

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

**Допущено до захисту:**  
Завідувач кафедри  
комп’ютерних  
технологій та  
економічної  
кібернетики  
д. е. н., проф. П. М. Грицюк

«\_\_\_\_\_» \_\_\_\_\_ 2026 р.

**КВАЛІФІКАЦІЙНА РОБОТА**  
**НА ЗДОБУТТЯ ОСВІТНЬО–КВАЛІФІКАЦІЙНОГО РІВНЯ**  
**«БАКАЛАВР»**

**«Комплексна інформаційна система для підвищення ефективності  
роботи кав’ярні»**

**Виконала:**  
здобувачка вищої освіти за ОПІ  
«Інформаційні системи та  
технології» спеціальності 126  
«Інформаційні системи та  
технології», групи ІСТ–41  
**Демчук Марина Володимирівна**

**Керівник:**  
к.е.н., доцент Волошин В. С.  
**Рецензент:**  
к.п.н., доцент Парфенюк О. В.

Рівне – 2026

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний університет водного господарства  
та природокористування

Навчально-науковий інститут кібернетики,  
інформаційних технологій та інженерії

Кафедра комп'ютерних технологій та економічної кібернетики

Освітньо-кваліфікаційний рівень - бакалавр  
Освітньо-професійна програма «Інформаційні системи і технології»  
Спеціальність 126 «Інформаційні системи та технології»

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
комп'ютерних технологій та  
економічної кібернетики  
д.е.н., професор П.М. Грицюк

“ ” \_\_\_\_\_ 2026 року

**З А В Д А Н Н Я  
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

**Демчук Марина Володимирівна**

(прізвище, ім'я, по батькові)

1.Тема роботи: Автоматизація процесів коворкінгу з використанням  
інтерактивних інформаційних систем

керівник роботи: **Волошин Володимир Степанович, к.е.н., доцент**

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджена наказом по університету С № 476 від 25.04.2026.

2. Термін здачі студентом закінченої роботи 08.06.2025 року

3. Вихідні дані до роботи \_\_\_\_\_

посібники, наукові статті, інтернет- джерела за темою роботи.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) :

1. Провести аналіз особливостей діяльності кав'ярень в умовах сучасного ринку та воєнного стану. 2. Дослідити існуючі інформаційні системи автоматизації закладів громадського харчування та визначити їхні основні функціональні можливості. 3. Виконати порівняльний аналіз вебсайтів сучасних кав'ярень і визначити основні вимоги до структури та функціоналу вебресурсу. 4. Розробити логічну модель бази даних MySQL для забезпечення обліку діяльності кав'ярні. 5. Створити інформаційну систему обліку для автоматизації основних процесів роботи закладу. 6. Розробити вебсайт кав'ярні «CoffeeShop» на основі WordPress та реалізувати його основні функціональні можливості. 7. Оцінити ефективність використання розробленої інформаційної системи та визначити переваги її впровадження для діяльності

кав'ярні.

## 5. Перелік графічного матеріалу

*Презентація за матеріалами бакалаврської роботи.*

## 6. Консультанти розділів роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата   |                  |
|--------|-------------------------------------------|----------------|------------------|
|        |                                           | Завдання видав | Завдання прийняв |
| 1      | Волошин В.С., доцент кафедри КТЕК         |                |                  |
| 2      | Волошин В.С., доцент кафедри КТЕК         |                |                  |
|        |                                           |                |                  |

Дата видачі завдання 19 січня 2026 р.

## КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів магістерської роботи                           | Термін виконання етапів (роботи) | Примітка |
|-------|-------------------------------------------------------------|----------------------------------|----------|
| 1     | Робота над Розділом 1                                       | 20.01.2026-01.03.2026            |          |
| 2     | Робота над Розділом 2                                       | 02.03.2026-25.04.2026            |          |
| 3     | Підготовка пояснювальної записки                            | 26.04.2026-25.05.2026            |          |
| 4     | Написання вступу, висновків, реферату                       | 26.05.2026-08.06.2026            |          |
| 5     | Попередній захист роботи                                    | 09.06.2026                       |          |
| 6     | Підготовка презентації роботи                               | 10.06.2026                       |          |
| 7     | Відгук керівника, рецензування роботи, перевірка на плагіат | 11.06.2026                       |          |
| 8     | Допуск до захисту                                           | 12.06.2026                       |          |

Студент(ка) \_\_\_\_\_ Демчук М.В.  
(підпис) (прізвище і ініціали)

Керівник кваліфікаційної роботи  
\_\_\_\_\_ Волошин В.С.  
(підпис) (прізвище і ініціали)

**ЗАЯВА**  
**щодо самостійності виконання випускної кваліфікаційної роботи**

Я, \_\_\_\_\_ (ПІБ),  
студент(ка) \_\_\_\_\_ курсу \_\_\_\_\_ групи \_\_\_\_\_ ННІ \_\_\_\_\_

**ЗАЯВЛЯЮ:**

моя випускна кваліфікаційна робота на тему « \_\_\_\_\_  
\_\_\_\_\_ »,   
яка надається в екзаменаційну комісію із захисту кваліфікаційних робіт зі спеціальності \_\_\_\_\_  
« \_\_\_\_\_ »  
для захисту, виконана самостійно і не містить плагіату.

Всі запозичення з друкованих та електронних джерел, у тому числі із захищених раніше випускових кваліфікаційних робіт мають відповідні посилання.

Я не використовував(ла) шахрайські методи маніпуляції з текстом (заміна букв, інтервали, мікропробіли, білі знаки, парафрази та ін.).

Інструменти штучного інтелекту використовував(ла) без порушення академічної доброчесності.

Я ознайомлений(а) з чинним Порядком перевірки навчальних, кваліфікаційних, навчально-методичних та наукових робіт на наявність ознак академічного плагіату в НУВГП та Положенням про академічну доброчесність в НУВГП, за яким виявлення плагіату є підставою для відмови в допуску моєї роботи до захисту та застосування відповідних санкцій (академічної відповідальності).

**АКТ**  
**перевірки випускної кваліфікаційної роботи**

Відповідно до даних системи StrikePlagiarism файл

містить: \_\_\_\_ % коефіцієнта подібності; \_\_\_\_ % коефіцієнта цитованості;  
\_\_\_\_ замінених букв; \_\_\_\_ інтервалів; \_\_\_\_ мікропробілів; \_\_\_\_ білих знаків;  
\_\_\_\_ % ймовірність використання ІІІ.

За результатами перевірки засвідчую, що:  
автор кваліфікаційної роботи \_\_\_\_\_:

- самостійно виконав(ла) кваліфікаційну роботу;
- коректно посилався(лась) на використані інформаційні джерела;
- в роботі відсутні ознаки академічної недоброчесності.

Керівник кваліфікаційної роботи \_\_\_\_\_ підпис \_\_\_\_\_  
(ПІП)

i **Метадані**📄 **ДОКУМЕНТ**

Заголовок

**2026\_126\_bakalavrska\_DemchukMV\_VoloshynVS .pdf**

Автор

**Демчук Марина Володимирівна**

Науковий керівник / Експерт

**Демчук Марина Володимирівна**

ІД документа

**334211660**🏢 **ОРГАНІЗАЦІЯ**

Назва організації

**National University of Water and Environmental Engineering**

підрозділ

**National University of Water and Environmental Engineering**📄 **ЗВІТ**

Дата звіту

**6/7/2026**

Дата редагування

---

**Обсяг знайдених подібностей**

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.

**6970**

Кількість слів

**56684**

Кількість символів

🔍 **Тривога**

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про **МОЖЛИВІ** маніпуляції в тексті. Спотворення в тексті можуть мати навісний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

|                        |     |    |
|------------------------|-----|----|
| Заміна букв            | 🔤   | 0  |
| Інтервали              | ⌈A→ | 0  |
| Мікропробіли           | ␣   | 2  |
| Білі знаки             | 🔤   | 0  |
| Парафрази (SmartMarks) | a   | 42 |

📄 **Джерела**

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Колір тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

**10 найдовших фраз**

Колір тексту

| # | НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ) | КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ<br>(ФРАГМЕНТІВ) |
|---|------------------------------------------|-------------------------------------------|
|---|------------------------------------------|-------------------------------------------|

## Реферат

Кваліфікаційна робота бакалавра складається з: 56 с., 47 рис., 4 табл., 15 літературних джерел.

**Актуальність теми** полягає в необхідності автоматизації діяльності закладів громадського харчування з метою підвищення ефективності управління, покращення обслуговування клієнтів та оптимізації бізнес-процесів.

**Об'єктом дослідження** є процеси організації та автоматизації діяльності кав'ярні.

**Предмет дослідження** – інформаційна система обліку діяльності кав'ярні, реалізована на основі WordPress та MySQL.

**Метою** бакалаврської роботи є розробка та реалізація комплексної інформаційної системи для автоматизації діяльності кав'ярні «CoffeeShop», яка забезпечує ефективне управління даними, підтримку процесу оформлення замовлень, ведення обліку та покращення взаємодії з клієнтами.

У роботі проведено аналіз діяльності кав'ярень, досліджено сучасні інформаційні системи автоматизації закладів громадського харчування та виконано порівняльний аналіз вебсайтів кав'ярень. На основі отриманих результатів розроблено логічну модель бази даних та створено інформаційну систему для кав'ярні «CoffeeShop».

У результаті виконання роботи розроблено вебсайт на платформі WordPress із використанням бази даних MySQL. Реалізовано функції перегляду інформації про заклад, публікації новин, оформлення замовлень, оплати та перегляду чеків. Для розробки та тестування використовувалися WordPress, MySQL, phpMyAdmin, XAMPP та Visual Studio Code.

Створена інформаційна система дозволяє автоматизувати основні бізнес-процеси кав'ярні, підвищити ефективність управління інформацією, покращити якість обслуговування клієнтів та забезпечити зручне адміністрування вебресурсу.

КЛЮЧОВІ СЛОВА: ІНФОРМАЦІЙНА СИСТЕМА, КАВ'ЯРНЯ,  
АВТОМАТИЗАЦІЯ, ВЕБСАЙТ, WordPress, MySQL, БАЗА ДАНИХ,  
ЗАМОВЛЕННЯ, КЛІЄНТИ, ОБЛІК

## ЗМІСТ

|                                                                                                                          |                  |
|--------------------------------------------------------------------------------------------------------------------------|------------------|
| <b><i>ВСТУП.....</i></b>                                                                                                 | <b><i>10</i></b> |
| <b><i>РОЗДІЛ 1. ІНФОРМАЦІЙНА ПІДТРИМКА ДІЯЛЬНОСТІ КАВ'ЯРЕНЬ..</i></b>                                                    | <b><i>12</i></b> |
| 1.1 Особливості діяльності кав'ярень в умовах військового стану ...                                                      | 12               |
| 1.2 Інформаційні системи для автоматизації діяльності закладів громадського харчування .....                             | 15               |
| 1.3 Порівняльна характеристика вебсайтів кав'ярень.....                                                                  | 20               |
| <b><i>РОЗДІЛ 2. РОЗРОБКА КОМПЛЕКСНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ<br/>ДЛЯ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ РОБОТИ КАВ'ЯРНІ .....</i></b> | <b><i>26</i></b> |
| 2.1. Логічна модель даних БД MySQL .....                                                                                 | 26               |
| 2.2. Інформаційна система обліку.....                                                                                    | 38               |
| Довідники в ІС .....                                                                                                     | 38               |
| Аналіз вихідної інформації .....                                                                                         | 51               |
| Інтерфейс користувача.....                                                                                               | 54               |
| 2.3. Функціональні можливості вебсайту кав'ярні.....                                                                     | 55               |
| <b><i>ВИСНОВКИ.....</i></b>                                                                                              | <b><i>61</i></b> |
| <b><i>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....</i></b>                                                                           | <b><i>63</i></b> |
| <b><i>ДОДАТКИ.....</i></b>                                                                                               | <b><i>64</i></b> |

## ВСТУП

**Актуальність теми:** у сучасних умовах цифровізації сфери громадського харчування використання інформаційних технологій стало важливим чинником підвищення ефективності діяльності закладів. Кав'ярні, як один із найпоширеніших видів закладів харчування, потребують швидкого обслуговування клієнтів, точного обліку продукції, контролю замовлень, зручної комунікації з відвідувачами та ефективного управління персоналом. Особливої актуальності ці питання набувають в умовах високої конкуренції та нестабільної економічної ситуації, пов'язаної з воєнним станом в Україні.

Впровадження комплексної інформаційної системи дає можливість автоматизувати основні бізнес-процеси кав'ярні, зменшити кількість помилок під час обробки даних, покращити якість обслуговування клієнтів та забезпечити оперативний доступ до необхідної інформації. Використання сучасних вебтехнологій і систем управління контентом дозволяє створити зручний вебсайт для взаємодії з клієнтами, а також інтегрувати його з базою даних для ведення обліку товарів, замовлень і працівників.

У межах даної бакалаврської роботи було розроблено комплексну інформаційну систему для кав'ярні «CoffeeShop», що включає вебсайт, створений на основі системи управління контентом WordPress, та базу даних MySQL для автоматизації процесів обліку й управління діяльністю закладу. Розробка такої системи є актуальним завданням, оскільки сприяє підвищенню ефективності роботи кав'ярні, оптимізації внутрішніх процесів та покращенню рівня обслуговування клієнтів.

**Об'єктом дослідження** є процеси організації та автоматизації діяльності кав'ярні, що охоплюють облік товарів і замовлень, управління персоналом, взаємодію з клієнтами та забезпечення інформаційної підтримки роботи закладу громадського харчування.

**Предметом дослідження** є розробка та впровадження комплексної інформаційної системи для підвищення ефективності роботи кав'ярні шляхом використання вебтехнологій, системи управління контентом WordPress та

бази даних MySQL.

Метою бакалаврської роботи є розробка та реалізація комплексної інформаційної системи для автоматизації діяльності кав'ярні «CoffeeShop», яка забезпечить ефективне управління даними, оптимізацію бізнес-процесів, покращення обслуговування клієнтів і підвищення загальної ефективності роботи закладу.

Для досягнення поставленої мети необхідно виконати такі завдання:

1. Провести аналіз особливостей діяльності кав'ярень в умовах сучасного ринку та воєнного стану.
2. Дослідити існуючі інформаційні системи автоматизації закладів громадського харчування та визначити їхні основні функціональні можливості.
3. Виконати порівняльний аналіз вебсайтів сучасних кав'ярень і визначити основні вимоги до структури та функціоналу вебресурсу.
4. Розробити логічну модель бази даних MySQL для забезпечення обліку діяльності кав'ярні.
5. Створити інформаційну систему обліку для автоматизації основних процесів роботи закладу.
6. Розробити вебсайт кав'ярні «CoffeeShop» на основі WordPress та реалізувати його основні функціональні можливості.
7. Оцінити ефективність використання розробленої інформаційної системи та визначити переваги її впровадження для діяльності кав'ярні.

# **РОЗДІЛ 1. ІНФОРМАЦІЙНА ПІДТРИМКА ДІЯЛЬНОСТІ КАВ'ЯРЕНЬ**

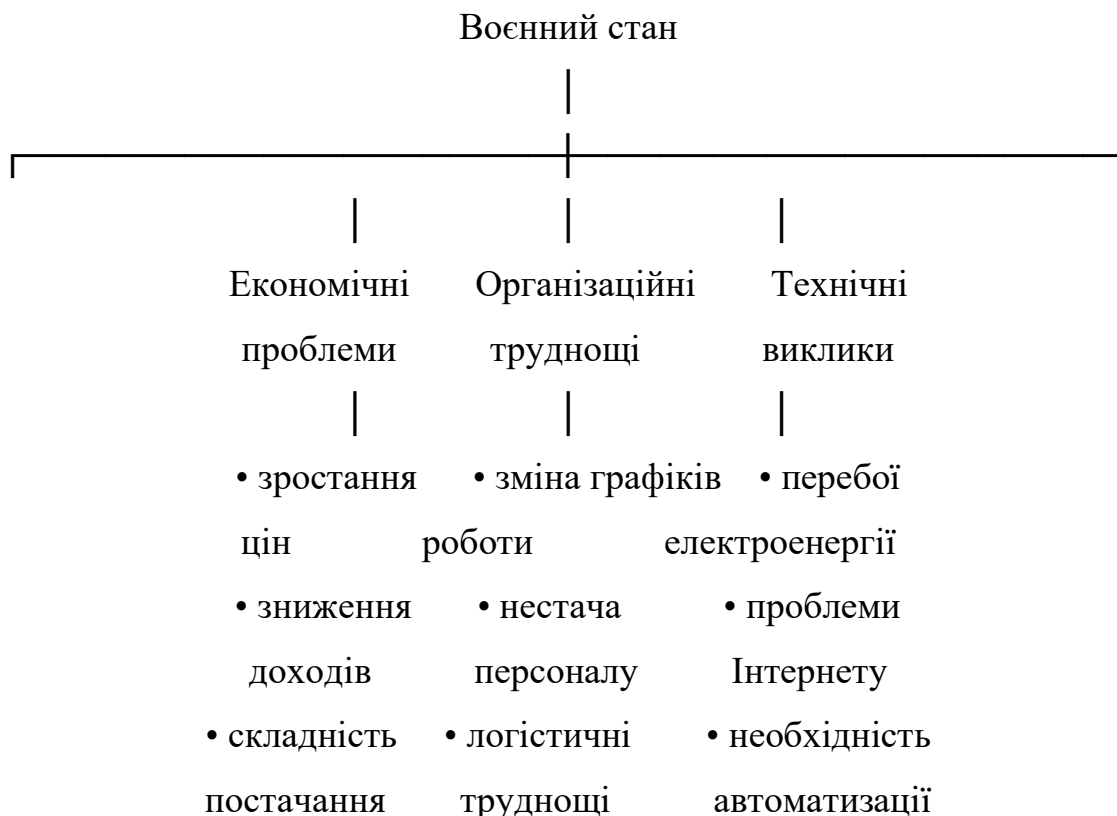
## **1.1 Особливості діяльності кав'ярень в умовах військового стану**

У сучасних умовах воєнного стану діяльність закладів громадського харчування зазнала значних змін. Кав'ярні, як один із найпоширеніших типів малого бізнесу, змушені адаптуватися до нових економічних, соціальних та організаційних умов. Основними викликами для роботи кав'ярень стали перебої з постачанням продукції, зменшення купівельної спроможності населення, нестабільність енергопостачання, зміни в логістиці, необхідність забезпечення безпеки працівників і клієнтів, а також посилення конкуренції на ринку.

В умовах військового стану ефективність роботи кав'ярні значною мірою залежить від використання сучасних інформаційних технологій та систем автоматизації. Інформаційна підтримка діяльності закладу дозволяє оперативно контролювати замовлення, вести облік товарів і фінансів, організовувати роботу персоналу та підтримувати комунікацію з клієнтами через вебсайт і соціальні мережі.

Однією з важливих особливостей роботи кав'ярень у цей період є необхідність швидкого реагування на зміни зовнішнього середовища. Через нестабільну ситуацію підприємства повинні оперативно змінювати графіки роботи, оновлювати меню, контролювати залишки продукції та адаптувати способи обслуговування клієнтів. Саме тому використання комплексних інформаційних систем стало необхідним елементом ефективного функціонування сучасної кав'ярні.

Основні фактори, які впливають на діяльність кав'ярень в умовах військового стану, наведено на рисунку 1.1.



**Рис. 1.1 – Основні фактори впливу на діяльність кав'ярень в умовах військового стану**

У сучасних умовах важливу роль відіграє автоматизація бізнес-процесів. Завдяки використанню вебсайтів, CRM-систем, електронного меню, онлайн-замовлень та баз даних кав'ярні можуть значно покращити якість обслуговування клієнтів і зменшити навантаження на працівників.

Особливого значення набуває використання вебсайту кав'ярні як інструмента взаємодії з клієнтами. Вебресурс дозволяє:

- переглядати меню та ціни;
- оформлювати онлайн-замовлення;
- отримувати інформацію про акції та новини;
- переглядати графік роботи;
- забезпечувати зворотний зв'язок із клієнтами.

Крім того, інформаційні системи забезпечують централізоване зберігання даних і дозволяють керівництву кав'ярні отримувати оперативну інформацію

про діяльність закладу.

У таблиці 1.1 наведено основні проблеми діяльності кав'ярень під час воєнного стану та можливі шляхи їх вирішення за допомогою інформаційних технологій.

Таблиця 1.1

| <b>Проблема</b>                 | <b>Наслідки для кав'ярні</b>                   | <b>Інформаційне рішення</b>                 |
|---------------------------------|------------------------------------------------|---------------------------------------------|
| Перебої електроенергії          | Затримка обслуговування                        | Хмарні сервіси та резервне збереження даних |
| Проблеми з постачанням          | Відсутність товарів                            | Автоматизований облік залишків              |
| Зменшення кількості клієнтів    | Зниження прибутку                              | Онлайн-реклама та вебсайт                   |
| Велике навантаження на персонал | Помилки в роботі                               | Автоматизація замовлень                     |
| Швидкі зміни умов роботи        | Необхідність оперативного оновлення інформації | Система управління контентом WordPress      |

Таким чином, в умовах воєнного стану інформаційна підтримка діяльності кав'ярень є важливим інструментом забезпечення стабільної роботи закладу. Використання сучасних інформаційних систем дозволяє автоматизувати основні бізнес-процеси, покращити якість обслуговування клієнтів, оптимізувати управління ресурсами та підвищити конкурентоспроможність кав'ярні на ринку. Розробка комплексної інформаційної системи для кав'ярні «CoffeeShop» є актуальним і практично значущим рішенням для підвищення ефективності діяльності закладу в сучасних умовах.

## 1.2 Інформаційні системи для автоматизації діяльності закладів громадського харчування

У сучасних умовах розвитку ресторанного бізнесу автоматизація діяльності закладів громадського харчування стала необхідною складовою ефективного управління. Використання інформаційних систем дозволяє оптимізувати процеси обслуговування клієнтів, вести облік товарів і фінансів, контролювати роботу персоналу, аналізувати продажі та підвищувати якість сервісу. Особливо актуальним впровадження таких систем є для кав'ярень, де швидкість обслуговування та точність обліку мають важливе значення.

Сучасні інформаційні системи для автоматизації закладів громадського харчування поєднують функції POS-систем, складського обліку, CRM-систем, аналітики та управління замовленнями. Більшість таких систем працюють у хмарному середовищі, що дозволяє отримувати доступ до даних із будь-якого пристрою та забезпечує збереження інформації навіть у разі технічних проблем.

Серед найпоширеніших систем автоматизації закладів громадського харчування можна виділити **Poster POS**, **SkyService POS** та **Restora POS**.

### 1. Система автоматизації Poster POS

Однією з найвідоміших інформаційних систем для автоматизації кав'ярень і ресторанів є **Poster POS** (URL: <https://joinposter.com/ua>). Дана система призначена для автоматизації роботи кафе, кав'ярень, барів та інших закладів громадського харчування. Вона забезпечує ведення касових операцій, складського обліку, фінансової аналітики та управління персоналом.

**Poster POS** підтримує роботу на планшетах і смартфонах, що дозволяє використовувати систему без дорогого спеціалізованого обладнання. Система забезпечує:

- автоматизацію продажів;
- контроль залишків товарів;
- формування статистики продажів;
- ведення фінансового обліку;

- програми лояльності для клієнтів;
- інтеграцію з ПРРО та онлайн-замовленнями.

Важливою перевагою **Poster POS** є можливість роботи навіть за відсутності Інтернету або електроенергії, що особливо актуально в умовах воєнного стану в Україні.

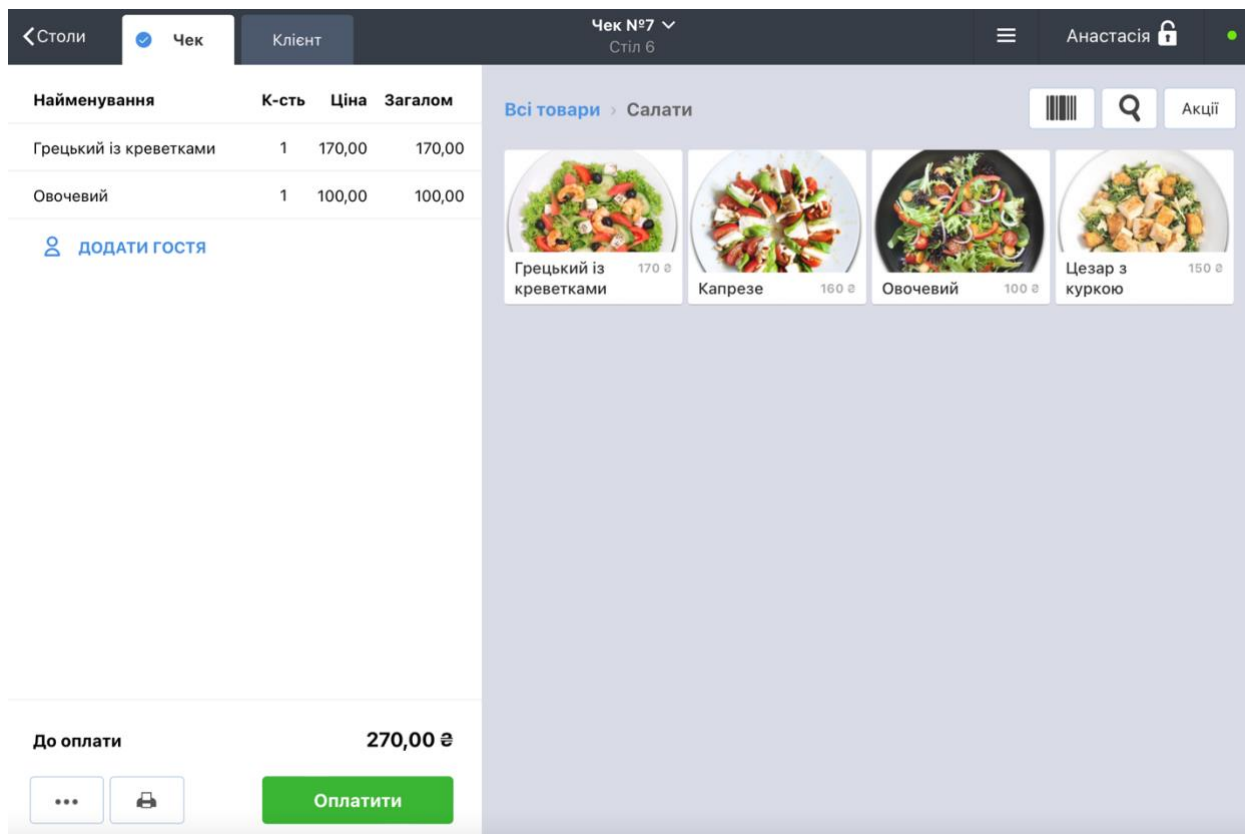


Рис. 1.2 Головна сторінка системи Poster POS

## 2. Система автоматизації SkyService POS

Ще однією популярною системою автоматизації є **SkyService POS**(URL: <https://skyservice.pro/uk/main/>). Дана система орієнтована на автоматизацію ресторанів, кафе, кав'ярень та служб доставки. Основною перевагою системи є універсальність та підтримка роботи на різних платформах.

**SkyService POS** дозволяє:

- вести облік продажів;
- контролювати складські операції;
- автоматизувати роботу офіціантів;
- створювати програми лояльності;
- аналізувати фінансові показники;

- організовувати доставку замовлень.

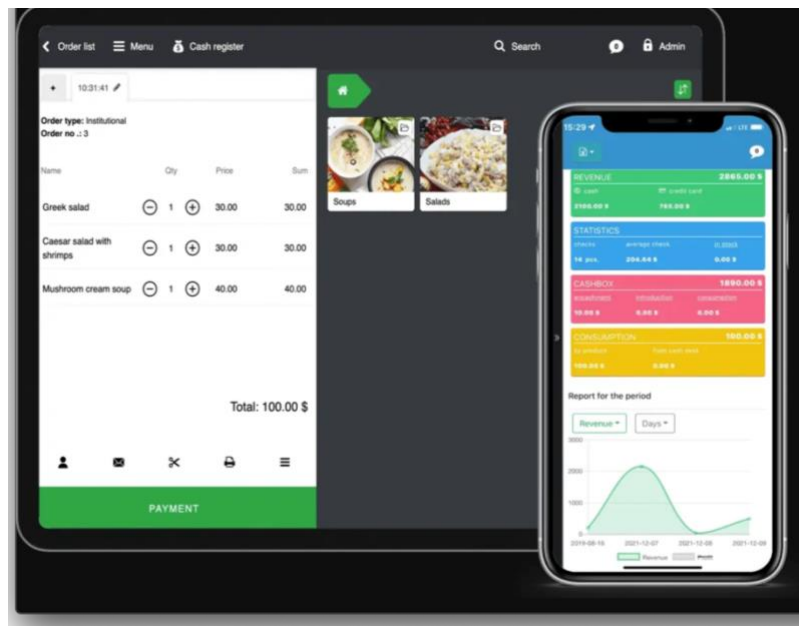


Рис. 1.3 Інтерфейс системи SkyService POS

Система також містить POS-термінал для швидкого оформлення замовлень і передачі їх на кухню або бар. Завдяки цьому зменшується кількість помилок і прискорюється процес обслуговування клієнтів.

### 3. Система автоматизації Restora POS

Для автоматизації роботи закладів громадського харчування також використовується система **Restora POS**(URL: <https://restorapos.com/>). Дана система забезпечує комплексне управління ресторанним бізнесом та підтримує роботу закладів різного масштабу.



Рис. 1.4 Інтерфейс системи Restora POS

**Restora POS** включає:

- систему POS-обслуговування;
- складський облік;
- формування звітності;
- онлайн-замовлення;
- мобільний додаток;

систему управління кухнею (KDS).

Особливістю **Restora POS** є орієнтація на комплексне управління всіма процесами ресторану в єдиній системі. Це дозволяє підвищити ефективність роботи персоналу та покращити контроль за діяльністю закладу.

Для порівняння основних функціональних можливостей розглянутих систем доцільно використати таблицю 1.2.

Таблиця 1.2 – Порівняльна характеристика інформаційних систем автоматизації

| Функції системи            | Poster POS | SkyService POS | Restora POS |
|----------------------------|------------|----------------|-------------|
| POS-система                | +          | +              | +           |
| Складський облік           | +          | +              | +           |
| Фінансова аналітика        | +          | +              | +           |
| Онлайн-замовлення          | +          | +              | +           |
| CRM та програми лояльності | +          | +              | +           |
| Робота на планшетах        | +          | +              | +           |
| Хмарне зберігання даних    | +          | +              | +           |
| Мобільний додаток          | +          | +              | +           |
| Підтримка доставки         | +          | +              | +           |

Таким чином, сучасні інформаційні системи для автоматизації закладів громадського харчування забезпечують комплексний підхід до управління діяльністю кав'ярень і ресторанів. Використання таких систем дозволяє значно підвищити ефективність роботи закладу, оптимізувати процеси обслуговування клієнтів, зменшити кількість помилок та забезпечити оперативний доступ до інформації. Аналіз існуючих рішень показує, що найбільш важливими функціями сучасних систем є автоматизація обліку, фінансова аналітика, контроль персоналу та інтеграція з вебсайтами й онлайн-сервісами. Саме ці можливості були враховані під час розробки інформаційної системи для кав'ярні «CoffeeShop».

### **1.3 Порівняльна характеристика вебсайтів кав'ярень**

У сучасних умовах вебсайт є важливою складовою діяльності будь-якого закладу громадського харчування. Для кав'ярень вебресурс виконує не лише інформаційну функцію, а й виступає інструментом маркетингу, онлайн-комунікації з клієнтами та організації замовлень. Якісний вебсайт дозволяє підвищити впізнаваність бренду, забезпечити доступність інформації про меню та послуги, а також покращити рівень обслуговування клієнтів.

Під час дослідження було проведено аналіз вебсайтів популярних кав'ярень та мереж закладів громадського харчування. Для порівняння були обрані такі вебсайти:

- **Starbucks**
- **Aroma Kava**
- **Lviv Croissants**
- **Coffee House**

Основною метою аналізу є визначення функціональних можливостей сучасних вебсайтів кав'ярень, їхнього дизайну, структури та зручності використання.

#### **1. Аналіз вебсайту Starbucks**

Вебсайт компанії **Starbucks**(URL: <https://www.starbucks.com/>) є прикладом сучасного корпоративного вебресурсу з високим рівнем функціональності та адаптивного дизайну. Головна сторінка містить інформацію про продукцію, сезонні пропозиції, акції та новини компанії.

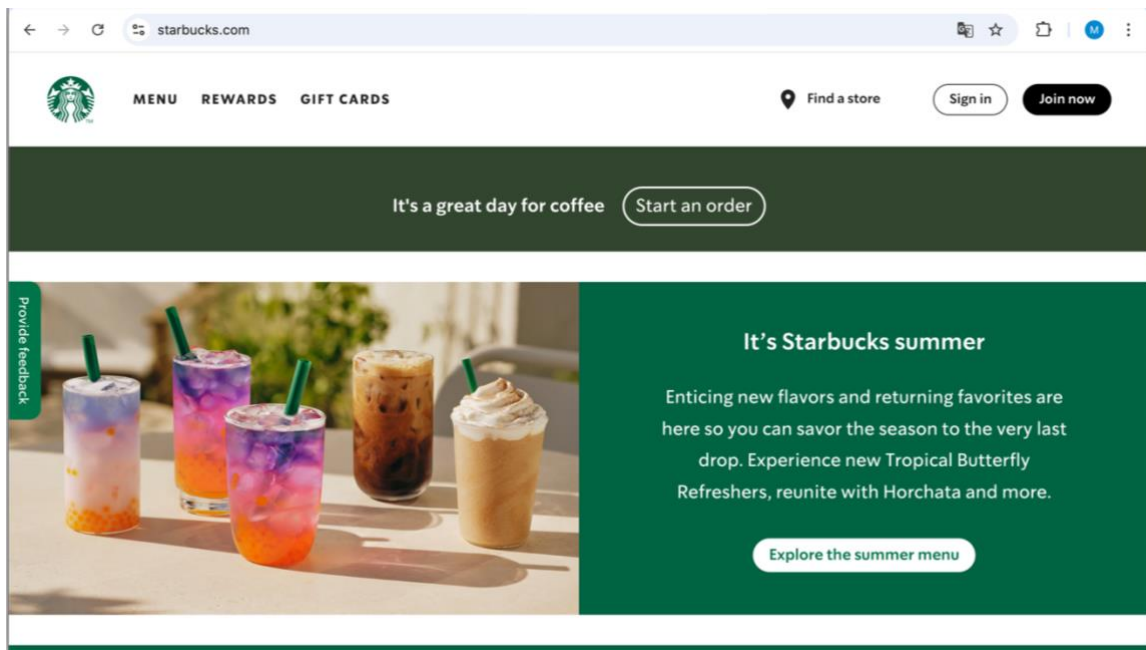


Рис. 1.5 Головна сторінка вебсайту **Starbucks**

До основних функцій сайту належать:

- онлайн-перегляд меню;
- пошук закладів на карті;
- можливість онлайн-замовлення;
- адаптивний дизайн для мобільних пристроїв;
- інтеграція із соціальними мережами.

Особливістю сайту є мінімалістичний стиль оформлення, швидка навігація та зручний інтерфейс користувача.

## 2. Аналіз вебсайту **Aroma Kava**

Вебсайт **Aroma Kava**(URL: <https://aromakava.ua/>) орієнтований на інформування клієнтів про асортимент продукції, акції та мережу кав'ярень. Сайт має сучасний дизайн і просту структуру.

Основні функціональні можливості:

- перегляд меню;
- інформація про адреси кав'ярень;
- новини та акції;
- контакти та зворотний зв'язок;
- адаптація під мобільні пристрої.

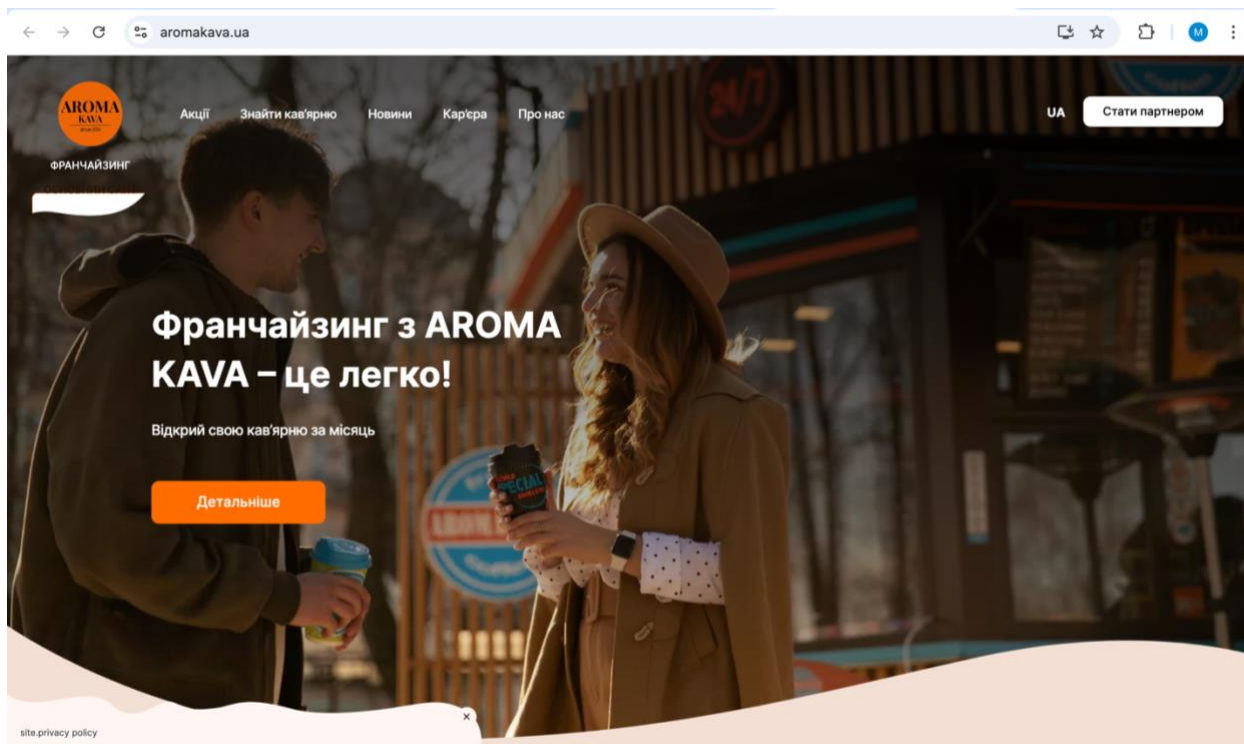


Рис. 1.6 Інтерфейс вебсайту **Aroma Kava**

Перевагою сайту є яскравий візуальний стиль та простота використання.

### 3. Аналіз вебсайту **Lviv Croissants**

Вебсайт **Lviv Croissants**(URL: <https://lvivcroissants.com/ua/>) поєднує функції інформаційного ресурсу та онлайн-сервісу для замовлення продукції. Сайт містить детальну інформацію про меню, франшизу, вакансії та доставку.

Основні можливості сайту:

- онлайн-замовлення;
- інтерактивне меню;
- інформація про заклади;
- особистий кабінет користувача;
- підтримка доставки.

Сайт характеризується сучасним дизайном, великою кількістю графічних елементів та зручною навігацією.

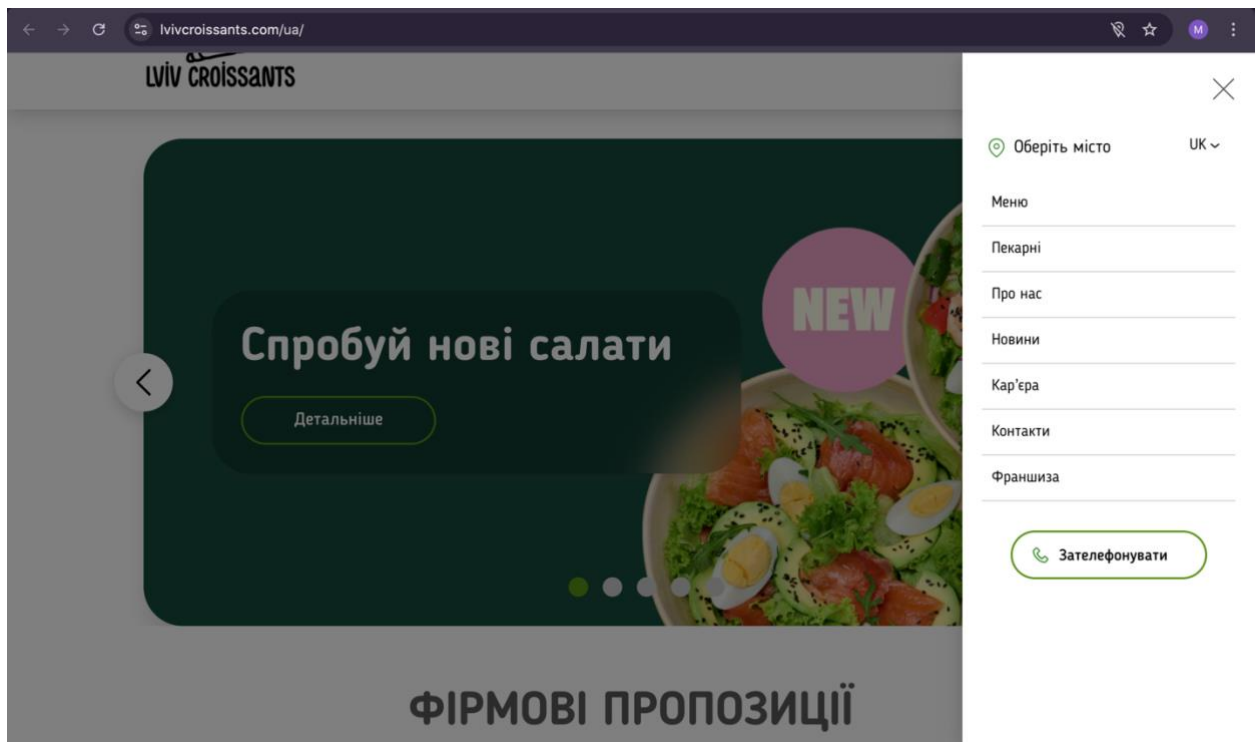


Рис. 1.7 Головна сторінка вебсайту **Lviv Croissants**

#### 4. Аналіз вебсайту **Coffee House**

Вебсайт **Coffee House**(URL: <https://coffee-house.org/ua/>) містить інформацію про меню, мережу закладів та новини компанії. Основна увага приділена презентації продукції та бренду.

Функціональні можливості:

- перегляд асортименту;
- інформація про кав'ярні;
- форма зворотного зв'язку;
- адаптивний дизайн;
- інтеграція із соціальними мережами.

Сайт має просту структуру та зручний користувацький інтерфейс.

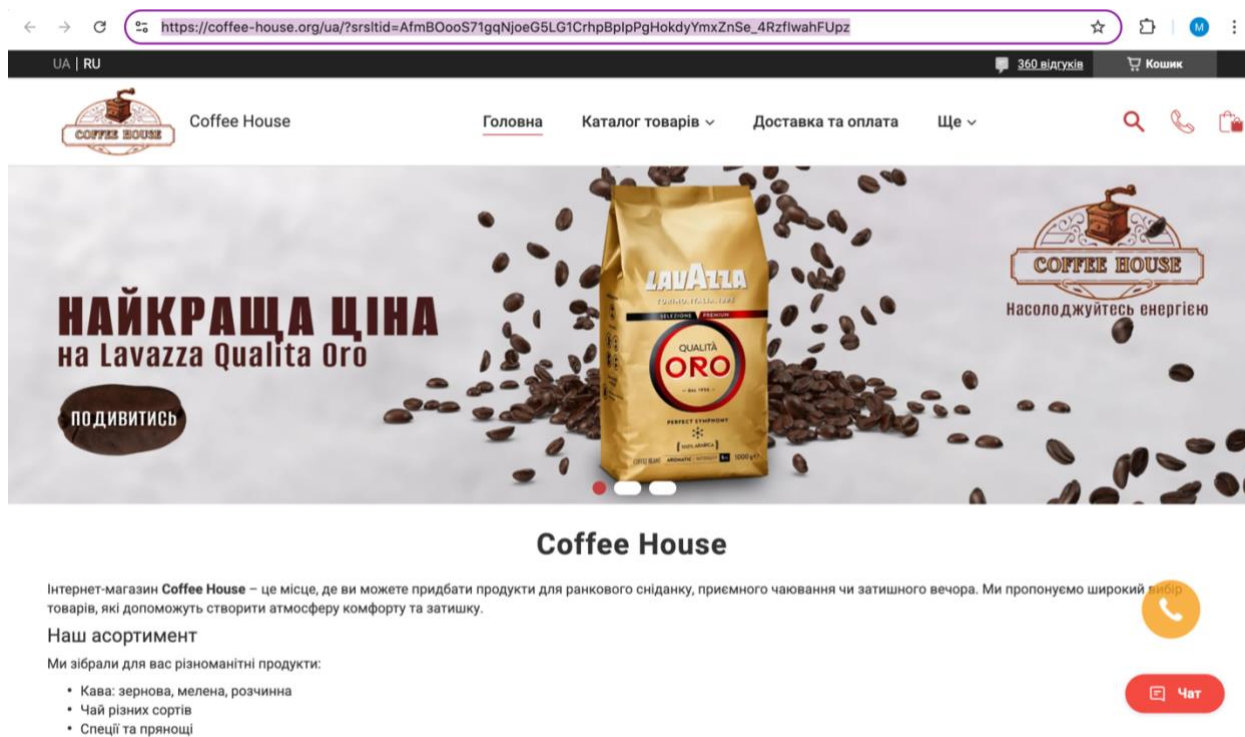


Рис. 1.8 Інтерфейс вебсайту Coffee House

Для більш наочного аналізу доцільно виконати порівняння функціональних можливостей досліджуваних вебсайтів.

Таблиця 1.3 – Порівняльна характеристика вебсайтів кав'ярень

| Характеристика            | Starbucks | Aroma Kava | Lviv Croissants | Coffee House |
|---------------------------|-----------|------------|-----------------|--------------|
| Адаптивний дизайн         | +         | +          | +               | +            |
| Онлайн-замовлення         | +         | —          | +               | —            |
| Перегляд меню             | +         | +          | +               | +            |
| Пошук закладів            | +         | +          | +               | +            |
| Інтеграція із соцмережами | +         | +          | +               | +            |
| Особистий кабінет         | +         | —          | +               | —            |
| Інформація про акції      | +         | +          | +               | +            |
| Сучасний дизайн           | +         | +          | +               | +            |

На основі проведеного аналізу можна зробити висновок, що сучасні вебсайти кав'ярень орієнтовані на забезпечення зручної взаємодії з клієнтами,

підтримку онлайн-замовлень та швидкий доступ до інформації. Основними тенденціями розвитку таких вебресурсів є адаптивний дизайн, інтеграція з онлайн-сервісами, автоматизація процесів обслуговування та використання сучасних систем управління контентом.

Під час розробки вебсайту кав'ярні «CoffeeShop» були враховані переваги досліджених вебресурсів, зокрема адаптивний інтерфейс, зручна структура меню, інтеграція з базою даних та можливість швидкого оновлення інформації за допомогою системи управління контентом WordPress.

## РОЗДІЛ 2. РОЗРОБКА КОМПЛЕКСНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ РОБОТИ КАВ'ЯРНІ

### 2.1. Логічна модель даних БД MySQL

Для забезпечення роботи інформаційної системи було розроблено логічну модель бази даних у середовищі MySQL. Структура бази даних призначена для накопичення, зберігання та обробки інформації про працівників закладу, клієнтів, товари, замовлення та робочі зміни персоналу. Використання централізованої бази даних дає змогу підтримувати цілісність інформації та спрощує виконання операцій із даними.

Під час проєктування логічної моделі основну увагу було приділено визначенню переліку сутностей, їхніх атрибутів і взаємозв'язків між ними. Для усунення дублювання інформації та забезпечення коректності зберігання даних структура бази була нормалізована.

До складу бази даних входить набір взаємопов'язаних таблиць, кожна з яких відповідає за зберігання певної категорії інформації. Основні таблиці містять відомості про працівників, робочі зміни, клієнтів, столи закладу, товари, категорії товарів, замовлення та їх складові.

Перелік основних таблиць бази даних та їхніх полів наведено в таблиці 2.1.

Таблиця 2.1

Таблиці БД

| Назви таблиці | Назви полів                                                                                                                         |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------|
| Employees     | <ul style="list-style-type: none"><li>– Id</li><li>– Login</li><li>– Name</li><li>– DateOfEmployment</li><li>– HourlyRate</li></ul> |
| WorkShifts    | <ul style="list-style-type: none"><li>– Id</li></ul>                                                                                |

|            |                                                                                                                                                                                                 |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|            | <ul style="list-style-type: none"> <li>– EmployeeId</li> <li>– Start</li> <li>– End</li> </ul>                                                                                                  |
| Clients    | <ul style="list-style-type: none"> <li>– Id</li> <li>– Name</li> <li>– HasDiscount</li> <li>– Comment</li> </ul>                                                                                |
| Tables     | <ul style="list-style-type: none"> <li>– Id</li> <li>– Placement</li> <li>– Seates</li> </ul>                                                                                                   |
| Orders     | <ul style="list-style-type: none"> <li>– Id</li> <li>– WorkShiftId</li> <li>– ClientId</li> <li>– TableId</li> <li>– Created</li> <li>– Comment</li> <li>– is_paid</li> <li>– status</li> </ul> |
| OrderItems | <ul style="list-style-type: none"> <li>– OrderId</li> <li>– ItemId</li> </ul>                                                                                                                   |
| Items      | <ul style="list-style-type: none"> <li>– Id</li> <li>– CategoryId</li> <li>– Name</li> <li>– Price</li> <li>– DiscountPrice</li> <li>– Available</li> <li>– Profit</li> </ul>                   |
| Categories | <ul style="list-style-type: none"> <li>– Id</li> </ul>                                                                                                                                          |

|       |                                                                                                                |
|-------|----------------------------------------------------------------------------------------------------------------|
|       | <ul style="list-style-type: none"> <li>– Name</li> </ul>                                                       |
| Users | <ul style="list-style-type: none"> <li>– id</li> <li>– username</li> <li>– password</li> <li>– role</li> </ul> |

Між таблицями бази даних встановлено систему взаємозв'язків, що відповідає логіці роботи кав'ярні. Основним типом зв'язків є «один до багатьох». Наприклад, одному працівнику може відповідати декілька робочих змін, одному клієнту — кілька замовлень, а одній категорії — багато товарів.

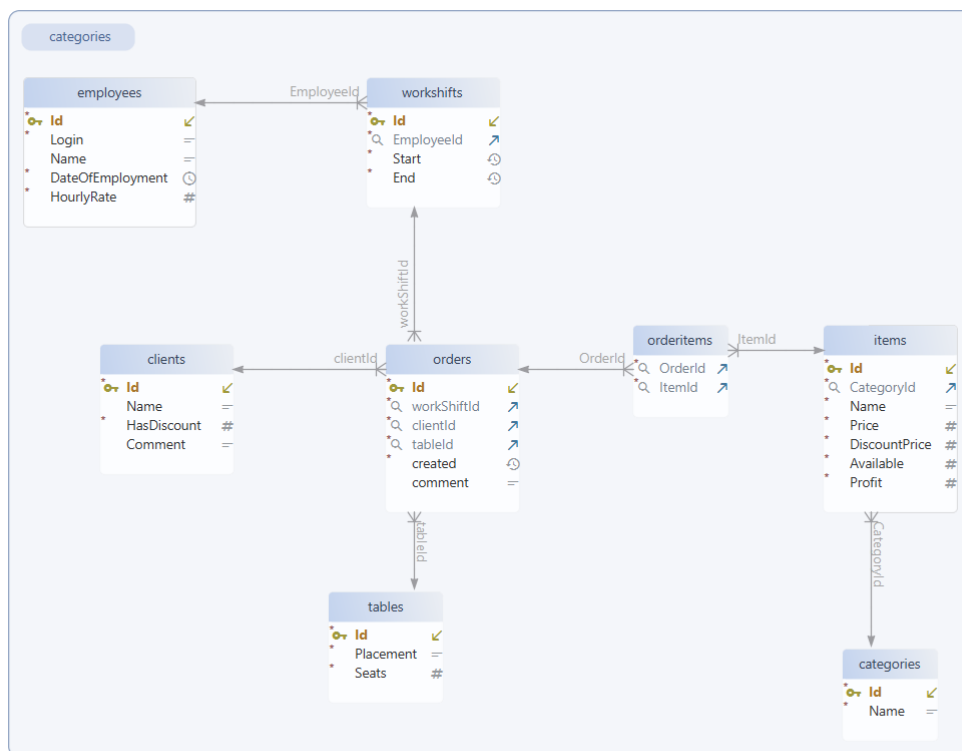


Рис. 2.1 – Логічна модель даних інформаційної системи кав'ярні.

Для зв'язку між замовленнями та товарами використано модель «багато до багатьох», яка реалізована за допомогою таблиці OrderItems. Такий підхід забезпечує гнучкість структури даних і дозволяє формувати замовлення будь-якої складності.

Зв'язок «один до багатьох» реалізований між таблицями Employees та WorkShifts. Це пояснюється тим, що одному працівнику може відповідати кілька робочих змін, тоді як кожна зміна належить лише одному співробітнику. Аналогічний зв'язок використовується між таблицями WorkShifts та Orders, де одна робоча зміна може містити декілька створених замовлень.

Між таблицями Clients та Orders також встановлено зв'язок «один до багатьох». Один клієнт має можливість оформлювати багато замовлень, проте кожне конкретне замовлення пов'язується лише з одним клієнтом. За таким самим принципом організовано взаємодію таблиць Tables та Orders, оскільки за одним столиком протягом часу може бути оформлено багато замовлень.

Для класифікації асортименту використовується взаємозв'язок між таблицями Categories та Items. Одна категорія може об'єднувати декілька товарів, але кожен товар належить лише до однієї категорії.

Зв'язок типу «багато до багатьох» між таблицями Orders та Items реалізовано через допоміжну таблицю OrderItems. Завдяки цьому одне замовлення може містити кілька товарних позицій, а кожен товар може бути включений до різних замовлень. Така структура забезпечує гнучкість обліку та спрощує формування аналітичної інформації.

Використання зовнішніх ключів дозволяє підтримувати коректність зв'язків між таблицями, запобігати появі неузгоджених записів та забезпечувати надійне функціонування інформаційної системи.

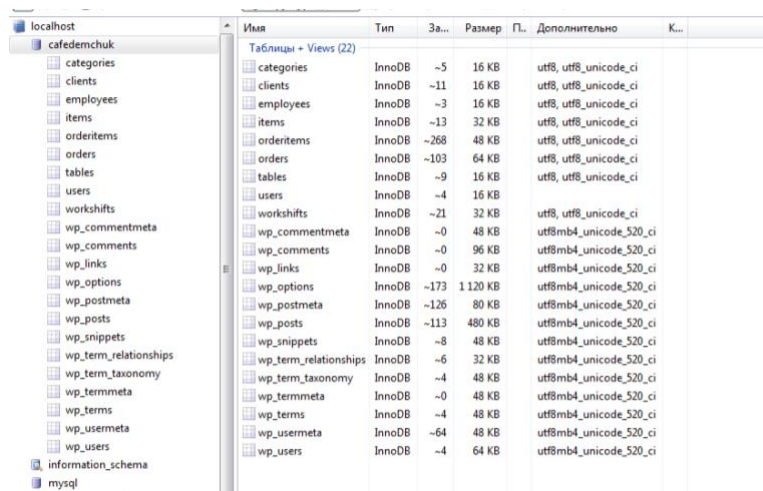
Таким чином, розроблена логічна модель бази даних повністю відображає структуру та бізнес-процеси кав'ярні, забезпечує ефективне зберігання інформації та створює основу для подальшої роботи інформаційної системи.

### 1.3 База даних інформаційної системи

Структура бази даних розроблена відповідно до особливостей предметної області та охоплює основні процеси діяльності кав'ярні, зокрема управління працівниками, облік клієнтів, ведення асортименту товарів, формування

замовлень та контроль робочих змін персоналу. Всі дані зберігаються у вигляді взаємопов'язаних таблиць, що дозволяє уникнути дублювання інформації та забезпечити її цілісність.

У процесі проєктування логічної моделі були визначені основні сутності предметної області, їхні характеристики та взаємозв'язки між ними. Для підвищення ефективності роботи системи структура бази даних була нормалізована, що дозволило мінімізувати надлишковість даних і спростити їх подальше обслуговування.

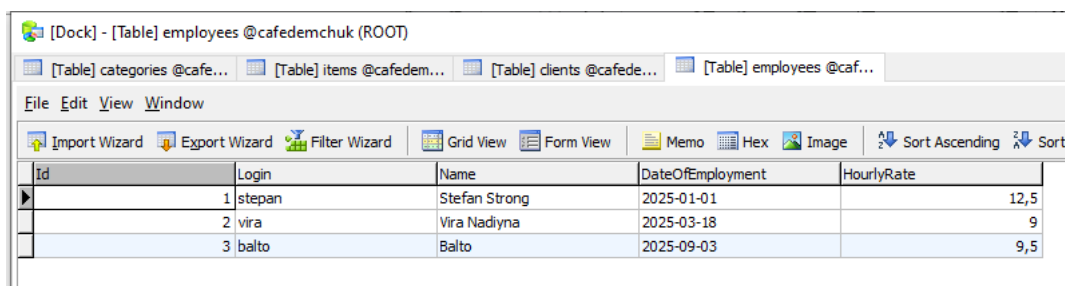


| Имя                   | Тип    | За... | Размер   | П... | Дополнительно          | К... |
|-----------------------|--------|-------|----------|------|------------------------|------|
| categories            | InnoDB | ~5    | 16 KB    |      | utf8, utf8_unicode_ci  |      |
| clients               | InnoDB | ~11   | 16 KB    |      | utf8, utf8_unicode_ci  |      |
| employees             | InnoDB | ~3    | 16 KB    |      | utf8, utf8_unicode_ci  |      |
| items                 | InnoDB | ~13   | 32 KB    |      | utf8, utf8_unicode_ci  |      |
| orderitems            | InnoDB | ~268  | 48 KB    |      | utf8, utf8_unicode_ci  |      |
| orders                | InnoDB | ~103  | 64 KB    |      | utf8, utf8_unicode_ci  |      |
| tables                | InnoDB | ~9    | 16 KB    |      | utf8, utf8_unicode_ci  |      |
| users                 | InnoDB | ~4    | 16 KB    |      | utf8, utf8_unicode_ci  |      |
| workshifts            | InnoDB | ~21   | 32 KB    |      | utf8, utf8_unicode_ci  |      |
| wp_commentmeta        | InnoDB | ~0    | 48 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_comments           | InnoDB | ~0    | 96 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_links              | InnoDB | ~0    | 32 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_options            | InnoDB | ~173  | 1 120 KB |      | utf8mb4_unicode_520_ci |      |
| wp_postmeta           | InnoDB | ~126  | 80 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_posts              | InnoDB | ~113  | 480 KB   |      | utf8mb4_unicode_520_ci |      |
| wp_snippets           | InnoDB | ~8    | 48 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_term_relationships | InnoDB | ~6    | 32 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_term_taxonomy      | InnoDB | ~4    | 48 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_termmeta           | InnoDB | ~0    | 48 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_terms              | InnoDB | ~4    | 48 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_usermeta           | InnoDB | ~64   | 48 KB    |      | utf8mb4_unicode_520_ci |      |
| wp_users              | InnoDB | ~4    | 64 KB    |      | utf8mb4_unicode_520_ci |      |

Рис.2.2 Таблиці БД

До складу бази даних входять таблиці Employees, WorkShifts, Clients, Tables, Orders, OrderItems, Items, Categories та Users. Кожна з них виконує окрему функцію та відповідає за зберігання певного набору даних.

Таблиця Employees використовується для зберігання відомостей про співробітників закладу. У ній містяться дані про обліковий запис працівника, його ім'я, дату працевлаштування та погодинну оплату праці. Інформація з цієї таблиці використовується під час формування робочих змін та здійснення контролю за персоналом.



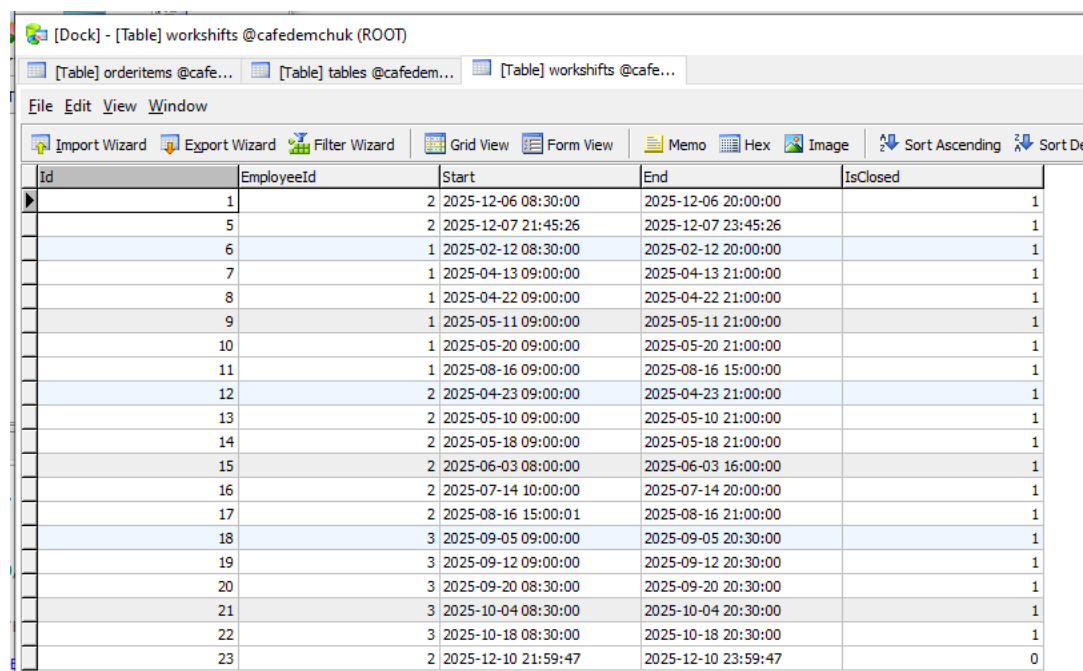
| Id | Login  | Name          | DateOfEmployment | HourlyRate |
|----|--------|---------------|------------------|------------|
| 1  | stepan | Stefan Strong | 2025-01-01       | 12,5       |
| 2  | vira   | Vira Nadiyna  | 2025-03-18       | 9          |
| 3  | balto  | Balto         | 2025-09-03       | 9,5        |

Рис. 2.3 Таблиця «Employees»

SQL запит для створення таблиці Employees:

```
CREATE TABLE `employees` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `Login` varchar(128) COLLATE utf8_unicode_ci NOT NULL,
  `Name` varchar(256) COLLATE utf8_unicode_ci DEFAULT NULL,
  `DateOfEmployment` date NOT NULL,
  `HourlyRate` decimal(10,2) NOT NULL,
  PRIMARY KEY (`Id`)
) ENGINE=InnoDB AUTO_INCREMENT=9 DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci
```

Таблиця WorkShifts призначена для обліку змін працівників. Для кожної зміни фіксуються дата і час початку роботи, час завершення зміни та працівник, який її виконував. Це дозволяє здійснювати контроль робочого часу та аналізувати ефективність роботи персоналу.



| Id | EmployeeId | Start               | End                 | IsClosed |
|----|------------|---------------------|---------------------|----------|
| 1  | 2          | 2025-12-06 08:30:00 | 2025-12-06 20:00:00 | 1        |
| 5  | 2          | 2025-12-07 21:45:26 | 2025-12-07 23:45:26 | 1        |
| 6  | 1          | 2025-02-12 08:30:00 | 2025-02-12 20:00:00 | 1        |
| 7  | 1          | 2025-04-13 09:00:00 | 2025-04-13 21:00:00 | 1        |
| 8  | 1          | 2025-04-22 09:00:00 | 2025-04-22 21:00:00 | 1        |
| 9  | 1          | 2025-05-11 09:00:00 | 2025-05-11 21:00:00 | 1        |
| 10 | 1          | 2025-05-20 09:00:00 | 2025-05-20 21:00:00 | 1        |
| 11 | 1          | 2025-08-16 09:00:00 | 2025-08-16 15:00:00 | 1        |
| 12 | 2          | 2025-04-23 09:00:00 | 2025-04-23 21:00:00 | 1        |
| 13 | 2          | 2025-05-10 09:00:00 | 2025-05-10 21:00:00 | 1        |
| 14 | 2          | 2025-05-18 09:00:00 | 2025-05-18 21:00:00 | 1        |
| 15 | 2          | 2025-06-03 08:00:00 | 2025-06-03 16:00:00 | 1        |
| 16 | 2          | 2025-07-14 10:00:00 | 2025-07-14 20:00:00 | 1        |
| 17 | 2          | 2025-08-16 15:00:01 | 2025-08-16 21:00:00 | 1        |
| 18 | 3          | 2025-09-05 09:00:00 | 2025-09-05 20:30:00 | 1        |
| 19 | 3          | 2025-09-12 09:00:00 | 2025-09-12 20:30:00 | 1        |
| 20 | 3          | 2025-09-20 08:30:00 | 2025-09-20 20:30:00 | 1        |
| 21 | 3          | 2025-10-04 08:30:00 | 2025-10-04 20:30:00 | 1        |
| 22 | 3          | 2025-10-18 08:30:00 | 2025-10-18 20:30:00 | 1        |
| 23 | 2          | 2025-12-10 21:59:47 | 2025-12-10 23:59:47 | 0        |

Рис. 2.4 Таблиця «WorkShifts»

Нижче наведено SQL запит для створення таблиці WorkShifts:

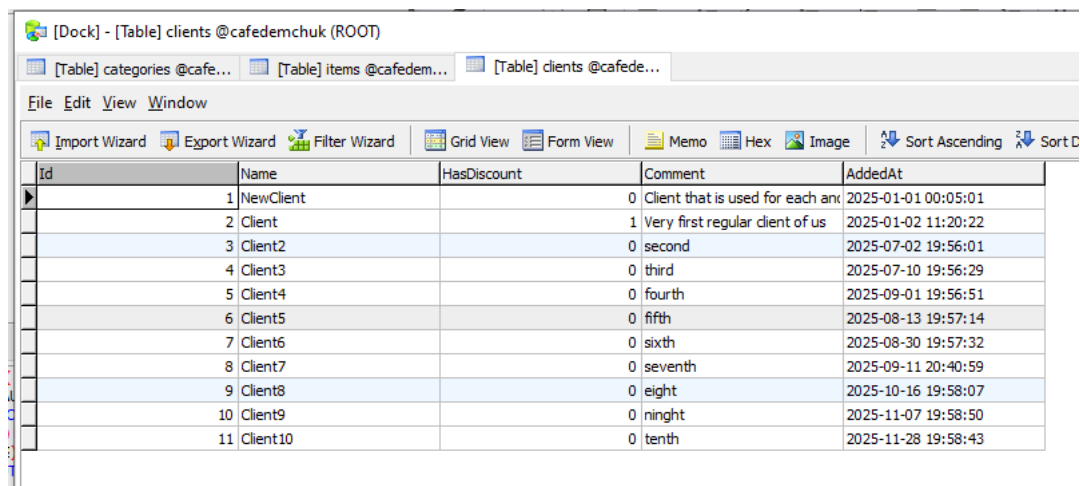
```
CREATE TABLE `workshifts` (
```

```

`Id` int(11) NOT NULL AUTO_INCREMENT,
`EmployeeId` int(11) NOT NULL,
`Start` datetime NOT NULL,
`End` datetime NOT NULL,
`IsClosed` tinyint(4) NOT NULL DEFAULT 0,
PRIMARY KEY (`Id`),
KEY `employee` (`EmployeeId`),
CONSTRAINT `employee` FOREIGN KEY (`EmployeeId`) REFERENCES
`employees` (`Id`) ON UPDATE CASCADE
) ENGINE=InnoDB AUTO_INCREMENT=25 DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci

```

Відомості про клієнтів зберігаються в таблиці Clients. Крім імені клієнта, таблиця містить інформацію про наявність персональної знижки та додаткові примітки. Використання цієї таблиці дає можливість вести історію взаємодії з постійними відвідувачами та застосовувати індивідуальні умови обслуговування.



| Id | Name      | HasDiscount | Comment                         | AddedAt             |
|----|-----------|-------------|---------------------------------|---------------------|
| 1  | NewClient | 0           | Client that is used for each an | 2025-01-01 00:05:01 |
| 2  | Client    | 1           | Very first regular client of us | 2025-01-02 11:20:22 |
| 3  | Client2   | 0           | second                          | 2025-07-02 19:56:01 |
| 4  | Client3   | 0           | third                           | 2025-07-10 19:56:29 |
| 5  | Client4   | 0           | fourth                          | 2025-09-01 19:56:51 |
| 6  | Client5   | 0           | fifth                           | 2025-08-13 19:57:14 |
| 7  | Client6   | 0           | sixth                           | 2025-08-30 19:57:32 |
| 8  | Client7   | 0           | seventh                         | 2025-09-11 20:40:59 |
| 9  | Client8   | 0           | eight                           | 2025-10-16 19:58:07 |
| 10 | Client9   | 0           | ninght                          | 2025-11-07 19:58:50 |
| 11 | Client10  | 0           | tenth                           | 2025-11-28 19:58:43 |

Рис. 2.5 Таблиця «Clients»

Запит на створення таблиці Clients виглядає так:

```

CREATE TABLE `clients` (
`Id` int(11) NOT NULL AUTO_INCREMENT,
`Name` varchar(256) COLLATE utf8_unicode_ci DEFAULT "",

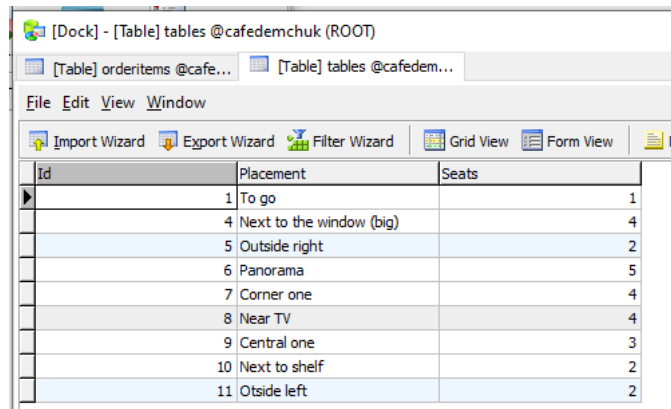
```

```

`HasDiscount` tinyint(4) NOT NULL DEFAULT 0,
`Comment` varchar(512) COLLATE utf8_unicode_ci DEFAULT "",
`AddedAt` datetime NOT NULL,
PRIMARY KEY (`Id`)
) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci

```

Для обліку місць у залі використовується таблиця Tables. У ній зберігається інформація про розташування столиків та кількість посадкових місць. Дані таблиці застосовуються під час оформлення замовлень і проведення аналітики використання залу.



| Id | Placement                | Seats |
|----|--------------------------|-------|
| 1  | To go                    | 1     |
| 4  | Next to the window (big) | 4     |
| 5  | Outside right            | 2     |
| 6  | Panorama                 | 5     |
| 7  | Corner one               | 4     |
| 8  | Near TV                  | 4     |
| 9  | Central one              | 3     |
| 10 | Next to shelf            | 2     |
| 11 | Outside left             | 2     |

Рис. 2.6 Таблиця «Tables»

Для того, щоб створити таблицю Tables SQL запит має такий вигляд:

```

CREATE TABLE `tables` (
`Id` int(11) NOT NULL AUTO_INCREMENT,
`Placement` varchar(256) COLLATE utf8_unicode_ci NOT NULL,
`Seats` int(11) NOT NULL,
PRIMARY KEY (`Id`)
) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci

```

Центральним елементом структури бази даних є таблиця Orders, яка містить інформацію про всі створені замовлення. Для кожного замовлення

зберігаються дані про робочу зміну, клієнта, столик, дату створення замовлення, його поточний стан та факт оплати. Така структура дозволяє відстежувати процес обслуговування клієнтів на всіх етапах.

| Id | workShiftId | clientId | tableId | created                | comment             | is_paid | status |
|----|-------------|----------|---------|------------------------|---------------------|---------|--------|
| 1  | 1           | 1        | 1       | 1 2025-12-06 10:52:42  | Order comment added | 0       | new    |
| 2  | 1           | 1        | 1       | 1 2025-12-07 21:39:12  |                     | 0       | new    |
| 3  | 5           | 1        | 1       | 1 2025-12-07 22:26:46  | Coffee and cake     | 0       | new    |
| 4  | 5           | 1        | 1       | 1 2025-12-07 22:31:33  | New                 | 0       | new    |
| 5  | 1           | 5        | 5       | 7 2025-12-06 09:12:00  | Morning coffee or   | 0       | new    |
| 6  | 1           | 11       | 4       | 4 2025-12-06 14:48:00  | Guest requested     | 0       | new    |
| 7  | 1           | 2        | 10      | 10 2025-12-06 19:33:00 | Birthday celebrati  | 0       | new    |
| 8  | 5           | 8        | 6       | 6 2025-12-07 09:05:00  | Warm-up breakfa     | 0       | new    |
| 9  | 5           | 2        | 11      | 11 2025-12-07 12:47:00 | Customer asked f    | 0       | new    |
| 10 | 5           | 10       | 4       | 4 2025-12-07 19:22:00  | Evening dinner ar   | 0       | new    |
| 11 | 5           | 1        | 1       | 1 2025-12-07 19:50:00  | Last for today      | 0       | new    |
| 12 | 6           | 1        | 4       | 4 2025-02-12 08:45:00  | Breakfast order     | 0       | new    |
| 13 | 6           | 2        | 7       | 7 2025-02-12 10:10:00  | Customer request    | 0       | new    |
| 14 | 6           | 1        | 9       | 9 2025-02-12 12:32:00  | Lunch menu sele     | 0       | new    |
| 15 | 6           | 2        | 11      | 11 2025-02-12 15:48:00 | Asked for gluten-f  | 0       | new    |
| 16 | 6           | 1        | 6       | 6 2025-02-12 17:05:00  | Coffee refill requ  | 0       | new    |

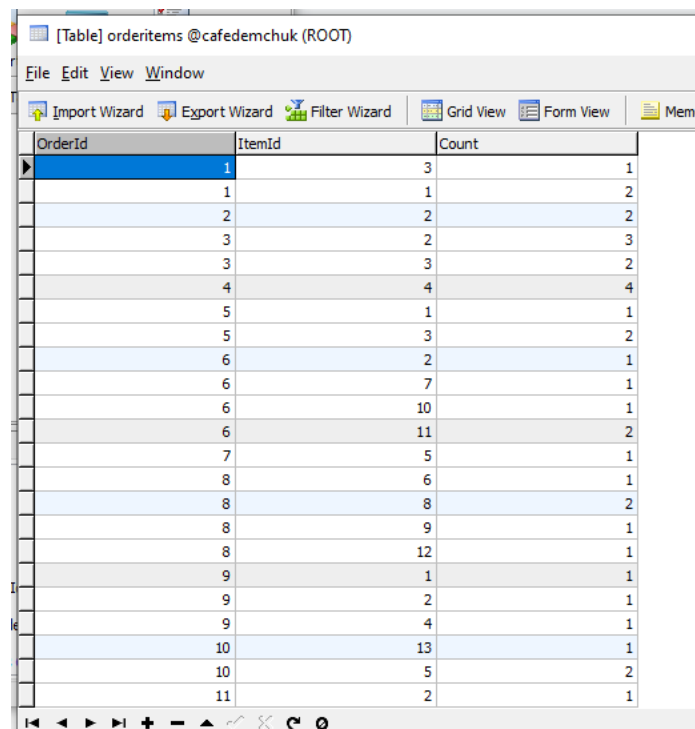
Рис. 2.7 Таблиця «Orders»

За допомогою наступного SQL запиту створено таблицю Orders:

```
CREATE TABLE `orders` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `workShiftId` int(11) NOT NULL,
  `clientId` int(11) NOT NULL,
  `tableId` int(11) NOT NULL,
  `created` datetime NOT NULL,
  `comment` varchar(512) COLLATE utf8_unicode_ci DEFAULT NULL,
  PRIMARY KEY (`Id`),
  KEY `client` (`clientId`),
  KEY `shift` (`workShiftId`),
  KEY `table` (`tableId`),
  CONSTRAINT `client` FOREIGN KEY (`clientId`) REFERENCES `clients` (`Id`) ON UPDATE CASCADE,
  CONSTRAINT `shift` FOREIGN KEY (`workShiftId`) REFERENCES `workshifts` (`Id`) ON UPDATE CASCADE,
  CONSTRAINT `table` FOREIGN KEY (`tableId`) REFERENCES `tables` (`Id`) ON UPDATE CASCADE
```

) ENGINE=InnoDB AUTO\_INCREMENT=93 DEFAULT CHARSET=utf8  
COLLATE=utf8\_unicode\_ci

Оскільки одне замовлення може включати декілька товарів, а один товар може бути присутнім у різних замовленнях, для реалізації такого зв'язку використовується проміжна таблиця OrderItems. Вона забезпечує зберігання інформації про склад кожного замовлення та кількість замовлених одиниць товару.



| OrderId | ItemId | Count |
|---------|--------|-------|
| 1       | 1      | 3     |
| 1       | 2      | 1     |
| 2       | 2      | 2     |
| 3       | 2      | 2     |
| 3       | 3      | 2     |
| 4       | 4      | 4     |
| 5       | 1      | 1     |
| 5       | 3      | 2     |
| 6       | 2      | 1     |
| 6       | 7      | 1     |
| 6       | 10     | 1     |
| 6       | 11     | 2     |
| 7       | 5      | 1     |
| 8       | 6      | 1     |
| 8       | 8      | 2     |
| 8       | 9      | 1     |
| 8       | 12     | 1     |
| 9       | 1      | 1     |
| 9       | 2      | 1     |
| 9       | 4      | 1     |
| 10      | 13     | 1     |
| 10      | 5      | 2     |
| 11      | 2      | 1     |

Рис. 2.8 Таблиця «OrderItems»

Створення SQL запиту для таблиця OrderItems:

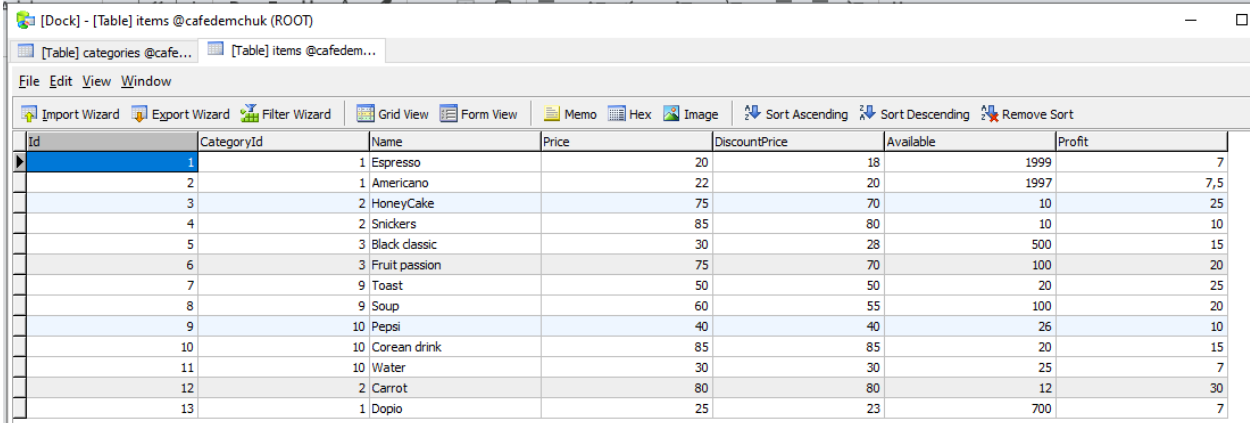
```
CREATE TABLE `orderitems` (  
  `OrderId` int(11) NOT NULL,  
  `ItemId` int(11) NOT NULL,  
  `Count` int(11) NOT NULL,  
  KEY `item` (`ItemId`),  
  KEY `order` (`OrderId`),  
  CONSTRAINT `item` FOREIGN KEY (`ItemId`) REFERENCES `items` (`Id`) ON  
  UPDATE CASCADE,
```

```

CONSTRAINT `order` FOREIGN KEY (`OrderId`) REFERENCES `orders` (`Id`)
ON UPDATE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci

```

Інформація про асортимент продукції кав'ярні зберігається в таблиці Items. Для кожної товарної позиції визначаються назва, категорія, вартість, ціна зі знижкою, доступна кількість та показник прибутковості. Це дозволяє здійснювати облік реалізації продукції та контролювати залишки товарів.



| Id | CategoryId | Name          | Price | DiscountPrice | Available | Profit |
|----|------------|---------------|-------|---------------|-----------|--------|
| 1  | 1          | Espresso      | 20    | 18            | 1999      | 7      |
| 2  | 1          | Americano     | 22    | 20            | 1997      | 7,5    |
| 3  | 2          | HoneyCake     | 75    | 70            | 10        | 25     |
| 4  | 2          | Snickers      | 85    | 80            | 10        | 10     |
| 5  | 3          | Black classic | 30    | 28            | 500       | 15     |
| 6  | 3          | Fruit passion | 75    | 70            | 100       | 20     |
| 7  | 9          | Toast         | 50    | 50            | 20        | 25     |
| 8  | 9          | Soup          | 60    | 55            | 100       | 20     |
| 9  | 10         | Pepsi         | 40    | 40            | 26        | 10     |
| 10 | 10         | Corean drink  | 85    | 85            | 20        | 15     |
| 11 | 10         | Water         | 30    | 30            | 25        | 7      |
| 12 | 2          | Carrot        | 80    | 80            | 12        | 30     |
| 13 | 1          | Dopio         | 25    | 23            | 700       | 7      |

Рис. 2.9 Таблиця «Items»

Розглянемо призначення таблиці Items і представимо SQL запит, яким вона створена:

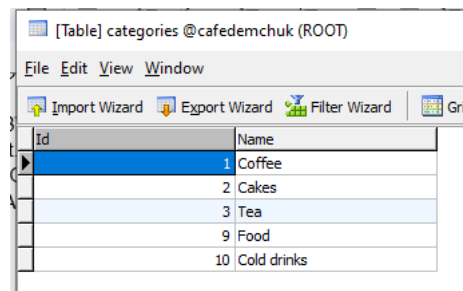
```

CREATE TABLE `items` (
  `Id` int(11) NOT NULL AUTO_INCREMENT,
  `CategoryId` int(11) NOT NULL,
  `Name` varchar(256) COLLATE utf8_unicode_ci NOT NULL,
  `Price` decimal(10,2) NOT NULL,
  `DiscountPrice` decimal(10,2) NOT NULL,
  `Available` int(11) NOT NULL,
  `Profit` decimal(10,2) NOT NULL,
  PRIMARY KEY (`Id`),
  KEY `category` (`CategoryId`),
  CONSTRAINT `category` FOREIGN KEY (`CategoryId`) REFERENCES
  `categories` (`Id`) ON UPDATE CASCADE

```

) ENGINE=InnoDB AUTO\_INCREMENT=14 DEFAULT CHARSET=utf8  
COLLATE=utf8\_unicode\_ci

Для систематизації асортименту використовується таблиця Categories, яка містить перелік категорій товарів. Поділ продукції на категорії спрощує пошук інформації та підвищує зручність роботи користувачів із системою.



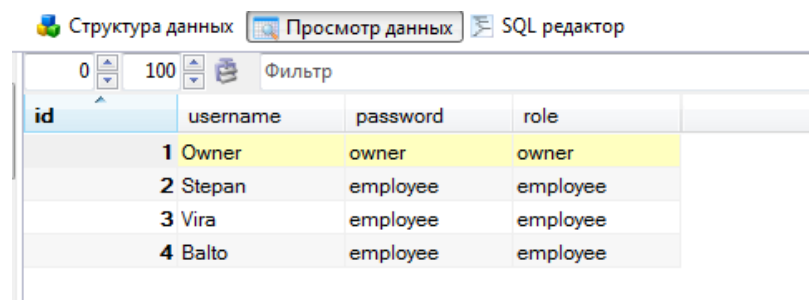
| Id | Name        |
|----|-------------|
| 1  | Coffee      |
| 2  | Cakes       |
| 3  | Tea         |
| 9  | Food        |
| 10 | Cold drinks |

Рис. 2.10 Таблица «Categories»

Categories має такий SQL запит:

```
CREATE TABLE `categories` (  
  `Id` int(11) NOT NULL AUTO_INCREMENT,  
  `Name` varchar(256) COLLATE utf8_unicode_ci NOT NULL,  
  PRIMARY KEY (`Id`)  
) ENGINE=InnoDB AUTO_INCREMENT=14 DEFAULT CHARSET=utf8  
COLLATE=utf8_unicode_ci
```

Таблиця Users забезпечує реалізацію механізму авторизації та розмежування доступу. У ній зберігаються облікові записи користувачів системи, їхні паролі та ролі. Завдяки цьому власник і працівники отримують доступ лише до тих функцій, які відповідають їхнім повноваженням.



| id | username | password | role     |
|----|----------|----------|----------|
| 1  | Owner    | owner    | owner    |
| 2  | Stepan   | employee | employee |
| 3  | Vira     | employee | employee |
| 4  | Balto    | employee | employee |

Рис. 2.11 Таблица Users

Код:

```
CREATE TABLE `users` (
```

```
`id` int(11) NOT NULL AUTO_INCREMENT,  
`username` varchar(50) NOT NULL,  
`password` varchar(50) NOT NULL,  
`role` varchar(20) NOT NULL,  
PRIMARY KEY (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_general_ci;
```

Крім спеціалізованих таблиць інформаційної системи кав'ярні, база даних містить службові таблиці WordPress (wp\_posts, wp\_users, wp\_options, wp\_comments, wp\_postmeta та інші), які забезпечують функціонування системи управління контентом, зберігання сторінок сайту, користувачів, налаштувань та медіафайлів.

## **2.2. Інформаційна система обліку**

Розроблена інформаційна система призначена для автоматизації основних процесів діяльності кав'ярні та забезпечує централізоване управління даними про працівників, клієнтів, товари, замовлення і робочі зміни. Система дозволяє спростити виконання щоденних операцій, підвищити швидкість обслуговування відвідувачів та покращити контроль за роботою закладу.

Під час створення системи особлива увага приділялася зручності використання, швидкому доступу до необхідної інформації та розмежуванню прав доступу між різними категоріями користувачів. Усі функціональні модулі об'єднані в єдине програмне середовище, що забезпечує взаємодію між ними та спільне використання даних.

### **Довідники в ІС**

Основу інформаційної системи складають довідники, які використовуються для накопичення та зберігання постійної інформації. Дані довідники формують інформаційну базу, необхідну для виконання щоденних операцій та створення звітності.

До основних довідників системи належать:

- працівники;

- клієнти;
- столики залу;
- категорії товарів;
- товари.

Довідник **працівників** містить інформацію про персонал закладу та використовується для організації роботи співробітників і контролю їхньої діяльності. У ньому зберігаються основні відомості про працівників, необхідні для формування робочих змін та аналітичних звітів.

Довідник **клієнтів** призначений для обліку постійних відвідувачів. Його використання дозволяє накопичувати інформацію про покупців, застосовувати персональні знижки та здійснювати аналіз клієнтської активності.

Довідник **столиків** забезпечує облік посадкових місць у залі закладу. Інформація про кількість місць та розташування столиків використовується під час оформлення замовлень і проведення аналізу ефективності використання приміщення.

Довідники **категорій товарів** та **товарів** формують структуру асортименту кав'ярні. Для кожного товару зберігаються відомості про його вартість, наявність на складі, можливі знижки та інші характеристики, необхідні для роботи персоналу.

## 2.1 Функціональні можливості системи

Після запуску програми користувач проходить процедуру авторизації шляхом введення облікових даних. На основі визначеної ролі система автоматично надає доступ до відповідного набору функцій та модулів.

У системі передбачено дві основні ролі користувачів: *власник* та *працівник*. *Власник* має повний доступ до всіх функціональних можливостей, включаючи налаштування системи, управління персоналом, перегляд фінансової інформації та формування звітів. *Працівник* має доступ лише до тих функцій, які необхідні для виконання його посадових обов'язків.

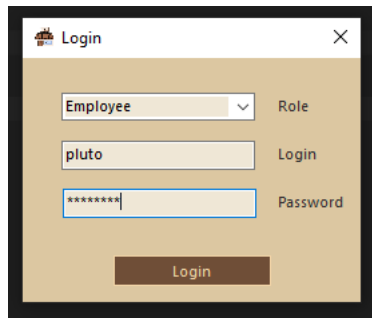


Рис.2.12 Форма авторизації користувача інформаційної системи

### Головне меню та рольовий доступ

Головне меню системи адаптується до ролі користувача. Такий підхід дозволяє приховати зайві функції та зробити роботу більш зручною і зрозумілою. Крім того, це підвищує рівень безпеки та захищає конфіденційну інформацію від несанкціонованого доступу. (рис. 2.13).

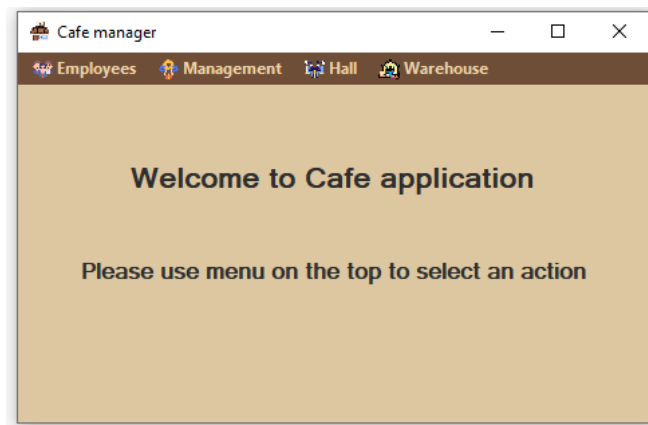


Рис. 2.13 Головне меню інформаційної системи для користувача з роллю власника.

Працівник не бачить пунктів меню, пов'язаних з управлінням персоналом та фінансами, але має кнопку швидкого створення замовлення для зменшення кількості дій під час роботи на зміні (рис. 2.14).

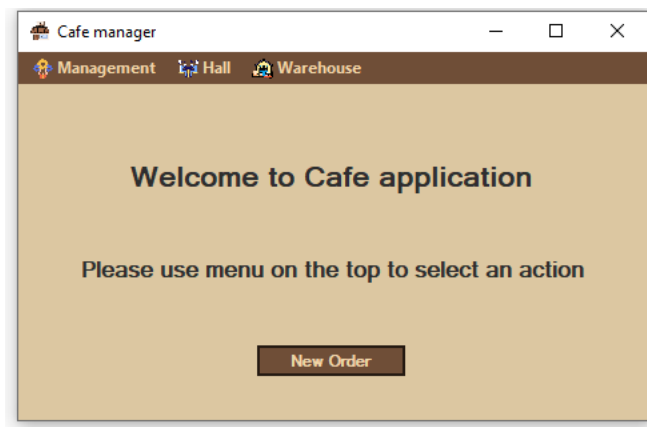


Рис. 2.14 Головне меню інформаційної системи для користувача з роллю працівника.

### Управління змінами (Management → Shifts)

У системі реалізовано механізм обліку робочих змін. Кожна зміна має статус *відкрита* або *закрита*. Працівник може мати лише одну відкриту зміну. Створення нових замовлень можливе виключно в межах відкритої зміни.

У модулі Shifts працівник бачить лише власні зміни, тоді як власник має доступ до змін усіх працівників (рис. 2.15)

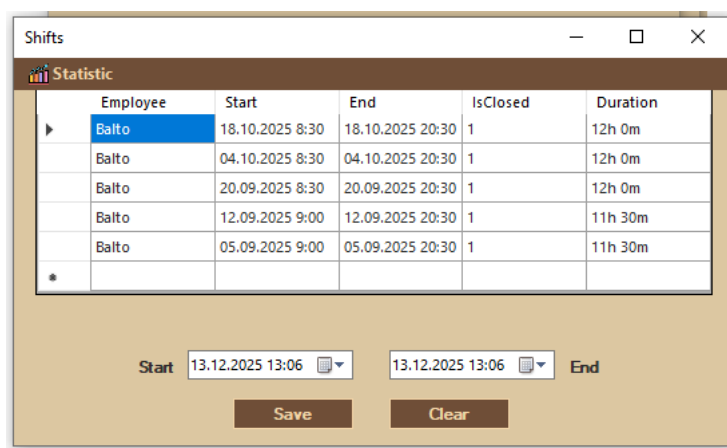


Рис. 2.15 Модуль управління робочими змінами працівника

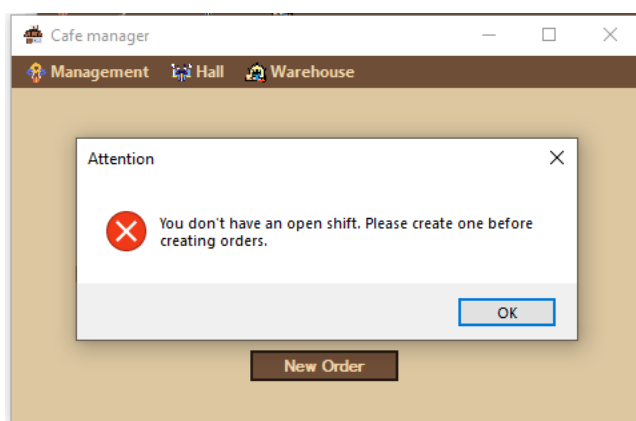
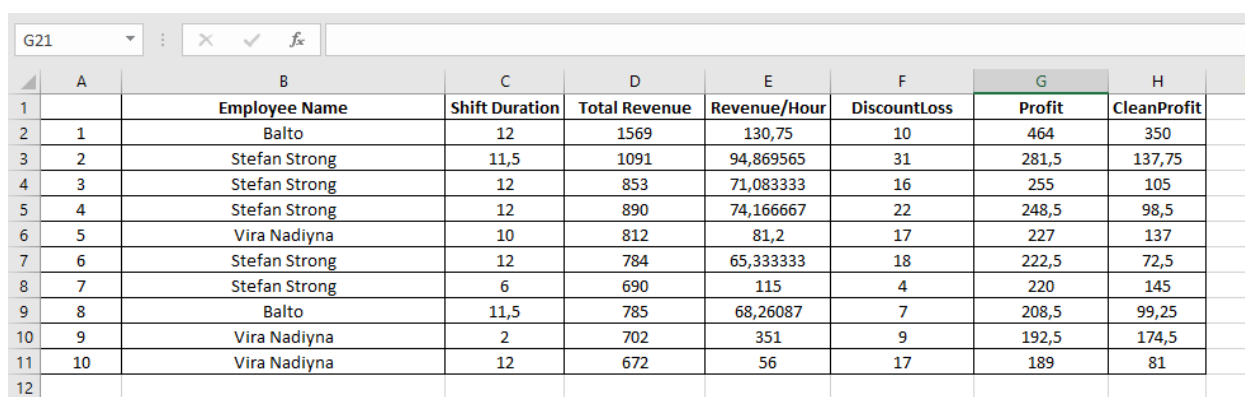


Рис. 2.16 Повідомлення про відсутність відкритої робочої зміни при спробі створення замовлення.

Закриття зміни здійснюється через контекстне меню. Після закриття зміни додавання нових замовлень стає неможливим, за винятком випадків, коли це виконує власник (рис. 2.16).

Система дозволяє вести історію змін та формувати аналітичні звіти щодо їх ефективності. На основі накопичених даних можна оцінювати продуктивність працівників, обсяг виконаних замовлень та інші показники діяльності. (рис. 2.17).



|    | A  | B             | C              | D             | E            | F            | G      | H           | I |
|----|----|---------------|----------------|---------------|--------------|--------------|--------|-------------|---|
|    |    | Employee Name | Shift Duration | Total Revenue | Revenue/Hour | DiscountLoss | Profit | CleanProfit |   |
| 1  |    | Balto         | 12             | 1569          | 130,75       | 10           | 464    | 350         |   |
| 2  | 1  | Stefan Strong | 11,5           | 1091          | 94,869565    | 31           | 281,5  | 137,75      |   |
| 3  | 2  | Stefan Strong | 12             | 853           | 71,083333    | 16           | 255    | 105         |   |
| 4  | 3  | Stefan Strong | 12             | 890           | 74,166667    | 22           | 248,5  | 98,5        |   |
| 5  | 4  | Vira Nadiyna  | 10             | 812           | 81,2         | 17           | 227    | 137         |   |
| 6  | 5  | Stefan Strong | 12             | 784           | 65,333333    | 18           | 222,5  | 72,5        |   |
| 7  | 6  | Stefan Strong | 6              | 690           | 115          | 4            | 220    | 145         |   |
| 8  | 7  | Balto         | 11,5           | 785           | 68,26087     | 7            | 208,5  | 99,25       |   |
| 9  | 8  | Vira Nadiyna  | 2              | 702           | 351          | 9            | 192,5  | 174,5       |   |
| 10 | 9  | Vira Nadiyna  | 12             | 672           | 56           | 17           | 189    | 81          |   |
| 11 | 10 |               |                |               |              |              |        |             |   |
| 12 |    |               |                |               |              |              |        |             |   |

Рис. 2.17 Звіт ТОП-10 робочих змін у форматі Excel

### Створення та управління замовленнями (Main Menu → New Order, Management → Orders)

Важливою складовою системи є модуль роботи із замовленнями(рис. 2.18). Процес створення замовлення складається з декількох етапів. Спочатку користувач вказує клієнта, обирає столик та додає необхідні коментарі. Після цього здійснюється вибір товарів із наявного асортименту.

The screenshot shows the 'View Orders' window with a 'Statistic' tab. The 'Employee' dropdown is set to 'Balto'. The date range is from '01.01.2025 00:01' to '31.12.2030 12:00'. The main table lists orders with columns: Employee, Client, ShiftStart, OrderedAt, Table, OrderTotal, and comment. Below this is a table of items with columns: Item, Count, PricePerItem, and TotalPrice.

| Employee | Client  | ShiftStart      | OrderedAt        | Table               | OrderTotal | comment             |
|----------|---------|-----------------|------------------|---------------------|------------|---------------------|
| Balto    | Client9 | 18.10.2025 8:30 | 18.10.2025 20:20 | Next to the wind... | 250,00     | Evening snack       |
| Balto    | Client8 | 18.10.2025 8:30 | 18.10.2025 19:05 | Corner one          | 760,00     | Dinner order        |
| Balto    | Client7 | 18.10.2025 8:30 | 18.10.2025 17:50 | Outside right       | 195,00     | Early evening or... |
| Balto    | Client5 | 18.10.2025 8:30 | 18.10.2025 16:30 | Otside left         | 185,00     | Late afternoon o... |
| Balto    | Client3 | 18.10.2025 8:30 | 18.10.2025 15:10 | Next to shelf       | 122,00     | Snack order         |
| Balto    | Client6 | 18.10.2025 8:30 | 18.10.2025 13:45 | Central one         | 175,00     | Afternoon coffee    |

| Item          | Count | PricePerItem | TotalPrice |
|---------------|-------|--------------|------------|
| Black classic | 1     | 30,00        | 30,00      |
| Fruit passion | 1     | 75,00        | 75,00      |
| Soup          | 1     | 60,00        | 60,00      |
| Water         | 1     | 30,00        | 30,00      |

Рис. 2.18 Форма створення нового замовлення

Під час додавання товарів система автоматично перевіряє наявність необхідної кількості продукції на складі. Якщо доступних залишків недостатньо, користувач отримує відповідне повідомлення про помилку. (рис. 2.19).

The screenshot shows the 'View Orders' window with a 'Statistic' tab. The 'Employee' dropdown is set to 'Vira Nadiyna'. The date range is from '01.01.2025 00:01' to '31.12.2030 12:00'. The main table lists orders with columns: Employee, Client, ShiftStart, OrderedAt, Table, OrderTotal, and comment. Below this is a table of items with columns: Item, Count, PricePerItem, and TotalPrice. A red arrow points to the 'Pepsi' item in the table. Another red arrow points to the 'Pepsi' dropdown in the 'Item' field. An 'Attention' dialog box is displayed in the foreground with a red 'X' icon and the text: 'Action failed. You can not add xisting item. Please remove it first'. The dialog box has an 'OK' button.

| Employee     | Client    | ShiftStart       | OrderedAt        | Table               | OrderTotal | comment            |
|--------------|-----------|------------------|------------------|---------------------|------------|--------------------|
| Vira Nadiyna | NewClient | 07.12.2025 21:45 | 07.12.2025 22:31 | To go               | 340,00     | New                |
| Vira Nadiyna | NewClient | 07.12.2025 21:45 | 07.12.2025 22:26 | To go               | 216,00     | Coffe and cake     |
| Vira Nadiyna | NewClient | 06.12.2025 8:30  | 07.12.2025 21:39 | To go               | 44,00      | added              |
| Vira Nadiyna | NewClient | 07.12.2025 21:45 | 07.12.2025 19:50 | To go               | 122,00     | Last for today     |
| Vira Nadiyna | Client9   | 07.12.2025 21:45 | 07.12.2025 19:22 | Next to the wind... | 85,00      | Evening dinner ... |
| Vira Nadiyna | Client    | 07.12.2025 21:45 | 07.12.2025 12:47 | Otside left         | 118,00     | Customer asked ... |

| Item          | Count | PricePerItem | TotalPrice |
|---------------|-------|--------------|------------|
| Americano     | 1     | 22,00        | 22,00      |
| Black classic | 2     | 30,00        | 60,00      |
| Pepsi         | 1     | 40,00        | 40,00      |

Рис. 2.19 Повідомлення про недостатню кількість товару на складі

Після підтвердження замовлення інформація зберігається в базі даних, а залишки товарів автоматично коригуються відповідно до кількості реалізованої продукції. Це дозволяє підтримувати актуальний стан складських запасів без необхідності ручного обліку.

Для власника передбачено розширені можливості роботи із замовленнями. Він може переглядати історію продажів, редагувати замовлення, здійснювати пошук за різними критеріями та формувати друковані документи.

Система також забезпечує автоматичне створення чеків та інших документів, необхідних для супроводу процесу обслуговування клієнтів (рис. 2.20).

Завдяки інтеграції всіх функціональних модулів інформаційна система забезпечує комплексну автоматизацію діяльності кав'ярні, скорочує час виконання операцій та підвищує ефективність управління закладом.

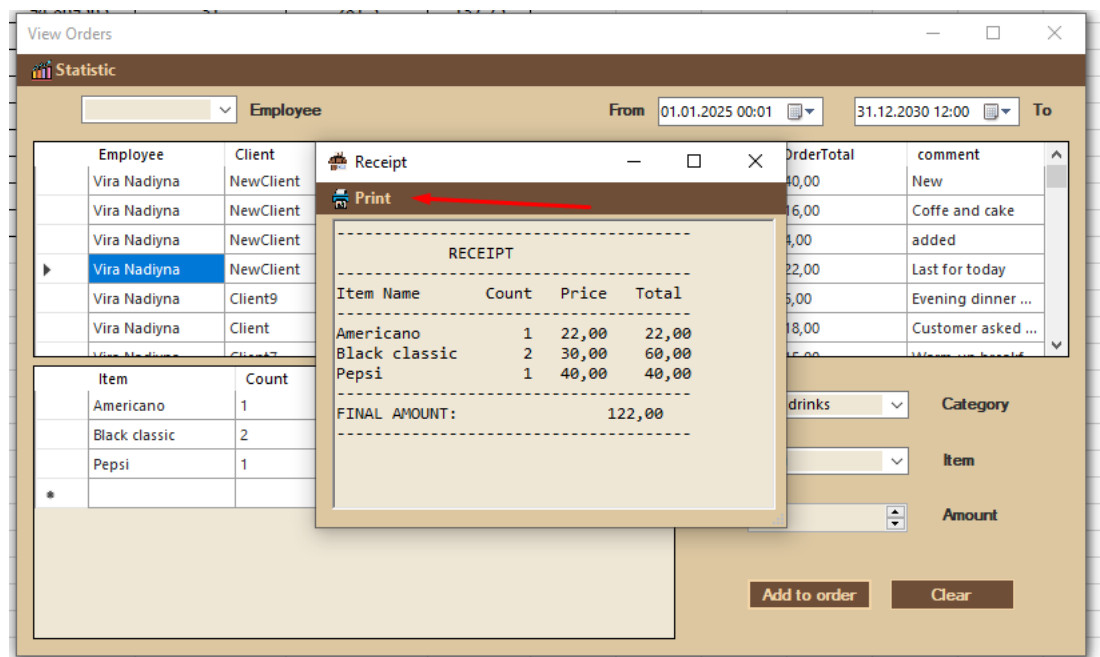


Рис. 2.20 Згенерований чек замовлення у форматі Word

### Управління працівниками, залом та складом

Для забезпечення ефективного функціонування кав'ярні в інформаційній системі реалізовано окремі модулі, призначені для управління персоналом, обслуговуванням клієнтів у залі та контролем товарних запасів. Використання спеціалізованих модулів дозволяє централізовано працювати з усіма

необхідними даними та оперативно виконувати адміністративні завдання.

### ***Модуль управління працівниками (Employees)***

Модуль **Employees** використовується для ведення обліку персоналу закладу. Він надає можливість переглядати інформацію про працівників, додавати нові записи, вносити зміни до наявних даних та видаляти співробітників, якщо це не порушує цілісність бази даних.

| Login  | HourlyRate | DateOfEmployment |
|--------|------------|------------------|
| stepan | 12,50      | 01.01.2025       |
| vira   | 9,00       | 18.03.2025       |
| balto  | 9,50       | 03.09.2025       |

Form fields:

- Name:
- Login:
- Salary rate:
- Employment date:

Buttons:

Рис. 2.21 Модуль управління працівниками

На головній формі модуля відображається список усіх працівників із зазначенням основних характеристик. Для редагування інформації достатньо обрати необхідний запис та перейти до режиму зміни даних.

З метою забезпечення цілісності інформації система контролює наявність пов'язаних записів. Якщо працівник має зареєстровані робочі зміни або інші залежні дані, операція видалення блокується автоматично. (рис. 2.22).

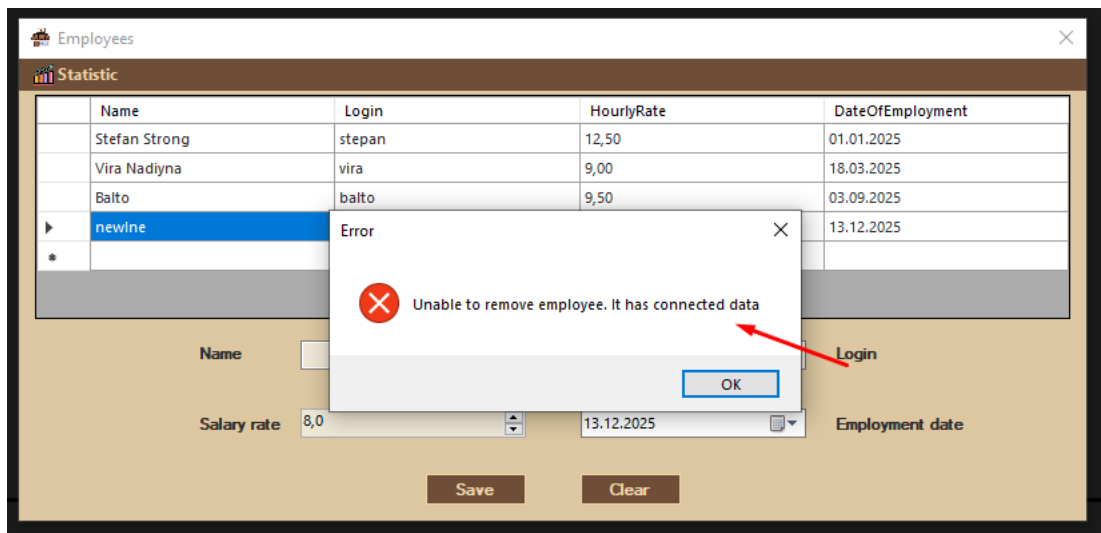


Рис. 2.22 Повідомлення про неможливість видалення працівника з пов'язаними даними

Крім виконання базових операцій, модуль підтримує формування звітів щодо діяльності персоналу. Користувач може вказати необхідний часовий інтервал, після чого система автоматично створює звіт із відповідними статистичними показниками(рис. 2.23).

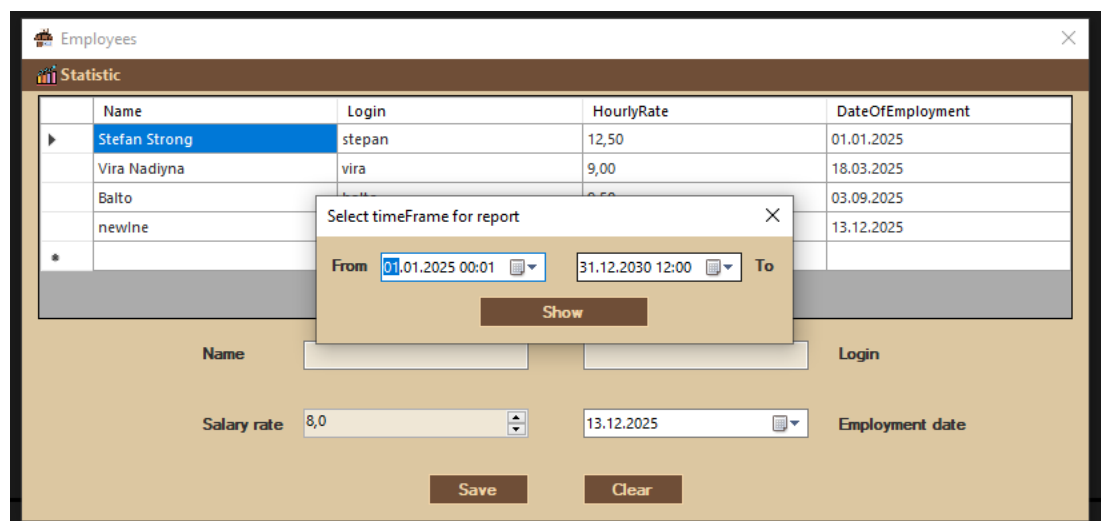


Рис. 2.23 Форма вибору періоду формування звіту по працівниках

### *Управління залом (Tables та Clients)*

Для обліку посадкових місць використовується модуль **Tables**. Його призначення полягає у зберіганні інформації про розташування столиків та кількість місць для відвідувачів. Наявність таких даних дозволяє

контролювати використання залу та проводити аналіз ефективності роботи закладу.

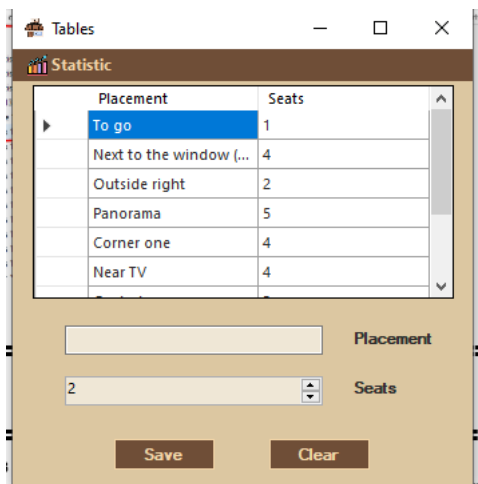


Рис. 2.24 Модуль управління столиками залу

Користувач із відповідними правами може створювати нові записи, змінювати характеристики існуючих столиків або видаляти непотрібні дані. Крім того, система підтримує формування аналітичних звітів щодо кількості оформлених замовлень для кожного столика та отриманого доходу. (рис. 2.25).

|    | A  | B                        | C      | D               | E     | F      | G               |
|----|----|--------------------------|--------|-----------------|-------|--------|-----------------|
| 1  |    | Table                    | Amount | Amount per Seat | Seats | Profit | Profit per Seat |
| 2  | 1  | Next to the window (big) | 2790   | 697,5           | 4     | 905,5  | 226,375         |
| 3  | 2  | Corner one               | 3746   | 936,5           | 4     | 844    | 211             |
| 4  | 3  | Next to shelf            | 2690   | 1345            | 2     | 831,5  | 415,75          |
| 5  | 4  | Panorama                 | 2140   | 428             | 5     | 699    | 139,8           |
| 6  | 5  | Outside left             | 2482   | 1241            | 2     | 613,5  | 306,75          |
| 7  | 6  | To go                    | 2072   | 2072            | 1     | 589    | 589             |
| 8  | 7  | Central one              | 1315   | 438,333333      | 3     | 330    | 110             |
| 9  | 8  | Outside right            | 510    | 255             | 2     | 178    | 89              |
| 10 | 9  | Near TV                  | 447    | 111,75          | 4     | 134,5  | 33,625          |
| 11 | 10 |                          |        |                 |       |        |                 |
| 12 |    |                          |        |                 |       |        |                 |

Рис. 2.25 Звіт по ефективності використання столиків

Для роботи з клієнтами передбачено модуль **Clients**. Його використання дозволяє накопичувати інформацію про відвідувачів, вести облік постійних клієнтів та застосовувати індивідуальні умови обслуговування. (рис. 2.26).

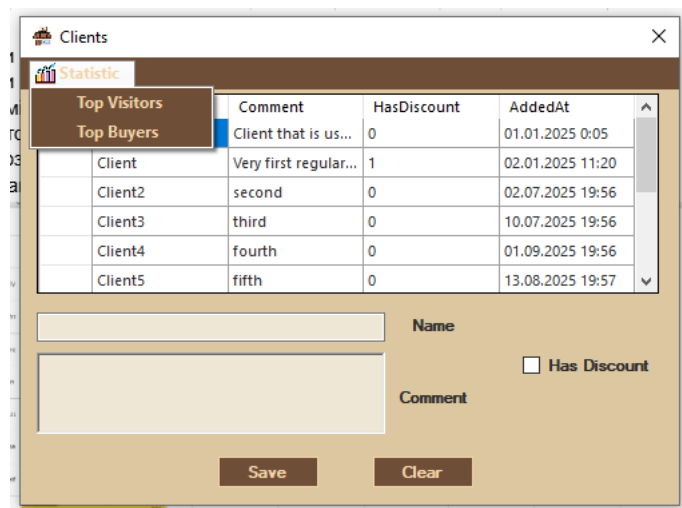


Рис. 2.26 Модуль управління клієнтами

Однією з переваг модуля є можливість формування аналітичних звітів, які допомагають визначати найактивніших відвідувачів та клієнтів, що приносять найбільший прибуток закладу. (рис. 2.17 а, б).

|    | A  | B         | C      |
|----|----|-----------|--------|
| 1  | №  | Name      | Visits |
| 2  | 1  | NewClient | 38     |
| 3  | 2  | Client    | 30     |
| 4  | 3  | Client4   | 5      |
| 5  | 4  | Client2   | 4      |
| 6  | 5  | Client7   | 4      |
| 7  | 6  | Client3   | 3      |
| 8  | 7  | Client5   | 2      |
| 9  | 8  | Client6   | 2      |
| 10 | 9  | Client9   | 2      |
| 11 | 10 | Client8   | 1      |
| 12 |    |           |        |

Рис. 2.27 (а) Звіт Top Visitors

|    | A  | B         | C         | D                |
|----|----|-----------|-----------|------------------|
| 1  |    | Customer  | Spendings | Generated Profit |
| 2  | 1  | NewClient | 6829      | 2113,5           |
| 3  | 2  | Client    | 6287      | 1581,5           |
| 4  | 3  | Client7   | 920       | 264              |
| 5  | 4  | Client4   | 717       | 234,5            |
| 6  | 5  | Client2   | 736       | 210,5            |
| 7  | 6  | Client3   | 486       | 179,5            |
| 8  | 7  | Client9   | 335       | 134              |
| 9  | 8  | Client6   | 585       | 129              |
| 10 | 9  | Client8   | 760       | 120              |
| 11 | 10 | Client5   | 320       | 97               |
| 12 |    |           |           |                  |

Рис. 2.27 (б) Звіт Top Buyers

Працівники мають доступ лише до перегляду загальної інформації про

клієнтів та окремих статистичних показників. Детальна фінансова аналітика доступна виключно власнику системи.

### ***Управління складом (Categories, Items та Check Stock)***

Контроль товарного асортименту здійснюється за допомогою модулів **Categories**, **Items** та **Check Stock**. Їх використання дозволяє вести облік продукції, контролювати залишки та аналізувати результати реалізації товарів. (рис. 2.28).

Модуль **Categories** призначений для роботи з категоріями товарів. Він забезпечує структурування асортименту та спрощує процес пошуку необхідної продукції.

Власник може створювати нові категорії, редагувати їх характеристики або видаляти непотрібні записи. При цьому система контролює наявність товарів у категорії та не допускає видалення категорій, які використовуються іншими об'єктами бази даних.

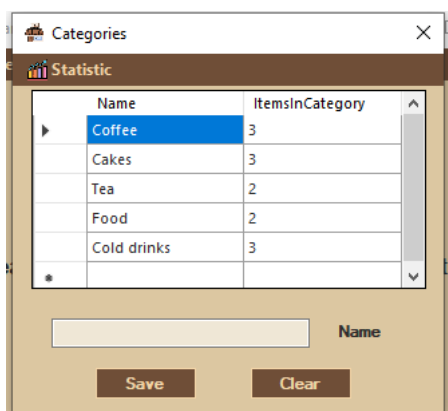


Рис. 2.28 Модуль управління категоріями товарів

Модуль **Items** використовується для обліку товарів, представлених у меню кав'ярні. Для кожної позиції зберігаються відомості про назву, вартість, розмір знижки, залишок на складі та інші характеристики. (рис. 2.29).

| Name          | Category | Price | DiscountPrice | Available | Profit |
|---------------|----------|-------|---------------|-----------|--------|
| Espresso      | Coffee   | 20,00 | 18,00         | 1999      | 7,00   |
| Americano     | Coffee   | 22,00 | 20,00         | 1997      | 7,50   |
| HoneyCake     | Cakes    | 75,00 | 70,00         | 10        | 25,00  |
| Snickers      | Cakes    | 85,00 | 80,00         | 2         | 10,00  |
| Black classic | Tea      | 30,00 | 28,00         | 500       | 15,00  |
| Fruit passion | Tea      | 75,00 | 70,00         | 100       | 20,00  |

Рис. 2.29 Модуль управління товарами

На основі накопичених даних система формує аналітичні звіти щодо реалізації продукції. У звітах відображається кількість проданих товарів, сума продажів та інші показники, необхідні для аналізу діяльності закладу. (рис. 2.30).

|    | A             | B          | C       | D      |
|----|---------------|------------|---------|--------|
| 1  | Item          | Items Sold | Revenue | Profit |
| 2  | Americano     | 26         | 690     | 226,00 |
| 3  | Espresso      | 26         | 644     | 215,00 |
| 4  | Black classic | 25         | 1000    | 490,00 |
| 5  | HoneyCake     | 24         | 2400    | 750,00 |
| 6  | Fruit passion | 23         | 2110    | 515,00 |
| 7  | Snickers      | 22         | 2670    | 270,00 |
| 8  | Carrot        | 20         | 2000    | 750,00 |
| 9  | Toast         | 18         | 1250    | 625,00 |
| 10 | Pepsi         | 18         | 960     | 240,00 |
| 11 | Corean drink  | 18         | 2040    | 360,00 |
| 12 | Soup          | 17         | 1235    | 395,00 |
| 13 | Water         | 16         | 780     | 182,00 |
| 14 | Dopio         | 14         | 413     | 107,00 |
| 15 |               |            |         |        |

Рис. 2.30 Звіт по реалізації товарів

Для контролю складських залишків використовується модуль **Check Stock**. Його основним завданням є автоматичний аналіз кількості товарів та своєчасне виявлення позицій, запас яких наближається до критичного рівня. (рис. 2.31).

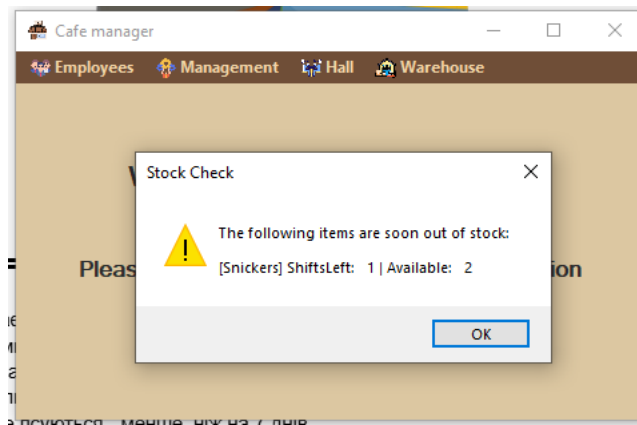


Рис. 2.31 Модуль контролю залишків товарів

Завдяки використанню даного модуля працівники та власник можуть оперативно реагувати на нестачу продукції та планувати своєчасне поповнення запасів. Це дозволяє уникнути ситуацій, коли окремі товари стають недоступними для продажу через відсутність на складі.

### **Аналіз вихідної інформації**

Вихідна інформація інформаційної системи кав'ярні формується на основі даних, що зберігаються у базі даних, та подається користувачу у зручному для аналізу вигляді. Основними формами представлення результатів є таблиці, графічні діаграми та автоматично сформовані звіти, які можуть експортуватися у формат Excel або інші офісні формати.

Генерація звітів у системі реалізована на основі попередньо визначених шаблонів, що забезпечує однакову структуру вихідних документів та спрощує подальшу аналітичну обробку даних. Такий підхід дозволяє уникнути дублювання логіки формування звітів у різних модулях системи.

Система підтримує широкий спектр аналітичних звітів, що охоплюють різні аспекти діяльності кав'ярні, зокрема роботу персоналу, ефективність змін, показники продажів та поведінку клієнтів.

#### *Звіти по працівниках*

Одним із важливих типів аналітики є звіти щодо діяльності працівників. Вони дозволяють оцінити продуктивність персоналу за певний період часу та

визначити внесок кожного співробітника у загальний результат роботи закладу (рис. 2.32).

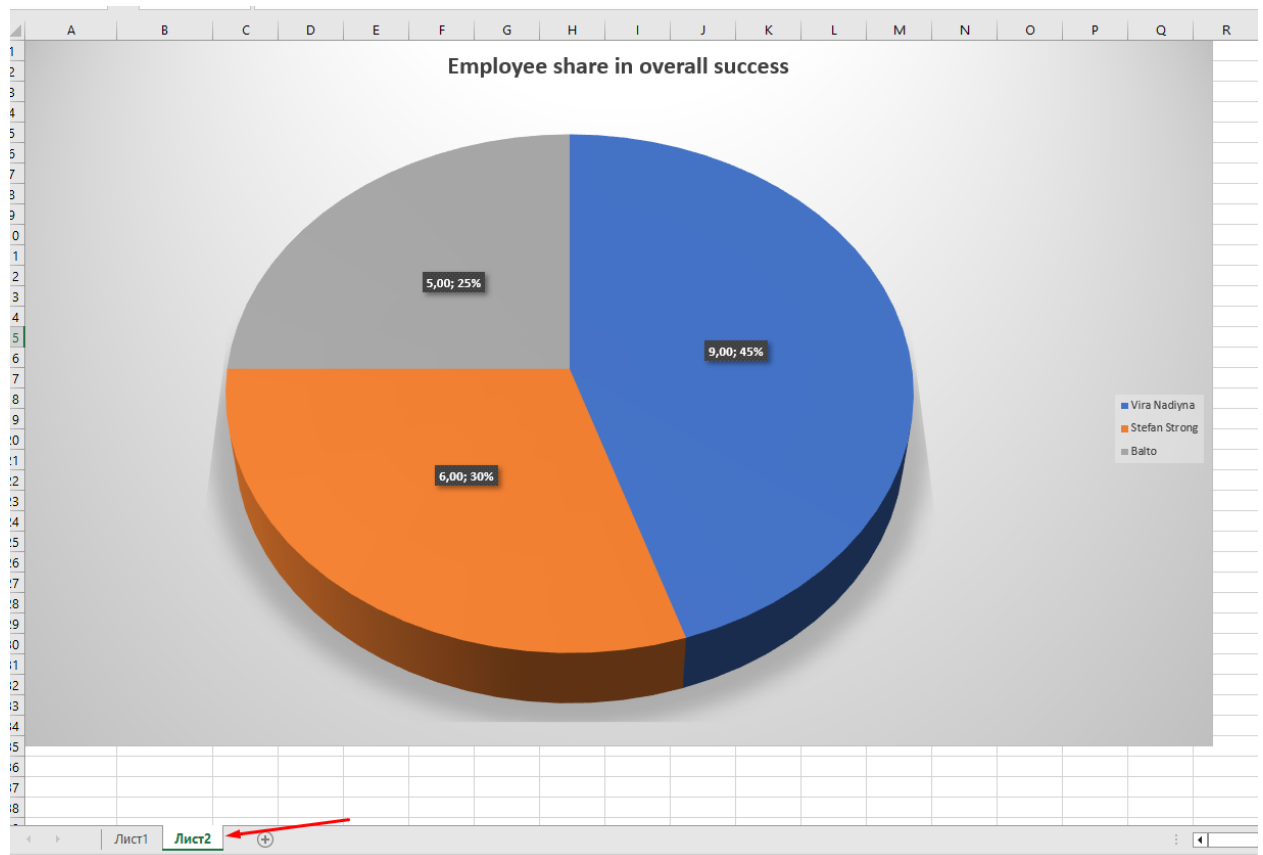
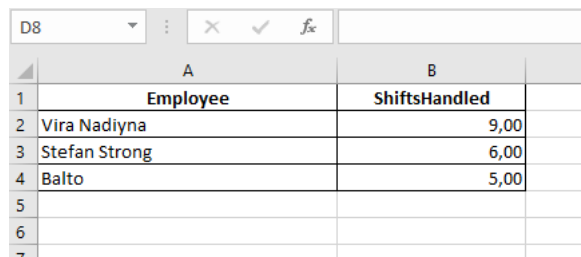


Рис. 2.32 Діаграма розподілу ефективності працівників

Такі звіти включають показники кількості відпрацьованих змін, обсягу виконаних замовлень, сум реалізації та отриманого доходу. Для наочності результати можуть бути представлені у вигляді графіків та діаграм, що спрощує їх інтерпретацію.

#### *Звіти по робочих змінах*

Окремий блок аналітики стосується робочих змін персоналу. У цих звітах відображається інформація про тривалість змін, кількість обслугованих замовлень, середню продуктивність за годину та фінансові результати кожної зміни.



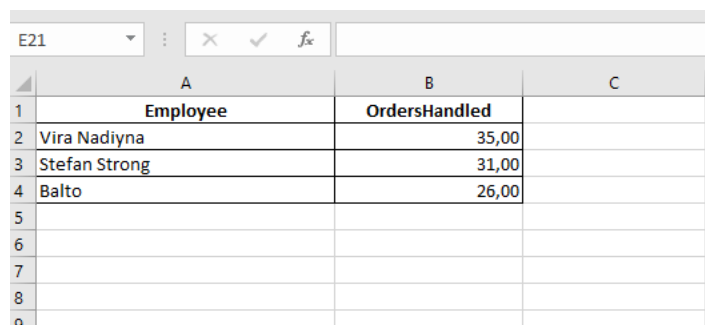
|   | A             | B             |
|---|---------------|---------------|
| 1 | Employee      | ShiftsHandled |
| 2 | Vira Nadiyna  | 9,00          |
| 3 | Stefan Strong | 6,00          |
| 4 | Balto         | 5,00          |
| 5 |               |               |
| 6 |               |               |

Рис. 2.33 Аналітичний звіт по робочих змінах

Отримані дані дозволяють порівнювати ефективність різних змін та виявляти найбільш результативні періоди роботи закладу. Крім того, така інформація використовується для оптимізації графіків роботи персоналу.

#### *Звіти по замовленнях*

Звіти щодо замовлень забезпечують аналіз динаміки продажів та структури реалізованої продукції. У табличному вигляді відображаються основні параметри кожного замовлення, включаючи дату створення, працівника, клієнта, столик та загальну суму. (рис. 2.34).



|   | A             | B             | C |
|---|---------------|---------------|---|
| 1 | Employee      | OrdersHandled |   |
| 2 | Vira Nadiyna  | 35,00         |   |
| 3 | Stefan Strong | 31,00         |   |
| 4 | Balto         | 26,00         |   |
| 5 |               |               |   |
| 6 |               |               |   |
| 7 |               |               |   |
| 8 |               |               |   |
| 9 |               |               |   |

Рис. 2.34 Табличний звіт по замовленнях

Для більш глибокого аналізу система також формує графічні представлення даних, що дозволяє відстежувати тенденції зміни кількості замовлень у часі та оцінювати стабільність роботи закладу. (рис. 2.35).

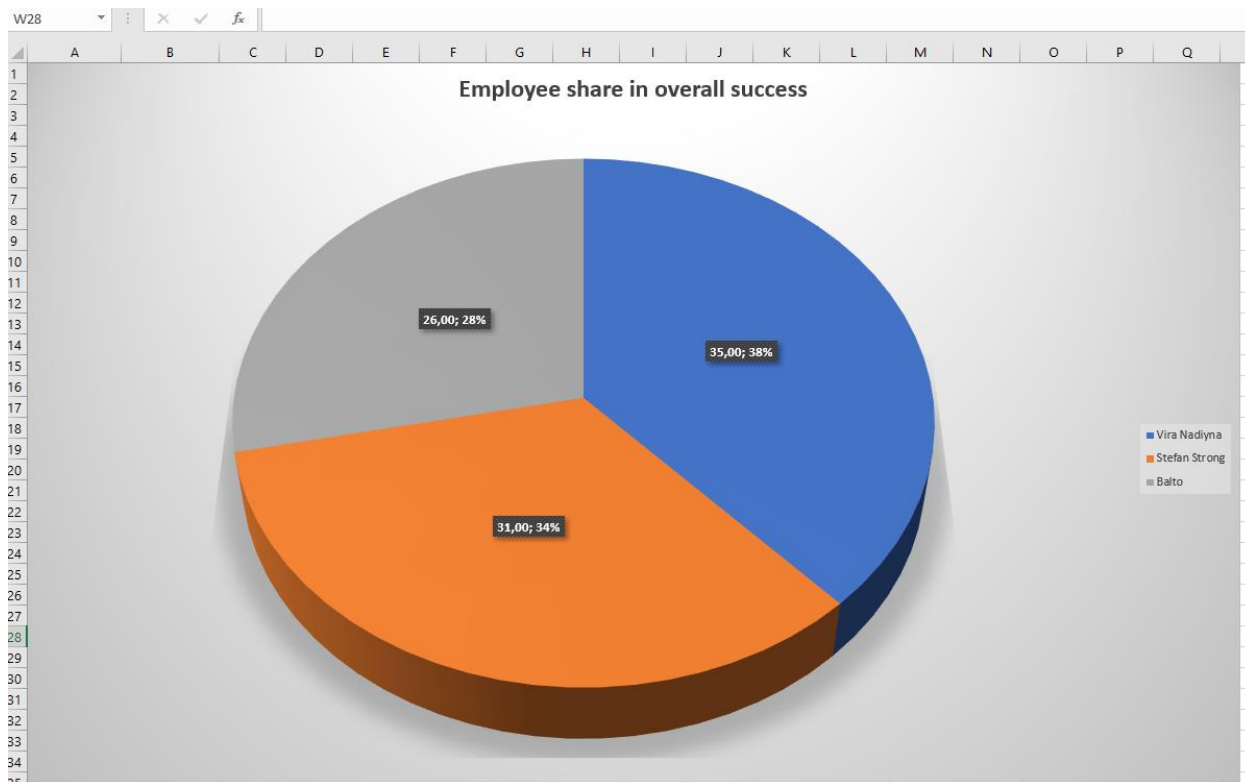


Рис. 2.35 Діаграма динаміки замовлень у часі

### *Рольовий доступ до вихідної інформації*

Важливою особливістю системи є реалізація рольової моделі доступу до аналітичної інформації. Залежно від ролі користувача обсяг доступних даних може відрізнятися.

Працівники мають доступ лише до базових показників, таких як кількість замовлень та загальні суми продажів. Власник системи, у свою чергу, отримує повний доступ до фінансової аналітики, включаючи показники доходу та прибутковості.

Такий підхід забезпечує захист конфіденційної інформації та відповідає принципам розмежування прав доступу.

### **Інтерфейс користувача**

Інтерфейс інформаційної системи розроблено з урахуванням принципів простоти, логічності та зручності використання. Основний акцент зроблено на мінімізації кількості дій користувача під час виконання типових операцій.

У системі реалізовано структуроване меню навігації, контекстні меню для

швидкого доступу до функцій, а також динамічне відображення елементів інтерфейсу відповідно до ролі користувача.

Крім того, система передбачає механізм повідомлень про помилки та результати виконання операцій, що дозволяє користувачу оперативно реагувати на зміну стану системи та уникати некоректних дій.

. Завдяки такій організації інтерфейсу підвищується ефективність роботи користувачів та зменшується ймовірність виникнення помилок під час експлуатації системи.

### **Висновки до другого розділу**

У даному розділі було описано принципи реалізації інформаційної системи кав'ярні, її основні функціональні можливості, структуру довідників, механізми обробки даних та формування вихідної інформації. Особливу увагу приділено рольовій моделі доступу, яка забезпечує розмежування прав між різними категоріями користувачів та підвищує рівень безпеки системи.

Розроблена система дозволяє автоматизувати ключові процеси діяльності кав'ярні, підвищити ефективність управління та забезпечити оперативний доступ до необхідної інформації.

### **2.3. Функціональні можливості вебсайту кав'ярні**

Для розробки та тестування вебсайту кав'ярні «CoffeeShop» було використано сучасні програмні засоби, які забезпечують зручність створення, налаштування та супроводу вебресурсу.

Основною платформою для створення сайту було обрано систему керування контентом WordPress. WordPress є однією з найпопулярніших CMS у світі, яка дозволяє створювати та адмініструвати вебсайти без необхідності розробки складних програмних рішень з нуля. Перевагами WordPress є простий інтерфейс керування, велика кількість готових тем оформлення, підтримка плагінів та можливість розширення функціоналу за допомогою власних програмних модулів. Використання WordPress дозволило реалізувати

сторінки сайту, систему публікації новин та забезпечити зручне адміністрування контенту.

Для організації локального серверного середовища використовувався програмний пакет XAMPP. Дане програмне забезпечення містить вебсервер Apache, систему управління базами даних MySQL (MariaDB), а також додаткові інструменти для розробки та тестування вебзастосунків. Використання XAMPP дозволило розгорнути локальний сервер на персональному комп'ютері та виконувати тестування функціональних можливостей сайту без необхідності використання віддаленого хостингу.

Для створення та адміністрування бази даних використовувалася система управління базами даних MySQL. Вона забезпечує надійне зберігання інформації про товари, категорії продукції, замовлення, клієнтів, працівників та інші дані інформаційної системи. Завдяки використанню реляційної моделі даних забезпечується цілісність інформації та підтримка зв'язків між таблицями бази даних.

Для керування базою даних використовувався вебінтерфейс phpMyAdmin, який входить до складу XAMPP. За допомогою phpMyAdmin виконувалося створення таблиць, налаштування зв'язків між ними, внесення тестових даних та контроль роботи інформаційної системи.

Розробка програмних модулів та налаштування функціональних можливостей сайту здійснювалися з використанням редактора програмного коду Visual Studio Code. Дане середовище забезпечує підтримку мов програмування PHP, HTML, CSS та JavaScript, що використовувалися під час створення окремих елементів вебсайту.

У рамках виконання бакалаврської роботи було створено вебсайт кав'ярні «CoffeeShop», який є складовою комплексної інформаційної системи для підвищення ефективності роботи закладу. Вебсайт розроблено з використанням системи керування контентом WordPress та бази даних MySQL. Такий підхід забезпечує зручне адміністрування ресурсу, можливість швидкого оновлення інформації та інтеграцію з інформаційною системою

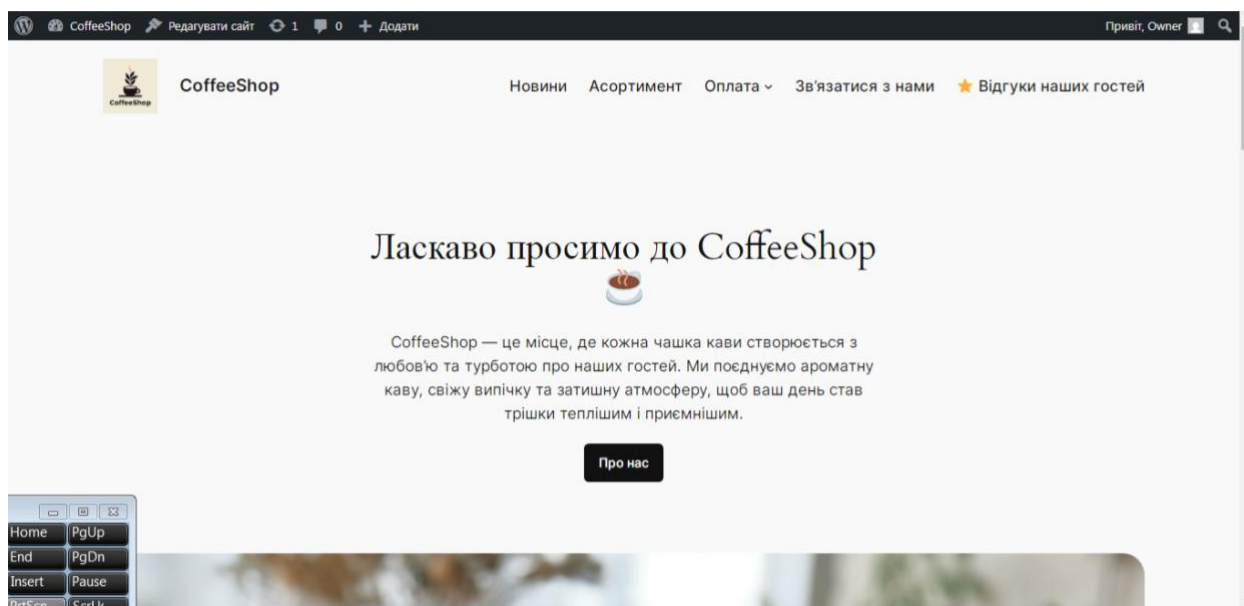
обліку кав'ярні.

Основною метою створення вебсайту є забезпечення інформаційної підтримки діяльності кав'ярні, автоматизація процесу оформлення замовлень та підвищення якості обслуговування клієнтів.

Структура вебсайту включає такі основні сторінки:

- головна сторінка;
- сторінка новин;
- сторінка оформлення замовлення;
- сторінка оплати;

Головна сторінка є основним елементом навігації сайту та містить інформацію про кав'ярню, її асортимент, переваги та актуальні пропозиції. Для створення позитивного враження використано сучасний дизайн, тематичні зображення та адаптивне оформлення, що забезпечує коректне відображення на різних пристроях.



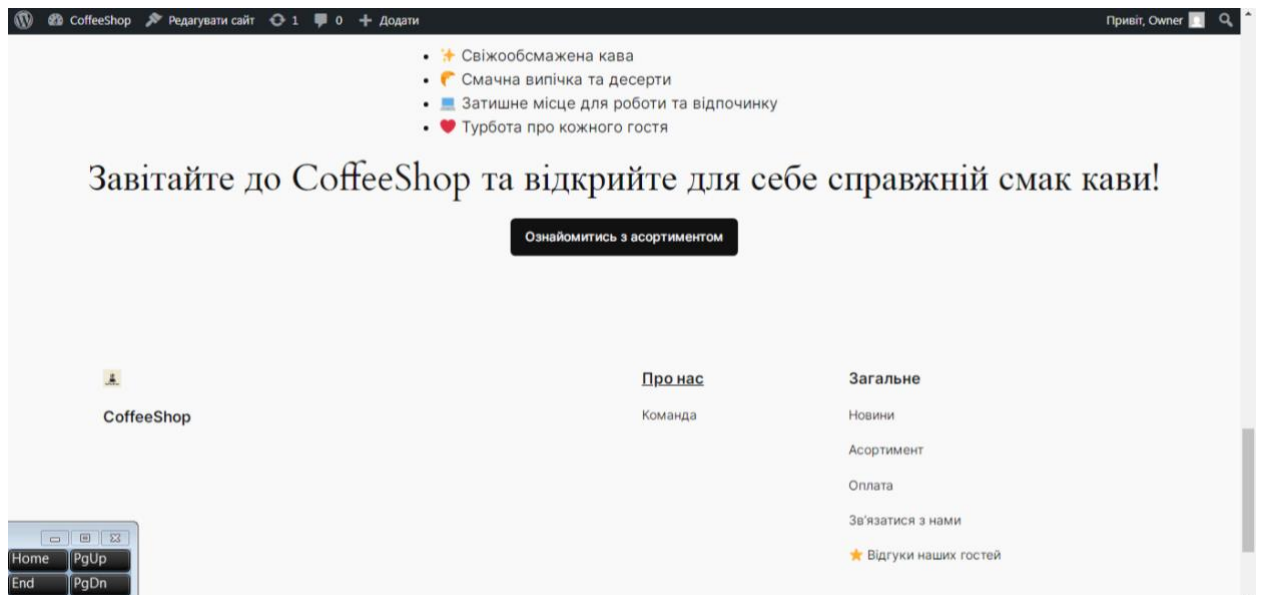


Рис. 2.36 – Головна сторінка вебсайту кав'ярні «CoffeeShop»

На сайті реалізовано розділ новин, який дозволяє публікувати інформацію про нові напої, акційні пропозиції, сезонне меню та інші події кав'ярні. Для керування новинами використовуються стандартні можливості WordPress щодо створення та редагування записів.

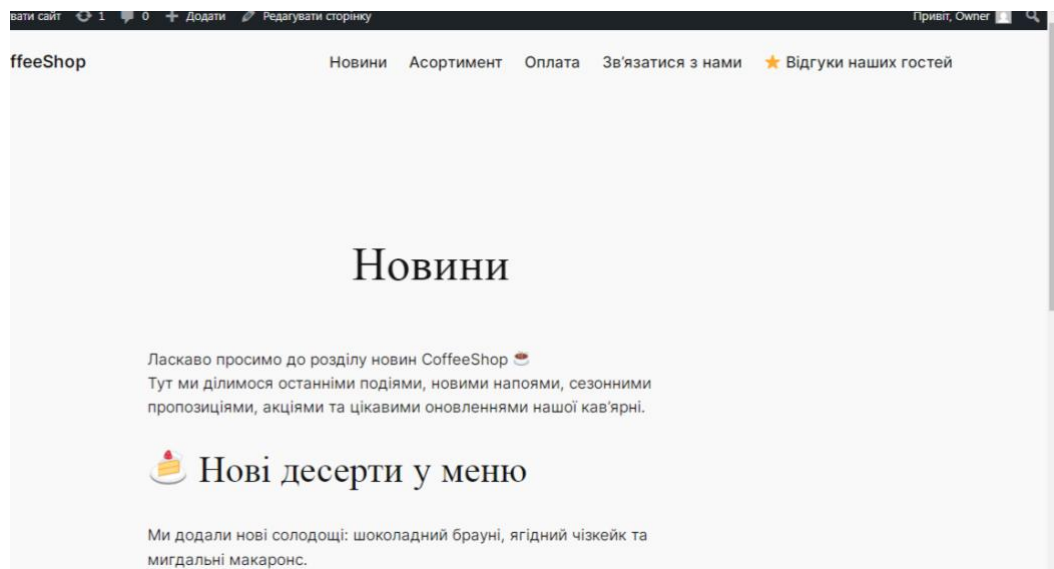


Рис. 2.37 Сторінка новин вебсайту

Однією з ключових функцій системи є можливість оформлення замовлення через вебінтерфейс. Користувач може обрати категорію продукції, переглянути доступні товари, вказати кількість позицій та обрати столик або

варіант замовлення «з собою». Після підтвердження інформація автоматично зберігається в базі даних.

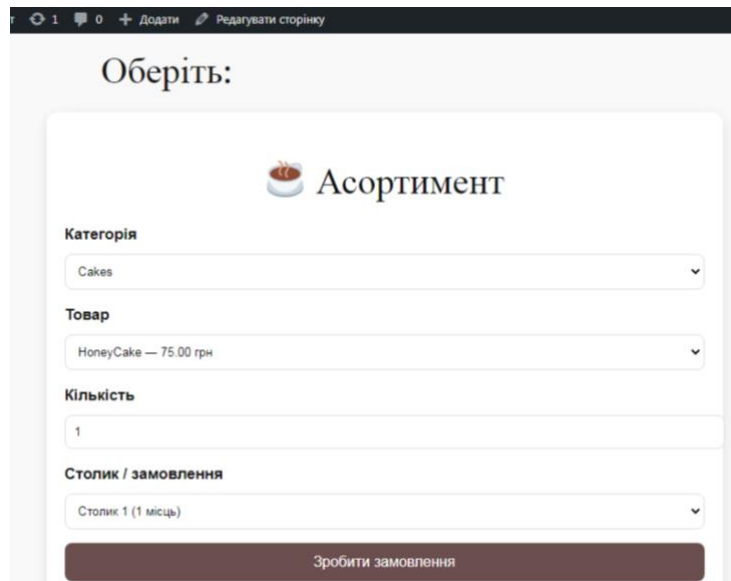


Рис. 2.38 Форма оформлення замовлення

Для забезпечення завершення процесу обслуговування реалізовано сторінку оплати замовлення. Після створення замовлення користувач переходить до сторінки оплати, де відображається інформація про номер замовлення, його статус, перелік обраних товарів та загальна сума до сплати.

Система підтримує два способи оплати:

- готівкою;
- банківською карткою.

Після підтвердження оплати статус замовлення автоматично змінюється на виконане, а інформація про оплату оновлюється в базі даних.

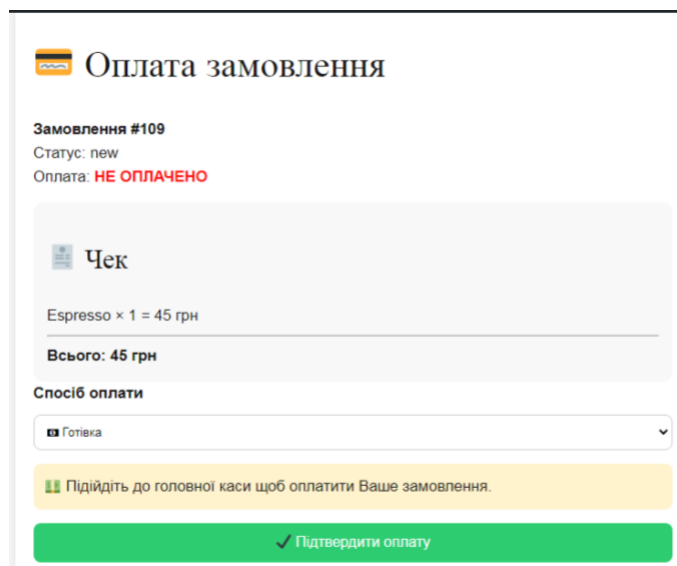


Рис. 2.39 Сторінка оплати замовлення

Для розширення функціональних можливостей вебсайту використовувалися плагіни WordPress та власні програмні модулі (сніпети), які забезпечують взаємодію між вебінтерфейсом і базою даних. Завдяки цьому реалізовано автоматичне створення замовлень та збереження інформації про оплату без необхідності ручного введення даних.

Розроблений вебсайт дозволяє автоматизувати основні процеси роботи кав'ярні, забезпечує зручну взаємодію з клієнтами та сприяє підвищенню ефективності діяльності закладу. Використання WordPress у поєднанні з базою даних MySQL забезпечує гнучкість, масштабованість та простоту подальшого супроводу системи.

## ВИСНОВКИ

У бакалаврській роботі було розглянуто питання розробки комплексної інформаційної системи для підвищення ефективності роботи кав'ярні. Проведене дослідження підтвердило актуальність використання сучасних інформаційних технологій для автоматизації діяльності закладів громадського харчування, особливо в умовах цифровізації бізнесу та підвищення вимог до якості обслуговування клієнтів.

У першому розділі було проаналізовано особливості діяльності кав'ярень в умовах воєнного стану, визначено основні проблеми та виклики, з якими стикаються сучасні заклади громадського харчування. Досліджено сучасні інформаційні системи автоматизації діяльності кав'ярень і ресторанів, зокрема Poster POS, SkyService POS та Restora POS, а також проведено порівняльний аналіз вебсайтів популярних кав'ярень. На основі проведеного аналізу було визначено основні функціональні вимоги до інформаційної системи та вебсайту кав'ярні.

У другому розділі розроблено логічну модель бази даних MySQL, яка забезпечує зберігання інформації про працівників, клієнтів, товари, категорії товарів, робочі зміни та замовлення. Структура бази даних реалізована з використанням реляційної моделі та підтримує необхідні зв'язки між таблицями для забезпечення цілісності даних.

У ході виконання роботи було створено комплексну інформаційну систему кав'ярні «CoffeeShop», яка забезпечує автоматизацію основних бізнес-процесів закладу. Реалізовано механізм авторизації користувачів із розмежуванням прав доступу між власником і працівниками, функції управління замовленнями, клієнтами, товарами, категоріями продукції та робочими змінами. Також забезпечено можливість формування звітної та аналітичної інформації для підтримки управлінських рішень.

Окрему увагу приділено розробці вебсайту кав'ярні «CoffeeShop» на базі системи управління контентом WordPress. Для реалізації проєкту використовувалися WordPress, MySQL, phpMyAdmin, XAMPP та Visual Studio

Code. Створений вебсайт забезпечує інформаційну підтримку діяльності кав'ярні, надає можливість перегляду новин, оформлення замовлень та виконання їх оплати через вебінтерфейс.

Розроблена система дозволяє скоротити час обробки замовлень, підвищити точність обліку інформації, зменшити кількість помилок під час роботи персоналу та забезпечити централізоване зберігання даних. Використання вебтехнологій і системи керування контентом WordPress забезпечує простоту адміністрування та можливість подальшого розвитку функціональних можливостей системи.

Таким чином, поставлена мета бакалаврської роботи — розробка комплексної інформаційної системи для підвищення ефективності роботи кав'ярні — була повністю досягнута. Усі поставлені завдання виконано, а отримані результати можуть бути використані для практичної автоматизації діяльності реальної кав'ярні та подальшого вдосконалення інформаційної підтримки закладів громадського харчування.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Дата К. Дж. Вступ до систем баз даних : пер. з англ. — К. : Видавнича група BHV, 2018. — 864 с.
2. Елмасрі Р., Наватхе Ш. Основи систем баз даних. — 7-ме вид. — К. : Діалектика, 2020. — 1200 с.
3. Соммервілл І. Інженерія програмного забезпечення. — 10-те вид. — К. : Вільямс, 2017. — 864 с.
4. Microsoft. C# documentation [Електронний ресурс]. — Режим доступу: <https://learn.microsoft.com/dotnet/csharp/> (дата звернення: 29.11.2025).
5. Інформаційна система автоматизації закладів громадського харчування Poster POS. URL: <https://joinposter.com/ua> (дата звернення: 31.05.2026).
6. Автоматизація ресторанів та кав'ярень SkyService POS. URL: <https://skyservice.pro/uk/main/> (дата звернення: 31.05.2026).
7. Система автоматизації ресторанів Restora POS. URL: <https://restorapos.com/> (дата звернення: 31.05.2026).
8. WordPress. Content Management System. URL: <https://wordpress.org> (дата звернення: 25.04.2026).
9. MySQL Database Management System. URL: <https://www.mysql.com> (дата звернення: 12.05.2026).
10. Apache Friends XAMPP. URL: <https://www.apachefriends.org> (дата звернення: 12.05.2026).
11. Starbucks Corporation. Official Website. URL: <https://www.starbucks.com> (дата звернення: 13.05.2026).
12. Aroma Kava. Офіційний сайт. URL: <https://aromakava.ua> (дата звернення: 21.05.2026).
13. Lviv Croissants. Офіційний сайт. URL: <https://lvivcroissants.com> (дата звернення: 21.05.2026).
14. Coffee House. Офіційний сайт. URL: <https://coffeehouse.ua> (дата звернення: 21.05.2026).
15. phpMyAdmin Documentation. URL: <https://www.phpmyadmin.net> (дата

звернення: 21.05.2026).

## ДОДАТКИ

### Код форми авторизації користувачів (LoginForm.cs)

```
using Cafes.Helpers;
using Cafes.Models;
using MySql.Data.MySqlClient;
using System;
using System.Collections.Generic;
using System.Windows.Forms;

namespace Cafes
{
    public partial class LoginForm : Form
    {
        static readonly public MySqlConnection Connection = new
        MySqlConnection();
        static public Employee Employee { get; set; }

        private readonly List<string> Roles = new List<string> {
        CafeConstants.Roles.Owner, CafeConstants.Roles.Employee };

        public static string Role { get; set; }

        public LoginForm()
        {
            InitializeComponent();

            rolesComboBox.DataSource = Roles;
            rolesComboBox.SelectedIndex = -1;
        }
    }
}
```

```

    }

    private void rolesComboBox_SelectedIndexChanged_1(object sender,
EventArgs e)
    {
        var selectedRole = rolesComboBox.SelectedValue?.ToString();

        loginTextBox.Enabled = selectedRole ==
CafeConstants.Roles.Employee;
    }

    private void loginBtn_Click(object sender, EventArgs e)
    {
        if (rolesComboBox.SelectedIndex == -1 ||
string.IsNullOrEmpty(passwordTextBox.Text))
        {
            MessageBox.Show("Incorrect input please select a role
and enter password.");
            return;
        }

        Role = rolesComboBox.SelectedItem.ToString();

        Connection.ConnectionString =
$"server={CafeConstants.Connection.Server};User
Id={Role.ToLowerInvariant()};password={passwordTextBox.Text};Persist
Security Info=True;database={CafeConstants.Connection.Database}";

        try
        {

```

```

        Connection.Open();

        TrySetEmployee();

        MainMenuForm mainMenu = new MainMenuForm();
        mainMenu.Show();

        Hide();
    }
    catch
    {
        Connection.Close();
        MessageBox.Show("Failed to connect", "Attention",
        MessageBoxButtons.OK, MessageBoxIcon.Question);
    }
}

private void TrySetEmployee()
{
    if (loginTextBox.Enabled)
    {
        using (var cmd = new
        MySqlCommand(CafeConstants.Queries.Employees.GetEmployeeByLogin,
        Connection))
        {
            cmd.Parameters.AddWithValue("@login",
            loginTextBox.Text);

            using (var reader = cmd.ExecuteReader())
            {

```

```

        if (reader.Read())
        {
            if
(reader.IsDBNull(reader.GetOrdinal("Id")))
            {
                throw new
KeyNotFoundException("Employee ID is NULL in the database.");
            }

            Employee = new Employee()
            {
                Id = reader.GetInt32("Id"),
                Name
= reader.GetString("Name")
            };
        }
        else
        {
            throw new
KeyNotFoundException("Employee ID is NULL in the database.");
        }
    }
}
}
}
}

```

### Код головного меню системы (MainMenuForm.cs)

```
using Cafes.Helpers;
```

```

using MySql.Data.MySqlClient;
using System;
using System.Text;
using System.Windows.Forms;

namespace Cafes
{
    public partial class MainMenuForm : Form
    {
        public MainMenuForm()
        {
            InitializeComponent();

            ShowMenuItemsAccordingToRole();
        }

        private void MainMenu_FormClosed(object sender,
FormClosedEventArgs e)
            => CloseApp();

        private void employeesToolStripMenuItem_Click(object sender,
EventArgs e)
            => new EmployeesForm().Show();

        private void tablesToolStripMenuItem_Click(object sender, EventArgs
e)
            => new TablesForm().Show();

        private void clientsToolStripMenuItem_Click(object sender,
EventArgs e)

```

```
=> new ClientsForm().Show();
```

```
private void categoriesToolStripMenuItem_Click(object sender,  
EventArgs e)
```

```
=> new CategoriesForm().Show();
```

```
private void itemsToolStripMenuItem_Click(object sender, EventArgs  
e)
```

```
=> new ItemsForm(-1).Show();
```

```
private void shiftsToolStripMenuItem_Click(object sender, EventArgs  
e)
```

```
=> new ShiftsForm().Show();
```

```
private void ordersToolStripMenuItem_Click(object sender,  
EventArgs e)
```

```
=> new ViewOrdersForm().Show();
```

```
private void orderButton_Click(object sender, EventArgs e)
```

```
{
```

```
    if (EmployeeHasOpenShift())
```

```
    {
```

```
        new NewOrderForm().Show();
```

```
    }
```

```
}
```

```
private void CloseApp()
```

```
{
```

```
    LoginForm.Connection.Close();
```

```

        Application.Exit();
    }

    private void ShowMenuItemsAccordingToRole()
    {
        if (LoginForm.Employee != null)
        {
            employeesToolStripMenuItem.Visible = false;
        }
        else
        {
            orderButton.Visible = false;
        }
    }

    private bool EmployeeHasOpenShift()
    {
        using (var cmd = new
        MySqlCommand(CafeConstants.Queries.Shifts.GetOpenShiftsByEmployee,
        LoginForm.Connection))
        {
            cmd.Parameters.AddWithValue("@employeeId",
            Convert.ToInt32(LoginForm.Employee?.Id));

            using (var reader = cmd.ExecuteReader())
            {
                if (!reader.Read())
                {
                    MessageBox.Show($"You don't have an open
shift. Please create one before creating orders.", "Attention",

```

```

MessageBoxButtons.OK, MessageBoxIcon.Error);

                return false;
            }
        }
    }

    return true;
}

private void checkStockToolStripMenuItem_Click(object sender,
EventArgs e)
{
    try
    {
        using (var cmd = new
MySQLCommand(CafeConstants.Queries.Items.AverageItemOrderingPerShift,
LoginForm.Connection))
        {
            using (var reader = cmd.ExecuteReader())
            {
                var sb = new StringBuilder();

                while (reader.Read())
                {
                    string itemName =
reader["ItemName"].ToString();
                    int available =
Convert.ToInt32(reader["Available"]);
                    double shiftsAvailable =
Convert.ToInt32(reader["ShiftsAvailable"]);

```

```

sb.AppendLine(
    $"[{itemName}]      ShiftsLeft:
{shiftsAvailable,3} | Available: {available,3}"
);
}

string message = sb.Length > 0
    ? "The following items are soon out of
stock:\n\n" + sb.ToString()
    : "All items have sufficient stock.";

MessageBox.Show(message, "Stock Check",
    MessageBoxButtons.OK,
    sb.Length > 0 ?
    MessageBoxIcon.Warning : MessageBoxIcon.Information);
}
}
}
catch (Exception ex)
{
    MessageBox.Show("Error checking stock:\n" +
ex.Message, "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
}
}

}
}

```

**Код форми створення замовлення (NewOrderForm.cs)**

```

public partial class NewOrderForm : Form
{
    private BindingList<NewOrderItem> NewOrderItems =
        new BindingList<NewOrderItem>();

    public NewOrderForm()
    {
        InitializeComponent();
        previewGridView.DataSource = NewOrderItems;
        FillComboBoxes();
    }

    private void addToOrderButton_Click(object sender, EventArgs e)
    {
        NewOrderItems.Add(new NewOrderItem
        {
            ItemId = Convert.ToInt32(itemComboBox.SelectedValue),
            Title = itemComboBox.Text,
            Count = Convert.ToInt32(amountUpDown.Value)
        });
    }

    private void saveButton_Click(object sender, EventArgs e)
    {
        int orderId = CreateOrder();
        AddItemsToNewOrder(orderId);

        ReceiptForm receipt = new ReceiptForm();
        receipt.LoadReceipt(orderId);
        receipt.ShowDialog();
    }
}

```

```
}  
}
```

### **Код форми формування та друку чека (ReceiptForm.cs)**

```
public partial class ReceiptForm : Form  
{  
    public void LoadReceipt(int orderId)  
    {  
        List<ReceiptItem> items = GetOrderItems(orderId);  
  
        StringBuilder receipt = new StringBuilder();  
        decimal total = 0;  
  
        foreach (var item in items)  
        {  
            receipt.AppendLine(  
                $"{item.Name} x {item.Count} = {item.TotalPrice}");  
  
            total += item.TotalPrice;  
        }  
  
        receipt.AppendLine($"Разом: {total}");  
  
        receiptRichTextBox.Text = receipt.ToString();  
    }  
}
```

### **Код форми управління товарами меню (ItemsForm.cs)**

```
using MySql.Data.MySqlClient;  
using System;
```

```

using System.Data;
using System.Globalization;
using System.Windows.Forms;

namespace Cafes.Helpers
{
    public partial class ItemsForm : Form
    {
        private bool IsEditMode { get; set; } = false;
        private int FilterCategoryId { get; set; }

        public ItemsForm(int categoryId)
        {
            InitializeComponent();

            FilterCategoryId = categoryId;
            filterCategoryBox.SelectedValue = FilterCategoryId;

            RefreshGrid();
            SetValuesForComboBoxes();
            SetFieldsStateBasedOnRole();
        }

        private void saveButton_Click(object sender, EventArgs e)
        {
            if (!IsValidInput())
            {
                return;
            }
        }
    }
}

```

```

        if (IsInEditMode)
        {
            MySqlCommand updateCommand = new
            MySqlCommand(CafeConstants.Queries.Items.UpdateItem,
            LoginForm.Connection);

            updateCommand.Parameters.AddWithValue("@name",
            nameTextBox.Text);

            updateCommand.Parameters.AddWithValue("@categoryId",
            categoryComboBox.SelectedValue);

            updateCommand.Parameters.AddWithValue("@price",
            priceUpDown.Value);

            updateCommand.Parameters.AddWithValue("@discountPrice",
            discountUpDown.Value);

            updateCommand.Parameters.AddWithValue("@available",
            availableUpDown.Value);

            updateCommand.Parameters.AddWithValue("@profit",
            profitUpDown.Value);

            updateCommand.Parameters.AddWithValue("@id",
            itemsDataGridView.CurrentRow.Cells[0].Value);

            updateCommand.ExecuteNonQuery();
        }
        else
        {
            MySqlCommand insertCommand = new
            MySqlCommand(CafeConstants.Queries.Items.AddNewItem,
            LoginForm.Connection);

            insertCommand.Parameters.AddWithValue("@name",

```

```
nameTextBox.Text);
```

```
        insertCommand.Parameters.AddWithValue("@categoryId",  
categoryComboBox.SelectedValue);
```

```
            insertCommand.Parameters.AddWithValue("@price",  
priceUpDown.Value);
```

```
            insertCommand.Parameters.AddWithValue("@discountPrice",  
discountUpDown.Value);
```

```
            insertCommand.Parameters.AddWithValue("@available",  
availableUpDown.Value);
```

```
            insertCommand.Parameters.AddWithValue("@profit",  
profitUpDown.Value);
```

```
            insertCommand.ExecuteNonQuery();
```

```
                FilterCategoryId =  
Convert.ToInt32(categoryComboBox.SelectedValue);  
                filterCategoryBox.SelectedValue = FilterCategoryId;  
            }
```

```
        RefreshGrid();
```

```
        itemsDataGridView.SelectUpsertedRow(nameTextBox.Text);
```

```
        ClearInputFields();
```

```
    }
```

```
private void clearButton_Click(object sender, EventArgs e)
```

```
    => ClearInputFields();
```

```
private void editToolStripMenuItem_Click(object sender, EventArgs
```

e)

```
=> GoToEditMode();
```

```
private void itemsDataGridView_CellDoubleClick(object sender,
DataGridViewCellEventArgs e)
```

```
=> GoToEditMode();
```

```
private void removeToolStripMenuItem_Click(object sender,
EventArgs e)
```

```
{
```

```
    DialogResult dialogResult = MessageBox.Show
```

```
        ("Are you sure?", "Remove", MessageBoxButtons.YesNo,
        MessageBoxIcon.Information);
```

```
    if (dialogResult == DialogResult.Yes)
```

```
    {
```

```
        try
```

```
        {
```

```
            MySqlCommand deleteCommand = new
            MySqlCommand(CafeConstants.Queries.Items.DeleteItem,
            LoginForm.Connection);
```

```
            deleteCommand.Parameters.AddWithValue("@id",
            itemsDataGridView.CurrentRow.Cells[0].Value);
```

```
            deleteCommand.ExecuteNonQuery();
```

```
            RefreshGrid();
```

```
            ClearInputFields();
```

```
        }
```

```
    catch
```

```
    {
```

```

        MessageBox.Show("Unable to remove an item. It
has connected data", "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
}

```

```

private void filterCategoryBox_SelectionChangeCommitted(object
sender, EventArgs e)
{
    FilterCategoryId =
Convert.ToInt32(filterCategoryBox.SelectedValue);

    RefreshGrid();
}

```

```

private void SetFieldsStateBasedOnRole()
{
    itemsDataGridView.Columns[0].Visible = false;
    itemsDataGridView.Columns[7].Visible = false;

    if (LoginForm.Employee != null)
    {
        itemsDataGridView.Columns[6].Visible = false;

        nameTextBox.Enabled = false;
        categoryComboBox.Enabled = false;
        priceUpDown.Enabled = false;
        discountUpDown.Enabled = false;
        availableUpDown.Enabled = false;
        profitUpDown.Enabled = false;
    }
}

```

```

        profitUpDown.Value = profitUpDown.Maximum;

        saveButton.Enabled = false;
        clearButton.Enabled = false;

        itemMenuStrip.Enabled = false;
    }
}

private void SetValuesForComboBoxes()
{
    using (MySqlDataAdapter dataAdapter = new
        MySqlDataAdapter(CafeConstants.Queries.Categories.GetCategories,
        LoginForm.Connection))
    {
        DataTable dataTable = new DataTable
        {
            Locale = CultureInfo.InvariantCulture
        };
        dataAdapter.Fill(dataTable);

        filterCategoryBox.DataSource = dataTable;
        filterCategoryBox.DisplayMember = "Name";
        filterCategoryBox.ValueMember = "Id";
        filterCategoryBox.SelectedValue = FilterCategoryId;

        categoryComboBox.DataSource = dataTable.Copy();
        categoryComboBox.DisplayMember = "Name";
        categoryComboBox.ValueMember = "Id";
        categoryComboBox.SelectedIndex = -1;
    }
}

```

```

    }
}

private void ClearInputFields()
{
    categoryComboBox.SelectedIndex = -1;
    nameTextBox.Clear();
    priceUpDown.Value = priceUpDown.Minimum;
    discountUpDown.Value = discountUpDown.Minimum;
    availableUpDown.Value = availableUpDown.Minimum;
    profitUpDown.Value = profitUpDown.Minimum;
    IsEditMode = false;
}

private void GoToEditMode()
{
    if (LoginForm.Employee != null ||
itemsDataGridview.RowCount < 1)
    {
        return;
    }

    IsEditMode = true;
    categoryComboBox.SelectedValue =
itemsDataGridview.CurrentRow.Cells[7].Value;
    nameTextBox.Text =
itemsDataGridview.CurrentRow.Cells[1].Value.ToString();
    priceUpDown.Value =
Convert.ToDecimal(itemsDataGridview.CurrentRow.Cells[3].Value);
    discountUpDown.Value =

```

```

Convert.ToDecimal(itemsDataGridView.CurrentRow.Cells[4].Value);
        availableUpDown.Value =
Convert.ToDecimal(itemsDataGridView.CurrentRow.Cells[5].Value);
        profitUpDown.Value =
Convert.ToDecimal(itemsDataGridView.CurrentRow.Cells[6].Value);
    }

    private bool IsInputValid()
    {
        if (categoryComboBox.SelectedIndex < 0)
        {
            MessageBox.Show("Please select a category",
"Attention", MessageBoxButtons.OK, MessageBoxIcon.Error);
            return false;
        }

        if (string.IsNullOrEmpty(nameTextBox.Text))
        {
            MessageBox.Show("Name cannot be empty", "Attention",
MessageBoxButtons.OK, MessageBoxIcon.Error);
            return false;
        }

        if (discountUpDown.Value > priceUpDown.Value)
        {
            MessageBox.Show("Discount should be less than price",
"Attention", MessageBoxButtons.OK, MessageBoxIcon.Error);
            return false;
        }
    }

```

```

        if (profitUpDown.Value > priceUpDown.Value)
        {
            MessageBox.Show("Profit should be less than price",
                "Attention", MessageBoxButtons.OK, MessageBoxIcon.Error);
            return false;
        }

        using (var cmd = new
            MySqlCommand(CafeConstants.Queries.Items.GetItemByName,
                LoginForm.Connection))
        {
            cmd.Parameters.AddWithValue("@name",
                nameTextBox.Text);

            using (var reader = cmd.ExecuteReader())
            {
                if (reader.Read() && !IsEditMode)
                {
                    MessageBox.Show("Name should be
                        unique", "Attention", MessageBoxButtons.OK, MessageBoxIcon.Error);
                    return false;
                }
            }
        }

        return true;
    }

    private void RefreshGrid()
    {

```

```

        MySqlDataAdapter dataAdapter = GetDataAdapter();

        DataTable dataTable = new DataTable();
        dataAdapter.Fill(dataTable);

        itemsDataGridView.DataSource = dataTable;
        itemsDataGridView.Refresh();
    }

    private MySqlDataAdapter GetDataAdapter()
    {
        MySqlDataAdapter adapter;

        if (FilterCategoryId != -1)
        {
            adapter = new
MySqlDataAdapter(CafeConstants.Queries.Items.GetItemByCategory,
LoginForm.Connection);

            adapter.SelectCommand.Parameters.AddWithValue("@categoryId",
FilterCategoryId);
        }
        else
        {
            adapter = new
MySqlDataAdapter(CafeConstants.Queries.Items.GetItem,
LoginForm.Connection);
        }

        return adapter;
    }

```

```
    }

    private void itemsPopularityToolStripMenuItem_Click(object sender,
EventArgs e)
    {
        => ReportExtensions.GenerateItemsPopularityReport();
    }
}
```

