

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК

ВИПУСКНА РОБОТА

на тему:

«Інформаційна система керування онлайн рекламою»

**Завідувач
випускаючої кафедри**

Довбиш А.С.

Керівник роботи

Шелехов І.В.

Студента групи ІН-71

Коломойцев М.С.

СУМИ 2021

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра комп'ютерних наук

Затверджую _____

Зав. кафедри Довбиш А.С.

“ _____ ” _____ 20__р.

ЗАВДАННЯ

до випускної роботи

Студента п'ятого курсу, групи ІН71 спеціальності “Комп'ютерні науки”
денної форми навчання Коломойцева Максима Сергійовича.

Тема: “ Інформаційна система керування онлайн рекламою ”

Затверджена наказом по СумДУ

№ _____ от _____ 20__р.

Зміст пояснювальної записки: 1) аналіз проблеми та постановка задачі; 2)
вибір методу розв'язання задачі; 3) розробка інформаційного і програмного
забезпечення

Дата видачі завдання “ _____ ” _____ 20__р.

Керівник випускної роботи _____ Шелехов І.В.

Завдання прийняв до виконання _____ Коломойцев М.С.

РЕФЕРАТ

Записка: 70 стор., 73 рис., 5 табл., 2 додатка, 27 бібліографічних джерел.

Об'єкт дослідження – процес проектування інформаційної системи керування онлайн рекламою.

Мета роботи – розробка інформаційної системи керування онлайн рекламою.

Методи дослідження – методи проектування інформаційних веб-орієнтованих систем, методи нормалізації структури баз даних.

Результати – проведено аналіз обраної предметної області, виконано огляд останніх актуальних досліджень та сервісів, аналіз аналогів, сформовано головну мету та задачі дослідження, проведено вибір засобів реалізації інформаційної системи, описано основні етапи проектування системи та бази даних, виконано реалізацію та детальний опис використання інформаційної системи зі сторони користувача та адміністратора.

ІНФОРМАЦІЙНА СИСТЕМА, ПРОЕКТ, СЕРВІС, РЕКЛАМА, ЗАМОВЛЕННЯ,
ЗАВКА

ЗМІСТ

ВСТУП.....	5
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	7
1.1 Сучасний стан та перспективи розвитку	7
1.2 Постановка задачі.....	10
2 ВИБІР МЕТОДУ РОЗВ'ЯЗАННЯ ЗАДАЧІ	12
2.1 Структура та функції інформаційної системи	12
2.2 Вибір програмних засобів реалізації інформаційної системи.....	13
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ	22
3.1 Розробка дизайну	22
3.2 Використання інформаційної системи зі сторони користувача.....	22
3.3 Адміністрування сайту	30
ВИСНОВКИ.....	38
СПИСОК ЛІТЕРАТУРИ.....	39
ДОДАТОК А. МАКЕТИ СТОРІНОК ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	42
ДОДАТОК Б. КОД РЕАЛІЗАЦІЇ ДОДАТКУ.....	49

ВСТУП

Легко уявити, що ваші зусилля з органічного маркетингу, як-от пропозиції блогу та контенту, та ваші оплачувані рекламні зусилля, як-от пошукові та соціальні, є окремими один від одного.

Маркетологи соціальних мереж зосереджені на побудові спільнот, обміні контентом та залученні своєї аудиторії. Блогери та маркетологи контенту оптимізують свої зусилля для звичайного пошуку та зосереджуються на таких показниках, як трафік та конверсія.

Тим часом рекламодавці зосереджені на таких показниках, як ціна за придбання та рентабельність витрат на рекламу. З такими різними цілями не дивно, що багато маркетологів розглядають органічні та оплачувані продукти як окремі один від одного.

Що, якби я сказав вам, що органічний та платний маркетинг можуть працювати не лише разом, але й те, що цифрова реклама може допомогти вам покращити ваші органічні зусилля? Це правда.

Існує три ключових способи, за допомогою яких цифрова реклама може допомогти вам покращити ефективність ваших органічних маркетингових зусиль.

Однак з іншого боку спектру технологія медійної реклами досягла рівня, коли рекламні мережі можуть використовувати дані та машинне навчання, щоб запропонувати рекламодавцям ефективніші стратегії націлювання, а споживачам більш релевантну рекламу.

Такі рекламні мережі, як Медійна мережа Google та Мережа аудиторій Facebook, є лідерами у відродженні рекламних банерів. Вони можуть показувати ваші оголошення правильній цільовій аудиторії в потрібному місці та в потрібний час. І якщо ви хочете отримати більший контроль над своєю рекламою, вони дозволять вам вирішити, де розмістити ваші оголошення.

Для швидкого та якісного вирішення поставленої проблеми було вирішено розробити сайт. Для надання зручної комунікації та надання якісних послуг у сфері

реклами. Для реалізації поставленої мети, перш за все, потрібно вирішити такі задачі:

- Ознайомитися із різновидами реклами.
- Провести аналіз сайтів-аналогів, що також надають послуги у сфері реклами.
- Визначити типи та категорії послуг.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Сучасний стан та перспективи розвитку

1.1.1. Основні тенденції реклами в Інтернеті

Інтернет-магазини та перегляди були більш поширеними, ніж раніше - 87% домогосподарств зробили покупки через Інтернет цього року порівняно з 82% у 2019 році - просто тому, що ми не могли покинути свої домівки [1]. Ті, хто вдосконалював свої рекламні стратегії в Інтернеті, процвітали, які ж навички було використано, буде розглянуто в табл.1.1.

Таблиця 1.1 – Тенденції реклами в 2021 році

№	Назва тенденції	Опис
1	Креативна оптимізація	Ці оптимізовані оголошення також виконують роботу з точки зору розташування. Даючи змогу визначити, де ваші клієнти пропонують більш цілеспрямовану рекламу. Плюс, навіть охопити ширшу аудиторію, ніж ви, можливо, вважали.
2	Реклама в грі	Внутрішньо-ігрова реклама дозволяє брендам потрапляти перед ширшу аудиторію, ніж будь-коли раніше. Незважаючи на те, що можуть бути аргументи щодо контекстного націлювання деяких з цих рекламних оголошень, безсумнівно, це щось, що слід розглянути, реалізуючи [2]. Ви можете охопити ширшу демографічну групу від мам до підлітків, які навчаються вдома!
3	Контекстне націлювання	Це тенденція, яка навряд чи зникне. Причиною є те, що ніколи не було важливішим, щоб ваша реклама потрапляла до потрібної аудиторії.

№	Назва тенденції	Опис
4	Більша персоналізація	Персоналізовані оголошення націлені на користувача на основі його демографічних показників та місцезнаходження [3]. Хороший приклад цього створив бренд Гіннеса, який застосував гіпер-локалізований підхід та рекламну кампанію, яка намагалася спонукати користувачів насолоджуватися пінтою у своєму місцевому пабі під час перегляду регбі.
5	Реклама на одній платформі	<p>Впорядкований підхід до всіх маркетингових каналів буде як ніколи важливим у 2021 році. Тут з'являється реклама на одній платформі. Це дозволяє створювати кампанії, що охоплюють усі ваші ЗМІ та пропонують одне і те ж повідомлення.</p> <p>Користувачі хочуть насолоджуватися однаковим рівнем обслуговування та якості, незалежно від того, відвідують цегельно-будівельний магазин чи роблять покупки в Інтернеті [4]. Це означає, що ваша рекламна кампанія повинна працювати в різних форматах.</p>
6	Agile	2020 рік вимагав, щоб маркетологи були більш спритними, ніж будь-коли раніше, адаптуючи свою рекламу до відображення поточної ситуації. Ми бачили, як багато брендів - а саме банки та будівельні товариства, такі як Natwest - створюють нову рекламу в домашніх ситуаціях, показуючи справжнім працівникам за столами вдома, що безпосередньо спілкуються зі споживачами про те, як вони можуть допомогти [5]. Це виявило сирішу, більш людську сторону бренду та поділило ідею, що ми всі в цьому разом.

Рік, що попереду, вимагає від нас адаптації як маркетологів, ставлячи споживачів на перше місце та гарантуючи, що ми як торгові марки пропонуємо їм досвід, який вони хочуть і потребують. Орієнтуючись на персоналізацію та контекстуалізацію, ми повинні мати змогу побудувати міцні стосунки та репутацію з тими, кого ми хочемо як своїх клієнтів.

1.1.2. Загальні витрати

Зростаючий вибір цифрових каналів та реклами в Інтернеті брендами, звичайно, також надихається іншими мотиваціями; наприклад, за рівнем витрат та потребою, яку багато організацій тепер мають отримувати ще швидші результати своїх зусиль.

Як це було в попередні кризи, брендам рекомендували не припиняти рекламу, а, натомість, продовжувати спілкуватися і, безумовно, також дивитись на довгострокову перспективу зростання. Однак не кожен може дозволити собі інвестувати в брендинг, і для багатьох швидких результатів завдяки перевіреним методам це єдиний варіант [6].

Хоча сильні бренди, які можуть це зробити завдяки своїй репутації та своїй діяльності, переглядають свої рекламні повідомлення / розповіді, оскільки вони мають намір продовжувати рекламувати, і тут і там, навіть активізувати свої зусилля завдяки гарному поєднанню між продажами та активацією маркетингу з одного боку з іншого боку, ми, безумовно, спостерігаємо зміну тактики та каналів. Можливо, не буде несподіванкою, що в контексті реклами інтернет-реклама буде найкращим способом витратити свої рекламні бюджети для брендів.

Незважаючи на заклики продовжувати інвестувати в брендинг та довгострокове зростання, коли це можливо, глобальна рекламна індустрія неминуче зазнає значного падіння світових доходів від реклами, а Інтернет-реклама взагалі не компенсує скорочення бюджету в інших ЗМІ та каналах.

1.2 Постановка задачі

Компанії витрачають на цифрову рекламу більше, ніж будь-коли раніше, своїх маркетингових бюджетів, і всі хочуть отримати певний результат. Існує багато різних способів створити агентство з цифрового маркетингу. Послуги, які надаватимуться, і спосіб оплати роботи становлять важливу частиною ефективності управління вашим бізнесом з часом.

У результаті, було обрано створити інформаційну систему керування онлайн рекламою, використовуючи метод із фіксованою ціною. Окрім простоти, це дозволяє зменшити будь-яке тертя, коли настає час надіслати рахунок-фактуру. Клієнт точно знає, скільки йому це коштуватиме, і якщо ви виправдаєте його очікування, їм не складе труднощів сплатити його. Головні цілі представлені в таблиці 1.2.

Таблиця 1.2 – Цілі інформаційної системи керування онлайн рекламою

№	Назва тенденції	Опис
1	Набути навички оптимізації пошукової системи	Важливо володіти цими навичками, оскільки ви отримаєте більше рекламних клієнтів, коли ваш веб-сайт має високий рейтинг у пошукових системах та приймає багато відвідувачів [7-9].
2	Створіть онлайн-каталог	Компанії використовують онлайн-довідники як спосіб реклами свого бізнесу, як це було б у телефонній книзі.
3	Створіть веб-сайт	Нішевий веб-сайт - це сайт, який обслуговує певну тему. міру зростання популярності вашого сайту ви можете пропонувати рекламний простір протягом певного періоду часу. Люди, які найбільше зацікавлені в рекламі на вашому сайті, матимуть справи, пов'язані з вашою нішею.

№	Назва тенденції	Опис
4	Послуги з копірайтингу	Якщо у вас є досвід роботи з копірайтерами, пропонуйте свої послуги багатьом компаніям, що працюють в Інтернеті та в режимі офлайн. Інтернет-маркетологи залежать від копірайтерів, щоб вони писали копії для продажів своїх інформаційних продуктів [10-12]. Замовники в режимі офлайн також залежать від копірайтерів для написання реклами, бюлетенів, брошур та інших маркетингових продуктів.
5	Встановіть свої рекламні ціни.	Ціни, які ви стягуєте за рекламу, залежатимуть від послуг, які ви пропонуєте, і тривалості часу, коли ви пропонуєте ці послуги.

2 ВИБІР МЕТОДУ РОЗВ'ЯЗАННЯ ЗАДАЧІ

2.1 Структура та функції інформаційної системи

Веб-структура - це спосіб організації та взаємозв'язку інформації на веб-сайті. Ефективна структура сайту покращує зручність використання та взаємодію з користувачем, що робить структурування веб важливим кроком у процесі веб-дизайну.

Основним принципом чудової структури веб-сайтів є Інформаційна архітектура. Це забезпечує, щоб вміст був організований, структурований та маркований ефективно та послідовно [13]. Щоб створити найкращу інформаційну архітектуру для свого веб-сайту, потрібно врахувати наступні фактори:

- Оскільки веб-сайти створені для обслуговування користувачів, важливо врахувати, як вони можуть відчувати або взаємодіяти з сайтом, а також їхні очікування щодо того, як веб-сайт повинен працювати.

- Структура веб-сайту також буде значною мірою визначатися типом та обсягом вмісту на сайті. Структура сайту електронної комерції буде відрізнятися від структури академічного сайту.

- Контекст веб-сайту визначається його бізнес-цілями, культурним контекстом, в якому він існує, та наявними ресурсами. Важливо враховувати цей факт під час структуризації веб-сайту.

Найпоширеніша структура веб-сайту - це ієрархічна структура, яка базується на одній батьківській сторінці (головна сторінка) та дочірніх сторінках (категорії та підкатегорії), що впливають із головної сторінки [14].

Інформаційна система керування онлайн рекламою повинна бути реалізований у вигляді веб-додатку. Рекламний сайт повинен пер за все складатися із взаємозалежних розділів із розділеними функціями в обраній сфері.

Інформаційна система повинна мати наступні можливості:

- Перегляд повної інформації про сервіс.
- Можливість реєстрації та авторизації користувачів.
- Залишати відгуки.

- Перегляд контактної інформації.
 - Залишення заявки користувачем про надання послуг у сфері реклами.
- Веб-додаток, повинен володіти такими особливостями як:
- Функціональний інтерфейс із зрозумілим та зручним інтерфейсом.
 - Веб-сервіс повинен підтримувати графічні вставки, та анімації.
 - Сайт повинен мати користувацьку та адміністративну частин

(табл.2.1).

Таблиця 2.1 – Цілі інформаційної систему керування онлайн рекламою

Користувацька сторінка	Адміністративна сторінка
– Головна.	– Головна.
– Послуги.	– Послуги.
– Приклади робіт.	– Додання робіти.
– Відгуки.	– Нові відгуки.
– Контакти.	– Загальна інформація.

2.2 Вибір програмних засобів реалізації інформаційної системи

Розробка веб-сайтів – це процес створення веб-сайтів та додатків для інтернету або для приватної мережі, відомий як інтернет. Розробка веб-сайтів не стосується дизайну веб-сайту; швидше, це все про кодування та програмування, що забезпечує функціональність веб-сайту.

Від найпростіших, статичних веб-сторінок до платформ і додатків соціальних медіа, від веб-сайтів електронної комерції до систем управління контентом (CMS); всі інструменти, якими ми користуємось через інтернет щодня, були розроблені веб-розробниками [15].

Веб-розробка може бути розбита на три шари (табл. 2.2): кодування на стороні клієнта (фронтенд), кодування на стороні сервера (бекенд) та технології баз даних.

Таблиця 2.2 – Частина веб розробки

№	Назва	Опис	Приклад технологій розробки
1	Клієнтська сторона	<p>Сценарії на стороні клієнта або розробка інтерфейсу стосується всього, що переживає кінцевий користувач безпосередньо.</p> <p>Код клієнта виконується у веб-браузері і безпосередньо стосується того, що бачать люди, відвідуючи веб-сайт [16]. Такі речі, як компонування, шрифти, кольори, меню та контактні форми, керуються фронтом.</p>	HTML, CSS, Vue.js, Vuetify, JavaScript, React.js, Bootstrap.
2	На стороні сервера	<p>Сценарії на стороні сервера або розробка бекенду – це все, що відбувається за кадром. Бекенд – це по суті частина веб-сайту, яку користувач насправді не бачить [17]. Він несе відповідальність за зберігання та впорядкування даних та забезпечує те, що все на стороні клієнта працює безперебійно.</p>	PHP, Laravel, Node.js, Python, Express.js, Docker.
3	Технологія баз даних	<p>Веб-сайти також покладаються на технологію баз даних. База даних містить усі файли та вміст, необхідний веб-сайту для функціонування, зберігаючи його таким чином, щоб полегшити його роботу. База даних працює на сервері, і більшість веб-сайтів зазвичай використовують певну форму управління реляційною базою даних</p>	MySQL, PostgreSQL, OracleSQL, MongoDB

2.2.1 Розробка інтерфейсу

Завдання розробника frontend – кодувати інтерфейс веб-сайту чи програми; тобто частина веб-сайту, яку бачить і взаємодіє користувач. Вони беруть дані і перетворюють їх у те, що легко зрозуміле, візуально приємне та повністю функціональне для повсякденного користувача. Вони працюють над розробками, наданими веб-дизайнером, і реалізують їх за допомогою HTML, JavaScript та CSS [18].

Мови розмітки використовуються для визначення форматування текстового файлу. Іншими словами, мова розмітки повідомляє програмне забезпечення, яке відображає текст, як слід форматувати текст. Мови розмітки є повністю розбірливими для читання – вони містять стандартні слова, але теги розмітки приховані від користувача в кінцевому варіанті роботи.

Дві найпопулярніші мови розмітки – HTML та XML. HTML розшифровується як мова розмітки НуретТекст та використовується для створення веб-сайтів. Додані до звичайного текстового документа, всі теги HTML описують, як цей документ повинен відображатися веб-браузером.

Невідмінна частина будь-якої веб сторінки – каскадні стилі розмітки сторінки або CSS. Стилi – це в основному сукупність стилістичних правил. Мови аркушів стилів використовуються, доволі буквально, для оформлення документів, написаних мовами розмітки [19].

CSS розшифровується як каскадні таблиці стилів з акцентом на «стиль». Хоча HTML використовується для структури веб-документа, визначає такі речі, як заголовки та абзаци, і дозволяє вставляти зображення, відео та інші медіа, CSS визначає стиль створеного документа.

2.2.2 Розробка серверної частини

Код, який створюють розробники бекенду, гарантує, що все, що будує розробник frontend, є повністю функціональним. Головне завдання розробника серверної частини – переконатися, що сервер, програма та база даних взаємодіють

між собою. Для вирішення такої складної задачі розробники використовують серверні мови, такі як PHP, Ruby, Python та Java для створення програми.

Веб-технології стрімко розвиваються, і щоб встигати за швидкістю розвитку нових технологій, у веб-програмістів часто бракує часу для реалізації повсякденних завдань з нуля, таких як: автентифікація, API, тестування, дебаг, кешування і т.д. Саме для спрощення розробки сайтів і швидкого вирішення цих завдань були придумані веб-фреймворки.

Фреймворк – це каркас, призначений для створення динамічних веб-сайтів, веб-додатків, служб або ресурсів. Він спрощує розробку і позбавляє від необхідності писати звичайний код [20]. Фреймворк спрощує доступ до бази даних, розробку інтерфейсу та зменшує дублювання коду.

Розглянемо один з популярних фреймворків Yii 2. Yii – це універсальний фреймворк, і він може бути задіяний у всіх типах веб-додатків.

Завдяки своїй складовій структурі та чудовій підтримці кешу, фреймворки особливо підходять для розробки великих проектів, таких як портали, форуми, CMS, магазини або програми RESTful. Yii – це командний проект[21-23]. Він підтримується і розвивається сильною командою та великою спільнотою розробників. Автор цього фреймворку стежить за тенденціями розвитку веб-сайтів та інших проектів.

Найбільш підходящі можливості і кращі практики регулярно впроваджуються в фреймворк в вигляді простих і елегантних інтерфейсів:

- Як і багато інших PHP фреймворків, для організації коду Yii використовує архітектурний патерн MVC.

- Yii дотримується філософії простого і елегантного коду не намагаючись ускладнювати дизайн тільки заради проходження будь-якими шаблонами проектування.

- Yii є full-stack фреймворком і включає в себе перевірені і добре зарекомендовані в себе можливості: ActiveRecord для реляційних і NoSQL баз даних, підтримку REST API, багаторівневе кешування та інше.

- Yii відмінно масштабований. Можна налаштувати або замінити практично будь-яку частину основного коду [24]. Використовуючи архітектуру розширень, легко ділитися кодом або використовувати код спільноти.

- Одна з головних цілей Yii – продуктивність.

Розглянемо також не менш популярний фреймворк Laravel. Laravel – доступний, але потужний. Має безліч відмінних інструментів для великих, надійних додатків:

- Як і багато інших PHP фреймворків, для організації коду Laravel використовує архітектурний патерн MVC.

- Laravel дотримується філософії виразного, красивого синтаксису. Він призначений для людей, які цінують елегантність, простоту і читаність.

- Функціонал Laravel є простим для розуміння і використання.

- Більшість функцій Laravel чудово працюють, не вимагаючи додаткових налаштувань. Вони спираються на загальноприйняті стандарти написання коду, роблячи його інтуїтивно зрозумілим.

- Документація Laravel закінчена і постійно оновлюється.

- Чудова IoC (Інверсія управління).

- Зручна система міграцій.

- Інтегрована система модульного тестування.

Обидва фреймворка для організації коду використовують архітектурний патерн MVC. При детальному розгляді ці фреймворки складаються з трьох основних компонентів.

За архітектурою дані фреймворки ідентичні, так як реалізують один і той же архітектурний патерн. Він дозволяє швидко реалізувати проект, а також легко супроводжувати і масштабувати його в подальшому.

Розширення важлива частина фреймворку, тому розглянемо їх докладніше. Розширення – це можливість підключати додаткові модулі або бібліотеки для фреймворка, які дозволяють розширити його можливості. Yii і Laravel підтримують таку можливість [25]. На даний момент, розширень набагато більше, звичайно ж, у

Yii, так як фреймворк живе набагато довше, ніж Laravel, однак другий, в свою чергу, розвивається дуже великими темпами і включає в себе багато всього для роботи відразу «з коробки».

Розширення є абсолютно для будь-якого завдання, будь то панель адміністратора або ж платіжна система.

Для належної роботи сайту важлива підтримка REST API. REST – це архітектурний стиль взаємодії між компонентами розподілених додатків у мережі. REST – це набір послідовних обмежень, які слід враховувати при проектуванні розподіленої системи гіпермедіа. У деяких випадках (інтернет-магазини, пошукові системи, інші системи на основі даних) це може підвищити продуктивність та спростити архітектуру. Загалом, компоненти REST взаємодіють із клієнтами та серверами у всесвітній мережі.

В Інтернеті віддалені процедурні виклики можуть бути звичайними HTTP-запитами (зазвичай "GET" або "POST"; це називається "REST-запит"), а необхідні дані передаються як параметри запиту.

Для веб-служб, які відповідають REST тобто, не порушуючи жодних обмежень, що застосовуються до нього, використовується термін "RESTful". На відміну від веб-служб на основі SOAP, веб-API RESTful не має "офіційного" стандарту. Справа в тому, що REST – це архітектурний стиль, а SOAP – протокол. Хоча сам REST не є стандартом, більшість реалізацій RESTful використовують стандарти HTTP, URL, JSON та XML.

На основі проведеного аналізу був зроблений вибір на сторону, що активно розвивається Laravel, його структура дозволяє легко масштабувати веб-додаток. Yii є більш стабільним, але менш масштабованим.

2.2.3 Розробка бази даних

Для розробки бази даних використовують такі інструменти, як MySQL, Oracle чи SQL Server, щоб знаходити, зберегти або відредагувати дані та повернути їх користувачеві за допомогою коду візуальної частини.

Вище перелічені мови використовуються не лише для створення веб-сайтів, програмного забезпечення та додатків; вони також використовуються для створення та управління базами даних.

Бази даних не розроблені для розуміння тих же мов, на яких запрограмовані програми, тому важливо мати мову, яку вони розуміють – як SQL, стандартну мову для доступу та маніпулювання реляційними базами даних. SQL означає структурована мова запитів.

Вона має власну розмітку і в основному дозволяє програмістам працювати з даними, що зберігаються в системі баз даних. Розглянемо детальніше різні типи, переваги та недоліки баз даних (табл.2.3).

Таблиця 2.3 – Бази даних

Назва	Переваги	Недоліки
Oracle	Інструменти управління базами даних Oracle також надзвичайно надійні, і ви можете знайти той, який може робити практично все, про що ви можете подумати.	Після встановлення система може вимагати значних ресурсів, тому для впровадження Oracle може знадобитися оновлення обладнання.
	Має останні інновації та функції, що надходять від їх продукції, оскільки Oracle прагне встановити планку для інших інструментів управління базами даних.	Вартість Oracle може бути непосильною, особливо для менших організацій.
MySQL	Безкоштовний.	Ви можете витратити багато часу і сил, щоб змусити MySQL робити те, що інші системи роблять автоматично.
	Пропонує безліч функціональних можливостей.	Немає вбудованої підтримки для XML або OLAP.
	Існують різноманітні інтерфейси користувачів.	Підтримка доступна для безкоштовної версії, але за неї потрібно заплатити.
	Це може бути зроблено для роботи з іншими базами даних, включаючи DB2 та Oracle.	

Назва	Переваги	Недоліки
Microsoft SQL Server	Це дуже швидко і стабільно.	Ціни на підприємства можуть перевищувати багато організацій.
	Пропонує можливість регулювання та відстеження рівнів продуктивності, що може зменшити використання ресурсів.	Навіть при налаштуванні продуктивності Microsoft SQL Server може обробляти ресурси.
	Є можливість отримати доступ до візуалізації на мобільних пристроях.	Багато людей мають проблеми з використанням інтеграційних служб SQL Server для імпорту файлів.
	Дуже добре працює з іншими продуктами Microsoft.	
PostgreSQL	Ця система управління базами даних є масштабованою і може обробляти терабайти даних.	Не чітка документація.
	Підтримує JSON.	Конфігурація може бути заплутаною.
	Існують різноманітні заздалегідь визначені функції.	Швидкість може постраждати під час великих масових операцій чи запитів.
	Доступна низка інтерфейсів.	
MongoDB	Швидка і проста у використанні.	Установки за замовчуванням не захищені
	Підтримує JSON та інші документи NoSQL.	Налаштування може бути тривалим процесом.
	Дані будь-якої структури можна зберігати та отримувати доступ до них швидко та легко.	Інструменти для перекладу SQL на запити MongoDB доступні, але вони додають додатковий крок до використання системи.

В ході виконання даного проекту, вибір було зроблено на користь MySQL. По-перше, дана СУБД є безкоштовною. По-друге, супутній продукт MySQL Workbench дозволяє просто взаємодіяти зі встановленою БД, надаючи можливість через зручний інтерфейс користувача виконувати будь-які маніпуляції з даними.

Та нарешті, ця СУБД є однією з найбільш розповсюджених, займаючи друге місце за популярністю у світі, поступаючись лише корпоративній СУБД від Oracle. А це означає, що на випадок проблем чи збоїв в роботі цієї системи, в мережі можна знайти величезних кількість матеріалу для їх вирішення.

Для розробки клієнтської частини сайт було обрано мови програмування HTML, CSS і JavaScript так як вони є най популярнішими мовами для веб розробки і досить легкі у вивченні.

Розробка серверної частини буде відбуватись за допомогою мови програмування PHP з використанням фреймворку Laravel. Ця мова активно використовується багатьма сайтами і досить легка у вивченні. Також найбільш популярні системи контролю контентом використовують саме цю мову програмування, що надає можливість розробляти додатковий функціонал.

3 ПРАКТИЧНА РЕАЛІЗАЦІЯ

3.1 Розробка дизайну

Дизайн був розроблений в додатку Figma [26]. Макети представлені у додатку А. Figma використовує найновіші та найкращі практики веб-дизайну та мобільного дизайну, а також інтерфейсу користувача та дизайну користувацького досвіду (UI / UX) [27]. Вей додаток зосереджений на тому, щоб ефективно дістати від нуля до точки, де можна розробити гарний дизайн для майбутнього додатку.

У додатку було розроблено прототипи основних сторінок інформаційної системи по керування онлайн рекламою, а саме:

- головна сторінка;
- приклади робіт;
- останні новини;
- відгуки клієнтів;
- наша команда;
- зв'яжіться з нами;
- оформлення замовлення.

3.2 Використання інформаційної системи зі сторони користувача

На головній сторінці користувач може знайти і переглянути готові проекти, підказку по оформленню замовлення, переваги сайту та без зайвих кліків заповнити заявку (рис.3.1-4).

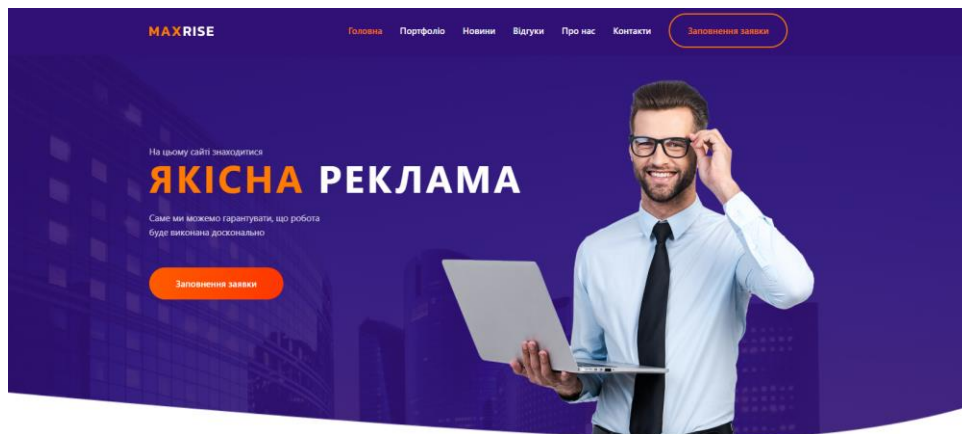


Рисунок 3.1 – Головна сторінка

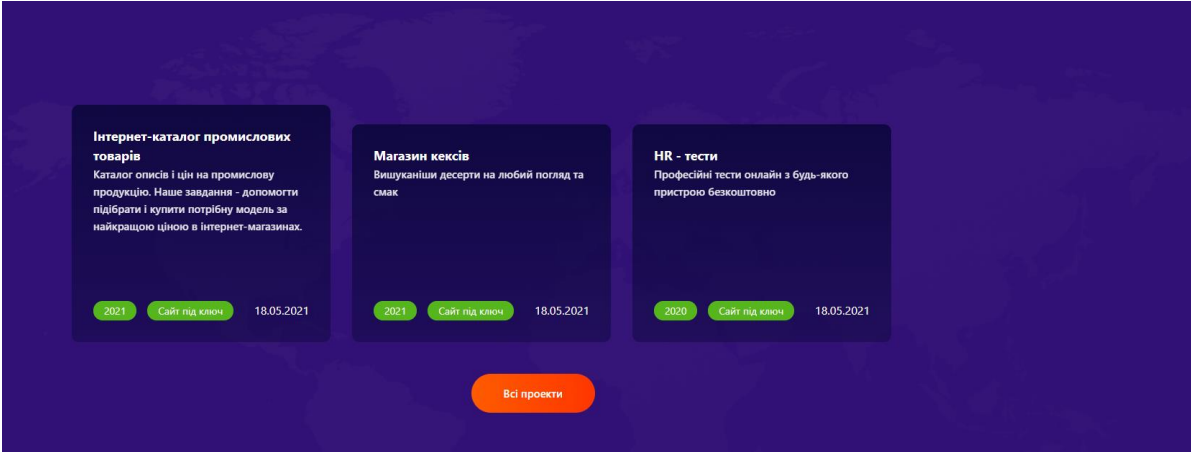


ЧОМУ МИ?

- Якісне обслуговування**
Ми здатні Вас здивувати. Замовте створення сайту під ключ і Ви побачите як студія дійсно повинні підходити до цієї роботи. Обговоримо, запропонуємо, покажемо та візьмемося за роботу.
- Кваліфіковані спеціалісти**
Наші фахівці точно визначать тематику Вашого сайту. Ваші послуги і на основі цього сформулюють правильну тактику і ключові слова для просування сайту в пошуковій системі Google.
- Якісна підтримка**
Вам не потрібно витрачати час, Ви з легкістю можете звернутися до нас, а ми все зробимо в кращому вигляді.



Рисунок 3.2 - Інформаційний блок із само-рекламою сайту



Інтернет-каталог промислових товарів
Каталог описів і цін на промислову продукцію. Наше завдання - допомогти підібрати і купити потрібну модель за найкращою ціною в Інтернет-магазинах.

2021 Сайт під ключ 18.05.2021

Магазин кексів
Вишуканіші десерти на любой погляд та смак


2021 Сайт під ключ 18.05.2021

HR - тести
Професійні тести онлайн з будь-якого пристрою безкоштовно

2020 Сайт під ключ 18.05.2021

[Всі проекти](#)

Рисунок 3.3 – Блок із останніми проектами



ВСЬОГО КІЛЬКА КРОКІВ

01

Створення заявки

Замовте створення сайту під ключ і Ви побачите як студія дійсно повинні підходити до цієї роботи

02


Виконання проекту

Наші фахівці з усією відповідальністю підходять до виконання поставлених задач

03

Підтримка

Кожен клієнт має гарантовану підтримку проектів



MAXRISE
Головна Про нас Оформити заявку

Рисунок 3.4 – Алгоритм подання заявки

Блок портфоліо надає можливість користувачу переглянути усі готові проекти. Як це виглядає можна побачити на рис.3.5-6. Користувач має можливість виконувати фільтрацію.

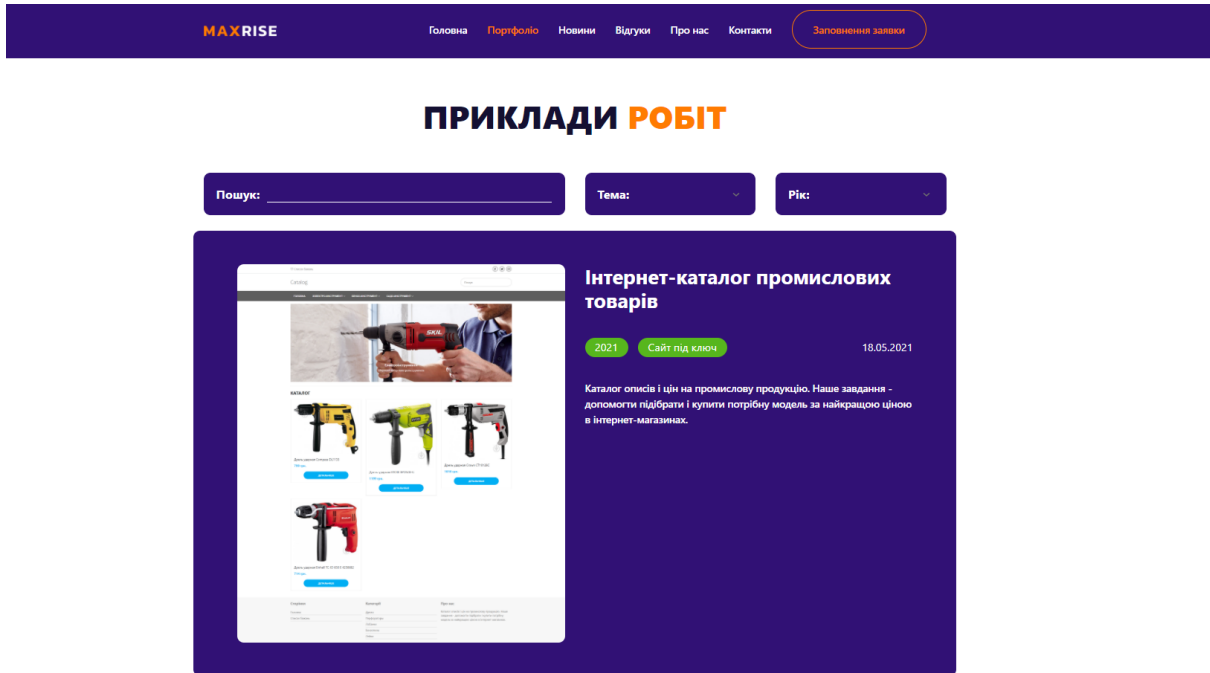


Рисунок 3.5 – Приклад готової роботи

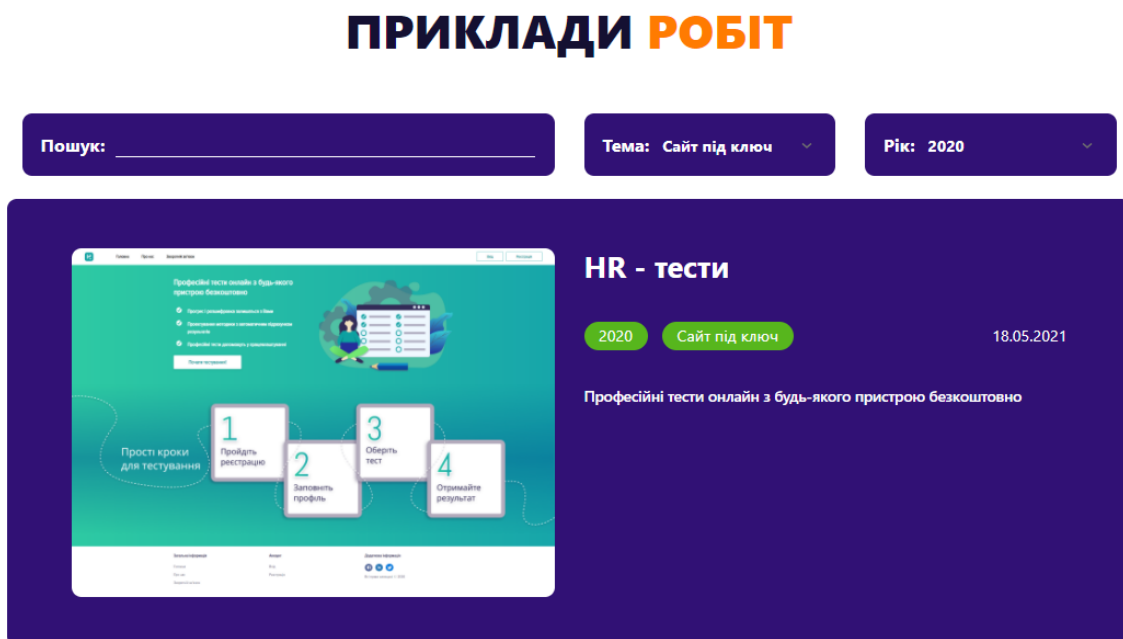


Рисунок 3.6 - Приклад фільтрації

Якщо користувачу надана можливість перейти на сайт та переглянути його, то є прикріплене посилання на нього. Відповідно до цього присутня кнопка, натиснувши на яку можна перейти на реалізацію даного проекту (рис.3.7).

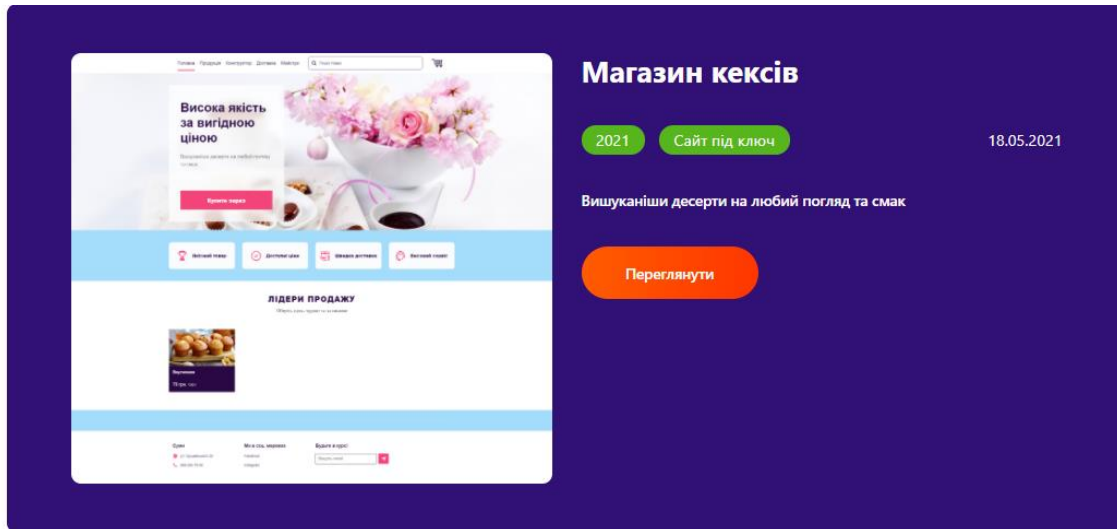


Рисунок 3.7 – Приклад роботи із посиланням

В блоці новин користувач може переглянути новини стосовно оновлень в команді, відкриття офісу, розширення компанії чи про пошук нової людини на певну посаду (рис. 3.8).

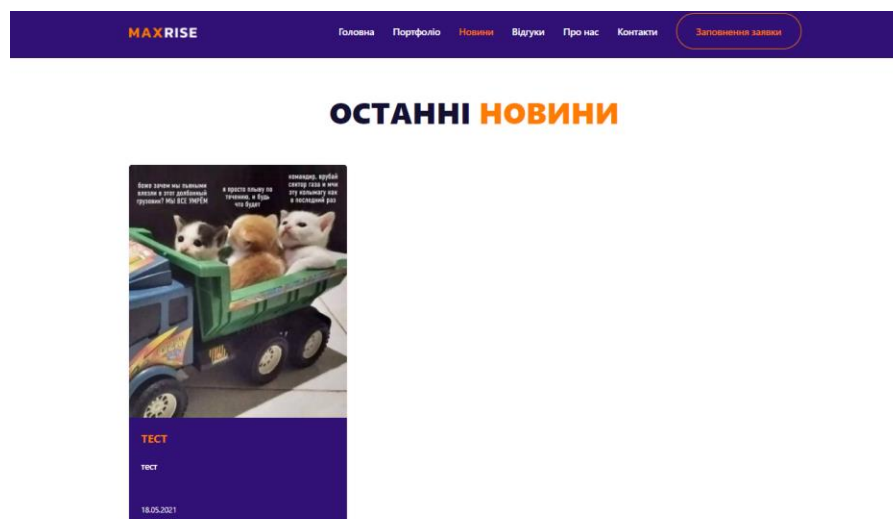


Рисунок 3.8 – Блок новин

Налаштування сайту дають змогу користувачу залишати відгуки про якість роботи, швидкість виконання, обслуговування та тому подібне. На рис.3.9-10 можна побачити як виглядає форма для написання коментаря.



Рисунок 3.9 – Форма залишення коментаря

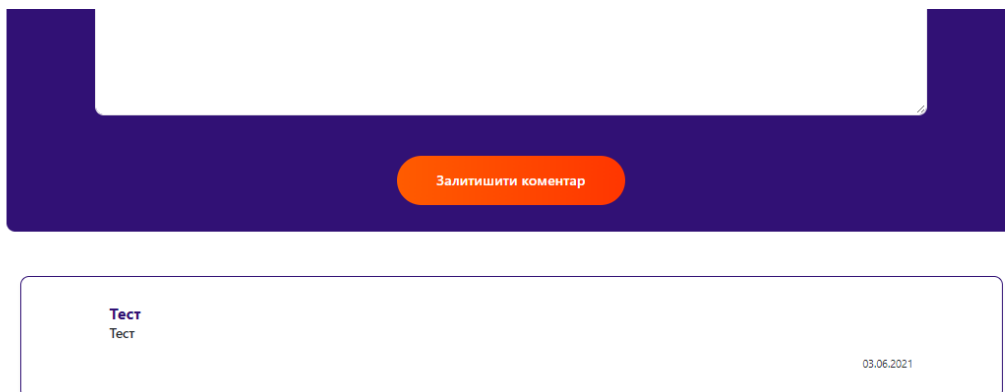


Рисунок 3.10 – Приклад коментаря

У блоці відгуків також передбачена фільтрація. За допомогою неї можна переглянути відгуки за будь-який рік, на різні теми та побачити рейтингові бали (рис.3.11).

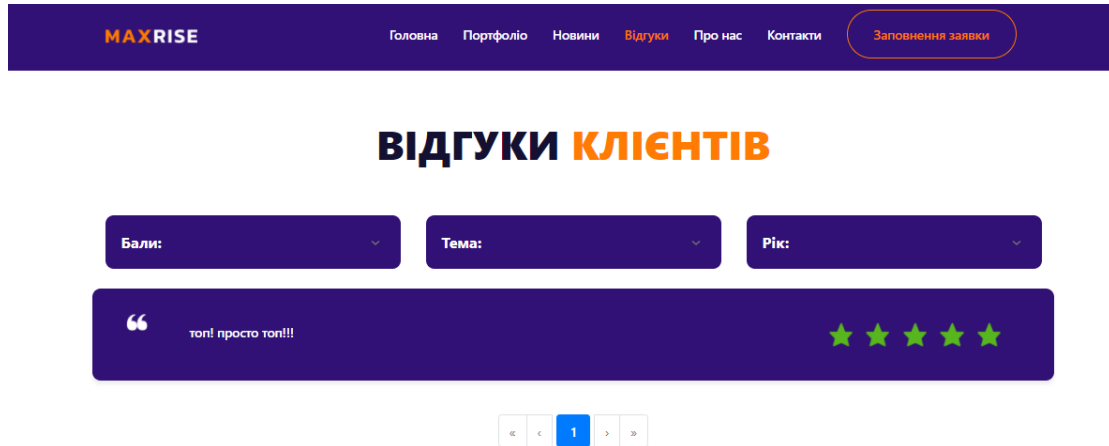


Рисунок 3.11 - Сторінка відгуків

В блоці «КОМАНДА» користувач з легкістю має змогу подивитись склад команди, їх фотографії, професійні навички кожного працівника та деяку інформацію про них (рис 3.12)

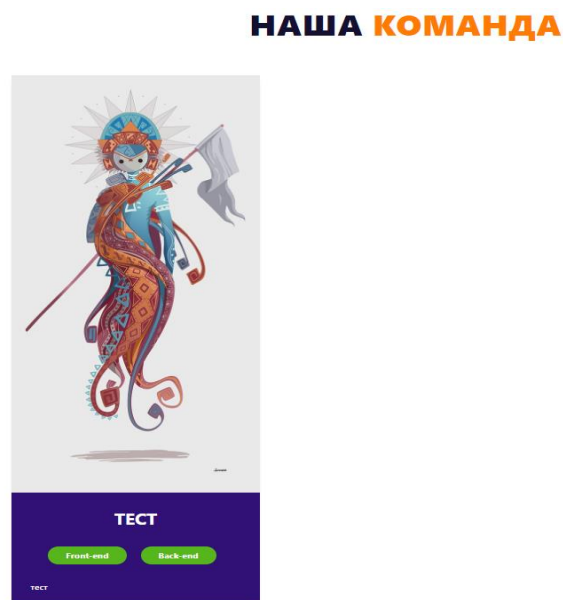


Рисунок 3.12 – Команда працівників

В розділі контактів надається інформація про місцезнаходження офісу, яка також представлена у вигляді геолокації на мапі, електронна адреса та номер телефону, за яким можна зателефонувати адміністратору (рис 1.13).

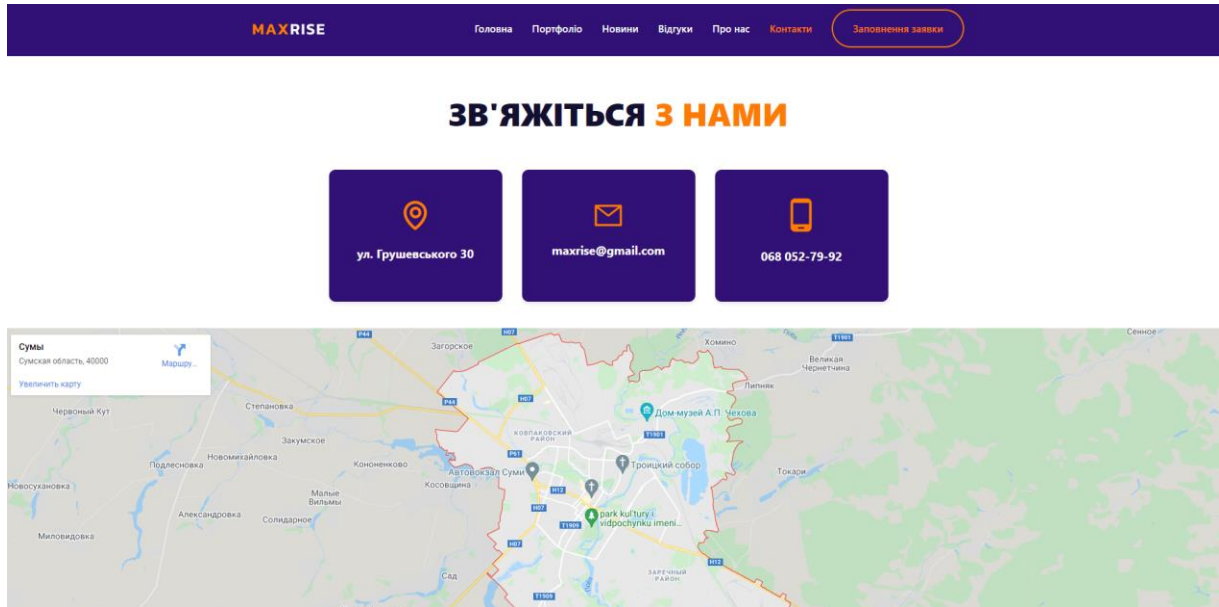


Рисунок 3.13 – Контактна інформація

Оформлення замовлення (рис 3.14) відбувається у вкладці «Заповнення заявки» шляхом заповнення бланку заявки, приклад наведено на рис. 3.15-16.

Рисунок 3.14 – Блок оформлення замовлення

ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

Будь ласка, надайте свою інформацію, і менеджер зв'яжеться з вами

Допомога
Потрібна допомога? Надішліть нам електронного листа безпосередньо на taxi@silniez.com, щоб отримати більше інформації про наші продукти.

Тест Тестович

silniez.group@gmail.com

+380506858992

Сайт під ключ

Я отримав гарні рекомендації, чекаю відповіді!

Оформити заявку

Рисунок 3.15 - Приклад заповнення

ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

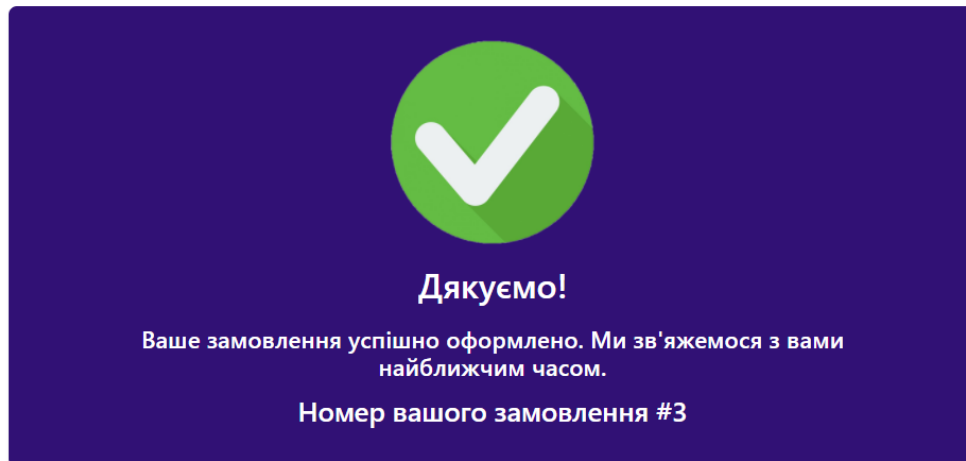
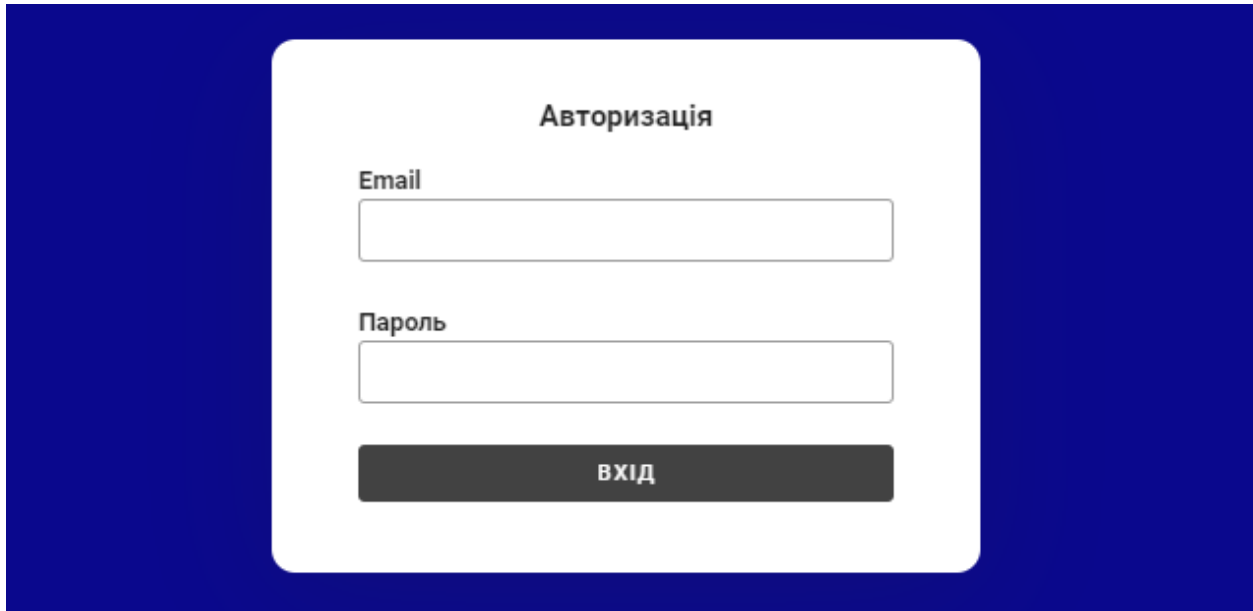


Рисунок 3.16 – Результат оформлення замовлення

3.3 Адміністрування сайту

Для того щоб скористатись веб-сайтом в якості адміністратора необхідно пройти авторизацію (рис. 3.17).



The image shows a login form with the following elements:

- Title: Авторизація
- Field: Email
- Field: Пароль
- Button: ВХІД

Рисунок 3.17 – Авторизація адміністратора

В можливості та обов'язки адміністратора сайту входить перегляд та додавання нових готових проектів, корегування тих, що були створені та завантажені раніше (рис.3.18-19).

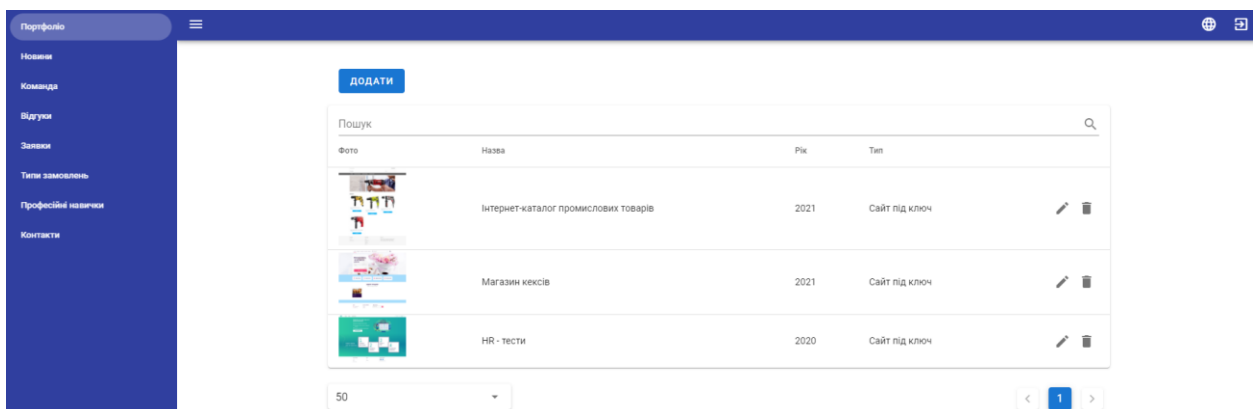


Рисунок 3.18 - Портфоліо

Тип

Назва

Рік

Посилання

Опис

Зображення

ЗБЕРЕГТИ

Рисунок 3.19 - Форма додавання роботи в портфоліо

Адміністратор може змінювати склад команди працівників, що розміщена на сайті, редагувати, змінювати або видаляти інформацію та дані стосовно того чи іншого працівника. Додавати нових працівників та видаляти тих, хто звільнився (рис. 3.20-21).

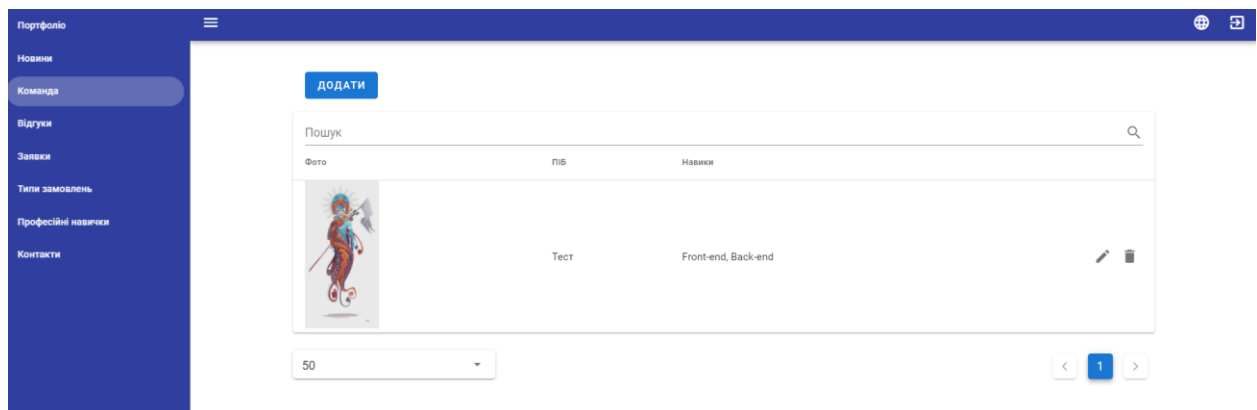


Рисунок 3.20 – Склад команди

Рисунок 3.21- Додавання новачка

Адміністратор має вповноваження видаляти відгуки, якщо вони порушують правила користування послугами (рис.3.22).

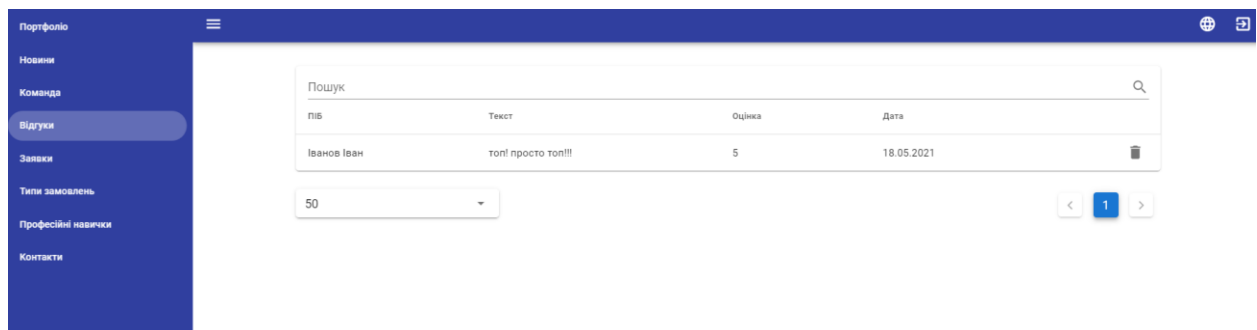


Рисунок 3.22 –Відгуки

Адміністратор переглядає, видаляє давно подані та виконані заявки зі списку, змінює статус нових заявок, що прийняті в обробку чи виконані (рис. 2.23-24).

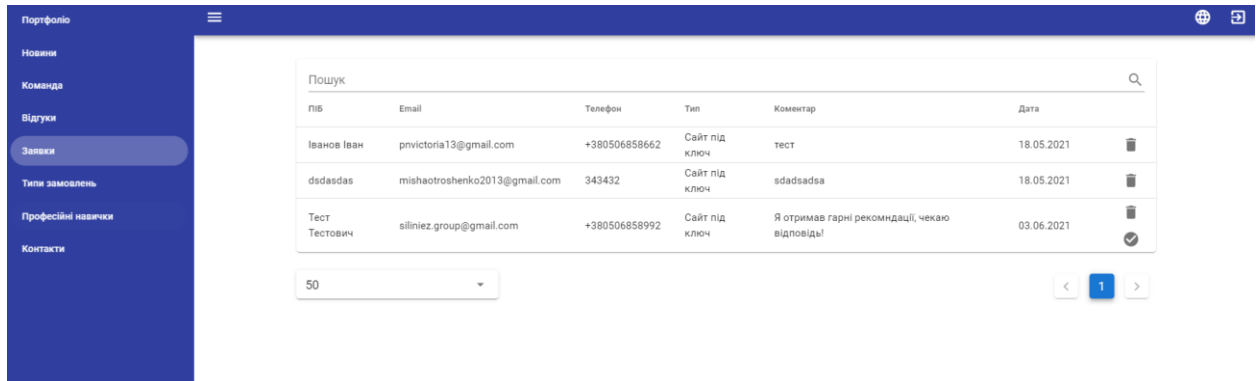


Рисунок 3.23 – Заявки

Нові заявки мають додаткову іконку, вона обирається після виконання замовлення.

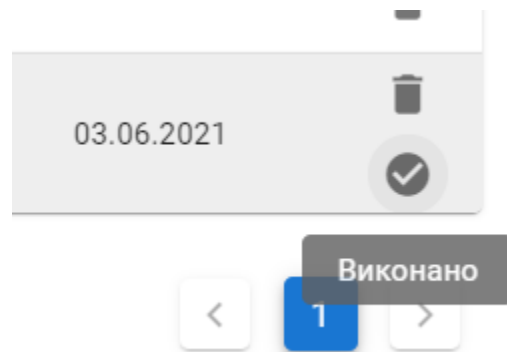


Рисунок 3.24 – Статус виконання замовлення

З'являється форма для додавання даної роботи до портфолію, а користувач отримує повідомлення для залишення відгуку (рис.3.25-26).


Тип

Назва

Рік

Посилання

Опис

 Зображення

ЗБЕРЕГТИ

Рисунок 3.25 – Форма для додавання нової роботи

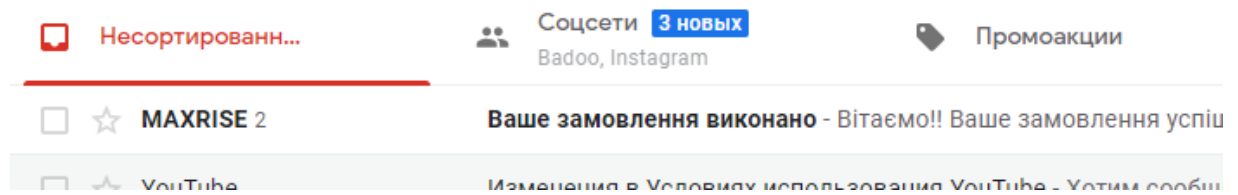


Рисунок 3.26 – Повідомлення про можливість залишення відгуку

У змісті повідомлення надається інформація про виконання замовлення та посилання на форму залишення відгуку (рис.3.27-29).

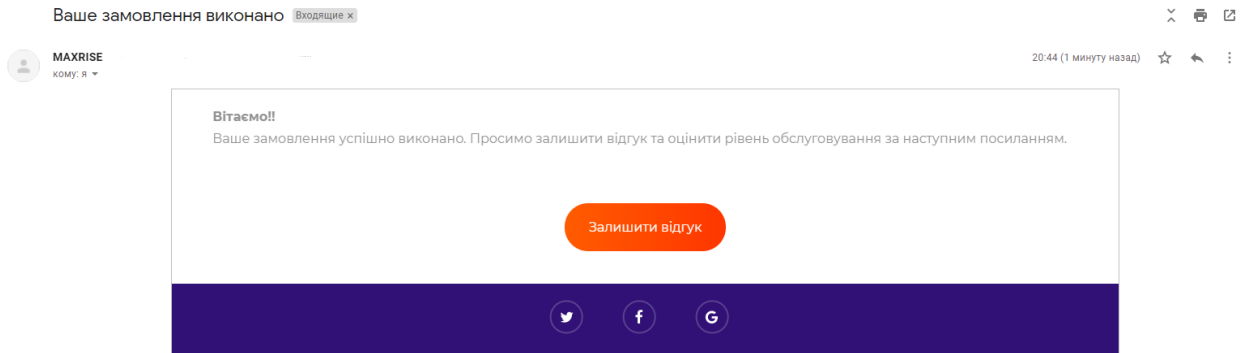


Рисунок 3.27 – Зміст повідомлення

ВАШ ВІДГУК

The image displays a form for leaving a review, set against a dark blue background. At the top, there is a dropdown menu labeled "Кількість балів" (Number of points). Below this is a large, empty white text area for entering the review. At the bottom center of the form, there is an orange button labeled "Надіслати" (Send).

Рисунок 3.28 – Форма залишення відгуку

ВАШ ВІДГУК

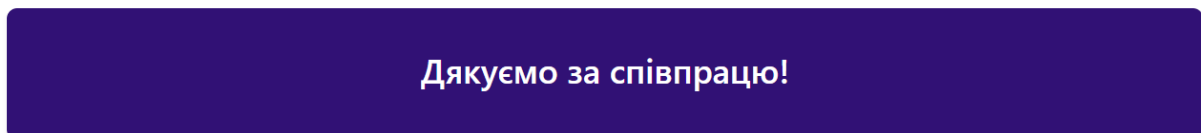


Рисунок 3.29 - Відгук додано

Залишений коментар відображається у розділі «ВІДГУКИ» (рис.3.30).

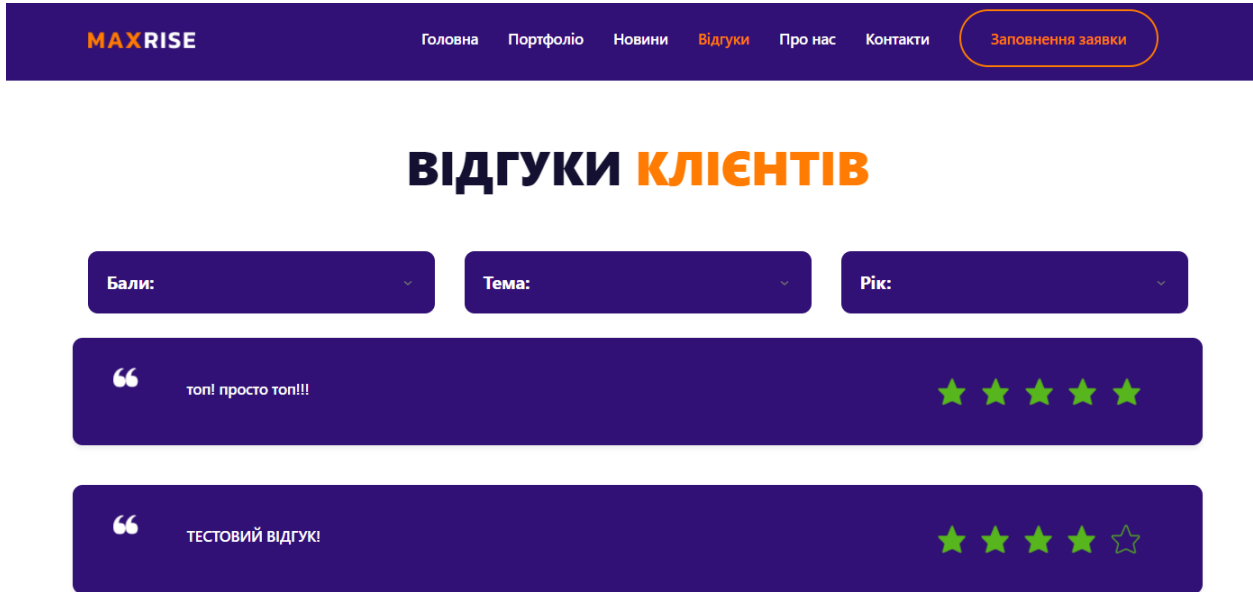


Рисунок 3.30 – Всі наявні відгуки

Адміністратор також додає та корегує замовлення згідно його типу (рис.3.31).



Рисунок 3.31 - Додавання типу замовлення

Авторизація в ролі адміністратора дає можливість редагувати, додавати та видаляти необхідні навички для персоналу (рис.3.32).

Портфоліо

Новини

Команда

Відгуки

Заявки

Типи замовлень

Професійні навички

Контакти

ДОДАТИ

Пошук

Назва

Front-end

Back-end

50

< 1 >

Рисунок 3.32 - Навички

Контактну інформацію також контролює адміністратор та корегує її коли це необхідно (рис.3.33).

Портфоліо

Новини

Команда

Відгуки

Заявки

Типи замовлень

Професійні навички

Контакти

Адрес

ул. Грушевського 30

Телефон

068 052-79-92

Email

maxrise@gmail.com

ЗБЕРЕГТИ

Рисунок 3.33 - Контакти

ВИСНОВКИ

Для швидкого та якісного вирішення поставленої проблеми було вирішено розробити інформаційну систему керування онлайн рекламою. Для надання зручної комунікації та надання якісних послуг у сфері реклами.

Для реалізації поставленої мети, перш за все, потрібно вирішити такі задачі:

- Ознайомитися із різновидами реклами.
- Провести аналіз сайтів-аналогів, що також надають послуги у сфері реклами.
- Визначити типи та категорії послуг.

В науково-дослідницькій практиці було проведено дослідження предметної області, а також проведено аналіз. Мета досягнута, поставлені завдання виконані.

СПИСОК ЛІТЕРАТУРИ

1. Online Advertising: Everything You Need to Know in 2020 [Електронний ресурс] – Режим доступу до ресурсу: <https://blog.hubspot.com/marketing/online-advertising>.
2. In Advertising and the World Wide Web (Psychology Press, 2021), pp. 109–128.
3. Essential online advertising trends 2021 [Електронний ресурс] – Режим доступу до ресурсу: <https://www.bannerflow.com/blog/essential-online-advertising-trends-2021/>.
4. S. I. Proshkina, The effectiveness of web advertising, its legal regulation and the problem of information systems security. Век информации (сетевое издание). 4, 1–15 (2020).
5. Keys to Starting Your Own Digital Marketing Agency [Електронний ресурс] – Режим доступу до ресурсу: <https://www.wordstream.com/blog/ws/2018/07/17/how-to-start-a-digital-marketing-agency>.
6. D. Riaño, R. Piñon, G. Molero-Castillo, E. Bárcenas, A. Velázquez-Mena, Regular Expressions for Web Advertising Detection Based on an Automatic Sliding Algorithm. Programming and Computer Software. 46, 652–660 (2020).
7. Y. Cai, G. O. M. Yee, Y. X. Gu, C.-H. Lung, Threats to Online Advertising and Countermeasures. Digital Threats: Research and Practice. 1, 1–27 (2020).
8. In Advertising and the World Wide Web (Psychology Press, 2021), pp. 37–50.
9. M. Sutherland, in Advertising and the Mind of the Consumer (Routledge, 2020), pp. 266–277.
10. How to Start an Online Advertising Company [Електронний ресурс] – Режим доступу до ресурсу: <https://bizfluent.com/how-8705383-start-online-advertising-company.html>.
11. Visual Examples to Illustrate Site Structure For Designers [Електронний ресурс] – Режим доступу до ресурсу: <https://www.uxpin.com/studio/blog/web-structures-explained/>.

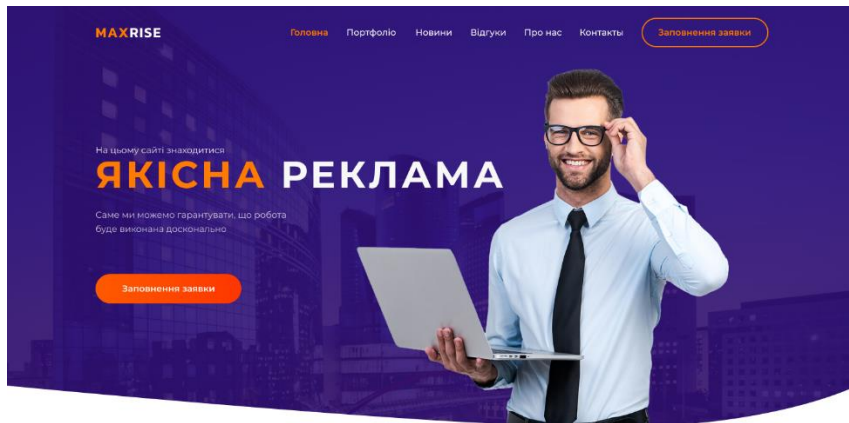
12. How to design an SEO-friendly [Електронний ресурс] – Режим доступу до ресурсу: <https://www.infidigit.com/blog/website-structure/>.
13. Laravel – Функції безпеки [Електронний ресурс] – Режим доступу до ресурсу: <https://webformymself.com/laravel-funkcii-bezopasnosti/>.
14. Марк Сафронов: Разработка веб-приложений в Yii 2, 2015. - 295 с.
15. Створення базового додатку MVC Laravel 5 Види сайтів [Електронний ресурс] – Режим доступу до ресурсу: <https://selftaughtcoders.com/from-idea-to-launch/lesson-17/laravel-5-mvc-application-in-10-minutes/>.
16. G. B. Santoso, T. M. Sinaga, A. Zuhdi, MVC Implementation In Laravel Framework For Development Web-Based E-Commerce Applications. Intelmatics. 1 (2021).
17. D. A. Pratama, DESIGN OF FISH SALES INFORMATION SYSTEM IN PT XYZ USING LARAVEL FRAMEWORK. Intelmatics. 1 (2021).
18. Best Personal and Niche Blogs (30+ Real Examples) [Електронний ресурс] – Режим доступу до ресурсу: <https://firstsiteguide.com/examples-of-blogs/>.
19. R. Y. He, in Proceedings of the 2014 International Conference on Computer Science and Electronic Technology (Atlantis Press, 2015)
20. The Pros and Cons of 8 Popular Databases [Електронний ресурс] – Режим доступу до ресурсу: <https://www.keycdn.com/blog/popular-databases>.
21. MySQL Stored Procedure Advantages [Електронний ресурс] – Режим доступу до ресурсу: <https://www.tutorialspoint.com/What-are-the-advantages-and-disadvantages-of-using-MySQL-stored-procedures>.
22. M. Ferati, B. Vogel, Accessibility in web development courses: A case study. Informatics. 7 (2020).
23. N. G. Maleki, R. Ramsin, Agile web development methodologies: A survey and evaluation. Studies in Computational Intelligence. 722, 1–25 (2018).
24. A. M. French, Web development life cycle: A new methodology for developing web applications. Journal of Internet Banking and Commerce. 16 (2011).

25. C. S. Miller, R. Connolly, Introduction to the special issue on web development. ACM Transactions on Computing Education. 15 (2015).

26. Прототип інформаційної системи керування онлайн рекламою в Figma [Електронний ресурс] – Режим доступу до ресурсу: <https://www.figma.com/file/2ac0n8No1L6wMui8H6pZCA/WebStudio?node-id=0%3A1>.

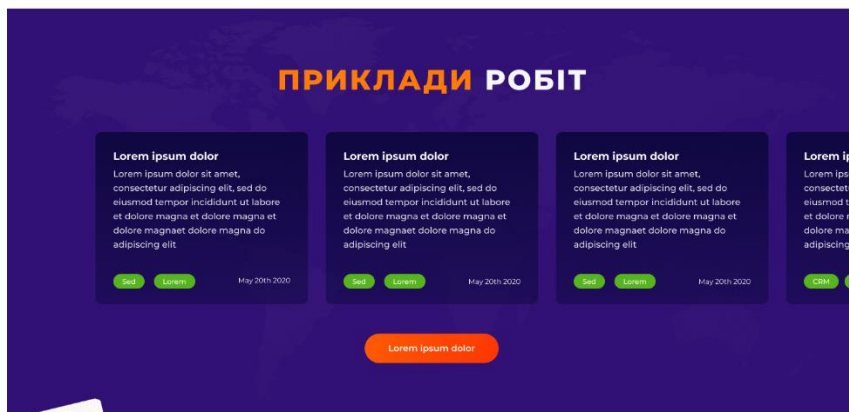
27. What is Figma? [Електронний ресурс] – Режим доступу до ресурсу: <https://webdesign.tutsplus.com/articles/what-is-figma--cms-32272>.

ДОДАТОК А. МАКЕТИ СТОРІНОК ІНФОРМАЦІЙНОЇ СИСТЕМИ



ЧОМУ МИ?

- Lorem ipsum dolor**
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco
- Lorem ipsum dolor**
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco
- Lorem ipsum dolor**
Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco



LOREM IPSUM DOLOR



Lorem ipsum dolor
Lorem ipsum dolor sit amet, consectetur adipiscing



Lorem ipsum dolor
Lorem ipsum dolor sit amet, consectetur adipiscing



Lorem ipsum dolor
Lorem ipsum dolor sit amet, consectetur adipiscing



ПРИКЛАДИ РОБИТ

Пошук: Designing

Тема: Dashboard

Рік: 2020




Designing Dashboards

2020 Dashboard

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint...

>Lorem ipsum dolor

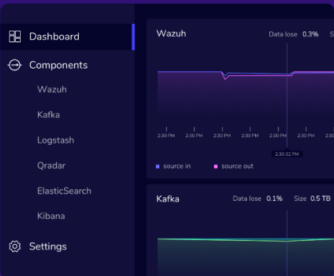


Designing Dashboards

2020 Dashboard

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint...

>Lorem ipsum dolor



Designing Dashboards

2020 Dashboard

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint...

>Lorem ipsum dolor

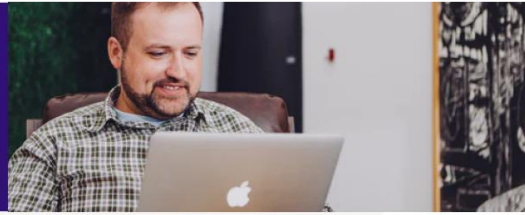


ОСТАННІ НОВИНИ

LONG ESTABLISHED

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is...

May 20th 2020



LONG ESTABLISHED

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout....

May 20th 2020



LONG ESTABLISHED

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout....

May 20th 2020



LONG ESTABLISHED

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout....

May 20th 2020



LONG ESTABLISHED

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout....

May 20th 2020



LONG ESTABLISHED

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout....

May 20th 2020



LONG ESTABLISHED

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout....

May 20th 2020





Finibus Bonorum et Malorum

Written by John doe

Monday, May 20

Section 1.10.33 of "de Finibus Bonorum et Malorum", written by Cicero in 45 BC. "Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto aeternae vitae dictae sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit, sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minima veniam, quis nostrum exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequatur? Quis autem vel eum iure reprehenderit qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat quo voluptas nulla paratur?"

1914 translation by H. Rackham. "But it must exist in you how all this mistaken idea of denouncing pleasure and praising pain was born and I will give you a complete account of the system, and expound the actual teachings of the great explorer of the truth, the master builder of human happiness. No one rejects, dislikes, or avoids pleasure itself, because it is pleasure, but because those who do not know how to pursue pleasure rationally encounter consequences that are extremely painful. Nor again is there anyone who loves or pursues or desires to obtain pain of itself, because it is pain, but because occasionally circumstances occur in which toil and pain can procure him some great pleasure. To take a trivial example, which of us ever undertakes laborious physical exercise, except to obtain some advantage from it? But who has any right to find fault with a man who chooses to enjoy a pleasure that has no annoying consequences, or one who avoids a pain that produces no resultant pleasure?"

Section 1.10.33 of "de Finibus Bonorum et Malorum" "At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat."

Section

"At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat."

Section 1.10.33

"At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga. Et harum quidem rerum facilis est et expedita distinctio. Nam libero tempore, cum soluta nobis est eligendi optio cumque nihil impedit quo minus id quod maxime placeat facere possimus, omnis voluptas assumenda est, omnis dolor repellendus. Temporibus autem quibusdam et aut officiis debitis aut rerum necessitatibus saepe eveniet ut et voluptates repudiandae sint et molestiae non recusandae. Itaque earum rerum hic tenetur a sapiente delectus, ut aut reiciendis voluptatibus maiores alias consequatur aut perferendis doloribus asperiores repellat."

Залишити коментар

Прізвище та ім'я

Електронна адреса

Text input field for comment content.

Залишити коментар

Пономаренко Вікторія

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do

May 20th 2020

Пономаренко Вікторія

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do

May 20th 2020

Пономаренко Вікторія

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do

May 20th 2020



ВІДГУКИ КЛІЄНТІВ

Пошук:

Тема: **Dashboard** ▼

Рік: **2020** ▼

“ Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sintet. Exercitation veniam consequat sunt nostrud amet.



Переглянути роботу

“ Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sintet. Exercitation veniam consequat sunt nostrud amet.



Переглянути роботу

“ Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sintet. Exercitation veniam consequat sunt nostrud amet.



Переглянути роботу

“ Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam consequat sunt nostrud amet. Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sintet. Exercitation veniam consequat sunt nostrud amet.




Переглянути роботу



НАША КОМАНДА


⏪



**ПОНОМАРЕНКО
ВІКТОРІЯ**

Дизайнер Менеджер


Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam nostrud amet.



**ПОНОМАРЕНКО
ВІКТОРІЯ**

Дизайнер

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam nostrud amet.



**ПОНОМАРЕНКО
ВІКТОРІЯ**

Дизайнер Менеджер

Amet minim mollit non deserunt ullamco est sit aliqua dolor do amet sint. Velit officia consequat duis enim velit mollit. Exercitation veniam nostrud amet.

⏩

ЗВ'ЯЖІТЬСЯ З НАМИ



6386 Spring St
United States



willie@example.com



(843) 555-0130



ОФОРМЛЕННЯ ЗАМОВЛЕННЯ

Please submit your information and a Designmodo representative will get in touch with you

Call sales at +1 555 505 5050

Help

Need help? Don't hesitate to ask us something. Email us directly info@designmodo.com or call us at 1-347-523-34-90. You can checkout our F.A.Q. and Help page to get more information about our products.

ДОДАТОК Б. КОД РЕАЛІЗАЦІЇ ДОДАТКУ

Index.vue

Код головної сторінки сайту.

```

<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">
          <span>якісна</span> реклама
        </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>
      <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 сформують
                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="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 проєктів
  </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);
    });
  },
  data() {
    return {
      data: []
    }
  },
  created() {
    this.getData();
  },
  methods: {
    getData() {
      axios.get('/api/portfolio')
        .then((response) => {
          this.data = response.data;
        })
    }
  }
}
</script>
<style lang="css" scoped>
.title {
  color: #141032;
  font-size: 48px;

```

```
font-weight: 800;
}
.title span {
  color: #FF7B01;
}
.block-1 {
  height: 700px;
  background: url('/img/index-header-bg-top.png') no-repeat;
  background-size: cover;
  position: relative;
}
.block-1 .man {
  position: absolute;
  right: 0;
  top: 40px;
}
.block-1 .bottom {
  height: 144px;
  width: 100%;
  background: url('/img/index-header-bg-bottom.png') no-repeat;
  background-size: cover;
  position: absolute;
  bottom: 0;
  z-index: 1;
}
.block-1 .content {
  color: #fff;
  padding-top: 150px;
}
.block-1 .content .subtitle {
  color: #F8F8F8;
  font-size: 15px;
  line-height: 160%;
  opacity: 0.8;
}
.block-1 .content .title {
  font-weight: bold;
  font-size: 64px;
  text-transform: uppercase;
  letter-spacing: 0.07em;
  line-height: 1;
  color: #fff;
}
.block-1 .content .title span {
  color: #FF7B01;
}

.block-2 {
  background: url('/img/apple.png') no-repeat;
  background-position: bottom right;
  margin-bottom: 100px;
}
.block-2 .steps {
  margin-top: 60px;
  color: #141032;
}
.block-2 .steps .step-item {
  border-left: 1px solid #EBEBEB;
  padding: 0 0 40px 25px;
  box-sizing: border-box;
}
```

```

.block-2 .steps .step-item:last-child {
  border-left: 0px;
}
.block-2 .steps .step-item h3 {
  font-weight: bold;
  font-size: 18px;
  margin-left: -29px;
  padding-left: 21px;
  border-left: 7px solid #FF7B01;
  box-sizing: border-box;
}
.block-2 .steps .step-item p {
  font-weight: 500;
  font-size: 15px;
}

.block-3 {
  background: url('/img/block-3-bg.png') no-repeat;
  background-size: cover;
  height: 650px;
  display: flex;
  align-items: center;
  position: relative;
}
.block-3 .link {
  position: absolute;
  bottom: 70px;
  left: 40%;
}
.block-3 ul li:first-child {
  margin-left: 150px;
}
.block-3 ul li .title {
  font-weight: bold;
  font-size: 18px;
  color: #fff;
}
.block-3 ul li .description {
  font-weight: 500;
  font-size: 15px;
  line-height: 160%;
  opacity: 0.9;
  height: 150px;
  overflow: hidden;
}
.block-3 ul li {
  display: inline-block;
  width: 360px;
  padding: 30px;
  border-radius: 10px;
  background: linear-gradient(180deg, #0F0841 0%, rgba(15, 8, 65, 0.5) 100%);
  white-space: normal;
  color: #fff;
  margin-right: 30px;
}
.block-3 ul {
  white-space: nowrap;
}
.block-3 ul li .footer {
  margin-top: 38px;
  display: flex;

```

```

    align-items: center;
  }
.block-3 ul li .footer span {
  background: #57B61D;
  border-radius: 99px;
  padding: 3px 15px;
  margin-right: 15px;
  font-size: 14px;
}
.works {
  max-width: 100%;
  overflow: hidden;
  width: 100%;
  display: block;
}

.block-4 {
  margin-top: -30px;
  padding-top: 100px;
  padding-bottom: 100px;
  background: url('/img/note.png') no-repeat, url('/img/glass.png') no-repeat;
  background-position: top left 40px, bottom right;
}
.block-4 .numbers {
  margin: 60px 0;
  background: url('/img/line.png') no-repeat;
  background-position: 50% 5%;
}
.block-4 .numbers .round {
  margin: 0 auto;
  width: 100px;
  height: 100px;
  color: #fff;
  background: #311175;
  border-radius: 50px;
  border: 10px solid #d1d1d1;
  box-sizing: border-box;
  font-weight: bold;
  font-size: 24px;
  display: flex;
  align-items: center;
  justify-content: center;
}
.block-4 .numbers .title {
  font-weight: bold;
  font-size: 18px;
  margin: 30px 0 10px 0;
}
.block-4 .numbers .subtitle {
  font-weight: 500;
  font-size: 15px;
  line-height: 160%;
}
@media screen and (max-width: 600px) {
  .block-1 .man {
    display: none;
  }
  .block-2 img {
    width: 100%;
  }
  .block-3 ul li:first-child {

```

```

margin-left: 0px;
}
.block-4 {
background: none;
}
.block-4 .numbers {
background: none;
}
.block-4 .numbers .subtitle {
margin-bottom: 20px;
}
}
</style>

```

Order.vue

Код сторінки замовлення послуг.

```

<template>
<div>
<Title first="оформлення" last="замовлення"></Title>
<b-container class="content" v-if="!status">
<b-row class="mb-5">
<b-col md="6" lg="6" xl="6" xs="12">
Будь ласка, надайте свою інформацію, і менеджер зв'яжеться з вами
</b-col>
<b-col md="6" lg="6" xl="6" xs="12">
<div class="title-2">Допомога</div>
Потрібна допомога? Надішліть нам електронного листа безпосередньо на maxrise@gmail.com, щоб отримати
більше інформації про наші продукти.
</b-col>
</b-row>

<b-row>
<b-col md="6" lg="6" xl="6" xs="12">
<input type="text" placeholder="Прізвище та ім'я" v-model="data.name">
</b-col>
<b-col md="6" lg="6" xl="6" xs="12">
<input type="text" placeholder="Електронна адреса" v-model="data.email">
</b-col>
<b-col md="6" lg="6" xl="6" xs="12">
<input type="text" placeholder="Номер телефону" v-model="data.phone">
</b-col>
<b-col md="6" lg="6" xl="6" xs="12">
<select v-model="data.work_type_id">
<option value="null" selected disabled hidden>Тип замовлення</option>
<option
v-for="item in work_type"
:key="item.id"
:value="item.id"
>{{ item.title }}</option>
</select>
</b-col>
<b-col cols="12">
<textarea cols="30" rows="10" v-model="data.comment"></textarea>
</b-col>
</b-row>
<div class="text-center">
<button @click="send()">Оформити заявку</button>
</div>

```

```

</b-container>
<b-container v-else class="text-center content" style="color: #fff">
  
  <h1 class="my-4">Дякуємо!</h1>
  <h3>Ваше замовлення успішно оформлено. Ми зв'яжемося з вами найближчим часом.</h3>
  <h2 class="mt-3">Номер вашого замовлення #{{ orderId }}</h2>
</b-container>
</div>
</template>
<script>
import Title from '../components/Title';
export default {
  components: {
    Title
  },
  data() {
    return {
      status: false,
      orderId: null,
      data: {
        name: "",
        email: "",
        phone: "",
        comment: "",
        work_type_id: ""
      },
      work_type: []
    }
  },
  created() {
    this.getWorkType();
  },
  methods: {
    getWorkType() {
      axios.get('/api/work-type')
        .then((response) => {
          this.work_type = response.data;
        })
    },
    send() {
      axios.post('/api/order', this.data)
        .then((response) => {
          this.orderId = response.data.id;
          this.status = true;
        })
    }
  }
}
</script>
<style lang="css" scoped>
.content {
  background: #311175;
  box-shadow: 0px 4px 4px rgba(192, 189, 189, 0.25);
  border-radius: 10px;
  padding: 40px 100px;
  color: rgba(255, 255, 255, 0.4);
  font-weight: 500;
  font-size: 15px;
  line-height: 160%;
  margin-bottom: 40px;
}

```



```

.content .title-2 {
  font-weight: bold;
  font-size: 18px;
  color: #fff;
  margin-bottom: 18px;
}
input, select {
  margin-bottom: 30px;
}
button {
  margin-top: 40px;
  border-radius: 50px;
}
@media screen and (max-width: 600px) {
  .content {
    padding: 20px;
  }
}
</style>

```

Portfolio.vue

Код сторінки портфоліо.

```

<template>
<div>
  <Title first="приклади" last="робіт"></Title>
  <b-container>
    <b-row class="mb-4">
      <b-col md="6" lg="6" xl="6" xs="12">
        <div class="form-block">
          <label for="search">Пошук:</label>
          <input type="text" id="search" v-model="filter.title" @input="getData()">
        </div>
      </b-col>
      <b-col md="3" lg="3" xl="3" xs="12">
        <div class="form-block">
          <label for="type">Тема:</label>
          <select id="type" v-model="filter.work_type" @change="getData()">
            <option value="" selected></option>
            <option
              v-for="item in work_type"
              :key="item.id"
              :value="item.id"
            >{{ item.title }}</option>
          </select>
        </div>
      </b-col>
      <b-col md="3" lg="3" xl="3" xs="12">
        <div class="form-block">
          <label for="year">Рік:</label>
          <select id="year" v-model="filter.year" @change="getData()">
            <option value="" selected></option>
            <option
              v-for="item in years"
              :key="item"
              :value="item"
            >{{ item }}</option>
          </select>
        </div>
      </b-col>
    </b-row>
  </b-container>
</div>

```

```

    </b-col>
  </b-row>
  <b-row v-for="item in data" :key="item.id" class="work">
    <b-col md="6" lg="6" xl="6" xs="12">
      
    </b-col>
    <b-col md="6" lg="6" xl="6" xs="12">
      <h3>{{ item.title }}</h3>
      <div class="subtitle">
        <span>{{ item.year }}</span>
        <span>{{ item.work_type.title }}</span>
        <div class="filler"></div>
        {{ item.created_at }}
      </div>
      <div class="description">{{ item.description }}</div>
      <a v-if="item.link && item.link != 'null'" :href="item.link" class="link" target="_blank">Переглянути</a>
    </b-col>
  </b-row>

  <div class="text-center mb-5 mt-5" v-if="data.length == 0">
    Інформація відсутня
  </div>

  <b-pagination v-if="data.length > 0" v-model="currentPage" :total-rows="data.length" :per-page="prePage"
align="center"></b-pagination>
</b-container>
</div>
</template>
<script>
import Title from './components/Title';
import Button from './components/Button';
export default {
  components: {
    Title,
    Button
  },
  data() {
    return {
      data: [],
      work_type: [],
      currentPage: 1,
      prePage: 9,
      filter: {
        work_type: "",
        year: "",
        title: ""
      }
    }
  },
  created() {
    this.getData();
    this.getWorkType();
  },
  computed: {
    paginatedList() {
      let start = (this.currentPage - 1) * this.prePage;
      let end = start + this.prePage;
      return this.filteredList.slice(start, end);
    },
    years() {
      const year = new Date().getFullYear();

```

```

        return Array.from({length: year - 2009}, (value, index) => 2010 + index).reverse();
    }
},
methods: {
  getData() {
    axios.get('/api/portfolio', {
      params: {
        work_type: this.filter.work_type,
        year: this.filter.year,
        title: this.filter.title
      }
    })
    .then((response) => {
      this.data = response.data;
    })
  },
  getWorkType() {
    axios.get('/api/work-type')
    .then((response) => {
      this.work_type = response.data;
    })
  }
}
}
</script>
<style lang="css" scoped>
  .work {
    margin-bottom: 40px;
    background: #311175;
    color: #fff;
    padding: 50px;
    border-radius: 10px;
  }
  .work img {
    width: 100%;
    border-radius: 10px;
  }
  .work h3 {
    font-weight: bold;
    font-size: 30px;
  }
  .work .subtitle {
    margin-top: 38px;
    display: flex;
    align-items: center;
  }
  .work .subtitle span {
    background: #57B61D;
    border-radius: 99px;
    padding: 3px 15px;
    margin-right: 15px;
  }
  .work .description {
    font-weight: 500;
    font-size: 15px;
    line-height: 160%;
    margin: 35px 0;
  }
  .work .link {
    display: inline-block;
    padding: 16px 45px;

```

```

color: #FFF;
background: linear-gradient(92.51deg, #FF5C00 0%, #FF3600 100%);
border-radius: 99px;
font-weight: 600;
font-size: 15px;
}
.work a.link:hover {
text-decoration: none;
}
@media screen and (max-width: 600px) {
.work {
padding: 10px;
padding-bottom: 30px;
}
.form-block {
margin-bottom: 10px;
}
.work img {
margin: 20px 0;
}
}
</style>

```

Team.vue

Код сторінки з командою.

```

<template>
<div>
<b-container>
<Title first="наша" last="команда"></Title>
<b-row>
<b-col v-for="item in data" :key="item.id" md="4" lg="4" xl="4" xs="12">
<div class="member">

<div class="name">{{ item.name }}</div>
<div class="skills">
<span v-for="(i, index) in item.skills" :key="index">{{ i.skill.title }}</span>
</div>
<p>{{ item.description }}</p>
</div>
</b-col>
</b-row>
</b-container>

<div class="text-center mb-5 mt-5" v-if="data.length == 0">
Інформація відсутня
</div>

<b-pagination v-if="data.length > 0" v-model="currentPage" :total-rows="data.length" :per-page="prePage"
align="center"></b-pagination>
</div>
</template>
<script>
import Title from '../components/Title';
export default {
components: {
Title
},
data() {

```

```

return {
  data: [],
  currentPage: 1,
  prePage: 9,
}
},
created() {
  this.getData();
},
computed: {
  paginatedList() {
    let start = (this.currentPage - 1) * this.prePage;
    let end = start + this.prePage;
    return this.data.slice(start, end);
  }
},
methods: {
  getData() {
    axios.get('/api/team')
      .then((response) => {
        this.data = response.data;
      })
  }
}
}
</script>
<style lang="css" scoped>
.member {
  background: #311175;
  box-shadow: 0px 3.64942px 3.64942px rgba(192, 189, 189, 0.25);
  border-radius: 10px;
  color: #fff;
  font-weight: 500;
  font-size: 13px;
  line-height: 160%;
}
.member img {
  width: 100%;
}
.member .name {
  font-weight: bold;
  font-size: 26px;
  line-height: 32px;
  text-align: center;
  text-transform: uppercase;
  padding: 27px;
}
.member .skills {
  text-align: center;
  padding: 0 27px;
}
.member .skills span {
  margin: 5px;
  display: inline-block;
  background: #57B61D;
  border-radius: 99px;
  padding: 5px 25px;
  margin-right: 15px;
  font-weight: bold;
  font-size: 13px;
}
}

```

```
.member p {
  padding: 22px 27px;
}
</style>
```

News.php

Код сторінки з новинами.

```
<template>
<div>
  <Title first="останні" last="новини"></Title>
  <b-container>
    <b-row>
      <b-col v-for="item in data" :key="item.id" md="4" lg="4" xl="4" xs="12" class="news-item">
        <div class="news">
          
          <div class="title"><router-link :to="'/news/'+item.id">{{ item.title }}</router-link></div>
          <div class="description">{{ item.description }}</div>
          <div class="date">
            {{ item.created_at }}
          </div>
        </div>
      </b-col>
    </b-row>

    <div class="text-center mb-5 mt-5" v-if="data.length == 0">
      Інформація відсутня
    </div>

    <b-pagination v-if="data.length > 0" v-model="currentPage" :total-rows="data.length" :per-page="prePage"
align="center"></b-pagination>
  </b-container>
</div>
</template>
<script>
import Title from '../components/Title';
export default {
  components: {
    Title
  },
  data() {
    return {
      data: [],
      currentPage: 1,
      prePage: 9,
    }
  },
  created() {
    this.getData();
  },
  computed: {
    paginatedList() {
      let start = (this.currentPage - 1) * this.prePage;
      let end = start + this.prePage;
      return this.data.slice(start, end);
    }
  },
  methods: {
    getData() {
```

```

        axios.get('/api/news')
        .then((response) => {
            this.data = response.data;
        })
    }
}
</script>
<style lang="css" scoped>
    .news-item {
        margin-bottom: 60px;
    }
    .news {
        border-radius: 5px;
        background: #311175;
        box-shadow: 0px 4px 4px rgba(192, 189, 189, 0.25);
        height: 100%;
        color: #fff;
    }
    .news img {
        width: 100%;
        border-radius: 5px 5px 0 0;
    }
    .news .title a {
        display: block;
        padding: 20px 20px 0 20px;
        text-transform: uppercase;
        color: #FF7B01;
        font-weight: bold;
        font-size: 18px;
    }
    .news .description {
        padding: 20px 20px 0 20px;
        font-weight: 500;
        font-size: 14px;
        line-height: 160%;
        max-height: 200px;
        overflow: hidden;
    }
    .news .date {
        font-size: 12px;
        padding: 50px 20px 20px 20px;
    }
</style>

```

Order.php

Модель заомвлень.

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Order extends Model
{
    use HasFactory;
    protected $table = 'order';
}

```

```

protected $fillable = [
    'name',
    'email',
    'phone',
    'comment',
    'work_type_id',
    'status'
];

protected $casts = [
    'created_at' => 'datetime:d.m.Y'
];

protected $hidden = [
    'updated_at',
];

function workType() {
    return $this->belongsTo('App\Models\WorkType', 'work_type_id');
}
}

```

Portfolio.php

Модель портфоліо.

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Portfolio extends Model
{
    use HasFactory;
    protected $table = 'portfolio';

    protected $fillable = [
        'title',
        'img',
        'description',
        'year',
        'link',
        'work_type_id'
    ];

    protected $casts = [
        'created_at' => 'datetime:d.m.Y'
    ];

    protected $hidden = [
        'updated_at',
    ];

    function workType() {
        return $this->belongsTo('App\Models\WorkType', 'work_type_id');
    }
}

```


Reviews.php**Модель відгуків.**

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Reviews extends Model
{
    use HasFactory;
    protected $table = 'reviews';

    protected $fillable = [
        'description',
        'rate',
        'order_id'
    ];

    protected $casts = [
        'created_at' => 'datetime:d.m.Y'
    ];

    protected $hidden = [
        'updated_at',
    ];

    function order() {
        return $this->belongsTo('App\Models\Order', 'order_id');
    }
}

```

News.php**Модель новин.**

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class News extends Model
{
    use HasFactory;

    protected $table = 'news';

    protected $fillable = [
        'title',
        'description',
        'img',
        'user_id'
    ];

    protected $casts = [

```

```

    'created_at' => 'datetime:d.m.Y'
];

protected $hidden = [
    'updated_at',
];

function user() {
    return $this->belongsTo('App\Models\User', 'user_id');
}

function comments() {
    return $this->hasMany('App\Models\Comments', 'id');
}
}

```

OrderController.php

Контролер обробки замовлень.

```

<?php

namespace App\Http\Controllers;

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

use App\Models\Order;

use Mail;

class OrderController extends Controller
{
    function get() {
        $model = Order::with('workType')->get();
        return response()->json($model);
    }
    function post(Request $request) {
        $model = new Order();
        $data = $request->all();
        $response = $model->create($data);
        return response()->json([
            'id' => $response['id']
        ]);
    }
    function delete($id) {
        Order::find($id)->delete();
        return response('ok', 200);
    }
    function postOrderReview($id) {
        $orderEmail = Order::find($id)->email;

        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, "Шановний клієнт")

```

```

->subject("Ваше замовлення виконано");

$message->from('mykhaylo.otroshchenko@student.sumdu.edu.ua', 'MAXRISE');
});

return response('ok', 200);
}
}

```

PortfolioController.php

Контролер для додання і редагування портфоліо.

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Models\Portfolio;

class PortfolioController extends Controller
{
    protected $publicStorage = "/uploads/portfolio/";

    function get(Request $request) {
        $model = Portfolio::with('workType')->orderBy('year', 'desc');

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

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

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

        $model = $model->get();

        return response()->json($model);
    }
    function getOne($id) {
        $model = Portfolio::find($id);
        return response()->json($model);
    }
    function post(Request $request) {
        $model = new Portfolio();
        $data = $request->all();
        if(isset($request['new_img']) && $request['new_img'] != "null" && $request['new_img'] != "undefined") {
            $name = $this->publicStorage . uniqid().'.'.$request['new_img']->getClientOriginalExtension();
            $request['new_img']->move(public_path().$this->publicStorage, $name);
            $data['img'] = $name;
        }
        $model->create($data);
        return response('ok', 200);
    }
    function put(Request $request, $id) {
        $model = Portfolio::find($id);

```

```

$data = $request->all();
if($request['new_img'] != "null" && $request['new_img'] != "undefined" && $request['new_img'] != null) {
    $name = $this->publicStorage . uniqid().'.'.$request['new_img']->getClientOriginalExtension();
    $request['new_img']->move(public_path()).$this->publicStorage, $name);
    $data['img'] = $name;
}
$model->update($data);
return response('ok', 200);
}
function delete($id) {
    Portfolio::find($id)->delete();
    return response('ok', 200);
}
}

```

ReviewController.php

Контролер відгуків користувачів.

<?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');

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

        if($request->work_type != "") {
            $model->whereHas('order', function($q) use ($request) {
                $q->where('work_type_id', $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
    ]);
  }
}
}
}
}

```

NewsController.php

КОНТРОЛЕР НОВИН.

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Auth;

use App\Models\News;
use App\Models\Comments;

class NewsController extends Controller
{
  protected $publicStorage = "/uploads/news/";

  function get() {
    $model = News::get();
    return response()->json($model);
  }
  function getOne($id) {
    $model = News::with('user', 'comments')->find($id);
    return response()->json($model);
  }
  function post(Request $request) {
    $model = new News();
    $data = $request->all();
    $data['user_id'] = Auth::id();
    if(isset($request['new_img']) && $request['new_img'] != "null" && $request['new_img'] != "undefined") {
      $name = $this->publicStorage . uniqid().'.'.$request['new_img']->getClientOriginalExtension();
      $request['new_img']->move(public_path().$this->publicStorage, $name);
      $data['img'] = $name;
    }
    $model->create($data);
    return response('ok', 200);
  }
  function put(Request $request, $id) {

```

```
$model = News::find($id);
$data = $request->all();
if(isset($request['new_img']) && $request['new_img'] != "null" && $request['new_img'] != "undefined") {
    $name = $this->publicStorage . uniqid().'.'.$request['new_img']->getClientOriginalExtension();
    $request['new_img']->move(public_path()).$this->publicStorage, $name);
    $data['img'] = $name;
}
$model->update($data);
return response('ok', 200);
}
function delete($id) {
    News::find($id)->delete();
    return response('ok', 200);
}
function postComment(Request $request, $id) {
    $model = new Comments();
    $data = $request->all();
    $data['news_id'] = $id;
    $model->create($data);
    return response('ok', 200);
}
}
```