

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр

(назва освітнього ступеня)

на тему: Аналіз стандартів інтеграції IoT-платформ та «розумних» застосунків

Виконала: студентка IV курсу, групи СНС-42

спеціальності 122 Комп'ютерні науки

(шифр і назва спеціальності)

Фанга С.Б.

(підпис)

(прізвище та ініціали)

Керівник

Станько А.А.

(підпис)

(прізвище та ініціали)

Консультант

Дуда І

(підпис)

(прізвище та ініціали)

Нормоконтроль

Марценко С.В.

(підпис)

(прізвище та ініціали)

Завідувач кафедри

Боднарчук І.О.

(підпис)

(прізвище та ініціали)

Рецензент

Голотенко О.С.

(підпис)

(прізвище та ініціали)

Тернопіль
2024

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних наук
(повна назва кафедри)

ЗАТВЕРДЖУЮ

Завідувач кафедри

Боднарчук І.О.
(підпис) (прізвище та ініціали)

«29» червня 2024 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 122 Комп'ютерні науки
(шифр і назва спеціальності)

Студенту Фанга Софія Богданівна
(прізвище, ім'я, по батькові)

1. Тема роботи Аналіз стандартів інтеграції IoT-платформ та «розумних» застосунків

Керівник роботи Станько Андрій Андрійович, доктор філософії, асистент кафедри КН
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «29» квітня 2024 року № 4/7-472

2. Термін подання студентом завершеної роботи 24 червня 2024р.

3. Вихідні дані до роботи Наукові публікації інтеграції IoT-платформ та «розумних» застосунків

4. Зміст роботи (перелік питань, які потрібно розробити)

Вступ. 1 Аналіз предметної області та постановка завдання. 1.1 IoT платформи та «розумні»

Застосунки. 1.2 Визначення термінів в проектах «розумне місто». 1.3 Інтегрованість

та стандартизація. 1.4 Послуги LOD.1.5. Послуги мешапування.1.6 Послуги вивісок та BEMS

1.7 Стандартна сервісна платформа IoT oneM2M. 1.8 Семантичні технології

1.9 Висновок до першого розділу. 2 Проектна частина. Онтологія та архітектура.

2.1 Використання онтологій. 2.2 Онтологія LOD та мешапів. 2.3 Вивіски та онтологія BEMS.

2.4 Архітектура та фреймворки. 2.5 Висновок до другого розділу. 3 Практична частина.

Графічний інтерфейс. 3.1 Графічний інтерфейс інформаційної панелі. 3.2 Графічний

Інтерфейс LOD. 3.3 Графічний інтерфейс мешапу. 3.4 Графічний інтерфейс вивіски.

3.5 Висновок до третього розділу. 4. Безпека життєдіяльності, основи охорони праці.

Висновки. Перелік джерел. Додатки .

5. Перелік графічного матеріалу

1 Титульна сторінка. 2 Тема, Мета, Об'єкт, Предмет дослідження. 3 Завдання дослідження.

4 Актуальність дослідження. 5 Архітектура фреймворку ЦОД. 6 Потік даних LOD.

7 Ієрархія ресурсів мешапу.8 Семантика Архітектури Mashup.9 Система енергоменеджменту

10 Система оповіщення населення. 11 Система оповіщення населення. 12 Розумне місто,

онтологія ЦОД. 13 Впровадження семантики/сервісів IoT. 14 Графічний інтерфейс.

15 Висновки. 13 Завершальний.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи охорони праці	Сенчишин В.С.	12.06.2024	15.06.2024

7. Дата видачі завдання 29 січня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	30.01.2024	<i>Виконано</i>
2.	Підбір джерел про стандарти інтеграції IoT-платформ та «розумних» застосунків	31.01.2024–03.02.2024	<i>Виконано</i>
3.	Опрацювання джерел по темі кваліфікаційної роботи	04.02.2024–06.02.2024	<i>Виконано</i>
4.	Виконання дослідження щодо онтологій та архітектури	07.02.2024–09.02.2024	<i>Виконано</i>
	Розроблення архітектури та графічних інтерфейсів	10.02.2024–11.02.2024	<i>Виконано</i>
5.	Оформлення розділу «Аналіз предметної області та постановка завдання»	03.06.2024–05.06.2024	<i>Виконано</i>
6.	Оформлення розділу «Проектна частина. Онтологія та архітектура»	06.06.2024–08.06.2024	<i>Виконано</i>
6.	Оформлення розділу «Практична частина. Графічний Інтерфейс»	09.06.2024–11.06.2024	<i>Виконано</i>
7.	Виконання завдання до підрозділу «Безпека життєдіяльності»	12.06.2024–13.06.2024	<i>Виконано</i>
8.	Виконання завдання до підрозділу «Основи охорони праці»	14.06.2024–15.06.2024	<i>Виконано</i>
9.	Оформлення кваліфікаційної роботи	16.06.2024–17.06.2024	<i>Виконано</i>
10.	Нормоконтроль	18.06.2024–19.06.2024	<i>Виконано</i>
11.	Перевірка на плагіат	20.06.2024	<i>Виконано</i>
12.	Попередній захист кваліфікаційної роботи	21.06.2024	<i>Виконано</i>
13.	Захист кваліфікаційної роботи	30.06.2024	

Студент

(підпис)

Фанга С.Б.

(прізвище та ініціали)

Керівник роботи

(підпис)

Станько А.А.

(прізвище та ініціали)

АНОТАЦІЯ

Аналіз стандартів інтеграції IoT-платформ та «розумних» застосунків // Кваліфікаційна робота освітнього рівня «Бакалавр» // Фанга Софія Богданівна // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра комп'ютерних наук, група СНс-42 // Тернопіль, 2024 // С. 74, рис. – 27, табл. – 1, кресл. – 16, додат. – 6, бібліогр. – 68.

Ключові слова: інтернет речей, семантичні технології, онтологія, інтероперабельність, розумне місто, інформаційні платформи, стандарти, інтелектуальні сервіси.

Кваліфікаційна робота присвячена дослідженню стандартів інтеграції IoT-платформ та "розумних" застосунків, зокрема використанню семантичних технологій для покращення інтероперабельності та ефективного управління даними у розумних містах. У роботі розглядаються методи семантичної анотації, розробка онтологій, а також впровадження інтелектуальних сервісів для забезпечення зручного та ефективного використання IoT-систем.

В першому розділі кваліфікаційної роботи освітнього рівня «Бакалавр»:

- Подано огляд існуючих IoT-платформ та "розумних" застосунків, їх структуру та основні функції.
- Розглянуто проблеми інтеграції та інтероперабельності між різними IoT-системами і платформами.
- Висвітлено важливість стандартизації даних для ефективного обміну інформацією між різними системами.
- Проаналізовано існуючі семантичні технології та їх застосування для покращення інтероперабельності в IoT-системах.

В другому розділі кваліфікаційної роботи:

- Досліджено використання онтологій та семантичних моделей для забезпечення інтероперабельності даних у IoT-системах.

- Обґрунтовано вибір існуючих онтологій, таких як SAREF і її розширення, для застосування в інтелектуальних мережах та системах енергоменеджменту.

- Сформовано семантичні моделі для конкретних застосунків у розумних містах, таких як системи управління енергоспоживанням та вивісок.

В третьому розділі кваліфікаційної роботи:

- Розроблено інформаційну панель адміністратора для управління IoT-сервісами та моніторингу даних.

- Запропоновано підхід до інтеграції різних IoT-платформ на основі семантичних технологій.

- Спроектовано графічні інтерфейси для взаємодії з даними та управління сервісами LOD, Mashup та BEMS.

- Протестовано розроблені сервіси та інтерфейси в умовах реальних проектів розумного міста, оцінено їх ефективність та зручність у використанні.

У розділі «Безпека життєдіяльності, основи охорони праці» висвітлено особливості організації праці при виконанні робіт в обчислювальному центрі та вплив способу життя людини на професійну діяльність.

Об'єкт дослідження: стандарти інтеграції IoT-платформ та «розумних» застосунків. Це включає аналіз технологій, протоколів, моделей даних та семантичних технологій, які використовуються для забезпечення інтероперабельності та ефективного управління даними в системах IoT для розумних міст.

Предмет дослідження: семантичні технології та онтології, які застосовуються для інтеграції IoT-платформ та «розумних» застосунків. Це включає в себе методи семантичної анотації даних, розробку та використання онтологій, а також реалізацію сервісів, що забезпечують інтероперабельність і ефективну обробку даних в розумних містах.

ANNOTATION

Analysis of Standards for the Integration of IoT Platforms and "Smart" Applications
// Qualification work of the educational level «Bachelor» // Fanha Sofiia Bohdanivna// Ternopil Ivan Pulyu National Technical University, Computer and Information Systems and Software Engineering Faculty, Computer Sciences Department, group SNs-42 // Ternopil, 2024 // P. 74, fig. – 27, tabl. – 1, chair. – 16, annexes. – 6, references – 68.

Keywords: Internet of Things, semantic technologies, ontology, interoperability, smart city, information platforms, standards, intelligent services.

The qualification work is devoted to the study of standards for the integration of IoT platforms and smart applications, in particular the use of semantic technologies to improve interoperability and effective data management in smart cities. The work covers semantic annotation methods, ontology development, and the implementation of intelligent services to ensure convenient and efficient use of IoT systems.

The first chapter of the qualification work of the educational level "Bachelor":

- An overview of existing IoT platforms and smart applications, their structure and basic functions is provided.
- The problems of integration and interoperability between different IoT systems and platforms are considered.
- The importance of data standardization for the efficient exchange of information between different systems is highlighted.
- Existing semantic technologies and their application to improve interoperability in IoT systems are analyzed.

In the second chapter of the qualification work:

- The use of ontologies and semantic models to ensure data interoperability in IoT systems is investigated.

- The choice of existing ontologies, such as SAREF and its extensions, for use in smart grids and energy management systems is substantiated.

- Semantic models for specific applications in smart cities, such as energy management and signage systems, have been formed.

In the third section of the qualification work:

- An administrator's dashboard for managing IoT services and monitoring data has been developed.

- An approach to the integration of various IoT platforms based on semantic technologies is proposed.

- Graphical interfaces for interacting with data and managing LOD, Mashup, and BEMS services were designed.

- The developed services and interfaces were tested in the context of real smart city projects, and their effectiveness and ease of use were evaluated.

In the section "Life safety, basics of labor protection" ... the peculiarities of labor organization when performing work in a computer center and the impact of a person's lifestyle on professional activities are highlighted.

Object of study: standards for integrating IoT platforms and smart applications. This includes the analysis of technologies, protocols, data models, and semantic technologies used to ensure interoperability and effective data management in IoT systems for smart cities.

Subject matter: There are semantic technologies and ontologies that are used to integrate IoT platforms and smart applications. This includes methods of semantic data annotation, development and use of ontologies, and implementation of services that ensure interoperability and efficient data processing in smart cities.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

IoT (англ. Internet of Things) – Інтернет речей

LOD (англ. Linked Open Data) – Зв'язані відкриті дані

BEMS (англ. Building Energy Management System) – Система управління енергоспоживанням будівель

SPARQL (англ. SPARQL Protocol and RDF Query Language) – Протокол SPARQL і мова запитів RDF

RDF (англ. Resource Description Framework) – Рамка опису ресурсів

OWL (англ. Web Ontology Language) – Мова веб-онтологій

API (англ. Application Programming Interface) – Інтерфейс програмування застосунків

SAREF (англ. Smart Appliances REference) – Референційна модель для розумних пристроїв

oneM2M (англ. One Machine-to-Machine) – Глобальний стандарт для M2M-комунікацій

BMS (англ. Building Management System) – Система управління будівлею

MQTT (англ. Message Queuing Telemetry Transport) – Транспорт телеметрії черги повідомлень

CoAP (англ. Constrained Application Protocol) – Протокол обмежених застосунків

REST (англ. Representational State Transfer) – Передача представлення стану

JSON (англ. JavaScript Object Notation) – Нотація об'єктів JavaScript

XML (англ. Extensible Markup Language) – Розширювана мова розмітки

ЗМІСТ

ВСТУП	10
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ.....	13
1.1 IoT–платформи та «розумні» застосунки	13
1.2 Визначення термінів в проектах «розумне місто».....	15
1.3 Інтероперабельність та стандартизація.....	16
1.4 Послуги LOD	17
1.5 Послуги мешапування.....	19
1.6 Послуги вивісок та BEMS	23
1.6.1 Сонячна енергетика	23
1.6.2 Система енергетичного менеджменту будівель (BEMS).....	25
1.6.3 Системи оповіщення.....	28
1.7 Стандартна сервісна платформа IoT oneM2M.....	29
1.8 Семантичні технології	34
1.9 Висновок до першого розділу	36
РОЗДІЛ 2. ПРОЕКТНА ЧАСТИНА. ОНТОЛОГІЯ ТА АРХІТЕКТУРА	37
2.1 Використання онтологій.....	37
2.2 Онтологія LOD та мешапів.....	38
2.3 Вивіски та онтологія BEMS	40
2.4 Архітектура та фреймворки	41
2.4.1 Реалізація бекенду.....	42
2.4.2 Реалізація фронтенду	45
2.5 Висновок до другого розділу	46
РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА. ГРАФІЧНИЙ ІНТЕРФЕЙС	47
3.1 Графічний інтерфейс інформаційної панелі.....	47
3.2 Графічний інтерфейс LOD	49
3.3 Графічний інтерфейс мешапу	53
3.4 Графічний інтерфейс вивіски.....	53

3.5 Висновок до третього розділу	57
РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ	58
4.1 Організація праці при виконанні робіт в обчислювальному центрі ..	58
4.2 Здоровий спосіб життя людини та його вплив на професійну діяльність	62
4.3 Висновок до четвертого розділу	65
ВИСНОВКИ.....	66
ПЕРЕЛІК ДЖЕРЕЛ.....	68
ДОДАТКИ	

ВСТУП

Актуальність теми. Технологія Інтернету речей (IoT), яка дозволяє багатьом пристроям підключатися один до одного. Однак, ці пристрої здебільшого походять від різних виробників з різними стандартами продукції, що створює проблеми з інтерактивним управлінням. Крім того, ці пристрої генерують великі обсяги даних, і ефективне вивчення цієї інформації для інтелектуальних сервісів, а також їх гетерогенна природа, речі спілкуються в різній семантиці, що робить інтеграцію складною і дорогою для досягнення. Крім того, інші проблеми, пов'язані з Інтернетом речей, полягають у забезпеченні інтероперабельності даних Інтернету речей, розумінні даних, що генеруються пристроями Інтернету речей, і підтримці розробників у виконанні цих обов'язків шляхом визначення правил.

Наразі багато платформ Інтернету речей пропонують послуги з моделювання даних та пошуку інформації, однак ці платформи використовують різні моделі даних і стандарти, що також призводить до проблем інтероперабельності. Семантичні технології усувають такі бар'єри, забезпечуючи семантичне коментування даних Інтернету речей на основі передових і поширених моделей даних. Тому семантичне просвітлення стає все більш важливою вимогою і ключовим завданням для платформ сервісного рівня IoT. Дані, зібрані з різних пристроїв і вертикальних сховищ, можуть отримати розширене значення і можуть бути більш ефективно проаналізовані за допомогою семантичної підтримки в системах IoT.

В даній дипломній роботі основна увага приділяється представленню ряду семантичних функцій в платформі IoT для інтелектуальних сервісів IoT, які базуються на семантиці для розумних міст з реалізацією сервісів розпізнавання семантики, а також візуалізації результатів роботи цих сервісів. Для проектування онтологій та розроблених сервісів для проекту розумного міста, сервіси розділені на три категорії: LOD, mashups та signage & BEMS. Інформаційна панель адміністратора керує всіма сервісами в одному місці.

Крім того, за допомогою цієї панелі також можна керувати іншими проектами, що імплементуються в IoT.

Тому є актуальним необхідність вирішення проблем інтеоперабельності та інтеграції різних IoT-пристроїв і платформ, що генерують великі обсяги гетерогенних даних. Використання семантичних технологій і стандартних моделей даних дозволяє усунути ці бар'єри, покращуючи обробку та аналіз інформації для створення інтелектуальних сервісів у розумних містах.

Мета і задачі дослідження. Метою даної кваліфікаційної роботи освітнього рівня «Бакалавр» є аналіз стандартів інтеграції IoT-платформ та «розумних» застосунків з використанням семантичних технологій, а також розробка і впровадження семантичних сервісів для покращення інтеоперабельності та ефективного управління даними у розумних містах.

Для досягнення поставленої мети потрібно виконати ряд завдань, зокрема:

- Проаналізувати стан досліджень в області інтеграції IoT-платформ та семантичних технологій для "розумних" застосунків.
- Вивчити існуючі стандарти та моделі даних, що застосовуються для забезпечення інтеоперабельності IoT-систем.
- Розробити онтології та семантичні моделі для специфічних застосунків у розумних містах.
- Реалізувати сервіси семантичного аналізу та інтеграції даних, такі як LOD, Mashup та BEMS.
- Створити інформаційну панель адміністратора для управління IoT-сервісами та забезпечення зручного доступу до даних і налаштувань.
- Провести тестування та оцінку ефективності розроблених рішень у реальних умовах розумного міста.
- Розробити рекомендації щодо впровадження семантичних технологій у існуючі та нові IoT-проекти.

Практичне значення одержаних результатів. Практичне значення одержаних результатів дипломної роботи полягає у покращенні інтеграції та

інтероперабельності IoT-систем завдяки використанню семантичних технологій і онтологій. Це забезпечує більш ефективну взаємодію між пристроями різних виробників, оптимізує управління даними та полегшує їх обробку і аналіз. Розробка нових інтелектуальних сервісів, таких як LOD, Mashup та BEMS, створює корисні індекси та метрики для міських сервісів. Універсальна інформаційна панель адміністратора спрощує управління різними IoT-проектами, забезпечуючи зручний доступ до налаштувань та моніторингу. Запропоновані рішення можуть бути адаптовані для інших IoT-проектів завдяки модульній структурі та використанню стандартних протоколів. Це підвищує ефективність розумних міст, де різні системи взаємодіють на основі єдиних стандартів і семантичних моделей, що сприяє підвищенню якості життя в містах..

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАННЯ.

1.1 IoT–платформи та «розумні» застосунки

Інтернет речей (IoT) об'єднує величезну кількість унікально ідентифікованих пристроїв, які збирають і передають інформацію до різних інтелектуальних додатків і сервісів. IoT є однією з найбільш швидкозростаючих технологічних галузей, яка використовується в різних сферах для підвищення енергоефективності та забезпечення економічної ефективності, таких як промисловість, транспорт, охорона здоров'я, «розумні» міста та будівництво/виробництво [1]. Однак, багато додатків Інтернету речей наразі впроваджуються у специфічних галузях і недостатньою мірою забезпечують обмін знаннями та можливостями один з одним. Таким вертикально адаптованим додаткам і системам не вистачає інтеперабельності [2] для створення горизонтально спільних платформ. Гетерогенна природа цих різноманітних додатків, які використовують інформацію з певної галузі, зумовлює гостру потребу у визначенні інноваційного стандартного способу абстрагування вертикально–специфічних моделей даних, який буде корисним для реалізації повного потенціалу систем IoT. Оскільки IoT характеризується різними джерелами даних з різними характеристиками, такими як залежність від часу та місця розташування, безперервність або зашумленість, важко обмінюватися значущою інформацією між такими різноманітними вертикалями. Відповідно, семантичний [3] підхід, що додає додаткове значення до даних Інтернету речей, привертає увагу як потенційне рішення для обміну корисною інформацією та отримання новітніх знань з даних Інтернету речей на глобальному рівні [4].

Платформи сервісного рівня IoT в основному призначені для підключення величезних пристроїв IoT та управління даними, зібраними з керуючих пристроїв. Однак більшість цих платформ забезпечують простий

спосіб зберігання даних, тому складно додати додаткове значення збереженим даним IoT [5]. Як наслідок, додатки Інтернету речей мають обмежений доступ до зібраного набору даних, оскільки вони не можуть зрозуміти значення та контекст даних. Ситуація навіть схожа на стандартизовані платформи IoT [6]. Якщо стандартизовані сервісні платформи IoT не надають жодних засобів для додавання значущої інформації про дані IoT разом з необробленими даними датчиків [7], такими як їх геолокація, одиниці та виробник, збережені додатки IoT не можуть використовувати дані IoT. Оскільки спосіб управління даними в IoT відрізняється від Інтернету, розробка стандартизованого механізму додавання семантичної інформації [8] до даних IoT стає важливим питанням у багатьох стандартах IoT. Глобальні стандарти IoT, такі як oneM2M, вже впровадили семантичну технологію для додавання додаткових метаданих до необроблених даних датчиків IoT. Ця дослідницька робота пропонує продемонструвати, як семантичні технології можуть бути включені до платформ сервісного рівня IoT у стандартизований спосіб.

Семантична інформація явно представляє значення того, що зберігається в ресурсах oneM2M. Концепції та пов'язані з ними властивості, відповідно до яких моделюється семантична інформація, визначаються в онтологіях. oneM2M не обмежує використання онтології, але дозволяє використовувати одні й ті ж онтології в різних додатках для підвищення ефективності. Семантична інформація в oneM2M моделюється у вигляді RDF-трийок і представляється в RDF/XML-серіалізації RDF. Онтології моделюються в OWL і також представлені в RDF/XML серіалізації. Існуюча система виявлення семантичних ресурсів в oneM2M допомагає виявляти різні пристрої IoT та їхні дані. У багатьох прикладних сценаріях виявлені дані потребують подальшої обробки (наприклад, інтегрованої/оркестрованої/комбінованої) на основі певної бізнес-логіки додатку [9]. Ця робота зосереджена на розробці стандартизованих сервісів Інтернету речей для трьох доменів розумного міста, таких як LOD центру обробки даних, об'єднання паркувальних і погодних сервісів та сервісів

сонячних вивісок. Інформаційна панель розроблена для управління цими послугами, а також для віртуалізації даних імплантованих IoT-проектів.

1.2 Визначення термінів в проектах «розумне місто»

– Розумне місто – це потужний набір технологій, які можуть підвищити якість життя людей, інтегруючи дані з різних датчиків, встановлених у містах.

– Семантика – це філософське та лінгвістичне дослідження значення в мові, мовах програмування, традиційній логіці та семіотиці. Вона пов'язана з відношеннями між дескрипторами, такими як слова, фрази, знаки та символи.

– Анотація – це додаткова інформація, пов'язана з певним місцем у документі, або додаткова частина інформації. Це може бути примітка, яка містить пояснення або опис/коментарі.

– Онтологія: У формально-семантичному плані онтологія – це специфікація концептуалізації, тобто абстрактний і спрощений погляд на світ, бажання представити, виражене формальною мовою.

– Семантична анотація – це технологія вбудовування метаданих і зв'язків з більшою кількістю веб-об'єктів з використанням загальної термінології, яка є однією з основних технологій для семантичної павутини і може бути використана для даних розумного міста для створення метаданих і виявлення зв'язків.

– Семантична павутина – це технологія, яка сприяла швидкому, ефективному пошуку великих обсягів даних з моменту появи Інтернету і забезпечує доступність всіх даних.

– Пов'язані дані – це важлива концепція, яка може забезпечити багато послуг розумного міста шляхом додавання і прикріплення метаданих до кожного з даних, що існують в розумному місті. Зв'язані дані – це метод розповсюдження структурованих даних за допомогою словників, таких як schema.org, які можуть бути пов'язані та інтерпретовані машинами.

Використовуючи зв'язані дані, твердження, перетворені в трійки, можна поширювати на різних веб-сайтах.

– Семантична інтероперабельність знаходиться на найвищому рівні розуміння даних, порівняно з іншими суміжними поняттями, такими як синтаксична інтероперабельність або транспортна інтероперабельність.

1.3 Інтероперабельність та стандартизація

Технології семантичного вебу широко використовуються для розробки інтерактивних веб-додатків у різних галузях. Наприклад, семантичний веб використовується в сферах екології та автоматизації для вирішення проблем інтероперабельності даних за допомогою контекстно-орієнтованих додатків і сервісів. Як і семантичний веб, використання семантичних технологій на платформах сервісного рівня Інтернету речей [10] може покращити доступність даних, їх виявлення та можливість отримання знань про дані. Оскільки спосіб управління даними в IoT відрізняється від веб, розробка стандартизованого механізму додавання семантичної інформації [8] до даних IoT стає важливим питанням у багатьох стандартах IoT. Глобальні стандарти IoT, такі як oneM2M, вже впровадили семантичну технологію для додавання додаткових метаданих до необроблених даних датчиків IoT.

У світі IoT семантична інтероперабельність [11] є важливою, оскільки машини набагато менш впевнені в обробці невизначеної інформації, ніж люди. Це вимагає, щоб дані були виконані правильно. Міжнародна організація зі стандартизації IoT, oneM2M [12], підтримує функціональність семантики та семантичний машап для забезпечення інтероперабельності як одну з ключових технологій з самого початку. Це економить витрати на розробку, інтеграцію та обслуговування системи. Проектувальники та розробники систем повинні обирати спільні моделі даних, онтології та семантичні інструменти у відкритих стандартах, таких як oneM2M [12]. Це забезпечує основу для побудови відкритої та гнучко взаємодіючої системної архітектури.

Семантичне моделювання створює чіткий опис значення даних у чітко визначений спосіб шляхом об'єднання знань про предметну область і контекстно–релевантних даних з необробленими вимірними даними. Семантика включає онтології [13], контексти та структуровані метадані [14]. Семантика дозволяє машинам розуміти значення інформації в Інтернеті. Онтологія стосується того, які види речей існують – які сутності або «речі» існують у всесвіті. Семантична інтероперабельність – це розуміння контексту комунікації та узгодження термінологій.

Отже, через гетерогенну природу пристроїв Інтернету речей, їхня інтероперабельність є однією з головних проблем, яку необхідно вирішити шляхом інтеграції з екосистемою oneM2M, щоб полегшити інтеграцію та розробку сервісів, додатків Інтернету речей та різноманітних фреймворків Інтернету речей. Ця робота пропонує продемонструвати, як семантичні технології можуть бути включені до платформ рівня послуг IoT у стандартизований спосіб. Мета полягає в тому, щоб забезпечити загальну стандартну платформу для підтримки семантики для досягнення інтероперабельних рішень в різних областях IoT. Таким чином, ця робота складається з трьох частин для проектування онтологій та розроблених сервісів для проекту розумного міста: LOD–сервіс дата–хабу, Mashup–сервіс та Signages & BEMS–сервіс. Для управління цими сервісами в одному місці розроблено інформаційну панель адміністратора. Крім того, до інформаційної панелі інтегровані інші проекти, що імплантуються в IoT.

1.4 Послуги LOD

У цьому розділі представлено огляд LOD дата–хабу та внесок у надання послуг. Проект дата–хабу є стійким інтелектуальним проектом, який забезпечує інтегроване управління, експлуатацію та інфраструктуру послуг «розумного» міста на основі даних, отриманих від мешканців міста, навколишнього середовища та інфраструктури інформаційних та комунікаційних технологій

(ІКТ), щоб різні міські суб'єкти могли брати участь у безперервному створенні послуг та інтелектуальному управлінні містом. Модулі семантики для проекту показані на Рисунку 1.1.

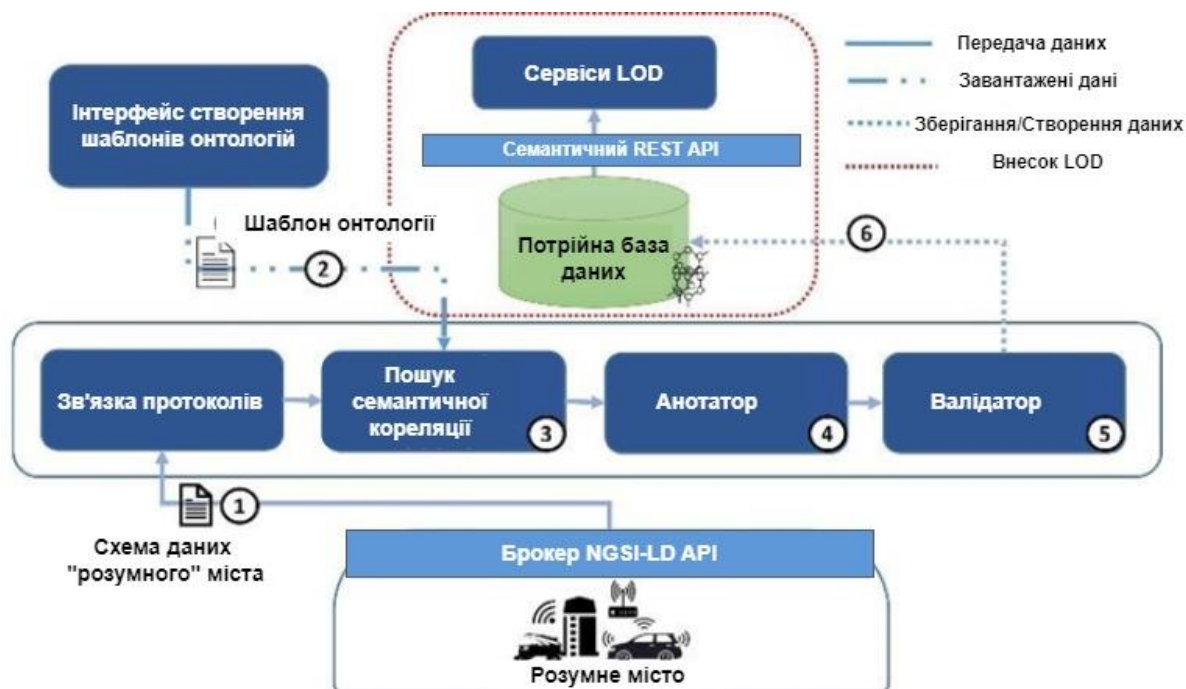


Рисунок 1.1 – Архітектура фреймворку Центру обробки даних

LOD – це концепція, яка поєднує в собі Linked Data [16], що об'єднує дані відповідно до взаємних зв'язків, та Open Data, тобто відкриті дані, якими може користуватися будь-хто. LOD має форму, в якій дані пов'язані різними відношеннями, і видає набір даних, щоб будь-хто міг повторно використовувати пов'язані дані через уніфікований ідентифікатор ресурсу (URI) в Інтернеті. Це покращує машиночитаність завдяки використанню мов на основі відношень, таких як Resource Description Framework (RDF) і W3C Web Ontology Language (OWL), а також зменшує дублювання даних, оскільки використовує URI, який є унікальною системою ідентифікації.

Суть роботи полягає в тому, що сервіс LOD видає змістовні [17] дані, пов'язуючи дані, що генеруються в різних галузях розумного міста, один з одним. Він забезпечує семантичний зв'язок на основі даних для управління, збереження та обслуговування міських ресурсів і надає функцію розподілу

наборів даних, щоб користувачі могли використовувати промислові та загальнодоступні дані з центру обробки даних. Крім того, він надає послуги, щоб користувачі без знань семантики могли шукати семантичні дані і отримувати доступ до пов'язаних даних, а також візуалізувати зв'язок між даними, щоб легко зрозуміти взаємозв'язок між ними. У модулі веб-сервісів LOD запит надсилається на сервер з деякими необхідними параметрами (ім'я, ліміт, тип або у вигляді файлу тощо) за допомогою інтерфейсу прикладного програмування (API) Representational state transfer (REST). Сервер відповідає на запит і повертає потрібні дані у форматі JSON, перетворюючи отриману карту в простий об'єкт перед тим, як відправити її клієнту. На рисунку 1.2 показано підсумок.

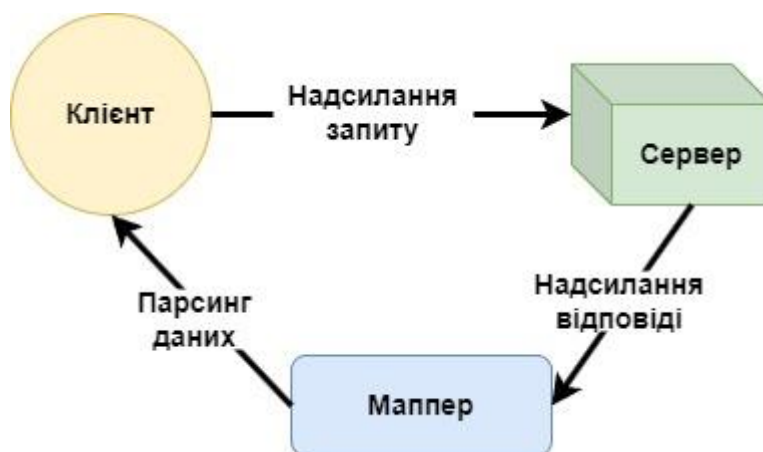


Рисунок 1.2 – Потік даних LOD

1.5 Послуги мешапування

Нижче описано огляд сервісів семантичного мешапу, підтримку семантичного мешапу в oneM2M та внесок сервісу. Останнім часом в IoT проекти вимагають використання сервісів мешапу [18], оскільки знання можуть бути об'єднані з різних ресурсів, а висновки можна зробити легко. Механізми міркувань і семантичні правила були розроблені для різних варіантів використання і призначені для автоматичної обробки подій і об'єднання ресурсів [19]. Процес збору даних з декількох додатків називається

семантичним машапом (Semantic Mashup) [20]. Він об'єднує різноманітні дані з багатьма семантичними анотаціями для легкої інтеграції і, в кінцевому підсумку, генерує значущі результати. У контексті oneM2M звичайний ресурс (наприклад, ресурс <AE>, що представляє датчик температури) може бути анотований семантичними описами, а потім може бути виявлений і ідентифікований як потенційне джерело даних для конкретного мешап-додатку за допомогою семантичного виявлення ресурсів [21].

- Підтримка Семантичного мешапу в oneM2M:

Існуючі засоби виявлення семантичних ресурсів в oneM2M можуть допомогти у виявленні різних пристроїв Інтернету речей та їхніх даних. Однак у багатьох сценаріях застосування виявлені дані потребують подальшої обробки на основі конкретної бізнес-логіки додатку (наприклад, інтегрованої/оркестрованої/комбінованої). Процес збору даних з декількох додатків називається семантичним мешапом [22]. Семантичний мешап підкреслює використання технологій, пов'язаних із семантикою, протягом усього мешапу.

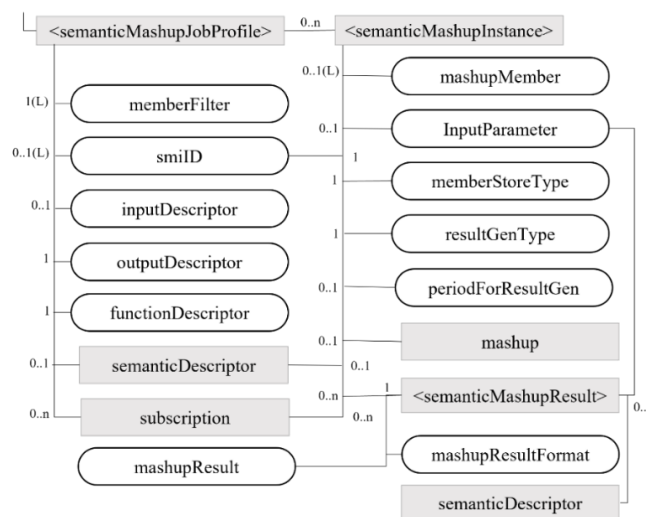


Рисунок 1.3 – Ієрархія ресурсів мешапу

У контексті oneM2M загальний ресурс (наприклад, ресурс <AE>, який представляє датчик температури) може бути анотований за допомогою

семантичних описів, а потім він може бути виявлений та ідентифікований як потенційне джерело даних для конкретного додатку мешапу за допомогою семантичного виявлення ресурсів. Функція семантичного мешапу [23] (Semantic Mashup Function, SMF) є сутністю, відповідальною за збір вхідних даних з джерел даних, розміщених на ресурсних хостах (RH), і генерування результатів мешапу на основі певної бізнес-логіки. Ресурс `<semanticMashupJobProfile>` в oneM2M представляє профіль завдання семантичного мешапу [18] (SMJP), який може бути наданий хостинговому CSE, що забезпечує функцію семантичного мешапу. CSE / AE як запитувач Mashup може запросити створення ресурсу `<semanticMashupInstance>` в іншому oneM2M CSE, який реалізує функцію семантичного мешапу. Кожен створений ресурс `<semanticMashupInstance>` відповідає профілю завдання семантичного злиття (тобто ресурсу `semanticMashupJobProfile`) і виконує операцію мешапу для обчислення результатів мешапу у відповідному

`<semanticMashupJobProfile>` ресурсу. Результат буде збережено в дочірньому ресурсі

`<semanticMashupResult>` відповідного ресурсу `<semanticMashupInstance>`.

`<semanticMashupInstance>` і пов'язані з ним `<semanticMashupJobProfile>` ресурси можуть бути розміщені на одному CSE або на різних CSE, а атрибут `smjpID` `<semanticMashupInstance>` дозволяє шукати відповідний `<semanticMashupJobProfile>` ресурс.

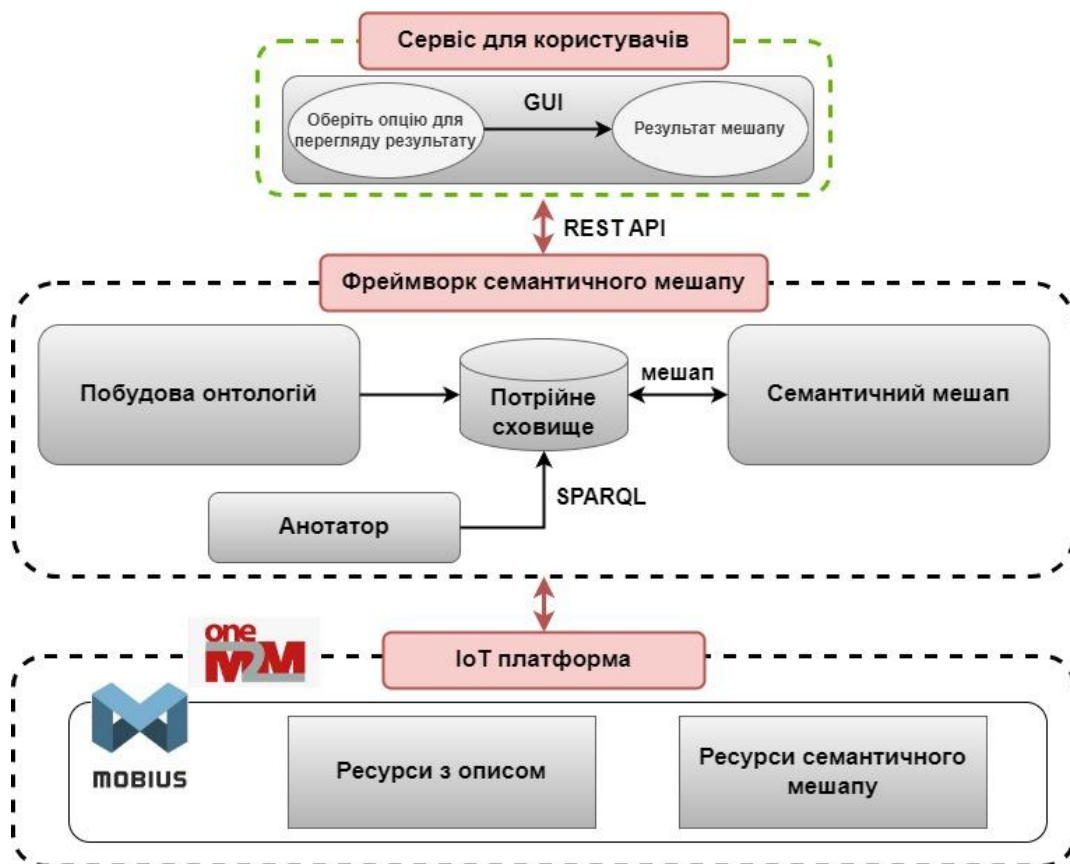


Рисунок 1.4 – Семантика Архітектури Mashup

Ієрархія ресурсів для мешапування показана на рисунку 1.3. У цій роботі були розроблені онтології для паркування та погоди і проведено їх мешап, а також проаналізовано підтримку мешапу в oneM2M, в якому збираються дані з двох різних доменів, таких як погода та паркування, і проводиться їх мешап за допомогою стандартних сервісів oneM2M.

- Архітектура сервісу Mashup

Архітектура системи поділяється на три частини, як показано на рисунку 1.4; Перша частина – це платформа IoT, яка визначає процес виявлення та збору даних з декількох джерел, що виконує функцію мешапу, пов'язану з певною бізнес-логікою на зібраних даних. oneM2M визначає три типи ресурсів (`semanticMashupJobProfile`, `semanticMashupInstance` та `semanticMashupResult`) для мешапу в семантичному управлінні. Друга частина – це фреймворк мешапу. У цьому розділі показано побудову онтології, мешап онтології та правила

міркувань мешапу. У розділі «Користувацькі сервіси» результати отримуються за допомогою REST API, а результат мешапу відображається графічно.

1.6 Послуги вивісок та BEMS

Цей сервіс видає змістовні дані, пов'язуючи дані, згенеровані в будівлі розумного міста, між собою. Щоб інтегрувати сонячні вивіски [24] з розумним містом, можна відстежувати споживання та генерацію енергії [25] в будівлі, визначивши правила [26]. Крім того, він надає сервіси, щоб користувачі без знання семантики могли отримувати дані та візуалізувати взаємозв'язок між даними, щоб легко побудувати графік взаємозв'язку між ними.

1.6.1 Сонячна енергетика

Сонце є найбільшим і найпоширенішим джерелом енергії [27] на Землі. Приблизно 173 000 терават сонячної енергії падає на Землю в будь-який момент часу – це в 10 000 разів більше, ніж загальна потреба світу в енергії. Отримуючи сонячну енергію та перетворюючи її на електрику для дому чи бізнесу, сонячна енергія є ключовим рішенням для подолання нинішньої кліматичної кризи та зменшення нашої залежності від викопних видів палива. Сонце – це природний ядерний контейнер. Воно випускає маленькі енергетичні пакети, що називаються фотонами, які долають 93 мільйони миль від Сонця до Землі приблизно за 8,5 хвилин. Щогодини на планету падає достатня кількість фотонів, щоб генерувати достатню кількість сонячної енергії для задоволення теоретично глобальних енергетичних потреб на цілий рік [28]. Сонячна енергія має безліч переваг [29], деякі з них згадані нижче,

– Першою і головною перевагою сонячної енергії є те, що вона, на відміну від виробництва панелей, не призводить до викидів парникових газів.

– Сонячна енергія виробляється шляхом управління сонячним випромінюванням – процес, який дозволяє уникнути будь-якого диму, газу або інших хімічних побічних продуктів.

– Це головна рушійна сила всіх технологій зеленої енергетики, оскільки країни намагаються виконати свої зобов'язання щодо зміни клімату, обмежуючи викиди парникових газів.

– Ще однією перевагою використання сонячної енергії є те, що, окрім початкового встановлення та обслуговування, сонячна енергія є безкоштовною.

– Сонячна енергетика не потребує дорогої та постійної сировини, такої як нафта чи вугілля, і потребує значно менше робочої сили, ніж звичайне виробництво електроенергії. Сировину не потрібно

Сировина не потребує постійного видобутку, збагачення та транспортування до електростанції.

– Сонячна енергія може вироблятися в мережі або поза нею.

Процес виробництва енергії: Коли фотони потрапляють на сонячну батарею, вони вивільняють електрони з атомів. Якщо провідники з'єднати з позитивною і негативною сторонами елемента, то утворюється електричний ланцюг. Коли електрони проходять через такий ланцюг, вони генерують електрику. Кілька фотоелементів утворюють одну сонячну панель, а кілька сонячних панелей можна утворити, об'єднавши кілька панелей (модулів) разом. Чим більше панелей ви зможете встановити, тим більше енергії ви зможете генерувати.

Сонячні панелі: Фотоелектричні (ФЕ) – складаються з великої кількості сонячних елементів. Сонячні елементи, як і напівпровідники, виготовляються з кремнію. Вони складаються з позитивного і негативного шарів, які разом утворюють електричне поле, так само, як і в акумуляторі.

Завдяки цим перевагам сонячної енергії та суттєвим особливостям світу в поточній ситуації [30], дуже вигідно інтегрувати сонячні вивіски з розумним містом. У цьому дослідженні розглядається інтеграція сонячних сервісів з розумними містами та моніторинг управління енергією в будівлі.

1.6.2 Система енергетичного менеджменту будівель (BEMS)

Система енергоменеджменту будівель (BEMS) [31] – це сучасний метод, який використовується для моніторингу та контролю енергетичних потреб будівлі. Останнім часом система енергетичного менеджменту будівель (BEMS) [32] набула популярності завдяки зростаючому інтересу до енергозбереження та економії енергії в будівлях [33]. Системи енергоменеджменту будівель можуть контролювати та моніторити енергетичні потреби будівлі. Вони не лише забезпечують тепловий комфорт у приміщенні, але й створюють безпечне та здорове середовище всередині житлових будинків, а також громадських, комерційних та промислових будівель і здатні зменшити споживання енергії. Підвищують загальну енергоефективність та заощаджують витрати на електроенергію в будівлі.

Вони можуть ефективно контролювати вісімдесят чотири відсотки енергоспоживання будівлі.

Крім того, система енергоменеджменту будівлі (BEMS) контролює і регулює електричне і механічне обладнання в будівлі, включаючи освітлення, опалення та вентиляцію. BEMS під'єднана до електрощитової будівлі та до головного комп'ютера, що дозволяє керувати періодами ввімкнення/вимкнення освітлення, вологості, температури тощо. Кабелі з'єднують кілька серій концентраторів по всій будівлі з центральним диспетчерським комп'ютером, за допомогою якого оператори можуть здійснювати моніторинг і управління будівлею. Система енергоменеджменту зображена на рисунку 1.5.



Рисунок 1.5 – Система енергоменеджменту

Програмне забезпечення ВЕМ забезпечує функції контролю, обстеження та сигналізації, а також дозволяє операторам покращувати експлуатаційні характеристики будівлі. Частка споживання енергії в будівлях становить близько 20,1% енергії, що використовується в усьому світі, і очікується, що до 2040 року вона зростатиме на 1,5% на рік. ВEMS мінімізує витрати на енергію та економить час. Для прикладу, деякі загальні переваги [34] ВEMS, узагальнені з останніх досліджень, наведені нижче,

– Мінімізація втрат енергії: Моніторинг та контроль енергоспоживання в режимі реального часу дозволяє заощаджувати енергію. Його основна мета – не зменшити використання освітлення, систем опалення, вентиляції та кондиціонування або будь-яких інших електроприладів, але він підкреслює і допомагає використовувати електроприлади в розумний спосіб, наприклад, рівень попиту, заповнюваність, рівень освітленості, рівень вологості, відповідно до часу. День, температура і т.д.

– Екологічно чистий: Оскільки зміна клімату стає реальною проблемою, зростання викидів парникових газів стало глобальною загрозою. Тому, коли споживачі говорять про теплиці або зелені будівлі, вони мають на увазі, що «це вправа в будівництві конструкцій і використанні процедур, які є екологічно надійними і ресурсоефективними». Використовуючи системи

енергоменеджменту будівель, які допомагають контролювати використання енергії (наприклад, тепла, світла та електроенергії в будівлях), і пропонуючи скоротити викиди CO₂, що є основною причиною глобального потепління, споживачі можуть успішно досягти цієї мети.

– Без стресу: На додаток до економії енергії, це також забезпечує душевний спокій. Користувачеві не потрібно постійно тримати електроприлади увімкненими та вимкненими. Вони вмикаються та вимикаються автоматично за потреби. BEMS відстежує енергоспоживання 24 години на добу і сповіщає клієнтів, коли вони відсутні в будівлі.

– Зменшуйте витрати: Система BEMS постійно відстежує та контролює використання енергії. Це автоматично знижує рахунки за електроенергію.

– Розслабтеся: BEMS контролює температуру, світло, якість повітря, вентиляцію, вологість тощо. Це робить середовище більш комфортним та затишним для працівників.

– Заощаджуйте час: Забезпечені легкими для інтерпретації діаграмами, графіками та результатами, користувацькі цифрові щоденні та щомісячні звіти з енергетичної аналітики надають усі дані, необхідні для вжиття заходів з енергозбереження, у вас під рукою.

Енергоменеджмент включає в себе розробку та експлуатацію установок для виробництва та споживання енергії, а також розподіл та зберігання енергії. Енергоменеджмент будівлі – це спосіб контролювати та зменшувати споживання енергії організацією за допомогою сонячних панелей і показувати цю інформацію на екрані користувачеві, як показано на рисунку 2–5. Це дуже важливо, оскільки дозволяє: зменшити витрати – це стає все більш важливим, оскільки ціни на енергію зростають. Щоб мати на увазі ці важливі моменти, я розробив онтологію, яка може охопити наступні декілька варіантів використання з використанням існуючих базових онтологій SEREF та їхніх розширень для моніторингу енергоменеджменту.

UC1: Знайти будівлю в місті (з широтою та довготою).

UC2: Знайти пристрої та інформацію про них, пов'язану з будівлею (наприклад, датчики, сонячні пристрої), та визначити стан пристроїв (наприклад, робота, зупинка тощо).

UC3: Знаходити та обчислювати виробництво енергії щодня, щомісяця та щороку у конкретній будівлі.

UM4: Знаходити та розраховувати споживання енергії щодня, щомісяця та щороку у конкретній будівлі.

UC5: Перевіряти рівень заряду акумулятора, потужність та інші функції, пов'язані з ним. Увімкнути/вимкнути його заряд або розряд відповідно.

1.6.3 Системи оповіщення

Система оповіщення населення управляється Національним центром контролю стану катастроф та безпеки [35]. Основна його роль полягає в інформуванні населення про надзвичайні ситуації за допомогою електричних знаків тощо. У разі виникнення надзвичайної ситуації, збирає інформацію про надзвичайну ситуацію від суміжних відомств для визначення ступеня її тяжкості, зони ураження та заходів реагування на неї. Визначена інформація використовується для розповсюдження попереджувальних повідомлень серед населення через цифрове мовлення та текстові повідомлення на мобільні телефони та електричні вивіски. Можемо розробити онтологію для громадського оповіщення і розширив її на основі онтології вивісок.



Рисунок 1.6 – Система оповіщення населення

1.7 Стандартна сервісна платформа IoT oneM2M

oneM2M – це глобальна ініціатива зі стандартизації, яка охоплює вимоги, архітектуру, рішення для забезпечення безпеки, специфікації API та інтероперабельність для технологій міжмашинної взаємодії (M2M) та IoT [36]. Загальний рівень послуг, визначений oneM2M, може бути вбудований в численні об'єкти IoT, такі як кінцеві пристрої, шлюзи і сервери. Він забезпечує різні загальні сервісні функції IoT, такі як реєстрація пристроїв, управління групами, а також конфіденційність і безпека [37]. Через гетерогенну природу пристроїв IoT, їх інтероперабельність є однією з головних проблем, яку необхідно вирішити шляхом інтеграції з екосистемою oneM2M для полегшення інтеграції та розвитку послуг, додатків IoT [38] і декількох фреймворків IoT [39].

Технологія блокчейн [40] [41] [42] [43] може застосовуватися в багатьох різних сферах. За допомогою платформи рівня послуг M2M/IoT на основі блокчейну, що базується на стандартах oneM2M, можна вирішувати питання конфіденційності та безпеки, впроваджувати смарт-контракти, а також додатки IoT [10]. Оскільки рішення IoT продовжують зростати, вони стають все більш і більш поширеними, для того, щоб вдосконалити кожен аспект рішень IoT, необхідно забезпечити стандартизацію для управління рішеннями IoT, такими

як платформи тестування, аналізу та автоматизації. oneM2M забезпечує можливість використання в якості платформи автоматизації тестування для тестування на сумісність та відповідність [44]. Він відіграє життєво важливу роль у сфері безпеки. Система безпеки жінок на основі Інтернету речей є прикладом системи, яка використовує стандарт oneM2M для збору і моніторингу даних в режимі реального часу і надання допомоги в найкоротші терміни після виникнення інциденту, пов'язаного з загрозою безпеці [45]. Стандарт oneM2M інтегрується в системи моніторингу здоров'я та інші сфери, щоб проаналізувати та продемонструвати інтероперабельність фреймворків [46].

У «розумному місті» різні елементи інфраструктури, такі як «розумні будівлі», «розумні мережі» та «розумні аварійні служби», зберігають і обробляють зібрані дані за допомогою протоколів зв'язку. Через вертикальний потік даних обмін інформацією між різними сегментами не відбувається. Ці асоціації можливі лише за умови безперешкодного обміну даними між різними вертикальними секціями на рівні міста та наявності чіткого семантичного контексту для даних. Таким чином, oneM2M може забезпечити ефективний спосіб горизонтального потоку інформації і може бути використаний як стандарт для взаємодії та архівування даних [47]. Платформи IoT з підтримкою семантики розпізнають пристрої IoT, щоб зрозуміти значення даних IoT, для вирішення проблеми сумісності семантики стандарт oneM2M гарантує, що пристрої IoT від різних постачальників можуть легко взаємодіяти [9]. Сервісний рівень стандарту підтримує засоби для підключення різних пристроїв Інтернету речей, окрім технологій доступу до них, збору даних з цих пристроїв та їх обробки.

Семантика була додана як загальна функція для інтелектуальних сервісів.

oneM2M використовує ресурсно-орієнтований підхід (тобто RESTful) для всіх компонентів системи IoT. Кожна функція і всі дані з пристроїв, бекенда і систем управління представлені у вигляді ресурсу в деревовидній структурі. Ці ресурси можуть бути виявлені, отримані та оновлені відповідно до

принципів CREATE, RETRIEVE, UPDATE, DELETE та NOTIFICATION (CRUDN). Наприклад

Наприклад, загальними сутностями, які відіграють життєво важливу роль у дизайні системи oneM2M, є сутність додатку (AE) та сутність спільного сервісу (CSE). CSE – це логічна сутність у вузлі IoT з набором сервісних функцій, таких як виявлення та реєстрація. AE також є логічною сутністю, яка забезпечує бізнес-логіку додатку, наприклад, віддалений моніторинг артеріального тиску.

Вбудовування семантичних технологій в платформи IoT було застосовано як одне з основних рішень в останні роки для заміни додатків, налаштованих у вертикальній області, на загальні і горизонтальні сервіси. Було проведено кілька дослідницьких робіт для визначення семантичних функцій і сервісів в IoT. Крім того, семантичні технології були прийняті в різних вертикальних областях, які використовують платформи Інтернету речей для управління пристроями і послугами, таких як розумні міста, розумні будинки, розумне здоров'я і енергетика [48].

Крім того, oneM2M працює над розробкою стандартів, які підтримують штучний інтелект (ШІ) / машинне навчання (МН) і глибинне навчання (ГН) [49] як частину загальних функцій сервісного рівня Інтернету речей [50]. У минулому для аналізу, оцінки та прогнозування послуг ШІ/МН раніше використовувалися дані, що зберігалися локально. Оскільки вся логіка ШІ/МЛ була вбудована в додаток, було складно ефективно розробляти та керувати моделями ШІ/МЛ. Процес розробки послуги ШІ/МЛ повинен включати багато важливих етапів, таких як збір даних, управління та навчання.

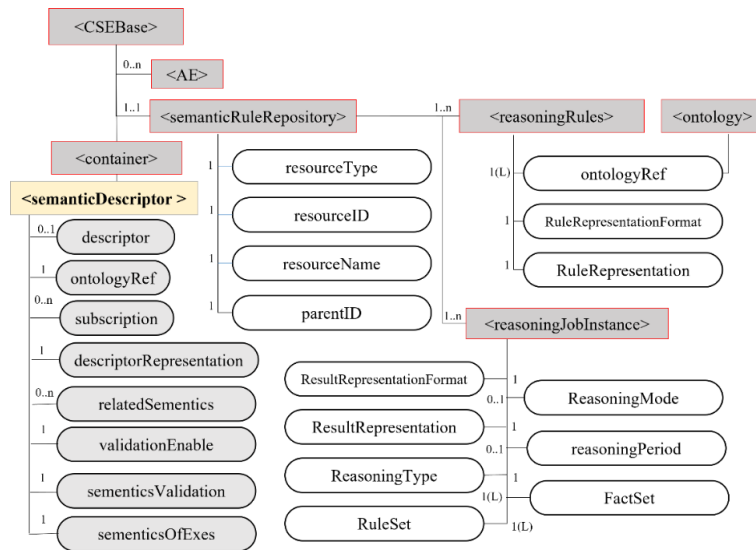


Рисунок 1.7 – Ієрархія семантичних дескрипторів ресурсів

Однак у рамках нового стандарту, розробленого oneM2M, розробники AI/ML можуть легко та ефективно керувати своїми моделями прогнозування та навчальними даними за допомогою функцій, що надаються загальним сервісним рівнем oneM2M. Про те, як інтегрувати AI/ML в сервісні платформи IoT, можна прочитати в моїй нещодавній публікації [51]. Стандарт oneM2M, який призначений для управління пристроями і даними IoT [52] [53] [54] [55] [56] [57] [58] в різних сферах, таких як розумні будинки, розумні заводи і розумні міста, також використовує семантичну технологію для інтеграції великомасштабних даних і розробки стандартів для максимізації використання даних.

Семантична анотація: Сервісний рівень oneM2M визначає ресурс `<semanticDescriptor>`, як показано на рисунку 1.7, для зберігання семантичної інформації. Додатки IoT і семантична функціональність платформи IoT можуть використовувати семантичну інформацію, що міститься в ресурсах `<semanticDescriptor>`. Семантична анотація – це процес вставки семантичної інформації в ресурси, такі як `<semanticDescriptor>`, в платформах oneM2M IoT, щоб анотований ресурс міг бути знайдений за семантикою різними додатками IoT. Крім того, цей процес анотування забезпечує більш змістовні описи даних IoT і дозволяє їм бути виявленими різними додатками IoT, пов'язаними з

традиційними системами IoT, які не надають семантичних анотацій. У системі oneM2M семантична інформація характеризується за допомогою RDF/RDFS (або OWL) у вигляді RDF-трижок. Наприклад, семантична інформація може бути пов'язана з речами IoT, які можна легко перетворити в RDF-триці. Онтологія та базова онтологія oneM2M: Опис за допомогою онтології та OWL використовуються в oneM2M для забезпечення інтероперабельності значення інформації та даних між oneM2M та зовнішніми системами.

Для цього oneM2M розробив стандартну специфікацію базової онтології oneM2M (oneM2M TS-0012 [59]), яка абстрагується від домену IoT. Базова онтологія oneM2M описує ключові класи, зв'язки та властивості, що мають відношення до додавання семантичної інформації до ресурсів oneM2M та забезпечення інтероперабельності між додатками та взаємодії з технологіями, що не належать до oneM2M. Якщо дані IoT надаються з семантичним описом відповідно до базової онтології oneM2M, система oneM2M може автоматично створювати належну структуру ресурсів за допомогою семантичної інтерпретації. Словниковий запас, наданий базовою онтологією oneM2M Base Ontology, підходить для опису можливостей і функцій пристроїв IoT і є досить невеликим, щоб добре працювати з обмеженими пристроями IoT. В oneM2M ресурс <ontologyRepository> дозволяє зберігати різні онтології посилань, включаючи базову онтологію oneM2M. Цей ресурс являє собою репозиторій онтологій, який може містити онтології, представлені як дочірні ресурси <ontology>.

Інші семантичні технології та oneM2M: oneM2M також підтримує різні семантичні технології, такі як семантична логіка та політика контролю доступу. Семантичне міркування [60] – це механізм для отримання нових знань з семантично анотованих даних IoT. Це дозволяє платформі IoT відповідати на складні запити користувачів. Набір правил для семантичного міркування визначає певну логіку для отримання нових знань на основі існуючої інформації, що зберігається та анотується в ресурсах <semanticDescriptor>. oneM2M визначає процес для безпосереднього застосування політик контролю

доступу (ACP) під час виконання семантичного запиту через потрібне сховище шляхом прийняття існуючих ACP oneM2M в RDF-трійках. У oneM2M політика контролю доступу може бути застосована до ресурсів <semanticDescriptor> шляхом додавання ACPID до ресурсів <semanticDescriptor>. Таким чином, операції, пов'язані з семантикою, які потрібно швидко виконати над RDF-трійками, збереженими в SGS, підпорядковуються цим політикам контролю доступу. Зараз система oneM2M визначає лише базові засоби для додавання політики контролю доступу до ресурсів <semanticDescriptor>.

Оптимізація та покращення продуктивності у випадку семантичних функцій, визначених в oneM2M, ще не були зроблені. У випадку політики контролю доступу, її необхідно застосовувати до різних семантичних функцій, наприклад, семантичний пошук, семантичний мешап, семантичний запит і т.д., для ефективного використання в рамках платформи IoT. Тому реалізовано семантичні сервіси в IoT-платформі на основі стандарту oneM2M.

1.8 Семантичні технології

Інструменти семантичного вебу [61], такі як загальні рамки представлення та опису даних, створюють дані в машинозчитуваній та інтероперабельній формі, тому їх можна використовувати в сфері Інтернету речей, яка базується на чудових результатах в Інтернеті. Семантичні технології, наприклад, додатки онтологій, є найкращими шляхами, які дозволяють обмінюватися інформацією, а на стороні одержувача можна проводити автоматичну інтерпретацію на основі отриманих результатів. Семантичні технології [62] можна використовувати для описових специфікацій, таких як виявлення гетерогенних ресурсів та інтеграція даних з різних доменів. Крім того, вони підтримують контроль поведінки та координацію агентів, що представляють ресурси. Загалом, семантичні технології можуть бути реалізовані за допомогою онтологій та структурованих описів даних.

Семантичні технології відіграють життєво важливу роль в управлінні даними і знаннями для розуміння контексту.

1. Структура опису ресурсів (RDF)

Опис ресурсів в Інтернеті. Читається і розуміється комп'ютерами. Написаний у форматі XML. Це компонент діяльності W3C у сфері семантичного вебу. RDF використовує веб-ідентифікатори (URI) для ідентифікації ресурсів. Він описує ресурси за допомогою властивостей і значень властивостей. RDF-заявка – це сукупність ресурсу, властивості та значення властивості. Resource Description Framework Schema (RDFS) – це мова представлення RDF для простої лексики у Всесвітній павутині. (наприклад, визначає, як писати щось).

2. Мова веб-онтології W3C (OWL)

Це мова семантичного вебу, призначена для позначення багатих і складних знань про речі, групи речей і зв'язки між речами. Онтологія – це формальний опис концептуалізації, що характеризує поняття як об'єкти, які мають свої властивості та зв'язки з іншими об'єктами. Загалом, інформація та операції в кожній системі IoT можуть бути проілюстровані за допомогою онтологій, які забезпечують універсальний словник зі структурою. Ці онтології, які можуть бути записані у вигляді загальних словників, наприклад, OWL і RDF, можуть бути використані між декількома системами за межами інтеграції онтологій або відображення (наприклад, визначення того, що писати).

3. JavaScript Object Notation (JSON)

Це легкий формат для зберігання та транспортування даних. Він часто використовується, коли дані надсилаються з сервера на веб-сторінку. Він «самоописуючий» і простий для розуміння.

4. eXtensible Markup Language (XML)

Часто використовується для розповсюдження даних через Інтернет. Вона призначена для зберігання і транспортування даних, а також для того, щоб їх можна було читати як людиною, так і машиною. Стандарт XML – це адаптивний підхід до створення інформаційних форматів та електронного

розповсюдження структурованих даних через загальнодоступний Інтернет, а також через корпоративні мережі.

5. Нотація3 (N3)

Більш відома як скорочена не–XML серіалізація RDF–моделей. Вона розроблена з урахуванням зручності для читання людиною, тому N3 є набагато ефективнішою та зручнішою для читання, ніж XML RDF нотація.

6. JSON–LD

Це полегшений формат зв'язаних даних. Він заснований на попередньому успішному форматі JSON і пропонує спосіб покращити взаємодію даних JSON на веб–рівні. Це найкращий формат даних для середовищ програмування, веб–сервісів REST та неструктурованих баз даних, таких як MongoDB та Apache CouchDB.

1.9 Висновок до першого розділу

В першому розділі кваліфікаційної роботи описано IoT–платформи та «розумні» застосунки які є ключовими елементами для створення ефективних і взаємодіючих систем у різних галузях, таких як промисловість, транспорт, охорона здоров'я, і «розумні» міста. Вони забезпечують підключення та управління великою кількістю пристроїв, що генерують значні обсяги даних. Однак, існують проблеми з інтероперабельністю та стандартизацією даних, що обмежує обмін значущою інформацією між різними системами. Використання семантичних технологій, таких як онтології та RDF, дозволяє додавати контекстуальне значення до даних, полегшуючи їх обробку та інтеграцію. Глобальні стандарти, такі як oneM2M, вже впроваджують ці технології для підвищення інтероперабельності та ефективного управління IoT–системами. Таким чином, подальший розвиток стандартизованих сервісних платформ IoT, що підтримують семантичні технології, є критичним для реалізації повного потенціалу «розумних» застосунків і забезпечення ефективного обміну даними між різними галузями.

РОЗДІЛ 2. ПРОЕКТНА ЧАСТИНА. ОНТОЛОГІЯ ТА АРХІТЕКТУРА

2.1 Використання онтологій

Повторне використання існуючої та перевіреної онтології економить час і зусилля. Багато понять, визначених в іншій онтології, можна безпосередньо імпортувати, з невеликими змінами, і застосувати до предметної області, яка вас цікавить. Розширення були розроблені на основі ядра SAREF, щоб відповідати специфічним потребам багатьох доменів. Розширення SAREF4ENER обслуговує енергетичний домен і домен «розумних» будівель S4BLDG. SAREF/SAREF4ENER пропонується як онтологія для впровадження в інтелектуальну мережу для досягнення семантичної інтеперабельності. Деякі загальні класи SAREF були розглянуті для повторного використання в онтології сонячних вивісок. Ресурси одного M2M-стандарту можуть відображатися з іншим стандартом. Наприклад, екземпляри SAREF можуть відображатися на ресурси oneM2M, як показано в Таблиці 2.1. Для відображення екземплярів SAREF на ресурси oneM2M використовується двоетапний підхід. На першому кроці класи зіставляються з класами базової онтології oneM2M шляхом визначення зв'язків між класами SAREF та oneM2M. На другому кроці екземпляри, змодельовані відповідно до тих класів SAREF, для яких існує таке визначення, також автоматично моделюються відповідно до відповідних класів базової онтології oneM2M. Правила інстанціювання oneM2M застосовуються до тих екземплярів класів SAREF, які є похідними від класів oneM2M.

Таблиця 2.1: Приклад зіставлення SAREF з ресурсами oneM2M

SAREF	Mapping	oneM2M
saref:Device	owl:equivalentClass	oneM2M:Device
saref:Service	owl:equivalentClass	oneM2M:Service
saref:Function	owl:equivalentClass	oneM2M:Function
saref:SensingFunction	owl:equivalentClass	oneM2M:MeasuringFunction
saref:ActuatingFunction	owl:equivalentClass	oneM2M:ControllingFunction
saref:Command	owl:equivalentClass	oneM2M:Command

SAREF	Mapping	oneM2M
saref:offers	owl:equivalentProperty	oneM2M:hasService
saref:hasFunction	owl:equivalentProperty	oneM2M:hasFunction
saref:represents	owl:equivalentProperty	oneM2M:exposesFunction
saref:hasCommand	owl:equivalentProperty	oneM2M:hasCommand
saref:consistsOf	owl:equivalentProperty	oneM2M:consistsOf

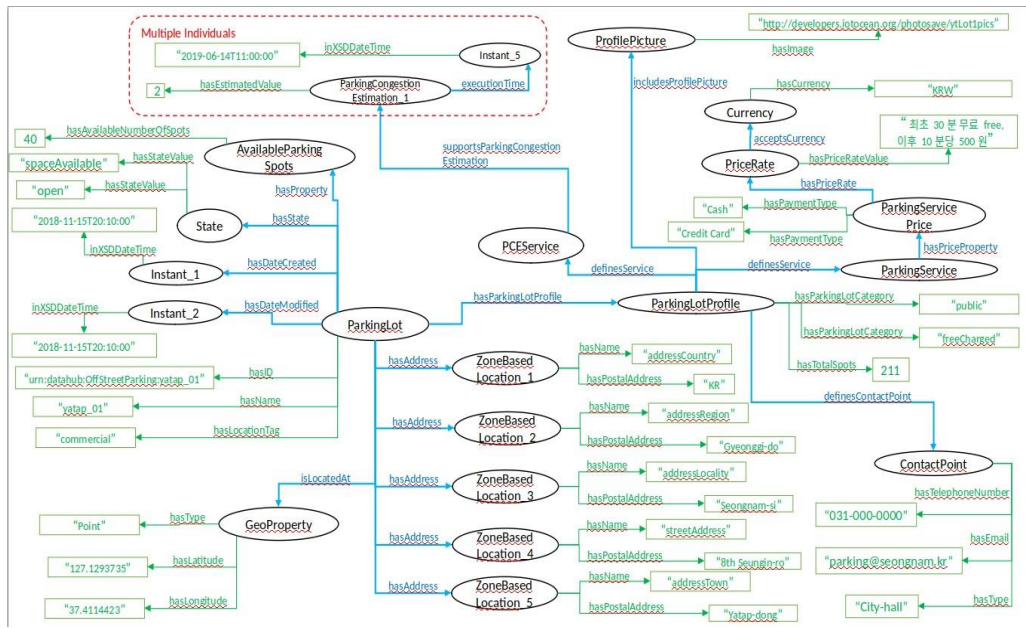


Рисунок 2.1 – Розумне місто, онтологія центру обробки даних.

2.2 Онтологія LOD та мешапів

Семантичний мешап – це процедура пошуку та збору даних з більш ніж одного джерела, виконання функцій мешапу, пов'язаних з бізнес–логікою, над зібраними даними і, зрештою, отримання значущих результатів мешапу. Наприклад, функція мешапу може генерувати метрику під назвою «індекс погоди» як результат мешапу на основі зібраних даних з декількох датчиків Інтернету речей. Рисунок 2.2 ілюструє дизайн онтології паркування та опис класів, згаданих у Додатку Б1. Рисунок 2.3 ілюструє дизайн онтології погоди та опис класів, згаданих у Додатку Б1. Рисунок 2.4 ілюструє структуру онтологій погоди та паркування та опис класів, згаданих у Додатку Б1.

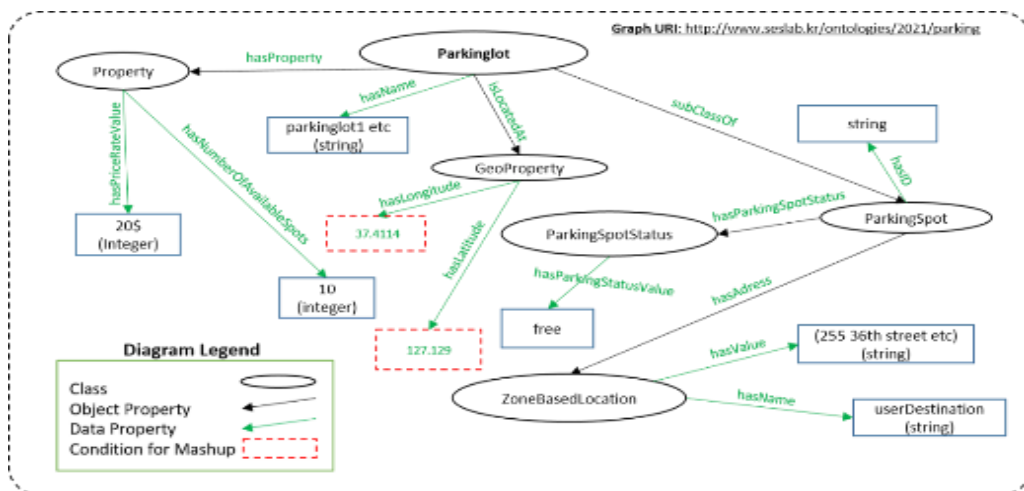


Рисунок 2.2 – Онтологія паркування для мешапу

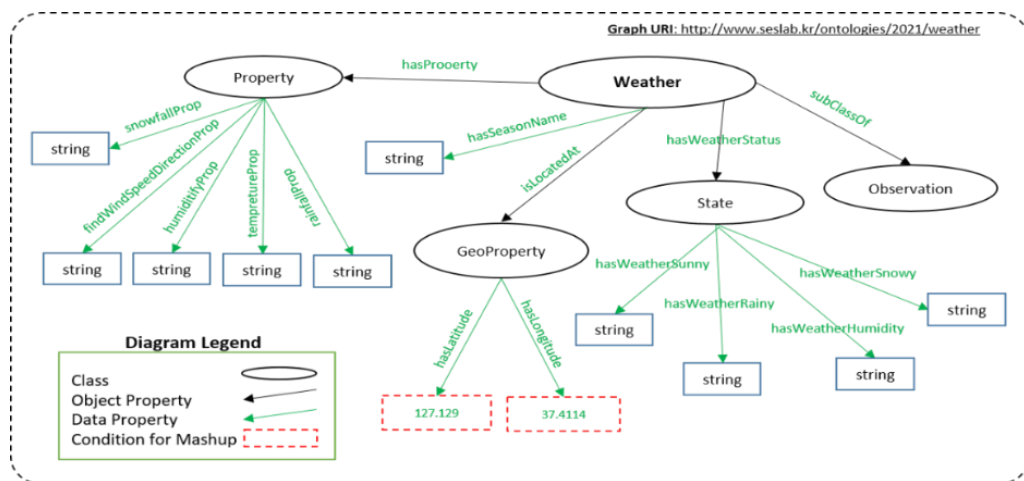


Рисунок 2.3 – Онтологія погоди для мешапу

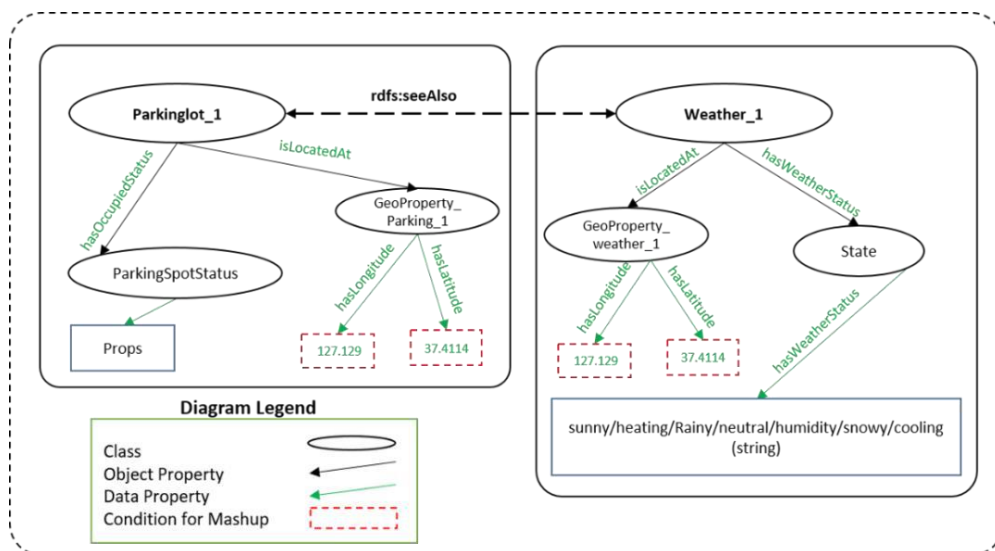


Рисунок 2.4 – Онтологія мешапу паркування та погоди (Результат)

2.3 Вивіски та онтологія BEMS

У цьому розділі наведено внесок сервісів та дизайн онтології для вивісок та BEMS з поясненнями. Рисунок 2.5 ілюструє дизайн онтології

Метою цієї роботи є управління енергетичними даними проекту «Розумне місто». Також забезпечення інтеперабельності енергетичних даних і можливість їх інтерпретації за допомогою різних зовнішніх систем, які допоможуть інтегрувати розумні будівлі та енергетичну сферу. Для цього я використовую деякі існуючі онтології, такі як The Smart Applications REference ontology (SAREF) [63], і визначаю нові класи відповідно до потреб BEMS.

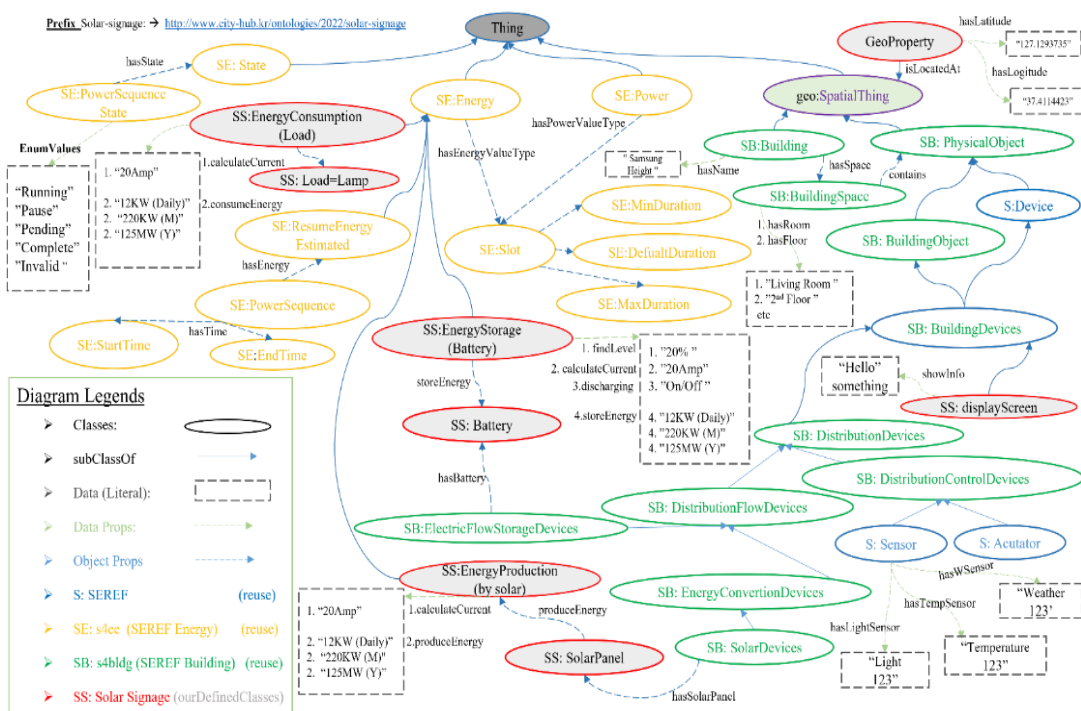


Рисунок 2.5 – Дизайн онтології сонячних вивісок

В онтології SAREF я використовував класи, які вже були створені для загальних понять, таких як будівля, потужність, пристрої, енергія або стан. У SAREF є класи первинного струму та вторинного струму, але я створив відсутні класи для загальних понять для розрахунку енергії/струму, таких як EnergyConsumption, EnergyProduction та EnergyStorage. Далі я спеціалізував ці

класи, створивши кілька підкласів, наприклад, Load, SolarPanal та Battery, які є джерелами споживання, виробництва та зберігання енергії відповідно. За допомогою спеціалізованих класів користувач може обчислювати споживання/виробництво/зберігання енергії щодня, щомісяця або щороку. Таким чином, я отримав фреймворк, який дозволив нам контролювати систему енергоменеджменту на моєму новому визначеному класі displayScreen, зручному для користувача, за допомогою розширень SAREF і додавання моїх нових класів в розумній будівлі.

Онтологія оповіщення населення походить від онтологій сонячних знаків та ВЕМС. Дизайн онтології системи громадського потепління для Південної Кореї описано на Рисунку 2.6, а повідомлення Error! Reference source not found. пояснює класи в онтології громадської безпеки.

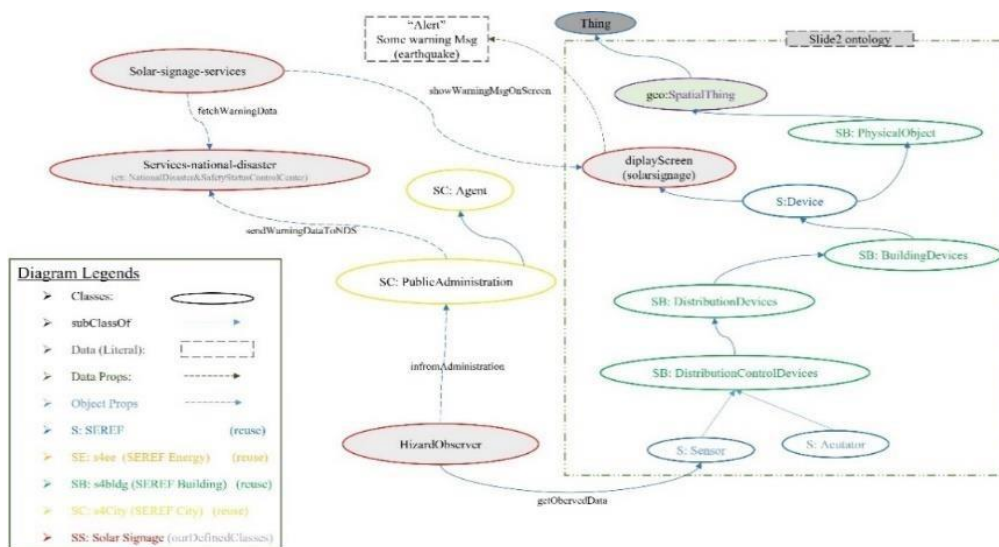


Рисунок 2.6 – Дизайн онтології публічного попередження

2.4 Архітектура та фреймворки

У цьому розділі описано побудову сервісів, архітектуру (потік реалізації та каталог коду), API (список API та відповіді на них), фреймворки (інструменти та бібліотеки), а також зберігання (інформація TDB та інтеграція і

запити SPARQL). Реалізація Сервісів розділена на дві частини: 1. Backend реалізація та 2. Реалізація інтерфейсу.

2.4.1 Реалізація бекенду

Сервер API реалізований на NodeJS (14.50), який повертає потрібні дані у форматі JSON за допомогою REST API. NodeJS – це серверне середовище з відкритим вихідним кодом. Функціональність запуску скриптів на стороні сервера створює динамічний вміст веб-сторінки до того, як сторінка буде відправлена у веб-браузер користувача.

- Архітектура потоку реалізації послуг:

Рисунок 2.7 ілюструє потік інтеграції клієнтів і серверів. У цьому потоці сервіси викликають

NodeJS-сервер для отримання даних з TDB. Потрібні дані розбираються у відповідному форматі JSON в модулі обробника JSON і повертаються клієнтам. Сервіси можна викликати як окремо, так і через клієнт дашборду. З іншого боку, дашборд інтегрований з проектами, що імплантуються в IoT, які звертаються до сервера mobius для отримання даних кожні п'ять секунд. Mobius – це серверна платформа IoT з відкритим вихідним кодом [64], заснована на стандарті oneM2M (<http://www.oneM2M.org>). Цей сервер mobius може бути інтегрований з сервером NodeJS. Вся структура коду для сервісів семантики показана на рисунку 4-2.

- Список API:

Семантичні дані повинні бути представлені в структурі формату ресурсу, щоб до них могла отримати доступ будь-яка система з підтримкою REST. Це певною мірою формулює складні функції запитів SPARQL в REST API, щоб дозволити несемантичним додаткам запитувати дані. В роботі включено запити GET, PUT, POST, DELETE, з визначеним форматом відповіді для кожного

запиту, що повертається як ресурс, замість того, щоб вбудовувати повний SPARQL-запит в API. Деякі запити SPARQL для проектів наведені в Додатку Б3. Використано різні REST API для отримання даних для сервісів. Список API наведено в Додатку Б3. Додано кілька таблиць відповідей для згаданого списку API для пояснення. Ці відповіді у форматі JSON трип-даних показані в Додатку Б3.

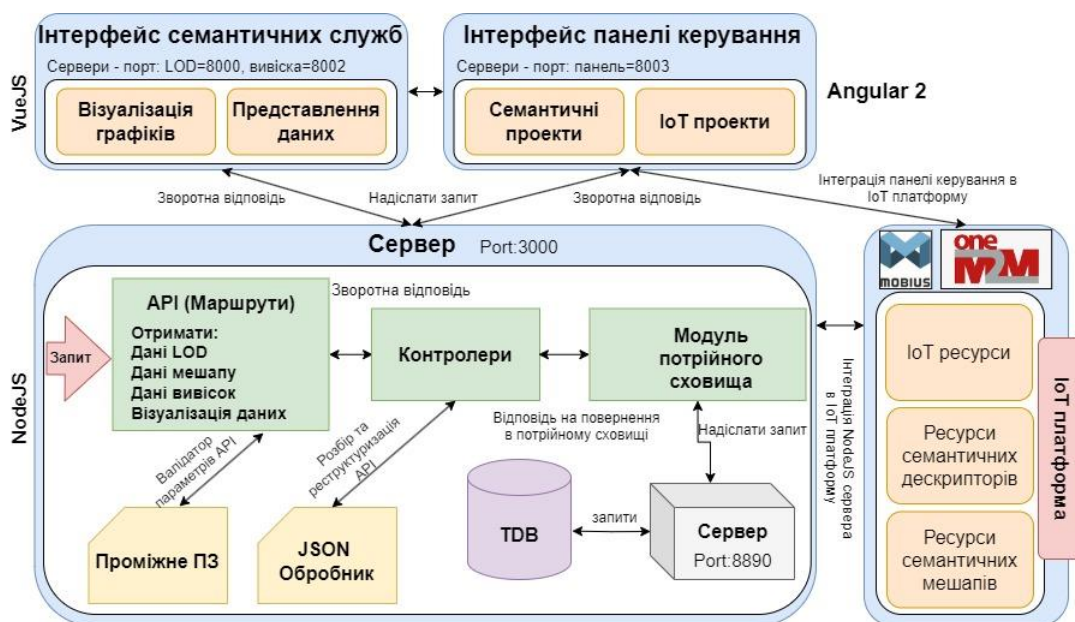


Рисунок 2.7 – Впровадження семантики/сервісів Інтернету речей

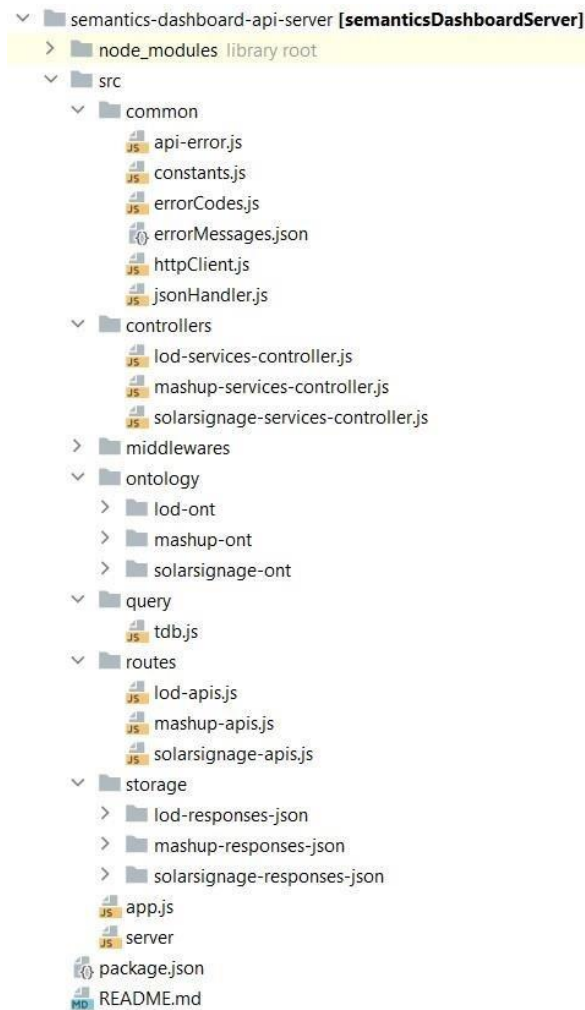


Рисунок 2.8 – Ієрархія каталогів коду на стороні сервера

- Потрійне сховище (TDB):

Сховище триплетів або RDF-сховище – це спеціально створена база даних, яка використовується для зберігання та пошуку триплетів за допомогою семантичних запитів. Тріпл – це сутність даних, що складається з суб'єкта–предиката–об'єкта, наприклад, будівля має сонячну панель (пристрій).

1. Опис віртуальних об'єктів і потоку даних для завантаження

Спочатку я визначаю онтологію за допомогою protégé, який є графічним інтерфейсом користувача для визначення онтологій. Це відкритий, безкоштовний, з відкритим вихідним кодом редактор керування онтологіями та система управління знаннями. Потім я завантажую граф онтології в потрійне сховище (TDB virtuoso) і вставляю дані.

2. Запити SPARQL

Це мова запитів для Resource Description Framework (RDF), може об'єднувати дані з різних баз даних, а також документів, механізмів міркувань або будь-чого іншого, що може передати свої знання у вигляді цільового маркованого графа. Запити SELECT, DESCRIBE, DISTICT, INSERT будуть використовуватися для цього варіанту використання відповідно до правил. У Додатку БЗ показано деякі запити SPARQL для проекту.

3. Інтеграція бекенда з магазином потрухів о Створити колекцію DAV для сервісів, наприклад, solar-signage для користувача dba з паролем dba, використовуючи кінцеву точку sparql.

- Вставка даних у графік за допомогою кінцевої точки sparql.
- Отримати дані за допомогою кінцевої точки sparql.

2.4.2 Реалізація фронтенду

Фронтенд сервісів (LOD Hub Data, Parking & Weather Mashup, Solar-Signage & BEMS) реалізовано за допомогою VueJS. Для візуалізації результатів (графіків) використовується бібліотека D3. Для дизайну використовується фреймворк Bootstrap.

Фреймворк JavaScript

VueJS: Це фреймворк JavaScript для розробки користувацьких інтерфейсів. Він розробляється на основі стандартних HTML, CSS та JavaScript/TypeScript і підтримує декларативну та компонентну модель програмування, яка допомагає ефективно розробляти користувацькі інтерфейси, як прості, так і складні.

Angular: Це фреймворк для розробки додатків і платформа для створення ефективних додатків. Angular JS базується на JavaScript, тоді як Angular 2 базується на TypeScript (TypeScript – це «смак» або «варіант» JavaScript, це надмножина JavaScript). Для розробки фронтенду можна обрати будь-який з

них, наприклад, Angular, React, Vuejs, Backbone.js тощо. Всі вони засновані на JavaScript/TypeScript.

Bootstrap: Це CSS-фреймворк з відкритим вихідним кодом, орієнтований на адаптивну, мобільну веб-розробку інтерфейсу. Він включає шаблони дизайну на основі HTML, CSS та JavaScript для типографіки, навігації, форм, кнопок та інших компонентів інтерфейсу.

2.5 Висновок до другого розділу

В другому розділі кваліфікаційної роботи демонструється важливість використання онтологій та семантичних технологій для забезпечення інтероперабельності та інтеграції даних у різних системах Інтернету речей (IoT). Повторне використання існуючих онтологій, таких як SAREF і її розширення SAREF4ENER та S4BLDG, дозволяє економити час і ресурси, адаптуючи перевірені концепції до нових доменів. Зокрема, використання SAREF/SAREF4ENER в інтелектуальних мережах сприяє досягненню семантичної інтероперабельності, що було продемонстровано у розробці онтологій для сонячних вивісок та системи енергетичного менеджменту будівель (BEMS). Семантичний мешап, який включає збір та інтеграцію даних з різних джерел для отримання значущих результатів, показує ефективність онтологій у поєднанні даних. Це дозволяє створювати нові індекси та метрики, які можуть бути корисні для міських сервісів. Розробка онтологій для вивісок та BEMS, а також інтеграція з існуючими системами IoT, сприяє ефективному управлінню енергетичними ресурсами та забезпечує можливість їхньої інтерпретації зовнішніми системами. Архітектура та реалізація сервісів на основі NodeJS і VueJS забезпечує зручність у використанні та масштабованість системи. Використання потрійного сховища (TDB) для зберігання семантичних даних та запитів SPARQL для їх обробки демонструє важливість семантичного підходу у сучасних IoT-системах.

РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА. ГРАФІЧНИЙ ІНТЕРФЕЙС

3.1 Графічний інтерфейс інформаційної панелі

Інформаційна панель реалізована для управління всіма проектами в одному місці і показує віджети для імплантованих проектів ІОТ, а також навігацію по семантичним проектам. Графічний інтерфейс панелі містить чотири основні розділи та їх підрозділи: (1) Головна сторінка панелі (графік статусу завдань проектів, список користувачів, нижній колонтитул соціальних мереж), (2) Проекти ІоТ (сторінка пульсу, віджети проектів для імплантації ІоТ), (3) Проекти LOD (сервіс LOD, сервіс Mashup, сервіс Signage) та (4) Налаштування користувача (сторінка реєстрації, сторінка входу).

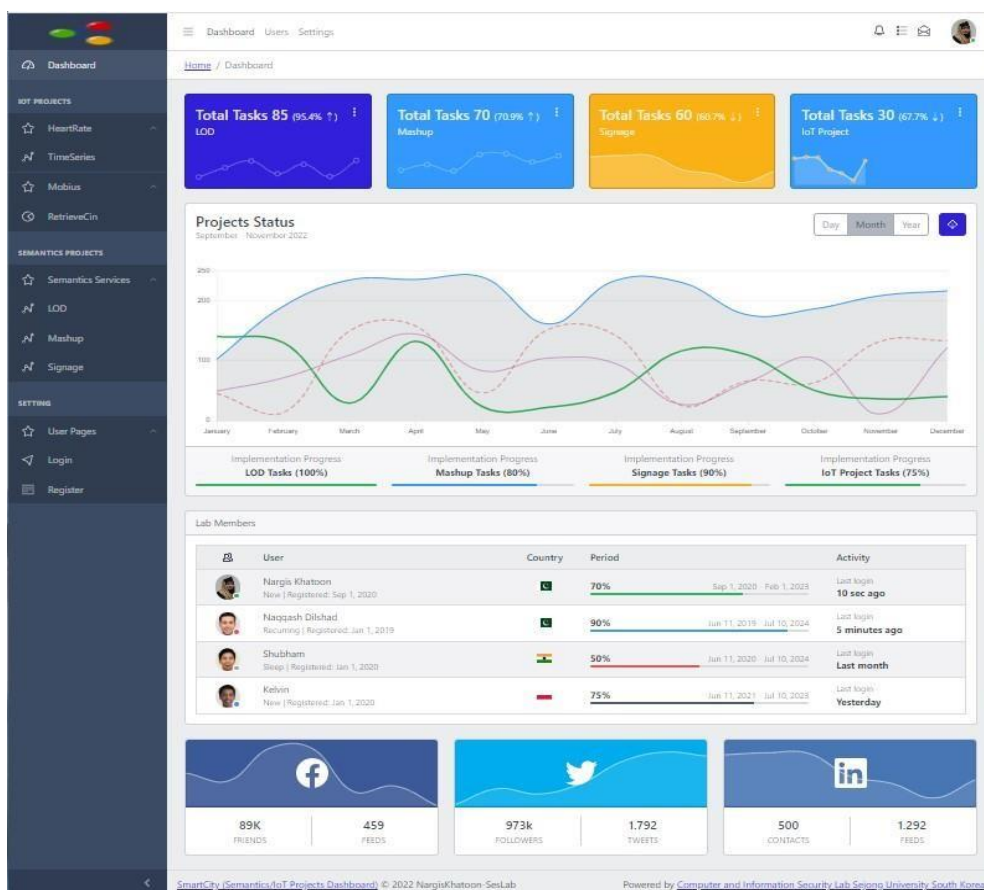


Рисунок 3.1 – Головна сторінка інформаційної панелі

Головна сторінка (Dashboard) – це головна сторінка інформаційної панелі, яка має чотири розділи, що включають віджети загальної кількості завдань у кожному проекті, статусу та прогресу проектів за днями, місяцями та роками, третій розділ показує список користувачів і, нарешті, у нижньому колонтитулі додано віджети соціальних мереж, як показано на Рисунку 3.1. Другий розділ «Проекти IoT» складається з двох компонентів (Серцевий ритм та віджети для імплантації IoT). Дані отримуються за допомогою REST API з імплантованого проекту IoT і відображаються у вигляді графіка на інформаційній панелі за годинами, днями та тижнями, як показано на рисунку 3.2. Інформаційна панель також інтегрована з імплантованими IoT-проектами. Всі віджети, такі як температура СНІ, датчик, струми Bluetooth і стан батареї тощо, додаються з імплантованих IoT-проектів. Ці дані отримуються за допомогою REST API шляхом інтеграції сервера MOBIUS до інформаційної панелі, як показано на рисунку 3.3.

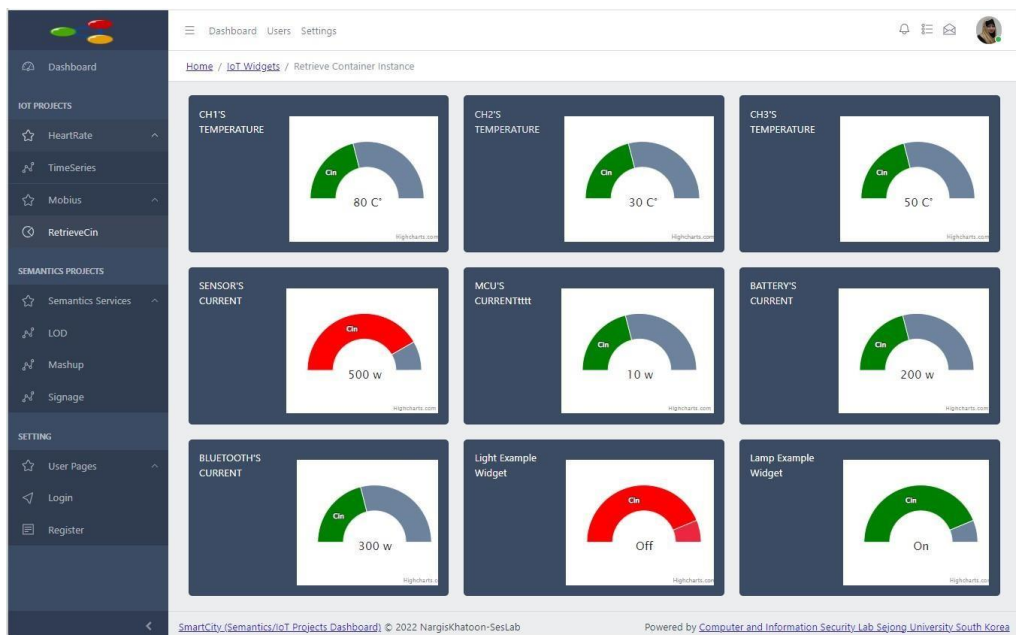


Рисунок 3.2 – Віджет IoT проектів, що імплантуються, на інформаційній панелі

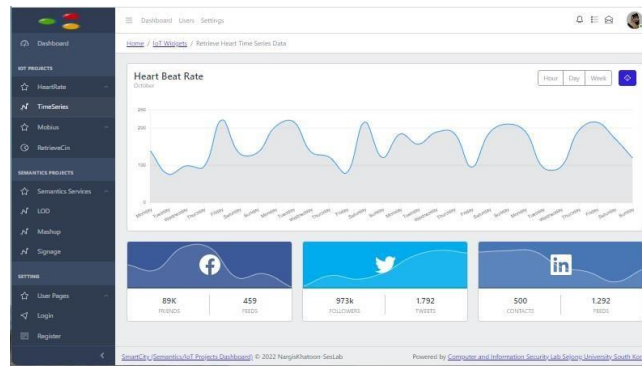


Рисунок 3.3 – Сторінка ритмів на інформаційній панелі

Третій розділ «Семантичний проект», цей розділ має такі компоненти (1. сервіс LOD дата-хабу, 2. сервіс Mashup та 3. сервіс Signage). У LOD до сервісу LOD дата-хабу можна отримати індивідуальний доступ через порт 8000. Ця послуга також інтегрована з інформаційною панеллю для зручного доступу. Детальніше про сервіс LOD та його навігацію в дашборді. У Mashup до цього сервісу можна отримати індивідуальний доступ через порт 8001. Ця послуга також інтегрована з дашбордом для легкого доступу. Деталі сервісу mashup та його навігація в дашборді. У Signage доступ до нього можна отримати окремо через порт 8002. Ця послуга також інтегрована з інформаційною панеллю для легкого доступу. Детальна інформація про сонячні табло та сервіс BEMS і його навігацію на панелі керування показана на Рисунку 3.4. Нарешті, останній розділ інформаційної панелі – це налаштування користувача. У розділі налаштувань користувача можна додати інформацію про налаштування профілю користувача.

3.2 Графічний інтерфейс LOD

Графічний інтерфейс містить чотири компоненти, а саме: (1) домашню сторінку (бічні панелі, верхній і нижній колонтитули), (2) сторінку пошуку, (3) вибір (список даних графіка) і (4) детальний перегляд (текстова вкладка, вкладка графіка і вкладка завантаження). Для компонента Головна сторінка, коли користувачі заходять на сайт, вони спочатку потрапляють на головну

сторінку, яка містить опис проекту, меню, верхній і нижній колонтитули та деяку іншу інформацію. У компоненті Search Page користувач шукає онтологію та сутності за ключовими словами. Ці ключові слова зберігаються в сховищі і надсилаються за допомогою API для отримання даних про збіги. Для компонента Select Page, спочатку на цій сторінці відображається результат пошуку за ключовим словом.

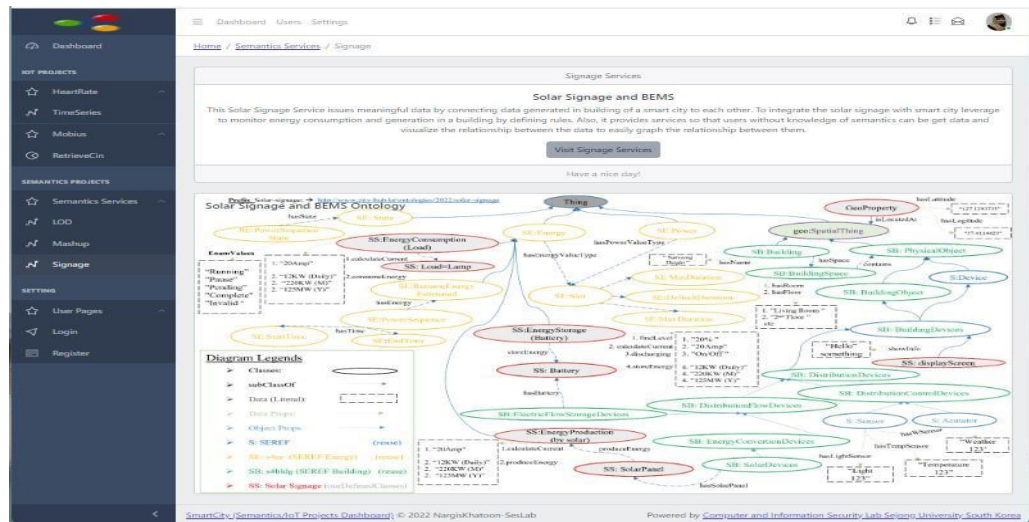


Рисунок 3.4 – Послуги вивісок на інформаційній панелі

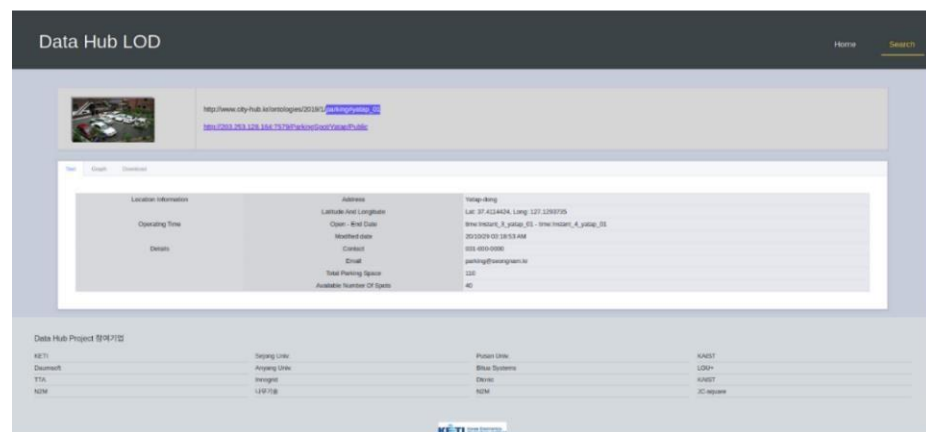


Рисунок 3.5 – Детальна текстова вкладка Data Hub LOD

Відповідь отримується у потрібному форматі JSON: спочатку відбувається парсинг і мапування результату, а потім ці дані подаються клієнту. По-друге, зображення завантажуються з локального сховища динамічно, відповідно до типу даних.

Наприклад, онтологія, паркування, якість повітря та погода. Відображення за ключовим словом зі стану. Нарешті, за допомогою кнопки LOD користувач отримує детальну інформацію про сутність/онтологію, як показано на Рисунку 3.7, а кнопка View Map наразі не реалізована.

Сторінка детального перегляду має три вкладки (текстова вкладка, вкладка графіків і вкладка завантаження). Коли користувач натискає на кнопку LOD у списку, він потрапляє на наступну (детальну) сторінку. Ця сторінка складається з двох основних частин, перша – назва сутності/онтології та їх зображення відповідно до типу (паркінг або онтологія), друга частина має 3 вкладки, які будуть детально описані за ключовим словом «паркінг» (на прикладі ключового слова «паркінг»). У вкладці «Текст» міститься основна інформація про паркування. Таблиця поділена на три категорії: Місцезнаходження (адреса, широта та журнал зі значеннями), Час (дата відкриття, закінчення та змінені дати зі значеннями) та проста інформація (контакт, електронна пошта, інформація про місця для паркування зі значеннями). У вкладці Графік, як показано на Рисунку 5–7, знаходиться візуалізація результатів, тобто графік. Граф – це сукупність сутностей, де типи та зв'язки між ними виражені вузлами та ребрами між цими вузлами. Розділ графів є найскладнішою частиною LOD. Відповідь API для графа отримуємо у вигляді трійки (предикат, суб'єкт, об'єкт) у форматі JSON. Перед тим, як показати результат, перша карта створює масив для вузлів і зв'язків графа. Композиція графа складається з наступних частин,

- Створення вузлів: Всі спільні вузли зображуються у вигляді кола.
- Створення вузлів властивостей даних: Об'єкт не має префікса; це властивість даних. Всі властивості даних зображуються квадратами.
- Створіть зовнішні вузли домену: Результат мешапу кожного графа для паркування має ще 5 потрібних об'єктів, ці 5 об'єктів показані на рисунку іншим кольором у формі кружечків, оскільки вони з іншого домену. Ім'я вузла: Кожен вузол має ім'я в центрі, яке співвідноситься з суб'єктом і об'єктом з відповіді. Створіть зв'язки: Шлях між 2 вузлами представляє зв'язок.

– Додати маркер: Стрілка вказує від вузла–суб'єкта до вузлів–об'єктів або властивостей даних.

– Мітка посилення: Шлях позначається, що представляє зв'язок між вузлами і відображається предикатом з відповіді. Забарвлюється зеленим кольором.

– Вибір кольорів: Кольори вибираються з різних категорій за допомогою фільтра за типом.

– Мітка посилення: Шлях позначений, який представляє зв'язок між вузлами і відображається предикатом з відповіді. Забарвлюється зеленим кольором.

– Методи: Основними методами роботи з графіком є збільшення, зменшення, перетягування.

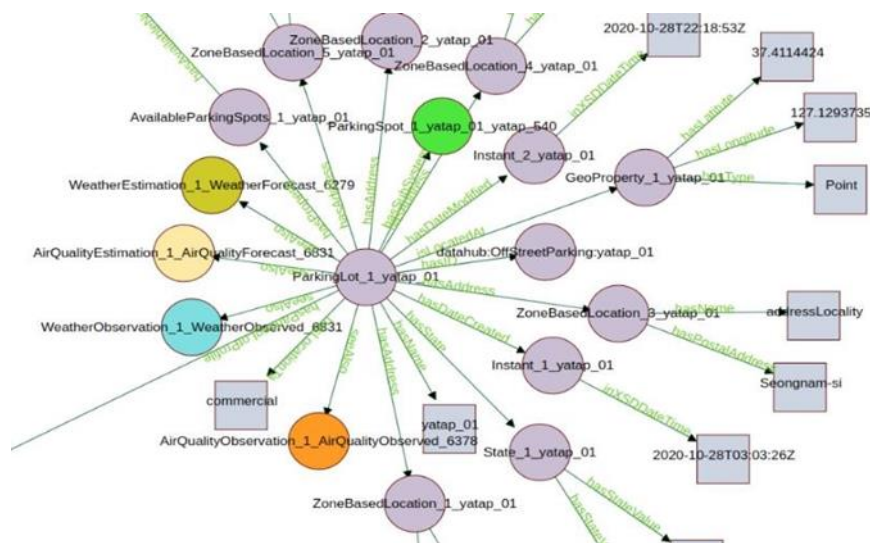


Рисунок 3.6 – Композиція графіка LOD Data Hub крупним планом

Рисунок 3.7 ілюструє склад графіка, про який йшлося вище. Граф було створено з результату з паркуванням (API з ключовим словом parking). Графік було створено на основі результату з вибором онтології. Нарешті, щодо вкладки завантаження, ця вкладка використовується для публікації файлів у різних форматах. Для файлу завантаження відповідь від API спочатку записує, а потім завантажує у потрібному форматі, наприклад, XML.

3.3 Графічний інтерфейс мешапу

У графічному інтерфейсі сервісу Mashup завантажуються дані про паркування та погоду і знаходжу результат мешапу за допомогою SPARQL-запиту до віртуозного потрійного сховища. Для цієї роботи було використано VueJS на фронтенді та NodeJS на бекенді для сервісів. Реалізовано графічний інтерфейс для вибору опції перегляду графічного представлення мешапу. Помилка! Reference source not found. показано опцію вибору, за допомогою якої можна перейти до графічного представлення результатів паркування, погоди та мешапу.

Три основні компоненти (Головна сторінка, Вибір, Результат). У випадку з Домашньою сторінкою, коли користувачі отримують доступ до веб-сайту сервісу, вони спочатку потрапляють на домашню сторінку, яка містить опис проекту на вкладці Сервіс. У віртуалізованій вкладці «Вибрати» користувач може вибрати відповідну опцію, щоб переглянути результат у віртуалізованій формі. Нарешті, на вкладці Mashup Result користувач може побачити результат об'єднання онтологій паркування та погоди, як показано на малюнку.

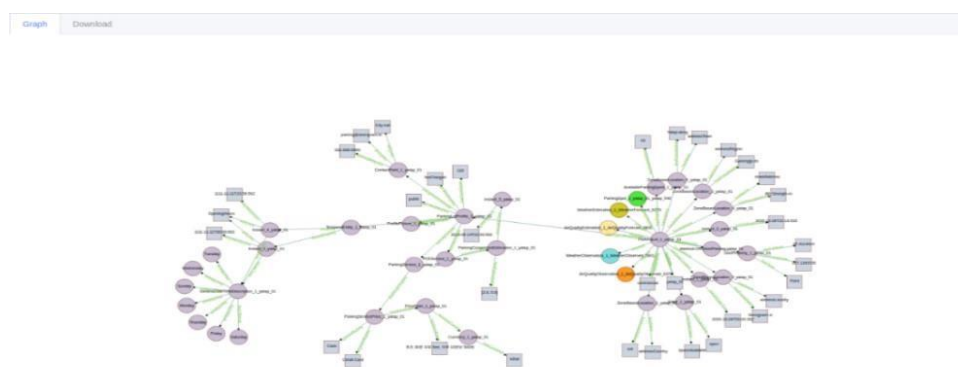


Рисунок 3.7 – Вкладка детального графіка LOD Data Hub

3.4 Графічний інтерфейс вивіски

Графічний інтерфейс містить два компоненти (1) Головна сторінка (бічні панелі, верхній і нижній колонтитули). Головна сторінка містить вступ до сервісу. Вона містить бічні панелі, верхній та нижній колонтитули. (2) Сторінка сервісу, в цьому компоненті користувач може вибрати один з розділів (1. Візуалізувати дані онтології, 2.

Секція Візуалізація даних онтології (граф) є найбільш складною частиною сонячної вивіски з точки зору дизайну. Відповідь API для графа отримується у вигляді трійки (предикат, суб'єкт, об'єкт) у форматі JSON. Перед тим, як показати результат, перша карта створює масив для вузлів і зв'язків графа. Всі спільні вузли представлені у вигляді кола. Об'єкт не має префікса; це властивість даних. Всі властивості даних зображуються квадратами. Результат мешапу кожного графа для паркування має ще 5 трійок, ці 5 об'єктів показані на рисунку іншим кольором у формі кружечків, тому що вони з іншого домену. Кожен вузол має назву в центрі, яка співвідноситься з суб'єктом та об'єктом з відповіді. Шлях між 2 вузлами представляє зв'язок. Шлях позначений міткою, яка відображає зв'язок між вузлами і співвідноситься з предикатом з відповіді. Він позначений зеленим кольором. Стрілка вказує від вузла–суб'єкта до вузлів–об'єктів або властивостей даних. Кольори вибираються з різних категорій за допомогою фільтра за типом.

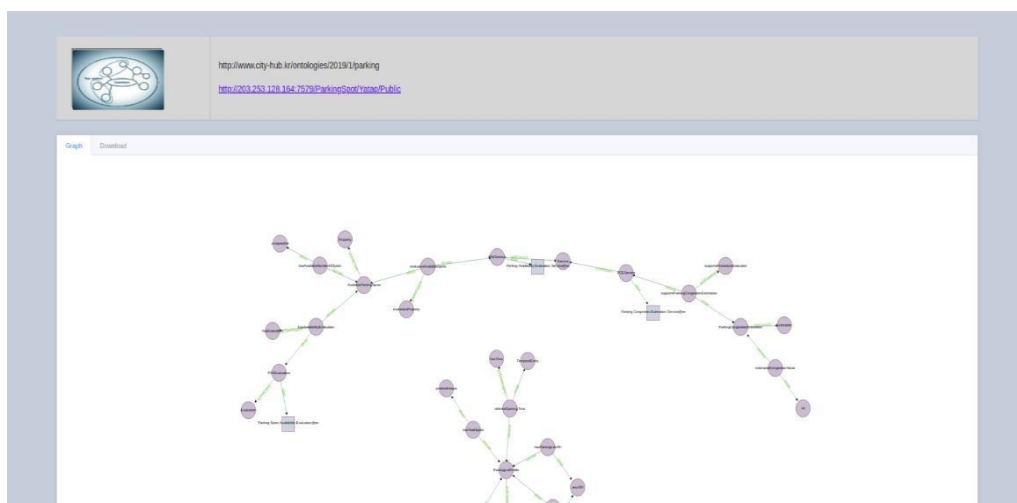


Рисунок 3.8 – Деталізація LOD хабу даних, вибравши вкладку графа онтології

Наприклад, для загальних вузлів колір сірий, для властивостей даних – лавандовий, назви вузлів – чорний, шлях – морський зелений, мітки посилань – зелений, а маркери – чорно–зелений. Деякі з основних методів роботи з графіком: Збільшення: за допомогою функції масштабування можна побачити вузол зблизька. Зменшення масштабу: весь графік можна побачити за допомогою функції зменшення масштабу. Функція перетягування використовується для утримання вузла і перетягування в будь–яке місце області графіка. Функція відпускання дозволяє утримувати вузол і відпускати його в будь–якому місці області графа, при цьому інші вузли будуть підлаштовуватися під нього. На графі Візуалізувати онтологію, користувач натискає на опцію доступу до даних на сервісній сторінці, користувач побачить вкладку Дані, яка містить основну інформацію про будівлю та пристрій. Таблиця розділена на сім категорій: інформація про будівлю (адреса, широта і довгота зі значеннями), площа будівлі (поверх, кімната тощо), інформація про пристрій (серійний номер і т.д. зі значеннями), інформація про подію (подія і час), споживання енергії, генерація енергії та батарея (струм, потужність, напруга тощо).

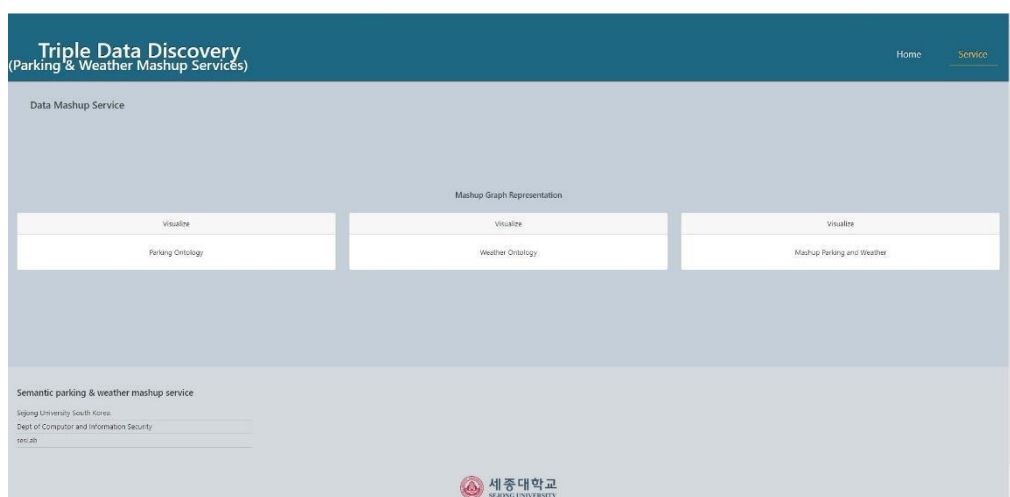


Рисунок 3.9 – Сторінка вибору мешапу

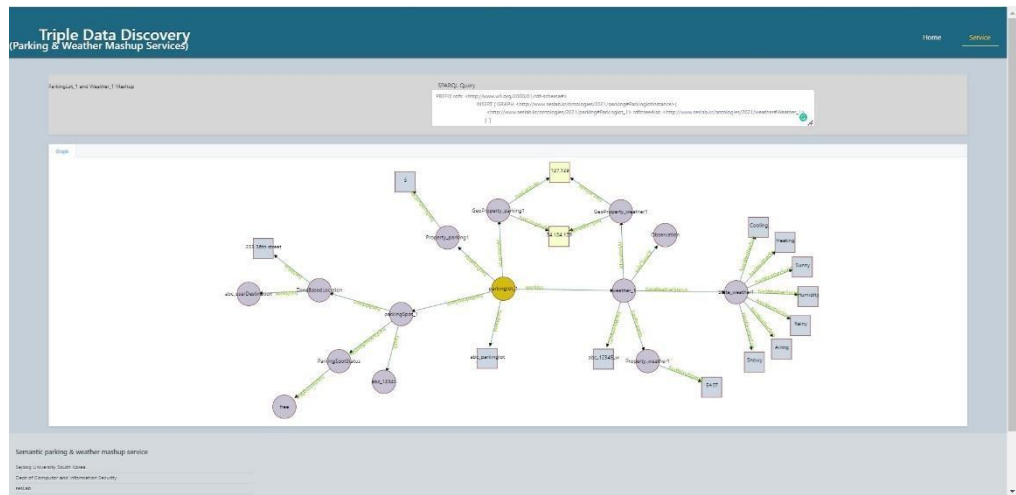


Рисунок 3.10 – Візуалізація результатів мешапу

The screenshot shows the 'Triple Data Discovery' interface displaying a table of results. The table has three main columns: 'Parking Information', 'Weather', and 'Weather Information'. The data is organized into rows, with some rows highlighted in green. The table contains various data points related to parking and weather services.

Parking Information	Weather	Weather Information
Parking Service	Address	Address
Parking Service	City	City
Parking Service	State	State
Parking Service	Country	Country
Parking Service	Zip	Zip
Parking Service	Latitude	Latitude
Parking Service	Longitude	Longitude
Parking Service	Temperature	Temperature
Parking Service	Humidity	Humidity
Parking Service	Wind Speed	Wind Speed
Parking Service	Wind Direction	Wind Direction
Parking Service	Cloud Cover	Cloud Cover
Parking Service	UV Index	UV Index
Parking Service	Air Quality	Air Quality
Parking Service	Visibility	Visibility
Parking Service	Pressure	Pressure
Parking Service	Dew Point	Dew Point
Parking Service	Relative Humidity	Relative Humidity
Parking Service	Sea Level Pressure	Sea Level Pressure
Parking Service	Mean Sea Level	Mean Sea Level
Parking Service	Altitude	Altitude
Parking Service	Time Zone	Time Zone
Parking Service	Daylight Saving Time	Daylight Saving Time
Parking Service	Local Time	Local Time
Parking Service	UTC Time	UTC Time
Parking Service	ISO 8601	ISO 8601
Parking Service	ISO 3166	ISO 3166
Parking Service	ISO 3166-2	ISO 3166-2
Parking Service	ISO 3166-3	ISO 3166-3
Parking Service	ISO 3166-4	ISO 3166-4
Parking Service	ISO 3166-5	ISO 3166-5
Parking Service	ISO 3166-6	ISO 3166-6
Parking Service	ISO 3166-7	ISO 3166-7
Parking Service	ISO 3166-8	ISO 3166-8
Parking Service	ISO 3166-9	ISO 3166-9
Parking Service	ISO 3166-10	ISO 3166-10
Parking Service	ISO 3166-11	ISO 3166-11
Parking Service	ISO 3166-12	ISO 3166-12
Parking Service	ISO 3166-13	ISO 3166-13
Parking Service	ISO 3166-14	ISO 3166-14
Parking Service	ISO 3166-15	ISO 3166-15
Parking Service	ISO 3166-16	ISO 3166-16
Parking Service	ISO 3166-17	ISO 3166-17
Parking Service	ISO 3166-18	ISO 3166-18
Parking Service	ISO 3166-19	ISO 3166-19
Parking Service	ISO 3166-20	ISO 3166-20
Parking Service	ISO 3166-21	ISO 3166-21
Parking Service	ISO 3166-22	ISO 3166-22
Parking Service	ISO 3166-23	ISO 3166-23
Parking Service	ISO 3166-24	ISO 3166-24
Parking Service	ISO 3166-25	ISO 3166-25
Parking Service	ISO 3166-26	ISO 3166-26
Parking Service	ISO 3166-27	ISO 3166-27
Parking Service	ISO 3166-28	ISO 3166-28
Parking Service	ISO 3166-29	ISO 3166-29
Parking Service	ISO 3166-30	ISO 3166-30
Parking Service	ISO 3166-31	ISO 3166-31
Parking Service	ISO 3166-32	ISO 3166-32
Parking Service	ISO 3166-33	ISO 3166-33
Parking Service	ISO 3166-34	ISO 3166-34
Parking Service	ISO 3166-35	ISO 3166-35
Parking Service	ISO 3166-36	ISO 3166-36
Parking Service	ISO 3166-37	ISO 3166-37
Parking Service	ISO 3166-38	ISO 3166-38
Parking Service	ISO 3166-39	ISO 3166-39
Parking Service	ISO 3166-40	ISO 3166-40
Parking Service	ISO 3166-41	ISO 3166-41
Parking Service	ISO 3166-42	ISO 3166-42
Parking Service	ISO 3166-43	ISO 3166-43
Parking Service	ISO 3166-44	ISO 3166-44
Parking Service	ISO 3166-45	ISO 3166-45
Parking Service	ISO 3166-46	ISO 3166-46
Parking Service	ISO 3166-47	ISO 3166-47
Parking Service	ISO 3166-48	ISO 3166-48
Parking Service	ISO 3166-49	ISO 3166-49
Parking Service	ISO 3166-50	ISO 3166-50
Parking Service	ISO 3166-51	ISO 3166-51
Parking Service	ISO 3166-52	ISO 3166-52
Parking Service	ISO 3166-53	ISO 3166-53
Parking Service	ISO 3166-54	ISO 3166-54
Parking Service	ISO 3166-55	ISO 3166-55
Parking Service	ISO 3166-56	ISO 3166-56
Parking Service	ISO 3166-57	ISO 3166-57
Parking Service	ISO 3166-58	ISO 3166-58
Parking Service	ISO 3166-59	ISO 3166-59
Parking Service	ISO 3166-60	ISO 3166-60
Parking Service	ISO 3166-61	ISO 3166-61
Parking Service	ISO 3166-62	ISO 3166-62
Parking Service	ISO 3166-63	ISO 3166-63
Parking Service	ISO 3166-64	ISO 3166-64
Parking Service	ISO 3166-65	ISO 3166-65
Parking Service	ISO 3166-66	ISO 3166-66
Parking Service	ISO 3166-67	ISO 3166-67
Parking Service	ISO 3166-68	ISO 3166-68
Parking Service	ISO 3166-69	ISO 3166-69
Parking Service	ISO 3166-70	ISO 3166-70
Parking Service	ISO 3166-71	ISO 3166-71
Parking Service	ISO 3166-72	ISO 3166-72
Parking Service	ISO 3166-73	ISO 3166-73
Parking Service	ISO 3166-74	ISO 3166-74
Parking Service	ISO 3166-75	ISO 3166-75
Parking Service	ISO 3166-76	ISO 3166-76
Parking Service	ISO 3166-77	ISO 3166-77
Parking Service	ISO 3166-78	ISO 3166-78
Parking Service	ISO 3166-79	ISO 3166-79
Parking Service	ISO 3166-80	ISO 3166-80
Parking Service	ISO 3166-81	ISO 3166-81
Parking Service	ISO 3166-82	ISO 3166-82
Parking Service	ISO 3166-83	ISO 3166-83
Parking Service	ISO 3166-84	ISO 3166-84
Parking Service	ISO 3166-85	ISO 3166-85
Parking Service	ISO 3166-86	ISO 3166-86
Parking Service	ISO 3166-87	ISO 3166-87
Parking Service	ISO 3166-88	ISO 3166-88
Parking Service	ISO 3166-89	ISO 3166-89
Parking Service	ISO 3166-90	ISO 3166-90
Parking Service	ISO 3166-91	ISO 3166-91
Parking Service	ISO 3166-92	ISO 3166-92
Parking Service	ISO 3166-93	ISO 3166-93
Parking Service	ISO 3166-94	ISO 3166-94
Parking Service	ISO 3166-95	ISO 3166-95
Parking Service	ISO 3166-96	ISO 3166-96
Parking Service	ISO 3166-97	ISO 3166-97
Parking Service	ISO 3166-98	ISO 3166-98
Parking Service	ISO 3166-99	ISO 3166-99
Parking Service	ISO 3166-100	ISO 3166-100

Рисунок 3.11 – Вивіска Дані про результати

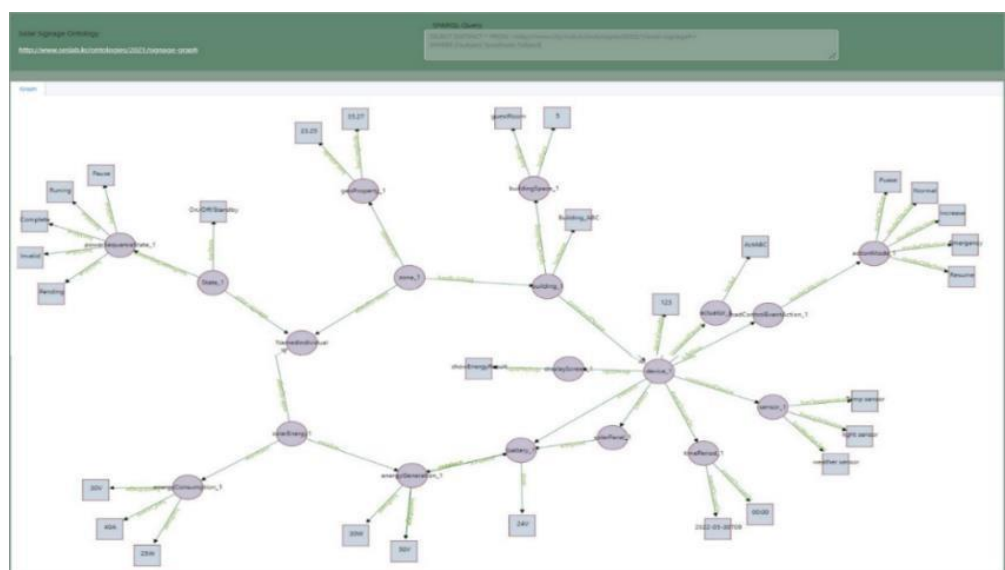


Рисунок 3.12 – Вивіска Візуалізація результатів

3.5 Висновок до третього розділу

В третьому розділі кваліфікаційної роботи проаналізовано розробку та впровадження графічних інтерфейсів для управління IoT-проектами та семантичними сервісами в рамках розумних міст. Графічний інтерфейс інформаційної панелі, який складається з чотирьох основних розділів, дозволяє ефективно управляти проектами, надаючи користувачам можливість відслідковувати статус завдань, інтегрувати дані з різних IoT-пристроїв та налаштовувати профілі користувачів. Інтерфейс надає зручний доступ до сервісів LOD, Mashup і Signage, забезпечуючи візуалізацію та керування даними через REST API та сервер MOBIUS.

Графічний інтерфейс LOD складається з кількох компонентів, що дозволяють користувачам шукати, вибирати та переглядати детальні дані онтологій. Інтерфейс забезпечує зручний доступ до семантичних даних, візуалізуючи їх у вигляді графів, що допомагає користувачам зрозуміти зв'язки між даними.

Інтерфейс для сервісу Mashup об'єднує дані про паркування та погоду, демонструючи результати об'єднання онтологій через SPARQL-запити. Використання VueJS та NodeJS забезпечує динамічну візуалізацію результатів, що полегшує аналіз даних та прийняття рішень.

Інтерфейс для вивісок включає в себе компоненти для візуалізації онтологій та даних енергоменеджменту будівель. Він дозволяє користувачам переглядати та аналізувати дані про споживання та генерацію енергії, інтегруючи їх з іншими системами розумного міста.

Розділ підкреслює важливість інтеграції семантичних технологій та графічних інтерфейсів для ефективного управління IoT-проектами. Використання стандартизованих онтологій та REST API забезпечує інтероперабельність між різними системами, полегшуючи обмін та обробку даних. Це сприяє створенню більш ефективних, взаємодіючих та зручних для користувача рішень у сфері розумних міст.

РОЗДІЛ 4. БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Організація праці при виконанні робіт в обчислювальному центрі

Тема кваліфікаційної роботи освітнього рівня «Бакалавр» присвячена аналізу стандартів інтеграції IoT–платформ та «розумних» застосунків. На даний час при проведенні наукових досліджень важко уявити платформи керування даними, інструменти візуалізації та аналітичні інструменти, де б не використовувались ресурси обчислювальних центрів. З того часу, як комп'ютери та обчислювальні центри увійшли в повсякдення людей, проводяться дослідження, як саме вони впливають на здоров'я громадян. Тому актуальним є питання організації праці при виконанні робіт в обчислювальному центрі. При цьому доцільно визначити, які фактори виробничого середовища і трудового процесу впливають на умови праці в обчислювальних центрах та яких заходів безпеки потрібно дотримуватися.

Діяльність більшості працівників сучасних обчислювальних центрів безпосередньо пов'язана з використанням комп'ютерної техніки. Комп'ютер для сучасних громадян є технічною необхідністю, як звичайні побутові пристрої. Адже ми використовуємо їх не задумуючись про шкідливість або нешкідливість, усвідомлюючи лише доступні переваги від їх наявності. Щодо комп'ютерної техніки, то існує обширний обсяг інформації про її безпечність та шкідливість.

Наказом Міністерства соціальної політики України від 14.02.2018 р. № 207 затверджено Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями (в подальшому Вимоги). Терміни у зазначених Вимогах [58] вживаються в таких значеннях:

– екранні пристрої – це електронні засоби для відтворення будь-якої графічної або алфавітно–цифрової інформації, зокрема на основі електронно–променевої трубки, рідкокристалічні, плазмові, проекційні, органічні

світлодіодні монітори та інші інноваційні розробки в галузі інформаційних технологій;

– робоче місце, або робоча станція – це сукупність обладнання, що включає екранний пристрій, який може доповнюватися клавіатурою або пристроями введення та програмним забезпеченням, що містить інтерфейси „оператор–дисплей” тощо. Зокрема, периферійні пристрої, електронні носії інформації, смартфони, модеми, друкувальні пристрої, сховища документів, робочі крісла, робочі столи або робочі поверхні „розумних” столів, а також інші необхідні елементи виробничого середовища.

Інші терміни у цих Вимогах вживаються у значеннях, наведених у Законі України «Про охорону праці».

При створенні обчислювальних центрів роботодавці повинні поінформувати працівників під розписку про умови праці та наявність на їх робочих місцях небезпечних та шкідливих виробничих факторів. Зокрема фізичних, хімічних, біологічних, психофізіологічних, які виникають під час роботи з екранними пристроями та які ще не усунуто. Також роботодавці повинні повідомити про можливі наслідки їх впливу на здоров'я працівників відповідно до вимог статті 5 Закону України «Про охорону праці».

Роботодавці повинні для працівників обчислювальних центрів забезпечити навчання і перевірку знань з питань охорони праці та безпечного використання екранних пристроїв до початку роботи з ними, а також у випадках модифікації та організації роботи обладнання. Роботодавці повинні вжити відповідних заходів, щоб забезпечити відповідність робочих місць працівників обчислювальних центрів до цих Вимог.

Під час облаштування робочого місця працівників обчислювальних центрів з екранними пристроями необхідно обирати таке обладнання, яке не створює зайвого шуму та не виділяє надлишкового тепла [58]. Рівні шуму на робочих місцях осіб, які працюють з екранними пристроями, мають відповідати вимогам Санітарних норм виробничого шуму, ультразвуку та інфразвуку ДСН

3.3.6.037–99, затверджених постановою Головного державного санітарного лікаря України від 01 грудня 1999 року № 37.

Роботодавці повинні за рахунок тривалості робочої зміни організувати внутрішні регламентовані перерви для відпочинку працівників обчислювальних центрів відповідно до Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно–обчислювальних машин ДСанПІН 3.3.2.007–98, затверджених постановою Головного державного санітарного лікаря України від 10 грудня 1998 року № 7 (далі – ДСанПІН 3.3.2.007–98).

Роботодавці повинні забезпечити за свій рахунок проведення медичних оглядів працівників обчислювальних центрів відповідно до вимог Порядку проведення медичних оглядів працівників певних категорій, затвердженого наказом Міністерства охорони здоров'я України від 21 травня 2007 року № 246, зареєстрованого в Міністерстві юстиції України 23 липня 2007 року № 846/14113. За результатами цих оглядів роботодавці за потреби повинні забезпечити виконання відповідних оздоровчих заходів.

Роботодавці зобов'язані за необхідності проводити лабораторні дослідження умов праці в обчислювальних центрах з метою виявлення шкідливих і небезпечних факторів виробничого середовища, важкості та напруженості трудових процесів. Зокрема, щодо виявлення ризиків, пов'язаних із погіршенням зору, порушенням фізичного стану, стресом та вживати заходів щодо усунення виявлених ризиків відповідно до статті 13 Закону України «Про охорону праці».

Робочі місця працівників обчислювальних центрів з екранними пристроями мають бути спроектовані так та мати такі розміри, щоб працівники мали простір для безперешкодної зміни робочого положення та рухів [59]. Для забезпечення безпеки та захисту здоров'я працівників обчислювальних центрів усе випромінювання від екранних пристроїв має бути зведене до гранично допустимого рівня. При цьому вплив на людину факторів довкілля – шуму, вібрації, забруднювачів, температури тощо, який не спричиняє соматичних або

психічних розладів, а також змін стану здоров'я, працездатності, поведінки, що виходять за межі пристосувальних реакцій з погляду безпеки та охорони здоров'я працівників.

Організація робочих місць працівників обчислювальних центрів з екранними пристроями має забезпечувати відповідність усіх елементів робочого місця та їх розташування ергономічним, антропологічним, психофізіологічним вимогам, а також характеру виконуваних робіт.

Освітлення робочих місць працівників обчислювальних центрів з екранними пристроями має створювати відповідний контраст між екраном і навколишнім середовищем з урахуванням складності та видів виконуваних робіт і відповідати вимогам ДСанПН 3.3.2.007–98.

Мікроклімат приміщень обчислювальних центрів з робочими місцями працівників з екранними пристроями має підтримуватись на постійному рівні та відповідати вимогам Санітарних норм мікроклімату виробничих приміщень ДСН 3.3.6.042–99, затверджених постановою Головного державного санітарного лікаря України від 01 грудня 1999 року № 42 (далі – ДСН 3.3.6.042–99).

Робочі столи або робочі поверхні повинні бути достатнього розміру та мати поверхню з низькою відбивною здатністю, допускати гнучкість під час розміщення екрана, клавіатури, документів та відповідного інформаційно–технологічного обладнання. Робочі крісла мають бути стійкими і дозволяти працівникам з обчислювальних центрів з екранними пристроями легко рухатися та займати зручне положення. Сидіння мають регулюватися по висоті, спинка сидіння – як по висоті, так і по нахилу. Слід передбачати підніжку для тих працівників обчислювальних центрів, кому це необхідно.

4.2 Здоровий спосіб життя людини та його вплив на професійну діяльність

Тема кваліфікаційної роботи освітнього рівня «Бакалавр» присвячена аналізу стандартів інтеграції IoT-платформ та «розумних» застосунків.. Дослідницька діяльність – це пошук нових знань або систематичне розслідування з метою встановлення фактів. Вона пов'язана з обширним переліком видів людської діяльності. Тому доцільно розглянути Здоровий спосіб життя людини та його вплив на професійну діяльність. Адже саме здоровий спосіб життя суттєво впливає на самопочуття дослідників.

Здоров'я людини ґрунтується на основі сукупності генетичних факторів, способу життя та екологічних умов. Однак певною мірою воно залежить також від свідомого ставлення людини до себе та оточуючого середовища. Здоров'я людини – стан повного соціально-біологічного комфорту коли функція всіх органів і систем людського організму врівноважені з природним і соціальним середовищем, відсутні будь-які хвилювання, хворобливі стани та фізичні дефекти. Критерій здоров'я визначається комплексом показників [60]. Однак за найзагальнішими рисами здоров'я індивідуума можна визначити як природний стан організму, що характеризується повною зрівноваженістю будь-яких виражених хворобливих змін. Слід пам'ятати, що здоров'я залежить від багатьох факторів які об'єднуються в одне інтегральне поняття – здоровий спосіб життя. Його метою є навчити людину розумно ставитися до свого здоров'я, фізичної та психічної культури, загартовувати свій організм, вміло організовувати працю і відпочинок.

До основних складових здорового способу життя належать спосіб життя, рівень культури, здоров'я в ієрархії потреб, мотивування, зворотні зв'язки, установка на довге здорове життя, навчання здоровому способу життя та психічний стан.

Спосіб життя. Має велике значення для здоров'я людини і складається з чотирьох категорій:

- економічної – рівень життя;
- соціологічної – якість життя;
- соціально–психологічної – стиль життя;
- соціально–економічної – устрій життя.

Рівень культури. Слід пам'ятати, що людина – суб'єкт і одночасно – головний результат своєї діяльності. Культура з цієї точки зору – це самосвідоме ставлення до самого себе. Однак люди дуже часто нехтують своїм здоров'ям, ведуть неправильний спосіб життя, не дотримуються режиму переїдають, курять. Тому для здоров'я потрібні знання, які увійшли б у повсякденну звичку людини [60].

Здоров'я в ієрархії потреб. Не завжди в житті людини здоров'я займає перше місце порівняно з речами та іншими матеріальними благами. У результаті це призводить до шкоди не лише своєму здоров'ю, а й здоров'ю майбутніх поколінь. Тому, здоров'я повинно займати перше місце в ієрархії потреб людини.

Мотивування. На превеликий жаль, ціну здоров'я більшість людей усвідомлює лише тоді, коли воно значно втрачено. Тільки тоді виникає прагнення вилікувати захворювання, стати здоровим [61].

Зворотні зв'язки – нерозумне і довге випробовування стійкості свого організму нездоровим способом життя (алкоголь, нікотин). Тільки через певний час спрацьовують зворотні зв'язки людини, коли вона кидає шкідливі звички, проте це вже часто запізно.

Установа на довге здорове життя. У повсякденному житті доцільно вміло мобілізувати резерви свого організму на подолання негараздів життєвого характеру, на зменшення ризику захворювань, що сприяє довголіттю.

Навчання здоровому способу життя. Джерелом навичок з цього питання є передусім приклад батьків, допомагає також і санітарна освіта. Важливим фактором, що визначає реакцію людини на екстремальну ситуацію, є її

психофізичні якості та загальний стан. Вони проявляються через чутливість людини до виявлення сигналів небезпеки, перед реакцією на ці сигнали. Показники, які зумовлюють можливості людини виявити небезпечну ситуацію та адекватно реагувати на неї, залежать від і, індивідуальних особливостей, зокрема від її нервової системи. На поведінку людини у небезпечній ситуації впливає й її психічний та фізичний стан.

Психічний стан. Сучасна людина зустрічається з багатьма факторами ризику, що негативно впливають на стан й нервової та серцево–судинної систем, знижує опірність організму [61]. При цьому виникає стресова реакція організму. Так, наприклад, психічна травма, отримана внаслідок конфлікту, виводить людину з нормального психічного стану, що може призвести до суттєвих змін у виконанні професійних функцій і загального функціонального стану. У перекладі «стрес» означає «напруження», тобто відповідь організму на поставлену перед ним проблему. Адже стрес – це сукупність загальних неспецифічних біохімічних, фізіологічних і психологічних реакцій організму внаслідок дії надзвичайних подразників різної природи і характеру, які викликають порушення функцій органів. Повне звільнення від стресу означає смерть, тому слабкий стрес є нормальним явищем у житті і потрібним для реалізації людської повноцінності. Однак якщо він інтенсивний і довготривалий, то може стати основою розвитку захворювань або зумовити смерть.

Медичні та соціологічні дослідження серед різних категорій населення показують, що люди по–різному реагують на надзвичайні ситуації. Є люди, стресостійкі до побутових негараздів, але дуже стресореактивні до сімейних проблем та невдач у коханні, інші боляче сприймають невдачі на роботі, ще інші – втрату соціального статусу.

Відомо, що в осіб до тридцяти років життєві потреби значно більші, ніж у людей старшого віку, а відтак стресові стани у них переважають.

Велике значення для розвитку стресового стану має поведінка в екстремальних умовах, наприклад аварія, кримінальна ситуація, стихійне лихо.

Неправильна поведінка у таких ситуаціях найчастіше є причиною шкідливих наслідків стресу. Вона зумовлює результат стресу більше, ніж фактори зовнішнього середовища. У цих випадках стрес може виявитись у вигляді паніки, суєти, істерики.

Стійкість організму до різноманітних стресових станів є дуже індивідуальною. Деякі люди без усіляких наслідків переносять надзвичайно складні екстремальні ситуації, ніколи не непритомніють, не втрачають сили волі, психологічної рівноваги. Інші вже при незначних екстремальних ситуаціях втрачають витримку і віру в себе.

4.3 Висновок до четвертого розділу

В четвертому розділі кваліфікаційної роботи освітнього рівня «Бакалавр» проаналізовано питання організації праці при виконанні робіт в обчислювальному центрі. Описано здоровий спосіб життя людини та його вплив на професійну діяльність. Отже, здоров'я працівників обчислювальних центрів, що працюють з екранними пристроями, є важливим аспектом, який потребує ретельного регулювання та захисту. Міністерство соціальної політики України встановило вимоги щодо безпеки та охорони здоров'я під час роботи з такими пристроями, зокрема щодо ергономіки робочих місць, рівнів шуму, освітлення та мікроклімату приміщень. Роботодавці зобов'язані інформувати працівників про можливі ризики, проводити навчання та медичні огляди, а також створювати умови для безпечної та комфортної праці. Водночас, працівники мають усвідомлювати важливість здорового способу життя та активно сприяти його підтриманню для збереження свого фізичного та психічного здоров'я.

ВИСНОВКИ

Технологія IoT, яка дозволяє великій кількості пристроїв підключатися один до одного. Однак ці пристрої, як правило, походять від різних виробників з різними стандартами продукції, що створює труднощі в інтерактивному управлінні. Крім того, інші виклики, пов'язані з IoT, включають забезпечення сумісності між даними IoT, інтерпретацію даних, отриманих пристроями IoT, і допомогу розробникам у виконанні цих завдань шляхом визначення правил. Отже, багато платформ IoT надають послуги моделювання даних та пошуку інформації. Однак використання різних моделей даних і стандартів також спричиняє проблеми інтегруєбельності. Семантичні технології спрямовані на усунення таких бар'єрів шляхом надання засобів для анування семантичної інформації до даних IoT на основі передових і поширених моделей даних. З цією метою семантична підтримка стає зростаючою вимогою і найважливішим викликом для платформ сервісного рівня IoT. Додавання семантичної допомоги розширює можливості системи IoT за допомогою розширеного значення даних, зібраних з різних пристроїв і вертикальних сховищ. У цій роботі була проведена робота над різними функціями семантики в платформі IoT для розумних сервісів IoT з використанням семантики для розумного міста, а також реалізовані сервіси виявлення семантики і візуалізації відповіді сервісів. Ця робота складається з частин, в яких розроблені онтології та сервіси для проекту розумного міста: (1) сервіс LOD, (2) сервіс Mashup та (3) сервіс вивісок та BEMS. Для управління всіма цими послугами в одному місці була розроблена інформаційна панель адміністратора. Крім того, інші проекти, що імплементуються в IoT, також інтегровані в інформаційну панель.

В першому розділі кваліфікаційної роботи освітнього рівня «Бакалавр»:

– Подано огляд існуючих IoT-платформ та "розумних" застосунків, їх структуру та основні функції.

– Розглянуто проблеми інтеграції та інтегруєбельності між різними IoT-системами і платформами.

- Висвітлено важливість стандартизації даних для ефективного обміну інформацією між різними системами.

- Проаналізовано існуючі семантичні технології та їх застосування для покращення інтероперабельності в IoT–системах.

В другому розділі кваліфікаційної роботи:

- Досліджено використання онтологій та семантичних моделей для забезпечення інтероперабельності даних у IoT–системах.

- Обґрунтовано вибір існуючих онтологій, таких як SAREF і її розширення, для застосування в інтелектуальних мережах та системах енергоменеджменту.

- Сформовано семантичні моделі для конкретних застосунків у розумних містах, таких як системи управління енергоспоживанням та вивісок.

В третьому розділі кваліфікаційної роботи:

- Розроблено інформаційну панель адміністратора для управління IoT–сервісами та моніторингу даних.

- Запропоновано підхід до інтеграції різних IoT–платформ на основі семантичних технологій.

- Спроектовано графічні інтерфейси для взаємодії з даними та управління сервісами LOD, Mashup та BEMS.

- Протестовано розроблені сервіси та інтерфейси в умовах реальних проектів розумного міста, оцінено їх ефективність та зручність у використанні.

У розділі «Безпека життєдіяльності, основи охорони праці» ... Висвітлено особливості організації праці при виконанні робіт в обчислювальному центрі та вплив способу життя людини на професійну діяльність.

ПЕРЕЛІК ДЖЕРЕЛ

- 1 Peng, Cong, Prashant Goswami, and Guohua Bai. "An ontological approach to integrate health resources from different categories of services." The Third International Conference on Informatics and Assistive Technologies for Health–Care, Medical Support and Wellbeing, HEALTHINFO, 2018–10–14~ 2018–10–18, Nice, France. International Academy, Research and Industry Association (IARIA), 2018.
- 2 Valtolina, Stefano, Luca Ferrari, and Marco Mesiti. "Ontology–based consistent specification of sensor data acquisition plans in cross–domain IoT platforms." *IEEE Access* 7 (2019): 176141–176169.
- 3 Kang, Seongju, and Kwangsue Chung. "IoT framework for interoperability in the oneM2M architecture." *Advances in Electrical and Computer Engineering* 20.2 (2020): 11–18.
- 4 Mersadier, Sylvie, and Nacira Debbiche. "Portail OATAO–Open Archive Toulouse Archive Ouverte <http://oatao.univ-toulouse.fr>."
- 5 Tange, Koen, et al. "A systematic survey of industrial Internet of Things security: Requirements and fog computing opportunities." *IEEE Communications Surveys & Tutorials* 22.4 (2020): 2489–2520.
- 6 Xu, Zhihui, et al. "Design and implementation of intelligent feeding system based–on the oneM2M." 2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE). IEEE, 2021.
- 7 Hamdan, Salam, Moussa Ayyash, and Sufyan Almajali. "Edge–computing architectures for internet of things applications: A survey." *Sensors* 20.22 (2020): 6441.
- 8 Lee, JiEun, et al. "SSF: Smart city Semantics Framework for reusability of semantic data." 2021 International Conference on Information and Communication Technology Convergence (ICTC). IEEE, 2021.

9 Guillemin, Patrick, et al. "Internet of Things Standardisation—Status, Requirements, Initiatives and Organisations." *Internet of Things*. River Publishers, 2022. 259–276.

10 Patonico, Simone, et al. "Toward the inclusion of end-to-end security in the OM2M platform." *The Journal of Supercomputing* 77 (2021): 4056–4080.

11 Cavalieri, Salvatore, and Salvatore Mulè. "Interoperability between OPC UA and oneM2M." *Journal of Internet Services and Applications* 12 (2021): 1–31.

12 Khatoon, Nargis. "Standardized smart Internet of Things (IoT) services using semantics for smart city." (2023).

13 Alrumaih, Hala, Abdulrahman Mirza, and Hessah Alsalamah. "Domain ontology for requirements classification in requirements engineering context." *IEEE Access* 8 (2020): 89899–89908.

14 Brewster, Christopher, et al. "Ontology-based access control for FAIR data." *Data Intelligence* 2.1–2 (2020): 66–77.

15 Jo, Sung-Su, et al. "Sustainable smart cities and industrial ecosystem: Structural and relational changes of the smart city industries in Korea." *Sustainability* 13.17 (2021): 9917.

16 Pliatsios, Antonios, Konstantinos Kotis, and Christos Goumopoulos. "A systematic review on semantic interoperability in the IoE-enabled smart cities." *Internet of Things* 22 (2023): 100754.

17 Phengsuwan, Jedsada, et al. "Ontology-based discovery of time-series data sources for landslide early warning system." *Computing* 102 (2020): 745–763.

18 Gu, Qi, Jian Cao, and Yancen Liu. "Csbr: A compositional semantics-based service bundle recommendation approach for mashup development." *IEEE Transactions on Services Computing* 15.6 (2021): 3170–3183.

19 Jung, Hyo-Sook, and Seong-Bin Park. "Mashup creation using a mashup rule language." *Journal of information science and engineering* 27.2 (2011): 761–775.

20 Kast, Adrian, et al. "Web of things system description for representation of mashups." *2020 International Conference on Omni-layer Intelligent Systems (COINS)*. IEEE, 2020.

21 Sun, Haofei, et al. "Smart home system design and implementation based on oneM2M." 2021 2nd International Conference on E-Commerce and Internet Technology (ECIT). IEEE, 2021.

22 Yadav, Geeta, and Kolin Paul. "Architecture and security of SCADA systems: A review." *International Journal of Critical Infrastructure Protection* 34 (2021): 100433.

23 Amaya, Mauricio Orlando Bermúdez, Octavio José Salcedo Parra, and Juan Pablo Rodríguez Miranda. "Ontological base models machine-to-machine M2M applied to the internet of things IOT." *Revista Boletín Redipe* 10.12 (2021): 148–161.

24 An, JongGwan, et al. "Synapse: Towards linked data for smart cities using a semantic annotation framework." 2020 IEEE 6th World Forum on Internet of Things (WF-IoT). IEEE, 2020.

25 Li, Han, and Tianzhen Hong. "A semantic ontology for representing and quantifying energy flexibility of buildings." *Advances in Applied Energy* 8 (2022): 100113.

26 Yu, Liang, et al. "A review of deep reinforcement learning for smart building energy management." *IEEE Internet of Things Journal* 8.15 (2021): 12046–12063.

27 Xu, Yizhe, et al. "Smart energy systems: A critical review on design and operation optimization." *Sustainable Cities and Society* 62 (2020): 102369.

28 Khatoon, Nargis. "Standardized smart Internet of Things (IoT) services using semantics for smart city." (2023).

29 Maradin, Dario. "Advantages and disadvantages of renewable energy sources utilization." *International Journal of Energy Economics and Policy* 11.3 (2021): 176–183.

30 Yu, Tai-Cheng, et al. "Visible light communication system technology review: Devices, architectures, and applications." *Crystals* 11.9 (2021): 1098.

31 Mariano-Hernández, Deyslen, et al. "A review of strategies for building energy management system: Model predictive control, demand side management,

optimization, and fault detect & diagnosis." *Journal of Building Engineering* 33 (2021): 101692.

32 Mishra, Priyanka, and Ghanshyam Singh. "Energy management systems in sustainable smart cities based on the internet of energy: A technical review." *Energies* 16.19 (2023): 6903.

33 Rathor, Sumit K., and D. Saxena. "Energy management system for smart grid: An overview and key issues." *International Journal of Energy Research* 44.6 (2020): 4067–4109.

34 Bajracharya, Manika. "An assessment of Building Energy Management Systems (BEMS) and Data-driven Life cycle engineering (LCE) in AEC projects." (2023).

35 Kim, Junic, and Kelly Ashihara. "National disaster management system: COVID-19 case in Korea." *International journal of environmental research and public health* 17.18 (2020): 6691.

36 N. Marres, "oneM2M semantics support technical report 2021," 2021, vol. 1, pp. 141–148, 2022, [Онлайн]. Доступ: <https://onem2m.org/technical/published-specifications/release-4>.

37 Dickerson, Keith, et al. "Standards for the IoT." *IoT Platforms, Use Cases, Privacy, and Business Models: With Hands-on Examples Based on the VICINITY Platform* (2021): 125–147.

38 Phung, Kieu-Ha, Hieu Tran, and Vinh Tran-Quang. "Service platform for integration of various M2M/IoT system." *Journal of Science & Technology* 144 (2020): 17–21.

39 Mante, Shubham, et al. "A Multi Layer Data Platform Architecture for Smart Cities using oneM2M and IUDX." *2022 IEEE 8th World Forum on Internet of Things (WF-IoT)*. IEEE, 2022.

40 Alizadeh, Morteza, Karl Andersson, and Olov Schelen. "A survey of secure internet of things in relation to blockchain." *Journal of Internet Services and Information Security (JISIS)* 10.3 (2020): 47–75.

41 Abdelmaboud, Abdelzahir, et al. "Blockchain for IoT applications: taxonomy, platforms, recent advances, challenges and future research directions." *Electronics* 11.4 (2022): 630.

42 Ahmed Teli, Tawseef, Faheem Masoodi, and Rameez Yousuf. "Security concerns and privacy preservation in blockchain based IoT systems: opportunities and challenges." (2020).

43 Ragab, Mahmoud, and Ali Altalbe. "A Blockchain-based architecture for enabling cybersecurity in the internet-of-critical infrastructures." *Comput. Mater. Contin* 72.1 (2022): 1579–1592.

44 Ziegler, Sebastien, et al. "F-interop-online platform of interoperability and performance tests for the internet of things." *Building the Future Internet through FIRE*. River Publishers, 2022. 603–611.

45 Tyagi, Vidhi, et al. "Architecture of an IoT-based women safety system." *Architecture* 29.5 (2020): 3670–3676.

46 Fang, Chen, et al. "A web-based and design-oriented structural health evaluation system for long-span bridges with structural health monitoring system." *Structural Control and Health Monitoring* 29.2 (2022): e2879.

47 Devendra, Deeksha, et al. "Electric vehicle charging station using open charge point protocol (ocpp) and onem2m platform for enhanced functionality." *TENCON 2021–2021 IEEE Region 10 Conference (TENCON)*. IEEE, 2021.

48 Babu, D. Vijendra, et al. "Intelligent high tech street lightning pole for smart city." *Annals of the Romanian Society for Cell Biology* (2021): 13752–13759.

49 Ragab, Mahmoud, and Abdullah Addas. "Low Complexity Encoder with Multilabel Classification and Image Captioning Model." *Computers, Materials & Continua* 72.3 (2022).

50 Chaurasia, Rohit, and Vandana Mohindru. "Unmanned aerial vehicle (UAV): A comprehensive survey." *Unmanned Aerial Vehicles for Internet of Things (IoT) Concepts, Techniques, and Applications* (2021): 1–27.

51 Khatoon, Nargis, Naqqash Dilshad, and JaeSeung Song. "Analysis of use cases enabling AI/ML to IOT service platforms." 2022 13th International Conference on Information and Communication Technology Convergence (ICTC). IEEE, 2022.

52 Dilshad, Naqqash, et al. "LocateUAV: Unmanned aerial vehicle location estimation via contextual analysis in an IoT environment." *IEEE Internet of Things Journal* 10.5 (2022): 4021–4033.

53 Dilshad, Naqqash, and JaeSeung Song. "Dual–stream siamese network for vehicle re–identification via dilated convolutional layers." 2021 IEEE International Conference on Smart Internet of Things (SmartIoT). IEEE, 2021.

54 Akbari, Younes, et al. "Applications, databases and open computer vision research from drone videos and images: a survey." *Artificial Intelligence Review* 54 (2021): 3887–3938.

55 Chen, Xi, Jun Tang, and Songyang Lao. "Review of unmanned aerial vehicle swarm communication architectures and routing protocols." *Applied Sciences* 10.10 (2020): 3661.

56 Zhao, Yiyue, et al. "Enhanced Scene Understanding and Situation Awareness for Autonomous Vehicles Based on Semantic Segmentation." *IEEE Transactions on Systems, Man, and Cybernetics: Systems* (2024).

57 Yasin, Jawad N., et al. "Unmanned aerial vehicles (uavs): Collision avoidance systems and approaches." *IEEE access* 8 (2020): 105139–105155.

58 Kumar, Ashok, and Krishan Kumar. "Multiple access schemes for cognitive radio networks: A survey." *Physical Communication* 38 (2020): 100953.

59 Liquori, Luigi, et al. "ETSI SmartM2M Technical Report 103715; Study for oneM2M; Discovery and Query solutions analysis & selection." (2021).

60 Pranavasri, V. J. S., et al. "Exploratory Study of oneM2M–based Interoperability Architectures for IoT: A Smart City Perspective." *Authorea Preprints* (2024).

61 Díaz, Vicente Hernández, et al. "Semantic as an interoperability enabler in Internet of Things." *Internet of Things*. River Publishers, 2022. 315–342.

62 Rhayem, Ahlem, Mohamed Ben Ahmed Mhiri, and Faiez Gargouri. "Semantic web technologies for the internet of things: Systematic literature review." *Internet of Things* 11 (2020): 100206.

63 Gyrard, Amelie, et al. "SAREF–Compliant Knowledge Discovery for Semantic Energy and Grid Interoperability." 2021 IEEE 7th World Forum on Internet of Things (WF–IoT). IEEE, 2021.

64 Pivoto, Diego GS, et al. "Cyber–physical systems architectures for industrial internet of things applications in Industry 4.0: A literature review." *Journal of manufacturing systems* 58 (2021): 176–192.

65 Stanko, A., Palka, O., Matiichuk, L., Martsenko, N., & Matsiuk, O. (2021, September). Smart City: A Review of Model Architecture and Technology. In 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT) (Vol. 2, pp. 273–277). IEEE.

66 Duda, O., Mykytyshyn, A., Mytnyk, M., & Stanko, A. Information technology sets formation and "TNTU Smart Campus" services network support. Proceedings of the 3rd International Workshop on Information Technologies: Theoretical and Applied Problems 2023, Ternopil, Ukraine, Opole, Poland, November 22–24, 2023.

67 Duda O., Kunanets N., Matsiuk O., Pasichnyk V., Cloud–based IT Infrastructure for "Smart City" Projects, in *Dependable IoT for Human and Industry: Modeling, Architecting, Implementation*. River Publishers, 2018. P. 389–410.

68 Безпека в надзвичайних ситуаціях. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання / укл.: Стручок В. С. Тернопіль: ФОП Паляниця В. А., 2022. 156 с.

ДОДАТКИ

Приклад моделі RDF в XML та N3

RDF model in XML notation

```
1. <rdf:RDFxmlns:rdf="http://www.w3.org/1999/02/22-rdf-  
syntaxns#"xmlns:dc12="http://purl.org/dc12/elements/1.1/">  
2. <rdf:Description rdf:about="https://en.wikipedia.org/  
wiki/Tony_Benn">  
3. <dc12:title>Tony Benn</dc12:title>  
4. <dc12:publisher>Wikipedia</dc:publisher>  
5. </rdf:Description>  
6. </rdf:RDF>
```

RDF model in Notation3

```
1. @prefix dc12: <http://purl.org/dc12/elements/1.1/>.  
2. <https://en.wikipedia.org/wiki/Tony_Benn>  
3. dc12:title "Tony Benn";  
4. dc12:publisher "Wikipedia".
```

Приклад JSON-LD

JSON-LD	
1.	{
2.	"@context": "https://json-ld.org/contexts/person14.jsonld",
3.	"@id": "http://dbpedia.org/resource/Nargis",
4.	"name": "Nargis",
5.	"born": "1989-13-08",
6.	"spouse": http://dbpedia.org/resource/Mudasir
7.	}

Приклад RDF XML

RDF	
Statement: <i>"The Student of https://sejong.ac.kr/rdf is Nargis"</i>	
The subject (Resource)	https://sejong.ac.kr/rdf
The predicate (Property)	student
The Object (Property)	Nargis, University
1.	<?xml version="1.0"?>
2.	<RDF>
3.	<Description about="https://sejong.ac.kr/rdf">
4.	<student>Nargis</student>
5.	<university> https://sejong.ac.kr </university>
6.	</Description>
7.	</RDF>
8.	</xml>

Приклад нотації об'єктів JavaScript

JSON	
1.	{
2.	"employees":
3.	[
4.	{"firstName":"Nargis", "lastName":"Khatoon"}, 5.
	{"firstName":"Anna", "lastName":"Khan"},
6.	{"firstName":"Mudasir", "lastName":"Hussain"}
7.]
8.	}

Приклад розширюваної мови розмітки EXtensible Markup Language

XML	
1.	<?xml version="1.0" encoding="UTF-8"?>
2.	<note>
3.	<to>Mudasir</to>
4.	<from>Nargis</from>
5.	<heading>Hi</heading>
6.	<body> have a nice day! </body>
7.	</note>
8.	<?xml>

Пояснення/опис класів/понять онтології мешапу

Класи/концепції	Описи
Опис класів онтології паркування та зв'язків між класами	
Parkinglot	Це основний клас онтології паркування. Парковка має одне або більше паркувальних місць
ParkingSpot	Цей клас представляє зону паркування з унікальним ідентифікатором. Вони мають властивості location та status, які використовуються для повної адреси паркувального місця та доступності паркувального місця відповідно
Property	Цей клас має багато атрибутів, пов'язаних з додатковою інформацією, такою як ціна послуг, кількість вільних місць для паркування тощо.
GeoProperty	Цей клас представляє точне місце розташування parkingSpot, виражене координатами (широта і довгота). І відіграє життєво важливу роль у мешапі
Опис класів онтології погоди та взаємозв'язків між класами	
Weather	Це основний клас онтології погоди. Цей клас має властивість назви сезону, користувач може приймати рішення на основі типу сезону.
Observation	Спостереження може мати сенсорні станції та збір даних з сенсорів, а також інші властивості об'єкта
Property	Цей клас має багато властивостей для визначення швидкості та напрямку вітру, а також інших атрибутів, наприклад, визначення вологості тощо.
State	Цей клас представляє поточний стан погоди, наприклад, дощ, сніг, сонце, вологість і т.д.
GeoProperty	Цей клас представляє точне місцезнаходження області погоди, виражене координатами (широта і довгота). І відіграють життєво важливу роль у мешапі

Пояснення/опис класів/понять онтології публічного оповіщення

Класи/концепції	Описи
HizardOberver	Клас представляє спостереження та збір даних про катастрофи з датчиків
PublicAdministration	Державне керівництво державними справами, безпосередньо відповідальне за виконавчу діяльність (Корейська метеорологічна адміністрація або місцеві органи влади)
Services–nationaldisaster	Основною функцією Національного центру контролю за станом надзвичайних ситуацій та безпеки є інформування громадськості про надзвичайні ситуації (Міністерство внутрішніх справ та безпеки).
Solar–signageservices	Отримуйте дані з сервісу Services–national–disaster та відображайте їх на вивісках
Agent	Агент, що здійснює дію в контексті міста. Агентом може бути людина, програмне забезпечення тощо.

Приклад вибірки запитів SPARQL для проекту

SPARQL queries				
<code>http://localhost:8890/sparql?query=</code>	<code>SELECT</code>	<code>DISTINCT</code>	<code>*</code>	<code>FROM</code>
<code><http://localhost:8890/DAV/solarsignageData> WHERE {? subject? predicate? object}</code>				
<code>http://localhost:8890/sparql?query=</code>	<code>SELECT</code>	<code>DISTINCT</code>	<code>*</code>	<code>FROM</code>
<code><http://localhost:8890/DAV/meshupData> WHERE {?subject ?predicate ?object}</code>				
<code>http://localhost:8890/sparql?query=</code>	<code>SELECT</code>	<code>DISTINCT</code>	<code>*</code>	<code>FROM</code>
<code><http://localhost:8890/DAV/parkingData> WHERE {?subject ?predicate ?object}</code>				
<code>http://localhost:8890/sparql?query=</code>	<code>SELECT</code>	<code>DISTINCT</code>	<code>*</code>	<code>FROM</code>
<code><http://localhost:8890/DAV/weatherData> WHERE {?subject ?predicate ?object}</code>				
<code>http://localhost:8890/sparql?query=</code>	<code>SELECT</code>	<code>DISTINCT</code>	<code>*</code>	<code>FROM</code>
<code><http://localhost:8890/DAV/solarsignageData> WHERE {? subject? predicate? object}</code>				
<code>http://localhost:8890/sparql?query=</code>	<code>SELECT</code>	<code>DISTINCT</code>	<code>*</code>	<code>FROM</code>
<code><http://localhost:8890/DAV/meshupData> WHERE {?subject ?predicate ?object}</code>				
<code>http://localhost:8890/sparql?query=</code>	<code>SELECT</code>	<code>DISTINCT</code>	<code>*</code>	<code>FROM</code>
<code><http://localhost:8890/DAV/parkingData> WHERE {?subject ?predicate ?object}</code>				
<code>http://localhost:8890/sparql?query=</code>	<code>SELECT</code>	<code>DISTINCT</code>	<code>*</code>	<code>FROM</code>
<code><http://localhost:8890/DAV/weatherData> WHERE {? subject? predicate? object}</code>				
<code>PREFIX parking: <http://www.city-hub.kr/ontologies/2022/1/parking#></code>				
<code>SELECT DISTINCT? subject? predicate? object</code>				
<code>WHERE {graph parking:yt_lot_3_spot_431_2024-03-22-11-51-06.617 {?</code>				
<code>subject? predicate? object}}</code>				
<code>PREFIX parking: <http://www.city-hub.kr/ontologies/2024/1/parking#> SELECT</code>				
<code>DISTINCT? individual</code>				
<code>WHERE</code>				
<code>{</code>				
<code>? individual rdf:type owl:NamedIndividual . {?</code>				
<code>individual rdf:type parking:ParkingLot } UNION</code>				
<code>{? individual rdf:type parking:ParkingSpot }</code>				
<code>}</code>				
<code>LIMIT 20</code>				
<code>PREFIX parking: <http://www.city-hub.kr/ontologies/2024/1/parking#> DESCRIBE</code>				
<code>parking: ParkingLot_yt_lot_1</code>				
<code>LIMIT 100</code>				
<code>SELECT DISTINCT? graph</code>				
<code>WHERE {GRAPH? graph</code>				