

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра інженерії програмного забезпечення автоматизованих систем

Пояснювальна записка

до магістерської роботи
на ступінь вищої освіти магістр

на тему: **«Розробка системи розумного міста з використанням
технології LoRaWAN»**

Виконав: студент 6 курсу, групи ІСДМ-61
спеціальності 126 Інформаційні системи та технології
освітня програма «Інформаційні системи та технології»
(шифр і назва спеціальності)

Асман М.Я.

(прізвище та ініціали)

Керівник Ткаленко О.М.

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

Нормоконтроль Данильченко В.М.

(прізвище та ініціали)

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення автоматизованих систем
 Ступінь вищої освіти - «Магістр»
 Спеціальність підготовки 126 Інформаційні системи та технології
 Освітня програма «Інформаційні системи та технології»

ЗАТВЕРДЖУЮ

Завідувач кафедри ІПЗАС

К.П.Сторчак

“ ____ ” ____ 2021 року

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

Асману Максиму Ярославовичу

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка системи розумного міста з використанням технології LoRaWAN»

Керівник роботи: Ткаленко Оксана Миколаївна, к.т.н., доцент

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

затверджені наказом вищого навчального закладу від 11 жовтня 2021 року №170

2. Строк подання студентом роботи 24.12.2021

3. Вхідні дані до роботи:

1. Науково-технічна література

2. Існуючі концепції розумного міста

3. Технологія LoRaWAN

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

1. Аналіз концепції «Розумного та безпечного міста»

2. Інтернет речей в концепції «Розумного міста». Технології на базі IoT

3. Розробка системи «Розумного міста»

5. Перелік графічного матеріалу:

1. Аналіз технологій на базі IoT

2. Особливості технології LoRaWAN

3. Необхідність систем автоматизації

4. Сфери застосування концепції «Розумного міста»

5. Вибір програмного забезпечення

6. Налаштування та підключення openHAB до MQTT брокера

7. Підключення шлюзу LoRaWAN до MQTT

6. Дата видачі завдання 12.10.2021

КАЛЕНДАРНИЙ ПЛАН

№ /п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	14.10.2021	
2	Вивчення матеріалів для опрацювання	22.10.2021	
3	Огляд концепцій розумного міста	01.11.2021	
4	Визначення технічного завдання	12.11.2021	
5	Розробка сценарію взаємодії	22.11.2021	
6	Розробка системи	29.11.2021	
7	Вступ, висновки, реферат	07.12.2021	
8	Розробка демонстраційних матеріалів	15.12.2021	
9	Попередній захист роботи	20.12.2021	

Студент _____ Асман М.Я.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Ткаленко О.М.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина бакалаврської роботи 60 с., 51 рис., 20 джерела.

РОЗУМНЕ МІСТО, АВТОМАТИЗАЦІЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ІНТЕРНЕТ РЕЧЕЙ, АРХІТЕКТУРА, ТЕХНОЛОГІЯ LoRaWAN, ОПЕРАЦІЙНА СИСТЕМА, ПРОТОКОЛ

Об'єкт дослідження: концепції побудови систем розумного міста.

Мета роботи: розробити систему розумного міста на базі технології LoRa з використанням програмного забезпечення OpenNAV.

У даній магістерській роботі розроблена концепція системи розумного міста з використанням безкоштовного програмного забезпечення. Проаналізовано принципи роботи протоколів передавання даних. Проведено огляд сучасних систем розумного міста та технологій їх побудови.

На базі отриманих даних розроблена власна система автоматизації з використанням OpenNAV.

ЗМІСТ

Вступ.....	8
1. АНАЛІЗ КОНЦЕПЦІЇ "РОЗУМНОГО МІСТА"	9
1.1 КОНЦЕПЦІЯ «Розумного та безпечного міста»	9
1.2 Функції «розумного міста».....	10
1.3 Задачі «розумного міста»	11
1.4 Сфери застосування концепції «розумного міста».....	12
1.5 ІНТЕРНЕТ РЕЧЕЙ В КОНЦЕПЦІЇ «РОЗУМНОГО МІСТА»	17
2. ТЕХНОЛОГІЇ НА БАЗІ ІОТ.....	19
2.1 Дослідження технології NB-IoT.....	19
2.2 Дослідження технології LTE-M.....	20
2.3 ТЕХНОЛОГІЇ LORA	22
3 РОЗРОБКА СИСТЕМИ "РОЗУМНОГО МІСТА"	31
3.1 Вибір програмного забезпечення	31
3.2 Встановлення openHAB на операційну систему Windows 10	34
3.3 Встановлення MQTT broker на WINDOWS 10	39
3.4 Налаштування та підключення openHAB до MQTT брокера.....	45
3.5 Підключення шлюзу LoRaWAN до MQTT.....	50
Висновки.....	62
ПЕРЕЛІК ПОСИЛАНЬ.....	63
ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація).....	64

Вступ

Останні роки характеризуються активною урбанізацією і розбудовою міст та значним зростанням чисельності їх населення. Все більше створюється нових технологій і пристроїв, та все гостріше постає питання їх взаємодії та ефективності. Тематика «Розумного міста» не є новою. Ця тема обговорюється вже більше десяти років і в світі з'являється все більше міст, де активно використовуються новітні технології та впроваджуються концепції «Розумного міста». Однак неможливо чітко зазначити яке місто є більш «розумним» через те, що немає єдиного стандарту, постійно змінюється технологічний ландшафт міст, все більше збирається різномірної інформації, яка може стати базою для реалізації концепції «Розумного міста».

Ми частіше говоримо про покращення якості життя людей та їх безпеки в наших містах, що динамічно розвиваються, і «Розумне місто» виглядає більш підготовленим і захищеним перед різними загрозами нашого часу та за допомогою цифрових та комунікаційних технологій надає своїм мешканцям цілий перелік сервісів та більш ефективно задовольняє їхні потреби. Мова йде про транспорт, медицину, освіту, житлово-комунальні сервіси та благоустрій, екологію, безпеку та ін. Впровадження технологій «Розумного міста» має на меті також покращити рівень управління комунальним майном, вирішенням щоденних питань життєдіяльності міста та ефективним використанням бюджетів.

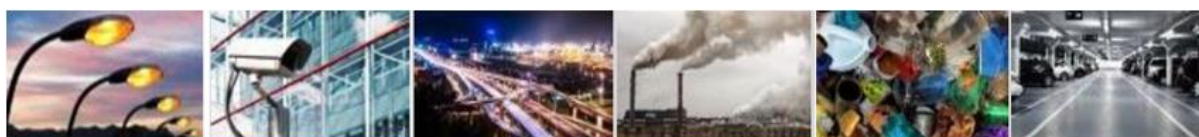
1. АНАЛІЗ КОНЦЕПЦІЇ "РОЗУМНОГО МІСТА"

1.1 КОНЦЕПЦІЯ «Розумного та безпечного міста»

Концепція «Розумного міста» базується на використанні цифрових та комунікаційних технологій для більшої інтелектуалізації міського середовища та має на меті покращення якості життя мешканців міст.

Через значні темпи зростання міст і кількості мешканців традиційний підхід до використання інфраструктури та ресурсів є неефективним з технічної та економічної точки зору:

- Дублювання інфраструктури та ресурсів – перенавантаження каналів зв'язку та обчислювальних ресурсів;
- Обмін інформації між департаментами і системами ускладнений або відсутній взагалі;
- Неефективне використання бюджетних коштів;
- Проблеми масштабування рішень



Освітлення Безпека Транспорт Екологія ЖКГ Паркування

Рисунок 1.1 Фрагментований підхід по вирішення проблем неефективний технічно і економічно

Сучасні електронні засоби інформатизації та комунікації відкривають нові безпрецедентні можливості для вирішення проблем міста – від систем управління майном до електронної мерії та уряду.

Концепція «Розумного міста» поєднує в єдину систему управління майном і активами міста на базі енергоефективних та енергозберігаючих технологій та інформатизацію виробничих процесів. Таке впровадження забезпечить підвищення

якості життя за рахунок інноваційних технологій, забезпечить координацію економічного зростання з екологічною складовою, а також сучасні і своєчасні методи прийняття рішень.



Рисунок 1.2 Дизайн концепції «Розумного міста»

1.2 Функції «розумного міста»

Концепція «Розумне місто» виконує такі важливі функції:

- збирання, передача та зберігання даних від пристроїв або громадян;
- налагодження взаємозв'язку між громадянами та керівництвом міста
- автоматизація всіх процесів життєдіяльності міста та зниження витрат
- покращення рівня життя мешканців та збереження навколишнього середовища.

Термін «Розумне місто» є достатньо новим і не має стандартного визначення цієї концепції. Але експерти сходяться на думці, що головний аспектом впливу є дані про чисельність населення. Чим точніші дані та більший їх обсяг, тим більше з'являється можливостей для оптимізації та впровадження технологій. Різноманітні датчики збирають інформацію з електронних пристроїв і від жителів міста. Ця інформація збирається та аналізується, і на базі цих даних приймаються

рішення в тому числі по усуненню неефективності наявних процесів життєдіяльності міста.



Рисунок 1.3 Розумне місто

1.3 Задачі «розумного міста»

Безпека населення:

- реактивна – розслідування інцидентів
- проактивна – попередження правопорушень

Безпека транспорту

- Муніципальний транспорт – метро, автобуси, трамваї і т.д
- Залізниця, аеропорти
- Затори, інциденти, оптимізація трафіка

Збереження та облік міського майна

- Захист від вандалізму
- Захист власності
- Облік майна, оперативність та актуальність інформації



Рисунок 1.4 Система відеонагляду

1.4 Сфери застосування концепції «розумного міста»

Безпека – одна з найважливіших базових потреб людини. У великих містах спроможність влади забезпечити безпекову складову є основним чинником розвитку міста, як економічного, так і соціально-культурного.

Приклади проектів:

- Предиктивна поліція, носимі камери і відеофіксація
- Карта злочинності в реальному часі
- Розумний відеонагляд
- Ідентифікація пострілів
- Оптимізація реагування на надзвичайні ситуації
- Системи раннього попередження стихійних лих
- Системи домашньої безпеки і охорони
- Інспекція будівель на базі даних
- Аплікації для персональних сповіщень та інше.

Охорона здоров'я.

Технології «Розумного міста» роблять охорону здоров'я більш ефективною та зручною, що відповідає стандартам життя сучасної людини. Змінюється підхід з традиційних засобів до впровадження IoT технологій: так розумні трекери, що використовуються для відстеження показників пацієнтів поза межами медичних закладів в поєднанні з моніторинговими системами здатні попередити захворювання та кризові стани здоров'я пацієнтів.

Приклади проектів:

- Телемедицина - використання комп'ютерних та телекомунікаційних технологій для обміну медичною інформацією. Є одним із найбільш швидко зростаючих сегментів охорони здоров'я у світі
- Віддалений моніторинг пацієнта - технологія, що дозволяє проводити моніторинг пацієнтів поза звичайними клінічними умовами, наприклад, вдома або у віддаленому районі, що може збільшити доступ до медичної допомоги та зменшити витрати на доставку медичної допомоги.
- Нагляд за інфекційними захворюваннями
- Пошук та бронювання місця для планового медичного обслуговування
- Інтегровані системи управління потоками пацієнтів
- Засоби по охороні здоров'я в часи пандемії на основі даних та інше.

Транспорт і мобільність пересування

- Інформація про громадський транспорт в реальному часі дозволяє користувачам планувати свою поїздку ефективніше, допомагає зменшити незручності та підтримує плавніший транспорт.

Карти Google, якими щодня користуються мільйони пасажирів, є незамінним джерелом даних про мобільність. Він не тільки надає інформацію про дорожній рух у режимі реального часу, а й дає змогу користувачам знайти правильне сполучення громадського транспорту за розкладом, наданим містами.

- Цифрова сплата за проїзд. Ефективність – запорука успіху. Завдяки оптимально ефективному збору тарифів задоволеність пасажирів зростає, а адміністративні витрати зменшуються
 - Інтелектуальні дорожні сигнали - система управління рухом транспортних засобів, яка поєднує в собі традиційні світлофори з масивом датчиків та штучним інтелектом для інтелектуальної маршрутизації руху транспортних засобів та пішоходів
 - Сервісне обслуговування та профілактика технічного стану транспортних засобів
 - Динамічна зміна маршрутів громадського транспорту для оптимізації транспортних потоків
 - Розумні паркування для автомобільного транспорту
 - Навігація містом в режимі реального часу
 - Каршерінг та шерінг велосипедів та самокатів та інше
- Енергозбереження

В умовах сучасних викликів, обмеженості енергоресурсів та їх високої вартості енергозбереженню немає альтернатив. Пріоритетним напрямком є модифікація систем енергозбереження та пошук альтернативних джерел.

Приклади проектів:

- Системи автоматизації в будівлях
- Моніторинг використання електроенергії в будівлях
- Системи автоматизації побутової техніки
- Розумне освітлення вулиць – розумні ліхтарі
- Системи автоматизації розподілення енергії
- Зарядні станції для електромобілів
- Дорожнє покриття, що акумулює кінетичні енергію внаслідок тертя коліс по дорожньому покриттю
- Використання енергії сонці, вітру і води
- Динамічне ціноутворення на електроенергію та інше.

Водні ресурси

Найбільш використаним і доступним ресурсом є вода, але і цей ресурс не є невичерпним. Із змінами клімату та забрудненням навколишнього середовища окремо постає питання якості питної води і її доступності.

Приклади впровадження проектів:

- Відслідковування та прогнозування витрат води
- Виявлення та усунення протікань в системах постачання та зберігання водних ресурсів
- Розумні системи поливу рослин
- Керування якістю води

Екологія

У великих містах серйозною проблемою є екологічний стан – перенаселення і засміченість, перевантаження доріг і проблеми викидів від автомобілів і промисловості, утилізація відходів і безпечно їх зберігання. Впровадження концепції «Розумного міста» дає приклади таких проектів:

- Контроль і управління екологічною ситуацією – стан повітря, води та інше
- Цифровий моніторинг оплати вивозу сміття
- Контроль накопичення сміття, оптимізація маршрутів вивезення сміття та інше

Залученість громад, економічний розвиток та рівень життя.

- Публічна міська мережа WiFi
- Місцеві електронні центри кар'єри
- Програми онлайн-перепідготовки професійних кадрів
- Цифровізація послуг по землеустрою, будівництву, ліцензуванню бізнесу та дозвільна система
- Цифрова подача декларацій
- Персоніфікована інклюзивна освіта та інше.



Рисунок 1.5 Сфери застосування

Бездротові мережі IoT діляться на типи:

- Мережі малого радіусу дії - Енергоефективні мережі малого радіусу дії мережі малого радіусу дії;
- Глобальні мережі малої потужності (LPWAN) – це енергоефективні мережі великого радіусу дії; - Глобальні мережі малої потужності (LPWAN) – це енергоефективні мережі великого радіусу дії.

Low Power Wide Area Networks (LPWAN) – енергоефективні мережі великої дальності;

- Стільникова мережа - технології, засновані на використанні стандартів Мережа стільникового зв'язку - технології, засновані на використанні стандартів стільникових мереж у ліцензійному діапазоні.

З дротових технологій важливу роль відіграє проникнення Інтернету на все грають рішення PLC - технології, що використовуються мережами через лінії лінії електропередач, тому що в багатьох різних приладах є доступ до електромережі. До них відносяться різні торгові автомати, банкомати, метрів, датчики освітлення, підключені до електромережі.

1.5 ІНТЕРНЕТ РЕЧЕЙ В КОНЦЕПЦІЇ «РОЗУМНОГО МІСТА»

Мережа фізичних об'єктів (речей), підключених до Інтернету та взаємодіючих із зовнішнім середовищем або один з одним, відноситься до терміну «Інтернет речей». Термін походить від Інтернету речей, або скорочено IoT.

Сьогодні технології IoT швидко проникають всі сфери життя суспільства. Активно використовуючи різні пристрої, ми робимо наше життя кращим. Удосконалення пристроїв робить їх здатними чути, бачити, думати, а іноді й діяти.

Для злагодженої та ефективної роботи прилади повинні спілкуватися один з одним якомога коректніше, щоб вони могли швидко приймати правильні рішення, які можуть вплинути на життя та безпеку людей або інших об'єктів – будівель, транспортних засобів та ін. Найсучасніші бездротові технології, різноманітні датчики, новітні технології дозволяють Інтернету речей проникнути в наше життя і завжди бути під рукою. Сьогодні з існуванням величезної кількості виробників, технологій і пристроїв, виникає багато проблем їхньої інтеграції та працездатності. Тому виникла потреба у створенні та прийнятті спеціальних стандартів і протоколів зв'язку.

Сама концепція Інтернету речей передбачає, що до мережі будуть підключені мільйони пристроїв. Серед основних проблем – оптимальний контроль і доступність: швидке відстеження збоїв, конфігурацій і продуктивності такої величезної кількості пристроїв за допомогою протоколів управління. Крім того, має бути забезпечена взаємодія в мережі: різноманітні пристрої та протоколи повинні працювати один з одним, зберігаючи конфіденційність та безпеку.

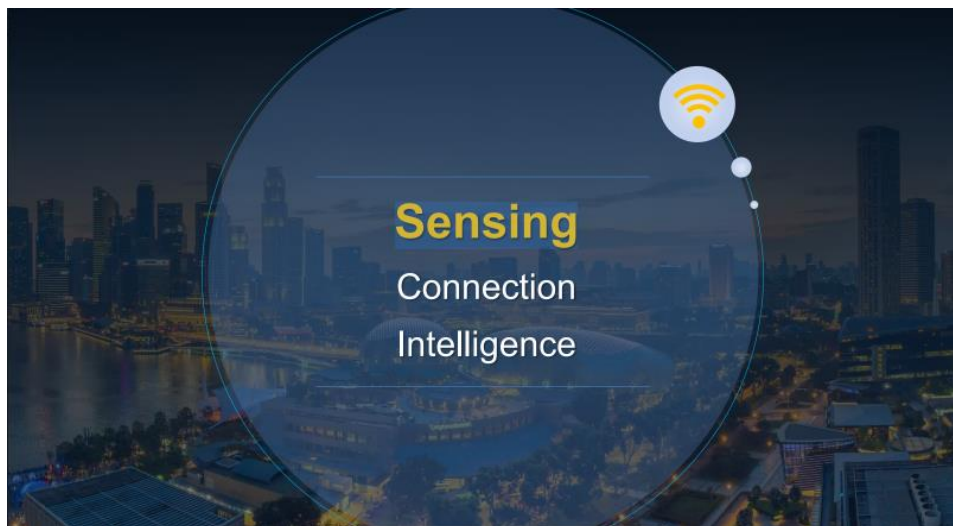


Рисунок 1.6 Концепції Розумного міста

2. ТЕХНОЛОГІЇ НА БАЗІ ІОТ

2.1 Дослідження технології NB-IoT

NarrowBand-Internet of Things (NB-IoT) — це заснована на стандартах технологія широкої зони з низьким енергоспоживанням (LPWA), розроблена для створення широкого спектру нових пристроїв і послуг IoT. NB-IoT значно покращує енергоспоживання користувальницьких пристроїв, ємність системи та ефективність використання спектру, особливо в умовах глибокого покриття. Термін служби батареї більше 10 років може підтримуватися в широкому діапазоні випадків використання.

Нові сигнали та канали фізичного рівня розроблені, щоб задовольнити вимоги розширеного покриття – у сільській місцевості та глибоко всередині приміщень – та надзвичайно низької складності пристроїв. Очікується, що початкова вартість модулів NB-IoT буде порівнянна з GSM/GPRS. Проте базова технологія набагато простіша, ніж сучасні GSM/GPRS, і очікується, що її вартість швидко знизиться в міру зростання попиту.

Підтримується всіма основними виробниками мобільного обладнання, наборів мікросхем і модулів, NB-IoT може співіснувати з мобільними мережами 2G, 3G і 4G. Він також отримує переваги від усіх функцій безпеки та конфіденційності мобільних мереж, таких як підтримка конфіденційності ідентифікаційних даних користувачів, автентифікації об'єктів, конфіденційності, цілісності даних та ідентифікації мобільного обладнання.

2.2 Дослідження технології LTE-M

LTE-M — це гілка технології LTE, яка була створена для зв'язку «машина-машина». Використовуючи LTE-M, пристрої IoT можуть передавати та отримувати великі обсяги даних без розрядження батареї, як у мережах 2G, 3G та 4G.

Основні причини, чому організації IoT покладаються на LTE-M:

Споживання енергії

Покриття

Швидкість передачі даних

Мобільність

Майбутня розстойка

Вартість пристрою

Споживання енергії

Мережі стільникового зв'язку були розроблені для забезпечення спонтанних з'єднань між мобільними телефонами. Вони постійно пингують мобільні пристрої, щоб отримати інформацію про місцезнаходження, що допомагає підтримувати телефон підключеним до найближчих веж і базових станцій. Ці оновлення області відстеження (TAU) розряджають акумулятор.

Але більшість пристроїв IoT не використовують стільникові мережі так само, як мобільні телефони. Вони надсилають або отримують невеликі пакети даних періодично або на основі певних тригерів (наприклад, датчика на системі сигналізації), і батареї повинні працювати набагато довше.

LTE-M дозволяє пристроям IoT використовувати режим енергозбереження, який дозволяє їм «перейти в режим сну», коли вони не використовуються. Їм також не потрібно прокидатися, щоб надіслати лише оновлення розташування, але вони можуть продовжити час сну на довший період часу. LTE-M також дозволяє пристроям використовувати розширений безперервний прийом (eDRX). Коли пристрій виходить із режиму PSM, він регулярно перевіряє інформацію за низхідним каналом. Завдяки eDRX час перевірки радіо збільшується, що зменшує споживання електроенергії.

Ідея цих функцій енергозбереження полягає в тому, що LTE-M повинен підтримувати 10-річний термін служби батареї від батареї ємністю 5 Вт/год (ват-годин). Ця технологія є частиною того, що робить це можливим.

Покриття

На відміну від інших мереж LPWAN, мережі LTE-M/Cat-M1/Cat-M2 можуть використовувати існуючу інфраструктуру 4G LTE. Це є значною перевагою, оскільки понад 50 відсотків глобальних мобільних з'єднань здійснюється в мережах 4G, а інфраструктура 4G, за прогнозами, охопить 95 відсотків світу до 2026 року.

Однак не всі оператори з покриттям 4G мають мережі LTE-M. Станом на грудень 2020 року 70 операторів у 40 країнах інвестували в технології LTE-M, 51 мережа вже була розгорнута.

З максимальними втратами зв'язку (MCL) 156 децибел (дБ) — на 14 дБ вище, ніж у LTE — мережі LTE-M також забезпечують більше покриття та краще проникнення в приміщення. Сигнал може впоратися з великою кількістю перешкод від будівель та інших споруд, які перешкоджають його шляху.

Швидкість передачі даних

У порівнянні з LTE, LTE-M не дуже швидкий. Але 1 мегабіт в секунду для передачі по висхідній і низхідній лінії є винятковим у порівнянні зі старими мережами, такими як 2G і 3G (UMTS) та іншими LPWAN. (Це в кілька разів швидше, ніж NB-IoT.) І цього більш ніж достатньо для більшості додатків IoT. Це можливо для програм з невеликими потребами в даних і навіть для випадків використання, які передбачають потокове відео.

У контексті IoT висока пропускна спроможність даних означає, що пристрої, що використовують LTE-M, можуть легко отримувати оновлення по повітрю (OTA), а передача даних споживатиме менше енергії, оскільки пристрій може повернутися в режим енергозбереження швидше.

2.3 ТЕХНОЛОГІЇ LORA

LoRaWAN – це енергоефективна мережа дальнього радіусу дії (Long Range Wide Area Network), яка відрізняється великим радіусом дії і відкритим стандартом, що надає широкий спектр використання для різноманітних задач.

На відміну від LTE технологій необхідна невелика кількість базових станцій(шлюзів), які «слухають» ефір в заданому діапазоні частот і відповідає пристрою на діапазоні його ж звернення.

Ширина каналу при цьому складає 125 кГц максимальна швидкість – трохи більше 5 кілобіт/с. Цей стандарт не створений для потокового відео, саме тому невелике за обсягом повідомлення з даними передається максимально швидко і гарантовано від датчика до базової станції. Станція приймає і передає дані, а обробка даних відбувається на сервері аплікації. На сервері також відбувається розшифрування даних з датчиків їх візуалізація в дашбордах і аплікаціях звітності або передача в систему білінгу.

Технологія LoRa (Long Range) народилася під егідою LoRa Alliance, неприбуткової організації, заснованої IBM, Semtech, Cisco, Kerlink, IMST, MultiTech, Microchip Technology та іншими, щоб прийняти та просувати LoRaWAN як загальний стандарт для Глобальна мережа малої потужності (LPWAN).

Технологія LoRa поєднує в собі метод модуляції LoRa для бездротової мережі LPWAN, розроблений і запатентований корпорацією Semtech Corporation, і відкритий протокол LoRaWAN, розроблений і запроваджений у 2015 році. Semtech Corporation і IBM Research.

Основними перевагами технології LoRa, які позиціонує «LoRa Alliance», є:

- Можливість автономної роботи кінцевих пристроїв до 10 років від однієї батареї AA завдяки наднизькому споживанню енергії модемами LoRa (від 9,7 мА в режимі прийому даних, від 40 мА в режимі передачі та від 200 нА в сплячому режимі);

- висока завадостійкість завдяки можливості демодуляції сигналу на ~ 20 дБ нижче рівня шуму.

Мережа LoRaWAN складається з таких елементів: кінцева точка, шлюзи, мережевий сервер і сервер додатків.

Кінцева точка - призначена для виконання контрольних або вимірювальних функцій. Він містить набір необхідних датчиків і елементів управління.

Шлюз - пристрій, який отримує дані від кінцевих пристроїв за допомогою радіоканалу і передає їх у транзитну мережу. В якості транзитної мережі можна використовувати Ethernet, WiFi або мобільні радіомережі. Шлюз і кінцеві пристрої утворюють топологію мережі зірка. Як правило, цей пристрій містить багатоканальні приймачі для обробки сигналів на кількох каналах одночасно або навіть кількох сигналів на одному каналі. Відповідно, декілька таких пристроїв забезпечують зону радіопокриття мережі та прозору двонаправлену передачу даних між кінцевими пристроями та сервером.

Мережевий сервер - призначений для управління мережею: налаштування розкладу, адаптації швидкості, зберігання та обробки отриманих даних.

Сервер додатків - може дистанційно керувати роботою кінцевих пристроїв і збирати з них необхідні дані

Мережі LoRaWAN зазвичай розташовуються як топологія «зірка-зірка», в якій шлюзи передають повідомлення між кінцевими пристроями та центральним сервером мережі. Усі шлюзи підключені до сервера основної мережі через стандартні IP-з'єднання, тоді як кінцеві пристрої використовують зв'язок LoRa з одним або кількома шлюзами. всі зв'язок ізначально двонаправлений, хоча зв'язок висхідної лінії зв'язку від кінцевого пристрою до мережі

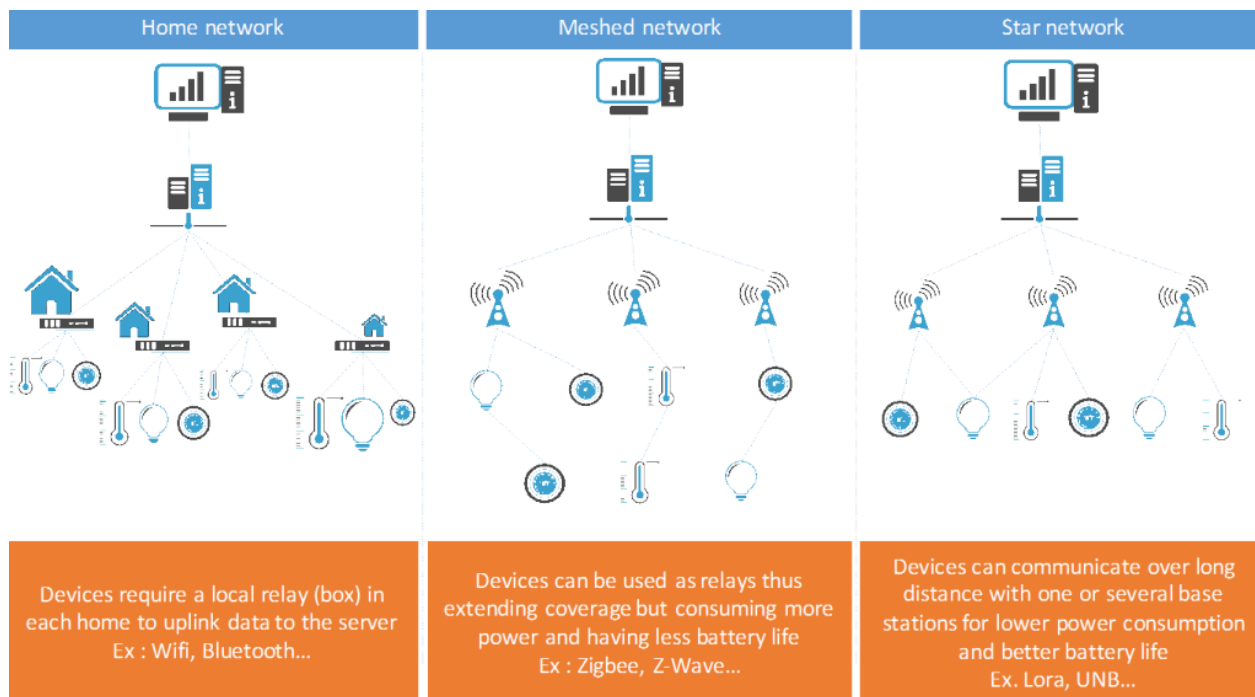


Рисунок 2.1 Звичайні топології мережі

Очікується, що сервер буде переважаючим випадком використання та шаблоном трафіку. Архітектура мережі «зірка» забезпечує найкращий компроміс між дальнім зв'язком, кількістю антени (базові станції) та час автономної роботи пристроїв. Зв'язок між кінцевими пристроями та шлюзами розподіляється на різних частотних каналах і швидкостях передачі даних.

Вибір оптимальної швидкості передачі даних є компромісом між діапазоном зв'язку та повідомленням тривалість. Зв'язок, що використовує різні швидкості передачі даних, не заважає один одному. LoRa підтримує швидкість передачі даних в діапазоні від 300 біт/с до 5 кбіт/с для смуги пропускання 125 кГц. Щоб максимально збільшити час автономної роботи кожного пристрою і загальну ємність, доступну через систему, мережеві інфраструктури LoRa використовують адаптивну швидкість передачі даних

(ADR) схема для керування індивідуальними швидкостями передачі даних і РЧ-виходом кожного підключеного пристрою.

Кінцеві пристрої можуть здійснювати передачу по будь-якому доступному каналу в будь-який час, використовуючи будь-яку доступну швидкість передачі даних, доки дотримуються наступні правила:

- Кінцевий пристрій змінює канал псевдовипадковим чином для кожної передачі. Отримана частота різноманітність робить систему більш стійкою до перешкод.
- У діапазоні EU 868 ISM кінцевий пристрій повинен відповідати максимальному робочому циклу передачі щодо піддіапазону використовуваних та місцеві норми (1% для кінцевих пристроїв).

У США кінцевий пристрій дотримується максимальної тривалості передачі (або часу затримки) відносно використовуваного піддіапазону та місцеві правила, що становить 400 мс.

Адаптивна швидкість передачі даних — це процедура, за допомогою якої мережа інструктує вузол виконати адаптацію швидкості використовуючи запитану швидкість передачі даних (і запитану потужність TX в майбутніх версіях LoRaWAN). Таблиця нижче ілюструє швидкість передачі даних як функція відстані та коефіцієнта поширення (SF). Як показано, LoRaWAN оптимізує швидкість передачі даних для мінімізації ефірного часу та споживання енергії пристроями. У порівнянні з фіксованими даними за технологіями LPWA, ця оптимізація може знизити середнє споживання енергії підключеним об'єктом на a коефіцієнт 100.

Мережі LoRaWAN працюють у діапазоні частот, які не потребують ліцензування. Вони мають високу стійкість до перешкод. Тривалість життя термін служби батареї близько 10 років. Одна базова станція обслуговує десятки тисяч пристроїв.

Мережа має багато переваг.

Термін служби батареї. Обмін даними відбувається лише асинхронно тільки в разі потреби. У звичайних мережах пристрої «прокидаються», синхронізуються з мережею, перевіряти повідомлення. Весь процес веде до

Весь процес споживає електроенергію і зношує акумулятор. Час роботи батареї в 3-5 разів вище, ніж в інших технологіях.

Ємність мережі. Щоб забезпечити оптимальну продуктивність мережі Шлюз повинен мати дуже високу пропускну здатність або прийом інформація з величезної кількості кінцевих пристроїв. Висока пропускну здатність мережі LoRaWAN досягається за допомогою швидкості самонастроювання велика пропускну здатність мережі LoRaWAN досягається за рахунок використання самонастроюваної швидкості передачі даних, а також багатоканального передавача у шлюзі. Це дозволяє увімкнути одночасний прийом повідомлень кілька каналів одночасно.

Шлюзи дозволяють одночасно приймати один канал інформація з пристроїв з різною швидкістю передачі даних. Адаптивний швидкість передачі даних також збільшує термін служби акумулятора.

Мережі LoRaWAN розгортаються з мінімальною кількістю інфраструктури. Зі збільшенням кількості пристроїв у мережі це можливо швидкість передачі даних можна змінити, а також збільшити кількість шлюзів.

Стійкість до радіозавад. Мережа має високу проникність здатність радіосигналу. Це забезпечить стабільне з'єднання з іншими технологіями технології не впораються. Модеми LoRa можуть пригнічувати перешкоди до 19, 5 дБ (фільтрація Гаусса). Така здатність придушувати перешкоди дозволить використовувати систему в сучасному великому місті.

LoRa використовує комбінацію змінної пропускну здатності та коефіцієнтів поширення (SF7-SF12), щоб адаптувати швидкість передачі даних до діапазону передачі. Вищий коефіцієнт розповсюдження забезпечує більший діапазон за рахунок нижчої швидкості передачі даних і навпаки. Комбінація пропускну здатності та коефіцієнта розширення може бути обрана відповідно до умов з'єднання та рівня даних, що підлягають передачі. Таким чином, більш високий коефіцієнт розширення покращує продуктивність передачі та чутливість для заданої пропускну здатності, але також збільшує час передачі в результаті нижчих швидкостей передачі даних. Вони можуть коливатися від 18 до 40 кбіт/с.

LoRaWAN забезпечує повну конфіденційність даних для всіх пристроїв, які беруть участь у ланцюжку, при цьому вміст пакета доступний лише відправнику (кінцевому пристрою) і одержувачу (додатку), для якого він призначений.

Мережевий сервер оперує даними в зашифрованому вигляді, виконує аутентифікацію та перевіряє цілісність кожного пакета, але не має доступу до корисного навантаження, тобто до інформації від підключених датчиків (за винятком використання nereкомендованих сценаріїв, у яких мережевий сервер виконує шифрування корисного навантаження за допомогою NwkSKey, але не сервер додатків; ми не будемо розглядати цей сценарій далі).

У мережі використовуються три типи ключів. Ключ автентифікації програми AppKey відомий лише кінцевому пристрою та серверу програм. Якщо кінцева точка підключена до мережі в режимі Over-The-Air-Activation (OTAA), ключ аутентифікації AppKey використовується для обчислення мережевого ключа NwkSKey і ключа програми AppSKey. Якщо кінцевий пристрій підключається до мережі в режимі Activation By Personalization (ABP), ключі NwkSKey і AppSKey попередньо встановлені на кінцевому пристрої. NwkSKey відомий мережевому серверу та кінцевому пристрою і використовується для перевірки цілісності кожного повідомлення за допомогою коду цілісності повідомлення (MIC). MIC обчислюється за допомогою алгоритму AES-CMAC, який подібний до контрольної суми, за винятком того, що він запобігає навмисному підробці повідомлення. Ключ програми AppSKey використовується для шифрування корисного навантаження за допомогою алгоритму AES-128 між кінцевим пристроєм і сервером додатків.

Вузли в мережі LoRaWAN є асинхронними і спілкуються, коли вони мати готові дані для надсилання, незалежно від того, керуються вони подіями або заплановані. Цей тип протоколу є зазвичай називають методом Алохи. У сітчастій мережі або з синхронною мереж, таких як стільниковий, вузлам часто доводиться

"прокидатися" для синхронізації

мережу та перевірити наявність повідомлень. Ця синхронізація вимагає значних витрат енергії і є драйвером номер один для скорочення терміну служби акумулятора. У недавньому вивченні та порівнянні GSMА різних технологій, що стосуються. Простір LPWAN, LoRaWAN показав перевагу в 3 - 5 разів порівняно з усіма іншими технологічними варіантами.

Для того, щоб зробити зіркову мережу далекого радіусу дії життєздатною, шлюз повинен мати дуже високий рівень ємності або можливість приймати повідомлення з дуже великого обсягу вузлів. Високий ємність мережі в мережі LoRaWAN досягається за рахунок використання адаптивних даних швидкість і за допомогою багатоканального багатомодемного трансивера в шлюзі, так що можна одночасно отримувати повідомлення на декількох каналах. Критичні фактори ефективною ємністю є кількість одночасних каналів, швидкість передачі даних (час в ефірі), довжину корисного навантаження та як часто передають вузли. Оскільки LoRa - це модуляція з розповсюдженим спектром, сигнали практично ортогональні один одному, коли різні використовуються фактори розповсюдження. Зі зміною коефіцієнта розповсюдження ефективна швидкість передачі даних також змінюється. Шлюз використовує цю властивість, отримуючи можливість прийому декілька різних швидкостей передачі даних на одному каналі одночасно. Якщо вузол має хороше посилення і близько до шлюзу, немає підстав завжди використовувати найнижчий швидкість передачі даних та заповнення наявного спектру довше, ніж це необхідно. Змістивши швидкість передачі даних вища, час в ефірі скорочується, відкриваючи більше потенційного простору для інші вузли для передачі. Адаптивна швидкість передачі даних також оптимізує час роботи акумулятора вузол. Для того, щоб адаптивна швидкість передачі даних працювала, симетричне посилення вгору та вниз потрібна з достатньою пропускною здатністю низхідної лінії зв'язку. Ці функції дозволяють

використовувати LoRaWAN мережа має дуже високу ємність і робить мережу масштабованою. Мережа може бути розгорнутою з мінімальною кількістю інфраструктури, а в міру необхідності потужності, можна додати більше шлюзів, зміщуючи швидкість передачі даних, зменшуючи їх кількість підслуховування до інших шлюзів і збільшення потужності в 6-8 разів. Інші LPWAN альтернативи не мають масштабованості LoRaWAN через компроміси технологій, які обмежують пропускну здатність низхідної лінії зв'язку або роблять діапазон низхідної лінії зв'язку асиметричним до діапазон висхідної лінії зв'язку.

Впровадження концепції «Розумного міста» має впроваджуватися як на нових об'єктах, так і на існуючих застарілих системах. Інфраструктура міст, житловий фонд, наявні системи обліку, дорожнє покриття не готові до переобладнання ні з технологічної точки зору ні з економічної. На сьогоднішній день можлива часткова і поступова заміна і оптимізація окремих систем і обладнання на нові і ефективні.

Також із впровадженням систем концепції «Розумного міста» виникає ряд ризиків:

1. Технологічні

- Ріст обсягів даних, що передаються різного виду мережами і перевантаження мереж «інформаційним сміттям»
- Ризики вразливості мереж з точки зору кібербезпеки і захисту даних
- Ризики перевантаження систем зберігання і обробки даних, консолідація чутливих даних в єдиних ситуаційних центрах – фізична безпека ЦОДів, інформаційна безпека (кіберзахист), безпека і забезпечення безперервної роботи інженерних систем (надійність гарантованого електроживлення, кондиціонування та інше)

2. Екологічні

- Зростання обсягів заміни обладнання і утилізації застарілих систем
- Зростання кількості «технологічного сміття»

3. Економічні

- Дефіцит бюджету та недостатній рівень інвестицій в модернізацію активів і обладнання
- Відсутність мотивації для виробників енергоресурсів до скорочення споживання
- Неможливість масштабування рішень для невеликих громад через високу вартість проекту



Рисунок 2.2 Сильні сторони автоматизації

5. Соціально-правові

- Ризик витоку персональних даних
- Ризик цифрової нерівності через неможливість контролювати коректність даних і через можливі помилки в базах даних
- Ризик психічного дискомфорту через надмірний контроль людини

3 РОЗРОБКА СИСТЕМИ "РОЗУМНОГО МІСТА"

3.1 Вибір програмного забезпечення

OpenHAB - це основана на Java платформа домашньої автоматизації з відкритим вихідним кодом, яка об'єднує широкий спектр різних систем і технологій «розумного будинку» в одне єдине рішення. На вершині уніфікованого рівня абстракції доступні всі підключені пристрої для всеосяжних механізмів правил автоматизації і різні користувальницькі інтерфейси. Більше 200 спеціальних додатків забезпечують підтримку брендів, пристроїв, технологій і протоколів зв'язку. Прикладами є Z-Wave, Philips Hue, Amazon Echo, Chromecast і Sonos.

- Рішення для домашньої автоматизації: Z-Wave, EnOcean, Netatmo, Homematic, Insteon.

- Освітлення: Philips Hue, Ikea Tradfri, LIFX, Lutron, Milight.

- Опалення: Max!, Nest, Vitotronic, Heatmiser.

- Розваги для будинку: телевізор Samsung, LG TV, Sonos, Pioneer AVR, Squeezebox, Kodi, Plex.

- Безпека: ZoneMinder, DSC.

- Відкриті протоколи: HTTP, TCP / UDP, MQTT, Serial. Деякі з сильних сторін openHAB:

- Його здатність інтегрувати безліч інших пристроїв і систем. openHAB включає в себе єдині системи домашньої автоматизації, (розумні) пристрої та інші технології

- Забезпечити єдиний користувальницький інтерфейс та загальний підхід до правил автоматизації для всієї системи, незалежно від кількості залучених виробників та підсистем

- Надання вам найбільш гнучких інструментів, що дозволяють здійснити практично будь-яке бажання домашньої автоматизації; якщо ви можете це думати, шанси на те, що ви можете реалізувати це з openHAB. 35 Таким чином, ми отримаємо доступний, керований та зручний для використання спосіб керування та

автоматизації системи клімат контролю. openHAB розроблений на Java і використовує OSGi для модульності. Apache Karaf служить контейнером із Eclipse Equinox як середовище виконання OSGi. Jetty включений як сервер HTTP. Технологія OSGi складається з набору специфікацій, еталонної реалізації для кожної специфікації та набору тестів на відповідність для кожної специфікації, які разом визначають динамічну модульну систему для Java. OSGi забезпечує незалежний від постачальників підхід, орієнтований на стандарти, до модуляції програмних засобів та інфраструктури Java. Його перевірена модель послуг дає можливість прикладним та інфраструктурним модулям спілкуватися локально та розповсюджуватися по всій мережі, забезпечуючи цілісну цілісну архітектуру. Технічні характеристики OSGi пройшли випробування на місцях і готові до використання. Наразі, випуск 7, OSGi забезпечив стабільну та розвинуту технологічну платформу для розробки проектів з відкритим кодом та комерційних продуктів протягом майже двох десятиліть. З точки зору коду, Equinox - це реалізація основної специфікації основної OSGi, набору пакетів, що реалізують різні необов'язкові сервіси OSGi та іншу інфраструктуру для запускених OSGi систем. Реалізація основної системи Equinox OSGi використовується як опорна реалізація, і як така вона реалізує всі необхідні функції останньої специфікації базової основи OSGi. Загалом, метою проекту Equinox є створення спільноти OSGi першого класу та сприяння баченню Eclipse як ландшафту розшарувань. В рамках цього він несе відповідальність за розробку та доставку OSGi-рамкової реалізації, що використовується для всіх Eclipse. На додачу. проект відкритий для:

- Впровадження всіх аспектів специфікації OSGi. Основна увага приділяється специфікаціям, що підтримуються Експертною групою основної 36 платформи (CPEG), але також може включати в себе реалізації, визначені іншими експертними групами, такими як Enterprise Expert Group (EEG)

- Дослідження та дослідження, пов'язані з майбутніми версіями специфікацій OSGi та пов'язаними з ними проблемами виконання

- Розвиток нестандартної інфраструктури, що вважається важливим для роботи та управління системами на базі OSGi

- Впровадження ключових базових служб та розширень, необхідних для роботи Eclipse (наприклад, адаптер Eclipse, реєстр розширень) і вважається загалом корисним для людей, що використовують OSGi. Як частина проекту RT, кодом OSGi Equinox керує RT PMC. Однак існує тісний зв'язок із проектом Eclipse та його PMC. Рівнодення, як база для всіх Eclipse, поставляється з усіма основними випусками. Різні інші пакети, розроблені тут, можуть відправлятися незалежно та за різними графіками. Eclipse Jetty забезпечує веб-сервер і контейнер javax.servlet, а також підтримку HTTP / 2, WebSocket, OSGi, JMX, JNDI, JAAS та багатьох інших інтеграцій. Ці компоненти є відкритим кодом та доступні для комерційного використання та розповсюдження. Eclipse Jetty використовується в найрізноманітніших проектах та продуктах, як у розробці, так і у виробництві. Jetty може бути легко вбудований у пристрої, інструменти, рамки, сервери додатків та кластери. Дивіться сторінку на базі Jetty, щоб дізнатися більше про використання Jetty. OpenNAV - це високомодульне програмне забезпечення, яке можна розширити через "Додатки". Додатки надають openNAV широкий спектр можливостей, починаючи від користувальницьких інтерфейсів, до можливості взаємодіяти з великою і зростаючою кількістю фізичних речей. Додатки можуть надходити з дистрибутива openNAV 2, застарілого дистрибутива openNAV 1 або з інших зовнішніх джерел.



Рисунок 3.1 openhab

3.2 Встановлення openHAB на операційну систему Windows 10

Передумови

Обов'язковою умовою для використання є встановлена оновлена платформа Java 8. На даний момент Zulu є рекомендованою платформою Java для openHAB. OpenHAB не працює добре з новішими версіями Java, такими як Java 9 або Java 11.

Zulu - повністю відкрита версія, повністю сертифікована збірка open SE для Java SE, вбудована версія, оптимізована для пристроїв ARM.

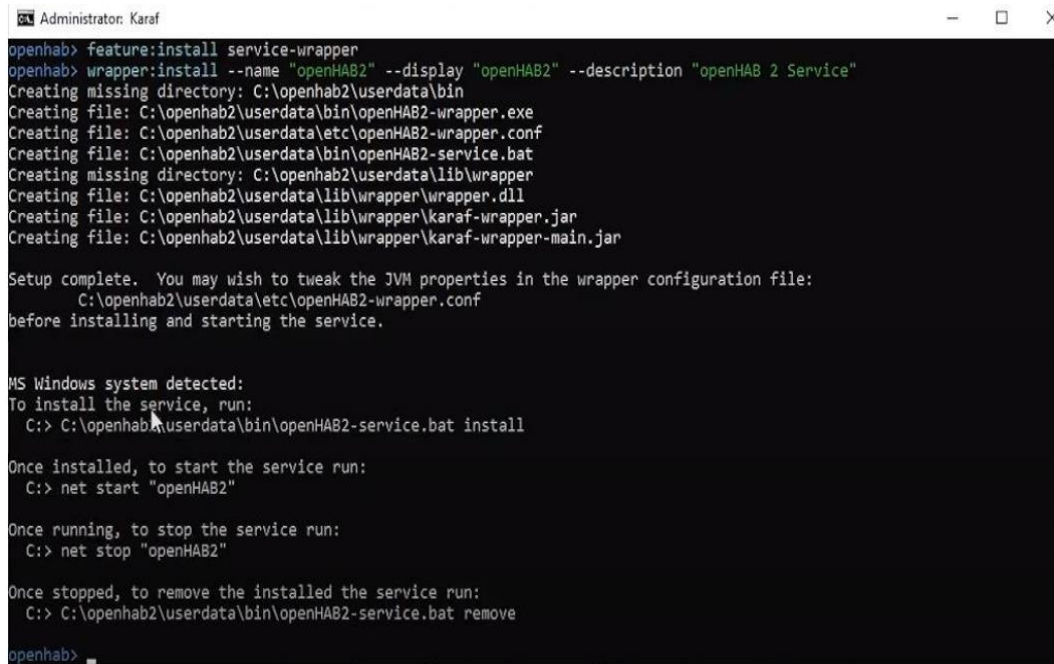
На платформах з використанням 64-розрядних ОС і процесор Intel або AMD рекомендується 64-розрядна версія на JVM

На ARM платформах, таких як Raspberry Pi рекомендується 32-розрядна версія з віртуальної машини Java. Перш ніж почати, необхідно прийняти рішення Stable або Snapshot.

Stable - використовує останній офіційний реліз.

Snapshot - нестабільна версія з щодневними змінами та додатками.

Розпакувавши файл у вибраному каталозі запускаємо сценарій start.bat



```
Administrator: Karaf
openhab> feature:install service-wrapper
openhab> wrapper:install --name "openHAB2" --display "openHAB2" --description "openHAB 2 Service"
Creating missing directory: C:\openhab2\userdata\bin
Creating file: C:\openhab2\userdata\bin\openHAB2-wrapper.exe
Creating file: C:\openhab2\userdata\etc\openHAB2-wrapper.conf
Creating file: C:\openhab2\userdata\bin\openHAB2-service.bat
Creating missing directory: C:\openhab2\userdata\lib\wrapper
Creating file: C:\openhab2\userdata\lib\wrapper\wrapper.dll
Creating file: C:\openhab2\userdata\lib\wrapper\karaf-wrapper.jar
Creating file: C:\openhab2\userdata\lib\wrapper\karaf-wrapper-main.jar

Setup complete. You may wish to tweak the JVM properties in the wrapper configuration file:
C:\openhab2\userdata\etc\openHAB2-wrapper.conf
before installing and starting the service.

MS Windows system detected:
To install the service, run:
C:> C:\openhab2\userdata\bin\openHAB2-service.bat install

Once installed, to start the service run:
C:> net start "openHAB2"

Once running, to stop the service run:
C:> net stop "openHAB2"

Once stopped, to remove the installed the service run:
C:> C:\openhab2\userdata\bin\openHAB2-service.bat remove

openhab>
```

Рисунок 3.3 - Виконання команд у командній строці

Щоб оновлення вступили в силу скопіюємо весь конфігураційний текст із інструкції з офіційного сайту openHAB та вставляємо його у свій «openHAB2- wrapper.conf»

```

#####
# openHAB2-wrapper.conf for Windows Service Installation
#####

# openHAB installation dir (Adapt this first setting to your system)
set.default.OPENHAB_HOME=C:\openhAB2

# Wrapper Properties
set.default.OPENHAB_CONF=${OPENHAB_HOME}\conf
set.default.OPENHAB_RUNTIME=${OPENHAB_HOME}\runtime
set.default.OPENHAB_USERDATA=${OPENHAB_HOME}\userdata
set.default.OPENHAB_LOGDIR=${OPENHAB_USERDATA}\logs
set.default.KARAF_HOME=${OPENHAB_RUNTIME}
set.default.KARAF_BASE=${OPENHAB_USERDATA}
set.default.KARAF_DATA=${OPENHAB_USERDATA}
set.default.KARAF_ETC=${OPENHAB_USERDATA}\etc
set.default.PATH=${PATH};%KARAF_BASE%\lib;%KARAF_HOME%\lib

# Java Application
wrapper.working.dir=${KARAF_BASE}
wrapper.java.command=${KARAF_HOME}\bin\java
wrapper.java.mainclass=org.apache.karaf.wrapper.internal.service.Main
wrapper.java.classpath.1=${KARAF_HOME}\lib\boot\*.jar
wrapper.java.classpath.2=${KARAF_DATA}\lib\wrapper\*.jar
wrapper.java.library.path.1=${KARAF_DATA}\lib\wrapper\

# Java Parameters
wrapper.java.additional.1=-Dkaraf.home="${KARAF_HOME}"
wrapper.java.additional.2=-Dkaraf.base="${KARAF_BASE}"
wrapper.java.additional.3=-Dkaraf.data="${KARAF_DATA}"
wrapper.java.additional.4=-Dkaraf.etc="${KARAF_ETC}"
wrapper.java.additional.5=-Dcom.sun.management.jmxremote
wrapper.java.additional.6=-Dkaraf.startLocalConsole=false
wrapper.java.additional.7=-Djava.startRemoteShell=true
wrapper.java.additional.8=-Djava.endorsed.dirs="${KARAF_HOME}\jre\lib\endorsed;%KARAF_HOME%\lib\endorsed;%KARAF_HOME%\lib\endorsed"
wrapper.java.additional.9=-Djava.ext.dirs="${KARAF_HOME}\jre\lib\ext;%KARAF_HOME%\lib\ext;%KARAF_HOME%\lib\ext"
wrapper.java.additional.10=-Dopendub.home="${OPENHAB_HOME}"
wrapper.java.additional.11=-Dopendub.conf="${OPENHAB_HOME}\conf"
wrapper.java.additional.12=-Dopendub.runtime="${OPENHAB_HOME}\runtime"
wrapper.java.additional.13=-Dopendub.userdata="${OPENHAB_HOME}\userdata"
wrapper.java.additional.14=-Dopendub.logdir="${OPENHAB_USERDATA}\logs"
wrapper.java.additional.15=-Dfelix.cmdir="${OPENHAB_HOME}\userdata\config"
wrapper.java.additional.16=-Dorg.osgi.service.http.port=8080
wrapper.java.additional.17=-Dorg.osgi.service.http.port.secure=8443
wrapper.java.additional.18=-Djava.util.logging.config.file="${KARAF_ETC}\java.util.logging.properties"
wrapper.java.additional.19=-Dkaraf.logs="${OPENHAB_LOGDIR}"
wrapper.java.additional.20=-Dfile.encoding=UTF-8
wrapper.java.memory=512

# Wrapper Logging Properties
wrapper.console.format=PN
wrapper.console.loglevel=INFO
wrapper.logfile=${OPENHAB_USERDATA}\logs\wrapper.log
wrapper.logfile.format=LPTM
wrapper.logfile.loglevel=INFO
wrapper.logfile.maxsize=10m
wrapper.logfile.maxfiles=5
wrapper.spring.loglevel=NONE

# Wrapper Windows Properties
wrapper.console.title=openHAB2
wrapper.ntsservice.name=openHAB2
wrapper.ntsservice.displayname=openHAB2
wrapper.ntsservice.description=openHAB 2 Service
wrapper.ntsservice.dependency.1=
wrapper.ntsservice.starttype=AUTO_START
wrapper.ntsservice.interactive=false

```

Рисунок 3.4 Конфігураційний текст із інструкції з офіційного сайту openHAB

Наступним кроком є введення у командну строку наступних команд:

- C:\openhAB2\userdata\bin\openhAB2-service.bat install
- net start "openhAB2"

```

Administrator: Command Prompt
Microsoft Windows [Version 10.0.17763.475]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>C:\openHAB2\userdata\bin\openHAB2-service.bat install
wrapper | openHAB2 installed.

C:\WINDOWS\system32>net start "openHAB2"
The openHAB2 service is starting..
The openHAB2 service was started successfully.

C:\WINDOWS\system32>

```

Рисунок 3.5 Введення необхідних команд

Перевіряємо чи openHAB працює як сервіс Windows і встановлено тип автоматичного запуску

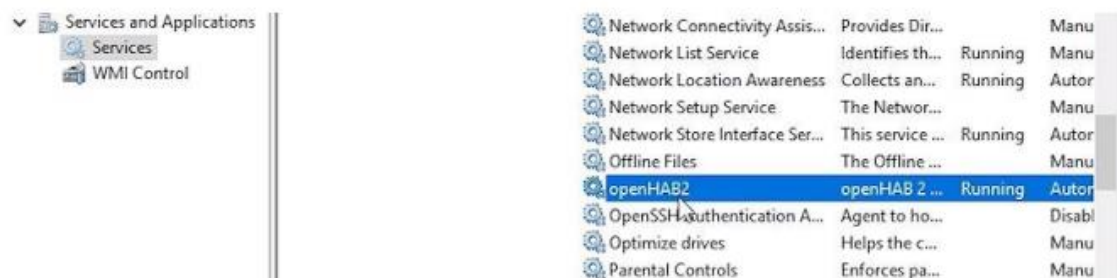


Рисунок 3.6 Сервіси Windows

На данному етапі openHAB встановлений на на операційну систему Windows 10.

3.3 Встановлення MQTT broker на WINDOWS 10

Для початку необхідно встановити Microsoft Visual C++ 2017 та Mosquitto MQTT Broker

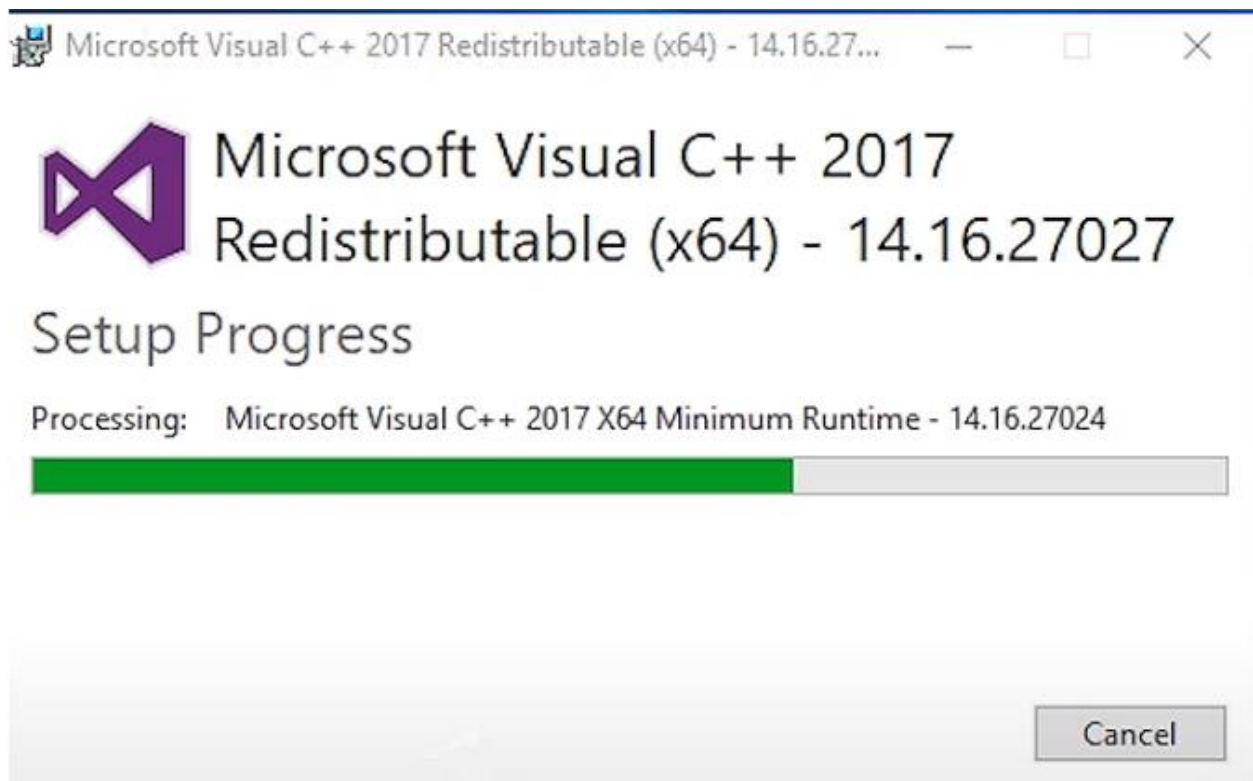


Рисунок 3.7 Встановлення Microsoft Visual C++ 2017

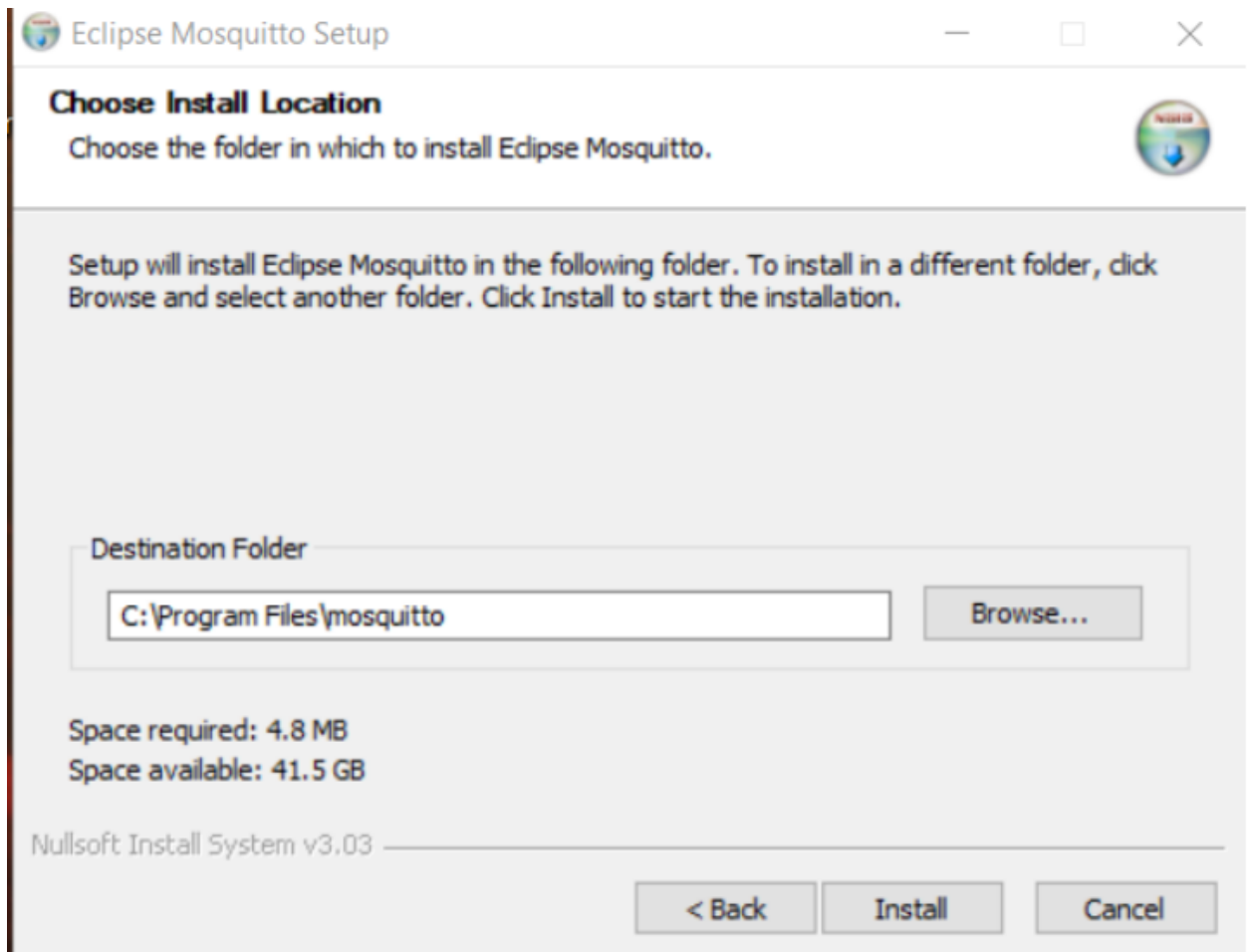


Рисунок 3.8 Встановлення Mosquitto MQTT Broker

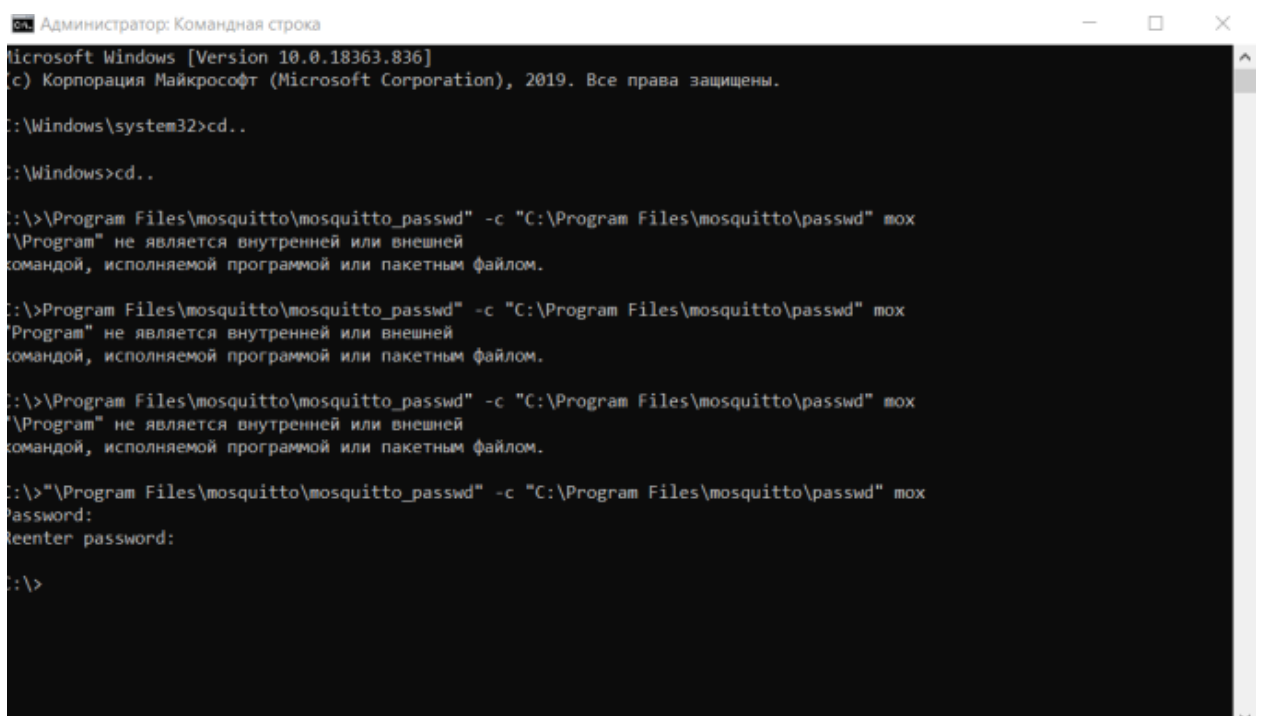
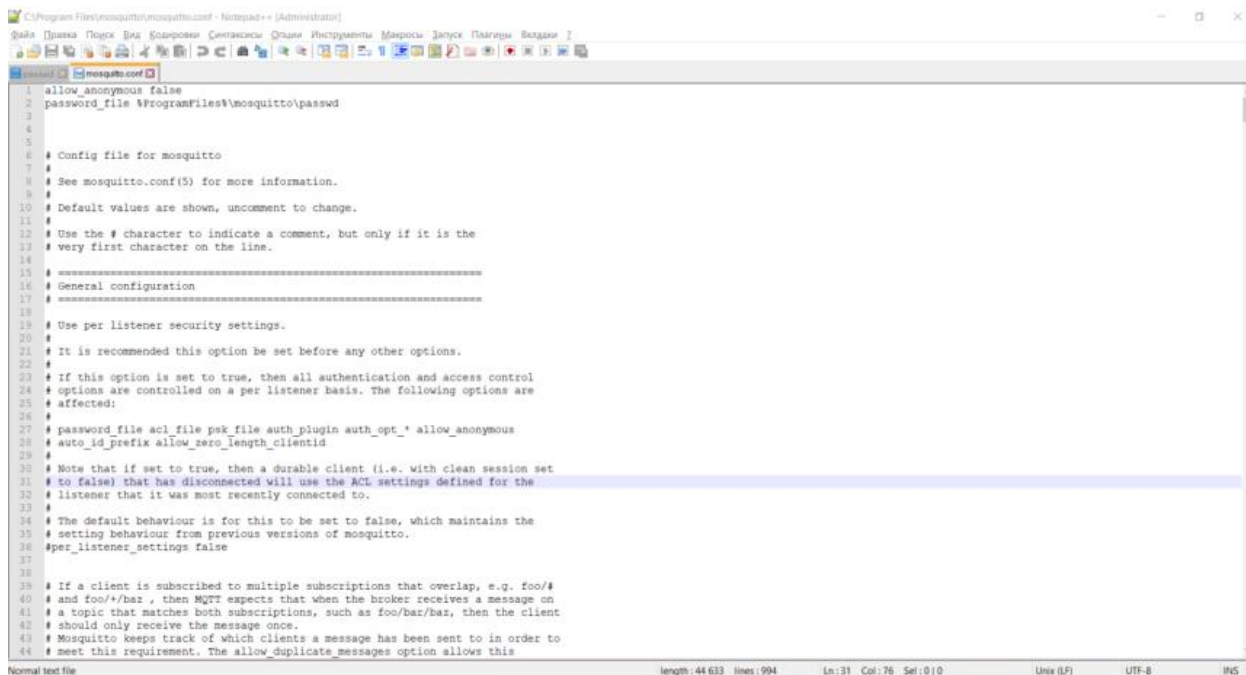


Рисунок 3.9 Створення файлу з паролем



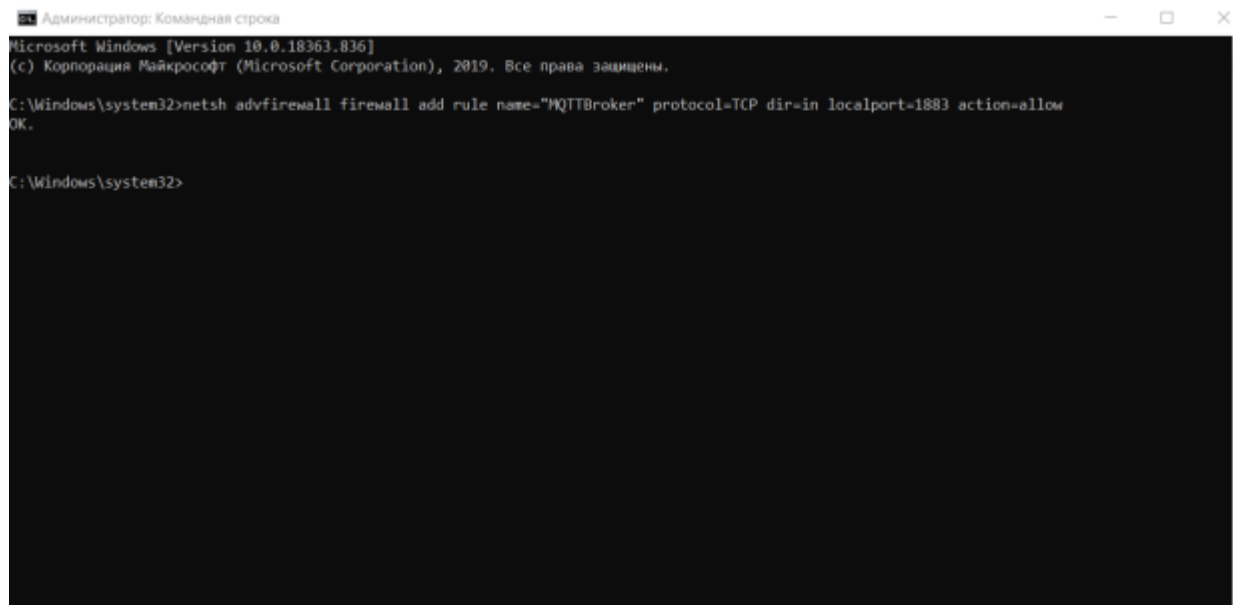
```

1 allow_anonymous false
2 password_file %ProgramFiles%\mosquitto\passwd
3
4
5
6 # Config file for mosquitto
7 #
8 # See mosquitto.conf(5) for more information.
9 #
10 # Default values are shown, uncomment to change.
11 #
12 # Use the # character to indicate a comment, but only if it is the
13 # very first character on the line.
14 #
15 # =====
16 # General configuration
17 # =====
18 #
19 # Use per listener security settings.
20 #
21 # It is recommended this option be set before any other options.
22 #
23 # If this option is set to true, then all authentication and access control
24 # options are controlled on a per listener basis. The following options are
25 # affected:
26 #
27 # password_file acl_file psk_file auth_plugin auth_opt_* allow_anonymous
28 # auto_id_prefix allow_zero_length_clientid
29 #
30 # Note that if set to true, then a durable client (i.e. with clean session set
31 # to false) that has disconnected will use the ACL settings defined for the
32 # listener that it was most recently connected to.
33 #
34 # The default behaviour is for this to be set to false, which maintains the
35 # setting behaviour from previous versions of mosquitto.
36 #per_listener_settings false
37 #
38 #
39 # If a client is subscribed to multiple subscriptions that overlap, e.g. foo/#
40 # and foo//bar, then MQTT expects that when the broker receives a message on
41 # a topic that matches both subscriptions, such as foo/bar/bar, then the client
42 # should only receive the message once.
43 # Mosquitto keeps track of which clients a message has been sent to in order to
44 # meet this requirement. The allow_duplicate_messages option allows this

```

Рисунок 3.10 Файл конфігурації

Наступним кроком додаємо порт 1883 у винятки фаєрвола.



```

Администратор: Командная строка
Microsoft Windows [Version 10.0.18363.836]
(c) Корпорация Майкрософт (Microsoft Corporation), 2019. Все права защищены.

C:\Windows\system32>netsh advfirewall firewall add rule name="MQTTBroker" protocol=TCP dir=in localport=1883 action=allow
OK.

C:\Windows\system32>

```

Рисунок 3.11- Відкриття порту 1883

Встановлюємо програму MQTT.fx для перевірки налаштованого MQTT брокера

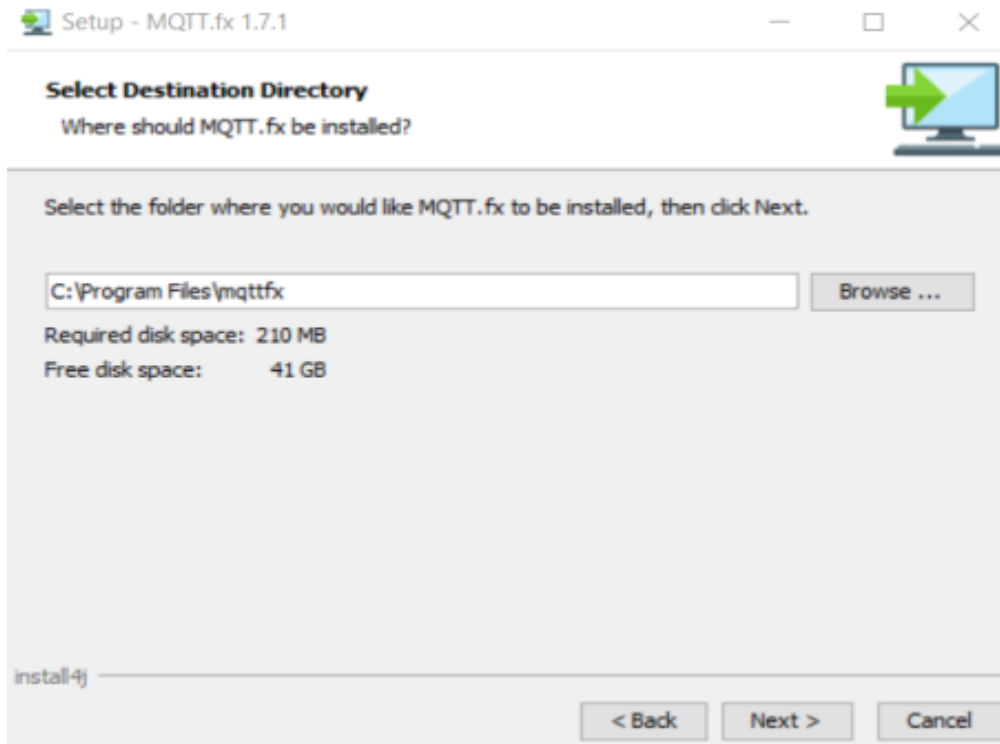


Рисунок 3.12 Встановлення програми MQTT.fx

Відкриваємо встановлену програму і вводимо свої данні

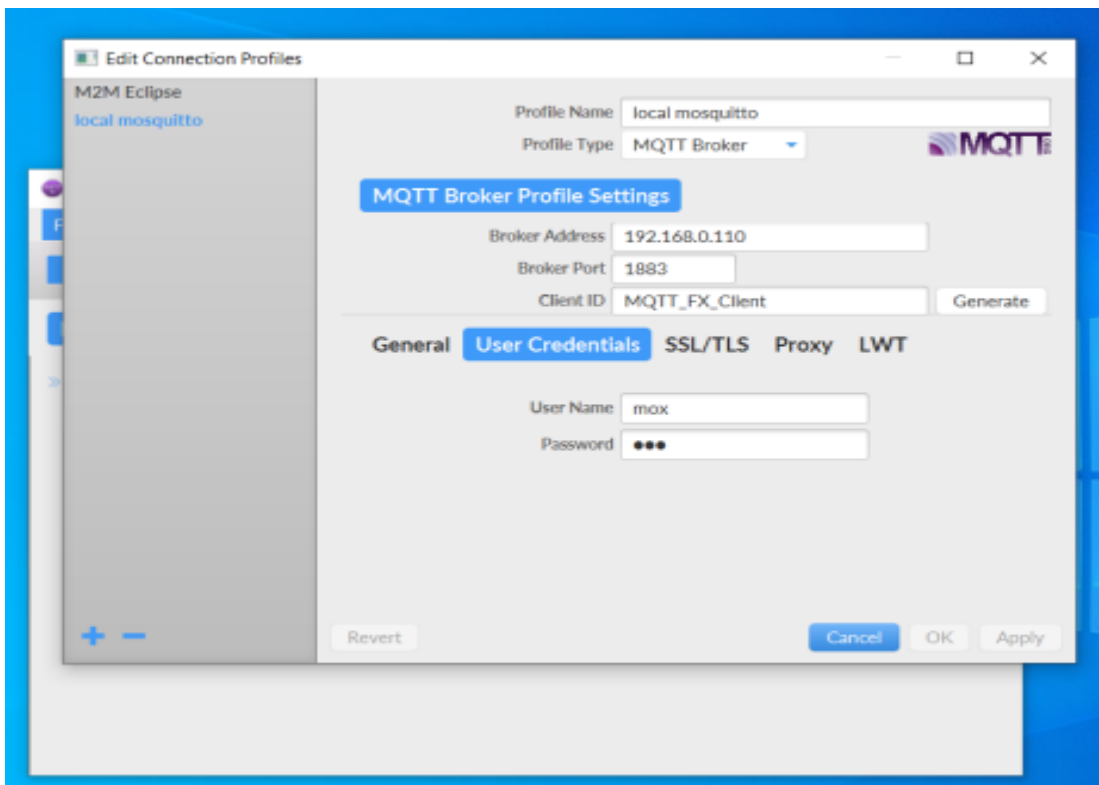


Рисунок 3.13 Налаштування MQTT.fx

Налаштувавши з'єднання через 1883 порт який є стандартним проводимо тести. У вкладці subscribe створюємо топик «test/». У вкладці publish і публікуємо записи у створений раніше топик.

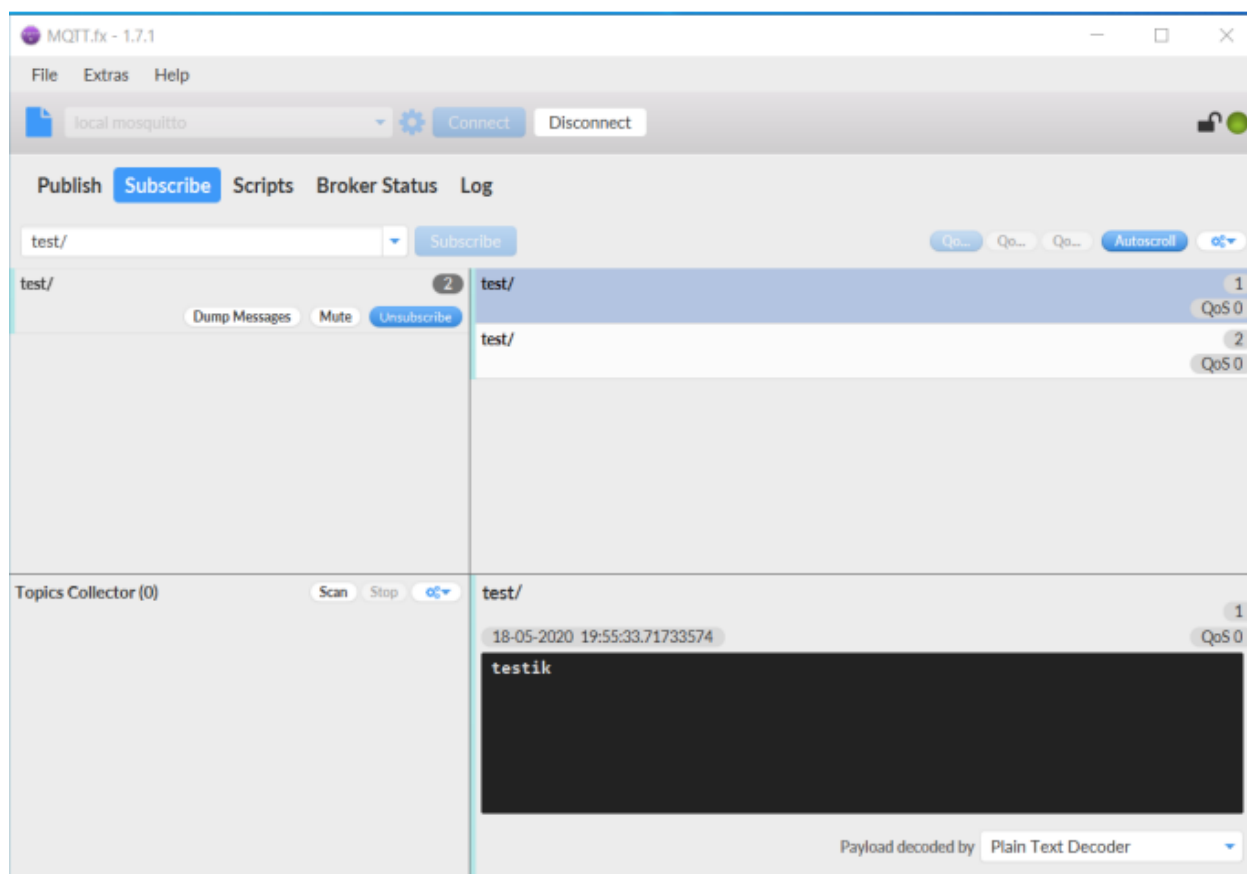


Рисунок 3.14 Перший тест працездатності MQTT брокера

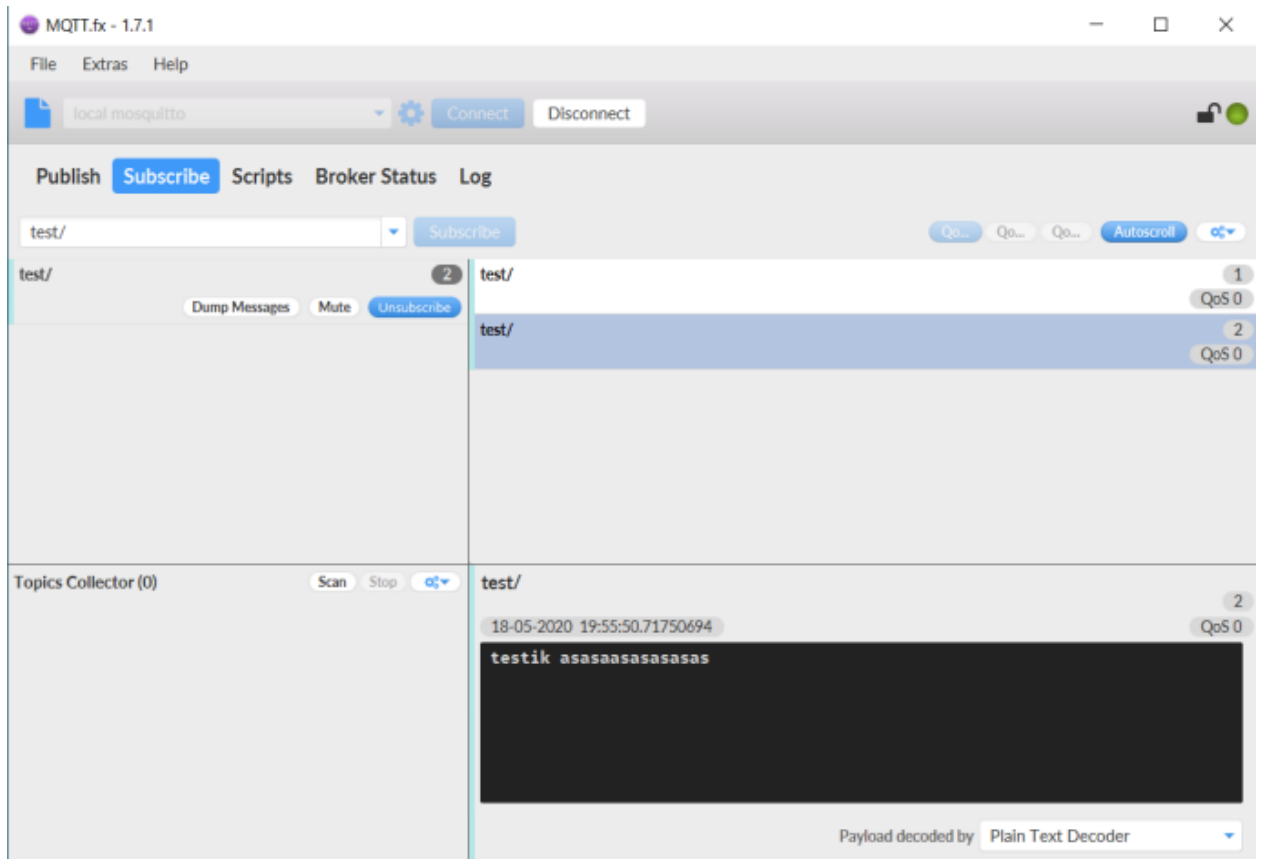


Рисунок 3.15 Другий тест працездатності MQTT брокера

Таким чином завдяки програмі MQTT.fx був перевірений MQTT брокер на працездатність.

3.4 Налаштування та підключення openHAB до MQTT брокера

Для початку заходимо на веб-інтерфейс openHAB «<http://localhost:8080>»
Переходимо у вкладку add-ons і встановлюємо необхідний MQTT binding.

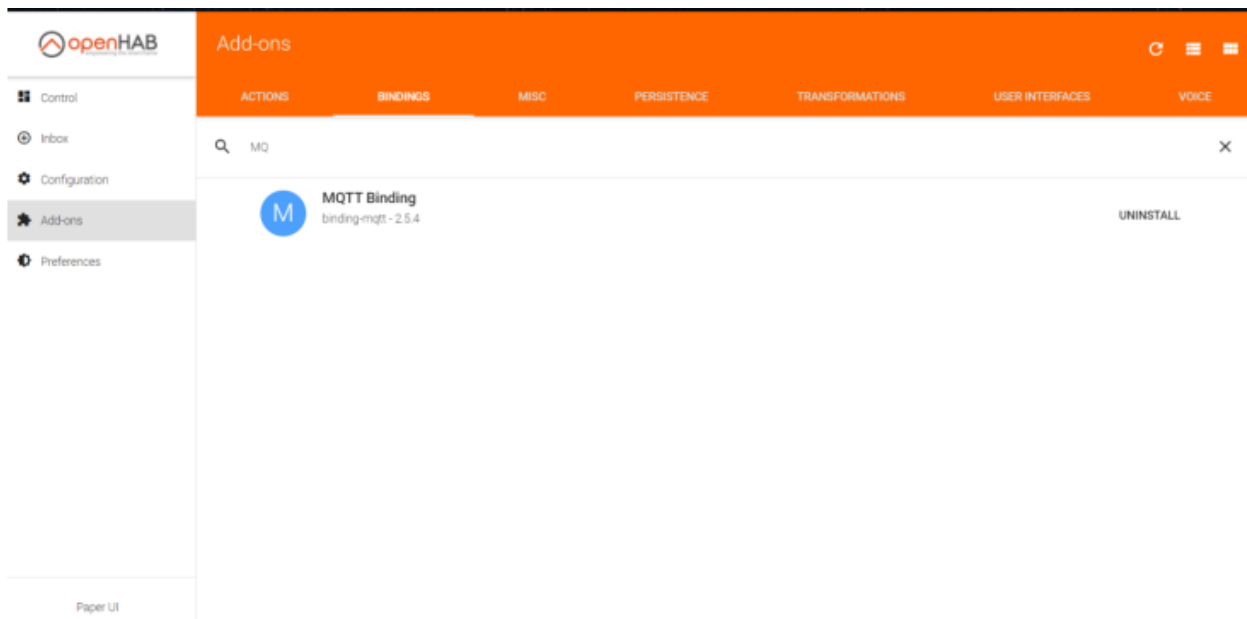


Рисунок 3.16 Панель керування openHAB

Наступним кроком буде налаштування MQTT binding. Необхідно перейти у вкладку Inbox. Ввівши налаштування раніше створеного MQTT брокера він буде підключений до openHAB.

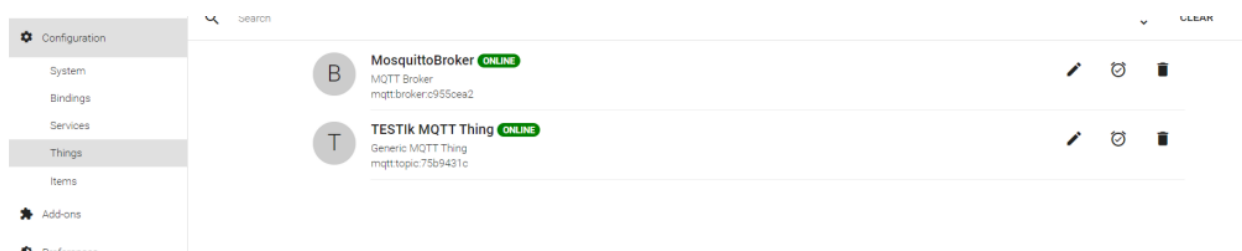


Рисунок 3.17 Підключені до openHAB прилади

Щоб перевірити працездатність підключення створюємо тест. Переходимо у вкладку Inbox, та генеруємо MQTT Thing.

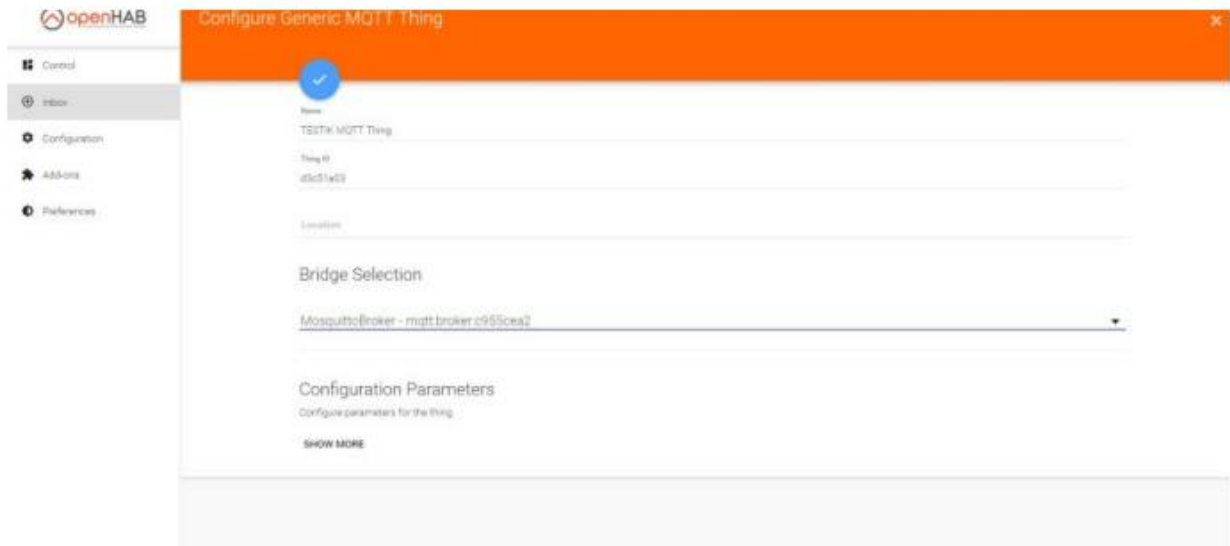


Рисунок 3.18 Генератор запитів

Наступним кроком буде створення items за шляхом файлу «C:\openHAB2\conf\items» та sitemap «C:\openHAB2\conf\sitemaps»

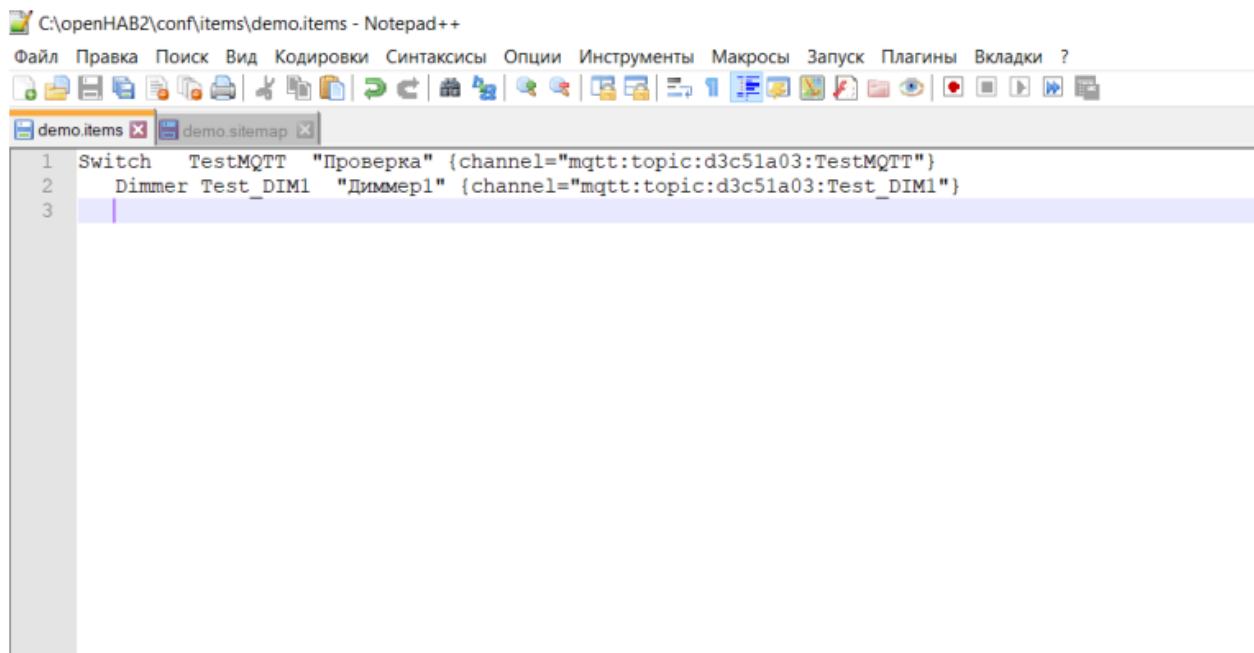
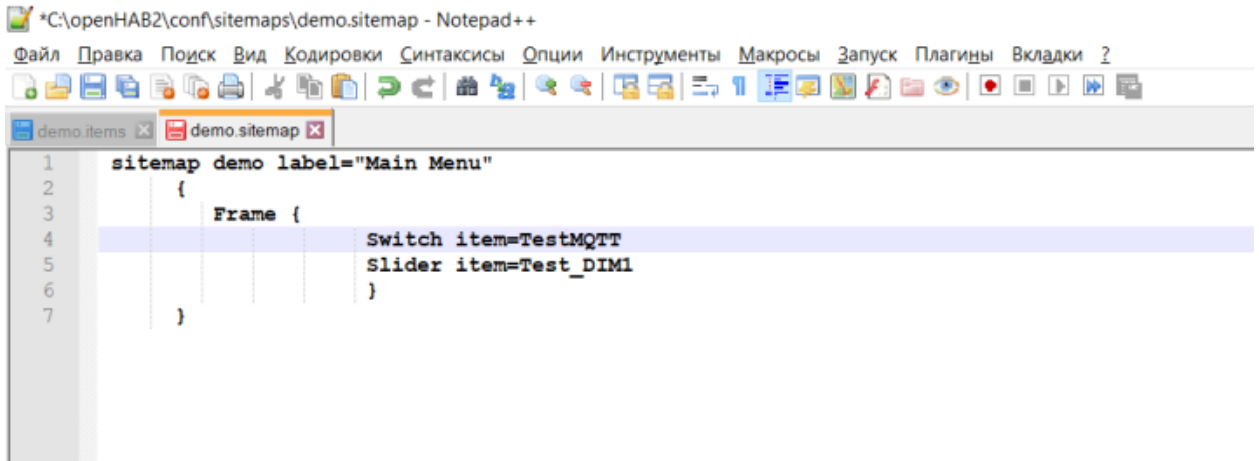


Рисунок 3.19 - Test item



```

1  sitemap demo label="Main Menu"
2  {
3      Frame {
4          Switch item=TestMQTT
5          Slider item=Test_DIM1
6      }
7  }

```

Рисунок 3.20 Test sitemap

Після створення файлу з необхідними налаштуваннями у веб-інтерфейсі openHAB з'являється віджет для керування сгенерованим приладом.

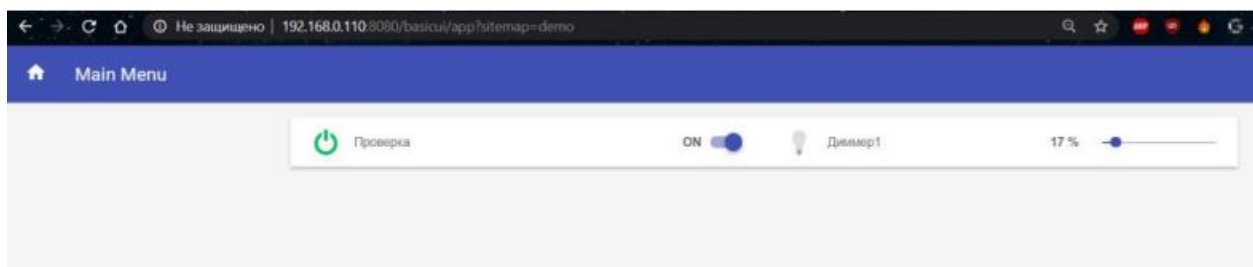


Рисунок 3.21 Віджет керування сгенерованим приладом

Описаним вище кодом було створено два канали які передають інформацію про зміни значень з панелі керування. Перевірити їх працездатність можливо за допомогою програми MQTT.fx.

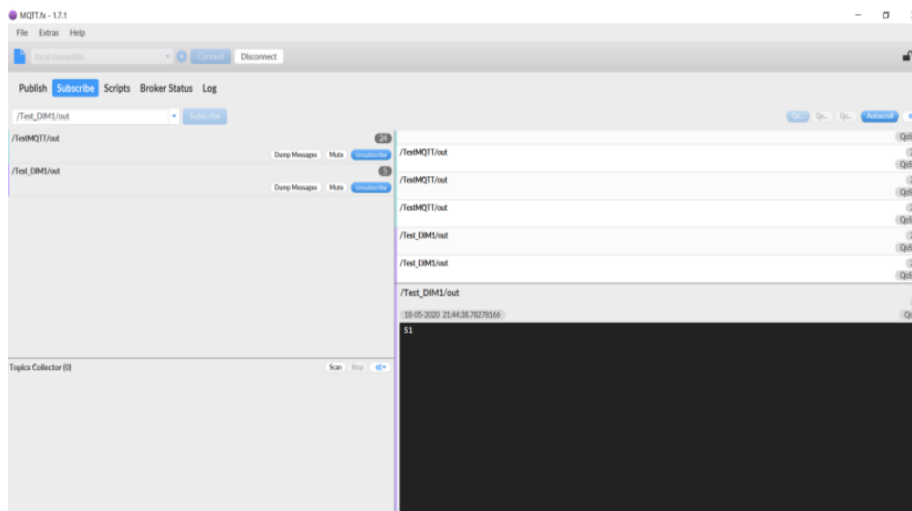


Рисунок 3.22 Тест підключення MQTT до openHAB

Вище була наочно продемонстрована можливість відправляти інформацію з панелі керування openHAB на MQTT брокер. Є можливість надсилати інформацію з MQTT брокера в openHAB.

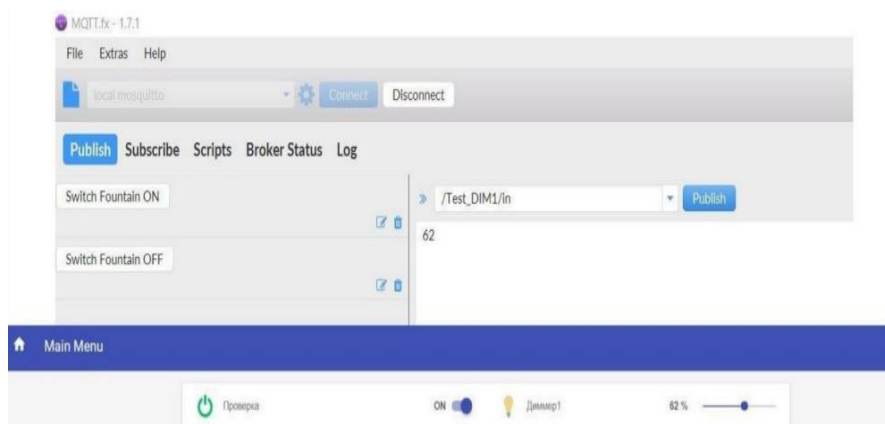


Рисунок 3.23 Тест працездатності віджета

Таким чином MQTT брокер був вдало під'єднаний до openHAB и пройшов усі тести на працездатність.

3.5 Підключення шлюз LoRaWAN до MQTT

Шлюзи Milesight LoRaWAN підтримують відправку пакетів даних на сторонні MQTT/HTTP/HTTPS-сервер. Ми можемо створити нову програму на шлюзі, яка може визначити метод декодування даних, надісланих з кінцевого пристрою LoRaWAN, і вибрати протокол транспортування даних MQTT для надсилання даних на сервер MQTT.

Увімкнути вбудований мережевий сервер шлюзу.

Перейдіть до Packet Forwarder > General, щоб увімкнути адресу локального сервера.

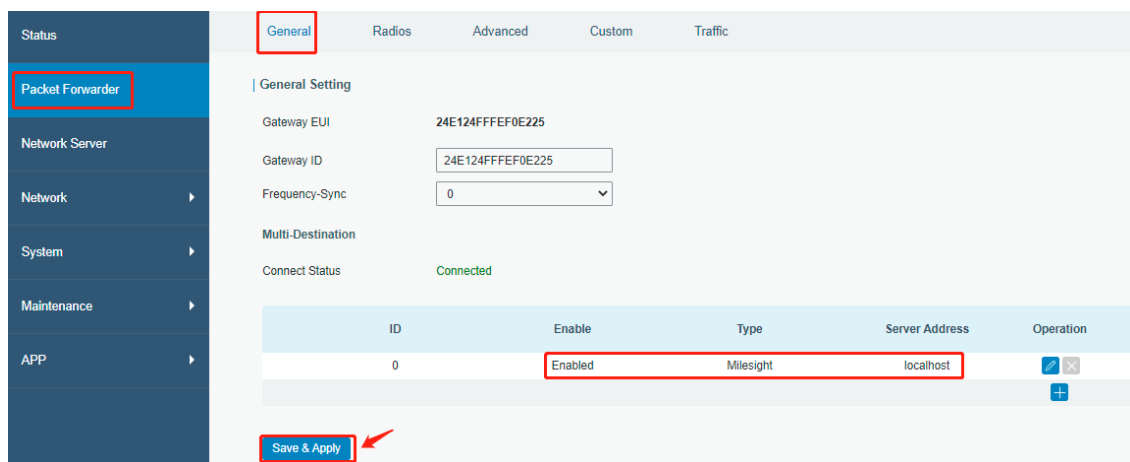


Рисунок 3.24 Packet Forwarder

Увімкніть мережевий сервер на сторінці Network Server > General.

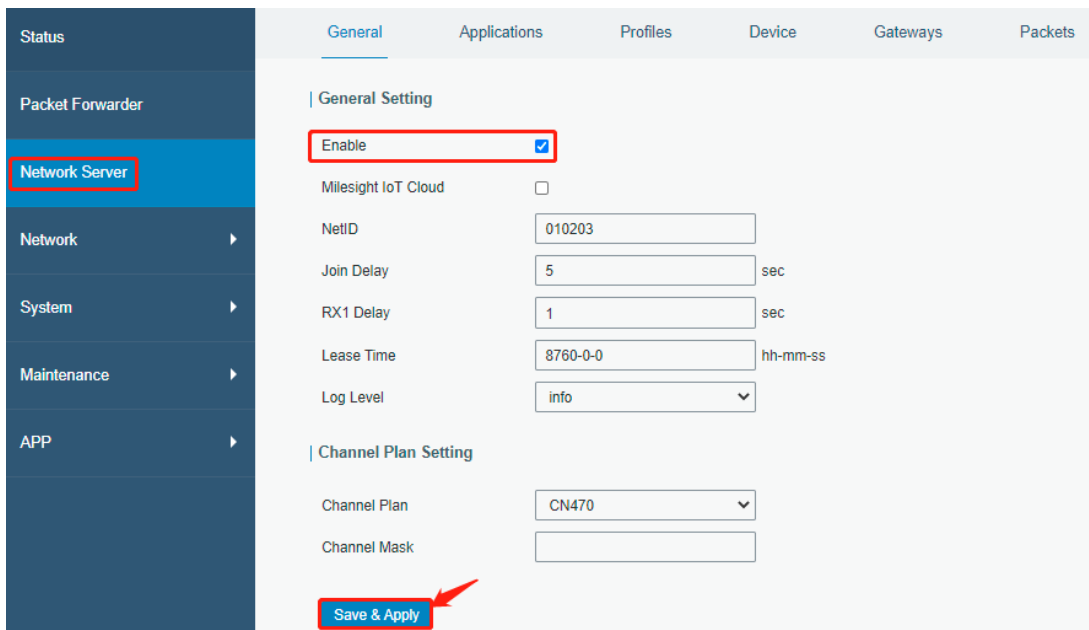


Рисунок 3.25 Network Server

Додаймо програму та профілі.

Перейдіть у Network Server>Applications, додайте нову програму, натисніть зберегти.

Name: cloud

Description: cloud

Payload Codec: None or custom your decoder

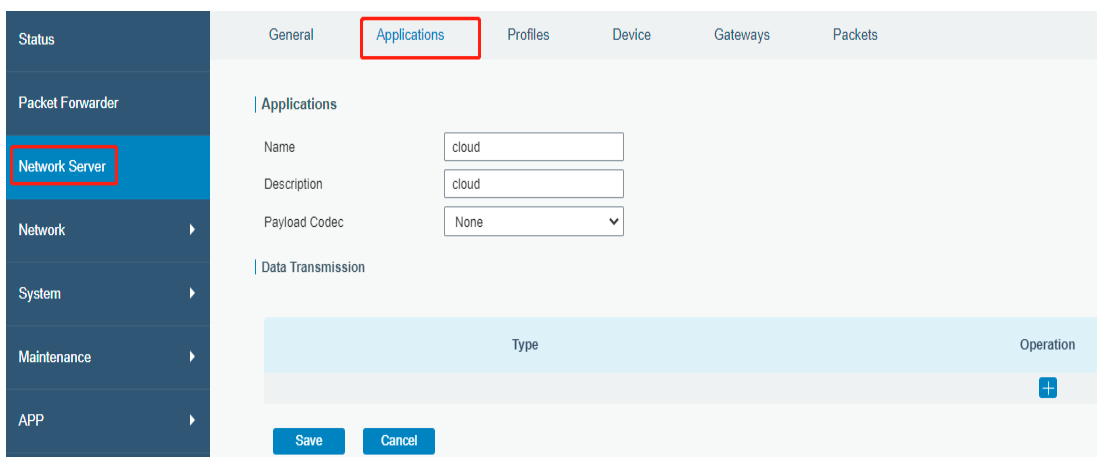


Рисунок 3.26 Applications

Переходимо до Network Server>Profiles, щоб додати новий профіль, потім натисніть «Зберегти».

Name: визначене користувачем, довільне значення

Макс. TXPower: значення за замовчуванням

Інші параметри можна перевірити в посібнику користувача вузлів LoRaWAN або залишити всі налаштування за замовчуванням.

The screenshot shows the 'Profiles' configuration page in the Network Server interface. The 'Profiles' tab is selected and highlighted with a red box. The 'Device Profiles' section contains several input fields: Name (empty), Max TXPower (0), Join Type (OTAA), Class Type (Class A), Advanced (checked), MAC Version (1.0.2), and Regional Parameters Revision (B). There are 'Save' and 'Cancel' buttons at the bottom.

Рисунок 3.27 Profiles

Додаємо вузли LoRaWAN до шлюзу.

Перейдіть у Network Server >Device, додаємо новий девайс

Device Name: визначене користувачем, довільне значення

Description: визначене користувачем, довільне значення

Device-Profile: виберіть один із відповідних профілів, доданих раніше.

Application: виберіть одну з відповідних програм, доданих раніше.

Інші параметри можна підтвердити у виробників вузлів LoRaWAN.

Device Name: lora-sensor

Description: a short description of your node

Device EUI: 0000000000000000

Device-Profile: ClassA-OTAA

Application: cloud

Frame-counter Validation:

Application Key:

Device Address:

Network Session Key:

Application Session Key:

Uplink Frame-counter: 0

Downlink Frame-counter: 0

Save & Apply

Рисунок 3.28 Додаємо вузли LoRaWAN до шлюзу.

Коли статус відображається, як показано нижче, це означає, що вищезазначені кроки виконано правильно.

General Applications Profiles **Device** Gateways Packets

Device

Add Bulk Import Delete All

Search

Device Name	Device EUI	Device-Profile	Application	Last Seen	Activated	Operation
cloud1	24E124127A270222	cloud1	cloud	7 minutes ago	✓	⚙️ ✖️

3.29 Network Server Device

Переселаємо дані брокеру MQTT.

Перейдіть до Network Server>Applications, щоб додати data transmission для програми.

Рисунок 3.30 Applications

Заповнюємо інформацію брокера MQTT і створіємо тему для зберігання різних типів даних

Broker Address: IP address/domain of MQTT server

Broker Port: порт брокера сервера MQTT, порт за замовчуванням 1883

Client ID: унікальний ідентифікатор клієнта для сервера.

Data Type	Topic	QoS 0
Uplink data	imlesightuplink	▼
Downlink data	imlesightdownlink3deveu	▼
Join notification	imlesightjon	▼
ACK notification	imlesightack	▼
Error notification	imlesighterror	▼

Рисунок 3.31 Network Server Applications

Після завершення конфігурації MQTT ви можемо перевірити стан підключення тут:

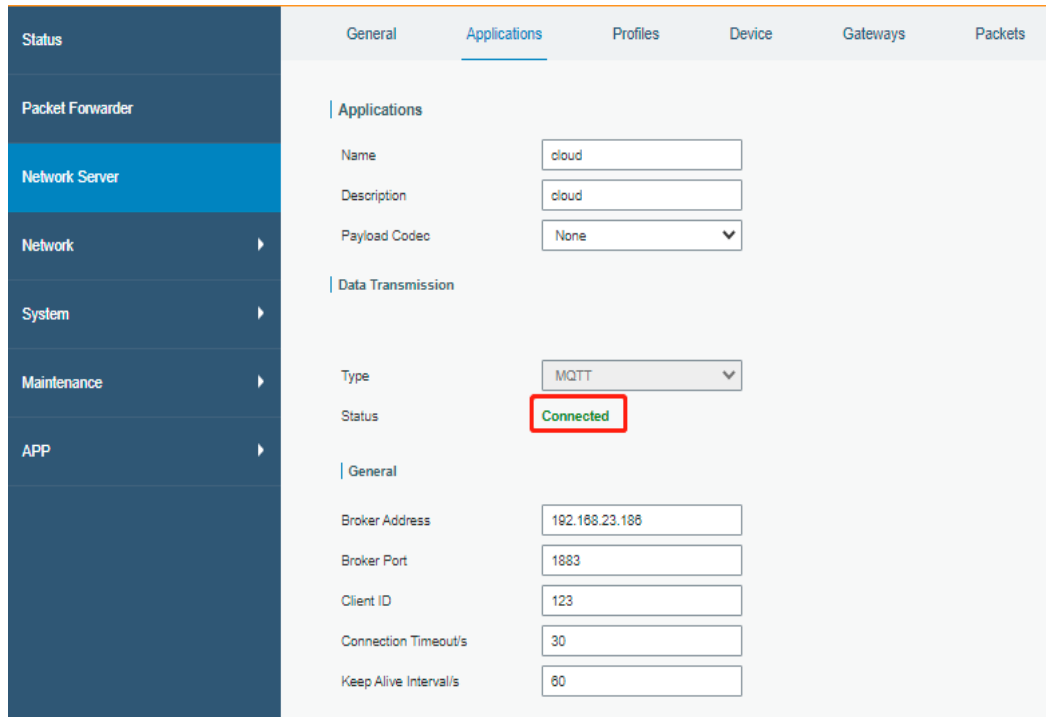


Рисунок 3.32 Статус

Підписуємось на відповідну тему від клієнта MQTT.

MQTT explorer — це комплексний клієнт MQTT, який можна замінити на інші види клієнтських інструментів MQTT (MQTT.fx, MQTT Box тощо)

Відкриваємо MQTT Explorer і заповнюємо інформацію про відповідний сервер MQTT у спливаючому вікні.

Name: user-defined

Protocol: mqtt://

Host: MQTT broker address

Port: broker port

User name/Password: якщо є облікові дані користувача, заповнюємо їх. Якщо ні, залишаємо їх пустими.

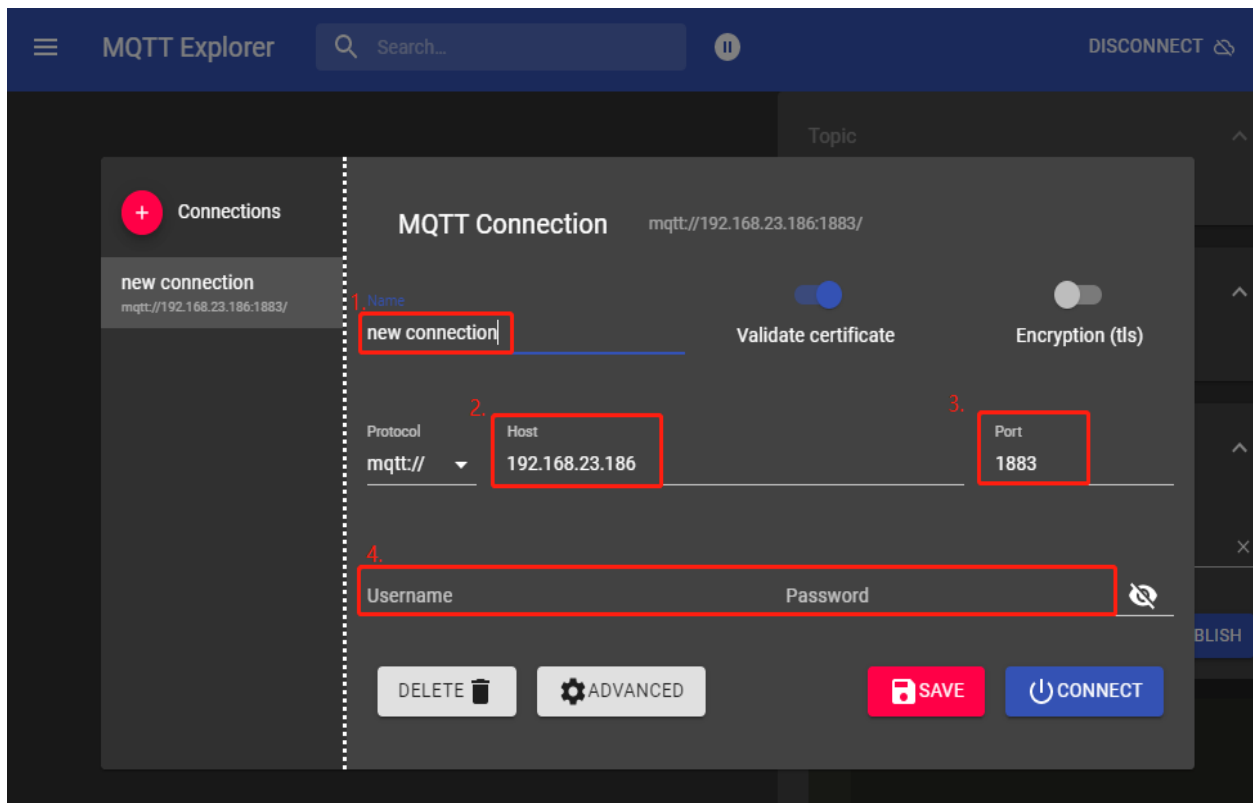


Рисунок 3.33 MQTT Explorer

Натисніть **ADVANCED**, скопіюйте тему даних Uplink на шлюз і вставте її в MQTT explorer, натисніть + **ADD**.

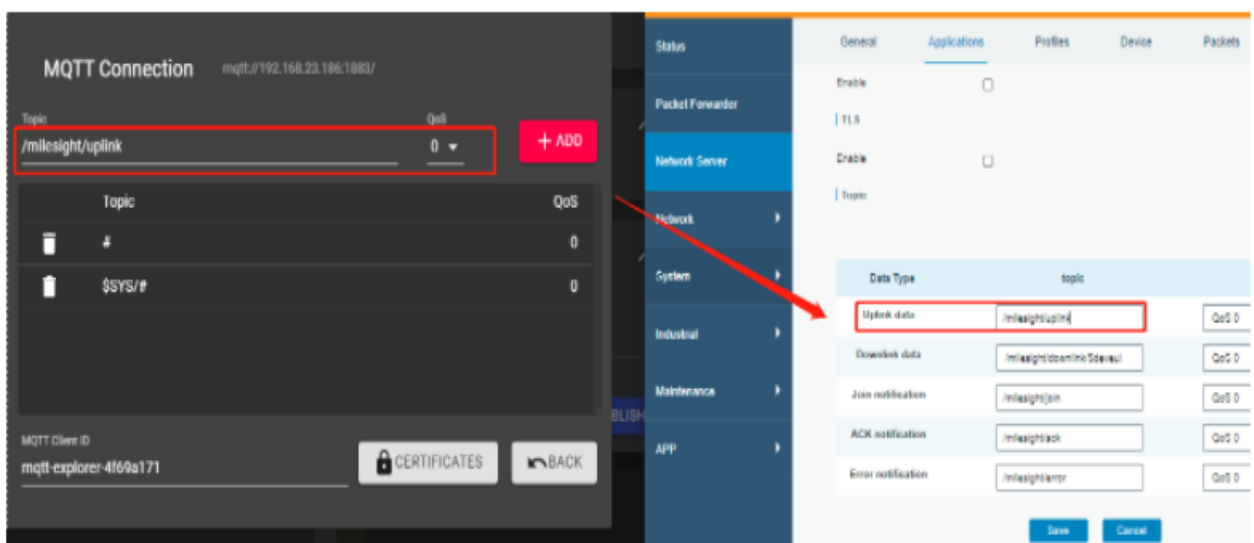


Рисунок 3.34 MQTT Explorer

Залишаємо ідентифікатор клієнта MQTT за замовчуванням і підключаємо

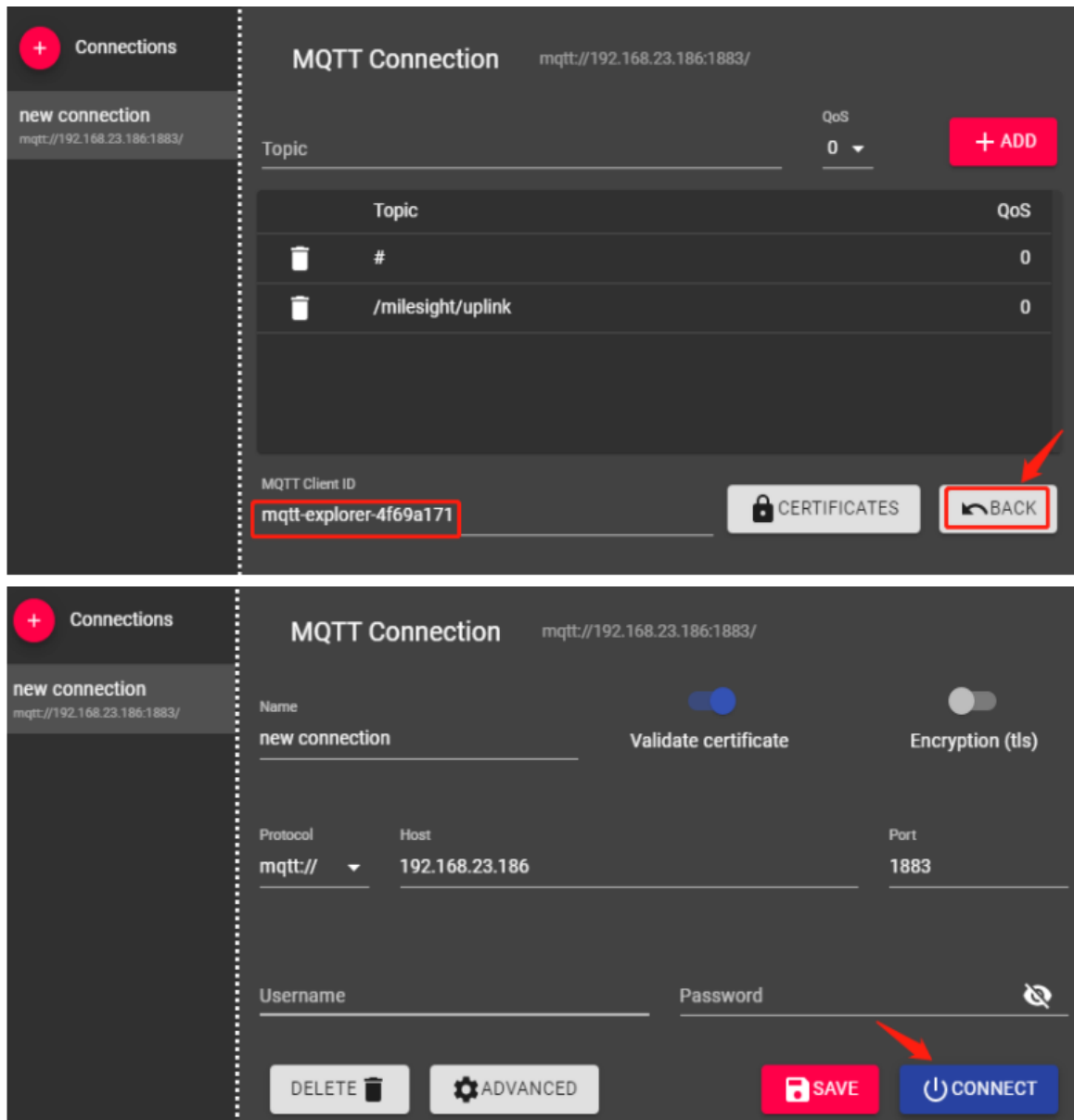


Рисунок 3.35 ідентифікатор клієнта MQTT

Через деякий час отримуємо дані від сервера MQTT.

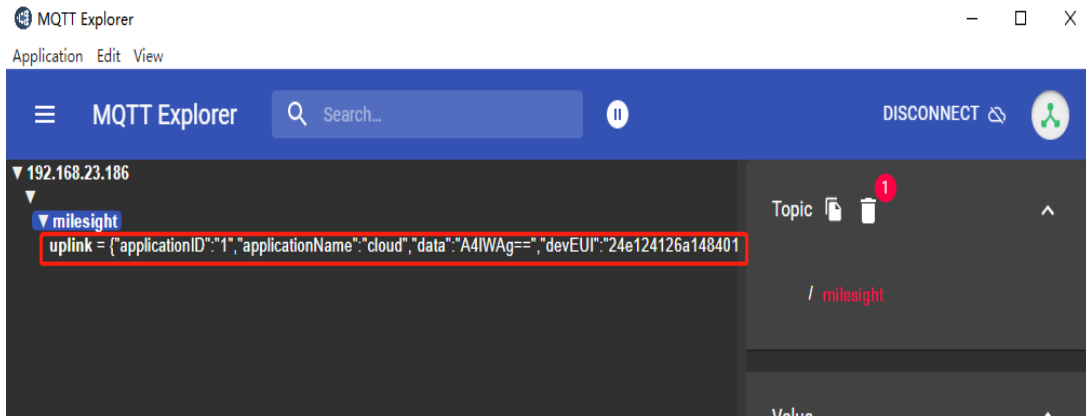


Рисунок 3.36 MQTT Explorer

Публікуємо тему в MQTT Explorer, щоб надіслати дані низхідного каналу на датчик.

Data Type	topic	QoS
Uplink data	<input type="text" value="/milesight/uplink"/>	QoS 0
Downlink data	<input type="text" value=""/>	QoS 0
Join notification	<input type="text" value="/milesight/join"/>	QoS 0
ACK notification	<input type="text" value="/milesight/ack"/>	QoS 0
Error notification	<input type="text" value="/milesight/error"/>	QoS 0

Рисунок 3.37 тема в MQTT Explorer

Зі шлюзу ми можемо отримати ідентифікатор програми та EUI пристрою для пристрою, яким ми хочемо керувати:

The image shows two screenshots of a web application interface. The top screenshot is the 'Applications' tab, displaying a table with columns: ID, Name, Description, Payload Codec, and Operation. The ID '1' is highlighted with a red box. The bottom screenshot is the 'Device' tab, displaying a table with columns: Device Name, Device EUI, Device-Profile, Application, Last Seen, Activated, and Operation. The Device EUI '24e124126a148401' is highlighted with a red box.

ID	Name	Description	Payload Codec	Operation
1	cloud	cloud	None	

Device Name	Device EUI	Device-Profile	Application	Last Seen	Activated	Operation
nancy_test	24e124126a148401	ClassA-OTAA	cloud	53 seconds ago	✓	

Рисунок 3.38 Applications

Тема: application/1/device/ 24e124126a148401/tx

Формат: json

Вміст: перевірте в документі про корисне навантаження датчика (отримайте від виробників датчиків)

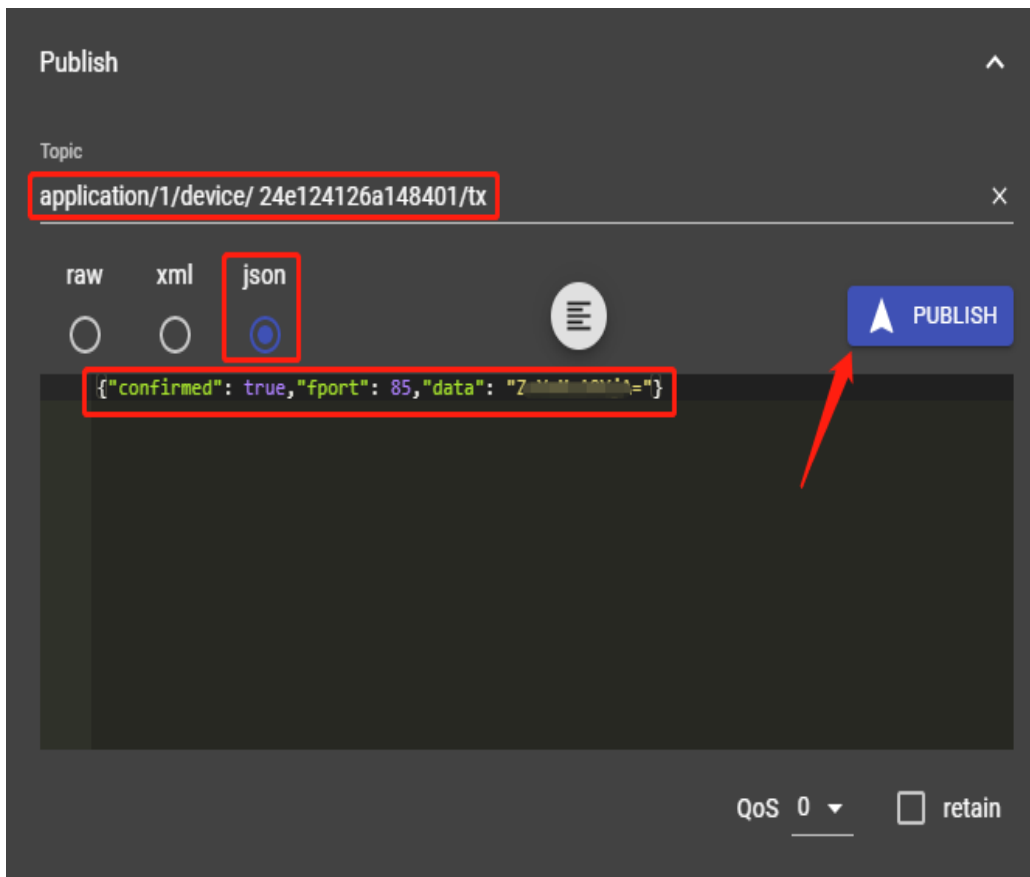


Рисунок 3.39 Applications

Після натискання Опублікувати ми можемо перейти до Network Server>Packets, щоб перевірити. Якщо шлюз успішно підписався на відповідні дані теми низхідного каналу, буде принаймні один запис пакету повідомлень сірого кольору.

General Applications Profiles Device **Packets**

Send Data To Device

Device EUI: 0000000000000000 Type: ASCII Payload: Port: Confirmed:

Send

Network Server

Clear

Device EUI	Frequency	Datarate	SNR	RSSI	Size	Font	Type	Time	Details
24e124120a148401	0				8	222	DnCnt		i
24e124120a148401	868500000	SF7BW125	-	-	0	221	DnUns	2021-03-24T20:16:35+08:00	i
24e124120a148401	868500000	SF7BW125	8.2	-41	4	219	ACK	2021-03-24T20:16:34+08:00	i
24e124120a148401	868500000	SF7BW125	8.2	-41	4	219	UpCnt	2021-03-24T20:16:34+08:00	i
24e124120a148401	868100000	SF7BW125	-	-	8	220	DnCnt	2021-03-24T20:15:35+08:00	i
24e124120a148401	868100000	SF7BW125	9.8	-35	4	218	UpCnt	2021-03-24T20:15:34+08:00	i
24e124120a148401	0				8	220	DnCnt		i
24e124120a148401	868500000	SF7BW125	-	-	0	219	DnUns	2021-03-24T20:14:35+08:00	i
24e124120a148401	868500000	SF7BW125	8.2	-32	4	217	UpCnt	2021-03-24T20:14:34+08:00	i
24e124120a148401	868300000	SF7BW125	-	-	0	218	DnUns	2021-03-24T20:13:35+08:00	i

Рисунок 3.40 список пакетів

Висновки

В рамках даної випускної кваліфікаційної проведеної детальний аналіз концепції, функцій та головних задач систем розумного міста: збирання, передача та зберігання даних, автоматизація процесів життєдіяльності, покращення рівня життя, збереження навколишнього середовища. Був проведений ретельний аналіз сфер застосування концепцій «Розумного міста», технологій для їх реалізацій.

На основі проведеного аналізу була визначена необхідна технологія та програмне забезпечення для її реалізації openHAB та LoRaWAN. Після визначення необхідних елементів була проведена реалізація.

Запропонований у даній роботі спосіб побудови оптимізації дозволяє для додаткової зручності чи функціоналу системи самостійно змінювати чи додавати окремі елементи.

Практична значущість мого дослідження полягає у значній економії, широкому застосуванні, надійності та зручності даної системи

ПЕРЕЛІК ПОСИЛАНЬ

1. [http://ird.gov.ua/sep/sep20184\(132\)/sep20184\(132\)_114_PinA.pdf](http://ird.gov.ua/sep/sep20184(132)/sep20184(132)_114_PinA.pdf)
2. <http://thefuture.news/smart-city/>
3. <https://www.telekom.sk/biznis/nb-iot>
4. https://www.gsma.com/iot/wp-content/uploads/2018/04/LTE-M_Deployment_Guide_v2_5Apr2018.pdf
5. <https://itechinfo.ru/content/%D0%BE%D0%B1%D0%B7%D0%BE%D1%80-%D1%82%D0%B5%D1%85%D0%BD%D0%BE%D0%BB%D0%BE%D0%B3%D0%B8%D0%B8-lora>
6. <http://lo-ra.ru/lorawan-networks/>
7. https://www.slovanet.net/files/biznis/produktovelisty/produktovy_list_lora_link.pdf
8. <https://www.openhab.org/>
9. <https://learntutorials.net/ru/mqtt/topic/9130/%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0-%D0%B8-%D0%BD%D0%B0%D1%81%D1%82%D1%80%D0%BE%D0%B9%D0%BA%D0%B0>

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (Презентація)

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Спеціальність: 126 Інформаційні системи та
технології

Розробка системи розумного міста з
використанням технології «LoRaWAN»

Студент групи ІСДМ-61
Асман Максим Ярославович
Керівник Ткаленко О.М.

2021

Мета роботи – розробка системи розумного міста

Об`єкт дослідження – процес створення системи
автоматизації.

Предмет дослідження – інтелектуальні системи
автоматизації всіх процесів життєдіяльності
міста та зниження витрат

Завдання:

- 1) Аналіз концепції розумного міста;
- 2) Дослідження технології LoRa;
- 2) Вибір програмного забезпечення;
- 4) Реалізація власної системи.

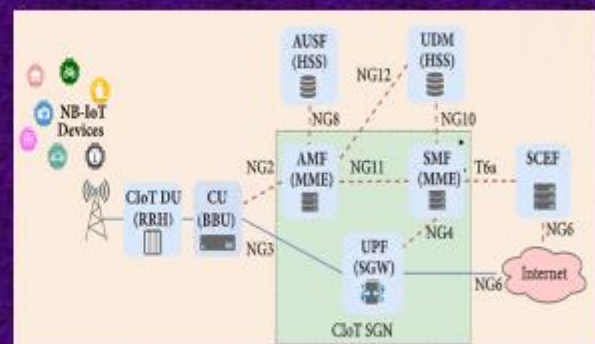
Технології IoT



Технології на базі IoT

- NarrowBand-Internet of Things (NB-IoT):

це заснована на стандартах технологій широкої зони з низьким енергоспоживанням (LPWA), розроблена для створення широкого спектру нових пристроїв і послуг IoT. NB-IoT значно покращує енергоспоживання користувальницьких пристроїв, ємність системи та ефективність використання спектру, особливо в умовах глибокого покриття. Термін служби батареї більше 10 років може підтримуватися в широкому діапазоні випадків використання.

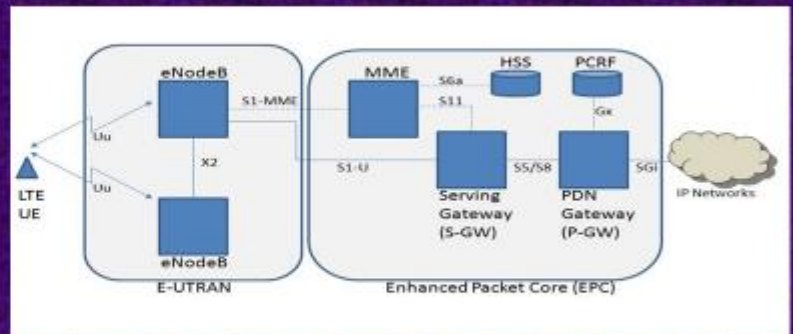


Технології на базі IoT

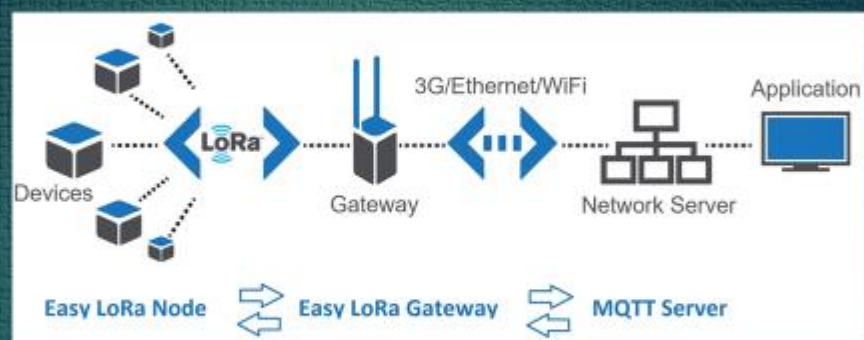
• LTE-M

гілка технології LTE, яка була створена для зв'язку «машина-машина». Використовуючи LTE-M, пристрої IoT можуть передавати та отримувати великі обсяги даних без розрядження батареї, як у мережах 2G, 3G та 4G. Основні причини, чому організації IoT покладаються на LTE-M:

Споживання енергії, Покриття, Швидкість передачі даних, Мобільність



технологія LoRaWAN



Основні переваги технології LoRaWAN

- Ємність мережі

Для того, щоб зробити зіркову мережу далекого радіусу дії життєздатною, шлюз повинен мати дуже високий рівень ємності або можливість приймати повідомлення з дуже великого обсягу вузлів. Високий ємність мережі в мережі LoRaWAN досягається за рахунок використання адаптивних даних швидкість і за допомогою багатоканального багатомодемного трансивера в шлюзі, так що можна одночасно отримувати повідомлення на декількох каналах. Критичні фактори ефективною ємністю є кількість одночасних каналів, швидкість передачі даних довжину корисного навантаження та як часто передають вузли.

Основні переваги технології LoRaWAN

- Стійкість до радіозавад:

Стійкість до радіозавад. Мережа має високу проникність здатність радіосигналу. Це забезпечить стабільне з'єднання з іншими технологіями технології не впораються. Модеми LoRa можуть пригнічувати перешкоди до 19,5 дБ (фільтрація Гаусса). Така здатність придушувати перешкоди дозволить використовувати систему в сучасному великому місті

Необхідність систем автоматизації.



- ▶ збирання, передача та зберігання даних від пристроїв або громадян;
- ▶ налагодження взаємозв'язку між громадянами та керівництвом міста
- ▶ автоматизація всіх процесів життєдіяльності міста та зниження витрат
- ▶ покращення рівня життя мешканців та збереження навколишнього середовища.

Сфери застосування концепції «Розумного міста»

- ▶ Безпека
- ▶ Охорона здоров'я.
- ▶ Транспорт і мобільність пересування
- ▶ Енергозбереження



- ▶ Екологія
- ▶ Залученість громад, економічний розвиток та рівень життя.
- ▶ Водні ресурси

Вибір програмного забезпечення

► openHAB:

один з найбільших проєктів у спільноті розробників програмного забезпечення з відкритим вихідним кодом. Проєкт, який перебуває у постійній розробці і який вже підтримує понад 1500 пристроїв. Це безкоштовне програмне забезпечення можете завантажити його в будь-якій операційній системі, не покладаючись на хмару.

► ioBroker:

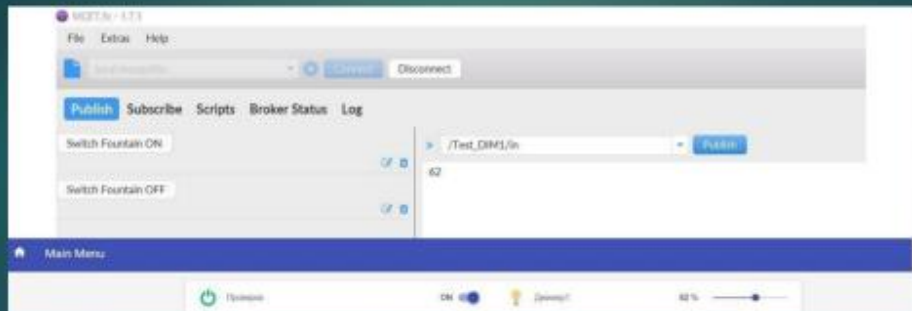
це система автоматизації, розроблена на платформі Node.js. Завдяки цьому її можна розвернути і запустити практично на будь-якій операційній системі або віртуальній машині: Windows, Mac, Linux, в контейнері Docker і навіть на Android. Це дозволяє масштабувати систему виходячи з потреб: поставити її на малину для дому, квест-кімнати або сервер для готелю.

Система ioBroker, як і більшість систем автоматизації, відкрита та розширювана. Новий функціонал або пристрої підключаються за допомогою додавання драйвера або адаптера

Вибір програмного забезпечення



Налаштування та підключення оренНAB до MQTT брокера

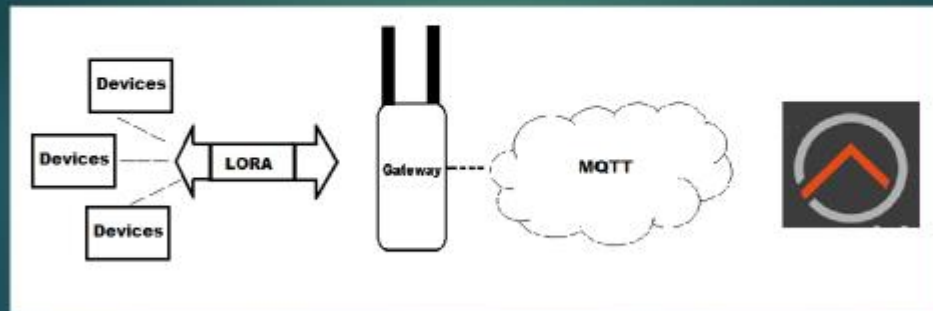


Підключення шлюзу LoRaWAN до MQTT

The screenshot shows the MQTT.fx 4.7.3 web interface with the 'Devices' tab selected. The 'Devices' section is active, and a table of devices is displayed. The table has columns for Device ID, Frequency, Edition, S/N, B.S., Dev, Type, and Date. Two rows are highlighted with red boxes, indicating the configuration for the LoRaWAN gateway.

Device ID	Frequency	Edition	S/N	B.S.	Dev	Type	Date
28012428044901	0				0	001	2014
28012428044901	86800000	07/04/02			0	001	2014
28012428044901	86800000	07/04/02	0.1	41	0	001	2014
28012428044901	86800000	07/04/02	0.2	41	0	001	2014
28012428044901	86800000	07/04/02			0	001	2014
28012428044901	86800000	07/04/02	0.0	40	0	001	2014
28012428044901	0				0	001	2014
28012428044901	86800000	07/04/02			0	001	2014
28012428044901	86800000	07/04/02	0.1	40	0	001	2014
28012428044901	86800000	07/04/02			0	001	2014

Реалізована система



Висновки

- Був проведений детальний аналіз концепції, функцій та головних задач систем розумного міста: збирання, передача та зберігання даних, автоматизація процесів життєдіяльності, покращення рівня життя, збереження навколишнього середовища
- Була визначена необхідна технологія та програмне забезпечення для реалізації – LoRaWAN, OpenHAB
- Після визначення необхідних елементів була проведена реалізація.

АПРОБАЦІЯ

Опубліковано тезу. XIII Міжнародна науково-технічна конференція студентства та молоді «СВІТ ІНФОРМАЦІЇ ТА ТЕЛЕКОМУНІКАЦІЙ». Ст. 126

Ткаленко О.М., Полоневич О.В., Асман М.Я.
Застосування алгоритму з навчанням для виявлення детермінованого сигналу в шумі невідомої інтенсивності, Зв'язок, 2022.



*Дякую за вашу
увагу!*