НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
Факультет прикладної математики
Кафедра прикладної математики

Пояснювальна записка
до курсової роботи
із дисципліни “Бази даних та інформаційні системи”
на тему
“**Бот для збирання інформації з вказаних ресурсів за темою лекції**”

Виконав:
студент групи КМ-62
Козирєв А. Ю.

Перевірили:
Терещенко І. О.
Ковальчук-Химюк Л. О.

Київ — 2019

ЗАВДАННЯ НА ВИКОНАННЯ КУРСОВОЇ РОБОТИ

Основним завданням курсової роботи є створення багатокористувацького сервісу архітектури “клієнт-сервер”, головна задача якого є генерація тексту лекції за темою вказаною користувачем, а також надання можливості зберегти згенеровані ресурси та лекції, редагувати та видаляти їх. За допомогою технології машинного навчання користувачу необхідно лише вказати тему лекції (як словосполучення або кілька словосполучень), яку він хоче згенерувати.

АНОТАЦІЯ

Додаток буде дуже популярним серед студентів та викладачів університетів. Більше не матиме необхідності шукати в інтернеті різні ресурси та перечитувати велику кількість літератури задля створення лекцій, тепер системи автоматизації будуть виконувати роботу по аналізу та написанню текстів. Для отримання “brand-new” лекції лише треба зробити одну просту дію: визначитись з темою лекції.

РЕФЕРАТ

У студентів та викладачів багато часу для підготовки доповідей та лекцій саме фільтрація контенту та групування по змісту. Теперь все це можна довірити штучному інтелекту! Нейронна мережа може генерувати осмислені корисні тексти, і створювати унікальний контент на задану тематику. Вже зараз такі відомі проекти як Botnic.AI можуть генерувати сіквели до серії книг “Гаррі Поттер”. Але нейромережі можуть генерувати не тільки нові романи, тексти пісень, а ще й ресурси для навчання. Незважаючи на великий потенціал даної сфери, подібні додатки до сих пір викликають недовіру серед користувачів в мережі Інтернет; все через єдиний вагомий недолік: *нейронні мережі є непередбачуваними*. У сучасних мережах знаходяться мільярди компонентів-нейронів, кожен з яких так чи інакше впливає на результат обчислення, тому точно визначити алгоритм, за яким працює мережа майже неможливо. Але у сфери машинного навчання великий потенціал, саме тому вони мають місце у сучасних мікро-сервісних додатках. Принцип данного додатка є простим: опишіть тему лекції - отримайте вашу лекцію. І нічого зайвого! Штучний інтелект все зробить за вас!

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ.....6
ВСТУП…….....7
ОСНОВНА ЧАСТИНА……....7
1. АНАЛІЗ ПІДПРИЄМСТВА АВТОМАТИЗАЦІЇ 8
a. Передпроектне дослідження 8
b. Мета 8
2. ПОСТАНОВКА ЗАДАЧІ.....9
a. Визначення категорії користувачів.....9
b. Бізнес-правила…….....10
c. Класи даних....…….....11
d. Матриця елементарних подій…..…….....14
3. МОДЕЛЮВАННЯ БІЗНЕС-ПРОЦЕСІВ.....16
a. Use-case.....16
b. Component diagram.....17
c. UML-diagram.....18
4. ІНФОЛОГІЧНЕ ПРОЕКТУВАННЯ 19
a. Моделі діючих сутностей 19
5. ВИСНОВКИ.....21
6. СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ.....22

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

БД - база даних;

DAL - Data Access Layer, рівень доступу до даних;

BLL - Business Logic Layer, рівень бізнес логіки;

API - Applied Programing Interface, прикладний інтерфейс програмування;

REST API - Representational State Transfer Applied Programing Interface (передача стану представлення);

U.S. – user story (критерій);

U.C. – use-case (сценарій виконання);

ВСТУП

Відповідно до поставленої задачі необхідно розробити клієнт-серверний додаток для генерації лекцій та редагування їх. Архітектура серверної частини виконана у форматі 3-х рівневої архітектури: API-BLL-DAL. Основною задачею рівня DAL є виконання запитів у БД та виконання операції mapping з визначиними сутностями, для BLL - обробка вибраних з БД сутностей, виключення “навігаційних” атрибутів та передача отриманих даних як скалярні значення або спеціальні DTO-класи, для API - визначення єдиного інтерфейсу передачі даних для клієнтської частини. Основною технологією серверної частини є мікро-фреймворк Flask, найвищий рівень архітектури серверної частини виконаний у стилі RESTful API. Для редагування даних використовується бібліотека (додаток для Flask) WTForms. Для збереження даних була обрана технологія реляційної бази даних PostgreSQL 12. Додатковим завданням до курсової роботи є інтеграція сервісу з не реляційною БД Apache Cassandra З МЕТОЮ ПРИШВИДШЕННЯ ПРОЦЕСУ НАВЧАННЯ НЕЙРОННИХ МЕРЕЖ. Клієнтська частина виконання за стеком технологій: Bootstrap 4, JQuery;

ОСНОВНА ЧАСТИНА

АНАЛІЗ ПІДПРИЄМСТВА АВТОМАТИЗАЦІЇ

Передпроектне дослідження

Дано датасет з текстів на різні теми. Основною задачею є дослідження кореляції між парою (поточне слово/словосполучення, тема лекції) -> наступне слово/словосполучення. При знаходженні алгоритму з найбільшим коефіцієнтом кореляції, необхідно інтегрувати алгоритм з сервісом по генерації тексту відповідно до теми. Загальним алгоритмом сервісу стає: користувач вводить тему лекції -> користувач отримує текст лекції.

Мета

1. Автоматизація процесу аналізу тексту та перетворення його до логічного та релевантного вигляду;
2. Залучення основної аудиторії додатку: студентів та лекторів;
3. Полегшення процесу готування текстів для зачитування лекції викладачам, а також для доступу до ресурсів студентам;

ПОСТАНОВКА ЗАДАЧІ

Оскільки генерація нейромережею нової лекції виконується за основі раніше зібраних даних, важливо періодично робити перенавчання даної мережі, виконуючи вибірку з БД. Саме для даної задачі використання не реляційної моделі є ключовим, оскільки такий тип даних найкраще підходить для машинного навчання завдяки швидкому доступу до даних порівняно з реляційною моделлю.

Нейронна мережа для генерації лекції винесена в окремий мікросервіс, що викликається головним сервісом додатку. Такий підхід дає можливість масштабувати додаток та додавати нові сервіси.

Основні вхідні дані для мікросервісу-генератора: логін користувача, токен користувача та тема лекції.

Вихідні дані для мікросервісу-генератора: отриманий контент лекції.

Задача кореляційного аналізу: дослідження залежності пари (слово/словосполучення, тема) та наступного слова/словосполучення.

Визначення категорії користувачів

Основними користувачами даного сервісу є студенти 1-6 курсу, лектори та наукові діячі.

Бізнес-правила

1. Користувач може зберегти лекцію лише за наявності заголовка та контенту лекції;
2. На сторінці автоматичної генерації лекції користувач може як зберегти готову лекцію так і проредагувати її;
3. Для будь-якого нового користувача надана йому роль буде **Default user**;
4. Для довільних дій у даному додатку (окрім реєстрації) користувачу необхідно попередньо пройти аутентифікацію;
5. Користувач може зберегти лекцію без автоматичної генерації лише якщо її заголовок є унікальним серед усіх лекцій, якими володіє користувач (тобто не має бути заголовків, що дублюються);
6. Користувач може зберегти лекцію без автоматичної генерації, якщо довжина заголовку не перевищує 250 символів, а довжина контенту не перевищує 5000 символів;
7. Довільна збережена лекція має початковий стан **Pending**;
8. Користувач може мати лише одну єдину роль у певний час;
9. Користувач може змінювати статус лекції лише за наявності ролі **Moderator**;
10. Користувач може задавати ролі іншим користувачам лише за наявності ролі **Admin**;

Класи даних

Зауваження: класи даних представлені за реляційною моделлю

Таблиця 2.1 - Клас даних “Роль користувача”

Сутність	Роль користувача	
Опис сутності	Сутність, що визначає привелегій користувача	
Атрибути сутності	**Опис атрибуту**	**Пов'язана сутність**
Ідентифікатор	Первинний ключ даної сутності	-
Назва	Текстове поле, що буде відображатися для користувача	-
Пріоритет	Визначення порядкового номера для надання відповідних привелегій	-

Таблиця 2.2 - Клас даних “Користувач”

Сутність	Користувач	
Опис сутності	Сутність, що буде ідентифікувати власне користувача та його дії на сервісі	
Атрибути сутності	**Опис атрибуту**	**Пов'язана сутність**
Логін	Унікальне ім'я для кожного користувача	-
Ідентифікатор ролі	Атрибут, що визначає єдину роль для користувача	Роль користувача
Хеш паролю	Масив байтів розмірності 256, зашифрована форма паролю користувача	-
Дата реєстрації	Дата, коли виконана реєстрація на сервісі	-

Таблиця 2.3 - Клас даних “Лекція”

Сутність	Лекція	
Опис сутності	Текстовий матеріал за певною темою, що належить користувачеві.	
Атрибути сутності	**Опис атрибуту**	**Пов'язана сутність**
Ідентифікатор	Первинний ключ даної сутності	-
Логін користувача	Ім'я користувача, що володіє лекцією	Користувач
Заголовок	Тема лекції	-

Контент	Власне лекція	-

Таблиця 2.4 - Клас даних “Ресурс”

Сутність	Ресурс	
Опис сутності	Посилання на текстовий інтернет-ресурс	
Атрибути сутності	**Опис атрибуту**	**Пов'язана сутність**
URL	Посилання на ресурс	-
Опис	Короткий опис ресурсу, що задає користувач	-
Кількість відвідувань	Кількість разів переходу на ресурс	-

Таблиця 2.5 - Клас даних “Користувач має ресурси”

Сутність	Користувач має ресурси	
Опис сутності	Навігаційна сутність між сутностями “Ресурс” та “Користувач”	
Атрибути сутності	**Опис атрибуту**	**Пов'язана сутність**
Логін користувача	Ім'я користувача, що володіє лекцією	Користувач
URL ресурсу	Посилання на ресурс	Ресурс

Матриця елементарних подій

Таблиця 2.6 - Процес “C.R.U.D. операції над сутностями”

Назва процесу	C.R.U.D. операції над сутностями
Сутності	Всі вище перелічені сутності
Вхідні атрибути сутності	Усі атрибути заданої сутності
Опис функціоналу	Виконання операцій додавання, редагування та видалення записів із таблиць.
Змінені атрибути сутності	Усі рядки відповідної сутності

Таблиця 2.7 - Процес “Автоматична генерація лекції”

Назва процесу	Автоматична генерація лекції
Сутності	Користувач, Лекція
Вхідні атрибути сутності	Логін користувача, заголовок лекції
Опис функціоналу	Генерація нейромережею контенту для лекції, відповідно до заданого заголовку
Змінені атрибути сутності	Контент лекції

Таблиця 2.9 - Процес “Підтвердження лекції”

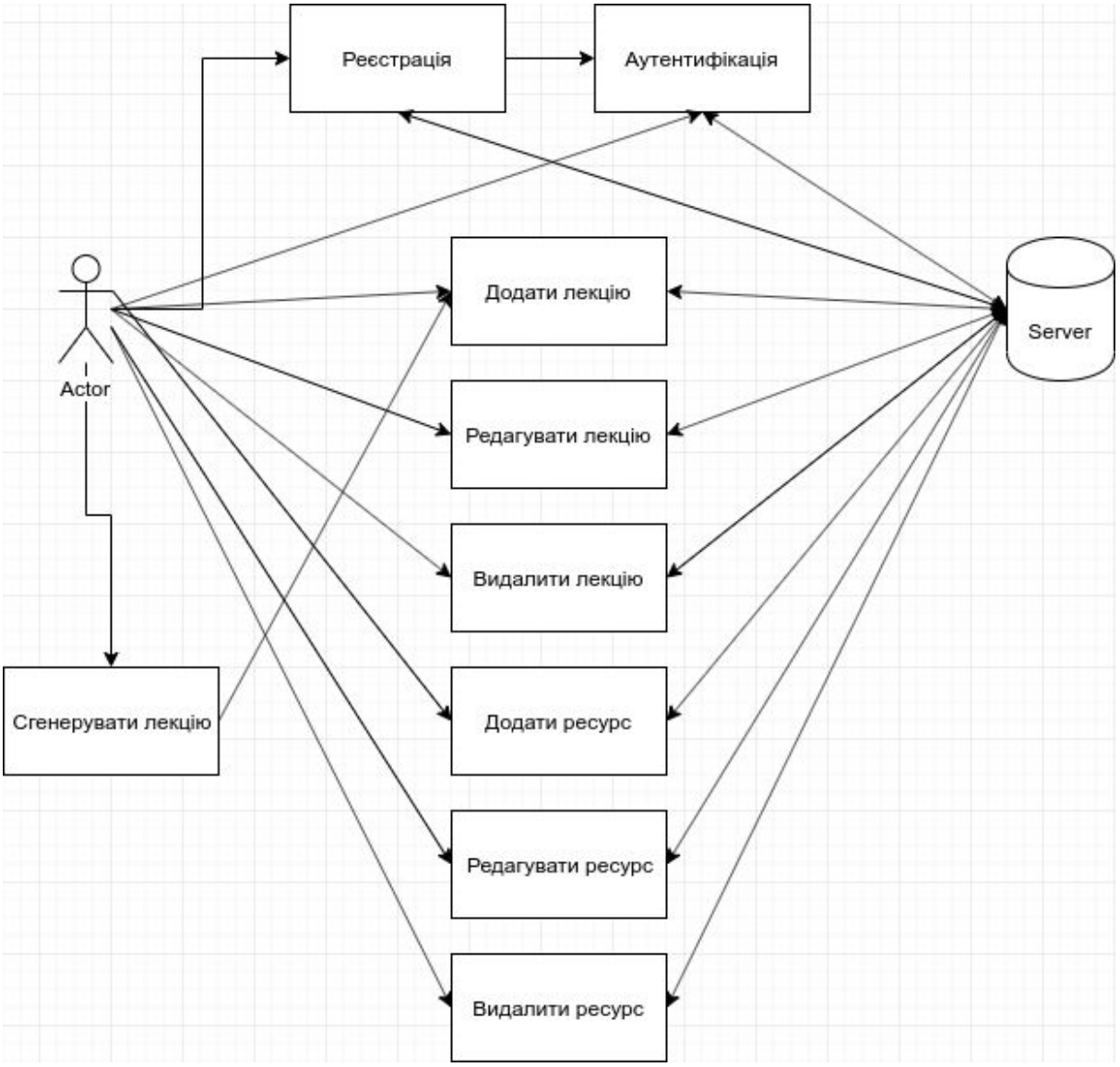
Назва процесу	Підтвердження лекції
Сутності	Користувач, Лекція
Вхідні атрибути сутності	Роль користувача, статус лекції
Опис функціоналу	При наявності у користувача відповідної ролі (Модератор), зміна статусу на Затверджена або Відхилена
Змінені атрибути сутності	Статус лекції

Таблиця 2.8 - Процес “Зміна ролі користувача”

Назва процесу	Зміна ролі користувача
Сутності	Користувач
Вхідні атрибути сутності	Роль користувача, логін користувача
Опис функціоналу	При наявності у користувача відповідної ролі (Адміністратор), зміна ролі у іншого користувача
Змінені атрибути сутності	Роль користувача

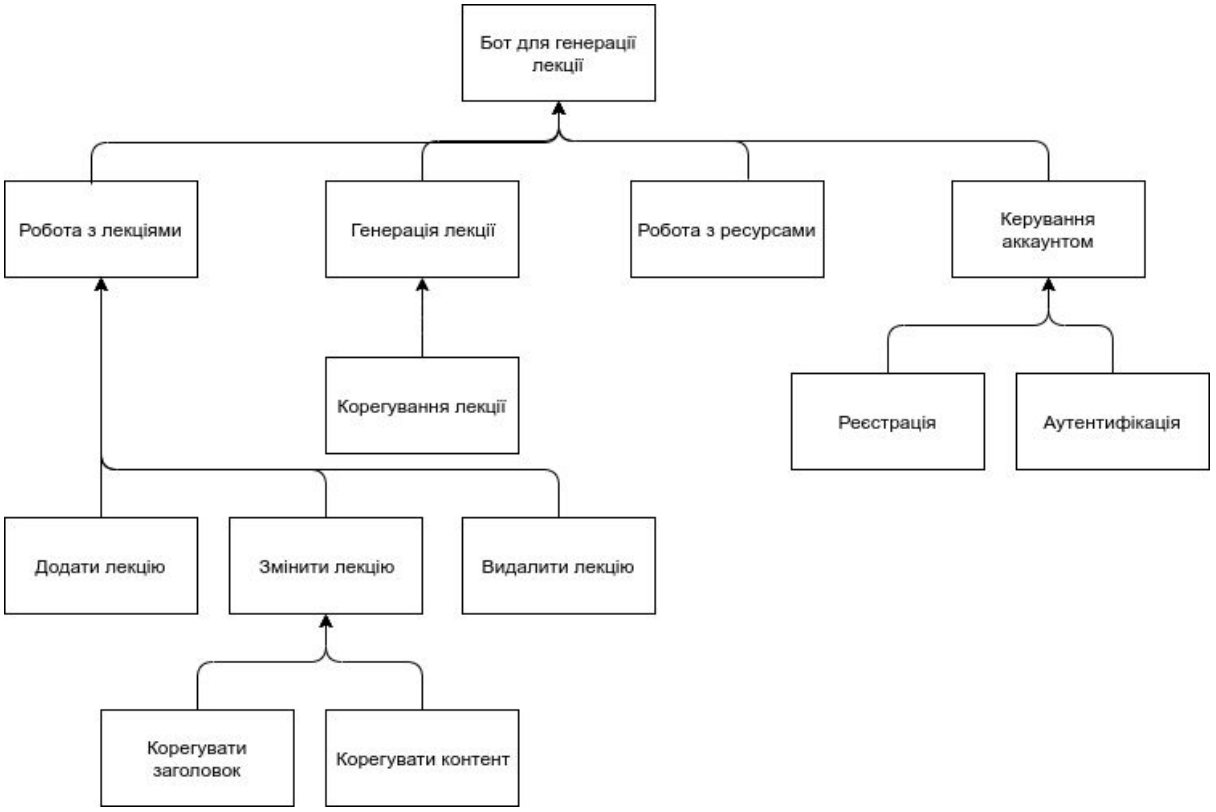
МОДЕЛЮВАННЯ БІЗНЕС-ПРОЦЕСІВ

Use-case



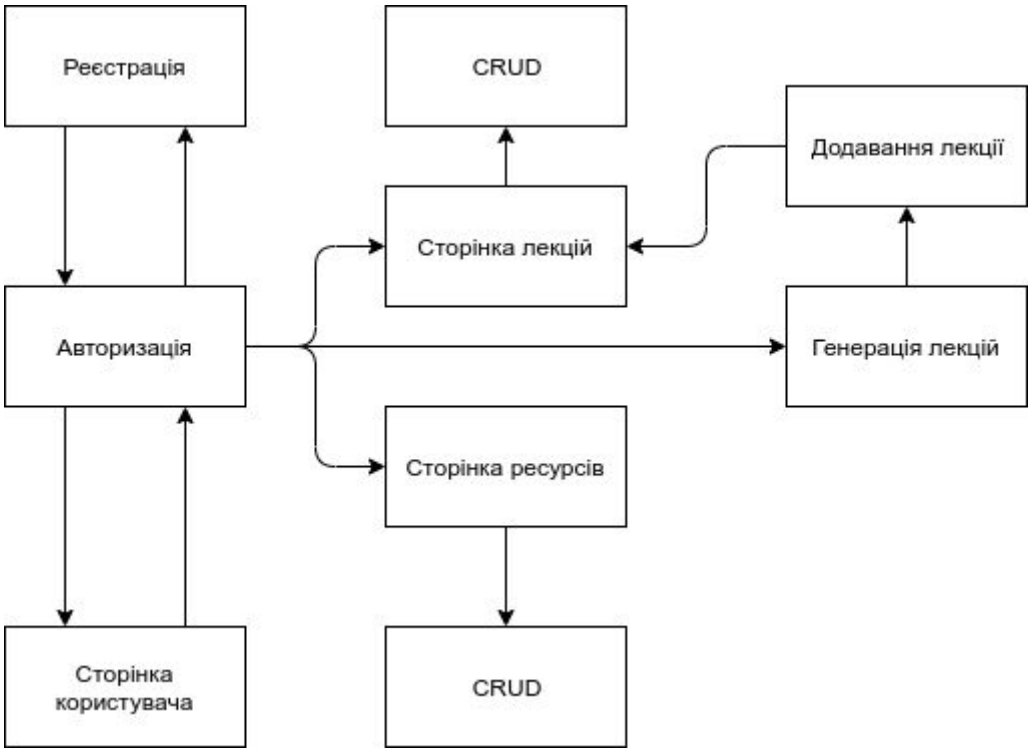
мал. 1 use-case diagram

Component diagram



мал. 2 Component diagram

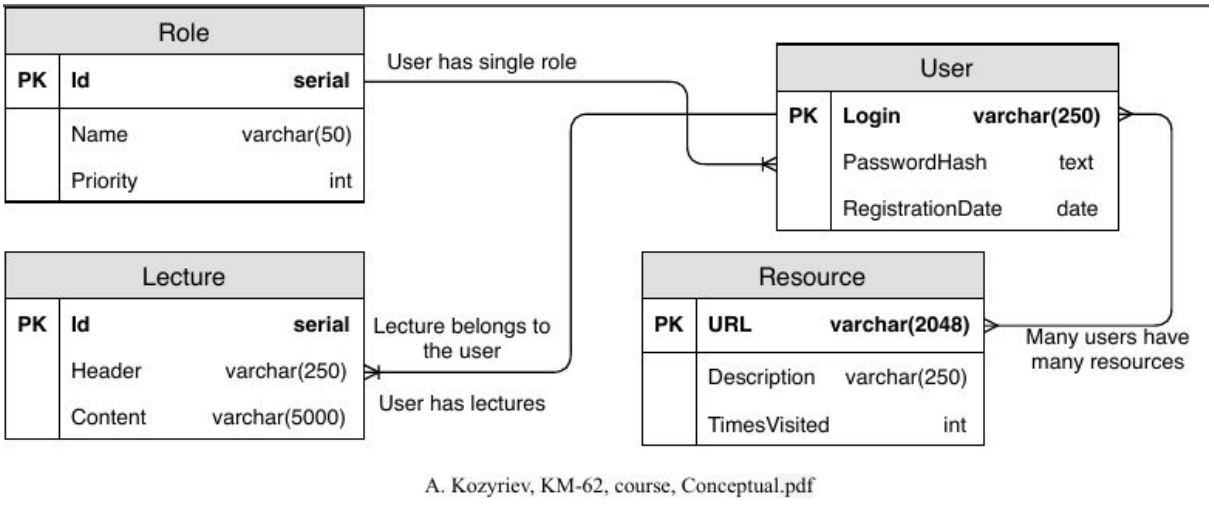
UML



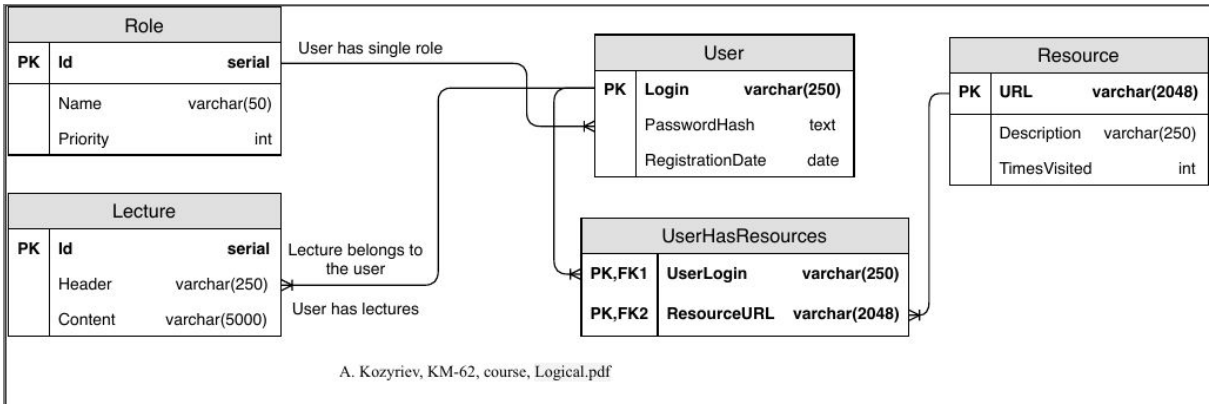
мал. 3 UML-diagram

ІНФОЛОГІЧНЕ ПРОЕКТУВАННЯ

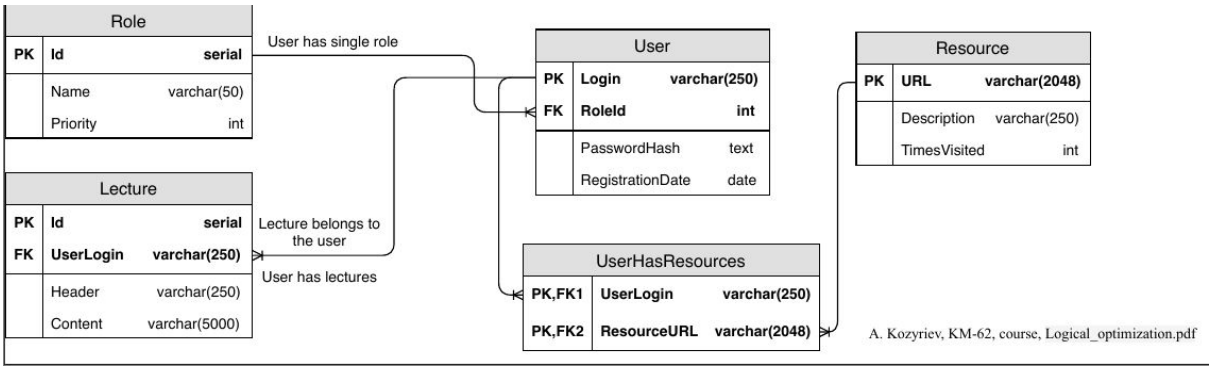
Моделі діючих сутностей



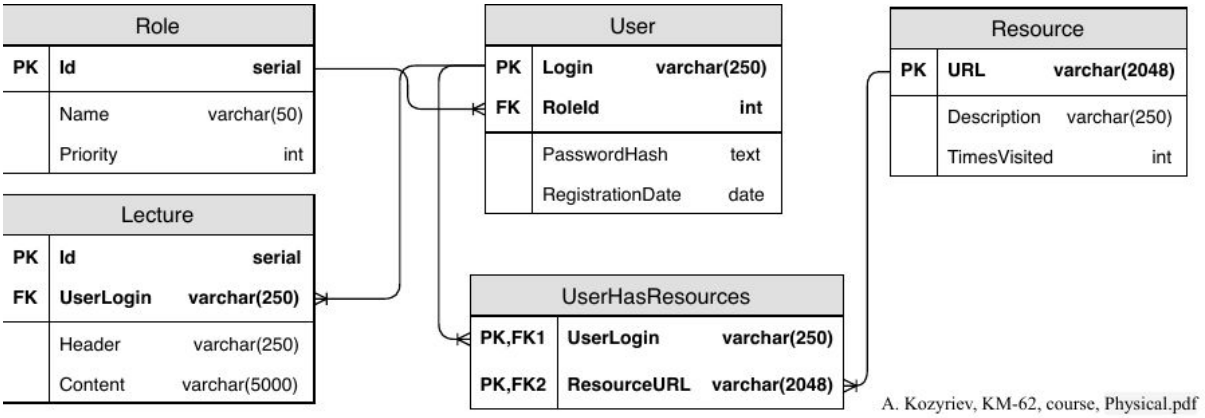
мал. 4 Conceptual ERD



мал. 5 Logical ERD



мал. 6 Logical-Optimized ERD



мал. 7 Physical ERD

ВИСНОВКИ

У ході даної роботи було стровнено додаток мікро сервісної архітектури, що генерує лекції за заданою темою. Додаток та його складові розміщено на серверах Heroku. Архітектура даного додатку є мікросервісною та архітектура виконанна по REST API, що дає можливість масштабувати даний додаток без проблем. Було впроваджено механизм DI, що дало наступні переваги: тестування компонентів окремо, усі компоненти є слабо зв'язаними.

Стек використаних технологій:

1. Flask, SQL-alchemy, WTForms
2. Tensorflow 2.0
3. JQuery, Bootstrap
4. PostgreSQL 12

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Neural Network Using Python and Numpy. Bernd Klein, Bodenseo; Design by Denise Mitchinson adapted for python-course.eu by Bernd Klein, 2011-2019, URL:https://www.python-course.eu/neural_networks_with_python_numpy.php
2. WTForms Documentation, WTForms Team, documentation generated by Sphinx, 2010, URL: https://wtforms.readthedocs.io/en/stable/
3. Heroku Documentation, HEROKU Softwere IS A COMPANY 2019, URL:https://devcenter.heroku.com/categories/reference