

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра комп'ютерної інженерії

КРИНИЦЬКИЙ Олександр Володимирович
«Веб-сайт інформаційного ресурсу із
мінімалістичним дизайном / Web resource of
professional achievements with minimalist design»

спеціальність: 123 - Комп'ютерна інженерія
освітньо-професійна програма - Комп'ютерна інженерія
Кваліфікаційна робота

Виконав студент групи КІ-41
О.В. Криницький

Науковий керівник:
к.т.н. Н.Я. Савка

Кваліфікаційну роботу допущено
до захисту:

" ___ " _____ 20__ р.

Завідувач кафедри
_____ О.Л. Дубчак

Тернопіль – 2024

АНОТАЦІЯ

Кваліфікаційна робота на тему «Веб-сайт інформаційного ресурсу із мінімалістичним дизайном» зі спеціальності 123 «Комп'ютерна інженерія» освітнього ступеня «бакалавр» містить 72 сторінки пояснюючої записки, 14 рисунків, 1 таблиця, 4 додатки. Обсяг графічного матеріалу 2 аркуші формату А3.

Метою кваліфікаційної роботи є розробка інформаційного веб-ресурсу з мінімалістичним дизайном.

Методи розробки полягають у застосуванні методів аналізу та синтезу, порівняння, проектування, комп'ютерного моделювання, технології React, Node.js, HTML, CSS.

Розглянуто задачу проектування веб-портфоліо у мінімалістичному дизайні. Описано види інформаційних веб-ресурсів, програмні засоби їх проектування. Описано основні принципи стилів при проектуванні дизайну. Обґрунтовано переваги застосування стилю мінімалізму для розробки веб-портфоліо.

Описано етапи розробки інформаційних веб-ресурсів, технічне завдання та технічне рішення розробки. Проаналізовано технічний стек розробки веб-портфоліо. Здійснено аналіз вимог до розробки системи.

Розроблено веб-портфоліо на основі технології React із мінімалістичним стилем. Описано програмну реалізацію клієнтської та серверної частини веб-портфоліо. Наведено результати тестування розробленого веб-ресурсу.

Ключові слова: ВЕБ-САЙТ, ВЕБ-ПОРТФОЛІО, ТЕХНОЛОГІЯ REACT, МІНІМАЛІСТИЧНИЙ ДИЗАЙН.

ANNOTATION

Qualification thesis “Web resource of professional achievements with minimalist design” in the specialty 123 "Computer Engineering" "Bachelor" education degree contains 72 pages of explanatory notes, 14 figures, 1 table, 4 appendixes. The volume of graphic material is 2 sheets of A3 format.

The aim of qualification work is to development of information web resource with minimalist design.

Development methods consist in the application methods of analysis and synthesis, comparison, design, computer modeling, React technology, Node.js, HTML, CSS.

The task of designing web portfolio in minimalist design is considered. Types of information web resources, software tools for their design are described. The main principles of styles in designing the design are described. The advantages of using the style of minimalism for development of web portfolio are substantiated.

The stages of development informational web resources, the technical task and the technical solution of the development are described. The technical stack of web portfolio development is analyzed. An analysis of the requirements for the development of the system was carried out.

Developed web portfolio based on React technology with minimalistic style. The software implementation of the client and server part web portfolio is described. The results of testing developed web resource are given.

Keywords: WEBSITE, WEB PORTFOLIO, REACT TECHNOLOGY, MINIMALIST DESIGN.

ВСТУП

Незважаючи на те, що інтернет давно і міцно увійшов у наше життя, багато осіб, підприємців і навіть великі фірми не розуміють, що їм дасть створення власного сайту, адже є інші перевірені способи самореклами, щоб добре зарекомендували себе: телебачення, радіо, засоби масової інформації, банери, флаєри та будь-яка інша реклама.

Сьогодні не новизною є наявність сайту у будь-якої компанії. Це один із елементів престижу, адже саме в Інтернеті потенційні клієнти насамперед шукатимуть інформацію, що стосується тієї чи іншої послуги. І якщо фірма немає хоча б односторінника з прайсом, то це здається в теперішній час здається підозрілим – наскільки це неуспішна компанія, якщо немає ресурсів на розробку бодай мінімального веб-ресурсу?!

Актуальність створення сайту полягає у донесенні інформації максимально швидко до величезної аудиторії людей. Адже веб-ресурс дозволяє подати інформацію про компанію та її товари чи послуги стисло і водночас повноцінно. Також сайт може повідомляти про новини фірми, про зміни в прайсі або режим роботи, містити відгуки вдячних клієнтів.

В той же час, не кожна фірма потребує великого порталу зі складним дизайном та функціоналом. Іноді буває досить невеликого сайту-візитки. Це ж саме стосується самореклами особистостей, які хочуть, наприклад, знайти престижну роботу чи успішних партнерів для ведення бізнесу. В цьому випадку доцільним є розробка веб-портфолію, яке представлятиме збірку досягнень людини в освіті, науці, професійному житті.

Таким чином, доцільність розробки сайту пояснюється такими факторами:

- швидкість подання інформації широкому колу осіб;
- покращення іміджу та підвищення популярності;
- можливість організувати зворотний зв'язок;

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

- оперативний зв'язок із представниками різних з різних куточків країни та за кордоном;
- організація маркетингових досліджень;
- реклама та залучення клієнтів;
- збільшення трафіку.

В реаліях сьогодення жоден інший ресурс не дасть стільки переваг, скільки власний сайт: чи це візитівка, портфоліо, Інтернет-магазин чи будь-який інший веб-ресурс.

В наш час існує буквально мільон веб-сайтів, і більшість з них мають стандартний дизайн, який не дуже зручний та складний для запам'ятовування. До того ж завантажений інтерфейс суттєво утруднює пошук необхідної інформації й до такого веб-ресурсу не залишається бажання повертатися знову. За таких умов на перший план виступають лаконічність та мінімалізм веб-інтерфейсу, що суттєво економить час пошуку необхідної інформації.

Зважаючи на зазначене, мета роботи полягає в розробці веб-сайту інформаційного ресурсу із мінімалістичним дизайном.

Об'єкт дослідження – процеси розробки веб-ресурсів.

Предмет дослідження – технології та методи розробки простих та зрозумілих веб-інтерфейсів інформаційних ресурсів

Практична цінність полягає у розробці портфоліо як інформаційного веб-ресурсу.

Досягнення мети роботи можливе за умов вирішення таких задач:

- охарактеризувати процедуру дизайн-проектування веб-сайтів;
- проаналізувати існуючі стилі у веб-дизайні;
- обґрунтувати застосування стилю мінімалізму у мультимедійному дизайні;
- проаналізувати композиційні та естетичні аспекти мультимедійного дизайну;
- проаналізувати технології розробки інформаційних веб-ресурсів;
- сформулювати вимоги до розробки проекту;

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

- розробити архітектуру веб-ресурсу;
- спроектувати веб-портфолію;
- дослідити та обґрунтувати техніко-економічні показники розробки проекту.

Роботу виконано згідно вимог, що описано у [9-11].

Основні результати кваліфікаційної роботи опубліковано у тезах доповіді на Міжнародній науковій конференції «Організація наукових досліджень у сучасних умовах' 2024» [33]. Копії публікації наведено у додатку А.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

1 МІНІМАЛІЗМ У ДИЗАЙНІ ВЕБ-САЙТІВ

1.1 Аналіз інформаційних веб-ресурсів

Незважаючи на те, що інтернет давно став невід'ємною частиною життя, питання створення сайтів не втрачає своєї актуальності. Мета практично будь-якого Інтернет-проекту зараз – отримання прибутку, і лише кілька відсотків із мільйонів вже існуючих онлайн-ресурсів створюються для некомерційного використання.

Досягти поставлену задачу може декількома способами. Найочевидніший – прямий продаж товарів через інтернет, тобто сайт в даному випадку є класичний інтернет-магазин з каталогом, кошиком, системою оплати та вибором варіанту доставки. Але є й інші способи залучення потоку клієнтів та коштів. Навіть якщо компанія не займається прямою реалізацією товарів, сайт може виступати проміжною точкою контакту з потенційно зацікавленою аудиторією і, якщо він створений досить професійно, конвертувати певний відсоток відвідувачів у діючих клієнтів.

Щоб краще зрозуміти, як це відбувається, і які прийоми використовуються у певних ситуаціях, слід уточнити, які взагалі види сайтів, крім вже згаданого інтернет-магазину, існують [12].

Сайт-візитка – найпростіший сайт з однієї або декількох сторінок, створений лише для того, щоб дати потенційному клієнту актуальні контактні дані для зв'язку або, наприклад, надати схему проїзду. Ідеальний варіант для бізнесу.

Лендінг – вузькоспеціалізований односторінник, що підштовхує до конкретної дії. У більшості випадків до замовлення певної послуги або отримання контактів клієнта, за допомогою яких з ним можна буде зв'язатися в майбутньому. Використовується як поодинокі, як основний ресурс компанії, так і для додаткового просування.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

Корпоративний сайт – інструмент для середніх та великих компаній, за допомогою якого можна розповісти аудиторії абсолютно про будь-які аспекти роботи: від історії успіху та списку іменитих партнерів до актуальних новин.

Портфоліо – інструмент самореклами. Воно здатне виділити особистість серед сотні кандидатів чи, навпаки, відштовхнути потенційного роботодавця чи замовника. Чим же портфоліо відрізняється від резюме?

Резюме – це коротка професійна біографія кандидата. Тут пишуть інформацію про освіту, попередні місця роботи, основні навички, знання та вміння. Окрім професійних компетенцій у резюме вказують особисті якості: наприклад, стійкість до стресів, комунікабельність, вміння працювати в команді.

Портфоліо доповнює резюме та наочно доводить, що вміє фахівець. Його цінність у тому, що воно напрацьовується роками: не можна сісти за один вечір зробити портфоліо. Це реальні документи та файли. Знадобиться кілька років, щоб розробити проекти, які доведуть професіоналізм та знання. Портфоліо допомагає рекрутеру оцінити до якого рівня доріс кандидат на попередніх місцях роботи. А кандидату посилити резюме і підвищити шанс отримати пропозицію від роботодавця.

Існують сфери, в яких без портфоліо ніяк, зокрема [17]:

– дизайн та графіка – андидати можуть додавати портфоліо скетчі, дизайн веб-інтерфейсів, проекти ландшафтного дизайну, макети, ілюстрації, банери;

– розробка – для портфоліо підійдуть тривимірні моделі, частини коду, готові програми, математичні моделі, алгоритми.

– контент – у портфоліо можуть потрапити статті, огляди, гайди, мануали, інструкції, книги. Якщо візуальний контент, то це відеоролики, моушн, ілюстрації, схеми, інфографіка.

– освіта – у цій сфері успіх викладача успіх його учнів. Наприклад, якщо репетитор англійської мови готує учнів до складання тестів IELTS або TOEFL, кейсами будуть сертифікати учнів із високими балами.

– творчі спеціальності – це фотографи, стилісти, візажисти, відеографи. У їхньому портфоліо будуть фото та відео з роботами.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

Портфоліо можна роздрукувати, перенести до електронного файлу або розмістити на окремому сайті. Який формат вибрати – залежить від професії та того, як роботодавцю буде зручніше вивчити роботи.

Друковане портфоліо розміщують на звичайному папері, розкладають на теми або в хронологічному порядку і беруть із собою на співбесіду. Портфоліо можна доповнити сертифікатами, рекомендаційними листами. Його зручно передати на співбесіді HR-менеджеру та одразу обговорити нюанси. Проте таке порт фоліо має недоліки, зокрема, деякі документи можуть випадково випасти із папки й тоді портфоліо доведеться відновлювати. А якщо це були документи, які не містять копії, можна втратити їх назавжди.

В електронному портфоліо всі документи та кейси зберігаються у хмарі або на флешці. Портфоліо можна дивитися з комп'ютера, ноутбука, планшетного ПК або смартфона. Таке портфоліо збирають розробники. Наприклад, після курсу Skypro «Веб-розробник» можна показати потенційному роботодавцю свій код у GitHub. За 10 місяців навчання можна створити свій музичний сервіс, дошку оголошень та програму для карткової гри.

Переваги електронного портфоліо – це безпека понад усе. Раптом з комп'ютером щось трапиться, наприклад, згорить жорсткий диск чи збій операційної системи. Краще створити кілька копій файлів та розмістити їх у різних сховищах. Ще можна обмежувати доступ до даних, наприклад, відкривати файл тільки для перегляду, без можливості зміни чи копіювання. Недоліком такого портфоліо може бути несумісність файлів із файлами на комп'ютері HR-менеджера під час співбесіди.

Онлайн-портфоліо створюють на сайті – саморобному або спеціальному тематичному. Там розміщують посилання на проекти, публікують тексти, відео. Коли HR-менеджер отримує посилання на сайт-портфоліо, йому не потрібно нічого завантажувати або перегортати: він скролює сторінку вниз і переглядає файли. Якщо сервер сайту раптом зламається або у HR-менеджера перестане працювати інтернет, портфоліо на якийсь час стане недоступним, що може бути тимчасовим недоліком.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

Існують й інші різновиди веб-ресурсів, зокрема, особисті блоги, портали новин, форуми та ін., які здебільшого не належать до комерційних розробок. Таким чином, якою б із видів не була веб-розробка, всі їх об'єднують такі переваги [7]:

- широке охоплення аудиторії: на відміну інших засобів взаємодії він доступний у світі;
- прямий контакт із клієнтами за допомогою форм зворотного зв'язку та відгуків;
- актуальність інформації: зміни можна зробити максимально швидко, і вони відразу будуть доступні всім відвідувачам;
- показник статусу – повноцінний, якісно опрацьований та наповнений сайт викликає більше довіри, ніж, наприклад, оформлена сторінка у соцмережах;
- єдине джерело інформації зацікавлених сторін;
- збір даних, відгуків та пропозицій для подальшого планування діяльності;
- нові методи реклами, включаючи SEO-просування – пасивне залучення нових клієнтів.

У наступному підрозділі розглянемо особливості розробки дизайну веб-ресурсів з метою зацікавлення якомога більшої аудиторії та донесення необхідної інформації.

1.2 Характеристика дизайн-проектування веб-ресурсів

З кожним роком спостерігається розвиток нових технологій, які успішно розвиваються й досягають у своєму прогресі суттєвих висот, що стосується й веб-індустрії. Якщо ще двадцять років тому веб-сторінки служили засобом виключно представлення інформації, то сьогодні веб-ресурси – це один із провідних двигунів реклами у світі.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

З урахуванням зазначеного, в особі веб-дизайнера замовники бачать не лише людину, що оформлює сторінки, а розглядають сам дизайн як основний інструмент продажу своїх послуг та просування в мережі. Таким чином, розуміючи навантаження та поставлені задачі, веб-дизайнер для досягнення цілей замовника має вирішувати спектр задач, що призводять до розробки повноцінного сайту. Серед таких задач виділяють: аналіз сприйняття різного роду інформації, використання веб-аналітики, рівень засвоєння користувачем інформації, наповнення текстових блоків, застосування шрифтів та елементів графіки[12, 14].

Короткий аналіз сприйняття різних видів інформації показує, що, «в першу чергу, необхідно задати емоційне забарвлення сайту. За допомогою одного кольору можна зробити сайт більш теплим і візуально затишним, або навпаки, суворим, холодним і класичним. Тому для веб-дизайнерів необхідними є глибокі знання теорії кольору і психології впливу кольорів на людину. За допомогою кольорів веб-дизайнери керують увагою майбутніх клієнтів і їх емоціями в момент перегляду» [27].

Окрім кольору, психологічний вплив мають веб-інструменти, зокрема, текст, графіка, символіка та зображення. Такі механізми направляють відвідувача веб-сторінки у таку психологічну область, яка необхідна для замовника, і тим самим робити акцент на потрібних речах. Таким чином, перед веб-дизайнером постає одна із ключових задач: не тільки розробити привабливий інтерфейс сторінки, але й привернути увагу відвідувачів на такі об'єкти, які потрібні замовнику для їх подальшого просування.

При відвідуванні веб-сайтів за кожним користувачем закріплюється так би мовити невидимий збирач даних, який фіксує – час користувача на сайті, його дії, зацікавлені категорії, чи корисна навігація і т.д. Збір таких даних необхідний не тільки для того, щоб в майбутньому пропонувати потрібну рекламу на основі зібраної статистики, але й щоб оцінити наскільки ефективно попрацював веб-дизайнер, чи зміг впоратися із поставленою задачею й досягти необхідних цілей замовника [19].

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						16
Змн.	Арк.	№ докум.	Підпис	Дата		

Засвоєння інформації користувачем. Перед створенням сторінки, веб-дизайнер мусить детально продумати кількість видимої для користувача інформації. «Це пов'язано з результатами психологічних досліджень, які показали, що людина здатна засвоїти лише близько 9 візуальних об'єктів одночасно, з яких запам'ятовуються лише 4. Це значить, що якщо використовувати більше 9 візуальних елементів, то сайт стає важким для сприйняття і з'являється бажання негайно закрити ресурс» [14].

Якщо відвідувач все ж таки залишиться на сайті, то через велику кількість об'єктів, у нього може виникнути відчуття візуального сміття і в кінцевому результаті це призведе до дуже слабкої концентрації уваги на тих елементах, які висвітлюються, як головні, й інформацію буде не засвоєно.

Інше дослідженнями показує, «що користувачу потрібно не більше 7 секунд, щоб оцінити наскільки добре продуманий сайт, чи варто на ньому затримуватися. Саме тому завдання зробити акцент на потрібних речах і не перевантажити сторінки інформацією є таким важливим» [14].

Наповнення текстових блоків. За допомогою тексту задають тон всього сайту, оскільки стилістика великою мірою сприяє створенню атмосфери від прочитаної інформації й залучення цільової аудиторії. Дотримання єдиного стилю, суцільні, візуально нерозривні блоки текстів, опис тем в стилі засобів масової інформації – до цього прагне весь сучасний світ.

Проте навіть за допомогою звичних слів керувати аудиторією на психологічному рівні. Для цього передбачені різні історії з життя, особисті приклади, оформлення прямої мови, окличних речень, як методи жвавого спілкування. «Не варто так само затягувати з пропозиціями товарів чи послуг. Вчені з'ясували, що оптимальний обсяг тексту для швидкого засвоєння і запам'ятовування – рядки від 26 до 40 символів. Все що довше – змушує людину занудьгувати або перечитати текст заново» [27].

Підвищити довіру чи спонукати до читання може також цікавий голосний заголовок або фотокартки. Таким чином, веб-дизайнер повинен не тільки грамотно розмістити блоки з текстом, а й обережно та ненав'язливо підкріпити

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

зображення, як доповнення до статті, проте основну увагу все-таки зосереджувати на публікації.

Крім грамотного розташування тексту на робочій області веб-сайту, дизайнер має дотримуватися єдиної шрифтової стилістики сторунки. Адже заданий шрифт формує загальну думку про подачу матеріалу, тому основним фактором його легке і просте читання. В той же час, веб-дизайнер повинен розуміти, які емоції від текстового оформлення мають отримати відвідувачі, оскільки кожен стиль подачі букв викликає різні асоціації.

«Якщо хочеться додати важливості та так званої гучності, то додають функцію жирності шрифту, якщо ж навпаки, метою є легкість, то використовують нахил. Так само має значення те, як пишеться текст: чи він із зарубками і яким розміром оформляється» [29].

Таким чином, вибір шрифту має відповідати призначенню веб-сторінки, а кількість шрифтів, використаних на одній сторінці, не повинна перевищувати трьох. У протилежному випадку, при великій кількості шрифтів втрачається візуальна гармонія сторінки. Уникнути такої ситуації можна за рахунок використання групи шрифтів, в якому є один основний шрифт та кілька його варіацій у різних стилях [29].

Графічні елементи. Щоб додати завершеність сторінки, застосовуються тематичні картинки, фотографії чи певні елементи графіки. В залежності від потрібного ефекту, цим також можна психологічно впливати на користувачів. Використання графіки надто великих розмірів зумовлює звернення уваги користувача на важливості цієї інформації. Малопомітні картинки вказують на скромність, щось не надто важливе, сопкійне.

Окремі елементи графічних об'єктів таклож відіграють важливу роль. «Занадто прямі кути і лінії – зчитуються нашою підсвідомістю як солідність, гострі кути говорять про наполегливість, для підкреслення ліберальності можна використати хвилясті лінії. Найскладнішим завданням для веб-дизайнера вважають правильність оцінки простору робочої області сторінки, де потрібно розмістити об'єкти, які візуально будуть перебувати нібито на своєму місці, і в

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

доповненні з текстом і картинками будуть складати одне ціле. Якщо цього не досягти, то увага користувача може розсіюватися і підсвідомість людини почне розцінювати побачене як неприємний хаос, з якого скоріше хочеться вибратися, тобто банально закрити сторінку» [29].

Зважаючи на вищенаписане, однією з основних цілей веб-дизайну є дослідження на основі веб-ресурсів впливу на сприйняття людини різних психологічних факторів. Для цього на розроблених веб-сайтах поміщається уся необхідна інформація для замовника, сторінку оформляють у потрібних кольорах і відтінках, створюють візуальний підхід, щоб зручно використовувати навігацію.

Після потрапляння відвідувача на поточний сайт, на основі веб-аналітики відстежують, які ж саме розділи і категорії його зацікавили, чи скористався він навігацією сайту, або чи час перебування на сторінці тривала кілька хвилин. За умови, що користувач, наприклад, не тільки придбав продукт чи послугу, а й за деякий час зайшов на сайт знову – тоді вважають, що веб-дизайнер досягнув мети замовника правильно, розробник зміг реалізувати проєкт таким чином, що відвідувачі сайту не лише принесли замовнику прибуток, а й стали постійними покупцями, які потенційно залучатимуть нову аудиторію.

1.3 Аналіз стилів веб-дизайну

Як і в інших сферах інформаційних технологій, в дизайні є також певні тенденції, які зорієнтовують розробників. Тенденції мають властивість зміни, проте існують такі стилі у веб-дизайні, які досить довго та ефективно знаходять своє застосування.

Існує понад 100 стилів сайтів, які суттєво відрізняються один від одного. На сьогодні досить популярними у світі вважають сайти таких стильових напрямків: класика, типографіка, мінімалізм, flat дизай, metro (картковий дизайн), organic & natural [2, 14, 18, 31].

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

Класичний стиль можна назвати найбільш універсальним, оскільки за своїм оформленням він підходить як офіційним організаціям, так і салонам краси. Така стилістика у візуальний спосіб повідомляє відвідувачу про надійність, стабільність, готовність до ділової співпраці та подальших взаєминах. Щоб розробити сайт у класичному стилі, потрібно розуміти, що такий напрямок опирається на досить строгі та чіткі норми оформлення, серед яких виділяють [32]:

- структура веб-сайту розроблена правильними колонками;
- у верхній частині сторінки завжди розміщені назва та логотип;
- меню, випадаючи або статичне, розташоване зверху або зліва сторінки;
- колірній гамі обов'язково притаманна гармонійність, стриманість, без зайвої яскравості і у світлих відтінках;
- якщо ж компанія має свою айдентику, то такий стиль включає фірмові кольори та елементи;
- відсутні мультимедіа, анімація та графіка;
- шрифти обирають максимально строгі, без надмірності та зайвої помітності.

Оформлення у класичному стилі сайтів не відволікає уваги відвідувачів на дизайні. Метою сайту є подача максимальної інформації про ділові стосунки та бізнес (див. рис. 1.1).



Рисунок 1.1 – Класичний стиль у веб-дизайні

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

Мінімалізм бере свій початок із архітектури, де використовували широкі простори і дуже непомітне оформлення. Задум стилю полягає в простоті, відсутності зайвого, тільки мінімалістичні картинки, трохи візуального контенту, текст здебільшого представлений у формі відео або наявно лише кілька основних фраз чи закликів, слоганів [2].

Основний акцент зумовлено на максимально спрощеному і мінімалістичному оформленні. Стиль базується на повній відмові від зайвого і всюди використання одного стилю, починаючи від графічних об'єктів і закінчуючи шрифтами. В мінімалізмі, на відміну від класики, оформляють лише цільові сторінки, корпоративні сайти, інтернет-магазини для середнього бізнесу (див. рис. 1.2).

ALL DESIGN DEVELOPMENT ILLUSTRATION

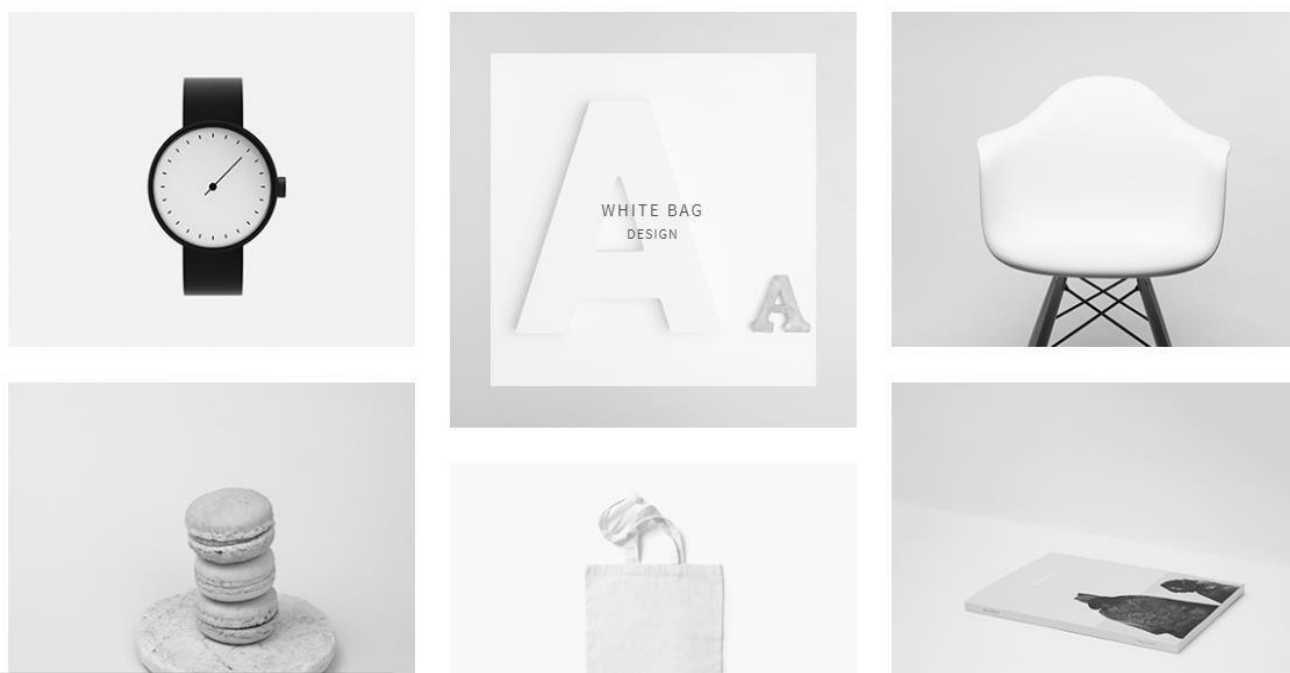


Рисунок 1.2 – Приклад стилю мінімалізм в оформленні сайту

Типографіка по суті є оригінальним стилем, який використовують різного типу друковані видання у вигляді газетних чи журнальних об'єктів. Зважаючи на це, оформляється такий стильна основі принципів типографіки, де:

- текст та графічний контент відображаються колонками;

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

- одночасно використовуються різні шрифти, які відрізняються формою, розмірами, типом;
- текстову частину представлено у вигляді ілюстрацій, відео та фотоматеріалів.

Відмінність стилю типографіки від інших полягає у тому, що такий стиль передбачає використання багато візуального і часто нестандартного оформлення, яке для інших стилів є неприпустимим (див. рис. 1.3).



Рисунок 1.3 – Оформлення сайту в стилі типографіки

Metro (картковий дизайн) складається із поєднання геометричних плиток прямокутників і квадратів, які однакові за форматом та виглядом. Такий стиль відрізняють від інших такі ознаки:

- оформлення мінімальне;
- яскрава палітра кольорів;
- нестандартна і явно помітна типографія;
- обов’язкова строгість у розмірах пропорціях;
- логічна структура, інтуїтивно зрозуміла навігація, множина посилань та функціональність.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

Стилістика цього напрямку універсальна, і при якісному виконанні підходить під потреби багатьом сайтам (див. рис. 1.4).

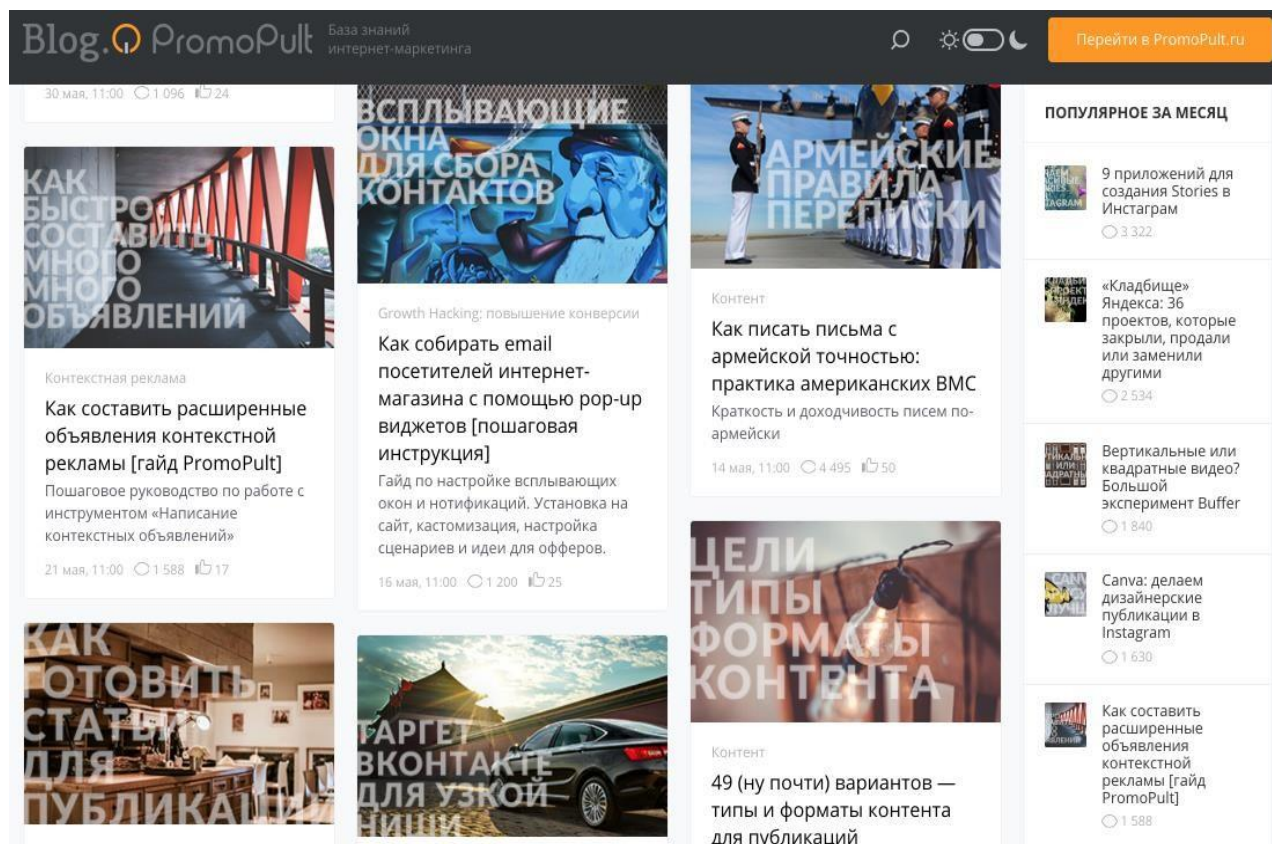


Рисунок 1.4 – Приклад Metro або карткового дизайну

Flat (плоский) дизайн передбачає яскравий стиль, який складається в основному з двовимірних, сплюснених геометричних компонентів. Основними ознаками такого стилю є:

- контент на першочерговість, де графічні елементи не відволікають уваги користувача, а сприяють отриманню необхідної інформації;
- значну роль відведено типографії, що легка для запам'ятовування;
- інтуїтивно зрозумілий шаблон розташування елементів і ресурсів;
- використання яскравих відтінків, відеоілюстрацій, великоформатних фото.

Стиль Flat поширений серед різних напрямків, зокрема, як у бізнесі, так і для веб-сайтів інших типів (див. рис. 1.5).

										Арк.
										23
Змн.	Арк.	№ докум.	Підпис	Дата						



Рисунок 1.5 – Приклад застосування Flat дизайну

Organic & Natural – досить популярний стиль, який практично не піддається впливам моди на різні доповнення. Натуральний стиль завжди в тренді, оскільки для цього напрямку головним є естетика. Тому логічно, що в межах естетики веб-дизайнери застосовують такі мотиви, які запозичені у самої природи:

- зображення дерев, квітів, трави, води або гір на текстурах;
- додавання графіки у вигляді елементів різних варіацій рослинності чи пейзажів є обов’язковим;
- першочергову роль відіграє візуальна частина, інформативність, як і контент займають другорядні ролі.

Сфери застосування натурального дизайну досить широкі й не мають меж, оскільки Organic & Natural виграшний не тільки для корпоративних сайтів (див. рис. 1.6), але й слугує справжньою прикрасою сторінок новин чи веб-ресурсів для бізнесу.



Рисунок 1.6 – Стиль Organic & Natural у веб-дизайні

Зважаючи на велику кількість різноманітних стилів та напрямків розробки та оформлення веб-ресурсів, дизайнер в першу чергу те створює те, що замовив клієнт. За будь-яких умов роботу виконують, враховуючи побажання і особисті потреби замовників. Це значить, що кожен веб-ресурс, не дивлячись на чітко задану стилістику і спрямованість – в кінцевому результаті має неповторну візуалізацію та своєрідну унікальність. На основі унікальності можна розробити особистий фірмовий стиль, який уможливить створити ознаку впізнаваності не тільки у повсякденному житті, а й допоможе у розвитку бізнесу чи збільшенні продаж в майбутньому.

Таким чином, від вдалого вибору веб-дизайну суттєво залежить не тільки привабливість оболонки веб-ресурсів, але й узагальнена популярність серед аудиторії та продуктивність сайту, що є допоміжним інструментом розвитку та ведення професійної діяльності.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

1.4 Постановка задачі кваліфікаційної роботи

Аналіз різновидів веб-ресурсів показує, що в сучасному світі бурхливого розвитку інформаційних технологій веб-розробка не втрачає своєї актуальності. Немає такої людини, яка б сьогодні не користувалася інтернет-ресурсами для пошуку та отримання інформації, оформлення послуги, спілкування, здійснення покупок та ін.

Ефективну саморекламу досить успішно забезпечують сайти-візитки, портфоліо. Проте поряд із наповненням веб-ресурсів ключову роль відіграє дизайн. Адже зручний, привабливий дизайн ніколи не буде поза увагою й сприятиме широкому охопленню аудиторії. Таким чином, грамотно оформлене портфоліо уможливить зарекомендувати себе як професіонала, швидше знайти бажану роботу, створить конкуренцію серед роботодавців.

Оскільки життя є швидкоплинним, для економії часу на вивчення структури веб-ресурсу популярністю користуються веб-розробки у стилі мінімалізму, де акцент зроблено на важливих деталях, без зайвої завантаженості. Зважаючи на зазначене, сформульовано мету дослідження – проєктування веб-сайту інформаційного ресурсу із мінімалістичним дизайном. На основі проведеного аналізу впливають такі задачі, що забезпечують досягнення мети:

- проаналізувати композиційні та естетичні аспекти мультимедійного дизайну;
- проаналізувати технології розробки інформаційних веб-ресурсів;
- сформулювати вимоги до розробки проєкту;
- розробити архітектуру веб-ресурсу;
- спроектувати веб-портфоліо;
- дослідити та обґрунтувати техніко-економічні показники розробки проєкту.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

2 ТЕХНОЛОГІЯ РОЗРОБКИ ВЕБ-САЙТІВ ІНФОРМАЦІЙНИХ РЕСУРСІВ

2.1 Композиційно-естетичні аспекти мінімалізму у веб-дизайні

В теперішній час попри широку варіативність дизайну веб-ресурсів, замовники хочуть у кінцевому результаті отримати щось абсолютно нове, яке обов'язково повинно містити таких три складових: зручний та зрозумілий функціонал, бажання повернутися на сторінку, а також, що найголовніше, – щоб контент веб-ресурсу активно себе продавав. Таких цілей можна досягти, якщо кваліфікація і досвід веб-дизайнера відповідають загальноприйнятим технічним запитам та стандартам у галузі веб-розробки, зокрема: відповідність сайту його призначенню, відповідність стилю щодо вибору графіки, колірної моделі, контенту, коректне відображення у всіх браузерях та на різних пристроях.

У мінімалістичному дизайні взаємозв'язок простору та об'єктів є ключовим аспектом. Вільний простір та розташування об'єктів визначають загальний вигляд дизайну. Однією з ключових рис мінімалістичного дизайну є використання вільного простору. Це означає, що об'єкти розташовані так, щоб вони не здавалися перевантаженими або перенасиченими. Простір навколо об'єктів дозволяє їм «дихати», надаючи дизайну відчуття легкості і простоти [2].

Використовуються об'єкти з мінімальною кількістю деталей. Це можуть бути геометричні фігури, базові кольори та прості текстові блоки. Об'єкти мають бути чіткими та простими, відображаючи сутність без зайвих відволікаючих деталей. У мінімалістичному дизайні можна зустріти як симетричні, так і асиметричні композиції. Симетрична композиція додає рівновагу та порядок, тоді як асиметрична може створювати динаміку та цікаві візуальні точки акценту.

На сьогодні існує 4 види типів графіки, зокрема, растрова графіка, векторна, фрактальна та 3D графіка (тривимірна графіка). У растровій «для відображення картинки на екрані зображення збирається з нескінченної кількості пікселів в своєрідну сітку з кольорових квадратів, побачити яку можна при збільшенні растрового зображення. Векторна графіка складається з простих геометричних

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						27
Змн.	Арк.	№ докум.	Підпис	Дата		

об'єктів, які будуються на основі зазначених координат в заздалегідь обраній векторній програмі» [18]. 3D графіка – це комплексний набір засобів, який уможливорює із двовимірних об'єктів отримати об'ємні та візуально наповнені. Фрактальна графіка – це покрокове повторення геометричних об'єктів з метою отримання єдиного складного зображення у вигляді композиції. Найбільш популярною вважають векторну графіку. Рідше використовують растрову графіку. 3D та фрактальну застосовують часто лише при додаванні декоративних елементів.

У мінімалістичному дизайні вибір шрифтів є ключовим елементом, який може впливати на сприйняття користувача та створювати відчуття бренду. Шрифти в мінімалістичному дизайні мають бути простими та легко читабельними. Чіткість літер та їхніх форм важлива для забезпечення зрозумілості повідомлень. Зазвичай, у мінімалістичних проектах використовують один або два шрифти з різними вагами для створення різних акцентів. Шрифт повинен гармонійно поєднуватися з іншими використовуваними у проекті елементами, такими як кольори, форми та графіка. Важливо, щоб шрифт доповнював і підкреслював загальний стиль проекту.

Мінімалістичний дизайн може використовувати контраст між кольорами, текстурами та формами для створення візуального інтересу. В той же час, гармонійні кольорові палітри та взаємодія між об'єктами можуть створювати спокій і внутрішню рівновагу. Правильно підібрана колірна гама уможливорює налаштування користувача сайту на потрібний настрій.

Як відомо, до теплих кольорів відносять: червоний, помаранчевий, жовтий, включаючи різні відтінки коричневого. До візуально холодних належать зелений, синій, фіолетовий та усі відтінки сірого. Теплі кольори, зазвичай, вважають життєрадісними, доброзичливішими, стимулюючішими до дій, а холодні є спокійними, розслаблюючими, що налаштовують на концентрацію уваги на потрібних деталях.

«Теорія кольору в веб-дизайні є темою широкою, що потребує багато практики для розуміння особливостей її правильного застосування. Вибір кольору

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

залежить від середовища, специфіки роботи і замовлень самого веб-дизайнера, оскільки на кольори і відтінки людина дивиться трохи інакше, ніж ті ж самі кольори сприймає комп'ютер» [27].

Геометричні форми грають ключову роль у створенні виразних та естетичних композицій. Вони є основними будівельними блоками мінімалістичного стилю. Ось деякі типові геометричні форми та їх символіка в мінімалістичному мультимедійному дизайні:

– коло може вказувати на єдність, вічність та гармонію. Його симетрія і безкінечність роблять його популярним елементом для створення відчуття спокою та рівноваги у мультимедійних проектах;

– квадрат символізує порядок, стабільність та справедливість. У мінімалістичному дизайні він може використовуватися для відображення тверджень та надійності. Його чіткі лінії та рівні сторони можуть створювати враження порядку і рівноваги;

– трикутник може вказувати на рух, динаміку та енергію. Ця форма може бути використана для передачі динамічних аспектів мультимедійного контенту. Також, в залежності від орієнтації (вершиною вгору або вниз), він може вказувати на піднесеність або стабільність;

– прямокутник може символізувати порядок і структуру. У мінімалістичному дизайні він може використовуватися для відображення об'єктів і дійсності в їхній найбільш простій та компактній формі.

У мінімалістичному дизайні прості форми, такі як кружки, квадрати та трикутники, можуть представляти ідеї та концепції у найбільш зрозумілому та виразному способі. Наприклад, круг може символізувати спільність, а квадрат - стабільність. Прості символи можуть бути використані для представлення глибоких або складних понять. Їхня спрощена природа може допомогти зрозуміти ідею за допомогою мінімуму деталей.

Використання мінімалістичних медіа-елементів є ключовим для створення естетичної та зрозумілої композиції. Це означає обмеження кількості деталей, які

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

включаєте в зображення, відео чи звук, з метою збереження простоти та ясності повідомлення.

Переважає використання малої кількості кольорів та обмеженої кількості деталей у зображеннях. Вони можуть бути абстрактними або містити мінімальний набір об'єктів. Використання простих сценаріїв та зйомок. Виділення основних об'єктів або елементів у відео та уникання зайвого руху чи динаміки, якщо вона не несе важливого значення. Застосування простих звуків, таких як тихі мелодії, єдині звукові ефекти або спокійні музичні теми. Звертається увага на ритм та темп, щоб вони підкреслювали спокійну та гармонійну атмосферу.

В свою чергу, мінімалістичні медіа-елементи мають такий емоційний вплив на глядача [28]:

- спокій та розслаблення, оскільки вони не перенасичені деталями та кольорами. Глядач може вільно сприймати інформацію без надмірного стимулювання;
- концентрація на важливому, оскільки відсутність зайвих деталей дозволяє глядачеві краще зосередитися на тому, що показано;
- професійний вигляд, що часто асоціюється зі стилем та впевненістю.

Інтерактивні елементи можуть бути використані для цікавішого та захоплюючого залучення глядача. Анімація може використовуватися для надання контексту елементам. Застосування якроінтеракції як невеликих анімацій або змін, які відбуваються у відповідь на конкретні дії користувача. Інтерактивність, анімація та мікроінтеракції використовуються не тільки для створення цікавого вигляду, але й для покращення досвіду користувача та забезпечення зручності взаємодії веб-сайтом.

Таким чином, мінімалістичний дизайн відзначається простотою, ясністю та функціональністю. Цей стиль не лише створює естетично приємний вигляд, але й сприяє кращому сприйняттю інформації та покращує досвід користувача. Використання мінімалістичних принципів та інтерактивних рішень у мультимедійному дизайні може забезпечити ефективну та емоційно насичену

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						30
Змн.	Арк.	№ докум.	Підпис	Дата		

взаємодію з аудиторією, зробити інформацію зрозумілішою та легкою для запам'ятовування.

2.2 Засоби проектування веб-ресурсів

Технології веб-розробки належать до багатьох мов програмування та інструментів, які використовуються для створення динамічних та повнофункціональних веб-сайтів та програм. Front-end (інтерфейсні) технології призначені для «клієнтської сторони» веб-сайту або програми. Вони використовуються для розробки інтерактивних компонентів сайту та створення елементів, які бачать користувачі і з якими взаємодіють. Сюди входять кольори та стилі тексту, зображення, кнопки та меню навігації.

Back-end (серверні) технології – внутрішні технології, призначені для "серверної частини" сайту. Вони зберігають та впорядковують дані та стежать за тим, щоб на інтерфейсі все працювало. Back-end технології також використовуються для оптимізації основних бізнес-процесів. У випадках, коли у великих об'ємах даних, які потрібно опрацювати, можна запустити скрипт у серверній частині, щоб створити змістовний звіт у зовнішній частині.

Клієнтська частина (фронтенд) відіграє ключову роль у взаємодії з кінцевим користувачем, представляючи інтерфейс, через який користувач взаємодіє з аплікацією. HTML та CSS є основою для створення структури та дизайну веб-сторінок, де HTML задає каркас веб-сторінки, а CSS відповідає за стилізацію та візуальне представлення елементів інтерфейсу. JavaScript, разом з бібліотекою React, відіграє центральну роль у створенні інтерактивного користувацького досвіду, дозволяючи розробляти односторінкові аплікації (SPA) з плавною взаємодією без необхідності перезавантаження сторінки [26].

React сприяє ефективному управлінню станом компонентів і оптимізує процес оновлення DOM, що робить веб-аплікацію швидкою та відгукованою. Використання сучасних інструментів та методологій, таких як Webpack для збірки

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						31
Змн.	Арк.	№ докум.	Підпис	Дата		

модулів та Babel для транспіляції сучасного JavaScript, забезпечує сумісність коду з різними браузерами та поліпшує загальну продуктивність фронтенду [22, 32].

Серверна частина (бекенд) є невід'ємною частиною веб-аплікацій, відповідаючи за логіку обробки даних, взаємодію з базами даних, а також за обробку запитів, що надходять від клієнтської частини. У цьому контексті, для розробки бекенду існує Node.js технологія – потужне середовище виконання JavaScript на сервері, що дозволяє використовувати JavaScript для написання серверного коду [20, 30]. Це сприяє єдності мови програмування як на клієнтській, так і на серверній стороні, що полегшує розробку та підтримку проєктів.

Node.js відомий своєю високою продуктивністю завдяки неблокуючій моделі вводу-виводу та асинхронному виконанню запитів, що дозволяє обробляти велику кількість запитів одночасно без значного збільшення навантаження на сервер. Це робить Node.js ідеальним рішенням для розробки масштабованих мережевих додатків, таких як веб-сервіси, API, а також додатки в реальному часі [20].

Для взаємодії з базами даних Node.js пропонує широкий вибір драйверів та ORM (Object-Relational Mapping) бібліотек, які спрощують роботу з різними СУБД, такими як MongoDB, PostgreSQL, MySQL, тощо. Це дозволяє розробникам ефективно організувати зберігання даних, їх запити та обробку, забезпечуючи гнучкість та ефективність в роботі з даними [24].

Крім того, екосистема Node.js включає велику кількість модулів та бібліотек, доступних через менеджер пакетів npm, що дозволяє розширювати функціонал додатку, інтегрувати зовнішні сервіси та використовувати найсучасніші розробки спільноти. Використання Node.js для бекенд-розробки в інформаційних системах забезпечує не лише високу продуктивність та масштабованість, а й гнучкість у реалізації бізнес-логіки, забезпечуючи швидке та ефективне впровадження нових функцій та сервісів. Архітектуру бібліотеки представлено нам рисунку 2.1.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

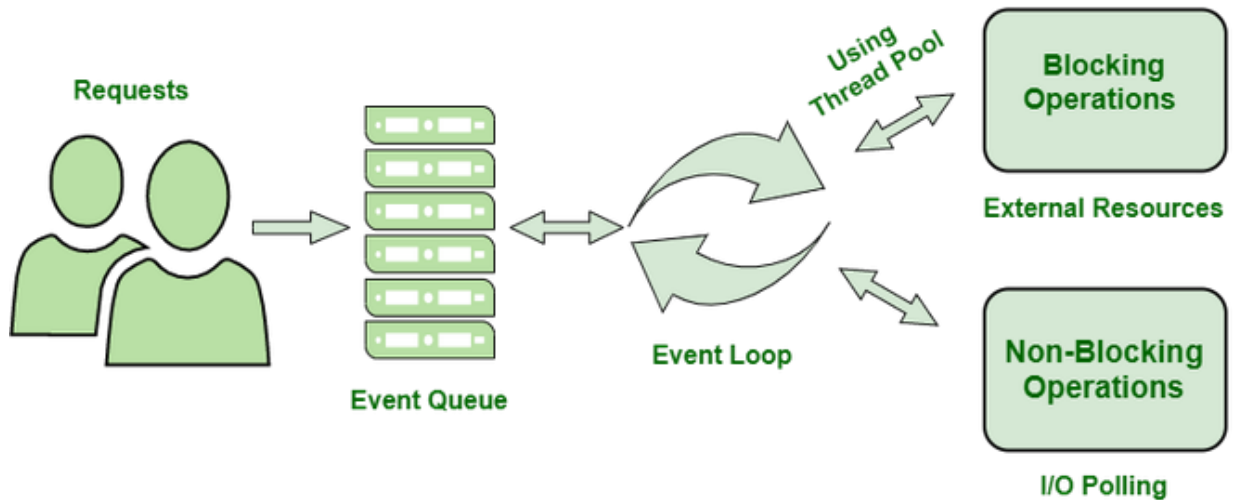


Рисунок 2.1 – Архітектура Node

Архітектура серверної частини на базі патерну Модель-Представлення-Контролер (MVC), широко застосованого у Node.js додатках, забезпечує ефективне розмежування відповідальності між компонентами системи. Модель в цій архітектурі відіграє ключову роль у визначенні структури даних аплікації та взаємозв'язків між різними елементами бази даних. Застосування шаблонів проєктування, таких як ORM, дозволяє розробникам працювати з даними бази даних як з об'єктами JavaScript, спрощуючи та оптимізуючи взаємодію з даними [30].

Представлення у Node.js відповідають за генерацію користувацького інтерфейсу та відображення даних, отриманих від сервера, у вигляді HTML-сторінок або інших форматів виводу (наприклад, JSON для API). Разом з технологіями на стороні клієнта (HTML, CSS, та JavaScript) та шаблонізаторами, такими як EJS або Handlebars, розробники можуть створювати динамічні та інтерактивні веб-сторінки.

Контролери у Node.js відіграють роль посередника між моделями та представленнями, обробляючи логіку аплікації, взаємодію з моделями для отримання або модифікації даних та вибору відповідних представлень для виводу користувачу. Контролери спрямовують запити від користувачів до системи, обробляють введені дані та керують відповідями сервера.

Цей підхід дозволяє створювати добре структуровані та легко підтримувані серверні додатки, оптимізувати повторне використання коду та підвищити гнучкість розробки за рахунок чіткого відокремлення відповідальності між компонентами. Використання баз даних, таких як MongoDB або PostgreSQL у поєднанні з Node.js і архітектурою MVC, забезпечує потужні та гнучкі інструменти для створення масштабованих, високопродуктивних веб-додатків, здатних задовольняти сучасні вимоги до швидкості та якості обробки даних [20].

Angular – це інтерфейсний фреймворк, який підтримує Google. У порівнянні з іншими інтерфейсними фреймворками, це найбільш повне рішення, тому що у нього найбільше функціональних можливостей. Завдяки своїй надійності та глибокій функціональності він використовується великими корпораціями та компаніями, яким потрібне потужне інтерфейсне рішення [21].

React.js є однією з найпопулярніших інтерфейсних бібліотек. З використанням React.js пов'язане довге і складне навчання, але воно виключно при створенні інтерфейсів (UI).

Vue.js – спеціально призначена бібліотека для створення інтерфейсу користувача. У порівнянні з React.js, Vue.js простий та зручний у використанні. Він має меншу спільноту, на яку можна опиратися для усунення несправностей і підтримки, але це трохи компенсується його швидкістю і відносно простим навчанням [12, 21].

Бази даних – це місце для зберігання даних. Однак, це не просто безсистемні цифрові картотеки. Вони забезпечують узгоджену, організовану структуру для зберігання та отримання великих обсягів даних. Бази даних SQL використовують мову структурованих запитів SQL визначення даних і управління ними. Бази даних NoSQL не використовують мову структурованих запитів, тому можна зберігати дані різними способами, без попереднього визначення їхньої структури. NoSQL дозволяє збереженим даним мати власну унікальну структуру, яка не визначається самою структурою бази даних.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						34
Змн.	Арк.	№ докум.	Підпис	Дата		

2.3 Технічне рішення розробки

Розробка будь-якого веб-ресурсу включає реалізацію певних етапів із відповідними результатами та документацією. Структурну схему етапів розробки інформаційних веб-ресурсів наведено у КР.КІ.8872531.00.00.000.С1.

При створенні веб-сайту інформаційного ресурсу визначення вимог є критично важливим етапом, що забезпечує чітке розуміння цілей, аудиторії, а також функціональних і нефункціональних вимог. Основна мета такого проекту полягає в саморекламі, залученні широкої аудиторії, створенні зручного інформаційного ресурсу для зацікавлених осіб (роботодавців) та покращенні взаємодії з ними через різноманітні інтерактивні функції.

Розуміння цільової аудиторії є ключовим для розробки ефективної стратегії контенту та дизайну. Цільова аудиторія – це роботодавці, які зацікавлені у кваліфікованому працівнику. Визначення їхніх потреб і переваг допоможе забезпечити максимальну корисність і зручність розробки.

Функціональні вимоги до сайту охоплюють широкий спектр характеристик, необхідних для забезпечення повноцінної роботи ресурсу. Серед них – привітання, про кандидата, технології, проекти, контакти. Такі функції вимагають від розробників уваги до деталей та здатності інтегрувати відповідні технічні рішення. Що стосується дизайну, то його слід виконати у стилі мінімалізму для забезпечення концентрації уваги на основних деталях.

Нефункціональні вимоги стосуються загальних стандартів якості, яким має відповідати сайт, включаючи швидкість завантаження, продуктивність, доступність на різних пристроях та захист користувацьких даних. Важливо також забезпечити зручність навігації по сайту, адаптивність мінімалістичного дизайну для комфортного перегляду на мобільних пристроях та високий рівень безпеки для захисту від потенційних кібератак.

Успішне визначення та реалізація цих вимог дозволить створити ефективний і залучаючий інформаційний веб-ресурс, який стане надійним

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		

джерелем інформації про потенційного кандидата та для зацікавлених роботодавців.

Технічне завдання (ТЗ) є фундаментальним документом у процесі розробки сайту, який детально описує всі аспекти проекту, від цілей до конкретних технічних вимог. Воно допомагає забезпечити розуміння проекту та його кінцеві цілі. Для портфолію ТЗ включатиме такі основні розділи [23].

Цілі та завдання проекту:

- забезпечення актуальної інформації про кандидата;
- створення зручного ресурсу для залучення роботодавців та інших зацікавлених осіб.

Аудиторія: визначення основних цільових груп сайту.

Функціональні вимоги:

- розділ про мене: освіта, кваліфікація, стажування, сертифікати;
- розділ технологій: короткий сучасних опис технологій, з якими наявний досвід роботи;
- проекти: опис проектів, які успішно виконано із участю кандидата або самостійно;
- контакти: розділ із контактами для забезпечення зворотного зв'язку;
- розділ для коментарів, так званий чат.

Нефункціональні вимоги:

- продуктивність: швидке завантаження на всіх типах пристроїв;
- масштабованість: можливість легкого розширення функціоналу та обробки зростаючого трафіку;
- безпека: захист від хакерських атак, шифрування даних;
- сумісність з браузером: коректна робота в усіх популярних браузерах.

Технічні вимоги [5-7]:

- вибір технологічного стеку (мови програмування, фреймворки, бази даних);
- вимоги до хостингу (тип хостингу, параметри сервера);
- SEO-оптимізація для підвищення видимості в пошукових системах.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						36
Змн.	Арк.	№ докум.	Підпис	Дата		

Етапи розробки та терміни: детальний план розробки з визначенням етапів проекту та орієнтовними термінами їх виконання.

Бюджет проекту: попередній розрахунок вартості розробки, включаючи роботу дизайнерів, розробників, тестувальників та інших учасників проекту.

Складання детального ТЗ є критично важливим для успішної реалізації проекту, оскільки воно слугує основою для узгодження всіх аспектів роботи між замовником та виконавцями.

Грамотно складене портфоліо зацікавить роботодавця. З його допомогою він зрозуміє, наскільки ви відповідаєте вимогам компанії та яку користь можете принести. Не варто сприймати портфоліо як просте сховище прикладів робіт – його важливо структурувати, доповнити контактами [8].

Титульна сторінка – це перша сторінка, яку дивиться роботодавець, тому потрібно вказати: прізвище, ім'я, по батькові, спеціальність, телефон, соціальні мережі та інші контакти, невелике вітання. Привітання має бути коротким: чим менше стороннього тексту, тим більше роботодавець зосереджений на кейсах.

Для інформації про себе слід виділити окрему сторінку. Стисло перерахувати професійні досягнення, з якими проектами працювали, що вдається найкраще. Якщо є посилання на блог, інтерв'ю чи подкаст, можна залишити його наприкінці тексту. Роботодавець ознайомиться із матеріалом, якщо захоче дізнатися про кандидата краще.

У кейсах потрібно використовувати не лише фотографії, графіки, скріншоти, таблиці та прототипи, а й короткі описи. Наприклад, як називається проект, хто замовник, якою є мета проекту, як проходив процес роботи, які результати отримали.

Портфоліо також просуває послуги. Тому наприкінці варто оформити заклик до дії. Наприклад, можна запропонувати знижку, надати гарантію або безкоштовну консультацію, якщо замовник чи роботодавець залишить заявку.

Наприкінці портфоліо залишають контакти для зворотного зв'язку:

- номер телефону;
- посилання на месенджери та соціальні мережі;

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						37
Змн.	Арк.	№ докум.	Підпис	Дата		

- дреса електронної пошти.

Кандидату не обов'язково вказувати всі контакти: трьох способів зв'язку достатньо.

Для розробки простого та зрозумілого веб-інтерфейсу слід опиратися на такі принципи:

- використання простору для організації зв'язаних елементів;
 - розбивка інформації на менші групи зв'язаних за змістом елементів
- допомагає структурувати та організувати інтерфейс. Це прискорює та спрощує його розуміння та запам'ятовування людьми.

Групувати елементи можна такими способами:

- розміщувати зв'язані елементи в одному контейнері;
- розташовувати зв'язані елементи поряд на сторінці;
- зробити так, щоб зв'язані елементи були подібними;
- вирівняти зв'язані елементи у безперервній лінії.

Використання контейнерів – це найпотужніша підмога для угруповання елементів інтерфейсу, але вона може приносити зайве безладдя. Слід шукати можливості задіяти інші способи групування. Вони нерідко виявляться тоншими і допоможуть спростити дизайн.

Використання простору є дуже ефективним та простим способом групувати зв'язані елементи. Для більш виразного відображення груп також можна поєднувати різні підходи.

Узгодженість у дизайні UI означає, що подібні елементи виглядають та працюють подібним чином. Це має бути присутнім у розробці поточного продукту й при його порівнянні з іншими успішними аналогами. Така прогнозована функціональність підвищує юзабіліті та зменшує кількість помилок, оскільки людям не потрібно заново з'ясовувати, як усе працює [14].

Якщо елементи подібні, люди чекатимуть від них подібної функціональності. Тому слід робити їх відповідними для зручності візуального сприйняття. І навпаки, елементи, що мають різне призначення, мають виглядати відмінними один від одного.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						38
Змн.	Арк.	№ докум.	Підпис	Дата		

Чи не вся інформація інтерфейсу має однаковий рівень значущості. Слід намагатися надати інформацію у порядку її важливості, виділяючи значніші елементи. Виразний порядок значущості або візуальна ієрархія, допомагає швидко сканувати інформацію і фокусуватися на областях, які найбільш цікавлять. Підвищити естетику можна також за рахунок створення ефекту порядку. Реалізувати виразну візуальну ієрархію можна за допомогою відмінностей у розмірі, кольорі, контрасті, відступах, розташуванні та глибині.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

3 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОГО ВЕБ-РЕСУРСУ

Для розробки бекенду обрано Node.js — потужне середовище виконання JavaScript на сервері, що дозволяє використовувати JavaScript для написання серверного коду. Для розробки інтерфейсу користувача застосовано бібліотеку React, що використовує концепцію віртуального DOM (Document Object Model), яка дозволяє відображати зміни на сторінці без перезавантаження її повністю. Технологія досить популярна серед розробників, оскільки пропонує простий та ефективний спосіб створення складних інтерфейсів.

3.1 Аналіз технологічного стеку

Інформаційний веб-ресурс призначений для спрощення управління особистими здобутками, надаючи користувачам зручний онлайн-доступ до них та детальної інформації про потенційного кандидата [1].

Фундаментальною частиною цього проекту є створення прозорого і ефективного інтерфейсу із мінімалістичним дизайном, який дозволяє користувачам легко навігувати системі та знаходити необхідну інформацію. Важливим аспектом є також забезпечення можливості зворотного зв'язку та обговорення поточних питань у чаті.

Технологічний стек проекту було обрано з огляду на потреби сучасного веб-сайту інформаційного ресурсу:

- React забезпечує багатий користувацький інтерфейс, дозволяючи швидко і ефективно реагувати на дії користувачів та забезпечує гнучкість у внесенні змін до компонентів інтерфейсу.
- Node.js є основою серверної частини, оскільки це середовище виконання для JavaScript дозволяє створювати надійні та масштабовані мережеві

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

додатки, ідеально підходить для реалізації серверної логіки та взаємодії з базою даних.

- Express.js використовується для створення веб-сервера та розробки API, надаючи прості інструменти для маршрутизації запитів і обробки відповідей.
- MySQL та PHP використовуються для зберігання і обробки даних, забезпечуючи надійне та ефективне управління документами у форматі JSON, які є основою для даних про кандидата, що вимагають гнучкості та швидкого доступу.

Фронтенд, створений з використанням бібліотеки React, надає інтуїтивно зрозумілий та швидкісний користувацький досвід. Завдяки компонентному підходу React із можливістю використання веб-запитів до API, вдалося відмовитися від традиційного серверного рендерингу сторінок, що сприяло значному збільшенню швидкості загрузки інтерфейсу.

Серверна частина побудована на базі Node.js з використанням Express.js, що забезпечує легкість розробки мережевих додатків. Node.js, завдяки своїй подієво-орієнтованій архітектурі та високій продуктивності, ідеально підходить для обробки запитів в реальному часі, що є ключовим для пошуку інформації, де оперативність критично важлива. Express.js допомагає у маршрутизації запитів та обробці відповідей, надаючи простий і зрозумілий інтерфейс для розробки API.

Базу даних реалізовано з використанням NoSQL рішення, яке забезпечує гнучкість у зберіганні даних. Такий підхід дозволяє легко змінювати структуру даних, вводити нові типи даних і ефективно масштабувати систему без втрати продуктивності.

Використання архітектурного патерну Модель-Вид-Шаблон (MVT) у розробці фронтенду та бекенду сприяє легкій інтеграції між різними частинами системи. Розділення логіки додатку, його інтерфейсу та даних, що відображаються, дозволяє гнучко розвивати кожен частину незалежно одна від одної та забезпечує легшу підтримку та масштабування системи в цілому. На рисунку 3.1 показано структурну WEB API схему взаємодії бази даних.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						41
Змн.	Арк.	№ докум.	Підпис	Дата		

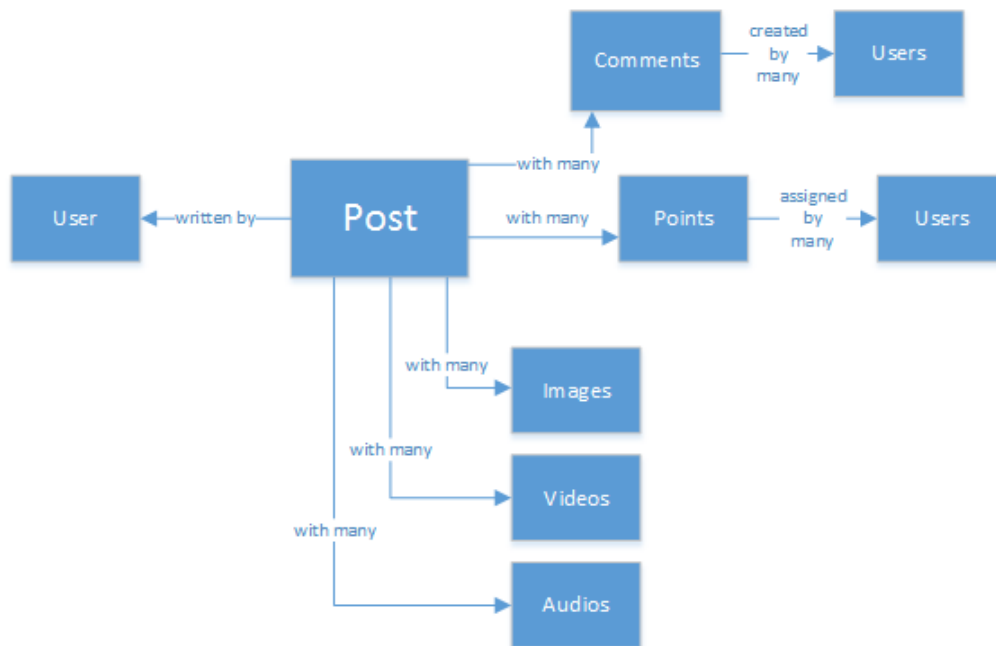


Рисунок 3.1 – Структурна WEB API схема взаємодії бази даних

WEB API, реалізоване у цій системі, діє як центральна точка зв'язку між клієнтською та серверною частинами, забезпечуючи гнучке управління даними та їх оперативне оновлення. Всі взаємодії з даними відбуваються через чітко визначені API виклики, що дозволяє легко масштабувати систему та інтегрувати нові сервіси.

Така сучасна архітектура дозволяє не тільки забезпечити високу продуктивність і надійність інформаційного веб-ресурсу, але й створює міцну основу для її майбутнього розвитку та інтеграції з додатковими функціональними можливостями.

3.2 Реалізація інформаційного веб-ресурсу

Під час реалізації інформаційного веб-ресурсу особлива увагу приділено не лише зовнішньому вигляду та інтуїтивно зрозумілому інтерфейсу, але й технічній реалізації функціональних компонентів. Цей підхід гарантував, що кінцевий

продукт не тільки відповідатиме очікуванням користувачів, але й забезпечить стабільність та високу продуктивність системи загалом.

Основу бекенд-розробки складає Node.js в комбінації з Express.js, що дозволяє створювати ефективні мережеві додатки, які можуть обробляти велику кількість запитів одночасно без втрати продуктивності. Використання цих технологій сприяє легкості інтеграції з різноманітними базами даних і зовнішніми API, що є критично важливим для систем, які управляють динамічними даними потенційних кандидатів для самореклами.

Фронтенд, створений на основі React, забезпечує не тільки високу швидкість роботи інтерфейсу, але й гнучкість у впровадженні змін. Використання компонентного підходу React дозволяє ефективно управляти станами інтерфейсу та реагувати на взаємодії користувачів у реальному часі, що є незамінним для відображення актуальних результатів власників портфоліо.

Таким чином, детальний огляд ключових компонентів дозволить не тільки оцінити якість та ефективність вже імплементованих рішень, але й визначити потенційні області для подальших удосконалень. Таким чином, варто зосередимось на конкретних технічних аспектах реалізації бекенду та фронтенду, а також на взаємодії цих складових, що забезпечують безперебійну роботу веб-ресурсу [34]. Загальну структуру веб-портфоліо наведено на КР.КІ.8872531.00.00.000.С2.

Модуль Головна сторінка (Home Module) є головним інтерфейсом користувача і відіграє ключову роль у взаємодії з відвідувачами сайту. Це перше, що вони бачать, відвідуючи портфоліо, тому важливо, щоб дизайн та функціональність були виконані на високому рівні. Вигляд модуля на клієнтській частині продемонстровано на рисунку 3.2.

Основна структура модуля Головна сторінка включає привітання від кандидата, його фотографія, а також логотип. Варто відзначити, що ресурс передбачає перегляд інформації на двох мовах – українською і англійською, що уможливорює залучення ширшої аудиторії.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

Такий модуль є серцем інформаційної системи, що об'єднує візуальний дизайн та користувацький досвід, забезпечуючи легкий та ефективний спосіб для користувачів взаємодіяти з основними ресурсами та функціоналом сайту. Використання CSS для стилізації та React для функціональності робить цей модуль гнучким та легким у масштабуванні, що є важливим аспектом для подальшого розвитку та оновлення системи.

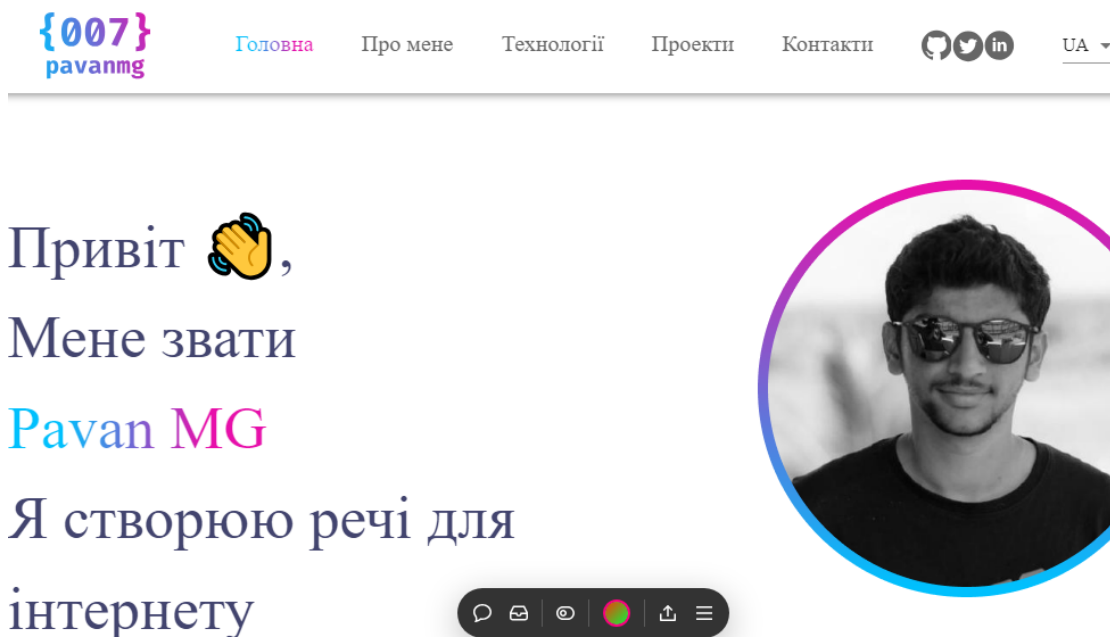


Рисунок 3.2 – Екранна форма вигляду головної сторінки

Модуль Portfolio відіграє ключову роль, надаючи користувачам унікальну можливість переглядати інформацію про кандидата. Завдяки використанню React компонентів, модуль дозволяє забезпечити динамічне оновлення вмісту без потреби перезавантаження сторінки. Коден проєкт доступний через спеціалізований компонент, який імпортує дані з файлу, наприклад Portfolio_data.js, оптимізуючи тим самим управління контентом.

Кожен проєкт представлений окремою карточкою з фото, заголовком, та коротким описом, що створено за допомогою компонента Card. Ці карточки не просто інформують про досягнення, а й надають візуальний вступ до них, залучаючи при цьому користувачів до перегляду. Головною функцією цього

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						44
Змн.	Арк.	№ докум.	Підпис	Дата		

компонента є забезпечення зручного доступу до проєктів та детальної інформації про них, що дозволяє користувачам глибше пізнати кандидата. Інтерактивність модуля виявляється в можливості кліка по карточці, що приводить до повного опису. Екранну форму фрагмента програмного коду, який відповідає за реалізацію описаного модуля наведено на рисунку 3.3. Вигляд програмного модуля на клієнтській частині показано на рисунку 3.4.

```

1 import React, { useState } from 'react'
2
3
4 const Card = (props) => {
5   const [modal, setModal] = useState(false);
6   const toggleModal = () => {
7     setModal(!modal);
8   };
9
10  return (
11    <div className="box box_shadow">
12      <div className="img">
13        <iframe
14          src={props.iframeSrc}
15          title="Football Match"
16          frameborder="0"
17          allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope; picture-in-picture; web-share"
18          referrerPolicy="strict-origin-when-cross-origin"
19          allowFullScreen
20          style={{ width: '100%', height: '100%' }}
21        />
22      </div>
23      <div className="category_d_flex">
24        <span onClick={toggleModal}>{props.category}</span>
25        <label>
26          <i className="far fa-heart"></i>
27          {props.totalLike}
28        </label>
29      </div>
30      <div className="title">
31        <h2 onClick={toggleModal}>{props.title}</h2>
32        <a href="#">{props.className} arrow</a>
33        <i className="fas fa-arrow-right"></i>
34      </div>
35    </div>
36  );
37
38

```

Рисунок 3.3 – Приклад реалізації модуля портфоліо

Проекти

Те, що я побудував до цього часу



Проект "Плитка" тут

Це приклад опису проєкту випадкові речі тут в описі Це приклад проєкту генератор lorem ipsum для фіктивного контенту

Технічний стек : HTML , JavaScript, SASS, React

Проект "Плитка" тут

Це приклад опису проєкту випадкові речі тут в описі Це приклад проєкту генератор lorem ipsum для фіктивного контенту

Технічний стек : HTML , JavaScript, SASS, React

Проект "Плитка" тут

Це приклад опису проєкту випадкові речі тут в описі Це приклад проєкту генератор lorem ipsum для фіктивного контенту

Технічний стек : HTML , JavaScript, SASS, React

Рисунок 3.4. – Екранна форма сторінки проєктів

										Арк.
										45
Змн.	Арк.	№ докум.	Підпис	Дата						

Компонент Testimonials важливий для створення позитивного іміджу кандидата. Він не тільки повертає увагу до якості та значення досягнень, але й демонструє доказ успішності через наявність рекомендацій, сертифікатів. Такий підхід підсилює довіру роботодавців до кандидата та стимулює нових відвідувачів до глибшого пізнання можливостей кандидата.

Інтегрований з використанням React, компонент функціональні компоненти, що динамічно керують відображенням відгуків. Функція автоматичного перегортання, реалізована за допомогою useEffect, дозволяє користувачам плавний перегляд без необхідності активного втручання. Екранну форму фрагмента програмного коду, який відповідає за реалізацію описаного компонента представлено на рисунку 3.5.

```

src > Component > Testimonial > Testimonial.js > ...
1 import React, { useEffect, useState } from 'react'
2 import './Testimonials.css'
3 import Slider from './Slider'
4 import TestimonialApi from './TestimonialApi'
5
6 const Testimonial = () => {
7   const [date, setDate] = useState(TestimonialApi)
8   const [index, setIndex] = useState(0)
9
10  useEffect(() => {
11    const lastIndex = date.length - 1
12    if (index < 0) {
13      setIndex(lastIndex)
14    }
15    if (index > lastIndex) {
16      setIndex(0)
17    }
18  }, [index, date])
19
20  useEffect(() => {
21    let slider = setInterval(() => {
22      setIndex(index + 1)
23    }, 3000)
24    return () => clearInterval(slider)
25  }, [index])
26
27
28  return (
29    <section className='Testimonial' id='clients'>
30      <div className='container'>
31        <div className='heading text-center'>
32          <h2>Кажуть наші партнери?</h2>
33          <h3>Будьте намі</h3>
34        </div>
35        <div className='slide'>
36          {date.map((value, valueIndex) => {
37            return <Slide key={value.id} {...value} valueIndex={valueIndex} index={index} />
38          })}
39        </div>
40        <div className='slide_button'>
41          <button className='btn_shadow prev_btn' onClick={() => setIndex(index - 1)}>
42            <i className='fas fa-arrow-left' />
43          </button>
44          <button className='btn_shadow next_btn' onClick={() => setIndex(index + 1)}>
45            <i className='fas fa-arrow-right' />
46          </button>
47        </div>
48      </div>
49    </section>
50  )
51
52
53 }
54
55 export default Testimonial
56

```

Рисунок 3.5 – Приклад реалізації модуля Testimonials

Модуль Contact на інформаційному веб-ресурсі – це інтерфейсна знахідка, яка перевіряє форму зворотного зв'язку у центр комунікації. Розроблений з урахуванням найкращих практик веб-дизайну і використанням реактивних

можливостей React, цей модуль забезпечує ефективний обмін інформацією та сприяє глибшому залученню користувачів. Вбудовані механізми валідації забезпечують точність та актуальність даних. Екранну форму вікна реалізації такого програмного модуля та вигляд його на клієнтській частині наведено на рисунках 3.6, 3.7 відповідно. Екранні форми розділів портфолію, зокрема, технологій, про мене наведено додатку Б.

```

3 const Contact = () => {
24   const formSubmit = (event) => {
25     event.preventDefault()
26     alert(
27       `My name is ${data.fullName},
28       My Phone Number is ${data.phone},
29       My email address is ${data.email},
30       My Subject on ${data.subject},
31       Here is my message I want to say: ${data.message}`
32     )
33   }
34   return (
35     <div>
36       <section className='Contact' id='contact'>
37         <div className='container top'>
38           <div className='heading text-center'>
39             <h2>Контакт зупинилося за хвилю!</h2>
40             <h3>Щоб нас сказати робота з нами, надішліть листа за формою нижче</h3>
41           </div>
42           <div className='content d-flex'>
43             <div className='left'>
44               <div className='box box_shadow'>
45                 <div className='img'>
46                   <img src={contact} alt="" />
47                 </div>
48                 <div>
49                   <p>Ми вдячні за ваш вибір!</p>
50                   <p>Для за'яву з нашим менеджерам зверніться за наступними реквізитами:</p>
51                   <p>Мобільний Телефон: +388 88 88 81</p>
52                   <p>Email: example@gmail.com</p>
53                   <p>А також знайдіть нас у соціальних мережах:</p>
54                   <div className='button f-flex'>
55                     <button className='btn_shadow'>
56                       <i className='fab fa-facebook-f' />
57                     </button>
58                     <button className='btn_shadow'>
59                       <i className='fab fa-instagram' />
60                     </button>
61                     <button className='btn_shadow'>
62                       <i className='fab fa-twitter' />
63                     </button>
64                   </div>
65                 </div>
66             </div>
67             <div className='right box_shadow'>
68               <form onSubmit={formSubmit}>
69                 <div className='f-flex'>
70                   <div className='input row'>
71                     <span>Name</span>
72                     <input type='text' name='fullName' value={data.fullName} onChange={InputEvent} />
73                   </div>
74                   <div className='input row'>
75                     <input type='text' />
76                   </div>
77                 </div>
78               </form>
79             </div>
80           </div>
81         </div>
82       </section>
83     </div>
84   )
85 }

```

Рисунок 3.6 – Приклад реалізації модуля Contact

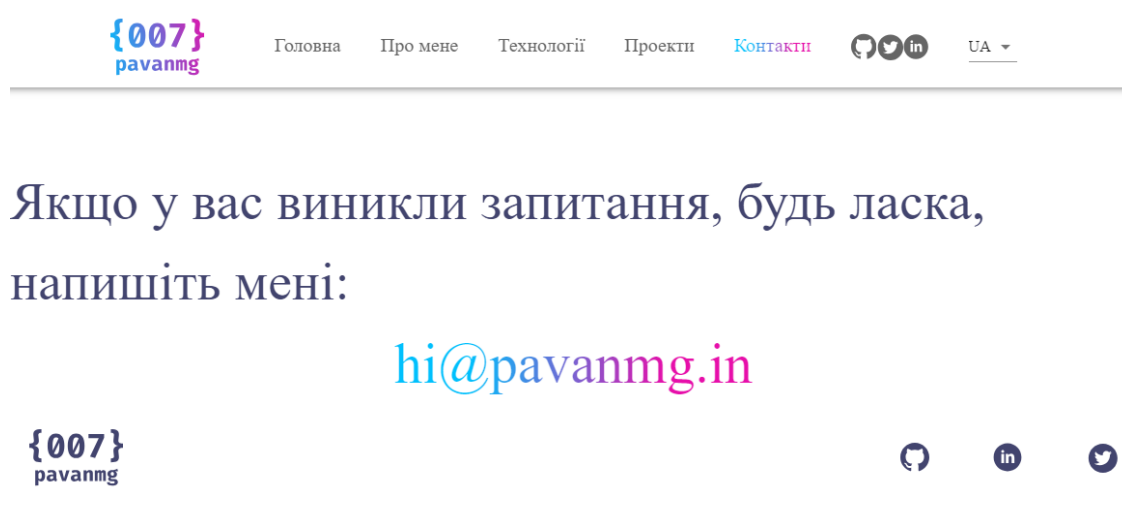


Рисунок 3.7 – Екранна форма вигляду вікна контактів

						КР.КІ.8872531.00.00.000.ПЗ	Арк. 47
Змн.	Арк.	№ докум.	Підпис	Дата			

Таким чином, кожен з цих модулів, будучи ретельно спланованим та реалізованим, сприяє загальному зростанню задоволеності користувачів інформаційного веб-сайту. Надійні технічні рішення, в поєднанні з якісним користувацьким інтерфейсом із мінімалістичним дизайном, створюють міцну основу для самореклами та забезпечення професійної реалізації у майбутньому.

3.3 Тестування інформаційного веб-сайту

Тестування є ключовим етапом у розробці будь-якого програмного забезпечення, включаючи веб-сайт інформаційного ресурсу. Його метою є забезпечення високої якості користувацького досвіду, стабільності системи та відповідності функціональних вимог, аналіз вибезпеченості функціональних вимог, витримування стильових принципів.

Для автоматизації тестування використовували такі інструменти, як Jest для юніт-тестів та Cypress для інтеграційних та енд-ту-енд тестів [16, 34]. Це дозволило автоматизувати перевірку як окремих модулів коду, так і користувацького інтерфейсу в цілому.

Юніт-тестування фокусувалось на перевірці індивідуальних функцій та компонентів системи, зокрема таких як:

- коректність роботи функцій для взаємодії з API;
- відповідність повернутих даних з API очікуваному формату;
- правильність станів компонентів React при різних вхідних даних.

Інтеграційне тестування полягало у перевірці взаємодії між різними частинами системи, такми як:

- взаємодія користувацького інтерфейсу з серверною частиною;
- роботу системи маршрутизації та відображення відповідних компонентів за певними URL;
- функціональність пошуку та відтворення відео.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						48
Змн.	Арк.	№ докум.	Підпис	Дата		

Енд-ту-енд тестування [16] зосереджувалось на сценаріях користування, переконуючись, що всі елементи системи працюють разом без помилок, та що користувацький потік відповідає задумам, стилю мінімалізму, відповідність запитам.

Для документування та відстеження тестування функціональності інформаційного веб-ресурсу розроблено відповідну таблицю тест-кейсів 3.1, що містить розділи очікуваний та фактичний результат, які уможливають здійснити порівняння та зробити відповідні висновки.

Таблиця 3.1 – Тест-кейси для розробленого веб-ресурсу

ID тесту	Опис тесту	Кроки	Очікуваний результат	Фактичний результат	Статус
TC01	Перевірка відображення зображень	1. Перейти на сторінку з проектами. 2. Перевірити чи відображаються актуальні проекти та їх опис	На сторінці відображається коректний матеріал та опис	Відображається актуальний матеріал та опис	Пройдено
TC03	Перевірка доступу до інтерактивних елементів	1. Обрати секцію та інтерактивний елемент 2. Використати обраний елемент	Елемент працює як зазначено	Елемент працює як зазначено	Пройдено

Продовження таблиці 3.1.

TC04	Перевірка функції додавання інформації про кандидата	1. Перейти до розділу керування проектами. 2. Натиснути кнопку "Додати проєкт". 3. Заповнити форму додавання проєкту. 4. Натиснути кнопку "Зберегти"	Проект успішно додано до портфоліо	Новий проєкт відображається у списку проєктів	Пройдено
TC05	Перевірка актуальності і контактів для зворотного зв'язку	Введені дані успішно надіслані на вказану електронну пошту	Форма надіслана успішно	Заповнити всі поля форми зворотного зв'язку. 2. Натиснути кнопку "Надіслати"	Пройдено

Проведені тести демонструють, що розроблений веб-сайт інформаційного ресурсу, зокрема, портфоліо відповідає поставленим технічним вимогам та функціональним очікуванням. Система проявила стабільність і надійність у різних сценаріях користувацької взаємодії. Виявлені в процесі тестування недоліки були виправлені, що забезпечило належну якість кінцевого продукту.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

Основні розрахунки, що стосуються техніко-економічного обґрунтування розробки проекту наведено у додатку В.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Проведені дослідження уможливили розв'язати у кваліфікаційній роботі практичну задачу розробки веб-сайту інформаційного ресурсу. Це у свою чергу дозволило отримати певні практичні результати.

1. Проаналізовано особливості існуючих інформаційних веб-ресурсів й зазначено, що одним із популярних засобів самореклами є портфоліо, яке містить збірку основних досягнень людини.

2. Проаналізовано процедуру проектування дизайну сучасних веб-ресурсів, основні принципи дизайн-проєкту, а також стилі веб-дизайну й показано, що саме правильне і привабливе оформлення сприяє залученню великої аудиторії й популяризує сам проєкт.

3. Досліджено композиційно-естетичні аспекти стилю мінімалізму у веб-дизайні й зазначено, що такому стилю притаманне акцентування на головному, відсутнє перевантаження ресурсу, що досить ефективно для таких інформаційних веб-ресурсів, як портфоліо.

4. Проаналізовано технології проектування веб-ресурсів, що стосуються розробки клієнтської та серверної частини. Відзначено перспективність застосування бібліотеки React, як інструменту, що забезпечує гнучкість та адаптивність до змін.

5. Розроблено технічне завдання, обґрунтовано технологічний стек й описано реалізацію інформаційного веб-ресурсу – портфоліо із зазначенням структурних модулів, їх взаємодії та вигляду на клієнтській частині розробки.

6. Проведено експериментальні дослідження розробленого веб-ресурсу, результатами тестування показано відповідність розробки технічним вимогам, які ставляться до типових об'єктів, функціональність, надійність та стабільність.

7. Обґрунтовано техніко-економічні результати ефективності розробки інформаційного веб-ресурсу.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						52
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вовк О. Б. Методи та засоби підвищення життєздатності веб-сайту як інформаційного продукту: автореф. дис. канд. техн. наук: 05.13.06. Нац. ун-т «ЛП». Львів, 2013. 20 с.

2. Вплив мінімалізму на веб-дизайн: Приклади та натхнення: веб-сайт. URL: <https://www.ranktracker.com/uk/blog/the-impact-of-minimalism-in-web-design-examples-and-inspiration/> (дата звернення 15.05. 2024 р.).

3. Гендін Н. І. Лінгвістичні засоби проектування контенту веб-сайтів // Наукові та технічні бібліотеки. 2008. № 3. С. 5-14.

4. Головань А.А. Побудова навігаційної моделі інформаційного веб-ресурсу: веб-сайт. URL: <https://jitas.donnu.edu.ua/article/view/15421/15327> (дата звернення 23.04.2024 р.).

5. Етапи розробки веб-сайту: веб-сайт. URL: <https://www.fishdigital.agency/etapi-rozrobki-saytu> (дата звернення 2.04.2024 р.).

6. Етапи створення сайту: веб-сайт. URL: <https://webcase.com.ua/uk/blog/iz-chego-sostoit-razrabotka-sajta/> (дата звернення 22.04.2024 р.).

7. Етапи розробки сайту: як пройти шлях від ідеї до релізу: веб-сайт. URL: <https://wezom.com.ua/ua/blog/etapy-razrabotki-sajta> (дата звернення 10.04.2024 р.)

8. Кондратьк С.Ю. Методика формування сайту-портфолію. Програмно-методичне забезпечення та комп'ютерна підтримка діяльності вчителя трудового навчання, технологій. Черкаси: ЧОПОПІ, 2014. 28 с.

9. Методичні вказівки до виконання практичних робіт з дисципліни «Техніко-економічне обґрунтування розробки комп'ютерних систем»/ Н.Я. Савка, І.Р. Паздрій / Під ред. О.М. Березького. Тернопіль: ТНЕУ, 2019. 40 с.

10. Методичні вказівки до оформлення курсових, звітів про проходження практики, випускних кваліфікаційних робіт для студентів спеціальності

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						53
Змн.	Арк.	№ докум.	Підпис	Дата		

«Комп'ютерна інженерія» / І.В. Гураль, Л.О.Дубчак / під ред. О.М. Березького. Тернопіль: ТНЕУ, 2019. 34 с.

11. Методичні рекомендації до виконання кваліфікаційної роботи з освітнього ступеня “Бакалавр” спеціальності 123 «Комп'ютерна інженерія» галузі знань 12 Інформаційні технології / О.М. Березький, Л.О.Дубчак, Г.М. Мельник, Ю.М. Батько, О.Й. Піцун / Під ред. Л.О.Дубчак. Тернопіль: ЗУНУ, 2024. 54 с.

12. Молчанов В. П., Пандорін О.К. Технології розробки WEB-ресурсів: навч. посібник. Харків : ХНЕУ ім. С. Кузнеця, 2019. 130 с.

13. Основні етапи створення сайту: веб-сайт. URL: <https://impulse-design.com.ua/ua/etapy-razrobotki-sajta.html> (дата звернення 01.05.2024 р.).

14. Пасічник О. Г., Пасічник О.В., Стеценко І.В. Основи веб-дизайну : посібник К.: ВНУ, 2008. – 335 с.

15. Пелецишин А. М. Інтернет-технології опрацювання консолідованих інформаційних ресурсів: навч. посіб. Львів : Вид-во Львів. політехніки, 2010. 247 с.

16. Пілат О. Ю. Інформаційна технологія оцінювання якості веб-сайтів : автореф. дис. канд. техн. наук : 05.13.06. Укр. акад. друкарства. Львів, 2015. 24 с.

17. Портфолію. Типи і види, методика формування: веб-сайт. URL: https://bibliotekacoledg.blogspot.com/2017/11/blog-post_7.html (дата звернення 10.05.2024 р.).

18. Сидоренко А.В. Мінімалізм як стильове направлення при розробці дизайну веб-сайтів: дип. бак. робота. К.: КНУТД, 2021. С. 88.

19. Створення інформаційного сайту: веб-сайт. URL: https://www.seotm.com/ua/services/web/informatsiyniy_sayt.html (дата звернення 12.05. 2024).

20. Ткаченко О., Боднар В. Деякі аспекти використання Node.js та MongoDB під час створення освітніх вебзастосунків // Цифрова платформа: інформаційні технології в соціокультурній сфері. 2023. Том 6. No 1. 76-89.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

21. Цюцюра М.І., Криворучко О.В., Жирова Т.О., Котенко Н.О. Сучасні технології тестування і захисту веб-сторінок // Управління розвитком складних систем. 2019. № 39. С. 100-105.

22. Чому React JS - найкраща фронтенд-технологія для стартапів: веб-сайт. URL: <https://www.ranktracker.com/uk/blog/why-react-js-is-the-most-favored-front-end-technology-for-startups/> (дата звернення 10.05.2024 р.).

23. Що входить в розробку сайту? : веб-сайт. URL: <https://voll.com.ua/uk/blog/shho-vxodit-v-rozrobku-sajtu> (дата звернення 15.04. 2024 р.).

24. Які існують технології для розробки сайтів та кому вони підходять? : веб-сайт. URL: <https://icstudio.online/post/tehnologii-dlya-rozrobky-sajtov> (дата звернення 14.04. 2024 р.).

25. Crowder D. A. Building a Web site for Dummies. John Wiley & Sons, 2010. 368 p.

26. HTML, CSS I JavaScript: Основи веб-розробки: веб-сайт. URL: <https://pravda.if.ua/html-css-i-javascript-osnovy-veb-rozrobky/> (дата звернення 20.04. 2024 р.).

27. Itten J. The elements of color. – John Wiley & Sons, 1970. Т. 4. 96 с.

28. MacDonald M. Creating a website: the missing manual. "O'Reilly Media, Inc.", 2009.

29. Mills A. Now you see me-now you don't: Journalists' experiences with surveillance //Journalism Practice. 2019. Т. 13. №. 6. С. 690-707.

30. Node.js – Run JavaScript Everywhere: веб-сайт. URL: <https://nodejs.org/en> (дата звернення 02.05.2024 р.).

31. Raskin J. The humane interface: new directions for designing interactive systems. Addison–Wesley Professional, 2000. 233 p.

32. React: веб-сайт. URL: <https://uk.legacy.reactjs.org> (дата звернення 12.04.2024 р.).

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						55
Змн.	Арк.	№ докум.	Підпис	Дата		

33. Savka N.Ya., Krynytskyi O.V., Zhigalo Ya., Pohorilets A.V Modeling web projects in the style of minimalism // Proceeding of International Scientific Conference “Organization of scientific research in modern conditions” March, 2024. USA: Washington, 2024. C. 37-39.

34. Teodorescu I. et al. Landing Pages Features to Attract Customers // OvidiusUniversity Annals, Economic Sciences Series. 2015. T. 15. No. 2. P. 360–363.

					КР.КІ.8872531.00.00.000.ПЗ	Арк.
						56
Змн.	Арк.	№ докум.	Підпис	Дата		