

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет
імені Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук,
проф.

_____ Ю.П. Кондратенко
«__» червня 2022 р.

БАКАЛАВРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

**на тему: «Інформаційна система створення онлайн платформи торгових
операцій за допомогою Shopify»**

Спеціальність 122 «Комп'ютерні науки»

122 – БКР – 401.21810107

Виконав студент 4-го курсу, групи 401

В.В. Голець

(підпис, ініціали та прізвище)

«__» червня 2022 р.

Керівник канд. техн. наук, доцент

(наук. ступінь, вчене звання)

І.О. Калініна

(підпис, ініціали та прізвище)

«__» червня 2022 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет ім. Петра Могили
Факультет комп'ютерних наук
Кафедра інтелектуальних інформаційних систем

Рівень вищої освіти **бакалавр**
Спеціальність **122 «Комп'ютерні науки»**
(шифр і назва)
Галузь знань **12 «Інформаційні технології»**
(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри інтелектуальних
інформаційних систем, д-р техн. наук,
проф.

_____ Ю. П. Кондратенко

«___» _____ 2022р.

ЗАВДАННЯ
на виконання бакалаврської кваліфікаційної роботи

Видано студенту групи 401 факультету комп'ютерних наук

_____ Голецю Владиславу Володимировичу

(прізвище, ім'я, по батькові студента)

1. Тема дипломної роботи

Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify

Затверджена наказом по ЧНУ від «7» грудня 2021 р. № 318

2. Строк представлення дипломної роботи «__» _____ 2022 р.

3. Вхідні (початкові) дані до роботи: інформація від замовника, дані для наповнення сайту, ТЗ замовника

Очікуваний результат роботи:

Очікуваним результатом кваліфікаційної роботи бакалавру є розроблена інформаційна система за допомогою платформи Shopify, зі застосуванням сучасних підходів, методів та використанням WEB-технологій
_____:

4. Перелік питань, що підлягають розробці: під час виконання роботи студент має ознайомитися із такими поняттями, як інтернет-магазин, CMS Shopify. Для створення інтернет-магазину потрібно розглянути принципи взаємодії клієнта та адмін-панелі, CMS Shopify та кастомних модулів. _____

5. Перелік графічних матеріалів: презентація .

6. Завдання до спеціальної частини Завдання до спеціальної частини «Охорона праці» складається з одного розділу та висновку до нього. У першому розділі було розглянуто поняття охорони праці, описано основні критерії для роботи програміста.

7. Консультанти:

Консультант	Кафедра (організація)	Частина роботи
ст. викл. О.В. Макарова	Кафедра екології	Спеціальна частина з охорони праці

Керівник роботи

канд. техн. наук, доцент Калініна Ірина Олександрівна

(посада, прізвище, ім'я, по батькові)

(підпис)

Завдання прийнято до виконання

Голець Владислав Володимирович

(прізвище, ім'я, по батькові студента)

(підпис)

Дата видачі завдання « _____ » _____ 2021 р.

КАЛЕНДАРНИЙ ПЛАН
виконання бакалаврської кваліфікаційної роботи
Тема: Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify

№	Найменування роботи	Початок	Закінчення	Примітки
1	Подання заяви на затвердження теми та керівників БКР	27.10.2021	28.10.2021	Виконано
2	Отримання завдання на виконання БКР	08.11.2021	12.11.2021	Виконано
3	Складання календарного плану роботи на весь період виконання КРБ	15.11.2022	16.11.2022	Виконано
4	Отримання завдання на переддипломну практику	21.04.2022	23.04.2022	Виконано
5	Проходження переддипломної практики, збір та аналіз матеріалів до БКР	23.05.2022	04.06.2022	Виконано
6	Розробка звіту з переддипломної практики	04.06.2022	06.06.2022	Виконано
7	Виконання БКР: аналіз сучасного стану задачі вебзастосунків обміну повідомленнями, огляд існуючих технологій, розробка ПЗ	28.02.2022 та 06.06.2022	27.03.2022 та 19.06.2022	Виконано
8	Попередній захист БКР на засіданні комісії кафедри	30.05.2022	31.05.2022	Виконано
9	Доробка та остаточне оформлення БКР	02.06.2022	20.06.2022	Виконано
10	Подання БКР рецензенту	16.06.2022	18.06.2022	Виконано
11	Подання БКР, її електронної копії та інших документів (відгуку, рецензії) до захисту	20.06.2022	20.06.2022	Виконано
12	Захист БКР перед екзаменаційною комісією (ЕК)	27.06.2022	29.06.2022	Виконано

Розробив студент: Голець В.В.
(прізвище, ім'я, по батькові студента)

_____ *(підпис)*

Керівник роботи: канд. техн. наук, доцент Калініна І. О.
(посада, прізвище, ім'я, по батькові)

_____ *(підпис)*

« 11 » _____ 12_____ 2021 р.

АНОТАЦІЯ

до бакалаврської кваліфікаційної роботи

Тема: «Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify»

Студент 401 групи

Голець Владислав Володимирович

Обрана тема вважається **актуальною** сьогодні, тому що число юзерів у всесвітній павутині від початку коронавірусу збільшилася до 4,54 мільярдів, тобто приблизно 298 мільйонів нових користувачів у рік. Такий швидкий темп приводить нових юзерів, з метою зайнятися власним онлайн продажем.

Мета – надання рішення для створення та подальшого управління онлайн платформою по продажу товарів, за рахунок розробки інформаційної системи та налаштування адмін-панелі для продавця, яка буде задовольняти усі поставлені вимоги замовника.

Об`єкт дослідження – автоматизована інформаційна система онлайн торгівлі товарів.

Предмет дослідження – інтернет-магазин на базі платформи e-commerce Shopify, за допомогою якого можна пришвидшити та покращити взаємодію користувача з продавцем.

У бакалаврській кваліфікаційній роботі вирішуються такі **основні завдання:**

- опанування теорій щодо створення інтернет-магазинів і аналізування їх недоліків;
- порівняння видів і вимог для створення інтернет-магазину;
- опис реалізації елементів контенту сайту;
- розробка динамічних веб-сторінок на основі онлайн платформи, якими можна керувати в адмін-панелі.

Бакалаврська кваліфікаційна робота складається з вступу, чотирьох розділів, спеціального розділу з охорони праці, висновків.

У вступі визначається актуальність теми, що приймається за мету та

невеликий огляд поставленої задачі, предмет дослідження та об'єкт дослідження.

У першому розділі бакалаврської кваліфікаційної роботи описано схему роботи інтернет-магазинів, їх класифікація за видами діяльності, огляд та порівняння вже існуючих розробок, вибір CMS-системи та їх порівняння між собою.

У другому розділі бакалаврської кваліфікаційної роботи описано поняття структури сайтів, їх класифікація, основні вимоги системи Google до структури сайту, створення власної схеми структури сайту, яка використана у розробці інформаційної системи.

У третьому розділі бакалаврської кваліфікаційної роботи створено тему в адмін-панелі Shopify, локальне середовище для ініціалізації проекту, вибрано та описано стек інструментів для розробки веб-платформи.

У четвертому розділі бакалаврської кваліфікаційної роботи описано процес програмної реалізації та підходів розробки відносно компонентів у застосунку.

У спеціальній частині бакалаврської кваліфікаційної роботи містяться загальні питання охорони праці, основні правила роботи для професії програміста, головні правила для подальшої безпеки під час праці.

У висновках проводиться аналіз виконаної роботи та розгляд отриманих результатів.

Бакалаврська кваліфікаційна робота викладена на __ сторінок, вона містить __ розділи, __ ілюстрацій, __ таблиць, __ джерел в переліку посилань.

Ключові слова: інформаційна система, розробка програмного забезпечення, продаж товарів, онлайн платформа Shopify, інтернет магазин, адмін-панель, CMS, HTML, CSS, SCSS, Js, VueJS, Liquid.

ANNOTATION

Bachelor's degree work

Topic: «Information system for creating an online trading platform with Shopify»

Student of 401 group

Holets Vladyslav Volodymyrovych

The chosen topic is **considered relevant** today, because the number of users on the World Wide Web since the beginning of the coronavirus has increased to 4.54 billion, or about 298 million new users per year. This fast pace leads to new users in order to engage in their own online sales.

The goal is to provide a solution for the creation and further management of an online platform for the sale of goods, by developing an information system and setting up an admin panel for the seller, which will meet all the requirements of the customer.

The object of study - an automated information system for online trade in goods.

The subject of the research - an online store based on the e-commerce platform Shopify, which can be used to speed up and improve user interaction with the seller.

The following **main tasks** are solved in the bachelor's qualification work:

- mastering theories about creating online stores and analyzing their shortcomings;
- comparison of types and requirements for creating an online store;
- description of the implementation of elements of site content;
- development of dynamic web pages based on the online platform, which can be managed in the admin panel.

Bachelor's thesis consists of an introduction, four sections, a special section on labor protection, conclusions.

The introduction determines the relevance of the topic to be taken as a goal and a brief overview of the task, the subject of research and the object of research.

The first section of the bachelor's thesis describes the scheme of online stores, their classification by type of activity, review and comparison of existing developments, the choice of CMS-system and their comparison with each other.

The second section of the bachelor's thesis describes the concept of site structure, their classification, the basic requirements of the Google system for site structure, creating your own scheme of site structure, which is used in the development of information systems.

In the third section of the bachelor's thesis, a topic was created in the Shopify admin panel, a local environment for project initialization, and a stack of tools for web platform development was selected and described.

The fourth section of the bachelor's thesis describes the process of software implementation and development approaches to the components in the application.

The special part of the bachelor's qualification work contains general issues of labor protection, the basic rules of work for the profession of programmer, the main rules for further safety at work.

In the conclusions the analysis of the executed work and consideration of the received results is carried out.

The bachelor's thesis is presented on __ pages, it contains __ sections, __ illustrations, __ tables, __ sources in the list of references.

Keywords: information system, software development, sales of goods, online platform Shopify, online store, admin panel, CMS, HTML, CSS, SCSS, Js, VueJS, Liquid.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ	4
ВСТУП	6
1 ІНФОРМАЦІЙНА СИСТЕМА СТВОРЕННЯ ОНЛАЙН ПЛАТФОРМИ ТОРГОВИХ ОПЕРАЦІЙ: ПОНЯТТЯ, КЛАСИФІКАЦІЯ, СТРУКТУРА.....	8
1.1 Схеми роботи інтернет-магазину	8
1.2 Класифікація інтернет-магазинів за видами діяльності.....	9
1.3 Постановка задачі.....	11
1.4 Огляд та аналіз існуючих розробок	12
1.5 Вибір CMS-системи	18
Висновки до розділу 1.....	20
2 ТЕХНОЛОГІЇ ТА ПІДХОДИ ДО ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ СТВОРЕННЯ ПЛАТФОРМИ ТОРГОВИХ ОПЕРАЦІЙ ЗА ДОПОМОГОЮ SHOPIFY	22
2.1 Вимоги системи Google до структури сайту	23
2.2 Створення схеми структури сайту	24
Висновки до розділу 2.....	25
3 ТЕХНІЧНЕ НАЛАШТУВАННЯ СЕРЕДОВИЩА РОЗРОБКИ ІНТЕРНЕТ-МАГАЗИНУ ТА ОПИС СТЕКУ ІНСТРУМЕНТІВ РОЗРОБКИ.....	26
3.1 Створення та налаштування теми в адмін-панелі Shopify перед початком роботи	26

3.2 Створення локального середовища для ініціалізації проекту та написання базового коду для швидкого старту роботи	33
3.3 Вибір та опис стеку інструментів для розробки веб-платформи	42
Висновки до розділу 3.....	56
4 ПРОЦЕС ПРОЕКТУВАННЯ СТОРІНОК ТА КОМПОНЕНТІВ ВЕБЗАСТОСУНКУ	57
4.1 Опис програмної реалізації.....	57
Висновки до розділу 4.....	83
5 ОХОРОНА ПРАЦІ.....	84
5.1 Загальні питання охорони праці.....	86
Висновки до розділу 5.....	100
ВИСНОВКИ	102
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	104

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

CSS	–	Cascading Style Sheets
HTML	–	HyperText Markup Language
JS	–	JavaScript
JSON	–	JavaScript Object Notation
SASS	–	Syntactically Awesome Stylesheets

Кафедра інтелектуальних інформаційних систем
Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify

Пояснювальна записка

до бакалаврської кваліфікаційної роботи

«ІНФОРМАЦІЙНА СИСТЕМА СТВОРЕННЯ ОНЛАЙН ПЛАТФОРМИ ТОРГОВИХ ОПЕРАЦІЙ ЗА ДОПОМОГОЮ SHOPIFY»

Спеціальність 122 «Комп'ютерні науки»

122-БКР-401.21810107

Виконав: студент 4-го курсу, групи 401
В.В. Голець
(підпис, ініціали та прізвище)
«__» _____ 2022 р.

Керівник: старший викладач кафедри ІІЗ
(наук. ступінь, вчене звання)
І.О. Калініна
(підпис, ініціали та прізвище)
«__» _____ 2022 р.

Миколаїв – 2022

ВСТУП

З року в рік застосування мережі Інтернет постійно зростає, це означає, що інформаційні технології не стоять на місці і набувають потужний вплив на життя людини.

Згідно до інформації, що представлена у ресурсі «Digital Overview», число юзерів у всесвітній павутині від початку коронавірусу збільшилася до 4,54 мільярдів, тобто приблизно 298 мільйонів нових користувачів у рік, якщо починати з 2020 року.

Не менш ніж 60% людей у всесвіті в поточний час мають доступ до мережі Інтернет, але проблема, що має назву «Цифровий розрив» залишається актуальною на сьогодні. Цифровим розривом називається різниця наявних умов у користувачів до Інтернету між цифровими технологіями.

У всіх країнах пріоритетом є збільшення кількості юзерів у мережі, тож інформаційні технології прямують до вершини розвитку.

З підрахунків ООН, в найближчий час відкриють що найменш 2 мільйони нових вакансій у сфері ІТ. Приблизно 85% цих професій потребуватимуть знань у сфері цифрових технологій. Звідси можна зробити висновок, що з зростом нових місць – також збільшується цифровий розрив.[1]

Обрана тема вважається **актуальною** сьогодні, тому що число юзерів у всесвітній павутині від початку коронавірусу збільшилася до 4,54 мільярдів, тобто приблизно 298 мільйонів нових користувачів у рік. Такий швидкий темп приводить нових юзерів, з метою зайнятися власним онлайн продажом. За допомогою інтернет-магазину можливо підвищити чинний цифровий продаж для малого і середнього бізнесу, розширити клієнтську базу підприємства в онлайн форматі. Це дозволить спростити та пришвидшити процеси торгових операцій щодо купівлі будь-яких продуктів не виходячи з дому.

Метою є надання рішення для створення та подальшого управління онлайн платформою по продажу товарів, за рахунок розробки інформаційної

системи та налаштування адмін-панелі для продавця, яка буде задовольняти усі поставлені вимоги замовника.

Від початку виконання роботи були визначені наступні **завдання**:

- опанування теорій щодо створення інтернет-магазинів і аналізування їх недоліків;
- порівняння видів і вимог для створення інтернет-магазину;
- опис реалізації елементів контенту сайту;
- розробка динамічних веб-сторінок на основі онлайн платформи, якими можна керувати в адмін-панелі.

Об`єктом дослідження є автоматизована інформаційна система онлайн торгівлі товарів.

Предметом дослідження є інтернет-магазин на базі платформи e-commerce Shopify, за допомогою якого можна пришвидшити та покращити взаємодію користувача з продавцем.

1 ІНФОРМАЦІЙНА СИСТЕМА СТВОРЕННЯ ОНЛАЙН ПЛАТФОРМИ ТОРГОВИХ ОПЕРАЦІЙ: ПОНЯТТЯ, КЛАСИФІКАЦІЯ, СТРУКТУРА

1.1 Схема роботи інтернет-магазину

Інтернет-магазин – це певний електронний ресурс, сайт з певним каталогом, головною ціллю якого є продаж товарів покупцю, враховуючи доставку до бажаного місця. В цілому покупка продукту і угода відбуваються безпосередньо на сайті магазину. Основною відмінністю інтернет-магазину від звичайного магазину є відсутність потреб будь-яких фізичних засобів. Якщо в звичайному магазині потрібні приміщення, вітрини, персонал, цінники, то в інформаційній системі вся ця структура реалізована у вигляді вебзастосунку.[2]

Зовні схема роботи в усіх інтернет-магазинів приблизно схожа: покупець переходить до сайту, обирає продукти серед усього каталогу і виконує замовлення на бажану адресу, потім замовлення надсилається до служби пошти, покупець отримує товар і оплачує його (якщо гроші не були перераховані раніше). Після цього зазвичай розпочинається робота з покупцем (email-маркетинг, реклама тощо задля зацікавлення його новими продуктами).

Інтернет-магазин в свою чергу є головним зв'язком між покупцем і продавцем, якому достатньо бути підключеним до мережі Інтернет. Саме в інтернет-магазинах ціни мінімальні, тому що продавцю необов'язково мати офіс, платити за оренду приміщення та наймати робітників. Зазвичай споживач отримує товар безпосередньо зі складу продавця.

Основним завдання інтернет-сервісу – є надання покупцям зручної взаємодії із продавцем. Важливо, щоб сайт був не перевантажений зайвою інформацією, тобто покупець повинен мати змогу швидко знайти зацікавлену річ, подивитися її більше детально та виконати замовлення.

1.2 Класифікація інтернет-магазинів за видами діяльності

Інтернет-магазини можна віднести до таких видів [3]:

- класичний;
- інтернет-магазин на аутсорсингу;
- drop-shipping магазин;
- партнерський інтернет-магазин;
- змішаний;
- інтернет-аукціон.

Класичний:

Це модель, що виконує продаж через офлайн систему, рано чи пізно приходиться до думки побудувати додаткове джерело клієнтів через глобальну мережу. Бухгалтери, юристи, партії товарів, різні типи співробітників відносяться до класичного бізнесу, який в свою чергу характеризується цим видом моделі.

Інтернет-магазин на аутсорсингу:

Це модель, за якої продавець виконує лише організаційні бізнес-проблеми та справи, тобто просто передає на аутсорс сторонній компанії. За допомогою цієї моделі можна отримувати значний прибуток у розмірі 65-75% від продаж в інтернет-магазині, а також не губити багато часу.

Drop-shipping магазин:

Напевно, зрозуміло, що цей тип здійснює продаж за так званою схемою «drop-shipping». У цьому випадку, продавець може не бути безпосередньо знайомим з товаром, що продається: не бачити, або не тримати його в руках, і в цьому, власне, немає потреби. Власник виступає у ролі продавця і може сам встановлювати ціну, вам не потрібен склад, оскільки за відправлення та доставку товарів відповідає постачальник. Коли покупець сплатив за товар, частину грошей ви залишаєте собі, а собівартість (оптову ціну) – платите постачальнику. Недоліками подібної моделі є повна передоплата клієнта (що

може викликати певною мірою недовіру, особливо якщо покупець вперше побачив ваш магазин), витрати часу та грошей на доставку. За такою схемою діє і ряд великих компаній, доходи яких обчислюються мільйонами доларів на рік, однак, можна було б заробляти в 3-4 рази більше, маючи власний склад і асортимент товарів.

Партнерський інтернет-магазин:

Це модель, коли в магазин виставляється тільки товар на вітрину (наприклад, особистий блог) і додається посилання на відповідні інтернет-магазини. Виходить, що у власника сайту немає магазину, але завдяки тому, що ресурс будуть відвідувати інші люди, тобто ймовірність, що хтось придбає товар партнера, а вам дістануться комісійні (приблизно 20-25% вартості товару). Також ніхто не заважає просувати свій магазин за допомогою особистого блогу, або зробити окремий блог на відповідну тематику продажу.

Змішаний:

Це модель, коли продавець враховує переваги та недоліки окремих типів Інтернет-магазинів та створює власну бізнес модель Інтернет-магазину, що зібрала кращі характеристики перераховані вище.

Інтернет-аукціон:

Це модель у вигляді аукціону, що проводиться в онлайн режимі. Цей тип є також видом інтернет-магазину, тому що підтримує функцію проведення онлайн-транзакцій. Час закінчення інтернет-аукціону завчасно призначається продавцем при наданні продуктів на торгову платформу. По закінченню інтернет-аукціону покупець повинен перевести кошти продавцю за безготівковим розрахунком (іноді – готівкою, при отриманні товару особисто), а продавець в свою чергу повинен відіслати продукт до покупця сервісом пошти, це може будь-яка точка в країні або всесвіту. Межі можливої пересилки продукту вказує сам продавець від початку продажу.

1.3 Постановка задачі

Основною цілю бакалаврської кваліфікаційної роботи є створення вебзастосунку для діяльності у сфері комерції (eCommerce). Інтернет-магазин має здатність продажу будь-яких товарів, так як створена адмін-панель є повністю динамічною і можна в будь який момент замінити усю тематику і контент сайту на необхідний, з простим і зрозумілим інтерфейсом як для адміністратора, так і для відвідувачів сайту.

Обрана тема вважається **актуальною** сьогодні, тому що число юзерів у всесвітній павутині від початку коронавірусу збільшилася до 4,54 мільярдів, тобто приблизно 298 мільйонів нових користувачів у рік. Такий швидкий темп приводить нових юзерів, з метою зайнятися власним онлайн продажом. За допомогою інтернет-магазину можливо підвищити чинний цифровий продаж для малого і середнього бізнесу, розширити клієнтську базу підприємства в онлайн форматі. Це дозволить спростити та пришвидшити процеси торгових операцій щодо купівлі будь-яких продуктів не виходячи з дому.

Метою є надання рішення для створення та подальшого управління онлайн платформою по продажу товарів, за рахунок розробки інформаційної системи та налаштування адмін-панелі для продавця, яка буде задовольняти усі вимоги замовника.

У ході виконання бакалаврської кваліфікаційної роботи, були поставлені наступні **завдання**:

1. Вивчення теоретичних аспектів створення інтернет-магазинів і виявлення їх недоліків;
2. Порівняння видів і вимог для створення інтернет-магазину;
3. Опис реалізації елементів контенту сайту;
4. Розробка динамічних веб-сторінок інтернет-магазину, якими можна керувати в адмін-панелі.

Об'єктом дослідження є автоматизована інформаційна система онлайн торгівлі товарів.

Предметом дослідження є інтернет-магазин на базі платформи e-commerce Shopify, за допомогою якого можна пришвидшити та покращити взаємодію користувача з продавцем.

1.4 Огляд та аналіз існуючих розробок

Аналізування аналогічних веб-сайтів – це ряд факторів оцінки сайтів подібних до ключового запиту пошуку.

«Zemal.com.ua» www.zemal.com.ua

Zemal.com.ua – це популярний сайт в Україні з продажу одяжі (див. рис. 1.1).

До переваг відносяться:

- наявність важливої інформації на головній сторінці;
- наявність пошуку на сайті;
- товар поділений на певні групи, за допомогою цього користувачу легше взаємодіяти з веб-сайтом;
- наявність форми реєстрації та сторінки з інформацією щодо покупця;
- наявність відгуків на продукти;
- наявність форми зворотного зв'язку при виникненні запитань та проблем;
- наявність вибору мови для користувача.

До недоліків відносяться:

- досить застарілий та незручний інтерфейс веб-сайту з позначки UX/UI дизайну;
- повільна швидкість веб-сайту;
- немає функції "Купити в один клік", щоб прискорити взаємодію сайту з користувачем;

- сайт не адаптований під людей з обмеженими можливостями (не розрахований на управління через клавіатуру та войсовери);
- в консолі розробника на сайті при завантаженні сайту з'являються помилки з виконуваних скриптів.

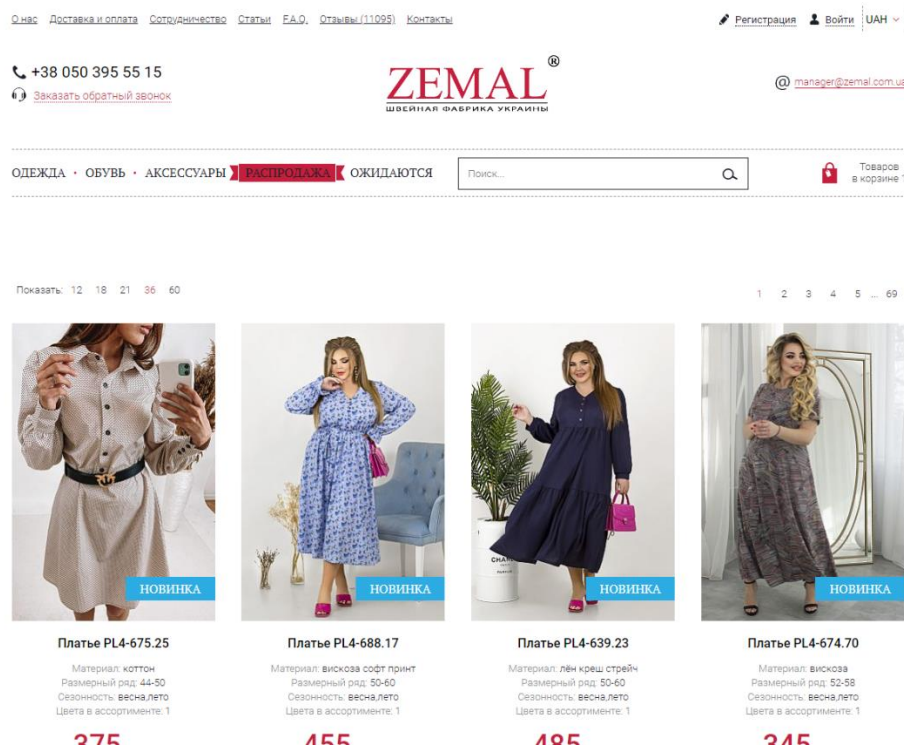


Рисунок 1.1 – головна сторінка сайту «Zemal.com.ua»

«foxtrot.com.ua» <https://www.foxtrot.com.ua/>

foxtrot.com.ua – одна з найбільших в Україні роздрібних мереж за кількістю магазинів та обсягами продажу електроніки та побутової техніки (див. рис. 1.2).

До переваг відносяться:

- на головній сторінці є поле магазинів, де можна подивитися адреси реальних магазинів у потрібному населеному пункті;
- наявність пошуку на сайті;
- товар поділений на певні групи, за допомогою цього користувачу легше взаємодіяти з веб-сайтом;

Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify

- наявність форми реєстрації та сторінки з інформацією щодо покупця;
 - наявність відгуків на продукти;
 - наявність форми зворотного зв'язку при виникненні запитань та проблем;
 - наявність вибору мови для користувача;
 - наявність функції додавання товару до улюблених, щоб потім можна було швидко до нього повернутися та придбати;
 - наявність усіх варіантів оплати товару (готівковий, безготівковий та оплата частинами);
 - наявність посилань на соціальні мережи.
- До недоліків відносяться:
- явних недоліків знайдено не було.

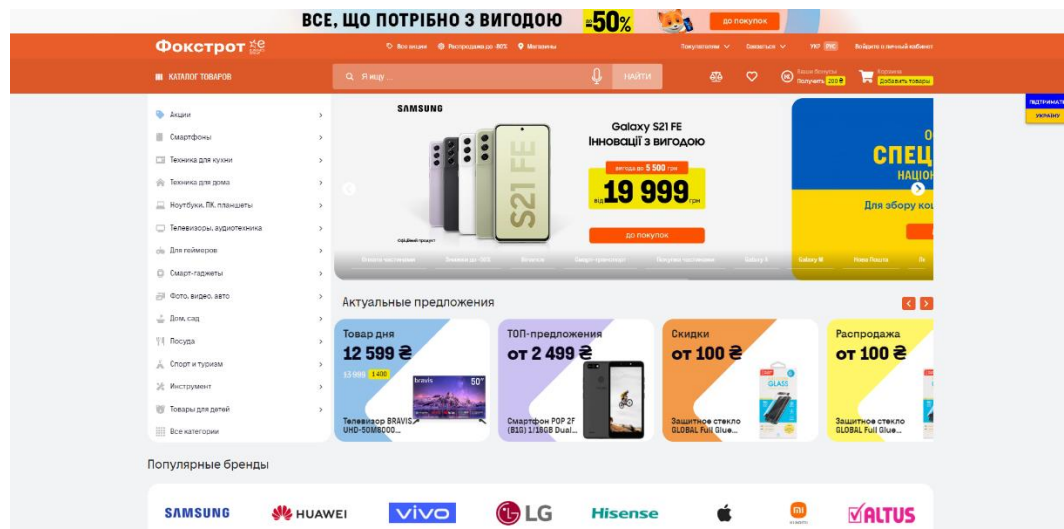


Рисунок 1.2 – головна сторінка сайту «foxtrot.com.ua»

«pervak.kiev.ua» <http://pervak.kiev.ua/>

pervak.kiev.ua – це український сайт популярного ресторану, який знаходиться у Києві (див. рис. 1.3).

До переваг відносяться:

- наявність важливої інформації на головній сторінці;

- товар поділений на певні групи, за допомогою цього користувачу легше взаємодіяти з веб-сайтом;
- наявність функції бронювання столу на веб-сайті;
- наявність вибору мови для користувача;
- наявність карти з точкою геолокації закладу.

До недоліків відносяться:

- мала кількість відгуків від клієнтів;
- деякі секції на сайті виглядають зламаними;
- сторінка певного блюда не має достатнього опису для користувача та виглядає погано з точки UX/UI дизайну;
- сайт неадаптований під мобільні пристрої.

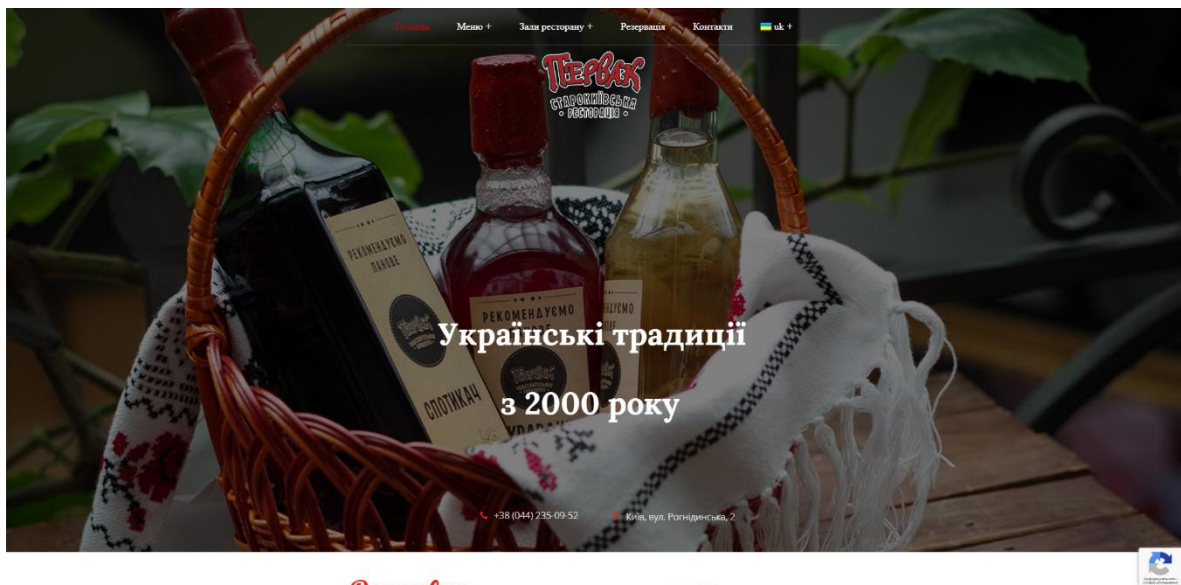


Рисунок 1.3 – головна сторінка сайту «pervak.kiev.ua»

«mebelnuy.com.ua» <https://mebelnuy.com.ua/>

mebelnuy.com.ua – це український сайт меблів, що надає асортимент від кращих фабрик за привабливими цінами (див. рис. 1.4).

До переваг відносяться:

- товар поділений на певні групи, за допомогою цього користувачу легше взаємодіяти з веб-сайтом;

Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify

- наявність важливої інформації на головній сторінці;
 - наявність усіх операторів для зв'язку з магазином;
 - наявність пошуку на сайті;
 - наявність усіх варіантів оплати товару (готівковий, безготівковий та оплата частинами);
 - наявність форми зворотного зв'язку при виникненні запитань та проблем;
- До недоліків відносяться:*
- досить застарілий та незручний інтерфейс веб-сайту з позначки UX/UI дизайну;
 - немає можливості вибрати мову на веб-сайті;
 - в наявності немає форми реєстрації та сторінки з інформацією щодо покупця;
 - сайт погано адаптований під мобільні пристрої.

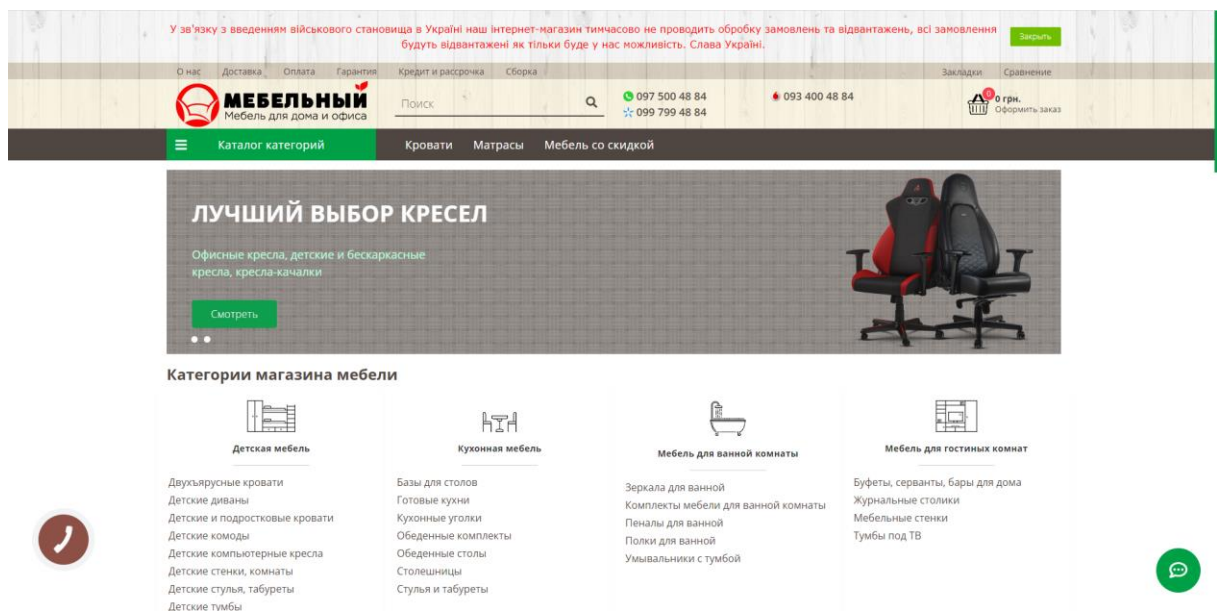


Рисунок 1.4 – головна сторінка сайту «mebelnyy.com.ua»

В результаті аналізу вищезазначених веб-сайтів, можна занести орієнтовні оцінки до таблиці відповідно до кожного критерію (див. табл. 1.1).

Таблиця 1.1 – Орієнтовна оцінка сайтів за певними критеріями

Критерії	zema.ua	foxtrot.ua	pervak.kiev.ua	mebelnuy.ua
Наповнення сайту	3	5	4	3
UX/UI	2	4	4	3
Оновлення наповнення сайту	3	4	4	3
Швидкість	3	4	4	3
Зворотній зв'язок	5	5	4	5
SEO	3	5	5	4
Соціальні мережі	3	5	4	3

1. **Наповнення сайту.** Коли покупець потрапляє до веб-сайту, він звертає увагу на те, з чого зроблений сайт, будь то текст, картинка, відео, аудіо, пости і т.д. Для цієї роботи є спеціальні професії, або при бажанні це можна зробити самостійно на власний розсуд.

2. **UX/UI.** Дизайн веб-сайту відіграє значну роль у формуванні першого враження у клієнта, який потрапив до магазину в перший раз. Але важливо, щоб стиль сайту був в міру, і дизайнер не забував про інтереси купця та не відштовхував його від покупки своїм винахідливим стилем.

3. **Оновлення наповнення сайту.** Досить часто в своїх веб-сайтах продавці забувають оновлювати контент сторінок свіжою інформацією, і це досить погано. Адже у відвідувачів може виникнути враження, що продавець не відстежує сайт і не приймає в ньому участі, при цьому може і знизитися конверсія з продажу продукції.

4. **Швидкість.** Є досить багато гарних сайтів відносно дизайну та наповнення контенту, але не всі з них мають високу оцінку швидкості та взаємодії з відвідувачем. В свою чергу, покупець, який зайшов до магазину – не буде чекати більше 5 секунд, поки сайт завантажиться, він просто вийде і зайде на аналогічний інтернет-магазин.

5. **Зворотній зв'язок.** На веб-сайті завжди повинна бути будь-яка інформація щодо зв'язку з магазином, в свою чергу це можуть бути: номер телефону, email адреса, онлайн-бот. За допомогою цього зв'язку – клієнт зможе контактувати з магазином в повній мірі.

6. **SEO.** В наповненні сайту повинні бути ключові слова та словосполучення, саме через ці параметри сайти індексуються і потрапляють до рекомендованих в пошукових запитах.

7. **Соціальні мережі.** Більшість часу люди проводять саме в соціальних мережах, і не погано було б саме в них зазначати посилання на веб-сайт, де проводиться сама торгівля продуктами.

За допомогою створеної таблиці набагато легше оцінити та порівняти все критерії охоплених веб-сайтів.

Тож, веб-сайт для продажу повинен бути легким у використанні зі сторони відвідувача, не мати перевантаження контенту на сторінках. Зовнішній вигляд застосунку має зацікавити покупця на придбання товару та подальшої взаємодії, щоб після успішних операцій користувач повертався до веб-сайту знову і знову.

1.5 Вибір CMS-системи

Досить великий відсоток інтернет-магазинів побудований саме на CMS-системах.

В наш час спектр веб-інструментів для розробки інтернет-магазину доволі великий. Серед CMS-систем в першу чергу краще використовувати

відкриті системи, тому що вони не складні в опануванні та не потребують великих затрат по проекту.

Список популярних CMS-систем [4]

1. Shopify – це спеціалізована платформа електронної комерції, яка виділяється серед ринку розробників інтернет-магазинів як лідер галузі завдяки широкому набору функцій електронної комерції та іншим. Він також поставляється з багатьма розширеними маркетинговими функціями, включаючи систему подарункових карток, багатоканальні продажі, інтеграцію із соціальними мережами, а також створення блогів та мобільних додатків;

2. WordPress – є однією з найпопулярніших систем, яка використовується від створення звичайних блогів до реалізації складних ресурсів новин, доволі легка у розумінні і використанні;

3. Joomla – це спеціалізована платформа, в якій присутні усі необхідні інструменти для розробки, для неї є досить багато сторонніх пакетів для завантаження і оптимізації в роботі;

4. Wix – це спеціалізована платформа, так званий конструктор. В цій платформі навички програмування зайві, тому що за користувача вже зроблений динамічний інтерфейс взаємодії з контентом, тож усе наповнення робиться досить швидко.

5. 1С Бітрікс – це спеціалізована платформа, зараз є однією з найпопулярніших CMS-систем, необхідна для створення вебзастосунків у вигляді магазинів. Ця CMS- система є платною.

6. Drupal – це спеціалізована платформа, за допомогою якої можна створити майже будь-який інтерфейс, але для цього потрібні певні знання у сфері програмування;

7. DataLife Engine – це спеціалізована платформа, яка потрібна для будування застосунків, в яких досить великий обсяг інформації.

8. OpenCart – це спеціалізована платформа, в якій є досить багато вбудованих речей для веб-сайту, а саме: відгуки до продуктів, форми для логіну та реєстрації користувача, шаблон сторінки для відвідувача. З використанням платних, так і безкоштовних інструментів розробки до системи можна створити досить потужний веб-сайт.

9. MODX Revolution – це спеціалізована платформа на основі MODX, орієнтована на роботу з програмістом, тому що розроблена за допомогою ООП і є повністю об'єктно-орієнтованою. Також надає доступ до усього коду платформи, що полегшує розробнику життя.

Висновки до розділу 1

Розглянувши всі вищевказані пункти приходимо до висновку, що в проєкті краще використовувати Shopify для створення інформаційної платформи. Під час порівняння усіх CMS-систем, було прийнято рішення використовувати платформу «Shopify».

Shopify має величезну кількість переваг, які визначили лідерство цієї системи в ніші створення інтернет-магазинів.

Серед них:

- відкритий вихідний код, який дозволяє стороннім розробникам створювати модулі та шаблони для CMS;
- декілька каналів продажів;
- підтримує понад 100 платіжних систем;
- відмінна продуктивність швидкості сайту;
- можливість обрати тему з тисяч платних і безкоштовних шаблонів;
- великі можливості інтеграції з сервісами, в тому числі для онлайн-оплати товарів;
- проста GUI адміністративна панель, що забезпечує низький поріг входження;

- велика СНГ спільнота користувачів, які допомагають розвивати CMS: створюють нові розширення і теми, формують базу знань, відповідають на питання на форумах;
- підтримка всіх видів електронної комерції;
- велика документація щодо самодопомоги;
- корисне доповнення для розширення магазину.

До недоліків можна віднести:

- необхідність знання мови шаблонізатору Liquid. Усі теми закодовані у цьому форматі. Це ускладнює налаштування теми, якщо ви не знаєте, як кодувати Liquid або не хочете найняти розробника, який знає, як кодувати теми Shopify;
- додаткові витрати на розширені функції та програми. Shopify Базовий план поставляється тільки з мінімальними функціями, які вам знадобляться для запуску інтернет-магазину. Розширені функції, такі як звіти, аналіз шахрайства, подарункові картки та вартість доставки у реальному часі, доступні лише у тарифних планах вищого рівня;
- немає поштового хостингу. Shopify не надає вам поштовий хостинг, хоча веб-хостинг включений у всі плани Shopify. Це означає, що ви не можете розмістити адресу електронної пошти на основі домену, наприклад, info@myshopname.com.

Попри перераховані мінуси, Shopify стає все більш популярною серед розробників інтернет-магазинів. І це зрозуміло, тому що всі недоліки можна виправити за гроші або безкоштовно при наявності необхідних навичок.

На основі проведеного аналізу аналогічних інтернет-магазинів можна зробити висновок, що проаналізовані системи мають недоліки у швидкості взаємодії між користувачем.

Результатом проведеної у розділі роботи став вибір CMS-системи до розроблюваного вебзастосунку.

2 ТЕХНОЛОГІЇ ТА ПІДХОДИ ДО ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ СТВОРЕННЯ ПЛАТФОРМИ ТОРГОВИХ ОПЕРАЦІЙ ЗА ДОПОМОГОЮ SHOPIFY

Перед початком програмної реалізації необхідно обрати структуру сайту, який буде мати вигляд шаблону, та як буде відбуватися взаємодія з клієнтом.

Структура сайту – це побудова сторінок сайту, категорій і підкатегорій з логічної точки зору. Структуру сайту також називають схемою, за її видом розташовуються усі сторінки на взаємопов'язані компоненти.

Такі критерії, як: простота, правильність, логічність схеми впливають на гарний відгук сайту [5].

Є два основних типи структури сайтів:

1) **Лінійна структура.** Лінійна структура сайту передбачає наявність головної сторінки «Home» та кількох внутрішніх сторінок другого рівня вкладеності, наприклад: «Home/page№1» та «Home/page№2» (див. рис. 2.1).

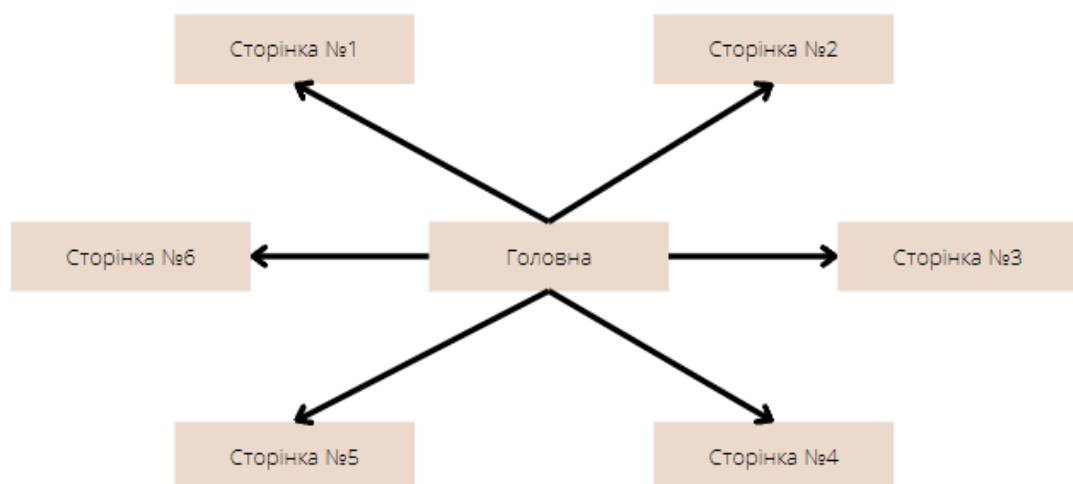


Рисунок 2.1 – приклад лінійної структури сайту

2) **Деревоподібна структура.** Деревоподібна структура сайту передбачає наявність трьох і більше рівнів вкладеності, наприклад: «Home/collections/collection№1» та «Home/products/product№1» (див. рис. 2.2).

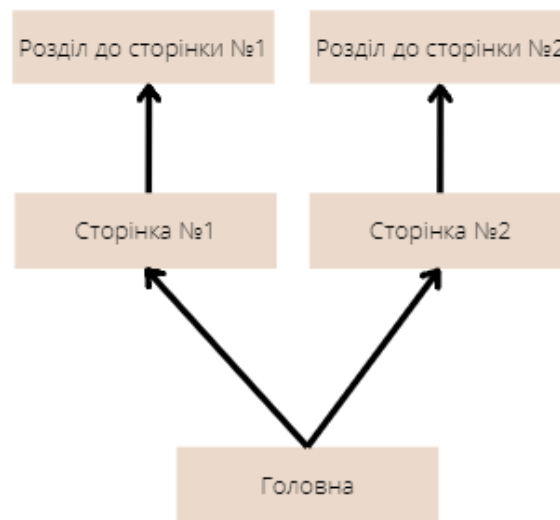


Рисунок 2.2 – приклад деревоподібної структури сайту

У результаті порівняння структур, була обрана деревоподібна, адже на веб-сайті будуть вкладені підрозділи всередині деяких сторінок.

2.1 Вимоги системи Google до структури сайту

Пошукові системи формують спеціальні вимоги до структури веб-сайту [6].

Пошукова система Google має в наявності наступні вимоги до веб-сайтів:

- створювати доволі логічну та зрозумілу для людини структуру веб-сайту;
- ставити дефіс («-»), а не нижній прочерк («_»);
- використовувати звичайні слова замість ідентифікаторів.

2.2 Створення схеми структури сайту

В цьому підрозділі потрібно зрозуміти та описати яким чином буде виглядати весь веб-сайт у вигляді схеми. Вся система необов'язково повинна містити багато сторінок, вона може складатися і з однієї, все залежить від складності проекту.

Список основних сторінок веб-сайту:

Homepage – звідси відвідувач потрапляє за посиланнями на запис, який його цікавить або в розділ ресурсу.

Contacts – важливий елемент будь-якого проекту. Його наявність та оформлення впливає на зв'язок компанії з клієнтами. При наявності безлічі каналів для зв'язку слід вибрати лише 1-2 найбільш пріоритетних і розмістити їх в шапці. А інші можна вказати в розділі "Контакти". Пошукові роботи завжди аналізують його, тому рекомендується розміщувати там якомога більше інформації по комунікації: соціальні мережі, менеджери, фізичну адресу (вставити карту), міські телефони, реквізити компанії.

При наявності декількох поштових скриньок і телефонів необхідно їх підписати і позначити, який за що відповідає. Якщо оператор один, то мобільний номер можна також підписати ім'ям. Необхідно вказати час роботи по днях тижня і час прийняття дзвінків (ці відомості можна повторювати в шапці).

About Us – більшість ресурсів помилково вказують тут інформацію про проект, особисті успіхи блогера або досягнення організації. Насправді цей розділ повинен містити приклади того, що ви можете дати клієнту + гарантії (кейси, досягнення, дипломи та ін.).

FAQS – розділ, який часто недооцінюють. Він дозволяє розвантажити роботу оператора, потрібно лише розмістити відповіді на найпоширеніші питання.

Blog – відмінне джерело залучення клієнтів з пошуку.

404 – сторінка, що не існує, на неї можна потрапити при виникненні помилки на сайті. Її відсутність або неправильне використання призводить до того, що користувачі йдуть з ресурсу.

Ознайомившись з основними сторінками веб-сайту, була побудована схема структури сайту (див. рис. 2.3):

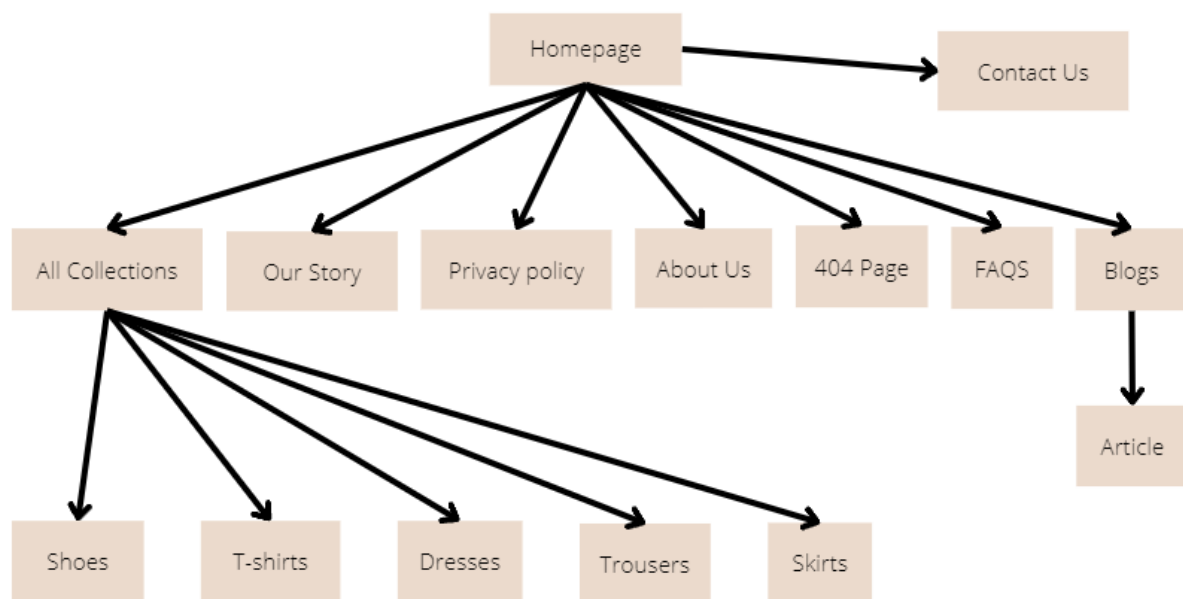


Рисунок 2.3 – схема структури сайту

При створенні схеми структури сайту було використано найпопулярніші та основні сторінки інтернет-магазину (потрібно пам'ятати, що веб-сайт не повинен буде перевантажений зайвою інформацією).

Висновки до розділу 2

У ході написання другого розділу ознайомився з поняттям структури сайту, вимогами пошукових систем до структури в цілому. У результаті побудував власну деревоподібну схему структури сайту, яка буде використовуватися для веб-сайту.

3 ТЕХНІЧНЕ НАЛАШТУВАННЯ СЕРЕДОВИЩА РОЗРОБКИ ІНТЕРНЕТ-МАГАЗИНУ ТА ОПИС СТЕКУ ІНСТРУМЕНТІВ РОЗРОБКИ

3.1 Створення та налаштування теми в адмін-панелі Shopify перед початком роботи

Перед початком роботи необхідно створити свій обліковий запис на платформі Shopify. Щоб виконати цей крок – необхідно перейти за посиланням <https://www.shopify.com/partners> та в моєму випадку натиснути кнопку «Log in» (див. рис. 3.1), якщо облікового запису немає – то потрібно зареєструватися. В формі реєстрації будуть поля вводу, такі як: назва магазину, особисті дані, номер телефону, місце знаходження магазину і т.д.

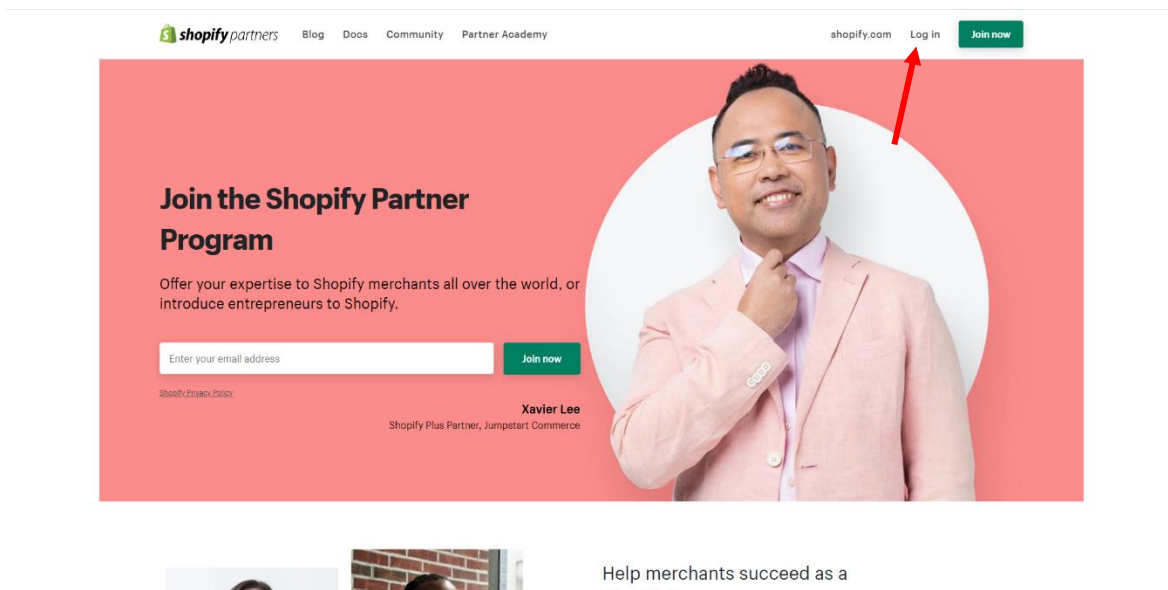


Рисунок 3.1 – авторизація на платформі Shopify

Після авторизації ми потрапимо до головної сторінки створеного магазину (див. рис. 3.2).

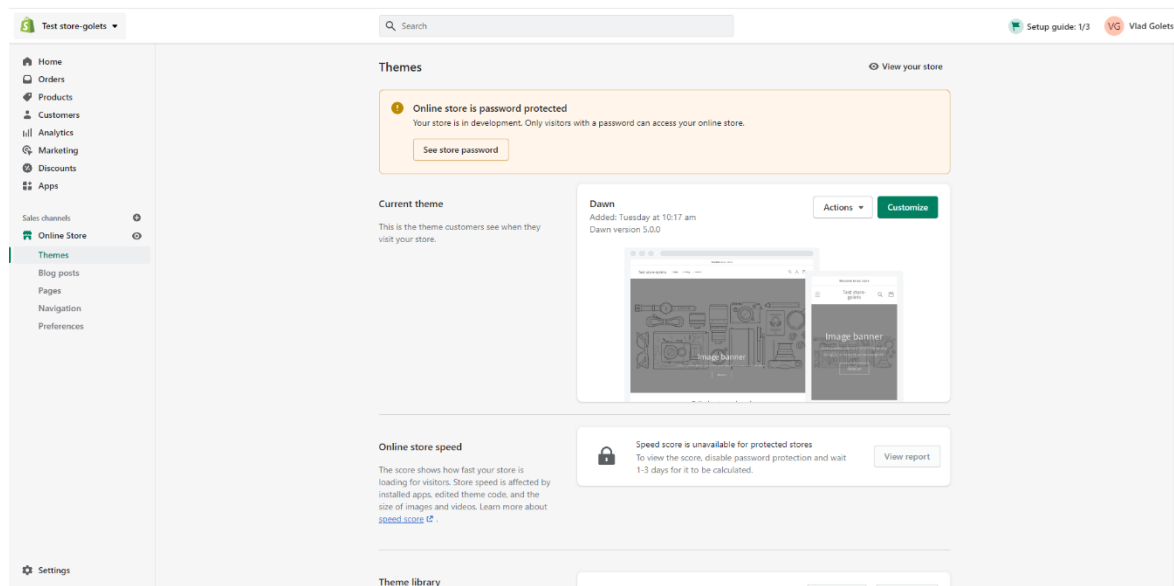


Рисунок 3.2 – головна сторінка створеного магазину

Наступним кроком буде налаштування підключення розробки власних програм від адміністратора Shopify, це необхідно для того, щоб отримати токени доступу до API та зв'язати зручне середовище розробки для розробника (наприклад: IDE PHPStrom) з самим магазином куди будуть потрапляти зміни після команди “deploy”. Для цього перейдемо в розділ «Apps» (див. рис. 3.3). Далі переходимо до розділу «Develop Apps» (див. рис. 3.4). Натискаємо на кнопку «Create an app» (див. рис. 3.5). У модальному вікні вводим назву нашого застосунку, це може бути будь-яке ім'я, бо знадобиться нам цей застосунок тільки для взаємодії розробника і платформи Shopify, я в прикладі дам назву «Credential tokens» та натискаємо кнопку «Create app» (див. рис. 3.6). Переходимо до вкладки «Configure admin API scopes» (див. рис. 3.7). Далі потрібно обрати усі чекбокси у вікні для інтеграції admin API та натиснути кнопку «Save» (див. рис. 3.8). Після цього у правому верхньому куту з'явиться кнопка «Install App», натискаємо її (див. рис. 3.9). Після цих кроків ми отримуємо Admin Access API Token, він нам знадобиться на етапі розробки пізніше (див. рис. 3.10).

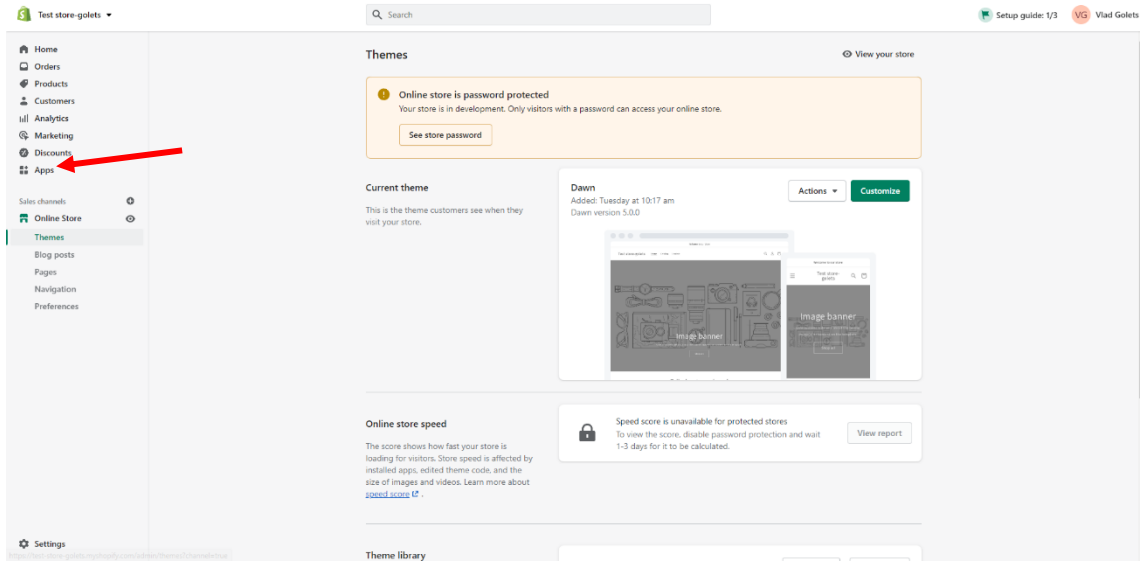


Рисунок 3.3 – перехід до розділу «Apps»

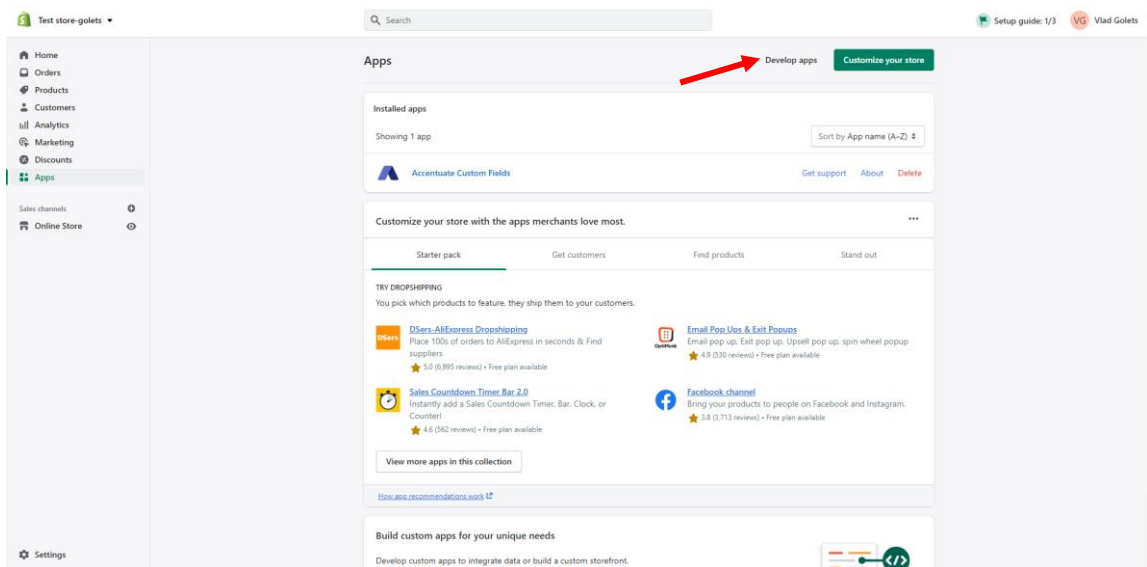


Рисунок 3.4 – перехід до розділу «Develop Apps»

Кафедра інтелектуальних інформаційних систем
Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify

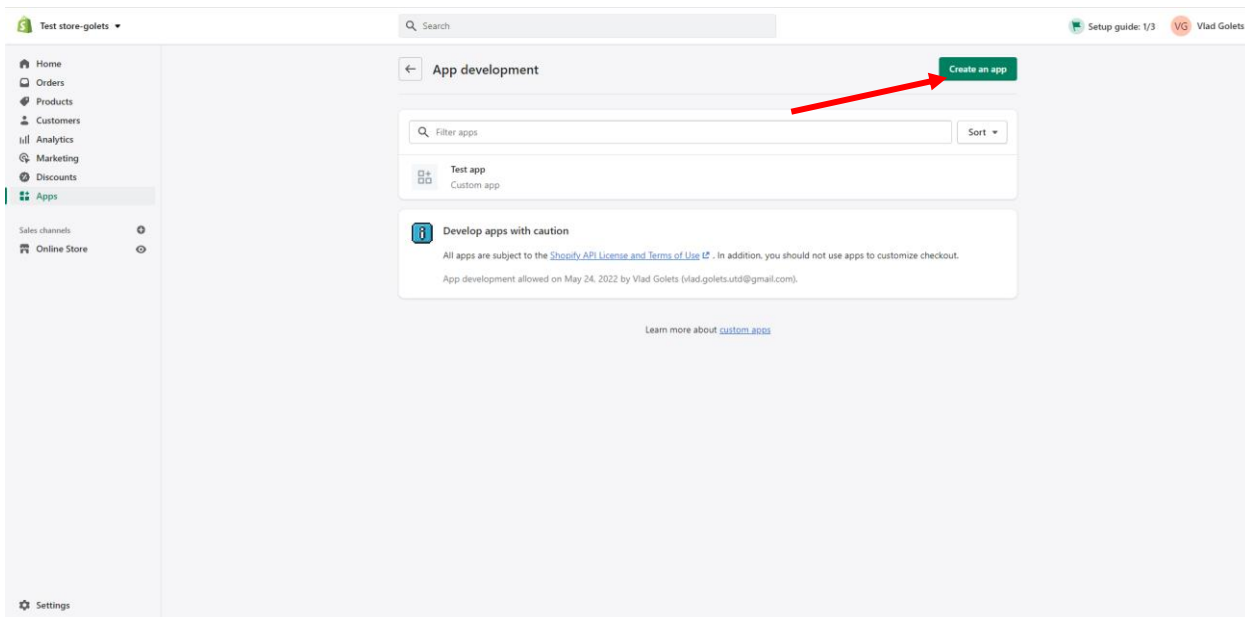


Рисунок 3.5 – перехід до розділу «Create an app»

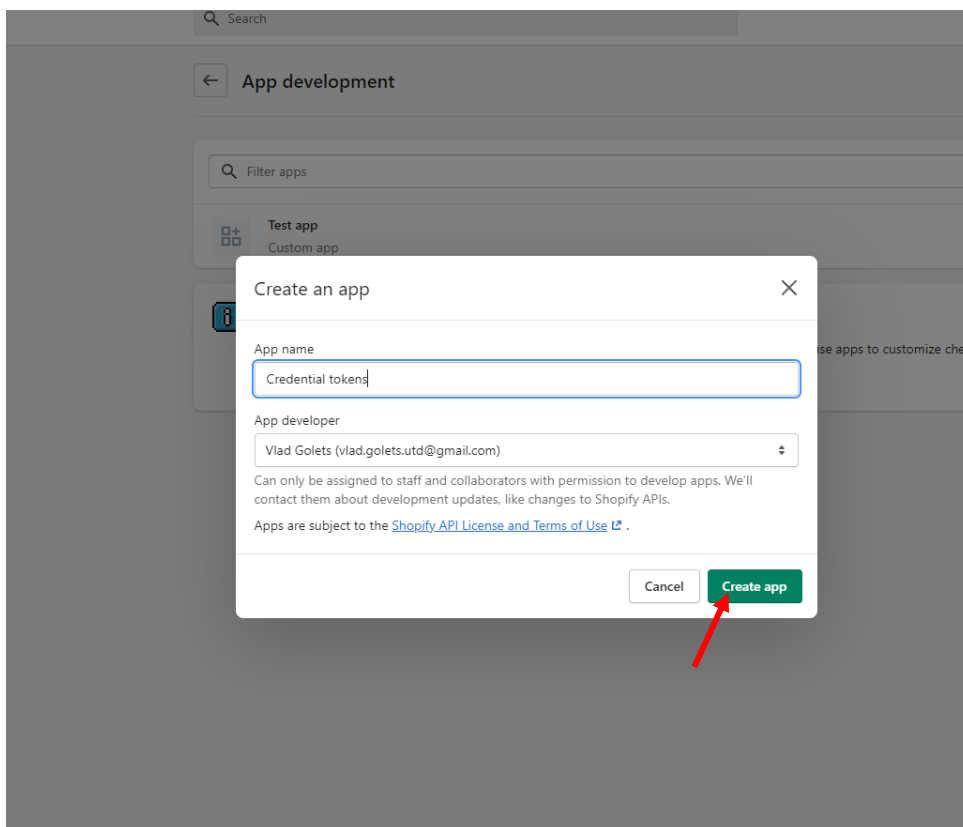


Рисунок 3.6 – приклад назви застосунку та його створення

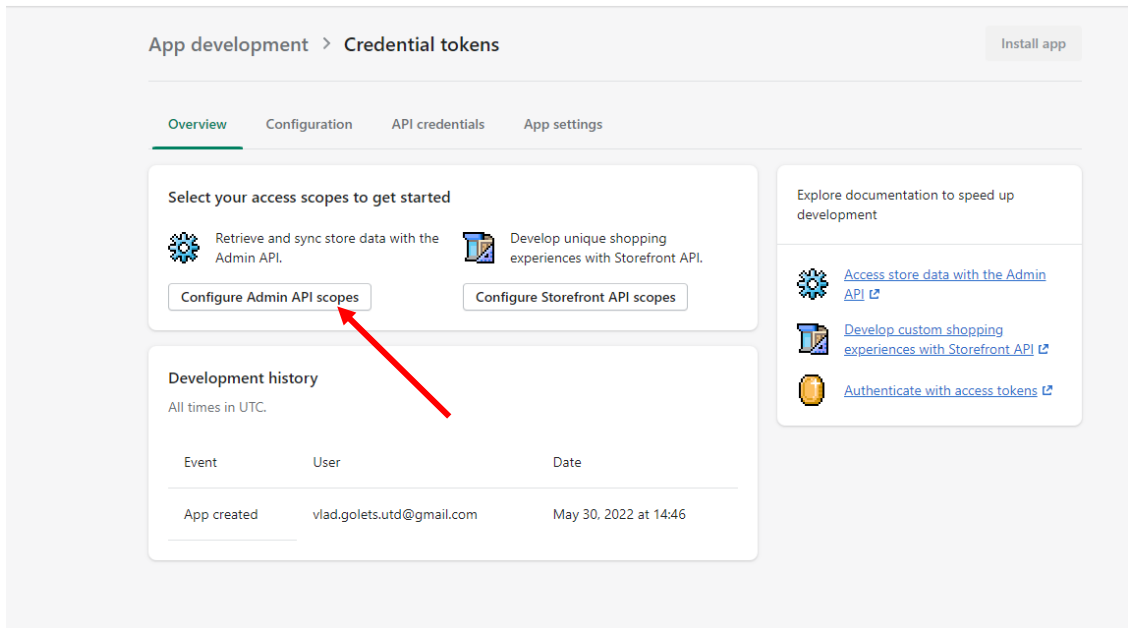


Рисунок 3.7 – перехід до вкладки «Configure admin api scopes»

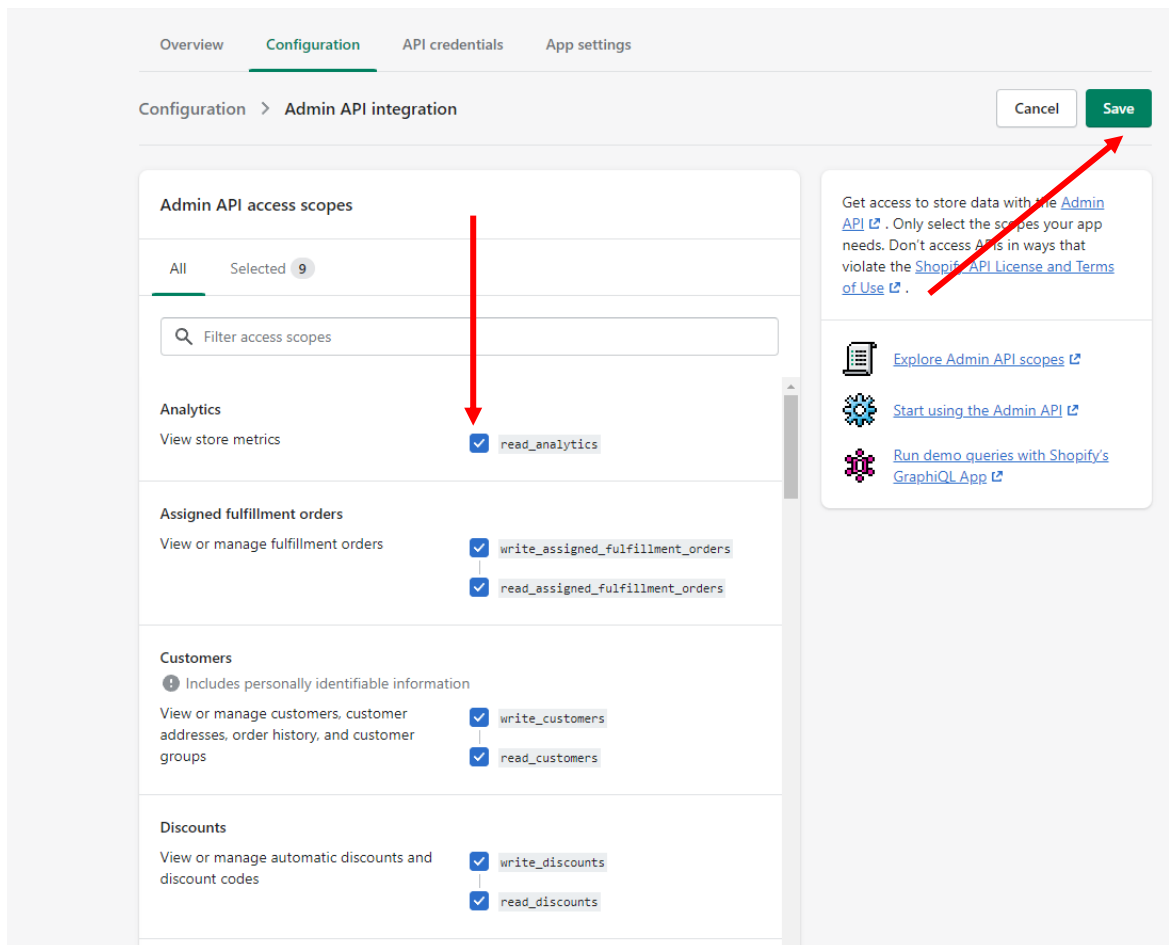


Рисунок 3.8 – перехід до розділу «Create an app»

Кафедра інтелектуальних інформаційних систем
Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify

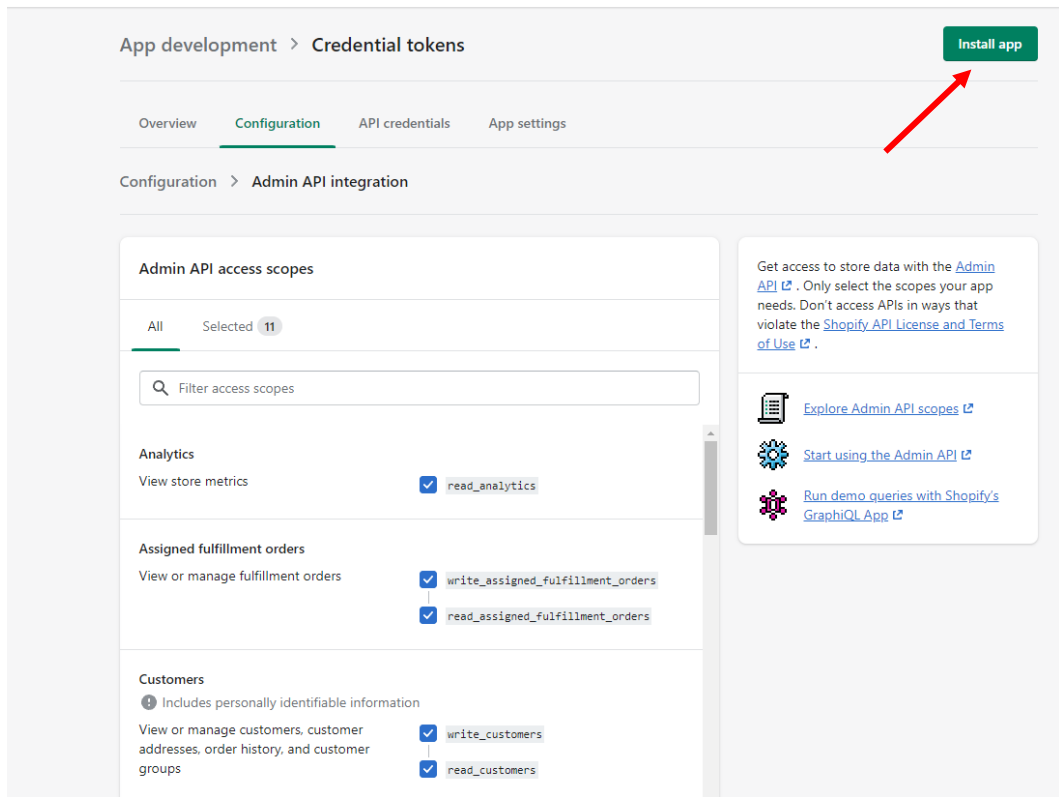


Рисунок 3.9 – інсталяція застосунку

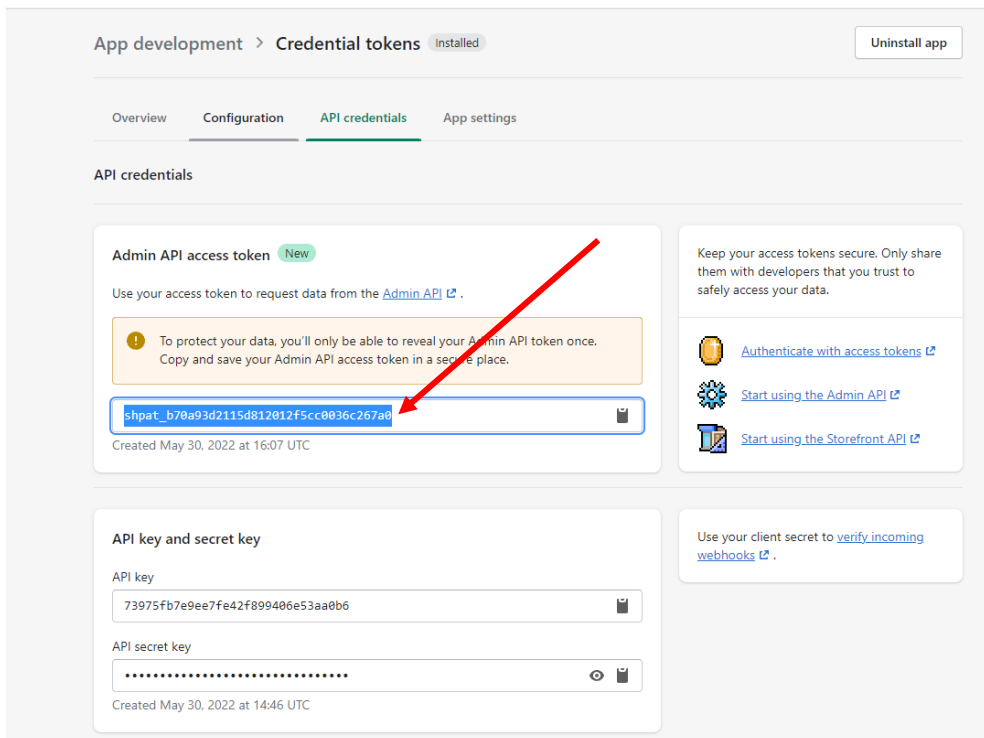


Рисунок 3.10 – отримання ключу до адмін-панелі

Після цього нам необхідно заповнити магазин продуктами та інформацією, які ми будемо надавати покупцям. Можна заповнювати продукти вручну, де будуть фотографії продукту, його ціна, знижка, варіанти, опції та багато іншого, але ми для прикладу можемо скористатися іншим способом, щоб в даному випадку пришвидшити процес. В інтернеті є досить багато файлів у .csv форматі, які можна завантажити в адмін-панель, тобто нам не треба буде вручну заповнювати інформацію для товару, але ж зрозуміло що для створення свого бренду та одягу необхідно буде завантажувати власний товар. Щоб завантажити .csv файл з продуктами перейдемо до вкладки «Products» (див. рис. 3.11). Далі натискаємо кнопку «Import» та завантажуюмо потрібний файл з інформацією о продуктах та натискаємо «Upload and continue» (див. рис. 3.12). Після цього в нашій панелі з'являться завантажені продукти з файлу (див. рис. 3.13).

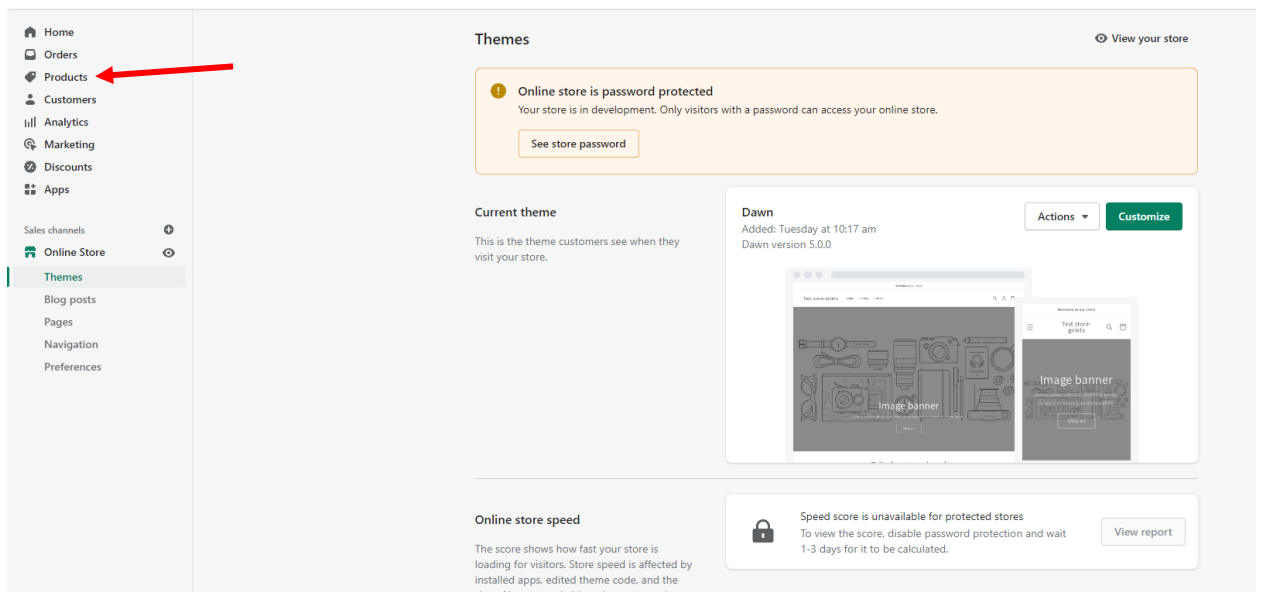


Рисунок 3.11 – перехід до вкладки «Products»

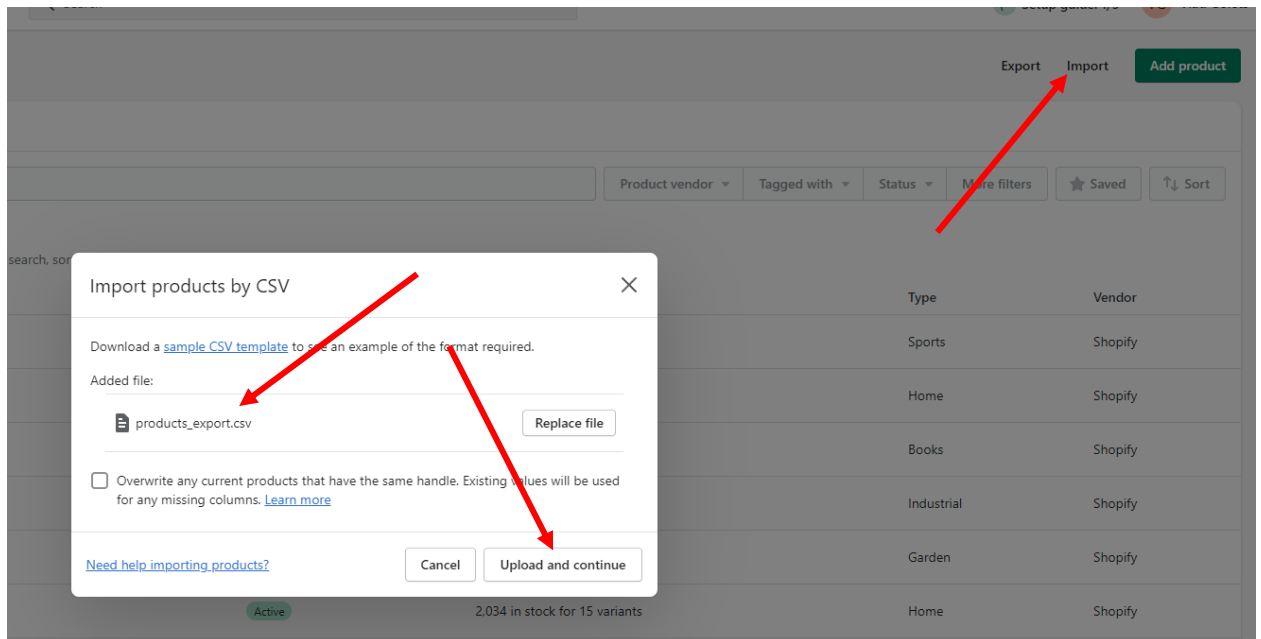


Рисунок 3.12 – імпорт .csv файлу з продуктами

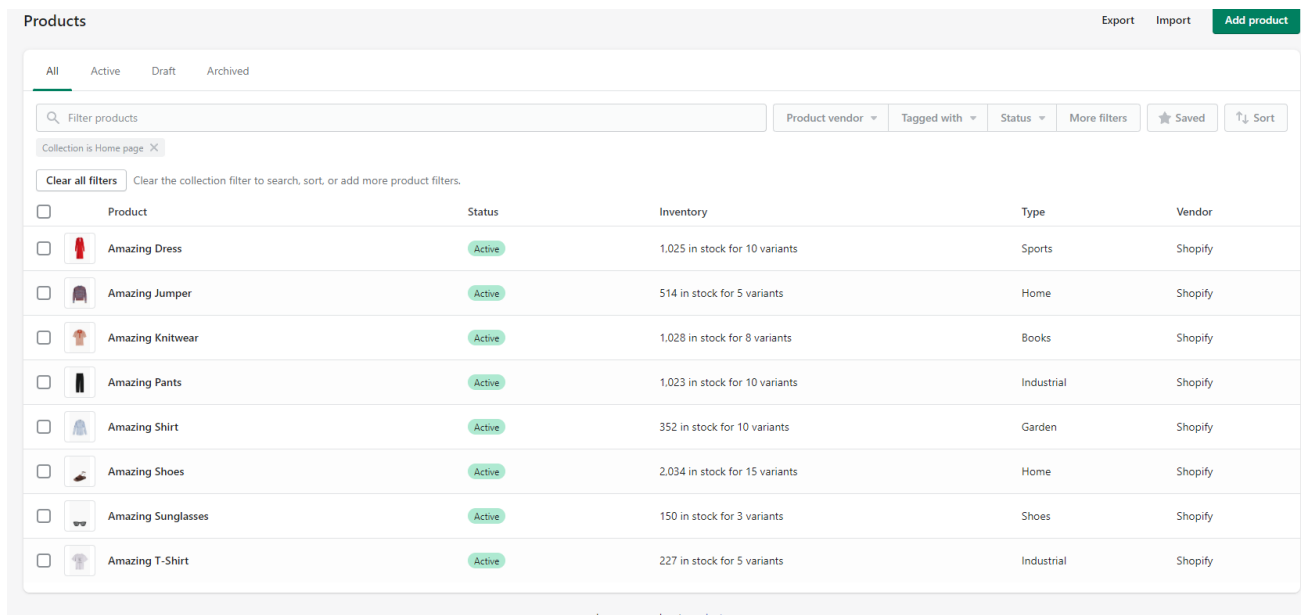


Рисунок 3.13 – вигляд завантажених продуктів

3.2 Створення локального середовища для ініціалізації проекту та написання базового коду для швидкого старту роботи

Для того, щоб збирати увесь проект та виконувати різні команди, такі як: deploy, start, build, watch, download та ін. потрібен збірник для усього проекту.

Я буду використовувати збірник для проекту «Themelify», він має в собі структуру створених компонентів, але з пустим вмістом, тож потім потрібно буде реалізувати усю логічну структуру відповідно до шаблону, збірник знаходиться в відкритому доступі репозиторію github за посиланням <https://github.com/the-utd/Themelify-Theme>.

Перш за все переходимо до необхідної директорії на комп'ютері та за допомогою git bash клонуємо репозиторій на комп'ютер (див. рис. 3.14-3.15).

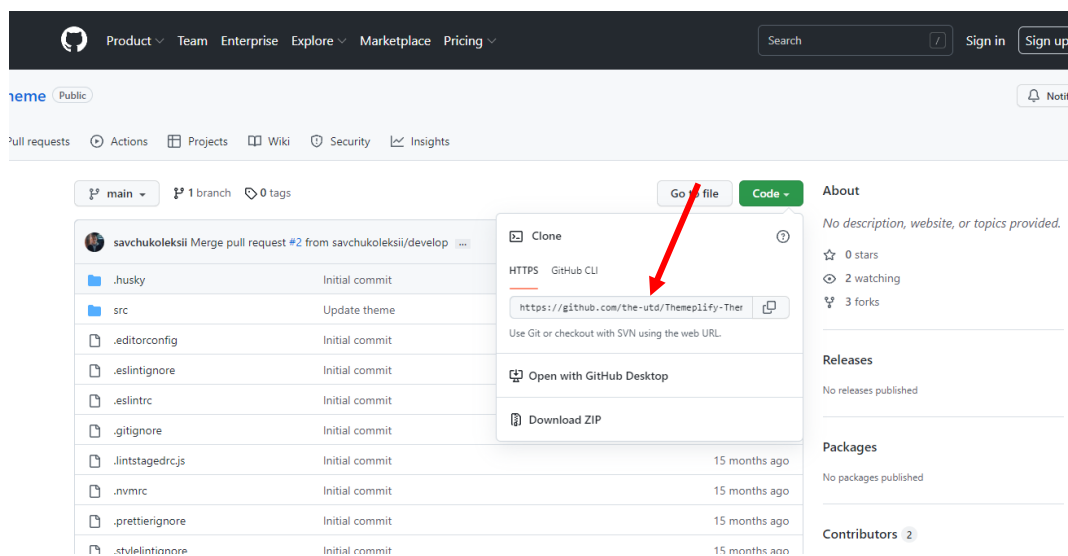


Рисунок 3.14 – клонування збірника для проекту

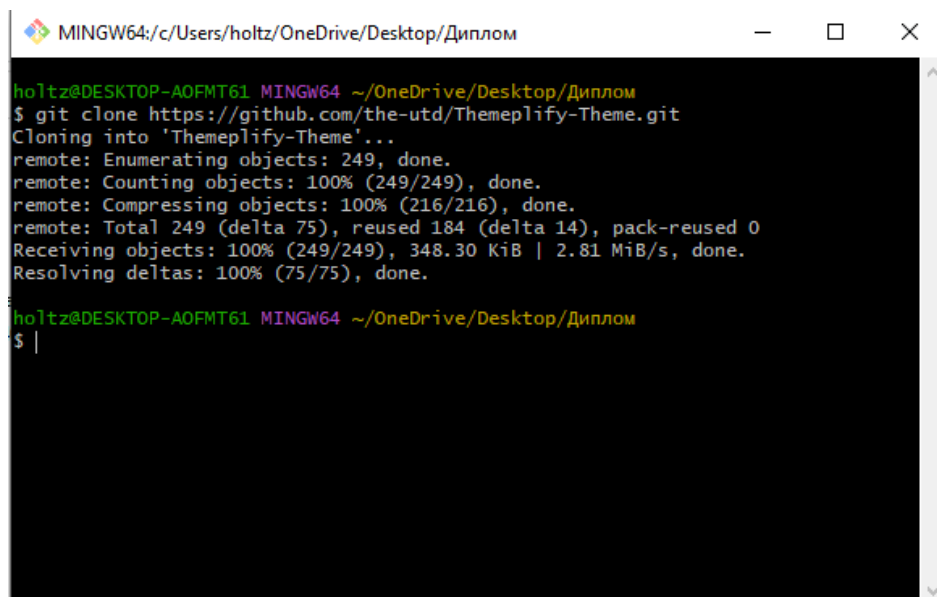


Рисунок 3.15 – клонування збірника для проекту

Після цього розгортаємо проект в IDE (див. рис. 3.16).

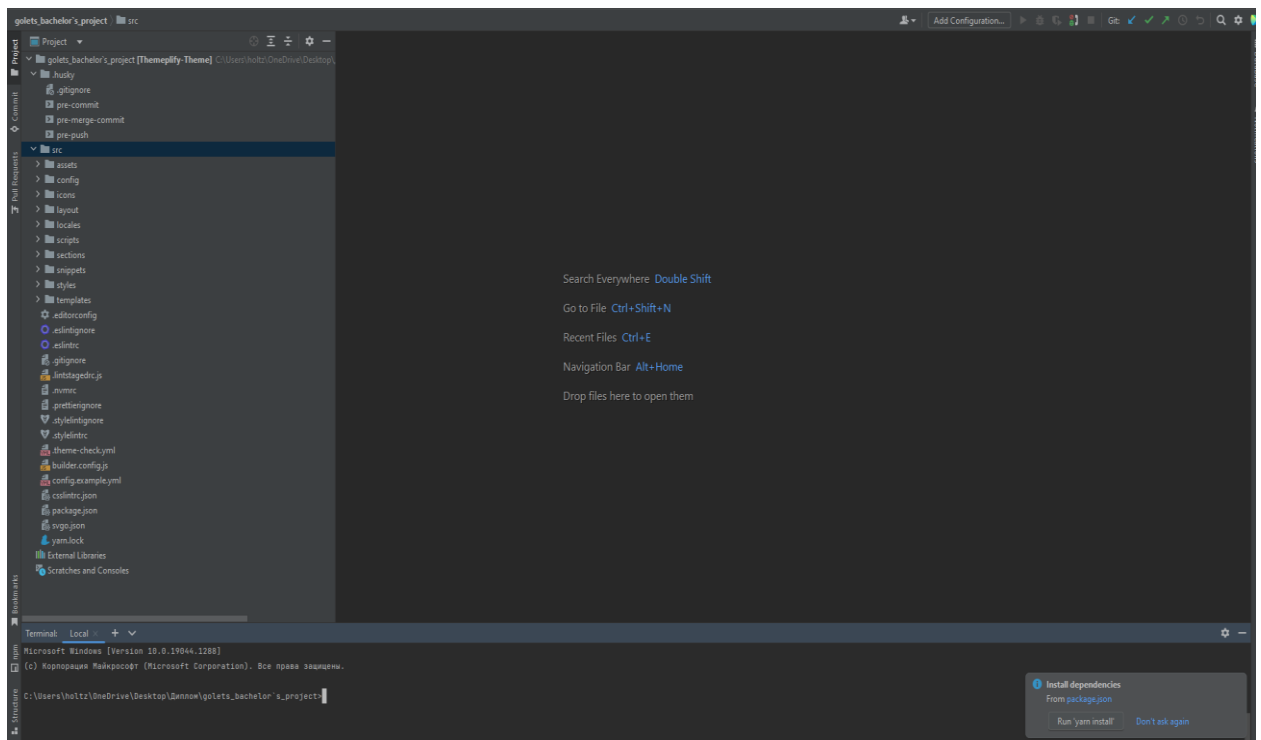


Рисунок 3.16 – розгортання проекту

Далі нам необхідно створити в адмін-панелі Shopify тему, де можна буде переглядати зовнішній вигляд інтерфейсу застосунку при змінах у кодї та вести розробку в режимі реального часу без перезавантаження сторінки та марнування часу. Для цього знову перейдемо до репозиторію зі збірником у github, але зараз завантажимо його у .zip форматі, потрібен саме .zip, тому що до адмін-панелі Shopify в тему можна завантажити тільки такий формат (див. рис. 3.17).

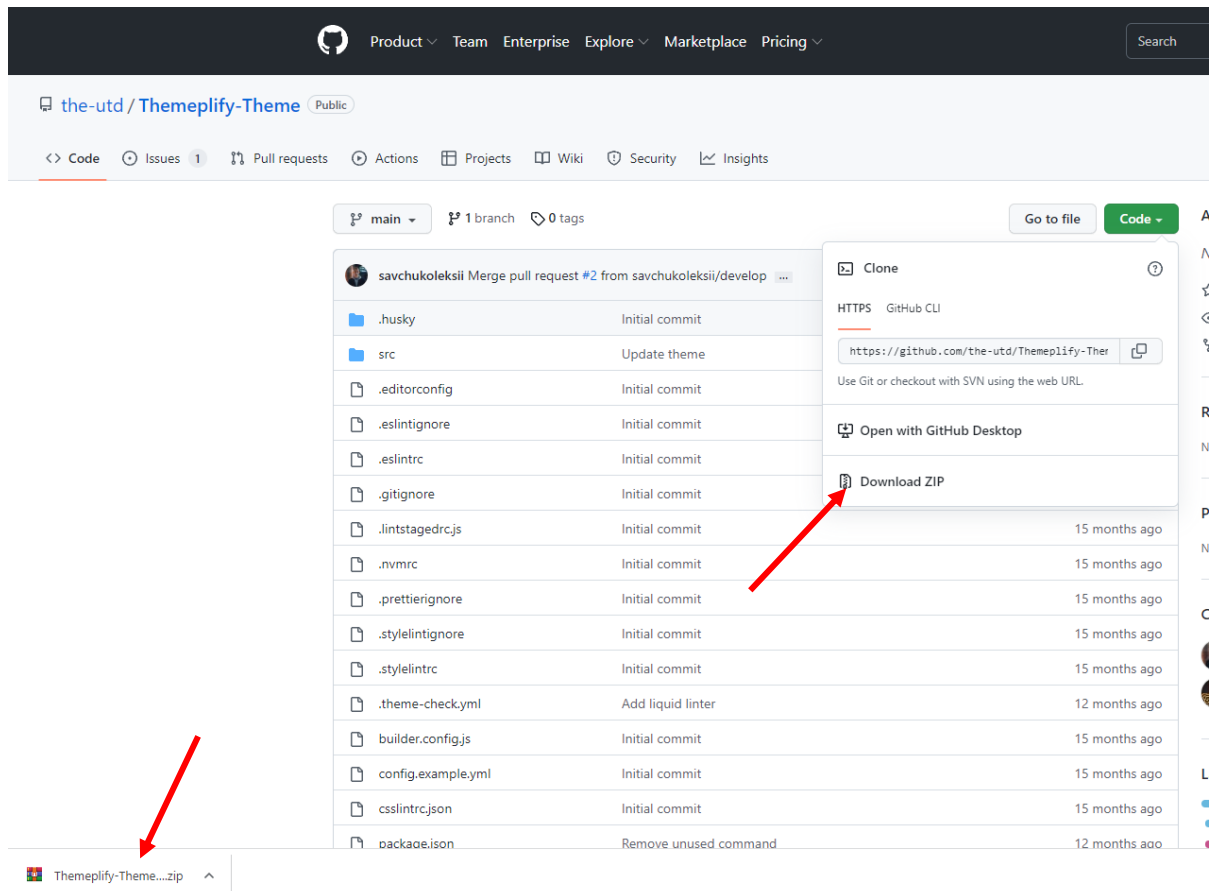


Рисунок 3.17 – завантаження збірки у .zip форматі на комп'ютер

Після цього треба перейти до адмін-панелі Shopify, щоб завантажити до теми цей .zip архів з усіма порожніми шаблонами та компонентами проекту. Для цього треба натиснути «Add theme» - «Upload ZIP file», у модальному вікні обрати zip-файл та натиснути «Upload File» (див. рис. 3.18-3.19).

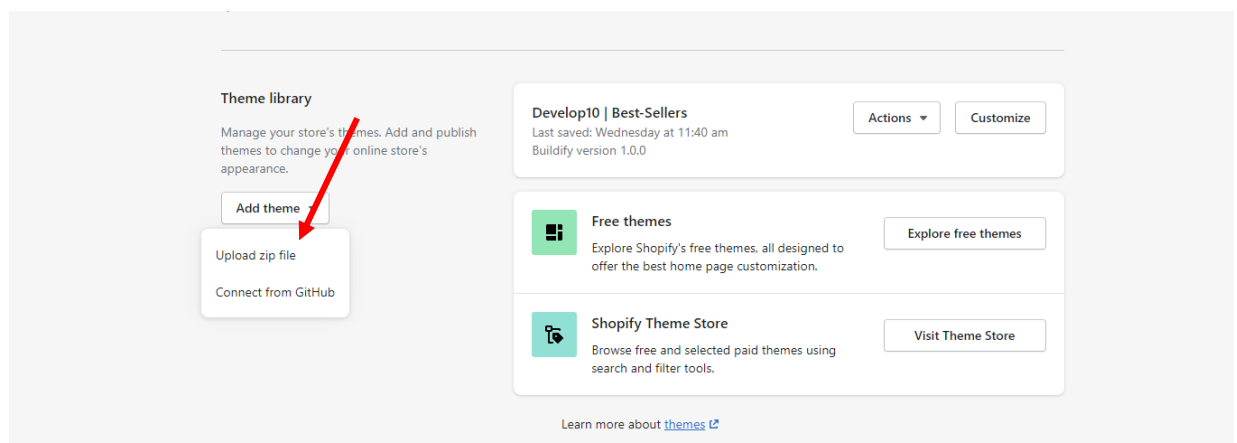


Рисунок 3.18 – завантаження збірки у .zip форматі до адмін-панелі

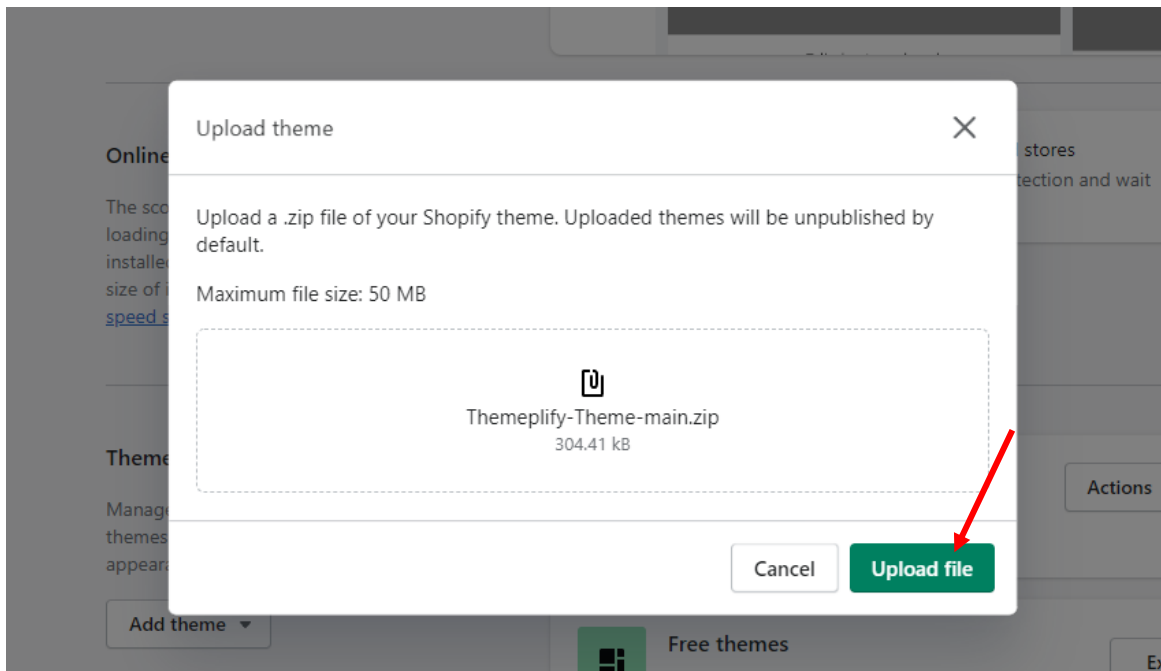


Рисунок 3.19 – завантаження збірки у .zip форматі до адмін-панелі

Після виконаних дій, в адмін-панелі з'явиться тема з назвою нашого архіву, вона буде слугувати основною темою для розробки усього проекту (див. рис. 3.20), ім'я теми можна замінити, натиснувши на кнопку «Actions» - «Rename» (див. рис. 3.21).

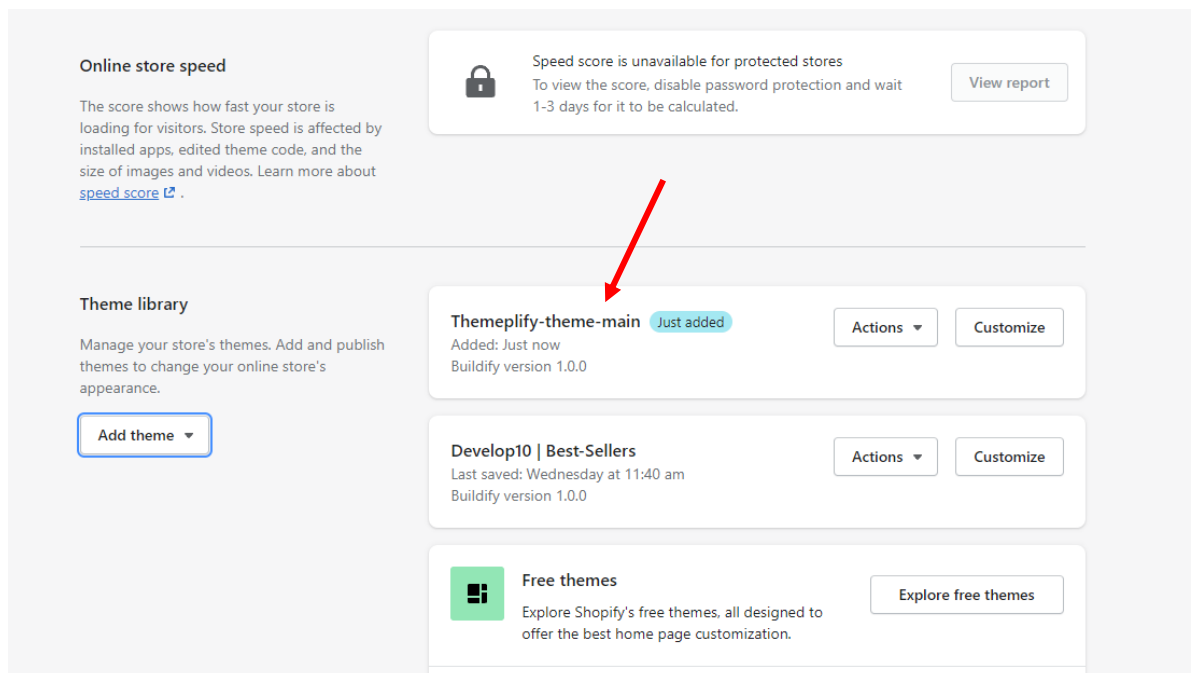


Рисунок 3.20 – створення теми для розробки

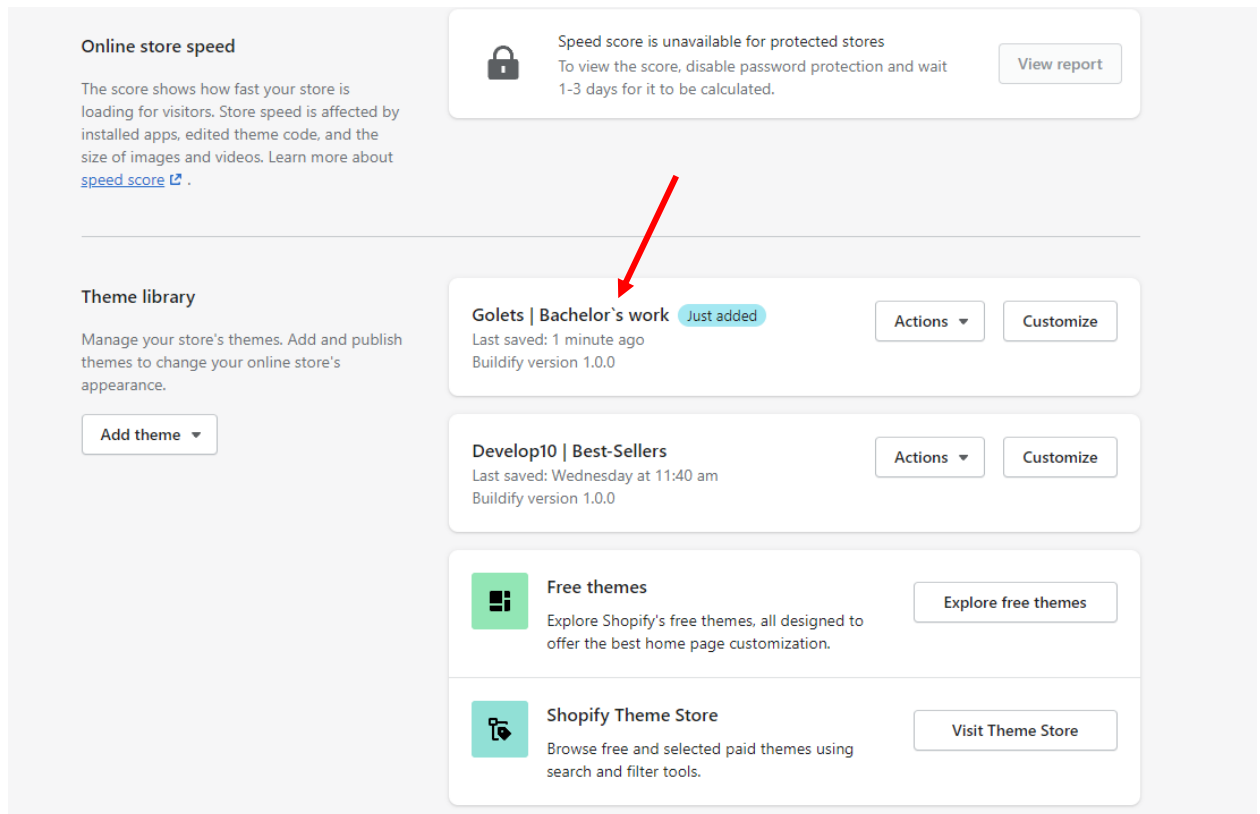


Рисунок 3.21 – зміна імені головної теми

Наступним кроком буде налаштування зв'язування між локальним середовищем розробки та самим магазином на платформі Shopify. В збірнику є файл-приклад «`config.example.yml`», вміст якого можна скопіювати та створити такий же файл, але без слова «`example`», тобто він буде мати назву «`config.yml`» (див. рис. 3.22).

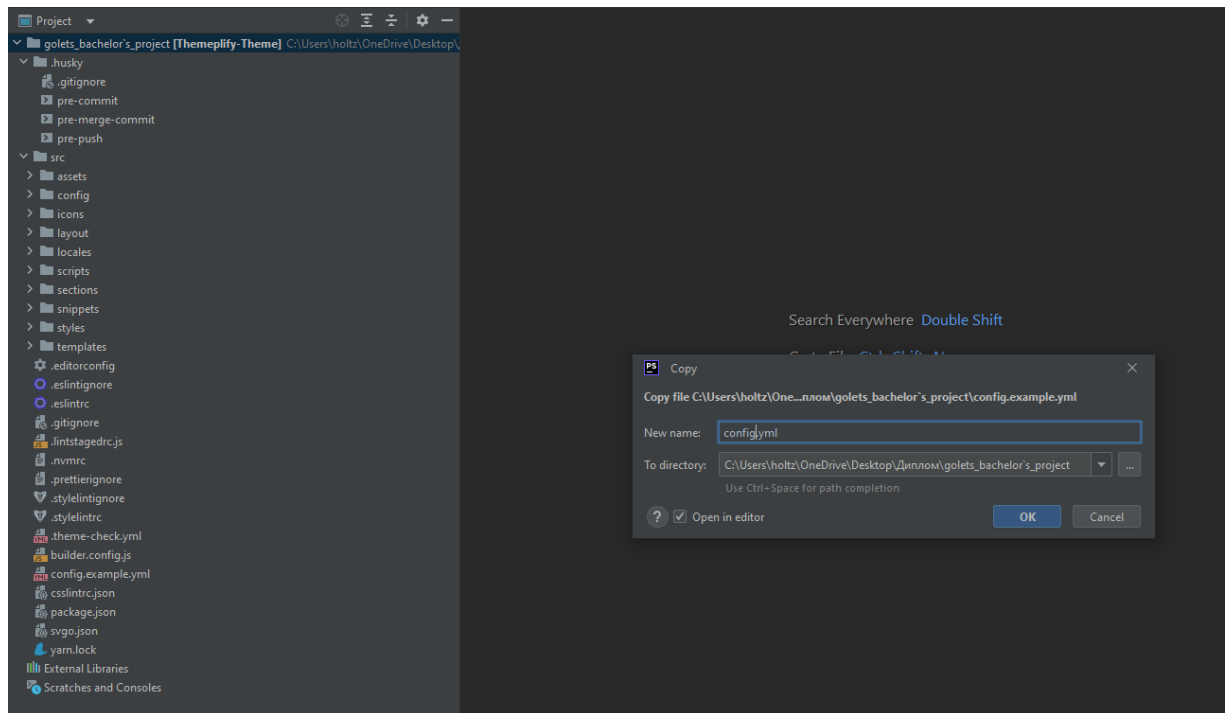


Рисунок 3.22 – створення файлу-конфігурації для зв’язування локального середовища розробки та самим магазином на платформі Shopify

Перейшовши до файлу-конфігурації, побачимо порожні поля, які потрібно буде заповнити, в нашому випадку нам знадобляться перші 4 поля: (див. рис. 3.23).

1. `namespace` – головне поле, яке відповідає за назву простору імен (всі поля, що знаходяться нижче відносяться саме до цього простору імен «namespace», ім’я є довільним у написанні);
2. `password` – поле для вводу API токена до Shopify адмін-панелі, саме раніше ми вже отримали цей токен в адмін-панелі Shopify, тому залишиться його тільки ввести (див. рис. 3.10);
3. `theme_id` – поле для вводу ідентифікатора необхідної теми, оскільки в адмін-панелі може бути створено багато тем в залежності від тарифного плану магазину, щоб дізнатися значення `theme_id` – потрібно перейти до панелі Shopify та натиснути по необхідній темі «Actions» - «Edit Code», цифри після останнього слесу – те що необхідно (див. рис. 3.24);

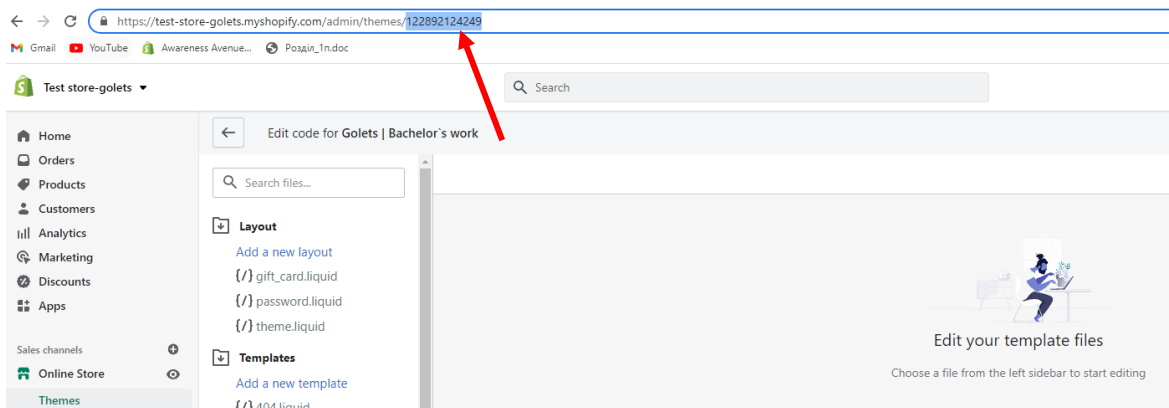


Рисунок 3.24 – значення theme_id

4. store – поле для вводу домену сайту, тобто його можна знайти в самій адмін-панелі в пошуковій стрічці сайту.

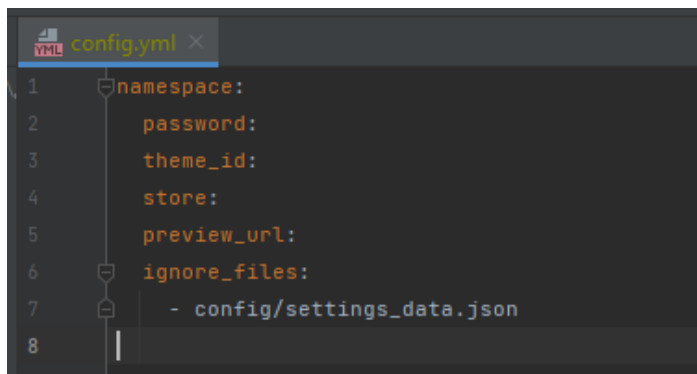


Рисунок 3.23 – поля у файлі-конфігурації

Після того, коли дізналися за що відповідає кожне поле – необхідно їх заповнити (див. рис. 3.25) і вже після цього можна вважати, що середовище налаштовано для запуску розробки проекту.

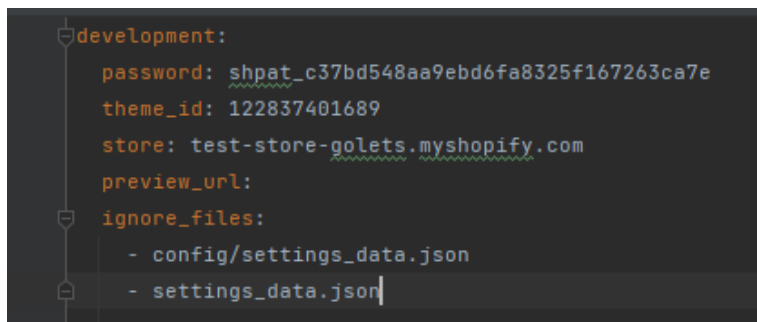
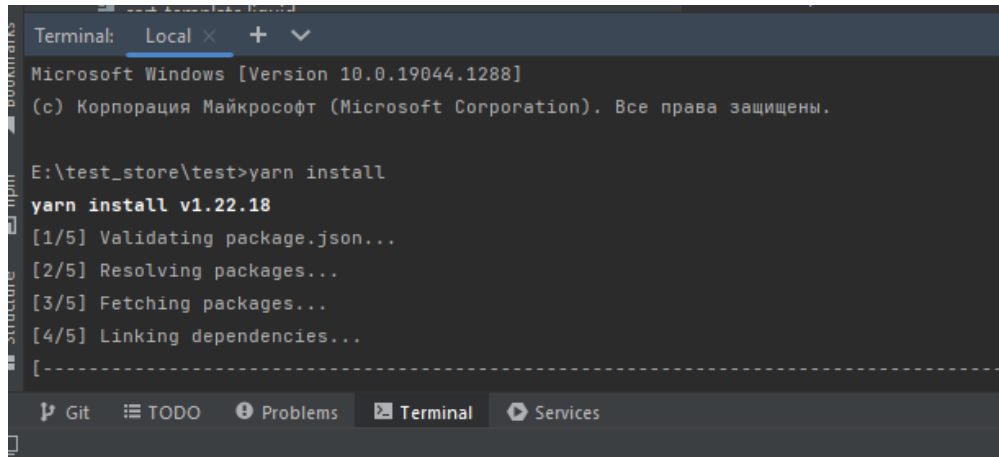


Рисунок 3.25 – приклад заповнених полів у файлі-конфігурації

Далі потрібно виконати інсталяцію пакетів за допомогою команди «yarn install» (див. рис. 3.26).

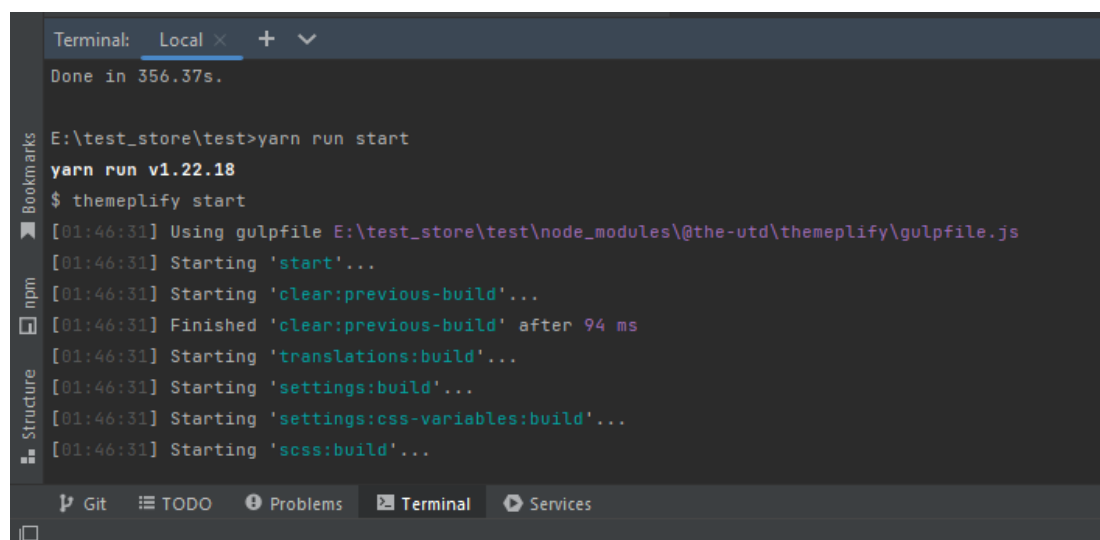


```
Terminal: Local x + v
Microsoft Windows [Version 10.0.19044.1288]
(c) Корпорація Майкрософт (Microsoft Corporation). Все права захищені.

E:\test_store\test>yarn install
yarn install v1.22.18
[1/5] Validating package.json...
[2/5] Resolving packages...
[3/5] Fetching packages...
[4/5] Linking dependencies...
[-----]
Git  TODO  Problems  Terminal  Services
```

Рисунок 3.26 – інсталяція пакетів

Коли модулі закінчать інсталяцію, потрібно запустити проект за допомогою команди «yarn run start» (див. рис. 3.27) та перевірити, що збірник проекту працює правильно. Щоб це перевірити у проекті на DOM-вузол <body> навішаний рожевий колір фону (див. рис. 3.28), тобто якщо при запуску проекту ми побачимо у браузері рожевий колір фону, то це буде означати, що проект скомпільований правильно.



```
Terminal: Local x + v
Done in 356.37s.

E:\test_store\test>yarn run start
yarn run v1.22.18
$ themeplify start
[01:46:31] Using gulpfile E:\test_store\test\node_modules\@the-utd\themeplify\gulpfile.js
[01:46:31] Starting 'start'...
[01:46:31] Starting 'clear:previous-build'...
[01:46:31] Finished 'clear:previous-build' after 94 ms
[01:46:31] Starting 'translations:build'...
[01:46:31] Starting 'settings:build'...
[01:46:31] Starting 'settings:css-variables:build'...
[01:46:31] Starting 'scss:build'...

Git  TODO  Problems  Terminal  Services
```

Рисунок 3.27 – запуск проекту


```
1  @import "variables";
2  @import "utd-reset";
3  @import "fonts";
4  @import "typography";
5
6  * {
7    outline: none;
8    box-sizing: border-box;
9  }
10
11  html,
12  body {
13    background-color: pink;
14  }
```

Рисунок 3.28 – запуск проекту

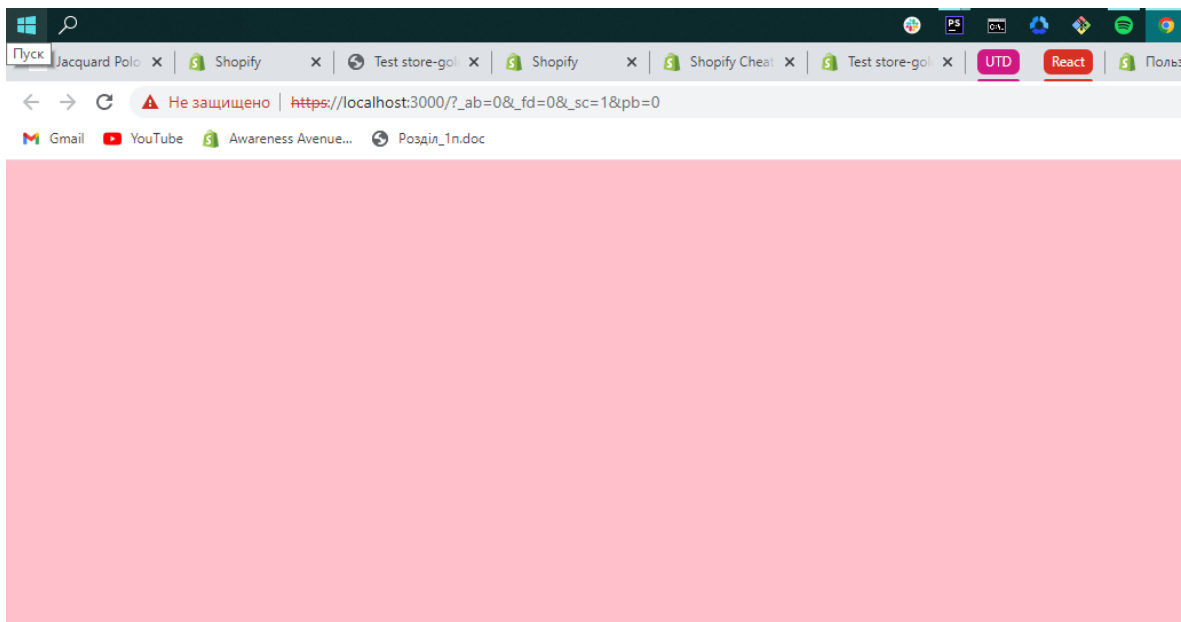


Рисунок 3.29 – перевірка збірки проекту

3.3 Вибір та опис стеку інструментів для розробки веб-платформи

Перед тим як почати створювати сторінки власного шаблону, потрібно вирішити які програмні середовища та мови програмування будуть застосовані.

Основною мовою при програмуванні інтернет-магазину на платформі Shopify служить мова Liquid.

Liquid – це мова шаблонів, створена Shopify і написана на Ruby. Наразі він доступний як проект з відкритим вихідним кодом на GitHub, який використовується у різних розробках багатьма компаніями – виробниками програмного забезпечення. Liquid є основою всіх тем Shopify та використовується для завантаження динамічного контенту на сторінки інтернет-магазинів. Дизайнери та розробники сайтів можуть використовувати мову шаблонів для створення веб-сторінок, що об'єднують статичний контент, який є єдиним для декількох сторінок, і динамічний контент, який змінюється при переході з однієї сторінки на іншу. Мова шаблонів дозволяє повторно використовувати статичні елементи, що визначають макет веб-сторінки, доки сторінка динамічно заповнюється даними магазину Shopify. Статичні елементи пишуться HTML, а динамічні елементи Liquid. Елементи Liquid у файлі діють як заповнювачі: коли код у файлі компілюється та відправляється до браузера, Liquid замінює їх даними з магазину Shopify [7].

HTML – це головна частина веб-сайту, так званий «скелет» сторінки, яку можна побачити всередині браузеру. Мова гіпертекстової розмітки містить та складається з парних та непарних тегів, за їх допомогою браузер розпізнає вигляд того чи іншого елемента на веб-сторінці. В мові найчастіше вживаються такі теги: заголовок, параграф, гіперпосилання, зображення, форма, кнопка і т.д. (див. рис. 3.30) [8].



Рисунок 3.30 – логотип HTML

CSS – набір стилів, за допомогою якого можна змінити зовнішній вигляд елементів веб-сторінки, використовується для стилізації [9].

CSS є потужною системою, яка доповнює можливості дизайну і верстки веб-сторінок.

Можна розрізнити стилі за типами, що можуть спільно існувати в одному документі, а саме стиль користувача, стиль браузера і стиль автора.

Стилі браузера – це стилі браузера за замовчуванням, які застосовуються до тегів веб-сторінки. Це оформлення можна побачити, якщо стилі не перевизначені автором або користувачем.

Стилі автора – це стилі, які вручну прописує до елементів розробник в середі розробки.

Стилі користувача – це стилі, які може включити користувач сайту через браузер в інспекторі. Цей стиль має більш високий пріоритет, ніж інші.

Зазначені типи стилів можуть застосовуватися одночасно, якщо вони не намагаються змінити вид одного елемента. У разі виникнення конфліктів спочатку будуть мати пріоритет стилі користувача, потім стилі автора і після іде стилі браузера (див. рис. 3.31).



Рисунок 3.31 – логотип CSS

JavaScript – динамічна мова програмування, яка виконується синхронно в основному потоці коду [10]. Створена для того, щоб робити веб-сторінки динамічними, мова відноситься до стандарту ECMAScript. Основними завданнями є взаємодія з браузером, асинхронний обмін даних з сервером, зміна структури DOM.

JavaScript визначають як прототипну мову програмування з нестрогою типізацією змінних. В цій мові є підтримка різних парадигм програмування (часткова-функціональна, імперативна та архітектурні властивості), тому її ще також називають мультипарадигменною (див. рис. 3.32).



Рисунок 3.32 – логотип JavaScript

SASS – це доповнення до таблиці стилів CSS, яке надає більше можливостей цій простій мові, його називають препроцесором. За допомогою SASS можливо використовувати змінні, вкладені правила, цикли, міксини, інлайновий імпорт та багато іншого, все пишеться на схожому CSS синтаксисі [11]. Тобто все пишеться за синтаксисом SASS, але до браузеру все приходиться в CSS форматі (див. рис. 3.33).



Рисунок 3.33 – логотип SASS

Можна перерахувати наступні переваги SASS над CSS:

Вкладеність. HTML має чітку вкладену і візуальну деревоподібну ієрархію. З CSS це не так. SASS робить синтаксис більш ізольованим і чистішим, через це зайві стилі не повторюються.

За «завісою» препроцесор все ще компілює його в стандартний код CSS, щоб він міг бути відображений в браузері. Це лише більш потужний спосіб написання CSS. (див. рис. 3.34).

```

CSS   Sass
nav
ul
  margin: 0
  padding: 0
  list-style: none

li
  display: inline-block

a
  display: block
  padding: 6px 12px
  text-decoration: none

CSS
nav ul {
  margin: 0;
  padding: 0;
  list-style: none;
}
nav li {
  display: inline-block;
}
nav a {
  display: block;
  padding: 6px 12px;
  text-decoration: none;
}

```

Рисунок 3.34 – вкладеності

Фрагментування. Можна створювати фрагменти SASS-файлу, які містять в собі невелику частину коду, яка в майбутньому може знадобитися в перевикористанні в різних частинах коду. За допомогою цього CSS стає більш модульним і компонентним. В цілому фрагмент – це звичайний SASS-файл, назва якого розпочинається з підкреслення, наприклад, `_reset.scss`. Підкреслення в імені SASS-компоненту повідомляє компілятор про те, що це тільки фрагмент і він не має компілюватися в CSS формат. За допомогою директиви `@import` підключаються фрагменти (див. рис. 3.35).

```

SCSS   Sass
// _reset.scss
html,
body,
ul,
ol
  margin: 0
  padding: 0

// base.scss
@import reset
body
  font: 100% Helvetica, sans-serif
  background-color: #efefef

CSS
html,
body,
ul,
ol {
  margin: 0;
  padding: 0;
}
body {
  font: 100% Helvetica, sans-serif;
  background-color: #efefef;
}

```

Рис. 3.35. Фрагментування

Змінні. Це так званий спосіб зберігання інформації, яку можна використовувати протягом написання усього проекту. Змінні SASS оголошуються з попереднім символом «\$» і записуються так само як і в CSS. За допомогою SASS, зазвичай визначають змінні для стилів: font-size, margin, padding і т.д. (див. рис. 3.36):

```

SCSS      Sass
$font-stack: Helvetica, sans-serif
$primary-color: #333

body
  font: 100% $font-stack
  color: $primary-color

CSS
body {
  font: 100% Helvetica, sans-serif;
  color: #333;
}

```

Рис. 3.36. Змінні

Міксини. У CSS властивостях є деякі речі, які постійно повторюються і їх потрібно повторювати в написанні стилів для коректної роботи. За допомогою міксин можна створювати певні декларації CSS, які потім можна перевикористати потрібну кількість разів і зекономити час при написанні. Наприклад, великого значення набувають міксини для крос-браузерних префіксів (див. рис. 3.37):

```

SCSS      Sass
=transform($property)
  -webkit-transform: $property
  -ms-transform: $property
  transform: $property
.box
+transform(rotate(30deg))

CSS
.box {
  -webkit-transform: rotate(30deg);
  -ms-transform: rotate(30deg);
  transform: rotate(30deg);
}

```

Рис. 3.37. Міксини (домішки)

Розширення / Спадкування. Використовується за допомогою директиви «@extend», щоб застосувати функцію наслідування набору властивостей CSS від одного елемента іншому, є однією з найбільш корисних

і застосовуваних директив. За допомогою цього можна тримати файли в чистоті і не перевантажувати його повторюваними стилями (див. рис. 3.38).

```

SCSS      Sass
/* This CSS will print because %message-shared is extended. */
%message-shared
  border: 1px solid #ccc;
  padding: 10px;
  color: #333;

// This CSS won't print because %equal-heights is never extended.
%equal-heights
  display: flex;
  flex-wrap: wrap;

.message
  @extend %message-shared;

.success
  @extend %message-shared;
  border-color: green;

.error
  @extend %message-shared;
  border-color: red;

.warning
  @extend %message-shared;
  border-color: yellow;

CSS
.message, .success, .error, .warning {
  border: 1px solid #ccc;
  padding: 10px;
  color: #333;
}

.success {
  border-color: green;
}

.error {
  border-color: red;
}

.warning {
  border-color: yellow;
}

```

Рисунок 3.38 – розширення / спадкування

Для цього проекту було вирішено описати стилі сайту саме за допомогою препроцесору SASS, адже це прискорює написання стилів в декілька разів.

Як тільки починається користування SASS, препроцесор запускає функцію обробки SASS файлів та перекомпілює їх до звичайного CSS формату.

VueJS – є реактивним фреймворком для створення вебзастосунків. Основною відмінністю Vue від інших реактивних фреймворків є те, що його було створено придатним для поступового впровадження. Ядро виконує функцію рівня «view» (представлення), за допомогою цього інтеграція інших бібліотек та проектів стає спрощеною. Але з іншої сторони, за допомогою Vue можна створювати складні односторінкові вебзастосунки (Single Page

Application), якщо в доповненні використовувати в проекті сучасні веб-технології та бібліотеки (див. рис. 3.39) [12].

Реактивність – це так звана парадигма у програмуванні, за допомогою якої можна пристосовуватися до змін декларативним чином. Змінні в JavaScript не реактивні за замовчуванням. Саму механізм JS Proxy забезпечує досягнення реактивності. Vue надає необхідну абстракцію до JavaScript Proxy і робить її простою. Ця реактивність використовується, щоб отримувати швидку зв'язку між даними та складними уявленнями компонентів з API.

В свою чергу Vue не являється в повній мірі фреймворком, це лише рівень представлення.



Рисунок 3.39 – логотип Vue

jQuery – це швидка, легка та багатофункціональна JavaScript бібліотека, заснована на принципі «пиши менше, роби більше» (див. рис. 3.40) [13].

Переваги, які дає нам бібліотека jQuery при її використанні для написання сценаріїв клієнтів:

- компактність коду. Дозволяє писати код компактніше ніж на чистому JavaScript, тобто за набагато меншу кількість рядків коду.
- простий та зрозумілий синтаксис. Значно спрощує написання багатьох речей, наприклад, таких як маніпулювання DOM елементами, обробка подій, додавання ефектів анімації на сторінку, AJAX запитів і т.д.

– кросбраузерність. Код, написаний на jQuery, буде гарантовано працювати у всіх основних браузерах. У той час як код, написаний на чистому JavaScript, потрібно буде однозначно перевіряти у всіх браузерах. Т.к. деякі фрагменти коду можуть не підтримуватися і їхню реалізацію для цих браузерів потрібно буде виконувати якимось по-іншому, або, наприклад, використовувати поліфіли.

– відкритий код. Бібліотека jQuery є безкоштовною як для особистих, так і для комерційних проектів.

Крім переваг, наведених вище, бібліотека jQuery має звичайно і недоліки. До недоліків jQuery можна віднести те, що вона може трохи збільшувати швидкість завантаження веб-сторінки (її стислий розмір становить близько 30КБ), а також трохи знижувати продуктивність виконання коду, ніж якщо він був написаний на чистому JavaScript.



Рисунок 3.40 – логотип jQuery

Bootstrap – найбільш популярний фреймворк HTML, CSS, JS для проектування в першу чергу адаптивного дизайну [14]. Bootstrap використовують, як великі компанії, так і звичайні фрілансери, а саме через його можливості.

Вивчивши ази Bootstrap, верстка буде здаватися полегшеною. Головне, що фреймворк включає в себе властивості кросбраузерності і адаптивності, і саме про ці речі потрібно думати при створенні веб-сайту. Ця бібліотека дозволяє створити HTML-шаблон навіть людині, яка раніше дуже мало

займалася створенням та стилізацією компонентів і не зовсім знайома з властивостями CSS.

Основною сферою застосування є створення веб-компонентів на веб-сайті. Бібліотека надає велику кількість вже готових компонентів та класів для розробки.

До переваг Bootstrap можна віднести:

- висока швидкість розробки крос-браузерних та адаптивних веб-сайтів, саме завдяки готових компонентів всередині;
- дизайн компонентів є сучасним, адже стилізація HTML елементів виконана згідно UX/UI дизайну;
- не потрібно мати великий досвід у роботі з HTML, CSS, JS (достатньо розуміти базові поняття);
- компоненти в бібліотеці адаптовані для всіх популярних ОС та браузерів;
- знаходиться у вільному доступі, є безкоштовною бібліотекою. Доступ до відкритого вихідного коду можна знайти на GitHub;
- наявність документації з гарним перекладом на українську мову, тому даний інтерфейс стає досить легким в розумінні.

До недоліків можна віднести:

- зазвичай не задовольняє умови, коли проекти мають унікальний дизайн. Коли трапляється така ситуація – в основу беруть лише класи для будування адаптивної сітки;
- щоб використати якийсь один компонент бібліотеки, або клас, необхідно буде все одно підключити їх всі, хоча вони зовсім не потрібні у проекті.



Рисунок 3.41 – логотип Bootstrap

Git - це консольна утиліта, для відстеження та ведення історії зміни файлів у вашому проєкті (див. рис. 3.42) [15]. Найчастіше його використовують для коду, але можна і інших файлів. Наприклад, для картинок – корисно для дизайнерів.

За допомогою Git ви можете повернути свій проєкт до старої версії, порівнювати, аналізувати або пушити свої зміни до репозиторію. Репозиторієм називають сховище вашого коду та історію його змін. Git працює локально і всі ваші репозиторії зберігаються у певних папках на жорсткому диску. Також ваші репозиторії можна зберігати і в інтернеті. Зазвичай для цього використовують три сервіси:

- github;
- bitbucket;
- gitlab.

Кожна точка збереження вашого проєкту має назву комміт (commit). У кожного commit-а є hash (унікальний id) та коментар. З таких commit-ів збирається гілка. Гілка – це історія змін. Кожна гілка має свою назву. Репозиторій може містити в собі кілька гілок, що створюються з інших гілок або вливаються в них.



Рисунок 3.42 – логотип Git

Swiper.js – чудовий слайдер заточений для роботи з мобільними сайтами або веб-додатками [16]. Swiper забезпечує апаратне прискорення анімації на нативному рівні. Проект за кілька днів зібрав понад 3000 зірок на GitHub (див. рис. 3.43).

Є кілька причин, чому Swiper є найкращим вибором слайдеру для сайту:

- swiper написано на чистому JavaScript і не вимагає підключення додаткових бібліотек. Це означає, що ви не матимете зайвого запиту до сервера;
- його API дозволяє створювати власну розбивку на сторінки та кнопки навігації;
- swiper використовує для побудови макетів із гнучкою сіткою технологію Flex-box;
- swiper використовує техніку лінивого завантаження. Він затримує завантаження зображень на неактивних слайдах, доки користувач не проведе пальцем. Це прискорює завантаження сторінки;
- swiper абсолютно безкоштовний і має відкритий вихідний код.



Рисунок 3.43 – логотип Swiper.js

VUEX – це бібліотека керування станом, створена командою Vue для керування даними у програмах Vue.js. Вона забезпечує централізований спосіб керування даними, що використовуються в додатку, та дозволяє легко ними маніпулювати (див. рис. 3.44).

Він створює глобальну область (сховище), у якому можна розмістити всі стани, які спільно використовуватимуться різними компонентами. Він також робить наш код більш структурованим, полегшує налагодження, оскільки ми можемо використовувати DevTools для відстеження помилок, що виникають, і, звичайно, забезпечує реактивність, на якій і побудований Vue. Ви можете уявити аналогію з Vuex як про глобальний об'єкт window у JavaScript — до якого кожен компонент має доступ [17].



Рисунок 3.44 – логотип VUEX

Висновки до розділу 3

У третьому розділі було налаштовано середу розробки, збірник для проекту та підключено локальне середовище розробки до адмін-панелі Shopify.

Були описані веб-інструменти розробки, за допомогою яких буде виконуватись проект, описані їх переваги та недоліки.

Налаштував файл конфігурації для роботи з CMS-системою.

4 ПРОЦЕС ПРОЕКТУВАННЯ СТОРІНОК ТА КОМПОНЕНТІВ ВЕБЗАСТОСУНКУ

4.1 Опис програмної реалізації

Після налаштування програмного середовища та адмін-панелі можна приступати безпосередньо до розробки вебзастосунку.

Спочатку потрібно підключити необхідні бібліотеки та пакети, які знадобляться для роботи (див. рис. 4.1 – 4.2).

```
7
8  const DOMContentLoadedPromise = new Promise( executor: (resolve) => {
9    document.addEventListener( type: "DOMContentLoaded", listener: async () => {
10     resolve();
11   });
12
13   document.documentElement.classList.remove( tokens: "no-js");
14 });
15
16 import Swiper, {
17   Navigation,
18   Pagination,
19   Thumbs,
20   EffectFade,
21   Lazy,
22   Autoplay,
23   Scrollbar
24 } from "swiper";
25
26 Swiper.use( modules: [
27   Navigation,
28   Pagination,
29   Thumbs,
30   EffectFade,
31   Lazy,
32   Autoplay,
33   Scrollbar
34 ]);
35 global.Swiper = Swiper;
36
37 import * as bodyScrollLock from "body-scroll-lock";
38 global.bodyScrollLock = bodyScrollLock;
39
40 import $ from "jquery";
41 global.jQuery = $;
42 global.$ = $;
43
44 window.theme = window.theme || {};
```



```

32   "dependencies": {
33     "@shopify/theme-currency": "^4.0.2",
34     "@shopify/theme-images": "^4.1.0",
35     "@shopify/theme-product": "^4.1.0",
36     "@shopify/theme-product-form": "^4.1.0",
37     "@shopify/theme-sections": "^4.0.2",
38     "axios": "^0.21.0",
39     "body-scroll-lock": "^3.1.5",
40     "form-serialize": "^0.7.2",
41     "jquery": "^3.5.1",
42     "qs": "^6.10.3",
43     "swiper": "^6.8.0"
44   },

```

Рисунок 4.1 – 4.2 – підключення додаткових бібліотек та пакетів

Усі сторінки веб-сайту будуються в напрямку потоку, тобто зверху-вниз, тож розпочнемо саме зверху.

Header:

Компонент, який завжди є на кожній сторінці сайту та знаходиться найпершим – це хедер, його називають шапкою сайту. Він включає в себе панель оголошень, логотип сайту, навігацію, стрічку пошуку, особистий кабінет користувача, та кошик товарів (див. рис. 4.3).

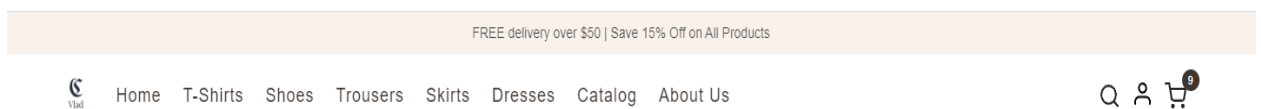


Рисунок 4.3 – зовнішній вигляд реалізованого хедеру

В хедері першою частиною виступає панель оголошень, яка вказує, що можна отримати безкоштовну доставку, якщо покупець придбає продуктів більш ніж на 50\$, також ця частина є гіперпосиланням на сторінку усіх продуктів, щоб користувачу було зручно одразу перейти до каталогу товарів.

До основної частини шапки відносяться логотип, навігація по сторінках, пошук, особистий кабінет, кошик обраних товарів.

Навігація представляє собою звичайні гіперпосилання на вказані веб-сторінки сайту.

Пошук надає змогу знайти потрібний продукт, або статтю серед усього каталогу магазину, щоб покупець не витрачав зайвий час на власноручний пошук, адже продуктів може бути досить багато, і це буде незручно для відвідувача. Пошук буде виконуватися при вводі щонайменше трьох символів у поле, щоб зменшити кількість вірогідних варіантів результату. Приклад роботи пошуку, коли продукт знайдено та навпаки – можна розглянути на рис. 4.4 – 4.5.

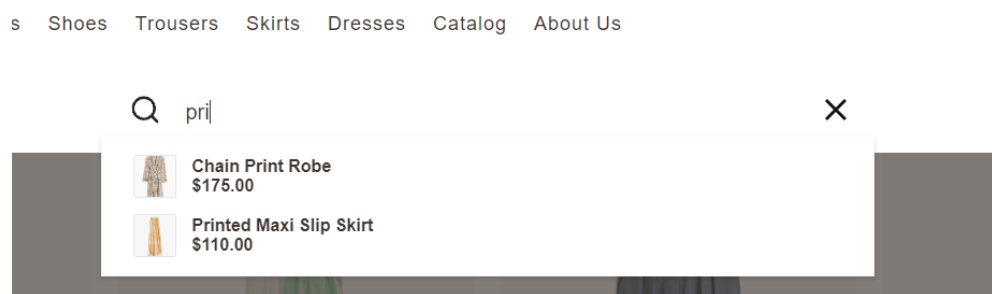


Рисунок 4.4 – зовнішній вигляд реалізованого пошуку з знайденими товарами

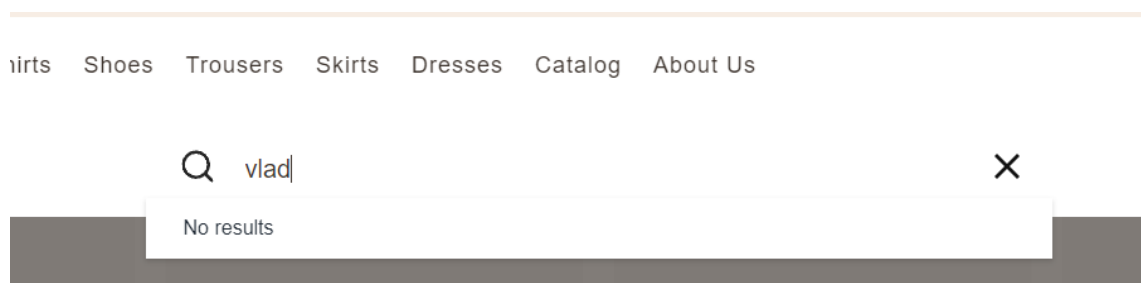


Рисунок 4.5 – зовнішній вигляд реалізованого пошуку без знайдених товарів

Особистий кабінет є дуже важливою річчю для користувача, адже уся інформація про його взаємодії з сайтом повинна знаходитися саме там.

Натиснувши на значок користувача в хедері можна потрапити до форм, в яких є вибір щодо реєстрації, або логіну до вже зареєстрованого акаунту (див. рис. 4.6).

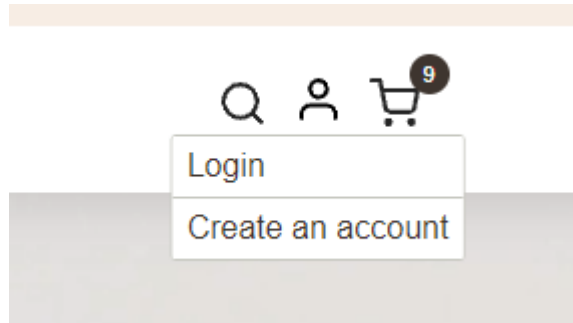
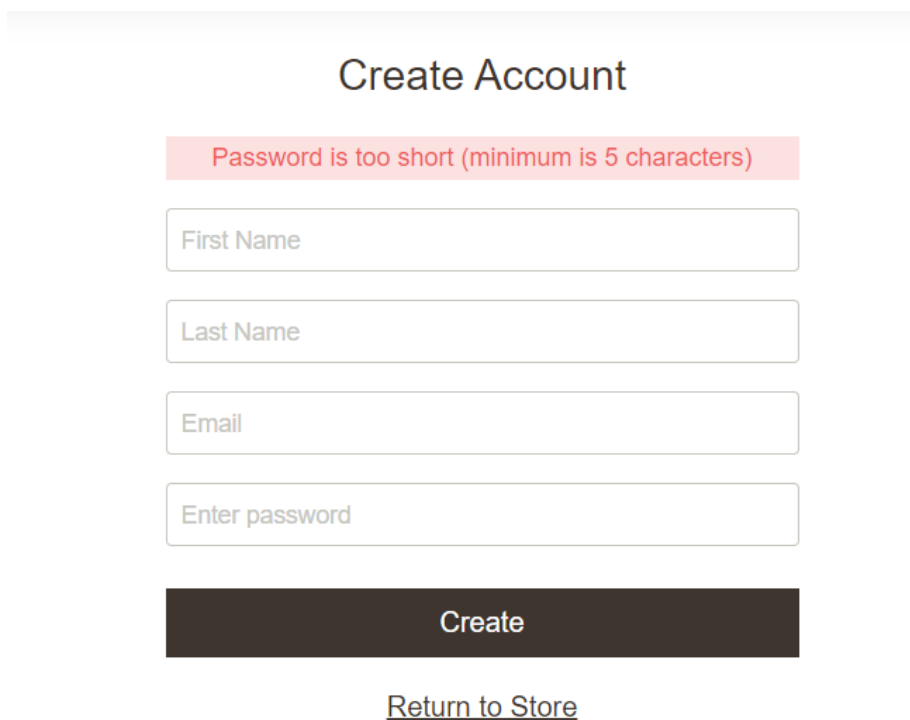


Рисунок 4.6 – перехід до особистого кабінету

Рисунок 4.7 – зовнішній вигляд реалізованої форми для реєстрації користувача

Форми в свою чергу мають валідацію, та сповістять користувача, якщо обов'язкове поле вводу порожнє, або заповнене некоректно (див. рис. 4.8).



The screenshot shows a 'Create Account' form. At the top, there is a red error message: 'Password is too short (minimum is 5 characters)'. Below this, there are four input fields: 'First Name', 'Last Name', 'Email', and 'Enter password'. At the bottom of the form is a dark 'Create' button. Below the button is a link that says 'Return to Store'.

Рисунок 4.8 – приклад валідації форми при спробі вказати короткий пароль

Після того, як поля будуть заповнені правильно – у профілі буде вказано активний акаунт користувача до якого був раніше виконана реєстрація (див. рис. 4.9).

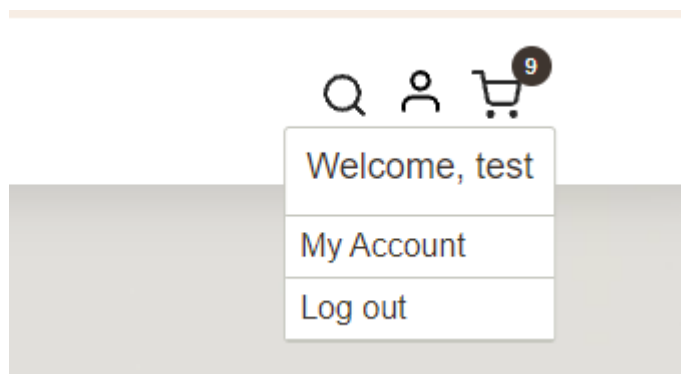


Рисунок 4.9 – успішно пройдена реєстрація та вхід до акаунту

Перша кнопка «My Account» буде відправляти користувача до сторінки особистого профілю (див. рис. 4.10), а за допомогою кнопки «Log out» буде здійснений вихід з особистого акаунту.

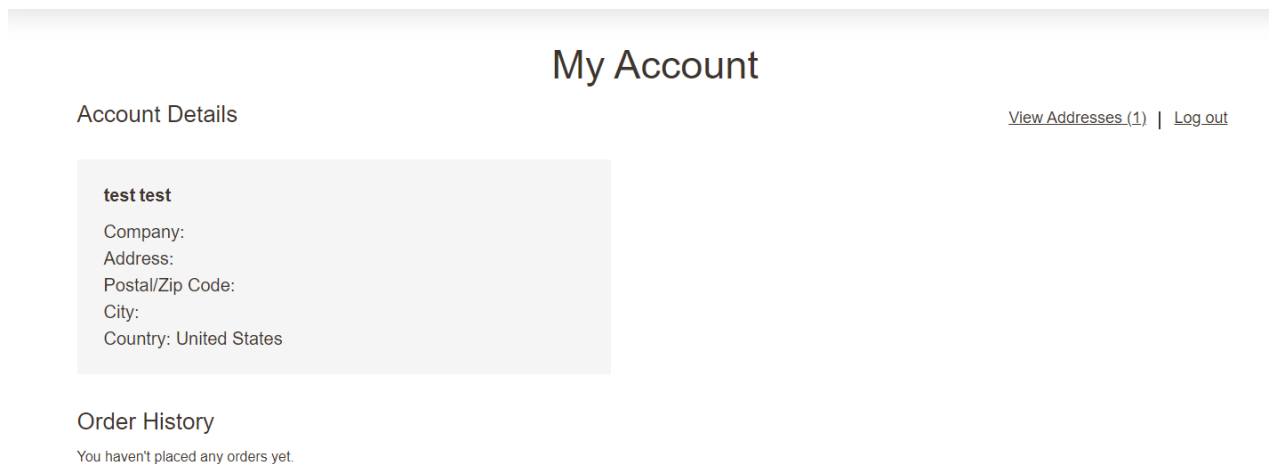
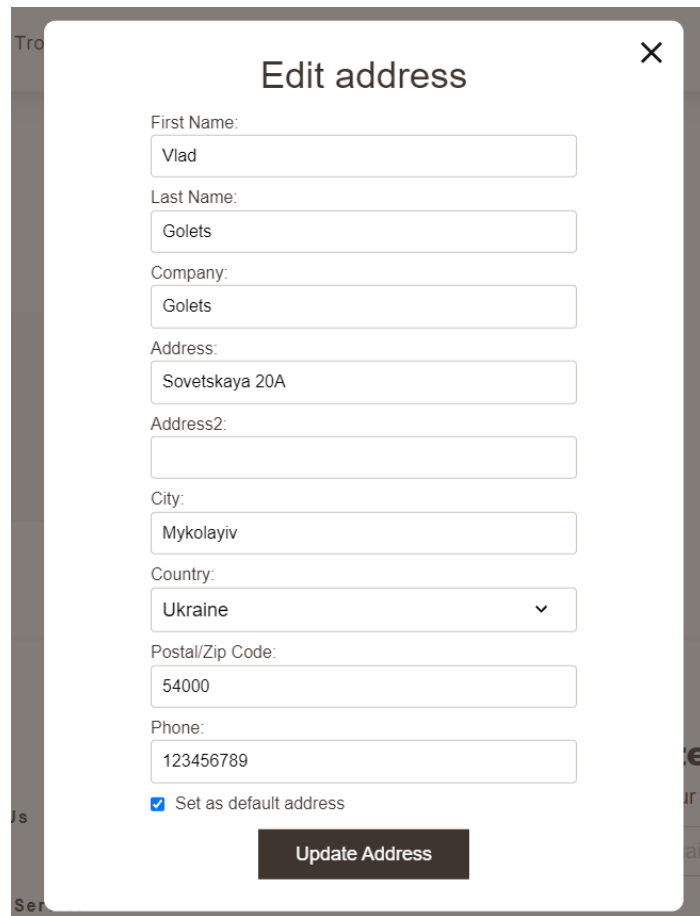


Рисунок 4.10 – зовнішній вигляд особистої сторінки користувача

На сторінці профілю надається інформація про деталі акаунту користувача, а саме його ім'я, прізвище, куди буде виконуватися доставка куплених продуктів, історія всіх покупок, посилання на сторінку з адресами, яку можна відредагувати для доставки товару в коректне місце, та кнопка виходу «Log Out». Перейшовши до вкладки «View Addresses» можна побачити поточну адресу для доставки продуктів (вказується за замовчуванням), її можна змінити натиснувши «Edit» (див. рис. 4.11 – 4.12), видалити, натиснувши «Delete» (див. рис. 4.13) та додати ще одну додаткову, натиснувши «Add a new address» (див. рис. 4.14 – 4.15).



Edit address

First Name:
Vlad

Last Name:
Golets

Company:
Golets

Address:
Sovetskaya 20A

Address2:

City:
Mykolayiv

Country:
Ukraine

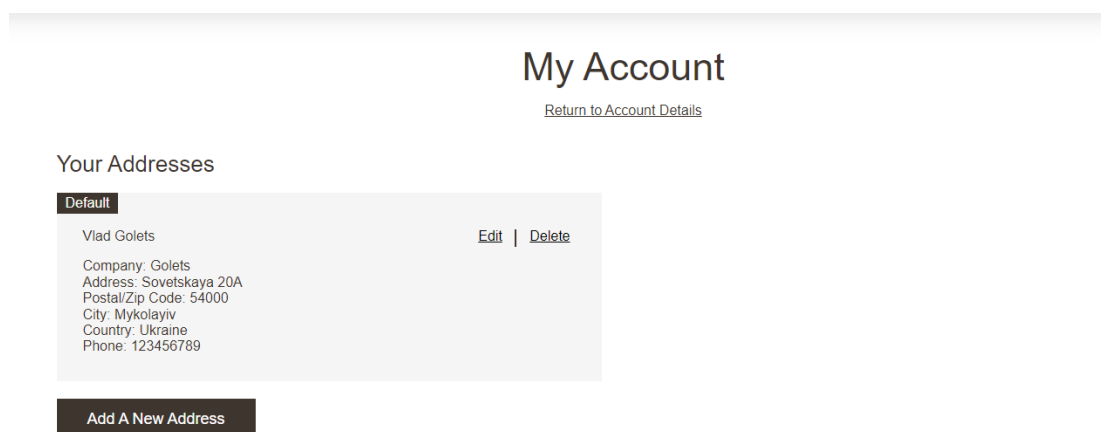
Postal/Zip Code:
54000

Phone:
123456789

Set as default address

Update Address

Рисунок 4.11 – зовнішній вигляд форми редагування поточної адреси покупця



My Account

[Return to Account Details](#)

Your Addresses

Default

Vlad Golets [Edit](#) | [Delete](#)

Company: Golets
Address: Sovetskaya 20A
Postal/Zip Code: 54000
City: Mykolayiv
Country: Ukraine
Phone: 123456789

Add A New Address

Рисунок 4.12 – результат оновлення поточної адреси доставки

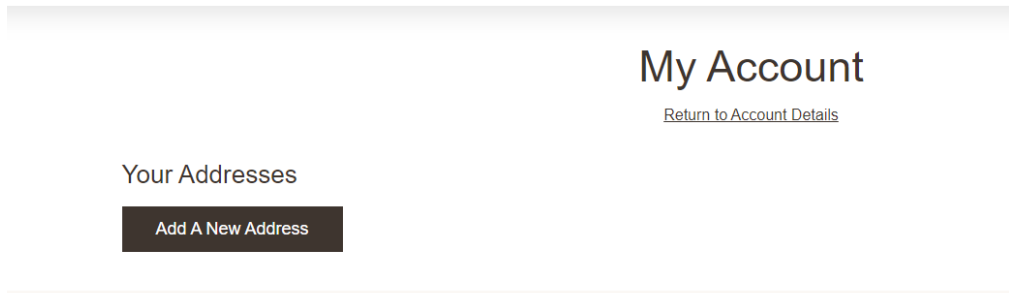
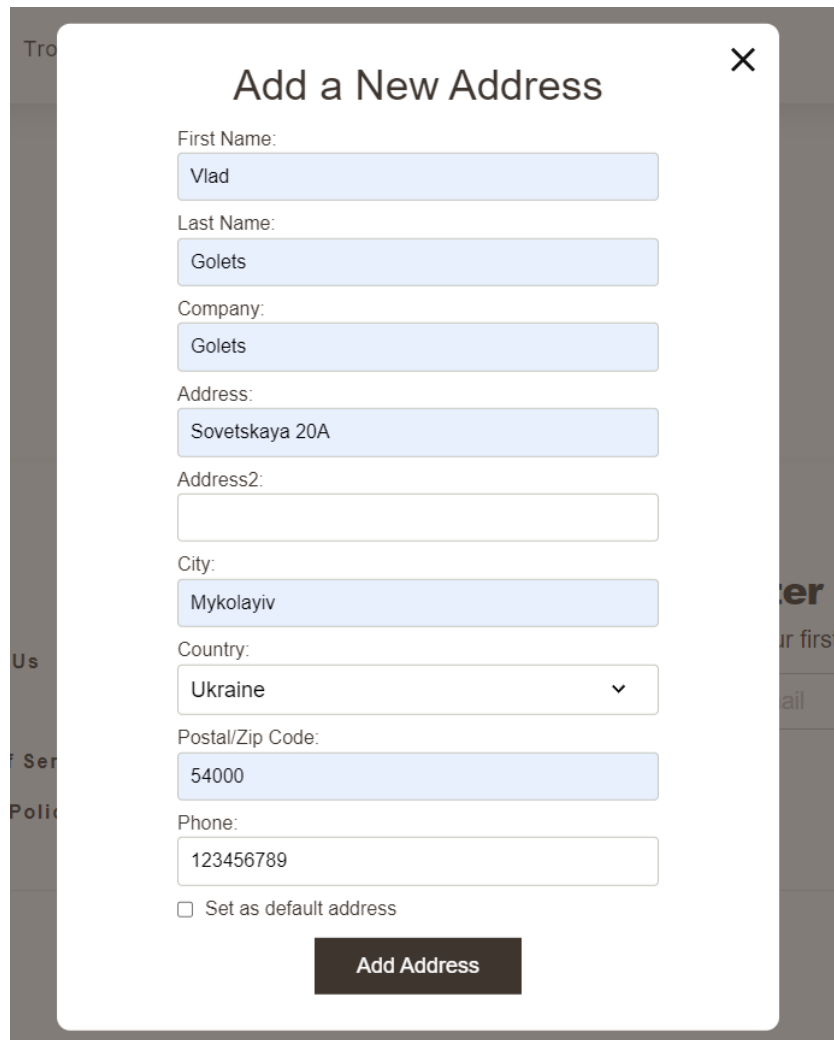


Рисунок 4.13 – результат видалення поточної адреси доставки



The screenshot displays a modal form titled 'Add a New Address'. The form contains the following fields and options:

- First Name: Vlad
- Last Name: Golets
- Company: Golets
- Address: Sovetskaya 20A
- Address2: (empty)
- City: Mykolayiv
- Country: Ukraine (dropdown menu)
- Postal/Zip Code: 54000
- Phone: 123456789
- Set as default address
- Add Address** button

Рисунок 4.14 – зовнішній вигляд форми додавання поточної адреси покупця

My Account

[Return to Account Details](#)

Your Addresses

Default

Vlad Golets [Edit](#) | [Delete](#)

Company: Golets
Address: Sovetskaya 20A
Postal/Zip Code: 54000
City: Mykolayiv
Country: Ukraine
Phone: 123456789

[Add A New Address](#)

Рисунок 4.15 – результат додавання поточної адреси доставки

На рис. 4.16. зображено форму логіну.

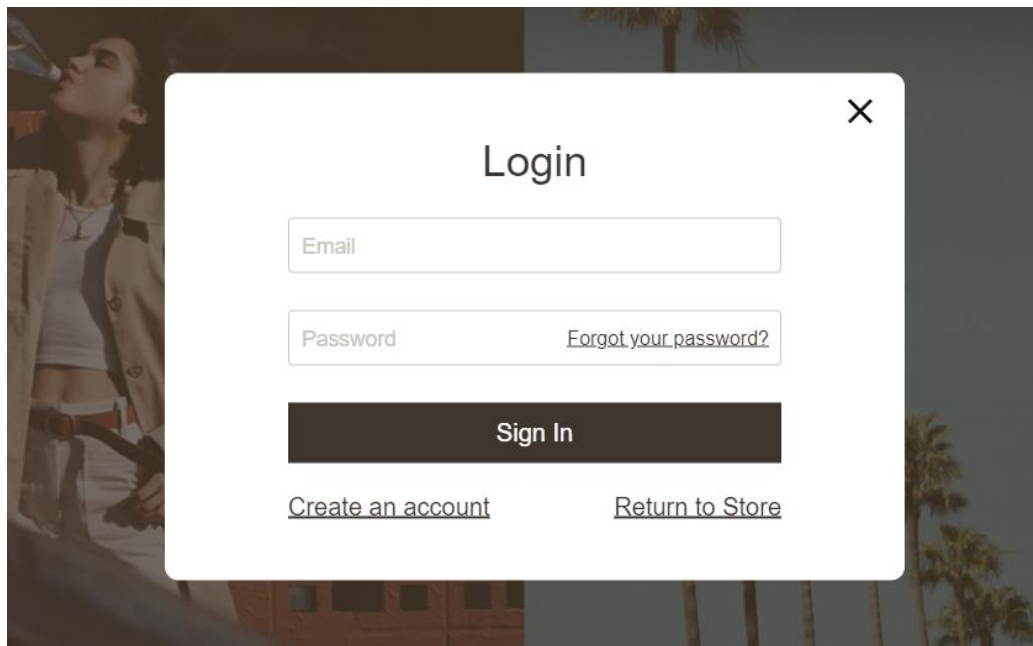


Рисунок 4.16 – зовнішній вигляд реалізованої форми для логіну у вже існуючий профіль користувача

Також присутня форма для відновлення акаунту, у випадку, якщо користувач забув пароль до входу в інтернет-магазин (див. рис. 4.17).

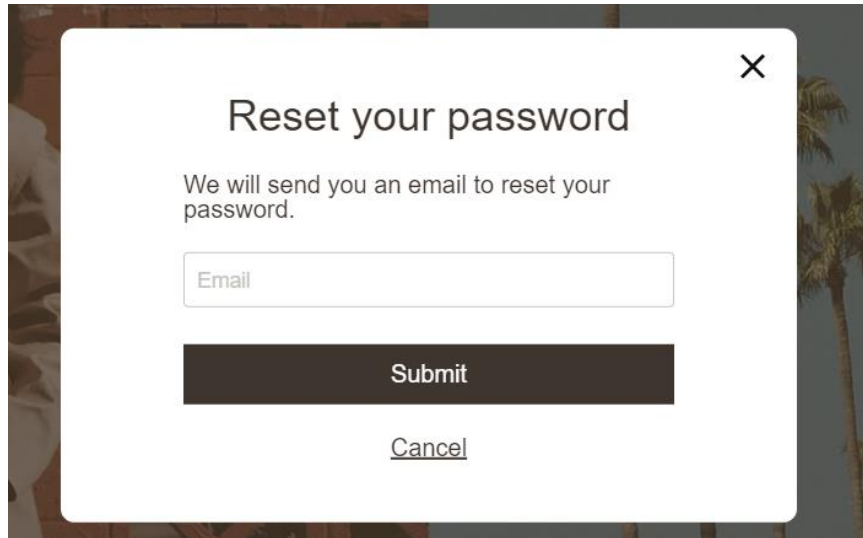


Рисунок 4.17 – зовнішній вигляд реалізованої форми для відновлення акаунту

Після вводу поштової скриньки, у формі буде виведено повідомлення про те, що на цю пошту було відправлено смс з підтвердженням про запит та встановлення нового паролю в акаунті (див. рис. 4.18 – 4.19).

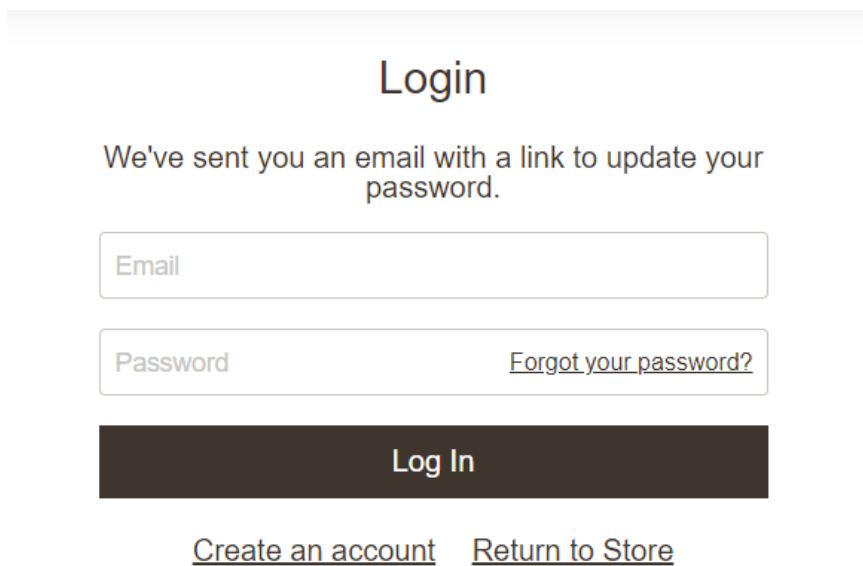


Рисунок 4.18 – повідомлення щодо відправленого підтвердження на пошту

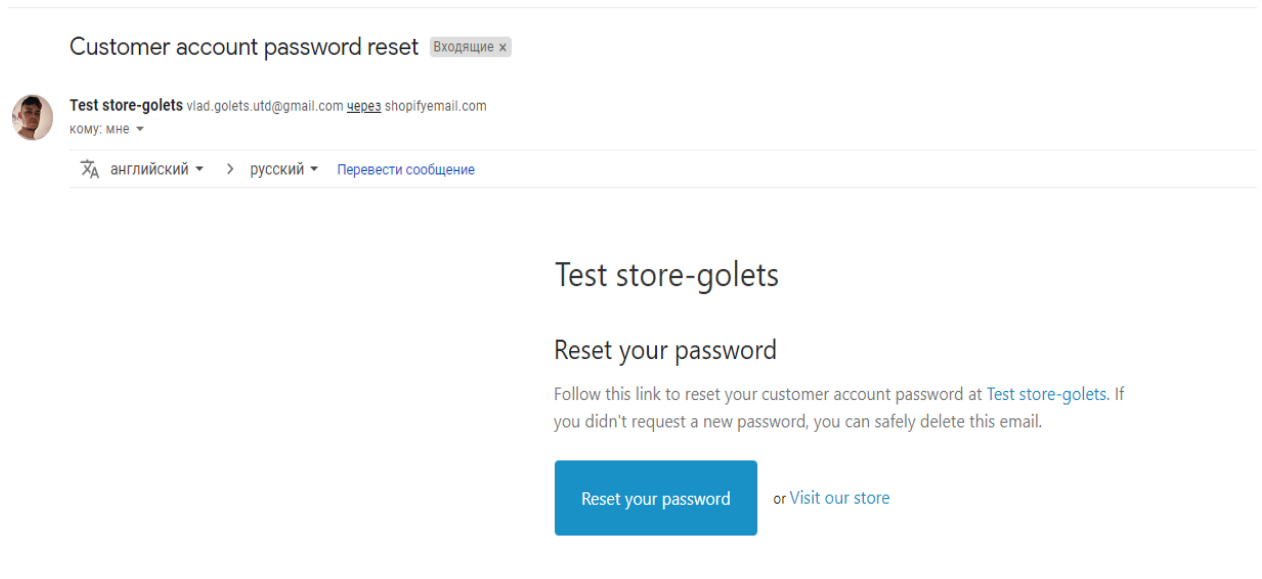


Рисунок 4.19 – наявність повідомлення на пошті

Далі користувача знову переведе на сайт, а саме на форму введення нового паролю для профіля (див. рис. 4.20).

The image shows a web form titled 'Reset account password'. Below the title is the instruction 'Enter a new account password'. There are two input fields, each containing six dots to represent masked text. At the bottom of the form is a dark button labeled 'Reset Password'.

Рисунок 4.20 – форма перезапису пароля користувача

В результаті отримав динамічну модель хедеру, вона включає в себе можливість зміни кольору, тексту, вставки гіперпосилання для панелі оголошень, вставку логотипу сайту та задання динамічних веб-сторінок у навігації, на малюнку з правої частини лежить динамічний контент (див. рис. 4.21).

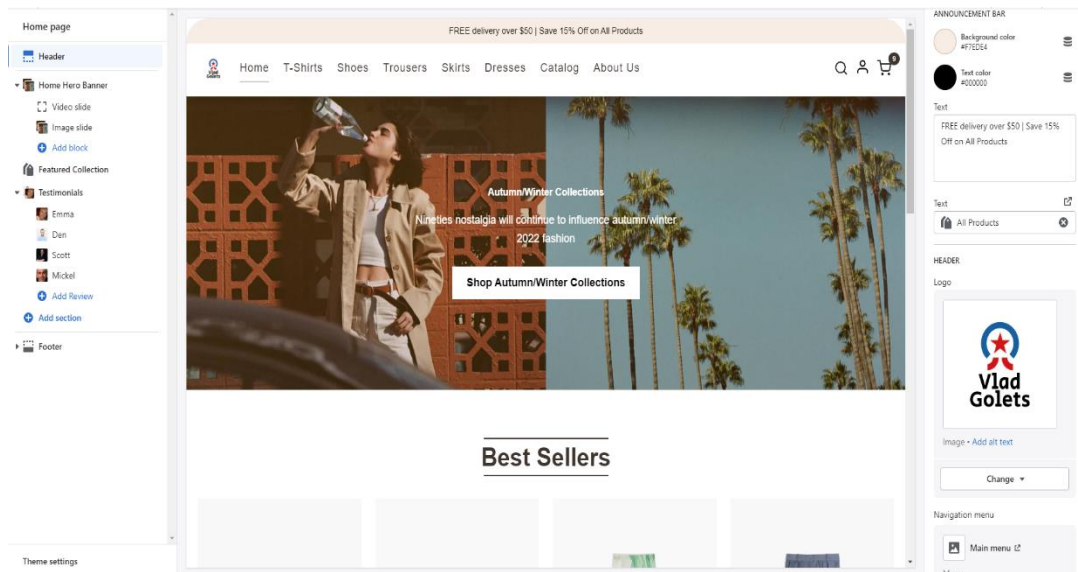


Рисунок 4.21 – вигляд створеної системи налаштувань для хедеру

Banner:

Наступним одним з найпопулярніших компонентів є банер. В мене він виконаний у вигляді слайдеру з можливістю динамічного вибору зображення, або відео, також є динамічна складова для заповнення текстового контенту, обрання палітри кольорів, який лежить зверху банера, та цей текстовий контент є змога вирівняти по вертикалі і горизонталі будь-як на вибір впродовж всього слайдеру. (див. рис. 4.22).



Рисунок 4.22 – зовнішній вигляд банеру (всередині лежить відео)

Featured Collection, Product-Cards:

Далі доволі часто використовуються картки з продуктами, саме на них покупець звертає увагу, адже це основний товар магазину. Картки з продуктами я помістив до секції, яка виконана у вигляді слайдеру. Секція має динамічний заголовок та вибір колекції в адмін-панелі, відповідно до якої будуть виведені картки з продуктами.

Кожен продукт в картці містить свої опції. Поєднання декількох опцій в одному продукті називається варіантом продукту, наприклад опція кольору, матеріалу та розміру для одяжі, тож можуть бути виведені різні варіації продукту відповідно до їх опцій (див. рис. 4.23).

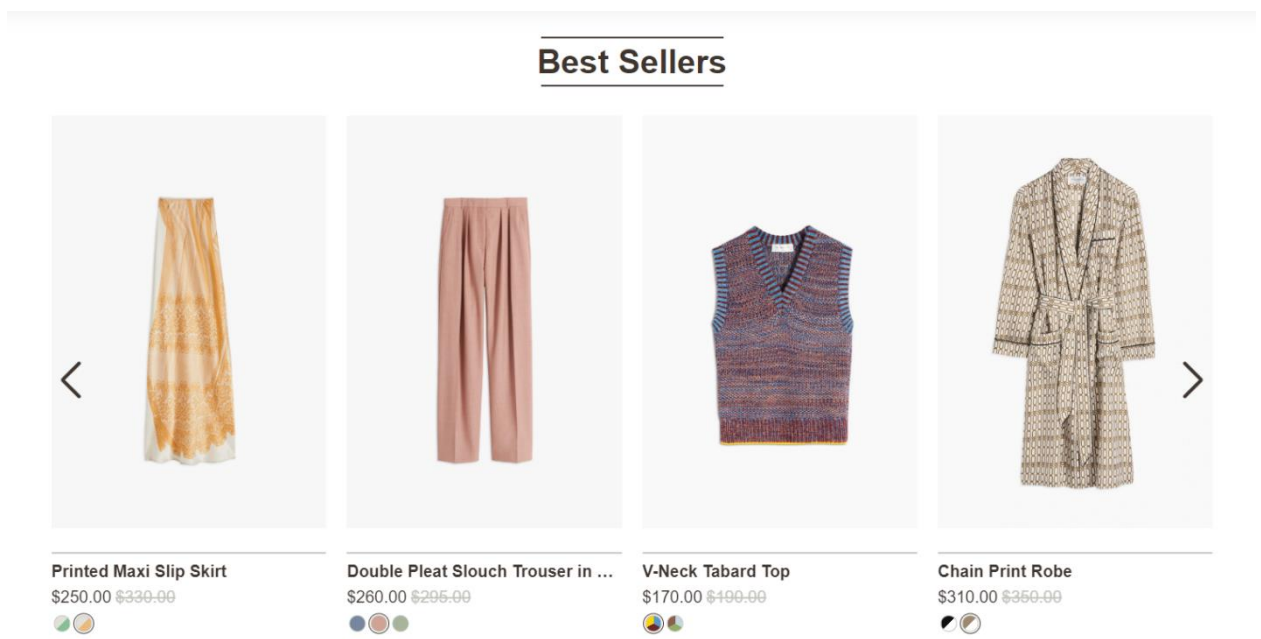


Рисунок 4.23 – зовнішній вигляд секції слайдеру з картками продукту

Для кожного варіанту динамічно задається колір перемикача в картці (може бути заданий один колір, так і декілька, в залежності від поєднання кольорів у зображення на продукті), відповідно до натискання на обраний колір – буде виконуватися рендер картки продукту і змінюватися наповнення картки відповідно від обраного варіанту. На десктоп версіях в

картках продукту наявний функціонал появи додаткових опцій та кнопки «Quick Add» при наведенні мишкою на товар (див. рис. 4.24).

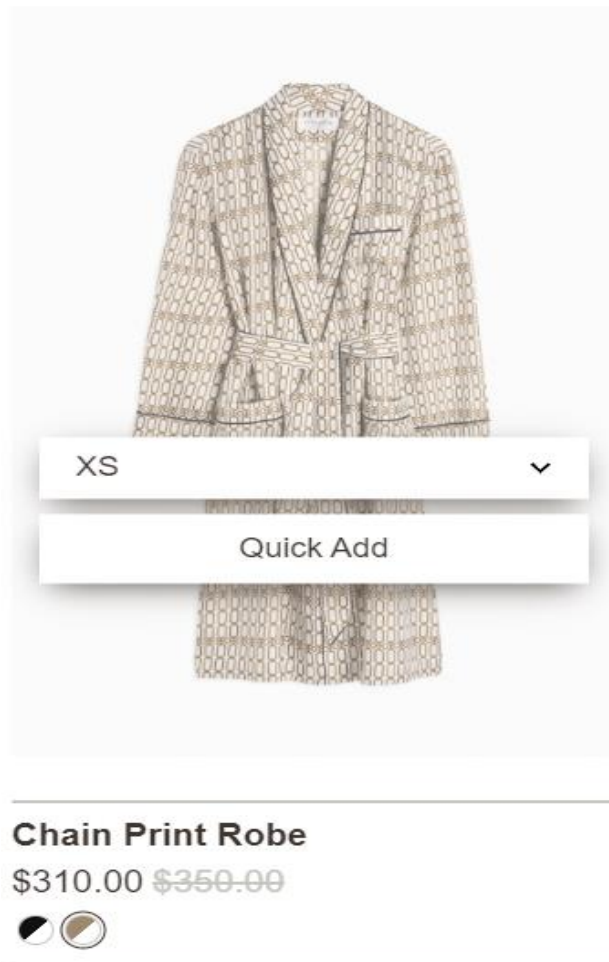


Рисунок 4.24 – десктоп функціонал появи опцій на кнопки для швидкого придбання товару при наведенні мишкою на картку

На пристроях з максимальною шириною до 1200 пікселів, ці опції та кнопка знаходяться знизу карточки, адже на мобільних пристроях немає можливості навести курсор (див. рис. 4.25).

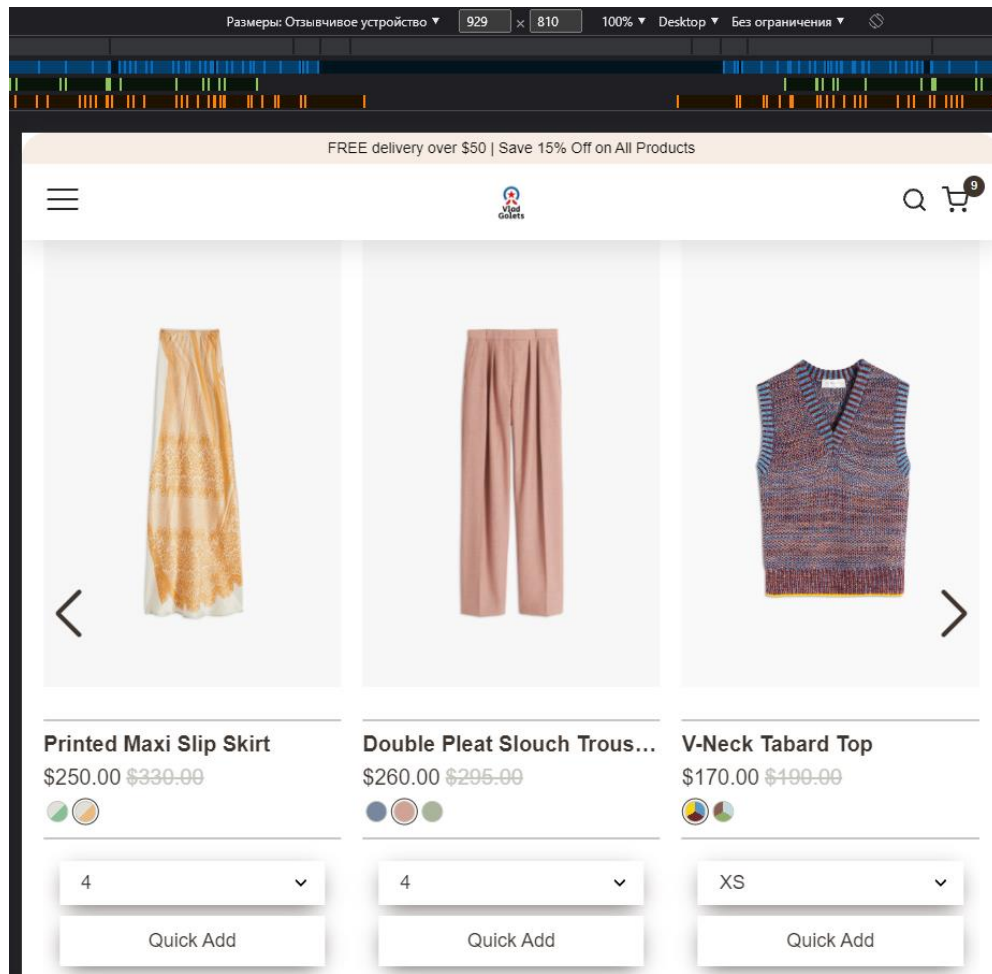


Рисунок 4.25 – зовнішній вигляд секції на планшетах та мобільних пристроях

Іноді не у всіх відвідувачів досить швидкий інтернет, або у когось в браузері можуть просто бути виключені скрипти, тому про це треба теж подбати, і робити це треба особливо в тих місцях де знаходяться продукти інтернет-магазину, але найкраще буде дбати про це взагалі звідусіль. В мене для кожної секції виконана додаткова логіка, щоб без написаних скриптів JavaScript – сторінка продовжувала функціонувати і користувач все одно зміг дійти до придбання товару. Тобто коли JavaScript нічого не виконує, тобто просто виключений, треба виконувати щось подібне, тому я за допомогою додатковий перевірок в коді не зламаю сторінку, а просто вийду з скрипту і почну обробляти випадки за допомогою звичайного шаблонізатору Liquid, HTML тегу <noscript> та допоміжних можливостей системи Shopify. Якщо в

користувача буде відключений JavaScript браузером, то він побачить наступну картину (див. 4.26).

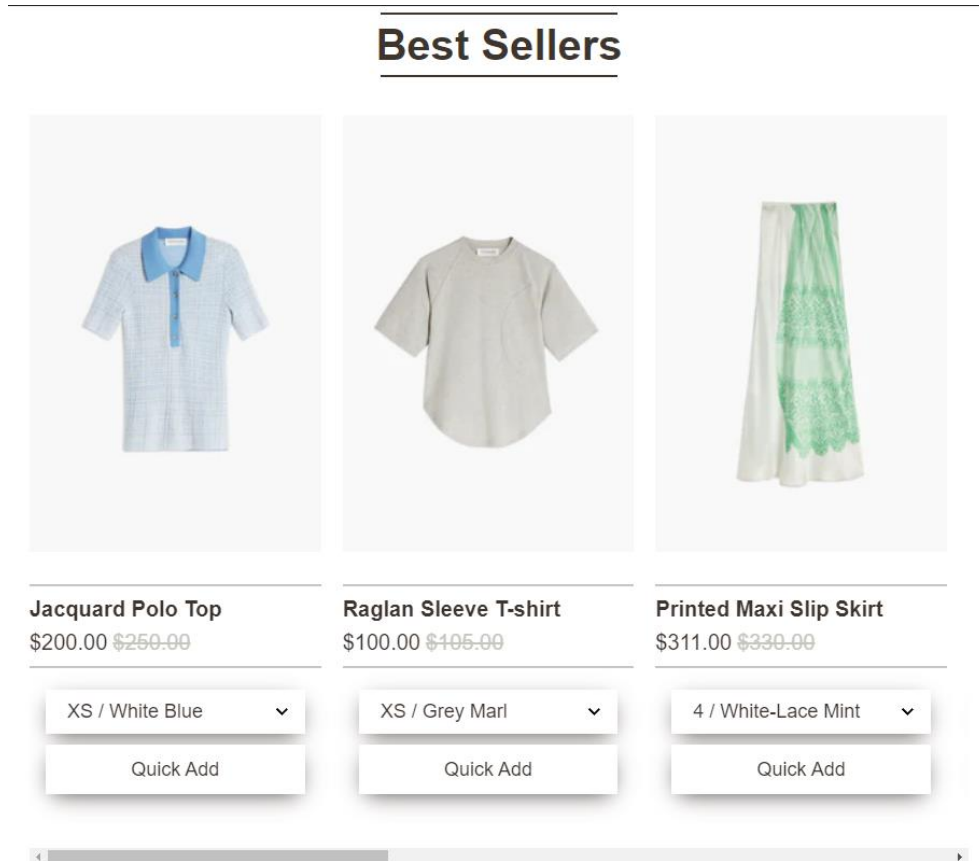


Рисунок 4.26 – зовнішній вигляд секції з відключеними скриптами

Тобто, якщо порівнювати, то за наявності скриптів буде працювати слайдер, але без них сторінка не зламається, а з'явиться скролл з такими ж картками, за наявності скриптів виконувався рендер варіантів продукту відповідно до обраних опцій розміру та кольору, тобто я робив асинхронний запит до CartAPI Shopify за допомогою HTTP-клієнту «axios» і користувачу не треба було покидати сторінку з продуктами, все виконувалося на фоні, але якщо скриптів не буде, то буде виконуватися звичайна відправка форми, і нас нажалі перенаправить на сторінку кошику, але все ж таки це дуже чудово, що нічого не зламається і клієнт надалі зможе придбати собі товар, тому що вбудована логіка Shopify дозволить це йому без проблем зробити. Тобто

можна зробити висновок, що якщо б не було прийнято до уваги ці моменти, то у застосунку користувача посипалися б помилки і все б загальмувалося, в свою чергу покупець пішов шукати наступний інтернет-магазин для покупки, а продавець залишився без оборту грошей, саме через це потрібно звертати на це увагу.

Reviews:

Важливою є секція з відгуками. Коли користувач заходить до сайту перший раз, можливо в нього можуть виникати сумніви щодо продавців, але коли покупець бачить відгуки до продуктів, психологія людини зазвичай бере верх і з'являється більше довіри до веб-магазину.

Реалізована секція «Reviews» виконана за допомогою слайдеру з наявною пагінацією, включає в собі динамічні картки з відгуками покупців, а саме текст, зображення, можливість зміни кольору для усієї частини, також можна задати в адмін-панелі швидкість переходу слайдів один до одного (див. рис. 4.27).

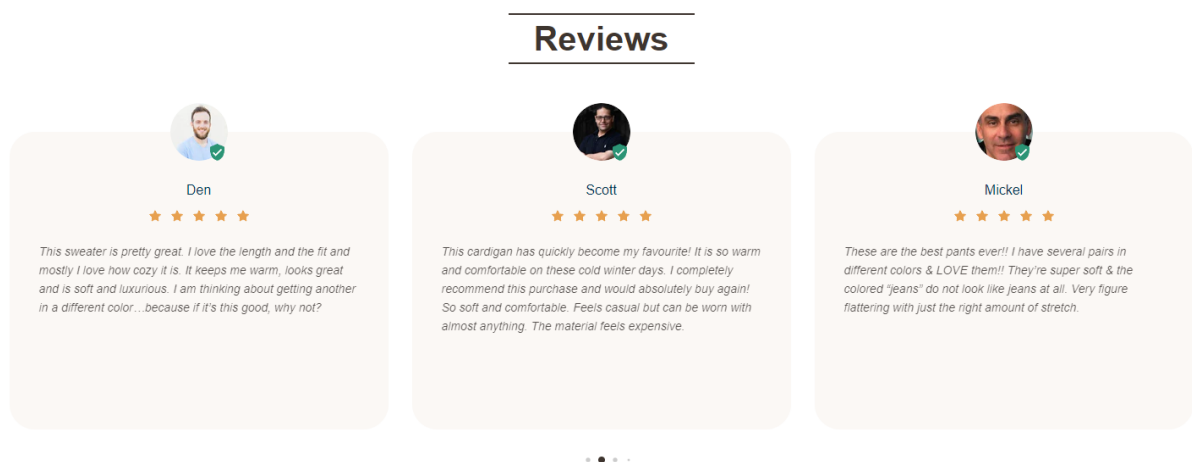


Рисунок 4.27 – реалізація секції «Reviews» у вигляді слайдеру

Footer:

Як і хедер, футер наявний на кожній з сторінок магазину, в собі він містить список гіперпосилань на інші веб-сторінки магазину, кнопку, яка перемістить користувача моментально вгору до сайту, що досить зручно, якщо на сторінці досить багато контенту і клієнт дійшов до самого низу веб-сайту, та форму з підпискою для розсилки майбутніх пропозицій та знижок. Обов'язковим моментом є те, що на веб-сайті повинен бути текст з копірайтом (див. рис. 4.28).

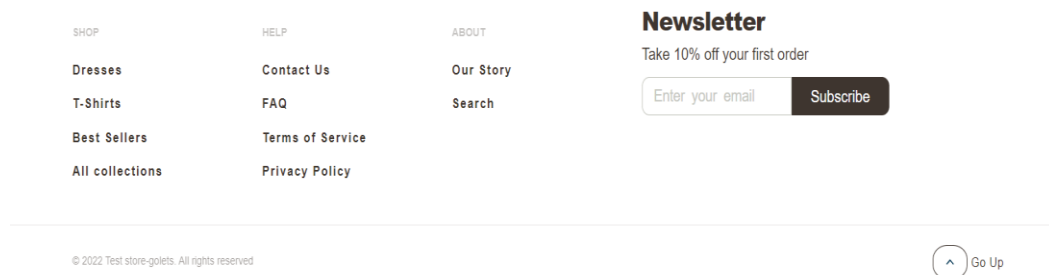


Рисунок 4.28 – реалізація футеру

Форма для підписки, що знаходиться у футері допомагає підтримувати з клієнтами більш тісний зв'язок, надаючи їм важливу інформацію про оновлення, нові наповнення товарами, майбутніми знижками і т.д.

Результат роботи форми з підпискою зображений на рис. 4.29 – 4.30.

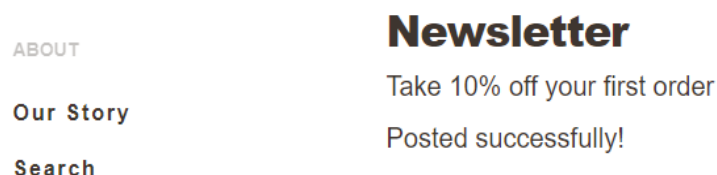


Рисунок 4.29 – реалізація форми підписки клієнта на розсилку (повідомлення «Posted Successfully» після вводу поштової скриньки у поле, тепер користувачу будуть надходити новини, які буде надавати продавець)

Сторінка FAQ:

FAQ – Frequently Asked Questions. На цій сторінці надають питання, які зустрічаються частіше за інші та відповіді на них. Контент сторінки повністю динамічний, задається в адмін-панелі (заголовок секції, запитання, відповідь на запитання). Секція FAQ виконана у вигляді акордеону, щоб контент не занадто розтягував сторінку (див. рис. 4.30).

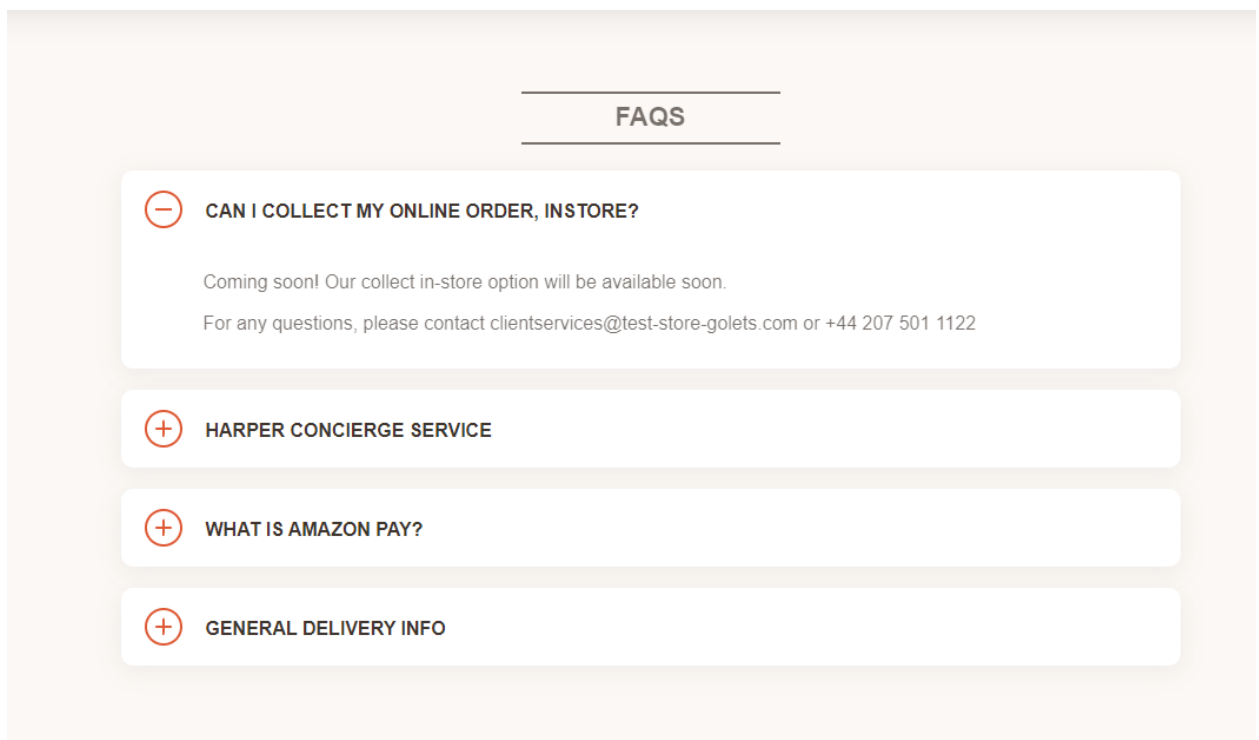


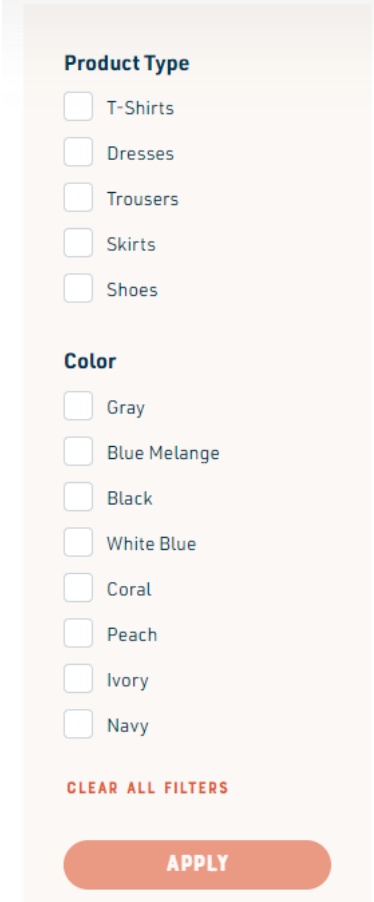
Рисунок 4.30 – реалізація сторінки FAQ

Сторінка колекцій (PLP):

Це так званий каталог усіх товарів, в якому присутні усі продукти магазину, фільтрація продуктів за певними параметрами та сортування за критеріями.

На сторінці зліва зображено сайдбар з наявною фільтрації продуктів (див. рис. 4.31), посередині сітка з продуктів (див. рис 4.32), справа зверху

функція сортування (див. рис. 4.33), знизу сторінки активна пагінація, якщо продуктів буде більше ніж 20 (див. рис. 4.34).



The image shows a vertical sidebar for product filtering. It is divided into two main sections: 'Product Type' and 'Color'. Each section contains a list of items with an unchecked checkbox to its left. Below the 'Color' section is a red link that says 'CLEAR ALL FILTERS'. At the bottom of the sidebar is a large, rounded orange button with the word 'APPLY' in white capital letters.

Product Type

- T-Shirts
- Dresses
- Trousers
- Skirts
- Shoes

Color

- Gray
- Blue Melange
- Black
- White Blue
- Coral
- Peach
- Ivory
- Navy

[CLEAR ALL FILTERS](#)

APPLY

Рисунок 4.31 – Реалізація сайдбару з наявною фільтрацією продуктів

Кафедра інтелектуальних інформаційних систем
Інформаційна система створення онлайн платформи торгових операцій за допомогою Shopify

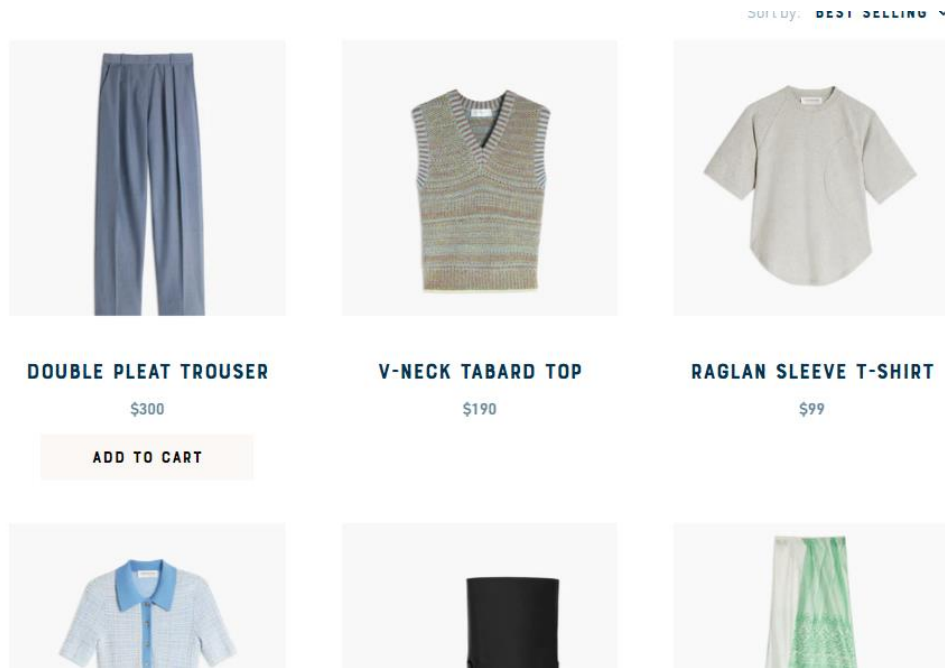


Рисунок 4.32 – Реалізація сітки з каталогу

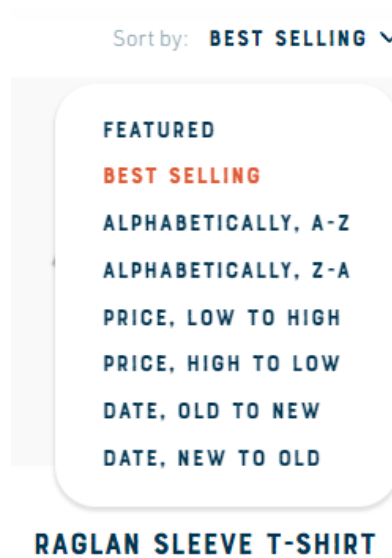


Рисунок 4.33 – Реалізація функції сортування



Рисунок 4.34 – Реалізація функції пагінації

Фільтрація функціонує за допомогою «AND» логіки, а значення фільтрів – за допомогою «OR». Тобто, можна повернути товари певної категорії та певного кольору (див. рис. 4.35), або можна повернути товари тієї чи іншої категорії (див. рис. 4.36). Коли фільтрація використовується, вона відображається в колекції або URL-адресі пошуку за допомогою URL параметрів.

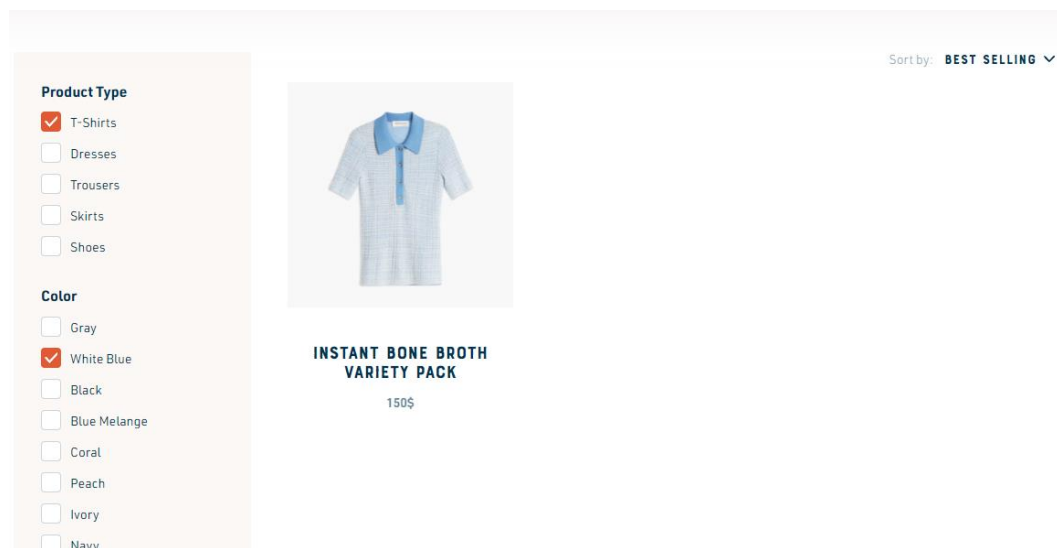


Рисунок 4.35 – Приклад роботи фільтрації за допомогою «AND»

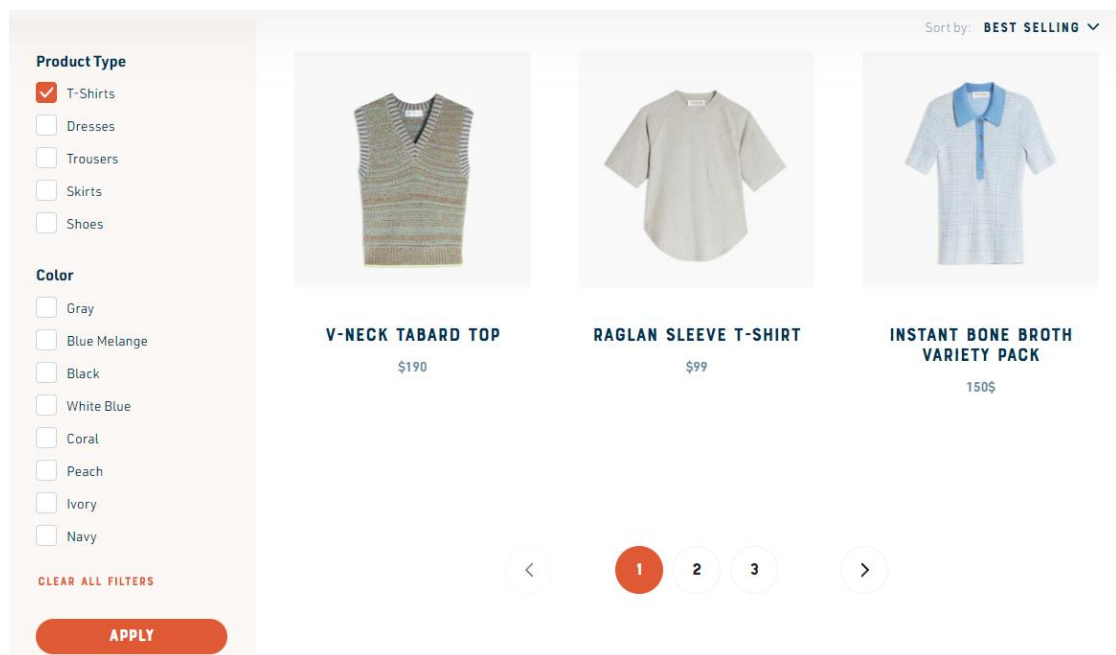


Рисунок 4.36 – Приклад роботи фільтрації за допомогою «OR»

Сортування товарів можна обрати відносно певного критерію за допомогою параметру URL-адреси «sort_by» на сторінці колекцій. Через Liquid object я отримав доступ до доступних параметрів з атрибутом «sort_options», вибраного параметру за атрибутом «sort_by», параметру аз замовченням з атрибутом «default_sort_by». Усі доступні параметри я вивів до HTML-елементу «select», щоб користувач на сайті міг зробити свій вибір щодо сортування, а потім я ініціалізую селектор на основі поточних параметрів і параметрів за замовчуванням відносно активного параметру. Кожен раз, коли покупець буде обирати нову опцію сортування – я використовую JS, щоб додати нове значення до параметру URL-адреси та виконую рендер сторінки з оновленим контентом.

Minicart:

Компонент «Minicart» є аналогом сторінки кошика, але результати обчислень та уся логіка виконуються в реальному часі, без перезавантаження сторінки, що робить вигляд інтерфейсу для користувача більш зрозумілим та не перевантаженим, виїжджає у вигляді сайдбару при додавання продукту до кошику, або при натискання кнопки кошику у хедері.

Використовується глобальне сховище задля управління станом для компоненту «Minicart» за допомогою Vuex, щоб отримати багатшу декомпозицію та зручність у використанні, управлінні станами.

Компонент має кілька станів вигляду, а саме: порожня, коли немає ні одного продукту (див. рис. 4.37), наповнена, хоча б з одним продуктом всередині (див. рис. 4.38) та в часі очікування відповіді, коли відбувається рендер контенту (див. рис. 4.39).

В головному компоненті «Minicart» реалізовано дочірній компонент «Shipping Progress», має вигляд прогрес-бару задля відображення суми, якої не вистачає у кошику для отримання безкоштовної доставки обраних продуктів,

початковий стан має значення «100», тобто покупець набуде безкоштовної доставки, коли придбає продуктів на суму більшу за 100\$ (див. рис. 4.40).

← MY CART

YOUR CART IS EMPTY

Рисунок 4.37 – Реалізація стану порожнього кошику

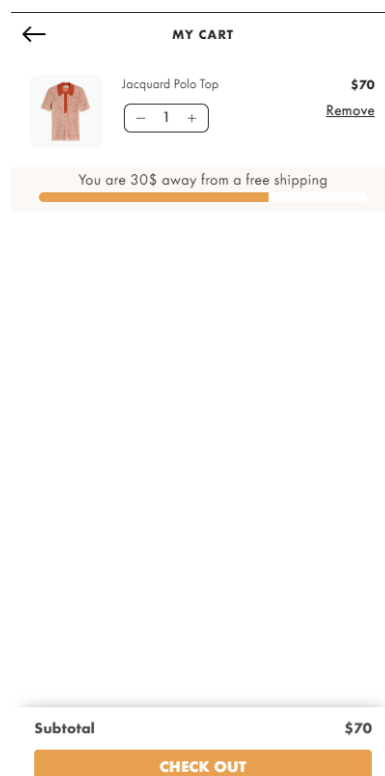


Рисунок 4.38 – Реалізація стану заповненого кошику (мінімум один продукт всередині)

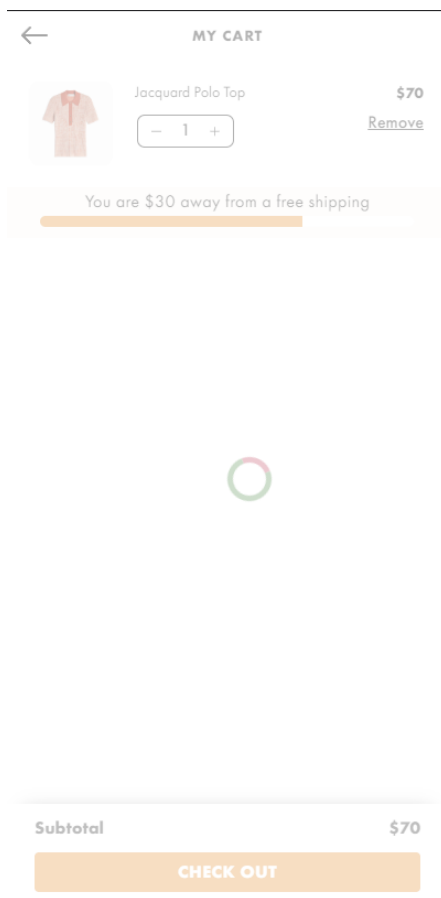


Рисунок 4.39 – Реалізація стану очікування (додано прелоадер, щоб при проблемах з інтернетом користувач розумів, що йде оновлення у кошику, без проблем з інтернетом спеціально задано затримку на півсекунди для імітації завантаження)



Рисунок 4.40 – Реалізація заповненого прогрес-бару у режимі реального часу та ререндер компоненту

Логіка роботи компонента «Minicart» написана за допомогою технологій VueJs та Vuex, що надає змогу роботи з реактивністю.

Сторінка продукту (PDP):

Сторінка містить в собі інформацію про продукт, його зображення у вигляді слайдеру, опції, відгуки, кількість, кнопку додавання до кошику.

Під час вибору вибору опцій кольору реалізовано динамічний рендерер опису варіанту продукту, його зображень слайдеру, ціни та відгуків (див. рис. 4.41).



Рисунок 4.41 – Реалізація сторінки продукту з варіантами опцій

Висновки до розділу 4

У ході виконання цього розділу було підключено необхідні бібліотеки та пакети до проекту.

Було розроблені компоненти секцій для таких сторінок, як: головна сторінка, сторінка колекцій, сторінка продукту, сторінка 404, сторінка Contact Us, сторінка FAQ, сторінка About Us, реактивний сайдбар кошику.

Описано роль компонентів на сторінках та їх застосування.

Спеціальний розділ

ОХОРОНА ПРАЦІ

до бакалаврської кваліфікаційної роботи

«ІНФОРМАЦІЙНА СИСТЕМА СТВОРЕННЯ ОНЛАЙН ПЛАТФОРМИ ТОРГОВИХ ОПЕРАЦІЙ ЗА ДОПОМОГОЮ SHOPIFY»

Спеціальність 122 – Комп'ютерні науки

122-БКР-401.21810107

Виконав: студент 4 курсу, групи 401

В.В. Голець

«__» _____ 2022 р.

Консультант _____ старший викладач _____

(наук. ступінь, вчене звання)

О.В. Макарова

(підпис, ініціали та прізвище)

«__» _____ 2022 р.

Миколаїв – 2022

5 ОХОРОНА ПРАЦІ

Однією з найголовніших складових на підприємствах різного напрямку є охорона праці. Слідуючи елементарним правилам безпеки і вимог охорони праці, роботодавець може досягти значного пониження виробничого травматизму. Якщо роботодавець не дотримується вимог охорони праці, то це може спричинити виробничі травми будь-якої важкості.

В охорону праці входить цілий комплекс заходів, що мають нормативно-правове забезпечення: організаційно-технічні, соціально-економічні, санітарно-гігієнічні та лікувально-профілактичні. Умови праці, безпека технологічних процесів, а також санітарно-побутові умови повинні відповідати вимогам нормативно-правових актів про охорону праці.

Організація роботи працівників та створення умов для зростання продуктивності, забезпечення трудової і виробничої дисципліни та дотримання законодавства про працю і правил охорони праці роботодавцем повинна виконуватись чітко та правильно. Працівники в свою чергу повинні працювати чесно, сумлінно, своєчасно виконувати розпорядження роботодавця, дотримуватись трудової і технологічної дисципліни, а також вимоги нормативно-правових актів про охорону праці.

Враховуючи, що законодавством України проголошено пріоритет життя і здоров'я людини, метою розділу «Охорона праці та безпека у надзвичайних ситуаціях» («Охорона праці» – для дипломних робіт бакалаврів) є створення безпечних і здорових умов праці на робочих місцях, в робочих зонах, у виробничих приміщеннях та забезпечення безпеки людини у надзвичайних ситуаціях. Досягається це опрацюванням питань умов праці, гігієни праці і виробничої санітарії, техніки безпеки, пожежної безпеки, цивільного захисту, екологічної безпеки та безпеки життєдіяльності людини в умовах надзвичайних ситуацій.

5.1 Загальні питання охорони праці

З розвитком науково-технічного прогресу важливу роль грає можливість безпечного виконання людьми своїх трудових обов'язків. У зв'язку з цим була створена і розвивається наука про безпеку праці і життєдіяльності людини.

Охорона праці - це система законодавчих, організаційно-технічних, соціально-економічних, санітарно-гігієнічних і лікувально-профілактичних мір і засобів, спрямованих на збереження життя, здоров'я й працездатності людини в процесі праці. Завдання охорони праці полягає в тому, щоб звести до мінімуму ймовірність поразки працюючого під дією небезпечного виробничого фактору або захворювання під дією шкідливого виробничого фактору з одночасним забезпеченням комфортних умов при максимальній продуктивності праці. Закон України "Про охорону праці" визначає основні положення по реалізації конституційного права громадян на охорону їх життя і здоров'я в процесі трудової діяльності; регулює взаємини між адміністрацією і працівником в незалежності від форм власності; встановлює єдиний порядок організації охорони праці в Україні [1].

Мета і зміст БЖД:

- виявлення і вивчення чинників навколишнього середовища, що негативно впливають на здоров'я людини;
- ослаблення дії цих чинників до безпечних меж або виключення їх якщо це можливо;
- ліквідація наслідків катастроф і стихійних лих.

Коло практичних задач БЖД перш за все обумовлений вибором принципів захисту, розробкою і раціональним використанням засобів захисту людини і природного середовища від впливу техногенних джерел і стихійних явищ, а також коштів, що забезпечують комфортний стан середовища життєдіяльності.

Охорона здоров'я трудящих, забезпечення безпеки умов праці, ліквідація професійних захворювань і виробничого травматизму складає одну з головних турбот людського суспільства. Звертається увага на необхідність широкого застосування прогресивних форм наукової організації праці, зведення до мінімуму ручної, малокваліфікованої праці, створення обстановки, що виключає професійні захворювання і виробничий травматизм [2].

На робочому місці повинні бути передбачені заходи захисту від можливого впливу небезпечних і шкідливих факторів виробництва. Рівні цих чинників не повинні перевищувати граничних значень, обумовлених правовими, технічними та санітарно-технічними нормами. Ці нормативні документи зобов'язують до створення на робочому місці умов праці, при яких вплив небезпечних і шкідливих чинників на працюючих або усунуто зовсім, або знаходиться в допустимих межах.

Даний розділ присвячений розгляду наступних питань:

- визначення оптимальних умов праці інженера - програміста;
- розрахунок освітленості;
- розрахунок рівня шуму.

Структура управління охороною праці на підприємстві

Система управління охороною праці (СУОП) є комплексом дій з підготовки, прийняття та реалізації рішень з метою виконання організаційних, технічних, санітарно-гігієнічних і лікувально-профілактичних заходів. Головна мета введення СУОП на підприємстві – забезпечення безпеки, збереження життя, здоров'я та працездатності працівників під час трудового процесу.

Характеристика умов праці програміста

Науково-технічний прогрес виконав серйозні зміни щодо умов виробничої діяльності робітників розумової праці. Їх праця стала більш інтенсивним, напруженим, які вимагають значних витрат розумової, емоційної і фізичної енергії. Це зажадало комплексного рішення проблем ергономіки, гігієни і організації праці, регламентації режимів праці та відпочинку.

В даний час комп'ютерна техніка широко застосовується у всіх областях діяльності людини. При роботі з комп'ютером людина піддається дії ряду небезпечних і шкідливих виробничих факторів: електромагнітних полів (діапазон радіочастот: ВЧ, УВЧ і СВЧ), інфрачервоного і іонізуючого випромінювання, шуму і вібрації, статичної електрики і ін [3].

Робота з комп'ютером характеризується значною розумовою напругою і нервово-емоційним навантаженням операторів, високою напруженістю зорової роботи і достатньо великим навантаженням на м'язи рук при роботі з клавіатурою ЕОМ. Велике значення має раціональна конструкція і розташування елементів робочого місця, що важливо для підтримки оптимальної робочої пози людини-оператора.

У процесі роботи з комп'ютером необхідно дотримувати правильний режим праці та відпочинку. В іншому випадку у персоналу наголошуються значна напруга зорового апарату з появою скарг на незадоволеність роботою, головні болі, дратівливість, порушення сну, втому і хворобливі відчуття в очах, в поясниці, в області шиї і руках.

Забарвлення і коефіцієнти віддзеркалення

Забарвлення приміщень і меблів повинні сприяти створенню сприятливих умов для зорового сприйняття, гарного настрою.

Джерела світла, такі як світильники і вікна, які дають віддзеркалення від поверхні екрану, значно погіршують точність знаків і тягнуть за собою перешкоди фізіологічного характеру, які можуть виразитися в значній напрузі, особливо при тривалій роботі. Віддзеркалення, включаючи віддзеркалення від вторинних джерел світла, повинне бути зведено до мінімуму. Для захисту від надмірної яскравості вікон можуть бути застосовані штори і екрани [4].

Залежно від орієнтації вікон рекомендується наступне фарбування стін і підлоги:

- *вікна орієнтовані на південь*: стіни зеленувато-блакитного або світло-блакитного кольору, підлога зеленого;
- *вікна орієнтовані на північ*: стіни світло-оранжевого або оранжево-жовтого кольору, підлога червонувато-оранжевого;
- *вікна орієнтовані на схід*: стіни жовто-зеленого кольору, підлога зеленого або червонувато-оранжевого;
- *вікна орієнтовані на захід*: стіни жовто-зеленого або голубувато-зеленого кольору, підлога зелена або червонувато-оранжевий.

У приміщеннях, де знаходиться комп'ютер, необхідно забезпечити наступні величини коефіцієнта віддзеркалення: для стелі: 60 ... 70%, для стін: 40 ... 50%, для підлоги: близько 30%. Для інших поверхонь і робочих меблів: 30 ... 40%.

Освітлення

Правильно спроектоване і виконане виробниче освітлення покращує умови зорової роботи, знижує стомлюваність, сприяє підвищенню продуктивності праці, благотворно впливає на виробниче середовище, надаючи позитивну психологічну дію на працюючого, підвищує безпеку праці і знижує травматизм.

Недостатність освітлення приводить до напруги зору, ослабляє увагу, приводить до настання передчасної стомленості. Надмірно яскраве освітлення викликає засліплення, роздратування і різь в очах. Неправильний напрямок світла на робочому місці може створювати різкі тіні, відблиски, дезорієнтувати працюючого. Всі ці причини можуть призвести до нещасного випадку або профзахворювань, тому такий важливий правильний розрахунок освітленості.

Існує три види освітлення – природне, штучне і поєднане (природне і штучне разом) [5].

Природне освітлення – освітлення приміщень денним світлом, що потрапляє через світлові прорізи в зовнішніх огорожувальних конструкціях приміщення. Природне освітлення характеризується тим, що змінюється в широких межах залежно від часу дня, пори року, характеру області і ряду інших чинників.

Штучне освітлення застосовується при роботі в темний час доби і вдень, коли не вдається забезпечити нормовані значення коефіцієнта природного освітлення (похмура погода, короткий світловий день). Освітлення, при якому недостатнє за нормами природне освітлення доповнюється штучним, називається змішаним освітленням.

Штучне освітлення підрозділяється на робоче, аварійне, евакуаційне, охоронне. Робоче освітлення, у свою чергу, може бути загальним або комбінованим. Загальне – освітлення, при якому світильники розміщуються у верхній зоні приміщення рівномірно, або, як розташоване устаткування. Комбіноване – освітлення, при якому до загального додається місцеве освітлення.

При виконанні робіт категорії високої зорової точності (найменший розмір об'єкту розрізнення 0,3 ... 0,5 мм) величина коефіцієнта природного освітлення (КЕО) повинна бути не нижче 1,5%, а при зоровій роботі середньої

точності (найменший розмір об'єкту розрізнення 0,5 ... 1,0 мм) КЕО повинен бути не нижче 1,0%. В якості джерел штучного освітлення звичайно використовуються люмінесцентні лампи типу ЛБ, або ДРЛ, які попарно об'єднуються в світильники, які повинні розташовуватися рівномірно над робочими поверхнями [4].

Вимоги до освітленості в приміщеннях, де встановлені комп'ютери, наступні: при виконанні зорових робіт високої точності загальна освітленість повинна складати 300лк, а комбінована – 750лк; аналогічні вимоги при виконанні робіт середньої точності – 200 і 300лк відповідно.

Крім того все поле зору повинне бути освітлено достатньо рівномірно – ця основна гігієнічна вимога. Іншими словами, ступінь освітлення приміщення і яскравість екрану комп'ютера повинні бути приблизно однаковими, оскільки яскраве світло в районі периферійного зору значно збільшує напруженість очей і, як наслідок, призводить до їх швидкої стомлюваності.

Параметри мікроклімату

Параметри мікроклімату можуть мінятися в широких межах, у той час як необхідною умовою життєдіяльності людини є підтримка постійності температури тіла завдяки терморегуляції, тобто здатності організму регулювати віддачу тепла в навколишнє середовище. Принцип нормування мікроклімату – створення оптимальних умов для теплообміну тіла людини з навколишнім середовищем.

Обчислювальна техніка є джерелом істотних тепловиділень, що може привести до підвищення температури і зниження відносної вологості в приміщенні. У приміщеннях, де встановлені комп'ютери, повинні дотримуватися певні параметри мікроклімату. У санітарних нормах СН-245-71 встановлені величини параметрів мікроклімату, що створюють комфортні

умови. Ці норми встановлюються в залежності від пори року, характеру трудового процесу і характеру виробничого приміщення (див. табл. 5.1) [3].

Об'єм приміщень, в яких розміщені працівники обчислювальних центрів, не повинен бути меншим $19,5 \text{ м}^3$ / людини з урахуванням максимального числа одночасно працюючих в зміні. Норми подачі свіжого повітря в приміщення, де розташовані комп'ютери, приведені в табл. 5.2.

Таблиця 5.1 – Параметри мікроклімату для приміщень, де встановлені комп'ютери

Період року	Температура повітря, град. С	Відносна вологість повітря, %	Швидкість руху повітря, м/с
Холодна	22 – 24	40 – 60	0,1
Тепла	23 – 25	40 – 60	0,1 - 0,2

Таблиця 5.2 – Норми подачі свіжого повітря в приміщення, де розташовані комп'ютери

Характеристика приміщення	Об'ємна витрата подається в приміщення свіжого повітря, м^3 / на одну людину в годину
Об'єм до 20м^3 на особу	Не менше 30
$20 \dots 40\text{м}^3$ на особу	Не менше 20
Більш 40м^3 на особу	Природна вентиляція

Для забезпечення комфортних умов використовуються як організаційні методи (раціональна організація проведення робіт залежно від пори року і доби, чергування праці і відпочинку), так і технічні засоби (вентиляція, кондиціонування повітря, опалювальна система).

Шум і вібрація

Шум погіршує умови праці надаючи шкідливу дію на організм людини. Працюючі в умовах тривалої шумової дії випробовують дратівливість, головні болі, запаморочення, зниження пам'яті, підвищену стомлюваність, зниження апетиту, біль у вухах і т.д. Такі порушення в роботі ряду органів і систем організму людини можуть викликати негативні зміни в емоційному стані людини аж до стресових. Під впливом шуму знижується концентрація уваги, порушуються фізіологічні функції, з'являється втома у зв'язку з підвищеними енергетичними витратами і нервово-психічним напруженням. Все це знижує працездатність людини і її продуктивність, якість і безпеку праці. Тривала дія інтенсивного шуму на слух людини приводить до його часткової або повної втрати [6].

У табл. 5.3 вказані граничні рівні звуку залежно від категорії тяжкості і напруженості праці, що є безпечними відносно збереження здоров'я і працездатності.

Таблиця 5.3 – Граничні рівні звуку, дБ, на робочих місцях

Категорія напруженості праці	Категорія важкості праці			
	I. Легка	II. Середня	III. Важка	IV. Дуже важка
I. Мало напружений	80	80	75	75
II. Помірно напружений	70	70	65	65
III. Напружений	60	60	-	-
IV. Дуже напружений	50	50	-	-

Рівень шуму на робочому місці математиків-програмістів і операторів відеоматеріалів не повинен перевищувати 50дБА, а в залах обробки інформації на обчислювальних машинах – 65дБА. Для зниження рівня шуму стіни і стеля

приміщень, де встановлені комп'ютери, можуть бути облицьовані звукопоглинальними матеріалами. Рівень вібрації в приміщеннях обчислювальних центрів може бути понижений шляхом встановлення устаткування на спеціальні віброізолятори.

Електромагнітне і іонізуюче випромінювання

Більшість вчених вважають, що як короткочасне, так і тривалий вплив усіх видів випромінювання від екрану монітора не небезпечно для здоров'я персоналу, що обслуговує комп'ютери. Проте вичерпних даних щодо небезпеки дії випромінювання від моніторів на працюючих з комп'ютерами не існує і дослідження в цьому напрямі продовжуються [3].

Допустимі значення параметрів неіонізуючого електромагнітного випромінювання від монітора комп'ютера представлені в табл. 5.4.

Максимальний рівень рентгенівського випромінювання на робочому місці оператора комп'ютера звичайно не перевищує 10мкбер / ч, а інтенсивність ультрафіолетового і інфрачервоного випромінювання від екрану монітора лежить в межах 10 ... 100МВт / м².

Таблиця 5.4 – Допустимі значення параметрів неіонізуючого електромагнітного випромінювання (відповідно до СанПіН 2.2.2.542-96)

Найменування параметра	Допустимі значення
Напруженість електричної складової електромагнітного поля на відстані 50см від поверхні монітора	10В / м
Напруженість магнітної складової електромагнітного поля на відстані 50см від поверхні монітора	0,3 А / м

<p>Напруженість електростатичного поля не повинна перевищувати:</p> <p>для дорослих користувачів</p> <p>для дітей дошкільних установ і що навчаються</p> <p>середніх спеціальних і вищих навчальних закладів</p>	<p>20кВ / м</p> <p>15кВ / м</p>
--	---------------------------------

Для зниження дії цих видів випромінювання рекомендується застосовувати монітори із зниженим рівнем випромінювання (MPR-II, TCO-92, TCO-99), встановлювати захисні екрани, а також дотримуватися регламентовані режими праці та відпочинку.

Режим праці

Як вже було неодноразово відзначено, при роботі з персональним комп'ютером дуже важливу роль грає дотримання правильного режиму праці та відпочинку. В іншому випадку у персоналу наголошуються значна напруга зорового апарату з появою скарг на незадоволеність роботою, головні болі, дратівливість, порушення сну, втому і хворобливі відчуття в очах, в поясиці, в області ший і руках [3].

У табл. 5.5 представлені відомості про регламентовані перерви, які необхідно робити при роботі на комп'ютері, залежно від тривалості робочої зміни, видів і категорій трудової діяльності з ВДТ (дисплейний термінал) і ПЕВМ (відповідно до СанПіН 2.2.2 542-96 «Гігієнічні вимоги до дисплейних терміналів, персональних електронно-обчислювальних машин і організації робіт »).

Таблиця 5.5 – Час регламентованих перерв при роботі на комп'ютері

Категорія роботи з ВДТ або ПЕОМ	Рівень навантаження за робочу зміну при видах роботи з ВДТ			Сумарний час регламентованих перерв, мін	
	Група А, кількість знаків	Група Б, кількість знаків	Група В, годин	При 8-годинній зміні	При 12-годинній зміні
I	до 20000	до 15000	до 2,0	30	70
II	до 40000	до 30000	до 4,0	50	90
III	до 60000	до 40000	до 6,0	70	120

Примітка. Час перерв дано при дотриманні зазначених санітарних правил і норм. У разі невідповідності фактичних умов праці вимогам санітарних правил і норм час регламентованих перерв слід збільшити на 30%.

Відповідно до СанПіН 2.2.2 546-96 всі види трудової діяльності, пов'язані з використанням комп'ютера, розділяються на три групи:

група А: робота з зчитування інформації з екрана ВДТ або ПЕОМ з попереднім запитом;

група Б: робота по введенню інформації;

група В: творча робота в режимі діалогу з ЕОМ.

Ефективність перерв підвищується при поєднанні з виробничою гімнастикою або організації спеціального приміщення для відпочинку персоналу із зручними м'якими меблями, акваріумом, зеленою зоною і т.п.

Розрахунок освітлення

Розрахунок освітлення робочого місця зводиться до вибору системи освітлення, визначенню необхідного числа світильників, їхнього типу і розміщення. Виходячи з цього, розрахуємо параметри штучного освітлення.

Зазвичай штучне освітлення виконується за допомогою електричних джерел світла двох видів: ламп накаливання і люмінесцентних ламп. Будемо використовувати люмінесцентні лампи, які порівняно з лампами розжарювання мають ряд істотних переваг [5]:

- за спектральним складом світла вони близькі до денного, природного світла;
- володіють більш високим ККД (у 1,5-2 рази вище, ніж ККД ламп розжарювання);
- мають підвищену світловіддачу (в 3-4 рази вище, ніж у ламп розжарювання);
- більш тривалий термін служби.

Розрахунок освітлення проводиться для кімнати площею 15 м^2 , ширина якої 5 м, висота – 3 м. Скористаємося методом світлового потоку [4].

Для визначення кількості світильників визначимо світловий потік, падаючий на поверхню за формулою:

, де

F – розраховується світловий потік, Лм;

E – нормована мінімальна освітленість, Лк (визначається за таблицею).

Роботу програміста, відповідно до цієї таблиці, можна віднести до розряду точних робіт, отже, мінімальна освітленість буде $E = 300\text{лк}$;

S – площа освітлюваного приміщення (у нашому випадку $S = 15\text{ м}^2$);

Z – відношення середньої освітленості до мінімальної (звичайно приймається рівним 1,1 ... 1,2, нехай $Z = 1,1$);

K – коефіцієнт запасу, враховує зменшення світлового потоку лампи в результаті забруднення світильників у процесі експлуатації (його значення залежить від типу приміщення й характеру проведених у ньому робіт і в нашому випадку $K = 1,5$);

n – коефіцієнт використання, (виражається відношенням світлового потоку, що падає на розрахункову поверхню, до сумарного потоку всіх ламп і обчислюється в частках одиниці; залежить від характеристик світильника, розмірів приміщення, фарбування стін і стелі, що характеризуються коефіцієнтами відображення від стін (P_C) і стелі (P_{II}), значення коефіцієнтів P_C і P_{II} були зазначені вище: $P_C = 40\%$, $P_{II} = 60\%$. Значення **n** визначимо по таблиці коефіцієнтів використання різних світильників. Для цього обчислимо індекс приміщення по формулі:

, де

S – площа приміщення, $S = 15 \text{ м}^2$;

h – розрахункова висота підвісу, $h = 2.92 \text{ м}$;

A – ширина приміщення, $A = 3 \text{ м}$;

B – довжина приміщення, $B = 5 \text{ м}$.

Підставивши значення отримаємо:

Знаючи індекс приміщення **I**, за таблицею 7 [4] знаходимо $n = 0,22$

Підставимо всі значення у формулу для визначення світлового потоку **F**:

Для освітлення вибираємо люмінесцентні лампи типу ЛБ40-1, світловий потік яких $F = 4320 \text{ Лк}$.

Розрахуємо необхідну кількість ламп за формулою:

N – обумовлений число ламп;

F – світловий потік, $F = 33750 \text{ Лм}$;

F_л – світловий потік лампи, $F_{л} = 4320 \text{ Лм}$.

При виборі освітлювальних приладів використовуємо світильники типу ОД. Кожен світильник комплектується двома лампами.

Розрахунок рівня шуму

Одним з несприятливих факторів виробничого середовища в ІОЦ є високий рівень шуму, створюваний друкованими пристроями, обладнанням для кондиціонування повітря, вентиляторами систем охолодження в самих ЕОМ.

Для вирішення питань про необхідність і доцільність зниження шуму необхідно знати рівні шуму на робочому місці оператора.

Рівень шуму, що виникає від декількох некогерентних джерел, що працюють одночасно, підраховується на підставі принципу енергетичного підсумовування випромінювання окремих джерел [6]:

де L_i – рівень звукового тиску i -го джерела шуму;

n – кількість джерел шуму.

Отримані результати розрахунку порівнюється з допустимим значенням рівня шуму для даного робочого місця. Якщо результати розрахунку вище допустимого значення рівня шуму, то необхідні спеціальні заходи щодо зниження шуму. До них відносяться: облицювання стін і стелі залу звукопоглинальними матеріалами, зниження шуму в джерелі, правильне планування обладнання і раціональна організація робочого місця оператора.

Рівні звукового тиску джерел шуму, що діють на оператора на його робочому місці представлені в табл. 5.6.

Таблиця 5.5 – Рівні звукового тиску різних джерел

Джерело шуму	Рівень шуму, дБ
Жорсткий диск	40
Вентилятор	45

Монітор	17
Клавіатура	10
Принтер	45
Сканер	42

Зазвичай робоче місце оператора оснащено наступним обладнанням: вінчестер в системному блоці, вентилятор (и) систем охолодження ПК, монітор, клавіатура, принтер і сканер.

Підставивши значення рівня звукового тиску для кожного виду обладнання у формулу, отримаємо:

$$L_{\Sigma} = 10 \cdot \lg (10^4 + 10^{4,5} + 10^{1,7} + 10^1 + 10^{4,5} + 10^{4,2}) = 49,5 \text{ дБ}$$

Отримане значення не перевищує допустимий рівень шуму для робочого місця оператора, рівний 65 дБ (ГОСТ 12.1.003-83). І якщо врахувати, що навряд чи такі периферійні пристрої як сканер і принтер будуть використовуватися одночасно, то ця цифра буде ще нижчою. Крім того при роботі принтера безпосередню присутність оператора необов'язково, тому що принтер обладнаний механізмом автоматичної подачі аркушів.

Висновки до розділу 5

В спеціальному розділі було розглянуто поняття охорони праці та його загальні поняття. Описано такі важливі теми під час роботи програміста:

- характеристика умов праці програміста;
- забарвлення і коефіцієнти віддзеркалення;
- освітлення;
- параметри мікроклімату;

- шум і вібрація;
- електромагнітне і іонізуюче випромінювання;
- режим праці;
- розрахунок освітлення;
- розрахунок рівня шуму.

ВИСНОВКИ

У ході виконання бакалаврській кваліфікаційної роботи було розроблено повноцінну інформаційну систему онлайн платформи торгових операцій за допомогою Shopify.

В результаті виконання були виконання наступні основні завдання:

- опанування теорій щодо створення інтернет-магазинів і аналізування їх недоліків;
- порівняння видів і вимог для створення інтернет-магазину;
- опис реалізації елементів контенту сайту;
- розробка динамічних веб-сторінок на основі онлайн платформи, якими можна керувати в адмін-панелі.

В першому розділі розглянуто:

- схема роботи інтернет-магазину;
- класифікація інтернет-магазинів за видами діяльності;
- задачі до бакалаврської кваліфікаційної роботи;
- огляд та порівняння вже існуючих розробок;
- вибір CMS-системи та їх порівняння.

В другому розділі розглянуто:

- основні вимоги системи Google до структури сайту;
- поняття структури сайту, їх класифікація;
- створення власної схеми структури сайту, яка була використана у розробці інформаційної системи.

В третьому розділі розглянуто:

- створення та налаштування теми в адмін-панелі Shopify перед початком розробки;
- створення локального середовища для ініціалізації проекту та написання базового коду для швидкого старту роботи;
- вибір та опис стеку інструментів для розробки веб-платформи.

В четвертому розділі розглянуто:

– опис програмної реалізації та підходів розробки відносно компонентів у застосунку.

В спеціальному розділі розглянуто:

- загальні питання охорони праці;
- основні правила роботи для професії програміста;
- головні правила для подальшої безпеки під час праці.

Кінцевим результатом виконання є повністю розроблена інформаційна система для подальшого динамічного наповнення контентом. В наш час, якщо брати до уваги те, що мережа Інтернет постійно розвивається – майже всю продукцію можна придбати в онлайн форматі за допомогою звичайного комп'ютеру та телефону, тому попит на різні інтернет-магазини і сервіси тільки набирає обертів, в майбутньому має гарну можливість на ще більший зріст.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Цифрові технології для всіх : веб-сайт. URL: <https://mind.ua/openmind/20236386-cifrovi-tehnologiyi-dlya-vsikh-yak-zmenshiti-rozriv>.
2. Бородаєв Д. В. Веб-сайт як об'єкт графічного дизайну: Монографія. 2006.
3. Комісарова І. Інтернет-магазин від А до Я. 2011.
4. ТОП-10 безкоштовних CMS для інтернет-магазину. URL: <https://project-seo.net/uk/blog-uk/10-populyarnyh-cms-dlya-internrt-magazyniv/>.
5. Дикан А. Клієнтське SEO. 2015.
6. Правильна структура веб сайту під SEO: приклади, види і рекомендації з розробки структури. БМВ. URL: <https://www.bmb.com.ua/2021/02/seo.html>.
7. Shopify Liquid Tutorial: How to Edit Shopify Theme Code. Create Exceptional Ecommerce Experiences | Shogun. URL: <https://getshogun.com/learn/shopify-liquid-tutorial>.
8. Элизабет Р. Изучаем HTML, XHTML и CSS 2-е изд. 2013.
9. Meyer E. A., Weyl E. CSS: The Definitive Guide: Visual Presentation for the Web. O'Reilly Media, 2017. 1090 p.
10. You Don't Know JS: Up & Going. O'Reilly UK Ltd., 2015. 72 p.
11. Sass Guidelines – Russian translation. Sass Guidelines. URL: <https://sass-guidelin.es/ru/>.
12. Macrae C. Vue.js: Up and Running: Building Accessible and Performant Web Apps. O'Reilly Media, 2018. 174 p.
13. McFarland D. S. JavaScript & jQuery: The Missing Manual. O'Reilly Media, 2014. 686 p.
14. Introduction. Bootstrap · The most popular HTML, CSS, and JS library in the world. URL: <https://getbootstrap.com/docs/4.1/getting-started/introduction/>.

15. Штрауб Б. Git для професіонального програміста.
16. Swiper - The Most Modern Mobile Touch Slider. Swiper. URL: <https://swiperjs.com/>.
17. Koutifaris A. Vuex Quick Start Guide: Centralized State Management for your Vue.js applications. Packt Publishing, 2018. 152 p.
18. Закон України “Про охорону праці” / Законодавство України про охорону праці. - К. Нова редакція 2002 р.
19. Дубовці В.А. Безпека життєдіяльності. / Учеб. посібник для дипломників. - К.: вид. Кірпи, 1992.
20. Мотузко Ф.Я. Охорона праці - М.: Вища школа, 1989. - 336с.
21. Безпека життєдіяльності. / Под ред. Н.А. Белова - М.: Знання, 2000 - 364с.
22. Самгін Е.Б. Освітлення робочих місць. - М.: МІРЕА, 1989. - 186с.
23. Довідкова книга для проектування електричного освітлення. / Под ред. Г.Б. Кнорринга. - Л.: Енергія, 1976.
24. Боротьба з шумом на виробництві: Довідник / Є.Я. Юдін, Л.А. Борисов; За заг. ред. Є.Я. Юдіна - М.: Машинобудування, 1985. - 400с., Іл.
25. Зінченко В.П. Основи ергономіки. - М.: МГУ, 1979. - 179с.