

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему: «Web-додаток організації замовлення дизайну сайту»

за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології проектування»

Виконавець роботи: студентка групи ІТ-81 Кириченко Вікторія Вікторівна

Кваліфікаційна робота бакалавра
захищена на засіданні ЕК
з оцінкою

_____ «__» _____ 2022 р.

Науковий керівник

(підпис)

к.т.н., доц., Парфененко Ю. В.

(науковий ступінь, вчене звання, прізвище та ініціали)

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____

(підпис)

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра інформаційних технологій
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

Зав. кафедрою ІТ

_____ В. В. Шендрик
«05» жовтня 2021 р.

ЗАВДАННЯ

на кваліфікаційну роботу бакалавра студентіві

Кириченко Вікторія Вікторівна

(прізвище, ім'я, по батькові)

1 Тема проекту Web-додаток організації замовлення дизайну сайту

затверджена наказом по університету від «27» квітня 2022 р. №0301- VI

2 Термін здачі студентом закінченого проекту « 10 » червня 2022 р.

3 Вхідні дані до проекту Фото та інформація про виконані роботи для портфолію, фото членів команди організації.

4 Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) аналіз web-дизайну, моделювання діаграм та проектування web-додатку організації замовлення дизайну сайту, реалізація.

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) вступ, аналіз web-дизайну, огляд останніх досліджень і публікацій web-дизайну, аналіз програмних web-продуктів, постановка задачі проекту, моделювання та проектування web-дизайну, структурно-функціональне моделювання web-додатку, моделювання варіантів використання web-додатку, проектування бази даних, практична реалізація web-додатку, архітектура web-додатку, програмна реалізація, використання web-додатку зі сторони користувача, адміністрування сайту, висновки.

6. Консультанти випускної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

Дата видачі завдання _____.

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів випускної проекту	Термін виконання етапів проекту	Примітка
1	Аналіз предметної області		
2	Аналіз аналогів		
3	Постановка задачі та формулювання вимог		
4	Проведення моделювання		
5	Розробка дизайну		
6	Проектування бази даних		
7	Розробка функціональних блоків		
8	Тестування		
9	Наповнення контентом		
10	Перенесення сайту на сервер		

Студент _____

Кириченко В.В.

Керівник роботи _____

к.т.н., доц. Парфененко Ю.В.

РЕФЕРАТ

Тема кваліфікаційної роботи бакалавра «Web-додаток організації замовлення дизайну сайту».

Кваліфікаційну роботу бакалавра присвячено проектуванню та розробці web-додатку організації замовлення дизайну сайту.

В першому розділі проаналізовано предметну область. Виконано огляд підходів по організації замовлення. Виконано аналіз аналогів web-додатку організації замовлення дизайну сайту.

В другому розділі було виконано проектування, розробка та декомпозиція IDEF0. Виконали проектування діаграми варіантів використання.

В третьому розділі виконали реалізацію та детальний опис кожного із модулів web-додатку зі сторони замовника дизайну сайту та адміністратора web-додатку.

Результатом проведеної роботи є розроблений web-додаток організації замовлення дизайну сайту.

Кваліфікаційна робота містить 98 сторінок, 3 таблиці, 94 рисунка, список літератури 20 найменувань, 4 додатки.

Ключові слова: дизайн, web-додаток, замовлення, користувач, команда, відгук, оплата.

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.1 Огляд останніх досліджень і публікацій.....	8
1.2 Аналіз програмних продуктів	10
1.3 Постановка задачі проекту	27
2 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ.....	29
2.1. Структурно-функціональне моделювання web-додатку.....	29
2.2. Моделювання варіантів використання web-додатку	30
2.3. Проектування бази даних	32
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ WEB-ДОДАТКУ.....	38
3.1 Архітектура web-додатку	38
3.2 Програмна реалізація	39
3.3 Використання web-додатку зі сторони користувача	44
3.4 Адміністрування сайту	59
ВИСНОВКИ.....	66
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	67
ДОДАТОК А.....	70
ДОДАТОК Б.....	78
ДОДАТОК В	89

ВСТУП

Web-розробка – це широка сфера з багатьма галузями спеціалізації, однією із яких є web-дизайн. Даний напрямок із легкістю назвати одним з найбільш перспективних напрямків сучасної діяльності, особливо в сфері інформаційних технологій. Попит на професійний веб-дизайн і розробку ніколи не був таким високим, як на сьогоднішній час.

Значний ріст попиту на якісний дизайн є досить прогнозований, адже з розвитком технологій все більше й більше компаній прагне підвищити продажі використовуючи сучасні можливості. Звичайно, запорукою успішності онлайн-бізнесу у двадцять першому столітті є яскравий та функціональний web-додаток. Ознайомившись зі статтями по веб-дизайну, можна зрозуміти, що така професія є не лише дуже прибутковою, але й дуже цікавою.

Для успішної реалізації бізнесу, власнику справи потрібно знайти та звернутись до якісного спеціаліста. Можна знайти дизайнера на фрілансі чи звернутися до організації для замовлення дизайну сайту, серед яких можна віднайти те, що саме підходить користувачу.

Щороку сотні людей запускають нові стартапи, пов'язані із даним напрямком. Для того, щоб відкрити успішний бізнес, потрібно пройти певний перелік етапів. Наприклад, обрання ніші, для професійного розвитку; робота над документацією; створення власного бренду; знаходження клієнтів. Не дивлячись на це все, головним етапом є саме створення власного web-додатку.

Web-додаток не лише служить цифровою базою та списком пропозицій, але й формує весь підхід до цифрового маркетингу. Крім того, web-додаток є головним інструментом та зв'язку із клієнтами. Поганий web-додаток може перешкодити комунікації та навпаки завадити успішному розвитку.

Метою кваліфікаційної роботи бакалавра є розробка web-додатку організації замовлення дизайну сайту.

Задачі, які потрібно реалізувати для розробки web-додатку організації замовлення дизайну сайту:

- ознайомлення з предметною областю сучасного дизайну сайтів;
- аналіз аналогів web-додатку організації замовлення послуг;
- розробка технічного завдання та вимог до проекту web-додатку;
- прототипування макету дизайну web-додатку;
- розробка функціоналу web-додатку організації замовлення;
- наповнення web-додатку контентом.

Результатом проекту з інформаційних технологій є створений web-додаток. Даний web-додаток орієнтований на вдосконалення функціоналу по замовленню дизайну сайтів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

Дизайнери на сьогоднішній час створюють web-додатки як інтерактивні проекти та сайти й такі, що існують просто для гри чи отримання інформації. Даний етап дуже схожий із самим початком становленням Інтернету, коли навіть дизайнери початківці знаходили способи продемонструвати нові та цікаві методи розробки або створювати веб-сайти заради них самих.

Сучасна мода та тенденції також активно запозичує елементи дизайну з минулих років, з розквіту користування інтернетом. У ті роки більша частина користувачів соціальних мереж проводили час вдома, бездіяльно перевіряли сторінки в Інтернеті, шукаючи щось цікаве та захоплююче [1-3].

Значний відсоток візуальної частини в Інтернеті бере приклади з попередніх епох. Все це формує певні тенденції. Дизайнери знаходять креатив за межами стандартних макетів, орієнтованих на фотографію. Натомість більшість проектів знаходять креативні способи використання різноманітної типографії, сіток та ліній, а також оновленої навігації. Оновлений стиль можна описати як вдалу комбінацію із великої кількості приглушених кольорів, витончених текстур та різних шрифтів із засічками.

Веб-дизайн також рухається в майбутнє із самими захоплюючими сучасними методами, такими як розширені взаємодії та анімація та візуальні ефекти. А дизайнери використовують інструменти без коду, щоб зробити це швидше та простіше, ніж будь-коли раніше.

Іноді найефективніший сайт є найменш складним [4-5,8]. Спостерігається зростання популярності односторінкового веб-сайту, який відмовляється від меню та навігації на користь простої навігації з прокручуванням. Односторінкові сайти найкраще працюють, коли їх тематика вужча, як-от портфоліо чи презентація однієї ідеї.

Ці сайти викликають відчуття, що користувач тримає листівку або читає плакат. Вся інформація, яку потрібно переглянути, знаходиться в одному місці, не відволікаючи навігацію чи пошук на кількох сторінках [3,6]. Це дозволяє його предмету бути в центрі уваги. Це збільшує ймовірність того, що випадковий глядач прочитає все резюме, оскільки все, що їм потрібно, є прямо на початку.

Розглянемо поетапність саме організації замовлення послуг (табл.1.1).

Таблиця 1.1. – Організація замовлення послуг

№	Назва	Опис
1	Погодження завдання	Завдання цієї роботи формується, виходячи з аналізу потреб та цілей.
2	Підготовка комерційної пропозиції	На основі поставлених цілей розраховується комерційна пропозиція на розробку з ціною, термінами та списком запланованих результатів.
3	Погодження договору	Після обговорення комерційних умов виконується підготовка договору, що містить узгоджену ціну, термін результати.
4	Розробка	Розробка поставлених цілей за договором.
5	Надання проміжних результатів	У погоджені терміни передається певний результат роботи. Можна ознайомитися з ними, поставити запитання, висловити зауваження протягом узгодженого договору терміну[7].
6	Виправлення зауважень	Якщо у з'являються зауваження, що відповідають договору [8], у встановлений договором строк передається новий варіант з доопрацюванням.
7	Остаточний розрахунок	Після усунення всіх зауважень настає час оплати роботи. У цей момент надсилається рахунок.
8	Передача результатів	Після оплати рахунку надсилається остаточний пакет документів у погодженому договорі форматах.

№	Назва	Опис
9	Супровід	Коли замовнику потрібна допомога у впровадженні розроблених документів у практичне життя, можлива допомога у супроводі застосунку.

Web-додаток організації замовлення дизайну сайту можна буде використовувати для формування та погодження технічного завдання, передача результатів, супровід.

1.2 Аналіз програмних продуктів

Аналіз програмних продуктів конкурентів – спосіб, що допомагає також скласти перелік важливих моментів, що повинні бути безпосередньо проконтрольовані.

Для аналізу було обрано перелік прикладів по організації замовлення дизайну:

- «ОНО» [9];
- «Brainlab» [10];
- «WebEra» [11].

Отже, перейдемо до детального аналізу кожного із представлених конкурентних web-додатків.

1.2.1 Аналіз web-додатку «ОНО»

При переході до web-додатку «ОНО» користувач перенаправляється на головну сторінку (рис.1.1), що несе в собі початкову інформацію про специфіку роботи додатку. Дизайн на головній сторінці є досить лаконічний та добре скомпонований. Користувач має можливість заповнити заявку на дизайн, переглянути соціальні мережі чи зателефонувати за номером, що доданий до web-додатку.

До недоліків можна віднести зливання кольору (рис.1.2), коли при наведенні на іконку додаткових контактів вона зникає, та недостатній розмір кнопки для оформлення запитання (рис. 1.3).

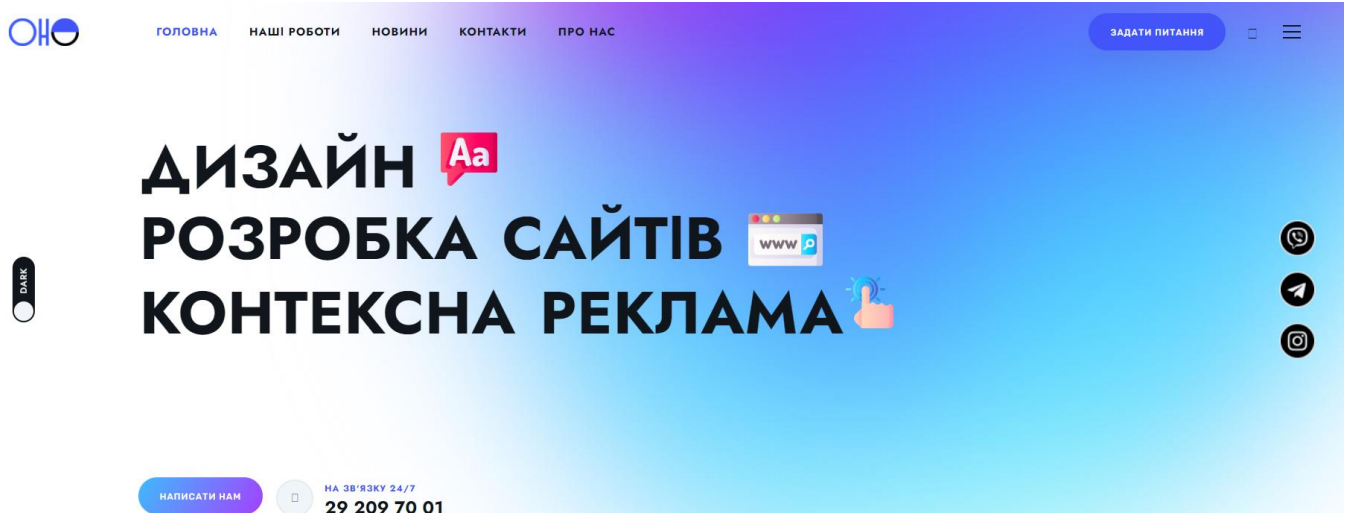


Рисунок 1.1 – Головна сторінка сайту «ОНО»

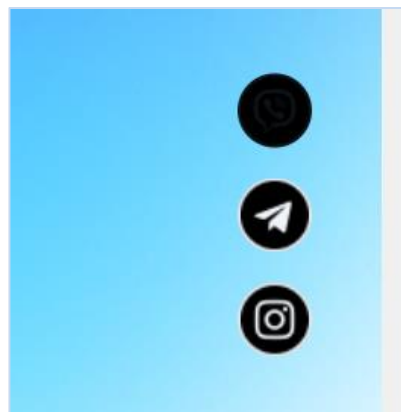


Рисунок 1.2 – Дефект на головній сторінці при наведенні на іконку

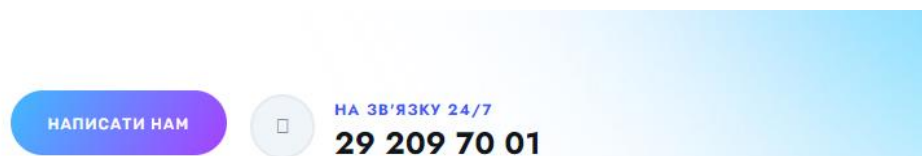


Рисунок 1.3 – Недостатній розмір кнопки для запитання

Окремо хотілось би виділити також підбір деяких кольорів. При використанні синього кольору на чорному (рис.1.4) або білого на сірому (рис.1.6) значно знижується читаємість тексту. Також на рис.1.5 представлено приклад заголовку одного із блоку на головній сторінці. Обрані відступи між блоками є невірними, адже все становиться не розбірливим та візуально формується в одну інформативну групу.

Все це призводить до ускладнення користування web-додатком «ОНО». Якщо у користувачів виникають труднощі при використанні, то це може їх збентежити й тим самим не оформити замовлення.

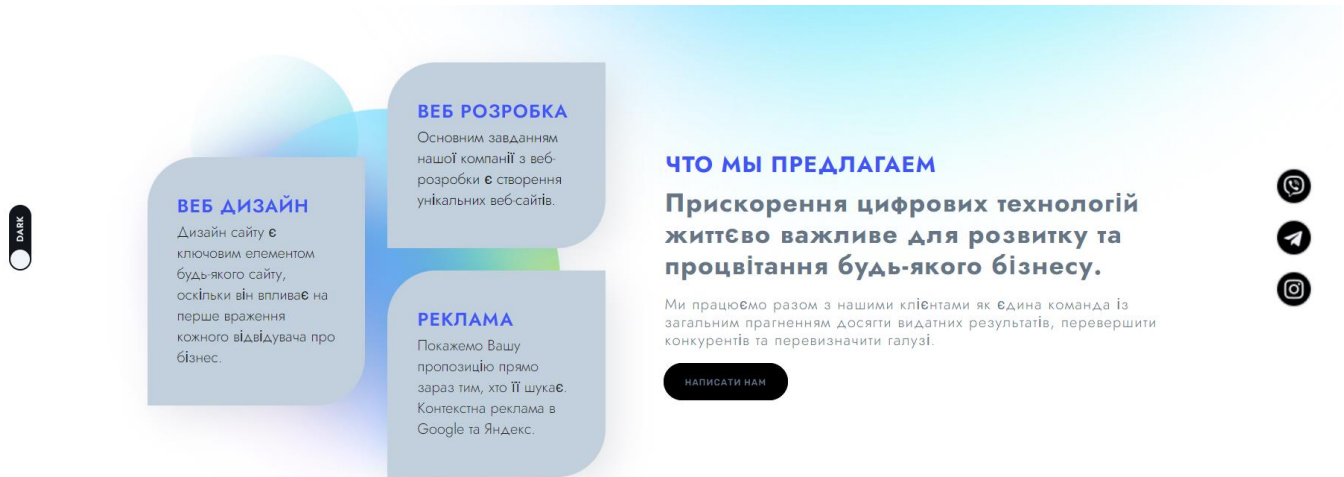


Рисунок 1.4 – Інформаційний блок на головній сторінці

Ми тут, щоб зробити відмінний продукт!

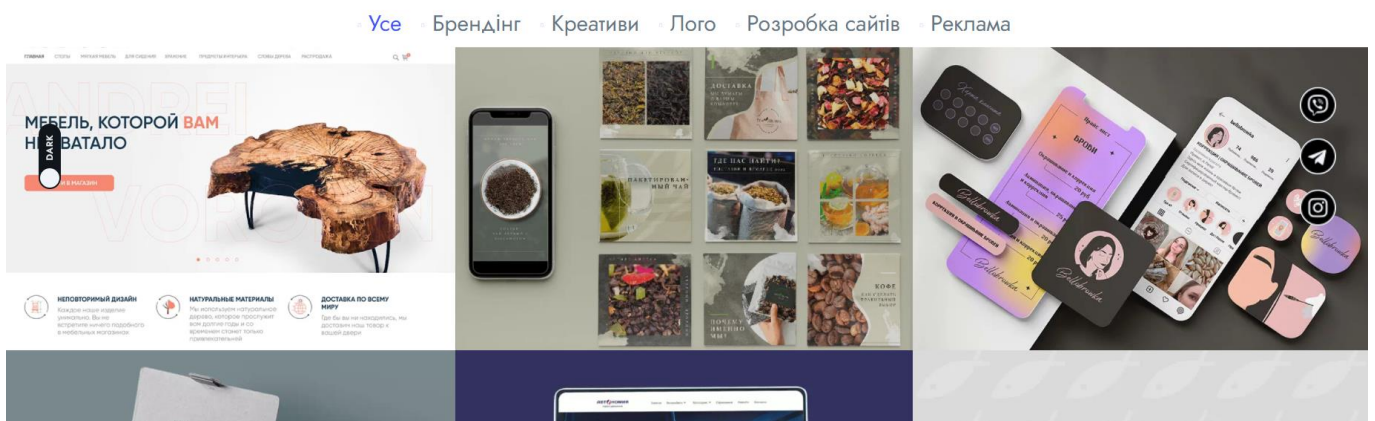


Рисунок 1.5 – Приклади з портфолію на головній сторінці

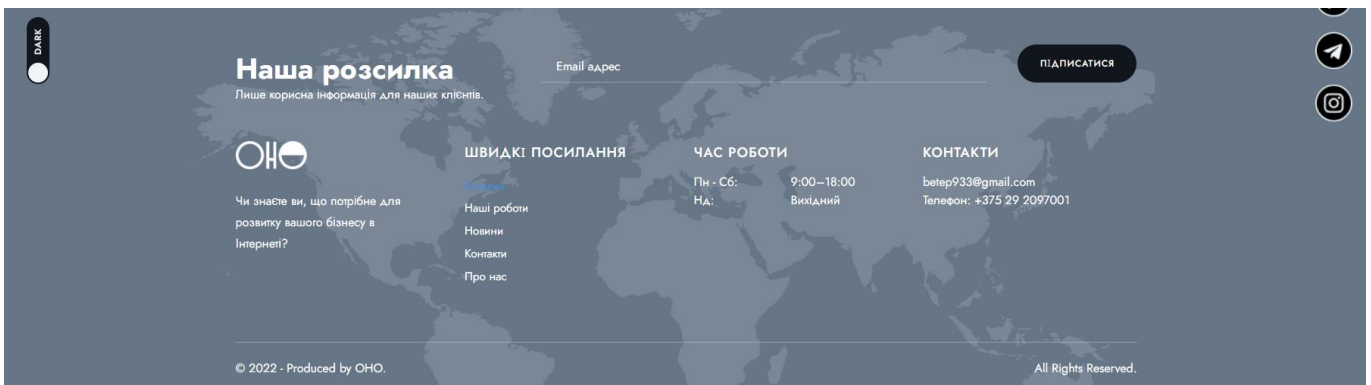


Рисунок 1.6 – Хедер із посиланнями

Однією із найважливіших сторінок для web-додатку організації по замовленню дизайну є сторінка із портфолію (рис.1.7). На даній сторінці все виконано досить лаконічно та відповідає загальному стилю. При натисканні на один із прикладів в портфолію користувач перенаправляється на сторінку із детальною інформацією про обрану роботу.

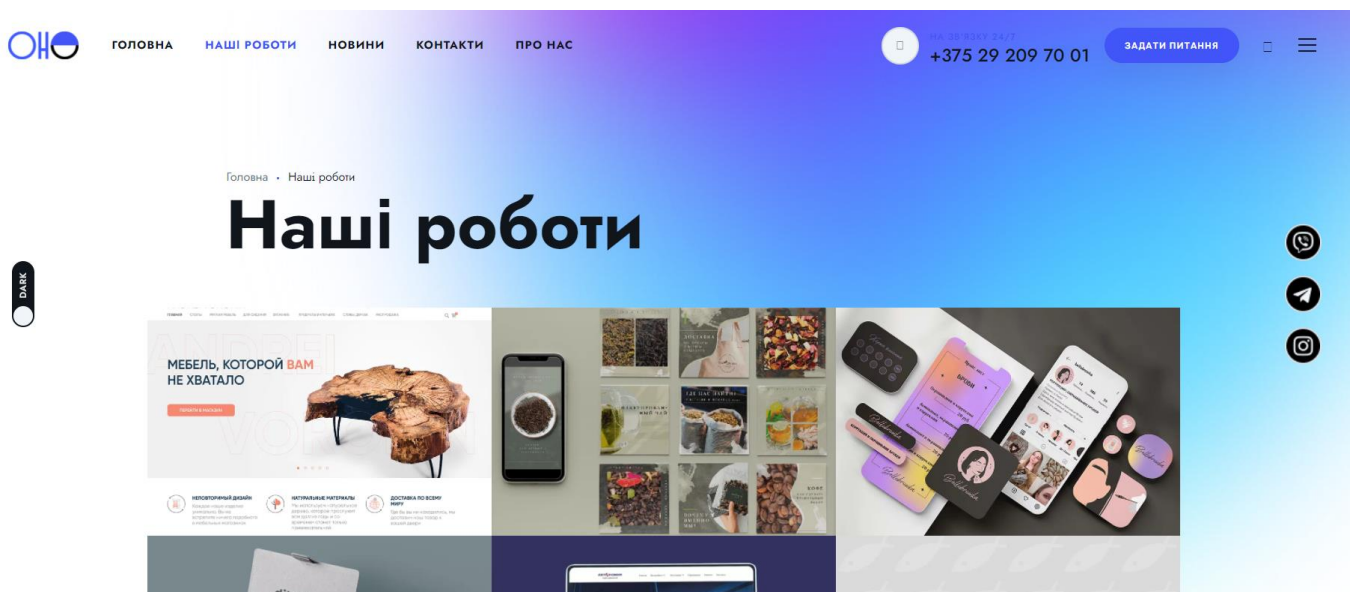


Рисунок 1.7 – Сторінка із виконаними роботами

Наступною є сторінка із новинами та іншими дописами (рис.1.8). Блок із новинами добре виконаний, заголовки зрозумілі та відповідають контексту допису. Відступи між заголовком та самим наповненням лише сприяють зручності у використанні та перегляді інформації.

Не дивлячись на це, в негативну сторону виділяється блок із короткою інформацією про сторінку, останніми дописами та формою для пошуку. Так як форма має сірий колір, тому відбувається злиття кольорів. Обране поєднання може збити користувача, який бажає виконати пошуку допису.

Крім того, відсутня можливість фільтрувати новини за певними критеріями, як наприклад, дата публікація, тематика, автор та тощо.

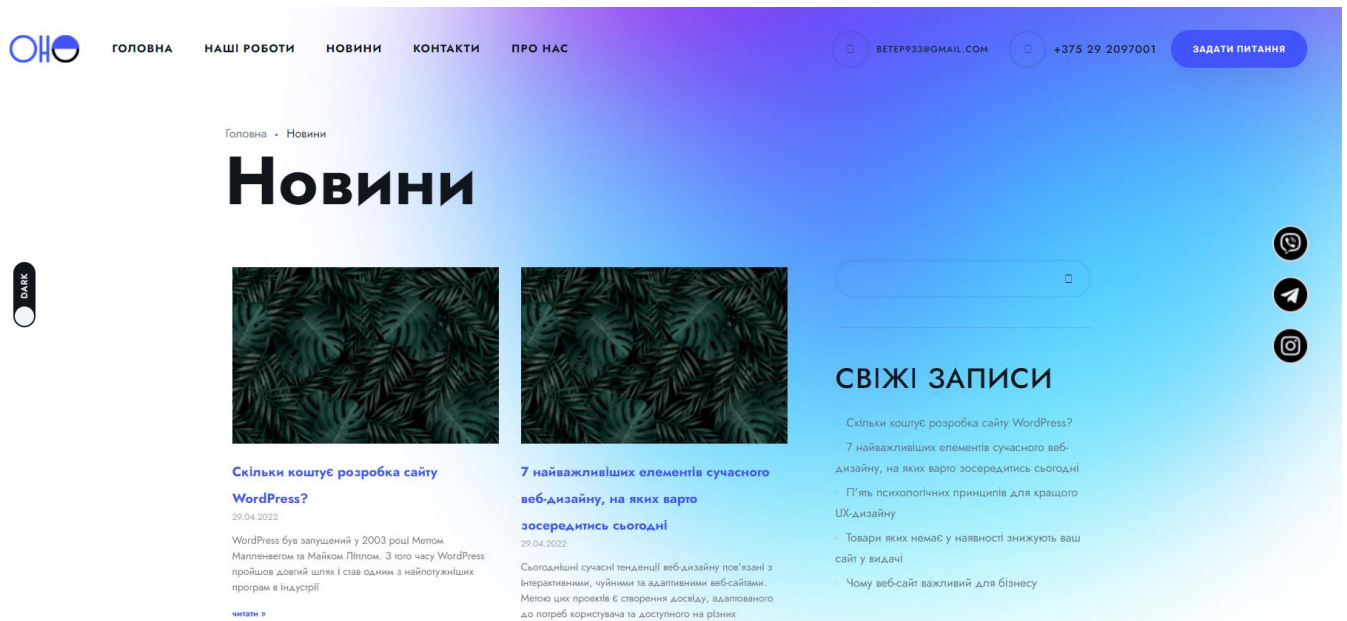


Рисунок 1.8 – Сторінка із новинами

Сторінка із контактами містить в собі всю необхідну інформацію для комунікації із потенційними замовниками (рис.1.9). На даній сторінці знаходиться як контактні номери, електрона пошта, так й час роботи організації.

Останньою сторінкою є інформація про організацію (рис.1.10). Гарна ідея та реалізація є додавання досягнень організації, що й представлено у web-додатку «ОНО».

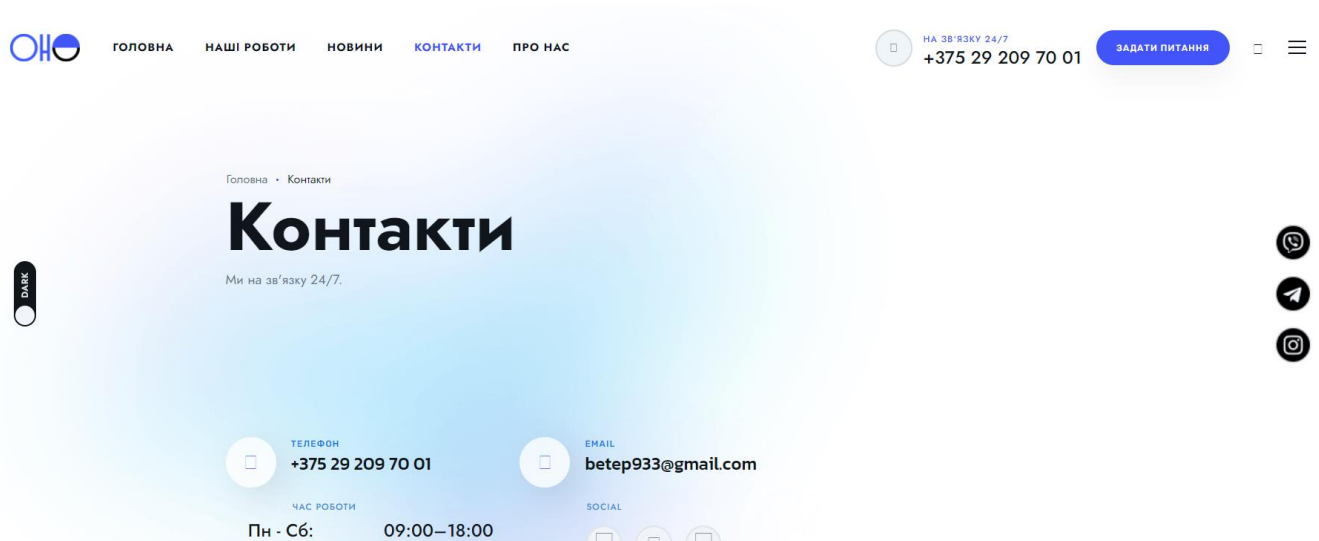


Рисунок 1.9 – Сторінка із контактами

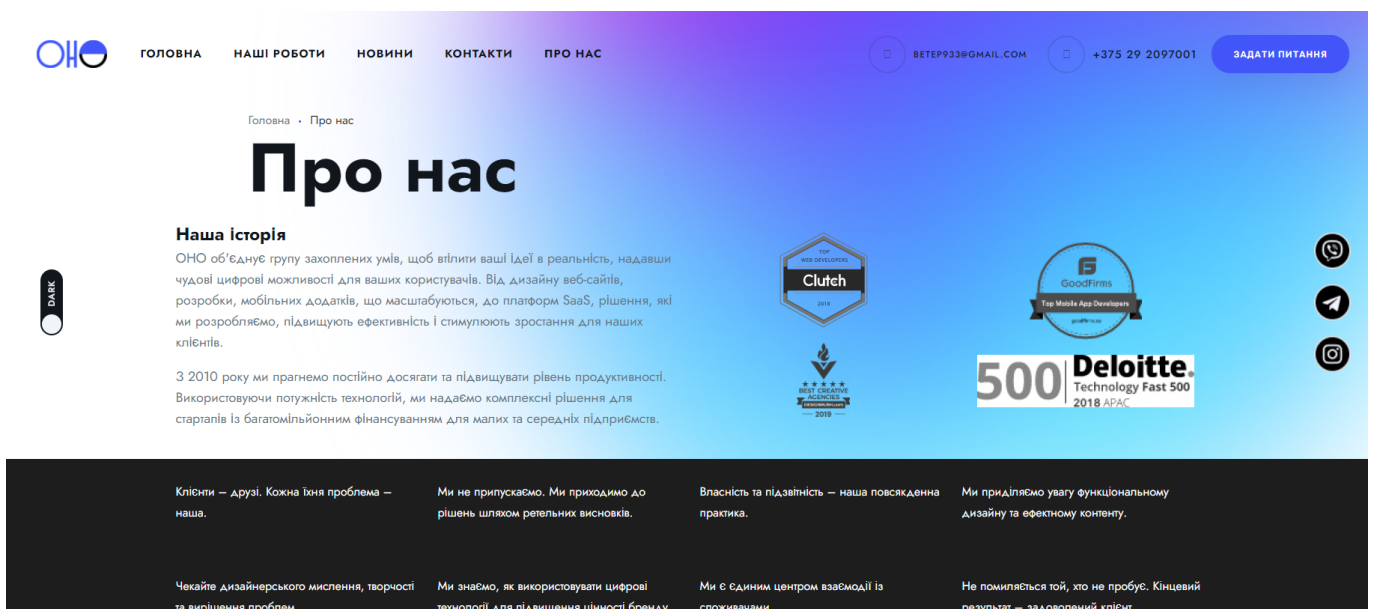


Рисунок 1.10 – Сторінка із короткою інформацією про організацію

Важливим аспектом є також адаптивність web-додатку під мобільні пристрої та планшети. Переглядаючи адаптивність web-додатку «ОНО» під мобільні пристрою проблем не було виявлено (рис.1.12).

Хоча, при адаптивності під планшети з'явилися певні дефекти (рис.1.11). Недоліки є незначнимим, але вони значно впливають на загальне сприйняття при користуванні.

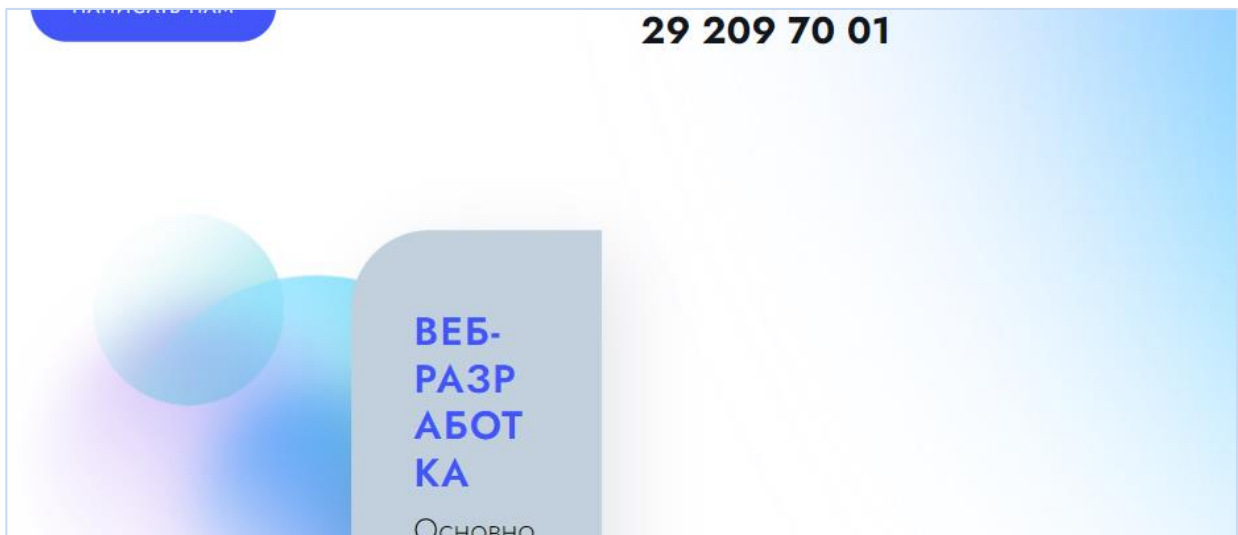


Рисунок 1.11 – Головна сторінка при розширенні для планшета

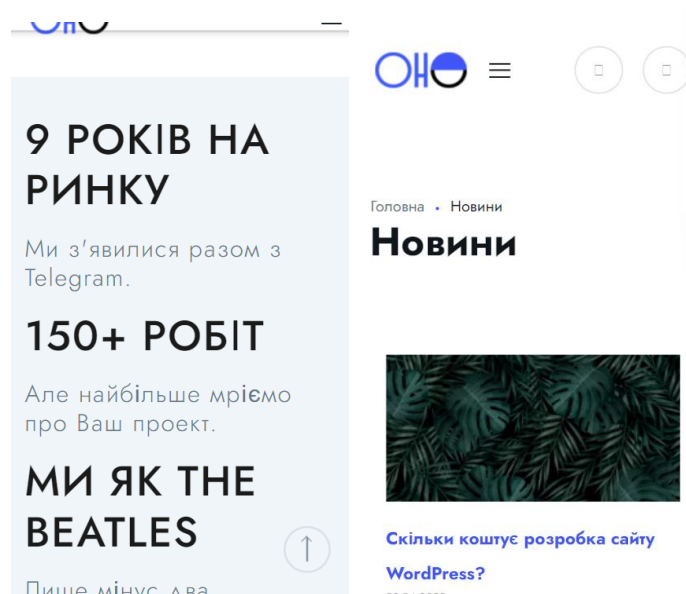


Рисунок 1.12 – Головна сторінка та новини при розширенні для мобільного телефону

1.2.2 Аналіз web-додатку «Brainlab»

Перейшовши за посиланням web-додатку «Brainlab», користувач перенаправляється на головну сторінку (рис.1.13). На даній сторінці знаходиться короткий заголовок та кнопки для швидкої навігації за цікавими напрямками.

Крім того, в даному web-додатку якісно виконаний дизайн навігаційного меню. В даному меню можна легко виконувати навігацію. Крім того хотілось би позитивно відмітити можливість змінити мову відразу в меню.

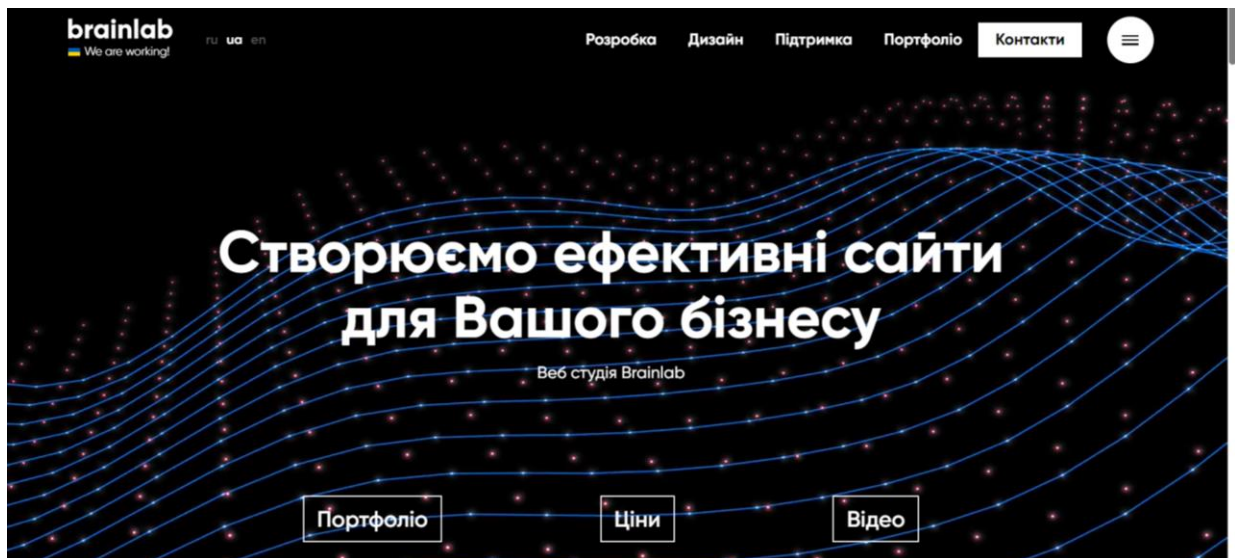


Рисунок 1.13 – Головна сторінка web-додатку «Brainlab»

Користувач web-додатку «Brainlab» також має можливість використовувати розширене навігаційне меню. В розширеному меню є можливість перейти на такі сторінки, як послуги, технології, портфоліо, інформація про організацію, блог та контакти.

При наведенні на один із пунктів меню з'являється розширене підменю. На рисунку 1.14 представлено приклад із пунктом «Технології». При наведенні користувачеві відкривається можливість перейти та ознайомитися із детальною інформацією що до WordPress, Opencart та OkayCMS.

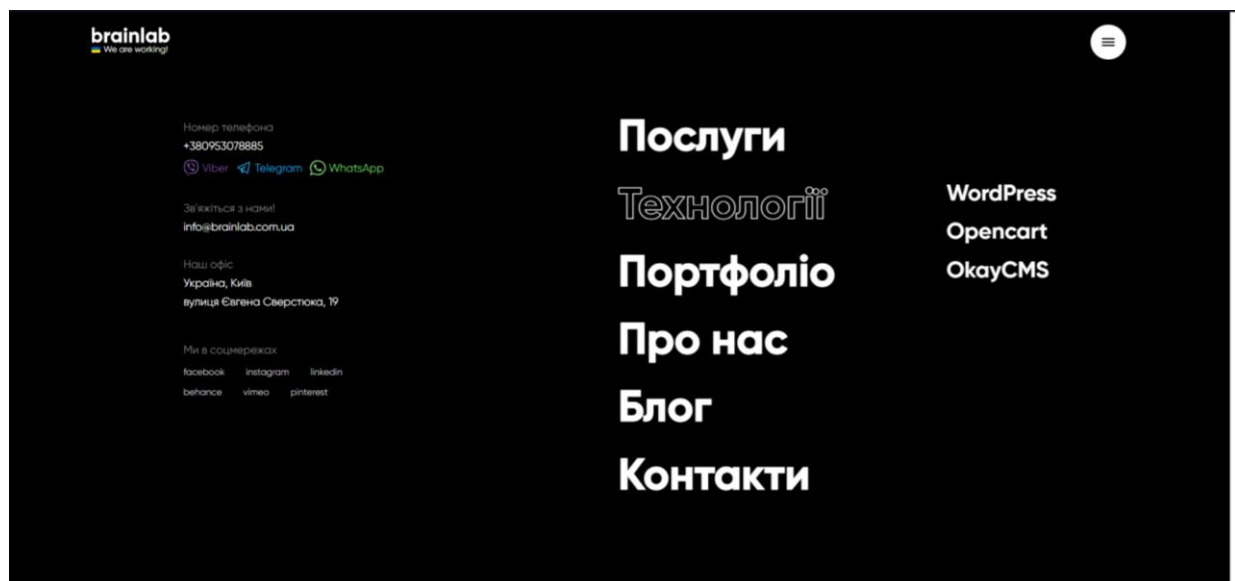


Рисунок 1.14 – Розширене навігаційне меню

На сторінці із послугами користувач може продивитися варіанти роботи, що виконує компанія (рис.1.15). Наприклад такі, як корпоративний сайт, інтернет магазин, дизайн сайту, бізнес-сайт, сайт-візитка та інше.

Крім того, користувач може відразу переглянути й ціну відповідної послуги. Даний момент значно полегшує користування web-додатком «Brainlab». При натисканні на послугу користувач перенаправляється на сторінку із детальною інформацією про обрану послугу.

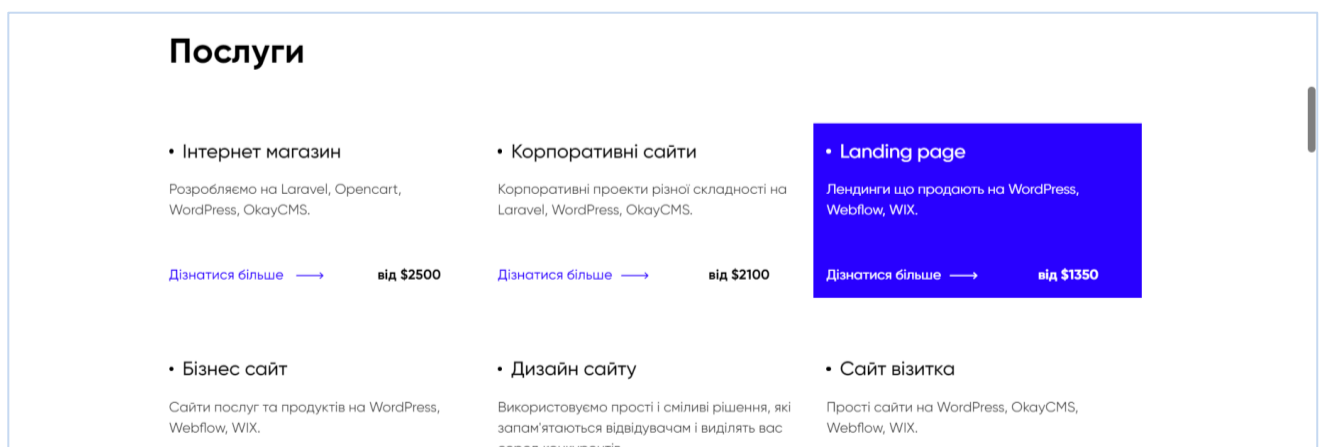


Рисунок 1.15 – Сторінка із послугами організації

Також користувач має можливість задати запитання (рис.1.16). Для того, щоб задати запитання користувач повинен заповнити форму, а саме:

- Ім'я;
- Номер телефону;
- Електронна пошта;
- Тип сайту.

brainlab ru ua en
We are working!

Розробка Дизайн Підтримка Портфоліо Контакти

Є питання?

Давайте обговоримо ваші завдання. Зв'яжіться сьогодні!

Введіть ваше ім'я *

Номер телефону *

Email *

Тип сайту

Надіслати

Рисунок 1.16 – Форма для запитань від користувачів

У web-додатку «Brainlab» окремо виведено також сторінка для замовлення редизайну сайту (рис.1.17). Також на кожній із сторінці пропозиції від організації знаходиться додаткова інформація. Як наприклад на рисунку 1.18 відображена інформація про поетапність створення інтернет магазину.

brainlab ru ua en
We are working!

Розробка Дизайн Підтримка Портфоліо Контакти

Веб студія • Редизайн сайту

Редизайн сайту

Замовити проект

Дизайн сайту - перше, на що звертають увагу його відвідувачі. Він вселяє довіру до компанії, підсилює бренд, збільшує продажі. Але якщо дизайн сайту не виконує ці завдання? Тут одне з двох: або він застарів, або зроблений непрофесійно. І в будь-якому випадку знадобиться редизайн сайту.

Що таке редизайн сайту і які завдання він включає?

Редизайн сайту - не просто зміна оформлення. Він також передбачає серйозні правки в контенті і функціональності веб-ресурсу. Аж до переїзду майбутнього на інший движок. Виходячи з вищого замовника редизайн веб сайту включає в себе різні пункти.

Рисунок 1.17 – Сторінка для замовлення редизайну

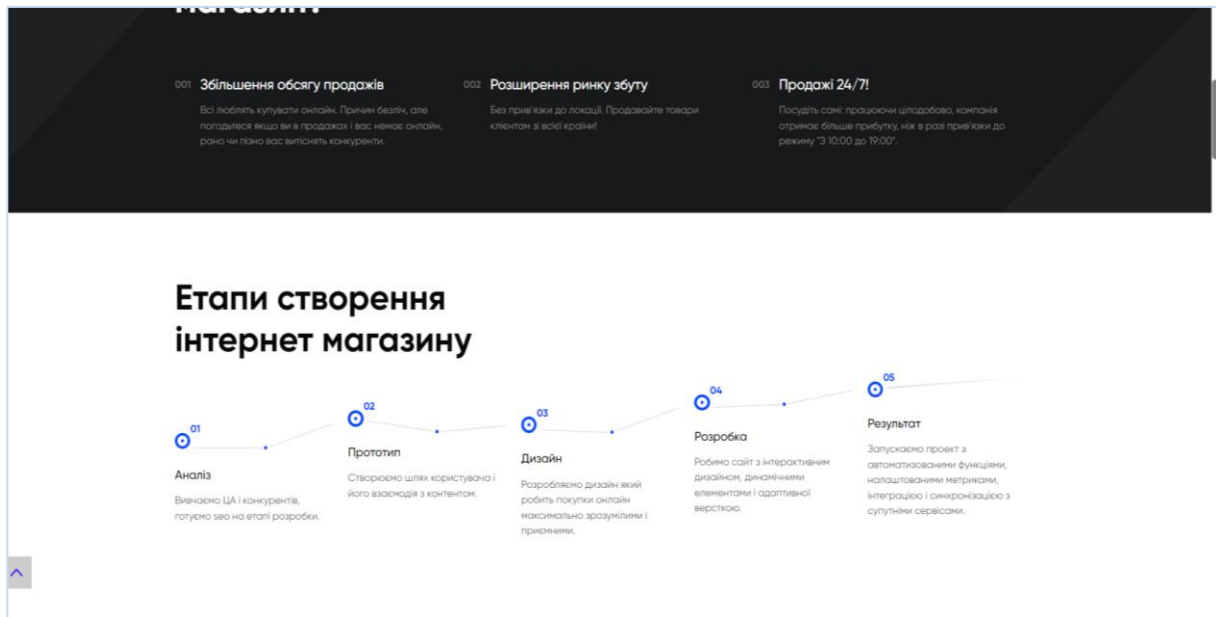


Рисунок 1.18 – Інформація про розробку інтернет магазину

На сторінці з прикладами можна розглянути популярний спосіб із категоріями робіт (рис.1.19). На даному сайті було згруповано роботи в такі теми як:

- Інтернет-магазини/Каталоги;
- Корпоративні/Бізнес сайти;
- Дизайн.

На сторінці із контактами також має можливість надіслати задати запитання (рис.1.20).

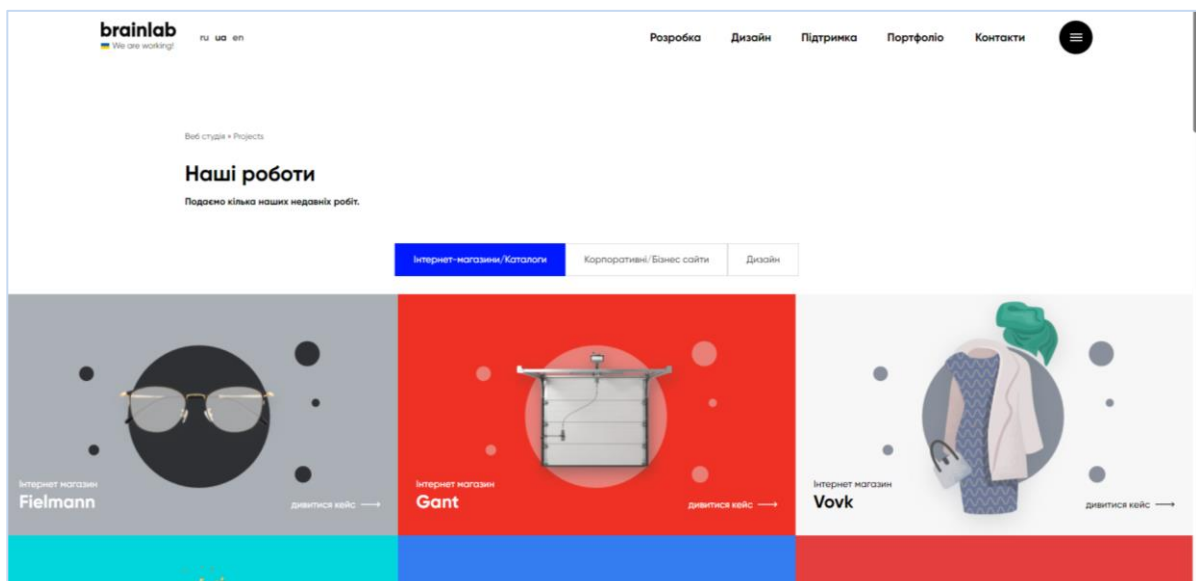


Рисунок 1.19 – Сторінка із прикладами робіт

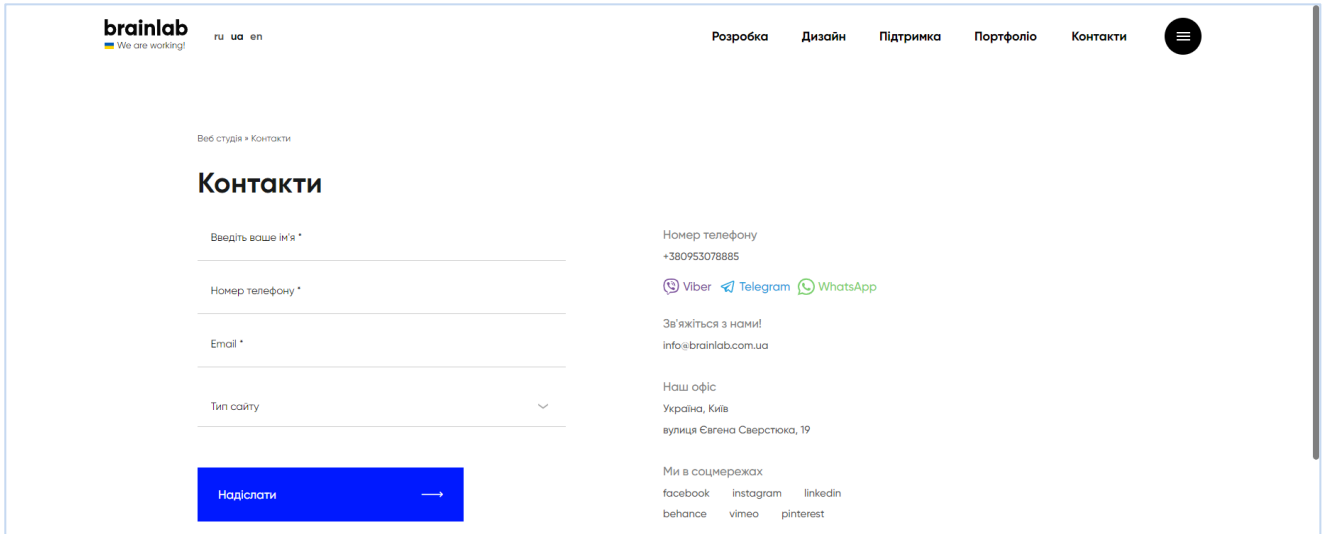


Рисунок 1.20 – Сторінка із контактами

При тестуванні мобільної адаптації ніяких помилок не було знайдено (рис.1.21). На прикладі представлено адаптивні сторінки web-додатку «Brainlab», а саме головна, послуги та меню.

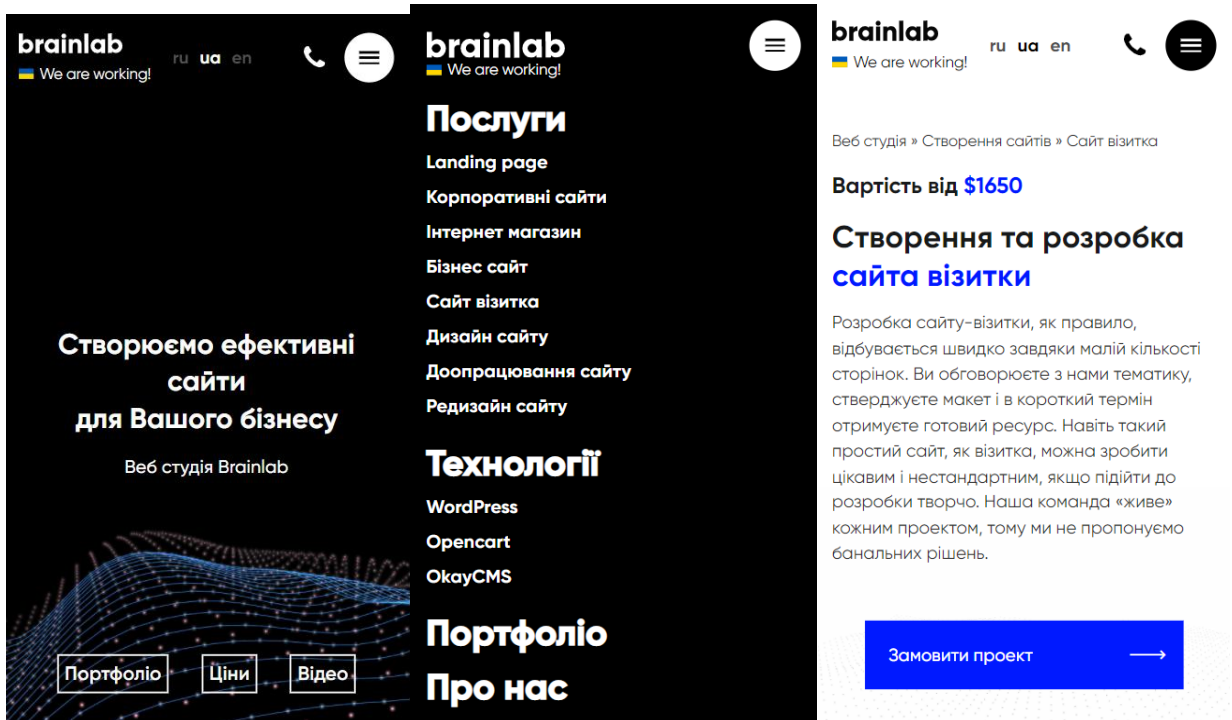


Рисунок 1.21 – Мобільна адаптація web-додатку «Brainlab»

1.2.2 Аналіз web-додатку «WebEra»

Перейшовши за посиланням на сторінку «WebEra», користувач web-додатку, як прийнято, потрапляє на головну сторінку (рис.1.22-23). На даній сторінці знаходиться перелік послуг, що представляє організація.

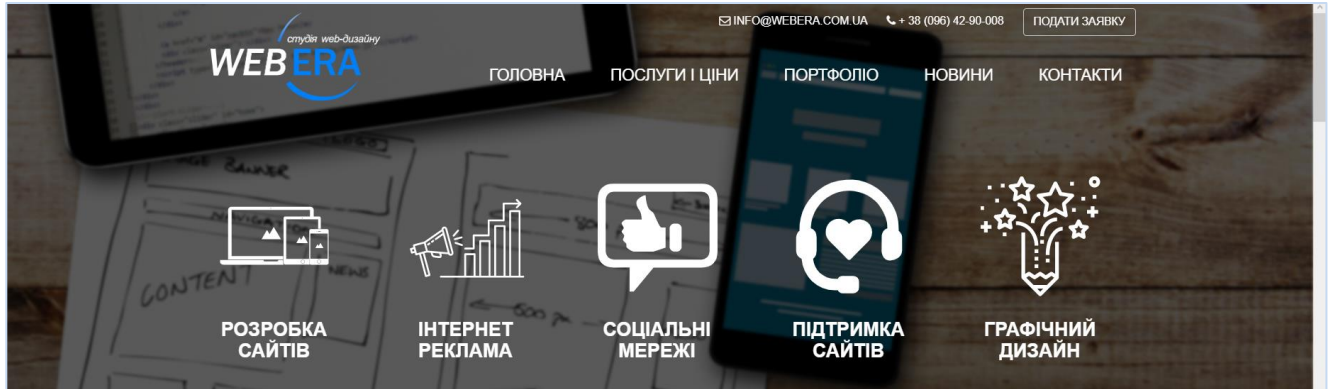


Рисунок 1.22 – Головна сторінка web-додатку «WebEra»

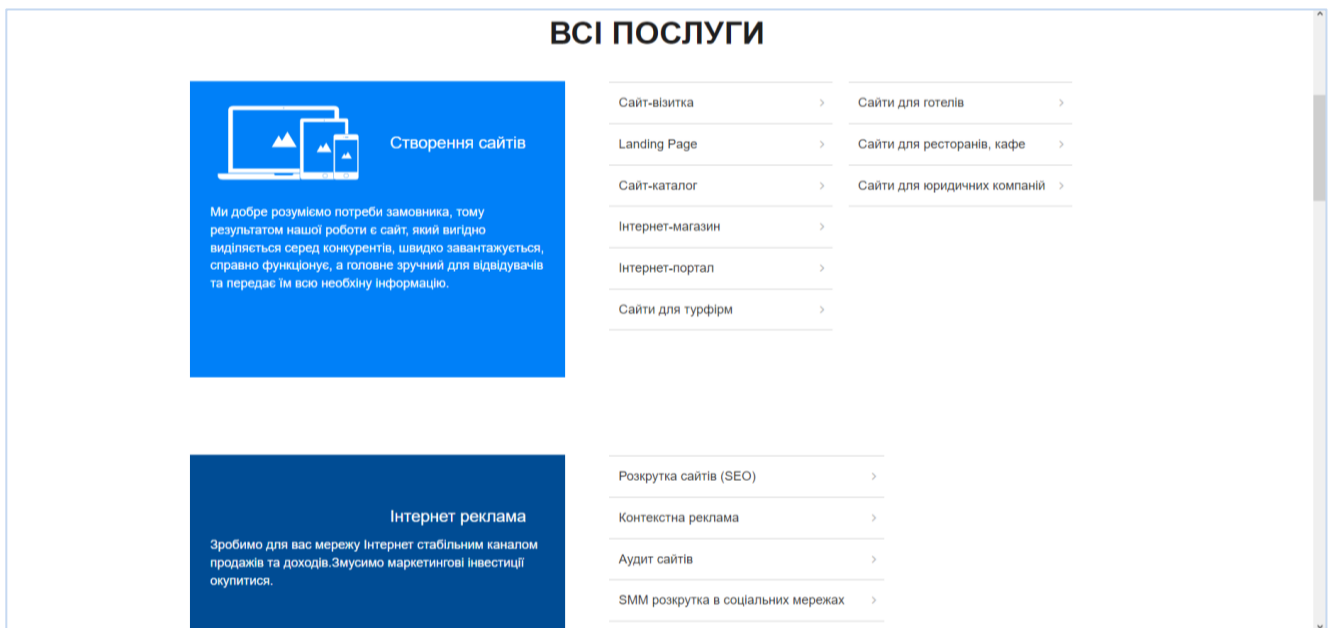


Рисунок 1.23 – Сторінка із переліком послуг web-додатку «WebEra»

Крім того, на головній сторінці після переліка послуг web-додатку «WebEra» присутній блок із кнопкою перенаправлення на надсилання заявки (рис.1.24).

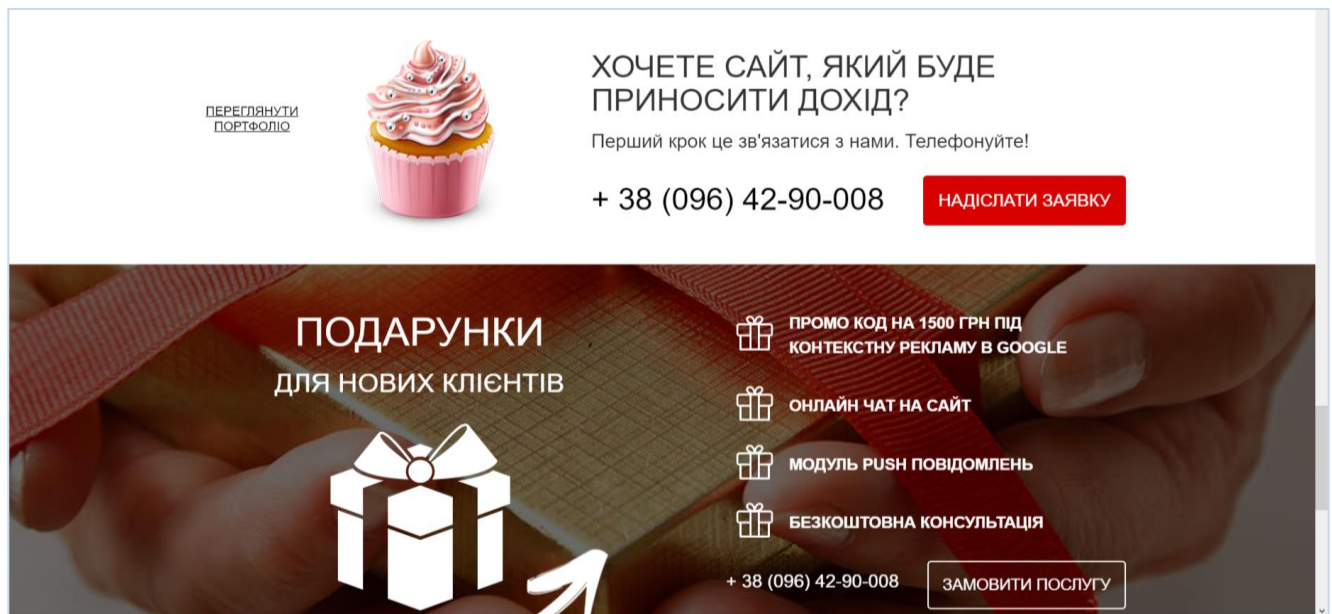


Рисунок 1.24 – Блок із кнопкою перенаправлення на надсилання заявки

При натисканні на кнопку «Надіслати заявку» відкривається модальне вікно (рис.1.25). Для надіслання заявки потрібно заповнити такі поля як:

- Ваше ім'я;
- Ваш номер телефону;
- Email адреса;
- Коментар.

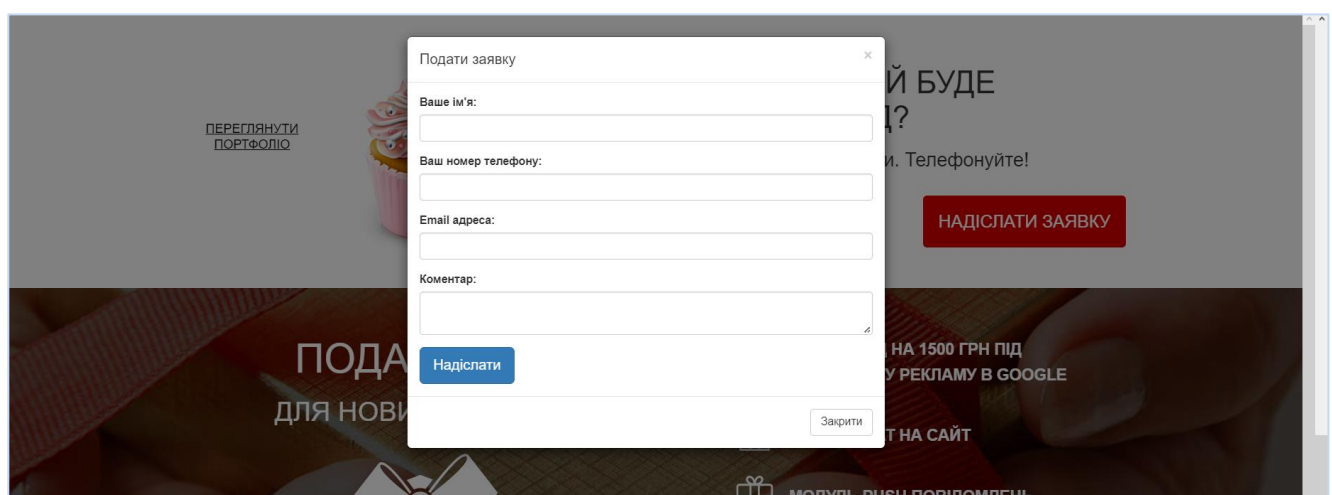


Рисунок 1.25 – Головна сторінка web-додатку «WebEra»

Звичайно, даний web-додаток має такі сторінки, як «Портфоліо» (рис.1.26), «Блог» (рис.1.27) та «Контакти» (рис.1.28).

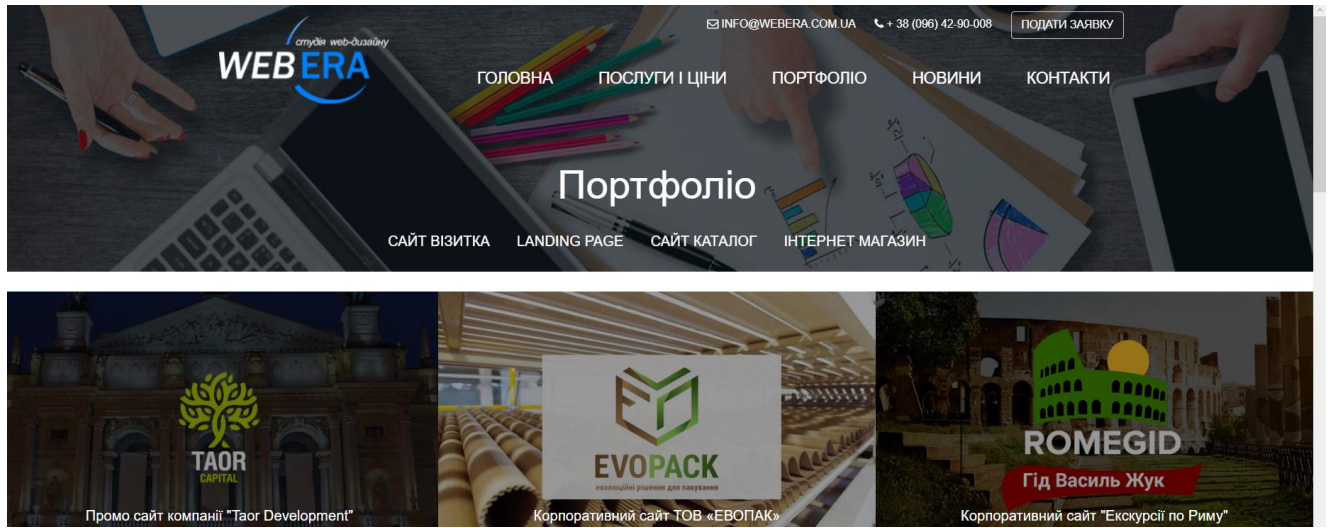


Рисунок 1.26 – Сторінка із портфоліо

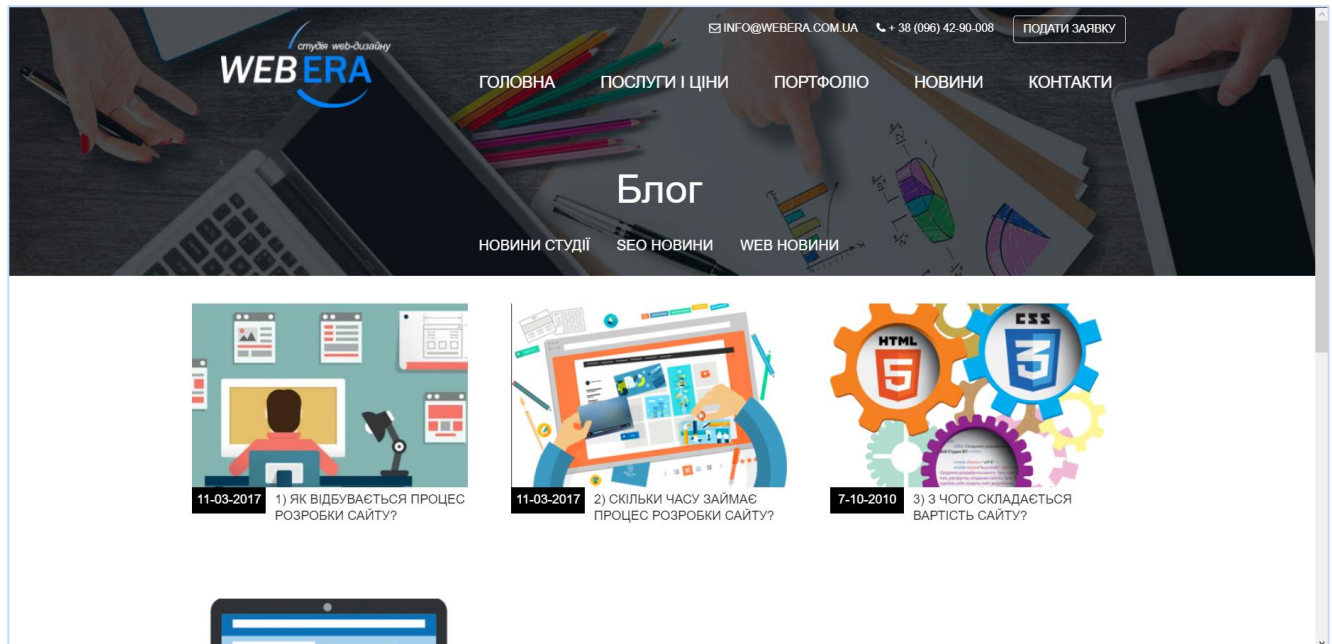


Рисунок 1.27 – Сторінка із блогом

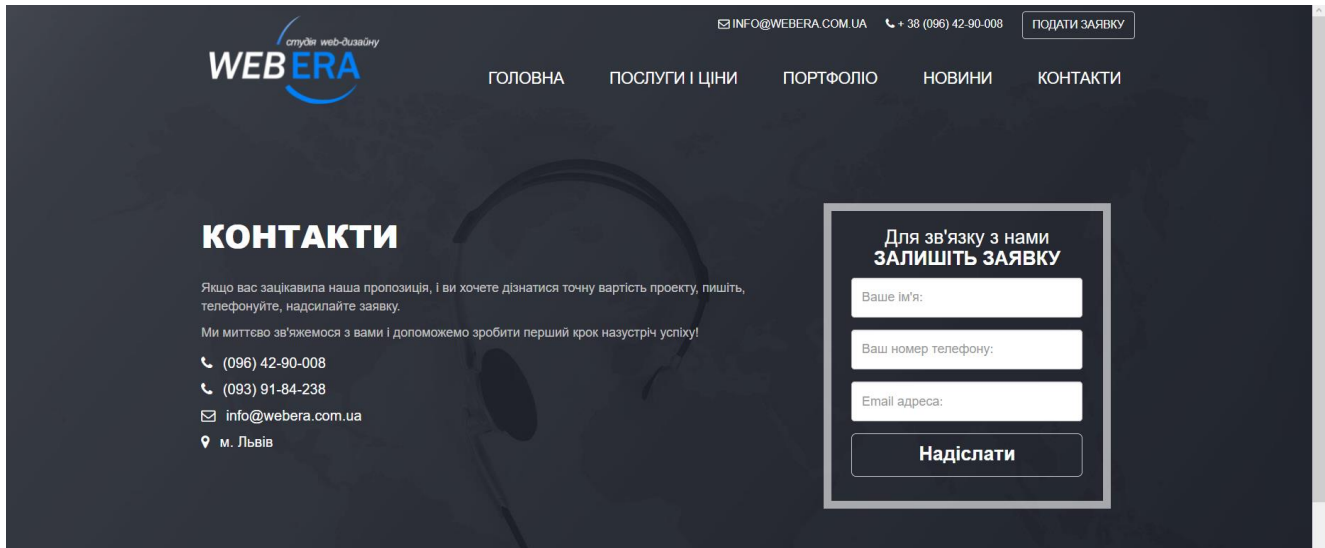


Рисунок 1.28 – Сторінка із контактами

При тестуванні мобільної адаптації (рис. 1.29) було знайдено декілька дефектів, що пов'язані із невірним використанням зображень та використання стилів (рис. 1.30).

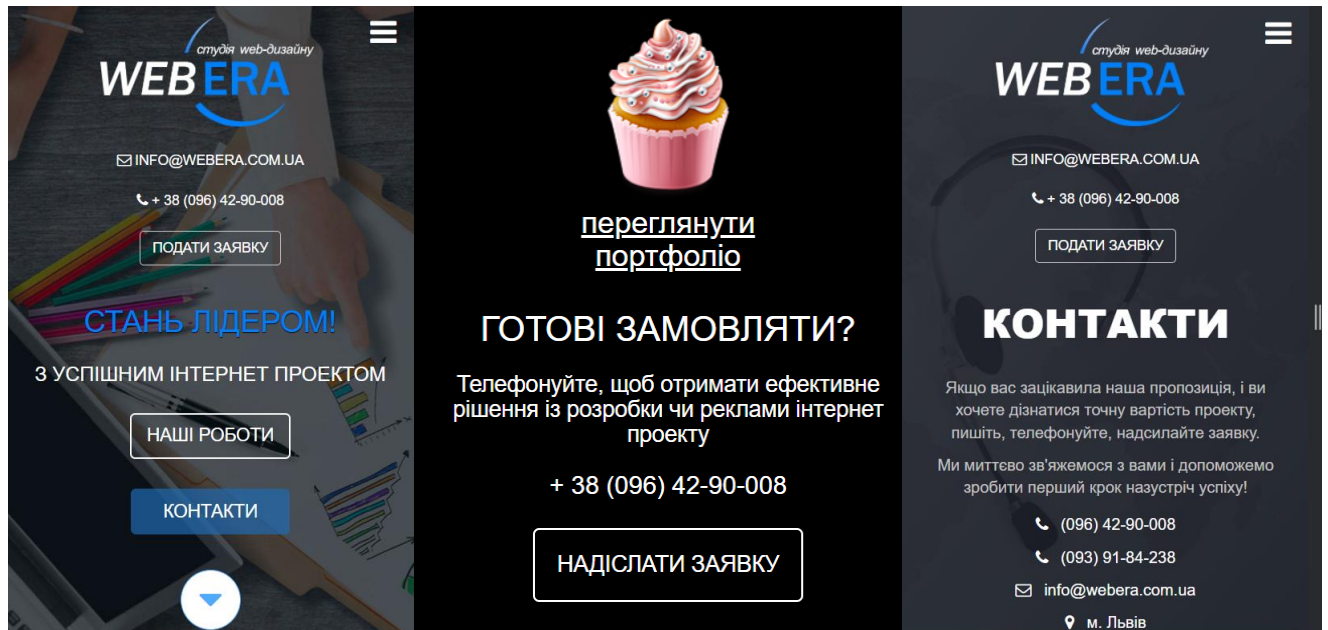


Рисунок 1.29 – Мобільна адаптація web-додатку «WebEra»

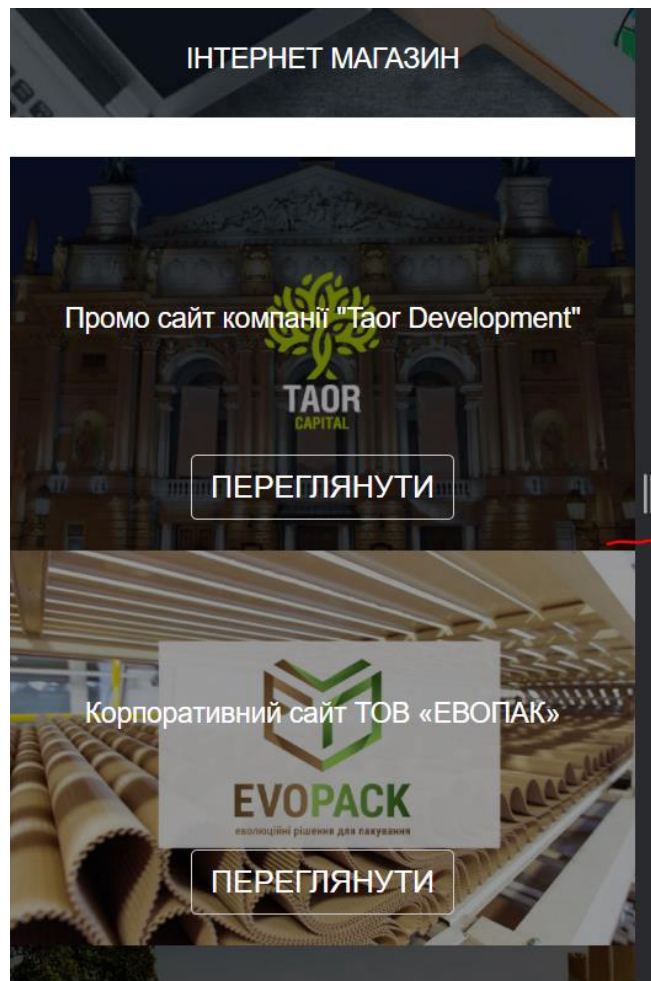


Рисунок 1.30 – Дефект мобільної адаптації

Проаналізувавши web-додатки організацій по замовленню дизайнів та сайтів, було виконано порівняльний аналіз (табл.1.2), де було виявлено позитивні та негативні сторони, які при створенні web-додатку для організації замовлення треба уникати.

До недоліків можна віднести:

- Не повна валідація даних;
- Відсутність відгуків про виконані роботи;
- Проблеми із мобільною адаптацією;
- Невдале розміщення інформації.
- Низький рівень дизайну та загального оформлення.

До переваг web-додатків, що використовувалися для порівняння, можна віднести:

- Розподіл по видам дизайнів та інших пропозицій;
- Можливість задавати запитання;
- Навігація.

Таблиця 1.2 – Аналіз web-додатків організацій по замовленню дизайнів

Критерії	«ОНО»	«Brainlab»	«WebEra»
Валідації даних в формах	+	+	–
Категоризація робіт	+	–	–
Реалізація зворотнього зв'язку	–	+	–
Навігація	–	+	+
Розміщення інформації	+	+	+
Web-дизайн	+	+	–
Адаптивність	+	+	+

Отже, web-додаток організацій замовленню дизайну сайтів повинен мати зручну навігацію, зрозумілий дизайн та широкий спектр функціоналу.

1.3 Постановка задачі проекту

Для успішної реалізації бізнесу, а саме організацій замовленню дизайну сайтів, потрібно мати гарну реалізацію web-додатку. Web-додаток не лише буде служити цифровою базою та списком пропозицій, але й формуватиме весь підхід до цифрового маркетингу. Крім того, web-додаток є головним інструментом.

Основні функції web-додатку для незареєстрованих користувачів:

- реєстрація користувачів;
- перегляд інформації на сайті про команду;
- створення пошуку для формування списку робіт за різними критеріями.

Основні функції web-додатку для зареєстрованих користувачів, що розширюють вже отриманий функціонал:

- авторизація користувачів;
- створення заявок на отримання роботи з можливістю прикріплення файлів;
- формування відгуків по виконаній роботі.

Основні функції web-додатку для адміністраторів:

- додавання приклади та історій виконаних проектів;
- розміщення на web-додатку прикладів роботи з можливістю прикріплення

фотографій та посилань на реалізацію;

- формування оповіщень;
- редагування даних зі сторони адміністративної панелі;
- взаємодія із замовником через адміністративну панель;
- можливість формування автоматичної відповіді та розсилок на електронну

пошту через адміністративну панель.

Отже, web-додаток використовуватиметься як система для організації замовлень дизайну сайту. Даний сервіс зможе допомогти полегшити роботу із користувачами. Крім того, самі користувачі зможуть із легкістю ознайомитися із роботами команди.

2 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ

2.1. Структурно-функціональне моделювання web-додатку

Структурно-функціональне моделювання web-додатку організації замовлення дизайну сайту представляє з себе розробку контекстної діаграми (рис.2.1). Головна задача web-додатку – формування та виконання замовлення по дизайну сайту.



Рисунок 2.1 – Контекстна діаграма в нотації IDEF0

При декомпозиції IDEF0 діаграми [12-14] web-додатку організації замовлення дизайну сайту було сформовано етапи формування та виконання замовлення по дизайну сайту. Дана декомпозиція представлена на рисунку 2.2.

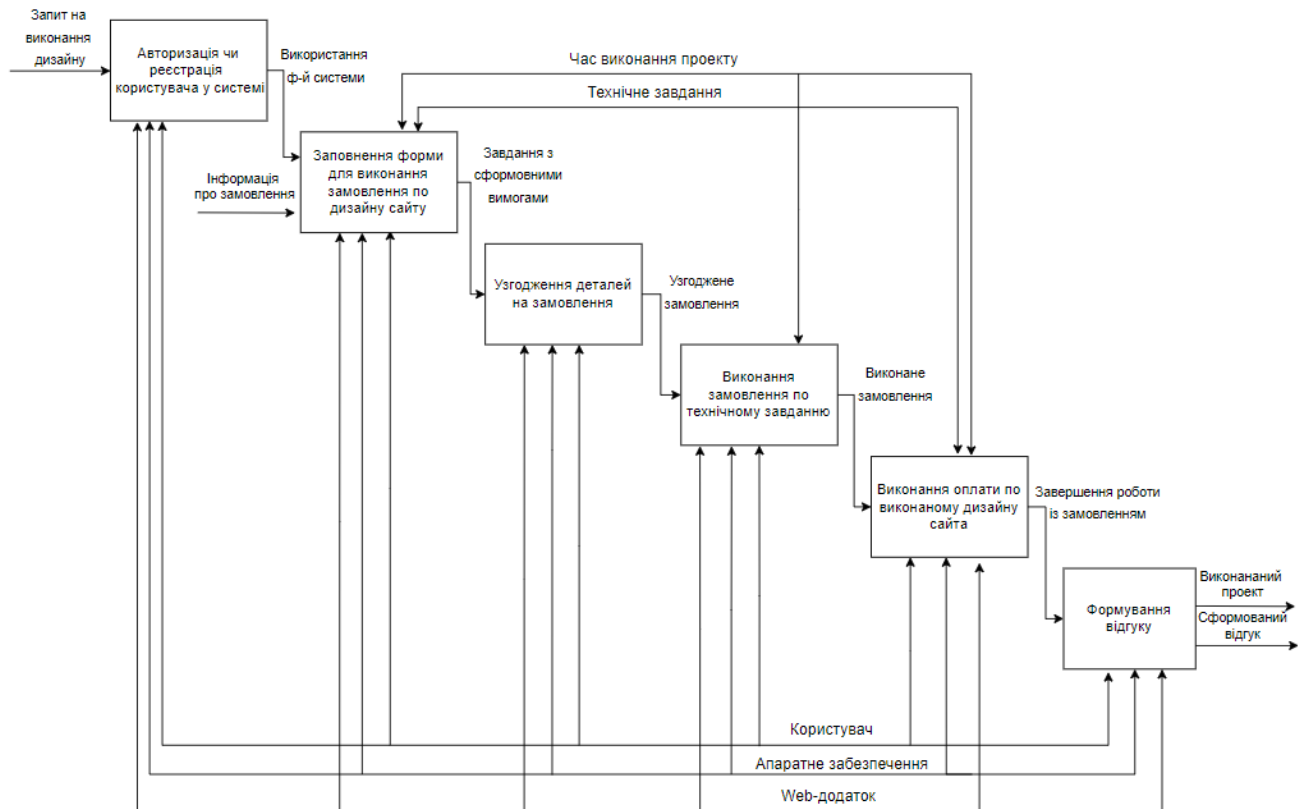


Рисунок 2.2 – Діаграма декомпозиції

Результатом декомпозиції є сформовані шість етапів. Результатом оформлення замовлення є формування відгуку та виконаний проект.

2.2. Моделювання варіантів використання web-додатку

Модель варіантів використання web-додатку організації замовлення дизайну сайту представляє з себе акторів та відповідний функціонал кожного із них [14-16].

Розглянемо акторів web-додатку (табл.2.1).

Табл. 2.1 – Опис акторів-користувачів web-додатку

Назва	Опис
Адміністратор	Користувач, що має розширений функціонал та може редагувати інформацію у web-додатку організації замовлення дизайну.
Замовник	Користувач, що може виконати замовлення у web-додатку.

Назва	Опис
MySQL	Зовнішній користувач web-додатку організації замовлення дизайну сайту, що використовується для збереження інформації та взаємодії з уже існуючою.

Табл. 2.2 – Опис варіантів використання web-додатку

Назва	Опис
Редагування інформації у web-додатку	Модуль, що дозволяє адміністратору редагувати інформацію у web-додатку.
Перегляд інформації у web-додатку	Модуль, що дозволяє будь-якому користувачеві переглядати інформацію у web-додатку.
Реєстрація	Модуль реєстрації користувача.
Авторизація	Модуль авторизації користувача.
Створення замовлення	Модуль, що дозволяє зареєстрованому користувачеві мати можливість виконати замовлення.
Заповнення форми для замовлення	Модуль, що дозволяє зареєстрованому користувачеві заповнити форму для створення замовлення.
Встановлення ціни на замовлення	Модуль, що дозволяє адміністратору встановити ціну за сформованим замовленням.
Оплата виконаного замовлення	Модуль, що дозволяє зареєстрованому користувачеві оплатити виконане замовлення.
Написання відгуку після співпраці	Модуль, що дозволяє користувачеві залишити відгук після співпраці.
Редагування даних на сторінці	Зареєстрований користувач може редагувати особисту інформацію.
Зміна паролю	Модуль, що дозволяє зареєстрованому користувачеві змінити пароль на новий.
Зміна ім'я користувача	Модуль, що дозволяє зареєстрованому користувачеві відредагувати особисту інформацію, таке як ім'я.

Назва	Опис
Додання особистого фото	Модуль, що дозволяє зареєстрованому користувачеві додати особисте фото.

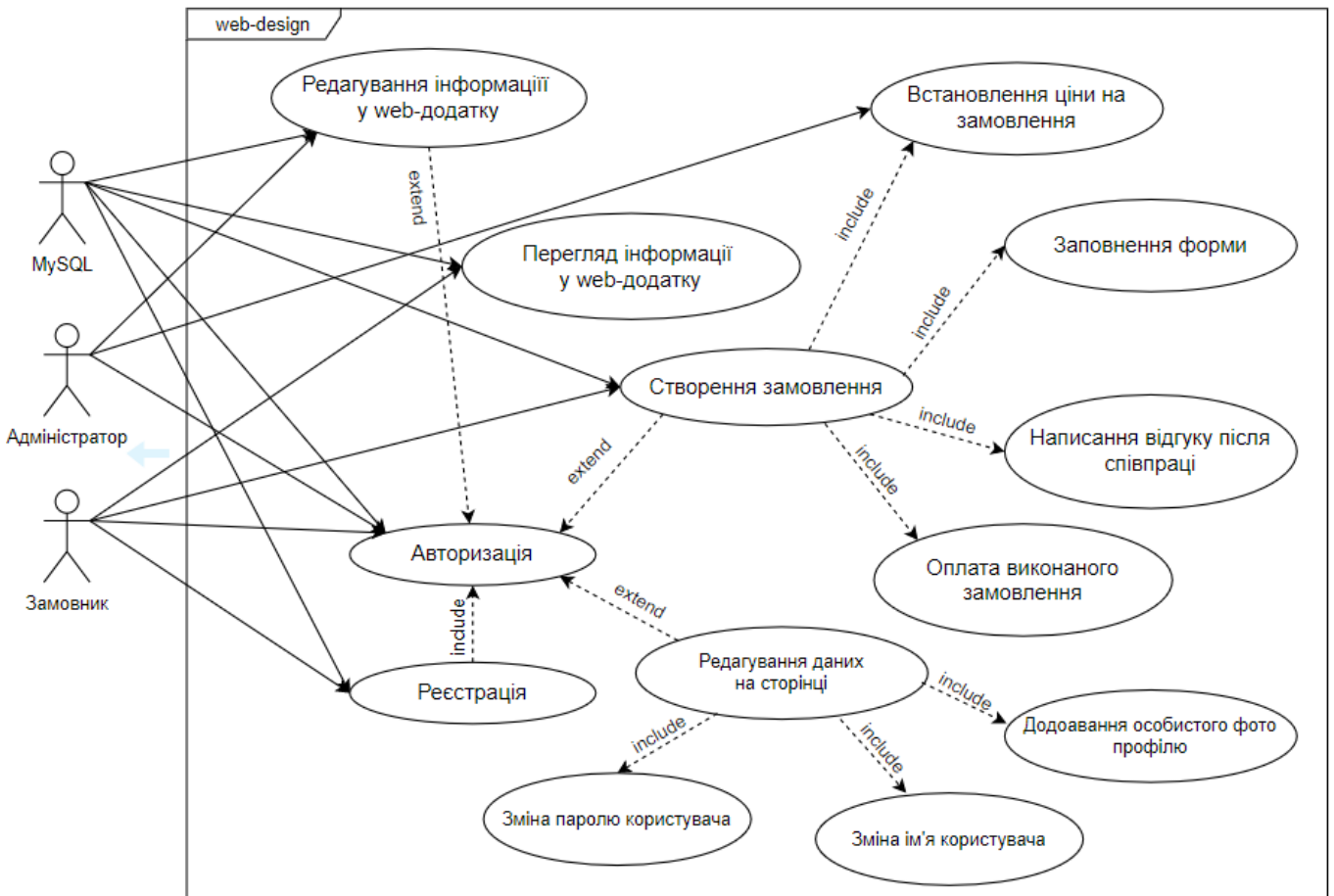


Рисунок 2.3 – Діаграма варіантів використання web-додатку організації замовлення дизайну сайту

Діаграма варіантів використання web-додатку організації замовлення дизайну сайту структурує функції в ієрархію, тобто коли один модуль залежить від іншого. Це дозволяє зрозуміти залежність модулів один від одного та від типу користувача.

2.3. Проектування бази даних

Для зберігання та повторного використання даних було спроектовано ER-діаграма базу даних web-додатку по замовленню дизайну сайту (рис.2.4)

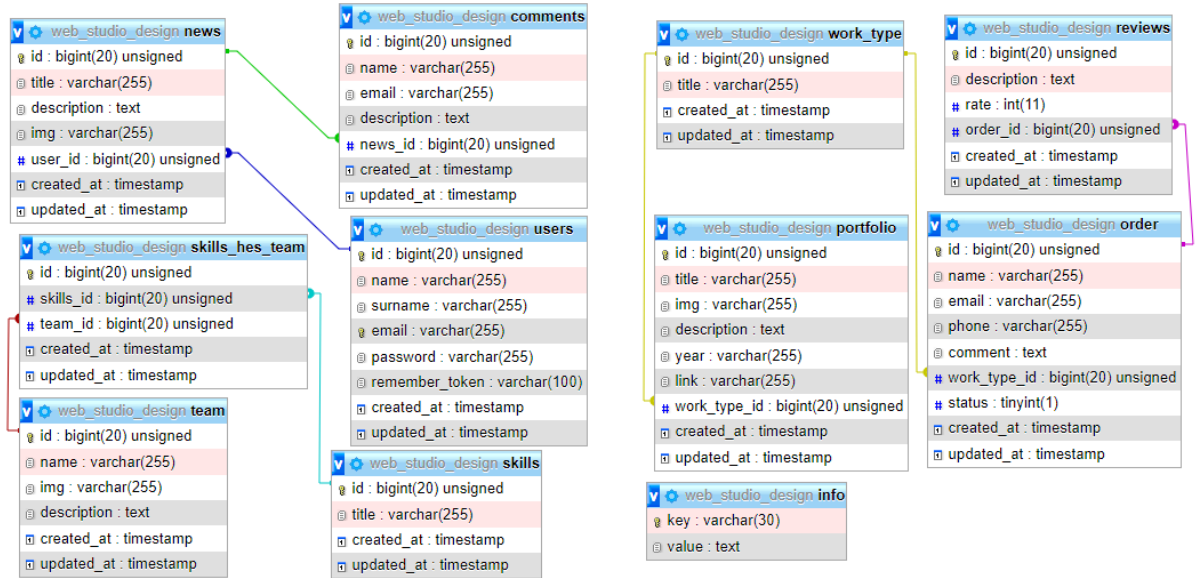


Рисунок 2.4 – ER-діаграма бази даних web-додатку організації замовлення дизайну сайту

Таблиця 3.5 – Опис таблиці ER-діаграми web-додатку організації замовлення дизайну сайту

№	Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
1	studio_team	id	Ідентифікаційний номер члена команди	BIGINT(20)	PK	NOT NULL
		name	Ім'я члена команди web-додатку	VARCHAR(255)		
		img	Фото члена команди web-додатку	VARCHAR(255)		
		description	Опис члена команди web-додатку	TEXT		
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Дата коли був відредагований допис	TIMESTAMP		
2	studio_skills	id	Ідентифікаційний номер навички	BIGINT(20)	PK	NOT NULL
		title	Назва навички	VARCHAR(255)		
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Дата коли був відредагований допис	TIMESTAMP		
3	studio_skill_has_team	id	Ідентифікаційний номер студії та її навичок	BIGINT(20)	PK	NOT NULL
		skills_id	Ідентифікаційний номер скілів студії	BIGINT(20)	FK	NOT NULL
		team_id	Ідентифікаційний номер команди	BIGINT(20)	FK	NOT NULL
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Дата коли був відредагований допис	TIMESTAMP		
4	studio_users	id	Ідентифікаційний номер	BIGINT(20)		
		name	Ім'я користувача	VARCHAR(255)		
		surname	Прізвище користувача	VARCHAR(255)		
		email	Електронна пошта користувача	VARCHAR(255)		

№	Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
		password	Пароль користувача	VARCHAR(255)		
		remember_token	Токен для входу користувача	VARCHAR(100)		
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Дата коли був відредагований допис	TIMESTAMP		
5	studio_news	id	Ідентифікаційний номер новини	BIGINT(20)	PK	NOT NULL
		title	Заголовок новини	VARCHAR(255)		
		description	Головна інформація	TEXT		
		img	Картинка новини	VARCHAR(255)		
		user_id	Ідентифікаційний користувача	BIGINT(20)	FK	NOT NULL
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Дата коли був відредагований допис	TIMESTAMP		
6	studio_comments	id	Ідентифікаційний номер коментарю	BIGINT(20)	PK	NOT NULL
		name	Ім'я користувача	VARCHAR(255)		
		email	Електронна адреса користувача	VARCHAR(255)		
		description	Коментар користувача	TEXT		
		news_id	Ідентифікаційний номер новини	BIGINT(20)	FK	NOT NULL
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Дата коли був відредагований допис	TIMESTAMP		
7	studio_work_type	id	Ідентифікаційний номер типу роботи	BIGINT(20)	PK	NOT NULL
		title	Назва типу роботи	VARCHAR(255)		
		create_at	Дата коли був створений допис	TIMESTAMP		

№	Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
		update_at	Дата коли був відредагований допис	TIMESTAMP		
8	studio_portfolio	id	Ідентифікаційний номер портфоліо	BIGINT(20)	PK	NOT NULL
		title	Заголовок роботи з портфоліо	VARCHAR(255)		
		img	Картинка роботи з портфоліо	VARCHAR(255)		
		description	Додатковий опис роботи з портфоліо	TEXT		
		year	Рік виконання	VARCHAR(255)		
		link	Посилання на роботу	VARCHAR(255)		
		work_type_id	Ідентифікаційний номер типу роботи	BIGINT(20)	FK	NOT NULL
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Дата коли був відредагований допис	TIMESTAMP		
9	studio_order	id	Ідентифікаційний номер замовлення	BIGINT(20)	PK	NOT NULL
		name	Назва замовлення	VARCHAR(255)		
		email	Електронна адреса замовлення	VARCHAR(255)		
		phone	Номер телефону замовлення	VARCHAR(255)		
		comment	Додатковий коментарій замовлення	TEXT		
		work_type_id	Ідентифікаційний номер типу роботи	BIGINT(20)	FK	NOT NULL
		status	Статус замовлення	TINYINT(1)		
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Зміст	TIMESTAMP		

№	Таблиця	Поле	Зміст	Тип	Ключі	Обмеження
10	studio_reviews	id	Ідентифікаційний номер члена команди	BIGINT(20)	PK	NOT NULL
		description	Ім'я члена команди web-додатку	TEXT		
		rate	Фото члена команди web-додатку	INT(11)		
		order_id	Опис члена команди web-додатку	BIGINT(20)	FK	NOT NULL
		create_at	Дата коли був створений допис	TIMESTAMP		
		update_at	Дата коли був відредагований допис	TIMESTAMP		
11	studio_info	key	Ідентифікаційний номер навички	VARCHAR(30)		
		value	Назва навички	TEXT		

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ WEB-ДОДАТКУ

3.1 Архітектура web-додатку

Архітектура web-додатку організації замовлення дизайну сайту представляє з себе реалізацію Model-View-Controller [17]. Дана архітектура використовується для розділення реалізації використання функціоналу та представлення (рис.3.1).

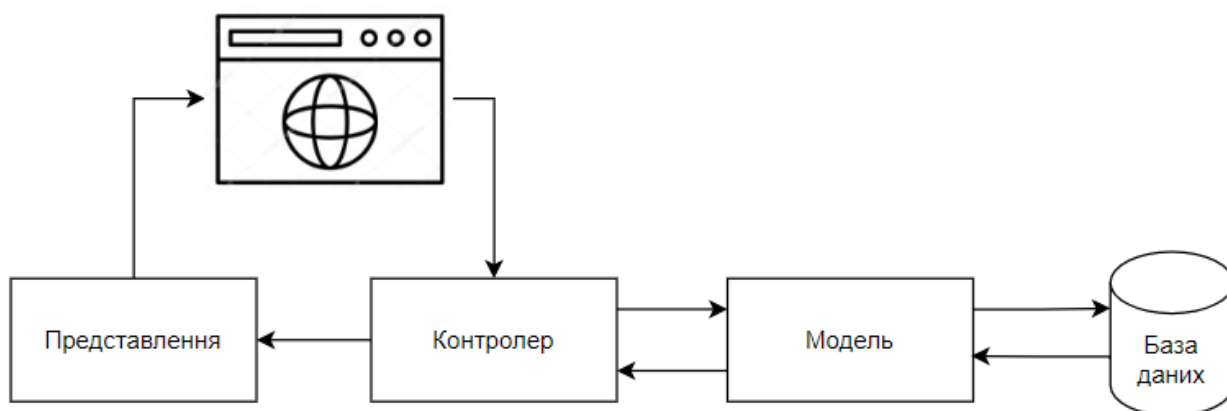


Рисунок 3.1 – Архітектура web-додатку організації замовлення дизайну сайту

Розглянемо кожен із об'єктів архітектури web-додатку:

1. Модель є об'єктом web-додатку організації замовлення дизайну сайту, що несе дані. Він також може мати логіку оновлення контролера, якщо його дані змінюються.
2. Представлення є візуалізацією даних web-додатку, що містяться в моделі та відповідно у базі даних.
3. Контролер взаємодіє як з моделлю, так з представленням. Він контролює потік від моделі до представлення та навпаки. Він поділяє у web-додатку уявлення та модель.

3.2 Програмна реалізація

Програмна реалізація web-додатку організації замовлення дизайну сайту реалізовувалася із використанням Laravel та Vue.js. Пакети, що були створені та використані, представлені на рис.3.2.

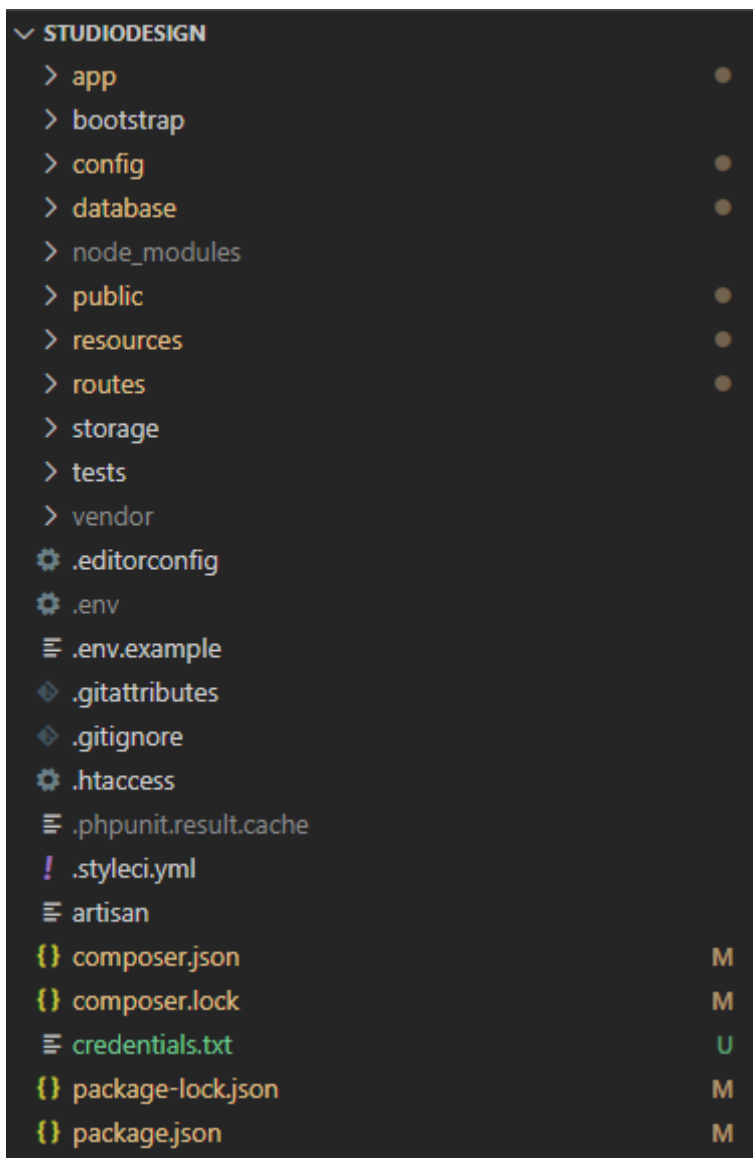


Рисунок 3.2 – Структура web-додатку організації замовлення дизайну сайту

Також було розроблено базу даних web-додатку [18-19]. Для цього використовувався популярний інструмент для роботи із MySQL – phpMyAdmin. Розглянемо створені таблиці на рис.3.3-14.

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	key	varchar(30)	utf8mb4_unicode_ci		Нет	Нет		
<input type="checkbox"/>	2	value	text	utf8mb4_unicode_ci		Нет	Нет		

Рисунок 3.3 – Таблица «studio_info»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id	bigint(20)	UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2	title	varchar(255)	utf8mb4_unicode_ci		Нет	Нет		
<input type="checkbox"/>	3	description	text	utf8mb4_unicode_ci		Нет	Нет		
<input type="checkbox"/>	4	img	varchar(255)	utf8mb4_unicode_ci		Нет	Нет		
<input type="checkbox"/>	5	user_id	bigint(20)	UNSIGNED	Нет	Нет			
<input type="checkbox"/>	6	created_at	timestamp		Да	NULL			
<input type="checkbox"/>	7	updated_at	timestamp		Да	NULL			

Рисунок 3.4 – Таблица «studio_news»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id	bigint(20)	UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2	data	text	utf8mb4_unicode_ci		Нет	Нет		
<input type="checkbox"/>	3	file	varchar(255)	utf8mb4_unicode_ci	Да	NULL			
<input type="checkbox"/>	4	user_id	bigint(20)	UNSIGNED	Нет	Нет			
<input type="checkbox"/>	5	status	tinyint(1)		Нет	0			
<input type="checkbox"/>	6	cost	varchar(255)	utf8mb4_unicode_ci	Да	NULL			
<input type="checkbox"/>	7	pay	tinyint(4)		Нет	0			
<input type="checkbox"/>	8	created_at	timestamp		Да	NULL			
<input type="checkbox"/>	9	updated_at	timestamp		Да	NULL			

Рисунок 3.5 – Таблица «studio_order»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 id 🔑	bigint(20)		UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2 title	varchar(255)	utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	3 img	varchar(255)	utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	4 description	text	utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	5 year	varchar(255)	utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	6 link	varchar(255)	utf8mb4_unicode_ci		Да	NULL			
<input type="checkbox"/>	7 work_type_id 🔑	bigint(20)		UNSIGNED	Нет	Нет			
<input type="checkbox"/>	8 created_at	timestamp			Да	NULL			
<input type="checkbox"/>	9 updated_at	timestamp			Да	NULL			

Рисунок 3.6 – Таблица «studio_portfolio»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 id 🔑	bigint(20)		UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2 description	text	utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	3 rate	int(11)			Нет	Нет			
<input type="checkbox"/>	4 order_id 🔑	bigint(20)		UNSIGNED	Нет	Нет			
<input type="checkbox"/>	5 created_at	timestamp			Да	NULL			
<input type="checkbox"/>	6 updated_at	timestamp			Да	NULL			

Рисунок 3.7 – Таблица «studio_reviews»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 id 🔑	bigint(20)		UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2 title	varchar(255)	utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	3 created_at	timestamp			Да	NULL			
<input type="checkbox"/>	4 updated_at	timestamp			Да	NULL			

Рисунок 3.8 – Таблица «studio_skills»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 id 🔑	bigint(20)		UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2 skills_id 🔑	bigint(20)		UNSIGNED	Нет	Нет			
<input type="checkbox"/>	3 team_id 🔑	bigint(20)		UNSIGNED	Нет	Нет			
<input type="checkbox"/>	4 created_at	timestamp			Да	NULL			
<input type="checkbox"/>	5 updated_at	timestamp			Да	NULL			

Рисунок 3.9 – Таблица «studio_skills_hes_team»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id	bigint(20)	UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2	name	varchar(255) utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	3	img	varchar(255) utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	4	description	text utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	5	created_at	timestamp		Да	NULL			
<input type="checkbox"/>	6	updated_at	timestamp		Да	NULL			

Рисунок 3.10 – Таблица «studio_team»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id	bigint(20)	UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2	name	varchar(255) utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	3	phone	varchar(255) utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	4	email	varchar(255) utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	5	password	varchar(255) utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	6	is_admin	tinyint(1)		Нет	0			
<input type="checkbox"/>	7	photo	varchar(255) utf8mb4_unicode_ci		Нет	/img/no-avatar.png			
<input type="checkbox"/>	8	photo_small	varchar(255) utf8mb4_unicode_ci		Нет	/img/no-avatar.png			
<input type="checkbox"/>	9	remember_token	varchar(100) utf8mb4_unicode_ci		Да	NULL			
<input type="checkbox"/>	10	created_at	timestamp		Да	NULL			
<input type="checkbox"/>	11	updated_at	timestamp		Да	NULL			

Рисунок 3.11 – Таблица «studio_users»

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1	id	bigint(20)	UNSIGNED	Нет	Нет		AUTO_INCREMENT	
<input type="checkbox"/>	2	title	varchar(255) utf8mb4_unicode_ci		Нет	Нет			
<input type="checkbox"/>	3	created_at	timestamp		Да	NULL			
<input type="checkbox"/>	4	updated_at	timestamp		Да	NULL			

Рисунок 3.12 – Таблица «studio_work_type»

При додаванні додаток на хостинг було виконано налаштування, що представлено на рис.3.13-14.

ГЛАВНАЯ · ХОСТИНГ · ДОМЕНЫ · СЕРВЕРЫ VPS · ВЫД. СЕРВЕРЫ · ВЕБ-САЙТЫ · БЛОГ · КОНТАКТ

Панель Почта FTP MySQL Форум Помощь

Панель

Обзор

Обзор для PRO #1

Статус ✔ Активный Использование физического дискового пространства

Пакет услуг **Pro** (Срок действия заканчивается: 2023-06-03) +

Ссылка на хостинг ZZZ.com.ua **Нет**

Количество бесплатных доменов **1 / 2 (50%)** +

Домены **studio-design.zzz.com.ua**


IP адрес вашего сервера **95.211.16.67**

Аккаунты электронной почты **1 / 2 (50%)** +

Аккаунты FTP **0 / 2 (0%)** +

Пользователи MySQL **1 / 1 (100%)** +

Базы данных MySQL **1 / 1 (100%)** +



Использованное дисковое пространство

Использованное количество файлов

Рисунок 3.13 – Домены

ГЛАВНАЯ · ХОСТИНГ · ДОМЕНЫ · СЕРВЕРЫ VPS · ВЫД. СЕРВЕРЫ · ВЕБ-САЙТЫ · БЛОГ · КОНТАКТ

Панель Почта FTP MySQL Форум Помощь

Панель

Домены для PRO #1 + Добавить

Домен	Аккаунт	RHP	Срок действия домена истекает	Срок действия SSL истекает	Статус	Действия
studio-design.zzz.com.ua	PRO #1	7.4	-	-	✔ Активный	✔ SSL + Субдомены ✎ Редактировать ✖ Удалить

Frequently Asked Questions

- How to renew a domain?
- How to configure the domain's DNS settings?
- How to add or renew an SSL certificate?
- How to force using an SSL connection on a website?
- Free SSL certificates for hosting plan VIP
- How to transfer a domain to ZZZ.com.ua?
- How to recover expired domains?

Рисунок 3.14 – Огляд акаунта

Отже, у результаті роботи було розроблено web-додаток організації замовлення дизайну сайту. Даний web-додаток має вірну структуру при використанні Laravel та Vue.js, нормалізовану та структуровану базу даних. Крім того, web-додаток був перенесений на хостинг.

3.3 Використання web-додатку зі сторони користувача

Переходячи за посиланням до web-додатку організації замовлення дизайну сайту, а саме «studio-design.zzz.com.ua», користувач направляється на головну сторінку. Головна сторінка є першим пунктом навігаційного меню web-додатку. Навігаційне меню складається з:

- «Головна сторінка»;
- «Портфоліо»;
- «Статті»;
- «Відгуки»;
- «Команда»;
- «Контакти»;
- «Оформлення замовлення».

Перший блок на головній сторінці web-додатку є вступним, тому містить в собі лише заголовок, рекламний слоган та посилання для виконання замовлення користувачем (рис.3.15).

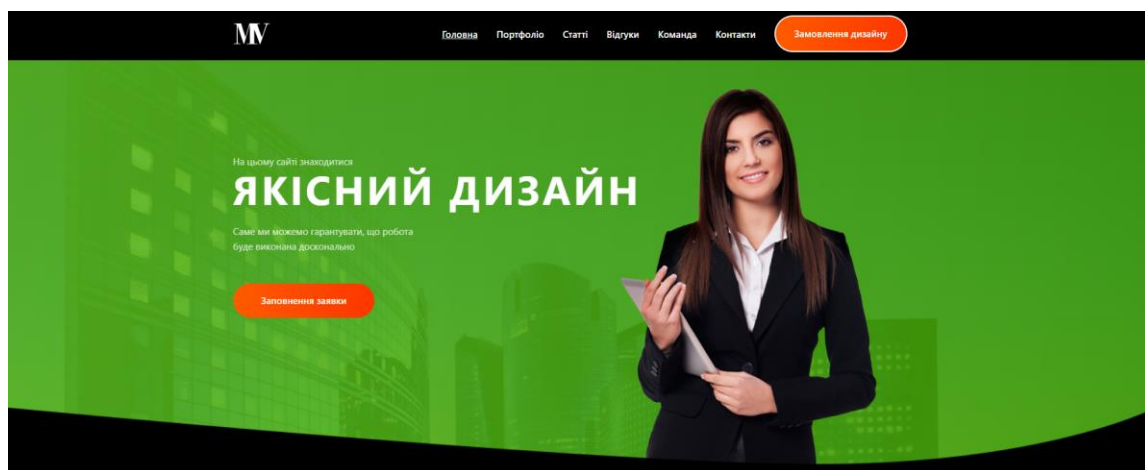


Рисунок 3.15 – Головна сторінка web-додатку, блок із вступною рекламою

За допомогою другого блоку на головній сторінці користувач може ознайомитися з додатковою інформацією про організацію по замовленню дизайну (рис.3.16).



Рисунок 3.16 – Головна сторінка web-додатку, блок із додатковою інформацією

Третій блок головної сторінки web-додатку відображає останні додані приклади дизайнів-замовлень (рис.3.17).

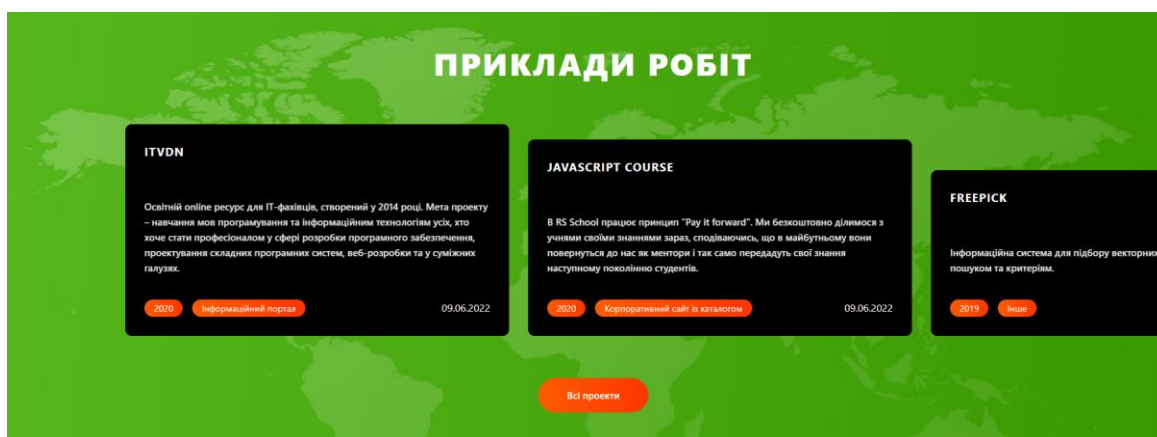


Рисунок 3.17 – Головна сторінка web-додатку, блок із вибіркою дизайнів-замовлень

Останній блок головної сторінки web-додатку несе в собі коротку інформацію про етапи розробку дизайну сайтів чи інших інформаційних систем (рис.3.18).

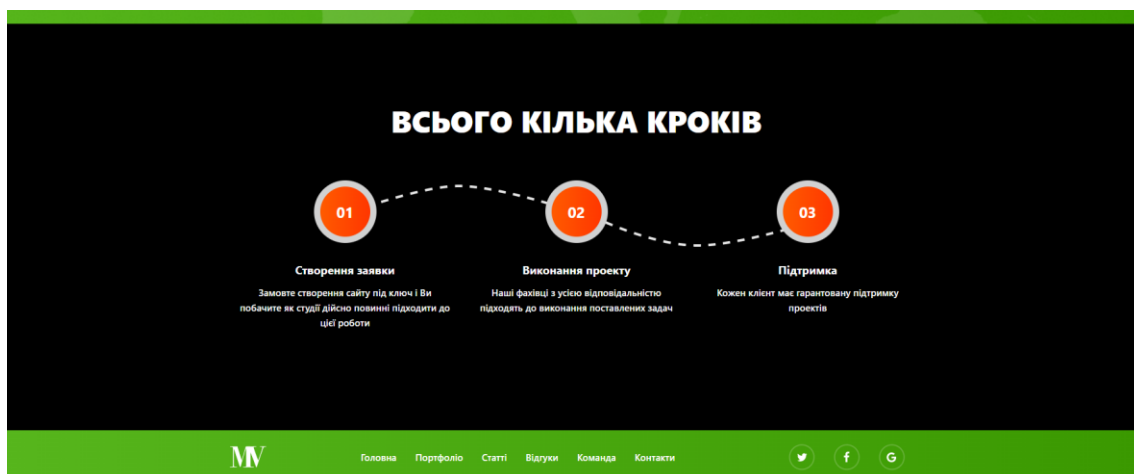


Рисунок 3.18 – Головна сторінка web-додатку, блок короткою інформацією про етапи

Наступним пунктом навігаційного меню є «Портфоліо». Дана сторінка дозволяє користувачу ознайомитися із різними проєктами web-додатку організації замовлення дизайну сайту (рис.3.19).

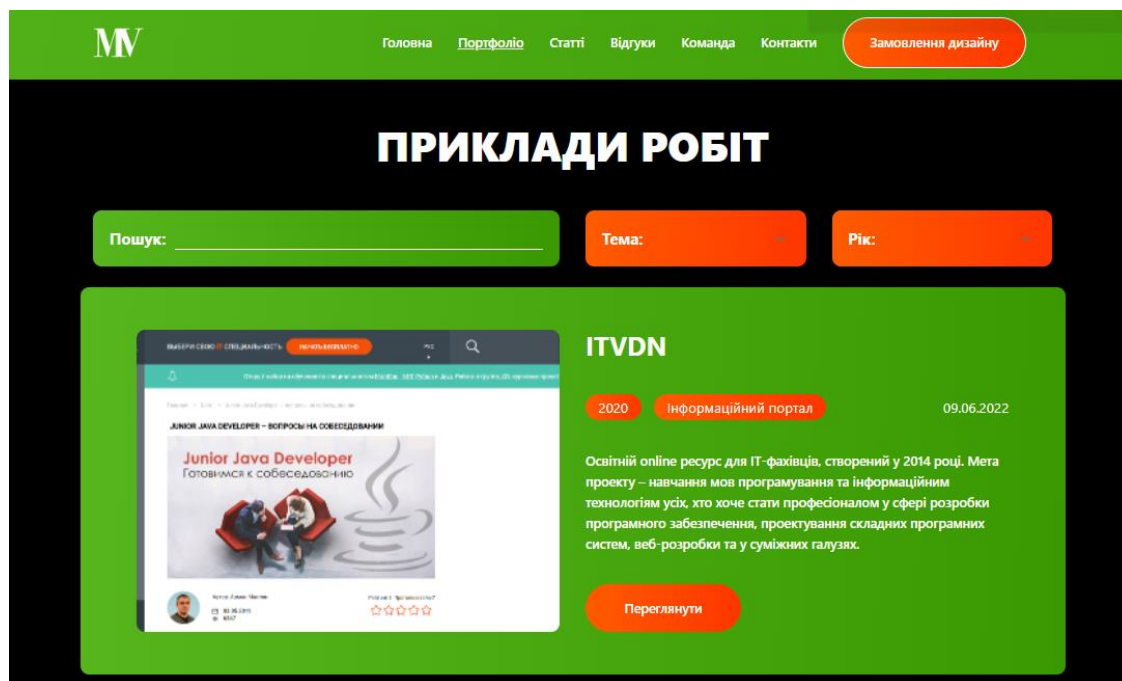


Рисунок 3.19 – Сторінка портфоліо web-додатку

Крім того, користувач web-додатку має можливість виконувати вибірковий пошук прикладу роботи за темою, роком чи навіть назвою (рис.3.20).

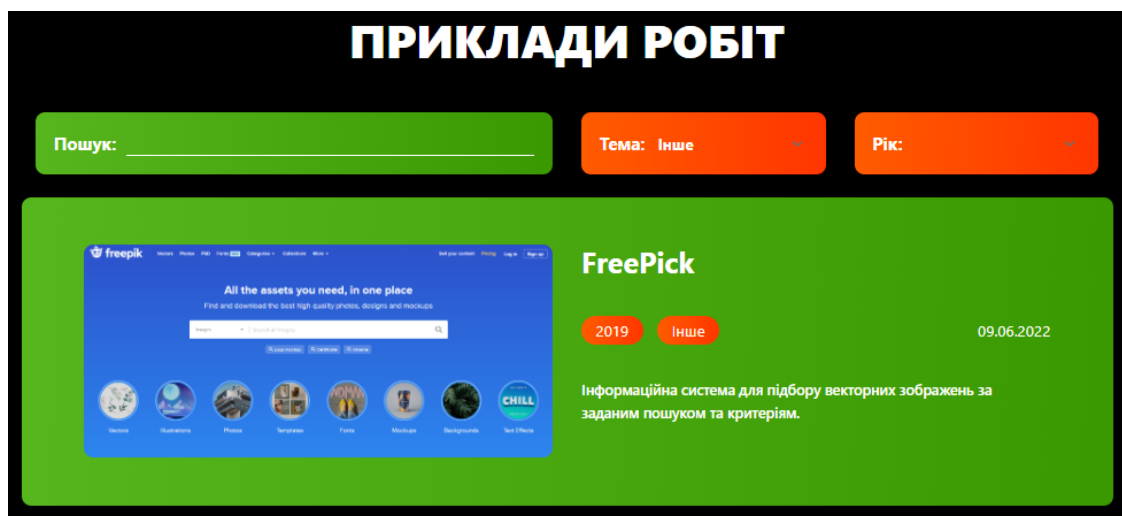


Рисунок 3.20 – Сторінка портфоліо web-додатку з фільтрацією за тематикою роботи

Якщо при виконанні вибіркового пошуку не було знайдено відповідних прикладів, то користувач отримує повідомлення на сторінці. Дане повідомлення говорить, що інформація за даними параметрами відсутня (рис.3.21).

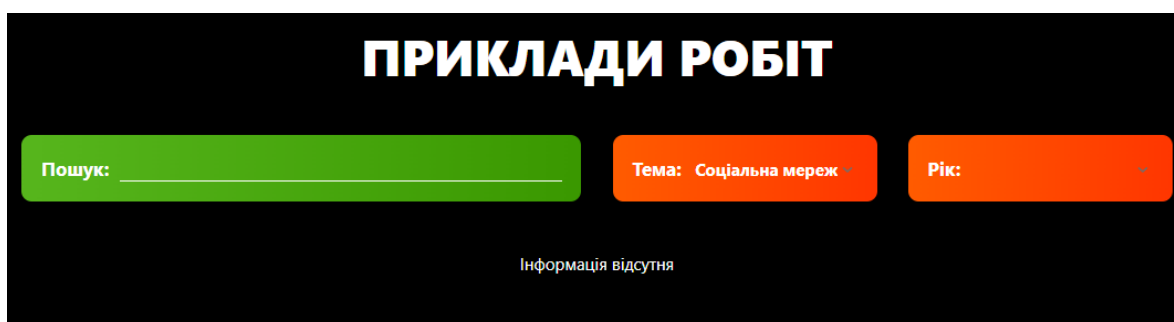


Рисунок 3.21 – Сторінка портфоліо web-додатку з фільтрацією за іншою тематикою

Наступним пунктом навігаційного меню є «Статті». На даній сторінці web-додатку користувач має можливість ознайомитися із останніми статтями та новинами (рис.3.22).

Натиснувши на одну із статей, користувач перенаправляється на сторінку із повною інформацією (рис.3.23).

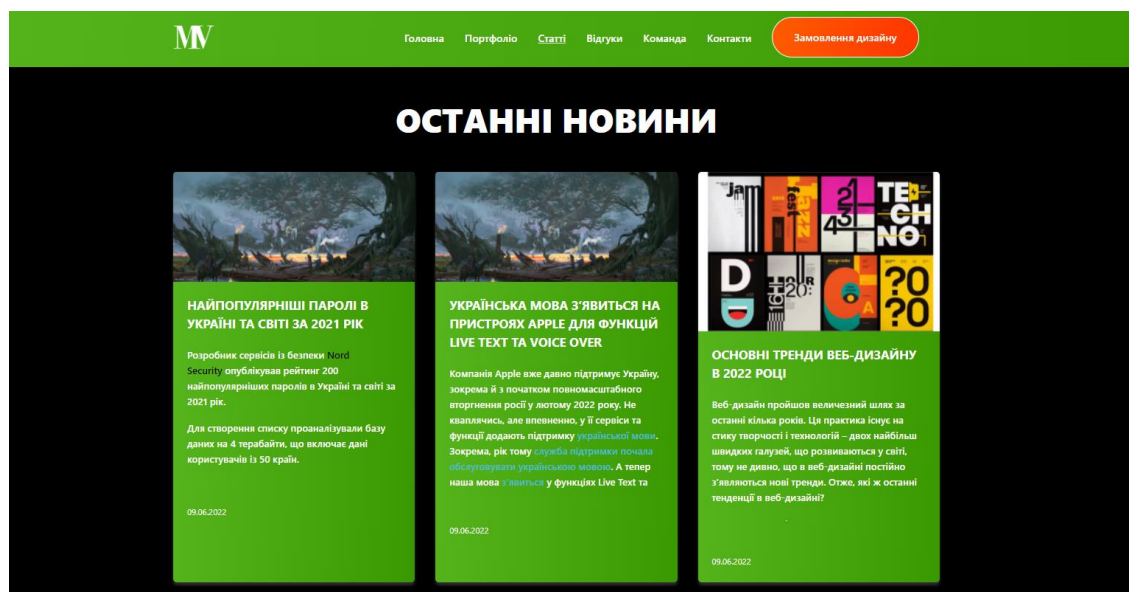


Рисунок 3.22 – Сторінка із новинами web-додатку



Рисунок 3.23 – Сторінка із детальною інформацією обраної новини

Під обраною статтею користувач web-додатку організації замовлення дизайну сайту має можливість додавання коментарю (рис.3.24). Для того що залишити коментар, користувачеві потрібно натиснути на кнопку розгортання блоку (рис.3.25).

Для публікації коментарю користувач web-додатку повинен ввести:

- прізвище, ім'я;
- електронна адреса користувача;
- текст коментарю.



Рисунок 3.24 – Можливість додавати коментар до новини

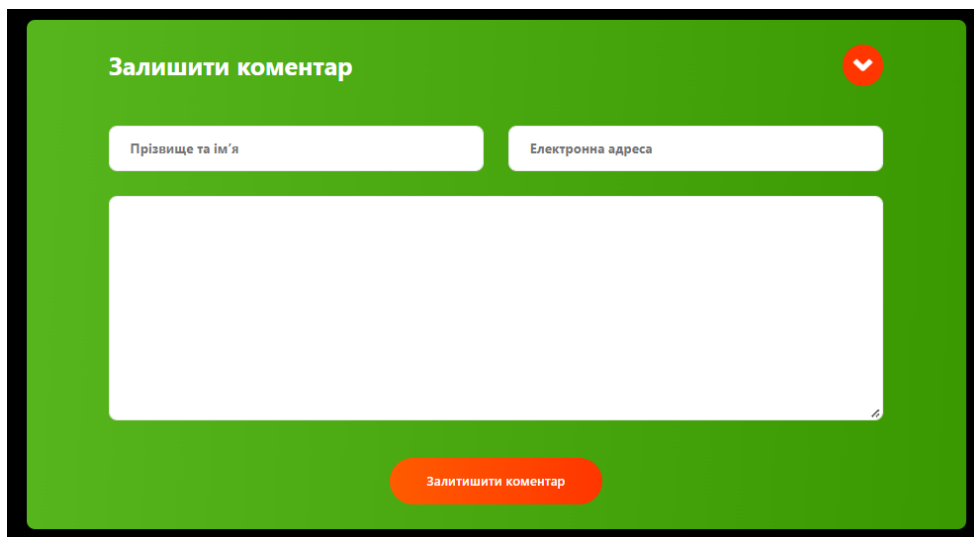
A green rectangular form with rounded corners. At the top left, it says 'Залишити коментар' in white. At the top right, there is a red circular icon with a white downward-pointing arrow. Below the title, there are two white input fields: the first is labeled 'Прізвище та ім'я' and the second is labeled 'Електронна адреса'. Below these fields is a large white text area for the comment. At the bottom center, there is a red rounded button with the text 'Залишити коментар' in white.

Рисунок 3.25 – Форма для додавання коментарю для новини

Наступним пунктом навігаційного меню web-додатку організації замовлення дизайну сайту є «Відгуки». На даній сторінці користувач має можливість переглянути відгуки до виконаних робіт організації (рис.3.26).

Також була додана фільтрація за пунктами, що відповідають представленим роботам у порфтоліо організації замовлення дизайну (рис.3.27).

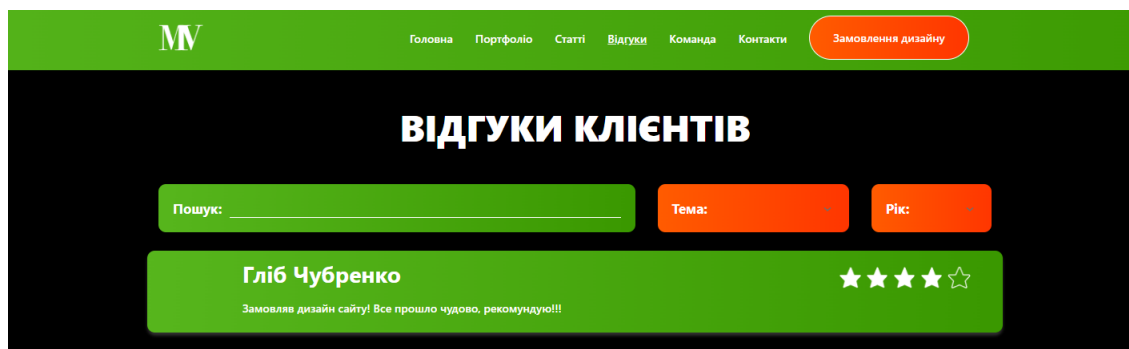


Рисунок 3.26 – Сторінка із відгуками робіт web-додатку

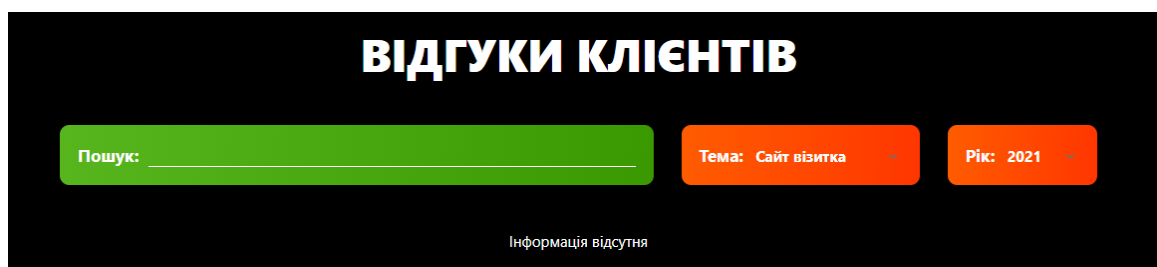


Рисунок 3.27 – Сторінка із відгуками робіт web-додатку, приклад фільтрації за темою роботи та датою

Наступним пунктом навігаційного меню web-додатку є «Команда». На даній сторінці знаходиться коротка інформація про кожного із членів команди організації замовлення дизайну сайтів (рис.3.28).

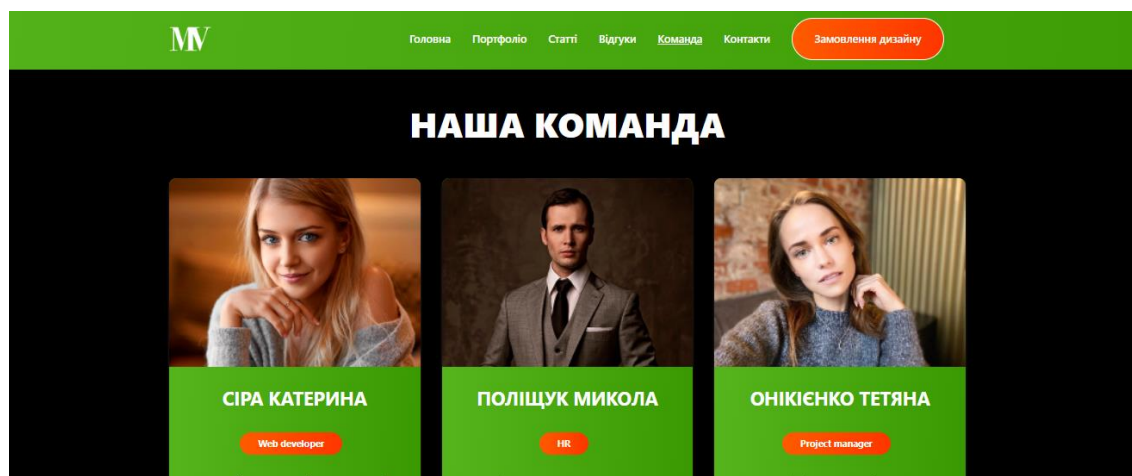


Рисунок 3.28 – Сторінка web-додатку із інформацією про команду організації web-додатку

Наступним пунктом навігаційного меню web-додатку є «Контакти». Дана сторінка є базовою по надання інформації про контактну інформацію організації. Такими даними є контактний номер телефону та організаційна електронна пошта (рис.3.29).

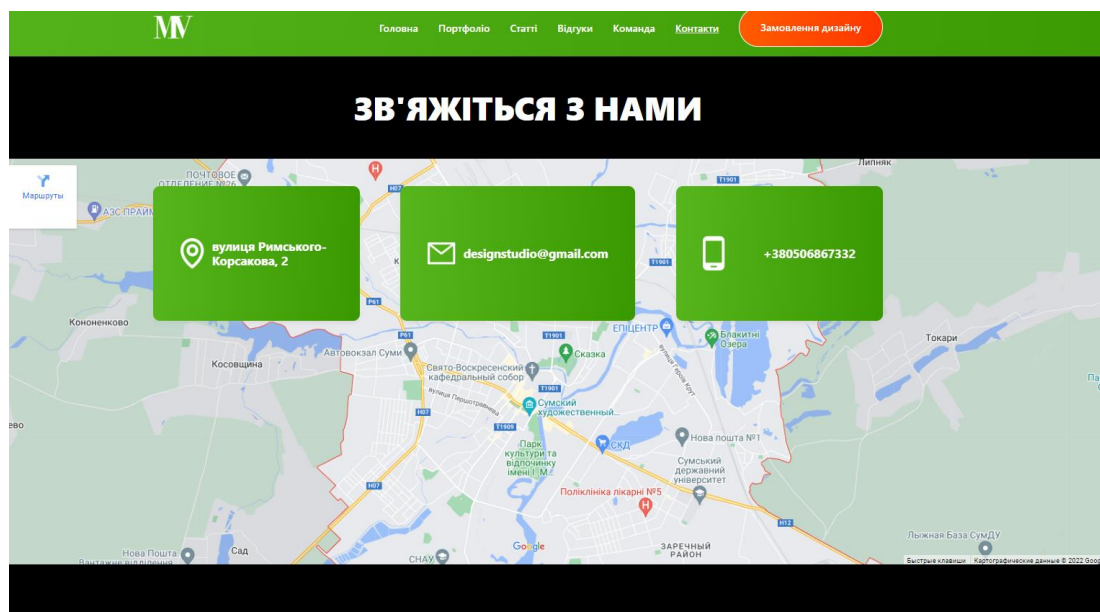


Рисунок 3.29 – Сторінка web-додатку із контактною інформацією команди web-додатку

Останнім, але самим головним, пунктом меню web-додатку є «Замовлення дизайну». Власне оформлення замовлення складається із декількох кроків:

1. авторизація чи реєстрація;
2. замовлення форми;
3. відображення інформаційного блоку.

На першому кроці, а саме авторизації чи реєстрації, користувач повинен заповнити основну інформацію для себе (рис.3.30-31). Дана інформація потрібна для подальшої взаємодії користувача із адміністратором web-додатку організації замовлення дизайну сайту.

Рисунок 3.30 – Сторінка web-додатку із оформленням замовленням, перший крок (реєстрація)

Рисунок 3.31 – Сторінка web-додатку із оформленням замовлення, перший крок (авторизація)

Другим кроком оформлення замовлення є заповнення форми. Користувач повинен заповнити інформацію про те, яке бачення він має про власний сайт (рис.3.32-33).

01 Крок 1
Ресстрація

02 Крок 2
Заповнення форми

03 Крок 3
Готово!

Ймовірна назва сайту

Ймовірний тип сайту

Сайт візитка
 Сайт-портфоліо
 Блог
 Соціальна мережа

Корпоративний сайт із каталогом
 Корпоративний без каталогу
 Інтернет магазин

Інформаційний портал
 Лендинг
 Інше

Рисунок 3.32 – Сторінка web-додатку із оформленням замовлення, другий крок

На таких запитаннях, як «настрій та асоціації дизайну» користувач має можливість вибрати декілька варіантів, за побажанням.

Настрій та асоціації, які має викликати дизайн

Суворий корпоративний дизайн
 Яскравий, помітний дизайн
 Позитивний та веселий

Дизайн насичений ілюстраціями/фотографіями
 Мінімалістичний дизайн. Упор на функціональність

Вимоги до зовнішнього вигляду (дизайну) сайту

- У нас є чітке бачення, яким має бути дизайн нашого сайту.
- Ми не маємо чіткого бачення дизайну, ми просто хочемо, щоб дизайн продовжував традиції корпоративного стилю нашої компанії.
- Ми повністю довіряємо смаку Ваших фахівців, які з портфоліо ознайомлені.
- Дизайн сайту може бути розроблений на розсуд розробників, але існують обов'язкові вимоги.

Приклади вдалих сайтів, кольорів, розташування елементів (не менше 3-х)

- <https://webdevandseo.com.ua/the-main-trends-of-web-design-in-2021/>
 - <https://enguideo.pro/ru/films/howls-moving-castle-2004/#>

Файли (Використовувати якісь конкретні графічні зображення надані вами?)

Документація.docx

Готово!

Рисунок 3.33 – Сторінка із оформленням замовлення, другий крок (детальна інформація)

Останній крок відображає інформацію про результат оформлення замовлення (рис.3.34).

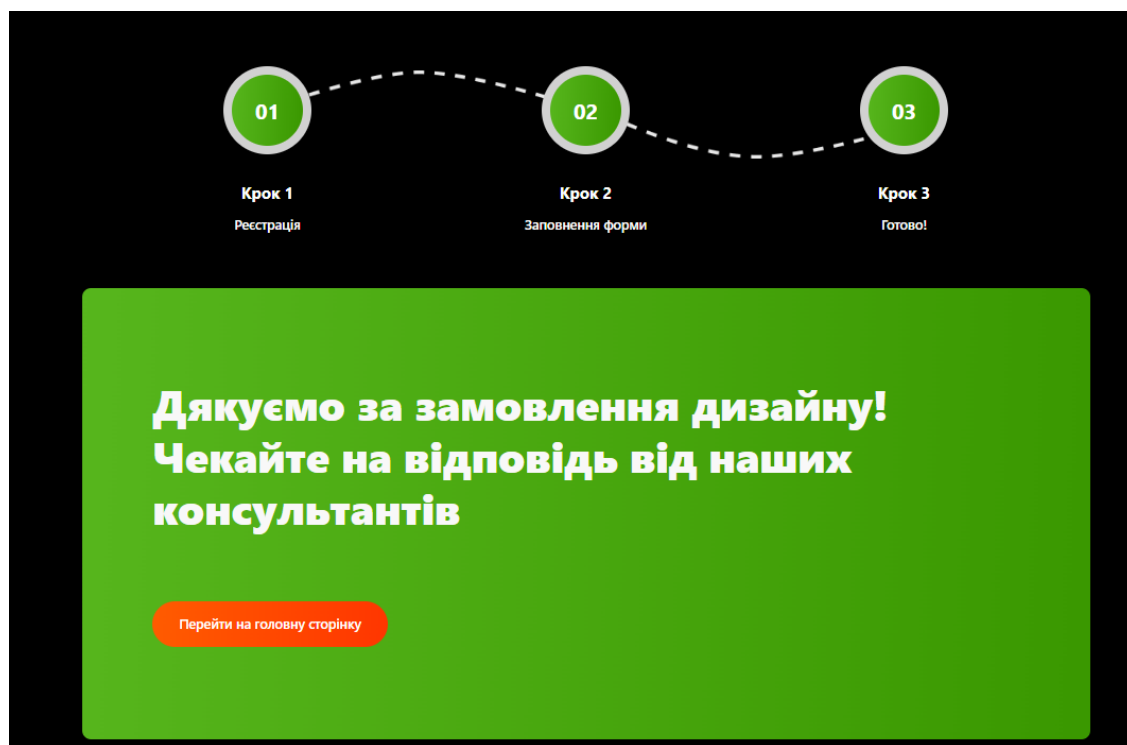


Рисунок 3.34 – Сторінка web-додатку із оформленням замовлення, третій крок

Після того, як користувач виконав авторизацію чи реєстрацію, відкривається новий функціонал для оформлення замовлення. Тепер користувач має можливість переглядати власні замовлення, налаштування та виходити із аккаунту у web-додатку (рис.3.35).

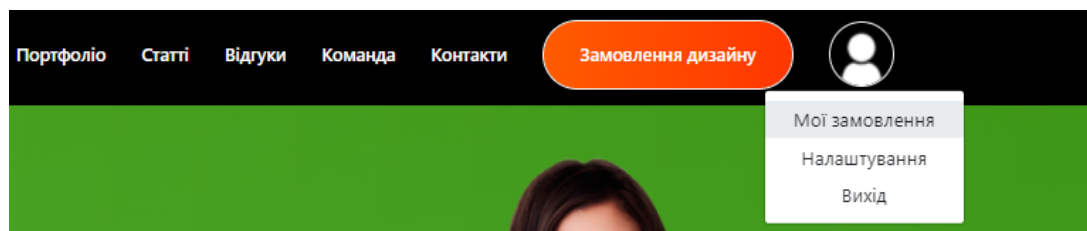


Рисунок 3.35 – Меню користувача для переходу на замовлення

Розглянемо першу сторінку, а саме «Мої замовлення». Користувач має можливість переглядати власні замовлення та виконувати пошук (рис.3.36).

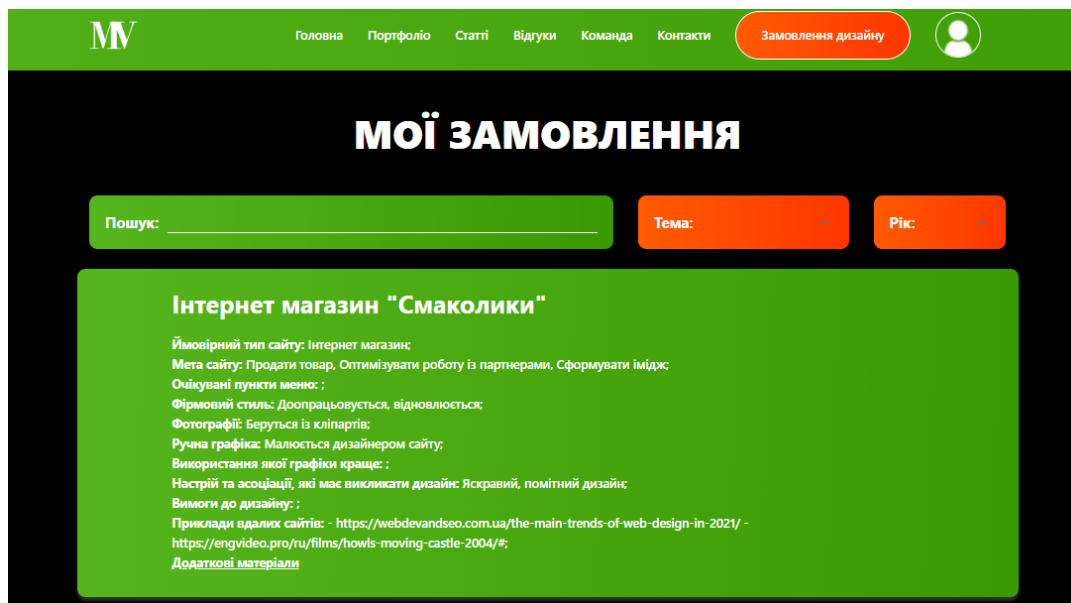


Рисунок 3.36 – Сторінка із замовленнями користувача

Після встановлення адміністратором ціни та виконання замовлення користувач має можливість виконати оплату замовлення. При натисканні на кнопку «Виконати оплату» у web-додатку організації замовленню дизайнів сайту користувач перенаправляється на нову сторінку (рис.3.37-39).

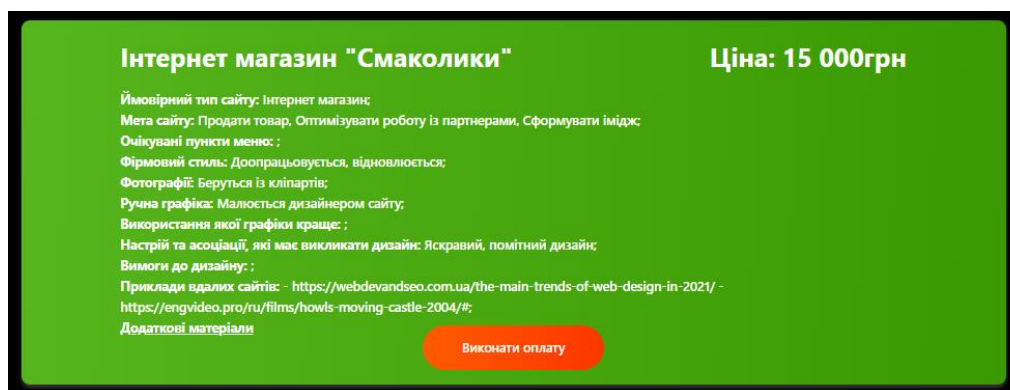


Рисунок 3.37 – Сторінка із замовленнями користувача, можливість виконати оплату

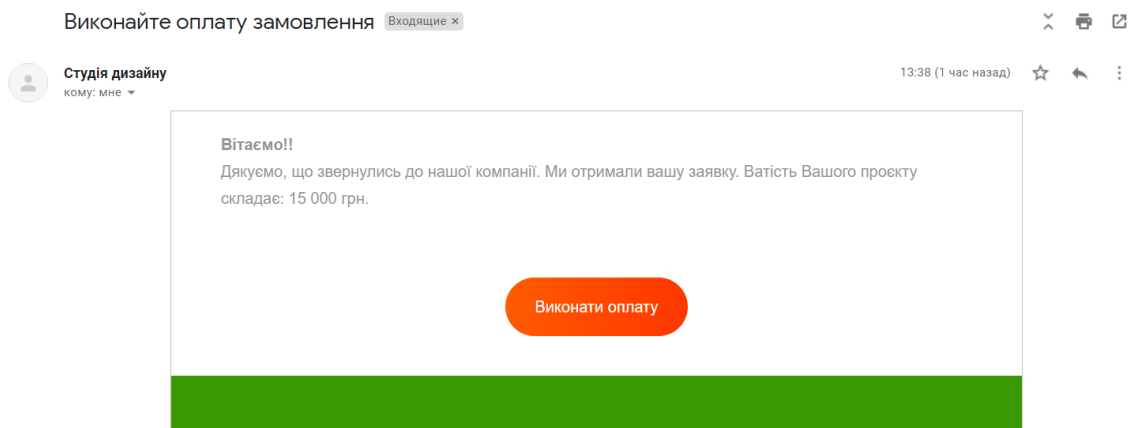


Рисунок 3.38 – Повідомлення на електронній пошті, нагадування про виконання оплати

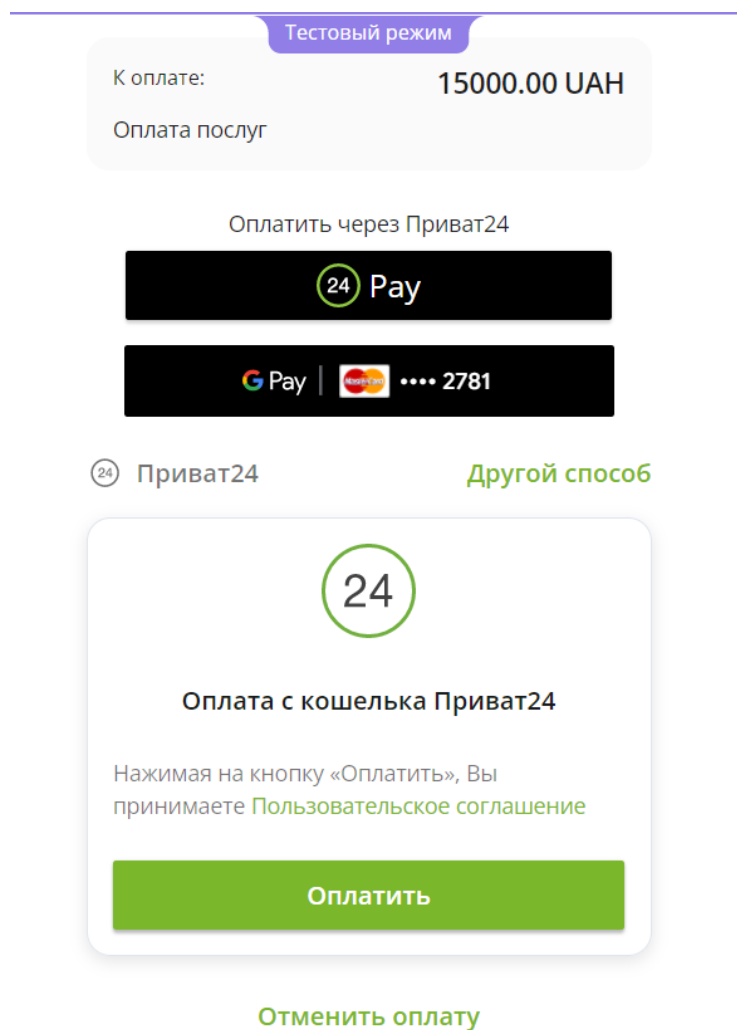


Рисунок 3.39 – Сторінка для оплати виконаного замовлення

Після оплати користувач отримує повідомлення, на вказану раніше електронну адресу, із можливістю залишення відгуку (рис.3.40-41)

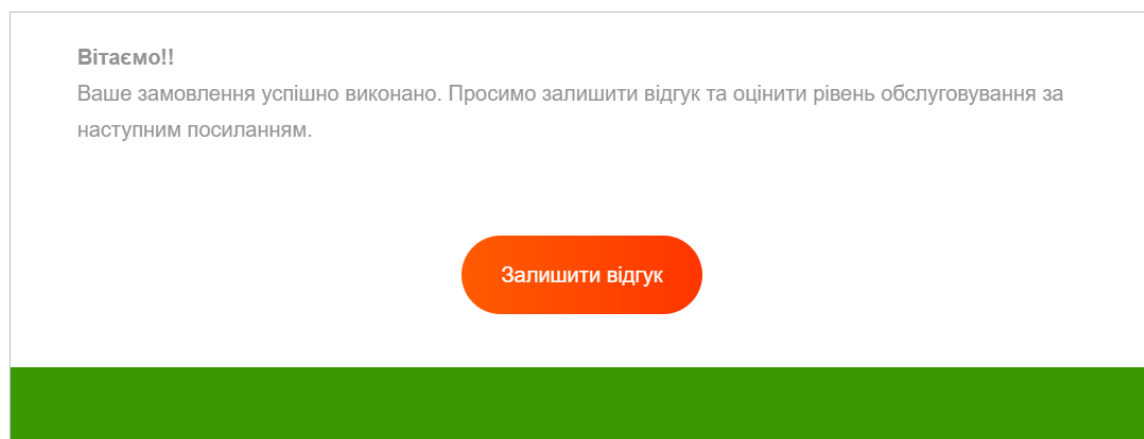


Рисунок 3.40 – Повідомлення на електронній пошті, нагадування про залишення відгуку

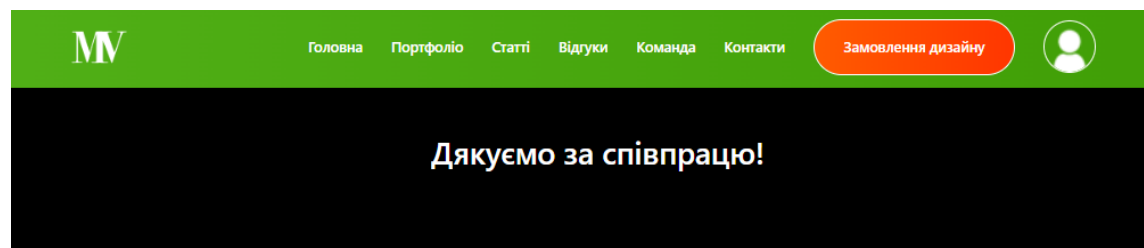


Рисунок 3.41 – Повідомлення після залишення відгуку

Наступною сторінкою, що стала відкритою для авторизованого у web-додатку користувача є сторінка «Редагування профілю». Користувач має можливість додати фото та відредагувати особисту інформацію (рис.3.41-44). Крім того, присутня можливість змінити пароль.

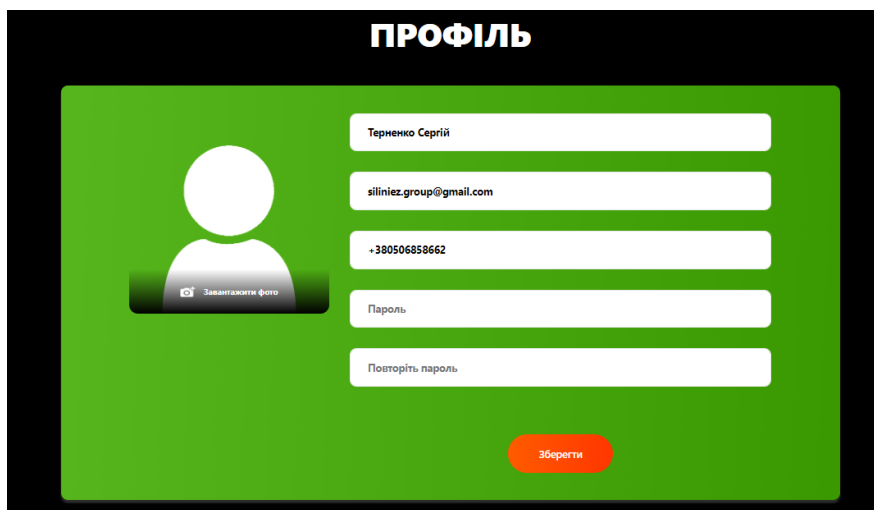


Рисунок 3.42 – Сторінка користувача для редагування профілю

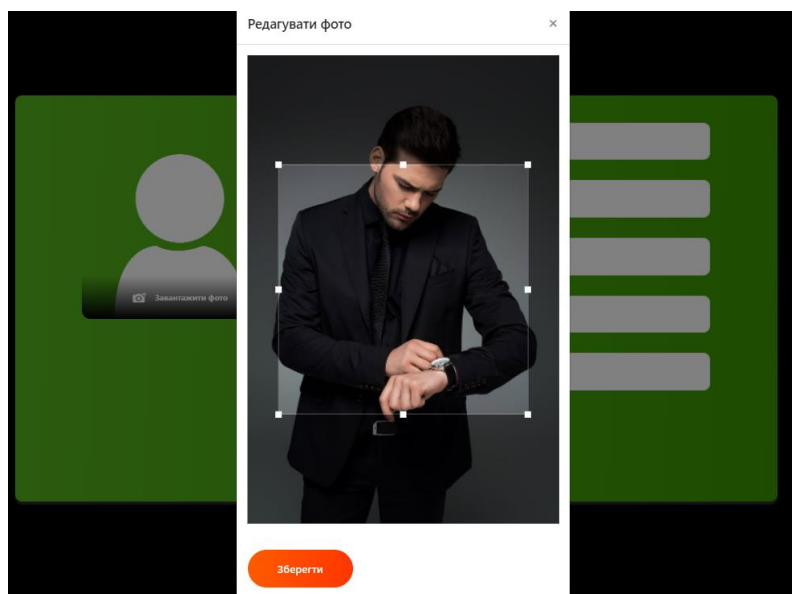


Рисунок 3.43 – Редагування та кадрування при додаванні особистого фото

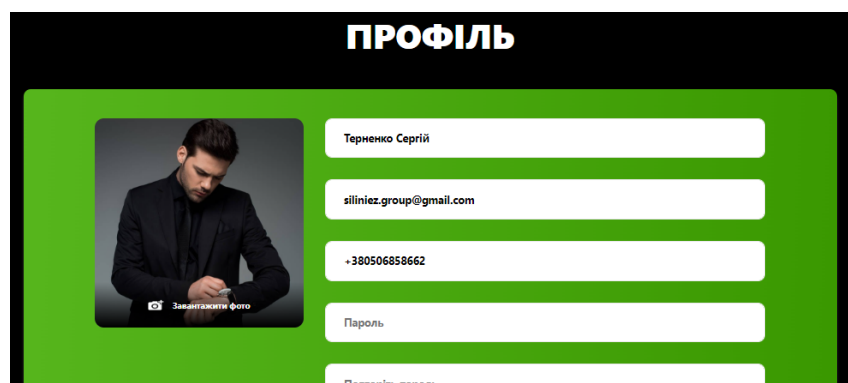
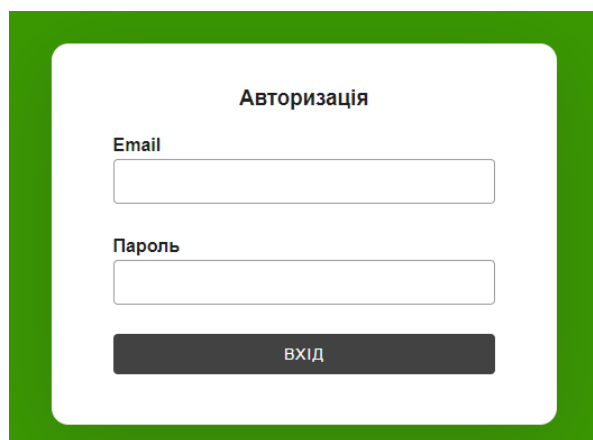


Рисунок 3.44 – Результат додавання особистого фото користувачу

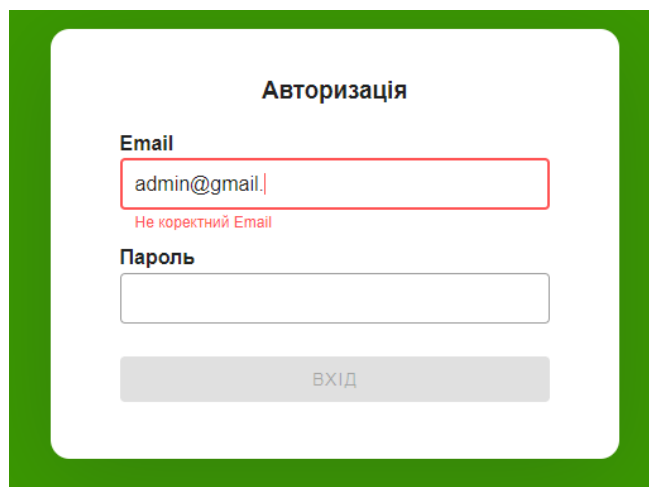
3.4 Адміністрування сайту

Переходячи за посиланням web-додатку організації замовлення дизайну сайту «studio-design.zzz.com.ua/admin» користувач перенаправляється на адмін-панель (рис.3.45-46).



The image shows a login form titled "Авторизація" (Authorization) enclosed in a green border. It contains two input fields: "Email" and "Пароль" (Password). Below the fields is a dark grey button labeled "ВХІД" (Login).

Рисунок 3.45 – Сторінка із формою авторизації адміністратора



The image shows the same login form as in Figure 3.45, but with a validation error. The "Email" field contains the text "admin@gmail.". A red border highlights the input field, and a red error message "Не коректний Email" (Incorrect Email) is displayed below it. The "Пароль" field is empty, and the "ВХІД" button is greyed out.

Рисунок 3.46 – Сторінка із формою авторизації адміністратора, приклад валідації даних

Першим пунктом навігаційного меню є «Портфоліо». Дана сторінка дозволяє адміністратору web-додатку організації замовлення дизайну сайту додавати або редагувати приклади виконаних замовлень по дизайну сайтів (рис.3.47-48).

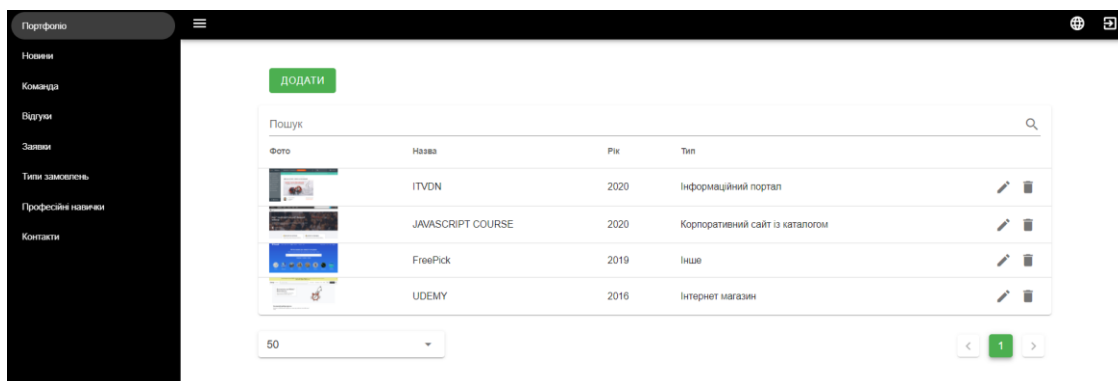


Рисунок 3.47 – Сторінка із портфоліо web-додатку організації замовлення дизайну сайту

Form for adding a portfolio item:

- Тип (dropdown menu)
- Назва (text input)
- Рік (text input)
- Посилання (text input)
- Опис (text area)
- Зображення (file upload input)
- ЗБЕРЕГТИ (submit button)

Рисунок 3.48 – Форма для додавання прикладу оформленого та виконаного портфоліо

Наступним пунктом навігаційного меню є «Новини». Дана сторінка дозволяє адміністратору web-додатку додавати або редагувати додані статі (рис.3.49-50).

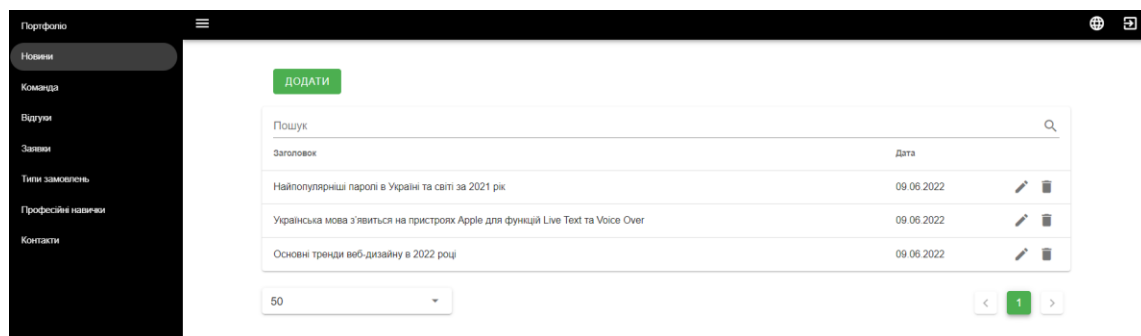


Рисунок 3.49 – Сторінка із статтями web-додатку

Рисунок 3.50 – Форма для додавання статі web-додатку

Першим пунктом навігаційного меню є «Команда». Дана сторінка дозволяє адміністратору додавати або редагувати інформацію про членів команди організації замовлення дизайну сайту (рис.3.51-52).

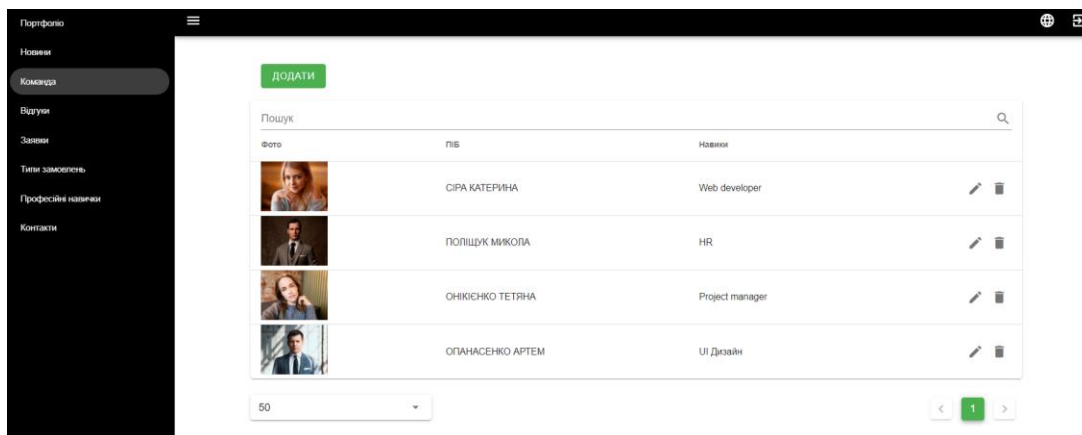



Рисунок 3.51 – Сторінка із членами команди організації

ПІБ
ОПАНАСЕНКО АРТЕМ

Опис
Розробляю логічний та структурований сайт найвищої якості.

Навики
UI Дизайн X

Зображення



ЗБЕРЕГТИ

Рисунок 3.52 – Форма для додавання/редагування інформації про члена команди організації

Наступним пунктом навігаційного меню є «Відгук». Дана сторінка дозволяє адміністратору web-додатку переглядати отримані відгуки від замовників виконаних завдань (рис.3.53).

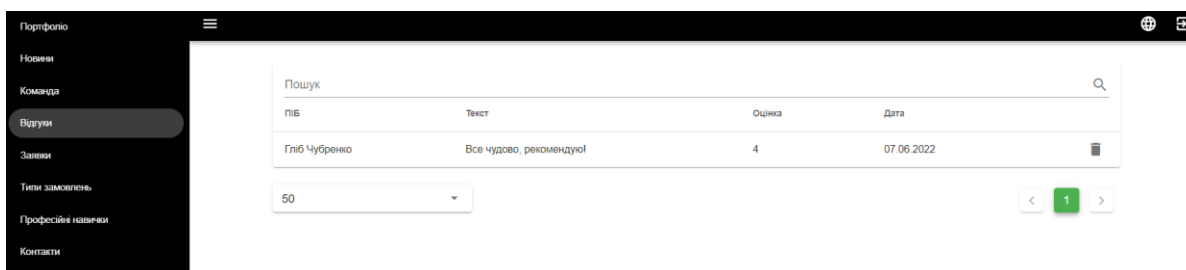


Рисунок 3.53 – Сторінка із відгуками web-додатку

Наступним пунктом навігаційного меню є «Заявки». Дана сторінка дозволяє адміністратору переглядати отримані замовлення від користувачів web-додатку (рис.3.54).

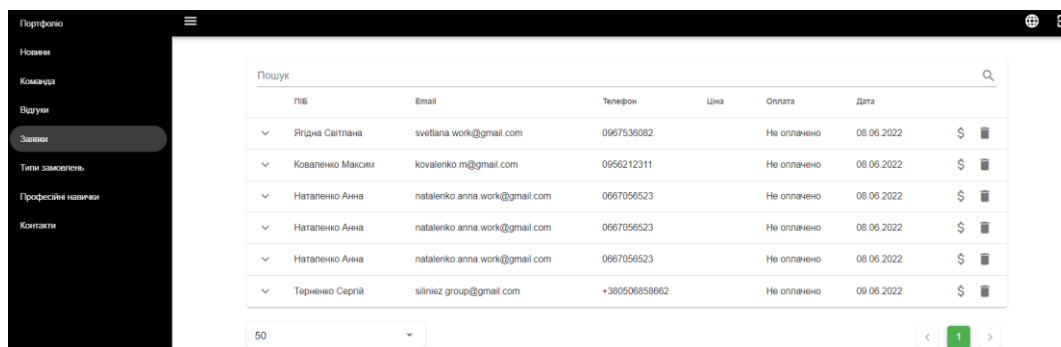


Рисунок 3.54 – Сторінка із заявками web-додатку

Кожну із заявок можна переглянути більш детально та вказати ціну виконання того чи іншого замовлення (рис.3.55-57).

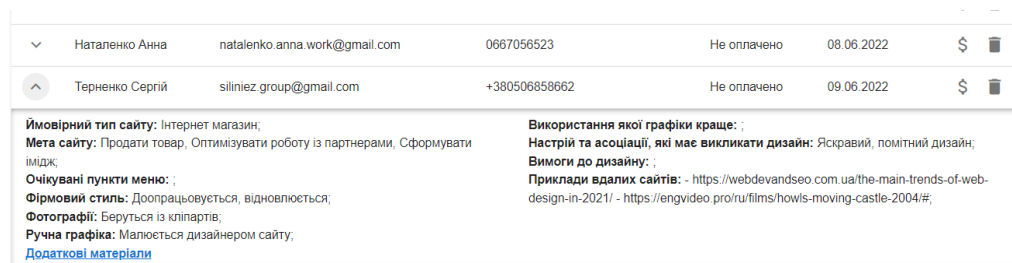


Рисунок 3.55 – Сторінка із заявками web-додатку, перегляд детальної інформації по замовленню

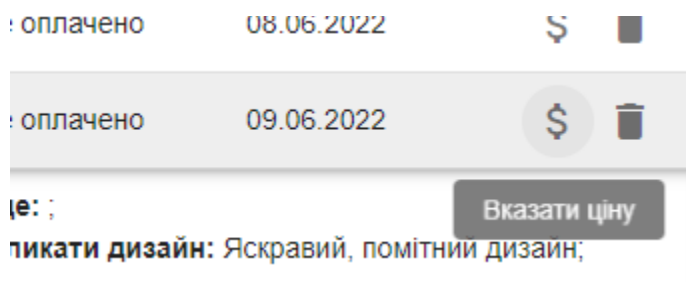


Рисунок 3.56 – Сторінка із заявками web-додатку, кнопка для додавання ціни виконання замовлення

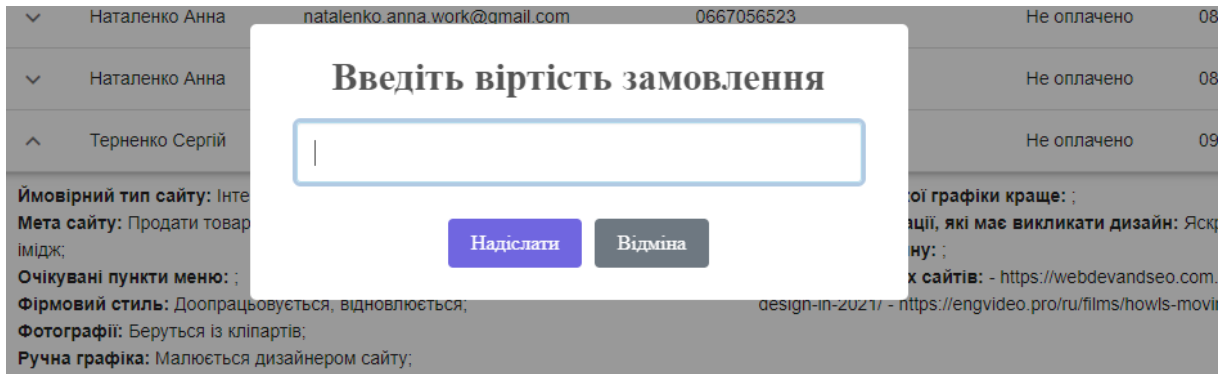


Рисунок 3.57 – Сторінка із заявками web-додатку, додавання ціни

Наступними пунктами навігаційного меню є «Тип замовлення», «Навичка» та «Контакти». Дані сторінки дозволяють адміністратору взаємодіяти із кожним типом інформації на відповідній сторінці (рис.3.58-60).



Рисунок 3.58 – Сторінка із типами замовлення web-додатку

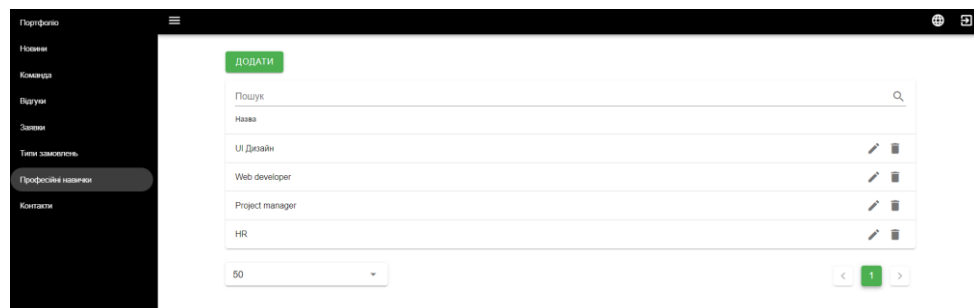


Рисунок 3.59 – Сторінка із професійними навичками членів команди web-додатку

Портфолио

Новини

Команда

Відгуки

Завши

Типи замовлень

Професійні навички

Контакти

Адрес
Вулиця Римського-Корсакова, 2

Телефон
+350506748693

Email
designstudio@gmail.com

ЗБЕРЕГТИ

Рисунок 3.60 – Сторінка із контактною інформацією web-додатку

ВИСНОВКИ

При виконанні роботи «Web-додаток організації замовлення дизайну сайту» було проаналізовано аспекти та важливі моменти при розробці web-додатку прийняття та ведення замовлень по web-дизайну.

Проаналізувавши web-додатки організацій по замовленню дизайнів та сайтів було виконано порівняльний аналіз, де було виявлено позитивні та негативні сторони, які при створенні web-додатку для організації замовлення треба уникати.

У результаті формування загального представлення про роботу було підсумовано, що web-додаток організації замовлення дизайну сайту можна буде використовувати для формування та погодження технічного завдання, передачі результатів, супроводу. Web-додаток може використовуватися як систему для організації замовлень дизайну сайту. Даний сервіс може допомогти полегшити роботу із користувачами. Крім того, самі користувачі можуть із легкістю ознайомитися із роботами команди.

При розробці та декомпозиції IDEF0 діаграми web-додатку організації замовлення дизайну сайту було сформовано етапи формування та виконання замовлення по дизайну сайту. Результатом декомпозиції є сформовані шість етапів.

Результатом виконаної роботи є web-додаток організації замовлення дизайну сайту, що покращує взаємодію із замовниками.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. How to Start a Web Design Business [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://zyro.com/blog/web-design-business/> (дата звернення: 02.06.2022).
2. 22 inspiring web design trends [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://webflow.com/blog/web-design-trends-2022> (дата звернення: 02.03.2022).
3. Web Design Tips You Can't Ignore in 2022 [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://www.entrepreneur.com/article/424937> (дата звернення: 02.06.2022).
4. How to Start an Online Website Design Business in 2022 [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://wperp.com/87877/how-to-start-an-online-website-design-business/> (дата звернення: 02. 06.2022).
5. Web Design Trends for Business Websites [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://1stwebdesigner.com/web-design-trends-for-business-websites-in-2022/> (дата звернення: 02.06.2022).
6. Best Website Designs [Електронний ресурс]. – 2020. – Режим доступу до ресурсу: <https://www.designrush.com/best-designs/websites/> (дата звернення: 02. 06.2022).
7. R. Anaya-Sánchez, J. M. Castro-Bonaño, E. González-Badía, Millennial consumer preferences in social commerce web design. *Revista Brasileira de Gestao de Negocios*. 22, 123–139 (2020).
8. Innovative design trends for inspiration – 2020. – Режим доступу до ресурсу: <https://www.insightsforprofessionals.com/marketing/digital-marketing/innovative-website-designs> (дата звернення: 02. 06.2022).

9. «ОНО» [Электронный ресурс]. – 2020. – Режим доступа до ресурсу: <https://www.designrush.com/best-designs/websites/> (дата звернення: 02.06.2022).
10. «Brainlab» [Электронный ресурс]. – 2020. – Режим доступа до ресурсу: <https://www.designrush.com/best-designs/websites/> (дата звернення: 02.06.2022).
11. «WebEra» [Электронный ресурс]. – 2020. – Режим доступа до ресурсу: <https://www.designrush.com/best-designs/websites/> (дата звернення: 02.06.2022).
12. I. Reid, D. Bamford, H. Ismail, Reconciling engineer-to-order uncertainty by supporting front-end decision-making. *International Journal of Production Research*. 57, 6856–6874 (2019);
13. M. Țichindelean, M. T. Țichindelean, I. Cetină, G. Orzan, A comparative eye tracking study of usability—towards sustainable web design. *Sustainability (Switzerland)*. 13 (2021);
14. K. Meitinger, T. Kunz, Visual Design and Cognition in List-Style Open-Ended Questions in Web Probing. *Sociological Methods and Research* (2022);
15. Information technology management - <https://www.pearson.com/pathways/areas-work-study/information-technology-management.html> (дата звернення: 02.03.2022).
16. What is IT Management? Skills and Standards to Build Credibility [Электронный ресурс]. – 2022. – <https://www.ittoolkit.com/articles/what-is-IT-management> (дата звернення: 02.06.2022).
17. Step-by-Step Guide - How to Create an IDEF0 Diagram [Электронный ресурс]. – 2022. – <https://www.edrawsoft.com/how-to-create-idef0-diagram.html> (дата звернення: 02.03.2022).
18. How to Create an IDEF0 Diagram for an Application Development [Электронный ресурс]. – 2020. – <https://www.conceptdraw.com/helpdesk/how-to-create-idef0-diagram> (дата звернення: 02.06.2022).
19. L. Campoli et al., UnDiFi-2D: An unstructured discontinuity fitting code for 2D grids. *Computer Physics Communications*. 271 (2022);

20. E. P. Williamson et al., Using static web technologies and git-based workflows to re-design and maintain a library website (quickly) with non-technical staff. *College and Undergraduate Libraries*. 28, 129–147 (2022).

ДОДАТОК А

ТЕХНІЧНЕ ЗАВДАННЯ

**для розробки кваліфікаційної роботи бакалавра
«Web-додаток організації замовлення дизайну сайту»**

1 ПРИЗНАЧЕННЯ Й МЕТА СТВОРЕННЯ WEB-ДОДАТКУ

1.1 Призначення web-додатку

Web-додаток використовуватиметься як система для організації замовлень дизайну сайту. Даний сервіс зможе допомогти полегшити роботу із користувачами. Крім того, самі користувачі зможуть із легкістю ознайомитися із роботами команди.

1.2 Мета створення web-додатку

Створення відкритої платформи для замовників тих чи інших проектів пов'язані із дизайном. Даний сервіс повинен полегшити отримання замовлень по дизайну та комунікацію між клієнтами та виконавцями.

1.3 Цільова аудиторія web-додатку

До цільової аудиторії web-додатку можна віднести практично всіх людей, що зацікавлені у послугах web-дизайнера.

Наведемо декілька прикладів цільової аудиторії даного web-додатку:

- Ресторанний бізнес;
- салони та майстри в сфері краси;
- власники малого бізнесу, що хочуть розвиватися в онлайн послугах;
- інтернет магазини;
- державні установи й тощо.

.

2 ВИМОГИ ДО WEB-ДОДАТКУ

2.1. Основні функції web-додатку

Функції web-додатку для не зареєстрованих користувачів:

- Реєстрація користувачів;
- Перегляд інформації на сайті про команду;
- Створення пошуку для формування списку робіт за різними критеріями.

Основні функції web-додатку для зареєстрованих користувачів, що розширяють вже отриманий функціонал:

- Авторизація користувачів;
- Створення заявок на отримання роботи з можливістю прикріплення файлів;
- Формування відгуків по виконаній роботі.

Основні функції web-додатку для адміністраторів:

- Додавання приклади та історій виконаних проектів.
- Розміщення на web-додатку прикладів роботи з можливістю прикріплення фотографій та посилань на реалізацію.
- Формування оповіщень.
- Редагування даних зі сторони адміністративної панелі.
- Взаємодія із замовником через адміністративну панель.
- Можливість формування автоматичної відповіді та розсилок на електронну пошту через адміністративну панель.

2.2. Загальна інформація про структуру web-додатку

Структура web-додатку являє собою набір сторінок, які також є пунктами головного меню.

Такими розділами є:

- Головна – на сторінці зображені головне меню та загальна інформація по типу роботи;
- Проекти – інформація виконані проекти;
- Відгуки – відгуки по виконаним роботам в портфолію;
- Команда – інформація про кожного члена команди;
- Статті – останні новини від команди;
- Контакти – список всіх можливих видів зв'язку;
- Замовлення дизайну – функція для замовлення дизайну.

2.2. Розподіл функцій за групами користувачів

Адміністратори ІС:

- Редагування інформації;
- Додавання статей;
- Додавання нові приклади робіт;
- Відповідати на отримані замовлення.

Користувач web-додатку:

- Перегляд інформації на сайті;
- Створення заявки на виконання роботи по дизайну;
- Коментувати статті;

2.3. Дизайн та структура додатку

Інтерфейс web-додатку повинен забезпечувати інтуїтивно зрозуміле уявлення структури розміщеної на ньому інформації, логічний перехід до розділів і сторінок. Посилання на сторінки повинні бути забезпечені заголовками. При виборі будь-якого з пунктів меню користувачем повинна завантажуватися

відповідна йому інформаційна сторінка ,а в блоці меню відкриватися список підрозділів.

Прототип майбутнього дизайну представлений на рисунку А.1 та А.2.



Рисунок А.1 – Прототип головної сторінки

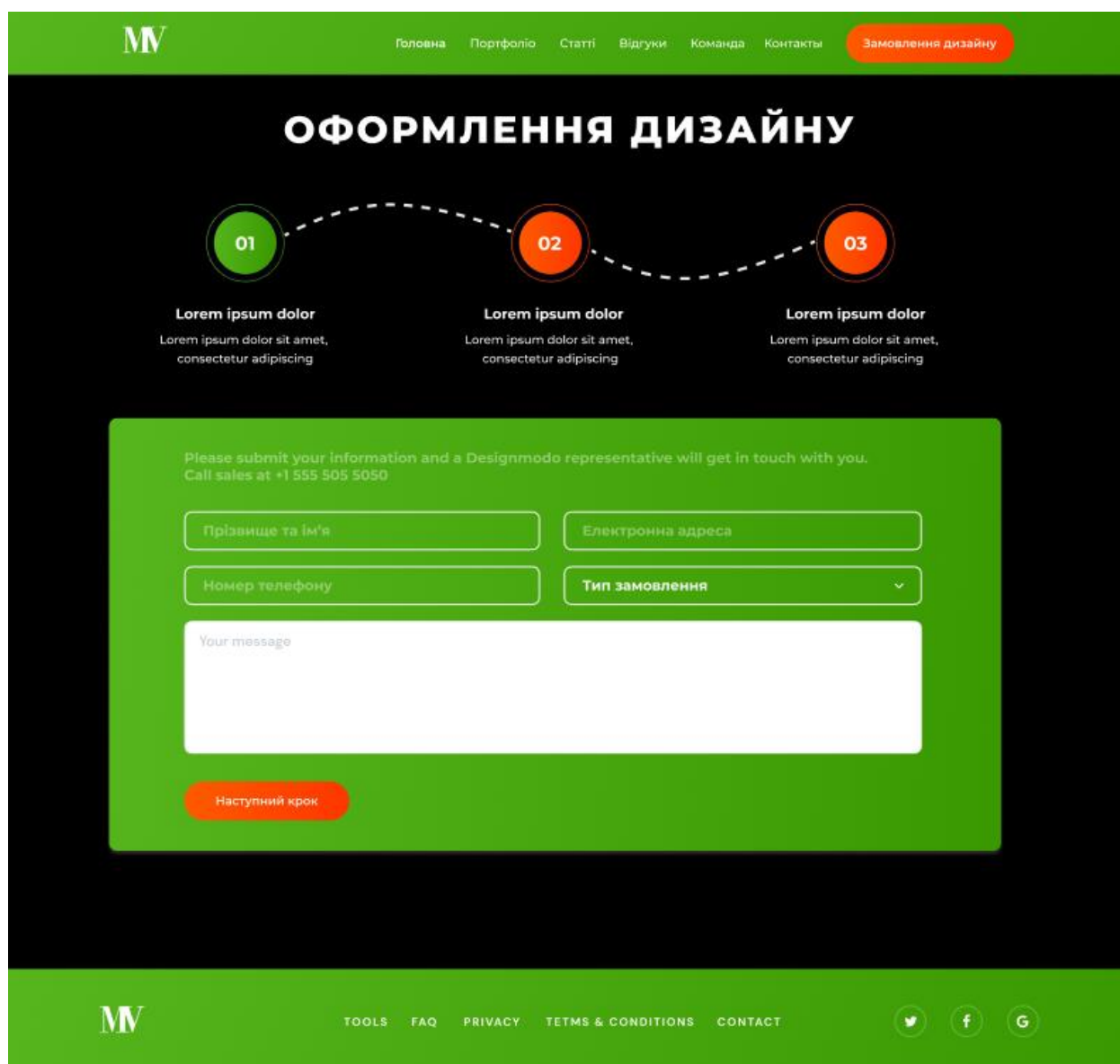


Рисунок А.2 – Оформлення замовлення

2.4. Вимоги до видів забезпечення

2.4.1 Вимоги до інформаційного забезпечення

Реалізація web-додатку відбувається з використанням:

- HTML&CSS
- PHP 7.4.4 та його фреймворки
- MySQL 8.0

2.4.2 Вимоги до лінгвістичного забезпечення

Web-додаток має бути виконаний українською мовою.

2.4.3 Вимоги до програмного забезпечення

Програмне забезпечення клієнтської частини повинне задовольняти наступним вимогам: веб-браузер: Firefox 3.5 і вище, або Opera 9.5 і вище, або Chrome 2 і вище.

2.5. Програмні засоби

Повинні бути виконані умови ліцензійності програмних продуктів, що будуть використані при розробці.

Для розробки передбачається використання наступних програмних засобів:

Front-end – HTML5, CSS3, JavaScript, Vue.js, BootstrapVue.

Back-end – MySQL, PHP, Laravel.

3 СКЛАД І ЗМІСТ РОБІТ ЗІ СТВОРЕННЯ WEB-ДОДАТКУ

Докладний опис етапів роботи зі створення сайту наведено в табл.А.1.

Таблиця А.1 – Етапи створення сайту

Тема/підтема	Час (дні)
Аналітика	10
Аналіз предметної області	1
Аналіз конкурентів	2
SWOT-аналіз	1
Розробка user story & user story map	4
Створення мережевого графіку та діаграми Ганта	2
Розробка дизайну	8
Створення low-дизайну	2
Створення high-дизайну	5
Розробка додаткових компонентів (анімація, зміна стану, взаємодія з користувачем)	1
Верстка сайту	15
Розробка каркасу	1
Створення компонентів	3
Стилізація	5
Розробка функціональних блоків	6
База даних	3
Проектування бази даних	1
Створення бази даних	1
Додання тестових даних в базу	1
Програмування	8
Створення контролерів	2
Створення моделей	2
Налаштування API	3
Налаштування рівнів доступу для користувачів сайту	1
Тестування	4
Перше тестування продукту	1
Трекінг та виправлення дефектів	2
Регресивне тестування	1
Наповнення контентом	2
Завантаження на сервер	1
Загальний час роботи над проектом	51

ДОДАТОК Б

ПЛАНУВАННЯ РОБІТ для розробки кваліфікаційної роботи бакалавра «Web-додаток організації замовлення дизайну сайту»

1 ІДЕНТИФІКАЦІЯ МЕТИ ІТ-ПРОЕКТУ

1.1 Розробка концепції проекту

1.1.1. Ідентифікація ідеї проекту.

Головна мета проекту – розробка web-додатку для організації замовлення дизайну сайту. Головною особливістю web-додатку є зручна у використанні форма для оформлення замовлення.

1.1.2. Деталізація мети методом SMART

Методика постановки цілей по SMART системі дозволила добитися бажаного результату тисячам людей. SMART техніка постановки цілей дозволяє досягти успіху в продажах, формує модель поведінки, виробляє стійкі правила життєвої позиції. Розглянемо детальніше інформацію за кожним з пунктів у таблиці Б.1.

Таблиця Б.1 – Таблиця опису проекту за SMART методом

Назва	Опис
Specific (конкретна)	Розробка web-додатку, за допомогою якої можна буде виконувати поставлені цілі та задачі.
Measurable (вимірювана)	Прогрес та якість буде вимірюватися в кількості користувачів, замовленнях та кількості позитивних відгуків у web-додатку.
Achievable (досяжна)	Данний web-додатку буде розроблятися з використанням HTML, CSS, Vue.js, PHP, MySQL та Laravel.
Relevant (реалістична)	Для виконання проекту є всі необхідні засоби та знання.
Time-framed (обмежена у часі)	Для роботи визначений дедлайн та побудована діаграма Ганта для відстеження часу виконання web-додатку.

1.2 Техніко-економічне дослідження

1.2.1. Дослідження IT-проекту, організації, регіону

Створений програмний продукт орієнтований на розвиток, вдосконалення послуг в сфері замовлення web-дизайну. Web-додаток буде розроблений для зручного використання послуг користувачем.

Даний проект матиме можливість використовуватися та приймати замовлення із будь-якого міста України. Звичайно, при розповсюдженні інформації за web-додаток у соціальних мережах та за допомогою таргет-реклами можна досягти й вищих цілей.

1.2.2. Оцінка цінності, життєздатності, економічної ефективності та життєсталості IT-проекту

Цінність – це особисте сприйняття зацікавленими сторонами здібностей продукту проекту створювати вигоди в соціальному, політичному, економічному та духовному аспектах їх життєдіяльності завдяки унікальним властивостям продукту проекту.

Життєздатність проекту — здатність зберігати існування й цінність в мінливих умовах середовища ринку та моди. Проект оцінюється як життєздатний, бо сфера зараз у розквіті й має великі перспективи.

Економічна ефективність — досягнення найбільших результатів за найменших затрат праці. Економічна ефективність є конкретною формою дії закону економії часу. Можливості програмного продукту є корисними та результативними для використання споживачами з усієї України, так як буде реалізована можливість повноцінної співпраці онлайн.

Життєздатність та життєсталість web-додатку підтверджується наявністю попиту на web-дизайн.

2 УПРАВЛІННЯ ВИМОГАМИ

2.1 Загальні вимоги до web-додатку

2.1.1 Вимоги до структури й функціонування web-додатку

Інформаційна система повинна бути реалізована у вигляді Web-додатку, доступного в мережі Інтернет. Web-додаток повинен складатися із взаємозалежних розділів із чітко розділеними функціями.

2.1.2. Вимоги до персоналу

Для підтримки web-додатку й його експлуатації при керування сайтом від персоналу повинно вимагатися загальні навички роботи з персональним комп'ютером та стандартним веб-браузером.

2.1.3. Вимоги до стилістичного оформлення web-додатку

Інтерфейс web-додатку повинен забезпечувати інтуїтивно зрозуміле уявлення структури розміщеної на ньому інформації, логічний перехід до розділів та сторінок. Посилання на сторінки повинні бути забезпечені заголовками.

2.2 Вимоги до розділів

Web-додаток повинен складатися з наступних розділів:

- Головна – на сторінці зображені головне меню та загальна інформація;
- Проекти – інформація виконані проекти;
- Відгуки – відгуки по виконаним роботам в портфолію;
- Команда – інформація про кожного члена команди;
- Статті – останні новини від команди;
- Контакти – список всіх можливих видів зв'язку;
- Оформлення замовлення.

3 ОПИСАННЯ ФАЗИ РОЗРОБКИ ІТ-ПРОЕКТУ

3.1 Планування змісту структури робіт ІТ-проекту

WBS проекту - це розбиття проекту на конкретні результати, які повинні бути досягнуті для досягнення цілей проекту. Важливо розуміти, що в WBS збираються саме результати робіт, а не завдання, які потрібно виконати для отримання цих результатів. Вона є ієрархічною структурою, що показує подальший розподіл необхідних для виконання мети зусиль; наприклад, програма, проект чи договір.

Загальна WBS структура представлена на рисунку Б.1.



Рисунок Б.1 – WBS-структура проекту

3.2. Планування структури організації

Структура розподілу організації або OBS – це ієрархічна модель, що описує встановлені організаційні рамки для планування проектів, управління ресурсами, відстеження часу та витрат, розподілу витрат, звітування про доходи / прибуток та управління роботою.

Ця структура стосується тільки внутрішньої організаційної структури проекту і не зачіпає відносин проектних груп чи учасників з батьківськими організаціями.

Загальна OBS структура представлена на рисунку Б.2.

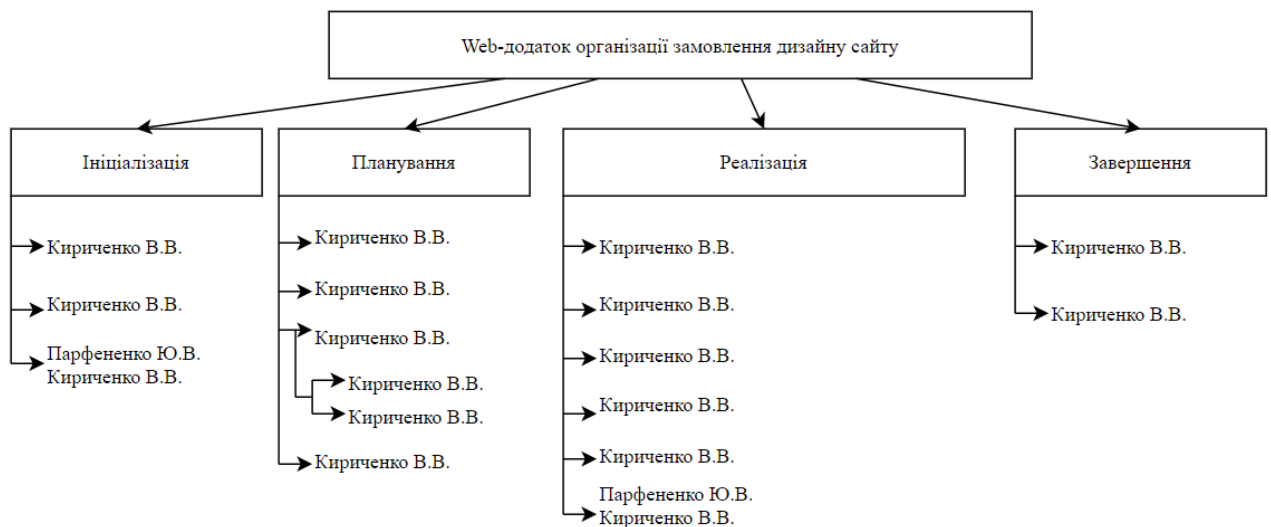


Рисунок Б.2 – OBS структура проекту

3.3. Побудова матриці відповідальності

На підставі OBS та WBS структур будується матриця відповідальності проекту. Матриця відповідальності закріплює за кожною елементарною роботою виконавця. Використовується для контролю відповідності розподілу ролей цілям проекту. На верхньому рівні розподіляються ролі та відповідальність по елементах

OBS. На нижньому — фази проекту. Матриця відповідальності представлена в таблиці Б.2.

Таблиця Б.2 – Матриця відповідальності

Назва етапу		Кириченко В.В.	Парфененко Ю.В.
1	Ініціалізація	+	+
1.1	Ознайомлення з предметної областю проекту	+	-
1.2	Аналіз конкурентів	+	-
1.3	Ідентифікація ідей	+	+
2	Планування	+	-
2.1	Підготовка матеріалу	+	-
2.2	Вивчення вимог проекту	+	-
2.3	Розробка календарного плану	+	-
2.4	Планування ризиків	+	-
3	Реалізація	+	+
3.1	Розробка прототипу	+	-
3.2	Підготовка макету дизайну сайту	+	-
3.3	Верстка	+	-
3.4	Робота над програмною частиною	+	-
3.5	Наповнення сайту контентом	+	-
3.6	Тестування	+	+
4	Завершення	+	-
4.1	Підготовка документації	+	-
4.1	Перенесення сайту на сервер	+	-

3.4. Побудова календарного графіку виконання ІТ-проекту

Діаграма Ганта – це горизонтальна діаграма з тимчасовою шкалою, яка використовується для ілюстрації плану робіт за проектом з прив'язкою до часу.

За допомогою діаграм Ганта керівники проектів і менеджери по продукту розбивають проекти на робочі завдання для зручності управління, підтримують порядок в роботі і роблять залежності між завданнями наочними.

Розглянемо створену діаграму Ганта до заданої інформаційної системи (рис.Б.3-4).

Завдання	Начало	Завершення	Длительность
	17.01.2022 09:00	03.06.2022 18:00	100д
1 «Інформаційна система пі...	17.01.2022 09:00	27.05.2022 18:00	95д
1.1 1 Аналітика	17.01.2022 09:00	28.01.2022 18:00	10д
1.1.1 1.1 Ознайомлення з пр...	17.01.2022 09:00	21.01.2022 18:00	5д
1.1.2 1.2 Ідентифікація ідей ...	17.01.2022 09:00	24.01.2022 18:00	6д
1.1.3 1.3 Аналіз додатків кан...	24.01.2022 09:00	28.01.2022 18:00	5д
1.1.4 1.4 Формування перелі...	24.01.2022 09:00	28.01.2022 18:00	5д
+ Додати завдання Додати вік			
1.2 2. Планування роботи н...	31.01.2022 09:00	21.02.2022 18:00	16д
1.2.1 2.1 Розробка технічаг...	31.01.2022 09:00	04.02.2022 18:00	5д
1.2.2 2.2 Розробка Діаграм...	07.02.2022 09:00	17.02.2022 18:00	9д
1.2.3 2.3 SWOT-аналіз	14.02.2022 09:00	21.02.2022 18:00	6д
1.2.4 2.4 Розробка user story ...	14.02.2022 09:00	21.02.2022 18:00	6д
1.2.5 2.5 Розробка Мережев...	07.02.2022 09:00	17.02.2022 18:00	9д
+ Додати завдання Додати вік			
1.3 3. Реалізація	07.02.2022 09:00	27.05.2022 18:00	80д
1.3.1 3.1 Розробка дизайну	07.02.2022 09:00	18.03.2022 18:00	30д
1.3.2 3.2 Верстка сайту	14.03.2022 09:00	22.04.2022 18:00	30д
1.3.3 3.3 База даних	21.03.2022 09:00	15.04.2022 18:00	20д
1.3.4 3.4 Програмування	04.04.2022 09:00	06.05.2022 18:00	25д
1.3.4.1 3.4.1 Створення кан...	04.04.2022 09:00	06.05.2022 18:00	25д
1.3.4.2 3.4.2 Створення мод...	04.04.2022 09:00	15.04.2022 18:00	10д
1.3.4.3 3.4.3 Налаштування ...	11.04.2022 09:00	06.05.2022 18:00	20д
1.3.4.4 3.4.4 Налаштування ...	18.04.2022 09:00	06.05.2022 18:00	15д
+ Додати завдання Додати вік			
1.3.5 3.5 Тестування	09.05.2022 09:00	27.05.2022 18:00	15д
+ Додати завдання Додати вік			
+ Додати завдання Додати вік			
2 4 Реліз додатку	09.05.2022 09:00	03.06.2022 18:00	20д
2.1 4.1 Наповнення контентом	09.05.2022 09:00	03.06.2022 18:00	20д
2.2 4.2 Завантаження на сервер	23.05.2022 09:00	03.06.2022 18:00	10д

Рисунок Б.3 – Діаграма Ганта (інформація)

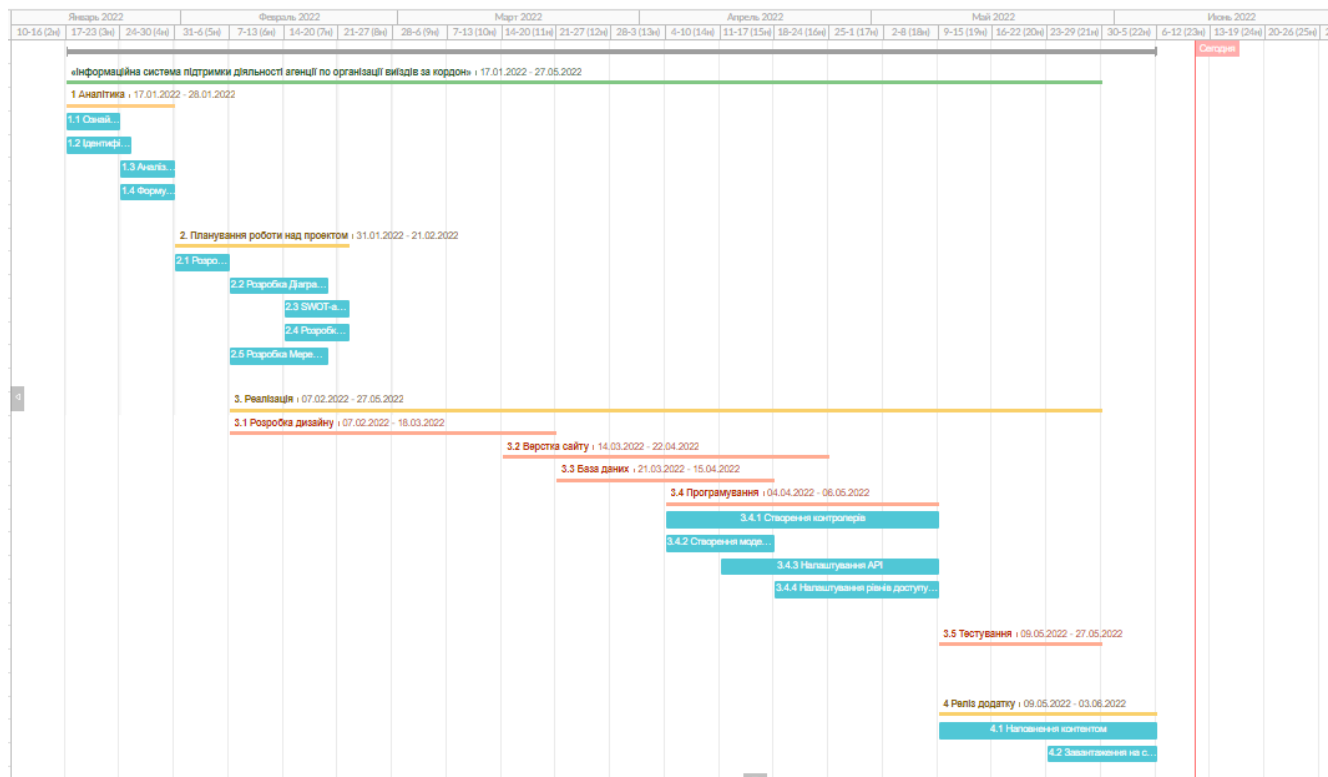


Рисунок Б.4 – Діаграма Ганта(графік)

4 УПРАВЛІННЯ РИЗИКАМИ ІТ-ПРОЕКТУ

Ризиком є ймовірна подія, яка у випадку її виникнення може як негативно, так і позитивно вплинути на конкретний проект. Управління ризиком – це процес зміни ризиків та реагування на події під час виконання проекту. При реалізації проекту важливою частиною є моніторинг ризиків. Отже, в даному випадку можна виділити деякі ризики.

Під час процесу аналізу для визначення числових значень ймовірності появи ступеня впливу, застосовувалася методика експертних оцінок. Виходячи цих оцінок можливо знайти ранг ризиків: $R = P * L$, де

- R – ранг ризику;
- P – ймовірність виникнення;
- L – ступінь впливу.

Шкала оцінки ризику може відповідати емпіричній шкалі оцінки ризику:

- 5 балів - критичний ризик (0,81 - 1);
- 4 бали - максимальний ризик (0,61 - 0,8);
- 3 бали - високий ризик (0,41 - 0,6);
- 2 бали - нормальний ризик (0,31 - 0,4); 1 бал - малий ризик (0 - 0,3).

Оцінки ризиків проекту наведено в табл. Б.4.

Таблиця Б.4 – Таблиця ризиків

№	Об'єкт ризику	Ризик	P	L	R
1	Час	Зміна пріоритету проекту	0,5	0,3	0,15
2	Якість	Системи, що не відповідають задачам , грубі помилки в алгоритмах процесів, критичні збої системи	0,1	0,4	0,04

№	Об'єкт ризику	Ризик	P	L	R
3	Трудові ресурси та їх кваліфікація	Неможливість участі в запланованих роботах по проекту необхідних співробітників зі сторони замовника та виконавця у зв'язку з відпусткою, відрядженням та ін.	0,1	0,1	0,01
4	Інтеграція	Не вірна інформація, щодо зовнішніх систем, з яких передбачена взаємодія в межах проекту	0,5	0,4	0,2
5	Ринок	Розширення функціональних характеристик програмних продуктів, які використовувалися замовником в межах мети проекту	0,1	0,05	0,005
6	Ринок	Зміна вимог замовника, економічні зміни, посилення конкуренції, втрата позицій на ринку;	0,5	0,1	0,05
7	Бюджет	Зриви планів робіт; неправильна стратегія; персонал без потрібної кваліфікації; переплати за роботу/матеріали; непогоджені частини проекту; невірний кошторис.	0,5	0,2	0,1

ДОДАТОК В. КОД РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

create_order_table.php

Міграція створення таблиці замовлення.

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateOrderTable extends Migration
{
    public function up()
    {
        Schema::create('order', function (Blueprint $table) {
            $table->id();
            $table->text('data');
            $table->string('file')->nullable();
            $table->foreignId('user_id');
            $table->boolean('status')->default(0);
            $table->boolean('pay')->default(0);
            $table->string('cost')->nullable();
            $table->timestamps();
        });

        Schema::table('order', function (Blueprint $table) {
            $table->index('user_id');
            $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
        });
    }

    public function down()
    {
        Schema::dropIfExists('order');
    }
}
```

create_review_table.php

Міграція створення таблиці відгуків.

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateReviewsTable extends Migration
{
    public function up()
    {
        Schema::create('reviews', function (Blueprint $table) {
            $table->id();
            $table->text('description');
            $table->integer('rate');
            $table->foreignId('order_id');
            $table->timestamps();
        });
        Schema::table('reviews', function (Blueprint $table) {
            $table->index('order_id');
            $table->foreign('order_id')->references('id')->on('order')->onDelete('cascade');
        });
    }
    public function down()
    {
        Schema::dropIfExists('reviews');
    }
}
```

OrderController.php

Контроллер замовлень.

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Crypt;

use App\Models\Order;

use Mail;
use Auth;

class OrderController extends Controller
{
    protected $publicStorage = "/uploads/orders/";

    function get() {
        $model = Order::with('user')->get();
        return response()->json($model);
    }
    function getMyOrder(Request $request) {

        $model = Order::where('user_id', Auth::id());

        if($request->search != "") {
```

```

$model->where('data', 'like', "%".$request->search."%");
}

if($request->work_type != "") {
    $model->where('data', 'like', "%".$request->work_type."%");
}

if($request->year != "") {
    $model->whereYear('created_at', $request->year);
}

$model = $model->get();

foreach ($model as $key => $value) {
    if($value->cost) {
        $test = [
            "public_key" => "sandbox_i13104730401",
            "version" => "3",
            "action" => "pay",
            "amount" => $value->cost,
            "currency" => "UAH",
            "description" => "Оплата послуг",
            "order_id" => rand(2,50),
            "mode" => "popup",
            "result_url" => "http://studio-design.zzz.com.ua/success-pay/" . Crypt::encryptString($value->id)
        ];

        $value->test = base64_encode(json_encode($test));
        $sign_string = 'sandbox_2PMuGYbC4mVZtH1II83WxzWWVkiVAQg75949nhFd' . $value->test .
'sandbox_2PMuGYbC4mVZtH1II83WxzWWVkiVAQg75949nhFd';
        $value->signature = base64_encode( sha1( $sign_string, true ) );
    }
}

return response()->json($model);
}

function post(Request $request) {
    $model = new Order();
    $data = $request->all();
    if(isset($request['file']) && $request['file'] != "null" && $request['file'] != "undefined") {
        $name = $this->publicStorage . uniqid().'.'.$request['file']->getClientOriginalExtension();
        $request['file']->move(public_path()).$this->publicStorage, $name);
        $data['file'] = $name;
    }
    $data['user_id'] = Auth::id();
    $newsResponse = $model->create($data);
    return response()->json([
        'id' => $newsResponse->id
    ]);
}

function delete($id) {
    Order::find($id)->delete();
    return response('ok', 200);
}

function postOrderReview($id) {
    $orderEmail = Order::with('user')->find($id);

    Order::find($id)->update([
        'status' => 1
    ]);
}

```

```

$link = $_SERVER['SERVER_NAME'] . '/review/' . Crypt::encryptString($id);

Mail::send('mail.review', ['text' => $link], function($message) use ($orderEmail) {
    $message
        ->to($orderEmail->user['email'], "Шановний клієнт")
        ->subject("Ваше замовлення виконано");

    $message->from('mishaotroshenko2013@gmail.com', 'Студія дизайну');
});
return response('ok', 200);
}

function postOrderPay(Request $request, $id) {
    $orderEmail = Order::with('user')->find($id);

    Order::find($id)->update([
        'cost' => $request->cost
    ]);

    $link = $_SERVER['SERVER_NAME'] . '/pay/' . Crypt::encryptString($id);

    $text = "Дякуємо, що звернулись до нашої компанії. Ми отримали вашу заявку. Ватість Вашого проекту складає:
    " . $request->cost . " грн.";

    Mail::send('mail.pay', ['link' => $link, 'text' => $text], function($message) use ($orderEmail) {
        $message
            ->to($orderEmail->user['email'], "Шановний клієнт")
            ->subject("Виконайте оплату замовлення");

        $message->from('mishaotroshenko2013@gmail.com', 'Студія дизайну');
    });

    return response('ok', 200);
}

function checkPay($key) {
    $id = Crypt::decryptString($key);
    $data = Order::find($id);

    $test = [
        "public_key" => "sandbox_i13104730401",
        "version" => "3",
        "action" => "pay",
        "amount" => $data->cost,
        "currency" => "UAH",
        "description" => "Оплата послуг",
        "order_id" => rand(2,50),
        "mode" => "popup",
        "result_url" => "http://studio-design.zzz.com.ua/success-pay/" . $key
    ];

    $data->test = base64_encode(json_encode($test));
    $sign_string = 'sandbox_2PMuGYbC4mVZtH1I183WxzWWVkiVAQg75949nhFd' . $data->test .
'sandbox_2PMuGYbC4mVZtH1I183WxzWWVkiVAQg75949nhFd';
    $data->signature = base64_encode( sha1( $sign_string, true ) );
    return response()->json($data);
}

function setPay($key) {

```

```

    $id = Crypt::decryptString($key);
    Order::find($id)->update([
        'pay' => 1
    ]);
    return redirect('/');
}
}

```

ReviewController.php

КОНТРОЛЛЕР ВІДГУКІВ.

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Crypt;

use App\Models\Reviews;
use App\Models\Order;

class ReviewController extends Controller
{
    function get(Request $request) {
        $model = Reviews::with('order.user');

        if($request->star != "") {
            $model->where('rate', $request->star);
        }

        if($request->search != "") {
            $model->where('description', 'like', "%".$request->search."%");
        }

        if($request->work_type != "") {
            $model->whereHas('order', function($q) use ($request) {
                $q->where('data', 'like', "%".$request->work_type."%");
            });
        }

        if($request->year != "") {
            $model->whereHas('order', function($q) use ($request) {
                $q->whereYear('created_at', $request->year);
            });
        }

        $model = $model->get();

        return response()->json($model);
    }
    function post(Request $request, $key) {
        $model = new Reviews();
        $data = $request->all();
        $id = Crypt::decryptString($key);
        $data['order_id'] = $id;
        $model->create($data);
        Order::find($id)->update([
            'status' => $id
        ]);
        return response('ok', 200);
    }
}

```

```

}
function delete($id) {
    Reviews::find($id)->delete();
    return response('ok', 200);
}
function checkReview($key) {
    $id = Crypt::decryptString($key);
    if(Reviews::where('order_id', $id)->exists()) {
        return response()->json([
            'status' => true
        ]);
    } else {
        return response()->json([
            'status' => false
        ]);
    }
}
}
}

```

AuthController.php

Контроллер авторизації.

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use App\Models\User;
use Auth;
use Cookie;

use Image;

class AuthController extends Controller
{
    protected $fileStorage = "/uploads/users/";

    function checkUser() {
        $model = User::where('id', Auth::id()->where('is_admin', 0));
        $data = $model->firstOrFail();
        return response()->json($data);
    }

    function checkAdmin() {
        $model = User::where('id', Auth::id()->where('is_admin', 1));
        $data = $model->firstOrFail();
        return response()->json($data);
    }

    function update(Request $request) {
        $model = User::find(Auth::id());
        $data = $request->all();

        if($request->new_photo != 'null') {
            $name = $this->fileStorage . uniqid() . '.' . $request['new_photo']->getClientOriginalExtension();
            $nameSmall = $this->fileStorage . uniqid() . '_small.' . $request['new_photo']->getClientOriginalExtension();
            $data['photo'] = $name;
            $data['photo_small'] = $nameSmall;
        }
    }
}

```

```

}

if($request->new_photo != 'null') {
    $this->cropImg(
        $request->file('new_photo'),
        $name,
        $nameSmall,
        $request->cropWidth,
        $request->cropHeight,
        $request->cropLeft,
        $request->cropTop
    );
}
$data["password"] = Hash::make($request->password);
$model->update($data);
$user = User::find(Auth::id());
return response()->json($user);
}

function register(Request $request) {
    $request->validate([
        'email' => 'required|string|email|unique:users',
        'password' => 'required|string'
    ]);
    $user = new User();
    $data = $request->all();
    $data["password"] = Hash::make($request->password);
    $user->create($data);
    $credentials = request(['email', 'password']);

    if (Auth::attempt($credentials)) {
        $user = Auth::user();
        $token = $user->createToken('Personal Access Token')->accessToken;
        $cookie = $this->getCookieDetails($token);
        return response()->json([
            'user' => $user,
            'access_token' => $token,
        ], 200)
        ->cookie($cookie['name'], $cookie['value'], $cookie['minutes'], $cookie['path'], $cookie['domain'], $cookie['secure'],
        $cookie['httponly'], $cookie['samesite']);
    } else {
        return response()->json(['message' => 'Unauthorized'], 401);
    }
}

function login(Request $request) {
    $request->validate([
        'email' => 'required|string|email',
        'password' => 'required|string',
    ]);
    $credentials = request(['email', 'password']);
    if (Auth::attempt($credentials)) {
        $user = Auth::user();
        $token = $user->createToken('Personal Access Token')->accessToken;
        $cookie = $this->getCookieDetails($token);
        return response()->json([
            'user' => $user,
            'token' => $token,
        ], 200)
    }
}

```

```

    ->cookie($cookie['name'], $cookie['value'], $cookie['minutes'], $cookie['path'], $cookie['domain'], $cookie['secure'],
$cookie['httponly'], $cookie['samesite']));
  } else {
    return response()->json([
      'error' => 'Помилка авторизації. Некоректний email, або пароль.'
    ], 422);
  }
}

private function getCookieDetails($token) {
  return [
    'name' => '_token',
    'value' => $token,
    'minutes' => 1440,
    'path' => null,
    'domain' => null,
    'secure' => null,
    'httponly' => true,
    'samesite' => true,
  ];
}

function logout(Request $request) {
  $request->user()->token()->revoke();
  $cookie = Cookie::forget('_token');
  return response('ok', 200)->withCookie($cookie);
}

function cropImg($file, $name, $nameSmall, $width, $height, $left, $stop) {
  $image = Image::make($file->getRealPath());
  $imageSmall = Image::make($file->getRealPath());
  $image->resize(800, 1200, function ($constraint) {
    $constraint->aspectRatio();
  }->save(public_path() . '/' . $name);
  $imageSmall->crop($width, $height, $left, $stop)->resize(300, 300, function ($constraint) {
    $constraint->aspectRatio();
  }->save(public_path() . '/' . $nameSmall);
}
}

```

Index.vue

КОМПОНЕНТ ГОЛОВНОЇ СТОРІНКИ.

```

<template>
<div>
  <div class="block-1">
    <b-container style="position: relative">
      <div class="content">
        <div class="subtitle">
          На цьому сайті знаходиться
        </div>
        <div class="title">
          якісний Дизайн
        </div>
        <div class="subtitle mt-4">
          Саме ми можемо гарантувати, що робота<br>буде виконана досконально
        </div>
        <div class="mt-5">
          <Button title="Заповнення заявки" link="/order"></Button>
        </div>
      </div>
    </b-container>
  </div>
</div>

```



```

</div>
<div class="man">
  
</div>
</b-container>
<div class="bottom"></div>
</div>

```

```

<div class="block-2">
  <b-container>
    <b-row>
      <b-col md="6" lg="6" xl="6" xs="12" class="d-none d-md-block">
        
      </b-col>
      <b-col md="6" lg="6" xl="6" xs="12">
        <div class="title">ЧОМУ <span>МИ?</span></div>
        <div class="steps">
          <div class="step-item">
            <h3>Якісне обслуговування</h3>
            <p>
              Ми здатні Вас здивувати. Замовте створення сайту під ключ і Ви побачите як студії дійсно повинні
              підходити до цієї роботи. Обговоримо, запропонуємо, покажемо та візьмемося за роботу.
            </p>
          </div>

```

```

          <div class="step-item">
            <h3>Кваліфіковані спеціалісти</h3>
            <p>
              Наші фахівці якісно вивчать тематику Вашого сайту, Ваших послуг і на основі цього сформуують
              правильну тактику і ключові слова для просування сайту в пошуковій системі Google.
            </p>
          </div>

```

```

          <div class="step-item">
            <h3>Якісна підтримка</h3>
            <p>
              Вам не потрібно витратити час, Ви з легкістю можете звернутися до нас, а ми все зробимо в кращому
              вигляді.
            </p>
          </div>

```

```

</div>
</b-col>
</b-row>
</b-container>
</div>

```

```

<div class="block-3">
  <div class="title text-center mt-5">Приклади Робіт</div>
  <div class="works" id="works">
    <ul>
      <li v-for="item in data" :key="item.id">
        <div class="title">{{ item.title }}</div>
        <div class="description">{{ item.description }}</div>
        <div class="footer">
          <span>{{ item.year }}</span>
          <span>{{ item.work_type.title }}</span>
          <div class="filler"></div>
          {{ item.created_at }}
        </div>
      </li>

```

```

    </ul>
  </div>
  <div class="link">
    <Button title="Всі проекти" link="/portfolio"></Button>
  </div>
</div>
<div class="block-4">
  <b-container>
    <div class="title text-center mt-5">ВСЬОГО <span>КІЛЬКА КРОКІВ</span></div>
    <b-row class="numbers">
      <b-col class="text-center">
        <div class="round">
          01
        </div>
        <div class="title">
          Створення заявки
        </div>
        <div class="subtitle">
          Замовте створення сайту під ключ і Ви побачите як студії дійсно повинні підходити до цієї роботи
        </div>
      </b-col>

      <b-col class="text-center">
        <div class="round">
          02
        </div>
        <div class="title">
          Виконання проекту
        </div>
        <div class="subtitle">
          Наші фахівці з усією відповідальністю підходять до виконання поставлених задач
        </div>
      </b-col>

      <b-col class="text-center">
        <div class="round">
          03
        </div>
        <div class="title">
          Підтримка
        </div>
        <div class="subtitle">
          Кожен клієнт має гарантовану підтримку проектів
        </div>
      </b-col>
    </b-row>
  </b-container>
</div>
</div>
</template>
<script>
import Button from '../components/Button';
export default {
  components: {
    Button
  },
  mounted() {
    jQuery(function ($) {
      $.fn.hScroll = function (amount) {
        amount = amount || 120;

```

```

$(this).bind("DOMMouseScroll mousewheel", function (event) {
    var oEvent = event.originalEvent,
        direction = oEvent.detail ? oEvent.detail * -amount : oEvent.wheelDelta,
        position = $(this).scrollLeft();
    position += direction > 0 ? -amount : amount;
    $(this).scrollLeft(position);
    event.preventDefault();
})
});
$(document).ready(function() {
    $('#works').hScroll(60);
});
let touchstartX = 0
let touchendX = 0
const slider = document.getElementById('works')

function handleGesture() {
    var position = $("#works").scrollLeft();
    if (touchendX < touchstartX) {
        position += touchendX*2;
    }
    if (touchendX > touchstartX) {
        position -= touchendX*2;
    }
    $('#works').animate({
        scrollLeft: position,
    }, 500, function() {
        //код по завершении анимации
    });
}

slider.addEventListener('touchstart', e => {
    touchstartX = e.changedTouches[0].screenX
})

slider.addEventListener('touchend', e => {
    touchendX = e.changedTouches[0].screenX
    handleGesture()
})

},
data() {
    return {
        data: []
    }
},
created() {
    this.getData();
},
methods: {
    getData() {
        axios.get('/api/portfolio')
            .then((response) => {
                this.data = response.data;
            })
    },
}
}
}
</script>

```