

ЗВО УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА

Факультет суспільних та прикладних наук

Кафедра інформаційних технологій

на правах рукопису

Бурлака Владислав Сергійович

УДК 004.378

Розробка веб-застосунку з продажу виробів ручної роботи засобами Java

Спеціальність 121 – «Інженерія програмного забезпечення»

Кваліфікаційна робота на здобуття кваліфікації бакалавра

Нормоконтроль

_____ Стисло О.В.
(підпис, дата, розшифрування підпису)

Студент

_____ Бурлака В.С.
(підпис, дата, розшифрування підпису)

Допускається до захисту
Завідувач кафедри

_____ к.т.н., доц. Пашкевич О.П.
(підпис, дата, розшифрування підпису)

Керівник роботи

_____ к.т.н., доц. Ващишак С.П.
(підпис, дата, розшифрування підпису)

Івано-Франківськ – 2023

ЗВО УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА
Факультет суспільних та прикладних наук
Кафедра інформаційних технологій

Освітній ступінь: «бакалавр»

Спеціальність: 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

« ____ » _____ 2023 року

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Бурлака Владислав Сергійович

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи

Розробка веб-застосунку з продажу виробів ручної роботи засобами Java
керівник роботи:

Ващишак Сергій Петрович, кандидат технічних наук

затверджена наказом вищого навчального закладу від «11» листопада 2022 року

№ 155/ІНВ

2. Термін подання студентом роботи 14.06.2023

3. Вихідні дані роботи: Мова програмування Java.

4. Зміст кваліфікаційної роботи (перелік питань, які потрібно розробити)

1. Аналіз проблеми.

2. Обґрунтування вибору платформи.

3. Реалізація сайту.

5. Дата видачі завдання 10.10.2022

КОНСУЛЬТАНТИ РОЗДІЛІВ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Розділ	Консультант (прізвище, ініціали та посада)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1.	Проектування та розробка сайту	15.02.2023	Виконано
2.	Програмна реалізація сайту	14.03.2023	Виконано
3.	Розробка структури веб-сайту	07.04.2023	Виконано
4.	Економічне обґрунтування розробки	28.04.2023	Виконано
5.	Формування висновків	05.05.2023	Виконано
6.	Оформлення пояснювальної записки	19.05.2023	Виконано
7.	Оформлення графічного матеріалу та підготовка до захисту роботи	03.06.2023	Виконано

Студент

(підпис)

Бурлака В.С.

(прізвище та ініціали)

Керівник роботи

(підпис)

Ващишак С.П.

(прізвище та ініціали)

Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Сторінка	Опис графічного матеріалу
14	Вітрина магазину «Творчий позитив»
15	Вітрина магазину «ZRK»
17	Головна сторінка сайту «расорку»
18	Реклама товару на сайті «расорку»
24	Архітектура Spring Boot
25	H2 Database
27	Структура Spring Data JPA
28	Принцип роботи Maven
30	Структура Thymeleaf
31	Структура Use Case Diagram
33	Use Case Diagram для мого сайту
34	Макет сайту
36	Меню реєстрації
38	Головна сторінка сайту
44	Корзина

АНОТАЦІЯ

У дипломній роботі розглянуто розробка веб-застосунку з продажу виробів ручної роботи засобами мови java . Сайт дозволяє знайти і придбати унікальні вироби ручної роботи . Розроблений сайт дозволяє користувачеві обрати річ, яка йому сподобалась і замовити її. В дипломній роботі розглянуто процес розробки сайту, включаючи вибір технологій та інструментів, потім був створений макет самого сайту, проектування інтерфейсу користувача, розробку та тестування функціоналу сайту.

SUMMARY

The thesis deals with the development of a web application for selling handmade products using java language. The site allows you to find and purchase unique handmade products. The developed site allows the user to choose the item he or she likes and order it. The thesis describes the process of website development, including the choice of technologies and tools, followed by the layout of the site itself, design of the user interface, development and testing of the site's functionality.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1 АНАЛІЗ ПРОБЛЕМИ.....	10
1.1 Виявлення потреб до інформації щодо виробів та послуг.....	10
1.2 Аналіз існуючих сайтів з виробами ручної роботи.....	13
1.3 Постановка задачі.....	18
Висновки до розділу 1	18
РОЗДІЛ 2. ОБҐРУНТУВАННЯ ВИБОРУ ПЛАТФОРМИ.....	20
2.1 Мова програмування Java.....	20
2.2 Обґрунтування вибору платформи.....	21
2.3 Використання Framework.....	22
2.4 H2: Легка та вбудована база даних.....	24
2.5 Spring Data JPA: Зручність та продуктивність роботи з базами данних.....	26
2.6 Maven. Ефективне управління залежностями та збиранням проєктів...27	27
2.7 Thymeleaf.....	29
2.8 Use Case Diagram.....	30
Висновок до розділу 2.....	32
РОЗДІЛ 3. РЕАЛІЗАЦІЯ САЙТУ.....	33
3.1 Написання Use Case Diagram.....	33
3.2 Створення макета сайту.....	34
3.3 Реалізація основного функціоналу процесу реєстрації.....	36
3.4 Процес створення головної сторінки сайту.....	37
3.5 Процес створення кошика.....	42
Висновок до розділу 3.....	43
ВИСНОВКИ.....	45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	46

ВСТУП

Актуальність теми. тема виробів ручної роботи залишається актуальною незалежно від часу, оскільки вона відображає прагнення до унікальності, індивідуальності та вишуканості в сучасному світі. Навіть у епоху технологій та масового виробництва, вироби ручної роботи мають свою особливу цінність та популярність. Ось декілька аргументів, які підтверджують актуальність цієї теми:

1. Унікальність і ексклюзивність: Вироби ручної роботи є унікальними, оскільки кожен з них створений вручну майстром. Вони відрізняються від масово вироблених товарів, мають свою особливу атмосферу і виражають індивідуальність свого творця.

2. Творчість та вишуканість: Вироби ручної роботи є результатом мистецтва та ручної праці. Вони втілюють в собі талант, креативність та навички майстра, що робить їх особливими і привабливими для людей, які цінують красу та естетику.

3. Підтримка місцевої економіки та майстрів: Придбання виробів ручної роботи сприяє розвитку малого бізнесу та підтримці майстрів. Вони часто працюють власноруч, створюючи свої унікальні вироби, і покупка їхніх творінь допомагає їм у розвитку та забезпечує достойний дохід.

4. Екологічність та сталий розвиток: Вироби ручної роботи можуть бути виготовлені з природних, екологічно чистих матеріалів. Це сприяє збереженню природних ресурсів та зменшенню впливу на довкілля.

5. Підтримка традицій та культури: Багато виробів ручної роботи мають глибокі корені в традиціях та культурі різних народів. Вони втілюють у собі народну спадщину, розповідаючи історію і зберігаючи цінні традиції.

Отже, тема виробів ручної роботи є актуальною та привабливою для багатьох людей, оскільки вона поєднує у собі творчість, унікальність, естетику та підтримку майстрів та культурної спадщини.

Мета роботи: створення зручного сайту для купівлі чудових виробів ручної роботи з використанням засобів Java.

Головні цілі створення сайту включають:

1. Розробку зрозумілого дизайну сайту для користувачів.
2. Продаж та просування виробів: головною ціллю має бути створення платформи для продажу виробів ручної роботи. Сайт повинен представляти різні категорії виробів, дозволяти майстрам реєструватися та розміщувати свої продукти для продажу. Основним завданням у продажі та просуванні виробів є залучення клієнтів та забезпеченні успішних транзакцій.

3. Промоція майстрів та їхніх послуг: сайт може виконувати роль візитної картки для майстрів ручної роботи. Він може представляти їхні портфоліо, детальні описи їхніх навичок та послуг. Особливістю такого сайту буде залучення нових клієнтів та організація замовлень для майстрів.

4. Спільнота та обмін досвідом: сайт може бути платформою для обміну досвідом між майстрами ручної роботи. Він може включати форум, блоги, відео-туторіали та інші інструменти для обговорення та навчання. Метою такого сайту буде сприяння взаємодії та зростанню учасників спільноти.

Об'єкт роботи. Процес створення сайтів за допомогою технологій Java.

Предмет роботи. Інформаційна система представлення, продажу та просування виробів, створених майстрами ручної роботи.

Завдання роботи. Відповідно до обраної теми сайт має виконувати такі задачі:

1. Представлення виробів: сайт має демонструвати широкий асортимент виробів ручної роботи, забезпечуючи їхню наочність та доступність для покупців. Важливо надати детальні описи, фотографії та ціни для кожного виробу.

2. Забезпечення зручного пошуку: сайт повинен мати функціонал, який дозволяє покупцям швидко та зручно знаходити потрібні вироби. Це може включати фільтри за категоріями, цінами, матеріалами, розмірами та іншими параметрами.

3. Забезпечення безпечної та зручної покупки: сайт має мати функціонал онлайн-корзини та безпечної оплати, щоб покупці могли зручно здійснювати покупки виробів. Також важливо надати інформацію про доставку та повернення товарів.

4. Промоція майстрів: сайт має бути платформою для просування та популяризації майстрів ручної роботи.

Методи дослідження. Для реалізації проекту та виконання поставлених завдань, було використано мову програмування Java та Spring фреймворк.

Практичне значення одержаних результатів. В результаті виконання кваліфікаційної бакалаврської роботи був розроблений сайт для демонстрації виробів ручної роботи з метою їх популяризації та організації продажу.

РОЗДІЛ 1. АНАЛІЗ ПРОБЛЕМИ

1.1 Виявлення потреб до інформації щодо виробів та послуг

В сучасному світі, насиченому технологіями та глобальним зв'язком, сайти стають невід'ємною частиною нашого повсякденного життя. Вони задовольняють нашу потребу у зручному та швидкому доступі до інформації, послуг та товарів.

Сайти - це віртуальні портали, які відкривають перед нами безмежний світ можливостей. Вони створюють зручний і доступний спосіб комунікації, інтерактивного взаємодії та отримання необхідної інформації. Давайте розглянемо деякі з головних можливостей, які надають нам сайти.

Джерело інформації: Сайти є безцінним джерелом інформації про різноманітні теми. Ми можемо швидко знайти відповіді на свої питання, читати новини, вивчати нові теми, отримувати освітні матеріали і навіть знаходити розвагу. Завдяки сайтам, інформація стала доступною всім, незалежно від місця проживання чи соціального статусу.

Комерційні можливості: Сайти відкривають широкі можливості для підприємців та бізнесів. Вони дозволяють продавати товари та послуги в Інтернеті, досягати нових клієнтів, розширювати географію свого бізнесу та спрощувати процес покупок для споживачів.

Комунікація та взаємодія: Сайти надають можливість взаємодіяти та спілкуватися з іншими людьми, незалежно від відстані. Ми можемо обмінюватися повідомленнями, електронною поштою, участь у форумах та соціальних мережах, вести відеодзвінки, організовувати вебінари та інші онлайн-заходи. Сайти створюють віртуальні спільноти, де можна знайти однодумців та знайомих.

У сучасному світі, який все більше орієнтується на цифрові технології, розвиток сайтів відіграє важливу роль у сприянні комунікації, доступу до інформації та розширенні можливостей людей. Сайти перетворилися на віртуальні вікна, що відкривають перед нами безмежний світ можливостей. З початку Інтернету, коли перші сайти були простими та статичними, розвиток веб-технологій приніс безліч нововведень. Сьогодні ми маємо динамічні, інтерактивні та мобільно-адаптивні сайти, які забезпечують зручний доступ до інформації на будь-якому пристрої. Сайти стали основою комунікації, незалежно від фізичних відстаней. Завдяки ним, ми можемо спілкуватися з рідними та друзями, обмінюватися новинами та ідеями з усього світу. Соціальні мережі, форуми та чати стали майданчиками для обговорень, співпраці та культурного обміну.

Сайти надають нам необмежений доступ до інформації. Вони стали віртуальними бібліотеками, де можна знайти відповіді на будь-які питання, вивчати нові теми, отримувати освітні матеріали та розвивати свої навички.

Онлайн-курси та навчальні платформи допомагають нам отримати знання в будь-який зручний для нас час та місце. Завдяки сайтам, ми маємо широкі можливості для розвитку особистості та кар'єри. Блоги та власні веб-сайти дозволяють нам поділитися своїми думками, ідеями та творчістю зі світом. Ми можемо просувати свої бізнеси, знаходити нові можливості співпраці та встановлювати контакти з клієнтами та партнерами.

Сайти стали платформою для культурного обміну та розвитку мистецтва. Музеї, галереї та театри можуть демонструвати свої колекції та виставки в онлайн-режимі, привертаючи нових шанувальників та заохочуючи взаємодію з аудиторією. Не можна забувати і про електронну торгівлю, яка стала невід'ємною частиною нашого життя.

Сайти дозволяють нам здійснювати покупки та продажі в Інтернеті, знайти потрібний товар або послугу, порівняти ціни та отримати його прямо до дверей. Розвиток сайтів відкриває перед нами безліч можливостей у всіх

сферах життя. Вони стали нашими віртуальними воротами до світу, де ми можемо спілкуватися, навчатися, творити, купувати та розвиватися. І цей розвиток триває, надаючи нам ще більше зручності, інновацій та можливостей. Сайти надають нам можливість:

1. Отримувати актуальну інформацію з різних галузей та джерел. Здійснювати покупки онлайн, зручно порівнюючи ціни та характеристики товарів.
2. Знаходити контакти та спілкуватися з компаніями, установами та іншими людьми.
3. Дізнаватися про події, новини та тренди, що відбуваються у світі. Відкривати нові можливості для розвитку особистості та бізнесу.

Сайти стають центральними точками нашого онлайн-існування, допомагаючи нам знайти потрібну інформацію, здійснити покупки та спілкуватися зі світом. Вони відображають сучасні вимоги до зручності та доступності, що робить їх невід'ємною складовою нашого способу життя.

Потреба у сайті з реклами виробів ручної роботи виникає з кількох важливих причин: платформу для презентації, глобальну доступність, зручність для покупців, забезпечення довіри та відгуків, маркетингу та просування товару. Розглянемо кожен з них.

Платформа для презентації: Майстри ручної роботи потребують майданчика, де вони можуть представити свої вироби та поділитися своїм талантом зі світом. Сайт надає їм можливість візуально продемонструвати свої унікальні вироби та привернути увагу потенційних клієнтів.

Глобальна доступність: Сайт дозволяє майстрам ручної роботи розширити свою аудиторію та залучити клієнтів не тільки зі свого місця проживання, але й з усього світу. Це дає їм можливість залучити більше замовлень та розширити свій бізнес

Зручність для покупців: Клієнти, які зацікавлені в унікальних виробах ручної роботи, можуть знайти все, що їм потрібно, на одному сайті. Вони

можуть шукати вироби, переглядати фотографії, отримувати детальну інформацію про продукти та здійснювати покупки зручним способом.

Забезпечення довіри та відгуків: Сайт може мати відділи для відгуків та рейтингів, де клієнти можуть залишати свої враження про придбані вироби та роботу майстрів. Це дозволяє збудувати довіру серед покупців та підтримує репутацію майстрів.

Маркетинг та просування: Сайт з виробів ручної роботи є важливим інструментом маркетингу та просування для майстрів.

1.2 Аналіз існуючих сайтів з виробами ручної роботи

Занепад виробів ручної роботи є непокоючою тенденцією, яка може бути спричинена декількома суттєвими факторами:

1. Конкуренція від масового виробництва: З появою промислового виробництва та автоматизованих технологій, вироби ручної роботи поступово витісняються більш швидкими, ефективними та дешевшими масовими виробами. Це зменшує попит на ручні вироби.

2. Зміна споживацьких пріоритетів: Сучасне суспільство все більше прагне до зручності, швидкості та високих технологій. Внаслідок цього, споживачі можуть віддалятися від виробів ручної роботи, які вимагають більше часу та зусиль для створення.

3. Зміна культурних та соціальних цінностей: В сучасному світі все більше цінується масова споживча культура та швидкість задоволення потреб. Це може призводити до зменшення інтересу до унікальних, індивідуальних виробів ручної роботи.

4. Недостатнє визнання та підтримка: Майстри, що займаються виробництвом ручних виробів, можуть зіткнутися з викликами у популяризації своєї роботи та отриманні достатньої підтримки, що може вплинути на їх мотивацію та продуктивність.

Незважаючи на ці фактори, важливо підтримувати і визнавати цінність виробів ручної роботи. Вони мають унікальність, розкривають творчий потенціал майстрів та відображають культурну спадщину. Популяризація та підтримка, яких відбувається дотепер.

Перед розробкою сайту я переглянув декілька сайтів-аналогів та оцінив їхні різні характеристики.

Першим я розглянув сайт Магазину “Творчий позитив”, головна сторінка якого наведена на (рис. 1.1).

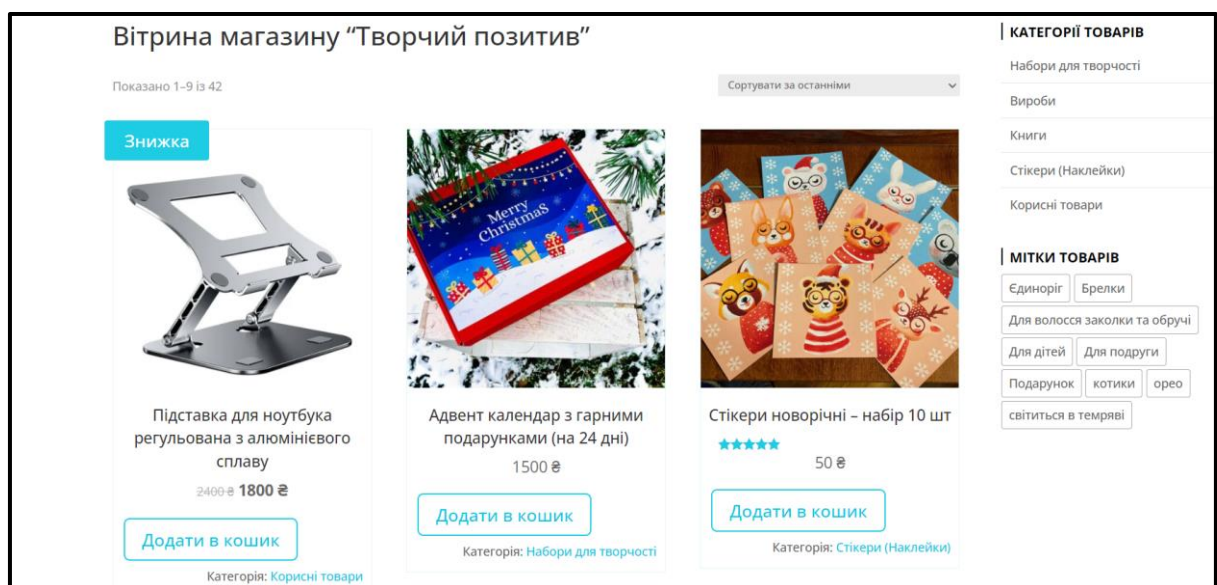


Рисунок 1.1 - Вітрина магазину “Творчий позитив”

Наразі сайт ще працює, тільки є одна проблема, що полягає в складності його використання.

Незручний пошук товарів. Якщо магазин має велику кількість товарів, але не забезпечує зручну систему пошуку, це може ускладнити користувачам знаходження потрібного їм товару. Це може призвести до погіршення користувацького досвіду.

Також це магазин створений однією майстринею і на ньому немає функції додати свій товар і так використовувати його у своїх цілях для

продажу товарів. Тому було вирішено спростити цю систему за допомогою нашого сайту.

Наступним проаналізовано сайт магазину "zrk", головна сторінка якого наведена на (рис. 1.2).

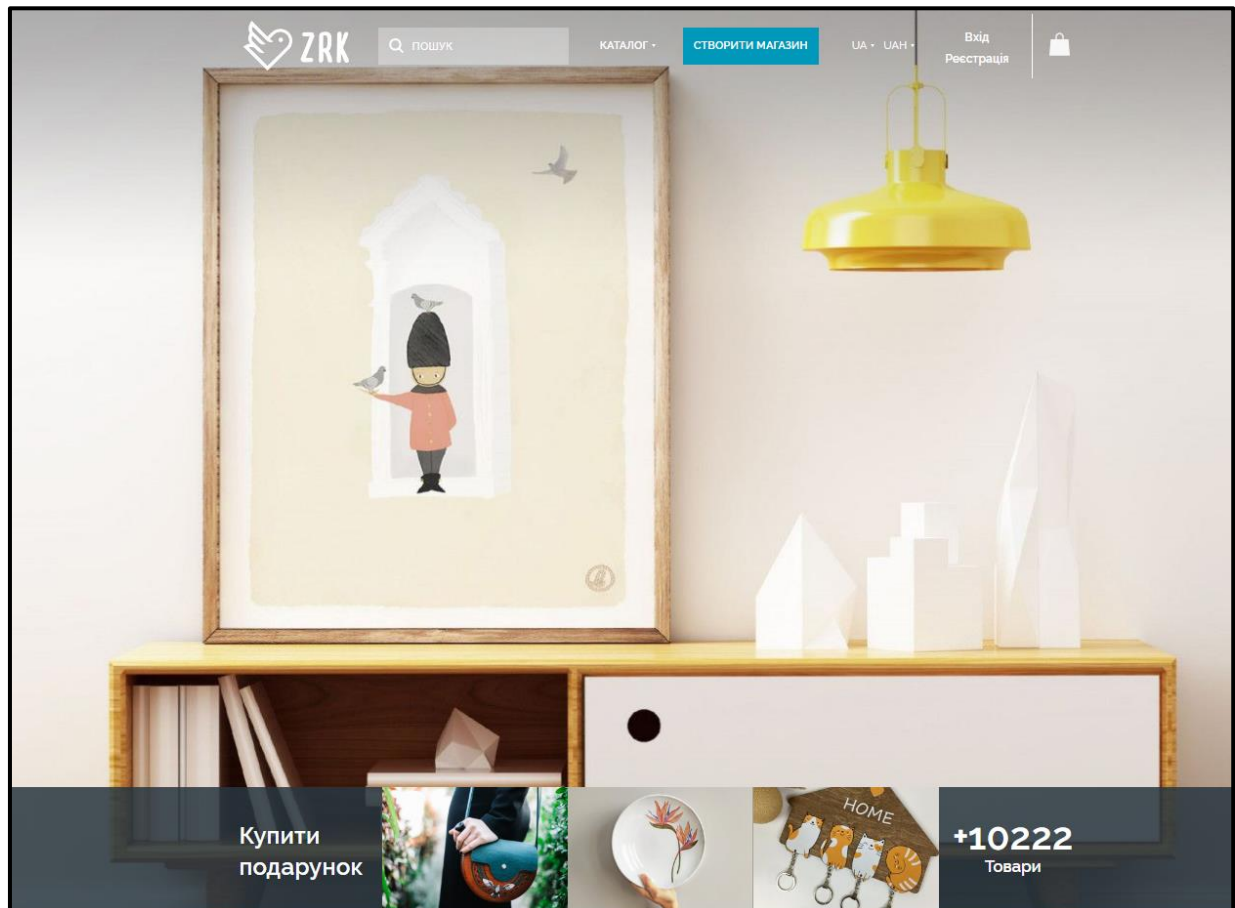


Рисунок 1.2 - Вітрина магазину "Zrk"

Золоті руки- це унікальний портал майстрів на всі руки, який поєднав у собі магазин авторських робіт, простір для спілкування та обміну досвідом усіх творчих людей, а також став платформою для відкриття та розвитку власного бізнесу.

Золоті руки - спільнота шанувальників мистецтва, майстрів, художників та дизайнерів, які обожнюють свою справу, сміливо

реалізують нові задуми та ідеї та отримують справжню візуальну та тактильну насолоду від створених виробів, особливістю яких є ручна робота.

На (рисунку 1.2) зображена вітрина магазину “Zrk” (Золоті руки). На головній сторінці ми одразу можемо побачити функцію створити магазин, що дає змогу новому майстру створити власний малий магазин для продажу своїх товарів.

Перевагою такого сайту є досить привабливий дизайн та зручність для користувачів, але проблема полягає в тому що це є платна платформа і для починаючих майстрів є трохи проблемно відразу переводити кошти незнаючи чи потрібний їх товар.

Наступний сайт – аналог - маркетплейс рукоділля (расorky.com), головна сторінка якого наведена на (рис. 1.3).

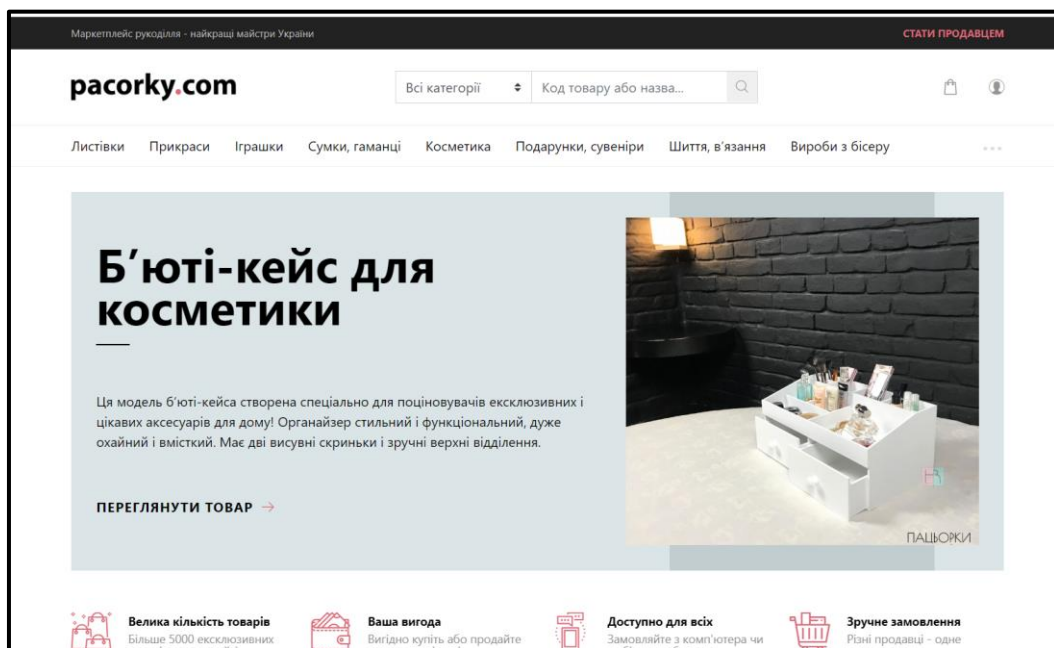


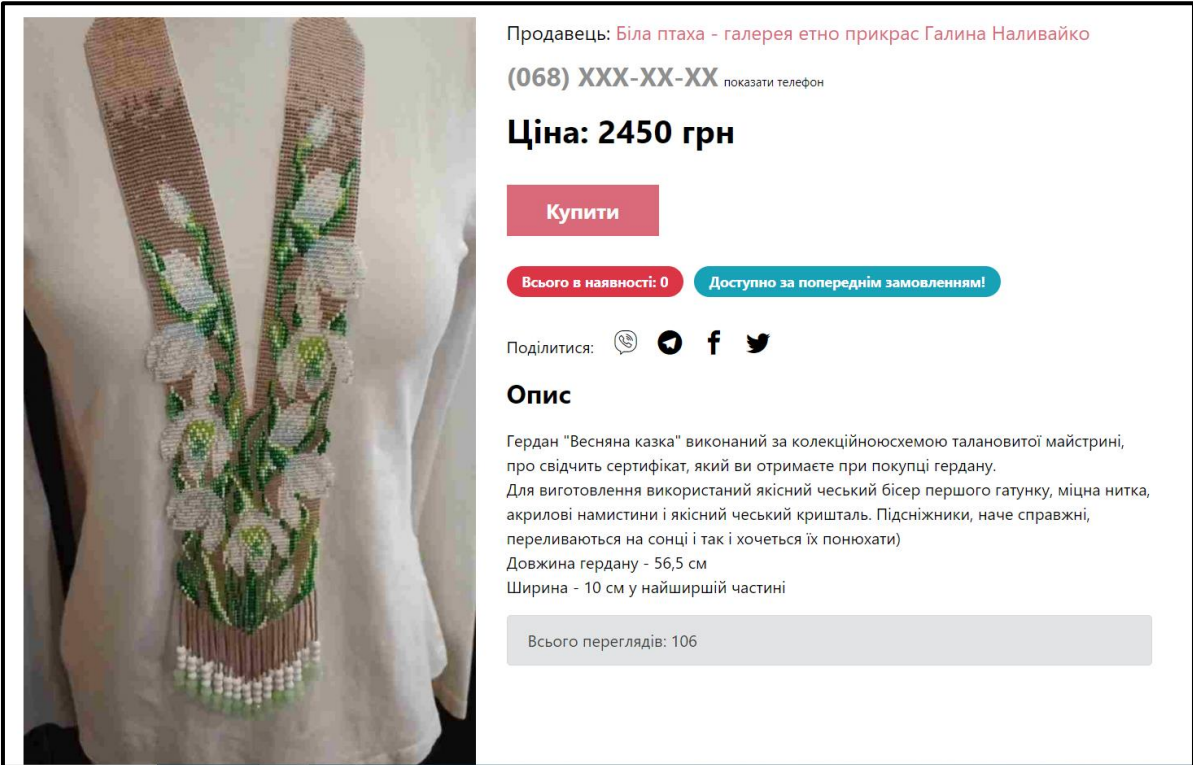
Рисунок 1.3 - Головна сторінка сайту расorky.com

Сайт расorky.com має за мету розвивати та популяризувати хенд-мейд як мистецтво створювати речі, що дарують позитивну енергетику та настрій як тим, хто їх створює, так і тим, хто їх та використовує собі на радість.

Отже хенд-мейд портал "Пацьорки" є онлайн-майданчиком, що допомагає майстрам, котрі власноруч виготовляють сувенірну та подарункову продукцію, успішно її продавати. Відповідно покупці такої продукції можуть переглянути пропозиції товарів з фото від усіх майстрів і купити ті товари, які їм до вподоби.

Хенд-мейд в Україні активно розвивається та набирає все більшої популярності тому, окрім обслуговування майданчику купівлі-продажу, ми намагаємося публікувати цікаву інформацію, котра стосується хенд-мейду - різноманітні новини, майстер-класи, анонси.

Сайт має досить широкий функціонал також працює і в соціальних мережах таких як: інстаграм і фейсбук (рис. 1.4).




Продавець: Біла птаха - галерея етно прикрас Галина Наливайко
(068) XXX-XX-XX показати телефон

Ціна: 2450 грн

Купити

Всього в наявності: 0 **Доступно за попереднім замовленням!**

Поділитися: 

Опис

Гердан "Весняна казка" виконаний за колекційною схемою талановитої майстрині, про свідчить сертифікат, який ви отримаєте при покупці гердану. Для виготовлення використаний якісний чеський бісер першого гатунку, міцна нитка, акрилові намистини і якісний чеський кристаль. Підсніжки, наче справжні, переливаються на сонці і так і хочеться їх понохати)
Довжина гердану - 56,5 см
Ширина - 10 см у найширшій частині

Всього переглядів: 106

Рисунок 1.4 - Реклама товару на сайті pacorky.com

Але основним недоліком сайту є те що більшості товарів на жаль немає у наявності, тільки під замовлення, а також багато товарів взагалі не роблять.

1.3 Постановка задачі

Основна мета кваліфікаційної роботи полягає в написанні веб-сайту для презентації та продажу виробів ручної роботи. Веб-сайт повинен мати такі функціональні можливості:

1. Головна сторінка: Створення привабливої головної сторінки, яка представлятиме асортимент виробів ручної роботи. На головній сторінці можуть бути відображені вибрані чи нові вироби, зразки робіт, акції або спеціальні пропозиції.

2. Каталог товарів: Створення категорій та підкатегорій для організації товарів ручної роботи. Кожен товар повинен мати свою сторінку з детальним описом, фотографіями, ціною та можливістю додати його в кошик для покупки в майбутньому.

3. Кошик покупок: Реалізація системи кошика, де користувач може додавати вибрані товари, переглядати їх кількість та загальну суму замовлення. Також повинна бути можливість редагувати замовлення та видаляти товари з кошика.

4. Профіль користувача: Реєстрація та авторизація користувачів на сайті. Користувачі повинні мати свої особисті профілі, де вони можуть переглядати свої замовлення, змінювати особисту інформацію та адресу доставки чи відслідковувати інші дії на сайті.

Висновок до розділу 1

У даному розділі було розглянуто причини занепаду виробів ручної роботи, що становить серйозну проблему, яка може мати наслідки як для самого ремісництва, так і для культурної та економічної спадщини. Під час виявлення потреб удосконалення цього було вирішено, що створення веб-

сайту може допомогти у запобіганню занепаду ремісництва і розвитку майстерності звичайних людей.

Дослідження існуючих сайтів показало що насправді їх дуже мало і більшість користуються платними платформами для онлайн-оголошень. Це свідчить про потребу більшості таких сайтів, які відповідають сучасним потребам людства.

РОЗДІЛ 2. ОБҐРУНТУВАННЯ ВИБОРУ ПЛАТФОРМИ

Від правильного вибору технології програмування, мови, алгоритму, структури та дизайну сайту залежить успіх будь-якого проекту в галузі програмування та розробки.

2.1 Мова програмування Java

Мова програмування Java стала справжньою революцією у світі програмування. Розроблена компанією Sun Microsystems (пізніше придбаною Oracle Corporation), Java завоювала серця розробників та стала однією з найпопулярніших мов програмування у світі. Її універсальність та потужність дозволяють створювати різноманітні програми та додатки, що працюють на різних платформах.

Однією з найважливіших рис Java є її кросплатформеність. Це означає, що програми, написані на Java, можуть працювати на будь-якій платформі, яка підтримує віртуальну машину Java (JVM). Це робить Java ідеальним вибором для розробки крос-платформових додатків, які можуть запускатися на різних операційних системах, таких як Windows, macOS, Linux тощо.

Крім того, Java дозволяє розробникам писати одну версію програми, яка працюватиме на різних пристроях, від комп'ютерів до мобільних пристроїв. Java володіє потужним набором бібліотек і фреймворків, які допомагають розробникам ефективно створювати складні програми. Наприклад, бібліотека Java Standard Library містить безліч класів та методів, які спрощують роботу з файлами, мережами, базами даних, графікою та багатьма іншими аспектами програмування. Крім того, Java має потужні фреймворки, такі як Spring, Hibernate, JavaFX, які допомагають розробникам швидко розробляти високоякісні та ефективні програми. Java також відома

своєю надійністю та безпекою. Вона має вбудовану систему управління пам'яттю, що дозволяє автоматично вирішувати проблеми зі звільненням пам'яті, уникаючи такі помилки, як некоректна робота з пам'яттю.

Крім того, Java має вбудовану систему безпеки, що дозволяє запускати програми у безпечному середовищі та контролювати доступ до ресурсів комп'ютера. Java також є популярною мовою для розробки веб-додатків, особливо на платформі Java Enterprise Edition (Java EE). Вона надає засоби для створення потужних серверних додатків, що працюють у розподіленому середовищі та забезпечують широкі можливості для масштабування та розширення. Розвиток Java постійно триває. Вона оновлюється та розширюється з'явленням нових версій, які принесли багато нововведень, включаючи покращення продуктивності, введення нових функціональних можливостей та підтримку сучасних технологій.

Мова програмування Java стала справжнім символом сучасного софтверного світу. Вона пропонує потужність, універсальність та безпеку, що робить її ідеальним вибором для розробки різноманітних програм та додатків. Її розширюваність та постійний розвиток забезпечують, що ми маємо безліч можливостей для творчості та інновацій у світі програмування.

2.2 Обґрунтування вибору платформи

Java є однією з найпопулярніших мов програмування у світі, і вибір цієї платформи для розробки програм може бути обґрунтований численними перевагами. Обґрунтування вибору платформи Java для розробки програмного забезпечення є суттєвим і має багато переконливих аргументів.

Вибір правильної платформи визначає успіх та ефективність проекту, і ось чому Java є превалюючим вибором для багатьох розробників.

По-перше, однією з найсуттєвіших переваг Java є його кросплатформовість. Код, написаний на Java, може працювати на різних

операційних системах, таких як Windows, macOS та Linux. Це дає можливість створювати програми, які легко переносити з однієї платформи на іншу, забезпечуючи гнучкість і широку аудиторію користувачів.

По-друге, Java має потужну та розширену екосистему. Вона включає в себе багато бібліотек, фреймворків і інструментів розробки, що спрощують процес розробки програм та дозволяють швидше втілення ідей. Багатство екосистеми дозволяє розробникам використовувати готові компоненти і зосередитися на основних функціях програми, що збільшує продуктивність та якість розробки.

По-третє, Java має високий рівень безпеки. Вона має вбудовану систему управління пам'яттю та механізми контролю доступу, що допомагають уникати помилок, таких як витіки пам'яті та доступ до некоректних об'єктів. Це особливо важливо для розробки додатків, які працюють з важливою та конфіденційною інформацією. Крім того, Java має велику активну спільноту розробників. Існує багато онлайн-ресурсів, форумів та спільнот, де розробники можуть обмінюватися досвідом, задавати питання та отримувати підтримку. Це дозволяє швидше розв'язувати проблеми та ефективніше працювати над проектами [1,5,9,13].

2.3 Використання Framework

При написанні я надав перевагу Spring Boot Framework адже Spring Boot - це інноваційний фреймворк, який дозволяє розробникам швидко створювати високоякісні додатки на основі Java з мінімальними зусиллями та складностями. Він поєднує в собі потужну екосистему Spring Framework та забезпечує просту конфігурацію та автоматичне управління залежностями.

Однією з головних переваг Spring Boot є його здатність до автоматичної конфігурації. Замість того, щоб витрачати час на ручну налаштування додатків, розробник може використовувати анотації та

стандартні налаштування для швидкої конфігурації. Spring Boot визначає стандартні налаштування за замовчуванням, що дозволяє розробникам розпочати роботу з мінімальними налаштуваннями та зосередитися на розробці функціональності.

Ще однією сильною стороною Spring Boot є його підтримка вбудованого сервера. Фреймворк постачається з різними вбудованими серверами, такими як Tomcat або Jetty, що дозволяє розробникам запускати та тестувати свої додатки без необхідності встановлення окремого сервера. Це забезпечує швидку розробку та спрощує розгортання додатків.

Spring Boot також має потужну систему автоматичного управління залежностями. Розробник може визначити необхідні залежності в файлі конфігурації та Spring Boot автоматично завантажуватиме та управлятиме цими залежностями. Це дозволяє швидко і легко використовувати різноманітні бібліотеки та модулі без необхідності вручну керувати версіями та завантаженням (рис. 2.1).

Завдяки всім цим функціям, Spring Boot став незамінним інструментом для розробки додатків на Java.

Він дозволяє розробникам швидко розпочати проект та розвивати його, забезпечуючи широкі можливості та простоту використання.

Дивлячись на всі переваги Spring Boot, я вирішив використати його для створення свого веб-сайту. Ця фреймворк-скелетон дозволив мені значно спростити процес розробки, забезпечуючи ряд переваг, такі як: швидкість розробки та інтеграція.

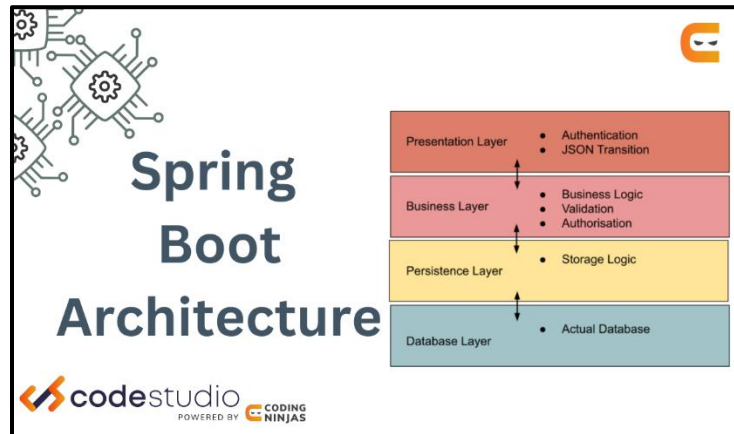


Рисунок 2.1 - Архітектура Spring Boot

Spring Boot спрощує процес розробки, забезпечуючи ефективність та надійність додатків [2,6,10,14].

2.4 H2: Легка та Ефективна Вбудована База Даних

H2 є однією з найпопулярніших вбудованих баз даних, яка широко використовується в розробці програмного забезпечення. Вона відома своєю легкістю використання, швидкістю та ефективністю, що робить її привабливим вибором для розробників.

Однією з ключових переваг H2 є його вбудована функціональність. Це означає, що база даних може бути вбудована безпосередньо в додаток, що спрощує розгортання та управління базою даних. Розробники можуть використовувати H2 для збереження та обробки даних, не потребуючи окремого сервера баз даних. H2 підтримує багато функціональних можливостей, включаючи підтримку SQL-запитів, транзакцій, індексування, процедурного програмування та багато іншого. Вона надає розробникам потужний інструментарій для роботи з даними та забезпечує високу продуктивність завдяки своїй оптимізації та швидкодії.

Крім того, H2 підтримує різні режими роботи, включаючи режим пам'яті, файловий режим та режим змішаного режиму. Це дозволяє

розробникам вибирати оптимальний режим збереження даних, враховуючи вимоги проекту та обсяг даних.

H2 також має гарну підтримку для розробників, забезпечуючи документацію, приклади та активну спільноту, що стимулює обмін знаннями та розвиток інструменту. Розробники можуть швидко знайти відповіді на свої питання та рішення для своїх завдань (рис. 2.2).

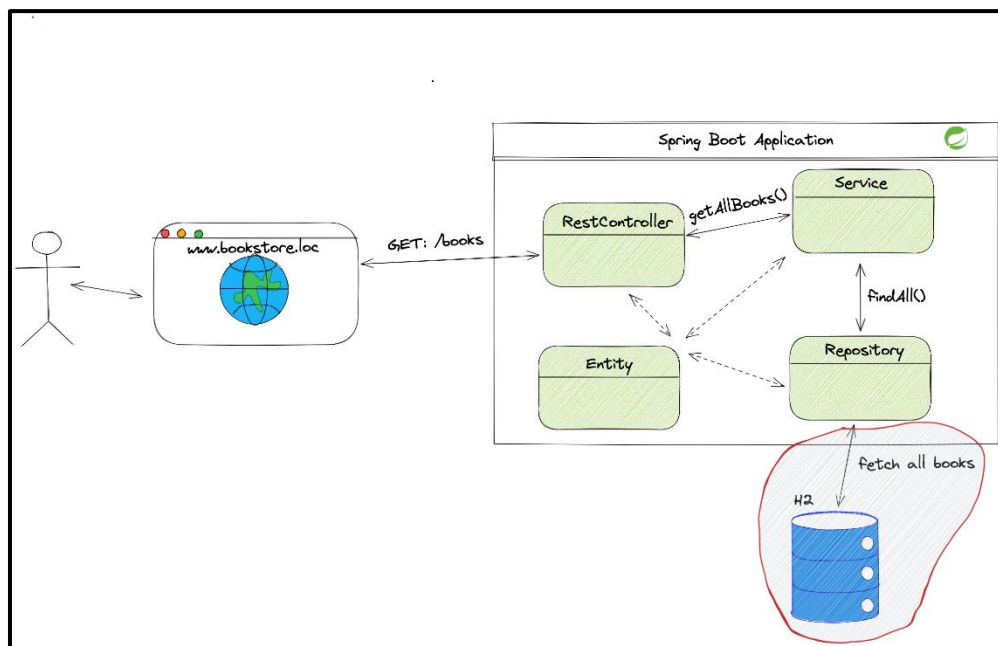


Рисунок 2.2 - H2 Database

Узагалі, H2 є потужним та простим в використанні інструментом для збереження та обробки даних. Вона дозволяє розробникам ефективно працювати з базою даних, незалежно від розміру проекту чи обсягу даних. Завдяки своїм перевагам, H2 є популярним вибором серед розробників програмного забезпечення [4,8,20].

2.5 Spring Data JPA: Зручність та Продуктивність Роботи з Базами Даних

Spring Data JPA є одним із ключових модулів у екосистемі Spring Framework, який надає простий та ефективний спосіб взаємодії з базами даних у Java-проектах. Він поєднує в собі силу Spring Framework та стандарти JPA (Java Persistence API), що дозволяє розробникам працювати з базами даних на більш вищому рівні абстракції. Однією з головних переваг Spring Data JPA є спрощення роботи з базами даних за допомогою JPA, що зображено на (рисунку 2.3). Розробники можуть визначати сутності (Entity) та їх взаємозв'язки за допомогою анотацій, що дозволяє автоматично створювати таблиці у базі даних, виконувати CRUD-операції (створення, читання, оновлення, видалення) та виконувати складні запити за допомогою методів з вбудованими назвами. Це спрощує розробку та підтримку базового шару доступу до даних.

Ще однією сильною стороною Spring Data JPA є його можливості автоматичного створення запитів. За допомогою розумного інференсу запитів, розробники можуть виконувати складні запити до бази даних, використовуючи назви методів, без потреби писати власні SQL-запити. Це робить розробку більш продуктивною та зменшує кількість коду, що потрібно написати та підтримувати.

Spring Data JPA також підтримує транзакції, кешування та інші важливі функції, що забезпечують надійність та продуктивність роботи з базами даних. Він підтримує багато різних баз даних, включаючи MySQL, PostgreSQL, Oracle та інші, що дає розробникам велику гнучкість у виборі платформи. Загалом, Spring Data JPA спрощує роботу з базами даних у Java-проектах та забезпечує швидкий та продуктивний доступ до даних. Він допомагає розробникам зосередитись на логіці додатку, забезпечуючи максимальну ефективність та гнучкість взаємодії з базами даних.[18].

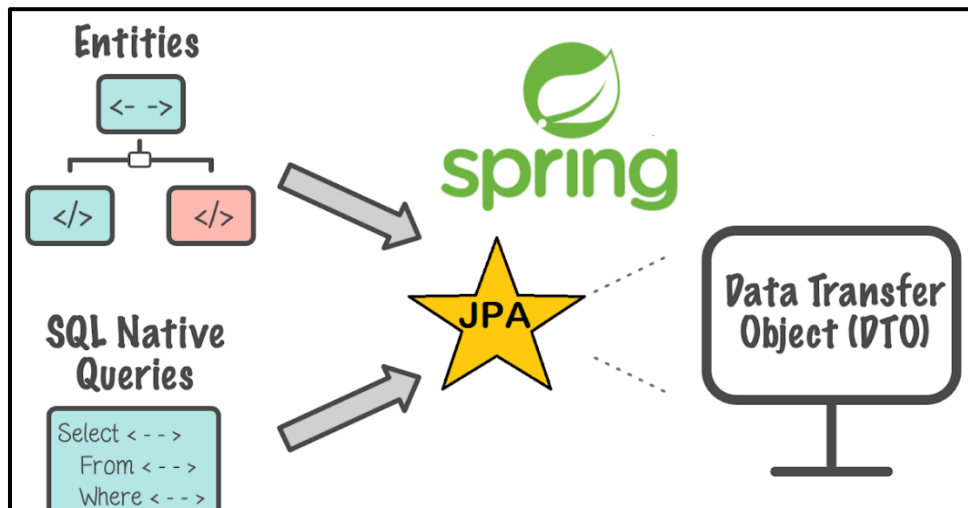


Рисунок 2.3 - Структура Spring Data JPA

2.6 Maven: Ефективне Управління Залежностями та Збиранням Проектів

Maven є однією з найпопулярніших систем збирання проектів у світі розробки програмного забезпечення. Вона дозволяє розробникам ефективно управляти залежностями проекту, автоматизувати процес збирання та розгортання, а також забезпечує стандартизацію та керування проектним життєвим циклом.

Однією з основних переваг Maven є його система керування залежностями. Розробники можуть визначати залежності проекту у файлі `pom.xml`, який є основним конфігураційним файлом Maven. Maven автоматично завантажує та встановлює залежності з центрального репозиторію Maven або локального репозиторію, що дозволяє уникнути ручного керування бібліотеками та їх версіями. Крім того, Maven надає стандартизовану структуру проекту, що сприяє кращій організації та керуванню проектом. Вона визначає конвенції структури каталогів та найменування файлів, що дозволяє розробникам швидко орієнтуватись у проекті та спрощує співпрацю між розробниками. Крім того, Maven

автоматично генерує проектну документацію, забезпечуючи її актуальність та доступність.

Ще однією перевагою Maven є його здатність автоматизувати процес збирання, тестування та розгортання проекту. За допомогою конфігураційного файлу `pom.xml`, розробники можуть описати необхідні кроки та плагіни, що виконуються під час збирання проекту. Maven забезпечує послідовне та надійне виконання цих кроків, що дозволяє зосередитись на розробці програмного забезпечення, а не на складних процедурах збирання. Принцип роботи системи Maven можемо подивитись на (рис.2.4).

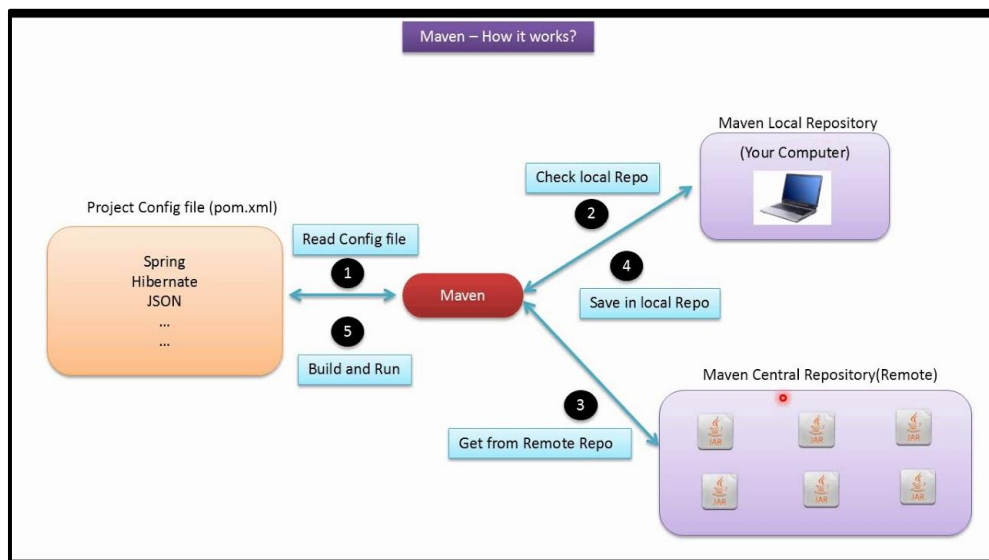


Рисунок 2.4 - Принцип роботи Maven

В цілому, Maven є потужним інструментом для управління залежностями та збиранням проектів, який спрощує та автоматизує процес розробки програмного забезпечення, як ми можемо бачити на рисунку 2.6. Використання Maven дозволяє зосередитись на основній функціональності проекту та забезпечити його ефективність та стабільність [12].

2.7 Thymeleaf

Thymeleaf є однією з найпопулярніших шаблонних систем для веб-розробки у Java-проектах. Вона надає розробникам зручний спосіб розробки динамічних веб-сторінок, що поєднує легкість використання та потужність функціоналу (рис.2.5). Однією з ключових переваг Thymeleaf є його синтаксис, який є природним та інтуїтивно зрозумілим. Він базується на HTML та дозволяє вставляти динамічні дані, виконувати умовні конструкції, цикли, обробляти форми та інші дії безпосередньо в HTML-коді. Це робить розробку та розуміння шаблонів більш приємною та ефективною для розробників. Thymeleaf також підтримує широкий спектр функціональності, що дозволяє розробникам працювати з даними, локалізацією, міжнародними кількостями, управлінням макетами та багато іншого. Він інтегрується з іншими фреймворками, такими як Spring Framework, і надає потужні можливості для розробки повнофункціональних веб-додатків.

Ще однією перевагою Thymeleaf є його підтримка серверного та клієнтського рендерингу. Він може працювати як на серверній стороні, генеруючи HTML на сервері та відправляючи його клієнту, так і на клієнтській стороні, використовуючи JavaScript для динамічного оновлення сторінки. Це надає розробникам більшу гнучкість та можливість вибору підходу в залежності від вимог проекту.

Повністю інтегрований з Spring: Thymeleaf добре поєднується з Spring Framework і особливо добре працює з Spring Boot. Він пропонує природну інтеграцію з іншими компонентами Spring, такими як контролери та сервіси, що дозволяє легко передавати дані між сервером та клієнтом.

Також перевага в нативній підтримці HTML: Thymeleaf підтримує синтаксис HTML, що дозволяє розробникам зручно впроваджувати шаблони та розмітку без необхідності вивчення нової мови розмітки.

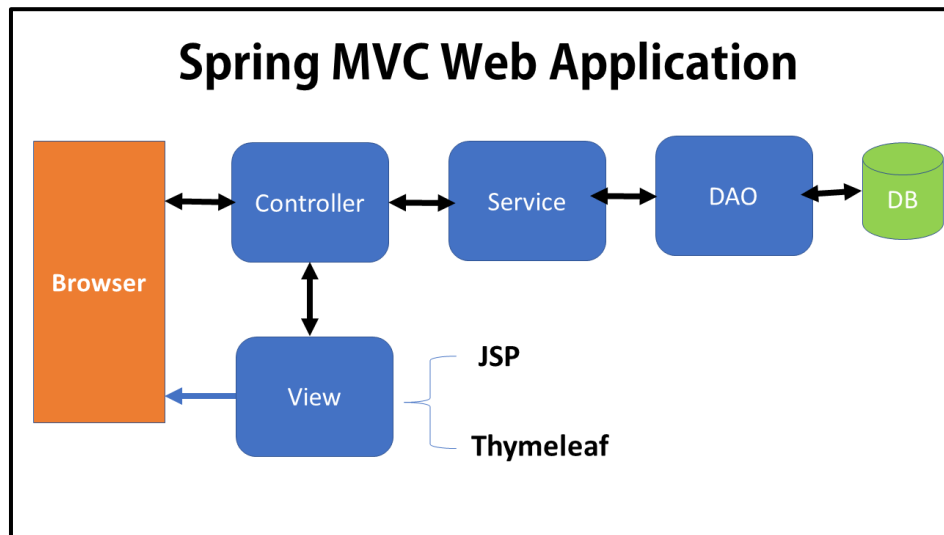


Рисунок 2.5 - Структура Thymeleaf

Загалом, Thymeleaf є потужною та елегантною шаблонною системою, яка дозволяє розробникам легко та ефективно створювати динамічні веб-сторінки. Вона є популярним вибором у Java-спільноті та надає розробникам потужні інструменти для реалізації різноманітних веб-додатків [20].

2.8 Use Case Diagram

Use Case Diagram (діаграма варіантів використання) - це діаграма, яка візуалізує функціональність системи та взаємодію між її акторами (користувачами) і варіантами використання (рис.2.6). Вона допомагає розуміти, які дії можуть виконувати актори системи та як вони взаємодіють з системою:

За допомогою Use Case Diagram можна визначити основні функції системи, зрозуміти потреби користувачів та забезпечити взаєморозуміння між командою розробників та замовниками проекту [1-8].

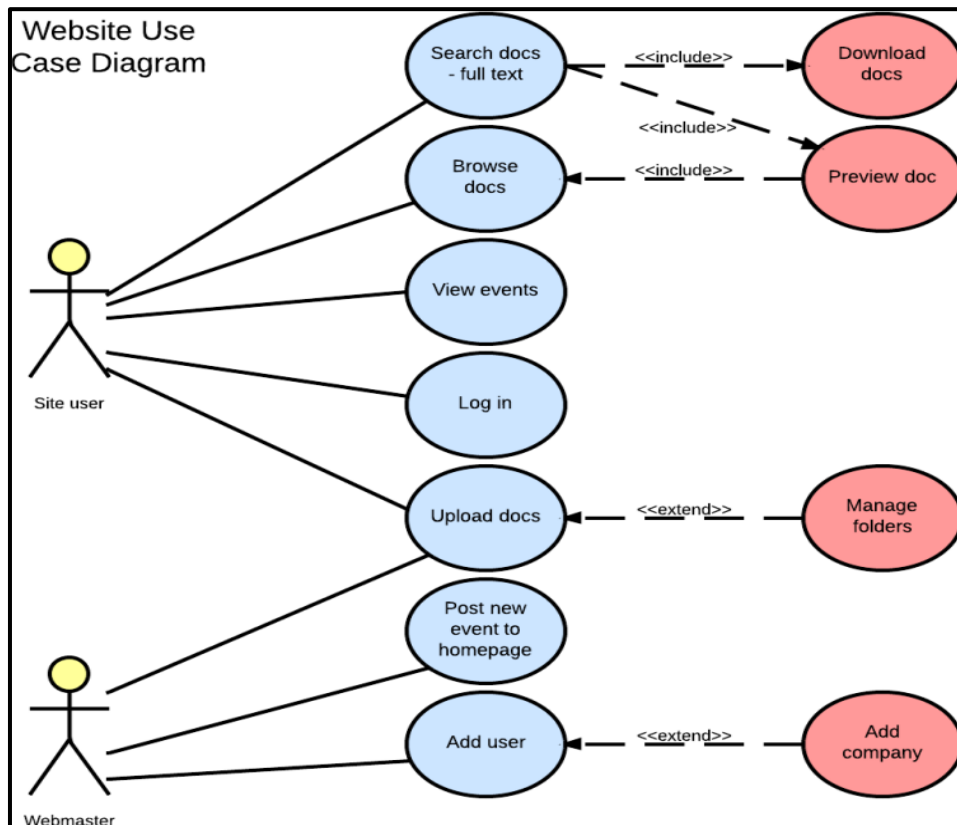


Рисунок 2.6 - Структура Use Case Diagram

Основні елементи Use Case Diagram:

Актори (Actors): Актори представляють ролі, які взаємодіють з системою. Актори можуть бути людьми, іншими системами або зовнішніми процесами чи явищами.

Варіанти використання (Use Cases): Варіанти використання описують конкретні дії або функціональність, які система надає акторам. Кожен варіант використання представляє окремий сценарій взаємодії між актором і системою.

Зв'язки (Relationships): Зв'язки вказують на взаємозв'язки між акторами та варіантами використання. Наприклад, актор може використовувати декілька варіантів використання або один варіант використання може бути доступний для кількох акторів.

Основна мета Use Case Diagram - показати, які функціональність доступна для кожного актора та як актори взаємодіють з системою. Вона

допомагає уточнити вимоги до системи, ідентифікувати акторів, їх потреби та варіанти використання.

Процес створення Use Case Diagram включає ідентифікацію акторів, варіантів використання, їх взаємозв'язків та подальше моделювання цих елементів на діаграмі. Це можна зробити за допомогою спеціалізованих інструментів для моделювання, таких як UML-редактори чи онлайн-інструменти для створення діаграм.

Висновок до розділу 2

Вибір платформи Java, разом з фреймворком Spring Boot, шаблонізатором Thymeleaf та базою даних H2, є відмінним варіантом для розробки сучасних та масштабованих веб-додатків.

Використання Java, Spring Boot, Thymeleaf та H2 database дозволяє розробникам швидко створювати високоякісні та масштабовані веб-додатки.

РОЗДІЛ 3. РЕАЛІЗАЦІЯ САЙТУ

3.1 Написання Use Case Diagram

На (рис. 3.1) зображено основні актори: "Користувач" і "Адміністратор". Основні варіанти використання включають "Реєстрацію", "Авторизацію", "Перегляд виробів", "Деталі продукту", "Відгуки про продукт", "Кошик", "Оформлення замовлення", "Оплату" та "Управління виробами".

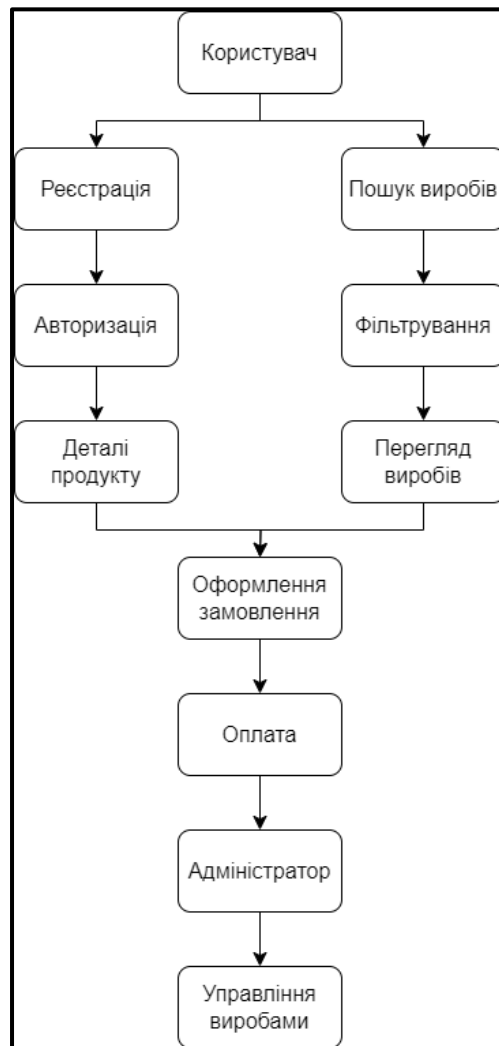


Рисунок 3.1 - Use Case Diagram для мого сайту

Ця діаграма демонструє взаємодію між акторами та функціональність сайту продажу виробів ручної роботи.

3.2 Створення макета сайту

Також перед написанням сайту був створений його макет в шаблонізаторі Thumefleaf, який зображено на (рис 3.2).

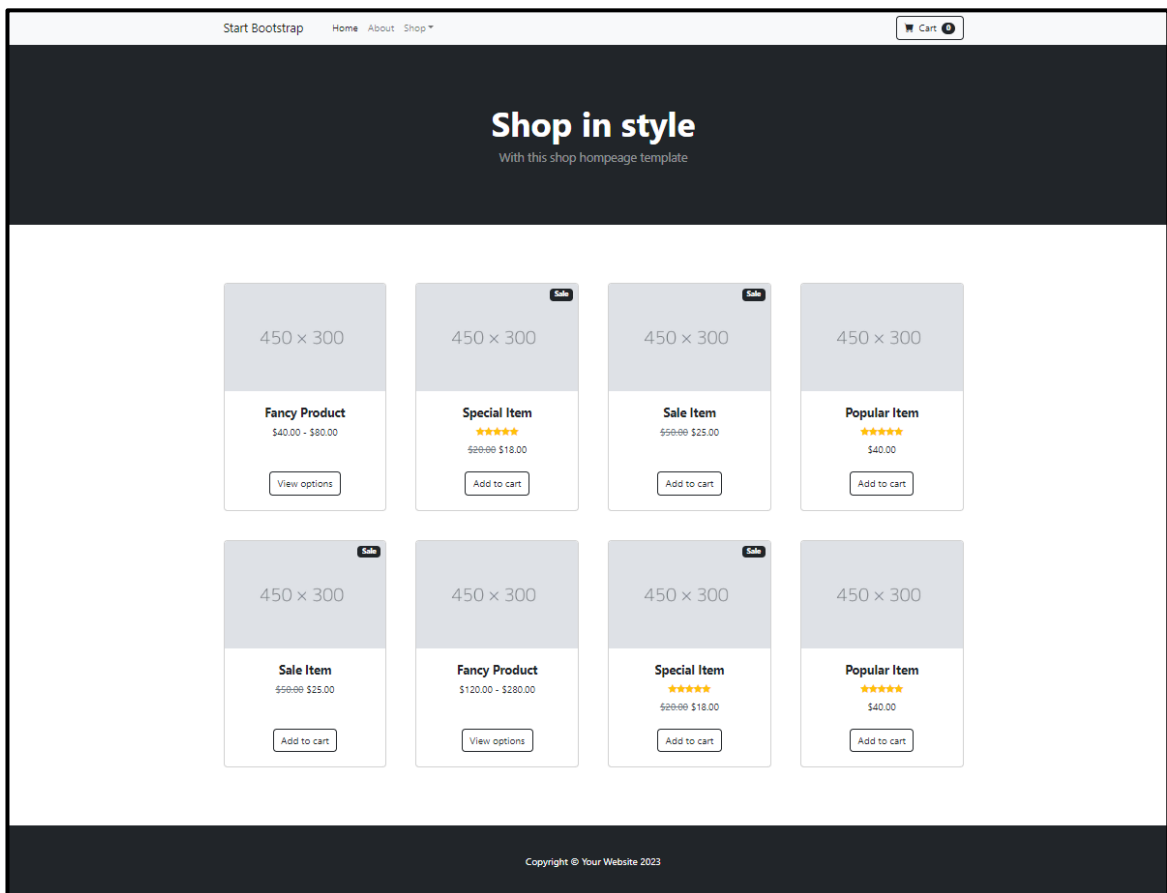


Рисунок 3.2 - Макет сайту

Сайт на головній сторінці буде мати головні функції користувача:

1. обрати товар;
2. додати в корзину;
3. детальний огляд товару і його характеристик;
4. у адміна є привілегії додавати товар і опис до нього + видалення.

Створення макету веб-сайту для виробів ручної роботи - це крок за кроком процес, який дозволяє зобразити вигляд та функціональність майбутнього сайту перед його реалізацією. Цей етап є важливим для забезпечення успіху проекту та задоволення потреб клієнтів. Починаючи зі створення макету, спеціалісти з дизайну та розробки працюють разом, враховуючи вимоги клієнта та розуміючи його цільову аудиторію. Основні кроки створення макету веб-сайту виробів ручної роботи включають наступне:

1. Збір вимог: отримавши детальні вимоги щодо веб-сайту. Обговорюються типи виробів, які будуть представлені, особливості бренду, цільова аудиторія, функціональні можливості та інші важливі аспекти.

2. Дослідження та конкурентний аналіз: я провів детальний аналіз веб-сайтів конкурентів, аналізує їх дизайн, функціональні можливості та інші елементи, які можуть бути використані як вихідна точка для власного макету.

3. Розробка структури сайту: команда визначає логіку та ієрархію сторінок сайту. Вони встановлюють, які категорії товарів будуть на сайті, які розділи та підкатегорії будуть доступні користувачам, як навігація буде впливати на взаємодію зі сторінками.

4. Створення скетчів та візуалізація: за допомогою ручних скетчів або спеціальних програм для дизайну (наприклад, Sketch або Adobe XD), я створив першу візуальну концепцію головної сторінки. Вона виробляє загальний вигляд сайту, включаючи розташування елементів, кольорову палітру, шрифти та інші важливі дизайнерські аспекти.

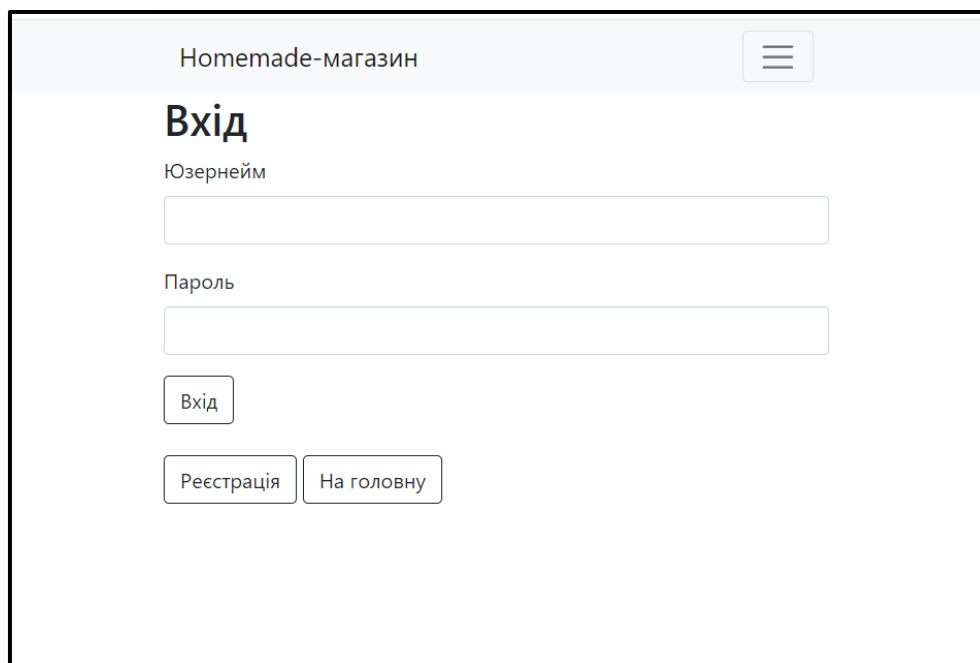
5. Прототипування та тестування: тестування потрібне щоб перевірити функціональність та взаємодію користувачів з сайтом. Прототипи можуть бути протестовані клієнтами або представниками цільової аудиторії, щоб отримати фідбек та внести виправлення.

Створення макету веб-сайту виробів ручної роботи є важливим етапом в процесі розробки. Цей макет служить основою для подальшого створення

та реалізації функціонального сайту, який відповідає потребам клієнтів та привертає увагу цільової аудиторії.

3.3 Реалізація основного функціоналу процесу реєстрації

Щоб клієнт міг спокійно знаходитись на сайт і замовляти товари так як і адмін міг коригувати сайт була створенна реєстрація на сайті. Основний функціонал процесу реєстрації ми можемо бачити на (рис 3.3).



Homemade-магазин

Вхід

Юзернейм

Пароль

Вхід

Реєстрація На головну

Рисунок 3.3 - Меню реєстрації

Щоб клієнт міг спокійно знаходитись на сайт і замовляти товари так як і адмін міг коригувати сайт була створенна реєстрація на сайті. Основний функціонал процесу реєстрації ми можемо бачити на (рис.3.3).

Код процесу реєстрації:

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <meta charset="UTF-8">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/css/bootstrap.min.css"
rel="stylesheet"
```

```

    integrity="sha384-
KyZXEAg3QHqLMpG8r+8fhAXLRk2vvoC2f3B09zVXn8CA5QIVfZOJ3BCsw2P0p/We"
crossorigin="anonymous">
  <title>Вхід</title>
</head>
<body>
<nav th:insert="~{fragments/navbar.html :: navbar}"></nav>
<div class="container">
  <h1>Вхід</h1>
  <form action="/login" method="post">
    <div class="mb-3">
      <label class="form-label">Юзернейм</label>
      <input type="text" class="form-control" name="username">
    </div>
    <div class="mb-3">
      <label class="form-label">Пароль</label>
      <input type="password" class="form-control" name="password">
    </div>
    <button type="submit" class="btn btn-outline-dark">Вхід</button>
  </form>
  <br>
  <a href="/registration" class="btn btn-outline-dark">Реєстрація</a>
  <a href="/" class="btn btn-outline-dark">На головну</a>
</div>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.0/dist/js/bootstrap.bundle.min.js"
  integrity="sha384-
U1DAWAznBHeqEIIVSCgzq+c9gqGAJn5c/t99JyeKa9xxaYpSvHU5awsuZVVFIhvj"
  crossorigin="anonymous"></script>
</body>
</html>

```

Одним із ключових етапів розробки функціоналу сайту була реалізація реєстрації клієнта для замовлення товару. Цей процес передбачає запису даних клієнта у базу даних та його подальше збереження.

3.4 Процес створення головної сторінки сайту

На головній сторінці нашого сайту, який зображено на (рис 3.4), привітно вас зустрічаємо в світ ручної роботи! Ми - ваш надійний партнер у пошуку унікальних та особливих виробів, створених з любов'ю і майстерністю. Ласкаво просимо до нашого віртуального магазину, де ви знайдете широкий асортимент продукції, створеної вручну талановитими

майстрами з усього світу. Ви зможете оцінити неповторний характер кожного виробу, його унікальний дизайн і бездоганну якість.

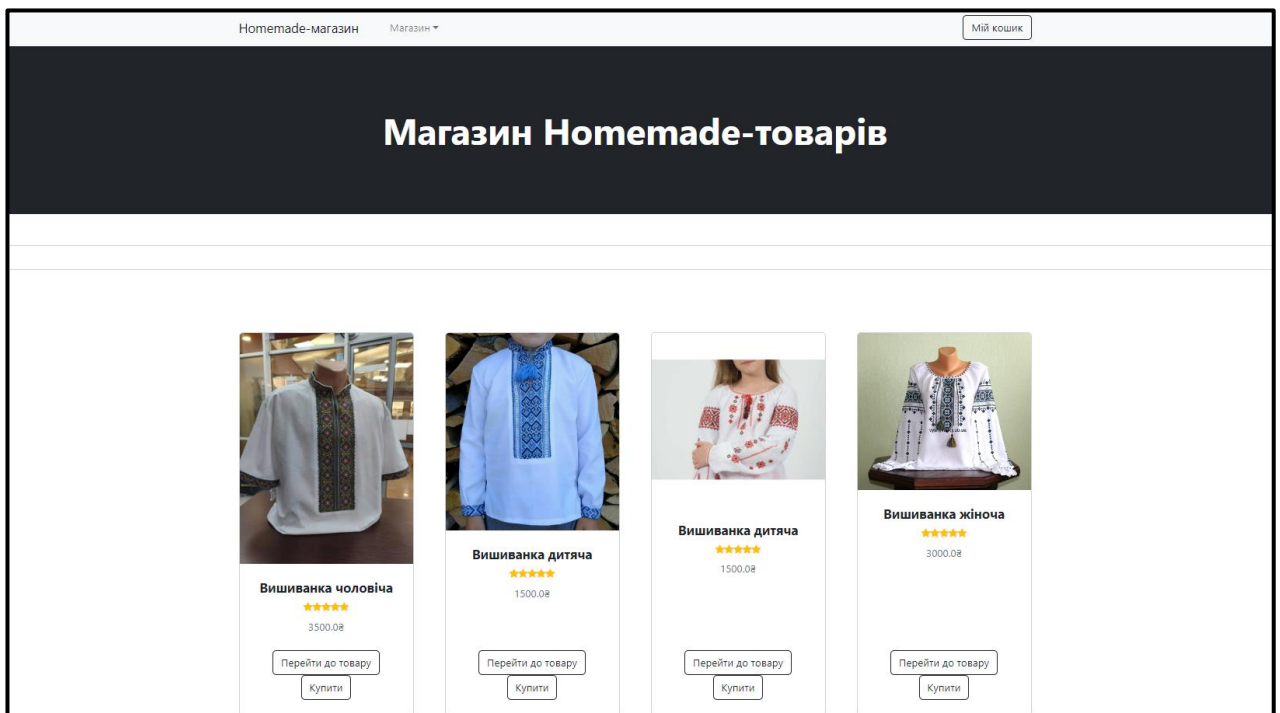


Рисунок 3.4 - Головна сторінка сайту

Наша головна сторінка пропонує вам зручну навігацію, щоб ви могли швидко знайти саме те, що вам потрібно. Ми пишаємося кожним майстром, який створює вироби для нашого магазину, і відчуваємо велику вдячність за їхню працю та талант. Наші вироби - це більше, ніж просто предмети. Вони втілюють душу та енергію майстрів, які вкладають у них частку своєї сутності. Я впевнений, що ви знайдете тут щось особливе, що стане незамінним аксесуаром для вашого дому або чудовим подарунком для вашого близького. Раді бачити вас на нашій головній сторінці! Відчуйте магію ручної роботи та зробіть свій вибір сьогодні [2,19].

Код головної сторінки:

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <meta charset="utf-8"/>
```

```

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no"/>
<meta name="description" content=""/>
<meta name="author" content=""/>
<title>Handmade-магазин</title>

<!-- Bootstrap icons-->
<link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.5.0/font/bootstrap-
icons.css" rel="stylesheet">
<!-- Core theme CSS (includes Bootstrap)-->
<link rel="stylesheet" th:href="@{/css/styles.css}">

</head>
<body>
<!-- Navigation-->
<nav th:insert="~{fragments/navbar.html :: navbar}"></nav>
<!-- Header-->
<header class="bg-dark py-5">
  <div class="container px-4 px-lg-5 my-5">
    <div class="text-center text-white">
      <h1 class="display-4 fw-bolder">Магазин Homemade-товарів</h1>

```

У даних рядках коду ми створювали основну частину сайту. Створення хедера - це важливий етап в процесі розробки веб-сайту. Хедер, або верхня частина сторінки, виконує різноманітні функції, такі як навігація, представлення бренду і створення першого враження у відвідувачів. Давайте розглянемо процес створення хедера крок за кроком.

1. Аналіз вимог: починаємо з аналізу вимог проекту. Розуміємо, яку функціональність має мати ваш хедер. Потрібно занти які елементи повинні бути включені, наприклад, логотип, навігаційне меню, кнопки соціальних мереж, пошукове поле тощо.

2. Розробка концепції: наступним кроком є розробка концепції дизайну. Врахуйте загальний стиль сайту, його кольорову палітру та атмосферу. Ви можете намалювати скетч або використовувати графічні редактори для візуалізації своїх ідей.

3. Створення логотипу: якщо у вас ще немає логотипу, це час його створити або оновити. Розробляємо унікальний логотип, який відобразатиме ваш бренд і відповідатиме стилю хедера.

4. Розташування елементів: розміщуємо елементи нашого хедера відповідно до їхньої важливості та логіки. Зазвичай логотип розміщується зліва або по центру, навігаційне меню - праворуч або горизонтально навколо логотипу. Потрібно приділити увагу чіткості зручності користування

Пам'ятаймо, що хедер повинен відповідати загальному стилю сайту та бути привабливим для відвідувачів.

Створення хедера - це творчий процес, який вимагає уваги до деталей та спритності у використанні дизайну та програмування. Інтегруйте свою уяву, вивчайте тренди та працюйте над створенням хедера, який буде ефективним і привабливим[10-17].

Далі було реалізовано пошук товарів по сайту:

```

</div>
  </div>
</header>
<!-- Section-->
<form class="d-flex mt-5" method="post" action="/product/filter">
  <input class="form-control ml-3 me-2" type="search" placeholder="Пошук"
name="name">
  <button class="btn btn-outline-dark mt-auto" type="submit">Пошук</button>
</form>
<section class="py-5">
  <div class="container px-4 px-lg-5 mt-5">
    <div class="row gx-4 gx-lg-5 row-cols-2 row-cols-md-3 row-cols-xl-4 justify-
content-center">
      <div th:each="product : ${products}" class="col mb-5">
        <div class="card h-100">
          
          <div class="card-body p-4">
            <div class="text-center">
              <!-- Product name-->
              <h5 class="fw-bolder" th:text="${product.getName()}">Special
Item</h5>
              <!-- Product reviews-->
              <div class="d-flex justify-content-center small text-warning mb-2">
                <div class="bi-star-fill"></div>
                <div class="bi-star-fill"></div>
                <div class="bi-star-fill"></div>
                <div class="bi-star-fill"></div>
                <div class="bi-star-fill"></div>
              </div>
              <!-- Product price-->
              <span class="text-muted"
th:text="${product.getPrice()} + '€'"></span>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```


А ще за допомогою наступного коду була створена кнопка для детального огляду кожного товару:

```

    </div>
    <!-- Product actions-->
    <div class="card-footer p-4 pt-0 border-top-0 bg-transparent">
      <div class="text-center"><a class="btn btn-outline-dark mt-auto"
        th:href="/product/' + ${product.getId()}">Перейти
до товару</a></div>
    </div>
  </div>
</div>
</div>
</div>
</div>
</div>
</section>

```

Також важливим елементом сайту є і футер. Він знаходиться внизу сторінки має також свої функціональності та естетичні завдання. Ми створювали футер за допомогою наступного коду:

```

<!-- Footer-->
<footer class="py-5 bg-dark">
  <div class="container"><p class="m-0 text-center text-white">Copyright &copy; Your
Website 2023</p></div>
</footer>
<!-- Bootstrap core JS-->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js"></sc
ript>
<!--Core theme JS-->
<!--<script src="js/scripts.js"></script>-->
</body>
</html>

```

Перед створенням футера потрібно ретельно продумати його функціонал та дизайн і постійно вдосконалювати його для досягнення кращих результатів [15].

3.5 Процес створення коззини

Корзина на сайті з продажу виробів ручної роботи, яку зображено на (рис 3.5), виконує важливу функцію обробки та збереження вибраних товарів перед оформленням покупки.

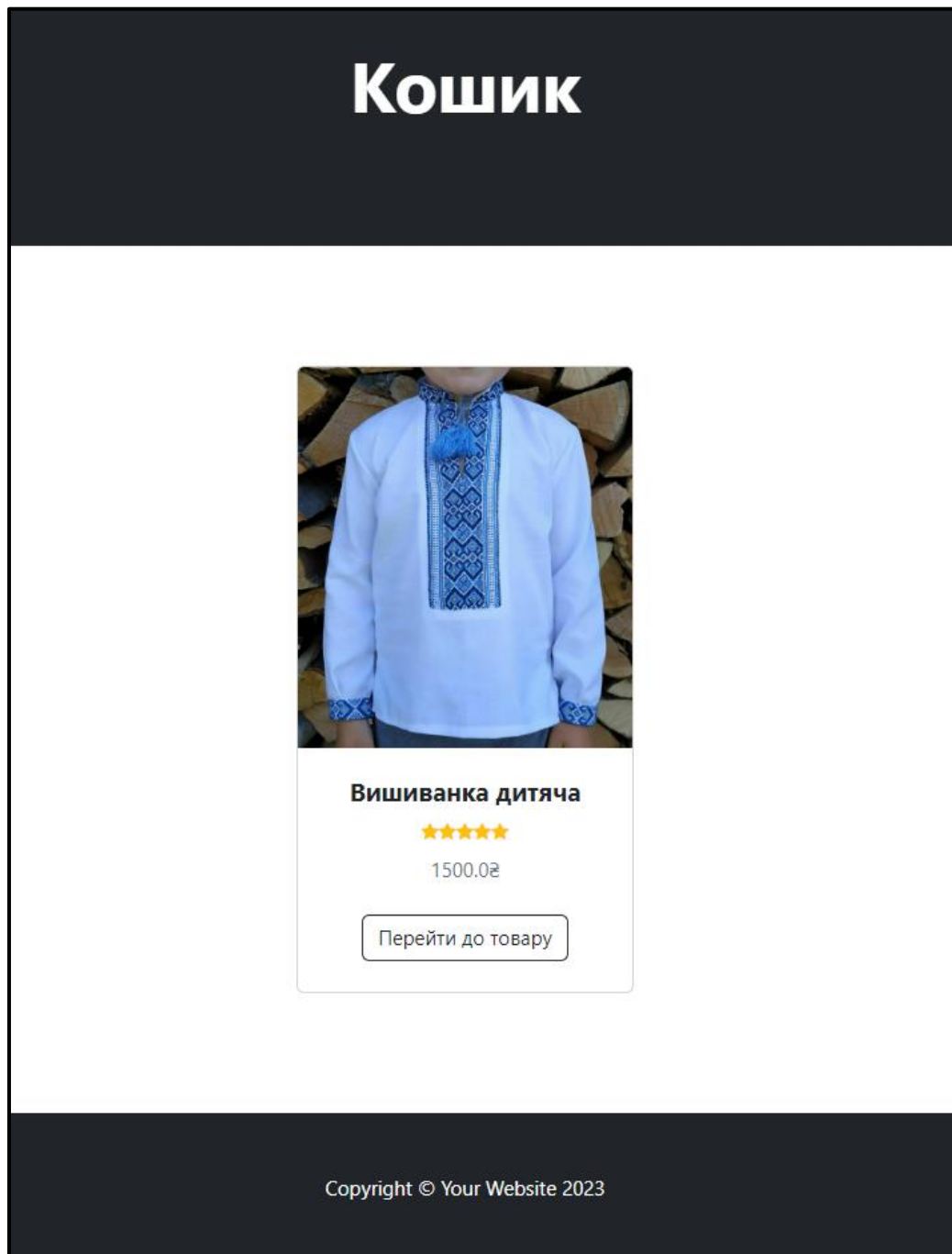


Рисунок 3.5 - Корзина

Код створення корзини:

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
  <meta charset="utf-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-
fit=no"/>
  <meta name="description" content=""/>
  <meta name="author" content=""/>
  <title>Кошик</title>
  <!-- Bootstrap icons-->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.5.0/font/bootstrap-
icons.css" rel="stylesheet">
  <!-- Core theme CSS (includes Bootstrap)-->
  <link rel="stylesheet" th:href="@{/css/styles.css}">
```

Ось деякі основні функції, які присутні в корзині на нашому сайті:

Додавання товарів: користувачі можуть додавати бажані товари до своєї корзини, натискаючи на кнопку "Додати до корзини" або аналогічний елемент, який зазвичай розміщений поруч з кожним товаром на сторінках каталогу або опису товару.

Перегляд корзини: користувачі можуть переглянути зміст своєї корзини, де вони побачать список доданих товарів разом з їх зображеннями, назвами, цінами та кількістю.

Видалення товарів: користувачі можуть видаляти товари з корзини, якщо вони вирішили не придбати їх. Зазвичай, для кожного товару в корзині є кнопка або посилання для видалення [16].

Висновок до розділу 3

У даному розділі було здійснено розробку ключових елементів, які мають значний вплив на якість веб-сайту і забезпечують зручну взаємодію з користувачами. Ці елементи є критичними для досягнення успіху будь-

якого веб-додатка і відповідна розробка їх є надзвичайно важливою для покращення взаємодії та задоволення користувачів.

Головна сторінка є візитною карткою веб-сайту. Вона відображає загальну інформацію про продукт або послуги, які пропонуються, і надає користувачам швидкий доступ до ключових функцій та розділів сайту. Вдало розроблена головна сторінка має чітку структуру, привабливий дизайн і легку навігацію, що сприяє залученню користувачів і збільшує їх інтерес до подальшого взаємодії з сайтом.

Діаграма є важливим інструментом, який візуалізує структуру веб-сайту та взаємозв'язки між різними його компонентами. Вона допомагає розуміти, як розділи та сторінки сайту пов'язані між собою, які функції доступні на кожній сторінці та як користувачі можуть переміщатися по сайту. Це сприяє покращенню навігації та забезпечує зручність використання для користувачів.

Корзина сайту є важливою функцією для веб-додатків, які мають можливість замовлення товарів або послуг. Вона дозволяє користувачам додавати товари до кошика, переглядати їх замовлення, змінювати кількість або видаляти товари, а також оформляти замовлення. Коректна розробка корзини сайту забезпечує зручність процесу покупки для користувачів і допомагає уникнути можливих проблем або помилок при замовленні товарів.

Процес реєстрації є необхідним для багатьох веб-додатків, які надають користувачам індивідуальні облікові записи або можливість зберігати персоналізовані налаштування.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи створено web-застосунок для продажу товарів ручної роботи, який надає зручну можливість онлайн обрати товар і замовити у звичайного майстра. Цей інноваційний інструмент дозволяє клієнтам розглянути чудові вироби мистецтва справжніх майстрів. За допомогою досить простої реєстрації клієнт може авторизуватись на сайті і з легкістю замовити улюблений витвір мистецтва. Сайт створений у допомогу майстрів для реалізації своєї продукції.

У процесі розробки даного проекту було здійснено наступні кроки:

- проведено детальний аналіз переваг та недоліків існуючих аналогічних веб-сайтів;
- вибрано набір технологій для реалізації, зокрема Java, H2 DataBase, Spring Boot, Maven;
- розроблено сучасний та легко сприйнятний дизайн застосунку;
- створено веб сайт з реєстрацією користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Java Documentation: веб-сайт URL: <https://docs.oracle.com/en/java/> (дата звернення: 02.03.2023)
2. Spring Boot Documentation: веб-сайт URL: <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/> (дата звернення 02.03.2023)
3. Thymeleaf Documentation: веб-сайт URL: <https://www.thymeleaf.org/documentation.html> (дата звернення 03.03.2023)
4. H2 Database Documentation: веб-сайт URL: <https://h2database.com/html/main.html> (дата звернення 03.03.2023)
5. Java Tutorials: веб-сайт URL: <https://docs.oracle.com/javase/tutorial/> (дата звернення 03.03.2023)
6. Spring Boot Guides: веб-сайт URL: <https://spring.io/guides#tutorials> (дата звернення 04.03.2023)
7. Thymeleaf Standard Dialect: веб-сайт URL: <https://www.thymeleaf.org/doc/tutorials/3.0/usingthymeleaf.html> (дата звернення 05.03.2023)
8. H2 Database Wiki: веб-сайт URL: <http://www.h2database.com/html/tutorial.html> (дата звернення 03.03.2023)
9. Java API Documentation: веб-сайт URL: <https://docs.oracle.com/en/java/javase/11/docs/api/index.html> (дата звернення 07.03.2023)
10. Spring Boot API Documentation: веб-сайт URL: <https://docs.spring.io/spring-boot/docs/current/api/> (дата звернення 07.03.2023)
11. Thymeleaf Layout Dialect: веб-сайт URL: <https://www.thymeleaf.org/doc/articles/layouts.html> (дата звернення 12.03.2023)
12. H2 Database Javadoc: веб-сайт URL: <http://www.h2database.com/javadoc/> (дата звернення 10.03.2023)

13. Java Language Specification: веб-сайт URL:
<https://docs.oracle.com/javase/specs/jls/se11/html/index.html>
(дата звернення 23.03.2023)
14. Spring Boot Actuator Documentation: веб-сайт URL:
<https://docs.spring.io/spring-boot/docs/current/reference/html/actuator.html> (дата
звернення 24.03.2023)
15. Thymeleaf Spring Integration: веб-сайт URL:
<https://www.thymeleaf.org/doc/articles/springmvcaccessdata.html>
(дата звернення 01.04.2023)
16. H2 Database User Guide: веб-сайт URL:
<http://www.h2database.com/html/quickstart.html> (дата звернення 09.04.2023)
17. Java Concurrency Utilities: веб-сайт URL:
[https://docs.oracle.com/en/java/javase/11/docs/api/java.base/java/util/concurrent/
package-summary.html](https://docs.oracle.com/en/java/javase/11/docs/api/java.base/java/util/concurrent/package-summary.html) (дата звернення 10.04.2023)
18. Spring Boot Testing: веб-сайт URL: [https://docs.spring.io/spring-
boot/docs/current/reference/html/spring-boot-features.html#boot-features-testing](https://docs.spring.io/spring-boot/docs/current/reference/html/spring-boot-features.html#boot-features-testing)
(дата звернення 11.04.2023)
19. Thymeleaf Conditional Expressions: веб-сайт URL:
[https://www.thymeleaf.org/doc/tutorials/3.0/usingthymeleaf.html#attribute-
expressions](https://www.thymeleaf.org/doc/tutorials/3.0/usingthymeleaf.html#attribute-expressions) (дата звернення 10.04.2023)
20. H2 Database SQL Grammar: веб-сайт URL:
<http://www.h2database.com/html/grammar.html> (дата звернення 15.04.2023)