

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. Статистика витрат та поставок продуктів.
3. Кількість продуктів на складі.
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	17
ВИСНОВКИ.....	30
ПЕРЕЛІК ПОСИЛАНЬ.....	31

ВСТУП

У сучасних умовах автоматизація бізнес-процесів відіграє ключову роль у підвищенні ефективності роботи підприємств різних сфер. Галузь харчування не є винятком — для організації роботи кухні важливо мати інструмент, який дозволяє швидко керувати замовленнями, стежити за наявністю продуктів, аналізувати витрати та контролювати виконання рецептів.

Метою даної курсової роботи є проектування та реалізація інформаційної системи для управління кухнею, яка забезпечує зручний інтерфейс для співробітників із різними ролями (повар, офіціант, шеф-кухар), дозволяє створювати та обробляти замовлення, зберігати інформацію про рецепти, інгредієнти, складські запаси та дії з продуктами.

У процесі роботи буде побудована ER-діаграма, сформована реляційна модель даних, обрана та реалізована фізична модель бази даних на основі СУБД SQLite. Для програмної реалізації буде застосовано технології Electron, React і TypeScript, що дозволить створити кросплатформовий настільний додаток.

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 страви які вони можуть готувати, та до списку запасів на складі для створення замовлень у постачальників та перегляду статистики по витратам/поставкам продукту.
- Шеф-де-парті, шеф-де-кузін, соус-шеф-де-кузін – мають мати доступ до рецептів які доступні їм по їх особливій спеціалізації, прийняття замовлень та зміни кількості продуктів на складі 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 – число, обов’язкове;
 - Одиниця виміру – число, обов’язкове;
- Дія з продуктом:
 - Ідентифікатор дії – унікальне число, обов’язкове;
 - Ідентифікатор продукту – унікальне число, обов’язкове;
 - Тип дії – (витрачення, додавання), обов’язкове;
 - Кількість витраченого – число, обов’язкове;
 - Дата здійснення дії – дата, обов’язкове;

Кількість продуктів на складі може змінюватися після закупки, після витрати продукту. Офіціант може тільки додавати замовлення, але не редагувати чи видаляти їх.

Трата зі складу та замовлення блюд в кухню робиться через програму.

Функції додатку:

- Замовлення продуктів на склад (має доступ тільки шеф-кухар);
- Редагування рецептів та цін на блюда (має доступ тільки шеф-кухар);
- Перегляд інформації про страви які можуть готувати кухари (має доступ тільки шеф-кухар);
- Перегляд статистики по продуктам на складі (має доступ тільки шеф кухар);
- Список товарів та їх кількість на складі (мають доступ тільки шеф кухар);
- 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).

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).

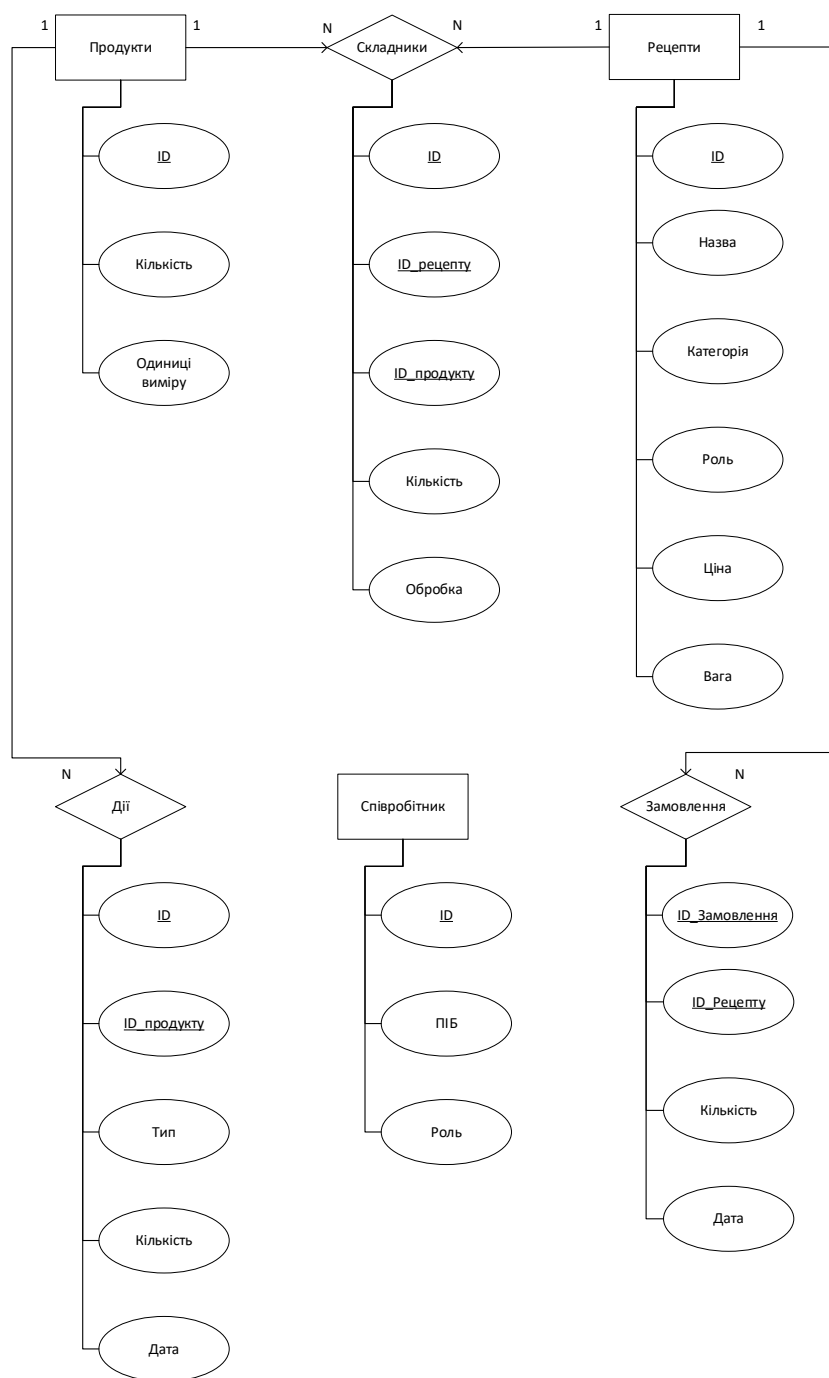


Рисунок 2.1.1 – ER-діаграма

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

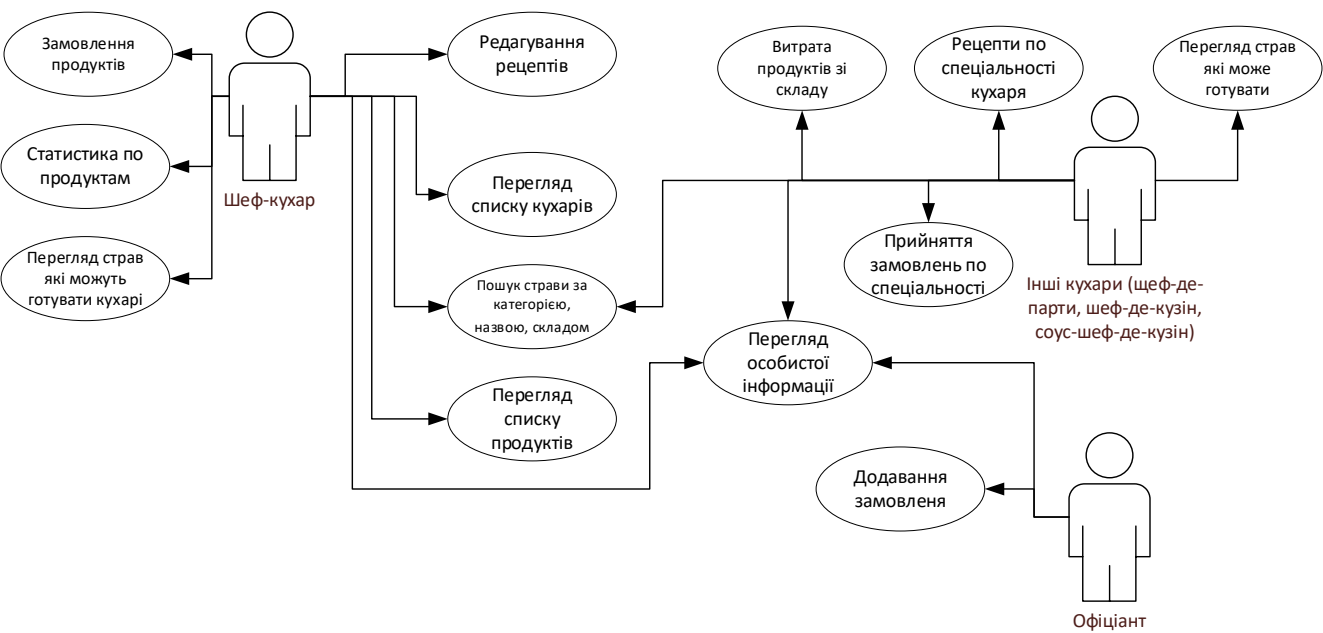


Рисунок 2.1.2 – Use Case

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	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
name	TEXT	not null, unique	Назва 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 NOT NULL PRIMARY KEY AUTOINCREMENT,  
    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 NOT NULL PRIMARY KEY AUTOINCREMENT,  
    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 NOT NULL PRIMARY KEY AUTOINCREMENT,  
    name TEXT NOT NULL UNIQUE,  
    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 NOT NULL PRIMARY KEY AUTOINCREMENT,
    recipe_id INTEGER NOT NULL,
    quantity INTEGER NOT NULL,
    created_at TEXT NOT NULL DEFAULT (DATETIME('now', 'localtime')),
    FOREIGN KEY (recipe_id) REFERENCES recipe(id) ON DELETE
CASCADE
);
```

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

```
CREATE TABLE IF NOT EXISTS "ingredient" (
    id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT,
    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) ON DELETE
CASCADE,
    FOREIGN KEY (product_id) REFERENCES product(id) ON DELETE
CASCADE
);
```

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 NOT NULL PRIMARY KEY AUTOINCREMENT,
    product_id INTEGER NOT NULL,
    action_type INTEGER CHECK(action_type IN (1, -1)) NOT NULL,
    amount INTEGER NOT NULL,
```

```

log_at TEXT NOT NULL DEFAULT (DATETIME('now', 'localtime')),
FOREIGN KEY (product_id) REFERENCES product(id) ON DELETE
CASCADE
);

```

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

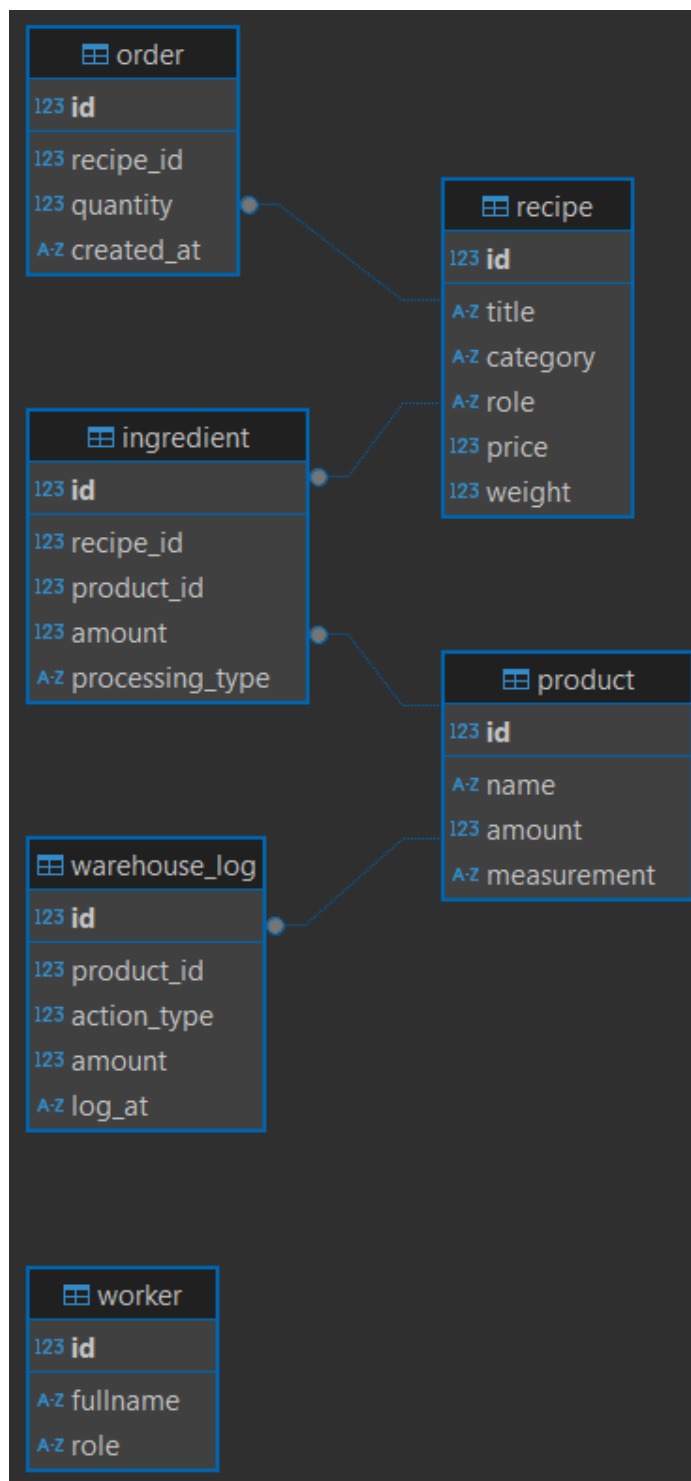


Рисунок 3.1.1 – Схема зв'язків таблиць між собою

3.2 Опис задач автоматизації та інтерфейсу користувача

Для забезпечення автоматизації робимо тригери для бази даних.

Створення тригера на видалення з таблиці замовлень (видалення відбувається тільки коли замовлення виконано, відміна замовлення не передбачена програмою) для запису витрат в журнал дій виглядає так:

```
CREATE TRIGGER log_expenses_after_order_delete AFTER DELETE ON
"order"
FOR EACH ROW
BEGIN
    INSERT INTO "warehouse_log" (product_id, action_type, amount)
    SELECT i.product_id, -1, i.amount FROM "ingredient" i
    WHERE i.recipe_id = OLD.recipe_id;
END;
```

Створення тригера для зміни кількості продуктів на складі виходячи зі змін у журналі дій складу виглядає так:

```
CREATE TRIGGER change_product_amount_after_warehouse_log_insert AFTER
INSERT ON "warehouse_log"
FOR EACH ROW
BEGIN
    UPDATE product
    SET amount = amount + (NEW.amount * NEW.action_type)
    WHERE id = NEW.product_id;
END;
```

Сам додаток для роботи з базою даних буде написаний на JavaScript. Для написання буде використаний фреймворк Electron з надстройками для праці з React через Vite та TypeScript. Для запитів до бази даних буде використана бібліотека better-sqlite3 яка виконує запити не асинхронно(тобто синхронно).

А тепер розглянемо інтерфейс користувача. Коли користувач входить до програми його зустрічає вікно логіну (Рисунок 3.2.1), куди треба ввести ім'я фамілію та роль користувача, щоб увійти в акаунт (приклад на Рисунку 3.2.2). Після входу до акаунту користувач потрапляє до головного вікна (Рисунок 3.2.3), де в лівому верхньому кутку є select панель, де користувач може обирати вікно (таблицю) з доступних йому для відображення, але початкове вікно – замовлення. На Рисунку 3.2.4 зображено вікно з інформацією про замовлення, де повар може сказати, що воно виконано або закрити його та обрати інше замовлення. Також, на головному вікні, якщо користувач натисне кнопку з його ім'ям – то відкриється інформація про користувача (Рисунок 3.2.5), де користувач може вийти з акаунту. На Рисунку 3.2.6 продемонстрований список таблиць доступних Шеф кухарю. На вікні з рецептами Шеф кухар може бачити усі рецепти (Рисунок 3.2.7). На Рисунку 3.2.8 зображено вікно пошуку рецепту, де можна знайти рецепт за назвою або категорію або інгредієнтом. Якщо один чи більше рецептів були знайдені – відобразиться таблиця з результатами (Рисунок 3.2.9). На вікні з рецептами Шеф кухарю є можливість додати новий рецепт за допомогою кнопки з відповідним текстом. Інтерфейс додавання нового рецепту продемонстровано на Рисунку 3.2.10. Також Шеф кухар може редагувати вже існуючі рецепти (Рисунок 3.2.11). Більш зручна таблиця для моніторингу цін та категорій різних блюд в одній таблиці – вікно «Меню», яке продемонстровано на Рисунку 3.2.12. На Рисунку 3.2.13 зображено вікно складу, там Шеф кухар може додати продукт на склад (створити замовлення) (Рисунок 3.2.15) чи переглянути скільки залишилось продуктів на складі. Якщо Шеф кухар натисне на рядок з продуктом, то йому відкриється таблиця з інформацією зі всіма діями з цим продуктом (Рисунок 3.2.14), там відображаються не тільки витрати а ще й поставки (Рисунок 3.2.16). І останнє вікно доступне для Шеф кухара – вікно з користувачами (Рисунок 3.2.17) на якому знаходиться список усіх користувачів маючих доступ до додатку та їх ролі. Якщо Шеф кухар натисне на рядок з цікавим йому користувачем – відкриється вікно зі всіма рецептами, що може готувати кухар. Тепер змінимо акаунт та зайдемо від імені Петра, офіціанта. Йому доступно лише два вікна – «Меню» та «Замовлення» (Рисунок 3.2.19). На вікні з замовленнями він може тільки переглядати

інформацію про замовлення, а на вікні з меню, при натисканні лівою кнопкою миші на позицію в меню – він отримує інформацію про рецепт, а правою кнопкою додає позицію до замовлення, після додавання висвічується віконце в якому треба обрати кількість порцій (Рисунок 3.2.20). А зараз зайдемо до аккаунту Соус-шеф-де-кузін Максиму, та побачимо що йому теж доступно тільки два вікна «Замовлення» та «Рецепти» (Рисунок 3.2.21), але він бачить тільки ті замовлення, які може прийняті та тільки ті рецепти, які може зробити.

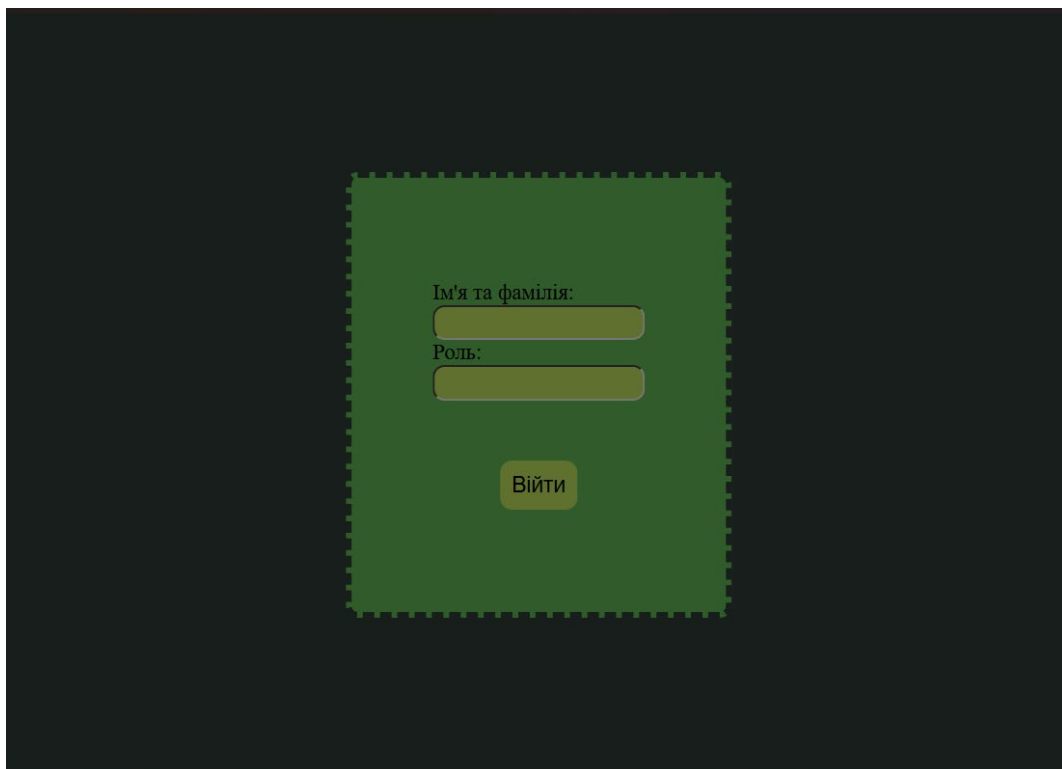


Рисунок 3.2.1 – Вікно логіну

Ім'я та фамілія:
Павло Пономарьов

Роль:
executive chef

Війти

Рисунок 3.2.2 – Вікно логіну з інформацією

Замовлення ▼

Павло Пономарьов

Назва страви	Кількість порцій	Роль кухаря	Час створення
Плов з куркою	1	cook chef	04-13 16:36
Млинці з м'ясом	11	sous chef	04-13 16:36
Олів'є	1	salad chef	04-13 16:38

Рисунок 3.2.3 – Головне вікно програми

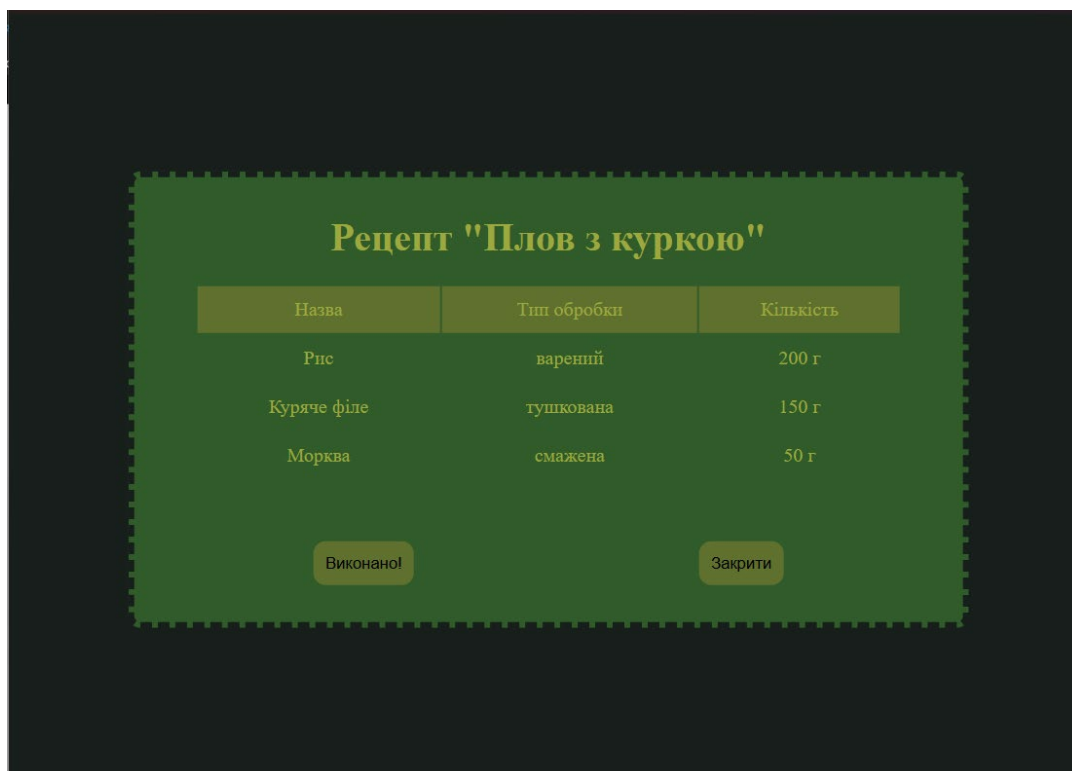


Рисунок 3.2.4 – Вікно з інформацією про замовлення

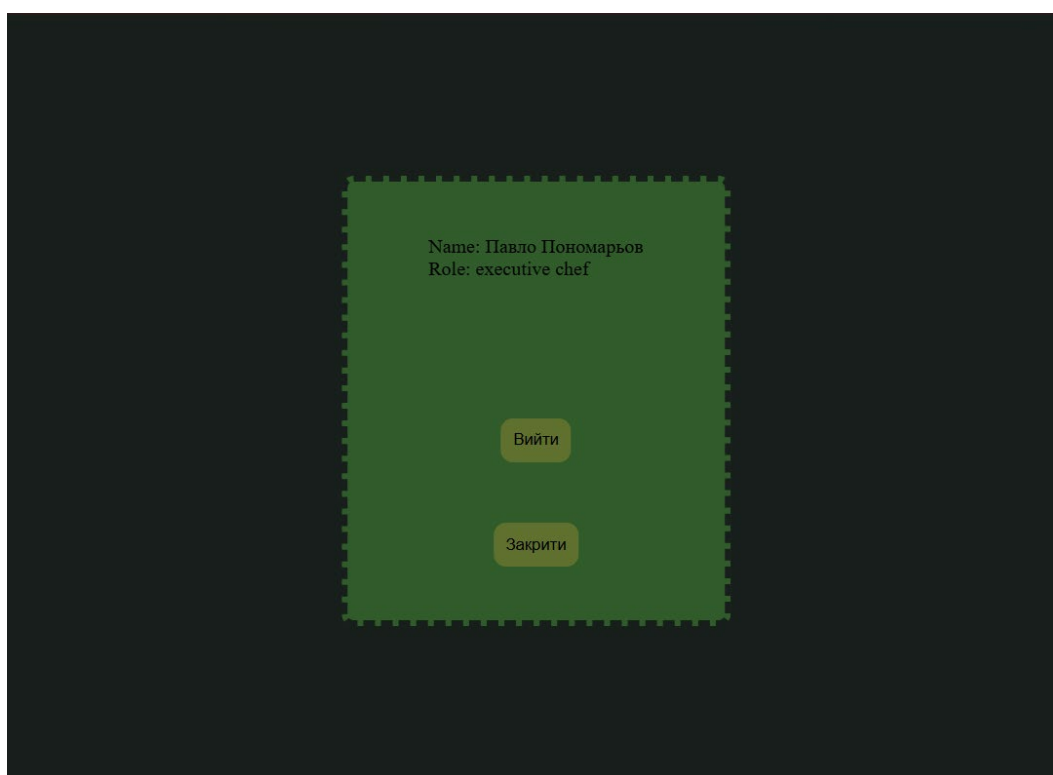


Рисунок 3.2.5 – Вікно з інформацією про користувача

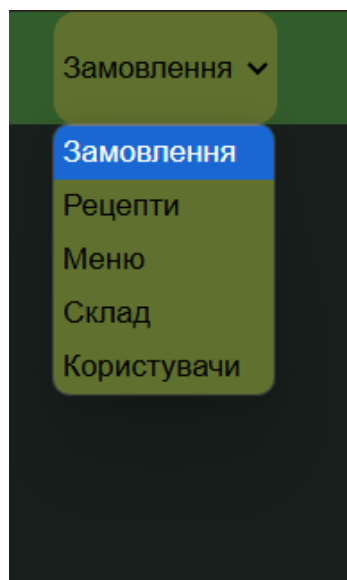


Рисунок 3.2.6 – Які таблиці доступні Шеф кухарю

Рецепти ▾

Павло Пономарьов

Пошук

Додати рецепт

Назва страви	Роль кухаря	Вага
Борщ український	cook chef	350
Картопляне пюре з котлетою	cook chef	400
Олів'є	salad chef	250
Вінегрет	salad chef	250
Сирніки зі сметаною	sous chef	300
Макарони по-флотськи	cook chef	400
Гречка з підливою	cook chef	350
Плов з куркою	cook chef	400
Салат "Весняний"	salad chef	200
Млинці з м'ясом	sous chef	300
Омлет з сиром	sous chef	250

Рисунок 3.2.7 – Вікно з рецептами

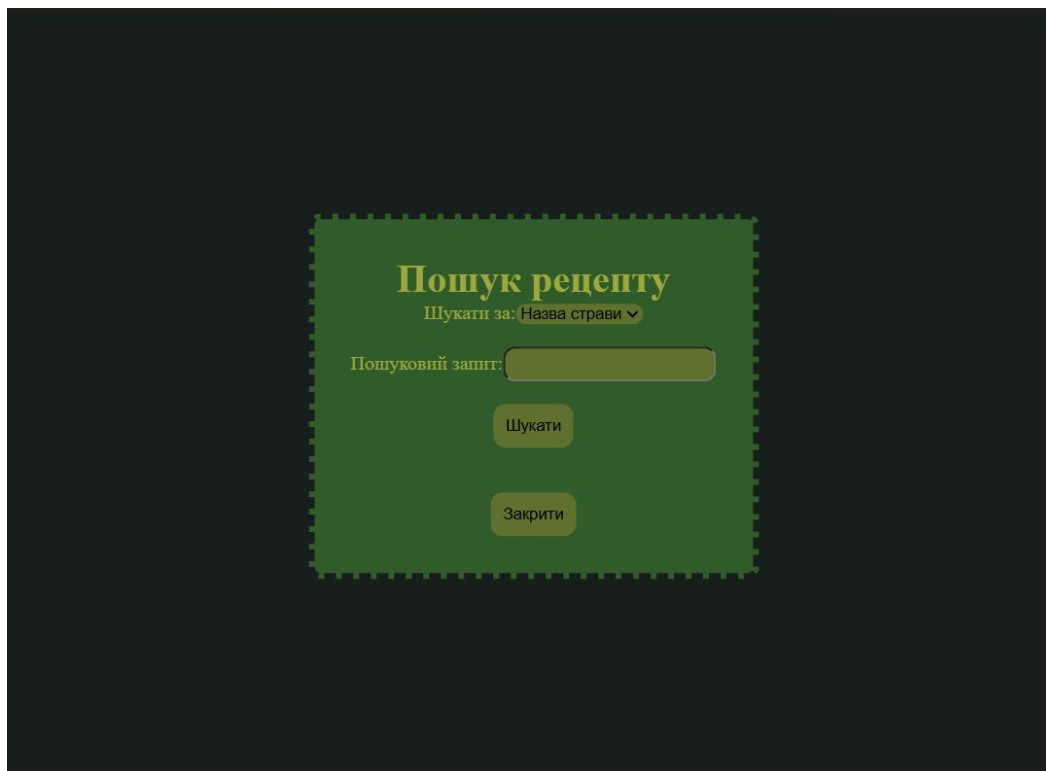


Рисунок 3.2.8 – Вікно пошуку рецепта

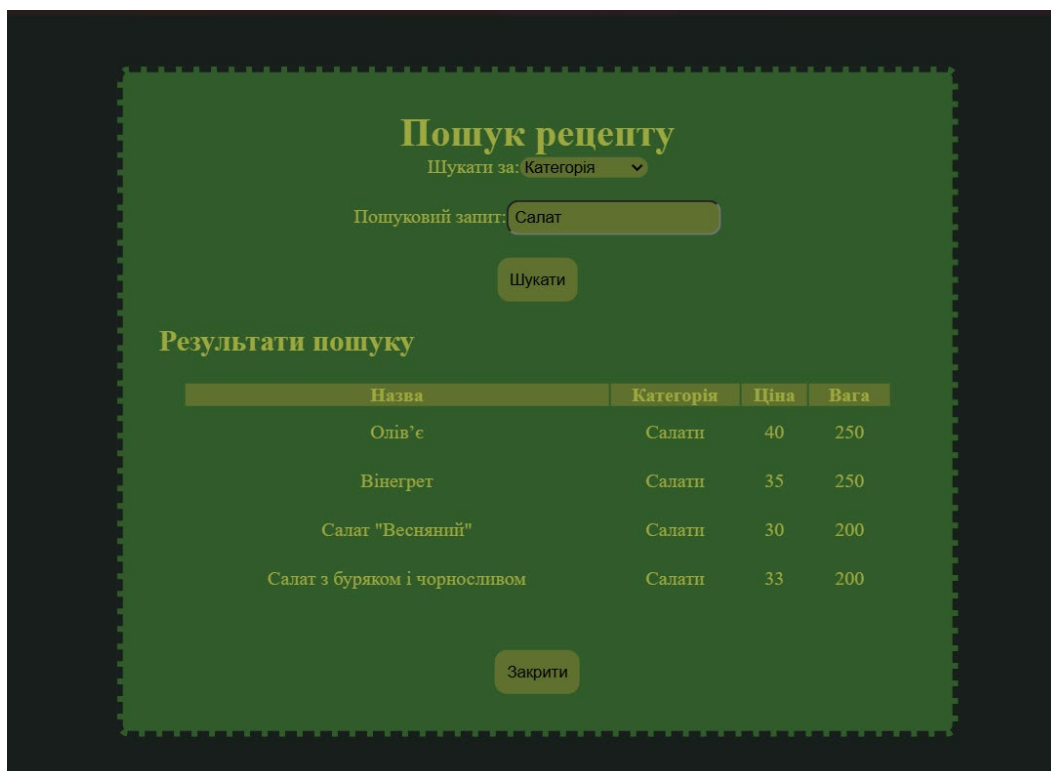


Рисунок 3.2.9 – Вікно пошуку коли є результати

Рецепт "Новий рецепт"

Назва	<input type="text" value="Новий рецепт"/>
Категорія	<input type="text"/>
Роль	<input type="text"/>
Ціна	<input type="text" value="0.00"/>
Вага	<input type="text" value="0"/>

Інгредієнти

Продукт	Кількість	Тип обробки	
<div style="display: flex; justify-content: space-between; align-items: center; margin-top: 10px;"> + Додати інгредієнт Зберегти Видалити рецепт Скасувати </div>			

Рисунок 3.2.10 – Вікно додавання рецепту

Рецепт "Борщ український"

Назва	<input type="text" value="Борщ український"/>
Категорія	<input type="text" value="Супи"/>
Роль	<input type="text" value="cook chef"/>
Ціна	<input type="text" value="45.50"/>
Вага	<input type="text" value="350"/>

Інгредієнти

Продукт	Кількість	Тип обробки	
<input type="text" value="Картопля"/>	<input type="text" value="300"/> г	<input type="text" value="варена"/>	<input type="button" value="✖"/>
<input type="text" value="Морква"/>	<input type="text" value="200"/> г	<input type="text" value="тушкована"/>	<input type="button" value="✖"/>
<input type="text" value="Цибуля ріпчаста"/>	<input type="text" value="150"/> г	<input type="text" value="тушкована"/>	<input type="button" value="✖"/>
<div style="display: flex; justify-content: space-between;"> + Додати інгредієнт Зберегти Видалити рецепт Скасувати </div>			

Рисунок 3.2.11 – Вікно редагування рецепту

Меню

Павло Пономарьов

Пошук

Назва страви	Категорія	Ціна за порцію	Вага (грамм)
Борщ український	Супи	45.5	350
Картопляне пюре з котлетою	Основні страви	60	400
Олів'є	Салати	40	250
Вінегрет	Салати	35	250
Сирники зі сметаною	Сніданки	50	300
Макарони по-флотськи	Основні страви	55	400
Гречка з підливою	Основні страви	45	350
Плов з куркою	Основні страви	65	400
Салат "Весняний"	Салати	30	200
Млинці з м'ясом	Сніданки	55	300
Омлет з сиром	Сніданки	35	250

Рисунок 3.2.12 – Вікно меню

Склад

▼

Павло Пономарьов

Додати продукт

Пошук

Назва продукту	Кількість
Батон	1101703 г
Борошно пшеничне	9900 г
Горошок зелений	1000 г
Гречка	3300 г
Зелень	500 г
Капуста білокачанна	6000 г
Картопля	189550 г
Кефір	2000 мл
Кукурудза консервована	950 г
Куряче філе	2850 г
Макарони	5000 г

Рисунок 3.2.13 – Вікно складу

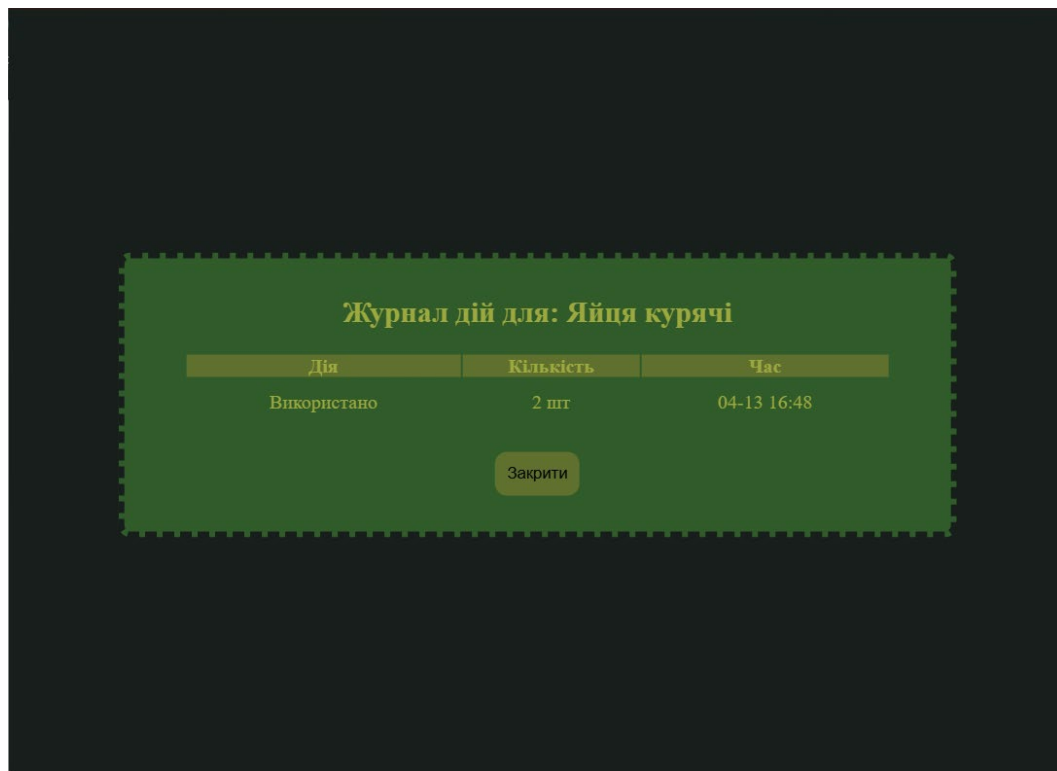


Рисунок 3.2.14 – Вікно перегляду витрат продукту

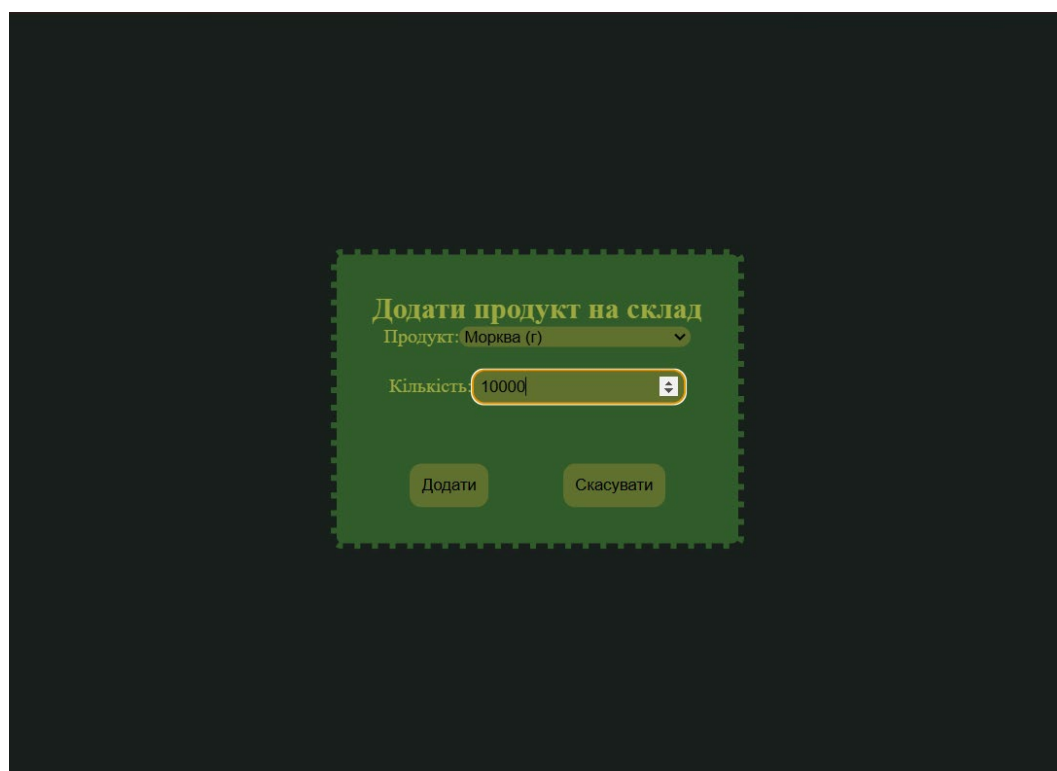


Рисунок 3.2.15 – Вікно додавання продукту на склад

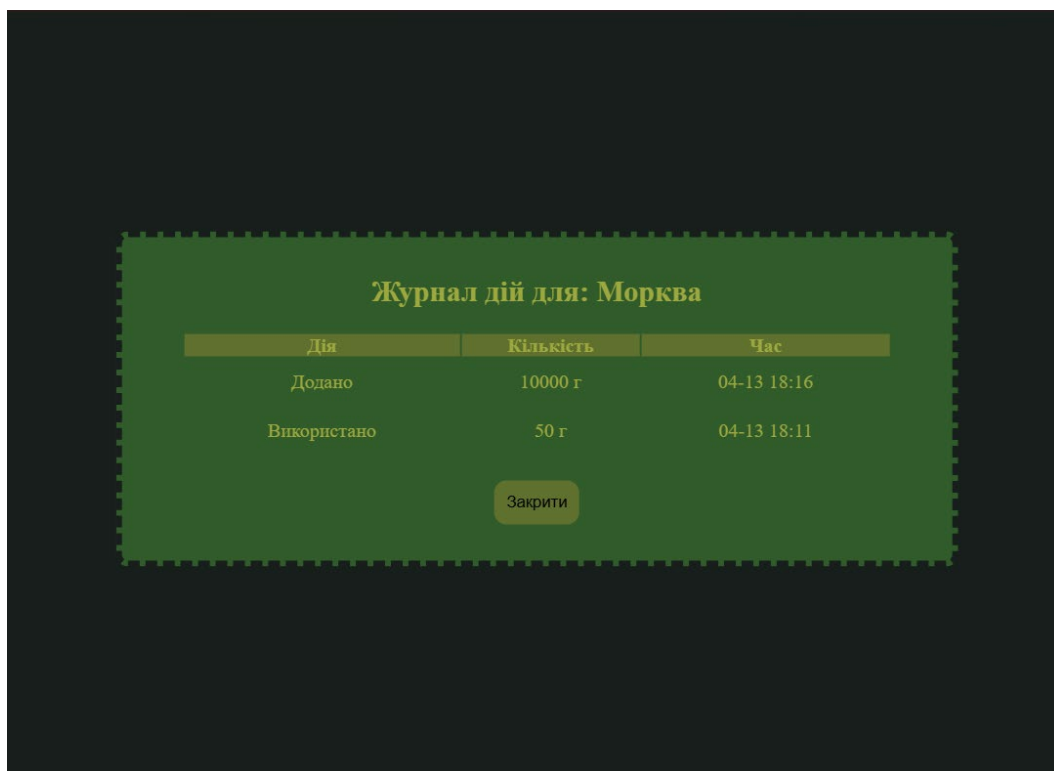


Рисунок 3.2.16 – Вікно для перегляду витрат та поставок продукту



Рисунок 3.2.17 – Вікно з користувачами

Рецепти доступні для "Олена Ковальчук"

Назва страви	Категорія
Борщ український	Супи
Картопляне пюре з котлетою	Основні страви
Макарони по-флотськи	Основні страви
Гречка з підливою	Основні страви
Плов з куркою	Основні страви
Крем-суп із гарбуза	Супи
Шніцель зі свинини	Основні страви
Тупкована капуста з м'ясом	Основні страви
Рагу овочеве	Основні страви
Котлета по-київськи	Основні страви
Суп з фрикадельками	Супи

Рисунок 3.2.18 – Вікно з рецептами доступними повару

Меню

Замовлення

Меню

Петро Сидоренко

Пошук

Назва страви	Категорія	Ціна за порцію	Вага (грамм)
Борщ український	Супи	45.5	350
Картопляне пюре з котлетою	Основні страви	60	400
Олів'є	Салати	40	250
Вінегрет	Салати	35	250
Сирники зі сметаною	Сніданки	50	300
Макарони по-флотськи	Основні страви	55	400
Гречка з підливою	Основні страви	45	350
Плов з куркою	Основні страви	65	400
Салат "Весняний"	Салати	30	200
Млинці з м'ясом	Сніданки	55	300
Омлет з сиром	Сніданки	35	250

Рисунок 3.2.19 – Вікна доступні офіціанту

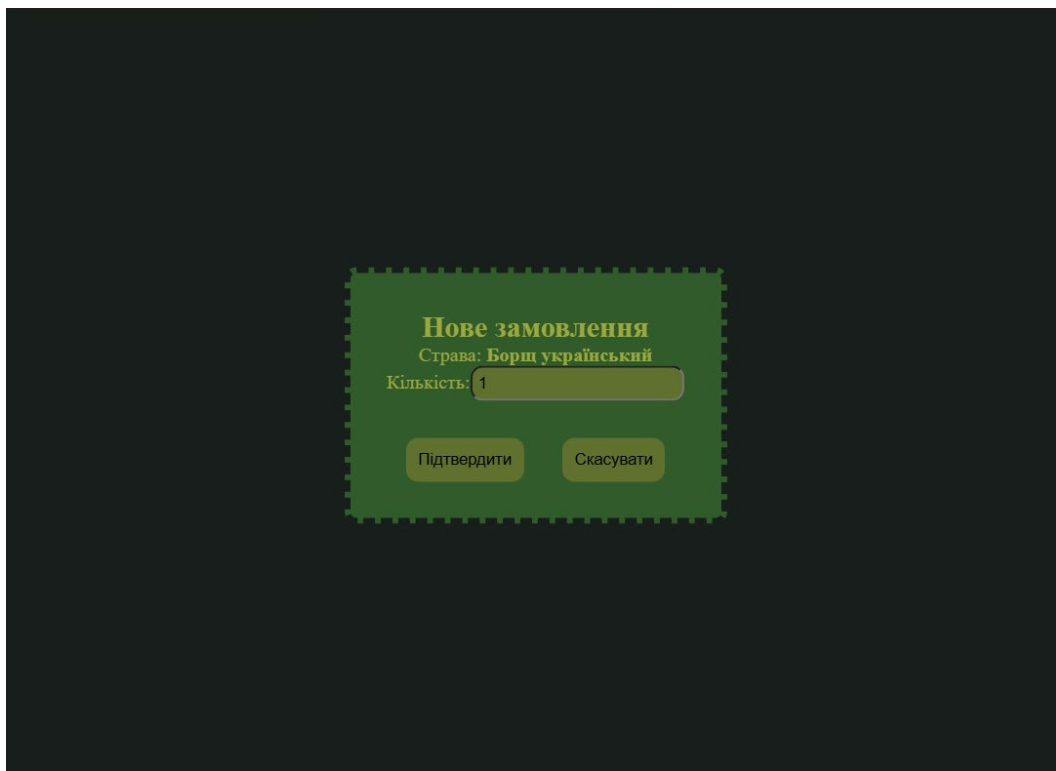


Рисунок 3.2.20 – Вікно додавання замовлення (ведення кількості порцій)



Рисунок 3.2.21 – Вікна доступні кухарю

ВИСНОВКИ

У результаті виконання курсової роботи було реалізовано повнофункціональний застосунок для автоматизації роботи кухні. Побудована реляційна модель даних, що охоплює основні сутності кухні, такі як співробітники, рецепти, замовлення, продукти та дії зі складом. Було створено фізичну модель бази даних на SQLite та забезпечено її взаємозв'язки за допомогою зовнішніх ключів і тригерів.

Програмна реалізація на базі Electron, React і TypeScript надала змогу створити гнучкий та масштабований додаток, що адаптується до ролей користувачів і надає різний функціонал в залежності від прав доступу.

Таким чином, поставлені задачі були успішно реалізовані. Система є зручною для користувачів, підтримує ведення складу, облік витрат, керування рецептами та формування замовлень. У перспективі можливе розширення функціоналу, інтеграція з хмарними сервісами або створення мобільної версії для зручності роботи персоналу.

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

<https://electron-vite.org> – Документація Electron Vite

<https://react.dev/reference/react> - Документація React

<https://www.electronjs.org> – Документація Electron

<https://www.typescriptlang.org> – Документація Typescript

<https://www.w3schools.com/css/> - Тutorials по CSS