
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«До захисту допущено»

В.о. завідувача кафедри

**Ігор
ШЕЛЕХОВ**

(підпис)

червня 2024 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня бакалавр

зі спеціальності 122 - Комп'ютерних наук,
освітньо-професійної програми «Інформатика»
на тему: «Мобільний додаток для електронної комерції «E-commerce app»»
здобувача групи ІН-06-2 Чумакова В'ячеслава Руслановича

Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на
відповідне джерело.

Керівник, доцент, кандидат
технічних наук

(підпис) **В'ячеслав ЧУМАКОВ**

(підпис) **Віктор ОБОДЯК**

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра комп'ютерних наук

«Затверджую»
В.о. завідувача кафедри
_____ Ігор ШЕЛЕХОВ
(підпис)

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття освітнього ступеня бакалавра

зі спеціальності 122 - Комп'ютерних наук, освітньо-професійної програми «Інформатика»
здобувача групи ІН-06-2 Чумакова В'чеслава Руслановича

1. Тема роботи: «Мобільний додаток для електронної комерції «E-commerce app»»
затверджую наказом по СумДУ від 22» квітня 2024 р. № 0414-VI
2. Термін здачі здобувачем кваліфікаційної роботи до 13 червня 2024 року
3. Вхідні дані до кваліфікаційної роботи
4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити)

1. Інформаційний огляд, аналіз предметної області, постановка задачі, моделювання інформаційної системи. 2. Вибір методів рішення задачі. 3. Програмна реалізація. 4. Аналіз отриманих результатів

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

Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

Дата видачі завдання « ____ » _____ 20 ____ р.

Завдання прийняв
до виконання

Керівник

_____ (підпис)

_____ (підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання	Примітка
1	Затвердження теми роботи	22.04.2024	
2	Вивчення та аналіз задачі	7.05.2024	
3	Поглиблене дослідження бібліотек, інструментів, технологій, що будуть використовуватися	10.05.2024	
4	Програмна реалізація системи	19.05.2024	
5	Аналіз отриманих результатів	22.05.2024	

Здобувач вищої освіти

Керівник

АНОТАЦІЯ

Записка: 36 стр., 22 рисунок, 2 додаток, 12 використаних джерел.

Обґрунтування актуальності теми роботи – тема кваліфікаційної роботи є актуальною, оскільки з огляду на стрімкий розвиток технологій та збільшення кількості користувачів мобільних пристроїв та підвищення вимог до швидкості та зручності обслуговування клієнтів підкреслюють необхідність впровадження інноваційних інструментів для підтримки бізнесу в цифровому середовищі.

Об’єкт дослідження — процес створення мобільного додатку для електронної комерції “E-commerce app”

Мета роботи — розробка та дослідження інформаційної системи управління контентом та замовленнями для роздрібної торгівлі.

Методи дослідження — аналіз користувацького досвіду, порівняльний аналіз з іншими ринковими учасниками.

Результати — створений мобільний додаток під платформу IOS та Android який включає в себе повний процес створення замовлення та перегляд актуального статусу виконання замовлення. Має мінімалістичний дизайн та зрозумілі для сприйняття елементи взаємодії.

ЕЛЕКТРОННА КОМЕРЦІЯ, REACT-NATIVE, MOBILE APP, JAVASCRIPT

ЗМІСТ

ВСТУП.....	5
1 ІНФОРМАЦІЙНИЙ ОГЛЯД.....	7
1.1. Що таке Електронна комерція.....	7
1.2. Типи Інтернет-магазинів.....	8
1.3. Огляд аналогів.....	9
1.4. Постановка задачі.....	13
2.1 Вибір технології.....	14
2.2 Проектування системи замовлення.....	14
3. ПРОГРАМНА РЕАЛІЗАЦІЯ.....	16
3.1. Опис процесу реалізації.....	16
3.2 Розробка зовнішнього вигляду системи.....	17
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	32
ДОДАТКИ.....	33
ДОДАТОК А.....	33
ДОДАТОК Б.....	35

ВСТУП

Інформаційний всесвіт все більше стає частиною нашого життя. Кожен аспект нашого життя пов'язаний з інтернетом, або так чи інакше взаємодіє з ним. Кожен день, мільйони людей активно використовують мобільні додатки для розваг, роботи, пошуку інформації, покупок, спілкування, тощо.

Популярні компанії та корпорації створюють мобільні додатки, щоб користувачі мали доступ до потрібної інформації, потрібних функцій. Завдяки розширенню можливостей у різних областях, мобільні додатки входять в еру WEB-2.0. Інтернет-банкінг, меседжери, інтрент-магазини, криптовалюта, NFT - усе це прекрасно демонструє можливості нової ери.

Ера WEB-2.0 показала, що миттєва оплата рахунків, спілкування з рідними та друзями коли забажаєш, купити бажаний товар, замовити доставку їжі додому, оплата інтернет грошима - все це не є далеким майбутнім. Всі ці мобільні додатки на прикладі продемонстрували нам корисність, швидкість та зручність їх використання.

На сьогоднішній день більшість людей не може уявити своє повсякденне життя без використання додатків. Для деяких з них це робота, для інших - складова щоденного відпочинку, для третіх - навчання. Мобільні додатки охопили всі доступні сфери і продовжують своє розширення. Основним показником зручності є доступність. У кожного є мобільний телефон, в якому виробник вже встановив необхідні додатки, а встановити нові не викликає складнощів.

За останні роки, у додатках активно почався розвиток приватності та цілісності конфіденційних даних. Всі компанії прагнуть максимально захистити користувачів та власні додатки від шахраїв які можуть зашкодити роботі компанії.у зв'язку між додатком і сервером використовується шифрування та захищений протокол https, а платформи для публікації мають особисту перевірку додатків. На сьогодні, для додатків які використовують

персональні дані користувача, потребують додати документи “правила користування” та “політика конфіденційності”, щоб користувачі знали яким чином компанія буде використовувати їх дані.

Метою даного дослідження є розробка мобільного додатку для електронної комерції, який буде давати можливість різним компаніям реалізувати себе та свій товар у інтернет-просторі. Через додаток компанії зможуть заповнити контактну інформацію та перенести свою вітрину товарів у додаток, щоб користувачі мали актуальну інформацію про наявність, знижки та стан бажаного товару. Також зможе оформити онлайн замовлення з доставкою “під двері”, та слідкувати за етапами обробки його замовлення в онлайн-режимі.

1 ІНФОРМАЦІЙНИЙ ОГЛЯД

1.1. Що таке Електронна комерція

Коли йдеться мова про електронну комерція, перше що спадає на думку це - інтернет магазини. Інтернет-магазин - це система яка дозволяє здійснювати пряму продаж товарів та послуг від компанії до покупця, враховуючи доставку. При цьому, розміщення інформації товару та укладання угоди відбувається в середині мережі (в інтернет-магазині).

Інтернет магазини можуть надавати різні можливості, такі як:

- Перегляд товарів: опис, стан наявності, фото\відео, коментарі, рейтинг.
- Оформлення замовлення: оформлення онлайн-замовлення обраних товарів з указанням необхідної контактної інформації для ідентифікування покупця.
- Контроль замовлення: перегляд стану обробки замовлення, контроль доставки, перевірка оплати.
- Онлайн оплата: можливість оплатити замовлення онлайн, з банківської картки.
- Авторизація: можливість авторизуватися у системі за допомогою зручних методів.
- Постійна підтримка: консультант та адміністратор завжди можуть допомогти у несподіваних ситуаціях.
- Історія: збереження переглянутих товарів, історія замовлень, рекомендації, обрані товари.

Інтернет магазини дуже корисні компаніям які мають великий асортимент товарів, та можуть доставляти їх в різні регіони або країни. Він допомагає систематизувати замовлення, звіт кількості товару, базу клієнтів та масштабування бізнесу у інформаційному світі.

1.2. Типи Інтернет-магазинів

Інтернет-магазини класифікуються за функціоналом, призначенням та використанням. Виділимо основні типи:

Таблиця 1.1 – Різновиди Інтернет-магазинів.

Назва	Особливості	Приклади функціональності
Сайт-візитка	Невелика кількість сторінок, скорочена інформація про товари та компанію	Загальна інформація, популярні товари, контакти з компанією
Сайт компанії з каталогом продукції	Представлений весь каталог товарів або послуг компаній які вона може запропонувати	Перегляд інформації продукції, має архітектуру категорій то товару, зв'язок з компанією (на деяких можна оформити замовлення)
Інтернет-магазин	Гіпермаркет - 1000-10000 неунікальних товарів Невеликі інтернет-магазини - унікальні товари (100-1000 пропозицій)	Перегляд продукції, оформлення замовлення, зв'язок з компанією
Корпоративний магазин	Потужний функціонал контролю замовлень, має велику кількість розділів та підрозділів. Великий вибір способів замовлення, коригування товарів, детальна інформація про всі процеси у мережі	Оптове замовлення товарів, підписання контакту з компанією, контроль всіх процесів які стосуються клієнта

1.3. Огляд аналогів

Prom.ua [1] - один із найбільших інтернет магазинів який пропонує велику кількість товарів які пропонують компанії або підприємства. Всі товари контролюються адміністрацією (рис. 1.1, 1.2).

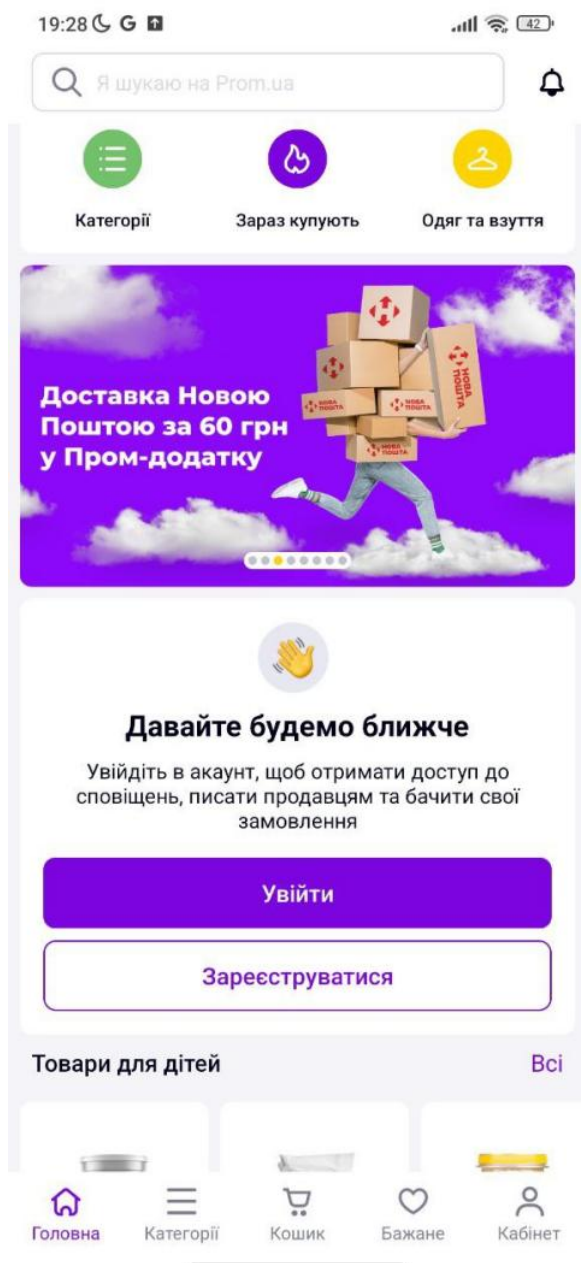


Рисунок 1.1 – Інтернет-магазин Prom.ua.

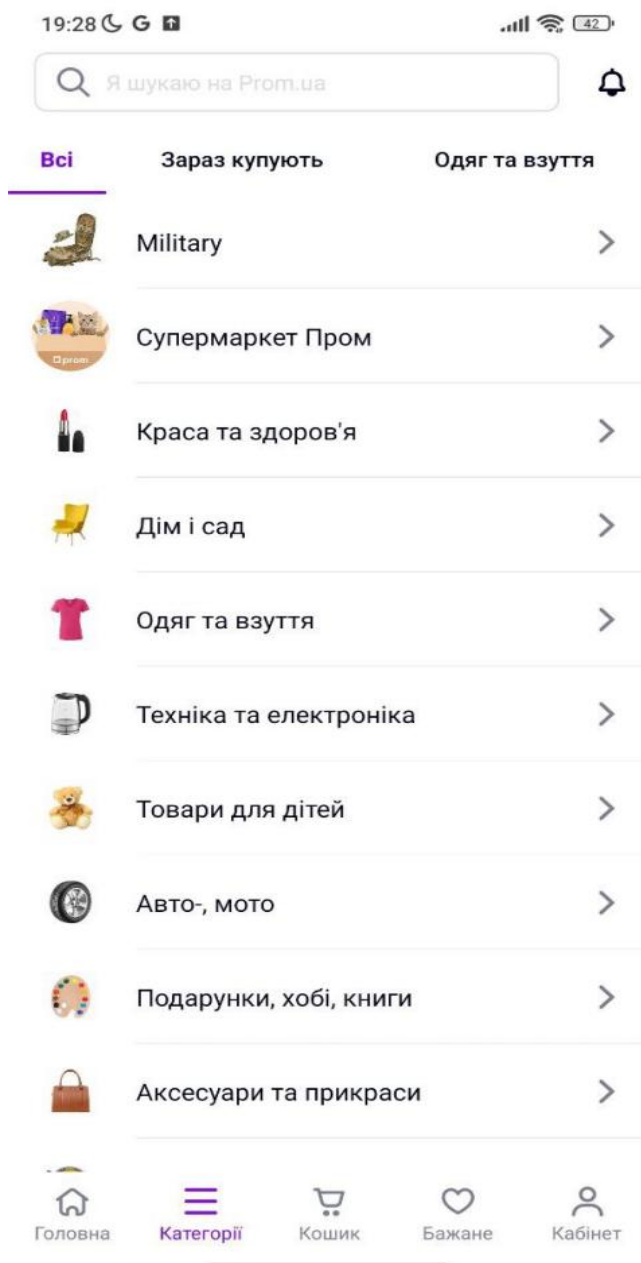


Рисунок 1.2 – Каталог Prom.ua.

Плюси:

- Багатий вибір продукції.
- Швидка реакція адміністрації.
- Мінімалістичний дизайн.
- Зручна форма замовлення.
- Індивідуальний підхід під продавця.
- Захищена оплата “Prom-оплата”.

Мінуси:

- Обробка замовлення залежить від покупця.
- Деяка документація не актуальна.

Rozetka [2] - Популярний та великий інтернет-магазин для покупок перевірених товарів. Товари регулюються компаніями та підприємствами, також на платформі є товари компанії “Rozetka” (рис. 1.3, 1.4).



Рисунок 1.3 - Головна сторінка додатку "Rozetka".

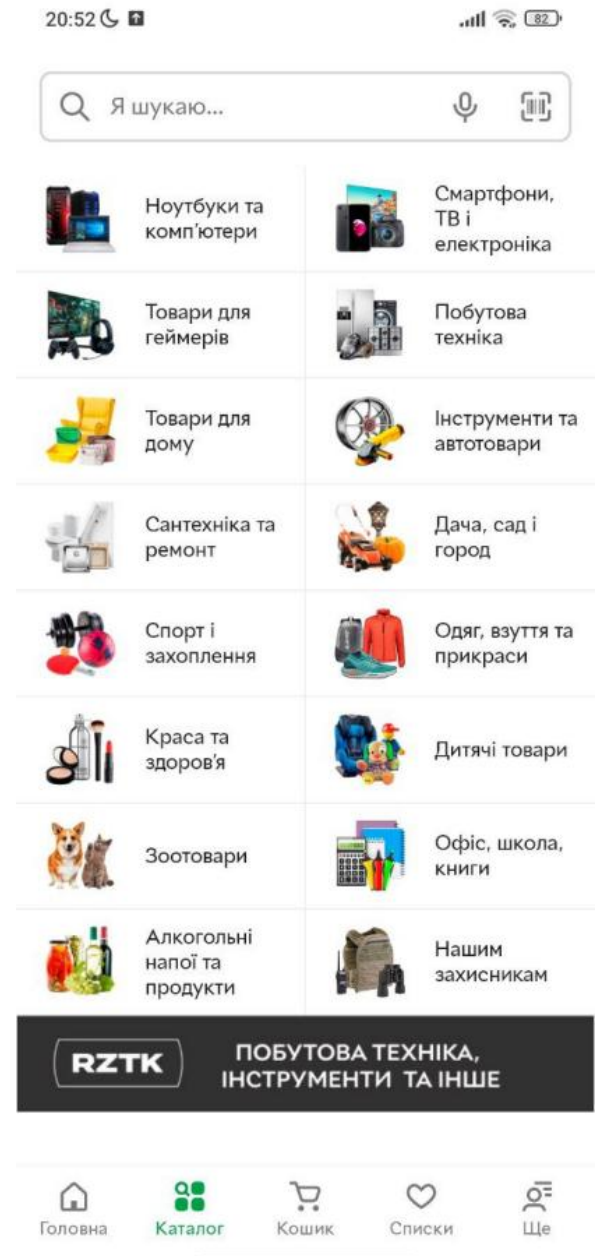


Рисунок 1.4 - Сторінка каталогу додатку "Rozetka".

Плюси:

- Великий вибір перевірених товарів.
- Детальний опис кожного товару.
- Регулярна перевірка стану товару.
- Швидка реакція магазину.
- Зручна доставка.

Мінуси:

- Вартість вище за ринкову, так як береться комісія з продавця.
- Жорсткі правила покупки та повернення для покупця та продавця.

Comfy [3] - Інтернет-магазин який пропонує велику кількість технічного товару. Замовити можна як у мобільному додатку так і у магазині (Рис. 1.5, 1.6).

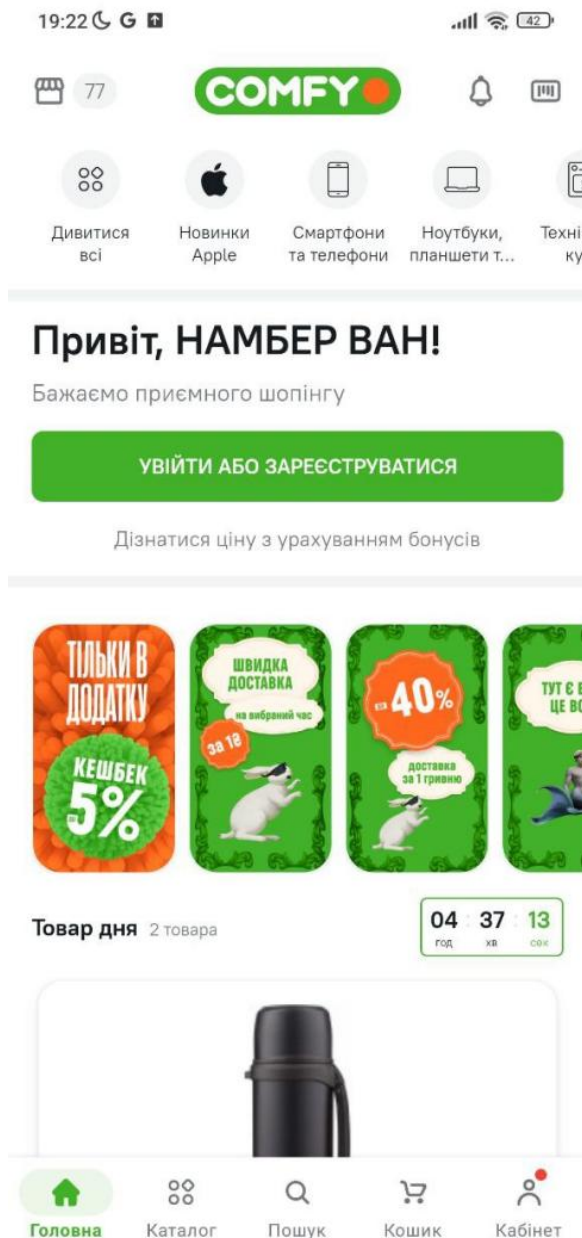


Рисунок 1.5 – Головна сторінка “Comfy”.

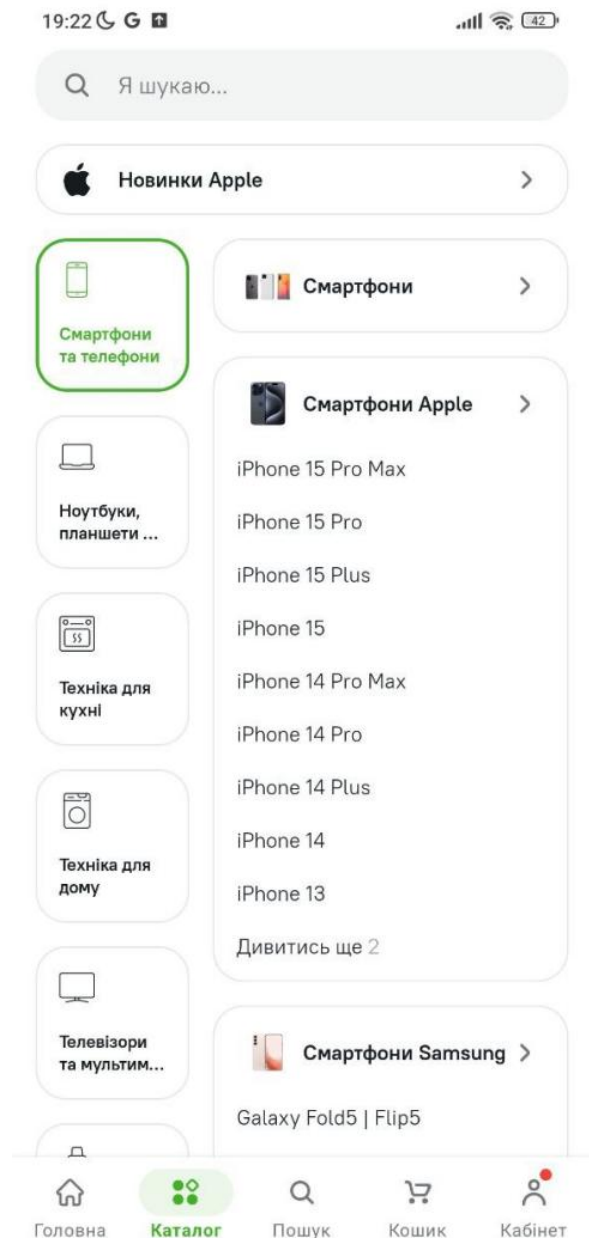


Рисунок 1.6 – Сторінка каталогу “Comfy”.

Плюси:

- Великий вибір товару та зручні фільтри за вибором.
- Детальний опис кожного продукту.
- Досвідчена підтримка.
- Підбір рекомендованих товарів.

Мінуси:

- Авторизація тільки одним методом.
- Тривала відповідь підтримки.

1.4. Постановка задачі

У результаті проведеного аналізу, поставлено мету роботи – проєктування та розробка мобільного додатку “E-commerce app” зі зручним та інтуїтивно зрозумілим інтерфейсом користувача та швидким доступом до всієї публічної інформації. Для досягнення даної мети потрібно завершити такі завдання:

- 1) створення макету згідно з дизайном;
- 2) додання функціоналу;
- 3) додання функціоналу оформлення та слідкування за замовленнями;
- 4) тестування працездатності функціоналу.

ВИБІР МЕТОДІВ РІШЕННЯ ЗАДАЧІ

2.1 Вибір технології

Для вибору технології розробки, в розгляд було взято новітні та сучасні технології. Для розробки додатку було обрано фреймворк React-Native [4].

React-Native - фреймворк працюючий на ядрі React, створений для розробки мобільних додатків для платформ Android та IOS. Мова написана на базі JavaScript, TypeScript, тим самим роблячи фреймворк дуже зручним та гнучким для розробки.

React [5] - Бібліотека написана на базі JavaScript, допомагає оновлювати контент на сайті без перезавантаження сторінки. Має дуже велику кількість допоміжних бібліотек які додають нових функціональних UI-компонентів, допомагають налаштувати зв'язок з сервером, та багато іншого.

React-Native зручний в написанні мобільних додатків своєю гнучкістю, оптимізованості та багато функціональності. На чистому фреймворку можна створити стандартні додатки, але завдяки популярності та адаптивності, розробники створили безліч бібліотек які додають новий функціонал у додаток, що розширило можливості в розробці. Завдяки гнучкості можливостей, фреймворк здатен створити мобільний додаток з різноманітною анімацією, складним функціоналом, інтеграцією з популярними системами та будь-якою архітектурою.

2.2 Проектування системи замовлення

Основним функціоналом додатку повинно бути керування та управління замовленнями. Для коректної роботи замовлень в додатку повинні бути такі сутності як:

- Клієнти - Користувачі додатку які заповнили контактні дані для оформлення та доставки замовлення;

- Товари - Об'єкт послуги або продукції які пропонує компанія клієнту для перегляду або покупки;

- Кошик - віртуальний аналог кошика товарів продукції, які користувач обирає та додає в процесі користування додатком;

- Замовлення - інформація про оформлене замовлення з даними про користувача, адресою доставки (якщо потрібно) та товарами і їх кількістю для доставки замовнику;

Користувачі користуючись додатком можуть переглянути послуги та товари компанії. Обравши потрібну пропозицію та варіацію (при наявності), вони додають його до кошика, який зберігає інформацію про пропозицію поки встановлений додаток або встановлений додаток на пристрій. Коли клієнт готовий оформити замовлення, у розділі кошик, він може перевірити правильність товарів які обрав, після чого переходить у розділ оформлення замовлення. Заповнивши всі необхідні поля, форма відправляється на сервер для подальшої обробки та оформлення замовлення клієнту згідно заповненої форми.

3. ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1. Опис процесу реалізації

При проектуванні додатку був обраний стек технологій що активно використовується при створенні подібного функціоналу та прекрасно себе зарекомендував серед розробників.

Вибір технологій для Frontend розробки:

Фреймворк: щоб отримати максимально оптимізацію під різні платформи було обрано React-Native

Навігація: React-navigation [6] - бібліотека дозволяє створювати гнучкі та різноманітні за використанням функції навігації.

Форми: для перевірки правильності заповнених даних було обрано react-hook-form. Популярна та гнучка бібліотека.

HTTP-запити: для здійснення HTTP-запитів до сервера було використано бібліотеку Axios [7].

State Management: щоб отримати максимально оптимізовану та функціональну систему сховища даних, було обрано бібліотеку redux/saga [8]. Бібліотека дозволяє налаштовувати та використовувати сховище різноманітним чином.

Дизайн: Для написання дизайну для мобільного додатку було обрано зручну бібліотеку [9].

Валідація форм: щоб користувач вводив коректні дані у необхідні поля було обрано бібліотеку для валідації React-Hook-Form [10].

Онлайн-оплата: для швидкої та зручної онлайн-оплати у додатку було обрано системи для онлайн-оплати Liqpay [11] та Stripe [12]

Збірка проекту: щоб отримати працюючий додаток ля встановлення на android було обрано програму Android Studio [13]

Процес реалізації:

Планування розробки: Спершу була проведена робота над архітектурою та оцінка обсягу проекту.

Налаштування: Ініціалізація проекту, налаштування середовища для розробки та встановлення основних бібліотек які використовуються у проекті.

Розробка frontend: Розробка інтерфейсу, функціоналу, інтеграція з API, написання документації.

Тестування: проведено тестування функціональних компонентів, робота та обробка помилок з API, перевірено кожен етап бізнес-логіки.

Публікація: Кінцевий результат потрібно завантажити на маркет-плейси Google play [14] та Appstore [15], з попереднім налаштуванням акаунтів.

3.2 Розробка зовнішнього вигляду системи

Головна сторінка (рис. 3.1).

Користувачі можуть переглядати публічну інформацію без додаткового входу в додаток. На головній сторінці розташовані товари та банери які зацікавлять користувача або покажуть потрібний йому товар, а якщо користувач увійшов в свій акаунт, у розділі “Рекомендації” йому будуть представлені товари які були підібрані з аналітики його історії замовлень та переглянутих товарів.

Код сторінки наведений у додатку А

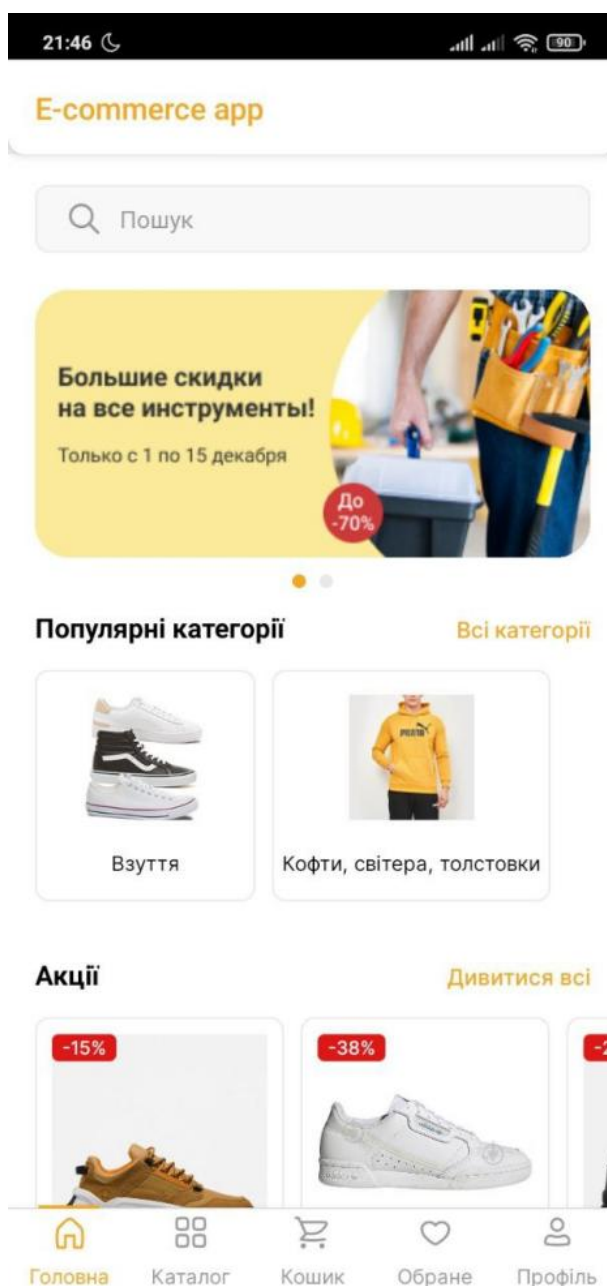


Рисунок 3.1 – Головна сторінка.

Сторінка каталогу (рис. 3.2).

Даний розділ містить в собі перелік категорій продукції, переходячи по яким користувач побачить товари які відсортовані згідно категорій (рис. 3.3). У списку користувач може побачити коротку інформацію про продукт (ціну, знижку, назву, фотографі) також, кнопки “Додати у кошик” та “Додати до обраного”. У цьому списку користувач може обрати тип сортування, перейти до фільтрів або переглянути сторінку продукт.

Код сторінки наведений у додатку Б

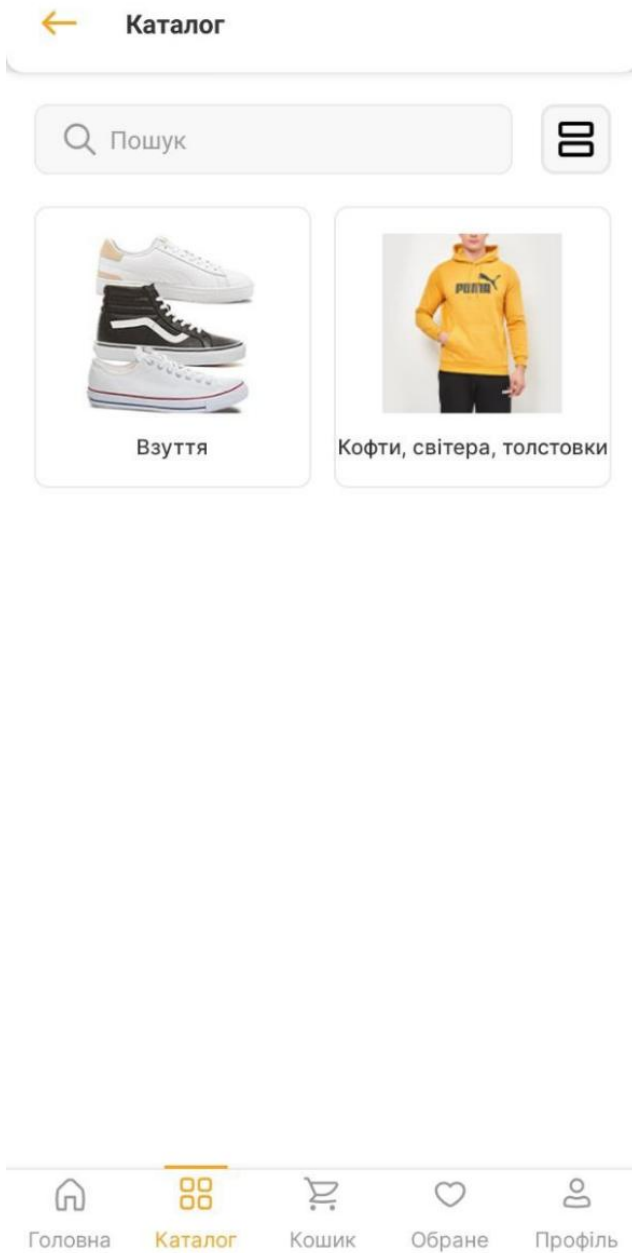


Рисунок 3.2 – Сторінка з списком категорій.

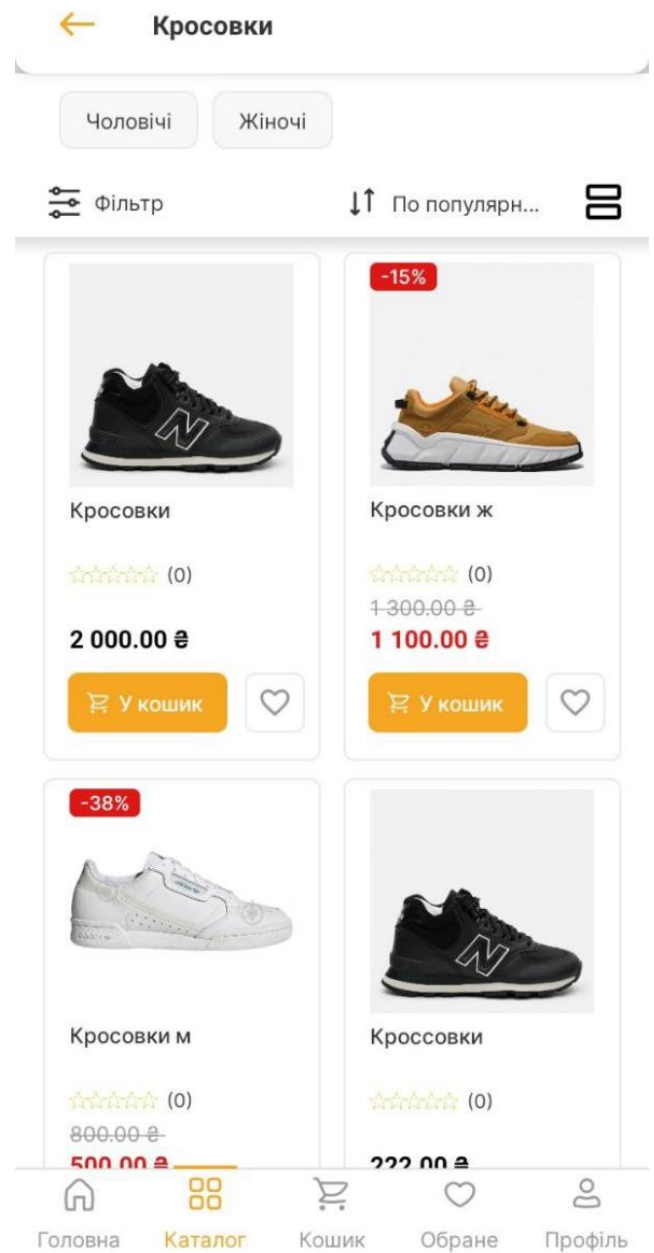


Рисунок 3.3 – Сторінка з списком продукції.

Сторінка продукту (рис. 3.4).

Тут знаходиться вся інформація про продукт : опис, галерея фотографій, характеристики, варіації, ціна, відгуки та бренд. Вся інформація оновлюється при заході на карточку. Також товаром можна поділитися у соцмережі, додати у кошик, додати у список порівняння, переглянути відгуки та додати до обраного. За бажанням можна написати коментар про товар (рис 3.5).

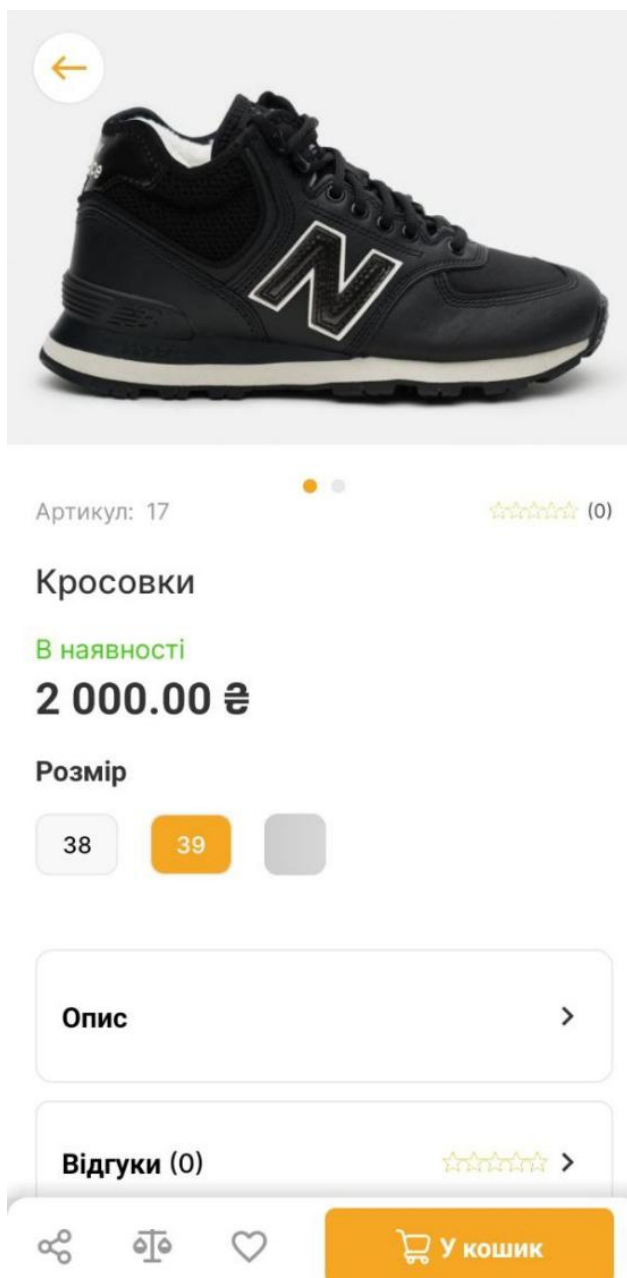


Рисунок 3.4 – Сторінка продукта.

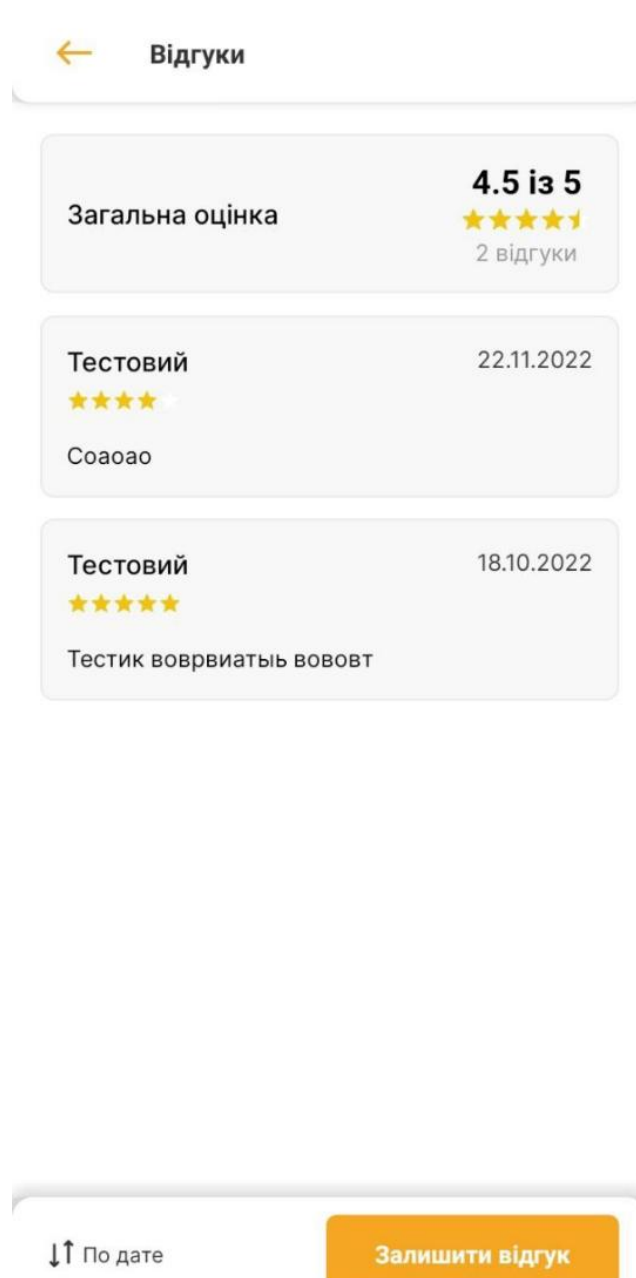


Рисунок 3.5 – Сторінка відгуків.

Сторінка кошика та оформлення замовлення.

На сторінці кошика (рис. 3.6) міститься інформація про товари, які користувач додав у свій кошик. Тут можна змінити кількість товару, змінити їх кількість та переглянути суму замовлення.

На сторінці оформлення замовлення (рис. 3.7) користувач повинен заповнити форму необхідну для коректної обробки його замовлення. Дані користувача (ім'я, прізвище та телефон), буде заповнений автоматично з його акаунту. Є можливість обрати “Іншого отримувача”, для цього потрібно заповнити поля “ім'я” та “телефон”. Обов'язковим є поле “доставка”, де користувач може обрати тип доставки замовлення (“Самовивіз”, “доставка Новою поштою” або кур'єр”). В залежності яких тип доставки був обраний, покажуться поля які будуть обов'язкові до заповнення. Спосіб оплати також присутній у формі, у користувача є 2 варіанти: онлайн-оплата, готівка. Також присутнє поле “коментар” де користувач може залишити повідомлення для адміністратора.

Додаток не дозволить оформити замовлення з некоректними даними, і покаже помилку під полем яке треба відредагувати.

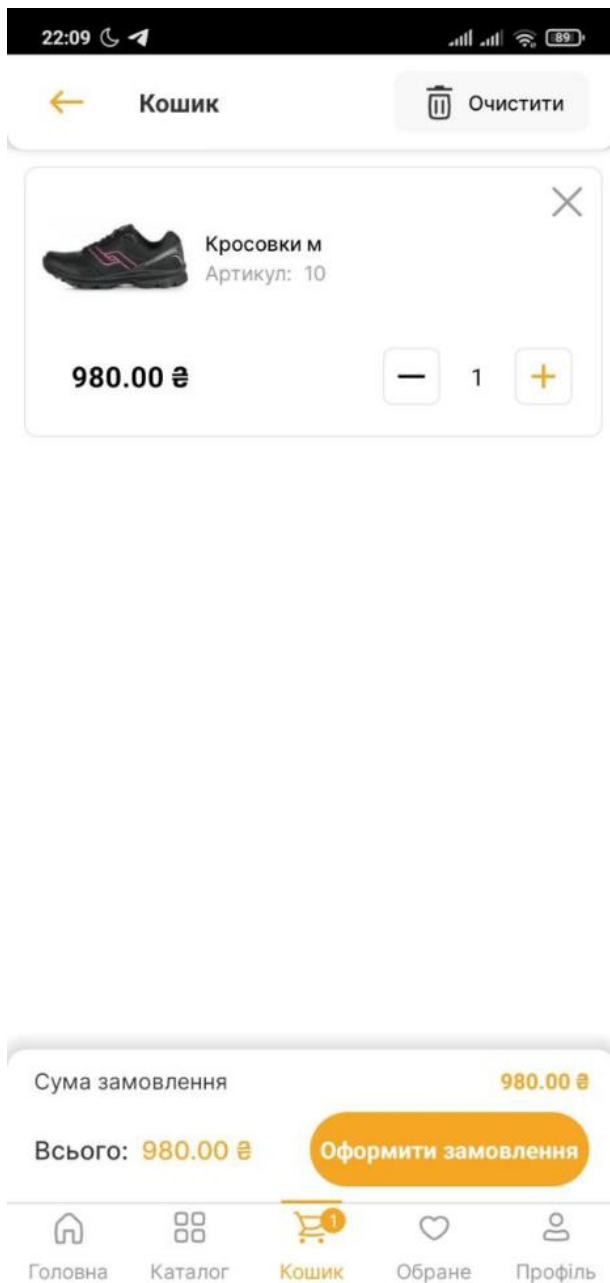


Рисунок 3.6 – Сторінка кошика.

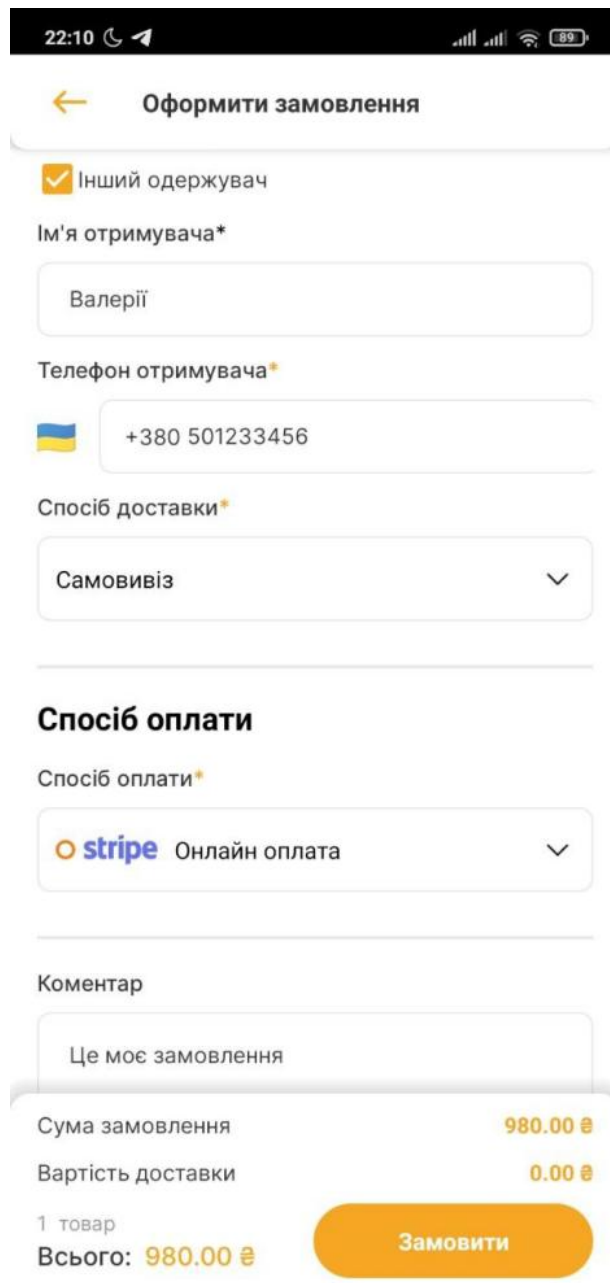


Рисунок 3.7 – Сторінка оформлення замовлення.

Сторінка “Обране” (рис. 3.8).

На сторінці користувач може переглянути ті товари, які додав в процесі перегляду товарів у каталозі. Даний список пов’язаний з акаунтом, тому коли користувач вийде з профілю і увійде на іншому пристрої, список збережеться.

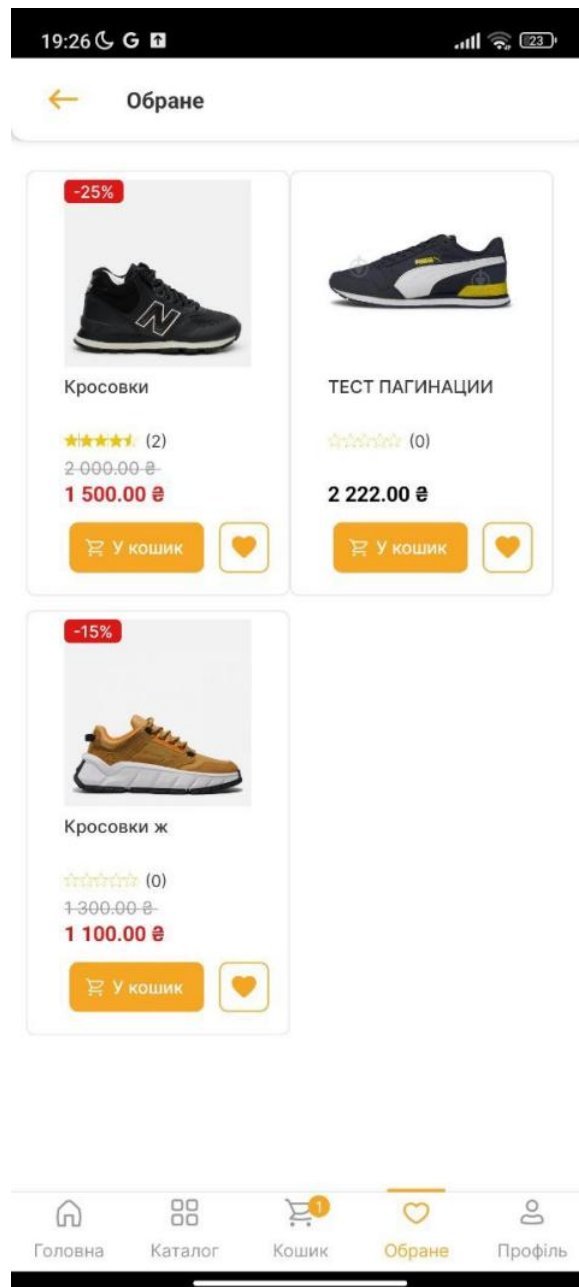


Рисунок 3.8 – Сторінка “Обране”.

Сторінка Профілю (рис 3.9, 3.10).

Головна сторінка у додатку для взаємодії з користувачем. У Цьому розділі користувач може увійти або зареєструвати акаунт, написати у підтримку, контролювати свої замовлення, переглянути список порівняння та список очікування, переглядати інформацію профілю та змінювати її.

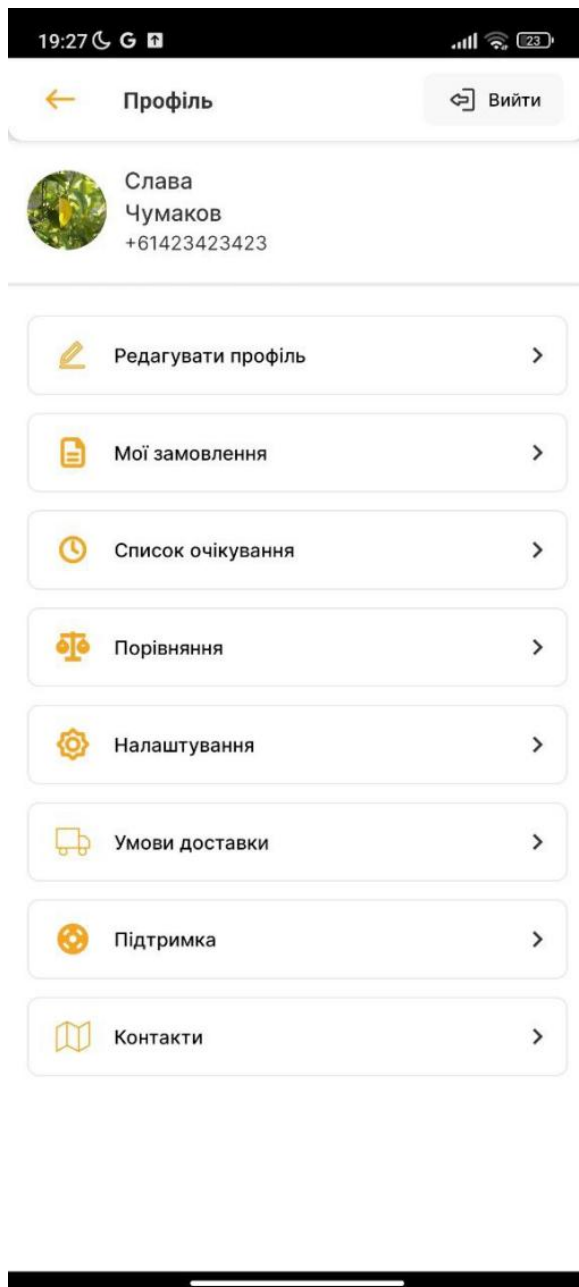


Рисунок 3.9 – Сторінка авторизованого користувача.

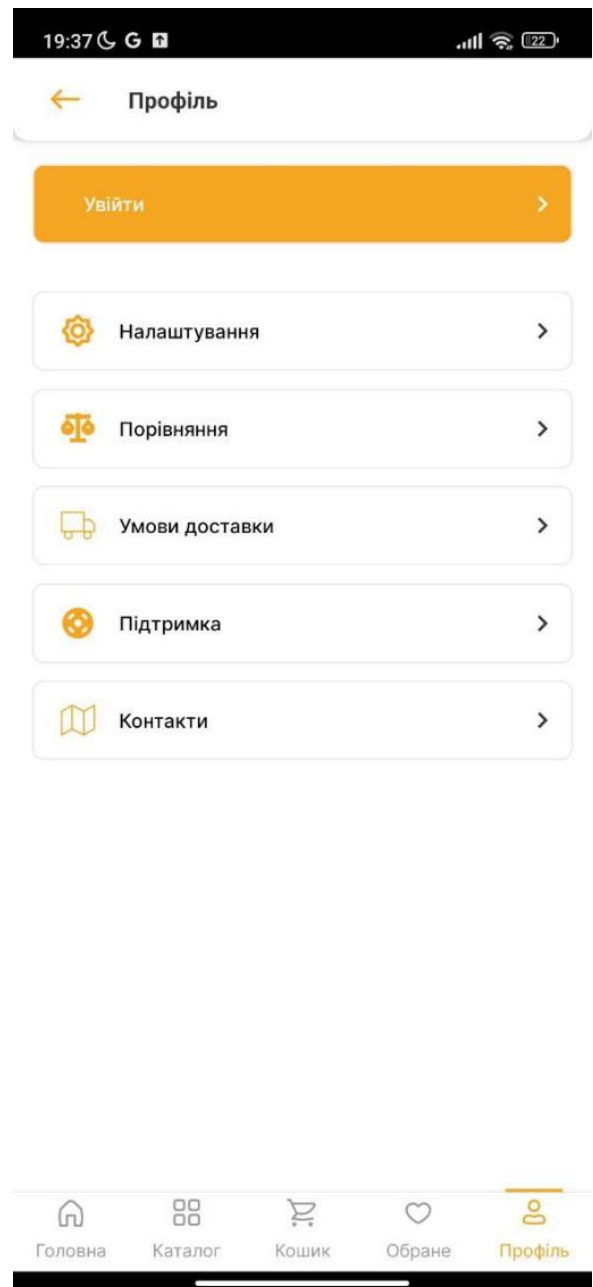
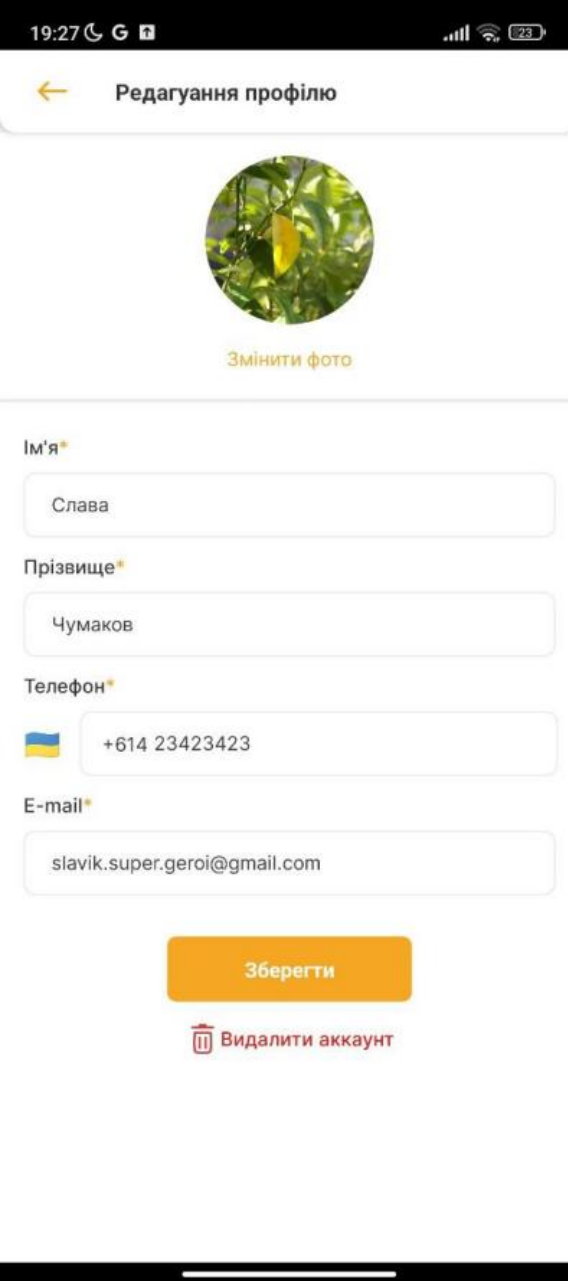


Рисунок 3.10 – Сторінка неавторизованого користувача.

Сторінка редагування Профілю (рис 3.11).

На даній сторінці користувач може переглянути інформацію щодо свого акаунту, також може редагувати її. У випадку якщо інформація введена некоректно, додаток покаже яке саме поле було введене не правильно, і напише повідомлення. Також наявна кнопка “Видалити акаунт”, натиснувши,

сервер видаляє дані про акаунт, і при повторному вході, потрібно все заповнювати знову.



19:27 100% 23

← Редагування профілю

Змінити фото

Ім'я*

Слава

Прізвище*

Чумаков

Телефон*

+614 23423423

E-mail*

slavik.super.geroi@gmail.com

Зберегти

Видалити акаунт

Рисунок 3.11 – Сторінка редагування Профілю.

Сторінка замовлень.

Інформація просто поточні та завершені замовлення знаходиться на сторінці “Мої замовлення” (рис. 3.12). Тут можна переглянути статус замовлення, його номер та загальну вартість. Натиснувши на “Більше”, можна переглянути всі деталі оформленого замовлення: товари, інформацію

про доставку, оплату, окрема вартість доставки, товарів та загальну (рис. 3.13). Якщо замовлення було оформлене щ онлайн-оплатою, то буде наявна кнопка сплати, яка відправить користувача на форму оплати, у разі успішної оплати, чек можна буде відправили на вказану пошту у формі.

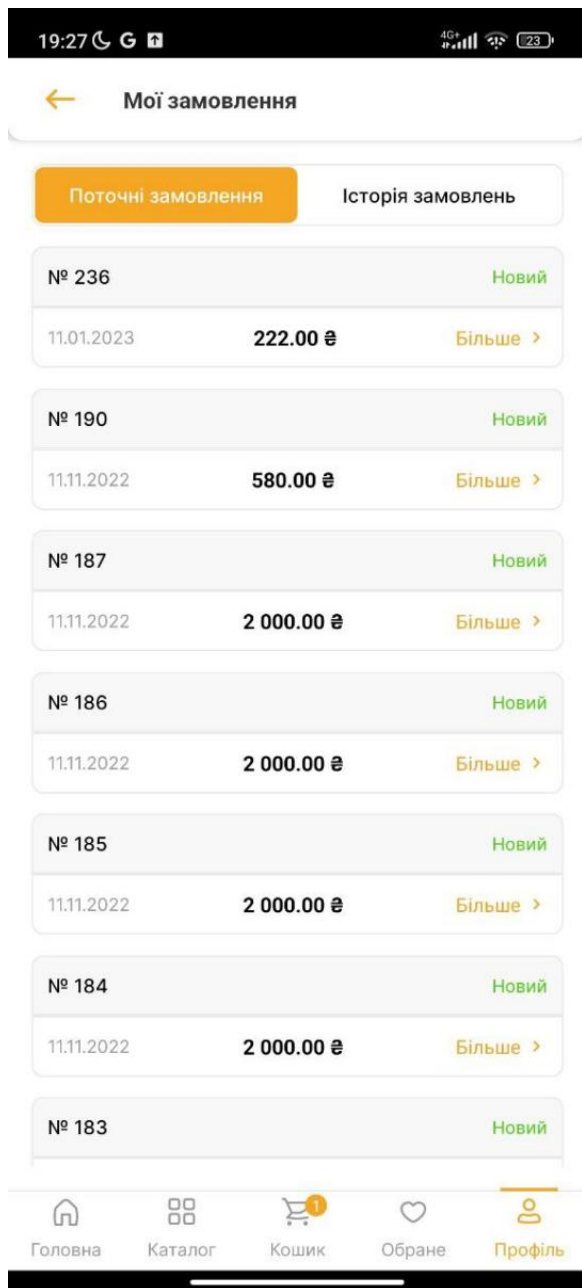


Рисунок 3.12 – Сторінка “Мої замовлення”.

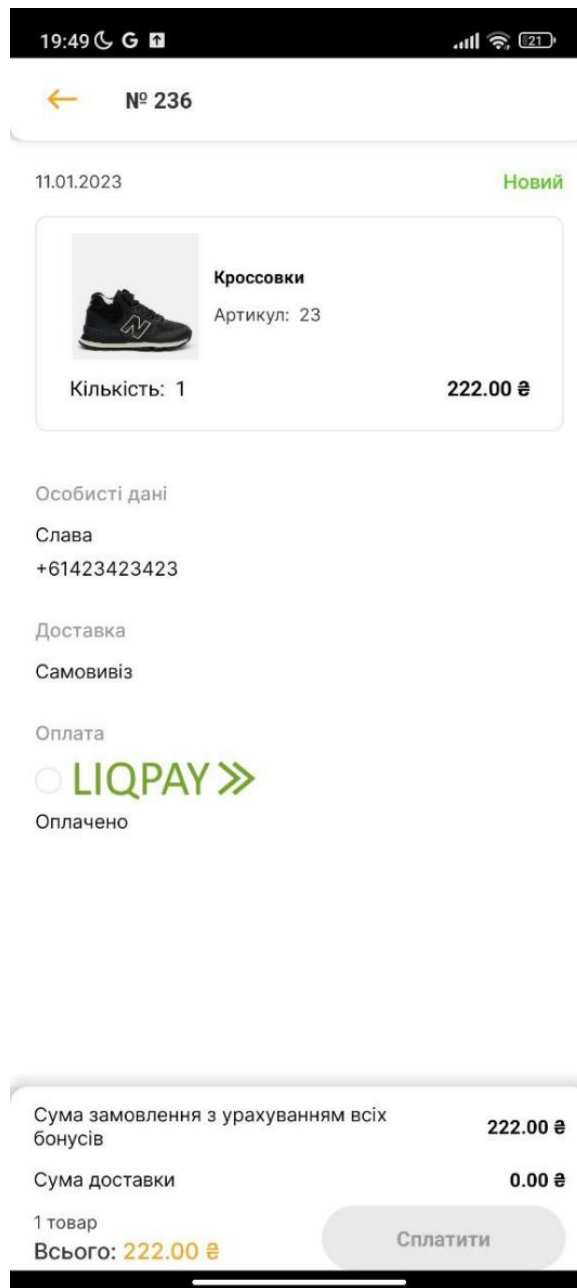


Рисунок 3.13 – Сторінка “Деталі замовлення”.

Сторінка Налаштувань (рис. 3.14).

Ця сторінка зроблена для зміни мови у додатку та вимкнення пуш-повідомлень, у разі якщо користувач не бажає їх отримувати.



Рисунок 3.14 - Сторінка “Налаштувань”.

Сторінка “Підтримки” (рис 3.15).

Ця сторінка зроблена для зв'язку з адміністраторами. Після заповнення форми, лист відправляється адміністратору, і у разі потреби, адміністратор зв'яжеться з користувач за для вирішення заявленого питання.

Рисунок 3.15 - Сторінка “Підтримка”.

Сторінка Авторизації (рис. 3.16).

На сторінці авторизації, користувач може увійти або створити акаунт для використання особливих функцій додатку (обране, кошик, замовлення, профіль). У користувача є вибір у методах авторизації, а саме: Google акаунт, Facebook акаунт, номер телефону і якщо користувач використовує пристрій з операційною системою “IOS”, то буде доступний метод “Apple ID”. Після успішної авторизації, додаток перевіряє наявність акаунту, у випадку наявності, користувач може продовжити використання, у іншому випадку, йому потрібно заповнити інформацію про профіль. Якщо користувач авторизувався за номером телефону, то на цей номер буде відправлене СМС-повідомлення з 6-ним кодом для верифікації, після чого буде перевірка наявності акаунту.

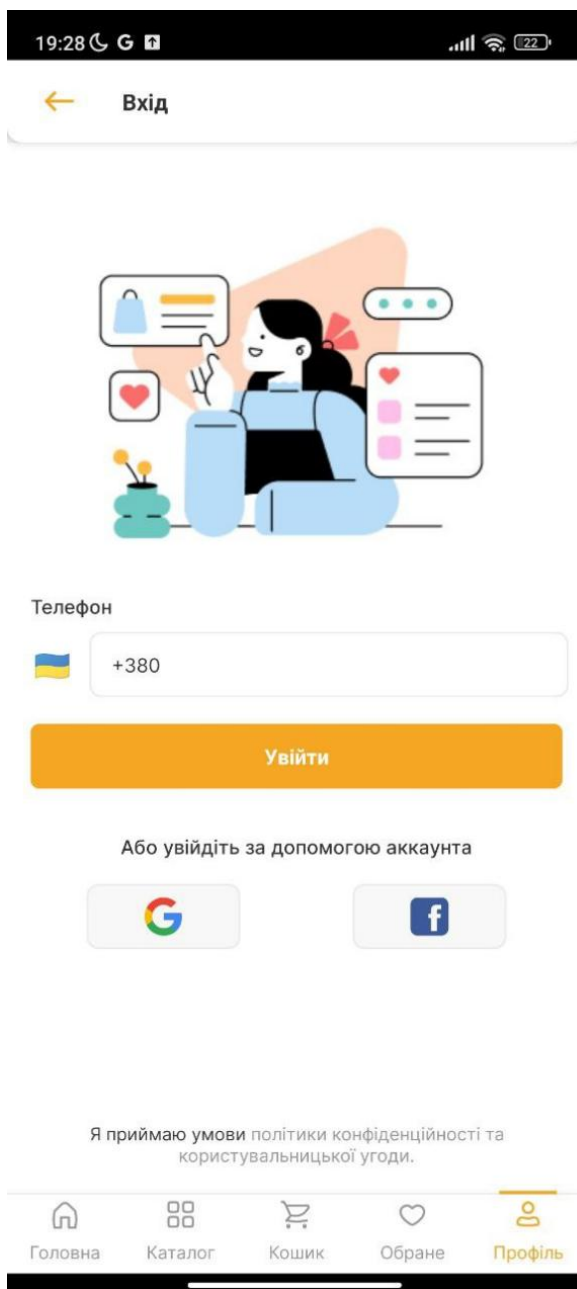


Рисунок 3.16 - Сторінка "Авторизація".

Таким чином у користувача є декілька зручних методів для входу у додаток, і він може обрати той який є зручнішим.

ВИСНОВОК

В ході роботи було розроблено мобільний додаток для управління замовленнями, що реалізує цілісний підхід до обробки та доставки товарів.

Макет зроблений з урахуванням мінімалістичного стилю для звільнення місця і для інтуїтивного розуміння користувачем. Також був доданий функціонал отримання та оперування товарами та категоріями. Доданий функціонал оформлення замовлення та подальше оперування, слідкування та оплата замовлення. Все це було протестоване у вигляді “реального” користування.

При виборі технології було вирішено застосувати фреймворк React-Native завдяки його гнучкості, оптимізації та можливості до праці на платформах Android та iOS. За допомогою цього інструменту було створено додаток, який включає сутності, такі як клієнти, товари, кошик і інформація про замовлення.

Користувачам надано можливість переглядати товари компанії, додавати їх до кошика та робити замовлення. Всі ці функції реалізовані таким чином, щоб забезпечити максимальну зручність та ефективність для користувача.

Було вирішено численні технічні завдання, зокрема розробка інтерфейсу, інтеграція API, написання документації та тестування. Результатом стало створення додатку, готового до завантаження на таких ринках як Google Play та AppStore.

Отже, в результаті роботи було розроблено повністю робочий мобільний додаток для ведення електронної комерції з можливістю створення онлайн замовлень та їх онлайн оплатою, а також слідкуванням за їх виконання

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Prom.ua - Інтернет-магазин -
<https://play.google.com/store/apps/details?id=ua.prom.b2c>
2. Rozetka - Інтернет-магазин -
<https://play.google.com/store/apps/details?id=ua.com.rozetka.shop>
3. Comfi - Інтернет-магазин онлайн -
<https://play.google.com/store/apps/details?id=ua.comfy.comfystore>
4. React-Native - <https://reactnative.dev/docs/getting-started>
5. React - <https://react.dev/>
6. React-navigation - <https://reactnavigation.org/docs/getting-started/>
7. Axios Getting Started - <https://axios-http.com/docs/intro>
8. Redux/Saga - <https://redux-saga.js.org/docs/introduction/GettingStarted>
9. Styled-components - <https://styled-components.com/docs>
10. React-Hook-Form Getting started - <https://www.react-hook-form.com/get-started>
11. Liqpay - <https://www.liqpay.ua/en/documentation/sdk>
12. Stripe - <https://stripe.com/>
13. Android Studio - <https://developer.android.com/studio>
14. Google Play - <https://play.google.com/store/apps?hl=ua>
15. Appstore - <https://apps.apple.com/ru/developer/apple/id284417353?mt=12>

ДОДАТКИ

ДОДАТОК А

Головна сторінка

```
import React, { FC, useEffect } from 'react'
import { Padding, Wrapper } from './styled'
import { useDispatch } from 'react-redux'
import { useTypedSelector } from '../../store/store'
import { getSectionHomeAction } from '../../store/section'
import {
  getCurrentColorAction,
  getFaviryteAction,
} from '../../store/profile'
import HomeSlider from './components/HomeSlider/HomeSlider'
import {
  getBanerListAction,
  getDiscountProductListAction,
  getRecomendedProductAction,
  getSeenProductAction,
} from '../../store/product'

import { strings } from '../../utils/string'
import SearchBlock from './components/SearchBlock/SearchBlock'
import HomeProduct from './components/HomeProduct/HomeProduct'
import HomeCategory from './components/HomeCategory/HomeCategory'
import { HOST } from '@env'
import { Header } from '@modules'
```

```
const Home: FC = ({}) => {
  const dispatch = useDispatch()

  const lang = useTypedSelector(s => s.profile.lang)

  const token = useTypedSelector(s => s.auth.token)

  useEffect(() => {
    dispatch(getSectionHomeAction.request({}))
    dispatch(getFaviryteAction.request({}))
    dispatch(getDiscountProductListAction.request({}))
    dispatch(getBanerListAction.request({}))
    dispatch(getSeenProductAction.request({}))
    dispatch(getRecomendedProductAction.request({}))

    dispatch(getCurrentColorAction.request({}))
  }, [strings.getLanguage(), lang, token])

  const seenProduct = useTypedSelector(s => s.product.seenProduct?.data)

  const recomemdedProduct = useTypedSelector(
    s => s.product.recomemdedProduct?.data,
  ).filter(el => el.hidden !== true)

  const BannerList = useTypedSelector(s => s.product.baner)

  const DiscountProduct = useTypedSelector(
    s => s.product.discount?.data,
  ).filter(el => el.hidden !== true)

  const RecomendCategory = useTypedSelector(s => s.section.sectionHome?.data)

  return (
    <>
      <Header.Home />
      <Wrapper bounces={false} showsVerticalScrollIndicator={false}>
        <Padding>
          <SearchBlock />
        </Padding>

        {BannerList && <HomeSlider data={BannerList.data} />}
    </>
  )
}
```

```
<HomeCategory data={RecommendCategory} />

{token && seenProduct && seenProduct.length != 0 && (
  <HomeProduct
    title={strings.screens.homeStack.homeStack.seen.title}
    data={seenProduct}
    linkText={strings.screens.homeStack.homeStack.seen.link}
    linkNav={'seen'}
  />
)}

{token && recomemdedProduct && recomemdedProduct.length != 0 && (
  <HomeProduct
    title={strings.screens.homeStack.homeStack.recommended.title}
    data={recomemdedProduct}
    linkText={strings.screens.homeStack.homeStack.recommended.link}
    linkNav={'recommended'}
  />
)}

<HomeProduct
  title={strings.screens.homeStack.homeStack.discount.title}
  data={DiscountProduct!}
  linkNav={'discount'}
  linkText={strings.screens.homeStack.homeStack.seen.link}
/>
</Wrapper>
</>
)
}
export default Home
```

ДОДАТОК Б

Сторінка каталогу продукції

```

import React, { FC, useEffect, useState } from 'react'
import { FlatList, ListRenderItem, StyleSheet } from 'react-native'
import { CatalogBlock, CatalogListBlock, Wrapper } from './styled'
import { TRenderCategoryItem } from './types'
import { useIsFocused } from '@react-navigation/native'
import { useDispatch } from 'react-redux'
import { useTypedSelector } from '../../../store/store'
import { getSectionAction, startLoaderAction } from '../../../store/section'
import { strings } from '../../../utils/string'
import HomeCategory from '../components/CatalogHeder/CatalogHeder'
import CatalogItem from '../components/CatalogItem/CatalogItem'
import { useNavigation } from '@common/hooks'
import { EScreens } from '@navigations/screens'
import { Header } from '@modules'
import { useTranslation } from 'react-i18next'

const Catalog: FC = ({}) => {
  const { navigate } = useNavigation()
  const { t } = useTranslation()
  const [visible, setVisible] = useState(false)

  const isFocused = useIsFocused()

  const dispatch = useDispatch()

  useEffect(() => {
    dispatch(getSectionAction.request({}))
  }, [isFocused, strings])

  const RecomendCategory = useTypedSelector(s => s.section.catalog?.data)

  const onSetVisible = () => {
    setVisible(!visible)
  }

  const onGoToSecCat = (title: string, id: string) => {
    startLoaderAction()

    navigate(EScreens.CatalogSecond, {
      title,
      id,
      isList: visible,
    })
  }

  const renderItem: ListRenderItem<TRenderCategoryItem> = ({ item }) => {
    return (
      <CatalogItem
        {...item}
        photoUri={'section'}
        mode={'block'}
        onPress={onGoToSecCat}
      />
    )
  }

  const renderItem: ListRenderItem<TRenderCategoryItem> = ({ item }) => {
    return (
      <CatalogItem
        {...item}
        photoUri={'section'}
        mode={'list'}
        onPress={onGoToSecCat}
      />
    )
  }

  return (
    <>

```

```

<Header.Standart title={t('catalog')} goBack />
<Wrapper>
  <HomeCategory visible={visible} onSetVisible={onSetVisible} />

  {visible ? (
    <CatalogListBlock>
      <FlatList
        data={RecomendCategory}
        style={style.flatList}
        bounces={false}
        keyExtractor={item => item._id}
        renderItem={renderListItem}
        showsVerticalScrollIndicator={false}
      />
    </CatalogListBlock>
  ) : (
    <CatalogBlock>
      <FlatList
        numColumns={2}
        columnWrapperStyle={style.wrapper}
        data={RecomendCategory}
        style={style.flatList}
        bounces={false}
        keyExtractor={item => item._id}
        renderItem={renderItem}
        showsVerticalScrollIndicator={false}
      />
    </CatalogBlock>
  )}
</Wrapper>
</>
)
}

const style = StyleSheet.create({
  flatList: {
    width: '100%',
    height: '100%',
  },
  wrapper: {
    justifyContent: 'space-between',
  },
})
export default Catalog

```