

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет водного господарства та природокористування
Навчально-науковий інститут кібернетики, інформаційних технологій та
інженерії
Кафедра комп'ютерних технологій та економічної кібернетики

Допущено до захисту:

Завідувач кафедри

_____ д. е. н., проф. П. М. Грицюк
« _____ » _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня «бакалавр»

за освітньо-професійною програмою «Інформаційні системи і технології»
спеціальності 126 «Інформаційні системи та технології»

на тему: **«Проектування та розробка інформаційної системи роботи
ресторану»**

Виконав:

здобувач вищої освіти 4 курсу, групи ІСТ-41
Бровчук Володимир Миколайович

Керівник:

канд. техн. наук, доцент Гладка О. М.

Рецензент:

канд. техн. наук, доцент Джоші О. І.

Рівне – 2025

Національний університет водного господарства та природокористування
 ННІ кібернетики, інформаційних технологій та інженерії
 Кафедра комп'ютерних технологій та економічної кібернетики

Освітньо-кваліфікаційний рівень – **бакалавр**
 за освітньо-професійною програмою **«Інформаційні системи і технології»**
 спеціальність **126 «Інформаційні системи та технології»**

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ д. е. н., проф. П. М. Грицюк

«_____» _____ 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачу _____ Бровчуку Володимиру Миколайовичу
 (прізвище, ім'я, по-батькові)

1. Тема роботи _____ Проектування та розробка інформаційної системи роботи ресторану

керівник роботи: _____ Гладка Олена Миколаївна, канд. техн. наук, доцент
 (прізвище, ім'я, по-батькові, науковий ступінь, вчене звання)

затверджена наказом по університету від “_____” _____ 2025 р. С № _____

2. Термін здачі здобувачем закінченої роботи “09” червня 2025 р.

3. Вихідні дані до роботи розробити інформаційну систему для обліку роботи ресторану, що передбачає проектування та створення бази даних, розробку відповідних форм і функціональних модулів інформаційної системи.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) вступ; аналіз предметної області; вибір та обґрунтування засобів розробки програмного продукту; розробка і опис створеної інформаційної системи.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв
1.	Гладка О. М., доцент		
2.	Гладка О. М., доцент		
3.	Гладка О. М., доцент		

7. Дата видачі завдання “_11_” листопада 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломної роботи	Термін виконання етапів роботи	Примітка
1.	Аналіз об’єкта дослідження, виявлення існуючих проблем	11.11.24 – 25.11.24	
2.	Аналіз існуючих інформаційних методів (технологій) вирішення проблеми	28.11.24 – 23.12.24	
3.	Вибір та обґрунтування засобів розробки інформаційної системи	30.01.25 – 24.02.25	
4.	Проектування, розробка та реалізація інформаційної системи	27.02.25 – 30.04.25	
5.	Підготовка тексту кваліфікаційної роботи	10.04.25 – 26.05.25	
6.	Підготовка презентації роботи	29.05.25 – 09.06.25	
7.	Відгук керівника, рецензування роботи, перевірка на плагіат	09.06.25 – 16.06.25	

Здобувач _____ **В. М. Бровчук**
 (підпис) (прізвище і ініціали)

Керівник кваліфікаційної роботи

_____ **О. М. Гладка**
 (підпис) (прізвище і ініціали)

АНОТАЦІЯ

Бровчук В.М. Проектування та розробка інформаційної системи підтримки роботи ресторану. Кваліфікаційна робота на здобуття освітнього ступеня «бакалавр»: 92 с., 22 рис., 16 табл., 5 додатків на 15 стор., 15 літературних джерел.

Метою кваліфікаційної роботи є розробка ефективної інформаційної системи для автоматизації роботи ресторану, яка дозволяє вести облік замовлень, клієнтів, меню, складу та звітності, а також забезпечити зручний і зрозумілий інтерфейс для користувачів. Об'єкт дослідження – автоматизація управління діяльністю підприємства ресторанного господарства, зокрема процесів, пов'язаних з обліком замовлень, управлінням поставаннями, роботою персоналу, а також контролем фінансових показників. Предмет дослідження – методи, засоби та інструменти проектування і розробки інформаційної системи для автоматизації основних бізнес-процесів ресторану, з використанням середовища розробки баз даних Microsoft Access.

У роботі розглянуто процес проектування та розробки інформаційної системи для автоматизації діяльності ресторану. Система реалізована за допомогою програмного середовища Microsoft Access і охоплює основні аспекти роботи закладу: облік персоналу, управління меню, реєстрацію замовлень, контроль залишків інгредієнтів і фіксацію поставань. У межах роботи було здійснено аналіз предметної області, спроектовано структуру бази даних, нормалізовано таблиці, створено форми для зручного введення та перегляду інформації, налаштовано події, макроси та звіти. Автоматизовано ключові операції, зокрема списання та додавання інгредієнтів при здійсненні замовлень і поставань. Результати дослідження можуть бути використані для впровадження у невеликих закладах харчування з метою підвищення ефективності управлінських і облікових процесів.

Ключові слова: інформаційна система, база даних, Microsoft Access, автоматизація управління, проектування.

ЗМІСТ

ВСТУП.....	6
РОЗДІЛ 1. ВИВЧЕННЯ МОЖЛИВОСТЕЙ АВТОМАТИЗАЦІЇ БІЗНЕС-ПРОЦЕСІВ УПРАВЛІННЯ ДІЯЛЬНІСТЮ РЕСТОРАНУ.....	9
1.1.АНАЛІЗ ПРОЦЕСІВ ДІЯЛЬНОСТІ РЕСТОРАНУ	9
1.2.Дослідження проблем діяльності РЕСТОРАНУ.....	12
1.3.АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ(ПРОГРАМ) ДЛЯ АВТОМАТИЗАЦІЇ РОБОТИ РЕСТОРАНУ... ..	14
1.4.АНАЛІЗ ПРОГРАМ АНАЛОГІВ	17
РОЗДІЛ 2. ФОРМАЛІЗОВАНА ПОСТАНОВКА ЗАДАЧІ ТА ПОБУДОВА МОДЕЛІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ РОБОТИ РЕСТОРАНУ	21
2.1. ФОРМАЛІЗОВАНА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ.....	21
2.2. ОПИС РОЗРОБЛЕНОЇ ІНФОРМАЦІЙНОЇ (АНАЛІТИЧНОЇ) МОДЕЛІ	24
2.3. МЕТОДИ ТА АЛГОРИТМИ РЕАЛІЗАЦІЇ МОДЕЛІ	27
РОЗДІЛ 3. ОПИС РОЗРОБЛЕНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ РОБОТИ РЕСТОРАНУ.....	30
3.1. ПРОЕКТУВАННЯ ТАБЛИЦЬ БАЗИ ДАНИХ	30
3.2. НОРМОВАНА ФОРМА БАЗИ ДАНИХ	35
3.3. ОПИС ФОРМ РОЗРОБЛЕНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ	39
3.4. ОПИС ПОДІЙ ТА МАКРОСІВ ДЛЯ АВТОМАТИЗАЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ... ..	54
3.5. ОПИС СТРУКТУРИ ЗВІТІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	60
3.6. РЕКОМЕНДАЦІЇ ЩОДО ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ТА КОРОТКІЙ ПОСІБНИК КОРИСТУВАЧА	67
ВИСНОВКИ	72
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	74
ДОДАТКИ	76

ВСТУП

У сучасних умовах високої конкуренції у сфері ресторанного бізнесу ефективне управління внутрішніми процесами є одним із ключових чинників успішної діяльності підприємства. Автоматизація обліку, замовлень, обслуговування клієнтів, контролю запасів та інших бізнес-процесів дозволяє значно підвищити швидкість, точність і якість обслуговування, що, у свою чергу, сприяє зростанню прибутковості та покращенню іміджу закладу.

Інформаційні технології сьогодні є невід'ємною складовою ефективного функціонування закладів громадського харчування. Ресторани, кафе та інші заклади харчування все частіше впроваджують автоматизовані системи для підвищення ефективності обслуговування клієнтів, оптимізації внутрішніх процесів і покращення якості управлінських рішень.

Проте багато ресторанів досі використовують застарілі або розрізнені системи, що ускладнює управління та веде до втрати даних, людських помилок і зниження загальної ефективності. Саме тому актуальним є розроблення комплексної інформаційної системи, яка дозволяє автоматизувати основні бізнес-процеси ресторану.

Метою даної кваліфікаційної бакалаврської роботи є проектування та розробка інформаційної системи, яка забезпечує автоматизацію основних бізнес-процесів ресторану: оформлення замовлень, ведення обліку страв та інгредієнтів, управління персоналом, формування звітності тощо.

Тема дослідження: Проектування та розробка інформаційної системи роботи ресторану з використанням засобів Microsoft Access.

Об'єкт дослідження - процеси прийому та обробки замовлень, управління меню, облік запасів на складі, контроль роботи персоналу, а також формування звітності для керівництва.

Предмет дослідження - методи, засоби та інструменти проектування і розробки інформаційної системи для автоматизації основних бізнес-процесів ресторану, з використанням середовища розробки баз даних Microsoft Access.

Актуальність теми визначається практичною необхідністю цифрової трансформації у сфері громадського харчування, що стає запорукою конкурентоспроможності й сталого розвитку закладів. Розроблена в межах цієї роботи система може бути використана як прототип для впровадження в невеликих та середніх закладах харчування, де важливими є доступність, гнучкість і простота у використанні програмного забезпечення.

Завдання дослідження:

1. Проаналізувати специфіку бізнес-процесів у ресторанній сфері та визначити потреби у їх автоматизації.
2. Обґрунтувати доцільність використання Microsoft Access як основного інструменту розробки інформаційної системи.
3. Розробити концептуальну модель інформаційної системи: визначити основні сутності, атрибути та взаємозв'язки між ними.
4. Реалізувати інформаційну систему у Microsoft Access, включаючи:
 - створення таблиць бази даних;
 - розробку форм для введення та перегляду інформації;
 - створення запитів для пошуку та обробки даних;
 - формування звітів для аналізу роботи ресторану.
5. Провести тестування системи на прикладі типових сценаріїв роботи ресторану.
6. Оцінити ефективність впровадженого рішення та запропонувати шляхи його удосконалення або розширення.

Виконання цих завдань дозволить створити прикладну інформаційну систему, адаптовану до потреб малого або середнього ресторану, з можливістю подальшої модифікації та масштабування.

У межах цієї роботи розглядаються питання побудови структури бази даних, логіки взаємодії між її основними складовими, розробки інтерфейсів користувача та забезпечення зручного доступу до даних для виконання щоденних операцій. Особливу увагу приділено аналізу функціональних вимог,

що притаманні ресторанному бізнесу, а також адаптації програмного рішення до цих вимог.

Для реалізації поставленої задачі було обрано середовище Microsoft Access, яке забезпечує зручні засоби для створення баз даних, форм введення та виводу інформації, а також підтримку макросів і модулів для реалізації логіки бізнес-процесів.

Використання Microsoft Access дозволяє швидко створити реляційну базу даних із дружнім інтерфейсом, що не потребує складного налаштування або додаткового програмного забезпечення. Завдяки можливості інтеграції з іншими продуктами Microsoft Office та простоті використання, ця платформа є доцільним вибором для малих та середніх підприємств, зокрема ресторанів.

У процесі розробки були застосовані методи аналізу вимог користувачів, моделювання бізнес-процесів, побудови реляційних баз даних, а також реалізовано елементи автоматизації за допомогою макросів та мови програмування VBA (Visual Basic for Applications), яка підтримується у Microsoft Access.

Результатом роботи є програмний продукт, створений у середовищі Microsoft Access, який забезпечує автоматизацію ключових процесів ресторану, має зручний інтерфейс користувача та забезпечує стабільну роботу в умовах реального використання.

Структура роботи включає вступ, три розділи, висновки, список використаних джерел та додатки. У першому розділі проведено аналіз предметної області, досліджено проблеми ресторанної діяльності та огляд існуючих програмних рішень. У другому розділі сформульовано постановку задачі, побудовано інформаційну модель та обґрунтовано вибір інструментальних засобів. Третій розділ присвячено опису створеної інформаційної системи, її структури, функціональних можливостей та рекомендацій щодо впровадження.

РОЗДІЛ 1. ВИВЧЕННЯ МОЖЛИВОСТЕЙ АВТОМАТИЗАЦІЇ БІЗНЕС-ПРОЦЕСІВ УПРАВЛІННЯ ДІЯЛЬНІСТЮ РЕСТОРАНУ

1.1. Аналіз процесів діяльності ресторану

Ресторан є закладом, що здійснює обслуговування клієнтів, приготування та реалізацію страв і напоїв, а також забезпечує комфортне перебування відвідувачів у відповідному інтер'єрі та атмосфері.

Ресторан як суб'єкт господарської діяльності має складну внутрішню структуру та специфіку роботи, яка потребує ефективного управління інформаційними потоками. У типовому закладі харчування щоденно виконується велика кількість операцій: прийом клієнтів, обслуговування замовлень, облік витратних матеріалів (продуктів), взаємодія між кухнею та обслуговуючим персоналом, а також фінансовий облік. Усі ці процеси потребують точного й оперативного ведення обліку, швидкої обробки даних та координації дій між різними працівниками ресторану, що зумовлює необхідність впровадження автоматизованої інформаційної системи. У традиційній формі більшість із них виконуються вручну або за допомогою окремих програмних засобів, що часто призводить до дублювання інформації, помилок і зниження ефективності управління.

У зв'язку з цим виникає потреба у впровадженні єдиної інформаційної системи, яка забезпечить централізоване зберігання, обробку та доступ до необхідних даних усім відповідним користувачам. Така система має забезпечити автоматизацію операцій, контроль над ресурсами, швидке обслуговування клієнтів та підтримку прийняття управлінських рішень [5; 6].

У рамках даної роботи об'єкт дослідження представлений у вигляді умовного ресторану середнього рівня, що має стандартний набір функціональних зон: зал обслуговування, кухня, склад, адміністративний блок. Персонал включає офіціантів, кухарів, адміністратора та менеджера. Усі ці ролі

мають різні рівні доступу до системи, що також враховано при розробці програмного забезпечення.

Інформаційна система, що розробляється, повинна забезпечувати підтримку наступних основних функцій:

- зберігання та обробка даних про страви та напої, їх категорії, ціни та рецептури;
- оформлення замовлень клієнтів і фіксація стану виконання кожного з них;
- ведення обліку залишків на складі та списання продуктів відповідно до витрат;
- управління персоналом (облік робочих змін, ролей та дій працівників);
- формування звітності про продажі, популярні страви, завантаженість персоналу тощо.

Усі ці процеси часто реалізуються за допомогою окремих або ручних методів (наприклад, замовлення записуються в блокнотах, звітність формується вручну в Excel), що призводить до:

- дублювання інформації;
- помилок у розрахунках;
- втрати даних;
- складнощів у контролі запасів;
- зниження загальної ефективності роботи закладу.

На підставі аналізу можна зробити висновок, що об'єкт дослідження має високий потенціал для впровадження **інформаційної системи**, яка:

- централізує зберігання та обробку даних;
- забезпечить доступ до необхідної інформації відповідно до ролі користувача (офіціант, кухар, адміністратор, власник);
- мінімізує вплив людського фактору;
- підвищить якість обслуговування клієнтів.

У рамках дослідження передбачається створення інформаційної системи на базі **Microsoft Access**, яка дозволяє організувати реляційну базу даних із зручним користувацьким інтерфейсом, побудованим на основі форм, запитів та звітів. Такий підхід є доцільним для малих і середніх ресторанів, оскільки не потребує значних фінансових витрат та складних технічних знань для обслуговування.

Вибір платформи **Microsoft Access** для реалізації системи зумовлений її доступністю, підтримкою реляційних баз даних, зручним інтерфейсом для створення форм і звітів, а також можливістю швидкого налаштування згідно з потребами конкретного закладу[10; 15].

Таким чином, аналіз об'єкта дослідження підтверджує доцільність розробки інформаційної системи, що охоплює всі ключові процеси діяльності ресторану та забезпечує їх автоматизацію в інтегрованому середовищі.

Предметом дослідження в даній роботі є інформаційні процеси, методи та засоби автоматизації діяльності ресторану з використанням програмного середовища Microsoft Access.

У межах ресторанного бізнесу інформаційні процеси охоплюють збирання, обробку, зберігання, передавання та використання даних, необхідних для здійснення операційної діяльності.

До таких процесів належать:

- введення та оновлення даних про страви, напої, категорії меню;
- реєстрація замовлень клієнтів;
- облік використаних інгредієнтів і контроль запасів;
- ведення даних про персонал і розподіл обов'язків;
- формування управлінських звітів і аналітики.

У контексті даної роботи предмет дослідження зосереджено на моделюванні, проектуванні та реалізації інформаційної системи, яка забезпечує автоматизовану підтримку цих процесів. Особливу увагу приділено розробці структури бази даних, створенню форм введення та перегляду даних,

написанню запитів для вибірки інформації та генеруванню звітів на основі збережених даних.

Програмне середовище Microsoft Access обрано як базову платформу через його інтегровану підтримку баз даних, зручний графічний інтерфейс і можливість швидкої розробки прикладних рішень без потреби у складному програмуванні. Інструменти Access дозволяють ефективно організувати взаємодію користувача з даними, забезпечити перевірку введеної інформації, а також реалізувати логіку бізнес-процесів на основі макросів і модулів VBA (Visual Basic for Applications).

Таким чином, предметом дослідження є методи та засоби створення автоматизованої інформаційної системи для ресторану, орієнтованої на підвищення ефективності управлінських і облікових процесів, з використанням можливостей Microsoft Access для побудови бази даних, форм, запитів і звітів.

1.2. Дослідження проблем діяльності ресторану

У результаті аналізу об'єкта та предмета дослідження було встановлено, що в діяльності ресторану існує низка проблем, пов'язаних із неефективною організацією обліку, контролю та обробки інформації. Більшість процесів виконується вручну або з використанням окремих програмних засобів, які не забезпечують комплексної автоматизації. Це створює такі основні проблеми:

1. Відсутність централізованої системи обліку

У багатьох закладах громадського харчування інформація про замовлення, запаси інгредієнтів, робочі зміни персоналу тощо ведеться вручну або у вигляді окремих електронних таблиць. Це призводить до розрізненості даних, дублювання інформації, складності в контролі та аналізі.

2. Низька швидкість обробки замовлень

При використанні паперових бланків або неавтоматизованих форм для оформлення замовлень виникають затримки, особливо у години пік. Людський фактор може спричинити помилки у передачі замовлень на кухню або в розрахунках, що негативно впливає на якість обслуговування клієнтів.

3. Відсутність або обмеженість аналітики та звітності

Без автоматизованої системи адміністратор або керівництво ресторану не мають оперативного доступу до фінансової та логістичної інформації. Формування звітів займає багато часу і є неточним, що ускладнює прийняття стратегічних рішень.

4. Складність у контролі залишків та плануванні закупки інгредієнтів

Без наявності інтегрованої інформаційної системи про складські залишки у ресторані виникає ризик перевитрат або дефіциту інгредієнтів. Це призводить до зривів у роботі кухні, неможливості приготування певних страв і, як наслідок, до незадоволення клієнтів.

5. Відсутність гнучкості у масштабуванні та адаптації

Багато існуючих рішень або надто складні й дорогі для малих підприємств, або не дозволяють швидко адаптуватися під зміни в організації (нове меню, сезонні пропозиції, зміни в штаті персоналу). Готові комерційні системи можуть не відповідати специфіці конкретного закладу.

6. Висока залежність від людського чинника

Виконання рутинних операцій вручну збільшує ймовірність помилок, втрати або перекручення даних, особливо при великому потоці клієнтів. Автоматизація дозволяє мінімізувати ці ризики шляхом стандартизації процесів.

Наявність перелічених проблем знижує ефективність роботи ресторану, створює додаткове навантаження на персонал, уповільнює бізнес-процеси та погіршує якість обслуговування клієнтів. Вирішення цих проблем можливе шляхом впровадження сучасної інформаційної системи, яка дозволить автоматизувати ключові процеси та забезпечить централізоване управління всією діяльністю закладу.

У результаті аналізу проблем встановлено, що діяльність ресторану охоплює складну систему взаємодії між персоналом, клієнтами та ресурсами, яка потребує комплексної автоматизації. Предмет дослідження – інформаційна система – має реалізовувати підтримку ключових процесів через доступний та гнучкий інструментарій. Обраний підхід на основі Microsoft Access є доцільним і економічно виправданим для закладу середнього масштабу. Отримані

результати аналізу лягли в основу подальшого проектування інформаційної системи.

1.3. Аналіз існуючих методів(програм) для автоматизації роботи ресторану

Для ефективного проектування інформаційної системи ресторану необхідно провести аналіз сучасних методів та засобів, які використовуються у сфері автоматизації ресторанного бізнесу.

Для вирішення проблем, пов'язаних з управлінням ресторанним бізнесом, на ринку існує низка інформаційних і програмних рішень, орієнтованих на автоматизацію бізнес-процесів. До таких рішень належать спеціалізовані системи управління рестораном (POS-системи), облікові програми, CRM-системи та універсальні офісні засоби, як-от Microsoft Excel або Access. У цьому розділі буде розглянуто основні підходи та засоби, які найчастіше застосовуються, а також їхні переваги й недоліки в контексті потреб малого або середнього ресторану.

1. Класичні (паперові та табличні) методи

У невеликих закладах досі часто використовуються паперові журнали або електронні таблиці (наприклад, Microsoft Excel) для ведення обліку замовлень, списання продуктів та складання звітності. Ці підходи є дешевими та простими у реалізації, але мають низку суттєвих недоліків:

- Низька надійність зберігання даних — висока ймовірність втрати або випадкового видалення інформації;
- Відсутність автоматизації — усі розрахунки та звіти потребують ручного введення;
- Високий ризик помилок через людський фактор;
- Складність масштабування — зі зростанням обсягів даних система стає повільною та неефективною.
- Обмежені можливості автоматизації бізнес-логіки.

2. Готові POS-системи

Існує велика кількість готових програмних рішень (Point of Sale — POS-систем), які надають комплексний функціонал для ресторанів: Poster, R-Keeper, iiko, Poster POS, KeeperCRM, та інші. Такі системи забезпечують високу функціональність, включаючи:

- Управління замовленнями в реальному часі;
- Зручний інтерфейс для персоналу;
- Синхронізацію з кухнею;
- Облік товарних залишків;
- Інтеграція з обладнанням (касові апарати, принтери, термінали тощо);
- Модулі аналітики та CRM.

Однак, незважаючи на переваги, ці рішення мають і недоліки:

- Висока вартість ліцензій або підписок, що є критичним для малих і середніх закладів;
- Обмежена гнучкість — неможливість зміни логіки роботи системи без втручання розробників;
- Залежність від сторонніх постачальників, технічної підтримки та доступу до інтернету (у разі хмарних рішень).
- Необхідність постійного доступу до Інтернету для хмарних рішень.

3. Використання середовищ для побудови власних систем (наприклад, Microsoft Access)

Серед доступних засобів для розробки власних інформаційних систем Microsoft Access є одним із найбільш придатних для швидкого створення баз даних із візуальними формами, запитами та звітами.

Цей інструмент поєднує переваги готових систем та індивідуальних розробок:

- Можливість адаптувати систему до потреб конкретного закладу;
- Простота реалізації та доступність середовища розробки;
- Інтуїтивно зрозумілий інтерфейс для кінцевих користувачів;

- Автоматизація процесів через макроси та мову VBA;
- Швидка розробка прототипу та внесення змін;
- Відсутність потреби в дорогому обладнанні чи підписах.

Обмеження Microsoft Access полягають у:

- Меншій масштабованості порівняно з промисловими СУБД;
- Можливих проблемах із продуктивністю при обробці великих обсягів даних;
- Потреба у початкових знаннях баз даних для створення ефективної структури.

Таблиця 1.1.

Порівняльна оцінка методів

Метод / Інструмент	Вартість	Гнучкість	Масштабованість	Простота впровадження	Автоматизація
Паперовий/Excel облік	Низька	Висока	Низька	Висока	Низька
Готові POS-системи	Висока	Середня	Висока	Середня	Висока
Microsoft Access	Низька	Висока	Середня	Висока	Висока

У результаті критичного аналізу було виявлено, що використання **Microsoft Access** є доцільним і збалансованим рішенням для створення індивідуальної інформаційної системи ресторану, особливо для невеликих та середніх закладів. На відміну від стандартних POS-систем, Access дозволяє враховувати специфіку конкретного бізнесу, змінювати логіку роботи системи та розширювати її функціональність у межах доступних ресурсів.

У цьому контексті доцільним виглядає створення **власної інформаційної системи на базі Microsoft Access**, яка забезпечує достатній рівень функціональності, простоту використання та можливість подальшого розширення. Такий підхід дозволяє зосередитись на автоматизації саме тих процесів, які є критичними для конкретного ресторану, без зайвих витрат і складностей.

1.4. Аналіз програм аналогів

На сучасному ринку існує велика кількість програмного забезпечення, що орієнтоване на автоматизацію процесів у сфері громадського харчування. Серед них найбільш популярними є готові POS-системи, CRM-платформи для ресторанів, а також спеціалізовані програмні комплекси з обліковими функціями. У даному розділі проведено порівняльний аналіз найбільш поширених рішень та визначено їхні сильні і слабкі сторони.

Аналіз популярних програм

1. R-Keer

Одна з найвідоміших комерційних POS-систем, орієнтована на великі та середні заклади.

Переваги:

- Професійна підтримка.
- Широкий функціонал (облік, аналітика, інтеграція з обладнанням).
- Висока продуктивність при великому навантаженні.

Недоліки:

- Висока вартість впровадження та обслуговування.
- Ускладнена модифікація під індивідуальні потреби.
- Залежність від постачальника.

2. Poster POS

Хмарна POS-система для ресторанів малого та середнього бізнесу.

Переваги:

- Інтерфейс, оптимізований під планшети.

- Доступ до даних з будь-якого пристрою.
- Зручна аналітика.

Недоліки:

- Щомісячна абонплата.
- Обмежена офлайн-функціональність.
- Вузька можливість розширення логіки системи.

3. 1С:Ресторан / BAS Ресторан

Багатофункціональні облікові системи, інтегровані з бухгалтерією та складом.

Переваги:

- Повна автоматизація управлінського й фінансового обліку.
- Гнучка система налаштувань.

Недоліки:

- Складність освоєння.
- Потреба в навчанні персоналу.
- Високі вимоги до ресурсоемності.

4. Microsoft Excel / Google Sheets

Використовуються в малих закладах як найпростіший інструмент для обліку.

Переваги:

- Безкоштовність або мінімальні витрати.
- Гнучкість ручного налаштування.

Недоліки:

- Високий ризик помилок.
- Відсутність автоматизації.
- Відсутність багатокористувацької взаємодії.

5. Microsoft Access

Платформа для створення індивідуальних баз даних з інтерфейсами форм і звітів.

Переваги:

- Можливість розробки власної логіки системи.
- Широка підтримка інтерфейсів введення/виведення.
- Інтеграція з іншими продуктами Microsoft.

Недоліки:

- Обмежена продуктивність при великих обсягах даних.
- Відсутність веб-інтерфейсу без додаткової розробки.
- Потреба в мінімальних знаннях розробки.

Таблиця 1.2.

Порівняльна таблиця

Система	Вартість	Автоматизація	Гнучкість	Масштабованість	Залежність від розробника	Підходить для малого ресторану
R-Keeper	Висока	Висока	Низька	Висока	Так	Ні
Poster POS	Середня	Висока	Середня	Середня	Так	Так
1C / BAS Ресторан	Висока	Висока	Середня	Висока	Так	Частково
Excel/Sheets	Низька	Низька	Висока	Низька	Ні	Так, з обмеженнями
Microsoft Access	Низька	Висока	Висока	Середня	Ні	Так

Виявлені недоліки існуючих рішень:

На основі аналізу можна визначити низку типових недоліків, властивих багатьом існуючим рішенням:

- Висока вартість впровадження та обслуговування (комерційні системи).
- Низька адаптивність під індивідуальні потреби малого бізнесу.
- Потреба в зовнішній підтримці або постійному підключенні до мережі (у випадку хмарних рішень).

- Відсутність можливості самостійного модифікування функціональності без залучення розробників.

- Складність освоєння для персоналу без технічної підготовки.

З огляду на виявлені недоліки, доцільним є розроблення власної інформаційної системи, яка:

- буде відповідати конкретним потребам конкретного ресторану;
- не потребуватиме щомісячної плати чи серверної інфраструктури;
- матиме простий у користуванні інтерфейс;
- дозволить легко змінювати структуру інформаційної системи та логіку роботи;
- працюватиме в автономному режимі (без інтернету).

Середовище Microsoft Access оптимально підходить для такої задачі, оскільки дозволяє швидко реалізувати базу даних, створити форми інтерфейсів, обробляти запити, формувати звіти та реалізовувати бізнес-логіку за допомогою VBA без складного налаштування або стороннього програмного забезпечення. У наступних розділах буде розглянуто розробку такої системи на практиці.

РОЗДІЛ 2. ФОРМАЛІЗОВАНА ПОСТАНОВКА ЗАДАЧІ ТА ПОБУДОВА МОДЕЛІ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ РОБОТИ РЕСТОРАНУ

2.1. Формалізована постановка задачі дослідження.

На основі проведеного аналізу предметної області, об'єкта та предмета дослідження, а також виявлених проблем ресторанного бізнесу сформульовано завдання, яке полягає у розробці інформаційної системи для автоматизації основних процесів функціонування ресторану. Для формалізації задачі розглянемо вхідні, вихідні дані, обмеження та цільову функцію.

Метою дослідження є розробка інформаційної системи, що забезпечить автоматизацію основних бізнес-процесів ресторану, підвищення ефективності обслуговування клієнтів та полегшення внутрішнього управління закладом.

Для досягнення мети необхідно вирішити такі завдання:

1. Аналіз предметної області, виявлення ключових об'єктів і процесів ресторанної діяльності.
2. Обґрунтування вибору технологічної платформи та засобів реалізації системи.
3. Побудова логічної моделі інформаційної системи, визначення сутностей, атрибутів і зв'язків.
4. Розробка інтерфейсів взаємодії користувача з системою, форм введення/виведення даних.
5. Автоматизація типових операцій через використання макросів та елементів VBA.
6. Тестування системи, перевірка її працездатності на прикладі умовного ресторану.
7. Оцінка ефективності впровадження з точки зору зручності, надійності та економічної доцільності.

1. Вхідні дані

Інформаційна система повинна працювати з наступними вхідними даними:

- Персонал ресторану — дані про співробітників (ПІБ, посада, дата прийняття, телефон, оклад).
- Меню страв — перелік страв, їхня категорія та вартість.
- Інгредієнти — список найменувань, одиниць виміру.
- Склад — залишки інгредієнтів на складі.
- Склад страви — рецептури страв, що визначають необхідні інгредієнти та їх кількість.
- Замовлення — інформація про обслуговування клієнтів, дата, персонал, стіл.
- Склад замовлення — деталізація замовлення по стравах і кількості.
- Постачання — дані про надходження інгредієнтів від постачальників.
- Склад постачання — дані про кожен інгредієнт у постачанні.

2. Вихідні дані

Система повинна надавати:

- Актуальну інформацію про залишки на складі.
- Списання інгредієнтів у момент оформлення замовлення.
- Додавання інгредієнтів після постачання.
- Автоматичне формування звітів за заданими параметрами (дата, місяць, рік).
- Звіти про використання інгредієнтів, кількість замовлень, популярність страв, ефективність роботи персоналу.

3. Функціональні вимоги

Система має підтримувати такі функції:

- Додавання, редагування та видалення інформації про страви, інгредієнти, замовлення, клієнтів.

- Оформлення замовлення з автоматичним списанням інгредієнтів зі складу.
- Оформлення постачань з автоматичним додаванням інгредієнтів до складу.
- Автоматичне формування звітів на основі даних у базі.
- Зручний пошук інформації через форми (за іменем клієнта, назвою страви, датою тощо).

4. Нефункціональні вимоги

- Простота у використанні для непідготовлених користувачів;
- Надійність та захист від помилок введення;
- Можливість модифікації структури інформаційної системи без повної перебудови системи.

5. Обмеження та припущення

- Усі поля в таблицях мають бути уніфіковані за типами даних.
- Заборонено дублювання записів ключових сутностей (наприклад, інгредієнтів або страв).
- Під час збереження замовлення має бути перевірка на наявність достатньої кількості інгредієнтів на складі.
- Збереження даних має бути забезпечене в реальному часі.
- Всі процеси мають бути доступні через зручний інтерфейс користувача.

6. Очікувані результати

У результаті реалізації системи повинні бути досягнуті такі цілі:

- Скорочення часу на обробку замовлень, постачань та складання звітності.
- Зменшення кількості помилок, пов'язаних із ручним введенням даних.
- Поліпшення контролю над запасами інгредієнтів та продажами.

- Створення централізованої, структурованої та доступної інформаційної системи.

Таким чином, задача створення інформаційної системи для ресторану полягає у побудові програмного продукту, який, оперуючи структурованими даними, забезпечує виконання визначеного переліку функцій з урахуванням заданих обмежень і вимог до зручності, точності та ефективності.

2.2. Опис розробленої інформаційної (аналітичної) моделі

Інформаційна модель — це формалізоване представлення об'єкта дослідження у вигляді сукупності взаємопов'язаних структур даних, які описують сутності предметної області та характер їх взаємодії. У контексті розробки інформаційної системи для ресторану, така модель забезпечує основу для подальшого проєктування інформаційної системи, логіки обробки даних і реалізації користувацьких функцій.

Аналітична складова моделі передбачає не лише зберігання інформації, але й її узагальнення, систематизацію та подальший аналіз (наприклад, оцінку популярності страв, обсягів продажу чи витрат інгредієнтів).

1. Основні сутності інформаційної моделі

У межах проекту виділено такі ключові сутності:

- **Персонал** – відображає працівників ресторану, зберігаючи дані про їхні ПІБ, посаду, дату прийому на роботу, контактний телефон та оклад. Ключовий елемент — ID_Персоналу.
- **Меню (Страви)** – включає перелік страв, які пропонуються відвідувачам. Для кожної страви зберігається назва, категорія (наприклад: перше, друге, десерт) та ціна. Ключ — ID_Страви.
- **Інгредієнти** – перелік продуктів, що використовуються при приготуванні страв. Містить назву інгредієнта та одиницю виміру. Ключ — ID_Інгредієнта.
- **Склад** – актуальні залишки інгредієнтів на складі. Пов'язується з таблицею «Інгредієнти» через зовнішній ключ.

- **Склад страви** – відображає рецептуру, тобто які інгредієнти входять до складу кожної страви та в якій кількості. Є проміжною таблицею між стравами та інгредієнтами.

- **Зала** – містить інформацію про столи в залі (номер, кількість місць), що дозволяє прив'язати замовлення до конкретного столу.

- **Замовлення** – реєструє факти замовлень клієнтів із зазначенням дати, номера столу, відповідального працівника, а також інформації про списання інгредієнтів.

- **Склад замовлення** – деталізує, які саме страви та в якій кількості входили до кожного замовлення.

- **Постачання** – фіксує інформацію про надходження інгредієнтів до складу, зокрема постачальника, дату поставки та факт оновлення залишків.

- **Склад постачання** – деталізує інгредієнти, що були отримані під час постачання, їх кількість та ціну за одиницю.

2. Взаємозв'язки між сутностями

Інформаційна модель передбачає логічні зв'язки між сутностями, які реалізуються через ключі (первинні та зовнішні). Основні зв'язки включають:

- Зв'язок «один-до-багатьох» між персоналом і замовленнями;
- Зв'язок між замовленнями та складом замовлення;
- Зв'язок «багато-до-багатьох» між стравами та інгредієнтами, реалізований через таблицю «Склад страви»;

- Зв'язок між постачаннями та складом постачання;
- Зв'язок інгредієнтів зі складом, складом страви та складом постачання.

3. Діаграма "Сутність-зв'язок" (ER-модель)

4. Призначення інформаційної моделі

Розроблена інформаційна модель виконує такі функції:

- Забезпечує цілісність і несуперечність даних;
- Є основою для побудови реляційної інформаційної системи;
- Дозволяє масштабувати систему за рахунок додавання нових сутностей (наприклад, постачальників або знижкових карт);
- Сприяє зменшенню дублювання інформації та підвищенню продуктивності обробки запитів.

З точки зору аналітики, інформаційна модель дає змогу формувати численні управлінські та фінансові звіти. Основні типи звітів:

- Замовлення/Постачання за день/місяць/рік
- Найпопулярніші страви
- Використання інгредієнтів за день/місяць/рік
- Залишки на складі

- Кількість оброблених замовлень кожним офіціантом за день/місяць/рік

Ці звіти формуються на основі SQL-запитів до реляційної інформаційної системи, що дозволяє керівництву ресторану приймати обґрунтовані рішення щодо закупівель, оптимізації меню, розподілу навантаження на персонал тощо.

Розроблена інформаційна та аналітична модель є основою для реалізації системи автоматизації ресторану. Вона дозволяє забезпечити повноту й цілісність даних, логічну структуру бази, а також підтримку подальшого аналізу й прийняття управлінських рішень. Такий підхід сприяє підвищенню ефективності роботи закладу, зниженню витрат часу персоналу та покращенню обслуговування клієнтів.

2.3. Методи та алгоритми реалізації моделі

У процесі проєктування та реалізації інформаційної системи для ресторану було застосовано такі основні методи:

- **Метод об'єктно-реляційного моделювання**, який дозволив подати інформацію про структуру інформаційної системи у вигляді взаємопов'язаних сутностей (таблиць), з урахуванням зв'язків типу «один до одного», «один до багатьох» та «багато до багатьох»[9; 13];
- **Метод декомпозиції задач**, що дав змогу розділити систему на функціональні підсистеми: облік персоналу, управління меню, обробка замовлень, контроль складу, облік поставок;
- **Метод нормалізації інформаційної системи** — для уникнення надлишковості даних і забезпечення цілісності зв'язків між таблицями[3; 4];
- **Алгоритмічний підхід** для автоматизації подій у системі (списання інгредієнтів, оновлення залишків, формування звітів).

Вибір технологій:

Для реалізації проєкту обрано Microsoft Access з таких причин:

- Наявність вбудованого механізму побудови реляційної інформаційної системи;

- Можливість створення інтерфейсів користувача у вигляді форм;
- Підтримка макросів та подій, які дозволяють реалізовувати бізнес-логіку без використання зовнішніх мов програмування;
- Засоби автоматизованої генерації звітів;
- Простота інтеграції з іншими продуктами Microsoft Office.

Додатково застосовувалися:

- SQL-запити для реалізації обчислень і фільтрацій;
- Вбудовані засоби Access для створення кнопок навігації, форм, підформ і діалогів;
- VBA (за потреби) для гнучкого налаштування подій.

Алгоритми реалізації:

1. Алгоритм списання інгредієнтів при збереженні замовлення:

- При збереженні замовлення програма зчитує перелік страв та їх кількість із таблиці «СкладЗамовлення».
- Для кожної страви визначається відповідний склад інгредієнтів з таблиці «СкладСтрави».
- Розраховується потрібна кількість інгредієнтів (кількість страв $\times$ норма інгредієнта).
- Зі складу (таблиця «Склад») віднімається відповідна кількість.
- Перевіряється, чи достатньо залишків; якщо ні — виводиться попередження.

2. Алгоритм додавання інгредієнтів при збереженні постачання:

- При збереженні нового запису постачання система зчитує дані з таблиці «СкладПостачання».
- Для кожного інгредієнта визначається кількість та ціна.
- До таблиці «Склад» додається відповідна кількість кожного інгредієнта.

3. Алгоритм додавання нового інгредієнта:

- Користувач створює новий запис у таблиці «Інгредієнти».

- Після підтвердження запису автоматично створюється відповідний запис у таблиці «Склад» з початковою кількістю 0.

4. Алгоритм формування звітів:

- Користувач у формі вибирає параметри фільтрації (дата, місяць, рік).
- Запускається запит до відповідних таблиць (Замовлення, СкладЗамовлення, СкладСтрави).
- Формується звіт із підрахунком загальної кількості замовлень, використаних інгредієнтів, вартості тощо.

РОЗДІЛ 3. ОПИС РОЗРОБЛЕНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПІДТРИМКИ РОБОТИ РЕСТОРАНУ

3.1. Проектування таблиць бази даних

Одним із основних етапів створення інформаційної системи є побудова структури бази даних. У межах цього проекту база даних реалізована на платформі Microsoft Access із використанням реляційної моделі.

Під час проектування враховувалися основні функціональні напрями діяльності ресторану: управління персоналом, облік меню, автоматизація обробки замовлень, контроль інгредієнтів та складських залишків, а також облік поставчань. Такий підхід базується на принципах організації баз даних, викладених у роботах [5; 11; 13].

Нижче подано опис і призначення кожної таблиці.

1. Персонал

Зберігає інформацію про працівників ресторану.

Таблиця 3.1.

Поле	Тип даних	Призначення
ID_Персоналу	Автономерація	Унікальний ідентифікатор працівника
ПІБ	Текст	Прізвище, ім'я, по батькові
Посада	Текст	Посада працівника (офіціант, кухар тощо)
Телефон	Текст	Контактний номер телефону
Дата_прийому	Дата/час	Дата прийому на роботу
Оклад	Числовий	Заробітна плата працівника

2. Меню / Страва

Містить перелік страв, що пропонуються клієнтам.

Таблиця 3.2.

Поле	Тип даних	Призначення
ID_Страви	Автономерація	Унікальний ідентифікатор страви
Назва	Текст	Назва страви
Категорія	Текст	Категорія (страва/напій)
Ціна	Грошовий	Вартість страви

3. Інгредієнт

Зберігає інформацію про сировину, що використовується у приготуванні страв.

Таблиця 3.3.

Поле	Тип даних	Призначення
ID_Інгредієнта	Автономерація	Унікальний ідентифікатор інгредієнта
Назва	Текст	Назва інгредієнта
Одиниця_виміру	Текст	Одиниця виміру (кг, л, шт тощо)

4. Склад

Містить поточні залишки інгредієнтів на складі.

Таблиця 3.4.

Поле	Тип даних	Призначення
ID_Складу	Автономерація	Унікальний запис у таблиці
ID_Інгредієнта	Числовий	Зв'язок із таблицею «Інгредієнт»

Кількість	Числовий	Кількість інгредієнта на складі
-----------	----------	---------------------------------

5. Зала

Описує розміщення столів у залі.

Таблиця 3.5.

Поле	Тип даних	Призначення
ID_Столу	Автонумерація	Унікальний ідентифікатор столу
Номер_столу	Текст	Назва або номер столу
Кількість_місць	Числовий	Кількість місць за столом

6. Склад страви

Визначає складові інгредієнти кожної страви.

Таблиця 3.6.

Поле	Тип даних	Призначення
ID_СкладСтрави	Автонумерація	Унікальний запис
ID_Страви	Числовий	Зв'язок із таблицею «Меню»
ID_Інгредієнта	Числовий	Зв'язок із таблицею «Інгредієнт»
Кількість	Числовий	Кількість інгредієнта у страві

7. Заовлення

Містить загальну інформацію про здійснені замовлення.

Таблиця 3.7.

Поле	Тип даних	Призначення
ID_Замовлення	Автономерація	Унікальний ідентифікатор замовлення
Дата_замовлення	Дата/час	Дата та час замовлення
ID_Столу	Числовий	Зв'язок із таблицею «Зала»
ID_Персоналу	Числовий	Зв'язок із таблицею «Персонал»
Інгредієнти_Списано	Так/Ні	Відмітка про списання інгредієнтів

8. Склад замовлення

Описує перелік страв у конкретному замовленні.

Таблиця 3.8.

Поле	Тип даних	Призначення
ID_СкладЗамовлення	Автономерація	Унікальний запис
ID_Замовлення	Числовий	Зв'язок із таблицею «Замовлення»
ID_Страви	Числовий	Зв'язок із таблицею «Меню»
Кількість	Числовий	Кількість одиниць конкретної страви

9. Постачання

Містить інформацію про надходження інгредієнтів.

Таблиця 3.9.

Поле	Тип даних	Призначення
ID_Постачання	Автономерація	Унікальний запис
Дата	Дата/час	Дата постачання
Постачальник	Текст	Назва або ПІБ постачальника
Інгредієнти_Додано	Так/Ні	Відмітка про додавання інгредієнтів до складу

10. Склад постачання

Описує деталі інгредієнтів у кожному постачанні.

Таблиця 3.10.

Поле	Тип даних	Призначення
ID_Запису	Автономерація	Унікальний запис
ID_Постачання	Числовий	Зв'язок із таблицею «Постачання»
ID_Інгредієнта	Числовий	Зв'язок із таблицею «Інгредієнт»
Кількість	Числовий	Кількість отриманого інгредієнта
Ціна_За_Одиницю	Грошовий	Вартість одиниці інгредієнта

Спроектowana структура бази даних дозволяє ефективно зберігати, обробляти та взаємопов'язувати всі необхідні дані для повноцінного функціонування ресторану. Завдяки чітко визначеним зв'язкам між таблицями система забезпечує реляційну цілісність і є основою для побудови запитів, форм, звітів та автоматизованих дій у середовищі Microsoft Access.

3.2. Нормована форма бази даних

Процес нормалізації бази даних є ключовим етапом проектування інформаційної системи [1; 2], що дозволяє забезпечити структурованість, уніфікованість та логічну узгодженість збереження даних. У контексті проекту автоматизації роботи ресторану, первинна нормалізація виконувалася згідно з вимогами Першої нормальної форми.

Перша нормальна форма вимагає:

- Наявність атомарних значень у кожному полі таблиці (тобто значення не поділяються на під-значення);
- Відсутність повторюваних груп даних;
- Унікальність кожного запису, що забезпечується наявністю первинного ключа.

Усі таблиці, що створені для реалізації інформаційної системи ресторану, були спроектовані з дотриманням правил Першої нормальної форми.

Приклад 1:

Таблиця 3.11.

Персонал

ID_ Персоналу	ПІБ	Посада	Телефон	Дата_прийому	Оклад
1	Іваненко О.О.	Офіціант	0971234567	2023-01-15	15000

- Кожне поле містить атомарне значення.
- Відсутні поля, що містять множини або списки значень.
- Використано первинний ключ ID_Персоналу.

Приклад 2:

Таблиця 3.12.

СкладСтрави

ID_СкладСтрави	ID_Страви	ID_Інгредієнта	Кількість
1	5	2	0.3

- Кожен запис відповідає одному інгредієнту конкретної страви.
- У таблиці не зберігаються набори інгредієнтів у вигляді списків — кожен зв'язок зберігається окремо.
- Присутній унікальний ідентифікатор ID_СкладСтрави.

Усі таблиці бази даних, створені для інформаційної системи ресторану, відповідають вимогам першої нормальної форми. Це забезпечує:

- логічну цілісність даних;
- зручність у реалізації запитів та звітів;
- зменшення надлишковості;
- підвищення гнучкості при розширенні структури даних.

Друга нормальна форма бази даних ґрунтується на першій нормальній формі і доповнюється ще однією вимогою:

Таблиця перебуває у Другій нормальній формі, якщо вона вже знаходиться в Першій нормальній формі, і усі неключові атрибути повністю функціонально залежать від усього первинного ключа, а не від його частини (тобто не допускається часткова залежність).

Ця форма особливо актуальна для таблиць з комбінованими (складеними) первинними ключами. Вона дозволяє уникнути дублювання інформації, яке виникає через залежність атрибутів лише від частини ключа.

У розробленій базі даних кілька таблиць містять складені ключі, зокрема:

- СкладСтрави (ID_Страви + ID_Інгредієнта)
- СкладЗамовлення (ID_Замовлення + ID_Страви)
- СкладПостачання (ID_Постачання + ID_Інгредієнта)

Приклад:

Таблиця 3.13.

Склад Страви

ID_СкладСтрави	ID_Страви	ID_Інгредієнта	Кількість
----------------	-----------	----------------	-----------

Усі неключові атрибути (зокрема Кількість) залежать одночасно від обох складових (ID_Страви і ID_Інгредієнта), оскільки інгредієнт має значення лише у контексті певної страви.

Таким чином, часткових залежностей немає, таблиця відповідає Другій нормальній формі.

Приклад:

Таблиця 3.14.

СкладПостачання

ID_Запису	ID_Постачання	ID_Інгредієнта	Кількість	Ціна_За_Одиницю
-----------	---------------	----------------	-----------	-----------------

- Кількість та Ціна_За_Одиницю залежать від пари ID_Постачання і ID_Інгредієнта.

- Немає часткових залежностей: жоден неключовий атрибут не залежить лише від ID_Постачання або ID_Інгредієнта.

Приклад:

Таблиця 3.15.

СкладЗамовлення

ID_СкладЗамовлення	ID_Замовлення	ID_Страви	Кількість
--------------------	---------------	-----------	-----------

- Атрибут Кількість залежить від конкретної пари Замовлення + Страва.

- Відповідно, таблиця також відповідає Другій нормальній формі.

Усі таблиці бази даних, які мають складені ключі, були перевірені на відповідність другій нормальній формі. Завдяки правильному поділу атрибутів та структуруванню даних:

- Уникнуто часткових залежностей;
- Мінімізовано дублювання;
- Забезпечено логічну та структурну цілісність даних.

У процесі проектування та реалізації інформаційної системи для автоматизації роботи ресторану однією з ключових вимог до бази даних є забезпечення логічної узгодженості, несуперечності та відсутності надлишковості збережених даних. Це дозволяє підвищити надійність системи, зменшити обсяг збереженої інформації, уникнути дублювання записів і помилок при обробці даних.

1. Забезпечення несуперечності даних

Несуперечність означає, що дані в базі не конфліктують між собою. Для досягнення цієї мети було реалізовано такі заходи:

- Визначення ключових зв'язків (зовнішніх ключів) між таблицями. Наприклад, поле ID_Персоналу в таблиці Замовлення має посилання на первинний ключ у таблиці Персонал, що унеможливорює появу запису з неіснуючим співробітником.
- Валідація значень полів через типи даних, обов'язковість заповнення та обмеження (NOT NULL, UNIQUE, тощо).
- Автоматичне оновлення або каскадне видалення пов'язаних записів за потреби, що запобігає виникненню "вісячих" або неактуальних посилань.

2. Відсутність надлишковості даних

Надлишковість даних призводить до дублювання інформації, що не лише збільшує обсяг збережених даних, а й створює ризик логічних помилок. У спроектованій базі надлишковість була усунена завдяки:

- Нормалізації структури бази до другої нормальної форми (2NF) — тобто поділу великих таблиць на взаємопов'язані, які зберігають лише необхідну інформацію.

- Чіткому розподілу функціональності між таблицями. Наприклад, інформація про інгредієнти міститься лише в таблиці Інгредієнт, а їх залишки — у таблиці Склад, що дозволяє уникнути дублювання одних і тих самих даних у різних таблицях.

- Використанню підформ і зв'язків "один до багатьох" — що дозволяє виводити пов'язані дані в межах одного інтерфейсу без їх фізичного дублювання.

3. Контроль цілісності даних

Контроль цілісності забезпечується також через:

- Обмеження у макросах та подіях, які автоматично списують або додають інгредієнти до складу при здійсненні замовлення або постачання.

- Запобігання введенню неконсистентної інформації за допомогою форм з перевіркою введення (маски, списки вибору, логічні прапорці).

Усі описані принципи дозволили створити базу даних, яка:

- Містить тільки необхідну інформацію без дублювань;
- Забезпечує логічну узгодженість між даними;
- Уникає суперечностей при додаванні або зміні інформації;
- Підтримує високий рівень структурної та логічної цілісності.

Таким чином, спроектована база даних повністю відповідає вимогам до сучасних інформаційних систем і є надійною основою для ефективного функціонування автоматизованої системи роботи ресторану.

3.3. Опис форм розробленої інформаційної системи

Навігаційна форма – Головна Форма

Головна Форма призначена для навігації між візуальними об'єктами (Формами) інформаційної системи за допомогою кнопок.

Головне меню системи складається з двох вкладок, а саме: “Діяльність” та “Формування звітів”.

На вкладці “Діяльність”(Рис 3.1.) знаходиться вісім кнопок, що при натисканні відкривають форми для введення або коригування даних.

На вкладці “Формування звітів”(Рис 3.2.) знаходиться чотири кнопки, що при натисканні відкривають форми для формування звітів.

Більш детальний опис елементів керування для навігації між формами в подальшому підрозділі 3.4..



Рис. 3.1. Вигляд форми *Головна Форма* вкладки “Діяльність” у режимі форми.

Головна Форма
Інформаційної системи
Підтримки роботи
ресторану

Діяльність **Формування Зітів**

Звіти За Датою

Звіти За Місяцем

Звіти За Роком

Інформаційні
звіти

Запис: 1 з 1 Без фільтра Пошук

Рис. 3.2. Вигляд форми *Головна Форма* вкладка “*Формування Зітів*” у режимі форми.

Форми для введення та редагування даних.

Редагування даних в таблицях інформаційної системи здійснюється за допомогою форм. З цією метою для кожної таблиці розроблено зв’язану, звичайну чи підпорядковану форму. Форми введення даних були реалізовані відповідно до методичних рекомендацій з проектування інтерфейсів у Access, що містяться в літературі [7; 8; 14]. В даному розділі описується лише поля для вводу та редагування даних, інші елементи керування такі як кнопки та полі для пошуку конкретного запису будуть описані в подальшому підрозділі 3.4..

Розглянемо детальніше кожну таку форму.

Для редагування даних таблиці Зала розроблено форму з назвою Зала(Рис 3.3.).

Формат форми – безперервні форми.

Дана форма містить різноманітні елементи керування для ефективного введення даних:

- Номер(Назва) столу вводять у звичайне текстове поле без маски вводу та формату поля, оскільки передбачається внесення все можливих значень;
- Кількість місць столу вводяться у звичайне поле числового формату.

Номер(Назва)_столу	Кількість_місць
21A	2
1A	2
2A	2
3A	2
4A	2

Рис. 3.3. Вигляд форми Зала у режимі форми.

Для редагування даних таблиці Інгредієнт розроблено форму з назвою Інгредієнт(Рис 3.4.).

Формат форми – безперервні форми.

Дана форма містить різноманітні елементи керування для ефективного введення даних:

- Назва інгредієнта вводиться у звичайне текстове поле без маски вводу та формату поля, оскільки передбачається внесення всіх можливих значень;
- Одиниця_виміру може бути однією з наперед визначеного набору, тому й обирається з поля зі списком

Назва	Одиниця_виміру
Яблоко	кг.
Груша	кг.
Олія	л.
Сахар	кг.
Сіль	кг.
Помідор	кг.
Сир	кг.
Молоко	л.
*	

Рис. 3.4. Вигляд форми *Інгредієнт* у режимі форми.

Форма Склад створена лише для перегляду даних оскільки інформаційна система спроектована так що дані в таблиці склад оновлюються автоматично при роботі з іншими формами.

Для перегляду даних таблиці Склад розроблено форму з назвою Склад(Рис 3.5.).

Формат форми – безперервні форми.

Інгредієнт	Залишок	Одиниця
Яблоко	64,8	кг.
Груша	97,4	кг.
Олія	76	л.
Сахар	65,93999	кг.
Сіль	65,96001	кг.
Помідор	65,4	кг.

Рис. 3.5. Вигляд форми *Склад* у режимі форми.

Для редагування даних таблиці Меню розроблено форму з назвою Меню(Рис 3.6.).

Формат форми – безперервні форми.

Дана форма містить різноманітні елементи керування для ефективного введення даних:

- Назва страви\напою вводиться у звичайне текстове поле без маски вводу та формату поля, оскільки передбачається внесення всіх можливих значень;
- Категорія може бути однією з наперед визначеного набору, тому й обирається з поля зі списком
- Ціна вводиться в числове поле з форматування грошової одиниці - гривня..

Меню		
Пошук за назвою		
Салат		
Додати запис		
↺		
📄		
Назва		
Салат		
Категорія	Страва	
Ціна	300,00₴	
Назва		
Помідори з сиром		
Категорія	Страва	
Ціна	150,00₴	
Назва		
Обман		
Категорія	Страва	
Ціна	564,00₴	
* Назва		
Категорія		
Ціна	,00₴	

Рис. 3.6. Вигляд форми *Меню* у режимі форми.

Для редагування даних таблиці Персонал розроблено форму Персонал(Рис 3.7.).

Формат форми – одна форма.

Дана форма містить різноманітні елементи керування для ефективного введення даних:

- ПІБ співробітника вводиться у звичайне текстове поле без маски вводу та формату поля, оскільки передбачається внесення всіх можливих значень;
- Телефон вводиться у звичайне текстове поле з маскою вводу \+000\:00\:0000\:000.
- Посада може бути однією з наперед визначеного набору, тому й обирається з поля зі списком;
- Дата_прийому вводиться в текстове поле формату дата й час, повного формату дати;
- Оклад вводиться в числове поле з форматування грошової одиниці - гривня.

Рис. 3.7. Вигляд форми *Персонал* у режимі форми.

Для редагування даних таблиці СкладСтрави розроблено форму СкладСтрави(Рис 3.8.) з підформою СкладСтрави Підформа.

Формат форми – одна форма.

Дана форма містить різноманітні елементи керування для ефективного введення даних:

- Поле Назва відображає назву страви для якої редагується склад інгредієнтів. Це поле є заблокованим.
- Кількість вводиться у звичайне поле числового формату.
- Інгредієнт може бути однією з наперед визначеного набору, тому й обирається з поля зі списком;
- Одиниця виміру що відображається з права від поля для введення кількості відповідає обраному інгредієнту.

Склад Страви

Додати запис

Скасувати запис

Пошук по назві

Назва

Салат

Склад

Інгредієнт	Кількість	Одиниця
Яблоко	0,200	кг.
Груша	0,100	кг.
Сахар	0,010	кг.

Запис: 1 з 4

Без фільтра

Пошук

Рис. 3.8. Вигляд форми *Склад Страви* у режимі форми.

Для редагування даних таблиць Постачання(Рис 3.9.) й СкладПостачання розроблено форму Постачання що містить підпорядковану форму СкладПостачання Підформа.

Формат форми – одна форма.

Дана форма містить різноманітні елементи керування для ефективного введення даних:

- Дата постачання вводиться в текстове поле формату дата й час, повного формату дати;
- Постачальник вводиться у звичайне текстове поле без маски вводу та формату поля, оскільки передбачається внесення все можливих значень;
- Інгредієнт може бути однією з наперед визначеного набору, тому й обирається з поля зі списком;
- Кількість вводиться у звичайне поле числового формату;
- Одиниця виміру що відображається з права від поля для введення кількості відповідає обраному інгредієнту;
- Ціна за одиницю інгредієнту вводиться в числове поле з форматування грошової одиниці - гривня;
- Ціна пункту відображає загальну ціну поставленого інгредієнту й розраховується автоматично (Ціну за одиницю інгредієнту * Кількість) ;
- Всього відображає загальну ціну всього постачання й розраховується автоматично (Сума(Ціна пункту)) ;
- “Інгредієнти Додано?” - поле типу прапорець. Змінюється автоматично.

Постачання

Додати запис
Скасувати запис

Пошук по даті

Дата Почтавання: 14.05.2025

Постачальник: ООО Андрій

СкладПостачання

Додати запис

Інгредієнт	Олія	Кількість	20	л.	Ціна за одиницю інгредієнту	28,00€	Ціна пункту:	560,00€
Інгредієнт	Помідор	Кількість	24	кг.	Ціна за одиницю інгредієнту	55,00€	Ціна пункту:	1320,00€
Всього:							1880,00€	

Запис: 1 з 2 Без фільтра Пошук

Зберегти Відмінити

Інгредієнти Додано? ☒

Запис: 1 з 1 Без фільтра Пошук

Рис. 3.9. Вигляд форми *Постачання* у режимі форми.

Для редагування даних таблиць *Замовлення* й *СкладЗамовлення* розроблено форму *Замовлення* (Рис 3.10.) що містить підпорядковану форму *СкладЗамовлення Підформа*.

Формат форми – одна форма.

Дана форма містить різноманітні елементи керування для ефективного введення даних:

- Дата_замовлення вводиться в текстове поле формату дата й час, повного формату дати;

- Стіл, Офіціант та Страва може бути однією з наперед визначеного набору, тому й обирається з поля зі списком;
- Кількість вводиться у звичайне поле числового формату;
- Ціна відповідає обраній страві;
- Сума відображає загальну ціну пункту замовлення й розраховується автоматично (Ціна * Кількість) ;
- Вартість всього замовлення відображає загальну ціну всього замовлення й розраховується автоматично (Сума(Сума)) ;
- “Інгредієнти Списано?” - поле типу прапорця. Змінюється автоматично.

Замовлення

Додати запис
Скасувати запис

Пошук за датою

Дата_замовлення: 07.05.2025
Офіціант: Бровчук Володимир Миколайович
Стіл: 1A

СкладЗамовлення

Страва	Кількість	Ціна	Сума
Салат	2	300,00€	600,00€
Вартість всього замовлення			600,00€

Зберегти
Відмінити

Інгредієнти списано? ☐

Рис. 3.10. Вигляд форми *Замовлення* у режимі форми.

Форми для формування звітів.

Форми ФормуванняЗвітівЗаДатою(Рис. 3.11.), ФормуванняЗвітівЗаМісяцями(Рис. 3.12.) та ФормуванняЗвітівЗаРоком(Рис. 3.13.) розроблені для формування звітів за обраним параметром.

Форма ІнформаційніЗвіти(Рис. 3.14.) розроблена для перегляду звітів без параметрів.

ФормуванняЗвітівЗаДатою

Формування Звітів За Обраною Датою

Вкажіть дату для формування звіту 01.06.2025

Звіт Дані Замовлень

Звіт Дані Постачань

Звіт Кількість замовлених страв

Звіт Виконаних Замовлень Офіціантами

Звіт Витрат На Постачання

Звіт Дохід Від Замовлень

Звіт Витрачено Інгредієнтів

Звіт Поставленню Інгредієнтів

Запис: 1 з 1 Без фільтра Пошук

Рис. 3.11. Вигляд форми *ФормуванняЗвітівЗаДатою* у режимі форми.

ФормуванняЗвітівЗаМісяцями

Формування Звітів За Обраним Місяцем

Оберіть місяць для формування звіту: Січень

Звіт Дані Замовлень

Звіт Дані Постачань

Звіт Кількість замовлених страв

Звіт Виконаних Замовлень Офіціантами

Звіт Витрат На Постачання

Звіт Дохід Від Замовлень

Звіт Витрачено Інгредиентів

Звіт Поставленню Інгредиентів

Звіт Прибуток За Місяць

Запис: 1 з 1 | Без фільтра | Пошук

Рис. 3.12. Вигляд форми *ФормуванняЗвітівЗаМісяцями* у режимі форми.

ФормуванняЗвітівЗаРоком

Формування Звітів За Обраним Роком

Введіть рік для формування звіту

Звіт Дані Замовлень

Звіт Дані Постачань

Звіт Кількість замовлених страв

Звіт Виконаних Замовлень Офіціантами

Звіт Витрат На Постачання

Звіт Дохід Від Замовлень

Звіт Витрачено Інгредієнтів

Звіт Поставленню Інгредієнтів

Звіт Прибуток За Рік

Запис: 1 з 1 Без фільтра Пошук

Рис. 3.13. Вигляд форми *ФормуванняЗвітівЗаРоком* у режимі форми.

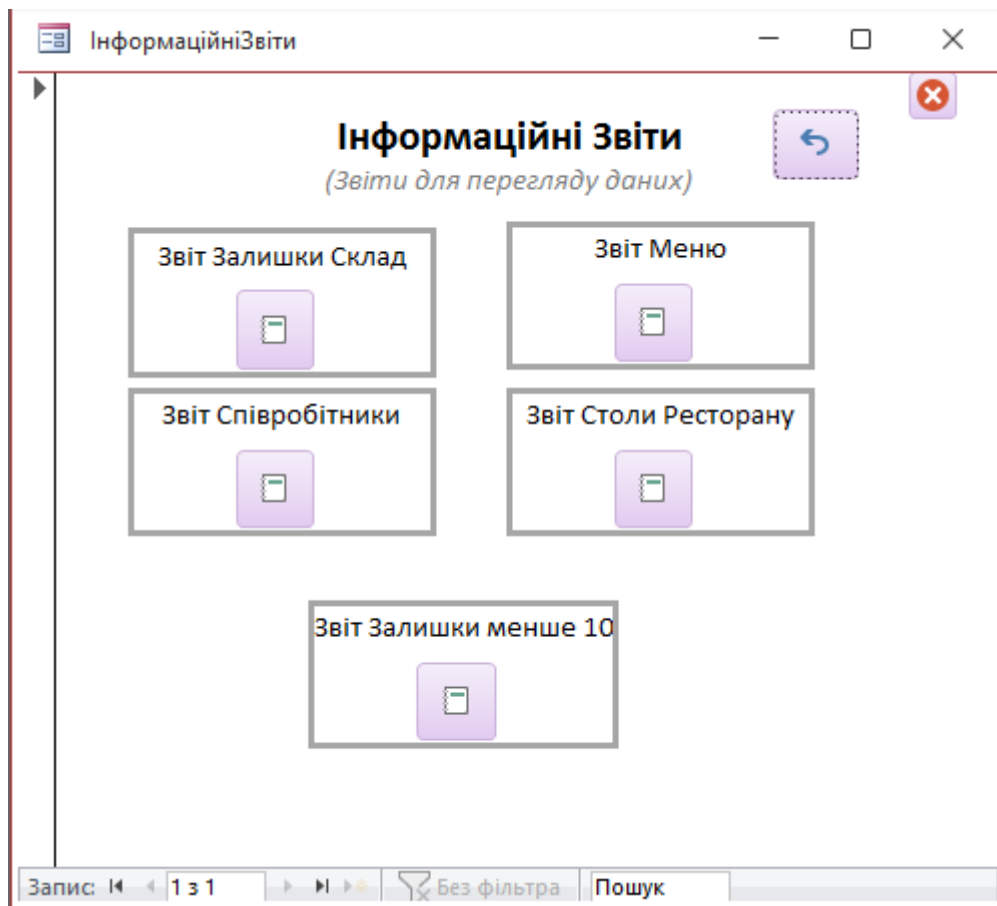


Рис. 3.14. Вигляд форми *ІнформаційніЗвіти* у режимі форми.

3.4. Опис подій та макросів для автоматизації інформаційної системи

Автоматизація процесів в інформаційній системі ресторану на базі Microsoft Access реалізується за допомогою вбудованих подій і макросів, які забезпечують логіку взаємодії між таблицями, формами та користувачем. Такі інструменти дозволяють зменшити кількість рутинних операцій, підвищити точність обліку та забезпечити узгодженість даних у системі.

1. Події, що реалізують бізнес-логіку

1.1. Автоматичне створення запису в таблиці «Склад» при додаванні нового інгредієнта

Опис:

Після створення нового запису у таблиці Інгредієнт система автоматично створює відповідний запис у таблиці Склад, встановлюючи початкову кількість інгредієнта на нуль.

Механізм реалізації:

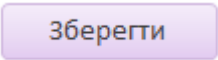
- Подія форми: Після вставлення
- Код VBA – Додаток А.

1.2. Автоматичне списання інгредієнтів зі складу після створення замовлення

Опис:

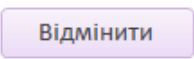
Після збереження замовлення система аналізує усі страви в замовленні, визначає, які інгредієнти входять до їх складу, і автоматично списує відповідну кількість з таблиці Склад.

Механізм реалізації:

- Подія: Кнопка Зберегти 
- Код VBA – Додаток Б
- Алгоритм:
 1. Перевіряється чи значення поля Інгредієнти_Списано цього замовлення дорівнює True.
 2. Якщо **ні** подія продовжує виконуватися. Якщо **так**, подія зупиняється, інгредієнти не списуються зі складу й виводиться повідомлення – "Для цього запису інгредієнти вже списано зі складу".
 3. Для кожної страви з таблиці СкладЗамовлення визнаються інгредієнти з СкладСтрави.
 4. Розраховується загальна кількість витрат кожного інгредієнта.
 5. Зменшується значення поля Кількість у таблиці Склад.
 6. Значення поля Інгредієнти_Списано цього замовлення змінюється на True.

Також реалізовано кнопку для відкату змін кількості інгредієнтів в таблиці Склад, при помилці введення даних замовлення.

Механізм реалізації:

- Подія: Кнопка Відмінити 

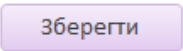
- Код VBA – Додаток В
- Алгоритм:
 1. Перевіряється чи значення поля Інгредієнти_Списано цього замовлення дорівнює False.
 2. Якщо **так** подія продовжує виконуватися. Якщо **ні**, подія зупиняється, інгредієнти не списуються зі складу й виводиться повідомлення – "Для цього запису інгредієнти ще не списано зі складу".
 3. Для кожної страви з таблиці СкладЗамовлення визнаються інгредієнти з СкладСтрави.
 4. Розраховується загальна кількість витрат кожного інгредієнта.
 5. Додається значення поля Кількість у таблиці Склад.
 6. Значення поля Інгредієнти_Списано цього замовлення змінюється на False.

1.3. Автоматичне поповнення складу при створенні постачання

Опис:

При додаванні нового запису в таблицю Постачання, інгредієнти з таблиці СкладПостачання автоматично додаються до складу.

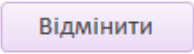
Механізм реалізації:

- Подія: Кнопка Зберегти 
- Код VBA – Додаток Д
- Алгоритм:
 1. Перевіряється чи значення поля Інгредієнти_Додано цього замовлення дорівнює True.
 2. Якщо **ні** подія продовжує виконуватися. Якщо **так**, подія зупиняється, інгредієнти не додаються до складу й виводиться повідомлення – "Для цього запису інгредієнти вже додано до складу".

3. Зі всіх записів з СкладПостачання, пов'язані з поточним постачанням розраховується загальна кількість поставленого інгредієнту.
4. Розрахована кількість додається до поля Кількість у таблиці Склад відповідно до кожного інгредієнту.

Також реалізовано кнопку для відкату змін кількості інгредієнтів в таблиці Склад, при помилці введення даних замовлення.

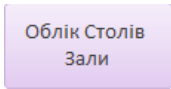
Механізм реалізації:

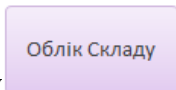
- Подія: Кнопка Відмінити 
- Код VBA – Додаток Ж
- Алгоритм:
 1. Перевіряється чи значення поля Інгредієнти_Додано цього замовлення дорівнює False.
 2. Якщо **так** подія продовжує виконуватися. Якщо **ні**, подія зупиняється, інгредієнти не додаються до складу й виводиться повідомлення – "Для цього запису інгредієнти вже додано до складу".
 3. Зі всіх записів з СкладПостачання, пов'язані з поточним постачанням розраховується загальна кількість поставленого інгредієнту.
 4. Розрахована кількість віднімається від поля Кількість у таблиці Склад відповідно до кожного інгредієнту.

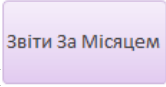
2. Макроси для навігації та керування

2.1. Кнопки переходу між формами

У головній навігаційній формі реалізовано кнопки для відкриття форм:

Облік Столів Зали  – відкриває форму Зала.

Облік Складу  – відкриває форму Склад.

Звіти За Місяцем  – відкриває форму ФормуванняЗвітівЗаМісяцями тощо.

Реалізація:

Макрос Відкритиформу з параметрами назви форми та режиму відкриття (наприклад, "Звичайний").

У всіх формах реалізовано кнопку для повернення до головної навігаційної форми.

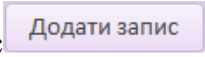
Повернутися до головної форми  – закриває поточну форму та відкриває головну форму.

Реалізація:

Макрос Відкритиформу з параметрами назви форми Головна Форма + Макрос Закритивікно з параметром назви об'єкта "Поточна Форма".

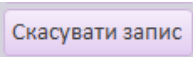
2.2. Кнопки для роботи з записами у формах

Кожна форма редагування даних має кнопки:

Додати запис  – переходить до нового запису.

Видалити запис  – видаляє поточний(виділений) запис.

Деякі форми редагування даних мають такі кнопки:

Скасувати запис  – скасовує додавання нового запису.

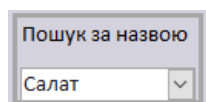
Попередній запис  – переходить до попереднього запису.

Наступний запис  – переходить до наступного запису.

2.3. Елементи керування для пошуку записів у формах

У деяких формах реалізовано елемент керування для прискореного пошуку даних.

Цей елемент містить поле зі списком для пошуку записів за назвою/датою/ПІБ.



Пошук за назвою
Салат

Реалізація:

Ці поля зі списком створено в режимі конструктора форми за допомогою майстра.

2.4. Завершення роботи застосунку

У Формах реалізовано — кнопку «Вихід» , що закриває застосунок.

Реалізація:

Макрос ВийтизAccess.

3. Логіка формування звітів

У формах для формування звітів реалізовано кнопки для відкриття звітів:

Звіт Дані Замовлення  — відкриває звіт Звіт Дані Замовлення За Датою/Місяцем/Роком.

Звіт Витрачено Інгредієнтів  — відкриває звіт Звіт Витрачених Інгредієнтів За Датою/Місяцем/Роком тощо.

Для вказування дати, місяця чи року за який потрібно сформувати звіт у формах для формування звітів реалізовано елемент керування для введення потрібних параметрів:

Поле для обирання дати Вкажіть дату для формування звіту — в формі ФормуванняЗвітівЗаДатою.

Поле зі списком для обирання місяця Оберіть місяць для формування звіту — в формі ФормуванняЗвітівЗаМісяцями.

Поле для вказування року Введіть рік для формування звіту — в формі ФормуванняЗвітівЗаРоком.

Запроваджені події та макроси дозволяють:

- Автоматизувати рутинні дії користувача;
- Забезпечити точність та узгодженість даних;
- Спростити навігацію та керування інформаційною системою;

- Реалізувати базову бізнес-логіку ресторану на рівні подій.

Це суттєво підвищує зручність користування системою, її функціональність та надійність.

3.5. Опис структури звітів інформаційної системи

У рамках створення інформаційної системи для автоматизації роботи ресторану важливим компонентом є **модуль звітності**, який дозволяє отримувати актуальну аналітичну інформацію для управлінських рішень. У системі, реалізованій засобами Microsoft Access, передбачено низку звітів, що формуються автоматично на основі даних, внесених через форми та збережених у таблицях бази даних.

1. Загальна концепція звітності

Звіти створено з метою:

- моніторингу фінансових показників (виручка, витрати),
- контролю залишків інгредієнтів на складі,
- аналізу активності персоналу,
- оцінки популярності страв.

Звіти поділяються на інформаційні (для перегляду поточних даних) та аналітичні (для прийняття управлінських рішень на основі фільтрації за датами/періодами).

2. Типи реалізованих звітів

2.1. Звіти «ІнформаційніЗвіти»

Призначення:

Виведення основної інформації з ключових таблиць у зручному для друку вигляді.

Структура:

Список персоналу з ПІБ, посадою, окладом.

Меню зі стравами, категоріями та цінами.

Інгредієнти з одиницями виміру та залишками на складі.

Столи з номерами(назвами) та кількістю місць.

Приклад:

Залишки інгредієнтів на складі

Назва	Кількість	
Груша	97,4	шт./кг./л.
Молоко	87	шт./кг./л.
Олія	76	шт./кг./л.
Помідор	65,4	шт./кг./л.
Сахар	65,93999	шт./кг./л.
Сир	65,8	шт./кг./л.
Сіль	65,96001	шт./кг./л.
Яблоко	64,8	шт./кг./л.

6 июня 2025 г.

Сторінка 1 з 1

Рис. 3.15. Вигляд звіту *Залишки інгредієнтів на складі* у режимі подання звіту.

Перелік співробітників ресторану

ПІБ	
Бровчук Володимир Миколайович	
Посада	Оклад
Офіціант	6500,00₴

ПІБ	
Матвій Олександр Олегович	
Посада	Оклад
Офіціант	5500,00₴

6 июня 2025 г.

Сторінка 1 з 1

Рис. 3.16. Вигляд звіту *Перелік співробітників ресторану* у режимі подання звіту.

2.2. Звіти «ФормуванняЗвітівЗаДатою»

Призначення:

Формування вибіркового даних за певною датою — наприклад, всі замовлення за конкретний день.

Структура:

Заголовок зі вказаною датою.

Таблиця замовлень із номерами столів, працівниками, стравами, кількість замовленої страви, ціною страви, ціною замовлення.

Підсумок: загальна виручка за день.

Аналогічна структура й в інших звітах що формуються за вказаною датою.

Приклад:

ЗапитФормуванняЗвітівЗаДатою

Звіт Дані Замовлень За Датою

Дата замовлень - 07.05.2025

IDНомер Замовлення	2	Ціна замовлення -	600,00€
Стіл	ПІБ Офіціанта		
1A	Бровчук Володимир Миколайович		
	Назва Страви	Кількість	Ціна Страви
	Салат	2	300,00€

IDНомер Замовлення	9	Ціна замовлення -	300,00€
Стіл	ПІБ Офіціанта		
4A	Матвій Олександр Олегович		
	Назва Страви	Кількість	Ціна Страви
	Салат	1	300,00€
IDНомер Замовлення	11	Ціна замовлення -	1200,00€
Стіл	ПІБ Офіціанта		
1A	Бровчук Володимир Миколайович		
	Назва Страви	Кількість	Ціна Страви
	Салат	4	300,00€

Загальна ціна замовлень за день - 2100,00€

6 юня 2025 г.

Сторінка 1 з 1

Рис. 3.17. Вигляд звіту *Звіт Дані Замовлень За Датою* у режимі подання звіту.

2.3. Звіти «ФормуванняЗвітівЗаМісяцями»

Призначення:

Аналіз ефективності за обраний місяць.

Структура:

Заголовок зі вказаним місяцем.

Таблиця замовлень із номерами столів, працівниками, стравами, кількість замовленої страви, ціною страви, ціною замовлення, датою замовлення.

Підсумок: загальна виручка за місяць.

Аналогічна структура й в інших звітах що формуються за вказаним місяцем.

Приклад:

ЗапитФормуванняЗвітівЗаДатою

Звіт Дані Замовлень За Місяцем

Місяць - Травень

IDНомер Замовлення

2

Ціна замовлення -

600,00€

Стіл

1А

ПІБ Офіціанта

Бровчук Володимир Миколайович

Дата замовлення -

07.05.2025

Назва Страви

Салат

Кількість

2

Ціна Страви

300,00€

IDНомер Замовлення

7

Ціна замовлення -

600,00€

Стіл

1А

ПІБ Офіціанта

Матвій Олександр Олегович

Дата замовлення -

16.05.2025

Назва Страви

Помідори з сиром

Кількість

2

Ціна Страви

150,00€

Назва Страви

Салат

Кількість

1

Ціна Страви

300,00€

IDНомер Замовлення

9

Ціна замовлення -

300,00€

Стіл

4А

ПІБ Офіціанта

Матвій Олександр Олегович

Дата замовлення -

07.05.2025

Назва Страви

Салат

Кількість

1

Ціна Страви

300,00€

IDНомер Замовлення

10

Ціна замовлення -

300,00€

Стіл

3А

ПІБ Офіціанта

Матвій Олександр Олегович

Дата замовлення -

16.05.2025

Назва Страви

Кількість

Ціна Страви

Рис. 3.18. Вигляд звіту *Звіт Дані Замовлень За Місяцем* у режимі подання звіту.

Також в формі *ФормуванняЗвітівЗаМісяцями* створений звіт з підсумками по доходах та витратах за місяць.

Структура:

Заголовок зі вказаним місяцем.

Поля що відображають загальні доходи та витрати: *Дохід з Замовлень*, *Витрачено На Постачання*, *Витрачено На Оклад*.

Підсумок: чистий дохід за місяць.

Приклад:

Звіт Чистий Прибуток За Місяць		
Місяць - Травень		
Дохід з Замовлень	Витрачено На Постачання	Витрачено На Оклад
3000,00€	9749,60€	12000,00€
Всього		-18749,60€

6 июня 2025 г. Сторінка 1 з 1

Рис. 3.19. Вигляд звіту *Звіт Чистий Прибуток За Місяць* у режимі подання звіту.

2.4. Звіти «ФормуванняЗвітівЗаРоком»

Призначення:

Фінансовий і операційний аналіз за рік.

Структура:

Заголовок зі вказаним роком.

Таблиця замовлень із номерами столів, працівниками, стравами, кількість замовленої страви, ціною страви, ціною замовлення, датою замовлення.

Підсумок: загальна виручка за рік.

Аналогічна структура й в інших звітах що формуються за вказаним роком.

Приклад:

ЗапитФормуванняЗвітівЗаДатою

Звіт Дані Замовлень За Роком

Рік - 2025

IDНомер Замовлення	2	Ціна замовлення -	600,00₴
Стіл	ПІБ Офіціанта	Дата замовлення -	07.05.2025
1А	Бровчук Володимир Миколайович		
	Назва Страви	Кількість	Ціна Страви
	Салат	2	300,00₴

IDНомер Замовлення	7	Ціна замовлення -	600,00₴
Стіл	ПІБ Офіціанта	Дата замовлення -	16.05.2025
1А	Матвій Олександр Олегович		
	Назва Страви	Кількість	Ціна Страви
	Помідори з сиром	2	150,00₴
	Назва Страви	Кількість	Ціна Страви
	Салат	1	300,00₴

IDНомер Замовлення	9	Ціна замовлення -	300,00₴
Стіл	ПІБ Офіціанта	Дата замовлення -	07.05.2025
4А	Матвій Олександр Олегович		
	Назва Страви	Кількість	Ціна Страви
	Салат	1	300,00₴

IDНомер Замовлення	10	Ціна замовлення -	300,00₴
Стіл	ПІБ Офіціанта	Дата замовлення -	16.05.2025
3А	Матвій Олександр Олегович		
	Назва Страви	Кількість	Ціна Страви
	Салат	1	300,00₴

IDНомер Замовлення	11	Ціна замовлення -	1200,00₴
Стіл	ПІБ Офіціанта	Дата замовлення -	07.05.2025
1А	Бровчук Володимир Миколайович		
	Назва Страви	Кількість	Ціна Страви
	Салат	4	300,00₴

Загальна ціна замовлень за рік - 3000,00₴

6 юня 2025 г. Сторінка 1 з 1

Рис. 3.20. Вигляд звіту *Звіт Дані Замовлень За Роком* у режимі подання звіту.

Також в формі *ФормуванняЗвітівЗаРоком* створений звіт з підсумками по доходах та витратах за рік.

Структура:

Заголовок зі вказаним роком.

Поля що відображають загальні доходи та витрати: *Дохід з Замовлень*, *Витрачено На Постачання*, *Витрачено На Оклад*.

Підсумок: чистий дохід за місяць.

Приклад:

Звіт Чистий Прибуток За Рік		
Рік - 2025		
Дохід з Замовлень	Витрачено На Постачання	Витрачено На Оклад
3000,00€	9749,60€	144000,00€
Всього		-150749,60€

6 июня 2025 г. Сторінка 1 з 1

Рис. 3.21. Вигляд звіту *Звіт Чистий Прибуток За Рік* у режимі подання звіту.

3. Особливості реалізації звітів у Microsoft Access

Динамічні фільтри: за допомогою форм звіти можуть формуватися на основі введених користувачем параметрів (дата, місяць, рік).

Групування і сортування: записи в звітах згруповано за датами, категоріями, працівниками тощо.

Підсумки: автоматично розраховуються загальні суми, середні значення, кількість записів.

Зручне виведення на друк: усі звіти мають друковану форму із заголовками та стилізованим оформленням.

Система звітів дозволяє адміністрації ресторану ефективно:

- відстежувати фінансову ситуацію,

- контролювати використання ресурсів,
- аналізувати популярність страв,
- оцінювати продуктивність персоналу.

Звіти реалізовані у зрозумілому форматі, що забезпечує легкість користування та підтримує прийняття обґрунтованих управлінських рішень.

3.6. Рекомендації щодо впровадження інформаційної системи та короткий посібник користувача

Впровадження розробленої інформаційної системи в роботу ресторану потребує дотримання певного плану, що забезпечить ефективне, безпечне та безперервне функціонування системи з мінімальним ризиком для основних бізнес-процесів. Рекомендовано дотримуватися наступних кроків:

1.1. Підготовка технічної бази:

- Перевірити наявність персональних комп'ютерів з встановленим Microsoft Access.
- Переконалися у наявності локальної мережі (за потреби) для мульти-користувацького доступу.

1.2. Навчання персоналу:

- Використовуючи короткий посібник користувача. Провести інструктаж для адміністраторів, офіціантів і бухгалтерів щодо користування формами, звітами та навігацією.

1.3. Початкове наповнення бази:

- Ввести всі постійні довідкові дані: меню, працівники, зали, типові інгредієнти.
- Провести тестове заповнення кількох замовлень та поставчань.

1.4. Тестування в робочому середовищі:

- Визначити період пілотного запуску (1–2 тижні).
- Зібрати відгуки персоналу, внести за потреби корективи в інтерфейс або логіку системи.

1.5. Перехід до повноцінної експлуатації:

- Здійснити резервне копіювання.
- Розпочати щоденне використання системи з одночасним моніторингом працездатності.

2. Обґрунтування ефективності впровадження та очікувані результати

Розроблена інформаційна система дозволяє значно підвищити ефективність операційної діяльності ресторану за такими напрямками:

2.1. Оптимізація управління ресурсами

- Контроль складу інгредієнтів у режимі реального часу.
- Автоматичне списання та додавання запасів на підставі операцій замовлення та постачання.

2.2. Підвищення продуктивності персоналу

- Спрощення роботи працівників завдяки інтуїтивному інтерфейсу.
- Мінімізація помилок при введенні інформації.

2.3. Швидкий доступ до звітності

- Автоматичне формування аналітичних звітів за періодами.
- Можливість швидко оцінити динаміку продажів, витрати, прибутковість окремих страв.

2.4. Зниження операційних витрат

- Зменшення потреби у паперових носіях.
- Мінімізація ручної обробки даних.

2.5. Масштабованість та гнучкість

- Система може бути доповнена новими формами, полями, звітами без повної перебудови архітектури.
- За потреби можливе підключення до зовнішніх платформ або перенесення бази даних на SQL-сервер.

2.6. Очікуваний результат

Очікується, що впровадження розробленої системи:

- Скоротить час обслуговування замовлень на 20–30%;
- Зменшить втрати інгредієнтів через неконтрольоване списання;

- Забезпечить більшу прозорість та контроль усіх господарських операцій;
- Покращить якість прийняття управлінських рішень.

Розроблена інформаційна система є оптимальним рішенням для невеликих та середніх закладів громадського харчування, де необхідне поєднання зручності, функціональності та економічної доцільності. Її впровадження дозволить істотно підвищити рівень автоматизації, що є важливим кроком до цифрової трансформації ресторанного бізнесу.

Короткій посібник користувача містить покрокову інструкцію щодо роботи з інформаційною системою ресторану, реалізованою у середовищі Microsoft Access. Вона призначена для персоналу, який здійснює внесення даних, обслуговування клієнтів, облік інгредієнтів та перегляд звітів.

Короткій посібник користувача

1. Запуск програми

- Відкрийте файл бази даних (наприклад, Ресторан.accdb) подвійним кліком.
- При появі повідомлення безпеки натисніть «Увімкнути вміст», щоб активувати всі макроси та події.
- Відкриється Головна форма (Навігаційна форма) з кнопками переходу до інших модулів системи.

2. Робота з формами

Усі основні дії виконуються через спеціальні форми. Нижче наведено призначення кожної форми:

2.1. Форма «Персонал»

- Дозволяє додавати, редагувати або видаляти працівників ресторану.
- Необхідно заповнювати поля: ПІБ, посада, телефон, дата прийому, оклад.

2.2. Форма «Меню»

- Внесення або зміна даних про страви: назва, категорія, ціна.

2.3. Форма «Інгредієнти»

- Додавання інгредієнтів з вказанням одиниці виміру.
- Після додавання автоматично створюється відповідний запис у таблиці «Склад».

2.4. Форма «Склад»

- Відображає наявні залишки інгредієнтів.
- Дані редагуються автоматично при замовленні або постачанні.

2.5. Форма «Зала»

- Реєстрація столів у залі: номер, кількість місць.

2.6. Форма «Замовлення»

- Створення нового замовлення.
- Вибирається стіл, дата, працівник.
- Підформа «СкладЗамовлення» дозволяє додавати страви до замовлення.
- При збереженні замовлення автоматично списуються відповідні інгредієнти зі складу.

2.7. Форма «Склад Страви»

- Визначає, які інгредієнти входять до складу кожної страви.
- Підформа «СкладСтрави» використовується для зручного введення.

2.8. Форма «Постачання»

- Фіксує отримання інгредієнтів від постачальників.
- Підформа «СкладПостачання» дозволяє вказати кількість та ціну.
- При збереженні постачання інгредієнти автоматично додаються на склад.

3. Робота зі звітами

Звіти формуються автоматично на основі наявних даних.

3.1. Форма «ІнформаційніЗвіти»

- Відображення базових таблиць у форматі для перегляду/друку.

3.2. Формування звітів за періодами

- Виберіть відповідну форму:
«За датою»
«За місяцем»
«За роком»
- Введіть необхідний параметр і натисніть кнопку «Сформувати звіт».

4. Макроси та кнопки навігації

- У формах є кнопки «Наступний запис», «Попередній запис», «Додати новий запис», «Скасувати додавання запису», «Зберегти», «Видалити», «Повернутися на головну форму».
- З головної форми доступні кнопки для переходу до будь-якого модуля системи.
- Кнопка «Вихід» дозволяє завершити роботу із застосунком.

5. Резервне копіювання даних

- Рекомендується регулярно створювати резервні копії файлу бази даних (*.accdb), особливо перед змінами структури або оновленням.

6. Технічна підтримка

У разі виникнення помилок або збоїв:

- Перевірте правильність введених даних.
- Упевніться, що всі макроси активовані.
- За потреби зверніться до адміністратора або розробника системи.

Інформаційна система має інтуїтивно зрозумілий інтерфейс і не потребує спеціальної технічної підготовки. Чітке дотримання інструкції забезпечує правильне функціонування системи та коректну обробку всіх операцій ресторану.

ВИСНОВКИ

У процесі виконання бакалаврської роботи було спроектовано та реалізовано інформаційну систему для автоматизації основних бізнес-процесів ресторану. Результатом роботи стала повноцінна база даних, реалізована засобами Microsoft Access, яка охоплює ключові напрямки діяльності закладу: облік персоналу, управління меню, реєстрацію замовлень, контроль складу інгредієнтів і постачання.

Основні досягнення:

- Проведено аналіз предметної області та виявлено основні проблеми, пов'язані з ручним веденням обліку;
- Виконано нормалізацію бази даних, що забезпечило структурованість, узгодженість і масштабованість системи;
- Спроектовано таблиці, зв'язки, форми, макроси та події, які реалізують логіку автоматизованого обліку;
- Реалізовано автоматичне списання та додавання інгредієнтів під час здійснення замовлень і постачань;
- Розроблено інтерактивні звіти для аналізу діяльності закладу за датами, місяцями та роками;
- Забезпечено дружній інтерфейс користувача для ефективної роботи навіть без спеціальних ІТ-знань.

Впровадження такої системи дозволяє зменшити ймовірність помилок, знизити витрати часу на адміністративні процеси та підвищити загальну ефективність роботи ресторану.

Перспективи розвитку:

Незважаючи на досягнутий результат, система має потенціал для подальшого вдосконалення. Серед можливих напрямів розвитку:

- Розгортання клієнт-серверної версії — перехід на Microsoft SQL Server або іншу СУБД з можливістю одночасної роботи з базою декількох користувачів.

- Веб-інтерфейс або мобільний додаток — для зручності користування з мобільних пристроїв, планшетів або з віддаленого доступу.
- Інтеграція з POS-системами — обмін даними з касовим обладнанням для прямого обліку фінансових операцій.
- Автоматичне формування заявок на постачання — на основі аналізу залишків на складі та історії використання інгредієнтів.
- Аналітика та BI (Business Intelligence) — впровадження засобів візуалізації даних і прогнозування продажів.
- Розширення функціоналу — ведення обліку постійних клієнтів, програми лояльності, бронювання столів тощо.

Таким чином, розроблена інформаційна система може стати основою для подальшого вдосконалення цифрової інфраструктури ресторану, сприяючи підвищенню конкурентоспроможності та якості обслуговування клієнтів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Microsoft. *Description of normalization* — опис нормалізації у Microsoft Access: процес організації даних, що забезпечує захист та гнучкість бази, усуваючи надлишковість та неконсистентні залежності. URL: <https://learn.microsoft.com/en-us/office/troubleshoot/access/database-normalization-description>. (дата звернення 05.06.2025)
2. Luke Chung (FMS, Inc.). *Microsoft Access and SQL Server Database Normalization Tips* — огляд головних концепцій нормалізації та типових помилок у проєктуванні баз даних. URL: <https://msaccess.org/tpapers/genaccess/databasenorm.html>. (дата звернення 05.06.2025)
3. SkillForge. *How to Normalize Data in Microsoft Access* — пояснення принципів поділу таблиць на невеликі для кращої гнучкості і керованості. URL: <https://skillforge.com/normalize-data-microsoft-access/>. (дата звернення 05.06.2025)
4. Pressbooks (eCampus Ontario). *Normalization – Relational Databases and Microsoft Access 365* — загальні принципи нормалізації та важливість розробки на 1NF–3NF. URL: <https://ecampusontario.pressbooks.pub/relationaldatabasesandmicrosoftaccess365/part/10-normalization/> (дата звернення 05.06.2025)
5. Морозов С. І., Ковальчук І. В. Інформаційні системи в економіці : навч. посіб. Київ : КНЕУ, 2010. 312 с.
6. Глушаков, С.В. Інформаційні системи і технології в обліку і аудиті : навч. посіб. Київ : Центр учбової літератури, 2020. – 256 с.
7. Жилінський, Б.М. Проєктування інформаційних систем : навч. посіб. Львів : Новий Світ–2000, 2019. – 348 с.
8. Гаврилюк, О.І. Бази даних: проєктування та реалізація : навч. посіб. Вінниця : ВНТУ, 2021. – 144 с.

9. Набоков, А.Ю., Ларін, І.О. Реляційні бази даних: навч. посіб. Одеса : ОНПУ, 2020. – 215 с.
10. Чернишов, В.О. Розробка баз даних у Microsoft Access : навч. посіб. Харків : ХНУ, 2022. – 188 с.
11. Шевчук В.Я., Шевчук Л.С. Реляційні бази даних : навч. посіб. Львів : Новий Світ – 2000, 2019. – 360 с.
12. Крамаренко Г.А., Крамаренко Н.Г. Інформаційні системи і технології : навч. посіб. Київ : Центр учбової літератури, 2020. – 248 с.
13. Слободяник Ю.О., Сафронов М.О. Базы даних: проектування, реалізація, застосування : навч. посіб. Харків : ХНУРЕ, 2020. – 284 с.
14. Панченко І.В., Беляєв І.О. Інформаційні системи у менеджменті : навч. посіб. Київ : КНЕУ, 2021. – 200 с.
15. Бублик С.Г., Грищенко О.А. Microsoft Access: практикум з розробки баз даних. Дніпро : Університет ім. А. Нобеля, 2022. – 148 с.

ДОДАТКИ

Додаток А

VBA код події форми «Інгредієнти»: Після вставлення

```
Private Sub Form_AfterInsert()
```

```
    Dim db As DAO.Database
```

```
    Dim rsСклад As DAO.Recordset
```

```
    Dim ID_Інгредієнт As Long
```

```
    Dim strSQL As String
```

```
    On Error GoTo Помилка
```

```
    Set db = CurrentDb()
```

```
    ID_Інгредієнт = Me.ID_Інгредієнта ' Поточний щойно доданий інгредієнт
```

```
    ' Перевірка, чи вже є запис у таблиці Склад
```

```
    strSQL = "SELECT * FROM Склад WHERE ID_Інгредієнта = " &  
ID_Інгредієнт
```

```
    Set rsСклад = db.OpenRecordset(strSQL, dbOpenDynaset)
```

```
    If rsСклад.EOF Then
```

```
        rsСклад.AddNew
```

```
        rsСклад!ID_Інгредієнта = ID_Інгредієнт
```

```
        rsСклад!Кількість = 0
```

```
        rsСклад.Update
```

```
    End If
```

```
    rsСклад.Close
```

```
    Set rsСклад = Nothing
```

```
    Set db = Nothing
```

```
Exit Sub
```

Помилка:

```
MsgBox "Помилка при додаванні до складу: " & Err.Description, vbCritical  
End Sub
```

Додаток Б

VBA код кнопки «Зберегти» форми «Замовлення»:

```
Private Sub КомандаЗбереження_Click()
```

```
    Dim db As DAO.Database
```

```
    Dim rsЗамовлення As DAO.Recordset
```

```
    Dim rsСкладСтрави As DAO.Recordset
```

```
    Dim rsСклад As DAO.Recordset
```

```
    Dim rsГоловнеЗамовлення As DAO.Recordset
```

```
    Dim strSQL As String
```

```
    Dim ID_Замовлення As Long
```

```
    Dim ID_Страва As Long
```

```
    Dim КількістьСтрав As Integer
```

```
    Dim ID_Інгредієнт As Long
```

```
    Dim КількістьІнгредієнта As Double
```

```
    Dim КількістьНаСкладі As Double
```

```
    On Error GoTo Помилка
```

```
    Set db = CurrentDb()
```

```
    ' Отримуємо ID останнього збереженого замовлення (або використати  
переданий ID)
```

```
    ID_Замовлення = Me.ID_Замовлення
```

```
    strSQL = "SELECT * FROM Замовлення WHERE ID_Замовлення = " &  
ID_Замовлення
```

```
    Set rsГоловнеЗамовлення = db.OpenRecordset(strSQL, dbOpenDynaset)
```

```
If rsГоловнеЗамовлення.EOF Then
```

```
    MsgBox "Постачання не знайдено.", vbExclamation
```

```
    Exit Sub
```

```
End If
```

```
If rsГоловнеЗамовлення!Інгредієнти_Списано = True Then
```

```
    MsgBox "Для цього запису інгредієнти вже списано зі складу.",  
vbInformation
```

```
    rsГоловнеЗамовлення.Close
```

```
    Exit Sub
```

```
End If
```

```
' Вибираємо всі страви з поточного замовлення
```

```
strSQL = "SELECT * FROM СкладЗамовлення WHERE ID_Замовлення = " &  
ID_Замовлення
```

```
Set rsЗамовлення = db.OpenRecordset(strSQL, dbOpenSnapshot)
```

```
Do While Not rsЗамовлення.EOF
```

```
    ID_Страва = rsЗамовлення!ID_Страви
```

```
    КількістьСтрав = rsЗамовлення!Кількість
```

```
' Для кожної страви — отримуємо інгредієнти
```

```
strSQL = "SELECT * FROM СкладСтрави WHERE ID_Страви = " &  
ID_Страва
```

```
Set rsСкладСтрави = db.OpenRecordset(strSQL, dbOpenSnapshot)
```

```
Do While Not rsСкладСтрави.EOF
```

```
    ID_Інгредієнт = rsСкладСтрави!ID_Інгредієнта
```

```
    КількістьІнгредієнта = rsСкладСтрави!Кількість * КількістьСтрав
```

```

' Отримуємо поточну кількість на складі
strSQL = "SELECT * FROM Склад WHERE ID_Інгредієнта = " &
ID_Інгредієнт
Set rsСклад = db.OpenRecordset(strSQL, dbOpenDynaset)

If Not rsСклад.EOF Then
    КількістьНаСкладі = rsСклад!Кількість

    If КількістьНаСкладі >= КількістьІнгредієнта Then
        rsСклад.Edit
        rsСклад!Кількість = КількістьНаСкладі - КількістьІнгредієнта
        rsСклад.Update
    Else
        MsgBox "Недостатньо інгредієнта: " & ID_Інгредієнт, vbExclamation
        Exit Sub
    End If
End If

rsСклад.Close
rsСкладСтрави.MoveNext
Loop

rsСкладСтрави.Close
rsЗамовлення.MoveNext
Loop

rsЗамовлення.Close

rsГоловнеЗамовлення.Edit
rsГоловнеЗамовлення!Інгредієнти_Списано = True

```


rsГоловнеЗамовлення.Update

rsГоловнеЗамовлення.Close

MsgBox "Замовлення збережено, продукти списано зі складу.", vbInformation

Exit Sub

Помилка:

MsgBox "Помилка при списанні: " & Err.Description, vbCritical

End Sub

Додаток В

VBA код кнопки «Відмінити» форми «Замовлення»:

```
Private Sub команда24_Click()
```

```
    Dim db As DAO.Database
```

```
    Dim rsЗамовлення As DAO.Recordset
```

```
    Dim rsСкладСтрави As DAO.Recordset
```

```
    Dim rsСклад As DAO.Recordset
```

```
    Dim rsГоловнеЗамовлення As DAO.Recordset
```

```
    Dim strSQL As String
```

```
    Dim ID_Замовлення As Long
```

```
    Dim ID_Страва As Long
```

```
    Dim КількістьСтрав As Integer
```

```
    Dim ID_Інгредієнт As Long
```

```
    Dim КількістьІнгредієнта As Double
```

```
    Dim КількістьНаСкладі As Double
```

```
    On Error GoTo Помилка
```

```
    Set db = CurrentDb()
```

```
    ' Отримуємо ID останнього збереженого замовлення (або використати  
переданий ID)
```

```
    ID_Замовлення = Me.ID_Замовлення
```

```
    strSQL = "SELECT * FROM Замовлення WHERE ID_Замовлення = " &  
ID_Замовлення
```

```
    Set rsГоловнеЗамовлення = db.OpenRecordset(strSQL, dbOpenDynaset)
```

```
    If rsГоловнеЗамовлення.EOF Then
```

```

MsgBox "Постачання не знайдено.", vbExclamation
Exit Sub
End If

If rsГоловнеЗамовлення!Інгредієнти_Списано = False Then
    MsgBox "Для цього запису інгредієнти ще не списано зі складу.",
vbInformation
    rsГоловнеЗамовлення.Close
    Exit Sub
End If

' Вибираємо всі страви з поточного замовлення
strSQL = "SELECT * FROM СкладЗамовлення WHERE ID_Замовлення = " &
ID_Замовлення
Set rsЗамовлення = db.OpenRecordset(strSQL, dbOpenSnapshot)

Do While Not rsЗамовлення.EOF
    ID_Страва = rsЗамовлення!ID_Страви
    КількістьСтрав = rsЗамовлення!Кількість

    ' Для кожної страви — отримуємо інгредієнти
    strSQL = "SELECT * FROM СкладСтрави WHERE ID_Страви = " &
ID_Страва
    Set rsСкладСтрави = db.OpenRecordset(strSQL, dbOpenSnapshot)

    Do While Not rsСкладСтрави.EOF
        ID_Інгредієнт = rsСкладСтрави!ID_Інгредієнта
        КількістьІнгредієнта = rsСкладСтрави!Кількість * КількістьСтрав

        ' Отримуємо поточну кількість на складі

```

```
strSQL = "SELECT * FROM Склад WHERE ID_Інгредієнта = " &
ID_Інгредієнт
```

```
Set rsСклад = db.OpenRecordset(strSQL, dbOpenDynaset)
```

```
If Not rsСклад.EOF Then
```

```
    КількістьНаСкладі = rsСклад!Кількість
```

```
    If КількістьНаСкладі >= КількістьІнгредієнта Then
```

```
        rsСклад.Edit
```

```
        rsСклад!Кількість = КількістьНаСкладі + КількістьІнгредієнта
```

```
        rsСклад.Update
```

```
    Else
```

```
        MsgBox "Недостатньо інгредієнта: " & ID_Інгредієнт, vbExclamation
```

```
        Exit Sub
```

```
    End If
```

```
End If
```

```
rsСклад.Close
```

```
rsСкладСтрави.MoveNext
```

```
Loop
```

```
rsСкладСтрави.Close
```

```
rsЗамовлення.MoveNext
```

```
Loop
```

```
rsЗамовлення.Close
```

```
rsГоловнеЗамовлення.Edit
```

```
rsГоловнеЗамовлення!Інгредієнти_Списано = False
```

```
rsГоловнеЗамовлення.Update
```

rsГоловнеЗамовлення.Close

MsgBox "Замовлення відмінено, продукти додано до складу.", vbInformation

Exit Sub

Помилка:

MsgBox "Помилка при списанні: " & Err.Description, vbCritical

End Sub

Додаток Д

VBA код кнопки «Зберегти» форми «Постачання»:

```
Private Sub команда19_Click()
```

```
    Dim db As DAO.Database
```

```
    Dim rsПостачання As DAO.Recordset
```

```
    Dim rsСклад As DAO.Recordset
```

```
    Dim rsГоловнеПостачання As DAO.Recordset
```

```
    Dim strSQL As String
```

```
    Dim ID_Постачання As Long
```

```
    Dim ID_Інгредієнт As Long
```

```
    Dim КількістьПоставлено As Double
```

```
    Dim ПоточнаКількість As Double
```

```
    On Error GoTo Помилка
```

```
    Set db = CurrentDb()
```

```
    ' Поточне постачання
```

```
    ID_Постачання = Me.ID_Постачання
```

```
    strSQL = "SELECT * FROM Постачання WHERE ID_Постачання = " &  
ID_Постачання
```

```
    Set rsГоловнеПостачання = db.OpenRecordset(strSQL, dbOpenDynaset)
```

```
    If rsГоловнеПостачання.EOF Then
```

```
        MsgBox "Постачання не знайдено.", vbExclamation
```

```
        Exit Sub
```

```
    End If
```

```

If rsГоловнеПостачання!Інгредієнти_Додано = True Then
    MsgBox "Для цього запису інгредієнти вже додано до складу.",
vbInformation
    rsГоловнеПостачання.Close
    Exit Sub
End If

' Отримуємо всі рядки з таблиці СкладПостачання для цього постачання
strSQL = "SELECT * FROM СкладПостачання WHERE ID_Постачання = " &
ID_Постачання
Set rsПостачання = db.OpenRecordset(strSQL, dbOpenSnapshot)

Do While Not rsПостачання.EOF
    ID_Інгредієнт = rsПостачання!ID_Інгредієнта
    КількістьПоставлено = rsПостачання!Кількість

    ' Отримуємо поточний запис із таблиці Склад
    strSQL = "SELECT * FROM Склад WHERE ID_Інгредієнта = " &
ID_Інгредієнт
    Set rsСклад = db.OpenRecordset(strSQL, dbOpenDynaset)

    If Not rsСклад.EOF Then
        ПоточнаКількість = rsСклад!Кількість
        rsСклад.Edit
        rsСклад!Кількість = ПоточнаКількість + КількістьПоставлено
        rsСклад.Update
    Else
        ' Якщо інгредієнт відсутній — створити новий запис
        rsСклад.AddNew

```

```

        rsСклад!ID_Інгредієнт = ID_Інгредієнт
        rsСклад!Кількість = КількістьПоставлено
        rsСклад.Update
    End If

    rsСклад.Close
    rsПостачання.MoveNext
Loop

rsПостачання.Close

' Оновлюємо прапорець "Інгредієнти_Додано"
rsГоловнеПостачання.Edit
rsГоловнеПостачання!Інгредієнти_Додано = True
rsГоловнеПостачання.Update
rsГоловнеПостачання.Close

MsgBox "Постачання збережено. Інгредієнти додано до складу.",
vbInformation
Exit Sub

Помилка:
MsgBox "Помилка під час оновлення складу: " & Err.Description, vbCritical
End Sub

```


Додаток Ж

VBA код кнопки «Відмінити» форми «Постачання»:

```
Private Sub команда13_Click()
```

```
    Dim db As DAO.Database
```

```
    Dim rsПостачання As DAO.Recordset
```

```
    Dim rsСклад As DAO.Recordset
```

```
    Dim rsГоловнеПостачання As DAO.Recordset
```

```
    Dim strSQL As String
```

```
    Dim ID_Постачання As Long
```

```
    Dim ID_Інгредієнт As Long
```

```
    Dim КількістьПоставлено As Double
```

```
    Dim ПоточнаКількість As Double
```

```
    On Error GoTo Помилка
```

```
    Set db = CurrentDb()
```

```
    ' Поточне постачання
```

```
    ID_Постачання = Me.ID_Постачання
```

```
    strSQL = "SELECT * FROM Постачання WHERE ID_Постачання = " &  
ID_Постачання
```

```
    Set rsГоловнеПостачання = db.OpenRecordset(strSQL, dbOpenDynaset)
```

```
    If rsГоловнеПостачання.EOF Then
```

```
        MsgBox "Постачання не знайдено.", vbExclamation
```

```
        Exit Sub
```

```
    End If
```

```

If rsГоловнеПостачання!Інгредієнти_Додано = False Then
    MsgBox "Для цього запису інгредієнти ще не додано до складу.",
vbInformation
    rsГоловнеПостачання.Close
    Exit Sub
End If

' Отримуємо всі рядки з таблиці СкладПостачання для цього постачання
strSQL = "SELECT * FROM СкладПостачання WHERE ID_Постачання = " &
ID_Постачання
Set rsПостачання = db.OpenRecordset(strSQL, dbOpenSnapshot)

Do While Not rsПостачання.EOF
    ID_Інгредієнт = rsПостачання!ID_Інгредієнта
    КількістьПоставлено = rsПостачання!Кількість

    ' Отримуємо поточний запис із таблиці Склад
    strSQL = "SELECT * FROM Склад WHERE ID_Інгредієнта = " &
ID_Інгредієнт
    Set rsСклад = db.OpenRecordset(strSQL, dbOpenDynaset)

    If Not rsСклад.EOF Then
        ПоточнаКількість = rsСклад!Кількість
        rsСклад.Edit
        rsСклад!Кількість = ПоточнаКількість - КількістьПоставлено
        rsСклад.Update
    Else
        ' Якщо інгредієнт відсутній — створити новий запис
        rsСклад.AddNew
        rsСклад!ID_Інгредієнт = ID_Інгредієнт
    End If
End Do

```

```

        rsСклад!Кількість = КількістьПоставлено
        rsСклад.Update
    End If

    rsСклад.Close
    rsПостачання.MoveNext
Loop

rsПостачання.Close

' Оновлюємо прапорець "Інгредієнти_Додано"
rsГоловнеПостачання.Edit
rsГоловнеПостачання!Інгредієнти_Додано = False
rsГоловнеПостачання.Update
rsГоловнеПостачання.Close

MsgBox "Постачання відмінено. Інгредієнти списано зі складу.",
vbInformation
Exit Sub

Помилка:
MsgBox "Помилка під час оновлення складу: " & Err.Description, vbCritical
End Sub

```