Національний університет «Полтавська політехніка імені Юрія Кондратюка»

(повне найменування вищого навчального закладу)

\_Навчально науковий інститут інформаційних технологій та робототехніки

(повна назва факультету)

\_Кафедра комп’ютерних та інформаційних технологій і систем\_

(повна назва кафедри)

**Пояснювальна записка**

**до дипломного проекту (роботи)**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_бакалавра\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(освітньо-кваліфікаційний рівень)

на тему

\_\_\_Розроблення інформаційної системи сфери обслуговування\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_на прикладі кав’ярні\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Виконала: студентка\_4\_ курсу, групи \_402-ТН\_\_

спеціальності

\_\_\_\_\_\_\_\_\_\_\_\_\_\_122 Комп’ютерні науки \_\_\_\_\_\_\_\_\_\_\_

(шифр і назва напряму)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Карунна Є. Ю.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище та ініціали)

Керівник \_\_\_\_\_\_\_\_\_\_Дмитренко Т. А.\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище та ініціали)

Рецензент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(прізвище та ініціали)

Полтава – 2021 року

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ**

**«ПОЛТАВСЬКА ПОЛІТЕХНІКА ІМЕНІ ЮРІЯ КОНДРАТЮКА»**

**НАВЧАЛЬНО НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА РОБОТОТЕХНІКИ**

**КАФЕДРА КОМП’ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ**

**КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА**

**спеціальність 122 «Комп’ютерні науки»**

**на тему**

**«Розроблення інформаційної системи сфери обслуговування на прикладі кав’ярні»**

**Студентки групи 402-ТН Карунної Євгенії Юріївни**

Керівник роботи

кандидат технічних наук,

доцент Дмитренко Т.А.

Завідувач кафедри

кандидат технічних наук,

доцент Головко Г.В.

Полтава – 2021

**РЕФЕРАТ**

Кваліфікаційна робота бакалавра: 57с., 34 малюнки, 2 таблиці, 2 додатки, 20 джерел.

**Об’єкт дослідження**: діяльність інформаційної системи кав’ярні для перегляду меню закладу та замовлення товарів системи.

**Мета роботи**: розробка інформаційної системи кав’ярні, що дозволяє переглядати меню закладу, наявні товари та повну інформацію про товар.

**Методи**: проєктування та розробка бази даних для функціонування інформаційної системи кав’ярні, розробка макетів користувацького інтерфейсу та програмна реалізація інтерфейсу.

**Ключові слова**: інформаційна система, програмний код, кав’ярня, дизайн, програмний модуль, база даних.

**ANNOTATION**

Bachelor's qualification work : 57p., 34 pictures, 2 tables, 2 appendices, 20 sources.

**Object of research:** activity of the information system of the coffee shop for viewing the menu of the establishment and ordering the goods of the system.

**Purpose:** development of an information system of the coffee shop, which allows you to view the menu of the institution, available products and complete information about the product.

**Methods:** database’s design and development for the operation of the information system of the coffee shop, development of user interface layouts and software implementation of the interface.

**Keywords:** information system, program code, coffee shop, design, software module, database.

**ЗМІСТ**

[ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ 5](#_Toc74441737)

[ВСТУП 6](#_Toc74441738)

[РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ 8](#_Toc74441739)

[1.1 Опис предметної області 8](#_Toc74441740)

[1.2 Аналіз наявних рішень 10](#_Toc74441741)

[1.3 Аналіз функціональних особливостей 15](#_Toc74441742)

[1.4 Основні функції інформаційних систем кав’ярні 17](#_Toc74441743)

[РОЗДІЛ 2 ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ КАВ’ЯРНІ 19](#_Toc74441744)

[2.1 Структурна та функціональна частини системи 19](#_Toc74441745)

[2.2 Проєктування бази даних 24](#_Toc74441746)

[2.3 Розробка дизайну інтерфейсу користувача 26](#_Toc74441747)

[РОЗДІЛ 3 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КАВ’ЯРНІ 29](#_Toc74441748)

[3.1 Вибір платформи 29](#_Toc74441749)

[3.2 Розробка бази даних 31](#_Toc74441750)

[3.3 Розробка користувацького інтерфейсу 33](#_Toc74441751)

[РОЗДІЛ 4 ТЕСТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ КАВ’ЯРНІ 43](#_Toc74441752)

[4.1 Вибір способу тестування 43](#_Toc74441753)

[4.2 Тест план 44](#_Toc74441754)

[4.3 Введення в експлуатацію 48](#_Toc74441755)

[ВИСНОВКИ 49](#_Toc74441756)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 50](#_Toc74441757)

[ДОДАТОК А ЛІСТИНГ КОДУ ДЛЯ РЕАЛІЗАЦІЇ РЕДАГУВАННЯ ТОВАРІВ 52](#_Toc74441758)

[ДОДАТОК Б ЛІСТИНГ КОДУ ДЛЯ РЕАЛІЗАЦІЇ КОШИКУ 54](#_Toc74441759)

# ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

**SQL** – Structured Query Language

**ER-діаграма** – Entity-relationship діаграма

**БД** – база даних

**СУБД** – система управління базами даних

**ПЗ** – програмне забезпечення

**ІС** – інформаційна система

**JS** – JavaScript

**HTTP** – HyperText Transfer Protocol

**HTML** – HyperText Markup Language

**PHP** – Hypertext Preprocessor

**CSS** – Cascading Style Sheets

**URL** – Uniform Resource Locator

**LAMP –** Linux Apache MariaDB / MySQL PHP

**КБ –** Кілобайт

# ВСТУП

Інформаційні технології все більше і більше зачіпають сфери діяльності людини і дають можливість значно спростити найрізноманітніші операції, які щодня виконуються самими різними фахівцями у великих і середніх компаніях. І зараз під натиском інформаційних і телекомунікаційних технологій необхідне введення інформаційних систем в ті області де вони не застосовуються або слабо розвинені і які допоможуть зменшити витрати, час на обробку даних, і збільшити продуктивність праці.

Сфера послуг є однією з найперспективніших сфер економіки й економічної діяльності. Вона розширює спектр економічної діяльності і єдиним комплексом. У процесі комерціалізації української економіки та при українських реаліях: великі відстані, гостра нестача часу; особливого значення набуває така важлива форма реалізації товарів і послуг, як торгівля і реклама через мережу Інтернет. У мережі можна рекламувати послуги, продавати товари або представляти споживачеві і те й інше. У сучасному бізнесі багато що залежить від самопрезентації компанії, її позиціювання на ринку послуг, що надаються і можливостей шукати нових клієнтів і ринки збуту. Одним з інструментів, як іміджевих, так і маркетингових, є наявність своєї системи в мережі Інтернет.

Інформаційна система перетворює компанію в сучасний бізнес і вона являється важливим джерелом інформації для використання клієнтів і людей, чия думка є суспільно значущою. Вже зараз на світовому ринку інформаційних систем спостерігається постійний попит з боку декількох компаній. Проблеми різняться в залежності від розміру бізнесу. Для малих підприємств контроль операцій і зберігання інформації менш складні. У міру зростання компаній, особливо тих, які пов'язані з глобальними відносинами, інформаційні системи використовуються для точного управління операціями без збільшення числа співробітників.

Актуальність даної роботи обумовлена попитом якісної системи для кав’ярні. Спосіб залучення клієнтів за допомогою системи відрізняється відносно низькими витратами й великий розмір аудиторії.

Метою дипломного проєкту є розробка інформаційної системи, за допомогою якої можна буде ефективно поширювати інформацію про заклад, наданий товар і меню, а також залучати нових потенційних клієнтів.

Відповідно до поставленої мети в роботі визначені наступні завдання:

1. Вивчити технології по розробці інформаційних систем.
2. Визначити цілі, ідеї, потреби інформаційної системи.
3. Розробити технічну концепцію системи (структуру).
4. Розробити програмний продукт для закладу, який дозволить підвищити ефективність діяльності аналізованого об'єкта.

# РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

# 1.1 Опис предметної області

Український ринок кави стрімко розвивається. Сьогодні каву можна придбати практично на кожному кроці. Разом з цим зростає і конкуренція, яка має позитивний вплив на якості продукції, договір про вільні постачання з провідними країнами-виробниками – В'єтнамом і Бразилією і вступ в дію угоди про асоціацію з ЄС щодо стандартизації товарів і послуг роблять посильний внесок на розвиток ринку [2].

Українці більше вживають каву в кафе або ресторані, а також зростає тенденція купівлі і вживання кави з собою. Майже половина споживачів на ринку вживають даний напій 1 раз на місяць, а третина – 1 раз у 2-3 тижні.

Разом зі збільшенням попиту зростає й чисельність кав'ярень. На українському кавовому ринку присутні близько 10 великих гравців, кількість котрих постійно збільшується [3].

Оскільки конкуренція в даному сегменті ринку тільки зростає (у 2019 році було відкрито на 16% більше кав'ярень, ніж у 2018-му. Даних по 2020-му немає, але, швидше за все, через пандемію такого зростання не передбачається) кожен власник намагається заманити нового клієнта різними методами, починаючи від дизайну закладу і розробки концепції, закінчуючи просуванням закладу в Інтернеті через соціальні мережі або розробку інформаційної системи [1].

Інформаційні системи призначені для збору, зберігання і обробки інформації, тому в основі будь-якої з них лежить середовище зберігання і доступу до даних. Інформаційні системи орієнтовані на кінцевого користувача, що не володіє високою кваліфікацією в області обчислювальної техніки.

Існує багато типів інформаційних систем, в залежності від того, для чого вони призначені. Система підтримки операцій, така як система обробки транзакцій, перетворює бізнес-дані (фінансові транзакції) в цінну інформацію. Аналогічним чином інформаційна система управління використовує інформацію з бази даних для виведення звітів, допомагаючи користувачам і підприємствам приймати рішення на основі витягнутих даних .

Інформаційні системи є головним корпоративним активом як з точки зору вигоди, яку вони надають, так і з точки зору їх високих витрат. Тому організації повинні довгостроково планувати, купуючи інформаційні системи та послуги, які підтримуватимуть ділові ініціативи. У той же час фірми повинні реагувати на нові можливості. На основі довгострокових корпоративних планів та вимог різних осіб, від працівників до вищого керівництва, визначаються основні програми та встановлюються пріоритети проєкту [5].

Основними універсальними завданнями ІС є:

1. Виконання процесів перетворення інформації і видача її в зручному для сприйняття вигляді.
2. Тримання більш раціональних варіантів вирішення управлінських завдань за рахунок впровадження математичних методів, інтелектуальних систем і т.д..
3. Виробництво достовірної, надійної, своєчасної та систематизованої інформації.
4. Звільнення працівників від рутинної роботи за рахунок її автоматизації.
5. Вдосконалення структури потоків інформації і системи документообігу в фірмі.

Впровадження інформаційних систем в організації має наступні позитивні сторони:

1. Поліпшення товарів або послуг за рахунок підвищення якості та зменшенню витрат.
2. Збільшення ефективності і продуктивності.
3. Надання своєчасної і надійної інформації, що дозволяє поліпшити процес прийняття рішень.
4. Покращення комунікацій з колегами або клієнтами, в залежності від того хто використовує систему.

Для власників кав’ярень та потенційних або частих клієнтів інформаційна система може надати наступні переваги у використанні:

1. Дати користувачеві загальне уявлення про заклад і більше приводів завітати.
2. Залучити додаткових клієнтів шляхом активного просування своєї системи в Інтернеті.
3. Більше можливостей робітникам закладу для реклами, аналітики та удосконалення.
4. Підвищити престиж закладу.
5. Збільшити кількість продаж, шляхом меню і додаткових пропозицій.
6. Забезпечити запам'ятовуваність у потенційних клієнтів і відвідувачів.

# 1.2 Аналіз наявних рішень

Перед розробкою власної системи було проведено аналіз наступних наявних інформаційних систем кав’ярень: Aroma Kava [6], Gangster\_Coffe Shop [7], Світ кави [8], Blue Cup Coffee Shop [9].

Інформаційна система мережі кав’ярень Aroma Kava (рис.1.1) надає клієнту інформацію про місце розташування закладів мережі, контактні дані та зворотний зв’язок. Клієнт має можливість зареєструватися / авторизуватися на сервісі. В профілі користувач може бачити коли і де останнього разу була проведена купівля кави, а також збирати бонуси та розплачуватися ними.

В даній системі наявна вкладка з меню, яке зображено в відео форматі без вказання цін та назв позицій. Меню в зручному текстовому форматі відсутнє, клієнт заздалегідь не може зорієнтуватися яка сума знадобиться для відвідування закладу та наявні зараз актуальні пропозиції.

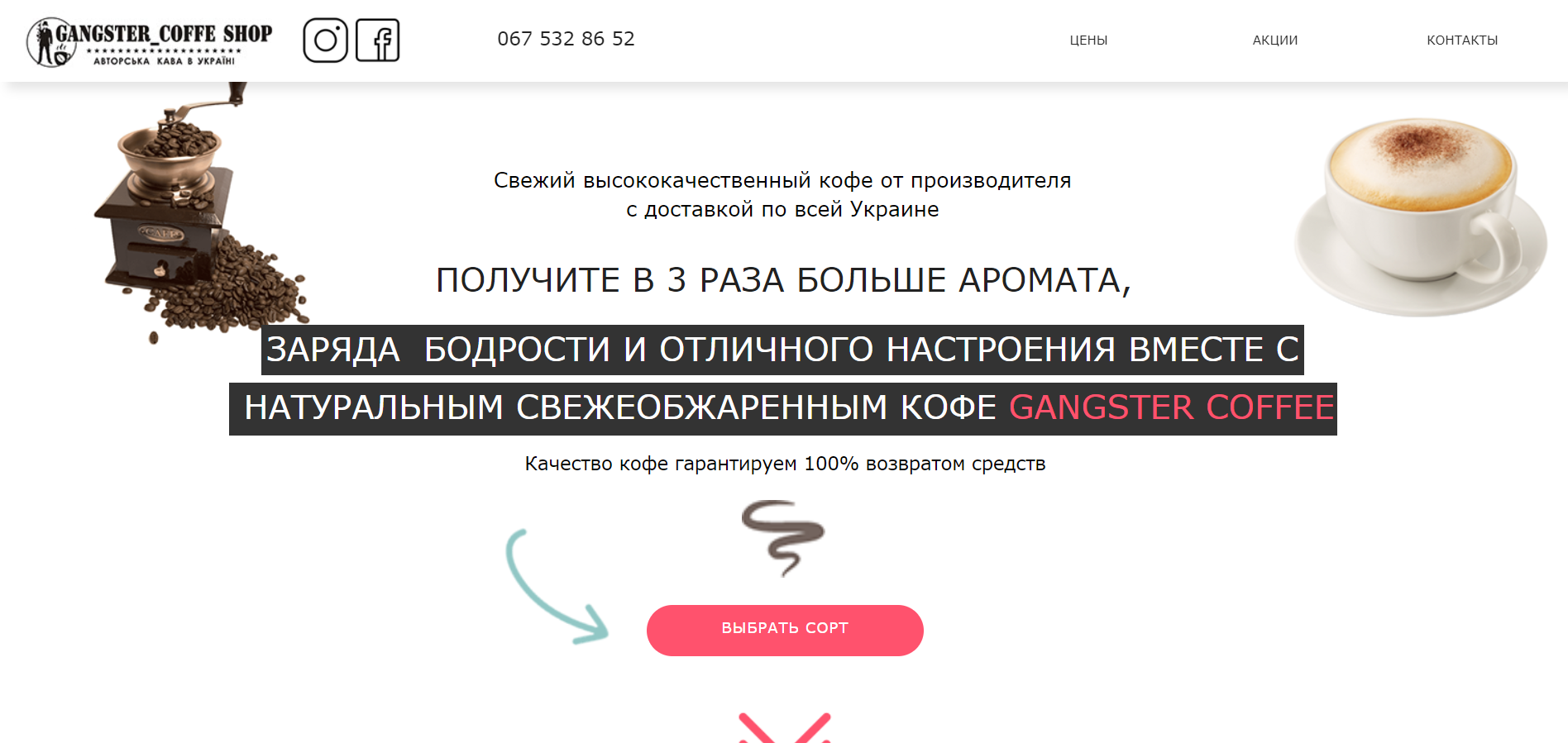
Інформаційна система надає дані про акції, які проводяться наразі, та загальні новини закладу (відкриття кав’ярні, нова позиція в меню, проведення різних заходів тощо).

  
Рисунок 1.1 – Головна сторінка кав’ярні “Aroma Kava”

Інформаційна система кав’ярні Gangster\_CoffeShop (рис.1.2) дає змогу купити обсмажену каву, надаючи при цьому інформацію про країну виробника продукту, ціну та вагу. Дану інформацію повинна надавати кожна система, котра продає каву на вагу, оскільки покупець має право знати що обирає та покупає.

Також в системі наявна інформація про місце розташування закладу, контактні дані та поле зі зворотним зв’язком.

Система не містить меню, що є великим недоліком і причиною повільного надходження нових клієнтів.

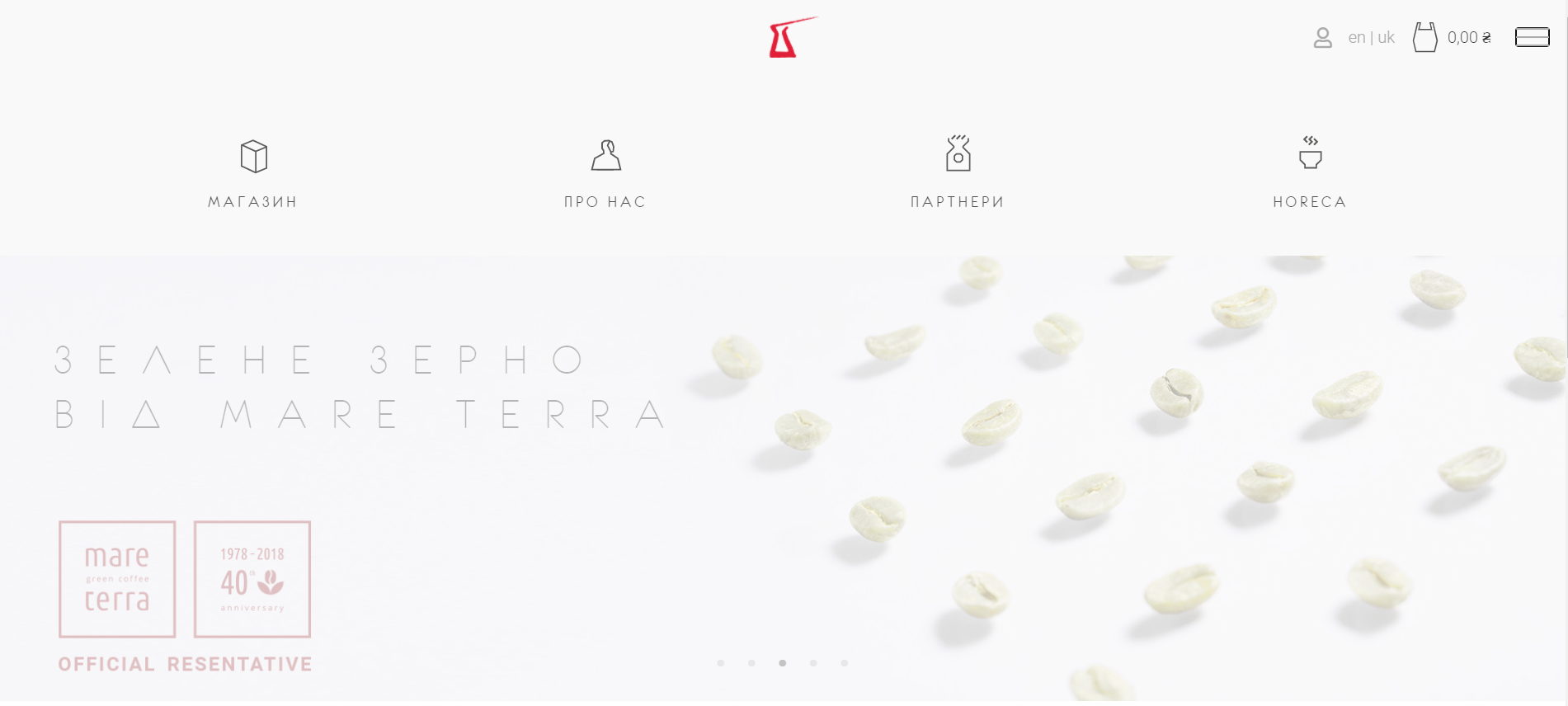
  
Рисунок 1.2 – Головна сторінка кав’ярні “Gangster\_Coffe Shop”

Інформаційна система кав’ярні Світ кави (рис.1.3) надає клієнту можливість зареєструватися / авторизуватися на сервісі. Наявність профілю в системі дає змогу переглядати придбані товари.

В системі наявне меню з актуальними цінами та позиціями. Клієнт зможе зорієнтуватися на середню ціну замовлення в чеку.

Також інформаційна система дає змогу на купівлю обсмаженої кави, надаючи при цьому інформацію про країну виробника продукту, ціну та вагу. Дану інформацію повинна надавати кожна система, котра продає каву на вагу, оскільки покупець має право знати що обирає та покупає.

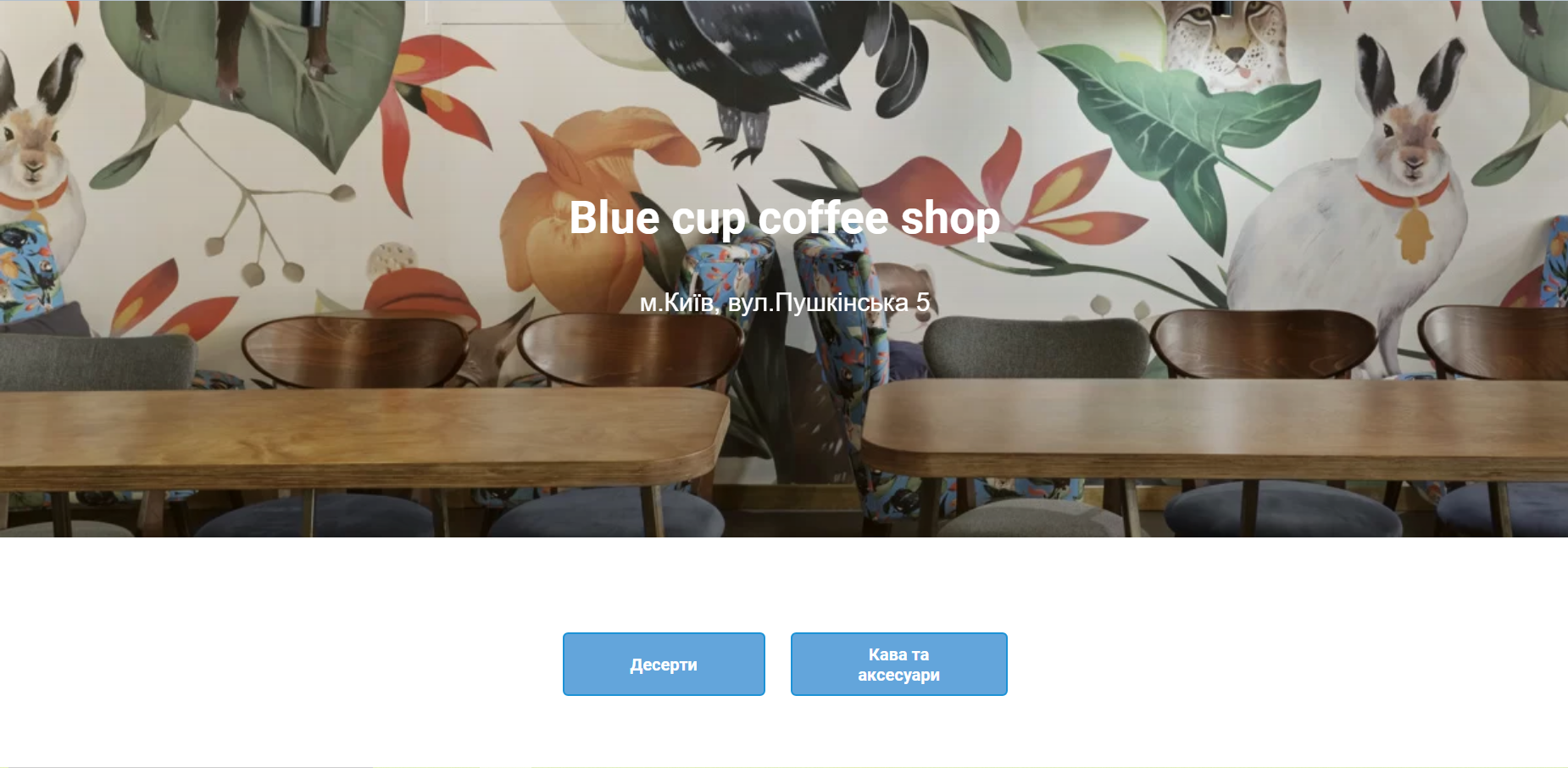
В системі наявна інформація про місце розташування закладу, контактні дані та поле зі зворотним зв’язком.

  
Рисунок 1.3 – Головна сторінка кав’ярні “Світ кави”

Інформаційна система кав’ярні Blue Cup Coffee Shop (рис.1.3) має меню з актуальними продуктами та цінами. Клієнт може заздалегідь зорієнтуватися в цінах та в наявних продуктах в кав’ярні.

Також система надає змогу придбати кавомолки, фільтри для заварювання кави та обсмажену каву, надаючи при цьому інформацію про країну виробника продукту, ціну та вагу. Дану інформацію повинна надавати кожна система, котра продає каву на вагу, оскільки покупець має право знати що обирає та покупає.

В системі наявна інформація про місце розташування закладу, контактні дані та поле зі зворотним зв’язком.

  
Рисунок 1.4 – Головна сторінка кав’ярні “ Blue Cup Coffee Shop”

Приведені вище аналоги мають зручний та інтуїтивно зрозумілий інтерфейс, також кожна система має індивідуальну кольорову гаму, яка не напружує очі. На всіх ресурсах наявні контакти, адреса і зворотний зв’язок. Результати видачі Google більшою мірою залежать від даної інформації, оскільки 65% користувачів бажають бачити контактну інформацію.

Частина представлених систем не мають адаптивної розмітки, тобто сайти з телефонів зображуються некоректно. Згідно зі статистикою, на мобільні пристрої доводиться 5,135 млрд людей – на 4% більше, ніж у 2017-му році, і це число зростає. Адаптивність – одне з важливих і обов'язкових умов для сучасного сайту.

Якщо сайт зображується правильно тільки з комп'ютера, то при перегляді з планшета або телефону на ньому важко орієнтуватися, блоки не масштабуються, важко потрапити пальцем на одне з посилань – в результаті користувач не може зробити цільову дію, втрачає час на спроби розібратися в незрозумілому інтерфейсі системи і йде з ресурсу.

# 1.3 Аналіз функціональних особливостей

Представлені системи були розроблені з допомогою мови програмування JavaScript, яка є незамінною при веброзробці. JavaScript – це мова програмування, що використовується як на стороні клієнта, так і на стороні сервера, що дозволяє робити вебсторінки інтерактивними. Дана мова програмування використовується для створення мобільних та веб додатків, вебсерверів та розробки серверних додатків, розробки ігор та іншого [10].

Перевагами JS являється швидкість для кінцевого користувача, тобто сценарії, що були написані на стороні клієнта не потребують підтримки вебсервера, також відсутність потреби компіляції на стороні клієнта надає JavaScript’у певні переваги у швидкості. Універсальність є ще однією з переваг, JS взаємодіє з іншими мовами й може використовуватися у величезному різноманітті додатків.

Недоліком JS являється постійна компіляція коду при відкритті або оновлені сторінки швидкість відгуку сторінки значно падає, навіть односекундна затримка може скоротити конверсію на 7%. Також JavaScript іноді інтерпретується по-різному різними браузерами. Різні механізми компонування можуть показувати JavaScript по-різному, що призводить до неузгодженості з точки зору функціональності та інтерфейсу .

Деякі з проаналізованих інформаційних систем використовують СУБД MySQL, яка є проста в використанні, відрізняться хорошою швидкістю роботи і має широкий функціонал. До того ж MySQL в рівній мірі легко може бути використана для роботи й з малими, і з великими об'ємами даних. Бази даних MySQL є реляційними, тобто зберігають дані в окремих таблицях, а не в одному великому сховищі. Структури баз даних організовані у фізичні файли та оптимізовані для швидкості. Програмне забезпечення MySQL є відкритим, тобто кожен може використовувати та модифікувати програмне забезпечення відповідно до своїх потреб. Сервер баз даних MySQL дуже швидкий, надійний, масштабований і простий у використанні. Програмне забезпечення баз даних MySQL – це система клієнт / сервер, яка складається з багатопотокового SQL-сервера, що підтримує різні фони, декілька різних клієнтських програм та бібліотек, адміністративні засоби та широкий спектр інтерфейсів прикладного програмування (API). Недоліком системи являється недостатня надійність у деяких процесах по роботі з даними (наприклад, зв'язок, транзакції, аудит) [11].

Всі з представлених систем використовують скриптову, інтерпретовану мову програмування PHP. Дана мова проста в освоєнні й дозволяє створювати якісні web-додатки за короткі терміни. Отриманий продукт легко модифікується і підтримується в майбутньому. Перевагами даної мови перед іншими є:

1. Простий і інтуїтивно зрозумілий синтаксис. Код PHP легкий до читання незалежно від способу використання.
2. Кросплатформеність і гнучкість. PHP сумісний з усіма популярними платформами та написані даною мовою додатки успішно працюють на різному серверному ПО.
3. Відмінна масштабованість. PHP дозволяє домогтися максимальної продуктивності додатків, написаних на ньому, зі зростанням апаратних ресурсів. Вебдодатки, розподілені на кілька серверів, здатні справлятися з істотними навантаженнями (великим трафіком).
4. Вбудованість в HTML-документи. На просту HTML-сторінку можна легко додати, змінюваний контент шляхом вставки блоків коду PHP. Вони додаються подібно HTML-тегами, не порушують структуру документа.
5. Активний розвиток і вдосконалення. Спільнота розробників постійно працює над впровадженням додаткового функціонала, що розширює можливості мови, спрощенням синтаксису і поліпшенням захисту від можливих атак.

До недоліків даної мови програмування можна віднести:

1. Відсутність спеціальних бібліотек для сучасних потреб. Наприклад, машинне навчання в наш час є гарячою тенденцією, і воно точно збереже свою популярність найближчим часом. Попри те, що PHP має набір бібліотек, він не може конкурувати з Python у розробці вебдодатків, що забезпечують машинне навчання. Зараз PHP не може запропонувати однаково швидкі та ефективні альтернативи Python's TensorFlow, Scikit-learn, Theano та Keras. Таким чином, якщо додаток вимагає функціональних можливостей ML або може знадобитися в майбутньому, коли бізнес почне масштабуватися, PHP не найкращий вибір.
2. Проблеми з безпекою. З роками занепокоєння щодо безпеки продуктів на основі PHP все ще зберігається з кількох причин. Одним з них є відкритий код PHP, що означає, що можливі вразливості коду стають загальновідомими після їх знаходження.
3. Деякі суперечності синтаксису. Оскільки над розвитком PHP працюють програмісти-ентузіасти, деякі їх дії виявилися неузгодженими. Код містить елементи, характерні для різних мов, а також різні найменування вбудованих функцій.

## **1.4 Основні функції інформаційних систем кав’ярні**

Після проведеного аналізу наявних інформаційних систем кав’ярень, було виділено основні функції, які повинна забезпечувати система, що розробляється:

1. Реєстрація та авторизація в системі користувачем.
2. Зміна особистих даних користувачем в системі.
3. Перегляд, ознайомлення та купівля товарів.
4. Зміна кількості товару та його видалення в кошику.
5. Зворотний зв’язок користувача з адміністрацією.
6. Розміщення, редагування та видалення товару або групи товарів.

Наступним кроком було визначено основні сценарії роботи інформаційних систем кав’ярень:

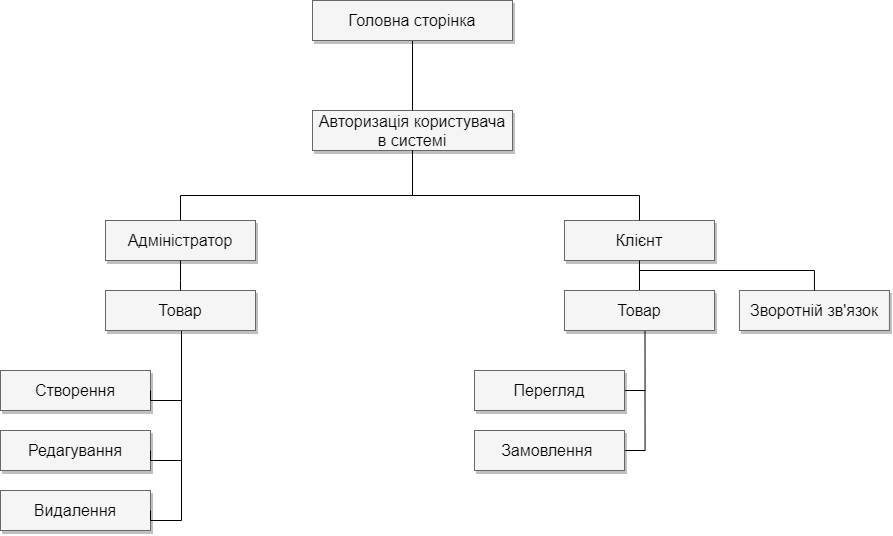
1. Розміщення товару в системі – адміністратор заходить на ресурс під своїм обліковим записом. Переходить на відповідну вкладку та додає новий товар, заповнюючи необхідні дані щодо назви, ціни, категорії опису та зображення товару.
2. Редагування товару в системі – адміністратор заходить на ресурс під своїм обліковим записом. Обирає товар, дані якого необхідно змінити. Змінює необхідні поля та зберігає зміни.
3. Видалення товару з системи – адміністратор заходить на ресурс під своїм обліковим записом. Обирає товар, який необхідно видалити та видаляє його. Система запитує адміністратора про впевненість в даному виборі.
4. Купівля товару – клієнт заходить на ресурс під своїм обліковим записом. Обирає товар, додає до кошика. Змінює кількість товару в кошику, або видаляє даний товар та обирає новий та оплачує покупку.
5. Зворотний зв’язок – клієнт залишає на відповідній вкладці системи свої дані для зв’язку та повідомлення-побажання щодо функціонування системи, наявних або придбаних товарів та ін.

Головними користувачами системи є клієнт та адміністратор. Адміністратор підтримує функціонування системи, а клієнт – користувач, що користується наявними функціями в системі. Кожен з даних користувачів має різний рівень доступу до тих чи інших функцій. Користувач не має змоги керувати товаром (редагування, видалення, додання), а адміністратор своєю чергою не має змоги купувати товар під обліковим записом клієнта системи.

# РОЗДІЛ 2 ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ КАВ’ЯРНІ

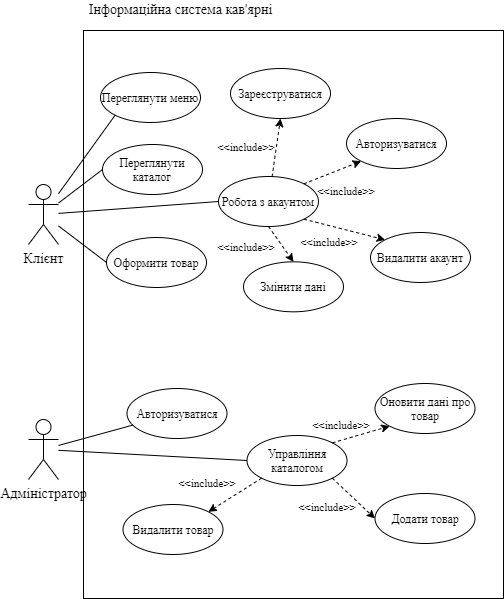
# 2.1 Структурна та функціональна частини системи

Для більш детального представлення системи, що розробляється було створено структурну частину, для визначення головних користувачів та їх можливих функцій в системі (рис. 2.1).

****Рисунок 2.1 – Структурна частина інформаційної системи

Зі структурної частини видно, що головними користувачами системи є клієнт та адміністратор. Кожен з них має певний набір функцій в системі. Для більш детального перегляду функцій було побудовано варіантів використання (рис. 2.2).

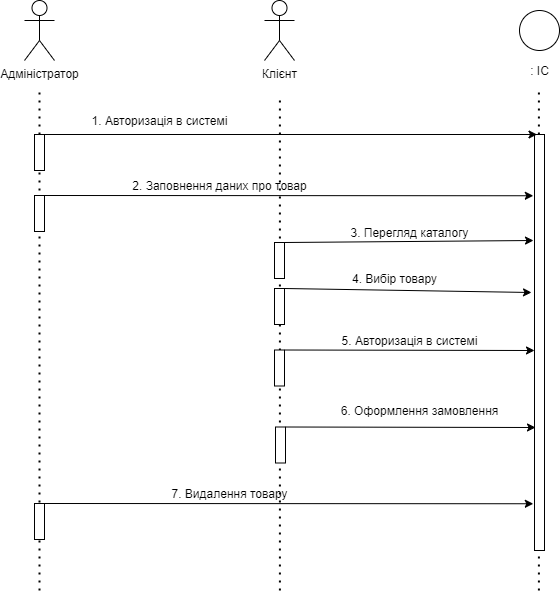
Дана діаграма представляє проєктовану систему у вигляді безлічі сутностей або акторів, які взаємодіють з системою за допомогою так званих варіантів використання. Актором (actor) може бути будь-хто, у кого є зв'язок з системою. Дійовою особою може бути пристрій, особа або інша система, яка впливає на систему що моделюється. Варіант використання (use case) описує сервіси, які були надані актору системою. Кожен варіант використання характеризує певний набір дій для взаємодії з системою при діалозі з актором. При цьому яким чином буде реалізована взаємодія акторів з системою невідомо.

  
Рисунок 2.2 – Use case діаграма для системи кав’ярні

З діаграми вже більше детально видно, які функції в системі має кожен користувач. Клієнт має змогу зареєструватися в системі, змінити дані у випадку необхідності та видалити обліковий запис. Також клієнт має змогу переглядати наявні товари в кав’ярні та покупати їх. Клієнт може переглядати меню кав’ярні, щоб знати в яку суму вийде середній чек при відвідуванні закладу.

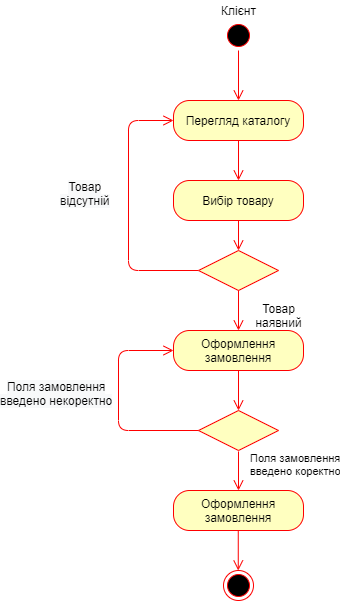
Адміністратор системи також перед роботою повинен авторизуватися. В системі даний актор може створювати нові товари, змінювати інформацію про них та видаляти товари з системи у разі їх неактуальності. Також даний актор може переглядати зареєстрованих користувачів системи та видаляти їх.

Наступним кроком було створено діаграму послідовності, яка описує відношення користувачів системи в різних умовах, які, своєю чергою, задаються з допомогою сценарію, отриманого при розробці діаграми варіантів використання. Діаграма показує повідомлення між учасниками та об’єктами системи, а також порядок обміну повідомленнями (рис. 2.3).

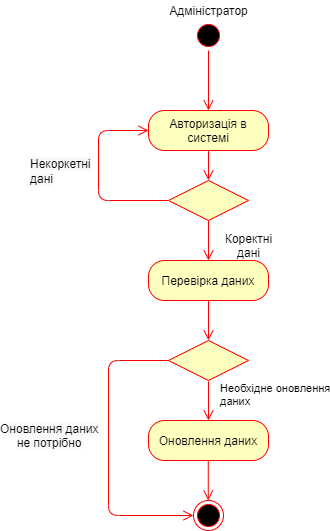
  
Рисунок 2.3 – Діаграма послідовності для системи кав’ярні

З діаграми видно, що при взаємодії з системою адміністратору необхідно авторизуватися. Після успішної авторизації даний користувач додає товар та інформацію про нього, щоб наступний користувач системи (клієнт) міг переглянути каталог з товаром та обрати той, який йому до вподоби. Після вибору товару клієнту необхідно авторизуватися, після чого відправити товар до кошика й при коректному заповнені даних товар переходить в стан оформлення. Коли товар було оформлено адміністратору необхідно видалити даний продукт з системи

Наступним кроком було розроблено діаграму станів для користувачів системи (рис.2.4 – 2.5). Дана діаграма зображає перехід об’єкту з одного стану в інший і допомагає моделювати динамічні аспекти системи. Від інших діаграм відрізняється описом процесу зміни станів тільки одного примірника певного класу.

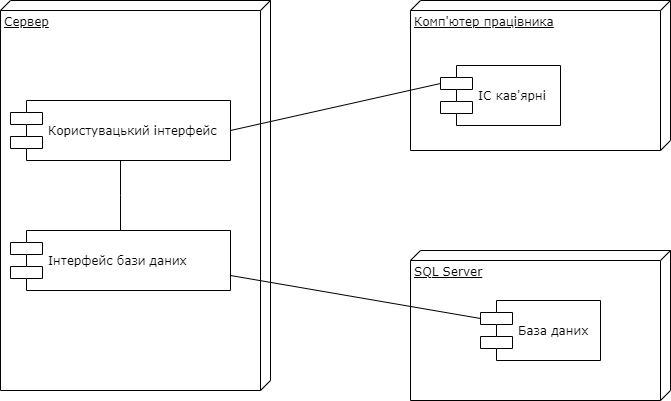
  
Рисунок 2.4 – Діаграма станів для клієнта

Клієнт починає свою роботу з системою відразу з відвідування головної сторінки системи. Наступним кроком клієнт переглядає каталог з товарами та обирає товар, що йому сподобався, додає його до кошика, якщо товар наявний в системі, то клієнт переходить до оформлення замовлення, якщо ні клієнт переходить знову до перегляду товарів. На етапі оформлення замовлення клієнт заповнює необхідні поля з даними і якщо все було введено правильно замовлення переходить в стан оформленого.

  
Рисунок 2.5 – Діаграма станів для адміністратора

Адміністратор починає роботу в системі з авторизації. При коректному введені даних адміністратор переходить до перевірки даних в системі, якщо ні – введення даних коректно. Якщо дані не потребують оновлення адміністратор виходить з системи, якщо ж потребують адміністратор оновлює дані системи та зберігає зміни.

Останнім кроком було розроблено діаграму розгортання (рис. 2.6). Дана діаграма допомагає уявити фізичне розташування системи, показуючи, на якому фізичному обладнанні запускається та чи інша складова програмного забезпечення.

  
Рисунок 2.6 – Діаграма розгортання

Користувач взаємодіє із системою шляхом її використання. Система відправляє запити та отримує відповіді від сервера. Сервер надає користувацький інтерфейс, а також – використовує інтерфейс бази даних для доступу до даних, їх зміни, видалення, модифікації.

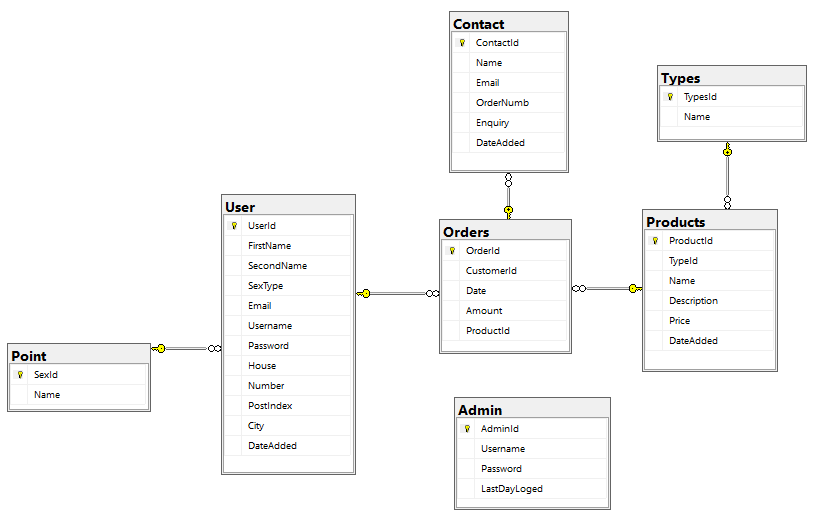
# 2.2 Проєктування бази даних

Проєктування бази даних є важливою частиною для коректної роботи з інформацією в системі. Для реалізації зберігання та оброблення даних було обрано SQL Server – систему управління реляційними базами даних. Використання даної СУБД надає ряд переваг, а саме:

1. СУБД має властивість до масштабування внаслідок чого надається можливість працювати з нею на портативних ПК або на мультипроцесорній техніці. Процесор може одночасно обробляти великий обсяг запитів.
2. Розмір сторінок – до 8 кб, тому дані отримуються швидко, детальну і складну інформацію зберігати зручніше.
3. Частина адміністративних завдань була автоматизована: управління пам'яттю, блокуваннями, редагування розмірів файлу.
4. Реалізація пошуку по тексту, фразах, а також реалізовано створення ключових індексів.
5. Інтегрований сервер інтерактивного аналізу в сервісі для процесу прийняття рішень, створення корпоративних звітів.
6. Формулювання запитів англійською мовою.
7. Підтримка роботи з іншими пакетами Microsoft Office: Access, MS Excel [12].

Здійснимо проектування бази даних (рис. 2.7). Визначимо структуру бази та основні сутності ІС:

1. Адміністратор.
2. Продукти.
3. Тип продуктів.
4. Замовлення.
5. Користувач.
6. Стать користувача.
7. Зворотний зв’язок.

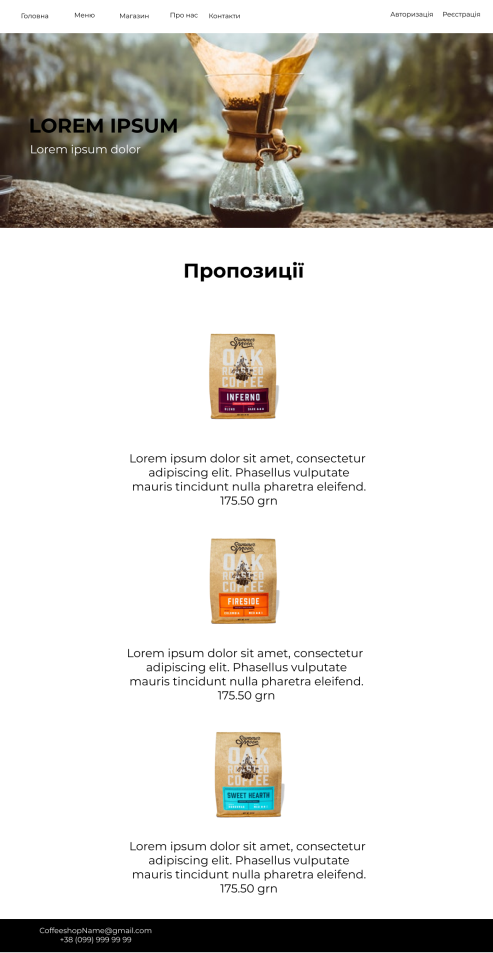
  
Рисунок 2.7 – Структура БД

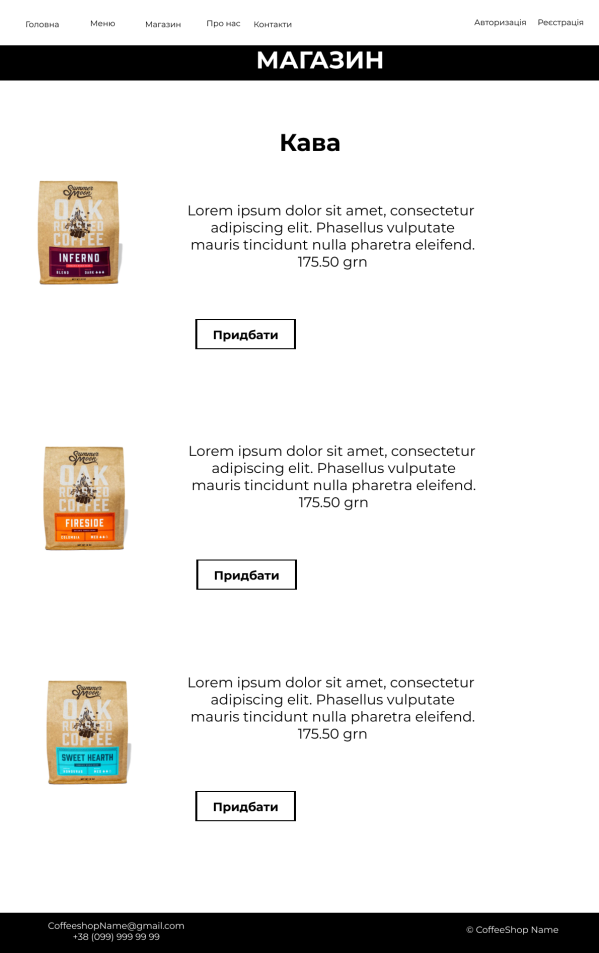
# 2.3 Розробка дизайну інтерфейсу користувача

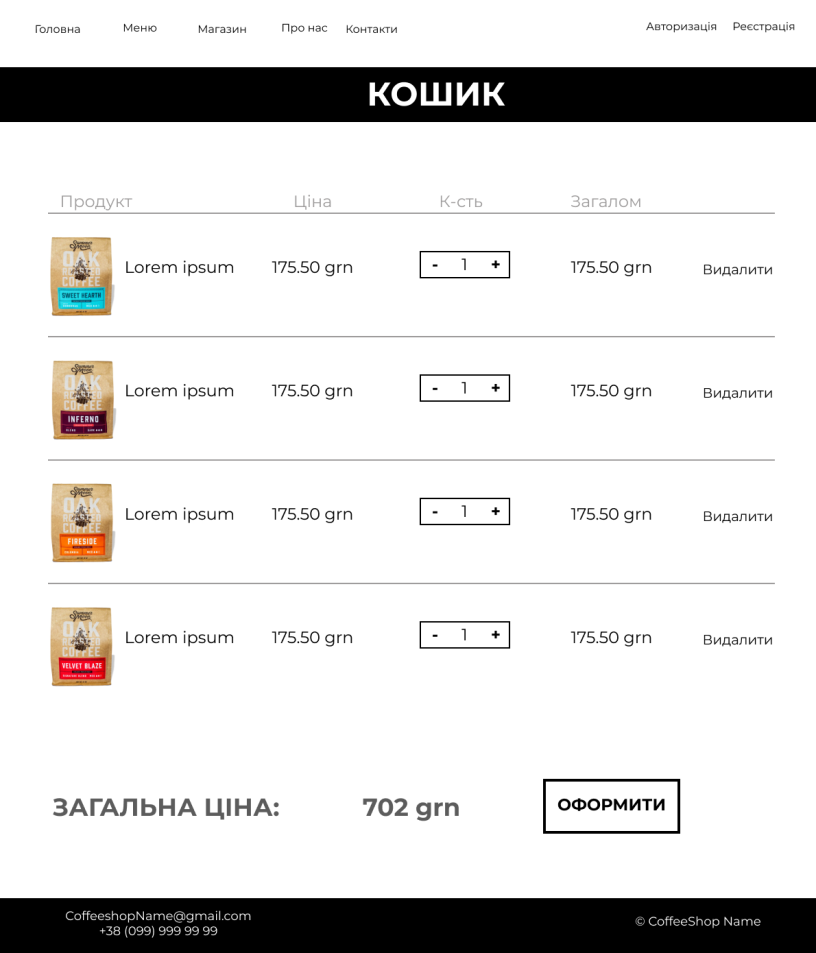
Для попереднього представлення інтерфейсу користувача було створено декілька макетів сторінок програмних модулів з допомогою середовища проєктування Figma.

Даний онлайн-сервіс є безплатним і дозволяє відразу починати роботу з шаблонів практично будь-якого портативного пристрою. Дане середовище також має інтегровані в себе елементи комунікації дизайнера з програмістом у спеціально відведеній частині коментарів.

Дані макети представлено нижче (рис. 2.8 – 2.12).

  
Рисунок 2.8 – Макет головної сторінки системи

  
Рисунок 2.9 – Макет сторінки з каталогом

  
Рисунок 2.10 – Макет сторінки з кошиком

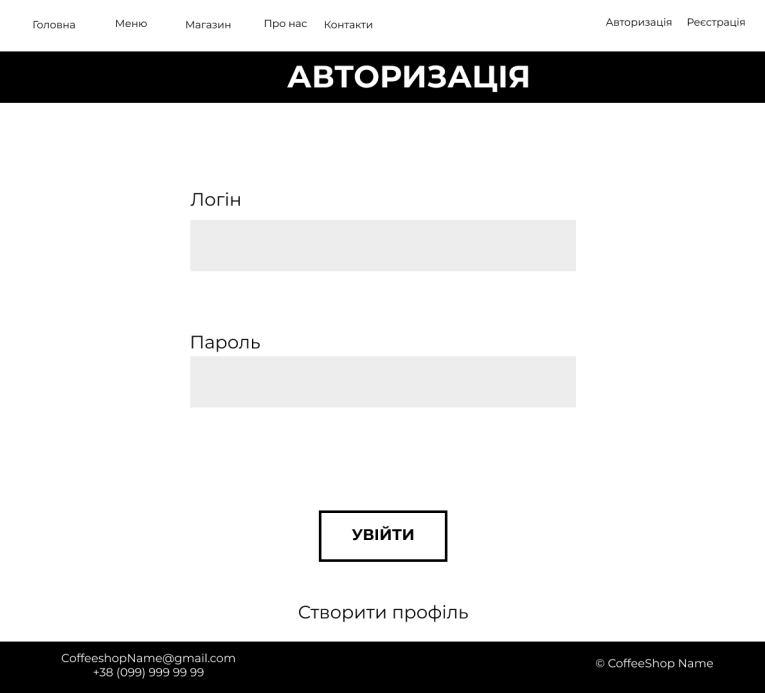
  
Рисунок 2.11 – Макет сторінки авторизації в системі

  
Рисунок 2.12 – Макет сторінки реєстрації в системі

Дані макети зображують дизайн для клієнтського модуля інформаційної системи кав’ярні, адміністративний модуль має схожий з клієнтським дизайн.

# РОЗДІЛ 3 РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ КАВ’ЯРНІ

# 3.1 Вибір платформи

Для розроблення ІС кав’ярні було обрано таку широко використовувану відкриту кодову мову загального призначення як PHP. Дана мова програмування легка у вивчені та має велику спільноту розробників та достатню кількість навчальних посібників. Тому при виникненні запитань можна завжди знайти швидко відповідь.

Перевагою використання PHP являється взаємодія мови з багатьма різними базами даних. Найпоширенішим вибором є MySQL, головним чином тому, що він безплатний, ефективний і популярний серед розробників. Іншими надійними варіантами систем управління базами даних, сумісними з PHP, є mSQL, MS-SQL, SQLite, PostgreSQL та ін. Крім того, PHP можна однаково добре використовувати з ElasticSearch, Redis, MongoDB та іншими нереляційними базами даних.

Найбільш поширеним сценарієм роботи вебсайту PHP є стек LAMP. Це означає, що вебсайт працює на вебсервері Apache HTTP, розгорнутому в системі Linux, і використовує MySQL як базу даних. Всі ці компоненти безплатні, а стек добре перевірений, що передбачає скорочення необхідного часу та коштів на розробку.

PHP пропонує вбудоване програмування HTML. У більшості випадків PHP-скрипт не заважає HTML-коду вебсторінки, а натомість завершує його, залишаючись всередині меж, визначених тегами <?Php?>. І навпаки, якщо сторінка була створена повністю на PHP, її завжди можна інтегрувати в HTML-код за допомогою сценарію.

Гнучкість дозволяє PHP ефективно поєднувати з багатьма іншими мовами програмування, щоб програмний продукт міг використовувати найбільш ефективну технологію для кожної конкретної функції. Також дана мова є міжплатформовою, що дозволяє використовувати будь-яку основну операційну систему – Windows, Linux, macOS – для виконання кодування.

Для збереження даних в системі було обрано MySQL. По суті, це програмне забезпечення, яке дозволяє ефективно зберігати, змінювати та керувати реляційними даними. Поряд з Oracle, Microsoft SQL Server, PostgreSQL та DB2, MySQL є однією з найпопулярніших систем управління реляційними базами даних на ринку. Однією з найважливіших функцій MySQL є надання даних іншим програмним додаткам.

Дана СУБД була обрана тому що на відміну від іншого програмного забезпечення для управління базами даних, інсталяція та конфігурація MySQL простіші. Для встановлення програмного забезпечення не потрібно мати певного набору інструментів, а оновлення відбувається повністю автоматично.

Однією з основних цілей MySQL є забезпечення безпеки бази даних за допомогою служби адміністрування баз даних MySQL. Це програмне забезпечення дозволяє працювати зі структурою таблиці, яка з'єднує функції та елементи даних, що допомагає захистити дані. Для баз даних, які містять інформацію про клієнта та інші конфіденційні дані, безпека та цілісність є обов’язковою умовою.

У разі переривання живлення або відключення сервера дані можуть пошкодитися, що створює велику проблему для підприємств, у яких майже не зберігається резервних копій. MySQL виключає ризик втрати даних завдяки функціям відновлення та відновлення даних.

Для роботи з базою даних було використано інтерфейс phpMyAdmin. Даний інтерфейс може управляти як цілим MySQL сервером (для цього необхідні права супер-користувача), так і окремою базою даних.

## **3.2 Розробка бази даних**

Визначимо основні сутності ІС:

Сутність “Користувач” містить інформацію про користувача системи та має наступні атрибути:

1. Id – ідентифікатор користувача в системі, має тип int(11), є первинним ключем, не може бути NULL.
2. Ім’я – ім’я користувача, має тип varchar(50), не може бути NULL.
3. Прізвище – прізвище користувача, має тип varchar(50), не може бути NULL.
4. Id статі – ідентифікатор статі користувача, має тип int(11), є вторинним ключем, не може бути NULL.
5. Еmail – електронна пошта користувача, має тип varchar(255), не може бути NULL.
6. Логін – логін користувача для входу в систему, має тип varchar(50), не може бути NULL.
7. Пароль – пароль користувача для входу в систему, має тип varchar(50), не може бути NULL.
8. Будинок – адреса користувача, має тип varchar(50), не може бути NULL.
9. Номер – номер будинку користувача, має тип int(11), не може бути NULL.
10. Поштовий індекс – поштовий індекс користувача, має тип int(11), не може бути NULL.
11. Місто – місто проживання користувача, має тип varchar(50), не може бути NULL.
12. Дата приєднання – дата приєднання користувача до системи, має тип date, не може бути NULL.

Сутність “Стать” має наступні атрибути:

1. Id – ідентифікатор статі користувача, має тип int(11), є первинним ключем, не може бути NULL.
2. Назва – назва статі користувача, має тип varchar(10), не може бути NULL.

Сутність “Замовлення” має наступні атрибути:

1. Id – ідентифікатор замовлення, має тип int(11), є первинним ключем, не може бути NULL.
2. Id користувача – ідентифікатор користувача, має тип int(11), є вторинним ключем, не може бути NULL.
3. Дата – дата оформлення замовлення, має тип date, не може бути NULL.
4. Кількість – кількість товару до оформлення, має тип int(11), не може бути NULL.
5. Id продукту – ідентифікатор продукту, має тип int(11), є вторинним ключем, не може бути NULL.

Сутність “Товар” має наступні атрибути:

1. Id – ідентифікатор товару, має тип int(11), є первинним ключем, не може бути NULL.
2. Id типу – ідентифікатор типу товару, має тип int(11), є вторинним ключем, не може бути NULL.
3. Назва – назва продукту, має тип varchar(255), не може бути NULL.
4. Опис – опис продукту, має тип mediumtext, не може бути NULL.
5. Ціна – ціна продукту, має тип float, не може бути NULL.
6. Дата додання – дата оформлення замовлення, має тип date, не може бути NULL.

Сутність “Типи” має наступні атрибути:

1. Id – ідентифікатор типу товару, має тип int(11), є первинним ключем, не може бути NULL.
2. Назва – назва типу товару, має тип varchar(50), не може бути NULL.

Сутність “Адміністратор” має наступні атрибути:

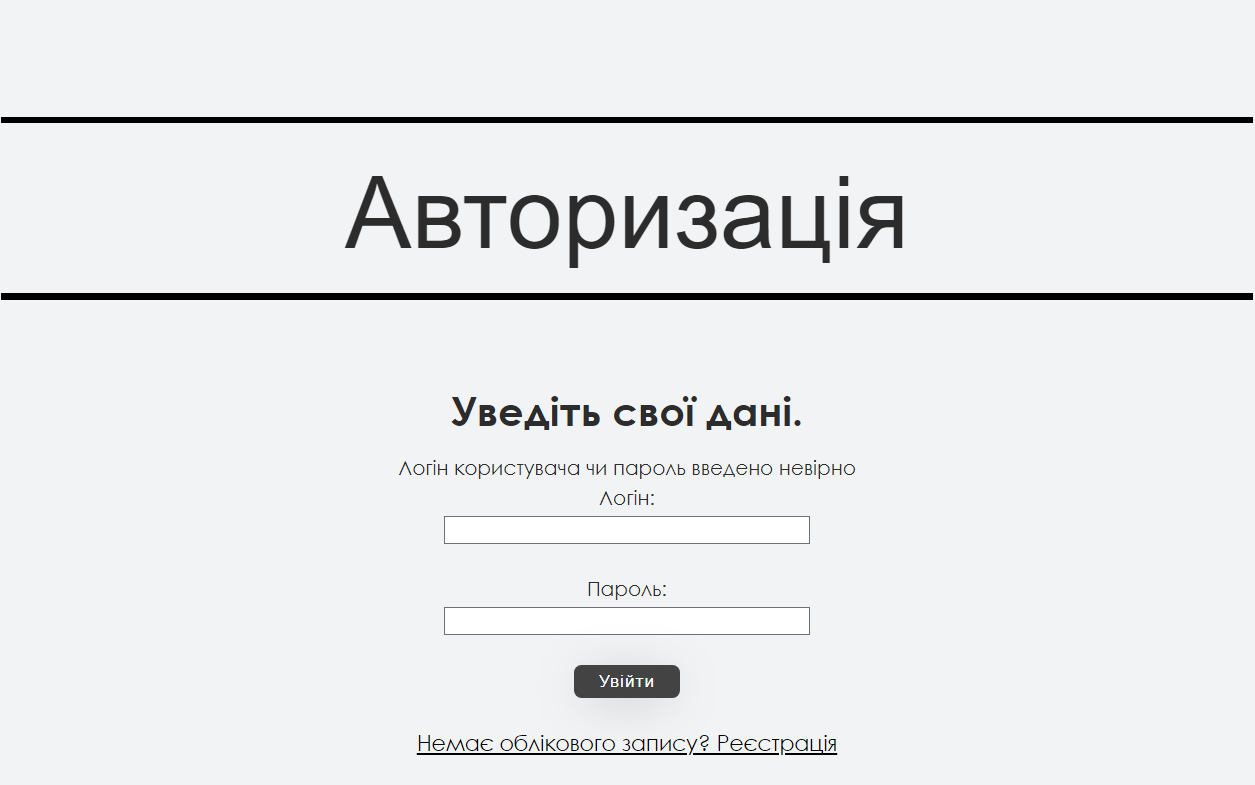
1. Id – ідентифікатор адміністратора, має тип int(11), є первинним ключем, не може бути NULL.
2. Логін – логін адміністратора для входу в систему, має тип varchar(50), не може бути NULL.
3. Пароль – пароль адміністратора для входу в систему, має тип varchar(50), не може бути NULL.
4. Дата входу – вказує на останню дату входу адміністратора в систему, тип date, не може бути NULL.

Сутність “Зворотний зв’язок” має наступні атрибути:

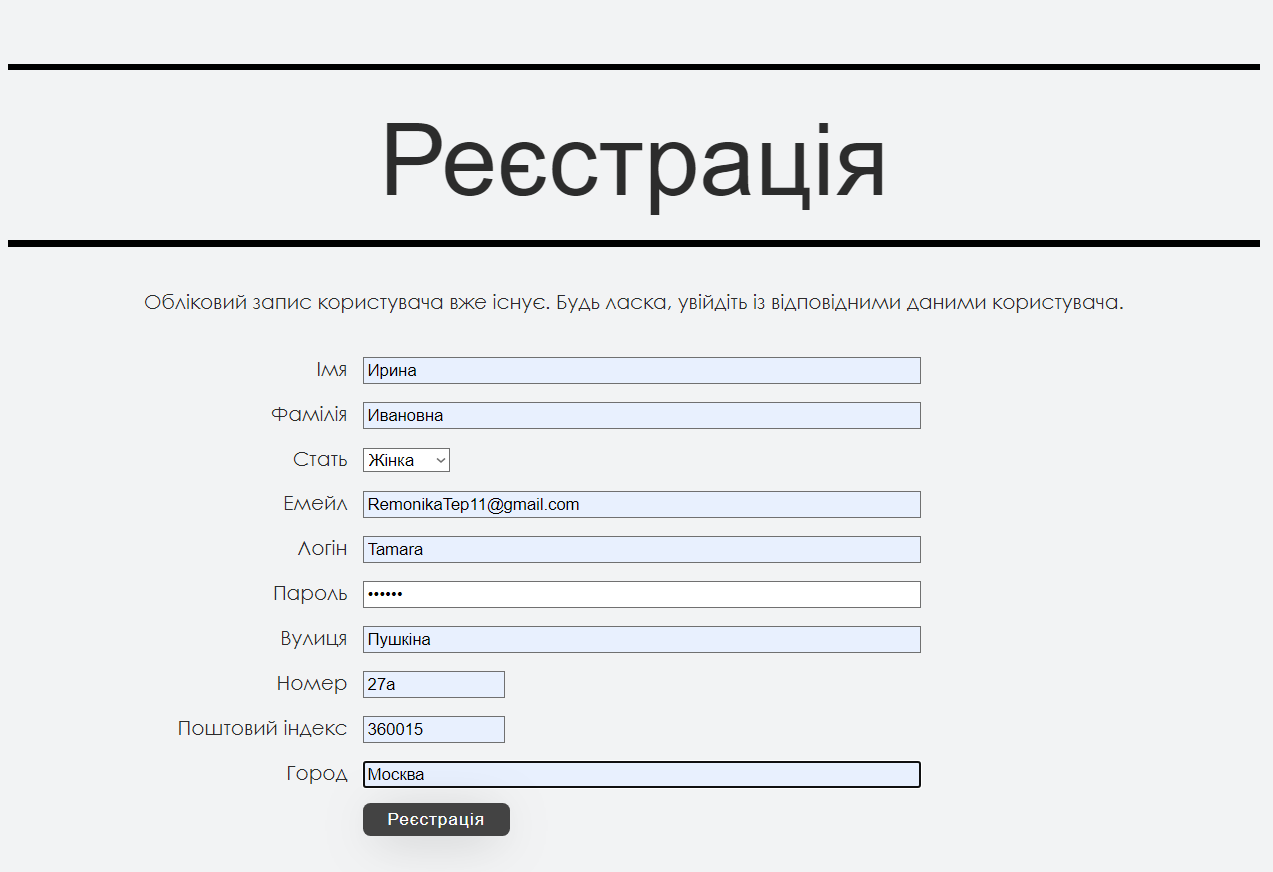
1. Id – ідентифікатор звернення, має тип int(11), є первинним ключем, не може бути NULL.
2. Ім’я – ім’я користувача, має тип varchar(50), не може бути NULL.
3. Еmail – електронна пошта користувача, має тип varchar(255), не може бути NULL.
4. Номер замовлення – ідентифікатор номеру замовлення, має тип int(11), є вторинним ключем, не може бути NULL.
5. Звернення – повідомлення користувача, має тип text, не може бути NULL.
6. Дата звернення – дата звернення з повідомленням користувачем, має тип date, не може бути NULL.

## **3.3 Розробка користувацького інтерфейсу**

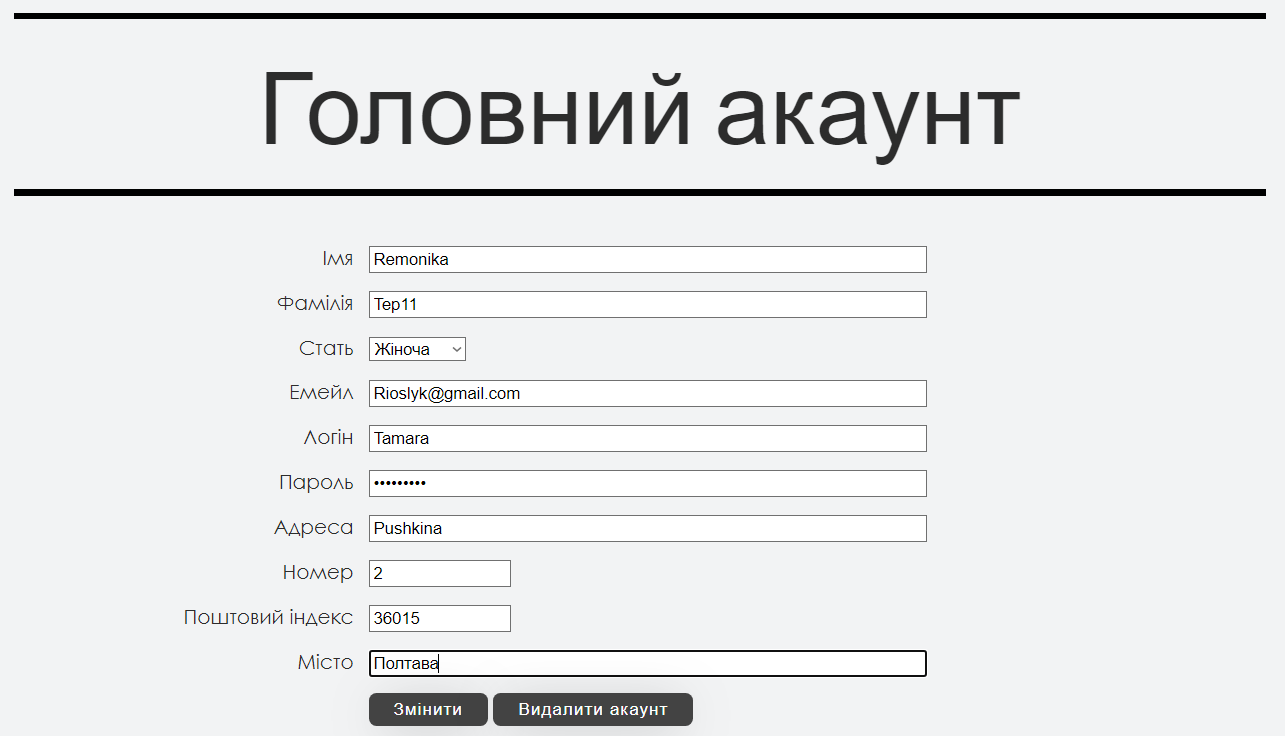
Для отримання доступу до всіх функцій в системі користувачу необхідно авторизуватися на відповідній вкладці меню (рис.3.1). При коректному введені полів, користувача перенесе на головну сторінку. В іншому випадку система запросить ввести дані коректно.

  
Рисунок 3.1 – Сторінка авторизації в системі

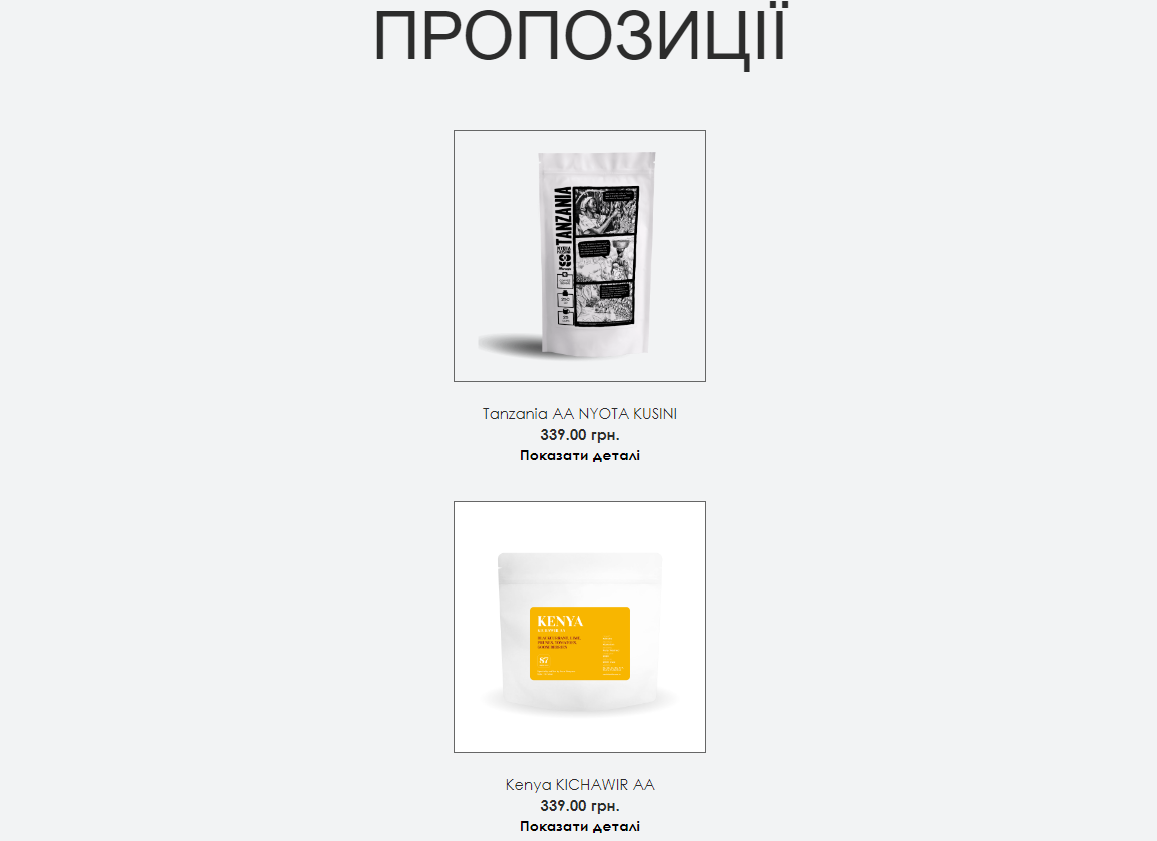
Якщо ж користувач не має облікового запису йому прийдеться зареєструватися на відповідній вкладці меню. Ввівши інформацію у всі поля користувач натискає “Реєстрація”. Якщо збіги з логіном в базі даних відсутні, то система реєструє користувача згідно поданих даних. Якщо ж базі даних наявний збіг по логіну система видає помилку і пропонує користувачу змінити логін або ж зайти під даним логіном в систему, якщо той йому належить (рис.3.2).

  
Рисунок 3.2 – Сторінка реєстрації в системі

Якщо користувач при реєстрації ввів хибні дані, наприклад невірне ім’я, фамілію, або ж дані застаріли, наприклад адреса, і хоче змінити їх, він завжди може перейти на вкладку профілю та змінити необхідні поля з даними (рис.3.3).

  
Рисунок 3.3 – Особиста сторінка з даними користувача

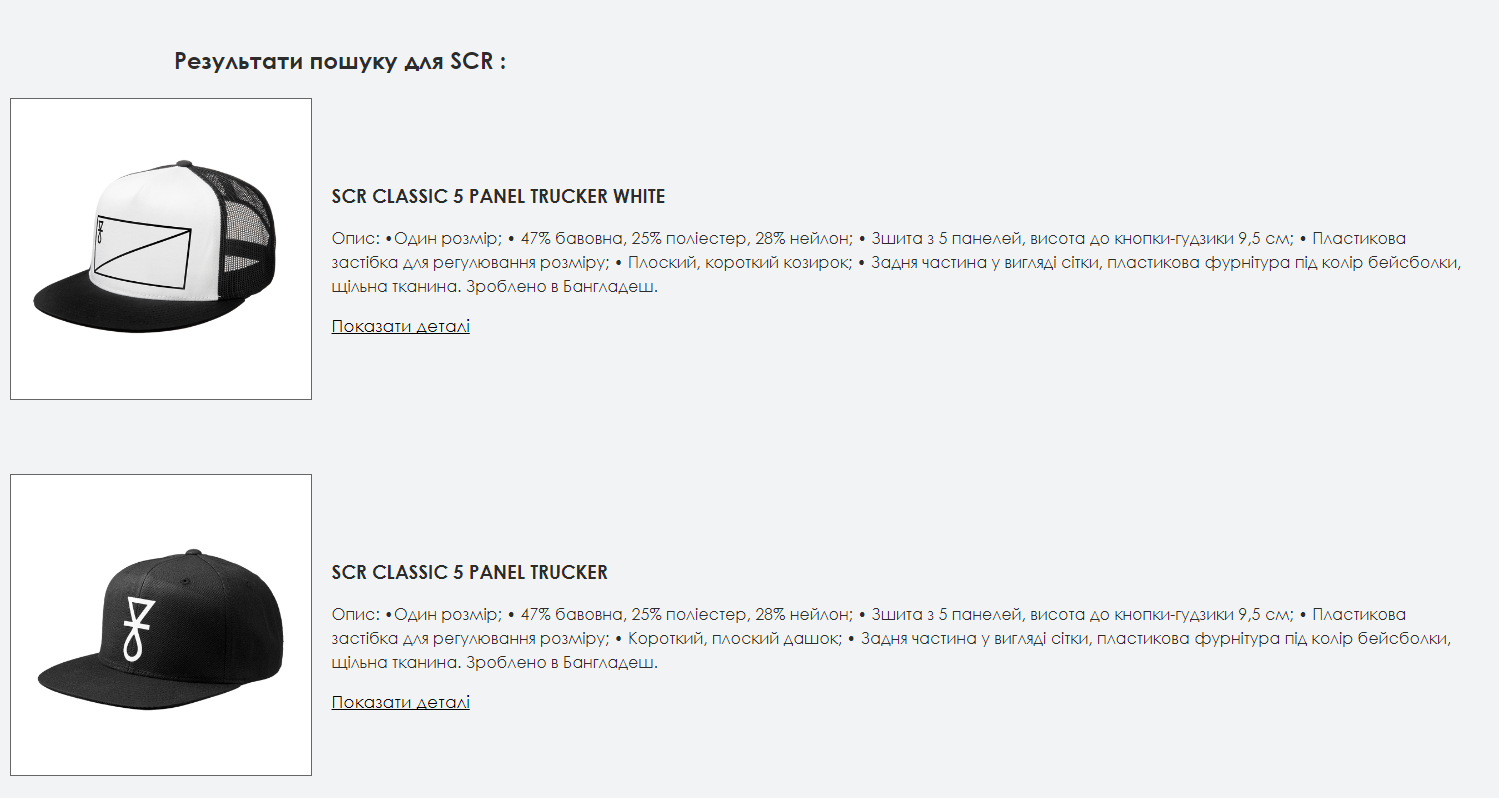
На головній сторінці користувачу надана інформація щодо актуальних товарів в системі (рис.3.4).

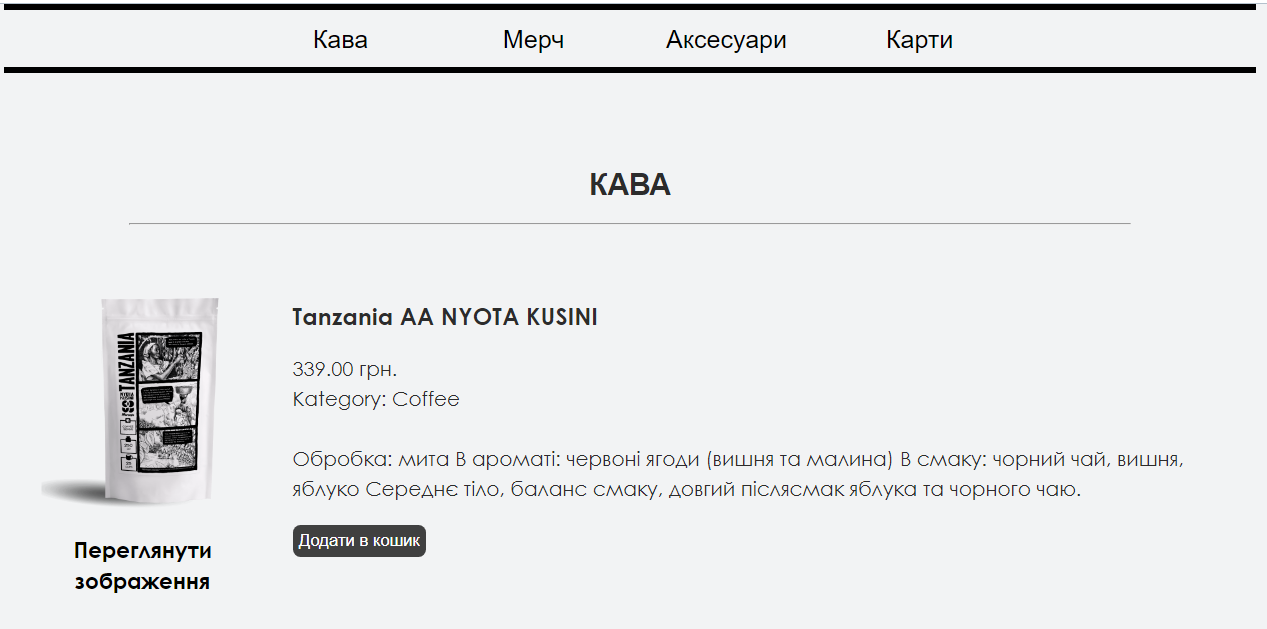
  
Рисунок 3.4 – Головна сторінка системи

Користувач може переглянути дані пропозиції та при необхідності перегланути їх, натиснувши показати деталі (рис.3.5). При перегляді деталей користувачу буде доступна інформація про товар (назва, ціна, опис, категорія).

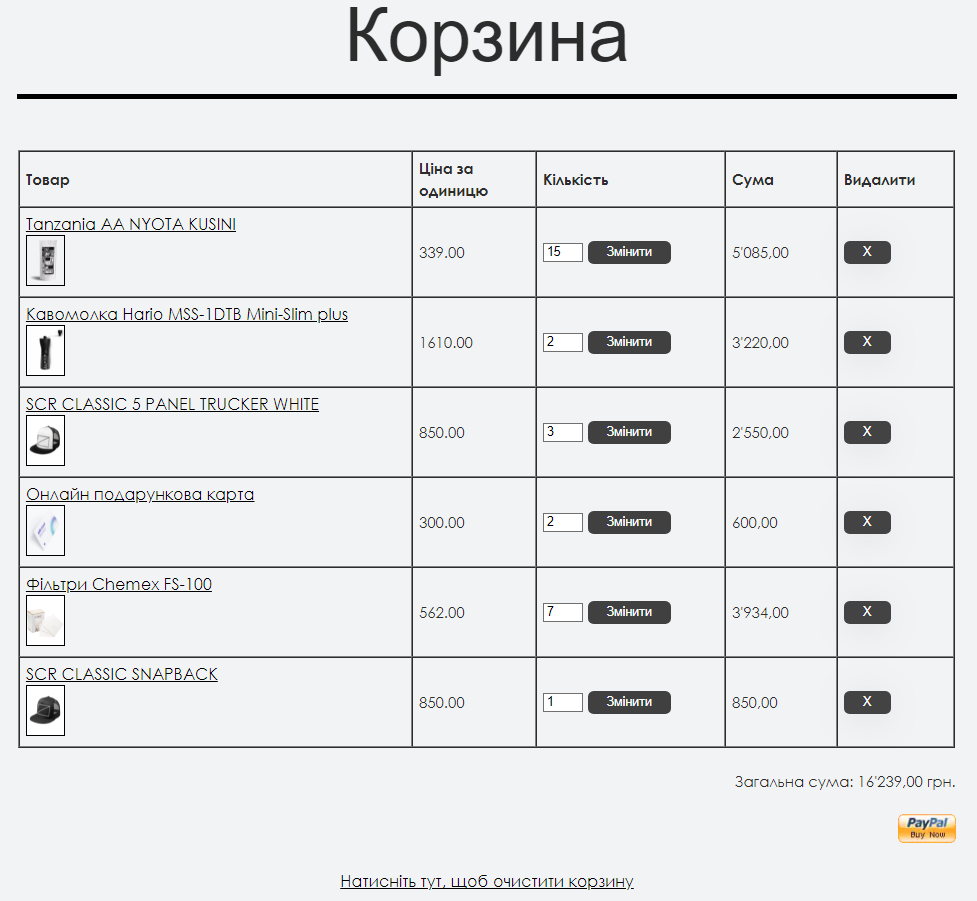
  
Рисунок 3.5 – Сторінка перегляду даних про товар

Користувач має змогу переглядати онлайн магазин кав’ярні та робити покупки. Для пошуку необхідного товару клієнт може скористуватися пошуком в системі (рис.3.6), або шукати необхідний товар по категорії (рис.3.7).

  
Рисунок 3.6 – Пошук товару через пошукову строку

  
Рисунок 3.7 – Пошук товару через категорії

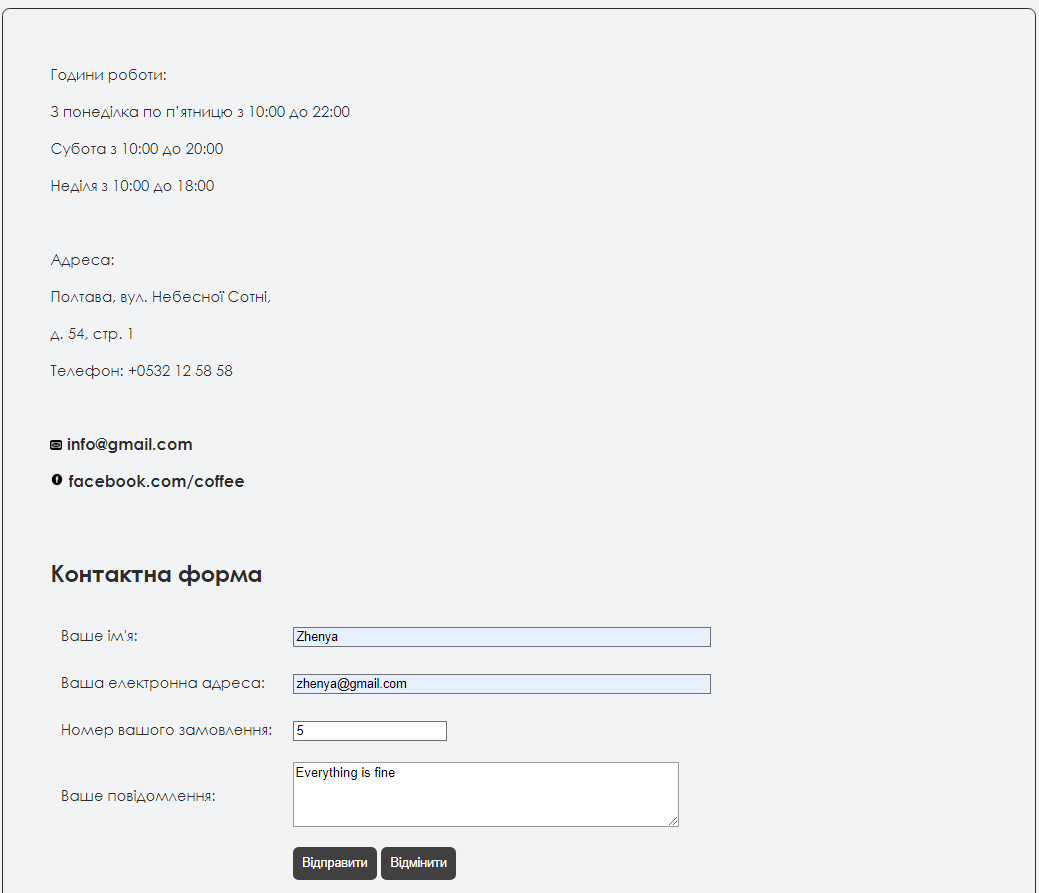
При доданні товару до кошика клієнт має змогу змінити кількість товару або видалити його, після чого придбати товар, що сподобався (рис.3.8).

  
Рисунок 3.8 – Кошик з покупками користувача

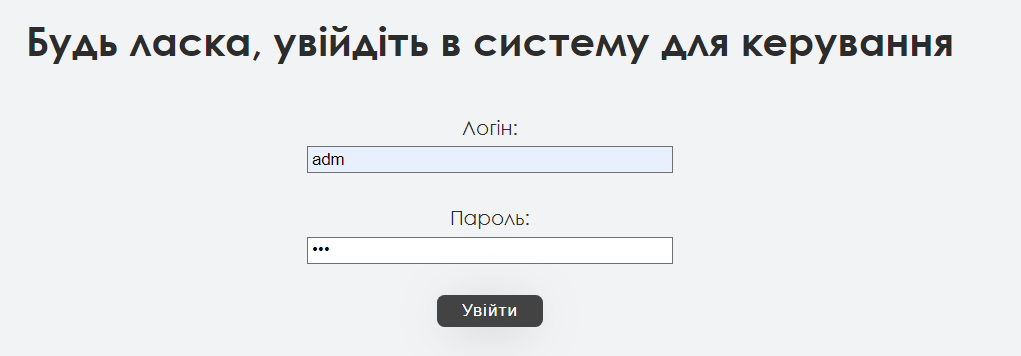
В інформаційній системі наявне меню закладу з актуальними продуктами та цінами, задля того щоб клієнт заздалегідь міг зорієнтуватися в цінах та в наявних продуктах в кав’ярні. Для перегляду меню клієнту необхідно обрати відповідну вкладку системи (рис.3.9).

  
Рисунок 3.9 – Сторінка з меню закладу

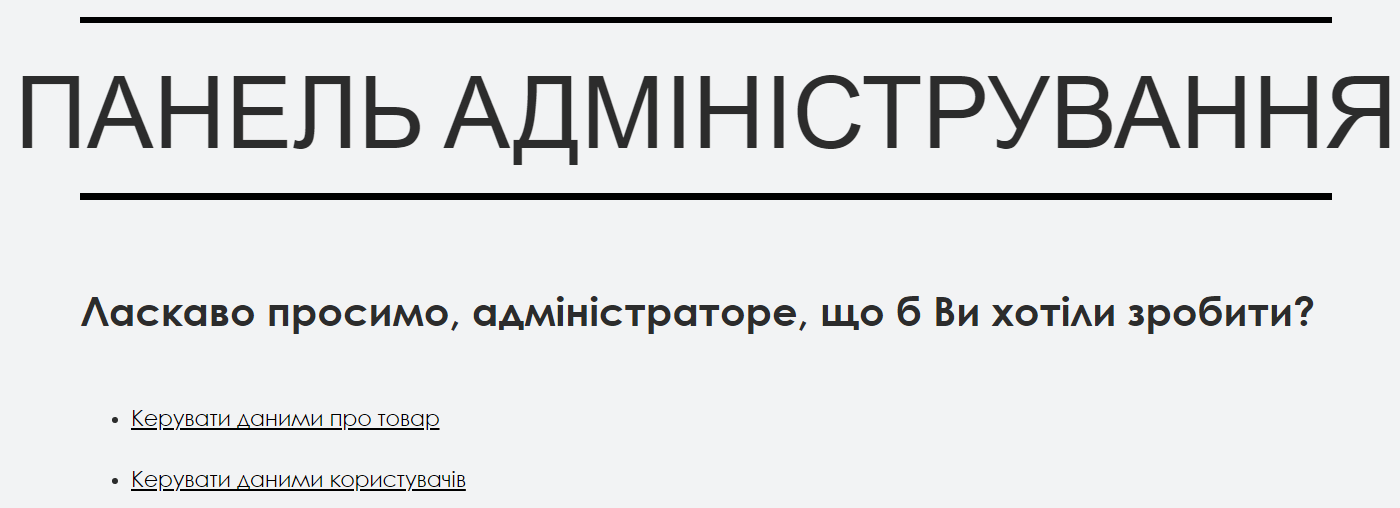
На вкладці з контактною формою клієнт має змогу дізнатися години роботи та адрес закладу. Також клієнт може залишити свої повідомлення або побажання щодо роботи системи, заповнивши необхідні поля даними (рис.3.10). Система надає змогу напряму зв’язатися з адміністратором ресурсу, завдяки наявної інформації щодо робочого телефону та пошти.

  
Рисунок 3.10 – Сторінка зворотного зв’язку

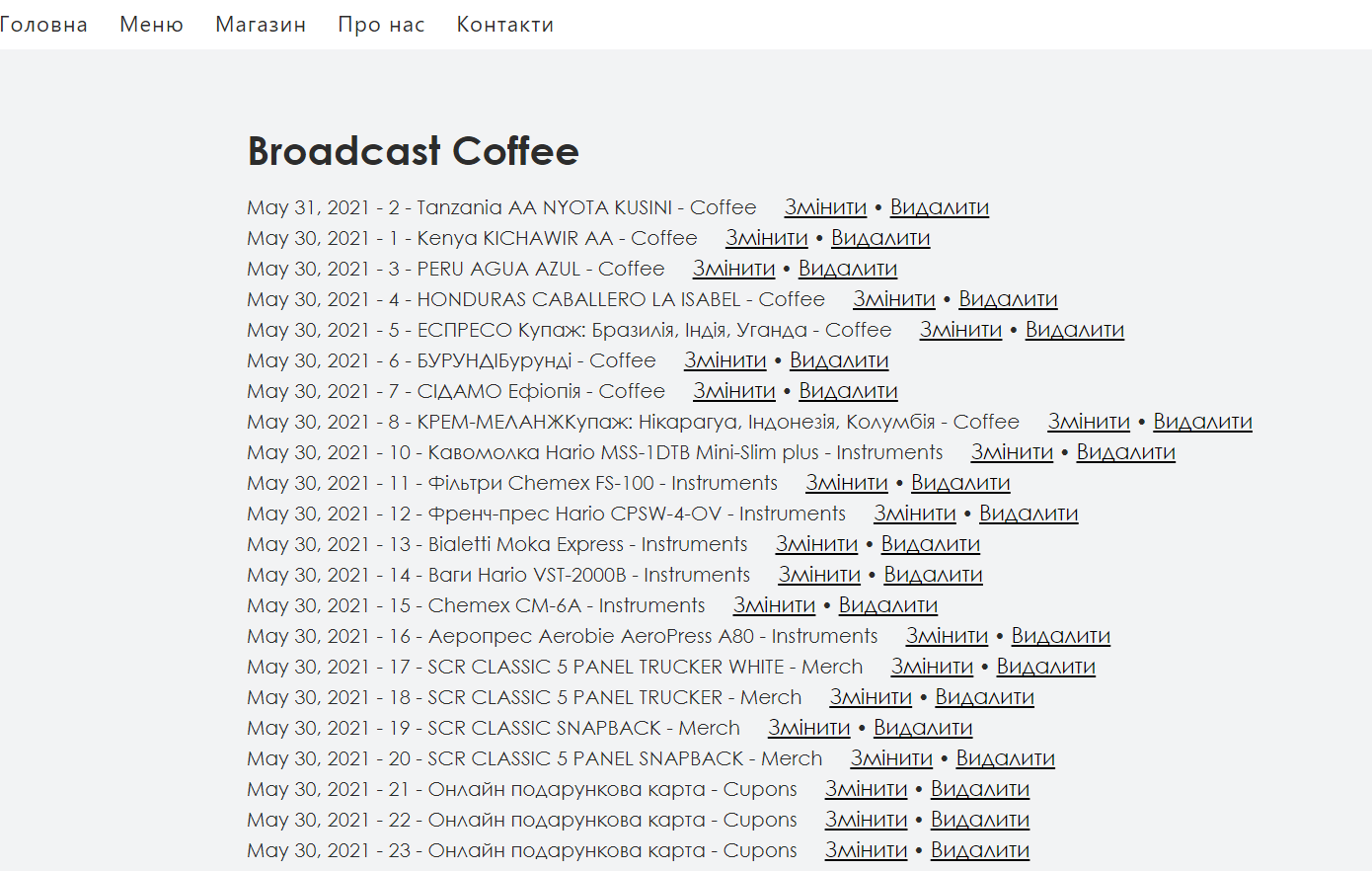
Для того, щоб почати роботу в системі у якості адміністратора також необхідно авторизуватися (рис.3.11).

  
Рисунок 3.11 – Сторінка авторизації адміністратора в системі

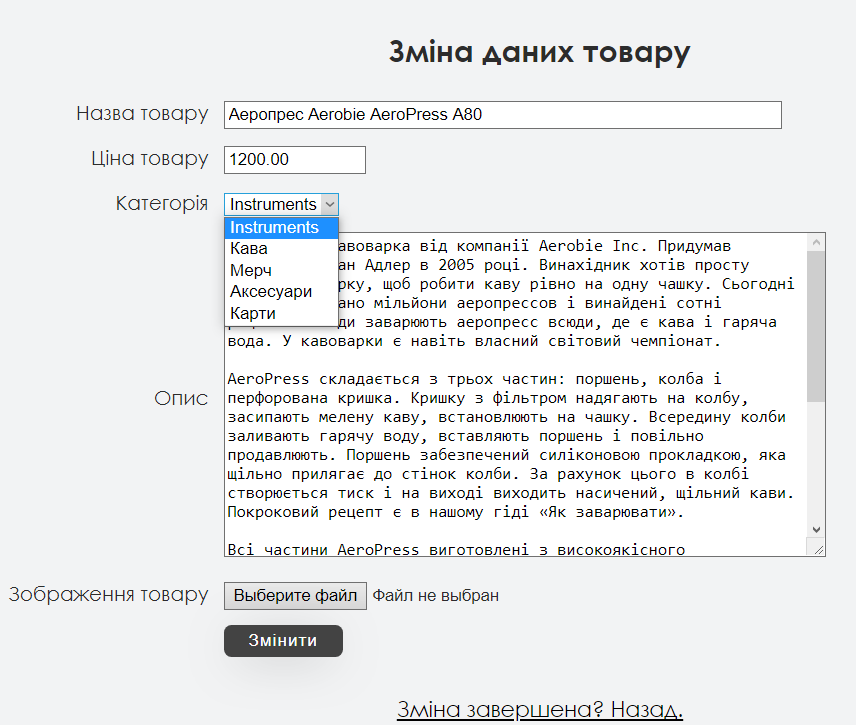
Після авторизації в системі адміністратору буде доступна панель адміністрування (рис.3.12).

  
Рисунок 3.12 – Сторінка з панеллю адміністрування

На вкладці керування даними про товар адміністратор має змогу переглянути всі наявні товари в системі (рис.3.13).

  
Рисунок 3.13 – Сторінка перегляду товарів з боку адміністратора

Адміністратор має змогу змінювати обраний товар, зберігати дані щодо зміни товару в системі або відміняти редагування товару (рис.3.14).

  
Рисунок 3.14 – Сторінка зі зміною даних товару.

Також на головній сторінці адміністрування користувач має змогу додавати новий товар до системи з відповідними назвою, ціною категорією, описом товару та його зображенням (рис.3.15-13.16).

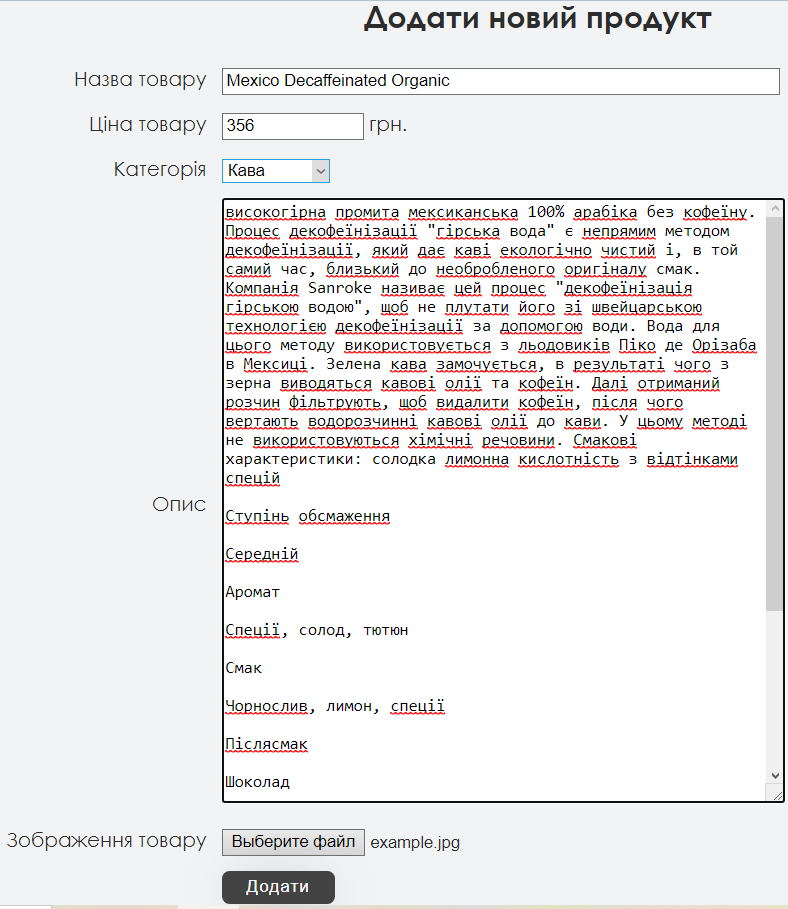
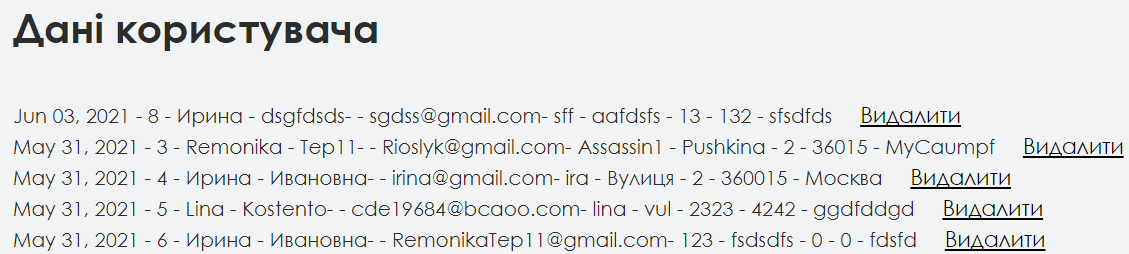
  
Рисунок 3.15 – Сторінка з доданням нового продукту

  
Рисунок 3.16 – Сторінка нового продукту в системі

Для видалення товару з системи адміністратори необхідно натиснути “Видалити” після чого система запитає чи дійсно користувач хоче видалити даний товар та при підтвердженні дії видалить його (рис.3.17).

  
Рисунок 3.17 – Сторінка з видаленням товару з системи

Адміністратору також доступна сторінка перегляду даних зареєстрованих користувачів системи (рис.3.18). Адміністратор може видалити користувача якщо той довго не появлявся в системі або з інших причин.

  
Рисунок 3.18 – Сторінка з даними користувачів

# РОЗДІЛ 4 ТЕСТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ КАВ’ЯРНІ

## **4.1 Вибір способу тестування**

Тестування програмного забезпечення є найважливішим компонентом розробки програмного продукту, оскільки воно покращує узгодженість та продуктивність. Основною перевагою тестування є виявлення та подальше усунення помилок.

Тестування – це процес оцінки функціональних можливостей програмного забезпечення з метою з'ясувати, чи відповідає розроблене програмне забезпечення зазначеним вимогам чи ні, та виявити дефекти, щоб забезпечити бездоганний виріб продукту.

Основними типами тестування є ручне та автоматизоване.

Ручне тестування – це процес тестування програмного забезпечення в ручну, щоб дізнатись більше про нього, знайти, що є, а що не працює. Зазвичай воно включає перевірку всіх функцій, зазначених у документах з вимогами, але часто також включає тестувальників, які пробують програмне забезпечення з урахуванням точки зору їхнього кінцевого користувача.

Плани ручного тестування варіюються від повністю написаних сценаріїв тестів, даючи тестерам детальні кроки та очікувані результати, до керівництв високого рівня, які спрямовують на пошукові сесії.

Автоматизоване тестування – це процес тестування програмного забезпечення за допомогою засобів автоматизації для виявлення дефектів.

У цьому процесі тестувальники виконують тестові сценарії та автоматично генерують результати тестування за допомогою засобів автоматизації.

Інформаційну систему кав’ярні було протестовано вручну.

## **4.2 Тест план**

Перед проведенням тестування було створено тест план в якому було описано основні функції системи, що тестується.

Функції, що повинні бути протестовані в адміністративному модулі:

1. Створення нового товару.
2. Редагування та видалення товарів.
3. Видалення користувачів.

Функції, що повинні бути протестовані в клієнтському модулі:

1. Реєстрація в системі.
2. Додавання товару до кошику.
3. Зміна кількості товару в кошику.
4. Видалення товару з кошику.

Розроблення тест-кейсів для відповідних функцій системи (табл. 4.1).

Таблиця 4.1 – Тест-кейси для адміністративного модуля

|  |  |
| --- | --- |
| **Опис:** | Додавання нового товару в систему. |
| **Умова:** | Виконано вхід в обліковий запис адміністратора. |
| Кроки: | Очікуваний результат: |
| 1. Відкрити сторінку з керуванням даними про товар. | Відображення наявних товарів в системі. |
| 2. Прогорнути до низу сторінки. | Відображення полів для додавання нового продукту. |
| 3. Ввести коректні дані. | Введено коректні дані. |
| 4. Натиснути кнопку “Додати”. | Відбувається перезавантаження сторінки з відображенням нового товару в списку. |
| **Опис:** | Редагування товару в системі. |
| **Умова:** | Виконано вхід в обліковий запис адміністратора. |
| Кроки: | Очікуваний результат: |
| 1. Відкрити сторінку з керуванням даними про товар. | Відображення наявних товарів в системі. |

Продовження таблиці 4.1

|  |  |
| --- | --- |
| 2. Натиснути кнопку “Змінити” будь-який товар. | Відбудеться переадресація на сторінку з даними товару. |
| 3. Ввести коректні дані. | Введено коректні дані. |
| 4. Натиснути кнопку “Змінити”. | Відбувається перезавантаження сторінки з відображенням товару та нової інформації про нього. |
| **Опис:** | Введення некоректних даних при додаванні нового товару в систему. |
| **Умова:** | Виконано вхід в обліковий запис адміністратора. |
| Кроки: | Очікуваний результат: |
| 1. Відкрити сторінку з керуванням даними про товар. | Відображення наявних товарів в системі. |
| 2. Прогорнути до низу сторінки. | Відображення полів для додавання нового продукту. |
| 3. Ввести некоректні дані. | Система повідомляє, що дані було введено некоректно. |
| **Опис:** | Видалення товару з системи. |
| **Умова:** | Виконано вхід в обліковий запис адміністратора. |
| Кроки: | Очікуваний результат: |
| 1. Відкрити сторінку з керуванням даними про товар. | Відображення наявних товарів в системі. |
| 2. Натиснути кнопку “Видалити” біля будь-якого товару. | Відбудеться переадресація на сторінку з підтвердженням вибору щодо видалення. |
| 3. Натиснути “Так”. | Відбудеться переадресація на сторінку з товарами системи з видаленим обраним товаром. |
| **Опис:** | Видалення користувача з системи. |
| **Умова:** | Виконано вхід в обліковий запис адміністратора. |
| Кроки: | Очікуваний результат: |

Продовження таблиці 4.1

|  |  |
| --- | --- |
| 1. Відкрити сторінку з керуванням даними про користувачів. | Відображення всіх користувачів системі |
| 2. Натиснути кнопку “Видалити” біля будь-якого користувача. | Відбудеться переадресація на сторінку з підтвердженням вибору щодо видалення. |
| 3. Натиснути “Так”. | Відбудеться переадресація на сторінку з користувачами системи з видаленим обраним користувачем. |

Розроблення тест-кейсів для перевірки функцій модуля клієнта.

Таблиця 4.2 – Тест-кейси для клієнтського модуля

|  |  |
| --- | --- |
| **Опис:** | Реєстрація користувача в системі. |
| **Умова:** | Виконано вхід на вкладку реєстрації в системі. |
| Кроки: | Очікуваний результат: |
| 1. Ввести коректні дані. | Введено коректні дані. |
| 2. Натиснути кнопку “Реєстрація”. | Відбудеться переадресація на сторінку з авторизацією. |
| **Опис**: | Введення некоректних даних при реєстрації. |
| **Умова**: | Виконано вхід на вкладку реєстрації в системі. |
| Кроки: | Очікуваний результат: |
| 1. Ввести некоректні дані. | Введено коректні дані. |
| 2. Натиснути кнопку “Реєстрація”. | Система повідомляє, що дані було введено некоректно. |
| **Опис:** | Зміна даних в профілі користувача. |
| **Умова:** | Відкрито вкладку з профілем користувача. |
| Кроки: | Очікуваний результат: |
| 1. Змінити дані користувача. | Дані було змінено. |

Продовження таблиці 4.2

|  |  |
| --- | --- |
| 2. Натиснути кнопку “Зберегти”. | Дані було збережено в системі. |
| **Опис:** | Видалення профілю користувача. |
| **Умова:** | Відкрито вкладку з профілем користувача. |
| Кроки: | Очікуваний результат: |
| 1. Натиснути кнопку “Видалити акаунт”. | Відбудеться переадресація на сторінку з підтвердженням вибору щодо видалення. |
| 2. Натиснути кнопку “Так”. | Відбудеться переадресація на головну сторінку системи. |
| **Опис:** | Додавання товару до кошику. |
| **Умова:** | Відкрито вкладку з необхідним товаром до придбання. |
| Кроки: | Очікуваний результат: |
| 1. Натиснути кнопку “Додати в кошик”. | Відбудеться переадресація на сторінку кошику з відповідним товаром в ньому. |
| **Опис**: | Зміна кількості товару в кошику. |
| **Умова**: | Виконано вхід на вкладку кошику. |
| Кроки: | Очікуваний результат: |
| 1. Змінити кількість товару в кошику. | Кількість товару була змінена. |
| 2. Натиснути кнопку “Змінити”. | Система перераховує загальну суму покупок клієнта. |
| **Опис**: | Введення некоректних даних в кошику. |
| **Умова**: | Виконано вхід на вкладку кошику. |
| Кроки: | Очікуваний результат: |
| 1. Ввести некоректні дані | Некоректні дані було введено. |
| 2. Натиснути кнопку “Змінити”. | Система повідомляє, що дані було введено некоректно. |

Продовження таблиці 4.2

|  |  |
| --- | --- |
| **Опис**: | Видалення товару з кошику. |
| **Умова**: | Виконано вхід на вкладку кошику. |
| Кроки: | Очікуваний результат: |
| 1. Натиснути на “X”. | Товар з кошику було видалено. |
| **Опис**: | Залишення побажання в контактній формі. |
| **Умова**: | Виконано вхід на контактів. |
| Кроки: | Очікуваний результат: |
| 1. Прогорнути до низу сторінки. | Відображення контактної форми. |
| 2. Заповнити поля форми. | Поля було заповнено. |
| 3. Натиснути кнопку “Відправити”. | Повідомлення було відправлено до бази даних системи. |

# 4.3 Введення в експлуатацію

Після того як система буде повністю готова її можна розгорнути на віддаленому хостингу, завантаживши файли і структуру каталогів системи. Також систему можна упакувати у файл WAR/JAR і розгорнути як один файл.

# ВИСНОВКИ

В ході дипломного проектування було створено інформаційну систему кав’ярні з використанням PHP5, HTML та СУБД MySQL, яка дозволяє:

1. Створювати та видаляти обліковий запис клієнту.
2. Переглядати меню закладу.
3. Переглядати наявні товари та купувати їх.
4. Добавляти, редагувати та видаляти товар з системи.
5. Видаляти облікові записи клієнтів.

Було проведено аналіз існуючих систем: технології створення, наявний функціонал та дизайн. На основі даного аналізу було визначено основні функції для системи, що була розроблена.

Наступним кроком було розроблено структурну частину системи, побудовано діаграму використання для відображення взаємодії користувачів системи, діаграму послідовності для відображення відношення користувачів системи в різних умовах, діаграми станів та діаграму розгортання. Також на даному етапі було спроектовано базу даних: визначено структуру бази та основні сутності ІС.

Далі було розроблено базу даних згідно проектування, та з допомогою обраних засобів програмування було розроблено користувацький інтерфейс згідно з розроблених макетів.

Останнім кроком було проведення ручного тестування системи згідно з тест плану для виявлення та запобігання дефектів. Також згідно тест плану було створено тест-кейси для відповідних функцій системи.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ринок кави в Україні – аналітичний огляд від Pro-Consulting спеціально для InVenture [Електронний ресурс] – Режим доступу до ресурсу: <https://pro-consulting.ua/ua/pressroom/rynok-kofe-v-ukraine-analiticheskij-obzor-ot-pro-consulting-specialno-dlya-inventure>.
2. Після підписання угоди з ЄС, ринок кави в україні значно розширився [електронний ресурс] – режим доступу до ресурсу: <https://vezha.net.ua/business/pislya-pidpisannya-ugodi-z-yes-rinok-kavi-v-ukrayini-znachno-rozshirivsya/>.
3. Inside Ukraine’s Booming Coffee Scene [Електронний ресурс] – Режим доступу до ресурсу: <https://perfectdailygrind.com/2020/04/insides-ukraines-booming-coffee-scene/>.
4. Ukrainian Coffee Culture Is Booming, And Here Is Why [Електронний ресурс] – Режим доступу до ресурсу: https://europeancoffeetrip.com/ukrainian-coffee-culture/.
5. Acquiring information systems and services [Електронний ресурс] – Режим доступу до ресурсу: <https://www.britannica.com/topic/information-system/Acquiring-information-systems-and-services>.
6. Aroma Kava [Електронний ресурс] – Режим доступу до ресурсу: <https://aromakava.ua/>.
7. Gangster\_Coffe Shop [Електронний ресурс] – Режим доступу до ресурсу: <https://gangster.coffeezgar.com.ua/>.
8. Світ кави [Електронний ресурс] – Режим доступу до ресурсу: <https://svitkavy.com/uk/>.
9. Blue Cup Coffee Shop [Електронний ресурс] – Режим доступу до ресурсу: <https://bluecup.com.ua/?fbclid=IwAR3OFvdw5SYYRat274_pWN4eZwZGOl-aMwDVuCIEiVVsCCjjE4DFn1dBHzs>.
10. Гвоздева Т. В. Проектирование информационных систем. Стандартизация : учебное пособие для вузов / Т. В. Гвоздева, Б. А. Баллод. – Санкт-Петербург: Лань, 2021. – 252 с.
11. Гайдаржи В. І. Бази даних в інформаційних системах / В. І. Гайдаржи, І. В. Ізварін. – Київ: Університет "Україна", 2018. – 418 с.
12. Анісімов А.В. Інформаційні системи та бази даних: Навчальний посібник для студентів факультету комп’ютерних наук та кібернетики. / Анісімов А.В., Кулябко П.П. − Київ. – 2017. – 110 с.
13. Морзе Н.В. Інформаційні системи. Навч. посібн. /за наук. ред. Н. В. Морзе; Морзе Н.В., Піх О.З. – Івано-Франківськ, «ЛілеяНВ», – 2015. – 384 с.
14. Haverbeke M. Eloquent JavaScript: A Modern Introduction to Programming / Marijn Haverbeke. – 3rd ed. – San Francisco: No Starch Press, 2018. – 463 c.
15. Atencio L. Functional Programming in JavaScript: How to improve your JavaScript programs using functional techniques / Luis Atencio – 1st ed. – New York: Manning Publications, 2016. – 272 c.
16. Learn to Code HTML & CSS [Електронний ресурс] – Режим доступу до ресурсу: https://learn.shayhowe.com/html-css/.
17. Маклафлин Б. PHP и MySQL. Исчерпывающее руководство. 2-е изд. / Б. Маклафлин. – Питер: Санкт-Петербург, 2017. – 544 с.
18. Nixon R. Learning PHP, MySQL & JavaScript / Robin Nixon – 5th ed. – Sebastopol: O’Reilly, 2018. – 832 c.
19. Kevin Tatroe Programming PHP, 4th Edition / Kevin Tatroe – 4th ed. – Sebastopol: O’Reilly, 2020. – 544 c.
20. PHP: The Right Way [Електронний ресурс] – Режим доступу до ресурсу: https://phptherightway.com/.

# ДОДАТОК А ЛІСТИНГ КОДУ ДЛЯ РЕАЛІЗАЦІЇ РЕДАГУВАННЯ ТОВАРІВ

<?php

header('Content-Type: text/html; charset=UTF-8');

?>

<?php

error\_reporting(E\_ALL);

ini\_set('display\_errors','1');

include "../storescripts/db\_connection.php";

?>

<!DOCTYPE HTML>

<html lang="uk">

<head>

<title>Зміна даних товару</title>

<link rel="shortcut icon" href="../style/favicon.ico" type="image/x-icon">

<link rel="stylesheet" type="text/css" href="../style/index.css" media="screen" />

<meta http-equiv="Content-Type" content="text/html" charset="UTF-8" />

<meta name="keywords" content="Coffee, online, shop, beans, merch, buy" />

<meta name="description" content="Online Shop" />

</head>

<body>

<?php include\_once 'template\_header3.php';?><br><br><br><br>

<?php

if (isset($\_GET['pid'])) {

$targetID = $\_GET['pid'];

$sql = mysqli\_query($conn, "SELECT \* FROM products WHERE id = '$targetID' LIMIT 1");

$productCount = mysqli\_num\_rows($sql);

if($productCount > 0) {

while ($row = mysqli\_fetch\_array($sql)) {

$product\_name = $row["product\_name"];

$price = $row["price"];

$category = $row["category"];

$detail = $row["detail"];

$date\_added = strftime("%b %d, %Y", strtotime($row["date\_added"]));

}} else {

echo "Товару з даним ідентифікатором не виявлено.";

exit();

}} ?>

<?php

if (isset($\_POST['edit'])) {

$pid = mysqli\_real\_escape\_string($conn, $\_POST['thisID']);

$product\_name = mysqli\_real\_escape\_string($conn, $\_POST['product\_name']);

$price = mysqli\_real\_escape\_string($conn, $\_POST['price']);

$category = mysqli\_real\_escape\_string($conn, $\_POST['category']);

$detail = mysqli\_real\_escape\_string($conn, $\_POST['detail']);

if ($\_FILES['image']['tmp\_name'] != ""){

$newname = "$pid.jpg";

move\_uploaded\_file( $\_FILES['image']['tmp\_name'], "../inventory\_images/$newname");

$sql = mysqli\_query($conn, "UPDATE products SET product\_name = '$product\_name', price = '$price', category = '$category', detail = '$detail' WHERE id='$pid' ");}

else{

$sql = mysqli\_query($conn, "UPDATE products SET product\_name = '$product\_name', price = '$price', category = '$category', detail = '$detail' WHERE id='$pid' ");

}

header("location: inventory\_edit.php?pid=$pid");

exit();

}

?>

<div id="content">

<div align="center">

<h2>Зміна даних товару</h2>

<form action="inventory\_edit.php?pid=<?php echo $targetID; ?>" enctype="multipart/form-data" name="myForm" id="myform" method="post">

<table border="0" cellspacing="0" cellpadding="6">

<tr>

<td width="20%" align="right">Назва товару</td>

<td width="80%"><label>

<input name="product\_name" type="text" id="product\_name" size="64" value="<?php echo $product\_name;?>" />

</label></td>

</tr>

<tr>

<td align="right">Ціна товару</td>

<td><label>

<input name="price" type="text" id="price" size="12" value="<?php echo $price;?>" />

</label></td>

</tr>

<tr>

<td align="right">Категорія</td>

<td><label>

<select name="category" id="category">

<option value="<?php echo $category;?>"> <?php echo $category;?> </option>

<option value="Coffee">Кава</option>

<option value="Merch">Мерч</option>

<option value="Instruments">Аксесуари</option>

<option value="Cupons">Карти</option>

</select>

</label></td>

</tr>

<tr>

<td align="right">Опис</td>

<td><label>

<textarea name="detail" id="detail" cols="64" rows="5" > <?php echo $detail;?> </textarea>

</label></td>

</tr>

<tr>

<td align="right">Зображення товару</td>

<td><label>

<input type="file" name="image" id="image" />

</label></td>

</tr>

<tr>

<td align="right"></td>

<td><label>

<input name="thisID" type="hidden" value="<?php echo $targetID;?>">

<input type="submit" name="edit" id="submit" class="submBtn" value="Змінити"/>

</label></td>

</tr></table></form>

<br>

<a class="white" href="inventory\_list.php">Зміна завершена? Назад.</a><br><br><br>

</div></div>

</body>

</html>

# ДОДАТОК Б ЛІСТИНГ КОДУ ДЛЯ РЕАЛІЗАЦІЇ КОШИКУ

<?php

session\_start();

if(!isset($\_SESSION["user"])) {

header("location: authorization.php");

exit();

}

$userID = preg\_replace('#[^0-9]#i', '', $\_SESSION["id"]);

$user = preg\_replace('#[^A-Za-z0-9]#i', '', $\_SESSION["user"]);

$user\_pass = preg\_replace('#[^A-Za-z0-9]#i', '', $\_SESSION["user\_pass"]);

include "storescripts/db\_connection.php";

$sql = mysqli\_query($conn, "SELECT \* FROM user\_account WHERE id='$userID' AND username='$user' AND password='$user\_pass' LIMIT 1");

$existCount = mysqli\_num\_rows($sql);

if ($existCount == 0) {

echo "Ваші дані для входу не зберігаються в базі даних.";

exit();

}

error\_reporting(E\_ALL);

ini\_set('display\_errors', '1');

include 'storescripts/db\_connection.php';

header('Content-Type: text/html; charset=UTF-8');

?>

<?php

if (isset($\_GET['log\_out'])){

unset($\_SESSION['user']);

unset($\_SESSION['user\_pass']);

unset($\_SESSION['cart\_array']);

header('location: index.php');

}

?>

<?php

if (isset($\_POST['pid'])) {

$pid = $\_POST['pid'];

$wasFound = false;

$i = 0;

if (!isset($\_SESSION["cart\_array"]) || count($\_SESSION["cart\_array"]) < 1) {

$\_SESSION["cart\_array"] = array(0 => array("item\_id" => $pid, "quantity" => 1));

}

else {

foreach ($\_SESSION["cart\_array"] as $each\_item) {

$i++;

while (list($key, $value) = each($each\_item)) {

if ($key == "item\_id" && $value == $pid) {

array\_splice($\_SESSION["cart\_array"], $i-1, 1, array(array("item\_id" => $pid, "quantity" => $each\_item['quantity'] + 1)));

$wasFound = true;

}

}

}

if ($wasFound == false) {

array\_push($\_SESSION["cart\_array"], array("item\_id" => $pid, "quantity" => 1));

}

}

header("location: cart.php");

exit();

}

?>

<?php

if (isset($\_GET['cmd']) && $\_GET['cmd'] == "emptycart") {

unset($\_SESSION["cart\_array"]);

}

?>

<?php

if (isset($\_POST['item\_to\_adjust']) && $\_POST['item\_to\_adjust'] != "") {

$item\_to\_adjust = $\_POST['item\_to\_adjust'];

$quantity = $\_POST['quantity'];

$quantity = preg\_replace('#[^0-9]#i', '', $quantity);

if ($quantity >= 100) { $quantity = 99; }

if ($quantity < 1) { $quantity = 1; }

if ($quantity == "") { $quantity = 1; }

$i = 0;

foreach ($\_SESSION["cart\_array"] as $each\_item) {

$i++;

while (list($key, $value) = each($each\_item)) {

if ($key == "item\_id" && $value == $item\_to\_adjust) {

array\_splice($\_SESSION["cart\_array"], $i-1, 1, array(array("item\_id" => $item\_to\_adjust, "quantity" => $quantity)));

}

}

}

header("location: cart.php");

exit();

}

?>

<?php

if (isset($\_POST['index\_to\_remove']) && $\_POST['index\_to\_remove'] != "") {

$key\_to\_remove = $\_POST['index\_to\_remove'];

if (count($\_SESSION["cart\_array"]) <= 1) {

unset($\_SESSION["cart\_array"]);

} else {

unset($\_SESSION["cart\_array"]["$key\_to\_remove"]);

sort($\_SESSION["cart\_array"]);

}

}

?>

<?php

$cartOutput = "";

$cartTotal = 0;

$pp\_checkout\_btn = '';

$product\_id\_array = '';

if (!isset($\_SESSION["cart\_array"]) || count($\_SESSION["cart\_array"]) < 1) {

$cartOutput = "<h2 align='center'>Кошик порожній</h2>";

}

else {

$pp\_checkout\_btn .= '<form action="https://www.paypal.com/cgi-bin/webscr" method="post">

<input type="hidden" name="cmd" value="\_cart">

<input type="hidden" name="upload" value="1">

<input type="hidden" name="business" value="you@youremail.com">';

$i = 0;

foreach ($\_SESSION["cart\_array"] as $each\_item) {

$item\_id = $each\_item['item\_id'];

$sql = mysqli\_query($conn, "SELECT \* FROM products WHERE id = '$item\_id' LIMIT 1");

while ($row = mysqli\_fetch\_array($sql)) {

$product\_name = $row["product\_name"];

$price = $row["price"];

}

$pricetotal = $price \* $each\_item['quantity'];

$cartTotal = $pricetotal + $cartTotal;

$pricetotal = number\_format($pricetotal, 2, ',', '\'');

$x = $i + 1;

$pp\_checkout\_btn .= '<input type="hidden" name="item\_name\_'.$x.'" value="'.$product\_name.'">

<input type="hidden" name="amount\_'.$x.'" value="'.$price.'">

<input type="hidden" name="quantity\_'.$x.'" value="'.$each\_item['quantity'].'">';

$product\_id\_array .= "$item\_id-".$each\_item['quantity'].",";

$cartOutput .= "<tr>";

$cartOutput .= '<td><a href="product.php?id='.$item\_id.'" style="color: black;">'.$product\_name.'<br />

<img src="inventory\_images/'.$item\_id.'.jpg" alt="'.$product\_name.'" width="40" height="52" border="1" /></a></td>';

$cartOutput .= "<td>".$price."</td>";

$cartOutput .= '<td><form action="cart.php" method="post">

<input name="quantity" type="text" value="'.$each\_item['quantity'].'" size="1" maxlength="2" />

<input name="adjustQuantity'.$item\_id.'" type="submit" value="Змінити" class="add" />

<input name="item\_to\_adjust" type="hidden" value="'.$item\_id.'" /></form></td>';

$cartOutput .= "<td>".$pricetotal."</td>";

$cartOutput .= '<td><form action="cart.php" method="post">

<input name="deleteProduct'.$item\_id.'" type="submit" value="X" class="submBtn" />

<input name="index\_to\_remove" type="hidden" value="'.$i.'" /></form></td>';

$cartOutput .= "</tr>";

$i++;

}

}

$cartTotal = number\_format($cartTotal, 2, ',', '\'');

$cartTotal = '<div align="right">Загальна сума: '.$cartTotal." грн.</div>";

$pp\_checkout\_btn .= '<input type="hidden" name="custom" value="'.$product\_id\_array.'">

<input type="hidden" name="notify\_url" value="storescripts/my\_ipn.php">

<input type="hidden" name="return" value="account.php">

<input type="hidden" name="rm" value="2">

<input type="hidden" name="cbt" value="Return to The Store">

<input type="hidden" name="cancel\_return" value="account.php">

<input type="hidden" name="lc" value="EU">

<input type="hidden" name="currency\_code" value="грн.">

<input type="image" src="http://www.paypal.com/en\_US/i/btn/x-click-but01.gif" name="submit" alt="Make payments with PayPal - its fast, free and secure!">

</form>';

?>

<!DOCTYPE html>

<html lang="uk">

<head>

<meta http-equiv="Content-Type" content="text/html" charset="UTF-8" />

<title>Кошик Покупця</title>

<link rel="shortcut icon" href="style/favicon.ico" type="image/x-icon">

<meta name="keywords" content="Coffee, online, shop, beans, merch, buy" />

<meta name="description" content="Product" />

<link rel="stylesheet" type="text/css" href="style/index.css" media="screen" />

</head>

<body>

<header>

<div id="barline">

<a href="index2.php">Головна</a>

<a href="product2.php">Пропозиції</a>

<a href="shop2.php">Магазин</a>

<a href="BroadcastCoffee2.html">Про нас</a>

<a href="kontakt2.php">Контакти</a>

<a class="right" href="?log\_out">Вийти</a>

<a class="right" href="cart.php">Корзина</a>

</div>

</header>

<br><br><br><br><br>

<br><br><br><br>

<hr color= "black" size=5px width=1000px align="center">

<div id="slogan">Корзина</div>

<hr color= "black" size=5px width=1000px align="center"><br><br>

<div id="content" align="left">

<table border="1" cellspacing="0" cellpadding="6">

<tr>

<td width="20%"><strong>Товар</strong></td>

<td width="4%"><strong>Ціна за одиницю</strong></td>

<td width="9%"><strong>Кількість</strong></td>

<td width="4%"><strong>Сума</strong></td>

<td width="3%"><strong>Видалити</strong></td>

</tr>

<?php echo $cartOutput; ?>

</table><br>

<?php echo $cartTotal; ?><br>

<div align="right"><?php echo $pp\_checkout\_btn; ?></div><br>

<center><a href="cart.php?cmd=emptycart" style="color: black;">Натисніть тут, щоб очистити корзину</a></center>

</div>

<br>

</body>

</html>