

Хмельницький національний університет
Факультет інформаційних технологій
Кафедра інженерії програмного забезпечення

КВАЛІФІКАЦІЙНА РОБОТА

Онлайн-ресурс для роздрібної торгівлі одягом

Назва теми

Рівень вищої освіти Перший (бакалаврський)
Галузь знань 12 «Інформаційні технології»
Спеціальність 121 «Інженерія програмного забезпечення»
Освітня програма Освітньо-професійна програма «Інженерія програмного
забезпечення»

Шифр КвРІПЗ. 200139.01.14.ПЗ

Виконав студент IV курсу група ІПЗ-19-1

Т. А. Монастирський

Підпис

Ініціали, прізвище

Керівник канд. техн. наук, доцент

Науковий ступінь, звання

О. М. Яшина

Підпис

Ініціали, прізвище

Нормоконтролер канд. техн. наук, доцент

Ю. В. Форкун

Підпис

Ініціали, прізвище

До захисту допускаю:

Завідувач кафедри інженерії
програмного забезпечення

Л. П. Бедратюк

Підпис

Ініціали, прізвище

5 червня 2023 р.

Хмельницький 2023

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій
Кафедра Інженерії програмного забезпечення
Рівень вищої освіти Перший (бакалаврський)
Галузь знань 12 «Інформаційні технології»
Спеціальність 121 «Інженерія програмного забезпечення»
Освітня програма Освітньо-професійна програма «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри ІПЗ

Л. П. Бедратюк

01 03 2023 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Монастирському Тарасу Андрійовичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Онлайн-ресурс для роздрібної торгівлі одягом

Керівник проекту (роботи) Яшина Оксана Миколаївна, канд. техн. наук, доцент

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 01.03.2023 р. № 5

2. Строк подання студентом проекту (роботи) на кафедру 01.06.2023 р.

3. Вихідні дані до проекту (роботи) Матеріали переддипломної практики

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

Дослідження предметної області та постановка задачі, проектування програмного забезпечення, програмна реалізація, тестування програмного забезпечення

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


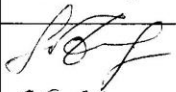
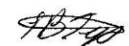

Три креслення у форматі А3:

1.Схема бази даних

2.Діаграма поведінки

3.Діаграма варіантів використання

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Форкун Ю. В., доцент кафедри ПЗ	 04.06.2023	 05.06.2023
Антиплагіат	Гурман І. В., доцент кафедри ПЗ	04.06.2023 	04.06.2023 

7. Дата видачі завдання « 01 » березня 2023 р. _____

КАЛЕНДАРНИЙ ПЛАН

Назва етапів (розділів) Кваліфікаційної роботи	Строк виконання етапів проекту (роботи)	Примітка
1 Ознайомлення з тематикою кваліфікаційної роботи бакалавра (КвР), визначення та узгодження індивідуальних тем КвР	01.12 – 30.12.2022	
2 Дослідження предметної області, в якій планується використання програмного засобу (ПЗ), визначення задач та вимог, розробка технічного завдання	02.01 – 31.01.2023	
3 Проектування програмного забезпечення	01.02 – 28.02.2023	
4 Програмна реалізація	01.03 – 10.04.2023	
5 Тестування програмного забезпечення	11.04 – 30.04.2023	
6 Написання вступу, загальних висновків, оформлення джерел посилання та додатків. Оформлення пояснювальної записки КвР згідно вимог стандартів	01.05 – 25.05.2023	
7 Попередній захист КвР	Травень 2023 (згідно графіка)	
8 Перевірка КвР на плагіат, нормоконтроль, отримання відгуків та рецензій. Брошування (зшиття) пояснювальної записки	26.05 – 1.06.2023	
9 Підготовка до захисту та захист КвР	з 01.06.2023	

Студент



Т. А. Монастирський

Підпис

Ініціали, прізвище

Керівник проекту (роботи)



О.М. Яшина

Підпис

Ініціали, прізвище

АНОТАЦІЯ

Тема кваліфікаційної роботи: Онлайн-ресурс для роздрібної торгівлі одягом.

Автор кваліфікаційної роботи: Монастирський Тарас Андрійович.

Керівник кваліфікаційної роботи: к.т.н., доцент Яшина Оксана Миколаївна.

Пояснювальна записка: 85 с., 29 рис., 1 табл., 2 дод., 40 джерел літератури.

Графічна частина: три креслення у форматі А3.

ВЕБ-ЗАСТОСУНОК, ОНЛАЙН-РЕСУРС, ІНТЕРНЕТ-МАГАЗИН,
ЕЛЕКТРОННА КОМЕРЦІЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, ЕЛЕКТРОННИЙ
РЕСУРС, ПРОГРАМНА СИСТЕМА, ЕЛЕКТРОННИЙ МАГАЗИН.

Мета роботи визначення вимог, проектування та розробка ресурсу, де при наявності мережі інтернет можна здійснювати купівлю одягу в роздріб. Дана мета свою чергу передбачає забезпечення зручної роботи користувачів зі всіма складовими сайту, такими як корзина, оформлення замовлення, а також можливість здійснення повернення товару.

У кваліфікаційній роботі було проаналізовано предметну область, здійснено визначення функціональні та нефункціональні вимог до розроблюваного програмного продукту, розроблена помодульна структура додатку, спроектовано архітектуру та користувацький інтерфейс, розроблено базу даних та програмні модулі. Для розробки онлайн-ресурсу використані такі сучасні інформаційні технології: мова C#, патерн MVC, фреймворк ASP.Net Core.

В результаті було розроблено онлайн-ресурс, який допомагає користувачам обирати, купляти, оплачувати та замовляти доставку одягу.

02.06.23р.

Дата

MartF.

Підпис

ВІДОМІСТЬ ДОКУМЕНТІВ

№ рядка	Формат	Позначення документа	Найменування документа	К-сть аркушів	№ екз.	Примітка
			<u>Текстові документи</u>			
1	A4	КвРІПЗ.190147.19.02.ПЗ	Пояснювальна записка	85		
2	A4		Завдання на КвР	1		
3	A4		Анотація	1		
			<u>Графічні документи</u>			
4	A3	КвРІПЗ.200139.01.14.E8	Схема бази даних	1		
5	A3	КвРІПЗ.200139.01.14.E8	Діаграма поведінки	1		
6	A3	КвРІПЗ.200139.01.14.E8	Діаграма варіантів використання	1		

КвРІПЗ. 200139.01.14.ВД				
Змн.	Арк.	№ докум.	Підпис	Дата
Виконав		Монастирський		02.06
Керівник		Яшина О.М.		05.06
Н. Контр.		Форкун Ю.В.		05.06
Зав. Каф.		Бедратюк Л.П.		05.06
Онлайн-ресурс для роздрібноі торгівлі одягом				
		Літ.	Арк.	Аркушів
			1	1
Відомість документів				
ХНУ, ІПЗ-19-1				

ЗМІСТ

Вступ	6
1 Дослідження предметної області та постановка задачі	9
1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей	9
1.2 Аналіз наявного програмно-технічного забезпечення предметної області	14
1.3 Визначення вимог до онлайн-ресурсу	17
1.4 Технічне завдання для розроблюваного онлайн-ресурсу	22
1.5 Висновки до 1-го розділу. Постановка задачі.	25
2 Проектування онлайн-ресурсу	27
2.1 Архітектура та функціональна структура онлайн-ресурсу	27
2.2 Визначення основних модулів онлайн-ресурсу	29
2.3 Проектування логічної моделі бази даних	31
2.4 Проектування інтерфейсу користувача	33
2.5 Аналіз та вибір технологій і методів реалізації онлайн-ресурсу	36
2.6 Висновки до 2-го розділу	41
3 Програмна реалізація та тестування	42
3.1 Реалізація модулів онлайн-ресурсу	42
3.2 Реалізація бази даних онлайн-ресурсу	43
3.3 Реалізація онлайн-ресурсу	44
3.4 Інструкція з експлуатації.....	48
3.5 Тестування онлайн-ресурсу	55
3.6 Висновки до 3-го розділу	58
Висновки	59
Перелік джерел посилання	61
Додаток А Код (лістинг).....	65
Додаток Б Презентаційні матеріали.....	71
Графічна частина.....	80

					КвРІПЗ. 200139.01.14.ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Онлайн-ресурс для роздрібної торгівлі одягом Пояснювальна записка	Літ.	Арк.	Аркушів
Виконав		Монастирський	<i>[підпис]</i>	02.06			4	85
Керівник		Яшина О.М.	<i>[підпис]</i>	05.06				
Рецензент			<i>[підпис]</i>					
Н. контр.		Форкун Ю.В.	<i>[підпис]</i>	05.06				
Зав. каф.		Бедратюк Л.П.	<i>[підпис]</i>	05.06				
						ХНУ, ІПЗ-19-1		

ВСТУП

Сьогодні дуже популярним є замовлення різноманітних товарів через Інтернет. Особливо це стосується одягу, оскільки жоден магазин одягу не має такого широкого асортименту, який можна було б виставити в онлайн-магазині. Це пов'язано з тим, що такі послуги не вимагають додаткової площі, оренди майданчика для оформлення магазину, тобто допомагають власникам економити кошти.

З боку користувача також є різні переваги і найбільшою з них є економія часу. Саме тому для великої кількості людей таке програмне забезпечення є фундаментом створення власного бізнесу. З часом це стає все більш актуальним та популярним і досить часто бізнес-сегмент замовляє онлайн-сервіси та ресурси саме такого спрямування.

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

Інтернет-магазин об'єднує елементи прямого маркетингу з відвідуванням традиційного магазину. Відмінною рисою інтернет-магазинів у порівнянні зі звичайною формою торгівлі є те, що інтерактивний магазин може запропонувати значно більшу кількість товарів та послуг, та забезпечити споживачів значно більшим обсягом інформації, необхідним для ухвалення рішення про покупку. Крім того, за рахунок використання комп'ютерних технологій можлива персоналізація підходу до кожного з клієнтів, виходячи з історії його відвідувань магазину та зроблених раніше покупок.

Проте звісно є і недоліки, такі як неможливість приміряти речі, проте це не є величезною проблемою, оскільки доступні сервіси, які дозволяють повернути товар та змінити на інший.

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						6
Зм.	Арк	№ докум.	Підпис	Дата		

Отже, темою кваліфікаційної роботи є розробка сайту «Онлайн-ресурс для роздрібною торгівлі одягом».

При створенні такого проекту важливо врахувати максимум факторів:

- конкуренцію;
- динаміку популярності товарів;
- переваги цільової аудиторії;
- мотиви споживача;
- майбутні методи просування;
- сильні та слабкі сторони власного підприємства.

Інтернет-магазини з продажу одягу відрізняються, крім іншого, великою кількістю товарів та фільтрів, які допомагають вибрати ідеально адаптовані рішення та швидко знайти те, що шукає потенційний покупець. Інформація, доступна в інтернет-магазині, повинна бути правильно оформлена, щоб розвіяти сумніви і відповісти на будь-які питання.

Мета кваліфікаційної роботи – визначення вимог, проектування та розробка ресурсу, де при наявності мережі інтернет можна здійснювати купівлю одягу в роздріб. Дана мета свою чергу передбачає забезпечення зручної роботи користувачів зі всіма складовими сайту, такими як корзина, оформлення замовлення, а також можливість здійснення повернення товару.

Об'єкт дослідження – процес інтернет-обслуговування користувачів з питань замовлення та купівлі одягу.

Предметом дослідження є організація обліку товарів, методи проектування функціоналу та зручного інтерфейсу, а також принципи створення інтернет-магазинів.

Виходячи із мети, об'єкту та предмету дослідження можна виділити такі завдання в ході роботи:

- аналіз та дослідження сфери купівлі-продажу одягу через інтернет;
- розробка технічного завдання;
- проектування, зокрема бази даних;

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						7
Зм.	Арк	№ докум.	Підпис	Дата		

- створення зручного, адаптивного інтерфейсу;
- розробка програмних модулів;
- наповнення контентом;
- виконати тестування.

У майбутньому проект може удосконалюватись, ставши платформою для більшої кількості продавців, які зможуть шукати та замовляти продукти через його онлайн-магазин. Тобто здійснити розширення до маркет-плейсу (платформи, де кожен продавець може зареєструватись та пропонувати свої товари та послуги).

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		8

1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей

Не зважаючи на зростаючу кризу в області інтернет комерції все-таки, хоч і з меншими темпами спостерігається сталий розвиток. Економічна криза штовхає продавців до розвитку онлайн-торгівлі, а також зміна споживчої поведінки: все більше споживачів перед покупкою шукають більш дешевий товар через інтернет. Більш того, на ринку спостерігається тенденція згортання офлайнових майданчиків та істотного їх скорочення. Однак, більша частина ретейлерів все ж таки прагнуть збудувати багатоканальний механізм збуту. Електронні магазини не витісняють традиційні, а розширюють сферу і ринок збуту. У певних сегментах торгівлі відсутність онлайн-магазину є стратегічним недоглядом.

Інтернет-торгівля – процес реалізації фізичних і нефізичних товарів за допомогою спеціалізованих електронних майданчиків, що надають дистанційне оформлення замовлення. Інтернет-торгівля в Росії розвивається досить успішно. За підсумками досліджень ринку інтернет-торгівлі в Україні обсяг продажів постійно збільшується як і збільшується кількість магазинів електронної торгівлі, тобто інтернет-магазинів.

Інтернет-магазин – це ті програми, які дозволяють продавати товари чи послуги через мережу Інтернет та зробити цей процес автоматизованим і швидким. Електронні магазини об'єднують елементи прямого маркетингу та традиційної торгівлі. Основними відмінностями Інтернет-магазину від звичайного – є можливість отримання великої кількості інформації та асортименту продукції. Одразу можна порівняти ціни товарів у багатьох інтернет-магазинах і обрати те, що найбільше підходить. Основним недоліком електронних магазинів є те, що не можна торкнутися товару та оцінити його

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						9
Зм.	Арк	№ докум.	Підпис	Дата		

візуально. Проте, цей недолік компенсується великою кількістю інформації, яку не зможе надати продавець в звичайному магазині.

При аналізі та побудові такого програмного забезпечення, потрібно розуміти, що такий продукт поділяється на дві частини. Принцип роботи дещо схожий на звичайний магазин, оскільки управління сайтом здійснює адміністратор, що можна сказати є здебільшого і власником і продавцем.

Адміністратор інтернет-магазину одягу є основним компонентом в бізнес-процесі, він повинен підтримувати актуальну базу даних про наявність товару, тобто постійно оновлювати її відповідно до сучасних трендів, адже мода змінюється постійно, і якщо така діяльність буде виконуватися непрофесійно та невчасно, інтерес та попит з боку користувачів значно зменшиться. Адміністратор працює з великим об'ємом інформації, веде базу клієнтів, приймає заявки клієнтів, інформує працівників складу про замовлення, передає інформацію про замовлення кур'єру.

Не дивлячись на різноманітність тематик, принцип роботи усіх сайтів магазинів однаковий. Необхідно переглянути наявний товар, додати в корзину при бажанні і вже потім оформити замовлення. Звісно для здійснення цих функцій необхідно спочатку зареєструватися чи авторизуватися, якщо вже є створений профіль. Після реєстрації доступний широкий функціонал сайту.

Інтернет-магазини одягу значно зменшують витрати виробника, заощадивши на утриманні звичайного магазину, розширюють ринки збуту, так само як і розширюють можливість покупця – купувати будь-який товар у будь-який час в будь-якій країні, в будь-якому місті, у будь-який час доби. Це дає інтернет-магазинам одягу перевагу перед звичайними магазинами. Цей момент є істотним під час переходу виробників із «звичайної» торгівлі на «електронну».

Звісно важливим фактором є ціна та якість продукції. Дуже важко знайти маленькі та чесні сайти, які не можуть тягатись з торговими гігантами, але пропонують дійсно хороші ціни і якісну роботу з покупцями.

Проте Internet-магазини мають і недоліки:

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						10
Зм.	Арк	№ докум.	Підпис	Дата		

- невизначеність реального існування товару;
- можлива невідповідність його основним параметрам якості;
- шахрайство при проведенні грошових трансакцій;
- проблеми з доставкою.

Основні вимоги, які ставляться користувачами до Internet-магазину є:

- зрозумілий інтерфейс та зручна система навігації по магазину;
- зручна система посилань, що дозволяє оптимальним способом одержати необхідну користувачеві інформацію;
- мінімальна кількість дій користувача для здійснення покупки.

Основними функціями інтернет-магазину є:

- надання якомога повнішої інформації про представлені товари та послуги;
- прийом та обробка замовлень;
- персоналізація відвідувачів;
- проведення платежів (за умови підключення до платіжної системи);
- збір та аналіз статистичної інформації.

Інтернет-магазин підійде для тих, хто хоче ефективно почати продажі, швидко і недорого організувавши онлайн вітрину з функцією замовлення товарів через Інтернет.

При створенні будь-якого програмного забезпечення є обов'язкові етапи, поступове виконання яких призведе до створення якісного продукту із дотриманням всіх вимог. Таким чином, поступово буде створюватися і сайт інтернет-магазину одягу.

Важливим етапом є проєктування. На цьому кроці визначаються архітектура, компоненти, інтерфейси і ряд інших характеристик системи й кінцевого результату. Цей етап, можна сказати, є фундаментом програмного забезпечення. Формується представлення про дизайн сторінок сайту, розміщення контенту. Це все виконується за допомогою створення макетів, на

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						11
Зм.	Арк	№ докум.	Підпис	Дата		

основі яких і ведеться подальша розробка програмного забезпечення. Визначається також і модель за якою будується сайт, тобто архітектура.

Кодування сайту – наступний етап, який є не менш важливим. Тут велику роль відіграють уміння та навички самого розробника програмного забезпечення, а саме – які інструменти використовуються, як вони поєднуються, чи правильно будується раніше визначена архітектура, чи виконується дотримання правил побудови коду.

Звичайно, важливу роль грає і кодування інтерфейсу, дизайну сайту, яке здійснюється на основі раніше створених макетів. Він повинен бути зручним і повністю зрозумілим для користувача. Не повинно виникати ситуацій, коли незрозуміло, що далі робити чи натискати. Інтернет-магазини одягу не є чимось новим для людей, і якщо виникають такі ситуації, то некоректно говорити про невміння людей працювати із таким програмним забезпеченням. Потрібно робити висновки, що сайт не є досконалим і потрібно щось змінювати, адже від цього також залежить попит користувачів.

Наступний етап – це тестування. Це обов'язковий крок перед початком використання програмного забезпечення. Тестування створюється і виконується для попередження та виправлення різнопланових помилок. Для інтернет-магазину одягу – це необхідний етап, оскільки операції проводяться із замовлення та грошовими переказами. Якщо якийсь елемент такої системи працює некоректно, це може призвести до грошових втрат та незадоволеності користувачів, в разі отримання не такого товару, який вони замовляли. Тому важливо перевірити кожен елемент, особливо чи правильно дані заносяться в базу даних, чи відображається саме та інформація, яка відповідає дійсності.

Розрізняють ручне та автоматизоване тестування. Ручне тестування виконується самою людиною, коли наприклад вона вводить певні дані і отримує відповідь, на основі якої формує висновки.

Автоматизоване тестування виконується за допомогою попередньо запрограмованих тестів, і має ряд переваг, таких як:

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						12
Зм.	Арк	№ докум.	Підпис	Дата		

- відбувається швидко, заощаджує багато часу;
- детально виконані скрипти виключають можливості виникнення помилок під час процесу тестування;
- після того, як тести створені, виконання тестування майже не вимагає часу працівників.

Недоліками автоматизованого тестування є:

- процес створення таких тестів вимагає додаткових зусиль з боку розробників;
- при зміні програмного забезпечення необхідне оновлення сценарію тесту;
- інколи можуть пропускати ті помилки, які б людина точно помітила.

Тому, для впевненості у продукті доцільно провести як ручне так і автоматизоване тестування.

Для ознайомлення користувачів з програмним продуктом важливо також надати певні відомості чи інструкцію, як його використовувати. Звісно цей пункт виконується, якщо програмний продукт створюється для продажу і необхідний опис дій для адміністратора. Звичайно, інтернет-користувач не буде читати такі інструкції, все повинно бути і так зрозумілим, тому для нього цей пункт не є доцільним.

Після виконання всіх цих кроків, на виході отримується програмний продукт, в даному випадку інтернет-магазин одягу, який буде повноцінним та працювати коректно.

Звісно під час виконання цих етапів, можливе коректування функціоналу, доповнення різними інструментами, які будуть сприяти тільки покращенню сприйняття цього сайту користувачем. Тому навіть після етапу введення в експлуатацію, нерідко вводяться якісь оновлення, покращується швидкодія, наповненість сторінки, виправляються помилки, адже інколи виникають ситуації, які важко передбачити в процесі формування сайту.

Отже, інтернет-магазини одягу є дуже поширеними в наш час. Вони постійно оновлюються і покращуються, тому розробникам такого програмного

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						13
Зм.	Арк	№ докум.	Підпис	Дата		

забезпечення дуже важливо слідкувати за цим, будувати продукт відповідно сучасним тенденціям, покращувати функціонал, вносити оновлення та зміни, щоб попит та зацікавленість користувачів не втрачався.

1.2 Аналіз наявного програмно-технічного забезпечення предметної області

В сучасному інформаційному просторі, зокрема в сфері комерційної торгівлі існує надзвичайно велика кількість різноманітних платформ, ресурсів, майданчиків для ведення бізнесу. Торгівля одягом не стала виключення, адже в мережі інтернет існує дуже багато інтернет-магазинів для продажу одягу.

При порівнянні різних інтернет-магазинів одягу спрямовується увага на такі фактори:

- безпосередня робота ресурсу, тобто наскільки зручно користуватись, чи доцільна та продумана навігація, розміщення різноманітних елементів;
- наявний асортимент товарів та послуг – чи є великою кількістю товарів та послуг, а також їх доступність для споживача;
- ціни – порівняння вартості товарів з конкурентами, наявність спецпропозицій;
- підтримка та консультації – швидкість відповідей на звернення покупця, опрацювання замовлення, допомога у вирішенні труднощів;
- зручність оплати – великий вибір способів оплати, можливість розтермінування та кредиту;
- доставка – швидкість опрацювання, кількість способів доставки;
- акції, знижки, розпродажі – можливість купити дешевше в інтернет-магазині;
- гарантійні зобов'язання – повернення товару, ремонт та обслуговування.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						14
Зм.	Арк	№ докум.	Підпис	Дата		

Для аналізу існуючого програмного забезпечення, що розроблено для полегшення ведення торгівлі в мережі Інтернет, розглянемо найбільш популярні з них. Це платформи, маркет-плейси та майданчики, на яких пропонується продає різноманітного товару, зокрема гуртом та вроздріб.

Вважається, що Intertop – кращий інтернет-магазин недорогого одягу в Україні і є так званою легендою українського інтернет-ртейлу. Із самого початку вийшов на ринок та позиціонувався як офлайн-магазин взуття. Однак, із плином часу даний інтернет-магазин набув свого розвитку і тепер на даному ресурсі можна замовити одяг з доставкою по Україні із післяплатою або по передоплаті. Дуже зручним саме у цьому магазині є те, що товар можна безкоштовно замовити у магазин-офлайн, прийти поміряти і купити, якщо підійде. Якщо ж товар не підходить, то його не потрібно купувати, він залишається у магазині і вартість доставки замовник не сплачує.

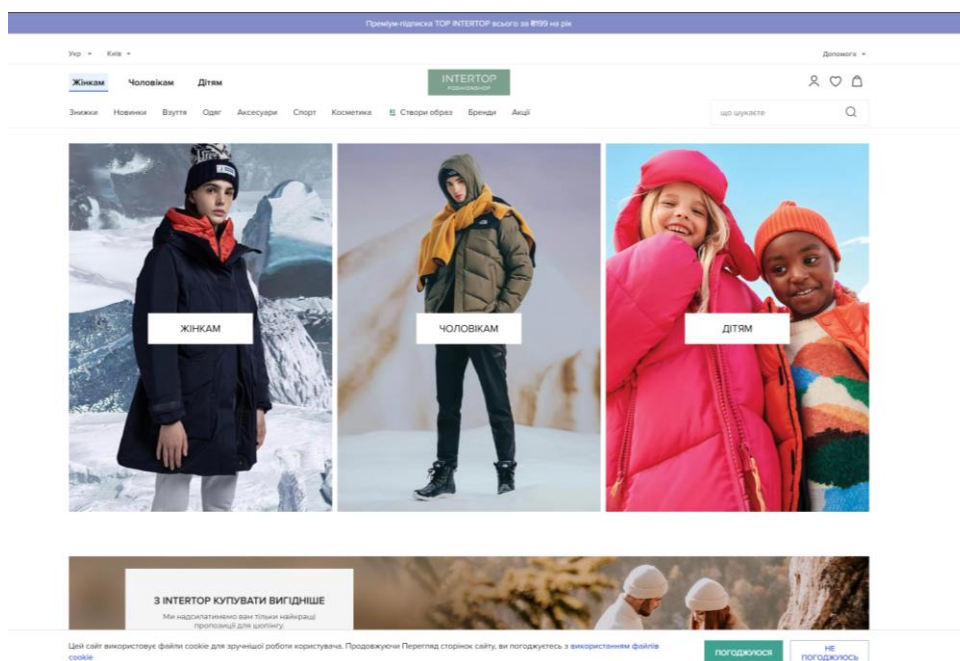


Рисунок 1.1 – Головне вікно інтернет-магазину Intertop

(<https://intertop.ua/uk-ua/>)

При ознайомленні із ресурсом, виділено такі переваги:

- великий асортимент;
- недорогий одяг;

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						15
Зм.	Арк	№ докум.	Підпис	Дата		

- накладений платіж;
- безкоштовна доставка замовлення дорожче певної суми.

Наступний онлайн-ресурс, який піддався аналізу – це dressa – кращий інтернет-магазин недорогого одягу з післяплатою на території України. Даний ресурс відрізняється тим, що тут розміщено відносно недорогий жіночий одяг, який можуть доставити у будь-яку точку світу (Рисунок 1.2).

Переваги даного ресурсу такі:

- великий асортимент;
- склад в Україні;
- накладений платіж;
- недорогий одяг.

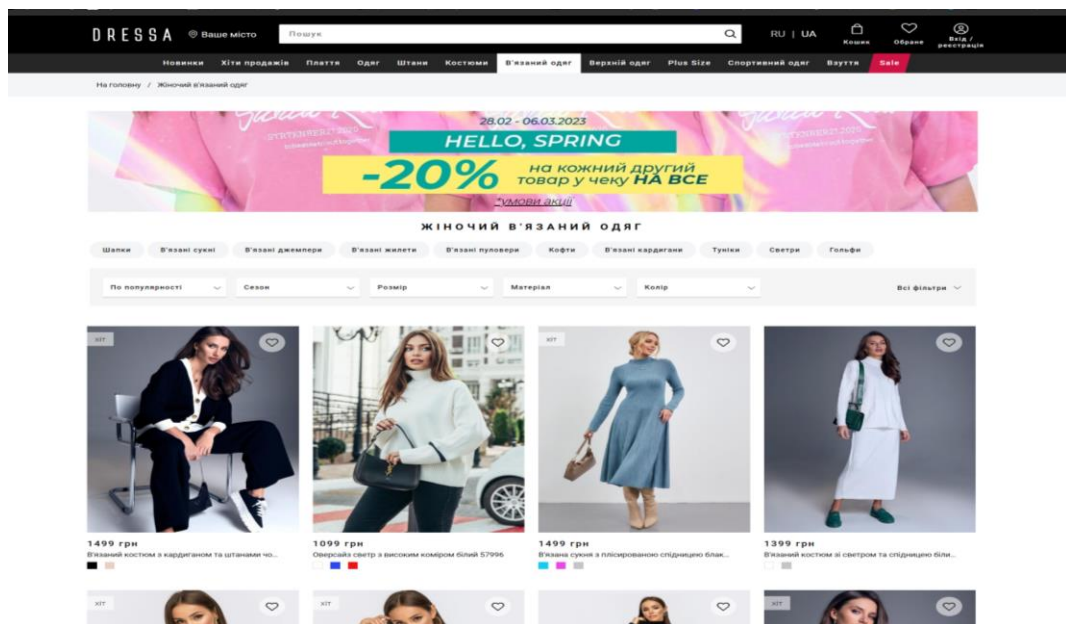


Рисунок 1.2 – Інтернет-магазин dressa (<https://dressa.com.ua/uk>)

Наступний інтернет-магазин, що вартий уваги це Issaplus, який вважається одним із кращих інтернет-магазинів, що представляє та продає одяг українських брендів одягу із доставкою по Україну та закордон і працює на ринку вже біля 10 років.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						16
Зм.	Арк	№ докум.	Підпис	Дата		

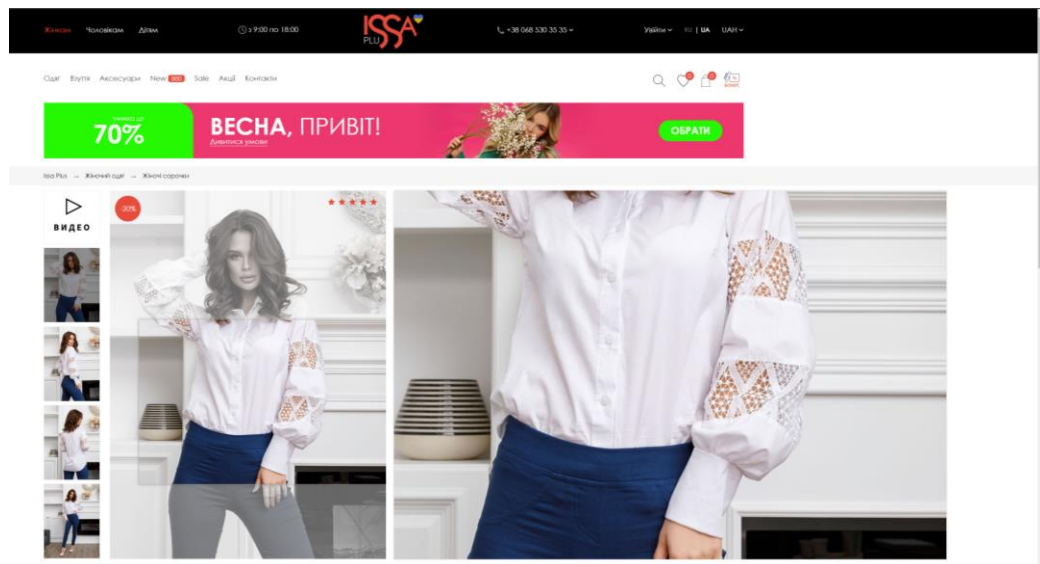


Рисунок 1.3 – Інтернет-магазин Issaplus (<https://issaplus.com/ua/>)

Переваги даного Інтернет-магазину:

- великий та різноманітний асортимент товару;
- наявність можливості розраховуватись накладеним платежем;
- досить велика кількість реальних відгуків про товари та послуги;
- різноманітні акції та сейли діють на постійній основі;
- великі склади в різних містах України.

Таким чином, можна стверджувати, що інтернет-комерція пропонує дуже широкий спектр веб-ресурсів для здійснення торгівлі, зокрема й одягом. Всі ці засоби мають свої переваги та недоліки, свій функціонал, а також є більш популярними та менш користуються увагою та попитом.

1.3 Визначення вимог до онлайн-ресурсу

Створення інтернет магазину одягу розпочинається з аналітики та визначення стратегії. Щоб створити якісний робочий проект, необхідно чітко розуміти, до чого йти і, що буде отримано в результаті. Аналітика дозволяє оцінити конкурентів, зрозуміти сильні та слабкі сторони, вивчити цільову аудиторію, її потреби. За підсумками зібраних даних розробляється стратегія. У

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						17
Зм.	Арк	№ докум.	Підпис	Дата		

результаті формується чітке бачення концепції, завдань та методів реалізації проекту.

Далі за допомогою графічного дизайну створюються елементи ресурсу та його повна промальовка. Підбираються потрібні кольори, форми, шрифти, визначається розміри елементів. Особлива увага приділяється зручності використання ресурсу. Він має бути зрозумілим, цікавим, простим, але водночас стильним та актуальним на довгий період часу.

Для наглядності створюється прототип, який допомагає побачити, як виглядатиме онлайн-ресурс у результаті. Він включає дизайнерські елементи, кольори, форми, вид інтерфейсу, технічні можливості. По суті сайт повністю створюється схематично, визначаються всі ідеї. Для цього використовуються спеціальні програми. Після затвердження прототипу створюється технічне завдання, далі відбувається реалізація проекту за створеним технічним завданням, тобто за допомогою кодингу відбувається технічне розроблення інтернет-магазину меблів. Враховуються всі особливості сфери бізнесу, впроваджуються необхідні інструменти, за необхідності інтегруються сторонні послуги. Після розробки проводиться тестування, запуск та можлива додаткова підтримка та просування ресурсу.

В ході розробки програми повинні бути підготовлені: текст програми, опис програми, програма і методика випробувань, керівництво користувача, техніко-економічне обґрунтування.

З огляду на обсяг проекту завдання передбачається вирішувати поетапно. При цьому модулі ПЗ, створені в різний час, повинні передбачати можливість нарощування системи і бути сумісні один з одним, тому документація на прийняте експлуатаційне ПЗ повинна містити повну інформацію, необхідну для роботи з ним програмістів.

Написання програмного продукту – це багатоетапний, складний процес. Він складається з певних стадій. Стадії розробки програмного продукту описано у таблиці 1.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						18
Зм.	Арк	№ докум.	Підпис	Дата		

Після виконання всіх цих стадій, у результаті буде готовий сайт інтернет-магазину одягу. Продукт буде повноцінним та готовим до експлуатації.

Таблиця 1 – Стадії і етапи розробки програмного продукту

Стадія	Етапи робіт	Опис робіт
Дослідження предметної області	Аналіз предметної області. Огляд існуючих рішень. Розробка ТЗ	На цій стадії виконуються роботи які передують проектуванню програмного забезпечення. Розробляється технічне завдання та визначаються основні вимоги до системи. Виконується аналіз схожого існуючого ПЗ.
Проектування системи	Проектування архітектури, інтерфейсу, бази даних. Вибір технологій.	Ця стадія передує кодуванню програмного забезпечення. Виконується детальне проектування всіх компонентів системи, на основі якого створюється проект.
Кодування	Кодування всіх елементів системи.	Після виконання цієї стадії, результатом є готовий, повнофункціональний програмний продукт.
Тестування програмного продукту	Ручне тестування. Модульне тестування.	Тестування виконується для перевірки на помилки чи неточності при роботі з сайтом. Після чого ці недоліки виправляються та проводиться тестування до тих пір, поки всі елементи системи не будуть працювати правильно.

Продовження Таблиці 1

Оформлення програмної документації		Документація формується у вигляді детального звіту, де описується кожний крок чи етап роботи над програмним продуктом. Для кращого сприйняття окремих етапів будуються діаграми, на яких зображено принципи функціонування системи.
------------------------------------	--	---

При виборі теми для проекту головним і найважливішим фактором є інтерес користувача до даного виду продукту. Покупки в Інтернеті стають все більш поширеними і популярними. На це є багато причин, оскільки інтернет-магазини одягу пропонують масу переваг.

По-перше, це економить час. Наприклад, якщо ви йдете на ринок, магазин чи торговий центр, вам потрібно враховувати, скільки часу знадобиться, щоб дістатися до потрібного магазину. Далеко не факт, що потрібний товар є в наявності в конкретній торговій точці - продавець може відправити покупця в іншу частину міста. Все це займає багато часу. Щоб вибрати саме те, що підходить саме вам, вам доведеться відвідати кілька магазинів і ознайомитися з їх асортиментом. Завдяки таким послугам, як інтернет-магазин, усе доступне лише за декілька кліків.

По-друге, можна значно заощадити. Слід враховувати, що для відкриття магазину вам потрібно буде орендувати приміщення, склади, найняти продавців, усе це потребує капітальних вкладів від власника, тому ціна товару буде вищою. Для інтернет-магазину достатньо онлайн-середовища та складу.

Звичайно, велика різноманітність є головним плюсом, і все це допомагає швидше знайти одяг, який вам подобається та підходить. Шукати товар в Інтернеті набагато зручніше, ніж у торговому центрі.

					КвРІПЗ.200139.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		20

Третя перевага полягає в тому, що це не має нічого спільного з місцем, часом і погодою, тому що інтернет-магазин працює цілодобово, на відміну від інших магазинів. Ви навіть можете замовити доставку кур'єром за вказаною адресою, нікуди не їздячи за своїми покупками.

Перевагою є ексклюзивність товару. В Інтернеті можна знайти рідкісні та унікальні речі - торгові центри більше орієнтуються на актуальні речі, які часто розкуповуються. При необхідності інтернет-магазин може запропонувати товари, замовлені з-за кордону (якщо товару немає в наявності в країні). Через усі ці фактори багато хто насторожено ставиться до таких покупок.

Упередженість проти онлайн-шопінгу існує в основному серед тих, хто ніколи не робив онлайн-шопінг, тому що їм важко уявити, як купувати, не побачивши і не помацавши продукт. Велику роль у цьому відіграє моральне небажання людей робити покупки в Інтернеті та невіра в те, що Інтернет – це щось нове й незвідане.

Ще однією причиною небажання звертатися в інтернет-магазин є те, що процес купівлі анонімний і немає прямого спілкування покупця з продавцем, тому будь-які проблеми, що виникають під час покупки або експлуатації товару, покупець не бачить гарантії, що магазин зможе їх вирішити. Люди бояться купувати в інтернет-магазинах, мабуть, тому, що у них сформувався стереотип, що їх обов'язково обдурять: не відправлять те, що замовили і заплатили.

Щоб потенційні покупці не сумнівались, що в магазині є постійна підтримка покупців після здійснення покупки/продажу, необхідно надати детальну інформацію про те, куди звертатися у разі виникнення проблем чи питань. Частота купівлі товарів через Інтернет безпосередньо пов'язана з видами товарів, які пропонує магазин, і конкретними умовами попиту на товари.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						21
Зм.	Арк	№ докум.	Підпис	Дата		

1.4 Технічне завдання для розроблюваного онлайн-ресурсу

Вступ

Робота виконується в рамках проекту розробки веб-ресурсу для роздрібної торгівлі одягом. Технічне завдання було спроектовано у відповідності до діючих стандартів щодо розробки програмного забезпечення.

Проект створюється з метою здійснення онлайн-замовлень одягу споживачами.

Підставою для розробки є «Завдання на кваліфікаційну роботу».

Найменування розробки: «Онлайн-ресурс для роздрібної торгівлі одягом».

Програмний продукт (Інтернет-магазин) створюється у вигляді сайту, що має зручну навігацію по окремих розділах та видах одягу, повинен мати чітку структуру із одночасним простим та зручним інтерфейсом. Також передбачено отримання необхідної інформації про товар, тобто зворотній зв'язок з адміністрацією сайту.

Користувачами програми є адміністрація магазину, що здійснює облік товарів, їх додавання та редагування а також користувачі інтернет-магазину, що здійснюють замовлення.

Передбачається розробка таких функцій для звичайного користувача як:

- реєстрація та авторизація на сайті для подальшої роботи з ним;
- перегляд товарів;
- здійснення замовлення;
- при здійсненні замовлення, є можливість обрати певні параметри одягу та спосіб доставки;
- робота з корзиною, для зручного вибору товару;
- надсилання коментарів.

Основна функція системи – це оформлення замовлень на товари, здійснення покупок. Для цього користувача потрібно забезпечити зручними сервісами для входу на сайт, тобто розробити сторінки реєстрації та авторизації.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						22
Зм.	Арк	№ докум.	Підпис	Дата		

Потрібно забезпечити зручну роботу із каталогом товарів, розробити систему фільтрації та пошуку. У користувача не повинно виникати незрозумілостей при роботі з цією частиною сайту.

Потрібно розробити сторінку перегляду кожного товару із можливістю залишати коментарі, та додавати товар до кошика, вказавши при цьому додаткові параметри товару.

Необхідно створити сторінку де виводяться всі товари кошика, переглянувши які, користувач зможе оформити замовлення. Для оформлення замовлення необхідно додати певні поля для вказання додаткових параметрів таких як спосіб доставки.

Врешті решт необхідно створити систему для оплати замовлень. Для адміністратора необхідно створити систему додавання товарів до бази даних. Також потрібно додати функції редагування та видалення цих даних.

Потрібно додати функціонал, з допомогою якого адміністратор зможе керувати ролями, що забезпечить різний доступ до системи користувачам.

Для кращого розуміння технічного завдання та принципів функціонування даної системи, в залежності від різних ролей, створено діаграми використання на мові UML. Дана діаграма зображена у додатку Б.

Функціональні вимоги

– Авторизація користувачів в системі: В системі повинна бути представлена можливість реєстрації користувача та присвоєння йому відповідної ролі (замовник, адміністратор). В іншому випадку користувач є гостем і не може користуватися всіма функціями сайту:

- розробка структури сайту і схем сторінок;
- розробка структури сторінки каталогу товарів;
- можливість збереження певних товарів у кошику до оформлення замовлення;
- можливість зворотного зв'язку із адміністрацією сайту;

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						23
Зм.	Арк	№ докум.	Підпис	Дата		

- підготовка прав на різні типи операцій на сайті для груп користувачів;
- створення необхідної кількості розділів і інформаційних блоків;
- узгодження необхідних характеристик товарів для порівняння і виводу;
- настройка сторінок замовлення і кошика;
- обробка зображень, наповнення сайту;
- відображення результатів роботи.

Нефункціональні вимоги

- система повинна бути простою, не вимагати додаткових знань та навичок з боку користувачів, оскільки розрахований всіх інтернет-користувачів;
- час відповіді системи для звичайних запитів не повинен перевищувати 5 секунд, а для більш складних запитів – 10 сек;
- система повинна бути надійною, не давати збоїв та помилок;
- система повинна підтримувати мінімум 100 одночасно працюючих користувачів, пов'язаних з спільною базою даних;

Система повинна задовольняти вимогам на комп'ютері, що знаходиться в наступній мінімальній комплектації:

- мінімальний об'єм пам'яті 28 Мб;
- мінімальна тактова частота 600 MHz ;
- операційна система Windows XP або пізніші версії;
- вимоги до операційного середовища (для сервера UNIX, до web-сервера Apache);
- браузер (Opera, Chrome).

Вимоги до апаратного забезпечення:

– програмний комплекс (ПК) розробляється за допомогою методів мови C# та фреймворку ASP.Net Core. Дане середовище розробки обране в результаті аналізу функціональних можливостей різних середовищ розробки. Воно має такі переваги:

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						24
Зм.	Арк	№ докум.	Підпис	Дата		

- висока ефективність і швидкодія коду;
- наочність та зручність інтерфейсу середовища;
- об'єктно-орієнтованість мови.

Вимоги до надійності

Розроблюване ПЗ повинно:

- самовідновлюватись після якихось перебоїв у роботі;
- мати парольний захист під час запуску програми;
- мати обмеження несанкціонованого доступу до даних;
- мати наявне розмежування прав користувачів;
- мати наявне розмежування функцій користувачів.

Вимоги до транспортування та зберігання

- програма поставляється захищених носіях інформації;
- програмна документація поставляється в електронному та друкованому вигляді;
- умови експлуатації програмного забезпечення збігаються з умовами експлуатації комп'ютера.

1.5 Висновки до 1-го розділу. Постановка задачі

Отже, в першому розділі даної кваліфікаційної роботи було здійснено детальний аналіз предметної області із виокремленням її меж, функціональних та нефункціональних вимог, проаналізовано існуючі онлайн-ресурси з продажу одягу онлайн, розроблено технічне завдання.

При порівнянні різних інтернет-магазинів одягу спрямовується увага на такі фактори: безпосередня робота ресурсу, зручний інтерфейс користувача, можливість оплати та доставки, наявність адмінпанелі, можливість авторизації та реєстрації користувача.

Для отримання позитивного результату поставлено такі задачі: аналіз та дослідження сфери купівлі-продажу одягу через інтернет;

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						25
Зм.	Арк	№ докум.	Підпис	Дата		

- розробка технічного завдання;
- проектування, зокрема бази даних;
- створення зручного, адаптивного інтерфейсу;
- розробка програмних модулів;
- наповнення контентом;
- виконати тестування.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		26

2 ПРОЕКТУВАННЯ ОНЛАЙН-РЕСУРСУ

2.1 Архітектура та функціональна структура веб-ресурсу

Проектування онлайн-ресурсу відноситься до важливого етапу розробки програмного забезпечення. Далі подано архітектури, що існують в сучасних умовах та відповідно до яких можна здійснювати побудову сайту.

До першого типу відносить так званий клієнт-серверний тип архітектури, що є нині найбільш поширеною архітектурою, в якій виконана, мабуть, більшість працюючих інформаційних систем.

Архітектура "клієнт-сервер" є класичним підходом до систем, де вона складається з двох компонентів - клієнта і сервера. Ці компоненти постійно взаємодіють між собою: клієнт формує запити і відправляє їх серверу, а сервер обробляє ці запити і надає відповідь клієнту у вигляді веб-сторінок чи повідомлень.

В іншій моделі, відомій як "модель сервісу", сервер також має складну структуру, яка складається з декількох серверів. Вона представляється клієнту як єдина система. Ця архітектура особливо корисна в ситуаціях, коли найменша затримка або відмова в обслуговуванні є критичною.

Іншим прикладом є архітектура "підключення через проху", яка використовується у звичайних системах з браузером, проху-сервером і веб-сервером. Тут клієнт не з'єднується безпосередньо з сервером, а підключається до посередника, який передає запити серверу від імені клієнта.

В архітектурі "сервер ініціює з'єднання" сервер починає діалог, "тискаючи" інформацію на клієнта. У цьому випадку роль клієнта полягає в реакції і перегляді повідомлень, надісланих сервером.

З усіх цих архітектур для реалізації інтернет-магазину одягу найбільш вдалою є архітектура "клієнт-сервер".

Для розробки сайту інтернет-магазину одягу планується використовувати архітектуру "клієнт-сервер". У цій архітектурі сервером є комп'ютер, що містить бази даних, систему керування базами даних та пов'язане з ними програмне

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						27
Зм.	Арк	№ докум.	Підпис	Дата		

забезпечення. Він налаштований для надання користувачам доступу до бази даних через інформаційну систему. Клієнти, що працюють з даними (можуть бути розташовані на різних комп'ютерах у мережі), надсилають відповідні запити серверу, який їх отримує, обробляє і надсилає відповідь клієнтам.

Для проєкту доцільно використовувати архітектуру MVC, яка дозволяє розділити код програми на три частини: модель (Model), вид або представлення (View) і контролер (Controller). Цей поділ спрощує роботу з великим обсягом коду. Якщо код писати одним великим скриптом, він стає складним для розуміння і внесення змін без помилок.

Модель містить всю логіку додатка, зберігає і обробляє дані, але не взаємодіє безпосередньо з користувачем (доступ до моделі можливий тільки з коду, викликаючи її функції). Наприклад, збереження інформації в базі даних або перевірка правильності введених даних у форму - це завдання моделі. Однак, отримання даних від користувача, відображення інформації на екрані або обробка натискання кнопок - цього немає у моделі. Можна сказати, що моделі є своєрідними класами. У цьому проєкті планується створення таких основних моделей:

- модель товару;
- модель кошика;
- модель складу;
- модель чату
- модель коментарів та ін.

Представлення відображає дані, які були передані йому. У веб-додатку зазвичай воно складається з HTML-шаблонів сторінок, а в десктопних або мобільних додатках - з відповідних елементів інтерфейсу. Представлення виконує код, який відповідає за відображення інформації на екрані, включаючи зображення кнопок та інших елементів інтерфейсу.

Контролер відповідає за обробку запитів, що надійшли від користувача. У веб-додатку контролер зазвичай розбирає параметри HTTP-запиту змінними \$

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						28
Зм.	Арк	№ докум.	Підпис	Дата		

_POST / \$ _GET, звертається до моделі для отримання або зміни певних даних, а після цього викликає представлення, щоб відобразити результат виконання запиту. Значить, контролер встановлює зв'язок між моделлю та представленням. В рамках цього проєкту передбачається створення трьох контролерів: User, Admin, Account.

Контролер User відповідає за всі дії, доступні звичайному користувачеві, тобто покупцеві. В ньому містяться методи, які викликають представлення, на яких відображається основна інформація про товари, такі як каталог товарів та сторінка з інформацією про окремий товар. Крім того, цей контролер відповідає за оформлення та керування корзиною, а також за оформлення замовлень.

Контролер Account відповідає за особисті дані користувачів. Основні його методи включають реєстрацію, авторизацію та вихід з сайту. Усі методи цього контролера також доступні звичайному користувачеві.

Контролер Admin відповідає за всі операції, які виконує адміністратор на сайті. Це методи, пов'язані з додаванням, редагуванням та видаленням товарів. Контролер керує такими сторінками, як сторінка з виведенням всіх товарів та інформацією про них, сторінки додавання та редагування товарів. Сторінки додавання мають варіації для кожного виду товару.

2.2 Визначення основних модулів

Для ефективної роботи розроблюваного продукту необхідно визначити основні модулі, з яких він буде складатись. У нашому випадку ці модулі - реєстрація та авторизація, корзина і адмінпанель.

Реєстрація та авторизація є початковими етапами при роботі з сайтом. Після їх виконання користувач має доступ до механізму замовлення товарів. До авторизації користувач є гостем у системі і має обмежені можливості. В системі присутній механізм перевірки правильності введеного пароля та логіну.

Перегляд товарів користувачів здійснюється через початкову сторінку сайту, на якій доступна інформація про всі товари, такі як найменування,

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						29
Зм.	Арк	№ докум.	Підпис	Дата		

модель, ціна тощо. Кожен товар може мати різні варіації, такі як кольори та розміри, і для кожної варіації є відповідні фотографії. Для зручності навігації по товарам на основній сторінці є певна кількість фільтрів, а також можливість сортування товарів.

Додавання товарів в козину відбувається після попереднього етапу. Користувач може додати товар безпосередньо на основній сторінці, де доступні всі товари, або після перегляду додаткової інформації та характеристик на відповідній сторінці товару. Під час додавання до козину користувач вказує колір та розмір товару. Після цього в козині доступні всі раніше додані товари, і користувач може видалити товар з козину, змінити кількість або параметри товару, такі як розмір чи колір.

Оформлення замовлення здійснюється на основі товарів, які були додані до козину. Користувач може вибрати, який саме одяг замовити зі списку товарів у козині. У цій функції користувач заповнює обов'язкові поля, такі як спосіб доставки та спосіб оплати. Результатом є створення чеку, який потрібно оплатити, після чого замовлення буде надіслане користувачу. Також відображається приблизний час доставки, який залежить від різних факторів, включаючи відстань між містами. Вартість доставки також може змінюватись в залежності від цих факторів. Адміністратор може встановити певну суму, при перевищенні якої користувач отримує акційну пропозицію.

Користувач також має можливість взаємодіяти з адміністратором сайту через систему повідомлень.

Крім того, користувач може коментувати товари, залишати відгуки щодо якості товарів, доставки та інших аспектів. Адміністратор має доступ до додаткових функцій, таких як:

- редагування чи видалення інформаційних записів про товари;
- додавання інформації про нові товари та категорії;
- редагування, створення та призначення ролей користувачам;
- ведення переписки із користувачами.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						30
Зм.	Арк	№ докум.	Підпис	Дата		

Адміністратор має повний доступ до всієї інформації про товари та може здійснювати операції з ними, такі як додавання, видалення та редагування записів. Процес додавання нового товару в систему складається з кількох етапів.

На початку адміністратор вносить основні дані про одяг, такі як модель, назва, бренд і категорія. Категорія вибирається з попередньо визначеного списку. При створенні товару можна вказати лише номер категорії зі списку доступних.

Після цього адміністратор додає різновиди товару, яких може бути кілька. Оскільки сам товар може мати різні кольори і розміри, для кожного різновиду вказується фото (можна додати кілька фотографій). Також потрібно вказати кількість одиниць для кожного різновиду.

Процес додавання товару розбитий на етапи, оскільки в даному випадку виникають зв'язки один до багатьох, і розділення цих даних сприяє зручній та коректній взаємодії адміністратора та користувача з сайтом.

Редагування та видалення основних даних про товар, його різновидів і категорій здійснюються окремо. Але якщо видаляється певна категорія товару, то всі записи про товари, пов'язані з цією категорією, також видаляються. Це здійснюється за допомогою зв'язків між даними, виконується каскадне видалення.

У системі присутні різні ролі, які мають різний рівень доступу до окремих частин системи. Наприклад, дії, доступні адміністратору, не будуть видимі для звичайних користувачів. Розподіл ролей та їх редагування також доступні лише адміністратору сайту. У разі потреби можна створювати нові ролі.

Графічно відображено логіку програми на діаграмі поведінки у графічній частині кваліфікаційної роботи.

2.3 Проектування логічної моделі бази даних

При проектуванні логічної моделі необхідно створити таблиці та встановити зв'язки між ними. Основні таблиці, які передбачаються, включають:

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						31
Зм.	Арк	№ докум.	Підпис	Дата		

- Таблиця користувачів: містить інформацію, яка вводиться під час реєстрації користувача в системі.
- Таблиця товарів: містить основну інформацію про товари.
- Таблиця кольорів товарів: містить додаткову інформацію про товари, таку як кольори та зображення.
- Таблиця розмірів: містить інформацію про наявні розміри підвидів товарів.
- Таблиця корзини: містить інформацію про товари, які користувач додав до корзини.
- Таблиця ролей: зберігає інформацію про ролі користувачів, яка визначає доступ до окремих функцій.
- Таблиця замовлень: зберігає інформацію про замовлення, здійснені користувачами.
- Таблиця коментарів: містить всі коментарі, які користувачі залишають щодо конкретного товару.
- Таблиця зображень: містить всі зображення товарів.

Після побудови логічної моделі бази даних можна перейти до створення фізичної моделі бази даних. Фізична модель визначає розташування даних у зовнішній пам'яті і залежить від обраної системи керування базами даних (СКБД). Оскільки використовується СКБД з реляційною моделлю даних, необхідно перенести таблиці разом з атрибутами та зв'язками в середовище СКБД, дотримуючись вимог до відповідних об'єктів бази даних.

На етапі фізичного проектування слід звернути особливу увагу на забезпечення цілісності бази даних. У СКБД цілісність даних забезпечується обмеженнями цілісності, які встановлюють правила для даних та їх зв'язків. Дотримання цілісності даних є важливим аспектом фізичного проектування бази даних. Схему бази даних подано на рисунку 2.1.

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						32
Зм.	Арк	№ докум.	Підпис	Дата		

правилами, яким буде слідувати кожна сторінка сайту. Ці правила можна зобразити за допомогою макету, який зображено на рисунку 2.2.



Рисунок 2.2 – Макет побудови веб-сторінок

Усі дані сторінки розміщені в тілі сторінки, як показано на попередньому макеті.

Для зручності навігації між сторінками сайту на кожній сторінці присутнє меню, розташоване у верхньому колонтитулі.

Основні сторінки, які складатимуть сайт, включають:

- Сторінку каталогу товарів.
- Сторінки реєстрації та авторизації.
- Сторінку перегляду конкретного товару та його характеристик.
- Сторінку корзини.
- Сторінку оформлення замовлення.

На макеті сторінки каталогу товарів, зображеному на рисунку 2.3, представлений перелік усіх товарів, де відображаються основні дані та зображення товарів. Вище товарів розташована система пошуку, яка відображає

товари залежно від введеного значення у полі пошуку. З лівого боку є можливість вибору товарів за певними критеріями.



Рисунок 2.3 – Макет сторінки каталогу товарів

На рисунку 2.4 зображено макет сторінки перегляду одного товару.

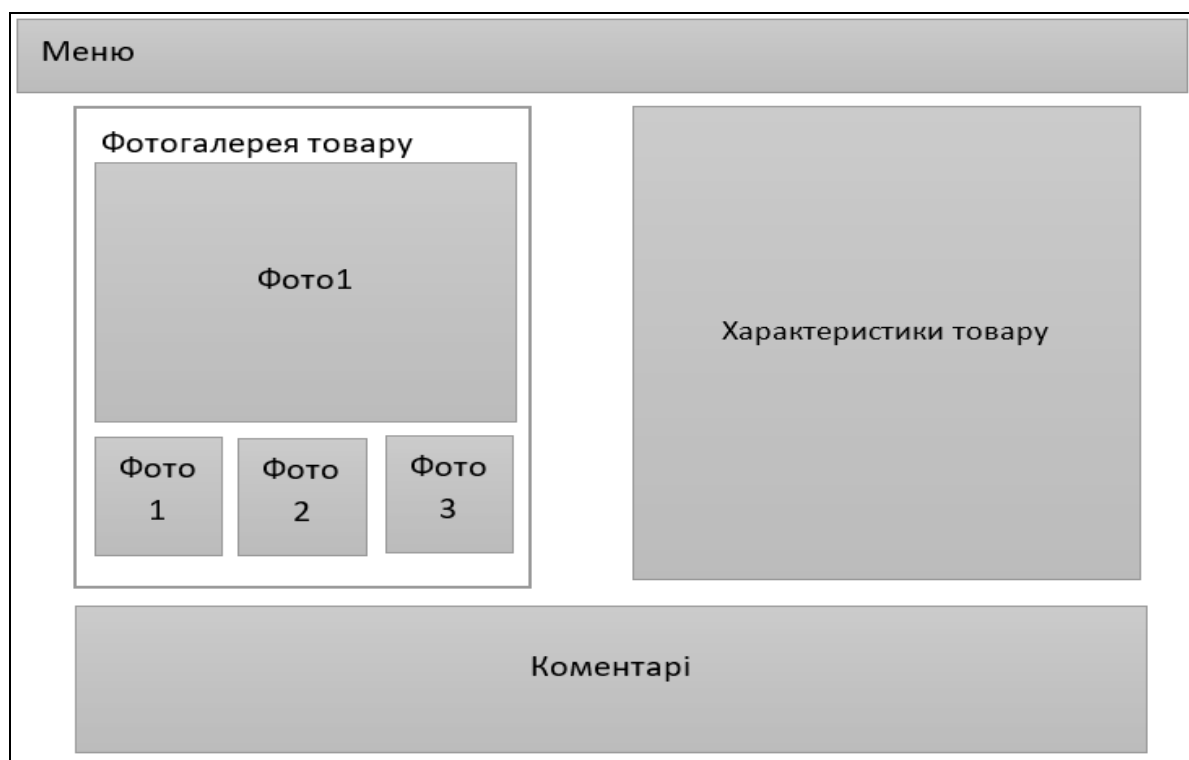


Рисунок 2.4 – Макет сторінки перегляду одного товару

Макет сторінку корзини користувача зображено на рисунку 2.5.



Рисунок 2.5 – Макет сторінки із корзиною

Отже, як бачимо інтерфейс користувача відіграє досить важливу роль при роботі із веб-застосунками, оскільки від цього напряму залежить чи будуть користувачі постійно ним користуватись і чи буде ця робота зручною та популярною.

2.5 Аналіз та вибір технологій і методів реалізації веб-ресурсу

Після завершення процесу проєктування архітектури, яка у цьому проєкті обрана як «клієнт-сервер», наступним кроком у створенні інтернет-магазину є вибір мови програмування та фреймворку для розробки програмного забезпечення.

Оскільки планується створення веб-ресурсу, важливо обрати мову програмування, яка є зручною і придатною для побудови сайтів. Існує безліч

різних мов, що дозволяють досягти цієї мети, таких як Java, Python, Ruby, Php, C#.

Java є популярною мовою програмування для кросплатформових додатків. Розробники на Java можуть створювати програми на своєму комп'ютері, а потім запускати їх на різних платформах, таких як телефони, сервери та інші. Крім того, Java підтримує велику кількість різноманітних платформ.

Php є скриптовою мовою програмування, яка була створена для генерації HTML-сторінок на стороні веб-сервера. Часто використовується при розробці веб-застосунків.

Ruby є мовою програмування, яка інтерпретується, повністю об'єктно-орієнтована і має чітку динамічну типізацію. Ця мова є досить ефективною для розробки програм. Багатоплатформова реалізація інтерпретатора Ruby поширюється як вільне програмне забезпечення.

Python є мовою програмування, яка досить проста у використанні, і її легко освоїти. За допомогою синтаксису цієї мови можна швидко розробляти прикладні програми у різних галузях на більшості платформ.

C# є мовою програмування, яка постійно розвивається. Оскільки ця мова створена і постійно супроводжується компанією Microsoft, вона постійно оновлюється та розширюється відповідно до сучасних трендів. Можна з впевненістю сказати, що розвиток C# продовжиться і в майбутньому. Ця мова є строго типізованою і добре опановується початківцями. Синтаксис C# досить простий і зрозумілий, і часто використовується для розробки масштабних проєктів. Вона також має широкі можливості, оскільки на неї написано багато бібліотек, які взаємодіють з іншими мовами, а також інтерпретатори багатьох сучасних мов програмування.

Після аналізу найпоширеніших мов програмування для веб-розробки було прийняте рішення розробляти на мові програмування C#.

Фреймворк є важливим інструментом при розробці сайту. Він представляє собою комплекс програмних рішень, який включає дизайн, логіку та базовий

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						37
Зм.	Арк	№ докум.	Підпис	Дата		

функціонал системи або підсистеми. Фреймворк можна розглядати як каркас або набір інструментів для швидкої розробки веб-сайтів та веб-додатків. Існує безліч фреймворків, які допомагають прискорити розробку сайту та правильно організувати його структуру. Після вибору мови програмування кількість доступних фреймворків зменшується, оскільки потрібно обрати той, який сумісний з мовою C#.

Для створення веб-застосунків можна використовувати фреймворки Asp.Net MVC або Asp.Net Core. Ці платформи мають деякі відмінності, і можна сказати, що Asp.Net Core має певні переваги порівняно з Asp.Net MVC.

ASP.Net Core є платформою, за допомогою якої можна розробляти додатки для різних операційних систем, таких як Windows, Linux, macOS та інші. Цей фреймворк забезпечує кросплатформенність.

У порівнянні з Asp.Net MVC, у Asp.Net Core змінено структуру проекту. Файли конфігурації були змінені, а також додано папку з назвою «wwwroot», яка використовується для зберігання статичних файлів, таких як CSS, HTML, зображення, JavaScript і т.д.

Asp.Net Core також забезпечує зручну роботу з базами даних за допомогою Entity Framework. Цей фреймворк має компонентну структуру і реалізує парадигму MVC (Model-View-Controller). Архітектура MVC розподіляє дані, логіку користувача та представлення на три окремі компоненти: модель, представлення та контролер. Це дозволяє незалежно модифікувати кожен компонент.

Asp.Net Core передбачає створення моделей, які представляють дані, відображення для відображення цих даних користувачу та контролери для обробки дій користувача та взаємодії з моделями, а проект буде побудований на основі об'єктно-орієнтованого програмування.

Razor Pages є новим компонентом в ASP.NET Core, який поліпшує продуктивність веб-сторінок на основі програмних сценаріїв. З технічної точки зору, Razor Pages - це модель програмування на основі веб-сторінок, що спрощує створення веб-інтерфейсу. Цей інструмент полегшує взаємодію з

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						38
Зм.	Арк	№ докум.	Підпис	Дата		

представленнями, оскільки дозволяє вставляти код C# в будь-яку частину HTML-сторінки.

Ще однією цікавою особливістю Asp.net Core є інтеграція з jQuery. Це спрощує роботу розробників фронтенду.

Інтерфейс веб-сайту є дуже важливою складовою. Він повинен бути простим, зручним і адаптивним. Для досягнення цього мети рекомендується використовувати такі фреймворки, як CSS і Bootstrap. Перший дозволяє створити привабливий дизайн веб-сайту за допомогою мови JavaScript, що забезпечує кращі візуальні ефекти, такі як анімація, слайдери і т.д.

CSS (Cascading Style Sheets - каскадні таблиці стилів) є однією з основних технологій в сучасному Інтернеті. Рідко можна знайти веб-сайт, який не використовує CSS. CSS-код - це набір інструкцій для браузера, як відображати елементи веб-сторінки. CSS використовується авторами і відвідувачами веб-сторінок для визначення кольорів, шрифтів, макету та інших аспектів вигляду сторінки. Однією з головних переваг є можливість розділити вміст сторінки (зазвичай HTML, XML або подібна мова розмітки) від її вигляду (що описується в CSS).

Bootstrap - це безкоштовний набір інструментів для створення веб-сайтів і веб-додатків. Він містить готові HTML- та CSS-шаблони для форм, кнопок, міток, навігаційних панелей та інших компонентів веб-інтерфейсу. Використання Bootstrap дозволяє зробити сайт адаптивним для різних типів пристроїв.

Отже, після розгляду різних методів розробки було вирішено написати сайт на мові програмування C# з використанням фреймворку Asp.Net Core. Для побудови інтерфейсу веб-сайту буде використано CSS для стильового оформлення, розміщення інформації та задання різних стилів. Для забезпечення адаптивності сайту будуть використані класи Bootstrap.

Для розширення функціональності планується використання JavaScript та бібліотеки jQuery. JavaScript - це динамічна мова програмування, яка дозволяє реалізовувати різноманітні функції на веб-сторінках. Бібліотека jQuery надає

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						39
Зм.	Арк	№ докум.	Підпис	Дата		

потужні інструменти для спрощення роботи з JavaScript, що допомагає взаємодіяти з HTML-документами, керувати подіями, анімацією та іншими елементами веб-сторінки, застосовується до HTML документу, і може забезпечити динамічну інтерактивність на веб-сайтах. Дана мова є досить універсальною і гнучкою.

У додатку JavaScript існує велика кількість додаткових функцій, які можна використовувати. Багато з цих функцій будуть застосовуватись у даному веб-сайті, зокрема програмні інтерфейси додатка (API), які вбудовані у браузері і надають різноманітні функціональні можливості, такі як динамічне створення HTML-елементів і застосування CSS-стилів. Також будуть використовуватись сторонні бібліотеки, що дозволять швидше створення сайту і додадуть більше функціональних можливостей.

Одна з важливих функцій, яку потрібно реалізувати на веб-сайті інтернет-магазину, це можливість натискання кнопок. Однак, за замовчуванням, функціонал мови C# при натисканні кнопки викликає відправку даних на сервер і повне перезавантаження сторінки, що не завжди є оптимальним для проекту.

Однак, для відправки даних на сервер без перезавантаження сторінки існує такий підхід, як AJAX-запит. AJAX (асинхронний JavaScript і XML) - це підхід до створення інтерактивних користувацьких інтерфейсів веб-додатків, при якому відбувається "фоновий" обмін даними між браузером і веб-сервером. Завдяки цьому, при оновленні даних на веб-сторінці не відбувається повне перезавантаження, що зробить сайт швидшим і зручнішим у використанні.

Для кращого розуміння можна порівняти дії, які відбуваються у класичній моделі веб-додатків з діями, що відбуваються при використанні AJAX-запитів.

У класичній моделі:

- користувач заходить на веб-сторінку і натискає на який-небудь її елемент;
- браузер формує і відправляє запит сервера;

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						40
Зм.	Арк	№ докум.	Підпис	Дата		

– у відповідь сервер генерує абсолютно нову веб-сторінку і відправляє її браузеру, після чого браузер повністю перезавантажує всю сторінку.

При використанні AJAX:

- користувач заходить на веб-сторінку і натискає на який-небудь її елемент;
- скрипт (на мові JavaScript) визначає, яка інформація необхідна для оновлення сторінки;
- браузер відправляє відповідний запит на сервер;
- сервер повертає лише ту частину документа, на яку прийшов запит;
- скрипт вносить зміни з урахуванням отриманої інформації (без повного перезавантаження сторінки).

Отже даний інструмент є досить зручним і важливим.

2.6 Висновки до 2-го розділу

Отже, у другому розділі було здійснено проектування всіх основних складових системи, до яких відноситься архітектура, логічна модель бази даних, користувацький інтерфейс.

Було визначено, що даний ресурс буде реалізовуватись за допомогою клієнт-серверної архітектури та патерну MVC. Це допоможе розділити проект на 3 складові: модель, представлення та контролер. Реалізація за допомогою даного патерну дозволяє значно спростити розробку будь-якого програмного забезпечення, а також покращити якість та адекватність кодової частини.

При проектуванні інтерфейсу використовувалось прототипування сторінок ресурсу.

Для реалізації було обрано мову програмування C# із відповідним фреймворком ASP.Net Core.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						41
Зм.	Арк	№ докум.	Підпис	Дата		

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

3.1 Реалізація модулів онлайн-ресурсу

Оскільки дана веб-сторінка не є складним програмним продуктом, і адміністративна та користувацька частини можуть бути об'єднані в одній програмі, було прийнято рішення відмовитись від створення декількох модулів. Замість цього було створено 4 контролери: UserController, AccountController, AdminController, і RoleController.

RoleController використовується для управління ролями в системі. Як було зазначено раніше, ролі використовуються для розмежування дій та обов'язків між різними користувачами. Цей модуль також базується на ASP.NET Core Identity. Саме цей механізм забезпечує доступ до різних контролерів та розділяє адміністративну та користувацьку частини.

AccountController відповідає за реєстрацію та авторизацію користувача в системі. Ці операції здійснюються за допомогою бібліотеки ASP.NET Core Identity. ASP.NET Core Identity є API від Microsoft, яке дозволяє реалізувати логіку управління користувачами в веб-додатку на платформі Core.

Оскільки передбачалося наявність двох основних ролей у системі - звичайного користувача і адміністратора, було розроблено два окремих контролери для користувачів кожної ролі, щоб полегшити функціонування системи в цілому.

Доступ до адміністративної частини є обмеженим. В ній розроблено ряд функцій, які відповідають за створення та редагування товарів, категорій, зображень тощо.

UserController відповідає за користувацьку частину. В ньому реалізовано декілька дій, які доступні для всіх користувачів. Однак, додавання товарів до кошика, перегляд кошика, оформлення замовлень та додавання коментарів можуть виконувати тільки зареєстровані користувачі.

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						42
Зм.	Арк	№ докум.	Підпис	Дата		

3.2 Реалізація бази даних

Для створення та редагування бази даних використовувалися міграції баз даних. Використання міграцій є дуже зручним, оскільки не потрібно створювати кожну таблицю окремо, додавати рядки або створювати зв'язки. Все це виконується автоматично під час міграцій. Достатньо лише створити моделі і визначити логіку зв'язків за допомогою агрегативних полів, а потім додати ці моделі до контексту даних. За допомогою команди для додавання міграцій з консолі автоматично створюється міграція, в якій прописані всі дії, пов'язані з моделями. Ще однією перевагою цього інструменту є можливість оновлення таблиць без втрати даних.

В результаті були створені таблиці, що визначені на етапі проектування бази даних, а саме:

- Таблиця «Користувач»;
- Таблиця «роль»;
- Таблиця «Корзина»;
- Таблиця «Замовлення»;
- Таблиця «Товар»;
- Таблиця «Модель»;
- Таблиця «Категорія»;
- Таблиця «Стать»;
- Таблиця «Колір».

Для забезпечення правильного функціонування етапів, таких як оформлення замовлення та робота з корзиною, було необхідно додати проміжні таблиці. Це було потрібно для коректної реалізації зв'язку «багато до багатьох». Наприклад, в корзині може бути багато товарів, і кожен товар може бути доданим до корзини багатьох користувачів. Тому була створена проміжна таблиця "Складова корзини", в якій вказано, який товар було додано до якої

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						43
Зм.	Арк	№ докум.	Підпис	Дата		

корзини. Тим же принципом була реалізована логіка оформлення замовлення, за допомогою проміжної таблиці «Складова замовлення».

Також була створена таблиця «Розмір» з тією самою метою, оскільки кожен колір товару може мати безліч розмірів. Створення таких таблиць в базі даних дозволяє зручно працювати з нею. Завдяки цьому можна легко виводити дані, здійснювати пошук та фільтрацію. Діаграму створеної бази даних можна переглянути у графічному додатку.

3.3 Реалізація онлайн-ресурсу

Першим етапом при створенні сайту за архітектурою MVC є створення основних моделей та встановлення залежностей між ними. Основними моделями для цього сайту є Товар, Замовлення та Корзина. Однак, зрозуміло, що цих моделей недостатньо. Для забезпечення зручного пошуку та фільтрації товарів були додані додаткові моделі, такі як Стать, Категорія та Модель товару.

Наступним кроком є створення контролерів та представлень. Щоб мати можливість відображати результати, наприклад, пошуку чи фільтрації товарів, необхідно заповнити сторінку відповідним контентом. Усі дії, пов'язані з додаванням, редагуванням та видаленням товарів, категорій та моделей, були реалізовані в контролері адміністратора. Ці дії є досить простими, але при додаванні товару необхідно реалізувати зв'язані списки для вибору категорії та моделі. Для реалізації такого списку зручно використовувати часткове представлення. Цей механізм працює за таким принципом: обирається категорія, і в залежності від цього формується наступний список, який містить тільки ті компоненти, що відповідають обраній категорії. Часткове представлення містить лише цей список моделей.

Для зміни списків при виборі було використано аjax-запит.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						44
Зм.	Арк	№ докум.	Підпис	Дата		

```

$('#category').change(function () {
    var id = $(this).val();

    $.ajax({
        type: 'GET',
        url: 'GetModells/' + id,
        success: function (data) {

            $('#modells').replaceWith(data);
        }
    });
});

```

Цей запит передає номер категорії, який обрано у методі контролера, що відповідає за отримання даних з бази даних для цієї категорії. Потім він передає часткове представлення цих даних у відповідь на аґах-запит, який замінює частину сторінки, де розташований випадаючий список, цим частковим представленням (результат на рисунку 3.1).

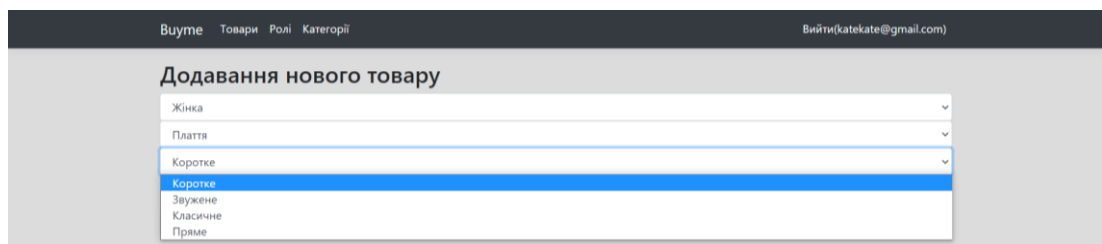


Рисунок 3.1 – Робота зв'язного списку

Після того, як здійснено введення всіх даних про товар, вони надсилаються через Post-запит до контролера. У контролері ці дані додаються до бази даних.

```

[HttpPost]
public async Task<IActionResult> Addtovar(Tovar user)
{
    if (ModelState.IsValid)
    {
        db.Tovars.Add(user);
        await db.SaveChangesAsync();
        return RedirectToAction("Tovarinf", new { id = user.Id });
    }
    else
        return View();
}

```

Редагування та видалення даних з бази даних також здійснюються за допомогою методів у контролері.

```
[HttpPost]
public async Task<IActionResult> Tovarinf(Tovar user)
{
    db.Tovars.Update(user);
    await db.SaveChangesAsync();
    return RedirectToAction("Index");
}
[HttpPost]
public async Task<IActionResult> Delete(int? id)
{
    if (id != null)
    {
        Tovar user = await db.Tovars.FirstOrDefaultAsync(p => p.Id == id);
        if (user != null)
        {
            db.Tovars.Remove(user);
            await db.SaveChangesAsync();
            return RedirectToAction("Index");
        }
        return NotFound();
    }
}
```

Щодо реалізації функціоналу для користувача, процес додавання замовлень або товарів до корзини відбувається за однаковим принципом. Користувачам зареєстрованим доступний перегляд галереї зображень при перегляді товару. Для реалізації цієї функціональності використано jquery slider бібліотеку.

```
$(".slider").slick({
    autoplay: false,
    dots: true,
    arrows: false,
    customPaging: function (slider, i) {
        var thumb = $(slider.$slides[i]).data('thumb');
        return '<a></a>';
    },
});
```

На головній сторінці, де відображається список всіх товарів, було реалізовано можливість фільтрування даних за статтю, категорією товарів та

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		46

моделями. Щоб уникнути перезавантаження сторінки при кожному фільтруванні, було використано аїах-запит.

```
$('.sex').bind('click', function () {  
    |   var id = $(this).attr('id');  
  
    $.ajax({  
        type: 'GET',  
        url: '@Url.Action("GetTovars")/' + id,  
        success: function (data) {  
            $('#mainn').replaceWith(data);  
        }  
    });  
});
```

При натисканні на статтю, створюється Post-запит з номером статті і відправляється до контролера. Потім контролер генерує відповідь у форматі часткового представлення. Цю відповідь приймає аїах-запит і замінює весь вміст сторінки на часткове представлення.

Код методу контролера подано нижче:

```
public ActionResult GetTovars(int id)  
{  
    List<Category> categories = db.Categories.Include(p => p.Modells).ThenInclude(c =>  
c.Tovars).ThenInclude(u => u.Kinds).ThenInclude(o => o.Images).Where(c => c.SexId ==  
id).ToList();  
    List<Tovar> tovars = new List<Tovar>();  
    foreach(var category in categories)  
    {  
        foreach(var modell in category.Modells)  
        {  
            foreach (var tovar in modell.Tovars)  
            {  
                tovars.Add(tovar);  
            }  
        }  
    }  
  
    return PartialView(tovars);  
}
```

Ті самі операції були виконані для фільтрації за категорією та моделлю товару. На рисунку 3.2 показано приклад використання фільтрації за жіночим розміром та категорією «Сукня».

При додаванні нових категорій вони автоматично додаються до списку доступних, так само як і моделі. Крім того, є можливість здійснювати пошук

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						47
Зм.	Арк	№ докум.	Підпис	Дата		

товарів. Для цього у верхній частині є спеціальний розділ, призначений для пошуку. У разі відсутності знайдених даних виводиться повідомлення про це.

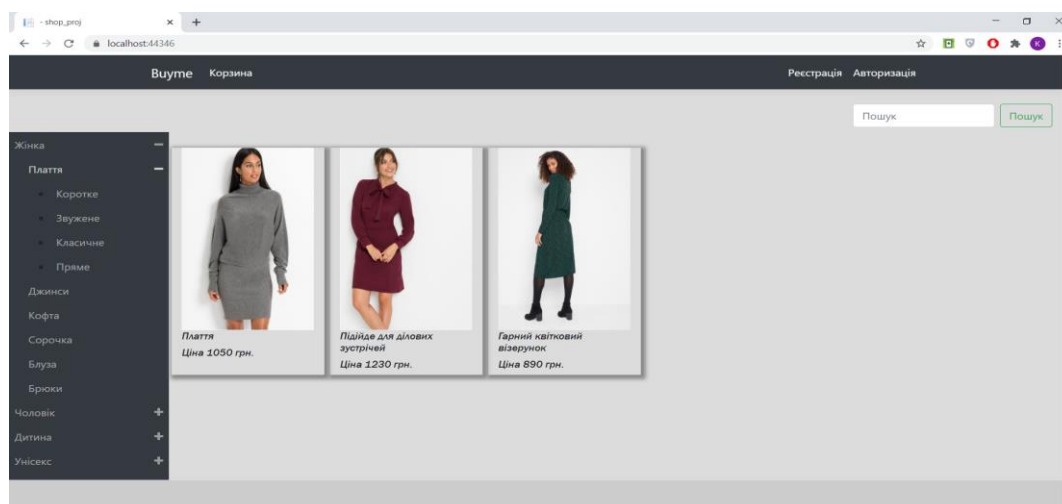


Рисунок 3.2 – Фільтрація даних

На рисунку 3.4 відображаються тільки ті дані, що обрав користувач, вивід цих даних виконується без перезапуснення сторінки.

3.4 Інструкція з експлуатації

Неавторизованому користувачу, при вході на сайт, доступні обмежені функції. Він може переглядати наявні товари, виконувати пошук серед них та використовувати фільтри для відображення даних за категорією, статтю та моделлю (рисунок 3.3).

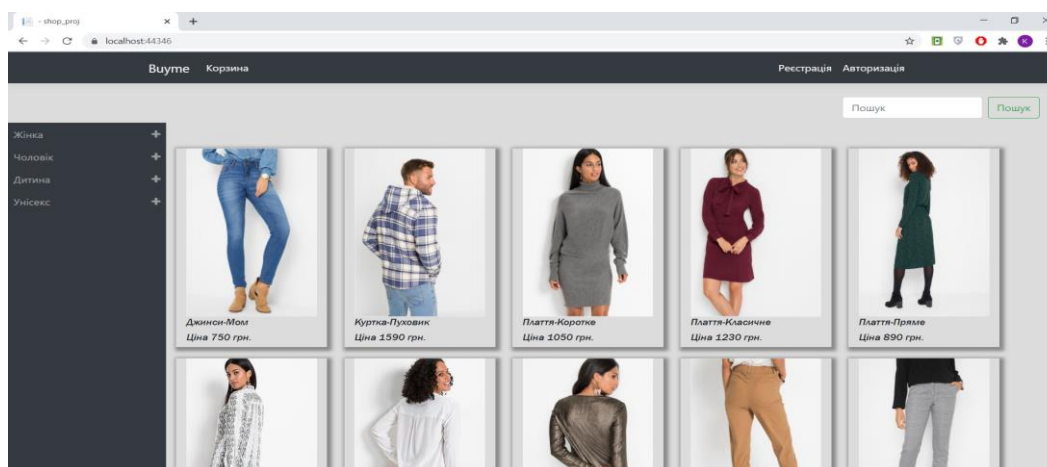


Рисунок 3.3 – Основна сторінка сайту

Для перегляду кожного товару окремо також доступна спеціальна сторінка, на якій можна побачити кілька зображень цього товару, відмінних від початкової сторінки (див. рисунок 3.4). Крім того, на цій сторінці відображається інформація про бренд, матеріал, а також короткий опис товару. Внизу сторінки можна знайти коментарі інших авторизованих користувачів про цей товар.

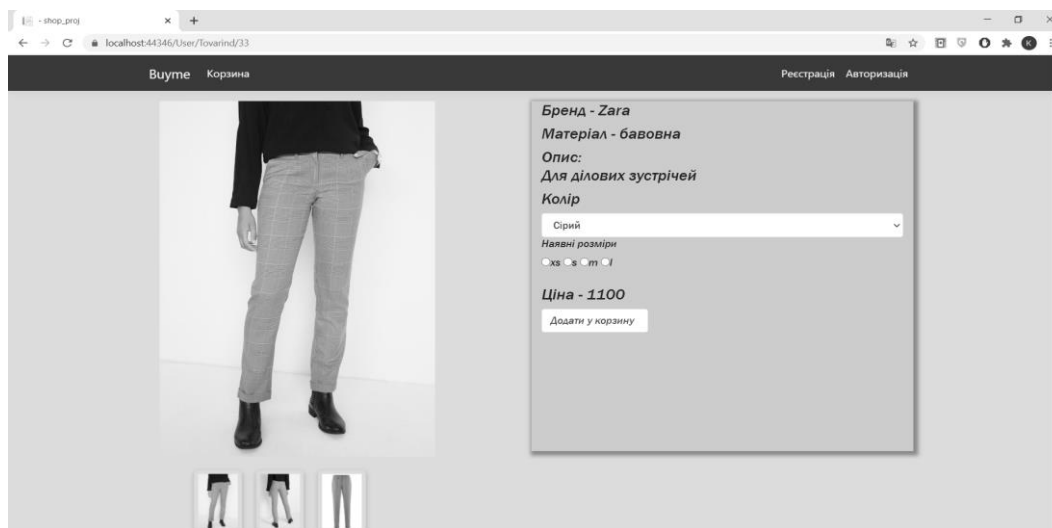


Рисунок 3.4 – Сторінка одного товару

Незареєстрований користувач має доступ тільки до цих дій. Для продовження роботи із ресурсом необхідно авторизуватися або зареєструватися (рисунок 3.5).

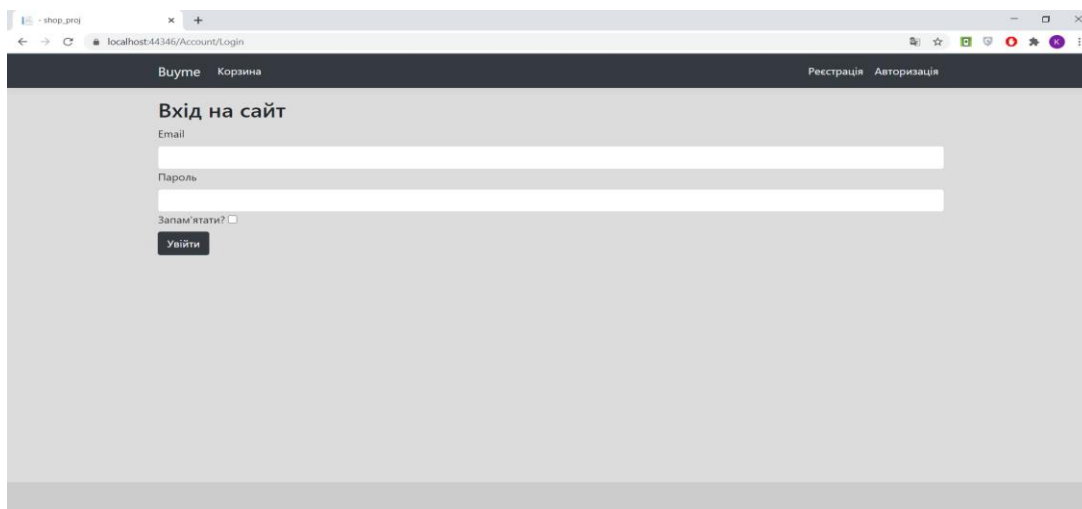


Рисунок 3.5 – Сторінка авторизації

Роль користувача в системі визначається його іменем під час входу. Залежно від цієї ролі, він може виконувати дії як адміністратор або звичайний користувач.

Авторизований користувач має доступ до таких дій:

- Додавання товару до корзини.
- Перегляд вмісту корзини.
- Оформлення замовлення.
- Перегляд історії замовлень.
- Написання коментарів до товарів.

Для додавання товару в корзину користувач повинен перейти на сторінку цього товару. Після вибору необхідного розміру товару, його можна додати до корзини, натиснувши відповідну кнопку (рисунок 3.6).

Бренд - H&M
Матеріал - бавовна
Опис:
Підійде для ділових зустрічей
Колір
Червоний
Наявні розміри
 m xl
Ціна - 1230
Додати у корзину

Рисунок 3.6 -Додавання товару в корзину

Після додавання товару до корзини, він буде збережений у корзині. Для переходу до корзини необхідно вибрати відповідний пункт у головному меню. На сторінці корзини будуть відображені всі товари, які були додані до корзини і ще не були замовлені (рисунок 3.7).

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						50
Зм.	Арк	№ докум.	Підпис	Дата		

Рисунок 3.7 – Корзина користувача

Для оформлення замовлення необхідно спочатку вибрати товари, які бажаєте замовити, та натиснути кнопку "Обрати". Після цього з'явиться доступна панель оформлення замовлення. Вам потрібно буде заповнити всі обов'язкові поля і надати платіжні дані (рисунок 3.8).

Рисунок 3.8 – Оформлення замовлення

Після надсилання платіжних даних користувач отримає чек, в якому вказано приблизний час доставки (рисунок 3.9).

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		51

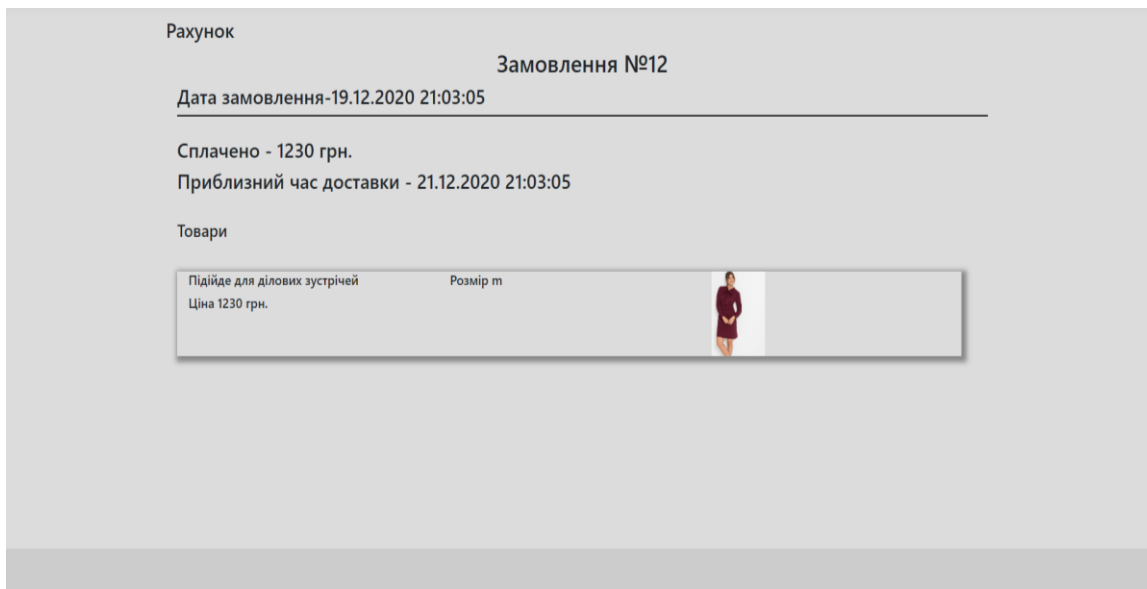


Рисунок 3.9 – Чек на замовлення

Також є можливість у користувача залишати коментарі на сторінці товару (рисунок 3.10).

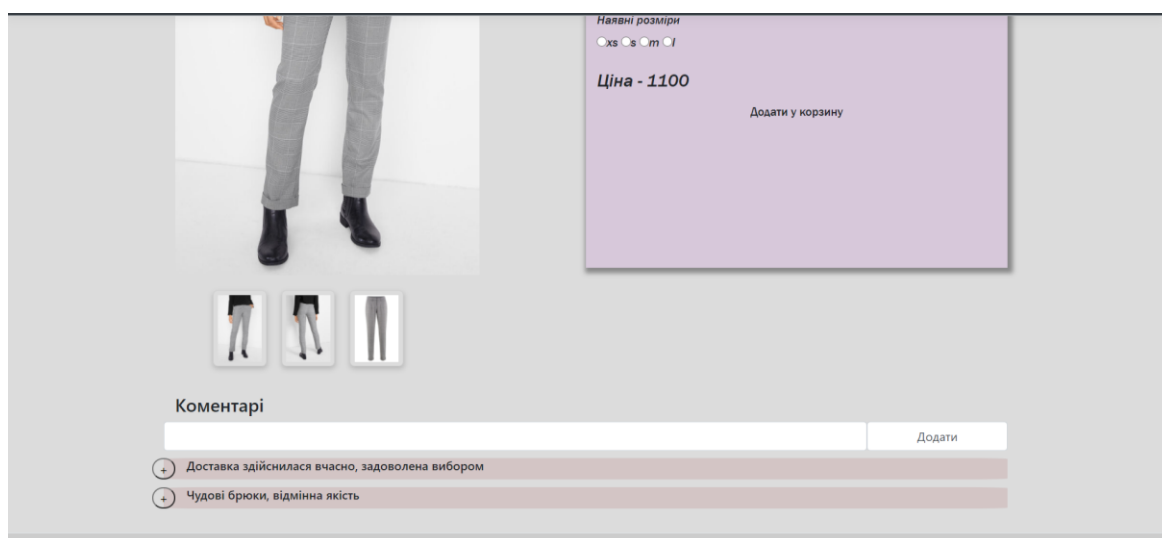


Рисунок 3.20 – Додавання коментарів на сторінці товару

Після закінчення роботи із ресурсом може вийти зі сторінки за допомогою пункту головного меню.

Для роботи адміністратора також передбачений певний алгоритм дій, після якого він може здійснювати роботу із даним ресурсом.

Після авторизації адміністратор може здійснювати додавання, редагування, видалення товарів, а також керувати всіма даними, що пов'язані з цим – додавання нових категорій, моделей. Також йому доступний список товарів (рисунок 3.11).

Всі товари

№	Категорія	Модель	Бренд	Матеріал	Ціна		
24	Джинси	Мом	Zara	Джинс	750	Редагувати	Видалити
25	Куртка	Пуховик	Zara	бавовна, поліестр	1590	Редагувати	Видалити
26	Плаття	Коротке	Zara	бавовна	1050	Редагувати	Видалити
27	Плаття	Класичне	H&M	бавовна	1230	Редагувати	Видалити
28	Плаття	Пряме	H&M	Трикотаж	890	Редагувати	Видалити
29	Сорочка	Довга	H&M	бавовна	680	Редагувати	Видалити
30	Сорочка	Довга	Zara	Віскоза	790	Редагувати	Видалити
31	Блуза	Рукави	Colin's	Поліамід	930	Редагувати	Видалити
32	Брюки	Короткі	H&M	бавовна	1090	Редагувати	Видалити
33	Брюки	Класичні	Zara	бавовна	1100	Редагувати	Видалити
34	Брюки	Класичні	Colin's	бавовна	900	Редагувати	Видалити

[Додати](#)

Рисунок 3.31 – Перелік всіх товарів

Для того, щоб додати інформацію про новий товар, потрібно обрати спершу статть, далі категорію та модель, після цього здійснюється введення всіх інших необхідних даних (рисунок 3.12).

Додавання нового товару

Жінка

Плаття

Коротке

Бренд

Матеріал

Ціна

Опис

[Додати](#)

Рисунок 3.42 – Додавання товару

Після додавання товару, буде здійснено автоматичний перехід користувача на сторінку редагування. На даній сторінці потрібно додати кольори товарів (рисунок 3.13).

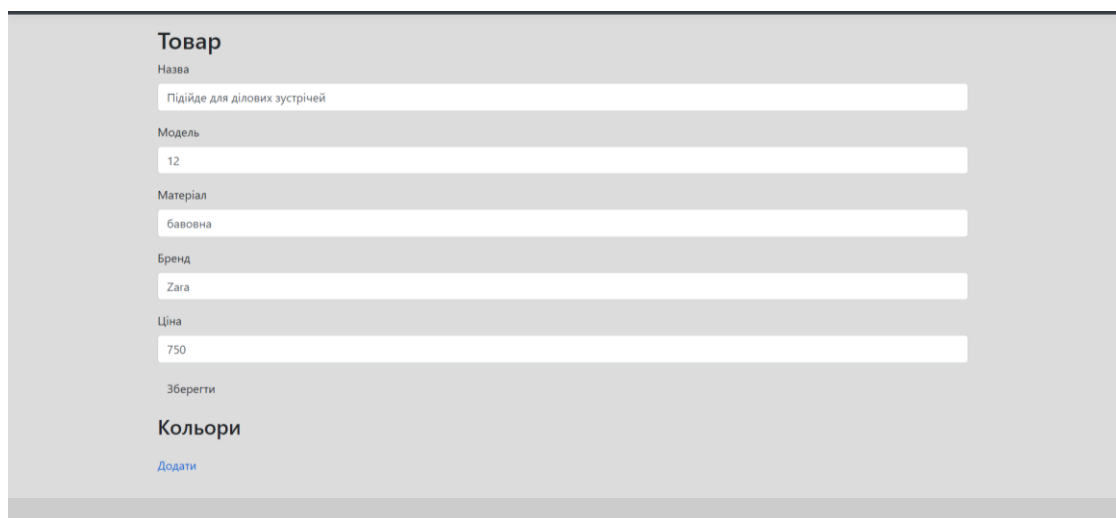


Рисунок 3.53 – Редагування товару

Для того, щоб здійснити додавання кольору необхідно також вказати кількість розмірів, що є в наявності, а також зображення товару. Даних зображень може бути декілька (рисунок 3.14).

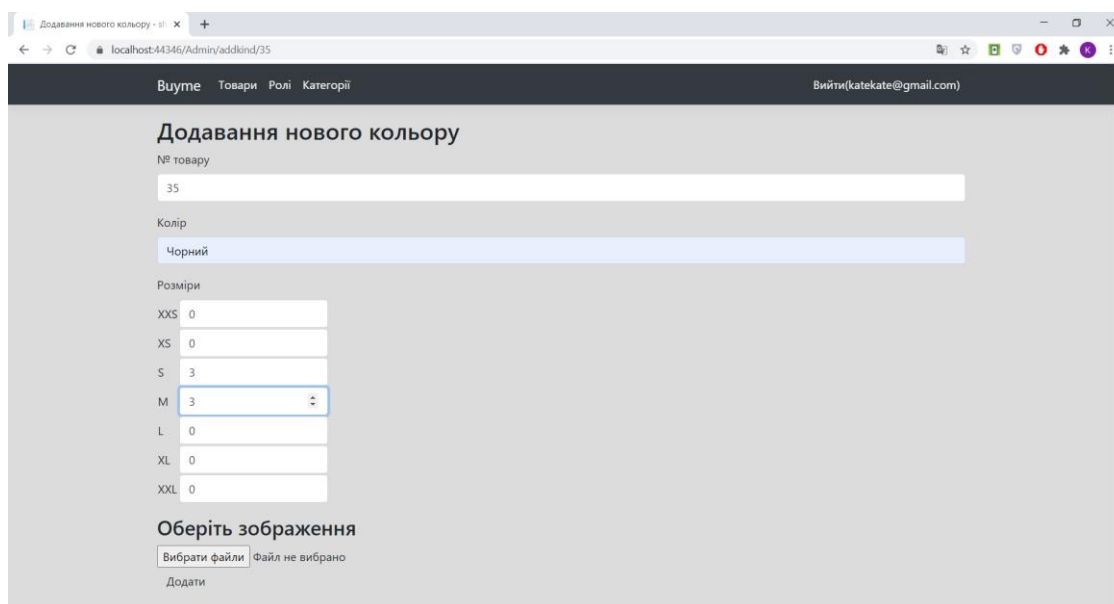


Рисунок 3.64 - Додавання нового кольору товару

Адміністратор також має доступ до переліку категорій, які також можна редагувати та видаляти (рисунок 3.15).

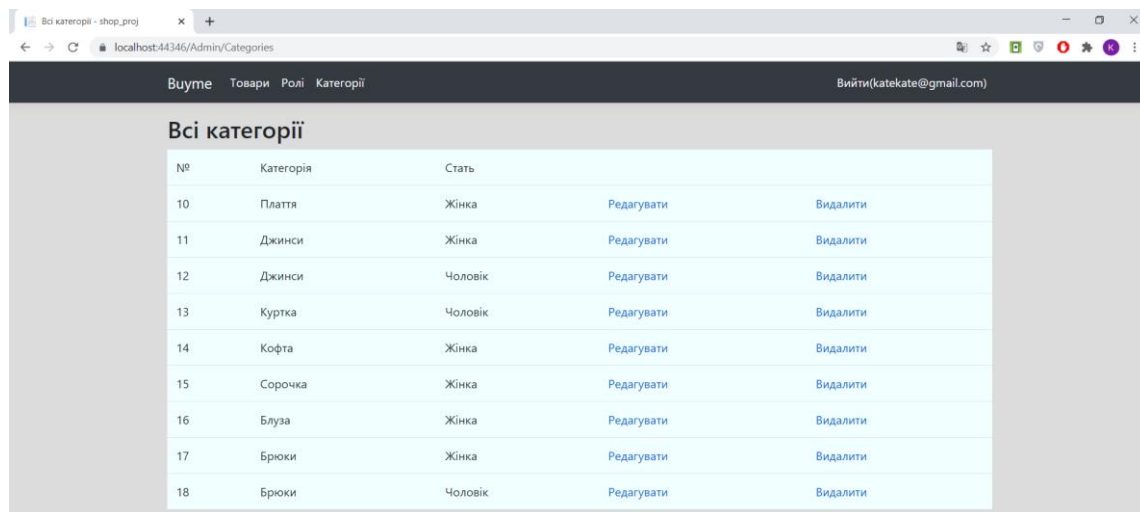


Рисунок 3.75 – Перелік категорій

Адміністратор також може розпоряджатись ролями в системі, шляхом додавання нових ролей. Також може видаляти, і присвоювати ці ролі користувачам (рисунок 3.16).

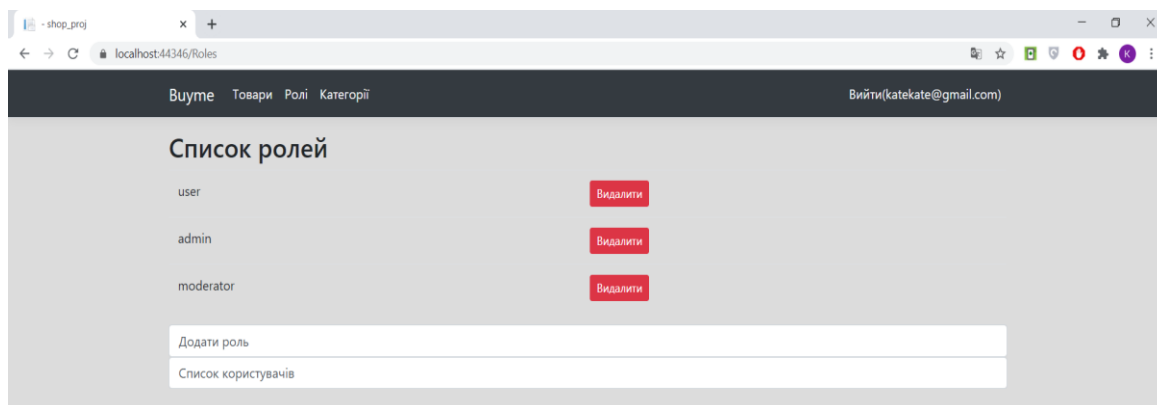


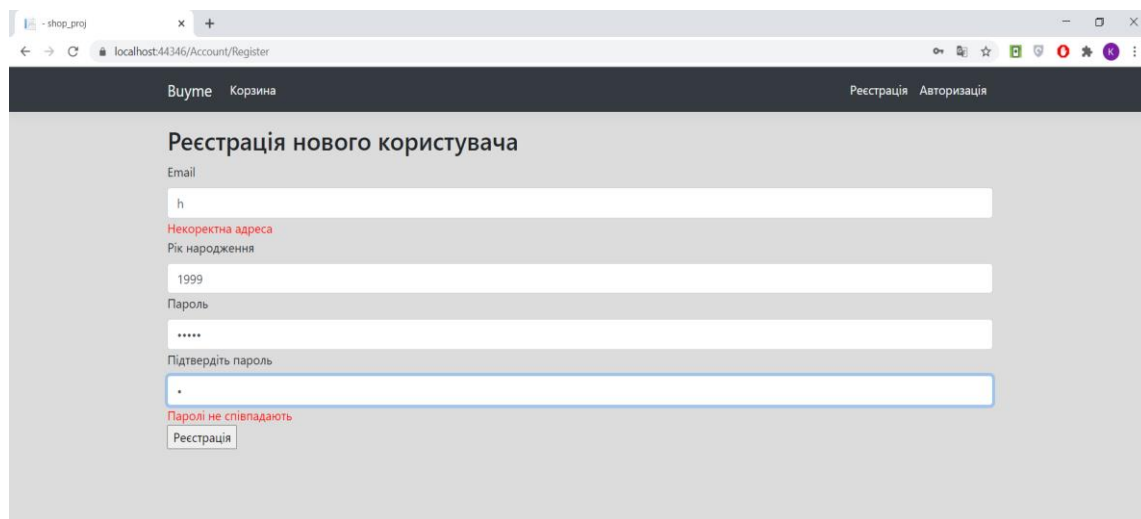
Рисунок 3.86 – Керування ролями

3.5 Тестування онлайн-ресурсу

Використання ручного тестування, як зазначалось раніше, робить неможливим відправлення некоректних даних на сервер. Завдяки цьому

виключається можливість отримання помилки при роботі з базою даних та самою програмою.

При реєстрації обов'язковим моментом є валідація. На рисунку 3.18 показано приклад, коли введено два однакових паролі, тобто повторне підтвердження паролю. Введені паролі мають бути однаковими. Таким же чином здійснюється перевірка адреси на коректність (рисунок 3.17).




The screenshot shows a web browser window with the URL localhost:44346/Account/Register. The page title is "Реєстрація нового користувача". The form contains the following fields and messages:

- Email: "h" (Error: Некоректна адреса)
- Рік народження: "1999"
- Пароль: "*****"
- Підтвердіть пароль: "." (Error: Паролі не співпадають)

Buttons: "Реєстрація"

Рисунок 3.18 – Валідація даних при реєстрації

Під час здійснення авторизації користувач проходить аутентифікацію, тобто під час ведення логіну, пароль має співпадати із тим, що вже записаний у базі даних. Тому валідація в даному випадку також є дуже важливим аспектом (рисунок 3.19).



The screenshot shows a web browser window with the URL localhost:44346/Account/Login. The page title is "Вхід на сайт". The form contains the following fields and messages:

- Email: "Olya12345@gmail.com"
- Пароль: (empty)
- Запам'ятати?:

Buttons: "Увійти"

Error message: "Невірний логін чи пароль"

Рисунок 3.19 – Валідація даних при авторизації

Під час здійснення додавання інформації про товар адміністратор також повинен заповнити всі поля, щоб надати покупцям повну інформацію (рисунок 3.20).

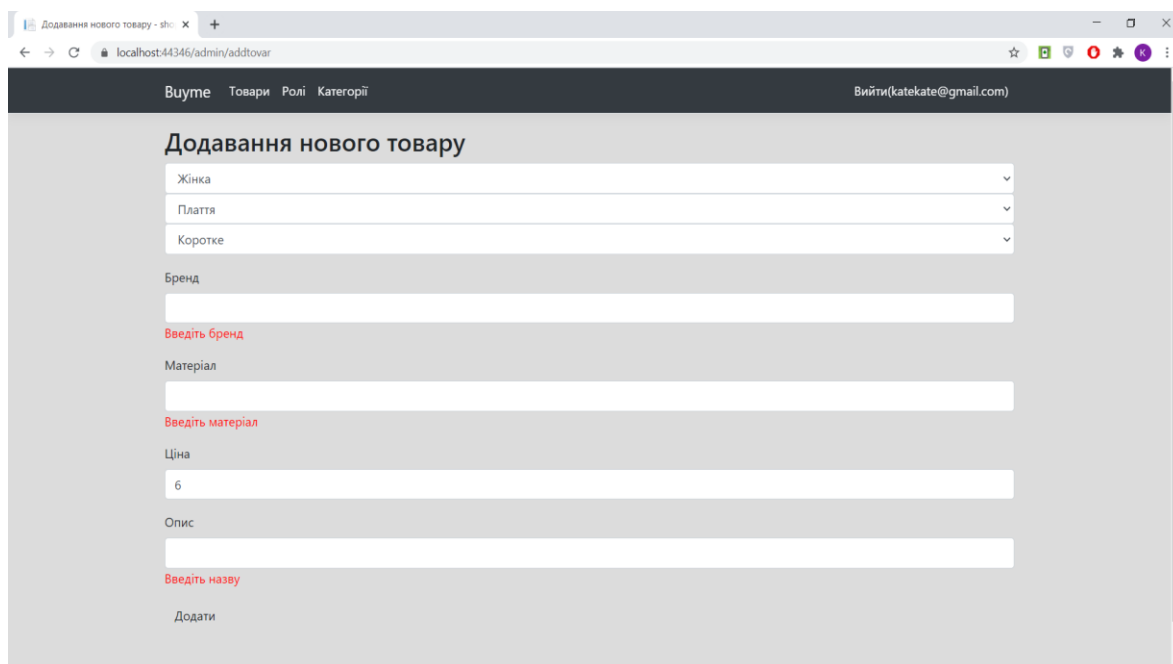


Рисунок 3.20 – Валідація даних при додавання нового товару

Для того, щоб здійснити перевірку модульним тестуванням, було написано додаткову програму, що містить тестові функції, які перевіряють методи контролера основної програми. Декілька модульних тестів із цієї програми подано нижче:

```
[Fact]
public void OrderAdd()
{
    var userManager = new FakeUserManager();
    var optionsBuilder = new
DbContextOptionsBuilder<shop_proj.Models.AppContext>().UseSqlServer("Server=(localdb)\MSS
QIlocalDB;Database=dbshop;Trusted_Connection=True;");

    var signInManager = new FakeSignInManager();
    using (var db = new shop_proj.Models.AppContext(optionsBuilder.Options))
    {
        using (var controller = new UserController(db, userManager))
        {
            var result = controller.GetTovarsMod(3);
            var redirectToActionResult = Assert.IsType<PartialViewResult>(result);
            Assert.Equal("GetTovars", redirectToActionResult.ViewName);
        }
    }
}
[Fact]
public void ViewTovar()
{
    var userManager = new FakeUserManager();
    var optionsBuilder = new
DbContextOptionsBuilder<shop_proj.Models.AppContext>().UseSqlServer("Server=(localdb)\MSS
QIlocalDB;Database=dbshop;Trusted_Connection=True;");
```

```

var signinManager = new FakeSignInManager();
using (var db = new shop_proj.Models.AppContext(optionsBuilder.Options))
{
    using (var controller = new UserController(db, userManager))
    {
        var result = controller.Tovarind(28);
        Assert.NotNull(result);
    }
}
}
}

```

Дані тести перевіряють основні методи контролера для користувача. Результати модульного тестування на рисунку подано на рисунку 3.21.

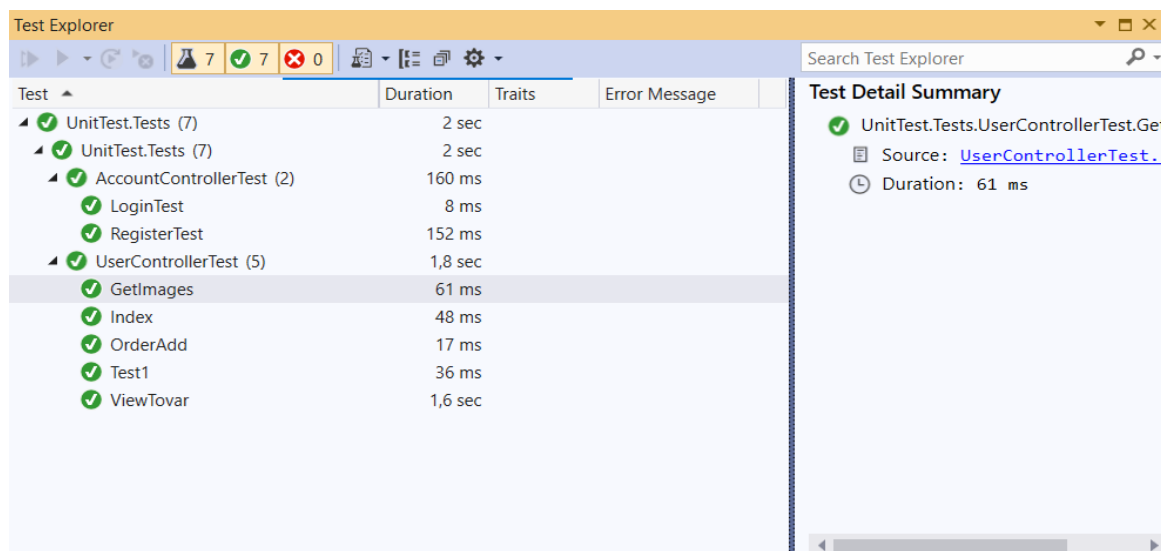


Рисунок 3.21 – Результати модульного тестування

3.6 Висновки до 3-го розділу

Отже, в результаті написання третього розділу було здійснено визначення загальної структури основних програмних модулів, створено моделі із наступною реалізацією баз даних із встановленими зв'язками між її таблицями. Після цього було запрограмовано всю логіку додатку MVC, включаючи 4 контролери та представлення для їх методів.

Також було розроблено інструкції для користувачів системи: адміністратора та покупця. Результатом етапу є програмна реалізація онлайн-ресурсу для роздрібної торгівлі одягом.

ВИСНОВКИ

Отже, за результатами написання даної кваліфікаційної роботи бакалавра можна зробити такі висновки.

У першому розділі подано детальний аналіз предметної області із визначенням її меж, а також виокремлення функціональних та нефункціональних вимог, проаналізовано існуючі онлайн-ресурси з продажу одягу онлайн, розроблено технічне завдання.

При порівнянні різних інтернет-магазинів одягу спрямовується увага на такі фактори: безпосередня робота ресурсу, зручний інтерфейс користувача, можливість оплати та доставки, наявність адміністративної панелі, можливість авторизації та реєстрації користувачів.

Актуальність та перспективи розвитку онлайн ресурсу для торгівлі одягом в роздріб є високими і обіцяють значний потенціал для успішного бізнесу. Ось декілька ключових факторів, що підтверджують цю актуальність та перспективи:

Зростання популярності онлайн-шопінгу: за останні роки спостерігається значний приріст уподобань споживачів до зручності та доступності онлайн шопінгу. Купівля одягу в Інтернеті стала звичайною практикою для багатьох людей, що створює великий ринок для онлайн роздрібних магазинів.

Широко охоплений ринок: онлайн-ресурси дозволяють досягти міжнародної аудиторії і працювати з клієнтами з різних країн. Це дозволяє розширити свій бізнес та збільшити обсяг продажів.

Зручність та персоналізація: клієнти можуть зручно шукати, переглядати та купувати товари в будь-який зручний для них час. Онлайн ресурси також надають можливість персоналізації, рекомендацій та підбору товарів згідно з індивідуальними потребами та вподобаннями клієнтів.

Можливості маркетингу та реклами: онлайн-ресурси надають широкий спектр можливостей для маркетингу та реклами продуктів. Це включає в себе

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		59

використання соціальних медіа, цільову рекламу, електронні розсилки та інші ефективні інструменти просування бренду та залучення клієнтів.

Інновації та технологічний прогрес: з постійним розвитком технологій, таких як розширена реальність (AR), віртуальна реальність (VR) та штучний інтелект (AI), відкриваються нові можливості для поліпшення користувацького досвіду, візуалізації продуктів та персоналізації покупок.

У другому розділі було здійснено проектування всіх основних складових системи, до яких відноситься архітектура, логічна модель бази даних, користувацький інтерфейс.

Було визначено, що даний ресурс буде реалізовуватись за допомогою клієнт-серверної архітектури та патерну MVC. Це допоможе розділити проект на 3 складові: модель, представлення та контролер. Реалізація за допомогою даного патерну дозволяє значно спростити розробку будь-якого програмного забезпечення, а також покращити якість та адекватність кодової частини.

При проектуванні інтерфейсу використовувалось прототипування сторінок ресурсу.

Для реалізації було обрано мову програмування C# із відповідним фреймворком ASP.Net Core.

В результаті написання третього розділу було здійснено визначення загальної структури основних програмних модулів, створено моделі із наступною реалізацією баз даних із встановленими зв'язками між її таблицями. Після цього було запрограмовано всю логіку додатку MVC, включаючи 4 контролери та представлення для їх методів.

Також було розроблено інструкції для користувачів системи: адміністратора та покупця. Результатом етапу є програмна реалізація онлайн-ресурсу для роздрібної торгівлі одягом.

Загалом, розвиток онлайн ресурсу для торгівлі одягом в роздріб має великий потенціал. Забезпечення високоякісного користувацького досвіду, широкий асортимент товарів, персоналізація та ефективні маркетингові стратегії є ключовими факторами успіху в цій сфері.

					КвРІПЗ.200139.01.14.ПЗ	Арк.
						60
Зм.	Арк	№ докум.	Підпис	Дата		

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Все про Е-шопінг. URL: <https://elnews.com.ua/uk/moda-onlajn-61-vsih-internet-prodazhiv-u-sviti-prypadaye-na-odyag-i-vzuttya/> (дата звернення: 15.02.2023).
2. Ivanka Menken Cloud Computing – The Complete Cornerstone Guide to Cloud Computing Best Practices: Concepts, Terms, and Techniques for Successfully Planning, Implementing and Managing Enterprise IT Cloud Computing Technology, Australia, 2008. – 203 с.
3. Jon Duckett Beginning HTML, XHTML, CSS and JavaScript. Wiley Publishing, Canada, 2020. – 834 с.
4. Rajkumar Buyya, James Broberg, Andrzej Goscinski Cloud computing Principles and Paradigms. John Wiley & Sons, Inc., New Jersey, Canada, 2021. – 637 с.
5. Ronald L. Krutz, Russell Dean Vines Cloud Security Comprehensive Guide to Secure Cloud Computing, Wiley Publishing, Inc., USA, Canada, 2020. – 358 с.
6. 20 найдорожчих інтернет-магазинів світу. Plerdy. URL: <https://www.plerdy.com/ua/blog/22-most-expensive-ecommerce-website/> (дата звернення: 25.02.2023).
7. Плахтій П. Д., Мендерецький В. В, Болібрух Б. В., Юхименко А. П., А. М. Савчук. Інформаційні системи та технології для роботи з ними: Навчальний посібник / Під ред. П.Д. Плахтія. – Кам'янець-Подільський: видавець М.І. Мошак (агенство «ІНФО»), 2013. – 374 с.
8. Jon Duckett Beginning HTML, XHTML, CSS and JavaScript. Wiley Publishing, Canada, 2020. – 834 с.
9. Агатова О. М. Автоматизовані системи документів: Виникнення та тенденції розвитку / Пашков О.М. // Бібл. планета. – К.: 2016. – №2. – С.33-34.
10. Michael Miller Cloud Computing Web-based Applications That Change the Way You Work and Collaborate Online. Que Publishing, USA, 2009. – 284 с.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		61

11. [Електронний ресурс] / Art-lemon. URL: <https://art-lemon.com/site-test> (дата звернення: 15.02.2023).
12. Модульне тестування. QaLight. URL: <https://qalight.ua/baza-znaniy/modulne-testuvannya/> (дата звернення: 15.02.2023).
13. Захарова О. В., Захарова Е. Г., Резниченко В. А. Каталог наукових електронних бібліотек в Інтернет // ІПС НАН України. – К., 2012. – 76 с.
14. Інтелектуальні інформаційні технології / О. Станкевич, С. Дубик // Зап. Львів. наук. б-ки ім. В. Стефаника. – 2018. – Вип. 13. – С. 519-522.
15. Волохін О. М. Каталогізація цифрових ресурсів Інтернет: Дублінське ядро метаданих: посібник / О.М. Волохін. – Кіровоград, 2013. – 70 с.
16. Розников В. А., Захарова О. В., Захарова Е. Г. Інформаційні системи та сервіси // Проблеми програмування. – 2017. – № 4 – С. 60-72.
17. Як тестувати веб-сайт: основні етапи і поради / Brainlab. URL: <https://brainlab.com.ua/uk/blog-uk/yak-testuvati-veb-sayt-osnovn-etapi-poradi> (дата звернення: 25.03.2022).
18. Web Application Testing: 8 Step Guide to Website Testing. Guru99. URL: <https://www.guru99.com/web-application-testing.html> (дата звернення: 15.04.2022).
19. How to test a website. Geteasyqa. URL: <https://geteasyqa.com/qa/test-website/> (дата звернення: 17.04.2023).
20. Винокуров В. Д., Макарчук О. М., Патланжоглу М. О., «Практичний курс інформаційних систем та технологій. Створення баз даних» за редакцією академіка АПН України Олійника А.М., Київ, 2014. – 348 с.
21. Сучасні інформаційні технології і популярні системи / О. Григоровська, С. Панасовович // Зап. Чернівці. наук. б-ки ім. В. Стефаника. – 2014. – Вип. 13. – С. 519–522.
22. Теорія та практика побудови баз даних. 8-е вид. Д. Кренке. – Вид.: Суми, 2017. – 800 с.

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						62
Зм.	Арк	№ докум.	Підпис	Дата		

23. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. – Уведено вперше; чинний від 2016–07–01. – Київ : ДП «УкрНДНЦ», 2016. – 17 с.

24. Уніфікована мова програмування UML. Портал знань. URL: <http://www.znannya.org/?view=uml>. (дата звернення: 15.04.2022).

25. Unit Testing Tutorial: What is, Types, Tools & Test EXAMPLE. [URL: <https://www.guru99.com/unit-testing-guide.html> (<https://www.guru99/>) (дата звернення: 16.02.2022).

26. NOSQL – переваги та недоліки нереляційних баз даних. QAinfo. URL: <https://www.quality-assurance-group.com/nosql-perevagy-ta-nedoliky-nerelyatsijnyh-baz-danyh/> (дата звернення: 15.02.2023).

27. SWB. URL: <https://superbwebsitebuilders.com/ru/kak-sozdat-svoj-prostoj-html-sajt-v-bloknote/> (дата звернення: 24.04.2023).

28. Adobe. URL: <https://helpx.adobe.com/ua/illustrator/using/design-website-layout.html> (дата звернення: 27.04.2023).

29. Дейл Р., МакДональд М. Професійний ASP.NET 4.5 в С# і VB. – Київ: Видавництво "Діалектика", 2013. – 1536 с.

30. Фріман А., Робінсон А. ASP.NET MVC 5: Розробка веб-додатків на С# з використанням Entity Framework, Bootstrap і Azure. – Київ: Видавництво "Діалектика", 2016. – 832 с.

31. Фріман А., Робінсон А. Pro ASP.NET MVC 5. – Berkeley, CA: Apress, 2013. – 832 с.

32. Адам Фріман. ASP.NET Core MVC: розробка Web-додатків на С#. – Київ: Видавництво "Діалектика", 2020. – 848 с.

33. Лоуренс Дейл. Розробка веб-додатків на ASP.NET Core MVC. – Київ: Видавництво "Діалектика", 2019. – 736 с.

34. Троелсен А. Програмування на С# для професіоналів. – Київ: Видавництво "Діалектика", 2013. – 1504 с.

35. Девідсон Дж. Розробка ASP.NET-додатків на С# і VB.NET для професіоналів. – Київ: Видавництво "Діалектика", 2019. – 928 с.

					КвРІПЗ.200139.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		63

36. Mozilla Developer Network (MDN) - офіційна документація від Mozilla про веб-стандарти, HTML, CSS, JavaScript та багато іншого. [Електронний ресурс] // URL: <https://developer.mozilla.org/> (дата звернення: 15.02.2023).

37. W3Schools - веб-сайт, який надає практичні приклади, посібники та документацію по HTML, CSS, JavaScript та багатьом іншим веб-технологіям. [Електронний ресурс] // URL: <https://www.w3schools.com/> (дата звернення: 15.02.2023).

38. Stack Overflow - популярний форум програмістів, де ви можете знайти відповіді на ваші питання про веб-технології. [Електронний ресурс] // URL: <https://stackoverflow.com/> (дата звернення: 15.02.2023).

39. CSS-Tricks - ресурс, який пропонує статті, посібники та демонстрації з CSS, HTML та JavaScript. [Електронний ресурс] // URL: <https://css-tricks.com/> (дата звернення: 15.02.2023).

40. Smashing Magazine - онлайн-журнал для веб-розробників, який містить статті та ресурси по дизайну, розробці та веб-технологіях. [Електронний ресурс] // URL: <https://www.smashingmagazine.com/> (дата звернення: 15.02.2023).

					КвРІПЗ. 200139.01.14.ПЗ	Арк.
						64
Зм.	Арк	№ докум.	Підпис	Дата		

ДОДАТОК А
(обов'язковий)

КОД ОСНОВНИХ МОДУЛІВ (ЛІСТИНГ)

```

using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.Linq;
using System.Threading.Tasks;

namespace shop_proj.Models
{
    public class Tovar
    {
        public int Id { get; set; }

        [Required(ErrorMessage = "Введіть назву")]
        [StringLength(300, MinimumLength = 3, ErrorMessage = "Довжина рядка повинна бути від
3 до 300 символів")]
        public string Name { get; set; }
        public int ModellId { get; set; }
        public Modell Modell { get; set; }
        [Required(ErrorMessage = "Введіть бренд")]
        public string Brend { get; set; }
        [Required(ErrorMessage = "Введіть матеріал")]
        public string Material { get; set; }

        [Required(ErrorMessage = "Введіть ціну")]
        public double Price { get; set; }
        public List<Kind> Kinds { get; set; }
        public List<Komentar> Kommentars { get; set; }

    }
}

using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace shop_proj.Models
{
    public class Order
    {
        public int Id { get; set; }
        public User User { get; set; }
        public string UserId { get; set; }
        public List<Orderitem> Items { get; set; }
        public DateTime datezam { get; set; }
        public float Price { get; set; }
        public string Delivery { get; set; }
        public string City { get; set; }
        public string address { get; set; }
        public string postoffice { get; set; }
        public string Card_num { get; set; }

        public string Card_date { get; set; }
        public string Card_cvv { get; set; }

    }
}
using Microsoft.AspNetCore.Authorization;
using Microsoft.AspNetCore.Hosting;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using Microsoft.AspNetCore.Mvc.Rendering;
using Microsoft.EntityFrameworkCore;
using shop_proj.Models;

```

```

using System;
using System.Collections;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.IO;
using System.Linq;
using System.Threading.Tasks;

namespace shop_proj.Controllers
{
    [Authorize(Roles = "admin")]
    public class AdminController : Controller
    {
        private Models.AppContext db;
        IWebHostEnvironment _appEnvironment;
        public AdminController(Models.AppContext context, IWebHostEnvironment
appEnvironment)
        {
            db = context;
            _appEnvironment = appEnvironment;
        }

        public async Task<IActionResult> Index()
        {
            return View(await
db.Tovars.Include(p=>p.Modell).ThenInclude(c=>c.Category).ToListAsync());
        }
        public async Task<IActionResult> Categories()
        {
            return View(await db.Categories.Include(h=>h.Sex).ToListAsync());
        }
        public async Task<IActionResult> Addcategory()
        {
            List<Sex> sex = await db.Sexs.ToListAsync();
            ViewBag.Sex = sex;
            return View();
        }
        [HttpPost]
        public async Task<IActionResult> Addcategory(Category user)
        {
            var results = new List<ValidationResult>();
            var context = new ValidationContext(user);

            if (!Validator.TryValidateObject(user, context, results, true))
            {
                return View();
            }
            else
            {
                db.Categories.Add(user);
                await db.SaveChangesAsync();
                return RedirectToAction("EditCategory", new { id = user.Id });
            }
        }
        public async Task<IActionResult> EditCategory(int? id)
        {
            if (id != null)
            {
                Category user = await db.Categories.FirstOrDefaultAsync(p => p.Id == id);
                if (user != null)
                    return View(await db.Categories.Include(u =>
u.Modells).FirstOrDefaultAsync(p => p.Id == id));
            }
            return NotFound();
        }
    }
}

```

```

[HttpPost]
public async Task<IActionResult> EditCategory(Category user)
{
    db.Categories.Update(user);
    await db.SaveChangesAsync();
    return RedirectToAction("Index");
}
public async Task<IActionResult> Addtovar()
{
    SelectList sex = new SelectList(db.Sexs, "Id", "Name");
    ViewBag.Sex = sex;
    SelectList category = new SelectList(db.Categories.Where(p => p.SexId == 1),
    "Id", "Name");
    List<Modell> modell = await db.Modells.Where(p => p.CategoryId ==
db.Categories.Where(p => p.SexId == 1).FirstOrDefault().Id).ToListAsync();
    ViewBag.Category = category;
    ViewBag.Modell = modell;
    return View();
}
public IActionResult AddModell(int? id)
{
    ViewBag.Id = id;
    return View();
}
[HttpPost]
public async Task<IActionResult> AddModell(Modell user)
{
    Category category = await db.Categories.FirstOrDefaultAsync(p => p.Id ==
user.CategoryId);

    Modell modell= new Modell { Name = user.Name, Category = category };

    db.Modells.Add(modell);
    await db.SaveChangesAsync();
    return RedirectToAction("EditCategory", new { id = category.Id });
}

[HttpPost]
public async Task<IActionResult> Addtovar(Tovar user)
{
    if (ModelState.IsValid)
    {
        db.Tovars.Add(user);
        await db.SaveChangesAsync();
        return RedirectToAction("Tovarinf", new { id = user.Id });
    }
    else
    {
        SelectList sex = new SelectList(db.Sexs, "Id", "Name");
        ViewBag.Sex = sex;
        SelectList category = new SelectList(db.Categories.Where(p => p.SexId == 1),
    "Id", "Name");
        List<Modell> modell = await db.Modells.Where(p => p.CategoryId ==
db.Categories.Where(p => p.SexId == 1).FirstOrDefault().Id).ToListAsync();
        ViewBag.Category = category;
        ViewBag.Modell = modell;
        return View();
    }
}
public async Task<IActionResult> Tovarinf(int? id)
{
    if (id != null)
    {
        Tovar user = await db.Tovars.FirstOrDefaultAsync(p => p.Id == id);
    }
}

```

```

        if (user != null)
            return View(await db.Tovars.Include(u => u.Kinds).FirstOrDefaultAsync(p
=> p.Id == id));
    }
    return NotFound();
}

[HttpPost]
public async Task<IActionResult> Tovarinf(Tovar user)
{
    db.Tovars.Update(user);
    await db.SaveChangesAsync();
    return RedirectToAction("Index");
}
public IActionResult AddKind(int? id)
{
    ViewBag.Id = id;
    ViewBag.sizes = new List<String> { "XS", "S", "M", "L", "XL", "XXL", "XXL" };
    return View();
}
[HttpPost]
public async Task<IActionResult> Addkind(Kind user, List<IFormFile> uploadedFiles,
int xxs, int xs, int s, int m, int l, int xl, int xxl)
{
    Image img = new Image();
    Tovar tovar = await db.Tovars.FirstOrDefaultAsync(p => p.Id == user.TovarId);
    Size size = new Size();

    Kind kind = new Kind { Colour = user.Colour, Tovar = tovar };

    db.Kinds.Add(kind);
    foreach (var uploadedFile in uploadedFiles)
    {
        if (uploadedFile != null)
        {
            img.Name = uploadedFile.FileName;
            string path = "/Images/" + uploadedFile.FileName;
            using (var fileStream = new FileStream(_appEnvironment.WebRootPath +
path, FileMode.Create))
            {
                await uploadedFile.CopyToAsync(fileStream);
            }
            Image file = new Image { Name = uploadedFile.FileName, Path = path, Kind
= kind };
            db.Images.Add(file);
        }
    }
    List<Size> tt = new List<Size> { new Size("xxs", xxs),
new Size("xs",xs), new Size ("s",s), new Size("m",m), new Size("l",l), new
Size("xl",xl), new Size("xxl",xxl) };

    foreach (var item in tt)
    {
        if (item.Count != 0)
        {
            Size file = new Size { Name = item.Name, Count = item.Count, Kind = kind
};
            db.Sizes.Add(file);
        }
    }
    await db.SaveChangesAsync();
    return RedirectToAction("Tovarinf", new { id=tovar.Id });
}
public async Task<IActionResult> Editkind(int? id)

```

```

{
    if (id != null)
    {
        Kind user = await db.Kinds.FirstOrDefaultAsync(p => p.Id == id);
        if (user != null)
            return View(user);
    }
    return NotFound();
}
[HttpPost]
public async Task<IActionResult> Editkind(Kind user)
{
    db.Kinds.Update(user);
    await db.SaveChangesAsync();
    return RedirectToAction("Tovarinf", new { id = user.TovarId });
}
[HttpGet]
[ActionName("Delete")]
public async Task<IActionResult> ConfirmDelete(int? id)
{
    if (id != null)
    {
        Tovar user = await db.Tovars.FirstOrDefaultAsync(p => p.Id == id);
        if (user != null)
            return View(user);
    }
    return NotFound();
}

[HttpPost]
public async Task<IActionResult> Delete(int? id)
{
    if (id != null)
    {
        Tovar user = await db.Tovars.FirstOrDefaultAsync(p => p.Id == id);
        if (user != null)
        {
            db.Tovars.Remove(user);
            await db.SaveChangesAsync();
            return RedirectToAction("Index");
        }
    }
    return NotFound();
}
[HttpGet]
public ActionResult GetCategories(int id)
{
    SelectList category = new SelectList(db.Categories.Where(p => p.SexId == id),
    "Id", "Name");
    List<Modell> modell = db.Modells.Where(p => p.CategoryId ==
db.Categories.Where(p => p.SexId == id).FirstOrDefault().Id).ToList();
    ViewBag.Category = category;
    ViewBag.Modell = modell;
    return PartialView();
}
[HttpGet]
public ActionResult GetModells(int id)
{
    List<Modell> modell = db.Modells.Where(p => p.CategoryId == id).ToList();

    ViewBag.Modell = modell;
    return PartialView();
}
}
}

```

ДОДАТОК Б
(обов'язковий)

ПРЕЗЕНТАЦІЙНИЙ МАТЕРІАЛ

Презентація до кваліфікаційної роботи

На тему: Онлайн-ресурс для роздрібної торгівлі одягом

Автор роботи:

ст. гр. ІПЗ-19-1 Монастирський Т.А.

Керівник роботи:

к. т. н., доцент Яшина О. М.

Рисунок Б.1- Слайд №1

Актуальність проекту

- ▶ Інтернет-магазин об'єднує елементи прямого маркетингу з відвідуванням традиційного магазину. Відмінною рисою інтернет-магазинів у порівнянні зі звичайною формою торгівлі є те, що інтерактивний магазин може запропонувати значно більшу кількість товарів та послуг, та забезпечити споживачів значно більшим обсягом інформації, необхідним для ухвалення рішення про покупку. Крім того, за рахунок використання комп'ютерних технологій можлива персоналізація підходу до кожного з клієнтів, виходячи з історії його відвідувань магазину та зроблених раніше покупок.
- ▶ Інтернет-магазини з продажу одягу відрізняються, крім іншого, великою кількістю товарів та фільтрів, які допомагають вибрати ідеально адаптовані рішення та швидко знайти те, що шукає потенційний покупець. Інформація, доступна в інтернет-магазині, повинна бути правильно оформлена, щоб розвіяти сумніви і відповісти на будь-які питання.

Рисунок Б.2- Слайд №2

Мета роботи

- ▶ **Мета роботи** - проектування та розробка ресурсу, де при наявності мережі інтернет можна здійснювати купівлю одягу вроздріб. Дана мета свою чергу передбачає забезпечення зручної роботи користувачів зі всіма складовими сайту, такими як корзина, оформлення замовлення, а також можливість здійснення повернення товару.
- ▶ **Об'єкт дослідження:** процес інтернет-обслуговування користувачів з питань замовлення та купівлі одягу.
- ▶ **Предмет дослідження:** організація обліку товарів, методи проектування функціоналу та зручного інтерфейсу, а також принципи створення інтернет-магазинів.

Рисунок Б.3- Слайд №3

Основні завдання

Основні завдання :

- аналіз та дослідження сфери купівлі-продажу одягу через інтернет;
- розробка технічного завдання;
- проектування, зокрема бази даних;
- створення зручного, адаптивного інтерфейсу;
- розробка програмних модулів;
- наповнення контентом;
- виконати тестування.

Рисунок Б.4- Слайд №4

Аналіз предметної області

Розроблюване програмне забезпечення створюється для здійснення автоматизації у режимі ручного вибору одягу, а також замовлення із можливістю оплати у режимі онлайн.

Предметною областю даного дослідження є торгівля одягом за допомогою веб-ресурсу в мережі Інтернет.

В ході даного дослідження було виокремлено межі даної предметної області.

Рисунок Б.5- Слайд №5

Аналіз стану проблеми та інших рішень

- ▶ При порівнянні різних інтернет-магазинів одягу спрямовується увага на такі фактори:
 - безпосередня робота ресурсу, тобто наскільки зручно користуватися, чи доцільна та продумана навігація, розміщення різноманітних елементів;
 - наявний асортимент товарів та послуг – чи є великою кількістю товарів та послуг, а також їх доступність для споживача;
 - ціни – порівняння вартості товарів з конкурентами, наявність спец-пропозицій;
 - підтримка та консультації – швидкість відповідей на звернення покупця, опрацювання замовлення, допомога у вирішенні труднощів;
 - зручність оплати – великий вибір способів оплати, можливість розтермінування та кредиту;
 - доставка – швидкість опрацювання, кількість способів доставки;
 - акції, знижки, розпродажі – можливість купити дешевше в інтернет-магазині;
 - гарантійні зобов'язання – повернення товару, ремонт та обслуговування.

Рисунок Б.6- Слайд №6

Визначено вимоги до ресурсу

- авторизація користувачів в системі: В системі повинна бути представлена можливість реєстрації користувача та присвоєння йому відповідної ролі (замовник, адміністратор). В інакшому випадку користувач є гостем і не може користуватися всіма функціями сайту;
- розробка структури сайту і схем сторінок;
- розробка структури сторінки каталогу товарів;
- можливість збереження певних товарів у кошику до оформлення замовлення;
- можливість зворотного зв'язку із адміністрацією сайту;
- підготовка прав на різні типи операцій на сайті для груп користувачів;

Рисунок Б.7- Слайд №7

Діаграма варіантів використання

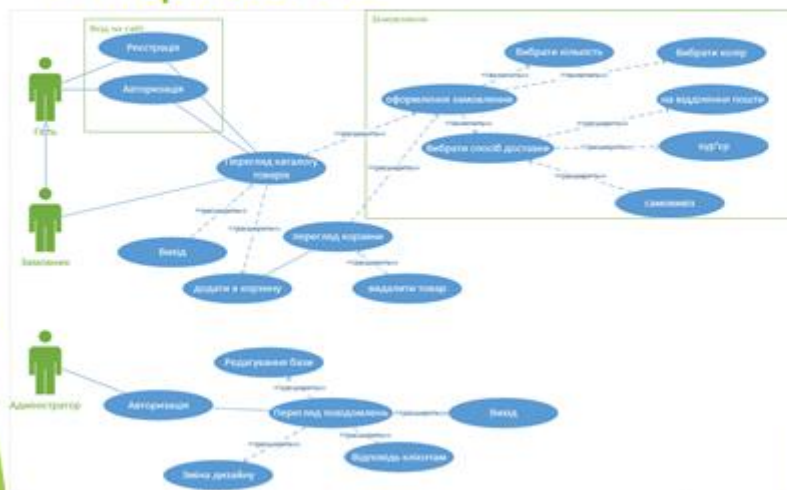


Рисунок Б.8- Слайд №8

Архітектура

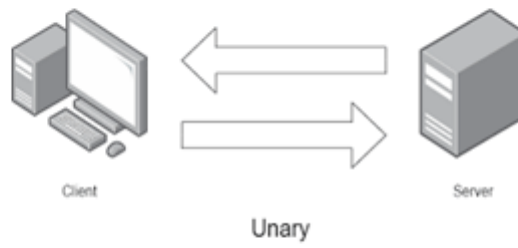


Рисунок Б.9- Слайд №9

Модель бази даних

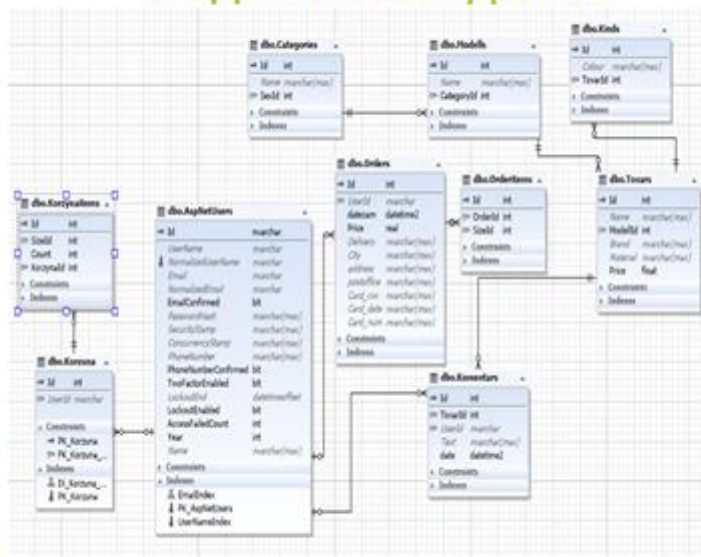


Рисунок Б.10- Слайд №10

Результати роботи ГОЛОВНА сторінка

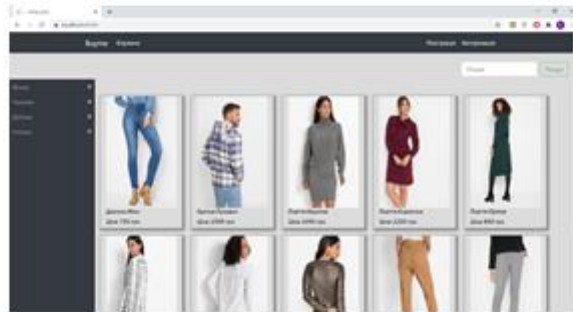


Рисунок Б.11- Слайд №11

Сторінка одного товару

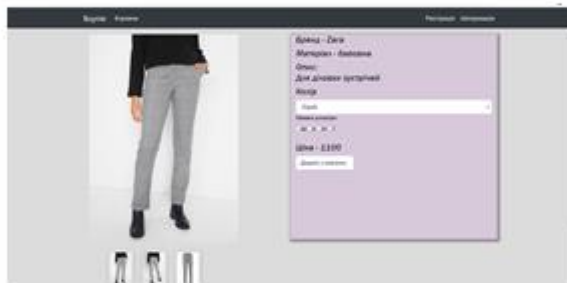
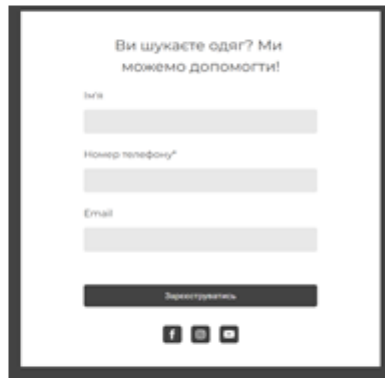


Рисунок Б.12- Слайд №12

Реєстрація користувача



Ви шукаєте одяг? Ми можемо допомогти!

Ім'я

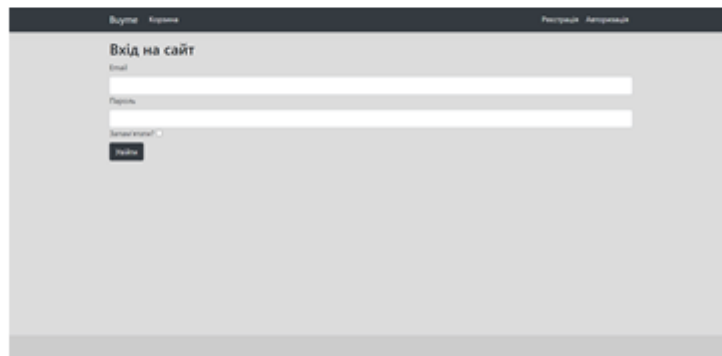
Номер телефону*

Емейл

[f](#) [g+](#) [v](#)

Рисунок Б.13- Слайд №13

Сторінка авторизації



Вхід на сайт

Емейл

Пароль

Запам'ятати?

Рисунок Б.14- Слайд №14

Сторінка адміністратора

The image displays two screenshots of a web application's administrator interface. The top screenshot shows a form titled "Додавання нового товару" (Adding a new item) with fields for Name, Type, Color, Size, Date, and a "Додати новий товар" button. The bottom screenshot shows a form titled "Товар" (Item) with fields for Name, ID, Material, Color, Date, and a "Кольори" section with a "Додати" button.

Рисунок Б.15- Слайд №15

Отримані результати

В результаті виконання даної кваліфікаційної роботи роботи отримано сервіс, що дозволяє здійснювати вибір та покупку одягу в роздріб (зі сторони клієнта) поповнення асортименту та продаж одягу (зі сторони продавця). Це може відбуватись автоматично у ручному режимі за допомогою мережі Інтернет в режимі онлайн.

Рисунок Б.16- Слайд №16

Висновки

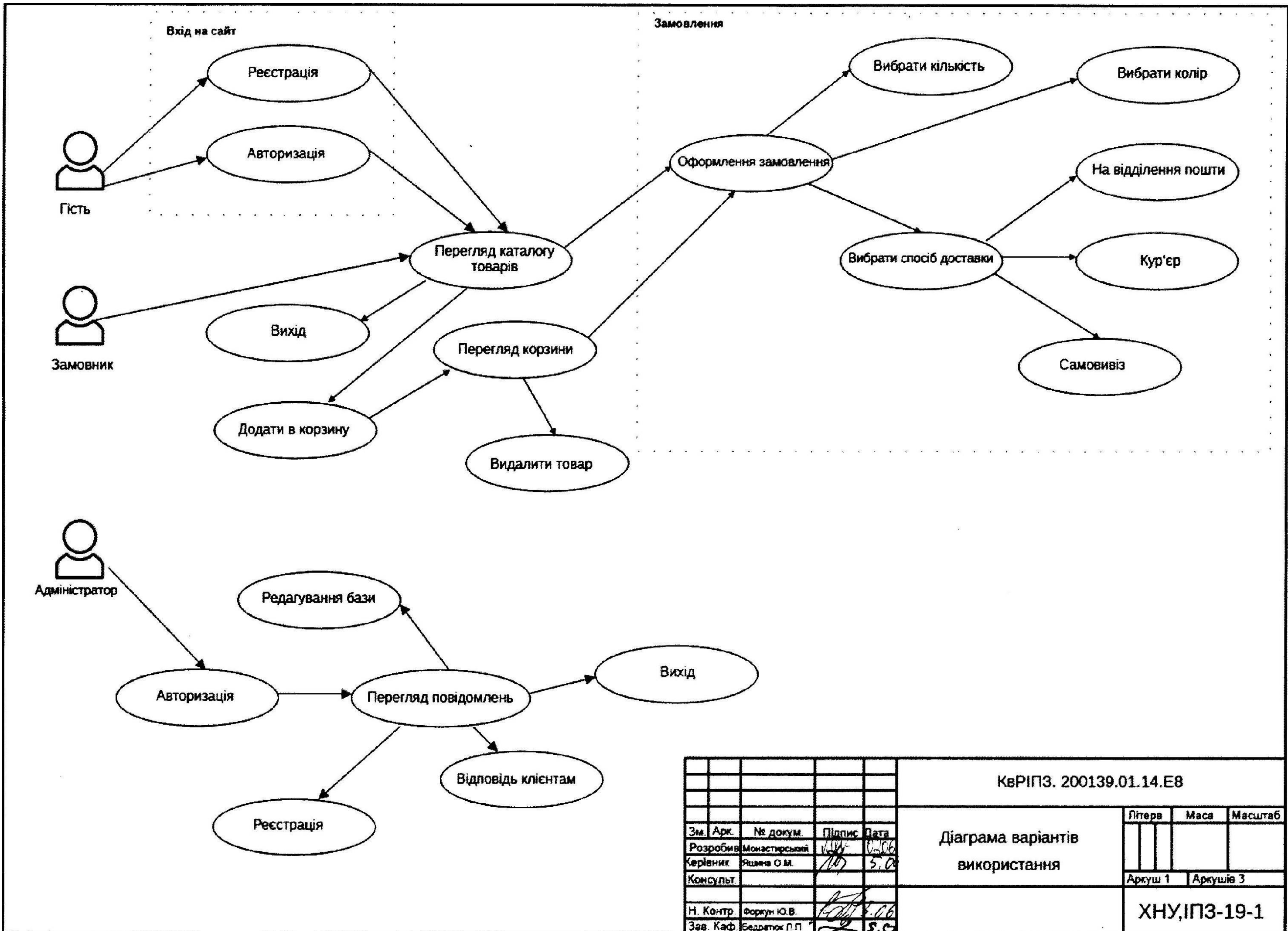
- проаналізовано предметну область із визначенням її головних особливостей;
- зроблено аналіз існуючого програмного забезпечення;
- розроблено технічне завдання (ТЗ);
- зроблено проектування програмного продукту;
- виконано програмну реалізацію проекту;
- здійснено тестування онлайн-ресурсу.

Рисунок Б.17- Слайд №17

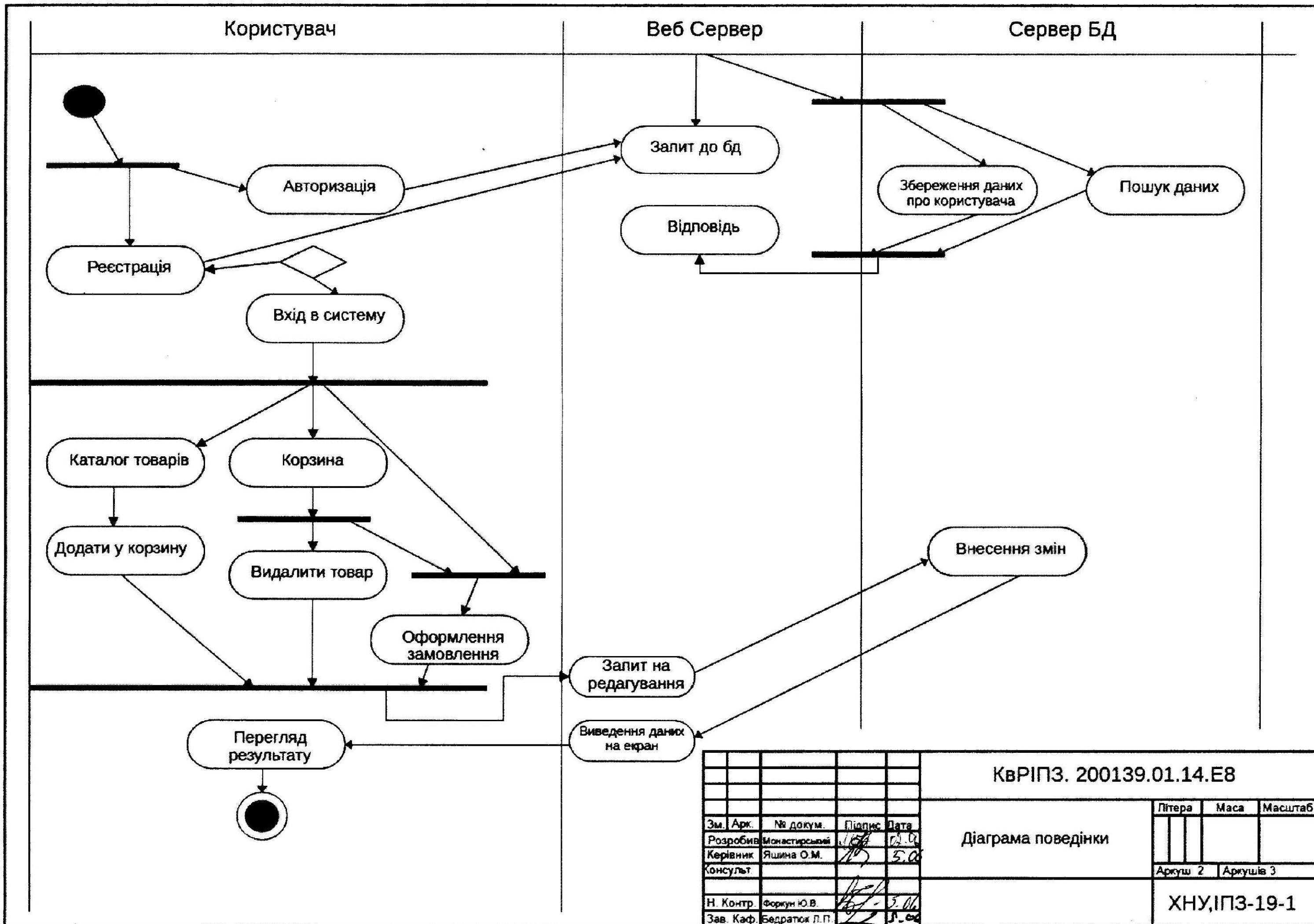
Дякую за увагу!

Рисунок Б.18- Слайд №18

ГРАФІЧНА ЧАСТИНА
(обов'язкова)



					КвРІПЗ. 200139.01.14.Е8			
Зм.	Арк.	№ докум.	Підпис	Дата	Діаграма варіантів використання	Літера	Маса	Масштаб
Розробив	Монастирський		<i>[Signature]</i>	03.06				
Керівник	Яценко О.М.		<i>[Signature]</i>	3.06		Аркуш 1	Аркушів 3	
Консульт.								
Н. Контр.	Форгун Ю.В.		<i>[Signature]</i>	3.06	ХНУ,ІПЗ-19-1			
Зав. Каф.	Бедричак П.П.		<i>[Signature]</i>	3.06				



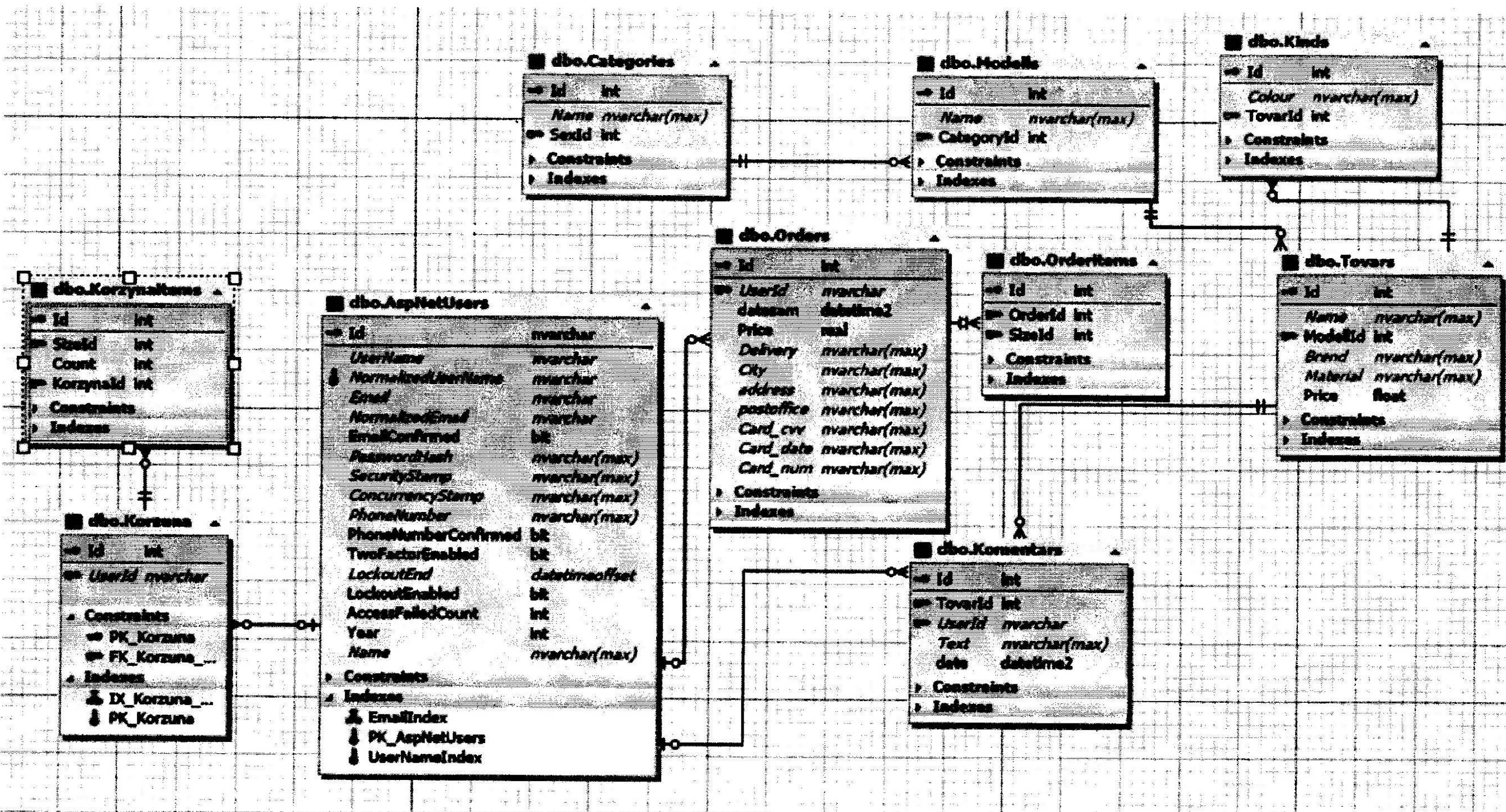
КВРІПЗ. 200139.01.14.E8

Зм.	Арк.	№ докум.	Підпис	Дата
Розробив		Монастирський	<i>[Signature]</i>	02.09
Керівник		Яшина О.М.	<i>[Signature]</i>	3.09
Консульт.				
Н. Контр.		Форкун Ю.В.	<i>[Signature]</i>	3.09
Зав. Каф.		Бедратюк Л.П.	<i>[Signature]</i>	3.09

Діаграма поведінки

Літера	Маса	Масштаб
Аркуш 2	Аркушів 3	

ХНУ,ІПЗ-19-1



					КВРІПЗ. 200139.01.14.Е8		
Зм	Арх.	№ докум	Підпис	Дата	Схема бази даних		
Розробив	Монастирський	10077	10.06				
Керівник	Яшина О.М	100	5.06				
Консульт.							
Н. Контр.	Форкун Ю.В		5.06		Літера	Маса	Масштаб
Зав. Каф.	Бедрачок Л.Г		5.06		Аркуш 3	Аркушів 3	
					ХНУ,ІПЗ-19-1		

Завідувачу кафедри
інженерії програмного забезпечення
проф. Бедратюку Л. П.
студента групи ІІЗ-19-1
Монастирського Т.А.
Прізвище, ініціали

ЗАЯВА

Прошу закріпити за мною тему кваліфікаційної роботи освітнього ступеня
«бакалавр» за спеціальністю 121 «Інженерія програмного забезпечення»:

Онлайн-ресурс для роздрібної торгівлі одягом.

(керівник роботи – Яшина Оксана Миколаївна)
Прізвище, ім'я, по батькові

02.06.2023
Дата


Підпис студента

Завідувачу кафедри інженерії програмного
забезпечення проф. Бедратюку Л. П.

здобувача вищої освіти

Монастирського Т.А.

Прізвище, ініціали

факультет ІТ, 4 курс, група ІПЗ-19-1

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності в Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту і застосування заходів дисциплінарної та академічної відповідальності, ознайомлений. Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений та надаю свою згоду на обробку й збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Unicheck та/або Anti-Plagiarism) і використання роботи для виявлення академічного плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

12.06.23р.

дата



підпис

Anti-Plagiarism v-15.257

Максимальне співпадіння з одним документом 4.0%

Словники перевірки: en_US, ru_RU, ua_UA. Помилки в документах: 7%

ID: 114637 Назва: БКР Онлайн-ресурс для роздрібної торгівлі одягом Додано в БД: 2023-06-04 Автора: Монастирський Т.А. Керівники: Яшина О.М. к.т.н. доц. Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	60212	939	5760 (10%)	94 (10%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми



Ім'я користувача:
Кафедра ІПЗ

Дата перевірки:
04.06.2023 08:37:50 EEST

Дата звіту:
04.06.2023 08:40:00 EEST

ID перевірки:
1015409983

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100005589

Назва документа: КаР_Монастирський_плагіат

Кількість сторінок: 72 Кількість слів: 10972 Кількість символів: 85246 Розмір файлу: 2.34 MB ID файлу: 1015073345

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

19.6% Схожість

Найбільша схожість: 5.7% з джерелом з Бібліотеки (ID файлу: 1011231407)

18.1% Джерела з Інтернету

366

Сторінка 74

8.62% Джерела з Бібліотеки

87

Сторінка 76

0% Цитат

Вилучення цитат вимкнено

Вилучення списку бібліографічних посилань вимкнено

0% Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

1

Підозріле форматування

15
сторінок

**РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ
КАФЕДРИ ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ
ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ**

Підтверджуємо ознайомлення з результатами звіту/звітів подібності щодо роботи, продуктованими програмно-технічним засобом (ами) перевірки текстів на плагіат:

Назва: «Онлайн-ресурс для роздрібної торгівлі одягом»

Автор: Монастирський Тарас Андрійович

Спеціальність: 121 – Інженерія програмного забезпечення

Освітня програма: Освітньо-професійна програма «Інженерія програмного забезпечення»

Науковий керівник: Яшина Оксана Миколаївна, кандидат технічних наук, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої й електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того, як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
5	Інше:	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) у тексті кваліфікаційної роботи системами перевірки на плагіат виявлено схожість з деякими документами в частині загальноживаних обов'язкових словосполучень у стандартних бланках (титулка, відомість документів), у структурі змісту, назвах розділів/підрозділів тощо, у назвах публікацій у переліку джерел посилання;

2) в якості запозичень системою було зафіксовано деякі послідовності вихідного коду і посилання на бібліотеки, які є стандартними мовними конструкціями програмування та не можуть розглядатися як об'єкт авторських прав і, відповідно, їх порушення;

3) усі запозичення є фрагментарними або мають належним чином оформленні посилання;

4) виявлені модифікації тексту не впливають на відсоток схожості.

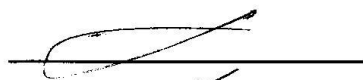


Сумарний обсяг всіх запозичень, визначений системою виявлення збігів ідентичності/схожості, складає 19,6% і адресується до 453 джерел, що, з урахуванням наведених обґрунтувань, відповідає характеру теми і свідчить на користь кваліфікаційної роботи.

Дата 05.06.23р.

Завідувач кафедри

Гарант освітньої програми

Керівник кваліфікаційної роботи

Леонід БЕДРАТЮК

Леонід БЕДРАТЮК

Оксана ЯШИНА

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ
освітнього ступеня «Бакалавр»

Дипломник Монастирський Тарас Андрійович

Тема Онлайн-ресурс для роздрібної торгівлі одягом

Спеціальність 121 – Інженерія програмного забезпечення

Обсяг кваліфікаційної роботи:

Кількість листів креслень 3 ; кількість сторінок записки 85 ;

1. Короткий зміст пояснювальної записки та прийнятих рішень У даній кваліфікаційній роботі досліджено і проаналізовано предметну область. Онлайн сервіси з продажу одягу відрізняються, крім іншого, великою кількістю товарів та фільтрів, які допомагають вибрати ідеально адаптовані рішення та швидко знайти те, що шукає потенційний покупець. Інформація, доступна в інтернет-магазині, повинна бути правильно оформлена, щоб розвіяти сумніви і відповісти на будь-які питання. Був проведений аналіз лідируючих інтернет магазинів в Україні та світі. Оцінено їхні слабкі та сильні сторони функціоналу, для редагування та удосконалення при створенні свого програмного продукту. Після завершення створення онлайн-ресурсу було проведене багатоетапне тестування, в результаті якого доведено, що кінцевий продукт безпроблемно та коректно працює, та готовий для використання в середовищі інтернет, для роздрібної інтернет торгівлі.

2. Висновок про відповідність роботи поставленому завданню Кваліфікаційна робота виконана відповідно до поставленого завдання та з дотриманням всіх вимог.

3. Характеристика виконання кожного розділу роботи, ступінь використання останніх досягнень науки і техніки та передових методів роботи Вступ містить аргументовані пояснення актуальності даної теми, визначено мету та завдання для кваліфікаційної роботи. У першому розділі даної кваліфікаційної роботи було здійснено детальний аналіз предметної області із виокремленням її меж, функціональних та нефункціональних вимог, проаналізовано існуючі онлайн-ресурси з продажу одягу онлайн, розроблено технічне завдання. У другому розділі було здійснено проектування всіх основних складових системи, до яких відноситься архітектура, логічна модель бази даних, користувацький інтерфейс. В результаті написання третього розділу було здійснено визначення загальної структури основних програмних модулів, створено моделі із наступною реалізацією баз даних із встановленими зв'язками між її таблицями. Після цього було запрограмовано всю логіку додатку MVC, включаючи 4 контролери та представлення для їх методів. Також було розроблено інструкції для користувачів системи: адміністратора та покупця.

4. Позитивні сторони роботи Тематика кваліфікаційної роботи є актуальною, оскільки 60% податків в Україні сплачується саме малими бізнесами які здебільшого і реалізують свої товари, дуже часто це саме роздрібна торгівля одягом, завдяки онлайн-ресурсам. В даній

кваліфікаційній роботі застосовано найкращі технологічні рішення які існують на сьогоднішній день. Це дало змогу розробити актуальний програмний продукт, який виконує свою мету та завдання.

5. Негативні сторони роботи У процесі редагування товарів від ролі адміністратора реалізовано лише редагування товару по кожній окремій позиції – було б доцільно додати редагування зразу декількох товарів наприклад по критерію –ціна.

6. Оцінка графічного оформлення та пояснювальної записки Графічне оформлення виконано відповідно до теми кваліфікаційної роботи та подано у вигляді діаграм і рисунків. Пояснювальна записка оформлена згідно вимог чинних стандартів.

7. Відгук про кваліфікаційну роботу в цілому Кваліфікаційна робота заслуговує позитивної оцінки. Пояснювальна записка містить правильно структурований, просто сформульований матеріал, що дозволяє легко та чітко зрозуміти всі етапи, результати та інструкцію з користування даним програмним продуктом. Якісно створений графічний матеріал, дозволяє наочно побачити структуру складових деталей проектування системи.

8. Інші зауваження _____

9. Оцінка кваліфікаційної роботи Кваліфікаційна робота виконана згідно всіх чинних стандартів у повному обсязі, відповідає поставленій задачі, містить всі поставленні завдання та заслуговує на оцінку «добре».

РЕЦЕНЗЕНТ Лисенко Сергій Миколайович, доктор технічних наук, професор кафедри комп'ютерної інженерії та інформаційних систем (КІІС) ХНУ

“ 02 ” червня 2023 р. _____
(підпис)