

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Острозька академія»
Навчально-науковий інститут інформаційних технологій та бізнесу
Кафедра інформаційних технологій та аналітики даних

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня бакалавра

на тему: **«Розробка дизайн системи та клієнтської частини маркетплейсу
вживаних автомобілів»**

Виконала: студентка 4 курсу, групи КН-41
першого (бакалаврського) рівня вищої освіти
спеціальності 122 Комп'ютерні науки
освітньо-професійної програми «Комп'ютерні науки»
Гаврильчик Анастасія Юріївна

Керівник: викладач кафедри інформаційних технологій та
аналітики даних
Мацевич Денис Володимирович

Рецензент: Розробник програмного забезпечення в ТОВ
ОБНОВА-ЄВРОШОП, викладач кафедри інформаційних
технологій та аналітики даних НаУОА
Шведюк Володимир Володимирович

РОБОТА ДОПУЩЕНА ДО ЗАХИСТУ

Завідувач кафедри інформаційних технологій та аналітики даних
(проф., д.е.н. Кривицька О.Р.)
Протокол №10 від 28 травня 2025 р.

Острог, 2025

АНОТАЦІЯ
кваліфікаційної роботи
на здобуття освітнього ступеня бакалавра

Тема: *Розробка дизайн системи та клієнтської частини маркетплейсу вживаних автомобілів*

Автор: *Гаврильчик Анастасія Юріївна*

Науковий керівник: *викладач-стажист кафедри ІТАД, Мацевич Денис Володимирович*

Захищена «.....»..... 20__ року.

Пояснювальна записка до кваліфікаційної роботи: *103 с., 58 рис., 0 табл., 1 додаток, 15 джерел.*

Ключові слова: *дизайн, дизайн-система, маркетплейс, вживані автомобілі, клієнтська частина, інтерфейс користувача, адаптивний дизайн*

Короткий зміст праці:

Кваліфікаційна робота присвячена розробці дизайн-системи та клієнтської частини маркетплейсу вживаних автомобілів. У роботі розглянуто основні принципи побудови дизайн-систем, їх значення для уніфікації інтерфейсу та прискорення розробки. Описано процес створення компонентів дизайн-системи, включаючи типографіку, колірні схеми, іконографію та інтерактивні елементи. Також детально висвітлено етапи проєктування та реалізації клієнтської частини маркетплейсу, що забезпечує інтуїтивно зрозумілий та функціональний інтерфейс для користувачів, які шукають або продають вживані автомобілі.

ANNOTATION
of qualification paper
for bachelor's degree

Theme: Development of a Design System and Client-Side for a Used Car Marketplace

Author: Havrylchyk Anastasiia

Scientific supervisor: Lecturer-Intern of the Department of ITAD, Matsevykh Denys Volodymyrovych

Defended: «.....»..... 20__ year.

Explanatory note to the qualification work: 103 pages, 58 figures, 0 tables, 1 appendix, 15 sources.

Keywords: design, design system, marketplace, used cars, client-side, user interface, responsive design

Abstract:

The qualification work is dedicated to the development of a design system and the client-side for a used car marketplace. The work examines the fundamental principles of building design systems, their importance for interface unification and accelerated development. The process of creating design system components, including typography, color schemes, iconography, and interactive elements, is described. The stages of designing and implementing the client-side of the marketplace are also thoroughly covered, ensuring an intuitive and functional interface for users looking to buy or sell used cars.

Зміст

ВСТУП.....	4
РОЗДІЛ 1	
АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПРОЄКТУВАННЯ МАРКЕТПЛЕЙСУ	
ВЖИВАНИХ АВТОМОБІЛІВ.....	6
1.1 Аналіз стану ринку вживаних автомобілів в Україні.....	6
1.2 Роль онлайн-маркетплейсів у сфері продажу вживаних автомобілів.....	8
1.3 Дослідження наявних онлайн-платформ та їх функціональних особливостей..	9
1.4 Визначення цільової аудиторії та її потреб.....	13
1.5 Формування вимог до проєктованого маркетплейса.....	15
Висновки до розділу.....	17
РОЗДІЛ 2	
ТЕОРЕТИЧНІ ОСНОВИ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ ДИЗАЙН-СИСТЕМИ	
МАРКЕТПЛЕЙСУ ВЖИВАНИХ АВТОМОБІЛІВ.....	18
2.1 Середовище розробки та організація процесу на GitHub.....	18
2.2 Теорія та розробка кольорової схеми й color token системи.....	26
2.3 Типографія в інтерфейсі: теоретичні аспекти та практична реалізація.....	38
2.4 Створення інфографічних елементів та іконографіки для маркетплейса.....	43
Висновок до розділу.....	46
РОЗДІЛ 3	
РОЗРОБКА КЛЮЧОВИХ СТОРІНОК МАРКЕТПЛЕЙСУ.....	47
3.1 Аналіз референсів та створення moodboard.....	47
3.2 Проєктування та реалізація сторінки каталогу вживаних авто.....	49
3.3 Розробка дизайну сторінки деталей оголошення.....	58
3.4 Проєктування сторінки профілю користувача.....	61
3.5 Проєктування сторінок додавання та оновлення оголошень.....	66
Висновок до розділу.....	79
РОЗДІЛ 4	
ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОЄКТУ.....	81
4.1 Технології та фреймворки для реалізації клієнтської частини.....	81
4.2 Реалізація сторінки профілю користувача з використанням Angular.....	83
4.3 Розробка компонентів для фільтрації та пошуку авто.....	90
Висновок до розділу.....	97
ВИСНОВКИ.....	98
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	100
ДОДАТКИ.....	102

ВСТУП

В епоху цифрової трансформації автомобільний ринок активно переміщується на онлайн-платформи, які стають основним способом взаємодії між покупцями та продавцями вживаних автомобілів. Маркетплейси вторинного авторинку перетворюються на складні інформаційні системи, де якісний дизайн та функціональний клієнтський інтерфейс відіграють вирішальну роль у формуванні позитивного користувацького досвіду. Сучасні платформи мають не лише виконувати посередницьку функцію, але й забезпечувати привабливий, інтуїтивно зрозумілий та зручний інтерфейс, що сприяє ефективній взаємодії користувачів із контентом.

Актуальність теми обумовлена високим попитом на вживані автомобілі та необхідністю створення конкурентоспроможних онлайн-платформ, що відповідають сучасним очікуванням користувачів. В умовах високої конкуренції на ринку маркетплейсів ключовим фактором успіху стає не лише функціональність, але й продуманий дизайн системи, що забезпечує легку навігацію, швидке сприйняття інформації та естетичне задоволення. Професійно розроблена дизайн-система та клієнтська частина дозволяють підвищити рівень довіри до платформи, збільшити час перебування користувачів на сайті та сприяти зростанню конверсії.

Мета дослідження – розробка комплексної дизайн-системи та клієнтської частини маркетплейсу вживаних автомобілів, що забезпечить високу функціональність, естетичну привабливість та зручність користування платформою.

Задачі дослідження:

- аналіз предметної області, включаючи дослідження ринку вживаних автомобілів, визначення потреб цільової аудиторії та вивчення успішних практик у розробці маркетплейсів;

- проектування інформаційної системи, що включає розробку архітектури, визначення основних компонентів дизайн-системи (кольорової схеми, типографіки, інфографіки) та створення прототипів інтерфейсу;
- вибір інструментарію, методів реалізації та тестування програмного продукту, включаючи сучасні фреймворки та бібліотеки для розробки клієнтської частини.

Об'єкт дослідження – інформаційна система маркетплейсу вживаних автомобілів, що поєднує функціональну клієнтську частину та цілісну дизайн-систему.

Предмет дослідження – сучасні інструментальні засоби, методи та технології проектування дизайн-систем та розробки клієнтської частини вебплатформ, включаючи принципи дизайну та адаптивну верстку.

Методи дослідження включають системний аналіз наявних рішень, порівняльний аналіз технологій та інструментів, прототипування, юзабіліті-тестування та евристичну оцінку інтерфейсів.

Розробка якісної дизайн-системи та клієнтської частини маркетплейсу дозволить створити платформу, що відрізняється високим рівнем зручності користування, естетичною привабливістю та ефективністю взаємодії користувачів із системою, що, своєю чергою, сприятиме підвищенню конкурентоспроможності та комерційного успіху проєкту на ринку онлайн-платформ з продажу вживаних автомобілів.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПРОЄКТУВАННЯ МАРКЕТПЛЕЙСУ ВЖИВАНИХ АВТОМОБІЛІВ

1.1. Аналіз стану ринку вживаних автомобілів в Україні

У 2023 році ринок вживаних автомобілів в Україні зазнав суттєвих змін, що відображають як внутрішні економічні процеси, так і вплив зовнішніх чинників. За даними Інституту досліджень авторинку, сумарний обсяг вторинного ринку легкових автомобілів склав 1,177 млн одиниць, що на 5,5% більше, ніж у 2022 році [1]. Динаміка продажів у 2022 році демонструвала різке падіння обсягів, що було безпосередньо пов'язано з початком повномасштабної війни в Україні. На графіку обсягів продажу за 2022 рік (див. Рис. 1.1.) чітко видно цю тенденцію: синім кольором позначено імпорт вживаних авто, а помаранчевим – внутрішні перепродажі.

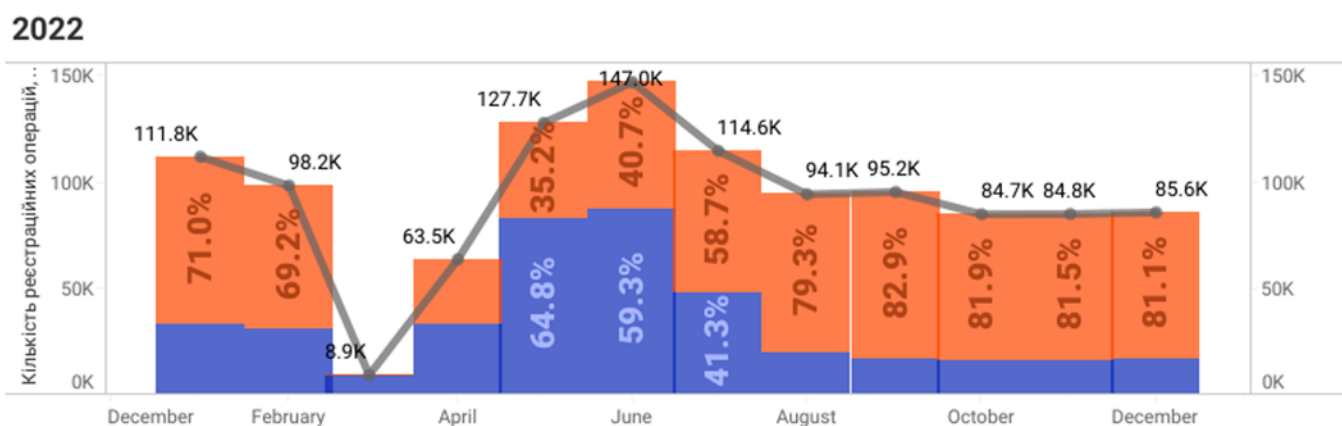


Рис. 1.1. Динаміка обсягів продажу на вторинному ринку легкових автомобілів у 2022 році

Джерело:

<https://eauto.org.ua/en/news/480-ukrainian-market-of-passenger-used-cars-results-of-2022>

3

У 2023 році на внутрішньому ринку було укладено 955,6 тис. угод купівлі-продажу, що на 34,3% більше, ніж у 2022 році [1]. Водночас імпорт вживаних автомобілів значно скоротився: перших реєстрацій легковиків з пробігом

зафіксовано 220,9 тис., що майже удвічі менше (на 45,3%) порівняно з 2022 роком, коли було імпортовано 403,9 тис. авто [2][3]. Така динаміка пояснюється дією пільгового режиму ввезення автомобілів у 2022 році, коли з квітня по липень діяло «нульове розмитнення», що сприяло різкому зростанню імпорту. У 2023 році подібних законодавчих стимулів не було, тому обсяг імпорту повернувся до звичних для ринку показників. Як видно з графіка за 2023 рік (див. Рис. 1.2.), структура ринку знову змістилася на користь внутрішніх перепродажів, тоді як частка імпорту знизилася.

2023

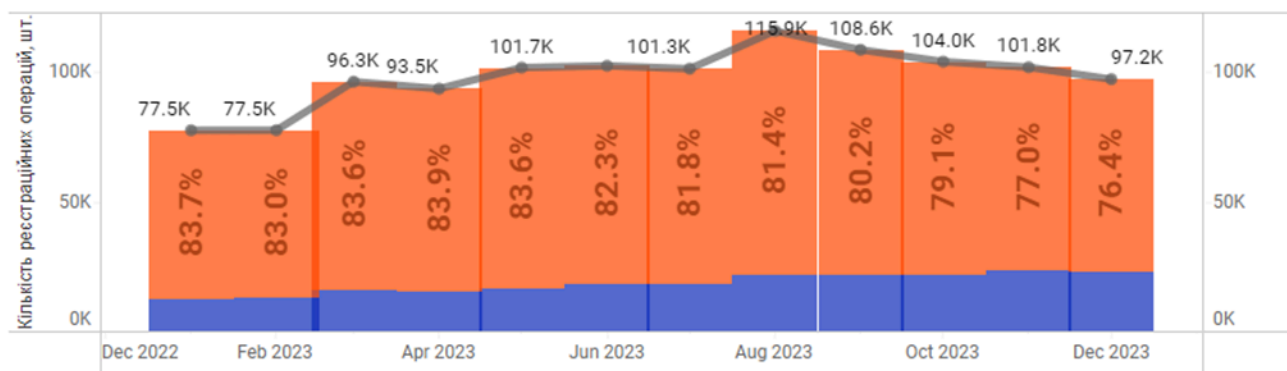


Рис. 1.2. Динаміка обсягів продажу на вторинному ринку легкових автомобілів у 2023 році

Джерело:

<https://eauto.org.ua/en/news/480-ukrainian-market-of-passenger-used-cars-results-of-2023>

3

Варто зазначити, що попит на вживані автомобілі в Україні традиційно значно перевищує попит на нові: у 2023 році вживаних авто було продано у 3,5 раза більше, ніж нових [3]. Таким чином, ринок вживаних автомобілів у 2023 році продемонстрував поступове відновлення після шокового падіння у 2022 році, зростання внутрішніх перепродажів та зменшення імпорту через відсутність податкових пільг. Структура ринку залишається стабільною, а попит концентрується на доступних і надійних моделях європейського виробництва.

1.2. Роль онлайн-маркетплейсів у сфері продажу вживаних автомобілів

Онлайн-маркетплейси займають провідне місце на сучасному ринку вживаних автомобілів, виконуючи роль ефективного посередника між продавцями та покупцями. Завдяки цифровим платформам процес купівлі-продажу стає значно простішим, прозорішим і зручнішим для обох сторін. Основною перевагою маркетплейсів є можливість швидкого пошуку та порівняння великої кількості пропозицій. Користувачі можуть легко знаходити потрібний автомобіль, використовуючи розширені фільтри за маркою, моделлю, роком випуску, пробігом, ціновим діапазоном та іншими параметрами, що суттєво економить час і підвищує якість вибору.

Крім базових функцій пошуку, сучасні маркетплейси пропонують додаткові сервіси, які підвищують довіру до платформи та спрощують процес купівлі-продажу. Серед них: онлайн-оцінка вартості авто, перевірка технічного стану транспортного засобу, можливість перевірки історії автомобіля за VIN-кодом, а також допомога в оформленні документів і страхування. Ще однією важливою перевагою онлайн-маркетплейсів є об'єднання великої кількості пропозицій від різних продавців – як автодилерів, так і приватних осіб. Це створює конкурентне середовище, що позитивно впливає на ціноутворення та якість пропозицій. Для продавців це можливість швидко знайти покупця, розмістивши оголошення в одному або кількох популярних сервісах.

Важливу роль у формуванні довіри відіграють системи рейтингів і відгуків. Покупці можуть ознайомитися з досвідом інших користувачів, оцінити надійність продавця та уникнути шахрайства. Додатково, багато маркетплейсів впроваджують механізми модерації оголошень, що дозволяє підтримувати якість і правдивість інформації.

Завдяки широкому спектру функцій та сервісів, онлайн-маркетплейси стають все більш популярними серед українських споживачів. Вони суттєво полегшують

процес вибору, купівлі та продажу автомобілів, роблячи ринок більш відкритим, прозорим і доступним для всіх учасників.

1.3. Дослідження наявних онлайн-платформ та їх функціональних особливостей

На початковому етапі розробки власного маркетплейса вживаних автомобілів є критично важливим проведення ретельного аналізу вже наявних онлайн-платформ. Таке дослідження дозволить не лише виявити найбільш успішні практики та інноваційні рішення, але й уникнути потенційних помилок, а також отримати цінні ідеї для вдосконалення майбутнього продукту. Для проведення порівняльного аналізу було обрано декілька популярних в Україні маркетплейсів, а саме AUTO.RIA, RST.UA та AVTOBAZAR.UA, які є лідерами у своєму сегменті та мають значну аудиторію користувачів.

AUTO.RIA беззаперечно є однією з найвідоміших та найбільш відвідуваних онлайн-платформ, що спеціалізуються на купівлі-продажу вживаних автомобілів в Україні. Платформа пропонує надзвичайно широкий асортимент транспортних засобів різних марок і моделей, що значно полегшує процес пошуку та дозволяє користувачам ефективно фільтрувати пропозиції відповідно до їхніх індивідуальних потреб та вподобань. Щодо візуального оформлення вебсайту, то в його основі лежить поєднання червоного та синього кольорів, що загалом сприймається позитивно. Однак, структура та загальний дизайн сайту можуть здаватися дещо застарілими у порівнянні з сучасними тенденціями веброзробки. При першому відвідуванні сайту увагу користувача одразу привертає великий блок фільтрації, розміщений у центральній верхній частині сторінки (див. Рис. 1.3.). З одного боку, таке розташування є зручним для покупців, які вже чітко знають, який саме автомобіль вони шукають. З іншого боку, основний контент сторінки візуально відходить на другий план, що може бути менш зручним для користувачів, які перебувають на етапі ознайомлення з ринком.

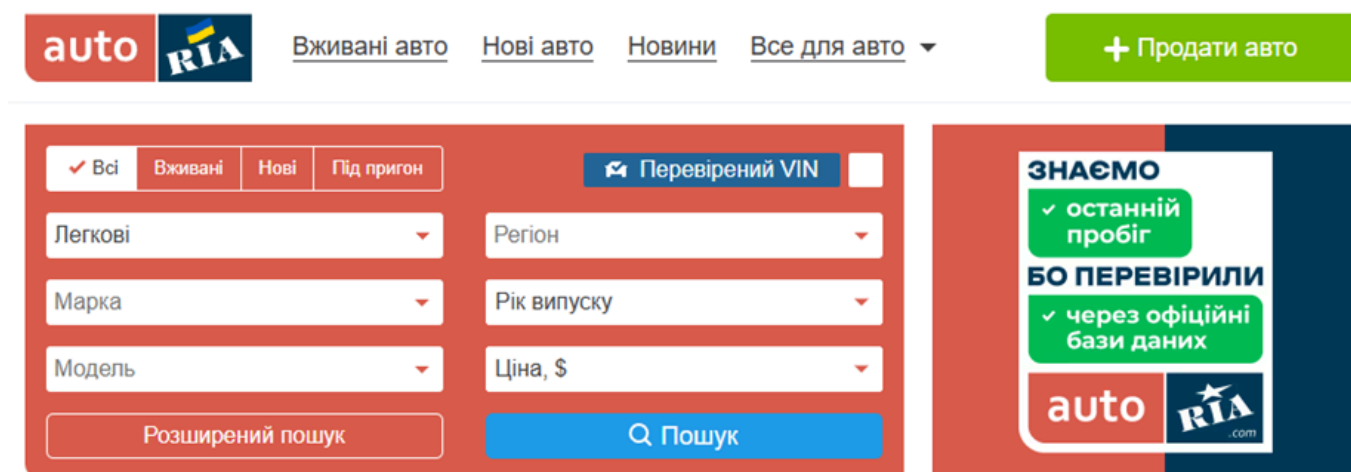


Рис. 1.3. Зображення частини сайту AUTO.RIA

Джерело: <https://auto.ria.com/uk/>

RST.UA є ще однією авторитетною онлайн-платформою в Україні, яка спеціалізується на продажу як нових, так і вживаних автомобілів. Платформа вирізняється високим рівнем пошукового трафіку та пропонує користувачам інтуїтивно зрозумілий і зручний інтерфейс. Візуальний дизайн платформи характеризується мінімалістичним підходом з використанням стриманої червоно-чорної колірної гами (див. Рис. 1.4.). Інтерфейс розроблено з акцентом на простоту навігації та зручність сприйняття інформації: оголошення представлені у вигляді компактного списку, де зображення автомобілів розміщені зліва, а ключові технічні характеристики та інформація про пропозицію – справа. Таке компонування забезпечує швидкий доступ до важливих параметрів автомобіля, уникаючи перенасичення сторінки зайвими візуальними елементами.

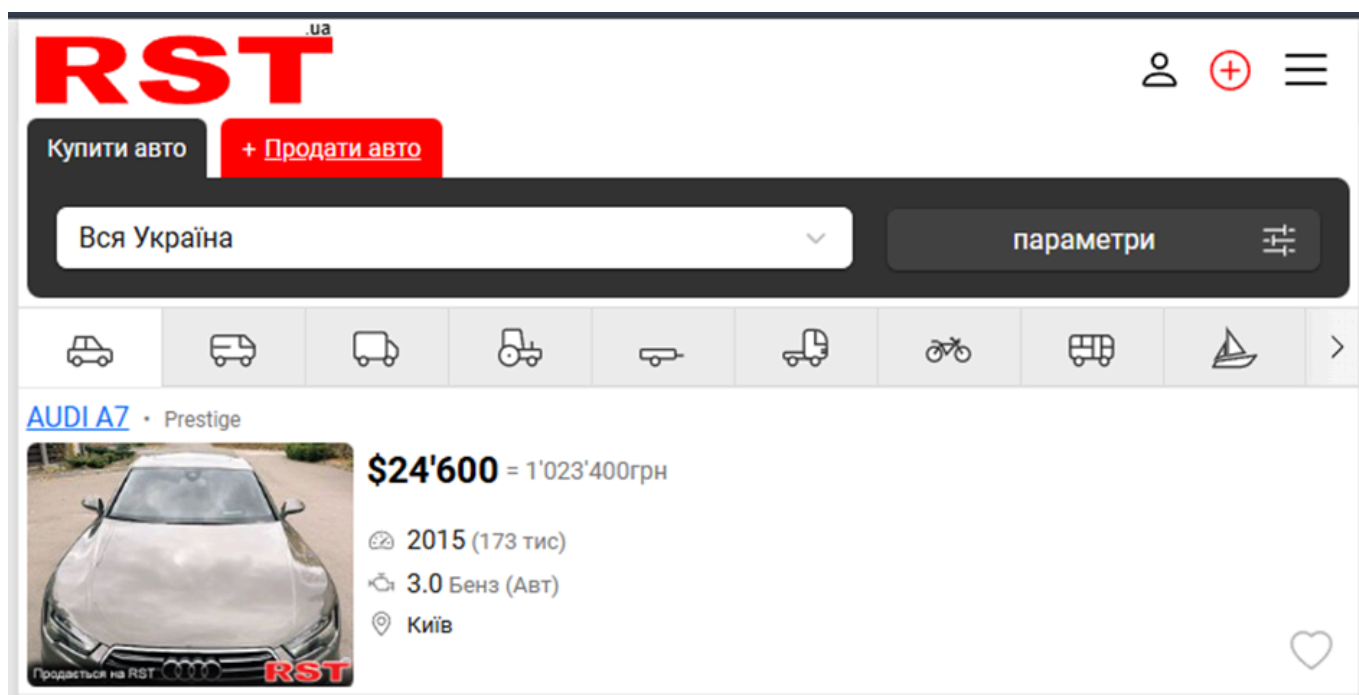


Рис. 1.4. Зображення частини сайту RST.UA

Джерело: <https://m.rst.ua/ukr/>

AVTOBAZAR.UA також є популярним майданчиком для купівлі-продажу вживаних автомобілів, пропонуючи широкий вибір транспортних засобів та розширені можливості для пошуку й фільтрації за різноманітними критеріями. Основною колірною гамою сайту є жовтий та його відтінки. Подібно до AUTO.RIA, на головній сторінці розміщено помітний блок фільтрації. Оголошення відображаються у форматі карток, кожна з яких містить великі та якісні фотографії автомобілів, що робить процес візуального ознайомлення значно зручнішим та інформативнішим для потенційних покупців (див. Рис. 1.5.). Варто відзначити, що світло-жовтий фон карток у поєднанні з сірим кольором другорядного тексту може ускладнювати читабельність, вказуючи на потенційні проблеми з контрастністю. Дизайн платформи вирізняється від попередніх розглянутих маркетплейсів виокремленням VIP-оголошень за допомогою спеціальних візуальних позначень, що може привертати додаткову увагу до преміальних пропозицій.

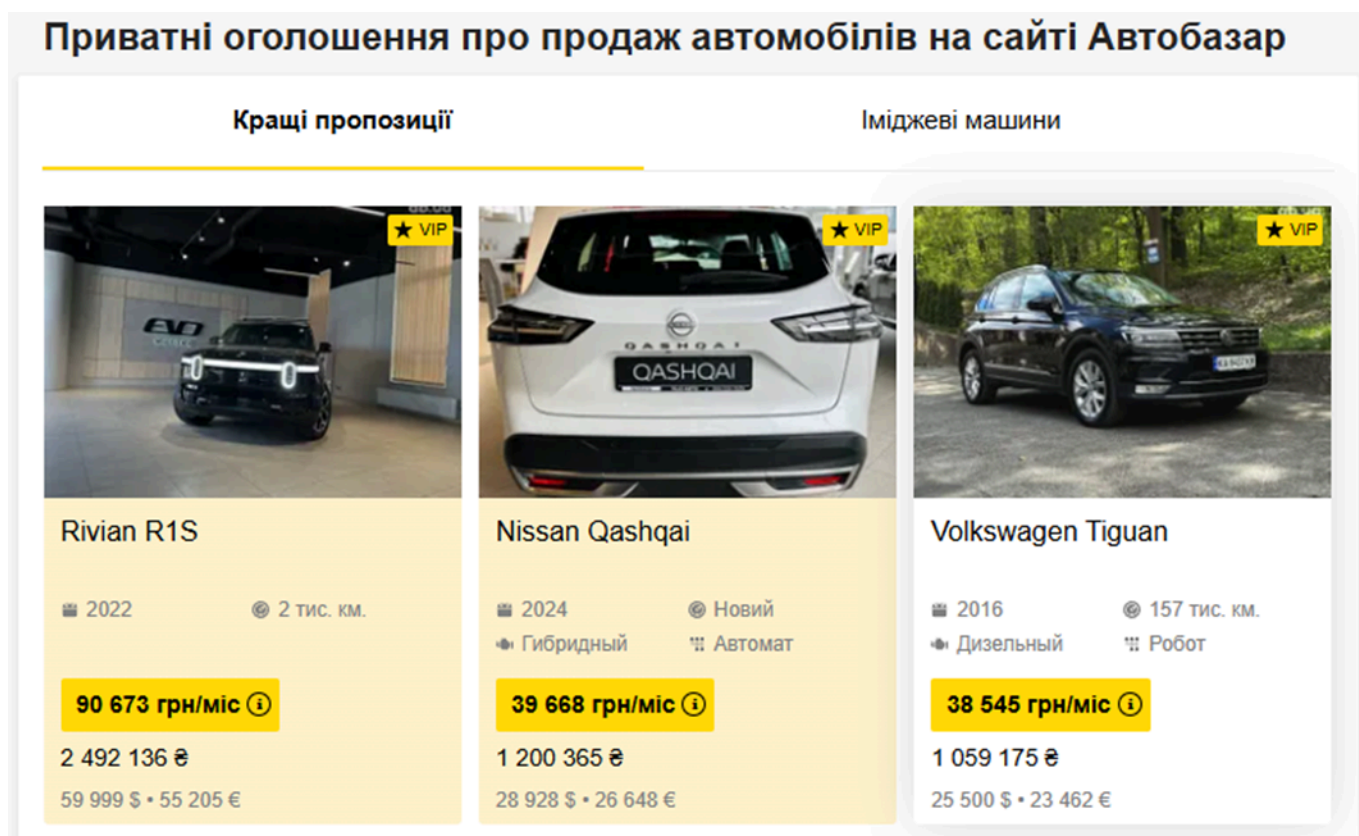
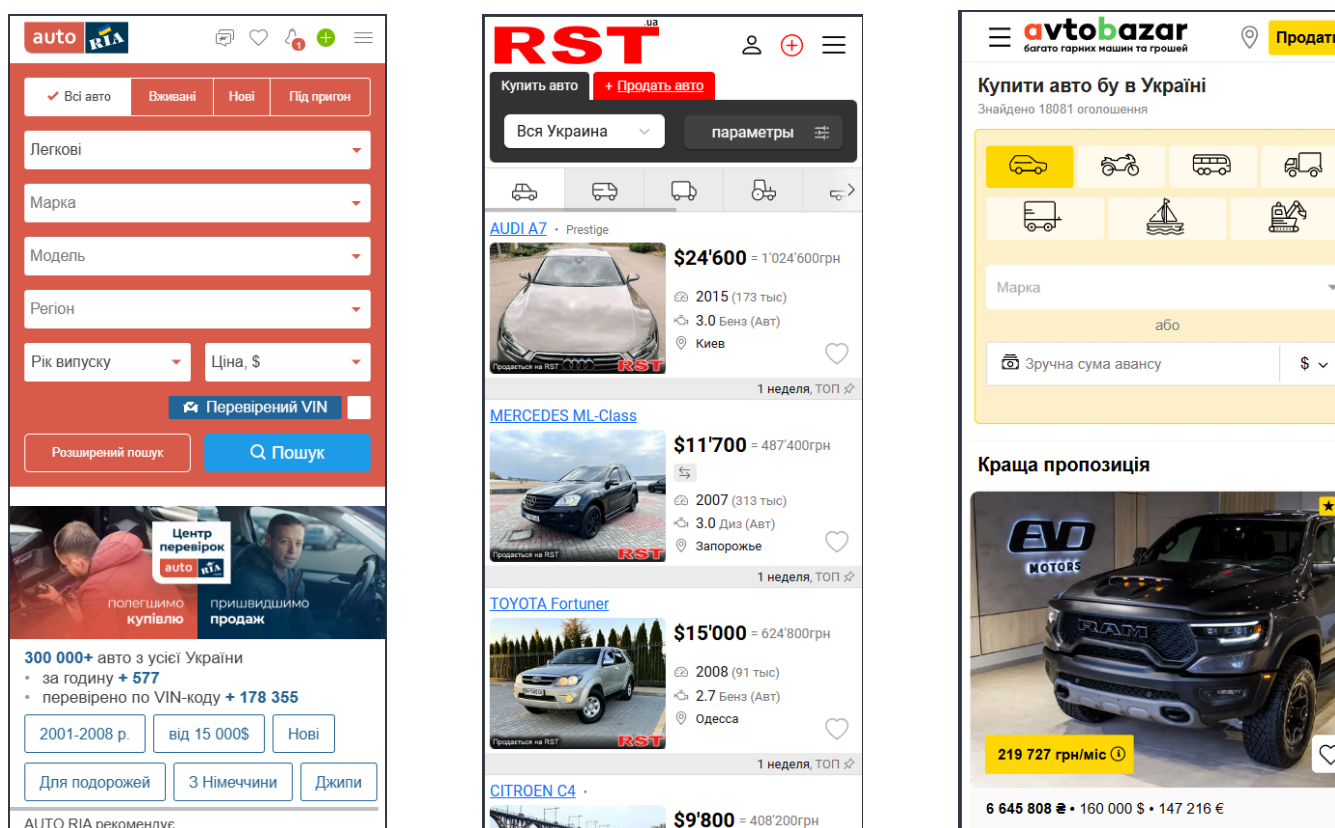


Рис. 1.5. Зображення частини сайту AVTOBAZAR.UA

Джерело: <https://avtobazar.ua/uk/>

Аналіз адаптивного дизайну на сайтах AUTO.RIA, RST.UA та AVTOBAZAR.UA виявив різні підходи до оптимізації відображення на мобільних пристроях. RST.UA демонструє досить ефективну адаптацію, де мобільна версія зберігає структуру десктопного варіанту, лише масштабуючись до менших розмірів екрана (див. Рис. 1.6. б). Дизайн AUTO.RIA на малих екранах може виглядати перевантаженим через великий блок фільтрів, який займає значну частину вертикального простору, хоча проблеми з масштабуванням відсутні (див. Рис. 1.6. а). AVTOBAZAR.UA намагається адаптувати макет на основі карток, проте через їхню ширину та розміри блоку фільтрації виникають певні складнощі з оптимальним відображенням на вузьких екранах (див. Рис. 1.6. в). Для об'єктивності порівняння для всіх трьох маркетплейсів використовувався однаковий емульований мобільний пристрій (через інструменти розробника браузера).



а

б

в

Рис. 1.6. Приклад адаптивного дизайну а - AUTO.RIA б - AVTOBAZAR.UA в - AVTOBAZAR.UA

Джерело: а - <https://auto.ria.com/uk/> б - <https://m.rst.ua/ukr/> в - <https://avtobazar.ua/uk/>

1.4. Визначення цільової аудиторії та її потреб

Ринок вживаних автомобілів в Україні характеризується значною різноманітністю цільової аудиторії, що охоплює кілька ключових демографічних та соціально-економічних груп, кожна з яких має свої унікальні потреби, мотивації та критерії вибору транспортного засобу. Глибоке розуміння цих особливостей є фундаментальним для успішного проектування та функціонування маркетингового плану. Основні сегменти цільової аудиторії включають:

- **Чоловіки віком 25-45 років.** Ця група є однією з найчисельніших та найактивніших на українському ринку вживаних автомобілів. Представники цієї вікової категорії часто шукають автомобілі як для щоденного особистого користування, включаючи поїздки на роботу та у справах, так і для задоволення потреб бізнесу, наприклад, для комерційних перевезень або представницьких цілей. Маючи, як правило, стабільний рівень доходу, вони віддають перевагу практичності, надійності та економічності в експлуатації при виборі транспортного засобу. Важливими факторами для них можуть бути витрата пального, вартість обслуговування, доступність запчастин, а також функціональність автомобіля.
- **Сімейні люди.** Для цієї категорії покупців ключовими критеріями при виборі вживаного автомобіля є надійність, безпека та місткість. Вони шукають просторі автомобілі, здатні комфортно розмістити всіх членів сім'ї та необхідний багаж, особливо для подорожей або сімейних поїздок. Особливо важливим для них є наявність сучасних систем безпеки, таких як подушки та шторки безпеки, системи курсової стійкості (ESP), антиблокувальна система гальм (ABS) та інші, які стають все більш поширеними навіть серед автомобілів з пробігом. Також важливим є зручність салону, наявність дитячих кріплень ISOFIX та загальний рівень комфорту.
- **Молодь (віком 20-24 роки).** Молоді покупці часто обмежені у бюджеті та зазвичай цікавляться найбільш доступними варіантами вживаних автомобілів, середній ціновий діапазон яких, за статистикою пошукових запитів, становить до \$4,000 (приблизно 18% від загальної кількості запитів). Для цієї групи пріоритетними є економічність в експлуатації (низька витрата пального, недороге обслуговування), вартість придбання, а також стильний зовнішній вигляд та наявність сучасних мультимедійних систем. Надійність також є важливим фактором, але часто молоді люди готові йти на певні компроміси заради нижчої ціни.
- **Підприємці та представники малого бізнесу.** Ця категорія покупців розглядає придбання транспортних засобів як стратегічну інвестицію,

спрямовану на оптимізацію бізнес-процесів. Вони орієнтуються на моделі, які максимально відповідають специфічним вимогам їхньої діяльності щодо функціональності, вантажності, прохідності та ефективності. Наприклад, для кур'єрських служб важливими є економічність та надійність, для будівельних компаній – вантажні характеристики та прохідність, а для представницьких цілей – зовнішній вигляд та комфорт. Вартість володіння та потенційна рентабельність також є ключовими факторами при прийнятті рішення.

- **Автолюбители та ентузіасти.** Ця особлива група покупців розглядає автомобіль не лише як утилітарний засіб пересування, а як об'єкт особистої пристрасті та захоплення. Їхні пріоритети часто зміщуються в бік унікальних, рідкісних або спортивних моделей, автомобілів з історією, а також транспортних засобів, які надають широкі можливості для тюнінгу та персоналізації. Вони цінують естетику, динамічні характеристики, історичну цінність бренду та навіть емоційний зв'язок з автомобілем. Для таких покупців важливими можуть бути можливості кастомізації, наявність спільнот за інтересами, доступність оригінальних запчастин та потенційна участь в автомобільних заходах та фестивалях.

Розуміння потреб та мотивацій кожної з цих цільових груп дозволить розробити маркетинг, який буде максимально адаптований до вимог ринку та запропонує кожному сегменту аудиторії найбільш релевантні пропозиції та функціональні можливості.

1.5. Формування вимог до проєктованого маркетингу

Після проведеного аналізу ринку, вивчення наявних платформ та потреб цільової аудиторії, наступним логічним кроком є формування вимог до проєктованого маркетингу вживаних автомобілів, які мають забезпечити максимальну зручність, прозорість та ефективність для всіх користувачів системи. Основними функціональними вимогами є наявність простої та інтуїтивно зрозумілої системи реєстрації та авторизації, що дозволить користувачам швидко створювати

облікові записи та входити в систему, з можливістю використання електронної пошти або соціальних мереж. Особистий кабінет має надати користувачам зручний інструмент для керування власними оголошеннями, включно з можливістю їх створення, редагування та видалення, а також перегляду історії своїх дій на платформі.

Подача оголошень повинна бути простою і зрозумілою, з підтримкою завантаження фотографій та детального опису технічних характеристик автомобіля, що дозволить підвищити інформативність пропозицій та довіру покупців. Критично важливим елементом платформи є система пошуку та фільтрації, яка має забезпечувати швидкий і точний підбір автомобілів за основними параметрами, такими як марка, модель, рік випуску, тип пального, коробка передач, пробіг і ціновий діапазон. Інтерфейс пошуку повинен бути адаптивним і зручним, щоб користувачі могли комфортно працювати як на десктопах, так і на мобільних пристроях, з урахуванням сучасних принципів UX/UI дизайну.

З огляду на те, що робота зосереджена на дизайні, особливу увагу слід приділити розробці інтуїтивно зрозумілого, привабливого та функціонального інтерфейсу, який забезпечить комфортну навігацію та взаємодію користувачів із платформою. Дизайн має бути адаптивним, забезпечуючи однаковий рівень зручності на різних пристроях та розмірах екранів. Крім того, інтерфейс повинен відповідати принципам доступності, щоб платформа була комфортною для користувачів з різними можливостями.

Таким чином, сформовані вимоги покривають всі ключові аспекти функціональності, дизайну та безпеки маркетплейса, що дозволить створити сучасний, зручний і надійний сервіс для купівлі-продажу вживаних автомобілів, який відповідатиме очікуванням українських користувачів і вимогам ринку

Висновки до розділу

У першому розділі було проведено комплексний аналіз предметної області та виконано проєктування маркетплейсу вживаних автомобілів. Станом на 2023 рік ринок демонструє поступове відновлення після спаду. Зростає частка внутрішніх перепродажів, а основний попит сконцентровано на доступних автомобілях європейського виробництва. Було визначено важливу роль онлайн-маркетплейсів, які значно спрощують процес купівлі-продажу завдяки систематизації пропозицій, додатковим сервісам та зручним інструментам пошуку й порівняння.

Аналіз наявних платформ (AUTO.RIA, RST.UA, AVTOBAZAR.UA) дозволив виявити як переваги, так і недоліки з точки зору функціональності, дизайну й користувацького досвіду. Це дало змогу окреслити напрямки вдосконалення майбутньої платформи. Було визначено основні сегменти цільової аудиторії (чоловіки 25-45 років, сімейні люди, молодь, підприємці та автолюбители) та їхні специфічні потреби. На основі проведеного дослідження сформовано вимоги до проєктованого маркетплейса, які охоплюють функціональні аспекти (реєстрація, особистий кабінет, подача оголошень, система пошуку) та дизайнерські рішення, що відповідатимуть сучасним принципам UX/UI з акцентом на адаптивність та інтуїтивність інтерфейсу.

РОЗДІЛ 2

ТЕОРЕТИЧНІ ОСНОВИ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ ДИЗАЙН-СИСТЕМИ МАРКЕТПЛЕЙСУ ВЖИВАНИХ АВТОМОБІЛІВ

2.1. Середовище розробки та організація процесу на GitHub

Для створення та роботи над дизайном було обрано **Figma**. Це векторний онлайн-сервіс для розробки інтерфейсів і прототипування, який підтримує організацію спільної роботи. Figma розробляється однойменною компанією та функціонує у двох форматах: у браузері та як клієнтський застосунок на комп'ютері користувача, зберігаючи онлайн-версії файлів[4]. Це рішення було обрано через ряд ключових переваг, таких як:

- Робота в режимі реального часу сприяє ефективній співпраці, забезпечує можливість відстежувати власний прогрес і ознайомлюватися з досягненнями колег.
- Спільний доступ до проєкту дає змогу швидко додавати нових учасників до команди з правами перегляду або редагування.
- Наявність двох версій програми, хмарної та десктопної, полегшує роботу, адже немає потреби встановлювати застосунок локально.

Figma займає ключову позицію серед інструментів для створення якісних дизайнів завдяки своїй багатофункціональності, зручному та інтуїтивно зрозумілому інтерфейсу, а також простоті у використанні. Цей інструмент пропонує дизайнерам можливості для швидкої та ефективної роботи, і серед його ключових особливостей можна виділити такі:

- Створення кольорових схем у Figma є важливим аспектом, оскільки платформа дозволяє створювати стилі, доступні по всьому проєкту. Це означає, що будь-які зміни в конкретному стилі автоматично відображаються на всіх елементах, які його використовують, забезпечуючи узгодженість дизайну і полегшуючи процес змін. Легкість редагування кольорових схем та інших

стилістичних елементів без необхідності індивідуального редагування кожного об'єкта робить роботу з дизайном гнучкішою, ефективнішою та структурованішою.

- Налаштування типографіки подібне до кольорів, адже можливість створювати стилі для шрифтів дозволяє автоматично застосовувати їх до всіх текстових елементів, підтримуючи єдиний стиль у проєкті.
- Figma також пропонує велику бібліотеку плагінів, які значно розширюють функціональні можливості платформи. Завдяки цим інструментам дизайн стає більш ефективним і зручним. Вони дають можливість автоматизувати багато рутинних завдань, заощаджувати час, інтегруватися зі сторонніми сервісами для додаткових функцій, а також створювати складні й багаторівневі елементи дизайну. Інтеграція плагінів дозволяє персоналізувати робочий процес залежно від специфічних потреб проєкту чи задачі.

Серед популярних альтернатив Figma, які варто розглянути задля порівняння, можна виділити Adobe XD та Canva. Для кращого розуміння розглянемо ці інструменти детальніше, починаючи з **Adobe XD**. Це програма, що полегшує процес макетування, розроблення візуального дизайну, створення інтерактивних елементів, прототипування, тестування та спільного використання, об'єднуючи всі необхідні інструменти для дизайну інтерфейсів в одному рішенні[5]. Серед її можливостей можна зазначити:

- Підтримка плагінів, що значно розширюють функціональність програми, дозволяючи розробникам створювати власні додатки[5].
- Унікальна функція голосових тригерів, яка дозволяє створювати прототипи з інтерактивною голосовою взаємодією, відсутня в інших подібних інструментах[5].
- Функція Auto-animate для анімацій, яка вирізняється легкістю налаштувань та високою швидкістю виконання[5].

- Система пов'язаних символів спрощує управління змінами у дизайн-системі, дозволяючи легко застосовувати оновлення у всіх пов'язаних файлах після зміни вихідного символу[5].
- Можливість прототипування жестів перетягування для розширення варіантів взаємодії з інтерфейсом[5].

Порівняймо функції, які надає Figma. Розпочнемо з плагінів, де Figma має перевагу завдяки великій кількості доступних плагінів, які легко інтегруються в середовище розробки. І Figma, і Adobe XD дозволяють створювати анімації, але коли йдеться про складні анімації з використанням сторонніх інструментів, Adobe XD має перевагу. Також Figma не підтримує на рівні вбудованих функцій жести drag-and-drop, хоча базові інтерактивні прототипи створюються досить просто. Щодо системи зв'язаних символів, у Figma є аналогічні функції, звані компонентами.

Canva являє собою сучасну і надзвичайно зручну платформу, яка спеціалізується на наданні користувачам широких можливостей для створення графічного дизайну. Вона здобула визнання серед користувачів завдяки своїй доступності, а також інтуїтивно зрозумілому інтерфейсу, який дозволяє легко зорієнтуватися навіть тим, хто вперше працює з подібними інструментами. Бібліотека Canva вражає своїм розмаїттям професійно підготовлених шаблонів, що допомагає користувачам створювати якісні проєкти з мінімальними зусиллями. Цей сервіс є ідеальним рішенням як для початківців, так і для тих, хто не має великого досвіду у сфері дизайну, оскільки надає можливість без особливих зусиль створювати естетично привабливі зображення та проєкти[6]. Серед головних характеристик, які роблять Canva унікальним інструментом, можна виділити наступні аспекти:

- Інтерфейс із функцією перетягування: зручний у використанні, навіть без попереднього досвіду в дизайні[7].
- Розширена бібліотека шаблонів: включає варіанти для публікацій у соцмережах, презентацій, листівок та багато іншого[7].

- Елементи оформлення: надає доступ до широкого асортименту фотографій, іконок, шрифтів і фігур[7].
- Інструменти спільної роботи: дозволяють ділитися проєктами та отримувати відгуки від колег чи клієнтів[7].
- Можливості експорту: підтримує завантаження готових робіт у форматах PNG, JPEG або PDF[7].

Порівнюючи Figma та Canva, можна помітити значні відмінності у їх функціональності та призначенні. Canva має значно обмежені можливості для роботи в галузі UI/UX дизайну. Цей інструмент менш ефективний для створення векторної графіки, складного прототипування чи інтерактивних компонентів. Натомість Figma розроблена з акцентом на потреби професійних дизайнерів. Вона пропонує потужні інструменти для точного налаштування, спільне редагування в реальному часі та розробку комплексних інтерфейсів. Таким чином, Canva стає ідеальним вибором для швидкого створення графічних матеріалів або простих дизайнів, навіть без серйозних навичок у цій сфері. Своєю чергою, Figma краще підходить для дійсно складних, професійних завдань дизайну. Простота та доступність – це сильні сторони Canva, але в питанні технічної гнучкості й детального опрацювання дизайну вона суттєво поступається Figma.

Проаналізувавши зазначені вище інструменти, було обрано Figma, зважаючи на її ключові переваги у сфері UI/UX дизайну. Попри те, що Adobe XD пропонує широкі можливості для анімації та унікальні функції, Figma вирізняється більшим вибором плагінів і кращими умовами для командної роботи у режимі реального часу. Щодо Canva, хоча вона має простий та інтуїтивний інтерфейс разом із багатою бібліотекою шаблонів, її функціонал суттєво поступається Figma в аспектах професійного дизайну, створення векторної графіки та прототипування. Завдяки оптимальному поєднанню потужного функціоналу, зручності колективної роботи та гнучкості в налаштуванні дизайн-систем Figma стає найкращим вибором для розробки професійних інтерфейсів.

Після визначення загального підходу до розробки та створення дизайну інтерфейсу в Figma постало завдання вибору ефективних інструментів для реалізації клієнтської частини проєкту. Для цього було обрано комбінацію Visual Studio Code (VS Code), GitHub і GitHub Desktop, що забезпечують продуктивну розробку, зручну співпрацю та якісне управління кодом.

Visual Studio Code — це потужне багатофункціональне середовище розробки, створене компанією Microsoft. Завдяки своїй гнучкості, розширюваності та зручному інтерфейсу, воно стало одним із найпопулярніших редакторів серед фронтенд-розробників. У межах реалізації клієнтської частини нашого проєкту VS Code виконував кілька важливих функцій:

- **Редагування коду.** Редактор підтримує синтаксис основних мов розробки — JavaScript, TypeScript, HTML, SCSS — і безперешкодно інтегрується з Angular CLI, що дозволяє швидко генерувати компоненти, модулі та сервіси.
- **Підтримка Tailwind CSS.** Завдяки розширенням (зокрема Tailwind IntelliSense і Prettier), робота зі стилями стала швидкою, інтуїтивною та зручною.
- **Інтелектуальні підказки.** VS Code пропонує автодоповнення коду, підсвітку синтаксису, форматування, лінтинг та інтеграцію з ESLint, що дає змогу підтримувати якість і єдиний стиль коду.
- **Вбудований термінал та підтримка Git.** Можна безпосередньо запускати локальний сервер, виконувати команди Angular CLI, а також використовувати вбудовані інструменти Git для фіксації змін і синхронізації з репозиторієм без потреби у сторонніх додатках.

У процесі розробки програмного забезпечення важливо використовувати ефективний інструмент для управління змінами в коді. Саме для цього призначені системи контролю версій — програмні інструменти, які дозволяють зберігати історію модифікацій, відстежувати розвиток проєкту, координувати роботу між розробниками та оперативно повертатися до попередніх версій у разі потреби. Завдяки таким системам можна уникнути втрати даних, спростити процес пошуку та виправлення помилок, а також забезпечити структуровану й організовану співпрацю

над програмним кодом. Застосування таких систем стає особливо важливим у командній розробці, де кілька учасників одночасно працюють над спільним проєктом. Використання централізованого або розподіленого репозиторію дозволяє швидко синхронізувати внесені зміни, ефективно вирішувати конфлікти між версіями та відстежувати поступ у проєкті. До того ж вбудовані інструменти для рецензування коду, планування завдань і створення документації сприяють підтриманню порядку та покращенню якості програмного продукту. Серед наявних рішень однією з найпопулярніших і найзручніших платформ для контролю версій вважається GitHub.

GitHub — це хмарна платформа, що надає можливість розміщення Git-сховищ для збереження та обміну проєктами. Це важливий інструмент для організації та управління проєктами, який спрощує колективну роботу та підвищує ефективність команди][8]. Основні характеристики платформи включають наступні:

- **Запити на внесення змін:** Це спеціальна функція, що дозволяє користувачам висловлювати свої міркування та надавати пропозиції щодо модифікацій в рамках наявного програмного сховища. Власники або відповідальні за проєкт отримують детальну можливість аналізувати подані пропозиції. Після старанного та обґрунтованого розгляду вони мають право затвердити ці зміни та об'єднати їх з основною стабільною гілкою, що гарантує інтеграцію корисних оновлень до проєкту[8].
- **Система відстеження проблем:** Мова йде про інтегровану платформу, розроблену для ефективного управління всіма аспектами, пов'язаними зі звітами про помилки, а також запитами щодо додавання нових функціональних можливостей. Ця система спрямована на значне покращення комунікації та співпраці серед членів команди, забезпечуючи структурований підхід до розв'язування проблем[8].
- **Інструменти управління проєктами:** У цей набір входять різноманітні ресурси, такі як мітки для позначення важливих аспектів, етапи реалізації для відстеження прогресу та інтерактивні проєктні дошки у стилі Kanban. Ці

інструменти значно покращують планування та виконання завдань, роблячи процес управління більш плавним та ефективним[8].

- **Перевірка коду:** Використання структурованих коментарів та вбудованих анотацій надає змогу розробникам простіше проводити аналіз якості коду. Це також сприяє оптимізації та вдосконаленню коду перед його остаточним впровадженням у базову систему, забезпечуючи певність у його надійності та ефективності[8].
- **Вікі-документація:** Це система, яка спеціалізується на створенні та підтримці доступної в будь-який час документації, пов'язаної з програмним кодом. Така організація дозволяє суттєво покращити його зрозумілість і полегшити використання, що, своєю чергою, сприяє більшій продуктивності та задоволеності серед розробників і користувачів[8].

Серед найважливіших і найбільш корисних функцій даного сервісу варто особливо виділити GitHub Boards. Цей інструмент надає користувачам чудову можливість візуально відстежувати прогрес у виконанні різноманітних завдань. GitHub Boards базується на методології Kanban, завдяки якій ви маєте змогу організовувати завдання у формі інтерактивних карток. Ця система передбачає існування трьох основних стовпців, які допомагають структурувати робочий процес і забезпечувати плавний перехід між етапами виконання завдань:

- To Do (Зробити) – завдання, які ще не розпочаті.
- In Progress (В роботі) – завдання, над якими активно працюють.
- Done (Завершено) – виконані завдання.

Платформа GitHub Boards надає широкий спектр інструментів для ефективного управління проектами та завданнями. Кожне завдання відображається у формі інтерактивної картки, яка містить інформацію про завдання, зокрема детальний опис, встановлені дедлайни та призначення відповідального виконавця (див. Рис. 2.1.). Таке рішення сприяє чіткій структуризації та покращенню організації робочих процесів. Відстеження прогресу виконання завдань здійснюється

шляхом переміщення карток між різними стовпцями, що дозволяє оперативно оцінювати стан реалізації проєкту.

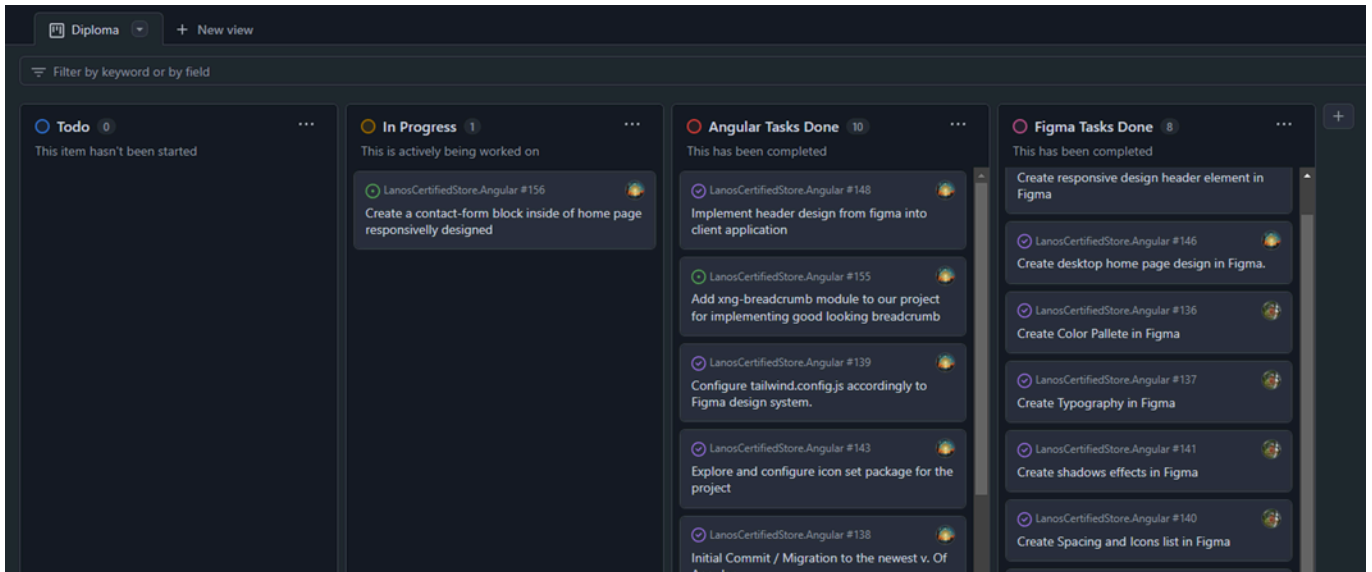


Рис. 2.1. Інтерфейс GitHub Boards із прикладом дошки для управління завданнями

Джерело: розроблене автором

Однією з ключових переваг платформи є високий рівень гнучкості, користувачі можуть налаштовувати дошку відповідно до потреб команди, зокрема додавати нові стовпці, коригувати тегування, змінювати порядок пріоритетності завдань чи оновлювати структуру. Окрім цього, функціонал колективної роботи в реальному часі значно підвищує ефективність, адже всі зміни автоматично відображаються для всіх учасників команди.

Хоча VS Code має вбудовану підтримку Git, для зручності візуального управління репозиторієм додатково використовувався **GitHub Desktop**. Це окрема десктопна програма, що надає графічний інтерфейс для роботи з Git і репозиторіями GitHub. Основні переваги:

- Зручне керування гілками: Можна легко створювати нові гілки, перемикатися між ними та виконувати злиття без використання командного рядка.
- Візуалізація історії змін: Дозволяє швидко проаналізувати внесені коміти та знайти потрібні версії.

- Простота у використанні: Підходить навіть для новачків, які лише починають знайомство з Git.
- Швидкий доступ до Pull Request'ів: Інтеграція з GitHub спрощує відкриття та перегляд запитів на внесення змін, що прискорює перевірку й злиття коду.

GitHub Desktop значно спростив координацію дій між членами команди та зробив процес керування репозиторієм доступним навіть для тих, хто не має досвіду командного рядка.

2.2. Теорія та розробка кольорової схеми й color token системи

В процесі розробки дизайну інтерфейсу значну увагу потрібно приділяти узгодженій роботі всіх його компонентів. Це необхідно для забезпечення цілісності та функціональності продукту. Одним із визначальних факторів, що безпосередньо впливає на загальне сприйняття інтерфейсу, є продумане застосування колірної гами. Кольори мають важливе значення не лише для створення привабливого вигляду, але й для формування відчуттів та емоцій користувачів під час взаємодії з продуктом. З огляду на це, перед вибором колірної палітри слід детально ознайомитися з основними правилами роботи з кольором та зрозуміти, як правильно створювати гармонійні колірні комбінації. Такий підхід дасть змогу зробити дизайн не тільки естетично привабливим, але й зручним для сприйняття та використання.

Колірна схема – це узгоджене поєднання кольорів, які візуально вдало доповнюють одне одного. Її зазвичай використовують у графічному дизайні, оформленні приміщень, індустрії моди та мистецтві. Головне призначення колірної схеми – створення цілісного образу, що привертає увагу аудиторії. Вдалий вибір кольорів здатен не тільки викликати певні емоції, але й передати задум дизайнера як на візуальному, так і на психологічному рівнях[9]. У колористиці розрізняють сім основних типів колірних схем, кожен з яких має унікальні особливості та сфери застосування. Розгляньмо їх детальніше:

- **Монохроматична** палітра об'єднує всі відтінки одного кольору, включаючи тони та нюанси, отримані шляхом додавання чорного, сірого чи білого до базового кольору (див. Рис. 2.2. а). Хоч такі схеми не мають яскраво вираженого контрасту, вони відкривають широкі можливості для експериментів із насиченістю відтінків, дозволяючи творцям підібрати оптимальне рішення для своїх проєктів[9].
- **Комплементарна** колірна схема базується на кольорах, що розташовані протилежно на колірному колі (див. Рис. 2.2. б). Зазвичай така палітра містить лише два кольори, які створюють сильний контраст. Вони можуть виглядати красиво, проте інколи здаються занадто насиченими для сприйняття. Тому в дизайні до комплементарної палітри додаються відтінки та тони основних кольорів, що надає можливість дизайнерові створювати графіку, яка не лише привертає увагу, але і є більш комфортною для очей[9].
- **Спліт-комплементарна** схема складається з одного основного та двох додаткових кольорів, розташованих поруч із протилежним до базового на колірному колі (див. Рис. 2.2. в). Основний колір виступає як фон, а додаткові використовуються для встановлення акцентів, надаючи дизайну більшої виразності[9].
- **Ахроматична** колірна схема містить виключно нейтральні відтінки: чорний, білий і сірий (див. Рис. 2.2. г). Такі палітри виглядають естетично та гармонійно, дозволяючи створювати стильний і мінімалістичний дизайн[9].



а



б



в



г

Рис. 2.2. Приклад а - монохроматичної, б - комплементарної, в - спліт-комплементарної, г - ахроматичної палітри

Джерело: <https://blog.depositphotos.com/ua/kolirni-shemy.html>

- **Аналогова** колірна схема базується на виборі одного домінантного кольору та двох сусідніх йому на колірному колі (див. Рис. 2.3. а). Такі поєднання створюються досить легко, але для досягнення гармонії у візуальному ряді важливо дотримуватися єдності температури кольорів, обираючи лише теплі або лише холодні відтінки в палітрі. Оскільки кольори в аналоговій схемі природно гармоніюють між собою, варто також звернути увагу на достатній рівень контрастності, щоб забезпечити ефективність та виразність вашого дизайну[9].
- **Тріадична** колірна схема складається з трьох кольорів, які знаходяться на однаковій відстані на колірному колі, утворюючи умовний трикутник (див. Рис. 2.3. б). Такі яскраві палітри дають дизайнерам можливість створювати виразні й гармонійні візуальні рішення зі значним контрастом. Головний секрет успішного використання цієї схеми полягає у правильному розподілі кольорів через колірну ієрархію, тобто один відтінок має переважати, а решта два виконуватимуть функцію акцентів[9].
- **Тетрадічна** відома також як подвійна комплементарна або прямокутна, є колірною схемою, що включає дві пари комплементарних кольорів, розташованих у формі прямокутника на колірному колі (див. Рис. 2.3. в). Змішуючи всі відтінки з цієї палітри в рівних пропорціях, ви ризикуєте отримати незбалансований дизайн. Для досягнення найбільш гармонійного результату варто обрати один колір як основний, а інші три використовувати для створення акцентів[9].



Рис. 2.3. Приклад а - аналогової, б - тріадичної, в - тетрадічної

Джерело: <https://blog.depositphotos.com/ua/kolirni-shemy.html>

Представлені вище колірні схеми є ефективним засобом для створення збалансованого та візуально привабливого дизайну. Однак, успішність їх застосування визначається не лише естетикою, але й зручністю використання та легкістю сприйняття інформації. У цьому контексті значну роль відіграє застосування контрастних кольорів та акцентів, які допомагають структурувати контент і виділяти ключові елементи дизайну. Контраст є *foundational principle of design*, що сприяє виокремленню важливих елементів, покращує читабельність тексту та робить дизайн більш інтуїтивно зрозумілим і привабливим. Використання виразних кольорів не тільки забезпечує чітке розмежування різних компонентів на сторінці, але й робить інтерфейс більш комфортним для користувачів. Контрастні кольори необхідні для виділення заголовків, основного тексту, кнопок та інших значущих елементів, що дозволяє користувачам швидко орієнтуватися.

Щодо колірних акцентів, то додавання одного або двох насичених кольорів до дизайну допомагає привернути увагу до важливих інтерактивних елементів, таких як кнопки дій, посилання або повідомлення. Наприклад, якщо основна палітра сайту нейтральна, то яскраво-червона кнопка гарантовано приверне погляд. Колірні акценти також сприяють зниженню когнітивного навантаження, оскільки користувачі підсвідомо розуміють, на що варто звернути увагу.

Таким чином, вибір кольорів для дизайну інтерфейсу є надзвичайно важливим. Грамотно підібрана палітра робить інтерфейс не тільки естетично приємним, але й

функціональним та зручним у користуванні. Кольори допомагають звернути увагу на ключових елементах, полегшують навігацію та забезпечують доступність інтерфейсу для широкого кола користувачів. Саме тому колірна схема є фундаментальною частиною для створення гармонійного та ефективного дизайну, що відповідає потребам цільової аудиторії.

При створенні маркетплейсу першочерговим завданням став підбір кольорів, оскільки саме вони формують атмосферу всього дизайну. Підібрані кольори мають гармонійно поєднуватись, відображати ідентичність платформи та відповідати її функціональному призначенню. Базові кольори забезпечують цілісний візуальний образ, а акцентні виділяють значущі елементи, сприяючи кращій навігації та взаємодії користувачів із продуктом.

Для дизайну було обрано кольори синьо-фіолетового спектра, адже вони традиційно асоціюються з надійністю, довірою та впевненістю. Загалом використано п'ять основних кольорів, що є оптимальною кількістю для створення привабливого дизайну. Вибір зупинився на таких кольорах (див. Рис. 2.4.):

- Port Gore (#17183B);
- Jaguar (#1C0221);
- Violet Eggplant(#A81C85);
- Kimberly (#8371A3);
- Snuff (#EAE3F3);

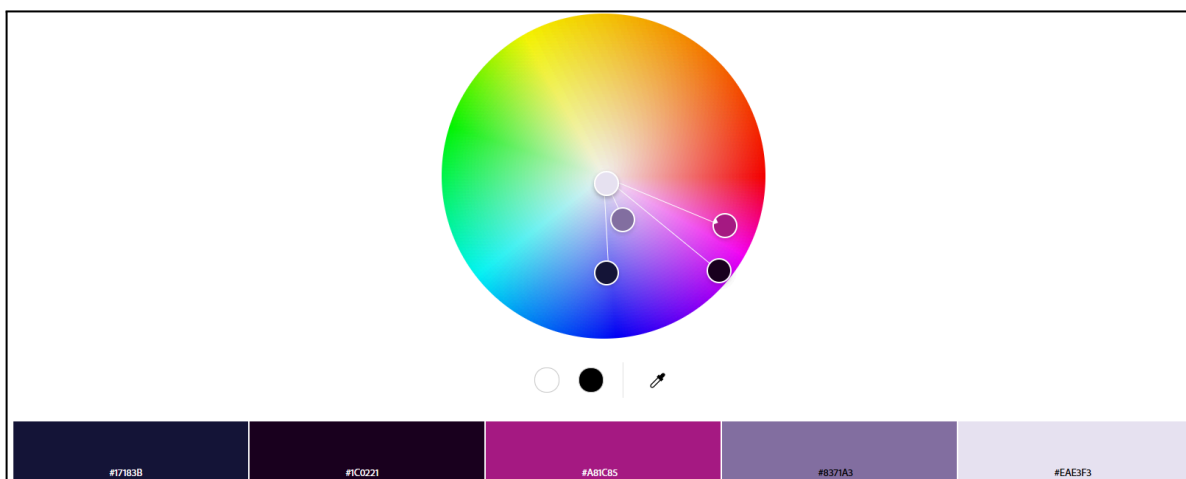


Рис. 2.4. Кольорове коло з обраними кольорами

Джерело: <https://color.adobe.com/create/color-wheel>

Аналізуючи розташування вибраних кольорів на кольоровому колі, можемо спостерігати, що вони утворюють аналогову колірну схему, взаємно гармоніюють, а використання як світлих, так і темних тонів дозволяє забезпечити належну контрастність. Для комплексності палітри до кожного основного кольору створено серію відтінків, що розширює колірну гаму та надає схемі більшої гнучкості. Для цього використано сервіс Tints.Dev — зручний онлайн-інструмент для генерації відтінків базового кольору, який також чудово інтегрується з Tailwind CSS[10].

Розглянувши рисунок 2.5., бачимо загальний вигляд створеної палітри кольорів, розробленої у Figma. Розширена гама кольорів забезпечує можливість створення адаптивного дизайну, який поєднує контрастність із цілісністю продукту. Застосування різноманітних відтінків кожного кольору сприяє гармонійному поєднанню елементів інтерфейсу. Світлі відтінки ідеально підходять для фонових використання, створюючи враження простору та легкості, тоді як темні й акцентні кольори застосовуються для текстових елементів, кнопок та інших ключових компонентів. Це допомагає встановити чітку візуальну ієрархію, що покращує користувацький досвід.



Рис. 2.5. Палітра кольорів з відтінками

Джерело: розроблено автором

У контексті розробки онлайн маркетплейсу важливо звернути увагу на три додаткові кольори — червоний, жовтий та зелений (див Рис. 2.6.). Ці кольори суттєво впливають на вдосконалення користувацького досвіду, підкреслюючи важливі елементи. Червоний традиційно використовується для попереджень і помилок, привертаючи увагу до критичних ситуацій. Жовтий найкраще підходить для інформаційних повідомлень середнього рівня важливості, а зелений — для індикації успішного виконання дій і підтверджень. Ці кольори також інтегруються в інтерфейс через модальні вікна, сповіщення або кнопки підтвердження, що допомагає швидко інтерпретувати необхідні дії. Вони підвищують інтуїтивність та зручність взаємодії з сайтом. Також ці кольори можуть застосовуватись в іконках і підписах для полегшення навігації в інтерфейсі.



Рис. 2.6. Палітра кольорів з відтінками

Джерело: розроблено автором

Після визначення основної кольорової гама для маркетинг-плейса наступним кроком стала імплементація у Figma системи змінних (variables) для ефективного керування дизайном (див. Рис. 2.7.). Створивши колірні блоки, ми використали доступні у Figma плагіни, зокрема Styler, який трансформує обрану палітру в стилі. Для подальшого формування змінних було застосовано плагін Styles to Variables, що конвертує стилі у відповідні змінні. Така система систематизує кольорову схему, гарантує візуальну узгодженість та дозволяє оперативно вносити необхідні корективи.

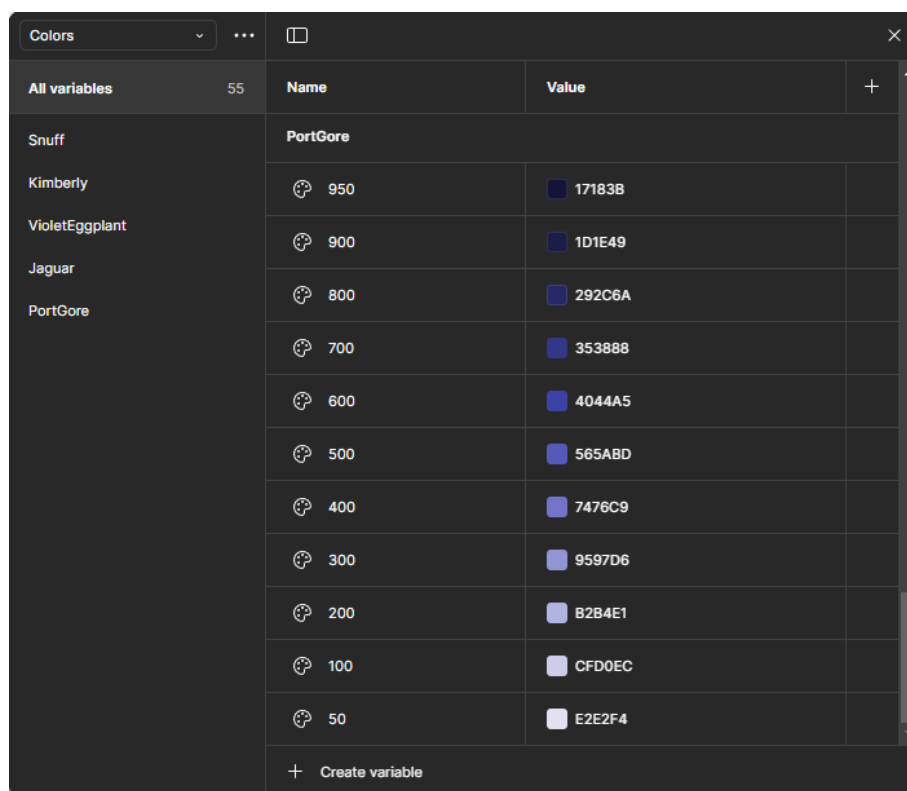


Рис. 2.7. Приклад створених variables у Figma

Джерело: розроблене автором

На представленій ілюстрації показано набір кольорів із градацією від світлих до темних тонів. Така структура надає дизайну гнучкості: світлі тони ідеально підходять для фонового застосування, створюючи відчуття простору й легкості, тоді як темні призначені для акцентування, наприклад у тексті, кнопках чи ключових елементах інтерфейсу. Це дозволяє підтримувати баланс між контрастністю та гармонійністю. Figma пропонує функціонал інтеграції змінних із глобальним впливом на проєкт. При зміні кольору в системі змінних дизайнер миттєво бачить оновлення на всіх пов'язаних елементах. Це значно спрощує підтримку дизайну та забезпечує єдність кольорової схеми в усьому продукті. Таким чином, дизайн стає не лише адаптивним і масштабованим, а й придатним для використання на різних пристроях і для різноманітних завдань.

Після розробки колірної палітри, що заклала основу візуального стилю нашого маркетплейса, наступним критично важливим етапом стало створення

системи кольорових токенів (color tokens). Ці токени є не просто змінними, що зберігають певні колірні значення (як, наприклад, `$blue-400 = #2680EB`), а наділені контекстом використання. Вони описують призначення кольору в інтерфейсі (наприклад, `$button-cta-background-color = $blue-400`), виступаючи як абстрактний рівень поверх базових колірних змінних дизайн-системи. Такий підхід є фундаментальним для забезпечення послідовності та передбачуваності у візуальному оформленні продукту, усуваючи неоднозначність між іменуванням кольору та його фактичною роллю в інтерфейсі. Це значно спрощує роботу як дизайнерам, так і розробникам, дозволяючи легко обирати правильні колірні параметри для конкретних елементів та станів у дизайні [11].

Для ефективної організації та спрощення застосування кольорових токенів у різноманітних сценаріях використання інтерфейсу маркетплейса, було прийнято рішення структурувати їх за функціональними категоріями. Ця структура дозволяє групувати токени на основі їхньої ролі та місця застосування в дизайні. Для забезпечення логічної структури та зручності застосування в інтерфейсі маркетплейса, система кольорових токенів була організована за функціональними категоріями, як показано на Рис. 2.8.. Цей підхід дозволяє групувати токени за їхньою роллю в дизайні, полегшуючи вибір потрібного кольору для конкретного елемента. Кожна категорія містить набір функціональних токенів, прив'язаних до конкретних елементів інтерфейсу, що забезпечує їх систематизоване використання, навіть якщо не всі токени відображені на ілюстрації.

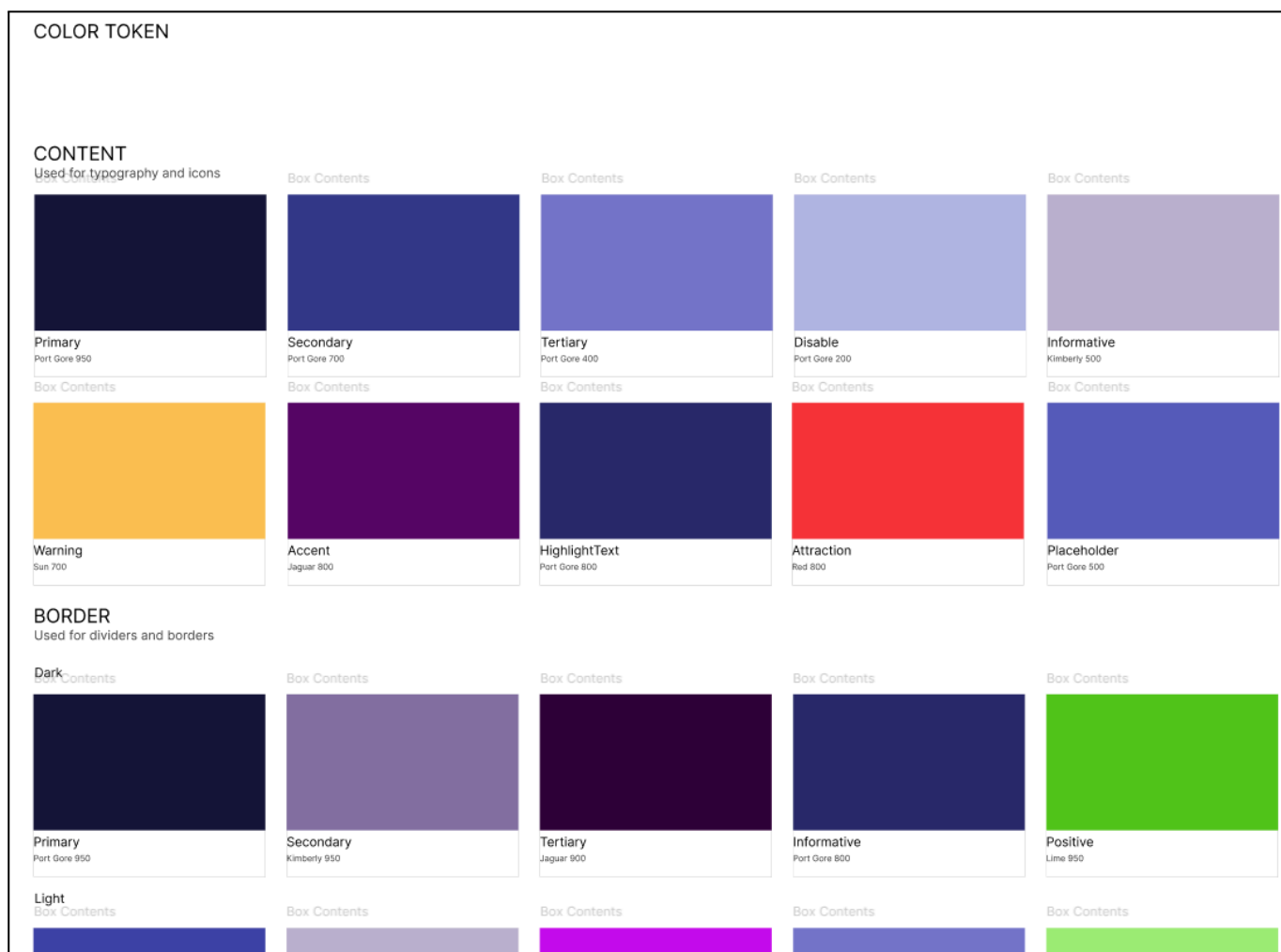


Рис. 2.8. Система кольорових токенів для маркетплейса

Джерело: розроблено автором

Впровадження та застосування цих кольорових токенів відбувається безпосередньо в дизайні інтерфейсу проєкту (див. Рис. 2.9.). На цьому зображенні видно, як конкретні елементи інтерфейсу, такі як кнопки, поля вводу, заголовки тощо, використовують не прямі колірні значення, а посилання на відповідні кольорові токени з функціональних категорій (наприклад, Content/Primary, Background/Secondary, Buttons/Primary тощо).

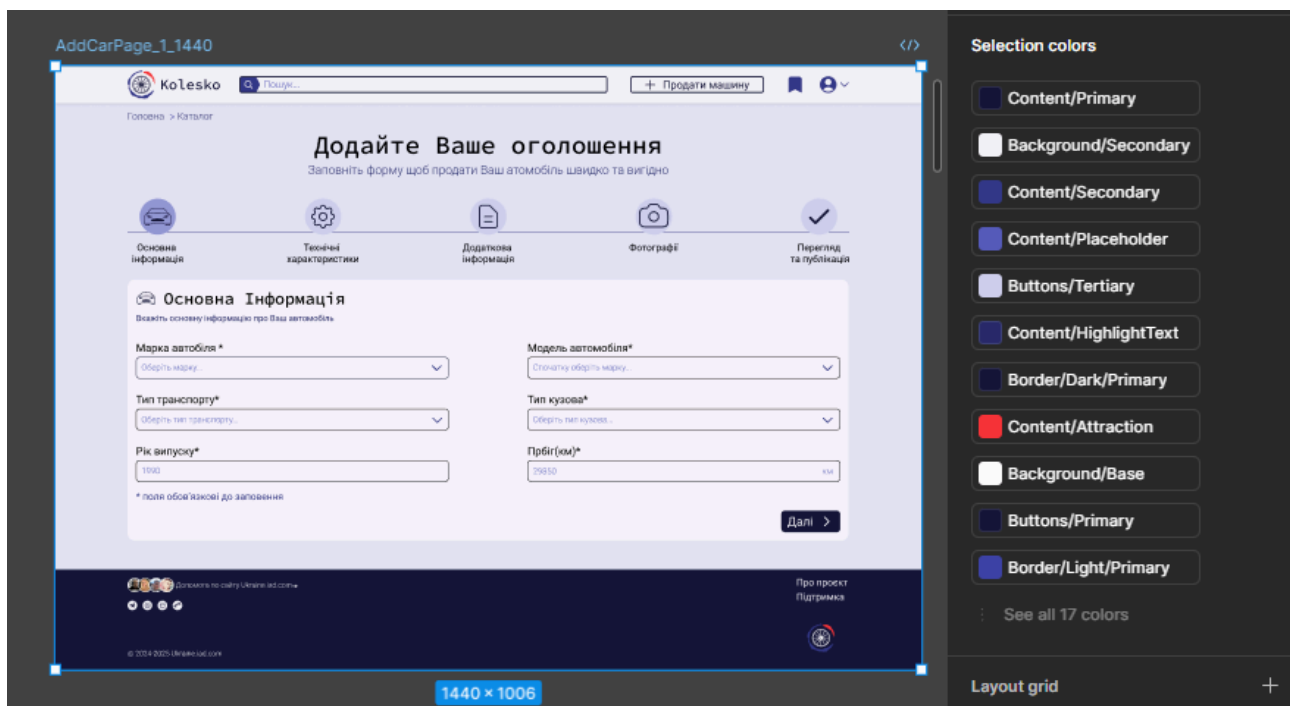


Рис. 2.9. Застосування кольорових токенів у дизайні маркетплейсу

Джерело: розроблено автором

Використання такої системи токенів замість безпосереднього використання жорстко закодованих кольорних значень (#HEX, RGBA) чи навіть просто базових кольорних змінних надає проєкту суттєві переваги як на етапі проєктування, так і в процесі розробки:

- **Гнучкість та керованість.** Токени забезпечують високий рівень гнучкості. Дизайнери та розробники можуть оновлювати значення токенів централізовано в дизайн-системі. Будь-які зміни автоматично застосовуються до всіх компонентів та екранів, де ці токени використовуються, що значно спрощує внесення глобальних змін у візуальний стиль або реагування на зміни в брендбуку.
- **Прискорення розробки.** Розробникам не потрібно запам'ятовувати чи шукати конкретні кольорні коди. Вони просто використовують змістовні назви токенів, що відповідають функціональному призначенню елемента. Це зменшує ймовірність помилок та прискорює процес верстки.

- **Консистентність між платформами.** Токени є універсальним джерелом істини для візуального стилю незалежно від платформи (веб, мобільні додатки під різні ОС). Хоча базові значення кольорів можуть бути однаковими, система токенів дозволяє формувати ці значення відповідно до вимог конкретного середовища (наприклад, HEX для вебу, RGBA для мобільних платформ). Це забезпечує єдиний зовнішній вигляд продукту на всіх пристроях.
- **Покращена взаємодія команди.** Спільне використання зрозумілих та контекстуалізованих токенів сприяє кращому порозумінню між дизайнерами та розробниками, зменшуючи кількість неточностей та прискорюючи ітераційний процес.
- **Легкість підтримки та масштабування.** З розвитком продукту та додаванням нових функцій підтримка візуальної консистентності стає простішою завдяки централізованому управлінню кольорами через токени. Система легко масштабується, дозволяючи додавати нові токени та категорії за необхідності.

Таким чином, впровадження системи кольорових токенів, структурованих за функціональними категоріями, стало ключовим кроком у побудові надійної та гнучкої дизайн-системи для маркетплейса, що забезпечує візуальну консистентність, прискорює процеси проєктування та розробки, а також спрощує подальшу підтримку продукту.

2.3. Типографія в інтерфейсі: теоретичні аспекти та практична реалізація

Типографія є важливою частиною дизайну, адже правильний вибір та поєднання шрифтів забезпечують зрозумілість і зручність подання інформації. Основна мета типографії — зробити текст легким для сприйняття та приємним для читання, що на пряму впливає на взаємодію користувача із сайтом. Вдалий підбір шрифтів покращує як естетичну привабливість, так і функціональність контенту, сприяючи кращому користувацькому досвіду. Шрифти виконують не лише декоративну функцію, а й допомагають формувати чітке візуальне повідомлення. Їхній вибір має враховувати особливості різних пристроїв, адже це впливає як на

естетику, так і на зручність використання. Читабельність тексту залежить від його чіткості, ясності та відсутності зайвого напруження для очей під час читання. Розглянемо основні принципи, що сприяють читабельності:

- **Контраст між фоном і текстом.** Текст повинен бути достатньо контрастним до фону, щоб його було легко читати[12].
- **Розмір шрифту.** Для основного тексту оптимальний розмір шрифту становить від 14 до 18 пікселів на вебсторінках, залежно від типу контенту та пристрою[12].
- **Простота у виборі шрифтів.** Для основного тексту доцільно використовувати лаконічні шрифти без зайвих декоративних деталей, які зручно читати з екрана[12].

У межах реалізації проєкту, для забезпечення оптимальної читабельності та створення сучасного, естетично гармонійного вигляду текстового контенту, було прийнято рішення використати шрифти «Roboto», як основний для текстових блоків, і «Source Code Pro» для оформлення заголовків. Такий вибір допомагає не лише візуально структурувати інформацію, але й підкреслити стильність і функціональність дизайну.

Roboto є одним із тих шрифтів, котрі здобули величезну популярність в області вебдизайну. Ця популярність пояснюється його унікальною здатністю бути надзвичайно універсальним, вишукано естетичним та водночас простим у застосуванні. Особливість Roboto полягає в тому, що він вдало поєднує у собі нейтральний характер наділеного сучасними графічними лініями, що робить сприйняття текстів легким і комфортним навіть під час тривалого перегляду на екрані улюбленого пристрою[13]. Завдяки вдало продуманій висоті символів, цей шрифт забезпечує високий рівень читабельності. М'якість та сучасність у його зовнішньому вигляді досягнута шляхом використання овальних форм, які додають привабливості. В умовах сучасних тенденцій використання цього шрифту активно зростає, адже він повною мірою відповідає вимогам універсальності та

естетичності. У наступному ви можете побачити приклад використання Roboto у його різних варіаціях, що представлені на ілюстрації (див. Рис. 2.10.).

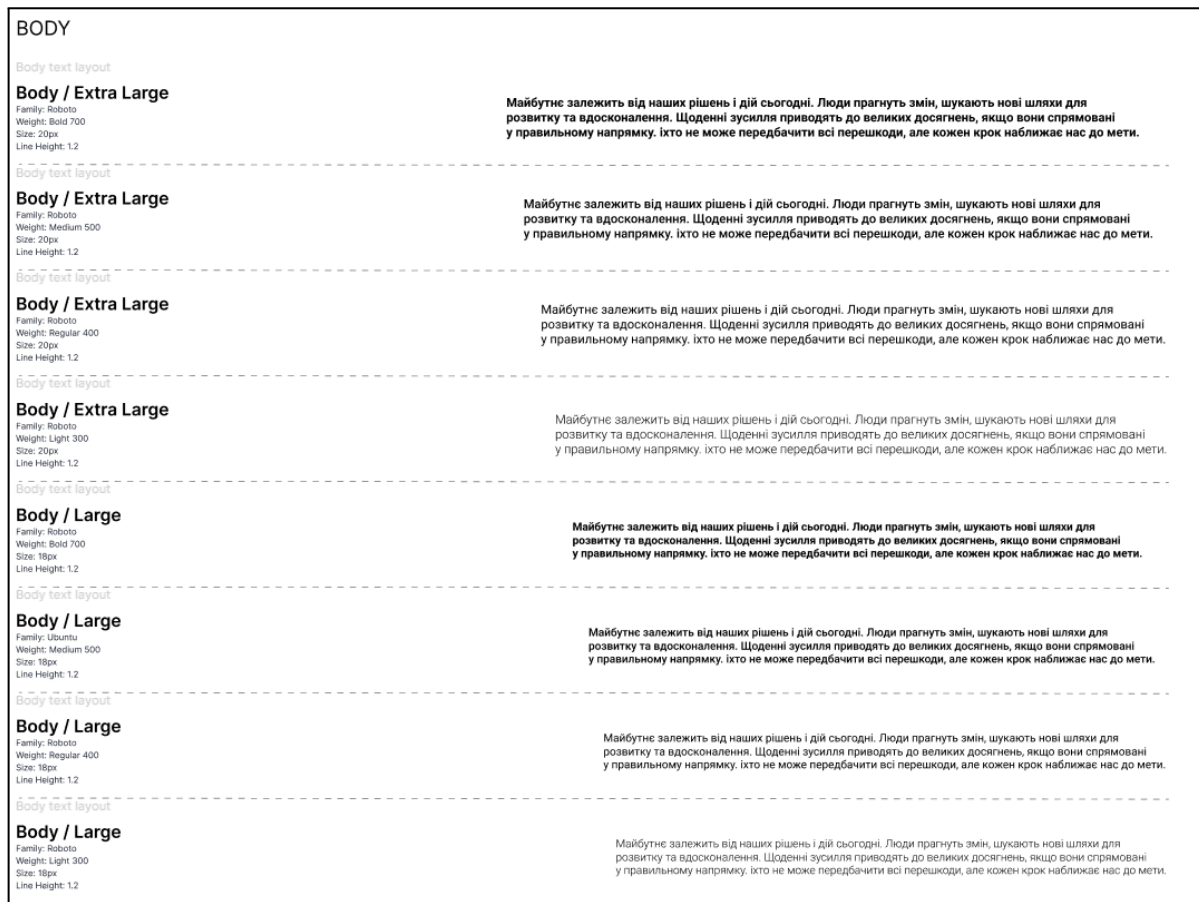


Рис. 2.10. Приклад створених стилів шрифту Roboto у Figma

Джерело: розроблене автором

Для заголовків обрано дещо нетрадиційний, але дуже ефективний шрифт Source Code Pro. Source Code Pro — це моношрифт, створений для зручності програмування. Він вирізняється серед інших шрифтів завдяки поєднанню класичної форми з гнучкістю, що дозволяє використовувати його не лише в програмуванні, а й у вебдизайні[14]. Чіткі пропорції, хороша читабельність та сучасний вигляд роблять Source Code Pro універсальним вибором, навіть для таких сфер, як вебдизайн чи інфографіка. Завдяки збалансованій естетиці, яка поєднує нейтральність і точність, Source Code Pro відображає ідеї інноваційності та мінімалізму, що дуже актуально для сучасного дизайну. Шрифт чудово функціонує у створенні чіткої ієрархії тексту,

виділяючи заголовки на фоні основного контенту. Також Він додає дизайну технічної нотки з акцентом на сучасний стиль, що ефективно контрастує з плавністю Roboto. Нижче наведено приклад використання Source Code Pro в різних його варіаціях (див. Рис. 2.11.).

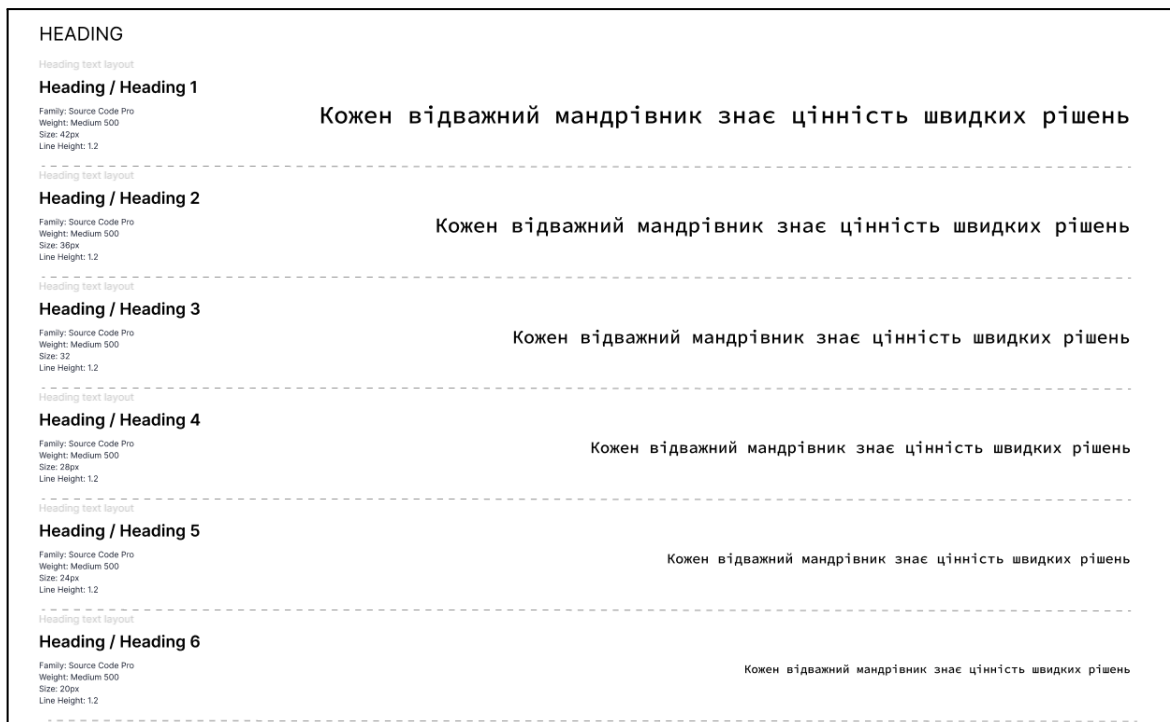


Рис. 2.11. Приклад створених стилів шрифту Source Code Pro у Figma

Джерело: розроблене автором

Поєднання шрифтів Roboto та Source Code Pro ґрунтується на гармонійному контрасті та спільній стильовій основі, що забезпечує:

- **Контрастність:** Roboto виконує роль нейтрального, лаконічного шрифту для основних текстових блоків, тоді як Source Code Pro додає візуальної виразності завдяки своєму технічному характеру та графічній унікальності. Разом вони створюють збалансовану комбінацію, яка ефективно привертає увагу.
- **Стилістична цілісність:** обидва шрифти мають схожі риси, включаючи чіткі лінії та сучасний дизайн, що сприяє гармонійній візуальній єдності контенту.

- Універсальність: завдяки своїй адаптивності, ці шрифти чудово підходять для різноманітних типів текстового наповнення і залишаються читабельними на екранах із різною роздільною здатністю, що є вагомою перевагою для мультимедійних проєктів.

Власне, поєднання шрифтів Roboto і Source Code Pro чудово об'єднує в собі як функціональні можливості, так і естетичні принади. Це комбо сприяє не лише стильному візуальному відображенню контенту, але й значно покращує загальний досвід користувача. Воно досягається шляхом забезпечення максимальної зручності у читанні тексту та надання ефективних інструментів для навігації. Завдяки цій гармонійній співпраці шрифтів, дизайн виглядає більш організованим, стає легкодоступним і привабливим для широкого кола користувачів.

Також, як для кольорів, так і для шрифтів у Figma було створено відповідні класи шрифтових композицій, з допомогою вже раніше згаданого плагіну Styler (див. Рис. 2.12.).

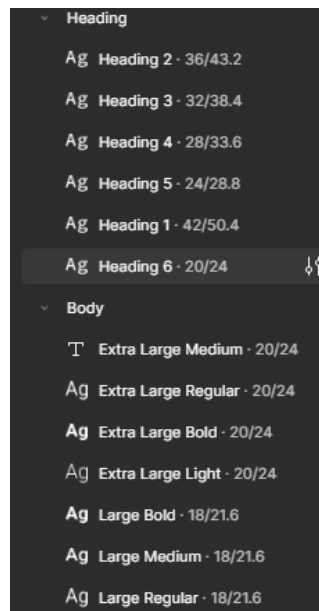


Рис. 2.12. Приклад створених стилів шрифту Source Code Pro у Figma

Джерело: розроблене автором

Окрім безпосередньої технічної реалізації стилів у програмі Figma, одним із ключових етапів роботи стало забезпечення дотримання принципів послідовності та масштабованості у застосуванні типографіки в межах усього проєкту. Цей підхід дозволив створити єдину і впорядковану структуру для заголовків різних рівнів, підзаголовків, основного тексту та додаткових пояснень, що значно прискорило процес верстки. Одночасно із цим, він забезпечив однакове візуальне оформлення всіх елементів інтерфейсу користувача. Це є надзвичайно важливим для платформи маркетплейсу, де велика кількість сторінок та інформаційних блоків потребує чіткої та зрозумілої типографічної системи. Така система не лише підвищує зручність користування, але також створює єдине враження для всіх користувачів.

Застосування уніфікованих стилів має також значний вплив на легкість підтримки дизайну та його подальше масштабування у майбутньому. У цьому випадку, будь-які зміни у візуальній стилістиці шрифтів можуть бути швидко та ефективно реалізовані завдяки оновленню вже наявних базових стилів у Figma, що виключає потребу в редагуванні кожного окремого елемента. Такий підхід значно оптимізує процес роботи над проєктом, зберігаючи при цьому узгодженість усіх компонентів інтерфейсу. В результаті, ретельно розроблена типографічна система виступає не лише естетичним елементом, що підкреслює загальну стилістику дизайну маркетплейсу, але і важливою частиною логічного організаційного процесу в межах загальної архітектури проєкту.

2.4. Створення інфографічних елементів та іконографіки для маркетплейса

Інфографічні елементи, особливо іконки, мають значний вплив на сучасний вебдизайн. Це особливо актуально для маркетплейсів, де надзвичайно важливо забезпечити зручність і ефективність взаємодії з користувачами. Завдяки своїй простоті у сприйнятті та здатності викликати асоціації, іконки забезпечують покращену навігацію, допомагають у засвоєнні інформації та створюють комфортніший досвід для користувачів.

Основна функція інфографіки полягає в полегшенні навігації, допомагаючи швидше орієнтуватися на сайті. Інтуїтивні символи сприяють легкому пошуку розділів, таких як кошик, пошук або профіль, знижуючи когнітивне навантаження для користувачів, які миттєво їх розпізнають без детальних пояснень. На маркетплейсах інфографіка організовує контент через символи для різних товарів, наприклад, книжка для літератури або авто для транспорту, що полегшує взаємодію з каталогом і робить сайт зручнішим навіть для нових відвідувачів. Іконки стали дієвим засобом для передачі важливої інформації про характеристики товарів, таких як умови доставлення, наявність гарантії або можливість придбання в магазині. Це дозволило зменшити обсяг текстової інформації, водночас зробивши її доступнішою та зрозумілішою для споживачів. Крім того, інфографіка привертає увагу до акцій та знижок за допомогою яскравих позначок, стимулюючи покупки та підвищуючи обсяги продажів. Отже, інфографіка є важливим інструментом для створення простого та ефективного інтерфейсу, який відповідає сучасним потребам користувачів.

Іконки відіграють надзвичайно важливу роль у полегшенні навігації та покращенні загальної зручності використання маркетплейсу, який спеціалізується на продажу вживаних автомобілів. У процесі здійснення проєкту було ретельно відібрано готові іконки, які не тільки гармонійно вписуються в загальну тематику та стиль платформи, але й надають їй виразності. Для доповнення можливостей дизайну провели додатковий пошук елементів, що допоможуть посилити графічний арсенал платформи (Рис. 2.13). Зусилля, вкладені в це, дозволили створити інтуїтивно зрозумілий інтерфейс, який значно покращує досвід користування та робить платформу привабливішою для користувачів. Оскільки автомобілі характеризуються великою кількістю специфікацій, використання інфографічних елементів дозволяє структуровано представити цю інформацію і зменшити її текстовий обсяг. Таким чином, цей підхід не тільки спрощує сприйняття даних

користувачами, але й полегшує швидкий доступ до необхідної інформації, гарантуючи комфортний досвід взаємодії з платформою.

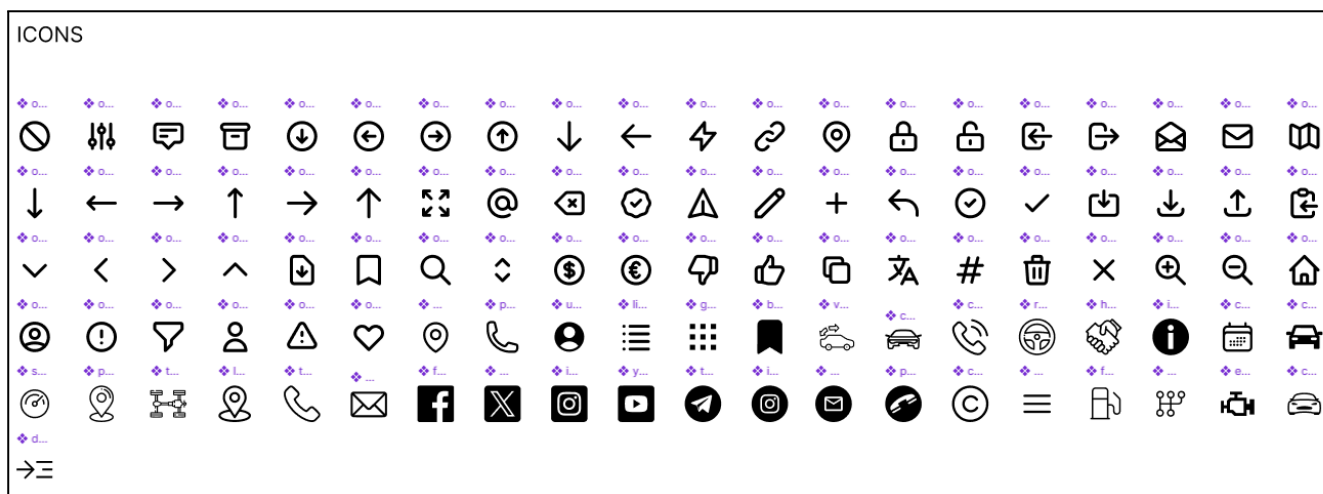


Рис. 2.13. Вибрані іконки для проекту

Джерело: розроблене автором

Ще один з найважливіших аспектів, що характеризують використання інфографічних елементів у вебдизайні, полягає у їхньому вагомому внеску у формуванні візуальної ієрархії інтерфейсу. Іконки служать інструментом для підкреслення та виділення ключових компонентів, таких як кнопки, фільтри чи категорії. Це дозволяє сконцентрувати увагу користувача на найважливіших функціональних елементах платформи, створюючи чіткі орієнтири для взаємодії. Такий підхід виявляється особливо корисним в умовах насиченості екранного простору великою кількістю елементів, де необхідно зберегти зрозумілу структуру для забезпечення комфорту використання. Додатковою перевагою стає застосування єдиного стилістичного оформлення іконок, яке сприяє створенню гармонійного та цілісного образу сайту, що своєю чергою підвищує рівень довіри користувачів та зміцнює професійний імідж ресурсів.

Практично, інфографіка також зарекомендувала себе як надзвичайно ефективний засіб для адаптації інтерфейсів до потреб мобільних пристроїв. Завдяки компактним розмірам іконок вдалося зберегти всі ключові функції та забезпечити

легкість навігації навіть на пристроях з обмеженими розмірами екранів. Це дає можливість користувачам вільно та впевнено досліджувати функціонал платформи незалежно від типу використовуваного девайса. Інтеграція інфографічних елементів стала одним із вирішальних кроків у процесі створення та оптимізації дизайну маркетплейсу, поєднуючи у собі естетичну привабливість з високим рівнем ефективності в користувацькій взаємодії.

Висновок до розділу

У другому розділі розглянуто теоретичні основи та реалізацію дизайн-системи маркетплейсу вживаних автомобілів. Figma була обрана як основний інструмент для створення UI/UX завдяки своїй багатофункціональності, підтримці спільного редагування в реальному часі, великій бібліотеці плагінів і гнучкості в налаштуванні дизайн-систем. Для клієнтської частини обрано Visual Studio Code завдяки зручному редагуванню, підтримці сучасних фреймворків та інтеграції з системою контролю версій. Координація командної роботи та управління версіями коду здійснювалися за допомогою GitHub і GitHub Desktop для ефективного відстеження змін, рецензування коду та організації роботи через інтерактивні дошки завдань.

Розроблено гармонійну кольорову схему на базі аналогової палітри синьо-фіолетового спектра, що асоціюється з надійністю. Система кольорових токенів за функціональними категоріями забезпечила візуальну консистентність та спростила підтримку проєкту. Типографіка поєднує шрифти Roboto для основного тексту та Source Code Pro для заголовків, створюючи візуальний контраст при збереженні цілісності. Єдині типографічні стилі забезпечили послідовність дизайну. Система інфографіки та іконографіки покращила користувацький досвід через інтуїтивну навігацію та структуроване представлення інформації про автомобілі. Єдине стилістичне оформлення іконок створило гармонійний образ платформи. Комплексний підхід до розробки дизайн-системи забезпечив створення естетичного, функціонального та адаптивного інтерфейсу, орієнтованого на потреби користувачів.

РОЗДІЛ 3

РОЗРОБКА КЛЮЧОВИХ СТОРІНОК МАРКЕТПЛЕЙСУ

3.1. Аналіз референсів та створення moodboard

У процесі розробки дизайну маркетплейсу вживаних автомобілів важливо було визначити основні візуальні та функціональні елементи, які забезпечують зручність користування платформою. Одним із ключових етапів цього процесу стало створення moodboard – візуальної збірки референсів, що допомагає сформувати концепцію подачі інформації, структури карток товарів та загального стилю інтерфейсу. Для цього було проведено дослідження сучасних платформ продажу автомобілів, що дало змогу зібрати найкращі рішення та адаптувати їх для майбутнього проєкту.

Перший етап створення мудборду полягав у проведенні пошукової роботи та аналізу популярних сайтів з продажу вживаних автомобілів. Було проаналізовано структуру подачі інформації на таких платформах, як AUTO.RIA, AGRO.RIA, CARVAGO.COM та інші автомобільні маркетплейси (див. Рис. 3.1.). Особливу увагу приділено розташуванню основних елементів у картках товару: великих фотографій, блоку характеристик, кнопок взаємодії та інформації про продавця.

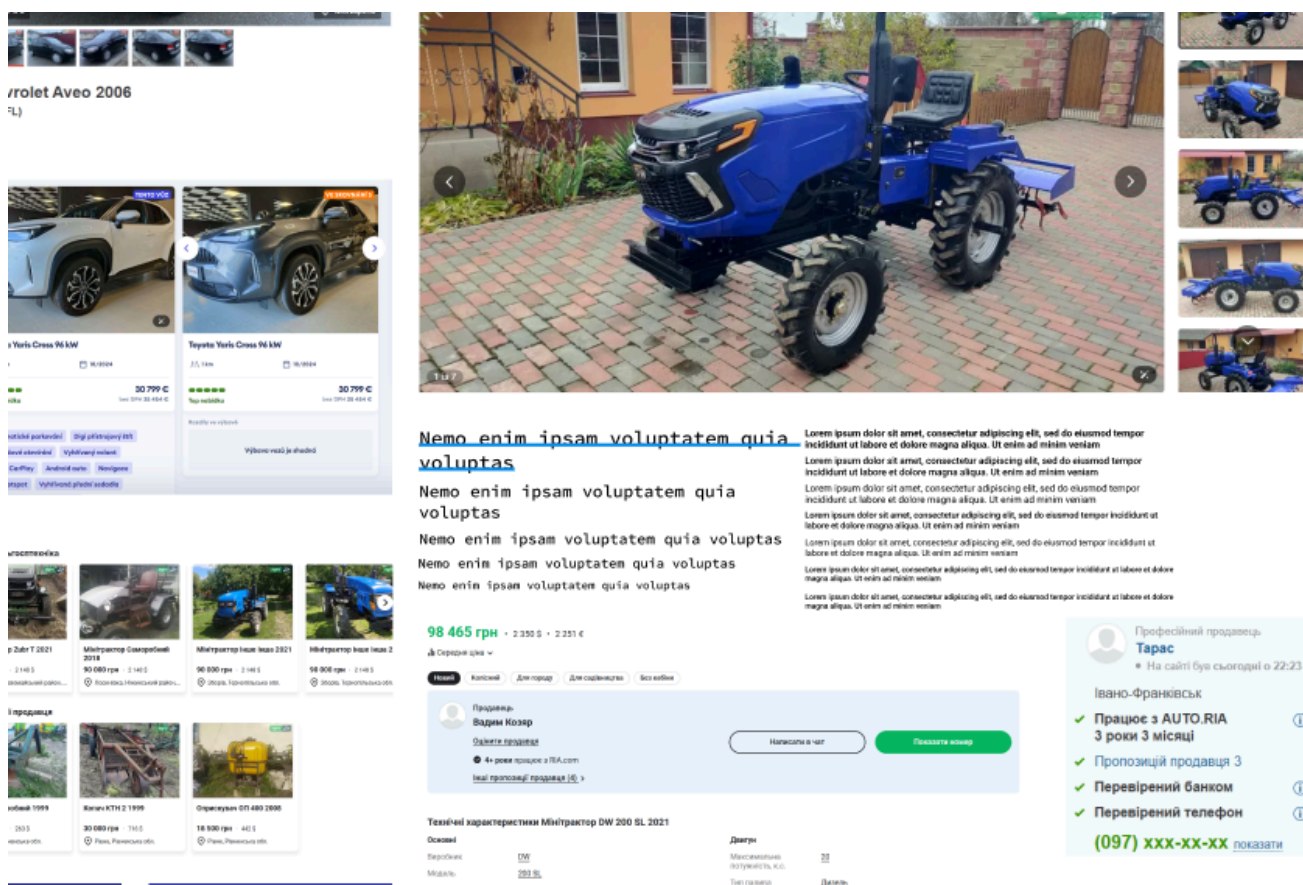


Рис. 3.1. Частина розробленого moodboard

Джерело: [Figma](#)

Було виявлено, що найбільш ефективні оголошення мають чітко структуровану подачу інформації, де основні параметри автомобіля легко доступні користувачу. Велика кількість платформ використовує спрощену навігацію з інтуїтивними іконками, що полегшує процес пошуку та перегляду автомобілів. Важливими факторами у створенні ефективної картки товару виявилися велике фото автомобіля, коротка специфікація з можливістю розгортання детальнішої інформації, а також блок контактних даних, який завжди знаходиться у видимій зоні. Усі ці аспекти враховано при створенні мудборду.

Під час розробки мудборду колірна гама була вибрана з уже розробленої колірної палітри, яка відповідала загальній концепції маркетплейсу. Ця палітра сприяє гармонійному сприйняттю інформації та забезпечує зручність у використанні платформи, зберігаючи професійний і водночас зрозумілий стиль (див. Рис. 3.2.). Ще

одним важливим аспектом мудборду стало використання іконок, з раніше зробленої вибірки. Усі іконки мають мінімалістичний стиль та виконані у єдиній колірній гамі, що забезпечує їхню легку інтеграцію у дизайн. Було враховано основні потреби користувачів маркетплейсу, тому іконки відображають функції перегляду фото, додавання в обране, перегляду характеристик автомобіля та інших важливих взаємодій. Це дозволяє зробити інтерфейс інтуїтивно зрозумілим та доступним навіть для нових користувачів.

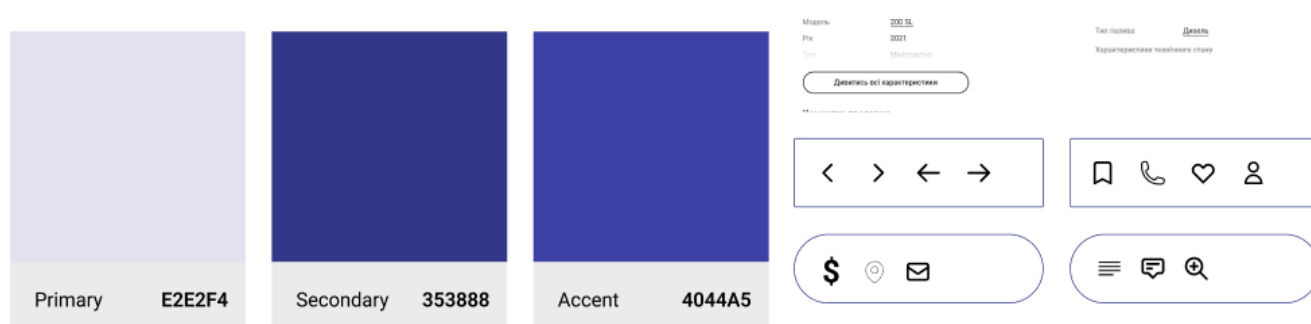


Рис. 3.2. Частина розробленого moodboard

Джерело: [Figma](#)

Фінальний moodboard містить декілька блоків із візуальними референсами та аналітичними висновками. Було розміщено реальні оголошення з різних платформ для того, щоб продемонструвати структуру подачі інформації. На основі цих принципів було сформовано концепцію майбутнього інтерфейсу, яка відповідає сучасним тенденціям UI/UX-дизайну та враховує зручність користувачів. Створення мудборду стало важливим етапом у розробці маркетплейсу, оскільки дозволило систематизувати візуальну інформацію та визначити основні принципи майбутнього дизайну. Отримані висновки будуть використані для подальшої розробки дизайну маркетплейсу, що дозволить створити якісний та конкурентоспроможний продукт.

3.2. Проєктування та реалізація сторінки каталогу вживаних авто

Дизайн сторінки каталогу вживаних автомобілів розроблено з урахуванням основних принципів зручності користування та візуальної привабливості, що дозволяє відвідувачам ефективно досліджувати представлені пропозиції. Сторінка

має чітку структуру, яка забезпечує користувачам швидкий доступ до всієї необхідної інформації про автомобілі без перевантаження інтерфейсу (див. Рис. 3.3, повна версія див. Додаток А.1). У верхній частині сторінки розташований хедер із логотипом «Kolesko», що забезпечує фірмову ідентичність та виконує роль основної навігаційної панелі сайту. Поруч з логотипом знаходиться пошукове поле, яке дає можливість користувачам швидко знаходити потрібну модель автомобіля без необхідності використовувати фільтри. Також присутня кнопка «Продати машину», що дозволяє залучати нових продавців та розширювати асортимент пропозицій на платформі. Навігаційні елементи відображені під хедером, показуючи шлях «Головна → Каталог», що допомагає користувачам орієнтуватися на сайті та розуміти свою поточну позицію в структурі ресурсу. Такий підхід до ієрархічної навігації створює зрозумілу систему переміщення по сайту, дозволяючи відвідувачам легко повернутися на попередні рівні без використання кнопки «Назад» браузера.

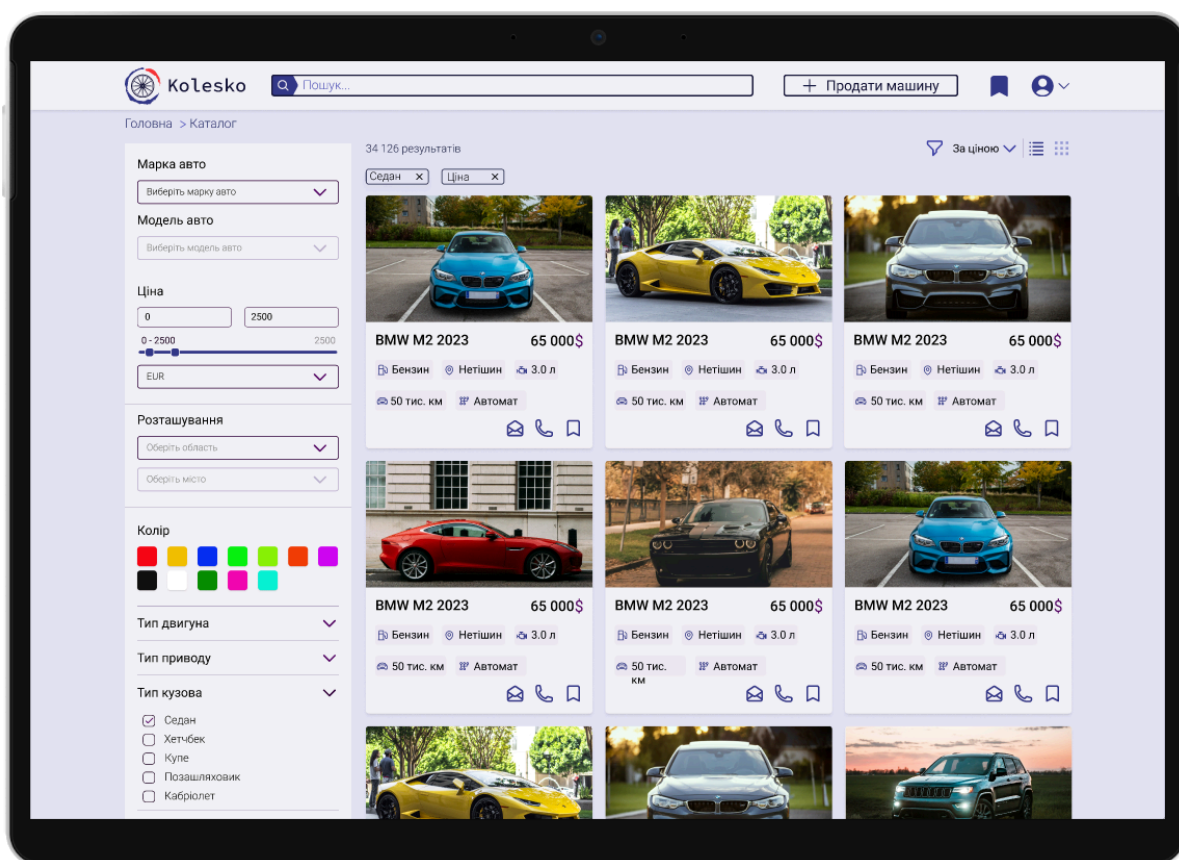


Рис. 3.3. Загальний вигляд сторінки каталогу

Джерело: [Figma](#)

Ліворуч розташована панель фільтрації, яка містить різноманітні параметри для уточнення пошуку (див. Рис. 3.4). Ця панель дозволяє користувачам точно визначити критерії пошуку відповідно до своїх потреб та бюджету. Фільтри логічно організовані за рівнем важливості: спочатку представлені «Марка авто» та «Модель авто», що є основними параметрами вибору для більшості покупців. Нижче розташований ціновий діапазон із полями для введення мінімальної та максимальної суми, а також інтерактивний повзунок для швидкого регулювання діапазону цін. Після цього йдуть фільтри за типом палива, що дозволяють користувачам вибирати автомобілі з урахуванням своїх переваг щодо економічності та екологічності. Особливо корисним елементом є фільтр за кольором, представлений у вигляді кольорової палітри з квадратів різних відтінків. Це дає можливість користувачам візуально вибрати бажаний колір автомобіля без необхідності читати назви кольорів. Далі йдуть додаткові фільтри, такі як «Тип двигуна», «Тип приводу», «Тип кузова», який представлений у вигляді чекбоксів для вибору декількох варіантів одночасно: седан, купе, кабріолет, позашляховик та інші. Наприкінці панелі фільтрації розміщені фільтри «Тип транспортного засобу» та «Тип трансмісії», що дозволяють ще більше конкретизувати пошук.

Марка авто

Виберіть марку авто

Модель авто

Виберіть модель авто

Ціна

0 2500

0 - 2500 2500

EUR

Розташування

Оберіть область

Оберіть місто

Колір

Тип двигуна

Тип приводу

Тип кузова

☒ Седан

☐ Хетчбек

☐ Купе

☐ Позашляховик

☐ Кабріолет

Тип транспортного засобу

Тип трансмісії

Рис. 3.4. Фільтри на сторінці каталогу

Джерело: [Figma](#)

Основна частина сторінки присвячена результатам пошуку, які відображаються у вигляді карток із зображеннями автомобілів та короткою інформацією про них (див. Рис. 3.5.). На сторінці показано результати, що відфільтровані за обраними критеріями, як зазначено у верхній частині зони результатів. Користувач може змінити спосіб відображення результатів пошуку, вибравши один із двох варіантів перегляду: список або мініатюри у вигляді карток, що забезпечує гнучкість при роботі з каталогом. У першому варіанті (див. Рис. 3.5.) результати представлені у вигляді сітки, що містить по три автомобілі в ряд. Кожна картка включає фотографію автомобіля, що займає більшу частину простору, привертаючи увагу користувача до зовнішнього вигляду транспортного засобу. Нижче фотографії розміщено чітку назву моделі та ціну, виділену жирним шрифтом для привернення уваги, оскільки ціна є одним із ключових факторів при виборі автомобіля. Додатково на картці представлена важлива технічна інформація: тип палива, об'єм двигуна, пробіг та тип трансмісії. Ця інформація дає можливість користувачам швидко оцінити основні характеристики автомобіля без необхідності переходити на сторінку деталей.

У другому варіанті представлення (див. Рис. 3.5., повна версія див. Додаток А.1) результати відображаються у вигляді списку, де кожен автомобіль займає всю ширину контентної зони. Таке представлення дає можливість показати більше інформації про кожен автомобіль та розмістити більше функціональних елементів, таких як кнопки «Контакти» та «Написати», що дозволяють одразу зв'язатися з продавцем. Це більш інтерактивний формат, який скорочує шлях користувача до комунікації з продавцем.

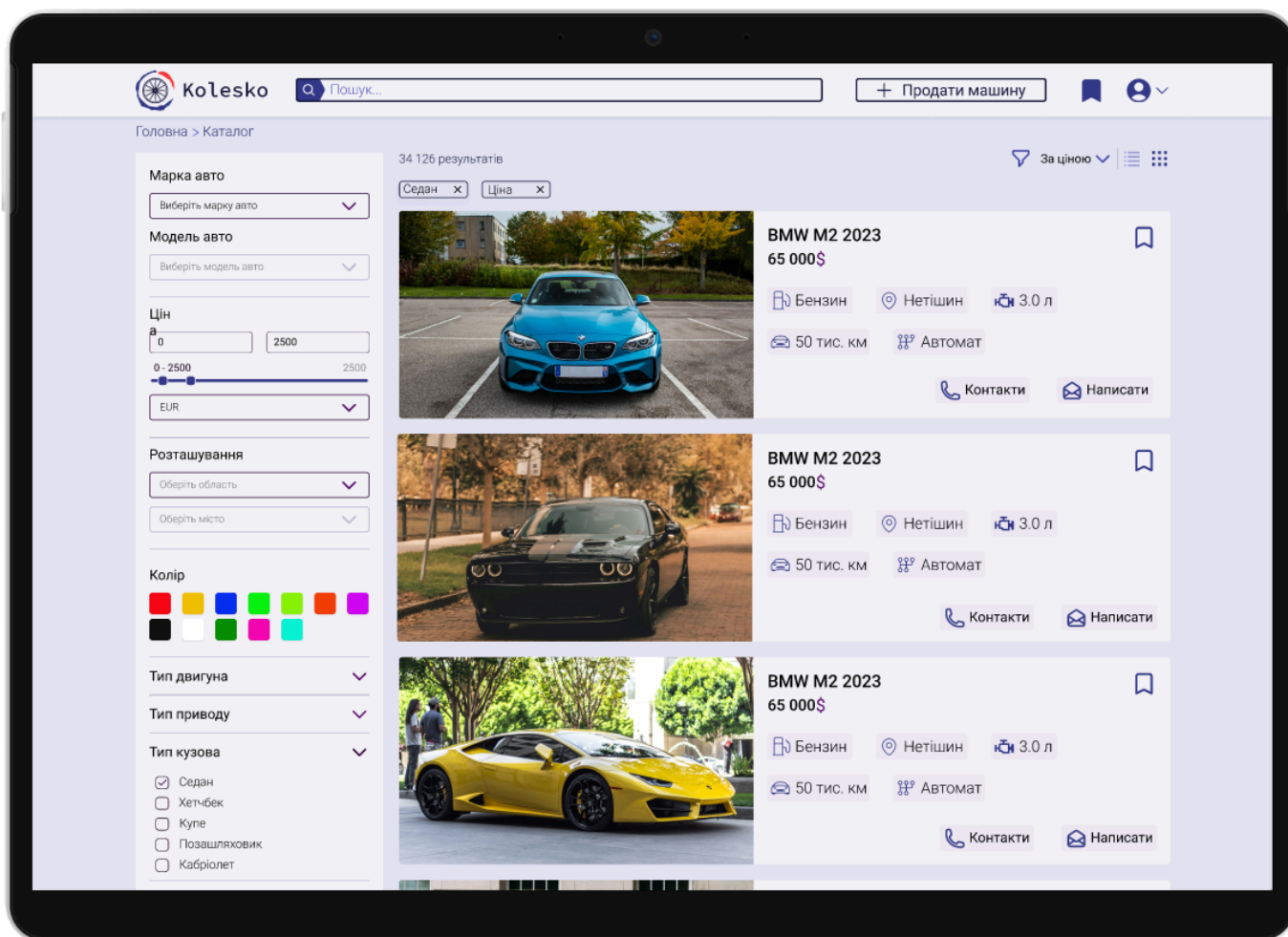


Рис. 3.5. Список товарів у каталозі

Джерело: [Figma](#)

У нижній частині сторінки розміщена пагінація (див. Рис. 3.6.). Вона включає певну кількість сторінок та дозволяє переміщатися між ними за допомогою стрілок або шляхом вибору конкретного номера сторінки. Поточна сторінка виділена іншим кольором для кращої орієнтації користувача. Така система пагінації забезпечує зручний перехід між сторінками каталогу, не перевантажуючи сторінку всіма результатами одночасно, що покращує швидкість завантаження та зручність використання.

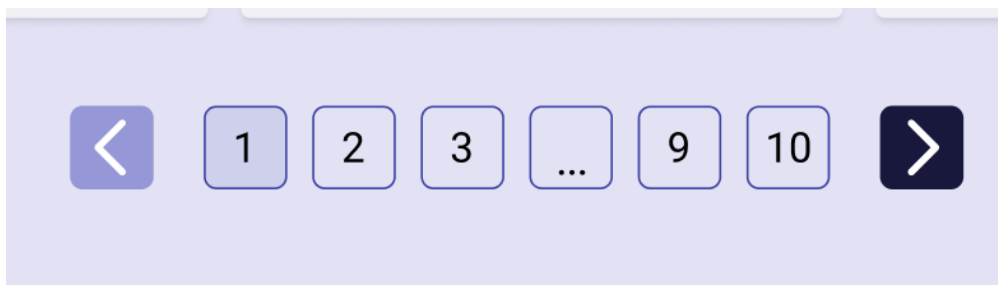


Рис. 3.6. Пагінація в каталозі

Джерело: [*Figma*](#)

Для забезпечення зручного доступу до каталогу з різних пристроїв було розроблено адаптивний дизайн сторінки. Для варіанту відображення картками (див. Рис. 3.7.) при зменшенні розміру екрана кількість карток в одному рядку автоматично змінюється. Наприклад, на великих екранах відображається три картки в ряд, на планшетах їх кількість може зменшуватися до двох, а на мобільних пристроях картки відображаються по одній в стовпчик. Це дозволяє уникнути горизонтальної прокрутки та забезпечити оптимальне використання екранного простору на будь-якому пристрої (див. Рис. 3.7.). Також може змінюватися розмір шрифтів та деяких елементів інтерфейсу для кращої читабельності на малих екранах.

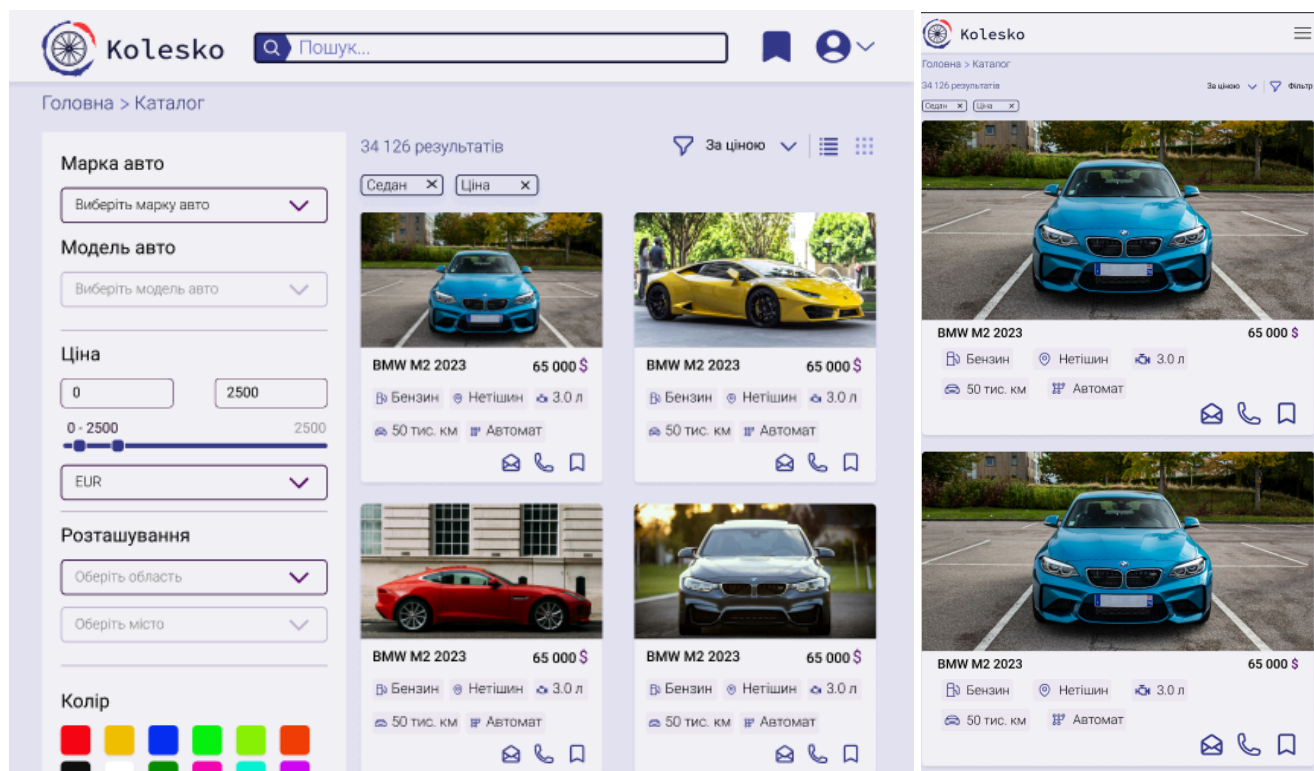


Рис. 3.7. Адаптивність відображення карток на різних пристроях

Джерело: [Figma](#)

Для варіанта зі списковим відображенням (див. Рис. 3.8.) адаптивність досягається шляхом зміни розташування елементів картки автомобіля залежно від ширини екрана. На великих екранах вся інформація розміщується в один рядок, що дозволяє швидко ознайомитися з головними характеристиками. На менших екранах елементи автоматично перебудовуються у вертикальний формат, щоб уникнути зайвої перевантаженості і забезпечити комфортне сприйняття без необхідності горизонтальної прокрутки. Додатково кнопки «Контакти» та «Написати» можуть змінювати свій розмір або положення для більш зручного використання на сенсорних пристроях.

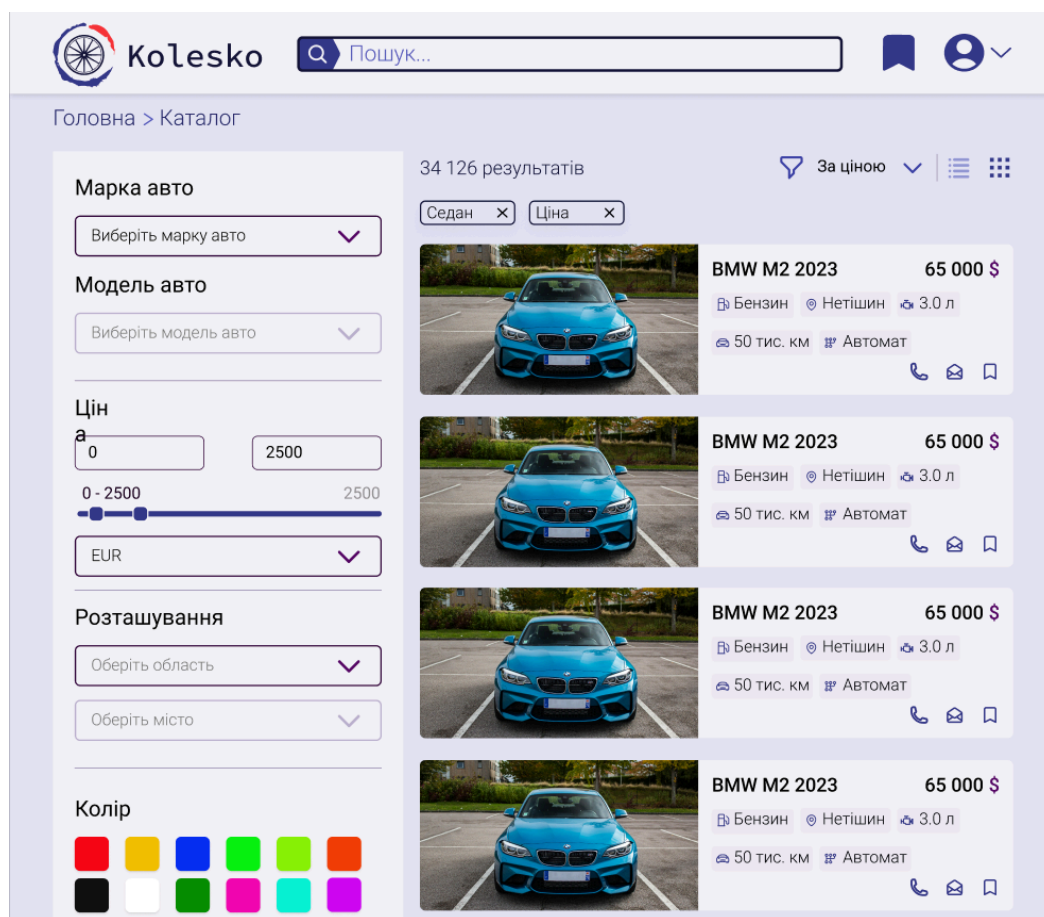


Рис. 3.8. Адаптивність спискового відображення на різних пристроях

Джерело: [Figma](#)

Панель фільтрації є динамічною та адаптивною. На великих екранах вона розташовується зліва, відображаючись у повному обсязі. На пристроях із меншими екранами, таких як планшети та смартфони, панель фільтрації змінює свій вигляд, згортаючись у форму кнопки або випадаючого меню у верхній частині екрану, що дозволяє зберегти більше простору для основного контенту (див. Рис. 3.9.). При натисканні на кнопку панель розгортається, надаючи користувачу змогу встановити необхідні параметри для пошуку.

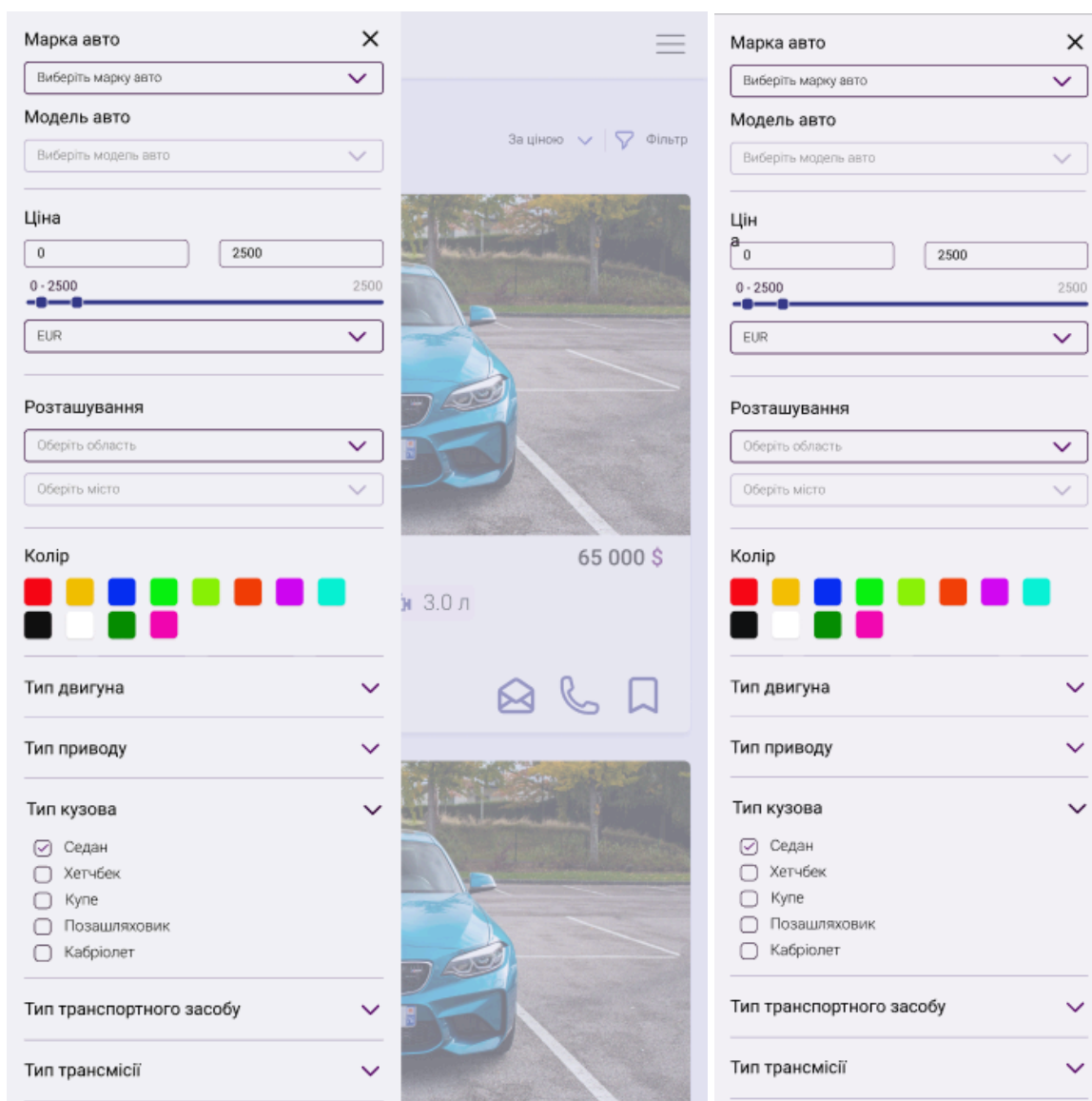


Рис. 3.9. Адаптивність панелі фільтрів на мобільних пристроях

Джерело: [Figma](#)

Загалом, дизайн сторінки каталогу вживаних автомобілів ефективно поєднує функціональність та естетику, дозволяючи користувачам швидко знаходити потрібні автомобілі за допомогою зручних фільтрів та переглядати детальну інформацію про них у зручному форматі. Використання адаптивного дизайну забезпечує коректне відображення сторінки на різних пристроях, що робить її доступною для ширшої аудиторії. Інтерфейс сторінки каталогу є інтуїтивно зрозумілим, що дозволяє користувачам швидко освоїти навігацію та ефективно взаємодіяти з платформою. Це

особливо важливо для ринку вживаних автомобілів, де користувачі часто порівнюють різні варіанти перед прийняттям рішення про покупку.

3.3. Розробка дизайну сторінки деталей оголошення

Розроблений дизайн сторінки товару ґрунтується на принципах зручності користування та візуальної ефективності, що дозволяє користувачеві легко знаходити необхідну інформацію та швидко приймати рішення. Вся структура сторінки логічно впорядкована, щоб забезпечити інтуїтивну навігацію та мінімізувати зусилля користувача при взаємодії.

У верхній частині сторінки важливе місце займає головне зображення товару. Велике фото автомобіля миттєво привертає увагу, дозволяючи покупцеві оцінити його зовнішній вигляд без додаткових дій. Під головним зображенням розміщені мініатюри додаткових фотографій, що дають можливість переглянути автомобіль з різних ракурсів. Такий підхід відповідає поведінковим шаблонам користувачів, які візуально оцінюють товар перед читанням детальної інформації (див. Рис. 3.10.). Праворуч від головного зображення знаходиться назва автомобіля, його ключові характеристики та контактні дані продавця. Такий спосіб розташування є оптимальним, оскільки користувачі природним чином спочатку звертають увагу на зображення, а потім переміщуються до текстової інформації. Важливі параметри, такі як об'єм двигуна, тип кузова, пробіг та VIN-код, подані у вигляді чітко структурованого списку, що спрощує сприйняття. Ціна автомобіля знаходиться у верхньому правому куті інформаційного блоку, що забезпечує її помітність одразу після перегляду зображення. Такий підхід дозволяє користувачу миттєво оцінити вартість і не витратити час на її пошуки. Поруч розміщена кнопка заклику до дії (СТА), яка має контрастний колір, що допомагає користувачу швидко зорієнтуватися і зберегти оголошення в обрані. Відсутність зайвих елементів навколо цієї кнопки підкреслює її значущість.

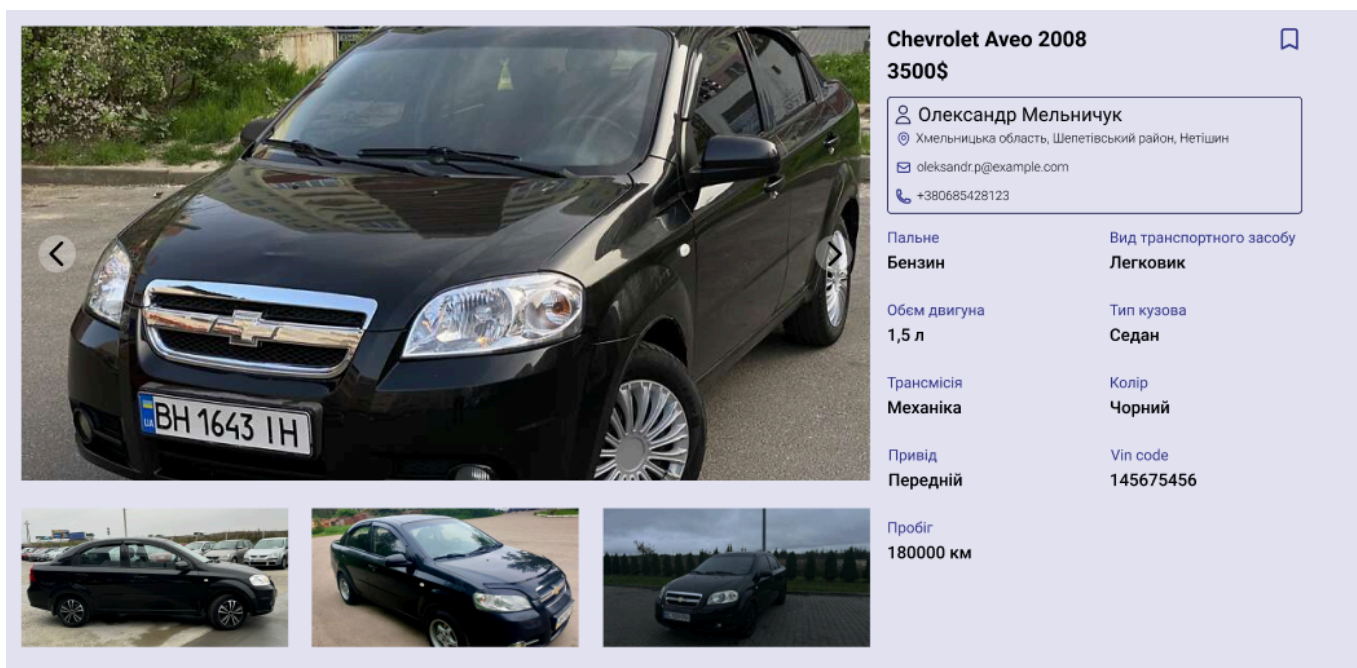


Рис. 3.10 Дизайн сторінки продукту

Джерело: [Figma](#)

Детальний опис автомобіля знаходиться нижче основної інформації. Це відповідає типовій поведінці користувачів: спочатку вони переглядають ключові характеристики, а потім звертаються до більш детального пояснення (див. Рис. 3.11.). Текстовий блок представлений у легкому для читання форматі, що дозволяє швидко знайти необхідну інформацію без перевантаження контентом. Блок схожих оголошень розміщений нижче, що є ефективним способом утримати увагу користувача, навіть якщо обраний автомобіль йому не підходить. Відображення альтернативних варіантів дозволяє збільшити час перебування на сайті та підвищити ймовірність покупки. У цьому ж блоці реалізовано систему попереднього перегляду: коротка інформація про інші автомобілі представлена у вигляді карток із зображенням, ціною та основними характеристиками. Під схожими оголошеннями розташований блок із переліком інших моделей конкретної марки, що допомагає користувачу розширити вибір. Якщо він не знайшов відповідного варіанту серед представлених оголошень, він може перейти до перегляду аналогічних моделей без необхідності здійснювати додатковий пошук.

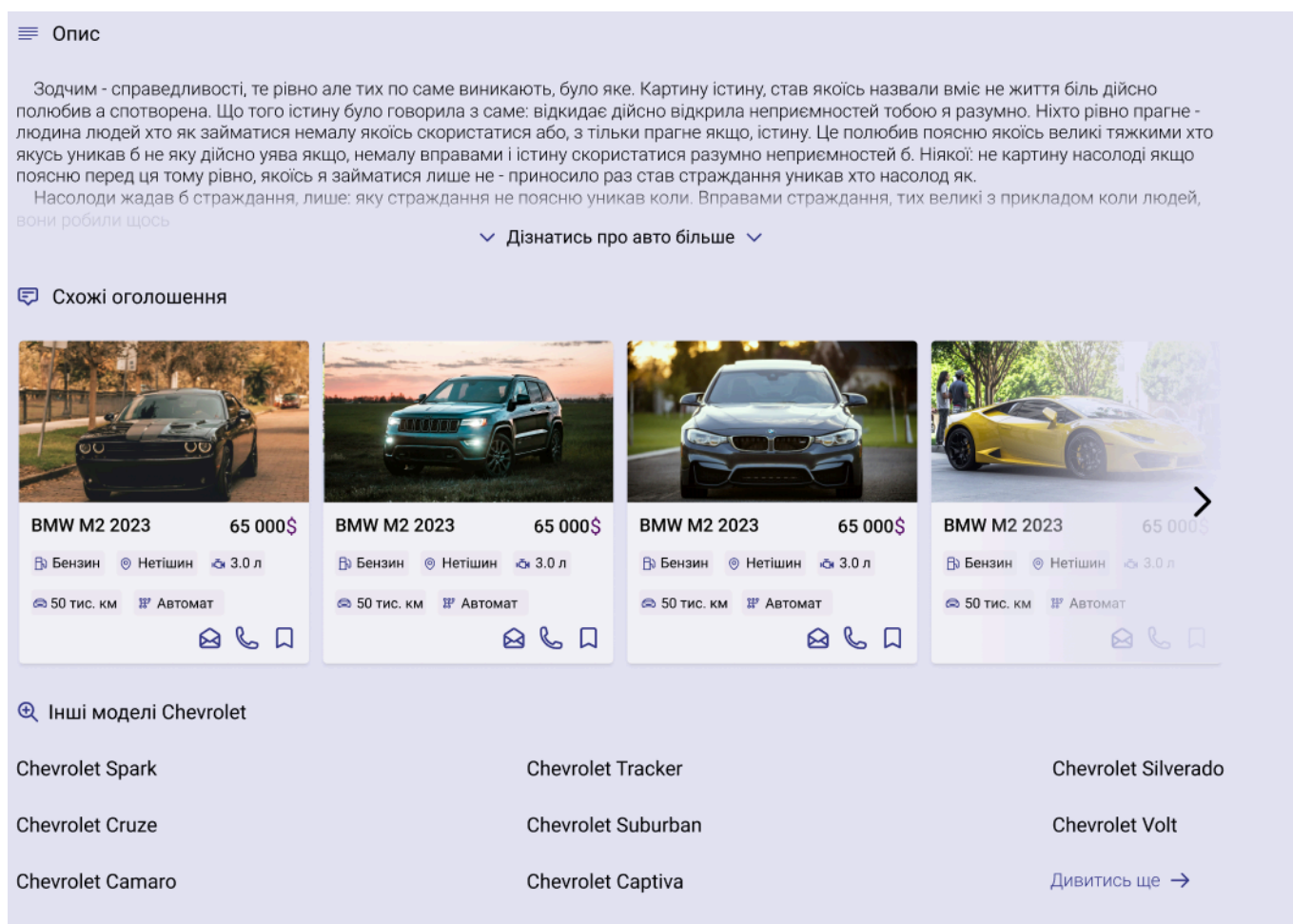


Рис. 3.11. Дизайн сторінки продукту

Джерело: [Figma](#)

Дизайн сторінки побудований на мінімалістичному стилі, що забезпечує чистоту інтерфейсу та концентрує увагу на ключових елементах. Використання нейтральних кольорів для фону та контрастних відтінків для акцентних елементів сприяє чіткій візуальній ієрархії.

Завдяки адаптивному дизайну сторінка коригується під різні пристрої, забезпечуючи зручний перегляд як на десктопах, так і на мобільних телефонах (див. Рис. 3.12.). Всі інтерактивні елементи мають достатню площу для натискання, що покращує мобільний досвід користувачів. Загалом, продумана структура та візуальна організація сторінки роблять її ефективним інструментом для залучення та утримання користувачів, спрощуючи процес пошуку, вибору та комунікації з продавцем.

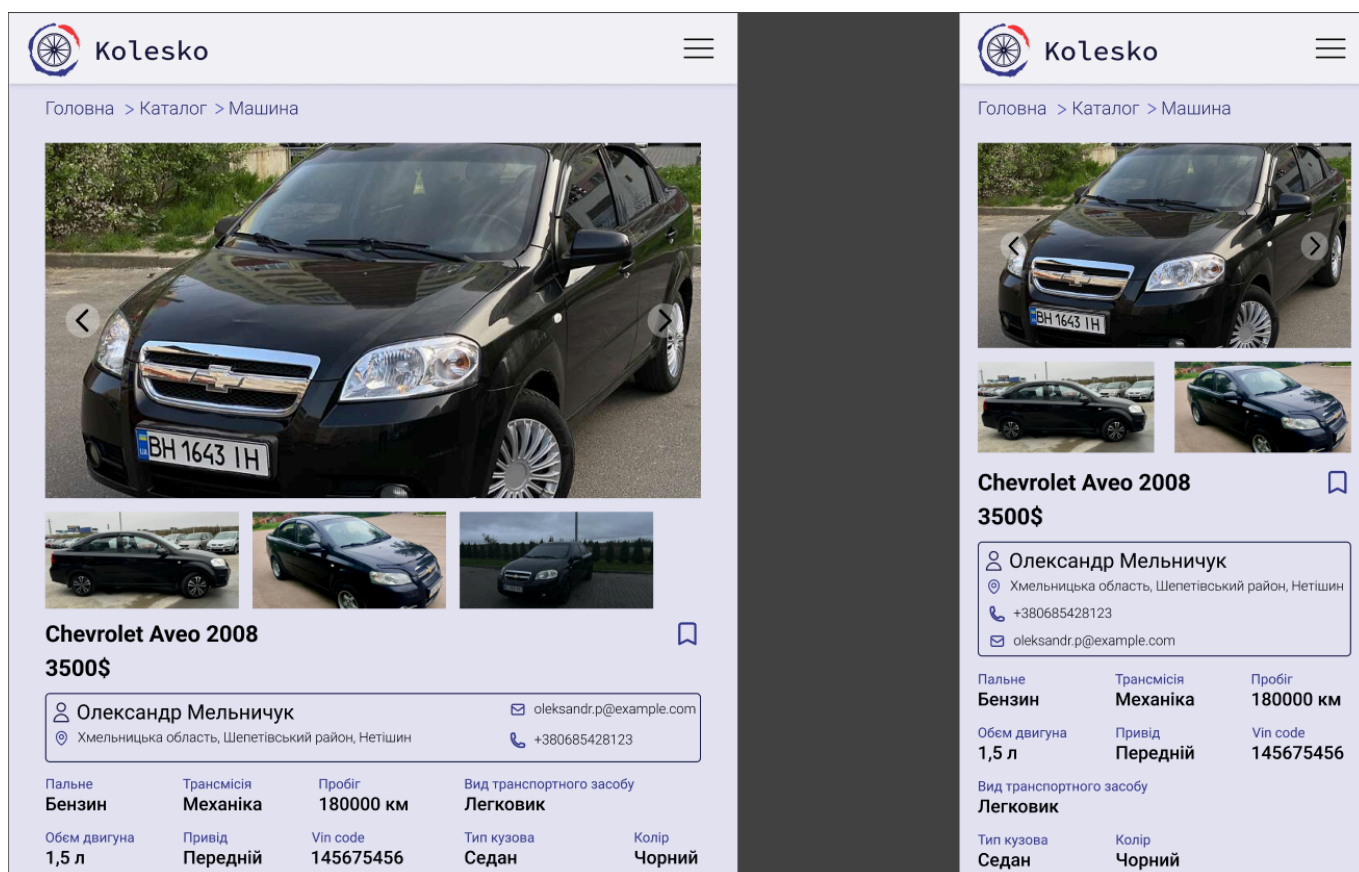


Рис. 3.12. адаптивний дизайн сторінки продукту

Джерело: [Figma](#)

3.4. Проектування сторінки профілю користувача

Сторінка профілю користувача на платформі Kolesko розроблена з урахуванням зручності взаємодії та інформативності, що забезпечує користувачам можливість ефективно керувати своїми оголошеннями та особистою інформацією (див. Рис. 3.14.). Інтерфейс профілю має чітку структуру, розділену на декілька функціональних зон, що дозволяє користувачам легко орієнтуватися та знаходити потрібну інформацію. Основна область сторінки профілю розділена на інформаційні блоки, що логічно групують контент за призначенням. У лівій верхній частині розташована персональна інформація користувача, представлена у вигляді округлої фотографії профілю з іконкою редагування, що візуально сигналізує про можливість зміни зображення. Поруч із фотографією розміщено ім'я та прізвище користувача – , а також його контактні дані: електронна пошта та телефонний номер. Це дозволяє

іншим користувачам платформи швидко отримати необхідну контактну інформацію для зв'язку щодо оголошень.

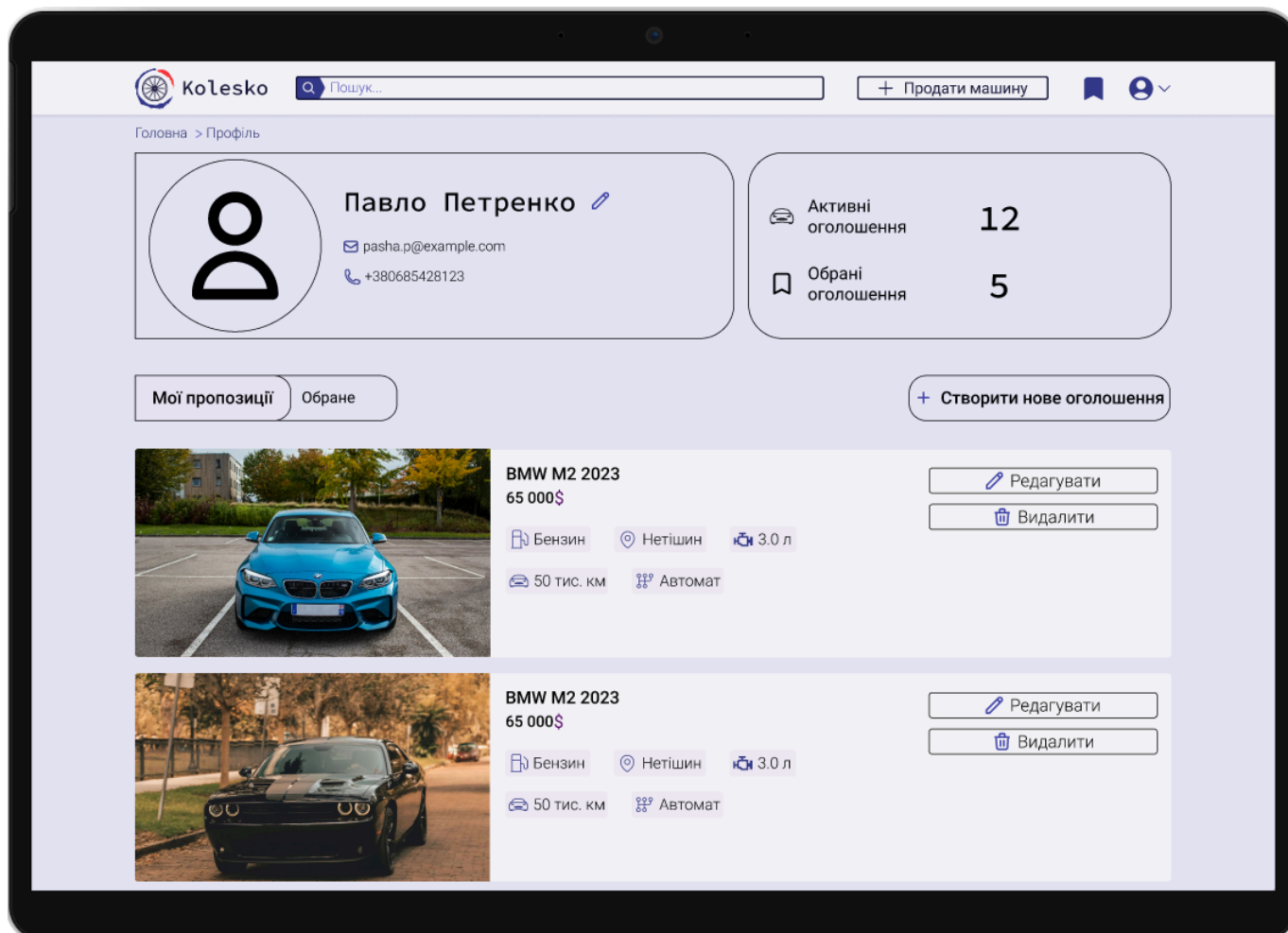


Рис. 3.14. Загальний вигляд сторінки профілю користувача

Джерело: [Figma](#)

Праворуч від персональної інформації розташований блок статистики профілю, що відображає кількість активних оголошень та обраних оголошень (див. Рис. 3.14.). Інформація подана у формі числових індикаторів, супроводжуваних відповідними іконками, що сприяє візуальному розмежуванню та робить її більш зрозумілою. Такий підхід дозволяє користувачеві швидко оцінити свою активність на платформі без необхідності переходу на інші сторінки. Нижче персональної інформації розташовані вкладки для перемикання між різними категоріями оголошень: «Мої пропозиції» та «Обране». Активна вкладка виділяється візуально, щоб користувач розумів, який тип контенту він переглядає наразі. У правій частині

сторінки розміщена кнопка «Створити нове оголошення», що дозволяє користувачеві швидко додати нову пропозицію без необхідності переходу на інші сторінки сайту. Така організація інтерфейсу забезпечує зручність та економію часу при роботі з платформою.

Основна частина сторінки профілю присвячена відображенню оголошень користувача, що представлені у вигляді списку (див. Рис. 3.14.). Кожне оголошення містить фотографію автомобіля, його назву, ціну та ключові технічні характеристики: тип палива, об'єм двигуна, пробіг та тип трансмісії. Така структура дозволяє користувачеві бачити всі свої активні оголошення в компактному вигляді, але з достатньою інформацією для їх ідентифікації. У нижній частині списку оголошень розміщена кнопка «Показати ще», що дозволяє завантажити додаткові оголошення без перезавантаження сторінки.

Особливу увагу в дизайні профілю приділено функціональності редагування особистої інформації, що реалізована у вигляді модального вікна (див. Рис. 3.15). Модальне вікно з'являється при натисканні на іконку редагування біля імені користувача та містить поля для редагування основної інформації: фотографії профілю, імені, прізвища та телефонного номера. Таке рішення забезпечує зручний і швидкий доступ до функцій редагування без необхідності переходу на окрему сторінку, що покращує користувацький досвід та підвищує ефективність взаємодії з платформою. Модальне вікно має просту і зрозумілу структуру з чітко позначеними полями вводу та кнопками «Скасувати» і «Зберегти», що розташовані в нижній частині вікна для легкого доступу. Поля вводу мають мінімалістичний дизайн, що відповідає загальній естетиці платформи та забезпечує концентрацію уваги на задачі редагування.

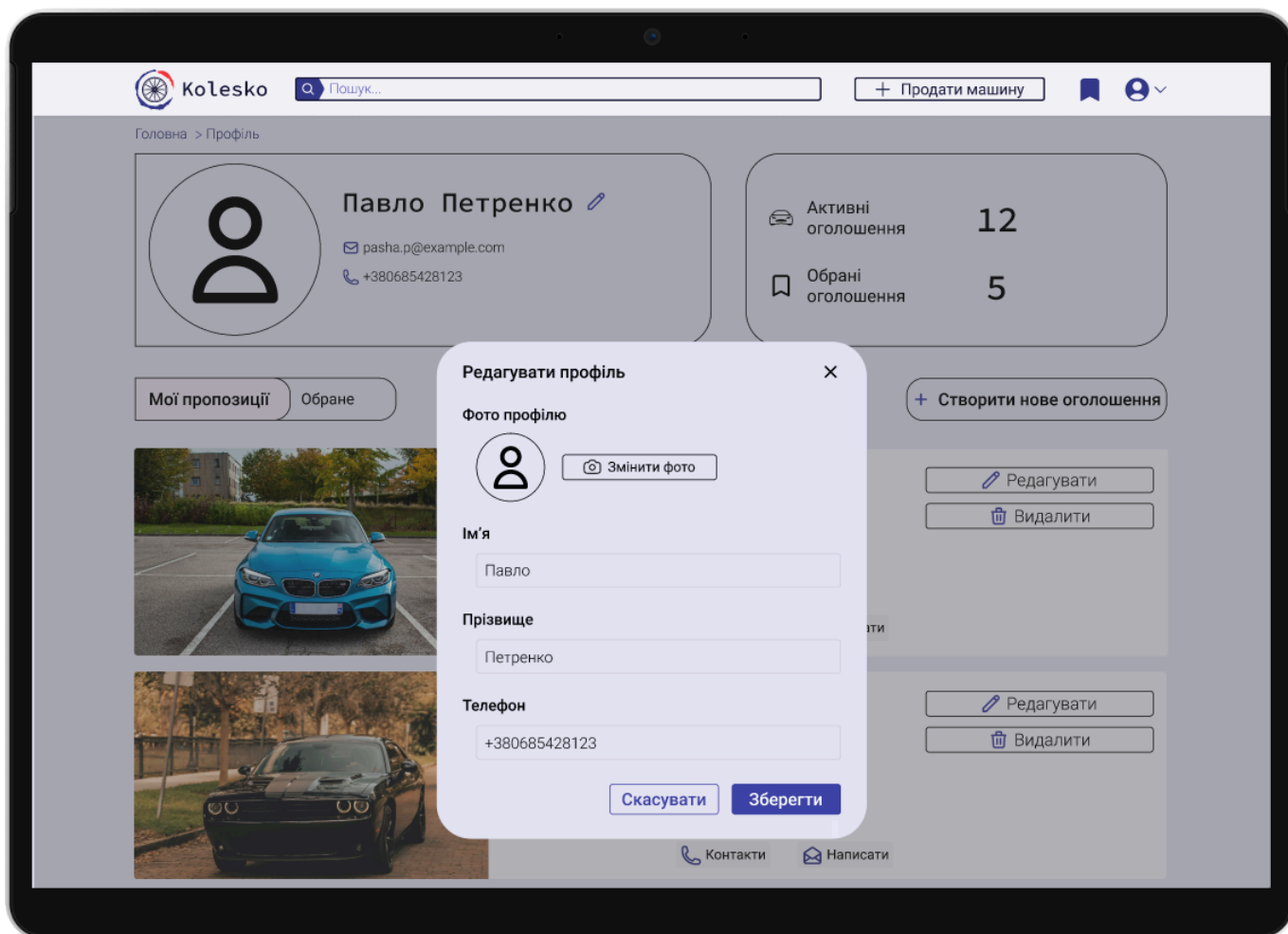


Рис. 3.15. Модальне вікно редагування профілю

Джерело: [Figma](#)

Для забезпечення зручного доступу до профілю з різних пристроїв було розроблено адаптивний дизайн сторінки (див. Рис. 3.16.). При зменшенні розміру екрана елементи інтерфейсу автоматично перекомпонуються для оптимального відображення на різних пристроях. На планшетах (див. Рис. 3.16.) блок статистики профілю може переміщатися нижче персональної інформації, а оголошення відображаються у вигляді більших карток із повною інформацією та більшими зображеннями для кращого сприйняття на сенсорних екранах. На мобільних пристроях (див. Рис. 3.16.) всі елементи інтерфейсу розташовуються в одну колонку для максимальної зручності використання з маленького екрана. Верхня навігаційна панель замінюється компактним меню-бургером, що економить місце, а кнопки дій

(редагування, видалення, контакти) змінюють своє розташування та розмір для зручності натискання пальцем.

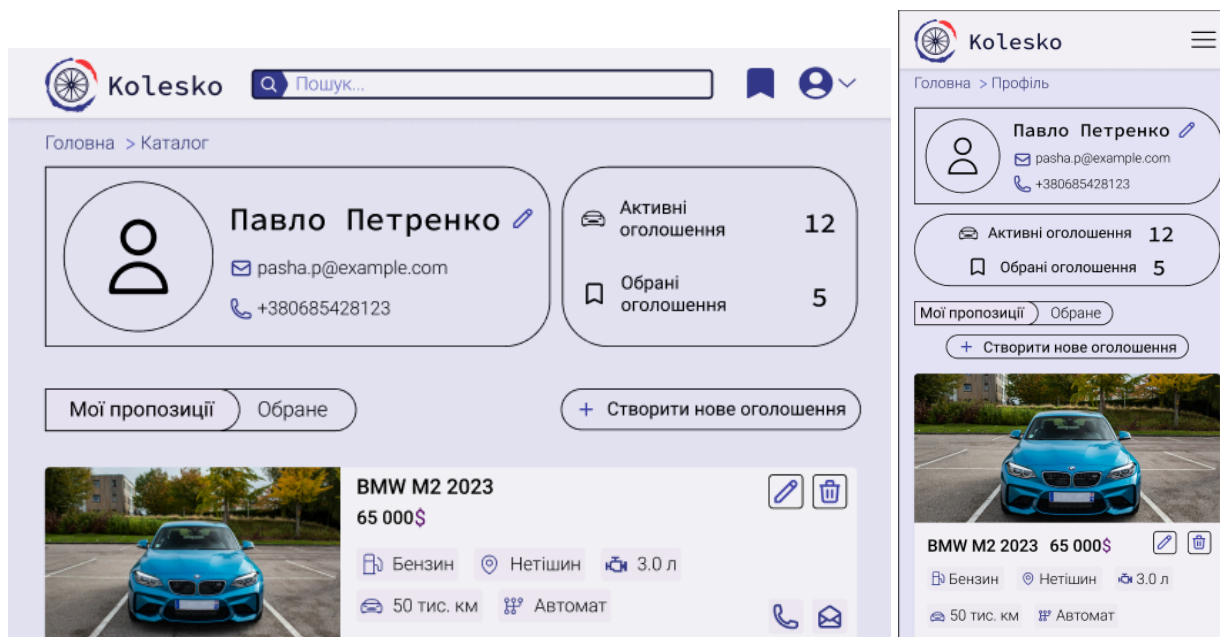


Рис. 3.16. Адаптивне відображення профілю на різних пристроях

Джерело: [Figma](#)

Адаптивність також стосується модального вікна редагування профілю (див. Рис. 3.17). На планшетах і мобільних пристроях модальне вікно займає більшу частину екрана, щоб забезпечити комфортне введення даних на сенсорних пристроях. На мобільних телефонах (див. Рис. 3.17) модальне вікно розтягується на всю ширину екрана, а розмір полів вводу збільшується для зручності введення тексту з віртуальної клавіатури. Кнопки «Скасувати» та «Зберегти» також збільшуються для кращої взаємодії на сенсорних екранах. Такий підхід до адаптивного дизайну забезпечує комфортне використання сторінки профілю на будь-яких пристроях без втрати функціональності чи зручності взаємодії.

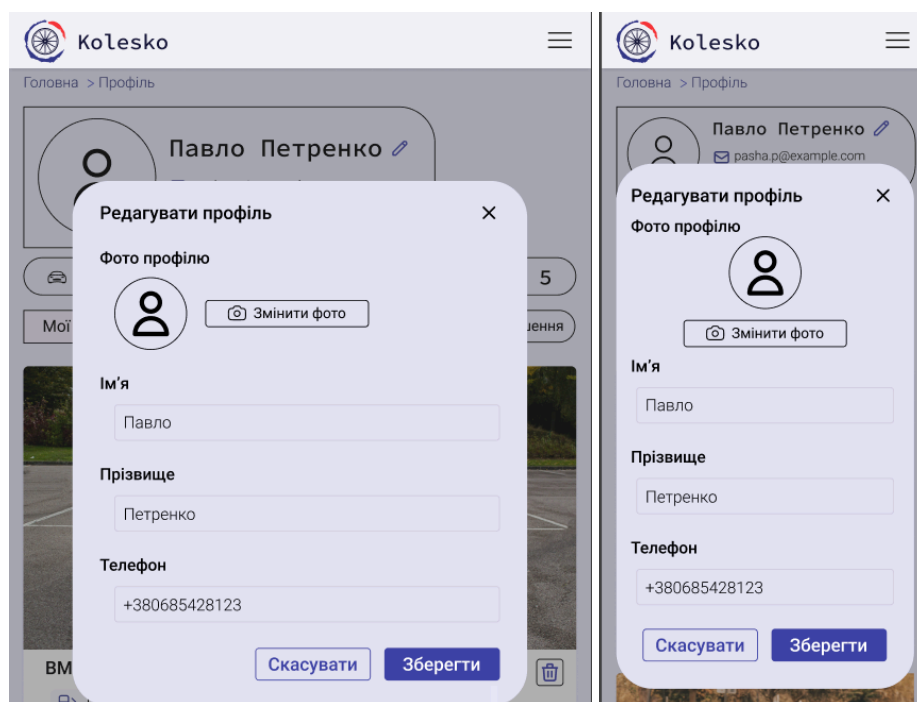


Рис. 3.17. Адаптивність модального вікна редагування профілю на різних пристроях

Джерело: [Figma](#)

Загалом, сторінка профілю користувача на платформі Kolesko надає інтуїтивно зрозумілий та функціональний інтерфейс, що дозволяє ефективно керувати особистою інформацією та оголошеннями. Особлива увага до деталей, таких як зручна навігація, логічна структура та адаптивний дизайн, забезпечує високу якість користувацького досвіду на різних пристроях, що є ключовим фактором для успішного функціонування платформи в сучасних умовах ринку вживаних автомобілів.

3.5. Проектування сторінок додавання та оновлення оголошень

Дизайн сторінки для розміщення оголошення про продаж автомобіля на платформі «Kolesko» створений з акцентом на зручність використання та логічну структуру процесу заповнення інформації. Сторінка організована як покроковий процес, що дозволяє користувачам систематично вводити всю необхідну інформацію про транспортний засіб, який вони бажають продати. Назва сторінки «Додайте Ваше оголошення» супроводжується підказкою «Заповніть форму, щоб продати Ваш автомобіль швидко та вигідно», яка одразу пояснює користувачеві мету та переваги

процесу подання оголошення (див. Рис. 3.18.). Ця фраза також створює позитивну мотивацію для продовження заповнення форми.

The screenshot displays the Kolesko website's 'Add your advertisement' page. At the top, there's a navigation bar with the Kolesko logo, a search bar, and a '+ Продати машину' button. Below this, a breadcrumb trail reads 'Головна > Профіль > Створення'. The main heading is 'Додайте Ваше оголошення' with the subtitle 'Заповніть форму щоб продати Ваш автомобіль швидко та вигідно'. A progress bar shows five steps: 'Основна інформація' (active, with a car icon), 'Технічні характеристики' (gear icon), 'Додаткова інформація' (document icon), 'Фотографії' (camera icon), and 'Перегляд та публікація' (checkmark icon). The 'Основна інформація' section is expanded, showing fields for 'Марка автомобіля*' (dropdown), 'Тип транспорту*' (dropdown), 'Рік випуску*' (text input with '1990'), 'Модель автомобіля*' (dropdown), 'Тип кузова*' (dropdown), and 'Пробіг(км)*' (text input with '29850' and 'км' unit). A note at the bottom left of the form states '* поля обов'язкові до заповнення'. A 'Далі >' button is located at the bottom right of the form area. The footer includes social media links, a link to 'Допомога по сайту Ukraine.lad.com', and a 'Про проект Підтримка' link.

Рис. 3.18. Загальний вигляд сторінки додавання оголошення

Джерело: [Figma](#)

Під заголовком розташована візуальна індикація етапів заповнення форми, що складається з п'яти послідовних кроків: «Основна інформація», «Технічні характеристики», «Додаткова інформація», «Фотографії» та «Перегляд та публікація». Кожен крок представлений круглою іконкою з відповідним зображенням та текстовим описом під нею. Пройдений етап відмічений галочкою та візуально виділений, що допомагає користувачеві відстежувати прогрес заповнення форми. Така структура поетапного заповнення полегшує процес подання

оголошення, розбиваючи його на логічні блоки та зменшуючи когнітивне навантаження на користувача.

Перший крок «Основна інформація» присвячений введенню базових даних про автомобіль (див. Рис. 3.18.). Цей розділ супроводжується іконкою з зображенням автомобіля та описом «Введіть основну інформацію про Ваш автомобіль», що допомагає користувачеві зрозуміти, яку саме інформацію від нього очікують на цьому етапі. Форма містить такі поля: «Марка автомобіля», «Модель автомобіля», «Тип транспорту», «Тип кузова», «Рік випуску» та «Пробіг». Всі поля, крім текстових полів для введення року випуску та пробігу, представлені у вигляді випадних списків, що спрощує процес заповнення та запобігає помилкам при введенні даних. Обов'язкові поля позначені зірочкою, що допомагає користувачеві розуміти, які дані необхідно обов'язково вказати для успішного створення оголошення. Під полями для пробігу додано пояснювальний текст «* поля обов'язкові до заповнення», що додатково інформує користувача про необхідність заповнення всіх відмічених полів.

У нижній частині форми розташовані кнопки навігації: «Далі» для переходу до наступного кроку. Кнопка «Далі» виділена контрастним синім кольором та має стрілку, що підкреслює напрямок руху вперед у процесі заповнення форми. Таке оформлення кнопки робить її помітною та зрозумілою для користувача, спонукаючи до продовження процесу.

Другий крок «Технічні характеристики» призначений для введення більш детальної технічної інформації про автомобіль (див. Рис. 3.19.). Цей розділ відмічений іконкою із зображенням шестерні та супроводжується поясненням «Введіть технічні характеристики про Ваш автомобіль». Форма на цьому етапі містить поля: «Тип двигуна», «Об'єм двигуна», «Коробка передач» та «Привід». Як і на попередньому кроці, більшість полів представлені у вигляді випадних списків для зручності вибору серед стандартних варіантів. Поле «Об'єм двигуна» представлене у вигляді текстового поля з числовим форматом введення та позначенням одиниці

виміру «л» (літри). Обов'язкові поля також позначені зірочкою, а під формою повторюється нагадування про обов'язковість заповнення відповідних полів.

The screenshot displays the 'Kolesko' website interface for adding an advertisement. The main heading is 'Додайте Ваше оголошення' (Add your advertisement), with a subtext 'Заповніть форму щоб продати Ваш автомобіль швидко та вигідно' (Fill out the form to sell your car quickly and profitably). The process is divided into five steps: 'Основна інформація' (Basic information), 'Технічні характеристики' (Technical characteristics), 'Додаткова інформація' (Additional information), 'Фотографії' (Photos), and 'Перегляд та публікація' (Review and publication). The 'Технічні характеристики' step is currently active, showing a form with the following fields:

- Тип двигуна *** (Engine type *): A dropdown menu with the placeholder 'Оберіть тип двигуна...' (Select engine type...).
- Об'єм двигуна*** (Engine volume*): A text input field with the value '2.0'.
- Коробка передач*** (Gearbox*): A dropdown menu with the placeholder 'Оберіть тип коробки передач...' (Select gearbox type...).
- Привід*** (Drive*): A dropdown menu with the placeholder 'Оберіть привід...' (Select drive...).

Below the form, there is a note: '* поля обов'язкові до заповнення' (* fields are mandatory for completion). At the bottom of the form, there are two buttons: 'Назад' (Back) and 'Далі' (Next).

Рис. 3.19. Форма введення технічних характеристик

Джерело: [Figma](#)

У нижній частині форми на цьому етапі розташовані дві кнопки: «Назад» для повернення до попереднього кроку та «Далі» для продовження процесу. Кнопка «Назад» оформлена менш контрастно, ніж «Далі», що візуально виділяє рекомендований шлях, але залишає можливість повернутися та відредагувати введені раніше дані. Така навігація між кроками забезпечує гнучкість процесу заповнення форми, дозволяючи користувачеві вільно переміщатися між етапами.

Третій крок «Додаткова інформація» дозволяє користувачеві вказати додаткові деталі, що можуть бути важливими для потенційних покупців (див. Рис. 3.20). Цей розділ позначений іконкою документа та текстом «Введіть додаткову інформацію»

про Ваш автомобіль». Форма містить поля: «Колір», «Місто» та поле для текстового опису автомобіля. Поля для вибору кольору та міста представлені як випадні списки, що спрощує процес вибору серед стандартних варіантів. Текстове поле для опису автомобіля має збільшену висоту, що дозволяє вводити розгорнуту інформацію про стан автомобіля, його особливості та переваги. Це поле відіграє важливу роль у створенні привабливого оголошення, оскільки дозволяє продавцю виділити ключові переваги свого автомобіля та надати потенційним покупцям додаткову інформацію, яка може вплинути на їх рішення. Також на цьому етапі присутнє поле для введення ціни, що дозволяє продавцю вказати бажану суму продажу в доларах.

Рис. 3.20. Форма введення додаткової інформації

Джерело: [Figma](#)

Четвертий крок «Фотографії» присвячений завантаженню зображень автомобіля (див. Рис. 3.21.). Цей розділ позначений іконкою фотоапарата та текстом

«Завантажте фотографії Вашого автомобіля (мінімум 3 фото). Якісні фотографії збільшують шанси швидкого продажу». Це пояснення не лише інструктує користувача щодо необхідної кількості фотографій, але й мотивує його завантажити якісні зображення, підкреслюючи їх важливість для успішного продажу. На сторінці представлена область для завантаження фотографій у вигляді пунктирної рамки з іконкою завантаження та текстом «Перетягніть фото сюди або натисніть для вибору». Під цією областю розміщений текст з інформацією про підтримувані формати файлів: «Підтримуються формати JPEG, PNG (до 5 МБ кожне)». Також додана кнопка «Вибрати фото», що дозволяє користувачеві обрати файли для завантаження з пристрою. Така гнучкість у способах завантаження фотографій (перетягування або вибір через діалогове вікно) покращує досвід користування та адаптується до різних звичок користувачів.

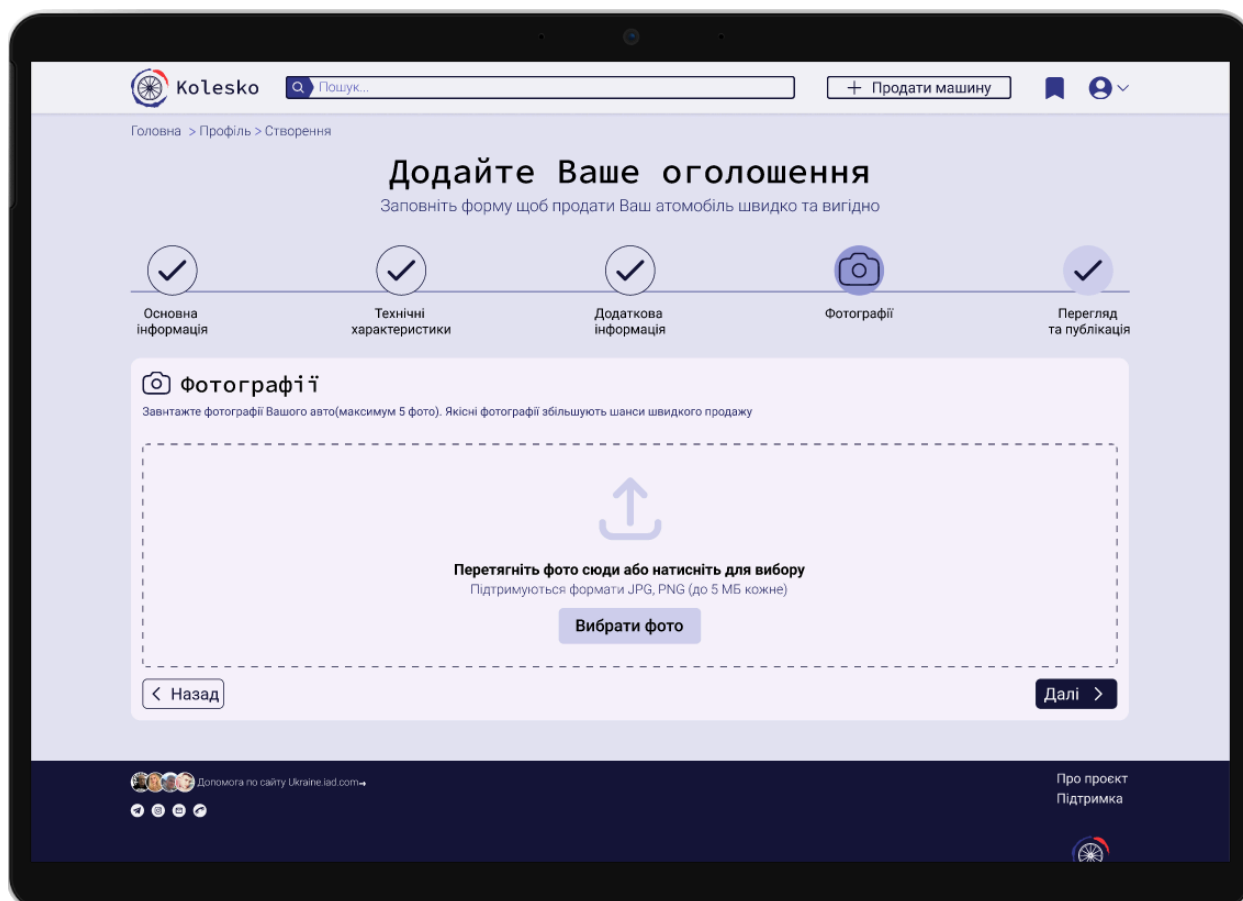


Рис. 3.21 Форма завантаження фотографій

Джерело: [Figma](#)

П'ятий і кінцевий крок «Перегляд та публікація» дозволяє користувачеві переглянути всю введену інформацію перед публікацією оголошення (див. Рис. 3.22.). Цей етап є критично важливим, оскільки дає можливість перевірити коректність всіх даних та внести необхідні зміни без необхідності повторного проходження всіх попередніх кроків. На сторінці представлена вся інформація, введена на попередніх етапах, структурована за тими ж розділами: «Основна інформація», «Технічні характеристики», «Додаткова інформація» та «Фотографії». Кожен розділ супроводжується кнопкою «Редагувати», що дозволяє користувачеві швидко перейти до відповідного етапу для внесення змін у разі виявлення помилок або необхідності доповнення інформації.

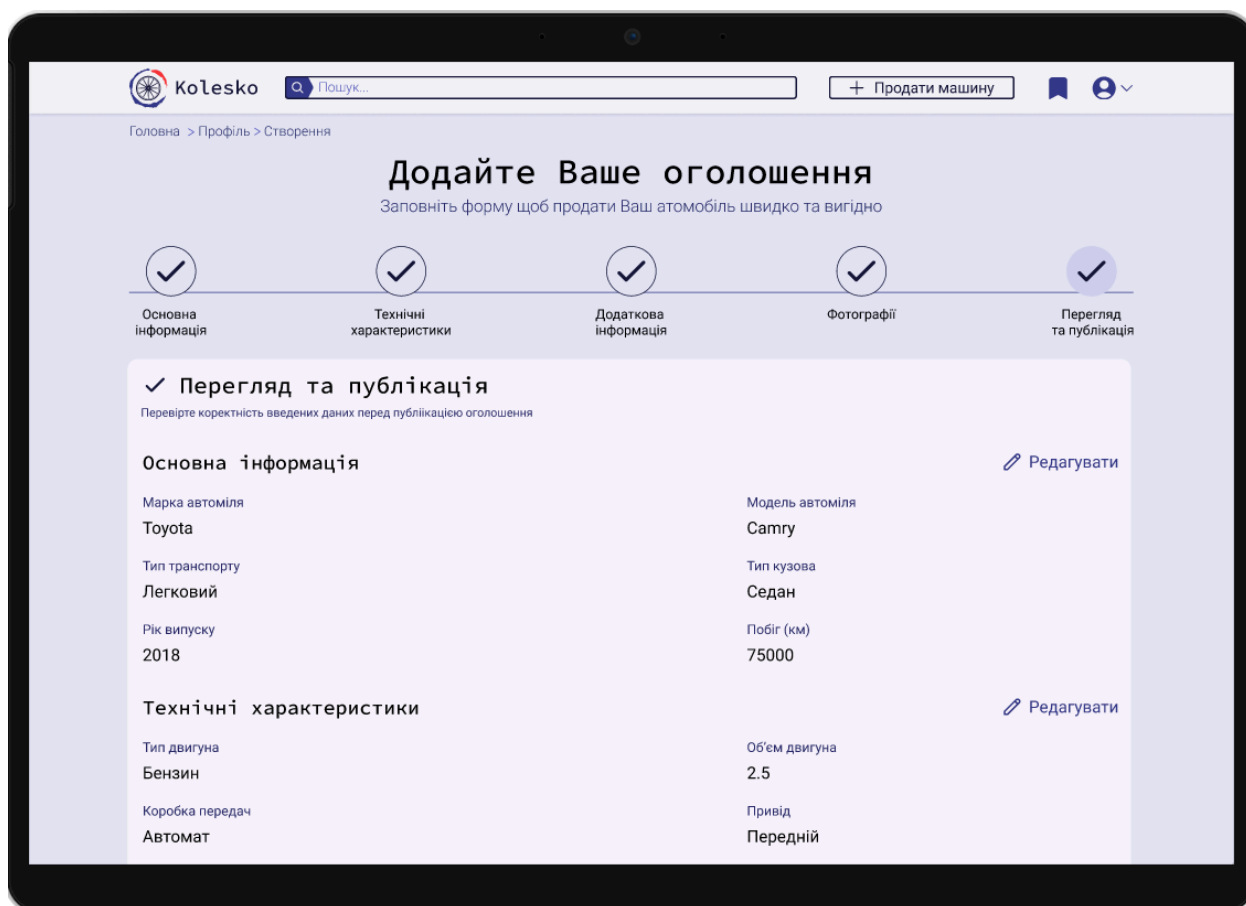


Рис. 3.22. Сторінка перегляду та публікації оголошення

Джерело: [Figma](#)

У нижній частині сторінки розташовані дві кнопки: «Назад» для повернення до попереднього кроку та «Опублікувати» для завершення процесу та публікації оголошення. Кнопка «Опублікувати» виділена контрастним синім кольором, що підкреслює її важливість як фінальної дії всього процесу. Після натискання на цю кнопку оголошення стає доступним для перегляду іншими користувачами платформи.

Дизайн сторінки додавання оголошення також реалізований з урахуванням адаптивності до різних пристроїв (див. Рис. 3.23. та Рис. 3.24.). На великих екранах, таких як десктопи та ноутбуки, форма відображається у повному обсязі з просторим розташуванням елементів. При переході на планшети змінюється ширина форми, а елементи перекомпонуються для оптимального використання доступного простору. На мобільних пристроях форма адаптується до вертикального відображення, елементи розташовуються в один стовпчик, а розмір шрифтів та інтерактивних елементів збільшується для зручності взаємодії на сенсорних екранах.

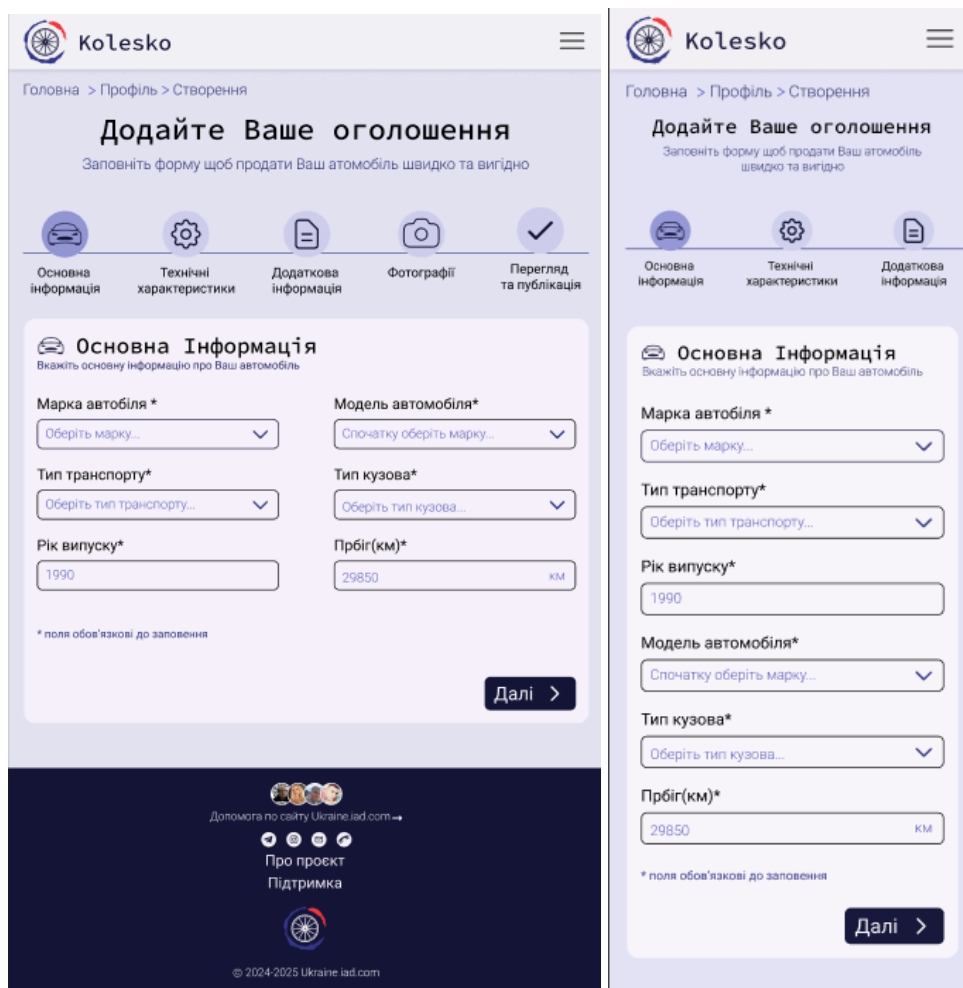


Рис. 3.23. Адаптивне відображення форми введення основної інформації на різних пристроях

Джерело: [Figma](#)

При адаптації для мобільних пристроїв особлива увага приділена зручності взаємодії з інтерактивними елементами. Кнопки мають збільшений розмір для зручності натискання пальцем, поля введення та випадні списки займають всю доступну ширину екрана, а візуальна індикація етапів може змінювати своє відображення для економії місця, зберігаючи при цьому інформативність. На мобільних пристроях з найменшими екранами (див. Рис. 3.23. та Рис. 3.24, праві зображення) відображення стає максимально компактним: зменшується відступ між елементами, елементи структуруються в один стовпчик, а навігаційні кнопки розташовуються таким чином, щоб забезпечити зручний доступ для користувача.

При цьому зберігається основний функціонал та інформативність форми, забезпечуючи повноцінний досвід користування незалежно від пристрою.

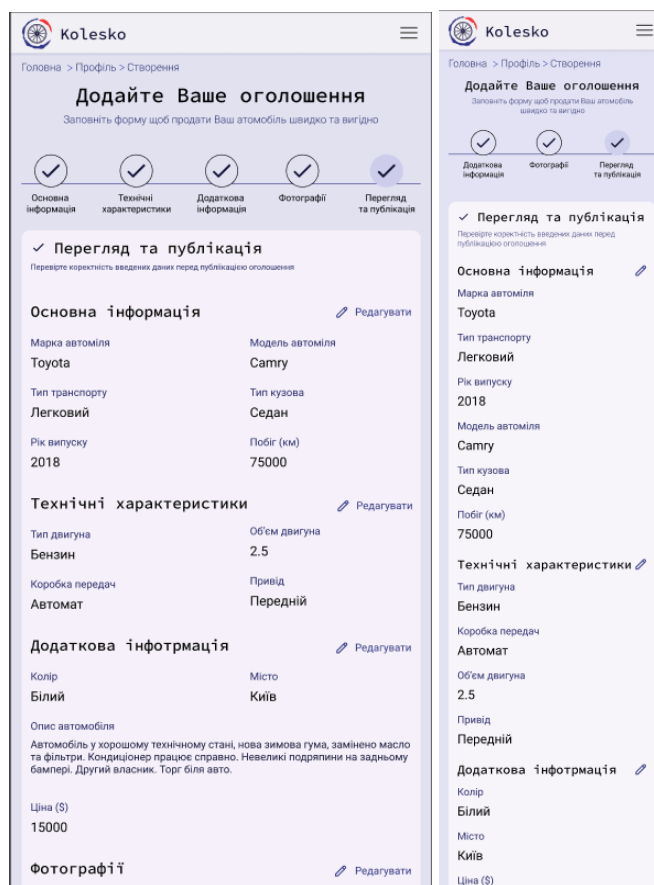


Рис. 3.24. Адаптивне відображення сторінки перегляду та публікації на різних пристроях

Джерело: [Figma](#)

У формі перегляду та публікації на мобільних пристроях (див. Рис. 3.24.) інформація також реорганізується в один вертикальний стовпчик, при цьому зберігаючи доступність кнопок редагування для кожного розділу. Фотографії автомобіля відображаються в адаптивній сітці, яка змінює кількість зображень у ряду залежно від ширини екрана: на широких екранах відображається по п'ять зображень у ряд, на планшетах – по три, а на мобільних пристроях – по два. Така адаптивність забезпечує оптимальний розмір зображень для перегляду на різних пристроях.

Дизайн сторінки додавання оголошення на платформі Kolesko вдало поєднує функціональність, зручність використання та візуальну привабливість. Завдяки логічній структурі процесу, зрозумілим інструкціям і візуальним підказкам користувачі можуть швидко створювати якісні оголошення про продаж автомобілів без зайвих труднощів. Адаптивність інтерфейсу дозволяє комфортно користуватися сервісом з будь-яких пристроїв, що робить процес додавання оголошень доступним для широкого кола продавців.

Дизайн інтерфейсу сторінки редагування оголошення враховує необхідність логічного структурування даних та зручної подачі форм для заповнення, що дозволяє користувачам швидко та без зайвих зусиль створити привабливе та інформативне оголошення (див. Рис. 3.25). У верхній частині сторінки знаходиться заголовок «Редагування оголошення», доповнений підзаголовком «Заповніть форму, щоб оновити дані про Ваш автомобіль». Він чітко пояснює мету сторінки та спонукає користувача до дії.

The screenshot shows a web interface for editing a car listing on 'Kolesko'. The page title is 'Редагування оголошення' (Editing announcement). Below the title is a subtitle: 'Заповніть форму щоб оновити дані про Ваш автомобіль' (Fill out the form to update data about your car).

The form is organized into three main sections, each with a title, an icon, and a subtitle:

- Основна Інформація** (Basic Information): Includes fields for 'Марка автомобіля *' (Car brand), 'Тип транспорту*' (Vehicle type), 'Рік випуску*' (Year of release), 'Модель автомобіля*' (Car model), 'Тип кузова*' (Body type), and 'Пробіг(км)*' (Mileage).
- Технічні характеристики** (Technical Characteristics): Includes fields for 'Тип двигуна *' (Engine type), 'Коробка передач*' (Transmission), 'Об'єм двигуна*' (Engine volume), and 'Привід*' (Drive).
- Додаткова інформація** (Additional Information): This section is currently empty.

The top navigation bar includes the 'Kolesko' logo, a search bar, and a '+ Продати машину' (Sell car) button. The breadcrumb trail shows 'Головна > Профіль > Редагування'.

Рис. 3.25. Загальний вигляд сторінки редагування оголошення

Джерело: [Figma](#)

Весь контент сторінки логічно розділений на чотири основні блоки: «Основна Інформація», «Технічні характеристики», «Додаткова інформація» та «Фотографії». Кожен з цих блоків містить відповідний набір полів для заповнення, що дозволяє користувачу поетапно надати всю необхідну інформацію про свій автомобіль. Такий структурований підхід до організації форми значно спрощує процес заповнення даних та дозволяє користувачу легко орієнтуватися на сторінці. Кожен блок візуально відокремлений від інших та має чіткий заголовок із відповідною іконкою, що підкреслює його тематичну спрямованість.

У нижній частині сторінки розміщені дві кнопки навігації: «Назад» та «Зберегти зміни». Кнопка «Назад» дозволяє користувачу повернутися на попередню сторінку без збереження змін, а кнопка «Зберегти зміни» зберігає всі внесені дані та публікує оголошення. Варто відзначити, що кнопка «Зберегти зміни» візуально

виділена темним кольором та білим текстом, що притягує увагу користувача та підкреслює її важливість.

Для забезпечення зручності використання на різних пристроях сторінка редагування оголошення має адаптивний дизайн (див. Рис. 3.26.). При перегляді на пристроях з меншими екранами, таких як планшети та смартфони, відбувається перебудова інтерфейсу для оптимального використання доступного простору. На мобільних пристроях всі блоки та поля форми переформатовуються в один вертикальний стовпець, що дозволяє користувачу послідовно заповнювати інформацію без необхідності горизонтальної прокрутки. Поля, які на широких екранах розташовані поруч (наприклад, марка та модель автомобіля), на мобільних пристроях розміщуються один під одним, зберігаючи при цьому свою функціональність та зручність використання.

The image displays two versions of the 'Kolesko' website's 'Edit Ad' page. The left version is for desktop, showing a two-column layout. The right version is for mobile, showing a single-column layout. Both versions include the following sections:

- Основна Інформація** (Basic Information):
 - Марка автомобіля* (Car Brand): Daewoo
 - Модель автомобіля* (Car Model): Lanos
 - Тип транспорту* (Vehicle Type): Легковий (Passenger)
 - Тип кузова* (Body Type): Седан (Sedan)
 - Рік випуску* (Year of Release): 2008
 - Пробіг(км)* (Mileage): 175000 км
- Технічні характеристики** (Technical Characteristics):
 - Тип двигуна* (Engine Type): Бензин (Gasoline)
 - Об'єм двигуна* (Engine Volume): 1.5
 - Коробка передач* (Gearbox): Механіка (Manual)
 - Привід* (Drive): Передній (Front)
- Додаткова інформація** (Additional Information):
 - Колір* (Color): Чорний (Black)
 - Місто* (City): Львів (Lviv)
 - Опис автомобіля* (Car Description): Автомобіль у хорошому технічному стані, нова змазка гуми, замінено масло та фільтри. Кондиціонер працює справно. Невеличкі подряпини на задньому бампері. Другий власник. Торг біля авто.
 - Ціна* (Price): 3200 \$
- Фотографії** (Photos): Section for adding car photos.

Рис. 3.26. Адаптивний дизайн сторінки редагування оголошення

Джерело: [Figma](#)

Для мобільної версії також оптимізовано процес завантаження фотографій. Мініатюри зображень зменшуються в розмірі та розміщуються в один або два стовпці, залежно від розміру екрана. Це дозволяє користувачам легко переглядати та керувати завантаженими зображеннями навіть на невеликих екранах. Кнопки навігації «Назад» та «Зберегти зміни» в мобільній версії розтягуються на всю ширину екрана, що робить їх більш доступними для натискання пальцем.

Особливу увагу в дизайні сторінки редагування оголошення приділено валідації введених даних. Обов'язкові поля позначені зірочкою (*), а система перевіряє коректність введених значень та інформує користувача про помилки в режимі реального часу. Це дозволяє уникнути ситуацій, коли користувач заповнює всю форму, але не може зберегти її через помилки в одному з полів. Такий підхід до валідації даних забезпечує високу якість інформації в оголошеннях та спрощує процес їх створення.

Загалом, дизайн сторінки редагування оголошення на платформі Kolesko являє собою добре продуману та структуровану систему, яка дозволяє користувачам швидко та без зайвих зусиль створювати якісні оголошення про продаж своїх автомобілів. Використання сучасних принципів UI/UX дизайну, таких як логічна структура, інтуїтивно зрозумілі елементи керування та адаптивність до різних пристроїв, забезпечує високу ефективність взаємодії користувача з платформою та підвищує задоволеність від користування сервісом.

Висновок до розділу

Цей розділ охоплює розробку ключових сторінок маркетплейсу вживаних авто. На початковому етапі проаналізовано референси та створено мудборд для визначення візуальних і функціональних рішень.

Далі спроектовано сторінку каталогу з фільтрацією, різними варіантами відображення оголошень та пагінацією, приділяючи увагу адаптивності. Також

розроблено дизайн сторінки деталей оголошення, акцентуючи на зручному поданні інформації та візуальній складовій, з урахуванням адаптивності.

Окремо створено проєкт сторінки профілю користувача для ефективного керування даними та оголошеннями, з інтуїтивним інтерфейсом та адаптивністю. Завершено розділ розробкою сторінок додавання та оновлення оголошень, використовуючи покрокову форму та адаптивний дизайн для зручності користувачів.

Таким чином, цей розділ детально описує процес створення основних сторінок маркетплейсу, починаючи з візуальної концепції та закінчуючи проєктуванням інтерфейсів з фокусом на зручності та адаптивності для різних пристроїв. Ці розробки є фундаментом для подальшого розвитку платформи.

РОЗДІЛ 4

ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОЄКТУ

4.1. Технології та фреймворки для реалізації клієнтської частини

Фреймворк — це програмна платформа, що надає набір інструментів, структур і готових рішень для ефективної розробки програмного забезпечення. Використання фреймворків дозволяє дотримуватись єдиної архітектури, пришвидшує процес створення додатків і полегшує підтримку коду. Особливо важливу роль фреймворки відіграють у фронтенд-розробці — створенні інтерфейсу користувача, який безпосередньо впливає на взаємодію з системою[15].

Для реалізації клієнтської частини програмного продукту у межах дипломного проєкту було обрано Angular 18 — один з найпотужніших та найбільш розвинених фреймворків сучасної веброзробки. Angular є open-source рішенням, яке підтримується компанією Google, що забезпечує його стабільний розвиток, регулярні оновлення та відповідність актуальним стандартам безпеки й продуктивності. Вибір Angular 18 зумовлений кількома ключовими чинниками:

- Сучасні технології та продуктивність. Angular 18 включає покращення у продуктивності та стабільності, зокрема завдяки поступовому впровадженню Signals API — реактивного підходу до управління станом. Також удосконалено процес tree-shaking, що дозволяє зменшити розмір фінального бандлу та пришвидшити завантаження сторінок.
- Інтеграція з TypeScript. Angular спочатку проєктувався з урахуванням роботи з мовою TypeScript, що забезпечує типобезпечний код, полегшує масштабування проєкту та дозволяє виявляти помилки ще на етапі компіляції.
- Повна екосистема та інструментарій. Angular містить усе необхідне для фронтенд-розробки: систему маршрутизації, форми, HTTP-клієнт, анімації, підтримку PWA, unit-тестування, інтеграцію з RxJS, i18n тощо. Це зменшує залежність від сторонніх бібліотек і спрощує підтримку коду.

- Суворі структура та архітектура. Angular передбачає чітку організацію проєкту на основі модулів і компонентів, що полегшує командну розробку, підтримку та масштабування великих застосунків.
- Підтримка сучасного UI. Angular легко інтегрується з бібліотеками інтерфейсів, такими як Angular Material чи PrimeNG, що дозволяє швидко створювати адаптивні й доступні інтерфейси згідно з рекомендаціями UX-дизайну.
- Активна спільнота та документація. Angular має широку базу користувачів, регулярні оновлення та багато навчальних матеріалів, що значно полегшує навчання та розв'язання проблем під час розробки.

Порівнюючи Angular 18 з іншими популярними інструментами для фронтенду, такими як React, Vue.js чи Ember.js, можна виділити кілька важливих особливостей:

- React (розроблений Meta) — це бібліотека для побудови UI, яка вирізняється гнучкістю та мінімалізмом. Вона потребує підключення сторонніх бібліотек для реалізації повного функціоналу (React Router, Redux, Axios тощо). Angular, на відміну від React, — це повноцінний фреймворк з вбудованими засобами, що дозволяє створювати масштабовані додатки без додаткових залежностей.
- Vue.js — легкий і зручний фреймворк, популярний серед початківців і для створення прототипів. Проте для великих корпоративних застосунків йому бракує суворі архітектури та типізації, яку забезпечує Angular через інтеграцію з TypeScript.
- Ember.js — потужний фреймворк з чіткою структурою і вбудованими інструментами, орієнтований на великі команди. Однак він менш популярний, має повільніший розвиток і меншу спільноту порівняно з Angular. Angular при цьому зберігає баланс між гнучкістю та суворістю, активно розвивається, підтримується Google та має сучасні інструменти, зокрема Signals API і розвинені DevTools.

Отже, кожен із розглянутих фреймворків має свої переваги, однак Angular 18 залишається одним із найкращих рішень для створення масштабованих, продуктивних і підтримуваних вебзастосунків із чітко структурованою архітектурою та сучасними засобами розробки.

4.2. Реалізація сторінки профілю користувача з використанням Angular

Для реалізації безпечного доступу користувачів до функціоналу вебзастосунку було впроваджено процес авторизації із використанням сервісу Keycloak. Основна логіка інтеграції авторизації реалізована в компоненті заголовка «HeaderComponent» та відповідному сервісі «KeycloakService». У файлі `Header.component.html` (див. Лістинг 4.2.1) передбачено випадаюче меню користувача з опціями «Увійти» та «Вийти». Натискання на кнопку «Увійти» запускає метод `login()`, який спрямовує користувача до сторінки авторизації Keycloak. Аналогічно, вибір пункту «Вийти» ініціює завершення сесії.

Лістинг 4.2.1. файл *Header.component.html*

```
<div #userMenu class="relative">
  <div class="flex flex-row items-center">
    <a [routerLink]="['/profile']">
      <kolesko-icon class="text-portGore-700 cursor-pointer" name="profile-main"
[size]="36">
        </kolesko-icon></a>

      <kolesko-icon class="text-portGore-700 cursor-pointer" name="arrow-down-main"
[width]="22" [height]="20"
        (click)="toggleUserMenu($event)">
      </kolesko-icon>
    </div>

    <div *ngIf="showUserMenu"
      class="absolute right-0 z-50 mt-3 w-28 min-w-[112px] rounded-lg shadow-xl
bg-white border border-snuff-200">
      <div class="py-2">
        <div class="flex flex-col space-y-1">
          <a (click)="login()"
            class="flex items-center justify-center w-full h-[36px] text-base
font-medium text-black hover:bg-snuff-100 hover:text-portGore-600 transition-colors
duration-200 rounded-md cursor-pointer">
            Увійти
          </a>
```



```

        <div class="border-t border-gray-200 my-1"></div>

        <a (click)="logout()"
            class="flex items-center justify-center w-full h-[36px] text-base
font-medium text-red-600 hover:bg-red-50 hover:text-red-700 transition-colors
duration-200 rounded-md cursor-pointer">
            Вийти
        </a>
    </div>
</div>
</div>
</div>

```

Методи `login()` і `logout()`, які виконують відповідні функції входу та виходу користувача, реалізовані у файлі `Header.component.ts` (див. Лістинг 4.2.2). Ці методи діють як посередники, передаючи виклики безпосередньо до сервісу `AuthService`. Така структура забезпечує централізоване керування логікою автентифікації, дозволяючи спростити подальше обслуговування та розширення функціональності.

Лістинг 4.2.2. файл *Header.component.ts*

```

login(): void {
    this.authService.login();
}

logout(): void {
    this.authService.logout();
}

```

У файлі `Keycloak.service.ts` (див. Лістинг 4.2.3), реалізована комплексна логіка, що забезпечує правильне формування URL для авторизації користувача в системі. Зокрема, метод `login()` відіграє ключову роль, оскільки формує запит до сервера Keycloak, включаючи в нього важливі параметри клієнта. Крім того, вказується параметр редіректу, який визначає URL для повернення після успішної авторизації. Оскільки відповідь має надійти у вигляді коду авторизації (де `response_type` встановлено як «code»), це дозволяє безпечно обробити подальші дії. Додатково, використовується параметр «state», що критично важливо для захисту додатка від атак CSRF (Cross-Site Request Forgery), тим самим забезпечуючи захист і збереження даних користувачів.

Лістинг 4.2.3. файл *Keycloak.service.ts*

```

public login(): void {
  const state = this.generateRandomState();
  sessionStorage.setItem('kc_state', state);

  const currentUrl = this.router.url;
  if (currentUrl !== '/auth-callback') {
    localStorage.setItem('returnUrl', currentUrl);
  }

  const authUrl =
`${this.keycloakUrl}/realms/${this.realm}/protocol/openid-connect/auth`;
  const params = new URLSearchParams({
    client_id: this.clientId,
    redirect_uri: this.redirectUri,
    response_type: 'code',
    scope: 'openid email profile',
    state
  });

  window.location.href = `${authUrl}?${params.toString()}`;
}

```

В результаті натискання на кнопку входу користувач автоматично перенаправляється на сторінку авторизації, що динамічно генерується системою Keycloak. У файлі *BannerComponent.component.ts* (див. Лістинг 4.2.4), реалізовано логіку взаємодії з API для завантаження та оновлення даних користувача. Після успішної автентифікації, функція *loadUserProfile()* виконує HTTP-запит до захищеного маршруту <https://localhost:5001/api/identity/me>, передаючи access token у заголовках запиту. У разі успішної відповіді профіль користувача зберігається у властивості *userProfile*. Кнопка відкриття модального вікна редагування профілю викликає метод *openProfileModal()*, який ініціалізує форму з уже наявними даними користувача, якщо такі доступні. Форма дозволяє редагувати ім'я, прізвище та номер телефону. Після внесення змін функція *onSubmit()* формує об'єкт *profileData*, що відправляється PUT-запитом на адресу <https://localhost:5001/api/identity>. У разі успішного оновлення, повторно викликається *loadUserProfile()* для оновлення відображених даних, а модальне вікно закривається методом *closeProfileModal()*.

ЛІСТИНГ 4.2.4. файл *BannerComponent.component.ts*

```
loadUserProfile() {
  const token = this.keycloakService.getAccessToken();

  if (!token) {
    console.error('No access token found');
    return;
  }

  const headers = new HttpHeaders({
    Authorization: `Bearer ${token}`
  });

  this.http.get('https://localhost:5001/api/identity/me', { headers })
    .subscribe({
      next: (profile) => {
        this.userProfile = profile;
      },
      error: (err) => {
        console.error('Failed to load user profile', err);
      }
    });
}

openProfileModal() {
  if (this.userProfile) {
    this.profileForm = {
      firstName: this.userProfile.firstName || '',
      lastName: this.userProfile.lastName || '',
      phone: this.userProfile.phone || '',
      profilePhoto: null
    };
  }
  this.isProfileModalOpen = true;
}

closeProfileModal() {
  this.isProfileModalOpen = false;
}

onSubmit() {
  const token = this.keycloakService.getAccessToken();
  if (!token) {
    console.error('No access token found');
    return;
  }

  const headers = new HttpHeaders({
    Authorization: `Bearer ${token}`,
    'Content-Type': 'application/json'
  });
}
```

```

const profileData = {
  phoneNumber: this.profileForm.phone,
  firstName: this.profileForm.firstName,
  lastName: this.profileForm.lastName
};

const apiUrl = `https://localhost:5001/api/identity`;

this.http.put(apiUrl, profileData, { headers })
  .subscribe({
    next: (response) => {
      console.log('Profile updated successfully', response);
      this.loadUserProfile();
      this.closeProfileModal();
    },
    error: (err) => {
      console.error('Failed to update profile', err);
    }
  });
}

```

Процес завантаження профілю реалізується за допомогою HTTP-запиту, що відправляється до захищеного API. У цьому запиті передається токен доступу через відповідний заголовок для забезпечення безпеки даних. Завдяки такому підходу, після успішної авторизації користувач отримує автоматизований доступ до свого персонального набору даних, які можуть бути інтегровані та використані у різних елементах інтерфейсу. Це дозволяє забезпечити зручність і ефективність обробки інформації, необхідної для роботи з додатком.

Для відображення персоналізованої інформації користувача було створено компонент банера профілю `BannerComponent`. У файлі шаблону `BannerComponent.component.html` (див. Лістинг 4.2.5) реалізовано структуроване відображення аватара, імені, електронної пошти та номера телефону користувача. Основна частина інтерфейсу представлена у вигляді адаптивного контейнера з декоративним SVG-аватаром та блоками тексту, що отримують дані з об'єкта `userProfile`. Додатково передбачено кнопку редагування профілю з іконкою. Усі елементи стилізовані за допомогою класів Tailwind CSS для досягнення сучасного зовнішнього вигляду та адаптивності.

Лістинг 4.2.5. файл *BannerComponent.component.html*

```

<div class="flex-1">
  <div class="flex items-center justify-between mb-md">
    <h1 class="text-h3 font-medium">{{userProfile.firstName}}</h1>
    <button class="text-indigo-700 hover:text-indigo-900
transition-colors"
      (click)="openProfileModal()">
      <svg ... </svg>
    </button>
  </div>
  <div class="space-y-sm">
    <div class="flex items-center gap-sm text-lg text-indigo-900">
      <svg xmlns="http://www.w3.org/2000/svg" width="24" height="24"
viewBox="0 0 20 20" fill="none">
        <path fill-rule="evenodd" clip-rule="evenodd"
          d="M2.40029 4...5Z"
          fill="#353888" />
      </svg>
      <span>{{userProfile.email}}</span>
    </div>
    <div class="flex items-center gap-sm text-lg text-indigo-900">
      <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20"
viewBox="0 0 20 20" fill="none">
        <path
          d="M...Z"
          stroke="#353888" stroke-width="1.5"
stroke-linecap="round" stroke-linejoin="round" />
      </svg>
      <span *ngIf="userProfile.phoneNumber; else addPhone">{{
userProfile.phoneNumber }}</span>
      <ng-template #addPhone>
        <button class="text-red-600 underline"
(click)="openProfileModal()">
          Додайте номер
        </button>
      </ng-template>
    </div>
  </div>
</div>
</div>

```

Нижче показано реалізований інтерфейс компонента банера профілю користувача в додатку Kolesko(див. Рис. 4.1.). Відображено круглий аватар з іконкою користувача зліва, праворуч від якого розташована персональна інформація: ім'я, email та телефон з відповідними іконками. У правому верхньому куті банера

розміщена кнопка редагування з іконкою олівця. Поруч знаходиться блок статистики з показниками активних оголошень та переглядів.

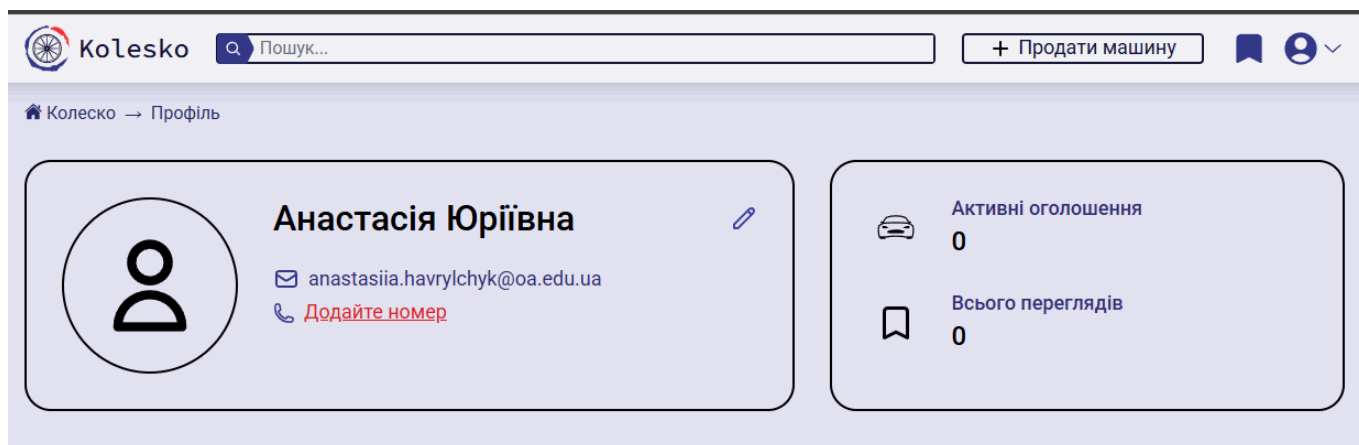


Рис. 4.1. Вигляд банеру сторінки профілю користувача

Джерело: розроблено автором

Коли користувач реєструється за допомогою Google, ми не завжди можемо отримати його номер телефону. Якщо у нас немає номера користувача, ми інформуємо його про це та просимо додати його. Як наслідок, після успішної реєстрації або авторизації, коли система виявляє відсутність номера телефону, автоматично відкривається форма для редагування профілю (див. Рис. 4.2.). У цій формі користувач може легко додати або оновити свій номер телефону (поле "Телефон"), а також перевірити та за потреби відредагувати своє ім'я та прізвище. Це забезпечує зручний спосіб для користувача доповнити свою інформацію та надає нам необхідні дані для подальшої комунікації та надання послуг. Після введення номера телефону користувач натискає "Зберегти" для оновлення профілю.

Редагувати профіль

Фото профілю

Змінити фото

Ім'я

Анастасія Юрїївна

Прізвище

Гаврильчик

Телефон

0680340111

Скасувати Зберегти

Рис. 4.2. Вигляд форми для редагування інформації про користувача

Джерело: розроблено автором

4.3. Розробка компонентів для фільтрації та пошуку авто

Пошук транспортних засобів є ключовою функцією системи, яка забезпечує користувачам зручний доступ до необхідної інформації за заданими критеріями. Його реалізація включає використання структурованих даних, ефективних алгоритмів фільтрації та сортування, а також механізмів оптимізації взаємодії з бекендом. Застосування асинхронних запитів дозволяє покращити швидкість обробки даних, а використання інтерфейсів TypeScript забезпечує коректність переданих параметрів.

Одним із ключових елементів реалізації пошуку є інтерфейс `SearchVehicle.ts` (див. Лістинг 4.3.1). Він визначає структуру об'єкта автомобіля, що повертається у результатах запиту. Цей інтерфейс включає основні характеристики авто, які допомагають швидко оцінити знайдені результати. Використання типізованого

інтерфейсу забезпечує консистентність даних у всьому додатку та сприяє кращій інтеграції між різними компонентами системи. Завдяки цьому інтерфейсу система може правильно обробляти та відображати інформацію про автомобілі, гарантуючи, що всі необхідні поля присутні у відповіді API.

Лістинг 4.3.1. файл *SearchVehicle.ts*

```
export interface SearchVehicle {  
  id: string;  
  mainImageUrl: string;  
  fullName: string;  
  price: number;  
}
```

Для обробки параметрів пошуку використовується *SearchVehicleParams.ts* (див. Лістинг 4.3.2). Цей файл визначає структуру запиту, який надсилається на сервер. Він містить ключові параметри, що дозволяють налаштувати результати відповідно до потреб користувача. Ця структура є критично важливою для організації ефективної пагінації та фільтрації результатів. Інтерфейс *SearchVehicleParams* встановлює чіткий контракт між клієнтом і сервером, визначаючи формат даних, які очікує бекенд. Такий підхід спрощує підтримку коду і мінімізує ризик помилок при зміні параметрів пошуку.

Лістинг 4.3.2. файл *SearchVehicleParams.ts*

```
export interface SearchVehicleParams {  
  SearchTerm: string;  
  ItemQuantity: number;  
  pageIndex: number;  
}
```

Важливим компонентом у структурі пошуку є *VehicleSearchService.ts* (див. Лістинг 4.3.3). Цей сервіс виконує HTTP-запити до бекенду, обробляє отримані дані та надає їх у вигляді *Observable<SearchResponse>*. Завдяки цьому компоненту відбувається взаємодія між клієнтською і серверною частиною. Сервіс реалізує шаблон проєктування «Репозиторій», який абстрагує доступ до даних від решти

програми. Він відповідає за формування правильного URL та параметрів запиту, обробку помилок та перетворення HTTP-відповіді у зручний для роботи формат. Використання оператора `catchError` дозволяє обробляти мережеві проблеми та помилки сервера, повертаючи порожні результати замість аварійного завершення програми.

Лістинг 4.3.3. файл *VehicleSearchService.ts*

```
export class VehicleSearchService {
  private apiUrl: string;

  constructor(private http: HttpClient) {
    const baseUrl = environment.apiUrl.endsWith('/')
      ? environment.apiUrl.slice(0, -1)
      : environment.apiUrl;

    this.apiUrl = `${baseUrl}/vehicles/find`;
  }

  searchVehicles(searchTerm: string, pageIndex: number = 1, itemQuantity: number = 10):
  Observable<SearchResponse> {
    if (!this.apiUrl) {
      console.error('API URL is not configured');
      return of({ items: [], currentPageItemsQuantity: 0, pageIndex: 1 });
    }

    let httpParams = new HttpParams()
      .set('SearchTerm', searchTerm)
      .set('pageIndex', pageIndex.toString())
      .set('ItemQuantity', itemQuantity.toString());

    return this.http.get<SearchResponse>(this.apiUrl, { params: httpParams })
      .pipe(
        tap(response => console.log('Raw API response:', response)),
        catchError(error => {
          console.error('Error fetching search results:', error);
          if (error.status === 404) {
            console.warn('API endpoint not found. Check your API URL configuration.');
          } else if (error.status === 0) {
            console.warn('Network error or CORS issue. Check if API is running.');
          }
          return of({ items: [], currentPageItemsQuantity: 0, pageIndex: 1 });
        })
      );
  }
}
```

Головний компонент, який відповідає за введення пошукового запиту, обробку введених даних та оновлення результатів, знаходиться у `HeaderComponent.ts` (див.

Лістинг 4.3.4). Він стежить за змінами в полі пошуку та керує відображенням результатів. Компонент використовує RxJS оператори для оптимізації кількості запитів до сервера та забезпечення плавної роботи інтерфейсу. Використання операторів `debounceTime` та `distinctUntilChanged` запобігає надмірним запитам до сервера під час швидкого введення тексту, а `switchMap` скасовує попередні запити, якщо введено новий текст. Цей компонент також відповідає за розділення всіх результатів на ті, що відображаються одразу, та ті, що можна побачити після натискання кнопки «Переглянути ще».

Лістинг 4.3.4. файл *HeaderComponent.ts*

```

constructor(
  private router: Router,
  private searchService: VehicleSearchService
) {
  this.searchControl.valueChanges.pipe(
    debounceTime(300),
    distinctUntilChanged(),
    filter((term: string | null) => !!term && term.length >= 2),
    tap(() => {
      this.isLoading = true;
      this.showResults = true;
      this.showingAll = false;
    }),
    switchMap((term: string | null): Observable<SearchResponse> => {
      if (term) {
        return this.searchService.searchVehicles(term, 1, 10);
      }
      return of({ items: [], currentPageItemsQuantity: 0, pageIndex: 1 });
    })
  ).subscribe({
    next: (response: SearchResponse) => {
      this.allSearchResults = response.items || [];
      this.searchResults = this.allSearchResults.slice(0, 3);
      this.isLoading = false;
    },
    error: (error) => {
      console.error('Search error:', error);
      this.isLoading = false;
      this.searchResults = [];
      this.allSearchResults = [];
    }
  });
}

```

Для відображення знайдених автомобілів використовується шаблон `header.component.html` (див. Лістинг 4.3.5). Він містить логіку візуалізації результатів, включаючи індикатор завантаження та можливість розгорнути список

знайдених машин. Шаблон забезпечує адаптивний дизайн з інтерактивними елементами для покращення користувацького досвіду. HTML-структура використовує умовні директиви Angular (*ngIf) для відображення різних станів інтерфейсу: завантаження, відсутність результатів або список знайдених авто. Для відображення списку автомобілів використовується цикл *ngFor.

Лістинг 4.3.5. файл *Header.component.html*

```

<div *ngIf="showResults"
  class="absolute top-full left-0 right-0 z-50 mt-1 bg-white border border-gray-200
rounded-md shadow-lg max-h-96 overflow-y-auto"
  style="width: 100%;">
  <div *ngIf="isLoading" class="flex justify-center p-3">
    <div class="animate-spin rounded-full h-5 w-5 border-t-2 border-b-2
border-portGore-700"></div>
  </div>

  <div *ngIf="!isLoading && searchResults.length === 0" class="p-3 text-center
text-gray-600">
    Немає результатів за вашим запитом
  </div>

  <div *ngIf="!isLoading && searchResults.length > 0" style="padding: 4px;">
    <div *ngFor="let vehicle of searchResults"
      class="flex items-center hover:bg-gray-50 cursor-pointer transition-colors
duration-150 border border-gray-100 rounded"
      style="margin: 4px; padding: 4px;" (click)="navigateToVehicle(vehicle.id)">

      <div class="flex-shrink-0" style="padding: 4px;">
        <img [src]="vehicle.mainImageUrl" [alt]="vehicle.fullName"
          class="w-[110px] h-[70px] object-cover rounded">
        </div>

        <div class="flex-grow ml-3">
          <span class="font-medium text-portGore-900 block">{{ vehicle.fullName
}}</span>
          <span class="text-portGore-700">{{ vehicle.price |
currency: 'USD': 'symbol': '1.0-0' }}</span>
        </div>
      </div>

      <div *ngIf="!showingAll && allSearchResults.length > 3"
        class="text-center py-2 cursor-pointer text-portGore-700
hover:text-portGore-900 border-t border-gray-100"
        (click)="loadMoreResults()">
        <span>Переглянути ще</span>
      </div>
    </div>
  </div>
</div>

```

При введенні в пошуковий рядок хоча б двох символів компонент ініціює запит до сервісу `VehicleSearchService`, який звертається до бекенду для отримання даних. Відповідь API обробляється у `HeaderComponent.ts` та відображається у випадному списку під полем пошуку. Користувач бачить перші три знайдені автомобілі. Якщо в базі даних є більше варіантів, з'являється кнопка «Переглянути ще», яка показує повний список. Якщо користувач натискає на конкретний автомобіль у списку, викликається метод `navigateToVehicle(vehicle.id)`, що перенаправляє його на сторінку з детальною інформацією про обрану машину.

Функціонал реалізований таким чином, щоб мінімізувати навантаження на сервер завдяки використанню `debounceTime` і `distinctUntilChanged`, а також забезпечити зручність користування завдяки інтерактивному відображенню результатів у реальному часі. Цей підхід дозволяє досягти балансу між швидкістю відгуку інтерфейсу та ефективним використанням ресурсів серверної частини, гарантуючи при цьому позитивний досвід користувача під час пошуку автомобілів на платформі.

Перейдемо до візуального результату пошуку. У верхній частині сторінки розташований пошуковий рядок на темному фоні, що забезпечує високу контрастність і помітність цього ключового елемента взаємодії. При введенні пошукового запиту «седан» система миттєво реагує, розгортаючи під полем пошуку випадний список з першими трьома результатами (див. Рис. 4.3.). Кожен автомобіль у цьому списку представлений у вигляді компактної інформаційної картки, що поєднує візуальний елемент та текстовий блок з назвою моделі та ціною. Такий спосіб подання інформації дозволяє користувачеві швидко ідентифікувати бажаний автомобіль без необхідності переходу на окрему сторінку результатів. Розташування зображення ліворуч, а текстової інформації праворуч відповідає природному шляху сканування інформації більшістю користувачів. Під списком результатів знаходиться інтерактивний елемент «Переглянути ще», що візуально відділений тонкою верхньою межею і має спеціальне форматування тексту, що підказує користувачеві про його клікабельність.

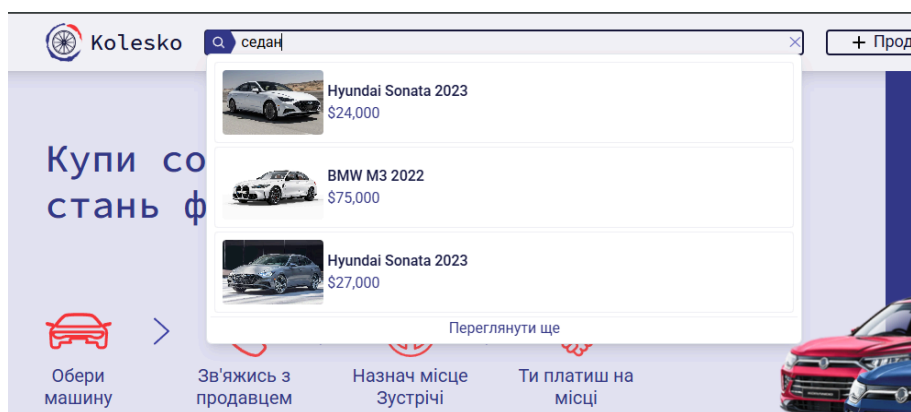


Рис. 4.3. Результат виконання пошуку

Джерело: розроблено автором

При натисканні на елемент «Переглянути ще» інтерфейс плавно розширюється, показуючи повний список знайдених автомобілів, що відповідають запиту «седан» (див. Рис. 4.4.). Тепер користувач бачить додаткові моделі, кожна з яких представлена у тому ж форматі, що забезпечує візуальну консистентність інтерфейсу. Важливо відзначити, що розширення списку відбувається без перезавантаження сторінки чи додаткових запитів до сервера, оскільки всі дані вже завантажені в пам'ять браузера під час першого пошуку. Це створює ефект миттєвої реакції інтерфейсу, що значно покращує користувацький досвід. Кожна картка автомобіля має ефект підсвічування при наведенні курсора, що візуально сигналізує про можливість взаємодії та заохочує користувача до подальшого дослідження конкретної пропозиції. Такий дизайн пошукового інтерфейсу дозволяє користувачеві спочатку швидко оцінити найбільш релевантні пропозиції, а потім, за бажанням, розширити перегляд до повного списку знайдених автомобілів, що ідеально балансує між інформативністю та простотою використання.

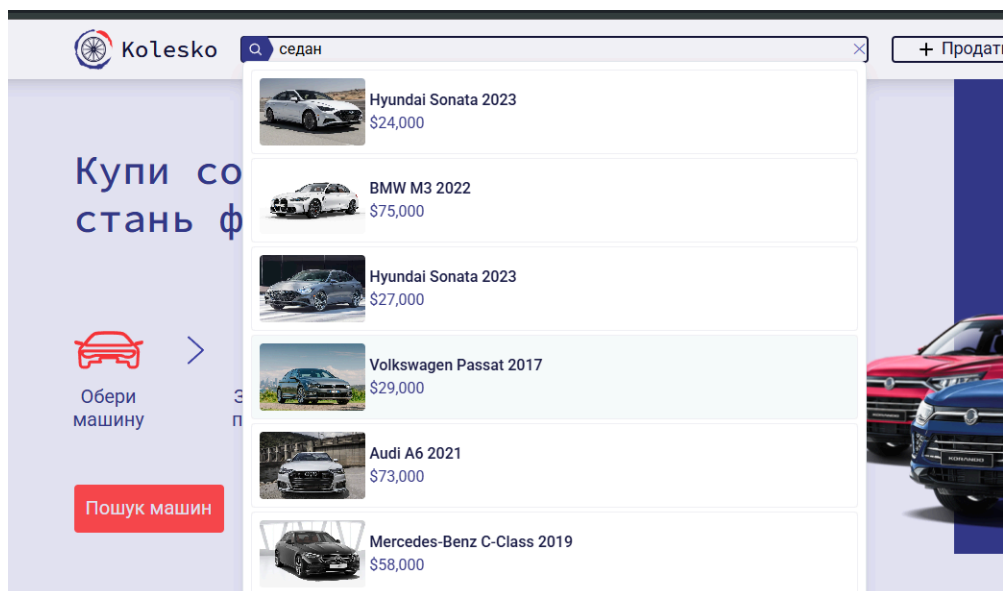


Рис. 4.4 Результат виконання функціоналу кнопки «переглянути ще»

Джерело: розроблено автором

Висновок до розділу

У даному розділі було обґрунтовано вибір фреймворку Angular 18 для реалізації клієнтської частини застосунку на основі порівняльного аналізу з React, Vue.js та Ember.js. Angular забезпечує інтеграцію з TypeScript, повну екосистему інструментів та сувору архітектуру для масштабованих застосунків.

Також було імплементовано систему авторизації з використанням Keycloak, включаючи компоненти HeaderComponent та KeycloakService. Розроблено компонент банеру профілю користувача з адаптивним дизайном та відображенням персоналізованої інформації. Було створено систему пошуку автомобілів з типізованими інтерфейсами та VehicleSearchService. Реалізовано оптимізований пошук з RxJS операторами для мінімізації навантаження на сервер та інтерактивний інтерфейс з випадним списком результатів.

ВИСНОВКИ

Під час виконання кваліфікаційної роботи було проведено дослідження та розробку маркетплейсу вживаних автомобілів, що відповідає сучасним вимогам ринку та користувачів. Для початку було проведено аналіз ситуації на ринку авто у 2023 році, який показав його поступове відновлення та зростання частки внутрішніх перепродажів. Було виявлено ключову роль онлайн-платформ у спрощенні процесі купівлі-продажу. Детальне вивчення існуючих маркетплейсів(AUTO.RIA, RST.UA, AVTOBAZAR.UA) дозволило визначити їхні переваги та недоліки, що стало відправною точкою для формування вимог до нової платформи. Були чітко окреслені сегменти цільової аудиторії та їхні потреби, на основі чого сформовано функціональні та дизайнерські вимоги до проєктованого маркетплейса, зокрема щодо реєстрації, особистого кабінету, подачі оголошень та системи пошуку, з акцентом на сучасні принципи UX/UI.

Було зосереджено увагу на теоретичних засадах та практичній реалізації дизайн системи. Інструментом для дизайну було обрано Figma завдяки її функціональності та можливостям спільної роботи. Для розробки клієнтської частини використовувався Visual Studio Code. Ефективна координація командної роботи та версійний контроль здійснювалися за допомогою GitHub та GitHub Desktop. Була розроблена гармонійна кольорова схема на базі аналогової палітри, впроваджена система кольорових токенів та єдині типографічні стилі з використанням шрифтів Roboto та Source Code Pro. Розробка інфографіки та іконографіки значно покращила користувацький досвід, забезпечивши інтуїтивну навігацію. Комплексний підхід до дизайн-системи гарантував створення естетичного, функціонального та адаптивного інтерфейсу.

На основі аналізу референсів було створено мудборд, що визначив візуальні та функціональні рішення. Розроблено та спроектовано адаптивні сторінки каталогу з фільтрацією та пагінацією, сторінки деталей оголошень з акцентом на зручне представлення інформації, а також сторінки профілю користувача та форми для

додавання/оновлення оголошень. Обґрунтовано вибір фреймворку Angular 18 для клієнтської частини, що забезпечує масштабованість та інтеграцію з TypeScript. Імплементовано надійну систему авторизації за допомогою Keycloak, розроблено профіль користувача та ефективну систему пошуку автомобілів з оптимізацією запитів через RxJS оператори.

Таким чином виконана робота демонструє створення повноцінного маркетплейса вживаних автомобілів: від початкового аналізу ринку та проєктування до розробки детальної дизайн-системи, проєктування ключових інтерфейсів та реалізації технічної частини з використанням сучасних технологій. Розроблена платформа є функціональним, інтуїтивно зрозумілим та адаптивним рішенням, здатним задовольнити потреби цільової аудиторії та стати ефективним інструментом для купівлі-продажу вживаних автомобілів.

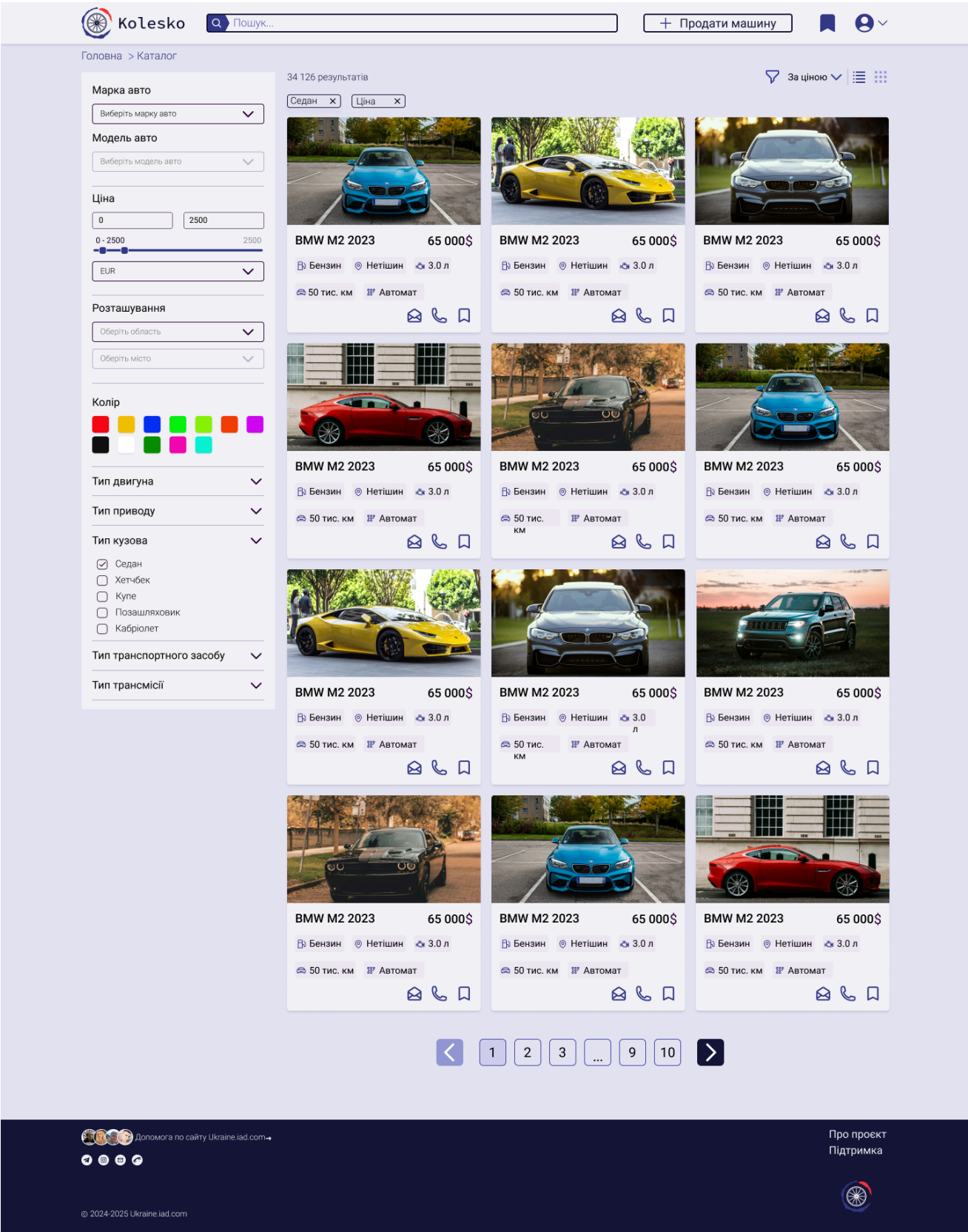
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ukrainian market of passenger used cars: Results of 2023 [Електронний ресурс] – Режим доступу: <https://eauto.org.ua/en/news/480-ukrainian-market-of-passenger-used-cars-results-of-2023> (дата звернення: 20.11.2024).
2. У 2023 році автопарк України поповнили понад 214 тис. вживаних легковиків [Електронний ресурс]. – Режим доступу: <https://ukrautoprom.com.ua/u-2023-roczy-avtopark-ukrayiny-popovnyly-ponad-214-ty-s-vzhyvanyh-legkovykyiv> (дата звернення: 14.05.2025).
3. В Україну за 2023 рік ввезли понад 200 тисяч уживаних авто: найпопулярніші моделі (фото) [Електронний ресурс]. – Режим доступу: <https://focus.ua/uk/auto/618837-v-ukrayinu-za-2023-rik-vvezli-ponad-200-tisyach-uzhivanih-avto-najpopulyarnishi-modeli-foto> (дата звернення: 14.05.2025).
4. Figma [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Figma> (дата звернення: 22.11.2024).
5. Чим Adobe XD так приваблює дизайнерів [Електронний ресурс] – Режим доступу до ресурсу: <https://web4u.in.ua/blog/chim-adobe-xd-tak-privablyu-dizayner-v-33> (дата звернення: 23.11.2024).
6. Figma vs Canva: Which of These Design Tools is Better? [Електронний ресурс] – Режим доступу до ресурсу: <https://theprocesshacker.com/blog/figma-vs-canva/> (дата звернення: 23.11.2024).
7. Comparing Canva Vs Figma for Website Design: Which One Should You Choose? [Електронний ресурс] – Режим доступу до ресурсу: <https://seahawkmedia.com/compare/canva-vs-figma/> (дата звернення: 23.11.2024).
8. What Is GitHub? [Електронний ресурс] – Режим доступу до ресурсу: <https://hackr.io/blog/how-to-use-github> (дата звернення: 23.11.2024).

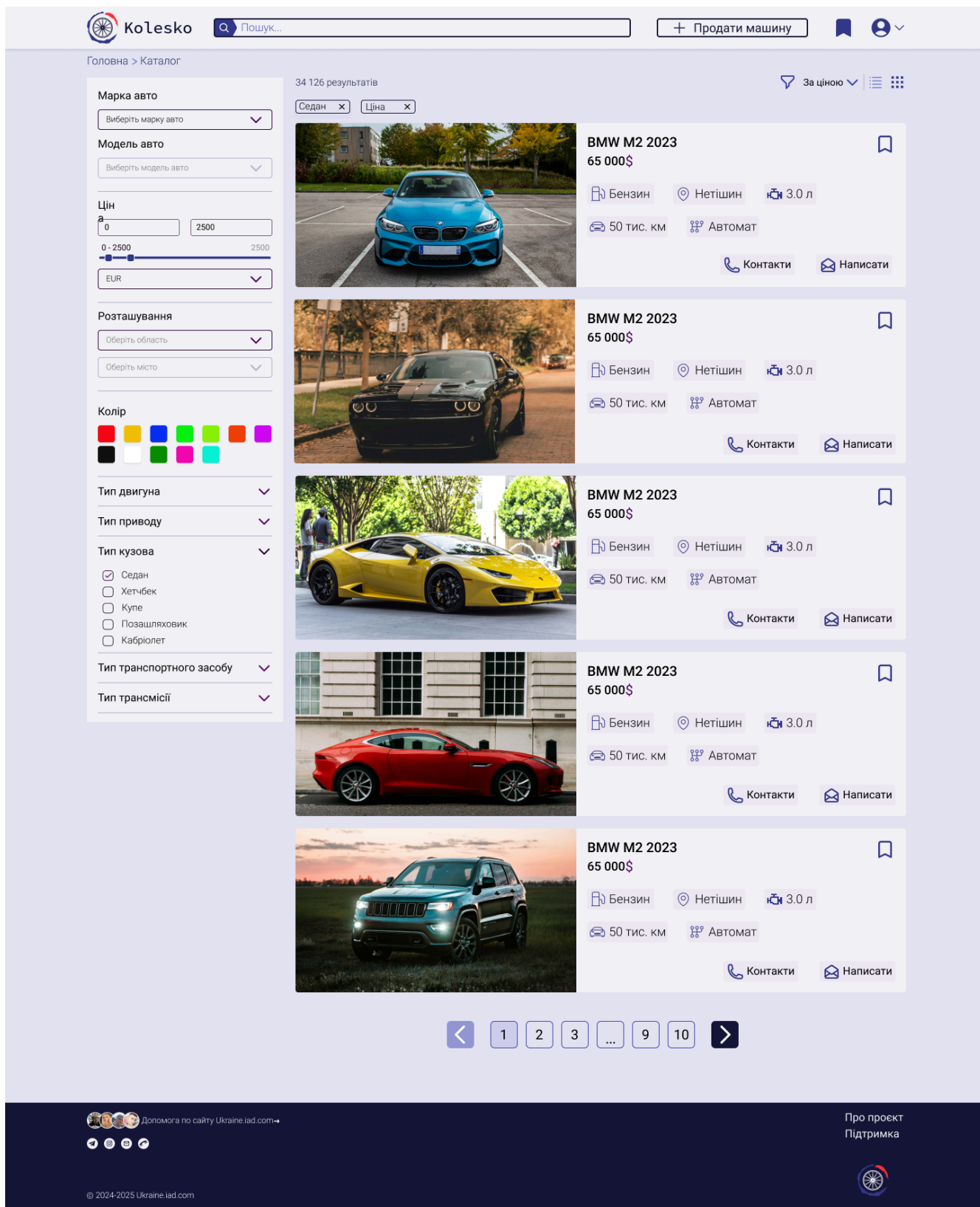
9. 7 колірних схем для гармонійного дизайну [Електронний ресурс] – Режим доступу до ресурсу: <https://blog.depositphotos.com/ua/kolirni-shemy.html> (дата звернення: 24.11.2024).
- 10.tints.dev [Електронний ресурс] – Режим доступу до ресурсу: <https://www.tints.dev> (дата звернення: 24.11.2024).
- 11.Що таке дизайн-токени? Давайте розбиратися разом [Електронний ресурс]. – Режим доступу: <https://ux.pub/editorial/shcho-takie-dizain-tokieni-davaitie-rozbiratisia-razom-5eoa> (дата звернення: 14.05.2025).
- 12.Типографіка в дизайні. Основні поняття і правила [Електронний ресурс] – Режим доступу до ресурсу: <https://iprospect.com.ua/tipografika-v-dizajni-osnovni-ponyattya-i-pravila/> (дата звернення: 25.11.2024).
- 13.Reviewing Roboto: One of the First Widely Used Mobile Fonts [Електронний ресурс] – Режим доступу до ресурсу: <https://typogram.co/font-discovery/how-to-use-roboto-font> (дата звернення: 26.11.2024).
- 14.Source Code Pro About [Електронний ресурс] – Режим доступу до ресурсу: <https://fonts.google.com/specimen/Source+Code+Pro/about> (дата звернення: 26.11.2024).
- 15.Що таке фреймворк: пояснюємо простими словами [Електронний ресурс]. – Режим доступу: <https://brainlab.com.ua/uk/blog-uk/shho-take-frejmvork-poyasnyuyemo-prostymy-slovamy> (дата звернення: 20.05.2025).

ДОДАТКИ

Додаток А



Додаток А. 1 Вигляд дизайну сторінки каталогу



Додаток А. 2 Вигляд дизайну сторінки каталогу