ВСТУП

Тема «Програмна система для контролю водопостачання у квартирі» є дуже актуальною для курсової роботи. У сучасних умовах швидкого розвитку технологій автоматизація побутових процесів набуває все більшого значення. Однією з актуальних проблем є ефективний контроль водопостачання в житлових приміщеннях, що може сприяти зниженню витрат води, зменшенню витрат на комунальні послуги та своєчасному виявленню можливих несправностей у системі водопостачання. Впровадження IoT технологій у дану сферу відкриває нові можливості для забезпечення більш точного моніторингу та автоматичного контролю за споживанням води.

Актуальність роботи визначається необхідністю створення надійної та ефективної програмної системи, яка б дозволила користувачам отримувати інформацію про стан водопостачання в режимі реального часу, вчасно реагувати на аномальні ситуації та оптимізувати використання води. Це, у свою чергу, сприяє підвищенню якості життя та зменшенню негативного впливу на навколишнє середовище.

Мета і завдання курсової роботи.Метою курсової роботи є розробити програмну систему для автоматизованого контролю та моніторингу водопостачання у квартирі, яка використовує IoT пристрої для збору даних, серверну частину для їх обробки та опціонально веб-інтерфейс для відображення інформації. Для досягнення поставленої мети необхідно вирішити такі завдання:

1. Провести аналіз існуючих систем контролю водопостачання та підходів до розробки IoT систем.
2. Розробити серверну частину програмної системи, яка забезпечить прийом, зберігання та обробку даних з IoT пристроїв.
3. Створити програмну частину IoT пристрою для збору даних про споживання води та передачі їх на сервер.
4. За необхідності, розробити веб-інтерфейс для відображення зібраних даних та управління системою.
5. Провести тестування розробленої системи на предмет її функціональності та надійності.
6. Надати рекомендації щодо впровадження та подальшого вдосконалення системи.
7. Написання Vision & Scope документа.

Курсова робота складається з чотирьох основних розділів, списку використаних джерел та 3-х додатків.

В першому розділі проводиться аналіз предметної області, що включає огляд існуючих технологій та методів, які будуть використані для розробки програмної системи контролю водопостачання у квартирі за допомогою IoT технологій. Детально розглядається постановка задачі, зокрема визначаються бізнес-вимоги, основні цілі проекту та критерії успіху. Також визначаються основні вимоги до програмної системи, її функціональність, припущення та залежності. Обговорюються можливі варіанти реалізації системи, їхні переваги та недоліки, а також обмеження та ризики, пов'язані з кожним з варіантів. Особлива увага приділяється концепції рішення, що включає детальний опис архітектури системи, основних компонентів та їх взаємодії.

В другому розділі розглядається постановка задачі та вимоги до системи. Описується, яка конкретно задача має бути вирішена за допомогою розробленої системи контролю водопостачання в квартирі. Визначаються основні функціональні вимоги до системи, такі як моніторинг витрати води, виявлення витоків, віддалене управління та сповіщення користувача. Розглядаються нефункціональні вимоги, включаючи ефективність системи, її масштабованість, безпеку даних і зручність користувацького інтерфейсу. Очікувані результати роботи описують, яким чином система має вплинути на управління водопостачанням у квартирі, зокрема, як вона покращить моніторинг і контроль витрат води, а також як вона може запобігти можливим витокам і аваріям. Визначаються метрики для оцінки успішності реалізації проекту та критерії для перевірки виконання вимог.

В третьому розділі розглядається проектування програмного проекту, яке включає три основні компоненти системи. Описується архітектура серверної частини на базі Node.js, включаючи розробку API для взаємодії з IoT-пристроями і клієнтським інтерфейсом, механізми аутентифікації та авторизації, а також взаємодію з базою даних MongoDB. Детально описується проектування IoT-пристрою, включаючи вибір сенсорів, розробку прошивки на C++ для зчитування і передачі даних, вибір протоколів зв'язку, а також перевірку точності сенсорів. Також, розглядається проектування клієнтського інтерфейсу, що включає розробку веб-додатку або мобільного додатку для перегляду даних, отримання сповіщень і управління системою, інтеграцію з сервером через API, а також можливості налаштування сповіщень і управлінських функцій.

В четвертому розділі зосередженося на структурі бази даних, що включає три основні аспекти. Описується побудова ER-діаграми, яка відображає структуру бази даних, включаючи сутності, атрибути та їх взаємозв'язки. Також, містить детальний опис сутностей і атрибутів бази даних, а також типів даних, які будуть використовуватися для зберігання інформації. Зокрема, розглядаються взаємозв'язки між сутностями, включаючи типи зв'язків (один до одного, один до багатьох, багато до багатьох) і способи їх реалізації для забезпечення цілісності та ефективності даних.

Опис використаних засобів розробкидля виконання курсової роботи включає вибір мови програмування, платформ і інструментів, що забезпечують реалізацію проекту. Для розробки серверної частини проекту була обрана мова програмування JavaScript з використанням платформи Node.js. Це дозволяє ефективно створювати серверні додатки, обробляти запити, реалізувати API і взаємодіяти з базою даних MongoDB. Для програмування IoT-пристрою використана мова C++, що забезпечує точне і швидке оброблення даних сенсорів і їх передачу на сервер. Для зберігання та обробки даних обрана база даних MongoDB, яка забезпечує гнучкість та масштабованість зберігання даних. У розробці клієнтської частини проекту використані сучасні веб-технології, такі як HTML, CSS, JavaScript, що забезпечує створення інтуїтивно зрозумілого інтерфейсу користувача для перегляду та управління даними. Інтеграція всіх компонентів забезпечує ефективну і зручну систему контролю водопостачання в квартирі.

Практична значимість виконаної роботи полягає вефективності використання розробленої системи для реального контролю водопостачання в квартирах, що сприяє зменшенню витрат води і виявленню проблем на ранніх стадіях.

# 1. **АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ**

## Бізнес-вимоги

### 1.1.1 Бізнес можливості

1. Система моніторингу водопостачання на базі IoT дозволяє точно відслідковувати споживання води в реальному часі, що може допомогти зменшити витрати на водопостачання. Виявлення аномальних витрат або витоків води допомагає швидше реагувати на проблеми, що дозволяє зменшити витрати на ремонти та комунальні послуги. Моніторинг водопостачання забезпечує виявлення проблем, таких як протікання або порушення в системі, до того, як вони призведуть до серйозних пошкоджень. Це дозволяє виконувати превентивне обслуговування, запобігаючи дорогим аваріям і ремонту, а також зменшує витрати на ремонтні роботи.
2. Завдяки точному моніторингу споживання води, користувачі можуть оптимізувати своє використання води. Система може надати детальні звіти та аналіз, що дозволяє виявити тренди і навести практики для раціонального використання води, що важливо для збереження природних ресурсів та зменшення витрат. Інтеграція з іншими IoT пристроями, такими як системи автоматизації будинку або датчики пожежі, створює можливості для розширення функціоналу системи. Це дозволяє створювати розширені рішення для розумного дому, де всі пристрої взаємодіють між собою для забезпечення максимальної зручності та безпеки. Забезпечення унікальних функцій та інтеграцій, які не пропонують конкуренти, може стати суттєвою конкурентною перевагою. Висока точність вимірювань, надійність роботи системи та простота інтеграції з іншими технологіями можуть залучити більше клієнтів та зміцнити позиції на ринку.
3. Система допомагає зменшити витрати води, що може бути важливим фактором для споживачів, які турбуються про навколишнє середовище. Підвищена ефективність використання води підтримує принципи сталого розвитку та допомагає зберігати ресурси. Система може слугувати основою для подальших інновацій у сфері управління водопостачанням, включаючи використання штучного інтелекту для прогнозування витрат, інтеграцію з новими типами датчиків та розширення функціоналу системи для забезпечення додаткових можливостей. Ці бізнес можливості створюють значний потенціал для розвитку та комерціалізації системи моніторингу водопостачання, що може принести користь як споживачам, так і розробникам, забезпечуючи їм можливості для розширення на ринку та зростання бізнесу.

### 1.1.2 Бізнес-цілі та критерії успіху

1. Основною бізнес-ціллю є створення надійної та доступної системи моніторингу водопостачання для квартир, яка буде проста у використанні та налаштуванні. Критеріями успіху є задоволення клієнтів, зменшення випадків протікання та перевитрат води, а також можливість легкої інтеграції з іншими IoT пристроями та системами.

### 1.1.3 Потреби клієнтів або ринку

1. Клієнти потребують рішення, яке дозволить їм в реальному часі відслідковувати витрати води, отримувати сповіщення про непередбачені ситуації, та аналізувати дані про використання води для оптимізації витрат. Ринок вимагає доступних, надійних та легко масштабованих IoT рішень, які можуть інтегруватися з іншими смарт-пристроями в межах "розумного будинку".

### 1.1.4 Бізнес-ризики

Основні ризики пов'язані з потенційною несправністю обладнання, що може призвести до неправильної обробки даних або втрати важливої інформації. Інші ризики включають недостатню захищеність системи від кібератак, що може призвести до компрометації даних користувачів. Важливо також враховувати ризики, пов'язані з надійністю підключення до інтернету, оскільки система залежить від передачі даних через мережу.

## Концепція рішення

### 1.2.1 Окреслення концепції

1. Концепція рішення полягає в розробці інтегрованої системи моніторингу водопостачання для квартир, що використовує IoT технології для збору та передачі даних про витрати води. Система складається з трьох основних компонентів: сенсорного вузла на базі мікроконтролера (наприклад, ESP8266 або ESP32), серверної частини для обробки та зберігання даних (з використанням MongoDB), та опціонального веб-інтерфейсу для візуалізації даних і управління системою. Основна мета – забезпечити користувачам реальний моніторинг та контроль за витратами води, оперативне виявлення аномалій і своєчасне реагування на можливі проблеми.

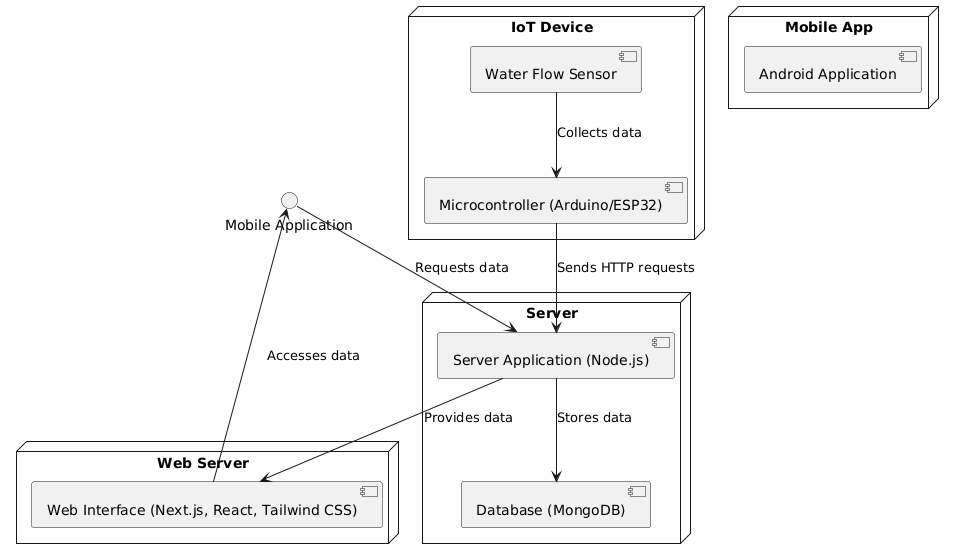
### 1.2.2 Головна функціональність

1. Збір даних забезпечує точність і актуальність інформації про витрати води. Передача даних гарантує, що дані з сенсорів своєчасно потрапляють на сервер. Обробка даних дозволяє зберігати інформацію і виконувати її агрегацію для подальшого аналізу. Аналіз допомагає виявити проблеми до того, як вони стануть критичними. Візуалізація даних забезпечує користувачам доступ до даних у зручному форматі, що сприяє кращому розумінню ситуації. Сповіщення надає можливість швидко реагувати на проблеми та запобігти їхній ескалації.
2. Таблиця 1.1 – Головна функціональність додатку

|  |  |
| --- | --- |
| Функціональність | Опис |
| Збір даних | Сенсорний вузол на основі мікроконтролера зчитує дані з датчика потоку води та визначає обсяг витрат води в реальному часі |
| Передача даних | Дані, зібрані сенсором, відправляються на сервер через HTTP POST запити, забезпечуючи своєчасну передачу інформації |
| Обробка даних | Сервер обробляє отримані дані, зберігає їх у базі даних MongoDB, виконує агрегацію, аналізує дані для виявлення аномалій |

Продовження таблицы Таблиця 1.1

|  |  |
| --- | --- |
| Аналіз | Система виконує аналіз даних для виявлення аномальних витрат води, таких як протікання або різкі зміни в споживанні |
| Візуалізація даних | Веб-інтерфейс (опціонально) надає користувачам можливість переглядати статистику, отримувати сповіщення про аномалії, керувати налаштуваннями системи |
| Сповіщення | Система надсилає сповіщення або попередження користувачам у разі виявлення проблем, таких як високий рівень споживання води або потенційні витоки |



1. Рисунок 1.2 – Алгоритм роботи

### 1.2.3 Припущення та залежності

Припущення визначають умови, при яких система буде функціонувати оптимально. Наприклад, стабільне з'єднання з Wi-Fi необхідне для безперебійної передачі даних. Залежності описують зовнішні компоненти та бібліотеки, від яких залежить функціональність системи. Це включає бібліотеки для роботи з мікроконтролерами, серверними технологіями і веб-браузерами.

1. Таблиця 1.2 – Припущення та залежності

|  |  |  |
| --- | --- | --- |
| 1. Категорія | 1. Припущення | 1. Залежності |
| 1. Припущення | 1. В приміщенні є стабільне з'єднання з Wi-Fi | 1. Віртуальна або реальна мережа Wi-Fi, яка підтримує зв'язок з IoT пристроєм |
| 1. Сервер завжди доступний і здатний обробляти запити | 1. Серверна інфраструктура, що забезпечує безперебійну роботу і обробку даних |
| 1. Датчики та інші компоненти системи працюють коректно | 1. Наявність і функціональність датчиків, які правильно зчитують і передають дані |
| 1. Залежності | 1. Код на мікроконтролері залежить від бібліотек для роботи з Wi-Fi і HTTP запитами. | 1. Бібліотеки для ESP8266/ESP32, які дозволяють виконувати HTTP запити і з'єднання з Wi-Fi |
| 1. Серверна частина залежить від бібліотек для роботи з MongoDB та обробки HTTP запитів | 1. Бібліотеки для Node.js, що забезпечують взаємодію з MongoDB та обробку запитів |

Продовження таблиці 1.2

|  |  |  |
| --- | --- | --- |
|  | 1. Веб-інтерфейс повинен бути сумісним з популярними веб-браузерами (Google Chrome, Mozilla Firefox, Edge) | 1. Браузери, що підтримують сучасні веб-технології, такі як HTML5, CSS3, JavaScript |

## Рамки та обмеження проєкту

### 1.3.1 Рамки первинного випуску

У первинному випуску проєкту буде реалізована основна функціональність системи моніторингу водопостачання. Це включає використання мікроконтролера ESP8266 або ESP32 для збирання даних з датчика потоку води та передачі цих даних на сервер через HTTP POST запити. Серверна частина буде реалізована на базі Node.js і MongoDB, що забезпечить зберігання та обробку отриманих даних. Веб-інтерфейс для перегляду даних і базової візуалізації буде реалізований за допомогою фреймворків Next.js, React та Tailwind CSS. Створення основної документації, включаючи інструкції з налаштування та використання системи, також входить до рамок первинного випуску.

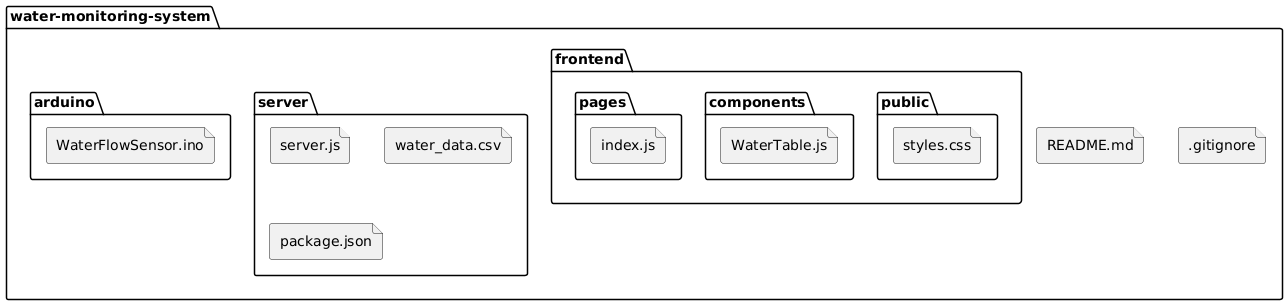


Рисунок 1.1 – Структура проєкту

### 1.3.2 Рамки наступних випусків

1. В наступних випусках проєкту планується розширення функціональності системи. Це може включати додавання нових можливостей, таких як детальний аналіз даних, інтеграція з іншими IoT пристроями та покращення функцій сповіщень. Поліпшення дизайну веб-інтерфейсу також буде частиною наступних випусків, зокрема впровадження нових візуалізацій і функцій для покращення зручності користувачів. Оптимізація продуктивності серверної частини для обробки великої кількості запитів і зменшення затримок буде реалізована в подальших версіях. Крім того, планується розробка мобільного додатка для платформ Android та iOS, що дозволить користувачам отримувати доступ до даних з мобільних пристроїв.

### 1.3.3 Обмеження та винятки

1. Проєкт має кілька обмежень і винятків, які слід врахувати. Технічні обмеження можуть включати неточності вимірювань датчиків і обмежену пропускну здатність серверної частини при високих навантаженнях. Система залежить від стабільного з'єднання з Wi-Fi та надійності серверної інфраструктури, що може вплинути на її роботу в умовах нестабільного з'єднання. Початковий випуск не включає підтримку багатьох типів датчиків і інтеграцію з іншими системами, а також не передбачає функціоналу для масштабованих установок. Перший випуск може мати обмеження в плані забезпечення високого рівня безпеки даних, що потребує подальшого вдосконалення.

## Бізнес-контекст

### 1.4.1 Профілі зацікавлених сторін

Основними користувачами системи будуть власники квартир, які бажають моніторити витрати води у своїх помешканнях. Їхні основні потреби включають точне відстеження споживання води, виявлення потенційних витоків та отримання сповіщень про аномалії. Вони зацікавлені в легкому у використанні інтерфейсі для перегляду даних та отримання інформації про споживання води.

Розробники відповідальні за реалізацію технічних компонентів системи, включаючи мікроконтролери, серверну частину і веб-інтерфейс. Їхні інтереси зосереджені на ефективній реалізації функціональності, інтеграції компонентів та забезпеченні стабільної роботи системи.

Адміністратори, які займаються підтримкою та управлінням серверною частиною системи. Їх завдання включають моніторинг сервера, управління базою даних і забезпечення безперебійної роботи системи. Вони також відповідають за технічне обслуговування і оновлення програмного забезпечення.

Інвестори або бізнес-замовники, які фінансують проєкт або отримують прибуток від його реалізації. Їхні інтереси пов'язані з успіхом проєкту, можливістю повернення інвестицій і забезпеченням економічної вигоди.

Кінцеві користувачі можуть бути орендарями або мешканцями квартир, які користуються системою. Їхні інтереси зосереджені на отриманні корисної інформації про витрати води та зменшенні витрат на воду через вчасне виявлення витоків.

### 1.4.2 Пріоритети проекту

Пріоритети проекту включають: функціональність, юзабіліті, надійність, безпека, масштабованість.

Найвищий пріоритет має забезпечення точності збору даних з датчиків, коректність передачі даних на сервер і надійність веб-інтерфейсу. Система повинна ефективно моніторити витрати води та надавати актуальні дані користувачам.

Інтерфейс користувача має бути інтуїтивно зрозумілим і зручним для користування. Це включає в себе простоту в перегляді даних, зрозумілі сповіщення та доступ до всіх основних функцій.

Система повинна працювати без збоїв і забезпечувати стабільну роботу як на рівні мікроконтролера, так і на сервері. Надійність і стабільність є критичними для підтримки безперервного моніторингу.

Забезпечення захисту даних користувачів та захист системи від можливих кіберзагроз є важливим аспектом. Першочергово важливо впровадити базові заходи безпеки, такі як шифрування даних і захист від несанкціонованого доступу.

Хоча первинний випуск не включає масштабованість, планується забезпечити можливість розширення системи в майбутньому, щоб підтримувати більшу кількість датчиків і користувачів.

### 1.4.3 Робоче середовище

Робоче середовище для реалізації проєкту включає кілька ключових компонентів.

Для розробки програмного забезпечення буде використовуватися IDE Visual Studio Code для Arduino і Node.js, а також Android Studio для мобільного додатку. Ці середовища забезпечують необхідні інструменти для програмування, тестування і відлагодження. Серверна частина проєкту буде розгорнута на хостингу або власному сервері, що підтримує Node.js і MongoDB. Сервер повинен мати надійне з'єднання з Інтернетом і відповідні ресурси для обробки запитів і зберігання даних. Для тестування функціональності IoT пристроїв і серверної частини використовуватиметься середовище Wokwi для емуляції мікроконтролерів. Це дозволяє перевірити код і функціональність без фізичного підключення реальних пристроїв. Веб-інтерфейс буде розроблено на основі фреймворків Next.js, React та Tailwind CSS. Для забезпечення коректної роботи інтерфейсу необхідно тестування на популярних веб-браузерах, таких як Google Chrome, Mozilla Firefox та Microsoft Edge. Мобільний додаток буде розроблений в Android Studio і тестуватися на пристроях з ОС Android. Це середовище забезпечить інструменти для розробки, тестування та відлагодження мобільного додатку.

# 2. ПОСТАНОВКА ЗАДАЧІ

## 2.1 Опис задачі

Основне завдання цього проєкту полягає у розробці системи моніторингу водопостачання в квартирі з використанням IoT технологій. Система має включати мікроконтролер (на основі Arduino ESP8266 або ESP32) для збору даних з датчика потоку води, серверну частину для зберігання та обробки цих даних, а також веб-інтерфейс для візуалізації та моніторингу показників споживання води. Метою є створення системи, яка дозволить користувачам отримувати реальну інформацію про витрати води в режимі реального часу, з можливістю виявлення витоків та аналізу даних.

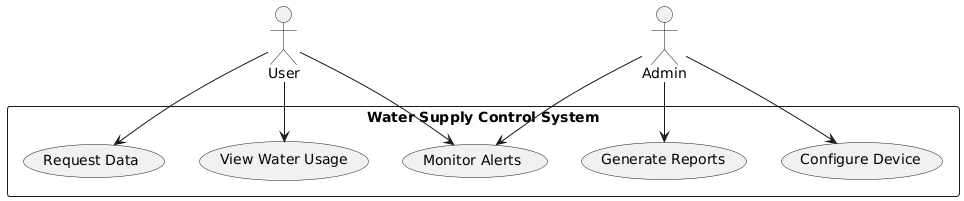
## 2.2 Вимоги до системи

Функціональні вимоги:

* Система повинна збирати дані про витрати води з датчика потоку, підключеного до мікроконтролера;
* Зібрані дані повинні передаватися на сервер через HTTP POST запити;
* Серверна частина повинна зберігати дані у базі даних (бажано MongoDB) та обробляти їх для подальшої візуалізації;
* Веб-інтерфейс повинен забезпечувати користувачів можливістю перегляду поточних і історичних даних про споживання води;
* Система повинна підтримувати відправлення сповіщень у разі виявлення аномалій у споживанні води (наприклад, можливих витоків).

Нефункціональні вимоги:

* Надійність: система повинна забезпечувати безперебійне збирання та передачу даних навіть при короткочасних збоях у з'єднанні;
* Безпека: дані повинні передаватися захищеним каналом для уникнення несанкціонованого доступу;
* Масштабованість: система повинна бути легко адаптована для підтримки більшої кількості користувачів або додаткових датчиків;
* Юзабіліті: веб-інтерфейс має бути інтуїтивно зрозумілим і простим у використанні;
* Продуктивність: система повинна обробляти дані у режимі реального часу, з мінімальними затримками у відображенні даних у веб-інтерфейсі.



1. Рисунок 2.1 – Діаграма прецедентів

## 2.3 Очікувані результати

У результаті виконання цього проєкту очікується створення робочої системи моніторингу водопостачання для квартир. Система повинна забезпечувати:

* Стабільне збирання та збереження даних про витрати води;
* Можливість користувачів відслідковувати показники водопостачання у режимі реального часу через веб-інтерфейс;
* Надання інформації про можливі витоки або аномалії у споживанні води;
* Забезпечення зручного та інтуїтивного інтерфейсу для кінцевих користувачів;
* Можливість подальшого розширення функціональності, включаючи підтримку нових типів датчиків та інтеграцію з іншими IoT системами.

# 3 ПРОЕКТУВАННЯ ПРОГРАМНОГО ПРОЕКТУ

## 3.1 Проектування серверної частини проекту

Проектування серверної частини включає створення інфраструктури для прийому, зберігання та обробки даних, отриманих від IoT-пристрою. Сервер буде розгорнутий на Node.js і використовуватиме MongoDB для зберігання даних.

Основні компоненти серверної частини:

1. REST API

Сервер надає REST API для отримання даних від IoT-пристрою. Основні ендпоїнти включають:

POST /data: Приймає дані від IoT-пристрою та зберігає їх у базі даних.

GET /data: Повертає збережені дані для їх відображення на веб-інтерфейсі.

GET /data/:id: Повертає дані за певним ідентифікатором.

1. База даних

Для зберігання даних про витрати води використовується MongoDB. Кожен запис у базі даних містить наступні поля:

timestamp: Час отримання даних.

flow\_rate: Показник витрати води.

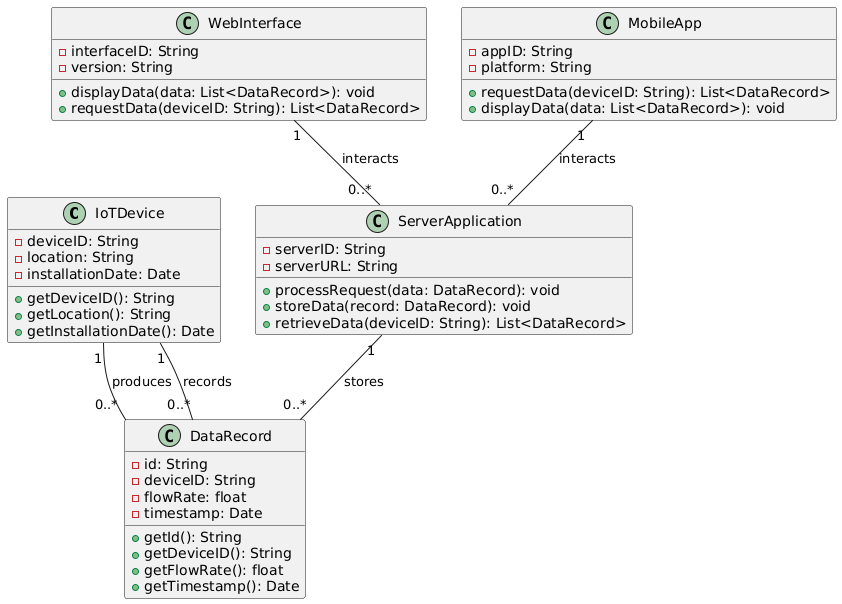
device\_id: Ідентифікатор IoT-пристрою.

1. Обробка даних

Сервер також здійснює первинну обробку даних, включаючи перевірку на наявність аномалій, таких як різке збільшення витрати води, що може свідчити про виток. У разі виявлення аномалій, сервер може відправляти сповіщення користувачам через електронну пошту або інші канали зв'язку.

1. Інтеграція з веб-інтерфейсом

Сервер надає дані для веб-інтерфейсу через API. Це дозволяє користувачам переглядати дані про витрати води, а також налаштовувати сповіщення та інші параметри системи.



1. Рисунок 2.1 – Діаграма класів

## 3.2 Проектування IOT-пристрою

IoT-пристрій відповідальний за збір даних з датчика потоку води та їх передачу на сервер. Основою для IoT-пристрою є мікроконтролер ESP32, який забезпечує зв'язок з датчиком та сервером через Wi-Fi. Основні компоненти IoT-пристрою:

1. Мікроконтролер

Використовується ESP32, який забезпечує всі необхідні функції для збору даних та їх передачі. Ці мікроконтролери мають вбудований Wi-Fi модуль, що дозволяє підключитися до Інтернету і відправляти дані на сервер.

1. Датчик потоку води

Датчик підключається до мікроконтролера і вимірює кількість води, що проходить через трубопровід. Дані передаються на мікроконтролер у вигляді електричних імпульсів, які конвертуються у значення витрати води.

1. Програмна частина

Програма, що працює на мікроконтролері, збирає дані з датчика, обчислює показники витрати води та відправляє їх на сервер через HTTP POST запит. Також можливе додавання функціональності для локального збереження даних у випадку втрати зв'язку з сервером.

1. Живлення
2. Пристрій потребує стабільного джерела живлення, наприклад, через адаптер або акумулятор. Це забезпечить безперервну роботу пристрою і передачу даних у режимі реального часу.

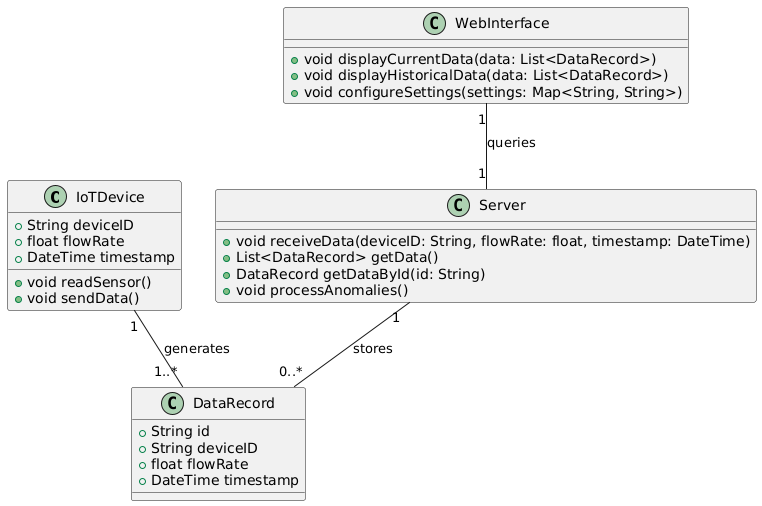


Рисунок 3.1 – UML-діаграма класів описує структуру системи

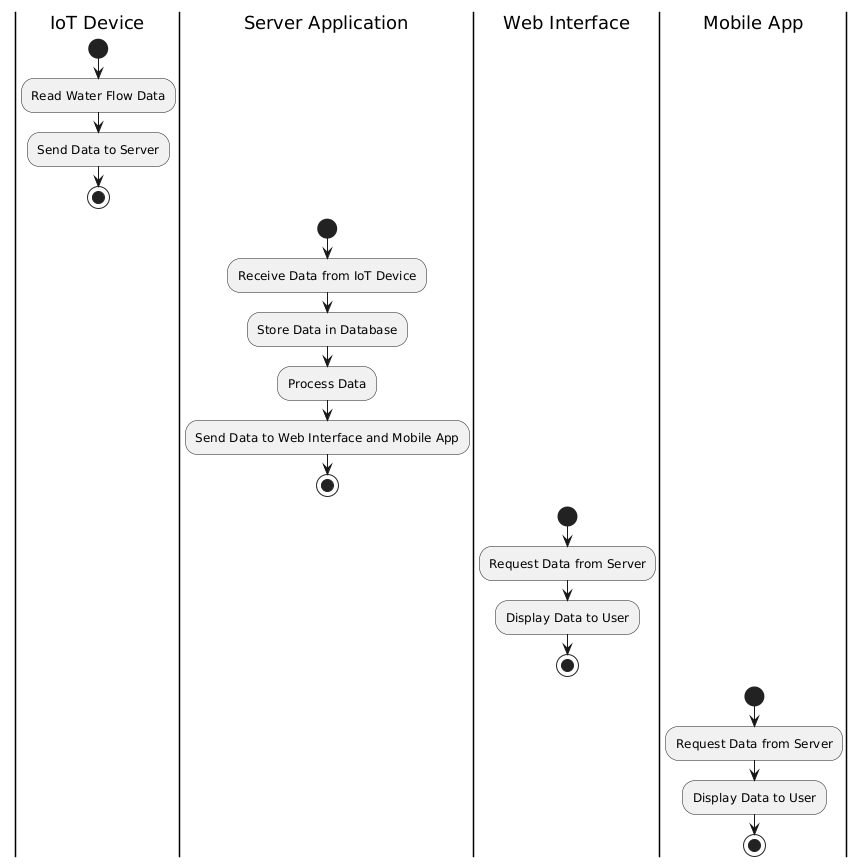


Рисунок 3.2 – UML-діаграма активності (послідовності)

## 3.3 Проектування клієнтської частини проекту

Клієнтська частина проекту включає веб-інтерфейс, який дозволяє користувачам взаємодіяти із системою, переглядати дані про витрати води, налаштовувати сповіщення та інші параметри. Основні компоненти клієнтської частини:

1. Веб-інтерфейс. Веб-інтерфейс розробляється з використанням фреймворку Next.js, React і Tailwind CSS. Він включає такі сторінки:

* Головна сторінка. Відображає основну інформацію про витрати води у вигляді графіків та таблиць;
* Налаштування. Дає можливість налаштувати параметри пристрою, наприклад, межі витрати води для сповіщень;
* Історія. Містить історичні дані про споживання води, які можна переглянути за обраний період.

1. Мобільний додаток (опціонально) у v. 2.0. Для зручності користувачів може бути розроблений мобільний додаток на базі Android, який дозволяє отримувати сповіщення та переглядати дані в реальному часі. Додаток може бути розроблений у Android Studio з використанням Java або Kotlin.
2. Адаптивний дизайн

Веб-інтерфейс має бути адаптивним, щоб користувачі могли зручно працювати з системою як на комп'ютерах, так і на мобільних пристроях. Використання Tailwind CSS дозволяє легко налаштувати інтерфейс під різні розміри екрану.

1. Інтеграція з API

Клієнтська частина активно використовує API серверної частини для отримання актуальних даних. Це включає динамічне завантаження графіків, таблиць і сповіщень на основі отриманих від сервера даних.

1. Тестування та налагодження

Клієнтська частина підлягає ретельному тестуванню для забезпечення стабільної роботи на різних платформах та браузерах. Це включає функціональні та юзабіліті-тести, що гарантують відповідність інтерфейсу потребам користувачів.

Детальне проектування забезпечує інтеграцію всіх компонентів системи для досягнення високої функціональності, надійності та зручності використання.

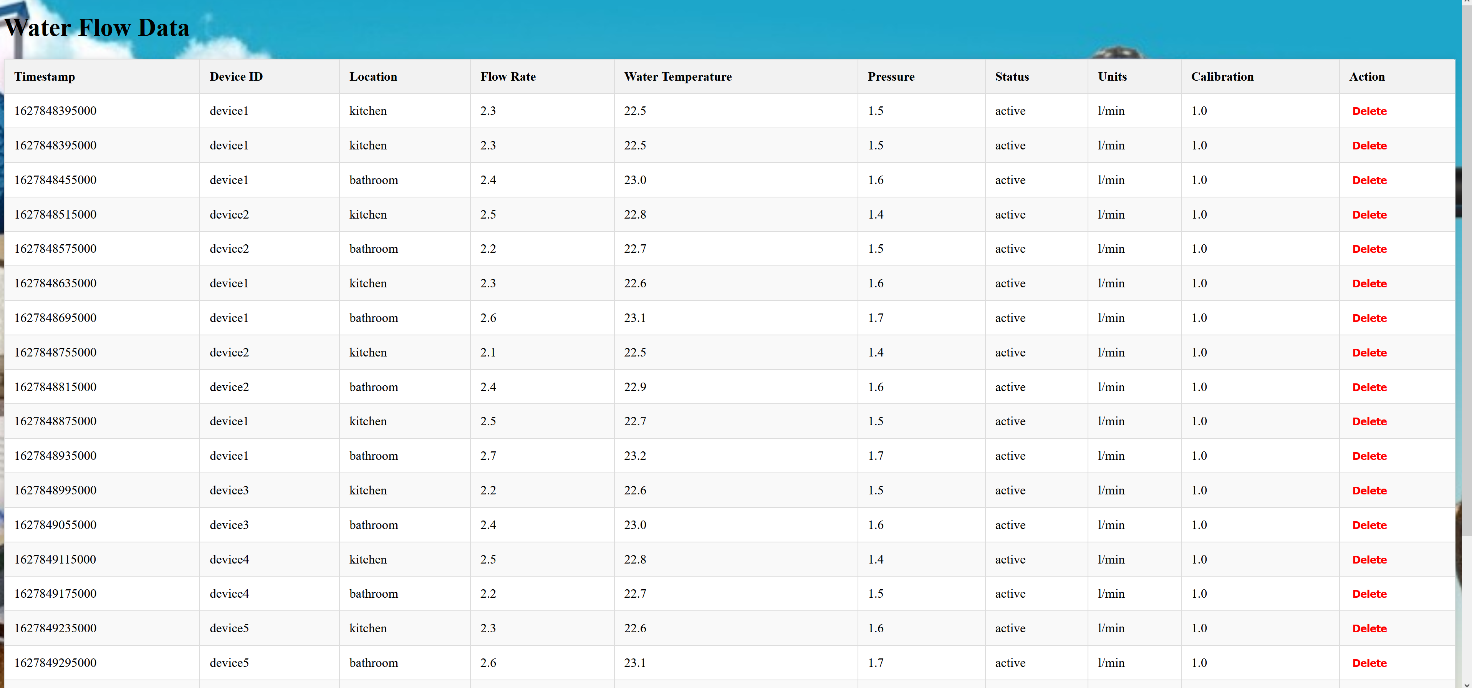


Рисунок 3.3 – Головне меню клієнтської частини

# 4. СТРУКТУРА БАЗИ ДАНИХ

## 4.1 Побудова ER-діаграми

ER-діаграма (Entity-Relationship) ілюструє сутності в базі даних, їх атрибути та взаємозв'язки між ними. Для системи моніторингу водопостачання ER-діаграма включає дві основні сутності: IoTDevice і DataRecord.

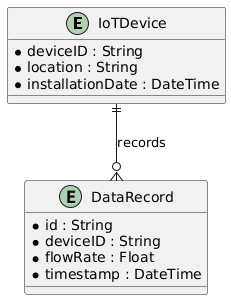


Рисунок 4.1 – ER-діаграма

## 4.2 Опис сутностей та атрибутів

1. IoTDevice є сутністю, яка представляє пристрій, що моніторить витрату води. Ця сутність має три ключові атрибути:

* deviceID: Унікальний ідентифікатор для кожного пристрою, який є первинним ключем у базі даних. Це поле використовують для ідентифікації кожного окремого пристрою;
* location: Місцезнаходження пристрою, яке допомагає визначити, де саме знаходиться пристрій. Це може бути корисним для організації та управління мережі пристроїв;
* installationDate: дата встановлення пристрою, що дозволяє відстежувати, коли пристрій був встановлений і може бути корисним для обслуговування та оновлень.

*DataRecord* є сутністю, що містить записи даних, зібраних з пристрою. Вона має такі атрибути:

* *id*: унікальний ідентифікатор запису даних, який є первинним ключем і допомагає ідентифікувати кожен окремий запис;
* *deviceid*: ідентифікатор пристрою, з якого були отримані дані. Це поле є зовнішнім ключем і забезпечує зв'язок між записом даних та відповідним пристроєм у сутності IoTDevice;
* *flowRate*: показник витрати води, виміряний датчиком. Це ключовий атрибут, який містить інформацію про обсяг витраченої води;
* *timestamp*: час, коли були зібрані дані. Це поле дозволяє відстежувати, коли саме були отримані вимірювання, і є важливим для аналізу історичних даних.

## 4.3 Взаємозв'язки між сутностями

Зв'язок між цими сутностями є один-до-багатьох, це означає означає, що один пристрій може створювати багато записів даних. Кожен запис даних містить посилання на конкретний пристрій через deviceID, що забезпечує коректне збереження і взаємодію даних у базі даних.

# ВИСНОВКИ

У даній курсовій роботі було розглянуто розробку програмної системи для контролю водопостачання у квартирі, що включає в себе як апаратну, так і програмну частини. Розроблена система забезпечує ефективний моніторинг витрати води, що дозволяє не тільки виявляти аномалії у споживанні, але й здійснювати контроль за витратами в режимі реального часу.   
 Концепція рішення базується на використанні IoT-пристроїв для збору даних про витрату води, які передаються на сервер для обробки і зберігання. Веб-інтерфейс забезпечує доступ до даних та можливість управління системою. Це забезпечує інтеграцію сучасних технологій для побутового контролю і управління ресурсами, що дозволяє реалізувати концепцію зручного  
і ефективного управління водопостачанням. У рамках первинного випуску було реалізовано основний функціонал системи, включаючи серверну частину,  
IoT-пристрій і базову версію веб-інтерфейсу. Наступні випуски   
можуть включати розширення функціоналу, такі як додаткові аналітичні   
інструменти або інтеграцію з іншими системами. Обмеження проекту   
зумовлені необхідністю забезпечення основного функціоналу без додаткових   
складнощів, таких як високі вимоги до безпеки або масштабованості.  
 Система має важливе значення для покращення управління водопостачанням у сучасних житлових приміщеннях. Профілі зацікавлених сторін включають як кінцевих користувачів, так і постачальників комунальних послуг. Пріоритети проекту зосереджені на забезпеченні ефективності, зручності використання та доступності для широкого кола користувачів.  
 Проектування включало детальний розгляд серверної частини системи, IoT-пристрою і клієнтської частини. Реалізація включала налаштування серверної частини з MongoDB, розробку IoT-пристрою на базі Arduino і створення веб-інтерфейсу для зручного доступу до даних. Це забезпечило функціональність, що відповідає сучасним вимогам до інтеграції та управління даними.  
 Отже, реалізація програмної системи для контролю водопостачання у квартирі забезпечує інтегроване рішення для моніторингу і управління ресурсами, що сприяє підвищенню ефективності використання води та зниженню витрат. Проект довів свою ефективність у вирішенні завдань, що стоять перед сучасними системами управління ресурсами, і має потенціал для подальшого розвитку та вдосконалення.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. ДСТУ 7.1:2006 "Система стандартів з інформації, бібліотечної та видавничої справи. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання
2. Wokwi – сайт для емуляції ІоТ-пристроїв: <https://wokwi.com/> (дата звернення – 15.08.2024)
3. Методичні вказівки до курсової роботи з навчальної дисципліни «Архітектура програмного забезпечення» для студентів усіх форм навчання спеціальності 121 – «Інженерія програмного забезпечення» [Електронне видання] / Упоряд. В.О. Лещинський, І.О. Лещинська, І.П. Сокорчук – Харків: ХНУРЕ, 2021. – 37 с.
4. Banica, L., Burtescu, E., & Banica, A. (2019). "IoT-Based Smart Home Automation and Monitoring System". In *Proceedings of the 2019 11th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pp. 1-6. IEEE. DOI: 10.1109/ECAI46879.2019.9042116. (дата звернення: 15 серпня 2024 р)
5. Shah, S. L., Sharma, V., & Kumar, P. (2020). "IoT-based Water Management System for Smart Cities". In *2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC)*, pp. 132-136. IEEE. DOI: 10.1109/PARC49193.2020.236650. (дата звернення: 15 серпня 2024 р)