

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

ПОЯСНЮВАЛЬНА ЗАПИСКА

до кваліфікаційної роботи освітнього ступеня «бакалавр»
за спеціальністю 121 «Інженерія програмного забезпечення»
(освітня програма «Інженерія програмного забезпечення»)

на тему:

«Розробка Web-сторінки «Інтернет магазин одягу та аксесуарів»»

Виконав студент групи ПЗ-19-3
АРЦРУНЯН Ншан Карапетович

Керівник роботи:
КРАВЧЕНКО Світлана Миколаївна

Рецензент:
ЧИЖМОТРЯ Олексій Володимирович

Житомир – 2023

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
ФАКУЛЬТЕТ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«ЗАТВЕРДЖУЮ»

В.о. зав. кафедри інженерії
програмного забезпечення

Андрій МОРОЗОВ

«02» серпня 2023 р.

ЗАВДАННЯ
на кваліфікаційну роботу

Здобувач вищої освіти: **АРЦРУНЯН Ншан Карапетович**

Керівник роботи: **КРАВЧЕНКО Світлана Миколаївна**

Тема роботи: **«Розробка Web-сторінки «Інтернет магазин одягу та аксесуарів»»**

затверджена Наказом закладу вищої освіти від **«02» серпня 2023 р., № 326/с**

Вихідні дані для роботи: **методи та засоби розробки веб-додатків, аналіз проблематики інтернет магазину одягу та аксесуарів, даний сайт має відповідати сучасному рівню наявних технологій та бути зручним у використанні.**

Консультанти з бакалаврської кваліфікаційної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Завдання видав	Завдання прийняв
1	Кравченко С.М.	02.08.2023р.	02.08.2023р.
2	Кравченко С.М.	02.08.2023р.	02.08.2023р.
3	Кравченко С.М.	02.08.2023р.	02.08.2023р.

Календарний план

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Постановка задачі	25 липня 2023 – 28 липня 2023	Виконано
2	Пошук, огляд та аналіз аналогічних розробок	02 серпня 2023 – 05 серпня 2023	Виконано
3	Формулювання технічного завдання та опрацювання літературних джерел	06 серпня 2023 – 08 серпня 2023	Виконано
4	Проектування структури системи	10 серпня 2023 – 14 серпня 2023	Виконано
5	Написання програмного коду	15 серпня 2023 – 18 серпня 2023	Виконано
6	Тестування роботи додатку	20 серпня 2023 – 25 серпня 2023	Виконано
7	Попередній захист	26 серпня 2023 – 28 серпня 2023	Виконано

РЕФЕРАТ

Кваліфікаційна робота бакалавра являє собою веб-сайт магазин одягу та аксесуарів. Пояснювальна записка до кваліфікаційної роботи містить 63 сторінки, 45 ілюстрацій, 1 таблицю, 1 додаток та 16 літературних джерел.

Метою кваліфікаційної роботи є ознайомлення з веб технологіями та патернами проектування, аналіз алгоритмів, архітектури та способів реалізації інтернет магазинів.

В роботі визначені головні завдання для розробки системи, проаналізовано методи вирішення поставленої задачі, обґрунтовано вибір патерна та технологій реалізації та наведено перелік бібліотек які на сьогоднішній день активно використовуються для реалізації інтернет магазинів. Представлено алгоритм роботи програмного комплексу, структуру та взаємодію його деяких компонентів. Також описана програмна реалізація інтернет магазину одягу.

В рамках кваліфікаційної роботи проводилось тестування програмного комплексу в результаті якого було визначено та оцінено працездатність системи.

КЛЮЧОВІ СЛОВА: ТОВАРИ, КАТЕГОРІЇ, ПОШУК, БАЗИ ДАНИХ.

					ІПЗ.КР.Б – 121 – 23 – ІПЗ			
					Пояснювальна записка Розробка WEB-сторінки «Інтернет магазин одягу та аксесуарів»			
Зм.	Арк.	№ докум.	Підпис	Дата				
Розроб.		Н. Арирунян						
Керівник		С. Кравченко						
Рецензент.		О. Чижморя						
Зав. каф.		А. Морозов						
						Літ.	Арк.	Аркушів
							4	63
						Житомирська політехніка, група ІПЗ-19-3		

ABSTRACT

The bachelor's thesis is website for a clothing and accessories store. The explanatory note to the final thesis contains 63 pages, 45 illustrations, 1 table, appendice and 16 literary sources.

The aim of the final work is to get acquainted with web technologies and design patterns, analysis of algorithms, architecture and methods of implementation of online stores.

The work defines the main tasks for the development of the system, analyzes the methods of solving the task, substantiates the choice of pattern and implementation technologies, and provides a list of libraries that are currently actively used for the implementation of online stores. The algorithm of the software complex, the structure and interaction of some of its components are presented. The software implementation of an online clothing store is also described.

As part of the graduation thesis, the software complex was tested, as a result of which the system's performance was determined and evaluated.

KEYWORDS: GOODS, CATEGORIES, SEARCH, DATABASES.

					ІІЗ.КР.Б – 121 – 23 – ІІЗ				
Зм.	Арк.	№ докум.	Підпис	Дата					
Розроб.		Н. Арцрунян			Пояснювальна записка Розробка WEB-сторінки «Інтернет магазин одягу та аксесуарів»		Літ.	Арк.	Аркушів
Керівник		С. Кравченко						4	63
							Житомирська політехніка, група ІІЗ-19-3		
Рецензент.		О. Чижмотря							
Зав. каф.		А. Морозов							

ЗМІСТ

РЕФЕРАТ.....	4
ABSTRACT.....	5
ВСТУП.....	7
РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ ІНТЕРНЕТ МАГАЗИНУ ОДЯГУ.....	10
1.1. Постановка задачі.....	10
1.2. Аналіз аналогів програмного продукту.....	12
1.3. Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення.....	17
Висновки до розділу 1.....	22
РОЗДІЛ 2. ПРОЕКТУВАННЯ СИСТЕМИ АВТОМАТИЗАЦІЇ ІНТЕРНЕТ МАГАЗИНУ ОДЯГУ	23
2.1. Визначення варіантів використання та об'єктно-орієнтованої структури системи.....	23
2.2. Опис структури даних з якими працює інтернет магазин.....	27
2.3. Реалізація інтернет магазину одягу	28
Висновки до розділу 2.....	35
РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ІНТЕРНЕТ МАГАЗИНОМ ОДЯГУ	37
3.1. Архітектура програмного комплексу та його розгортання	37
3.2. Структура інтерфейсу та порядок роботи з інтернет магазином одягу	38
3.3 Функціональне тестування програмного забезпечення	49
Висновки до розділу 3.....	51
ВИСНОВКИ	53
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	54
ДОДАТКИ	56

ВСТУП

Актуальність теми. Інтернет-торгівля - процес реалізації фізичних і нефізичних товарів за допомогою спеціалізованих електронних майданчиків, що надають дистанційне оформлення замовлення. Інтернет-торгівля в Україні розвивається досить успішно. За підсумками досліджень обсяг ринку інтернет-торгівлі в Україні постійно збільшується як і збільшується кількість магазинів електронної торгівлі. Обсяг ринку інтернет-торгівлі за підсумками 2017 р перевищив 537 млрд і протягом останніх 3 років ріс в середньому на 42,2% в рік. Найбільшими сегментами є «Побутова техніка і електроніка» (125,5млрд), «Гіпермаркети» (85 млрд) і «Одяг і взуття» (74,2 млрд).

Вдалий web-сайт - це ефективний інструмент торгівлі, здатний привернути увагу самої різнобічної аудиторії. Як і будь-який інший маркетинговий інструмент, заснований на принципі безпосереднього відгуку, перш за все він повинен зацікавити відвідувача, а потім спонукати його на певні дії, в частині придбання пропонованого сайтом комерційного продукту.

Слід зазначити, що часто розробники сайтів не приділяють належної уваги головній сторінці сайту, і як правило, не отримують очікуваного ефекту від відвідувачів. Такі web-сайти, нехай навіть містять іноді значну кількість корисних порад та статей, практично ніколи не досягають передбачуваного рівня відвідуваності, не кажучи вже про продажі.

Існують як невеликі сайти, на яких розташовується коротка інформація про компанію і послуги, так і об'ємні інтернет-каталоги фірм з найдокладнішими характеристиками товарів, їх зображеннями та цінами. Найчастіше такий інтернет-каталог створюється для того, щоб відвідувач, міг знайти докладний опис і зображення товару. Тобто сайт в даному випадку є ні чим іншим як ілюстрованим рекламним каталогом товарів.

Web-сайт, здатний привернути увагу і викликати цікавість, викличе у клієнтів бажання не тільки переглянути інформацію про товар і зробити покупки, але і знову відвідати його через деякий час, а також рекомендувати своїм друзям і знайомим.

Переваги інтернет-магазину:

- не вимагає витрат на оренду площі, найму продавців та іншого персоналу;
- доступ до віртуальних вітрин магазину може отримати будь-який покупець, не залежно від місця проживання;
- інтернет-магазин не має обмежень на віртуальну площу;
- власник інтернет-магазину може здавати в оренду свої віртуальні торгові площі так само, як і власник звичайного магазину;
- термін та вартість створення інтернет-магазину незрівнянно нижча, ніж звичайного магазину;
- потенційний покупець отримує вичерпну інформацію про цікавить його товар або послугу без допомоги посередників (продавця, менеджера);
- інтернет-магазин, який працює в режимі on-line, передбачає зворотний зв'язок з покупцем (форуми, чати, гостьові книги), що дозволяє оперативно відповісти на запитання користувачів, провести маркетингові дослідження, отримати відгуки і побажання;
- можна розмістити як завгодно багато товарів або описати будь-яку кількість послуг;
- можливість робити покупки, не виходячи з дому, роблять інтернет-магазини ще більш привабливими для покупців.

Роблячи свій вибір на користь відкриття інтернет-магазину, компанія переходить на більш високий рівень розвитку свого бізнесу і робить свій асортимент більш доступним для широкого кола покупців.

Виходячи з вище наведених фактів актуальність створення власного інтернет магазину для компанії, що продає одяг є очевидною.

Метою кваліфікаційної роботи є дослідження особливостей застосування веб-технологій та підходів для створення інтернет-магазинів та розробка веб-додатку магазину одягу.

У кваліфікаційній роботі ми досліджуємо архітектуру CMS для створення інтернет магазину та її оптимізація.

Результатом кваліфікаційної роботи є реалізований програмний додаток для автоматизації роботи інтернет магазину із адміністративною частиною.

З мети можна визначити низку наступних завдань:

- вивчення основних принципів роботи інтернет магазинів;
- проведення аналізу задач, що мають бути реалізовані в рамках проекту;
- визначення алгоритму та архітектури для реалізації програмного додатку;
- обґрунтування та вибір засобів реалізації програмного додатку;
- реалізація програмного додатку;
- оптимізація реалізованого програмного додатку завдяки застосуванню різних веб технологій;
- тестування програмного додатку, аналіз результатів у вигляді остаточних висновків.

Об’єктом дослідження є веб-технології та засоби проектування веб-магазинів для торгівлі одягом на прикладі інтернет магазину одягу.

Предметом дослідження є дослідження процесу організації та проектування інтернет магазину.

В роботі було проаналізовано системи розробки інтернет магазинів, розглянуто загальні поняття проектування інформаційних систем типу «інтернет-магазин», обрано технології та архітектура проектування системи.

Пояснювальна записка до кваліфікаційної роботи містить 63 сторінки, 45 ілюстрацій та 1 таблицю, список літературних джерел містить 16 найменувань.

РОЗДІЛ 1. АНАЛІЗ НАПРЯМКІВ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ ІНТЕРНЕТ МАГАЗИНУ ОДЯГУ

1.1. Постановка задачі

Головною метою даної роботи є ознайомлення з практикою програмування та реалізацією CMS систем для розробки інтернет магазинів [4].

Електронний магазин (віртуальний магазин, інтернет магазин) можна розглядати як прикладну систему, яка побудована як елемент технології системи електронної комерції. Подібно до звичайного магазину, електронний магазин реалізує наступні основні функції: представлення товарів (послуг) покупцю, обробку замовлень, продаж і доставку товарів.

В інтернеті часто зустрічається інше визначення цього поняття, більш примітивне, - сайт, який торгує товарами і послугами в мережі інтернет.

Електронний магазин дозволяє користувачам сформувати замовлення на покупку, вибрати спосіб оплати і доставки замовлення в мережі Інтернет.

Підсумовуючи вищесказане, відзначимо основні етапи для реалізації поставленого завдання:

1. Постановка задачі
 - 1.1. Вибір архітектури програми.
 - 1.2. Підбір тестових та тренувальних даних.
 - 1.3. Визначення особливостей розробки системи автоматизації.
2. Аналіз вимог
 - 2.1 Створити головну сторінку, яка даватиме можливість вибору різних категорій товарів;
 - 2.2 Додати можливість реєстрування, вхід у свій профіль, редагування або видалення облікового запису;

2.3 Створити сторінку, на якій користувачі зможуть вибирати, фільтрувати, сортувати товари, а також змінити категорію без переходу на головну сторінку;

2.4 Створити сторінку, де користувачі зможуть дізнатися більше інформації про певний товар, повернутися назад до каталогу чи додати бажане до кошика;

2.5 Додати можливість перегляду бажаних товарів у кошику, а також подальшого оформлення замовлення;

2.6 Додати можливість адміністраторам додавання та видалення певних товарів;

3 Планування

3.1. Проаналізувати вже існуючі способи програмної реалізації інтернет магазинів.

3.2. Спроектувати загальну концепцію взаємодії класів системи;

3.3. Відшукати шляхи зменшення відсотку помилок системи.

3.4. Підготувати матеріали для тестування системи.

3.5. Розробити зручний дизайн для роботи з програмою який буде використовувати технології GUI.

4. Реалізація

4.1. Розробка системи на основі патерна проектування MVC

4.1.1. Написати базові класи для:

- роботи з базою даних BD та підготовки вибірок для роботи із інтернет магазином;

- роботи з категоріями та товарами – Product та Category [6];

- роботи із користувачами та кабінетами користувачів – User;

- загальний клас переадресації та роботи інтернет магазину – Router;

- класи для роботи адміністративної системи.

4.1.2. Реалізувати окремі методи роботи із кошиком та замовленнями користувачів.

4.2. Реалізувати структуру збереження даних наступного виду:

					ІПЗ.КР.Б – 121 – 23 – ПЗ	
						11

4.2.1. База даних MySQL [3].

4.3. Тестування системи:

4.3.1. Підготувати контрольні приклади для тестування інтернет магазину.

4.3.2. Визначити відсоток помилок.

4.3.3. Перевірити на можливість оптимізації системи.

Результат реалізації поставленого завдання – реалізований інтернет магазин на мові програмування PHP, базою даних MySQL.

1.2. Аналіз аналогів програмного продукту

Щоб досягнути поставлених завдань і, при можливості, додати щось нове потрібно проаналізувати існуюче програмне забезпечення. Це дасть багато корисної інформації, яку бажано застосувати при розробці веб-додатку, а також можна уникнути багатьох різних помилок.

Всі інтернет магазини відрізняються за можливостями і функціями, тому порівнюємо функціональні та технологічні характеристики.

Опис будь-якої системи починається з загальних показників, що характеризують її особливості і індивідуальність. До таких належить інформація про розробника, вартість використання, знижки при покупці і так далі.

Загальна інформація про систему:- Розробник (приватна особа, студія); - Вартість ліцензії;- Умови використання Ліцензії (наприклад, обмеження по кількості доменів, орендна плата і так далі);- Тип CMS (коробка, студійна, SaaS); - Наявність оновлень і їх вартість (багато розробників комерційних платформ надають безкоштовні оновлення, таким чином Ваш магазин завжди сучасний з технічної точки зору);- Дата виходу останньої версії (показує «свіжість» розробки);- Наявність демо-продукту або тріал-версії (наявність демо завжди позитивний фактор у виборі, так як приємно реально подивитися, що купуєш)- Наявність документації користувача (досить важливий пункт, так як після створення магазину дуже часто адміністратором стає його власник, який може бути і не програмістом.

Наявність технічної документації розробника (будь-якому професіоналу, краще мати на руках інформацію про систему, це полегшує роботу з нею);-

Наявність, вартість і умови надання технічної підтримки від розробників CMS;- Мова поширення (наприклад, англійська, російська, японська, німецька).

Технічні характеристики

Залежно від заданих параметрів знаходиться вся подальша робота з системою, тому технічними показниками варто приділити пильну увагу.

- Захищеність вихідного коду (закритий чи відкритий)
- Мова програмування (PHP, Perl, Asp.Net, Ruby і Java + Ajax) [1].
- Тип бази даних (Файл, MySQL, PostgreSQL, MSSQL, Oracle, SQLite, Firebird)
- Движок шаблонів і використання шаблонизатору (наприклад, Smarty)
- Технічні вимоги до хостингу
- Налаштування зовнішнього вигляду інтернет магазину (наявність платних / безкоштовних шаблонів і умови їх установки, наявність вбудованих шаблонів)

Функції клієнтської частини

Широта можливостей для користувача інтерфейсу - це необхідна умова успішності інтернет магазину. Бо якщо покупцям зручно і комфортно здійснювати покупки в Вашому інтернет магазині, то вони повернуться знову.

Пошуковий функціонал (відповідає за пошук і підбір товарів)

- Пошук по слову
- «Живий пошук» з підказками зі зворотним відліком
- Пошукове хмара тегів
- Фільтри і підборники за характеристиками продуктів. Полегшують пошук клієнта потрібних продуктів і товарів. Найчастіше мають вузькоспеціалізований, тематичний вид, наприклад, підборник шин і дисків або фільтр характеристик електронних товарів.
- Сортування товарів за різними параметрами (ціна, рейтинг, виробник, ім'я, дата завантаження та інше) подання товарів

- Картка товару повинна містити наступну інформацію (опис, галерею зображень, ціну, характеристики продукту, наявність на складі, відгуки до продуктів, рейтингову оцінку, версію для друку)

- Функція «лупа» для детального розгляду товару (збільшує зображення при наведенні на нього мишкою);

- Функція 3D -перегляд (рідкісна опція в сучасних інтернет магазинах, проте, дуже корисна);

- Змінні характеристики товарів (надає вибір для покупця. При бажанні він може замовити товар в тій конфігурації, в якій йому потрібно без додаткового дзвінка і узгодження до продавця. До таких характеристик може відноситься розмір, колір, вага, різні опції і так далі).

- висновок супутніх товарів і послуг;

- висновок аналогів і замінників (розширює вибір покупця);

- функція «недавно дивилися» для більш зручної навігації;

- висновок і наявність рекламних банерів

- висновок новинок, хітів продажів, знижених у ціні товарів і т.д.

Особистий кабінет покупця (додавання редагування контактних даних)

- перевірка статусу замовлення

- Історія замовлень

Кошик покупок і оформлення замовлення (різні види з реєстрацією або без)

Інформаційний та комунікаційний функціонал

- Новини та їх розсилка

- Публікація інформаційних сторінок (про компанії, доставка, оплата і так далі).

- месенджер (чати, ICQ-месенджер, безкоштовний дзвінок з сайту, відправка запит про товар з сайту, замовлення дзвінка, зворотний зв'язок);

- Гостьова книга;- Словник;- Голосування і опитувальники; - Функція розповісти другу; - Соціальні закладки; - Список бажаних товарів.

При аналізі вже існуючого ПЗ за тематикою роботи було виявлено декілька проектів. Перший інтернет-магазин зображено на рисунку 1.1.

					ІПЗ.КР.Б – 121 – 23 – ПЗ	
						14

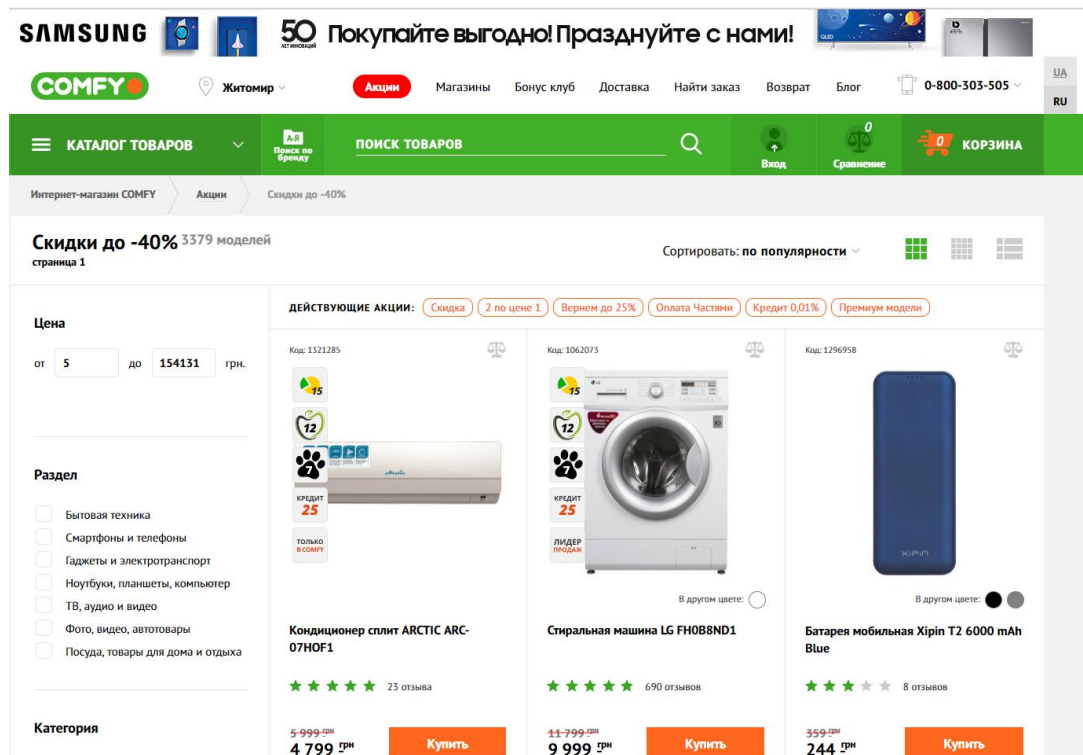


Рисунок 1.1 - Интернет-магазин «Comfy»

Ще один інтернет-магазин зображено на рисунку 1.2.

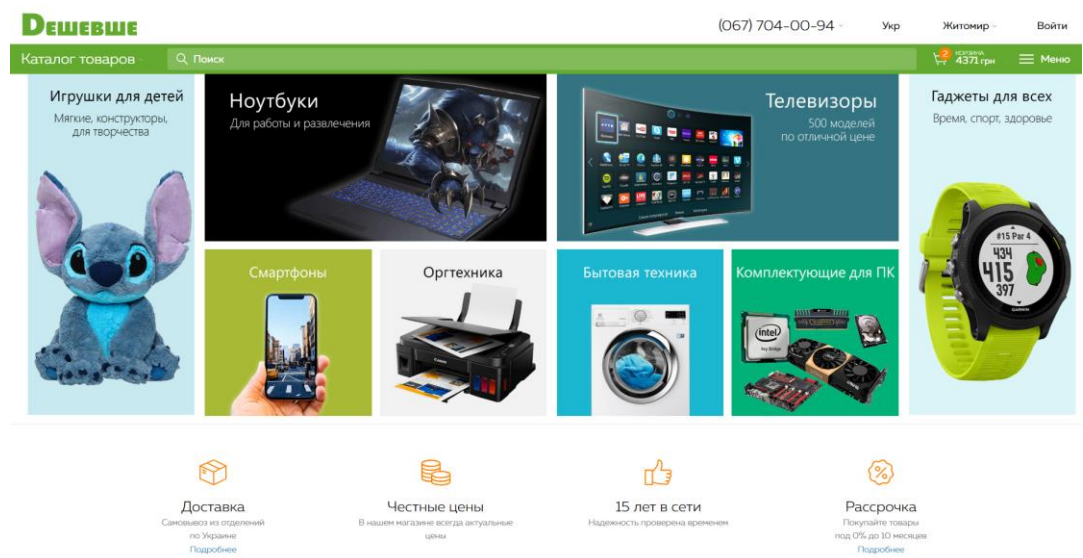


Рисунок 1.2 – Интернет-магазин «Дешевше»

Також варто згадати відомий американський інтернет-магазин «Ебай», який зображено на рисунку 1.3.

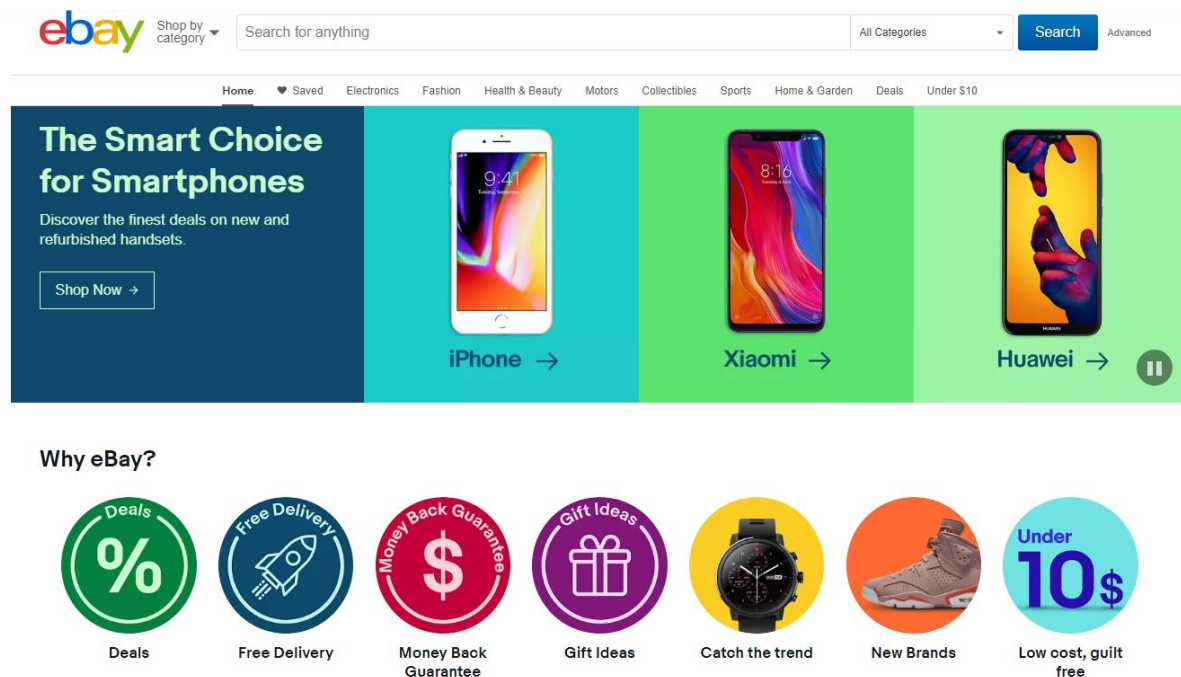


Рисунок 1.3 – Інтернет-магазин «eBay»

Краще ознайомившись із вище вказаними веб-додатками, було проаналізовано їхні відмінності які зображені в таблиці 1.1.

Таблиця 1.1

Назва сайту	Дизайн	Зручність використання	Різноманітність товарів
Comfy	Простий	Простий у використанні	Побутова та цифрова техніка
Дешевше	Приємний	Все чітко, зрозуміло та просто	Побутова та цифрова техніка
Ebay	Простий	Незвично, проте також все зручно та просто	Товари на різний смак

Ознайомившись із таблицею 1.1 можна зробити висновок, що:

– Comfy володіє простим та зручним інтерфейсом та пропонує вибір з побутової та цифрової техніки. Але не реалізовано ряд функцій таких як: підбор супутніх товарів і послуг; підбор аналогів і замінників.

– Дешевше має більш кращий дизайн, а також демонструє зручність, чіткість та простоту у використанні. Щодо товарів, на сайті користувачі можуть знайти також побутову та цифрову техніку.

– Ebaу це відомий американський інтернет-магазин, де можна замовити практично будь-яку річ. Цей веб-сайт має незвичний інтерфейс, проте також простий у використанні.

Проаналізувавши аналоги програмного додатку, було зроблені певні висновки, які є водночас новими ідеями для створення даного проекту, а саме інтернет-магазину одягу.

1.3. Обґрунтування вибору інструментальних засобів та вимоги до апаратного забезпечення

Для реалізації інтернет магазину що з функціональністю продаж одягу було використано різні технології для створення клієнтської та серверної частини проекту [7].

Клієнтські технології (JavaScript, CSS, HTML5, jQuery, Bootstrap) [5] [8].

HTML5 [15] - нова відкрита платформа, призначена для створення веб додатків використовують аудіо, відео, графіку, анімацію та ін. HTML5 вводить цілий ряд нових елементів, які спрощують структуру сторінок. Більшість сторінок на HTML4 містять типові елементи, такі як «шапка», «підвал» і колонки. Нині, як правило, в коді документа вони позначаються за допомогою елементів <div>, описуючи кожен атрибутом id або class [11].

HTML5 залишився мовою розмітки, але відрізняється від попередньої версії досить серйозно. Якщо минула четверта версія була тільки мовою розмітки, то нова версія містить мову розмітки, засоби програмування CANVAS на базі JavaScript [16], розширені можливості роботи з медіа, розширені елементи форм. Дана мова взагалі називають мовою семантичної розмітки через чіткої структури представлення елементів сайту, так як в даній версії замість єдиного тега визначає блок, з'явилася можливість розбиття статті на логічні області за

допомогою нових тегів <HEADER> <BODY> <FOOTER> і семантичного розбиття <SECTION> <ARTICLE> <DIV> <ASIDE> <NAV>

У сучасній розробці клієнтської частини сайту на етапі розмітки, тобто побудови макета сторінок сайту використовується HTML5 в зв'язці з таблицями стилів CSS3 [12].

CSS3 [9]. Розширені таблиці стилів. CSS3 - це новий стандарт оформлення HTML документів значно розширює можливості попереднього стандарту CSS2. Дані стилі дозволяють створювати візуальні ефекти і навіть анімацію, причому досить високого рівня. Короткий опис нових можливостей:

- Створення елементів зі згладженими кутами;
- Створення лінійних і сферичних градієнтів;
- Гнучке оформлення фонових картинок елементів;
- Додавання до елементів і до тексту тіні;
- Створення анімацій і різних ефектів переходів;
- Створення різних трансформацій;
- Завдання квітів декількома новими способами;
- Оформлення кордонів, фонів, тексту.

Найбільш вузьким місцем при роботі з сайтом є його постійний зв'язок з сервером. До недавнього часу було чітке розділення – оформленням займається клієнтська частина сайту, а функціоналом - серверна. але кількість користувачів як мережі, в загальному, так і окремих сайтів, зокрема, постійно росте. Більш того зростання кількості відвідувачів сайту є однією з кінцевих цілей розробки сайту, тому щоб зняти навантаження на сервер, частина функцій сайту, в сучасній веб розробці, передається на сторону клієнта. Для цих цілей розроблено декілька технологій, найбільш розвинута з яких - AJAX (асинхронний JavaScript і XML). Мета даної технології реалізація деяких функцій сайту без звернення до сервера. Для створення динамічного контенту і реалізації клієнтської частини функціоналу сайту використовуються клієнтські мови програмування. Як стандарти мови клієнтського веб програмування визначено JavaScript. Однією з кардинальних особливостей клієнтських мов програмування полягає в тому, що

вони виконуються браузером і вони не мають прямого доступу до файлової системи, а значить, і до баз даних (є виключення для XML файлів). Це стосується і JavaScript. Дана мова є об'єктною, об'єктно-орієнтованою і, як об'єктна мова підтримує DOM (об'єктну технологію подання документа або подання веб документа у вигляді групи об'єктів на деякій підкладці, яка носить назва window. Вікно може бути тільки одне і воно має низку властивостей.

У вікні розташовується документ (який так само має ряд властивостей), а вже на документі розташовуються елементи в певному порядку. Порядок розташування елементів може бути порушений при використанні CSS стилів, які, в свою чергу, можуть управлятися командами JavaScript.

Найбільш поширеним сьогодні є фреймворк jQuery з фреймворком всередині себе jQuery UI. jQuery дозволяє виконувати всі функції клієнтської веб розробки і значно розширює і спрощує код завдяки розробленим в ньому функцій. jQuery UI побудований на базі фреймворка і спрямований на оформлення основних елементів інтерфейсної частини сайту (меню, текстові блоки, галереї і т.д.). Використання фреймворків в розробці є характерною ознакою сучасного сайту і, в даний час, є необхідним елементом.

На базі jQuery розроблений новий фреймворк керуючий динамічним оформленням клієнтської частини сайту Bootstrap - популярний фреймворк, який дозволяє швидко і якісно створювати статичні веб-сайти і веб-додатки. По суті, це безкоштовний набір інструментів, що дозволяє використовувати Html, CSS і JavaScript «великими мазками».

Вимоги до клієнтського програмного забезпечення

Сайт повинен бути доступний для повнофункціонального перегляду з допомогою наступних браузерів:

- Mozilla Firefox 10.0 і вище;
- Google Chrome 12.0 і вище;
- Safari 5.0 і вище;
- MS IE 8.0 і вище;
- Opera 8.0 і вище.

1.3.2. Серверні технології (PHP, MySQL)

Як вже згадувалося вище, на серверну частину сайту покладається весь функціонал сайту, тому кількість мов програмування серверної частини сайту значно більше, ніж клієнтської і, крім того, немає однієї технології, яка була б визначена в якості стандарту розробки. Серед серверних мов можна виділити Ruby, ASP, JSP, Python, Perl, але найбільш поширеною (більше 5 млн. серверів) є мова PHP версії 7 [2]. Особливістю даної мови, саме в 7 версії, є повна підтримка об'єктно-орієнтованої технології програмування, що спрощує роботу з повторного використання коду, покращує читаність коду і дозволяє працювати над створенням сайту команді розробників. На обраній мові написана досить велика кількість фреймворків (каркасів розробки структурних і функціональних елементів сайту). Серед найбільш поширених фреймворків можна виділити Symphony, Zend, Yii, Doctrine, Smarty, Drupal, Wordpress і ін. Серед цих фреймворків існують універсальні і спеціалізовані (Призначені для реалізації тільки деяких функцій і завдань веб розробки). Використання фреймворків дозволяє значно прискорити процес розробки, зробити код більш стандартним і читаним. На базі фреймворків розроблені CMS - системи управління контентом сайту, спеціальні рішення, які дозволяють розгорнути структуру сайту і потім допрацьовувати функціонал, в залежності від вимог до сайту.

Сучасні динамічні сайти обов'язково використовують базу даних як для зберігання різного контенту, так і для реалізації своїх функцій. Найбільш поширеними системами управління базами даних є MySQL, PostgreSQL, ORACLE. Ці бази даних є полегшеними, тобто мають урізаним функціоналом, достатнім для задач веб розробки. Найбільш поширеною є MySQL. Дана база досить проста в використанні і має високу швидкість обробки запитів, що важливо при великій кількості звернень від користувачів до сервера, на якому розташовується сайт [18].

Oracle Database або Oracle RDBMS. Об'єктно-реляційна система управління базами Даних компанії Oracle Corporation. Ця СУБД Забезпечує ефективне, надійне і безпечне управління даними таких критично важливих для

бізнесу додатків, як онлайнові середовища, Виконує масштабну обробку транзакцій (OLTP), сховище даних з скроню інтенсивністю потоку запитів, а також ресурсомісткі інтернет-додатки. Система Oracle Database дозволяє звертатися до даних з будь-якого додатку. Oracle Database є комерційною СУБД, но є її безкоштовна версія, яку можна без проблем скачати прямо з офіційного сайту компанії.

Microsoft SQL Server. Система Microsoft SQL Server відштовхується від Концепції Платформи Даних Майкрософт: вона спрощує управління будь якими даними в будь-якому місці и в будь-який момент часу. Вона дозволяє зберігати в базах даних інформацію, отриману з структурованих, напівструктурованих и неструктурованих джерел, таких як зображення та музика. У SQL Server 2014 є великий набір інтегрованих служб, Які розширюють можливості використання даних: можна складати запити, виконувати пошук, проводити синхронізацію, розробляти звіти, аналізувати дані. Всі дані зберігаються на основних серверах, що входять до складу центру обробки даних. До них здійснюється доступ з настільних комп'ютерів або мобільних пристроїв. Таким чином, можна Повністю контролювати дані.

SQL Server дозволяє створити надійну, продуктивну, інтелектуальну платформу, що відповідає всім вимогам по роботі з даними. SQL Server є комерційною СУБД [13].

PostgreSQL. Безкоштовна об'єктно-реляційна система управління базами даних. Систему PostgreSQL засновано на ядрі, створення безліччю розробників. У подібних випадках розумно зосередитися на оснащенні системи новими можливостями, але не займатися оптимальним їх втіленням, оскільки у випадку виникнення необхідності завжди можна буде повернутися до оптимізації відповідних ділянок коду.

MySQL. Безкоштовна система управління базами даних. MySQL є власністю компанії Oracle Corporation, що отримала її разом з поглиненою Sun Microsystems, що здійснює розробку і підтримку програми. Розповсюджується під GNU General Public License або під власною комерційною ліцензією. Крім

цього, розробники створюють функціональність за замовленням ліцензійних користувачів. MySQL є рішенням для малих і середніх додатків. Входить до складу серверів WAMP, LAMP і в портативні збірки серверів Denver, XAMPP.

Зазвичай MySQL використовується як сервер, до якого звертаються локальні або віддалені клієнти, проте в дистрибутив входить бібліотека внутрішнього сервера, що дозволяє включати MySQL в автономні програми.

Оскільки інтерфейс системи буде веб-додатком, потрібно обрати СУБД, з якою можуть співпрацювати мови програмування для написання веб-додатків. Вибір було зупинено на MySQL. Висока продуктивність MySQL на вузькому колі задач – прямий наслідок її функціональної простоти.

Отже, для наших цілей буде обрано MySQL. У back-end розробці найбільш поширеною є зв'язка PHP + MySQL. PHP має достатньо засобів для організації повномасштабної роботи з базою даних.

Висновки до розділу 1

Поглиблене вивчення тематики створення інтернет магазину дало підстави визначити основні задачі, що необхідні для реалізації проекту.

Огляд популярних реалізованих інтернет магазинів дав підстави для визначення обґрунтованої архітектури для реалізації власного інтернет магазину одягу. Після вибору підходящої архітектури були поставлені додаткові питання щодо реалізації, особливостей та методів які відповідають обраній архітектурі.

Визначено основну функціональність проекту, як клієнтської так і адміністративної частини.

Для реалізації проекту в даному розділі було проаналізовано та вирішено використовувати наступне:

- мову програмування PHP 7 [10];
- інтегроване середовище розробки JetBrains PhpStorm 2021.1.4 x64;
- базу даних MySQL;
- засоби реалізації клієнтської частини – JavaScript, CSS, HTML5 [14].

РОЗДІЛ 2. ПРОЕКТУВАННЯ ІНТЕРНЕТ МАГАЗИНУ ОДЯГУ

2.1. Визначення варіантів використання та об'єктно-орієнтованої структури системи

Програмна реалізації інтернет магазину одягу має певні цілі. Найголовнішою ціллю реалізації даної програми є пришвидшення та збільшення об'ємів продаж товарів компанії, донесення актуальної інформації про послуги та товари компанії до користувачів, спрощення обрання та пошуку товарів. Реалізувавши інтернет магазин за сучасними вимогами та рекламуючи його можна буде зробити певні висновки.

Бізнес вимоги

1. Основні цілі: оптимізація процесу пошуку товарів, автоматизація процесу оформлення замовлення, збільшення об'єму продаж.
2. Можливості: надання можливості підбору товару.
3. Представлення проекту: проект буде реалізовано у вигляді веб додатку – інтернет магазину.

Вимоги користувачів

1. Перегляд основної інформації про магазин та контакти;
2. Можливість пошуку та перегляду запитань магазину;
3. Можливість пошуку та підбору товарів;
4. Перегляд товарів, що рекомендує компанія;
5. Можливість детального перегляду інформації про товар;
6. Можливість додавання товарів у власний кошик;
7. Можливість оформлення замовлення та зміни замовлення;
8. Можливість пошуку товарів за назвою або частиною назви та характеристиками;
9. Можливість реєстрації на сайті.

Вимоги адміністратора

1. Додавання товарів (видалення, редагування, пошук);
2. Додавання категорій товарів (видалення, редагування, пошук);
3. Додавання характеристик товарів;
4. Пошук та сортування товарів;
5. Адміністрування замовлень (зміна статусу, видалення).

Вимоги авторизованого користувача

1. Реєстрація на сайті;
2. Перегляд інформації та статусу власних замовлень.

Функціональні вимоги:

1. Покупка в інтернет магазині: в системі повинна бути представлена можливість покупки товарів на основі введених параметрів або використання параметрів за замовчуванням.
2. Реалізація технологій для швидкого пошуку.

Нефункціональні вимоги:

1. Сприйняття
 - час завантаження сторінки не повинен перевищувати 1с.
 - інтерфейс додатку повинен бути інтуїтивно зручним для користувача та не вимагати від нього додаткової підготовки
2. Продуктивність
 - додаток має бути адаптивним та кроссбраузерним

Системні вимоги:

Система повинна підтримуватися комп'ютером, який знаходиться в наступній комплектації:

- операційна система Windows 7 або вище;
- 32-розрядний (x86) або 64-розрядний (x64) процесор із тактовою частотою 1 ГГц або вище;
- 1 ГБ (для 32-розрядних процесорів) або 2 ГБ (для 64-розрядних процесорів) ОЗУ;

- 16 ГБ (для 32-розрядної системи) або 20 ГБ (для 64-розрядної системи) вільного місця на жорсткому диску;
- дисплей: 800 x 600, 256 кольорів (мінімум);
- графічний пристрій DirectX 9 або пізнішої версії.

Аналіз вимог користувачів дав підстави для визначення варіантів використання інтернет магазину. На рис.2.1 наведено діаграму використання програмної системи.

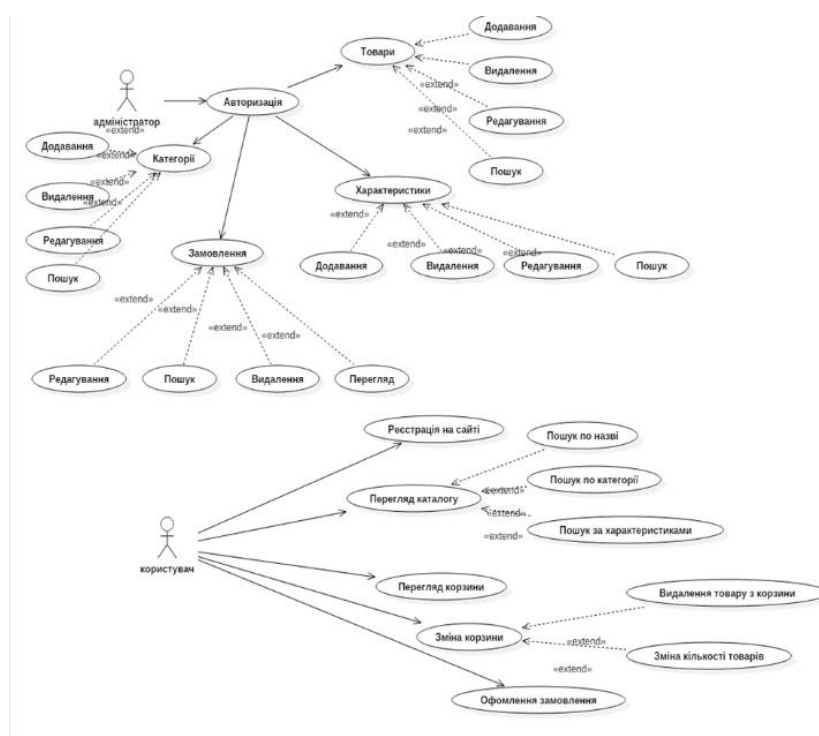


Рисунок 2.1 - Варіанти використання інтернет магазину

Діаграма дій користувача між сторінками - це схема роботи користувача з функціональністю системи. Така діаграма показує, як користувач виконує одну зі своїх задач, переходячи від сторінки до сторінки. Діаграми переходів між сторінками охоплюють не всю систему, а її конкретний модуль і малюють процес роботи користувача з ним в усіх дрібницях. Крім того, якщо в карті сайту і схемі навігації важливі ієрархічні відносини «Розділ / сторінка», то описувані діаграми концентруються на послідовності переходу між ними. Приклад діаграми переходів між сторінками наведемо нижче.

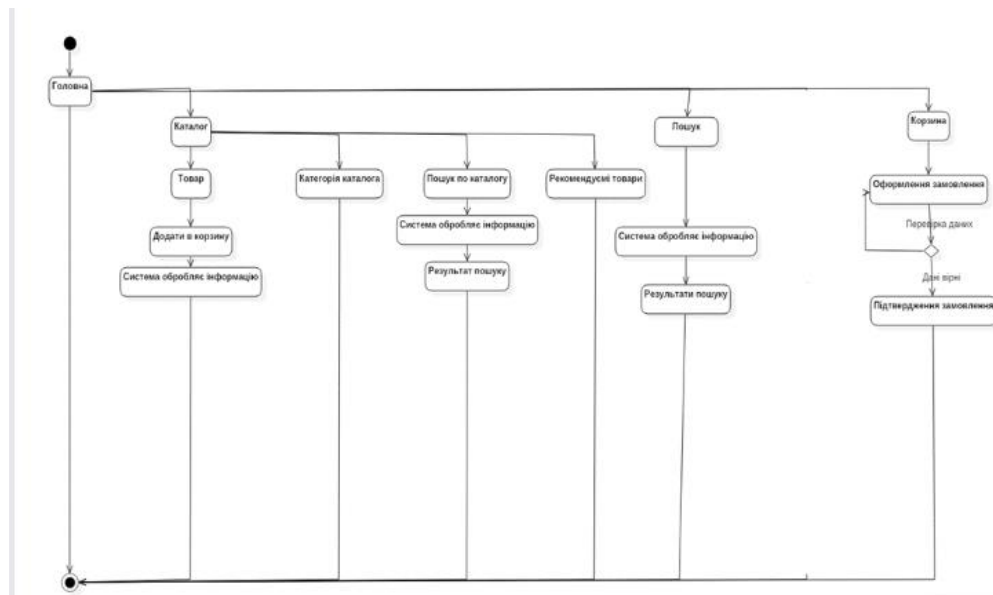


Рисунок 2.2 - Приклад діаграми дій між сторінками

Алгоритм розпочинається головною сторінкою, на якій кожен може зайти в кабінет чи зареєструватися. Також можна вибирати категорію товарів, після чого користувач побачить список продуктів, який він може відфільтрувати чи відсортувати. Вибравши певний товар, він може дізнатися про нього детальну інформацію, після чого може повернутися назад до каталогу або додати до корзини бажане. Після цього він може покинути сайт або продовжити вибирати.

Щодо адміністратора, при вході у свій кабінет відбувається перевірка на адміністратора. Отримавши доступ, користувач може додавати чи видаляти продукти, категорії, характеристики товарів та замовлення.

На діаграмі діяльності, або активності зображено процес роботи користувача із інтернет магазином рис.2.3.

З головної сторінки неавторизований користувач може здійснити пошук товарів, обрати категорії товарів, зайти одразу у каталог продукції, переглянути запитання, контакти.

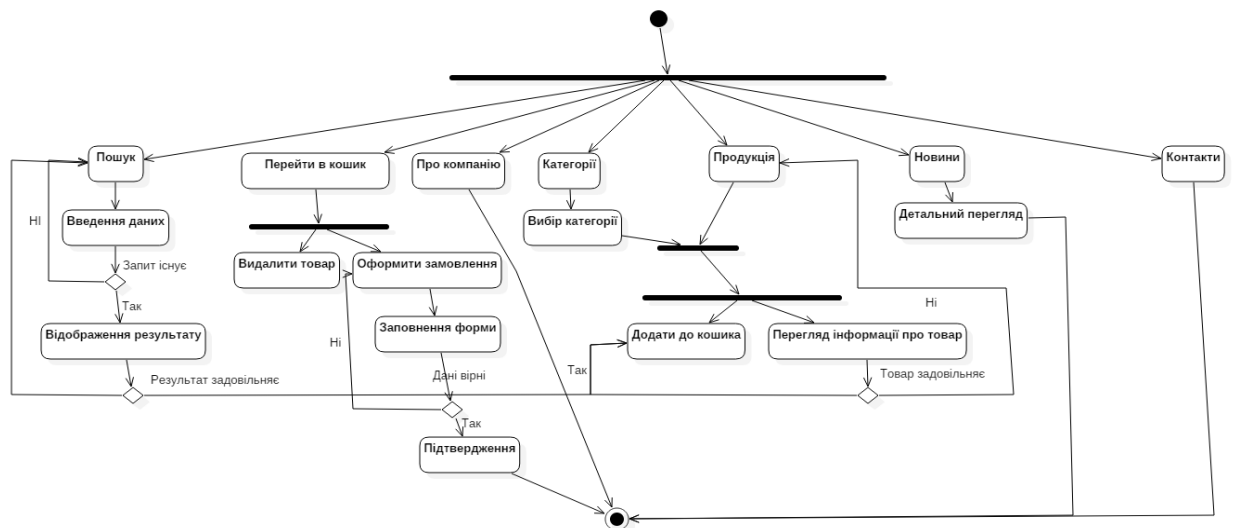


Рисунок. 2.3 - Діаграма діяльності

Завдяки виявленню функціональних вимог було виділено наступні сутності для програмної реалізації додатку.

2.2. Опис структури даних з якими працює інтернет магазин

Інтернет магазин одягу працює з даними, що збережені в базі даних MySQL. В програмному коді реалізована функціональність для отримання даних з бази. Сама база даних має назву store та її структура наведена нижче (рис.2.5). База даних спроектована в редакторі phpMyAdmin та складається із 6 таблиць опис яких наведено в таблиці 2.1. За принципами проектування реляційних баз даних всі зв'язні таблиці мають зв'язки та тригери, що підтримують цілісність даних в таблицях.

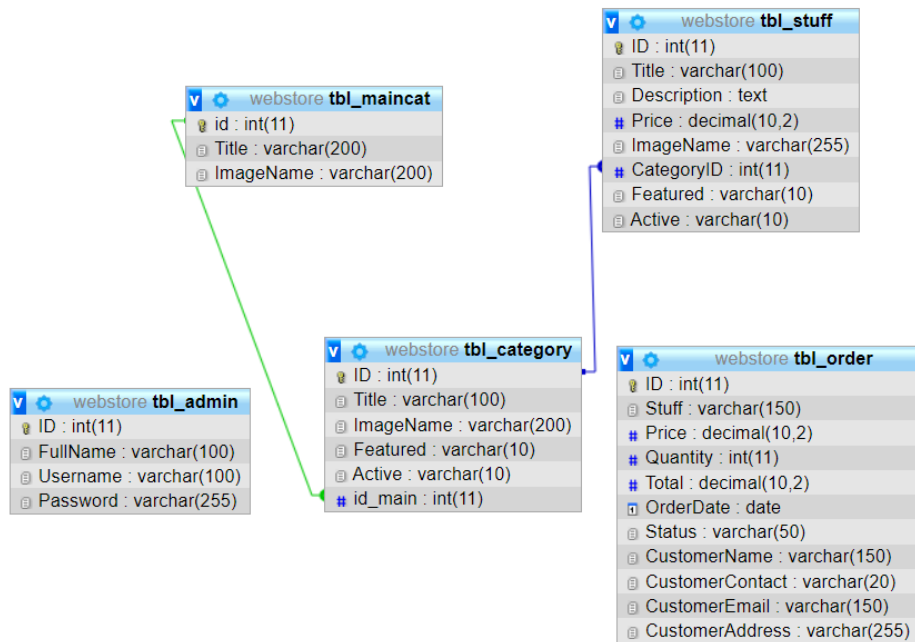


Рисунок 2.5 – Структура бази даних

2.3. Реалізація інтернет магазину одягу

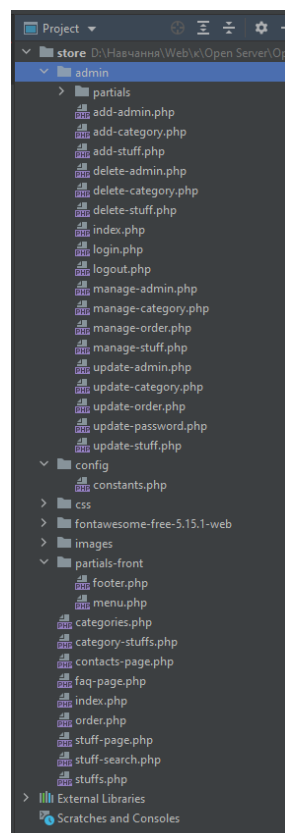


Рис.2.8 - Структура проекту

Виділемо основні php файли які слід розглянути

index.php (головна сторінка сайту)

Містить дизайн головної сторінки сайту, на ній виводяться рекомендовані категорії, товари. На сторінці присутній пошук та навігаційна панель.

Розглянемо php код, який по суті буде зразковий, адже таким же чином або дуже схожим розміщуються категорії та товари на інших сторінках сайту.

Щоб вивести категорії на сторінку, створюємо змінну в який буде виконуватись SQL запит, який ми далі виконуємо. Підраховуємо рядки, щоб перевірити доступна категорія чи ні. Після чого, за допомогою розгалуження будемо виводити дані на сторінку спочатку отримавши дані з БД та перевіривши чи є категорія, також зробивши перевірку чи є зображення категорії.

Лістинг коду:

```
<?php

//створити запит sql для відображення категорій з БД
$sql = "SELECT * FROM tbl_category WHERE Active = 'Yes' AND Featured = 'Yes'
LIMIT 3";
//виконати запит
$res = mysqli_query($conn, $sql);
//підрахуйте рядки, щоб перевірити, доступна категорія чи ні
$count = mysqli_num_rows($res);

if($count>0)
{
    //доступні категорії
    while($row=mysqli_fetch_assoc($res))
    {
        //отримати значення, такі як id, title тощо
        $id = $row['ID'];
        $title = $row['Title'];
        $imageName = $row['ImageName'];
        ?>

        <a href="<?php echo SITEURL; ?>/category-stuffs.php?category_id=<?php
echo $id;?>">
            <div class="box-3 float-container">
                <?php

                // перевіряємо доступність зображення чи ні
                if($imageName=="")
                {
                    //відобразити повідомлення
                    echo "<div class='error'>Зображення недоступне</div>";
                }
            }
        }
    }
}
```

```

        else
        {
            // доступне зображення
            ?>

            <?php
        }
    ?>

    <h3 class="float-text text-white"><?php //echo $title; ?></h3>
</div>
</a>

<?php
}
}
else
{
    // категорії недоступні
    echo "<div class='error'>Категорія не додана</div>";
}
?>

```

Одним з найважливіших php файлів у проєкті є цей файл розміщений в спеціальній папці config для зручної роботи та користування цим кодом

constants.php

Після старту сесії ми створюємо константи змінні, які будуть використовуватися в нашому проєкті постійно.

Це константи переходу на головну сторінку сайту, підключення до БД, щоб брати, заносити, видаляти та редагувати в ній дані.

Лістинг коду:

```

<?php

//розпочати сеанс
ob_start();
session_start();

//Створіть константи для зберігання значень
define('SITEURL', 'http://store');
define('LOCALHOST', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', 'root');
define('DB_NAME', 'webstore');

$conn = mysqli_connect(LOCALHOST, DB_USERNAME, DB_PASSWORD) or
die(mysqli_error()); //підключення до бази даних
$dbSelect = mysqli_select_db($conn, DB_NAME) or die(mysqli_error()); //база
даних відбору

```

?>

Оскільки наш попередній код буде використовуватися протягом всього проекту, то нам слід розумно його включити на всі наші сторінки.

Це ми зробимо однією строкою коду в таких файлах

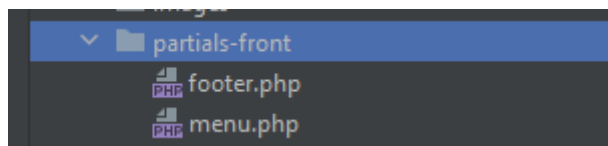


Рис.2.9.1 - Футер та навігаційна панель сайту головної сторінки

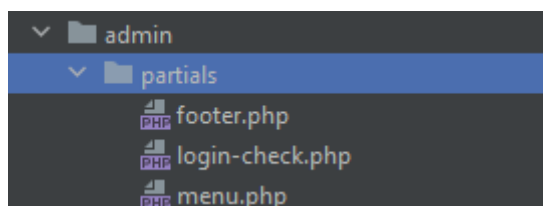


Рис.2.9.2 - Футер та навігаційна панель для сторінок адміна

У цих файлах ми верстаємо навігаційну панель та футер, який буде однією строкою коду підключатися до наших сторінок.

Головне в файлах `menu.php` підключити `php` файл з нашими константами, для того, щоб надалі його зручно використовувати в нашому проекті

Лістинг коду:

```
<?php include('config/constants.php'); ?>
```

Також слід пояснити, як відбувається додавання даних до нашої БД

Усі маніпуляції над даними може робити тільки адміністратор.

`add-admin.php`

Такий код, звичайно змінений та підлаштований під категорії, товари та замовлення, буде використовуватися в інших файлах.

Оскільки ми будемо вводити дані у форму, то нам їх треба якось додати.

Додавання відбувається після натиснення відповідної кнопки на сторінці.

По коду ми перевіряємо чи була натиснута кнопка, далі отримуємо дані з форми за відповідними назвами форми, також шифруємо пароль за допомогою `md5`.

Далі створюємо запит SQL та заносимо дані в нашу таблицю в БД та виводимо повідомлення при успішному занесенні або при помилці на сорінку керування адміністраторами.

Лістинг коду:

```
<?php
//Обробка форми значення та збережить її в базі даних

//Перевірка, натиснуто кнопку чи ні

if(isset($_POST['submit']))
{
    //клацнула кнопка
    //echo "Button clicked";

    //1.отримати дані з форми
    $fullName = $_POST['full name'];
    $username = $_POST['username'];
    $password = md5($_POST['password']); //Шифрування пароля за допомогою md5

    //2.Створити запит для збереження даних у базі даних
    $sql = "INSERT INTO tbl_admin SET
        FullName = '$fullName',
        Username = '$username',
        Password = '$password'
    ";

    //3.Виконання запитів та збереження даних у базі даних
    $res = mysqli_query($conn, $sql) or die(mysqli_error());

    //4. Перевірка, чи введені дані (Запит виконується) чи ні
    // і відобразити відповідне повідомлення
    if($res==TRUE)
    {
        //Дані вставлено
        //echo "Data inserted";
        //Створення змінної сеансу для відображення повідомлення
        $_SESSION['add'] = "Адміністратор доданий успішно";
        //Переадресація
        header("location:".SITEURL.'/admin/manage-admin.php');
        //header('Location: http://store/admin/manage-admin.php');
    }
    else
    {
        //не вдалося вставити дані
        //echo "failed to insert data";
        //Створення змінної сеансу для відображення повідомлення
        $_SESSION['add'] = "Помилка додавання адміністратора";
        //Сторінка переадресації для додавання адміністратора
        header("location:".SITEURL.'/admin/add-admin.php');
    }
}

?>
```

У таблиці категорії та товарів, присутнє поле для зображення, адже ми потім будемо завантажувати зображення на нашу головну сторінку і на сторінку керування, тому слід акцентувати увагу на те, як завантажується зображення в БД

add-stuff.php (частина завантаження зображення в БД)

Після перевірки натиснуто кноаку завантаження чи ні, ми будемо присвоювати нове ім'я нашому зображенню, щоб не було проблем з занесенням його до БД.

Після перейменування ми вказуємо вихідний шлях, тобто поточне розташування файлу та шлях призначення, в якому буде зберігатися наш новий файл.

Далі ми вже завантужуємо наше нове зображення та перевіряємо чи воно завантажилось.

Лістинг коду:

```
//2. Завантажити зображення, якщо вибрано
//перевірка, натиснуто кнопку завантаження чи ні
if(isset($_FILES['image']['name']))
{
    //клацнула кнопка завантаження
    $imageName = $_FILES['image']['name']; // нове ім'я зображення

    //перевірка доступності файлу
    if($imageName!="")
    {
        //доступне зображення

        //А. завантаження нового зображення

        // автоматично перейменувати наше зображення
        // отримуємо розширення нашого зображення (jpg, png, gif, etc) e.g.
        specialstuff1.jpg
        //$ext = explode('.', $imageName);
        $dot = ".";
        $array = explode($dot, $imageName);
        $ext = end($array);

        //перейменувати зображення
        $imageName = "Stuff_Name_" . rand(000, 999) . '.' . $ext; // e.g.
        Stuff_Category_123.jpg

        $sourcePath = $_FILES['image']['tmp_name'];

        $destinationPath = "../images/stuff/" . $imageName;

        //завантаження зображення
        $upload = move_uploaded_file($sourcePath, $destinationPath);
    }
}
```

```

        //перевірка, чи завантажено зображення чи ні
        //і якщо зображення не буде завантажено, ми зупинимо процес і
        переадресуємо з повідомленням про помилку
        if ($upload == false) {
            //встановити повідомлення
            $_SESSION['upload'] = "<div class='error'>Помилка завантаження
зображення</div>";
            //Переспрямування на керування сторінкою категорії
            header("location:".SITEURL.'/admin/manage-stuff.php');
            //зупинит процес
            die();
        }
    }
    else
    {
        $imageName = $currentImage; //зображення за замовчуванням, коли
зображення не вибрано
    }
}
else
{
    $imageName = $currentImage;
}
}

```

delete-category.php

У видаленні даних головне правильно вказати SQL запит, інший код так чи інакше вже був продемонстрований

Лістинг коду:

```

<?php

include("../config/constants.php");

//echo "Delete page";
//перевірка, чи встановлено значення ідентифікатора та Ім'я зображення чи ні
if(isset($_GET['id']) AND isset($_GET['imageName']))
{
    //отримати значення та видалити
    //echo "Get value and delete";
    $id = $_GET['id'];
    $imageName = $_GET['imageName'];

    //видалити файл фізичного зображення доступний
    if($imageName != "")
    {
        //зображення доступне. тому видаляємо його
        $path = "../images/category/".$imageName;
        //видалити зображення
        $remove = unlink($path);

        //якщо не вдається видалити зображення, то буде помилка і зупинка
        процесу
        if($remove==false)
        {
            //встановити повідомлення сеансу
            $_SESSION['remove'] = "<div class='error'>Помилка видалення
категорії</div>";
        }
    }
}

```

```

        //переспрямування для управління категорією
        header("location:".SITEURL.'/admin/manage-category.php');

        //зупинити процес
        die();
    }
}

//видалити дані з бази даних
//запит sql видалення даних з БД
$sql = "DELETE FROM tbl_category WHERE ID=$id";

//виконати запит
$res = mysqli_query($conn, $sql);

//перевірити, чи видаляються дані з БД чи ні
if($res==true)
{
    //встановити повідомлення про успіх і перенаправити
    $_SESSION['delete'] = "<div class='success'>Категорія видалена
успішно</div>";
    //переспрямування для управління категоріями
    header("location:".SITEURL.'/admin/manage-category.php');
}
else
{
    //використовуємо повідомлення про помилку та переспрямування
    $_SESSION['delete'] = "<div class='error'>Помилка видалення
категорії</div>";
    //переспрямування для управління категоріями
    header("location:".SITEURL.'/admin/manage-category.php');
}
}
else
{
    //переспрямування для управління категоріями
    header("location:".SITEURL.'/admin/manage-category.php');
}
}
?>

```

Висновки до розділу 2

В розділі детально розглянута структура та вимоги до даних з якими працює система; спроектований та описаний алгоритм роботи системи в цілому; визначені конкретні вимоги та варіанти використання.

Завдяки проектуванню діаграм були визначені основні об'єкти в системі та зв'язки між ними. Це дало змогу правильно оцінити структуру програмного комплексу і заздалегідь вирішити певні проблеми які могли б виникнути в процесі реалізації.

Була описана логіка роботи інтернет магазину, розглянуті її певні ключові моменти. Також, був представлений програмний код, що демонструє основні моменти роботи інтернет магазину.

					ІІЗ.КР.Б – 121 – 23 – ІІЗ	
						36

РОЗДІЛ 3. ІНТЕРФЕЙС ТА ПОРЯДОК РОБОТИ З ІНТЕРНЕТ МАГАЗИНОМ ОДЯГУ

3.1. Архітектура програмного комплексу та його розгортання

Для розробки сайту на базі будь-якої CMS існує два основних підходи: 1. Розгорнути систему на локальному комп'ютері, потім створити сайт, наповнити вмістом і потім перенести на сервер. Установка CMS на локальний комп'ютер досить проста на комп'ютер необхідно встановити Сервер Apache, мова PHP, базу даних MySQL. Це реалізується установкою на комп'ютер програмної платформи XAMPP або OpenServer і її налаштуванням. Після установки XAMPP засобами компонента phpMyAdmin створюється база даних. Мінімальні вимоги до системи:

1. Процесор 1 ГГц.
2. ОЗУ 512 МБ.
3. Мінімальне місце на диску для 32-розрядної системи 850 МБ та 2 ГБ для 64-розрядної.
4. Підтримка бази даних

Таким чином розроблений веб сайт не вимагає спеціалізованого апаратного забезпечення, додаткових налаштувань та засобів для розгортання крім стандартних для більшості сучасних настільних Windows додатків.

На рис. 3.1 представлена діаграма розгортання системи. З діаграми видно що дана система має три вузли: програма, інтерфейс та користувач.

Вузол WebServser Apache – це той компонент який виконує програму. Для виконання програми вузол також використовує базу даних що містить таблиці з даними для роботи сайту.

Вузол Web browser – це компонент що є перехідним від компоненту *Програма* до компоненту *Користувач*. Даний вузол забезпечує взаємодію між програмою та користувачем.

Вузол Користувач – це компонент що відповідає за дії користувача та містить наступні компоненти:

					ІПЗ.КР.Б – 121 – 23 – ПЗ	37

- запуск програми;
- введення даних;
- оформлення замовлення;
- пошук даних.

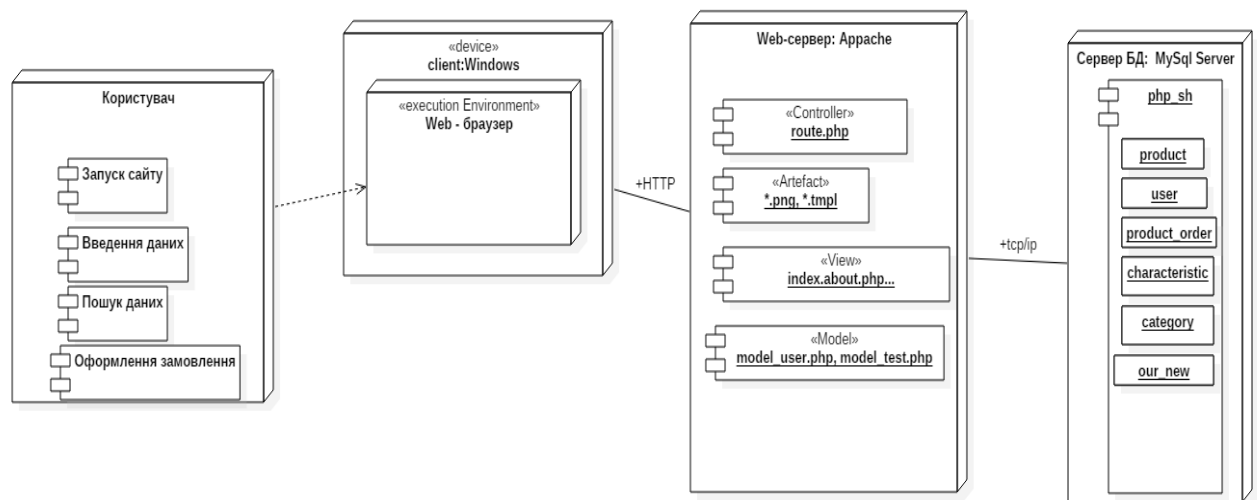


Рисунок 3.1 - Діаграма розгортання системи

3.2. Структура інтерфейсу та порядок роботи з інтернет магазином одягу

На даному етапі курсової роботи буде продемонстрована робота вже реалізованого ІМ. Буде протестовано можливості сайту, весь його функціонал та можливості.

Почнемо з головної сторінки

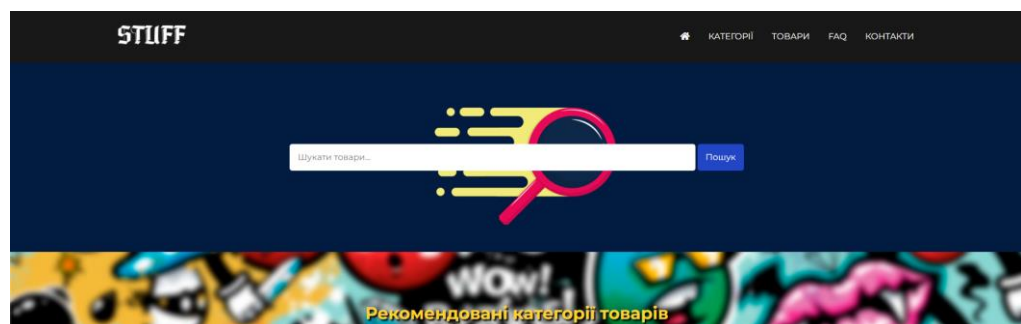


Рис.3.1 - Навігаційна панель та пошук

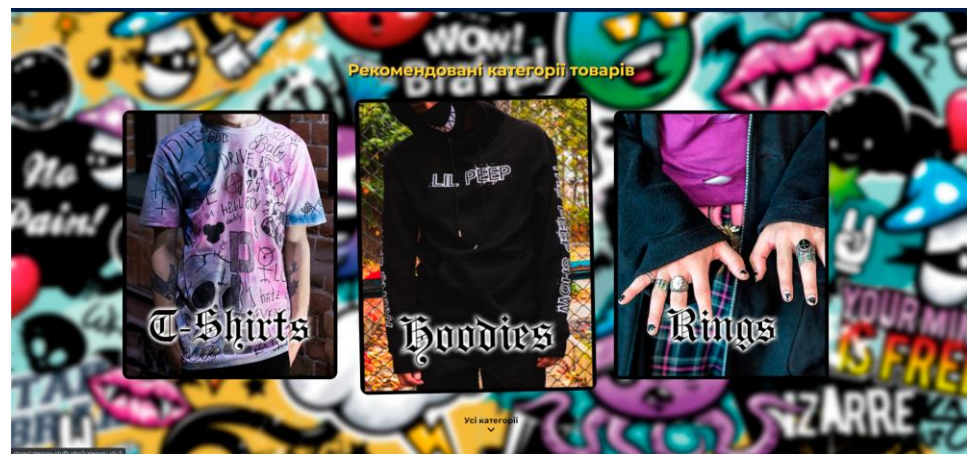


Рис.3.2 - Рекомендовані категорії

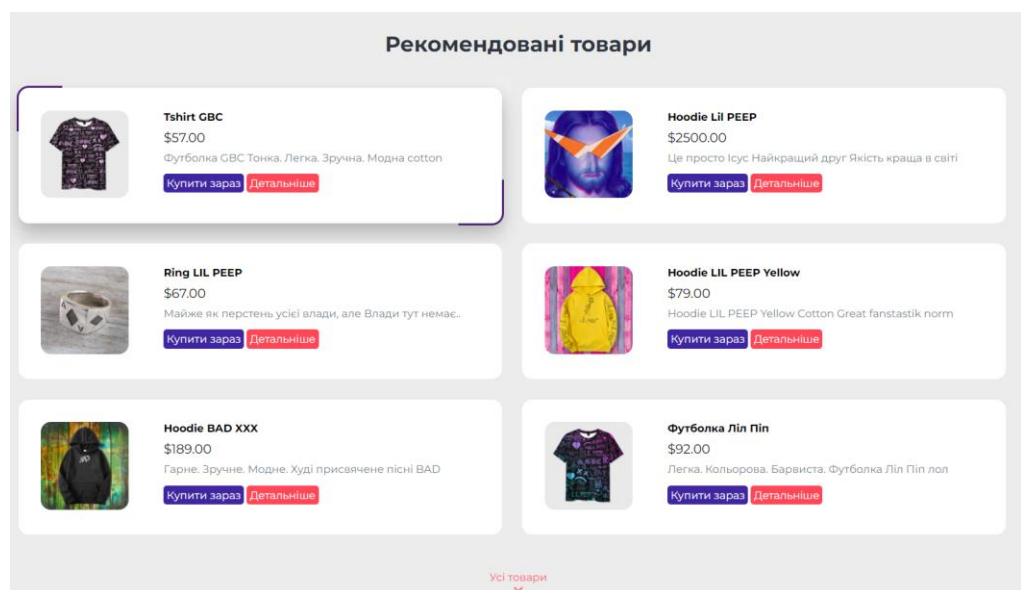


Рис.3.3 - Рекомендовані товари

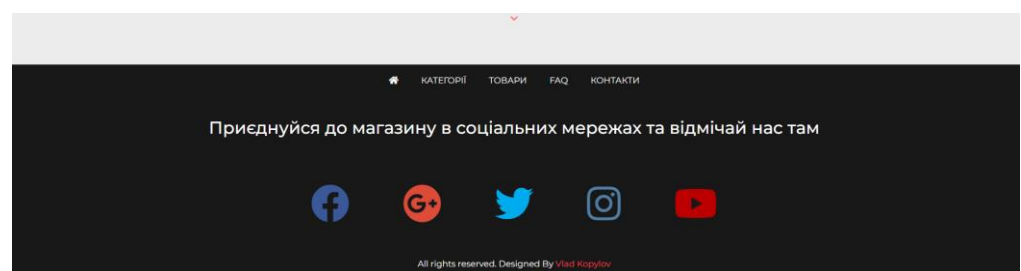


Рис.3.4 - Футер сайту

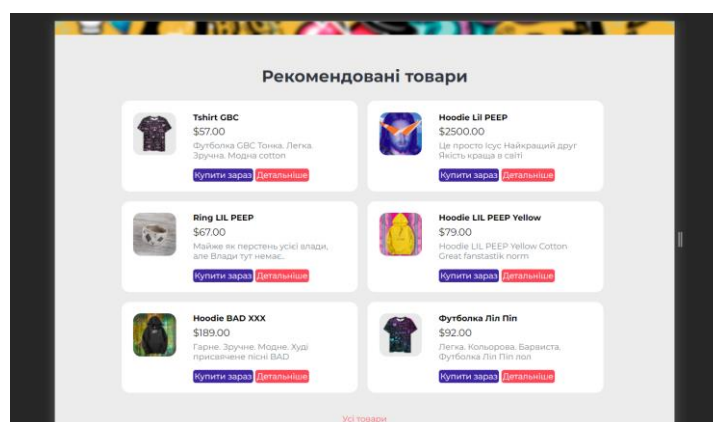
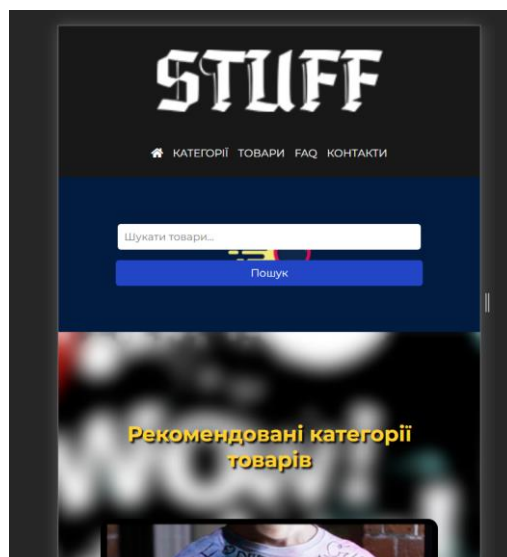


Рис3.5 - Приклад адаптивності сайту

Якщо натиснути на кнопку «Детальніше» то перейдемо на сторінку товару

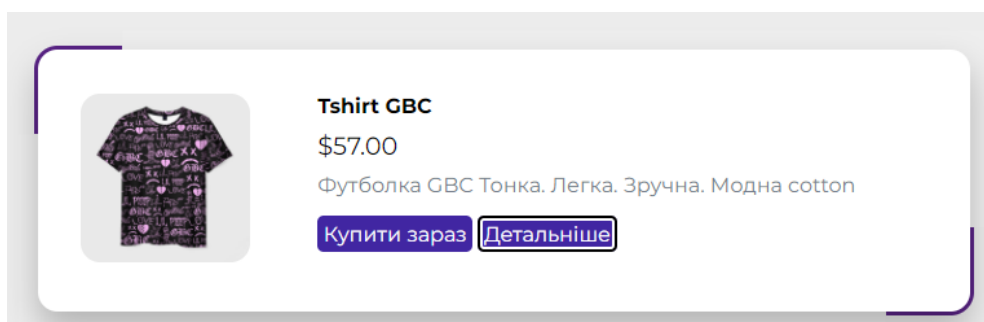


Рис.3.6.1 - Детальніше

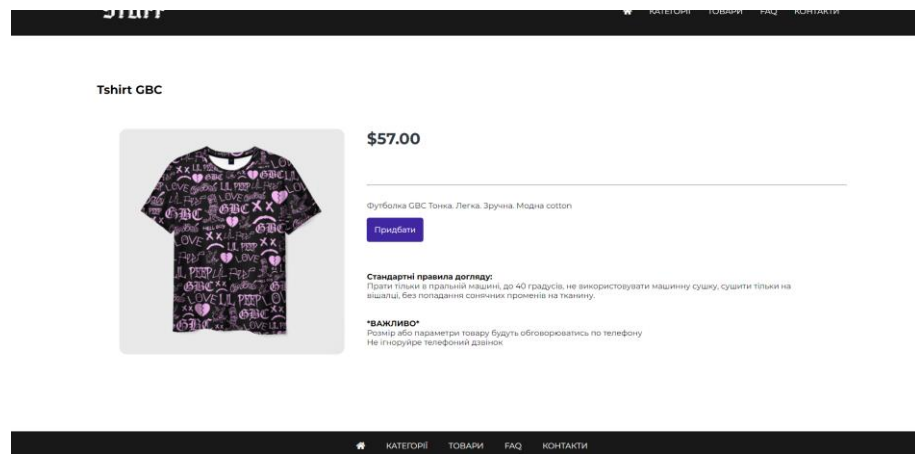


Рис.3.6.2 - Сторінка товару

Якщо натискнути на «Купити зараз», то ми перейдемо на сторінку замовлення, у якій ми вказуємо потрібні нам дані

Заповніть форму для підтвердження замовлення

Вибраний товар



Hoodie LIL PEEP Yellow
\$79.00
Кількість
2

Деталі замовлення:

Повне ім'я
Владислав Копилов

Номер телефону
0985226722

Пошта
tammskay25@gmail.com

Адреса
м. Житомир вул. Пушкіна 25, кв.2212

[Підтвердити замовлення](#)

Рис.3.7.1 - Форма замовлення

Після успішного замовлення нам виведеться про це повідомлення на голвоній сторінці

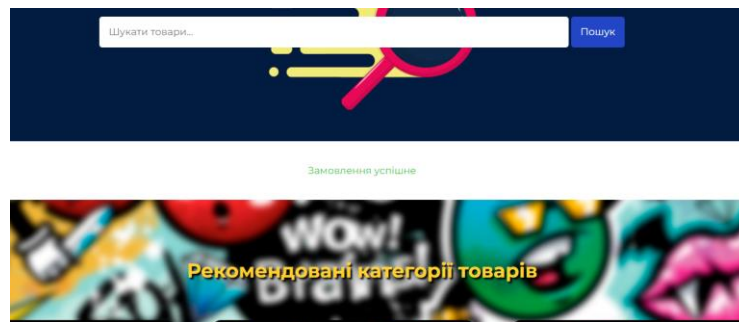


Рис.3.7.2 - Повідомлення про успішне замовлення товару

Пошук товару працює таким чином. Введем назву товару або його опис:

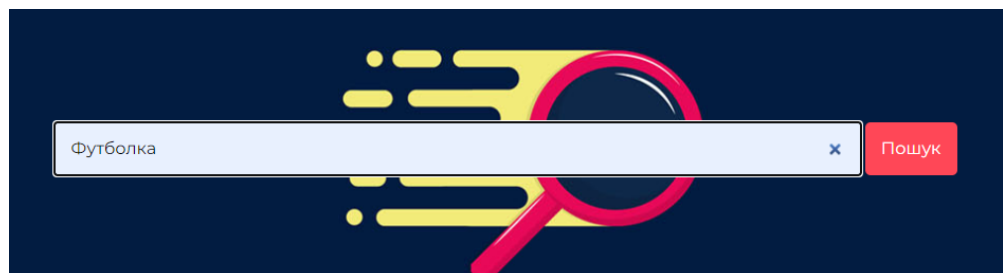


Рис.3.8.1 - Пошук

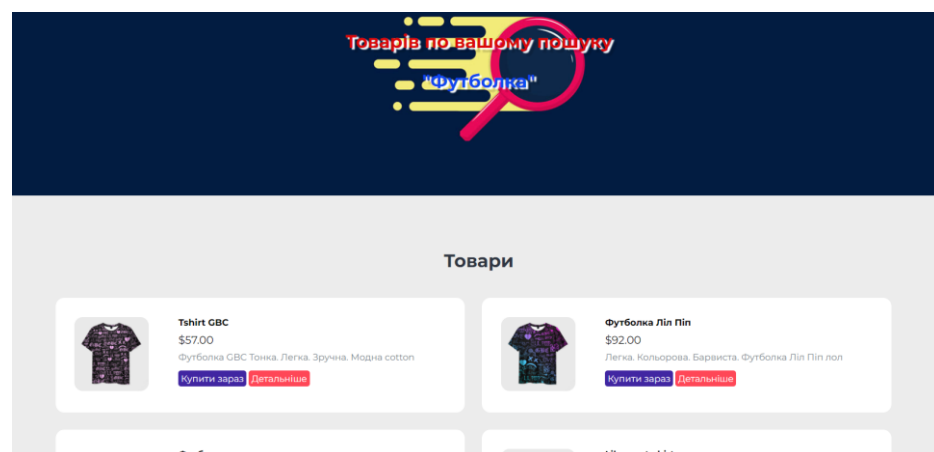


Рис.3.8.2 - Результат пошуку

Перевіримо сторінки, на які ми можемо перейти за допомогою меню

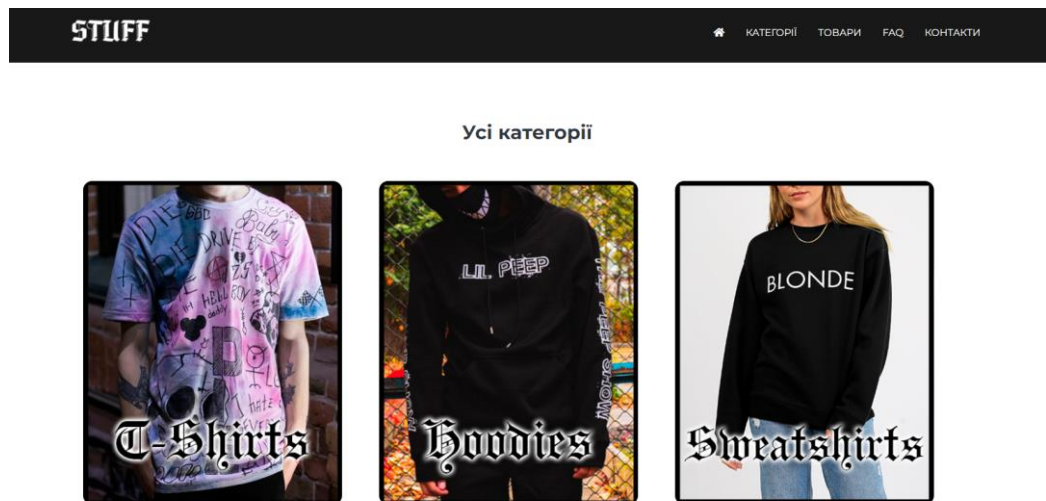


Рис.3.9.1 - Категорії

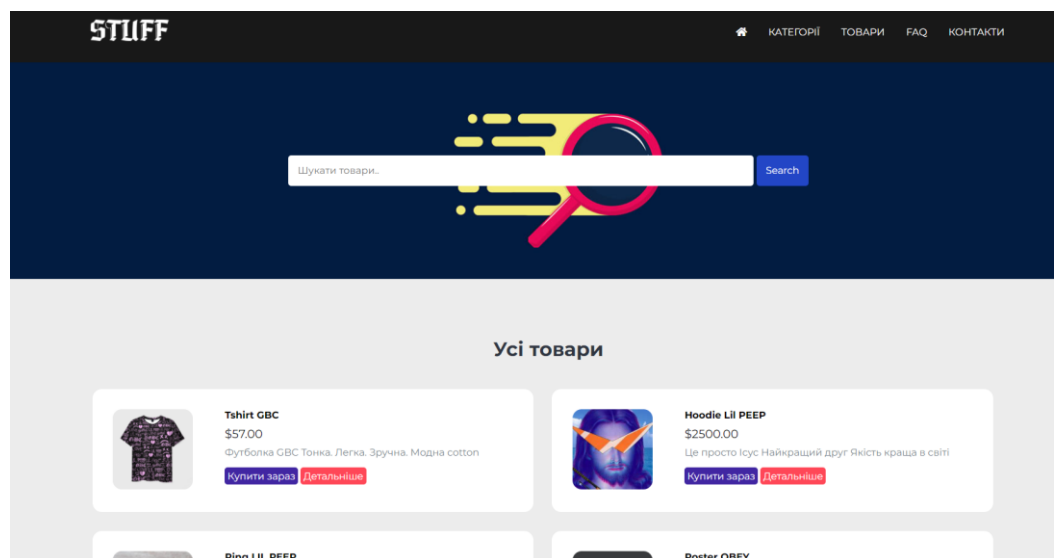


Рис.3.9.2 - Товари

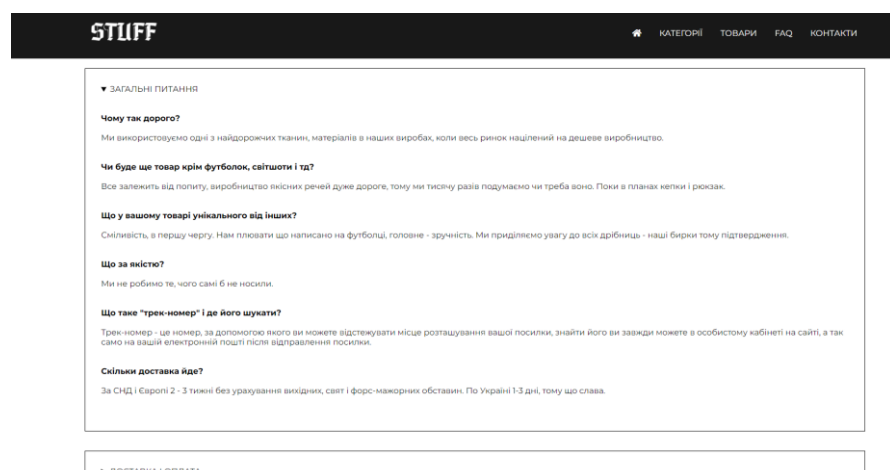


Рис.3.9.3 - Поширені товари

STUFF

🏠

КАТЕГОРІЇ

ТОВАРИ

FAQ

КОНТАКТИ

🛒 Stuff Store

8 800 535 3535

example25@daasas.com

вул.Якась, м.Житомир

Часи роботи

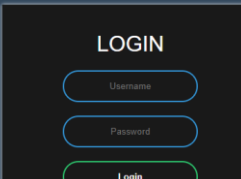
Пн-Пт: 9-18

Вихідні: 10-16

Додаткова інформація

8 800 535 3535

Основні можливості сайту показані, тепер перейдемо до сторінки адміністратора:

A login form with a dark background and rounded input fields. The form is centered on a dark blue background. It features the title "LOGIN" in white, followed by three rounded input fields: "Username" (with a blue border), "Password" (with a blue border), and "Login" (with a green border).

Якщо логін і пароль будуть введені невірні, то:

Login Failed

LOGIN

Username

Password

Login

Рис.3.10.2 - Невірно введено дані

У іншому випадку

Повідомлення про успішно авторизацію та інформаційна панель адміністратора:

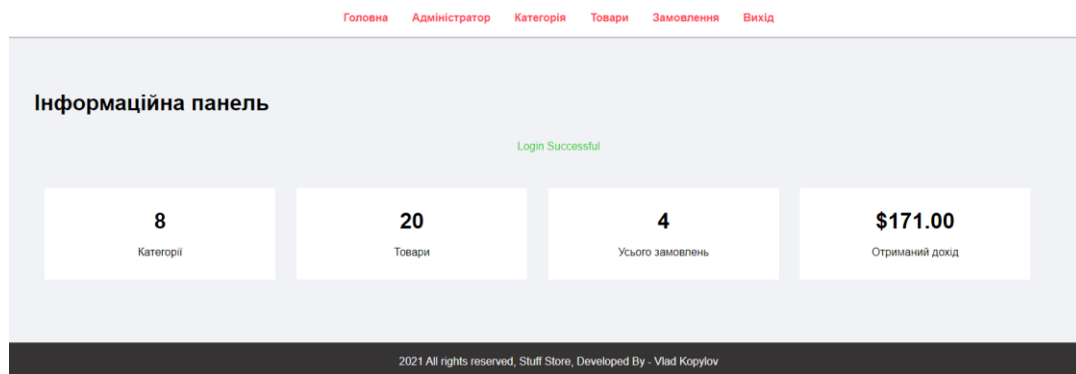


Рис.3.10.3 - Інформаційна панель та успішна авторизація

Пройдемося по меню:

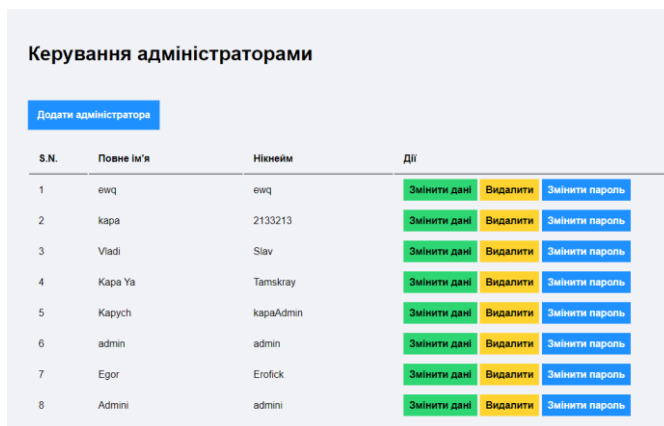


Рис.3.11.1 - Керування адміністраторами

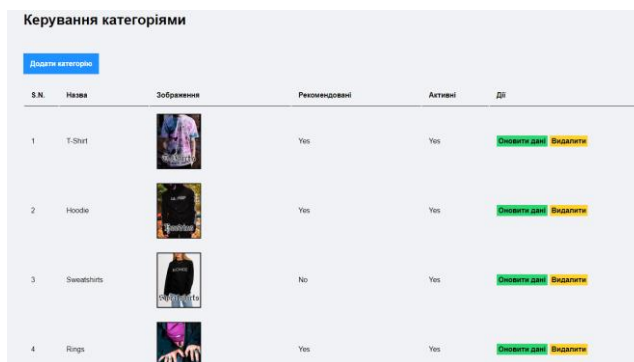


Рис.3.11.2 - Керування категоріями

Керування товарами

Додати товар

№	Назва	Ціна	Зображення	Рекомендовані	Активні	Дії
1	T-shirt OBC	57.00		Yes	Yes	Оновити дані Видалити
2	Hoodie LI PEEP	2500.00		Yes	Yes	Оновити дані Видалити
3	Ring LI PEEP	67.00		Yes	Yes	Оновити дані Видалити
4	Poster OBEY	524.00		No	Yes	Оновити дані Видалити

Рис.3.11.3 - Керування товарами

Головна

Адміністратор

Категорії

Товари

Замовлення

Вихід

Керування замовленнями

S.N. Товар	Ціна	Кількість	Товар	Дата замовлення	Статус	Ім'я клієнта	Контакти	Email	Адреса	Дії	
1	Hoodie LIL PEEP Yellow	79.00	2	158.00	2021-01-27	Ordered	Владислав Кошлов	0985226722	tammskay25@gmail.com	м. Житомир вул. Пушкіна 25, кв.2212	Оновити дані
2	Tshirt GBC	57.00	3	171.00	2021-01-25	Delivered	Карпа22	0985236729	mirkpylov25@gmail.com	atstefas	Оновити дані
3	Hoodie BAD XXX	189.00	3	567.00	2021-01-25	On Delivery	Кіо	098522729	tam3kay25@gmail.com	waifsh	Оновити дані
4	Ring LIL PEEP	67.00	3	201.00	2021-01-25	Cancelled	Капа	0985226729	tammskay25@gmail.com	fafefas	Оновити дані

Рис.3.11.4 - Керування замовленнями

Тепер переглянемо як виглядають форми додачі та оновлення на прикладів адміністратора та замовлення

Додавання Адміністратора

Повне ім'я:

Нікнейм:

Пароль:

Керування адміністраторами

Адміністратор доданий успішно

№	Повне ім'я	Нікнейм	Дії
1	elwq	elwq	Змінити дані Видалити Змінити пароль
2	kapra	2133213	Змінити дані Видалити Змінити пароль
3	Vladi	Slav	Змінити дані Видалити Змінити пароль
4	Капа Яа	Tamskray	Змінити дані Видалити Змінити пароль
5	Kapuch	kapraAdmin	Змінити дані Видалити Змінити пароль
6	admin	admin	Змінити дані Видалити Змінити пароль
7	Egor	Endick	Змінити дані Видалити Змінити пароль
8	Admini	admini	Змінити дані Видалити Змінити пароль
9	Vlad25	Tamskray777	Змінити дані Видалити Змінити пароль

Рис.3.12.1 - Результат

Також перевіримо БД

+ Параметри					
	ID	FullName	Username	Password	
☐	1	ewq	ewq	6bdcbb606161c47eab0615ff6e13313f	
☐	20	kapa	2133213	508b72dab5b32a519c5c8988b55aef0	
☐	21	Vladi	Slav	544f4e7aceab82fab9301d2d72a625	
☐	23	Kapa Ya	Tamskray	9bd5ee6fe55aeb673025dbcb8f939c1	
☐	26	Kapych	kapaAdmin	a680e33d814c13c83e712619e03a4bf2	
☐	27	admin	admin	202cb962ac59075b964b07152d234b70	
☐	28	Egor	Erofick	827ccb0eea8a706c4c34a16891f84e7b	
☐	29	Admini	admini	827ccb0eea8a706c4c34a16891f84e7b	
☐	30	Vlad25	Tamskray777	25f9e794323b453885f5181f1b624d0b	

Рис.3.12.2 - Результат

Як бачимо, все працює успішно

Оновимо дані про замовлення:

Оновити дані замовлення

Назва товару: **Hoodie LIL PEEP Yellow**

Ціна: **\$79.00**

Кількість:

Статус: **Delivered**

Ім'я клієнта:

Контакти:

Email:

Адреса:

Оновити дані замовлення

Керування замовленнями

Дані замовлення збережені успішно

S.N.	Товар	Ціна	Кількість	Товар	Дата замовлення	Статус	Ім'я клієнта	Контакти	Email	Адреса
1	Hoodie LIL PEEP Yellow	79.00	3	237.00	2021-01-27	Delivered	Арцрунн Ншан	0985226722	skray25@gmail.com	вул. Пушкіна 25

Рис.3.13.1 - Результат

Також перевіримо БД

☐ Показать все	Количество строк: 50	Фильтровать строки: Поиск в таблице	Сортировать по ключу: Нижнего										
+ Параметры													
☐	☐	☐	ID	Stuff	Price	Quantity	Total	OrderDate	Status	CustomerName	CustomerContact	CustomerEmail	CustomerAddress
☐	☐	☐	1	Ring LIL PEEP	67.00	3	201.00	2021-01-25	Cancelled	Kapa	0985226729	tamskray25@gmail.com	fafasfas
☐	☐	☐	2	Hoodie BAD XXX	189.00	3	567.00	2021-01-25	On Delivery	Kto	098522729	tam3kay25@gmail.com	waffsh
☐	☐	☐	3	Tshirt GBC	57.00	3	171.00	2021-01-25	Delivered	Kapa22	0985236729	mrik3pylov25@gmail.com	afsfasfas
☐	☐	☐	4	Hoodie LIL PEEP Yellow	79.00	3	237.00	2021-01-27	Delivered	Арцрунн Ншан	0985226722	skray25@gmail.com	вул. Пушкіна 25

Рис.3.13.2 - Результат

Дані успішно змінились

Також можна продемонструвати видалення адміністратора:

+ Параметри

	ID	FullName	Username	Password
☐	1	ewq	ewq	6bdcbb606161c47eab0615ff6e13313f
☐	20	kapa	2133213	508b72dab5b32a519c5c8988b55aeef0
☐	21	Vladi	Slav	544f4e7aceeab82ffab9301d2d72a625
☐	23	Kapa Ya	Tamskray	9bd5ee6fe55aaeb673025dbcb8f939c1
☐	26	Kapych	kapaAdmin	a680e33d814c13c83e712619e03a4bf2
☐	27	admin	admin	202cb962ac59075b964b07152d234b70
☐	28	Egor	Erofick	827ccb0eea8a706c4c34a16891f84e7b
☐	29	Admini	admini	827ccb0eea8a706c4c34a16891f84e7b
☐	30	Vlad25	Tamskray777	25f9e794323b453885f5181f1b624d0b

Рис.3.14.1 - До видалення

Керування адміністраторами

[Додати адміністратора](#)

S.N.	Повне ім'я	Нікнейм	Дії
1	ewq	ewq	Змінити дані Видалити Змінити пароль
2	kapa	2133213	Змінити дані Видалити Змінити пароль
3	Vladi	Slav	Змінити дані Видалити Змінити пароль
4	Kapa Ya	Tamskray	Змінити дані Видалити Змінити пароль
5	Kapych	kapaAdmin	Змінити дані Видалити Змінити пароль
6	admin	admin	Змінити дані Видалити Змінити пароль
7	Egor	Erofick	Змінити дані Видалити Змінити пароль

Рис.3.14.2 - Видаляємо адміна

Керування адміністраторами

Адміністратор видалений успішно

[Додати адміністратора](#)

S.N.	Повне ім'я	Нікнейм	Дії
1	ewq	ewq	Змінити дані Видалити Змінити пароль
2	kapa	2133213	Змінити дані Видалити Змінити пароль
3	Kapa Ya	Tamskray	Змінити дані Видалити Змінити пароль
4	Kapych	kapaAdmin	Змінити дані Видалити Змінити пароль
5	admin	admin	Змінити дані Видалити Змінити пароль
6	Egor	Erofick	Змінити дані Видалити Змінити пароль
7	Admini	admini	Змінити дані Видалити Змінити пароль
8	Vlad25	Tamskray777	Змінити дані Видалити Змінити пароль

Рис.3.14.3 - Повідомлення про видалення

Перевіримо БД

☐ Показати все
Количество строк: 50
Фильтровать строки: Поиск в та

	ID	FullName	Username	Password
☐	1	ewq	ewq	6bdcbb606161c47eab0615ff6e13313f
☐	20	kapa	2133213	508b72dab5b32a519c5c8988b55aeeef0
☐	23	Kapa Ya	Tamskray	9bd5ee6fe55aaeb673025dbcb8f939c1
☐	26	Kapych	kapaAdmin	a680e33d814c13c83e712619e03a4bf2
☐	27	admin	admin	202cb962ac59075b964b07152d234b70
☐	28	Egor	Erofick	827ccb0eea8a706c4c34a16891f84e7b
☐	29	Admini	admini	827ccb0eea8a706c4c34a16891f84e7b
☐	30	Vlad25	Tamskray777	25f9e794323b453885f5181f1b624d0b

☐ Отметить все
С отмеченными:

Рис.3.14.4 - Результат

Як бачимо, адмін с ім'ям Vladі та нікнеймом Slav – покинув ряди адміністраторів.

3.3 Функціональне тестування програмного забезпечення

Функціональне тестування – тестування що використовується для перевірки працездатності веб-сайту. У ньому тестуються окремі функції всього веб-додатку.

Реалізація модульного тестування.

Кожен модуль був протестований окремо.

Працездатність модулів коректна.

Результати функціонального тестування

Кожна функція тестувалася згідно з функціональними вимогами додатку.

Тестування кожної функції проводилось окремо, так як вони є незалежними один від одного. Функціональне тестування наведене в таблиці 3.1.

Таблиця 3.1

Функція	Результат
Реєстрація	Користувач реєструється успішно, якщо подібного запису не існує в базі даних

Авторизація	Користувач авторизується якщо пароль даного логіну співпадає з введеним
Перегляд кабінету	Користувач може переглянути власний обліковий запис, редагувати чи видалити його
Перегляд товарів	Натиснувши на кнопку, користувач отримує перелік категорія, вибирає потрібну і отримує список товарів.
Фільтр, сортування	Користувач може відфільтрувати чи відсортувати перелік товарів за певними полями
Додавання товару	Якщо користувач проходить перевірку на адміністратора, то стає доступною кнопка додавання товару, при натисканні на яку відкривається відповідна форма.
Видалення товару	Якщо користувач проходить перевірку на адміністратора, то стає доступною кнопка видалення товару, при натисканні на яку видаляється товар
Перегляд додаткової інформації	Користувач може переглянути детальну інформацію про певний товар
Додавання товару в кошик	Користувач може додавати товари до кошика, а потім оформлювати замовлення

Також проведено ряд тестів на зручність використання - засвідчується в простоті й ефективності використання продукту при використанні стандартних практик тестування зручності використання.

Сценарії тестування зручності використання:

Зміст веб-сторінки вірний, без граматичних та орфографічних помилок.

Всі шрифти відповідають вимогам.

Всі тексти правильно вирівняні.

Всі повідомлення про помилки вірні, без орфографічних і граматичних помилок, і відповідають заголовку вікна.

Підказки існують для всіх полів.

Всі поля правильно вирівняні.

Між полями, колонками, рядами і повідомленнями про помилки залишено достатньо вільного місця.

Всі кнопки повинні мати стандартний формат і розмір.

Посилання на домашню сторінку повинна бути на кожній сторінці сайту.

Неактивні поля повинні бути сірими.

Перевірте, що на сайті немає битих посилань і зображень.

Підтверджують повідомлення повинні відображатися для всіх операцій оновлення і видалення.

Перевірте сайт при різних дозволах екрану ((640 x 480, 600x800 і т. Д.)

Перевірте, що користувач може користуватися системою без роздратування.

Перевірте, що TAB правильно працює.

Панель скролла повинна з'являтися тільки тоді, коли вона потрібна.

Якщо при відправці форми є повідомлення про помилку, в ньому повинна міститися інформація, передана користувачем.

Заголовок повинен відображатися на кожній сторінці.

Всі поля (текстові, що випадають меню, радіо-кнопки і т. Д.) І кнопки повинні бути доступні з клавіатури, і користувач повинен бути в змозі користуватися сайтом, використовуючи тільки клавіатуру.

Результатом тестування є висновок, що основні функції інтернет магазину працюють та інтерфейс спроектовано згідно стандартним вимогам GUI.

Висновки до розділу 3

В даному розділі розглянуто архітектуру та працездатність програмного комплексу інтернет магазин одягу. Були наведені приклади та інструкції роботи

з розробленим додатком. В розділі основна увага зосереджена на дослідження розробленого програмного комплексу.

Розглянута робота як користувача так і адміністратора інтернет магазину.

В розділі детально наведено алгоритм встановлення та завантаження інтернет магазину.

ВИСНОВКИ

Під час роботи над кваліфікаційною роботою бакалавра було спроектовано та реалізовано інтернет магазин одягу що автоматизує процес підбору та управління замовленнями товарів. Основна увага в роботі приділялась розробці оптимального для користувачів інтерфейсу для підбору та купівлі товарів. При розробці проекту інтернет магазин одягу були вирішені наступні завдання:

1. Задля чіткого розуміння поставленої задачі було детально описано постановку задачі, проаналізовано популярні реалізації аналогічних сайтів.
2. Вивчаючи проектування CMS систем та технологій для реалізації веб додатків, а саме інтернет магазинів ми наблизились до більш чіткого розуміння того, як саме вирішуються задачі реалізації веб додатків.
3. Для обґрунтування вибору архітектури була проведена дослідницька робота та була обрана найкраща з огляду простоти та якості вирішення задачі автоматизації.
4. Було обрано оптимально зручний варіант для програмної реалізації програмної системи.
5. Було проведено аналіз функціональних вимог, який допоміг в реалізації програмної системи та в розробці зручного інтерфейсу.
6. Описано алгоритм роботи із інтернет магазином для користувача та адміністратора.
8. Протягом усього часу розробки проводилося постійна оптимізація коду, тестування роботи окремих частин.

В результаті проведеної роботи ми вирішили поставлену нами задачу та зуміли розробити систему яка реалізує основний функціонал роботи інтернет магазину.

Реалізований програмний комплекс інтернет магазин одягу має весь необхідний функціонал для підбору та замовленню товарів. В даному випадку проект знаходиться на етапі тестування та занесення реального контенту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Уллман Л. - "PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide" (PHP та MySQL для створення динамічних веб-сайтів: Візуальний посібник для швидкого старту) - Індіанаполіс: Peachpit Press, 2017. - 704 с.
2. Робін Н. - "Створення динамічних веб-сайтів за допомогою PHP, MySQL, JavaScript, CSS та HTML5" - Київ: Видавництво "Інтерпрес ЛТД", 2016. - 768 с.
3. Веллінг, Томсон - "Розробка веб-додатків за допомогою PHP та MySQL" - Київ: Видавництво "БІНОМ", 2017. - 768 с.
4. Грачов А. - "Створити свій сайт на WordPress, робота з CMS WordPress" - Київ: Видавництво "Форекс", 2011. - 282 с.
5. Гаєвський А.Ю. - "100% підручник зі створення веб-сторінок і веб-сайтів: HTML і JavaScript" - Київ: Видавництво "Діалектика", 2008. - 457 с.
6. Гарднер Л., Грігсбі Д. - "Розробка веб-сайтів для мобільних пристроїв" - Київ: Видавництво "Альтерпрес", 2013. - 528 с.
7. Венедюхін Олександр, Воробйов Андрій - "Створення сайтів" (CD-ROM) - Харків: Видавництво "Комп'ютерна академія", 2011. - 448 с.
8. Дакетт Джон - "HTML та CSS. Розробка та дизайн веб-сайтів" (+ CD-ROM) - Харків: Видавництво "Комп'ютерна академія", 2013. - 480 с.
9. CSS Flexbox: A Complete Guide: CSS-Tricks [Електронний ресурс]/ [css-tricks.com](https://css-tricks.com/snippets/css/a-guide-to-flexbox/). – 2022. – Режим доступу до ресурсу: <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
10. "PHP Best Practices: PHP: The Right Way [Електронний ресурс]/ [phptherightway.com](https://phptherightway.com/#best_practices). – 2021. – Режим доступу до ресурсу: https://phptherightway.com/#best_practices
11. HTML Semantic Elements: CSS-Tricks [Електронний ресурс]/ [css-tricks.com](https://css-tricks.com/semantic-html/). – 2022. – Режим доступу до ресурсу: <https://css-tricks.com/semantic-html/>

12. CSS Selectors: CSS-Tricks [Электронный ресурс]/ css-tricks.com. – 2022. – Режим доступа до ресурсу: <https://css-tricks.com/almanac/selectors>
13. SQL Tutorial: W3Schools [Электронный ресурс]/ [w3schools.com](https://www.w3schools.com/sql/). – 2022. – Режим доступа до ресурсу: <https://www.w3schools.com/sql/>
14. Understanding JavaScript Promises: DigitalOcean [Электронный ресурс]/ [digitalocean.com](https://www.digitalocean.com/community/tutorials/understanding-javascript-promises). – 2020. – Режим доступа до ресурсу: <https://www.digitalocean.com/community/tutorials/understanding-javascript-promises>
15. HTML5 Introduction: W3Schools [Электронный ресурс]/ [w3schools.com](https://www.w3schools.com/html/html5_intro.asp). – 2022. – Режим доступа до ресурсу: https://www.w3schools.com/html/html5_intro.asp
16. Introduction to JavaScript: Mozilla Developer Network (MDN) [Электронный ресурс]/ [developer.mozilla.org](https://developer.mozilla.org/enUS/docs/Web/JavaScript/Guide/Introduction_to_JavaScript) – 2021. – Режим доступа до ресурсу: https://developer.mozilla.org/enUS/docs/Web/JavaScript/Guide/Introduction_to_JavaScript

ДОДАТКИ

					ІІЗ.КР.Б – 121 – 23 – ІІЗ	
						56

index.php (адміністратор)

```

<?php include('partials/menu.php') ?>

<!-- Main content Section Starts-->
<div class="main-content">
  <div class="wrapper">
    <h1>Інформаційна панель</h1>

    <br><br>

    <?php
    if(isset($_SESSION['login']))
    {
      echo $_SESSION['login'];
      unset($_SESSION['login']);
    }
    ?>

    <br><br>

    <div class="col-4 text-center">

      <?php

      $sql = "SELECT * FROM tbl_category";
      //виконати запит
      $res = mysqli_query($conn, $sql);
      //підрахувати рядки
      $count = mysqli_num_rows($res);

      ?>

      <h1><?php echo $count; ?></h1>
      <br>
      Категорії
    </div>
    <div class="col-4 text-center">

      <?php

      $sql2 = "SELECT * FROM tbl_stuff";
      //виконати запит
      $res2 = mysqli_query($conn, $sql2);
      //підрахувати рядки
      $count2 = mysqli_num_rows($res2);

      ?>

      <h1><?php echo $count2; ?></h1>
      <br>
      Товари
    </div>
    <div class="col-4 text-center">

      <?php

      $sql3 = "SELECT * FROM tbl_order";
      //виконати запит
      $res3 = mysqli_query($conn, $sql3);
      //підрахувати рядки
      $count3 = mysqli_num_rows($res3);

```

```

        ?>

        <h1>php echo $count3; ?&gt;&lt;/h1&gt;
        &lt;br&gt;
        Усього замовлень
    &lt;/div&gt;
    &lt;div class="col-4 text-center"&gt;

        &lt;?php

        //створити SQL-запит для отримання загального доходу
        //сукупна функція в SQL
        $sql4 = "SELECT SUM(Total) AS total FROM tbl_order WHERE Status =
'Delivered'";

        //виконати запит
        $res4 = mysqli_query($conn, $sql4);

        //отримати значення
        $row4 = mysqli_fetch_assoc($res4);

        //отримати загальний дохід
        $totalRevenue = $row4['total'];

        ?&gt;

        &lt;h1&gt;$&lt;?php echo $totalRevenue; ?&gt;&lt;/h1&gt;
        &lt;br&gt;
        Отриманий дохід
    &lt;/div&gt;

    &lt;div class="clearfix"&gt;&lt;/div&gt;

&lt;/div&gt;
&lt;/div&gt;
&lt;!-- Main content Ends--&gt;

&lt;?php include('partials/footer.php') ?&gt;
</pre

```

constants.php

```

<?php

//розпочати сеанс
ob_start();
session_start();

//Створюйте константи для зберігання значень
define('SITEURL', 'http://store');
define('LOCALHOST', 'localhost');
define('DB_USERNAME', 'root');
define('DB_PASSWORD', 'root');
define('DB_NAME', 'webstore');

$conn = mysqli_connect(LOCALHOST, DB_USERNAME, DB_PASSWORD) or
die(mysqli_error()); //підключення до бази даних
$dbSelect = mysqli_select_db($conn, DB_NAME) or die(mysqli_error()); //база
даних відбору

?>

```

add-stuff.php

```
<?php include('partials/menu.php'); ?>

<div class="main-content">
  <div class="wrapper">
    <h1>Додати товар</h1>

    <br><br>

    <?php
      //if(isset($_SESSION['add']))
      //{
        //  echo $_SESSION['add'];
        //  unset($_SESSION['add']);
      //}

      if(isset($_SESSION['upload']))
      {
        echo $_SESSION['upload'];
        unset($_SESSION['upload']);
      }

    ?>

    <form action="" method="POST" enctype="multipart/form-data">
      <table class="tbl-30">
        <tr>
          <td>Назва: </td>
          <td>
            <input type="text" name="title" placeholder="Введіть
назву">
          </td>
        </tr>
        <tr>
          <td>Опис: </td>
          <td>
            <textarea name="description" cols="30" rows="5"
placeholder="Введіть опис товару"></textarea>
          </td>
        </tr>
        <tr>
          <td>Ціна: </td>
          <td>
            <input type="number" name="price">
          </td>
        </tr>
        <tr>
          <td>Оберіть зображення:</td>
          <td>
            <input type="file" name="image">
          </td>
        </tr>
        <tr>
          <td>Категорія: </td>
          <td>
            <select name="category">

            <?php
              //створення php-коду для відображення категорій з БД
              //1. Створення sql для отримання активних категорій
з БД
              $sql = "SELECT * FROM tbl_category WHERE
```

```

Active='Yes'";

//виконання запиту
$res = mysqli_query($conn, $sql);

//підрахунок рядків, щоб перевірити, чи є у нас
категорії чи ні
$count = mysqli_num_rows($res);

//if count > 0, у нас є категорії, else - у нас
немає їх
if($count>0)
{
    //у нас є категорії
    while($row=mysqli_fetch_assoc($res))
    {
        //отримати деталі категорій
        $id = $row['ID'];
        $title = $row['Title'];
        ?>

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

        <?php
    }
}
else
{
    //у нас немає категорій
    ?>

    <option value="0">Не знайдено категорій</option>

    <?php
}

//2. відобразити на розкритому меню
?>

</select>
</td>
</tr>
<tr>
<td>Featured: </td>
<td>
        <input type="radio" name="featured" value="Yes"> Так
        <input type="radio" name="featured" value="No"> Ні
    </td>
</tr>
<tr>
<td>Active: </td>
<td>
        <input type="radio" name="active" value="Yes"> Так
        <input type="radio" name="active" value="No"> Ні
    </td>
</tr>
<tr>
<td colspan="2">
        <input type="submit" name="submit" value="Додати товар"
class="btn-secondary">
    </td>
</tr>

```

```

        </table>
    </form>

    <?php

        //перевірка, натиснуто кнопку чи ні
        if(isset($_POST['submit']))
        {
            //echo "Clicked";
            //1. Отримання даних з форми
            $title = $_POST['title'];
            $description = $_POST['description'];
            $price = $_POST['price'];
            $category = $_POST['category'];

            //перевірка, чи встановлені перемикачі для ознак та активності
            if(isset($_POST['featured']))
            {
                //отримання значення з форми
                $featured = $_POST['featured'];
            }
            else
            {
                //встановити значення за замовчуванням
                $featured = "No";
            }

            if(isset($_POST['active']))
            {
                $active = $_POST['active'];
            }
            else
            {
                $active = "No";
            }

            //2. Завантаження зображення, якщо вибрано
            //перевірка, чи клацнута кнопка на вибраному зображенні чи ні, і
            //завантаження вибраного зображення
            if(isset($_FILES['image']['name']))
            {
                //отримання деталей обраного зображення
                $imageName = $_FILES['image']['name'];

                //перевірка, чи вибрано зображення чи ні, і завантаження
                //зображення, якщо вибрано
                if($imageName!="")
                {
                    //вибрано зображення
                    //Перейменування зображення
                    //отримати розширення вибраного зображення (jpg, png, gif,
                    //etc) e.g. specialstuff1.jpg
                    $ext = explode('.', $imageName);
                    $dot = ".";
                    $array = explode($dot, $imageName);
                    $ext = end($array);

                    //перейменувати зображення
                    $imageName = "Stuff_Name_".rand(0000, 9999).".$ext; //
                    //e.g. Stuff_Name_1233.jpg

                    //отримати вихідний шлях і шлях призначення
                    //вихідний шлях - це поточне розташування зображення

```

```

        $src = $_FILES['image']['tmp_name'];

        //шлях призначення для завантажуваного зображення
        $dst = "../images/stuff/".$imageName;

        //завантаження зображення
        $upload = move_uploaded_file($src, $dst);

        //перевірка, чи завантажено зображення чи ні
        //а якщо зображення не буде завантажено, ми зупинимо процес
і переадресуємо з повідомленням про помилку
        if ($upload == false) {
            //встановити повідомлення
            $_SESSION['upload'] = "<div class='error'>Помилка
завантаження зображення</div>";
            //Переспрямування на керування сторінкою категорії
            header("location:" . SITEURL . '/admin/add-stuff.php');
            //зупинити процес
            die();
        }
    }
}
else
{
    $imageName = ""; //встановити значення за замовчуванням як
порожнє
}

//3. Вставити в БД
//створити запит sql для збереження або додавання товарів
//для числового значення нам не потрібно передавати значення в лапки
'',

// але для рядкового значення обов'язково додавати лапки ''
$sql2 = "INSERT INTO tbl_stuff SET
        Title = '$title',
        Description = '$description',
        Price = $price,
        ImageName = '$imageName',
        CategoryID = '$category',
        Featured = '$featured',
        Active = '$active'
";

//виконати запит
$res2 = mysqli_query($conn, $sql2);

//перевірити, чи вставлені дані чи ні

if($res2==true)
{
    //дані успішно вставлені
    $_SESSION['add'] = "<div class='success'>Товар доданий
успішно</div>";
    header("location:" . SITEURL . '/admin/manage-stuff.php');
}
else
{
    //не вдається вставити дані
    $_SESSION['add'] = "<div class='error'>Помилка додавання
товару</div>";
    header("location:" . SITEURL . '/admin/manage-stuff.php');
}
}

```

```
    }  
    ?>  
  
    </div>  
</div>  
  
<?php include('partials/footer.php'); ?>
```