

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування інституту, назва факультету)

Кафедра програмних засобів

(повне найменування кафедри )

## Пояснювальна записка

до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ МЕТОДІВ  
СЕМАНТИЧНИХ РЕКОМЕНДАЦІЙ ДЛЯ ПІДТРИМКИ ОНЛАЙН  
НАВЧАННЯ

RESEARCH AND SOFTWARE IMPLEMENTATION OF METHODS FOR  
SEMANTIC RECOMMENDATIONS TO SUPPORT ONLINE LEARNING

Виконав: студент(ка) 2 курсу, групи КНТ-121м  
Спеціальності 121 Інженерія програмного  
забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

Баранік Д.С.

(прізвище та ініціали)

Керівник Степаненко О.О.

(прізвище та ініціали)

Рецензент Терещенко Е.В.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний університет «Запорізька політехніка»  
(повне найменування закладу вищої освіти)

Інститут, факультет ФКНТ  
Кафедра програмних засобів  
Ступінь вищої освіти магістр  
Спеціальність 121 Інженерія програмного забезпечення  
(код і найменування)  
Освітня програма (спеціалізація) Інженерія програмного забезпечення  
(назва освітньої програми (спеціалізації))

**ЗАТВЕРДЖУЮ**

Завідувач кафедри ПЗ, д.т.н, проф.  
С.О. Субботін  
«      »        20   року

**ЗАВДАННЯ**

**НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)**

Бараніка Дмитра Сергійовича

(прізвище, ім'я, по батькові)

- Тема проєкту (роботи) Дослідження та програмна реалізація методів семантичних рекомендацій для підтримки онлайн навчання. Research and Software Implementation of Methods for Semantic Recommendations to Support Online Learning
- керівник проєкту (роботи) Степаненко Олександр Олексійович, к.т.н., доцент,  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)  
затвержені наказом закладу вищої освіти від «04» листопада 2022 року №363
- Строк подання студентом проєкту (роботи) 1 грудня 2022 року
- Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання
- Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області.  
2. Дослідження методів семантичних рекомендацій. 3. Розробка архітектури системи. 4. Практична реалізація компонентів системи.  
5. Висновки. 6. Додатки
- Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Слайди презентації

## 6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-6 Основна частина	Степаненко О.О. доцент		
Нормоконтролер	Камінська Ж.К., асистент		

7. Дата видачі завдання “4” вересня 2022 року.

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	2–3 тижні	Розділ 1
3	Розробка та удосконалення методів, моделей й алгоритмів вирішення задачі.	4–5 тижні	Розділ 2
4	Розробка архітектури системи.	6 тиждень	Розділ 3
5	Розробка програмних компонентів системи	7–8 тижні	Розділ 4
6	Тестування та експериментальне дослідження програмного забезпечення.	9 тиждень	Розділ 5
7	Оформлення пояснювальної записки та документів до неї.	10-13 тижні	Додатки
8	Нормоконтроль та рецензування.	14–15 тижні	
9	Захист роботи.	16 тиждень	

Студент(ка)

\_\_\_\_\_ Баранік Д.С.  
( підпис ) (прізвище та ініціали)

Керівник проєкту (роботи)

\_\_\_\_\_ Степаненко О.О.  
( підпис ) (прізвище та ініціали)

## РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:  
97 с., 6 табл., 30 рис., 3 дод., 35 джерел.

ОНТОЛОГІЯ, ОНЛАЙН НАВЧАННЯ, СЕМАНТИКА,  
РЕКОМЕНДАЦІЇ, РЕЛЕВАНТНІСТЬ, JAVA, OWL, MOODLE, PROTÉGÉ.

Об'єкт дослідження – програмні засоби для онлайн навчання.

Предмет дослідження – семантичні рекомендація для підтримки  
онлайн навчання студентів.

Мета роботи – дослідження та програмна розробка системи  
семантичних рекомендація для підтримки онлайн навчання.

Матеріали, методи та технічні засоби: структурне та об'єктно-  
орієнтоване програмування, мови програмування Java, OWL, SPARQL,  
програмні середовища Protégé, Pellet, Eclipse, персональний комп'ютер з  
процесором Intel Core 2 Duo під управлінням операційної системи Microsoft  
Windows 8.

Результати. Створено програмний Windows-застосунок для  
відображення релевантних курсів навчання в залежності від мети студента.

Висновки. На підставі досліджень семантичних технологій та  
стандартів онлайн навчання розроблені онтології предметної області та  
створене програмне забезпечення.

Галузь використання – системи онлайн навчання.

## ABSTRACT

Explanatory note to the diploma qualifying work of the master: 97 pages, 29 figures, 6 tables, 30 appendixes, 35 sources.

ONTOLOGY, ONLINE LEARNING, SEMANTICS, RECOMMENDATIONS, RELEVANCE, JAVA, OWL, MOODLE, PROTÉGÉ.

Object of study is a software for online learning.

Subject of study are semantic recommendation to support online learning of students..

The purpose of the work is to research and software development of a semantic recommendation system to support online learning.

Materials, methods, and tools: structural and object-oriented programming, Java, OWL, SPARQL programming languages, Protégé, Pellet, Eclipse software environments, a personal computer with an Intel Core 2 Duo processor running the Microsoft Windows 8 operating system.

Results. The Windows software application was created to display relevant training courses depending on the student's goal.

Conclusions Based on research on semantic technologies and online learning standards, ontologies of the subject area were developed and software was created.

The field of use is a learning management systems.

## ЗМІСТ

С.

ВСТУП .....	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....	10
1.1 Сфера онлайн навчання .....	10
1.2 Розробка системи онлайн навчання .....	11
1.3 Рекомендації для підтримки онлайн навчання .....	12
1.4 Висновки за розділом.....	13
2 ДОСЛІДЖЕННЯ МЕТОДІВ СЕМАНТИЧНИХ РЕКОМЕНДАЦІЙ.....	15
2.1 Стандарти онлайн навчання.....	15
2.2 Семантичні технології .....	22
2.3 Різновиди онтологій та опис предметної області .....	27
2.4 Висновки за розділом.....	29
3 РОЗРОБКА АРХІТЕКТУРИ СИСТЕМИ.....	32
3.1 Загальна архітектура системи .....	32
3.2 Підсистема онтологій .....	38
3.3 Висновки за розділом.....	42
4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ КОМПОНЕНТІВ СИСТЕМИ .....	44
4.1 Налаштування онтології системи онлайн навчання.....	44
4.2 Рекомендації щодо семантичного змісту.....	47
4.3 Розробка додатку.....	51
4.4 Висновки за розділом.....	54
ВИСНОВКИ.....	57
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	59
Додаток А. Файл опису онтології на мові OWL .....	63
Додаток Б. Сервіс рекомендацій на основі онтологій.....	85
Додаток В. Слайди презентації.....	91

**ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК**

AD	– Adaptive Learning;
API	– Application Programming Interface;
IDE	– Integrated Development Environment;
LMS	– Learning Management System;
LTSC	– Learning Technology Systems Framework;
OWL	– Web Ontology Language;
RDF	– Resource Description Framework;
SCORM	– Sharable Content Object Reference Model;
SPARQL	– Query Language for RDF;
SWT	– Semantic Web Technology;
ПЗ	– програмне забезпечення;
СС	– семантична система;
ШІ	– штучний інтелект.

## ВСТУП

За останні кілька десятиліть розвиток галузі інформаційних технологій був дуже швидким, і різноманітні додатки, що залежать від інформаційних технологій, також різко розвиваються [1]. Вебнавчання та викладання може бути одним із поширених застосувань інформаційних технологій.

Онтології та семантичні вебсервіси є основою семантичної мережі наступного покоління. Ці майбутні технології корисні в багатьох галузях, таких як біоінформатика, бізнес-співпраця, інтеграція даних тощо. Електронне навчання також є сферою, в якій технології семантичної мережі можуть бути використані для забезпечення динамізму в методології навчання. Електронне навчання включає набір завдань, які можуть бути розробкою інструкцій, розробкою контенту, авторством, доставкою, оцінюванням, зворотним зв'язком тощо. Вебсистеми управління навчанням повинні зосередитися на тому, як задовольнити вимоги до онлайн учнів.

Багато університетів по всьому світу запустили свою онлайн систему управління навчанням. Традиційні заняття, орієнтовані на вчителя та учня, перетворюються на повторювані курси, орієнтовані на учня, де забезпечується повторне використання та порушуються часові та просторові обмеження [2]. Зараз багато учнів, як студентів, так і науковців, вірять у безперервні процеси навчання та викладання, які покладаються на всесвітню мережу. Кожен учень має свій власний темп, попередній рівень знань з курсу та вимоги. Система керування навчанням на основі семантичної мережі, яка є адаптивною з точки зору пошуку стилю учня та динамічно надає навчальні об'єкти чи послуги відповідно до стилю учня, є вимогою сьогодення. Учні розуміють і засвоюють вивчені концепції та застосовують їх у практичних рішеннях. Найновіші та майбутні системи навчання повинні враховувати ці індивідуальні шляхи навчання для їх розробки та впровадження.



Динамічна композиція вмісту електронного навчання та послуги електронного навчання, а також семантичні запити для навчальних матеріалів і конструювання курсів, яким уподобаються учні – це функції, які мають бути реалізовані в будь-якій адаптивній системі керування електронним навчанням. Саме тому, розробка та дослідження семантичних рекомендацій для підтримки системи онлайн навчання є актуальною задачею.

Метою даної магістерської роботи є дослідження та програмна розробка семантичних рекомендацій системи онлайн навчання.

Для досягнення даної мети треба вирішити наступні завдання:

- провести аналіз предметної області онлайн навчання;
- дослідити переваги та недоліки сучасних систем онлайн навчання;
- дослідити методи семантичних рекомендацій;
- розробити модель семантичної системи онлайн навчання;
- розробити базу знань предметної області;
- дослідити результати роботи семантичних рекомендацій.

Предметом дослідження кваліфікаційної роботи магістра є система онлайн навчання.

Практична цінність роботи полягає в тому, що на підставі розробки семантичного опису предметної області розроблено адаптивну систему онлайн навчання.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Сфера онлайн навчання

Програма електронного навчання вимагає використання онлайн-інструментів або методів для доступу до навчальних програм поза аудиторною практикою. У загальних випадках це стосується будь-якого ступеня, програми чи курсу, які повністю доступні онлайн. У той час як технологія розвивається, вона вдосконалюється та трансформується з появою електронних гаджетів, таких як настільні комп'ютери, ноутбуки, мобільні телефони, планшети та інші подібні технології. Знаннями можна ділитися або бути доступними в будь-який час і в будь-якому місці, 24x7 через Інтернет у рамках електронного навчання, тим самим підвищуючи якість освітніх підходів. Така інтерактивна навчальна програма на базі Інтернету містить величезну кількість даних щодо тексту, відеозаписів, презентацій Power-Point тощо з конкретною навчальною програмою [3].

Методи інтелектуального аналізу даних спрямовані на виявлення нових тенденцій шляхом обробки великих обсягів даних. Протягом останнього десятиліття методи інтелектуального аналізу даних переважно використовувалися в бізнес-додатках, а тепер вони також широко використовуються в освітніх системах [4]. У цей час можна навіть розробити інтелектуальний аналіз даних для виділення інформації за допомогою систем електронного навчання шляхом збору даних, наданих користувачами. Однак через часте збільшення кількості вебкористувачів і їхню необхідність отримувати інформацію за метою пошуку в розробникам вебсторінок важко зробити доступними релевантні дані результатів пошуку. Щоб виконати певні завдання, дані слід надавати зрозуміло. Пан Тім Бернерс Лі, голова консорціуму Всесвітньої павутини (W3C), оголосив про нову перспективну роль вебтехнології під назвою Semantic Web Technology. Семантичний веб – це дійсно сучасна платформа Всесвітньої павутини, яка охоплює не лише

вебсторінки, але й формалізовану семантику [5]. Ця семантична мережа надає інформацію в чітко визначеному значенні, дозволяючи комп'ютерам, а також окремим особам співпрацювати ефективніше. Головна передумова використання семантичної мережі полягає в тому, щоб делегувати багато завдань з пошуку комп'ютерам [6].

Рух до модернізації пронизав систему електронного навчання на основі Інтернету, яка використовується для надання вищої освіти через використання Всесвітньої павутини. Електронне онлайн навчання, яке зараз практикується, переважно надає навчальні дані на основі цифрової інформації – це, по суті, статичні дані, які відображаються за допомогою веб програмного забезпечення або встановлені на оптичному запам'ятовуючому пристрої за допомогою незалежних програм для настільних комп'ютерів. Однак у новому розвитку інформаційних технологій інформація надаватиметься не лише в текстовому чи мультимедійному форматах, але й буде представлена в інтелектуальний спосіб, схожий на людського вчителя, відповідно до рівня знань онлайн-учня. Таким чином, інструменти інтелектуального аналізу даних дозволяють оброблювати дані та робити висновки у майбутньому [7].

## **1.2 Розробка системи онлайн навчання**

Проблеми, які виникли у зв'язку з онлайн навчанням:

- розробка фреймворку інтелектуального мобільного сервіса для онлайн-системи електронного навчання, орієнтованої на семантичну мережу;
- за необхідності передавати якомога більше інформації мобільному сервісу;
- побудувати систему онлайн навчання, яка є гнучкою, інтерактивною, а також структурованою;
- створити набагато більш налаштовану, а також складну онлайн-систему електронного навчання;

- використання спеціаліста з рекомендацій, щоб запропонувати відповідну інформацію для вивчення обраних курсів;
- правильна категоризація інформації про спроможність користувача до навчання.

Недоліки дизайну для розробки ефективної системи онлайн навчання такі:

- візуальне представлення тем повинно надаватися системою онлайн навчання;
- перш ніж використовувати конкретну онлайн-тему, можна провести статистичний аналіз, а також надати певні точки зору на цю тему;
- система повинна співпрацювати та експериментувати з новими знаннями з цього питання;
- для взаємодії з користувачем можна використовувати штучний інтелект із розпізнаванням голосу;
- система повинна мати здатність до навчання;
- слід надати відповіді з внутрішньої бази даних, Wiki, Інтернету, попереднього спілкування.

### **1.3 Рекомендації для підтримки онлайн навчання**

Дослідники адаптивного електронного навчання аналізують і розробляють адаптивні методи, які забезпечують кращий освітній досвід для студентів. Вчені пропонують студентам точний та персоналізований контент ефективним способом [8], який може дозволити коригувати зміст курсу на основі останньої успішності студента. Ця техніка дозволяє учневі пропускати непотрібні та/або зайві навчальні дії, забезпечуючи автоматизовану та персоналізовану підтримку для учня [9]. Студенти з різною освітою та різними стилями навчання є головною проблемою для систем електронного та мобільного навчання. Ці системи пропонують персоналізовані розділи курсу для задоволення освітніх потреб різних студентів.

Протягом останніх десятиліть різноманітні вчені досліджували можливість зміни моделі навчання зі звичайної універсальної моделі навчання на більш адаптивну та індивідуальну модель навчання.

Більшість методів розраховують оптимальний шлях навчання залежно від характеристик у профілі учня, щоб зробити курс більш персоналізованим. Однак ми не бачили жодної техніки, яка б динамічно оновлювала профілі за допомогою семантичної мережі для обміну інформацією між навчальними закладами [10].

Профіль студента містить інформацію про студента, таку як ім'я, прізвище, адреса, розділи курсу, які були завершені, і оцінки цих розділів курсу.

Цю модель можна застосувати для мобільного навчання та електронного навчання в громадських коледжах, а також у звичайних курсах випускників або підготовчих курсів. Студенти можуть отримати вигоду та персоналізувати свій досвід коледжу та виконання своїх вимог. Однак ця модель не стосується студентів K-12, оскільки вони виходять за межі нашого дослідження.

Адаптивне навчання (Adaptive Learning, AL)– це навчальний метод, який допомагає учням у процесі навчання відповідно до їхніх потреб. Крім того, AL допомагає викладачам передавати своїм студентам зміст курсу в індивідуальний спосіб на основі здібностей студентів і досвіду. Крім того, з точки зору розробників, адаптивне навчання — це техніка, яка використовує комп'ютери та інші ресурси для надання кращого досвіду навчання.

Одним із найскладніших завдань для адаптивного мобільного навчання є створення адаптивного курсу.

#### **1.4 Висновки за розділом**

Програма електронного навчання вимагає використання онлайн-інструментів або методів для доступу до навчальних програм поза

аудиторною практикою. Електронне онлайн навчання, яке зараз практикується, переважно надає навчальні дані на основі цифрової інформації. Дослідники адаптивного електронного навчання аналізують і розробляють інтелектуальні методи, які забезпечують кращий освітній досвід для студентів. Існуючі методи обробки інформації пропонують студентам точний та персоналізований контент на основі пошукових запитів студента.

Необхідність розробки адаптивної системи онлайн навчання з семантичними рекомендаціями полегшить процес пошуку необхідної інформації в системі та формування навичок студентів в залежності від вже отриманих знань.

## **2 ДОСЛІДЖЕННЯ МЕТОДІВ СЕМАНТИЧНИХ РЕКОМЕНДАЦІЙ**

### **2.1 Стандарти онлайн навчання**

Правильне розуміння теорії стилю навчання потрібне для розробки робочої системи AL. Моделі стилю навчання були запропоновані різними дослідниками. Це модель VAK, модель Колба, модель Хані та Мамфорда та модель FSM-Felder-Silverman. У цих моделях було дотримано або спільний, або автоматичний підхід. Стили навчання базуються на результатах досліджень когнітивної психології щодо обробки інформації, активного навчання та структури інформації [11].

Перша ідея надання персоналізованого навчання офіційно представлена Кумі [12]. Він впровадив адаптивну систему управління навчанням на основі мультимедіа, в якій комп'ютер пропонує індивідуальний контроль та інтерактивність. Пристосування контенту системи навчання, щоб задовольнити індивіда, є складним процесом. Набіла запропонував метод виявлення стилю навчання, за яким спостережувана поведінка та дії учнів відстежуються та перетворюються на стиль [13]. Фарман Алі Хан та інші відстежували прогрес знань учнів, надавали оцінку та підтримували учня, пропонуючи підказки, допомогу та відгуки в запропонованому ними інструменті/структурі навчання та оцінювання [14]. Вони обговорювали, що емоційний стан учнів відіграє важливу роль в ефективності електронного навчання.

Данг і Флореа зосереджуються на інтелектуальних агентах, які можуть надати учням особистих помічників для виконання навчальної діяльності відповідно до їхніх стилів навчання та рівня знань [15]. Інтерес учня, цілі та його попередні знання з цього предмету є основними проблемами в будь-якій системі онлайн навчання. В системі семантичних рекомендацій, орієнтованому на студента, надається більше значення даним студента ніж викладача. Обґрунтування AL полягає в тому, що врахування індивідуальних відмінностей учнів (щодо рівня знань, цілей, стилю навчання,

когнітивних здібностей тощо) є корисним для учня, що призводить до підвищення ефективності навчання та/або успішності учня [1].

Поведінку учнів онлайн неможливо передбачити заздалегідь, і вона змінюється з часом. Wolf перевірів, чи покращують навчальні об'єкти, засновані на мультимедіа, слабку успішність учнів чи ні [16]. Melody Siadaty і Fattaneh Taghiyareh використовували Jackson's Learning Styles Profiler (LSP) для моделювання стилів навчання учнів [17, 18]. Джексон визначив п'ять стилів навчання: шукач відчуття, цілеспрямований досягаючий, емоційно інтелектуальний успішний, глибоко навчальний успішний і сумлінний досягаючий [19].

У нашому проєкті модуль навчання використовує техніку моделі Sabine Graf і Jackson Learning Style Profiler і використовує семантику та базу знань Ontologies для отримання стилів навчання в системі семантичних рекомендацій [20]. Ми вважаємо за краще використовувати цю модель, оскільки учень підійде до будь-якої з цих категорій емоційний, успішний, навчаний [21].

Звичайне онлайн навчання – це в основному статичний веб-інформаційний пакет або збірка автономних персональних комп'ютерних програм, що надає лише електронні дані, що стосуються освіти. Сучасна концепція інформаційних систем може зробити ще один крок у тому, щоб вона класифікувала учнів у навчанні відповідно до їхнього рівня знань.

Електронне навчання онлайн є дуже складним напрямом дослідження, тому IEEE Learning Technology Systems Framework (LTSC), Sharable Content Object Reference Model (SCORM), Blackboard, Moodle, Schoology та Sakai є одними з найпоширеніших онлайн-стандартів. Далі розглянемо ці стандарти більш детально.

IEEE Learning Technology Systems Framework (LTSC) В основному, як показано на рис. 2.1, ця модель IEEE Learning Technology Systems Framework” складається з 6 елементів [10]. Сутність учня – це дійсно графічний інтерфейс користувача (GUI) для учнів, які прагнуть створити



інформацію для вебсистеми електронного навчання. Модуль «Learning Resources» містить інформацію з каталогу, яка може визначити, як вміст було доставлено до студента через компонент розподілу. Найважливішою особливістю цієї структури є подальше зростання, яке постійно перевіряє поведінку учнів, прогрес у навчанні шляхом оцінювання тестів, підсумкового оцінювання або будь-якого іншого типу тесту, який проводить система. Заключний розділ, дані студентів, зберігає особисті дані користувача, включаючи історію та поточні дані, а також професійні та академічні дані.

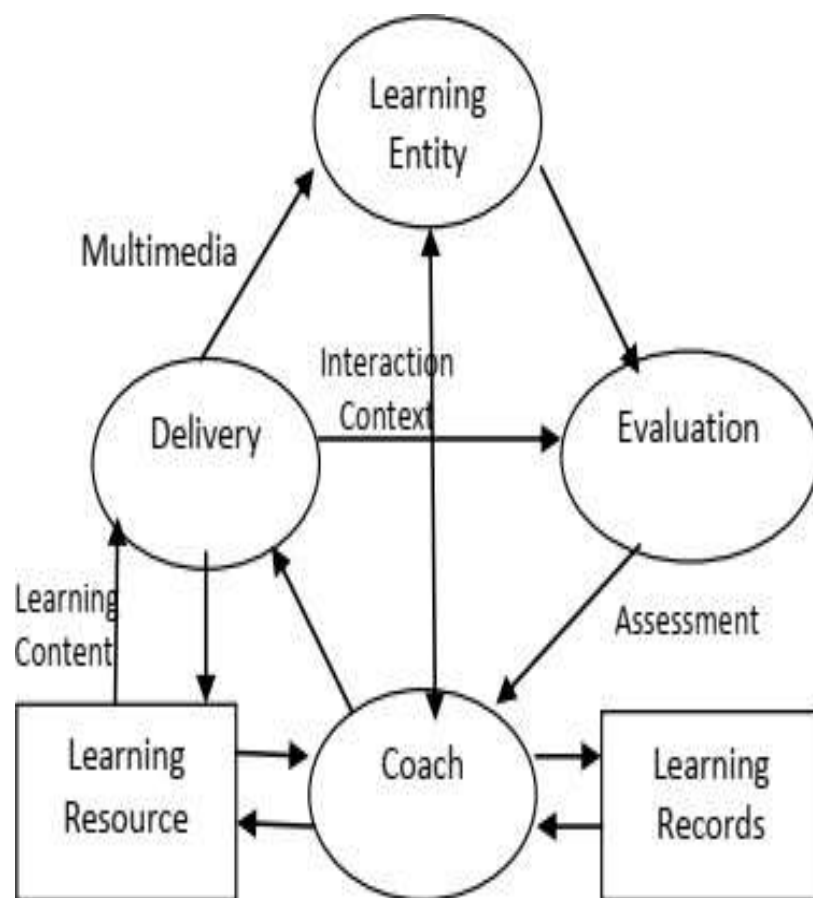


Рисунок 2.1 - IEEE Learning Technology Systems Framework [22]

Sharable Content Object Reference Model (SCORM). Архітектура цієї еталонної моделі є досить поширеною серед онлайн-систем електронного навчання на базі Інтернету, як показано на рис. 2.2. Додатковим сервісом, який надає така модель, є система керування контентом. Постачальник платформи має гнучкість, щоб зробити доступним

специфічний для користувача вміст навчального матеріалу. Він також має адаптер API, який може забезпечити інтерфейс на рівні програми, який не залежить від мов програмування. Він надає інформацію лише через веббраузер.

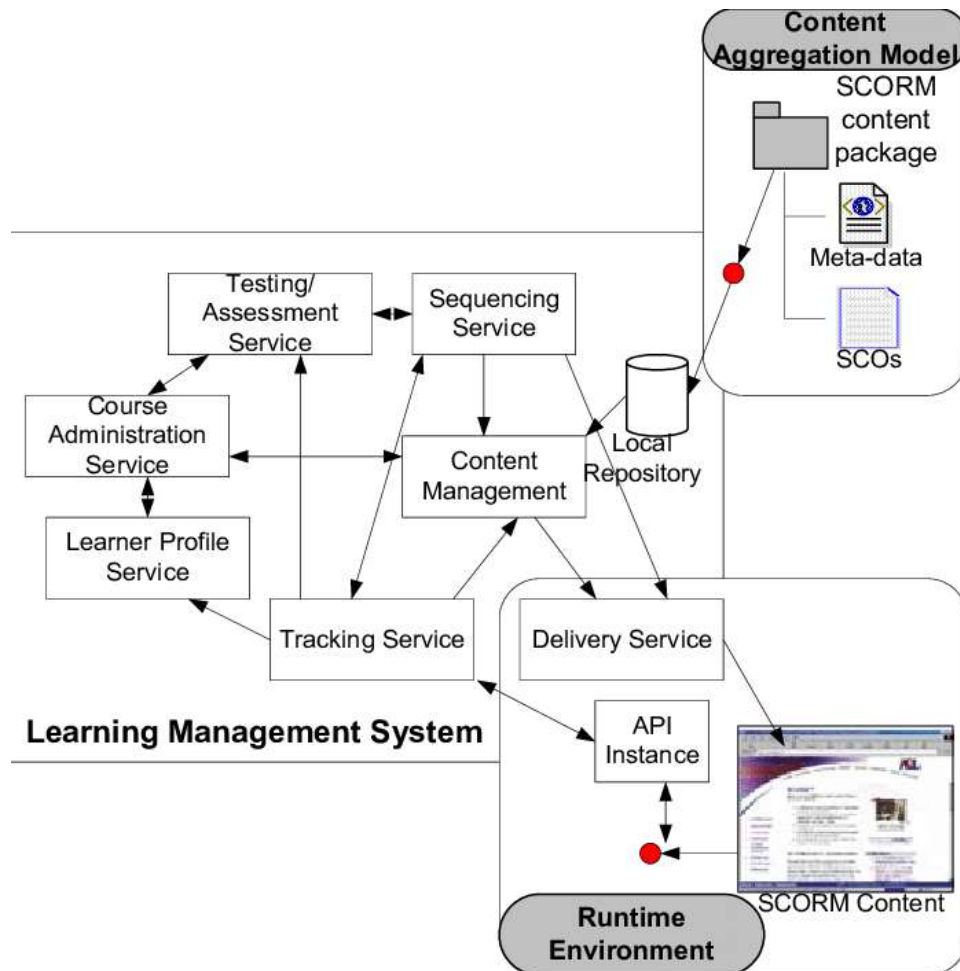


Рисунок 2.2 - Sharable Content Object Reference Model (SCORM) [23]

Blackboard відноситься до платформи електронного навчання в Інтернеті [24]. Однією з переваг цієї моделі є те, що вона використовує концепцію сервіса. Варто відзначити свіжий підхід до використання інтелектуальної концепції за допомогою сервісної технології. Як показано на рис.2.3, архітектура має систему клієнт-сервер для обміну інформацією. Це забезпечує вчителю гнучкість для завантаження великого обсягу інформації, скільки потрібно про предмет, щоб використовувати формат мультимедіа. Така система розділена на два компоненти. Перший — це сервер

ідентифікації, який аутентифікує та авторизує користувача для входу. Наступний етап включає усі матеріали електронного навчання, необхідні для вивчення конкретного предмету.

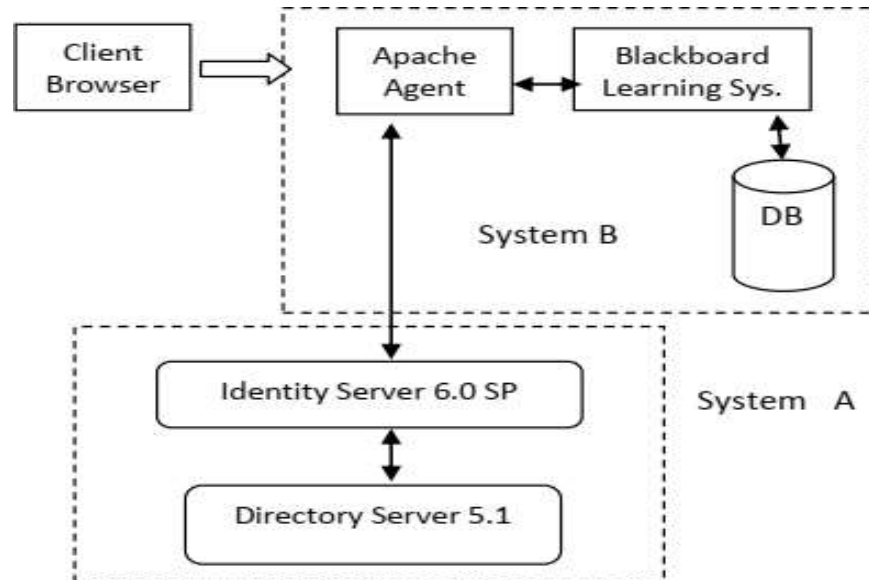


Рисунок 2.3 – Blackboard [25]

Moodle є однією з найскладніших вебпрограм електронного навчання [24]. Це трирівневий дизайн, зображений на рис. 2.4. Виходить, що 1-й рівень є презентаційним рівнем, який надає своїм користувачам досвід роботи через Інтернет-браузер. 2-й рівень став доменно-спеціальним рівнем, який змінюється відповідно до домену, для якого організація бажає розробити онлайн-систему електронного навчання. Останнім і найважливішим рівнем для управління даними є рівень, який зберігає всі дані в базі даних MySQL. Він містить рівень підключення, який дозволяє інтегрувати зовнішні програми для ефективного поширення навчальних матеріалів серед студентів.

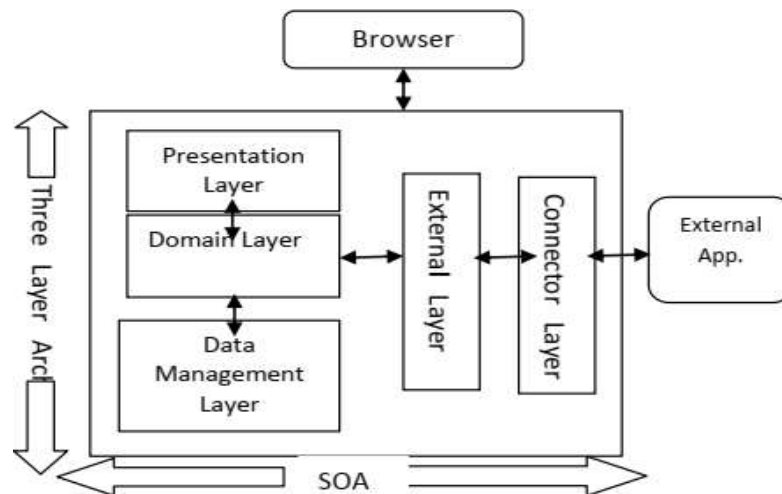


Рисунок 2.4 – Moodle [24]

Schoology. Для досягнення більшого збагачення або індивідуального досвіду навчання ця платформа дозволяє студентам переглядати творчий цифровий контент, створений викладачем, і дозволяє їм вибрати між різними методами дослідження: у класі; напівприсутності, дистанційно або в Інтернеті. Система також надає вчителям повний набір внутрішніх і зовнішніх ресурсів для управління навчанням, таких як: керування існуючим контентом, створення та впровадження нового контенту, створення та перегляд анкет, оцінка результатів студентів, взаємодія зі студентами та викладачами, а також ефективно їх заохочення. в онлайн-середовищі [25].

Sakai - це набір вебмодулів на основі Java, які слабо підключені до сервіс-орієнтованої структури [26]. Підтримуваним вебсервером є Tomcat; Oracle, а також MySQL, які допомагають зберігати дані. Sakai має структуру шарів. Кожен рівень розширюється: простіше розміщувати нові сервіси, програмне забезпечення, а також агрегатори. У формі онлайн-сервісів ядро Sakai надає спільну інфраструктуру, а потім відкриває її. Для вирішення таких питань, як авторизовані користувачі або керування сайтом, усі підпрограми, відомі як інструменти в Sakai, покладаються на відповідні служби. У типі інструментів закладено майже всю зручність використання.

Після вивчення численних дослідницьких робіт, присвячених стандартам систем онлайн навчання, можна казати кожна структура має свої

якості, переваги та недоліки. Дана порівняльна таблиця 2.1 враховує найбільш поширені параметри використання систем.

Таблиця 2.1 – Дослідження параметрів стандартів систем онлайн навчання

№	Система онлайн навчання	Об'єкти навчання	Розширене представлення інформації Технології	Додаткові можливості	Недоліки
1	2	3	4	5	6
1	e-AULA[27]	Метадані від Semantic	RDF, Онтологія	Налаштування	Педагогічна нестандартизація
2	e-LKG[27]	-----	Відкрите джерело	Студенти розвивають свої математичні навички	-----
3	Adaptive elearning System [28]	Теорія реагування на завдання	Онтологія	Фонд гуманізації	-----
4	ADL-SCORM[22]	Багаторазове використання б'єкта попереднього матеріалу	JAVASCRIPT, XML, технологія агента	Доступ до мобільних пристроїв дозволено	Відсутність інтерактивного дизайну, а також розумного підхід
5	Розподілена платформа, сфокусована ресурси [23]	Вебсервіс	SOAP, XML, WSDL, UDDI	Інтероперабілітр	-----
6	Концептуальна система електронного навчання CELF [30]	Самонаведення	-----	Андрогогіка-мистецтво і наука виховання дорослих	Результати дослідження не були емпіричними перевірено
7	AEHS-LS[27]	Реалізувати партнерській принцип	MySQL, Apache, PHP	VARK Анкета	-----

Продовження таблиці 2.1

1	2	3	4	5	6
8	Автоматично рекомендована система на основі Framework [27]	Рекомендація на основі профілю	XML, AJAX, JavaScript	Представлені розподілені середовища	Потрібні неявні та явні поради
9	AeL[30]	Ресурси для мультимедіа	SVG, XML, математикаal Мова розмітки як MathML	Переносний на стільникових пристроях	-----
10	FABULA [31]	Онлайн програми електронного навчання	Онтологія, Мультиагентна система	Допомога для смартфонів, AGORA для корпоративних мереж	-----
11	Спільний ELMS[26]	Стратегія колективного навчання	Онтологія, Мультиагентна система	Використовується як типова основа для колективу Навчання	-----
12	Система електронного навчання на основі голосування [27]	Відповідь за голосуванням	RMI, XML, JavaSpace, Autonomous Agent, а також JATLite	Налаштування	Критерій відбору курсу залежить від того, чи працює інший курс або ні.

Було зауважено, що майже всі інтелектуальні фреймворки електронного навчання мають бути автономними як доменами, так і ролями.

## 2.2 Семантичні технології

Система електронного навчання має користувачів на платформі соціальних технологій, що покращує навчальну діяльність [32]. Онлайн

навчання, навчання за допомогою комп'ютера, дистанційне навчання та навчання через Інтернет також називають електронним навчанням. «Історично існувало два поширених стилі електронного навчання: дистанційне навчання та навчання за допомогою комп'ютера. Дистанційна освіта включає в себе інформаційні технології, щоб надати учням, які перебувають у віддалених районах, керівництво з централізованого місця. Комп'ютерне навчання включає комп'ютери, які допомагають надавати автономні мультимедійні пакети для навчання та викладання» [3].

Електронне навчання описується як «надання в режимі реального часу індивідуального, комплексного, динамічного навчального контенту, що сприяє зростанню інформаційних спільнот, з'єднує учнів і практиків з експертами» [14]. Друкер ввів фразу «своєчасна освіта в поєднанні з високошвидкісними ланцюжками створення вартості» для визначення електронного навчання [25]. Це розповсюдження персоналізованих динамічних навчальних матеріалів у режимі реального часу, систематичне заохочення до створення спільнот знань і встановлення зв'язків між учнями або практиками та експертами. Мета електронного навчання – замінити застаріле місце, час, матеріал і заздалегідь визначена можливість навчання в поєднанні з робочим середовищем або своєчасним або настроюваним освітнім середовищем або на вимогу. Системи електронного навчання виявилися життєздатною платформою, яка може легко та задовільно полегшити процес навчання. У майбутньому вони побудують для нас нові світи в галузі навчання, яких ми ніколи раніше не бачили. З їхньою підтримкою потрібна людина може отримати потрібні знання в потрібний час.

Синтаксично організовані текстові архіви з ефективними пошуковими системами є об'єктами навчання та базами даних курсу. Тим не менш, смисловий зв'язок між інформаційними вимогами користувача та матеріалом статей відсутній. Через очевидні численні обмеження використання Web 2.0

для розробки програм керування електронним навчанням люди наразі щодня застосовують платформу 3.0, яку також називають семантичною мережею.

Найсучасніша ера Інтернет-технологій відома як Web 2.0. Інформація про користувача в рамках соціальної мережі включає характер, інформацію соціального графа, яка ідентифікує особу, і дані вмісту. Доступність цього величезного обсягу вебвмісту все ще обмежена людьми, а не комп'ютерами, і з цим існує ряд проблем [28]. Все це стосується розміщення інформації та документів в Інтернеті, щоб машини могли ефективно зберігати, інтерпретувати, збирати та реагувати на це. Це розширення поточної традиційної мережі, яка дозволяє машинам і людям співпрацювати. Його специфіка має дуже чітко визначене значення. Вебінформація має бути структурована у машиночитаному форматі, щоб дозволити машинам виконувати додаткові операції. Це досягається завдяки розробці нових технологій, а також мови, на якій вебдані можуть бути представлені цими засобами. Семантична павутина стане доповненням до Всесвітньої павутини, яка дозволить людям отримувати доступ до зображень далеко за межами вебсайтів і програм.

Що таке семантична павутина, існує кілька різних теорій. За словами Робіна Д. Морріса [33], семантична мережа, яку також називають Web 3.0, є " не нова мережа, а скоріше розширення існуючої, де інформація надається в чітко визначеній структурі, що дозволяє машинам і окремим особам працювати разом і краще функціонувати у співпраці» [29] і передбачив, що машини зможуть розвивати інформацію на вебсайті, а не просто передавати результати. Комп'ютери здатні аналізувати всі вебдані, інформацію, взаємодії, а також транзакції між особами та комп'ютерами [30]. Семантична павутина стала баченням, що базується на його концепції, і вважається, що вона стане ще одним кроком в еволюції Інтернету. Вся справа в тому, щоб отримати інформацію та документацію в Інтернеті, оскільки тоді комп'ютери можуть ефективно зберігати, інтерпретувати, аналізувати та діяти на основі інформації. Одним із найбільш захоплюючих аспектів Semantic Web є



гнучкість, з якою її можна використовувати для виявлення та розпізнавання матеріалу [28].

В таблиці 2.2 наведені переваги використання семантичної мережі в AL.

Таблиця 2.2 - Порівняння використання семантичної мережі

Вимоги	Електронне навчання	Семантичний веб
1	2	3
Розклад	Pull-Student уточнює порядок денний	Об'єкти знань (навчальні матеріали) поширюються в Інтернеті, але пов'язані з онтологіями, які заздалегіть погоджені. Це дозволяє, шляхом семантичного запиту відповідних тем, створити конкретний навчальний план.
Чуйність	Реакційний - відповідає на поставлене запитання	Широко прийнята мова обслуговування може використовуватися програмними агентами переважно в семантичній мережі, що дозволяє спілкуватися між агентами, а також конструктивно подавати навчальні матеріали в розумінні реальних проблем. Мета полягає в тому, щоб кожен користувач мав власного налаштованого агента для взаємодії з іншим агентів.
Доступ	Нелінійний – дозволяє надавати пряму інформацію в будь-якій серії, яка має сенс речей під рукою.	Користувач повинен визначити суть проблеми (мета навчання, попередні знання) і виконати семантичні запити для необхідного навчального контенту. Новий користувач також передбачений. Доступ до інформації можна розширити за допомогою семантично визначеного управління.

## Продвження таблиці 2.2.

1	2	3
Симетрія	Симетричне навчання приймає місце як інтегрована практика.	Семантична мережа надає можливість усім бізнес-операціям на підприємстві, таким як навчальні практики, стати ітераційною платформою.
Модальність	Безперервний навчання синхронізується з діяльністю компанії, але ніколи не припиняється	Активне поширення знань (на основі налаштованих агентів) забезпечує динамічне навчання, досвід, який включений в процеси навчання.
Автори-тет	Розподілений вміст походить від взаємодії між учасниками та викладачами	Семантична мережа була б настільки децентралізованою, наскільки це можливо. Це дозволяє налаштувати успішний кооператив контроль контенту.
Персоналізація	Персоналізований вміст керується розміром окремої людини, а також прагне задовольнити потреби кожного користувача.	Клієнт шукає навчальний матеріал, адаптований до його потреб (використовуючи свого індивідуального агента). Онтологія – це зв'язок між потребами користувача та характеристиками змісту навчання.
Адаптивність	Динамічна – через відгуки користувачів, перспективи, нові методи, бізнес-правила, а також евристики, вміст постійно змінюється	Семантична павутина дає змогу використовувати спільну інформацію, надану різними способами, що дозволяє семантично анотувати вміст. Розподілений дизайн семантичної мережі дозволяє безперервно розвиватися навчальні матеріали».

У спосіб, який полегшує обробку комп'ютерів у глобальному масштабі, семантичний веб є мережею пов'язаних даних. Це найновіша версія Інтернету, яка дозволяє збирати інформацію у визначеному машинно-інтерпретованому форматі, який готовий до обробки, спільного використання та повторного використання через програмні агенти, а також дозволяє

програмним агентам вивчати значення термінології використовується для представлення даних. Семантична павутина надає розробникам вебдодатків нові технології, спрямовані на забезпечення інтелектуальнішого доступу та управління знаннями в Інтернеті та семантично нюансованого моделювання програм та їх користувачів. Ключовою роботою семантичного вебу є вираження сенсу. Семантична мережа не є окремою мережею, але також є розширенням існуючої мережі, яка дає можливість обмінюватися інформацією та повторно використовувати її. Технологія Semantic Web (SWT) пропонує таку атмосферу, щоб комп'ютери могли спілкуватися один з одним, надаючи правильну інформацію відповідно до потреб користувачів. SWT використовує ряд інструментів, таких як XML RDF і SPARQL, для досягнення цієї місії.

### **2.3 Різновиди онтологій та опис предметної області**

Ключовою концепцією семантичної консультативної системи для студентів університету є семантичне моделювання знань предметної області електронного навчання (наприклад, нормативні документи університету, опис курсів, правила прийому), а також індивідуальна модель користувача, яка відображає поточну ситуацію навчання (наприклад, здані іспити, поточні курси). У цих моделях визначаються фундаментальні структури доступних знань предметної області, а також основні факти (наприклад, пропоновані курси).

У нашій системі структурна частина цих знань електронного навчання моделюється за допомогою онтологій, які формально визначають сутності домену та зв'язки між ними. Для цього ми використовуємо технологію Semantic Web на основі XML і RDF/схеми RDF, відповідно.

Онтологія як термін походить від грецьких слів *onto*, що означає буття, і слова *logia*, що означає письмовий або усний дискурс. Онтологія

означає різні речі для різних людей. У філософії це дослідження існування та природи буття. У семантичній мережі онтології — це формальні визначення або представлення словників або знань, які дозволяють користувачеві визначати класи ресурсів, властивості ресурсів і зв'язки між членами класів ресурсів [34].

Айзенштадт і Вінсент [17] сказали, що «онтологія — це часткова специфікація концептуального словника, який буде використовуватися для формулювання теорій рівня знань про область дискурсу». Як ми можемо бачити рисунку 2.5 три частини RDF Triple.

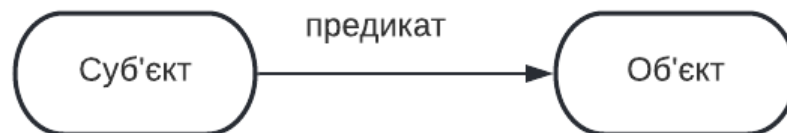


Рисунок 2.5 - Структура опису ресурсу (RDF) Triple

Є кілька переваг використання онтологій, зокрема:

- публікація даних з використанням загальноприйнятої лексики та граматики;
- збереження семантичного опису даних в онтологіях;
- дані готові для висновку;
- розширюваність;
- гнучкість;
- видимість;
- можливість додавати нові властивості в будь-який момент без порушення сумісності [18, 19].

Таблиця 2.4 є приблизним тлумаченням термінів, які використовуються для опису реляційних баз даних і онтологій.

Таблиця 2.4 – Реляційна база даних і онтологія

Реляційна база даних	Онтологія
Рядок	тема
Колонка	Префікс
дані таблиці	концепти

Мова, яка використовується для запитів до онтологій, — це SPARQL (протокол SPARQL і мова запитів RDF), який є набором стандартів W3C для запитів і оновлення даних відповідно до моделі RDF (Resource Description Framework) [35].

Програмні сервіси використовують цю інформацію як основу для своїх міркувань і, завдяки стандартизації цих технологій, вони можуть отримати доступ до розподілених джерел інформації з різних університетів. Таким чином, розроблені онтології можуть служити стандартизованими та відкритими інтерфейсами для взаємодії систем електронного навчання.

Мова онтології OWL є спробою усунути недоліки специфікації RDF/схеми RDF шляхом включення додаткових функцій. OWL+OIL включає підтримку класифікації, обмеження властивостей і можливості для визначення типів. На першому кроці ми обрали мову OWL+OIL для моделювання знань електронного навчання. Основною причиною стала наявність різних інструментів для розробки бази знань.

Для нашої системи електронного навчання буде розроблено дві різні онтології: з одного боку, онтологія, що описує організаційну структуру університетського факультету, з іншого боку, онтологія, що містить знання про конкретного користувача системи.

## 2.4 Висновки за розділом

Правильне розуміння теорії стилю навчання потрібне для розробки робочої системи AL. Моделі стилю навчання були запропоновані різними

дослідниками. Це модель VAK, модель Колба, модель Хані та Мамфорда та модель FSM-Felder-Silverman. У цих моделях було дотримано або спільний, або автоматичний підхід. Стили навчання базуються на результатах досліджень когнітивної психології щодо обробки інформації, активного навчання та структури інформації.

У нашому проєкті модуль навчання використовує техніку моделі Sabine Graf і Jackson Learning Style Profiler і використовує семантику та базу знань онтологій для отримання стилів навчання в системі семантичних рекомендацій. Ми вважаємо за краще використовувати цю модель, оскільки за допомогою онтологій можливо описати модель студента, модель кафедри та модель системи онлайн навчання.

Електронне навчання онлайн є дуже складним напрямом дослідження, тому IEEE Learning Technology Systems Framework (LTSC), Sharable Content Object Reference Model (SCORM), Blackboard, Moodle, Schoology та Sakai є одними з найпоширеніших онлайн-стандартів.

Семантична павутина надає розробникам вебдодатків нові технології, спрямовані на забезпечення інтелектуальнішого доступу та управління знаннями в Інтернеті та семантично нюансованого моделювання програм та їх користувачів.

Семантична мережа не є окремою мережею, але також є розширенням існуючої мережі, яка дає можливість обмінюватися інформацією та повторно використовувати її. У нашій системі структурна частина цих знань електронного навчання моделюється за допомогою онтологій, які формально визначають сутності домену та зв'язки між ними. Для цього ми використовуємо технологію Semantic Web на основі XML і RDF/схеми RDF, відповідно. Програмні сервіси використовують цю інформацію як основу для своїх міркувань і, завдяки стандартизації цих технологій, вони можуть отримати доступ до розподілених джерел інформації з різних університетів. Таким чином, розроблені онтології можуть служити стандартизованими та відкритими інтерфейсами для взаємодії систем електронного навчання. Для

нашої системи електронного навчання буде розроблено дві різні онтології: з одного боку, онтологія, що описує організаційну структуру університетського факультету, з іншого боку, онтологія, що містить знання про конкретного користувача системи.

## **3 РОЗРОБКА АРХІТЕКТУРИ СИСТЕМИ**

### **3.1 Загальна архітектура системи**

Зменшення змісту навчання до значущих одиниць за допомогою ручних, напівавтоматичних і автоматичних процедур є важким процесом. В архітектурі вебсервісів WSDL-Web Service Description Language використовується для опису вебсервісу, який враховує синтаксичні аспекти. Операції, пов'язані з вебслужбами, дуже важко виконати за допомогою цих синтаксичних описів. Але онтології профілю та процесів OWL-S базуються на семантиці, за допомогою якої полегшується інтеграція, виявлення та створення вебслужб. Створення робочих процесів вручну, як і будь-який керований користувачем процес, є завданням, схильним до помилок і невідповідностей.

Під час обробки електронного навчання учень, який не знає про свої вимоги та чий навчальний профіль потрібно знайти, може не брати участь у створенні вебслужб електронного навчання. За профілем користувача та його поведінкою в онлайн-системі навчання має визначити вимоги користувача. Просте зв'язування вхідних і вихідних параметрів у вебслужбах не допоможе. Автоматизація цієї композиції робочого потоку допомагає звільнитися від тягаря вирішувати, які послуги їм потрібні, і потребує опису семантики за допомогою OWL-S (рис. 3.1).



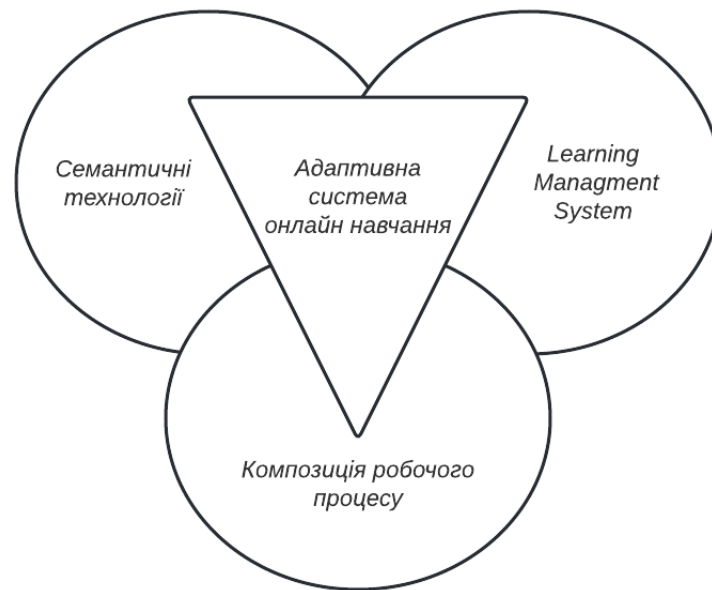


Рисунок 3.1 - Концептуальна модель адаптивної системи онлайн навчання

До переваг даної системи треба віднести три аспекти: визначення стилю навчання, представлення знань і склад робочого процесу, щоб забезпечити наступні переваги для будь-якого електронного учня.

Сфера електронного навчання складається з безлічі складних дій, таких як створення контенту або відстеження й адміністрування учнів, які взаємодіють із ресурсами (зокрема з такими людьми, як учні та автори), один з одним (одні види діяльності викликають інші) та з зовнішній світ (наприклад, існуючі програмні системи) заздалегідь визначеним способом. Jihie Kim та інші [25] пояснили про методи планування штучного інтелекту, у яких багата база знань використовується для автоматичної композиції. Вони винайшли підхід до інтерактивної композиції робочого процесу, який включає багаті знаннями описи окремих компонентів та їхніх обмежень та формальне алгоритмічне розуміння часткових робочих процесів на основі методів планування ШІ. Еран Чінтака та інші [20] пропонують підхід аргументації на основі випадків (СВР) для сприяння створенню робочих процесів на основі характеристик входів і виходів багаторазово

використовуваних компонентів робочого процесу, полегшуючи користувачам використання існуючих сервісів і робочих процесів під час створення робочого процесу.

Хорхе Кардосо та Аміт Шет [14] запропонували техніку композиції робочого процесу на основі QoS, у якій для композиції надається важливе значення оперативним вимогам, якості доставлених продуктів, вартості обслуговування та надійності. Xubo Fei і Shiyong Lu [15] використовують підхід, керований потоком даних, для складання наукового робочого процесу. У цій роботі вони запропонували наукову модель робочого процесу на основі потоку даних, яка відокремлює оголошення інтерфейсу робочого процесу від визначення його функціонального тіла; набір конструкцій робочого циклу, включаючи Map, Reduce, Tree, Loop і Conditional, які повністю композиційні одна з одною, підхід до обробки винятків на основі потоку даних для підтримки ієрархічного розповсюдження винятків і визначена користувачем обробка винятків є унікальними особливостями цієї статті.

Наш підхід полягає в розробці семантичної системи (СС) рекомендацій для онлайн навчання. Розроблена структура наведена на рисунку 3.2. Вона фокусується на циклічній моделі, яка включає безперервне навчання – цикл зворотний зв'язок-оновлення-навчання.



Рисунок 3.2 – Структурна модель системи

Структуру СС також можна розглядати як модель QoS, яка використовує стиль учнів як операційну метрику для визначення робочого процесу служб електронного навчання. Цю систему розділено на наступні модулі, згадані нижче.

Модуль введення. Цей модуль базується на техніці, яка використовує поєднання як спільних, так і автоматичних концепцій, де поведінка, взаємодія та ставлення учнів відстежуються на основі певного списку дій. Якщо профіль учня доступний у базі знань, це також буде отримано. Тут користувачам не пропонують заповнювати довгу анкету, замість цього вони виконують певні види діяльності (типу ігор) для відповідного вивчення предмета. Наприклад, користувача можуть попросити вказати певні ключові слова, які стосуються теми.

Модуль отримання семантики. Онтології, введені користувачем, створюються та зберігаються. Тут будується контекстна онтологія, онтологія, пов'язана з навчальним об'єктом:

- онтологія учня;
- онтологія стилю учня;
- онтологія домену;
- педагогічна онтологія;
- онтологія адаптації;
- онтологія робочого процесу. Цей модуль не залучає користувачів.

Цей модуль автоматично виявляє доступні навчальні об'єкти та приймає онтології для цього предмета, указанного користувачем.

Модуль вилучення знань/виявлення стилю учнів. Онтології, отримані з модуля отримання знань, і відомості про профіль/історію учня використовуються для визначення типу учня, іншими словами, визначаються переваги учня щодо навчання. Наша методика використовує методологію, у якій поведінка, дії, взаємодії та профіль учня (якщо він уже існує в базі знань) зберігаються в термінах онтологій. Семантика, отримана з онтологій, є вхідними для визначення вимог учня. Розсудники онтологій задіяні для

виконання різних операцій, таких як об'єднання, перетин, злиття та уточнення онтологій, які можна виконати, щоб дізнатися стиль навчання електронного учня.

Модуль створення потоку робіт. Вебсервіси є незалежними сутностями. Досягнення сумісності між різнорідними вебсервісами є складним завданням. Стандарти, які використовуються для створення незалежних, портативних, сумісних вебсервісів, це WSDL, SOAP і UDDI. Але WSDL використовує синтаксичний опис вебсервісу. Робочий процес, який є абстракцією бізнес-процесу, що складається з одного або кількох вебсервісів, послідовних для вирішення бізнес-запиту. Синтаксична природа WSDL не дозволяє компонувати або оркеструвати ці служби статично чи іншим способом за допомогою втручання користувача. Автоматична композиція може бути досягнута за допомогою мови вебонтології OWL-S у створенні семантичних вебсервісів. Профіль та онтологія процесу використовуються для динамічного створення сервісів. Планувальник або композитор відповідає за створення плану робочого процесу. Сервіси електронного навчання на основі семантичної мережі, які надають різні навчальні об'єкти як послідовність, будуть визначені для уподобань учнів.

Модуль виконання реалізує послідовність робочого процесу, отриману в модулі створення. Обмін повідомленнями відбувається під час спілкування між різними об'єктами в послідовності. Учням буде надано правильну послідовність виконання послуг і послідовність

Таблиця 3.1 описує модулі створеної системи.

Таблиця 3.1 – Модулі СС онлайн навчання

Назва модулю	Опис
1	2
Модуль введення.	Цей модуль базується на техніці, яка використовує поєднання як спільних, так і автоматичних концепцій, де поведінка, взаємодія та ставлення учнів відстежуються на основі певного списку дій.

## Продовження таблиці 3.1

1	2
Модуль отримання семантики.	Цей модуль не залучає користувачів. Цей модуль автоматично виявляє доступні навчальні об'єкти та приймає онтології для цього предмета, указанного користувачем.
Модуль вилучення знань/виявлення стилю учнів.	Онтології, отримані з модуля отримання знань, і відомості про профіль/історію учня використовуються для визначення типу учня, іншими словами, визначаються переваги учня щодо навчання. Наша методика використовує методологію, у якій поведінка, дії, взаємодії та профіль учня (якщо він уже існує в базі знань) зберігаються в термінах онтологій. Семантика, отримана з онтологій, є вхідними для визначення вимог учня.
Модуль створення потоку робіт.	Профіль та онтологія процесу використовуються для динамічного створення сервісів. Планувальник або композитор відповідає за створення плану робочого процесу.

Для реалізації програмної системи, що відображає цю ситуацію, ми обрали сервісну архітектуру. На рисунку 3.3 показано структуру системи електронного навчання з двома різними типами сервісів: користувача та кафедри.

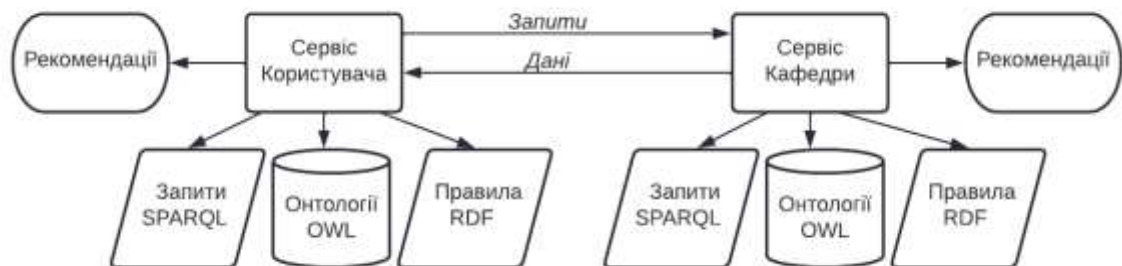


Рисунок 3.3 - Сервісна архітектура

Сервіс користувача реалізований у класі «Сервіс Користувача» і містить онтологію користувача з усією необхідною інформацією про особисті дані, курси, навички тощо. Коли сервіс користувача запускається,

він читає онтологію користувача зі своїми персональними даними за допомогою програмного модуля. Потім правила та запити, специфічні для сервіса користувача, завантажуються та затверджуються в системі. Сервіс кафедри має всі знання про відділ, наприклад, навчальний план і правила іспитів, які змодельовані у його власному модулі.

### 3.2 Підсистема онтологій

Онтологія кафедри моделює основні частини організаційної структури університету (рис. 3.4). Акцент робиться на окремих відділах, різних ролях осіб у відділі та курсах.

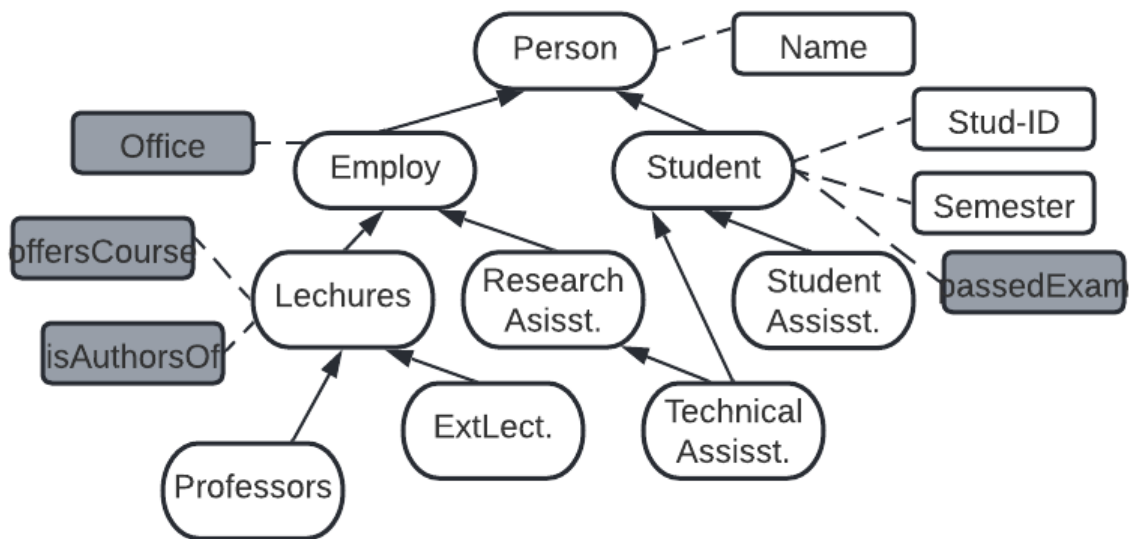


Рисунок 3.4 - Онтологія кафедри – гілка особи

Вона моделюється наступним чином. Кожна організаційна одиниця визначається як підклас організації. Для цього суперкласу визначено транзитивну властивість, таким чином можна легко змодельувати ієрархію екземплярів. У OWL ця транзитивність моделюється таким чином:

```
<OWL:TransitiveProperty rdf:ID="subOrg">
```

```

<rdfs:label>підорганізація</rdfs:label>
<rdfs:domain rdf:resource="#Організація"/>
<rdfs:range rdf:resource="#Організація"/>
</OWL:TransitiveProperty>

```

Існує чотири властивості об'єкта OWL, які надаються в protege. Функціональна властивість обмежена лише одним відношенням до іншого індивіда для зазначеного індивіда. У цьому випадку властивості типу функціональних даних student: hasGgroup розглядаються як функціональні властивості. Властивості, які описують індивіда, є зворотними іншим індивідам. У цьому дослідженні hasStudent є зворотною властивістю hasTeacher.

Транзитивність використовується у файлах екземплярів для моделювання конкретної ієрархії. Наприклад, студентський проект є дочірньою організацією кафедри, а кафедра Програмних засобів є дочірньою організацією університету НУ «Запорізька Політехніка».

```

<owl:department rdf:ID="Програмні засоби">
  <owl:subOrg>
    <owl:NUZP rdf:about="#Національний Університет «Запорізька
Політехніка"/>
  </owl:subOrg>
</owl:department>
<owl:project rdf:ID="Проект1">
  <owl:subOrg>
    <owl:department rdf:about="#Програмні засоби"/>
  </owl:subOrg>
</owl:project>

```

Усі інші частини онтології належать організації. Це моделюється властивістю `<OWL:ObjectProperty rdf:ID="isPartOf"/>`, яка обмежена конкретним підкласом організації.

Частина онтології, яка моделює користувача, показана на рисунку 3.4. Семантика успадкування в цій таксономії дещо відрізняється від об'єктно-орієнтованого програмування. В об'єктно-орієнтованому програмуванні він виражає спеціалізацію зв'язку «є-є», тоді як у контексті онтологій він служить головним чином як категоризація знань.

Для ясності графічне представлення не показує всю інформацію про зв'язки. Зокрема, не показано, до якого класу чи властивості іншої гілки онтології йдеться. Одним із прикладів є властивість `offersCourse` класу `Lecturer`. У нотації XML це визначено таким чином:

```
<OWL: Object property rdf:ID="offersCourse">
<rdfs:label> пропонує курс </rdfs:label>
<rdfs:domain rdf:resource="#Лектори"/>
<rdfs:range rdf:resource="#Курс"/>
<OWL:minCardinality>1
</OWL:minCardinality>
</OWL:ObjectProperty>
```

Лектор викладає один або кілька курсів, і можна переходити від лектора до певного курсу. У курсовій гілці онтології можна знайти властивість `hasLecturer` зі схожою семантикою зі зворотним напрямком навігації. Це можна визначити як зворотну властивість у OWL.

```
<OWL: Object property rdf:ID="hasLecturer">
<OWL:label> пропонується </OWL:label>
<OWL:inverseOf rdf:resource="#offersCourse"/>
</OWL:ObjectProperty>
```



На рисунку 3.5 зображена гілка курсу онтології електронного навчання. Інші характеристики підкласів не візуалізуються графічним позначенням. Наприклад, курс є диз'юнктивним об'єднанням своїх підкласів. У OWL це моделюється наступним чином.

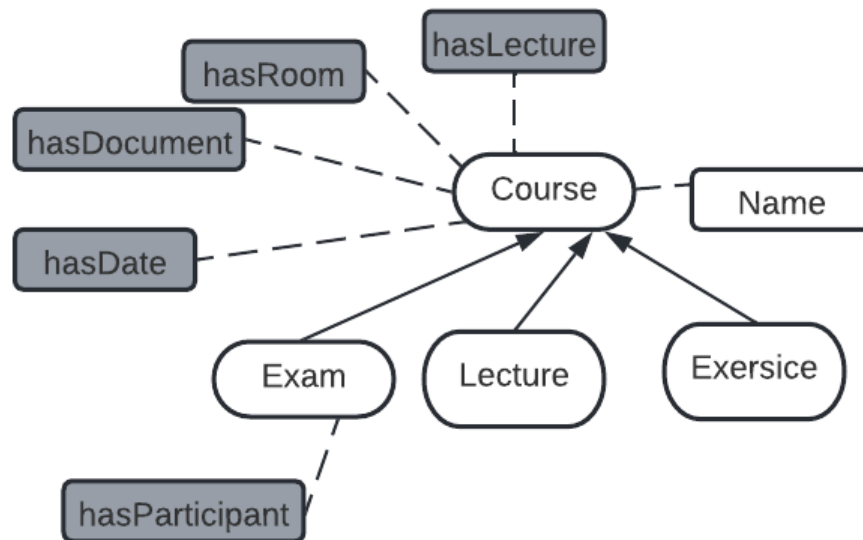


Рисунок 3.5 – Онтологія кафедри – гілка курсу

```
< OWL:Class rdf:about="#Курс">
  <OWL:disjointUnionOf
rdf:parseType="http://www.OWL.org/2011/03/OWL+ #collection">
  <OWL:Class rdf:about="#Лекція"/>
  <OWL:Class rdf:about="#Вправа"/>
  <OWL:Class rdf:about="#Іспит"/>
  </OWL:disjointUnionOf>
</OWL:Class>
```

Ця конструкція гарантує, що курс є лекцією, вправою або іспитом.

Онтологія користувача служить моделлю знань конкретного користувача, наприклад, студента або викладача. Основним класом онтології є користувач. Користувач - це особа онтології кафедри. Це моделюється

властивістю об'єкта `sameClassAs`, яка є елементом OWL для моделювання інтеронтологічної еквівалентності.

```
<OWL:Class rdf:about="#User">
  <OWL:sameClassAs #Людина"/>
</OWL:Class>
```

Додаткові властивості моделюють усі відповідні дані особи, наприклад ім'я для входу, ідентифікатор студента, поточний семестр, пройдені/не пройдені курси, дата останнього входу, навички тощо.

### 3.3 Висновки за розділом

Під час обробки електронного навчання учень, який не знає про свої вимоги та чий навчальний профіль потрібно знайти, може не брати участь у створенні вебслужб електронного навчання. За профілем користувача та його поведінкою в онлайн-системі навчання має визначити вимоги користувача. Просте зв'язування вхідних і вихідних параметрів у вебслужбах не допоможе. Автоматизація цієї композиції робочого потоку допомагає звільнитися від тягаря вирішувати, які послуги їм потрібні, і потребує опису семантики за допомогою OWL-S. Наш підхід полягає в розробці семантичної системи (СС) рекомендацій для онлайн навчання. Вона фокусується на циклічній моделі, яка включає безперервне навчання – цикл зворотний зв'язок-оновлення-навчання.

Структуру СС також можна розглядати як модель QoS, яка використовує стиль учнів як операційну метрику для визначення робочого процесу служб електронного навчання. Для реалізації програмної системи, що відображає цю ситуацію, ми обрали сервісну архітектуру.

Сервіс користувача реалізований у класі «Сервіс Користувача» і містить онтологію користувача з усією необхідною інформацією про

особисті дані, курси, навички тощо. Коли сервіс користувача запускається, він читає онтологію користувача зі своїми персональними даними за допомогою програмного модуля. Потім правила та запити, специфічні для сервіса користувача, завантажуються та затверджуються в системі. Сервіс кафедри має всі знання про відділ, наприклад, навчальний план і правила іспитів, які змодельовані у його власному модулі.

Онтологія кафедри моделює основні частини організаційної структури університету. Акцент робиться на окремих відділах, різних ролях осіб у відділі та курсах. Транзитивність використовується у файлах екземплярів для моделювання конкретної ієрархії. Наприклад, студентський проект є дочірньою організацією кафедри, а кафедра Програмних засобів є дочірньою організацією університету НУ «Запорізька Політехніка». В результаті створення онтологій даної предметної області ми отримали 160 концептів, 70 властивостей та 50 ролів основних знань про систему.

## 4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ КОМПОНЕНТІВ СИСТЕМИ

### 4.1 Налаштування онтології системи онлайн навчання

Розроблені онтології базуються на електронному навчанні Moodle. За замовчуванням онтологія містить клас під назвою Thing, цей клас є найвищим класом, який представляє весь набір індивідів, тому весь клас є підкласом Thing. Класи для цієї онтологічної структури створюються шляхом додавання підкласу, як показано на рисунку 4.1.

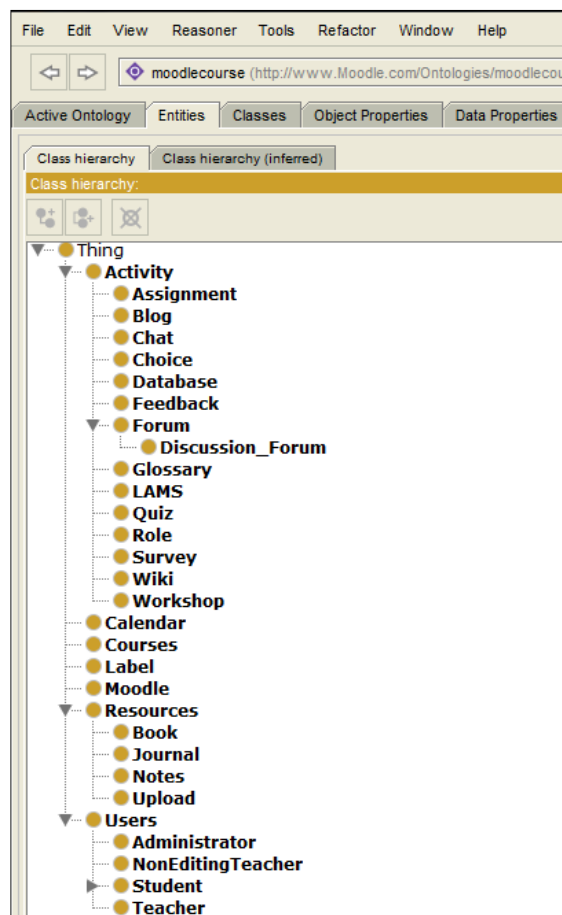


Рисунок 4.1 - Створення класу для онтології Moodle у Protege

Онтологія Moodle складається з кількох основних класів, а деякі класи мають підкласи. Підклас, який відображає завдання, яке дається студенту для кожного курсу для вимірювання успішності студента. Блог – це підклас

активності Moodle, який можна використовувати для експресивності та спілкування з іншими людьми.

Чат – це Підклас активності Moodle, який може допомогти людям зустрітися в певній віртуальній кімнаті та спілкуватися один з одним безпосередньо за допомогою повідомлень.

Піджурнал, який керує процесом маркування кожної діяльності складається з наступних сутностей:

- користувач;
- студент;
- викладач;
- адміністратор;
- курси.

Щоб завершити категоризацію підкласу, клас Користувачі може складатися з кількох непересічних підкласів.

Властивості в OWL описують зв'язок між класами. Існує два основних типи властивостей: властивості об'єкта та типу даних. Властивості об'єкта пов'язані зі зв'язком між окремими, тоді як властивості типу даних є окремим посиланням на тип даних XML-схеми. Третя властивість — властивості анотації. Цю властивість можна використовувати для додавання інформації.

Детальний опис властивостей об'єктів, отриманих з електронного навчання Moodle, проілюстровано на рисунку 4.2.

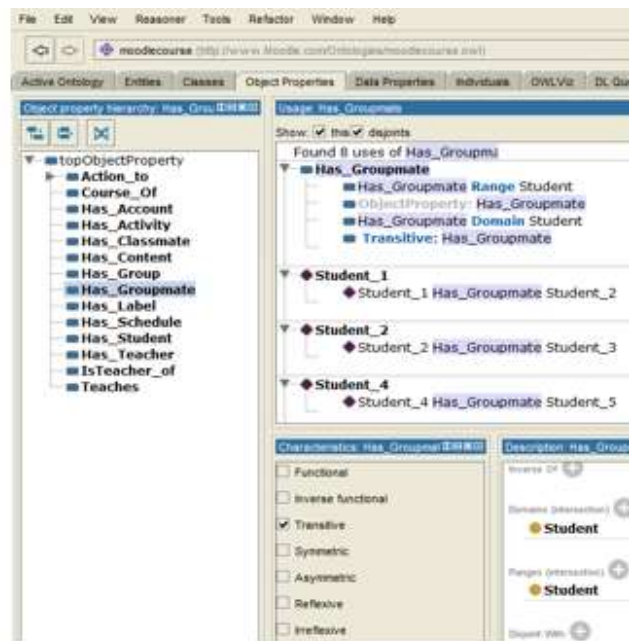


Рисунок 4.2 - Властивості об'єктів електронного навчання Moodle

Властивості типу даних підключають окрему особу до XML-схеми Datatype або rdf. Вони описують зв'язки між індивідом і значеннями даних. Властивості користувачів домену та курсу з рядком діапазону та типом властивостей типу даних є функціональними. Це означає, що користувачі та курс повинні мати лише один Id\_member.

Докладні властивості типу даних Has\_idMember у Protege показані на рисунку 4.3.

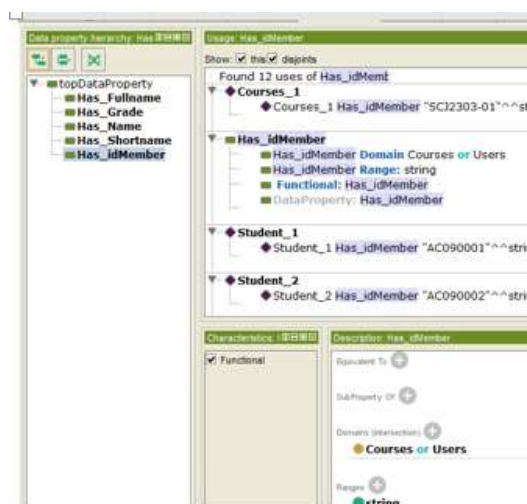


Рисунок 4.3 – Функціональні властивості

Останнім кроком є перевірка аномалій або невідповідності дизайну онтології, який було завершено на попередніх етапах. Інструменти для перевірки є у вільному доступі. Для перевірки узгодженості структури онтології Moodle використовується резонер Pellet, як показано на рисунку 4.4.

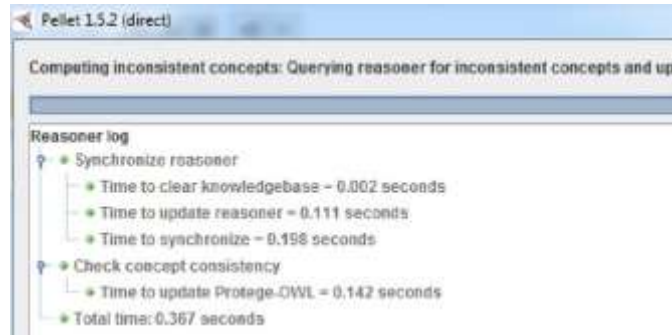


Рисунок 4.4 - OWL перевірка Pellet

На рисунку 4.4 перевіряється узгодженість резонером Pellet яка зайняла 0,367 секунди.

## 4.2 Рекомендації щодо семантичного змісту

Рекомендація щодо змісту навчання складається з чотирьох кроків, як показано на рисунку 4.5. По-перше, розрахунок семантичної релевантності обчислює семантичну подібність між студентом та змістом навчання, а потім відповідно створює список рекомендацій. По-друге, уточнення рекомендацій забезпечує інтерактивний спосіб коригування результату, доки не буде досягнуто кількох прийнятних варіантів. Коли студент вибирає один предмет із кандидатів, Learning Path Generation будує навчальний маршрут, що складається з попереднього запиту та цільового змісту навчання, який керує процесом навчання. Нарешті, доповнення рекомендацій об'єднує додатковий вміст, пов'язаний з основним курсом. Кожен крок рекомендації виконується шляхом використання знань про студента, знань про систему навчання або

знань предметної області. База онтології забезпечує постійне зберігання та ефективний пошук таких знань.

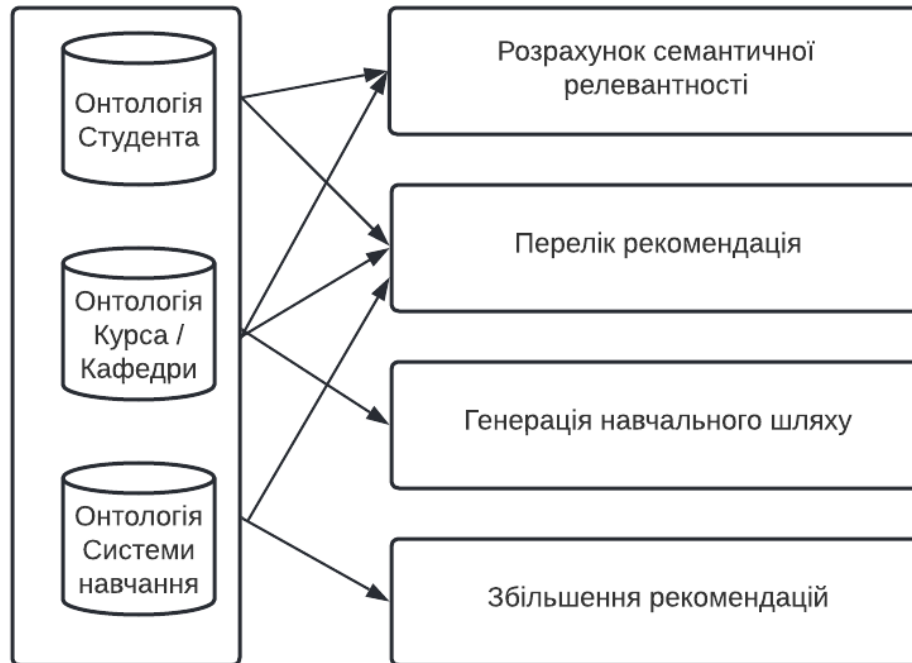


Рисунок 4.5 - Процедура рекомендації контенту

Розрахунок семантичної релевантності.

Щоб отримати рекомендацію, нам спочатку потрібно ранжувати навчальний вміст щодо того, наскільки вміст задовольняє контекст студента. Наша система використовує семантичну релевантність між метою студента та змістом курсу як критерій рейтингу.

Семантична релевантність надихається теорією категорій і концептуальним графом [13]. Інтуїтивно зрозуміло, що об'єкти в одному або пов'язаному домені можуть мати певну схожість один з одним. Іншими словами, екземпляри в ієрархії категорій мають певну спільність. Подібність між двома об'єктами в ієрархії категорій можна виміряти відповідно до їх кореляції в моделі ієрархії. Це робиться шляхом аналізу позицій об'єктів у моделі ієрархії. Чим ближче два об'єкти, тим більшою буде схожість між ними.



Семантична релевантність обчислюється за допомогою таких кроків:

- відобразити мету користувача в онтології домену;
- знайти предмет навчального змісту в онтології домену;
- оцінити концептуальну близькість між відображеним елементом і

предметним вузлом змісту навчання.

Концептуальна близькість  $C(e1, e2)$  формально визначається відповідно до таких правил .

Правило 1. Концептуальна близькість завжди позитивне число, тобто

$$C(e1, e2) > 0. \quad (1)$$

Правило 2. Понятійна близькість має властивість симетрії, тобто

$$C(e1, e2) = C(e2, e1). \quad (2)$$

Правило 3. Якщо  $e1$  збігається з  $e2$ , то

$$C(e1, e2) = Dep(e1) / M. \quad (3)$$

Правило 4. Якщо  $e1$  відрізняється від  $e2$  і між ними немає зв'язку «предок-нащадок»,

$$C(e1, e2) = Dep(LCA(e1, e2)) / M \quad (4)$$

У рівняннях (3), (4)  $M$  позначає загальну глибину онтології ієрархії домену;  $Dep(e)$  — це глибина вузла  $e$  в ієрархії (кореневий вузол завжди має найменшу глибину, скажімо, 1);  $LCA(x, y)$  означає вузол найменшого спільного предка для вузла  $x$  і  $y$ .

Семантична релевантність базується на інтуїтивному уявленні про те, що ступінь релевантності між навчальною метою та змісту курсу зростає, чим вони ближчі, а про них більше відомо. Розрахувавши семантичну релевантність, ми можемо рекомендувати курс, семантична релевантність якого перевищує попередньо встановлене порогове значення.

Список рекомендацій може бути наданий студенту одразу. Однак він все ще може впливати на результати рекомендацій. Наша система пропонує інтерактивне уточнення рекомендацій [14], за допомогою якого студент може взаємодіяти з системою, критикуючи її та інтерактивно уточнюючи результати, доки не буде досягнуто кількох прийнятних варіантів. Результат

рекомендації можна уточнити за такими ознаками: спеціальність, складність та інтерактивність.

Спеціальність. Якщо результат містить дуже мало елементів і студент хоче отримати більш узагальнений зміст, система може надати весь зміст, предмет якого опускається вище рівень LCA (тут ми визначаємо LCA як найменш спільного предка поточних елементів рекомендації, який може містити або не містити підклас) в ієрархії. Подібним чином, якщо результат включає багато елементів і студент хоче отримати більш спеціалізований зміст, система може повернути той зміст, предметом якого є один нижчий рівень LCA в ієрархії. Коли запускається «Більш спеціалізована» дія уточнення, з'явиться діалогове вікно, у якому студентові буде запропоновано вибрати один підклас LCA.

Складність. Студент може уточнити результат, щоб вибрати легший або складніший зміст. Цього можна досягти за допомогою властивості `hasDifficulty` змісту. Кожному курсу при створенні призначається рівень складності, який включає «дуже легкий», «легкий», «середній», «складний» і «дуже складний».

Інтерактивність. Подібно до складності, студент може отримати зміст із бажаною інтерактивністю, збільшуючи або зменшуючи рівень інтерактивності певного елемента. Корективку можна здійснити за допомогою властивості `hasInteractivity` змісту. Під час створення контенту надається рівень інтерактивності відповідно до методу його представлення та компонування. Рівень інтерактивності коливається від «дуже низького», «низького», «середнього», «високого» до «дуже високого».

Зазвичай окремий навчальний контент буде неможливим для студента, щоб досягти своєї мети, оскільки сам навчальний курс може мати передумови, які користувач ще не засвоїв. Тому нам потрібно надати студентові навчальний шлях, щоб керувати процесом навчання, і запропонувати користувачеві отримати деякі попередні знання перед вивченням цільового курсу.

Коли студент вибирає один елемент зі списку рекомендацій, система може створити навчальний шлях, пов'язаний із обов'язковим вмістом і цільовим навчальним курсом.

### 4.3 Розробка додатку

Винесення спільних функцій в окремі компоненти, придатні для багаторазового застосування, дозволяє досягати значної економії в кодуванні та більшої ясності в архітектурі системи. Таким чином, процес побудови системи зводиться до вибору необхідних компонентів та зв'язування їх належним чином. Можливість взаємозв'язку різних компонентів досягається за рахунок стандартних методологій та структур різних мов програмування Java, Ruby, Python тощо. Вони описують загальну структуру та набір інтерфейсів для створення системи, а також можуть містити деякі готові компоненти. В таблиці 4.1 наведена відповідність окремих мов програмування до функцій роботи з онтологіями.

Таблиця 4.1 – Порівняння мов програмування відповідно до функцій роботи з онтологіями.

Функції	Java	Ruby	Python
Функції GUI	+	+	+
Підтримка XML	+	+	+
Підтримка OWL/RDF	+	+	-
Робота з базами даних	+	+	+
Робота з Protege	+	-	-
Функції роботи з SPARQL	+	+	-

Після аналізу мов програмування для побудови додатку була обрана мова Java, тому що в нею присутні функції для роботи з онтологіями, вона підтримує вбудовані в середу Protégé.

За допомогою запропонованого рекомендаційного підходу ми створили систему семантичних рекомендацій онлайн навчання. Функціональна схема додатку показана на рисунку 4.6.

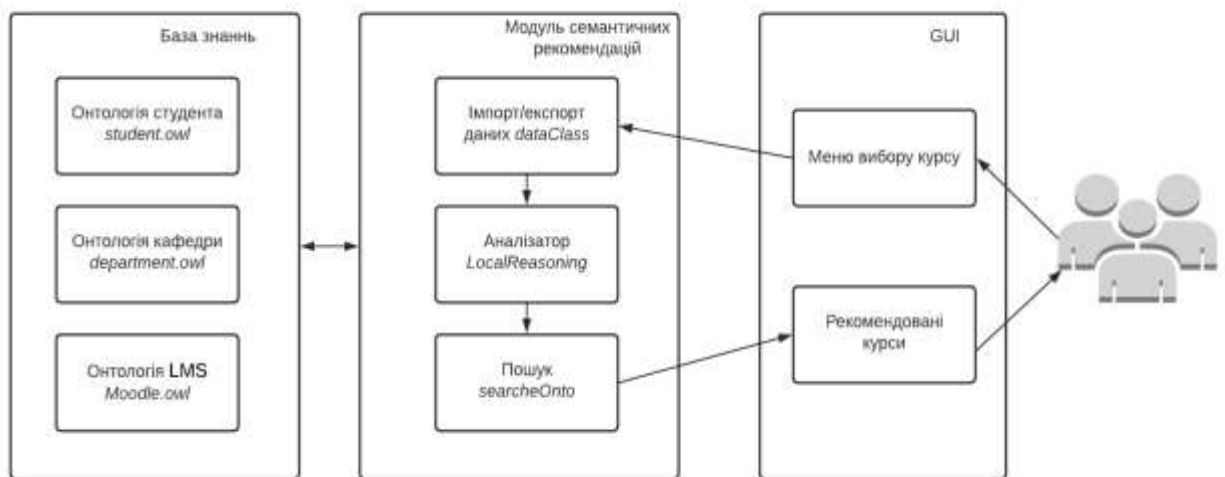


Рисунок 4.6 – Функціональна схема додатку семантичних рекомендацій

Для порівняння онтологій був розроблений окремий модуль, який зчитує дві онтології та порівнює їх за вершинами графа та знаходить найближчу відстань між однаковими значеннями:

```
private static Double lexEquality(OntClass c1, long
depthFirst, OntClass c2, long depthSecond) {
    Double result = Math.log(2 * depthFirst /
distance(c1));
    result /= Math.log(2 * depthSecond / distance(c2));
    return result;
}
```

Значення коцептів порівнюється за допомогою класу:

```

public class LocalReasoner {
    public static double checkSim(OntModel first, OntModel
second) {
        long depthFirst = countDepth(first);
        long depthSecond = countDepth(second);
        AtomicReference<Double> result = new
AtomicReference<>(Double.valueOf(0));
        AtomicLong equalClassesCounter = new AtomicLong(0);
        first.listClasses().forEachRemaining(c1 -> {
            second.listClasses().forEachRemaining(c2 -> {
                double currentResult;
                if (equalURI(c1, c2)) {
                    currentResult = 1.0;
                } else {
                    if (equalNames(c1, c2)) {
                        currentResult = lexEquality(c1,
depthFirst, c2, depthSecond);
                    } else currentResult = 0.0;
                }
                result.getAndUpdate(value -> value +=
currentResult);
                if (currentResult > 0.0) {
                    equalClassesCounter.incrementAndGet();
                }
            });
        });
    });
}

```

На рисунку 4.7 показано клієнтський інтерфейс для отримання рекомендації семантичного вмісту. В основному він складається з чотирьох частин. Верхня частина надає інтерфейс, за допомогою якого студент може ввести навчальну мету та вибрати вже вивчені курси. Щоб полегшити введення цілей навчання, ми надаємо дерево предметів.

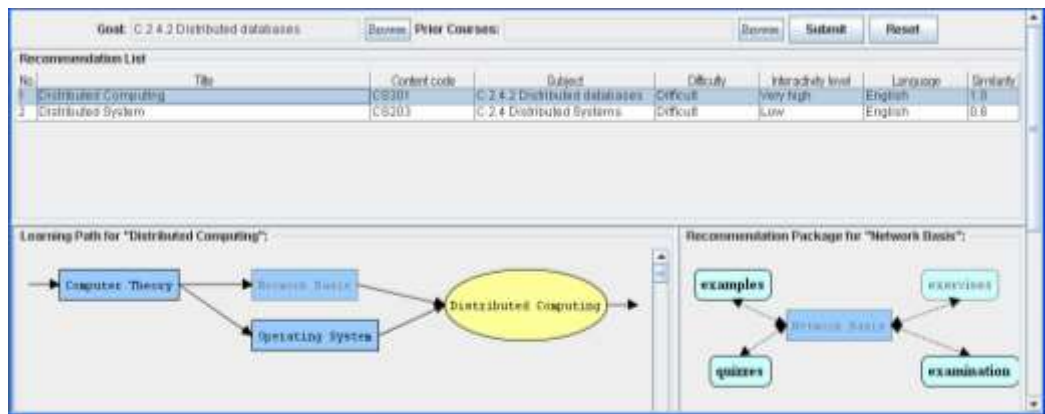


Рисунок 4.7 – Інтерфейс системи рекомендацій

Було протестовано накладні витрати на рекомендацію семантичного вмісту з точки зору часу відповіді. Час роботи для кожного кроку, тобто створення списку рекомендацій вмісту, уточнення рекомендацій, створення шляху навчання та генерування пакету, було середнє значення 10 прогонів. Ми помітили, що час для створення списку рекомендацій вмісту був найбільшим, скажімо, 78 мс; генерація шляху навчання зайняла 16 мс; як очищення, так і створення упаковки коштують менше 1 мс. Таким чином, загальний час семантичної рекомендації становить менше 100 мс. Завдяки цьому експерименту зроблено висновок, що наш підхід є легким і можливим для розгортання.

#### 4.4 Висновки за розділом

Розроблена онтологія базується на електронному навчанні Moodle. За замовчуванням онтологія містить клас під назвою Thing, цей клас є найвищим класом, який представляє весь набір індивідів, тому весь клас є підкласом Thing. Класи для цієї онтологічної структури створюються шляхом додавання підкласу. Онтологія Moodle складається з кількох основних класів, а деякі класи мають підкласи. Властивості в OWL описують зв'язок між класами. Існує два основних типи властивостей: властивості об'єкта та типу даних. Властивості об'єкта пов'язані зі зв'язком між окремими, тоді як

властивості типу даних є окремим посиланням на тип даних XML-схеми. Третя властивість — властивості анотації. Цю властивість можна використовувати для додавання інформації. Останнім кроком є перевірка аномалій або невідповідності дизайну онтології, який було завершено на попередніх етапах. Інструменти для перевірки є у вільному доступі. Для перевірки узгодженості структури онтології Moodle використовується резонер Pellet.

Рекомендація щодо змісту навчання складається з чотирьох кроків. По-перше, розрахунок семантичної релевантності обчислює семантичну подібність між студентом та змістом навчання, а потім відповідно створює список рекомендацій. По-друге, уточнення рекомендацій забезпечує інтерактивний спосіб коригування результату, доки не буде досягнуто кількох прийнятних варіантів. Коли студент вибирає один предмет із кандидатів, Learning Path Generation будує навчальний маршрут, що складається з попереднього запиту та цільового змісту навчання, який керує процесом навчання. Нарешті, доповнення рекомендацій об'єднує додатковий вміст, пов'язаний з основним курсом. Кожен крок рекомендації виконується шляхом використання знань про студента, знань про систему навчання або знань предметної області. База онтології забезпечує постійне зберігання та ефективний пошук таких знань.

Семантична релевантність базується на інтуїтивному уявленні про те, що ступінь релевантності між навчальною метою та змісту курсу зростає, чим вони ближчі, а про них більше відомо. Розрахувавши семантичну релевантність, ми можемо рекомендувати курс, семантична релевантність якого перевищує попередньо встановлене порогове значення. За допомогою запропонованого рекомендаційного підходу ми створили систему семантичних рекомендацій онлайн навчання. Вона була розроблена за допомогою мови Java.

Було протестовано накладні витрати на рекомендацію семантичного вмісту з точки зору часу відповіді. Завдяки цьому експерименту зроблено висновок, що наш підхід є легким і можливим для розгортання.



## ВИСНОВКИ

Оскільки онлайн навчання є процесом, що розвивається, а також вимоги до онлайн-навчань розвиваються щодня. Також було помічено, що кожна структура цифрового електронного навчання має недоліки. Це дуже важка робота для постачальників послуг, а також для науковців, щоб задовольнити всі вимоги студентів. Поєднання технологій семантичного вебу, а також інтелектуального оператора програмного забезпечення з можливістю налаштування для кожного студента є одним із шляхів вирішення цієї дилеми. Можна буде побудувати але успішні ініціативи цифрового електронного навчання в реальному світі за підтримки обох.

Персоналізація — це бачення Web 3.0, що є мережею наступного покоління. Розробка інструкцій - це сфера, якій досі не надається значення щодо семантичної мережі. Досягти цієї мети в службах електронного навчання є складним завданням. Концептуальна структура системи поєднує виявлення стилю навчання, онтологічні операції та управління знаннями для міркування та динамічної композиції послуг електронного навчання з використанням алгоритму планування, щоб забезпечити перелічені нижче переваги:

- відповідний робочий процес для динамічних учнів, поведінка яких змінюється з часом;
- співпраця з людьми зі схожим досвідом;
- забезпечує безперервне навчання;
- підтримує категорії стилів учнів;
- допомагає розробникам інструкцій створити найбільш відповідний дизайн для їхньої студентської спільноти.

У цьому контексті ми розглядаємо послуги електронного навчання як основну увагу. Але цей підхід можна застосувати до різних бізнес-доменів, де динамічна композиція є обов'язковою.

За результатами досліджень були розроблені онтології предметної області онлайн навчання за допомогою системи Moodle. Онтології були створені в середовищі Protégé та перевірені на узгодженість за допомогою Pellet. Система рекомендацій розроблена на мові Java.

Тестування системи відображає накладні витрати на рекомендацію семантичного вмісту з точки зору часу відповіді. Час роботи для кожного кроку, тобто створення списку рекомендацій вмісту, уточнення рекомендацій, створення шляху навчання та генерування пакету, було середнє значення 10 прогонів. Загальний час семантичної рекомендації становить менше 100 мс. Завдяки цьому експерименту зробили висновок, що наш підхід є легким і можливим для розгортання.

В даній кваліфікаційній роботі можна зробити висновок, що поставлені задачі були вирішені.

Практична цінність роботи полягає в тому, що на підставі досліджень семантичних технологій та стандартів онлайн навчання розроблені онтології предметної області та створене програмне забезпечення для рекомендацій курсів в залежності від мети навчання студента.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Шевчук Т., Кравчук Н. Стан і перспективи розвитку інформаційних технологій в Україні / Т. Шевчук, Н. Кравчук // *Scientific Bulletin of UNFU.* – 2018. – №28. – с. 114-118. DOI: 10.15421/40280922.
2. Беліков О., Ватаманюк Н., Струк В. Дистанційне навчання як принцип відкритого заняття / О. Беліков, Н. Ватаманюк, В. Струк // *Clinical Dentistry.* – 2020. – с. 78-81. DOI: 10.11603/2311-9624.2020.3.11574.
3. Яремчук Н., Лавро О. Дистанційне навчання у ЗВО: можливості та перспективи / Н. Яремчук, О. Лавро // *Молодий вчений.* – 2021. – с. 79-81. DOI: 10.32839/2304-5809/2021-8-96-17.
4. Гавриленко К. Нові виклики дистанційного навчання / К. Гавриленко // *Collection of Research Papers Pedagogical sciences.* – 2021. – с. 133-137. DOI: 10.32999/ksu2413-1865/2020-93-20.
5. Han, J., Sarica, S., Shi, F., and Luo, J. Semantic Networks for Engineering Design: State of the Art and Future Directions // *ASME. J. Mech. - Des.* February 2022. – Vol. 144(2). – pp. 020802. DOI: 10.1115/1.4052148.
6. John F. Sowa. Principles of Semantic Networks: Explorations in the Representation of Knowledge / Sowa F. Jonh. - Morgan Kaufmann Publishers, INC, 1991. – 565 p.
7. Roger T. Hartley, John A. Barnden, Semantic networks: visualizations of knowledge/ Hartley T. Roger, John A. Barnden // *Trends in Cognitive Sciences.* – 1997. - Volume 1, Issue 5. – pp. 169-175. DOI: 10.1016/S1364-6613(97)01057-7.
8. Ing-Long Wu, Pi-Jung Hsieh, Shwu-Ming Wu. Developing effective e-learning environments through e-learning use mediating technology affordance and constructivist learning aspects for performance impacts: Moderator of learner involvement / Ing-Long Wu, Pi-Jung Hsieh, Shwu-Ming Wu // *The Internet and Higher Education.* – 2022. - Volume 55. – pp. 100871. DOI: 10.1016/j.iheduc.2022.100871.

9. Kailash Kumar, Abdulaziz Al-Besher. IoT enabled e-learning system for higher education / Kailash Kumar, Abdulaziz Al-Besher // *Measurement: Sensors*. – 2022. – Volume 24. – pp. 100480. DOI: 10.1016/j.measen.2022.100480.
10. Veeramanickam M. R. M., P. Ramesh. Analysis on quality of learning in e-Learning platforms / Veeramanickam M. R. M., P. Ramesh // *Advances in Engineering Software*. – 2022. – Volume 172. – pp. 103168. DOI: 10.1016/j.advengsoft.2022.10316.
11. Стадній А.С. Моделі дистанційного навчання / А.С. Стадній // *Актуальні питання гуманітарних наук*. – 2022. – Т. 4, № 29. – с. 151-156.
12. Заблоцька А.Ю. Сучасний стан дистанційного навчання у ВНЗ України / А.Ю. Заблоцька // *Вісник Дніпропетровського університету імені Альфреда Нобеля. Серія: Педагогіка і психологія*. – 2016. – № 2. – с. 19-23.
13. Рижов О.А. Інтелектуальна адаптивна система дистанційного навчання ratos-ai®.(частина 1) концепція системи / О.А. Рижов // *Запорозький медичинський журнал*. – 2010. – №12.1. – с. 23-027.
14. Федорук П.І, Пікуляк М.В. Технологія побудови індивідуальної адаптивної траєкторії навчання у системі дистанційної освіти і контролю знань / П.І. Федорук, М.В. Пікуляк // *Математические машины и системы*. – 2010. – №1.1. – с. 68-76.
15. Коваленок І.А., Алізаде І. Системи дистанційного навчання: мета і завдання / І.А. Коваленко, І. Алізаде // *In: Science, innovations and education: problems and prospects*. CPN Publishing Group, 2021.
16. Anwar M. Designing an expert system for determining student learning styles using forward chaining in engineering education / M. Anwar // *Jurnal Konseling dan Pendidikan*. – 2021. – Vol. 9.1. – pp. 93-101.
17. Misra P. K. Learning About Learning / P.K. Misra // *In: Learning and Teaching for Teachers*. - Springer, Singapore, 2021. - pp. 1-15.
18. Halmann Heidi L., et al. Teacher Education's Moment: From Solution to Challenge // *Journal of Teacher Education*. – 2022. – Vol. 73.2. – pp.

127-128.

19. Bernard J., Popescu E., Graf S. Improving online education through automatic learning style identification using a multi-step architecture with ant colony system and Artificial Neural Networks / J. Bernard, E. Popescu, S. Graf // *Applied Soft Computing*/ - 2022. – pp. 109779.

20. Chandrasekaran D., Chang M., Graf S. A Learning Analytics Approach to Build Learner Profiles Within the Educational Game OMEGA+ / D. Chandrasekaran, M. Chang, S. Graf // *International Conference on Intelligent Tutoring Systems*. - Springer, Cham, 2022. - pp. 139-147.

21. Pranani Binar Kurnia, et al. Learning Management System (LMS) Research During 1991-2021: How Technology Affects Education // *International Journal of Emerging Technologies in Learning*. – 2022. - Vol. 17 Issue 17. – pp. 28-49.

22. Neumann W. Patrick, et al. Industry 4.0 and the human factor—A systems framework and analysis methodology for successful development // *International journal of production economics*. – 2021. – Vol. 233. – pp. 107992.

23. Setiawan A., Zuhri M., Zhang Y. Learning Management System Using Scorm and iSpring Free on Physics Learning / A. Setiawan, M. Zuhri, Y. Zhang // *JSEP (Journal of Science Education and Practice)*. – 2022 – Vol. 6(2). – pp. 95-106.

24. AlKhunzain A., Khan R. The Use of M-Learning: A Perspective of Learners' Perceptions on M-Blackboard Learn. *International Association of Online Engineering*. Retrieved November 29, 2022 [Electronic resource] - Access mode: <https://www.learntechlib.org/p/218918/>.

25. Bessadok A., Abousinahad E., Rabie O. Analyzing students' digital behavior in an e-learning environment within the blackboard learning management system / A. Bessadok, E. Abousinahad, O. Rabie // *Innovative and Intelligent Technology-Based Services for Smart Environments—Smart Sensing and Artificial Intelligence*. - CRC Press, 2021. - pp. 127-133.

26. Xin Ng Syuan, et al. A systematic review for online learning

management system // *Journal of Physics: Conference Series*. - IOP Publishing, 2021. - p. 012030.

27. Prakash R., Kumar S. E-Learning Framework for Semantic-Web // *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*. – 2021 – Vol. 12(14). – pp. 2994-3006.

28. Katsaris I., Vidakis N. Adaptive e-learning systems through learning styles: A review of the literature / I. Katsaris, N. Vidakis // *Advances in Mobile Learning Educational Research*. – 2021. Vol. 1(2). – pp. 124-145.

29. Пальоний А. С., Колівашко В. В. Method of adaptive pre-simulation training of air traffic controllers based on self-directed learning / A.C. Пальоний, В.В. Колівашко // *Aerospace technic and technology*. – 2022. Vol. 1. – pp. 64-77.

30. Hu H., Wei F., Wang L. Semi-supervised semantic segmentation via adaptive equalization learning // *Advances in Neural Information Processing Systems*. – 2021. Vol. 34. – pp. 22106-22118.

31. Banik Debajyoty, et al. Fabula: Hybridized Weightage Based Book Recommendation System // *Iberoamerican Knowledge Graphs and Semantic Web Conference*. - Springer, Cham, 2021.- pp. 182-196.

32. Rachana L., Shridevi S. A literature survey: semantic technology approach in machine learning / L/ Rachana, S. Shridevi // *Advances in Smart Grid Technology*. – 2021. – pp. 467-477.

33. Patel A., Jain S. Present and future of semantic web technologies: a research statement / A. Patel, S. Jain // *International Journal of Computers and Applications*, 2021. – Vol. 43(5). – pp. 413-422.

34. Van Assche Dylan, et al. Declarative RDF graph generation from heterogeneous (semi-) structured data: A systematic literature review // *Journal of Web Semantics*. – 2022. – pp. 100753.

35. Ali W., Saleem M., Yao B., Hogan, A., Ngomo A. A survey of RDF stores & SPARQL engines for querying knowledge graphs // *The VLDB Journal*. – 2021. – pp. 1-26.

**ДОДАТОК А****Файл опису онтології на мові OWL**

```

    <?xml version="1.0"?>
    <rdf:RDF
xmlns="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#"

xml:base="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"

    xmlns:xml="http://www.w3.org/XML/1998/namespace"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#">
    <owl:Ontology
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3"/>

```

```

    <!--

```

```

////////////////////////////////////
////////////////////////////////////

```

```

//
// Object Properties
//

```

```

////////////////////////////////////
////////////////////////////////////

```

```

-->

```

```

    <!--

```

```

http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Action_To -->

```

```

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Action_To"/>

```

```

    <!--

```

```

http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Name -->

```

```

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Name"/>

```



```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#Office -->
```

```
<owl:ObjectProperty  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-  
titled-ontology-3#Office"/>
```

```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#Semester -->
```

```
<owl:ObjectProperty  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-  
titled-ontology-3#Semester"/>
```

```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#Stud-ID -->
```

```
<owl:ObjectProperty  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-  
titled-ontology-3#Stud-ID"/>
```

```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#hasAccount -->
```

```
<owl:ObjectProperty  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-  
titled-ontology-3#hasAccount"/>
```

```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#hasActivites -->
```

```
<owl:ObjectProperty  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-  
titled-ontology-3#hasActivites"/>
```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#hasCourse -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#hasCourse"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#hasDate -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#hasDate"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#hasDocument -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#hasDocument"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#hasGroup -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#hasGroup"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#hasLecture -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#hasLecture">
        <rdfs:subPropertyOf
      rdf:resource="http://www.w3.org/2002/07/owl#topObjectProperty"/>
        <rdfs:domain>
          <owl:Restriction>

```

```

        <owl:onProperty
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#hasLecture"/>
        <owl:someValuesFrom
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Lecture"/>
        </owl:Restriction>
    </rdfs:domain>
</owl:ObjectProperty>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#hasParticipant -->

```

```

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#hasParticipant"/>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#hasRoom -->

```

```

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#hasRoom"/>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#hasStudent -->

```

```

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#hasStudent"/>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#hasTeacher -->

```

```

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#hasTeacher"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#isAuthors -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#isAuthors"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#isCourse -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#isCourse"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#isStudent -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#isStudent"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#isTeachers -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#isTeachers"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#offerCourse -->

```

```

      <owl:ObjectProperty
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un-
      titled-ontology-3#offerCourse"/>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#passedExam -->

```

```

    <owl:ObjectProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#passedExam"/>

```

```

    <!--

```

```

////////////////////////////////////
////////////////////////////////////

```

```

//
// Data properties
//

```

```

////////////////////////////////////
////////////////////////////////////

```

```

-->

```

```

    <!--

```

```

http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#hasID -->

```

```

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#hasID"/>

```

```

    <!--

```

```

http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#hasLogin -->

```

```

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#hasLogin"/>

```

```

    <!--

```

```

http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#hasName -->

```

```

    <owl:DatatypeProperty
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#hasName"/>

```



```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#Activity -->

```

```

      <owl:Class
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
      titled-ontology-3#Activity">
        <rdfs:subClassOf
      rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
      /untitled-ontology-3#Moodle"/>
      </owl:Class>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#Admin -->

```

```

      <owl:Class
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
      titled-ontology-3#Admin">
        <rdfs:subClassOf
      rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
      /untitled-ontology-3#Users"/>
      </owl:Class>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#Blog -->

```

```

      <owl:Class
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
      titled-ontology-3#Blog">
        <rdfs:subClassOf
      rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
      /untitled-ontology-3#Activity"/>
      </owl:Class>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#Books -->

```

```

      <owl:Class
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
      titled-ontology-3#Books">

```

```

        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Resource"/>
        </owl:Class>

```

```

        <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Chat -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Chat">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Activity"/>
        </owl:Class>

```

```

        <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Choice -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Choice">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Activity"/>
        </owl:Class>

```

```

        <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Course -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Course"/>

```

```

        <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Courses -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Courses">

```



```

        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Moodle"/>
        </owl:Class>

```

```

        <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#DataBase -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#DataBase">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Activity"/>
        </owl:Class>

```

```

        <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Department -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Department"/>

```

```

        <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Employ -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Employ">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Person"/>
        </owl:Class>

```

```

        <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Exam -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Exam">

```

```

        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Course"/>
        </owl:Class>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Exersice -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Exersice">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Course"/>
        </owl:Class>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#ExtLector -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#ExtLector">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Lechures"/>
        </owl:Class>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Forum -->

```

```

        <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Forum">
        <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Activity"/>
        </owl:Class>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Journals -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Journals">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Resource"/>
    </owl:Class>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Lechures -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Lechures">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Employ"/>
    </owl:Class>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Lecture -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Lecture">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Course"/>
    </owl:Class>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Moodle -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Moodle"/>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#NonEditTeacher -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#NonEditTeacher">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Users"/>
    </owl:Class>

```

```

    <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Note -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Note">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Forum"/>
    </owl:Class>

```

```

    <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Notes -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Notes">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Resource"/>
    </owl:Class>

```

```

    <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Person -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Person">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Department"/>
    </owl:Class>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#Professors -->

```

```

      <owl:Class
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
      titled-ontology-3#Professors">
        <rdfs:subClassOf
      rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
      /untitled-ontology-3#Lechures"/>
      </owl:Class>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#Research_Asisstent -->

```

```

      <owl:Class
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
      titled-ontology-3#Research_Asisstent">
        <rdfs:subClassOf
      rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
      /untitled-ontology-3#Employ"/>
      </owl:Class>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#Resource -->

```

```

      <owl:Class
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
      titled-ontology-3#Resource">
        <rdfs:subClassOf
      rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
      /untitled-ontology-3#Moodle"/>
      </owl:Class>

```

```

      <!--
      http://www.semanticweb.org/user/ontologies/2022/11/untitled-
      ontology-3#Student -->

```

```

      <owl:Class
      rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
      titled-ontology-3#Student">
        <rdfs:subClassOf
      rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
      /untitled-ontology-3#Person"/>
      </owl:Class>

```

```

    <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#StudentMoodle -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#StudentMoodle">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Users"/>
    </owl:Class>

```

```

    <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Student_Assisstent -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Student_Assisstent">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Student"/>
    </owl:Class>

```

```

    <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Teacher -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Teacher">
    <rdfs:subClassOf
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Users"/>
    </owl:Class>

```

```

    <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Technical_Asisstent -->

```

```

    <owl:Class
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Technical_Asisstent">

```



////////////////////////////////////  
////////////////////////////////////

-->

<!--

http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Gdanevich\_V.V. -->

<owl:NamedIndividual

rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Gdanevich\_V.V.">

<rdf:type

rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Person"/>

</owl:NamedIndividual>

<!--

http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Ivanov\_V.V. -->

<owl:NamedIndividual

rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Ivanov\_V.V.">

<rdf:type

rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Person"/>

</owl:NamedIndividual>

<!--

http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Petrov\_V.V. -->

<owl:NamedIndividual

rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Petrov\_V.V.">

<rdf:type

rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Person"/>

</owl:NamedIndividual>

<!--

http://www.semanticweb.org/user/ontologies/2022/11/untitled-ontology-3#Polyanski\_A.A. -->



```

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Polyanski_A.A.">
    <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Person"/>
    </owl:NamedIndividual>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Гудименко_А.В. -->

```

```

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Гудименко_А.В.">
    <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Student"/>
    </owl:NamedIndividual>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Колумбів_А.В. -->

```

```

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Колумбів_А.В.">
    <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Student"/>
    </owl:NamedIndividual>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Кузякін_М.Т. -->

```

```

    <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Кузякін_М.Т.">
    <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#ExtLector"/>
    </owl:NamedIndividual>

```

```

      <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Ньятон_К.В. -->

```

```

      <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Ньятон_К.В.">
      <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Student"/>
      </owl:NamedIndividual>

```

```

      <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Поляк_М.Т. -->

```

```

      <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Поляк_М.Т.">
      <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Student"/>
      </owl:NamedIndividual>

```

```

      <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Полянський_Т.Р. -->

```

```

      <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Полянський_Т.Р.">
      <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Student"/>
      </owl:NamedIndividual>

```

```

      <!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Програмних_засобів -->

```

```

      <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Програмних_засобів">
      <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11
/untitled-ontology-3#Department"/>
      </owl:NamedIndividual>

```

```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#Соборний_Т.В. -->
```

```
<owl:NamedIndividual  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un  
titled-ontology-3#Соборний_Т.В.">  
  <rdf:type  
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11  
/untitled-ontology-3#Student"/>  
</owl:NamedIndividual>
```

```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#Степаненко_О.О. -->
```

```
<owl:NamedIndividual  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un  
titled-ontology-3#Степаненко_О.О.">  
  <rdf:type  
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11  
/untitled-ontology-3#Professors"/>  
</owl:NamedIndividual>
```

```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#Суботин_С.О. -->
```

```
<owl:NamedIndividual  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un  
titled-ontology-3#Суботин_С.О.">  
  <rdf:type  
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11  
/untitled-ontology-3#Professors"/>  
</owl:NamedIndividual>
```

```
<!--  
http://www.semanticweb.org/user/ontologies/2022/11/untitled-  
ontology-3#Табунщик_Г.В. -->
```

```
<owl:NamedIndividual  
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un  
titled-ontology-3#Табунщик_Г.В.">
```

```

        <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11/
untitled-ontology-3#Professors"/>
        </owl:NamedIndividual>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Третьяк_І.В. -->

```

```

        <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Третьяк_І.В.">
        <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11/
untitled-ontology-3#Student"/>
        </owl:NamedIndividual>

```

```

<!--
http://www.semanticweb.org/user/ontologies/2022/11/untitled-
ontology-3#Трубочинін_К.П. -->

```

```

        <owl:NamedIndividual
rdf:about="http://www.semanticweb.org/user/ontologies/2022/11/un
titled-ontology-3#Трубочинін_К.П.">
        <rdf:type
rdf:resource="http://www.semanticweb.org/user/ontologies/2022/11/
untitled-ontology-3#ExtLector"/>
        </owl:NamedIndividual>
</rdf:RDF>

```

```

<!-- Generated by the OWL API (version 4.5.9.2019-02-
01T07:24:44Z) https://github.com/owlcs/owlapi -->

```

**ДОДАТОК Б****Сервіс рекомендацій на основі онтологій**

## Лістинг Б.1 - LocalReasoner.java

```

import com.hp.hpl.jena.ontology.OntClass;
import com.hp.hpl.jena.ontology.OntModel;

import java.util.ArrayList;
import java.util.List;
import java.util.OptionalLong;
import java.util.concurrent.atomic.AtomicLong;
import java.util.concurrent.atomic.AtomicReference;

public class LocalReasoner {
    public static double checkSim(OntModel first, OntModel
second) {
        long depthFirst = countDepth(first);
        long depthSecond = countDepth(second);
        AtomicReference<Double> result = new
AtomicReference<>(Double.valueOf(0));
        AtomicLong equalClassesCounter = new AtomicLong(0);
        first.listClasses().forEachRemaining(c1 -> {
            second.listClasses().forEachRemaining(c2 -> {
                double currentResult;
                if (equalURI(c1, c2)) {
                    currentResult = 1.0;
                } else {
                    if (equalNames(c1, c2)) {
                        currentResult = lexEquality(c1,
depthFirst, c2, depthSecond);
                    } else currentResult = 0.0;
                }
                result.getAndUpdate(value -> value +=
currentResult);
                if (currentResult > 0.0) {
                    equalClassesCounter.incrementAndGet();
                }
            }
        }
    }
}

```

```

        });
    });
    long totalClassesInUnion = countClasses(first) +
countClasses(second) - equalClassesCounter.get();
    return result.get() / totalClassesInUnion;
}

private static long countDepth(OntModel model) {
    List<OntClass> classes = new ArrayList<>();
    model.listClasses().forEachRemaining(c) ->
classes.add(c));
    OptionalLong result = classes.parallelStream()
        .filter(ontClass ->
!ontClass.isHierarchyRoot()) ->
        .mapToLong(ontClass -> distance(ontClass))
        .max();
    if (result.isPresent()) {
        return result.getAsLong();
    }
    return 0;
}

private static long distance(OntClass ontClass) {
    if (ontClass.isHierarchyRoot()) return 0;
    return distance(ontClass.getSuperClass()) + 1;
}

private static boolean equalNames(OntClass c1, OntClass
c2) {
    return c1.getURI().equals(c2.getURI());
}

private static long countClasses(OntModel model) {
    AtomicLong counter = new AtomicLong(0);
    model.listClasses().forEachRemaining(c ->
counter.incrementAndGet());
}

```

```

        return counter.get();
    }

    private static boolean equalURI(OntClass c1, OntClass
c2) {
        String firstURI = constructURI(c1);
        String secondURI = constructURI(c2);
        return firstURI.equals(secondURI);
    }

    private static String constructURI(OntClass ontClass) {
        if (ontClass.isHierarchyRoot()) return
ontClass.getNameSpace().concat(ontClass.getURI());
        return
constructURI(ontClass.getSuperClass()).concat(ontClass.getURI())
;
    }

    private static Double lexEquality(OntClass c1, long
depthFirst, OntClass c2, long depthSecond) {
        Double result = Math.log(2 * depthFirst /
distance(c1));
        result /= Math.log(2 * depthSecond / distance(c2));
        return result;
    }
}

```

### ЛІСТИНГ Б.2 - Main.java

```

import com.hp.hpl.jena.ontology.OntClass;
import com.hp.hpl.jena.ontology.OntModel;
import com.hp.hpl.jena.ontology.OntModelSpec;
import com.hp.hpl.jena.rdf.model.ModelFactory;

public class Main {

```



```

public static void main(String[] args) {
    String NS = "http://www.example.org/rdf/";
    OntModel          m1          =
ModelFactory.createOntologyModel (OntModelSpec.OWL_MEM);
    m1.setNsPrefix("class", NS);
    OntClass          cl          =
m1.createClass(m1.expandPrefix("first"));
    cl.addSubClass(m1.createClass("Child1"));
    cl.addSubClass(m1.createClass("Child2"));
    cl.addSubClass(m1.createClass("Child3"));
    cl.addSubClass(m1.createClass("Child4"));
    //      m1.write(System.out, "Turtle");

    OntModel          m2          =
ModelFactory.createOntologyModel (OntModelSpec.OWL_DL_MEM);
    m2.createClass("second");
    cl = m2.createClass("second");
    cl.addSubClass(m2.createClass("Child1"));
    cl.addSubClass(m2.createClass("Child"));
    cl.addSubClass(m2.createClass("Child3"));
    cl.addSubClass(m2.createClass("Child4"));
    cl.addSubClass(m2.createClass("Child5"));

    cl.getSubClass().addSubClass(m2.createClass("Child2"));
    //      m2.write(System.out, "Turtle");

    OntModel          m3          =
ModelFactory.createOntologyModel (OntModelSpec.OWL_DL_MEM);
    m3.createClass("third");
    cl = m3.createClass("a");
    cl.addSubClass(m3.createClass("b"));
    cl.addSubClass(m3.createClass("c"));
    cl.addSubClass(m3.createClass("d"));
    cl.addSubClass(m3.createClass("e"));
    cl.addSubClass(m3.createClass("f"));
    cl.getSubClass().addSubClass(m3.createClass("g"));

```

```

//          m3.write(System.out, "Turtle");

        checkSim(m2, m2);
        performanceCheck(new int[]{15, 20, 30, 50, 80, 100,
120, 150, 200});
        OntModel m4 = createOntology(200);
        OntModel m5 = createOntology(20);
        checkSim(m4, m5, "Size differs greatly, first
try");
        checkSim(m5, m4, "Size differs greatly, second
try");
    }

    private static OntModel createOntology(int classes) {
        OntModel          model          =
ModelFactory.createOntologyModel(OntModelSpec.OWL_DL_MEM);
        for (; classes > 0; classes--) {
            model.createClass("class" + classes);
        }
        return model;
    }

    private static void checkSim(OntModel first, OntModel
second, String info) {
        long time = System.currentTimeMillis();
        double result = LocalReasoner.checkSim(first,
second);
        time = System.currentTimeMillis() - time;

        private static void performanceCheck(int[] classes) {
            for (int i = 0; i < classes.length; i++) {
                OntModel model = createOntology(classes[i]);
                checkSim(model, model, "Same ontology " +
classes[i] + " classes");            } }}

```

**ДОДАТОК В**  
**Слайди презентації**

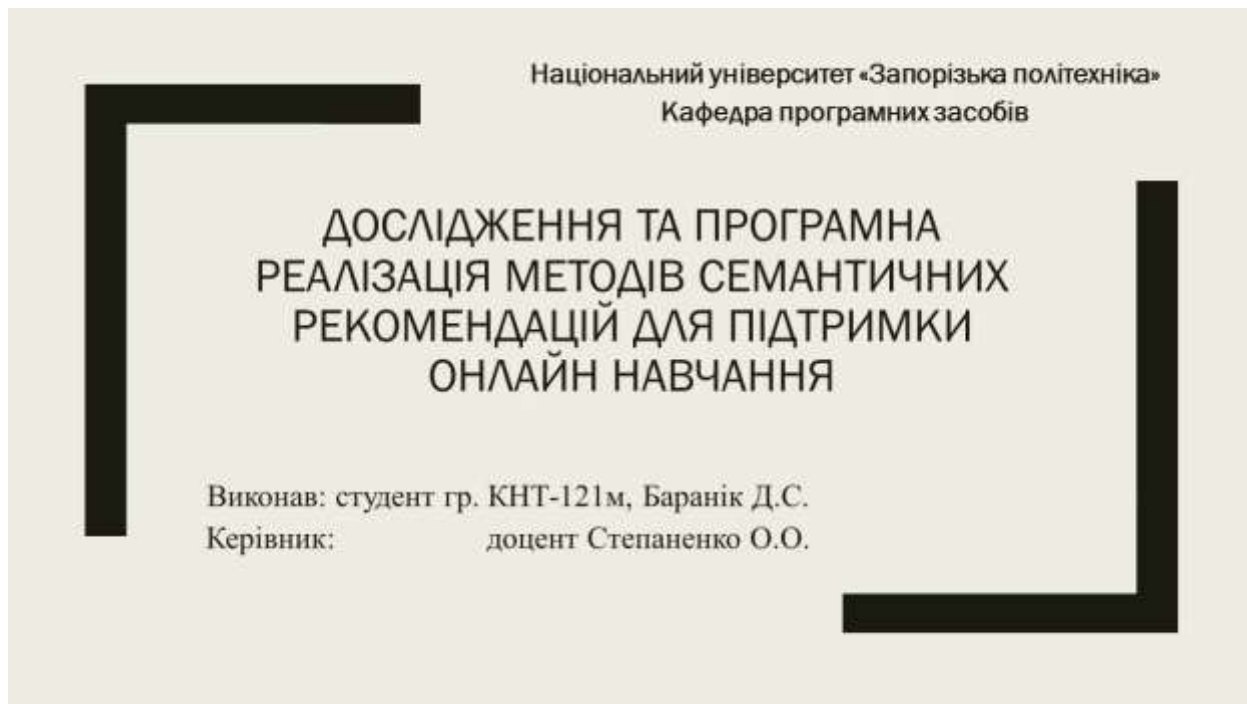


Рисунок В.1 – Слайд 1

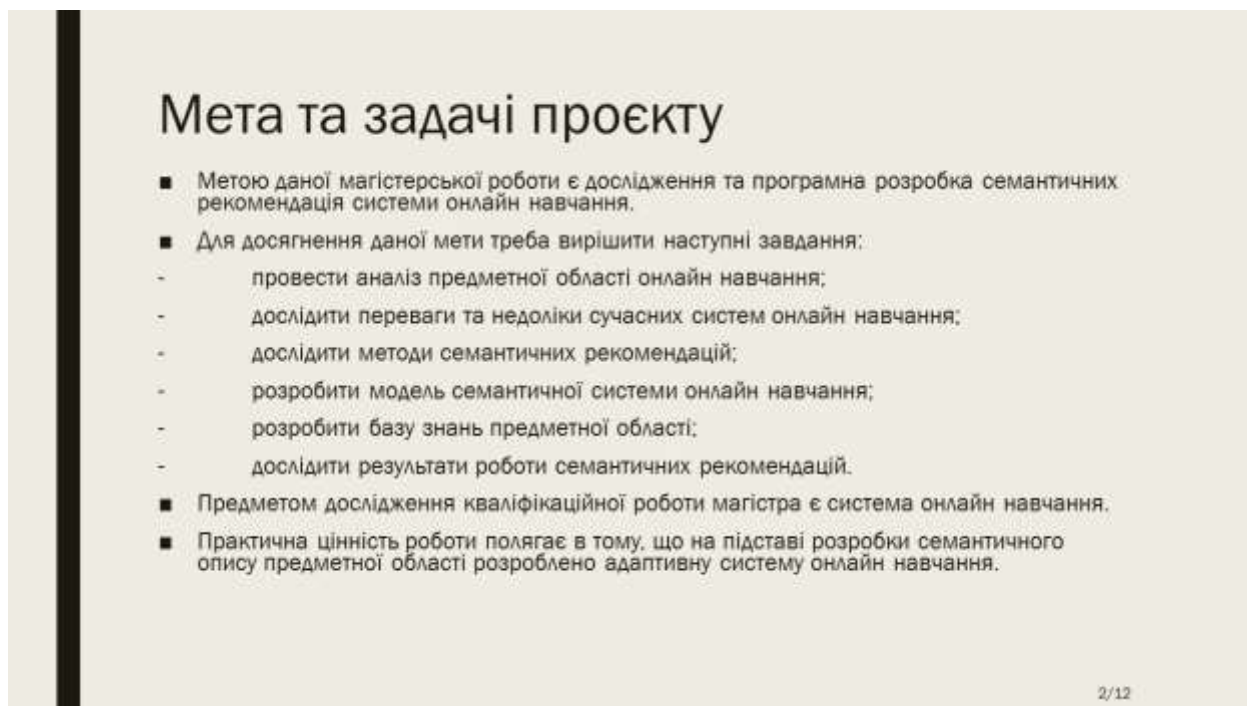
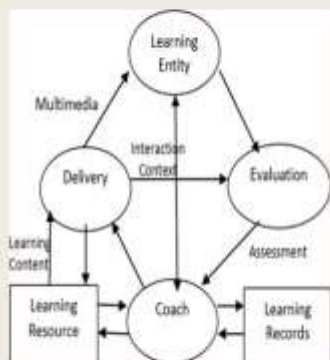
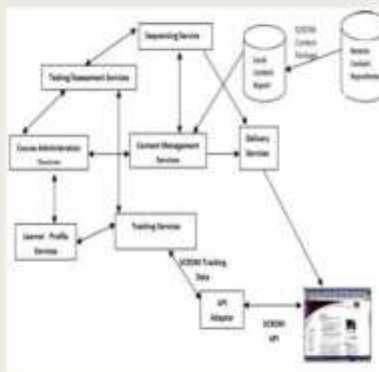


Рисунок В.2 – Слайд 2

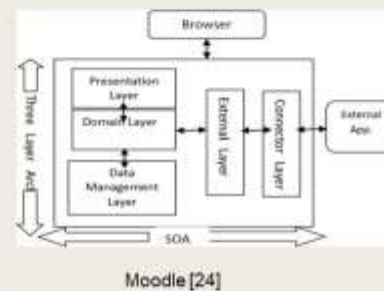
## Стандарти онлайн навчання



IEEE Learning Technology Systems Framework [22]



Sharable Content Object Reference Model (SCORM) [23]



Moodle [24]

3/12

Рисунок В.3 – Слайд 3

## Порівняння мов програмування для роботи з онтологіями

Функції	Java	Ruby	Python
Функції GUI	+	+	+
Підтримка XML	+	+	+
Підтримка OWL/RDF	+	+	-
Робота з базами даних	+	+	+
Робота з Protege	+	-	-
Функції роботи з SPARQL	+	+	-

4/12

Рисунок В.4 – Слайд 4

## Порівняння використання семантичної мережі та електронного навчання студента

Вимога	Електронне навчання	Семантична мережа
Розмір	Результативні результати порівняно великі	Об'єкти знань (металева інтеграція) поширюються в Інтернеті, але пов'язані з емоційними, які викликають інтерес. Це дозволяє швидко оцінювати якість виконання та, створює когнітивний навчальний простір.
Цілі	Навчальні цілі вивчають на розробленій платформі	Широка проблема знань існує у певних межах, але адаптується до потреб студента. Знання передаються в семантичій мережі, що дозволяє створювати як контент, а також використовувати знання як інструмент в розробці навчального процесу. Мета знань в тому, щоб вони користуватися на власному навчальному шляху для навчання з інших знань.
Доступ	Навчальні – дозволяють надавати будь-яку інформацію в будь-якій формі, яка має повний доступ до знань.	Користувач повинен визначити шлях проблеми (якщо знання, пов'язані з іншими) і викласти семантичну мережу для використання знань. Доступ до інформації можна розширити на повільно семантично зв'язаного утворення.

1	2	3
Семантика	Семантичне навчання є результатом процесу, який включає практику.	Семантична мережа є результатом процесу, який включає практику, створюючи платформу.
Мотивація	Використання семантичного навчання, яке включає практику, але не включає знань.	Адаптивне навчання є результатом процесу, який включає знання, а також практику, створюючи платформу.
Адаптивність	Розширений доступ до знань, який включає знання, але не включає знань.	Семантична мережа є результатом процесу, який включає знання, а також практику, створюючи платформу.
Персоналізація	Персоналізація є результатом процесу, який включає знання, але не включає знань.	Ключові знання включають знання, а також практику, створюючи платформу.
Адаптивність	Адаптивність – це результативний процес, який включає знання, але не включає знань.	Семантична мережа є результатом процесу, який включає знання, а також практику, створюючи платформу.

5/12

Рисунок В.5 – Слайд 5

## Загальна архітектура системи



Рисунок - Концептуальна модель адаптивної системи онлайн навчання

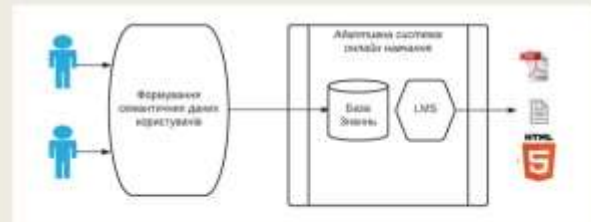


Рисунок – Структурна модель системи

6/12

Рисунок В.6 – Слайд 6

## Модулі семантичної системи онлайн навчання

Назва модулю 1	Опис 2
Модуль введення	Цей модуль базується на техніці, яка використовує поєднання як спільних, так і автоматичних концепцій, де поведінка, взаємодія та ставлення учня відстежуються на основі певного списку дій.
Модуль отримання семантики	Цей модуль не залучає користувачів. Цей модуль автоматично виклав доступні навчальні об'єкти та приймає онтології для цього предмета, указанного користувачем.
Модуль вилучення знань/визначення стилю учня	Онтології, отримані з модуля отримання знань, і відомості про профіль/історію учня використовуються для визначення типу учня, інших слів, визначаються переваги учня щодо навчання. Наша методика використовує методологію, у якій поведінка, дії, взаємодія та профіль учня (якщо він уже існує в базі знань) зберігаються в термінах онтологій. Семантика, отримана з онтологій, є вхідними для визначення виного учня.
Модуль створення потоку робіт	Профіль та онтологія процесу використовуються для динамічного створення сервісів. Планувальник або композитор відповідає за створення плану робочого процесу.

7/12

Рисунок В.7 – Слайд 7

## Підсистема онтологій

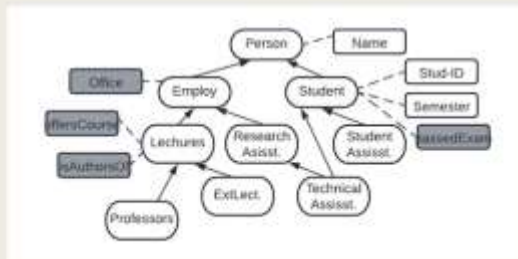


Рисунок - Онтологія кафедри – гілка особи

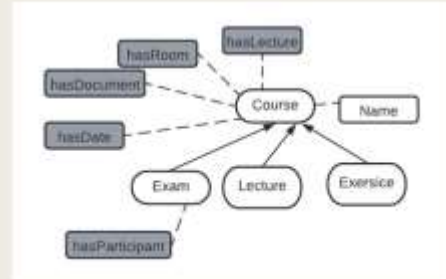
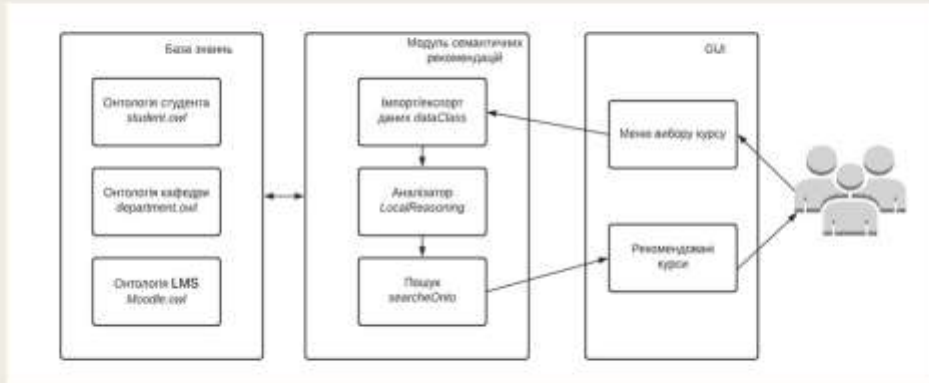


Рисунок – Онтологія кафедри – гілка курсу

8/12

Рисунок В.8 – Слайд 8

## Функціональна схема додатку



9/12

Рисунок В.9 – Слайд 9

## Рекомендації щодо семантичного змісту



10/12

Рисунок В.10 – Слайд 10



## Результати

- Протестовано накладні витрати на рекомендацію семантичного вмісту з точки зору часу відповіді. Час роботи для кожного кроку, тобто створення списку рекомендацій вмісту, уточнення рекомендацій, створення шляху навчання та генерування пакету, було середнє значення 10 прогонів. Позначено, що час для створення списку рекомендацій вмісту був найбільшим, скажімо, 78 мс; генерація шляху навчання зайняла 16 мс; як очищення, так і створення упаковки займають менше 1 мс. Таким чином, загальний час семантичної рекомендації становить менше 100 мс. Завдяки цьому експерименту зроблено висновок, що підхід є легким і можливим для розгортання.



11/12

Рисунок В.11 – Слайд 11

## Висновки

- За результатами досліджень були розроблені онтології предметної області онлайн навчання за допомогою системи Moodle. Онтології були створені в середовищі Protégé та перевірені на узгодженість за допомогою Pellet. Система рекомендацій розроблена на мові Java.
- Тестування системи відображає накладні витрати на рекомендацію семантичного вмісту з точки зору часу відповіді. Час роботи для кожного кроку, тобто створення списку рекомендацій вмісту, уточнення рекомендацій, створення шляху навчання та генерування пакету, було середнє значення 10 прогонів. Загальний час семантичної рекомендації становить менше 100 мс. Завдяки цьому експерименту зроблено висновок, що наш підхід є легким і можливим для розгортання.
- В даній кваліфікаційній роботі можна зробити висновок, що поставлені задачі були вирішені.
- Практична цінність роботи полягає в тому, що на підставі досліджень розроблено семантичних технологій та стандартів онлайн навчання розроблені онтології предметної області та створене програмне забезпечення для рекомендацій.

12/12

Рисунок В.12 – Слайд 12