

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**Західноукраїнський національний університет**  
**Факультет комп'ютерних інформаційних технологій**  
**Кафедра інформаційно–обчислювальних систем і управління**

**КОНЦОГРАДА Світлана Михайлівна**

**Веб-орієнтована інформаційна система організації заходів у  
ресторанному бізнесі / Web-based information system of organizing events  
in the restaurant business**

**спеціальність: 122 – Комп'ютерні науки**  
**освітньо–професійна програма – Комп'ютерні науки**

**Кваліфікаційна робота**

Виконала студентка групи КН–42  
С.М. Концограда

---

Науковий керівник  
к.т.н., доц. І.В. Турченко

---

Кваліфікаційну роботу допущено до  
захисту «\_\_\_» \_\_\_\_\_ 20\_\_\_ р.

В.о. завідувача кафедри  
\_\_\_\_\_ Н.В. Дзюбановська

**ТЕРНОПІЛЬ – 2026**

**Факультет комп'ютерних інформаційних технологій**  
Кафедра інформаційно-обчислювальних систем і управління  
Освітній ступінь «бакалавр»  
спеціальність: 122 – Комп'ютерні науки  
освітньо-професійна програма – Комп'ютерні науки

ЗАТВЕРДЖУЮ  
В.о. завідувача кафедри  
\_\_\_\_\_ Н.В. Дзюбановська  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ  
КОНЦОГРАДІ Світлані Михайлівні**

---

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи

Веб-орієнтована інформаційна система організації заходів у ресторанному бізнесі / Web-based information system of organizing events in the restaurant business

керівник роботи к.т.н., доцент І.В. Турченко

затверджений наказом по університету від 24 квітня 2026 року № 266.

2. Строк подання студентом закінченої кваліфікаційної роботи 25 травня 2026 р.

3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.

4. Основні питання, які потрібно розробити

- провести аналіз предметної області та огляд існуючих веб-орієнтованих інформаційних систем організації заходів у ресторанному бізнесі, виявити їхні переваги та недоліки;

- дослідити особливості автоматизації процесу оформлення заходів та взаємодії клієнта із системою;

- спроектувати структуру веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі;

- спроектувати інформаційне забезпечення системи;

- реалізувати веб-орієнтовану інформаційну систему організації заходів у ресторанному бізнесі;

- провести тестування програмної системи та оцінити коректність її роботи.

5. Перелік графічного матеріалу у роботі

- Схема алгоритму оформлення замовлення

- ER-діаграма бази даних інформаційної системи

- UML-діаграма станів

- UML-діаграма класів

- Діаграма прецедентів

## 6. Консультанти розділів кваліфікаційної роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата   |                  |
|--------|-------------------------------------------|----------------|------------------|
|        |                                           | Завдання видав | Завдання прийняв |
|        |                                           |                |                  |
|        |                                           |                |                  |
|        |                                           |                |                  |

7. Дата видачі завдання 01 грудня 2025 р.

## КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів кваліфікаційної роботи                                                                                      | Строк виконання етапів кваліфікаційної роботи | Примітка |
|-------|--------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|----------|
| 1     | Огляд літературних джерел відповідно до предметної області та складання плану роботи                                     | до 01.01.2026 р.                              |          |
| 2     | Написання 1 розділу кваліфікаційної роботи                                                                               | до 01.02.2026 р.                              |          |
| 3     | Написання 2 розділу кваліфікаційної роботи                                                                               | до 15.03.2026 р.                              |          |
| 4     | Написання 3 розділу кваліфікаційної роботи                                                                               | до 01.05.2026 р.                              |          |
| 5     | Представлення попереднього варіанту кваліфікаційної роботи, перевірка та внесення змін керівником                        | до 15.05.2026 р.                              |          |
| 6     | Опрацювання зауважень та представлення завершеного варіанту кваліфікаційної роботи. Підготовка супроводжуючих документів | до 25.05.2026 р.                              |          |
| 7     | Перевірка кваліфікаційної роботи на оригінальність тексту                                                                | до 30.05.2026 р.                              |          |
| 8     | Оформлення кваліфікаційної роботи та отримання допуску до захисту                                                        | до 10.06.2026 р.                              |          |
| 9     | Подання кваліфікаційної роботи до захисту на засіданні атестаційної комісії                                              | до 10.06.2026 р.                              |          |

Студент \_\_\_\_\_ С.М. Концограда  
підпис

Керівник роботи \_\_\_\_\_ к.т.н., доцент І.В. Турченко  
підпис

## АНОТАЦІЯ

Кваліфікаційна робота на тему «Веб-орієнтована інформаційна система організації заходів у ресторанному бізнесі» на здобуття освітнього ступеня «бакалавр» зі спеціальності 122 «Комп'ютерні науки» освітньої програми «Комп'ютерні науки» написана обсягом в 65 сторінок і містить 25 ілюстрацій, 3 додатків та 27 використаних джерел.

Метою кваліфікаційної роботи є проєктування і реалізація веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі.

Об'єктом дослідження є процеси отримання, зберігання, обробки, передачі та використання інформації в системах організації заходів у ресторанному бізнесі.

Результатом роботи є веб-орієнтована інформаційна система організації заходів у ресторанному бізнесі, що включає користувацький веб-інтерфейс, серверну частину обробки даних та базу даних для збереження інформації. Система забезпечує автоматизацію основних бізнес-процесів ресторанної діяльності, наприклад організацію заходів, бронювання залів, самостійне формування меню, управління замовленнями та взаємодію з користувачами.

Результати роботи можуть бути використані у діяльності ресторанних закладів для підвищення якості обслуговування клієнтів, оптимізації роботи персоналу та забезпечення швидкої адаптації до змін ринкового середовища.

Ключові слова: ВЕБ-ОРІЄНТОВАНА ІНФОРМАЦІЙНА СИСТЕМА, РЕСТОРАННИЙ БІЗНЕС, ЗАМОВЛЕННЯ, БІЗНЕС-ПРОЦЕСИ, СЕРВЕРНА ЧАСТИНА, БАЗА ДАНИХ.

## ANNOTATION

The qualification work entitled “Web-based information system of organizing events in the restaurant business”, submitted for the Bachelor’s degree in Specialty 122 “Computer Science” under the Educational Program “Computer Science”, consists of 65 pages, 25 figures, 3 appendices, and 27 references.

The aim of the qualification work is to design and implement a web-oriented information system for event management in the restaurant business.

The object of the study is the processes of obtaining, storing, processing, transmitting, and using information in event management systems within the restaurant business.

The result of the work is a web-oriented information system for event management in the restaurant business, which includes a user web interface, a server-side data processing component, and a database for storing information. The system provides automation of the main business processes of restaurant operations, including event organization, hall reservation, independent menu creation, order management, and interaction with users.

The results of the work can be used in restaurant establishments to improve the quality of customer service, optimize staff performance, and ensure rapid adaptation to changes in the market environment.

Keywords: WEB-ORIENTED INFORMATION SYSTEM, RESTAURANT BUSINESS, ORDER, BUSINESS PROCESSES, SERVER-SIDE COMPONENT, DATABASE.

## ЗМІСТ

|                                                                                                                                     |    |
|-------------------------------------------------------------------------------------------------------------------------------------|----|
| Вступ.....                                                                                                                          | 7  |
| 1 Аналіз предметної області, існуючих рішень та постановка задачі проектування.....                                                 | 9  |
| 1.1 Опис предметної області та аналіз бізнес-процесів організації заходів у ресторанному бізнесі.....                               | 9  |
| 1.2 Огляд і аналіз існуючих аналогів .....                                                                                          | 16 |
| 1.3 Постановка задачі проектування веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі .....          | 18 |
| 2 Алгоритмічне та інформаційне забезпечення веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі ..... | 21 |
| 2.1 Загальна структура веб-орієнтованої інформаційної системи .....                                                                 | 21 |
| 2.2 Алгоритмічне забезпечення системи .....                                                                                         | 25 |
| 2.3 Інформаційне забезпечення системи.....                                                                                          | 28 |
| 3 Реалізація веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі .....                                | 37 |
| 3.1 Програмна реалізація інформаційної системи .....                                                                                | 37 |
| 3.2 Інтерфейс користувача системи.....                                                                                              | 43 |
| 3.3 Тестування розробленої системи.....                                                                                             | 47 |
| Висновки .....                                                                                                                      | 51 |
| Список використаних джерел .....                                                                                                    | 52 |
| Додаток А Фрагмент програмного коду серверної частини інформаційної системи .....                                                   | 55 |
| Додаток Б Фрагмент програмної реалізації користувацького інтерфейсу системи .....                                                   | 57 |
| Додаток В Копія опублікованих результатів .....                                                                                     | 59 |

## ВСТУП

У сучасному світі розвиток інформаційних технологій активно використовують у різних сферах діяльності людини. Однією з таких сфер є ресторанний бізнес, який постійно розвивається і потребує використання сучасних програмних рішень для підвищення продуктивності роботи закладів. Використання інформаційних систем дозволяє оптимізувати багато процесів, пов'язаних з сферою обслуговування.

Важливу роль у розвитку ресторанної сфери відіграють веб-орієнтовані сервіси. Вони дозволяють клієнтам швидко і зручно організовувати заходи, сформувати замовлення, бронювати столики або отримувати інформацію про послуги ресторану. Використання таких сервісів покращує взаємодію між клієнтом та рестораном, а також підвищує швидкість та якість обслуговування.

У багатьох закладах громадського харчування процес оформлення замовлень або формування меню для заходів може виконуватися вручну або за допомогою простих програм. Такі способи роботи не завжди є зручними та займають багато часу, і можуть виникати помилки під час передачі інформації між клієнтом та працівниками закладу. Також, відсутність єдиної системи зберігання інформації ускладнює організацію роботи ресторану та аналіз замовлень.

Використання сучасних програмних рішень допомагає зробити процес оформлення замовлень швидшим і зручнішим. Одним із таких рішень є інформаційні системи, які забезпечують взаємодію між різними частинами системи.

Це дозволяє організувати обмін даними між клієнтською частиною сервісу, серверною логікою та базою даних, що забезпечує швидку роботу всієї системи.

Використання таких систем є основою для сучасних веб-сервісів. Це дозволяє створювати програми, які легко поєднуються з іншими сервісами, наприклад, оплатою онлайн, доставкою або сповіщеннями. Для ресторанного бізнесу це дає можливість швидко розвивати цифрові послуги.

Актуальність даної роботи полягає у необхідності створення рішень, які дозволяють спростити процес організації заходу, бронювання столиків чи

оформлення меню у ресторанній сфері. Впровадження таких систем сприяє підвищенню швидкості обробки замовлень, зменшенню кількості помилок та покращенню організації роботи закладу.

Метою роботи є розробка веб-орієнтованої інформаційної системи для організації заходів у ресторанному бізнесі. Для досягнення мети необхідно виконати наступні завдання:

- опис предметної області та аналіз бізнес-процесів організації заходів у ресторанному бізнесі;
- огляд і аналіз існуючих аналогів;
- постановка задачі проєктування веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі;
- загальна структура веб-орієнтованої інформаційної системи;
- алгоритмічне забезпечення системи;
- інформаційне забезпечення системи;
- програмна реалізація інформаційної системи;
- інтерфейс користувача системи;
- тестування розробленої системи.

Об'єктом дослідження є процеси отримання, зберігання, обробки, передачі та використання інформації в системах організації заходів у ресторанному бізнесі.

Практичне значення розробленої веб-орієнтованої інформаційної системи полягає в автоматизації процесів організації заходів у ресторанному бізнесі, спрощенні взаємодії між клієнтами та персоналом закладу. Результати роботи можуть бути використані у діяльності ресторанних закладів для оптимізації процесу оформлення замовлень, спростити взаємодію між клієнтом і закладом та підвищити управління ресторанными послугами.

Результати роботи опубліковано в матеріалах міжнародної науково-практичної конференції студентів і молодих вчених «ICSPM 2026: Інтелектуальні обчислювальні системи та управління проєктами», м. Тернопіль, 21–22 травня 2026 р. (додаток В).

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ, ІСНУЮЧИХ РІШЕНЬ ТА ПОСТАНОВКА ЗАДАЧІ ПРОЄКТУВАННЯ

## 1.1 Опис предметної області та аналіз бізнес-процесів організації заходів у ресторанному бізнесі

У сучасних умовах ресторанний бізнес все більше орієнтується на використання цифрових сервісів, щоб покращити взаємодію з гостями та оптимізацією внутрішніх процесів. Особливо актуальним це є для організації заходів тому, що такий процес включає велику кількість етапів, починаючи від інформації про гостя і завершуючи формуванням фінальної заявки та узгодженням деталей із адміністратором або директором ресторану. Без використання цифрових інструментів ці дії зазвичай виконуються вручну, і це призводить до втрати часу, помилок у розрахунках або незручностей як для гостей, так і для персоналу ресторану [1-2].

Предметною областю даної роботи є процес організації та управління заходами у ресторанному бізнесі, який включає оформлення заходів, формування меню, облік клієнтів, взаємодію між адміністрацією ресторану і клієнтами.

Наявність веб-орієнтованої системи забезпечує зручний спосіб взаємодії між клієнтами і рестораном, це дозволяє клієнту самостійно ввести необхідні дані, обрати параметри заходу, сформувавши меню та отримати попередній розрахунок вартості. Для ресторану означає, що це структурований збір інформації, збереження даних у єдиній системі та можливість їх подальшого аналізу.

Процес організації заходу в рамках цифрового сервісу включає декілька основних функцій. Насамперед це введення персональних даних гостей, таких як ім'я, контактний номер телефону, електронну пошту. Ця інформація є необхідною для подальшого зворотного зв'язку з клієнтом, та підтвердження заявки. Далі гість задає основні параметри заходу, тобто дату та час проведення, кількість гостей і вказує який захід буде проводитися (весілля, день народження, корпоратив).

Важливою частиною розробленої інформаційної системи є функція автоматичного підбору залу. Замість простого списку, система самостійно виконує аналіз запиту користувача: тобто, він порівнює кількість гостей та тип заходу з технічними характеристиками залів у базі даних.

Процес працює за алгоритмом перевірки відповідності, якщо кількість запрошених перевищує місткість залу, система автоматично відфільтровує такий варіант, і пропонує лише ті зали, де гостям буде комфортно. Цей етап є аналітичною складовою бізнес-процесу тому, що система самостійно обробляє цифрові показники, і цим звільняє адміністратора від ручних перевірок.

Наступний етап – це формування меню заходу. Інформаційна система дозволяє гостям вибрати страви з різних категорій, обирати їх кількість, а також залишати побажання до окремих позицій. Усі ці дані зберігаються в системі та використовуються для подальших розрахунків. Саме на цьому етапі активно задіяна транзакційна складова процесу, оскільки відбувається збір та накопичення даних про замовлені страви, їх кількість та вартість [3].

На основі сформованого меню система автоматично виконує розрахунок загальної вартості заходу. Автоматизація цього етапу дозволяє уникнути арифметичних помилок і це забезпечує прозорість фінансових розрахунків для гостей.

Завершальним етапом є формування та надсилання заявки. Усі зібрані дані об'єднуються в єдину структуру і передаються на сервер ресторану. Таким чином забезпечується швидка передача інформації між клієнтською частиною системи та серверною частиною системи. Адміністратор ресторану отримує готову заявку з усіма деталями заходу, і це спрощує подальшу комунікацію з клієнтом.

Аналітична складова дозволяє ресторану використовувати зібрані дані для аналізу діяльності. Наприклад, дослідити популярність певних страв чи проаналізувати кількість замовлень за певний період часу. Такий багатоаспектний аналіз створює інформаційну основу для прийняття управлінських рішень, спрямованих на покращення сервісу та підвищення продуктивності роботи ресторану.

Також використання веб-орієнтованої інформаційної системи формування меню для заходів дозволяє спростити взаємодію між гостями і рестораном. Тобто, гість отримує можливість самостійно керувати процесом створення замовлення, не звертаючись одразу до адміністратора. Це особливо зручно в умовах сучасного темпу життя, коли люди часто не мають можливості приїхати до ресторану

особисто або витратити час на телефонні консультації. Це надає доступ до всієї необхідної інформації в будь-який час, і це підвищує загальний рівень сервісу.

З точки зору для ресторану, впровадження такого цифрового рішення дозволяє організувати внутрішні бізнес-процеси. Уся інформація про замовлення зберігається в електронному вигляді, що зменшує ризик втрати даних та помилок при їх обробці. Крім того, дані, отримані від гостей, можуть використовуватись не лише для виконання конкретного замовлення, але й для подальшого аналізу та планування роботи закладу.

Важливою особливістю предметної області є те, що організація заходів складається як з послідовних, так і з взаємопов'язаних етапів. Наприклад, оформлення меню залежить від кількості гостей та типу заходу, а підбір залу – від дати та часу проведення. Тому система повинна враховувати ці залежності та правильно обробляти введені дані. Саме це відіграє головну роль у забезпеченні взаємодії між різними частинами системи та збереженні цілісності інформації [4].

Транзакційна складова алгоритму обробки замовлення відповідає за збереження всіх дій користувача в системі. До таких дій належать введення персональних даних, вибір параметрів заходу, додавання страв до меню, а також підтвердження замовлення. Кожна з цих операцій супроводжується збереженням відповідних даних у базі, що дозволяє в будь-який момент відновити повну інформацію сформованого замовлення. Надійність транзакційної складової є важливою, тому що, будь-яка помилка може призвести до некоректного виконання замовлення.

Аналітична складова алгоритму обробки замовлення забезпечує можливість роботи з накопиченими даними. Вона дозволяє аналізувати статистику замовлень, виявляти найбільш популярні страви, оцінювати сезонні коливання попиту та визначати найбільш завантажені дні й години. На основі такого аналізу ресторан може приймати обґрунтовані рішення щодо оновлення меню або коригування цін.

Крім того, аналітична складова може використовуватись для оцінки поведінки клієнтів. Наприклад, аналіз повторних замовлень дозволяє визначити постійних клієнтів а вивчення типів заходів – зрозуміти, які формати подій є

найбільш затребуваними. Це створює передумови для подальшого розвитку персоналізованих пропозицій та підвищення конкурентоспроможності ресторану.

Інформаційна система також сприяє підвищенню прозорості процесу організації заходу. Клієнт чітко бачить перелік обраних страв, їхню вартість та загальну суму замовлення. Це зменшує кількість непорозумінь і конфліктних ситуацій, які можуть виникати при усному узгодженні деталей. Прозорість і зрозумілість процесу позитивно впливають на довіру гостей до ресторану.

Таким чином, предметна область даної роботи охоплює багато взаємопов'язаних процесів, спрямованих на автоматизацію організації заходів у ресторанному бізнесі. Використання системи забезпечує гнучкість, масштабованість та можливість подальшого розвитку. Це робить обрану тему актуальною і важливою в умовах цифрової організації у сфері обслуговування.

Окремо варто зазначити, що інформаційна система для ресторану повинна бути не лише функціональною, але й зручною у використанні. Якщо програмний інтерфейс є складним або перевантаженим, то зайвими діями, клієнт може відмовитися від оформлення заявки ще на початковому етапі. Тому при формуванні системи дуже важливо продумати логіку розміщення полів, послідовність кроків та зрозумілі підказки для користувача. Інтерфейс має бути інтуїтивно зрозумілим, щоб навіть людина, яка не має знань у сфері технологій, могла легко створити замовлення. Таким чином, предметна область охоплює не лише технічну реалізацію через прикладний програмний інтерфейс, а й організацію зручної взаємодії між людиною та системою.

Ще одним важливим етапом є забезпечення надійності та безпеки даних. Оскільки клієнт вводить персональну інформацію, зокрема ім'я та контактні дані, система повинна гарантувати їх збереження та захист. Дані повинні передаватися на сервер у структурованому вигляді та зберігатися в базі даних без втрат і спотворень. У цьому контексті система виконує роль посередника між клієнтською частиною та серверною частиною системи. Відбувається передача, обробка та збереження інформації. Це дозволяє чітко розмежувати функції інтерфейсу та логіки обробки даних, що підвищує стабільність і масштабованість системи.

Крім того, важливо розуміти, що цифровий сервіс у ресторанному бізнесі може розширюватися та вдосконалюватися з часом. Наприклад, у майбутньому можливо додати функцію онлайн-оплати або автоматичне підтвердження бронювання. Саме тому при формуванні цифрового сервісу необхідно закладати можливість гнучкого розвитку системи. Предметна область в даній роботі розглядається не як статична структура, а як динамічна система, яка може адаптуватися до змін потреб клієнтів і ринку ресторанних послуг.

Отже, описаний процес формування заходу через веб-орієнтовану інформаційну систему включає як транзакційну складову, що забезпечує збір, збереження та обробку кількісних даних, так і аналітичну складову, яка дозволяє проводити аналіз цих даних. Такий підхід забезпечує інформаційну підтримку прийняття управлінських рішень та сприяє підвищенню продуктивної діяльності ресторану [5].

Для кращого розуміння структури та логіки описаного бізнес-процесу доцільно представити його у вигляді діаграми дерева функцій, що представлена на рисунку 1.1. Така діаграма дозволяє наочно відобразити головну функцію системи та всі підпорядковані їй функції та підфункції. Вона показує, як загальна мета системи розкладається на окремі етапи та операції, між якими існують логічні зв'язки. Використання діаграми дерева функцій спрощує сприйняття структури системи та допомагає чітко визначити місце кожної функції в межах загального бізнес-процесу. Саме тому в межах даної роботи побудова такої діаграми є важливим етапом моделювання предметної області.

Дана діаграма відображає структуру процесу та показує, як основна функція інформаційної системи розкладається на підпорядковані функції та операції. Головною функцією системи є формування замовлення клієнтом у зручному онлайн-форматі. Саме від цієї функції відходять усі інші складові процесу, які забезпечують повноцінну роботу системи.

Першою групою функцій є введення та оброблення даних про захід. До цієї групи належать дії, пов'язані з вибором дати та часу проведення події, введення кількості гостей, а також визначенням типу заходу. Дані операції є обов'язковими, оскільки вони впливають на подальші етапи формування замовлення, зокрема на

вибір залу та рекомендації щодо меню. Усі введені дані передаються до системи через інтерфейс користувача та зберігаються для подальшої обробки.

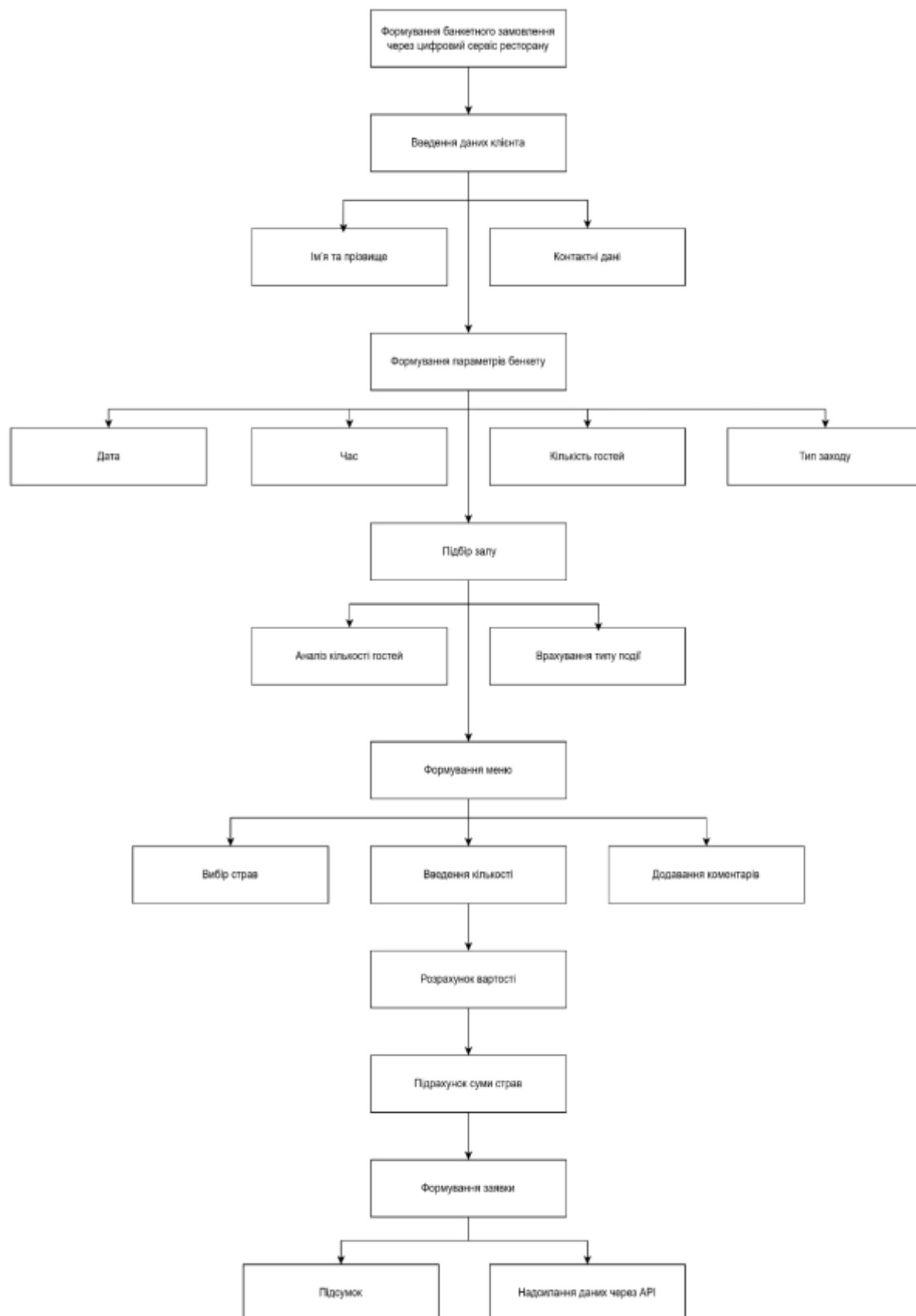


Рисунок 1.1 – Діаграма дерева функцій цифрового сервісу

Наступною важливою функціональною групою є підбір та вибір залу. На цьому етапі система аналізує введену кількість гостей та тип заходу і на основі цих параметрів пропонує доступні зали ресторану. Додатково може формуватися рекомендація щодо найбільш оптимального залу, що дозволяє спростити процес прийняття рішення для клієнта.

Окрему гілку діаграми становить формування меню заходу. Ця функція включає перегляд переліку доступних страв, вибір позицій, визначення кількості кожної страви та автоматичний розрахунок загальної вартості замовлення. На цьому етапі інформаційна система збирає кількісні показники, які використовуються для подальшого аналізу популярності страв і фінансових результатів. Формування меню є головною частиною, оскільки саме від нього залежить кінцева вартість заходу.

Окремо в діаграмі виділена функція введення контактних даних клієнта. Вона передбачає введення імені та прізвища, номер телефону, а також електронної пошти. Ці дані необхідні для зворотного зв'язку з клієнтом та подальшої обробки заявки адміністратором ресторану. Зібрана інформація передається на сервер через програмний інтерфейс програми та зберігається в базі даних.

Завершальною функцією є формування підсумку замовлення та передача заявки на сервер. На цьому етапі гість може переглянути всі введені дані, складене меню та загальну вартість заходу. Після підтвердження замовлення інформація надсилається до серверної частини системи, де вона використовується для подальшої роботи персоналу ресторану. Таким чином, діаграма дерева функцій показує послідовність та взаємозв'язок усіх етапів цифрового сервісу.

Використання діаграми дерева функцій дозволяє виділити основні та допоміжні функції, а також чітко визначити межі транзакційної та аналітичної складових. Це створює основу для подальшого проектування архітектури інформаційної системи та реалізації цифрового сервісу в ресторанному бізнесі.

Таким чином, побудована діаграма дерева функцій дозволяє наочно відобразити логіку роботи сервісу для ресторанного бізнесу та показати взаємозв'язок між основними етапами формування замовлення. Представлена

структура підтверджує взаємозв'язок усіх процесів. Система послідовно обробляє дані користувача, формує замовлення та видає підсумковий результат

Отже, опис предметної області та розроблене дерево функцій підтверджують доцільність використання цифрового сервісу [6] в ресторанному бізнесі для замовлення заходів. Запропонований підхід забезпечує оптимізацію процесів, зменшує навантаження на персонал та підвищує зручність взаємодії клієнтів із рестораном, що є важливим етапом у створенні сучасного цифрового сервісу.

## 1.2 Огляд і аналіз існуючих аналогів

На сучасному етапі розвитку цифрових технологій ресторанний бізнес активно впроваджує онлайн-сервіси для автоматизації процесів обслуговування клієнтів. Однак детальний аналіз ринку показує, що повноцінних веб-сервісів, які дозволяють клієнту самостійно сформувати банкетне меню онлайн із вибором залу, страв та автоматичним розрахунком вартості, практично не існує. Більшість сайтів ресторанів обмежуються наданням загальної інформації, переглядом меню у вигляді списку PDF файлу або скануванням QR-коду, а також простою формою зворотного зв'язку для попереднього бронювання столику.

Сучасні ресторани мають офіційну веб-орієнтовану інформаційну систему, де розміщується загальна інформація про заклад, фотографії залів, приклади банкетного меню та контактні дані.

Зазвичай на таких сайтах [7-8], наприклад, присутня кнопка «Замовити банкет» або «Забронювати стіл», після натискання якої відкривається проста форма. Ця форма, як правило, містить поля для введення імені, номера телефону, дата час та орієнтовної кількості гостей. Після заповнення форми інформація надсилається адміністратору ресторану, який зв'язується з клієнтом для подальшого узгодження деталей. Таким чином, фактичне формування меню відбувається вже поза межами сайту – телефоном або при особистій зустрічі. Нижче наведено приклад таких веб-сайтів (рисунки 1.2 – 1.3).

The screenshot shows the 'Самогонна Ресторация' website with a 'Замовити столик' (Book a table) form. The form includes fields for 'Ім'я' (Name), 'Телефон' (Phone), 'Дата' (Date) set to 26.09.2020, 'Час' (Time) set to 20:00, and 'Кількість гостей' (Number of guests) set to 2. There are also checkboxes for 'Важко чи важко' (Difficult or not) and 'Важко чи важко' (Difficult or not). A 'Замовити' (Book) button is at the bottom.

Рисунок 1.2 – Сторінка веб-орієнтованої інформаційної системи ресторану “Самогонна Ресторация” для замовлення столика

The screenshot shows the 'Steak House' website with a 'ЗАРЕЗЕРВУВАТИ СТОЛИК' (Reserve a table) form. The form includes fields for 'Ім'я' (Name), 'Телефон' (Phone), 'Дата' (Date) set to 26.09.2020, and 'Час' (Time) set to 20:00. There is a 'Забронювати' (Book) button at the bottom. The background features a large image of a steak and a 'НАЙКРАЩЕ Власної ви' (Best Own) banner.

Рисунок 1.3 – Сторінка веб-орієнтованої інформаційної системи ресторану ресторану “Steak House” для резервування столика

Інтерфейс таких форм зазвичай є мінімалістичним: кілька текстових полів, кнопка відправлення та коротке повідомлення про успішну передачу заявки. З точки зору оптимізації бізнес-процесів це рішення є частковим, оскільки воно лише фіксує початковий запит, але не дозволяє здійснити повний цикл формування банкетного замовлення в онлайн-режимі. Відсутня можливість одразу обрати конкретні страви, розрахувати їхню кількість або побачити кінцеву суму замовлення.

Такий підхід не забезпечує автоматизацію процесу формування меню та не дозволяє клієнту одразу отримати інформацію про вартість заходу. У ході аналізу

також було встановлено, що навіть у сучасних ресторанах відсутні сервіси, де користувач може самостійно обрати тип заходу, отримати рекомендацію щодо залу, вибрати конкретні страви, задати їх кількість та одразу побачити загальну суму замовлення. Саме тому було прийнято рішення розробити власну веб-орієнтовану інформаційну систему, яка поєднує всі ці функції в одному веб-інтерфейсі.

І саме тому у межах даної кваліфікаційної роботи була розроблена веб-орієнтована система для організації заходів у ресторанному бізнесі.

### 1.3 Постановка задачі проєктування веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі

На основі проведеного аналізу предметної області та огляду існуючих рішень було встановлено, що процес формування замовлення у більшості ресторанів залишається частково автоматизованим або взагалі немає повноцінної цифрової підтримки. Як правило, клієнт залишає заявку через просту форму зворотного зв'язку, після чого всі деталі узгоджуються телефоном або під час особистої зустрічі. Такий підхід не дозволяє забезпечити облік даних, створює додаткове навантаження на персонал та не дає можливості проводити подальший аналіз отриманої інформації.

На основі аналізу було сформульовано загальну концепцію. Вона полягає у створенні веб-орієнтованої інформаційної системи, яка дозволяє клієнту самостійно організувати захід в онлайн-режимі, вибрати параметри заходу, зал та меню, отримати автоматичний розрахунок вартості та передати структуровані дані на сервер. Такий підхід дозволяє поєднати транзакційну складову (збір і збереження даних) з аналітичною складовою (можливість подальшого аналізу замовлень).

Метою роботи є розробка інформаційної системи для організації заходів у ресторанному бізнесі. Це підвищує стабільність роботи ресторану та зручність для клієнтів.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- опис предметної області та аналіз бізнес-процесів організації заходів у ресторанному бізнесі;
- огляд і аналіз існуючих аналогів;
- постановка задачі проєктування веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі;
- загальна структура веб-орієнтованої інформаційної системи;
- алгоритмічне забезпечення системи;
- інформаційне забезпечення системи;
- програмна реалізація інформаційної системи;
- інтерфейс користувача системи;
- тестування розробленої системи.

Веб-орієнтована інформаційна система має спростити роботу із замовленнями. Замість ручного узгодження меню замовлення буде формуватися автоматично. Це допоможе швидше обробляти заявки, зменшить кількість помилок під час передачі інформації та дозволить зберегти всі дані про замовлення в одному місці. Також система дасть можливість аналізувати статистику замовлень і зробить процес оформлення більш зручним для клієнтів.

Інформаційна система повинна реалізовувати такі основні функції, як надання користувачу можливості введення основних параметрів заходу (дата, час, кількість гостей, тип події). Автоматичний підбір залу відповідно до кількості гостей та типу заходу. Відображення переліку доступних страв із можливістю вибору кількості кожної позиції. Автоматичний розрахунок вартості замовлення в режимі реального часу. Введення та збереження контактних даних клієнта. Передача структурованих даних через програмний інтерфейс на сервер. Збереження інформації в базі даних для подальшої обробки та аналізу.

Крім функціональних характеристик, інформаційна система повинна відповідати таким вимогам:

- безпосередньо зрозумілий та простий інтерфейс;
- адаптивність для роботи на мобільних пристроях;
- достатня швидкість обробки запитів;
- забезпечення цілісності та збереження даних;

- можливість масштабування системи в майбутньому;
- модульна архітектура для зручності розширення функціоналу.

Для вирішення цих завдань система побудована на базі архітектури клієнт-сервер. Користувач взаємодіє з веб-інтерфейсом (frontend), який через прикладний програмний інтерфейс передає дані на сервер (backend). Де дані обробляються та зберігаються в базі даних. Така структура дозволяє розділити логіку інтерфейсу та логіку обробки даних, і це підвищує надійність та гнучкість системи.

Для візуалізації логіки роботи інформаційної системи у межах підрозділу доречно використати діаграму прецедентів (англ. Use Case Diagram), яка наведена на рисунку 1.4.

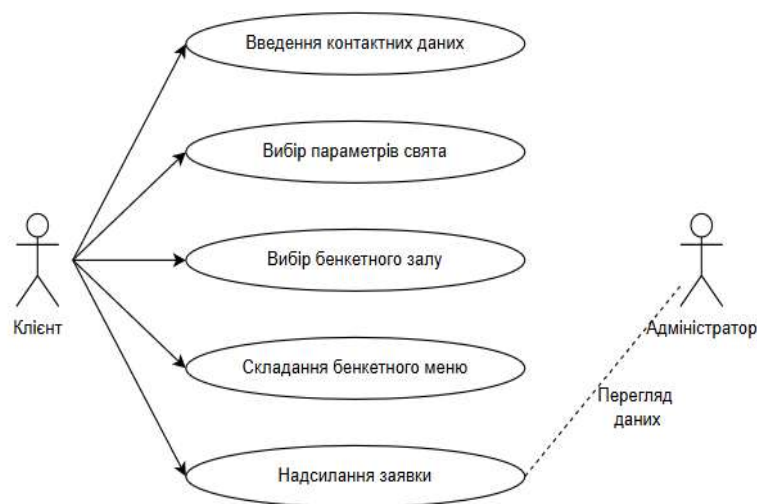


Рисунок 1.4 – Діаграма прецедентів

На діаграмі прецедентів основним користувачем є клієнт, який через веб-орієнтовану інформаційну систему оформляє заявку для проведення заходу. Спочатку клієнт вводить контактні дані, щоб адміністратор закладу міг зв'язатися з ним для уточнення деталей. Наступним кроком клієнта – це визначення параметрів заходу, наприклад тип події, кількість гостей, дата та час. Після цього він обирає зал, який підходить для проведення заходу. Далі клієнт формує меню, обираючи страви та напої. Після заповнення всієї інформації заявка зберігається в базі даних. І адміністратор має можливість переглядати дані, тобто бачити створені клієнтами заявки та працювати з інформацією про замовлення.

## 2 АЛГОРИТМІЧНЕ ТА ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОРГАНІЗАЦІЇ ЗАХОДІВ У РЕСТОРАННОМУ БІЗНЕСІ

### 2.1 Загальна структура веб-орієнтованої інформаційної системи

Веб-орієнтована інформаційна система для організації заходів у ресторанному бізнесі призначена для покращення процесу оформлення заходу та складання меню. Основною метою розробки системи є спрощення взаємодії між клієнтом і рестораном, зменшення навантаження на персонал та зменшення помилок під час прийому замовлень.

Архітектура системи побудована за клієнт-серверним принципом, що передбачає розділення системи на окремі логічні рівні: рівень представлення (інтерфейс), рівень обробки логіки та рівень збереження даних. Такий підхід забезпечує гнучкість, масштабованість та можливість подальшого розширення функцій.

Схема архітектури веб-орієнтованої інформаційної системи для організації заходів у ресторанному бізнесі, побудована на основі прикладного програмного інтерфейсу взаємодії, представлена на рисунку 2.1.

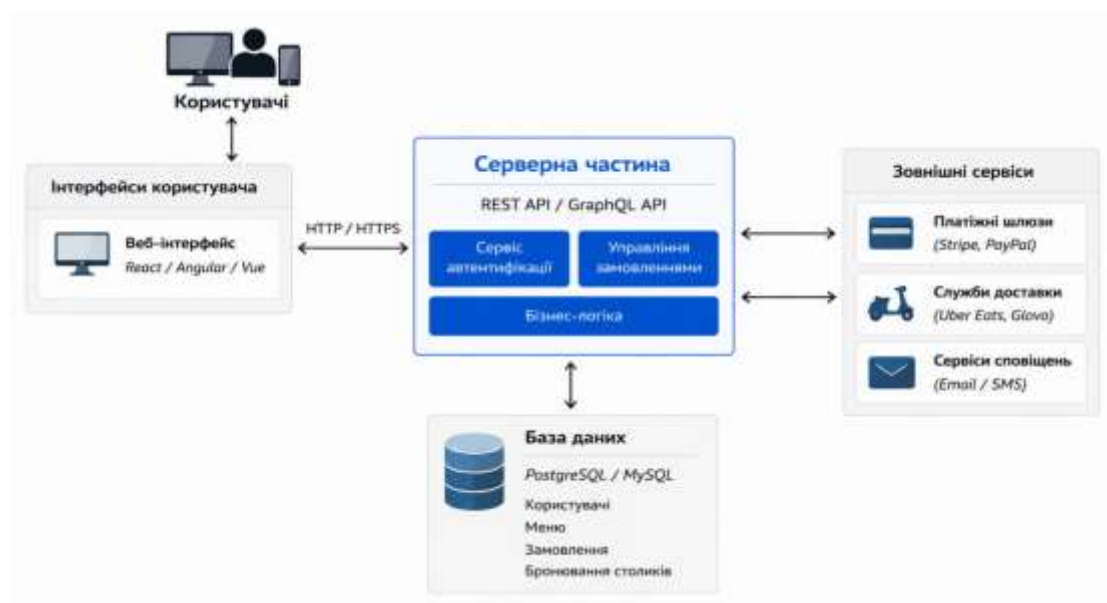


Рисунок 2.1 – Архітектура веб-орієнтованої інформаційної системи для організації заходів у ресторанному бізнесі

Архітектура системи складається з кількох основних компонентів, кожен з яких виконує окрему функцію у процесі роботи. Нижче наведено пояснення основних елементів запропонованої архітектури.

Frontend є клієнтською частиною системи, яка забезпечує взаємодію користувача з сервісом. Взаємодія з функціоналом інформаційної системи здійснюється через веб-сайт ресторану. Основними завданнями Frontend є відображення інтерфейсу користувача, введення даних та передача запитів до серверної частини системи через прикладний програмний інтерфейс [9]. Саме через клієнтський інтерфейс користувач може переглядати меню, вибирати страви, вводити параметри події або оформляти замовлення.

Прикладний програмний інтерфейс є центральним компонентом системи, який забезпечує взаємодію між клієнтською частиною та серверною логікою. Через нього клієнтські застосунки надсилають запити до системи та отримують відповіді у структурованому вигляді. До основних функцій прикладного програмного інтерфейсу належить обробка запитів від клієнтських застосунків, перевірка та обробка введених даних, взаємодія з базою даних для отримання або збереження інформації. Завдяки використанню програмного інтерфейсу система може легко інтегруватися з іншими сервісами та підтримувати різні типи клієнтських застосунків.

База даних [10] є компонентом системи, яка відповідає за зберігання всієї інформації, необхідної для роботи сервісу. У базі даних зберігаються різні типи даних, наприклад: інформація про меню ресторану, дані про замовлення клієнтів, інформація про користувачів, про бронювання. Зберігання інформації дозволяє швидко отримувати необхідні дані, і це забезпечує їх цілісність та полегшує подальший аналіз статистики замовлень.

Зовнішні сервіси – це сервіси, з якими може інтегруватися система для розширення функціональності. До таких сервісів можуть належати: платіжні системи для онлайн-оплати замовлень, сервіси сповіщень для інформування користувачів про статус замовлення. Інтеграція з такими сервісами дозволяє підвищити зручність використання системи та розширити її можливості.

Отже, запропонована архітектура інформаційної системи будується за принципом клієнт-сервер [11], де кожен компонент виконує окрему функцію:

- frontend – забезпечує інтерфейс взаємодії з користувачем;
- прикладний програмний інтерфейс – реалізує основну логіку роботи системи та обробку запитів;
- база даних – забезпечує зберігання та управління даними;
- зовнішні сервіси – відповідають за інтеграцію із зовнішніми системами.

Такий підхід дозволяє створити гнучку та масштабовану систему, яка може бути легко розширена у майбутньому та адаптована до потреб ресторанного бізнесу.

Веб-інтерфейс це клієнтська частина інформаційної системи. Простими словами, це той сайт або "обличчя" програми, яке бачить користувач у своєму браузері. Саме тут розташовані всі візуальні елементи: форми для введення тексту, кнопки, картинки страв та списки вибору.

Основні функції, які виконує система:

- показ меню та страв: програма демонструє назви та ціни страв у гарному вигляді, щоб клієнту було легко обирати;
- покрокова робота: зроблено так, щоб оформлення замовлення йшло поступово. Користувач не бачить усе одразу, а переходить від кроку до кроку. Це допомагає не заплутатися у великій кількості страв;
- перевірка помилок: сайт сам перевіряє, чи правильно заповнені поля. Наприклад, якщо людина забула вказати номер телефону або ввела забагато гостей для маленького залу, система одразу це підкаже;
- створення запитів до сервера: інтерфейс збирає всі обрані страви та контакти в один пакет і через прикладний програмний інтерфейс відправляє їх у хмарну базу даних;
- зворотний зв'язок: інформаційна система в реальному часі рахує суму замовлення, а в самому кінці видає повідомлення: "Ваше замовлення успішно надіслано".

Яким має бути інтерфейс:

– простим і зрозумілим: навіть якщо людина вперше зайшла на такий сайт, вона повинна без інструкцій зрозуміти, куди натискати;

– безпомилковим: логіка навігації продумана так, щоб зменшити помилки користувача. Наприклад, неможливо відправити порожню заявку без контактів.

Тобто, веб-інтерфейс – це не просто картинка, це робочий інструмент, який робить процес замовлення заходу швидким та легким.

Серверна частина забезпечує виконання основної бізнес-логіки системи. Саме на цьому рівні здійснюється обробка отриманої від користувача інформації та формування структури замовлення.

Основні функції серверної частини:

- прийом даних від клієнтської частини;
- перевірка та коректності отриманої інформації;
- розрахунок загальної вартості замовлення;
- формування об'єкта замовлення;
- збереження інформації у базі даних;
- передача результату обробки назад у веб-інтерфейс.

Серверна частина виконує роль центрального керуючого елемента системи та забезпечує узгодженість роботи всіх компонентів.

База даних є компонентом інформаційної системи, призначеним для збереження, структурування та подальшого використання інформації.

У базі даних зберігаються:

- контактні дані користувачів;
- параметри заходу (тип події, кількість гостей);
- перелік обраних страв;
- розрахована сума замовлення;
- дата створення заявки.

Наявність єдиної бази даних дозволяє:

- уникнути втрати інформації;
- здійснювати аналіз попиту;
- формувати статистику замовлень;
- забезпечувати швидкий доступ адміністратора до даних.

База даних [12] є логічно відокремленою від інтерфейсу, що підвищує безпеку та надійність системи.

Адміністратор є внутрішнім користувачем веб-орієнтованої інформаційної системи, який отримує доступ до збережених замовлень та здійснює їх подальшу обробку. Ним може бути менеджер ресторану або відповідальна особа.

Основні функції адміністратора:

- перегляд списку отриманих заявок;
- доступ до контактної інформації клієнтів;
- аналіз параметрів замовлення;
- використання даних для організації підготовки заходу.

Адміністратор не бере участі у процесі формування замовлення, але є кінцевим споживачем інформації, що генерується системою.

Взаємодія між елементами інформаційної системи здійснюється послідовно і логічно структуровано. Користувач через веб-інтерфейс передає інформацію на серверну частину. Сервер виконує обробку та зберігає дані в базі даних. Після цього адміністратор отримує доступ до збереженої інформації для подальших дій.

## 2.2 Алгоритмічне забезпечення системи

Алгоритмічне забезпечення системи встановлює логічну структуру функціонування інформаційної системи, послідовність обробки вхідних даних та механізми формування результатів [13]. В основі розробленої веб-орієнтованої системи лежить алгоритм покрокового формування замовлення, що дозволяє розбити складний процес підбору меню на послідовність простих логічних операцій.

Алгоритм роботи веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі описує цикл сесії користувача від моменту завантаження веб-інтерфейсу до підтвердження збереження даних у базі даних і складається з наступних кроків:

### 1. Завантаження головного веб-застосунку:

#### 1.1 користувач відкриває веб-інтерфейс у браузері;

1.2 виконується ініціалізація клієнтської частини (завантаження сторінки, стилів, скриптів).

2. Завантаження персоналізованого інтерфейсу:

2.1 система визначає роль користувача (адміністратор, клієнт);

2.2 відображається відповідна панель керування та функціонал.

3. Взаємодія з системою:

3.1 користувач виконує дії (введення деталей заходу, створення замовлення, редагування даних);

3.2 дані вводяться через форми веб-інтерфейсу.

4. Валідація даних на стороні клієнта:

4.1 перевірка коректності введених даних (формат, обов'язкові поля);

4.2 у разі помилки користувач отримує відповідне повідомлення.

5. Передача даних на сервер:

5.1 дані надсилаються через API-запит.

6. Обробка даних на сервері:

6.1 виконуються бізнес-правила (створення замовлення, розрахунки);

6.2 дані готуються до збереження.

7. Збереження у базі даних:

7.1 інформація записується у базу даних;

7.2 підтверджується успішність операції.

8. Формування відповіді системи:

8.1 сервер повертає результат операції (успіх/помилка).

9. Оновлення інтерфейсу користувача:

9.1 клієнтська частина відображає оновлені дані.

10. Завершення або продовження сесії:

10.1 користувач може виконувати нові операції;

10.2 після виходу – сесія завершується та токен стає недійсним.

Процес оформлення замовлення в системі представлений у вигляді схеми алгоритму на рисунку 2.2.



Рисунок 2.2 – Схема алгоритму оформлення замовлення

Крім створення замовлення, система повинна забезпечувати можливість управління вже існуючими замовленнями.

До основних процесів управління замовленням можна віднести:

- формування замовлення;
- редагування замовлення;
- видалення замовлення;
- надсилання інформації про замовлення.

Після створення замовлення користувач має можливість переглядати інформацію про нього. Система отримує дані із бази даних замовлень та відображає їх у вигляді списку попереднього замовлення.

У разі необхідності користувач може внести зміни до замовлення. Наприклад, змінити кількість гостей, додати або прибрати страви. Система перевіряє коректність введених даних і після підтвердження користувачем оновлює інформацію в базі даних.

Також передбачена можливість видалення замовлення. У цьому випадку інформаційна система отримує запит користувача, перевіряє його права доступу та видаляє відповідний запис із бази даних.

Таким чином, управління замовленнями забезпечує повний контроль над процесом роботи із банкетними замовленнями, що значно спрощує їх адміністрування та підвищує продуктивність роботи.

Розроблені алгоритми даних забезпечують:

- логічну структурованість системи;
- повноту обробки всіх можливих сценаріїв;
- контроль достовірності даних;
- точність фінансових розрахунків;
- можливість подальшого розширення функціоналу.

Алгоритмічне забезпечення відповідає вимогам побудови інформаційних систем та створює основу для подальшої програмної реалізації.

### 2.3 Інформаційне забезпечення системи

Інформаційне забезпечення веб-орієнтованої інформаційної системи ресторанного бізнесу – це основа, на якій тримається весь функціонал. Воно включає в себе не тільки саму базу даних, а й правила, за якими інформація потрапляє в систему, обробляється та видається користувачу.

Основна мета це – організувати дані так, щоб клієнт міг швидко забронювати зал, а адміністратор не загубив жодного замовлення.

До складу інформаційного забезпечення відноситься:

- вхідна інформація: це все, що вводять користувач. Сюди відносяться параметри заходу (дата, час, кількість гостей), вибір конкретного банкетного залу,

перелік страв із меню та контактні дані клієнта (ім'я, номер телефону, електронна пошта);

– вихідна інформація: це результат роботи системи. Наприклад, сформоване замовлення, загальна вартість святкування заходу, а також звіти для адміністратора про завантаженість залів на певні дати.

Процес створення інформаційної структури відбувався у три етапи. Перший це – концептуальне проєктування: тут призначені основні об'єкти (сутності) системи та зв'язки між ними. Було виділено ключові сутності: «Клієнт», «Замовлення», «Банкетний зал», «Страва» та «Категорія меню». Це дозволило побачити загальну картину без прив'язки до конкретних технологій

Наступний етап – логічне проєктування. Тут концептуальна модель була перетворена на реляційну структуру. Було визначено атрибути для кожної таблиці (наприклад, для страви це назва, ціна) та встановлено зв'язки (наприклад, одне «Замовлення» може містити багато «Страв»). Також було проведено нормалізацію даних, щоб уникнути дублювання та помилок.

Завершальний етап – фізична реалізація. Це перенесення моделі безпосередньо в середовище керування базами даних [14]. Для реалізації була обрана реляційна база даних, що забезпечує високу швидкість запитів та надійне зберігання інформації про замовлення.

Для того, щоб дані були достовірними, у системі налаштовані автоматичні перевірки. Наприклад, неможливо зберегти замовлення без контактного номера або вибрати страву, якої немає в меню. Крім цього, структура бази даних передбачає розмежування прав: клієнт бачить лише свої дані та меню, а адміністратор має повний доступ до керування всіма заявками.

### 2.3.1 Опис вхідної та вихідної інформації

У процесі роботи веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі обробляється значний обсяг інформації, що надходить від користувачів та формується в результаті роботи програмних модулів. До вхідної

інформації належать дані, які вводяться користувачем або адміністратором під час роботи з системою.

Основними видами вхідної інформації є:

- дані користувача, а саме ім'я, прізвище, номер телефону, електронна пошта;
- дані замовлення: тип події, дата та час проведення заходу, кількість гостей, вибір певного залу, список страв, кількість кожної страви;
- дані меню, а саме назва страви, категорія страви, короткий опис та ціна.

Вся вхідна інформація проходить перевірку на коректність перед збереженням у базі даних.

Вихідна інформація формується у процесі обробки вхідних даних. До основних видів вихідної інформації належать:

- сформоване банкетне замовлення;
- список доступних страв;
- розрахунок вартості замовлення;
- підтвердження створення замовлення;
- інформація для адміністратора про всі замовлення.

Вихідна інформація може відображатися у вигляді таблиць, форм або звітів у користувацькому інтерфейсі.

### 2.3.2 Концептуальне інфологічне проєктування

Концептуальне інфологічне проєктування є етапом створення моделі даних, яка відображає структуру предметної області без прив'язки до конкретної інформаційної системи управління базами даних. На цьому етапі визначаються основні інформаційні об'єкти предметної області, їх атрибути та зв'язки між ними. У межах веб-орієнтованої системи організації заходів у ресторанному бізнесі було визначено такі основні сутності:

- користувач;
- замовлення;
- страва;
- позиція замовлення;

– зал.

Локальна інфологічна модель описує структуру даних для окремих функціональних підсистем. У системі можна виділити такі локальні моделі: модель користувачів містить інформацію про зареєстрованих користувачів системи.

Основні атрибути:

- id;
- ім'я;
- прізвище;
- телефон;
- email;
- дата реєстрації.

Модель меню містить інформацію про страви.

Атрибути:

- id;
- назва страви;
- опис;
- категорія;
- ціна.

Модель замовлень містить інформацію про замовлення.

Атрибути:

- id замовлення;
- id користувача;
- тип події;
- кількість гостей;
- дата створення;
- загальна вартість.

Інфологічна модель об'єднує всі локальні моделі в єдину систему даних.

Основні зв'язки між сутностями:

- користувач створює замовлення (1:N);
- замовлення містить позиції замовлення (1:N);
- позиція замовлення пов'язана зі стравою (N:1);

– замовлення може бути прив’язане до конкретного залу (N:1).

Глобальна модель відображається за допомогою ER-діаграми, що представлена на рисунку 2.3).

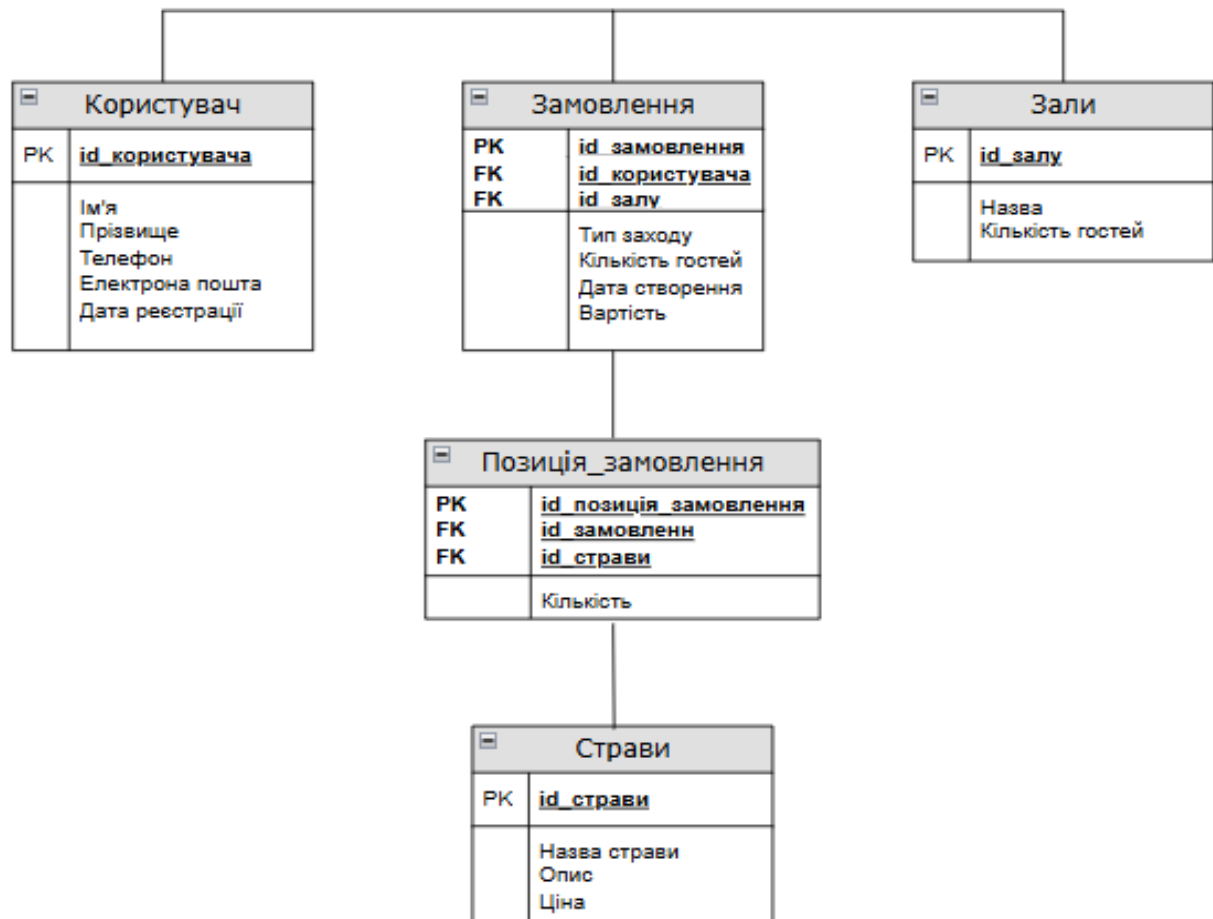


Рисунок 2.3 – ER-діаграма бази даних інформаційної системи

### 2.3.3 Проєктування даталогічної моделі даних

Даталогічна модель даних є логічним представленням структури бази даних, що базується на концептуальній інфологічній моделі та враховує особливості реляційної моделі даних. На цьому етапі здійснюється перехід від абстрактного опису предметної області до конкретної структури таблиць, їх атрибутів, ключів і зв'язків.

База даних інформаційної системи містить такі основні таблиці.

Таблиця «Користувачі» призначена для зберігання інформації про клієнтів, основними полями є – id (первинний ключ), ім'я, прізвище, телефон, email, дата\_реєстрації.

Таблиця «Страви» містить дані про меню, основними полями є – id (первинний ключ), назва\_страви, опис, категорія, ціна.

Таблиця «Зали» зберігає інформацію про доступні зали для проведення подій. Основними полями є – id (первинний ключ), назва, кількість гостей.

Таблиця «Замовлення» відображає інформацію про банкетні замовлення, основними полями є id (первинний ключ), id\_користувача (зовнішній ключ), id\_залу (зовнішній ключ), тип\_події, кількість\_гостей, дата\_створення, загальна\_вартість

Таблиця «Позиції\_замовлення» реалізує зв'язок між замовленнями та стравами. Основні поля – id (первинний ключ), id\_замовлення (зовнішній ключ), id\_страви (зовнішній ключ), кількість

У даталогічній моделі передбачено кілька зв'язків між таблицями. Таблиці «Користувачі» та «Замовлення» пов'язані за принципом один-до-багатьох, тобто один користувач може мати кілька замовлень. Цей зв'язок реалізується за допомогою поля id\_користувача.

Аналогічно, таблиці «Зали» та «Замовлення» також мають зв'язок один-до-багатьох: один зал може бути використаний у багатьох замовленнях. Таблиця «Замовлення» пов'язана з таблицею «Позиції замовлення» за принципом один-до-багатьох, оскільки одне замовлення може містити кілька позицій (страв).

Таблиця «Страви» також має зв'язок один-до-багатьох із таблицею «Позиції замовлення», адже одна страва може входити до різних замовлень.

Таким чином, через таблицю «Позиції замовлення» реалізується зв'язок багато-до-багатьох між замовленнями та стравами.

Спроектowana структура бази даних відповідає вимогам нормалізації: усі таблиці приведені до першої нормальної форми, оскільки містять атомарні значення; відсутні часткові залежності; відсутні транзитивні залежності.

Це забезпечує мінімізацію дублювання даних, підвищення цілісності бази даних та надійність її використання.

### 2.3.4 Проектування фізичної моделі даних

Проектування фізичної моделі даних є завершальним етапом розробки структури бази даних, на якому логічна модель перетворюється у конкретну реалізацію в середовищі системи керування базами даних. На цьому етапі визначаються таблиці, їх поля, типи даних, обмеження цілісності, первинні та зовнішні ключі, а також особливості збереження даних.

У розробленій інформаційній системі фізична модель даних реалізована у вигляді реляційної бази даних. Основною метою є забезпечення зручного, зберігання інформації про користувачів, замовлення, страви та зали.

У межах фізичної моделі було створено такі основні таблиці:

- таблиця «Користувачі» призначена для збереження інформації про користувачів, які оформляють замовлення. Вона містить такі поля: унікальний ідентифікатор користувача, ім'я, номер телефону та контактну інформацію;
- Таблиця «Зали» містить інформацію про доступні зали ресторану. До неї входять поля: ідентифікатор залу, назва залу та максимальна місткість;
- Таблиця «Страви» використовується для збереження даних про страви. Вона включає такі поля: ідентифікатор страви, назва, ціна, опис, зображення та категорія;
- Таблиця «Замовлення» є центральною у системі та містить інформацію про оформлені замовлення. Вона включає поля: ідентифікатор замовлення, ідентифікатор користувача, дата, час, кількість гостей, тип події, ідентифікатор залу та загальна вартість.

Для реалізації зв'язку між замовленням і стравами використовується проміжна таблиця Замовлення\_страв, яка дозволяє реалізувати зв'язок типу «багато–до–багатьох». Вона містить такі поля: ідентифікатор, ідентифікатор замовлення, ідентифікатор страви, кількість та коментар.

Усі таблиці пов'язані між собою за допомогою зовнішніх ключів. Наприклад, поле Користувач\_id у таблиці Замовлення є зовнішнім ключем, що посилається на таблицю Користувачі, а поле Зал\_id – на таблицю Зали. Таблиця Замовлення\_страви пов'язує таблиці Замовлення та Страви (рисунки 2. 4).

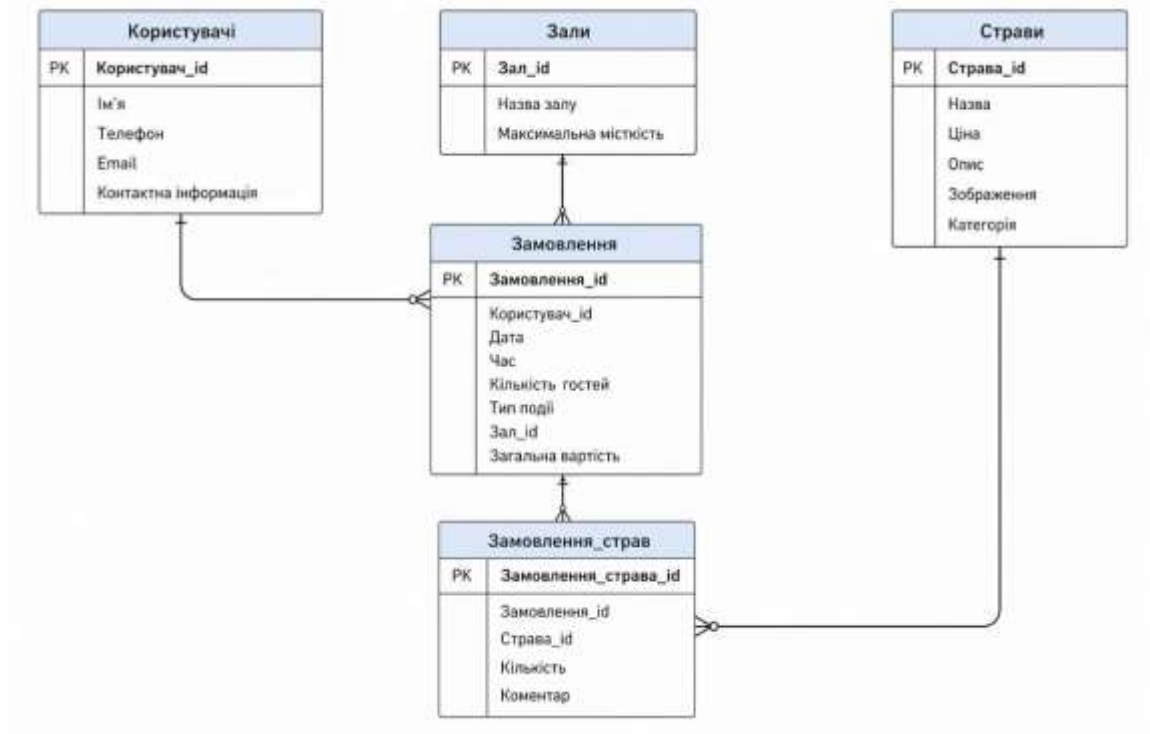


Рисунок 2.4 – Фізична модель бази даних системи організації заходів

Фізична модель також враховує можливість масштабування системи та обробки запитів, що є важливим для роботи з великими обсягами даних.

Таким чином, фізична модель даних забезпечує надійне збереження даних, обробку інформації в межах розробленої інформаційної системи, що є необхідною умовою для стабільної роботи програмного продукту.

### 2.3.5 Реалізація бази даних

Програмна реалізація бази даних передбачає створення структури за допомогою мови структурованих запитів SQL та її подальше використання в програмному середовищі. Було реалізовано базу даних в середовищі MySQL, яка складається з таких основних таблиць:

- користувачі;
- страви;
- зали ;
- замовлення ;
- позиції замовлення.

Створення таблиць здійснювалося за допомогою SQL-запитів типу CREATE TABLE, у яких визначаються структура таблиць, типи даних, первинні та зовнішні ключі.

Для роботи з базою даних використовувалися такі основні операції:

- додавання даних (INSERT) – для створення нових записів;
- отримання даних (SELECT) – для формування запитів і відображення інформації;
- оновлення даних (UPDATE) – для редагування існуючих записів;
- видалення даних (DELETE) – для видалення записів.

Таким чином, програмна реалізація бази даних забезпечує збереження, обробку та доступ до даних, необхідних для функціонування системи організації заходів.

### 3 РЕАЛІЗАЦІЯ ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ОРГАНІЗАЦІЇ ЗАХОДІВ У РЕСТОРАННОМУ БІЗНЕСІ

#### 3.1 Програмна реалізація інформаційної системи

Структурна схема веб-орієнтованої інформаційної системи показує внутрішню організацію. А саме як вхідні дані про замовлення проходять через блоки перевірки та обробки, і як кінцевий результат зберігається у відповідних таблицях бази даних. Користувач взаємодіє із системою через веб-інтерфейс, реалізований на основі React [15]. Усі введені дані (наприклад, інформація про захід, обрані страви, контактні дані) передаються на сервер за допомогою HTTP-запитів.

Серверна частина, реалізована з використанням Node.js, приймає запити, обробляє їх і виконує необхідну логіку [16], формування замовлення та збереження інформації. Для зберігання даних використовується база даних, у якій зберігається інформація про клієнтів, замовлення та страви. Таким чином, забезпечується чітке розділення відповідальності між компонентами системи та швидка обробка інформації.

UML-діаграма класів [17] відображає структуру веб-орієнтованої інформаційної системи та описує основні сутності, які беруть участь у формуванні організації заходу. У розробленій системі виділено кілька ключових класів, що реалізують основну логіку роботи системи (рисунки 3.1).

Основним класом є клас «Замовлення», де міститься інформація про дату, час проведення заходу, кількість гостей, який захід, обраний зал та загальну вартість заходу. Цей клас є центральним елементом системи, оскільки саме він об'єднує всі інші елементи.

Клас «Користувач» використовується для збереження контактної інформації клієнта, наприклад, ім'я, прізвище, номера телефону та спосіб зв'язку.

Клас «Страва» описує доступні позиції меню та містить назву, ціну та опис страви. Для реалізації вибору декількох страв використовується проміжний клас «ПозиціяЗамовлення», який дозволяє зберігати кількість обраних страв та додаткові коментарі до них.

Клас «Зал» містить інформацію про доступні зали ресторану, тобто короткий опис та місткість. Одне замовлення пов'язане з одним залом, але один зал може використовуватися у багатьох замовленнях. Таким чином, побудована UML-діаграма класів дозволяє чітко описати структуру системи та взаємозв'язки між її елементами, що спрощує реалізацію програмного забезпечення.

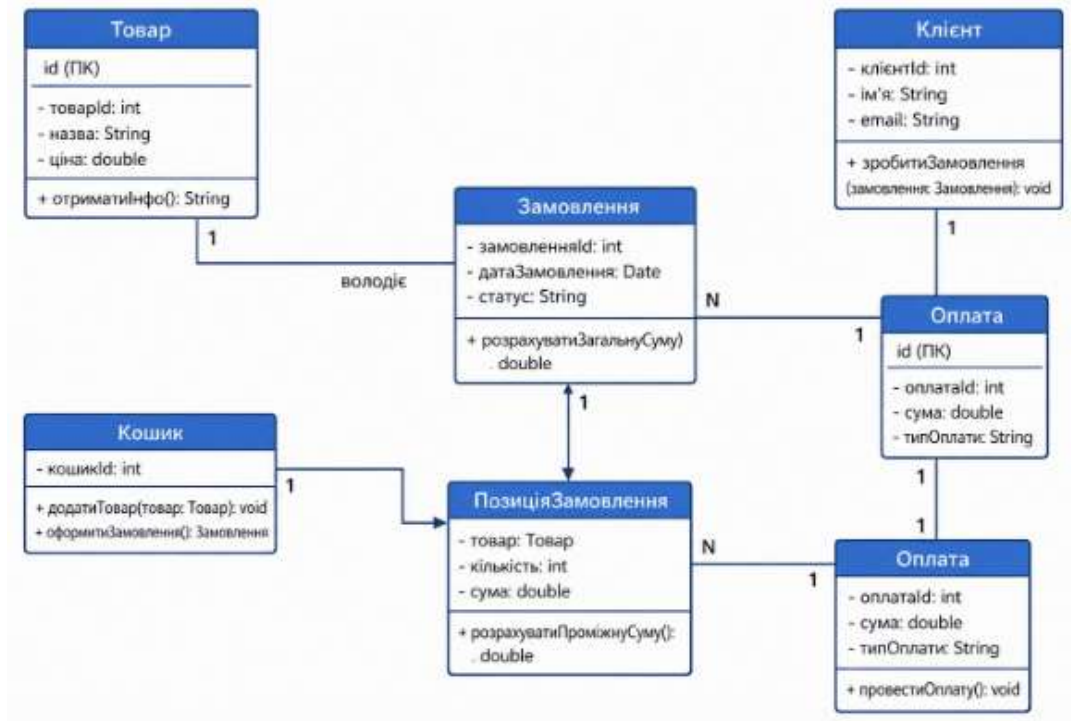


Рисунок 3.1 – UML-діаграма класів

Можна використовувати діаграму станів для візуалізації життєвого циклу взаємодії користувача з інтерфейсом. Вона показує переходи між різними кроками оформлення заходу – від початкового екрана вітання до фінального підтвердження замовлення, це дозволяє уникнути логічних помилок при розробці навігації системи (рисунок 3.2).

Ця діаграма дозволяє наочно показати логіку роботи інтерфейсу та всі можливі алгоритми взаємодії користувача з системою.

Початковим станом є введення даних користувача, де клієнт заповнює основну інформацію про захід. Після цього виконується перевірка правильності введених даних. У випадку виявлення помилок система повертає користувача до редагування даних.

Наступним етапом є вибір страв, де користувач формує своє меню. Після завершення вибору веб-орієнтована інформаційна система переходить до стану формування замовлення, де обчислюється загальна вартість.

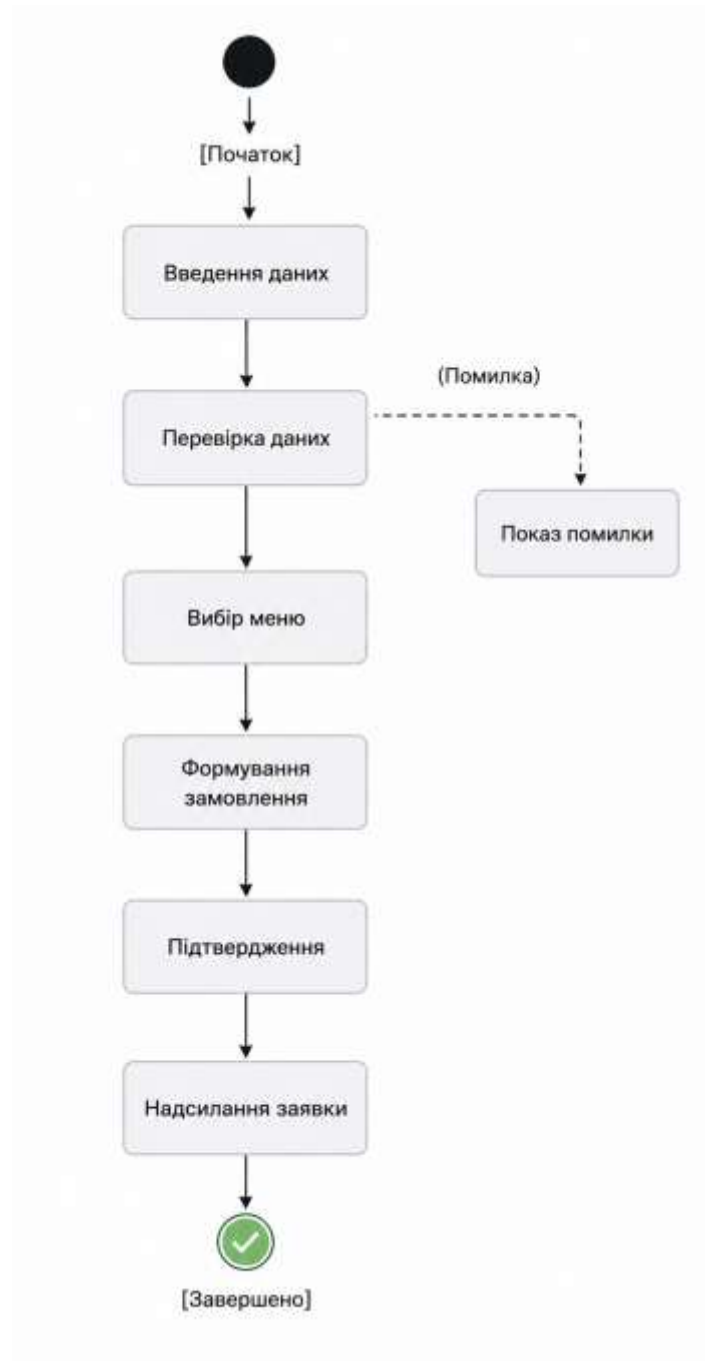


Рисунок 3.2 – UML-діаграма станів

Далі користувач переходить до підтвердження замовлення та його надсилання. Завершальним станом є успішна відправка заявки, після чого система може повернутися до початкового стану.

Для реалізації інформаційної системи було обрано сучасні веб-технології, які забезпечують зручність розробки, швидкість роботи та хорошу масштабованість.

Клієнтська частина реалізована з використанням бібліотеки React [18]. Це дозволило створити інтерактивний інтерфейс, який швидко реагує на дії користувача без перезавантаження сторінки. Використання хуків дозволило розділити логіку обробки даних та їх візуальне відображення. Це робить систему легшим у підтримці та дозволяє швидко масштабувати функціонал – наприклад, додати нові замовлення.

Для стилізації інтерфейсу використано CSS та утилітарні класи, що дозволило швидко створити сучасний дизайн [19].

Обмін даними з сервером реалізовано за допомогою прикладного програмного інтерфейсу та технології fetch, що забезпечує передачу даних у форматі JSON. Це дозволяє легко інтегрувати систему з серверною частиною.

Серверна частина відповідає за прийом, обробку та збереження даних замовлення. Такий підхід дозволяє розділити логіку інтерфейсу та обробки даних, що є хорошою практикою сучасної розробки.

У ході виконання роботи було використано сучасні та поширені технології веб-розробки, які активно застосовуються у реальних програмних проєктах, що підтверджує доцільність їх використання для реалізації даної системи.

Таким чином, створений набір інструментів повністю автоматизує шлях клієнта від першого початкового етапу до моменту реєстрації замовлення в системі ресторану.

Основна бізнес-логіка системи реалізована у вигляді функціональних компонентів та допоміжних функцій.

Фрагмент програмного коду серверної частини інформаційної системи наведений у додатку А.

Для збереження даних, введених користувачем, використовується об'єкт стану formData, який містить основні параметри замовлення: ім'я клієнта, номер телефону, контактні дані, дату та час проведення події, кількість гостей, тип заходу, обраний зал, а також список замовлених страв.

Наприклад, структура даних користувача реалізована наступним чином (рисунок 3.3).

```
const App = () => {
  const [step, setStep] = useState(0);
  const [formData, setFormData] = useState({
    name: "",
    phone: "+380",
    email: "",
    date: "",
    time: "",
    guests: "",
    eventType: "",
    hall: "",
    dishes: [],
  });
};
```

Рисунок 3.3 – Модель збереження вхідних даних користувача

Така структура дозволяє правильно зберігати всі дані та передавати їх між компонентами системи.

Для реалізації вибору страв було створено окремий компонент DishSelection, який відповідає за відображення списку страв, зміну їх кількості та додавання коментаря. Кількість страв змінюється за допомогою функції updateQuantity, яка обробляє як кнопки “+” та “-”, так і ручне введення значення (рисунок 3.4).

```
const updateQuantity = (dish, value, isAbsolute = false) => {
  const updated = [...formData.dishes];
  const index = updated.findIndex((d) => d.name === dish.name);
  let newQty;

  if (isAbsolute) {
    newQty = Math.max(0, value);
  } else if (index !== -1) {
    newQty = Math.max(0, updated[index].quantity + value);
  } else if (value > 0) {
    newQty = 1;
  } else {
    return;
  }

  if (newQty > 0) {
    if (index !== -1) {
      updated[index].quantity = newQty;
    } else {
      updated.push({ ...dish, quantity: newQty, comment: "" });
    }
  } else {
    if (index !== -1) updated.splice(index, 1);
  }

  setFormData({ ...formData, dishes: updated });
};
```

Рисунок 3.4 – Програмний код логіки зміни кількості страв у замовленні

Ця функція забезпечує коректну обробку всіх можливих ситуацій: додавання страви, зміна кількості або її видалення. Для додавання коментаря до страви була реалізована окрема функція: (рисунок 3.5).

```
const updateDishComment = (dish, newComment) => {
  setFormData(prev => ({
    ...prev,
    dishes: prev.dishes.map(d =>
      d.name === dish.name ? { ...d, comment: newComment } : d
    )
  }));
  setFormData({ ...formData, dishes: updated });
};
```

Рисунок 3.5 – Програмний код функції обробки текстових коментарів до страв

Це дозволяє користувачу залишати додаткові побажання до кожної страви.

Після завершення формування замовлення користувач переходить до сторінки підсумку, де відображається вся введена інформація та обрані страви. Загальна вартість замовлення розраховується автоматично на основі кількості та ціни кожної страви.

Серверна частина реалізована з використанням Node.js. Для передачі даних використовується HTTP-запит методом POST [20]. Відправка даних із клієнтської частини реалізована наступним чином (рисунок 3.6).

```
const sendFormData = async () => {
  try {
    const dataToSend = {
      ...formData,
      contact: formData.email
    };
    const response = await fetch("http://localhost:5000/submit", {
      method: "POST",
      headers: { "Content-Type": "application/json" },
      body: JSON.stringify(dataToSend),
    });
  }
};
```

Рисунок 3.6 – Програмний код клієнтського запиту для передачі замовлення на сервер

Таким чином усі дані про замовлення передаються на сервер для подальшої обробки або збереження.

Структурно система реалізована за принципом розподілу відповідальності: інтерфейс користувача відповідає за введення та відображення даних, тоді як серверна частина – за їх обробку. Такий підхід дозволяє зробити веб-орієнтовану інформаційну систему більш гнучкішим та зручнішим для подальшого розширення.

### 3.2 Інтерфейс користувача системи

Користувацький інтерфейс розробленої веб-орієнтованої інформаційної системи є важливою складовою, оскільки саме через нього здійснюється взаємодія користувача з інтерфейсом. Основною метою при розробці інтерфейсу було створення простого, зрозумілого та зручного середовища для оформлення замовлення. Фрагмент програмної реалізації користувацького інтерфейсу системи наведено в додатку Б.

Інтерфейс реалізовано у вигляді веб-орієнтованої інформаційної системи, що працює у браузері. Він побудований таким чином, щоб користувач крок за кроком проходить процес створення замовлення: від введення основних даних до вибору меню та підтвердження заявки.

При проєктуванні інтерфейсу було враховано основні принципи зручності використання. Зокрема, інтерфейс не перевантажений зайвими елементами, всі дії є логічно послідовними, а користувач завжди розуміє, що потрібно зробити на наступному етапі

Головний екран системи, який представлений на рисунку 3.7, містить привітання та перехід до оформлення замовлення.

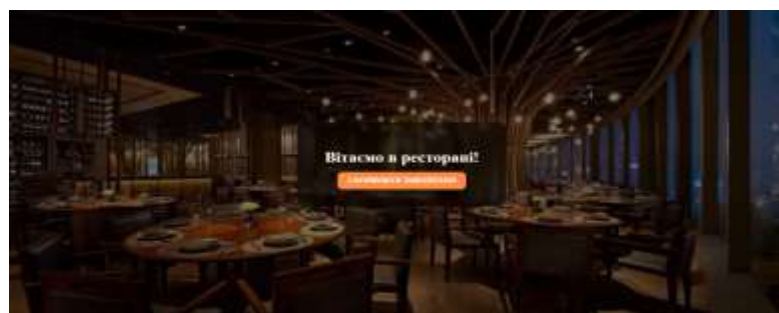


Рисунок 3.7 – Головний екран веб-орієнтованої системи

На наступному етапі користувач вводить основну інформацію про подію: ім'я, номер телефону, контактні дані, дату, час, кількість гостей, тип бенкету та обирає зал (рисунок 3.8).

The image shows a web application interface for event registration. It features a semi-transparent white form overlaying a background image of a restaurant interior with round tables and chairs. The form contains several input fields for user information, a date and time picker, a dropdown menu for event type, and a dropdown menu for room selection. A 'Зареєструватися' (Register) button is at the bottom right of the form.

Рисунок 3.8 – Форма введення даних користувача

Для забезпечення коректності даних реалізовано перевірку заповнення полів. Якщо користувач не заповнив обов'язкові поля, веб-орієнтована інформаційна система виводить відповідне повідомлення (рисунок 3.9).

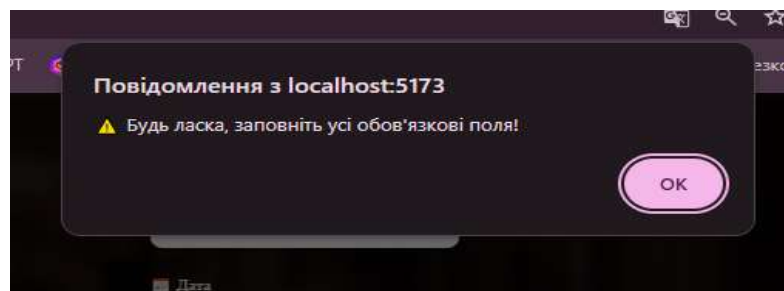


Рисунок 3.9 – Виведення повідомлення про заповнення усіх полів

Також реалізовано автоматичну рекомендацію залу залежно від кількості гостей та типу події, що спрощує процес вибору (рисунок 3.10).

The image shows a close-up of the bottom part of the registration form. It includes a text input field for 'Кількість гостей (до 80 осіб)' with the value '20'. Below it is a dropdown menu for 'Тип бенкету' with 'Сімейне посидіння' selected. There is a green checkmark next to the text 'Рекомендований зал: 3 зал'. Below that is another dropdown menu for 'Зал' with '3 зал (до 20 осіб)' selected. At the very bottom is a button labeled 'Далі до вибору меню'.

Рисунок 3.10 – Рекомендація залу

На наступному етапі користувач переходить на іншу сторінку, щоб скласти меню і обирає страви з меню (рисунок 3.11).



Рисунок 3.11 – Вибір страв для заходу

Для кожної страви доступна можливість:

- змінювати кількість;
- додавати коментар (наприклад, побажання щодо приготування) (рисунок 3.12).

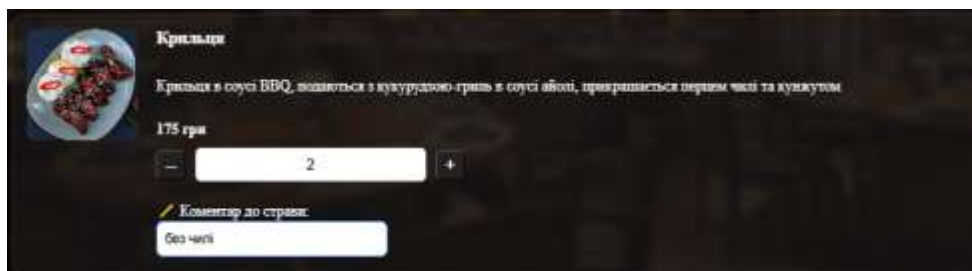


Рисунок 3.12 – Додавання коментаря до страви

Інтерфейс побудований у вигляді карток, що містять зображення, назву, опис та ціну страви (рисунок 3.13).

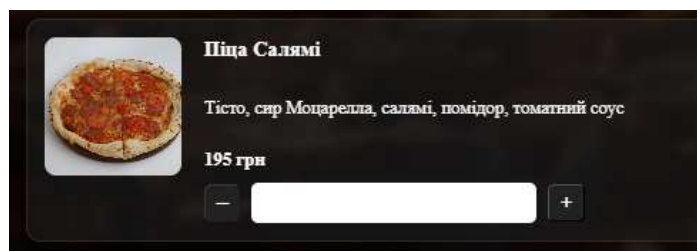


Рисунок 3.13 – Візуальне представлення страви

Це робить процес вибору більш наочним та зручним. На фінальному етапі користувач бачить повну інформацію про своє замовлення:

- введені дані;
- список страв;
- коментарі до страв;
- загальну вартість (рисунки 3.14 - 3.15).



Рисунок 3.14 – Підсумок замовлення



Рисунок 3.15 – Підсумок замовлення

Також передбачена можливість редагування замовлення або повернення до попередніх етапів (рисунк 3.16).



Рисунок 3.16 – Візуальне представлення інтерфейсу для редагування замовлення

Після відправки заявки система повідомляє користувача про успішне оформлення замовлення. У випадку помилки також виводиться відповідне повідомлення (рисунок 3.17).

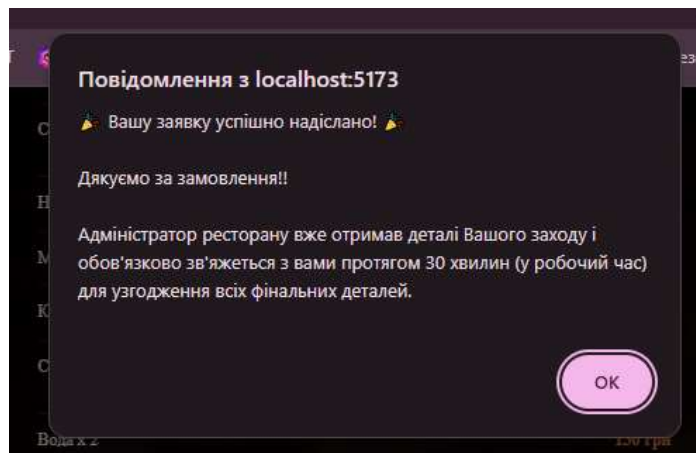


Рисунок 3.17 – Повідомлення про успішне відправлення

У результаті розробки інтерфейсу користувача було створено зручний та зрозумілий інструмент для оформлення заходу. Інтерфейс відповідає основним принципам, забезпечує простоту використання та не створює труднощів для користувача.

Завдяки покроковій структурі користувач легко орієнтується у процесі оформлення замовлення, а наявність підказок і перевірок мінімізує ймовірність помилок.

### 3.3 Тестування розробленої системи

Після завершення розробки було проведено тестування, щоб переконатися у правильній роботі всіх функцій веб-орієнтованої інформаційної системи [21] та виключити виникнення серйозних помилок. Основна увага приділялася перевірці взаємодії користувача з інтерфейсом та стабільності передачі даних на серверну частину через систему. Тестування проходило за двома ключовими напрямками: перевірка клієнтської логіки (frontend) та контроль обміну даними з базою (backend). У межах клієнтської частини тестувалося, як система реагує на введення даних, чи коректно перемикаються кроки оформлення банкету та чи зберігається

інформація при переході між екранами. Особлива увага була приділена «захисту від помилок»: система була протестована на здатність блокувати некоректні запити, наприклад, спроби замовити банкет без вказання дати або з нульовою кількістю гостей.

Другий етап тестування підтвердив, що сформований об'єкт замовлення у форматі JSON без втрат передається на сервер і правильно фіксується в системі закладу. Результати проведеної перевірки виявили, що система працює стабільно, забезпечує високу швидкість інтерфейсу та є повністю готова до інтеграції у цифровий сервіс ресторану.

Функціональне тестування проводилось шляхом перевірки роботи основних можливостей системи.

Було перевірено:

- введення даних користувача (ім'я, телефон, контакт);
- вибір дати та часу;
- введення кількості гостей;
- вибір типу події та залу;
- додавання та видалення страв;
- зміну кількості страв;
- додавання коментарів до страв;
- формування підсумку замовлення;
- відправку даних на сервер.

Результати тестування показали, що всі функції працюють коректно, а дані обробляються без помилок.

Окрему увагу було приділено перевірці правильності введених користувачем даних.

Було протестовано наступні ситуації:

- залишення обов'язкових полів порожніми;
- введення некоректного номера телефону;
- введення нульової або від'ємної кількості гостей;
- спроба перейти до наступного етапу без заповнення форми.

У таких випадках система коректно виводить повідомлення про помилку та не дозволяє продовжити роботу, що запобігає введенню некоректних даних.

Було перевірено зручність використання інтерфейсу та правильність відображення елементів. Зокрема:

- коректне відображення всіх сторінок;
- адаптивність інтерфейсу;
- правильна робота кнопок;
- відображення повідомлень користувачу;
- плавний перехід між етапами.

Результати тестування показали, що інтерфейс є зрозумілим і не викликає труднощів у користувача.

Було перевірено правильність розрахунку загальної вартості замовлення.

Під час тестування також було встановлено, що інформаційна система правильно обчислює вартість при зміні кількості страв, їх додаванні або видаленні.

Було перевірено роботу функції надсилання даних на сервер.

У результаті тестування встановлено, що:

- дані передаються у форматі JSON;
- сервер коректно приймає запит;
- у разі успішної відправки відображається повідомлення;
- у разі помилки виводиться відповідне повідомлення.

Результати тестування представлені в таблиці 3.1.

Таблиця 3.1 – Результати тестування

| Тестовий випадок | Очікуваний результат | Фактичний результат    |
|------------------|----------------------|------------------------|
| Порожні поля     | Помилка              | Помилка відображається |
| Вибір страв      | Додається            | Працює                 |
| Зміна кількості  | Перерахунок          | Працює                 |
| Коментар         | Зберігається         | Працює                 |
| Відправка        | Успіх                | Працює                 |

У результаті проведеного тестування встановлено, що розроблений програмний продукт працює коректно та відповідає поставленим вимогам. Усі основні функції реалізовані та протестовані, помилок не виявлено.

Веб-орієнтована інформаційна система організації заходів у ресторанному бізнесі забезпечує правильну обробку даних, зручну взаємодію з користувачем та стабільну роботу під час виконання основних сценаріїв використання.

## ВИСНОВКИ

У процесі виконання кваліфікованої роботи було:

1. Розглянуто особливості організації заходів, обслуговування та проаналізовано сучасні підходи до автоматизації таких процесів.

2. Досліджено предметну область, визначено основні бізнес-процеси, а також виділено їх транзакційну та аналітичну складові. Це дозволило зрозуміти, які саме функції повинна виконувати інформаційна система, щоб було корисно як для клієнта, так і для ресторану.

3. Сформульовано вимоги до системи та побудовано відповідні моделі, зокрема діаграми, що відображають структуру, логіку роботи та інформаційні зв'язки між елементами системи. Також було розроблено алгоритмічне забезпечення, яке описує процес оформлення замовлення та взаємодію користувача із системою.

4. Спроектовано базу даних, представлено концептуальну, логічну та фізичну моделі даних, які забезпечують збереження, обробку та передачу інформації про замовлення, користувачів, страви та зали. Реалізація бази даних дозволяє стабільно працювати з даними та забезпечує коректну роботу системи.

5. Реалізовано веб-орієнтовану інформаційну систему для організації заходів у ресторанному бізнесі, яка дозволяє користувачу самостійно сформувати банкетне меню, вибрати параметри заходу та надіслати заявку до ресторану. Реалізація виконана з використанням сучасних веб-технологій, що забезпечують зручний інтерфейс та швидку взаємодію з користувачем.

7. У процесі тестування перевірено роботу всіх основних функцій. Результати показали, що все працює стабільно, правильно обробляє введені дані та коректно формує результати.

8. Розроблена інформаційна системи має практичну цінність і може бути впроваджена у реальних умовах роботи ресторану, а також доопрацьована у майбутньому шляхом додавання нових функцій, таких як онлайн-оплата, інтеграція з CRM-системами або розширена аналітика замовлень.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Tonkykh O., Sushchenko O., Lychko S. Features of Managing the Event-Management System in the Hotel and Restaurant Business. *Management*. 2025. Vol. 42, №. 2. URL: <https://jrnل.knutd.edu.ua/index.php/mng/article/view/1922>
2. Tripleseat: Event Management Software for Restaurants, Hotels, & Venues. URL: <https://tripleseat.com/>
3. Roy D., Spiliotopoulou E., de Vries J. Restaurant analytics: emerging practice and research opportunities. *Production and Operations Management*. 2022. Vol. 31, № 10. P. 3687–3709. URL: <https://doi.org/10.1111/poms.13809>
4. Dio K., Puente D., Estrada J. Hotel and restaurant management system core transaction (order management with POS, kitchen and bar module, food and beverage costing, inventory management, resto analytics) *Procedia Computer Science*. 2023. Vol. 3. URL: <https://ojs.aaresearchindex.com/index.php/aasgbcjpmra/issue/view/87>
5. Tkachenko V., Lukianiuk S. Analysis of the use of the Redis in the distributed order processing system in the restaurant network. *Technology audit and production reserves*. 2021 URL: <https://ojs.aaresearchindex.com/index.php/aasgbcjpmra/issue/view/87>
6. Kansakar P., Munir A., Shabani N. Technology in Hospitality Industry: Prospects and Challenges. 2017. URL: Режим доступу: <https://arxiv.org/abs/1709.00105>
7. Ресторан Namangal. URL: <https://namangalsteakhouse.ua/>
8. Ресторан Samogon. URL: <https://samogon.org/>
9. Що таке REST API і як він працює : стаття Dan.it Education. URL: <https://dan-it.com.ua/uk/blog/shho-take-rest-api-i-yak-vin-praczyuye/>
10. Ушенко Ю.О., Ковальчук М.Л., Гавриляк М.С., Негрич А.Л. Методологія інформаційних систем та баз даних: теоретичний і практичний підходи: навч. посібник. Чернівці: ЧНУ ім. Ю. Федьковича, 2021. 240 с.
11. IBM. Client-server architecture. Cloud Learn Hub. URL: <https://www.ibm.com/think/topics>

12. Берко А.Ю., Верес О.М., Пасічник В.В. Системи баз даних та знань. Книга 1. Організація баз даних та знань: підручник. Львів: «Магнолія 2006», 2024. 675 с.
13. Pressman R. S., Maxim B. R. Software Engineering: A Practitioner's Approach. 9th ed. New York : McGraw-Hill Education, 2019. URL: <https://www.mheducation.com/highered/product/software-engineering-practitioner-s-approach-pressman-maxim/M9781259872976.html>
14. Берко А.Ю., Верес О.М., Пасічник В.В. Системи баз даних та знань. Книга 2. Системи управління базами даних та знань: навч. посібник. Львів: «Магнолія-2006», 2024. 584 с.
15. Документація бібліотеки React.js для розробки користувацьких інтерфейсів. URL: <https://react.dev/>
16. Технічна документація середовища виконання Node.js. URL: <https://nodejs.org/>
17. Для чого потрібні UML діаграми? URL: <https://foxminded.ua/uml-diagramy/>
18. Панченко О. В. Сучасні фреймворки JavaScript: React та Vue.js у розробці високонавантажених систем : навч. посіб. Київ : Політехніка, 2023. 180 с.
19. Методичні вказівки до лабораторних робіт з дисципліни «Веб-технології та веб-дизайн», уклад. О. В. Пшенична. Запоріжжя : ЗНУ, 2022. 45 с.
20. Саммервілл І. Інженерія програмного забезпечення. 10-те вид. Київ : Вільямс, 2020. 712 с.
21. Концограда С.М., Турченко І.В. Веб-орієнтована інформаційна система організації заходів у ресторанному бізнесі. ICSPM 2026: Інтелектуальні обчислювальні системи та управління проєктами: збірник тез доповідей міжнародної науково-практичної конференції студентів і молодих вчених (Тернопіль, 21-22 травня 2026 р). Тернопіль: ЗУНУ, 2026. С. 142-145.
22. UML Tutorial. Tutorialspoint. URL: <https://www.tutorialspoint.com/uml/index.htm>
23. Rocha R. Serverless Web Applications with React and Firebase. Birmingham: Packt Publishing, 2024. 310 p.

24. Діаграма послідовності (sequence diagrams). URL: <https://www.maxzosim.com/sequence-diagrams/>.
25. Документація інтегрованого середовища розробки Visual Studio Code. URL: <https://code.visualstudio.com/>
26. Пасічний В. UX/UI дизайн: навч. посіб. для здобувачів ступеня вищої освіти бакалавра. Київ : ResearchGate, 2021. 154 с.
27. Комар М.П., Саченко А.О., Васильків Н.М., Гладій Г.М., Коваль В.С., Ліп'яніна-Гончаренко Х.В. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за першим (бакалаврським) рівнем вищої освіти. Тернопіль: ЗУНУ, 2024. 52 с.

## Додаток А

### Фрагмент програмного коду серверної частини інформаційної системи

```
const HALLS_DATA = [
  { name: "1 зал", capacity: 50, events: ["День народження", "Хрестини",
"Весілля", "Сімейні посиденьки", "Корпоратив"] },
  { name: "2 зал", capacity: 80, events: ["День народження", "Хрестини",
"Весілля", "Сімейні посиденьки", "Корпоратив"] },
  { name: "3 зал", capacity: 20, events: ["День народження", "Поминки",
"Сімейні посиденьки", "Корпоратив"] },
  { name: "VIP зал", capacity: 12, events: ["День народження", "Поминки",
"Сімейні посиденьки", "Корпоратив"] },
];
const FIRST_DISHES = [
  { name: "Борщ український", price: 185, image: "https://cdn-
media.choiceqr.com/prod-eat-arcobaleno/menu/bUTBGcy-ZUGVJyL-gCICJWm.webp",
description: "Традиційний український борщ з сметаною, розбіфом, грінками та
салом" },
  { name: "Бульйон з куркою", price: 98, image: "https://cdn-
media.choiceqr.com/prod-eat-arcobaleno/menu/tgHsIHo-kkQJwCD-
kgLmbyS_original.jpeg.webp", description: "Суп з курячим філе, морквою та
фетучіні" },
  { name: "Грибна юшка", price: 120, image: "https://cdn-
media.choiceqr.com/prod-eat-arcobaleno/menu/JmXCTTA-GDGuUUh-
FDoXCrk_original.jpeg.webp", description: "Ароматний суп з грибами та зеленню"
},
  { name: "Млинці з жульеном", price: 216, image: "https://cdn-
media.choiceqr.com/prod-eat-arcobaleno/menu/GbtIYKW-sktDrDS-
hXRObFE_original.jpeg.webp", description: "Нижні млинці з вершковим жульеном з
грибів та курки" },
];
const DishSelection = ({ title, dishes, formData, updateQuantity,
updateComment, nextStep, setStep, currentStep }) => {
const total = formData.dishes.reduce((sum, d) => sum + d.price * d.quantity,
0);
return (
<div className="menu-container">
<h2 className="text-3xl font-bold mb-6 text-white">{title}</h2>
{dishes.map((dish) => {
const existing = formData.dishes.find((d) => d.name === dish.name);
const currentQuantity = existing ? existing.quantity : 0;
```

```

const currentComment = existing ? existing.comment : "";
return (
<div key={dish.name} className="dish-card">
<img src={dish.image} alt={dish.name} />
<div className="dish-info">
<h3>{dish.name}</h3>
{dish.description && <p className="text-gray-300 text-sm mb-2">{dish.description}</p>}
<strong className="text-xl block mb-2">{dish.price} ррн</strong>
<div className="quantity-controls mb-3">
<button
onClick={() => updateQuantity(dish, -1)}
disabled={currentQuantity === 0}
/>
<input
type="number"
min="0"
value={currentQuantity === 0 ? "" : currentQuantity}
onChange={(e) => {
const rawValue = e.target.value;
const newQty = rawValue === "" ? 0 : parseInt(rawValue) || 0;
updateQuantity(dish, newQty, true);
}}
className="quantity-input"
/>
<button onClick={() => updateQuantity(dish, 1)}></button>
</div>
{currentQuantity > 0 && (
<div className="comment-input-container">
<label className="text-sm font-semibold mb-1 block text-gray-300">
</label>
<input
type="text"
placeholder="Комментар:"
value={currentComment || ""}
onChange={(e) => updateComment(dish, e.target.value)}
className="w-full p-2 rounded bg-white text-black text-sm outline-none"
/>
</div>
)}
)

```

## Додаток Б

### Фрагмент програмної реалізації користувацького інтерфейсу системи

```
body {
  margin: 0;
  padding: 0;
  min-width: 320px;
  min-height: 100vh;
  background-image:
url("https://png.pngtree.com/thumb_back/fh260/background/20230721/pngtree-d-
rendering-of-luxurious-modern-bar-restaurant-with-chic-furnishings-and-
image_3720690.jpg");
  background-color: rgba(0, 0, 0, 0.6);
  background-blend-mode: darken;
  background-repeat: no-repeat;
  background-size: cover;
  background-position: center;
  background-attachment: fixed;
  color: white;
}
textarea {
  color: black;
  background: white;
}
.menu-container {
  width: 75%;
  margin: 40px auto;
  background: rgba(0, 0, 0, 0.45);
  padding: 30px;
  border-radius: 20px;
  backdrop-filter: blur(4px);
}
.dish-card {
  display: flex;
  gap: 20px;
  background: rgba(255, 255, 255, 0.1);
  padding: 20px;
  border-radius: 15px;
  margin-bottom: 20px;
  color: white;
}
```

```

.dish-card img {
  width: 150px;
  height: 150px;
  object-fit: cover;
  border-radius: 10px;
}

.dish-info {
  flex: 1;
  color: white;
}

.quantity-controls {
  display: flex;
  align-items: center;
  gap: 10px;
}

.quantity-controls button {
  width: 32px;
  height: 32px;
  border-radius: 6px;
  background: #222;
  color: white;
  font-size: 20px;
}

.welcome-container {
  min-height: 100vh;
  display: flex;
  justify-content: center;
  align-items: center;
  text-align: center;
}

.welcome-box {
  background: rgba(0, 0, 0, 0.55);
  padding: 40px 60px;
  border-radius: 20px;
  backdrop-filter: blur(4px);
  max-width: 600px;
}

```

Додаток В  
Копія опублікованих результатів

Міністерство освіти і науки України  
Західноукраїнський національний університет  
Факультет комп'ютерних інформаційних технологій  
Кафедра інформаційно-обчислювальних систем і управління



**Матеріали**

I Міжнародної науково-практичної конференції студентів і молодих вчених  
«ICSPM 2026: Intelligent Computing Systems and Project Management /  
Інтелектуальні обчислювальні системи та управління проєктами»  
21–22 травня 2026 р.



м. Тернопіль - 2026

УДК 004.8:005.8](063)

Присвячено 60-річчю Західноукраїнського національного університету

**ICSPM 2026: Інтелектуальні обчислювальні системи та управління проектами** : збірник тез доповідей I Міжнародної науково-практичної конференції студентів і молодих вчених (Тернопіль, 21–22 травня 2026 року). – Тернопіль : ЗУНУ, 2026. – 190 с.

До збірника увійшли тези доповідей учасників I Міжнародної науково-практичної конференції студентів і молодих вчених «ICSPM 2026: Intelligent Computing Systems and Project Management / Інтелектуальні обчислювальні системи та управління проектами», що відбулася на базі кафедри інформаційно-обчислювальних систем і управління факультету комп'ютерних інформаційних технологій Західноукраїнського національного університету.

Матеріали відображають результати досліджень за такими напрямками:

1. Інтелектуальні обчислення та програмні системи.
2. Проектно-орієнтоване управління та процеси прийняття рішень.
3. Штучний інтелект, аналітика даних і кібернетика.
4. Прикладні технології та інформаційно-комунікаційні системи.
5. Сучасні інформаційні технології в економіці, праві та освіті.

*Рекомендовано до друку на засіданні кафедри інформаційно-обчислювальних систем і управління Західноукраїнського національного університету (протокол №12 від 26.05.2026 р.).*

За зміст наукових праць, точність наведених цитувань, перекладу та достовірність фактологічних матеріалів відповідальність несуть автори публікацій. У збірнику збережено авторську стилістику, пунктуацію та орфографію.

© Кафедра інформаційно-обчислювальних систем і управління ЗУНУ, 2026  
© Західноукраїнський національний університет, 2026

|                                                                                                                                          |     |
|------------------------------------------------------------------------------------------------------------------------------------------|-----|
| Воїнський Ю.В., Коваль В.С.<br>ІНТЕЛЕКТУАЛЬНА СИСТЕМА КЕРУВАННЯ ОПОВІЩЕННЯМИ В<br>ОСВІТНЬОМУ ЗАКЛАДІ .....                               | 125 |
| Ворожбит Р.О., Турченко І.В.<br>КРОСПЛАТФОРМЕННИЙ МОБІЛЬНИЙ ЗАСТОСУНОК ДЛЯ ВИКОНАННЯ<br>СПОРТИВНИХ ВПРАВ ТА САМОКОНТРОЛЮ .....           | 129 |
| Гнатів С.В., Васильків Н.М.<br>АЛГОРИТМИ ФУНКЦІОНУВАННЯ ІоТ-СИСТЕМИ ОБЛІКУ РОБОЧОГО<br>ЧАСУ З БАГАТОФАКТОРНОЮ АВТЕНТИФІКАЦІЄЮ .....      | 132 |
| Заблоцький М.М., Хміль В.А., Васильків Н.М.<br>ВПЛИВ ОПЕРАТИВНОГО ПЕРСОНАЛУ НА ФУНКЦІОНУВАННЯ<br>ІНФОРМАЦІЙНОЇ СИСТЕМИ .....             | 136 |
| Клюд Д.В., Васильків Н.М.<br>ІНТЕРАКТИВНИЙ UI/UX МОДУЛЬ МОБІЛЬНОГО СЕРВІСУ ОНЛАЙН-<br>ЗАМОВЛЕНЬ ЛІКАРСЬКИХ ЗАСОБІВ.....                  | 139 |
| Концограда С.М., Турченко І.В.<br>ВЕБ-ОРІЄНТОВАНА ІНФОРМАЦІЙНА СИСТЕМА ОРГАНІЗАЦІЇ ЗАХОДІВ<br>У РЕСТОРАННОМУ БІЗНЕСІ .....               | 142 |
| Олізар М., Морохович В.<br>РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОЦІНЮВАННЯ<br>ЕФЕКТИВНОСТІ ГІБРИДНИХ ДЖЕРЕЛ ВІДНОВЛЮВАНОЇ ЕНЕРГІЇ ..... | 146 |
| Путькало М.П., Домбровський М.З.<br>ОЦІНЮВАННЯ ПРОДУКТИВНОСТІ GPU У ЗАДАЧАХ ВІЗУАЛІЗАЦІЇ ПРИ<br>AI- ТА EDGE-НАВАНТАЖЕННЯХ .....          | 150 |
| Старостюк В.С., Турченко І.В.<br>ВЕБ-ОРІЄНТОВАНА ІНФОРМАЦІЙНА СИСТЕМА ВЕТЕРИНАРНОЇ<br>КЛІНІКИ.....                                       | 155 |
| Телька М.І., Мимоход О.Б., Федорук М.Ю.<br>ПРОЄКТУВАННЯ DASHBOARD-СИСТЕМИ ДЛЯ МОНІТОРИНГУ<br>МЕДИЧНИХ ПОКАЗНИКІВ .....                   | 159 |
| Федіков В.О., Турченко І.В.<br>ВЕБ-ОРІЄНТОВАНА ІНФОРМАЦІЙНА СИСТЕМА ЗАМОВЛЕНЬ<br>КОНДИТЕРСЬКИХ ВИРОБІВ .....                             | 162 |

УДК 004.77:004.42

## **ВЕБ-ОРІЄНТОВАНА ІНФОРМАЦІЙНА СИСТЕМА ОРГАНІЗАЦІЇ ЗАХОДІВ У РЕСТОРАННОМУ БІЗНЕСІ**

Концограда Світлана Михайлівна,  
здобувач першого (бакалаврського) рівня вищої освіти,  
kontsohradas@gmail.com

Турченко Ірина Василівна,  
к.т.н., доцент, доцент кафедри  
інформаційно-обчислювальних систем  
і управління,  
itu@wunu.edu.ua  
ORCID: 0000-0002-9441-6669  
Західноукраїнський національний університет

У сучасних умовах цифрової трансформації ресторанного бізнесу набуває введення веб-орієнтованих інформаційних систем, які забезпечують оптимізацію процесів бронювання столиків, організації заходів, управління замовленнями та взаємодії з клієнтами. У сучасних дослідженнях розглядають питання автоматизації ресторанних бізнесів, використання інформаційних технологій та цифрових систем управління.

У праці [1] автори розглядають важливість використання інформаційних систем у діяльності сучасних ресторанів. Дослідники вказують, що автоматизація дозволяє оптимізувати управлінські процеси, удосконалити облік ресурсів та підвищити швидкість прийняття управлінських рішень. Особлива увага приділена на те, що такі інформаційні системи є необхідною складовою конкурентоспроможного ресторанного бізнесу.

У статті [2] досліджуються теоретичні аспекти інформаційних процесів управління ресторанными бізнесами. Автори аналізують структуру інформаційних систем, їх програмні та організаційні компоненти, а також визначають роль цифрових технологій у забезпеченні якісного та швидкого обслуговування клієнтів. У роботі наголошується на те, що сучасні інформаційні системи сприяють автоматизації бізнес-процесів, підвищують швидкість обслуговування та покращують комунікації між персоналом і клієнтами.

Також важливий внесок у дослідження цифрової трансформації ресторанної сфери зроблено у [3]. У роботі розглядаються сучасні цифрові інструменти та ІТ-рішення, які використовуються у ресторанному бізнесі, наприклад, CRM-системи, аналітичні платформи та веб-орієнтовані сервіси. Автори звертають увагу на те, що використання цифрових систем сприяє вдосконаленню процесів організації заходів, бронювання столиків, залів та взаємодії з клієнтами, що є важливим елементом для розробки веб-орієнтованих інформаційних систем.

На основі проведеного аналізу джерел можна зробити висновок, що сучасні веб-орієнтовані інформаційні системи повинні забезпечувати автоматизацію основних бізнес-процесів ресторанної діяльності, наприклад організацію заходів, бронювання залів, самостійне формування меню, управління замовленнями та взаємодію з користувачами. Тому було сформовано загальну структуру веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі, яка представлена на рисунку 1.



Рисунок 1 – Структура веб-орієнтованої інформаційної системи

Визначено основні елементи системи: користувачів, замовлення та зали, а також встановлено зв'язки між ними. Тобто, один користувач може створювати кілька замовлень, а кожен зал може бути пов'язаний із різними замовленнями. Завдяки такому способу, інформацію про замовлення можна впорядкувати і уникнути дублювання даних та забезпечити управління інформаційними потоками.

Під час розробки системи основна увага приділялася швидкості роботи, зручності використання та можливості подальшого розширення функціоналу. Важливим етапом стало створення простого та зрозумілого інтерфейсу, який дозволяє користувачам без технічних знань легко оформлювати замовлення та взаємодіяти із системою. Складний або перевантажений інтерфейс може ускладнювати роботу користувача та стати причиною відмови від оформлення заявки ще на початковому етапі.

Окремий функціонал управління даними системи є для адміністратора. Він має доступ до інформації про всі замовлення, може перевіряти, переглядати замовлення та видаляти в разі потреби. Важливою складовою системи є забезпечення захисту та надійності даних. У системі реалізовано різні права

доступу відповідно до ролей користувачів, завдяки чому звичайні користувачі та адміністратор мають різні можливості й можуть виконувати лише передбачені для них функції. Тобто клієнти мають можливість лише оформляти свої замовлення, а адміністратор може переглядати дані сформовані клієнтами.

Таким чином, запропонована інформаційна система є сучасним засобом оптимізації процесу організації заходів, що відповідає сучасним вимогам до цифрових рішень у сфері управління даними та сервісами. Веб-орієнтована інформаційна система для організації заходів у ресторанному бізнесі (головна сторінка представлена на рисунку 2) дозволяє оптимізувати процес оформлення замовлень, спростити взаємодію між клієнтом і закладом та підвищити управління ресторанными послугами.

**Інформація про захід:**

1. **Назва заходу**

2. **Номер телефону**

Потрібно ввести цифру: 9

3. **Електронна пошта**

4. **Дата**

5. **Час**

6. **Кількість гостей (до 10 осіб)**

7. **Тип заходу**

8. **Тип**

[Дати до вибору меню](#)

Рисунок 2 - Головна сторінка веб-орієнтованої інформаційної системи організації заходів у ресторанному бізнесі

Розроблена інформаційна система не лише забезпечує зручну роботу з даними та сервісами, а й закладає основу для подальшого розвитку проєкту. Завдяки модульній структурі, систему можна легко розширювати у майбутньому, додавати нові функції відповідно до потреб бізнесу і впроваджувати у реальних умовах роботи ресторану, а також доопрацьовувати у майбутньому шляхом додавання нових функцій, таких як онлайн-оплата, інтеграція з CRM-системами або розширена аналітика замовлень. Це дозволить

ресторану швидко адаптовуватися до змін на ринку, покращувати якість обслуговування клієнтів та швидко масштабувати цифрові послуги.

#### Список використаних джерел

1. Golovko G., Borozdin M., Tokar Y. The need to implement automation and information system in the management of the restaurant business. *Control, navigation and communication systems*. 2021. № 2(64). P. 67-69. URL: <https://journals.nupp.edu.ua/sunz/article/view/2309>
2. Небаба Н.О., Кучер М. М., Язіна В. А., Сайгак Є. Л., Головко Н. О. Теоретичні аспекти інформаційного забезпечення процесів управління підприємствами ресторанного господарства. *Системи та технології*. 2023. № 2 (66). С. 138-147.
3. Шулежко О., Гончар Л. Формування системи цифрового менеджменту в ресторанному бізнесі крізь призму методологічних підходів. *Економіка та суспільство*. 2024. № 69. URL: <https://economyandsociety.in.ua/index.php/journal/article/view/5163>