk/design-and-construction/access-control-systems (дата звернення: 22.02.2025).

8. Архітектура системи. Мережеві системи СКУД. URL: https://cyphrax.com/system-architecture/ (дата звернення: 22.02.2025).

9. Технічні засоби Інтернету речей : навч. посіб. для студ. спеціальності 171 «Електроніка», спеціалізації «Електронні системи мультимедіа та засоби Інтернету речей» / уклад.: Ю. О. Оникієнко, О. О. Титаренко. Київ : КПІ ім. Ігоря Сікорського, 2020. 124 с.

10. Історія виникнення. Інтернет речей. URL: https://itsinternetofthings.blogspot.com/p/blog-page_50.html (дата звернення: 28.02.2025).

11. Machine to Machine — M2M. MultiTech. URL: https://multitech.com/iot-wiki/machine-to-machine-m2m/ (дата звернення: 28.02.2025).

12. Жураковський Б. Ю., Зенів І. О. Технології інтернету речей : навчальний посібник для здобувачів ступеня бакалавра за освітньою програмою «Інформаційне забезпечення робототехнічних систем» за спеціальністю 126 «Інформаційні системи та технології». Київ : КПІ ім. Ігоря Сікорського, 2021. 271 с.

13. Що таке Інтернет речей або IoT? Fiberroad. URL: https://fiberroad.com/uk/resources/resources/what-is-the-internet-of-things-iot/ (дата звернення: 28.02.2025).

14. Пулеко І. В. Єфіменко А. А. Архітектура та технології Інтернету речей: навч. посіб. Житомир : Державний університет «Житомирська політехніка», 2022. 234 с.

15. Методичні вказівки до виконання лабораторних робіт з курсу Інтернет речей. Вінниця : ДонНУ імені Василя Стуса, 2019. 60 с.

16. Тема 3. Стандартні сумісності IoT. Еталонні моделі IoT. URL: https://e-tk.lntu.edu.ua/pluginfile.php/20159/mod_resource/content/0/Тема%203.%20Стандарти%20сумісності%20IoT.%20Еталонні%20моделі%20IoT.pdf (дата звернення: 28.02.2025).

17. Сервіс Blynk. Безпека та відеоспостереження. URL: https://oxorona.com/blynk/ (дата звернення: 27.02.2025).

18. Blynk документація. Blynk UA. URL: https://shoorik007.github.io/ (дата звернення: 27.02.2025).

19. Blynk. Blynk Documentation. URL: https://docs.blynk.io/ (дата зверення: 28.02.2025).

20. How Blynk works for your IoT project. Blynk. URL: https://blynk.io/blynk-iot-low-code-software-platform (дата звернення: 28.02.2025).

21. Міхненко Я. О., Курдеча, В. В. (2019). Порівняння видів хмарних сервісів в IoT. Перспективи телекомунікацій: матеріали XIII міжнар. наук.-техн. конф. (м. Київ, 15-19 квіт. 2019 р.,). Київ, 2019. URL: https://cutt.ly/gpqg4id (дата звернення: 03.03.2025 р.).

22. Brivo Onair. Brivo. URL: https://www.brivo.com/brivo-onair-access-control

(дата звернення: 03.03.2025).

23. Configure Brivo Onair Identity Connector for automatic user provisioning with Microsoft Entra ID. Microsoft Learn. URL: https://learn.microsoft.com/en-us/entra/identity/saas-apps/brivo-onair-identity-connector-provisioning-tutorial (дата звернення: 03.03.2025).

24. Brivo Onair. Saviynt. URL: https://saviynt.com/exchange/apps/brivoonair (дата звернення: 03.03.2025).

25. ButterflyMX. ButterflyMX. URL: https://butterflymx.com/ (дата звернення: 04.03.2025).

26. Benefits for everyone. ButterflyMX. URL: https://butterflymx.com/features/ (дата звернення: 02.03.2025).

27. Integrations. ButterflyMX. URL: https://butterflymx.com/features/integrations/ (дата звернення: 04.03.2025).

28. IoT-Enabled Access Control System for Smart Security. Versionx. URL: https://www.versionx.in/access-control-system (дата звернення: 10.03.2025).

29. IOT-Based Building Management System – 10 Reasons They’re Smart. Versionx. URL: https://www.versionx.in/iot-based-building-management-system (дата звернення: 10.03.2025).

30. VersionX Product Information and Latest Updates. Product Hunt. URL: https://www.producthunt.com/products/versionx?utm_campaign=producthunt-api&utm_medium=api-v2&utm_source=Application%3A%20oghunt%20(ID%3A%20128879)&launch=version (дата звернення: 10.03.2025).

31. Wemos D1 Mini ESP8266 FT232 NodeMcu Lua Wi-F. Мій проект. URL: https://myproject.com.ua/wemos-d1-mini-esp8266-ch340-nodemcu-lua-wifi-ua.html (дата звернення: 15.03.2025).

32. Модуль реле 1-канальний для Arduino 5V. Radio Store. URL: https://radiostore.com.ua/ua/p229350226-modul-rele-kanalnyj.html (дата звернення: 15.03.2025).

33. Світлодіоди. SVL. URL: https://svl.ua/16-svitlodiodi (дата звернення: 16.03.2025).

34. Все, що потрібно знати про потенціометри: типи, символ, програми. Ariat technology. URL: https://ua.ariat-tech.com/blog/everything-you-need-to-know-about-potentiometers-types,characteristics,and-applications.html (дата звернення: 17.03.2025).

35. Arduino - Home. Arduino. URL: https://www.arduino.cc/ (дата звернення: 20.03.2025).

36. Low-code IoT cloud platform with user experience at its core. Blynk. URL: https://blynk.io/ (дата звернення: 20.03.2025).

37. Проект автовідкривачки вікна. Blynk.Community. URL: https://community.blynk.cc/t/topic/37813/1 (дата звернення: 28.03.2025).

38. Web Dashboard. Blynk.Community. URL: https://community.blynk.cc/t/web-dashboard/54082 (дата звернення: 28.03.2025).

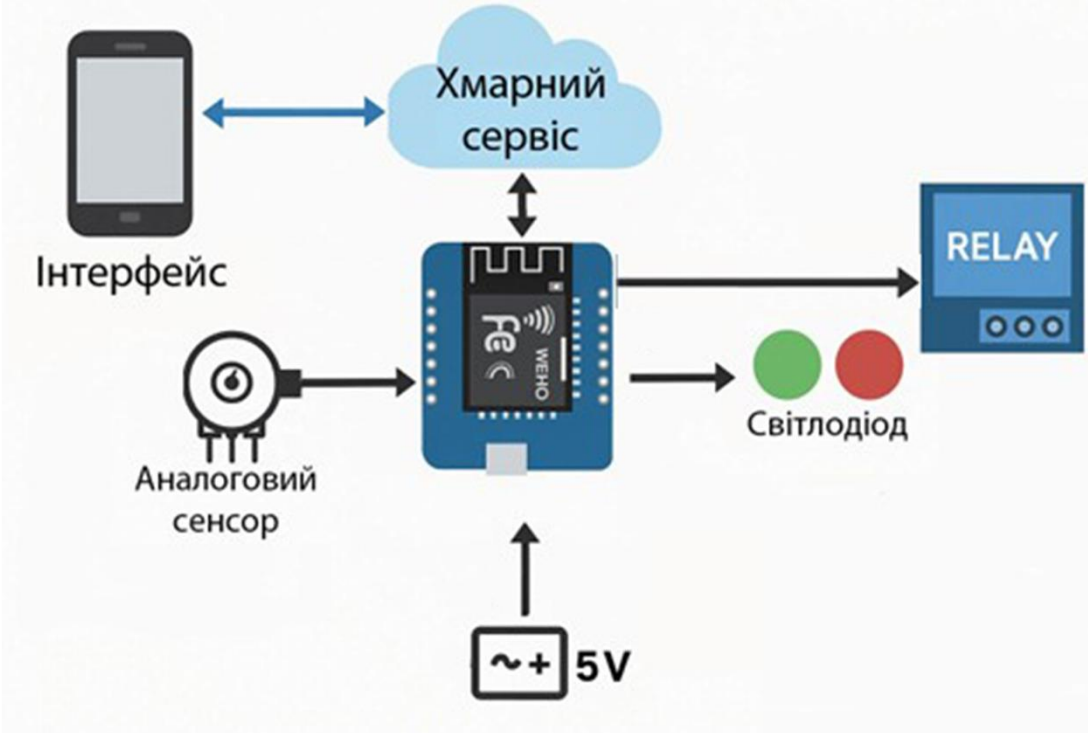
39. Devices. Blynk. URL: https://blynk.cloud/dashboard/ (дата звернення: 25.03.2025).

Мікроконтролер ESP8266. IT MASTER. URL: https://itmaster.biz.ua/directory/microcontrollers/esp8266.html (дата звернення: 30.03.2025).

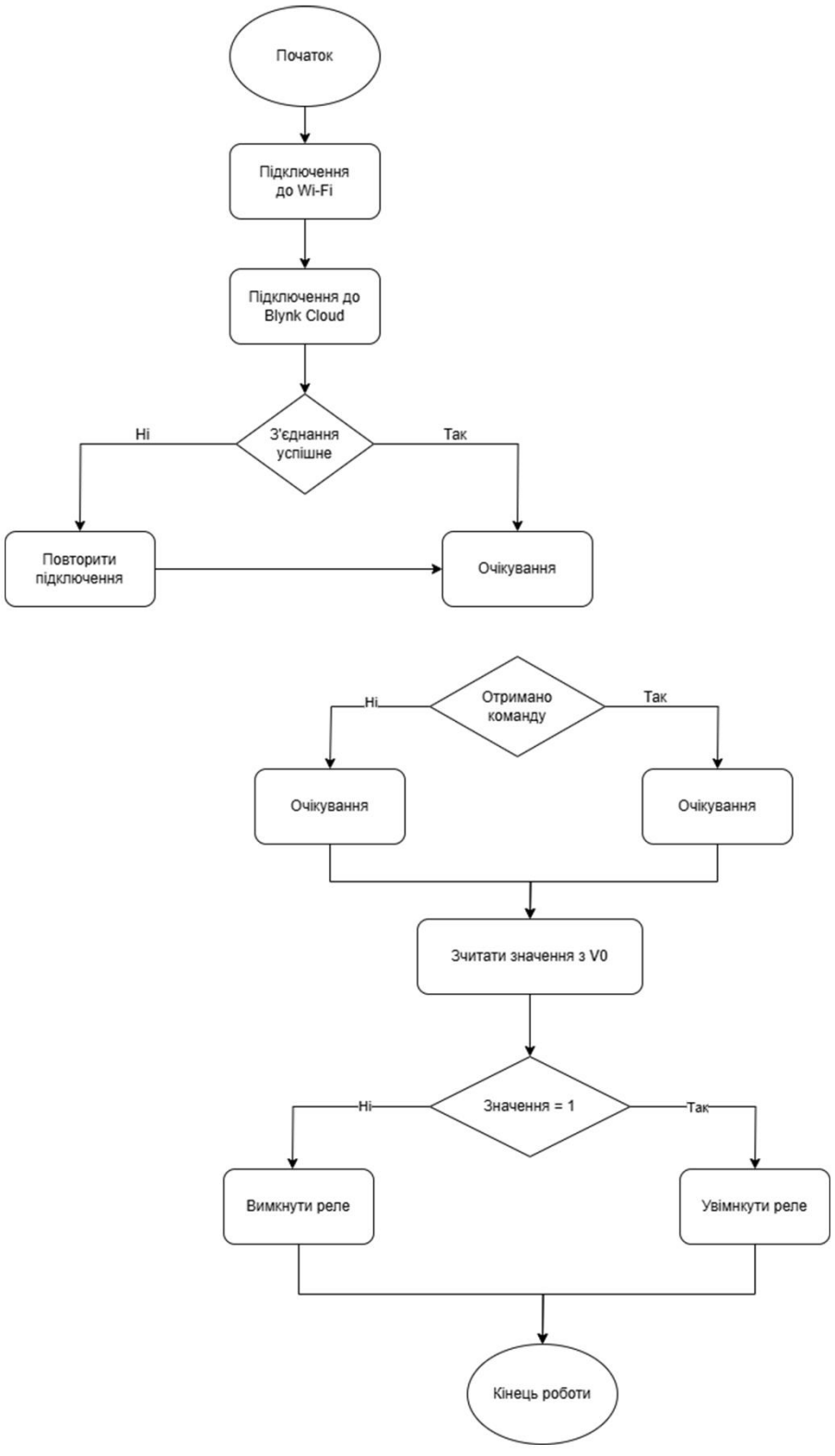
ДОДАТОК А

Копії графічної частини

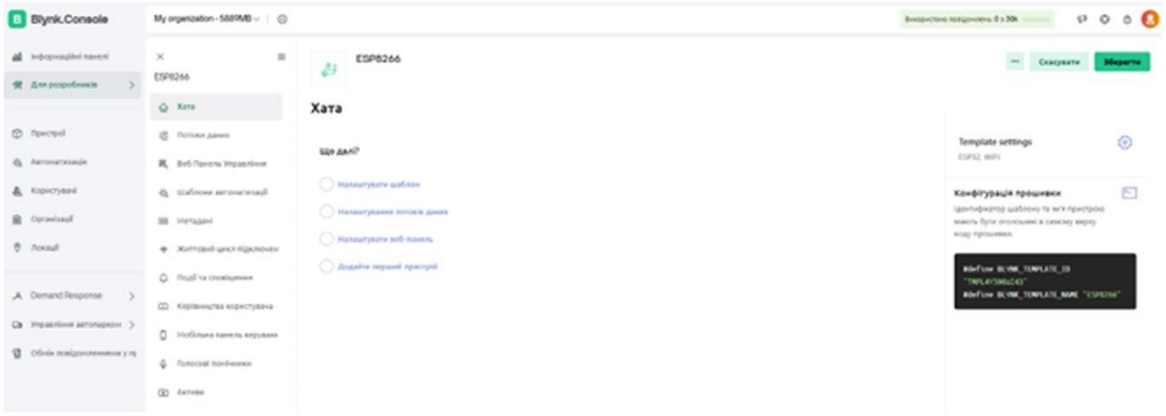
КРБКБ.2102148.21.02.27 Е8

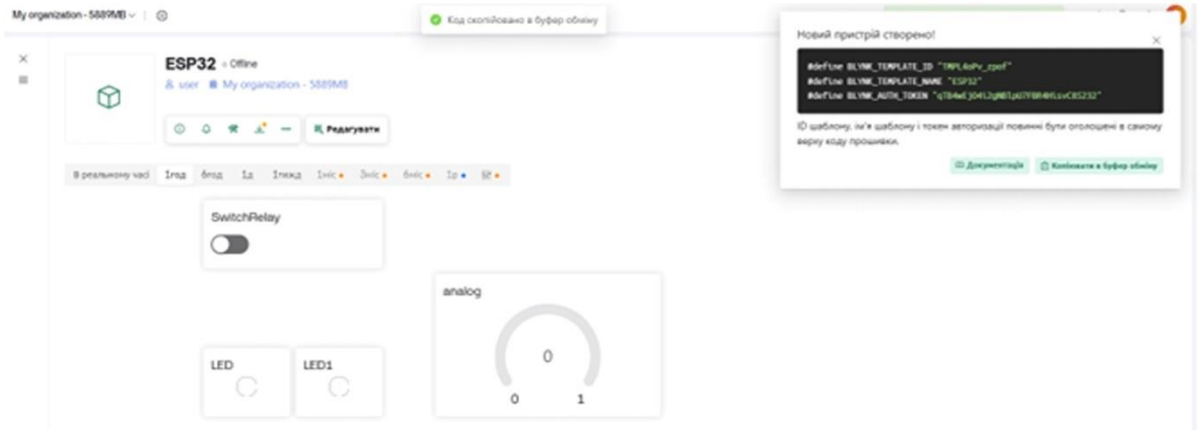


					КРБКБ.2102148.21.02.27 Е8	
Зм.	Арк.	№ докум.	Підпис	Дата	Система допуску на основі сервісу Blynk IoT Схема апаратного забезпечення	Літера / Маса / Масштаб
Розроб.		Когут А.В.				
Перевір.		Петрушак В.С.				Аркуш 1 / Аркушів 3
Н.контр.						
Т. контр.		Мостовий С.В.				ХНУ, КБ-21-2
Затв.		Кльоц Ю.П.				

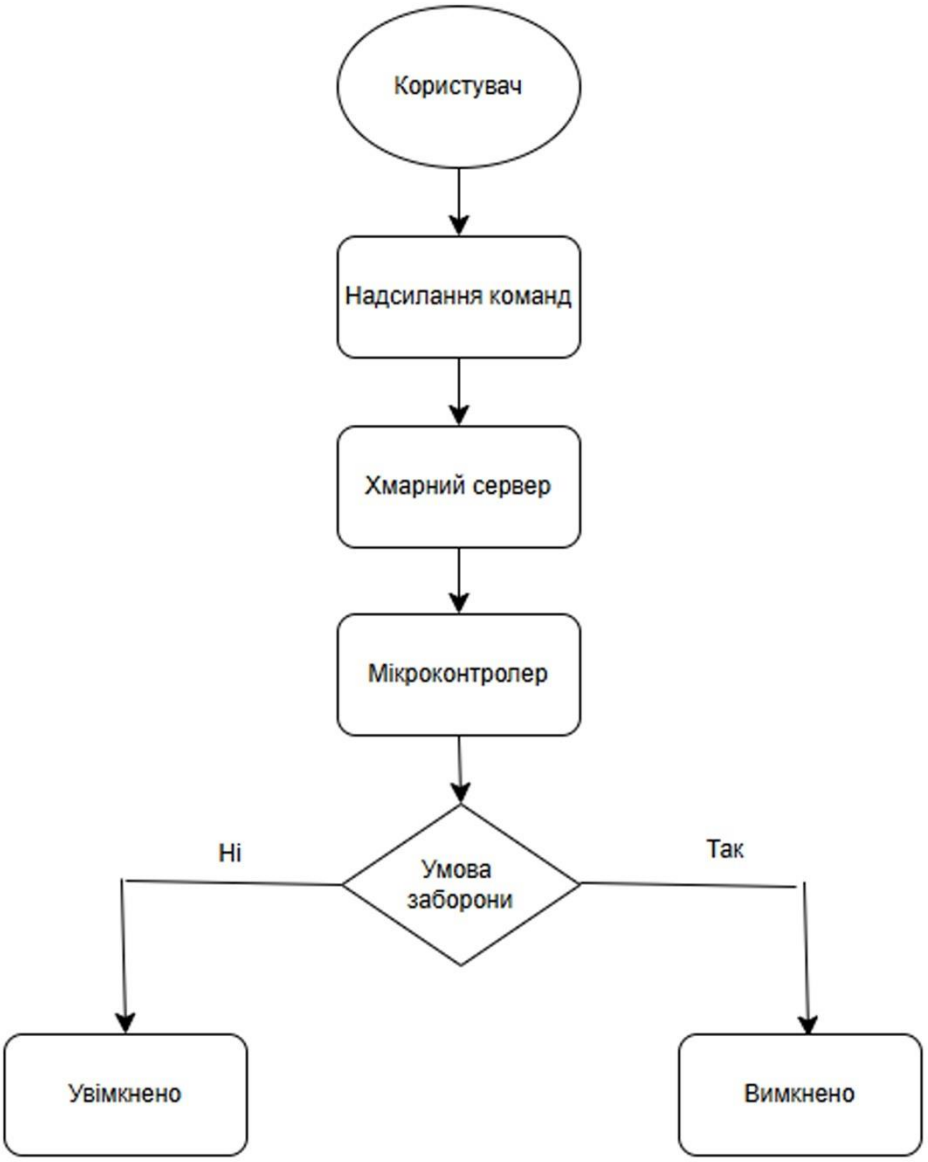


					КРБКБ.2102148.21.02.27 Е8		
Зм.	Арк.	№ докум.	Підпис	Дата	Система допуску на основі сервісу Blynk IoT Схема роботи алгоритму	Літера	Маса / Масштаб
Розроб.		Котут А.В.					
Перевір.		Петрушак В.С.				Аркуш 1	Аркушів 3
Н.контр.							
Т. контр.		Мостовий С.В.				ХНУ, КБ-21-2	
Затв.		Кльод Ю.П.					





					КРБКБ.2102148.21.02.27 Е8	
Зм.	Арк.	№ докум.	Підпис	Дата	Система допуску на основі сервісу Blynk IoT Blynk Dashboard	Літера / Маса / Масштаб
Розроб.		Когут А.В.				
Перевір.		Петрушак В.С.				Аркуш 2 / Аркушів 3
Н.контр.						
Т. контр.		Мостовий С.В.				ХНУ, КБ-21-2
Затв.		Кльоц Ю.П.				



					КРБКБ.2102148.21.02.27 Е8		
Зм.	Арк.	№ докум.	Підпис	Дата	Система допуску на основі сервісу Blynk IoT Сценарій тестування	Літера	Маса / Масштаб
Розроб.		Котут А.В.					
Перевір.		Петрушак В.С.				Аркуш 3	Аркушів 3
Н.контр.						ХНУ, КБ-21-2	
Т. контр.		Мостовий С.В.					
Затв.		Кльод Ю.П.					

ДОДАТОК Б

```
#define BLYNK_TEMPLATE_ID "TMPL4oPv_zpof"
#define BLYNK_DEVICE_NAME "ESP8266"
#define BLYNK_AUTH_TOKEN "elzaQzSrABhYunRaZKpOr_FGZyQIrYE5"
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SimpleTimer.h>
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "khnu.km.ua";
char pass[] = "te3s4q1";
#define RELAY_PIN D1
#define ANALOG_SENSOR A0
#define LED1 D2
#define LED2 D3
#define LED3 D4
WidgetLED vLED1(V1);
WidgetLED vLED2(V2);
WidgetLED vLED3(V3);
SimpleTimer timer;
BLYNK_WRITE(V0) {
  int pinValue = param.asInt();
  digitalWrite(RELAY_PIN, pinValue);
  Blynk.virtualWrite(V0, pinValue);
}
void checkSensor() {
  int sensorValue = analogRead(ANALOG_SENSOR);
  if (sensorValue < 300) {
    digitalWrite(LED1, HIGH);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, LOW);
```

```
    vLED1.on(); vLED2.off(); vLED3.off();
  } else if (sensorValue < 700) {
    digitalWrite(LED1, LOW);
    digitalWrite(LED2, HIGH);
    digitalWrite(LED3, LOW);
    vLED1.off(); vLED2.on(); vLED3.off();
  } else {
    digitalWrite(LED1, LOW);
    digitalWrite(LED2, LOW);
    digitalWrite(LED3, HIGH);
    vLED1.off(); vLED2.off(); vLED3.on();
  }
  Blynk.virtualWrite(V4, sensorValue);
}
void reconnectBlynk() {
  if (!Blynk.connected()) {
    Blynk.connect();
  }
}
void setup() {
  pinMode(RELAY_PIN, OUTPUT);
  digitalWrite(RELAY_PIN, LOW);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  Blynk.begin(auth, ssid, pass);
  timer.setInterval(1000L, checkSensor);
}
void loop() {
  if (Blynk.connected()) {
```

```
    Blynk.run();
  } else {
    reconnectBlynk();
  }
  timer.run();
}
```

Завідувачу кафедри кібербезпеки
канд., тех. наук, доц. Кльоцу Ю.П.
Когута Артура Віталійовича
ПІБ здобувача вищої освіти

Студента ФІТ, 4 курсу, групи КБ-21-2

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 31.08.2023, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений. Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

02.06.2025
дата


підпис

Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Когут Артур Віталійович

Співавтор:

Назва: Система контролювання доступу на основі сервісу Blynk IoT

Науковий керівник:

Підрозділ: Кафедра кібербезпеки

Коефіцієнт подібності 1:1.1%

Коефіцієнт подібності 2:0%

Мікропробіли: 0

Заміна букв: 0

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-06-05 14:34:35.0

Після аналізу Звіту подібності констатую наступне:

☑ **Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.**

☐ **Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.**

☐ **Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.**

Обгрунтування:

06.06.2025р.

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 1.0%

Dictionaries check: en_US, ru_RU, ua_UA. **Errors in the documents: 10%**

ID: 243555 Title: Система контролювання доступу на основі сервісу Blynk IoT Added in a DB: 2025-06-05 Authors: Когут Артур Віталійович Heads: Петрушак В.С. Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	81853	587	604 (1%)	8 (1%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КІБЕРБЕЗПЕКИ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Система контролювання доступу на основі сервісу Blynk IoT

Автор: Когут Артур Віталійович

Спеціальність: 125 – Кібербезпека

Освітня програма: Кібербезпека

Науковий керівник: Володимир ПЕТРУШАК, канд. техн. наук, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних). Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи.	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
5	Інше:	

Підтвердження:

Оригінальність тексту роботи за результатами перевірки системою StrikePlagiarism складає 98,9%, оригінальність тексту роботи за результатами перевірки системою Anti-Plagiarism складає 99%.

Згідно з правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 24.09.2024, авторська робота, обсяг оригінального тексту у відсотках до загального обсягу матеріалу в якій складає 90-100%, визначається роботою з високою унікальністю тексту і допускається до захисту.

Керівник роботи _____ Володимир ПЕТРУШАК

Гарант ОП _____ Віктор ЧЕШУН

Завідувач кафедри кібербезпеки _____ Юрій КЛЬОЦ

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

освітнього ступеня «бакалавр»

Студент Когут Артур Віталійович

Тема Система контролювання доступу на основі Blynk IoT

Спеціальність 125 – Кібербезпека

Обсяг кваліфікаційної роботи освітньо-кваліфікаційного рівня «бакалавр»:

кількість листів креслень 4; кількість сторінок записки 65.

1. Короткий зміст роботи та прийнятих рішень У кваліфікаційній роботі, відповідно до поставленого завдання, проведене дослідження предметної області, розглянуто принципи функціонування сервісу Blynk та його взаємодію з мікроконтролером. Описано практичну реалізацію системи допуску. Описано процес налаштування Blynk-сервісу, як для ПК, так і для мобільного додатку. Проведено тестування системи.

2. Висновок про відповідність кваліфікаційної роботи завданню У кваліфікаційній роботі виконано поставлене завдання як у теоретичній, так і в практичній частині.

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У розділі 1 проаналізовано системи контролю доступу, основи Інтернету речей. Розглянуто можливості сервісу Blynk. Проаналізовано можливі альтернативні рішення, їх переваги та недоліки. У розділі 2 розроблено систему допуску. Описано вибір компонентів та інструментів, розробку та налаштування Blynk Dashboard, програмування пристрою та підключення елементів. У розділі 3 проведено тестування системи та описано результати роботи. Описано методики тестування, які випробування були проведені та проведена оцінка відповідності системи.

4. Позитивні сторони роботи Робота базується на детальному аналізі вимог нормативних документів та законів України, що регулюють питання проєктування, впровадження і супроводу комплексних систем захисту інформації. Кваліфікаційна робота має практичну цінність і орієнтована на вдосконалення захисту інформації в автоматизованій системі кафедри кібербезпеки.

5. Негативні сторони роботи В роботі недостатньо уваги приділено опису налаштувань Dashboard для мобільного додатку системи. Відсутні розрахунки для системи.

6. Оцінка графічного оформлення та пояснювальної записки роботи Графічне оформлення кваліфікаційної роботи відповідає темі роботи та виконане з дотриманням стандартів. В цілому, графічне оформлення є якісним, а пояснювальна записка відповідає нормам оформлення.

7. Відгук про роботу в цілому Кваліфікаційна робота заслуговує позитивної оцінки, оскільки весь матеріал роботи є структурованим, чітким та послідовним. Усі розділи роботи мають логічну послідовність, що сприяє зрозумінню викладеного матеріалу в рамках теми роботи. Графічний матеріал допомагає наочно продемонструвати доцільність та ефективність прийнятих рішень у проєктуванні системи допуску на основі сервісу Blynk IoT.

8.Інші зауваження _____

9. Оцінка кваліфікаційної роботи Враховуючи всі позитивні сторони кваліфікаційної роботи, а також негативні сторони, рекомендованою оцінкою є «задовільно».

РЕЦЕНЗЕНТ (прізвище, ім'я, по батькові, посада, місце роботи)

професор кафедри телекомунікацій, медійних та інтелектуальних технологій, д.т.н, Бойко Юлій Миколайович

«03» 06 2025. _____ (підпис)