

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

магістр

(ступінь вищої освіти)

на тему ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ МЕТОДІВ
ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ ОНЛАЙН-ТОРГІВЛІ НА
ПРИКЛАДІ ВЕБСАЙТУ ПРОДАЖУ КОМП'ЮТЕРНОЇ ТЕХНІКИ
RESEARCH AND SOFTWARE IMPLEMENTATION OF METHODS
FOR INCREASING THE EFFICIENCY OF ONLINE TRADE BY
EXAMPLE OF A WEBSITE SELLING COMPUTER EQUIPMENT

Виконав: студент 2 курсу, групи КНТ-222м

Спеціальності 122 Комп'ютерні науки

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Комп'ютерні науки

КУЧЕРЕНКО Я.К.

(прізвище та ініціали)

Керівник СТЕПАНЕНКО О.О.

(прізвище та ініціали)

Рецензент ТЕРЕЩЕНКО Е.В

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Факультет ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти магістр
 Спеціальність 122 Комп'ютерні науки
(код і найменування)
 Освітня програма (спеціалізація) Комп'ютерні науки
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н., проф.
Сергій СУББОТІН
 “ ” 2023 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА

КУЧЕРЕНКО Ярослава Костянтиновича

(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна реалізація методів
Підвищення ефективності онлайн-торгівлі на прикладі вебсайту продажу
комп'ютерної техніки

Research and Software Implementation of Methods for Increasing the Efficiency of
Online Trade-By Example of a Website Selling Computer Equipment

керівник проєкту(роботи) СТЕПАНЕНКО Олександр Олексійович, к.т.н, доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “14” листопада 2023 року
№ 437

2. Строк подання студентом проєкту (роботи) 01 грудня 2023 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне
завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Реферат. 1. Аналіз проблеми та постановка завдань дослідження.

2. Матеріали та методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

4. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Слайди презентації

5. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-6 Основна частина	СТЕПАНЕНКО О.О. к.т.н., доцент.		
Нормоконтролер	ДЕЙНЕГА Л.Ю., ст. викладач		

6. Дата видачі завдання “04” ли 2023 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	2–4 тижні	Розділ 1
3	Вибір засобів розробки програмного продукту.	5 тиждень	Розділ 2
4	Розробка архітектури програми.	6 тиждень	Розділ 3
5	Розробка програмного продукту	7-9 тижні	Розділ 4-5
6	Тестування та експериментальне дослідження програмного забезпечення.	10 тиждень	Розділ 6
7	Оформлення пояснювальної записки та документів до неї.	11 тиждень	Додатки
8	Нормоконтроль та рецензування.	12–13 тижні	
9	Захист роботи.	14 тиждень	

Студент

(підпис) Ярослав КУЧЕРЕНКО
(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис) Олександр СТЕПАНЕНКО
(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:

113 с., 5 табл., 16 рис., 4 дод., 27 джерела.

ОНЛАЙН-ТОРГІВЛЯ, ВЕБСАЙТ, СОРТУВАННЯ, ЕФЕКТИВНІСТЬ
ПРОГНОЗУВАННЯ, JAVASCRIPT.

Об'єкт дослідження – процес онлайн-торгівлі на прикладі вебсайту.

Предмет дослідження – методи підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту.

Мета роботи – дослідження та програмна реалізація методів підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту

Метод дослідження – оптимізація вебсайту, аналіз користувацького досвіду, аналіз даних.

Матеріали, методи та технічні засоби: мова розмітки гіпертексту HTML, каскадні таблиці стилів CSS, мова програмування JavaScript, середовище програмування Sublime Text, персональний комп'ютер з процесором Intel Core i3 під управлінням операційної системи Microsoft Windows 7.

Результати. Створено вебсайт з відповідним інтерфейсом, за допомогою якого користувач може отримати інформацію про товар, характеристики та наявність, а також залишити заявку на замовлення та відгук про товар.

Висновки. Створено вебсайт для продажу комп'ютерної техніки за допомогою якого можна оцінити методи підвищення ефективності процесу онлайн-торгівлі.

Галузь використання – електронна комерція.

ABSTRACT

Explanatory note to the diploma qualifying work of the master: 113 pages, 16 figures, 5 tables, 4 appendixes, 27 sources.

ONLINE TRADE, WEBSITE, SORTING, FORECASTING EFFICIENCY, JAVASCRIPT.

The object of investigation is the process of online trading on the website.

The subject of investigation is methods for increasing the efficiency of the online trading process on a website.

The purpose of this work - research and program implementation of methods to improve the efficiency of the online trading process on the website.

Research method – website optimization, analysis of customer data, data analysis.

Materials, methods and technical means: language layout for HTML hypertext, cascading style sheets CSS, language JavaScript programming, Sublime Text programming middleware, personal computer with an Intel Core i3 processor running the Microsoft Windows 7 operating system.

Results. A website has been created with a user-friendly interface, through which you can access information about the product, characteristics and availability, as well as submit a request for information and information about the product.

Conclusions. A website has been created for the sale of computer equipment to help you evaluate methods for increasing the efficiency of the online trading process.

The area of use is electronic commerce.

ЗМІСТ

	С.
Перелік скорочень та умовних познач	8
Вступ.....	9
1 Аналіз предметної області	10
1.1 Основні поняття.....	10
1.2 Огляд існуючих методів.....	11
1.3 Порівняльний аналіз існуючих методів вирішення завдання підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту	23
1.4 Дослідження інформаційних систем	25
1.5 Порівняння існуючих систем дослідження, що можуть бути використані для підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту	33
1.6 Постановка завдання до роботи.....	35
1.7 Висновок	36
2 Вибір та обґрунтування структури системи, яка проєктується	37
2.1 Вибір мови програмування	37
2.2 Порівняння мов програмування.....	45
2.3 Вибір середовища розробки.....	47
2.4 Порівняння середовищ розробки.....	51
2.5 Огляд мініфікаторів коду	52
2.6 Структурна схема програмного продукту.....	54
2.7 Висновок	56
3 Основні рішення щодо реалізації компонентів системи	57
3.1 Попередня обробка даних	57
3.2 Оптимізація конверсії	59
3.3 Висновок	60
4 Керівництво програміста	61
4.1 Призначення й умови використання	61

	7
4.2 Звертання до програмного продукту (файли проєкту).....	61
4.3 Вхідні і вихідні дані	62
4.4 Повідомлення	63
4.5 Висновок	64
5 Керівництво оператора	66
5.1 Експлуатація та тестування	66
5.2 Виконання програмного продукту	66
5.3 Повідомлення оператору.....	71
5.4 Висновок	71
6 Експерименти та результати.....	73
6.1 Проведення експериментів	73
Висновки.....	74
Перелік джерел посилань.....	76
Додаток А Технічне завдання.....	80
Додаток Б Опис програмного продукту.....	87
Додаток В Текст програми	97
Додаток Г Слайди презентації.....	108

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

JS	– JavaScript;
IDE	– Integrated Development Environment;
URL	– Uniform Resource Locator;
HTTP	– Hypertext Transfer Protocol;
UI	– User Interface;
UX	– User Experience;
CMS	– Content Management System;
DNS	– Domain Name System;
FTP	– File Transfer Protocol;
SSL	– Secure Sockets Layer;
ПП	– програмний продукт;

ВСТУП

В сучасному цифровому світі онлайн-торгівля стала ключовим компонентом глобальної економіки, розширюючи можливості бізнесу і забезпечуючи споживачам доступ до широкого асортименту товарів та послуг. З плином часу конкуренція в цьому сегменті зростає, і бізнесу необхідно постійно удосконалювати свої стратегії, щоб залишатися конкурентоспроможними та забезпечувати високий рівень задоволення клієнтів.

Сучасний електронний бізнес є динамічним та піддається постійним змінам. Зростання числа онлайн-платежів, розвиток технологій та зміни у споживацькій поведінці викликають необхідність постійного адаптування до нових умов. Дослідження ефективних методів впровадження стратегій онлайн-торгівлі стає важливим завданням для забезпечення конкурентоспроможності та стійкості бізнесу.

Україна, як і багато інших країн, переживає бурхливий розвиток електронної комерції, що відбувається на тлі швидкого зростання доступу до Інтернету та зміни споживацьких уподобань. Розуміння та впровадження ефективних методів для підвищення результативності вебсайтів електронної комерції має велике значення для успіху місцевих бізнесів та розвитку економіки в цілому.

Ця робота має на меті дослідити різні методи підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту для продажу комп'ютерної техніки. Дослідження включає в себе аналіз сучасних тенденцій у сфері електронної комерції, виявлення факторів, які впливають на успіх онлайн-торгівлі, та розгляд ефективних стратегій для підвищення конверсії та задоволення клієнтів.

Отже, завдання дослідження та програмна реалізація методів підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту є актуальним завданням.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Основні поняття

Електронна комерція (e-commerce) – це термін, що описує процес купівлі та продажу товарів чи послуг через Інтернет. Вона включає в себе весь спектр електронних транзакцій, починаючи від онлайн-магазинів і закінчуючи електронними платіжними системами.

Онлайн-торгівля (Online Retail) – це частина електронної комерції, яка концентрується на роздрібній торгівлі та продажу товарів чи послуг кінцевим споживачам через Інтернет. Це може включати в себе віддалені покупки через вебсайти, мобільні додатки та інші онлайн-платформи.

Останні роки свідчать про ряд ключових тенденцій у сфері онлайн-торгівлі. Збільшення зацікавленості у споживачів до персоналізації, розширення використання штучного інтелекту для покращення експерієнції користувачів та зростання популярності мобільного шопінгу є лише кількома з напрямів, які визначають динаміку цього ринку. Звичайно ж, питання безпеки в онлайн-торгівлі є також важливим аспектом для споживачів. Заходи забезпечення конфіденційності особистої інформації, використання шифрування та інші методи кібербезпеки можуть бути ключовими у забезпеченні довіри споживачів.

Конверсія (Conversion) - поняття, яке вказує на те, як часто відвідувачі вебсайту виконують бажану дію, таку як покупка товару, заповнення форми або інші цільові дії. Вимірювання та оптимізація конверсії є важливими аспектами стратегій електронної комерції.

Важливими типами конверсій є конверсія в продажі, заповнення форми та завантаження файлу. Конверсія в продажі є ключовим показником ефективності, визначаючи відсоток відвідувачів, які вчинили покупку.

Цей показник не лише вимірює ефективність вебсайту, а й визначає ступінь виконання цілей. Зрозуміння конверсійного процесу дозволяє бізнесам

оптимізувати свої вебсайти, покращувати інтерфейс, контент та інші аспекти для підвищення конверсійної ефективності.

Фактори, що впливають на конверсію, включають дизайн та інтерфейс вебсайту, якість контенту, способи оплати та доставки, а також взаємодію з клієнтами. Аналіз та використання аналітичних інструментів є важливими для визначення чинників, які впливають на конверсію, та для подальшої оптимізації бізнес-процесів у сфері онлайн-торгівлі.

Мобільна торгівля (Mobile Commerce) – поняття, що описує процес здійснення торгівлі за допомогою мобільних пристроїв, таких як смартфони чи планшети. Зростання використання мобільних пристроїв суттєво впливає на стратегії онлайн-торгівлі, спонукаючи бізнеси створювати мобільно-оптимізовані вебсайти та додатки.

1.2 Огляд існуючих методів

В останні роки з розвитком комп'ютерних технологій та штучного інтелекту ландшафт електронної комерції розвивається швидкими темпами, і важливо розуміти тенденції для успішної конкуренції. Зростання популярності глобальних ринків, зосереджених на досвіді користувача та персоналізації послуг, а також використання розширених технологій, таких як віртуальна та доповнена реальність, є ключовими тенденціями[1]. Звичайно ж необхідно розглянути всі існуючі методи та виділити переваги і недоліки кожного, для того щоб ефективно оцінювати ті чи інші підходи. Поглиблений аналіз методів, які вже були впроваджені може позитивно вплинути на процес підвищення ефективності онлайн-торгівлі, а також виявити всі ключові аспекти, спрямовані на оптимізацію вебсайтів та покращення взаємодії з аудиторією, що безумовно є дуже актуальним та необхідним.

1.2.1 Оптимізація вебсайту

Оптимізація вебсайту — це комплекс заходів, спрямованих на поліпшення його функціональності, швидкодії, технічних характеристик та користувацького досвіду. Цей процес включає в себе ряд ключових аспектів. Зокрема, важливо підтримувати оптимальну швидкість завантаження, адаптувати вебсайт під мобільні пристрої, покращити інтерфейс та навігацію. Додатково, використання SEO-прийомів, таких як ключові слова та оптимізація мета-тегів, сприяє привертанню органічного трафіку. Крім того, важливо розглядати можливості багатомовності та локалізації, щоб забезпечити доступ користувачам різних ринків. Оптимізація вебсайту вимагає постійного моніторингу та вдосконалення, спрямованого на створення позитивного користувацького досвіду та підвищення конверсії. Варто зазначити, що у процесі оптимізації вебсайту важливо враховувати якість контенту, оскільки інформаційний матеріал впливає на сприйняття та зацікавленість відвідувачів. Зменшення кількості зайвих елементів та використання чистого дизайну також може поліпшити візуальний вигляд та сприяти зосередженню на ключових елементах.

Особливу увагу слід приділяти важливості безпеки вебсайту. Використання шифрування даних, встановлення сертифікату безпеки (SSL) та застосування двофакторної аутентифікації допомагає зберегти конфіденційність і надійність для користувачів.

Нарешті, впровадження механізмів збору та аналізу даних про відвідувачів, таких як використання аналітичних інструментів, допомагає розуміти поведінку користувачів та виявляти області для подальших вдосконалень. Аналіз конверсій, вивчення відмов та слідкування за покупками є важливими аспектами для ефективного управління та підвищення результативності онлайн-торгівлі.

Ще одним ефективним методом оптимізації вебсайту, про який точно варто додати кілька слів є використання технік A/B-тестування. Цей підхід

дозволяє порівнювати різні варіанти дизайну, контенту чи функціональності, визначаючи, який з них впливає на конверсію найбільше ефективно. А/В-тестування надає можливість точно визначити, які зміни призводять до кращих результатів і приймати рішення на основі об'єктивних даних.

Подальшим аспектом є впровадження системи зворотного зв'язку від користувачів. Збір відгуків та врахування їхніх побажань чи зауважень дозволяє виправити можливі недоліки та вдосконалити вебсайт відповідно до реальних потреб аудиторії.

Важливо також розглядати питання соціальної взаємодії, інтегруючи функції обміну відгуками, коментування та спільного використання контенту через соціальні мережі. Це сприяє створенню спільноти користувачів та залучає нових відвідувачів через рекомендації та взаємодію в мережі.

Враховуючи динаміку технологічного розвитку, використання новітніх технологій, таких як штучний інтелект для персоналізації контенту чи використання віртуальної реальності для покращення користувацького досвіду, може визначати конкурентоспроможність вебсайту в сучасному електронному бізнесі.

1.2.2 Персоналізація та рекомендації

Персоналізація вебсайту визначається як стратегія, що передбачає індивідуалізацію взаємодії з користувачами на основі їхніх особистих властивостей та попереднього поведінкового аналізу. Основною метою персоналізації є створення унікального досвіду для кожного відвідувача, спрямованого на задоволення його конкретних потреб та очікувань.

Процес персоналізації включає в себе аналіз даних користувачів, таких як історія переглядів, покупки, частота відвідування, а також демографічні характеристики. Ці дані використовуються для створення індивідуальних рекомендацій, які можуть включати товари, послуги чи контент, що відповідає унікальним інтересам та потребам кожного користувача.

Важливим аспектом персоналізації є використання алгоритмів машинного навчання, які допомагають прогнозувати, які товари чи послуги можуть бути найбільш цікавими для конкретного користувача. Ці алгоритми враховують попередні взаємодії та звички користувача, підвищуючи точність рекомендацій з кожним відвідуванням.

Результатом успішної персоналізації є не лише підвищення конверсії, а й збільшення лояльності користувачів. Вони сприймають вебсайт як індивідуально адаптоване до своїх потреб місце, що сприяє позитивному враженню та залученню на тривалий термін. Персоналізація вебсайту стає необхідною складовою стратегії електронної комерції, спрямованою на підвищення конкурентоспроможності та задоволення зростаючих очікувань сучасних користувачів.

Вдосконалення процесу персоналізації може включати в себе не лише рекомендації товарів, а й індивідуально налаштовані акції, знижки та спеціальні пропозиції, спрямовані безпосередньо на конкретного користувача. Це може створювати враження унікальних персональних вигід та підтримувати залученість.

Підходи до персоналізації можуть також включати адаптацію мови та валюти в залежності від місцезнаходження користувача, щоб забезпечити максимальний комфорт та зручність взаємодії[2]. Динамічне оновлення контенту в залежності від контексту та актуальних подій також входить у ряд ефективних стратегій.

Додатково, важливим елементом є створення персоналізованих електронних листів та повідомлень, які враховують індивідуальні інтереси та активність користувача. Це сприяє підвищенню ефективності рекламних кампаній та сприяє збільшенню участі.

Загалом, персоналізація є важливою стратегією для ефективної онлайн-торгівлі, яка дозволяє створювати індивідуальний та значущий досвід для кожного користувача, сприяючи покращенню конверсії та взаємодії з брендом.

До того ж ще одним аспектом в контексті персоналізації є використання інтерактивних елементів, таких як відгуки, рейтинги товарів та можливість залишити власний відгук. Це сприяє створенню спільноти користувачів, яка обмінюється думками та рекомендаціями, що має значний вплив на рішення покупців.

Індивідуально налаштовані "особисті кабінети" користувачів, де вони можуть зберігати свої улюблені товари, ведення списків бажань чи перегляд історії покупок, також є важливою частиною персоналізації. Це дозволяє створити персональний простір для кожного клієнта, що зберігає та відображає їхні індивідуальні вподобання.

Подальшим розвитком стратегії персоналізації є використання технологій розширеної реальності (AR) та віртуальної реальності (VR). Це може включати можливість випробування товарів в онлайн-режимі, перегляду виробу в просторовому форматі чи навіть взаємодії з продуктами у віртуальному середовищі перед покупкою.

Взагалі, персоналізація вебсайту відображає стратегічний підхід до забезпечення не тільки ефективності у сфері електронної комерції, але й побудови стійкого та значущого відношення з кожним користувачем.

Також в контексті персоналізації є використання контекстної реклами та персоналізованих пропозицій прямо на головній сторінці вебсайту. Враховуючи попередні покупки та інтереси користувача, можна надавати індивідуально орієнтовані акції та пропозиції, що підвищує ймовірність їх прийняття.

Створення динамічних лендінг-сторінок, які автоматично адаптуються до властивостей та інтересів відвідувача, також є частиною розвитку персоналізації. Це дозволяє створювати унікальний маршрут для кожного користувача, спрямований на оптимізацію конверсії та задоволення його конкретних потреб.

Розширення можливостей персоналізації також може включати в себе використання голосових та чат-ботів, які взаємодіють із користувачами, надаючи індивідуальні консультації та рекомендації.

Важливим етапом є врахування етичних аспектів персоналізації, таких як захист конфіденційності даних та можливість користувачів контролювати рівень персоналізації, щоб уникнути негативного сприйняття або відчуття вторгнення у приватність.

Узагальнюючи можна сказати що, стратегія персоналізації постійно розвивається, охоплюючи нові технології та засоби взаємодії, щоб створювати максимально індивідуалізований та задовільний досвід для кожного користувача і працювати з кожним клієнтом, з кожним користувачем для отримання максимальних результатів.

1.2.3 Маркетинг впливів

Маркетинг впливів — це стратегія маркетингу, яка базується на використанні впливових осіб чи авторитетних особистостей для підвищення усвідомленості бренду, збільшення продажів та взаємодії з цільовою аудиторією. Цей підхід використовує впливових людей, які мають значущий вплив та авторитет у своєму сегменті ринку, для ефективного сприяння бренду чи продукту.

Однією з ключових переваг маркетингу впливів є вбудована довіра, яку мають прихильники впливових осіб до своїх "кумирів". При цьому впливові особи можуть бути блогерами, особистостями у соціальних мережах, експертами в конкретних галузях, або навіть знаменитостями. Вони здатні створювати зв'язок з аудиторією та впливати на її рішення.

Стратегія маркетингу впливів включає в себе пошук та вибір впливових осіб, створення співпраці та контенту, аналіз та вимірювання результатів та будівництво довгострокових відносин. Ефективний маркетинг впливів не обмежується одноразовими кампаніями, але спрямований на будівництво стійкої репутації бренду та постійний вплив на цільову аудиторію.

Важливо відзначити, що маркетинг впливів стає особливо ефективним у віці цифрових технологій та соціальних мереж, де впливові особи мають

можливість донести свої повідомлення до широкої аудиторії. Бренди використовують цей підхід для залучення нових клієнтів, підвищення свого обличчя та зміцнення позитивного сприйняття у споживачів.

Окрім того, маркетинг впливів відкриває можливості для творчих стратегій та нестандартних підходів до реклами. Співпраця з впливовими особами дозволяє брендам подавати свої повідомлення через призму особистого досвіду та персонального підходу, що часто виявляється більш ефективним для споживачів, порівняно із стандартними рекламними кампаніями.

У світі, насиченому інформацією, маркетинг впливів стає не тільки інструментом залучення уваги, але й також важливим і функціональним засобом для побудови відмінної репутації та установлення довіри від споживачів, що безпосередньо впливає на успіх майбутнього розроблюваного програмного продукту.

1.2.4 Використання аналітики та досліджень

Використання аналітики та досліджень є невід'ємною частиною сучасного інтернет-середовища. Цей підхід надає можливість отримувати об'єктивні дані необхідні для прийняття інформованих рішень про подальшу розробку або доопрацювання ресурсу.

Аналітика включає в себе збір, аналіз та інтерпретацію даних з різних джерел. Це може стосуватися аналітики для оцінки ефективності вебсайтів, аналізу соціальних мереж для визначення реакції аудиторії, чи досліджень ринку для розуміння тенденцій і потреб клієнтів.

Дослідження включає в себе систематичне збирання та аналіз інформації для отримання нових знань та розуміння ринкових умов. Це може бути дослідження можливостей конкурентів, вивчення потреб споживачів, або оцінка ефективності нового продукту на ринку.

Використання аналітики та досліджень на вебсайтах є ключовим елементом сучасного управління та оптимізації віртуальних платформ.

Аналітичні інструменти надають можливість збирати, аналізувати та інтерпретувати великі обсяги даних, спрямованих на розуміння поведінки користувачів та оптимізацію функціонування сайту.

Дослідження та аналітика дозволяють отримувати глибоке розуміння того, як відвідувачі взаємодіють із вебсайтом, відкриваючи можливості для вдосконалення користувацького досвіду та підвищення ефективності функціоналу. Важливим аспектом є аналіз конверсій, де вивчаються кроки, які ведуть до досягнення конкретних цілей, таких як покупка товару чи підписки.

Аналітичні засоби також допомагають відстежувати рекламні кампанії та їх вплив на залучення нових відвідувачів. Це дозволяє оптимізувати рекламні зусилля та вкладення коштів у ті канали, які найбільше сприяють досягненню поставлених цілей[3].

Додатково, аналітика може включати в себе вивчення демографічних характеристик аудиторії, що допомагає персоналізувати контент та пристосовувати стратегії маркетингу. Дослідження користувацької взаємодії та зворотного зв'язку важливі для виявлення слабких сторін вебсайту та можливостей для вдосконалення.

Аналітика надає можливість виявляти та аналізувати тренди в поведінці користувачів, що є важливим для адаптації сайту до змінних потреб аудиторії.

Однією з ключових складових є аналіз взаємодії на різних типах пристроїв, що дозволяє оптимізувати вебсайт для різних платформ та забезпечити зручний досвід для мобільних та настільних користувачів. Це особливо важливо в епоху мобільного користування, де зростає кількість відвідувачів, які взаємодіють з веб-ресурсом через смартфони та планшети.

Проведення А/В тестувань є ефективним методом для визначення того, які зміни на сайті мають позитивний вплив на користувачів. Це може охоплювати варіації дизайну, структури меню, чи текстового контенту. Аналіз результатів тестувань надає конкретні дані для прийняття управлінських рішень.

Крім того, використання аналітики в сфері електронної комерції дозволяє відстежувати покупкові поведінки користувачів, вивчати їхні вподобання та створювати персоналізовані пропозиції, що сприяє збільшенню конверсії.

В цілому, аналітика та дослідження на вебсайтах є невід'ємною частиною стратегії ведення бізнесу в онлайн середовищі, де постійна адаптація та вдосконалення стають важливими факторами для досягнення успіху.

1.2.5 Штучний інтелект та аналіз даних

Штучний інтелект (ШІ) – це галузь комп'ютерних наук, яка вивчає створення систем, які можуть виконувати завдання, які зазвичай вимагають людського інтелекту. Основна мета ШІ полягає в створенні програм та алгоритмів, що надають комп'ютерам можливість розв'язувати проблеми, робити висновки, вчитися з досвіду та виконувати інші завдання, які зазвичай вимагають людської інтелектуальної діяльності.

Штучний інтелект (ШІ) та аналіз даних в сфері електронної комерції представляють потужний інструментарій, спрямований на оптимізацію функцій вебсайтів та поліпшення взаємодії з користувачами. Ці технології дозволяють бізнесам отримувати глибокі інсайти, автоматизувати процеси та персоналізувати взаємодію з клієнтами.

Штучний інтелект використовується для аналізу великих обсягів даних, щоб виявляти патерни та тенденції у поведінці користувачів. Алгоритми машинного навчання можуть адаптувати рекомендації та пропозиції на основі індивідуальних виборів, покупок та переглядів, надаючи користувачам персоналізований досвід.

Аналіз даних включає в себе обробку, інтерпретацію та використання інформації для прийняття стратегічних рішень. В онлайн-торгівлі це означає ретельне вивчення даних щодо відвідувань, конверсій, витрат та інших метрик, щоб зрозуміти ефективність стратегій та виправити можливі недоліки.

ІІІ та аналіз даних також допомагають у вирішенні завдань управління запасами, прогнозуванні попиту та вдосконаленні логістики. Вони дозволяють автоматизувати процеси визначення оптимальних цін, а також підбору та рекомендацій товарів, що сприяє зростанню конверсії.

Штучний інтелект та аналіз даних революціонізують індустрію онлайн-торгівлі через свою здатність ефективно обробляти великі обсяги інформації та витягати цінні інсайти для прийняття рішень[4]. Один з ключових аспектів цих технологій - це автоматизація багатьох рутинних завдань, що дозволяє бізнесам фокусуватися на стратегічних аспектах розвитку.

Інтелектуальні системи можуть аналізувати поведінку користувачів та прогнозувати їхні потреби, що дозволяє персоналізувати пропозиції та рекомендації. Наприклад, система може автоматично пропонувати товари, які ймовірно зацікавлять конкретного покупця на основі його попередніх виборів.

Аналіз даних використовується для виявлення та аналізу моделей споживання, що дозволяє підприємствам адаптувати свою стратегію до змін у попиті. Велика кількість даних, які генеруються в процесі онлайн-торгівлі, може бути використана для прогнозування трендів та оптимізації запасів.

Застосування ІІІ та аналізу даних в сфері обробки замовлень та логістики допомагає вдосконалити розподіл та доставку товарів, зменшуючи терміни обробки та підвищуючи точність управління запасами. Такі системи дозволяють визначати оптимальні маршрути доставки та стратегії розміщення товарів на складі.

Штучний інтелект та аналіз даних в онлайн-торгівлі взаємодіють для створення інтелектуальних та високоефективних стратегій. ІІІ забезпечує автоматизацію обробки даних, виявлення складних взаємозв'язків та вдосконалення рішень, в той час як аналіз даних використовує ці інсайти для покращення різноманітних аспектів бізнесу.

Один із важливих аспектів застосування ІІІ та аналізу даних - це підвищення персоналізації. Системи можуть аналізувати історію покупок,

переглядів та поведінки користувачів для створення індивідуальних рекомендацій та адаптації інтерфейсу вебсайту під конкретного відвідувача.

ІІІ також допомагає в управлінні ризиками та забезпеченні безпеки. Системи можуть виявляти підозрілу активність, виявляти шахраїв та автоматизовані загрози безпеки. Аналіз даних в той же час може використовувати історичні дані для передбачення можливих ризиків та прийняття запобіжних заходів.

Оптимізація цінової політики та маркетингових стратегій є ще однією областю, де ІІІ та аналіз даних виявляються надзвичайно ефективними. Вони дозволяють адаптувати ціни в реальному часі, враховуючи конкурентне середовище та попит, а також створювати точніші та спрямовані маркетингові кампанії.

1.2.6 Мобільна оптимізація

Мобільна оптимізація вебсайту є критично важливим аспектом в умовах зростаючого використання мобільних пристроїв. Ця стратегія враховує особливості та обмеження мобільних пристроїв для покращення користувацького досвіду та ефективності взаємодії з вебсайтом.

Мобільна оптимізація включає в себе адаптацію дизайну вебсайту до різних розмірів екранів мобільних пристроїв, забезпечення швидкості завантаження сторінок та зручного навігаційного досвіду. Одним із ключових аспектів є респонсивний дизайн, який дозволяє адаптувати вміст вебсайту до конкретного розміру екрану, забезпечуючи оптимальне візуальне відображення на будь-якому пристрої.

Зменшення розміру графічних елементів та ресурсів, таких як зображення та відео, допомагає забезпечити швидке завантаження сторінок на мобільних пристроях з обмеженим доступом до мережі. Крім того, кешування контенту та оптимізація коду можуть покращити ефективність завантаження.

Для покращення навігації на мобільних пристроях важливо ретельно розробляти меню, розділи та інші елементи інтерфейсу, щоб забезпечити зручний та інтуїтивний доступ до важливої інформації. Важливо також враховувати відмінності в способі взаємодії користувачів з екранами сенсорних пристроїв.

Також, ефективне використання метатегів та заголовків допомагає покращити SEO на мобільних пристроях, що важливо для підвищення видимості в пошукових системах та привертання нових відвідувачів.

Урахування контексту використання мобільних пристроїв також є важливим елементом стратегії мобільної оптимізації. Наприклад, забезпечення можливості швидкого зворотнього зв'язку через форми зворотного зв'язку, використання клікабельних телефонних номерів для зворотного дзвінка та інші елементи, що спрощують взаємодію на мобільних пристроях.

Не менш важливим є тестування вебсайту на різних мобільних пристроях та браузерах, щоб впевнитися, що оптимізація працює на різних платформах. Це включає в себе перевірку як на популярних, так і на менш популярних пристроях для забезпечення універсальності та доступності.

Ще однією важливою частиною стратегії мобільної оптимізації є управління рекламою та маркетингом на мобільних пристроях. Важливо розробляти рекламні кампанії, які адаптовані до мобільного середовища, з урахуванням розмірів екрану та особливостей взаємодії з рекламним контентом на мобільних пристроях.

Застосування push-сповіщень та мобільних додатків також може значно покращити взаємодію з аудиторією на мобільних пристроях. Забезпечення зручного та відповідного досвіду користувача через мобільні додатки може стати ключовим фактором в залученні та утриманні клієнтів.

Також, важливо враховувати геолокаційні можливості мобільних пристроїв для персоналізації взаємодії та надання користувачам релевантної інформації. Аналіз поведінки користувачів з урахуванням місцезнаходження

може допомогти у вдосконаленні стратегій маркетингу та збільшенні ефективності рекламних кампаній[5].

Нарешті підсумовуючи можна виділити що, важливим елементом є забезпечення доступності, тому що мобільні пристрої використовуються людьми з різними потребами та обмеженнями. Адаптація вебсайту таким чином щоб він був доступним для всіх користувачів, включаючи тих, хто використовує допоміжні технології, є ключовим аспектом сучасної мобільної оптимізації. підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту

1.3 Порівняльний аналіз існуючих методів вирішення завдання підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту

З минулого пункту можна зрозуміти наявну кількість вже існуючих методів, за допомогою яких можна вирішувати поставлене завдання і це включає дослідження та оцінку різних стратегій та підходів, що використовуються для оптимізації електронної комерції. Основна мета порівняльного аналізу полягає в ідентифікації найбільш ефективних та пристосованих до конкретного вебсайту методів та розумінні, який саме метод має найбільше переваг для конкретного випадку. Звичайно коли нам стало відомо, що мета порівняльного аналізу полягає в ідентифікації найбільш ефективних та пристосованих до конкретного вебсайту методів та розумінні, який саме метод має найбільше переваг для конкретного випадку нам треба безпосередньо отримати результат такого порівняння. Для того, щоб отримати такий результат нам необхідно провести порівняльний аналіз і віднайти найкращі та найсильніші сторони кожного методу. Це дуже важливий етап, тому що саме він може дати нам розуміння про те в який бік ми можемо рухатись. Проаналізувавши існуючі методи вирішення завдання було проведено порівняльний аналіз досліджених методів для прогнозування підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту(табл.1.1).

Таблиця 1.1 – Порівняльна характеристика існуючих методів для підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту

Можливості	Оптимізація вебсайту	Персоналізація	Маркетинг впливів	Аналітика	Штучний інтелект
Простота реалізації	Так	Так	Ні	Так	Ні
Автоматизація	Ні	Так	Ні	Так	Так
Гнучкість	Ні	Ні	Ні	Ні	Так
Маштабованість	Ні	Так	Ні	Ні	Так
Безпека	Так	Так	Ні	Так	Так
Користувацький досвід	Так	Так	Так	Ні	Ні
Легкість інтерпретації одержаних результатів	Так	Ні	Ні	Так	Ні
Адаптація до змін у вирішуваному завданні	Ні	Ні	Ні	Так	Так
Конверсія	Ні	Так	Ні	Ні	Ні
Реакція на зміну	Так	Так	Так	Ні	Ні

Провівши порівняльний аналіз досліджених методів для прогнозування підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту (табл. 1.2) було встановлено, що використання машинного навчання та персоналізації дійсно допомагає підвищити конверсію та збільшити середній чек. Оптимізація веб-дизайну та взаємодії, також, суттєво може поліпшити користувацький досвід і збільшити залученість користувачів.

Впровадження штучного інтелекту у клієнтському обслуговуванні, зокрема чат-ботів та аналіз даних взаємодії з клієнтами, зменшило б час

очікування та підвищило б рівень задоволеності клієнтів. Використання аналітики для оптимізації запасів та динамічного ціноутворення сприяло би підвищенню конкурентоспроможності.

Співпраця з платіжними системами, включаючи інтеграцію швидких та безпечних оплат, призвела б до збільшення кількості успішних транзакцій та підвищила б рівень довіри користувачів. Комбінація цих методів виявилася оптимальною для досягнення цілей підвищення ефективності онлайн-торгівлі на вивченому вебсайті. Важливим є постійне вдосконалення та моніторинг результатів для адаптації стратегій до змін у бізнес-середовищі та потребах користувачів.

1.4 Дослідження інформаційних систем

1.4.1 CRM-системи

Системи управління відносинами з клієнтами (CRM) - це комплексні інформаційні системи, призначені для ефективного збору, аналізу та управління даними про клієнтів та їх взаємодією з компанією. Основна мета CRM-систем полягає в покращенні взаємодії з клієнтами та оптимізації всіх аспектів бізнес-процесів, пов'язаних із залученням та обслуговуванням клієнтів.

Інтеграція з системами управління відносинами з клієнтами (CRM) представляє собою важливий аспект оптимізації процесу онлайн-торгівлі. Цей підхід передбачає сполучення функціоналу електронної комерції із засобами, які дозволяють збирати, аналізувати та використовувати дані про клієнтів для покращення взаємодії та ефективності бізнес-процесів.

Інтеграція з CRM-системами дозволяє автоматизувати та оптимізувати управління клієнтською базою та взаємодію з клієнтами в режимі реального часу. Збір та обробка інформації про покупки, перегляди товарів, поведінку на сайті і взаємодію з рекламою дозволяє створити повний образ клієнта. Це допомагає визначити індивідуальні потреби та вподобання кожного клієнта.

За допомогою інтеграції CRM-системи з онлайн-торговою платформою, ми отримуємо можливість автоматизувати процеси обробки замовлень, відстеження стану доставок та надання клієнтам персоналізованих рекомендацій. Це дозволяє знижувати час обробки та покращує загальний обслуговування клієнтів.

Крім того, інтеграція дозволяє вести детальний облік історії взаємодії з кожним клієнтом, включаючи консультації, звернення до служби підтримки та участь у рекламних акціях. Це створює можливість для більш глибокого розуміння потреб та попередження можливих проблем.

Аналіз даних, що збираються через систему CRM, дозволяє набагато легше ідентифікувати цільову аудиторію та налаштовувати персоналізовані пропозиції.

Однією з ключових переваг інтеграції є здатність виявлення нових можливостей та тенденцій через аналіз даних про клієнтів. Це дає можливість швидко реагувати на зміни в ринкових умовах та адаптувати стратегії збуту.

Завдяки централізованому зберіганню даних, інтеграція з CRM допомагає уникнути втрати інформації та забезпечує доступ до актуальних даних у будь-який момент[6]. Це важливо для прийняття обґрунтованих рішень та ефективного управління.

Інтеграція з CRM-системами може також включати аналітику здатності клієнта, що дозволяє визначати вплив різних маркетингових заходів на лояльність та участь клієнтів в програмах знижок.

Ще однією важливою перевагою інтеграції з CRM-системами є можливість використання автоматизованих процесів управління продажами. Спільне використання даних про клієнтів та їхню історію взаємодії дозволяє ефективно відслідковувати стадії угод та управляти потоком продажів.

Також важливо відзначити, що інтеграція з CRM дозволяє покращити взаємодію між різними відділами компанії, зокрема між відділами продажів, маркетингу та обслуговування клієнтів. Це забезпечує єдиний погляд на клієнта та сприяє збільшенню комунікації всередині організації.

Окрім того, інтеграція з CRM може включати функціонал аналізу ефективності кампаній та промо-заходів. Аналіз даних дозволяє визначати ті стратегії, які найбільш ефективно привертають та утримують клієнтів, що є ключовим для подальшого розвитку проєкту.

Загалом, інтеграція з CRM-системами детально оптимізує управління взаємодією з клієнтами, робить проєкт більш гнучким у відповідь на потреби ринку та дозволяє підтримувати високий рівень обслуговування, що є важливим фактором успіху в галузі онлайн-торгівлі.

1.4.2 Електронна система управління запасами (ERP)

Електронна система управління запасами (ERP) є інтегрованою інформаційною системою, спрямованою на автоматизацію та управління різнобічними аспектами процесів у сфері логістики, виробництва, фінансів, управління персоналом та іншими діяльностями в організації. Основна мета ERP-систем - це створення єдиної інтегрованої платформи, яка об'єднує дані та процеси всіх відділів підприємства.

Ця система дозволяє підприємствам ефективно управляти ресурсами, використовуючи централізовану базу даних, яка забезпечує швидкий доступ до інформації та покращує комунікацію між всіма рівнями організації. ERP-система зазвичай включає в себе модулі для обліку фінансів, управління запасами, виробництва, замовлень, управління взаємовідносинами з клієнтами (CRM) та багато інших функцій.

Важливим аспектом ERP є стандартизація та оптимізація процесів пов'язаних з проєктами. Вона дозволяє підприємствам ефективно взаємодіяти з різними ділянками діяльності, зменшуючи час на адміністративні операції та покращуючи оперативні рішення.

ERP-системи можуть бути використані в різних секторах бізнесу, включаючи виробництво, роздрібну торгівлю, логістику, і що найголовніше онлайн-торгівлю. Ці системи допомагають підприємствам підтримувати

конкурентоспроможність, покращувати ефективність та сприяти інноваціям у своїй галузі.

Електронна система управління запасами (ERP) є комплексним інструментом, призначеним для автоматизації та оптимізації управління всіма аспектами ланцюга постачання та запасами в організації[7]. Ця система об'єднує в собі різні функції, включаючи управління запасами, закупівлями, виробництвом, фінансами та іншими бізнес-процесами.

Запаси є важливим ресурсом для проєктів пов'язаних з онлайн-торгівлею, і їх ефективне управління є ключовим для забезпечення неперервності виробництва та задоволення потреб клієнтів. ERP-система виступає в ролі централізованого механізму, що дозволяє організації отримати повний контроль над запасами та пов'язаними з ними процесами.

Однією з ключових функцій ERP є автоматизація процесів управління запасами, включаючи моніторинг рівня запасів, прогнозування попиту, оптимізацію замовлень та управління постачальниками. Це дозволяє уникати перевантажень або недостачі запасів, оптимізувати витрати та забезпечити своєчасність виробництва та постачань.

ERP також забезпечує інтеграцію з іншими процесами, такими як фінанси та виробництво. Це дозволяє отримати повний обсяг інформації про весь життєвий цикл продукції — від постачання сировини до постачання готової продукції клієнтам.

Застосування ERP-систем також покликане підвищити ефективність праці та зменшити ризики людського фактору, оскільки багато рутинних завдань можуть бути автоматизовані. Це допомагає витратити менше часу на адміністративні процеси та більше часу на стратегічне планування та розвиток проєкту.

Однією з переваг ERP є можливість створення єдиної бази даних, яка об'єднує інформацію з усіх розділів та функцій. Це дозволяє уникати дублювання даних та покращує координацію між різними відділами.

ERP також враховує різні аспекти управління запасами, такі як контроль якості, сертифікація та забезпечення дотримання регуляторних вимог. Це особливо важливо для компаній у секторах онлайн-торгівлі, де вимоги до якості та безпеки є високими.

Крім того, ERP-системи дозволяють здійснювати ефективний моніторинг та аналіз руху матеріальних ресурсів в реальному часі. Це допомагає оперативно реагувати на зміни в попиті, управляти запасами та оптимізувати логістичні процеси.

1.4.3 Аналітична платформа Google Analytics

Google Analytics є потужним інструментом для аналізу та вимірювання ефективності вебсайту в контексті онлайн-торгівлі. Цей сервіс дозволяє вивчати дані про відвідувачів, їх поведінку та взаємодію з вебсайтом. Важливий аспект для нас - аналіз конверсій, де можна відстежувати та прогнозувати ефективність стратегій продажів. Також, інструмент дозволяє визначити ефективні джерела трафіку, щоб оптимізувати маркетингові зусилля.

Сегментація аудиторії дозволяє персоналізувати стратегії маркетингу та обслуговування, а аналіз відкликання товарів вказує на їхню популярність. Google Analytics також вимірює швидкість завантаження сторінок, що впливає на користувацький досвід і є важливим для оптимізації.

Окремо варто зазначити, що Google Analytics дозволяє аналізувати відвідуваність вебсайту, щоб зрозуміти, з яких регіонів приходить аудиторія, які пристрої вони використовують, та як вони взаємодіють з контентом. Це надає можливість оптимізувати вебсайт для різних типів користувачів.

Важливим аспектом є аналіз вихідних джерел трафіку, що дозволяє визначити ефективність каналів привертання відвідувачів. Також, вимірювання часу, який користувачі витрачають на сторінках, і вивчення шляхів їхньої взаємодії допомагають визначити, як поліпшити користувацький досвід.

Google Analytics дозволяє розглядати відомості про користувачів з точки зору їхньої взаємодії з конкретним контентом. Аналіз популярних сторінок та продуктів може вказати на те, що привертає увагу аудиторії. Це дозволяє адаптувати стратегії контентного маркетингу та вивчати, які елементи приводять до більшої активності користувачів.

Також, Google Analytics допомагає визначати ефективність мобільної оптимізації вебсайту. Аналіз трафіку з мобільних пристроїв та взаємодії користувачів з мобільним контентом дозволяє визначити, наскільки добре оптимізований вебсайт для мобільних платформ.

Google Analytics дозволяє глибоко аналізувати аудиторію та розуміти її характеристики, такі як місце розташування та тип пристрою. Здійснюється вивчення поведінки користувачів, включаючи перегляд сторінок та шляхи взаємодії. Можливість вимірювати конверсії допомагає оцінювати ефективність стратегій продажу.

Аналіз вихідних джерел трафіку дозволяє визначати, які канали привертають найбільше уваги, для подальшої оптимізації маркетингових зусиль[8]. Засіб сегментації аудиторії дозволяє персоналізувати стратегії відповідно до різних груп користувачів.

Як переваги використання сервісу можна виділити аналіз відкликання товарів, тому що він вказує на їхню популярність, що вже в свою чергу дозволяє оптимізувати асортимент та підбирати ефективні рекомендації. Також, Google Analytics вимірює час завантаження сторінок, впливаючи на загальний користувацький досвід. Все це дозволяє прогнозувати та оптимізовувати електронний бізнес для досягнення високої ефективності в онлайн-торгівлі.

Щодо недоліків звичайно слідкування за відвідувачами або певний збір даних може викликати етичні питання щодо приватності. Також, затримка оновлення може призвести до ускладнень у отриманні актуальної інформації. Для деяких користувачів може виникнути певна складність використання, особливо для новачків, через велику кількість параметрів. Додатковими недоліками можуть бути неоптимальна точність даних, особливо при

неадекватній налаштуванні коду вебсайту, обмеження у безкоштовній версії, зокрема щодо обсягу зібраних даних, залежність від Інтернет-з'єднання для коректної роботи і звичайно ж недостатня інтеграція з іншими інструментами аналізу даних.

1.4.4 Magento

Magento - це потужна платформа для створення і управління вебсайтами електронної комерції, яка надає ряд інструментів для прогнозування та підвищення ефективності процесів онлайн-торгівлі.

Magento, як потужна платформа електронної комерції, дозволяє прогнозувати та підвищувати ефективність онлайн-торгівлі на прикладі вебсайту за допомогою аналізу конверсій, сегментації аудиторії, відстеження популярності товарів, веб-аналітики та різноманітних маркетингових інструментів.

Перш за все, вона дозволяє детально вивчати конверсії, враховуючи кожен етап воронки продажів, та оптимізовувати шлях користувача. Сегментація аудиторії у Magento допомагає створювати персоналізовані стратегії маркетингу для різних сегментів користувачів.

Також, платформа дозволяє детально відслідковувати популярність товарів і категорій, що вказує на попит та допомагає управляти асортиментом товарів. Magento легко інтегрується з різними системами веб-аналітики, надаючи можливість докладно аналізувати трафік та поведінку користувачів.

Додатково, платформа пропонує широкий набір маркетингових інструментів, таких як розсилки, знижки, купони, що дозволяє створювати привабливі пропозиції та підвищувати залучення аудиторії.

Magento виокремлюється своєю високою функціональністю та низкою переваг, які сприяють успішному розвитку електронної комерції[9]. Його відкритий код надає безліч можливостей для кастомізації та адаптації до конкретних потреб бізнесу. Magento підтримує створення потужних та

масштабованих інтернет-магазинів, забезпечуючи високий рівень контролю над процесами. Його широкий функціонал включає аналіз конверсій, сегментацію аудиторії, ефективне відстеження товарів та інтеграцію з аналітичними інструментами.

Платформа підтримує маркетингові ініціативи через різноманітні інструменти, такі як розсилки та знижки, що сприяє створенню привабливих пропозицій для клієнтів. Magento також володіє гнучкою системою розширень, яка дозволяє розширювати функціонал без значних зусиль.

Magento, несумлінно, є потужною платформою для електронної комерції, але вона також має свої недоліки. Наприклад, складність встановлення та налаштування може становити виклик для користувачів, особливо для тих, хто не має великого досвіду у сфері веб-розробки. Також, Magento може бути вимогливою до ресурсів, і для оптимальної продуктивності потребує потужного хостингу.

Недоліками є також неоптимальна точність даних у випадках неадекватної настройки коду вебсайту, а також складність для новачків у використанні ряду функцій та параметрів платформи. Деякі користувачі вказують на затримки оновлень та можливість зіткнення з проблемами швидкодії через розширені можливості Magento, що, у свою чергу, може призвести до складнощів у використанні для менших бізнесів.

Ще одним недоліком є обмежені можливості безкоштовної версії Magento, і для повного використання всіх функцій платформи може знадобитися певна інвестиція в платні розширення та підтримку. Важливо також враховувати, що Magento вимагає стійкого Інтернет-з'єднання для нормальної роботи, що може бути проблемою у випадку непостійного з'єднання.

Незважаючи на ці недоліки, Magento залишається потужним інструментом для електронної комерції, і багато з цих проблем можуть бути вирішені за допомогою правильного налаштування платформи.

1.5 Порівняння існуючих систем дослідження, що можуть бути використані для підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту

Сучасний електронний комерційний ландшафт розвивається зі швидкістю блискавки, перетворюючи традиційний спосіб торгівлі та створюючи нові можливості для бізнесу. Одним із ключових аспектів цього трансформаційного процесу є впровадження онлайн-торгівлі, яка надає споживачам широкий вибір продуктів та послуг без необхідності фізичного відвідування магазинів. Для підвищення ефективності процесу онлайн-торгівлі існують різноманітні системи дослідження, які можуть бути використані для аналізу та оптимізації діяльності вебсайтів. Підвищення ефективності процесу онлайн-торгівлі вимагає використання різних систем дослідження для збору та аналізу різноманітної інформації. При виборі системи для вебсайту онлайн-торгівлі важливо враховувати специфіку бізнес-потреб, розмір сайту, технічні можливості та бюджет. Комбінування кількох інструментів може допомогти отримати повний обсяг даних для ефективного вдосконалення процесу онлайн-торгівлі.

Серед вже існуючих систем дослідження, що можуть бути використані для підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту, за допомогою яких можна вирішувати поставлене завдання треба за допомогою порівняння обрати ті які найкраще можуть вирішити поставлене завдання. Основна мета порівняльного аналізу полягає в ідентифікації найбільш ефективних та пристосованих до конкретного вебсайту системи та розумінні, яка саме з систем має найбільше переваг для конкретного випадку.

Проаналізувавши всі існуючі системи вирішення завдання було проведено порівняльний аналіз досліджених систем для прогнозування підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту(табл.1.2).

Таблиця 1.2 – Порівняльна характеристика існуючих програмних засобів

Можливості	CRM	ERP	Gogle Analytics	Magento
Ліцензія	Ні	Ні	Так	Ні
Операційна система	Кросплат форменне	Windows, Linux, macOS	Windows, Linux, macOS	Windows, Linux, macOS
Доступність	Так	Так	Так	Ні
Наявність пробної версії	Так	Так	Так	Так
Можливість аналізу великих даних	Ні	Ні	Так	Так
Інтелектуальний аналіз даних	Ні	Так	Так	Ні
Візуалізації даних	Так	Ні	Так	Так
Можливість машинного навчання	Ні	Так	Ні	Так
Можливість адміністрування	Ні	Так	Ні	Так
Імпорт/експорт даних	Так	Так	Так	Так
Інструменти для мобільної оптимізації	Ні	Ні	Так	Так
Наявність API	Так	Так	Ні	Так
Індикація трендів та проблем	Так	Так	Так	Так
Прогнозування та передбачувана аналітика	Так	Так	Так	Так
Статистичний аналіз	Так	Так	Так	Так
Можливість доступу для багатьох користувачів	Ні	Ні	Ні	Ні

1.6 Постановка завдання до роботи

Метою роботи є підвищення ефективності процесу онлайн-торгівлі на вебсайті. Для досягнення цієї мети передбачено проведення глибокого аналізу та оптимізації різних аспектів електронної комерції, зосереджуючись на покращенні конверсій, вивченні поведінки користувачів, сегментації аудиторії та вдосконаленні маркетингових стратегій.

У роботі розглядається комплексний аналіз та оптимізація процесу онлайн-торгівлі на вебсайті. В основу дослідження покладено аналіз різних аспектів електронної комерції з метою вдосконалення стратегій та підвищення ефективності діяльності.

Детально розглядається конверсія як ключовий показник успішності вебсайту, аналізуються чинники, що впливають на неї, та виробляються рекомендації щодо покращення цього показника. Також проводиться дослідження поведінки користувачів для кращого розуміння їхніх потреб та вподобань.

У роботі вивчається сегментація аудиторії з метою розроблення персоналізованих стратегій маркетингу та обслуговування. Окрема увага приділяється оптимізації асортименту товарів на основі аналізу їхньої популярності та відгуків клієнтів.

Під час порівняльного аналізу існуючих методів було встановлено, що найбільш ефективним методом для вирішення поставленого завдання є інтеграція аналізу даних. Цей підхід відзначився найвищою точністю та адаптабельністю до змін у поведінці користувачів та ринкових умов.

Аналіз даних в контексті електронної комерції надає можливість отримувати глибокі інсайти щодо поведінки покупців, ефективності товарів, тенденцій ринку та інших важливих аспектів. Ця інформація стає основою для прийняття стратегічних рішень та оптимізації бізнес-процесів.

Отже, з метою вирішення проблем, що виникають при використанні досліджуваних методів, було прийнято рішення розробити вебсайт, який

комбінує переваги аналізу даних. Цей вебсайт буде спрямований на оптимізацію всіх аспектів електронної комерції, забезпечуючи ефективну адаптацію до змінних умов ринку та користувацьких потреб.

Розробка такого вебсайту передбачає створення алгоритмів сортування, які здатні автоматично адаптувати сторінку під бажання покупця переглядати той чи інший товар. Аналітичні інструменти будуть використовуватися для постійного моніторингу та аналізу результатів, щоб оперативно реагувати на зміни.

Цей вебсайт буде спрямований на покращення користувацького досвіду та ефективного управління асортиментом товарів. Враховуючи специфіку вебсайту та його цільову аудиторію, розробка такої системи визначається як ключовий крок у вдосконаленні ефективності процесу онлайн-торгівлі.

1.7 Висновок

У цьому розділі було проведено аналіз предметної області, визначено основні поняття, пов'язані з предметною областю і об'єктом дослідження та розробки.

Було досліджено основні методи підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту та проведено порівняльний аналіз цих методів, який показав перевагу використання аналізу даних для вирішення поставленого завдання.

Проведено порівняльну характеристику існуючих програмних засобів, що можуть бути використані створення вебсайту та дослідження підвищення ефективності процесу онлайн-торгівлі.

Було сформульовано і поставлено завдання до роботи.

2 ВИБІР ТА ОБҐРУНТУВАННЯ СТРУКТУРИ СИСТЕМИ, ЯКА ПРОЄКТУЄТЬСЯ

2.1 Вибір мови програмування

Існує дуже багато різнопланових мов програмування, кожна з яких звичайно ж має свої переваги і недоліки, і ту, яку було зручно використовувати в одному проєкті, може бути зовсім не функціональною для іншого, тому саме вибір мови програмування для розробки ПП є важливим етапом. Вибір мови програмування для сайту є ключовим етапом в процесі розробки. Для оптимального вибору слід враховувати ряд факторів, таких як тип сайту, його функціональність, масштаби проєкту та власність мови програмування. Найчастіше використовувані мови програмування для розробки сайтів якщо врахувати ці фактори, такі як JavaScript, PHP, Python, Go чи Ruby.

2.1.1 JavaScript

JavaScript є мовою програмування, яка використовується для розробки динамічних та інтерактивних вебсайтів. Вона використовується для взаємодії з користувачем, зміни вмісту сторінок, анімацій, валідації форм, роботи з асинхронними запитами на сервер та інших аспектів веб-розробки. Вона виконується на браузері клієнта і дозволяє розробникам створювати багатофункціональні та привабливі інтерфейси для користувачів.

Однією з основних характеристик JavaScript є можливість динамічно змінювати вміст сторінки без необхідності перезавантаження. Це досягається шляхом маніпулювання Document Object Model (DOM), представленням структури HTML-документа. JavaScript також взаємодіє з користувачем, обробляючи події, такі як кліки мишею або введення з клавіатури.

Мова підтримує асинхронність, що дозволяє виконувати запити до сервера без блокування інтерфейсу користувача. Це особливо важливо для отримання або відправки даних без очікування повного завантаження сторінки.

JavaScript використовується для валідації та обробки даних, введених користувачем у формах. Він також використовується для створення анімацій та динамічних ефектів, надаючи вебсайтам більшу естетичність та привабливість.

Завдяки широкій підтримці у сучасних браузерях, JavaScript став ключовим інструментом у розробці фронтенду веб-додатків. Разом із HTML та CSS, він утворює основу для створення інтерактивних та цікавих вебсайтів, а його використання може розширюватися й на розробку серверної частини додатків за допомогою платформи Node.js[9].

JavaScript володіє рядом переваг, які роблять його незамінним інструментом для веб-розробки.

JavaScript забезпечує динамічність та інтерактивність вебсайтів. За допомогою цієї мови можна динамічно змінювати вміст сторінок, відповідати на події користувача, анімувати елементи та забезпечувати користувачеві більш поглиблений та динамічний досвід взаємодії з сайтом.

Однією з ключових переваг є асинхронність. JavaScript використовує концепцію асинхронного програмування, що дозволяє виконувати операції без очікування завершення попередніх операцій. Це особливо важливо для завантаження ресурсів на сторінці без блокування інтерфейсу користувача.

Мова програмування JavaScript є клієнтською мовою, що дозволяє виконувати код безпосередньо на браузері користувача. Це спрощує взаємодію з DOM (Document Object Model) та надає змогу динамічно маніпулювати вмістом сторінки.

JavaScript підтримує роботу з AJAX, що дає змогу виконувати асинхронні HTTP-запити без перезавантаження сторінки. Це дозволяє динамічно оновлювати вміст сторінки на основі даних, отриманих з сервера.

Завдяки великому спільноті розробників та широкій підтримці, JavaScript має безліч бібліотек і фреймворків, таких як React, Angular, та Vue.js, які роблять розробку більш швидкою та ефективною.

JavaScript використовується для створення односторінкових застосунків (SPA), що дозволяє зменшити час завантаження сторінок та покращити користувацький досвід.

Ці переваги роблять JavaScript ключовим інструментом для веб-розробки, дозволяючи створювати сучасні та інтерактивні веб-додатки.

Звичайно JavaScript має і ряд недоліків, основні з яких я можу виділити безпеку - JavaScript виконується на клієнтській стороні, що робить його вразливим до атак на рівні клієнта. В деяких випадках можлива залежність від браузера - різні браузери можуть мати відмінності в реалізації JavaScript, що може викликати проблеми сумісності. Також важливою темою для нас з огляду на поставлене завдання можуть бути і SEO проблеми - деякі пошукові системи можуть мати обмежену можливість індексації та зрозуміння вмісту, який генерується за допомогою JavaScript.

Варто зазначити і про обробку помилок, так як відлагодження JavaScript вважається складнішим, особливо в тих випадках, коли помилка стається на клієнтській стороні.

Ці недоліки теж слід враховувати при розробці вебсайтів для забезпечення оптимальної продуктивності та безпеки.

2.1.2 PHP

PHP (Hypertext Preprocessor) - це скриптова мова програмування, яка використовується головним чином для розробки вебсайтів та взаємодії з базами даних. Вона вбудована безпосередньо в HTML-код та використовується для створення динамічного вмісту веб-сторінок.

PHP ідеально підходить для розробки вебсайтів, оскільки вона може взаємодіяти з HTTP-серверами і генерувати HTML-код. Вона використовує слабку та динамічну типізацію, що дозволяє легко пристосовувати код до різних сценаріїв.

PHP підтримує багато різних систем управління базами даних, таких як MySQL, PostgreSQL, SQLite та інші. Вона легко обробляє дані, введені користувачем через форми, що дозволяє розробникам легко створювати взаємодію з користувачами.

PHP також підтримує об'єктно-орієнтований підхід, що дозволяє структурувати і організовувати код більш ефективно. Завдяки широкому спектру вбудованих функцій, PHP стає ефективним інструментом для розробки[10].

Ця мова використовується для створення різноманітних вебсайтів, від невеликих особистих блогів до великих корпоративних порталів. Вона є ефективним інструментом для серверного програмування та надає можливості для розробки різноманітних веб-застосунків.

PHP є мовою програмування, широко використовуваною для розробки динамічних вебсайтів і у неї є певні переваги. По-перше це гнучкість, бо PHP може використовуватися як самостійна мова програмування, так і вбудовуватися безпосередньо в HTML-код. Сюди ж можна додати і простоту вивчення, багато хто відмічає, що PHP має низький поріг входження для новачків, і вивчення його зазвичай відбувається швидше порівняно з іншими мовами.

По-друге у мови доволі широкі можливості, мова PHP використовується для розробки різноманітних веб-застосунків, від простих сайтів до складних веб-порталів і систем управління контентом. Для розробників є велика спільнота, PHP має активну спільноту розробників, що робить його важливим інструментом для обміну знаннями та отримання допомоги.

Третьою перевагою можу виділити швидкість розробки, PHP дозволяє швидко створювати прототипи та реалізовувати функціональність вебсайтів.

PHP, хоч і є популярною мовою програмування для створення вебсайтів та веб-додатків, та звичайно, як і інші, має свої недоліки, які варто враховувати при використанні.

Першим недоліком є безпека, PHP має історію питань з безпекою, особливо якщо не дотримуватися найкращих практик програмування та використання захисних методів. Також синтаксична гнучкість, бо така перевага

як легкість вивчення та використання, звичайно є зручною, але може призводити до написання неефективного та вразливого коду.

Другим недоліком можу виділити інтерпретованість, РНР - це мова з високим рівнем інтерпретації, що може впливати на продуктивність у порівнянні з компільованими мовами.

Третім, але не менш важливим є залежність від розробників, тому що, якщо вебсайт розробляється непрофесійним розробником, це може призвести до виникнення непередбачених проблем та безпекових загроз.

2.1.3 Python

Python – дуже часто використовувана, інтерпретована, об'єктно-орієнтована мова програмування високого рівня з динамічною семантикою, яка призначається перш за все для програмування загального призначення. Python є поширенішою мовою розробників та спеціалістів з обробки даних, що робить її невід'ємною частиною сучасних робочих навантажень у галузі науки про дані. Також мова програмування Python широко використовується і при розробці веб-застосунків.

Більшість аналітиків даних використовують мову Python тому, що вона включає в себе багато спеціалізованих бібліотек, таких як SciPy, Matplotlib, та безліч інших, ключовою задачею яких є значно спростити завдання з обробки даних. Переваг у цієї мови теж доволі багато і ключовими я можу виділити декілька.

Перша це простота вивчення та читання коду, синтаксис Python дуже читабельний та зрозумілий, що робить його ідеальним вибором для початківців та розвитку продуктивності розробників. Але не зважаючи на синтаксис та простоту ця мова має широкі можливості, Python підтримує різноманітні стилі програмування, включаючи процедурний, об'єктно-орієнтований та функціональний підходи. Це надає розробникам багато варіантів для розв'язання задач.

Другою перевагою є багата екосистема, це виділяється коли Python має широкий спектр бібліотек та фреймворків для різноманітних областей, таких як наука про дані, штучний інтелект, веб-розробка, аналіз даних та багато інших. Це полегшує розробку та прискорює процес створення програм.

Ну і звичайно не можна не зазначити те що у цієї мови, дуже гарна інтеграція з іншими мовами, Python може взаємодіяти з кодом, написаним на інших мовах програмування, що робить його високоякісним вибором для проектів, які використовують різноманітні технології.

Якщо говорити про недоліки, то Python має відносно повільну швидкість в порівнянні з іншими мовами, такими як C++ чи Java, що робить його менш придатним для великих обчислювальних завдань або додатків з високою продуктивністю[11]. Також глобальний інтерпретатор використовує багато пам'яті, що може стати проблемою для обмежених ресурсів.

Особливістю мови є відсутність строгого контролю типів і це може викликати проблеми під час виконання програми через неправильне використання типів даних. При цьому розробникам інколи важко підтримувати код через недостатність інструментів, які допомагають визначити структуру програми, особливо в великих проектах.

Ще одним недоліком може бути те що, у мови менше підтримки для мобільних і веб-розробок порівняно з іншими мовами програмування, такими як Java або JavaScript, а відсутність вбудованих інструментів для паралельного програмування може затруднити створення ефективних багатозадачних програм.

2.1.4 Go

Go – це мова програмування, також відома як Golang, була розроблена компанією Google. Вона створена з орієнтацією на ефективність та простоту використання. Мова Go використовується для розробки високопродуктивних програм, зокрема веб-серверів, мікросервісів та інших застосунків.

Синтаксис мови Go дещо схожий на синтаксис C, але відрізняється деякими особливостями, такими як автоматичне управління пам'яттю, вбудована підтримка конкурентності та простота використання. Мова Go також відома своєю швидкодією та ефективністю в роботі з багатозадачністю.

Мова програмування Go має численні переваги. Вона володіє високою продуктивністю та ефективністю завдяки компіляції в машинний код. Go також відзначається простотою синтаксису, що полегшує розробку і зрозуміння коду. Інструменти для тестування та документації вбудовані безпосередньо в мову, що сприяє покращенню якості програмного забезпечення. Вбудована підтримка конкурентності робить Go ідеальним вибором для розробки сучасних, високопродуктивних програм. Наявність автоматичного управління пам'яттю спрощує роботу програміста та зменшує ймовірність помилок. Крім того, мова Go добре підтримує паралельне програмування та забезпечує зручність у взаємодії з системними викликами.

Мова Go має кілька недоліків, які можуть впливати на задоволення користувачів та розвиток програм. В першу чергу, відсутність універсального спадку та поліморфізму може створити проблеми для програмістів, звиклих до мов, що підтримують ці концепції. Крім того, відсутність вбудованих засобів роботи зі зв'язками з базами даних може змушувати розробників використовувати сторонні бібліотеки, що не завжди ефективно.

Ще однією недолікою є відсутність засобів управління версіями вихідних кодів та залежностей прямо в мові. Це може ускладнювати роботу команд, які спільно розробляють проекти. Також, мова Go дещо відстає від інших мов щодо екосистеми бібліотек та інструментів, адже вона має менший обсяг сторонніх рішень для різноманітних завдань.

Нарешті, синтаксис Go може виглядати відносно стримано, що може не влаштовувати деяких програмістів, звиклих до більш виразного синтаксису інших мов програмування. Також, обмежений функціонал для обробки виключень (панік та відновлення) може викликати труднощі в роботі з винятками.

Не зважаючи на ці недоліки, Go має свої переваги, такі як ефективність виконання, простота вивчення та використання, а також потужний механізм конкурентної обробки. Кожен вибір мови програмування пов'язаний з компромісами, і важливо обирати мову залежно від конкретних потреб та умов проекту і саме тому в підсумковому варіанті треба оцінювати переваги і недоліки всіх вищезазначених мов.

2.1.5 Ruby

Ruby - це високорівнева, об'єктно-орієнтована мова програмування з простим синтаксисом. Вона підтримує функціональне та імперативне програмування, має динамічну типізацію та автоматичне управління пам'яттю. Ruby використовується для розробки різноманітних програм, включаючи веб-додатки та скрипти.

Перевагою цієї мови можна однозначно виділяти простоту та читабельність, тому що синтаксис Ruby лаконічний і приємний для читання, що полегшує розробку та підтримку коду. Також мова має гнучкість та динамічність, надає велику гнучкість у програмуванні, дозволяючи змінювати типи даних під час виконання. Це полегшує розробку та прискорює вирішення завдань. Додатковою особливістю є й те, що вона об'єктно-орієнтована, все у Ruby - об'єкти. Це полегшує роботу з кодом, а також сприяє його структуруванню та перевикористанню.

Мова Ruby є досить гнучкою та зручною для програмування, проте має свої недоліки. Однією з проблем є швидкодія – Ruby вважається мовою інтерпретації, що може призвести до менш ефективної продуктивності порівняно з компільованими мовами.

Ще однією проблемою є велике споживання пам'яті, що робить Ruby менш ефективним для великих проектів або завдань, які вимагають оптимізації ресурсів.

Також, Ruby не є ідеальним вибором для високопродуктивних завдань або систем реального часу через свою нестабільність щодо часу виконання операцій.

Хоча існують оптимізації та покращення, Ruby не завжди є найшвидшою мовою виконання, що робить її менш придатною для деяких вимогливих завдань. Це може бути однією з причин того що ця мова не завжди є першим вибором серед розробників для реалізації подібного типу завдань та проектів.

2.2 Порівняння мов програмування

Я вважаю, що вибір мови програмування для вебсайту – це важливе і завжди стратегічне рішення, що визначає якість та продуктивність проекту. Розглядаючи різні мови, які були наведені в минулому розділі дуже важливо враховувати їхні переваги та недоліки, тому що це впливатиме на підсумковий результат, особливо, якщо йдеться про серверну частину. Порівняння мов програмування для вебсайтів можна провести, враховуючи кілька ключових аспектів. Однак важливо зазначити, що вибір мови програмування залежить від конкретних потреб проекту та особистих вподобань розробника проекту. Звичайно, важливим є також і те що, порівняння різних мов програмування для розробки вебсайтів може бути доволі складним завданням через різноманітність вимог та сценаріїв використання, але проаналізувавши роботи [12]–[18] було проведено порівняльний аналіз мов програмування, за основними характеристиками та ключовими аспектами який наведений у табл. 2.1.

Таблиця 2.1 – Порівняльна характеристика мов програмування

Критерії порівняння	JavaScript	PHP	Python	Go	Ruby
Ліцензія	GNU	–	Python Software Foundation License	–	MIT
Простота навчання та написання коду	+	–	+	–	–
Кросплатформеність	+	+	+	+	+
Можливість обробляти операції з великим обсягом даних	+	–	+	+	–
Паралельне програмування	+	+	+	+	+

Продовження таблиці 2.1

Критерії порівняння	JavaScript	PHP	Python	Go	Ruby
Наявність бібліотек для машинного навчання	±	±	+	±	+
Інтеграція з іншими мовами	+	+	+	±	±
Наявність зрозумілої та повної документації	+	+	+	+	+
Висока візуалізація даних	+	—	+	—	±
Підвищена продуктивність	+	+	+	+	+
Великий список предметно-орієнтованих пакетів з відкритим вихідним кодом	+	+	+	+	—
Підтримка спільноти	+	+	+	+	—

Як видно з табл. 2.1 найбільші переваги має мова програмування JavaScript, тому що вона є простою в навчанні та використанні, є кросплатформеною, має можливість працювати з великим обсягом даних, гарно інтегрується з іншими мовами, має сильну підтримку спільноти і наявність гарної документації, а також може ефективно працювати у будь-якому середовищі. Крім того, вона підтримує також паралельне програмування, має високий рівень візуалізації даних. JS має багатий набір інструментів, що спрощує виконання завдань з аналізу даних, а також підтримку багатьох фреймворків, що є теж безумовно дуже важливим. Додатково до того ж, у JS є великий список предметно-орієнтованих інструментів з відкритим вихідним кодом. Отже, підбиваючи вищесказане для розробки ПП було обрано мову програмування JavaScript.

2.3 Вибір середовища розробки

Вибір середовища розробки є ключовим етапом у процесі створення програмного продукту. Середовище розробки визначає зручність, продуктивність та ефективність роботи розробника. Правильний вибір сприяє швидшому розвитку проекту та полегшує управління ним.

Наявність потужного інтегрованого середовища або текстового редактора з розширеними функціональностями може значно полегшити написання коду, відлагодження та спільну роботу над проектом в команді[19]. Додаткові інструменти, такі як системи контролю версій, допомагають зберігати історію змін та сприяють колективній роботі.

Важливо також враховувати підтримку вибраного середовища мов програмування, оскільки це може вплинути на доступність бібліотек, фреймворків та інших ресурсів, необхідних для ефективної розробки.

Враховуючи різні вимоги та уподобання розробників, важливість вибору середовища розробки полягає у забезпеченні оптимальних умов для творчої та продуктивної діяльності, впливає на якість та швидкість розробки програмного продукту.

Найчастіше використовувані середовища розробки для створення ПП є Sublime Text, PyCharm, IntelliJ IDEA, Visual Studio Code.

2.3.1 PyCharm

PyCharm - це інтегроване середовище розробки (IDE) для мови програмування Python, яке може бути використане для створення вебсайтів для аналізу даних та підвищення ефективності процесу онлайн-торгівлі. Забезпечує потужні інструменти для розробки веб-додатків на фреймворках Django та Flask, підтримує віртуальні оточення, надає інтерактивний відлагоджувач, інтегрується з системами контролю версій, а також має інструменти аналізу та автоматичного виправлення коду. Додатково, забезпечує можливість використання розширень та плагінів для оптимізації та полегшення роботи з різними фреймворками і бібліотеками. Загалом, сервіс PyCharm завжди додатково визначається своєю потужністю, зручним інтерфейсом та широким функціоналом для підтримки веб-розробки з фокусом на аналіз даних та ефективність онлайн-торгівлі і якнайкраще стане в нагоді веброзробнику.

2.3.2 Sublime Text

Sublime Text - це текстовий редактор, який широко використовується для створення вебсайтів з аналізом даних та підвищення ефективності онлайн-торгівлі. Завдяки своїй швидкості та простоті використання, він є зручним інструментом для розробників. Sublime Text дозволяє легко написати та відредагувати код, забезпечуючи при цьому розширені можливості відлагодження. Його широкий вибір плагінів і розширень дозволяє розробникам адаптувати середовище до власних потреб та використовувати різноманітні інструменти для аналізу даних та оптимізації процесів онлайн-торгівлі. Sublime Text ставить на передній план простоту та продуктивність, роблячи його популярним вибором для розробників, що працюють над веб-проектами з акцентом на аналіз даних та вдосконалення ефективності онлайн-торгівлі.

Він забезпечує інтуїтивний інтерфейс, який спрощує роботу розробників у написанні та відлагодженні коду[20]. Велика кількість доступних тем і плагінів дозволяє налаштувати середовище розробки згідно зі специфічними потребами проекту.

Однією з ключових особливостей Sublime Text є його швидкість реакції та можливість миттєво виконувати операції на великих обсягах коду. Це сприяє покращенню продуктивності та зручності роботи розробників, особливо в умовах великих та складних проектів.

Sublime Text також підтримує роботу з різними мовами програмування, що робить його універсальним інструментом для веб-розробки. Його переваги включають швидкість та легкість використання, що робить його привабливим для швидкого написання та відлагодження коду. Sublime Text також має велику кількість розширень і плагінів, що полегшує розширення його функціональності для конкретних потреб.

Недоліками Sublime Text можуть бути менша функціональність порівняно з іншими інтегрованими середовищами розробки, особливо для великих та складних проектів. Відсутність вбудованих інструментів для

відлагодження коду та аналізу проекту може бути обмеженням у випадках, коли потрібно глибоко вдаватися в розробку та оптимізацію.

2.3.3 IntelliJ IDEA

IntelliJ IDEA - інтегроване середовище розробки (IDE) для мов програмування, зокрема для Java, яке також може бути використане для створення вебсайтів для аналізу даних та підвищення ефективності процесу онлайн-торгівлі. Воно надає зручні інструменти для розробки веб-додатків, забезпечуючи підтримку різних фреймворків та технологій. IntelliJ IDEA інтегрується з системами контролю версій, включає вбудовані інструменти відлагодження коду та аналізу його якості.

IntelliJ IDEA є потужним інтегрованим середовищем розробки, призначеним для створення вебсайтів з аналізом даних та оптимізацією процесу онлайн-торгівлі. Визначаючись широким набором функціональностей, IntelliJ IDEA забезпечує зручний інтерфейс та продуктивне середовище для розробників[21]. Серед його переваг - підтримка різних мов програмування, включаючи Java, Kotlin, і інші, що робить його універсальним для веб-розробки. Інтелектуальний відлагоджувач, підтримка систем контролю версій та автоматичне виявлення помилок полегшують процес розробки. IntelliJ IDEA можна обирати для різноманітних задач, що підтверджують також і розробки в дописах.

Однак, деякі розробники можуть вважати IntelliJ IDEA менш легким у порівнянні з іншими середовищами розробки. Додатково, його більш високий рівень функціональності може вимагати вивчення для повного використання його потенціалу, що може бути витратно в часі для новачків. Однак, з точки зору функціональності та продуктивності, IntelliJ IDEA залишається однією з популярних виборів серед розробників для роботи над веб-проектами, які вимагають аналізу даних та оптимізації процесу онлайн-торгівлі.

2.3.4 Visual Studio Code

Visual Studio Code - це інтегроване середовище розробки (IDE), призначене для створення вебсайтів з аналізом даних та підвищення ефективності процесу онлайн-торгівлі. Його використання спрощує розробку завдяки потужним інструментам, включаючи підтримку різних мов програмування та фреймворків. Visual Studio Code забезпечує зручний відлагоджувач, інтеграцію з системами контролю версій, а також широкий вибір розширень для розширення функціональності та полегшення роботи з різними технологіями. Visual Studio Code, текстовий редактор від Microsoft, є зручним інструментом для розробки вебсайтів з аналізом даних та підвищення ефективності процесу онлайн-торгівлі. Його переваги включають інтуїтивний інтерфейс, велику кількість розширень та плагінів, що полегшують роботу з різними мовами програмування та фреймворками. Visual Studio Code забезпечує інтеграцію з системами контролю версій, віртуальними оточеннями та іншими інструментами, що сприяє легкості ведення проектів.

Наслідком його популярності є активна спільнота, що забезпечує постійне оновлення та підтримку. Однак, Visual Studio Code, будучи текстовим редактором, може бути менш потужним порівняно з інтегрованими середовищами розробки, що мають більше розширених функцій та інструментів для роботи з великими проектами. Також, його ефективність може залежати від конкретного контексту веб-розробки, але якщо розглядати цей інструмент без відриву від контексту він є досить зручним.

2.4 Порівняння середовищ розробки

Порівняння різних середовищ розробки є ключовим аспектом вибору оптимального інструменту для створення програмних продуктів.

Проаналізувавши роботи [22]–[27] було проведено порівняльних аналіз середовищ розробки, який наведений у табл. 2.2.

Таблиця 2.2 – Порівняльна характеристика середовищ розробки

Критерії порівняння	Sublime Text	PyCharm	IntelliJ Idea	Visual Studio Code
Ліцензія	+	+	—	+
Кросплатформність	+	+	+	—
Автозавершення коду	+	+	+	+
Підтримка репозиторіїв	+	+	+	+
Простота встановлення	+	—	+	—
Перевірка та аналіз коду	+	+	+	+
Швидкий та безпечний рефакторинг коду	+	+	+	+
Наявність інтерактивної консолі	+	+	+	+
Наявність зручного користувацького інтерфейсу	+	+	+	+
Швидкий та зручний пошук	+	+	+	+
Підсвічування синтаксису	+	+	+	+
Підтримка наукових пакетів	+	+	+	+
Наявність функції статичного аналізу коду	+	+	+	+
Потребує малої кількості пам'яті для зберігання.	+	—	—	—

З табл. 2.2 можна побачити, що найбільші переваги має середовище розробки Sublime Text, тому що, воно ліцензоване, це є дуже важливим для розробника, також воно має такі переваги як простота в освоєнні, кросплатформне з відкритим вихідним кодом, яке було спеціально створене для аналізу даних і можливість рефакторингу коду. Sublime Text має можливість перевірки та аналізу коду, а також автозавершення коду. Окремо хочу зазначити що Sublime Text також містить важливі функції, як підсвітка коду, автозавершення та за можливості кожен розробник може встановити необхідні додатки для того щоб зробити середовище зручним саме для себе і працювати з максимальним комфортом. Також цей редактор потребує не багато пам'яті для зберігання і має підтримку користувацького інтерфейсу. Отже, підбиваючи вищезазначене для розробки ПП мною було обрано середовище розробки Sublime Text.

2.5 Огляд мініфікаторів коду

Мініфікатори коду - це інструменти, які використовуються для автоматичного зменшення обсягу та об'єму файлів із вихідним кодом, зазвичай JavaScript, CSS, а іноді і HTML. Головною метою мініфікації є оптимізація ресурсів, які передаються браузером під час завантаження веб-сторінок.

2.5.1 UglifyJS

UglifyJS - це інструмент для зменшення розміру та оптимізації коду JavaScript, що може бути використаний при створенні вебсайтів для аналізу даних та підвищення ефективності процесу онлайн-торгівлі. У своїй суті, UglifyJS допомагає зменшити обсяг передачі даних та прискорює завантаження веб-сторінок, що особливо важливо для оптимізації продуктивності онлайн-торгівлі і дуже цікаво для нас з огляду на тему роботи. Він має кілька переваг, таких як зменшення обсягу файлів, що передаються клієнту, що покращує час завантаження веб-сторінок. UglifyJS допомагає знижувати бандл JavaScript до мінімального розміру, що є важливим для оптимізації продуктивності веб-додатків та забезпечення кращого досвіду користувача.

Недоліками UglifyJS можуть бути певні обмеження у зручності читання мінімізованого коду, оскільки він часто стає важчим для розуміння та відладки. Важливо враховувати, що мінімізація коду може впливати на видимість та читабельність, що може бути важливим фактором при розробці вебсайтів для аналізу даних та підвищення ефективності онлайн-торгівлі.

2.5.2 Terser

Terser є інструментом, призначеним для створення вебсайтів з аналізом даних та підвищення ефективності процесу онлайн-торгівлі. Він відзначається

тим, що дозволяє здійснювати оптимізацію та мініфікацію JavaScript-коду, що покращує швидкість завантаження веб-сторінок. Використовуючи Terser, розробники можуть забезпечити оптимальну продуктивність та ефективність вебсайту, зокрема в контексті обробки та аналізу великих обсягів даних, а також оптимізації процесів онлайн-торгівлі. Використання Terser має свої переваги та недоліки в контексті створення вебсайтів для аналізу даних та підвищення ефективності процесу онлайн-торгівлі.

Переваги Terser включають в себе його здатність значно зменшити розмір JavaScript-коду, що призводить до швидшого завантаження веб-сторінок. Це особливо важливо для вебсайтів, які фокусуються на аналізі великих обсягів даних та оптимізації онлайн-торгівлі, де швидкість завантаження може впливати на користувацький досвід.

Однак, серед недоліків Terser слід відзначити можливі проблеми з читабельністю коду, оскільки мінімізація може робити код менш зрозумілим для розробників під час відлагодження. Також, варто зазначити і такий доволі важливий і красномовний факт, про те що невірна конфігурація Terser може викликати, звичайно ж проблеми з функціональністю додатка, що може бути критичним для вебсайтів, орієнтованих на аналіз даних та ефективність онлайн-торгівлі.

2.5.3 CSSnano

CSSnano - це інструмент для стиснення та оптимізації CSS-коду, який може бути використаний при створенні вебсайтів для аналізу даних та підвищення ефективності процесу онлайн-торгівлі. Його основна мета полягає в зменшенні розміру CSS-файлів, що допомагає прискорити завантаження веб-сторінок та зменшити трафік, що особливо важливо в умовах онлайн-торгівлі. cssnano автоматично виконує оптимізації, такі як прибирання зайвих пропусків та пробілів, мінімізація значень властивостей та інші оптимізації, що сприяють покращенню продуктивності та швидкості завантаження вебсайту. CSSnano є

інструментом для оптимізації та мінімізації CSS-коду, що може бути використаний при створенні вебсайтів для аналізу даних та підвищення ефективності процесу онлайн-торгівлі. Він дозволяє зменшити розмір CSS-файлів, що покращує швидкість завантаження веб-сторінок та сприяє загальній продуктивності сайту.

Переваги CSSnano включають в себе ефективну мінімізацію коду, здатність видалення зайвих пробілів та непотрібних символів, що дозволяє знизити обсяг передаваних даних і покращити час завантаження сторінки. Також, CSSnano може здійснювати оптимізації, такі як стиснення значень та видалення непотрібних префіксів для властостей CSS.

Однак, важливо враховувати, що недоліками CSSnano може бути те, що в деяких випадках він може викликати проблеми з читабельністю коду, особливо для розробників, які працюють безпосередньо з мінімізованим кодом. Також, під час використання CSSnano слід бути обережним, щоб впевнитися, що він не видаляє необхідні стилі.

2.6 Структурна схема програмного продукту

Розроблюваний програмний продукт призначений для підвищення ефективності онлайн-торгівлі на прикладі вебсайту. Структурна схема розроблюваного вебсайту для підвищення ефективності онлайн-торгівлі включає фронтенд і бекенд. На фронтенді передбачено головну сторінку з категоріями товарів, каталогом, сторінками товарів, кошиком та формою оформлення замовлення. Бекенд включає модулі обробки залишених відгуків користувачем, а також модуль оплати та доставки.

На рис. 2.1 представлена діаграма класів програмного продукту.

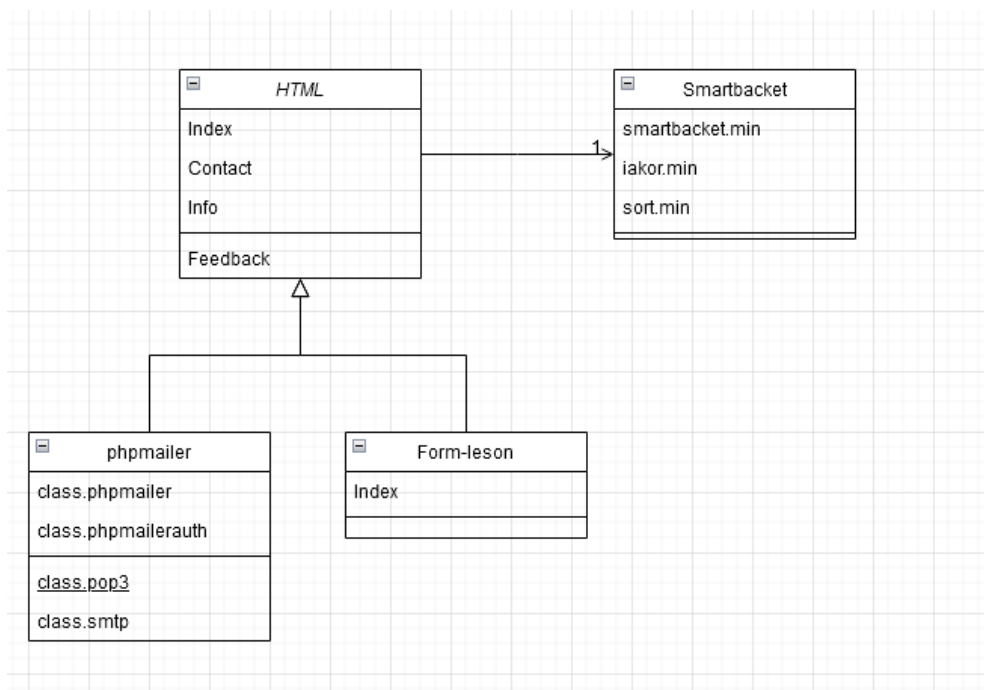


Рисунок 2.1 – Діаграма класів програмного продукту

Модуль HTML– головний модуль проєкту, в якому буде відбуватися відображення програмного продукту та елементів інтерфейсу користувача, він має містити такі методи:

- index– призначений для виводу інтерфейсу програмного продукту;
- contact– призначений для можливості заповнити форму та зв’язатись з розробником;
- info– призначений для завантаження інформації;
- feedback– призначений для обробки відгуків по часу, імені та найменуванню товару.

Модуль phpmailer– призначений для попередньої обробки даних, він зберігає дані та надсилає їх в подальшому на фронтенд.

Модуль smartbasket– призначений для реалізації можливості замовлення товару, підрахунку кількості, суми і іншої інформації про товар, а також валідності введених даних користувачем по деяким полям, він має містити такі методи:

- iakor.min– призначений для прив’язки товару по заданому раніше номеру;

- smartbasket.min– призначений для візуалізації оброблених даних кошику і передачу їх на фронтенд в прийнятному для користувача вигляді з урахуванням виконаних операцій;
- sort.min– призначений для сортування товару за певними категоріями, як то ціна товару(по зростанню, по спаданню), також по найвідвідуванішим товарам.

2.7 Висновок

У цьому розділі виконано аналіз та вибір інструментарію для розробки програмного продукту (ПП) і спроектовано його архітектуру. Порівняльний аналіз мов програмування для розробки ПП показав, що JavaScript є найбільш вигідним вибором, завдяки легкій адаптабельності, розгалуженому набору бібліотек і інструментів, що полегшують аналіз даних, та найкраще підходить для вебсайтів.

Також був проведений порівняльний аналіз середовищ розробки, і серед них Sublime Text визнано найбільш оптимальним завдяки його простоті в освоєнні, кросплатформенності та відкритому вихідному коду, а також наявності широкого набору бібліотек для науки про дані.

Додатково були описані інструменти, що допоможуть в аналізі даних а також мініфікатори коду, що допоможуть оптимізувати вебсайт. Структурна схема розроблюваного продукту також була детально описана.

3 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ

Завданням роботи є програмна реалізація вебсайту а також дослідження методів методів підвищення ефективності процесу онлайн-торгівлі.

Для досягнення поставленої мети необхідно розробити адаптивний вебсайт призначений для продажу комп'ютерної техніки для дослідження на основі його методів підвищення ефективності процесу онлайн-торгівлі.

3.1 Попередня обробка даних

Процес онлайн-торгівлі може бути оптимізований через різноманітні методи, включаючи попередню обробку даних. Попередня обробка даних - це етап, на якому дані підготовлюються та аналізуються перед тим, як вони використовуються для прийняття рішень або виведення результатів.

Сегментація клієнтів - це стратегічний підхід до розділення аудиторії на групи з подібними характеристиками та поведінкою для забезпечення більш ефективного спілкування та персоналізованого обслуговування. Для реалізації цього процесу використовуються різні методи та інструменти.

Одним із методів є аналіз даних про клієнтів, включаючи їхні покупки, перегляди товарів, часові рамки активності та інші параметри. Цей аналіз може виконуватися за допомогою аналітичних платформ, таких як Google Analytics або спеціалізовані CRM-системи.

Важливим інструментом в сегментації є CRM (Customer Relationship Management), який дозволяє збирати та управляти інформацією про клієнтів. Це дозволяє створювати детальні профілі клієнтів та визначати їхні потреби.

Також використовуються алгоритми машинного навчання для визначення схожості між клієнтами та прогнозування їхнього майбутнього поведінки. Це дозволяє автоматизувати процес сегментації та піддавати аудиторію більш точному аналізу.

Також можна використовувати інструменти для вивчення відгуків та поведінки в соціальних мережах, щоб отримати інсайти щодо думок та уподобань клієнтів.

Для розробки CRM (Customer Relationship Management) для сайту продажу комп'ютерної техніки, спочатку ми будемо зосереджуватися на зборі та аналізі різноманітної інформації про клієнтів. Це включатиме збір базових особистих даних, таких як ім'я, контактна інформація, а також історію покупок та переглядів на сайті. Наведення звітності та аналіз даних з CRM допоможе в намірі розуміти ефективність стратегій продажу та виявляти можливості для вдосконалення обслуговування клієнтів.

Ми будемо використовувати аналітичні інструменти для відстеження поведінки користувачів на сайті, а також для визначення попиту на конкретні товари та категорії. Це дозволить нам створювати детальні профілі клієнтів та розуміти їхні потреби та уподобання.

Запровадження системи оцінки клієнтів допоможе визначити їхню цінність для бізнесу та присвоїти різні рівні обслуговування в залежності від їхнього внеску. Таким чином, ми зможемо надавати персоналізовані пропозиції та знижки, спрямовані на збільшення лояльності клієнтів.

CRM для сайту продажу комп'ютерної техніки також буде включати інтеграцію з системами оплати, щоб забезпечити зручну та безпечну процедуру оформлення замовлення. Збереження інформації про оплату та доставку у профілях клієнтів дозволить швидко та ефективно обробляти замовлення і в майбутньому додатково реалізувати цю безперечно зручну та цікаву функцію на сторінку.

Наведення звітності та аналіз даних з CRM допоможе в намірі розуміти ефективність стратегій продажу та виявляти можливості для вдосконалення обслуговування клієнтів. Цей цикл аналізу та оптимізації сприятиме постійному покращенню взаємодії з клієнтами та ефективному управлінню відносинами з ними.

3.2 Оптимізація конверсії

Оптимізація конверсії (CRO) для сайту продажу комп'ютерної техніки передбачає впровадження різноманітних стратегій та підходів для підвищення відсотка відвідувачів, які здійснюють покупку. Правильний підхід до CRO включає кілька ключових етапів.

По-перше, проведено глибокий аналіз потенційних аналітичних даних заснованих на статистиці подібного типу ресурсів, щоб зрозуміти поведінку користувачів на сайті. Це включає в себе вивчення шляхів навігації, часу перебування на сторінках, точок виходу та інших метрик. На основі цієї інформації визначено слабкі місця та можливості для поліпшення.

Далі, ми вдосконалено користувацький інтерфейс сайту, забезпечуючи легкість навігації та інтуїтивність. Покращено описи товарів, включені відгуки клієнтів та визначені вигідні пропозиції. Дизайн сторінок оптимізований для швидкості завантаження та адаптивності для різних пристроїв.

Застосовано стратегії A/B-тестування, щоб порівнювати різні версії сторінок та елементів дизайну з метою визначення оптимальних варіантів, які сприяють збільшенню конверсії.

Спрощено процес оформлення замовлення та оплати, мінімізовано кількість кроків для завершення транзакції. Забезпечено чітку інформацію щодо умов доставки та повернення товарів, щоб підвищити вірогідність завершення покупки.

Паралельно з цим, розвинено стратегії персоналізації, використовуючи дані про попередні покупки та перегляди товарів, щоб надавати користувачам індивідуалізовані пропозиції та рекомендації.

Запроваджено ефективні виклики до дії (CTA), використовуючи привабливі та переконливі виклики, щоб спонукати відвідувачів до виконання бажаних дій, таких як оформлення замовлення або підписки на розсилку.

Усі ці заходи мають на меті створити оптимальні умови для відвідувачів сайту, сприяти їхній довірі та позитивному досвіду, що в кінцевому підсумку

приведе до підвищення конверсії та збільшення обсягів продажів комп'ютерної техніки.

3.3 Висновок

У цьому розділі було проаналізовано методи підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту, як метод сегментації та оптимізації конверсії. Принципова відмінність запропонованих методів сегментації та оптимізації конверсії в онлайн-торгівлі полягає в їхній комбінації та синергії, спрямованій на комплексне покращення результатів проекту. Обидва методи розглядаються як частини стратегічного підходу до підвищення ефективності в електронній комерції

4 КЕРІВНИЦТВО ПРОГРАМІСТА

4.1 Призначення й умови використання

Розроблюваний програмний продукт призначений для моделювання продажу комп'ютерної техніки і дослідження на основі програмного продукту можливостей підвищення ефективності процесу онлайн-торгівлі. Він являє собою набір файлів за допомогою яких можна переглядати, змінювати та додавати нову інформацію на сторінку, а також змінювати налаштування сортування. Програмний продукт представлений у вигляді вебсайту завдяки якому можна подивитись асортимент товарів, оформити замовлення, залишити відгук та відсортувати товари і на основі вебсайту дослідити можливості підвищення ефективності процесу онлайн-торгівлі та скласти прогноз кількості відвідувачів та замовлених товарів.

Для коректної роботи ПП необхідні такі мінімальні системні вимоги:

- платформа x64-based;
- операційна система Microsoft Windows 10/8/7 (рекомендується встановити всі останні оновлення);
- процесор з мінімальною тактовою частотою 1,8 ГГц. Рекомендується використовувати як мінімум двоядерний процесор;
- оперативна пам'ять від 2 ГБ (рекомендується 4 ГБ)
- мінімум 2 ГБ вільного місця на жорсткому диску.

У своїй роботі ПП використовує такі необхідні компоненти:

- середовище Sublime Text;
- браузер (будь-який, але рекомендується слідкувати за оновленнями)

4.2 Звертання до програмного продукту (файли проєкту)

Розроблюваний ПП складається з таких модулів:

Модуль HTML– головний модуль проєкту, в якому буде відбуватися відображення програмного продукту та елементів інтерфейсу користувача, він

має містити такі методи:

- index– призначений для виводу інтерфейсу програмного продукту;
- contact– призначений для можливості заповнити форму та зв'язатись з розробником;
- info– призначений для завантаження інформації;
- feedback– призначений для обробки відгуків по часу, імені та найменуванню товару.

Модуль `phpmailer`– призначений для попередньої обробки даних, він зберігає дані та надсилає їх в подальшому на фронтенд.

Модуль `smartbasket`– призначений для реалізації можливості замовлення товару, підрахунку кількості, суми і іншої інформації про товар, а також валідності введених даних користувачем по деяким полям, він має містити такі методи:

- `iaakor.min`– призначений для прив'язки товару по заданому раніше номеру;
- `smartbasket.min`– призначений для візуалізації оброблених даних кошику і передачу їх на фронтенд в прийнятному для користувача вигляді з урахуванням виконаних операцій;
- `sort.min`– призначений для сортування товару за певними категоріями, як то ціна товару(по зростанню, по спаданню), також по найвідвідуванішим товарам.

4.3 Вхідні і вихідні дані

Модуль `HTML` призначений для відображення інтерфейсу програмного продукту та включає методи `index`, `contact`, `info`, `feedback`. Вхідні дані - це запитання або дії користувача, які викликають відповідні методи, а вихідні - це відображення програмного інтерфейсу, форми для зв'язку, завантаженої інформації та оброблених відгуків.

Модуль `phpmailer` служить для попередньої обробки даних, їх збереження та відправлення на фронтенд. Вхідні дані - це дані, які користувач вводить через форму, а вихідні - це оброблені дані, готові для відправлення або збереження.

Вебсайт і телеграм-бот взаємодіють через API Телеграму для обміну даними. Щоб це реалізувати, мною було створено телеграм-бота за допомогою `@BotFather` та отримано токен для доступу до API. Для зручності і розширення функціональності використано бібліотеку `python-telegram-bot` для взаємодії з телеграм API. Токен бота включений в HTTP-запити для аутентифікації під час взаємодії з API телеграму. Таким чином вебсайт отримав можливість відправляти повідомлення боту за допомогою API-запитів. Програмна структура ПП представлена на рис. 4.1.

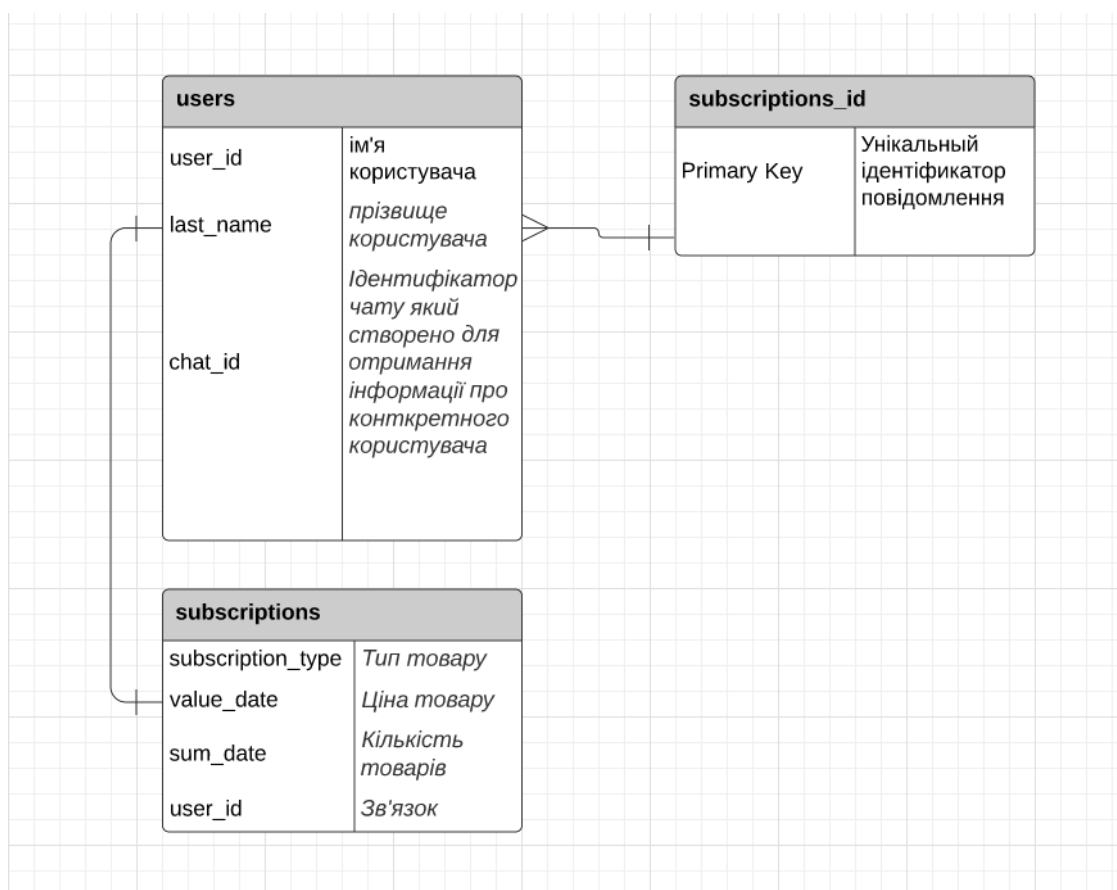


Рисунок 5.1 – Програмна структура ПП

Модуль `smartbasket` реалізує можливість замовлення товару, підрахунку інформації про товар та валідності введених даних. Вхідні дані - це дії

користувача, що викликають методи `iakor.min`, `smartbasket.min`, `sort.min`, а вихідні - це прив'язаний товар, візуалізовані дані кошику та відсортований товар відповідно.

Модуль "smartbasket" реалізує набір функціональностей, пов'язаних із замовленням товарів, обчисленням інформації про товар та перевіркою правильності введених даних. Вхідні дані цього модуля представлені діями користувача, які викликають методи "iakor.min", "smartbasket.min" і "sort.min". Результатом роботи модуля є зв'язаний товар, візуалізована інформація про кошик та товар, впорядкований відповідно до вказаних параметрів. Модуль "smartbasket" призначений для реалізації різноманітних функцій, пов'язаних з управлінням та обробкою інформації про товари та замовлення в інтернет-магазині або системі електронної комерції

4.4 Повідомлення

Під час своєї роботи створений ПП виводить деякі текстові повідомлення, які описані у табл. 4.1.

Таблиця 4.1 – Повідомлення під час роботи ПП

Повідомлення	Опис
Вітаємо на сайті!	Повідомлення означає, що програмний продукт готовий до роботи.
Будь ласка, заповніть порожнє поле.	Повідомлення появляється, коли поле порожнє і це означає, що потрібно обов'язково заповнити це поле для виконання прогнозування.

Продовження таблиці 4.1

Повідомлення	Опис
Будь ласка, виберіть товар	Повідомлення з'являється, коли користувач не обрав товар який хоче додати до кошика .
Помилка в завантаженні даних.	Повідомлення означає, що дані отримані від користувача не є коректними.
Будь ласка, введіть електронну пошту, ім'я.	Повідомлення з'являється, коли користувач ввів не валідне значення поля і це означає, що потрібно знову заповнити поле

4.5 Висновок

У цьому розділі було наведено призначення та функції розробленого ПП та мінімальні системні вимоги для його коректної роботи. Було наведено детальний опис модулів з яких складається ПП. Розроблений ПП– це інтерактивна платформа, яка надає користувачам можливість взаємодії з різноманітними функціями та отримання необхідної інформації. ПП складається з кількох ключових модулів, які спрямовані на зручність користувачів та оптимізацію обробки даних. Додатково розглядаю можливість в майбутньому для розширення функціональності реалізувати механізм на вебсайті, використовуючи веб-хуки або періодичну перевірку. використання власних команд для бота, щоб реагувати на конкретні запити.

5 КЕРІВНИЦТВО ОПЕРАТОРА

5.1 Експлуатація та тестування

Розроблюваний програмний продукт призначений для моделювання продажу комп'ютерної техніки і дослідження на основі програмного продукту можливостей підвищення ефективності процесу онлайн-торгівлі. Він являє собою набір файлів за допомогою яких можна переглядати, змінювати та додавати нову інформацію на сторінку, а також змінювати налаштування сортування. З використанням вебсайту також можна проводити дослідження з метою визначення можливостей підвищення ефективності процесу онлайн-торгівлі. Програмний продукт надає засоби для складання прогнозів кількості відвідувачів та замовлених товарів на основі зібраної інформації. Програмний продукт представлений у вигляді вебсайту завдяки якому можна подивитись асортимент товарів, оформити замовлення, залишити відгук та відсортувати товари і на основі вебсайту дослідити можливості підвищення ефективності процесу онлайн-торгівлі та скласти прогноз кількості відвідувачів та замовлених товарів.

Для коректної роботи ПП необхідні такі мінімальні системні вимоги:

- платформа x64-based;
- операційна система Microsoft Windows 10/8/7 (рекомендується встановити всі останні оновлення);
- процесор з мінімальною тактовою частотою 1,8 ГГц. Рекомендується використовувати як мінімум двоядерний процесор;
- оперативна пам'ять від 2 ГБ (рекомендується 4 ГБ)
- мінімум 2 ГБ вільного місця на жорсткому диску.

У своїй роботі ПП використовує такі необхідні компоненти:

- середовище Sublime Text;
- браузер (будь-який, але рекомендується слідкувати за оновленнями).

5.2 Виконання програмного продукту

Після вдалого запуску ПП відкривається головна сторінка, яка представлена на рис. 5.1, на якій буде відображено товари і кожен з них готовий до більш детального перегляду.

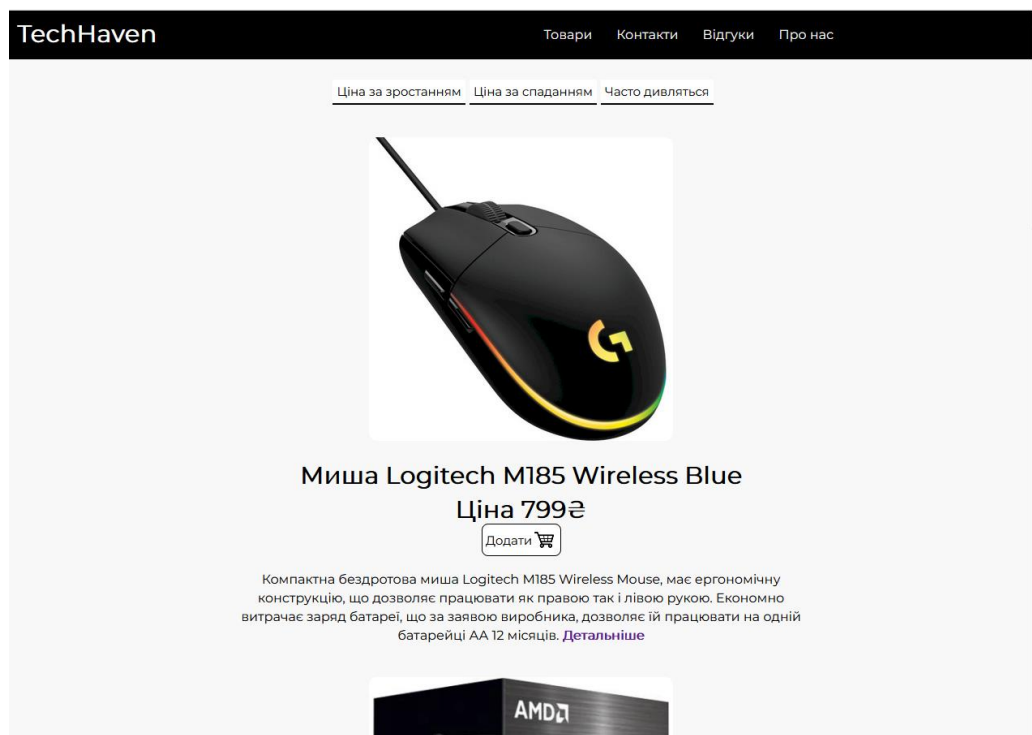


Рисунок 5.1 – Головне вікно ПП

Для того, щоб здійснити замовлення товару, спочатку необхідно обрати один або декілька товарів із запропонованих, можливо, за потреби перед цим переглянувши більш детальні характеристики товарів клікнувши на кожен з них. Потім необхідно натиснути на кнопку Додати і в такому випадку користувач побачить на екрані інтерактивне спливаюче вікно в якому буде можливість обрати дві опції, а саме – перейти до кошика, або продовжувати покупки. При цьому інтерактивне спливаюче вікно буде автоматично підлаштовувати інформацію для клієнта, якщо він вже раніше обрав цей клас товарів, вікно покаже на екрані повідомлення що даний товар вже був раніше обраний, якщо ж користувач обрав новий товар то повідомлення на екрані буде містити інформацію, що товар додано до кошика. Доволі важливо було реалізувати це

саме таким чином для того щоб користувач, якщо він володіє низьким рівнем навичок не заплутався в ПП. Приклад виведення результатів додавання товару до кошика в наведеному ПП приведено на рис. 5.2.



Рисунок 5.2 – Вікно виводу додавання товару до кошика

Якщо користувач натиснув кнопку перейти до кошика, щоб здійснити покупку користувач переходить до наступної сторінки. На цій сторінці відбувається процес залишення замовлення на купівлю товару. В вікні відображається ID товару який замовив покупець, його ціна та кількість одиниць товару, при чому користувач може змінювати кількість безпосередньо в кошику і ціна буде за нею підлаштовуватись.

Також є кілька полів з інформацією, яку користувач повинен ввести для того щоб замовити товар – ім'я, пошту та номер телефону. Поля імені та номеру телефону є обов'язковими, бо заявка на товар буде надсилатись до месенджера телеграм-бота і для подальшого зв'язку з покупцем адміністратору необхідно мати такі дані. Також покупець повинен прийняти умови користувацької угоди і надалі або продовжити покупки або залишити заявку на замовлення товару. Приклад реалізованої можливості замовлення товару в рамках ПП приведено на рис. 5.3

Товар	ID	Ціна / Грн	Кількість метрів	Загальна ціна / Грн	Видалити
 Acer Aspire 7	79	30999	- 1 +	30999	×
 TP-Link Archer C64	83	1399	- 2 +	2798	×
 AMD Ryzen 5	97	6239	- 1 +	6239	×
 Logitech M185 Wireless Blue	31431	799	- 6 +	4794	×

Загальна вартість: 8 Загальна вартість: 44830.00 Грн

☒ Я приймаю умови користувацької угоди

Продовжити покупки
Зробити замовлення

Рисунок 5.3 – Вікно замовлення товару

Для того щоб користувачу було зручніше обирати товар в ППP реалізована можливість сортування товарів на сторінці. Сортування можливе по трьом параметрам – по зростанню ціні, спаданню та найбільш популярним товарам. Приклад реалізованої можливості сортування товарів в рамках ППP приведено на рис. 5.4

Ціна за зростанням
Ціна за спаданням
Часто дивляться

Рисунок 5.4 – Вікно сортування товарів

Після кожного товару в ППP реалізована можливість залишення відгуку на сторінці, для того щоб кожен користувач міг поділитись власним досвідом по роботі з товаром і допомогти іншим користувачам обрати потрібний саме їм товар по певним характеристикам. Для того щоб залишити відгук потрібно заповнити поле імені та безпосередньо відгук, а час вираховується автоматично.

Приклад залишення відгуку в рамках ПП приведено на рис. 5.5 – 5.6

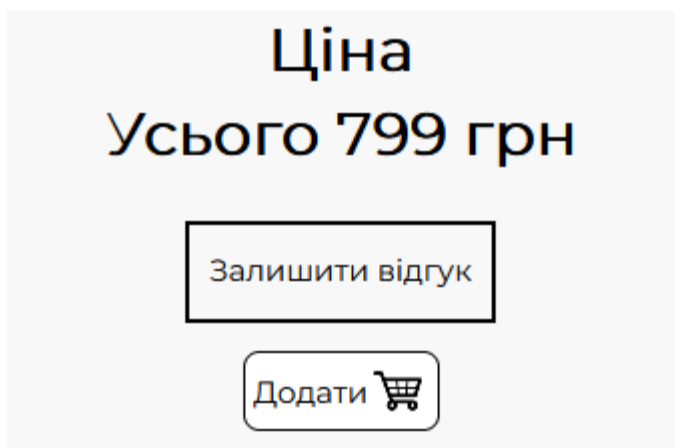


Рисунок 5.5 – Вікно залишення відгуку

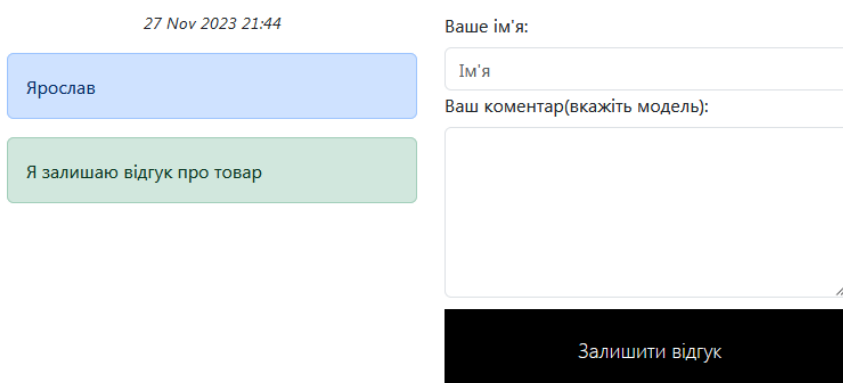


Рисунок 5.6 – Вікно залишення відгуку

Якщо користувач виявить бажання забрати замовлений товар у продавця самостійно, не чекаючи доставки, в ПП реалізовано можливість подивитись де знаходиться склад на мапі та прокласти маршрут від свого місцеположення до складу за допомогою наведеної можливості. Приклад реалізації навігації в рамках ПП приведено на рис. 5.7

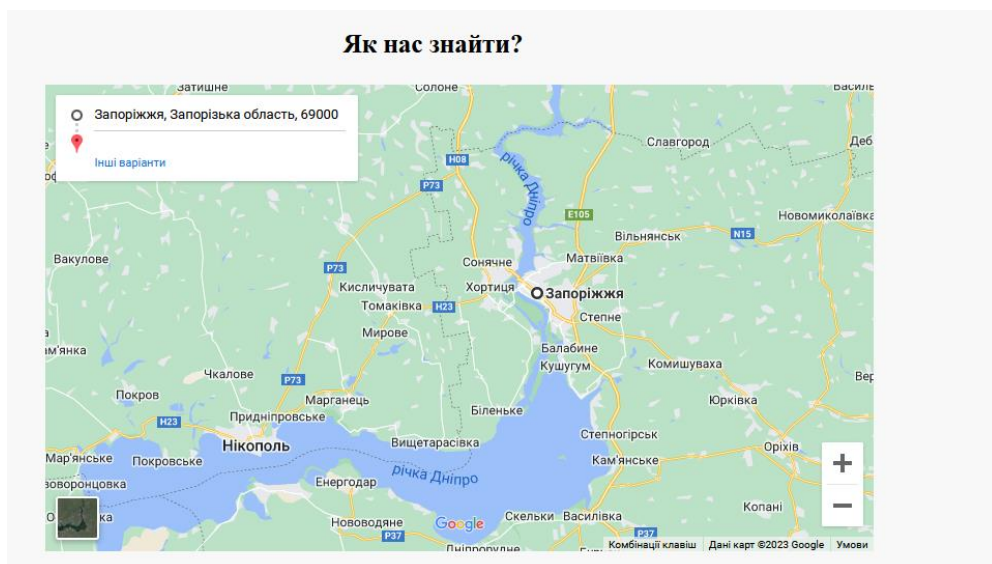


Рисунок 5.7 – Вікно навігації

5.3 Повідомлення оператора

Під час роботи оператора з розробленим ПП виводяться різні текстові повідомлення, вони інформують оператора про помилки або його неправильні дії, а також про закінчення розрахунків прогнозування. Список всіх текстових повідомлень, описання їх змісту та дії оператора, які він повинен виконати при отриманні цих повідомлень представлені у розділі «Керівництво програміста» в таблиці 4.1.

5.4 Висновок

У цьому розділі було наведено призначення та функції розробленого ПП та мінімальні системні вимоги для його коректної роботи. Було описана покрокова інструкція роботи оператора з розробленим ПП, а саме було зазначено як виконувати замовлення товару, вибір, перегляд детальних характеристик та інших функцій. Було перераховано список всіх текстових повідомлень, які можуть з'являтися під час роботи оператора з ПП. Була наведена детальна інструкція по роботі оператора з ПП, а саме приклад замовлення товару та описано кроки і проблемні ситуації, які можуть виникнути при цьому.

Розроблений інтерфейс ПП є інтуїтивно зрозумілий та простим і додаткове навчання оператора не потребується, якщо виникають питання по роботі з ПП присутня довідка.

6 ЕКСПЕРИМЕНТИ ТА РЕЗУЛЬТАТИ

6.1 Проведення експериментів

Для ефективності експерименту на вебсайті важливо врахувати кілька ключових статистичних показників. Конверсія, або відсоток відвідувачів, які виконали бажану дію, може бути визначена як кількість конверсій поділена на загальну кількість відвідувачів, помножена на 100%. Кількість замовлень - це просто кількість покупок, здійснених під час експерименту. Середній чек обчислюється як загальна сума замовлень поділена на кількість замовлень. Результат ключових статистичних показників в рамках ПП приведено на рис. 6.1



Рисунок 6.1 – Ключові статистичні показники

Крім того, важливо визначити контрольну та тестову групи для порівняння результатів. Аналіз статистичної значущості допоможе визначити, чи є різниця між групами статистично значущою. Тривалість експерименту, визначена в днях, тижнях або місяцях, також важлива для правильної інтерпретації результатів.

Нарешті, Retention Rate, або ставка утримання, визначає відсоток користувачів, які залишаються активними на сайті після конверсії. Ці показники надають зрозумілі дані для прийняття обґрунтованих рішень щодо впроваджених змін на вебсайті. Приклад метрики Retention Rate в рамках ПП приведено на рис. 6.2

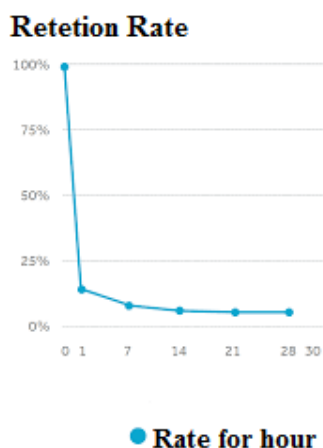


Рисунок 6.2 – Retention Rate

Додатково, статистична значущість є критичною для оцінки експерименту. Використовуючи статистичні тести, такі як t-тест або аналіз дисперсії (ANOVA), можна визначити, чи є різниця між групами статистично значущою. Аналіз дисперсії (ANOVA) - це статистичний метод, який використовується для порівняння середніх значень основних трьох чи більше груп та визначення, чи є різниця між ними статистично значущою. У випадку вебсайту ANOVA може бути застосований для порівняння результатів отриманих раніше метрик між різними версіями веб-сайту або між різними групами користувачів. Це допомагає у визначенні реального впливу змін на сайті та прийнятті обґрунтованих рішень на основі даних, які не є випадковими, а заснованими на аналізі різних груп користувачів. Результат статистичного тесту (ANOVA), згідно раніше зібраних даних в рамках ПП приведено на рис. 6.3

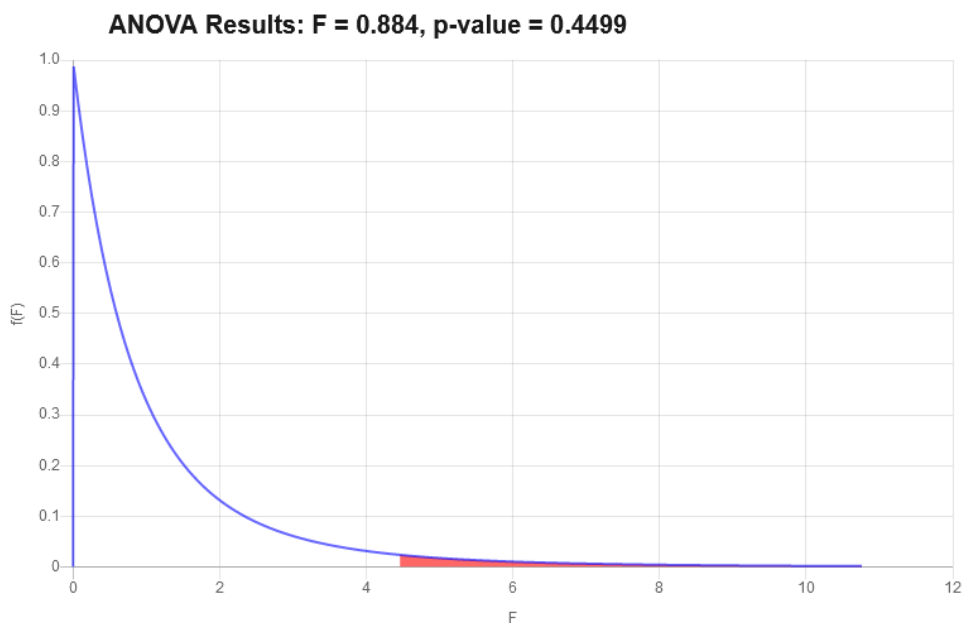


Рисунок 6.3 – Результат статистичного тесту (ANOVA)

Нарешті, тривалість експерименту важлива для правильного аналізу результатів. Визначення часового періоду, протягом якого відбувається експеримент, допомагає зрозуміти стійкість змін та їхній вплив на тривалості користувачів на сайті після здійснення конверсії. На ці фактори дуже важливо зважати при проведенні експерименту для того щоб отримати об'єктивні дані про той чи інший аспект.

ВИСНОВКИ

Під час роботи проведено аналіз предметної області та визначено ключові поняття, пов'язані з розробкою вебсайту для продажу комп'ютерної техніки. Виявлено актуальність та доцільність поставленого завдання, включаючи розгляд питань ефективності, взаємодії з користувачами та оптимізації процесів.

Проведено дослідження та порівняльний аналіз існуючих програмних засобів для веб-розробки та управління електронною комерцією, зокрема тих, які можуть бути використані для створення вебсайту з продажу комп'ютерної техніки.

Вивчено основні методи реалізації функцій вебсайту, таких як замовлення товарів, зв'язок з розробником, завантаження інформації та обробка відгуків користувачів. Розглянуто аспекти валідації введених даних та візуалізації оброблених даних кошику.

Зазначено, що для оптимізації та покращення функціоналу сайту використовуються методи сортування товару за різними категоріями, такими як ціна, найвідвідуваніші товари тощо.

У рамках розроблення вебсайту враховано потреби користувачів у зручному інтерфейсі, можливості легкого замовлення товарів та взаємодії з розробником через форму зворотного зв'язку.

Зазначено, що важливим етапом є виведення статистичних даних, таких як конверсія, кількість замовлень та середній чек, для оцінки ефективності розробленого вебсайту.

У висновках вказано, що виведені результати аналізу становлять підставу для подальшої розробки та вдосконалення вебсайту з метою оптимізації користувацького досвіду та досягнення більшої ефективності у продажах комп'ютерної техніки.

Наукова новизна цього проекту проявляється у новому підході до обробки замовлень у віртуальному середовищі та впровадженні автоматизованої системи отримання та обробки замовлень через миттєве сповіщення. Коли

користувач заповнює форму на вебсайті для придбання комп'ютерної техніки, дані автоматично надходять до спеціального боту. Це забезпечує оперативне реагування на замовлення, забезпечуючи миттєвий доступ до інформації про нові покупки.

Система надає можливість оперативно реагувати на замовлення, надаючи зручний і ефективний обіг інформації. Також система дозволяє операторам не лише виправляти помилки, але й персоналізувати замовлення в реальному часі з урахуванням конкретних потреб клієнта. Оператор, отримуючи повідомлення від системи, негайно взаємодіє з новими замовленнями, виправляє помилки та відповідає на запитання користувачів.

Такий підхід вносить швидкість та ефективність у процес продажу комп'ютерної техніки, забезпечуючи його більш зручний та гнучкий характер. Отримання інформації про замовлення в реальному часі, навіть за межами офісу, робить систему особливо привабливою для підприємств, які цінують оперативність та високий рівень обслуговування клієнтів. Цей інноваційний підхід може сприяти поліпшенню конкурентоспроможності в сучасному віртуальному ринковому середовищі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Operation strategy optimization of desulfurization system based on data mining / [S. Liu, L. Sun, S. Zhu et al.] // Applied Mathematical Modelling. – 2020. – Vol. 81. – P. 144–158.
2. Кожемя'кін А. А. HTML та CSS в прикладах. Створення Web-сторінок / А. А Кожемя'кін. – М. : Альтекс-А, 2014. – 397 с.
3. Макфарланд Д. Велика книга CSS3 : навчальний посібник / Д. Макфарланд. – Київ: Альтекс-А, 2014. – 608 с.
4. Global Web-Technologies clean code a handbook of agile soft/ [Q. Zhong, H. Shen, X. Yun et al.] // Environmental Science & Technology. – 2020. – Vol. 54.– P. 6508–6517.
5. Горнаков С.Г. Вчимо популярні системи управління сайтом/ С.Г. Горнаков. – М. : ДМК Пресс, 2009. – 336 с.
6. An investigation on data mining and operating optimization for wet flue gas desulfurization systems / [Z. Qiao, X. Wang, H. Gu et al.] // Fuel. – 2019.– Vol. 258. – P. 1–10.
7. Евдокімов Н.С. Створення сайтів / Н.С. Евдокимов. – Київ: BookChef, 2020. – 410 с.
8. Simpson Kyle. Wet Programming JavaScript Applications: Robust Web Architecture / Kyle Simpson. – London: Packt Publishing, 2018. – 390 p.
9. Kiil, S. Experimental The Complete Guide to Modern JavaScript. / Kiil, S. – London: Independently, 2021.– 682 p.
10. Miletsky J.I. Principles of Internet Marketing: New Tools and Methods for WebDevelopers : Course Technology Press / J.I. Miletsky. – Boston, MA, United States, 2009.– 480 p.
11. Мейер Є. CSS: повна збірка / Є. Мейер., Є.Уэйл. – Київ: Komubook, 2019. – 1088 с
12. Modeling Study of Selenium Migration Behavior in Wet Flue Gas Desulfurization Spray Towers / [R. Zou, G. Luo, C. Fang et al.] // Environmental

Science & Technology. – 2020. – Vol. 54. – P. 16128–16137.

13. Ethan Marcotte. Responsive website / Marcotte E. – Boston: MA, 2009.– 180 p.
14. Shi D. Philosophy of CSS design / D. Shi. – Galati: University of Galati, 2000. – 312 p.
15. Базові поняття і терміни веб-технологій / [А. В. Кільченко, О. І. Поповський, О- р В. Тебенко, О-й. В. Тебенко, Н. М. Матросова]; Упорядник: Кільченко, А. В. – К. : ІТЗН НАПН України, 2014. – 49 с.
16. Щедріна, О. І. Інтернет-технології в бізнесі : навч. посіб. / О. І. Щедріна, М. М. Агютін. — К. : КНЕУ, 2012. — 303 с.
17. Матросов А. HTML 4.0 / А. Матросов А, Сергєєв, М. Чаунін. - Київ: Komubook, 2019. — 304 с.
18. The evaluation on artificial neural networks (ANN) and multiple linear regressions (MLR) models for predicting concentration / [S. Shams, A. Jahani, S. Kalantary et al.] // Urban Climate. – 2021. – Vol. 37. – P. 1–10.
19. Гаєвський А.Ю. 100% підручник зі створення веб-сторінок і вебсайтів: HTML і JavaScript - Одеса: MDFs, 2019. — 200 с.
20. Magnussen, S. Thek-nearest neighbor technique with local linear regression / S. Magnussen, E. Tomppo // Scandinavian Journal of Forest Research.— 2013. – Vol. 29, Issue 2. – P. 120–131.
21. John Resig. Secrets of the JavaScript Ninja / John Resig, Bear Bibeault, Josip Maras. – Boston: Manning Publications, 2016. – 393 с.
22. Eloquent JavaScript: A Modern Introduction to Programming of Web // The Stata Journal: Promoting communications on statistics and Stata. – 2020.– Vol. 20, Issue 1. – P. 3–29.
23. Олійник, А. О. Інтелектуальний аналіз даних / А. О. Олійник, С. О. Субботін, О. О. Олійник. – Запоріжжя : ЗНТУ, 2012. – 278 с.
24. Субботін, С. О. Нейронні мережі : Навчальний посібник /С. О. Субботін, А. О. Олійник ; під заг. ред. проф. С. О. Субботіна. – Запоріжжя : ЗНТУ, 2014. – 135 с.

25. Park, Y. Artificial Neural Networks: Multilayer Perceptron for Ecological Modeling / Y. Park, S. Lek // Developments in Environmental Modelling. – 2016. – Vol. 28. – P. 123–140.
26. Субботін, С. О. Нейронні мережі : теорія та практика: навч. посіб. / С. О. Субботіна. – Житомир : Вид. О. О. Євенок, 2020. – 184 с.
27. McKinney. Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython / W. McKinney. – Beijing : O'Reilly Media, 2017. –550 p.

ДОДАТОК А

Технічне завдання

Вступ

Розроблюваний програмний продукт використовується для дослідження методів підвищення ефективності онлайн-торгівлі на прикладі вебсайту продажу комп'ютерної техніки. Він являє собою набір файлів за допомогою яких можна переглядати інтерфейс ПП в браузері та виконувати відповідні дії для продажу комп'ютерної техніки. ПП представлений у вигляді окремого вебсайту завдяки якому можна переглянути перелік товарів, переглянути детальні характеристики товару, ознайомитись з ціною та умовами, додати товар до кошика, а також надіслати заявку на купівлю товару до телеграм-бота, що є можливістю підвищення ефективності онлайн-торгівлі для невеликих проектів.

A.1 Мета розробки і її призначення

Об'єкт дослідження – процес онлайн-торгівлі на прикладі вебсайту продажу комп'ютерної техніки.

Предмет дослідження – методи підвищення ефективності онлайн-торгівлі на прикладі вебсайту продажу комп'ютерної техніки.

Мета роботи – дослідження та програмна реалізація методів підвищення ефективності онлайн-торгівлі на прикладі вебсайту продажу комп'ютерної техніки.

Метод дослідження – тестування вебсайту продажу комп'ютерної техніки

Актуальність даної роботи проявляється у багатьох аспектах. Спочатку, слід враховувати зростаючий тренд онлайн-торгівлі, що визначає необхідність вивчення та вдосконалення методів цього процесу, зокрема на прикладі вебсайту продажу комп'ютерної техніки. Крім того, конкуренція в індустрії електронної комерції висока, і ефективні методи приваблення та утримання клієнтів стають ключовими для успіху.

Зростання технологічних можливостей, а також постійні інновації в галузі програмування та технологій, роблять важливим стеження за останніми

тенденціями та адаптацію вебсайтів до новацій для поліпшення користувацького досвіду.

У світлі цих факторів важливою стає ідея поліпшення процесу покупки, розуміння та зручність інтерфейсу, а також оперативна реакція на замовлення, що може вплинути на клієнтське задоволення та лояльність.

Крім того, аналіз результатів та даних, зібраних під час дослідження, надає обґрунтовану базу для прийняття рішень, що є ключовим аспектом при розробці стратегій підвищення ефективності онлайн-торгівлі.

Усі ці аспекти сприяють не лише академічному дослідженню, але й практичній реалізації знань для вдосконалення вебсайтів продажу комп'ютерної техніки, що в кінцевому підсумку сприяє їхній конкурентоспроможності та відповідності сучасним стандартам електронної комерції.

A.2 Стислий зміст основних теоретичних посилань до розробки на основі проведених пошуків

Основні теоретичні концепції для розробки вебсайту продажу комп'ютерної техніки включають розуміння сутності електронної комерції та ключових аспектів онлайн-торгівлі. Також вивчається роль користувацького досвіду (UX) та принципи дизайну інтерфейсу для забезпечення зручності користувача. Застосування методів АБ-тестування для оцінки ефективності нововведень та визначення ключових метрик. Досліджуються сучасні технологічні тенденції, зокрема використання розширених можливостей JavaScript для поліпшення функціоналу. Також проводиться аналіз стратегій та методів конкурентів у сфері електронної комерції. Визначається важливість аналізу даних для прийняття обґрунтованих рішень та формулювання висновків та рекомендацій на основі результатів дослідження. Ці теоретичні підходи слугують фундаментом для розробки та вдосконалення вебсайтів продажу комп'ютерної техніки, враховуючи сучасні стандарти та очікування користувачів.

А.3 Підстави для розробки

Підставою для виконання розробки є завдання на дипломний проєкт на тему “Дослідження та програмна реалізація методів підвищення ефективності онлайн-торгівлі на прикладі вебсайту продажу комп’ютерної техніки ”, затверджене наказом по НУ «Запорізька політехніка» № 437 від 14 листопада 2023 року.

А.4 Основні вимоги до програми, що розробляється, початковим даним і результату

А.4.1 Вимоги до функціональних характеристик

Вимоги до функціональних характеристик:

- повинен бути реалізований інтуїтивний та зрозумілий інтерфейс для користувачів;
- повинен бути реалізований естетичний дизайн, який відображає професіоналізм та модерність;
- повинен бути реалізований зручний каталог товарів з можливістю швидкого та зручного пошуку;
- повинна бути реалізована система фільтрації за різними параметрами (ціною, характеристиками тощо);
- повинна бути реалізована можливість простого та зрозумілого процесу оформлення замовлення;
- повинна бути реалізована можливість перегляду та редагування кошика перед підтвердженням;
- повинен бути реалізований мінімальний час завантаження сторінок та швидка реакція на дії користувачів;
- повинна бути реалізована адаптація для різних типів пристроїв та браузерів;

- повинна бути реалізована можливість залишення відгуків та оцінок продуктів;
- необхідно забезпечити використання сучасних технологій для підтримки різноманітних функцій.

A.4.2 Вимоги до надійності

ПП повинна виконувати наступні вимоги до надійності:

- захищати ПП від спроби ввести дані до форми, тип яких не відповідає вимогам системи;
- захищати ПП від шкідливих дій користувача (наприклад користувач постійно вводить одні і ті ж дані для того щоб порушити функціональність ПП).

ПП повинен нормально функціонувати при безперервній роботі комп'ютера. У разі виникнення збоїв в роботі апаратури, відновлення нормальної роботи ПП повинно проводитися після перезавантаження системи.

A.4.3 Умови експлуатації

Умови експлуатації:

- програмне забезпечення не потребує додаткового обслуговуючого персоналу;
- для користування програмним забезпеченням не потрібна додаткова кваліфікація.

A.4.4 Вимоги до складу та параметрів технічних засобів

Вимоги до складу та параметрів технічних засобів:

- платформа x64-based;
- операційна система Microsoft Windows 10/8/7 (рекомендується встановити всі останні оновлення);

- процесор з мінімальною тактовою частотою 1,8 ГГц. Рекомендується використовувати як мінімум двоядерний процесор;
- оперативна пам'ять від 2 ГБ (рекомендується 4 ГБ та мінімум 2,5 ГБ при використанні на віртуальній машині);
- роздільна здатність екрану 1024x768 пікселів з 16-бітовим кольором;
- мінімум 2 ГБ вільного місця на жорсткому диску.
- середовище SublimeText;
- браузер.

А.4.5 Вимоги до маркування та упакування

Програмне забезпечення не потребує додаткових дій при маркуванні та упакуванні.

А.4.6 Вимоги до транспортування та збереження

Вимоги до транспортування й зберігання аналогічні вимогам, які пропонуються до носія, на якому записана програма.

A.5 Вимоги до програмної документації

Склад програмної документації повинен включати в себе:

- технічне завдання;
- керівництво програміста;
- керівництво оператора.

A.6 Основні обмеження на установку і використання програмного продукту

Розроблюваний програмний продукт не потребує інсталяції. Програмний продукт повинен використовуватись на комп'ютерах які задовільняють вимогам та параметрам технічних засобів. Програмний продукт може використовуватись користувачем у будь-який момент часу, не конфліктуючи з іншими програмами.

ДОДАТОК Б

Опис програмного продукту

Б.1 Загальні відомості

Назва програмного продукту – «TechHaven».

Розроблюваний програмний продукт використовується для дослідження методів підвищення ефективності онлайн-торгівлі на прикладі вебсайту продажу комп'ютерної техніки. Він являє собою набір файлів за допомогою яких можна переглядати інтерфейс ПП в браузері та виконувати відповідні дії для продажу комп'ютерної техніки. ПП представлений у вигляді окремого вебсайту завдяки якому можна переглянути перелік товарів, переглянути детальні характеристики товару, ознайомитись з ціною та умовами, додати товар до кошика, а також надіслати заявку на купівлю товару до телеграм-бота, що є можливістю підвищення ефективності онлайн-торгівлі для невеликих проектів

Для коректної роботи ПП необхідні такі мінімальні системні вимоги:

- платформа x64-based;
- операційна система Microsoft Windows 10/8/7 (рекомендується встановити всі останні оновлення);
- процесор з мінімальною тактовою частотою 1,8 ГГц. Рекомендується використовувати як мінімум двоядерний процесор;
- оперативна пам'ять від 2 ГБ (рекомендується 4 ГБ та мінімум 2,5 ГБ при використанні на віртуальній машині);
- роздільна здатність екрану 1024x768 пікселів з 16-бітовим кольором;
- мінімум 2 ГБ вільного місця на жорсткому диску.
- середовище SublimeText;
- браузер.

Б.2 Функціональне призначення

Вимоги до функціональних характеристик:

- повинен бути реалізований інтуїтивний та зрозумілий інтерфейс для користувачів;

- повинен бути реалізований естетичний дизайн, який відображає професіоналізм та модерність;
- повинен бути реалізований зручний каталог товарів з можливістю швидкого та зручного пошуку;
- повинна бути реалізована система фільтрації за різними параметрами (ціною, характеристиками тощо);
- повинна бути реалізована можливість простого та зрозумілого процесу оформлення замовлення;
- повинна бути реалізована можливість перегляду та редагування кошика перед підтвердженням;
- повинен бути реалізований мінімальний час завантаження сторінок та швидка реакція на дії користувачів;
- повинна бути реалізована адаптація для різних типів пристроїв та браузерів;
- повинна бути реалізована можливість залишення відгуків та оцінок продуктів;
- необхідно забезпечити використання сучасних технологій для підтримки різноманітних функцій.

Б.3 Опис логічної структури

На рис. Б.1 представлена діаграма класів програмного продукту.

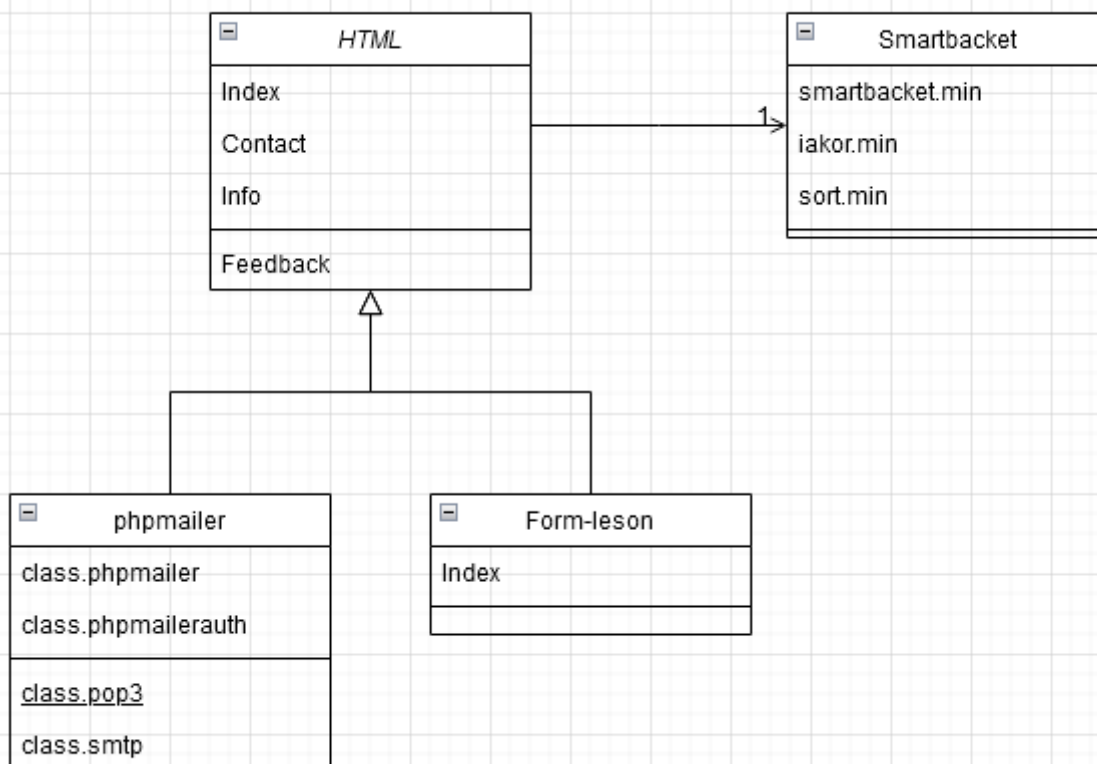


Рисунок Б.1 – Діаграма класів програмного продукту

Модуль HTML– головний модуль проєкту, в якому буде відбуватися відображення програмного продукту та елементів інтерфейсу користувача, він має містити такі методи:

- index– призначений для виводу інтерфейсу програмного продукту;
- contact– призначений для можливості заповнити форму та зв’язатись з розробником;
- info– призначений для завантаження інформації;
- feedback– призначений для обробки відгуків по часу, імені та найменуванню товару.

Модуль phpmailer– призначений для попередньої обробки даних, він зберігає дані та надсилає їх в подальшому на фронтенд.

Модуль smartbasket– призначений для реалізації можливості замовлення товару, підрахунку кількості, суми і іншої інформації про товар, а також

валідності введених даних користувачем по деяким полям, він має містити такі методи:

- `ia.kor.min`– призначений для прив'язки товару по заданому раніше номеру;
- `smartbasket.min`– призначений для візуалізації оброблених даних кошику і передачу їх на фронтенд в прийнятному для користувача вигляді з урахуванням виконаних операцій;
- `sort.min`– призначений для сортування товару за певними категоріями, як то ціна товару(по зростанню, по спаданню), також по найвідвідуванішим товарам.

Б.4 Вибір інструментарію розробки програмного продукту

Для розробки програмного продукту було обрано мову програмування JavaScript, бо вона володіє тими перевагами, що вона легко адаптується і може ефективно працювати у будь-якому середовищі. JavaScript обрано як мову програмування для розробки програмного продукту через її універсальність та широкі можливості у веб-розробці. JavaScript дозволяє створювати динамічні та інтерактивні вебсайти, реагувати на події користувача, а також взаємодіяти з сервером без перезавантаження сторінки. Ця мова програмування підтримує багатофункціональний та асинхронний підхід, що робить її ідеальним вибором для створення сучасних та ефективних веб-додатків. JavaScript також є стандартом у веб-розробці, що забезпечує його велику спільноту та велику кількість ресурсів для навчання та підтримки. Такий вибір дозволяє розробникам створювати масштабовані, швидкі та інноваційні веб-додатки з урахуванням сучасних стандартів розробки.

Для розробки ПП було обрано середовище SublimeText, бо воно має такі переваги як простота в освоєнні, кросплатформне з відкритим вихідним кодом. Середовище розробки Sublime Text вибрано для його зручності та швидкості.

Програмісти відзначають його легкість використання, можливості розширення за допомогою плагінів та швидкодію. Sublime Text надає ефективний інтерфейс для написання та відлагодження коду, що робить його відмінним вибором для розробників у роботі над програмним продуктом.

Б.5 Виклик і завантаження

Для запуску ПП необхідно запустити на робочому столі файл index.html, після чого відкриється головне вікно, яке представлено на рис. Б.2. Програмний продукт «TechHaven» завантажений і готовий до роботи».

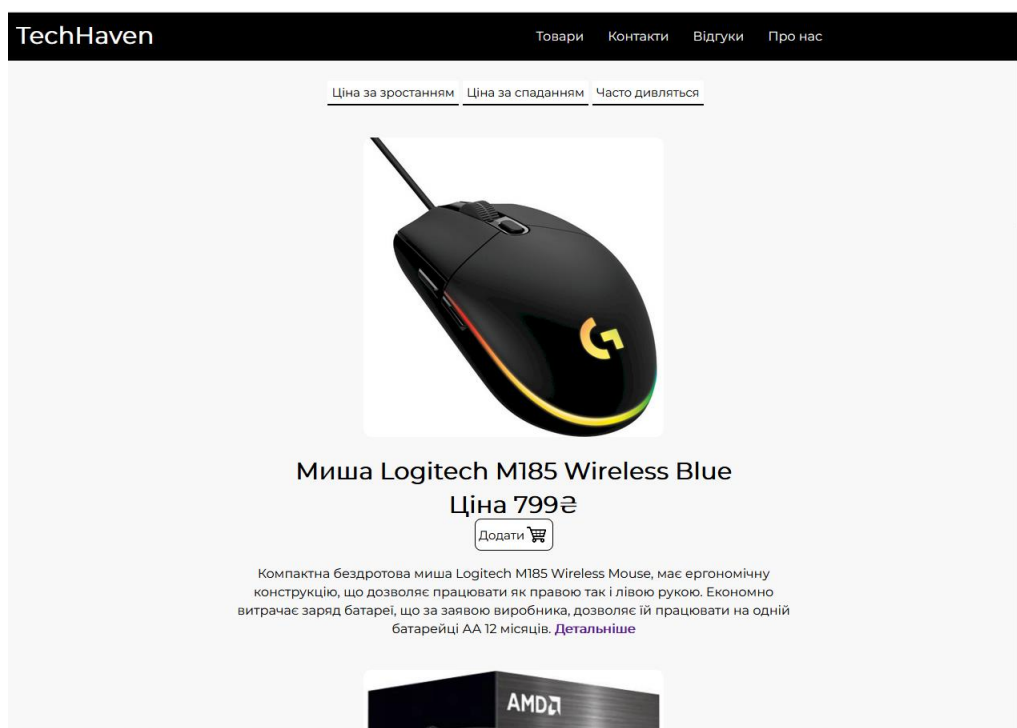


Рисунок Б.2 – Головне вікно ПП

Б.6 Приклади реалізації

Після вдалого запуску ПП відкривається головна сторінка, яка представлена на Б. 3, на якій буде відображено товари і кожен з них готовий до більш детального перегляду.

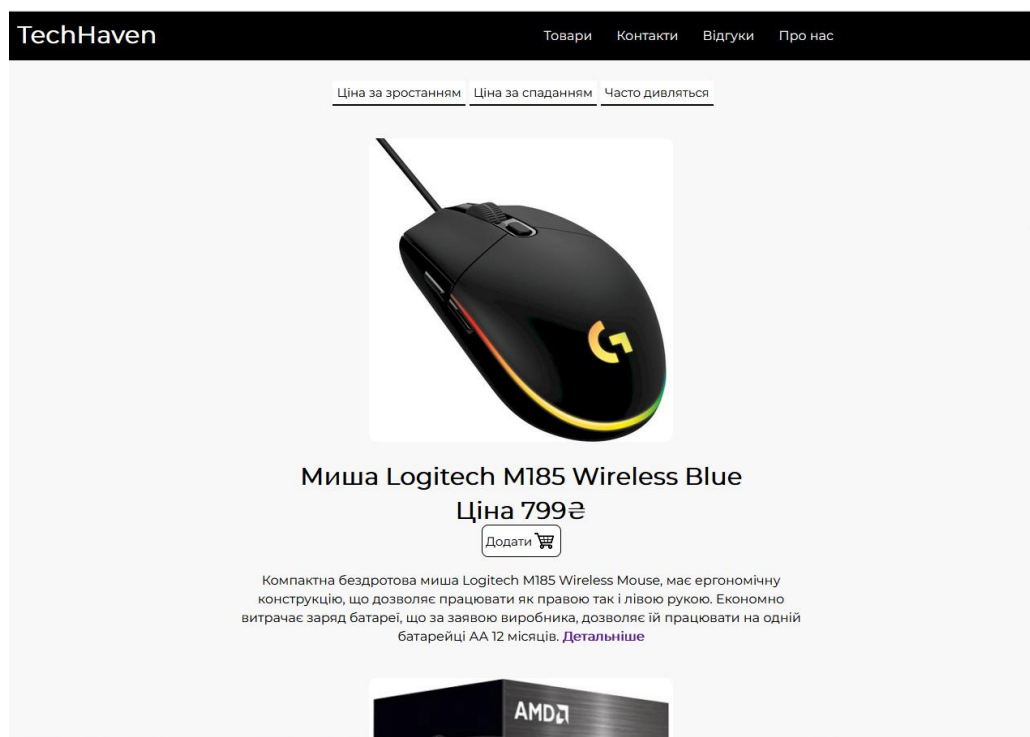


Рисунок Б.3 – Головне вікно ПП

Для того, щоб здійснити замовлення товару, спочатку необхідно обрати один або декілька товарів із запропонованих, можливо, за потреби перед цим переглянувши більш детальні характеристики товарів клікнувши на кожен з них. Потім необхідно натиснути на кнопку Додати і в такому випадку користувач побачить на екрані інтерактивне спливаюче вікно в якому буде можливість обрати дві опції, а саме – перейти до кошика, або продовжувати покупки. При цьому інтерактивне спливаюче вікно буде автоматично підлаштовувати інформацію для клієнта, якщо він вже рвніше обрав цей клас товарів, вікно покаже на екрані повідомлення що даний товар вже був раніше обраний, якщо ж користувач обрав новий товар то повідомлення на екрані буде містити інформацію, що товар додано до корзини. Доволі важливо було реалізувати це саме таким чином для того щоб користувач, якщо він володіє низьким рівнем навичок не заплутався в ПП. Приклад виведення результатів додавання товару до кошика в наведеному ПП приведено на рис. Б.4.



Рисунок Б.4 – Вікно виводу додавання товару до кошика

Якщо користувач натиснув кнопку перейти до кошика, щоб здійснити покупку користувач переходить до наступної сторінки. На цій сторінці відбувається процес залишення замовлення на купівлю товару. В вікні відображається ID товару який замовив покупець, його ціна та кількість одиниць товару, при чому користувач може змінювати кількість безпосередньо в кошику і ціна буде за нею підлаштовуватись.

Також є кілька полів з інформацією, яку користувач повинен ввести для того щоб замовити товар – ім'я, пошту та номер телефону. Поля імені та номеру телефону є обов'язковими, бо заявка на товар буде надсилатись до месенджера телеграм-бота і для подальшого зв'язку з покупцем адміністратору необхідно мати такі дані. Також покупець повинен прийняти умови користувацької угоди і надалі або продовжити покупки або залишити заявку на замовлення товару. Приклад реалізованої можливості замовлення товару в рамках ПП приведено на рис. Б.5

Товар	ID	Ціна / Грн	Кількість метрів	Загальна ціна / Грн	Видалити
 Acer Aspire 7	79	30999	- 1 +	30999	×
 TP-Link Archer C64	83	1399	- 2 +	2798	×
 AMD Ryzen 5	97	6239	- 1 +	6239	×
 Logitech M185 Wireless Blue	31431	799	- 6 +	4794	×

Загальна вартість: 8 Загальна вартість: 44830.00 Грн

☒ Я приймаю умови користувацької угоди

Продовжити покупки

Зробити замовлення

Рисунок Б.5 – Вікно замовлення товару

Для того щоб користувачу було зручніше обирати товар в ПП реалізована можливість сортування товарів на сторінці. Сортування можливе по трьом параметрам – по зростанню ціні, спаданню та найбільш популярним товарам. Приклад реалізованої можливості сортування товарів в рамках ПП приведено на рис. Б.6

Ціна за зростанням

Ціна за спаданням

Часто дивляться

Рисунок Б.6 – Вікно сортування товарів

Після кожного товару в ПП реалізована можливість залишення відгуку на сторінці, для того щоб кожен користувач міг поділитись власним досвідом по роботі з товаром і допомогти іншим користувачам обрати потрібний саме їм товар по певним характеристикам. Для того щоб залишити відгук потрібно заповнити поле імені та безпосередньо відгук, а час вираховується автоматично. Приклад залишення відгуку в рамках ПП приведено на рис. Б.7

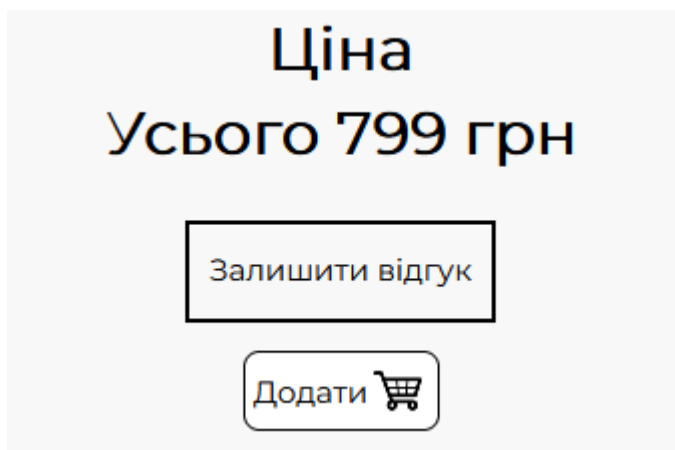


Рисунок Б.7 – Вікно залишення відгуку.

Б.7 Вхідні та вихідні дані

Модуль HTML призначений для відображення інтерфейсу програмного продукту та включає методи `index`, `contact`, `info`, `feedback`. Вхідні дані - це запитання або дії користувача, які викликають відповідні методи, а вихідні - це відображення програмного інтерфейсу, форми для зв'язку, завантаженої інформації та оброблених відгуків.

Модуль `phpmailer` служить для попередньої обробки даних, їх збереження та відправлення на фронтенд. Вхідні дані - це дані, які користувач вводить через форму, а вихідні - це оброблені дані, готові для відправлення або збереження.

Модуль `smartbasket` реалізує можливість замовлення товару, підрахунку інформації про товар та валідності введених даних. Вхідні дані - це дії користувача, що викликають методи `iaikor.min`, `smartbasket.min`, `sort.min`, а вихідні - це прив'язаний товар, візуалізовані дані кошику та відсортований товар відповідно

ДОДАТОК В

Текст програми

В.1 Файл index.html

```

<div class="content">
  <div class="button">
    <button id="sort-up">Ціна за зростанням</button>
    <button id="sort-down">Ціна за спаданням</button>
    <button id="sort-rating">Часто дивляться</button>
  </div>
  <div id="nav">
    <!-- data-sort - Уживаємо ціну товару по якій буде сортування; data-
rating - вказуємо 1-2 пріоритет товару; -->
    <div data-sort="799" data-rating="1" class="content-shop-product">
      <div class="img1">
        <a href="Sintelon Dragon.html">
          
        </a>
      </div>
      <div class="text">
        <h3><a href="Sintelon Dragon.html">Миша Logitech
M185 Wireless Blue</a></h3>
        <h3>
          <a href="Sintelon Dragon.html">Ціна
7999</a>
          <button class="basket"
            data-sb-id-or-vendor-code="31431"
            data-sb-product-name="Logitech M185
Wireless Blue"
            data-sb-product-price="799"
            data-sb-product-quantity="1"
            data-sb-product-img="img\1.webp">
            <p>Додати</p>
            
          </button>
        </h3>
      </div>
      <span>
        <p>Компактна бездротова миша Logitech M185
Wireless Mouse, має ергономічну конструкцію, що дозволяє працювати як правою так і
лівою рукою. Економно витрачає заряд батарей, що за заявою виробника, дозволяє їй
працювати на одній батарейці AA 12 місяців. <strong id="link"><a href="Sintelon
Dragon.html"> Детальніше</a></strong></p>
      </span>
    </div>
    <div data-sort="6329" data-rating="2" class="content-shop-product">
      <div class="img1">
        <a href="Betap Turbo 97.html">
          
        </a>
      </div>
      <div class="text">
        <a href="Betap Turbo 97.html"><h3>Процесор AMD
Ryzen 5 5600X 3.7 GHz / 32 MB</h3></a>
        <h3><a href="Betap Turbo 97.html">Ціна 6 3299</a>
          <button class="basket"
            data-sb-id-or-vendor-code="97"
            data-sb-product-name="AMD Ryzen 5"
            data-sb-product-price="6239"
            data-sb-product-quantity="1"
            data-sb-product-img="img\2.webp">

```

```

        <p>Додати</p>
        
    </button>
</h3>

```

В.2 Файл feedback.html

```

<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.7/dist/umd/popper.min.js"
integrity="sha384-
zYPOMqeu1DAVhKiLqWBUTcbYfZ8osu1Nd6Z89ify25QV9guujx43ITvfi12/QExE"
crossorigin="anonymous"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/js/bootstrap.min.js"
integrity="sha384-
Y4oOpwW3duJdCWv5ly8SCFYWqFDsfob/3GkgExXKV4idmbt98QcxXYs9UoXAB7BZ"
crossorigin="anonymous"></script>
    <div class="top-skrol"></div>
<script src="js/iakor.js"></script>
<script src="js/feedback.js"></script>
<script src="https://code.jquery.com/jquery-3.3.1.min.js"></script>
<link rel="stylesheet" href="smartbasket/css/smartbasket.min.css">

<div class="smart-basket__wrapper"></div>

<script src="./smartbasket/js/smartbasket.min.js"></script>

<script>
    $(function () {
        $('.smart-basket__wrapper').smbasket({
            productElement: 'content-shop-product',
            buttonAddToBasket: 'basket',
            productPrice: 'product__price-number',
            productSize: 'product__size-element',

            productQuantityWrapper: 'product__quantity',
            smartBasketMinArea: 'basket--head',
            countryCode: '+38',
            smartBasketCurrency: 'Грн',

            agreement: {
                isRequired: true,
                isChecked: true,
                isLink: '#',
            },
            nameIsRequired: false,
        });
    });
</script>

```

В.3 Файл smartbasket.min

```

document.addEventListener("DOMContentLoaded", function () {
    var t, a, e, s = s || {};
    s.scope = {}, s.findInternal = function (t, a, e) {
        t instanceof String && (t = String(t));
        for (var s = t.length, r = 0; r < s; r++) {
            var n = t[r];
            if (a.call(e, n, r, t)) return {
                i: r,

```

```

        v: n
    }
}
return {
    i: -1,
    v: void 0
}
}, s.ASSUME_ES5 = !1, s.ASSUME_NO_NATIVE_MAP = !1, s.ASSUME_NO_NATIVE_SET = !1,
s.SIMPLE_FROUND_POLYFILL = !1, s.defineProperty = s.ASSUME_ES5 || "function" == typeof
Object.defineProperty ? Object.defineProperty : function (t, a, e) {
    t != Array.prototype && t != Object.prototype && (t[a] = e.value)
}, s.getGlobal = function (t) {
    return "undefined" != typeof window && window === t ? t : "undefined" != typeof global && null != global ?
global : t
}, s.global = s.getGlobal(this), s.polyfill = function (t, a, e, r) {
    if (a) {
        for (e = s.global, t = t.split("."), r = 0; r < t.length - 1; r++) {
            var n = t[r];
            n in e || (e[n] = {}), e = e[n]
        }
        (a = a(r = e[t = t[t.length - 1]])) != r && null != a && s.defineProperty(e, t, {
            configurable: !0,
            writable: !0,
            value: a
        })
    }
}, s.polyfill("Array.prototype.find", function (t) {
    return t || function (t, a) {
        return s.findInternal(this, t, a).v
    }
}, "es6", "es3"), t = function (t) {
    var a = function (a, e, s) {
        var r = {
            invalid: [],
            getCaret: function () {
                try {
                    var t = 0,
                        e = a.get(0),
                        s = document.selection,
                        n = e.selectionStart;
                    if (s && -1 === navigator.appVersion.indexOf("MSIE 10")) {
                        var i = s.createRange();
                        i.moveStart("character", -r.val().length), t = i.text.length
                    } else (n || "0" === n) && (t = n);
                    return t
                } catch (t) {}
            },
            setCaret: function (t) {
                try {
                    if (a.is(":focus")) {
                        var e = a.get(0);
                        if (e.setSelectionRange) e.setSelectionRange(t, t);
                        else {
                            var s = e.createTextRange();
                            s.collapse(!0), s.moveEnd("character", t), s.moveStart("character", t), s.select()
                        }
                    }
                } catch (t) {}
            },
            events: function () {
                a.on("keydown.mask", function (t) {
                    a.data("mask-keycode", t.keyCode || t.which), a.data("mask-previous-value", a.val()), a.data("mask-
                    previous-caret-pos", r.getCaret()), r.maskDigitPosMapOld = r.maskDigitPosMap
                }).on(t.jMaskGlobals.useInput ? "input.mask" : "keyup.mask", r.behaviour).on("paste.mask drop.mask",

```

```

function () {
    setTimeout(function () {
        a.keydown().keyup()
    }, 100)
    }).on("change.mask", function () {
        a.data("changed", !0)
    }).on("blur.mask", function () {
        o === r.val() || a.data("changed") || a.trigger("change"), a.data("changed", !1)
    }).on("blur.mask", function () {
        o = r.val()
    }).on("focus.mask", function (a) {
        !0 === s.selectOnFocus && t(a.target).select()
    }).on("focusout.mask", function () {
        s.clearIfNotMatch && !n.test(r.val()) && r.val("")
    })
},
getRegexMask: function () {
    for (var t, a, s, r, n = [], o = 0; o < e.length; o++)(t = i.translation[e.charAt(o)]) ? (a =
t.pattern.toString().replace(/.{1}$|^.{1}/g, ""), s = t.optional, (t = t.recursive) ? (n.push(e.charAt(o)), r = {
    digit: e.charAt(o),
    pattern: a
    }) : n.push(s || t ? a + "?" : a)) : n.push(e.charAt(o).replace(/[-\\$*+?.(){}]/g, "\\$&"));
    return n = n.join(""), r && (n = n.replace(new RegExp("(" + r.digit + "(.*)" + r.digit + ")?"),
"$($1)?").replace(new RegExp(r.digit, "g"), r.pattern)), new RegExp(n)
},
destroyEvents: function () {
    a.off("input keydown keyup paste drop blur focusout ".split(" ").join(".mask "))
},
val: function (t) {
    var e = a.is("input") ? "val" : "text";
    return 0 < arguments.length ? (a[e]() !== t && a[e](t), e = a) : e = a[e](), e
},
calculateCaretPosition: function (t) {
    var e = r.getMasked(),
        s = r.getCaret();
    if (t !== e) {
        var n = a.data("mask-previous-caret-pos") || 0;
        e = e.length;
        var i, o = t.length,
            l = t = 0,
            c = 0,
            p = 0;
        for (i = s; i < e && r.maskDigitPosMap[i]; i++) l++;
        for (i = s - 1; 0 <= i && r.maskDigitPosMap[i]; i--) t++;
        for (i = s - 1; 0 <= i; i--) r.maskDigitPosMap[i] && c++;
        for (i = n - 1; 0 <= i; i--) r.maskDigitPosMapOld[i] && p++;
        s > o ? s = 10 * e : n >= s && n !== o ? r.maskDigitPosMapOld[s] || (n = s, s = s - (p - c) - t,
r.maskDigitPosMap[s] && (s = n)) : s > n && (s = s + (c - p) + l)
    }
    return s
},
behaviour: function (e) {
    e = e || window.event, r.invalid = [];
    var s = a.data("mask-keycode");
    if (-1 === t.inArray(s, i.byPassKeys)) {
        s = r.getMasked();
        var n = r.getCaret(),
            o = a.data("mask-previous-value") || "";
        return setTimeout(function () {
            r.setCaret(r.calculateCaretPosition(o))
        }, t.jMaskGlobals.keyStrokeCompensation), r.val(s), r.setCaret(n), r.callbacks(e)
    }
},

```

```

getMasked: function (t, a) {
  var n, o = [],
      l = void 0 === a ? r.val() : a + "",
      c = 0,
      p = e.length,
      d = 0,
      u = l.length,
      m = 1,
      k = "push",
      b = -1,
      f = 0;
  if (a = [], s.reverse) {
    k = "unshift", m = -1;
    var v = 0;
    c = p - 1, d = u - 1;
    var _ = function () {
      return -1 < c && -1 < d
    }
  } else v = p - 1, _ = function () {
    return c < p && d < u
  };
  for (; _(); ) {
    var h = e.charAt(c),
        g = l.charAt(d),
        y = i.translation[h];
    y ? (g.match(y.pattern) ? (o[k](g), y.recursive && (-1 === b ? b = c : c === v && c !== b && (c = b -
m), v === b && (c -= m)), c += m) : g === n ? (f--, n = void 0) : y.optional ? (c += m, d -= m) : y.fallback ?
(o[k](y.fallback), c += m, d -= m) : r.invalid.push({
      p: d,
      v: g,
      e: y.pattern
    })), d += m) : (t || o[k](h), g === h ? (a.push(d), d += m) : (n = h, a.push(d + f), f++, c += m)
  }
  return t = e.charAt(v), p !== u + 1 || i.translation[t] || o.push(t), o = o.join(""), r.mapMaskdigitPositions(o, a,
u), o
},
mapMaskdigitPositions: function (t, a, e) {
  for (t = s.reverse ? t.length - e : 0, r.maskDigitPosMap = {}, e = 0; e < a.length; e++)
r.maskDigitPosMap[a[e] + t] = 1
},
callbacks: function (t) {
  var n = r.val(),
      i = n !== o,
      l = [n, t, a, s],
      c = function (t, a, e) {
        "function" === typeof s[t] && a && s[t].apply(this, e)
      };
  c("onChange", !0 === i, l), c("onKeyPress", !0 === i, l), c("onComplete", n.length === e.length, l),
c("onInvalid", 0 < r.invalid.length, [n, t, a, r.invalid, s])
}
};
a = t(a);
var n, i = this,
    o = r.val();
e = "function" === typeof e ? e(r.val(), void 0, a, s) : e, i.mask = e, i.options = s, i.remove = function () {
  var t = r.getCaret();
  return i.options.placeholder && a.removeAttr("placeholder"), a.data("mask-maxlength") &&
a.removeAttr("maxlength"), r.destroyEvents(), r.val(i.getCleanVal()), r.setCaret(t), a
}, i.getCleanVal = function () {
  return r.getMasked(!0)
}, i.getMaskedVal = function (t) {
  return r.getMasked(!1, t)
}, i.init = function (o) {

```

```

        if (o = o || !1, s = s || {}, i.clearIfNotMatch = t.jMaskGlobals.clearIfNotMatch, i.byPassKeys =
t.jMaskGlobals.byPassKeys, i.translation = t.extend({}, t.jMaskGlobals.translation, s.translation), i = t.extend(!0, {}, i,
s), n = r.getRegexMask(), o) r.events(), r.val(r.getMasked());
    else {
        s.placeholder && a.attr("placeholder", s.placeholder), a.data("mask") && a.attr("autocomplete", "off"), o =
0;

        for (var l = !0; o < e.length; o++) {
            var c = i.translation[e.charAt(o)];
            if (c && c.recursive) {
                l = !1;
                break
            }
        }
        l && a.attr("maxlength", e.length).data("mask-maxlength", !0), r.destroyEvents(), r.events(), o =
r.getCaret(), r.val(r.getMasked()), r.setCaret(o)
    }
}, i.init(!a.is("input"))
};
t.maskWatchers = {};
var e = function () {
    var e = t(this),
        r = {},
        n = e.attr("data-mask");
    if (e.attr("data-mask-reverse") && (r.reverse = !0), e.attr("data-mask-clearifnotmatch") &&
(r.clearIfNotMatch = !0), "true" === e.attr("data-mask-selectonfocus") && (r.selectOnFocus = !0), s(e, n, r)) return
e.data("mask", new a(this, n, r))
},
s = function (a, e, s) {
    s = s || {};
    var r = t(a).data("mask"),
        n = JSON.stringify;
    a = t(a).val() || t(a).text();
    try {
        return "function" === typeof e && (e = e(a)), "object" !== typeof r || n(r.options) !== n(s) || r.mask !== e
    } catch (t) {}
},
r = function (t) {
    var a = document.createElement("div"),
        e = (t = "on" + t) in a;
    return e || (a.setAttribute(t, "return;"), e = "function" === typeof a[t]), e
};
t.fn.mask = function (e, r) {
    r = r || {};
    var n = this.selector,
        i = t.jMaskGlobals,
        o = i.watchInterval;
    i = r.watchInputs || i.watchInputs;
    var l = function () {
        if (s(this, e, r)) return t(this).data("mask", new a(this, e, r))
    };
    return t(this).each(l), n && "" !== n && i && (clearInterval(t.maskWatchers[n]), t.maskWatchers[n] =
setInterval(function () {
        t(document).find(n).each(l)
    }, o)), this
}, t.fn.masked = function (t) {
    return this.data("mask").getMaskedVal(t)
}, t.fn.unmask = function () {
    return clearInterval(t.maskWatchers[this.selector]), delete t.maskWatchers[this.selector], this.each(function () {
        var a = t(this).data("mask");
        a && a.remove().removeData("mask")
    })
}, t.fn.cleanVal = function () {
    return this.data("mask").getCleanVal()
}

```

```

}, t.applyDataMask = function (a) {
  ((a = a || t.jMaskGlobals.maskElements) instanceof t ? a : t(a)).filter(t.jMaskGlobals.dataMaskAttr).each(e)
}, r = {
  maskElements: "input,td,span,div",
  dataMaskAttr: "[data-mask]",
  dataMask: !0,
  watchInterval: 300,
  watchInputs: !0,
  keyStrokeCompensation: 10,
  useInput: !/ChromeV[2-4][0-9]|SamsungBrowser/.test(window.navigator.userAgent) && r("input"),
  watchDataMask: !1,
  byPassKeys: [9, 16, 17, 18, 36, 37, 38, 39, 40, 91],
  translation: {
    0: {
      pattern: /\d/
    },
    9: {
      pattern: /\d/,
      optional: !0
    },
    "#": {
      pattern: /\d/,
      recursive: !0
    },
    A: {
      pattern: /[a-zA-Z0-9]/
    },
    S: {
      pattern: /[a-zA-Z]/
    }
  }
}, t.jMaskGlobals = t.jMaskGlobals || {}, (r = t.jMaskGlobals = t.extend(!0, {}, r, t.jMaskGlobals)).dataMask &&
t.applyDataMask(), setInterval(function () {
  t.jMaskGlobals.watchDataMask && t.applyDataMask()
}, r.watchInterval)
}, a = window.jQuery, e = window.Zepto, "function" == typeof define && define.amd ? define(["jquery"], t) :
"object" == typeof exports && "undefined" == typeof Meteor ? module.exports = t(require("jquery")) : t(a || e)
}), document.addEventListener("DOMContentLoaded", function () {
  var t, a, e;
  t = jQuery, a = {
    productElement: "product__element",
    buttonAddToBasket: "product__add-to-cart-button",
    countryCode: "+38",
    smartBasketCurrency: "€",
    productQuantityWrapper: !1,
    productPrice: !1,
    productSize: !1,
    smartBasketMinArea: "header__basket-min",
    smartBasketMinIconPath: "/smartbasket/img/shopping-basket-wight.svg",
    smartBasketMinText: !1,
    tellsRequired: !0,
    emailIsRequired: !1,
    nameIsRequired: !0,
    localStorageName: "basketItems",
    yaCounter: {
      counterID: "",
      targetID: ""
    }
  }, e = {
    getProducts: function () {
      return JSON.parse(localStorage.getItem(a.localStorageName))
    },
    setProducts: function (t) {

```

```

    let e = JSON.stringify(t, "", 4);
    return localStorage.setItem(a.localStorageName, e), !1
  },

```

B.4 Файл sort.js

```

function mySortWrapper(sortType, isDescending) {
  let nav = document.querySelector('#nav');
  for (let i = 0; i < nav.children.length; i++) {
    for (let k = i; k < nav.children.length; k++){
      let firstElemValue = +nav.children[i].getAttribute(sortType);
      let secondElemValue = +nav.children[k].getAttribute(sortType);
      let isNeedSwap = isDescending ? (firstElemValue < secondElemValue) : (firstElemValue > secondElemValue);
      if (isNeedSwap) {
        replacedNode = nav.replaceChild(nav.children[k], nav.children[i]);
        insertAfter(replacedNode, nav.children[i]);
      }
    }
  }
}

function insertAfter(elem, refElem){
  return refElem.parentNode.insertBefore(elem, refElem.nextSibling);
}

document.querySelector('#sort-up').onclick = function(){ mySortWrapper('data-sort', false); }
document.querySelector('#sort-down').onclick = function(){ mySortWrapper('data-sort', true); }
document.querySelector('#sort-rating').onclick = function(){ mySortWrapper('data-rating', false); }

```

B.5 Iacor.js

```

const goTopBtn = document.querySelector('.top-skrol');

goTopBtn.addEventListener("click", function() {
  if (window.pageYOffset > 0) {
    window.scrollTo(0, -35);
    setTimeout(arguments.callee, 0);
  }
});

window.addEventListener("scroll", function() {
  const offset = window.pageYOffset;
  const coords = document.documentElement.clientHeight;
  if (offset > coords) {
    goTopBtn.classList.add("top-skrol--show");
  } else {
    goTopBtn.classList.remove("top-skrol--show");
  }
}

```

ДОДАТОК Г

Слайди презентації

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЗАПОРІЗЬКА ПОЛІТЕХНІКА»
КАФЕДРА ПРОГРАМНИХ ЗАСОБІВ

Дослідження та програмна реалізація методів підвищення ефективності онлайн- торгівлі на прикладі вебсайту продажу комп'ютерної техніки

ВИКОНАВ СТУДЕНТ ГРУПИ КНТ-222М
КУЧЕРЕНКО Я.К.

КЕРІВНИК
ДОЦЕНТ, К.Т.Н СТЕПАНЕНКО О.О

Рисунок Г.1 – Слайд № 1

Мета, об'єкт, предмет дослідження

Об'єкт дослідження – процес онлайн-торгівлі на прикладі вебсайту.

Предмет дослідження – методи підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту.

Мета роботи – дослідження та програмна реалізація методів підвищення ефективності процесу онлайн-торгівлі на прикладі вебсайту

Рисунок Г.2 – Слайд № 2

Актуальність даної роботи

Актуальність даної роботи проявляється у багатьох аспектах.

Слід враховувати постійно зростаючий тренд онлайн-торгівлі, що визначає необхідність **вивчення та вдосконалення** методів цього процесу.

Крім того, конкуренція в індустрії електронної комерції висока, і ефективні методи приваблення та утримання клієнтів стають **ключовими для успіху**.

Зростання технологічних можливостей, а також постійні інновації в галузі програмування та технологій, роблять важливою **адаптацію вебсайтів** до новацій для поліпшення користувацького досвіду. Особливо важливим є поліпшення процесу покупки, розуміння та зручність інтерфейсу, а також **оперативна реакція на замовлення**, що може вплинути на клієнтське задоволення та лояльність.

Рисунок Г.3 – Слайд № 3

Вибір мови програмування

Вибір мови програмування для сайту є **ключовим етапом** в процесі розробки. **Провівши порівняльний аналіз** мною було обрано для реалізації роботи мову програмування **JavaScript**, тому що вона є простою в навчанні та використанні, є кросплатформеною, має можливість працювати з великим обсягом даних, гарно інтегрується з іншими мовами, має сильну підтримку спільноти і наявність гарної документації

Критерії порівняння	JavaScript	PHP	Python	Go	Ruby
Ліцензія	GNU	—	Python Software Foundation License	—	MIT
Простота навчання та написання коду	+	—	+	—	—
Кросплатформеність	+	+	+	+	+
Можливість обробляти операції з великим обсягом даних	+	—	+	+	—
Паралельне програмування	+	+	+	+	+
Наявність бібліотек для машинного навчання	±	±	+	±	+
Інтеграція з іншими мовами	+	+	+	±	±
Наявність зрозумілої та повної документації	+	+	+	+	+
Висока візуалізація даних	+	—	+	—	±
Підвищена продуктивність	+	+	+	+	+
Великий список предметно-орієнтованих пакетів з відкритим вихідним кодом	+	+	+	+	—
Підтримка спільноти	+	+	+	+	—

Рисунок Г.4 – Слайд № 4

Вибір середовища розробки

Важливо обрати середовище розробки найбільш зручне для розробника в вирішенні певної задачі.

Проаналізувавши можливі варіанти я прийшов до висновку, що найбільші переваги має середовище розробки **Sublime Text**, тому що, воно ліцензоване, що є дуже важливим для розробника, також воно має такі переваги як простота в освоєнні, має можливість перевірки та аналізу коду, а також автозавершення коду, містить важливі функції, як то підсвітка коду та за можливості кожен розробник може встановити необхідні додатки для того щоб зробити середовище зручним саме для себе.

Критерії порівняння	Sublime Text	PyCharm	IntelliJ Idea	Visual Studio Code
Ліцензія	+	+	–	+
Кросплатформність	+	+	+	–
Автозавершення коду	+	+	+	+
Підтримка репозиторіїв	+	+	+	+
Простота встановлення	+	–	+	–
Перевірка та аналіз коду	+	+	+	+
Швидкий та безпечний рефакторинг коду	+	+	+	+
Наявність інтерактивної консолі	+	+	+	+
Наявність зручного користувацького інтерфейсу	+	+	+	+
Швидкий та зручний пошук	+	+	+	+
Підсвічування синтаксису	+	+	+	+
Підтримка наукових пакетів	+	+	+	+
Наявність функцій статичного аналізу коду	+	+	+	+
Потребує малої кількості пам'яті для зберігання.	+	–	–	–

Рисунок Г.5 – Слайд № 5

Схема структури ПП

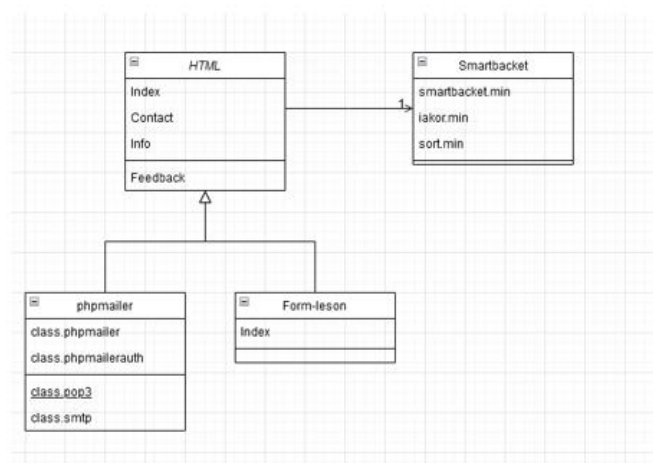


Рисунок Г.6 – Слайд № 6

Програмна структура

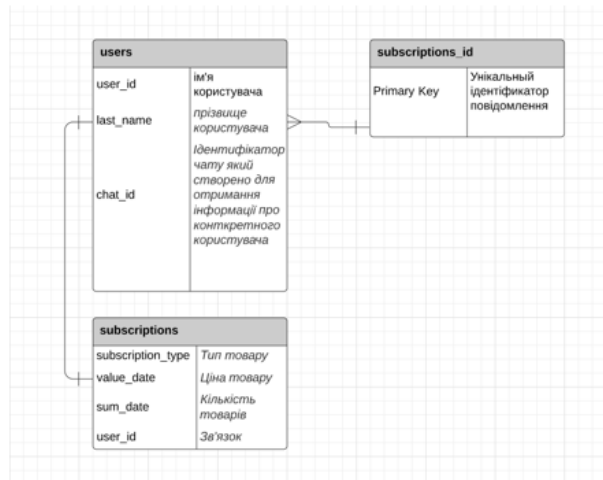


Рисунок Г.7 – Слайд № 7

Інтерфейсні вікна ПП

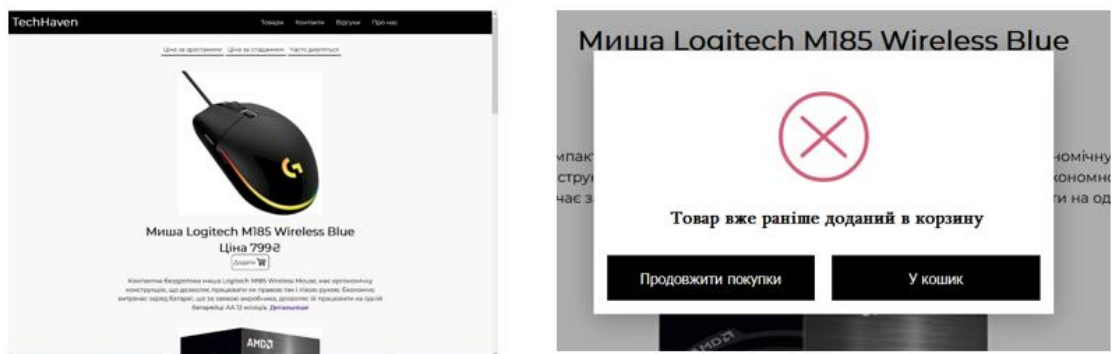


Рисунок Г.8 – Слайд № 8

Інтерфейсні вікна ПП

Товар	ID	Ціна / грн	Кількість настрої	Загальна ціна / грн	Виділити
 Acer Aspire 7	79	30999	- 1 +	30999	✖
 TP-Link Archer C64	83	1399	- 2 +	2798	✖
 AMD Ryzen 5	97	6299	- 1 +	6299	✖
 Logitech M185 Wireless Blue	35421	799	- 6 +	4794	✖

Загальна вартість: 0 Загальна вартість: 44830.00 грн

Введіть ім'я: Введіть телефон:

Введіть e-mail: ☒ Я приймаю умови користування сайтом

Ціна за зростанням Ціна за спаданням Часто дивляться

Ціна

Усього 799 грн



Рисунок Г.9 – Слайд № 9

Інтерфейсні вікна ПП

27 Nov 2023 21:44

Ваше ім'я:

Ярослав

Ваш коментар(вкажіть модель):

Я залишаю відгук про товар

Як нас знайти?

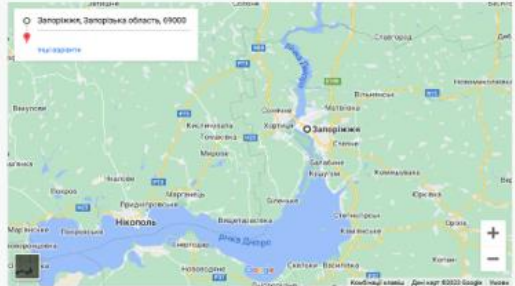


Рисунок Г.10 – Слайд № 10

Проведення експериментів

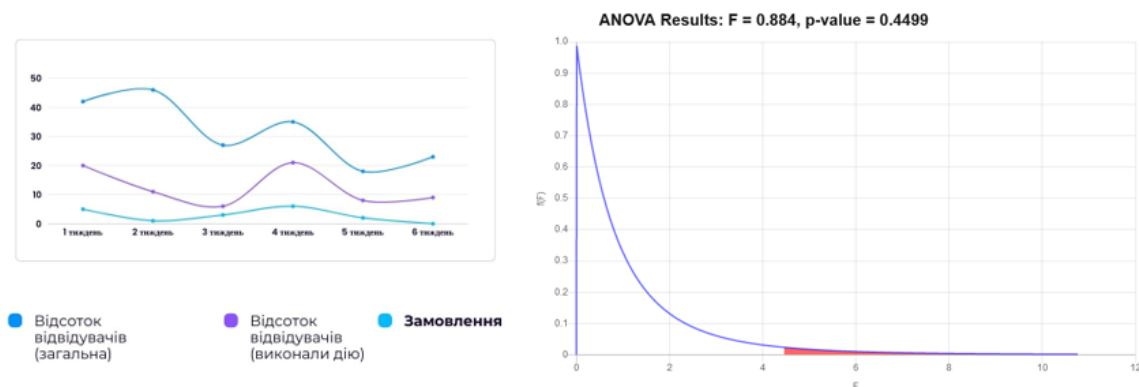


Рисунок Г.11 – Слайд № 11

Висновок

Під час роботи **проведено аналіз** предметної області та визначено ключові поняття, пов'язані з розробкою вебсайту для продажу комп'ютерної техніки. **Виявлено актуальність** та доцільність поставленого завдання, включаючи розгляд питань ефективності, взаємодії з користувачами та оптимізації процесів.

Проведено дослідження та порівняльний аналіз існуючих програмних засобів для веб-розробки та управління електронною комерцією, розглянуто основні методи реалізації функцій вебсайту, таких як замовлення товарів, зв'язок з розробником, завантаження інформації та обробка відгуків користувачів. **Розглянуто аспекти** валідації введених даних та візуалізації оброблених даних кошику.

Зазначено, що для оптимізації та покращення функціоналу сайту **використовуються методи** сортування товару за різними категоріями, такими як ціна, найвідвідуваніші товари тощо.

У рамках розробки вебсайту **враховано потреби** користувачів у зручному інтерфейсі, можливості легкого замовлення товарів та взаємодії з розробником через форму зворотного зв'язку

Рисунок Г.12 – Слайд № 12