

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
ІМЕНІ ІГОРЯ СІКОРСЬКОГО”
КАФЕДРА ПРИКЛАДНОЇ МАТЕМАТИКИ

КУРСОВА РОБОТА
з навчальної дисципліни “Бази даних та інформаційні системи-2”
на тему
СЕРВІС ВИБОРУ РЕЦЕПТІВ НА ОСНОВІ ХАРАКТЕРИСТИК СТРАВ

Виконала студентка групи КМ-61
Серватманд Марьям

Керівник
ТЕРЕЩЕНКО І.О.

Оцінка:
Кількість балів:

КИЇВ – 2020

ЗАВДАННЯ НА ВИКОНАННЯ КУРСОВОЇ РОБОТИ

Створення веб-сервісу для зручного вибору страв з детальним описом рецепту, типом, інгредієнтами що до неї входять. Вибір страв відбувається на основі обраного фільтру за типом страви та на основі введених користувачем характеристик його біологічних ознаків (вік, зріст, вага) таким чином, щоб калорійність страви максимально наближалася до норми, що була обчислено сервісом.

АНОТАЦІЯ

Дана курсова робота містить у собі детальний опис створення веб-сервісу для пошуку страв та їх рецептів, для найбільш оптимального формування проекту було розглянуто аналіз підприємства автоматизації, основні сутності, що беруть участь у формуванні інформаційної системи, конкретну постановку задачі, моделювання бізнес-процесів, даталогічне та інфологічне дослідження, формування проекту виконувалось на мові Python, застосовуючи мікро-фреймворк Flask, уся необхідна інформація буде зберігається в базі даних PostgreSQL.

РЕФЕРАТ

Для реалізації веб-сервісу першочергово було проаналізовано підприємства автоматизації в рамках якого:

- A. створено ієрархію процесів, що ілюструє весь функціонал сервісу
- B. детально описані процеси
- C. було змодельовано use-case для наявних процесів.
- D. описано use-case сценарії.

Також було визначено постановку задачі, для цього було:

- A. визначено 4 основні сутності, для яких було створено окремі таблиці в базі даних.
- B. створено бізнес-правила, які більш детально обмежують можливості користувача

Після чого було виконано інфологічне проектування для якого:

- A. описано кожен сутність та операцію, їх атрибути та пов'язані сутності.
- B. створено концептуальну модель даних

та даталогічне для якої було створено фізичну модель даних.

Стек технологій:

Мова програмування - Python, мікро-фреймворк для веб-додатків - Flask, який є максимально підходящим для побудови простих сайтів зі статичним контентом. Він забезпечує весь необхідний функціонал і дозволяє кастомізацію в великих обсягах. Для спрощеного взаємозв'язку між базою даних та сервером використовується SQLAlchemy, вона реалізує технологію програмування ORM (Object-Relational Mapping), яка пов'язує бази даних з концепціями об'єктно-орієнтованих мов програмування. Базою даних була обрана PostgreSQL, однією з сильних сторін PostgreSQL є її архітектура. Як і багато комерційних СУБД, PostgreSQL може застосовуватися в середовищі клієнт-сервер, що дає масу

переваг як користувачам, так і розробникам, для стилізування проекту було обрано популярний css-фреймворк Bootstrap.

ЗМІСТ

ВСТУП.....	8
1 АНАЛІЗ ПІДПРИЄМСТВА АВТОМАТИЗАЦІЇ.....	9
1.1 Ієрархія процесів.....	9
1.2 Опис бізнес-процесів	9
2 ПОСТАНОВКА ЗАДАЧІ процесів.....	12
2.1 Категорії користувачів.....	12
2.2 Класи даних.....	12
2.3 Бізнес-правила.....	13
3 МОДЕЛЮВАННЯ БІЗНЕС-ПРОЦЕСІВ.....	14
3.1 Створення Use-Cases.....	14
3.2 Use-Cases сценарії.....	14
4 ІНФОЛОГІЧНЕ ПРОЕКТУВАННЯ.....	18
1.2 Опис сутностей.....	18
1.2 Концептуальна діаграма.....	21
5 ДАТАЛОГІЧНЕ ПРОЕКТУВАННЯ.....	22
ВИСНОВКИ.....	23
СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ.....	24

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

- CRUD — (англ. create read update delete) 4 базові функції управління даними «створення, зчитування, зміна і видалення».
- ІС - інформаційна система
- ШІ - штучний інтелект
- БД - база даних

ВСТУП

Їжа є основною потребою людства, але від первісних часів, коли люди харчувалися примітивної їжею пройшло багато часу, сьогодні кулінарія перетворилося на ціле мистецтво. Різноманітність і кількість страв може заплутати будь-яку людину, а оскільки харчування є невід'ємною частиною в житті кожного, раціональний вибір страв є ключем до здорового і благополучного життя, саме тому існує безліч різноманітних сервісів, які допомагають полегшити вибір страви, кожен з них має в собі єдину ідею - дати людині можливість підібрати рецепти відповідно до їх переваг або критеріям, в той же час кожен сервіс є унікальним оскільки розробники самостійно продумують інтерфейс, особливості, пріоритети за якими користувач може вибирати страви і так далі.

Ця тема є актуальною у сучасному світі та є цікавою для мене, оскільки дану тематику можна розширити постійно покращуючи існуючі, та додаючи нові можливості, тому в цій роботі буде описана спроба реалізувати основний функціонал подібного веб-сервісу з додаванням особливої властивості - пошук страв по параметрам людини за допомогою нейронних мереж.

1 АНАЛІЗ ПІДПРИЄМСТВА АВТОМАТИЗАЦІЇ

Дана ієрархія призначена для візуального розгляду усіх існуючих процесів, що реалізовані у даному сервісі.

Основними модулями даного сервісу є - авторизація, CRUD з сутністю “Страва”, підбір страв за параметрами.

1.1 Ієрархія процесів

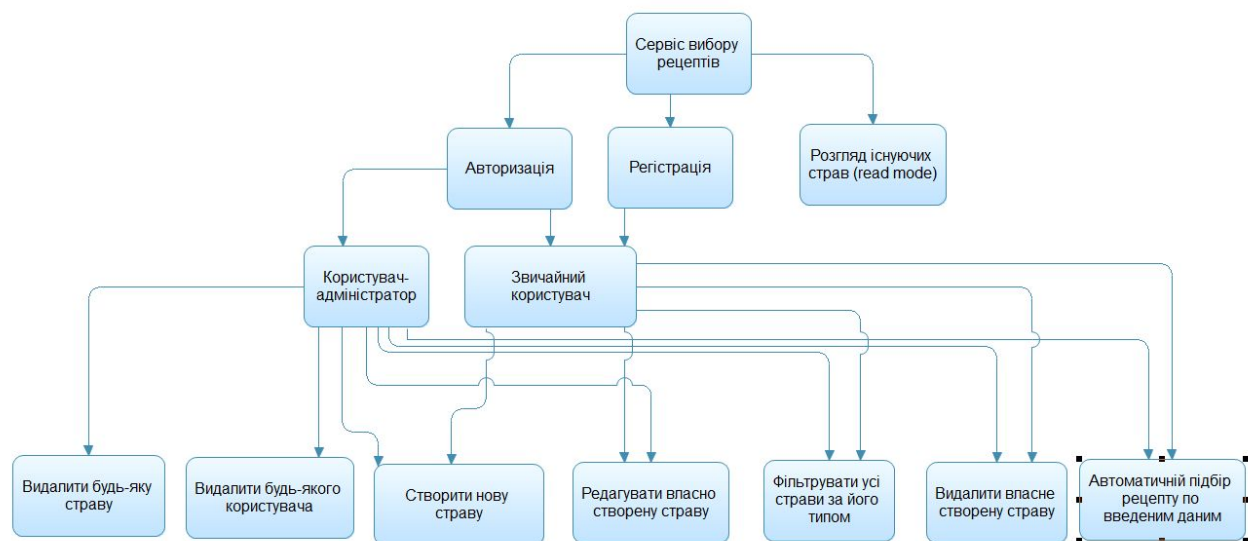


Рисунок 1.1 – Ієрархія процесів

1.2 Опис бізнес-процесів

Таблиця 1.1 - Опис процесу “Реєстрація”

Назва процесу	Реєстрація
Сутності	User
Вхідні дані	email, login, password
Опис функціоналу	Дані вводяться до відповідної форми та записуються до БД, у випадку, якщо логін та емейл унікальні.

Таблиця 1.1 - Опис процесу “Авторизація”

Назва процесу	Авторизація
Сутності	User
Вхідні дані	login, password
Опис функціоналу	Дані вводяться до відповідної форми та перевіряються на наявність у сутності User у БД, у випадку, якщо логін та емейл існують відбувається авторизація.

Таблиця 1.1 - Опис процесу “Додавання страви”

Назва процесу	Додавання страви
Сутності	User, Dish, Type, Ingridients
Вхідні дані	dishname, type, ingridients, receipt_content
Опис функціоналу	Дані вводяться до відповідної форми, інгредієнти та тип виводяться із відповідних сутностей з БД та записуються до сутності Dish.

Таблиця 1.1 - Опис процесу “Видалення страви”

Назва процесу	Видалення страви
Сутності	User, Dish
Вхідні дані	id страви, що передається у url при натиску кнопки “видалити”
Опис функціоналу	Дані приймає сервер та ідентифікує страву та видаляє її із БД.

Таблиця 1.1 - Опис процесу “Редагування страви”

Назва процесу	Редагування страви
Сутності	User, Dish
Вхідні дані	id страви, що передається у url при натиску кнопки “редагувати”
Опис функціоналу	Сервер отримує id страви, ідентифікує її, відкриває форму для редагування, отримує нові дані та перезаписує їх у БД

Таблиця 1.1 - Опис процесу “Видалення користувачів”

Назва процесу	Видалення користувачів
Сутності	User
Вхідні дані	Отримання id користувача при натиску на кнопку “Видалити”, що передається у url.
Опис функціоналу	Дані приймає сервер та ідентифікує користувача та видаляє його із БД.

Таблиця 1.1 - Опис процесу “Автоматичний підбір рецепту”

Назва процесу	Автоматичний підбір рецепту
Сутності	User
Вхідні дані	Параметри: вік, вага, зріст, що вводяться до форми
Опис функціоналу	Дані приймає сервер та робить розрахунки завдяки яким можна знайти правильно підібрані страви.

2 ПОСТАНОВКА ЗАДАЧІ

Головною метою створення даної ІС є реалізація веб-сервісу для пошуку страв з їх рецептами. Основними вимогами є можливість виконання операції CRUD над стравами, можливість реєстрації та авторизації, модерування адміністратором, фільтрування за типом страв, підбір страв за допомоги нейронної мережі та кореляційного аналізу.

Результат роботи сервісу є вивід страв, калорійність яких підібрана за допомогою задачі ІІІ, а також вивід страв, що є профільтованими за їх типом.

2.1 Підрозділ категорії користувачів

Існує два типи користувачів у даній ІС:

1. Звичайний користувач (надалі: користувач) та адміністратор, користувач має можливість виконувати операції CRUD над стравами, фільтрувати за типом, знаходити індивідуальні страви.
2. Адміністратор має ті самі функції, але додатково має можливість видаляти користувачів із ІС та видаляти страви будь-які страви.

2.2 Підрозділ класи даних

На початку, якщо користувач не зареєстрований вхідними даними будуть його дані при реєстрації (логін, емейл, пароль) які будуть автоматично записуватись до бази даних (у випадку, якщо такий логін і емейл вже не наявні у БД), при авторизації введені дані (логін та пароль) будуть перевірятись на наявність у БД. Для створення страви вхідними даними буде інформація про страву, що була введена користувачем у відповідні поля форми. Для знаходження індивідуальних

страв вхідними даними будуть параметри користувача, які він відповідно введе у необхідну форму.

2.3 Підрозділ бізнес-правила

Бізнес правила:

1. Тільки авторизовані користувачі мають доступ до функціоналу сервісу.
2. Неможливо створити більше одного адміністратора.
3. Тільки у адміністратора є окремі права на видалення користувачів та їх страві.
4. Логін та емейл мають бути унікальними.
5. Користувач має можливість видалити та редагувати лише власно створені страви.
6. Підбір рецептів створених за допомогою нейронної мережі не забезпечує 100% точність.
7. Адміністратор має право модерувати (видаляти) будь-яку страву на свій розсуд.
8. Лише адміністратор може додавати нові типи та інгредієнти для страв.
9. Користувач не може додавати страви анонімно.
10. Якщо користувач не виявляв активність упродовж 1-го року, адміністратор має право заблокувати даного користувача.

3 МОДЕЛЮВАННЯ БІЗНЕС-ПРОЦЕСІВ

Змодельований Use-Case для процесів, що можна виконувати наразі у веб-застосунку, зображено на рисунку 3.1.

3.1 Use-case діаграма

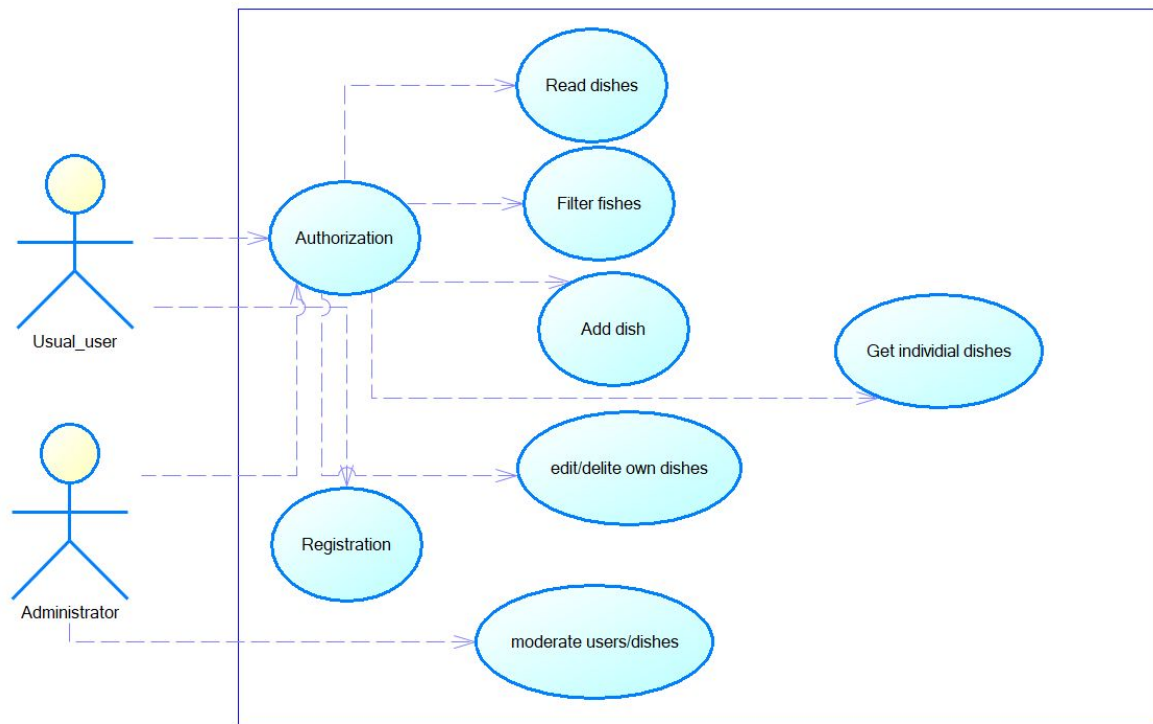


Рисунок 3.1 – Use-case diagram для веб-застосунку

3.2 Use-case сценарії

Таблиця 3.1 - сценарій “Авторизація”

Назва	Авторизація
Виконавці	Користувач, сервіс
Мета	Ідентифікувати користувача в системі для надання йому доступу до функціоналу сервісу
Передумова	Користувач відкрив сервіс, не

	авторизувався
Успішний сценарій	Користувач вводить дані (логін, пароль) до відповідної форми, база даних знаходить їх, та записує дані користувача як “поточного”
Результат	Користувач має доступ до функціоналу сервісу під власним логіном

Таблиця 3.2 - сценарій “Реєстрація”

Назва	Реєстрація
Виконавці	Користувач, сервіс
Мета	Записати нові дані користувача в БД системі для надання йому доступу до функціоналу сервісу
Передумова	Користувач відкрив сервіс, не авторизувався і не має даних для авторизації
Успішний сценарій	Користувач вводить дані (логін, емейл, пароль) до відповідної форми, база даних додає новий запис, та записує дані користувача як “поточного”
Результат	Користувач автоматично авторизується з новими даними та має доступ до функціоналу сервісу під власним логіном

Таблиця 3.3 - сценарій “Створення нової страви”

Назва	Створення нової страви
Виконавці	Користувач, система
Мета	Додати нову страву для відображення на сервісі

Передумова	Користувач має бажання створити нову власну страву
Успішний сценарій	Користувач вводить дані про нову страву у відповідні форми, при збереженні БД додає новий запис у таблицю “Страв”
Результат	Будь-який користувач має змогу бачити нову страву

Таблиця 3.4 - сценарій “Видалення страви”

Назва	Видалення страви
Виконавці	Користувач, система
Мета	Видалити власну страву на сервісі
Передумова	Користувач має видалити створену ним страву
Успішний сценарій	Запис про дану страву видаляється з БД та сервісу
Результат	Видалена страву більше ніде не відображається

Таблиця 3.5 - сценарій “Редагування страви”

Назва	Редагування страви
Виконавці	Користувач, система
Мета	Змінити дані про створені страви для відображення на сервісі актуальної інформації
Передумова	Користувач має редагувати створену ним страву
Успішний сценарій	Користувач вводить нові дані про страву у відповідні форми, при збереженні БД змінює запис у таблицю “Страв”

Результат	Будь-який користувач має змогу бачити нові дані про страву
-----------	--

Таблиця 3.6 - сценарій “Пошук індивідуальної страви”

Назва	Пошук індивідуальної страви
Виконавці	Користувач, система
Мета	Знайти найбільш задовільні страви
Передумова	Користувач перейшов до відповідної вкладки сервісу
Успішний сценарій	Користувач ввів власні параметри (вік, вага, зріст) у відповідну форму, натиснув “ок” та отримав страви, що були обчислені сервером
Результат	Сервіс відображає лише відповідні страви до введених користувачем параметрам

Таблиця 3.7 - сценарій “Видалення користувачів”

Назва	Видалення користувачів
Виконавці	Адміністратор, система
Мета	Модерація користувачів системи
Передумова	Перехід до вкладки “усі користувачі”
Успішний сценарій	Адміністратор знаходить користувача, якого хоче видалити з системи, натискає на кнопку “видалити”, користувач однозначно ідентифікується та видаляється запис з його даними з БД
Результат	Даного користувача більше нема в системі, він не може авторизуватися з такими даними.

4 ІНФОЛОГІЧНЕ ПРОЕКТУВАННЯ

4.1 ОПИС СУТНОСТЕЙ

Для даного інформаційного сервісу було визначено 5 сутностей, що повністю покривають усі можливі взаємодії системи.

1. Користувач - User
2. Страва - Dish
3. Інгредієнт - Ingridients
4. Тип - Type
5. Рецепт - Receipt

Таблиця 4.1 - сутність “User”

Назва	User
Призначення сутності	Зберігає інформацію, яка необхідна для того, щоб однозначно визначити користувачів ІС,
Атрибути:	Призначення атрибуту
login	Логін користувача, який може бути довільним, який однозначно ідентифікує користувача
email	Додаткова інформація для користувача, має бути унікальним.
password	Додаткова інформація для користувача, яка допомагає ідентифікувати користувача з логіном, може бути довільним, не унікальним.

Таблиця 4.2 - сутність “Dish”

Назва	Dish
Призначення сутності	Містить всю інформацію про страву, яка виводиться у сервісі.
Атрибути:	Призначення атрибуту
dishname	Назва страви
calories_amount	Кількість калорій, що містить страва
author	Логін користувача, що додав страву
type	Тип страви
receipt	Текстовий контент, що містить детальний опис про приготування страви
ingridients	Інгредієнти, що належать до рецепту страви

Таблиця 4.3 - сутність “Ingridient”

Назва	Ingridient
Призначення сутності	Для збереження інгредієнтів
Атрибути:	Призначення атрибуту
ingridient	Містить назву інгредієнта

Таблиця 4.4 - сутність “Type”

Назва	Type
Призначення сутності	Для збереження всіх можливих типів страв
Атрибути:	Призначення атрибуту
typename	Назва типу страви

Таблиця 4.5 - сутність “Receipt”

Назва	Receipt
Призначення сутності	Містить рецепт
Атрибути:	Призначення атрибуту
receipt_content	Містить текстовий контент рецепту

Таблиця 4.6 - сутність “Dish_type”

Назва	Dish_type
Призначення сутності	Містить інформацію про те, який тип має страва
Атрибути:	Призначення атрибуту
dishname	Назва страви
type	Назва типу, що має страва

Таблиця 4.7 - сутність “Dish_ingredients”

Назва	Dish_ingredients
Призначення сутності	Містить інформацію про те, які інгредієнти має страва
Атрибути:	Призначення атрибуту
dishname	Назва страви
ingredients	Масив інгредієнтів, що містить страва

Таблиця 4.8 - сутність “User_dish”

Назва	User_dish
Призначення сутності	Інформація про те, які страви має користувач
Атрибути:	Призначення атрибуту

login	Логін користувача, що додав страву
dishname	Назва страви, що була створена користувачем з відповідним логіном

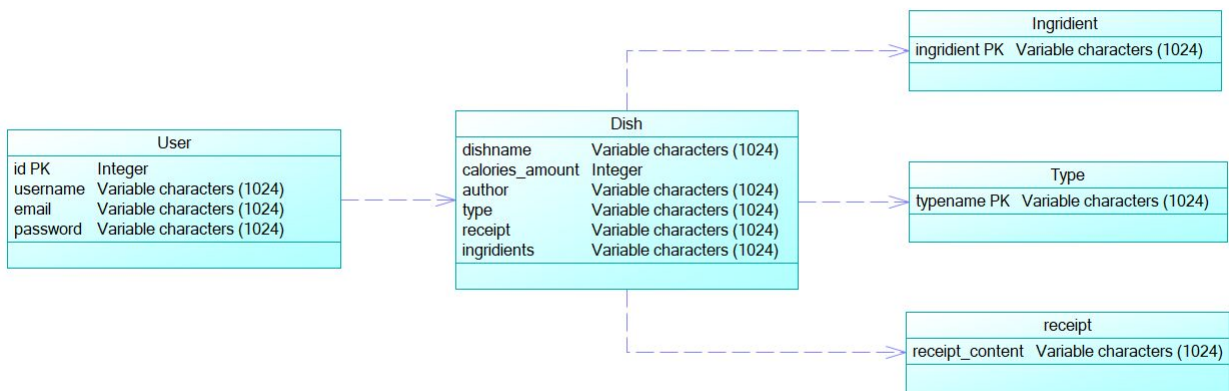


Рисунок 4.1 – Концептуальна діаграма бази даних

5 ДАТАЛОГІЧНЕ ПРОЕКТУВАННЯ

Для даного етапу було створено фізичну діаграму, що ілюструє взаємодію сутностей один з одним

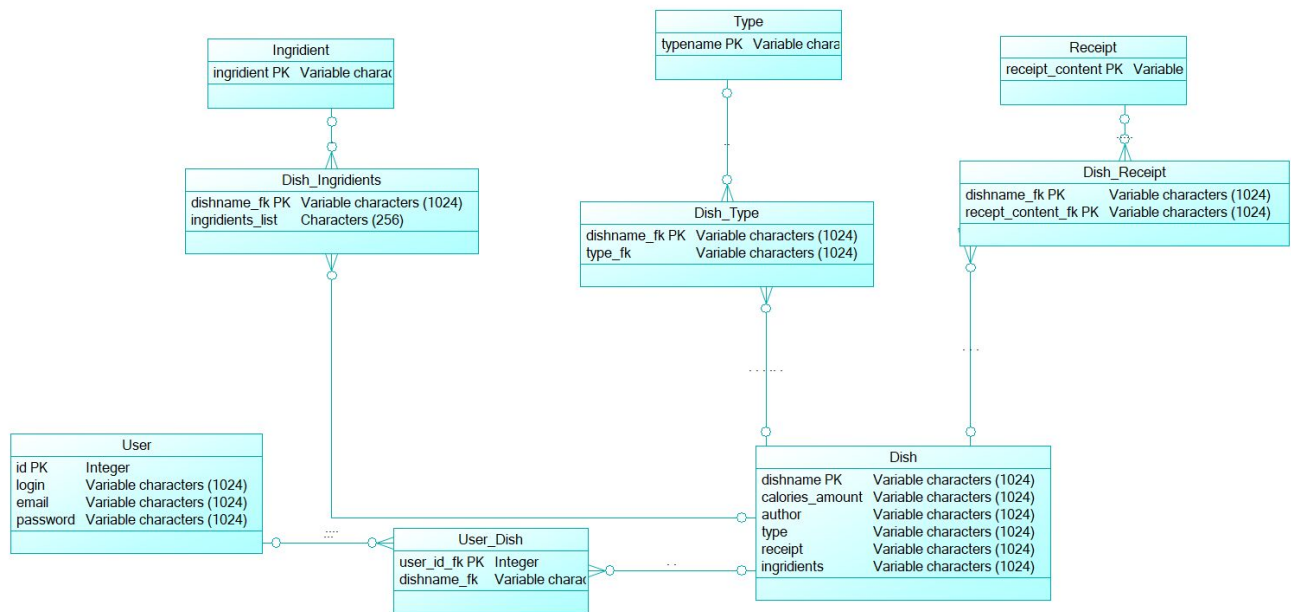


Рисунок 5.1 – Фізична діаграма бази даних

ВИСНОВКИ

У даній курсовій роботі було розроблено інформаційну систему, метою якої було створення спрощеного сервісу для пошуку страв під вимоги користувачів. Було визначено основні способи за якими можна класифікувати страви (а саме: за типом), також було реалізовано індивідуальний підбір страв під параметри, які користувач власноруч вводить в сервіс.

Для більш оптимізованої діяльності перш ніж приступити до безпосередньої реалізації веб-застосунку необхідно було виконати декілька підготовчих етапів: визначити основні сутності, які активно використовуються у даній системі, виокремити усі процеси, що становлять собою повноцінний функціонал сервісу, створити усі наглядні діаграми та use-case сценарії для того, щоб візуально створити уявлення про майбутній проект.

Також, можна виокремити факт, що для створення подібного сервісу дуже зручним виявився мікро-фреймворк Flask, завдяки якому дуже швидко можна генерувати HTML-сторінки, особливо при використанні SQLAlchemy, яка дозволяє отримати доступ до БД та описувати структури таблиць за допомогою мови Python.

У результаті було створено прототип веб-сервісу для пошуку страв, який містить у собі базові функції, хоча він ще не є максимально зручним у користуванні, оскільки користувацький інтерфейс є на доволі примітивному рівні, функціонал не є обширний та не задовольняє всі потреби користувачів, проте даний сервіс задовольняє мету, яка була описана у постановці задачі даної курсової та може стати підґрунтям для майбутнього повноцінного веб-застосунку, для цього необхідно продовжити процес аналізу потреб користувачів на аналогічних сервісах та впровадити їх до вже готової системи.

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Електроний ресурс - CRUD

URL: <https://uk.wikipedia.org/wiki/CRUD>

2. Електроний ресурс - Flask

URL: <https://ru.wikipedia.org/wiki/Flask>

3. Електроний ресурс - SQLAlchemy

URL: <https://ru.wikibooks.org/wiki/SQLAlchemy>

4. Електроний ресурс - Flask documentation

URL: <https://flask-russian-docs.readthedocs.io/ru/latest/quickstart.html>

5. Електроний ресурс - Posgres documentation

URL: <https://postgrespro.ru/docs/postgresql/9.6/index>

6. Електроний ресурс - WTForms documentation

URL: <https://wtforms.readthedocs.io/en/stable/>