

Київський національний торговельно-економічний університет
Кафедра інженерії програмного забезпечення та кібербезпеки

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ ЦИФРОВИХ ГАДЖЕТІВ

Студента 4 курсу, 6 групи,
спеціальності 121 «Інженерія
програмного забезпечення»

підпис студента

Максимовича Данила
Олександровича

Науковий керівник,
кандидат технічних наук,
доцент

підпис керівника

Цензура Микола
Олександрович

Гарант освітньої програми
кандидат технічних наук,
доцент

підпис керівника

Цензура Микола
Олександрович

КИЇВ 2021

Київський національний торговельно-економічний університет

Факультет інформаційних технологій

Кафедра інженерії програмного забезпечення та кібербезпеки

Освітній ступінь бакалавр

Спеціальність 121 Інженерія програмного забезпечення

Затверджую

Зав. кафедри інженерії
програмного
забезпечення
та кібербезпеки
Криворучко О.В.
"20" жовтня 2020 р.

Завдання на випускню кваліфікаційну роботу студентів

Максимовичу Данилу Олександровичу

(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи Розробка інтернет-магазину цифрових гаджетів

Затверджена наказом ректора від "30" жовтня 2020 р. № 3225

2. Строк здачі студентом закінченої роботи

3. Цільова установка та вихідні дані до роботи

Мета роботи розробка інтернет-магазину з продажу цифрових гаджетів за допомогою мови гіпертекстової розмітки HTML, каскадної таблиці стилів CSS та препроцесора SCSS, розробка функціоналу вебдодатку за допомогою JavaScript фреймворка ReactJS з використанням бази даних, розробленою за допомогою Microsoft SQL Server та мови SQL. Дана система буде працювати в рамках певного підприємства і виконувати функції основного майданчика для ведення комерційної діяльності.

Об'єкт дослідження інтернет-магазин цифрових гаджетів "eStore"

Предмет дослідження розробка інтернет-магазину цифрових гаджетів "eStore"

4. Консультанти роботи із зазначенням розділів, які консультують:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв

5. Зміст випускної кваліфікаційної роботи (перелік питань за кожним розділом)

ВСТУП

РОЗДІЛ 1. ЗАГАЛЬНІ ВІДОМОСТІ

1.1 Поняття інтернет-магазину

1.2 Відомості про бренд

1.3 Технічне завдання

1.4 Висновок до розділу

РОЗДІЛ 2 ПРОЕКТУВАННЯ ІНТЕРНЕТ-МАГАЗИНУ

2.1 Архітектура веб-додатку

2.2 Структура файлів проекту

2.3 Структура регулювання інтернет магазину

2.4 Послідовність оформлення замовлення

2.5 Структура замовлення

2.6 Схема роботи клієнта з інтернет-магазином

2.7 Висновок до розділу 2

РОЗДІЛ 3 РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ

3.1 Середовище розробки

3.2 Ініціалізація програми

3.3 Розробка інтернет магазину

3.4 Висновок до розділу 3

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

№ пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		за планом	фактично
1	2	3	4
1.	<i>Вибір теми випускної кваліфікаційної роботи</i>	21.09.2020	21.09.2020
2.	<i>Вступ та перелік літературних джерел</i>	14.12.2020	14.12.2020
3.	<i>Розділ 1. «ЗАГАЛЬНІ ВІДОМОСТІ»</i>	19.02.2021	19.02.2021
4.	<i>Розділ 2. «ПРОЕКТУВАННЯ ІНТЕРНЕТ-МАГАЗИНУ»</i>	05.03.2021	05.03.2021
5.	<i>Розділ 3. «РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ ЦИФРОВИХ ГАДЖЕТІВ»</i>	10.04.2021	10.04.2021
6.	<i>Висновки</i>	24.09.2021	24.04.2021
7.	<i>Здача випускної кваліфікаційної роботи на кафедрі (перша перевірка)</i>	14.09.2021	14.05.2021
8.	<i>Підготовка автореферату та презентації доповіді</i>	17.05.2021	17.05.2021
9.	<i>Попередній захист випускної кваліфікаційної роботи</i>	18.05.2021-21.05.2021	
10.	<i>Зовнішнє рецензування випускної кваліфікаційної роботи</i>	01.06.2021	
11.	<i>Здача прошого випускної кваліфікаційної роботи на кафедрі</i>	02.06.2021	
12.	<i>Публічний захист випускної кваліфікаційної роботи</i>		

7. Дата видачі завдання «20» жовтня 2020 р.

8. Науковий керівник випускної кваліфікаційної роботи

Цензура М. О.

(прізвище, ініціали, підпис)

9. Гарант освітньої програми

Цензура М. О.

(прізвище, ініціали, підпис)

10. Завдання прийняв до виконання студент

Максимович Д. О.

(прізвище, ініціали, підпис)

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There is no text or other markings on the paper.

Відмітка про попередній захист _____ (ПІБ, підпис, дата)

Випускна кваліфікаційна робота студента Максимовича Д. О.
(прізвище, ініціали)

«_____» _____ 2021 г.

АНОТАЦІЯ

Дипломна робота на тему «РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ ЦИФРОВИХ ГАДЖЕТІВ» містить 41 сторінок, 22 рисунків та 5 додатків.

Метою випускної кваліфікаційної роботи є розробка інтернет-магазину з продажу цифрових гаджетів за допомогою мови гіпертекстової розмітки HTML, каскадної таблиці стилів CSS та препроцесора SCSS, розробка функціоналу вебдодатку за допомогою JavaScript фреймворка ReactJS з використанням бази даних, розробленою за допомогою Microsoft SQL Server та мови SQL. Дана система буде працювати в рамках певного підприємства і виконувати функції основного майданчика для ведення комерційної діяльності.

Об'єктом дослідження роботи є інтернет-магазин цифрових гаджетів “eStore”.

Предметом дослідження випускної кваліфікаційної роботи є процес розробки веб-додатку, що складається з двох частин : бази даних та веб-сторінки.

У першому розділі проаналізовано існуючі інтернет-магазини, їх переваги та недоліки, а також розглянемо найефективніший спосіб торгівлі через інтернет. У даному розділі також створено компанію “eStore”, відібрано асортимент товарів даного магазину та його постачальників. Розроблене технічне завдання для реалізації інтернет-магазину.

У другому розділі створено архітектуру додатку: проектування веб-додатку, структура файлів, структура регулювання, послідовність оформлення замовлення, структура замовлення, схема взаємодії клієнта з інтернет-магазином. Завдяки чому був створений та підготовлений до активної стадії розробки проект, що дозволило почати розробку.

У третьому розділі описано послідовну розробку програмного забезпечення для продажу цифрових гаджетів засобами фреймворку React, та препроцесора SCSS з використанням допоміжних бібліотек.

ANNOTATION

Diploma work on the theme "Development of an online store of digital gadgets" contains 41 pages, 22 rails and 5 applications.

The purpose of graduation work is a development online store for the sale of digital gadgets for helping language hypertext marking HTML, cascading SCS Styles and Preprocessor SCSS, web development functionality for help JavaScript Framework ReactJs with used to quench the Microsoft SQL Server software. . This system will be labor in the framework of a certain enterprise and perform the functions of the main site for commercial activity.

Object of research of the Yenternet-store of digital gadgets "Estore".

The subject of distribution graduation qualification work is the development process of web application, which makes up with documents: databases and web pages.

The first section analyzes existing online stores, their advantages and disadvantages, and will consider inefficient trade of trading over the Internet. In this section, altogether, the company ostore "is also a product assortment of the goods of this store and its suppliers. The technical task for the sale of an online store has been developed.

In the second section, an application architecture is created: Web application design, file structure, adjustment structure, order sequence, order structure, customer interaction scheme from the online store. Due to which it was created and developed to the active stage of development project, which allowed to start developing.

The third section describes a consistent development software for sale of cifflass gadgets React frames, and Preprocessor SCSS from used auxiliary libraries.

ЗМІСТ

ВСТУП.....**ERROR! BOOKMARK NOT DEFINED.**

РОЗДІЛ 1. ЗАГАЛЬНІ ВІДОМОСТІ**ERROR! BOOKMARK NOT DEFINED.**

1.1 Поняття інтернет-магазину**Error! Bookmark not defined.**

1.2 Відомості про бренд..... 8

1.3 Технічне завдання 11

1.4 Висновок до розділу 1 15

РОЗДІЛ 2 ПРОЕКТУВАННЯ ІНТЕРНЕТ-МАГАЗИНУ 17

2.1 Архітектура веб-додатку.....**Error! Bookmark not defined.**

2.2 Структура файлів проекту 19

2.3 Структура регулювання інтернет магазину 22

2.4 Послідовність оформлення замовлення. 23

2.5 Структура замовлення 23

2.6 Схема роботи клієнта з інтернет-магазином ... **Error! Bookmark not defined.**

2.7 Висновок до розділу 2..... 25

РОЗДІЛ 3 РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ 26

3.1 Середовище розробки 26

3.2 Ініціалізація програми..... 27

3.3 Розробка інтернет магазину..... 28

3.4 Висновок до розділу 3..... 38

ВИСНОВКИ ТА ПРОПОЗИЦІЇ..... 39

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 41

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка інтернет-магазину цифрових гаджетів	Стадія	Аркуш	Аркушів
Зав. каф.	Криворучко О.В.					Зміст	2	42
Керівник	Цензура М.О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М.О.							
Розроб	Максимович Д.О.							
					ЗМІСТ			

ВСТУП

Актуальність:

За останніми даними аудиторія в інтернеті стрімко зростає, а продажі через інтернет у великих містах досягають до 25%, при цьому фахівці підкреслюють тенденцію до зростання продажів саме через інтернет. Щорічно кількість інтернет-магазинів збільшується, так як це дійсно вигідно і зручно для покупця, не кажучи про економії бюджету і часу. Інтернет-магазин працює цілодобово і може продавати певні товари в автоматичному режимі без участі продавця. До переваг так само можна віднести те, що не треба закуповувати товар заздалегідь, а це істотна економія, на складських приміщеннях. Потрібно лише домовитися з постачальниками, і в потрібний момент, просто викупити товар, який у вас замовлять. У порівнянні зі звичайним магазином, територія продажів якого обмежується населенням міста або району, територія охоплення інтернет-магазину збільшується на всю Україну і українську аудиторію в інших країнах, адже товар можна доставляти не тільки кур'єрською службою, але і поштою.

На даний момент існує багато інтернет-магазинів цифрових гаджетів. У кожного з них є красиве оформлення, достатньо багатий вибір різноманітних гаджетів та аксесуарів. Але в них у них є достатньо мінусів:

1. Постійна реклама (при вході на сторінку, на задньому фоні сторінки, при натисканні на будь-яку невідому кнопку);
2. Ціни на гаджети частіше за все перебільшені;
3. Недостатня інформація про товар;
4. Більшість сайтів неоптимізована, через що користувач змушений довго чекати завантаження сайту або навіть спостерігати пусту сторінку.

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата				
Зав. каф.	Криворучко О.В				Розробка інтернет-магазину цифрових гаджетів	Стадія	Аркуш	Аркушів
Керівник	Цензура М.О.					В	3	42
Гарант	Цензура М.О.					Факультет інформаційних технологій, 4 курс, 6 група		
Розроб	Максимович Д.О.							
					ВСТУП			

Так як в час розповсюдження коронавірусу інтернет-магазини стали найпопулярнішим способом купівлі товарів для різних сфер життя, тема, обрана автором, є досить актуальною.

Метою випускної кваліфікаційної роботи є розробка інтернет-магазину з продажу цифрових гаджетів за допомогою мови гіпертекстової розмітки HTML, каскадної таблиці стилів CSS та препроцесора SCSS, розробка функціоналу вебдодатку за допомогою JavaScript фреймворка ReactJS з використанням бази даних, розробленою за допомогою Microsoft SQL Server та мови SQL. Дана система буде працювати в рамках певного підприємства і виконувати функції основного майданчика для ведення комерційної діяльності.

Об'єктом дослідження роботи є інтернет-магазин цифрових гаджетів “eStore”.

Предметом дослідження випускної кваліфікаційної роботи є процес розробки веб-додатку, що складається з двох частин : бази даних та веб-сторінки.

Завданням дослідження даної роботи є:

- Розробка інтерфейсу користувача на JSX та SCSS;
- Знайомство з перевагами використання фреймворка ReactJS;
- Використання вмінь по розробці функціоналу даного фреймворка;
- Підключення бази даних до додатку.

РОЗДІЛ 1

ЗАГАЛЬНІ ВІДОМОСТІ

1.1 Теоретичні дані

1.1.1 Поняття інтернет-магазину

Електронний магазин – це прикладна система, побудована з використанням технології системи електронної торгівлі. Подібно до звичайного магазину, електронний магазин реалізує наступні основні функції: представлення товарів (послуг) покупцю, обробку замовлень, продаж і доставку товарів.

Основні проблеми реалізації інтернет-магазину лежать на стику технологій Інтернету і традиційної комерційної діяльності. У звичайній торгівлі покупець звик до того, що є можливість оцінити товар візуально, визначити його якість і характеристики. В електронній торгівлі він такої можливості позбавлений. Найчастіше візуальної інформації досить, але тут вступають в дію емоційні і психологічні чинники. Часто існують проблеми з доставкою товарів, особливо якщо їх ціна невелика.

1.1.2 Переваги та недоліки віртуальних магазинів

Переваги віртуальних магазинів

Інтернет-магазин багато в чому виграє перед реальним магазином. Зменшується чисельність персоналу за рахунок скорочення обсягу взаємодії з клієнтами, оренда дискового простору і розміщення «електронної вітрини» дешевше і простіше оренди торговельних приміщень і розміщення товарів на полицях, немає потреби в касовому обслуговуванні і т.д.

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка інтернет-магазину цифрових гаджетів	Стадія	Аркуш	Аркушів
Зав. каф.	Криворучко О.В					РІ	5	42
Керівник	Цензура М.О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М.О.							
Розроб	Максимович Д.О.				ЗАГАЛЬНІ ВІДОМОСТІ			

Основні переваги Інтернет-магазинів для споживачів – **економія часу**.

Людині яка працює 6 днів на тиждень з 10 до 19,шноді не буває часу щоб сходити в магазин. Інтернет-магазин дозволяє зробити покупку не виходячи з офісу в будь-який час, а вибір і замовлення товару займе у нього кілька хвилин, якщо він точно знає, що хоче купити. Служба доставки інтернет-магазину доставить обраний товар у зручний час і місце. Крім цього вибір і оцінка властивостей товару відбувається в інтернет-магазині набагато швидше і зручніше ніж в звичайному магазині. Уявіть, що ви прийшли в звичайний магазин побутової техніки і хочете порівняти параметри 20-30 телефонів. Для цього ви повинні вивчити всі ціnnики: запам'ятати характеристики, ціни і назви моделей. Можна звернутися до продавця-консультанта, якщо він не зайнятий, з проханням розповісти про всі ці товари. Але зазвичай жоден продавець не має стільки вільного часу для роботи з одним клієнтом, і немає гарантії, що він володіє всією інформацією.

В Інтернет-магазині можна необмежено збільшувати асортимент і інформативність.

Крім того асортимент інтернет-магазину нічим не обмежений (як, наприклад, асортимент звичайного магазину обмежений площею торгового павільйону). А якщо передбачений пошук по параметрам, то Ви просто можете вказати характеристики, яким повинен відповідати товар, і потім вже вибирати зі списку моделей, що задовольняють вашому запиту. Крім того, інтернет-магазин в змозі видати таку інформацію, як купівельний рейтинг, поради та відгуки про товар, статті про товар, які можуть зумовити Ваш вибір.

Наступний плюс **економія грошей**.

Витрати на роботу інтернет-магазину, включаючи доставку, істотно нижче ніж у звичайного. На відміну від звичайного магазину інтернет-магазин здатний обслужити кілька сотень клієнтів одночасно. Хоча на практиці такого не зустрінеш. Крім того, якщо покупець проживає в іншому місті, він отримує

можливість не витрачатися на міжміські телефонні консультації. Всю інформацію він може знайти на сторінках інтернет-магазину.

До недоліків інтернет-магазинів для покупця ставитися:

- не можна "помацати", не можна дізнатися більше, ніж написано (приклад: меблі, одяг);
- проблеми гарантії, супроводу;
- часто довга доставка (до 6 тижнів), особливо якщо ми говоримо про міжнародні доставки.

Головний недолік електронної торгівлі – це неякісні товари. Так, дослідження, проведені на початку 2010р. англійською фірмою ArmorGroup, показали, що 35% дорогих товарів, що продаються через магазини, що працюють в Інтернет, є підробками.

1.1.3 Класифікація віртуальних магазинів

Класифікувати віртуальні магазини можна за різними критеріями. Найважливішою класифікацією є класифікація за моделлю бізнесу:

- Онлайн магазин;
- Поєднання офлайнового бізнесу з онлайнним (коли інтернет-магазин був створений на основі вже діючої реальної торгової структури).

По відношенню з постачальниками:

- мають власний склад (наявність реальних товарних запасів)
- працюють за договорами з постачальниками (відсутність значних власних запасів).

Класифікація за товарним асортиментом - книги, аудіо, відеокасети, CD, DVD, комп'ютерна, побутова техніка, цифрові гаджети.

Серед методів роздрібного продажу товарів у Мережі можна виділити:

- Інтернет-магазини (автоматичні магазини);
- Web-вітрини;
- Торгові автомати.

Інтернет-вітрина – швидше це рекламний сервер. На вітрині викладають інформацію про товари, яку постійно оновлюють. Витрати на її створення та адміністрування можуть бути досить низькими, а практична користь такої вітрини очевидна. Але це ще не торгівля. Потенційний покупець, відвідавши вітрину, повинен зателефонувати на фірму, оплатити товар, домовитися про доставку. Тому Інтернет-вітрина виправдана в тих випадках, коли покупця треба познайомити зі складною продукцією, на вивчення якої в торговому залі у нього піде дуже багато часу.

Торговий автомат може не тільки виконувати функції вітрини, але і приймати замовлення і передавати їх менеджеру, тобто оформляти замовлення і виписувати рахунки на оплату без присутності покупця.

Автоматичний магазин - ефективне і комплексне рішення в торговому бізнесі. Він не тільки виписує рахунки, а й відстежує замовлення, приймає електронні платежі і формує заявки на доставку товарів покупцям. Тут завдання менеджера – контролювати роботу системи, складну в обслуговуванні.

1.2 Відомості про бренд

1.2.1 Назва інтернет магазину

При виборі назви інтернет магазину слід враховувати такі фактори:

- Назва не має бути плагіатом;
- Вона має бути креативною;
- Назва має добре запам'ятовуватися;

Провівши дослідження ринку та визначившись з назвами компаній конкурентів, було виділено декілька варіантів для назви магазину, серед яких було обрано одну найкращу – «eStore».

1.2.2 Асортимент товарів

Спектром продажу товарів інтернет-магазину – це цифрові гаджети. Але це будуть не прості комп'ютери та ноутбуки, а ігрові, тобто оснащені потужними

модулями для якісної та тривалої гри. На сьогоднішній день є досить багато ігрових комп'ютерів, але не кожен з них якісний та спроможний служити довго.

Серед ігрових ноутбуків які можуть задовольнити потребу клієта, на сучасному ринку добре показують себе такі фірми як "ASUS", "Acer" та деякі моделі "HP" та "Lenovo". Саме такі моделі будуть представлені на вітрині інтернет магазину.

Серед ігрових комп'ютерів важко виділити найкращий по всім критеріям бренд, тому що всі вони достатньо потужні та якісні. Тому в інтернет магазині будуть представлені комп'ютери різноманітних фірм, таких як: "ARTLine", "Everest", "Cobra" та "QUBE".

Також магазин зможе знайти прихильників серед клієнтів що бажають придбати собі якісний та потужний телефон від всесвітньовідомого бренду "Apple". Автор був користувачем різноманітних смартфонів: Lenovo, Sony, Nokia. Але, спробувавши техніку від Apple, ніякий телефон, на думку автора, не зможе працювати так як iPhone.

Отже асортимент товару в інтернет-магазині буде наступний:

- Ігрові комп'ютери;
- Ігрові ноутбуки;
- Телефони;

1.2.3 Унікальність

Кожен з всесвітньовідомих підприємств, незалежно від сфери його діяльності, приваблює своїх клієнтів однією серед трьох переваг, таких як:

- Якість;
- Ціна;
- Зручність;

Наприклад:

Компанія *MacDonald's* виділяється **зручністю** серед інших ресторанів фаст фуду. По-перше це мережа ресторанів, яка має філіали в усьому світі у

					КНТЕУ 121 06-15.БР		Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата			9

кожному великому місті. Так у Києві, де б ви не знаходилися, MacDonald's завжди десь поряд. По-друге в кожному з ресторанів дуже розвинений менеджмент: прийшовши в заклад рідко можна помітити великі черги або довгий час оформлення замовлення. Також на території закладу є можливість замовити їжу власноруч через електронний автомат, або є послуга MacDrive що не потребує відвідування закладу, що зекономить ваш час.

AliExpress приваблює клієнтів своєю **ціною** за товари. В магазині можна придбати речі за сміхотворною ціною та вибір товару дуже великий. Але якість цих товарів сумнівна. Та зі зручністю справа така сама: чекати замовлення можна понад місяць. Та саме через привабливу ціну за товари *AliExpress* має прихильників в багатьох країнах по всьому світу.

А ось безсумнівною **якістю** своєї техніки може похизуватися Apple. Компанія з початку виходу на світовий ринок одразу знайшла прихильників саме через якість товару. Виходячи зі статистики, серед 100% лише 6% техніки виходять з ладу не за провиною користувача у перший рік користування. Так, ціна за телефон значно вища порівнюючи з іншими брендами та фірмові модулі для заміни при поломці теж недешеві. Також, маючи телефон досить старої моделі, Apple перестає підтримувати оновлення операційної системи, яка не дає підтримку використання більшості програм. Що змушує користувача купити техніку новішої версії. Але, незважаючи на це, Apple – це найпопулярніший постачальник цифрових гаджетів в усьому світі.

Якщо одна з вищеперерахованих компаній вирішила б зробити вибір на користь іншої переваги, або навіть схрестити дві одразу, вона б потерпіла крах майже одразу.

Інтернет-магазин *eStore* буде мати таку перевагу над іншими магазинами як **якість**. На думку автора, така річ як техніка має бути в першу чергу надійною. Гаджети в наш час – це та річ без якої не можливо працювати, вчитися та підтримувати комунікацію, саме тому зараз без надійної та якісної техніки – як без рук.

1.2.4 Постачання

Так як основною перевагою компанії eStore буде якість, товари мають бути суворо надійними та обов'язково оригінальними. Тому постачальниками техніки на вітрину інтернет-магазину будуть лише компанії виробники відповідного товару. Через те що більшість техніки яка буде на вітринах виробляється в Америці, слід враховувати ціни на доставку та розмитнення. Тому ціни за одиницю товару будуть більші ніж в інших магазинах. Магазин буде надавати ліцензії від компанії виробника, через що клієнт буде впевнений що гаджет оригінальний. Також при несправності гаджет буде переправлений до сервісного центру компанії виробника.

Отже, скориставшись послугами нашого магазину, клієнт зрозуміє: *щовпевненість у власній техніці значно важливіша за ціну.*

1.3 Технічне завдання

1.3.1 Предмет розробки

Предметом розробки є інтернет-магазин компанії eStore, з системою динамічного відображення та пошуку цифрових гаджетів.

Призначення сайту:

- Представлення інформації про компанію та її діяльність;
- Відображення товарів та інформації про них;
- Ведення торгівлі використовуючи інтернет портал;

Ціль інтернет-магазину – прямі продажі, однак сайт також буде використовуватися формування іміджу компанії в очах цільової аудиторії і спілкування з клієнтами.

1.3.2 Структура сайту

Вибір техніки – це складно, по-перше матеріально, по-друге не кожен клієнт має достатньо знань щоб обрати дійсно гідний гаджет. Тому структура сайту повинна бути лаконічна та зрозумілою для користувача. Тобто він не

має «загубитися» на сайті. Щоб уникнути таких випадків треба продумати саме таку структуру (Рис. 1.1).

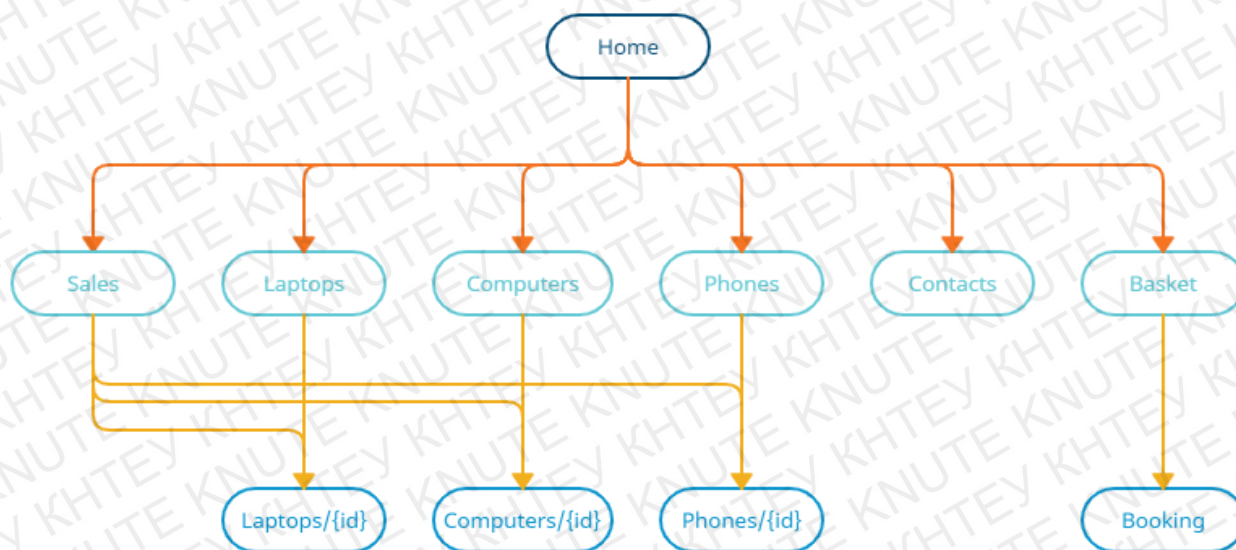


Рис. 1.1. Структура інтернет-магазину eStore

1.3.3 UXдизайн сайту

UX дизайн (User Experience) – це те, який досвід або враження отримує користувач від роботи з вашим інтерфейсом. Простими словами – це структура самої веб-сторінки без графічно красиво відображених елементів. На основі UX дизайну створюється UI дизайн (Рис. 1.2).



Рис. 1.2. UX дизайн сайту

1.3.4 UI дизайн сайту

UI дизайн (User Interface) – це те, як виглядає інтерфейс і те, які фізичні характеристики набуває. Тобто, використовуючи графічні редактори (PhotoShop / Figma), створення візуального шаблону сайту (як він має виглядати в браузері) (Рис. 1.3).

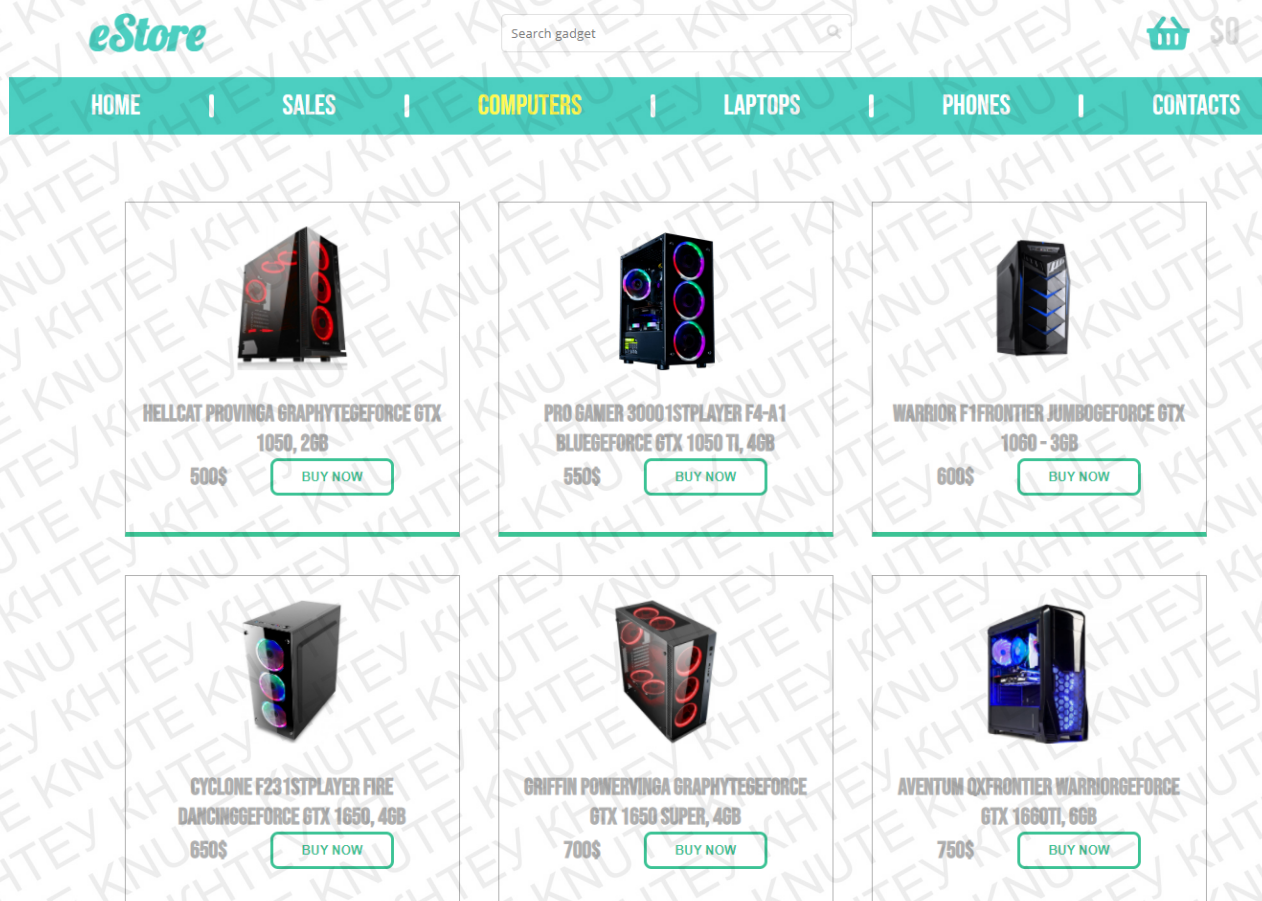


Рис. 1.3. UI дизайн сайту

1.3.5 Технічні вимоги

Перелік технічних вимог:

- *Адаптивна верстка.* Сайт має виглядати красивим як у десктопній версії, так і в мобільній. Усі блоки не мають вилазити за межі сторінки та не повинно бути зайвих відступів між ними. Також слід попрацювати над текстом, щоб він не був занадто великим або навпаки малим для обох версій;

- *Система пошуку по сайту.* Щоб користувач за бажанням міг ввести назву товару у поле для вводу та, натиснувши кнопку, побачити список знайдених товарів;
- *Додавання товару у кошик.* На кожній сторінці з детальною інформацією про товар має бути кнопка «Купити», при натисканні на яку, відповідний товар має бути доданий у сховище;
- *Видалення товару з кошика.* На сторінці «кошика» додані товари мають динамічно відображатися з інформацією про товар а також кнопкою «видалити з кошика», при натисканні на яку, товар має видалятися зі сховища та з самого кошика.
- *Оформлення заказу.* На сторінці кошика у кінці списку добавлених товарів має бути кнопка «оформити заказ», при натисканні на яку, користувача має перевести на сторінку з формою для заповнення. Поля для вводу мають бути валідовані, щоб користувач вводив лише правдиву інформацію. Після натискання кнопки «оформити» форма з даними: ім'я, прізвище, номер телефону, адрес пошти для доставки а також масив із товарів, вибраних користувачем відправляється на сервер для обробки. Після відправки валідної форми має прийти повідомлення у вигляді мольного вікна що заказ оформлений успішно та після цього список добавлених товарів у кошику самоочищається.
- *Слайдер, плавні переходи між сторінками, анімації.* Для того щоб клієнту було цікаво відвідувати сайт, треба додати анімації. Слайдер добре приваблює користувачів своєю анімацією, що сприяє покупці товару. Також при завантаженні сторінки всі блоки мають плавно з'являтися на сайті, що неабияк порадує користувача.

1.3.6 Зміст сайту

Дані сайту будуть динамічно підтягуватись з бази даних, яка буде розроблена розробником. Дані в базі даних будуть взяті з офіційної

характеристики відповідного товару, взятої з онлайн ресурсу компаній-виробників.

1.3.7 Етапи створення сайту

- Створення концепції інтернет-магазину, розробка та узгодження технічного завдання;
- Розробка макета сайту, що включає всі графічні і інтерактивні елементи;
- Програмування та підключення модулів управління;
- Підготовка контенту - створення, оптимізація, узгодження.
- Тестування сайту, при необхідності - внесення корективів;
- Запуск.

1.4 Висновок до розділу 1

Отже, віртуальний магазин – ефективний спосіб продажу будь-яких видів товарів. Існує три основних видів віртуальних магазинів: інтернет-магазин, веб-вітрина, торговий автомат. Кожен з них має свої переваги та недоліки, але найбільш зручним для клієнтів та менеджерів компанії – це *інтернет-магазин*, тому що для його функціонування, менеджер не повинен брати участь у заповненні замовлення.

У даному розділі було визначена назва магазину – eStore, його переваги та асортимент товару. На сьогоднішній день без цифрових гаджетів не можливо вчитися, працювати, підтримувати комунікацію один з одним, саме тому їх потребує кожен. Основною перевагою компанії являється саме якість, тому що надійність у власному гаджеті – це першочергова необхідність. Постачальниками товарів для магазину будуть лише компанії виробники відповідної продукції, це знизить відсоток несправності гаджетів та підвищить відсоток задоволених клієнтів.

Також у розділі було постановлене технічне завдання, за яким буде проходити розробка. Була визначена ціль інтернет-магазину:

- Для вивчення асортименту;

- Для отримання інформації про продукцію бренда;
- Для продажу товарів;

Була розроблена структура веб-сайту: легка для сприйняття користувачем та лаконічна. Були визначені та розроблені UX таUI дизайн.

Були перераховані та роз'яснені такі вимоги до функціонування сайту:

- *Адаптивна верстка;*
- *Система пошуку по сайту;*
- *Додавання товару у кошик;*
- *Видалення товару з кошика;*
- *Оформлення заказу;*
- *Слайдер, плавні переходи між сторінками, анімації.*

Також були розписані етапи розробки даного інтернет-магазину для компанії eStore.

РОЗДІЛ 2

ПРОЕКТУВАННЯ ІНТЕРНЕТ МАГАЗИНУ

2.1 Архітектура веб-додатку

Для того, щоб магазин був оптимізований (тобто працював швидко та якісно), а також функціонував правильно треба розробити архітектуру його програмного забезпечення.

Архітектура програмного забезпечення – це сукупність найважливіших рішень про організацію програмної системи. Архітектура включає:

- вибір структурних елементів і їх інтерфейсів, за допомогою яких складена система, а також їх поведінки в рамках співпраці структурних елементів;
- з'єднання обраних елементів структури і поведінки у все більш крупні системи;
- архітектурний стиль, який спрямовує всю організацію – всі елементи, їх інтерфейси, їх співпраця і їх з'єднання.

Документування архітектури програмного забезпечення спрощує процес комунікації між розробниками, дозволяє зафіксувати прийняті проектні рішення і надати інформацію про них експлуатаційного персоналу системи, повторно використовувати компоненти і шаблони проекту в інших.

Правильна архітектура має такі характерні риси:

- ефективність системи;
- гнучкість системи;
- можливість розширення системи;
- масштабування процесу розробки;
- тестування системи;

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка інтернет-магазину цифрових гаджетів	Стадія	Аркуш	Аркушів
Зав. каф.	Криворучко О.В					Р2	17	42
Керівник	Цензура М.О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М.О.							
Розроб	Максимович Д.О.				ПРОЕКТУВАННЯ ІНТЕРНЕТ МАГАЗИНУ			

- повторне використання та простий супровід.

При розробці архітектури інтернет-магазину «eStore», до уваги було взято три головних аспекти: залучення до проекту бази даних для спрощення з роботою інформацією, використання JavaScript фреймворк ReactJS та допоміжних бібліотек (таких як react-router-dom, react-slick, react-helmet і тд.) а також мову гіпертекстової розмітки (HTML) та каскадну таблицю стилів (CSS) (Рис. 2.1).

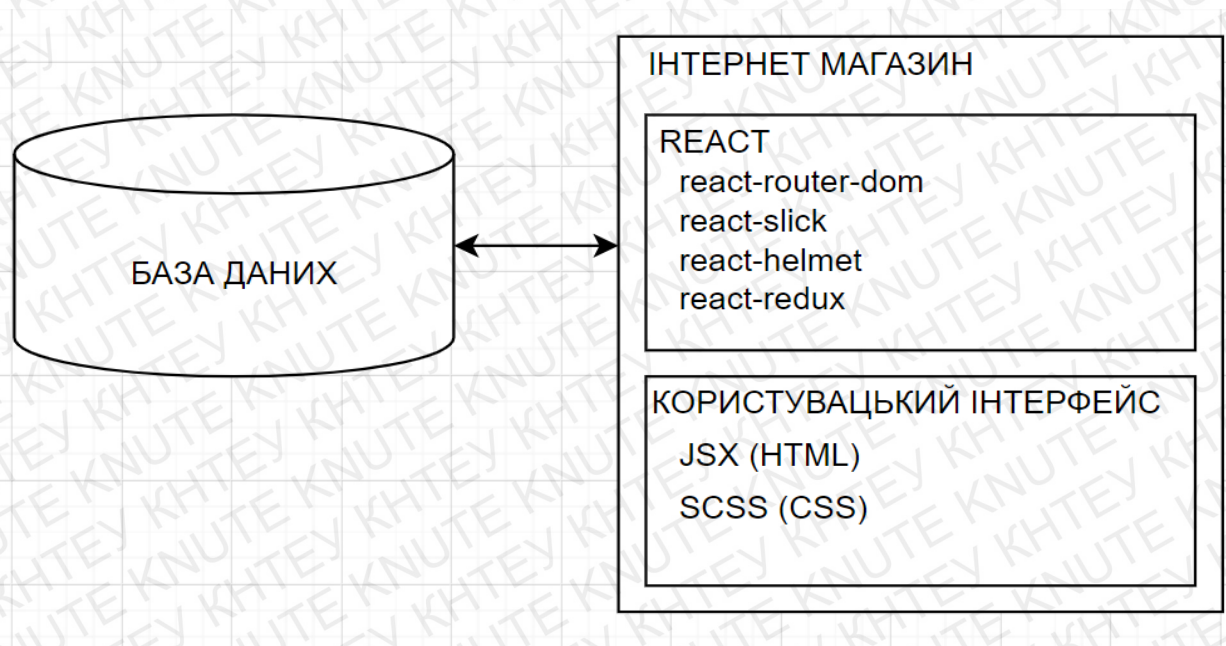


Рис. 2.1 Архітектура інтернет-магазину

В розробці систем, не залежно від її складності, першим із критеріїв якості стоїть доступність. Тобто структура розробленої системи має бути зрозумілою для інших розробників, тому що більшість проектів потребують оновлень та корективів, та дуже часто проекти створюються і редагуються не одним і тим самим розробником.

Одним із кращих способів вирішення великої задачі (розробки інтернет-магазину) – це розбиття її на менші задачі, а їх в свою чергу – на ще менші задачі, і так до тих пір поки задачі не стануть досить простими. Це не лише єдине вірне існуюче рішення, а і універсальне рішення, що вирішує великий спектр задач забезпечуючи пониження складності, воно також забезпечує

гнучкість системи, дає зручні можливості для масштабування та покращує її стійкість.

2.2 Структура файлів проекту

При написанні будь-якого програмного забезпечення, велику увагу слід приділити файловій структурі, адже правильна файлова структура – перша ознака професіоналізму. Бо якщо не дотримуватися правил по її формуванню, то проект перетвориться в безлад, та розібратися в ньому буде важко не тільки іншим розробникам, а ще і самому власнику даного проекту.

Звичайний вебсайт складається з таких складових: .html, .css, .js документи та інші файли такі як картинки (.jpg, .png) та шрифти (.woff, .ttf). Для кожного з них мають бути створені окремі папки, а html файл має лежати у корені папки (сайту). Але структура вебдодатку на React дещо відрізняється від звичайної: кожен об'єкт на сайті – це окремий компонент, який імпортується в головний файл, який вже відображає (рендерить) всі ці помпонки на сторінці. Кожен компонент в свою чергу має дві складові: це js та scss файл. Але логіка, що кожен тип файлів має бути в окремій папці зберігається.

При ініціалізації, фреймворк самостійно створює класичну структуру побудовану на системі зборки проектів Webpack. Ця система передбачає собою компіляцію будь-якого файлу, який не приймає html документ (наприклад scss, sass, less файли – файли які спрощують css розробку), у файл який html підтримує.

Кожен вебсайт починається із HTML файла, він в свою чергу складається з елементів мови розмітки (тегів), кожен з яких має свою роль та використовується в окремих випадках. Та при розробці масштабних проектів (таких як інтернет-магазин, соціальна мережа, блог тощо), тегів стає досить багато, а схожість блоків дуже велика, саме для вирішення таких проблем були створені JavaScript фреймворки: AngularJS розроблений компанією

Google, ReactJS компанією Facebook та VueJS розробником Еваном Ю. Кожен з них має свою специфіку, але вирішують вони одну й ту саму проблему.

Розробка на фреймворку передбачає собою створення окремих компонентів які будуть замінювати десятки, а то і сотні строчок коду. Наприклад у нас на сторінці є такий блок з товаром (Рис. 2.2):

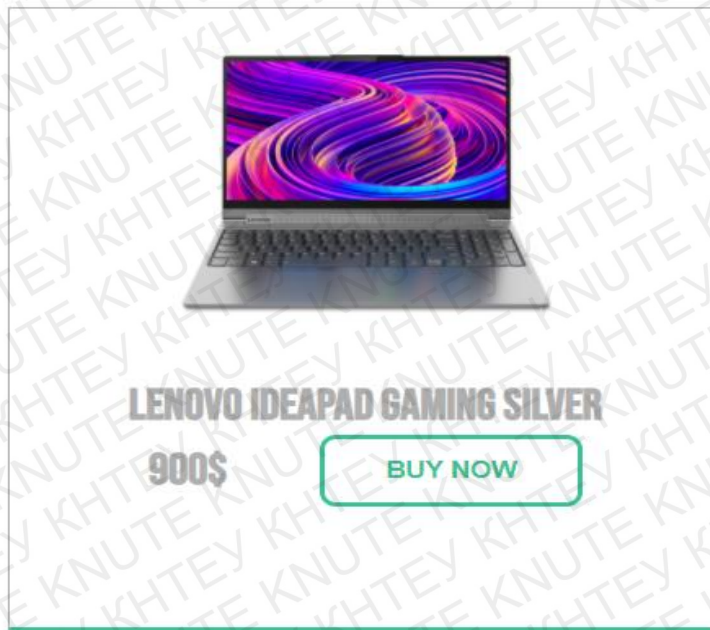


Рис. 2.2 Блок з товаром

Так він буде представлений у вигляді коду:

```
<div class = "product_block">
  <img src = "../img/laptop1.png" alt = "product" />
  <div class = "product_block-content">
    <div>
      <h2> Lenovo Ideapad Gaming Silver </ h2>
    </ div>
    <div className = "product_block-content-button">
      <h2> 900$ </ h2>
      <button class = "button"> BUY NOW </ button>
    </ div>
  </ div>
</ div>
```

Зм.	Архиви	№ докум	Підпис	Дата

КНТЕУ 121 06-15.БР

Архиви

20

Для відображення даного блоку з товаром було використано 12 строчок коду, а якщо на сторінці треба відобразити 100 таких товарів? То вже буде складно оперувати з таким об'ємом коду, не кажучи вже про те що буде якщо потрібно підправити ці блоки. React цю проблему вирішує таким способом:

- Прописуємо цей блок один раз;
- Називаємо його відповідним іменем;
- Викликаємо його як свій власний тег (наприклад `<ProductBlock/>`).

Таким чином веб-додаток буде складатися із компонентів, які будуть лежати в окремих папках, та буде мати таку структуру (Рис. 2.3):

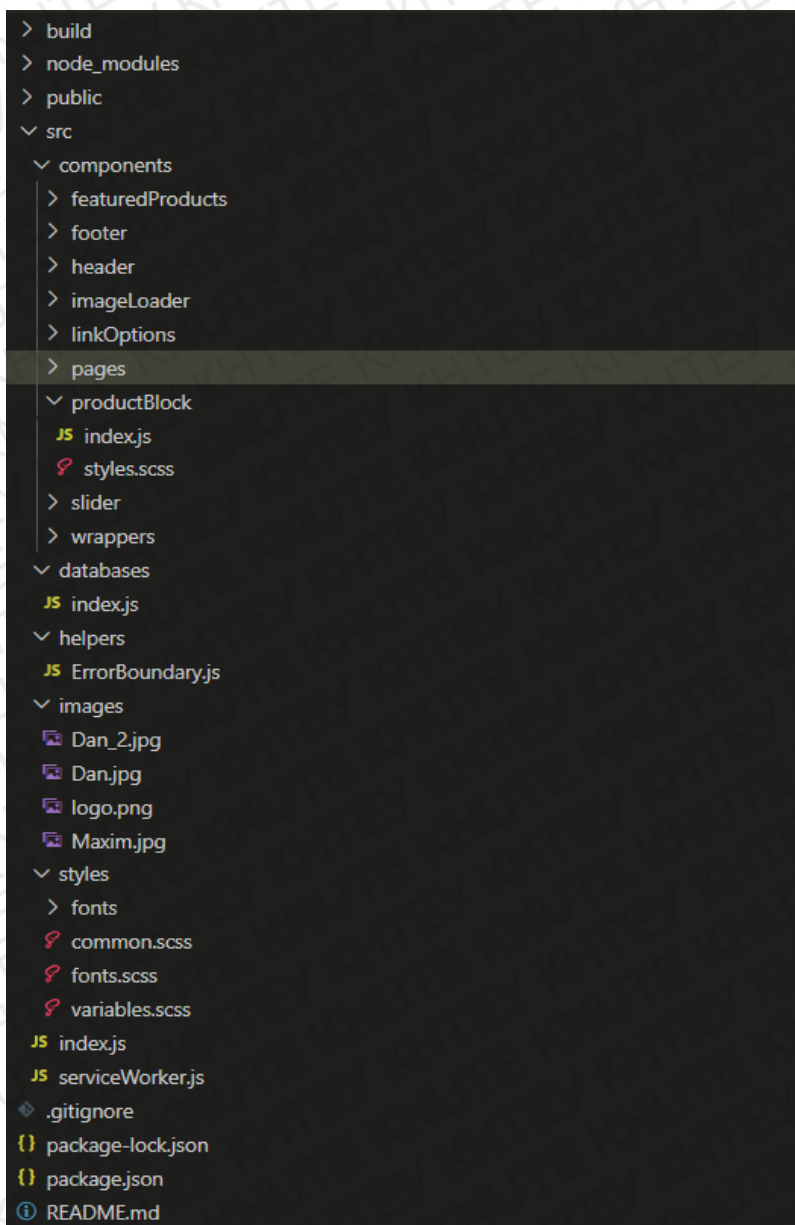


Рис. 2.3. Файлова структура веб-додатку.

2.3 Структура регулювання інтернет магазину

На даному етапі слід розібратись хто буде виконувати яку роль використовуючи веб-додаток. Є такі користувачі додатку:

- Відвідувач – особа яка щойно зайшла на сторінку. Основними функціями веб-сайту, якими відвідувач буде користуватися – це огляд товарів інтернет магазину, після додавання товару до корзини, відвідувач стає покупцем;
- Покупець – особа яка збирається зробити замовлення. Основні функції користування – це додавання даних до корзини, змінення її вмісту, та власне оформлення замовлення способом заповнення відповідної форми;
- Менеджер (продавець) – особа яка супроводжує покупця під час замовлення. Продавець узгоджує деталі замовлення способом телефонного дзвінка клієнту по номеру який він залишив. Менеджер уточнює деталі доставки та спосіб оплати, коли все вирішено та замовлення підтверджено, воно переходить до бухгалтера;
- Бухгалтер – особа яка приймає оплату за замовлення, надає підтвердження менеджеру про оплату, після чого продавець відсилає замовлення клієнту. Також бухгалтер веде облік рахунку компанії.

Ось графічне відображення даної схеми. (Рис. 2.4)



Рис. 2.4 Структура регулювання інтернет-магазину

2.4 Послідовність оформлення замовлення.

На даному етапі слід зрозуміти послідовність дій кожного з діючих об'єктів (покупця, гаджет, продавець, замовлення), які приймають участь у замовленні гаджету. Отже маємо таку послідовність дій цих об'єктів:

1. Клієнт обирає гаджет;
2. Клієнт подає заявку продавцю;
3. Продавець оформлює замовлення;
4. Клієнт узгоджує замовлення з продавцем;
5. Продавець робить запит на гаджет зі складу;
6. Продавець отримує гаджет;
7. Клієнт робить оплату замовлення;
8. Продавець видає замовлення.

Дана послідовність у виді діаграми (Рис. 2.5)

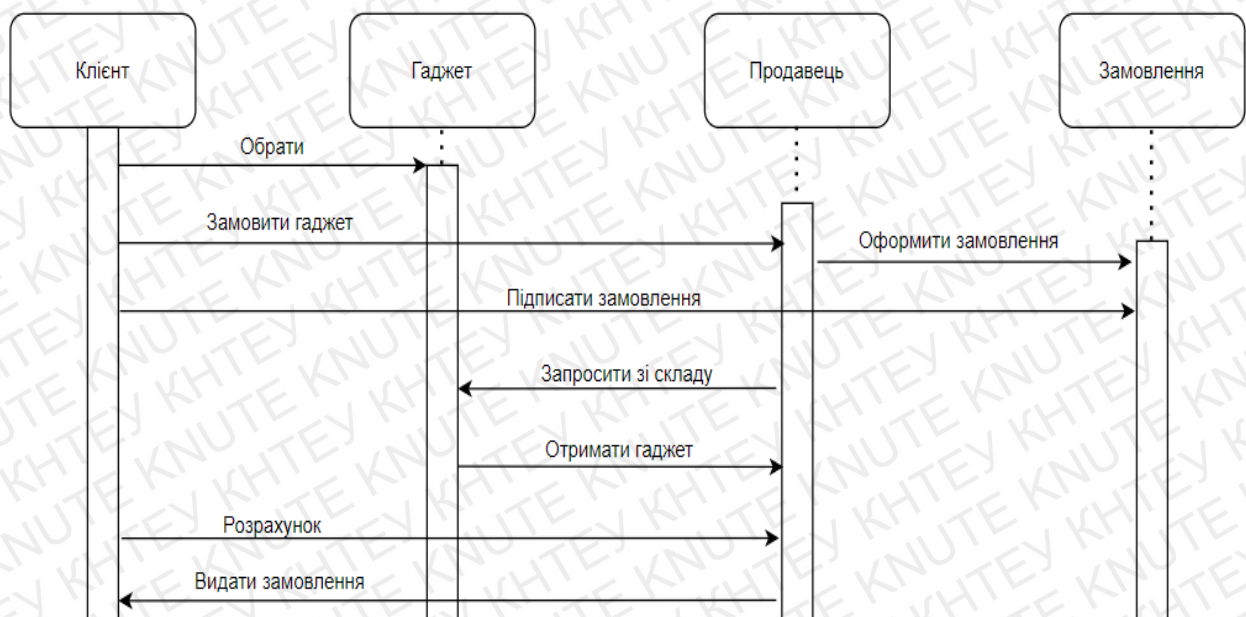


Рис. 2.5. Діаграма послідовностей створення замовлення.

2.5 Структура замовлення.

Для того щоб було легше регулювати замовлення та надавати звіт по ньому, слід його правильно оформити. Треба визначити основні деталі замовлення такі як: клієнт, товар, менеджер та вид оплати. Кожна із складових

має в свою чергу обов'язковий айді, по якому можна швидко визначити та знайти у пошуку по базі даних той чи інший об'єкт. Загальна структура замовлення та його складових має такий вигляд (Рис 2.6).

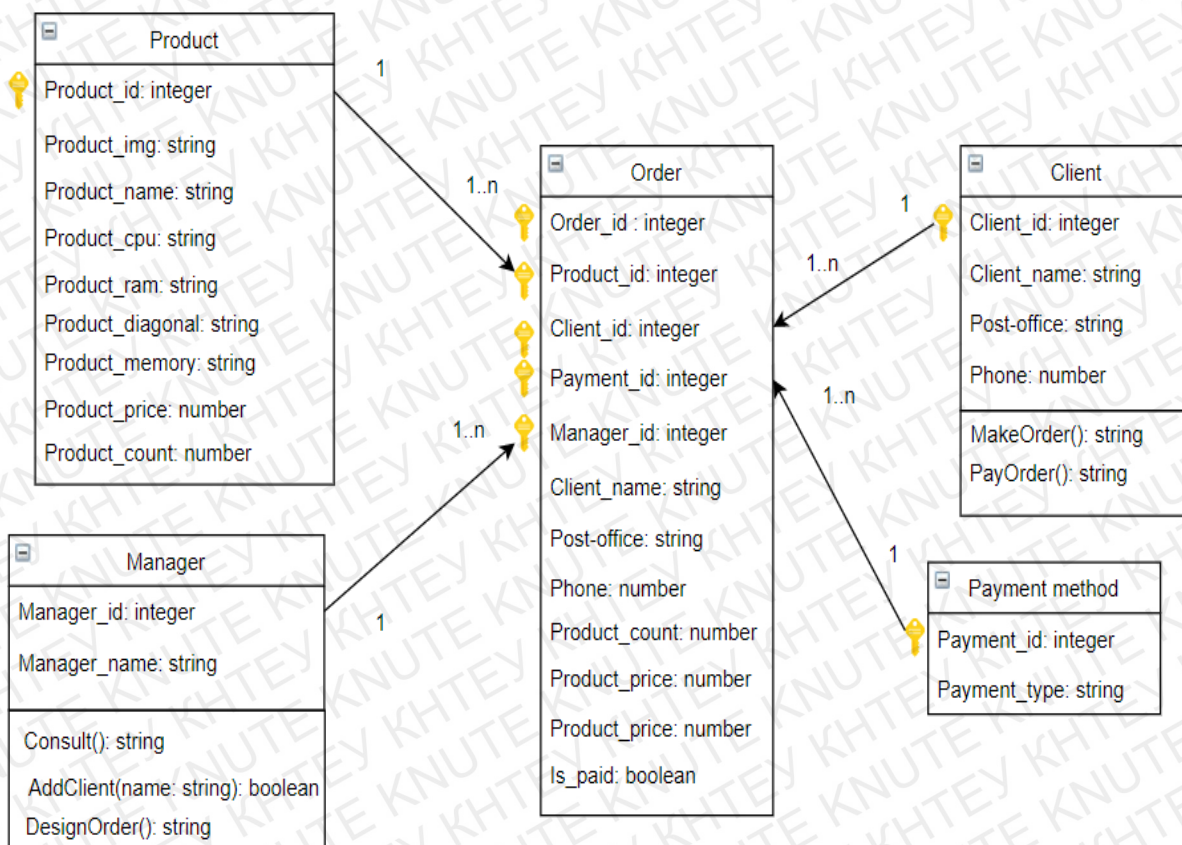


Рис. 2.6 Структура замовлення

2.6 Схема роботи клієнта з інтернет-магазином

Головна діюча особа під час користування веб-додатком – це клієнт (покупець). Щоб надати клієнту можливість зручно користуватися можливостями сайту треба поставити розробника на місце клієнта та виявити: яка послідовність дій буде у нього під час користування магазином у ролі покупця. Цей етап дуже важливий, так як саме зручність у користуванні – є вирішальним у питанні чи зробить користувач замовлення. На даній схемі видно послідовність дій клієнта, що користується інтернет-магазином. (Рис. 2.5).

Вхід в інтернет-магазин
цифрових гаджетів

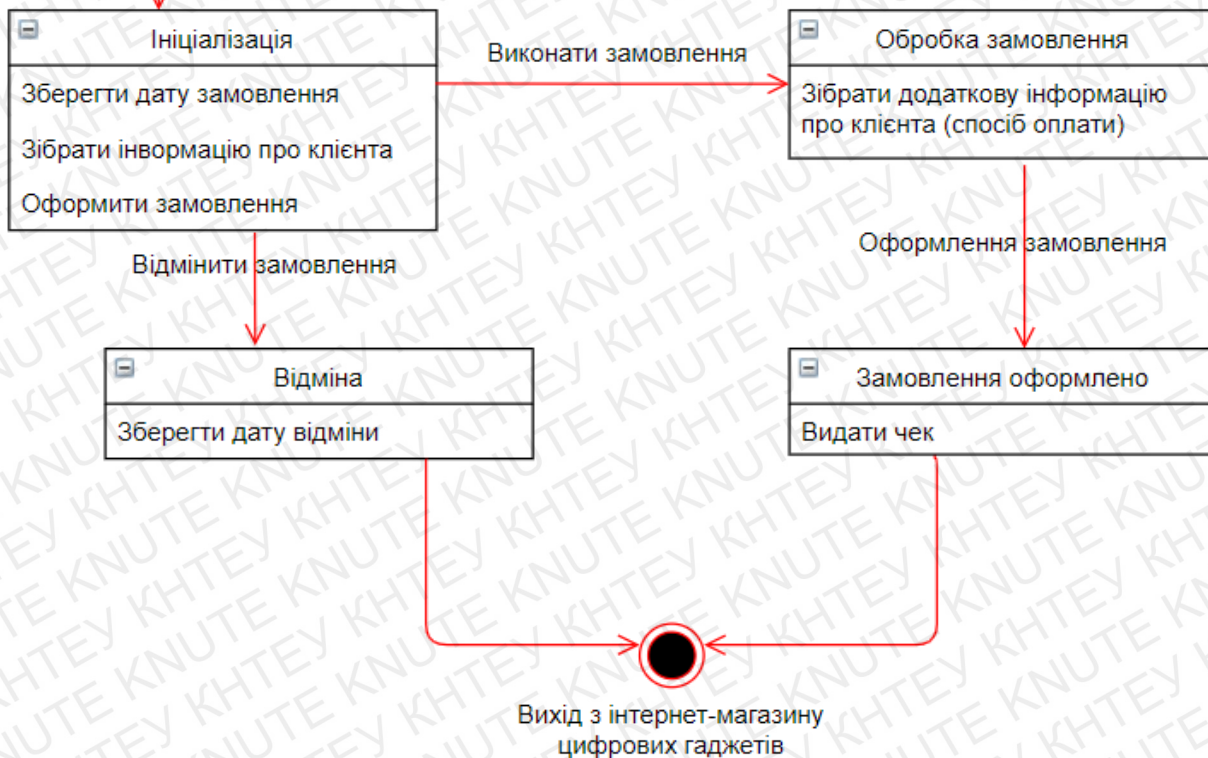


Рис. 2.6 Схема взаємодії клієнта з інтернет-магазином

2.7 Висновок до розділу 2

Найважливіший та найскладніший етап будь-якого проекту – це початок. Формування технічного завдання та вибір архітектури додатку мають великий вплив на подальшу розробку додатку, саме тому варто до цього етапу підходити серйозно із розумінням та баченням фінального продукту. Саме після вирішення архітектури додатку був створений та підготовлений до активної стадії розробки проект, що дозволило розробити користувацький інтерфейс засобами середовища розробки Visual Studio Code.

РОЗДІЛ 3

РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ ЦИФРОВИХ ГАДЖЕТІВ

3.1 Середовище розробки

Розробка на фреймворку React виконується шляхом написання коду на мові програмування JavaScript та стилізується за допомогою препроцесора SCSS. JavaScript файл, як SCSS файл – це звичайний текстовий документ, та середовищем розробки даних файлів є текстовий редактор. Розробляти можна навіть за допомогою стандартної програми «Блокнот», але головне у розброді – це якість та швидкість, які може надати текстовий редактор «Visual Studio Code».

Visual Studio Code – частково безкоштовний текстовий редактор, що має безліч плагінів для прискореної розробки. Також в ньому є вбудовані функції для виявлення та виправлення синтаксичних помилок у коді. Окрім корисних функцій, даний текстовий редактор має набір тем, які змінюють інтерфейс та шрифти для зручного користування редактором.

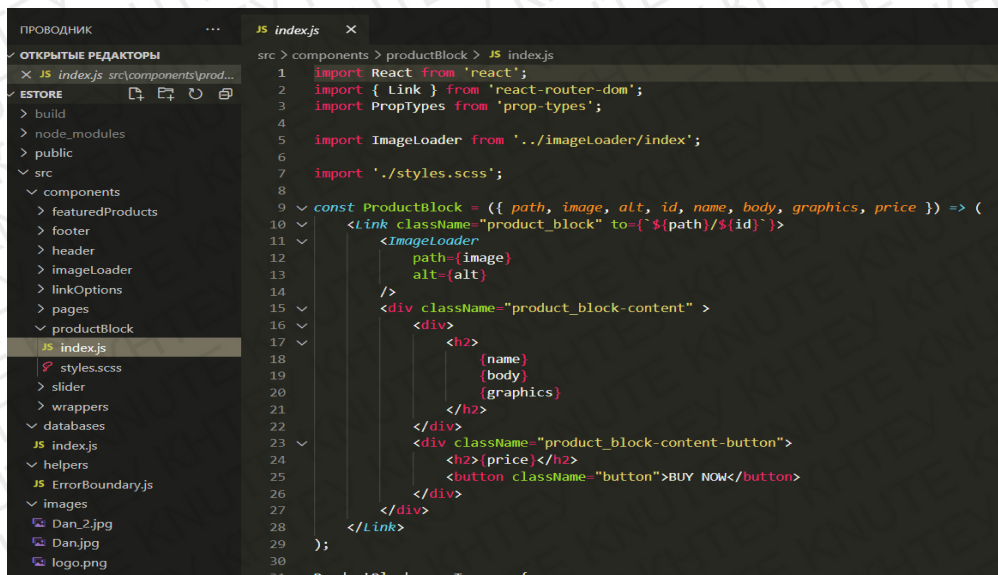


Рис.3.1. Інтерфейс Visual Studio Code

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка інтернет-магазину цифрових гаджетів	Стадія	Аркуш	Аркушів
Зав. каф.	Криворучко О.В					РЗ	26	42
Керівник	Цензура М.О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М.О.							
Розроб	Максимович Д.О.				РОЗРОБКА ІНТЕРНЕТ-МАГАЗИНУ			

3.2 Ініціалізація програми

Як вже було зазначено, React додаток побудований на системі збірки Webpack, налаштування якого відбувається в консолі шляхом написання команд.

Для використання Webpack на комп'ютері спочатку треба установити програмну платформу Node.js, що перетворює JavaScript з вузькоспеціалізованою мовою в мову загального призначення, та пакетний менеджер NPM. Маючи дані складові, можна приступати до створення React додатку.

Для ініціалізації обираємо папку у якій буде знаходитись проект та відкриваємо дану папку у редакторі. Visual Studio Code також має консоль для розробки. У консолі вводимо команду `npm create-react-app {app name}`, де `npm` – означає «зробити», `create-react-app` – {створення реакт додатку}, {app name} – довільна назва проекту. Після завершення установки маємо таку файлову структуру документа. (Рис. 3.2).

Имя	Дата изменения	Тип	Размер
.git	12.04.2021 13:57	Папка с файлами	
build	12.04.2021 13:59	Папка с файлами	
node_modules	12.04.2021 12:52	Папка с файлами	
public	12.04.2021 12:29	Папка с файлами	
src	12.04.2021 12:29	Папка с файлами	
.gitignore	12.04.2021 12:29	Исходный файл G...	1 КБ
package.json	12.04.2021 12:29	JSON File	2 КБ
package-lock.json	12.04.2021 12:29	JSON File	758 КБ
README.md	12.04.2021 12:29	Исходный файл ...	3 КБ

Рис. 3.2 Файлова структура проекту при ініціалізації

При виконанні команди було створено декілька тек, але є одна найголовніша, в якій і буде проходити розробка – це тека `src`. Для того щоб запустити проект треба перейти у папку з додатком та прописати команду

npm start, після чого на локальному сервері буде відкрито проект у браузері (Рис. 3.3).

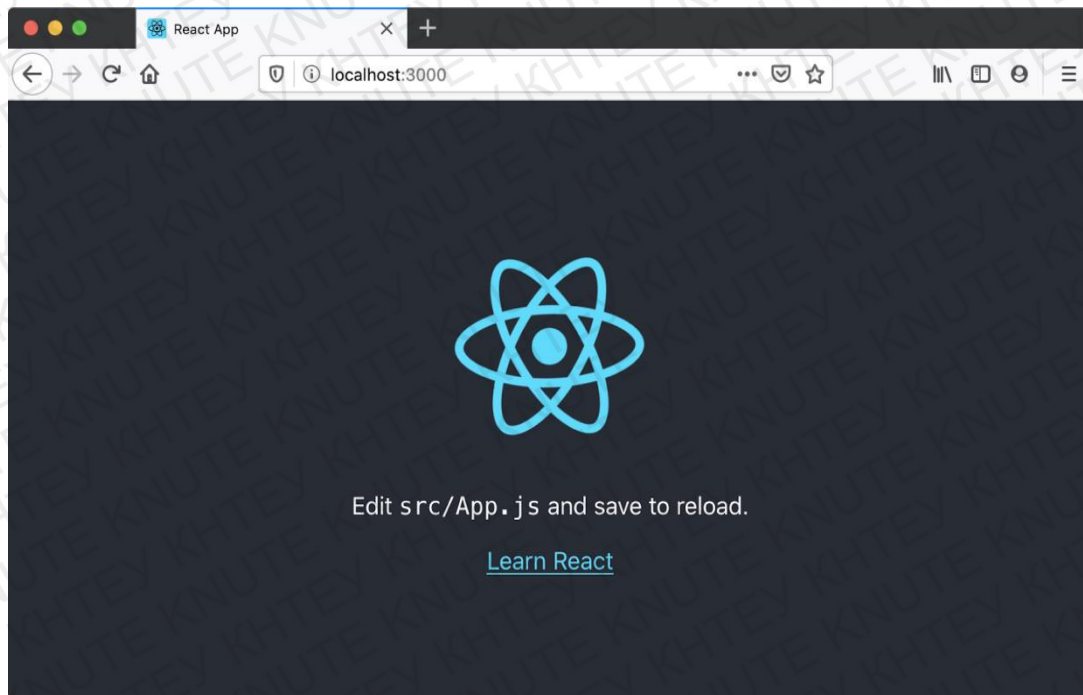


Рис. 3.3 Результат запуску проекту одразу після ініціалізації.

Отже проект ініціалізовано і можна приступати до розробки.

3.3 Розробка інтернет магазину.

Для початку треба видалити усе зайве із теки src, що було створено автоматично. Далі створюємо глобальні теки стилів, картинок, компонентів, баз даних і тд.

3.3.1 Принцип роботи React

Весь додаток на Реакт побудований на компонентах, які імпортуються один в інший до одного батьківського компонента App, який в свою чергу імпортується в головний файл index.js який виконує рендер (прорисовку) усього вмісту App (Рис. 3.4)



3.3.2 Створення компонента

Компонент – це звичайна JavaScript функція, яка повертає JSX код. JSX в свою чергу це те, що повертає функція у вигляді тегів, класів і тд. Тобто простими словами JSX – це HTML код всередині JavaScript коду. Для створення компоненту треба:

1. Створити .js файл;
2. У даному файлі імпортуємо бібліотеку React командою `import React from "React";`;
3. Імпортуємо допоміжні бібліотеки, компоненти та стилі;

4. Створюємо функцію з довільною назвою але обов'язково вона повинна починатися з великої букви (в даному випадку ProductBlock);
5. В тілі функції пишемо return, відкриваємо круглі скобки та пишемо JSX код, але даний код має бути обгорнутий у один батьківський елемент, якщо функція буде повертати два елемента, то буде помилка;
6. Експортуємо даний компонент командою export default ProductBlock:

В результаті маємо такий код (Рис. 3.5):

```
src > components > productBlock > JS index.js
1  import React from 'react';
2  import { Link } from 'react-router-dom';
3  import PropTypes from 'prop-types';
4
5  import ImageLoader from '../imageLoader/index';
6
7  import './styles.scss';
8
9  const ProductBlock = ({ path, image, alt, id, name, body, graphics, price }) => (
10    <Link className="product_block" to={`/${path}/${id}`}>
11      <ImageLoader
12        path={image}
13        alt={alt}
14      />
15      <div className="product_block-content" >
16        <div>
17          <h2>
18            {name}
19            {body}
20            {graphics}
21          </h2>
22        </div>
23        <div className="product_block-content-button">
24          <h2>{price}</h2>
25          <button className="button">BUY NOW</button>
26        </div>
27      </div>
28    </Link>
29  );
30
31  export default ProductBlock
```

Рис. 3.5 Компонент ProductBlock

На даному скріншоті (Рис. 3.4), Link та ImageLoader – це також компоненти. Link компонент з бібліотеки react-router-dom, який робить навігацію по веб-додатку. ImageLoader – це компонент створений автором який, в залежності від того завантажилась картинка чи ні, відображає спочатку анімовану картинку загрузки, а коли картинка завантажилась – саму картинку.

3.3.3 Props

На даному скріншоті (Рис. 3.5) можна побачити ряд параметрів які передаються у функцію ProductBlock. Ці параметри у бібліотеці React називаються props.

Як ми знаємо, в інтернет магазині буде необмежена кількість таких блоків з продуктом, та відмінність їх мінімальна: назва товару, ціна за товар, картинка та шлях до продукту який передається в компонент Link. Для вирішення проблеми повторення коду, створили props – звичайний об'єкт JavaScript який при визові компонента відображає його залежно від його самого. Наприклад: якщо викликати компонент і передати в параметр name значення laptop, то в результаті у нас відобразиться компонент з ім'ям ноутбук.(Рис. 3.6).

```
< ProductBlock
  image={ 'https://www.lenovo.com/medias/
  lenovo-laptop-yoga-c940-hero-14.png?
  context=bwFzdGVyfnHJvb3R8MTY4MTQxMmhpbnZS9wbmd8aDY0L2gzNy8xMDU0NzY
  4MjI3OTQ1NC5wbmd8OGZmWFJmZgzMmQxNzY4ZDI1ZDUxZDg1NDIzMWwFky2M1NzhmYW
  Nm2VlMzNlZDVlMGNlYjc3NDgzOWQ2OTdiNQ' }
  name="laptop"
  price="1000$"
/>
```

Рис. 3.6. Виклик компонента

В поле image записана url адреса картинки, в name – ім'я гаджета, в price – ціна за гаджет. Після збереження даного коду бачимо такий результат (Рис. 3.7):

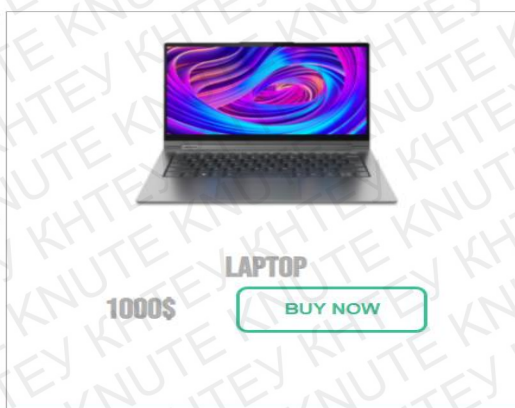


Рис. 3.7. Результат виклику компонента

3.3.4 Логіка компонентів

Так як компонент – це не лише JSX код, який, окрім відображення HTML, ніяк не функціонує. Він також може мати свою логіку, тобто JS код. Логіка компонента прописується у самому компоненті або поза ним, але до виклику “return”. (Рис. 3.6)

```
const Slider = ({arr}) => {  
  
  const [x, setX] = useState(0);  
  const goPrev = () => x === 0 ? setX(-100 * (arr.length - 1)) : setX(x + 100);  
  const goNext = () => (x === -100 * (arr.length - 1)) ? setX(0) : setX(x - 100);  
  
  return(  
    <section className="slider">  
      {  
        arr && arr.map((item, key) => {  
          return(  
            <div key={key} className="slide" style={{transform: `translateX(${x}%)`} to="/sales">  
              <div className="container">  
                <ImageLoader  
                  path={item.img}/>  
              </div>  
            </div>  
          )  
        })  
      }  
    </section>  
  )  
}
```

Рис. 3.7. Логіка компонента Slider

3.3.5 База даних

Для кращої оптимізації проекту та для швидшої розробки створюємо базу даних за допомогою Microsoft SQL Server та мови SQL. Етапи створення бази даних:

1. Командою CREATE DATABASE GadgetsDB, створюємо базу даних. CREATE DATABASE – два ключові слова для створення нової бази даних, після цих слів пишеться назва бази даних в один рядок. Так ми створили базу даних з назвою GadgetsDB.
2. Командою CREATE TABLE { Table_name } ({row} NOT NULL {type} (IDENTITY)) створюємо таблицю, де { Table_name } – назва таблиці; {row} – назва стовбця, їх може бути безліч; {type} – тип стовбця; **NOTNULL** означає, що значення цього атрибута не може бути пропущеним при додаванні записів у таблицю. Слово «**IDENTITY**» означає, що при додаванні записів у таблицю значення атрибута, позначеного цим ключовим словом, вказувати не треба, оскільки значення будуть додаватися автоматично (найчастіше

вказується для полів id). Таким чином створюємо таблиці Computers, Laptops, Phones, Sales.

3. Командою ALTER TABLE { Table_name } ADD PRIMARY KEY ({Table_name_id}) створюємо первинні ключі для кожної з таблиць. Первинний ключ – це поле у відповідній таблиці, яке завжди має бути унікальним, найчастіше це id.
4. Командою ALTER TABLE { Table_name } ADD FOREIGN KEY ({Table_name_id}) REFERENCES { Table_name }({ Table_name_id }) створюємо відношення між таблицями.
5. Командою INSERT INTO { Table_name } ({...rows}) VALUES ({...values}) вносимо дані у таблиці;
6. Движок браузерів підтримує базу даних лише формату .json, тобто стандартне розширення бази даних .mdf не буде відображати базу даних у веб-додатку. Для цього експортуємо базу даних у потрібний формат командою: SELECT {row_name} AS {name} FROM {Table_name} FOR JSON PATH, ROOT ({variable});

Таким чином маємо такий результат (Рис. 3.8)

```
const computers = [{
  id: "1",
  img: "https://brain.com.ua/static/images/prod_img/3/5/U0269135_big.jpg",
  name: "HELLCAT PRO",
  body: "Vinga Graphyte",
  graphics: "GeForce GTX 1050, 2GB",
  cpu: "Ryzen 3 1200 3.4GHz",
  motherboard: "Socket AM4 (AMD A320 Chipset, 2 x DDR4, mATX",
  ram: "8GB DDR4-2400",
  cooling: "Boxed cooler",
  memory: "500GB HDD",
  power: "500W DeepCool",
  price: "500$",
},
{
  id: "2",
  img: "https://i2.rozetka.ua/goods/15901515/158971303_images_15901515176.jpg",
  name: "PRO GAMER 3000",
  body: "1stPlayer F4-A1 Blue",
  graphics: "GeForce GTX 1050 ti, 4GB",
  cpu: "Intel Pentium Gold G5600F",
  motherboard: "Socket 1151v2 (Intel® H310 Chipset, 2 x DDR4, mAT)",
  ram: "8GB DDR4-2400",
  cooling: "Boxed cooler",
  memory: "1TB HDD",
  power: "500W DeepCool",
  price: "550$"
},
]
```

Рис. 3.8 База даних JSON

Підключення бази даних на фреймворку відбувається значно легше ніж у нативному JS. Щоб підключити базу даних у JS треба робити XMLHttpRequest

запроси, на це піде близько десяти рядків коду. У React це робиться набагато легше та швидше командою: “import data from ‘!./../databases/index’;”, де ‘!./../databases/index’ – це відносний шлях до файла JSON, де і лежить база даних.

3.3.7 Навігація

Навігація у веб-додатку на React відбувається не зовсім звичним способом. Якщо у звичайному HTML ми створюємо тег а із атрибутом href, значення якого – відносний шлях до іншого файла: То у випадку з React, застосовується спеціальна бібліотека “**react-router-dom**”.

“**react-router-dom**” – бібліотека Реакт, яка містить набір необхідних компонентів для навігації. Для того щоб створити навігацію, треба:

1. В першу чергу встановити дану бібліотеку локально прописавши в консолі відповідну команду `npm install react-router-dom`;
2. Імпортувати дану бібліотеку із допоміжними компонентами `Switch`, `Router` у головний компонент;
3. На найвищому рівні, тобто батьківський компонент обгорнути у компонент даної бібліотеки. Найпоширеніший компонент – це `BrowserRouter`, через його оптимізованість і зручність використання;
4. Всередині `BrowserRouter` вставляємо компонент `Switch`, дочірними компонентами якого будуть компоненти `Route` з атрибутами `path`, `component` та `exact`. `Path` – це url, в залежності від якого буде з’являтися та чи інша сторінка, за яку відповідальний атрибут `component`.

Пройшовши ці етапи, маємо такий компонент `App` (Рис. 3.8).

```

23 const App = () => {
24   const supportsHistory = 'pushState' in window.history;
25   return (
26     <BrowserRouter
27       forceRefresh={!supportsHistory}
28       basename="eStore"
29     >
30     <ErrorBoundary>
31       <Helmet>
32         <link rel="icon" href={logo}/>
33       </Helmet>
34       <Header>
35         <Switch>
36           <Route exact path="/" component={Home}/>
37           <Route exact path="/sales" component={Sales}/>
38           <Route exact path="/contactus" component={ContactUs}/>
39           <Route exact path="/laptops" component={Laptops}/>
40           <Route exact path="/computers" component={Computers}/>
41           <Route exact path="/phones" component={Phones}/>
42           <Route exact path="/search" component={Search}/>
43           <Route exact path="/bag" component={Bag}/>
44           <Route exact path="/booking" component={Booking}/>
45           <Route path="/computers/:computerid" component={ProductInfo}/>
46           <Route path="/laptops/:laptopid" component={ProductInfo}/>
47           <Route path="/phones/:phoneid" component={ProductInfo}/>
48           <Route component={NotFound}/>
49         </Switch>
50       </Header>
51     </ErrorBoundary>
52   </BrowserRouter>
53 )
54 }
55
56 export default App;

```

Рис. 3.8 Реалізація навігації додатку

Для того щоб перейти на відповідну сторінку, треба імпортувати такі компоненти з бібліотеки react-router-dom як Link або NavLink – виконують вони одну й ту саму роль: переходять на задану сторінку за допомогою атрибута to.

3.3.8 Відображення товарів на сторінці

У базі даних ми маємо чотири масиви JavaScript: computers, laptops, phones та sales. Також маємо чотири сторінки computers, laptops, phones та sales. Тепер потрібно відобразити список товарів залежно від сторінки яку обрав користувач. Для того щоб не дублювати власний код (елементів може бути десятки, а може й навіть сотні), а зробити це швидко та з використанням найменшої кількості коду, треба задіяти цикли.

Окрім простих циклів forEach, for і while, з виходом ES7 (EcmaScript 7 – версія, стандарт написання коду на JavaScript) додалися нові цикли які вміють не лише перебирати масив, а і змінювати його або повертати новий в залежності від відповідних умов. До таких циклів відносяться: sort, filter, map.

Sort та filter перебирають масив та повертають потрібний елемент на кожній ітерації, а map – повертає змінений масив, для відображення товарів на сторінці треба використовувати саме його, так як нам потрібно перетворити звичайний об'єкт JavaScript у HTML код. (Рис. 3.9).

```
import { laptops } from '../../databases/index';

const Laptops = ({history}) => {
  const path = history.location.pathname;
  return (
    <>
      <Helmet>
        <title>Laptops</title>
      </Helmet>

      <div className="container">
        <div className="wrapper">
          <div>
            {
              laptops.map((item, key) => {
                return(
                  < ProductBlock
                    path={path}
                    key={key}
                    image={item.img}
                    id={item.id}
                    name={item.name}
                    price={item.price}
                  />
                )
              })
            }
          </div>
        </div>
      </div>
    </>
  )
}

export default Laptops
```

Рис. 3.9 Код відображення товарів на сторінці

3.3.9 Реалізація функціоналу корзини

Головна задача інтернет-магазину – це відправка продавцю інформації про заказ, а саме: ім'я, прізвище, номер телефону, адреса та найголовніше – це список замовлених товарів. Для реалізації даного функціоналу нам потрібне сховище яке, незалежно від часу, відвідуваної сторінки або гаджета, буде запам'ятовувати таку інформацію. Існує три таких сховища: **LocalStorage**, **SessionStorage** та **Cookies**.

- **SessionStorage** – сховище що, запам'ятовує набір інформації в межах однієї сесії, тобто якщо закрити вкладку сторінки, то дані зітруться;
- **Cookie** – запам'ятовує дані на протязі відповідного часу, який встановлюється розробником, але не назавжди;

- **LocalStorage** – сховище, що запам'ятовує інформацію назавжди, поки користувач не зітре її через консоль розробника або за допомогою функціонала розробленим на сайті.

Найкращим варіантом для вирішення даної задачі є LocalStorage. Синтаксис LocalStorage це ключ – значення, тобто для довільного ключа є своє значення, наприклад “time”: “12AM”, де “time” це ключ, “12AM” – значення. LocalStorage - звичайний JavaScript об’єкт, що має набір методів, серед яких у додатку буде використано get та set.

Синтаксис методу get об’єкту LocalStorage: window.localStorage.get(‘key’); де key – ім’я ключа. Синтаксис методу set: window.localStorage.set(‘key’, ‘value’); де value – значення.

Таким чином залишилось обробити подію (Рис. 3.10) яка викличе функцію (Рис. 3.11), яка в сховище додасть новий об’єкт – товар котрий клієнт збирається купити.

```
<div className="button_wrapper">
  <h3>{machedObj.lastPrice && machedObj.lastPrice}</h3>
  <button onClick={buyGadget}>{machedObj.price}</button>
</div>
```

Рис. 3.10 Обробник події “click”

```
const buyGadget = () => {
  modal.current.style.display = 'flex';
  setTimeout(() => {
    modal.current.classList.add('shown');
  }, 100);
  const buys = JSON.parse(window.localStorage.getItem('buys')) || [];
  const buy = {
    id: machedObj.id,
    img: machedObj.img,
    name: machedObj.name,
    price: machedObj.price,
    count: 1,
    type: type
  }
  const index = buys.findIndex(item => item.id === machedObj.id && item.name === machedObj.name);
  if (index > -1) buys[index].count++;
  else buys.unshift(buy);
  window.localStorage.setItem('buys', JSON.stringify(buys));
}
```

Рис. 3.11 Додавання обраного товару у сховище

Тепер при вході на сторінку корзини треба застосувати метод get для перебору масиву обраних товарів та відобразити їх. На останньому етапі

розроблюємо форму у якій користувач залишить усі необхідні дані для оформлення замовлення, які потім прийдуть на пошту менеджера.

3.4 Висновок до розділу 3

У даному розділі були повністю розглянуті засоби реалізації розробленого інтерфейсу, логіки додатку. Також були розглянуті та реалізовані основні можливості фреймворку React, а саме: ініціалізація Реакт проекту та створення правильної структури; принципи роботи даного фреймворка; створення компонента та його логіки; відображення компонентів за допомогою циклів; створення навігації, завдяки якій і відбувається перехід між сторінками у додатку розробленого на Реакт та багато іншого. Завдяки правильно продуманій базі даних, був розроблений інтернет-магазин цифрових гаджетів з достатньо широким об'ємом товарів, можливістю переходу на обраний товар для його детального аналізу та додавання до кошика. Також було розроблено пошукову систему для сайту, котра дозволяє шукати гаджет не лише по назві фірми, але і по моделі. Був розроблений функціонал кошика, яка відображає список обраних товарів з можливістю видалення та кнопкою «Купити», при натисканні на яку, користувача переводить на сторінку з формою, де він повинен внести свої дані для оформлення замовлення менеджером.

ВИСНОВКИ ТА ПРОПОЗИЦІЇ

В ході виконання дипломної роботи було створено компанію eStore, використовуючи сучасні технології, було розроблено інтернет-магазин даної компанії. Також був обраний асортимент товарів магазину для заданої цільової аудиторії. В ході розробки інтернет-магазину eStore була розглянута архітектура даного програмного забезпечення, а саме: файлова структура, архітектура взаємодії користувачів із програмним забезпеченням, послідовність оформлення замовлення у веб-додатку та взаємодія клієнта з інтернет-магазином. Завдяки вибору крос-платформного фреймворку React був розроблений повноцінний інтернет-магазин цифрових гаджетів, котрий виділяється серед конкурентів адаптивністю, оптимізованістю та відсутністю реклами.

Під час розробки були задіяні такі технології:

- PhotoShop – графічний редактор для створення UX / UI веб-дизайну;
- Star UML – для розробки діаграм, завдяки яким було створено візуальну структуру проекту;
- React – бібліотека JavaScript, розроблена компанією Facebook;
- Webpack – система зборки проекту;
- SCSS – CSS препроцесор, для комфортної та швидкої розробки;

Задіяні такі бібліотеки:

- React-helmet – бібліотека, яка дозволяє додавати імена та іконки для кожної із сторінок;
- React-router-dom – для навігації по проекту;
- React-slick – для реалізації спайдерів у проєкті;

					КНТЕУ 121 06-15.БР			
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка інтернет-магазину цифрових гаджетів	Стадія	Аркуш	Аркушів
Зав. каф.	Криворучко О.В.					ВП	39	42
Керівник	Цензура М.О.					Факультет інформаційних технологій, 4 курс, 6 група		
Гарант	Цензура М.О.							
Розроб	Максимович Д.О.							
					ВИСНОВКИ ТА ПРОПОЗИЦІЇ			

- Prop-types – для типізації компонентів, щоб уникнути помилок у проєкті;

У процесі розробки інтернет-магазину були викладені та показані на практиці основні функції фреймворку React JS, що може дати впевнений старт для початку розробки веб-додатків починаючим програмістам.

У подальшому розвитку додатку можливе додавання інших мов інтерфейсу та побудова більш раціонального у використанні інтерфейсу. Результат виконаної роботи: <https://daniil-maksimovich.github.io/eStore>

							Архив
Зм.	Архив	№ докум	Підпис	Дата	КНТЕУ 121 06-15.БР		40

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Грабер Мартін . SQL для простих смертних: керівництво по створенню баз даних, 395 ст., 2014.
2. Мартін Фаулер. UML. Основи. Керівництво по комп'ютерному моделюванню, 192 ст., 2011.
3. Маргулис Ден. Photoshop для професіоналів, 656 ст., 2007.
4. Білл Карвін. Програмування баз даних SQL: Типові помилки і їх усунення, 336 ст., 2012
5. Джеймс Р. Грофф. SQL: повний довідник / Джеймс Р. Грофф, Пол Н. Вайнберг, Ендрю Дж. Опель, 960 ст., 2015.
6. Алекс Кригель. SQL: Біблія користувача / Алекс Кригель, Борис Трухнов, 752 ст., 2009.
7. Робін Ніксон. Навчання PHP, MySQL, JavaScript, CSS та HTML5: Покрокове керівництво зі створення динамічних веб-сайтів, 3-е видання, 816 ст., 2016.
8. Девід Фленаган. JavaScript: кишеньковий довідник, 320 ст., 2019.
9. Е. Браун. Вивчаємо JavaScript: керівництво по створенню сучасних веб-сайтів, 368 ст., 2020.
10. Дуглас Крокфорд. JavaScript: повне керівництво зі створення динамічних веб-сайтів 1-е видання, 355 ст., 2019.

Інтернет ресурси

11. Photoshop [Електронний ресурс] - Режим доступу:
<https://www.adobe.com/ua/products/photoshop.html>.
12. Webpack [Електронний ресурс] - Режим доступу:
<https://habr.com/ru/post/514838/>.

					<i>КНТЕУ 121 06-15.БР</i>		
Зм.	Аркуш	№ докум	Підпис	Дата	Розробка інтернет-магазину цифрових гаджетів СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ		
Зав. каф.	Криворучко О.В.						
Керівник	Цензура М.О.						
Гарант	Цензура М.О.						
Розроб	Максимович Д.О.						
					Стадія	Аркуш	Аркушів
					СД	41	42
					Факультет інформаційних технологій, 4 курс, 6 група		

13.ReactJS [Электронный ресурс] Режим доступа: <https://learn-reactjs.com/home>.

14.JavaScript [Электронный ресурс] Режим доступа: <https://learn.javascript.com/>.