

КВАЛІФІКАЦІЙНА РОБОТА

КР.ІПЗ – 21.00.00.000 ІЗ

Група ІПЗс-2017

Янків Р. С.

2021

**ЗАКЛАД ВИЩОЇ ОСВІТИ
УНІВЕРСИТЕТ КОРОЛЯ ДАНИЛА**

**Факультет суспільних і прикладних наук
Кафедра інформаційних технологій**

на правах рукопису

Янків Руслан Степанович

УДК 004.415

WEB-застосування підтримки діяльності магазину з продажу електроніки

Спеціальність 121 — «Інженерія програмного забезпечення»

Кваліфікаційна робота на здобуття освітнього ступеня бакалавра

Науковий Керівник

к.т.н., доц.

Ващишак Сергій Петрович

Івано-Франківськ — 2021

**ЗВО «Університет Короля Данила»
Факультет суспільних і прикладних наук
Кафедра інформаційних технологій**

Освітній ступінь: «бакалавр»

Спеціальність: 121 «Інженерія програмного забезпечення»

**ЗАТВЕРДЖУЮ
Завідувач кафедри**

« _____ » _____ 202__ року

**З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**

Янківу Руслану Степановичу

(прізвище, ім'я, по-батькові)

1. Тема роботи

Web-застосування підтримки діяльності магазину з продажу електроніки

керівник роботи

Ващишак Сергій Петрович, кандидат технічних наук, доцент

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

затверджені наказом вищого навчального закладу від «__» _____ 202__ року
№ _____

2. Строк подання студентом роботи

01.06.2021 р.

3. Зміст бакалаврської роботи (перелік питань, які потрібно розробити)

1. Опис аналогів та використаних технологій

2. Проектування та розробка Web-застосунку

3. Реалізація та огляд платформи

4. Охорона праці

4. Дата видачі завдання

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Основні тенденції розробки інтернет-аплікацій	25.03.2021	
2.	Проектування та розробка застосунку	15.04.2021	
3.	Програмна реалізація застосунку	15.05.2021	
4.	Формування висновків	23.05.2021	
5.	Охорона праці	25.05.2021	
6.	Оформлення пояснювальної записки	29.05.2021	
7.	Оформлення графічного матеріалу та підготовка до захисту роботи	01.06.2021	

Студент _____
(підпис) (прізвище та ініціали)

Керівник роботи _____
(підпис) (прізвище та ініціали)

Вихідні дані:

Мова програмування JavaScript, Node.js, Vue.js

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

Сторінка	Опис граф. матеріалу
12	Головна сторінка сайту IT-SKLAD.com
13	Головна сторінка сайту STOREinUA
14	Головна сторінка сайту Mobileplanet
16	Функції магазину
19	Електронна комерція
26	Односторонні потоки даних
27	Backend API
33	Use Case діаграма
36	Головна сторінка сайту.
37	Вхід користувача.
	Заповнені поля входу.
38	Товар у корзині.
	Декілька товарів у корзині.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ.....	7
ВСТУП.....	8
РОЗДІЛ 1. ОПИС АНАЛОГІВ ТА ВИКОРИСТАНИХ ТЕХНОЛОГІЙ.....	12
1.1 Огляд існуючих аналогів.....	12
1.2 Вибір засобів реалізації завдання.....	16
1.3 Поставлення завдання на кваліфікаційну роботу.....	25
Висновки до розділу 1	25
РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА WEB-ЗАСТОСУНКУ	27
2.1 Огляд використаних технологій.....	27
2.2 Короткий опис системи модулів.....	31
2.3 Розробка основних функцій платформи.....	39
Висновки до розділу 2	41
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ОГЛЯД ПЛАТФОРМИ	42
3.1 Розробка з модулем користувача.....	42
3.2 Робота з модулем продукту.....	45
3.3 Створення необхідних компонентів.....	46
Висновки до розділу 3	53
РОЗДІЛ 4. ОХОРОНА ПРАЦІ	55
4.1 Умови роботи на робочому місці з комп'ютером	55
4.2 Забезпечення правильного мікроклімату у приміщенні	57
4.3 Вимоги до безпечної експлуатації комп'ютерної техніки.....	61
Висновки до розділу 4	61
ВИСНОВКИ.....	62
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	63

					КР.ІІЗ – 21.00.000 ІЗ			
<i>Зм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>	WEB-застосування підтримки діяльності магазину з продажу електроніки. Пояснювальна записка	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушів</i>
Розроб.	Янків Р.С.					6	63	
Перевір.	Ващишак С.П.							
Реценз.	Гуменюк Т.В							
Н. Контр.	Зорін В.О.							
Затверд.	Пашкевич О.П.				ЗВО «УКД» ІІЗс-2017			

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

- CSS — Cascading Style Sheets;
- HTML — HyperText Markup Language;
- JS — JavaScript;
- Vue.js — JavaScript-ф реймворк що використовує шаблон MVVM;
- API — прикладний програмний інтерфейс (Application Programming Interface).

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.А
Змн.	Арк.А	№ докум.№	Підп.Під	Дата		7

ВСТУП

Актуальність теми. За останній час відбулися значні зміни у способі робити покупки. Хоча споживачі продовжують купувати в фізичному магазині, але тепер покупцям дуже зручно робити покупки в Інтернеті, оскільки це звільняє клієнта від особистого відвідування магазину. Інтернет покупки мають свої значні переваги, і це зменшує зусилля на подорож до фізичного магазину. Рішення можна приймати вдома, спокійно дивлячись на різні товари з інтернет магазину.

Покупки в інтернеті дозволяють покупцям безпосередньо купувати товари чи послуги у продавця через Інтернет, в цьому їм допомагає веб браузер або мобільний додаток. Споживачі знаходять товар, що їх цікавить, безпосередньо відвідуючи веб сайт продавця або здійснюючи пошук серед альтернативних продавців за допомогою пошукової системи покупок, яка відображає доступність і ціни одного і того ж товару в різних електронних магазинах. Починаючи з 2020 року, клієнти можуть робити покупки в Інтернеті, використовуючи різноманітні комп'ютери та пристрої, включаючи настільні комп'ютери, ноутбуки, планшетні комп'ютери та смартфони.

Інтернет-магазин наводить фізичну аналогію купівлі продуктів чи послуг у звичайному роздрібному магазині чи торговому центрі «побутова техніка чи послуга для дому (меблі)», процес називається Інтернет-магазином від бізнесу до споживача (B2C). Коли Інтернет-магазин створений для того, щоб дозволити компаніям купувати товари у іншого бізнесу, цей процес називається Інтернет-магазином від бізнесу до бізнесу (B2B). Типовий Інтернет магазин дозволяє клієнту переглядати асортимент товарів та послуг фірми, переглядати фотографії чи зображення товарів, а також інформацію про технічні характеристики товару, особливості та ціни.

Інтернет-магазини зазвичай дозволяють покупцям використовувати функції «пошуку» для пошуку конкретних моделей, брендів або товарів.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Інтернет-клієнти повинні мати доступ до Інтернету та дійсний спосіб оплати, щоб здійснити транзакцію, таку як кредитна картка, дебетова картка, що підтримує Interac, або така послуга, як PayPal. Електронний магазин продає товари замовнику, для цифрових продуктів, таких як цифрові аудіо файли з пісень або програмного забезпечення, електронна Tailer зазвичай відправляє файл клієнту через Інтернет. Найбільшою з цих корпорацій, що займається роздрібною торгівлею в Інтернеті, є Alibaba, Amazon.com та eBay.

Поява Інтернет магазинів, в сьогоднішній час, дуже широко розвинулась із появою Інтернету. Спочатку ця платформа працювала тільки як рекламний інструмент для компаній, надаючи інформацію про свою продукцію. Це швидко перейшло від цієї простої функції до фактичної транзакції покупок в Інтернеті завдяки розробці інтерактивних веб сторінок та безпечних передач. Електронна комерція приносить зручність для клієнтів, оскільки їм не потрібно виходити з дому, а для того щоб щось замовити потрібно лише переглядати веб сайти в Інтернеті. Це дуже допомагає покупцям заощадити свій час та придбати ширший асортимент товарів. Споживачі також отримують силу завдяки Інтернет-магазинам. Покупець також може досліджувати продукцію та порівнювати ціни серед роздрібних торговців.

Мета роботи — розробка онлайн застосунку для продажу електронних товарів із простим функціями які дозволяють легко та швидко здійснити покупку.

Об'єкт роботи — процес продажу електронних товарів за допомогою інтернет магазину.

Предмет роботи — покращення функціоналу сайту.

Завдання роботи. Відповідно до теми в роботі поставлені такі задачі:

- проаналізувати недоліки існуючих сайтів-аналогів;
- вибрати технології та мову програмування;
- розробити зручний дизайн, по якому користувачі будуть інтуїтивно пересуватись;
- забезпечити стабільність роботи сайту.

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Методи роботи. При вирішенні поставленого завдання були використані такі технології: Vue.js, Vuex, Axios.

Результати роботи. Результатом виконання кваліфікаційної роботи є розроблений інтернет магазин з продажу електроніки.

Структура роботи. Розділи — 4. Загальний обсяг частини — 55. Список використаних джерел — 15 позиції.

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

РОЗДІЛ 1. ОПИС АНАЛОГІВ ТА ВИКОРИСТАНИХ ТЕХНОЛОГІЙ

1.1 Огляд та опис існуючих аналогів

На сьогодні в мережі Інтернет існує певна кількість аналогічних web-сайтів з продажу електроніки, якими володіють також не малі компанії. Зручність використання таких сайтів полягає у тому, що можна знаходячись в будь-якому місці, купити потрібну тобі річ. «it-sklad.com.ua», «STOREinUA», «Mobileplanet». На основі наведених вище сайтів проведемо порівняльну характеристику. Розглянемо дані аналоги, визначимо їх переваги та недоліки.

IT-SKLAD.com.ua сайт із електронними товарами для офісу. IT-SKLAD.com є дешевим сайтом для придбання електронної техніки для офісу, ремонт та обслуговування комп'ютерної техніки; серверні апаратні та програмні рішення; створення та обслуговування локальних мереж. Містить розділи «для покупців», а також «мій обліковий запис». Вигляд головної сторінки сайту IT-SKLAD.com.ua наведено на рисунку 1.1, у розділі «для покупців» є категорії:

- товари зі знижками;
- нові поступлення;
- лідери продажів;
- наші магазини;
- написати нам;
- мапа сайту.

Переваги сайту IT-SKLAD.com:

- наявність розділу «Низькі ціни кожного дня»;
- офіційна гарантія;
- можливість створити особистий ПК для себе.

Недоліки:

- незручний інтерфейс сайту;

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

- немає чат бота який зміг би відповісти на твої питання;
- щоб додати товар до кошику потрібна реєстрація.

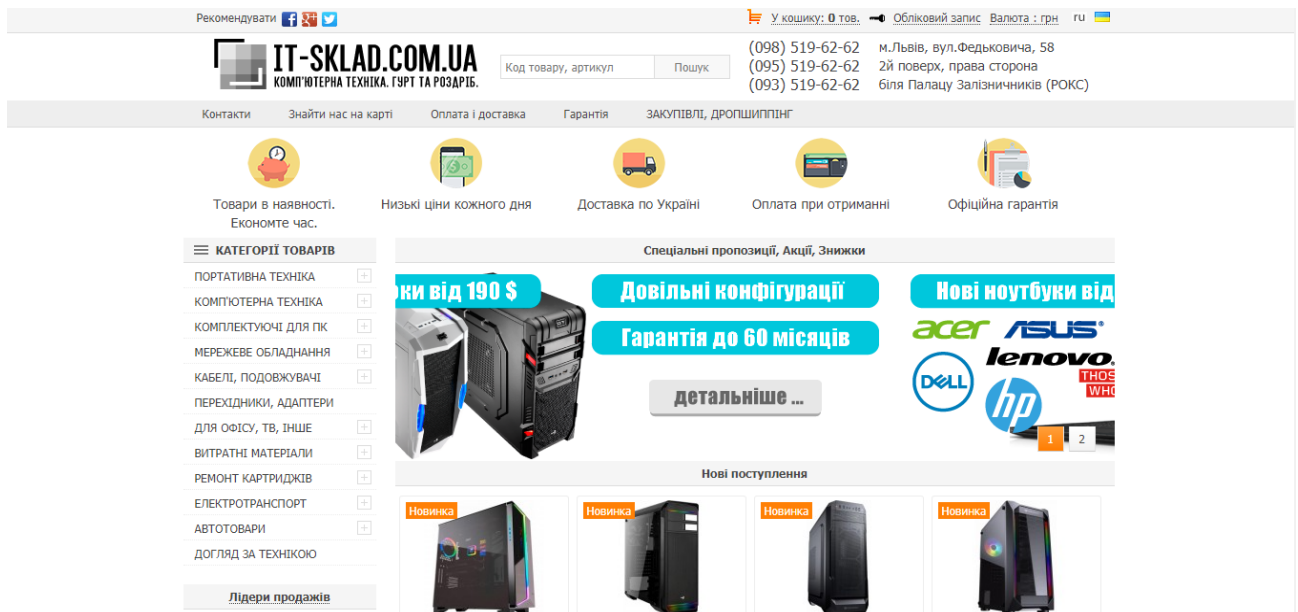


Рисунок 1.1 — Головна сторінка сайту IT-SKLAD.com

Сайт STOREinUA. Здійснює продаж товарів по всій території України. В електронному магазині приймаються всі види оплати. На весь товар надається гарантія. Можна придбати техніку та аксесуари.

На сьогоднішній день здійснює роздрібну продаж техніки ведучих світових виробників: Samsung, та Apple.

Оформлення замовлення. Якщо вас зацікавив певний товар розміщений в інтернет-магазині необхідно оформити замовлення на придбання товару. Процедура замовлення та отримання товару виглядає наступним чином. Замовлення на сайті здійснюється у такій послідовності:

- вибираєте товар та оформляєте замовлення вказавши своє ім'я, поштову та електронну адреси;
- перед остаточним підтвердженням замовлення перевірте вказану інформацію, телефон та електронну адресу для зв'язку;
- після цього, в підтвердження вище переліченого необхідно підтвердити замовлення натиснувши кнопку в кінці сторінки листа.

									Арк.
Змн.	Арк.	№ докум.	Підп.	Дата					

КР.ІПЗ – 21.00.00.000 ІЗ

Реєстрація з метою спрощення процесу оформлення замовлення в Інтернет-магазині, пропонується зареєструватися або авторизуватися ввівши свій логін і пароль, якщо ви вже є зареєстрованим користувачем сайту STOREinUA. Вигляд головної сторінки сайту STOREinUA наведено на рисунку

1.2. Переваги сайту STOREinUA:

- можливість брати товари в кредит, при необхідності повертати товар або зробити обмін;
- можливість задавати питання чат боту 24/7;
- функція яка дозволяє порівняти товар по цінам та характеристикам.

Недоліки:

- при фільтрації товару працює зі збоями;
- не зручний інтерфейс;
- часто з'являється великий обсяг реклами.

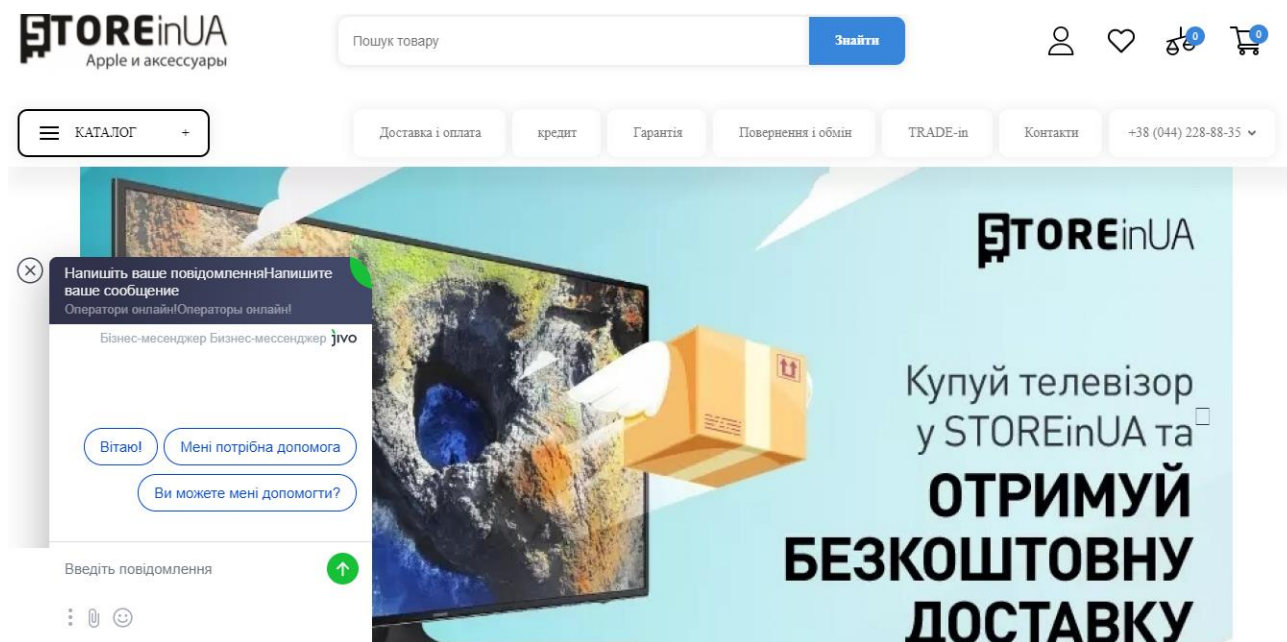


Рисунок 1.2 — Головна сторінка сайту STOREinUA

Сайт Mobileplanet. Мобільні телефони, комп'ютерна техніка, аудіо відео фото техніка, аксесуари та побутова техніка. Гарантує хорошу якість і доступні ціни для покупців.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Варіанти фільтрування можна вибрати автоматично або власноручно. У описі кожної конкретної моделі товару вказаний термін гарантійного обслуговування, яке забезпечується магазином або ж безпосередньо сервісними центрами виробників. Вигляд головної сторінки сайту Mobileplanet наведено на рисунку 1.3. Нижче перерахований список послуг з сервісного обслуговування, які надає «Mobileplanet»:

- гарантійне обслуговування, ремонт електротехніки;
- проектування, прокладка і модернізація локальних мереж;
- ремонт моніторів всіх видів;

Переваги сайту Mobileplanet:

- лояльні ціни;
- тільки кращі моделі цифрової техніки;
- зручний інтуїтивно зрозумілий онлайн каталог;
- професійна допомога у виборі товару і якісний сервіс;
- широкий асортимент пристроїв.

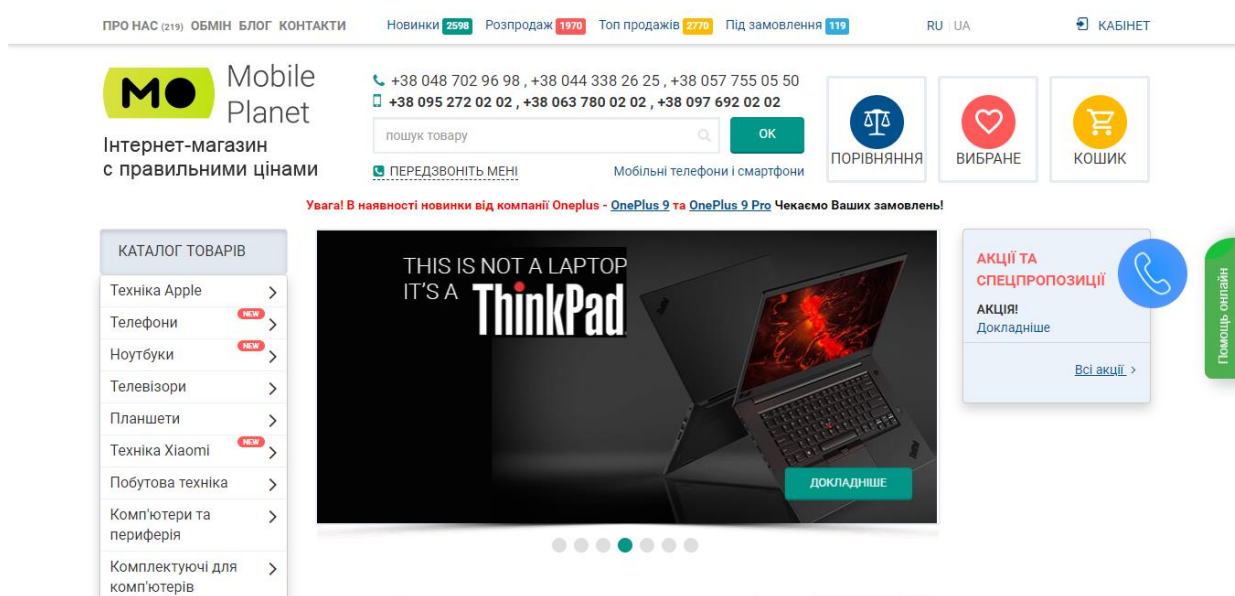


Рисунок 1.3 — Головна сторінка сайту Mobileplanet

Недоліки:

- при фільтрації товару працює зі збоями;

					КР.ПЗ – 21.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

- не зручний інтерфейс;
- часто з'являється великий обсяг реклами.

1.2 Вибір засобів реалізації завдання

Необхідна гіпертекстова структура HTML для управління матеріалом. Для реалізації представлення матеріалу необхідна використання HTML та CSS. Саме ця комбінація засобів дозволяє забезпечити підтримку внутрішнього вебсайта.

Для створення web-сервісу надання послуг з продажу електроніки було використано наступні програмні засоби:

- мова гіпертекстової розмітки HTML5;
- каскадні таблиці стилів CSS;
- Vue.js — JavaScript-фреймворк, що використовує шаблони.

Існує кілька систем класифікації Internet-магазинів, за методом роздрібного продажу товарів у мережі:

- Internet-магазини, Web-вітрини, торгові системи, торгові ряди, контентні проекти (споживацькі енциклопедії, системи Internet-замовлень товарів тощо);
- за бізнес моделлю, повністю онлайн-ових магазинів та суміщення оффлайн-ового бізнесу з онлайн-овим, (коли Internet-магазин створюється на основі вже діючої реальної торгової структури);
- за взаємовідносинами з постачальниками: магазини, які володіють власним складом (наявність реальних товарних запасів), магазини що працюють за договорами з постачальниками (відсутність значних товарних запасів);
- за ступенем автоматизації серед торгових систем електронних магазинів розрізняють Web-вітрини, власне Internet-магазини та торгові Internet-системи (TIS).

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Web-вітрина товарного каталогу, або системи навігації та оформлення замовлення з наступною передачею його менеджеру для оформлення. Той у свою чергу зв'язується зі складом, організовує доставку товару покупцеві, контролює процес оплати за товар. Паралельно ведеться рекламна робота, вивчення попиту, аналітична робота.

Робота Web-вітрини багато в чому нагадує посилкову торгівлю, проте замовлення на товар та ознайомлення з ним робиться через мережу Internet. В даному випадку Web-вітрина є інструментом залучення покупця, інтерфейсом для взаємодії з ним та проведення маркетингових заходів, на рисунку 1.4, зображено функції магазину.



Рисунок 1.4 — Функції магазину

Характерною рисою Internet-магазину є повна автоматизація системи обробки замовлень, завдяки чому можна працювати індивідуально з кожним зареєстрованим клієнтом.

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Спільною рисою для та Internet-магазину та TIS є можливість здійснювати повний торговий цикл у режимі підключення до мережі. При цьому TIS додатково інтегрована в систему внутрішнього документообігу компанії. Неавтоматизованими для Internet-магазину та TIS залишаються системи доставки товару.

Пошук товарів в Internet-магазині може здійснюватися за допомогою каталогу або через внутрішню пошукову систему.

Каталог товарів повинен містити як найповнішу інформацію про товар, мати зручну структуру, пошук необхідного товару повинен займати якнайменше часу. Саме за допомогою каталогу можна здійснити огляд товару (найчастіше у вигляді фото), ознайомитися з його споживчими та технічними характеристиками (у вигляді тексту та спеціальних символів), ціною. Важливу роль в даному випадку можуть відігравати технології 3D (технології тривимірного зображення), завдяки яким товар можна оглянути з усіх боків, відкрити кришку тощо. Проте використання таких технологій висуває додаткові вимоги до технічних можливостей комп'ютера клієнта.

Інформаційна підтримка потенційного покупця полягає в наданні йому в будь-який момент відповіді на питання, що виникають при здійсненні покупки. Найчастіше така інформація стосується умов після продажного сервісу, знижок на певні товари, особливостей схем оплати і т.п. У багатьох випадках на сайтах електронних магазинів існують спеціальні сторінки, де подані відповіді на найважливіші питання.

У процесі перегляду і відбору товару покупець формує власний віртуальний кошик. Кошик покупця являє собою список відібраного товару з вказанням його ціни, кількості та загальної вартості (з урахуванням можливих знижок). Такий список постійно доступний покупцеві. За його бажанням у будь-який момент будь-який товар може бути вилучений з кошика з відповідним подальшим перерахуванням вартості, або може відбутися повне очищення кошика.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Коли зроблено остаточний вибір товару, покупцеві слід підтвердити замовлення, зареєструватися за встановленою процедурою з визначенням форми оплати та доставки товару.

Загалом типова схема взаємодії покупця з Internet-магазином здійснюється наступним чином:

- покупець за допомогою браузера заходить на сайт Internet-магазину, який містить електронну вітрину, де представлений каталог товарів та необхідні елементи інтерфейсу для виконання операцій відбору та купівлі товарів;
- перегляд товарного каталогу та вибір товарів (формування кошика покупця);
- реєстрація покупця;
- вибір форми оплати та доставки товару;
- підтвердження замовлення;
- оплата товару;
- доставка придбаного товару покупцеві.

Реєстрація полягає у заповненні покупцем спеціальної форми, що включає в себе інформацію про покупця, його поштову або електронну адресу, особистий пароль та деяку іншу. Процедура реєстрації дозволяє Internet магазину убезпечити себе від можливих шахрайств та полегшити процедуру покупки для покупця наступного разу.

Реєстрація може здійснюватися до і після вибору товару. В першому випадку створюється спеціальний вхід для постійних клієнтів, для яких реалізується спеціальна система обслуговування та оплати. Реєстрація після вибору товару дозволяє покупцю зберегти анонімність відвідування магазину та заощаджує час, якщо покупець нічого не вибрав.

Під час проведення реєстрації особиста інформація покупця забезпечується шляхом передачі даних з використанням спеціальних методів захисту. Такими засобами можуть виступати протоколи SET або SSL.

Обробка замовлення покупця здійснюється безпосередньо торговою

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

системою Internet-магазину і починається з перевірки наявності товарів на складі та його резервування. Якщо певний товар в даний момент відсутній, то система інформує покупця про можливу затримку виконання замовлення. Пізніше, якщо здійснюється оплата через Internet, виконується запит до визначеної платіжної системи та оформлення замовлення на доставку товару. Покупець в цей час може отримувати інформацію про проходження замовлення, на рисунку 1.5, зображено як працюють магазини в бізнесі.



Рисунок 1.5 — Електронна комерція

Оплата покупцем придбаного товару в Internet-магазинах може здійснюватися шляхом передоплати та при одержанні товару.

До варіанту передоплати можна віднести оплату при передачі інформації через Internet, при доставці звичайної (нової пошти), чи експрес пошти, магістральним транспортом, за допомогою захищених угод (схем, пов'язаних із резервуванням суми покупки на рахунок покупця з дійсним переказом коштів після здійснення доставки).

					КР.ПЗ – 21.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Оплата при одержанні товару здійснюється в таких формах як оплата кур'єрові чи під час візиту в реальний магазин, післяплатою в момент отримання товару на пошті.

Недоліком такої форми є те, що масштаб діяльності кур'єрської служби, що належить Internet-магазину є досить обмеженою (в місті, де працює склад Internet-магазину або у кількох інших великих містах). Кур'єрові також незручно оплачувати інформацію, що реалізується через Internet та надання деяких послуг:

- за допомогою платіжних систем мережі Internet. Така форма оплати є найбільш прогресивною. Її характеристика буде подана далі (тема б);
- оплата поштовим (телеграфним) переказом. Така форма оплати займає досить багато часу (поштовим переказом до 6 тижнів, телеграфним до 1 тижня) і потребує додаткових коштів для оплати поштових послуг (10% вартості покупки);
- післяплата при одержанні товару на пошті також потребує оплати поштових послуг (до 20% вартості покупки). Тут значно продовжується час здійснення угоди, а відповідно, і ризик. Крім того, в цьому випадку значно ускладнена процедура повернення товару або грошей;
- оплата банківським переказом серед безготівкових форм є найбільш безпечним, гроші надходять за порівняно невеликий термін (до 1 тижня), оплата за послуги невелика (1 — 5%).

У комерційній практиці аукціон є формою продажу товарів або послуг на публічних конкурентних торгах, в процесі яких встановлюється їх кінцева ціна. Віртуальний (електронний) аукціон (Internet-аукціон) проводиться за тим же принципом в мережі Internet за допомогою спеціального програмного забезпечення, яке встановлюється на сайті організатора торгу.

Аукціони мають спосіб продажу товарів, який базується на цінній конкуренції між покупцями. Він є процедурою встановлення рівноважних цін

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

на товар, при цьому ціну встановлює покупець, а продавець правила, за якими має проводитися аукціон.

Характерними рисами Internet-аукціонів є:

- можливість залучення великої кількості продавців і покупців (значна концентрація попиту і пропозиції);
- цілодобова інтерактивна взаємодія продавців і покупців;
- можливість надання деталізованого опису і зображення товарів.
- Internet аукціони дають можливість не тільки реалізувати товари та послуги через Internet, але й здійснювати тестові продажі, визначати обсяг початкового попиту та ринкову ціну для нових продуктів.

Електронні торговельні майданчики такі сайти належать до категорії B2B, на яких укладаються угоди між продавцями та покупцями та здійснюється проведення фінансово торгових трансакцій.

На таких сайтах зібрана інформація про ціни на продукцію у всіх зареєстрованих виробників і постачальників, умови її оплати і доставки. Також існує можливість пошуку та сортування даних за різними параметрами (наприклад, за видом продукції, за певним ціновим діапазоном тощо), і часто можливість безпосереднього здійснення замовлення та його оплати за допомогою системи електронних платежів.

В більшості випадків на сайті подаються дані про потенційних покупців певної продукції, інформаційно аналітичні ресурси та новини щодо даного сегмента ринку.

Основні принципи роботи електронних торговельних майданчиків, простота і зручність рівний та простий доступ всіх учасників до інформації та послуг, що полягає в створенні зрозумілого інтерфейсу, застосуванні простого процесу реєстрації відвідувачів.

Гнучкість управління каталогами додавання нових каталогів чи учасників, заміна старої інформації не повинні впливати на функціонування системи в цілому.

Інтеграція діяльності полягає у підтримці усіх аспектів електронної

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

комерції від виконання трансакцій до підтримки мережі постачання, що дає змогу спростити документообіг та збільшити вигоду учасників.

Крім того, має бути забезпечена організація взаємодії з торговими процесами і системами учасників торговельного майданчика, яка сприятиме фірмам в оптимізації процесів планування, прогнозування, управління ресурсами, одержанні інформації з зовнішніх джерел в реальному часі.

Адміністрування полягає в регулюванні роботи та взаємодії додатків сайту, які розроблених в різних середовищах.

Коли таких додатків небагато, керівництво користувача і контроль доступу вбудовані в кожен додаток окремо. При значній кількості додатків адміністрування має велике значення для організації роботи всього сайту.

Бізнес аналіз на сайтах даного типу для ефективно організації комерційних зв'язків слід не лише подавати певну інформацію, а й проводити її аналіз, здійснювати моніторинг ділової активності на торговельному майданчику, складати різноманітні звіти, що дозволяє учасникам об'єктивно і своєчасно аналізувати ситуацію на ринку, виявляти існуючі та прогнозні тенденції.

Надання додаткових послуг здійснюється для залучення додаткових учасників та одержання додаткового прибутку.

З цією метою доцільно здійснювати підписку на ряд послуг, що могли б цікавити окремих учасників. До такого переліку можуть включатися організація логістики, аукціони, фінансові послуги і т.д.

Забезпечення безпеки розробники проекту та учасники комерційних процесів повинні бути впевнені в безпеці проведенні платежів та передачі інформації.

Основні джерела доходів електронних торговельних майданчиків формуються на таких рівнях:

- операційному (доходи від трансакцій, передплатних внесків, надання аукціонних послуг);
- функціональному (доходи від реклами, маркетингових досліджень,

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

публікацій у каталогах);

- організаційно-технологічному (доходи від ліцензування та сертифікації товарів і послуг, страхування ризиків при здійсненні електронних операцій, організації безпеки проведення фінансових транзакцій);
- у колаборативному електронному бізнесі (доходи від оптимізації бізнес-процесів у ланцюжках доданої вартості).

1.3 Поставлення завдання на дипломне проектування

Створення програми електронної комерції було розроблено за допомогою Vue.js. Web-програми електронної комерції пропонують цілий ряд нових можливостей для бізнесу, це допомагає компаніям зменшити витрати і може мати справу із меншими накладними витратами та меншою кількістю ризиків.

Більша частина оптимізації продуктивності веб додатків для електронної комерції пов'язана з інтерфейсом, отже використання видатного фрейм ворку, наприклад, фронтенд центрик, який часто віддають перевагу за його прямолінійність. Прогресивна, поступово прийнятна структура JavaScript для побудови інтерфейсу в Інтернеті. Просте, мінімальне ядро з поступово застосовуваним стеком, який може обробляти програми будь-якого масштабу. Vue розроблений з нуля, щоб його можна було поступово застосовувати.

Для реалізації web-застосування підтримки діяльності магазину з продажу електроніки необхідно зробити наступне:

- проаналізувати переваги та недоліки існуючих сайтів-аналогів;
- для реалізації web-застосування вибрати технології та мову програмування;
- розробити зручний дизайн, по якому користувачі будуть інтуїтивно пересуватись, що сприятиме популяризації сайту;
- провести тестування готової програми;
- розробити заходи з охорони праці.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Висновки до розділу 1

В процесі роботи було розглянуто описано та проведено аналіз існуючих аналогів електронної комерції. Також провели дослідження і описали завдання на виконання роботи.

Під час створення програми функціонування нашого інтернет-магазину ми використовуватимемо Vuex для управління станом програми, нам також знадобиться Node.js для установки Vue, маршрутизатор Vue для навігації та Axios для отримання даних з API.

За допомогою Vue.js ми можемо писати той самий код JavaScript, що забезпечує однакову функціональність, набагато простішим способом та зручнішим для читання та розуміння.

У цьому електронному магазині ми створимо вебсайт електронної комерції за допомогою Vue.js з такими функціями:

- отримувати продукти з API;
- перелік товарів з API;
- деталі продукту;
- основне управління кошиком;
- основна автентифікація користувача;
- додати товари в кошик;
- сторінка оформлення замовлення.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

РОЗДІЛ 2. ПРОЕКТУВАННЯ ТА РОЗРОБКА WEB-ЗАСТОСУНКУ

2.1 Огляд використаних технологій

У центрі будь-якого Vuex додатку знаходиться сховище. Сховище в якому зберігається стан застосунку. Два моменти відрізняють сховище Vuex від простого глобального об'єкта.

Сховище Vuex реактивне, коли компоненти покладаються на його стан, то вони будуть реактивно і ефективно оновлюватися, якщо стан сховища змінюється.

Не можна безпосередньо змінювати стан сховища. Єдиний спосіб в нести зміни і викликати мутацію. Це гарантує, що будь-яка зміна стану залишає слід і дозволяє використовувати інструментарій, щоб краще розуміти хід роботи програми. Він також інтегрується з офіційним розширенням Vue devtools (відкриває нове вікно), що забезпечує розширені функції, такі як налагодження з нульовою конфігурацією в часі та експорт імпорт знімка стану.

Це автономний додаток із наступними частинами:

- State, який керує нашим додатком;
- відображення нашого State;
- дії, які може змінити користувач.

Однак простота швидко руйнується, коли ми маємо кілька компонентів, які мають спільний State:

- кілька переглядів можуть залежати від одного і того ж стану;
- дії які можуть потребувати зміни одного і того ж State.

Вирішуючи першу проблему, доведеться передавати одні й ті ж дані вхідними параметрами в глибоко вкладені компоненти. Що стосується другої проблеми, ми часто стикаємося з рішеннями, такими як звернення до прямих посилань батьківського дочірнього екземплярів.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

То чому б нам не витягти спільний стан із компонентів і не управляти ним глобально. Завдяки цьому наше State компонентів стає великим «видом», і будь-який компонент може отримати доступ до стану або активувати дії, незалежно від того, де вони знаходяться в дереві. На рисунку 2.1, представлено концепції «одностороннього потоку даних».

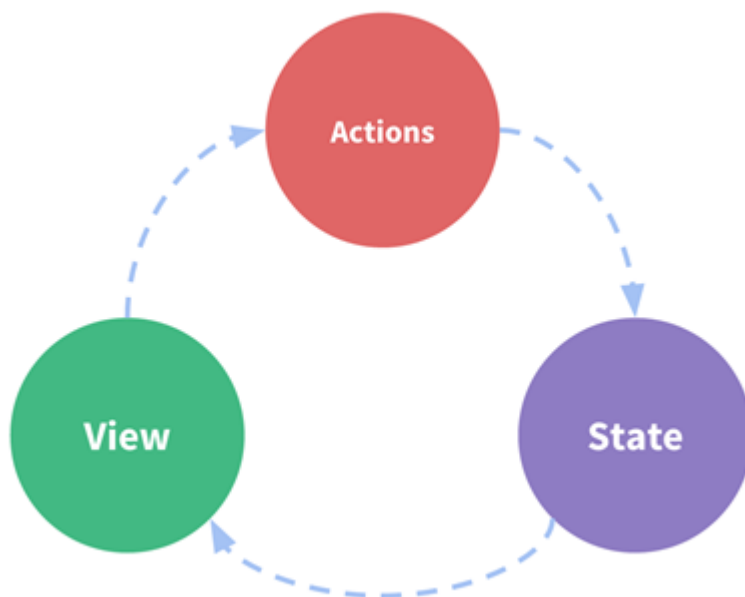


Рисунок 2.1 — Односторонні потоки даних

Визначаючи та відокремлюючи поняття, що стосуються управління державою та дотримання правил, що підтримують незалежність між поглядами та державами, ми надаємо нашому кодексу більше структури та ремонтпридатності.

Vueх допомагає нам вирішувати питання спільного управління державою, витрачаючи на більшу кількість концепцій та шаблонів. Це компроміс між короткостроковою та довгостроковою продуктивністю.

Vueх використовує єдиний State станів — тобто цей єдиний об'єкт містить усі стани вашого рівня програми і служить єдиним джерелом істини.

Це також означає, що у вас буде лише один магазин для кожної програми. Одне дерево держав дозволяє легко знаходити певний фрагмент стану і

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

дозволяє нам легко робити знімки поточного стану програми для налагодження.

Єдиний state станів не суперечить модульності, і у наступних розділах ми обговоримо, як розділити наш стан та мутації на під модулі. Дані, які ви зберігаєте у Vuex, дотримуються тих самих правил, що і data у екземплярі Vue, тобто об'єкт стану повинен бути простим. То як ми можемо відображати стан усередині магазину в наших компонентах Vue. Оскільки магазини Vuex реактивні, найпростіший спосіб "отримати" стан із нього просто повернення деякого стану магазину з обчислюваної властивості. На рисунку 2.2, зображено Backend API.

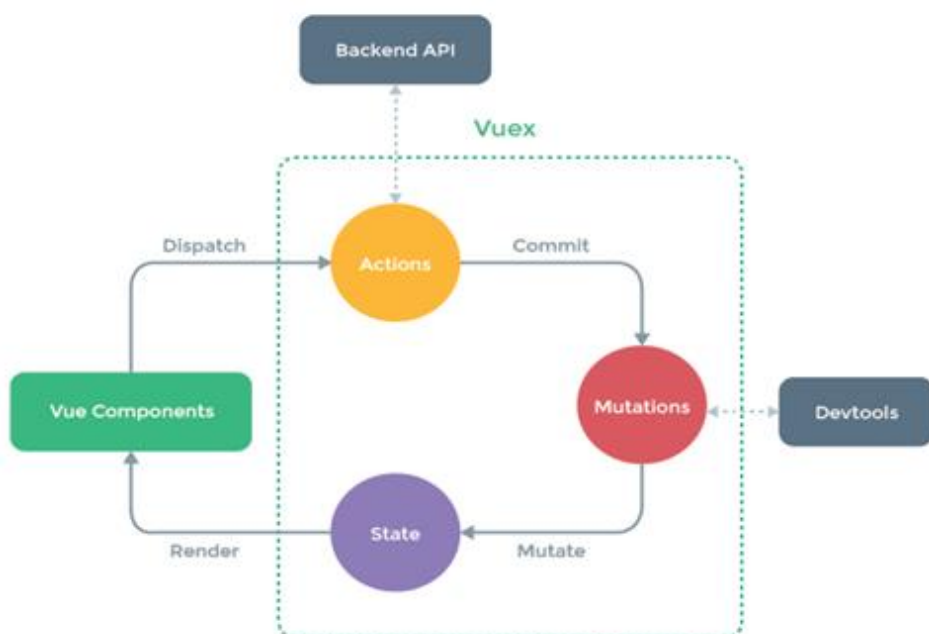


Рисунок 2.2 — Backend API

Що разу, коли `store.state.count` змінюється, це спричиняє повторну оцінку обчислюваної властивості та запускає відповідні оновлення DOM. Однак ця закономірність змушує компонент покладатися на глобальний магазин. При використанні модульної системи потрібно імпортувати сховище в кожен компонент, що використовує стан магазину, а також вимагає глузування під час тестування компонента. Vuex «вводить» магазин у всі дочірні компоненти від кореневого компонента через систему плагінів Vue, і буде доступний на них як

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

this.\$store. Коли компоненту потрібно використовувати декілька властивостей стану сховища або отримувачів, оголошення всіх цих обчислюваних властивостей може отримати повторювані та багатослівні. Для вирішення цього ми можемо скористатися `mapState` помічником, який генерує для нас обчислювані функції `getter`, економивши нам кілька натискань клавіш.

Ми також можемо передавати масив рядків, `mapState` коли ім'я відображеного обчислюваного властивості збігається з іменем під дерева стану. Зверніть увагу, що `mapState` повертає об'єкт. Як ми можемо використовувати його в поєднанні з іншими локальними обчислюваними властивостями. Зазвичай нам довелося б використовувати утиліту для об'єднання кількох об'єктів в один, щоб ми могли передати остаточний об'єкт `computed`. Однак за допомогою оператора розкидання об'єктів ми можемо значно спростити синтаксис.

Використання `Vuex` не означає, що ви повинні помістити весь стан у `Vuex`. Незважаючи на те, що введення більше стану у `Vuex` робить ваші мутації стану більш явними та налагоджувальними, іноді це також може зробити код більш багатослівним і непрямим.

Якщо частина держави суворо належить до однієї складової, може бути цілком нормально залишити її місцевою державою. Вам слід зважити компроміси та прийняти рішення, які відповідають потребам розвитку вашого додатка.

Єдиний спосіб фактично змінити стан у магазині. Зробити зміни у `Vuex` дуже схожі на події: кожна мутація має тип рядка та обробник. Функція обробника це місце, де ми виконуємо фактичні зміни стану, і вона отримує стан як перший аргумент. Ви не можете безпосередньо викликати обробник мутації. Подумайте про це більше як про реєстрацію події: «Коли `increment` спрацьовує мутація з типом, зателефонуйте цьому обробнику». Щоб викликати обробник мутації, вам потрібно зателефонувати `store.commitz` його типом.

Часто зустрічається закономірність використання констант для типів мутацій у різних реалізаціях Flux. Це дозволяє коду скористатися перевагами

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

таких інструментів, як літери, а розміщення всіх констант в одному файлі дозволяє вашим співавторам отримати короткий огляд можливих мутацій у всьому додатку. Ви можете зафіксувати мутації в компонентах за допомогою `this.$store.commit` або скористатися `mapMutations` помічником, який відображає методи компонентів у `store.commit` викликах (потрібно Store введення кореневої системи). Дії подібні до мутацій, різниця полягає в тому, що:

- замість того, щоб безпосередньо змінювати стан, дії ініціюють змінні;
- дії можуть використовуватися для асинхронних операцій.

Обробники дій отримують об'єкт контексту, який виставляє той самий набір методів властивостей на екземплярі сховища, тому ви можете зателефонувати `context.commit` для здійснення мутації або отримати доступ до стану та `getter` через `context.state` та `context.getters`. Ми навіть можемо називати інші дії з `context.dispatch`. Тому цей об'єкт контексту не є самим екземпляром магазину.

2.2 Короткий опис системи та модулів

Internet shop-web застосунок для продажу електронної техніки. Головна сторінка — основна інформація про продукт відразу буде перед очима покупця: назва, фотографія, короткий опис, ціна. Кабінет клієнта для того щоб придбати товар, клієнт повинен зареєструватись на сайті. Після успішної реєстрації клієнт може увійти до кабінету, (додавати товар у корзину, та оформляти доданий товар). Деталі продукту може бути повний опис певного продукту та його характеристика. (Опис певного товару, основні характеристики, додаткові, фізичні характеристики і інші).

Додати товари в «кошик» відкриваючи вкладку «кошик» покупець побачить інформацію про замовлення: перелік всіх товарів які він добавив у корзину, а також пропозицію продовжити покупки або ж одразу приступити до оформлення свого товару. Відгуки клієнтів чи поточна функція буде зберігати

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

відгуки від замовників, в туж чергу технічний працівник буде переглядати, та покращувати роботу сайту.

Сторінка оформлення замовлення в цьому полі ми побачимо що для оформлення товару нам потрібно ввести такі дані, (прізвище ім'я та по батькові одержувача, номер телефону та електронну адресу email. Далі замовнику буде представлено спосіб отримання товару, «само вивіз, відправлення новою поштою», у разі вибрання функції само вивозу буде відобразатись адреса «пункт видачі»). Нижче наведена таблиця 2.1, із описом функціоналу роботи сайту.

Таблиця 2.1
Опис функціоналу

Назва модуля	Назва функціоналу	Опис функціоналу
Головна сторінка	Основна інформація	Дає змогу переглянути покупцю наявний товар.
	Назва	Дає змогу переглянути назву, фотографію, опис та ціну наявного товару.
	Кабінет клієнта	Можливість додати, видалити товар до кошику, та в подальшому продовжувати покупки, також залишати коментарі (скарги та пропозиції).
	Опис	Короткий опис про товар.
	Ціна	Ціна наявного товару.

Діаграма є графічним зображенням можливих взаємодій користувача з системою. Діаграма випадків використання показує різні варіанти

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

використання та різні типи користувачів, які система має, і часто супроводжуватимуться також іншими типами діаграм.

Варіанти використання представлені кругами або еліпсами. Акторів часто зображують у вигляді фігурок. UML Class diagram яке використовується для візуалізації об'єктно орієнтованої системи. Класова діаграма відноситься за типом до статично структурних діаграм, які описують систему показуючи:

- класи;
- їхні атрибути;
- операції або методи;
- відношення між об'єктами.

Через їх спрощений характер, діаграми використання можуть бути хорошим інструментом комунікації для зацікавлених сторін. Креслення намагаються імітувати реальний світ і дають зацікавленій стороні уявлення про те, як буде розроблена система.

Було виявлено, що діаграми випадків використання передають намір системи більш спрощеним чином зацікавленим сторонам і що вони "інтерпретуються більш повно, ніж діаграми класів".

Метою діаграми використання є відображення динамічного аспекту системи. Вони забезпечують спрощене графічне представлення того, що система повинна робити у випадку використання. Подальші схеми та документація потрібні для повного функціонального та технічного погляду на систему.

UML сам по собі не є методом розробки однак він був розроблений, щоб бути сумісним з провідними об'єктно орієнтованими методами розробки програмного забезпечення свого часу, з яким був спочатку призначений для використання на початку роботи в Rational Software.

Важливо розрізняти модель UML від набору діаграм системи. Діаграми частково графічні зображення моделі системи.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Набір діаграм не повинен повністю охоплювати модель, а видалення діаграми не змінює модель. Модель може також містити документацію, яка керує елементами моделі та діаграмами.

Структурні діаграми представляють статичні аспекти системи. Він підкреслює речі, які повинні бути присутніми в системі, що моделюється. Оскільки структурні діаграми представляють структуру, вони широко використовуються для документування програмної архітектури програмних систем. Наприклад, схема компонентів описує розподіл програмної системи на компоненти та демонструє залежності між цими компонентами. На рисунку 2.3, зображена візуальна частина проекту.

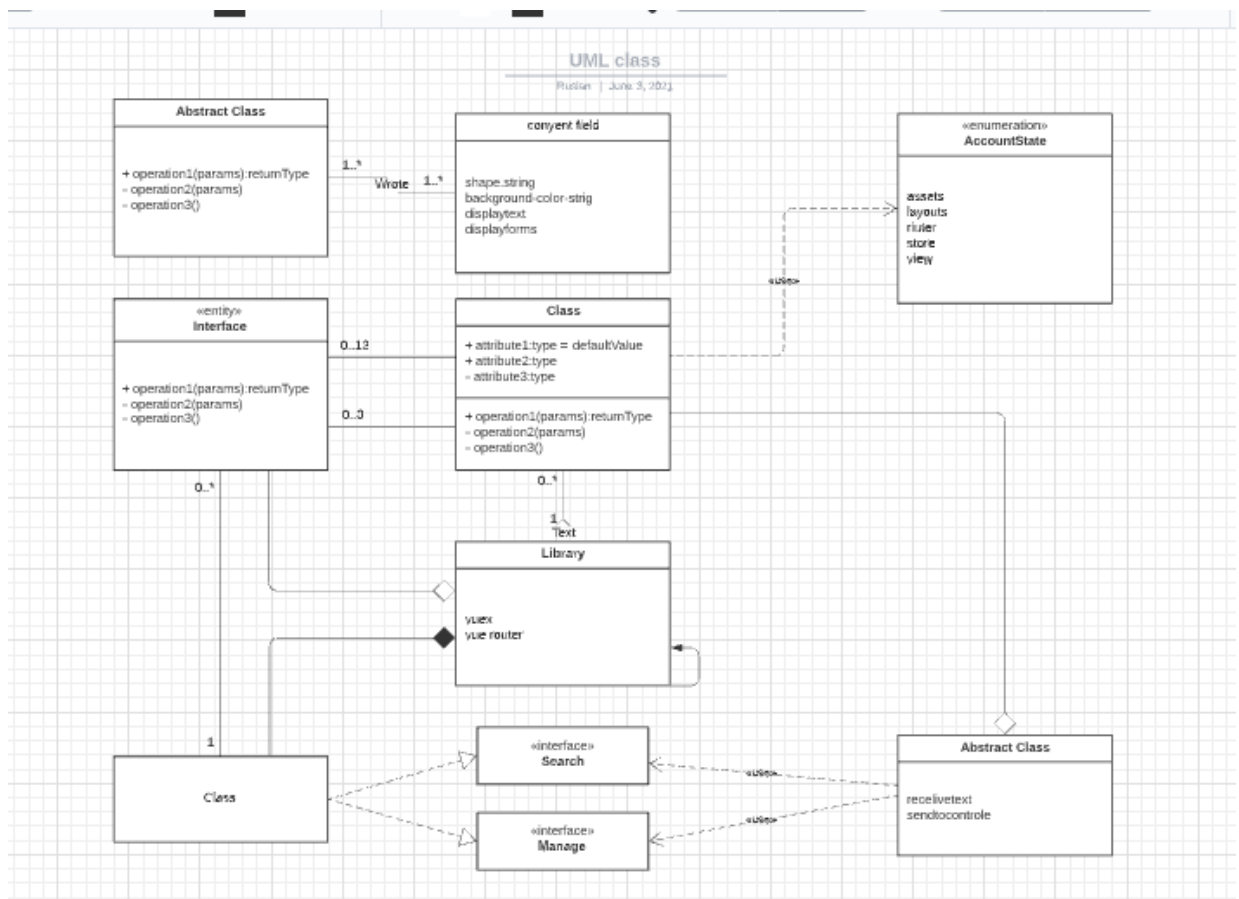


Рисунок 2.3 — UML діаграма

Use Case diagram форма вимог системи програми для створення нового програмного забезпечення. Use case показує очікувану поведінку (what) і метод який виконує цю поведінку (how). Ключовим концептом use case моделювання

					КР.ІПЗ – 21.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

це те, що воно допомагає нам спроектувати систему з ретроспективи кінцевого користувача. У розробці програмного забезпечення та системному проектуванні це опис поведінки системи, як вона відповідає на зовнішні запити. Іншими словами, різновид використання описує, «хто» і «що» може зробити з розглянутою системою. Методика різновидів використання застосовується для виявлення вимог до поведінки системи, відомих також як функціональні вимоги.

У системному проектуванні різновиди використання застосовуються на більш високому рівні ніж при розробці програмного забезпечення, часто представляючи цілі зацікавлених осіб або місії. На стадії аналізу вимог різновиди використання можуть бути перетворені на ряд детальних вимог і задокументовані за допомогою діаграм вимог SysML або інших подібних механізмів. графічним зображенням можливих взаємодій користувача з системою. Діаграма випадків використання показує різні варіанти використання та різні типи користувачів, які система має, і часто супроводжуватимуться також іншими типами діаграм. Варіанти використання представлені кругами або еліпсами. Акторів часто зображують у вигляді фігурок. На рисунку 2.4, зображено use case діаграму нашого сайту.

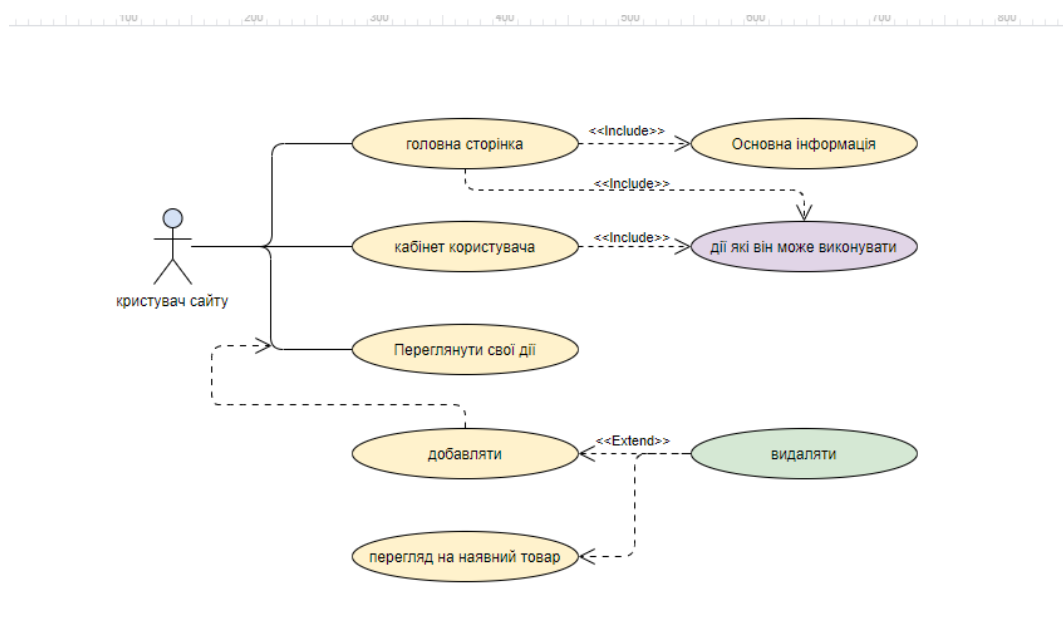


Рисунок 2.4 — Use Case діаграма

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Отже коротко розкажу що таке схема використання діаграми. UML діаграма прецедентів є основною формою системи вимоги до програмного забезпечення для нового програмного забезпечення слаборозвинених.

У випадках використання вказується очікувана поведінка що, а не точний спосіб її здійснення як. Описані випадки використання можуть бути позначені як текстовим, так і візуальним поданням, (тобто діаграмою випадків для використання).

Ключова концепція моделювання діаграм випадків використання полягає в тому, що це допомагає нам розробити систему з точки зору кінцевого користувача. Це ефективна техніка передачі поведінки системи з точки зору користувача шляхом визначення всієї видимої зовні системи поведінки.

Діаграма використання зазвичай проста. Він не відображає деталей випадків використання:

- він лише узагальнює деякі взаємозв'язки між випадками використання, суб'єктами та системами;
- він не відображає порядок, у якому виконуються кроки для досягнення цілей кожного випадку використання;

Як вже було сказано, схема використання може бути простою і містити лише кілька фігур.

Також зверніть увагу на те що, існує багато різних діаграм UML, які служать різним цілям (ми можемо бачити з дерева діаграм UML).

Мета моделювання виробництва полягають у наступному:

- визначення структури й робочих процесів організації, у якій буде використовуватися розроблювальна система;
- осмислення поточних проблем організації й шляхів поліпшення;
- забезпечення загального розуміння роботи організації замовниками й кінцевими користувачами;
- визначення вимог до системи, необхідних для підтримки;
- виробничих процесів організації.

										Арк.
Змн.	Арк.	№ докум.	Підп.	Дата	КР.ІПЗ – 21.00.00.000 ІЗ					

Варіанти використання представляють лише функціональні вимоги системи. Інші вимоги, такі як ділові правила, вимоги до якості обслуговування та обмеження впровадження, повинні бути представлені окремо знову ж таки, з іншими діаграмами UML.

Давайте розглянемо опис позначень які стосуються для діаграм:

- користувач;
- хтось взаємодіє з випадком використання (функція системи);
- названий іменником;
- користувач відіграє певну роль у бізнесі;
- схоже на поняття користувача, але користувач може грати різні ролі;
- наприклад: профіль. може бути інструктором, а також дослідником;
- грає дві ролі з двома системами;
- варіанти використання тригерів актора;
- актор несе відповідальність за систему, а користувач очікування від системи.

Варіанти використання поділяють різні типи відносин. Визначення взаємозв'язку між двома діаграмами використання є рішенням програмних аналітиків схеми випадків використання.

Взаємозв'язок між двома діаграмами використання в основному є моделюванням залежності між двома діаграмами використання.

Повторне використання існуючого діаграми використання за допомогою різних типів відносин зменшує загальні зусилля, необхідні для розробки системи.

Тепер перевіримо чи правильно все буде працювати за цією схемою, щоб побачити як ефективно застосувати варіант діаграми у своєму програмному проєкті:

- завжди структуруйте та впорядкуйте схему використання з точки зору акторів;

					КР.ІІЗ – 21.00.00.000 ІІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

- варіанти використання повинні починатися просто і з максимально можливого огляду;
- тільки тоді їх можна уточнити та детально описати далі;
- діаграми випадків використання засновані на функціональних можливостях і, отже, повинні зосереджуватися на «що», а не «як».

2.3 Розробка інтерфейсу платформи

На цьому етапі було продемонстровано як буде виглядати наша головна сторінка сайту з продажу електроніки (рис. 2.5).

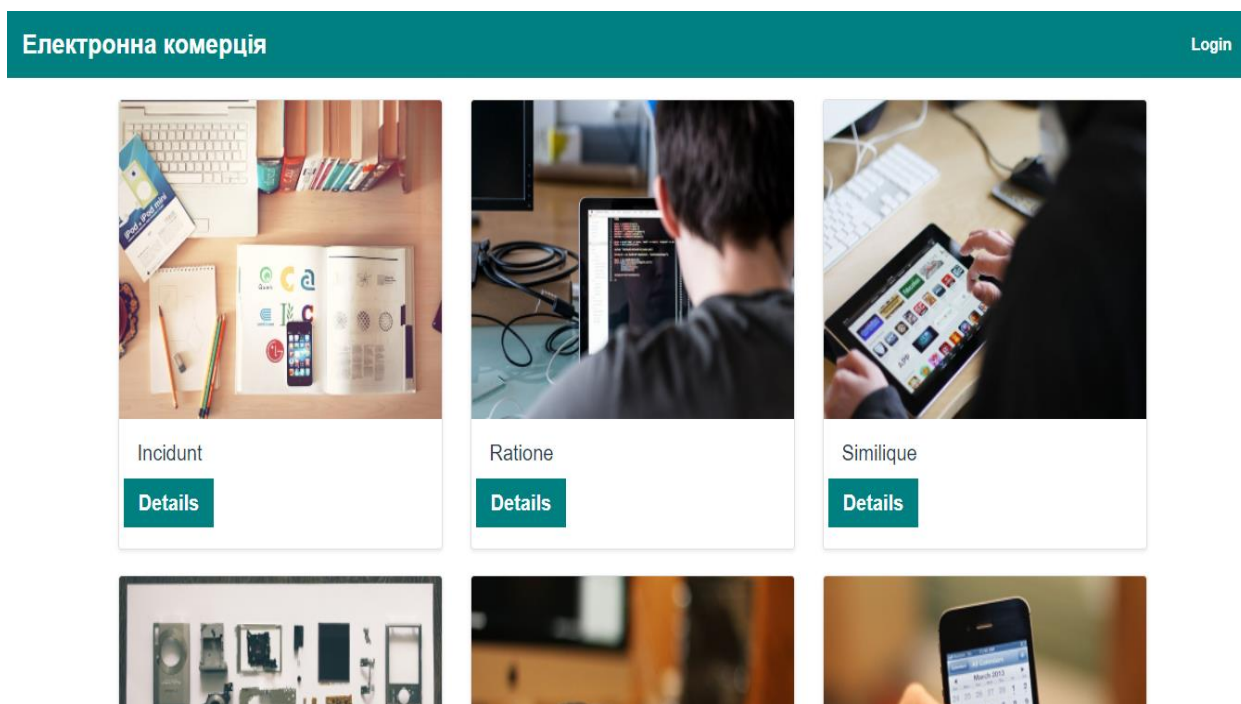


Рисунок 2.5 — Головна сторінка сайту

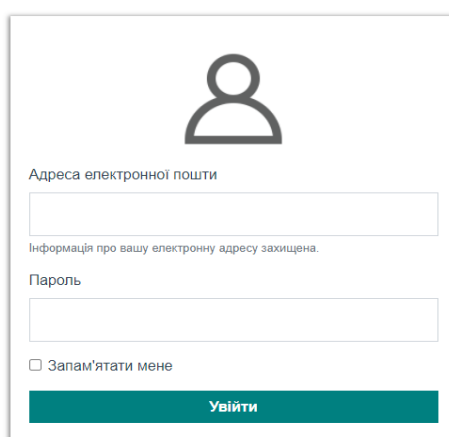
На головній сторінці ми бачимо шапку, яка буде спільна для усіх секцій сайту, і буде різнитися тільки в залежності від таких факторів як:

- чи є користувач адміністратором;
- чи зареєструвався користувач;

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

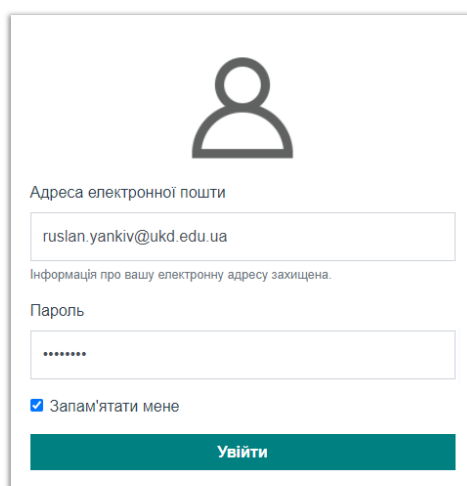
Так ми бачили поле, в якому зручно розташовані меню фільтрування, пошуку та сортування. Центральна частина сторінки, це список товарів, що належать до вибраної категорії.

На цьому етапі, для отримання можливості використовувати сайт на максимум, користувач буде змушений авторизуватись. ProductItem компонент, що зображений у вигляді блоку, та містить дані про товар. Щоб переглянути як виглядає форма для входу свого кабінету (рис. 2.6, 2.7).



The screenshot shows a login form with a user icon at the top. Below it are two input fields: 'Адреса електронної пошти' (Email address) and 'Пароль' (Password). A checkbox labeled 'Запам'ятати мене' (Remember me) is located below the password field. A green button labeled 'Увійти' (Login) is at the bottom.

Рисунок 2.6 — Вхід користувача



The screenshot shows the same login form as in Figure 2.6, but with the input fields filled. The email field contains 'ruslan.yankiv@ukd.edu.ua' and the password field contains '*****'. The 'Remember me' checkbox is now checked. The 'Увійти' (Login) button is at the bottom.

Рисунок 2.7 — Заповнені поля входу

На сторінці кошика ми вказуємо перелік товарів, які були додані в

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

кошик. В тому ж кошику будуть відображені такі деталі, як зображення, назва, ціна та кількість товару.

Також розраховується загальна ціна кожного товару та загальна сума, яку потрібно сплатити.

Для того щоб це все побачити ми маємо форму для входу з полями електронної пошти та пароля (рис. 2.8).

Для покупки товару, його спершу потрібно додати в корзину. Додати в корзину товар можна з головної сторінки.

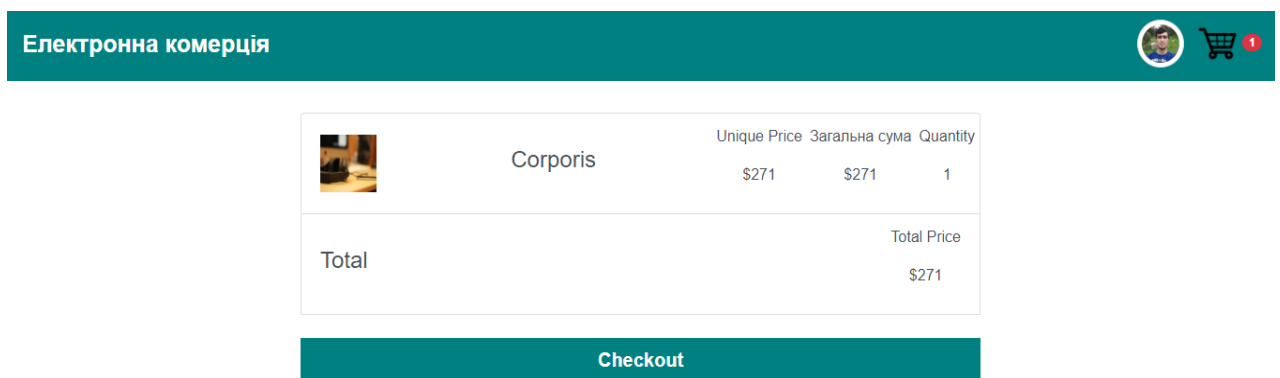


Рисунок 2.8 — Товар у корзині

Також надається можливість ми можемо дода у корзину декілька товарів одночасно, в підсумку ціна рахується і вибиває загальна ціна за товар який ми обрали (рис. 2.9).

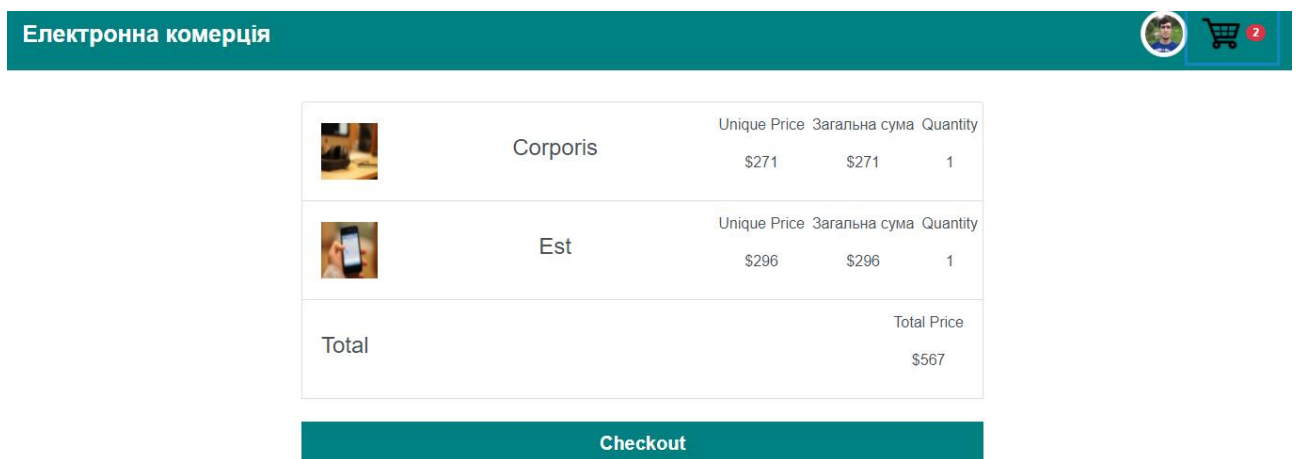


Рисунок 2.9 — Декілька товарів у корзині

					КР.ПЗ – 21.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Висновки до розділу 2

Було розроблено та описано основні інструменти розробки, якими є Visual studio призначений для створення, редагування та для застосування веб застосунків для хмарних систем. Також було використано HTML та CSS. Для використання ми обрали (Vue.js, JavaScript-фреймворк) яка використовує шаблони MVVM для створення інтерфейсів користувача на основі моделей.

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

РОЗДІЛ 3. РЕАЛІЗАЦІЯ ТА ОГЛЯД ПЛАТФОРМИ

3.1 Робота з модулем користувача

Під час створення цієї програми ми використовували Vuex для управління станом нашої програми, маршрутизатор Vue для навігації та Axios для отримання даних з API. Нам також знадобилося Node.js для установки Vue.

Встановлюємо необхідні пакети. Центральний магазин програм Vue.js. До даних, що зберігаються у сховищі Vuex, можна отримати доступ із будь-якого компонента програми.

Щоб використовувати Vuex у нашому проєкті, нам потрібно встановити його в наш проєкт. Після його встановлення ми можемо брати з будь-якої частини нашого проєкту.

Vue Router офіційний пакет маршрутизації для Vue.js. Router Vue допомагає нам орієнтуватися на сторінках нашого додатку. Кожна сторінка має певний шлях або URL-адресу або маршрут, який ми реєструємо в маршрутах нашого проєкту. Axios пояснює який пакет NPM для створення HTTP-запитів у наших програмах вузлів.

За допомогою Axios наш додаток може звертатися до інших веб-програмам. Ми можемо надсилати дані та отримувати дані з іншого веб-програмного забезпечення через Rest API. Існують інші методи спілкування з веб-API, але ми встановимо та використовуємо Axios для їхньої простоти та практичності в нашому проєкті.

Створення початкових файлів. Вже встановивши Vuex, нам потрібно створити папку з назвою store у кореневому шляху нашого проєкту. У цьому каталозі ми створимо два модулі, один для користувача та інший для продуктів. Ми також повинні створити index.js файл, куди ми будемо імпортувати наші модулі магазину. У середині папок облікового запису та продукту нам потрібно створити п'ять файлів:

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

- state.js для зберігання даних або станів;
- getters.js для отримання даних;
- actions.js щоб створити наші функції;
- mutations.js змінити держави;
- vue щоб імпортувати всі файли з цього модуля;

Ми можемо розміщувати стани, геттери, мутації та дії лише у файлі JavaScript. Але для організації ми помістимо кожен з них у різні файли та всередині папки, яку ми будемо називати модулем.

На першому кроці ми встановили Vue CLI на свій комп'ютер, створили проект Vue, встановили необхідні пакунки Npm Vuex, Axios та Vue Router і нарешті, створили початкові файли, необхідні для Vuex.

У index.js шляху магазину ми імпортувати модулі із модулів магазину. Назвемо його серце Vuex. Далі ми використали state.js для зберігання даних, до яких може отримати доступ будь-яка частина програми. Для облікового запису нам потрібно зберігати лише дані користувача.

Найважливішою функцією платформи є авторизація, без якої неможлива правильна робота застосунку. Варто оглянути частину розробки функціоналу пов'язану саме з цим. У стані користувачького модуля ми створимо функцію, яка повертає об'єкт із станами цього модуля; у цьому випадку ми маємо лише один стан: userData.

```
export default function ()
{
  return {
    userData: {},
  }
}
```

Геттери повертає штати, для отримувачів модуля облікового запису ми створимо функцію, яка повертає userData стан.

```
export function user (state) {
  return state.userData
}
```

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Змінювати стани можливо лише за допомогою мутацій. Під час мутації модуля облікового запису ми створимо функцію, яка отримує userData стан як параметр і значення, яке йому буде присвоєно, actions.js коли якась функція його вчинить.

Також було використано API, який був створений для того щоб отримувати дані через Axios. В action.js модулі облікового запису ми отримуємо доступ до фіктивного API з Axios, щоб отримати користувача, а потім збережемо його у нашому штаті userData, що здійснює мутації setData.

```
import router from 'router'
import Axios from 'axios';
export function login({ commit }) {
  let url = 'https://randomuser.me/api/';
  Axios.get(url).then(function (response) {
    let userData = {
      displayName: response.data.results[0].name.first,
      email: response.data.results[0].email,
      photoURL: response.data.results[0].picture.thumbnail,
      uid: response.data.results[0].login.uuid }
    commit("setData", userData)
    router.push('/')})
  .catch(function (error){
    console.log(error)}
  );
}
```

У index.js ми імпортуємо метод getter, та робимо зміни, дії та стани. Функція мутації призначить отримане значення userData стану.

```
import state from './state'
import * as getters from './getters'
import * as mutations from './mutations'
import * as actions from './actions'
export default {
  namespaced: true,
  getters,
  mutations,
  actions,
  state
}
```

На кроці ми створили користувальницький модуль, щоб відокремити користувацькі стани від інших станів програми.

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

3.2 Робота з модулем продукту

У модулі товару нам потрібен стан товарів, стан товару для деталей товару та стан кошика для зберігання товарів кошика:

- products state — буде зберігати перелік товарів;
- product state — буде зберігати конкретний товар;
- cart state — збереже список товарів у кошику.

У getters.js модуля продукту ми створимо три функції:

- products — функція повертає стан, в якому зберігається список товарів;
- product — повертає стан, в якому зберігається конкретний товар, коли користувач хоче побачити деталі товару;
- cart — повертає стан, що зберігає товари у кошику.

У mutations.js модулі продукту ми створимо чотири функції:

- setProducts — призначити перелік товарів державним продуктам;
- setProduct — призначити об'єкт із певним товаром до стану товару;
- setCart — призначає список доданих товарів у кошику до стану кошика.

Усередині actions.js було розроблено функцію отримання всіх продуктів за допомогою Axios, функцію отримання деталей товару, функцію додавання товару в кошик та функцію видалення товару з кошика. У нас є кінцева точка або URL-адреса для отримання даних із сервера, а потім за допомогою Axios ми робимо отримання в цій кінцевій точці.

Для деталей конкретного товару концепція не змінюється від попередньої. Різниця полягає в тому, що в Axios ми передаємо параметр для отримання продукту.

Параметр дозволяє ідентифікувати товари, для якого ми хочемо побачити деталі. Щоб видалити певний об'єкт з кошика, ми повторимо те, що є в масиві кошика, і до дамо елементи, які не мають однаковий ідентифікатор товару, який

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

ми хочемо видалити, до тимчасової змінної, а потім до дамо кошик, здійснивши його мутацію.

На третьому кроці ми створили модуль продукту, щоб відокремити його стани від інших станів нашої програми. Модуль товару стосується лише станів товару. Ми також використовували API відпочинку, який був створений для отримання даних через Axios.

3.3 Створення необхідних компонентів

Після того, як у нашому магазині буде все готово, ми будемо працювати з візуальною частиною, нашими компонентами. У нашому проєкті ми будемо використовувати завантажувальний ремінь. Щоб додати bootstrap, нам потрібно додати bootstrap CSS CDN всередину тегів файлу index.html у загальній папці. Тепер ми можемо використовувати класи bootstrap в нашому проєкті. Тепер було створено батьківський компонент, який в подальшому називатиметься layout.

```
<template>
  <div>
  <div>
    <nav class="navbar navbar-expand-lg navbar-dark fixed-top">
<router-link class="navbar-brand"
to="/">CS-eCommerce</router-link>
    <div class="ml-auto">
    <div v-if="user.photoURL">
    <img:src="user.photoURL"
    class="img-thumbnail profile-image"
    alt/>
    <router-link class="btn btn-primary my-2 my-sm-0" to="/cart">
```

Компонент макета, у кореневому шляху нашого проєкту ми створимо папку з назвою layout, а всередині цієї папки створимо файл, який називатиметься Base.vue макетом для всіх наших сторінок.

```

```

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

```

    <span class="badge badge-danger badge-pill">{{ cart.length }}</span>
  </router-link>
  <router-link v-else class="btn btn-primary my-2my-sm-0"
to="/login">Login</router-link>

```

Щоб додати інформацію про користувачів та кошик, ми отримаємо дані з `vuex` with `mapGetters`, який поверне нам дані користувачів та кошика, які перебувають у станах, створених раніше.

```

<div class="page-container">
  <router-view />
</template>
<script>
import { mapGetters } from "vuex";
export default {
  name: "Base",
  computed: {
    mapGetters("account", ["user"]),
    mapGetters("product", ["cart"])
  }
};

```

В основному, у нашому макеті ми маємо панель навігації, представлену `<nav>` тегами, і основний контейнер у `div` з класом `page-container`, куди перегляд маршрутизатора буде вводити всі сторінки. Інформація про користувача та кошик може відображатися лише за умови автентифікації користувача.

```

<div v-if="user.photoURL">
  <img:src="user.photoURL" class="img-thumbnail profile-image"
  alt/>
  <router-link class="btn btn-primary my-2 my-sm-0" to="/cart">
  <span class="badge badge-danger badge-pill">{{ cart.length }}</span>
</router-link>
</div>

```

Для користувача ми покажемо лише його зображення профілю, а для кошика нам потрібно лише порахувати, скільки елементів було додано, бачу чи довжину цього масиву.

Тепер ми створимо папку під назвою `Home`, яка буде містити всі компоненти продуктів. У середині папки `Home` ми створимо картковий компонент продуктів `ProductCard.vue`.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

```

<template>
<div class="card mb-4 shadow-sm">

<div class="card-body">
<h5 class="card-title text-left">{{ product.name }}</h5>
<div class="row">
<router-link
type="button"
class="btn btn-primary btn-lg"
:to="/details/" + product.id"
>Details</router-link>

```

Щоб додати інформацію про користувачів та кошик, ми отримаємо дані з `vueX` with `mapGetters`, який поверне нам дані користувачів та кошика, які перебувають у станах, створених раніше.

```

</template>
<script>
export default {
name: "ProductCard",
props: ["product"]};
</script>
<style>
.card .product-image {
height: 300px;}
</style>

```

Після створення картки товару далі імпортуємо її до подання продукту, а потім розмістимо в ітерації продукту. для цього створіть новий файл під назвою `Products.vue` всередині домашньої папки.

```

<template>
<div style="padding: 25px;">
<div class="container">
<div class="row">
<div class="col-md-4" v-for="product in products" :key="product.id">
<ProductCard :product="product" />

```

Ми встановимо товар і кількість у стан, який називається кошик. Щоб перевірити, чи додано товар у кошик, ми маємо спосіб перевірити `if this item is in the array`. Цей метод викликається, коли стан товару чи кошик стану мають певні зміни.

```

</template>

```

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

```

import { mapActions, mapGetters } from "vuex";
import ProductCard from "../../components/ProductCard";
  computed: {
    mapGetters("product", ["products"]),},
  components: { ProductCard },
  methods: {
    mapActions("product", ["getProducts", "addCart", "removeCart"]),
  },
  mounted() {
    this.getProducts();
  }
};
</script>
<style>
</style>

```

Метод дії для отримання продуктів: дані надходять з API, і збережемо їх у стані, здійснивши мутацію цього стану. У такому випадку це стан товару.

```

export function getProducts({ commit }) {
  let url = "https://my-json-server.typicode.com/Nelzio/ecommerce-fake-json/products";
  axios.get(url).then((response) => {
    commit("setProducts", response.data);
  }).catch(error => {
    console.log(error);
  });
}

```

Щоб перелічити товари на сторінці Продукти, ми беремо дані у стані продуктів, повторюємо ці дані та розміщуємо кожен товар на картці. Додати до кошика компонент: спочатку давайте ми створимо папку з назвою деталі та компонент, щоб додати товар у кошик. Ми встановимо товар і кількість у стан, який називається кошик. Щоб перевірити, чи додано товар у кошик, ми маємо спосіб перевірити `if this item is in the array`. Цей метод викликається, коли стан товару чи кошик стану мають певні зміни. Отже, вони перебувають під вахтовим блоком цих штатів. Щоб додати товар у кошик, ми беремо його інформацію та кількість, щоб додати його до одного об'єкта, а потім зберігаємо в стані.

```

export function addCart({ commit, getters }, payload) {
  let cart = getters.cart
  let data = payload.product
  data["quantity"] = payload.quantity
  cart.push(data)
  commit("setCart", cart)
}

```

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Щоб видалити товар із кошика, потрібно розробити повторення масиву у стані товарів і додати кожен елемент до тимчасового масиву, якщо це не товар, який потрібно видалити, і додати цей тимчасовий масив до стану кошика.

У поданні деталей було показано більше деталей товару та імпортування компонент, щоб додати деталі товару в кошик. Компонент для додавання товару в кошик повинен відображатися лише за умови автентифікації користувача. також директива v-if використовується для умовного візуалізації блоку.

Компонент для додавання товару в кошик повинен відображатися лише за умови, що користувач пройшов автентифікацію. На сторінці кошика було перелічено товари, додані в кошик, та такі деталі, як зображення, назва, ціна та кількість товару.

Також було розраховано загальну ціну кожного товару та загальну суму, яку потрібно сплатити. всередині домашньої папки ми створили файл із назвою Cart.vue. Далі ми створимо просту форму автентифікації, але не будемо отримувати дані. Для входу ми маємо форму з полями електронної пошти та пароля, які не використовуватимуться. І у нас є кнопка, яка викликає функцію входу в дію попередньо створеного модуля облікового запису.

```
<template>
  <div>
    <div class="container" style="padding-top: 10%">
      <div class="row d-flex justify-content-center">
        <div class="col-5 text-left login-form-container">
          <div class="d-flex justify-content-center">
            
            <div class="form-group">
              <label for="exampleInputEmail1">Email address</label>
              <input
                type="email"
                class="form-control"
                id="exampleInputEmail1"
                aria-describedby="emailHelp"/>
          </div>
        </div>
      </div>
    </div>
  </div>
</template>
```

Компонент для додавання товару в кошик повинен відображатися лише за умови, що користувач пройшов автентифікацію. На сторінці кошика було перелічено товари, додані в кошик, та такі деталі, як зображення, назва, ціна та

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

кількість товару.

```
id="emailHelp"
  class="form-text text-muted"
  >We'll never share your email with anyone else.</small>
</div>
<div class="form-group">
  <label for="exampleInputPassword1">Password</label>
    <input type="password" class="form-control" id="exampleInputPassword1"
  />
</div>
<div class="form-group form-check">
  <input type="checkbox" class="form-check-input" id="exampleCheck1" /><label
  class="form-check-label"
  for="exampleCheck1">Check me out</label>
</div>
  <button @click="login()" type="submit" class="btn btn-primary btn-
  block">Login</button>
```

Він також може забезпечити механізми розширення, щоб користувачі могли розширити існуючі функціональні можливості в різних способах і різною мірою.

```
</template>
<script>
import { mapActions } from "vuex";
export default {
  name: "Login",
  methods: {
    mapActions("account", ["login"]),
  },
};
</script>
<style>
</style>
```

Для входу потрібно отримати дані з API та зберегти їх у користувацькому стані. API являє собою інтерфейс, який визначає взаємодії між декількома програмними додатками або змішаними апаратними Software посередниками. Він визначає типи дзвінків або запитів, які можна зробити, спосіб їх здійснення, формати даних, які слід використовувати, домовленості, яких слід дотримуватися тощо. У нашому додатку ми не будемо входити. Ми візьмемо користувача з randomuser.me і збережемо його в нашому користувацькому стані. Якщо об'єкт у користувацькому стані порожній, ми будемо вважати, що користувач не пройшов автентифікацію.

На четвертому кроці спочатку ми додали завантажувальний ремінь у свій

					КР.ІПЗ – 21.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

проект, потім створили необхідні компоненти vue, необхідні у нашому додатку, і нарешті, ми використали API randomuser.me для зберігання користувача з API у наш стан користувача.

У фрагменті CSS було розроблено основні класи та основні теги та внесли деякі необхідні зміни. у нас є також деякі CSS селектори в тому числі .profile-image, .page-container, .navbar-brand, .btn, .btn:hover і .btn:focus.

- .profile-image встановлює розмір зображення профілю в нашому додатку;
- .page-container просто визначає верхнє заповнення в нашому додатку;
- .input робить кнопки з квадратними каймами;
- .btn, ми змінили колір;
- .btn:hover;

Ми встановлюємо колір наведення, це означає, що коли користувач буде наводити мишу на кнопку, колір зміниться на інший колір, який ми туди встановили. Для компонента входу ми додали тіньову межу до div форми, змінили колір кнопки та зробили введення кнопки квадратними, login-form-container селектор CSS задає заповнення цієї форми входів з коробкою тінню навколо входів. На п'ятому кроці ми додали стиль до своєї програми та зробили додаток вигляд чистішим та приємнішим.

Висновки до розділу 3

Було розроблено необхідні пакунки Npm Vuex, Axios та Vue Router. Далі створили модуль продукту, щоб відокремити його стани від інших станів нашої програми. Модуль товару стосується лише станів товару.

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1 Умови роботи на робочому місці з комп'ютером.

Програмування з використанням комп'ютера, відбувається в середовищі, яке в певній мірі впливає на функціональний стан програмістів, які перебувають у комп'ютерному класі. Найважливішими несприятливими факторами середовища при роботі за комп'ютером є:

- фізичні параметри мікроклімату, освітлення електромагнітні випромінювання різних частотних діапазонів; виробничий шум та вібрація; іонізація повітря; статична електрика;
- психофізіологічні перенапруження зорового аналізатора, недостатня рухома діяльність, нервово-емоційне напруження;
- аналіз умов праці виконується для приміщення площею 50м² яке знаходиться, на 3 поверсі п'ятиповерхової будівлі.

Відповідно до ДСТУ 0.00-1.28-10 є неприпустимим розташування приміщень, призначених для роботи з ВДТ у підвалах та цокольних поверхах. Також забороняється розташування вибухонебезпечних приміщень категорії А і Б (ОНТП 24-86) та виробництв з мокрими технологічними процесами поряд з приміщенням, де розташовуються ЕОМ (ПЕОМ), а також над такими приміщеннями, або під ними. Окрім того, виробничі приміщення для роботи з ВДТ не повинні межувати з приміщеннями, у яких рівень шуму та вібрації перевищує допустимі значення.

У процесі роботи з комп'ютером необхідно дотримувати правильний режим праці і відпочинку. У іншому випадку в працюючого за ПК відзначаються значна напруга зорового апарата з появою скарг на незадоволеність роботою, дратівливість, порушення сну, утома і хворобливі відчуття в очах, у попереку, в області шиї і руках та головні болі.

Розташування робочого місця особи, що працює з ПК, в даному випадку

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

студента, повинне відповідати вимогам ДСТУ 22.269-79 «Робоче місце оператора. Взаємне розташування елементів робочого місця».

Облаштування робочого місці повинно забезпечувати:

- правильне розміщення робочого місця у виробничому приміщенні;
- належні умови освітлення приміщення і робочого місця, відсутність відблисків;
- належні ергономічні характеристики основних елементів робочого місця;
- характер та особливості трудової діяльності.

Для зменшення втоми, місця користувачів ЕОМ мають бути зручними. Конструкція робочого місця користувача ПК, (при роботі сидячи) має забезпечувати підтримання оптимальної робочої пози з такими ергономічними характеристиками:

- ступні ніг на підлозі або на підставці для ніг;
- стегна в горизонтальній площині;
- передпліччя вертикально;
- лікті під кутом 70-90 до вертикальної площини;
- зап'ястя зігнуті під кутом не більше 20 відносно горизонтальної площини;
- нахил голови 15-20 відносно вертикальної площини.

Робоче місце користувача ПК, обладнується робочим столом, стільцем і підставкою для ніг. Висота робочого стола має бути в межах від 0,65 до 0,8 м, а ширина повинна забезпечувати можливість виконання операцій в зоні досяжності моторного поля.

Клавіатуру слід розташовувати на поверхні столу на відстані 200 мм від краю, звернутого до працюючого. У конструкції клавіатури має передбачатися опорний пристрій (виготовлений із матеріалу з високим коефіцієнтом тертя, що перешкоджає його переміщенню), який дає змогу змінювати кут нахилу поверхні клавіатури у межах 5...15.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

4.2 Забезпечення правильного мікроклімату у приміщенні

Як один з факторів виробничого середовища, мікроклімат впливає на теплообмін організму людини з цим середовищем. Необхідною умовою життєдіяльності людини є терморегуляція, тобто здатність організму регулювати віддачу тепла в оточуюче середовище.

Цей процес визначається параметрами мікроклімату. Метеорологічні умови визначаються такими параметрами:

- температурою повітря в приміщенні, С;
- відносною вологістю повітря, %;
- рухливістю повітря, м/с;
- тепловим випромінюванням, Вт/м³.

Принцип нормування мікроклімату створення оптимальних умов для теплообміну тіла людини з навколишнім середовищем.

Параметри мікроклімату, вміст шкідливих речовин на робочих місцях, оснащених моніторами, відповідають вимогам ДСН 3.3.6.042-99 «Державні санітарні норми мікроклімату виробничих приміщень», ДСТУ 12.1.005-88 «ССБТ. Повітря робочої зони. Загальні санітарно-гігієнічні вимоги до повітря робочої зони».

Обчислювальна техніка є джерелом істотних тепловиділень, що може привести до підвищення температури і зниження відносної вологості в приміщенні.

В приміщеннях, де встановлені комп'ютери, повинні дотримуватися оптимальні параметри мікроклімату, які визначають комфортні умови. Ці параметри залежать від періоду року, категорії робіт за важкістю, і від теплових характеристик виробничого приміщення, (табл. 4.1).

Робота за комп'ютером характеризується малими фізичним навантаженнями, цей вид діяльності належить до категорії легких робіт за критерієм загальних енерговитрат організму (ДСН 3.3.6.042-99).

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Таблиця 4.1

Параметри мікроклімату для приміщень

Період року	Категорія робіт	Параметр мікроклімату	Величина
Холодний	Роботи легкі I а	Температура повітря в приміщенні	22÷24°C
		Відносна вологість	40÷60%
		Швидкість руху повітря	до 0,1м/с
Теплий	а	Температура повітря в приміщенні	23÷25°C
		Відносна вологість	40÷60%
		Швидкість руху повітря	0,1,0,2м/с

Системи опалення та кондиціонування повітря в приміщеннях з ПК повинні відповідати СНиП 2.04.05-91 «Опалення, вентиляція і кондиціонування». Під час роботи комп'ютерної техніки в повітряному середовищі відбувається суттєва трансформація іонного складу, істотно знижується концентрація легких, середніх та важких негативно зарядних частинок. Така зміна балансу іонного складу призводить до негативного впливу на здоров'я працюючих.

Рівні іонізації повітря приміщень при роботі на персональних комп'ютерах визначені в (таблиці 4.2), відповідно до НПАОП 0.03-3.06-80.

Таблиця 4.2

Рівні іонізації повітря в приміщенні з ПК

Рівні	Кількість іонів в 1 см ³ повітря	
	n+	n-
Мінімально необхідні	400	600
Оптимальні	1500-3000	3000-5000
Максимально допустимі	50000	50000

Для підтримки оптимальних значень мікроклімату та підтримання нормальної концентрації позитивних та негативних іонів в приміщенні аудиторії пропонується удосконалити системи опалення, природної вентиляції та встановити кондиціонер.

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Дисплеї на основі ЕПТ є потенційним джерелом випромінювання кількох діапазонів електромагнітного спектра: рентгенівського, оптичного, радіочастотного.

Кожний вид випромінювання відрізняється своїми особливими характеристиками впливу на організм людини.

Рентгенівське випромінювання. Дослідження показують, що відео термінал не несе небезпеки для користувача ПЕОМ, оскільки інтенсивність такого випромінювання нижча за гранично допустимі норми (ГДН).

Відповідно до «Норм радіаційної безпеки України» (НРБУ-97), гранично допустима потужність експозиційної дози рентгенівського випромінювання на відстані 5 см від поверхні екрана відео терміналу становить $7,74 \cdot 10^{-12}$ Кл/кг, що відповідає еквівалентній дозі 0,1 мбер/год. (100 мкР/год.).

Оптичне випромінювання. Оптичні види випромінювання виникають завдяки взаємодії електронів з шаром люмінофору, нанесеного на екран ВДТ. Область оптичного випромінювання включає ультрафіолетове (УФ), світлове та інфрачервоне (ІЧ) випромінювання.

УФ-випромінювання впливає на шкіру та очі людини. Такий вплив на шкірі проявляється досить швидко, а для очей характерним є період прихованої дії. Рівень УФ випромінювання, який був виявлений, досить низький і становить 1 середньому $0,001 \text{ Вт/м}^2$.

Світлове випромінювання впливає в основному на око і призводить до втоми очей, запалення райдужної оболонки. Однак ці симптоми швидко минають і не викликають патологічних змін.

ІЧ-випромінювання довжина хвиль обмежена від 0,76мм до 1мм. Для більшості біологічних матеріалів випромінювання цього діапазону вважаються непрозорими. Інтенсивність інфрачервоних випромінювана нижча за значення, передбачені ДсанПіН 3.3.2. 007-98.

Робоче місце користувача ПК, обладнується робочим столом, стільцем і підставкою для ніг. Висота робочого стола має бути в межах від 0,65 до 0,8 м, а

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

ширина повинна забезпечувати можливість виконання операцій в зоні досяжності моторного поля.

Клавіатуру слід розташовувати на поверхні. Електромагнітні випромінювання (ЕМВ) радіочастотного діапазону. Джерелом ЕМВ є відео термінал. Тому, обираючи робоче місце для комп'ютера, необхідно пам'ятати, що його задня і бокові стінки можуть бути джерелом значно більшого ЕМВ, ніж екран. З метою профілактики несприятливого впливу електромагнітного випромінювання від ВДТ на користувача необхідно:

- встановити на робочому місці відео термінал, що відповідає сучасним вимогам стосовно захисту від випромінювані;
- не переобтяжувати приміщення значною кількістю робочих місць з ВДТ;
- не концентрувати на робочому місці великої кількості радіо електронних пристроїв. Гранично допустимі рівні випромінювань на робочих місцях з ПК наведено в (таблиці 4.3).

Такий вплив на шкірі проявляється досить швидко, а для очей характерним є період прихованої дії.

Таблиця 4.3

Гранично допустимі рівні випромінювань

Вид випромінювання	Діапазон хвиль (частот)	Гранично допустимий рівень
Іонізуюче випромінювання		
М'яке рентгенівське випромінювання (на відстані 0,05 м від екрана та корпусу ВДТ)	0,01 – 1 нм	100мкР/год
Оптичні випромінювання		
Ультрафіолетове випромінювання	315 – 400 нм	10 Вт/м ²

Продовження таблиці 4.3

Видиме випромінювання (яскравість)	400 – 700 нм	1000 кд/м ²
Інфрачервоне випромінювання	700 нм – 1мм	100 Вт/м ²
Електромагнітні випромінювання (поля радіочастотного та низькочастотного діапазонів)		
Напруженість електромагнітного поля на відстані 0,5 м навколо монітору за електричною складовою	2 кГц – 400 кГц 5 Гц – 2 кГц	2,5 В/м 25 В/м
Щільність магнітного потоку	2 кГц – 400 кГц 5 Гц – 2 кГц	25 нТл 250 нТл
Електростатичні поля		
Поверхневий електростатичний потенціал	–	500 В

Відомо, що шум несприятливо діє на слуховий аналізатор та інші органи та системи організму людини. Визначальне значення щодо такої дії має інтенсивність шуму, його частотний склад, тривалість щоденного впливу, індивідуальні особливості людини, а також специфіка виробничої діяльності. Ті види діяльності, у яких поєднується напружена розумова робота та інтенсивне використання комп'ютера (редагування тексту, верстка оригіналу, "запуск" та від лагодження програм тощо) характеризується відчутним впливом навіть незначних рівнів шуму. Цей вплив виражається у зниженні розумової працездатності, швидкій втомлюваності, послабленні уваги, появі головного болю та ін.

Рівні звукового тиску в октавних смугах частот, рівні звуку та еквівалентні рівні звуку на робочих місцях, обладнаних ВДТ і ПК визначені ДСанПіН 3.3.2-007-98 не повинні перевищувати 65 дБА. Для приміщення аудиторії, рівень шуму не перевищує 60 дБА.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

Для забезпечення нормованих рівнів шуму у виробничих приміщеннях та на робочих місцях застосовуються шумопоглинальні засоби, вибір яких обґрунтовується спеціальними інженерно-акустичними розрахунками.

Близько 90% всієї інформації, що отримується людиною, приходиться на органи зору. Організація освітленості робочих місць грає велику роль у житті людини. Недостатнє та нераціональне освітлення веде до стомлення очей, розладу центральної нервової системи, зниженню розумової та фізичної працездатності, а у ряді випадків може бути причиною травматизму (близько 5% травм приходиться на частку нераціонального та недостатнього освітлення).

Щоб уникнути перевтоми, а також для профілактики професійних захворювань та виробничого травматизму потрібно дотримуватись наступних вимог:

- створювати на робочій поверхні освітленість, що відповідає характеру зорової роботи і не є нижчою за встановлені норми;
- забезпечити достатню рівномірність та постійність рівня освітленості у виробничих приміщеннях, щоб уникнути частоті преадаптації органів зору;
- не створювати засліплюваної дії як від самих джерел освітлення, так і від інших предметів, що знаходяться в полі зору;
- не створювати на робочій поверхні різких та глибоких тіней (особливо рухомих);
- повинен бути достатній для розрізнення деталей контраст поверхонь, що освітлюються;
- не створювати небезпечних та шкідливих виробничих чинників (шум, теплові випромінювання, небезпека ураження струмом, пожежо та вибухонебезпека світильників);
- повинно бути надійним і простим в експлуатації, економічним та естетичним .

Приміщення аудиторії №330 має природну та штучну системи освітлення, які повинні відповідати вимогам ДБН В.2.5-28-2996 «Інженерне

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

обладнання будинків та споруд. Природне та штучне освітлення».

Штучне освітлення в приміщенні з комп'ютеризованим робочим місцем здійснюється системою загального освітлення.

Як джерела світла в разі штучного освітлення застосовуються світильники серії LED-353. Світильники укомплектовані високочастотними пускорегулювальними апаратами (ВЧ ПРА). Яскравість світильників загального освітлення в зоні кутів випромінювання від 50 до 90 град. з вертикаллю в повздовжній та поперечній площинах становить не більше 200 кд/м, захисний кут світильників не менше ніж 40 град. Рівень освітленості у аудиторії, $E = 300$ лк.

Електробезпека система організаційних та технічних заходів і засобів, що забезпечують захист людей від шкідливого та небезпечного впливу електричного струму, електричної дуги, електромагнітного поля і статичної електрики.

Приміщення аудиторії де знаходяться комп'ютери відноситься до приміщень з підвищеною безпекою.

ЕОМ, периферійні пристрої ЕОМ та устаткування для обслуговування, ремонту та налагодження ЕОМ, електропроводи та кабелі за виконанням та ступенем захисту мають відповідати класу зони за ПВЕ, мати апаратуру захисту від струму короткого замикання та інших аварійних режимів.

Використання нульового робочого провідника як нульового захисного провідника забороняється.

Нульовий захисний провід прокладається від склейки групового розподільчого щита до розеток живлення.

Не допускається підключення на щиті до одного контактного затискача нульового робочого та нульового захисного провідників. Площа перерізу нульового робочого та нульового захисного провідника в груповій три провідній мережі повинна бути не менше площі перерізу фазового провідника. Усі провідники повинні відповідати номінальним параметрам мережі та навантаження, умовам навколишнього середовища, умовам розподілу

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

провідників, температурному режиму та типам апаратури захисту, вимогам ПВЕ.

З метою підвищення рівня електробезпеки в приміщенні аудиторії де знаходяться комп'ютери не припустимим є:

- експлуатація кабелів та проводів з пошкодженою або такою, що втратила захисні властивості за час експлуатації, ізоляцією;
- залишення під напругою неізольованих кабелів та проводів;
- застосування саморобних подовжувачів, які не відповідають вимогам ПВЕ до переносних електропроводок;
- користування пошкодженими розетками, розгалужувальними та з'єднувальними коробками, вимикачами та іншими електровиробами;
- застосування для опалення приміщення нестандартного (саморобного) електронагрівального обладнання.

ВДТ є джерелом електростатичних зарядів. Тривале перебування в електростатичному полі, створеному цими зарядами, негативно впливає на здоров'я працюючих: бронхо-легеневі захворювання, порушення серцево-судинної та нервової систем, ураження шкіри тощо .

Напруженість електростатичного поля на робочих місцях, в тому числі й з ВДТ, не повинна перевищувати 20 кВ/м відповідно до ДСТУ 12.1.045-84 «ССБТ.

Електростатичні поля. Допустимі рівні на робочих місцях і вимоги до проведенню контролю», поверхнево електростатичній Потенціал відео термінала не винних перевищувати 500В.

Для запобігання створенню значної напруженості поля та захисту від статичної електрики необхідно:

- встановити нейтралізатори статичної електрики;
- підтримувати в приміщенні з ВДТ відносну вологість повітря не нижче 45-50% (чим сухіше повітря тим більше електростатичних зарядів);

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

- можна для цього використати навіть побутові зволожувачі;
- застелити підлогу в приміщеннях з ВДТ антистатичним лінолеумом і проводити щоденне вологе прибирання;
- складати всі полімерні покриття (чохли) ВДТ у найбільш віддаленому від користувачів місці розміщення;
- протирати екран та робоче місце спеціальною антистатичною серветкою або зволоженою тканиною.

Електробезпека система організаційних та технічних заходів і засобів, що забезпечують захист людей від шкідливого та небезпечного впливу електричного струму, електричної дуги, електромагнітного поля і статичної електрики.

4.3 Вимоги до безпечної експлуатації комп'ютерної техніки

Пожежна безпека при роботі за комп'ютером повинна забезпечуватись у відповідності з вимогами Закону України «Про пожежну безпеку» та НПАОП 0.01-1.01-95 «Правил пожежної безпеки в Україні» та інших нормативних документів.

Відповідно до ОНТП 24-86 «Визначення категорій приміщень і будинків по вибух пожежної і пожежної небезпеки» приміщення аудиторії №330 відноситься до категорії Д.

Найбільш імовірною причиною виникнення пожеж в приміщенні є порушення вимог при експлуатації комп'ютерної техніки та займання електропроводки внаслідок коротких замикань.

Для того щоб уникнути виникнення пожежі, потрібно дотримуватися наступних заходів:

- дотримання правил пожежної безпеки при роботі з комп'ютером, електрообладнанням та освітлювальними приладами;
- періодичний контроль цілісності і надійності електроізоляції;
- наявність інструкцій з пожежної безпеки;

					КР.ІІЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

- навчання, атестація і переатестація персоналу з пожежної безпеки;
- наявність системи захисту від атмосферної електрики;
- періодичне зняття зарядів статичної електрики;
- заборона куріння в приміщенні.
- застосування будівельних конструкцій із ступенем вогнестійкості не нижче II, а також використання важко горючих або негорючих матеріалів в інтер'єрі виробничого приміщення;
- наявність схеми евакуації;
- наявність пристроїв автоматичного вимкнення ПЕОМ та іншого електроустаткування на випадок пожежі;
- наявність первинних засобів пожежогасіння (вогнегасник ВПК-5-2 шт).

Висновки до розділу 4

Було описано охорону праці, як потрібно себе поводити під час роботи з комп'ютерами. Щоб забезпечити нормальні умови праці на робочому місці з комп'ютерною технікою. Також ми описали що потрібно забезпечення правильного мікроклімату у приміщенні. Не мало важливо є вимоги до безпечної експлуатації комп'ютерної техніки. Будучи на роботі ми повинні виконувати вимоги, для власного забезпечення.

					КР.ІПЗ – 21.00.00.000 ІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи було розроблено інтернет-ресурс з продажу електроніки з повним функціоналом. В процесі роботи над першим розділом було розглянуто описано та проведено аналіз існуючих аналогів електронної комерції. Були описані основні інструменти розробки, якими є Visual studio призначений для створення, редагування та для застосування веб застосунків для хмарних систем.

Також було використано HTML та CSS. Для використання ми обрали Vue.js JavaScript фреймворк яка використовує шаблони MVVM для створення інтерфейсів користувача на основі моделей даних, через реактивне зв'язування даних. Встановлено необхідні пакунки Npm Vuex, Axios та Vue Router. Створено модуль продукту, для відокремлення його стану від інших станів нашої програми. Модуль товару стосується тільки станів товару. У фрагменті CSS взято основні класи та основні теги та внесено необхідні зміни, що покращило дизайн та змінило функціонал на більш зрозумілий та інтуїтивний.

При виникненні необхідності можливе покращення створеного програмного продукту, на який буде потрібно мінімум змін вихідного коду програми, мінімум змін в дизайні.

					КР.ІІЗ – 21.00.00.000 ІІЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Інтернет магазин. *Wikipedia*: вебсайт. URL: <https://uk.wikipedia.org/>
2. Існуючі аналоги. *Wikipedia*: вебсайт. URL: <http://it-sklad.com.ua/index.php>
3. CSS. *Wikipedia*: вебсайт. URL: <https://uk.wikipedia.org/wiki/CSS>
4. React. *Wikipedia*: вебсайт. URL: <https://ua.wikipedia.org/wiki/React>
5. Node.js. *Wikipedia*: вебсайт. URL: <https://ua.wikipedia.org/wiki/Node.js>
6. Vue.js. *Wikipedia*: вебсайт. URL: <https://en.wikipedia.org/wiki/Vue.js>
7. Vue. *Wikipedia*: вебсайт. URL: <https://vuex.vuejs.org/#what-is-a-state-management-pattern>
8. UML. вебсайт. URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/>
9. Use case. *Wikipedia*: вебсайт. URL: <https://www./pages/uml-use-case-diagram>
10. Vue.js. *Wikipedia*: вебсайт. URL <https://codesource.io/building-an-e-commerce-app-with-vue-js-vuex-axios>
11. TOS.IN. *tos.in.ua*: вебсайт. URL: <https://tos.in.ua/>
12. JAZZ-CLUB-SERVICE. *Jcs.ua* вебсайт. URL: <https://jcs.ua/>
13. MVC. *Wikipedia*: вебсайт. URL: <https://ru.wikipedia.org/wiki/Model-View-Controller>
14. Клієнт-серверна архітектура. *Wikipedia*: вебсайт. URL: https://uk.wikipedia.org/wiki/Клієнт-серверна_архітектура
15. Організація і технологія роботи інтернет-магазину. *pidru4niki.com*: вебсайт. URL: <https://pidru4niki.com/10931123/informatika/organizat>

					КР.ІПЗ – 21.00.00.000 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підп.	Дата		

