

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ім. Ю.М. Потебні
ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ
КАФЕДРА ЕЛЕКТРОНІКИ, ІНФОРМАЦІЙНИХ СИСТЕМ ТА
ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Кваліфікаційна робота

перший (бакалаврський)

(рівень вищої освіти)

на тему **Розробка розширення для реалізації відображення тривимірних
моделей товарів на сайті електронної комерції**

Виконав: студент 4 курсу, групи 6.1210-пзс
спеціальності 121 Інженерія програмного забезпечення

(код і назва спеціальності)

освітньої програми Програмне забезпечення систем

(код і назва освітньої програми)

С. О. Ветров

(ініціали та прізвище)

Керівник к.т.н., доцент В. І. Заяц

(посада, вчене звання, науковий ступінь, підпис, ініціали та прізвище)

Рецензент директор ТОВ Дискус П.О. Лютий

(посада, вчене звання, науковий ступінь, підпис, ініціали та прізвище)

Запоріжжя
2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ІНЖЕНЕРНИЙ НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ім. Ю.М. Потебні
ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ

Кафедра Електроніки, інформаційних систем та програмного забезпечення
Рівень вищої освіти перший(бакалаврський)
Спеціальність 121 Інженерія програмного забезпечення
(код та назва)
Освітня програма Програмне забезпечення систем
(код та назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри Т. В. Критська
" 01 " 2024 року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТОВІ

Ветрову Сергію Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції

керівник роботи Заяц Валерій Іванович, к.т.н., доцент
(прізвище ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом ЗНУ від 29.12.2022 № 1893-с

2. Строк подання студентом кваліфікаційної роботи _____

3. Вихідні дані кваліфікаційної роботи бакалавра

- комплект нормативних документів ;
- технічне завдання до роботи.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

- огляд та збір літератури стосовно теми кваліфікаційної роботи;
- огляд та аналіз існуючих рішень та аналогів;
- створення програмного продукту та його опис;
- дослідження поставленої проблеми та розробка висновків.

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

слайдів презентації

6. Консультанти розділів бакалаврської роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата
		Завдання прийняв

7. Дата видачі завдання 01.03.2024

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів бакалаврської роботи	Примітка
1	Аналіз предметної області	01.03 – 10.03.24	виконано
2	Формулювання основної задачі дипломної роботи та узгодження її з науковим керівником	11.03 – 15.03.24	виконано
3	Збір літератури стосовно теми кваліфікаційної роботи	16.03 – 25.03.24	виконано
4	Аналіз програмних продуктів – аналогів	26.03 – 05.04.24	виконано
5	Аналіз сучасних технологій для створення клієнтської частини веб застосунків	06.04 – 15.04.24	виконано
6	Аналіз бібліотек для демонстрації об'ємних моделей	16.04 – 25.04.24	виконано
7	Порівняльний аналіз засобів збірки застосунка	26.04 – 05.05.24	виконано
8	Аналіз існуючих систем керування вмісту для сайтів електронної комерції	06.05 – 15.05.24	виконано
9	Розробка архітектури системи	16.05 – 25.05.24	виконано
10	Розробка модулів та алгоритмів	26.05 – 31.05.24	виконано
11	Розробка інтерфейсу системи	01.06 – 05.06.24	виконано
12	Оформлення звіту	06.06 – 10.06.24	виконано

Студент _____ С.О Ветров
(підпис) (прізвище та ініціали)

Керівник роботи _____ В.І. Заяц
(підпис) (прізвище та ініціали)

Нормоконтроль пройдено

Нормоконтролер _____ І.А. Скрипник
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Сторінок – 67

Рисунків – 13

Таблиць – 1

Джерел – 19

Ветров С. О. Розробка розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції: кваліфікаційна робота бакалавра спеціальності 121 «Інженерія програмного забезпечення» / наук. керівник В.І. Заяц. Запоріжжя : ЗНУ, 2024.

Одним з ключових аспектів успішного онлайн-продажу є якісне візуальне представлення товарів. Використання зображень, відео та інших візуальних форматів допомагає покупцям прийняти рішення про покупку. Впровадження модуля для відображення тривимірних моделей товарів на сайті дозволяє користувачам розглянути товар з усіх боків, покращуючи їхню взаємодію з продуктом та підвищуючи ймовірність покупки.

Проведено порівняльний аналіз існуючих продуктів-конкурентів. Виходячи з результатів цього аналізу було розроблено модуль для відображення об'ємних моделей на сайті електронної комерції. Реалізований модуль дозволяє користувачам досліджувати об'ємні моделі товарів, що допомагає покупцю краще вивчити товар перед його покупкою.

Мета проєкту – Розробити розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції.

Ключові слова: *Об'ємна модель, тривимірна модель, візуалізація, електронна комерція, модуль, застосунок.*

ABSTRACT

Pages — 67

Drawings — 13

Tables — 1

Sources — 19

Vietrov S.O. Development of an extension for the implementation of displaying three-dimensional models of goods on an e-commerce site: bachelor's thesis in specialty 121 “Software Engineering” / supervisor V.I. Zayats. Zaporizhzhia : ZNU, 2024.

One of the key aspects of successful online sales is a high-quality visual representation of goods. The use of images, videos and other visual formats helps customers make a purchase decision. Implementation of a module for displaying three-dimensional product models on the website allows users to view the product from all sides, improving their interaction with the product and increasing the likelihood of purchase.

A comparative analysis of existing competing products was conducted. Based on the results of this analysis, a module was developed to display three-dimensional models on an e-commerce website. The implemented module allows users to explore three-dimensional models of goods, which helps the buyer to better study the product before buying it.

Project goal - To develop an extension to implement the display of three-dimensional models of goods on an e-commerce site.

Keywords: Three-dimensional model, three-dimensional model, visualization, e-commerce, module, application.

ЗМІСТ

ВСТУП	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	10
1.1 Огляд літературних джерел	10
1.2 Аналіз програмних продуктів – аналогів	12
1.3 Постановка задачі	18
2 ДОСЛІДЖЕННЯ ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ МОДУЛЯ.....	20
2.1 Аналіз бібліотек для демонстрації об’ємних моделей.....	20
2.2 Порівняльний аналіз засобів збірки застосунка	22
2.3 Аналіз існуючих систем керування вмісту для сайтів електронної комерції.....	26
2.4 Висновок	Помилка! Закладку не визначено.
3 РОЗРОБКА РОЗШИРЕННЯ ДЛЯ САЙТУ ЕЛЕКТРОННОЇ КОМЕРЦІЇ....	35
3.1 Огляд предметної сфери.....	35
3.2 Архітектура системи.....	36
3.3 Функціональні вимоги системи.....	41
3.4 Вимоги до апаратного та програмного забезпечення	46
3.5 Модулі і алгоритми.....	47
3.6 Проєкт інтерфейсу	58
ВИСНОВКИ.....	64
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	65

ВСТУП

Актуальність теми

Візуальне представлення товарів в інтернет-магазинах є одним з найголовніших факторів при купівлі через інтернет. Використовуючи зображення, відео та інші способи візуального представлення, покупець вирішує купляти товар, або ні. Модуль для відображення тривимірних моделей дозволить користувачу побачити тривимірну модель товару безпосередньо на сайті продавця.

Сучасні методи рендерингу тривимірних моделей дозволяють відображати об'ємну модель товару на сторінці продукту без втрат у якості та продуктивності роботи сайту. Цей метод візуалізації вирішує одразу декілька проблем: покупцю дозволяє визначитись, чи підходить товар для нього, а продавцю дозволяє краще продавати товар.

Застосування тривимірних моделей дозволяє створити інтерактивний досвід, що значно підвищує залученість покупців. Користувачі можуть обертати модель, змінювати кут огляду, збільшувати або зменшувати масштаб, що дає їм відчуття реальної взаємодії з товаром. Це особливо важливо для складних або високотехнологічних продуктів, де візуалізація деталей може суттєво вплинути на рішення про покупку.

Тривимірна візуалізація також підвищує довіру покупців до товару та магазину. Коли користувачі бачать, що магазин інвестує в сучасні технології для покращення їхнього досвіду, це створює враження надійності та професіоналізму. Крім того, можливість детально розглянути товар з усіх боків знижує ймовірність повернення товарів через невідповідність очікуванням.

Для продавців використання тривимірних моделей означає покращення показників конверсії та середньої вартості замовлень. Інтерактивні моделі дозволяють виділити унікальні особливості та переваги продукту, що може бути складно зробити через статичні зображення або текстові описи. Це

особливо важливо в умовах високої конкуренції, де кожна деталь може вплинути на вибір покупця.

Мета дослідження

Розробка розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції.

Завдання дослідження

Розглянути існуючі рішення для відображення тривимірних моделей на сайтах електронної комерції під керуванням CMS Magento 2. Проаналізувати існуючі підходи та технології, які використовувались для відображення об'ємних моделей на сторінці продукту.

Об'єкт дослідження

Об'єктом дослідження є процес розробки розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції.

Предмет дослідження

Предметом дослідження є технології створення розширення для перегляду об'ємних моделей товарів на сторінці продукту, яке дозволить задовільнити потреби власників інтернет-магазинів та покупців, які хочуть більш детально побачити товар перед його покупкою.

Методи дослідження

Теоретичні – аналіз і порівняння технологій створення розширень для сайтів електронної комерції; порівняльний аналіз для вибору фреймворку;

Практичне значення одержаних результатів

Практичне значення одержаних результатів дослідження полягає у тому щоб надати можливість продавцю додавати архів з об'ємною моделлю продукту та переглядати її на сторінці продукту.

Апробація результатів

Результати роботи було представлено на XVII університетській науково-практичній конференції студентів, аспірантів, докторантів і молодих вчених «Молода наука-2024» [17].

Глосарій

Розширення для відображення тривимірних моделей – Модуль для системи керуванням вмісту, який дозволяє завантажувати та переглядати тривимірні моделі продуктів на сайті електронної комерції.

Продукт – це основна сутність, що представляє товари або послуги, які продаються в інтернет-магазині.

Three.js – JavaScript бібліотека для створення 3D графіки у веб-браузерах, використовуючи WebGL.

Magento 2 — платформа для електронної комерції з відкритим вихідним кодом, яка дозволяє створювати та керувати онлайн-магазинами різної складності.

EAV (Entity-Attribute-Value) — це модель даних, яка використовується для зберігання різноманітних атрибутів сутностей без необхідності зміни схеми бази даних, яка активно використовується у Magento 2.

Рендеринг (Rendering) — це процес створення зображення або відео з тривимірної моделі або сцени за допомогою комп'ютерних програм.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд літературних джерел

За останнє десятиліття в індустрії електронної комерції спостерігалось значне зростання, оскільки вона фокусується на зручності та доступності, що призводить до сплеску онлайн-покупок, і все більше споживачів обирають саме онлайн-магазини. Однією з найважливіших переваг індустрії електронної комерції є підвищена зручність для споживачів [1].

Онлайн-шопінг дає змогу клієнтам робити покупки, коли вони захочуть, не стикаючись із проблемами у вигляді поїдок у різні кінці міста за товарами, довгими чергами або пошуком товару в наявності. Покупки в Інтернеті стали доволі популярні серед працюючого населення, оскільки це позбавляє від необхідності відвідувати звичайні магазини в робочий/неробочий час. Крім того, онлайн-шопінг також може надати споживачам ширший вибір товарів, оскільки багато веб-сайтів електронної комерції пропонують товари з усього світу. Ця підвищена доступність платформ електронної комерції забезпечує легкий доступ до широкого асортименту продуктів не витрачаючи час та пальне на пошуки [2].

У той самий час індустрія електронної комерції стикається з різними проблемами, такими як захист персональних даних клієнтів, обмін інформацією про продукти, час простою платформи, розробка юзер-інтерфейсу, візуалізація продукту (2D, 3D, VR). Щоб повністю використати потенціал цієї галузі, важливо визначити її переваги та проблеми та зосередитися на шляхах пом'якшення проблем та забезпечення її зростання [3].

В Україні поява та розвиток електронної комерції відбулися дещо пізніше, ніж у розвинених країнах, зокрема європейських. Незважаючи на це, у 2018 році Україні вдалося продемонструвати друге за темпами зростання сектору електронної комерції серед європейських країн — 31%, згідно з European Ecommerce Report. Це приголомшливе зростання ринку перевищило

лише Румунія, тоді як Україна значно випередила середній світовий показник у 25%.

За останні шість років ринок електронної комерції в Україні продемонстрував сукупний річний темп зростання в 24,7%, що навіть перевищило зростання світового ринку в 21,5%. Станом на 2019 рік загальний обсяг ринку в Україні досяг 76 млрд грн, або 7% від загального обсягу роздрібних продажів.

Виявлено, що до 2022 року в Україні відбувалося значне зростання обсягу та темпів росту ринку електронної комерції. Проте із початком війни ринок пережив шокове падіння обсягів, проте станом на 2023 рік помітні стабілізаційні процеси. Зниження частки платоспроможного населення частково компенсується за рахунок збільшення частоти купівель [5].

Електронна комерція — це купівля та продаж товарів через Інтернет. На відміну від звичайної комерції, коли покупці можуть перевірити свій продукт, споживачі повинні покладатися на візуальні елементи, щоб зробити вибір покупки в електронній комерції. Завдяки використанню 3D-моделей в електронній комерції споживачі можуть перевіряти речі під будь-якими кутами та відстанями. Це дозволяє споживачеві отримати додаткову інформацію про продукт.

Дослідження показують, що надання споживачам контролю над інформацією позитивно впливає на їхні наміри щодо покупки та розуміння продукту. Тривимірні візуалізації продуктів сприяють цьому, дозволяючи користувачам детально досліджувати продукт. На взаємодію з онлайн-магазином впливають різні когнітивні фактори. Інтерактивні 3D-продукти також викликають минулий досвід, спогади та емоції споживачів, забезпечуючи позитивний сенсорний досвід. Було виявлено, що 3D-презентації 360° є найбільш розумними для споживачів, які отримують доступ через комп'ютери, де вони сприймають підвищену цінність продукту, кращий споживчий досвід і, що найважливіше, більшу ймовірність покупки [6].

1.2 Аналіз програмних продуктів – аналогів

Ідея створення розширень для Magento 2, які б дозволили демонструвати об'ємні моделі товарів не є новою. Проблема обмеженості можливостей демонстрації продукту для покупця існує вже давно, тому на ринку розширень для платформ електронної комерції існує декілька готових модулів для вирішення цієї задачі.

Milople 3D Product View

Розширення 3D Product View від компанії Milople є платним модулем для Magento 2, який дозволяє додавати файли з об'ємними моделями до продукту та переглядати їх користувачами на сторінці продукту. Модуль надає можливість завантаження об'ємних моделей продуктів, перегляд на 360 градусів та їх видалення [7].

Адміністратори можуть легко керувати файлами моделей з адміністративної панелі, завантажувати нові моделі, переглядати їх і видаляти за потреби. Інтерактивний перегляд на 360 градусів дозволяє клієнтам обертати, масштабувати та переміщати тривимірні моделі товарів, забезпечуючи повний огляд з будь-якого кута, що покращує користувацький досвід і допомагає клієнтам краще зрозуміти, як виглядає продукт. Режим попереднього перегляду моделі на сторінці продукту в адмін-панелі дозволяє адміністраторам переглядати модель до застосування змін, що забезпечує точність і дозволяє уникнути помилок перед публікацією продукту на сайті.

Модуль використовує бібліотеку model-viewer від компанії Google, яка є потужним інструментом для перегляду об'ємних моделей. Model-viewer представляє собою HTML елемент, у який прописуються параметри відображення, такі як шлях до об'ємної моделі, параметри згладжування моделі, налаштування світла та інші. Модуль підтримує сучасні формати об'ємних моделей, такі як glTF, GLB, USDZ, що забезпечує сумісність з більшістю 3D-моделей, створених у різних програмах для 3D моделювання.

Використання бібліотеки `model-viewer` дозволяє налаштувати зовнішній вигляд сцени, включаючи параметри освітлення та згладжування моделі, що дає можливість створити привабливий і реалістичний вигляд 3D-моделей.

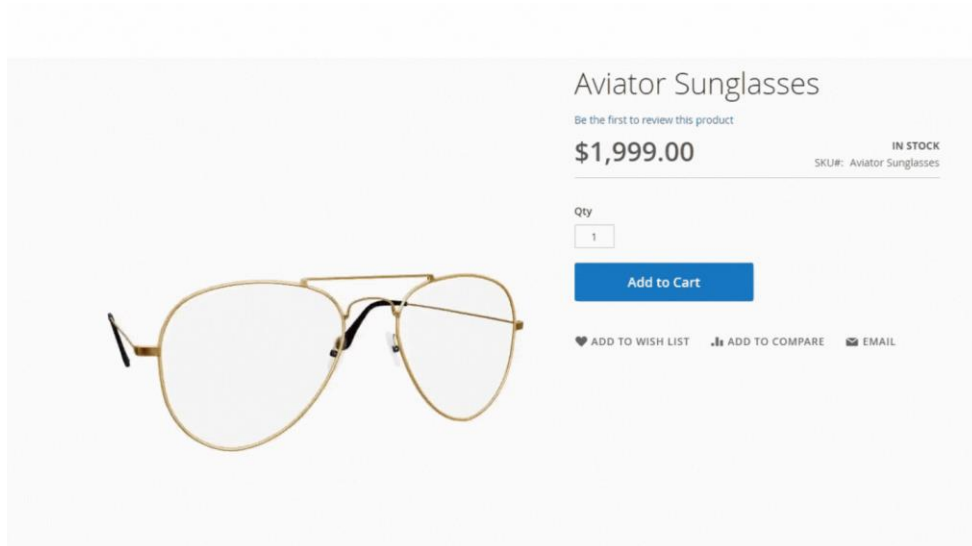


Рисунок 1 — *Сторінка продукту з об'ємною моделлю*

Підтримка теми Нува, сучасної теми для Magento 2, яка не використовує вбудовану бібліотеку `require.js` для ініціалізації JavaScript, забезпечує кращу продуктивність і швидкість завантаження сторінок. Серед переваг модуля можна виділити можливість попереднього перегляду моделі, підтримку сучасних форматів об'ємних моделей та використання потужної бібліотеки `model-viewer`, яка підтримує велику кількість налаштувань зовнішнього вигляду сцени.

Розширення 3D Product View від компанії Milople є потужним і гнучким інструментом для демонстрації об'ємних моделей товарів у інтернет-магазині на платформі Magento 2. Завдяки інтерактивному перегляду, який дозволяє досліджувати модель у трьох площинах, підтримці сучасних форматів моделей, можливості попереднього перегляду та використанню потужної бібліотеки `model-viewer`, цей модуль покращує користувацький досвід і допомагає збільшити конверсії та зменшити кількість повернень товарів. Сумісність з темою Нува додатково забезпечує високу продуктивність і

швидкість роботи магазину, роблячи цей модуль незамінним для будь-якого сучасного інтернет-магазину.

Серед недоліків модуля варто відзначити кілька ключових аспектів, які можуть вплинути на користувацький досвід і зручність використання.

По-перше, недостатній рівень налаштувань може стати суттєвою проблемою для менеджерів магазину, які потребують більш гнучкого і детального налаштування сцени продуктів. Обмежені можливості налаштувань можуть ускладнити адаптацію модуля до специфічних потреб проєкту.

По-друге, поганий користувацький інтерфейс може значно знизити зручність роботи з модулем. Користувачі модуля можуть не зрозуміти як їм користуватись, що може негативно вплинути на досвід користування як модулем зокрема, так і інтернет-магазином в цілому.

Крім того, модуль має незрозумілий екран завантаження, через що може створитися враження, що модуль завис або не працює належним чином. Відсутність чіткої інформації про стан завантаження може викликати роздратування у користувачів і змусити їх думати, що система не відповідає.

Всі ці фактори можуть негативно вплинути на загальне враження від використання модуля і знизити його ефективність у практичному застосуванні.

Webkul Magento 2 3D Product Configurator

Розширення Webkul 3D Product Configurator для Magento 2 є платним модулем, що надає можливість інтегрувати тривимірний перегляд і налаштування продуктів у інтернет-магазині [7].

Цей модуль дозволяє адміністраторам додавати 3D моделі продуктів, керувати варіаціями товарів і налаштовувати їх відображення на сторінці продукту. Адміністратори можуть додавати різні варіації продуктів, такі як колір, розмір і тип продукту, створюючи опції для продуктів, що дозволяє

покупцям вибирати бажані характеристики товару безпосередньо у модальному вікні з об'ємною моделлю.

Модуль працює з форматами GLTF і GLB, що є одним з найпопулярніших для 3D моделей завдяки їх компактності і швидкому завантаженню. В адмін-панелі адміністратори можуть завантажувати ZIP файли з 3D моделями, налаштовувати варіації і переглядати попередній вигляд моделі перед тим, як застосувати зміни. Крім того, адміністратори можуть додавати окремі зображення для кожної опції варіацій, що допомагає створити реалістичний вигляд продукту.

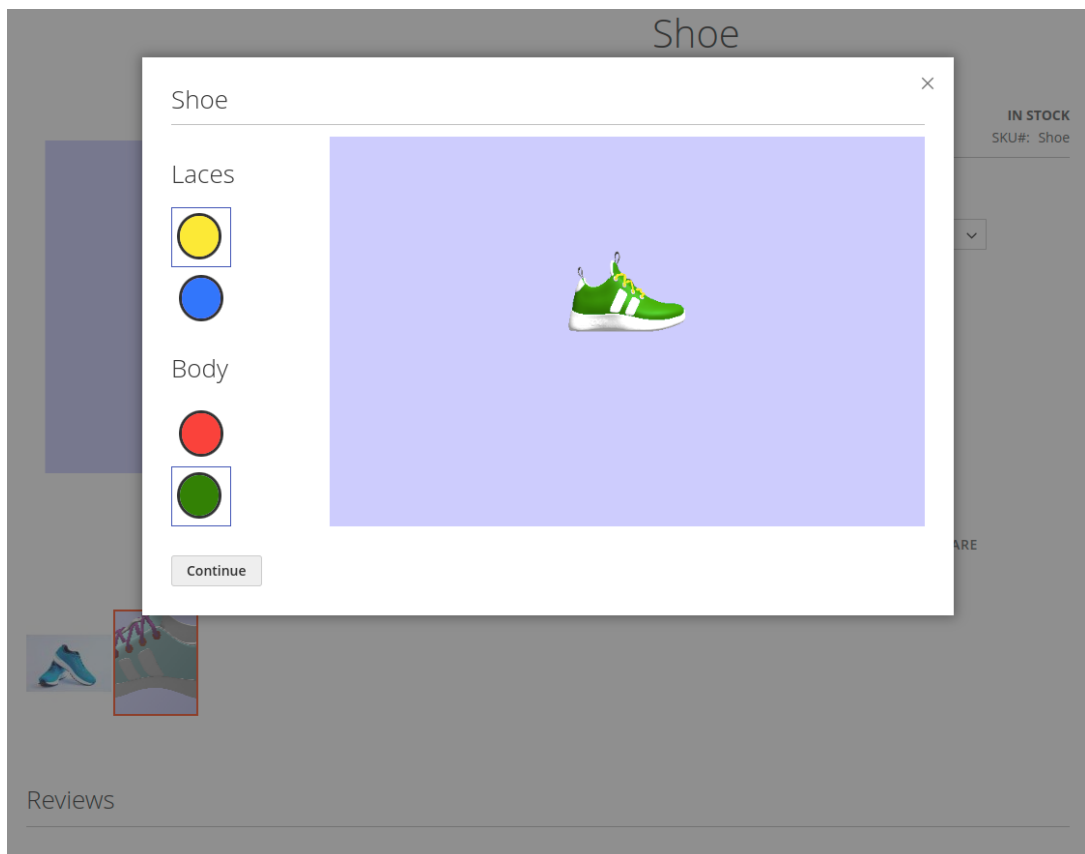


Рисунок 2 — Модальне вікно об'ємної моделі з вибором конфігурації продукту

Основною особливістю є підтримка конфігурованих продуктів, що дозволяє налаштовувати конфігуровані продукти. Інтерактивний 3D перегляд і можливість налаштування продуктів підвищують залученість клієнтів,

допомагаючи їм краще зрозуміти і оцінити продукт перед покупкою. Це також допомагає знизити кількість повернень товарів, оскільки клієнти отримують більш точне уявлення про товар. Використання потужної бібліотеки `model-viewer` забезпечує високу якість відображення і широкий спектр налаштувань для створення реалістичних 3D моделей, що підвищує довіру клієнтів і сприяє збільшенню конверсій і продажів.

Серед недоліків модуля слід підкреслити кілька ключових аспектів, які можуть вплинути на користувацький досвід та ефективність роботи з 3D моделями.

Недостатньо зручний інтерфейс значно знижує зручність та естетичну привабливість модуля. Недоліки в дизайні можуть ускладнити навігацію та роботу з модулем, що негативно позначається на користувацькому досвіді.

Маленьке вікно для рендерингу 3D моделей обмежує можливості перегляду та взаємодії з тривимірними об'єктами. Користувачам може бути складно детально розглянути модель або взаємодіяти з нею через недостатній розмір вікна рендерингу.

Також, погана робота зі світлом на сцені є серйозним недоліком, оскільки належне освітлення є ключовим для відображення деталей та реалістичності 3D моделей.

Webrotate 360 Product Viewer

На відміну від попередніх прикладів, модуль 1.1.1 `Webrotate 360 Product Viewer` реалізує зовсім інший підхід для демонстрації об'ємних моделей продуктів. Розширення дозволяє завантажувати масив зображень продукту з усіх сторін. Під час прокрутки продукту – зображення поступово змінюються, утворюючи тим самим ефект обертання об'ємної моделі [8].

Модуль підтримує різні формати зображень, забезпечуючи гнучкість у використанні і легкість інтеграції з `Magento 2`. Адміністратори можуть швидко налаштовувати і керувати 360-градусними зображеннями продуктів без необхідності глибоких технічних знань.

Одним з переваг модулю є оптимізація завантаження, завдяки чому модуль забезпечує швидке завантаження і відображення зображень, що сприяє покращенню користувацького досвіду і зменшенню часу завантаження сторінок.

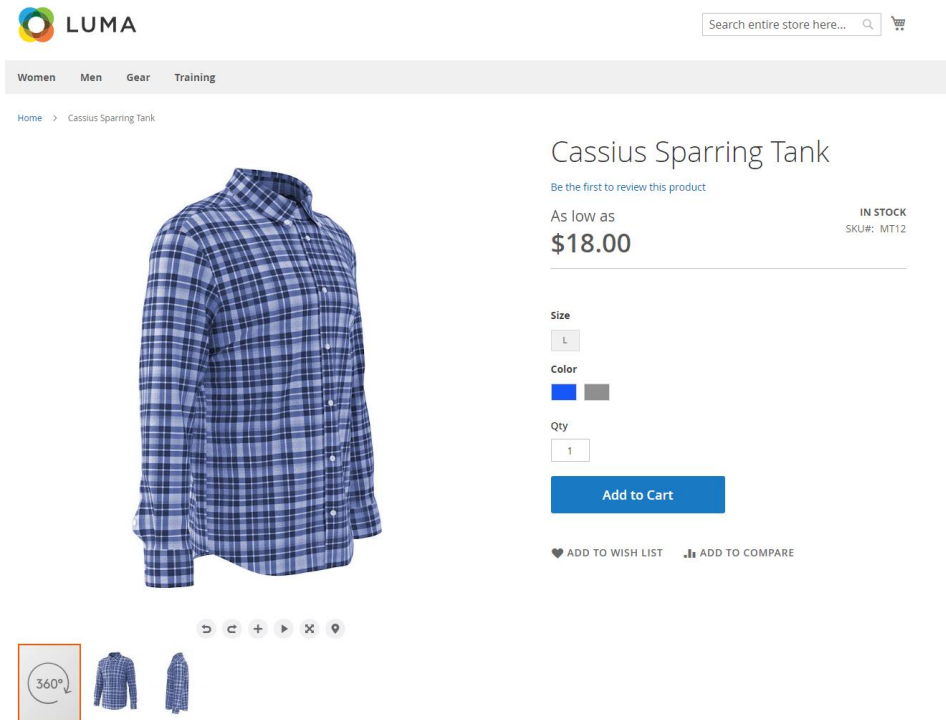


Рисунок 3 — Приклад роботи розширення *WebRotate 360*

Модуль також включає можливість інтеграції з іншими системами управління контентом та платформами електронної комерції, такими як Shopify, OpenCart, PrestaShop, BigCommerce, WordPress та Joomla. Це забезпечує високу сумісність та легкість інтеграції, роблячи WebRotate 360 Product Viewer універсальним рішенням для багатьох бізнесів. Це робить модуль універсальним і зручним інструментом для створення привабливих візуалізацій продуктів.

Загалом, розширення WebRotate 360 є зручним та потужним інструментом з цікавою реалізацією демонстрації моделей продуктів на сайтах електронної комерції Magento 2.

Як висновок можна сказати, що на ринку розширень для демонстрації тривимірних моделей продуктів є достатня кількість продуктів-конкурентів, які пропонують достатній функціонал для поставлених цілей. Однак, кожен з продуктів-конкурентів не позбавлений недоліків, таких як погано продуманий користувацький інтерфейс, недостатня кількість розширень, які підтримуються модулем, або ж відсутність достатньої можливості налаштування сцени безпосередньо через адміністративну панель адміністратора.

Серед недоліків модуля слід виділити кілька важливих аспектів, які впливають на його функціональність та зручність використання.

Відсутність підтримки повноцінних об'ємних моделей товарів обмежує можливість користувачів детально розглянути товар з усіх ракурсів. Це означає, що потенційні покупці не можуть повністю оцінити товар перед покупкою, що знижує ефективність модуля для електронної комерції.

Модуль вимагає завантаження великої кількості зображень для створення тривимірного ефекту, що може значно збільшити час завантаження сторінки і споживання трафіку. Це може негативно вплинути на користувацький досвід, особливо для користувачів з повільним інтернет-з'єднанням.

Також, слід враховувати складність використання модуля через необхідність врахування послідовності фотографій, що значно ускладнює процес налаштування модуля.

1.3 Постановка задачі

Метою кваліфікаційної роботи є розроблення розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції. Дослідження торкається наступних тем:

1. Аналіз вимог: Проведення аналізу для визначення основних вимог та завдань, які має виконувати розширення. Модуль має забезпечувати можливість інтерактивного перегляду тривимірних моделей.
2. Вивчення існуючих продуктів-конкурентів. Аналіз ринку розширень Magento 2, які пропонують сумісний функціонал. Визначення переваг та недоліків продуктів-конкурентів.
3. Вивчення кращих практик створення модулів для Magento 2. Аналіз архітектури розширень, вивчення вимог до модулів.
4. Проектування та розробка серверної частини застосунку. Визначення основних функціональних вимог до серверної частини модуля, розробка основної архітектури застосунку.
5. Розробка API. Визначення основних задач запитів до інтерфейсу програмних додатків. Розробка кінцевих точок API та їх обробка.
6. Розробка інтерфейсу користувача. Реалізація основних компонентів користувацького інтерфейсу модуля на сторінці продукту сайту електронної комерції.
7. Розробка персоналізації сцени. Проектування та створення можливості запису й зчитування налаштувань сцени з адміністративної панелі.
8. Тестування працездатності програмного забезпечення. Виконання тестування основного функціоналу модуля, виправлення помилок та дефектів.

Результатом роботи є модуль для демонстрації об'ємних моделей продуктів, який було розроблено для системи керування вмісту Magento 2.

2 ДОСЛІДЖЕННЯ ПРОГРАМНИХ ЗАСОБІВ РЕАЛІЗАЦІЇ МОДУЛЯ

2.1 Аналіз бібліотек для демонстрації об'ємних моделей

Існує декілька сучасних бібліотек для реалізації перегляду об'ємних моделей у вікні браузера. Кожна з бібліотек має різний рівень продуктивності, налаштувань, можливостей та функцій. На момент проведення порівняльного аналізу бібліотек актуальними варіантами були наступні:

Three.js

Бібліотека Three.js є однією з найпопулярніших бібліотек для створення 3D графіки у вікні браузера. Пакет був створений для роботи з WebGL, що дозволяє відображати 3D сцени без використання додаткових плагінів [9]. Бібліотека підтримує великий набір форматів 3D моделей, таких як OBJ, STL, FBX, glTF, та багато інших, що робить її надзвичайно гнучкою. З її допомогою можна легко реалізувати різноманітні анімації, що додає динамічності візуалізації.

Three.js пропонує широкий набір матеріалів і шейдерів, що дозволяють створювати реалістичні текстури та ефекти. Однією з переваг є потужна система освітлення та тіней, яка додає реалістичності тривимірним сценам. Бібліотека має велику спільноту та детальну документацію, що полегшує навчання та використання. Завдяки своїй багатофункціональності, Three.js використовується в різних галузях, від ігор до наукових візуалізацій. У підсумку, серед переваг бібліотеки можна виділити наступні:

1. Підтримка різних форматів 3D-моделей;
2. Підтримка складних анімацій тривимірних моделей;
3. Сумісність з різними матеріалами та шейдерами;
4. Можливість додавання різних видів освітлення на сцену та відтворення тіней.
5. Велика спільнота та гарна документація

Однак, бібліотека, також, має наступні недоліки:

1. Високий поріг входження;
2. Помірна швидкість завантаження об'ємних моделей.

Babylon.js

Babylon.js є потужною бібліотекою для створення 3D графіки та ігор у браузері, яка використовує WebGL. Вона забезпечує багатий набір функцій для реалізації фізичних взаємодій, що робить її ідеальним варіантом для створення інтерактивних ігор у веб-браузері. Babylon.js підтримує різні анімаційні можливості, які дозволяють легко додати динамічності до 3D об'єктів [10].

Бібліотека має розширені можливості для роботи з матеріалами та шейдерами, що дозволяє створювати фотореалістичні сцени. Однією з унікальних особливостей Babylon.js є вбудована підтримка віртуальної та доповненої реальності, що дозволяє створювати інтерактивні VR і AR додатки безпосередньо у браузері. Бібліотека має непогану документацію з прикладами, що допомагає швидко почати працювати з нею.

Завдяки високій продуктивності, Babylon.js може обробляти складні сцени та моделі без значного навантаження на систему.

Серед переваг бібліотеки слід підкреслити:

1. Вбудована підтримка фізичних двигунів для реалістичної взаємодії з 3D об'єктами;
2. Підтримка анімації об'ємних моделей;
3. Вбудована підтримка VR та AR;
4. Висока продуктивність.

З іншої сторони, недоліками бібліотеки є:

1. Надмірний функціонал для поставленої задачі
2. Відсутність великої спільноти, що значно ускладнює процес написання коду новачкам.
3. Підтримка невеликої кількості форматів об'ємних моделей

Враховуюче поставлене завдання було обрано бібліотеку Three.js. Вона має одну з найбільших і найактивніших спільнот серед 3D бібліотек, що забезпечує доступ до великої кількості прикладів, навчальних матеріалів та документації, полегшуючи процес навчання та вирішення можливих проблем.

Пакет підтримує широкий спектр форматів 3D моделей, таких як OBJ, STL, FBX, glTF, що робить її вдалим вибором для демонстрації продуктів на сайтах електронної комерції, адже заздалегідь невідомо який формат об'ємної моделі буде використано продавцем інтернет-магазину.

Бібліотека пропонує потужний набір функцій для роботи з освітленням, матеріалами, шейдерами, та тінями, що дозволяє створювати високоякісні та реалістичні сцени.

2.2 Порівняльний аналіз засобів збірки застосунка

Засоби збірки є важливими інструментами в процесі розробки програмного забезпечення, оскільки допомагають забезпечити ефективність та швидкість розробки продукту. Засоби збірки автоматично компілюють вихідний код на різних мовах програмування і мінімізують його, зменшуючи розмір файлів і захищаючи від несанкціонованого використання. Також, вони можуть об'єднувати кілька файлів у один або кілька більших файлів, зменшуючи кількість запитів до сервера при завантаженні веб-сторінок.

Інструменти збірки можуть виконувати статичний аналіз коду для виявлення помилок та забезпечення дотримання стандартів кодування. Вони допомагають оптимізувати продуктивність кінцевого додатка і забезпечують гарячий модульний перезапуск, що значно пришвидшує розробку програмного продукту.

Під час розробки кваліфікаційної роботи було проведено порівняльний аналіз засобів збірки з метою виявлення найбільш ефективного засобу для поставленої задачі.

Vite

Vite.js є сучасною системою збірки, яка використовує ES модулі для забезпечення швидкого початкового завантаження та ефективного гарячого модульного перезапуску. Переваги Vite включають швидке початкове завантаження, оскільки не потребується попередня компіляція всього коду, що значно скорочує час старту додатку [11].

Гарячий модульний перезапуск забезпечує миттєве оновлення змін без перезавантаження сторінки, що підвищує ефективність розробки. Мінімальна конфігурація дозволяє швидко почати роботу з Vite без необхідності складних налаштувань. Підтримка популярних фреймворків, таких як Vue.js та React робить Vite універсальним інструментом.

Серед переваг Vite варто відмітити такі:

1. Швидке початкове завантаження. Замість того, щоб попередньо компілювати весь проєкт, Vite миттєво починає обслуговувати вихідні файли JavaScript як модулі ES. Це значно зменшує час початкового завантаження додатка, дозволяючи розробникам швидше бачити результати своєї роботи.
2. Миттєва робота гарячого модульного перезапуску. Однією з ключових переваг Vite є його блискавично швидкий HMR, який дозволяє автоматично оновлювати змінені модулі без повного перезавантаження сторінки. Це значно підвищує ефективність розробки, оскільки розробники можуть миттєво бачити свої зміни в реальному часі.
3. Мінімальна конфігурація. На відміну від традиційних систем збірки, які часто вимагають складних файлів конфігурації, Vite постачається з готовими до використання налаштуваннями.
4. Підтримка різних фреймворків. Система має вбудовану підтримку для популярних фреймворків, таких як Vue.js або React. Це робить його універсальним інструментом для різних типів проєктів, незалежно від обраного фреймворка.

Webpack

Webpack — це система збірки для JavaScript додатків, що дозволяє пакувати всі ресурси додатка в один або кілька файлів. Однією з головних переваг Webpack є його висока гнучкість і налаштування, що дозволяє адаптувати інструмент під будь-які потреби проєкту [12].

Велика екосистема плагінів забезпечує широкі можливості розширення функціональності. Підтримка динамічного імпорту дозволяє оптимізувати продуктивність додатка. Висока популярність і велика спільнота забезпечують доступ до великої кількості ресурсів, прикладів і підтримки.

Однак, складна конфігурація може бути викликом для початківців, що вимагає значного часу на налаштування та освоєння. Також початковий збір великих проєктів може бути повільним, що впливає на продуктивність.

Серед основних переваг Webpack варто виділити наступні:

1. Висока гнучкість конфігурації. У той час як Vite спрямований на мінімальну конфігурацію для швидкого початку роботи, Webpack не пропонує такого рівня деталізації та гнучкості в налаштуванні процесу збірки.
2. Велика екосистема плагінів. Webpack має одну з найбільших екосистем плагінів серед інструментів збірки. Це дозволяє розробникам легко додавати нові можливості, такі як оптимізація зображень або автоматичне вставлення CSS у HTML.
3. Добре працює з великими проєктами. Webpack добре підходить для великих і складних проєктів завдяки своїй здатності обробляти великі обсяги коду та численні залежності з високою гнучкістю налаштувань. Він дозволяє створювати складні конфігурації, які можуть включати кілька точок входу, складні правила обробки файлів і багаторівневу оптимізацію.

Rollup

Rollup було створено для пакування бібліотек JavaScript, створюючи ефективний і мінімізований код. Його головною перевагою є те, що він підтримує Tree Shaking, що дозволяє виключати невикористані модулі з кінцевої збірки.

Мінімальна конфігурація робить Rollup простим у налаштуванні та використанні. Система чудово підходить для створення бібліотек завдяки своїм оптимізаційним можливостям [13]. Однак, у порівнянні з іншими системами, Rollup має меншу кількість плагінів, що може обмежувати його функціональність. Інтеграція з великими додатками може бути складнішою, що вимагає додаткових налаштувань.

Основні переваги Rollup включають в себе:

1. Високоякісний вихідний код. Система створює більш чистий і компактний код, який є більш читабельним і легше підтримується розробниками.
2. Tree Shaking. Ця концепція дозволяє значно зменшити розмір кінцевих збірок, виключаючи невикористані частини коду.
3. Модульна архітектура. Підходить для створення бібліотек, де кожен модуль може імпортуватися окремо.

Під час проведеного аналізу засобів збірки та аналізу порівняння параметрів, які наведено у таблиці 1 було досліджено, що найвдалішим варіантом для поставленої мети кваліфікаційної роботи є система збірки Vite, тому що саме ця система забезпечує швидке початкове завантаження, гарячий модульний перезапуск та мінімальну конфігурацію, яка дозволяє швидко почати працювати з системою збірки.

Швидка робота гарячого модульного перезапуску дозволяє миттєво бачити зміни без перезавантаження сторінки, що підвищує ефективність розробки. Мінімальна конфігурація спрощує налаштування і дозволяє швидко розпочати роботу над проєктом.

Таблиця 1 — Порівняння характеристик засобів збірки застосунків
Vite, Webpack та Rollup

	Vite	Webpack	Rollup
Основне призначення	Швидка розробка і збірка сучасних SPA	Універсальний збирач модулів	Збірка бібліотек і додатків
Конфігурація	Проста і мінімальна	Гнучка, але складна	Проста, але менш потужна
Швидкість збірки	Дуже швидка (до 10 разів швидше за Webpack)	Помірна (залежить від конфігурації)	Висока, але може бути повільніше за Vite
Час HMR	< 50 мс	100-300 мс	50-200 мс
Tree-shaking	Вбудований	Вбудований	Вбудований
Спільнота	Активна	Дуже велика і активна	Активна
Розмір бандла	Оптимізований (зазвичай менше 500KB)	Залежить від конфігурації	Оптимізований (може бути менше 500KB)
Навантаження на процесор	Низьке	Може бути високим	Низьке

Враховуючи масштаб проєкту та його мету було вирішено використовувати Vite, тому що він поєднує у собі саме ті переваги, які найбільш вигідно можна використовувати під час розробки кваліфікаційної роботи.

2.3 Аналіз існуючих систем керування вмісту для сайтів електронної комерції

Вибір системи керування вмістом для інтернет-магазину є критичним для успіху бізнесу. Існує багато платформ, які пропонують різні можливості для електронної комерції, але три з найпопулярніших - це Shopify, Magento 2 та BigCommerce. Кожна з цих платформ має свої особливості, переваги та недоліки, що робить їх вдалим вибором для різних типів бізнесу. У цьому

розділі буде розглянуто найбільш популярні на сьогоднішній день системи керуванням вмісту.

Magento 2

Magento 2 є однією з найпотужніших і найгнучкіших платформ для електронної комерції. Вона пропонує багатий набір функціональних можливостей, які можна легко налаштовувати та розширювати за допомогою модулів. Це робить Magento 2 ідеальним вибором для великих магазинів з складними потребами, що вимагають високого рівня налаштувань і персоналізації [14].

Однією з ключових переваг Magento 2 є її здатність підтримувати велику кількість товарів і обробляти великі обсяги трафіку, що робить її вдалим вибором для масштабних інтернет-магазинів. Платформа також забезпечує високу гнучкість у налаштуванні бізнес-процесів, управлінні каталогами товарів, маркетинговими кампаніями та аналітикою.

Система має велику спільноту розробників і користувачів, що створює багатий ресурс для вирішення будь-яких проблем і питань. Містить багато документації, форуми і підтримку. Це забезпечує доступ до великої кількості ресурсів для вирішення будь-яких проблем, які можуть виникнути під час використання платформи.

Magento 2 є однією з найбільш гнучких і розширюваних платформ для електронної комерції, що робить її ідеальним вибором для інтеграції складних модулів, таких як відображення тривимірних моделей товарів. Magento 2 дозволяє розробникам легко створювати та інтегрувати власні модулі, використовуючи різноманітні інструменти та фреймворки. Це забезпечує високий рівень контролю над усіма аспектами роботи модуля, включаючи збереження, рендеринг та взаємодію з моделями.

Платформа також підтримує використання сучасних технологій, таких як WebGL і Three.js, що дозволяє забезпечити високоякісний рендеринг тривимірних моделей без значних затримок. Magento 2 вимагає значних

технічних знань для налаштування і підтримки, але це компенсується високою гнучкістю та можливістю масштабування.

Проте Magento 2 також має свої недоліки. Навіть безкоштовна версія може вимагати значних витрат на хостинг, розробку і підтримку. Magento 2 вимагає високого рівня технічних знань для установки та обслуговування.

Крім того, Magento 2 має високі вимоги до ресурсів, що може призвести до додаткових витрат на серверне обладнання і хостинг.

Незважаючи на ці виклики, Magento 2 є потужним інструментом та пропонує високу гнучкість, масштабованість і розширюваність, що робить його ідеальним для великих магазинів з високими вимогами.

До переваг Magento 2 слід віднести наступні:

- Гнучкість і розширюваність: Magento 2 пропонує багатий набір функціоналу та можливість налаштування і розширення через модулі. Це робить його ідеальним для великих магазинів з складними потребами.
- Підтримка великої кількості товарів: Magento 2 може легко обробляти тисячі товарів, що робить його вдалим вибором для великих магазинів.
- Професійна підтримка та велика спільнота: Містить багато документації, форуми і платну підтримку від Adobe. Це забезпечує доступ до великої кількості ресурсів для вирішення будь-яких проблем.

Серед недоліків Magento 2 варто відмітити наступні:

- Висока вартість: Вартість професійної версії (Magento Commerce) є високою, що може бути перешкодою для малих і середніх підприємств.
- Високі вимоги до ресурсів: Magento 2 вимагає потужних серверних ресурсів для ефективної роботи, що може збільшити витрати на хостинг.

Shopify

Shopify відомий своєю простотою використання, що робить його популярним вибором серед підприємців, які не мають технічного досвіду. Ця платформа пропонує інтуїтивно зрозумілий інтерфейс, завдяки якому налаштування магазину є швидким і зручним [15].

Всі інструменти, необхідні для запуску інтернет-магазину, вже вбудовані в Shopify, що значно спрощує процес старту. Однією з найбільших переваг Shopify є те, що це повністю хмарне рішення. Користувачам не потрібно турбуватися про хостинг, технічне обслуговування або безпеку, оскільки всі ці аспекти обслуговуються командою Shopify.

Крім того, існує велика кількість навчальних матеріалів, вебінарів і форумів спільноти, що допомагають користувачам швидко освоїти платформу та вирішити будь-які питання, які можуть виникнути.

Однак Shopify має і свої недоліки. Одним з основних є постійна абонентська плата, яка може бути значною залежно від вибраного тарифного плану. Це може стати проблемою для малих підприємств з обмеженим бюджетом.

Крім того, можливості налаштування у Shopify обмежені в порівнянні з іншими платформами. Це може обмежити можливість реалізації складних проєктів, інтеграції важких розширень або специфічних бізнес-процесів. Багато розширень та тем у Shopify є платними, що може збільшити загальні витрати на підтримку магазину.

Незважаючи на ці недоліки, Shopify залишається одним з найпопулярніших рішень для електронної комерції завдяки своїй простоті, надійності та широким можливостям підтримки.

Головними перевагами Shopify є:

- Простота використання: Shopify відомий своїм інтуїтивно зрозумілим інтерфейсом, що робить його ідеальним для користувачів, які не мають технічного досвіду. Процес

налаштування магазину є швидким і простим, завдяки вбудованим інструментам і шаблонам.

- **Хмарне рішення:** Shopify є повністю хмарним рішенням, що означає, що користувачам не потрібно турбуватися про хостинг, технічне обслуговування або безпеку. Це забезпечує високу надійність і масштабованість.
- **Підтримка:** Shopify надає цілодобову підтримку через різні канали, включаючи чат, електронну пошту та телефон. Крім того, є велика кількість навчальних матеріалів, вебінарів і форумів спільноти.

Серед недоліків варто відмітити наступні:

- **Обмежені можливості налаштування:** У порівнянні з Magento, Shopify має менш гнучкі можливості налаштування, що може обмежити складні проєкти та ускладнює інтеграцію важких розширень у систему.
- **Додаткові витрати:** Багато розширень та тем у Shopify є платними, що може збільшити загальні витрати на підтримку магазину.

BigCommerce

BigCommerce є ще однією популярною платформою для електронної комерції, яка відома своєю простотою використання і потужними вбудованими інструментами.

Платформа надає інтуїтивно зрозумілий інтерфейс, що робить її легкою у використанні для підприємців без технічного досвіду. Процес налаштування магазину є досить простим і швидким завдяки широкому набору вбудованих функцій і шаблонів [16].

BigCommerce пропонує масштабоване рішення, яке може обробляти велику кількість товарів і високі обсяги трафіку, що робить його підходящим для швидкозростаючих бізнесів. Однією з ключових переваг BigCommerce є

його здатність забезпечувати стабільну роботу магазину навіть при великій кількості відвідувачів, що важливо для бізнесів, які планують розширюватися.

Крім того, є велика база знань та навчальні матеріали, які допомагають користувачам освоїти платформу і максимально використувати її можливості.

Однак BigCommerce має і свої недоліки. Можливості налаштування у BigCommerce можуть бути обмеженими у порівнянні з Magento 2. Хоча платформа надає багато вбудованих функцій, складніші проекти можуть вимагати додаткових платних модулів та розширень, що може збільшити загальні витрати.

BigCommerce підтримує використання сторонніх інструментів і бібліотек для рендерингу тривимірних моделей, але можливості налаштування серверного середовища для оптимізації продуктивності можуть бути обмеженими. Це може вплинути на швидкість і якість рендерингу тривимірних моделей, що є критично важливим для розширення.

Незважаючи на ці недоліки, BigCommerce залишається популярним вибором для бізнесів, які шукають потужне і просте у використанні рішення для електронної комерції. Платформа забезпечує баланс між простотою використання і функціональністю, що робить її гарним вибором для підприємств, які швидко ростуть і потребують стабільної платформи з хорошою підтримкою.

Серед переваг BigCommerce є:

- Масштабованість: BigCommerce здатний обробляти великий обсяг товарів і трафіку, що робить його підходящим для швидкозростаючих бізнесів.
- Вбудовані функції: Платформа має багато вбудованих функцій, таких як інструменти для SEO, маркетингу, управління запасами та аналітики.

- Сучасні шаблони та теми: BigCommerce пропонує різноманітні сучасні шаблони та теми, які можна легко налаштовувати під специфічні потреби бренду.

Однак, серед недоліків варто підкреслити наступні:

- Обмежені можливості налаштування: Хоча BigCommerce пропонує багато вбудованих функцій, можливості налаштування можуть бути обмеженими у порівнянні з більш гнучкими платформами, такими як Magento.
- Менші можливості для складних адаптацій: У порівнянні з Magento, BigCommerce має менш розширені можливості для складних конфігурацій, що може обмежити реалізацію специфічних бізнес-процесів і унікальних функцій.
- Обмежений контроль над сервером: Як хмарне рішення, BigCommerce не надає користувачам повного контролю над сервером, що може бути обмежуючим фактором для бізнесів, які потребують специфічних налаштувань серверного середовища для оптимізації продуктивності.

Виходячи з проведеного порівняльного аналізу платформ для електронної комерції, можна сказати, що кожна з розглянутих систем - Shopify, Magento 2 та BigCommerce - має свої унікальні переваги та недоліки. Кожна платформа здатна забезпечити необхідний рівень функціональності та підтримки для різних типів бізнесу. Проте, враховуючи специфічні потреби кваліфікаційної роботи, вибір було зроблено на користь Magento 2.

Magento 2 є найбільш гнучкою і потужною платформою серед розглянутих. Вона забезпечує високий рівень налаштувань та розширюваності, що є критичним для інтеграції складних модулів, таких як відображення тривимірних моделей товарів. Можливість підтримки великої кількості товарів і обробки великих обсягів трафіку робить Magento 2 ідеальним вибором.

Отже, обираючи платформу для якої буде розроблено розширення, акцент було зроблено саме на масштабованості, гнучкості та зручності розробки розширень. Magento 2 найкраще відповідає цим вимогам, що робить її оптимальним вибором для інтеграції модуля відображення тривимірних моделей товарів. Висока гнучкість і можливість налаштування платформи дозволять реалізувати всі необхідні функціональні можливості і забезпечити стабільну та ефективну роботу модуля. Таким чином, Magento 2 є найкращим вибором для розробки розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції.

З проведеного аналізу технологій і платформ для реалізації проєкту витікає, що кожна з розглянутих систем і бібліотек має свої унікальні переваги та недоліки. Однак, враховуючи специфічні потреби кваліфікаційної роботи, найкращим вибором для кожної категорії є: Three.js для рендерингу тривимірних моделей, Vite як засіб збірки та Magento 2 як платформа для електронної комерції.

Three.js є найбільш відповідною бібліотекою для рендерингу тривимірних моделей завдяки своїй простоті використання, широкому набору функцій і великій спільноті. Бібліотека підтримує всі необхідні функції для реалізації проєкту, включаючи рендеринг тривимірних об'єктів, роботу зі світлом, тінями та матеріалами. Ця бібліотека має широке визнання в спільноті розробників, що забезпечує доступ до великої кількості навчальних матеріалів, прикладів та підтримки.

Серед засобів збірки було обрано Vite завдяки його швидкому початковому завантаженню, ефективному гарячому модульному перезапуску і мінімальній конфігурації. У порівнянні з Webpack і Rollup, Vite надає найшвидший і найзручніший досвід розробки. Webpack, хоча і є потужним і гнучким інструментом, може бути складним у налаштуванні і часто вимагає значного часу для початкової збірки проєкту. Rollup, з іншого боку, відмінно підходить для пакування бібліотек завдяки своїм оптимізаційним можливостям, але не надає тієї ж зручності для розробки веб-застосунків, що

і Vite. Вибір Vite дозволяє нам зосередитися на розробці функціональності без зайвих затримок, що є критично важливим для виконання поставленої задачі.

Як платформу електронної комерції було обрано Magento 2, оскільки вона пропонує найбільшу гнучкість і можливість налаштування серед розглянутих варіантів. Одним з головних її переваг є висока гнучкість та масштабованість, що дозволяє легко й зручно інтегрувати нові розширення з системою. Magento 2 надає розробникам великий контроль над усіма аспектами роботи модуля, включаючи збереження, рендеринг та взаємодію з моделями, що робить її оптимальним вибором для проєкту.

Отже, враховуючи всі технічні та функціональні вимоги для розробки було обрано Three.js для рендерингу тривимірних моделей, Vite як засіб збірки і Magento 2 як платформу для електронної комерції. Такий вибір дозволяє нам забезпечити високу якість, ефективність та масштабованість проєкту, що є важливим фактором для успіху в довгостроковій перспективі. Ці рішення забезпечують найкращий баланс між функціональністю, зручністю використання та можливістю розширення, що робить їх ідеальними варіантами для реалізації поставленої задачі.

3 РОЗРОБКА РОЗШИРЕННЯ ДЛЯ САЙТУ ЕЛЕКТРОННОЇ КОМЕРЦІЇ

3.1 Огляд предметної сфери

Візуальне представлення товарів в інтернет-магазинах є одним з найважливіших факторів, що впливають на рішення покупців щодо придбання товару. У сучасному світі, де онлайн-шопінг стає все більш популярним, забезпечення якісного та детального візуального контенту є ключовим елементом успішної стратегії продажів.

Інтеграція тривимірних моделей товарів на сайти електронної комерції надає покупцям можливість детально розглянути продукт з усіх боків, що створює відчуття реальної взаємодії з товаром. Це дозволяє покупцям краще розуміти характеристики і особливості продукту, що вони збираються придбати, що, в свою чергу, знижує ризик розчарування і повернень.

Тривимірні моделі дозволяють покупцям обертати модель, змінювати кут огляду, збільшувати або зменшувати масштаб, що значно покращує користувацький досвід. Впровадження об'ємних моделей особливо важливе для складних або високотехнологічних товарів, де кожна деталь може бути вирішальною для прийняття рішення про покупку.

Крім того, використання 3D моделей підвищує довіру покупців до товару та магазину, оскільки демонструє інвестиції компанії в сучасні технології для покращення клієнтського досвіду. Це не тільки покращує взаємодію з клієнтами, але й створює конкурентну перевагу на ринку.

З технічної точки зору, використання бібліотеки Three.js та технологій WebGL дозволяє реалізувати високоякісний рендеринг тривимірних моделей без втрати продуктивності. Це забезпечує швидке завантаження і відображення моделей на різних пристроях і браузерах, роблячи процес інтеграції доступним навіть для невеликих інтернет-магазинів на основі Magento 2.

Це робить 3D візуалізацію важливим інструментом у стратегії електронної комерції, сприяючи зростанню продажів та зміцненню позицій на ринку. Таким чином, впровадження розширення для відображення тривимірних моделей товарів у інтернет-магазині є стратегічно важливим кроком, який дозволяє забезпечити більш інтерактивний і привабливий користувацький досвід, підвищуючи задоволеність клієнтів та ефективність продажів.

3.2 Архітектура системи

Архітектура модуля складається з двох частин – серверної, яка використовує широкі можливості системи керуванням вмісту Magento 2, та клієнтської, яка поєднує у собі стандарти платформи для відображення елементів інтерфейсу та бібліотеки three.js, яка використовується для виконання рендерингу моделей.

Бекенд обробляє зберігання та управління даними про тривимірні моделі, використовуючи моделі та контролери для додавання, редагування та видалення моделей через адміністративну панель. Ці дані зберігаються у базі даних Magento та включають інформацію про файли моделей і параметри рендерингу. EAV-атрибути визначають нові атрибути товарів для зберігання налаштувань 3D моделей.

Дані про тривимірні моделі товарів, такі як шлях до моделі у каталозі media, налаштування оточення, інформація про активність моделі та колір фону, зберігаються в Magento 2 за допомогою відповідних EAV-атрибутів.

EAV модель дозволяє гнучко додавати нові атрибути до товарів без зміни схеми бази даних. Це забезпечує зручне зберігання та управління додатковою інформацією, необхідною для рендерингу тривимірних моделей, дозволяючи адміністраторам магазину легко керувати цими налаштуваннями через адміністративну панель. Завдяки цьому, інтеграція та управління тривимірними моделями стають простішими і ефективнішими.

Приклад створення EAV атрибуту у Magento 2 наведено у лістингу 1.

Лістинг 1 Створення EAV атрибуту для збереження даних про файл моделі.

```
public function apply(): void
{
    $eavSetup = $this->eavSetupFactory->create(['setup'
=> $this->moduleDataSetup]);

    $attribute = [
        'group' => '3D Model',
        'type' => 'varchar',
        'label' => '3D Model File',
        'input' => 'file',
        'backend' => File::class,
        'frontend' => '',
        'class' => '',
        'source' => '',
        'global' =>
ScopedAttributeInterface::SCOPE_GLOBAL,
        'visible' => true,
        'required' => false,
        'user_defined' => true,
        'default' => '',
        'searchable' => false,
        'filterable' => false,
        'comparable' => false,
        'visible_on_front' => false,
        'unique' => false,
        'apply_to' =>
'simple,configurable,grouped,virtual,bundle,downloadable',
        'used_in_product_listing' => true
    ];

    $eavSetup->addAttribute(Product::ENTITY,
'volume_model', $attribute);
}
```

Код, який наведено у лістингу 1 додає новий атрибут «volume_model» до продуктів в Magento 2 за допомогою скрипта налаштування даних модуля. Атрибут призначений для завантаження даних про файл тривимірних моделей продуктів. Нижче наведено опис роботи методу apply, та короткий опис кожного з параметрів EAV атрибуту.

Метод `apply` виконує основну роботу по додаванню нового атрибута до продуктів.

Створюється об'єкт `EavSetup` за допомогою фабрики `eavSetupFactory` та передається налаштування `moduleDataSetup`.

Визначається масив параметрів для нового атрибута `volume_model`, який буде використовуватися для зберігання файлів тривимірних моделей.

Переглянемо масив параметрів, які використовуються для створення нового EAV атрибуту:

1. `group` — Група атрибутів, до якої належить новий атрибут.
2. `type` — Тип даних (в даному випадку, `varchar`).
3. `label` — Назва атрибута, яка буде відображатися в адмін-панелі.
4. `input` — Тип введення (файл).
5. `backend` — Клас бекенду для обробки файлів (вказується клас `File`).
6. `global` — Сфера застосування атрибута (глобальна).
7. `frontend` — Клас фронтенду, який обробляє відображення атрибута на сторінці продукту.
8. `class` — Додатковий CSS клас для поля введення атрибута.
9. `source` — Джерело значень для атрибута, якщо він має тип введення `select`.
10. `visible` — Вказує, чи буде атрибут видимим в адмін-панелі.
11. `required` — Вказує, чи є атрибут обов'язковим для заповнення.
12. `user_defined` — Вказує, чи створений атрибут користувачем.
13. `default` — Значення за замовчуванням для атрибута.
14. `searchable` — Вказує, чи можна виконувати пошук за цим атрибутом.
15. `filterable` — Вказує, чи можна фільтрувати продукти за цим атрибутом.
16. `comparable` — Вказує, чи можна порівнювати продукти за цим атрибутом.

17. `visible_on_front` — Вказує, чи буде атрибут видимим на фронтенді.
18. `unique` — Вказує, чи має атрибут унікальні значення для кожного продукту.
19. `apply_to` — Вказує, до яких типів продуктів застосовується атрибут.
20. `used_in_product_listing` — Вказує, чи буде атрибут використовуватися у списках продуктів.

Метод `addAttribute` використовується для додавання атрибута до сутності продукту (`Product::ENTITY`) з кодом атрибута `volume_model` та визначеними параметрами.

На фронтенді бібліотека `Three.js` забезпечує рендеринг моделей у браузері. JavaScript модуль відповідає за інтеграцію `Three.js` з `Magento 2`, ініціалізацію сцени, завантаження моделей та їх відображення на сторінці продукту. HTML шаблони визначають структуру сторінки продукту та місце для тривимірної моделі.

Коли адміністратор завантажує модель через панель управління `Magento`, дані зберігаються у базі даних. Під час відкриття модального вікна з об'ємною моделлю на сторінці, продукту фронтенд запитує ці дані, після чого відбувається ініціалізація сцени за допомогою `Three.js`, яка створює сцену, камеру, світло і завантажує тривимірну модель. Користувач може взаємодіяти з моделлю, обертати її, змінювати кут огляду та масштаб, отримуючи багатший досвід взаємодії з товаром. Така архітектура дозволяє інтегрувати 3D моделі в `Magento 2` ефективно і з високою продуктивністю, забезпечуючи сучасний та інтерактивний користувацький досвід.

Отримання даних про модель з серверної частини відбувається за допомогою API запитів. Під час відкриття модального вікна на стороні клієнта відбувається запит на отримання всіх даних про об'ємну модель, після чого дані передаються у скрипт обробника та відбувається ініціалізація

відповідного завантажувача. Приклад відповіді на запит для отримання конфігурації можна побачити у лістингу 2.

В залежності від даних про модель, які були отримані у API запиті, відбувається ініціалізація відповідного завантажувача моделі, тому що для кожного формату моделі використовується свій власний завантажувач.

Після визначення формату моделі та вибору відповідного завантажувача, бібліотека Three.js обробляє модель і інтегрує її в тривимірну сцену. Цей процес включає створення сцени, додавання камери, налаштування освітлення та застосування матеріалів для моделі, щоб забезпечити її реалістичне відображення. Завантажувач відповідає за обробку даних моделі, включаючи геометрію, матеріали, текстури та анімації, якщо такі є.

Лістинг 2 Приклад відповіді на запит для отримання конфігурації

```
{
  "volume_model_enabled": "1",
  "volume_model_executive_file":
  "{ \"dirname\": \"\\home\\devserhiiv\\www\\m1.devserhiiv.m8.wdgtest.com\\public_html\\pub\\media\\wg\\models\\20\", \"basename\": \"scene.glTF\", \"extension\": \"glTF\", \"filename\": \"scene\" }\",
  "volume_model": "wg/models/20",
  "volume_model_environment":
  "https://m1.devserhiiv.m8.wdgtest.com/static/version1718186177/frontend/Magento/luma/en_US/Whidegroup_VolumetricModels/environment/mall/texture.hdr",
  "volume_model_background": "#FFFFFF",
  "fov_value": "60",
  "enable": "1",
  "rotation_enable": "0",
  "button_displaying_mode": "both",
  "environment": "mall"
}
```


3.3 Функціональні вимоги системи

Розширення для відображення тривимірних моделей товарів у Magento 2 має забезпечувати широкий спектр функціональності, яка дозволяє як адміністраторам, так і кінцевим користувачам ефективно взаємодіяти з тривимірними моделями товарів. В основі цієї системи лежить можливість завантаження, зберігання, налаштування та відображення тривимірних моделей на сторінках продуктів в інтернет-магазині.

Нижче наведені функціональні вимоги до розширення:

1. Можливість завантаження тривимірних моделей
 - Адміністратор має можливість завантажувати тривимірні моделі товарів через адміністративну панель Magento 2.
 - Підтримуються формати тривимірних моделей, такі як glTF, glb та fbx.
 - Після завантаження модель зберігається у медіа-каталозі Magento 2.
2. Зберігання даних про моделі
 - Дані про моделі, включаючи шлях до файлу, налаштування оточення, колір фону та інформацію про активність моделі мають зберігатися за допомогою EAV-атрибутів до бази даних.
 - Адміністратор повинен мати можливість змінювати ці налаштування через адміністративну панель.
3. Відображення тривимірних моделей
 - Тривимірні моделі товарів мають відображатися на сторінках продуктів в інтернет-магазині.
 - Користувачі повинні мати можливість взаємодіяти з моделями: обертати, масштабувати та змінювати кут огляду.
 - Модель повинна завантажуватися і рендеритися без значних затримок.

4. Ініціалізація завантажувачів моделей

- Система повинна автоматично визначати формат моделі і ініціалізувати відповідний завантажувач.
- Завантажувачі повинні правильно обробляти дані моделі, включаючи геометрію, текстури та анімації.

5. Налаштування рендерингу

- Адміністратор може налаштувати параметри, такі як кут огляду камери, автоматичне обертання моделей, налаштування оточення.
- Система повинна зберігати ці налаштування і застосовувати їх при завантаженні моделі.

6. Сумісність з пристроями та браузерами

- Модуль повинен коректно відображатися на всіх основних сучасних браузерах.
- Розширення має бути адаптовано для коректного відображення на мобільних пристроях і планшетах.

7. Безпека

- Система повинна бути захищена від потенційних вразливостей, пов'язаних із завантаженням та відображенням тривимірних моделей.
- Дані про моделі повинні бути надійно збережені та захищені від несанкціонованого доступу.

8. Інтеграція з іншими функціональними можливостями Magento 2

- Система повинна інтегруватися з існуючими функціональними можливостями Magento 2, такими як управління товарами, каталогом і темами оформлення.
- Модуль не має порушувати сумісний функціонал стандартної теми Magento 2.

9. Вимоги до стандартів Magento 2

- Модуль має задовольняти усі вимоги до коду та відповідати актуальним стандартам Magento 2
- Розширення повинно успішно проходити тестування на відповідність стандартам.

Враховуючи вищенаведені функціональні вимоги було створено дві діаграми варіантів використання (Use Case) для адміністратора та для користувача, які ілюструє взаємодію із модулем для відображення тривимірних моделей товарів у Magento 2.

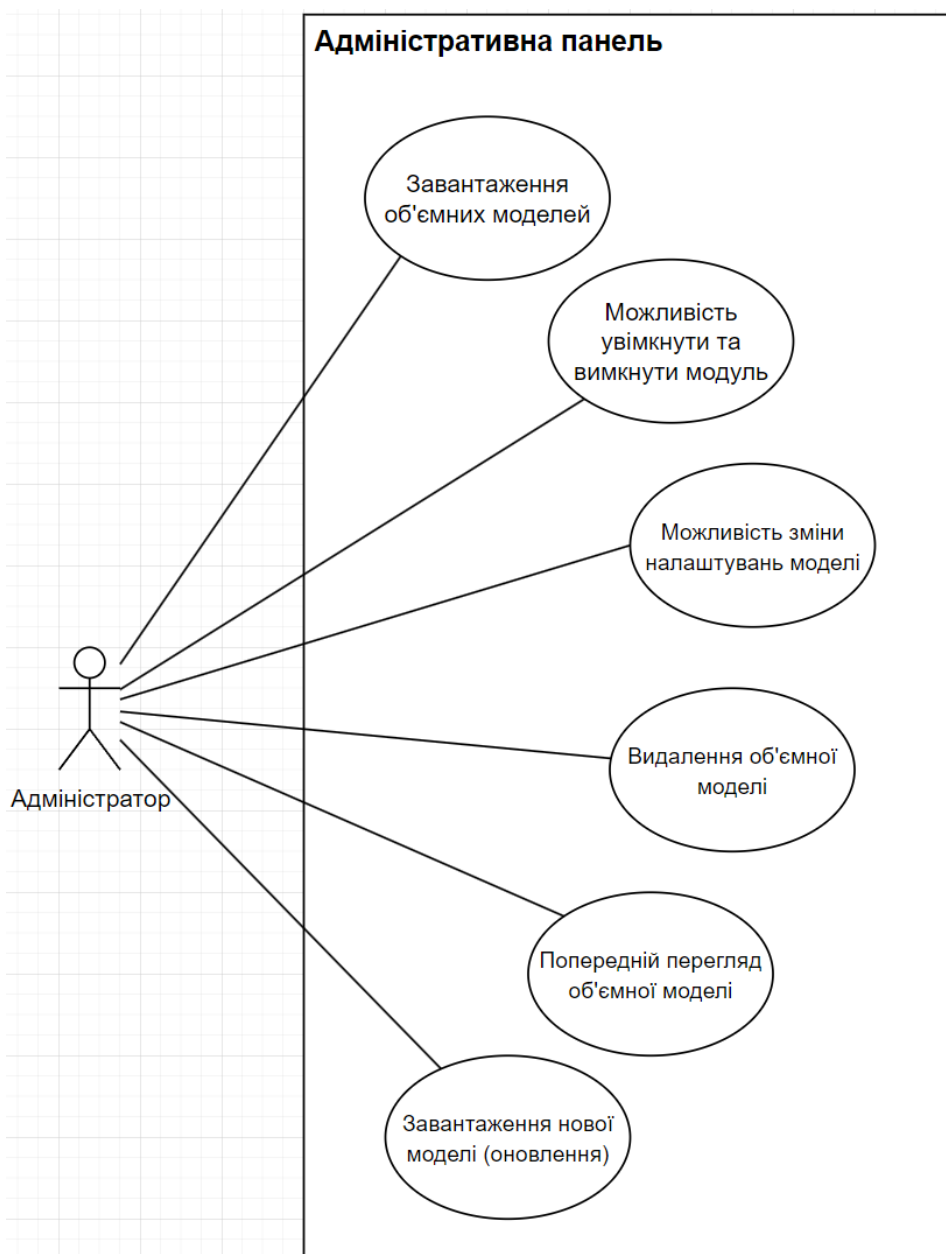


Рисунок 4 — Use Case діаграма взаємодії адміністратора з розширенням

Діаграма взаємодії адміністратора з розширенням, яка представлена у рисунку 4 надає візуальне представлення щодо функціональних можливостей розширення зі сторони адміністративної панелі.

Елементи діаграми взаємодії адміністратора з розширенням має наступні елементи:

1. Актори:

1.1 Адміністратор: основний користувач, який взаємодіє з системою для управління об'ємними моделями.

2. Сценарії використання:

1.1 Завантаження об'ємних моделей: Адміністратор має можливість завантажувати нові тривимірні моделі через адміністративну панель.

1.2 Можливість увімкнути та вимкнути модуль: Адміністратор може активувати або деактивувати модуль відображення об'ємних моделей.

1.3 Можливість зміни налаштувань моделі: Адміністратор може змінювати налаштування оточення та інші параметри.

1.4 Видалення об'ємної моделі: Адміністратор може видаляти тривимірні моделі з системи.

1.5 Попередній перегляд об'ємної моделі: Адміністратор може переглянути модель перед її остаточним збереженням і відображенням на сторінці товару.

1.6 Завантаження нової моделі (оновлення): Адміністратор може оновлювати існуючі моделі, завантажуючи нові версії тривимірних моделей.

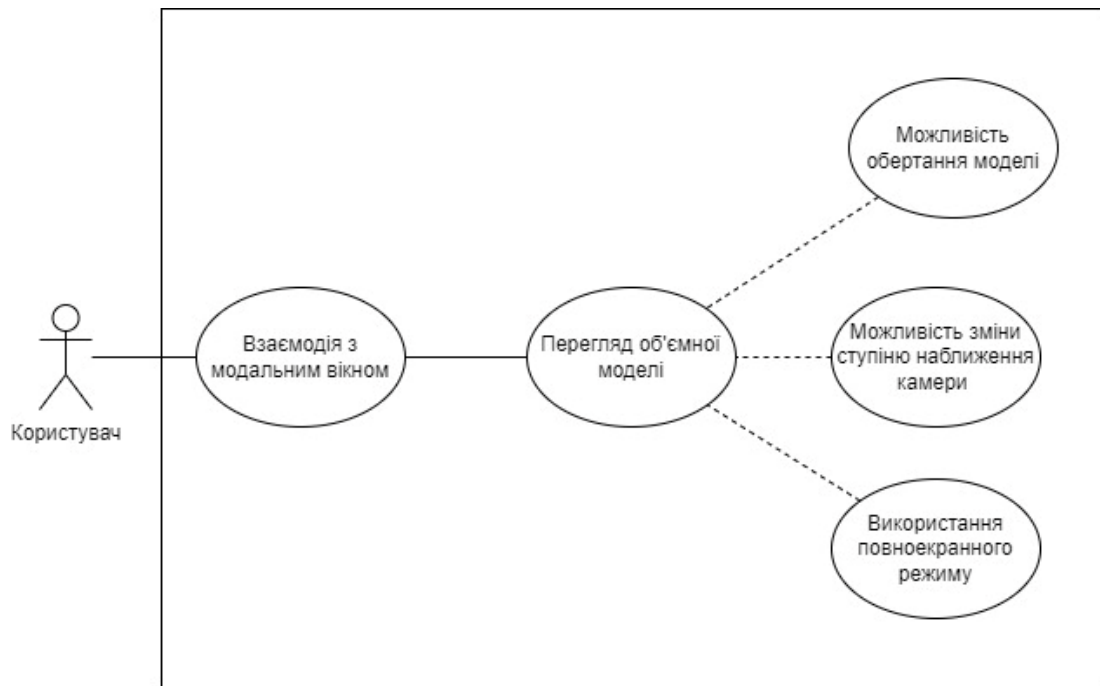


Рисунок 5 — Use Case діаграма взаємодії користувача з розширенням

На рисунку 5 зображено діаграму варіантів використання (use case) зображено взаємодію кінцевого користувача з розширенням для перегляду об'ємних моделей на сайті електронної комерції.

Елементи діаграми взаємодії адміністратора з розширенням має наступні елементи:

1. Актори:

1.1 Користувач: основний користувач, який взаємодіє з модулем тривимірних моделей товарів на сайті.

2. Сценарії використання:

1.1 Взаємодія з модальним вікном: Користувач може взаємодіяти з модальним вікном, у якому відображається тривимірна модель товару. Це включає відкриття та закриття вікна, а також взаємодію з елементами керування всередині нього.

1.2 Можливість увімкнути та вимкнути модуль: Адміністратор може активувати або деактивувати модуль відображення об'ємних моделей.

1.3 Можливість обертання моделі: Користувач може обертати тривимірну модель для перегляду з різних кутів.

1.4 Можливість зміни ступеню наближення камери: Користувач може змінювати масштаб моделі, наближаючи або віддаляючи її для детальнішого огляду.

1.5 Використання повноекранного режиму: Користувач може перемикатися в повноекранний режим для більш зручного і детального перегляду тривимірної моделі.

3.4 Вимоги до апаратного та програмного забезпечення

Для забезпечення належної роботи розширення для відображення тривимірних моделей товарів у Magento 2 необхідно врахувати як апаратні, так і програмні вимоги. Нижче наведено детальні вимоги до серверного та клієнтського обладнання, а також програмного забезпечення, яке забезпечить ефективну і стабільну роботу системи.

Сервер:

- Оперативна пам'ять: Мінімум 2 ГБ оперативної пам'яті, рекомендовано 8 ГБ і більше для високої продуктивності.
- Дисковий простір: SSD з об'ємом не менше 4 ГБ для забезпечення швидкого доступу до даних та достатнього місця для зберігання тривимірних моделей.
- Операційна система: Linux
- СКБД: MySQL 5.6 - 5.7, або MariaDB 10.0 - 10.4.
- PHP: Версії 7.4 або новіша для забезпечення сумісності з Magento
- Magento 2: Версії 2.4.5 або вище.
- Composer: Менеджер залежностей для PHP, необхідний для встановлення та оновлення Magento 2 і його компонентів.

Клієнт:

- Веб-браузер: Google Chrome версії 60 та вище, Mozilla Firefox версії 54 і вище, Microsoft Edge версії 79 і вище, Safari версії 11 і вище, Opera версії 47 і вище, або мобільні браузери з операційними системами iOS 13 або новіше чи Android версії 8.0.0 або вище
- Оперативна пам'ять: Мінімум 4 ГБ оперативної пам'яті для забезпечення плавної роботи веб-браузера.
- Графічний процесор: Вбудований або дискретний графічний процесор з підтримкою WebGL.
- Екран: Роздільна здатність екрану не менше 1280x720 пікселів для зручного перегляду тривимірних моделей.
- Клавіатура та миша для взаємодії з модулем

Вищезазначені вимоги гарантують стабільну та ефективну роботу розширення, забезпечуючи високий рівень продуктивності та зручності використання для адміністраторів та кінцевих користувачів.

3.5 Модулі і алгоритми

Для забезпечення функціональності розширення для відображення тривимірних моделей товарів у Magento 2, було розроблено і інтегровано кілька ключових модулів та алгоритмів. Ці компоненти забезпечать завантаження, зберігання, рендеринг і взаємодію з тривимірними моделями, а також оптимізацію продуктивності і безпеки системи.

Одним з найголовніших алгоритмів модуля є алгоритм пошуку і збереження виконуваного файлу зі сценою. Для реалізації цього алгоритму було використано властивість EAV атрибуту використовувати спеціально створену модель, яка спрацьовує під час збереження конфігурації продукту.

Лістинг 3 Метод для збереження файлу об'ємної моделі

```

/**
 * Method for saving volume module data to the database
and media storage
 *
 * @param $object
 * @return File
 */
public function afterSave($object): File|static
{
    $this->_targetObject = $object;

    $path = $this->_filesystem
        ->getDirectoryRead(DirectoryList::MEDIA)
        ->getAbsolutePath('wg/models/');

    try {
        $uploader = $this->_fileUploaderFactory-
>create([
            'fileId' => 'product[' . $this-
>getAttribute()->getName() . ']'
        ]);

        $uploadedFileExtension = $uploader-
>getFileExtension();

        if ($uploadedFileExtension === "zip" ||
$uploadedFileExtension === "rar") {
            return $this-
>unpackAndSaveModelFromArchive($path,
$uploadedFileExtension);
        }
        return $this->saveModelFile($path);
    } catch (Exception $e) {
        if ($e->getCode() !=
UploaderAlias::TMP_NAME_EMPTY) {
            $this->_logger->critical($e);
            $this->messageManager->addErrorMessage(
                __('An error occurred while saving the
3D model of product: %1', $e->getMessage())
            );
        }
    }

    return $this;
}

```


Лістинг 3 представляє демонстрацію методу `afterSave`, який відповідає за збереження даних про тривимірні моделі до бази даних та медіа-сховища після збереження продукту у адміністративній панелі. Він приймає об'єкт `$object` як параметр, який встановлюється як цільовий об'єкт для подальшої обробки. Метод отримує абсолютний шлях до директорії медіа-сховища, де будуть збережені моделі.

У блоці `try` створюється об'єкт `FileUploader` за допомогою фабрики `FileUploaderFactory`, який налаштовується для завантаження файлу з форми продукту. Метод отримує розширення завантаженого файлу. Якщо розширення файлу є `"zip"` або `"rar"`, викликається метод `unpackAndSaveModelFromArchive`, який представлено у Лістингу 4 – він розпаковує архів і зберігає модель у визначеній директорії. Якщо розширення файлу відрізняється, викликається метод `saveModelFile`, який зберігає файл моделі у вказаному шляху.

У разі виникнення виключної ситуації метод обробляє помилку - записується критична помилка в лог і додається повідомлення про помилку в менеджер повідомлень у адміністративній панелі.

Лістинг 4 *Алгоритм для розархівування та збереження виконаного файлу зі сценою.*

```
/**
 * Unpack archive and save model
 *
 * @param $path
 * @param $uploadedFileExtension
 * @return File
 * @throws FileSystemException
 */
private function unpackAndSaveModelFromArchive($path,
$uploadedFileExtension): static
{
    $uploader = $this->_fileUploaderFactory-
>create(['fileId' => 'product[' . $this->getAttribute()-
>getName() . ']);
    $uploader->setAllowRenameFiles(true);
    $uploader->save($path);
}
```

```

        $fileName = $uploader->getUploadedFileName();

        $modelDirectoryPath = $path . $this->_targetObject-
>getData('entity_id');
        $archiveDirectoryPath = $path . $fileName;
        $this-
>createOrUpdateModelFolder($modelDirectoryPath);

        if ($uploadedFileExtension === "zip") {
            $this->extractZipArchive($archiveDirectoryPath,
$modelDirectoryPath);
        } elseif ($uploadedFileExtension === "rar") {
            $this->extractRarArchive($archiveDirectoryPath,
$modelDirectoryPath);
        }

        $executiveFilePath = $this-
>getFileExecutionFolder($modelDirectoryPath);
        $executionFileData = $this-
>searchExecutiveFile($executiveFilePath);
        $this->removeModelArchive($archiveDirectoryPath);
        $this->saveModelDataToEav($executiveFilePath,
$executionFileData);

        return $this;
    }

```

Лістинг 4 описує алгоритм, який розпаковує архівний файл і зберігає тривимірну модель у визначеній директорії. Метод використовується для обробки завантажених архівів з моделями (у форматах ZIP або RAR), їх розпаковки та збереження даних про модель.

Роботу алгоритму можна описати наступним чином:

Метод приймає шлях до директорії та розширення завантаженого файлу, щоб розпакувати архів і зберегти модель. Спочатку алгоритм створює об'єкт FileUploader за допомогою фабрики FileUploaderFactory для завантаження файлу з форми, дозволяючи перейменування файлів під час збереження, щоб уникнути конфліктів імен.

Завантажений файл зберігається у вказаній директорії, а його ім'я отримується для подальшої обробки. Далі створюється або оновлюється директорія для збереження розпакованих файлів моделі, використовуючи

entity_id продукту. Якщо завантажений файл має розширення zip, він розпаковується за допомогою методу extractZipArchive, а якщо rar, то використовується метод extractRarArchive.

Після розпакування визначається шлях до директорії, де розташовані виконувані файли, і виконується пошук цих файлів у вказаній директорії.

Після успішного розпакування оригінальний архів видаляється, а дані про модель зберігаються в EAV-атрибутах Magento, використовуючи шлях до файлу і дані виконуваного файлу. Наприкінці метод повертає поточний об'єкт для можливості виклику методів ланцюжком.

Таким чином виконується розархівування та збереження виконуваного файлу об'ємної моделі.

Іншою важливою частиною коду є отримання даних про модель на стороні клієнта. Для виконання цієї задачі було розроблено спеціальний API запит, який дозволяє автоматизувати процес збору та підготовки необхідної інформації для рендерингу тривимірних моделей, що робить систему більш гнучкою та легкою в налаштуванні. Метод, який обробляє API запит наведено у Лістингу 5.

Лістинг 5 Метод для обробки конфігураційних даних

```
/**
 * @param $productId
 * @return bool|string
 * @throws NoSuchEntityException
 */
public function
getProductVolumeModelConfig($productId): bool|string
{
    $configuration = $this->helper-
>getGlobalSettings();
    $product = $this->productRepository-
>getById($productId);

    $resultArray = array_filter($product->getData(),
function ($value, $key) {
        return str_starts_with($key, "volume_");
    }, ARRAY_FILTER_USE_BOTH);
```

```

        $resultArray = array_merge($resultArray,
$configuration);
        $resultArray = $this-
>buildEnvironmentMapUrl($resultArray);

        return $this->serializer->serialize($resultArray);
    }

```

Метод `getProductVolumeModelConfig`, який представлено у лістингу 5, призначений для отримання конфігураційних даних тривимірної моделі для заданого продукту. Він приймає ідентифікатор продукту як параметр і повертає серіалізовані конфігураційні дані у вигляді строки JSON або булеве значення `false` у разі помилки. Спочатку метод отримує глобальні налаштування модуля за допомогою методу `getGlobalSettings` з об'єкта `helper`. Потім він використовує `productRepository`, щоб отримати об'єкт продукту за заданим ідентифікатором `$productId`. Якщо продукт з таким ідентифікатором не знайдено, буде викинуто виняток `NoSuchEntityException`.

Метод отримує всі дані продукту і фільтрує їх, залишаючи тільки ті, ключі яких починаються з префіксу `"volume_"`. Це робиться за допомогою функції `array_filter`, яка переглядає всі ключі та значення масиву даних продукту. Після фільтрації, метод об'єднує отримані дані з глобальними налаштуваннями, використовуючи функцію `array_merge`. Це дозволяє поєднати специфічні налаштування продукту з загальними конфігураційними даними. Далі метод викликає `buildEnvironmentMapUrl`, щоб обробити конфігураційні дані і додати повний URL до карти оточення, якщо таке налаштування існує. Цей крок забезпечує, що всі необхідні ресурси будуть правильно налаштовані для відображення тривимірної моделі.

Нарешті, метод серіалізує об'єднані конфігураційні дані у формат JSON за допомогою `serializer` і повертає результат. Це дозволяє зручно передавати дані у фронтенд або інші частини системи, які потребують цих налаштувань. Метод виконує комплексну обробку конфігураційних даних для тривимірних моделей продуктів, збираючи дані з різних джерел, фільтруючи і об'єднуючи

їх, додаючи необхідні параметри і серіалізуючи результат для подальшого використання.

Окремою важливою частиною роботи розширення є видалення моделей з медіа сховища при видаленні продуктів з системи. Це робиться задля того щоб у медіа сховищі не залишались збереженими моделі від продуктів, яких вже не існує у системі. Такий підхід допомагає підтримувати чистоту і організованість файлової системи, запобігаючи накопиченню непотрібних даних, що можуть негативно вплинути на продуктивність та використання ресурсів сервера.

Задля реалізації цієї задачі були використані Magento плагіни – спеціальні класи, які дозволяють змінювати або доповнювати поведінку функцій публічних класів, перехоплюючи виклик функції до, після або під час виклику функції. У лістингу 6 представлено оголошення плагіну, який буде перехоплювати виклик функції видалення продукту та виконувати видалення директорії об'ємної моделі з медіа сховища під час видалення продукту.

Лістинг 6 Оголошення плагіну для видалення директорії моделі

```
<?xml version="1.0"?>

<config xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
xsi:noNamespaceSchemaLocation="urn:magento:framework:Object
Manager/etc/config.xsd">
    <type
name="Magento\Catalog\Api\ProductRepositoryInterface">
        <plugin name="wg_volumetricmodels_delete_directory"
type="Whidegroup\VolumetricModels\Plugin\ProductRepositoryP
lugin" sortOrder="1"/>
    </type>
</config>
```

Таким чином, під час видалення продукту буде викликано метод `aroundDelete`, який представлено у класі `ProductRepositoryPlugin`, реалізацію якого можна побачити у лістингу 7.

Лістинг 7 Метод для видалення директорії моделі

```

/**
 * Plugin method to delete volume model folder on
product delete.
 *
 * @param ProductRepositoryInterface $subject
 * @param Closure $proceed
 * @param ProductInterface $product
 * @return mixed
 * @throws FileSystemException
 * @throws NoSuchEntityException
 */
public function aroundDelete(
    ProductRepositoryInterface $subject,
    Closure $proceed,
    ProductInterface $product
): mixed
{
    $path = $this->_filesystem
        ->getDirectoryRead(DirectoryList::MEDIA)
        ->getAbsolutePath('wg/models/');

    $productId = $product->getEntityId();
    $product = $this->productRepository-
>getById($productId);
    $productId = $product->getId();
    $volumeModelPath = $path . $productId;

    $result = $proceed($product, false);

    if ($this->_file->isExists($volumeModelPath) &&
isset($productId)) {
        $this->_file-
>deleteDirectory($volumeModelPath);
    }

    return $result;
}

```

У методі `aroundDelete` реалізовано логіку, яка забезпечує видалення директорій з тривимірними моделями при видаленні відповідних продуктів. Спочатку метод отримує шлях до директорії `wg/models/` у медіа-каталозі. Потім, за допомогою репозиторію продуктів, метод отримує об'єкт продукту за його ідентифікатором, щоб переконатися, що продукт дійсно існує.

Після цього метод визначає шлях до директорії, в якій зберігаються тривимірні моделі продукту. Викликається оригінальний метод видалення продукту через замикання `$proceed`, що забезпечує видалення продукту з бази даних.

Після успішного видалення продукту з бази даних, метод перевіряє наявність директорії з тривимірними моделями. Якщо така директорія існує, вона видаляється за допомогою методу `deleteDirectory` об'єкта `File`. Таким чином, метод гарантує, що всі пов'язані з продуктом файли моделей також будуть видалені з системи.

Цей підхід забезпечує ефективне управління файлами в системі, запобігаючи накопиченню зайвих даних та забезпечуючи відповідність між наявними продуктами і файлами моделей у медіа сховищі. Це також спрощує адміністрування та обслуговування системи, оскільки видалення непотрібних файлів виконується автоматично і не потребує додаткових зусиль з боку адміністратора.

Іншою важливою частиною розширення є скрипт, який додається на сторінку продукту та ініціалізує функціонал для перегляду об'ємних моделей. Реалізацію цього коду представлено у лістингу 8.

Лістинг 8 *Скрипт ініціалізації 3D сцени*

```
async function init3DScene(config) {
  displayLoader();
  const modelUrl = get(config, "volume_model", null);
  const modelBackground = get(config,
    "volume_model_background", "#CCCCCC");
  const autoRotate = get(config, "rotation_enable",
    false);
  const moduleEnvironmentPath = get(config,
    "volume_model_environment", null);
  const fov = get(config, "fov_value", 60);
  camera.fov = Number(fov);

  const executionFileData = get(config,
    "volume_model_executive_file", "");
```

```

    const fileExtension =
fileHandler(executionFileData).extension;
    const fileName = fileHandler(executionFileData).name;

    if (moduleEnvironmentPath) {
        loadEnvironment(moduleEnvironmentPath, scene);
    } else {
        scene.background = new
THREE.Color(modelBackground);
        environmentLoaded = true;
        checkLoadingCompletion();
    }

    const loaderByExtension =
extensionLoaderMapping[fileExtension];

    if (!loaderByExtension || !modelUrl) {
        console.error("Произошла ошибка при рендеринге 3D
модели.");
        return null;
    }

    const loader = new loaderByExtension(loaderManager);

    loader.load(`/media/${modelUrl}/${fileName}`, function
(loaderModel) {
        let model = loaderModel;
        if (fileExtension === "stl") {
            model = new THREE.Mesh(loaderModel, new
MeshPhongMaterial({ color: "#e6e6e6" }));
        }

        if (Object.hasOwn(loaderModel, 'scene') ||
fileExtension === "gltf") {
            model = loaderModel.scene;
        }
        setRendererSize(camera, renderer);
        controls.autoRotate = autoRotate;

        model = scaleModel(model);
        cameraPositionSetup(model);

        scene.add(model);
    });

    addLight(scene, fileExtension);
}

```


Скрипт, який ініціює сцену є асинхронною функцією `init3DScene`, яка ініціалізує та налаштовує тривимірну сцену для рендерингу 3D моделей.

Функція приймає конфігураційний об'єкт `config` і виконує низку дій. Спочатку відображається завантажувач за допомогою функції `displayLoader()`, потім з об'єкта `config` витягуються необхідні параметри, такі як URL моделі (`modelUrl`), колір фону сцени (`modelBackground`), прапорець автоматичного обертання моделі (`autoRotate`), шлях до файлу середовища (`moduleEnvironmentPath`), значення кута огляду камери (`fov`) і дані файлу для виконання моделі (`executionFileData`). Кут огляду камери встановлюється на основі значення `fov`, яке було задано у адміністративній панелі в розділі налаштувань розширення.

Якщо в конфігурації задано шлях до файлу середовища, викликається функція `loadEnvironment` для завантаження цього середовища в сцену. Інакше встановлюється колір фону сцени на основі значення `modelBackground`. Визначається тип файлу моделі за допомогою функції `fileHandler`, яка повертає розширення файлу та його назву. Потім з мапінгу `extensionLoaderMapping` отримується відповідний завантажувач для цього типу файлів.

За допомогою завантажувача завантажується модель з заданого URL. Після завантаження моделі виконуються наступні дії: якщо модель має властивість `scene` або розширення файлу `glTF`, використовується властивість `scene`. Налаштовується розмір рендерера за допомогою функції `setRendererSize`, прапорець автоматичного обертання моделі встановлюється на основі значення `autoRotate`, масштабується модель за допомогою функції `scaleModel`, налаштовується позиція камери за допомогою функції `cameraPositionSetup`.

Завантажена і налаштована модель додається до сцени. До сцени також додається освітлення за допомогою функції `addLight`, яка використовує інформацію про тип файлу моделі.

Таким чином, цей код забезпечує повну ініціалізацію тривимірної сцени, налаштування камери, завантаження та рендеринг 3D моделі, а також обробку

середовища і додавання освітлення. Він використовує різноманітні методи для обробки конфігураційних даних, налаштування рендерера та взаємодії з бібліотекою Three.js.

3.6 Проєкт інтерфейсу

Проєкт інтерфейсу для розширення відображення тривимірних моделей товарів на сайті електронної комерції має на меті забезпечити користувачів зручним і функціональним інструментом для взаємодії з 3D моделями товарів. Інтерфейс повинен бути інтуїтивно зрозумілим і надавати всі необхідні функції для перегляду та налаштування тривимірних моделей.

Для реалізації цієї задачі було створено стандартне модальне вікно Magento 2 на сторінці продукту. Воно відкривається по натисканню на відповідну кнопку 3D Model, яка знаходиться під блоком інформації про продукт, або на кнопку 3D View під зображенням товару. В залежності від обраного режиму відображення кнопок, на сторінці буде доступна або кнопка під блоком інформації про продукт, або кнопка під зображенням товару, або обидві кнопки, як показано на рисунку 6.

Після натискання на кнопку відбувається відкриття модального вікна, яке можна побачити на рисунку 7 та ініціалізація 3D сцени. Під час завантаження всіх налаштувань розширення та об'ємної моделі користувач буде бачити екран завантаження, на якому присутній графічно зображений процес завантаження.

Верхня частина модального вікна має заголовок 3D Model, що вказує на призначення цього вікна. В правому верхньому куті вікна є кнопка для переходу у повноекранний режим, яка дозволяє користувачу розгорнути вікно перегляду на весь екран для кращого огляду моделі. Інтерфейс має простий та мінімалістичний дизайн, що робить його зручним для користування та не перевантажує зайвими елементами.

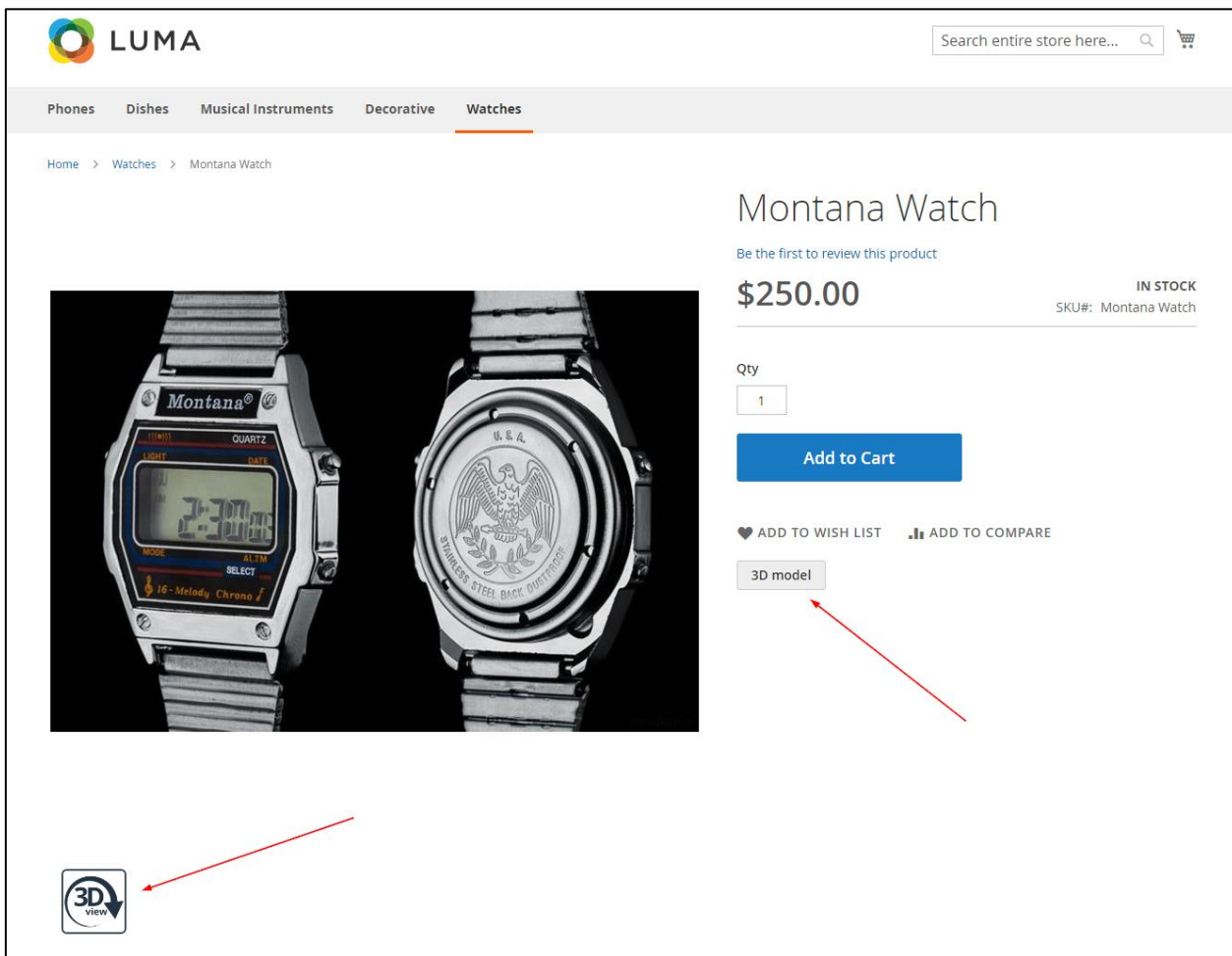


Рисунок 6 — Кнопки для відкриття модального вікна

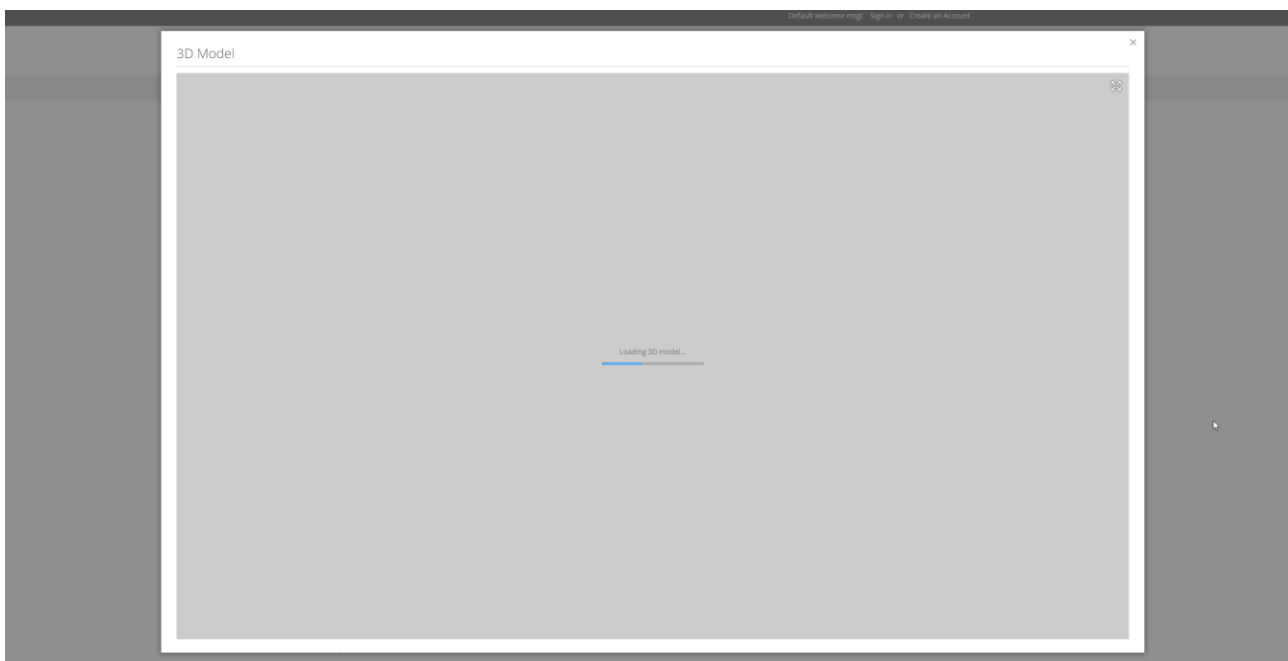


Рисунок 7 — Модальне вікно з процесом завантаження моделі

Після закінчення завантаження моделі, користувачу буде доступна взаємодія з об'ємною моделлю шляхом натискання лівої клавіші миші та її пересуванням. На рисунку 8 зображено завантажену модель, з якою можна взаємодіяти. Після взаємодії зі сценою, модель буде відповідним чином реагувати – обертатись, збільшуватись або зменшуватись, тощо (див. Рисунок 9).



Рисунок 8 — Модальне вікно з завантаженою моделлю



Рисунок 9 — Демонстрація обертання об'ємної моделі

Окрім цього, користувач має змогу перейти у повноекранний режим шляхом натискання відповідної кнопки, яка розташована у верхньому-правому куті вікна, як показано на рисунку 10. Після натискання кнопки, вікно рендеру буде займати увесь можливий простір сторінки браузера (див. рисунок 11), а іконка повноекранного режиму зміниться на протилежну.



Рисунок 10 — Модальне вікно до переходу у повноекранний режим



Рисунок 11 — Повноекранний режим відображення

У адміністративній панелі розширення додає нову секцію до сторінки редагування продукту, яка містить кілька налаштувань для керування тривимірними моделями товару та демонструється у рисунку 12.

У верхній частині секції розташований перемикач "3D Model Enabled", який дозволяє увімкнути або вимкнути функцію відображення тривимірної моделі для цього товару.

Далі йдуть кнопки дій:

- Preview Model: Дозволяє попередньо переглянути завантажену тривимірну модель.
- Load New Model: Дозволяє завантажити нову тривимірну модель для цього товару.
- Delete Model: Дозволяє видалити поточну тривимірну модель.

Під кнопками дій розташовані додаткові налаштування:

- Environment of the scene for 3D Model: Вибір середовища для сцени для тривимірної моделі.
- Background of 3D viewer: Поле для вибору кольору фону тривимірного переглядача.

Ці налаштування дозволяють адміністраторам легко керувати тривимірними моделями товарів, налаштовувати параметри відображення та забезпечувати користувачам можливість взаємодіяти з тривимірними зображеннями товарів на сайті.

Окрім цього, додаткові глобальні налаштування розташовані на сторінці конфігурації інтернет-магазину (див. рисунок 13). Вони розташовані у спеціальній секції модуля, та мають наступні налаштування:

- Enable Extension — відповідає за стан модуля.
- Enable Model Rotation — включити режим обертання моделей.
- Button Displaying Mode — режим відображення кнопок з модальним вікном
- FOV of the camera — відповідає за параметр FOV камери.

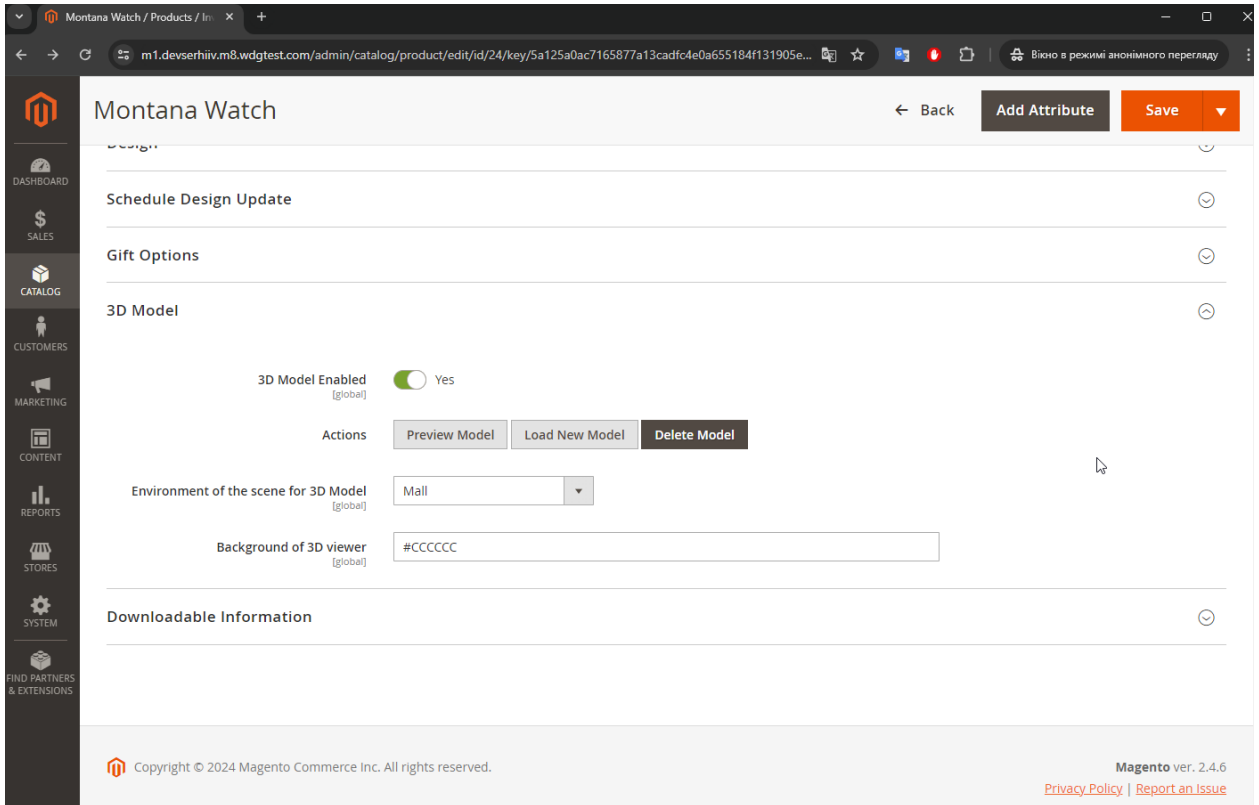


Рисунок 12 — Секція налаштувань моделі на сторінці редагування товару

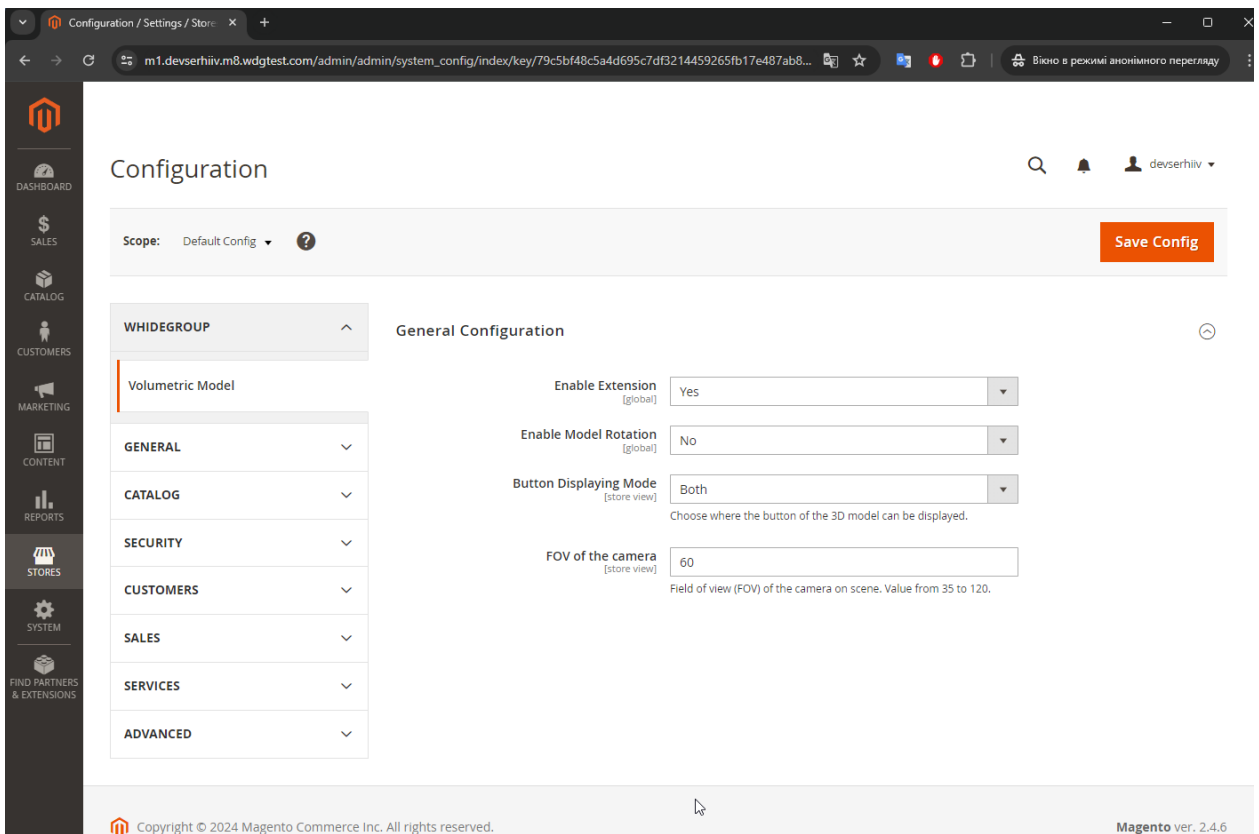


Рисунок 13 — Секція загальних налаштувань модуля

ВИСНОВКИ

1. Виконано аналіз предметної області та літературних джерел на тему розробки розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції.
2. Проведена аналітична робота з метою визначення оптимальних технологій для реалізації відображення тривимірних моделей. В рамках проєкту був проведений детальний аналіз і вибір засобу збирання застосунку. Це дозволило скоротити час розробки та забезпечити ефективну роботу над проєктом, надаючи інструменти для швидкої і зручної інтеграції змін.
3. Здійснено розробку та адаптацію розширення для сайту електронної комерції на основі системи керування вмісту Magento 2. Під час розробки було створено архітектуру системи та виконані всі вимоги, які були поставлені до застосунку. Розроблено серверну та клієнтську частини застосунку.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Xuemei Zhang, Gengui Zhou, Jian Cao, Anqi Wu. Evolving strategies of e-commerce and express delivery enterprises with public supervision // *Research in Transportation Economics*. 2020. Vol. 80. Article 100810. ISSN 0739-8859. URL: <https://doi.org/10.1016/j.retrec.2019.100810>. (дата звернення 10.04.2024).
2. Moiseev, N., Mikhaylov, A., Dinçer, H., Yüksel, S. Market capitalization shock effects on open innovation models in e-commerce. // *Financial Innovation*. 2023. Vol. 9, No. 55. URL: <https://doi.org/10.1186/s40854-023-00461-x>. (дата звернення 10.04.2024).
3. Gupta S., Kushwaha P.S., Badhera U., Chatterjee P., Santibanez Gonzalez E.D.R. Identification of benefits, challenges, and pathways in E-commerce industries. // *Sustainable Operations and Computers*. 2023. Vol. 4. Pp. 200-218. ISSN 2666-4127. URL: <https://doi.org/10.1016/j.susoc.2023.08.005>. (дата звернення 10.04.2024).
4. Kublitska, O. Electronic commerce market in ukraine: current state and trends of post-war recovery. 2023. Pp. 98-108. URL: [https://doi.org/10.25140/2411-5215-2023-3\(35\)-98-108](https://doi.org/10.25140/2411-5215-2023-3(35)-98-108). (дата звернення 10.04.2024).
5. Rathee, V., Kumari, S. 3D Technology in E-tail: A Synthesis through Literature Review // *Orissa Journal of Commerce*. 2023. URL: <https://doi.org/10.54063/ojc.2023.v44i04.07>. (дата звернення 10.04.2024).
6. Hewawalpita, S., Perera, I. Effect of 3D Product Presentation on Consumer Preference in E-Commerce // 2017. URL: <https://doi.org/10.1109/MERCon.2017.7980532>. (дата звернення 10.04.2024).
7. 3D Product Viewer For Magento 2. Milople site. URL: <https://www.milople.com/magento-2-3d-product-view.html> (дата звернення 15.04.2024).

8. Webkul. Magento 2 3D Product Configurator Customizer. Webkul site. URL: <https://store.webkul.com/magento2-3d-product-configurator-customizer.html> (дата звернення 15.04.2024).
9. WebRotate 360. Plugin for Magento. WebRotate 360 site. URL: <https://www.webrotate360.com/products/cms-and-e-commerce-plugins/plugin-for-magento.aspx> (дата звернення 15.04.2024).
10. Three.js. Three.js Documentation. Three.js site. URL: <https://threejs.org/docs/> (дата звернення 21.04.2024).
11. Babylon.js. Babylon.js Documentation. Babylon.js site. URL: <https://doc.babylonjs.com/> (дата звернення 21.04.2024).
12. Vite. Why Vite. Vite site. URL: <https://vitejs.dev/guide/why.html> (дата звернення 21.04.2024).
13. Webpack. Webpack Concepts. Webpack site. URL: <https://webpack.js.org/concepts/> (дата звернення 21.04.2024).
14. Rollup. Introduction to Rollup. Rollup site. URL: <https://rollupjs.org/introduction/> (дата звернення 21.04.2024).
15. Adobe. Adobe Commerce Documentation. Adobe site. URL: <https://developer.adobe.com/commerce/docs/> (дата звернення 15.05.2024).
16. Shopify. Shopify Documentation. Shopify site. URL: <https://shopify.dev/docs> (дата звернення 15.05.2024).
17. BigCommerce. BigCommerce Developer Documentation. BigCommerce site. URL: <https://developer.bigcommerce.com/> (дата звернення 15.05.2024).
18. Ветров С.О., Заяц В.І. доцент, канд. технічних наук – науковий керівник. Розробка розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції. Збірник наукових праць студентів, аспірантів, докторантів і молодих вчених «Молода наука-2024» / Запорізький національний університет. – Запоріжжя : ЗНУ, 2024. – Т.5

**Декларація
академічної доброчесності
здобувача ступеня вищої освіти ЗНУ**

Я, Ветров Сергій Олександрович, студент 4 курсу, форми навчання денної, Інженерного навчально-наукового інституту, спеціальність 121 Інженерія програмного забезпечення, адреса електронної пошти ipz20bd-203@stu.zsea.edu.ua, — підтверджую, що написана мною кваліфікаційна робота на тему **«Розробка розширення для реалізації відображення тривимірних моделей товарів на сайті електронної комерції»** відповідає вимогам академічної доброчесності та не містить порушень, що визначені у ст.42 Закону України «Про освіту», зі змістом яких ознайомлений.

- заявляю, що надана мною для перевірки електронна версія роботи є ідентичною її друкованій версії;

згоден/згодна на перевірку моєї роботи на відповідність критеріям академічної доброчесності у будь-який спосіб, у тому числі за допомогою інтернет-системи, а також на архівування моєї роботи в базі даних цієї системи.

Дата 17.06.2024 Підпис _____ Ветров Сергій Олександрович
(студент)

Дата 18.06.2024 Підпис _____ Заяц Валерій Іванович
(науковий керівник)