

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ  
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ  
ТЕХНОЛОГІЙ КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО  
ЗАБЕЗПЕЧЕННЯ  
АВТОМАТИЗОВАНИХ СИСТЕМ**

**КВАЛІФІКАЦІЙНА РОБОТА**

**на тему: «Розробка моделі інформаційної системи трекінгу об'єктів з  
використанням ІоТ-технологій»**

на здобуття освітнього ступеня магістра  
зі спеціальності 126 Інформаційні системи та технології  
(код, найменування спеціальності)  
освітньо-професійної програми Інформаційні системи та технології  
(назва)

*Кваліфікаційна робота містить результати власних досліджень.  
Використання ідей, результатів і текстів інших авторів мають  
посилання на відповідне джерело*

\_\_\_\_\_ Данііл ДОРОХІН  
(підпис) *Ім'я, ПРІЗВИЩЕ здобувача*

Виконав: здобувач вищої освіти гр. ІСД-42

Данііл ДОРОХІН

Ім'я, ПРІЗВИЩЕ

Керівник:

*Науковий ступінь  
вчене звання*

phD Валентина ДАНИЛЬЧЕНКО

Ім'я, ПРІЗВИЩЕ

Рецензент:

*науковий ступінь,  
вчене звання*

\_\_\_\_\_   
Ім'я, ПРІЗВИЩЕ

**Київ 2024**

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ  
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**  
**Навчально-науковий інститут інформаційних технологій**

Кафедра Інженерії програмного забезпечення автоматизованих систем

Ступінь вищої освіти Бакалавр

Спеціальність Інформаційні системи та технології

Освітньо-професійна програма Інформаційні системи та технології

**ЗАТВЕРДЖУЮ**

Завідувач кафедрою ІПЗАС

\_\_\_\_\_ Каміла СТОРЧАК  
« \_\_\_\_\_ » \_\_\_\_\_ 2024 р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Дорохін Данііл В'ячеславович

*(прізвище, ім'я, по батькові здобувача)*

1. Тема кваліфікаційної роботи: Аналіз використання IoT в галузі охорони здоров'я для попередження захворювань та вдосконалення медичної діагностики..

керівник кваліфікаційної роботи Валентина Данильченко *phD*,

*(Ім'я, ПРИЗВИЩЕ науковий ступінь, вчене звання)*

затвердені наказом Державного університету інформаційно-комунікаційних технологій від «27» лютого 2024 р. № 36

2. Строк подання кваліфікаційної роботи «31» травня 2024 р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, технічна документація систем попередження захворювань, вимоги до використання в медичній сфері ІОТ компонентів.

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

Моніторинг використання ІОТ в галузі трекінгу об'єктів

Апаратне забезпечення ІОТ в галузі трекінгу об'єктів

5. Перелік графічного матеріалу: *презентація*
1. Теоретична частина
  2. Апаратні складові систем
  3. Моніторинг програмного забезпечення систем

6. Дата видачі завдання «27» лютого 2024 р.

### КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	27.02-05.03.2024	
2	Вивчення теоретичних основ інтернет речей	06.03-11.03.2024	
3	Дослідження технічних аспектів використання та впровадження ІоМТ	12.03-27.03.2024	
4	Аналіз проблем впровадження ІоМТ	28.03-10.04.2024	
5	Огляд практичного впровадження та прикладів застосування ІоМТ побуті	11.04-15.05.2024	
6	Аналіз оптимальних технологій украління та з'язку	16.05-22.05.2024	
7	Оформлення роботи: вступ, висновки, реферат	23.05-24.05.2024	
8	Розробка демонстраційних матеріалів	24.05-25.05.2024	

Здобувач(ка) вищої освіти

\_\_\_\_\_ (підпис)

Данііл ДОРОХІН

(Ім'я, ПРІЗВИЩЕ)

Керівник кваліфікаційної роботи

\_\_\_\_\_ (підпис)

Валентина ДАНИЛЬЧЕНКО

(Ім'я, ПРІЗВИЩЕ)

## РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня бакалавра: стор.65, рис.38, джерел 49.

*Мета роботи* – глибокий та системний аналіз використання IoT в цій сфері для ефективного трекінгу об'єктів та покращення управління ними.

*Об'єкт дослідження* – Об'єктом дослідження є використання IoT в галузі трекінгу об'єктів з метою оптимізації процесів управління та контролю за ними.

*Предмет дослідження* – Предметом дослідження є аналіз технологій IoT, які використовуються для створення інформаційних систем трекінгу об'єктів, а також дослідження існуючих пристроїв, сенсорів та систем моніторингу, що використовуються для збору даних про об'єкти.

*Короткий зміст роботи:* робота включає аналіз наукових статей, книг, конференцій та інших джерел для отримання розуміння сучасного стану IoT в сфері для трекінгу об'єктів та вивчення конкретних випадків успішного впровадження IoT- рішень.

**КЛЮЧОВІ СЛОВА:** ІНТЕРНЕТ РЕЧЕЙ, ТРЕКІНГ ОБ'ЄКТІВ, ІОТ-ТЕХНОЛОГІЇ, ІНФОРМАЦІЙНІ СИСТЕМИ, УПРАВЛІННЯ ОБ'ЄКТАМИ.





## ЗМІСТ

ВСТУП.....	8
Розділ 1. ТЕОРЕТИЧНІ ОСНОВИ РОЗРОБКИ ІНФОРМАЦІЙНИХ СИСТЕМ ТРЕКІНГУ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ ІОТ_ТЕХНОЛОГІЙ.....	10
1.1 Поняття та класифікація систем трекінгу об'єктів.....	10
1.2 ІоТ-технології та їх застосування в системах трекінгу об'єктів.....	12
1.3 Архітектура інформаційних систем трекінгу об'єктів.....	15
1.4 Методи та моделі розробки інформаційних систем.....	17
1.5 Забезпечення інформаційної безпеки в системах трекінгу об'єктів.....	21
РОЗДІЛ 2. АНАЛІЗ ІСНУЮЧИХ СИСТЕМ ТРЕКІНГУ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ ІОТ-ТЕХНОЛОГІЙ.....	24
2.1 Огляд існуючих систем трекінгу об'єктів.....	24
2.2 Класифікація систем трекінгу об'єктів за типами об'єктів, що відстежуються.....	25
2.3 Порівняльний аналіз існуючих систем трекінгу об'єктів за функціональністю.....	34
2.4 Аналіз протоколів та технологій, що використовуються в системах трекінгу об'єктів.....	37
2.5 Оцінка ефективності існуючих систем трекінгу об'єктів.....	38
РОЗДІЛ 3. ВИБІР ПРОГРАМНИХ ТА АПАРАТНИХ ЗАСОБІВ ДЛЯ ПОБУДОВИ ІНФОРМАЦІЙНЕ.....	41
3.1 Вибір апаратних засобів для інформаційної системи трекінгу об'єктів..	41
3.2 Вибір програмного забезпечення для інформаційної системи трекінгу об'єктів.....	45
3.3 План роботи систем трекінгу об'єктів.....	53
3.4 Розробка макету GPS-трекера.....	54
3.5 Розробка серверу для моделі.....	54
ВИСНОВКИ.....	62
ПЕРЕЛІК ПОСИЛАНЬ.....	63

## ВСТУП

Актуальність теми дослідження. В сучасному світі все більшого значення набувають системи трекінгу об'єктів, які використовуються в різних сферах життя, таких як логістика, транспорт, охорона, сільське господарство, промисловість та багато інших. Ці системи дозволяють відстежувати місцезнаходження та інші характеристики об'єктів в режимі реального часу, що дає можливість оптимізувати процеси, підвищити ефективність роботи та безпеку. Зростання популярності систем трекінгу об'єктів обумовлене розвитком Інтернету речей (IoT), який об'єднує фізичні об'єкти з віртуальним світом за допомогою датчиків, мережевих технологій та програмного забезпечення. IoT-технології дозволяють збирати дані про об'єкти в режимі реального часу, що робить системи трекінгу більш точними, надійними та інформативними. Розробка та вдосконалення інформаційних систем трекінгу об'єктів з використанням IoT-технологій є актуальною науково-практичною проблемою, яка має значний потенціал для розвитку різних галузей економіки та суспільства.

**Метою** цієї дипломної роботи є розробка моделі інформаційної системи трекінгу об'єктів з використанням IoT-технологій.

**Для досягнення поставленої мети необхідно вирішити наступні завдання:** Провести аналіз існуючих систем трекінгу об'єктів та їх класифікацію за різними критеріями; Вивчити IoT-технології та їх застосуваність в системах трекінгу об'єктів; Розробити архітектуру інформаційної системи трекінгу об'єктів з використанням IoT-технологій; Розробити функціональну, інформаційну та процесну моделі інформаційної системи трекінгу об'єктів; Вибрати апаратне та програмне забезпечення для інформаційної системи трекінгу об'єктів; Реалізувати інформаційну систему трекінгу об'єктів; Провести тестування та оцінку ефективності інформаційної системи трекінгу об'єктів.

**Об'єктом** дослідження є інформаційні системи трекінгу об'єктів з використанням IoT-технологій.

**Предметом** дослідження є методи та моделі розробки інформаційних систем трекінгу об'єктів з використанням IoT-технологій.

Методи дослідження. Для вирішення поставлених завдань у дипломній роботі будуть використовуватися наступні методи дослідження:

1. Аналітичний метод;
2. Системний аналіз;
3. Моделювання;
4. Експериментальні дослідження.

**Наукова новизна** дипломної роботи полягає в розробці моделі інформаційної системи трекінгу об'єктів з використанням IoT-технологій, яка буде враховувати сучасні вимоги до таких систем.

**Практична значимість** дослідження полягає в тому, що розроблена модель



інформаційної системи трекінгу об'єктів може бути використана для розробки та впровадження нових систем трекінгу об'єктів в різних сферах діяльності.

# 1 АНАЛІЗ МЕТОДІВ СКОРОЧЕННЯ ОБСЯГІВ ПЕРЕДАВАНИХ ДАНИХ В СИСТЕМАХ ІНТЕРНЕТУ РЕЧЕЙ.

## 1.1 Поняття та класифікація систем трекінгу об'єктів

Система трекінгу об'єктів (СТО) - це комп'ютерна система, яка використовується для відстеження та моніторингу місцезнаходження та інших характеристик об'єктів в режимі реального часу. Ці системи можуть використовуватися в різних сферах, таких як: Логістика та транспорт: Відстеження вантажів, транспортних засобів та людей під час їх пересування.

Охорона: Відстеження людей та майна на об'єктах, що охороняються.

Сільське господарство: Відстеження тварин та сільгосптехніки.

Промисловість: Відстеження деталей та обладнання на виробничих лініях.

Інші сфери: Відстеження активів, пацієнтів, спортсменів та інших об'єктів.

*Компоненти системи трекінгу об'єктів*

Основними компонентами СТО є:

1. Об'єкти, що відстежуються: Об'єкти, які оснащені датчиками або мітками, що дозволяють системі їх ідентифікувати та відстежувати. Ці датчики можуть бути різного типу, в залежності від специфіки системи та об'єктів, які відстежуються.

**GPS-датчики:** Використовуються для визначення точного місцезнаходження об'єкта за допомогою супутників GPS.

**RFID-мітки:** Радіочастотні мітки, які дозволяють ідентифікувати об'єкти на коротких відстанях.

**Bluetooth-маяки:** Використовуються для визначення відстані до об'єкта та його напрямку.

**Датчики руху:** Виявляють рух об'єкта та його швидкість.

**Датчики температури:** Вимірюють температуру об'єкта або навколишнього середовища.

**Датчики вібрації:** Виявляють вібрацію об'єкта, що може бути ознакою неполадки або несанкціонованого доступу.

**Інші датчики:** Можуть використовуватися для збору інших даних про об'єкт, таких як тиск, рівень рідини, освітленість тощо.

2. Інфраструктура зв'язку: Мережа датчиків, базових станцій, Wi-Fi-точок доступу та інших пристроїв, які забезпечують зв'язок між об'єктами, що відстежуються, та системою. Ця інфраструктура може бути дротовою або бездротовою, залежно від потреб та можливостей об'єкта. Базові станції: Використовуються для прийому сигналів з GPS-датчиків та RFID-міток. Wi-Fi-точки доступу: Забезпечують зв'язок з Bluetooth-маяками та іншими датчиками, які використовують Wi-Fi для передачі даних. Мобільні мережі: Можуть використовуватися для передачі даних з об'єктів, які знаходяться поза зоною дії Wi-Fi або базових станцій.

3. Центр керування: Програмне забезпечення, яке збирає дані з датчиків, обробляє їх та візуалізує на дисплеї користувача. Центр керування може мати різні функції, такі як: Відображення місцезнаходження об'єктів на карті: Користувач може бачити, де знаходяться всі об'єкти, що відстежуються, в режимі реального часу. Відстеження маршрутів: Система може записувати маршрути об'єктів та візуалізувати їх на карті. Отримання оповіщень: Система може надсилати оповіщення користувачам, коли об'єкт виходить за межі дозволеної зони, зупиняється на непередбачуваний час або коли виникають інші події. Аналіз даних: Система може аналізувати дані про місцезнаходження, рух та інші характеристики об'єктів для виявлення закономірностей та прийняття кращих рішень.

4. Користувацький інтерфейс: Інтерфейс, який дозволяє користувачеві взаємодіяти з системою трекінгу об'єктів. Цей інтерфейс може бути веб-застосунком, мобільним додатком або програмним забезпеченням, встановленим на комп'ютері. Користувацький інтерфейс повинен бути зручним та інтуїтивно зрозумілим, щоб користувачі могли легко отримувати необхідну інформацію та керувати системою.

Принципи роботи системи трекінгу об'єктів

Збір даних: Датчики, встановлені на об'єктах, що відстежуються, збирають дані про їх місцезнаходження, стан та інші характеристики.

Передача даних: Дані передаються через інфраструктуру зв'язку до центру керування. Це може відбуватися в режимі реального часу або з певною періодичністю, залежно від типу даних та потреб системи.

Обробка даних: Центр керування отримує дані, очищає їх від помилок, проводить необхідні обчислення та аналіз.

Візуалізація даних: Оброблені дані відображаються на користувацькому інтерфейсі у вигляді карт, графіків, таблиць та інших елементів.

**Генерація оповіщень:** Система може генерувати оповіщення для користувачів, коли відбуваються певні події, наприклад, коли об'єкт виходить за межі дозволеної зони або коли датчики фіксують підозрілу активність.

**Переваги використання систем трекінгу об'єктів:**

**Підвищення ефективності:** Системи трекінгу об'єктів дозволяють оптимізувати маршрути, скорочувати час доставки, зменшувати втрати та кращим чином управляти ресурсами. Наприклад, логістичні компанії можуть використовувати ці системи для відстеження вантажів та оптимізації маршрутів доставки, що дозволяє скоротити витрати на паливо та підвищити ефективність роботи.

**Підвищення безпеки:** Системи трекінгу об'єктів дозволяють відстежувати людей та майно на об'єктах, що охороняються, запобігати крадіжкам та іншим злочинам. Охоронні компанії можуть використовувати ці системи для відстеження патрульних та реагування на інциденти безпеки в режимі реального часу.

**Покращення прийняття рішень:** Системи трекінгу об'єктів дозволяють збирати дані про місцезнаходження та інші характеристики об'єктів, що відстежуються, які можуть бути використані для прийняття кращих рішень. Наприклад, сільськогосподарські підприємства можуть використовувати ці системи для відстеження тварин та збору даних про їх стан здоров'я та місцезнаходження, що дозволяє приймати обґрунтовані рішення щодо годування, лікування та догляду за тваринами.

**Зниження витрат:** Системи трекінгу об'єктів дозволяють скоротити витрати на логістику, охорону, ремонт та інші сфери. Наприклад, промислові підприємства можуть використовувати ці системи для відстеження обладнання та запобігання поломкам, що дозволяє знизити витрати на ремонт та простій виробництва.

*Недоліки використання систем трекінгу об'єктів*

**Висока вартість:** Впровадження та експлуатація систем трекінгу об'єктів може бути дорогою. Це пов'язано з вартістю датчиків, інфраструктури зв'язку, програмного забезпечення та обслуговування системи.

**Проблеми конфіденційності:** Збір та зберігання даних про місцезнаходження людей може викликати проблеми конфіденційності. Користувачі повинні бути впевнені, що їхні дані використовуються тільки за призначенням та захищені від несанкціонованого доступу.

**Технічні складності:** Застосування систем трекінгу об'єктів може бути складним з технічної точки

## **1.2 IoT-технології та їх застосування в системах трекінгу об'єктів**

Інтернет речей (IoT) революціонує різні сфери, і системи трекінгу об'єктів не є

винятком. IoT-технології надають безліч нових можливостей для відстеження й моніторингу місцезнаходження, стану та інших характеристик об'єктів в режимі реального часу.

#### *Переваги застосування IoT в системах трекінгу об'єктів*

**Впровадження IoT в системи трекінгу об'єктів несе в собі значні переваги:** Підвищена точність та деталізація відстеження: IoT-датчики збирають широкий спектр даних, таких як температура, вібрація, тиск, освітленість, що забезпечує більш глибоке розуміння стану та місцезнаходження об'єктів.

**Зменшення розміру та вартості міток:** IoT-мітки, такі як RFID-мітки або Bluetooth-маяки, значно менші та дешевші за традиційні GPS-трекери, роблячи відстеження доступнішим для ширшого кола об'єктів.

**Більша енергоефективність:** Багато IoT-датчиків мають низьке споживання енергії, що дозволяє їм працювати довше без заміни батарейок, економлячи час та кошти.

**Підтримка двосторонньої комунікації:** Деякі IoT-пристрої не лише збирають дані, але й дозволяють керувати об'єктами на відстані, роблячи систему більш гнучкою та динамічною.

#### *Приклади застосування IoT в системах трекінгу об'єктів*

Різноманітні галузі вже використовують IoT для трекінгу об'єктів:

**Логістика та транспорт:** Відстеження вантажів, транспортних засобів та людей під час їх пересування, оптимізація маршрутів, запобігання крадіжкам та контроль умов зберігання.

**Охорона:** Відстеження людей та майна на об'єктах, що охороняються, виявлення руху, відкриття дверей та вікон, швидке реагування на інциденти.

**Сільське господарство:** Відстеження тварин, сільгосптехніки, параметрів ґрунту та навколишнього середовища, оптимізація виробництва, поліпшення догляду за тваринами.

**Промисловість:** Відстеження стану обладнання, виявлення несправностей, проведення профілактичного обслуговування, запобігання поломкам та скорочення простоїв.

**Охорона здоров'я:** Відстеження пацієнтів, медичного обладнання, лікарських препаратів, контроль за станом здоров'я пацієнтів, оптимізація логістики лікарських засобів.

**Інші сфери:** Відстеження активів, спортсменів, домашніх тварин, посилок, збір даних для аналізу поведінки та прийняття кращих рішень.

IoT-технології, що використовуються в системах трекінгу  
Різноманітні IoT-технології забезпечують функціональність систем трекінгу:  
RFID-мітки: Невеликі радіочастотні мітки, що прикріплюються до об'єктів для

ідентифікації на відстані до декількох метрів.

**GPS-датчики:** Використовуються для визначення точного місцезнаходження об'єкта за допомогою супутників GPS.

**Датчики Bluetooth:** Відстежують об'єкти на коротких відстанях (до 10 метрів) за допомогою Bluetooth-сигналів.

**Wi-Fi-маяки:** Визначають місцезнаходження об'єктів у приміщеннях, фіксуючи Wi-Fi-сигнал з IoT-пристроїв.

**Датчики руху:** Виявляють рух об'єкта, використовуються для відстеження переміщення або виявлення підозрілої активності.

**Датчики температури:** Вимірюють температуру об'єкта або навколишнього середовища, що допомагає контролювати умови зберігання, виявляти несправності обладнання тощо.

**Датчики вібрації:** Виявляють вібрацію об'єкта, допомагаючи відстежувати стан обладнання, виявляти механічні пошкодження.

**Датчики вологості:** Вимірюють рівень вологості повітря, що важливо для сільського господарства, складського зберігання, будівельної індустрії тощо.

**Датчики освітлення:** Визначають рівень освітленості, який може бути корисним для розумних теплиць, автоматичного керування освітленням у будівлях тощо.

**Відеокамери:** Забезпечують візуальний моніторинг об'єктів та оточення, допомагаючи у вирішенні складних ситуацій та аналізі інцидентів.

*Вибір IoT-технологій для конкретної системи трекінгу*

Вибір технологій залежить від кількох ключових факторів:

**Тип об'єктів, що відстежуються:** Різні об'єкти потребують різних технологій. Наприклад, для транспортних засобів підійдуть GPS-датчики, а для відстеження інструментів на будмайданчику краще використати RFID-мітки.

**Точність відстеження:** Деякі технології, як GPS, забезпечують високу точність, інші, як Bluetooth, призначені для меншої відстані з меншою погрешністю.

**Відстань відстеження:** Технології мають обмеження по відстані. GPS-датчики працюють на великих відстанях, а RFID-мітки ефективніші на короткій дистанції.

**Вартість:** Рішення повинно бути економічно обґрунтованим. Дешевші RFID-мітки можуть підійти для певних випадків, тоді як для критично важливих об'єктів знадобиться точніший, але й дорожчий GPS-трекінг.

**Енергоспоживання:** Деякі датчики, як RFID-мітки, мають низьке споживання, що дозволяє їм працювати роками на одній батареї. Інші, наприклад камери, потребують частішої заміни живлення.

Розвиток IoT-технологій та майбутнє систем трекінгу

Сфера IoT постійно розвивається, пропонуючи нові рішення для трекінгу об'єктів:

**Мініатюризація датчиків:** Зменшення розміру датчиків дозволить відстежувати ширший спектр об'єктів, не порушуючи їхньої функціональності.  
**Збільшення дальності дії:** Потужність сигналу та методи передачі даних удосконалюються, дозволяючи відстежувати об'єкти на більших відстанях.

**Самозарядні датчики:** Розвиваються технології живлення датчиків від вібрації, температури або навколишнього середовища, що мінімізує потребу в заміні батарейок.

**Штучний інтелект (ШІ):** Застосування ШІ для аналізу даних з датчиків дозволить прогнозувати технічні несправності, оптимізувати логістичні процеси та приймати більш ефективні рішення.

**Інтернет речей (IoT)** революціонізує системи трекінгу об'єктів, надаючи безліч нових можливостей для відстеження й моніторингу місцезнаходження, стану та інших характеристик об'єктів в режимі реального часу.

IoT-технології пропонують значні переваги: Підвищена точність та деталізація відстеження, Зменшення розміру та вартості міток, Більша енергоефективність, Підтримка двосторонньої комунікації

Різноманітні галузі вже використовують IoT для трекінгу об'єктів, від логістики та охорони до сільського господарства та промисловості. Широкий спектр IoT-технологій використовується в системах трекінгу: RFID-мітки, GPS-датчики, Bluetooth-маяки, Wi-Fi-датчики, датчики руху, температури, вібрації, вологості, освітлення, відеокамери.

Вибір IoT-технологій залежить від типу об'єктів, що відстежуються, необхідної точності, відстані, бюджету та інших факторів.

Розвиток IoT-технологій відкриває нові можливості для систем трекінгу, роблячи їх більш потужними, надійними та універсальними, що сприятиме розвитку різних галузей.

### **1.3 Архітектура інформаційних систем трекінгу об'єктів**

Інформаційні системи трекінгу об'єктів (ІСТО) - це складні програмні комплекси, що збирають, обробляють та візуалізують дані про місцезнаходження та стан об'єктів в режимі реального часу. Архітектура ІСТО відіграє ключову роль у забезпеченні їх ефективності, масштабованості та безпеки.

Типова архітектура ІСТО складається з наступних компонентів:

1. Рівень збору даних: Датчики: Цей рівень включає різні датчики, які

збирають дані про об'єкти, що відстежуються. Це можуть бути GPS-датчики, RFID-мітки, датчики руху, датчики температури, датчики вібрації тощо. Шлюзи: Шлюзи збирають дані з датчиків, форматують їх та передають на наступний рівень.

2. Рівень передачі даних: Мережі: Цей рівень використовує різні мережеві технології, такі як мобільні мережі, Wi-Fi, Bluetooth, супутниковий зв'язок, для передачі даних з шлюзів на сервери. Маршрутизатори: Маршрутизатори направляють дані до відповідних серверів.

3. Рівень обробки даних: Сервери: Сервери приймають дані з рівня передачі даних, обробляють їх, зберігають та аналізують. Бази даних: Бази даних зберігають зведені дані про об'єкти, їх місцезнаходження, стан тощо. Аналітика даних: Системи аналітики даних дозволяють виявляти закономірності, прогнозувати події та приймати кращі рішення.

4. Рівень візуалізації даних: Інтерфейси користувача: Інтерфейси користувача надають користувачам доступ до даних про об'єкти в зручній візуальній формі. Це можуть бути веб-інтерфейси, мобільні додатки, дисплеї тощо. Карти: Карти візуалізують місцезнаходження об'єктів на картах. Графіки та діаграми: Графіки та діаграми візуалізують стан об'єктів, зміни в часі тощо.

5. Рівень управління: Система адміністрування: Система адміністрування дозволяє керувати користувачами, налаштовувати систему, оновлювати програмне забезпечення тощо. Система безпеки: Система безпеки захищає систему від несанкціонованого доступу, кібератак тощо. Існують різні типи архітектур ІСТО, які обираються залежно від конкретних потреб та вимог. Ось кілька прикладів: Централізована архітектура: У централізованій архітектурі всі дані збираються, обробляються та зберігаються на центральному сервері. Це проста та масштабована архітектура, але вона може бути менш гнучкою та стійкою до збоїв. Розподілена архітектура: У розподіленій архітектурі дані обробляються на декількох серверах, розташованих у різних місцях. Це більш гнучка та стійка до збоїв архітектура, але вона може бути складнішою та дорожчою в реалізації. Гібридна архітектура: Гібридна архітектура поєднує в собі елементи централізованої та розподіленої архітектур. Це може бути оптимальним рішенням для систем з великою кількістю даних та складними вимогами.

При проектуванні архітектури ІСТО важливо враховувати наступні аспекти: Масштабованість: Архітектура повинна бути спроектована таким чином, щоб легко масштабуватися та підтримувати зростаючий обсяг даних та кількість об'єктів, що відстежуються. Безпека: Система повинна бути захищена від



несанкціонованого доступу, кібератак та втрати даних. Інтеграція: ІСТО повинна інтегруватися з іншими системами, такими як ERP, CRM, системами управління транспортом тощо. Надійність: Система повинна бути надійною та працювати без збоїв 24/7. Реальний час: Дані повинні оброблятися та візуалізуватися в режимі реального часу. Енергоефективність: Особливо важливо для проектів, де використовуються датчики на батарейках.

Вибір та проектування правильної архітектури ІСТО є критично важливим для успішного функціонування системи трекінгу об'єктів. Враховуючи різноманітні потреби та вимоги, необхідно обрати архітектуру, яка забезпечить масштабованість, безпеку, інтеграцію, надійність, обробку даних в реальному часі та енергоефективність.

#### **1.4 Методи та моделі розробки інформаційних систем**

Життєвий цикл (ЖЦ) ІС - це не просто хронологічний порядок етапів, а динамічний процес, що охоплює всі аспекти розробки, впровадження та експлуатації системи. Кожен етап ЖЦ має чітко визначені цілі, завдання та результати, які забезпечують планомірний перехід до наступного етапу.

Етапи ЖЦ ІС:

##### **1. Планування:**

Ціль: Визначення цілей проекту, вимог до системи та її функціональних можливостей.

Завдання: Проведення аналізу потреб користувачів та бізнесу. Визначення обмежень проекту (бюджет, терміни, ресурси). Розробка технічного завдання (ТЗ). Створення плану проекту

Результат: Чітке розуміння того, що система повинна робити, як вона буде використовуватися та які ресурси необхідні для її розробки.

##### **2. Проектування:**

Ціль: Розробка детальної архітектури системи, інтерфейсів користувача, алгоритмів та програмного забезпечення.

Завдання: Розробка архітектури системи (програмне забезпечення, апаратне забезпечення, мережа). Проектування інтерфейсів користувача. Розробка алгоритмів та програмного забезпечення. Документування проекту

Результат: Детальний план, що описує, як система буде побудована та як вона буде працювати.

### 3.Реалізація:

Ціль: Написання програмного коду, тестування компонентів системи та інтеграція всіх частин системи.

Завдання: Написання програмного коду. Тестування модулів та компонентів системи. виправлення помилок та доопрацювання системи. Інтеграція всіх частин системи

Результат: Працююча система, готова до впровадження.

### 4.Впровадження:

Ціль: Встановлення системи у користувачів, навчання персоналу та надання підтримки користувачам.

Завдання: Встановлення та налаштування програмного забезпечення. Навчання персоналу роботі з системою. Надання підтримки користувачам. Тестування системи в реальних умовах

Результат: Система впроваджена та готова до використання.

### 5.Експлуатація:

Ціль: Використання системи користувачами, моніторинг її роботи та усунення помилок.

Завдання: Надання користувачам доступу до системи. Моніторинг роботи системи та виявлення проблем. Усунення помилок та доопрацювання системи. Оновлення програмного забезпечення

Результат: Система використовується користувачами та відповідає їхнім потребам.

### 6.Підтримка:

Ціль: Надання технічної підтримки користувачам, оновлення програмного забезпечення та внесення змін до системи за потребою.

Завдання: Надання технічної підтримки користувачам. Оновлення програмного забезпечення. Внесення змін до системи за потребою. Навчання нових користувачів

Результат: Система постійно підтримується та оновлюється, щоб відповідати потребам користувачів.

#### 7. Виведення з експлуатації:

Ціль: Виведення системи з експлуатації, видалення даних та утилізація обладнання.

Завдання: Створення резервних копій даних. Видалення даних з системи. Демонтаж обладнання. Утилізація обладнання та програмного забезпечення

Результат: Система виведена з експлуатації, дані видалені, обладнання утилізовано.

Вибір та переваги окремих моделей ЖЦ ІС:

Вибір моделі ЖЦ залежить від багатьох факторів, зокрема:

Розмір та складність проекту: Для великих і складних проектів потрібна більш формальна модель ЖЦ, наприклад каскадна або спіральна, з чітким розподілом відповідальності та ретельним плануванням.

Вимоги до гнучкості: Якщо вимоги до системи можуть змінюватися протягом проекту, потрібна гнучкіша модель, наприклад ітеративна або екстремального програмування, що дозволяє швидко реагувати на зміни.

Досвід команди: Якщо команда не має досвіду роботи з певною моделлю, краще обрати простішу, як каскадна.

Бюджет проекту: Деякі моделі, як спіральна, можуть бути дорожчими через додаткові етапи аналізу ризиків.

Розглянемо переваги та недоліки поширених моделей ЖЦ:

Каскадна (водоспадна) модель:

Переваги: Простота та легкість управління. Чітке розмежування відповідальності між командами. Підходить для проектів з чіткими вимогами, що не змінюються.

Недоліки: Негнучка, складно реагувати на зміни вимог на пізніх стадіях. Високий ризик виникнення помилок на пізніх стадіях, що призводить до додаткових витрат.

Ітеративна й інкрементальна модель:

Переваги: Гнучкість, можливість реагувати на зміни вимог. Швидке отримання зворотного зв'язку від користувачів та врахування його в наступних ітераціях. Зниження ризику виникнення критичних помилок.

Недоліки: Складність планування та контролю через постійні зміни. Можливість виникнення проблем з інтеграцією компонентів, розроблених на різних ітераціях.

Спіральна модель:

Переваги: Поєднує гнучкість ітеративної моделі з контролем каскадної. Знижує ризики завдяки постійному аналізу на кожному витку спіралі. Підходить для складних проектів з високою невизначеністю.

Недоліки: Складність планування та управління. Висока вартість через додаткові етапи аналізу ризиків. Окрім класичних моделей, існують альтернативні підходи, що набирають популярності:

Модель прототипування: Швидке створення прототипів системи з мінімальним функціоналом для збору зворотного зв'язку від користувачів на ранніх етапах. Дозволяє визначити ключові функції та інтерфейси до початку повноцінної розробки.

Модель RAD (Rapid Application Development): Швидка розробка ІС за короткий проміжок часу (тижні або місяці) шляхом використання готових компонентів, CASE-засобів та тісної взаємодії з користувачами. Підходить для невеликих проектів з чіткими вимогами.

Модель екстремального програмування (XP): Постійне вдосконалення ІС

протягом коротких циклів розробки (тижні або менше). Кожен цикл включає планування, програмування, тестування та огляд коду. Підходить для гнучких проєктів, де вимоги можуть швидко змінюватися.

Вибір моделі залежить від конкретного проєкту та потреб команди розробників. Іноді використовують гібридний підхід, що поєднує елементи різних моделей. Методи та моделі розробки ІС є важливими інструментами для створення ефективних, надійних та функціональних інформаційних систем. Розуміння етапів ЖЦ, їх завдань та результатів дозволяє

### **1.5 Забезпечення інформаційної безпеки в системах трекінгу об'єктів**

Системи трекінгу об'єктів (СТО) збирають, обробляють та візуалізують дані про місцезнаходження та рух об'єктів. Ці дані можуть бути надзвичайно чутливими, адже містять інформацію про: Пересування людей та товарів: Локація вантажів, маршрути кур'єрів, місцезнаходження співробітників. Активність у певних зонах: Відвідування конфіденційних об'єктів, перебування в потенційно небезпечних місцях. Ідентифікацію та поведінку об'єктів: Відстеження переміщення підозрілих осіб, аналіз траєкторій руху транспорту.

Компрометація СТО може призвести до серйозних наслідків: Фінансові втрати: Викрадення цінних вантажів, шахрайство з логістичними операціями, втрата клієнтів. Загроза безпеці: Відстеження руху людей, виявлення вразливих місць, планування злочинів. Втрата репутації: Зливи інформації про клієнтів, порушення конфіденційності, шкода іміджу компанії.

Окрім зазначених вище загроз, СТО можуть бути вразливими до: Несанкціонованого прослуховування: Перехоплення даних про місцезнаходження та рух об'єктів з метою шпигунства або промислового шпіонажу. Втручання в сигнали: Зміна або блокування даних про місцезнаходження для маніпулювання системами або приховування злочинів. Атаки типу "людина-в-середині": Зловживання довірою авторизованого користувача для отримання доступу до СТО та викрадення даних. Фізичні атаки: Пошкодження або знищення обладнання СТО для виведення її з ладу або крадіжки даних.

Ефективний захист СТО потребує комплексного підходу, що поєднує в собі:

1. Технічні заходи: Контроль доступу: Обмеження доступу до СТО лише авторизованим користувачам з чітко визначеними ролями та повноваженнями. Шифрування: Захист даних СТО на всіх етапах: зберігання, передача, обробка. Використання стійких алгоритмів шифрування та протоколів Захист від шкідливого програмного забезпечення: Встановлення та регулярне оновлення антивірусного та антишпійонського програмного забезпечення. Фільтрація мережевого трафіку: Блокування підозрілих з'єднань, захист від DDoS-атак та інших мережевих загроз. Резервне копіювання даних: Регулярне створення резервних копій даних СТО для їх швидкого відновлення у разі збою або кібератаки. Аудит та тестування: Регулярні перевірки СТО на наявність вразливостей та проведення тестувань на проникнення для виявлення потенційних загроз.

2. Організаційні заходи: Розробка та впровадження політики інформаційної безпеки: Визначення чітких правил та процедур роботи з інформацією, що обробляється СТО. Навчання персоналу: Підвищення обізнаності користувачів СТО про інформаційні загрози та методи їх запобігання. Облік та моніторинг активності користувачів: Відстеження дій користувачів в системі для виявлення підозрілих операцій та запобігання несанкціонованому доступу.

3. Фізичні заходи: Контроль доступу до обладнання СТО: Обмеження фізичного доступу до датчиків, серверів та інших компонентів системи для запобігання їх пошкодженню або маніпуляціям.

Системи відеоспостереження та охорони: Застосування відеокамер та охоронних систем для фізичного захисту обладнання СТО та прилеглих територій. Використання багатофакторної автентифікації: Застосування додаткових факторів підтвердження особи користувача, крім пароля (відбитків пальців, кодів з SMS тощо).

Регулярне оновлення програмного забезпечення СТО: Встановлення останніх версій програмного забезпечення для усунення виявлених вразливостей. Утилізація даних за певним регламентом: Видалення застарілих або непотрібних даних про місцезнаходження та рух об'єктів відповідно до встановлених термінів зберігання.

Страхування інформаційних ризиків: Передача відповідальності за потенційні збитки від кібератак страховим компаніям.

Для забезпечення ефективного захисту СТО рекомендується дотримуватися міжнародних стандартів та національних регуляторних актів щодо інформаційної безпеки. Деякі з ключових документів:

- ISO/IEC 27001:2013 - міжнародний стандарт, що визначає вимоги до системи управління інформаційною безпекою організації.
- NIST Cybersecurity Framework (Національний інститут стандартів і технологій США) - рекомендації з розробки системи кіберзахисту.
- Закон України "Про захист інформації в інформаційно-комунікаційних системах" - визначає правові засади захисту інформації в Україні.

Забезпечення інформаційної безпеки є критично важливим завданням для будь-якої системи трекінгу об'єктів. Впровадження багаторівневого захисту, що поєднує технічні, організаційні та фізичні заходи, дозволить мінімізувати ризики витоку або втрати конфіденційних даних. Регулярний моніторинг, оновлення системи та підвищення кваліфікації персоналу дозволять підтримувати високий рівень захищеності СТО.

## **2 АНАЛІЗ ІСНУЮЧИХ СИСТЕМ ТРЕКІНГУ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ ІОТ-ТЕХНОЛОГІЙ**

### **2.1.Огляд існуючих систем трекінгу об'єктів**

Системи трекінгу об'єктів відіграють важливу роль у багатьох галузях завдяки можливості реального моніторингу та управління активами, транспортними засобами та персоналом. Ці системи використовують різноманітні технології для точного визначення місцезнаходження та відстеження об'єктів. Ось короткий огляд деяких існуючих систем трекінгу об'єктів:

**GPS-системи:** Системи трекінгу на основі GPS (Глобальна система позиціонування) використовують супутникові сигнали для визначення точного місцезнаходження об'єктів, що обладнані приймачами GPS. Широко застосовуються в транспорті, логістиці та управлінні активами, GPS-системи пропонують надійний трекінг місцезнаходження та можливості оптимізації маршрутів.

**RFID-системи:** Системи RFID (радіочастотна ідентифікація) складаються з RFID-міток, прикріплених до об'єктів, та RFID-читачів, які захоплюють дані з міток за допомогою радіохвиль. Використовуються в управлінні запасами, логістиці та роздрібній торгівлі, технологія RFID дозволяє автоматизовано відстежувати та ідентифікувати об'єкти в різних середовищах.

**ІоТ-системи:** Системи трекінгу на базі ІоТ (Інтернет речей) інтегрують датчики, пристрої та технології підключення для збору та аналізу даних в реальному часі. Використовуються для комплексного моніторингу активів, транспортних засобів та персоналу, системи на основі ІоТ надають цінні уявлення для прийняття рішень та оптимізації процесів.

**Системи трекінгу внутрішнього розташування (IPS):** IPS використовують технології, такі як Wi-Fi, Bluetooth та ультразвукові сигнали, для відстеження об'єктів у внутрішніх середовищах. Використовуються у торгових мережах, лікарнях та виробничих підприємствах, системи IPS забезпечують точний трекінг місцезнаходження та управління активами всередині приміщень.

**Системи управління транспортним флотом:** Системи управління транспортним флотом зосереджуються на відстеженні та управлінні транспортними засобами у комерційних флотах. Пропонують функції, такі як GPS-трекінг, моніторинг поведінки водіїв та планування технічного обслуговування для підвищення ефективності та безпеки флоту.

**Системи трекінгу персоналу:** Системи трекінгу персоналу використовують носимі пристрої, біометричні технології та додатки з визначенням місцезнаходження для моніторингу рухів та безпеки осіб. Використовуються в



медичних закладах, будівельній сфері та сфері безпеки для підвищення безпеки та захисту персоналу.

Загалом, існуючі системи трекінгу об'єктів пропонують різноманітні рішення, які відповідають конкретним потребам трекінгу в різних галузях. З розвитком технологій ці системи очікується ще більше удосконалення, розширення функціоналу та підвищення їх ефективності та використовуваності. Завдяки постійному розвитку технологій очікується зростання можливостей у сфері трекінгу об'єктів, що дозволить вдосконалити процеси моніторингу та управління, забезпечуючи більш точну, ефективну і безпечну роботу в різних сферах діяльності.

## **2.2. Класифікація систем трекінгу об'єктів за типами об'єктів, що відстежуються**

Системи трекінгу об'єктів (СТО) використовуються для відстеження й моніторингу переміщення людей, тварин, транспортних засобів та інших об'єктів. Їх застосовують у різних сферах, таких як логістика, транспорт, охорона, безпека, виробництво, сільське господарство та багато інших.

За типом об'єктів, що відстежуються:

1. Люди: Відстеження пересування людей в реальному часі, аналіз їх маршрутів та поведінки.
2. Транспортні засоби: Відстеження руху транспортних засобів, контроль логістичних операцій, запобігання викраденню.
3. Тварини: Відстеження диких або свійських тварин, контроль їх здоров'я та поведінки.
4. Вантажі: Відстеження пересування вантажів, контроль їх доставки, запобігання крадіжкам.
5. Інші об'єкти: Відстеження будь-яких об'єктів, оснащених відповідними датчиками або мітками.

За технологією трекінгу:

1. GPS: Найпоширеніша технологія, що використовує супутникові сигнали для визначення місцезнаходження об'єктів.
2. RFID: Використання радіочастотних міток для ідентифікації та локалізації об'єктів.

3. Wi-Fi: Відстеження об'єктів, оснащених Wi-Fi-пристроями, в приміщеннях або зонах з Wi-Fi покриттям.
4. Bluetooth: Відстеження об'єктів на коротких відстанях за допомогою Bluetooth-пристроїв.
5. Ультразвук: Використання ультразвукових датчиків для локалізації об'єктів в приміщеннях.
6. Відеокамери: Відстеження об'єктів та людей за допомогою відеокамер та систем аналізу зображень.

За сферою застосування:

1. Логістика: Відстеження пересування вантажів, контроль поставок, оптимізація маршрутів.
2. Транспорт: Відстеження руху транспортних засобів, контроль пасажирських перевезень, забезпечення безпеки.
3. Охорона: Відстеження людей та об'єктів на охоронюваних територіях, контроль доступу, запобігання злочинам.
4. Безпека: Відстеження людей та об'єктів в potentially dangerous environments, emergency response, disaster management.
5. Виробництво: Відстеження переміщення обладнання, контроль виробничих процесів, оптимізація ланцюгів постачання.
6. Сільське господарство: Відстеження тварин на пасовищах, контроль стану посівів, моніторинг ресурсів.
7. Інші: Застосування в будь-яких сферах, де потрібне відстеження об'єктів в реальному часі.

За функціональними можливостями:

1. Базовий трекінг: Відображення місцезнаходження об'єктів на карті в реальному часі.
2. Аналіз маршрутів: Збір та аналіз маршрутів пересування об'єктів, виявлення аномалій.
3. Візуалізація даних: Відображення даних про місцезнаходження, швидкість, напрямок руху та інші характеристики об'єктів у зручному форматі.
4. Геозонування: Створення віртуальних зон на карті та отримання оповіщень при вході/виході об'єктів з цих зон.
5. Оповіщення: Надсилання оповіщень про важливі події, такі як зміна маршруту, зупинка об'єкта, вихід з геозони.
6. Аналіз поведінки: Відстеження та аналіз поведінки об'єктів, виявлення підозрілих дій.

7. Інтеграція з іншими системами: Обмін даними з іншими програмними комплексами, такими як ERP, CRM, GIS.
8. Штучний інтелект: Використання машинного навчання та штучного інтелекту для аналізу даних, прогнозування маршрутів, виявлення аномалій.

За рівнем складності:

1. Прості системи: Базові системи трекінгу об'єктів пропонують обмежений набір функцій, наприклад, відображення місцезнаходження об'єктів на карті в реальному часі. Вони можуть використовувати лише одну технологію трекінгу, як-от GPS, та підходять для нескладних задач.
2. Складні системи: Складні системи трекінгу пропонують широкий набір функцій, включаючи аналіз маршрутів, візуалізацію даних, геозонування, оповіщення, аналіз поведінки, інтеграцію з іншими системами та використання штучного інтелекту. Вони можуть використовувати комбінацію різних технологій трекінгу для забезпечення більшої точності та функціональності.

Вибір системи трекінгу об'єктів залежить від конкретних потреб користувача. Для простих задач, таких як відстеження руху кур'єра по місту, може бути достатньо простої системи на базі GPS. Для складних задач, таких як контроль переміщення вантажів на великі відстані або аналіз поведінки людей на виробничому майданчику, потрібна складна система з широким набором функцій.

На ринку представлено безліч систем трекінгу об'єктів (СТО), які відрізняються за технологіями, функціональністю, сферою застосування та ціною. Ось кілька прикладів популярних СТО:

### GPS-трекери:

**GPSTOGO:** Широкий спектр GPS-трекерів для людей, транспортних засобів, тварин та вантажів. Пропонує базовий трекінг, аналіз маршрутів, геозонування та оповіщення.



Рис 2.1 GPSTOGO GPS tracker

**Traclogik:** GPS-трекери для транспортних засобів та вантажів. Пропонує детальний контроль маршрутів, витрати палива, поведінку водія та багато іншого.



Рис 2.2 Traclogik GPS tracker

Spot X:GPS-трекери для людей та активного відпочинку. Пропонує базовий трекінг, оповіщення про SOS, прогноз погоди та інші функції.



Рис 2.3 Spot X GPS tracker

RFID-системи:

Zebra Technologies:RFID-системи для логістики, виробництва та роздрібної торгівлі. Пропонує автоматичну ідентифікацію та відстеження товарів, контроль запасів та інші функції.



Рис 2.4 Zebra Technologies RFID system

Impinj:RFID-системи для різних сфер застосування, включаючи охорону, контроль доступу та IoT. Пропонує високу точність та надійність відстеження.



Рис 2.5 Impinj RFID system

Alien Technology:RFID-системи для складних задач, таких як відстеження контейнерів та вантажів. Пропонує широкий спектр RFID-міток та читачів.



Рис 2.6 Alien Technology RFID system

Axis Communications: Широкий спектр IP-камер для внутрішнього та зовнішнього використання. Пропонує високу якість зображення, аналітику відео та інтеграцію з системами трекінгу.



Рис 2.7 Axis Communications CCTV system

Hikvision: Доступні IP-камери та системи відеоспостереження для різних бюджетів. Пропонує широкі функції, такі як розпізнавання облич, аналіз руху та тепловізійні камери.



Рис 2.8 Hikvision CCTV system

Hanwha Techwin: IP-камери та системи відеоспостереження з акцентом на кібербезпеку. Пропонує шифрування даних, контроль доступу та інші функції захисту.



Рис 2.9 Hanwha Techwin CCTV system

Системи трекінгу на основі Wi-Fi:

IndoorAtlas: Система трекінгу в приміщеннях на основі Wi-Fi. Пропонує високу точність відстеження людей та активів в приміщеннях.

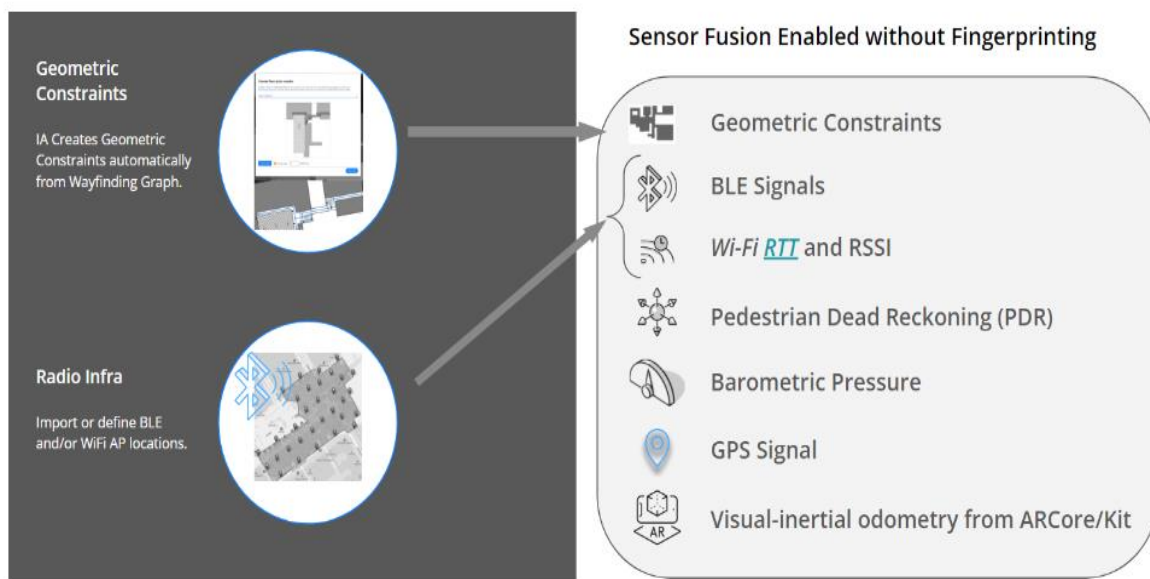


Рис 2.10 IndoorAtlas WiFi tracking system



Centilience: Система трекінгу в приміщеннях на основі Wi-Fi та Bluetooth. Пропонує відстеження людей, активів та транспортних засобів в приміщеннях та на відкритих просторах.

Here Indoor Services: Система трекінгу в приміщеннях на основі Wi-Fi, розроблена компанією HERE. Пропонує інтеграцію з картографічними сервісами HERE та інші функції.

Системи трекінгу на основі Bluetooth:

Tile: Bluetooth-трекери для відстеження ключів, гаманців та інших предметів. Пропонує базовий трекінг, оповіщення про втрату та інші функції.



Рис 2.11 Tile Bluetooth tracker

Chipolo: Bluetooth-трекери з широким спектром функцій, включаючи базовий трекінг, оповіщення про втрату, селфі-камеру та багато іншого.

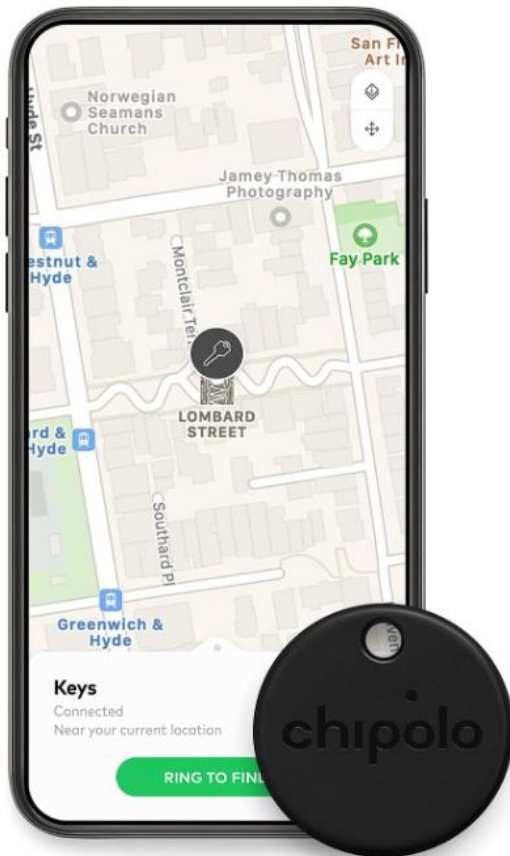


Рис 2.12 Chipolo Bluetooth tracker

RadBeacon: Bluetooth-маяки для відстеження людей та активів в приміщеннях. Пропонує високу точність відстеження та інтеграцію з системами автоматизації будівель

### **2.3 Порівняльний аналіз існуючих систем трекінгу об'єктів за функціональністю**

Порівняймо існуючі систем трекінгу об'єктів (СТО) на основі таких критеріїв:

- Тип об'єктів, що відстежуються.
- Сфера застосування.
- Технологія трекінгу.

Таблиця 2.1

Порівняймо систем трекінгу на базі технології трекінгу.

Система	Тип об'єктів	Сфера застосування	Технологія трекінгу
<b>GPSGO</b>	Люди, транспортні засоби, тварини, вантажі	Логістика, транспорт, охорона, безпека, виробництво, сільське господарство	GPS
<b>Traclogik</b>	Транспортні засоби, вантажі	Логістика, транспорт, охорона, безпека	GPS
<b>Spot X</b>	Люди, активний відпочинок	Туризм, альпінізм, велоспорт, водні види спорту	GPS
<b>Zebra Technologies</b>	Товари, активи	Логістика, виробництво, роздрібна торгівля	RFID
<b>Impinj</b>	Товари, активи, люди	Логістика, виробництво, охорона, контроль доступу, IoT	RFID
<b>Alien Technology</b>	Контейнери, вантажі	Логістика, транспорт, морські перевезення	RFID
<b>Axis Communications</b>	Люди, транспортні засоби, активи	Охорона, безпека, контроль доступу, роздрібна торгівля, виробництво	Відеокамери
<b>Hikvision</b>	Люди, транспортні засоби, активи	Охорона, безпека, контроль доступу, роздрібна торгівля, виробництво	Відеокамери
<b>Hanwha Techwin</b>	Люди, транспортні засоби, активи	Охорона, безпека, контроль доступу, роздрібна торгівля, виробництво	Відеокамери
<b>IndoorAtlas</b>	Люди, активи	Охорона, безпека, контроль доступу, роздрібна торгівля, виробництво, музеї, лікарні	Wi-Fi
<b>Centilience</b>	Люди, активи, транспортні засоби	Охорона, безпека, контроль доступу, роздрібна торгівля, виробництво, склади	Wi-Fi, Bluetooth
<b>HERE Indoor Services</b>	Люди, активи	Охорона, безпека, контроль доступу, роздрібна торгівля, виробництво, аеропорти, вокзали	Wi-Fi
<b>Tile</b>	Ключі, гаманці, інші предмети	Особисте використання	Bluetooth
<b>Chipolo</b>	Ключі, гаманці, інші предмети	Особисте використання	Bluetooth
<b>RadBeacon</b>	Люди, активи	Охорона, безпека, контроль доступу, виробництво, склади	Bluetooth

Таблиця 2.1

Порівняймо систем трекінгу на базі технології сферизастосування.

Система	Технологія	Сфера застосування	Функціональні можливості	Переваги	Недоліки
<b>GPS-трекери</b>	GPS	Логістика, транспорт, охорона, безпека	Базовий трекінг, аналіз маршрутів, візуалізація даних, геозонування, оповіщення	Точність, надійність, доступність	Обмежена дальність дії, залежність від супутникових сигналів
<b>RFID-системи</b>	RFID	Логістика, виробництво, складське господарство	Базовий трекінг, ідентифікація об'єктів, контроль доступу	Швидкість, точність, відстеження великої кількості об'єктів	Висока вартість, необхідність RFID-міток
<b>Системи відеоспостереження</b>	Відеокамери	Охорона, безпека, контроль доступу	Візуальне відстеження, аналіз поведінки, розпізнавання облич	Детальна інформація, можливість запису відео	Висока вартість, складність монтажу
<b>Системи трекінгу на основі Wi-Fi</b>	Wi-Fi	Внутрішнє відстеження, IoT-пристрої	Базовий трекінг, локалізація в приміщеннях	Точність, доступність	Обмежена дальність дії, залежність від Wi-Fi
<b>Системи трекінгу на основі Bluetooth</b>	Bluetooth	Відстеження на коротких відстанях, IoT-пристрої	Базовий трекінг, локалізація в приміщеннях	Низька вартість, простота використання	Обмежена дальність дії

## 2.4. Аналіз протоколів та технологій, що використовуються в системах трекінгу об'єктів

Проаналізуємо протоколи та технології, що використовуються в системах трекінгу об'єктів (СТО).

### Протоколи

СТО використовують різні протоколи для передачі даних про місцезнаходження об'єктів. Деякі з найпоширеніших протоколів:

**GPS:** GPS (Global Positioning System) - це супутникова система навігації, яка використовується для визначення місцезнаходження об'єктів на Землі. GPS-трекери передають дані про своє місцезнаходження на сервери за допомогою протоколу NMEA (National Marine Electronics Association).

**GSM/GPRS:** GSM (Global System for Mobile Communications) та GPRS (General Packet Radio Service) - це мобільні мережі, які використовуються для передачі даних про місцезнаходження об'єктів. GSM/GPRS-трекери передають дані про своє місцезнаходження на сервери за допомогою протоколу GPRS.

**LoRaWAN:** LoRaWAN (Long Range Wide Area Network) - це мережа низькоенергетичного широкомасштабного зв'язку, яка використовується для передачі даних про місцезнаходження об'єктів. LoRaWAN-трекери передають дані про своє місцезнаходження на сервери за допомогою протоколу LoRaWAN.

**Wi-Fi:** Wi-Fi (Wireless Fidelity) - це технологія бездротової локальної мережі, яка використовується для передачі даних про місцезнаходження об'єктів. Wi-Fi-трекери передають дані про своє місцезнаходження на сервери за допомогою протоколу 802.11.

**Bluetooth:** Bluetooth - це технологія бездротового зв'язку ближнього радіуса дії, яка використовується для передачі даних про місцезнаходження об'єктів. Bluetooth-трекери передають дані про своє місцезнаходження на сервери за допомогою протоколу Bluetooth.

### Технології

СТО використовують різні технології для визначення місцезнаходження об'єктів. Деякі з найпоширеніших технологій:

**GPS:** GPS-трекери використовують супутникові сигнали для визначення місцезнаходження об'єктів.

**Cell ID:** GSM/GPRS-трекери використовують ідентифікатори комірок (Cell ID) мережі мобільного зв'язку для визначення місцезнаходження об'єктів.

**Wi-Fi:** Wi-Fi-трекери використовують сигнали Wi-Fi для визначення місцезнаходження об'єктів за допомогою технології Wi-Fi Fingerprinting або Wi-Fi RTT (Round Trip Time).

**Bluetooth:** Bluetooth-трекери використовують сигнали Bluetooth для визначення місцезнаходження об'єктів за допомогою технології iBeacon або Eddystone.

**RFID:** RFID-трекери використовують радіочастотні сигнали для визначення місцезнаходження об'єктів.

#### *Вибір протоколу та технології*

Вибір протоколу та технології для СТО залежить від таких факторів: Тип об'єктів, що відстежуються, Сфера застосування, Необхідна точність, Радіус дії, Вартість, Енергоспоживання

Існує багато різних протоколів і технологій, що використовуються в системах трекінгу об'єктів. Найкращий протокол та технологія для вас залежатимуть від ваших конкретних потреб. При виборі протоколу та технології слід врахувати всі вищезазначені фактори.

## **2.5 Оцінка ефективності існуючих систем трекінгу об'єктів**

Оцінка ефективності існуючих систем трекінгу об'єктів є важливим етапом у дослідженні та впровадженні таких систем. Цей розділ спрямований на аналіз різних показників та метрик, які дозволяють зрозуміти якість та результативність використання систем трекінгу. Нижче наведено деякі ключові аспекти, що розглядаються під час оцінки ефективності таких систем:

**Точність:** Визначення точності системи трекінгу об'єктів є однією з найважливіших метрик. Це оцінюється порівнянням фактичного місцезнаходження об'єкта з даними, отриманими від системи трекінгу.

**Часова продуктивність:** Час, необхідний для виявлення та відстеження об'єкта, також є важливим фактором. Швидкість оновлення даних та реакція системи на зміни місцезнаходження може впливати на загальну ефективність.

**Надійність:** Надійність системи оцінюється з точки зору стабільності та безперебійності її роботи. Вона враховує можливість виникнення помилок та відмов, а також можливість відновлення після збоїв.

**Вартість:** Оцінка ефективності також включає в себе аналіз вартості впровадження та експлуатації системи трекінгу об'єктів. Це може включати як витрати на обладнання та програмне забезпечення, так і витрати на підтримку та обслуговування.

**Масштабованість:** Здатність системи трекінгу об'єктів до масштабування є ще однією важливою метрикою. Це включає в себе можливість використання системи в різних масштабах, від невеликих підприємств до великих корпорацій.

**Сумісність:** Оцінка сумісності системи з іншими технологіями та платформами також є важливим аспектом. Забезпечення інтеграції з існуючими системами та можливість взаємодії з ними може покращити загальну ефективність

та використаність системи трекінгу об'єктів.

Управління енергоспоживанням: Оцінка ефективності також включає аналіз споживання енергії системою трекінгу об'єктів. Енергоефективність є важливим аспектом, особливо для пристроїв, що працюють на батареях або в умовах обмеженого живлення.

Тривалість служби та обслуговування: Оцінка ефективності включає аналіз тривалості служби та потреби в обслуговуванні системи трекінгу об'єктів. Це важливо для визначення загальних витрат на експлуатацію системи протягом тривалого періоду часу.

Підтримка функціональності: Важливим аспектом є також оцінка можливостей системи трекінгу об'єктів у виконанні необхідних функцій та завдань. Наявність потрібних функцій, а також їх якість та додаткові можливості, визначає рівень ефективності системи.

Задоволення потреб користувачів: Кінцеві користувачі виставляють вимоги до систем трекінгу об'єктів. Оцінка ефективності також включає аналіз задоволеності користувачів з роботи системи, їхніх відгуків та пропозицій щодо покращення.

Безпека та конфіденційність даних: Оцінка ефективності системи трекінгу об'єктів також має враховувати рівень безпеки та захищеності від несанкціонованого доступу до даних, а також конфіденційність зібраних даних.

Скальованість та гнучкість: Важливим аспектом ефективності є можливість системи трекінгу об'єктів адаптуватися до змін у потребах та обсязі даних, а також її здатність до масштабування у відповідь на зростання обсягу даних чи кількості об'єктів, що відстежуються.

Інтеграція та сумісність з іншими системами: Ефективність системи трекінгу об'єктів може також визначатися її здатністю інтегруватися з іншими існуючими системами та платформами. Це може включати здатність обміну даними та взаємодії з системами управління складами, системами логістики, системами управління виробництвом та іншими.

Легкість використання та інтерфейс користувача: Часто ефективність системи трекінгу об'єктів визначається його зручністю та простотою використання для кінцевих користувачів. Інтуїтивно зрозумілий інтерфейс користувача та легка навігація можуть значно покращити прийняття та ефективне використання системи.

Стабільність та відновлення: Ефективна система трекінгу об'єктів має бути стійкою до відмов та забезпечувати можливість швидкого відновлення після збоїв або втрати зв'язку. Здатність системи швидко відновлюватися та продовжувати надавати послуги є важливою для забезпечення неперервного відстеження об'єктів. Забезпечення відповідності правовим та регуляторним вимогам: Оцінка

ефективності також включає аналіз того, наскільки система трекінгу об'єктів відповідає вимогам законодавства та регуляторних стандартів, таких як правила збереження даних, захист конфіденційності та інші регуляції.

**GPSGO:** Ця система пропонує широкий спектр функцій трекінгу, включаючи базовий трекінг, аналіз маршрутів, геозонування та оповіщення.

**Traclogik:** Traclogik пропонує подібний набір функцій до GPSGO, але також включає детальний контроль маршрутів та аналіз поведінки.

**Here Tracking:** Here Tracking фокусується на високоточній локації всередині приміщень та інтеграції з іншими системами.

Таблиця 2.3

#### Порівняння існуючих систем трекінгу об'єктів

<b>Критерій</b>	<b>GPSGO</b>	<b>Traclogik</b>	<b>Here Tracking</b>
<b>Точність</b>	Середня (GPS)	Середня (GPS)	Висока (різні технології)
<b>Надійність</b>	Висока	Висока	Висока
<b>Масштабованість</b>	Висока	Висока	Середня
<b>Простота використання</b>	Середня	Середня	Складна
<b>Вартість</b>	Низька - середня	Середня - висока	Висока

Враховання цих аспектів у процесі оцінки дозволяє здійснити комплексний підхід до визначення ефективності системи трекінгу об'єктів. Це допомагає забезпечити вибір та впровадження оптимального рішення, яке задовольняє потреби та очікування користувачів і відповідає вимогам сучасного бізнес-середовища.



## 3 ВИБІР ПРОГРАМНИХ ТА АПАРАТНИХ ЗАСОБІВ ДЛЯ ПОБУДОВИ ІНФОРМАЦІЙНЕ

### 3.1 Вибір апаратних засобів для інформаційної системи трекінгу об'єктів

Обрання Arduino для побудови інформаційної системи є виправданим з кількох причин. По-перше, Arduino відомий своєю доступною вартістю, що робить його привабливим вибором для проектів з обмеженим бюджетом. По-друге, мікроконтролер Arduino має відкриту архітектуру, що дає можливість розробникам вносити зміни в програмне забезпечення та пристрої самостійно. Це дозволяє адаптувати систему до конкретних потреб користувача. Крім того, Arduino підтримує різноманітні датчики та модулі, що розширює можливості системи трекінгу об'єктів. Також важливою перевагою є широке співробітництво та активна спільнота розробників Arduino, що забезпечує швидку відповідь на питання та підтримку у вирішенні будь-яких проблем, які можуть виникнути під час розробки. Порти введення-виведення мікроконтролерів представлені у вигляді штирьових лінійок та живляться від 5 В. Отже, порти мають той самий діапазон допустимих вхідних та вихідних напруг.

Arduino Uno

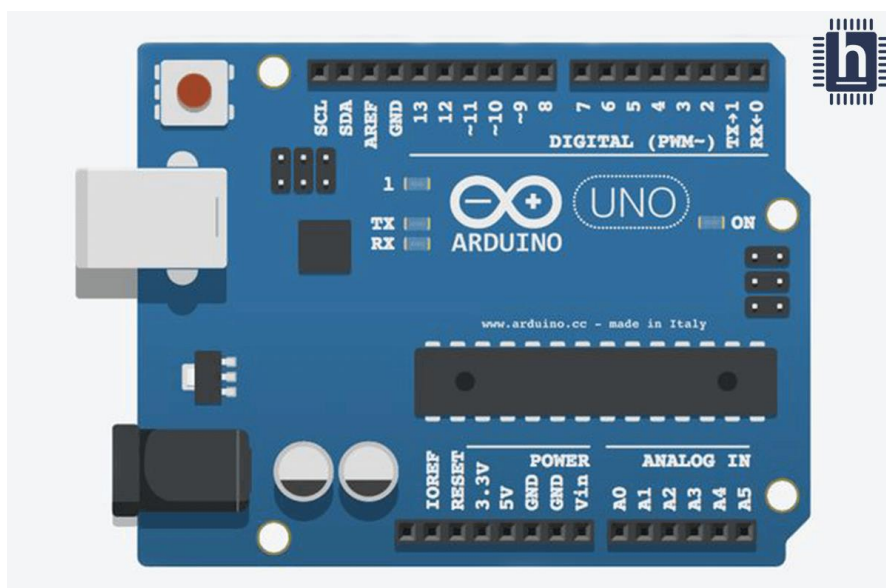


Рис 3.1 Arduino UNO

Arduino UNO - це одна з найпопулярніших та поширених моделей мікроконтролерів в лінійці Arduino. Вона базується на мікроконтролері

ATmega328P від Microchip (раніше - Atmel). Arduino UNO має простий у використанні інтерфейс, що робить його ідеальним вибором як для початківців, так і для досвідчених розробників.

*Характеристики Arduino UNO:*

1. Процесор: Мікроконтролер ATmega328P з тактовою частотою 16 МГц.
2. Пам'ять: 32 КБ флеш-пам'яті для зберігання програмного коду, 2 КБ оперативної пам'яті (SRAM) і 1 КБ EEPROM.
3. Інтерфейси: 14 цифрових входів/виходів (з яких 6 можуть бути використані як PWM виходи), 6 аналогових входів, UART (Serial), I2C та SPI.
4. Живлення: Може живитися від USB або від зовнішнього джерела напруги від 7 до 12 вольт.
5. Розмір та форм фактор: Arduino UNO має компактний розмір 68.6 мм на 53.4 мм, що робить його зручним для використання в різних проектах.

Arduino UNO має широкий спектр застосувань, включаючи автоматизацію, IoT, робототехніку, сенсорні системи та багато інших. Його простота використання разом з великою кількістю доступних бібліотек і документацією робить його досить популярним серед розробників у всьому світі.

## ESP32-S2-WROOM



Рис 3.2 ESP32\_S2

ESP32-S2-WROOM - це модуль Wi-Fi з мікроконтролером Xtensa® LX7 32-bit, що належить до сімейства ESP32 від Espressif Systems. Він є потужним та

універсальним рішенням для різноманітних застосувань в Інтернеті речей (ІоТ), таких як:

- Вбудовані системи
- "Розумний дім"
- Носимі пристрої
- Системи промислової автоматизації

Характеристики:

- Процесор: Xtensa® LX7 32-bit CPU з частотою до 240 МГц
- Пам'ять:
  - Оперативна пам'ять (RAM): 320 Кб
  - Flash-пам'ять: 4 Мб SPI
  - Підтримка додаткової зовнішньої пам'яті
- Інтерфейси:
  - Wi-Fi 4 (IEEE 802.11 b/g/n)
  - GPIO (загального призначення вхід/вихід)
  - SPI ( послідовний периферійний інтерфейс)
  - I2C (двохдротовий послідовний інтерфейс)
  - I2S (інтерфейс для передачі звуку)
  - UART (універсальний асинхронний прийомопередавач)
  - LCD (інтерфейс для рідкокристалічного дисплея)
  - Камера (інтерфейс для підключення камери)
  - ADC (аналого-цифровий перетворювач)
  - DAC (цифро-аналоговий перетворювач)
  - Датчик дотику
  - Датчик температури
  - USB OTG 1.1 (для підключення периферійних пристроїв)
- Напруга живлення: 3.0 ~ 3.6 В
- Робоча температура: -40 ~ 85 °С
- Безпека: шифрування WPA/WPA2/WPA3
- Компактний розмір

Переваги:

- Потужний процесор та достатня кількість пам'яті для виконання складних задач
- Багатий набір інтерфейсів для підключення різноманітних датчиків та компонентів

- Підтримка Wi-Fi забезпечує бездротове з'єднання з мережею інтернет
- Низьке споживання енергії, що робить його придатним для автономних пристроїв
- Компактний розмір дозволяє використовувати його в мініатюрних пристроях

neo-7m

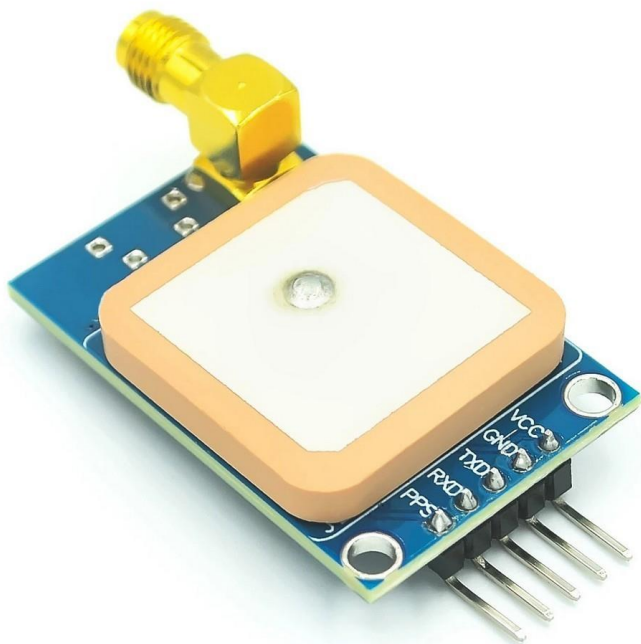


Рис 3.3 Neo-7m

Модуль GPS NEO-7M - це мульти-GNSS приймач (підтримує декілька поширених стандартів супутникової навігації), що ідеально підходить для портативних та автономних систем, яким потрібна інформація про місцезнаходження та точний час з малим споживанням енергії. Він підтримує такі супутникові системи:

- GPS (L1 C/A)
- GLONASS (L1 FDMA)
- BeiDou (B1I) (опціонально, залежно від моделі)
- Galileo (E1) (опціонально, залежно від моделі)
- QZSS (L1 C/A) (опціонально, залежно від моделі)

Характеристики:

- Кількість каналів: 56 каналів (7-го покоління від u-blox)

- Точність позиціонування:
  - GPS: до 2.5 метрів
  - GLONASS: до 4 метрів
- Частота оновлення даних: до 10 Гц
- Інтерфейс: TTL (UART)
- Напруга живлення: 3.3 В
- Споживання струму: мінімальне - 50 мА, максимальне - 220 мА
- Робоча температура: -40°C ~ +85°C
- Розміри: 23.6 x 21.6 x 6.1 мм

Особливості:

- Швидкий холодний старт
- Підтримка SBAS: WAAS, EGNOS, MSAS
- Підтримка A-GPS (Assisted GPS)
- Вихід даних NMEA
- Компактний розмір та низьке споживання енергії

### **3.2 Вибір програмного забезпечення для інформаційної системи трекінгу об'єктів**

Програмування Arduino здійснюється за допомогою власного програмного середовища (IDE), яке знаходиться у вільному доступі на сайті Arduino і працює на всіх доступних платформах (Windows, Mac OS і Linux). До складу оболонки входять текстовий редактор, менеджер проектів, препроцесор, компілятор та інструменти для завантаження програми в мікроконтролер; програма для Arduino написана простою і зрозумілою мовою C++. Вона підтримується простими, зрозумілими функціями для управління вводом/виводом виводів.

C++ - це об'єктно-орієнтована мова програмування, яка пропонує модульність, роздільну компіляцію, обробку виключень та абстрагування даних. Це також одна з найпоширеніших мов програмування. Використовується не тільки для розробки програмного забезпечення, але й для розробки драйверів для різних пристроїв.

Плюси використання програмування Arduino:

1. Простота використання: Інтегроване середовище розробки (IDE) Arduino доступне безкоштовно на всіх основних операційних системах і

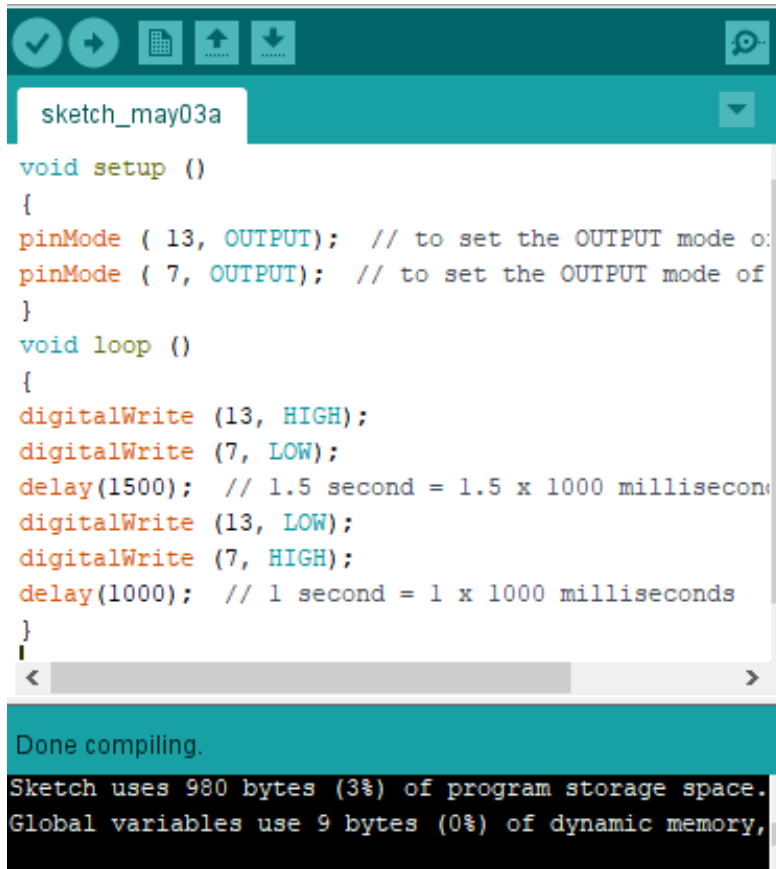
забезпечує зручний інтерфейс для програмування.

2. Зручне завантаження програм: Програми можна легко завантажувати в Arduino за допомогою одного кліка, що спрощує роботу з мікроконтролером.
3. Комплексність оболонки: Інтегроване середовище розробки містить текстовий редактор, менеджер проектів, препроцесор, компілятор і інші інструменти, що полегшують розробку програм для мікроконтролерів.
4. Використання мови програмування C++: Arduino використовує звичайний C++, що дозволяє використовувати розширені функції для керування введенням/виведенням на контактах.

Мінуси використання програмування Arduino:

1. Обмежена функціональність IDE: Інтегроване середовище розробки Arduino може бути обмеженим для складних або великих проектів, оскільки не завжди забезпечує всі необхідні функції.
2. Складність масштабування: Для деяких складних проектів або проектів, що потребують великої кількості обчислень, Arduino може бути не досить потужним рішенням.
3. Не всі функції C++ доступні: Деякі розширені функції мови програмування C++ можуть бути недоступні або складні у використанні на платформі Arduino.

4. Обмежені можливості для розробки додатків: Arduino може бути не найкращим вибором для розробки додатків, що вимагають високої швидкості або обробки великих обсягів даних.



```
void setup ()
{
  pinMode ( 13, OUTPUT); // to set the OUTPUT mode of
  pinMode ( 7, OUTPUT); // to set the OUTPUT mode of
}
void loop ()
{
  digitalWrite (13, HIGH);
  digitalWrite (7, LOW);
  delay(1500); // 1.5 second = 1.5 x 1000 milliseconds
  digitalWrite (13, LOW);
  digitalWrite (7, HIGH);
  delay(1000); // 1 second = 1 x 1000 milliseconds
}

```

Done compiling.

Sketch uses 980 bytes (3%) of program storage space.  
Global variables use 9 bytes (0%) of dynamic memory,

Рис 3.4 Arduino IDE

API (Application Programming Interface) - це набір правил та специфікацій, які визначають, як різні програми та пристрої можуть взаємодіяти та обмінюватися даними. REST API (Representational State Transfer) - це тип API, який ґрунтується на принципах REST (Representational State Transfer) - архітектурного стилю передачі стану ресурсів. REST API часто називають RESTful API.

REST API використовує HTTP-запити для виконання CRUD-операцій (Create, Read, Update, Delete) з ресурсами. Наприклад:

- GET-запит: використовується для отримання даних з ресурсу.
- POST-запит: використовується для створення нового запису в ресурсі.
- PUT-запит: використовується для оновлення існуючого запису в ресурсі.
- DELETE-запит: використовується для видалення запису з ресурсу.

Всі методи HTTP можна використовувати у REST API.

Дані, що передаються через REST API, можуть бути представлені у різноманітних форматах, таких як JSON (JavaScript Object Notation), HTML, XML, Python, PHP, або звичайний текст. JSON є популярним форматом, оскільки він легко читається як людьми, так і машинами, а також не залежить від мови програмування.

Заголовки та параметри запитів відіграють важливу роль у REST API, адже вони містять важливу інформацію про запит, таку як:

- Метадані: надають додаткову інформацію про запит.
- Авторизація: використовується для автентифікації та авторизації користувачів.
- URI (Uniform Resource Identifier): унікальний ідентифікатор ресурсу.
- Кешування: використовується для покращення продуктивності за допомогою кешування даних.
- Файли cookie: зберігають інформацію про стан сеансу користувача.

Переваги REST API:

- Простота використання: REST API ґрунтується на простих та зрозумілих принципах, що робить його легким для розробки та використання.
- Гнучкість: REST API може використовуватися для різноманітних задач, що робить його універсальним рішенням.



- Масштабованість: REST API може масштабуватися для обслуговування великої кількості користувачів та запитів.
- Надійність: REST API є надійним та стійким до збоїв.
- Безпека: REST API може бути захищений за допомогою протоколів HTTPS та OAuth.

REST API - це потужний та гнучкий інструмент для розробки веб-сервісів та API. Його простота використання, гнучкість, масштабованість, надійність та безпека роблять його чудовим вибором для розробки систем, які потребують надійної та ефективної взаємодії між трекерами та сервером.

Заголовки запитів та відповідей, а також стандартні коди статусу HTTP є невід'ємною частиною правильно розроблених REST API.

Щоб API вважалось RESTful, воно має відповідати наступним принципам:

Архітектура клієнт-сервер:

- Система складається з клієнтів, серверів та ресурсів.
- Запити до ресурсів керуються через протокол HTTP.

Безстанний зв'язок клієнт-сервер:

- Сервер не зберігає інформацію про стан клієнта між запитами.
- Кожен запит є незалежним та самодостатнім.

Кешування даних:

- Дані кешуються для покращення продуктивності та зменшення навантаження на сервер.

Єдиний інтерфейс:

- Всі компоненти системи взаємодіють через єдиний, стандартизований інтерфейс.
- Це гарантує узгодженість та простоту використання.

Принципи RESTful інтерфейсу:

- Ідентифікація ресурсів: Запитувані ресурси мають унікальні ідентифікатори, які відокремлені від їх представлень.

- Маніпулювання ресурсами: Клієнти можуть змінювати стан ресурсів за допомогою представлень, що отримують від сервера.
- Самоописні повідомлення: Повідомлення, які сервер повертає клієнту, містять всю необхідну інформацію для їх правильної обробки.
- Доступність гіпертексту/гіпермедіа: Клієнти можуть використовувати гіперпосилання, щоб знаходити нові ресурси та дії, доступні їм.

Багатошарова система:

- Система може складатися з декількох рівнів серверів (наприклад, для безпеки, балансування навантаження тощо).
- Клієнт не має знати про внутрішню структуру системи.
- Запити до інформації надсилаються ієрархічно, приховуючи деталі реалізації.

JSON (JavaScript Object Notation) - це текстовий формат даних, який відповідає синтаксису об'єктів JavaScript. Він є легким для читання та розуміння як людьми, так і машинами. JSON широко використовується для обміну даними між веб-серверами та клієнтами.

Переваги використання JSON:

- Простота: JSON має просту та зрозумілу структуру, що робить його легким для читання, написання та обробки.
- Гнучкість: JSON може використовуватися для представлення різних типів даних, таких як масиви, об'єкти, числа, рядки та булеві значення.
- Незалежність від мови: JSON не залежить від JavaScript, і його можна використовувати з будь-якою мовою програмування, яка має бібліотеку JSON.
- Широка підтримка: JSON підтримується практично всіма мовами програмування та веб-фреймворками.

Вибір технології сервера:

Node.js та PHP - це дві популярні технології, які добре підходять для розробки REST API з підтримкою JSON.

Node.js - це середовище виконання JavaScript з відкритим вихідним кодом та міжплатформним, що дозволяє виконувати JavaScript-код поза браузером, на сервері. Однією з ключових особливостей Node.js є використання двигуна JavaScript V8,

який також використовується в браузері Google Chrome, що робить Node.js дуже продуктивним.

Node.js дозволяє виконувати програми в одному процесі, що не створює новий потік для кожного запиту. Воно надає в своїй стандартній бібліотеці набір асинхронних примітивів вводу-виводу, які допомагають уникнути блокування JavaScript-коду. Більшість бібліотек в Node.js також працюють в асинхронному режимі, що робить взаємодію з файловою системою, мережею та іншими ресурсами ефективнішою і продуктивнішою.

Node.js дозволяє розробникам створювати ефективні та масштабовані застосунки для серверів, мереж та інших веб-проектів. Вона є популярним інструментом для створення веб-серверів, API, мережеских програм, чат-ботів, розширень для браузерів та інших веб-додатків. Благодаря широким можливостям та великій кількості розширень (модулів), Node.js дозволяє швидко розробляти та впроваджувати різноманітні веб-проекти.

Крім того, Node.js є популярним серед розробників через його активну спільноту, багатий екосистему і велику кількість інструментів, що спрощують розробку та підтримку програм. Він підтримується багатьма відомими компаніями та використовується для створення високонавантажених систем, які обробляють тисячі одночасних запитів.

PHP (Hypertext Preprocessor) - це скриптова мова програмування загального призначення, яка використовується для розробки динамічних веб-сайтів та веб-додатків. Вона виконується на сервері, а не на комп'ютері користувача, і генерує HTML-код, який відображається в браузері. PHP дозволяє створювати веб-сторінки, які можуть взаємодіяти з користувачем, обробляти форми, отримувати та зберігати дані у базі даних, створювати сесії для зберігання інформації про користувача та багато іншого. Він є однією з найпопулярніших мов програмування для веб-розробки завдяки своїй простоті в освоєнні та потужним функціональним можливостям.

Вибір був зроблений в користь Node.js через його використання JavaScript, простоту та швидкість, а також через його зручність для створення REST API. Щодо бази даних, вирішено використовувати MongoDB, яка є системою керування базами даних NoSQL з відкритим вихідним кодом. NoSQL використовується як альтернатива традиційним реляційним базам даних і особливо корисний для обробки великих обсягів розподілених даних. MongoDB спрощує керування документ-орієнтованою інформацією та забезпечує зручний доступ до даних.

MongoDB використовує записи у формі документів, що містять структуру даних у вигляді пар полів і значень. Документи є основною одиницею даних у MongoDB і подібні до об'єктів JavaScript, але використовують формат Binary JSON (BSON). Використання BSON дозволяє вміщувати більш різноманітні типи даних. Поля у цих документах подібні до стовпців у реляційних базах даних і можуть містити значення різних типів, включаючи інші документи, масиви та масиви документів, згідно з документацією користувача MongoDB. Документи також мають первинний ключ як унікальний ідентифікатор.

Набір документів у MongoDB називається колекцією, яка працює як еквівалент таблиць у реляційних базах даних. Колекції можуть містити будь-які типи даних, але є обмеження, яке полягає в тому, що дані в колекціях не можуть бути розподілені між різними базами даних.

MongoDB, як і будь-яка інша система управління базами даних, має свої переваги і обмеження. Однією з його переваг є гнучкість у роботі з даними через документи, що використовуються для збереження інформації. Вони дозволяють зберігати дані у вигляді об'єктів, що спрощує розробку та підтримку програмного забезпечення.

Крім того, MongoDB є масштабованим рішенням, яке дозволяє легко розширювати бази даних з великим обсягом даних або великою кількістю запитів. Ви можете налаштувати реплікацію та шардування для забезпечення високої доступності та продуктивності вашої системи.

### **3.3 План роботи систем трекінгу об'єктів**

Датчики, встановлені на вантажах, транспортних контейнерах, транспортних засобах або суднах, використовують телематику та GPS для збору даних про місцезнаходження під час подорожі. Ці дані надсилаються в хмару через мережу Wi-Fi. Коли ці дані знаходяться в хмарі або на сервері, уповноважені сторони можуть отримати доступ до них з будь-якого місця та в будь-який час.

Відмінною особливістю Інтернету речей в логістиці є те, що він надає зображення місця розташування, стану і обстановки відправляються товарів практично в режимі реального часу. Наявність доступу до цих даних робить ланцюжок поставок повністю прозорим. Вам не потрібно чекати, поки вантаж прибуде на контрольний пункт, щоб перевірити його стан. В

результаті, оскільки час відгуку значно скорочується, постачальники можуть передбачати, виправляти, а іноді і запобігати проблемам до їх ескалації. Крім того, Жовтнева прозорість означає меншу ймовірність порушення ланцюжка поставок.

Нарешті, аналіз даних ІОТ може бути дуже корисним для оптимізації маршрутів доставки, підвищення безпеки управління автопарком, оцінки рівня продуктивності операторів, аналізу вуглецевого сліду логістики та мінімізації пов'язаних з цим бізнес-ризиків.

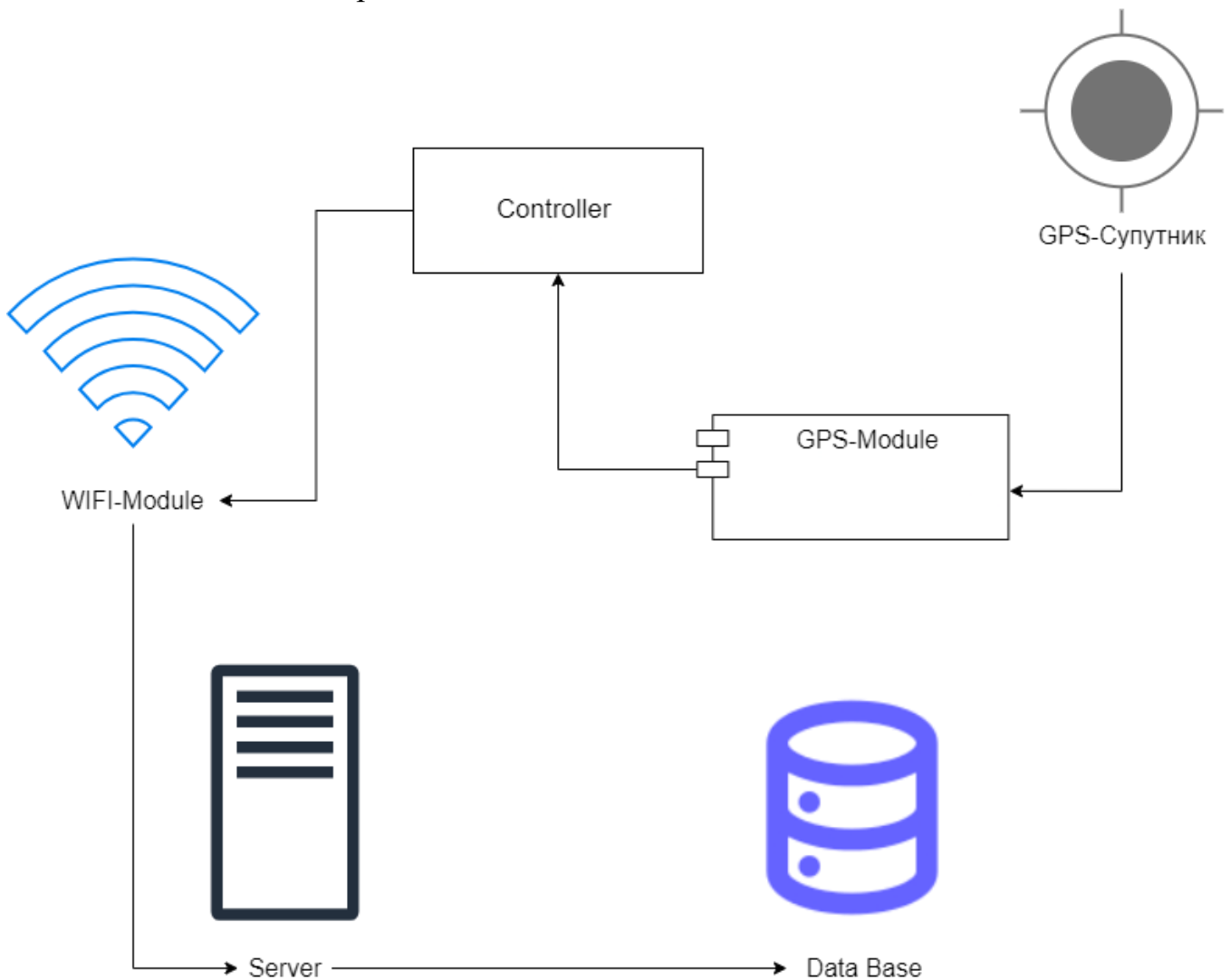


Рис 3.5 План роботи системи

Використання Інтернету речей також відкриває багато можливостей в інших сферах, таких як експлуатація терміналів, безпека на транспорті, моніторинг та маршрутизація автопарку, прогнозне обслуговування та управління терміном

служби продукту. Оскільки організації переходять від системи на основі ручних контрольних точок до безперервного потоку даних з підтримкою Інтернету речей, ланцюжок поставок практично миттєво стає ефективним і прозорим. Але цей перехід нелегкий

### 3.4 Розробка макету GPS-трекера

Розробка макета почалася зі створення діаграм. Для схеми були обрані наступні компоненти:

- Arduino Uno-виконайте всі обчислення та надайте команди елементам макета, щоб запрограмувати весь макет.
- Модуль Wi-Fi ESP8266NodeMCU V3-для відправки даних з модуля GPS на сервер.
- GPS-модуль GPS NEO - бМ-отримати геолокацію.
- З'єднувальний кабель - для підключення всіх елементів схеми

Алгоритм нормальної роботи схеми:

1. Модуль GPS підключається до супутника, отримує координати, час і дату
2. Модуль GPS надішле дані на Arduino Uno
3. Arduino аналізує дані, що надходять з модуля GPS, створює форму, необхідну для передачі, і відправляє їх по цифрових каналах в модуль Wi-Fi
4. Модуль Wi-Fi створює та надсилає запит на сервер

### 3.5 Розробка серверу для моделі

Як описано вище, сервер створюється за допомогою вузла.js та MongoDB як база даних. Він повинен мати деякі основні характеристики, а саме.:

Якщо ви зробите запит GET без вказівки ідентифікатора пристрою, прочитайте всі дані пристрою в базі даних і передайте їх клієнту

Якщо ви хочете отримати його з вказаним ідентифікатором пристрою, прочитайте дані цього пристрою з бази даних і передайте їх клієнту

Приймає і записує в базу даних поштові запити з координатами і Жовтневої

додатковою інформацією

Якщо ви хочете видалити запит із зазначеним ідентифікатором пристрою, видаліть усі дані в базі даних для цього пристрою.

Платформа Express для розробки спрощує створення та налаштування бібліотеки mongoose, що полегшує взаємодію з базою даних, а також із серверами. Почнемо з створення серверу та підключенням його до БД.

```
const express = require('express');
const mongoose = require('mongoose');
const app = express();

mongoose.connect('mongodb://localhost/gps_arduino_db', {
  useNewUrlParser: true,
}).catch(err => {
  console.log(`Database connection error: ${err}`);
  process.exit(1);
});

const db = mongoose.connection;

db.once('open', () => {
  console.log('Connected to Database');
});

db.on('error', err => {
  console.log(`Database error: ${err}`);
  process.exit(1);
});

app.listen(6060, () => {
  console.log('Server started on port 6060');
});
```

Рис 5.5 - Створення інстансу сервера та підключення його до БД

Далі необхідно визначити заголовки запитів які сервер буде приймати.

```
app.use((req, res, next) => {
  res.append('Access-Control-Allow-Origin', '*');
  res.append('Access-Control-Allow-Methods', 'GET, PUT, POST, DELETE');
  res.append('Access-Control-Allow-Headers', 'Content-Type');
  if (req.headers.hasOwnProperty('withCredentials')) {
    res.append('Access-Control-Allow-Credentials', 'true');
  }
  next();
});
```

Рис 5.6 - Визначення заголовків запитів

Важливою частиною являється обробки всіх помилок які можуть трапитися та відображення їх клієнту

```
app.use((req, res, next) => {
  const error = new Error('Error');
  error.status = 101;
  next(error);
});

// Обробка всіх помилок у програмі
app.use((error, req, res, next) => {
  res.status(error.status || 500);

  res.json({
    error: {
      message: error.message,
    },
  });
});

app.listen(PORT, () => {
  console.log('Server started');
});
```

Рис5.7 - Обробки помилок



Для взаємодії з БД визначаємо схему за якою повинні записуватися дані, та колекцію в чку вони будуть записані.

```
const mongoose = require('mongoose');

const gpsCoordsModelSchema = new mongoose.Schema({
  lat: {
    type: Number,
    required: true,
  },
  lon: {
    type: Number,
  },
});

const gpsCoordsModel = mongoose.model('gpsCoords', gpsCoordsModelSchema);

const gpsTrackerDataModelSchema = new mongoose.Schema(
  {
    deviceId: {
      type: String,
      required: true,
    },
    date: {
      type: String,
      required: true,
    },
    time: {
      type: String,
      required: true,
    },
    coords: {
      type: mongoose.Schema.Types.ObjectId,
      ref: 'gpsCoords',
      required: true,
    },
  },
  { collection: 'gps_tracker_data' }
);

module.exports = mongoose.model('gpsTrackerData', gpsTrackerDataModelSchema);
```

Рисунок 5.8 - Модель даних для запису в Б

Для подальшої роботи необхідно визначити шлях за яким будуть робитися всі запити, в нашому випадку він буде виглядати як `http://10.10.11.100:6000/gps` в ньому вказаний IP-адрес сервера та шлях за яким і буде оброблятися запити.

Перший оброблюваний запит-це запит GET без вказівки ідентифікатора пристрою. Усі записи, що відповідають наведеній вище схемі, підраховуються та відображаються з бази даних. Якщо читання виконано успішно, клієнт отримує об'єкт JSON, що містить дані з усіх пристроїв у базі даних, і видає помилку у разі виникнення помилки.

```
const express = require('express');
const router = express.Router();
const GpsTrackerData = require('../models/gpsTrackerData');

//Отримання всіх GPS-трекерів
router.get('/', async (req, res) => {
  try {
    const gpsTrackerData = await GpsTrackerData.find({ date: { $gte: new Date('2022-01-01') } }).sort({ date: -1 });
    res.status(200).json({
      status: 'ok',
      result: { gpsTrackerData },
    });
  } catch (err) {
    res.status(500).json({ status: 'error', message: err.message });
  }
});
```

Рисунок 5.9 - Обробка GET запиту без вказання ідентифікатора пристрою

Другий запит-це запит GET, який визначає ідентифікатор пристрою. Посилання буде виглядати так `http://10.10.11.100:6000/gps`. Для пристроїв з ідентифікаторами, зазначеними у поданні, дані враховуються з бази даних. Якщо читання успішне, клієнт отримує об'єкт JSON, що містить дані для всіх пристроїв у базі даних, і повертає помилку у разі виникнення помилки.

```

const { NotFoundError } = require('http-errors'); // Бібліотека обробки помилок

router.delete('/:id', getGpsTrackerByQueryDeviceId, async (req, res) => {
  try {
    const deletedTracker = await res.gpsTrackerData.deleteOne();
    if (!deletedTracker.deletedCount) {
      throw new NotFoundError('Device not found');
    }
    res.status(204).json({ status: 'ok', message: 'Deleted' });
  } catch (err) {
    if (err instanceof NotFoundError) {
      res.status(404).json({ status: 'error', message: err.message });
    } else {
      res.status(500).json({ status: 'error', message: err.message });
    }
  }
});

async function getGpsTrackerByQueryDeviceId(req, res, next) {
  try {
    const gpsTrackerData = await GpsTrackerData.findById(req.params.id);
    if (!gpsTrackerData) {
      throw new NotFoundError('Device not found');
    }
    res.gpsTrackerData = gpsTrackerData;
    next();
  } catch (err) {
    if (err instanceof NotFoundError) {
      res.status(404).json({ status: 'error', message: err.message });
    } else {
      res.status(500).json({ status: 'error', message: err.message });
    }
  }
}

//Отримання всіх GPS-трекерів
router.get('/:id', getGpsTrackerByQueryDeviceId, async (req, res) => {
  res.json(res.gpsTrackerData);
});

```

Рисунок 5.10 - Обробка GET запиту з вказання ідентифікатора пристрою

Третім для обробки. запит-це запит POST, який постачається з даними пристрою, ідентифікатор пристрою береться з цих даних, якщо такий пристрій вже існує в базі даних, ці дані оновлюються та генеруються, якщо такого пристрою немає. У відповідь, якщо читання виконано успішно, клієнт отримує об'єкт JSON, що містить записані дані, і в разі виникнення помилки відправляється повідомлення з помилкою.

```
const { BadRequestError } = require('http-errors');

router.post('/', async (req, res) => {
  const { deviceId, date, time, coords } = req.body;
  try {
    let gpsTrackerData;
    gpsTrackerData = await GpsTrackerData.findOne({ deviceId });

    if (gpsTrackerData) {
      // Оновлення наявності даних пристрою
      gpsTrackerData.set({ date, time, coords });
      await gpsTrackerData.save();
      res.status(200).json({ status: 'ok', message: 'Device data updated' });
    } else {
      // Створення нових даних пристрою та збереження
      const newGpsTrackerData = new GpsTrackerData({ deviceId, date, time, coords });
      gpsTrackerData = await newGpsTrackerData.save();
      res.status(201).json({ status: 'ok', message: 'Device created', result: { gpsTrackerData } });
    }
  } catch (err) {
    if (err.name === 'ValidationError') {
      res.status(400).json({ status: 'error', message: 'Validation error: ' + err.message });
    } else {
      res.status(500).json({ status: 'error', message: err.message });
    }
  }
});
```

Рис 5.11 - Обробка POST запиту

Червертим запитом який буде обробляться буде DELETE запит для нього також потрібен ідентифікатора пристрою. при успішному виконанні з бази даних видалиться запис а клієнт отримає повідомлення про успішне виконання операції, а в разі помилки повідомлення з помилкою яка відбулась.

```
router.delete('/:id', getGpsTrackerByQueryDeviceId, async (req, res) => {
  try {
    const deletedTracker = await res.gpsTrackerData.deleteOne();
    if (!deletedTracker.deletedCount) {
      return res.status(404).json({ status: 'error', message: 'Device not found' });
    }
    res.status(204).json({ status: 'ok' });
  } catch (err) {
    res.status(500).json({ status: 'error', message: err.message });
  }
});
```

Рис 5.12 - Обробка DELETE запит

## ВИСНОВКИ

Використання Інтернету речей (IoT) у сфері транспорту та логістики кардинально змінює спосіб управління й оптимізації логістичних процесів. Проведене дослідження показує, що впровадження IoT-технологій сприяє підвищенню ефективності, безпеки та прозорості у всіх етапах транспортного ланцюга, від відстеження вантажів до моніторингу стану транспортних засобів.

**Підвищення ефективності:** Завдяки реальному часу збору та аналізу даних, підприємства можуть швидше реагувати на зміни умов перевезення, оптимізувати маршрути і мінімізувати витрати на паливо та обслуговування транспортних засобів.

**Покращення безпеки:** IoT-системи дозволяють відстежувати стан транспортних засобів і вантажів у реальному часі, що допомагає запобігати аваріям, крадіжкам та іншим інцидентам. Системи моніторингу можуть повідомляти про несправності або небезпечні умови, що дозволяє вчасно вжити відповідних заходів.

**Прозорість і контроль:** Завдяки IoT, логістичні компанії можуть надавати своїм клієнтам більш детальну інформацію про стан і місцезнаходження вантажів. Це сприяє підвищенню довіри клієнтів і дозволяє краще планувати логістичні операції.

**Екологічна стійкість:** Оптимізація маршрутів і ефективне управління ресурсами допомагають знижувати викиди вуглекислого газу, що сприяє зменшенню впливу транспортної галузі на навколишнє середовище.

В цілому, впровадження IoT у транспорті та логістиці є стратегічно важливим кроком для підвищення конкурентоспроможності підприємств на сучасному ринку. Подальші дослідження і розробки в цій області дозволять ще більше розкрити потенціал IoT і забезпечити нові інноваційні рішення для оптимізації логістичних процесів.

## ПЕРЕЛІК ПОСИЛАНЬ

1. RFID: <https://www.rfidjournal.com/>
2. "The Internet of Things: A View from the Top" by Andrew S. Tanenbaum and David K. Gerdes
3. "Designing the Internet of Things" by Adrian McEwen and Hakim Cassimally
4. "Building the Internet of Things" by Vijay Madisetti and Andy J. King
5. "IoT Security: Securing Connected Devices and Systems" by Bruce Potter and David Mimiea
6. "IoT Analytics: Transforming Data into Knowledge" by John Soldatos, Konstantinos Papagiannis, and Michael Lambrinos
7. "The State of IoT in 2023" by Statista (2023)  
<https://www.statista.com/topics/2637/internet-of-things/>
8. "The Future of IoT" by Gartner (2023) <https://www.gartner.com/en/information-technology/insights/internet-of-things>
9. "How IoT is Changing the World" by Forbes (2023)  
<https://www.forbes.com/sites/forbestechcouncil/2022/04/28/iot-and-where-its-going/>
10. "The State of IoT in 2023" by Statista (2023)  
<https://www.statista.com/topics/2637/internet-of-things/>
11. "The Future of IoT" by Gartner (2023) <https://www.gartner.com/en/information-technology/insights/internet-of-things>
12. "IoT Architecture Guide" by Cisco (2023)  
<https://www.cisco.com/c/en/us/solutions/internet-of-things/iot-design-in.html>
13. "Object Tracking System Architecture" by Honeywell (2022)  
<https://aerospace.honeywell.com/us/en/about-us/press-release/2022/10/honeywell-360-display-provide-vehicle-operators-greatly-enhanced-situational-awareness>
14. "IoT Security: Securing Connected Devices and Systems" by Bruce Potter and David Mimiea (2016)  
<https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119896746.ch10>
15. "The Architecture of Object Tracking Information Systems: A Survey" by Youssef Benanni, Mohamed El Saddik, and Abdellah Ait El Haj (2017)  
<https://dl.acm.org/doi/10.1145/1177352.1177355>
16. "Design and Implementation of a Real-time Object Tracking System" by Yingying Zhang, Yongjun Sun, and Huimin Cheng (2016)

<https://ieeexplore.ieee.org/document/6731341>

17. "A Survey on Object Tracking Systems" by Alireza Fekri-Asl and Hamidreza Shirazi (2011) <https://ieeexplore.ieee.org/document/7546127>
18. "Життєвий цикл інформаційної системи: теоретичні аспекти та практичні рекомендації" - [http://www.visnyk-econom.uzhnu.uz.ua/archive/15\\_1\\_2017ua/33.pdf](http://www.visnyk-econom.uzhnu.uz.ua/archive/15_1_2017ua/33.pdf)
19. "Моделі життєвого циклу інформаційних систем" - [http://www.visnyk-econom.uzhnu.uz.ua/archive/15\\_1\\_2017ua/33.pdf](http://www.visnyk-econom.uzhnu.uz.ua/archive/15_1_2017ua/33.pdf)
20. "Методологія розробки інформаційних систем" - <https://core.ac.uk/download/pdf/84273953.pdf>
21. GPSGO: <https://www.gpsgo.com/>
22. Zebra Technologies: <https://www.zebra.com/>
23. Impinj: <https://www.impinj.com/>
24. Alien Technology: <https://www.alientechnology.com/>
25. Axis Communications: <https://www.axis.com/>
26. Hikvision: <https://www.hikvision.com/>
27. IndoorAtlas: <https://www.indooratlas.com/>
28. Tile: <https://www.thetileapp.com/>
29. Chipolo: <https://www.chipolo.net/>
30. RFID Journal: <https://www.rfidjournal.com/>
31. GPS World: <https://www.gpsworld.com/>
32. IndoorAtlas: <https://www.indooratlas.com/>
33. Tile: <https://www.thetileapp.com/>
34. Chipolo: <https://www.chipolo.net/>
35. "A Survey of Object Tracking Systems" by A. El-Saddik et al. (2015)
36. "Object Tracking: A Survey" by A. Yilmaz et al. (2006)
37. "Performance Evaluation of GPS Tracking Systems" by J. Chen et al. (2010)
38. GSM Association: <https://www.gsma.com/>
39. LoRa Alliance: <https://lora-alliance.org/>
40. Wi-Fi Alliance: <https://www.wi-fi.org/>
41. Bluetooth Special Interest Group: <https://www.bluetooth.com/>



42.What is GPS?: <https://www.geotab.com/blog/what-is-gps/>

43.LBS BASE STATION POSITIONING AND GPS SATELLITE POSITIONING

44.COMPARISON: <https://www.roadragon.com/news/lbs-base-station-positioning-and-gps-satellite-positioning-comparison>

45.MongoDB : <https://searchdatamanagement.techtarget.com/definition/MongoDB>

# ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ (презентація)

## ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення автоматизованих систем

### КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Розробка моделі інформаційної системи трекінгу об'єктів з використанням IoT-технологій»

на здобуття освітнього ступеня бакалавра  
зі спеціальності 126 Інформаційні системи та технології  
освітньо-професійної програми Інформаційні системи та технології

Виконав(ла): Дорохін Д.В, ІСД-42  
Науковий керівник роботи:  
Данильченко В.М.

Київ - 2024

1

**Наукова новизна:** Покращення та здешевлення методів трекінгу з використанням ІОТ технологій

**Об'єкт дослідження:** Процеси трекінгу предметів, та їх оптимізація

**Предмет дослідження:** Розробка системи трекінгу на базі платформи Arduino

**Мета дослідження:** Дослідити перспективи інформаційних систем трекінгу з використання ІОТ

### **Завдання дослідження:**

1. Дослідження існуючих систем трекінгу
2. Аналіз існуючих платформ для трекінгу на базі ІОТ
3. Розробка на базі оптимальної системи

2

## Основні принципи IoT

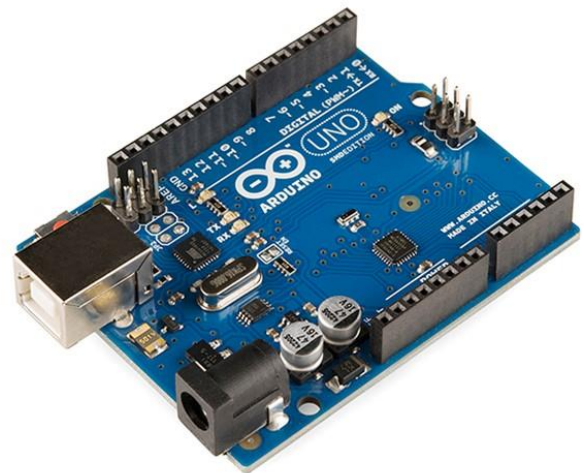
**IoT** - це концепція, що передбачає підключення різних об'єктів до мережі для обміну даними. Використання датчиків та мікроконтролерів дозволяє забезпечити зв'язок та збір інформації.



3

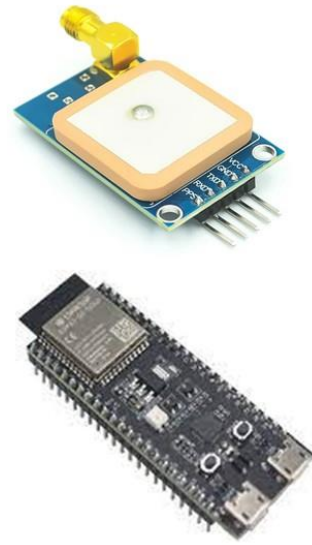
## ARDUINO В IOT

Платформа Arduino є популярним інструментом для розробки IoT-проектів. Вона надає зручний інтерфейс для програмування та підключення датчиків та периферійних пристроїв.



## РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ

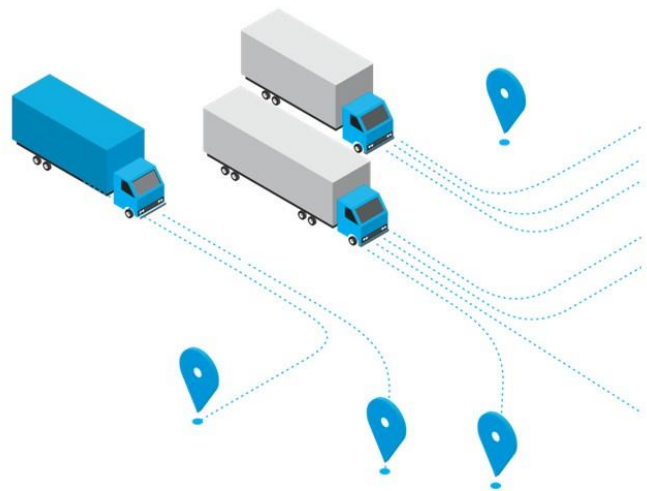
Для розробки інформаційної системи трекінгу використовуються датчики GNSS та модуль wifi. Вони дозволяють відстежувати рух об'єктів та збирати дані для аналізу.



5

## ЗАСТОСУВАННЯ ІОТ-ТЕХНОЛОГІЙ

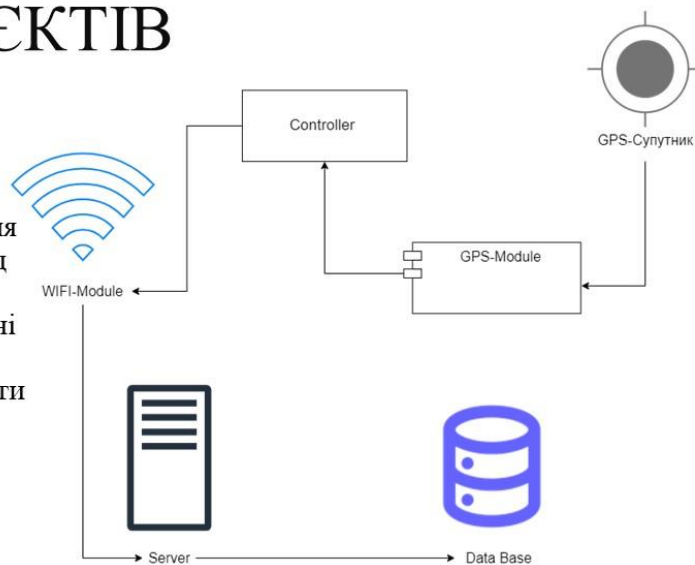
Використання ІоТ-технологій у системі трекінгу дозволяє забезпечити постійний моніторинг та аналіз даних про рух об'єктів. Це важливо для безпеки, логістики та управління ресурсами.



6

## ПЛАН РОБОТИ СИСТЕМ ТРЕКІНГУ ОБ'ЄКТІВ

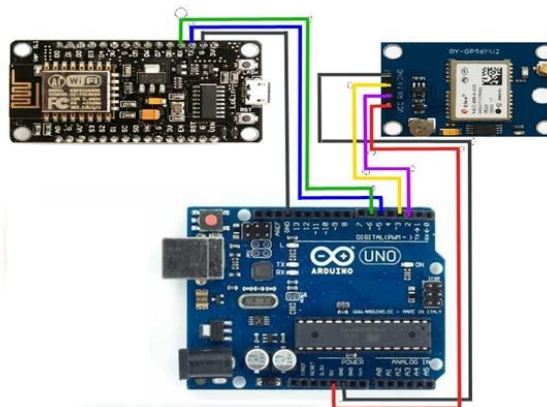
Датчики, встановлені на вантажах, транспортних контейнерах, транспортних засобах або суднах, використовують телематику та GPS для збору даних про місцезнаходження під час подорожі. Ці дані надсилаються в хмару через мережу Wi-Fi. Коли ці дані знаходяться в хмарі або на сервері, уповноважені сторони можуть отримати доступ до них з будь-якого місця та в будь-який час.



7

## РОЗРОБКА МАКЕТУ GPS-ТРЕКЕРА

- Модуль GPS підключається до супутника, отримує координати, час і дату
- Модуль GPS надішле дані на Arduino Uno
- Arduino аналізує дані, що надходять з модуля GPS, створює форму, необхідну для передачі, і відправляє їх по цифрових каналах в модуль Wi-Fi
- Модуль Wi-Fi створює та надсилає запит на сервер



8

## ВИСНОВКИ

Інтернет речей (IoT) у транспорті та логістиці значно розширює можливості галузі, забезпечуючи безперервні потоки даних, недосяжні традиційними системами. У поєднанні з машинним навчанням і штучним інтелектом, IoT робить ланцюжки поставок прозорими та ефективними. У роботі проаналізовано архітектури та топології систем, протоколи бездротової передачі даних, технології визначення геолокації, а також необхідні компоненти для побудови GPS-трекера та серверного програмного забезпечення. Розроблено макет GPS-трекера для отримання і передачі геолокаційних даних на сервер, який обробляє і зберігає ці дані.