

Хмельницький національний університет
Факультет інформаційних технологій
Кафедра інженерії програмного забезпечення

КВАЛІФІКАЦІЙНА РОБОТА

«Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень»

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.2201108.01.14.ПЗ

Виконав студент IV курсу групи ПЗ-22-1



Підпис

Іван РАЙЧУК

Ім'я, прізвище

Керівник канд. пед. наук, доцент

Науковий ступінь, звання


Підпис

Оксана ОНИШКО

Ім'я, прізвище

Нормоконтролер канд. пед. наук, доцент


Підпис

Наталія ПРАВОРСЬКА

Ім'я, прізвище

До захисту допускаю:

Завідувач кафедри інженерії
програмного забезпечення


Підпис

Леонід БЕДРАТЮК

Ім'я, прізвище

2 червня 2026 р.

Хмельницький 2026

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

02 01 2026 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Райчуку Івану Вадимовичу

Прізвище, ім'я, по батькові студента

1. Тема роботи Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень

Керівник роботи Онишко Оксана Григорівна, канд. пед. наук, доцент

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

Затверджена наказом ректора університету від 20.01.2026 р. №7

2. Строк подання студентом роботи на кафедру 01.06.26 р.

3. Вихідні дані до роботи Методичні матеріали до кваліфікаційної роботи

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити) Аналіз предметної області, визначення її функціональних та нефункціональних вимог, постановка задачі, проектування програмного забезпечення, програмна реалізація, проведення тестування вебресурсу.

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

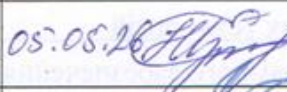
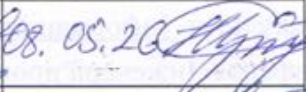
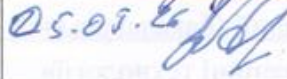
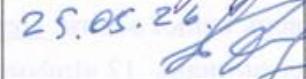
Три креслення:

1. Діаграма варіантів використання для зареєстрованого користувача

2. Схема бази даних

3. Архітектурне рішення.

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Праворська Н. І., доцент кафедри ІПЗ	05.05.26 	08.05.26 
Антиплагіат	Форкун Ю. В., доцент кафедри ІПЗ	05.05.26 	25.05.26 

7. Дата видачі завдання «02» січня 2026 р.

КАЛЕНДАРНИЙ ПЛАН

Назва етапів (розділів) кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1 Ознайомлення з тематикою кваліфікаційної роботи (КвР), визначення та узгодження індивідуальних тем КвР	01.12 – 31.12.2025	
2 Збір матеріалу за темою КвР; дослідження предметної області, в якій планується використання програмного забезпечення (ПЗ), визначення задач та вимог, розробка технічного завдання	02.01 – 31.01.2026	
3 Проектування програмного забезпечення	01.02 – 28.02.2026	
4 Програмна реалізація з використанням відповідних засобів розробки	01.03 – 10.04.2026	
5 Тестування програмного забезпечення	11.04 – 30.04.2026	
6 Написання вступу, загальних висновків, оформлення переліку джерел посилання та додатків. Оформлення пояснювальної записки КвР згідно вимог	01.05 – 25.05.2025	
7 Попередній захист КвР	травень 2026 (згідно графіка)	
8 Перевірка КвР на плагіат, нормоконтроль, отримання відгуків, рецензій та інших супровідних документів. Брошування (зшиття) пояснювальної записки.	26.05 – 30.05.2026	
9 Здача КвР на кафедрі; підготовка КвР для розміщення у репозиторії ХНУ; підготовка до захисту та захист КвР	з 01.06.2026	

Студент


Підпис

Іван РАЙЧУК

Ініціали, прізвище

Керівник роботи


Підпис

Оксана ОНИШКО

Ініціали, прізви

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень».

Автор роботи: Райчук Іван Вадимович.

Керівник роботи: Онишко Оксана Григорівна.

Пояснювальна записка: 83 с., 19 рис., 5 табл., 2 дод., 30 джерел.

ВЕБТЕХНОЛОГІЇ, ОНЛАЙН-РЕСУРС, ВЕБРЕСУРС, ІНТЕРНЕТ-МАГАЗИН, ЕЛЕКТРОННА КОМЕРЦІЯ, РОЗРОБКА ПРОГРАМНОГО ПРОДУКТУ, ПРОДАЖ ЗАРЯДНИХ ПРИСТРОЇВ, ВЕБЗАСТОСУНОК.

Метою кваліфікаційної роботи є розробка вебресурсу для електронної торгівлі зарядними пристроями із кошиком замовлень.

Під час роботи над програмним забезпеченням здійснено аналітичний огляд предметної області, існуючих програмних рішень та засобів, окреслено мету та завдання для кваліфікаційної роботи, в кінці кожного розділу зроблено висновки, а також подано загальні висновки.

Вебзастосунок розроблявся за допомогою технологій обробки даних RESTful API сервер, що побудований на платформі Node.js з використанням фреймворку Express.

Результатом кваліфікаційної роботи став вебресурс для електронної торгівлі зарядними пристроями, що містить кошик замовлень. Це надасть змогу покупцям здійснювати пошук, фільтрування та покупку потрібного зарядного пристрою чи станції, завдяки наявності доступу до мережі Інтернет.

29.05.20

Дата


Підпис





ВІДОМІСТЬ ДОКУМЕНТІВ

№ рядка	Формат	Позначення документа	Найменування документа	К-сть аркушів	№ екз.	Примітка
			<u>Текстові документи</u>			
1	A4	КвРІПЗ.2201108.01.14.E8	Пояснювальна записка	83		
2	A4		Завдання на кваліфікаційну роботу	2		
3	A4		Анотація	1		
			<u>Графічні документи</u>	3		
4	A3	КвРІПЗ.2201108.01.14.E8	Діаграма варіантів використання	1		
5	A3	КвРІПЗ.2201108.01.14.E8	Схема бази даних	1		
6	A3	КвРІПЗ.2201108.01.14.E8	Архітектура застосунку	1		

КвРІПЗ.2201108.01.14.ПЗ								
Змн.	Арк.	№ докум.	Підпис	Дата	Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень Відомість документів	Літ.	Арк.	Аркушів
Виконав		Райчук І.В.		29.05			1	1
Керівник		Онишко О.Г.		29.05				
Н. контр.		Праворська Н.І.		29.05				
Зав. каф.		Бедратюк Л.П.		29.05				
						ХНУ, ІПЗ-22-1		

ЗМІСТ

Вступ.....	8
1 Дослідження предметної області та постановка задачі.....	10
1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей.....	10
1.2 Аналіз наявного програмно-технічного забезпечення предметної області.....	17
1.3 Визначення функціональних та нефункціональних вимог.....	24
2 Проектування вебресурсу.....	30
2.1 Проектування архітектури та структури вебресурсу.....	30
2.2 Проектування логічної моделі бази даних.....	33
2.3 Проектування інтерфейсу користувача.....	38
2.4 Аналіз та вибір технологій і методів реалізації вебресурсу.....	43
3 Програмна реалізація та тестування.....	46
3.1 Реалізація бази даних.....	46
3.2 Реалізація інтерфейсу користувача. Інструкція користувача.....	48
3.3 Тестування вебресурсу.....	53
Висновки.....	66
Перелік джерел посилання.....	68
Додаток А Презентаційні матеріали.....	71
Додаток Б Технічне завдання.....	78

					КвРІПЗ.2201108.01.14.ПЗ			
Змн.	Арк.	№ докум.	Підпис	Дата	Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень	Літ.	Арк.	Аркушів
Виконав		Райчук І.В.		29.05			6	88
Керівник		Онишко О. Г.		29.05				
Н. контр.		Праворська Н.І.		29.05				
Зав. каф.		Бедратюк Л.П.		29.05				ХНУ, ІПЗ-22-1

ВСТУП

В умовах повномасштабної війни в Україні розробка вебресурсу для торгівлі зарядними пристроями перетворюється з комерційного проекту на створення критично важливого інструменту забезпечення життєдіяльності та зв'язку.

Сьогодні наявність якісного зарядного обладнання безпосередньо корелює з рівнем безпеки громадян, оскільки в умовах постійних обстрілів енергетичної інфраструктури можливість швидко та ефективно поповнити енергозапас гаджетів стає питанням виживання.

Спеціалізований ресурс стає актуальним через гостру потребу в енергонезалежності, де звичайні побутові адаптери поступаються місцем потужним GaN-зарядкам та інтелектуальним системам Power Delivery. Такі пристрої дозволяють максимально ефективно використовувати короткі проміжки часу, коли електропостачання відновлюється, заряджаючи ноутбуки та павербанки на максимальній швидкості. Крім того, ринок потребує професійної фільтрації товарів, яка допомогла б цивільним та волонтерам безпомилково обирати кабелі-тригери та перехідники для живлення роутерів чи терміналів Starlink від портативних джерел живлення.

Вебресурс також виконує важливу роль у протидії ризикам використання низькоякісної продукції, яка в умовах нестабільної напруги в мережі може призвести до займання або виходу з ладу дорогої техніки.

Створення платформи, що фокусується на сертифікованому обладнанні з вбудованим захистом від перепадів напруги, є відповіддю на запит суспільства щодо збереження працездатності електроніки.

Таким чином, актуальність розробки полягає у формуванні надійного ланцюга постачання інструментів енергопідтримки, що забезпечують безперебійну роботу зв'язку, навігації та медичного обладнання в екстремальних умовах.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		8

Оскільки дана тематика є надзвичайно актуальною в умовах сьогодення, то мета кваліфікаційної роботи – це розробка вебресурсу для електронної торгівлі зарядними пристроями із кошиком замовлень.

В ході виконання кваліфікаційної роботи повинні бути вирішені такі завдання:

- провести аналітичний огляд предметної області;
- провести аналіз вже існуючих вебзастосунків, що допомагають здійснити торгівлю зарядними пристроями;
- визначити переваги та недоліки проаналізованих бізнес-рішень;
- здійснити постановку задачі та розробити технічне завдання;
- здійснити вибір та опис архітектури розроблюваного програмного продукту;
- провести аналіз та вибір технологій для розробки програмного продукту;
- реалізувати та протестувати розроблений програмний продукт.

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		9

1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Змістовний аналіз предметної області, її структурних та функціональних особливостей

У часи повномасштабного вторгнення роль спеціалізованих вебресурсів зазнає фундаментальної трансформації. Вони перестають бути звичайними майданчиками для електронної комерції та перетворюються на цифрові вузли критичної інфраструктури, що забезпечують життєдіяльність цивільного населення та ефективність економічного фронту.

Масовані атаки на енергетичну систему України зробили доступ до електроенергії питанням фізичного виживання. Зарядні пристрої та станції сьогодні гарантують безперебійну роботу смартфонів, які є основним джерелом сповіщень про небезпеку та засобом зв'язку з рідними. Ще критичнішою є роль автономного живлення в медицині, де робота кисневих концентраторів або холодильників для ліків безпосередньо залежить від надійності павербанків великої ємності та портативних станцій. Таким чином, вебресурс виконує функцію прямого постачальника інструментів для збереження життя.

Окрім гуманітарного аспекту, подібні платформи підтримують економічну витривалість держави. Малий та середній бізнес, від невеликих аптек до кав'ярень, адаптується до блекаутів завдяки швидкому доступу до професійних енергетичних рішень. Можливість онлайн-моніторингу характеристик та оперативного замовлення дозволяє підприємцям підтримувати роботу навіть у найскладніші періоди, що зберігає робочі місця та наповнює бюджет країни.

Важливою є також просвітницька місія ресурсу. В умовах ажіотажного попиту та ризику придбання небезпечного обладнання, вебресурс стає джерелом експертних знань. Пояснення технічних нюансів щодо безпеки використання різних типів акумуляторів у житлових приміщеннях запобігає

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		10

побутовим пожежам та нещасним випадкам. Калькулятори потужності допомагають громадянам раціонально використовувати власні кошти, обираючи обладнання, що точно відповідає їхнім критичним потребам.

Специфіка війни також диктує особливі вимоги до мобільності та автономності. Вебресурс акумулює рішення, що дозволяють людям бути енергонезалежними навіть у тимчасових укриттях або під час евакуації. Волонтерські організації та військові підрозділи часто використовують такі цивільні платформи для швидкого закриття потреб фронту, адже портативні станції є незамінними для заряджання дронів, засобів зв'язку та розвідувальних планшетів.

Зрештою, існування надійного та зрозумілого енергетичного ресурсу має потужний психологічний ефект. Коли громадянин має змогу підготуватися до ймовірних викликів зими чи чергових атак, рівень суспільної тривоги знижується. Відчуття контролю над базовими умовами побуту посилює загальну стійкість нації. Отже, VoltSpace виступає не просто магазином, а стратегічним інструментом адаптації українського суспільства до екстремальних умов війни, сприяючи формуванню повної енергетичної автономії на рівні кожного домогосподарства.

Розробка спеціалізованого вебресурсу для торгівлі зарядними пристроями є стратегічно важливою через трансформацію ринку споживчої електроніки та зміну стандартів енергоспоживання.

Ось ключові аргументи, що підтверджують актуальність такого роду проєкту:

- ефект так званої порожньої коробки;
- перехід на технологію GaN (Gallium Nitride);
- складна енергетична ситуація в Україні.

Більшість провідних виробників смартфонів, таких як, Apple, Samsung, Google, Хіаомі остаточно відмовилися від комплектації гаджетів блоками живлення. Це перетворило зарядний пристрій із аксесуара в комплекті на

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		11

обов'язкову окрему покупку. Попит на зовнішні зарядні пристрої став стабільним та масовим.

Старі кремнієві зарядки стають історією, бо технологія GaN дозволяє створювати надпотужні пристрої від 65W до 240W у компактному корпусі. Користувачам потрібен ресурс, який допоможе замінити три різні блоки, наприклад, для ноутбука, планшета та телефона одним універсальним GaN-адаптером.

Через ризики нестабільного енергопостачання для українського споживача критично важливими стали:

1. Швидкість заряджання та попит на протоколи Quick Charge 5.0 та Power Delivery 3.1, що дозволяють зарядити пристрій за лічені хвилини.

2. Спеціалізовані рішення, що забезпечують кабелі та адаптери для живлення роутерів DC/USB-C Step-up, які підтримують роботу інтернету під час відключень.

3. Уніфікація за стандартом USB-C, тобто введення єдиного стандарту USB-C у ЄС та світі спрощує логістику, але водночас заплутує користувачів у характеристиках (ватах, амперах, вольтах).

Спеціалізований сайт може запропонувати розумні фільтри, які підберуть зарядку саме під конкретну модель техніки, уникаючи ризику пошкодження акумулятора.

4. Ринок портативного транспорту та гаджетів. Зарядні пристрої сьогодні - це не лише про телефони. Це сегмент, що включає:

- зарядки для електросамокатів та моноколіс;
- станції для дронів (актуально для волонтерського та комерційного секторів);
- бездротові магнітні станції по типу MagSafe та стандарти Qi2.

Більшість маркетплейсів пропонують тисячі низькоякісних копій, які псують техніку. Актуальність проекту полягає у експертному підборі сертифікованого обладнання, де покупець впевнений у безпеці та відповідності заявлених характеристик реальним.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		12

Для ефективної роботи такого вебресурсу в умовах блекаутів, система фільтрації має базуватися на конкретних сценаріях використання та технічних стандартах, які забезпечують сумісність пристроїв у критичних ситуаціях.

Нижче наведено перелік ключових параметрів, які варто інтегрувати в логіку пошуку та картки товарів. Технічні параметри для фільтрації включають протоколи швидкої зарядки та стандарт Power Delivery, тип компонентної бази, вихідну напругу для DC-ліній, кількість та тип портів або комбіноване навантаження, захисні характеристики (Safety Features).

Протоколи швидкої зарядки та стандарт Power Delivery (PD). В умовах обмеженого часу доступу до мережі, фільтрація за потужністю від 20W для смартфонів до 100W+ для ноутбуків є першочерговою. Важливо виокремити підтримку стандарту PPS (Programmable Power Supply), який дозволяє зарядним пристроям динамічно змінювати напругу та струм, мінімізуючи нагрівання акумулятора та прискорюючи процес заряджання.

Тип компонентної бази (GaN або Silicon). Фільтр за технологією GaN (нітрид галію) дозволяє користувачам знаходити найбільш енергоефективні та компактні пристрої. Це критично для тих, хто постійно переміщується або має обмежений простір у тривожній валізі чи рюкзаку, оскільки один такий блок замінює декілька важких адаптерів.

Вихідна напруга для DC-ліній, а саме живлення роутерів, куди входить окремий фільтр для кабелів-тригерів та адаптерів за параметрами 9V або 12V. Це дозволить власникам павербанків з функцією PD швидко знайти кабель, що перетворює вихідний сигнал павербанка у потрібну напругу для роботи Wi-Fi роутера або оптичного терміналу (ONU) під час відсутності світла.

Кількість та тип портів або комбіноване навантаження. Фільтрація за комбінацією портів USB-C та USB-A з описом розподілу потужності. Користувач повинен бачити, яку потужність видасть кожен порт при одночасному підключенні двох або трьох пристроїв, щоб уникнути перевантаження та забезпечити зарядку ноутбука паралельно зі смартфоном.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		13

Захисні характеристики (Safety Features). В умовах нестабільної напруги та роботи від генераторів, критичним є фільтр за наявності вбудованого захисту від:

- OVP (Over Voltage Protection) - захист від перенапруги;
- OTP (Over Temperature Protection) - захист від перегріву;
- SCP (Short Circuit Protection) - захист від короткого замикання.

Для того щоб покупець міг швидко прийняти рішення в стресових умовах, картка товару має бути максимально інформативною та технічно точною. Нижче наведено структуру, яка допоможе користувачу миттєво зрозуміти, чи підійде цей пристрій для його сценарію енергонезалежності.

Структура картки товару містить візуальний блок та статуси, блок енергетичної сумісності, секція готовності до блекауту.

Візуальний блок та статуси. Окрім стандартних фото, обов'язково додається інфографіка з реальною потужністю. Важливим елементом є статус *Перевірено з генератором/інвертором*, що вказує на стійкість компонентів до нестабільної синусоїди напруги, яка часто зустрічається в польових умовах.

Блок енергетичної сумісності. Замість довгого опису використовується інтерактивна таблиця. Вона показує, скільки часу знадобиться для зарядки типових пристроїв, наприклад, iPhone 15, MacBook Air або павербанка на 20 000 mAh, саме від цього адаптера. Це дозволяє користувачу оцінити ефективність пристрою в межах обмеженого часу роботи графіків відключень.

Блок розумної специфікації портів є критичним блоком для багатопортових зарядок. Оскільки при підключенні другого пристрою потужність першого зазвичай падає, необхідно чітко прописати логіку розподілу. Наприклад, при використанні порта C1 + C2 потужність розподіляється як 65W + 35W, що дозволяє покупцю зрозуміти, чи вистачить залишкової енергії для живлення ноутбука.

У технічному паспорті безпеки вказується діапазон вхідної напруги, наприклад, 100V-240V. В умовах України це важливо, оскільки при просіданні напруги в мережі якісний блок живлення має продовжувати стабільно

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		14

працювати, не перегріваючись і не вимикаючись. Окремо виділяється інформація про вогнетривкість корпусу за стандартом UL94-V0.

Секція Готовність до блекауту є додатковим блоком, який рекомендує супутні товари для повної автономності. Якщо клієнт купує потужний блок PD, система автоматично пропонує сумісний кабель-тригер для роутера або кабель із вбудованим ватметром для контролю споживання енергії в реальному часі. Така структура дозволить мінімізувати кількість повернень та помилок при виборі.

Важливим елементом даного вебресурсу є робота Кошика замовлень. Логіка кошика та оформлення замовлення (Checkout) в умовах війни повинна враховувати два критичні фактори, а саме: нестабільний зв'язок у користувача та динамічну логістику, тобто зміни в роботі відділень через безпекову ситуацію.

В умовах блекаутів сесія може перерватися в будь-який момент, тому кошик має автоматично зберігатися не лише в сесії сервера, а й у локальній історії браузера користувача. Це гарантує, що при раптовому зникненні інтернету після відновлення зв'язку всі обрані позиції залишаться на місці, і користувачеві не доведеться шукати потрібний кабель чи адаптер заново.

Також має бути спрощений мобільний Checkout (One-tap), тобто, коли телефон заряджений на 5%, а інтернет ледве працює (Edge/2G), кожна зайва сторінка чи важкий скрипт - це ризик не встигнути замовити.

Мінімізація полів, тобто доцільно використовувати тільки ім'я, телефон та номер відділення. Автозаповнення, що відбувається шляхом інтеграції з Google Pay / Apple Pay та API Нової Пошти для миттєвого вибору міста та відділення без завантаження повного списку.

Відсутність обов'язкової реєстрації, бо створення акаунту має бути опціональним, щоб не створювати зайвих бар'єрів.

Розумна перевірка наявності та резервування, оскільки в пікові періоди після обстрілів енергосистеми попит на зарядки зростає в сотні разів за годину. Логіка кошика повинна підтримувати короткочасне резервування: як тільки

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		15

товар додано в кошик, він бронюється за користувачем на 10-15 хвилин. Це запобігає ситуації, коли товар закінчується прямо під час заповнення даних доставки.

Інтерактивна логістика та безпека, що забезпечується шляхом інтеграції з API служб доставки, таких як Нова Пошта, Meest тощо повинна в реальному часі фільтрувати відділення. Має бути відображення працюючих точок, тобто приховування відділень, що тимчасово зачинені через безпекову ситуацію або відсутність світла, якщо вони не мають генераторів. Доцільно надавати пріоритет на поштомати, оскільки для багатьох це найшвидший спосіб отримати замовлення без черг.

Доцільною є система сповіщень через месенджери, оскільки Email-сповіщення в умовах війни перевіряються рідше, логіка замовлення повинна включати автоматичну відправку статусу в Telegram/Viber/Signal. Це споживає менше трафіку та дозволяє клієнту отримати номер ТТН навіть при дуже слабкому сигналі мережі.

Процес замовлення має бути легким, тобто містити мінімум JS-коду та адаптованим під критично низьку швидкість передачі даних. Головне завдання - дати можливість людині оформити покупку швидко.

Проектування бази даних для сервісу з продажу зарядних пристроїв базується на реляційній моделі у середовищі PostgreSQL, що забезпечує цілісність та швидкий доступ до інформації. Основний масив даних розподіляється між декількома логічно пов'язаними таблицями, де центральне місце займає сутність товарів, яка містить ідентифікатор, найменування, технічні характеристики (потужність, тип роз'єму), ціну та актуальну кількість одиниць на складі. Окремо функціонує таблиця категорій, що дозволяє групувати зарядні пристрої за призначенням, наприклад, для мобільних телефонів або ноутбуків. Для забезпечення роботи кошика та процесу оформлення замовлень реалізується таблиця замовлень, яка фіксує дані покупця, загальну вартість та часову мітку транзакції, а зв'язок між конкретними товарами та замовленнями забезпечується через проміжну

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		16

таблицю елементів замовлення, де вказується кількість кожної позиції та ціна на момент купівлі. Така архітектура дозволяє зберігати історію покупок та коректно відображати вміст кошика користувача протягом поточної сесії.

Для перевірки працездатності та якості реалізованого вебсервісу застосовується комплексний підхід до тестування, який охоплює функціональні та нефункціональні параметри системи. Першочергово проводиться функціональне тестування механізму кошика, що включає перевірку коректності додавання товарів, динамічного перерахунку ціни при зміні кількості та валідацію полів форми оформлення замовлення.

Окремий етап присвячено тестуванню продуктивності за допомогою інструменту Google Lighthouse, де основними метриками є час завантаження найбільшого елемента контенту (LCP) та індекс швидкості (Speed Index), що повинні відповідати стандартам сучасного вебу.

Важливим аспектом є кросбраузерне та адаптивне тестування, яке гарантує ідентичне відображення та роботу кошика у браузерях Chrome, Safari та Firefox, а також на різних типах мобільних пристроїв. Безпека системи перевіряється через контроль коректності передачі даних по протоколу HTTPS та захищеність API-запитів від несанкціонованого доступу до персональних даних клієнтів. Завершальним етапом стає стрес-тестування, що визначає здатність сервера на базі Unix витримувати пікові навантаження при одночасній роботі великої кількості користувачів, забезпечуючи стабільну швидкість обробки запитів через Wi-Fi з'єднання.

1.2 Аналіз наявного програмно-технічного забезпечення предметної області

Для аналізу ринку вебресурсів з продажу зарядних пристроїв та портативних станцій в Україні було обрано 5 найбільших сайтів компаній: Rozetka, Comfy, Citrus, MoYo, Brain (рисунки 1.1-1.5). Аналіз базується на

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		17

актуальних даних за 2025-2026 роки, враховуючи зручність інтерфейсу, зокрема кошика, асортимент та відгуки користувачів.

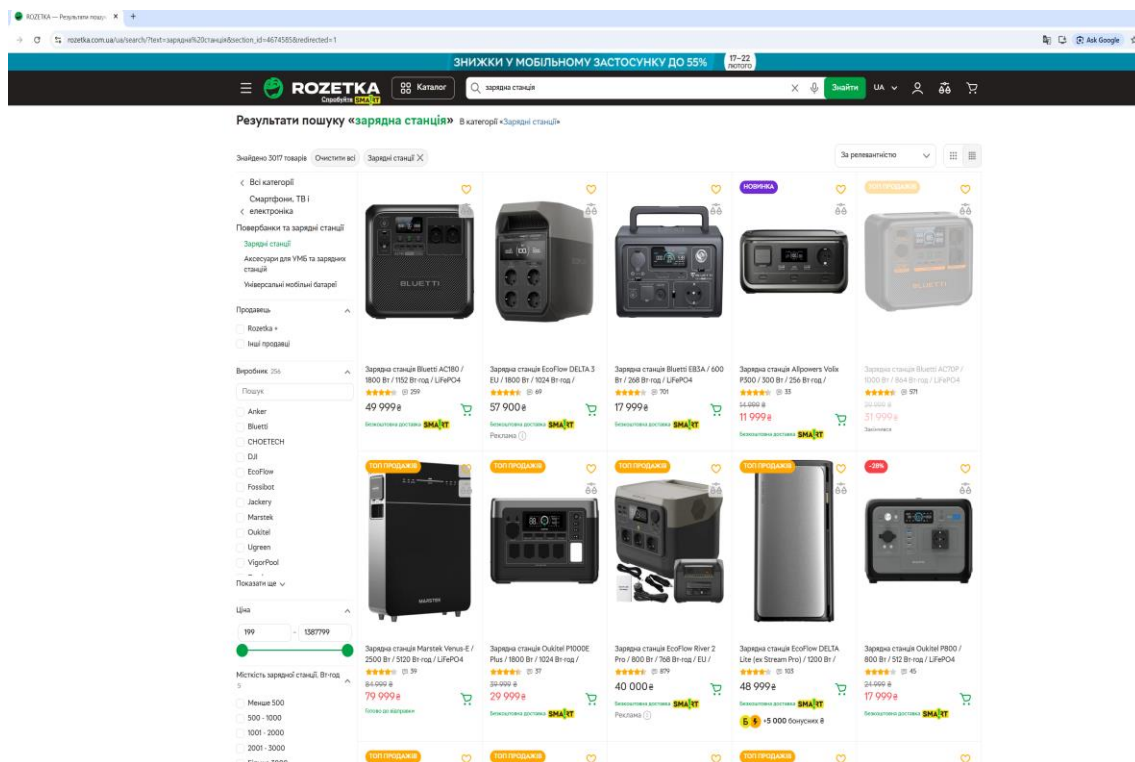


Рисунок 1.1 - Rozetka (rozetka.com.ua)

До плюсів ресурсу Rozetka можна віднести:

- найбільший асортимент;
- потужну систему фільтрів;
- велику кількість реальних відгуків з фото та відео;
- зручний кошик із списком бажань.

Серед мінусів можна виділити:

- велику кількість сторонніх продавців (маркетплейс), що може ускладнити гарантійне обслуговування;
- платну доставку у деяких випадках.

						Арк.
					КВРІПЗ.2201108.01.14.ПЗ	
Зм.	Арк	№ докум.	Підпис	Дата		18

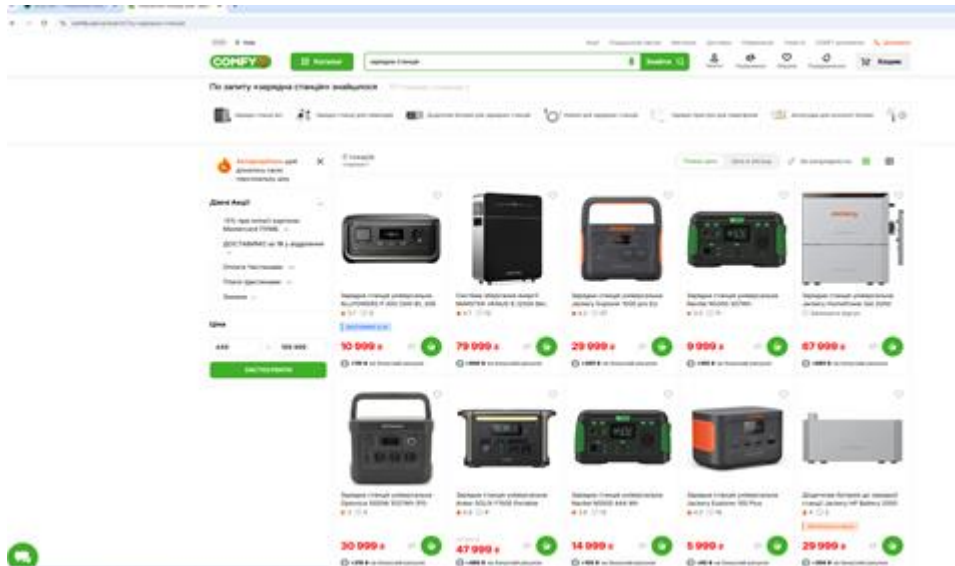


Рисунок 1.2 – Comfy (comfy.ua)

Наявність функції швидкого замовлення та наскрізної зарядки в описах; швидка обробка замовлень; зручний мобільний застосунок; програма лояльності (кешбек).

Недоліками є обмеженіший вибір специфічних професійних брендів порівняно з маркетплейсами; ціни на аксесуари часто вищі за середні.

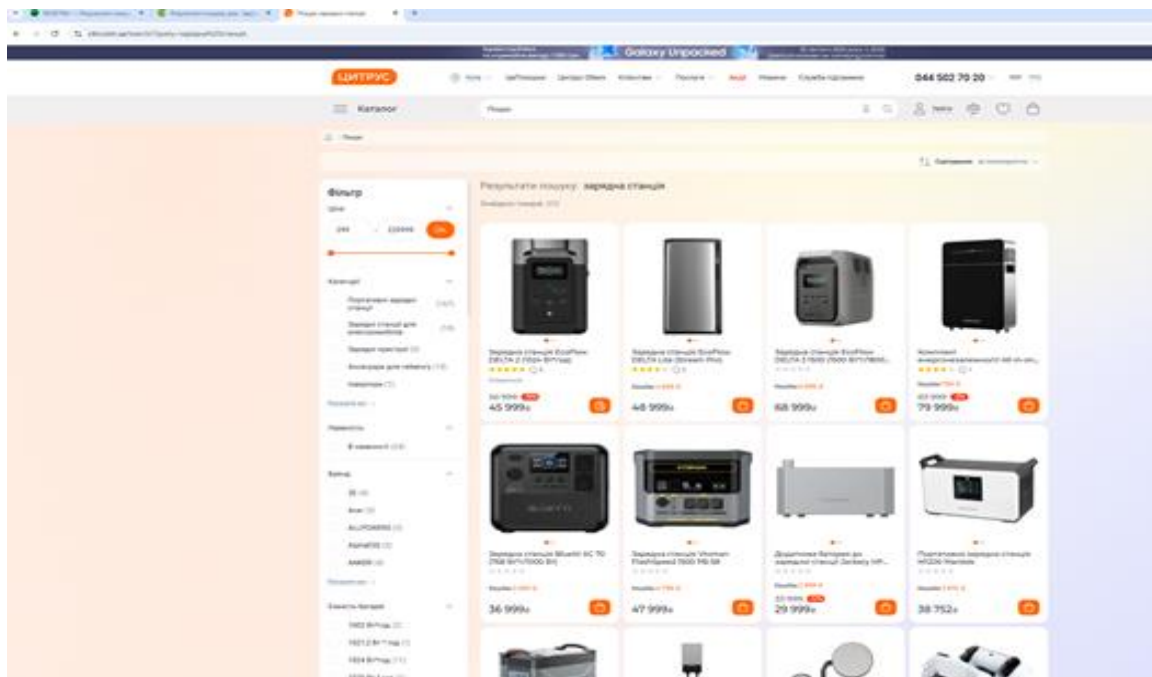


Рисунок 1.3 – Citrus (ctrs.com.ua)

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		19

Переваги ресурсу Citrus:

- якісний контент (відеоогляди);
- можливість оформлення розстрочки 0% онлайн;
- фокус на екосистемах (EcoFlow, Bluetti);
- сучасний UI/UX дизайн.

Недоліки ресурсу Citrus:

- скарги на роботу сервісного центру та повернення товару;
- агресивний маркетинг та нав'язування додаткових послуг.

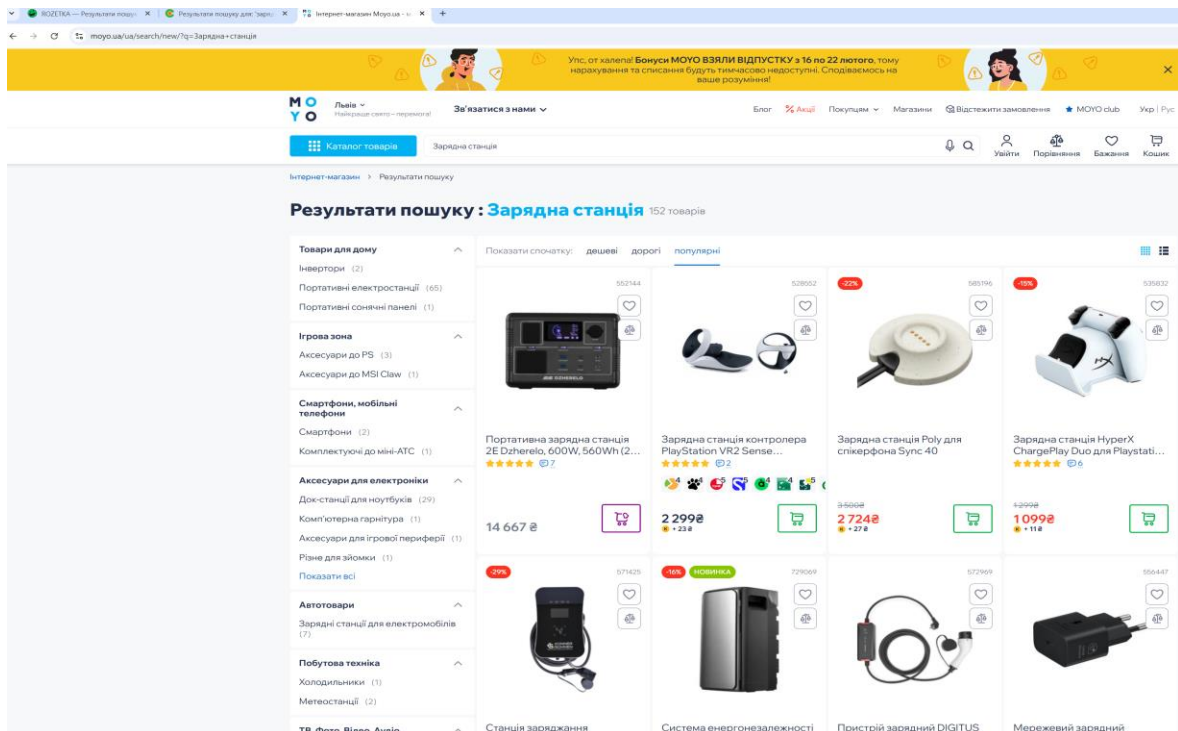


Рисунок 1.4 – Моюо (moюo.ua)

Переваги:

- цілодобова доставка у великих містах;
- функція «Моюо 24/7»;
- детальні технічні специфікації товарів;
- можливість обміну старої техніки (Trade-in).

Недоліки:

- менша кількість точок видачі у порівнянні з конкурентами;
- іноді складний процес підтвердження замовлення через кол-центр.

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		20

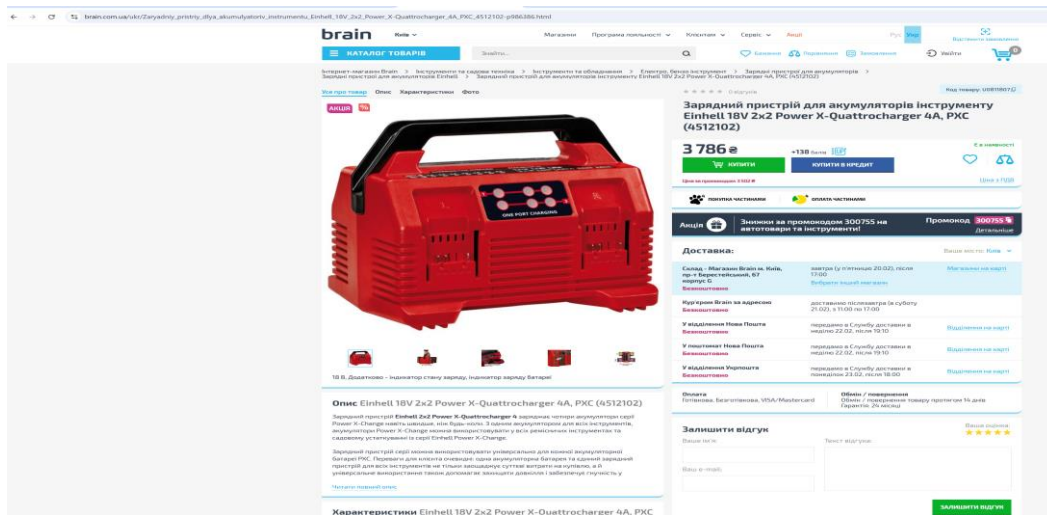


Рисунок 1.5 – Brain (brain.com.ua)

Переваги:

- експертний підхід до ІТ-техніки;
- наявність глибокої аналітики та рейтингів, наприклад, ТОП-10 зарядних пристроїв чи станцій;
- вигідні умови для корпоративних клієнтів.

Недоліки:

- сайт виглядає дещо перевантаженим інформацією;
- дизайн кошика та процесу Checkout є менш інтуїтивним, ніж у Citrus чи Comfy.

Порівняльний аналіз функціональності кошика дозволяє зробити висновки про те, що в межах розробки технічного завдання, варто звернути увагу на наступні особливості цих ресурсів, що описані далі.

Динамічне оновлення у Rozetka та Citrus, через те, що кошик реалізований як AJAX-елемент або бічна панель, що дозволяє продовжувати покупки без перезавантаження сторінки.

Збереження стану, бо всі ресурси використовують LocalStorage, тому товари залишаються в кошику навіть після закриття браузера.

						Арк.
						21
Зм.	Арк	№ докум.	Підпис	Дата	КВРІПЗ.2201108.01.14.ПЗ	

Cross-sell тобто супутні товари, Comfy та Моюо активно пропонують у кошику кабелі та перехідники, що підвищує середній чек - це варто врахувати у технічному завданні та під час безпосередньої розробки програмного продукту.

Аналіз ресурсу Rozetka показує складну мікросервісну архітектуру, яка здатна витримувати екстремальні навантаження під час пікових періодів. Позитивним аспектом є висока швидкість пошуку за великою базою даних завдяки використанню потужних пошукових рушіїв, проте мінусом є надмірна кількість завантажуваних JavaScript-скриптів, що негативно впливає на метрику Total Blocking Time. Платформа Comfy виділяється якісною реалізацією Single Page Application (SPA) елементів, що забезпечує миттєвий відгук кошика без перезавантаження сторінок. Технічним недоліком тут можна вважати іноді занадто складне DOM-дерево, що ускладнює рендеринг на слабких мобільних пристроях при використанні Wi-Fi з низькою пропускнуою здатністю.

Вебресурс Citrus демонструє передовий рівень Front-end інженерії з використанням сучасних анімацій та швидкого рендерингу медіаконтенту, що вказує на ефективну роботу мережі доставки контенту. З іншого боку, з точки зору розробника, сайт перевантажений трекінговими пікселями та сторонніми аналітичними сервісами, що може створювати проблеми з безпекою даних користувачів.

Моюо пропонує найбільш збалансований підхід до Server-Side Rendering (SSR), що забезпечує відмінну індексацію сторінок пошуковими роботами та швидкий перший байт. Проте архітектура їхнього Checkout-процесу виглядає застарілою порівняно з конкурентами, оскільки містить забагато послідовних синхронних запитів до API, що збільшує ризик помилок при нестабільному з'єднанні.

Сайт Brain побудований на більш консервативних технологіях, що робить його дуже стабільним і передбачуваним у роботі на різних операційних системах, включаючи Unix-подібні системи. Головною перевагою тут є чітка логіка фільтрації технічних параметрів, яка працює швидше за конкурентів

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		22

через меншу кількість візуальних надлишкових елементів. Основним технічним мінусом є відсутність сучасної оптимізації зображень, наприклад, формату WebP, що призводить до зайвого споживання трафіку. Для майбутнього проекту найкращою стратегією буде поєднання швидкості SSR від Moyo з легкістю SPA-елементів для кошика, як це реалізовано у Comfy, при цьому зберігаючи чистоту коду та мінімальну кількість сторонніх залежностей.

Отже, можна зробити висновок про те, що для розроблюваного вебсервісу рекомендується запозичити структуру фільтрів у Rozetka за потужністю, типом роз'ємів та швидкість Checkout-процесу у Comfy.

Оскільки доцільно працювати із Unix, то це дозволить реалізувати швидкий серверний рендеринг кошика, що забезпечить конкурентну перевагу у швидкодії порівняно з великими, але іноді «важкими» порталами.

Для забезпечення максимальної швидкодії системи на рівні бекенду необхідно використовувати архітектуру, орієнтовану на події, що дозволить мінімізувати час обробки запитів до бази даних PostgreSQL. Враховуючи роботу в Unix-середовищі, доцільно реалізувати кешування частих запитів, наприклад, списку категорій зарядних пристроїв за допомогою Redis, що розвантажить основну базу даних та зменшить час реакції сервера до мінімально можливих значень. При розробці фронтенду слід уникати помилок Rozetka та Citrus, пов'язаних із надмірним завантаженням JavaScript-бандлів; натомість рекомендується використовувати метод розділення коду, завантажуючи лише ті скрипти, які необхідні для поточної сторінки, що значно покращить показник Total Blocking Time.

Особливу увагу слід приділити логіці кошика, реалізувавши її як незалежний модуль на стороні клієнта з синхронізацією через API. Використання технології Optimistic UI дозволить миттєво відображати зміни в кількості товарів у кошику, не чекаючи відповіді від сервера, що створить відчуття безшовної роботи інтерфейсу. Для оптимізації передачі контенту через Wi-Fi з'єднання необхідно впровадити автоматичне стиснення зображень товарів у формат WebP безпосередньо на сервері Unix, а також використовувати

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		23

так зване ліниве завантаження для медіафайлів, які знаходяться поза межами видимої області екрана.

З точки зору безпеки, на відміну від багатьох комерційних платформ, рекомендується впровадити сувору політику безпеки контенту (Content Security Policy - CSP), щоб унеможливити виконання сторонніх шкідливих скриптів та XSS-атаки.

Для Checkout-процесу варто відмовитися від складних багатоступеневих форм на користь архітектури Single Page Checkout, де всі дані збираються в один асинхронний запит, що мінімізує ризик втрати даних при нестабільному інтернет-з'єднанні. Такий комплексний інженерний підхід дозволить створити легкий, швидкий та безпечний сервіс, що відповідає найвищим стандартам сучасної веброзробки.

1.3 Визначення функціональних та нефункціональних вимог

Аналіз вимог до платформи електронної комерції, що спеціалізується на зарядних пристроях, вимагає глибокого розуміння як технічних характеристик товару, так і психології покупця, який часто шукає розв'язання конкретної проблеми - відсутності живлення. Оскільки зарядний пристрій є технічно складним продуктом, центральним елементом аналізу стає система фільтрації та сумісності. Користувач не просто купує блок живлення, а він шукає відповідність вольтажу, силі струму та типу роз'єму, тому функціональні вимоги мають першочергово забезпечувати точність цих параметрів у каталозі.

Основним викликом при проектуванні є створення інтелектуального пошуку, коли користувачі часто не знають назви моделі зарядного, але знають модель свого гаджета. Отже, база даних має підтримувати зв'язки багато до багатьох, де один зарядний пристрій підходить до сотень моделей ноутбуків чи смартфонів. Опис товару має бути структурованим не як звичайний текст, а як набір атрибутів, що дозволяють системі автоматично пропонувати

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		24

альтернативи, якщо обраної позиції немає в наявності. Це мінімізує ризик відмови від покупки через технічну невпевненість клієнта.

Кошик замовлень - це не просто тимчасове сховище, а інструмент перевірки. Вимоги до цього вузла включають динамічний перерахунок вартості з урахуванням податків та вартості доставки, а також можливість збереження сесії для неавторизованих користувачів. Важливо реалізувати функцію крос-сейлу безпосередньо в кошику, наприклад, пропозиція придбати захисний кабель або перехідник. Процес оформлення замовлення має бути максимально лінійним, щоб покупець, який уже знайшов потрібний технічний елемент, не спіткнувся на надмірній кількості полів для заповнення.

Оскільки йдеться про фінансові транзакції, нефункціональні вимоги виходять на передній план. Саме тому вебресурс повинен витримувати пікові навантаження, забезпечувати шифрування персональних даних через SSL-протоколи та відповідати стандартам PCI DSS для обробки платежів. Швидкість завантаження сторінок є критичною, адже затримка за одну секунду може коштувати відсотка конверсії, особливо у мобільній версії, з якої часто здійснюються термінові покупки на ходу. Адміністративна панель своєю чергою повинна надавати зручний інструмент для масового оновлення цін та залишків, що важливо для динамічного ринку електроніки.

Для візуалізації ролі зареєстрованого користувача в системі VoltSpace найкраще підійдуть два типи UML-діаграм: діаграма прецедентів (Use Case), яка описує доступні дії, та діаграма класів (Class Diagram), яка показує структуру даних профілю та його зв'язки з іншими сутностями.

Діаграма прецедентів, яка ілюструє поведінкову модель користувача, наочно демонструє розподіл можливостей між загальним функціоналом магазину та закритим розділом особистого кабінету. Зареєстрований клієнт має доступ до перегляду історії, перевірки бонусного рахунку (Volt-кешбеку) та управління власними даними, тоді як процес оформлення замовлення обов'язково включає етап оплати.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		25

На рисунку 1.6 подано діаграму варіантів використання для зареєстрованого та незареєстрованого користувача

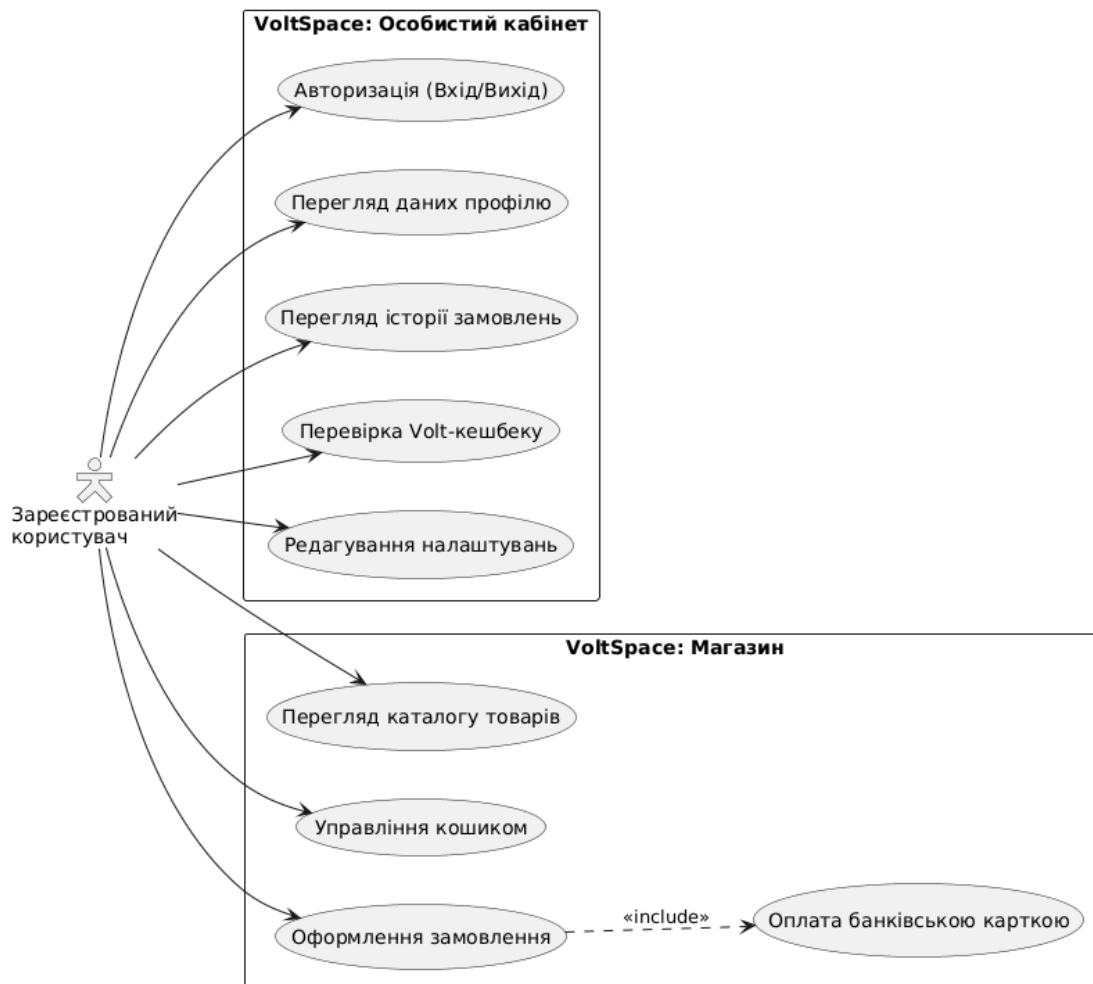


Рисунок 1.6 – Діаграма варіантів використання для зареєстрованого користувача

Для технічного розуміння архітектури проєкту також необхідна діаграма класів. Вона відображає внутрішню структуру даних, яку доцільно реалізувати в React-стані. Цей код показує атрибути самого користувача, такі як ім'я, електронна пошта та кількість накопичених бонусів. Крім того, діаграма фіксує структурні зв'язки: один користувач може мати безліч замовлень у своїй історії, а кожне замовлення, як і поточний кошик, складається з окремих товарних позицій (рисунок 1.7).

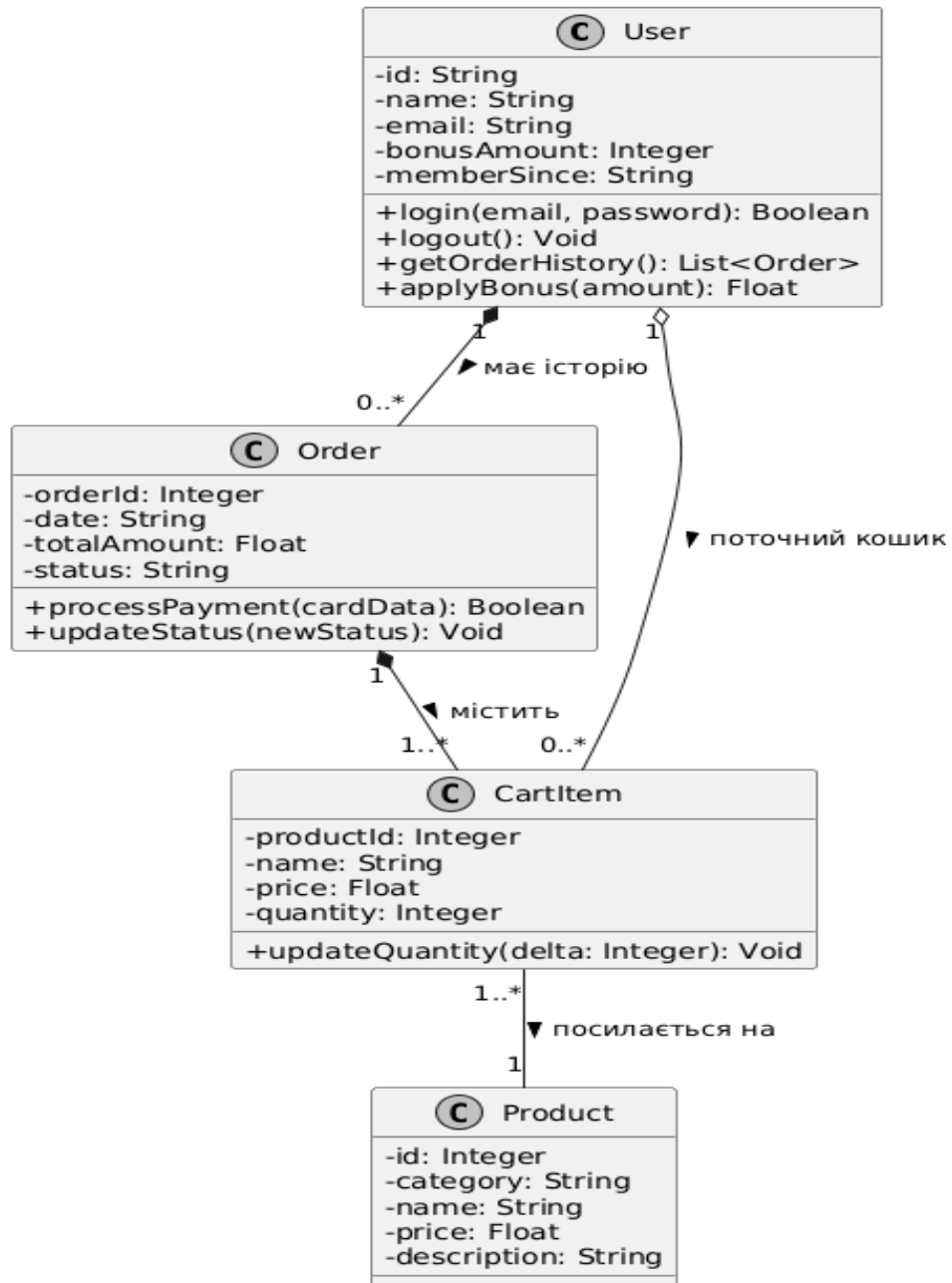


Рисунок 1.7 – Діаграма класів

Отже, діаграми варіантів використання, класів тощо дозволяють продемонструвати наочно як буде виглядати програмний продукт та його бізнес-структуру.

1.4. Висновки. Постановка задачі

Отже, для виконання кваліфікаційної роботи було визначено мету та задачі. Мета кваліфікаційної роботи – це розробка вебресурсу для електронної торгівлі зарядними пристроями із кошиком замовлень.

В ході виконання кваліфікаційної роботи повинні бути вирішені такі завдання:

- провести аналітичний огляд предметної області;
- провести аналіз вже існуючих вебзастосунків, що допомагають здійснити торгівлю зарядними пристроями;
- визначити переваги та недоліки проаналізованих бізнес-рішень;
- здійснити постановку задачі та розробити технічне завдання;
- здійснити вибір та опис архітектури розроблюваного програмного продукту;
- провести аналіз та вибір технологій для розробки програмного продукту;
- реалізувати та протестувати розроблений програмний продукт.

Також в ході роботи над першим розділом було встановлено, що платформи електронної комерції, які спеціалізуються на зарядних пристроях, вимагають чіткого розуміння як технічних характеристик товару, так і психології покупця, який часто шукає розв'язання конкретної проблеми - відсутності живлення.

База даних має підтримувати зв'язки багато до багатьох, де один зарядний пристрій підходить до сотень моделей ноутбуків чи смартфонів. Опис товару має бути структурованим не як звичайний текст, а як набір атрибутів, що дозволяють системі автоматично пропонувати альтернативи, якщо обраної позиції немає в наявності. Це мінімізує ризик відмови від покупки через технічну невпевненість клієнта.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
						28
Зм.	Арк	№ докум.	Підпис	Дата		

Через те, що у даному ресурсі будуть здійснюватись фінансові транзакції, то визначення нефункціональних вимог відіграє ключову роль. Через це вебресурс повинен витримувати пікові навантаження, забезпечувати шифрування персональних даних через SSL-протоколи та відповідати стандартам PCI DSS для обробки платежів. Швидкість завантаження сторінок є критичною, адже затримка за одну секунду може коштувати відсотка конверсії, особливо у мобільній версії, з якої часто здійснюються термінові покупки на ходу.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		29

2 ПРОЕКТУВАННЯ ВЕБЗАСТОСУНКУ

2.1 Проектування архітектури та структури вебзастосунку

Проектування архітектури даного рішення полягає у тому, щоб визначити яким чином необхідно працювати під час розробки програмного продукту. Архітектура платформи VoltSpace базується на класичній багаторівневій клієнт-серверній моделі з чітким розділенням обов'язків (Separation of Concerns). Такий модульний підхід дозволяє системі бути стійкою до високих навантажень, забезпечує безпеку фінансових операцій та створює фундамент для безболісного масштабування функціоналу в майбутньому.

Взаємодія користувача з платформою відбувається через Single Page Application (SPA), що розроблений на базі бібліотеки React. Цей рівень ізолювано відповідає за рендеринг інтерфейсу, управління локальним станом (відображення модальних вікон, наповнення кошика) та взаємодію з користувачем. Замість традиційного підходу з повним перезавантаженням HTML-сторінок при кожній дії, клієнтський застосунок працює асинхронно. Він відправляє легкі HTTP-запити до сервера та отримує у відповідь виключно необхідні дані у форматі JSON. Це суттєво зменшує використання інтернет-трафіку та забезпечує плавність роботи, що є критично важливим для користувачів із нестабільним мобільним зв'язком під час блекаутів.

Центральним вузлом обробки даних виступає RESTful API сервер, побудований на платформі Node.js з використанням фреймворку Express. Цей рівень діє як надійний бар'єр та координатор між відкритою клієнтською частиною та закритою базою даних. Він приймає мережеві запити, перевіряє криптографічні токени авторизації, валідує вхідні дані та виконує бізнес-правила платформи. Наприклад, саме тут перевіряється, чи достатньо у користувача бонусів для знижки, або чи коректно розрахована загальна сума кошика. Жоден клієнтський пристрій не має прямого доступу до сховища

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		30

даних, тобто уся комунікація відбувається виключно через стандартизовані маршрути цього сервера.

Для безпечної та передбачуваної взаємодії серверної логіки з базою даних імплементовано технологію об'єктно-реляційного відображення Prisma. Вона виконує роль інтелектуального перекладача, трансформуючи зрозумілі JavaScript-методи у складні SQL-запити. Використання цього прошарку автоматично нівелює ризики найпоширеніших кібератак, зокрема SQL-ін'єкцій, оскільки всі користувацькі дані проходять суворе екранування. Важливою архітектурною перевагою Prisma є підтримка транзакцій на рівні коду. Завдяки цьому процес оформлення замовлення, а саме створення запису чека та збереження історії придбаних товарів відбувається як єдина неподільна операція, що гарантує фінансову цілісність.

Фундаментом для збереження всієї інформації обрано потужну реляційну систему PostgreSQL. Ця система забезпечує суворе дотримання принципів ACID, тобто атомарність, узгодженість, ізолюваність, довговічність, що є обов'язковим стандартом для систем електронної комерції. Завдяки жорстко визначеній схемі таблиць та системі зовнішніх ключів, база даних на фізичному рівні унеможлиблює створення замовлень для неіснуючих користувачів або втрату історії чеків при видаленні товарів із каталогу.

Впровадження такої децентралізованої багаторівневої архітектури дозволяє команді розробників розвивати кожен компонент незалежно. Наприклад, у разі пікових навантажень можна легко збільшити обчислювальні потужності серверного рівня, наприклад, за допомогою Node.js без необхідності зупиняти або модифікувати базу даних, що забезпечує безперебійну роботу ресурсу в критичні моменти.

Наведена на рисунку 2.1 діаграма компонентів ілюструє багаторівневу клієнт-серверну архітектуру платформи. Кожен вузол системи є ізолюваним і виконує виключно свою функцію, спілкуючись із сусідніми рівнями через чітко визначені протоколи HTTP для мережі та SQL для баз даних.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		31

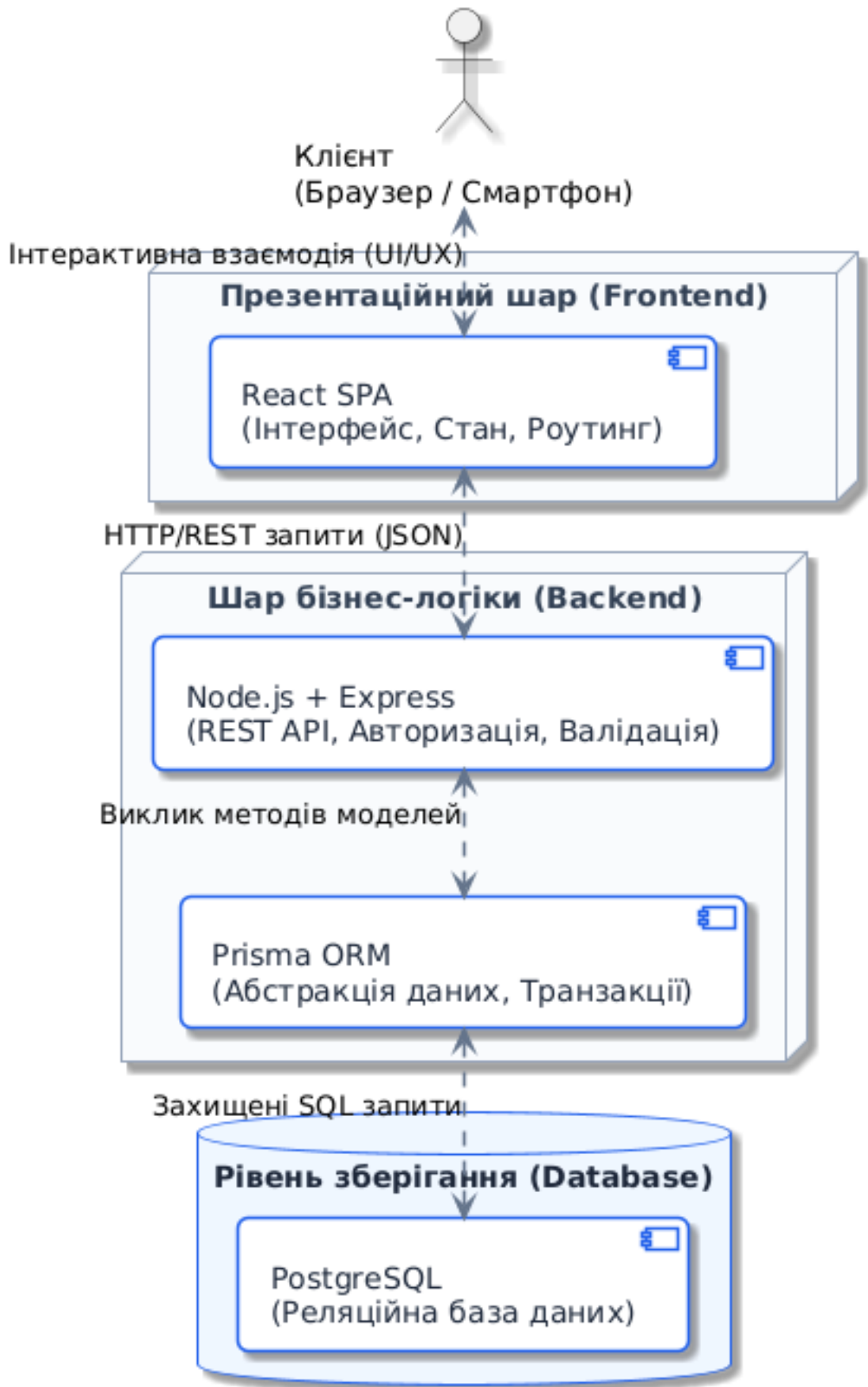


Рисунок 2.1 – Архітектура застосунку

Дана схема переводить складний текстовий опис архітектури у зрозумілий візуальний формат. Вона демонструє, що система побудована за сучасними стандартами розробки з повним розділенням відповідальності між фронтендом та бекендом.

2.2 Проектування логічної моделі бази даних

База даних відіграє суттєву роль під час розробки та проектування будь-якого програмного продукту. На рисунку 2.2 продемонстровано структуру бази даних, що входить до бізнес-логіки даного програмного застосунку. Для детального опису сутностей та розуміння структури бази даних, необхідно описати таблиці, що будуть входити у розроблювану базу даних. Перехід від імітації даних до повноцінної реляційної бази - це фундаментальний етап для масштабування та безпеки проєкту.

Нижче показано логічну архітектуру бази даних та готові SQL-скрипти для її швидкого розгортання в середовищі PostgreSQL. Архітектура спроектована таким чином, щоб зберегти цілісність фінансових даних та історію покупок навіть у випадку зміни цін на товари у майбутньому.

Логічна схема бази даних та її структура складається з чотирьох основних сутностей, пов'язаних між собою реляційними зв'язками. Центральними таблицями є користувачі та товари, які існують незалежно. Їх об'єднує таблиця замовлень, а деталізація кожного чека зберігається у спеціальній транзитній таблиці елементів замовлення.

У таблиці “Користувачі” зберігається персональна інформація та баланс віртуального рахунку. Використання унікального ідентифікатора та перевірки на унікальність електронної пошти гарантує відсутність дублікатів акаунтів.

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		33

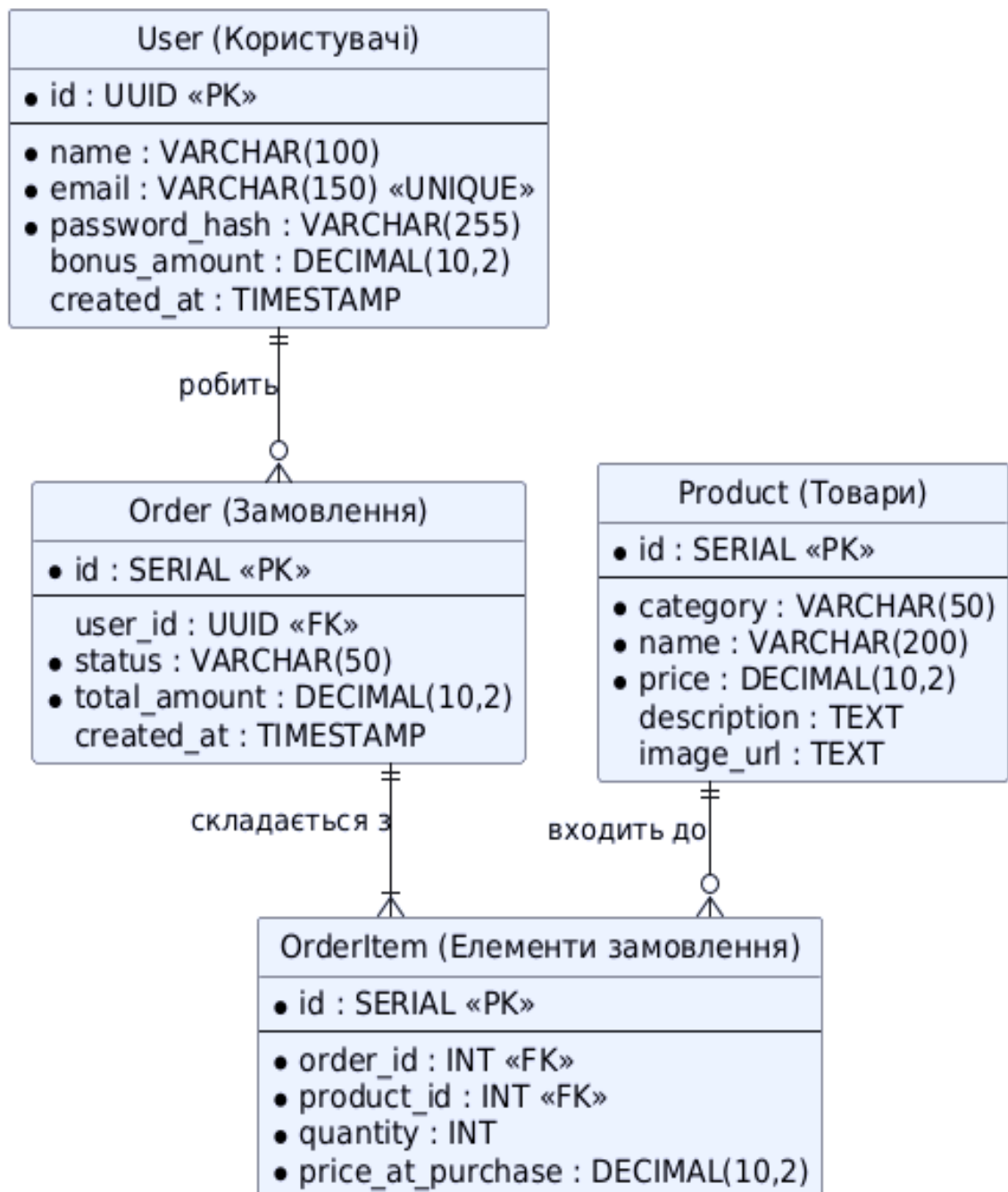


Рисунок 2.2 – Структурна схема бази даних

Таблиця 2.1 - Користувачі

Поле	Тип даних	Обмеження	Опис
id	UUID	PRIMARY KEY	Унікальний ідентифікатор клієнта
name	VARCHAR(100)	NOT NULL	Ім'я користувача
email	VARCHAR(150)	UNIQUE, NOT NULL	Електронна пошта (логін)
password_hash	VARCHAR(255)	NOT NULL	Зашифрований пароль
bonus_amount	DECIMAL(10,2)	DEFAULT 0.00	Кількість Volt-кешбеку
created_at	TIMESTAMP	DEFAULT NOW()	Дата реєстрації

Таблиця Товари містить каталог енергетичних рішень. Ціна зберігається у форматі DECIMAL для уникнення проблем з округленням фінансових значень, які часто виникають при використанні звичайних чисел із рухомою комою.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		35

Таблиця 2.2 -Товари

Поле	Тип даних	Обмеження	Опис
id	SERIAL	PRIMARY KEY	Автоматичний номер товару
category	VARCHAR(50)	NOT NULL	Категорія, наприклад, павербанки
name	VARCHAR(200)	NOT NULL	Назва пристрою
price	DECIMAL(10,2)	NOT NULL	Актуальна вартість
description	TEXT	NOT NULL	Опис характеристик
image_url	TEXT	NOT NULL	Посилання на зображення

Таблиця Замовлення фіксує факт здійснення покупки конкретним користувачем. Загальна сума зберігається окремо для швидкого відображення в особистому кабінеті без необхідності щоразу перераховувати всі товари.

Таблиця 2.3 - Замовлення

Поле	Тип даних	Обмеження	Опис
id	SERIAL	PRIMARY KEY	Номер замовлення
user_id	UUID	FOREIGN KEY	Посилання на покупця

Продовження таблиці 2.3:

status	VARCHAR(50)	DEFAULT 'Оплачено'	Поточний стан (Оплачено, Доставляється)
total_amount	DECIMAL(10,2)	NOT NULL	Загальна сума чека
created_at	TIMESTAMP	DEFAULT NOW()	Час оформлення

Таблиця Елементи замовлення є критично важливою таблицею для електронної комерції. Вона фіксує, за якою саме ціною був проданий товар у момент оформлення замовлення. Це гарантує, що зміна ціни в каталозі products не змінить суму вже закритих історичних чеків у профілі користувача.

Таблиця 2.4 - Елементи замовлення

Поле	Тип даних	Обмеження	Опис
id	SERIAL	PRIMARY KEY	Ідентифікатор запису
order_id	INT	FOREIGN KEY	Посилання на замовлення
product_id	INT	FOREIGN KEY	Посилання на товар
quantity	INT	NOT NULL	Кількість одиниць
price_at_purchase	DECIMAL(10,2)	NOT NULL	Зафіксована ціна на момент покупки

2.3 Проектування інтерфейсу користувача

Проектування користувацького інтерфейсу для платформи VoltSpace базується на принципах функціонального мінімалізму та максимальної доступності. Враховуючи специфіку товарів - систем автономного живлення, які часто купуються в умовах стресу або обмеженого часу під час відключень електроенергії - інтерфейс має забезпечувати найшвидший шлях від пошуку до оформлення замовлення. Головною концепцією є Mobile-First, що гарантує ідеальне відображення каталогу та кошика на екранах смартфонів, звідки надходить лєвова частка трафіку під час блекаутів. Усі інтерактивні елементи, такі як кнопки та поля введення, мають збільшену площу для зручного натискання пальцем на сенсорних дисплеях.

Візуальна мова проєкту побудована навколо довіри та технологічності. Основна колірна палітра використовує глибокі відтінки синього, що символізують енергію, стабільність та безпеку, у поєднанні з нейтральними світло-сірими та білими фонами для створення ефекту чистого простору. Це дозволяє зняти візуальне навантаження з користувача та сфокусувати його увагу виключно на фотографіях товарів і ключових характеристиках. Типографіка спирається на сучасні рубані шрифти без зарубок із високим рівнем контрастності. Таке рішення робить текст добре читабельним навіть при мінімальній яскравості екрана смартфона, коли користувач намагається економити заряд батареї.

Структурно інтерфейс поділений на кілька взаємопов'язаних зон, які працюють без перезавантаження сторінки завдяки React-архітектурі. Верхня панель навігації є липкою, тобто вона залишається на екрані під час скролінгу, надаючи постійний доступ до категорій, рядка пошуку та стану кошика. Головний екран зустрічає користувача чіткою ціннісною пропозицією та бейджами довіри, після чого одразу розгортається інтерактивний каталог. Картки товарів спроектовані з акцентом на високоякісне зображення, виділену ціну та контрастну кнопку швидкого додавання. При взаємодії з елементами

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		38

активуються плавні CSS-мікроанімації, які дають миттєвий зворотний зв'язок про успішну дію, наприклад, зміну лічильника на іконці кошика.

Окремої уваги заслуговує проектування користувацького досвіду (UX) у процесі покупки та авторизації. Замість переходу на громіздкі окремі сторінки, кошик та особистий кабінет реалізовані у вигляді бічних панелей, які плавно виїжджають поверх основного контенту з легким ефектом розмиття фону. Це зберігає контекст каталогу і дозволяє покупцеві будь-якої миті повернутися до вибору товарів одним кліком. Сам процес оформлення замовлення зведений до мінімальної кількості етапів із використанням інтуїтивних форм введення без зайвих полів, що суттєво зменшує когнітивне навантаження та безпосередньо підвищує загальну конверсію платформи.

Нижче наведено структурні схеми (wireframes) ключових екранів вебресурсу, куди входить:

- головна сторінка, на якій розміщується Каталог;
- бічна панель, куди входить Кошик та Оформлення;

Головна сторінка, на якій розміщується Каталог є екраном, що фактично відображає сутність та візуал платформи. Він розділений на навігаційну панель, яка фіксується при прокрутці, зону привернення уваги або Hero Section та безпосередньо сітку каталогу з фільтрами. На рисунку 2.3 продемонстровано як орієнтовно може виглядати головна сторінка вебресурсу.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		39



Рисунок 2.3 – Вигляд головної сторінки

Модуль бічної панелі, куди входять елементи Кошик та Оформлення є важливою складовою даного вебресурсу. Кошик не відкривається на новій сторінці, а плавно виїжджає з правого краю екрана, затемнюючи основний контент. Це дозволяє користувачу не втрачати контекст покупок (рисунок 2.4).

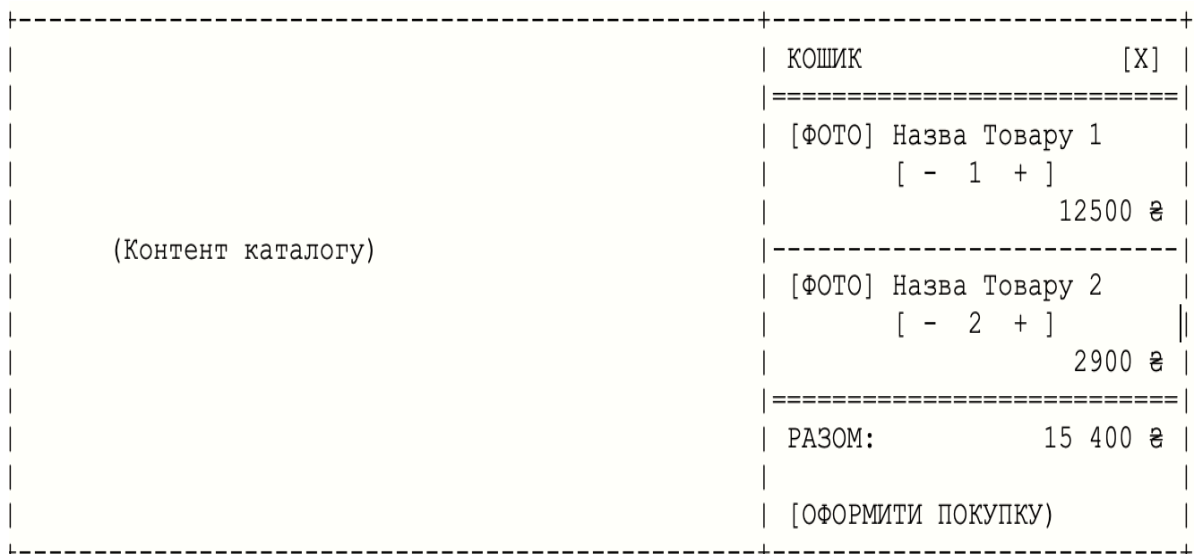


Рисунок 2.4 – Макет бічної панелі

Не менш важливою є бічна панель, на якій розгортається Особистий кабінет. Аналогічно до кошика, профіль розгортається праворуч. Тут зібрана ключова інформація про користувача, його фінансова вигода (бонуси) та історія взаємодії з магазином (рисунок 2.5).

(Контент каталогу)	МІЙ КАБІНЕТ	[X]
	=====	
	(А) Олександр	
	alex@example.com	

	[Кешбек]	[Замовлень]
	450 ₴	5

	ОСТАННІ ЗАМОВЛЕННЯ:	
	№123 від 20.03 [Оплачено]	
2 тов.	15 400 ₴	
=====		
[Налаштування профілю]		
[Вийти з акаунту]		

Рисунок 2.5 – Макет бічної панелі Особистий кабінет

Для вікна Авторизації форма входу з'являється по центру екрана. Вона максимально спрощена, щоб не створювати бар'єрів для користувача (рисунок 2.6).

(Затемнений контент каталогу)	З ПОВЕРНЕННЯМ!	[X]
	=====	
	[Email]	
	[Пароль]	
	[УВІЙТИ >]	

	Немає акаунту?	
	[Зареєструватися]	

Рисунок 2.6 – Макет вікна авторизації

Сторінка товару – це детальна сторінка конкретного пристрою. Дизайн акцентує увагу на фотографії, ключових характеристиках та чіткому заклику до дії, а саме додавання в кошик (рисунок 2.7).

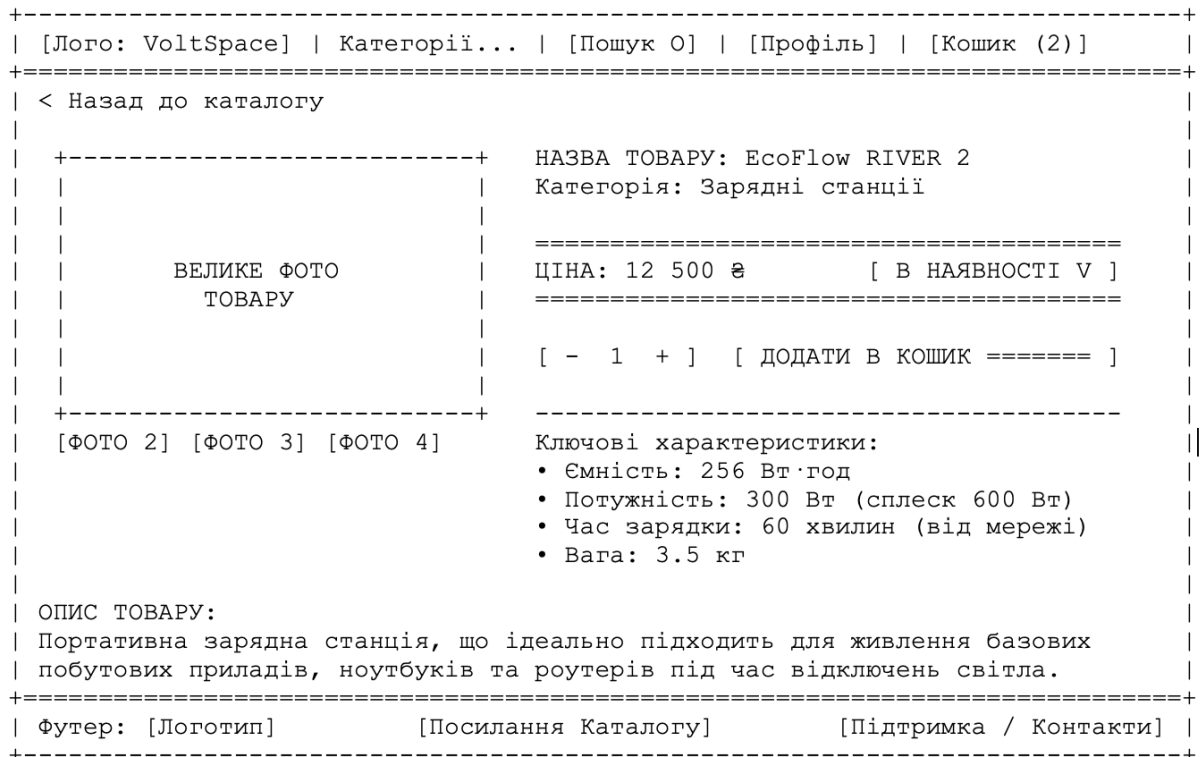


Рисунок 2.7 – Макет сторінки товару

2.4 Аналіз та вибір технологій і методів реалізації вебсервісу

Технологічний стек проєкту VoltSpace включає:

- основний фреймворк та логіку;
- мову програмування;
- стилізація та дизайн;
- іконографіку;
- базу даних та архітектуру збереження.

Вебресурс VoltSpace побудований як сучасний односторінковий додаток (Single Page Application). Архітектура проєкту орієнтована на високу швидкість

роботи, швидку розробку інтерфейсу та зручну взаємодію з користувачем без необхідності постійного перезавантаження сторінок.

В основі проєкту лежить бібліотека React, яка забезпечує створення динамічних користувацьких інтерфейсів. React використовує концепцію віртуального DOM (Virtual DOM), що дозволяє оптимізувати рендеринг. Тобто замість повного оновлення сторінки фреймворк обчислює мінімально необхідні зміни та точково застосовує їх до реального дерева елементів браузера. Завдяки цьому інтерфейс оновлюється миттєво навіть при складних змінах стану, наприклад, під час миттєвої фільтрації каталогу товарів або перерахунку вартості кошика. Односпрямований потік даних гарантує передбачуваність роботи додатку, оскільки зміни завжди передаються від батьківських компонентів до дочірніх. Для управління станом застосунку (відкриття модальних вікон, авторизація, збереження історії замовлень) активно використовуються React Hooks, зокрема useState. Щоб забезпечити високу продуктивність під час пошуку, застосовано хук useMemo, який запам'ятовує результати важких обчислень і запобігає зайвим перемальовкам інтерфейсу.

Проєкт повністю написано на сучасному JavaScript (ES6+) з використанням розширення синтаксису JSX. Застосування сучасного стандарту мови відкриває доступ до таких потужних інструментів, як деструктуризація об'єктів, стрілкові функції, константи та шаблонні рядки. Це значно скорочує обсяг шаблонного коду і підвищує його читабельність для інших розробників. Використання JSX дозволяє безшовно поєднувати декларативну HTML-розмітку з логікою JavaScript безпосередньо в одному файлі компонента. Складна бізнес-логіка, зокрема робота з кошиком, а саме, додавання нових позицій, видалення, підсумовування вартості, елегантно реалізована за допомогою сучасних методів трансформації масивів, таких як map, filter та reduce. Крім того, архітектура мови дозволяє легко імітувати асинхронні процеси, що критично важливо для обробки подій оплати та авторизації.

Для створення візуальної частини використовується Tailwind CSS - utility-first фреймворк. Замість написання окремих CSS-файлів, стилі застосовуються

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		43

безпосередньо в розмітці через класи. Це дозволило швидко реалізувати адаптивний дизайн, який коректно відображається на мобільних телефонах, планшетах та десктопах (завдяки префіксам sm:, md:, lg:). Tailwind також забезпечує плавні анімації, ефекти наведення (hover), розмиття фону (backdrop-blur) та мікроінтеракції кнопок без написання складного CSS-коду.

Усі іконки на сайті кошик, профіль, пошук, категорії товарів імпортуються з бібліотеки Lucide React. Вона надає доступ до легкої та стилістично однорідної колекції векторних іконок, які легко масштабуються та змінюють колір за допомогою класів Tailwind.

На етапі створення життєздатного продукту (MVP) проєкт використовує архітектуру Mock Data, тобто імітацію бази даних у пам'яті клієнта. Уся інформація про каталог товарів, категорії, сесії користувачів та історію їхніх замовлень зберігається у вигляді чітко структурованих JSON-об'єктів. Такий підхід дозволяє миттєво тестувати бізнес-гіпотези та поведінку інтерфейсу без витрат часу на розгортання та налаштування серверної інфраструктури. Проте структура цих імпровізованих даних спроектована за всіма правилами реляційних систем, коли кожна сутність, а саме товар, замовлення, користувач вже має унікальні ідентифікатори та визначені типи полів. Це означає, що у майбутньому перехід на повноцінну хмарну базу даних, таку як PostgreSQL для жорсткої структури або Firebase/MongoDB для гнучкої документо-орієнтованої моделі, пройде максимально безшовно для клієнтської частини додатку, потребуючи лише заміни локальних функцій на реальні API-запити.

Висновки до 2-го розділу

Отже, у другому розділі представлено проектування розроблюваного програмного забезпечення для здійснення покупок зарядних пристроїв, систем тощо. Продемонстровано вигляд мокапів даного застосунку по модулях, описано предметну область шляхом подання всіх таблиць, що входять у базу даних.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		44

Також обрано технологічний стек для даного проєкту VoltSpace, що включає: основний фреймворк та логіку, мову програмування, стилізація та дизайн, іконографіку, базу даних та архітектуру збереження.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		45

3 ПРОГРАМНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

3.1 Реалізація бази даних

Реалізація бази даних відіграє ключову роль у створенні програмного продукту. Для цього доцільно продемонструвати створені SQL-запити для ініціалізації бази даних.

Поданий нижче лістинг коду, створює всі необхідні таблиці у правильній послідовності, щоб уникнути конфліктів зовнішніх ключів, встановлює зв'язки та додає тестові дані для перевірки працездатності.

Активація розширення для генерації UUID

```
CREATE EXTENSION IF NOT EXISTS "uuid-ossf";
```

1. Створення таблиці користувачів

```
CREATE TABLE users (  
    id UUID PRIMARY KEY DEFAULT uuid_generate_v4(),  
    name VARCHAR(100) NOT NULL,  
    email VARCHAR(150) UNIQUE NOT NULL,  
    password_hash VARCHAR(255) NOT NULL,  
    bonus_amount DECIMAL(10,2) DEFAULT 0.00,  
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

2. Створення таблиці товарів

```
CREATE TABLE products (  
    id SERIAL PRIMARY KEY,  
    category VARCHAR(50) NOT NULL,  
    name VARCHAR(200) NOT NULL,  
    price DECIMAL(10,2) NOT NULL,  
    description TEXT,
```

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		46

```
image_url TEXT
);
```

3. Створення таблиці замовлень

```
CREATE TABLE orders (
    id SERIAL PRIMARY KEY,
    user_id UUID REFERENCES users(id) ON DELETE SET NULL,
    status VARCHAR(50) DEFAULT 'Оплачено',
    total_amount DECIMAL(10,2) NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

Створення таблиці елементів замовлення

```
CREATE TABLE order_items (
    id SERIAL PRIMARY KEY,
    order_id INT REFERENCES orders(id) ON DELETE CASCADE,
    product_id INT REFERENCES products(id) ON DELETE RESTRICT,
    quantity INT NOT NULL CHECK (quantity > 0),
    price_at_purchase DECIMAL(10,2) NOT NULL
);
```

Додавання тестових даних включає в себе додавання тестового товару.

Код для таких тестових наборів подано нижче.

```
INSERT INTO products (category, name, price, description, image_url)
VALUES
```

('Зарядні станції', 'EcoFlow RIVER 2', 12500.00, 'Портативна зарядна станція ємністю 256 Вт-год.', 'https://example.com/ecoflow.jpg'),

('Для смартфонів', 'GaN Charger 65W Pro', 1450.00, 'Трьохпортовий зарядний пристрій.', 'https://example.com/charger.jpg');

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		47

Додавання тестового користувача із імітованим паролем.

```
INSERT INTO users (name, email, password_hash, bonus_amount)
VALUES ('Олександр', 'alex@example.com', 'hashed_password_123',
450.00);
```

При створенні замовлення використовується email для знаходження UUID користувача, що виглядає наступним чином:

```
INSERT INTO orders (user_id, status, total_amount)
VALUES ((SELECT id FROM users WHERE email = 'alex@example.com'),
'Оплачено', 13950.00);
```

Для додавання товарів до створеного замовлення доцільно прописати наступний код:

```
INSERT INTO order_items (order_id, product_id, quantity,
price_at_purchase)
VALUES
(1, 1, 1, 12500.00), -- EcoFlow (1 шт)
(1, 2, 1, 1450.00); -- Зарядний пристрій (1 шт)
```

3.2 Реалізація інтерфейсу та інструкція користувача

Розроблюваний проєкт включає:

- інтуїтивний каталог з категоріями, наприклад, для смартфонів, ноутбуків, павербанки;
- динамічний кошик, який працює в режимі реального часу;

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		48

- адаптивний дизайн для зручного перегляду на мобільних телефонах та комп'ютерах;
- сучасний інтерфейс з використанням Tailwind CSS для професійного вигляду.

Було створено прототип магазину зі наступними можливостями:

- каталог товарів, у якому представлено різні типи зарядних пристроїв та аксесуарів;
- пошук та фільтрація, де покупець може шукати товари за назвою або фільтрувати за категоріями;
- повноцінний кошик, що дозволяє здійснювати додавання, видалення та зміну кількості товарів з автоматичним перерахунком суми;
- емуляцію оформлення замовлення, тобто при кліку на кнопку замовлення з'являється анімація завантаження та повідомлення про успіх;
- сучасний дизайн, де використано чистий стиль з акцентом на контент та зручну навігацію.

Функціонал магазину може бути розширено за рахунок додавання розширеного асортименту, шляхом додавання різних категорій, наприклад:

- зарядні станції (EcoFlow тощо) та більше моделей кабелів і адаптерів, а також сонячні панелі для повної енергонезалежності;
- системи користувачів, де реалізовано модальні вікна для реєстрації та авторизації з валідацією;
- можливість онлайн-оплата, де інтегровано форму введення даних банківської картки при оформленні замовлення;
- управління станом, коди додаток відстежує стан авторизації користувача;
- можливість реєстрації та авторизації.

Щоб відкрити модальне вікно входу необхідно натиснути на іконку користувача у верхньому меню,. Після входу інтерфейс змінюється, відображаючи ім'я користувача.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		49

Для здійснення онлайн-оплати необхідно здійснити ряд дій. Спочатку потрібно обрати та додати товар у кошик. Після цього натиснути кнопку Продовжити. Далі з'явиться стилізована форма оплати банківською карткою. У даному застосунку реалізовано імітацію обробки транзакції з анімацією завантаження та фінальним підтвердженням.

Покращений UI, що має преміальний вигляд з використанням заокруглених кутів (3xl), блюр-ефектів та плавних анімацій переходу. Також у даному ресурсі є реалізація особистого кабінету користувача, що доступний після авторизації (іконка профілю). Там можна переглянути історію замовлень, налаштування та персональні бонуси, тобто кожне успішне замовлення тепер зберігається в профілі користувача. Також є система бонусів через додавання віртуального рахунку Volt-кешбек. Є управління профілем через можливість виходу з акаунту та редагування даних.

Щоб протестувати особистий кабінет необхідно натиснути на іконку входу (чоловічок) у верхньому меню та ввести будь-які дані. Після авторизації потрібно ввести своє ім'я, після чого відкриється панель профілю. Далі користувач повинен зробити замовлення шляхом додавання товару у кошик та оплатити його. Якщо користувач знову заїде у профіль, то він зможе побачити це замовлення в історії, а також кількість ваших бонусів.

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
						50
Зм.	Арк	№ докум.	Підпис	Дата		

Енергія, що завжди поруч з тобою

Магазин преміальних енергорішень: від кабелів до потужних станцій.

 Гарантія 2 роки  Оплата частинами



150+
Товарів



24/7
Підтримка

Всі

Зарядні станції

Для смартфонів

Для ноутбуків

Павербанки

Авт...

Рисунок 3.1 – Головне вікно

Створити акаунт



Керуйте замовленнями та отримуйте бонуси.

ІМ'Я

Ivan

EMAIL

12356@gmail.com

ПАРОЛЬ

.....

Зареєструватися

Вже маєте акаунт? [Увійти](#)

Рисунок 3.2 – Діалогове вікно реєстрації акаунту

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		51

З поверненням!



Керуйте замовленнями та отримуйте бонуси.

EMAIL

12356@gmail.com

ПАРОЛЬ

.....

Увійти

Немає акаунту? [Створити](#)

Рисунок 3.3 – Діалогове вікно авторизації



Пошук...



Рисунок 3.4 – Вікно пошуку

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		52



Ваш кошик порожній

[Почати покупки](#)

Рисунок 3.5 – Діалогове вікно кошика

Отже, як бачимо даний ресурс є досить простим та лаконічним у використанні, що дозволить великій кількості користувачів здійснювати покупки в режимі онлайн та оплачувати їх, маючи доступ до мережі Інтернет у будь-якій точці світу.

3.3 Тестування вебзастосунку

Головна ціль даного тестування переконатися, що всі ключові бізнес-процеси, такі як пошук товарів, управління кошиком, симуляція оформлення замовлення, авторизація працюють коректно, стан додатку надійно оновлюється, а інтерфейс є адаптивним на різних пристроях.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		53

Загалом, для забезпечення високої якості застосунку застосовуються наступні види тестування:

- функціональне тестування, коли відбувається перевірка роботи модулів згідно з бізнес-логікою;
- UI/UX та адаптивність, тобто перевірка коректності відображення компонентів на мобільних, планшетних та десктопних екранах;
- тестування стану, а саме перевірка того, як зміна даних в одному компоненті, наприклад, додавання в кошик впливає на інші, наприклад, лічильник на іконці кошика в навігації;
- регресійне тестування, а саме перевірка збереження працездатності після додавання нових фіч.

Для проведення розроблюваного застосунку було розроблено функціональні тест-кейси, що перевіряють наявність тих чи інших помилок при створенні даного програмного продукту.

Модуль 1, що перевіряє каталог товарів та пошук та використовує useМето для оптимізації фільтрації. До 1 тестового сценарію входить фільтрація за категоріями. В якості дії потрібно натиснути на будь-яку категорію, наприклад, «Павербанки». Очікуваним результатам має бути той варіант, що у сітці товарів відображається лише товар «Ultra PowerBank 30000mAh». Активна категорія підсвічується синім кольором.

До 2-го тестового сценарію входить пошук за назвою. В якості дії необхідно ввести в поле пошуку слово «pro». Очікуваним результатом буде відображення товарів «GaN Charger 65W Pro» та «MacBook Pro Adapter 140W» незалежно від регістру вводу.

До 3-го тестового сценарію належить комбінована фільтрація. В якості дії потрібно обрати категорію «Для смартфонів» і ввести «cable». Очікуваним результатом може бути порожній список товарів, оскільки кабелі знаходяться в іншій категорії. Також не повинно бути помилок рендерингу.

Модуль 2, що здійснює перевірку кошика та оформлення (Cart & Checkout), який є ключовим бізнес-модулем, що відповідає за розрахунок

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		54

вартості. Першим тестовим сценарієм є додавання товару в кошик. В якості дії необхідно натиснути кнопку «+» на картці «Car Fast Charger 45W» (750 ₴). Очікуваним результатом є поява червоного бейджу над іконкою кошика з цифрою «1» (з анімацією bounce). Відкривається сайдбар кошика.

Другим тестовим сценарієм є зміна кількості та підрахунок суми. Дією є натискання «+» для збільшення кількості до 2. Очікуваним результатом є зміна кількості на 2, сума рядка стає 1500 ₴, загальна сума «Разом до оплати» оновлюється до 1500 ₴.

Третім тестовим сценарієм є захист від від'ємної кількості. В якості дії необхідно натиснути «-», коли кількість дорівнює 1. Очікуваним результатом є спрацювання $\text{Math.max}(1, \text{item.quantity} + \text{delta})$, при цьому кількість не зменшується нижче 1.

Четвертий тестовий сценарій тестує функціонал видалення товару. Дією має бути натиснути іконку кошика біля товару. Очікуваним результатом є зникнення товару з кошика. Якщо кошик стає порожнім, відображається повідомлення «Кошик порожній» і зникає кнопка «Оформити».

П'ятий тестовий сценарій, що стосується симуляції оплати. Дією має бути натиснення кнопки «Оформити», ввести будь-які дані в симуляцію форми картки, натиснути «Оплатити». Очікуваним результатом є статус, що змінюється на pending (спінер) на 2 секунди, потім success (зелене повідомлення) на 5 секунд. Кошик очищується (`setCart([])`).

Модуль 3, що відповідає за авторизацію та профіль користувача. До першого тестового сценарію відноситься вхід у систему. У якості очікуваної дії потрібно натиснути кнопку «Увійти», ввести email, натиснути кнопку відправки. Очікуваним результатом є закриття модального вікна, відповідно кнопка «Увійти» змінюється на кнопку профілю з іменем. Тестовий сценарій номер 1, що відповідає за відображення історії замовлень. Очікуваною дією є відкриття профілю після успішного оформлення замовлення.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		55

Очікуваним результатом є поява у блоці «Останні замовлення» нового запису зі статусом «Оплачено», поточною датою та правильною сумою транзакції. Лічильник замовлень збільшується на 1.

Другий тестовий сценарій вихід з акаунту. Дією має бути натиснення кнопки вийти з акаунту у профілі натиснути. Очікуваним результатом є Стан user, який скидається в null, сайдбар профілю закривається, в хедері знову з'являється кнопка «Увійти».

Модуль 4, що демонструє UI/UX та адаптивність (Mobile/Desktop). Тестовим сценарієм номер 1 є мобільна навігація. Дією є зменшення ширини вікна до розмірів смартфона (< 1024px). Очікуваним результатом є горизонтальне меню категорій ховається, з'являється кнопка «Гамбургер» (<Menu />).

Тестовий сценарій номер 2, що тестує горизонтальний скрол категорій. Дією є перевірка блоку кнопок категорій на мобільному пристрої. Очікуваним результатом є те, що категорії можна скролити горизонтально (клас overflow-x-auto no-scrollbar), кнопки не стискаються.

Тестовий сценарій номер 3, що тестує Z-Index конфлікти. Дією є відкриття кошику, після цього відкриття модального вікна авторизації шляхом маніпуляцій з UI. Очікуваним результатом є модальне вікно авторизації (z-[200]), що перекриває оверлей кошика (z-[100]), спінер завантаження перекриває все (z-[300]).

Оскільки наразі логіка зав'язана на локальному стані React (без серверного бекенду), тестування найкраще проводити наступними інструментами:

- Unit-тестування (Jest + React Testing Library);
- перевірка хука фільтрації (useMemo);
- перевірка логіки додавання чи видалення товарів (функції addToCart, updateQuantity);
- рендеринг компонентів.

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		56

Автоматизація повного шляху користувача від відкриття сторінки, фільтрації, додавання в кошик до симуляції успішної оплати та перевірки появи замовлення в особистому кабінеті.

Важливим є тестування, що автоматизує найважливіший шлях користувача, відображений у тест-кейсах: авторизація, пошук товару, додавання в кошик, зміна кількості, симуляція успішної оплати, перевірка появи замовлення в історії. Як приклад, далі подано E2E тест оформлення замовлення реальні дії користувача в браузері, що містить такі кроки користувача:

- клікає по елементах, знаходячи їх за текстом (cy.contains) або селекторами (cy.get);
- вводить текст у поля (.type());
- виконує перевірки (.should('be.visible')), гарантуючи, що інтерфейс реагує правильно, наприклад, сума в кошику змінюється після збільшення кількості;
- автоматично чекає на анімації та таймаути, наприклад, симуляцію підтвердження транзакції сервером.

Далі подано лістинг коду:

```
/// <reference types="cypress" />

describe('VoltSpace : Повний цикл покупки', () => {
  beforeEach(() => {
    // Пропускаємо, що застосунок запущено на стандартному порту
    Vite/Create-React-App
    cy.visit('http://localhost:3000');
  });
});
```

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
						57
Зм.	Арк	№ докум.	Підпис	Дата		

```

it('повинен дозволити користувачу увійти, знайти товар, додати в
кошик та успішно оплатити', () => {
  // Крок 1: Авторизація
  cy.contains('button', 'Увійти').click();

  // Заповнюємо форму логіну
  cy.get('input[name="email"]').type('test@voltspace.ua');
  cy.get('input[name="password"]').type('securepassword123');

  // В React-кодi у нас одна форма i для логіну, i для реєстрації (залежить
від стейту)
  // Натискаємо сабміт форми
  cy.contains('button', 'Увійти').click();

  // Перевіряємо, що модалка закрилась i з'явилась кнопка з іменем
профілю
  cy.contains('button', 'Іван').should('be.visible');

  // Крок 2: Пошук та додавання товару
  cy.get('input[placeholder="Пошук..."]').type('Car Fast');

  // Переконаємося, що товар знайдено
  cy.contains('h3', 'Car Fast Charger 45W').should('be.visible');

  // Знаходження картки товару i натискання кнопки "+" (додати в
кошик)
  cy.contains('h3', 'Car Fast Charger 45W')
    .parent() // переходимо до контейнера з інформацією про товар
    .find('button') // знаходимо кнопку всередині
    .click();

```

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		58

// Крок 3: Перевірка кошика та зміна кількості (ТС-2.01, ТС-2.02)

// Кошик має відкритися автоматично

cy.contains('h2', 'Кошик').should('be.visible');

cy.contains('h4', 'Car Fast Charger 45W').should('be.visible');

// Збільшуємо кількість до 2

// Знаходимо блок товару і натискаємо другу кнопку (кнопка "+")

cy.contains('h4', 'Car Fast Charger 45W')

.parent() // контейнер flex-grow

.find('button')

.eq(1) // індекс 1 - це друга кнопка (плюс)

.click();

*// Перевіряємо, що сума рядка оновилась (750 * 2 = 1500)*

// Використовуємо регулярний вираз, щоб ігнорувати можливі нерозривні пробіли (NBSP)

cy.contains(/1\s?500\s?\$/).should('be.visible');

// Крок 4: Оформлення замовлення (Checkout) (ТС-2.05)

cy.contains('button', 'Оформити').click();

// Переконаємось, що відкрилась форма оплати

cy.contains('h3', 'Оплата карткою').should('be.visible');

// Заповнюємо симуляцію картки

*cy.get('input[placeholder="**** * 4141 4141 4141 4141]').type('4141 4141 4141 4141');*

cy.get('input[placeholder="MM/YY"]').type('12/25');

*cy.get('input[placeholder="***"]').type('123');*

// Клікаємо кнопку оплати (шукаємо по слову "Оплатити")

cy.contains('button', 'Оплатити').click();

// Перевіряємо статус завантаження (спінер)

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		59

```

cy.contains('h2', 'Проводимо транзакцію...').should('be.visible');
    // Чекаємо повідомлення про успіх (в компоненті затримка 2 секунди,
    ставимо таймаут 5с для надійності)
    cy.contains('Оплата пройшла успішно!', { timeout: 5000
}).should('be.visible');
    // Крок 5: Перевірка історії замовлень (ТС-3.02)
    // Очікуємо, доки вікно успіху зникне (в коді це 5 секунд)
    cy.wait(5500);

    // Відкриваємо профіль користувача
    cy.contains('button', 'Олександр').click();
    cy.contains('h2', 'Мій кабінет').should('be.visible');

    cy.contains('Останні замовлення').parent().within() => {
        cy.contains('Оплачено').should('be.visible');
        cy.contains(/I\s?500\s?Є/).should('be.visible');
    });
    });
    });

```

Отже, в результаті проведення ручного функціональне та E2E автоматизації теоретичної симуляції можна зробити наступні висновки, що подано у таблиці 3.1.

Таблиця 3.1 - Результати виконання тест-кейсів

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		60

Модуль	Опис сценарію (Дія)	Фактичний результат	Статус	Примітки / Час виконання
Каталог	Фільтрація товарів за категорією «Павербанки»	Відображено лише товари відповідної категорії. Рендеринг миттєвий.	+	Хук useMemo оптимізує фільтрацію (< 10мс)
Каталог	Пошук товару за текстовим запитом («pro»)	Знайдено товари «GaN Charger 65W Pro» та «MacBook Pro Adapter». Регістр проігноровано.	+	Працює коректно завдяки toLowerCase()
Каталог	Комбінована фільтрація (несумісні категорія + текст)	Відображається порожній простір, застосунок не видає помилок (crash).	+	Можна додати UI-заглушку «Товари не знайдені»
Кошик	Додавання товару з каталогу	Товар з'являється в кошику, бейдж в хедері	+	Стан cart оновлено коректно

		оновлюється (+1) з анімацією bounce.		
Кошик	Зміна кількості товару та перерахунок суми	При натисканні «+» кількість зростає, загальна сума (cartTotal) перераховуєть ся миттєво.	+	Логіка reduce відпрацьовує без похибок
Кошик	Захист від від'ємної кількості товарів	При натисканні «-» на кількості 1, значення залишається 1.	+	Захист Math.max(1, ...) працює
Кошик	Видалення товару (іконка кошика)	Товар зникає, при порожньому кошику з'являється повідомлення Кошик порожній.	+	Кнопка Оформити зникає згідно з умовою
Оформ- лення	Симуляція повного циклу оплати (Checkout)	Форма оплати приймає дані. Стан pending (спінер) →	+	Таймери (2с та 5с) відпрацювали ідеально

Зм.	Арк	№ докум.	Підпис	Дата

КвРІПЗ.2201108.01.14.ПЗ

Арк.

62

		success (зелене повідомлення)		
Профіль	Вхід користувача (Mock Auth)	Модальне вікно закривається, з'являється кнопка профілю «Олександр».	+	preventDefault блокує перезавантаження
Профіль	Відображення збереженої історії замовлень	У профілі наявне нове замовлення зі статусом «Оплачено» та точною сумою кошика.	+	Зв'язок між cart та orders підтверджено
Профіль	Вихід з акаунту (Logout)	Стан user скинуто (null), інтерфейс повертається до стану гостя (кнопка «Увійти»).	+	Сайдбар профілю автоматично закрито
UI/UX	Адаптивність мобільної навігації (< 1024px)	Основне меню категорій ховається, з'являється	+	Tailwind класи hidden lg:flex працюють

		«Гамбургер» кнопка.		
UI/UX	Горизон- тальний скрол фільтрів на мобільних	Блок категорій можна плавно скролити свайпом. Кнопки не стискаються.	+	Клас по- scrollbar приховує візуальну смугу
UI/UX	Коректність Z-Index (перекриття вікон)	Модалка авторизації (z- [200]) перекриває кошик (z- [100]), спінер перекриває все (z-[300]).	+	Конфліктів шарів не виявлено

Висновки до третього розділу

Отже, у третьому розділі було продемонстровано реалізацію бази даних, інтерфейсу користувача та надано інструкцію для користувача. Також проведено тестування розроблюваного програмного застосунку та було виявлено усі ключові функції електронної комерції від пошуку до симуляції оплати працюють добре. Стан застосунку синхронізовано між компонентами. Завдяки використанню useMetho для фільтрації та локальному керуванню станом, інтерфейс реагує миттєво.

За результатами тестування рекомендується додати візуальний компонент «Нічого не знайдено», коли результати пошуку порожні. Для форми оплати у майбутньому варто додати валідацію введення, тобто форматування номеру картки з пробілами.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		64

Можна стверджувати, що MVP-версія застосунку повністю готова до демонстрації або розгортання в тестовому середовищі.

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		65

ВИСНОВКИ

Розробка мінімально життєздатного продукту (MVP) електронного магазину «VoltSpace» успішно завершена, а результати тестування підтверджують високу якість та стабільність реалізованого рішення. Програмний продукт повністю виконує своє головне завдання - надає користувачам швидкий, інтуїтивно зрозумілий та безперебійний доступ до критично важливого енергетичного обладнання. Усі ключові бізнес-процеси, включаючи динамічну фільтрацію каталогу, управління кошиком, розрахунок вартості та симуляцію оформлення замовлення з подальшим збереженням в історії профілю, працюють коректно та без збоїв.

Обраний технологічний стек виправдав себе повною мірою. Використання бібліотеки React у поєднанні з сучасними підходами до управління станом дозволило створити інтерактивний інтерфейс, який миттєво реагує на дії користувача без необхідності перезавантаження сторінки. Застосування фреймворку Tailwind CSS забезпечило ідеальну адаптивність вебзастосунку, гарантуючи високий рівень комфорту як на настільних комп'ютерах, так і на мобільних пристроях. Крім того, архітектура на основі імітації бази даних (Mock Data) дозволила швидко протестувати інтерфейсні рішення та підготувати міцний фундамент для майбутньої інтеграції з реальними серверами.

Окремо варто відзначити високу соціальну та практичну значущість даного проєкту в умовах сучасних викликів. Застосунок виступає не лише комерційним майданчиком, а й важливим інструментом забезпечення енергетичної стійкості населення та бізнесу. Зручний доступ до портативних станцій, павербанків та інших рішень для автономного живлення допомагає користувачам ефективно долати наслідки нестабільності електромережі, підтримувати зв'язок та зберігати працездатність.

У перспективі поточна версія застосунку є абсолютно готовою до переходу на наступний етап розвитку. Закладена гнучка архітектура дозволяє безболісно замінити локальні функції на справжні API-запити, підключити реальні платіжні

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		66

шлюзи та розгорнути повноцінну хмарну базу даних. Після впровадження кількох візуальних покращень, виявлених під час тестування, продукт буде повністю готовий до повноцінного комерційного запуску, просування на ринку та масштабування на значно ширшу аудиторію клієнтів.

					КвРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		67

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Про електронну комерцію : Закон України від 03.09.2015 р. № 675-VIII. *Відомості Верховної Ради України*. 2015. № 45. Ст. 410.
2. ДСТУ ISO/IEC 25010:2016. Системна та програмна інженерія. Вимоги до якості систем і програмних продуктів та їх оцінювання (SQuaRE). Моделі якості систем і програмних продуктів (ISO/IEC 25010:2011, IDT). [Чинний від 2016-11-01]. Київ : ДП «УкрНДНЦ», 2016. 44 с.
3. Банг С. С. Гнучка розробка вебсайтів. Київ : Фабула, 2019. 256 с.
4. Бем К. Розробка застосунків на React. Пер. з англ. Київ : Видавнича група BHV, 2020. 320 с.
5. Верес О. М., Обельницька С. П. Тестування програмного забезпечення : навч. посібник. Львів : Видавництво Львівської політехніки, 2022. 244 с.
6. Використання бібліотеки React для побудови інтерфейсів користувача : метод. вказівки / уклад. П. І. Антонюк. Київ : КПП ім. Ігоря Сікорського, 2021. 56 с.
7. Горелов О. А. Основи побудови Single Page Applications. *Сучасна інформатика*. 2021. № 4. С. 12–18.
8. Грегори П. Проектування та розробка веб-інтерфейсів. Львів : Старий Лев, 2021. 412 с.
9. Громов Ю. В. Ефективна фільтрація даних у React-застосунках. *Комп'ютерні системи та мережі*. 2022. Вип. 15. С. 45–52.
10. Дакетт Д. HTML і CSS: розробка та дизайн веб-сайтів. Київ : Основи, 2016. 482 с.
11. Дакетт Д. JavaScript і jQuery: інтерактивна веб-розробка. Київ : Основи, 2017. 640 с.
12. Канібор О. М. Сучасні тенденції в UI/UX дизайні мобільних пристроїв. *Мистецтво та технології*. 2023. № 2. С. 88–94.

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		68

13. Колісниченко Д. М. PHP, MySQL та JavaScript для створення динамічних веб-сайтів. Київ : МК-Прес, 2020. 512 с.
14. Крокфорд Д. JavaScript: Сильні сторони. Санкт-Петербург : Питер, 2013. 176 с.
15. Кузнецов І. А. Оптимізація продуктивності веб-додатків на стороні клієнта. *Вісник ЖДТУ*. 2022. № 1 (89). С. 110–117.
16. Лефтер А. С. Основи розробки веб-додатків : підручник. Одеса : ОНПУ, 2019. 305 с.
17. Мельник А. О. Архітектура комп'ютера : підручник. 3-тє вид. Львів : Магнолія 2006, 2021. 544 с.
18. Шевченко С. І. Моделювання баз даних та оптимізація запитів у сучасних вебсистемах : навч. посібник. Харків : ХНУРЕ, 2022. 188 с.
19. Осмоні Е. Шаблони проектування JavaScript. Київ : Діалектика, 2021. 288 с.
20. Пасічник В. В., Жежнич П. І., Гнатишина Р. Б. Глобальні інформаційні системи та технології (моделі ефективного аналізу, опрацювання та захисту даних). Львів : Видавництво Львівської політехніки, 2006. 350 с.
21. Поліщук О. С. Застосування компонентного підходу у фронтенд-розробці. *Інформаційні технології та системи*. 2023. Т. 2. С. 33–40.
22. Резіг Д. JavaScript. Книга ніндзя. 2-е вид. Москва : Вільямс, 2017. 544 с.
23. Сидоренко О. В. Методи тестування адаптивних веб-інтерфейсів. *Кибернетика та системний аналіз*. 2021. № 3. С. 156–162.
24. Сміт Д. Тестування React-компонентів. Київ : Наукова думка, 2022. 190 с.
25. Тарасенко Р. П. Еволюція CSS фреймворків: від Bootstrap до Tailwind. *Наукові записки НаУКМА*. 2023. Т. 5. С. 22–29.
26. Марченко О. В., Коваль А. М. Проектування та розробка серверної частини вебзастосунків на базі Node.js. *Інформаційні технології та комп'ютерна інженерія*. 2023. № 1. С. 45–53

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		69

27. Трофименко О. Г., Прокоп Ю. В. Вебтехнології та вебдизайн : навч. посібник. Одеса : Фенікс, 2019. 252 с.

28. Фрімен Е., Робсон Е. Head First JavaScript Programming. Пер. з англ. Київ : Видавнича група BHV, 2019. 640 с.

29. Чінназамі П. Тестування React застосунків за допомогою Jest. Київ : МК-Прес, 2021. 240 с.

30. Швайгер С. Розробка інтерфейсів на основі компонентів. *Software Engineering Today*. 2022. № 8. С. 67–74.

					КВРІПЗ.2201108.01.14.ПЗ	Арк.
Зм.	Арк	№ докум.	Підпис	Дата		70

Додаток А
(обов'язковий)

ТЕХНІЧНЕ ЗАВДАННЯ

1. Загальні відомості.

Назва проєкту VoltSpace (v2.5).

Тип програмного продукту - односторінковий вебзастосунок (SPA) для електронної комерції.

Призначення - продаж та розповсюдження обладнання для автономного енергозабезпечення - зарядні станції, павербанки, аксесуари.

2. Мета та цілі проєкту - створення зручного інтерфейсу для вибору та придбання товарів в умовах критичного попиту із забезпеченням високої швидкості роботи на мобільних пристроях із реалізацією механізму керування станом кошика та профілю користувача без перезавантаження сторінок.

3. Технічний та технологічний стек.

Frontend - React 18+ (Hooks: useState, useEffect, useMemo).

Стилізація- Tailwind CSS (Utility-first framework).

Іконки - Lucide React (SVG-пакет).

Сховище даних (MVP) - In-memory state (JSON-структури), з можливістю подальшої міграції на Firebase або PostgreSQL.

4. Функціональні вимоги.

4.1. Каталог та Пошук мають містити відображення сітки товарів з фото, ціною та коротким описом.

Має бути додана динамічна фільтрація за категоріями, наприклад, зарядні станції, павербанки тощо.

Миттєвий пошук за назвою товару (case-insensitive).

Оптимізація рендерингу фільтрів за допомогою показу результатів.

4.2. Система кошика повинна містити додавання товарів у кошик з будь-якої частини каталогу.

Коригування кількості одиниць товару з обмеженням мінімальної кількості = 1.

Видалення окремих позицій.

Автоматичний підрахунок загальної вартості в режимі реального часу.

Візуальна індикація кількості товарів на іконці кошика в хедері.

4.3. Оформлення замовлення (Checkout)

Модальне вікно або сайдбар для введення платіжних даних.

Симуляція процесу оплати з візуальним фідбеком (Loader → Success Message).

Очищення кошика після успішної транзакції.

Додавання завершеного замовлення до історії в профілі користувача.

4.4. Авторизація та Профіль

Симуляція входу в систему (Mock Login).

Відображення персональних даних (ім'я, email, бонусний баланс).

Перегляд списку попередніх замовлень зі статусами.

5. Нефункціональні вимоги.

5.1. Адаптивність (Responsiveness).

Desktop (1280px+) з повною сіткою товарів (4 колонки), розгорнуте меню.

Tablet (768px - 1024px) з сіткою в 2-3 колонки, прихований пошук за іконкою.

Mobile (< 768px) із сіткою в 1-2 колонки, гамбургер-меню для категорій, оптимізовані тач-зони для кнопок.

5.2. Продуктивність та UI/UX.

Час відгуку інтерфейсу на дію користувача - не більше 100 мс.

Використання backdrop-blur та плавних переходів (Transitions) для покращення візуального досвіду.

Забезпечення читабельності тексту, тобто контрастність згідно з WCAG.

6. Структура даних (Schema)

Product Object:

```
{
```

```
"id": "number",  
"name": "string",  
"category": "string",  
"price": "number",  
"image": "url",  
"description": "string"  
}
```

Order Object:

```
{  
  "id": "number",  
  "date": "string",  
  "items": "Array<Product>",  
  "total": "number",  
  "status": "string"  
}
```

7. Етапи розвитку (Roadmap)

Фаза 1 (Поточна) - розробка клієнтської частини, статика, локальний стейт.

Фаза 2 - інтеграція з Backend API (Node.js/Python) та базою даних.

Фаза 3 - впровадження реальної системи авторизації (JWT/OAuth).

Фаза 4 - підключення платіжного шлюзу (LiqPay/Stripe).

Фаза 5 - додавання кабінету адміністратора для управління складом.

Додаток Б
(обов'язковий)
ПРЕЗЕНТАЦІЙНІ МАТЕРІАЛИ

Хмельницький національний університет
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень

ВИКОНАВ ЗДОБУВАЧ:
Іван РАЙЧУК (гр. ППЗ-22-1)

НАУКОВИЙ КЕРІВНИК:
канд. пед. наук, доцент Оксана ОНИШКО

Хмельницький 2026

Рисунок А.1 – Титульний слайд, представлення

Актуальність теми

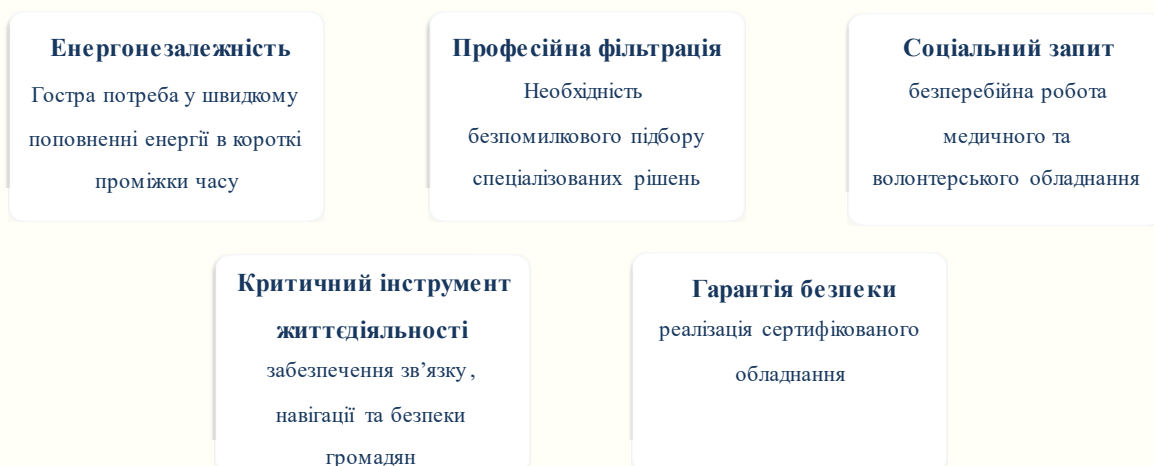


Рисунок А.2 – Актуальність теми

Мета і завдання роботи

Мета роботи:

Розробка вебресурсу для електронної торгівлі зарядними пристроями з кошиком замовлень.

Завдання:

- ✓ Аналітичний огляд предметної області
- ✓ Аналіз існуючих вебзастосунків
- ✓ Переваги та недоліки проаналізованих рішень
- ✓ Постановка задачі та розробка ТЗ
- ✓ Вибір та опис архітектури
- ✓ Аналіз та вибір технологій
- ✓ Програмна реалізація та тестування

Рисунок А.3 – Мета та Завдання

Предметна область



Рисунок А.4 – Змістовий аналіз предметної області, її структурних та функціональних особливостей

Аналіз існуючих рішень

Платформа	Rozetka	Comfy	Citrus	Mojo	Brain
Ключові переваги	Найбільший асортимент, потужна фільтрація	Швидкий SPA -кошик, програма лояльності	Сучасний UI/UX, якісний відеоконтент	Баланс SSR, функція Trade-in	Стабільність, швидка фільтрація параметрів
Технічні недоліки	Перевантаження JS-скриптами (високий TBT)	Складне DOM-дерево (важке для мобільних)	Надлишок трекінгових пікселів та сторонньої аналітики	Застаріла логіка Checkout (синхронні запити)	Відсутність оптимізації зображень (немає WebP)
Архітектурна особливість	Мікросервісна архітектура	Single Page Application елементи	Ефективна CDN для медіа	Server-Side Rendering (SSR)	Консервативний та стабільний стек

Рисунок А.5 – Аналіз наявного програмно-технічного забезпечення

Функціональні та нефункціональні вимоги

Функціональні:

- ✓ Система фільтрації та сумісності
- ✓ Інтелектуальний пошук
- ✓ Багатофункціональний кошик
- ✓ Швидке оформлення замовлення
- ✓ Особистий кабінет з історією та бонусним рахунком

Нефункціональні:

- ✓ Стійкість до пікових навантажень.
- ✓ Висока швидкість завантаження сторінок
- ✓ Підтримка мобільної версії
- ✓ Шифрування даних
- ✓ Відповідність стандартам PCI DSS для обробки платежів

Рисунок А.6 – Визначення функціональних та нефункціональних вимог до ПЗ

Архітектура вебресурсу

Система побудована на багаторівневій клієнт-серверній моделі з чітким розділенням обов'язків

- ✓ **Frontend:** React SPA (Single Page Application);
- ✓ **Backend:** Node.js + Express (REST API);
- ✓ **ORM:** Prisma для безпечних запитів
- ✓ **Database:** PostgreSQL (Реляційна БД).

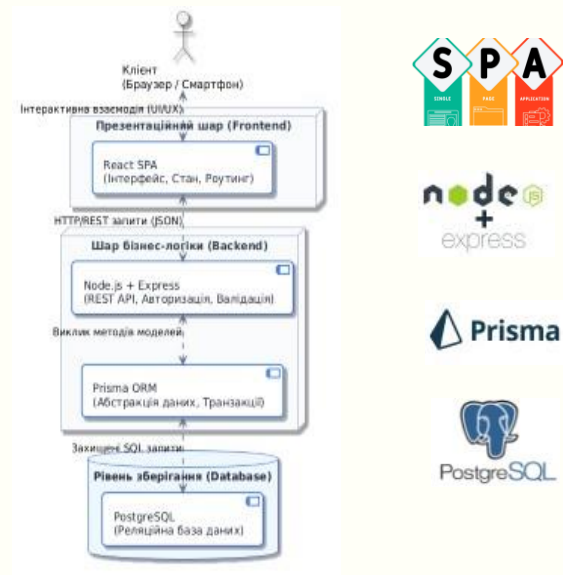


Рисунок А.7 – Вибір типу архітектури та шаблонів проектування

Декомпозиція, залежності, інтерфейси

Декомпозиція

Багаторівнева модель із розділенням обов'язків.
Ізольовані вузли: SPA-клієнт, RESTful API сервер та БД PostgreSQL.

Залежності

Відсутність прямого доступу клієнта до сховища; комунікація виключно через маршрути сервера та технологію Prisma.

Інтерфейси

Чітко визначені протоколи HTTP (JSON -запити) та SQL для взаємодії між рівнями системи.

Рисунок А.8 – Опис декомпозиції, залежностей, інтерфейсів

Проектування модулів і даних

- ❗ **Логічна структура:** чотири базові сутності: Користувачі, Товари, Замовлення та Елементи замовлення.
- ❗ **Реляційні зв'язки:** Використання зовнішніх ключів (Foreign Keys) для забезпечення цілісності.
- ❗ **Фінансова точність:** фіксація ціни на момент купівлі та використання типу DECIMAL(10,2) для точності розрахунків.
- ❗ **Дотримання ACID:** Атомарність та ізолюваність кожної транзакції.
- ❗ **Валідація:** Унікальність Email та перевірка балансу на рівні бази даних.

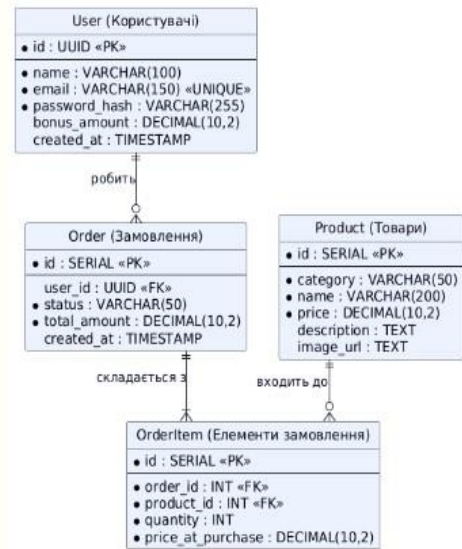


Рисунок А.9 – Проектування модулів і даних

Вибір технологій

- ✓ **SPA:** сучасний односторінковий додаток
- ✓ **React:** концепція Virtual DOM, React Hooks, useState, useMemo
- ✓ **JavaScript (ES6+):** з використанням розширення синтаксису JSX
- ✓ **Tailwind CSS :** utility-first фреймворк, адаптивний дизайн
- ✓ **Lucide React:** колекція векторних іконок
- ✓ **Mock Data :** чітко структуровані JSON-об'єкти

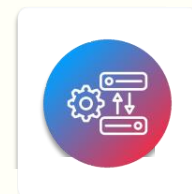
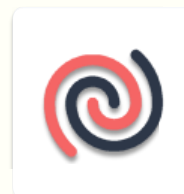
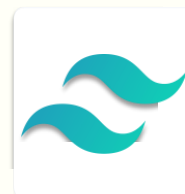
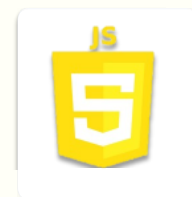
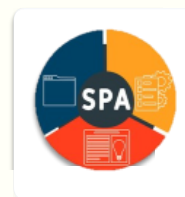


Рисунок А.10 – Аналіз та вибір технологій

Реалізація модулів та база даних



Рисунок А.11 – Реалізація модулів і база даних

Тестування ПЗ

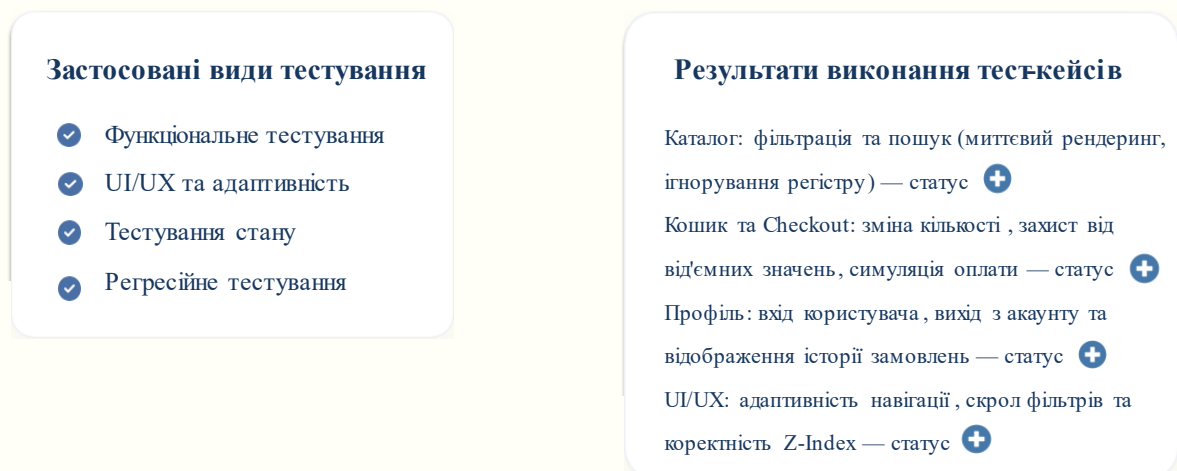


Рисунок А.12 – Тестування ПЗ

Висновки

Пункт завдання	Результат виконання
Аналітичний огляд предметної області	Встановлено, що платформи електронної комерції вимагають чіткого розуміння як технічних характеристик товару, так і психології покупця.
Аналіз вже існуючих вебзастосунків для торгівлі зарядними пристроями	Проведено аналіз вже існуючих вебзастосунків, що допомагають здійснити торгівлю зарядними пристроями.
Переваги та недоліки проаналізованих бізнес-рішень	Визначено переваги та недоліки проаналізованих бізнес-рішень.
Постановка задачі та ТЗ	Здійснено постановку задачі та розроблено технічне завдання.
Вибір та опис архітектури розроблюваного ПП	Здійснено вибір та опис архітектури розроблюваного програмного продукту; обрано багаторівневу клієнт-серверну архітектуру.
Аналіз та вибір технологій для розробки ПП	Проведено аналіз та вибір технологій; обраний технологічний стек виправдав себе повною мірою.
Реалізувати та протестувати розроблений ПП	Розробка MVP успішно завершена; результати тестування підтверджують високу якість та стабільність реалізованого рішення.

Рисунок А.13 – Висновки

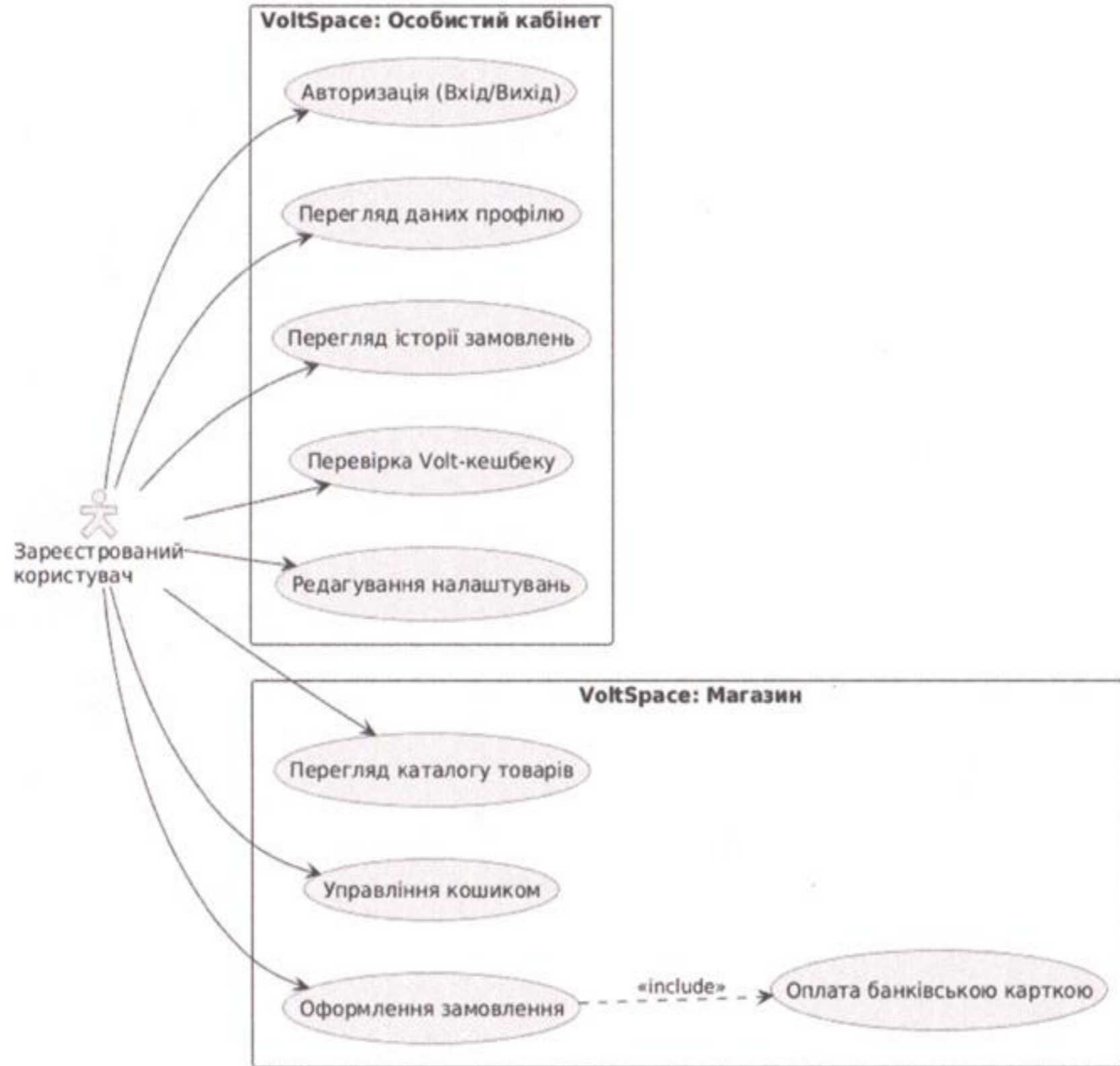
ДЯКУЮ ЗА УВАГУ!

Готовий відповісти на ваші запитання

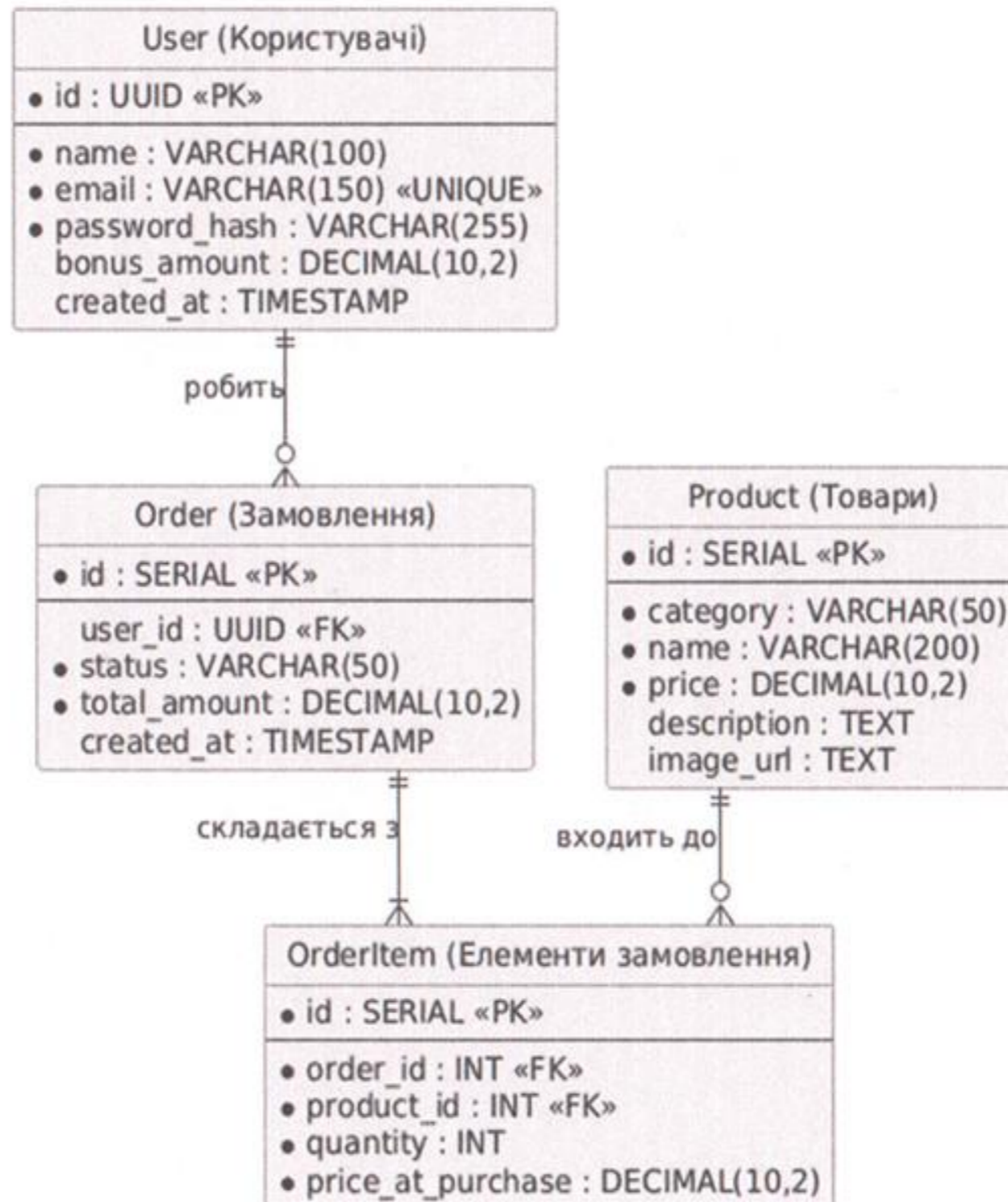
 Іван Райчук | ХНУ 2026

Рисунок А.14 – Дякую за увагу

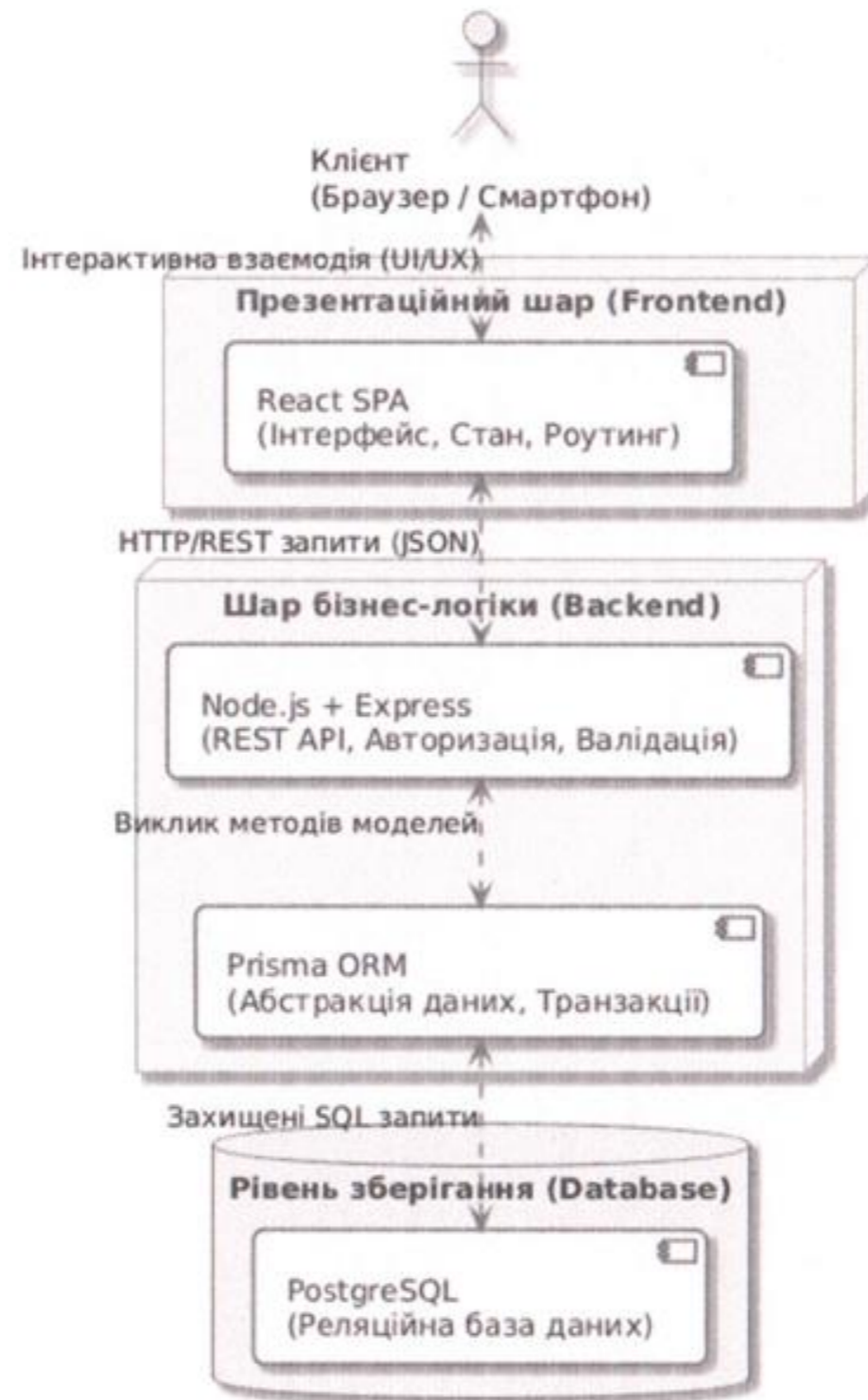
ГРАФІЧНА ЧАСТИНА



					КвРІПЗ. 2201108.01.14.ПЗ		
					Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень		
					Діаграма варіантів використання		
Зм.	Арк.	№ докум.	Підпис	Дата	Літера	Маса	Масштаб
Розробник		Райчук І.В.	<i>[Signature]</i>	29.05			
Керівник		Онишко О.Г.	<i>[Signature]</i>	29.05			
Консульт.					Аркуш 1	Аркушів 3	
Н. Контроль		Праворська І.Г.	<i>[Signature]</i>	29.05	ХНУ, ІПЗ-22-1		
Зав. каф.		Бедратюк Л. П.	<i>[Signature]</i>	29.05			



					КвРІПЗ. 2201108.01.14.ПЗ					
					Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень			Літера	Маса	Масштаб
Зм. Арк.	№ докум.	Підпис	Дата	Схема бази даних						
Розробив	Райчук І.В.	<i>[Signature]</i>	29.05				Аркуш 2	Аркушів 3		
Керівник	Онишко О.Г.	<i>[Signature]</i>	29.05							
Консульт.										
Н. Контр.	Праворська Н.І.	<i>[Signature]</i>	29.05							
Зав. каф.	Бедратюк Л. П.	<i>[Signature]</i>	29.05							
							ХНУ, ІПЗ-22-1			



					КвРІПЗ. 2201108.01.14.ПЗ			
					Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень Архітектура застосунку	Літера	Маса	Масштаб
Зм.	Арк.	№ докум.	Підпис	Дата				
Розробив		Райчук І.В.		29.05				
Керівник		Онишко О.Г.		29.05				
Консульта.								
Н. Контроль		Праворська Н.Г.		29.05				
Зав. каф.		Бедратюк Л.П.		29.05				
						Аркуш 3	Аркушів 3	
					ХНУ, ІПЗ-22-1			

СУПРОВІДНІ ДОКУМЕНТИ

Завідувачу кафедри інженерії
програмного забезпечення проф.
Леоніду БЕДРАТЮКУ
здобувача вищої освіти
Райчука Івана Вадимовича
факультет ІТ, ІV курс, група ІПЗ-22-1

ЗАЯВА

З правилами чинного Положення про систему забезпечення академічної доброчесності в Хмельницькому національному університеті, згідно з яким виявлення академічного 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 наявність академічного плагіату оповіщений та надаю свою згоду на обробку й збереження університетом моєї роботи в інституційному репозитарії Хмельницького національного університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-обчислювального комплексу StrikePlagiarism та/або програмно-технічного засобу AntiPlagiarism 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 текстових збігів у роботах.

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

01.05.26

дата


підпис

Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Іван РАЙЧУК

Співавтор:

Назва: Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень

Науковий керівник: канд. пед. наук, доцент Оксана ОНИШКО

Підрозділ: Кафедра інженерії програмного забезпечення

Коефіцієнт подібності 1: 4.84%

Коефіцієнт подібності 2: 1.57%

Мікропробіли: 14

Заміна букв: 0

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2026-05-25 01:41:25.0

Після аналізу Звіту подібності констатую наступне:

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедур. Таким чином робота не приймається.

Обґрунтування:

Дата 25.05.2026

експерт



Anti-Plagiarism (<http://ap.km.ua>) v-16.718

Максимальне співпадіння з одним документом 26.0%

Словники перевірки: UA, US, RU. Помилки в документах: 12%

ID: 272084 Назва: Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень Додано в БД: 2026-05-25 Автора: Іван РАЙЧУК Керівники: канд. пед. наук, доцент Оксана ОНИШКО Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	69353	529	19609 (28%)	146 (28%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми
269635	Назва: Переддипломна практика Додано в БД: 2026-03-02 Автора: Іван РАЙЧУК Керівники: О. Г. Онишко, канд. пед. наук, доцент Консультанти: Опоненти:	17907 (26.0%)	121 (23.0%)

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ
освітнього ступеня «Бакалавр»

Дипломник Райчук Іван Вадимович

Тема Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень

Спеціальність 121 – Інженерія програмного забезпечення

Обсяг кваліфікаційної роботи:

Кількість листів креслень 3; кількість сторінок записки 83

1. Короткий зміст пояснювальної записки та прийнятих рішень У кваліфікаційній роботі досліджено і проаналізовано предметну область електронної комерції та визначено усі функціональні та нефункціональні вимоги. Був проведений аналіз п'яти провідних платформ на ринку України, розглянуто їхні архітектурні переваги й недоліки та доведено актуальність вебресурсу «VoltSpace». Розглянуто багаторівневий клієнт-серверний стек технологій на базі React, Node.js та PostgreSQL, в результаті чого створено програмне забезпечення. Також було проведено комплексне функціональне, UI/UX та автоматизоване E2E тестування, за результатами якого доведено, що розроблене програмне забезпечення працює коректно та готове до експлуатації

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 У вступі обґрунтовано актуальність теми, визначено мету та завдання бакалаврської роботи. У першому розділі проведено змістовний аналіз предметної області електронної комерції, виконано технічний огляд п'яти найбільших наявних рішень на ринку та чітко визначено функціональні й нефункціональні вимоги до вебплатформи. У другому розділі проаналізовано сучасні архітектурні підходи й визначено, що система базуватиметься на багаторівневій клієнт-серверній моделі, шаблоні SPA з використанням технологій React, Node.js та PostgreSQL. У третьому розділі розгорнуто реляційну базу даних за допомогою SQL-скриптів, реалізовано програмні модулі інтерфейсу з використанням Tailwind CSS та описано інструкцію користувача. Також у цьому розділі виконано функціональне, адаптивне й автоматизоване E2E тестування за допомогою Cypress, за результатами якого повністю підтверджено коректність та стабільність роботи створеного вебресурсу.

4. Позитивні сторони роботи Тематика кваліфікаційної роботи є актуальною, оскільки в умовах сучасних викликів в Україні швидкий і безперервний доступ до сертифікованого енергетичного обладнання є критично важливим для життєдіяльності громадян та бізнесу. Також автором було застосовано новітній клієнт-серверний стек технологій на базі React та Node.js, впроваджено сучасні методи оптимізації інтерфейсу для роботи в умовах слабкого зв'язку та реалізовано автоматизоване E2E тестування за допомогою інструменту Cypress.

5. Негативні сторони роботи У роботі на етапі створення MVP-версії було використано архітектуру імітації бази даних (Mock Data) на стороні клієнта – у майбутньому доцільно повністю підключити розроблені SQL-скрипти до хмарної бази даних. Також для форми оплати варто впровадити реальну валідацію введення даних банківської картки та її форматування з пробілами.

6. Оцінка графічного оформлення та пояснювальної записки Графічне оформлення виконано відповідно до теми кваліфікаційної роботи та подано у вигляді схем архітектури, макетів інтерфейсу платформи й результатів тестування. Пояснювальна записка оформлена згідно з вимогами чинних стандартів.

7. Відгук про кваліфікаційну роботу в цілому Кваліфікаційна робота заслуговує позитивної оцінки. Матеріал пояснювальної записки структурований, послідовний, чіткий та простий, що дозволяє чітко зрозуміти викладений матеріал у рамках тематики розробки вебресурсу «VoltSpace». Графічний матеріал дає можливість наочно побачити деталі архітектури, схему бази даних та макети інтерфейсу системи.

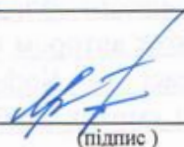
8. Інші зауваження

9. Оцінка кваліфікаційної роботи Кваліфікаційна робота виконана у повному обсязі, відповідає поставленій задачі та заслуговує на оцінку «добре».

РЕЦЕНЗЕНТ (прізвище, ім'я, по-батькові, посада, місце роботи) Капустян
Марія Вікторівна, кандидат технічних наук, доцент, за
кафедрою Інженерії та Систем Інформації РХНУД

“29” травня

2026 р.


(підпис)

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

Підтверджуємо ознайомлення з результатами звіту/звітів перевірки роботи, продуктованими програмно-технічним засобом (ами), на наявність текстових збігів.

Назва кваліфікаційної роботи: «Вебресурс для електронної торгівлі зарядними пристроями з кошиком замовлень»

Автор: Райчук Іван Вадимович

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

Спеціальність: 121 – Інженерія програмного забезпечення

Науковий керівник: Онишко Оксана Григорівна, канд. пед. наук, доцент

Після аналізу звіту/звітів зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається до захисту.	відповідає
2	Виявлені запозичення не є академічним 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.	
3	Виявлені запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Виявлені запозичення 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 досягнуті. Робота може бути допущена до захисту після того, як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
5	Інше:	

Підтвердження:

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

1) у тексті кваліфікаційної роботи системою перевірки на плагіат Anti-Plagiarism виявлено схожість з деякими документами у частині загальноживаних обов'язкових словосполучень у стандартних бланках, у структурі змісту, назвах назвах розділів/підрозділів, рамках форм, у назвах та URL-адресах публікацій переліку джерел посилання;

2) запозичення, виявлені у тексті роботи, є фрагментарними.

Максимальний обсяг запозичень, визначений системою Anti-Plagiarism, складає 26.0% з одного джерела. Загальна сумарна подібність у базі даних складає 28% за символами та 28% за лексемами.

Крім того, за результатами додаткового аналізу коефіцієнт подібності 1 становить 4.84%, коефіцієнт подібності 2 – 1.57%. Виявлено 14 мікропробілів. Не виявлено зайвих білих знаків або маніпуляцій з інтервалами. З урахуванням наведених обґрунтувань, відповідає характеру теми і свідчить на користь кваліфікаційної роботи.

Дата 28.05.26

Завідувач кафедри

Гарант освітньої програми

Керівник кваліфікаційної роботи



Леонід БЕДРАТЮК

Леонід БЕДРАТЮК

Оксана ОНИШКО