

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) автоматизації і комп'ютерних систем імені проф. І.В. Ельперіна

Кафедра інформаційних технологій, штучного інтелекту і кібербезпеки

Освітній ступінь бакалавр

Спеціальність 122 Комп'ютерні науки

(код і назва)

Освітньо-професійна програма Комп'ютерні науки

(назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри інформаційних
технологій, штучного інтелекту і
кібербезпеки Сергій ГРИБКОВ

«05» листопада 2025 року

З А В Д А Н Н Я

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Крушельницької Дарини Геннадіївни

(прізвище, ім'я, по батькові)

1. Тема роботи «Розроблення інформаційної системи для управління виконання замовленнями ТОВ «БОДО УКРАЇНА»»

керівник роботи Мазуренко Ольга Олександрівна, старший викладач, кандидат технічних наук.

затверджені наказом закладу вищої освіти від «05» листопада 2025 р. № 907-кв

2. Строк подання здобувачем роботи: 02.02.2026 р.

3. Вихідні дані до роботи: Загальна інформація про компанію, Налагоджений документообіг фірми, Дані про працівників та організаційну структуру підприємства

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

1. Загальна характеристика компанії

2. Структура організації

3. Опис поточних бізнес-процесів

4. Виявлені проблеми

5. Огляд існуючих рішень

6. Задачі на автоматизацію

7. Економічний ефект від впровадження нової системи

8. Технічне завдання

9. Опис функцій системи

10. Проектування бази даних

11. Реалізація описаних функцій

5. Перелік графічного матеріалу:

Моделі бази даних

Фрагменти коду інформаційної системи

6. Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	ст. викл. Мазуренко О.О.		
2	ст. викл. Мазуренко О.О.		
3	ст. викл. Мазуренко О.О.		

7. Дата видачі завдання: 05 листопада 2025 року**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Вибір теми бакалаврської кваліфікаційної роботи	листопад 2025	Виконано
2	Затвердження теми та наукового керівника наказом	листопад 2025	Виконано
3	Розроблення та затвердження індивідуального завдання	листопад 2025	Виконано
4	Підбір і опрацювання літературних джерел	листопад 2025	Виконано
5	Написання теоретичного розділу (Розділ 1)	листопад 2025	Виконано
6	Подання та погодження Розділу 1 з керівником	листопад 2025	Виконано
7	Виконання аналітичної / технологічної частини (Розділ 2)	листопад – грудень 2025	Виконано
8	Проведення розрахунків, аналіз результатів	грудень 2025	Виконано
9	Написання практичної / проєктної частини (Розділ 3)	січень 2026	Виконано
10	Формування загальних висновків	січень 2026	Виконано
11	Оформлення списку використаних джерел та додатків	січень 2026	Виконано
12	Підготовка доповіді та презентації до захисту	січень 2026	Виконано

Здобувач

Дарина КРУШЕЛЬНИЦЬКА

Керівник роботи

Ольга МАЗУРЕНКО

АНОТАЦІЯ

У кваліфікаційній роботі розглянуто питання проектування та реалізації інформаційної системи управління замовленнями для ТОВ «БОДО Україна». Актуальність теми зумовлена необхідністю автоматизації процесів обробки замовлень, обліку оплат та формування аналітичної звітності в умовах цифровізації бізнес-процесів.

У роботі проведено аналіз предметної області та існуючих підходів до автоматизації управління замовленнями. Побудовано функціональну модель бізнес-процесу з використанням методології IDEF0, розроблено логічну модель бази даних та ER-діаграму, виконано нормалізацію структури даних.

Розроблено вебзастосунок на основі клієнт-серверної архітектури з використанням мови програмування Python та вебфреймворку Flask. Реалізовано модулі авторизації користувачів, управління замовленнями, обліку оплат та формування аналітичних звітів. Забезпечено розмежування прав доступу відповідно до ролей користувачів. Проведено тестування програмного продукту.

Практичне значення роботи полягає у створенні програмного рішення, що може бути використане для автоматизації процесу управління замовленнями підприємства.

Ключові слова: ІНФОРМАЦІЙНА СИСТЕМА, УПРАВЛІННЯ ЗАМОВЛЕННЯМИ, КЛІЄНТ-СЕРВЕРНА АРХІТЕКТУРА, БАЗА ДАНИХ, IDEF0, UML, FLASK.

SUMMARY

This qualification thesis is devoted to the design and implementation of an information system for order management for LLC “BODO Ukraine”. The relevance of the research is determined by the need to automate order processing, payment accounting, and analytical reporting within the context of digital transformation of business processes.

The thesis includes an analysis of the subject area and the existing approaches to order management automation. A functional model of the business process was developed using the IDEF0 methodology. A logical database model and an entity-relationship (ER) diagram were designed, and the database structure was normalized to the third normal form.

The information system was implemented as a web application based on a client-server architecture using Python and the Flask web framework. The system provides user authentication and role-based access control, order creation and management, payment tracking, and analytical report generation. Data integrity is ensured through relational database mechanisms, including primary and foreign keys and database triggers.

Functional testing of the developed software product was performed to verify the correctness of system operations and compliance with specified requirements.

The practical significance of the thesis lies in the development of a software solution that can be used to automate order management processes and improve operational efficiency in the enterprise.

Keywords: INFORMATION SYSTEM, ORDER MANAGEMENT, CLIENT-SERVER ARCHITECTURE, DATABASE, IDEF0, UML, FLASK.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ.....	9
1.1. Загальна характеристика підприємства ТОВ «БОДО Україна»	9
1.2. Аналіз предметної області управління замовленнями.....	10
1.3. Аналіз існуючих інформаційних систем та обґрунтування вибору підходу	13
1.4. Виявлені проблеми організації процесу управління замовленнями.....	15
1.5. Обґрунтування необхідності розроблення інформаційної системи	15
1.6. Розрахунок техніко-економічного обґрунтування впровадження створюваного програмного забезпечення	16
1.7. Постановка задачі кваліфікаційної роботи.....	19
1.8. Висновки до розділу 1	19
РОЗДІЛ 2. ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ЗАМОВЛЕННЯМИ.....	21
2.1. Функціональне моделювання процесу управління замовленнями.....	21
2.2. Проєктування логічної моделі бази даних	22
2.3. Аналіз зв'язків між сутностями.....	24
2.4. Нормалізація структури бази даних.....	24
2.5. Архітектура інформаційної системи	25
2.6. Формалізація вимог до інформаційної системи.....	26
2.7. Модель прецедентів (Use Case Model).....	30
2.8. UML-моделювання інформаційної системи.....	33
2.9. Висновки до розділу 2	33
РОЗДІЛ 3. РОЗРОБЛЕННЯ ТА РЕАЛІЗАЦІЯ ПРОГРАМНОГО ПРОДУКТУ .	35
3.1. Загальна характеристика програмної реалізації	35
3.2. Архітектура програмного продукту	35
3.3. Структура програмного проєкту	36
3.4. Реалізація основних функціональних модулів.....	37

	7
3.5. Вимоги до програмного забезпечення та технічного середовища	38
3.6. Тестування програмного продукту	39
3.7. Забезпечення надійності та безпеки.....	39
3.8. Інструкція користувача інформаційної системи управління замовленнями	40
3.9. Висновки до розділу 3	44
ВИСНОВКИ.....	45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	47
ДОДАТКИ.....	51

ВСТУП

В умовах цифровізації бізнес-процесів автоматизація управління замовленнями є важливою складовою підвищення ефективності діяльності підприємств. Особливо актуальною є автоматизація для компаній, діяльність яких пов'язана з великою кількістю клієнтів, партнерів та фінансових операцій [11–15,34,36].

ТОВ «БОДО Україна» здійснює діяльність у сфері реалізації подарункових сертифікатів та організації послуг-вражень. Процес управління замовленнями включає приймання замовлень, обробку даних клієнтів, облік оплат, взаємодію з партнерами та формування звітності. За відсутності спеціалізованої інформаційної системи обробка замовлень може супроводжуватися дублюванням інформації, затримками та обмеженими можливостями аналітики.

Актуальність теми зумовлена необхідністю розроблення інформаційної системи, що забезпечить централізацію даних, автоматизацію основних операцій та підтримку прийняття управлінських рішень [7-8].

Метою роботи є розроблення інформаційної системи управління замовленнями для ТОВ «БОДО Україна».

Для досягнення поставленої мети необхідно:

1. Провести аналіз предметної області.
2. Побудувати функціональну модель бізнес-процесу.
3. Розробити логічну модель бази даних.
4. Реалізувати вебзастосунок із використанням клієнт-серверної архітектури.
5. Забезпечити механізм розмежування доступу користувачів.
6. Провести тестування програмного продукту.
7. Розробити інструкцію користувача.

Практичне значення роботи полягає у створенні програмного продукту, який може бути використаний для автоматизації процесу управління замовленнями підприємства.

Робота складається зі вступу, трьох розділів, висновків, списку використаних джерел та додатків.

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1. Загальна характеристика підприємства ТОВ «БОДО Україна»

ТОВ «БОДО Україна» є підприємством, що здійснює діяльність у сфері надання послуг та реалізації подарункових сертифікатів на враження. Компанія працює на ринку України, співпрацюючи з партнерами у різних містах та пропонуючи клієнтам широкий перелік послуг розважального, туристичного, спортивного та культурного характеру.

Основними напрямками діяльності підприємства є:

- продаж подарункових сертифікатів;
- приймання та обробка замовлень;
- координація надання послуг партнерами;
- обслуговування клієнтів;
- супровід оплат та контроль виконання замовлень.

Компанія використовує багатоканальну модель взаємодії з клієнтами:

- офіційний вебсайт;
- телефонні звернення;
- месенджери;
- фізичні точки продажу.

Головний офіс підприємства розташований у місті Київ, а партнерська мережа охоплює великі міста України. Надання послуг здійснюється безпосередньо партнерами компанії, які виконують конкретні замовлення клієнтів.

Організаційна структура підприємства включає:

- відділ продажу;
- службу підтримки клієнтів;
- відділ роботи з партнерами;
- адміністративний персонал;
- ІТ-підтримку.

Кожен підрозділ бере участь у життєвому циклі замовлення – від його створення до завершення та закриття.

Організаційна структура підприємства визначає розподіл функціональних обов'язків між підрозділами та їх участь у процесі управління замовленнями. На рисунку 1.1 наведено організаційну структуру ТОВ «БОДО Україна».

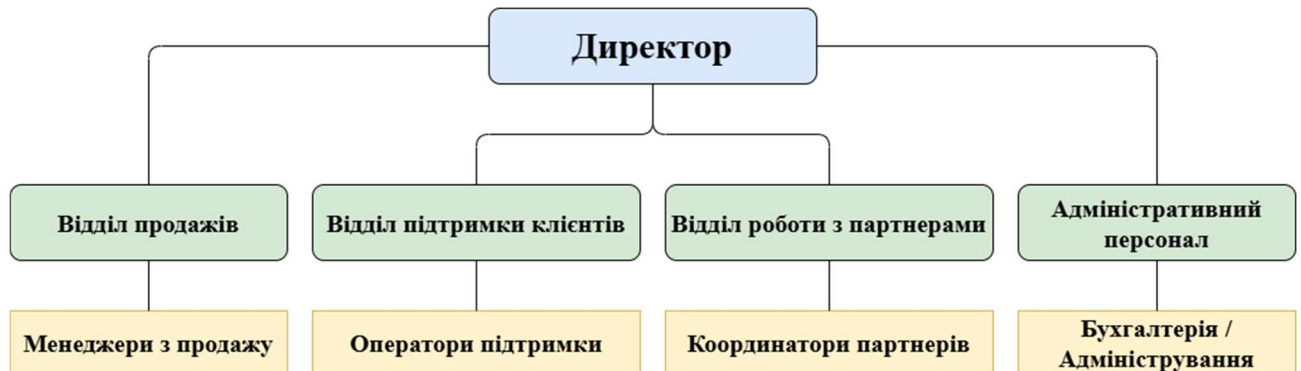


Рисунок 1.1 – Організаційна структура ТОВ «БОДО Україна»

Як видно зі схеми, ключову роль у процесі управління замовленнями відіграють відділ продажу, служба підтримки клієнтів та відділ роботи з партнерами.

1.2. Аналіз предметної області управління замовленнями

Діяльність ТОВ «БОДО Україна» пов'язана з наданням послуг у сфері подарункових сертифікатів та організації вражень. Основою бізнес-процесів підприємства є формування, обробка та контроль виконання замовлень клієнтів. Саме процес управління замовленнями є центральним елементом функціонування компанії та визначає ефективність її операційної діяльності.

1.2.1. Загальна характеристика процесу управління замовленнями

Процес управління замовленнями включає комплекс взаємопов'язаних дій, спрямованих на:

- реєстрацію замовлення клієнта;
- вибір послуги або подарункового сертифіката;
- формування вартості;

- облік оплат;
- контроль виконання послуги;
- зміну статусу замовлення;
- завершення та архівування замовлення.

Замовлення в межах даної роботи розглядається як факт придбання клієнтом конкретної послуги або сертифіката, що супроводжується фінансовою операцією та подальшим контролем виконання.

Ключовими учасниками процесу є:

- клієнт;
- менеджер з продажу;
- оператор служби підтримки;
- адміністратор системи;
- партнери (виконавці послуг).

1.2.2. Життєвий цикл замовлення

Життєвий цикл замовлення можна подати у вигляді послідовності етапів:

- створення замовлення.

Менеджер реєструє замовлення у системі із зазначенням клієнта та послуги.

- Очікування оплати

Замовлення має статус «Нове» до моменту внесення оплати.

- Фіксація оплати

Після додавання оплати система автоматично перераховує загальну суму та змінює статус (за потреби).

- Виконання послуги

Замовлення переходить у статус «В роботі».

- Завершення виконання

Після фактичного надання послуги замовлення отримує статус «Виконане».

- Закриття замовлення

Остаточне завершення операцій та архівування.

Таким чином, статус є важливим елементом контролю та відображає поточний стан замовлення.

1.2.3. Інформаційні потоки підприємства

В процесі управління замовленнями формуються такі інформаційні потоки:

- від клієнта → до менеджера (заявка на послугу);
- від менеджера → до системи (введення даних замовлення);
- від системи → до служби підтримки (контроль статусів);
- від клієнта → до бухгалтерії (оплата);
- від системи → до керівництва (звіти).

Ці потоки повинні бути синхронізованими та централізованими, що забезпечується інформаційною системою.

1.2.4. Основні проблеми предметної області

У процесі аналізу предметної області було визначено низку проблем, характерних для систем управління замовленнями середнього бізнесу:

- дублювання інформації про клієнтів;
- помилки ручного введення;
- відсутність централізованої аналітики;
- складність контролю оплат;
- відсутність єдиного механізму формування звітності;
- затримки в оновленні статусів замовлень;
- складність аналізу ефективності послуг.

1.2.5. Ризики при відсутності автоматизації

За відсутності автоматизованої системи можливі:

- фінансові втрати через помилки в розрахунках;
- невчасне оновлення статусів;
- складність формування управлінських звітів;
- зниження рівня обслуговування клієнтів;
- ускладнення масштабування бізнесу.

1.2.6. Вимоги до інформаційної системи

На основі аналізу предметної області сформовано базові вимоги до інформаційної системи:

1. Централізоване зберігання даних.
2. Розмежування прав доступу.
3. Автоматичний перерахунок оплат.
4. Контроль статусів замовлень.
5. Формування аналітичних звітів.
6. Можливість масштабування.
7. Зручний веб-інтерфейс.

1.2.7. Обґрунтування необхідності розробки системи

З урахуванням виявлених проблем та вимог до автоматизації доцільним є розроблення спеціалізованої інформаційної системи управління замовленнями, адаптованої до специфіки діяльності ТОВ «БОДО Україна».

- Розробка власної системи дозволяє:
- мінімізувати витрати на впровадження;
- забезпечити гнучкість налаштувань;
- інтегрувати специфічну бізнес-логіку;
- отримати повний контроль над даними.

1.3. Аналіз існуючих інформаційних систем та обґрунтування вибору підходу

На сучасному ринку представлено значну кількість інформаційних систем, призначених для автоматизації процесів управління замовленнями та взаємодії з клієнтами.

До найбільш поширених рішень належать:

- CRM-системи (Customer Relationship Management) [11, 12];
- ERP-системи (Enterprise Resource Planning) [12];
- SaaS-платформи електронної комерції;
- галузеві спеціалізовані рішення.

Порівняльний аналіз типових рішень.

CRM-системи.

Переваги:

- готова функціональність;
- інтеграція з каналами комунікації;
- можливість аналітики.

Недоліки:

- висока вартість ліцензій;
- надлишковий функціонал;
- складність кастомізації.

ERP-системи

- Переваги:
- комплексна автоматизація;
- інтеграція фінансових процесів.

Недоліки:

- складність впровадження;
- значні фінансові витрати;
- надмірність для малого та середнього бізнесу.

SaaS-рішення

Переваги:

- швидке розгортання;
- відсутність необхідності локальної інфраструктури.

Недоліки:

- залежність від стороннього провайдера;
- обмеження налаштувань;
- ризики безпеки даних.

Обґрунтування розроблення власної системи.

З урахуванням специфіки діяльності ТОВ «БОДО Україна» доцільним є створення спеціалізованої інформаційної системи, що:

- повністю відповідає внутрішнім бізнес-процесам;
- не містить надлишкового функціоналу;

- забезпечує гнучкість масштабування;
- дозволяє адаптувати систему під майбутні зміни.
- Власна розробка дозволяє:
- контролювати архітектуру системи;
- забезпечити відповідність вимогам безпеки;
- мінімізувати витрати на ліцензування;
- реалізувати необхідний аналітичний функціонал.

У результаті проведеного аналізу встановлено, що використання типових рішень не забезпечує повної відповідності потребам підприємства.

Отже, прийнято рішення про розроблення власної інформаційної системи управління замовленнями із використанням сучасних веб-технологій.

1.4. Виявлені проблеми організації процесу управління замовленнями

У результаті аналізу предметної області було визначено низку проблем, що виникають при обробці замовлень:

- фрагментарність використання програмних інструментів;
- відсутність єдиного інтегрованого середовища управління замовленнями;
- дублювання інформації у різних джерелах;
- складність контролю актуального статусу замовлення;
- обмежені можливості аналітичної обробки даних;
- ризик помилок при ручному введенні інформації.

Використання окремих інструментів (табличних процесорів, локальних програм або вебсервісів) не забезпечує повної автоматизації бізнес-процесу.

Це зумовлює необхідність створення єдиної інформаційної системи.

1.5. Обґрунтування необхідності розроблення інформаційної системи

Автоматизація процесу управління замовленнями дозволить:

- забезпечити централізоване зберігання даних;
- скоротити час обробки замовлень;
- зменшити кількість помилок;
- забезпечити прозорість контролю виконання;

- підвищити якість обслуговування клієнтів;
- реалізувати аналітичну звітність у розрізі періодів і статусів.

Впровадження спеціалізованої інформаційної системи сприятиме підвищенню ефективності управлінських рішень та оптимізації роботи підприємства.

1.6. Розрахунок техніко-економічного обґрунтування впровадження створюваного програмного забезпечення

Враховуючи масштаби діяльності ТОВ «БОДО Україна», яке здійснює онлайн-та офлайн-продаж подарункових сертифікатів по всій території України, автоматизація процесу управління замовленнями має безпосередній вплив на операційну ефективність підприємства.

1.6.1. Визначення витрат на розроблення

Розроблення системи здійснюється власними силами підприємства. Загальні витрати на створення програмного продукту визначаються за формулою:

$$V_r = V_{op} + V_{am} + V_{eksp} \quad (1.1)$$

де:

V_r — загальні витрати на розроблення, грн;

V_{op} — витрати на оплату праці розробника, грн;

V_{am} — витрати на амортизацію обладнання, грн;

V_{eksp} — експлуатаційні витрати (електроенергія, інтернет, тестування), грн.

Розрахунок витрат наведено в таблиці 1.1.

Таблиця 1.1 – Розрахунок витрат на розроблення програмного забезпечення

Стаття витрат	Сума, грн
Оплата праці розробника (2 міс × 30 000 грн)	60 000
Амортизація техніки та ПЗ	3 000
Електроенергія та інтернет	2 000
Тестування та впровадження	5 000
Разом	70 000

Отже, відповідно до формули (1.1), загальні витрати на створення програмного забезпечення становлять:

$$V_p = 70000 \text{ (грн)} \quad (1.2)$$

1.6.2. Розрахунок економічного ефекту

Середній місячний фонд оплати праці відділу продажів становить 180 000 грн. Очікуване скорочення трудових витрат після автоматизації процесів становить 25%.

Місячний економічний ефект визначається за формулою:

$$E_m = \Phi \cdot k \quad (1.3)$$

де:

E_m — місячна економія, грн;

Φ — фонд оплати праці, грн;

k — коефіцієнт скорочення витрат (0,25).

Підставляючи значення у формулу (1.3), отримуємо:

$$E_m = 180000 \times 0,25 = 45000 \text{ (грн)} \quad (1.4)$$

Річний економічний ефект визначається за формулою:

$$E_p = E_m \times 12 \quad (1.5)$$

$$E_p = 45000 \times 12 = 540000 \text{ (грн)} \quad (1.6)$$

1.6.3. Розрахунок строку окупності

Строк окупності визначається як відношення витрат на розроблення до місячного економічного ефекту:

$$T = V_p / E_m \quad (1.7)$$

Підставляючи значення з формул (1.2) та (1.4), отримуємо:

$$T = \frac{70000}{45000} \approx 1,55 \text{ місяця} \quad (1.8)$$

Отже, строк окупності впровадження системи становить приблизно 1,5–2 місяці.

1.6.4. Розрахунок рентабельності інвестицій

Рентабельність інвестицій (ROI) визначається за формулою:

$$ROI = E_p - V_p / V_p \times 100\% \quad (1.9)$$

Підставляючи значення з формул (1.2) та (1.6), отримуємо:

$$ROI = 540\,000 - 70\,000 / 70\,000 \times 100\% \approx 671\% \quad (1.10)$$

Отримане значення свідчить про високу економічну ефективність проєкту.

1.6.5. Порівняння з альтернативними рішеннями

Вартість готових CRM-рішень для підприємств масштабу ТОВ «БОДО Україна» становить у середньому від 200 000 грн на рік з урахуванням ліцензій та технічної підтримки. У порівнянні з одноразовими витратами на власну розробку (70 000 грн) запропоноване рішення є більш економічно вигідним та адаптованим до специфіки підприємства.

На підставі проведених розрахунків встановлено, що впровадження інформаційної системи управління замовленнями для ТОВ «БОДО Україна» є

економічно доцільним. Строк окупності становить менше двох місяців (формула (1.8)), річний економічний ефект перевищує 500 тис. грн (формула (1.6)), а показник рентабельності інвестицій становить понад 600% (формула (1.10)).

Отже, розроблення та впровадження створюваного програмного забезпечення повністю обґрунтоване з технічної та економічної точки зору.

1.7. Постановка задачі кваліфікаційної роботи

Метою кваліфікаційної роботи є розроблення інформаційної системи управління замовленнями для ТОВ «БОДО Україна», яка забезпечує:

- реєстрацію клієнтів;
- створення та редагування замовлень;
- управління статусами;
- облік оплат;
- формування аналітичних звітів;
- розмежування прав доступу користувачів.

Для досягнення мети необхідно вирішити такі задачі:

1. Провести аналіз предметної області.
2. Побудувати модель бізнес-процесу.
3. Спроекувати структуру бази даних.
4. Реалізувати серверну частину системи.
5. Розробити користувацький інтерфейс.
6. Забезпечити тестування програмного продукту.

1.8. Висновки до розділу 1

У першому розділі кваліфікаційної роботи було здійснено дослідження предметної області функціонування ТОВ «БОДО Україна» та проаналізовано особливості організації процесу управління замовленнями.

Розглянуто організаційну структуру підприємства та визначено підрозділи, залучені до обробки замовлень. Проведено аналіз життєвого циклу замовлення та розмежовано поняття завершення і закриття замовлення.

У результаті аналізу виявлено проблемні аспекти існуючої організації обробки замовлень, зокрема фрагментарність використання програмних інструментів та обмежені можливості аналітичної обробки даних.

Обґрунтовано необхідність розроблення спеціалізованої інформаційної системи управління замовленнями, яка забезпечить централізацію даних, підвищення ефективності роботи персоналу та підтримку прийняття управлінських рішень.

Таким чином, сформовано постановку задачі та визначено основні вимоги до розроблюваного програмного продукту.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ УПРАВЛІННЯ ЗАМОВЛЕННЯМИ

2.1. Функціональне моделювання процесу управління замовленнями

Для побудови функціональної моделі бізнес-процесів за методологією IDEF0 було використано онлайн-інструмент draw.io (diagrams.net), який забезпечує можливість створення структурованих діаграм відповідно до стандартів функціонального моделювання та дозволяє наочно відобразити логіку роботи інформаційної системи [23, 35].

Контекстна діаграма IDEF0 процесу управління замовленнями наведена у додатку А (рисунок А.1).

Центральна функція (A0).

Основною функцією системи є:

- «Управління замовленнями»:

Входи:

- дані клієнта;
- інформація про обрану послугу;
- дані про оплату;
- звернення клієнта.

Керуючі впливи:

- внутрішні регламенти підприємства;
- правила обробки замовлень;
- договірні умови співпраці з партнерами;
- вимоги до ведення фінансового обліку;
- політика інформаційної безпеки.

Механізми:

- менеджер з продажу;
- оператор служби підтримки;
- адміністратор системи;
- вебзастосунок;
- серверна частина;

- база даних.

Виходи:

- сформоване замовлення;
- оновлений статус замовлення;
- фінансова інформація;
- аналітичні звіти;
- інформація для партнерів.

Декомпозиція функції A0.

Деталізація процесу наведена у додатку А (рисунок А.2).

Основні підпроцеси:

- Реєстрація клієнта.
- Створення замовлення.
- Управління статусами.
- Облік оплат.
- Формування звітності.

Застосування IDEF0 дозволило чітко визначити межі функціональності майбутньої інформаційної системи та встановити логіку інформаційних потоків.

2.2. Проектування логічної моделі бази даних

Інформаційна система реалізується на основі реляційної моделі даних.

ER-діаграма системи наведена у додатку Б (рисунок Б.1) [18–20].

База даних включає такі сутності:

- Clients
- Services
- Orders
- Payments
- Users
- Partners
- Order_Status
- Payment_Method

2.2.1. Опис сутностей та їх атрибутів

В базі даних кожній сутності відповідають наступні атрибути.

Clients:

- `client_id` – первинний ключ;
- `full_name` – ПІБ клієнта;
- `email` – електронна пошта;
- `phone` – номер телефону;
- `created_at` – дата реєстрації.

Services:

- `service_id` – первинний ключ;
- `name` – назва послуги;
- `description` – опис;
- `price` – вартість;
- `duration` – тривалість;
- `partner_id` – зовнішній ключ на таблицю Partners.

Orders (центральна таблиця системи):

- `order_id` – первинний ключ;
- `created_at` – дата створення;
- `status_id` – зовнішній ключ на Order_Status;
- `client_id` – зовнішній ключ на Clients;
- `service_id` – зовнішній ключ на Services;
- `user_id` – відповідальний менеджер;
- `total_amount` – загальна сума оплат.

Payments:

- `payment_id` – первинний ключ;
- `order_id` – зовнішній ключ;
- `amount` – сума оплати;
- `paid_at` – дата оплати;
- `method_id` – спосіб оплати.

Users:

- user_id – первинний ключ;
- username – логін;
- password_hash – хеш пароля;
- role – роль (sales, support, admin).

Order_Status (довідник)

- status_id;
- name.

Payment_Method (довідник):

- method_id;
- name.

2.3. Аналіз зв'язків між сутностями

У структурі бази даних реалізовано такі типи зв'язків 1:N (один до багатьох):

- Один клієнт може мати багато замовлень.
- Одна послуга може бути використана у багатьох замовленнях.
- Одне замовлення може мати декілька оплат.
- Один статус може відповідати багатьом замовленням.

Зв'язки реалізовано за допомогою зовнішніх ключів із забезпеченням цілісності.

2.4. Нормалізація структури бази даних

Проектування структури виконано з приведенням до третьої нормальної форми [18, 19].

1НФ – всі атрибути атомарні.

2НФ – відсутні часткові залежності від складених ключів.

3НФ – відсутні транзитивні залежності між неключовими атрибутами.

Використання довідників Order_Status та Payment_Method усуває дублювання текстових значень.

2.5. Архітектура інформаційної системи

Система реалізована за клієнт-серверною архітектурою [11, 14, 36].

Архітектурна схема наведена у додатку В (рисунок В.1).

Клієнтська частина:

- веббраузер;
- HTML;
- CSS;
- шаблонізатор Jinja2 [27–30, 34].

Серверна частина:

- Flask;
- обробка HTTP-запитів;
- бізнес-логіка;
- взаємодія з базою даних SQLite.

База даних:

- централізоване зберігання інформації;
- механізм тригерів для автоматичного підрахунку сум.

Обґрунтування вибору архітектури.

Клієнт-серверний підхід забезпечує:

- централізацію обробки даних;
- зручність супроводу;
- масштабованість;
- можливість розширення функціоналу.

Для забезпечення надійності системи застосовано:

- первинні ключі;
- зовнішні ключі;
- обмеження NOT NULL;
- каскадні зв'язки;
- механізм авторизації;
- розмежування доступу за ролями.

Паролі користувачів зберігаються у вигляді хешів, що відповідає сучасним вимогам інформаційної безпеки.

2.6. Формалізація вимог до інформаційної системи

Формалізація вимог є важливим етапом проектування інформаційної системи, оскільки дозволяє чітко визначити функціональні можливості програмного продукту, обмеження, критерії якості та параметри експлуатації.

Вимоги до розроблюваної інформаційної системи управління замовленнями ТОВ «БОДО Україна» сформовано на основі аналізу предметної області та бізнес-процесів підприємства. Визначення життєвого циклу програмного забезпечення та критеріїв якості здійснюється відповідно до міжнародних стандартів [34, 35, 42].

Вимоги поділяються на:

- функціональні;
- нефункціональні;
- вимоги до безпеки;
- вимоги до продуктивності;
- обмеження та припущення.

2.6.1. Функціональні вимоги

Функціональні вимоги визначають перелік операцій, які повинна забезпечувати система.

FR-01. Авторизація користувачів.

Система повинна забезпечувати механізм входу користувача за логіном і паролем.

FR-02. Розмежування доступу.

Система повинна підтримувати ролі користувачів:

- менеджер з продажу;

- оператор служби підтримки;
- адміністратор.

FR-03. Перегляд замовлень.

Система повинна відображати список замовлень із можливістю пошуку та фільтрації.

FR-04. Фільтрація замовлень.

Система повинна підтримувати фільтрацію за:

- номером замовлення;
- ПІБ клієнта;
- статусом;
- періодом створення.

FR-05. Створення замовлення.

Система повинна забезпечувати створення нового замовлення із вибором клієнта та послуги.

FR-06. Автоматичне присвоєння номера.

Після збереження замовлення система повинна автоматично генерувати унікальний ідентифікатор.

FR-07. Зміна статусу замовлення.

Система повинна дозволяти зміну статусу замовлення відповідно до бізнес-логіки.

FR-08. Додавання оплат.

Система повинна підтримувати додавання однієї або кількох оплат до замовлення.

FR-09. Автоматичний перерахунок суми.

Після внесення оплати система повинна автоматично перераховувати загальну суму замовлення.

FR-10. Управління клієнтами.

Система повинна забезпечувати додавання та редагування даних клієнтів.

FR-11. Формування звітів.

Система повинна формувати аналітичні звіти за:

- періодом;

- статусом;
- оборотом;
- середнім чеком.

FR-12. КРІ-аналітика.

Система повинна розраховувати:

- загальний оборот;
- середній чек;
- відсоток оплачених замовлень;
- дебіторську заборгованість.

FR-13. Формування ТОП-5.

Система повинна визначати:

- ТОП-5 послуг за оборотом;
- ТОП-5 клієнтів за сумою оплат.

FR-14. Візуалізація динаміки

Система повинна відображати графік обороту по днях.

FR-15. Завершення сеансу.

Система повинна забезпечувати можливість виходу з облікового запису.

2.6.2. Нефункціональні вимоги

Нефункціональні вимоги визначають характеристики якості системи. Критерії якості програмного забезпечення формуються відповідно до міжнародних стандартів оцінювання якості ПЗ [43].

NFR-01. Продуктивність.

Система повинна забезпечувати відображення сторінки не більше ніж за 3 секунди при навантаженні до 1000 записів.

NFR-02. Масштабованість.

Система повинна підтримувати можливість розширення функціоналу без зміни базової архітектури.

NFR-03. Надійність.

Дані повинні зберігатися у централізованій базі даних із механізмом забезпечення цілісності.

NFR-04. Зручність використання.

Інтерфейс повинен бути інтуїтивно зрозумілим та відповідати сучасним стандартам веб-дизайну.

NFR-05. Портативність.

Система повинна працювати у сучасних веббраузерах без додаткових модулів.

NFR-06. Сумісність.

Система повинна функціонувати на операційних системах Windows, Linux та macOS.

NFR-07. Захист даних.

Система повинна запобігати несанкціонованому доступу до даних.

2.6.3. Вимоги до безпеки

- зберігання облікових записів користувачів;
- розмежування прав доступу;
- заборона доступу до адміністративних функцій неавторизованим особам;
- контроль доступу до розділу звітності.

2.6.4. Вимоги до бази даних

- підтримка реляційної моделі;
- використання первинних та зовнішніх ключів;
- забезпечення цілісності посилань;
- використання тригерів для автоматичного оновлення сум.

2.6.5. Обмеження та припущення

- система розроблена для внутрішнього використання підприємства;
- кількість одночасних користувачів не перевищує 20;
- обсяг бази даних не перевищує 100 000 записів;
- використовується СУБД SQLite як легковагове рішення.

2.7. Модель прецедентів (Use Case Model)

Модель прецедентів (Use Case) використовується для опису функціональної взаємодії користувачів із системою. Вона дозволяє формалізувати сценарії використання інформаційної системи з точки зору кінцевих користувачів та визначити межі відповідальності програмного продукту.

Для інформаційної системи управління замовленнями ТОВ «БОДО Україна» було визначено три основні актори:

- Менеджер з продажу
- Оператор служби підтримки
- Адміністратор

Кожен із акторів взаємодіє із системою відповідно до своїх посадових обов'язків.

2.7.1. Актори системи

Менеджер з продажу виконує такі дії:

- авторизація в системі;
- перегляд списку замовлень;
- створення нового замовлення;
- редагування даних клієнтів;
- зміна статусу замовлення;
- додавання оплат до замовлення.

Оператор служби підтримки виконує:

- авторизацію;
- перегляд замовлень;
- оновлення статусів;
- контроль виконання замовлень;
- додавання оплат.

Адміністратор має повний доступ до системи:

- управління користувачами;

- формування звітів;
- перегляд КРІ-аналітики;
- контроль доступу;
- адміністрування довідників.

2.7.2. Основні прецеденти використання

Нижче наведено текстовий опис ключових сценаріїв використання системи.

UC-01. Авторизація користувача.

Актор: будь-який користувач

Мета: отримання доступу до системи

Основний сценарій:

1. Користувач відкриває сторінку входу.
2. Вводить логін і пароль.
3. Натискає кнопку «Увійти».
4. Система перевіряє облікові дані.
5. У разі успіху відбувається перехід на головну сторінку.

Альтернативний сценарій:

- Якщо дані некоректні — система відображає повідомлення про помилку.

UC-02. Створення замовлення.

Актор: менеджер з продажу

Основний сценарій:

1. Користувач переходить у розділ «Замовлення».
2. Натискає кнопку «Створити замовлення».
3. Обирає клієнта.
4. Обирає послугу.
5. Встановлює початковий статус.
6. Зберігає замовлення.
7. Система присвоює унікальний номер.

UC-03. Зміна статусу замовлення.

Актор: менеджер або оператор

Основний сценарій:

1. Користувач відкриває конкретне замовлення.
2. Обирає новий статус.
3. Підтверджує зміну.
4. Система оновлює дані в базі.

UC-04. Додавання оплати.

Актор: менеджер або оператор

Основний сценарій:

1. Користувач відкриває замовлення.
2. Натискає «Додати оплату».
3. Вводить суму платежу.
4. Підтверджує збереження.
5. Система:
 - зберігає платіж;
 - автоматично перераховує загальну суму;
 - оновлює відображення.

UC-05. Формування звітів.

Актор: адміністратор

Основний сценарій:

1. Користувач переходить у розділ «Звіти».
2. Обирає параметри фільтрації.
3. Система формує:
 - КРІ-показники;
 - зведену таблицю;
 - графік обороту;
 - ТОП-5 послуг та клієнтів.

2.7.3. Взаємозв'язок прецедентів

Деякі прецеденти пов'язані між собою:

1. «Додавання оплати» розширює прецедент «Обробка замовлення» (<<extend>>).
2. «Авторизація» є обов'язковою умовою виконання всіх інших прецедентів (<<include>>).

Таким чином, модель прецедентів дозволяє формалізувати поведінку системи та чітко визначити межі функціонування програмного продукту.

2.8. UML-моделювання інформаційної системи

Для формалізації взаємодії користувачів із системою та опису структури програмного продукту було побудовано UML-діаграми відповідно до специфікації UML [21, 22, 41].

Діаграма прецедентів (Use Case Diagram) наведена у додатку Е (рис. Е.1) і відображає ролі користувачів та доступні їм функції системи.

Діаграма класів (Class Diagram) наведена у додатку Е (рис. Е.2) та демонструє структуру основних класів програмного продукту, їх атрибути, методи та зв'язки між ними.

Діаграма послідовності створення замовлення (Sequence Diagram) наведена у додатку Е (рис. Е.3) та ілюструє взаємодію між користувачем, вебінтерфейсом, серверною частиною та базою даних під час формування нового замовлення.

Діаграма діяльності (Activity Diagram) процесу управління замовленнями наведена у додатку Е (рис. Е.4) і відображає логіку переходів між станами замовлення.

Таким чином, використання UML-моделювання дозволило формалізувати як структурні, так і поведінкові аспекти інформаційної системи.

2.9. Висновки до розділу 2

У розділі здійснено проектування функціональної та логічної структури інформаційної системи управління замовленнями. Побудовано функціональну

модель процесу за методологією IDEF0, розроблено логічну модель бази даних, визначено основні сутності та зв'язки між ними.

Проведено нормалізацію структури бази даних до третьої нормальної форми, що забезпечує цілісність і узгодженість інформації. Обґрунтовано вибір клієнт-серверної архітектури та визначено механізми забезпечення безпеки даних.

Таким чином, створено структурну основу для реалізації програмного продукту, описаного у наступному розділі.

РОЗДІЛ 3. РОЗРОБЛЕННЯ ТА РЕАЛІЗАЦІЯ ПРОГРАМНОГО ПРОДУКТУ

3.1. Загальна характеристика програмної реалізації

У межах кваліфікаційної роботи реалізовано інформаційну систему управління замовленнями для ТОВ «БОДО Україна», що функціонує як вебзастосунок клієнт-серверної архітектури.

Система забезпечує:

- реєстрацію та облік клієнтів;
- створення та редагування замовлень;
- управління статусами;
- облік оплат;
- формування аналітичних звітів;
- розмежування прав доступу за ролями.

Програмний продукт реалізовано з використанням мови програмування Python та вебфреймворку Flask [24, 27]. Зберігання даних здійснюється у реляційній базі даних SQLite [30].

3.2. Архітектура програмного продукту

Програмна реалізація системи здійснена відповідно до розробленої архітектури (додаток Д, рис. Д.1).

Модульна структура включає:

- модуль авторизації користувачів;
- модуль управління клієнтами;
- модуль управління замовленнями;
- модуль обліку оплат;
- адміністративний модуль.

Взаємодія компонентів системи відповідає структурній моделі, наведеній на діаграмі класів (додаток Е, рис. Е.2).

Інформаційна система побудована за трирівневою моделлю:

1. Рівень представлення (Presentation Layer)

- HTML
- CSS
- шаблонізатор Jinja2 [28]
- веббраузер користувача

Забезпечує взаємодію користувача з системою.

2. Рівень прикладної логіки (Application Layer)

- Flask
- маршрутизація HTTP-запитів
- реалізація бізнес-логіки
- перевірка прав доступу

Саме на цьому рівні реалізовано:

- створення замовлень;
- зміну статусів;
- обробку оплат;
- формування звітів.

3. Рівень даних (Data Layer)

- SQLite
- SQL-запити
- механізм тригерів
- зовнішні ключі

Забезпечує зберігання та цілісність інформації.

3.3. Структура програмного проєкту

Структура програмного проєкту має модульний характер та включає:

- app.py – основний модуль;
- init_db.py – ініціалізація бази даних;
- templates/ – HTML-шаблони;
- static/ – стилі оформлення;
- база даних orders.db.

Такий підхід забезпечує розділення відповідальності та спрощує супровід.

3.4. Реалізація основних функціональних модулів

3.4.1. Модуль авторизації

Модуль забезпечує:

- перевірку логіну та пароля;
- визначення ролі користувача;
- створення сесії;
- обмеження доступу до функціоналу.

Ролі користувачів:

- sales – створення замовлень, робота з клієнтами;
- support – оновлення статусів;
- admin – повний доступ.

Паролі зберігаються у вигляді хешів відповідно до рекомендацій OWASP та NIST [31–33, 39, 40].

3.4.2. Модуль управління замовленнями

Модуль реалізує:

- створення замовлення;
- редагування даних;
- зміну статусу;
- перегляд списку замовлень;
- фільтрацію.

При створенні замовлення автоматично:

- генерується унікальний ідентифікатор;
- встановлюється початковий статус;
- фіксується дата створення.

3.4.3. Модуль обліку оплат

Модуль дозволяє:

- додавати одну або кілька оплат;
- визначати спосіб оплати;
- автоматично перераховувати суму замовлення.

Автоматичний перерахунок реалізовано за допомогою тригерів бази даних [30].

3.4.4. Модуль формування звітів

Система підтримує формування таких звітів:

- звіт за періодом (кількість замовлень та сума оплат);
- звіт за статусами;
- звіт за менеджерами;
- комбінований звіт (період + статус).

Формування звітів здійснюється за допомогою SQL-запитів із групуванням та агрегацією даних.

3.5. Вимоги до програмного забезпечення та технічного середовища

Мінімальні системні вимоги:

- ОС Windows 10 / Linux;
- Python 3.10 або вище;
- 4 ГБ оперативної пам'яті;
- веббраузер (Google Chrome, Mozilla Firefox).

Програмні залежності:

- Flask;
- sqlite3;
- Jinja2.

Система не потребує встановлення додаткового серверного програмного забезпечення.

3.6. Тестування програмного продукту

Тестування здійснювалося на етапі розроблення із застосуванням функціонального тестування [7].

Таблиця 3.1 – Основні перевірені сценарії

№	Тестовий сценарій	Очікуваний результат
1	Авторизація з правильними даними	Успішний вхід
2	Авторизація з помилковими даними	Повідомлення про помилку
3	Створення замовлення	Запис у БД
4	Додавання оплати	Оновлення total_amount
5	Зміна статусу	Оновлення поля status

Усі тестові сценарії виконані успішно.

3.7. Забезпечення надійності та безпеки

У системі реалізовано :

- перевірку прав доступу;
- розмежування ролей;
- захист від несанкціонованого доступу;
- цілісність даних через зовнішні ключі;
- перевірку коректності введених даних.

Паролі користувачів зберігаються у вигляді хешів [9, 31–33].

Реалізація механізмів автентифікації та контролю доступу відповідає сучасним стандартам інформаційної безпеки [39, 40].

Розроблення програмного продукту здійснювалося з урахуванням вимог до структури та змісту кваліфікаційних робіт, визначених НУХТ [1–5].

3.8. Інструкція користувача інформаційної системи управління замовленнями

Інструкція користувача описує порядок роботи з розробленою інформаційною системою управління замовленнями для ТОВ «БОДО Україна» та визначає послідовність виконання основних операцій у системі.

3.8.1. Запуск та вхід у систему

Для початку роботи користувач відкриває веббраузер та переходить за адресою інформаційної системи, наданою адміністратором.

Після завантаження сторінки авторизації необхідно:

1. Ввести логін у відповідне поле.
2. Ввести пароль.
3. Натиснути кнопку «Увійти».

У разі коректного введення облікових даних система автоматично перенаправляє користувача на головну сторінку.

У випадку помилки відображається повідомлення про некоректні дані доступу.

Зовнішній вигляд форми авторизації наведено у додатку А (рис. А.1).

3.8.2. Головна сторінка системи

Після успішної авторизації користувач переходить на головну сторінку системи, яка містить:

- головне меню навігації;
- таблицю замовлень;
- елементи пошуку та фільтрації;
- службові кнопки для виконання основних дій.

У таблиці відображається така інформація:

- номер замовлення;

- ПІБ клієнта;
- назва послуги;
- статус замовлення;
- загальна сума оплат;
- дата створення.

Інтерфейс головної сторінки представлено у додатку А (рис. А.2).

3.8.3. Робота з клієнтами

Для переходу до розділу «Клієнти» користувач обирає відповідний пункт меню.

Додавання нового клієнта

Для створення нового клієнта необхідно:

1. Натиснути кнопку «Додати клієнта».
2. Заповнити поля форми (ПІБ, телефон, електронна пошта).
3. Натиснути кнопку «Зберегти».

Після збереження дані клієнта автоматично додаються до бази даних та відображаються у списку.

Форма додавання нового клієнта наведена у додатку А (рис. А.3).

3.8.4. Створення та обробка замовлень

Для створення замовлення користувач виконує такі дії:

1. Перейти до розділу «Замовлення».
2. Натиснути кнопку «Створити замовлення».
3. Обрати клієнта зі списку.
4. Обрати послугу.
5. Встановити початковий статус замовлення.

6. Натиснути кнопку «Зберегти».

Після збереження замовленню автоматично присвоюється унікальний номер.

Форма створення замовлення наведена у додатку А (рис. А.4).

3.8.5. Зміна статусу замовлення

У процесі виконання замовлення користувач може змінювати його статус відповідно до етапів обробки (наприклад: «Нове», «Оплачене», «В роботі», «Виконане», «Закрите»).

Для цього необхідно:

1. Обрати замовлення зі списку.
2. Встановити новий статус.
3. Підтвердити зміну.

Після збереження статусу автоматично оновлюється у базі даних.

Вікно зміни статусу представлено у додатку А (рис. А.5).

3.8.6. Облік оплат

Система підтримує можливість додавання оплат до замовлення.

Для додавання оплати необхідно:

1. Відкрити конкретне замовлення.
2. Натиснути кнопку «Додати оплату».
3. Ввести суму платежу.
4. Підтвердити збереження.

Після додавання платежу:

- інформація зберігається у базі даних;

- автоматично перераховується загальна сума замовлення;
- оновлюється відображення у списку замовлень.

3.8.7. Формування звітів

У розділі «Звіти» (доступний для адміністратора) реалізовано аналітичний функціонал. Тексти SQL-запитів, що використовуються для формування аналітичних звітів, наведені у додатку Є (лістинги Є.1–Є.5).

Окрім базових вибірок, у системі реалізовано аналітичні звіти з використанням агрегування, підзапитів та умовної логіки CASE (додаток Є, лістинги Є.6–Є.9).

Система дозволяє:

- переглядати загальний оборот;
- визначати середній чек;
- отримувати кількість замовлень;
- аналізувати відсоток оплачених замовлень;
- переглядати дебіторську заборгованість;
- визначати найприбутковіший день;
- формувати ТОП-5 послуг;
- формувати ТОП-5 клієнтів;
- переглядати динаміку обороту по днях (графік).

Звіти можуть фільтруватися:

- за статусом;
- за періодом (дата від – дата до).

Інтерфейс формування звітів наведено у додатку А (рис. А.6).

Звіти формуються автоматично на основі даних бази даних та можуть використовуватись для аналізу діяльності підприємства.

3.8.8. Завершення роботи

Після завершення роботи користувач повинен:

1. Натиснути кнопку «Вийти» у головному меню.

2. Закрити веббраузер.

Це забезпечує коректне завершення сеансу роботи та підвищує рівень інформаційної безпеки.

3.9. Висновки до розділу 3

У розділі здійснено розроблення та реалізацію програмного продукту відповідно до спроектованої моделі системи. Описано архітектуру, структуру програмного проєкту, реалізацію основних функціональних модулів та результати тестування.

Розроблена інформаційна система забезпечує автоматизацію процесу управління замовленнями, централізацію зберігання даних та підтримку прийняття управлінських рішень.

ВИСНОВКИ

У кваліфікаційній роботі виконано проєктування та реалізацію інформаційної системи управління замовленнями для ТОВ «БОДО Україна». У процесі виконання роботи було здійснено комплексний аналіз предметної області, визначено особливості бізнес-процесу обробки замовлень та обґрунтовано необхідність автоматизації відповідних процесів.

У першому розділі проведено дослідження діяльності підприємства та визначено основні проблемні аспекти організації управління замовленнями. Встановлено, що використання розрізнених програмних інструментів ускладнює контроль виконання замовлень та обмежує можливості аналітичної обробки інформації.

У другому розділі виконано проєктування інформаційної системи. Побудовано функціональну модель бізнес-процесу за методологією IDEF0, розроблено логічну модель бази даних та ER-діаграму, проведено нормалізацію структури до третьої нормальної форми. Застосування UML-діаграм (додаток Е, рис. Е.1–Е.4) дозволило формалізувати функціональні вимоги та визначити ролі користувачів системи. Проєктування виконано з урахуванням сучасних підходів до розроблення програмних систем та управління якістю програмного забезпечення [34–38].

У третьому розділі реалізовано програмний продукт на основі мови програмування Python та вебфреймворку Flask. Розроблено модулі авторизації користувачів, управління замовленнями, обліку оплат та формування аналітичних звітів. Забезпечено розмежування прав доступу відповідно до ролей користувачів. Проведено функціональне тестування системи, яке підтвердило коректність роботи програмного забезпечення.

Розроблена інформаційна система забезпечує централізоване зберігання даних, автоматизацію основних операцій з обробки замовлень та формування аналітичної звітності. Використання системи сприяє підвищенню ефективності управління діяльністю підприємства та зменшенню ризику помилок, пов'язаних із людським фактором.

Таким чином, поставлена мета роботи досягнута, а розроблений програмний продукт може бути використаний для автоматизації процесу управління замовленнями підприємства.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Інтелектуальні системи [Електронний ресурс]: методичні рекомендації до вивчення дисципліни та виконання контрольної роботи для здобувачів освітнього ступеня «Бакалавр» спеціальності 122 «Комп'ютерні науки» / уклад. Н. М. Бреус; Національний університет харчових технологій. – Київ: НУХТ, 2023. – 199 с. – Режим доступу: <https://library.nuft.edu.ua/> (дата звернення: 11.02.2026).
2. Web-технології та web-дизайн [Електронний ресурс]: методичні рекомендації / уклад. О. О. Мазуренко; НУХТ. – Київ: НУХТ, 2023. – 22 с. – Режим доступу: <https://library.nuft.edu.ua/> (дата звернення: 11.02.2026).
3. Основи програмування та розробка алгоритмів [Електронний ресурс]: методичні рекомендації / уклад. С. В. Грибков, О. Л. Сєдих; НУХТ. – Київ: НУХТ, 2023. – 75 с. – Режим доступу: <https://library.nuft.edu.ua/> (дата звернення: 08.02.2026).
4. Шикуча О. М. Система керування базами даних MS Access: навчальний посібник. – Київ: ІПДО НУХТ, 2017. – 177 с.
5. Ладанюк А. П., Смітюх Я. В., Власенко Л. О. та ін. Системний аналіз складних систем управління: навчальний посібник. – Київ: НУХТ, 2013. – 274 с.
6. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. – Київ: УкрНДНЦ, 2016. – 16 с.
7. ISO/IEC/IEEE 12207:2017. Systems and software engineering – Software life cycle processes. – Geneva: International Organization for Standardization, 2017.
8. ISO/IEC 25010:2011. Systems and software engineering – Systems and software quality models. – Geneva: International Organization for Standardization, 2011.
9. ISO/IEC 27001:2022. Information security management systems – Requirements. – Geneva : International Organization for Standardization, 2022.
10. ISO 8601-1:2019. Date and time – Representations for information interchange – Part 1. – Geneva: International Organization for Standardization, 2019.

11. Pressman R. S., Maxim B. R. Software Engineering: A Practitioner's Approach. – 9th ed. – New York : McGraw-Hill Education, 2022.
12. Sommerville I. Software Engineering. – 11th ed. – Pearson, 2023.
13. Martin R. C. Clean Agile: Back to Basics. – Pearson, 2021.
14. Martin R. C. Clean Architecture: A Craftsman's Guide to Software Structure and Design. – Boston : Pearson, 2017.
15. Fowler M. Refactoring: Improving the Design of Existing Code. – 2nd ed. – Boston : Addison-Wesley, 2018.
16. Freeman E., Robson E. Head First Design Patterns. – 2nd ed. – Sebastopol : O'Reilly Media, 2021.
17. McConnell S. Code Complete. – 2nd ed. – Redmond : Microsoft Press, 2004.
18. Elmasri R., Navathe S. B. Fundamentals of Database Systems. – 7th ed. – Boston : Pearson, 2016.
19. Silberschatz A., Korth H., Sudarshan S. Database System Concepts. – 7th ed. – New York : McGraw-Hill Education, 2019.
20. Connolly T., Begg C. Database Systems. – 7th ed. – Pearson, 2022.
21. Booch G., Rumbaugh J., Jacobson I. The Unified Modeling Language User Guide. – 2nd ed. – Boston : Addison-Wesley, 2005.
22. OMG. Unified Modeling Language (UML) Specification [Электронный ресурс]. – Режим доступа: <https://www.omg.org/spec/UML/> (дата звернення: 08.02.2026).
23. Mayer R. J. IDEF0 Function Modeling Method Report. – Knowledge Based Systems Inc., 1992.
24. Python 3 Documentation [Электронный ресурс]. – Режим доступа: <https://docs.python.org/3/> (дата звернення: 08.02.2026).
25. PEP 8 – Style Guide for Python Code [Электронный ресурс]. – Режим доступа: <https://peps.python.org/pep-0008/> (дата звернення: 08.02.2026).
26. PEP 249 – Python Database API Specification v2.0 [Электронный ресурс]. – Режим доступа: <https://peps.python.org/pep-0249/> (дата звернення: 08.02.2026).
27. Flask Documentation [Электронный ресурс]. – Режим доступа: <https://flask.palletsprojects.com/> (дата звернення: 08.02.2026).

28. Jinja Documentation [Электронный ресурс]. – Режим доступа: <https://jinja.palletsprojects.com/> (дата звернення: 08.02.2026).
29. Werkzeug Documentation [Электронный ресурс]. – Режим доступа: <https://werkzeug.palletsprojects.com/> (дата звернення: 08.02.2026).
30. SQLite Documentation [Электронный ресурс]. – Режим доступа: <https://www.sqlite.org/docs.html> (дата звернення: 08.02.2026).
31. OWASP Top 10:2021 [Электронный ресурс]. – Режим доступа: <https://owasp.org/Top10/> (дата звернення: 08.02.2026).
32. OWASP Cheat Sheet Series [Электронный ресурс]. – Режим доступа: <https://cheatsheetseries.owasp.org/> (дата звернення: 08.02.2026).
33. NIST SP 800-218. Secure Software Development Framework (SSDF) [Электронный ресурс]. – Режим доступа: <https://csrc.nist.gov/> (дата звернення: 08.02.2026).
34. MDN Web Docs. HTML Reference [Электронный ресурс]. – Режим доступа: <https://developer.mozilla.org/> (дата звернення: 08.02.2026).
35. diagrams.net Documentation [Электронный ресурс]. – Режим доступа: <https://www.diagrams.net/doc/> (дата звернення: 08.02.2026).
36. ISO/IEC 27002:2022. Information security controls. – Geneva : International Organization for Standardization, 2022.
37. OWASP Application Security Verification Standard 4.0.3 [Электронный ресурс]. – Режим доступа: <https://owasp.org/> (дата звернення: 11.02.2026).
38. NIST SP 800-53 Rev. 5 [Электронный ресурс]. – Режим доступа: <https://csrc.nist.gov/> (дата звернення: 11.02.2026).
39. Ford N., Richards M., Parsons R., Kua P. Software Architecture: The Hard Parts. – Sebastopol : O'Reilly Media, 2021.
40. Richardson C. Microservices Patterns. – 2nd ed. – Shelter Island : Manning Publications, 2023.
41. ISO/IEC 25023:2021. Systems and software engineering – Systems and software Quality Requirements and Evaluation – Measurement of system and software product quality. – Geneva : International Organization for Standardization, 2021.

42. PostgreSQL 16 Documentation [Электронный ресурс]. – Режим доступа: <https://www.postgresql.org/docs/> (дата звернения: 11.02.2026).

43. MDN Web Docs. Web Security Guidelines [Электронный ресурс]. – Режим доступа: <https://developer.mozilla.org/> (дата звернения: 11.02.2026).

ДОДАТКИ

Додаток А. Матеріали інструкції користувача

В ТОВ «БОДО Україна» Вхід

Вхід у систему

Введіть логін та пароль.

Логін Пароль

sales1

[Увійти](#)

Тестові: sales1/1234, support1/1234, admin/admin

Рисунок А.1 – Форма входу користувача до системи

В ТОВ «БОДО Україна» Замовлення Клієнти Звіти Вийти (admin)

Список замовлень

Пошук, фільтрація та контроль виконання.

[+ Створити замовлення](#)

Пошук (ПІБ або №) Фільтрувати

Іваненко або 12 Скинути

Статус Дата від Дата до

Усі дд . мм . рррр дд . мм . рррр

№	ПІБ КЛІЄНТА	ПОСЛУГА	СТАТУС	СУМА	ДАТА	ДІЇ
3	Шевченко Олена Сергіївна	Квест-кімната	В роботі	0.0 грн	2026-02-10 13:16:05	Статус Оплата
2	Петренко Петро Петрович	SPA-враження	Оплачене	2500.0 грн	2026-02-09 13:16:05	Статус Оплата
1	Іваненко Іван Іванович	Подарунковий сертифікат 1000 грн	Нове	0.0 грн	2026-02-08 13:16:05	Статус Оплата

Рисунок А.2 – Головна сторінка інформаційної системи

В ТОВ «БОДО Україна» [Замовлення](#) [Клієнти](#) [Звіти](#) [Вийти \(admin\)](#)

Клієнти

Додавання та перегляд клієнтів.

ПІБ: Грибков Сергій Віталійович Телефон: +3809652315 Email: aks@test.ua

[Зберегти](#)

ID	ПІБ	ТЕЛЕФОН	EMAIL	дії
3	Шевченко Олена Сергіївна	+380931112244	shevchenko@test.ua	Редагувати
2	Петренко Петро Петрович	+380671234567	petrenko@test.ua	Редагувати
1	Іваненко Іван Іванович	+380501112233	ivanenko@test.ua	Редагувати

Рисунок А.3 – Форма додавання нового клієнта

В ТОВ «БОДО Україна» [Замовлення](#) [Клієнти](#) [Звіти](#) [Вийти \(admin\)](#)

Створення замовлення

Оберіть клієнта, послугу та статус.

Клієнт: Шевченко Олена Сергіївна Послуга: Подарунковий сертифікат 1000 грн (1000.0 грн) Статус: Нове

[Зберегти](#) [Скасувати](#)

Рисунок А.4 – Форма створення замовлення

В ТОВ «БОДО Україна» [Замовлення](#) [Клієнти](#) [Звіти](#) [Вийти \(admin\)](#)

Зміна статусу замовлення №4

Шевченко Олена Сергіївна — Подарунковий сертифікат 1000 грн

Новий статус:

- Нове
- Оплачене
- В роботі
- Виконане
- Закрите

Рисунок А.5 – Вікно зміни статусу замовлення

Звіти

KPI, частки статусів, ТОПи, дебіторка та динаміка по днях

Усі Тільки Нові Тільки Оплачені

Статус Оплачене Дата від 26.01.2026 Дата до 01.02.2026 Застосувати Скинути

Загальний оборот
85602.00 грнСередній чек
6584.77 грнКількість замовлень
13% оплачених (конверсія)
100.00%Дебіторська заборгованість
36329.00 грнНайприбутковіший день
2026-01-26 • 36015.00 грн

Статуси (кількість / сума / частка)

СТАТУС	К-СТЬ	СУМА, ГРН	%
Оплачене	13	85602.00	100.00%

ТОП-5 послуг (оборот + середній чек)

ПОСЛУГА	К-СТЬ	ОБОРОТ	СЕР. ЧЕК
Вечера в ресторани №38	1	14660.00	14660.00
Подарунковий сертифікат №11	1	12597.00	12597.00
SPA-враження №10	1	12251.00	12251.00
Квест-кімната №6	1	11652.00	11652.00
Картинг №37	1	11167.00	11167.00

ТОП-5 клієнтів (оборот)

КЛІЄНТ	К-СТЬ	ОБОРОТ
Іваненко Андрій Петрович	1	14660.00
Ткаченко Світлана Іванович	1	12597.00
Іваненко Дмитро Олександрович	1	12251.00
Петренко Марія Петрович	1	11652.00
Олійник Юрій Сергійвна	1	11167.00

Детальні замовлення (останні 200)

ТОП-5 послуг (оборот + середній чек)

ПОСЛУГА	К-СТЬ	ОБОРОТ	СЕР. ЧЕК
Вечера в ресторани №38	1	14660.00	14660.00
Подарунковий сертифікат №11	1	12597.00	12597.00
SPA-враження №10	1	12251.00	12251.00
Квест-кімната №6	1	11652.00	11652.00
Картинг №37	1	11167.00	11167.00

ТОП-5 клієнтів (оборот)

КЛІЄНТ	К-СТЬ	ОБОРОТ
Іваненко Андрій Петрович	1	14660.00
Ткаченко Світлана Іванович	1	12597.00
Іваненко Дмитро Олександрович	1	12251.00
Петренко Марія Петрович	1	11652.00
Олійник Юрій Сергійвна	1	11167.00

Детальні замовлення (останні 200)

ID	КЛІЄНТ	ПОСЛУГА	СТАТУС	СУМА	ДАТА
72	Шевченко Марія Іванівна	Йога-ретрит №22	Оплачене	1604.00	2026-02-01 18:33:33
111	Бондаренко Дмитро Сергійович	Квест-кімната №35	Оплачене	1103.00	2026-01-31 18:33:33
145	Петренко Марія Петрович	Квест-кімната №6	Оплачене	11652.00	2026-01-31 18:33:33
105	Іваненко Андрій Петрович	Вечера в ресторани №38	Оплачене	14660.00	2026-01-30 18:33:33
127	Бондаренко Дмитро Сергійович	Вечера в ресторани №32	Оплачене	0.00	2026-01-30 18:33:33
51	Ткаченко Петро Іванович	Квест-кімната №24	Оплачене	7444.00	2026-01-29 18:33:33
138	Шевченко Дмитро Олександрівна	Йога-ретрит №22	Оплачене	0.00	2026-01-29 18:33:33
7	Іваненко Наталія Петрович	Йога-ретрит №27	Оплачене	0.00	2026-01-28 18:33:33
22	Шевченко Іван Іванівна	Йога-ретрит №21	Оплачене	10228.00	2026-01-28 18:33:33
52	Шевченко Дмитро Петрович	Квест-кімната №17	Оплачене	2896.00	2026-01-27 18:33:33

Рисунок А.6 – Сторінка формування звітів

Додаток Б. ER-діаграма бази даних

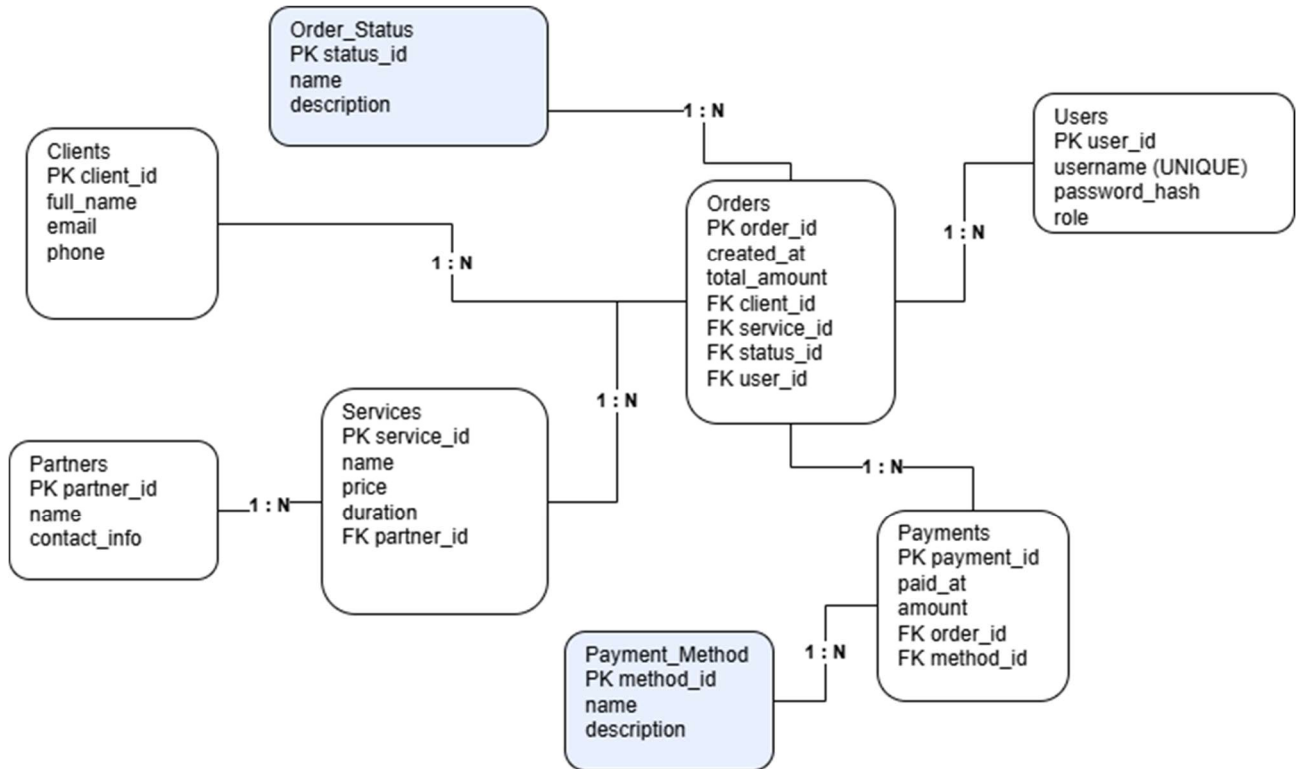


Рисунок Б.1 – ER-діаграма інформаційної системи

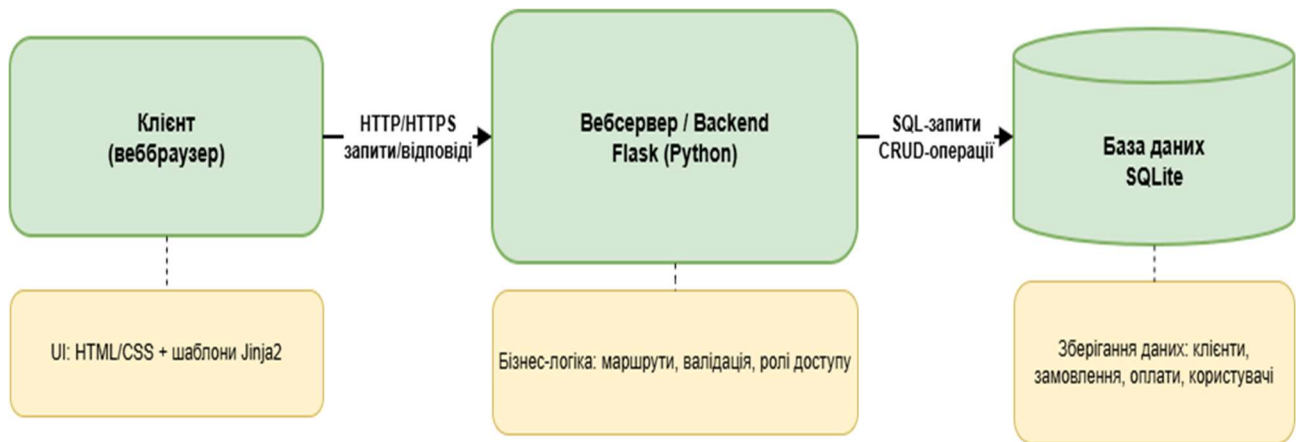
Додаток В. Архітектурна схема інформаційної системи

Рисунок В.1 – Архітектура інформаційної системи (клієнт–сервер / 3 рівні)



Рисунок Г.1 – Контекстна діаграма IDEF0 (A0)



Рисунок Г.2 – Декомпозиція IDEF0 (A3)

Додаток Д. Архітектурна схема програмного продукту

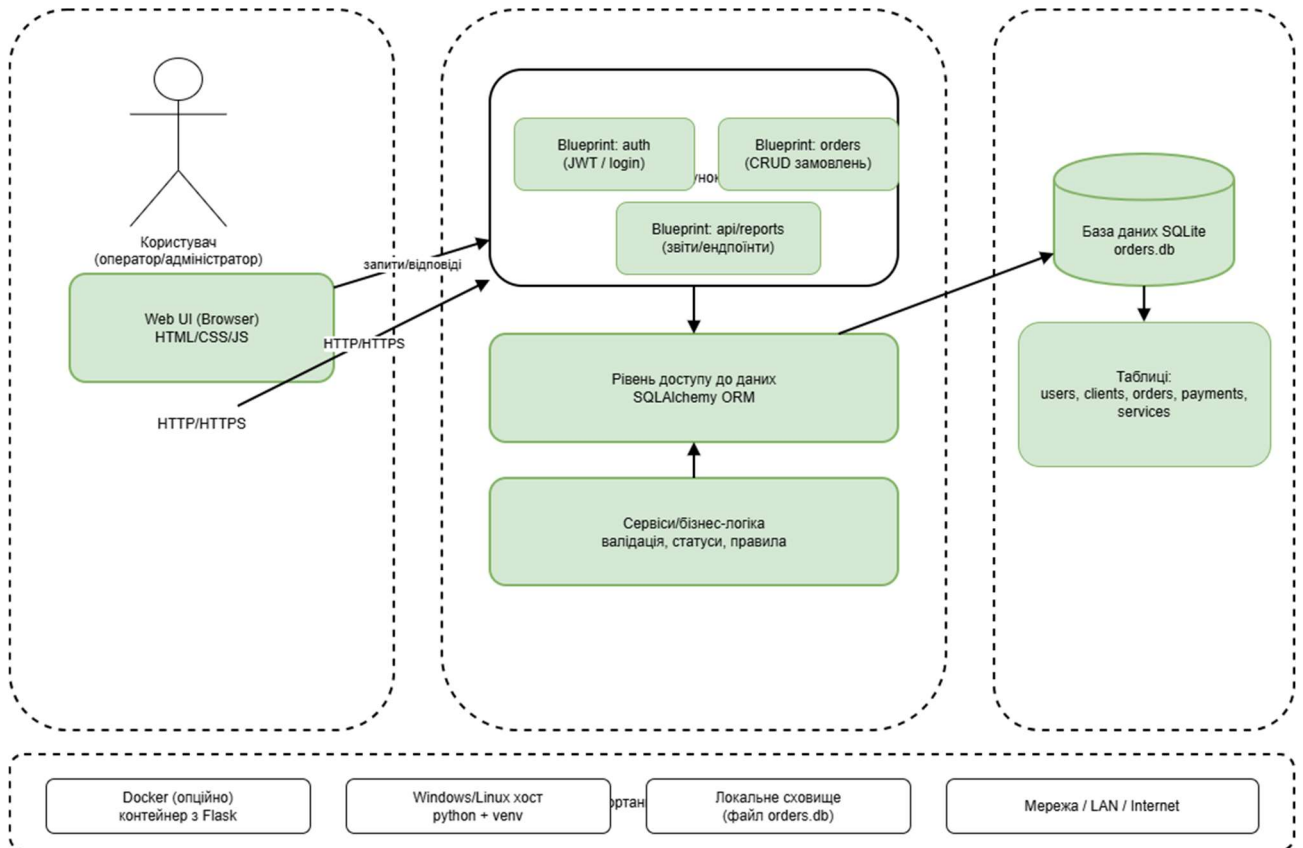


Рисунок Д.1 – Архітектурна схема програмного продукту

Додаток Е. UML-діаграми інформаційної системи

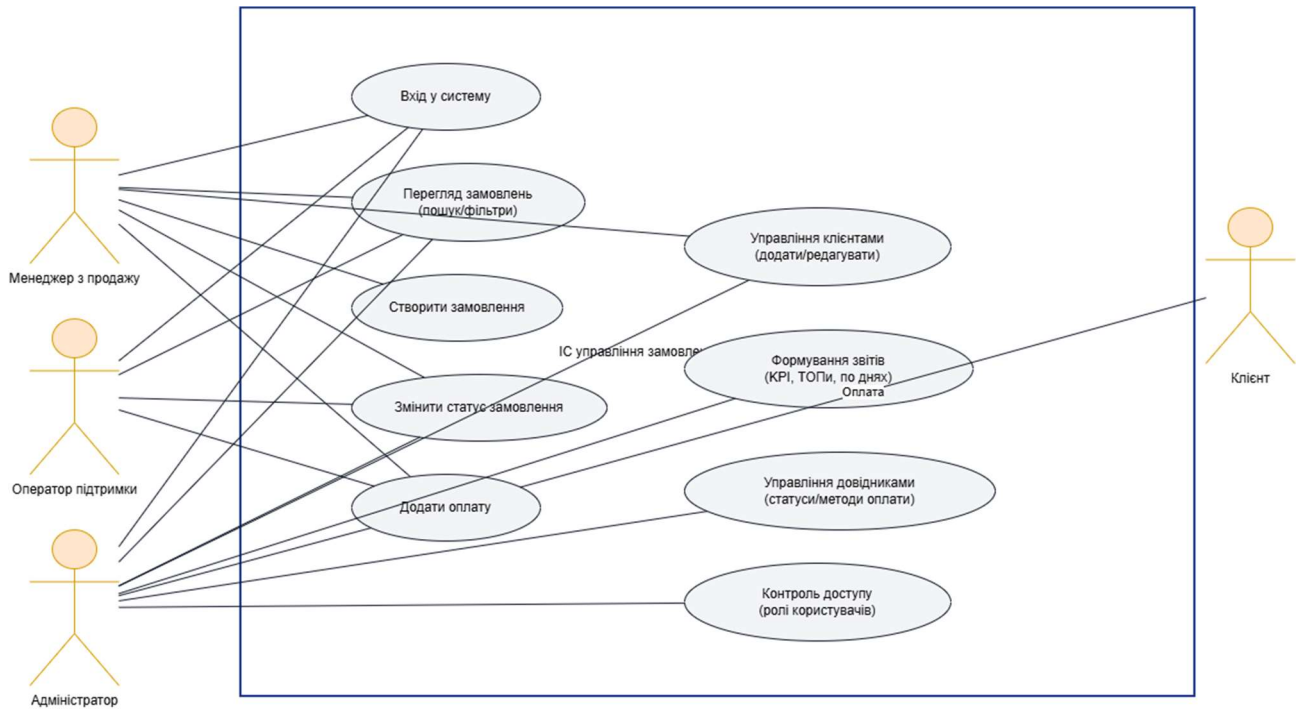


Рисунок Е.1 – Use Case Diagram інформаційної системи

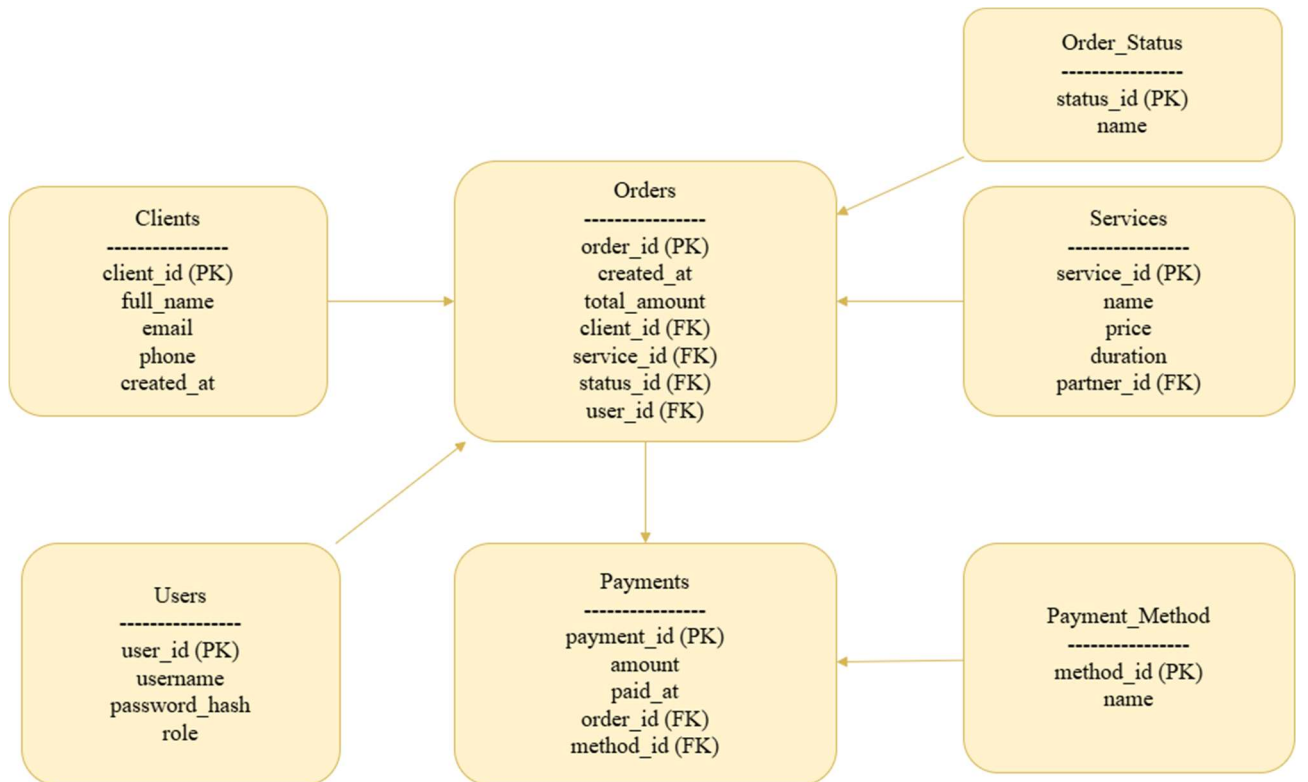


Рисунок Е.2 – Class Diagram системи

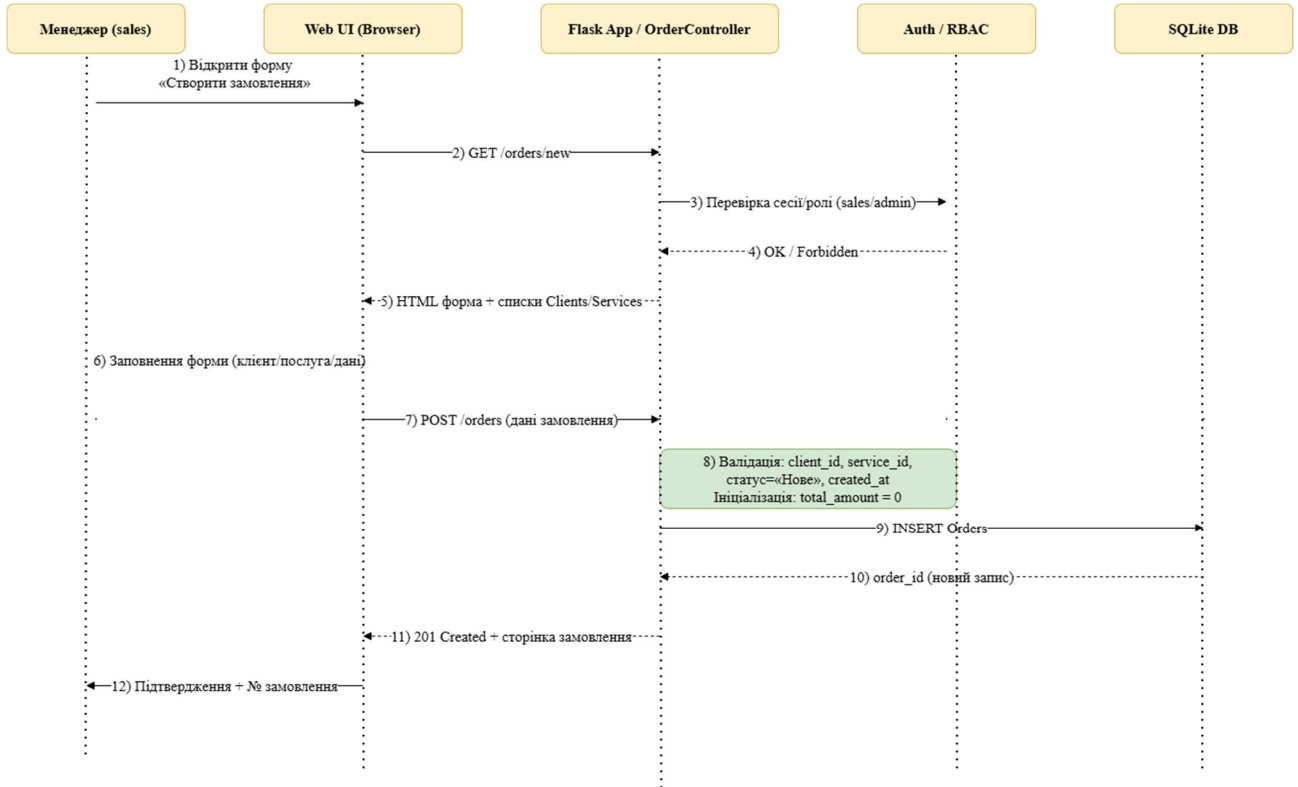


Рисунок Е.3 – Sequence Diagram створення замовлення

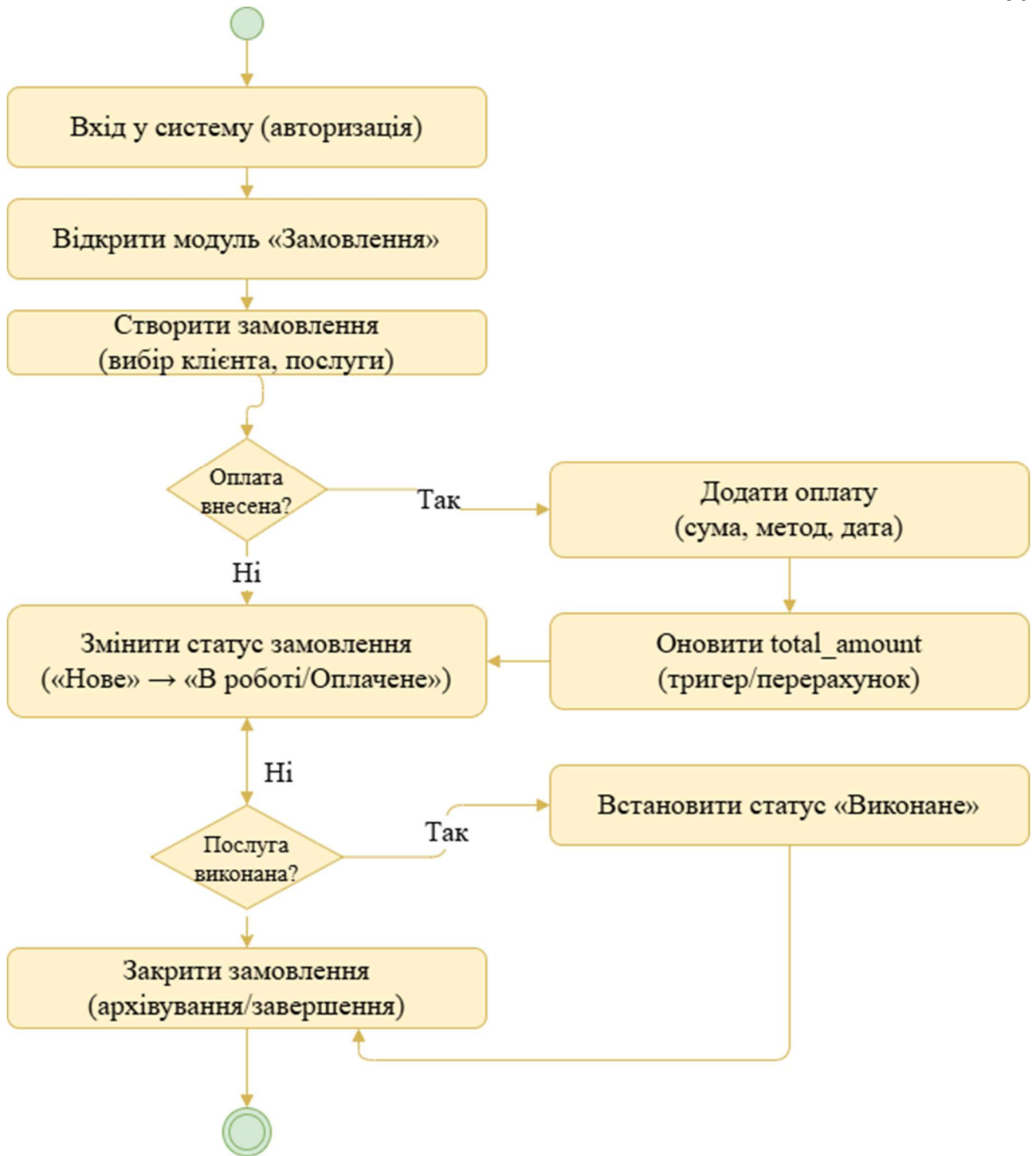


Рисунок Е.4 – Activity Diagram процесу управління замовленнями

ДОДАТОК Є. SQL-запити формування звітності інформаційної системи

Лістинг Є.1. Звіт по всіх замовленнях (деталізація).

```
SELECT

o.order_id,

c.full_name AS client,

s.name AS service,

o.created_at,

o.total_amount,

os.name AS status,

u.username AS manager

FROM Orders o

JOIN Clients c ON c.client_id = o.client_id

JOIN Services s ON s.service_id = o.service_id

JOIN Order_Status os ON os.status_id = o.status_id

JOIN Users u ON u.user_id = o.user_id

ORDER BY o.created_at DESC;
```

Лістинг Є.2. Оплати за період (сума оплат + кількість оплат).

```
SELECT

o.order_id,
```

```
c.full_name AS client,  
  
SUM(p.amount) AS total_paid,  
  
COUNT(p.payment_id) AS payments_count  
  
FROM Orders o  
  
JOIN Clients c ON c.client_id = o.client_id  
  
JOIN Payments p ON p.order_id = o.order_id  
  
WHERE p.paid_at BETWEEN :start_date AND :end_date  
  
GROUP BY o.order_id, c.full_name  
  
ORDER BY total_paid DESC;
```

Лістинг Є.3. Звіт по статусах замовлень.

```
SELECT  
  
os.name AS status,  
  
COUNT(o.order_id) AS total_orders  
  
FROM Order_Status os  
  
LEFT JOIN Orders o ON o.status_id = os.status_id  
  
GROUP BY os.name  
  
ORDER BY total_orders DESC;
```

Лістинг Є.4. Популярність послуг.

```
SELECT
```

```
s.name AS service,  
  
COUNT(o.order_id) AS orders_count  
  
FROM Services s  
  
LEFT JOIN Orders o ON o.service_id = s.service_id  
  
GROUP BY s.name  
  
ORDER BY orders_count DESC;
```

Лістинг Є.5. Звіт по менеджерах.

```
SELECT  
  
u.username,  
  
COUNT(o.order_id) AS created_orders,  
  
COALESCE(SUM(o.total_amount), 0) AS total_amount_sum  
  
FROM Users u  
  
LEFT JOIN Orders o ON o.user_id = u.user_id  
  
GROUP BY u.username  
  
ORDER BY created_orders DESC;
```

Лістинг Є.6. Дебіторка: оплаченість замовлень (CASE + підзапит)

```
SELECT  
  
o.order_id,
```

```
c.full_name AS client,

o.created_at,

o.total_amount,

COALESCE(paid.total_paid, 0) AS total_paid,

(o.total_amount - COALESCE(paid.total_paid, 0)) AS debt,

CASE

    WHEN COALESCE(paid.total_paid, 0) = 0 THEN 'Не оплачено'

    WHEN COALESCE(paid.total_paid, 0) < o.total_amount THEN 'Частково

оплачено'

    ELSE 'Оплачено'

END AS payment_status

FROM Orders o

JOIN Clients c ON c.client_id = o.client_id

LEFT JOIN (

    SELECT order_id, SUM(amount) AS total_paid

    FROM Payments

    GROUP BY order_id

) paid ON paid.order_id = o.order_id

ORDER BY debt DESC, o.created_at DESC;
```

Лістинг Є.7. Платіжна дисципліна: дні до першої оплати (підзапит + CASE).

```
SELECT

    o.order_id,

    c.full_name AS client,

    o.created_at,

    fp.first_paid_at,

CASE

    WHEN fp.first_paid_at IS NULL THEN NULL

    ELSE CAST(julianday(fp.first_paid_at) - julianday(o.created_at) AS
INTEGER)

END AS days_to_first_payment,

CASE

    WHEN fp.first_paid_at IS NULL THEN 'Оплати немає'

    WHEN (julianday(fp.first_paid_at) - julianday(o.created_at)) <= 1 THEN
'Швидко (0–1 дн.)'

    WHEN (julianday(fp.first_paid_at) - julianday(o.created_at)) <= 3 THEN
'Нормально (2–3 дн.)'

    ELSE 'Повільно (4+ дн.)'

END AS payment_speed_group

FROM Orders o
```

```

JOIN Clients c ON c.client_id = o.client_id

LEFT JOIN (

    SELECT order_id, MIN(paid_at) AS first_paid_at

    FROM Payments

    GROUP BY order_id

) fp ON fp.order_id = o.order_id

ORDER BY days_to_first_payment DESC, o.created_at DESC;

```

Лістинг Є.8. Комбінований звіт: період + статус (аналітика для адміністратора).

```

SELECT

    os.name AS status,

    COUNT(DISTINCT o.order_id) AS orders_count,

    COALESCE(SUM(p.amount), 0) AS paid_sum

FROM Order_Status os

LEFT JOIN Orders o ON o.status_id = os.status_id

LEFT JOIN Payments p ON p.order_id = o.order_id

    AND p.paid_at BETWEEN :start_date AND :end_date

GROUP BY os.name

ORDER BY orders_count DESC;

```

Лістинг Є.9. Кумулятивна виручка по днях (СТЕ + віконна функція).

```
WITH daily_paid AS (  
  
    SELECT  
  
        date(paid_at) AS day,  
  
        SUM(amount) AS paid_sum  
  
    FROM Payments  
  
    GROUP BY date(paid_at)  
  
)  
  
SELECT  
  
    day,  
  
    paid_sum,  
  
    SUM(paid_sum) OVER (ORDER BY day) AS paid_cumulative  
  
FROM daily_paid  
  
ORDER BY day.
```