

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами

Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ



к.т.н, доц. Михелєв І.Л.

" ____ " _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "бакалавр"

на тему: Розроблення інформаційної системи магазину

музичних інструментів "MuzSvit"

Виконав: студент групи 4147 ст

 *Новожилов О.Д.*

(підпис)

Керівник роботи:

(посада, науковий ступінь вчене звання)

 *Беркунська Н.М.*

(підпис)

м. Миколаїв – 2025 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет кораблебудування імені адмірала
Макарова

Навчально-науковий інститут комп'ютерних наук та управління проектами
 Кафедра інформаційних управляючих систем та технологій
 Спеціальність 122 "Комп'ютерні науки"
 Освітня програма "Комп'ютерні науки"

"ЗАТВЕРДЖУЮ"
 Гарант освітньої програми

 к.т.н., доцент Гайда А.Ю.
 " ____ " _____ 2025 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ
 на здобуття ступеня вищої освіти "бакалавр"

Студенту Новожилову Олександр Дмитровичу

(прізвище, ім'я, по батькові)

1. Тема роботи: Розроблення інформаційної системи магазину музичних інструментів

"MuzSvit"

Керівник роботи Беркунська Наталя Михайлівна

Затверджені наказом ректора № 295-УЧ від "27" березня 2025 року.

2. Термін подання роботи: 20 червня 2025р.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем.

4. Перелік питань, що належать до розробки (найменування розділів): Вступ, розробка концепції інформаційної системи, розробка проєктних рішень, реалізація проєкту інформаційної системи, охорона праці, висновки.

5. Перелік презентаційних матеріалів: Титульний слайд, мета та завдання дослідження, актуальність теми, аналіз предметної області, формування вимог до ІС, проєктні рішення, інтерфейс та функціональність, механізми безпеки та перевірки даних, результати реалізації, висновки та перспективи розвитку.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Беркунська Н. М.	03.02.2025	22.02.2025
2	Беркунська Н. М.	22.02.2025	21.03.2025
3	Беркунська Н. М.	21.03.2025	25.04.2025
4	Беркунська Н. М.	25.04.2025	07.06.2025

7. Дата видачі завдання 03 лютого 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів роботи	Примітка
1	Аналіз предметної галузі	13.02.2025	
2	Постановка задачі	15.02.2025	
3	Розробка концепції	22.02.2025	
4	Розробка проєкту ІС	21.03.2025	
5	Реалізація проєкту	25.04.2025	
6	Розробка питань з охорони праці	07.06.2025	
7	Розробка графічних матеріалів	12.06.2025	
8	Подання роботи на перевірку на плагіат	10.06.2025	
9	Подання роботи рецензенту	15.06.2025	
10	Подання роботи на захист	20.06.2025	

Студент



(підпис)

Новожилов О.Д.

(ПІБ)

Керівник роботи



(підпис)

Беркунська Н.М.

(ПІБ)

АНОТАЦІЯ

Дана кваліфікаційна робота присвячена розробці інформаційної системи (ІС) магазину музичних інструментів "MuzSvit". Основною метою є створення ефективної та зручної системи управління контентом, замовленнями, клієнтськими запитами та іншими адміністративними функціями магазину.

У ході розробки проведено аналіз вимог до системи, визначено ключові функціональні можливості, такі як управління каталогом товарів, обробка замовлень, управління користувачами та ведення аналітики.

Результатом роботи є інформаційна система, яка дозволяє зменшити навантаження на персонал, покращити обслуговування клієнтів та підвищити ефективність бізнес-процесів.

Основна частина включає 53 сторінки друкованого тексту, 12 рисунків, 10 таблиць, 2 додатки, перелік використаних джерел містить 20 найменувань. Робота виконана українською мовою.

Ключові слова: інформаційна система, інтернет-магазин, веб-інтерфейс, клієнт-серверна архітектура, Node.js, React.js, MySQL.

ABSTRACT

This qualification work is dedicated to the development of the information system (IS) for the musical instruments store "MuzSvit". The primary objective is to create an efficient and user-friendly system for managing content, orders, customer inquiries, and other administrative functions of the store.

Throughout the development, the system requirements were analyzed, and key functionalities were identified, including product catalog management, order processing, user management, and analytics tracking.

The outcome of this work is an information system that reduces staff workload, improves customer service, and enhances efficiency of business processes.

The main part consists of 53 pages of printed text, 12 figures, 10 tables, and 2 appendices. The reference list contains 20 sources. The work is completed in Ukrainian language.

Keywords: information system, online store, web interface, client-server architecture, Node.js, React.js, MySQL.

ЗМІСТ

ВСТУП	9
1 РОЗРОБКА КОНЦЕПЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ	10
1.1 Аналіз предметної області	10
1.1.1 Характеристика діяльності інтернет-магазину "MuzSvit"	10
1.1.2 Опис об'єкту автоматизації	11
1.1.3 Обґрунтування необхідності створення ІС магазину	12
1.2 Формування вимог до інформаційної системи	14
1.2.1 Аналіз функцій ІС	14
1.2.2 Аналіз системних вимог	15
1.2.3 Вибір логічної архітектури програмного забезпечення	15
1.2.4 Вибір фізичної архітектури програмного забезпечення ІПС магазину	17
1.3. Розробка концепції ІС магазину	17
1.3.1 Розробка концепції	17
1.3.2 Техніко-економічне обґрунтування обраної концепції	19
1.3.3 Орієнтовний план реалізації обраного варіанта концепції системи	20
1.3.4 Умови приймання системи	21
1.4 Постановка задачі	22
1.4.1 Загальні відомості	22
1.4.2 Призначення і мета створення ІС інтернет-магазину музичних інструментів "MuzSvit"	23
1.4.3 Вимоги до ІС "MuzSvit"	23
1.4.4 Перспективи використання та подальшого розвитку ІС «MuzSvit»	24
1.5 Вибір засобів та середовища розробки	25

2 РОЗРОБКА ПРОЄКТНИХ РІШЕНЬ	28
2.1 Структура інформаційної системи	28
2.2 Інформаційне забезпечення системи	29
2.2.1 Визначення сутностей та концептуальна модель даних	30
2.2.2 Розробка логічної моделі даних	33
2.2.3 Розробка фізичної моделі даних	33
2.3 Вибір алгоритмів та обробка даних	36
3 РЕАЛІЗАЦІЯ ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ	39
4 ОХОРОНА ПРАЦІ	46
4.1 Вступ	46
4.2 Аналіз небезпечних і шкідливих виробничих факторів	47
4.3 Розробка заходів щодо забезпечення сприятливих умов праці при роботі з персональним комп'ютером	48
ВИСНОВКИ	51
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	52
ДОДАТКИ	54

ВСТУП

В сучасних умовах розвитку інформаційних технологій ефективне функціонування підприємств торгівлі, зокрема спеціалізованих магазинів, потребує впровадження автоматизованих інформаційних систем. Це дозволяє не тільки оптимізувати внутрішні бізнес-процеси, а й підвищити якість обслуговування клієнтів, покращити облік товарів та забезпечити оперативний доступ до даних.

Магазин музичних інструментів "MuzSvit" пропонує широкий асортимент продукції для різних категорій покупців. Для забезпечення ефективної роботи магазину виникла потреба у створенні спеціалізованої інформаційної системи, яка б автоматизувала процеси управління товарами, обліку продажів, формування звітності та взаємодії з клієнтами.

Метою даної кваліфікаційної роботи є підвищення ефективності управління товарними запасами та продажами.

Завданням роботи є проектування та розробка інформаційної системи для магазину "MuzSvit", яка буде відповідати сучасним вимогам до швидкості, надійності та зручності роботи з даними.

Для досягнення поставленої мети буде проведено аналіз предметної області, визначено функціональні вимоги до системи, спроектовано базу даних та реалізовано основні компоненти інформаційної системи.

Актуальність теми визначається необхідністю переходу від частково автоматизованого обліку до комплексної цифровізації діяльності магазину, що дозволить покращити управління товарними запасами, прискорити обробку замовлень і забезпечити конкурентоспроможність на ринку музичних інструментів.

1 РОЗРОБКА КОНЦЕПЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

1.1 Аналіз предметної області

1.1.1 Характеристика діяльності інтернет-магазину "MuzSvit"

Магазин музичних інструментів "MuzSvit" спеціалізується на продажі широкого спектра музичних товарів: струнних, клавішних, духових інструментів, електроніки для музикантів, а також супутніх товарів (аксесуарів, звукового обладнання).

В процесі своєї діяльності магазин здійснює:

- прийом та облік товарів від постачальників;
- зберігання інформації про асортимент продукції (характеристики, кількість, ціна);
- оформлення продажу товарів клієнтам;
- облік залишків на складі;
- формування фінансової та товарної звітності.

Основними учасниками процесу є:

1. Адміністратор магазину – відповідає за управління контентом, обробку замовлень, взаємодію з клієнтами та контроль за фінансовими операціями.
2. Клієнти – користувачі, які переглядають каталог товарів, оформлюють замовлення та здійснюють оплату.
3. Постачальники – юридичні або фізичні особи, що постачають продукцію магазину.
4. Служби доставки – логістичні компанії, що забезпечують доставку замовлень клієнтам.

Основні бізнес-процеси предметної області:

- Управління каталогом товарів – додавання, оновлення та видалення позицій.
- Обробка замовлень – прийом, підтвердження та виконання замовлень.

- Опрацювання платежів – взаємодія з банками та платіжними системами.
- Логістика та доставка – передача замовлень до служб доставки та відстеження.
- Клієнтська підтримка – комунікація з клієнтами щодо їхніх замовлень та запитів.

На даний момент більшість облікових процесів виконується вручну або за допомогою стандартних офісних програм, що знижує ефективність роботи персоналу та підвищує ризик допущення помилок.

Тому виникла потреба у створенні цілісної інформаційної системи для автоматизації основних бізнес-процесів магазину.

Аналіз предметної області дозволить сформулювати ефективну концепцію автоматизованого робочого місця адміністратора, що оптимізує управління магазином та покращить рівень обслуговування клієнтів.

1.1.2 Опис об'єкту автоматизації

Об'єктом автоматизації є магазин музичних інструментів "MuzSvit", який здійснює торгівлю музичними інструментами та супутніми товарами, надає консультаційні послуги покупцям та проводить акції для постійних клієнтів.

Основними характеристиками об'єкта є:

- Асортимент товарів: струнні, клавішні, духові, ударні інструменти, студійне обладнання, музичні аксесуари.
- Постачальники: кілька постійних постачальників з різних країн, з якими укладено договори поставки.
- Клієнти: як роздрібні покупці, так і організації (музичні школи, творчі колективи), які купують товари оптом.
- Персонал: продавці-консультанти, комірники, адміністратори магазину, які забезпечують роботу торгового залу та складу.

Основні бізнес-процеси:

- Приймання товару на склад.

- Облік наявності товарів та їхніх характеристик.
- Продаж товарів через касу магазину.
- Повернення товарів.
- Формування фінансової звітності та аналітики продажів.
- Програма лояльності для постійних покупців.

Проблеми, що існують на об'єкті:

- Відсутність єдиної централізованої бази даних для обліку товарів та клієнтів.
- Низька швидкість обробки інформації через використання паперових журналів та електронних таблиць.
- Високий ризик помилок при обробці замовлень та проведенні інвентаризацій.
- Труднощі у формуванні оперативних звітів для прийняття управлінських рішень.

Метою автоматизації є:

- Підвищення ефективності управління товарними запасами та продажами.
- Зменшення часу обробки операцій із клієнтами.
- Підвищення точності даних обліку.
- Забезпечення швидкого доступу до аналітичної інформації для прийняття рішень.

1.1.3 Обґрунтування необхідності створення ІС магазину

Зі зростанням кількості замовлень ручна обробка може стати занадто часозатратною та призвести до помилок. Інформаційна система дозволить адміністратору швидко переглядати нові замовлення, оновлювати їхні статуси, відстежувати оплати та координувати доставку в єдиному інтерфейсі. Це значно скоротить час обробки кожного замовлення та підвищить задоволеність клієнтів.

Музичні інструменти мають безліч характеристик, які важливо точно відображати на сайті. ІС магазину зі зручними інструментами для додавання, редагування та класифікації товарів (з урахуванням усіх їхніх особливостей) мінімізує ризик помилок в описі, полегшить пошук для покупців та забезпечить актуальність інформації про наявність і ціни.

Аналіз продажів, популярності товарів, ефективності рекламних кампаній є ключовим для прийняття обґрунтованих бізнес-рішень. Автоматизована система звітності в ІС магазину дозволить швидко отримувати необхідні дані у зручному форматі, без необхідності вручну збирати та обробляти інформацію з різних джерел.

Ручне введення даних, особливо при великій кількості товарів та замовлень, неминуче призводить до помилок. Автоматизація багатьох процесів через ІС мінімізує цей ризик, що позитивно впливає на точність обліку, обробки замовлень та задоволеність клієнтів.

Швидкий доступ до інформації про замовлення та клієнтів дозволить адміністратору оперативно відповідати на їхні запитання, надавати актуальну інформацію про статус замовлення та вирішувати можливі проблеми. Це сприятиме підвищенню лояльності клієнтів.

У міру зростання інтернет-магазину "MuzSvit" ручні методи управління ставатимуть все менш ефективними. Впровадження ІС магазину закладає міцну основу для подальшого масштабування бізнесу, дозволяючи обробляти більшу кількість замовлень та керувати ширшим асортиментом товарів без значного збільшення адміністративного персоналу.

Сучасний та ергономічний інтерфейс зробить роботу адміністратора більш комфортною та продуктивною, зменшить час на навчання нових співробітників та підвищить їхню задоволеність роботою.

Отже впровадження ІС магазину значно підвищить ефективність роботи магазину, дозволить зменшити витрати часу на обробку інформації, знизити кількість помилок і покращити загальне управління ресурсами підприємства.

1.2 Формування вимог до інформаційної системи

1.2.1 Аналіз функцій ІС

Інформаційна система інтернет-магазину «MuzSvit» забезпечує комплекс функцій, які дозволяють ефективно організувати діяльність магазину. Основні функції ІС включають:

Управління каталогом продукції:

- Додавання, редагування та видалення товарів.
- Організація товарів за категоріями та брендами.
- Завантаження та управління зображеннями товарів.

Управління замовленнями:

- Прийом та обробка замовлень від клієнтів.
- Автоматична реєстрація замовлень та статусів їх виконання.
- Генерація історії замовлень у профілі користувача.

Управління користувачами:

- Реєстрація та автентифікація користувачів.
- Редагування профілів та збереження контактних даних.
- Розмежування доступу (користувач, адміністратор).

Аналітика та звітність:

- Формування звітів щодо продажів, популярності товарів.
- Відстеження кількості замовлень та інших важливих бізнес-показників.

Реалізація зазначених функцій забезпечує підвищення ефективності управління інтернет-магазином.

Аналіз функцій показує, що ІС магазину забезпечує обробку даних, автоматизацію рутинних операцій та створення основи для прийняття ефективних управлінських рішень.

1.2.2 Аналіз системних вимог

Для забезпечення стабільної роботи ІС визначено наступні системні вимоги:

Програмні вимоги:

Операційна система: Windows або Linux.

Серверне середовище: Node.js версії 18.x або новіше.

База даних: MySQL 8.x.

Клієнтська частина: React.js.

Інструменти управління залежностями: npm.

Апаратні вимоги:

Процесор: не нижче Intel Core i3.

Оперативна пам'ять: не менше 4 GB.

Жорсткий диск: не менше 20 GB вільного місця.

Стабільне підключення до мережі Інтернет.

1.2.3 Вибір логічної архітектури програмного забезпечення

Розробка інтернет-магазину музичної техніки передбачає вибір відповідної логічної архітектури, яка забезпечить ефективну роботу системи, масштабованість та безпеку.

Обрана архітектура базується на трьох рівнях:

- Презентаційний рівень (Front-end) – відповідає за взаємодію з користувачем. Реалізується на основі веб-технологій, таких як React.js.
- Логічний рівень (Back-end) – містить бізнес-логіку, обробку запитів та взаємодію між рівнями. Використовується Node.js.
- Рівень даних (Database) – відповідає за збереження інформації, управління запитам та цілісність даних. Використовується MySQL.

Основні компоненти архітектури:

- Контролер користувачів – керує реєстрацією, автентифікацією та управлінням доступом.
- Модуль управління товарами – забезпечує додавання, редагування та видалення товарів.

- Модуль замовлень – обробляє замовлення, змінює їх статуси та взаємодіє з платіжними системами.
- Аналітичний модуль – відповідає за формування звітів та аналіз поведінки клієнтів.
- Інтеграційний модуль – дозволяє підключати зовнішні API (служби доставки, платіжні системи, CRM).

Переваги обраної архітектури:

- Масштабованість – можливість додавання нових функцій без порушення роботи системи.
- Безпека – чіткий розподіл рівнів забезпечує захист даних користувачів.
- Продуктивність – використання сучасних технологій та кешування даних підвищує швидкість обробки запитів.
- Гнучкість – можливість легкої інтеграції з новими сервісами.

Застосування такої архітектури забезпечує ефективність, надійність та адаптивність автоматизованого робочого місця адміністратора інтернет-магазину.

1.2.4 Вибір фізичної архітектури програмного забезпечення ІПС магазину

При розробці ІС магазину "MuzSvit" важливим етапом є вибір фізичної архітектури програмного забезпечення. Правильний вибір забезпечує високу продуктивність системи, її надійність, масштабованість та безпечний обмін даними.

Враховуючи аналізовані вимоги та функціональні можливості ІПС, для проєкту було обрано клієнт-серверну архітектуру, яка включає наступні компоненти:

Клієнтська частина (frontend) - реалізована на React.js, що забезпечує інтуїтивний інтерфейс користувача, адаптивний дизайн за допомогою Tailwind CSS, та асинхронний обмін даними із сервером.

Серверна частина (backend) - реалізована на Node.js з використанням фреймворку Express.js, яка забезпечує прийом та обробку запитів, авторизацію користувачів, роботу з базою даних MySQL, та виконання бізнес-логіки.

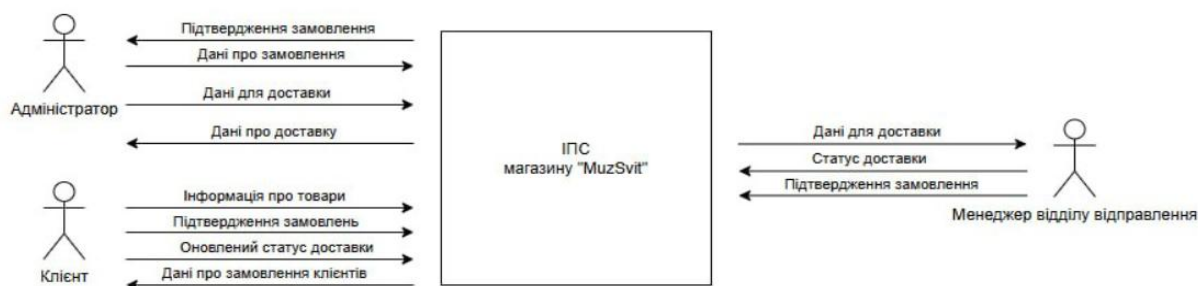
База даних MySQL, що використовується для зберігання та управління інформацією про товари, користувачів та замовлення.

Така архітектура забезпечує масштабованість, ефективність та надійність інформаційної системи магазину «MuzSvit».

1.3. Розробка концепції ІС магазину

1.3.1 Розробка концепції

Проаналізувавши функції ІПС і вимоги до системи, можна розробити організаційно-функціональну структуру (рис.1.1).



Р

исунок 1.1 – Організаційно-функціональна структура ІПС магазину музичних інструментів "MuzSvit"

Функціональні модулі ІПС магазину:

Управління товарами: додавання/редагування/видалення інформації про товари, ведення обліку залишків на складі.

Керування постачальниками: облік постачальників адміністратором. (контактів, контрактів і термінів поставок.)

Обробка замовлень: прийом контактної інформації за замовником, підтвердження, контроль виконання, надання інформації менеджеру відділу відправлення.

Робота з клієнтами: Робота за клієнтом на рівні виводу розширеної інформації щодо товару, формування кабінету клієнта, збереження історії покупок.

Звітність: створення щоденних, щомісячних звітів про продажі, прибутки, залишки на складі.

Безпека: управління правами доступу користувачів, системи забезпечення безпеки даних користувачів.

Функціональні модулі пов'язані з базою даних, яка є джерелом зберігання інформації. Адміністратор, права якого захищені системою безпеки через інтерфейс АРМ взаємодіє з вбудованими модулями, забезпечуючи повноцінне управління бізнес-процесами магазину.

1.3.2 Техніко-економічне обґрунтування обраної концепції

Метою створення веб-застосунку є автоматизація процесу замовлення, обліку товарів і управління клієнтськими даними у сфері онлайн-продажу музичних інструментів. Це дозволить оптимізувати роботу підприємства, знизити людський фактор, підвищити ефективність та забезпечити зручність користувачам.

Впровадження автоматизованої системи дає змогу скоротити витрати часу на обробку замовлень та ведення обліку, зменшити кількість помилок, спричинених людським фактором, швидко отримувати звіти для прийняття управлінських рішень, покращити обслуговування клієнтів за рахунок швидкого пошуку товарів.

Таблиця 1.1 - Використане програмне забезпечення

Компонент	Технологія	Вартість / ліцензія
Серверна частина	Node.js + Express	Безкоштовно
Клієнтська частина	React + Tailwind CSS	Безкоштовно

База даних	MySQL	Безкоштовно
Середовище розробки	VS Code	Безкоштовно
Проектування UI	draw.io	Безкоштовно

Таблиця 1.2 - Розрахунок витрат на розробку

Витрати	Кількість годин	Ставка (грн/год)	Загалом (грн)
Аналіз та проектування	24	1000	24000 грн
Розробка серверної частини	50	1000	50000 грн
Розробка клієнтської частини	48	1000	48000 грн
Тестування та налагодження	18	1000	18000 грн
Разом			140000 грн

В результаті впровадження веб-системи очікується зменшення витрат на обробку замовлень та покращення ефективності роботи персоналу. На основі статистики з подібних проєктів:

- Зниження часу обробки одного замовлення з 8 хв до 2 хв.
- Зменшення кількості помилок при обробці на 90%.
- Очікуване щомісячне навантаження — 300 замовлень.
- Щомісячна економія часу — ~30 год.
- Орієнтовна вартість часу (1000 грн/год): 30000 грн/міс економії.
- Орієнтовна вартість розробки: 140000 грн.
- Очікувана економія: 30000 грн/міс.
- Строк окупності: 4.7 місяців (~4 міс).

Впровадження інформаційної системи в магазині "MuzSvit" є технічно можливим і економічно доцільним. Очікується підвищення ефективності бізнес-процесів, зменшення витрат та покращення якості обслуговування.

1.3.3 Орієнтовний план реалізації обраного варіанта концепції системи

Орієнтовний план реалізації обраного варіанта концепції інформаційної системи наведено в таблиці 1.3.

Таблиця 1.3 - Орієнтовний план реалізації обраного варіанта концепції системи

Етапи розробки	Терміни виконання
Технічне завдання (визначення й аналіз вимог)	03.02.2025-14.02.2025
Ескізне проєктування (розробка специфікацій)	17.02.2025-06.03.2025
Технічне проєктування	07.03.2025-07.04.2025
Робоче проєктування (кодування, налагодження, тестування)	07.04.2025-07.05.2025
Впровадження	07.05.2025-14.05.2025

Ключовий результат: працездатна, адаптована до бізнес-потреб інформаційна система, що підтримує діяльність адміністратора магазину.

1.3.4 Умови приймання системи

Умови приймання системи — це набір чітких технічних, функціональних і організаційних критеріїв, за якими замовник оцінює готовність інформаційної системи до експлуатації.

Основні умови приймання:

Відповідність технічному завданню: усі функції реалізовані згідно з погодженим технічним завданням (ТЗ). Інтерфейс користувача відповідає затвердженим прототипам і макетам. Усі основні модулі системи реалізовані: управління товарами, замовленнями, клієнтами, звітністю та безпекою.

Функціональна повнота: реалізовано повноцінну роботу з базою товарів і клієнтів. Система підтримує авторизацію, реєстрацію та розмежування доступу (ролі користувачів). Забезпечено сортування, пошук і фільтрацію інформації.

Надійність та стабільність роботи: відсутні критичні помилки, що призводять до втрати даних чи збоїв роботи. Стандартні дії не викликають збоїв, а система демонструє стабільну роботу.

Продуктивність: основні запити та операції виконуються не довше ніж за 2 секунди. Система ефективно працює з базою даних обсягом до 10 000 записів.

Захист інформації: реалізовано механізми контролю доступу до даних (логіни, паролі, ролі). Дані зберігаються у захищеній СУБД з обмеженням доступу.

Юзабіліті: інтерфейс інтуїтивно зрозумілий і зручний навіть для користувачів без технічної підготовки. Основні функції доступні з мінімальною кількістю дій.

Документація та навчання: наявна повна документація - користувацька інструкція (у PDF та друкованому вигляді), технічні описи. Проведено навчання персоналу, надано консультації з налаштування та використання системи.

Приймально-здавальні випробування: розробник і замовник проводять спільне тестування за заздалегідь узгодженим сценарієм. Виявлені недоліки фіксуються та усуваються до фінального приймання.

Після перевірки всіх умов складається акт приймання-передачі системи, який підписують обидві сторони — розробник і замовник. Цей акт є юридичним підтвердженням готовності ІС до впровадження у промислову експлуатацію. програмного забезпечення, що є підставою для впровадження системи у постійну експлуатацію.

1.4 Постановка задачі

1.4.1 Загальні відомості

В умовах розвитку електронної комерції все більше підприємств прагнуть автоматизувати свої бізнес-процеси. Одним з ефективних рішень для досягнення цієї мети є створення інформаційної системи, яка дозволяє централізовано керувати даними, взаємодіяти з клієнтами, обробляти замовлення та аналізувати діяльність компанії. Система «MuzSvit» має на меті підтримку діяльності інтернет-магазину музичних інструментів, забезпечуючи

ефективне управління асортиментом, обслуговуванням клієнтів та виконанням замовлень.

1.4.2 Призначення і мета створення ІС інтернет-магазину музичних інструментів "MuzSvit"

Призначення ІС — автоматизація операцій інтернет-магазину, включаючи управління каталогом продукції, обробку замовлень, ведення клієнтської бази та формування звітів.

Метою створення системи є:

- підвищення ефективності внутрішніх бізнес-процесів;
- зменшення ручної роботи персоналу;
- покращення обслуговування клієнтів завдяки швидкому доступу до інформації;
- забезпечення прозорості операцій і зберігання історії дій.

1.4.3 Вимоги до ІС "MuzSvit"

Загальні вимоги до інформаційної системи:

- Забезпечення доступності системи через браузер без потреби встановлення локального ПЗ.
- Надійна авторизація користувачів з розмежуванням прав доступу.
- Висока швидкодія при обробці даних і виконанні запитів.
- Адаптивний дизайн інтерфейсу для роботи з мобільних пристроїв.
- Захист персональних даних користувачів.

Інформаційна система повинна включати такі функціональні можливості:

- Реєстрація та авторизація користувачів.
- Перегляд товарів, фільтрація за категоріями та брендами.
- Додавання товарів до кошика та оформлення замовлення.
- Автоматичне збереження історії покупок.
- Управління товарами з боку адміністратора (додавання, редагування, видалення).

- Перегляд замовлень та їх статусів в адмін-панелі.
- Генерація звітів про продажі, залишки на складі, популярність товарів.

Інформаційна система повинна працювати з наступними групами даних:

- Товари: назва, опис, ціна, категорія, бренд, кількість на складі, зображення.
- Користувачі: ім'я, електронна пошта, телефон, адреса, роль (користувач/адміністратор).
- Замовлення: дата, склад замовлення, контактна інформація клієнта, статус, сума.
- Звіти: дані про обсяги продажів, залишки, популярність товарів.

Вся інформація має зберігатися у структурованій формі у реляційній базі даних (MySQL), з дотриманням норм цілісності та уніфікації. Доступ до даних має бути обмежений відповідно до ролей користувачів.

1.4.4 Перспективи використання та подальшого розвитку ІС «MuzSvit»

Інформаційна система інтернет-магазину «MuzSvit» спроектована з урахуванням можливості подальшої модернізації, масштабування та адаптації під змінні бізнес-умови. Очікується, що після впровадження основного функціоналу система не лише покриватиме поточні потреби підприємства, але й стане основою для стратегічного розвитку.

Можливі напрями розвитку: — Інтеграція з CRM-системами для покращення взаємодії з клієнтами, управління контактами, угодами та маркетинговими кампаніями.

— Інтеграція з 1С або ERP для повної автоматизації бухгалтерського обліку, документообігу та управління закупівлями.

— Розширення мультимовного інтерфейсу з метою виходу на ринки інших країн.

— Мобільна версія або застосунок для зручного керування з планшетів або смартфонів адміністратора.

— Підключення до нових платіжних систем (Apple Pay, Google Pay, Stripe тощо) для розширення платіжних можливостей.

— Автоматичні рекомендації для клієнтів на основі історії покупок, переглядів, рейтингу — з використанням базових алгоритмів машинного навчання.

— Інтеграція онлайн-чату та консультаційних сервісів для підвищення конверсії на сайті та якості підтримки.

— Підтримка промо-кодів та гнучких знижок, управління рекламними кампаніями безпосередньо з адмін-панелі.

Таким чином, система «MuzSvit» не обмежується статичним функціоналом, а має потенціал до динамічного розвитку відповідно до стратегічних цілей бізнесу. Її модульна структура дозволяє гнучко впроваджувати нові компоненти без суттєвого впливу на існуючу архітектуру, забезпечує легке розширення функціональності без значних витрат часу чи ресурсів, створена як гнучка та адаптивна основа для майбутнього розвитку.

1.5 Вибір засобів та середовища розробки

Для розробки інформаційної системи інтернет-магазину «MuzSvit» обрано сучасні, вільно розповсюджувані технології, які забезпечують надійність, масштабованість і високу продуктивність.

Основні інструменти:

- Мова програмування (бекенд): JavaScript (Node.js) — забезпечує гнучкість та високу продуктивність обробки запитів.
- Фреймворк (бекенд): Express.js — мінімалістичний фреймворк для побудови RESTful API.
- Мова програмування (фронтенд): JavaScript (React) — бібліотека для створення динамічного, компонентного користувацького інтерфейсу.

- СУБД: MySQL — реляційна база даних з підтримкою транзакцій та надійного зберігання.
- Інтерфейс користувача: Tailwind CSS — утилітарна CSS-бібліотека для швидкої адаптивної верстки.
- Середовище розробки: Visual Studio Code — легке й функціональне середовище для написання коду.
- Проектування UI: Draw.io — інструмент для створення блок-схем, макетів інтерфейсів.

Вибір цих інструментів обумовлений відкритістю та безкоштовністю ліцензій; активною підтримкою спільноти та великою кількістю документації; можливістю розгортання як на локальному сервері, так і у хмарі.

Такий стек технологій дозволяє ефективно реалізувати повнофункціональну інформаційну систему з подальшою можливістю розширення та інтеграції з іншими сервісами.

ВИСНОВКИ:

У першому розділі було проведено комплексний аналіз предметної області, визначено потреби та особливості діяльності інтернет-магазину «MuzSvit», обґрунтовано необхідність впровадження інформаційної системи. Здійснено формулювання вимог до системи, описано її функціональні та технічні особливості, запропоновано концепцію автоматизованого робочого місця адміністратора та визначено етапи впровадження.

Запропонована система покликана оптимізувати управління товарами, замовленнями, користувачами та звітністю, забезпечити стабільну роботу за рахунок клієнт-серверної архітектури та сучасного технологічного стеку.

Обґрунтовано технічну доцільність, економічну вигідність і перспективи подальшого розвитку ІС. Усі прийняті рішення враховують специфіку галузі, потреби персоналу та клієнтів, а також можливості масштабування системи в майбутньому.

Таким чином, результати першого розділу створюють фундамент для подальшої розробки конкретних проєктних рішень у наступному розділі.

2 РОЗРОБКА ПРОЄКТНИХ РІШЕНЬ

2.1 Структура інформаційної системи

Інформаційна система інтернет-магазину "MuzSvit" побудована на модульному принципі, що дозволяє розподілити функціональність за логічними блоками та забезпечити гнучкість при розробці, супроводі та подальшому масштабуванні. Кожен модуль відповідає за певний набір операцій і взаємодіє з іншими через визначені інтерфейси обміну даними.

Основна структура системи включає такі модулі:

— Модуль управління товарами — відповідає за додавання, редагування, видалення товарів, зберігання характеристик, зображень, залишків на складі та категорій.

— Модуль управління користувачами — забезпечує реєстрацію, авторизацію, розмежування доступу (звичайні користувачі та адміністратори), оновлення профілів, збереження історії активності.

— Модуль замовлень — відповідає за обробку замовлень, зміну їх статусів, формування рахунків, інтеграцію з платіжними системами та службами доставки.

— Модуль звітності — генерує звіти щодо продажів, залишків на складі, замовлень за період, найпопулярніших товарів.

— Модуль повідомлень — надсилає сповіщення клієнтам (лист-підтвердження, зміна статусу замовлення) та адміністраторам (нові замовлення, мінімальний залишок товару).

— Модуль інтеграції — надає можливість взаємодії з зовнішніми API: служби доставки (Нова Пошта, Укрпошта), платіжні шлюзи (LiqPay, WayForPay), аналітичні платформи (Google Analytics).

Усі модулі працюють у межах клієнт-серверної архітектури з централізованим зберіганням даних у базі MySQL. Комунікація між клієнтською частиною (React.js) та сервером (Node.js + Express) здійснюється через RESTful API.

Використання подібної структурної моделі дозволяє не лише забезпечити логічну цілісність ІС, але й створити зручне середовище для подальшого розвитку системи: впровадження нових функцій, оптимізацію існуючих процесів, підтримку багатокористувацької роботи з розмежуванням прав.

2.2 Інформаційне забезпечення системи

Інформаційне забезпечення ІС «MuzSvit» охоплює організацію збору, зберігання, передачі та обробки всіх даних, необхідних для ефективного функціонування системи. На етапі проєктування було прийнято рішення використовувати реляційну модель даних, реалізовану в СУБД MySQL, що дозволяє чітко структурувати інформацію, підтримувати цілісність та забезпечувати швидкий доступ до потрібних записів.

У структурі системи передбачено такі основні інформаційні об'єкти:

- Товари: мають розширений опис, категоризацію, прив'язку до бренду, ціну, кількість та медіа-матеріали.
- Користувачі: включають імена, контактні дані, ролі та історію активності.
- Замовлення: містять дані про клієнта, список товарів, статус, дату, підсумкову суму.
- Додаткові сутності: бренди, категорії, працівники — служать для довідкового та адміністративного управління.

Інформація зберігається у таблицях бази даних, пов'язаних між собою зовнішніми ключами, що забезпечує логічну цілісність і відповідність вимогам до нормалізації.

Захист інформації реалізується на кількох рівнях:

- Хешування паролів (bcrypt) та авторизація з використанням токенів (JWT);
- Розмежування прав доступу між ролями (користувач/адміністратор);
- Обмеження запитів та валідація даних на бекенді;

— Обмеження доступу до конфіденційної інформації через REST API згідно з авторизованими сесіями.

Зображення товарів та інші мультимедійні ресурси не зберігаються безпосередньо у базі, а лише прив'язуються через URL-шляхи, що дозволяє уникнути надмірного навантаження на сховище.

Таке інформаційне забезпечення гарантує структурованість, достовірність і актуальність даних, що обробляються системою, а також підтримує масштабованість і розвиток у майбутньому.

2.2.1 Визначення сутностей та концептуальна модель даних

Першим етапом побудови концептуальної моделі даних є визначення основних сутностей, які відіграють ключову роль у функціонуванні інформаційної системи магазину «MuzSvit». Кожна сутність характеризується певним набором атрибутів, а між сутностями встановлюються логічні зв'язки.

Виділені сутності:

— Клієнт (Customer):

CustomerID (PK) — унікальний ідентифікатор клієнта

FullName — повне ім'я

Email — електронна пошта

Phone — номер телефону

Address — адреса доставки

RegistrationDate — дата реєстрації

user_id (FK) — зв'язок із таблицею users

— Користувач системи (User):

id (PK) — унікальний ідентифікатор користувача

username — логін

email — електронна адреса

password — хешований пароль

isAdmin — ознака адміністратора

— Категорія інструменту (Category):

CategoryID (PK) — унікальний ідентифікатор категорії

CategoryName — назва категорії

ImagePath — шлях до зображення категорії

— Бренд (Brand):

BrandID (PK) — унікальний ідентифікатор бренду

BrandName — назва бренду

— Товар (Instrument):

InstrumentID (PK) — унікальний ідентифікатор інструменту

Name — назва інструменту

CategoryID (FK) — зв'язок із категорією

BrandID (FK) — зв'язок із брендом

Price — ціна

QuantityInStock — кількість на складі

Description — опис товару

Photo — зображення у форматі BLOB

— Замовлення (Order):

OrderID (PK) — унікальний ідентифікатор замовлення

CustomerID (FK) — клієнт, який оформив замовлення

OrderDate — дата оформлення

TotalAmount — загальна сума

Status — статус замовлення

— Позиція замовлення (OrderItem):

OrderItemID (PK) — унікальний ідентифікатор позиції

OrderID (FK) — зв'язок із замовленням

InstrumentID (FK) — товар, що входить до замовлення

Quantity — кількість одиниць

Price — ціна на момент замовлення

Між сутностями встановлено такі зв'язки:

— Один користувач (User) може мати один профіль клієнта (Customer);

— Один клієнт може оформити багато замовлень (Orders);

— Кожне замовлення може містити багато товарів (OrderItems);

- Кожна позиція замовлення відповідає певному товару (Instrument);
- Товар належить до однієї категорії (Category) та бренду (Brand).

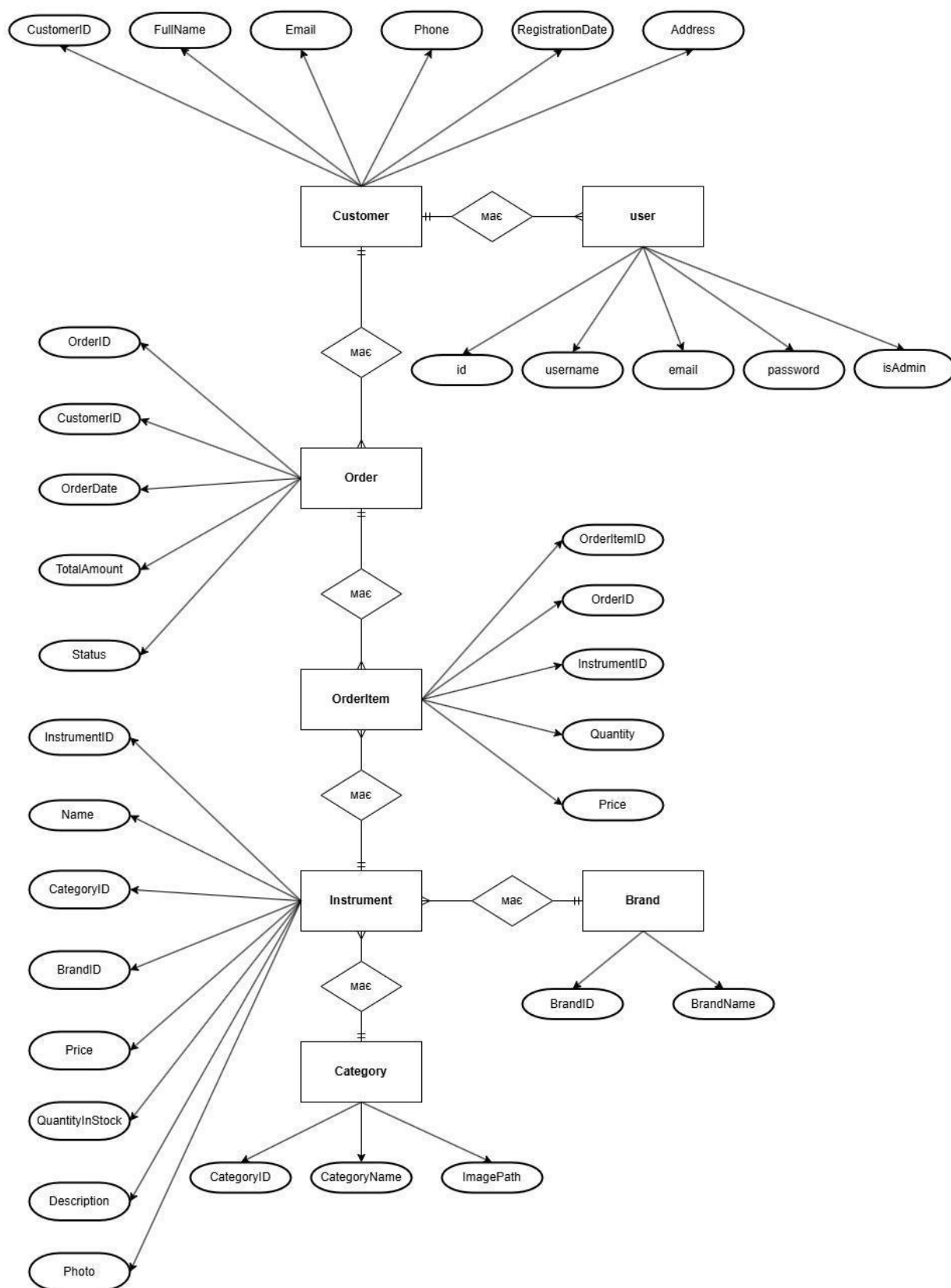


Рисунок 2.1 – Концептуальна модель БД

2.2.2 Розробка логічної моделі даних

На основі визначених сутностей і зв'язків між ними сформуємо ER-діаграму логічної моделі, яка ілюструє взаємозалежності в базі даних. Діаграма включає ключові таблиці та типи зв'язків між ними: один-до-багатьох, багато-до-одного тощо. Вона дозволяє візуально представити логіку структури бази даних та підтвердити її відповідність вимогам бізнес-логіки.

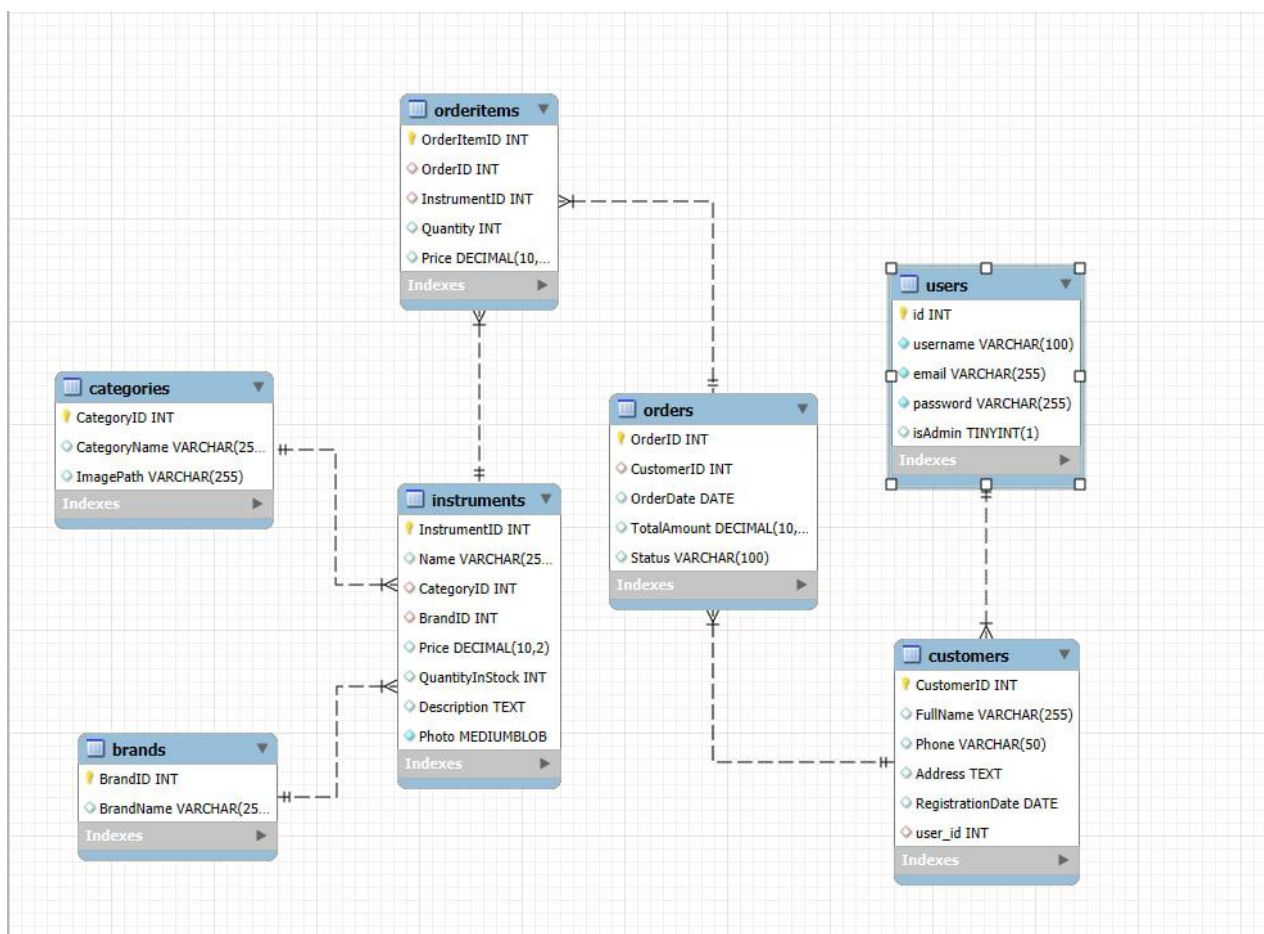


Рисунок 2.2 – ER-діаграма логічної моделі

2.2.3 Розробка фізичної моделі даних

Розробимо фізичну структуру кожної таблиці в базі даних.

Таблиця 2.1 - Customer (Клієнт)

Поле	Тип даних	Опис
CustomerID	INT (PK)	Унікальний ідентифікатор клієнта
FullName	VARCHAR	Повне ім'я
Email	VARCHAR	Електронна пошта
Phone	VARCHAR	Номер телефону
Address	TEXT	Адреса доставки
RegistrationDate	DATE	Дата реєстрації
user_id	INT (FK)	Зв'язок із таблицею users

Таблиця 2.2 - User (Користувач системи)

Поле	Тип даних	Опис
id	INT (PK)	Унікальний ідентифікатор користувача
username	VARCHAR	Логін
email	VARCHAR	Електронна адреса
password	VARCHAR	Хешований пароль
isAdmin	BOOLEAN	Ознака адміністратора

Таблиця 2.3 - Category (Категорія інструменту)

Поле	Тип даних	Опис
CategoryID	INT (PK)	Унікальний ідентифікатор категорії
CategoryName	VARCHAR	Назва категорії
ImagePath	VARCHAR	Шлях до зображення категорії

Таблиця 2.4 - Brand (Бренд)

Поле	Тип даних	Опис
BrandID	INT (PK)	Унікальний ідентифікатор бренду
BrandName	VARCHAR	Назва бренду

Таблиця 2.5 - Instrument (Товар)

Поле	Тип даних	Опис
InstrumentID	INT (PK)	Унікальний ідентифікатор інструменту
Name	VARCHAR	Назва інструменту
CategoryID	INT (FK)	Зв'язок із категорією
BrandID	INT (FK)	Зв'язок із брендом
Price	DECIMAL	Ціна
QuantityInStock	INT	Кількість на складі
Description	TEXT	Опис товару
Photo	BLOB	Зображення у форматі BLOB

Таблиця 2.6 - Order (Замовлення)

Поле	Тип даних	Опис
OrderID	INT (PK)	Унікальний ідентифікатор замовлення
CustomerID	INT (FK)	Клієнт, який оформив замовлення
OrderDate	DATE	Дата оформлення
TotalAmount	DECIMAL	Загальна сума
Status	VARCHAR	Статус замовлення

Таблиця 2.7 - OrderItem (Позиція замовлення)

Поле	Тип даних	Опис
OrderItemID	INT (PK)	Унікальний ідентифікатор позиції
OrderID	INT (FK)	Зв'язок із замовленням
InstrumentID	INT (FK)	Товар, що входить до замовлення
Quantity	INT	Кількість одиниць
Price	DECIMAL	Ціна на момент замовлення

Зв'язки між таблицями:

users.id → customers.user_id

customers.CustomerID → orders.CustomerID

orders.OrderID → orderitems.OrderID

orderitems.InstrumentID → instruments.InstrumentID

instruments.CategoryID → categories.CategoryID

instruments.BrandID → brands.BrandID

2.3 Вибір алгоритмів та обробка даних

На етапі проєктування інформаційної системи інтернет-магазину «MuzSvit» було здійснено попередній аналіз вимог до швидкодії, безпеки та гнучкості. На основі цих вимог приймаються ключові рішення щодо алгоритмів обробки даних та вибору відповідного програмного забезпечення для реалізації системи.

Алгоритми обробки замовлень (буде реалізовано): — Перевірка наявності товарів на складі під час оформлення замовлення; — Автоматичне формування записів у таблицях Orders та OrderItems; — Розрахунок

підсумкової суми за всіма позиціями; — Ініціалізація замовлення зі статусом "Нове" та можливістю подальшого оновлення.

Пошук, сортування та фільтрація: — Передбачено впровадження SQL-запитів з підтримкою умов WHERE, LIKE, ORDER BY, LIMIT для реалізації пошуку товарів, фільтрації за категоріями, брендами та ціною.

Генерація звітів: — Заплановані запити на основі агрегатних функцій SUM, COUNT, GROUP BY для формування статистичних та аналітичних звітів про замовлення, залишки на складі, популярність товарів.

Захист даних: — Паролі користувачів будуть зберігатися у зашифрованому вигляді за допомогою алгоритму bcrypt; — Авторизація користувачів передбачається з використанням токенів JWT; — Ролі користувачів (адміністратор, клієнт) дозволять розмежувати доступ до функціоналу системи через middleware-рішення.

Обробка зображень: планується зберігання зображень товарів у таблиці Instruments для оперативного виведення в інтерфейсі; додаткові зображення (категорії, банери, логотипи) зберігатимуться у файловій системі або хмарному сховищі, а в БД — лише шлях (URL).

Такий підхід дозволяє: уникати перевантаження бази даних великими файлами, спростити обробку зображень, прискорити завантаження сторінок для користувачів.

Передбачається використання multer для завантаження зображень із перевіркою типу файлу та обмеженням розміру.

Обрані технології для реалізації системи:

- Node.js + Express.js для реалізації серверної частини — забезпечують гнучку обробку запитів, middleware і зв'язок із базою;

- React.js як клієнтська частина для створення сучасного, динамічного та адаптивного інтерфейсу;

- MySQL як основна реляційна СУБД з підтримкою транзакцій, індексів, цілісності та зручного масштабування;

- JWT для реалізації токен-базованої авторизації;

- bcryptjs для хешування паролів;
- express-validator — для перевірки вхідних даних;
- Multer — для обробки файлів при завантаженні.

Використання зазначених інструментів дозволить створити надійну, безпечну та масштабовану інформаційну систему, адаптовану до умов онлайн-торгівлі. Крім того, застосування Tailwind CSS на фронтенді забезпечить швидку адаптивну верстку та високу продуктивність інтерфейсу користувача завдяки утилітарному підходу до стилізації.

ВИСНОВКИ:

У другому розділі було представлено ключові технічні рішення, що лягли в основу розробки інформаційної системи інтернет-магазину «MuzSvit». Описано структуру системи на модульному рівні, що забезпечує чітке розмежування функцій між окремими компонентами, такими як управління товарами, користувачами, замовленнями, звітністю, повідомленнями та зовнішніми інтеграціями.

Ретельно обґрунтовано інформаційне забезпечення системи, яке базується на реляційній моделі даних у СУБД MySQL. Побудовано концептуальну, логічну та фізичну моделі БД, що демонструють повну відповідність бізнес-логіці магазину. Таблиці пов'язані зовнішніми ключами, що гарантує логічну цілісність і можливість ефективної обробки даних.

У рамках обробки даних описано передбачувані алгоритми для роботи з замовленнями, пошуку, сортування, генерації звітів, захисту інформації та обробки зображень. Обґрунтовано вибір відповідного програмного забезпечення та технологій: Node.js, Express, React, MySQL, JWT, bcrypt, Tailwind CSS, Multer та ін.

Таким чином, розділ закладає чітку технічну основу для реалізації інформаційної системи: від структурної побудови до деталей обробки даних, забезпечуючи масштабованість, безпеку та продуктивність, необхідні для ефективного функціонування сучасного інтернет-магазину.

3 РЕАЛІЗАЦІЯ ПРОЄКТУ ІНФОРМАЦІЙНОЇ СИСТЕМИ

При постановці задачі визначені вимоги до документації. Відповідно до цих вимог для користувачів та адміністраторів системи підготовлено робочу документацію, яка містить детальні інструкції щодо всіх функцій системи та порядку їх використання. Повну інформацію про використання системи можна знайти у додатку (А). Переглянути код....

Інформаційну систему створено з фокусом на інтуїтивній взаємодії та ефективності процесу покупки. Користувацький інтерфейс побудований на React із використанням Tailwind CSS, що гарантує швидке завантаження та адаптивність для будь-яких екранів. Інтелектуальний пошук і система фільтрації дозволяють клієнтам легко знаходити товари за назвою, категорією, брендом чи ціною.

Ключовим елементом дизайну є карусель популярних категорій із плавними анімаціями, яка одразу привертає увагу й спрощує навігацію. Картки товарів підвантажуються динамічно, а модальне вікно корзини дає змогу переглянути вибрані інструменти, змінити їх кількість та побачити загальну суму.

API розподілена за функціональними модулями: працює просте і зрозуміле керування товарами, замовленнями та профілями користувачів. Адміністратор має доступ до панелі, де одним кліком створює чи редагує категорії, бренди та інструменти, завантажуючи зображення з попереднім переглядом.

Для захисту даних застосовано шифрування паролів, перевірку авторизації через JWT-токени і обмеження кількості запитів, що запобігає зловисним атакам. Система логування фіксує помилки та події, допомагаючи швидко знаходити й усувати проблеми.

Після підтвердження замовлення система підставляє автозаповнені контактні дані користувача і дозволяє обрати спосіб доставки з відображенням остаточної вартості. Корзина та історія замовлень тісно інтегровані з

особистим кабінетом: після оформлення покупки всі дані зберігаються в профілі, де клієнт бачить статус замовлення і може переглянути деталі.

Додатковою перевагою є можливість перемикання інтерфейсу на темну тему, що підвищує комфорт роботи в умовах низького освітлення.

Усі основні модулі системи пройшли комплексну перевірку: ручне тестування підтвердило коректність роботи форм і валідації, надійність обміну даними між клієнтом та сервером, коректну інтеграцію зі сторонніми сервісами та повноту запису в базу даних.

Результати апробації показали, що система досягає поставлених завдань: вона значно спрощує пошук та оформлення замовлень, підвищує зручність користувацького досвіду й відповідає вимогам проєкту.

Головна сторінка веб-додатку інформаційної системи представлена на рисунку 3.1. Інші форми на рисунках 3.2 – 3.9

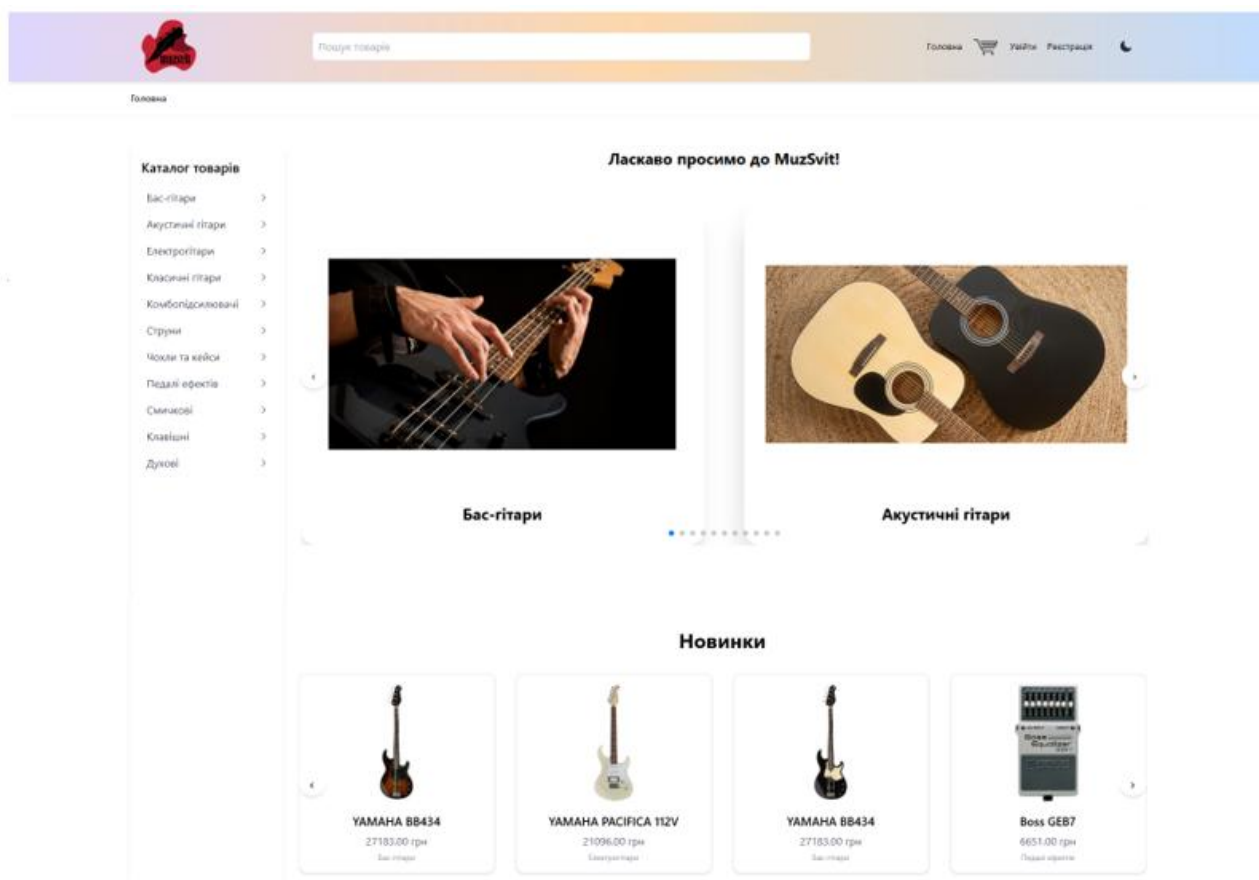


Рисунок 3.1 – Головна форма веб-додатку

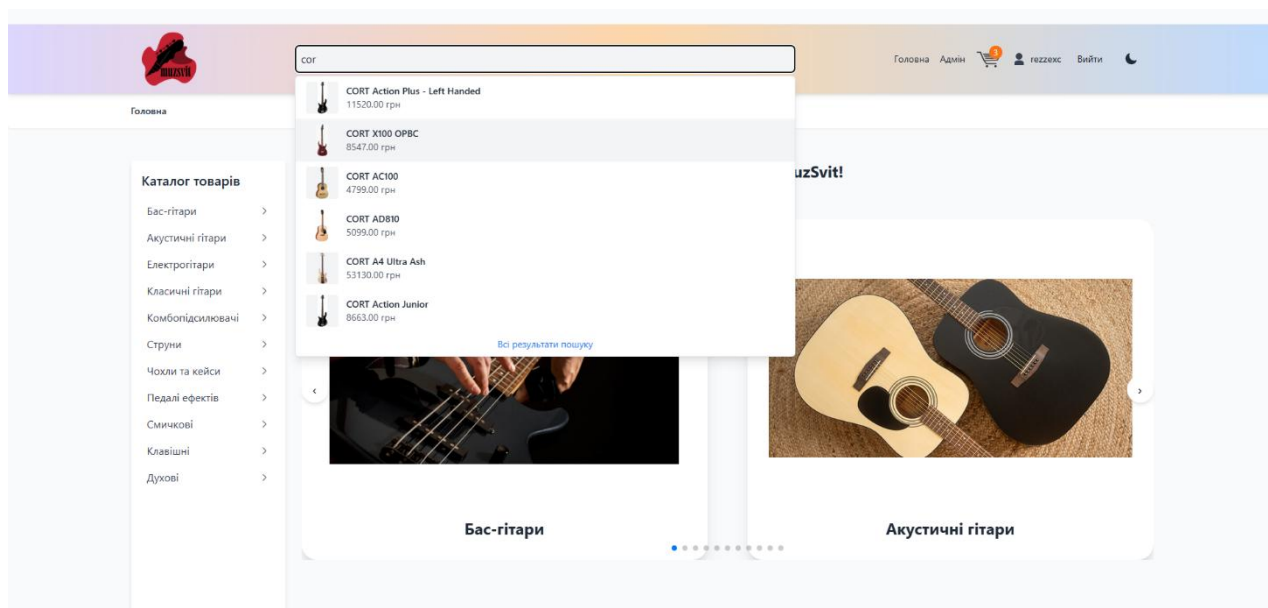


Рисунок 3.2 – Поле пошуку товара

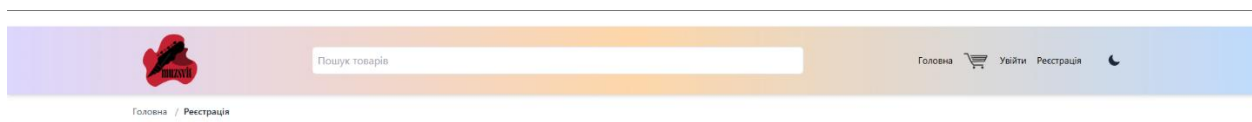


Рисунок 3.3 – Реєстрація користувача

Кошик

✕

☒



CORT Action Plus - Left Handed

Продавець: Cort

-

1

+

11520.00 грн

⋮

☒



CORT X100 OPBC

Продавець: Cort

-

1

+

8547.00 грн

⋮

☒



D'Addario EJ10 80/20 Bronze Extra Light

Продавець: D'Addario

-

3

+

369.00 грн

⋮

☐



CORT AD810

Продавець: Cort

-

1

+

5099.00 грн

⋮

21 174 грн

Оформити замовлення

Рисунок 3.5 - Кошик

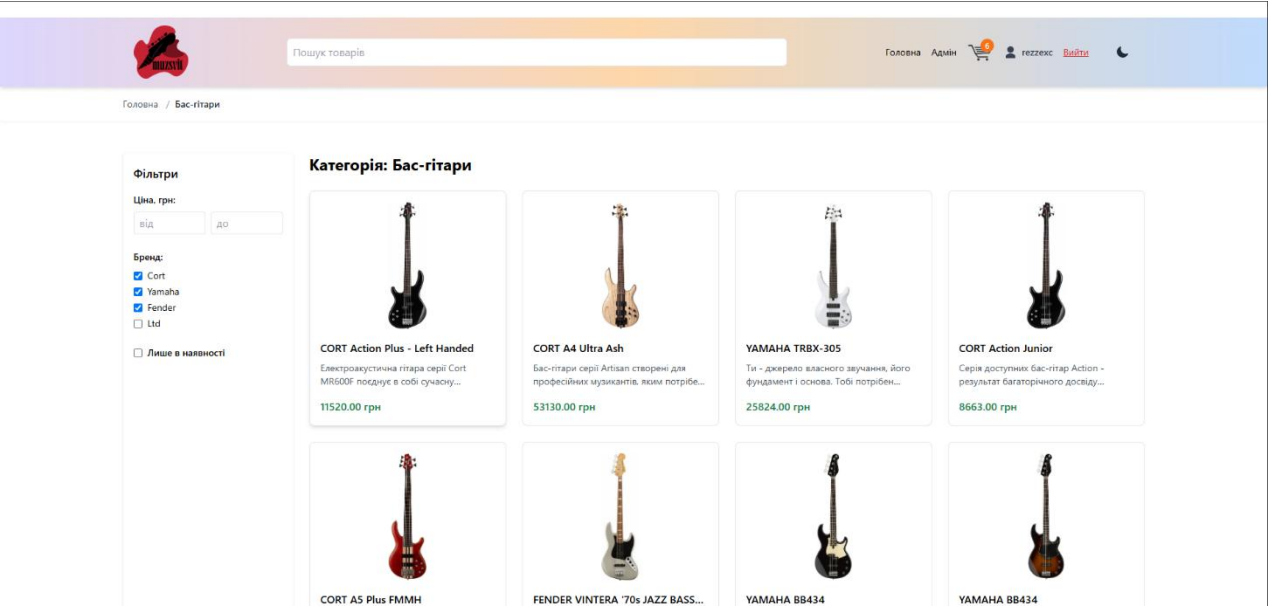


Рисунок 3.6 – Вибір товару за категоріями

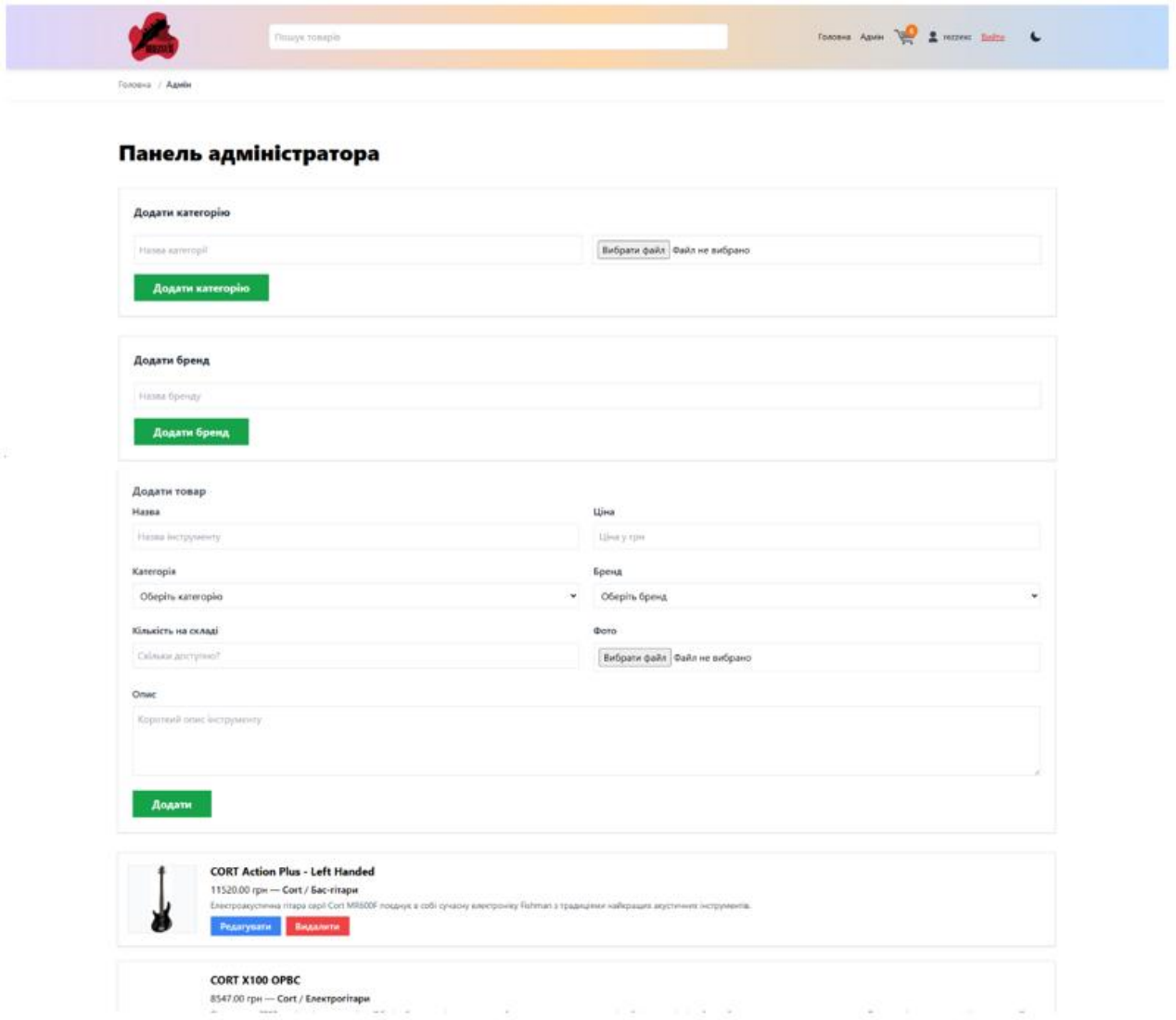
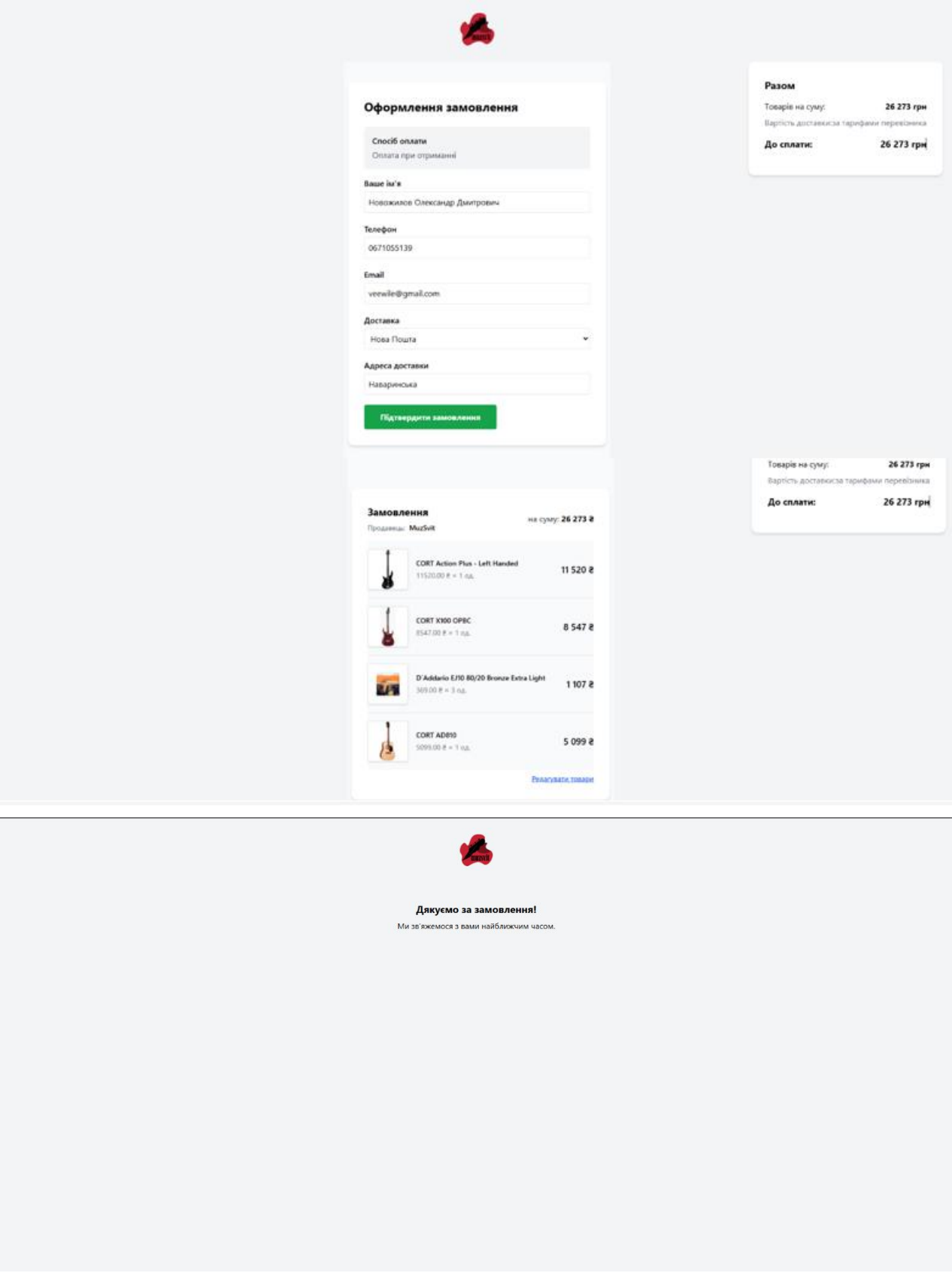


Рисунок 3.7 – Панель адміністратора



Оформлення замовлення

Спосіб оплати
Оплата при отриманні

Ваше ім'я
Новожилів Олександр Дмитрович

Телефон
0671055139

Email
vevile@gmail.com

Доставка
Нова Пошта

Адреса доставки
Наваринська

Підтвердити замовлення

Замовлення на суму: 26 273 ₴
Продавець: MuzSvit

	CORT Action Plus - Left Handed 11520.00 ₴ × 1 од.	11 520 ₴
	CORT X100 OPBC 8547.00 ₴ × 1 од.	8 547 ₴
	D'Audario EJ10 80/20 Bronze Extra Light 369.00 ₴ × 3 од.	1 107 ₴
	CORT AD810 5099.00 ₴ × 1 од.	5 099 ₴

[Редагувати товари](#)

Разом
Товарів на суму: 26 273 грн
Вартість доставки та тарифи перевізника
До сплати: 26 273 грн

Товарів на суму: 26 273 грн
Вартість доставки та тарифи перевізника
До сплати: 26 273 грн

Дякуємо за замовлення!
Ми зв'яжемося з вами найближчим часом.

Рисунок 3.8 – Оформлення замовлення

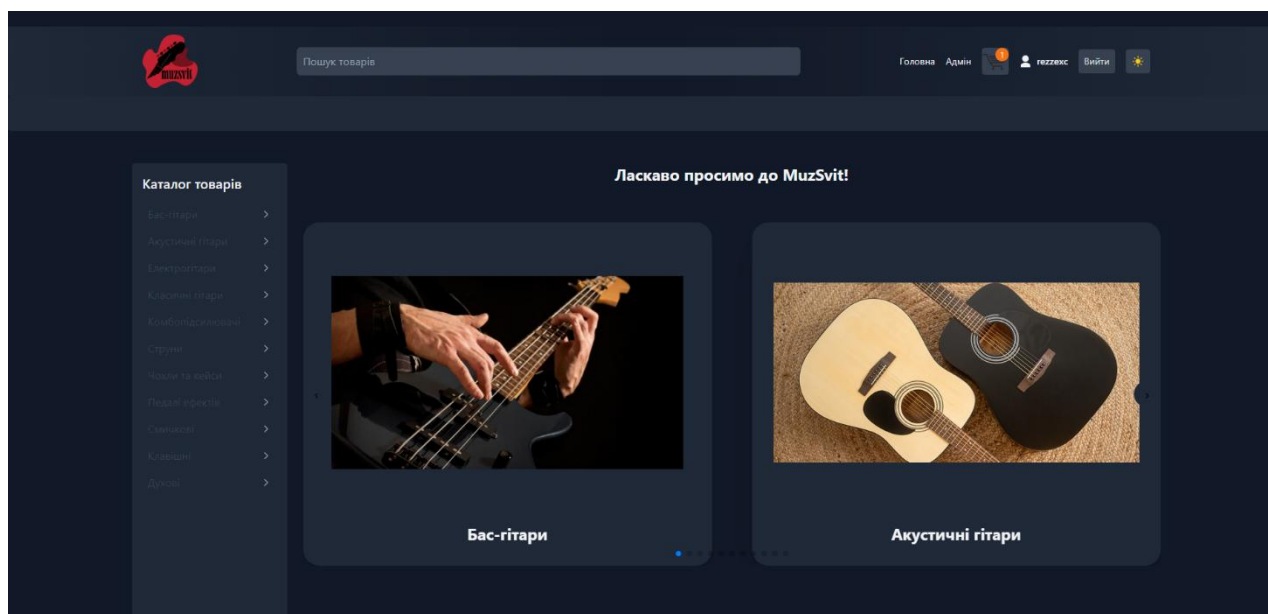


Рисунок 3.9 - Вибір теми

ВИСНОВОК: Інформаційна система готова до експлуатації та підтримки: вона забезпечує стабільну роботу, легко масштабується й відповідає вимогам як користувачів, так і адміністраторів. Завдяки компактному дизайну й зрозумілій логіці взаємодії, подальший розвиток проєкту та інтеграція нових сервісів будуть здійснюватися швидко та без складнощів.

4. ОХОРОНА ПРАЦІ

4.1 Вступ

Охорона праці — один із ключових аспектів, що мають враховуватись під час організації діяльності в будь-якій сфері, зокрема в галузі інформаційних технологій. Незважаючи на те, що робота фахівців з розробки програмного забезпечення не супроводжується безпосереднім контактом із небезпечними механізмами або фізичними об'єктами підвищеної небезпеки, вона пов'язана з низкою чинників, які за певних умов можуть призводити до погіршення стану здоров'я та зниження працездатності.

Зокрема, працівники, що беруть участь у розробці, тестуванні, впровадженні та супроводі інформаційних систем, більшу частину робочого часу проводять у закритих приміщеннях, у малорухомому положенні, працюючи з комп'ютерами та іншою офісною технікою. Їхня діяльність відзначається високим рівнем зорового, розумового та психоемоційного навантаження, що, за відсутності належних умов праці, може спричинити розвиток хронічних захворювань, порушення постави, зору, нервової системи тощо.

Особливе значення охорона праці набуває в умовах довготривалої роботи над проєктами, коли програмісти, веб-дизайнери, тестувальники, адміністратори баз даних та інші учасники команди працюють у режимі підвищеної інтенсивності, стикаючись з дедлайнами, технічно складними задачами, необхідністю швидкої адаптації до змін у проєкті.

У зв'язку з цим, у процесі реалізації інформаційної системи інтернет-магазину «MuzSvit» надзвичайно важливо закласти основи безпечного, комфортного та здорового робочого середовища. Це включає як дотримання вимог ергономіки та санітарно-гігієнічних норм, так і забезпечення психологічного добробуту працівників, їх інформованості щодо можливих професійних ризиків та засобів їх попередження.

Забезпечення високого рівня охорони праці сприяє не лише збереженню здоров'я працівників, але й покращенню загальної атмосфери в команді, зниженню рівня стресу та професійного вигорання, зменшенню кількості помилок і, як наслідок, підвищенню якості створюваного програмного продукту. Тому питання охорони праці повинні розглядатися не як формальність, а як один із основоположних факторів стабільної та ефективної роботи команди розробників.

4.2 Аналіз небезпечних і шкідливих виробничих факторів

Для працівників галузі інформаційних технологій, особливо тих, що залучені до процесу розробки та обслуговування веб-застосунків, характерні певні фактори ризику, які хоч і не є прямими загрозами життю, однак можуть істотно впливати на здоров'я при тривалому впливі. Аналіз таких факторів є необхідним для подальшого формування комплексу захисних заходів.

Одним із найбільш значущих факторів є психоемоційне навантаження. Це обумовлено тим, що працівники постійно стикаються з необхідністю концентрації уваги, прийняття швидких рішень, відповідальністю за результати роботи, роботою в умовах частих дедлайнів та змін. Ці умови можуть викликати тривалий стрес, який призводить до психологічного вигорання, розвитку нервових розладів, зниження імунітету та загального погіршення стану здоров'я.

Значну роль відіграє і високе зорове навантаження. Тривала робота з монітором комп'ютера спричиняє значне напруження очних м'язів, що може призводити до короткозорості, сухості очей, синдромів комп'ютерного зору, головних болей, а також до серйозних порушень зору, якщо не вживати належних заходів профілактики.

Фізична статичність також є серйозною загрозою для працівників, які довго знаходяться в сидячому положенні. Недостатня рухливість та неправильне облаштування робочого місця може викликати проблеми з поставою, захворювання хребта, порушення кровообігу, що у перспективі значно впливає на працездатність і якість життя.

Не менш важливими є мікрокліматичні умови, що включають в себе параметри температури, вологості, вентиляції приміщень. Невідповідність цих показників нормативним вимогам призводить до погіршення загального самопочуття, зниження концентрації уваги та працездатності, алергічних реакцій, захворювань органів дихання.

Шумове навантаження є додатковим фактором, що знижує комфортність роботи. Постійні звуки вентиляторів комп'ютерів, кондиціонерів, галас офісного приміщення можуть створювати постійний дискомфорт, призводити до втоми, зниження концентрації, головних болей та погіршення загального самопочуття.

Недостатнє або неправильне освітлення може значно підсилювати зорове навантаження, приводити до напруження зору, розвитку головних болей та підвищеної стомлюваності. Важливо, щоб освітлення було рівномірним, достатньої інтенсивності і не створювало зайвих відблисків на моніторах.

Електромагнітне випромінювання, хоч і не є критично високим у сучасній техніці, проте при тривалому впливі може негативно впливати на здоров'я працівників, сприяючи швидкій втомі та зниженню імунітету.

Також не слід забувати про вплив закритого робочого простору, в якому працівники проводять більшу частину свого робочого часу. Недостатнє природне освітлення, одноманітність інтер'єру, відсутність періодичних змін обстановки можуть призводити до психологічної втоми, зниження мотивації, погіршення загального психоемоційного стану.

Аналізуючи ці фактори, стає очевидною необхідність комплексного підходу до організації робочого процесу з метою створення максимально безпечних та комфортних умов праці.

4.3 Розробка заходів щодо забезпечення сприятливих умов праці при роботі з персональним комп'ютером

З метою забезпечення сприятливих умов праці працівників, що працюють з персональним комп'ютером, необхідно впровадити комплекс заходів, спрямованих на зниження впливу шкідливих факторів.

Одним із ключових аспектів є організація ергономічного робочого місця. Для цього рекомендується використовувати спеціалізовані меблі, що відповідають ергономічним стандартам. Стілець має бути регульованим за висотою та забезпечувати підтримку попереку. Робочий стіл також повинен мати регульовану висоту і достатньо простору для комфортного розміщення рук, клавіатури та монітора. Відстань від очей до екрана монітора повинна становити не менше 50-70 см, верхній край екрану має бути на рівні очей або трохи нижче.

Важливим є правильне налаштування монітора: оптимальною є яскравість екрану, що відповідає освітленню в приміщенні. Потрібно уникати прямого сонячного або штучного світла, що створює відблиски на екрані. Рекомендується використовувати монітори з високою роздільною здатністю та антибліковим покриттям.

Регулярні перерви під час роботи мають бути частиною робочого процесу. Короткі перерви (5-10 хвилин щогодини) дозволяють уникнути перевтоми очей і м'язів, допомагають підтримувати концентрацію та продуктивність. Під час цих перерв рекомендується виконувати легкі фізичні вправи для рук, шиї, спини та очей.

Забезпечення належного освітлення є важливою умовою комфортної роботи. Оптимальним є комбіноване освітлення, яке включає як природне, так і штучне світло. Освітленість робочого місця повинна відповідати нормативним значенням і бути рівномірною по всій поверхні робочого простору. Варто використовувати світильники з розсіяним світлом, які не створюють різких тіней та відблисків.

Необхідно створити комфортний мікроклімат на робочому місці. Температура в приміщенні має бути в межах 20-24 °C, вологість повітря — близько 40-60%. Для забезпечення належної циркуляції повітря та зменшення

рівня пилу рекомендується регулярне провітрювання приміщення, використання кондиціонерів з функцією очищення повітря.

Особливу увагу слід приділяти профілактиці психологічного вигорання працівників. Доцільно регулярно проводити тренінги з управління стресом, організовувати психологічну підтримку та консультації, заохочувати командну взаємодію та підтримку серед співробітників. Важливим є створення умов для неформального спілкування та відпочинку.

Питання шумового навантаження також потребує уваги. Для його зниження рекомендується використовувати комп'ютерну техніку з низьким рівнем шуму, облаштувати приміщення звукопоглинаючими матеріалами, застосовувати індивідуальні засоби захисту, такі як навушники з шумозаглушенням.

Для зниження впливу електромагнітного випромінювання необхідно розмішувати комп'ютери таким чином, щоб забезпечити достатню дистанцію між робочими місцями, регулярно проводити профілактичні огляди техніки та за можливості використовувати сучасне обладнання з низьким рівнем випромінювання.

Отже, комплексне впровадження зазначених заходів дозволить забезпечити сприятливі умови праці для співробітників, що працюють з персональним комп'ютером, підвищити їхню продуктивність, зберегти здоров'я та знизити ймовірність професійних захворювань.

ВИСНОВОК: У межах цього розділу було розглянуто основні небезпеки, що супроводжують роботу за комп'ютером, зокрема вплив тривалого сидіння, навантаження на зір та психоемоційне виснаження. Проведено аналіз характерних виробничих факторів та сформульовано комплекс заходів, що сприяють мінімізації негативного впливу робочого середовища на організм користувача. Особливу увагу приділено ергономіці робочого місця, режиму праці й відпочинку, а також належному мікроклімату в приміщенні. Запропоновані рекомендації дозволяють створити комфортні,

безпечні та відповідні нормативам умови праці для розробника інформаційної системи.

ВИСНОВКИ

У ході виконання дипломного проєкту було здійснено повний цикл створення інформаційної системи інтернет-магазину «MuzSvit»:

- Проведено глибинний аналіз предметної області та бізнес-процесів магазину, сформульовано функціональні й технічні вимоги.
- Спроектовано концептуальну, логічну та фізичну моделі даних, що забезпечують цілісність і швидкий доступ до інформації.
- Розроблено модульну архітектуру програмного забезпечення з чітким розподілом функцій між клієнтською частиною (пошук, фільтрація, корзина, темна тема) та серверною (зберігання даних, авторизація, логування).
- Реалізовано інтуїтивний інтерфейс для користувачів і зручну адмін-панель для керування товарами, категоріями та замовленнями.
- Запроваджено базові механізми захисту (шифрування паролів, JWT-токени, обмеження запитів) та систему валідації даних.
- Проведено ручне тестування ключових сценаріїв, що підтвердило стабільність, коректність обробки даних і відповідність проєктним цілям.
- Підготовлено детальну документацію для користувачів і адміністраторів, що спрощує впровадження та подальшу підтримку системи.

Отримана інформаційна система готова до експлуатації: вона оптимізує процеси пошуку, оформлення та обробки замовлень, підвищує зручність роботи як для клієнтів, так і для персоналу, та закладає міцну основу для масштабування й розширення функціоналу в майбутньому.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Офіційна документація React— “Getting Started”, React.js Documentation, [<https://reactjs.org/docs/getting-started.html>].
2. Офіційна документація Express.js— “Guide”, Express.js Documentation, [<https://expressjs.com/en/guide/>].
3. Офіційна документація MySQL— “MySQL Reference Manual”, Oracle, [<https://dev.mysql.com/doc/>].
4. Офіційна документація Tailwind CSS—“Documentation”, [Tailwindcss.com](https://tailwindcss.com), [<https://tailwindcss.com/docs>].
5. [JWT.IO](https://jwt.io)— “Introduction to JSON Web Tokens”, jwt.io, [<https://jwt.io/introduction/>].
6. OWASP Foundation— “Top 10 Web Application Security Risks”, OWASP, [<https://owasp.org/www-project-top-ten/>].
7. MDN Web Docs— “Using Fetch”, Mozilla Developer Network, [https://developer.mozilla.org/docs/Web/API/Fetch_API/Using_Fetch].
8. D. Flanagan, “JavaScript: The Definitive Guide”, O’Reilly Media, 7th Edition, 2020.
9. E. Freeman, E. Robson, B. Bates, K. Sierra, “Head First Design Patterns”, O’Reilly Media, 2nd Edition, 2020.
10. A. Freeman, J. Hartig, “Pro Spring Security”, Apress, 2021.
11. J. Jon Duckett, “HTML and CSS: Design and Build Websites”, Wiley, 2011.
12. M. Fischer, “Building Microservices: Designing Fine-Grained Systems”, O’Reilly Media, 2015.
13. I. Sommerville, “Software Engineering”, 10th Edition, Pearson, 2015.
14. K. E. Wiegers, J. Beatty, “Software Requirements”, 3rd Edition, Microsoft Press, 2013. M. Fowler, “UML Distilled: A Brief Guide to the Standard Object Modeling Language”, 3rd Edition, Addison-Wesley, 2003.

15. P. Harmon, “Business Process Change: A Business Process Management Guide for Managers and Process Professionals”, 4th Edition, Morgan Kaufmann, 2019.
16. S. Ambler, “Agile Modeling: Effective Practices for eXtreme Programming and the Unified Process”, Wiley, 2002.
17. М. Зінов’єв, О. Соловйов. Основи проєктування інформаційних систем. — Київ: Каравела, 2021.
18. В. М. Жуков, Ю. І. Щербина. Проєктування баз даних: Навчальний посібник. — Київ: Центр учбової літератури, 2020.
19. С. О. Семерня. Інженерія програмного забезпечення. — Харків: ХНУРЕ, 2019.
20. Н. І. Гнатюк, В. М. Теслюк. Аналіз, моделювання та проєктування ІС. — Львів: Видавництво Львівської політехніки, 2022.

ДОДАТКИ

Додаток А - Інструкція користувача

При переході на сайт, користувача вітає головна сторінка. Знаходячись на сторінці користувач повинен перейти в бажану категорію товару через будь яку з інтегрованих панелей.

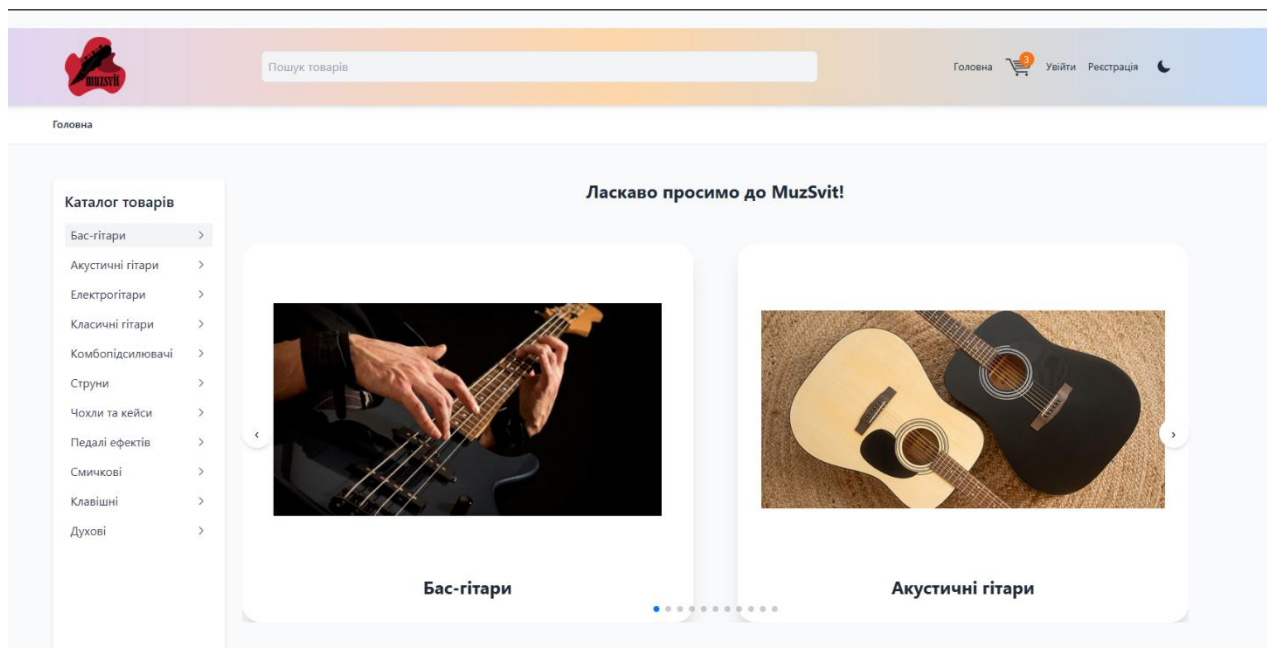


Рисунок А1 – Головна сторінка

Або прокрутити сторінку вниз і в розділі «Новинки» одразу перейти на сторінку товару.

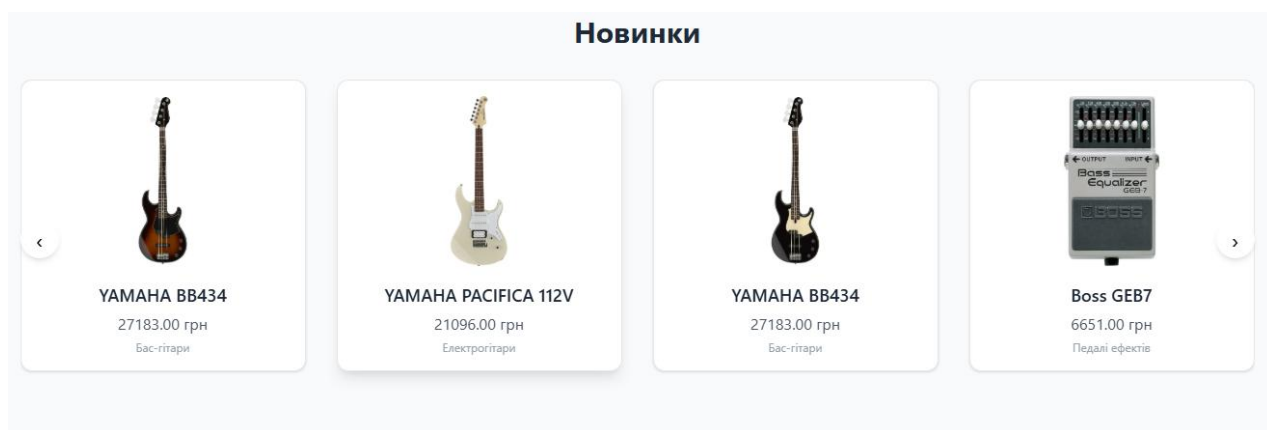


Рисунок А2 - Розділ «Новинки»

Або через інтегрований в навігаційну панель пошуку знайти по назві товару, що потрібен користувачеві.

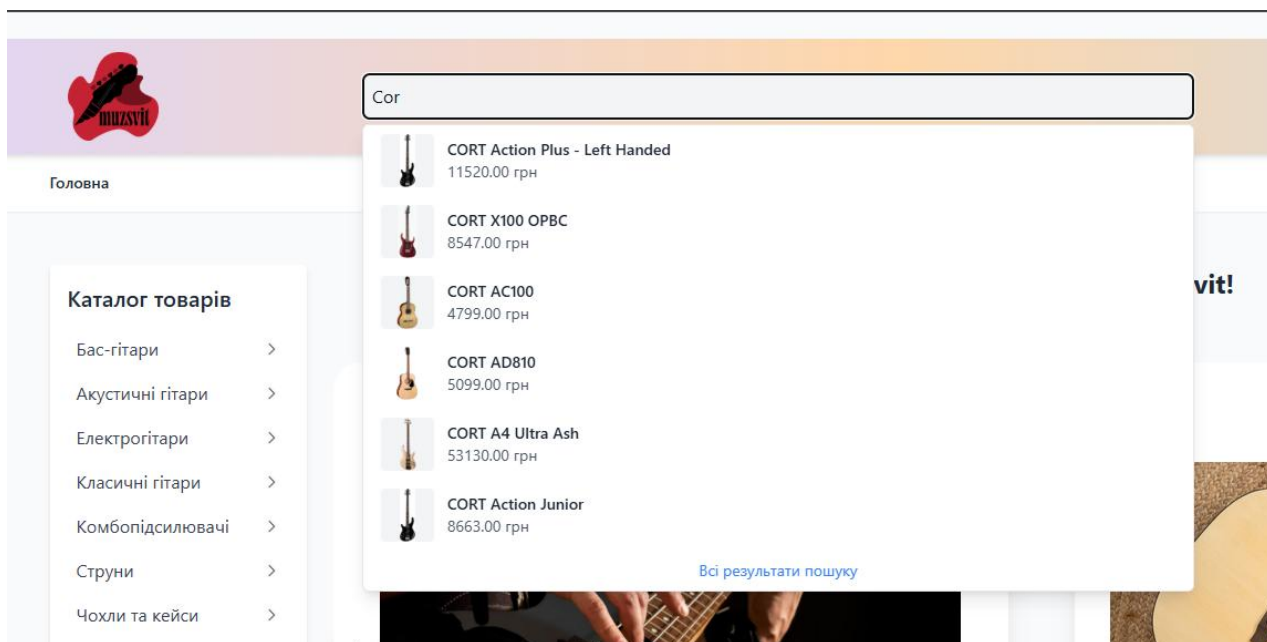


Рисунок А3 – Навігаційна панель пошуку

На сторінці «Категорії» користувач має вибрати необхідні йому фільтри товарів і перейти по бажаному товару.

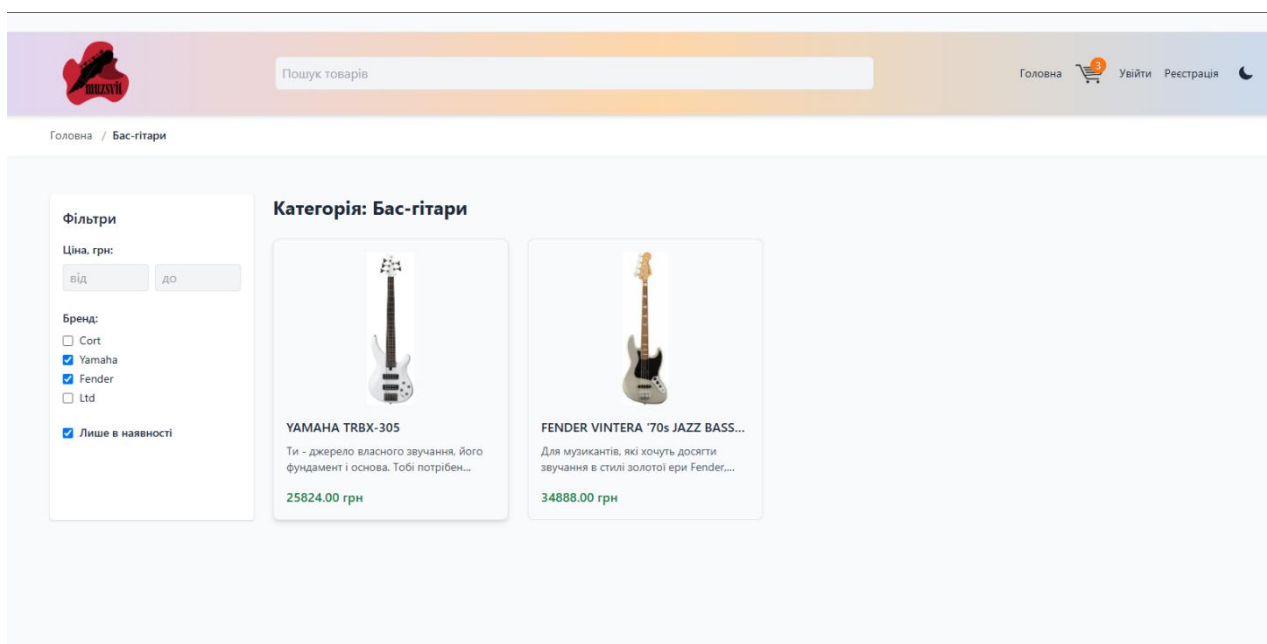


Рисунок А4 – Фільтри товарів

Перейшовши за необхідним товаром, користувача зустріне сторінка з відповідним йому наповненням і кнопкою «додати в кошик». Потрібно натиснути її і користувача перекине в модальне вікно кошику.

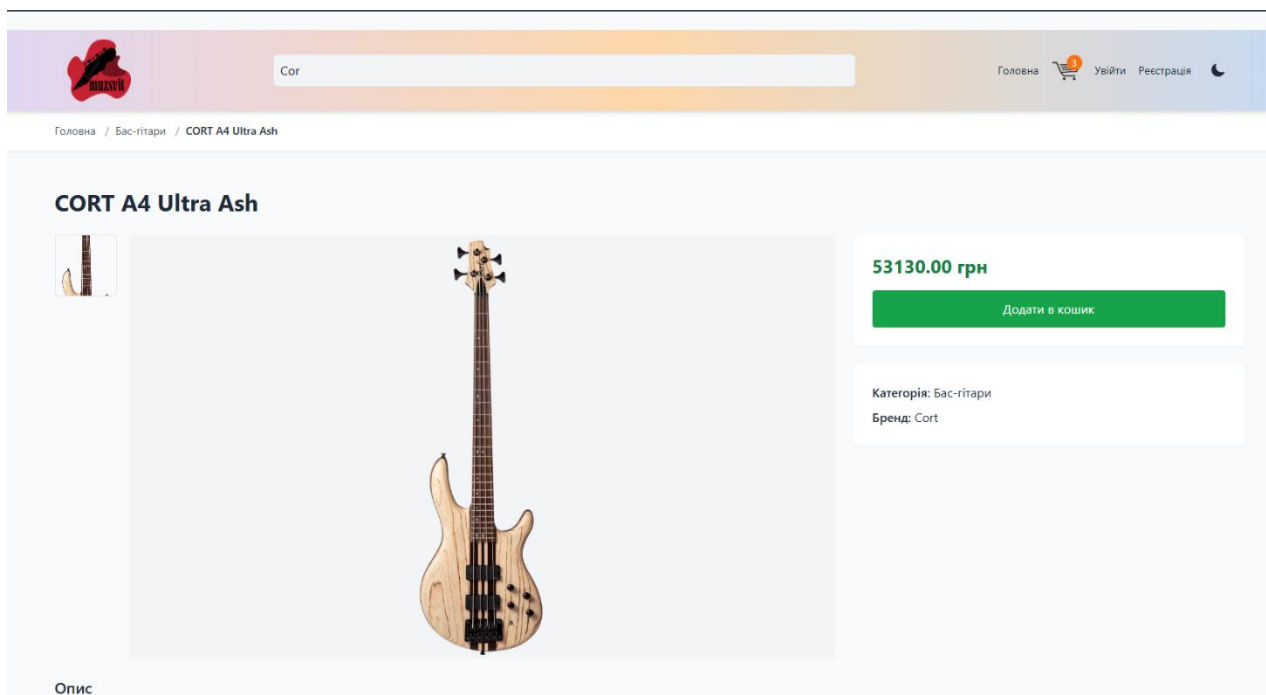


Рисунок А5 – Додавання товару в кошик

Перейшовши до кошику користувач має такі функції:

1. Змінювати кількість обраних товарів через кнопки «+» і «-»
2. Видаляти товари, відкривши відповідне додаткове меню справа
3. Оформити замовлення, натиснувши відповідну кнопку.

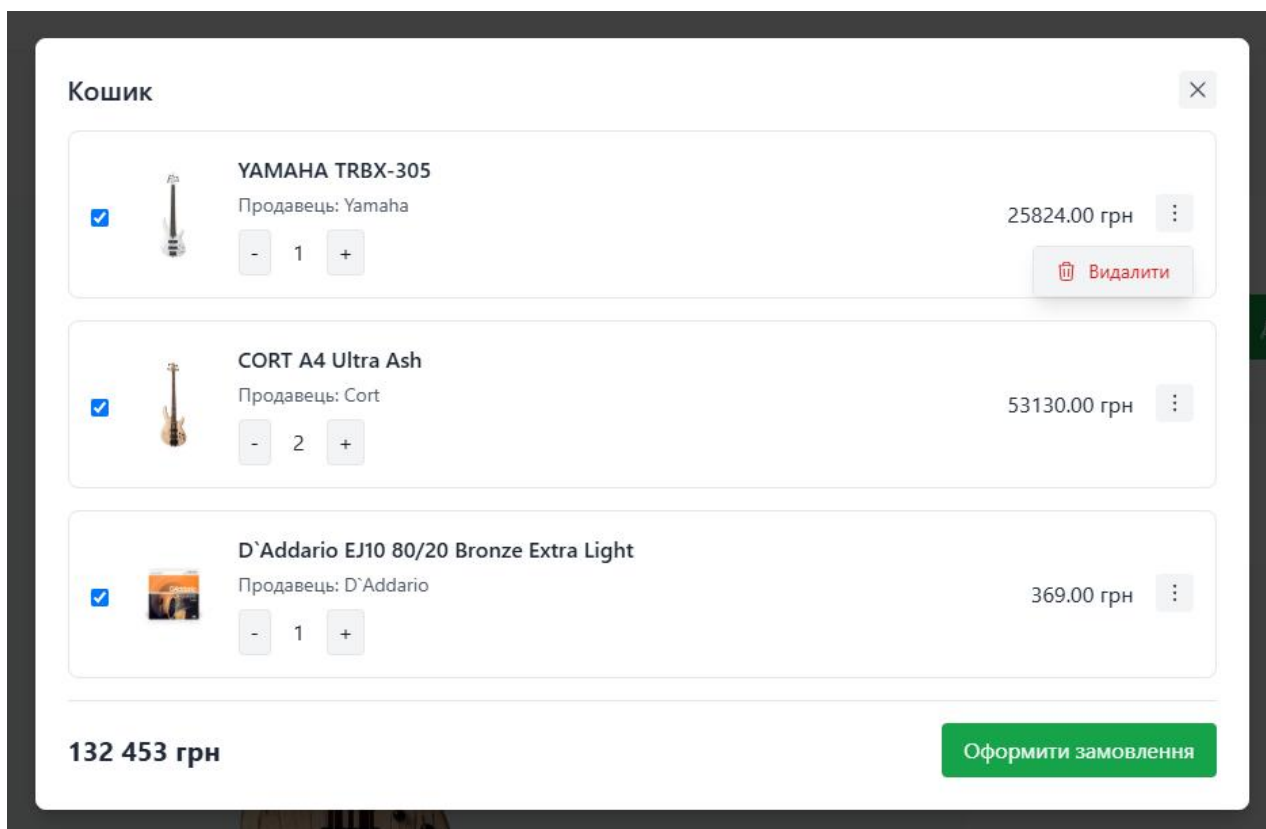


Рисунок А6 - Кошик

Опинившись на сторінці оформлення замовлення користувач має ввести в відповідні форми свої(або іншого отримувача) персональні данні, а також спосіб доставки. Потім прокрутивши сторінку нижче, користувач може натиснути на кнопку «редагувати товар» в інтегрованому вікні з виводом товарів і відкриється кошик, де користувач зможе відредагувати товари с замовлення. Далі необхідно натиснути кнопку «Підтвердити замовлення».

Рисунок А7 – Оформлення замовлення

Після підтвердження замовлення, користувача перекине на сторінку, де його повідомлять про успіх і, що агент з постачання зв'яжеться з ним найближчим часом. Користувач повинен натиснути на логотип сайту, щоб вернутися на головну сторінку для подальшого користування сайтом.

Щоб використовувати повний функціонал сайту, користувач повинен зареєструватися в ньому. Для того, щоб це зробити треба перейти на навігаційній панелі до сторінки реєстрації натиснувши кнопку «Реєстрація»

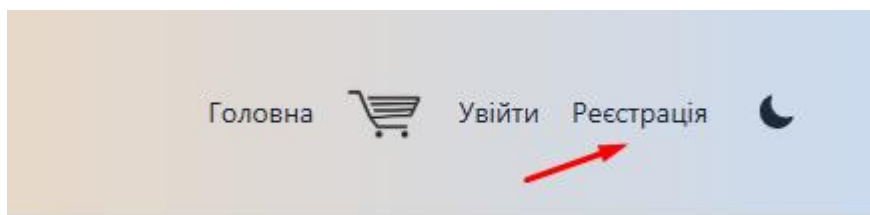


Рисунок А8 – Навігаційна панель реєстрації

Опинившись на сторінці реєстрації користувач повинен ввести індивідуальний логін, будь який пароль і свій email у відповідні поля. Після чого натиснути кнопку «зареєструватися».

 A screenshot of a web page titled 'Реєстрація' (Registration). At the top, there is a header bar with a logo on the left, a search bar with the text 'Пошук товарів' (Search for goods), and navigation links on the right: 'Головна' (Home), a shopping cart icon, 'Увійти' (Login), 'Реєстрація' (Registration), and a moon icon. Below the header, the main content area has a light blue background. It features the title 'Реєстрація' in bold. Underneath are three input fields: 'Логін' (Login) with the placeholder 'Введіть логін', 'Email' with the placeholder 'Введіть email', and 'Пароль' (Password) with the placeholder 'Введіть пароль'. At the bottom of these fields is a button labeled 'Зареєструватися' (Register).

Рисунок А9 – Сторінка реєстрації

Після реєстрації користувача перенесе на сторінку входу в аккаунт, або якщо у користувача вже був створений акаунт, потрібно самому перейти на цю сторінку через навігаційну панель. На сторінці входу треба ввести логін і свій пароль у відповідні поля і натиснути кнопку «увійти».

Рисунок A10 – Сторінка входу в аккаунт

Після натискання на кнопку «увійти» користувача перекине на сторінку його персонального кабінету, де буде виведено всю інформацію о користувачеві і його замовленнях, якщо ця інформація була вже занесена в базу. На цій сторінці користувач може натиснути кнопку «редагувати», після чого відредагувати свої персональні дані, або ввести їх, якщо користувач не повністю записаний в базу даних.

№	Дата	Сума	Статус
1	25.05.2025	74322.00 грн	pending
2	25.05.2025	117309.00 грн	pending
3	25.05.2025	44758.00 грн	pending
4	25.05.2025	5468.00 грн	pending
5	25.05.2025	11520.00 грн	pending

Рисунок A11 – Редагування даних

Після того, як користувач ввів свої дані в особистий кабінет, на сторінці з оформленням замовлення, вони будуть автоматично підтягуватись.

Якщо користувач наділений правами адміністратора, після входу в акаунт, на навігаційній панелі з'явиться кнопка «Адмін», натиснувши яку, користувач перейде на сторінку адміністративної панелі. На цій сторінці адміністратор може додавати категорії, додавати бренди, і додавати або редагувати вже додані товари. Для цього потрібно в відповідні поля ввести необхідну інформацію, підказки до якої є в самих полях форми, у поле «фото», натиснути на відповідну кнопку, вибрати файл зі свого пристрою і натиснути кнопку додавання, наприклад «Додати товар».

The screenshot displays the 'Панель адміністратора' (Administrator Panel) interface. At the top, there is a navigation bar with a logo, a search bar, and links for 'Головна', 'Адмін', 'Користувач', and 'Вихід'. Below the navigation bar, the main content area is titled 'Панель адміністратора' and contains several sections:

- Додати категорію**: A form with a text input for 'Назва категорії', a file upload button 'Вибрати файл', and a status indicator 'Файл не вибрано'. A green 'Додати категорію' button is at the bottom.
- Додати бренд**: A form with a text input for 'Назва бренду' and a green 'Додати бренд' button.
- Додати товар**: A complex form with multiple fields:
 - Назва**: Text input for 'Назва інструменту'.
 - Ціна**: Text input for 'Ціна у грн'.
 - Категорія**: Dropdown menu with 'Оберіть категорію'.
 - Бренд**: Dropdown menu with 'Оберіть бренд'.
 - Кількість на складі**: Text input for 'Скільки доступно?'.
 - Фото**: File upload button 'Вибрати файл' and status 'Файл не вибрано'.
 - Опис**: Text area for 'Короткий опис інструменту'.
 A green 'Додати' button is at the bottom.
- Product List**: A table showing existing products:
 - CORT Action Plus - Left Handed**: Price 11520.00 грн, Brand Cort / Бас-гітари. Description: Електроакустична гітара серії Cort M9800F поєднує в собі сучасну електроніку і традиційні найкращі акустичні інструменти. Buttons: Редагувати, Видалити.
 - CORT X100 OPBC**: Price 8547.00 грн, Brand Cort / Електрогітари.

Рисунок А12 – Панель адміністратора

Прокрутивши сторінку нижче, користувач побачить вже додані товари. Щоб редагувати доданий товар, треба натиснути кнопку «редагувати», або «видалити». При натисканні на кнопку «редагувати», а панель товару буде

введена інформація доданого товару. Щоб зберегти її, треба натиснути кнопку «Оновити».

Додати товар

Назва: CORT Action Plus - Left Handed

Ціна: 11520.00

Категорія: Бас-гітари

Бренд: Cort

Кількість на складі: 8

Фото: Вибрати файл (Файл не вибрано)

Опис: Електроакустична гітара серії Cort MR600F поєднує в собі сучасну електроніку Fishman з традиціями найкращих акустичних інструментів.

Оновити

CORT Action Plus - Left Handed
11520.00 грн — Cort / Бас-гітари
Електроакустична гітара серії Cort MR600F поєднує в собі сучасну електроніку Fishman з традиціями найкращих акустичних інструментів.

Редагувати Видалити

Рисунок А13 – Оновлення даних

Для переведення сайту в темну тему, на навігаційній панелі треба натиснути відповідну кнопку.

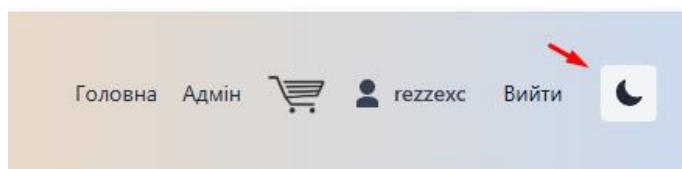


Рисунок А14 – Навігаційна панель переведення в темну тему

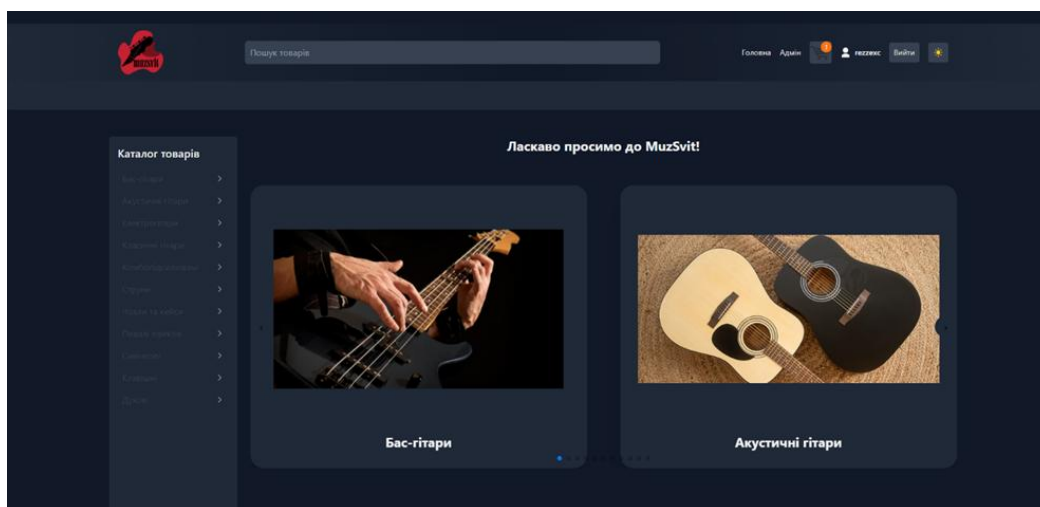


Рисунок А15 – Темна тема

ДОДАТКИ

Додаток Б - Текст програми

```

Server.js:

// backend/server.js
require('dotenv').config();
const express = require('express');
const cors = require('cors');
const helmet = require('helmet');
const cookieParser = require('cookie-parser');
const bodyParser = require('body-parser');
const rateLimit = require('express-rate-limit');
const path = require('path');

const ordersRoutes = require('./routes/orders');
const brandsRoutes = require('./routes/brands');
const db = require('./db');
const authRoutes = require('./routes/authRoutes');
const authMiddleware = require('./middleware/authMiddleware');
const productsRoutes = require('./routes/products');
const categoriesRoutes = require('./routes/categories');
const customersRoutes = require('./routes/customers');

const app = express();
const port = process.env.PORT || 5000;

// Middleware
app.use(cors({ origin: 'http://localhost:3000', credentials:
true }));
app.use(
  helmet({
    crossOriginResourcePolicy: { policy: 'cross-origin' },
  })
);
app.use(cookieParser());
app.use(bodyParser.json({ limit: '1mb' }));
app.use(express.urlencoded({ extended: true }));

// Serve static files from uploads
app.use('/uploads', express.static(path.join(__dirname,
'uploads')));

// Rate limiting for login/register
const loginLimiter = rateLimit({

```

```

    windowMs: 15 * 60 * 1000,
    max: 5,
    message: 'Too many login attempts. Please try again later.'
  });
app.post('/api/login', loginLimiter);
app.post('/api/register', loginLimiter);

// Auth routes (login, register)
app.use('/api', authRoutes);

// Admin protected route
app.use('/api/admin', authMiddleware, (req, res) => {
  if (!req.user.isAdmin) return res.sendStatus(403);
  res.json({ message: 'Welcome admin' });
});

App.js:
import React, { useContext } from 'react';
import { Routes, Route, Navigate, Outlet } from 'react-router-dom';

import HomePage from './pages/HomePage';
import SearchResultsPage from './pages/SearchResultsPage';
import CategoryPage from './pages/CategoryPage';
import ProductPage from './pages/ProductPage';
import CheckoutPage from './pages/CheckoutPage';
import CheckoutLayout from './layouts/CheckoutLayout';
import RegisterPage from './pages/RegisterPage';
import LoginPage from './pages/LoginPage';
import AdminDashboard from './pages/admin/AdminDashboard';
import ProfilePage from './components/ProfilePage';
import Navbar from './components/Navbar';
import Breadcrumbs from './components/Breadcrumbs';
import { CartModal } from './components/CartModal';

import { AuthProvider, useAuth } from './context/AuthContext';
import { SearchProvider } from './context/SearchContext';
import { CartProvider, CartContext } from './context/CartContext';
import { ThemeProvider } from './context/ThemeContext';

function MainLayout() {
  return (
    <>
    <Navbar />

```

```

        <Breadcrumbs />
        <Outlet />
    </>
);
}

function ProtectedAdminRoute({ children }) {
    const { user, loading } = useAuth();
    if (loading) return null;
    if (!user) return <Navigate to="/login" replace />;
    return user.isAdmin ? children : <Navigate to="/" replace />;
}

// Listener для открытия/закрытия модальки корзины
function CartListener() {
    const { isCartOpen, closeCart } = useContext(CartContext);
    return <CartModal isOpen={isCartOpen} onClose={closeCart} />;
}

export default function App() {
    return (
        <ThemeProvider>
            <AuthProvider>
                <SearchProvider>
                    <CartProvider>
                        {/* Слушатель открывает модальку при изменении флага
*/}

                        <CartListener />
                        <Routes>
                            {/* Основной макет */}
                            <Route element={<MainLayout />}>
                                <Route path="/" element={<HomePage />} />
                                <Route path="/search" element={<SearchResultsPage
/>} />

                                <Route path="/category/:categoryId"
element={<CategoryPage />} />
                                <Route path="/product/:id" element={<ProductPage
/>} />

                                <Route path="/register" element={<RegisterPage />}
/>

                                <Route path="/login" element={<LoginPage />} />
                                <Route path="/profile" element={<ProfilePage />}
/>

                                <Route

```

```

        path="/admin"
        element={
            <ProtectedAdminRoute>
                <AdminDashboard />
            </ProtectedAdminRoute>
        }
    />
</Route>
{/* Оформление заказа */}
<Route element={<CheckoutLayout />}>
    <Route path="/checkout" element={<CheckoutPage />}
/>

</Route>
<Route path="*" element={<Navigate to="/" replace
/>} />

</Routes>
</CartProvider>
</SearchProvider>
</AuthProvider>
</ThemeProvider>
);
}

```

```

Products.js:
const express = require('express');
const router = express.Router();
const db = require('../db');
const authMiddleware = require('../middleware/authMiddleware');
const multer = require('multer');
const cors = require('cors');
const ordersRoutes = require('./orders');

const upload = multer({ storage: multer.memoryStorage() });

router.use(cors({ origin: 'http://localhost:3000', credentials:
true }));

// GET /api/instruments
router.get('/', (req, res) => {
    let sql = `
        SELECT i.*, c.CategoryName, b.BrandName
        FROM instruments i
        LEFT JOIN categories c ON i.CategoryID = c.CategoryID
        LEFT JOIN brands b ON i.BrandID = b.BrandID`;

```

```

const params = [];
if (req.query.categoryId) {
  sql += ' WHERE i.CategoryID = ?';
  params.push(req.query.categoryId);
}
db.query(sql, params, (err, results) => {
  if (err) return res.status(500).json({ error: err.message });
  res.json(results);
});
});

// GET /api/instruments/:id
router.get('/:id', (req, res) => {
  const sql = `
    SELECT i.InstrumentID, i.Name, i.Price, i.QuantityInStock,
    i.Description,
           i.CategoryID, c.CategoryName, i.BrandID, b.BrandName
    FROM instruments i
    LEFT JOIN categories c ON i.CategoryID = c.CategoryID
    LEFT JOIN brands b ON i.BrandID = b.BrandID
    WHERE i.InstrumentID = ?`;
  db.query(sql, [req.params.id], (err, results) => {
    if (err) return res.status(500).json({ error: err.message });
    if (!results.length) return res.status(404).json({ message:
'Instrument not found' });
    res.json(results[0]);
  });
});

router.options('/:id/photo', cors({ origin:
'http://localhost:3000', credentials: true }));

// GET /api/instruments/:id/photo
router.get('/:id/photo', (req, res) => {
  db.query(
    'SELECT Photo FROM instruments WHERE InstrumentID = ?',
    [req.params.id],
    (err, results) => {
      if (err) return res.sendStatus(500);
      if (!results.length || !results[0].Photo) return
res.sendStatus(404);
      const image = results[0].Photo;
      // Детектируем PNG vs JPEG по сигнатуре
      if (image[0] === 0x89 && image[1] === 0x50) {

```

```

        res.type('png');
    } else {
        res.type('jpeg');
    }
    res.setHeader('Access-Control-Allow-Origin',
'http://localhost:3000');
    res.setHeader('Access-Control-Allow-Credentials', 'true');
    res.send(image);
}
);
});

// POST /api/instruments
router.post('/', authMiddleware, upload.single('photo'), (req,
res) => {
    if (!req.user.isAdmin) return res.sendStatus(403);
    const { Name, Price, CategoryID, BrandID, QuantityInStock,
Description } = req.body;
    if (!Name || isNaN(Price)) {
        return res.status(400).json({ error: 'Invalid data' });
    }
    const photoBuffer = req.file ? req.file.buffer : null;
    const sql = `
        INSERT INTO instruments
            (Name, Price, CategoryID, BrandID, QuantityInStock,
Description, Photo)
        VALUES (?, ?, ?, ?, ?, ?, ?)`;
    const params = [Name, Price, CategoryID, BrandID,
QuantityInStock, Description, photoBuffer];
    db.query(sql, params, (err, result) => {
        if (err) return res.status(500).json({ error: err.message });
        res.status(201).json({ id: result.insertId });
    });
});

// PUT /api/instruments/:id
router.put('/:id', authMiddleware, upload.single('photo'), (req,
res) => {
    if (!req.user.isAdmin) return res.sendStatus(403);
    const { Name, Price, CategoryID, BrandID, QuantityInStock,
Description } = req.body;
    if (!Name || isNaN(Price)) {
        return res.status(400).json({ error: 'Invalid data' });
    }

```



```

    const fields = ['Name = ?', 'Price = ?', 'CategoryID = ?',
'BrandID = ?', 'QuantityInStock = ?', 'Description = ?'];
    const params = [Name, Price, CategoryID, BrandID,
QuantityInStock, Description];
    if (req.file) {
        fields.push('Photo = ?');
        params.push(req.file.buffer);
    }
    params.push(req.params.id);
    const sql = `UPDATE instruments SET ${fields.join(', ')} WHERE
InstrumentID = ?`;
    db.query(sql, params, err => {
        if (err) return res.status(500).json({ error: err.message });
        res.sendStatus(200);
    });
});

// DELETE /api/instruments/:id
router.delete('/:id', authMiddleware, (req, res) => {
    if (!req.user.isAdmin) return res.sendStatus(403);
    db.query('DELETE FROM instruments WHERE InstrumentID = ?',
[req.params.id], err => {
        if (err) return res.status(500).json({ error: err.message });
        res.sendStatus(204);
    });
});

module.exports = router;

```