

Міністерство освіти і науки України
Харківський радіотехнічний фаховий коледж
Циклова комісія «Комп'ютерних наук»

КУРСОВА РОБОТА
з навчальної дисципліни «БАЗИ ДАНИХ»
на тему «Ресторан. Кухня.»

Виконав:
студент гр. ПІ-312
Пономарьов Павло Ігорович
Керівник роботи:
Ахмедзянова О.А.

м. Харків – 2025 рік

Харківський радіотехнічний фаховий коледж

Циклова комісія КОМП'ЮТЕРНИХ НАУК
Спеціальність 121 «Інженерія програмного забезпечення»
Курс - 3 Група – ПІ - 312 Семестр - 6

Завдання на курсову роботу

студента Пономарьова Павла Ігоровича

Тема курсової роботи: «Ресторан. Кухня.»

ОСНОВНІ ДАНІ: У Базі даних повинна зберігатися інформація про страви, рецепти (технологія приготування страви), вихід (вага порції), продукти, що входять до складу страви, рівень (розряд) кухарів та їх перелік страв за рівнем. Запити:

1. Пошук страв за категоріями, складом, назвою.
2. Підрахунок кількості витрачених та залишилися продуктах.
3. Кількість поставлених продуктів.
4. Рівень (розряд) кухарів та перелік доступним їм страв.

Склад пояснювальної записки:

Вступ

1 Аналіз предметної області і постановка задачі

2 Проєктування реляційної БД

2.1 Побудова ER-діаграми

2.2 Побудова схеми реляційної моделі даних

2.3 Вибір СУБД та опис фізичної моделі даних

2.4 Ескіз інтерфейсу користувача

3 Опис програмної реалізації

3.1 Опис структури бази даних

3.2 Опис задач автоматизації та інтерфейсу користувача

Висновки

Перелік використаних джерел

Дата видачі завдання: 17 січня 2025р

Дата здачі виконаної роботи: _____

Керівник роботи _____ /О.А.Ахмедзянова /

Розглянуто на засіданні ЦК

Голова ЦК _____ /Федосєєва А.О./

ЗАЛІКОВИЙ ЛИСТ

Харківський радіотехнічний фаховий коледж

Циклова комісія КОМП'ЮТЕРНИХ НАУК
Спеціальність 121 «Інженерія програмного забезпечення»
Курс - 3 Група – ПІ - 312 Семестр - 6

Курсова робота

студента Пономарьова Павла Ігоровича
на тему «Ресторан. Кухня»

ДОПУЩЕНИЙ ДО ЗАХИСТУ

Керівник роботи _____ /Ахмедзянова О.А. /

ОЦІНКА _____

Члени комісії: _____ /Ахмедзянова О.А./
 _____ /Радченко О.П. /
 _____ /Пироженко В.В./

« ____ » « _____ » 2025 р

ЗМІСТ

ВСТУП	5
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ.....	6
2 ПРОЄКТУВАННЯ БАЗИ ДАНИХ.....	9
2.1 Побудова діаграм	9
2.2 Побудова реляційної моделі даних	10
2.3 Вибір СУБД та опис фізичної моделі даних	11
3 ОПИС ПРОГРАММНОЇ РЕАЛІЗАЦІЇ	14
3.1 Опис структури бази даних.....	14
3.2 Опис задач автоматизації та інтерфейсу користувача	16
ВИСНОВКИ.....	17
ПЕРЕЛІК ПОСИЛАНЬ.....	18

ВСТУП

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ І ПОСТАНОВКА ЗАДАЧІ

Предметна область даної курсової – Кухня ресторану.

Необхідно реалізувати зберігання інформації про різні страви, інгредієнти з яких вони виготовлені, рецепти за якими вони зроблені, рівень кухаря для виготовлення цієї страви, та інформацію про персонал який працює на кухні.

Кожен працівник кухні має мати доступ до важливих йому даних (бізнес ролі):

- Шеф-кухар – до усіх рецептів, їх цін та їх редагування, повного списку усіх кухарів, інформації про них та її редагування, та до списку запасів на складі для створення замовлень у постачальників та його редагування.
- Шеф-де-парті, шеф-де-кузін, соус-шеф-де-кузін – мають мати доступ до рецептів які доступні їм по їх особливій спеціалізації, коли вони готують страву – кількість залишків, прийняття замовлень та зміни кількості залишившихся продуктів на складі якщо був брак або після виконання замовлення.
- Нижчі рівні працюють по вказівкам шефа-де-парті, тому їм доступ до бази даних не потрібен, коли досягнуть більшого рівню – отримають доступ.
- Офіціант – має доступ до додавання замовлення на кухню, тобто додавання блюд, які треба приготувати.

Інформація, яку буде зберігати база даних:

- Співробітник:
 - Ідентифікатор співробітника – унікальне число, обов'язкове;
 - Призвище Ім'я по Батькові – текст, обов'язкове;
 - Роль на кухні – текст (шеф-кухар, шеф-де-парті, шеф-де-кузін, соус-шеф-де-кузін, офіціант), обов'язкове;
- Замовлення (на кухні кожне блюдо вважається замовленням):
 - Ідентифікатор замовлення – унікальне число, обов'язкове;
 - Ідентифікатор рецепту – число, обов'язкове;
 - Кількість порцій – число, обов'язкове;

- Дата замовлення – дата, обов’язкове;
- Рецепт:
 - Ідентифікатор рецепту – унікальне число, обов’язкове;
 - Назва – унікальний текст, обов’язкове;
 - Категорія – текст, обов’язкове;
 - Роль кухаря – текст, обов’язкове;
 - Ціна – число, обов’язкове;
 - Вага (в грамах) – число, обов’язкове;
- Складник:
 - Ідентифікатор складнику – унікальне число, обов’язкове;
 - Ідентифікатор рецепту – число, обов’язкове;
 - Ідентифікатор продукту – число, обов’язкове;
 - Кількість у рецепті у грамах – число, обов’язкове;
 - Тип обробки – текст, обов’язкове;
- Продукт:
 - Ідентифікатор продукту – унікальне число, обов’язкове;
 - Кількість на складі у грамах – число, обов’язкове;
- Дія з продуктом:
 - Ідентифікатор дії – унікальне число, обов’язкове;
 - Ідентифікатор продукту – унікальне число, обов’язкове;
 - Тип дії – (витрачання, додавання), обов’язкове;
 - Кількість витраченого у грамах – число, обов’язкове;
 - Дата здійснення дії – дата, обов’язкове;

Кількість продуктів на складі може змінюватися після закупки, після витрати продукту (брак, спорченність та інша марна трата входить в витрату продуктів). Офіціант може тільки додавати замовлення, але не редагувати чи видаляти їх.

Трата зі складу та замовлення блюд в кухню формується через програму. Купівля відбувається з можливістю експорту чеку накладної з програми.

Функції додатку:

- Формування замовлення продуктів на склад (має доступ тільки шеф-кухар);
- Редагування рецептів та цін на блюда (має доступ тільки шеф-кухар);
- Редагування та перегляд інформації про кухарів (має доступ тільки шеф-кухар);
- Підрахунок кількості витрачених та поставлених продуктів (має доступ тільки шеф кухар);
- Список товарів та їх кількість на складі (мають доступ усі кухарі);
- Трата продуктів зі складу (мають доступ усі кухарі);
- Додавання замовлення блюда на кухню (має доступ тільки офіціант);
- Пошук страв за категоріями, складом, назвою (мають доступ усі, але з обмеженнями для деяких кухарів);
- Розряд кухаря та перелік його страв (має доступ тільки шеф-повар або кухарь про свою інформацію).

2 ПРОЄКТУВАННЯ БАЗИ ДАНИХ

2.1 Побудова діаграм

База даних буде побудована на підставі схеми бази даних (ER-діаграми), де вказані усі сутності та зв'язки (Рисунок 2.1.1).

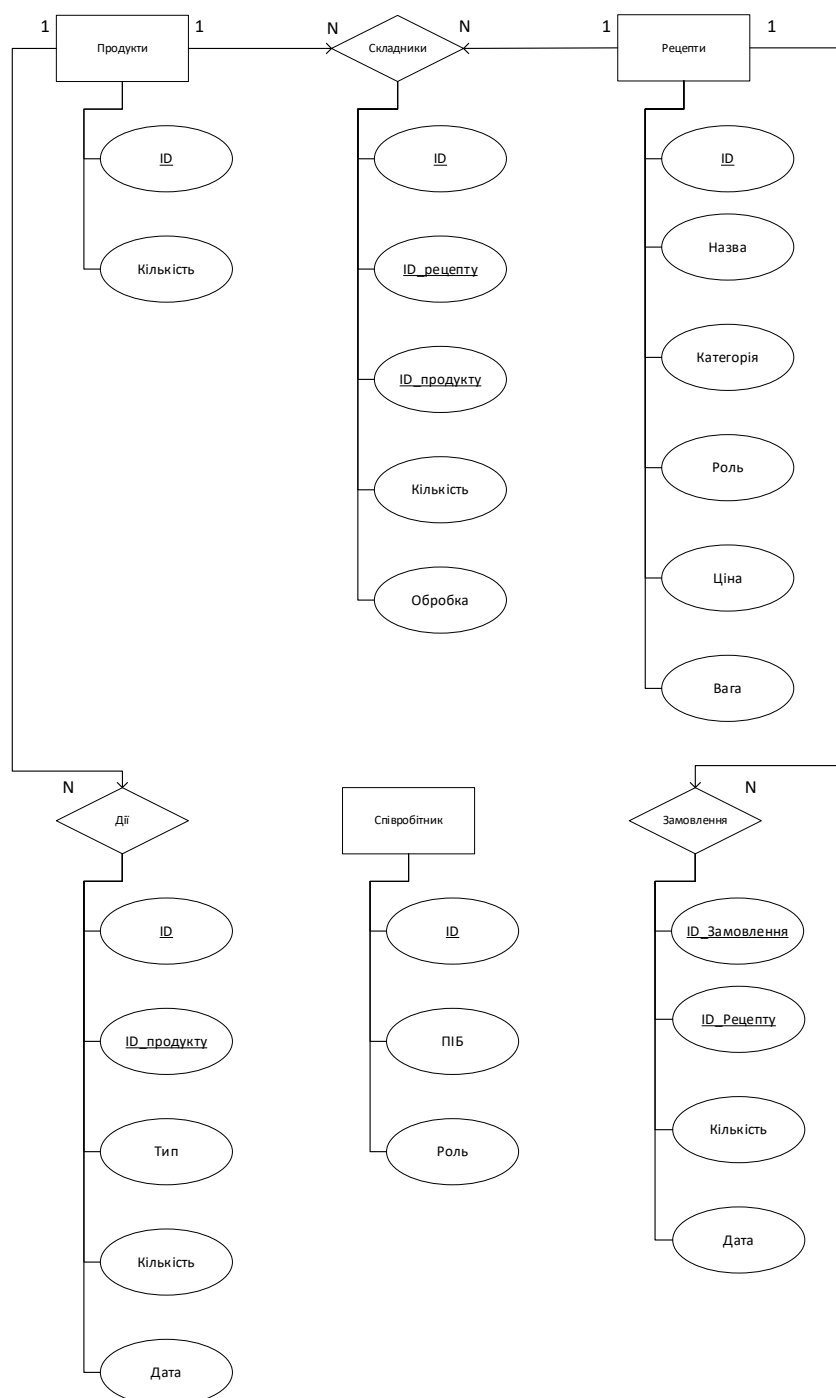


Рисунок 2.1.1 – ER-діаграма

Також зобразимо сценарії, де кожен робітник (користувач системи) може використовувати функції, тобто Use Case (Рисунок 2.1.2).

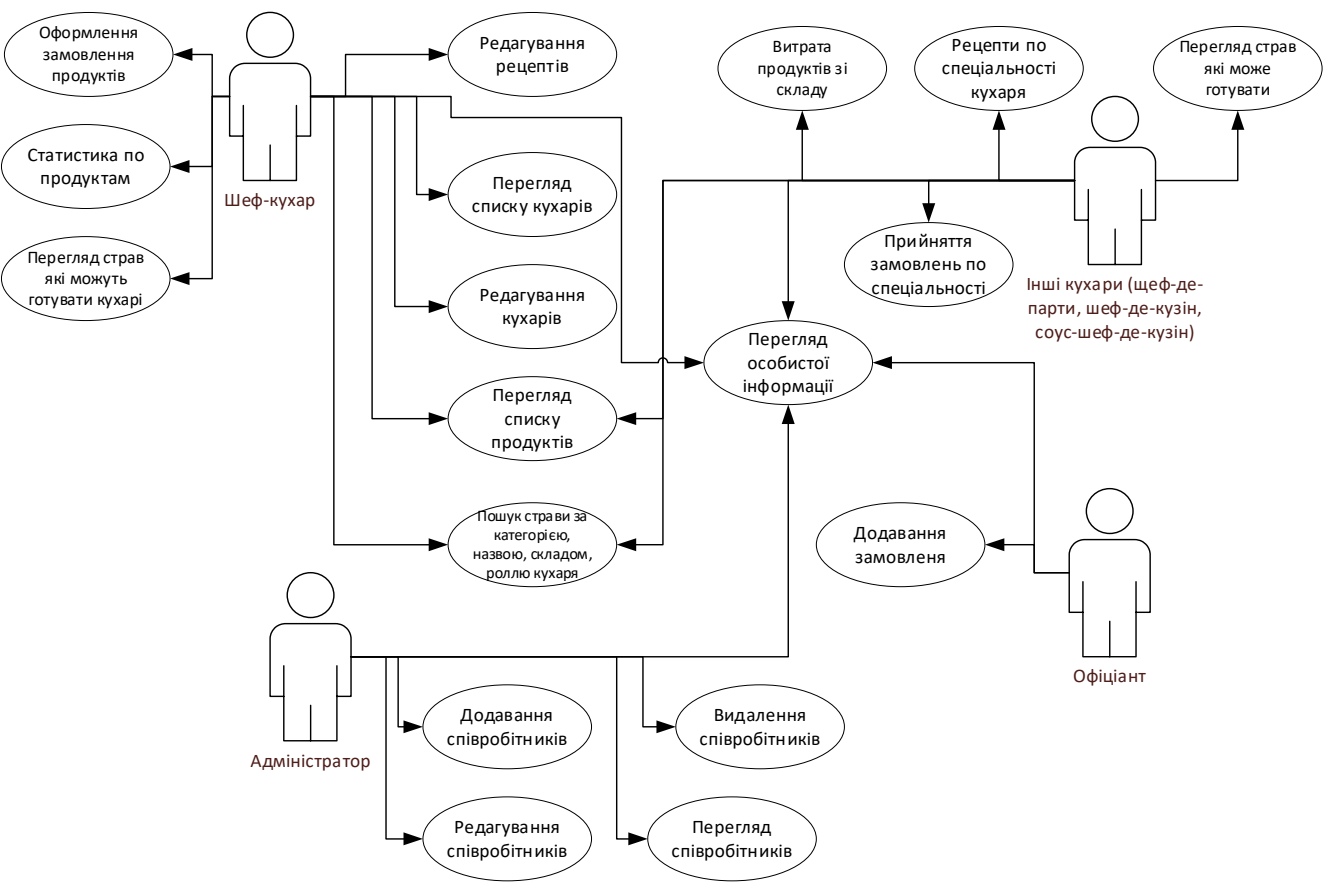


Рисунок 2.1.2 – Use Case

2.2 Побудова реляційної моделі даних

Використовуючи ER-діаграму з попереднього розділу побудуємо реляційні відносини бази даних.

Схеми відношень відображені в таблицях 2.2.1 – 2.2.6.

Таблиця 2.2.1 – Відношення «Співробітник»

Ідентифікатор	Прізвище	Роль на кухні
---------------	----------	---------------

Таблиця 2.2.2 – Відношення «Замовлення»

Ідентифікатор замовлення	Ідентифікатор рецепту	Кількість порцій	Дата створення
-----------------------------	--------------------------	------------------	----------------

Таблиця 2.2.3 – Відношення «Рецепт»

Ідентифікатор	Назва	Категорія	Роль кухаря	Ціна	Вага в грамах
---------------	-------	-----------	----------------	------	------------------

Таблиця 2.2.4 – Відношення «Складник»

Ідентифікатор складнику	Ідентифікатор рецепту	Ідентифікатор продукту	Кількість у грамах	Тип обробки
----------------------------	--------------------------	---------------------------	-----------------------	-------------

Таблиця 2.2.5 – Відношення «Продукт»

Ідентифікатор	Кількість на складі	Тип виміру (штуки, грами, міліграми та т.п.)
---------------	---------------------	---

Таблиця 2.2.6 – Відношення «Дія з продуктом»

Ідентифікатор дії	Ідентифікатор продукту	Тип дії	Кількість витраченого	Дата здійснення дії
----------------------	---------------------------	---------	--------------------------	------------------------

2.3 Вибір СУБД та опис фізичної моделі даних

Буде використана СУБД SQLite тому, що вона швидка, та гарно підходить для невеликих додатків, яким кухня і є.

Фізична модель бази даних відображена у таблиці 2.3.1.

Таблиця 2.3.1 – Фізична модель бази даних

Назва стовпця	Тип даних	Обмеження	Підпис
Таблиця «Співробітник»			
id	INT	primary key, not null, unique	Ідентифікатор співробітника
fullname	TEXT	not null	Прізвище Ім'я по Батькові
role	TEXT	not null	Роль на кухні
Таблиця «Замовлення»			
id	INT	primary key, not null, unique	Ідентифікатор замовлення
recipe_id	INT	not null	Ідентифікатор рецепту
quantity	INT	not null	Кількість порцій
created_at	TEXT	not null	Дата створення замовлення
Таблиця «Рецепт»			
id	INT	primary key, not null, unique	Ідентифікатор рецепту
title	TEXT	not null, unique	Назва
category	TEXT	not null	Категорія
role	TEXT	not null	Роль кухаря
price	REAL	not null	Ціна
weight	INT	not null	Вага в грамах
Таблиця «Складник»			
id	INT	primary key, not null, unique	Ідентифікатор складнику
recipe_id	INT	not null	Ідентифікатор рецепту

Продовження таблиці 2.3.1

Назва стовпця	Назва стовпця	Назва стовпця	Назва стовпця
product_id	INT	not null	Ідентифікатор продукту
amount	INT	not null	Кількість в рецепті у грамах
processing_type	TEXT	not null	Тип обробки
Таблиця «Продукт»			
id	INT	primary key, not null, unique	Ідентифікатор продукту
amount	INT	not null	Кількість на складі
measurement	TEXT	not null	Тип виміру (штуки, грами, міліграми та т.п.)
Таблиця «Дія з продуктом»			
id	INT	primary key, not null, unique	Ідентифікатор дії
product_id	INT	not null	Ідентифікатор продукту
action_type	INT	not null	Тип дії (додавання або витрата)
amount	INT	not null	Кількість витраченого у грамах
log_at	TEXT	not null	Дата здійснення дії

3 ОПИС ПРОГРАММНОЇ РЕАЛІЗАЦІЇ

3.1 Опис структури бази даних

База даних “kitchen” створена за допомогою Sqlite (тому sql коду для створення бази немає).

Таблиця “worker” зберігає у собі інформацію про усіх робітників кухні, яким потрібен доступ до бази даних. SQL код для створення таблиці:

```
CREATE TABLE IF NOT EXISTS "worker" (  
    id INTEGER PRIMARY KEY,  
    fullname TEXT NOT NULL,  
    role TEXT NOT NULL  
);
```

Таблиця “recipe” зберігає у собі інформацію про рецепти. SQL код для створення таблиці:

```
CREATE TABLE IF NOT EXISTS "recipe" (  
    id INTEGER PRIMARY KEY,  
    title TEXT NOT NULL UNIQUE,  
    category TEXT NOT NULL,  
    role TEXT NOT NULL,  
    price REAL NOT NULL,  
    weight INTEGER NOT NULL  
);
```

Таблиця “product” зберігає у собі інформацію про продукти на складі. SQL код для створення таблиці:

```
CREATE TABLE IF NOT EXISTS "product" (  
    id INTEGER PRIMARY KEY,  
    amount INTEGER NOT NULL,  
    measurement TEXT NOT NULL  
);
```

Таблиця “ order” зберігає у собі інформацію про замовлення. SQL код для створення таблиці:

```
CREATE TABLE IF NOT EXISTS "order" (  
    id INTEGER PRIMARY KEY,  
    recipe_id INTEGER NOT NULL,  
    quantity INTEGER NOT NULL,  
    created_at TEXT NOT NULL,  
    FOREIGN KEY (recipe_id) REFERENCES recipe(id)  
);
```

Таблиця “ ingredient” зберігає у собі інформацію про інгредієнти. SQL код для створення таблиці:

```
CREATE TABLE IF NOT EXISTS "ingredient" (  
    id INTEGER PRIMARY KEY,  
    recipe_id INTEGER NOT NULL,  
    product_id INTEGER NOT NULL,  
    amount INTEGER NOT NULL,  
    processing_type TEXT NOT NULL,  
    FOREIGN KEY (recipe_id) REFERENCES recipe(id),  
    FOREIGN KEY (product_id) REFERENCES product(id)  
);
```

Таблиця “ warehouse_log” зберігає у собі інформацію про дії на складі. SQL код для створення таблиці:

```
CREATE TABLE IF NOT EXISTS "warehouse_log" (  
    id INTEGER PRIMARY KEY,  
    product_id INTEGER NOT NULL,  
    action_type INTEGER NOT NULL,  
    amount INTEGER NOT NULL,  
    log_at TEXT NOT NULL,  
    FOREIGN KEY (product_id) REFERENCES product(id)  
);
```

На рисунку 3.1.1 зображена схема зв'язків таблиць між собою.

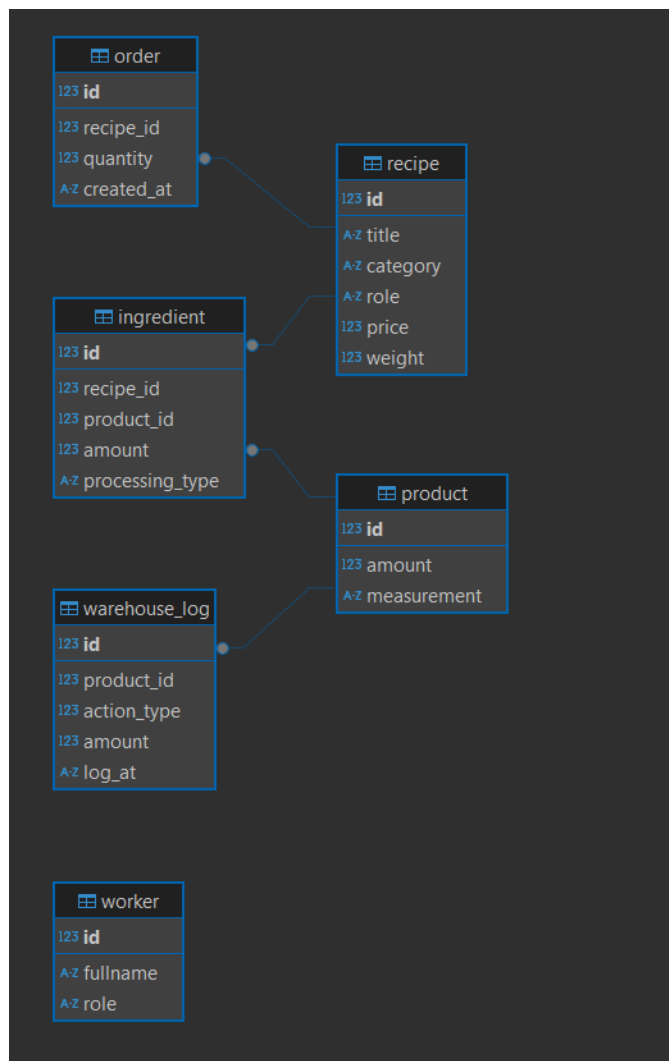


Рисунок 3.1.1 – Схема зв'язків таблиць між собою

3.2 Опис задач автоматизації та інтерфейсу користувача

ВИСНОВКИ

ПЕРЕЛІК ПОСИЛАНЬ

<https://www.indeed.com/career-advice/finding-a-job/kitchen-staffs> - Інформація
про професії на кухні ресторану

<https://www.sqlite.org/docs.html> - Документація SQLite