

Міністерство освіти і науки України
Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА
за спеціальністю 123 «Комп'ютерна інженерія»

Тема наукової роботи: ЗАСОБИ РОЗРОБКИ ТА ОПТИМІЗАЦІЇ САЙТІВ
ДЛЯ НАДАННЯ ФРІЛАНС-ПОСЛУГ НА ОСНОВІ JAVA SCRIPT

Виконав	_____	С. Б. Анцигін
Керівник роботи	_____	О. М. Маркова
Консультант	_____	
Нормоконтроль	_____	Д. І. Кузнецов
Завідувач кафедри	_____	А. І. Купін

Кривий Ріг
2023

Криворізький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерних систем та мереж

Ступінь вищої освіти
Спеціальність

магістр
123 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ

Завідувач кафедри, голова циклової комісії

_____ А. І. Купін

“ ____ ” _____ 20__ року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

_____ (прізвище, ім'я, по батькові)

1. Тема роботи _____

керівник роботи _____,

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “ ____ ” _____ 20__ року № ____

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи _____

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) _____

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

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської роботи	Строк виконання етапів роботи	Примітка

Студент _____
(підпис) (прізвище та ініціали)Керівник роботи _____
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка: 100 сторінок, 12 рисунків, 2 таблиці, 1 додаток, 18 використаних джерел.

Об'єкт дослідження – Java script, як засіб розробки сайту для надання фріланс-послуг.

Робота складається з трьох розділів.

Перший розділ присвячено аналізу JavaScript для веб-сайтів. Зроблено огляд сучасних, існуючих підходів щодо розробки сайту. Обґрунтовано необхідність розробки веб-сайтів для різних сфер. Висвітлено питання актуальності розробки веб-сайтів на JavaScript.

Другий розділ розкриває питання вже конкретної розробки веб-сайту та архітектури. Проведено моделювання, розроблено математичну модель процесу передачі даних для веб-сайтів на JavaScript.

У третьому розділі виконано розрахунки для визначення витрат щодо створення сайту, реалізовано точні розрахунки.

Ключові слова: КОМП'ЮТЕРНА МЕРЕЖА, ВЕБ-САЙТ, ПРОГРАМУВАННЯ, РОЗРОБКА, РОЗРАХУНКИ, JAVASCRIPT.

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата	РЕФЕРАТ			
Розробив	Анцигін							
Перевірів	Купін							
Н.контроль	Кузнецов							
Затвердив	Купін				ЗКІ-22м			

Master's work: 100 pages, 12 figures, 2 tables, 1 appendix, 18 references.

The object of research is the process of optimizing data transmission routes in computer networks with a heterogeneous structure based on the IGRP protocol.

The paper consists of three chapters.

The first section is devoted to the analysis of JavaScript for websites. An overview of modern, existing approaches to website development is given. The necessity of developing websites for various fields is substantiated. The issue of the relevance of JavaScript website development is highlighted.

The second section covers the issues of specific website development and architecture. Modeling is carried out, a mathematical model of the data transfer process for JavaScript websites is developed.

In the third section, calculations are performed to determine the costs of creating a website, and accurate calculations are implemented.

COMPUTER NETWORK, WEBSITE, PROGRAMMING,
DEVELOPMENT, CALCULATIONS, JAVASCRIPT.

					KHY.PM.123.23.05.P	Арк.
	Арк.	№ документа	Підпис	Дата		

ЗМІСТ

ВСТУП	8
I. АНАЛІТИЧНА ЧАСТИНА.....	10
1.1. Характеристика сфери, для якої створюється сайт.....	10
1.2. Огляд JavaScript для розробки веб-сайтів.....	15
1.3. Аналіз потреб цільової аудиторії та вимог до сайту.....	22
II. ПРАКТИЧНА ЧАСТИНА	31
2.1. Розробка архітектури сайту та його інтерфейсу.....	31
2.2. Використання JavaScript для створення динамічного контенту	53
2.3. Адаптивність сайту для різних пристроїв	57
III. РОЗРАХУНОК ВАРТОСТІ ПРОЕКТУ	59
IV. ТЕСТУВАННЯ ТА НАЛАГОДЖЕННЯ САЙТУ	63
ВИСНОВКИ.....	65
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	66

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата	ЗМІСТ	Літера	Аркуш	Аркушів
Розробив	Анцигін							
Перевірив	Купін							
Н.контроль	Кузнєцов							
Затвердив	Купін					ЗКІ-22м		

ВСТУП

Актуальність дипломної роботи полягає в тому, що однією з унікальних переваг JavaScript є його поширеність. Цю мову можна зустріти буквально всюди. Вона підтримується на всіх операційних системах, у всіх видах браузерів і настільних комп'ютерах, і на мобільних пристроях. JavaScript є потужною мовою програмування, яка може бути використана для різноманітних цілей при створенні сайту для роботи з фрілансу. JavaScript дозволяє створювати динамічні інтерфейси, такі як анімація, спливаючі вікна, інтерактивні форми і кнопки. Можна використовувати JavaScript для створення зручних способів взаємодії із замовниками та фрілансерами на сайті. Використовуючи JavaScript можна перевіряти та проводити валідацію форм, для запобігання некоректним або порожнім полям при введенні інформації користувачами. Можна використовувати JavaScript для завантаження даних з сервера без перезавантаження сторінки, що дозволяє користувачам швидко та зручно отримувати з сайту нову інформацію. JavaScript дозволяє взаємодіяти з різними API сторонніх сервісів, такими як платіжні системи, поштові служби, геолокація та інші, що розширює можливості сайту.

Використання JavaScript для автозаповнення поля пошуку або форм дозволить зменшити зусилля користувачів та зробить процес пошуку проектів або фрілансерів більш зручним. JavaScript дозволяє змінювати зміст та структуру HTML сторінки, що допоможе забезпечити динамічну зміну контенту на сторінці у реальному часі. Є можливість використовувати JavaScript для збереження невеликих обсягів даних на пристроях клієнтів.

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата	ВСТУП			
Розробив	Анцигін							
Перевірив	Купін							
Н.контроль	Кузнецов							
Затвердив	Купін							
						Літера	Аркуш	Аркушів
						ЗКІ-22м		

Використання технологій, таких як WebSockets або Server-Sent Events, дозволяє оновлювати вміст сторінки в реальному часі без необхідності перезавантаження сторінки, що може бути корисним при швидкому перегляді нових проектів чи повідомлень. Використання JavaScript для створення анімації та інших візуальних ефектів може зробити сайт привабливішим і цікавішим для користувачів.

Об’єкт дослідження - сайт сайту для надання фріланс-послуг.

Предмет дослідження – Javascript, як засіб розробки сайту для надання фріланс-послуг.

Метою дипломної роботи є розробка та реалізація функціонального веб-сайту для роботи з фрілансу, який використовує JavaScript для забезпечення динамічних можливостей та взаємодії з користувачем.

Основні **завдання** включають:

- 1) Аналіз та дослідження потреб та очікувань цільової аудиторії фрілансерів та замовників.
- 2) Вивчення основних можливостей JavaScript та його інтеграції з HTML та CSS.
- 3) Розробка архітектури сайту та проектування його інтерфейсу для оптимальної навігації та зручного використання.
- 4) Реалізація динамічних елементів, таких як анімації, модальні вікна, зміна контенту без перезавантаження сторінки, валідація форм тощо.
- 5) Розробка взаємодії з базою даних (за потреби) для збереження інформації про користувачів, замовлення, історії операцій тощо.
- 6) Забезпечення адаптивності сайту для різних пристроїв (мобільних телефонів, планшетів, настільних комп'ютерів).
- 7) Тестування та налагодження сайту для забезпечення його стабільної та ефективної роботи.

Структура роботи. Робота складається з чотирьох частин – аналітичної, практичної, економічної та тестування. Загальна кількість сторінок - 87.

					KHUY.PM.123.23.05.P	Арк.
	Арк.	№ документа	Підпис	Дата		

I. АНАЛІТИЧНА ЧАСТИНА

1.1. Характеристика сфери, для якої створюється сайт

Сучасні зміни в українському суспільстві впливають на всі сфери життя, включаючи зайнятість населення. Проблема працевлаштування набула особливої актуальності під час поточної кризи в Україні. Безробіття стало однією з найважливіших соціально-економічних проблем, яка потребує негайної уваги з боку держави та суспільства. Для подолання цієї проблеми необхідно створювати нові робочі місця, що потребує значних витрат. Проте на фоні цього швидко розвивається таке явище як фріланс, коли робочі місця практично створюються з "нічого" (принаймні для держави), але вони дійсно користуються попитом.

Останнім часом багато досліджень зосереджуються на самостійній неформальній зайнятості, дистанційній роботі та гнучкому графіку. Це пов'язано, зокрема, із тим, що самостійна зайнятість у наш час залишається мало вивченим поняттям.

В світовій економіці зароджується новий клас працівників, які не зорієнтовані на постійне працевлаштування, а готові працювати за наймом на короткі періоди. Цих працівників називають фрілансерами, а таку форму зайнятості - фрілансом.

Фрілансер (від freelancer) – поняття, яке зародилося ще у середньовіччі. Спочатку цим терміном називали професійних військових, що працювали на різних замовників за гонорар. В романі "Айвенго" Вальтер Скотт називав їх "вільними списоносцями". Згодом, починаючи з XIX століття, термін стали застосовувати у Європі для "вільних художників" – незалежних журналістів та інших творчих професій.

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Анцигін				АНАЛІТИЧНА ЧАСТИНА	Літера	Аркуш	Аркушів
Перевірив	Купін							
Н.контроль	Кузнецов					ЗКІ-22м		
Затвердив	Купін							

На сьогоднішній день фріланс є поширеною формою самостійної зайнятості. Фрілансер - це людина, яка виконує роботу без укладання довгострокового контракту з роботодавцем, а наймають її лише для виконання конкретних замовлень.

Фріланс дозволяє керувати своїм часом і проектами, а також пропонувати свої послуги на міжнародному ринку, що збільшує заробіток. У США понад третина працівників є фрілансерами, хоча в Україні цей вид зайнятості менш популярний. Однак, зростання зацікавленості молодих спеціалістів до інформаційних технологій може призвести до збільшення кількості фрілансерів в майбутньому.

Суть фрілансу полягає у виконанні роботи на віддаленій основі, коли працівник не залежить від фізичної присутності на робочому місці організації. Цей вид самозайнятості можна також називати "дистанційною роботою". У країнах Західної Європи та США фріланс поширений і переважно відбувається поза штатом організацій.

Однак, інститут фрілансу мало вивчений у сучасній економічній науці, тому існує різне визначення цього явища. Взагалі, фрілансери є висококваліфікованими незалежними працівниками, які діють без довгострокових трудових договорів завдяки використанню інформаційно-комунікаційних технологій.

Фрілансерами, як правило, стають особи, що володіють творчими професіями, такими як письменники, журналісти, дизайнери, художники та фотографи. Також часто спеціалісти з інформаційних технологій, такі як веб-розробники, веб-дизайнери, контент-менеджери та адміністратори сайтів, обирають роботу фрілансерами. Значна кількість фрілансерів також представлена архітекторами, інженерами, маркетологами і фахівцями з реклами.

Навіть якщо у вас немає глибоких знань у сфері дизайну, web-дизайну, програмування чи маркетингу, все одно є можливість стати фрілансером.

					КНУ.РМ.123.23.05.Р	Арк.
	Арк.	№ документа	Підпис	Дата		

Наприклад, ви можете працювати перекладачем, коректором, писати студентські роботи на замовлення, займатися рерайтингом або копірайтингом.

Фріланс – це співпраця з роботодавцем на тимчасовій або постійній основі, що відбувається через інтернет. Це означає, що виконання роботи або проекту здійснюється на території виконавця, а не роботодавця.

Фрілансер має велику свободу дій, може виконувати роботу у зручний для себе час і спосіб. Це дозволяє їм брати проекти, які їх цікавлять і задовольняють.

Однією з основних переваг цього виду зайнятості є те, що фрілансери можуть обирати роботу, яка їх задовольняє і відповідає їхнім інтересам. Для багатьох фрілансерів це стає не лише способом заробітку на віддаленій роботі, але й стилем життя та мислення, дозволяючи насолоджуватися свободою дій та відсутністю жорстких регламентів. Фрілансери йдуть вільним шляхом, відступаючи від загальноприйнятої думки і стереотипів.

Широко поширено вважати, що фріланс є популярним вибором для молодого покоління. Однією з причин такого вибору є те, що молодим людям, особливо без досвіду роботи, важко знайти традиційну роботу, яка б задовольняла їх потреби. З іншого боку, молодим людям подобається фріланс через свободу і незалежність, яку ця форма праці надає. Зростання покоління в епоху комп'ютерів і гаджетів також допомагає, адже для них робота з дистанції стає звичною і цікавою. Зазначено, що більшість фрілансерів у світі молодше 35 років, а середній вік фрілансера в Україні становить 26 років.

Ті хто обирає фріланс, повинні мати такі якості, як комунікабельність, вміння вести переговори та здатність побудувати міцні ділові зв'язки. Для успіху в цій сфері, фрілансер має добре рекламувати себе та свої здібності, забезпечивши повторні замовлення від клієнтів.

Серед переваг роботи фрілансера можна виділити:

- Незалежність та свобода вибору робочих завдань.
- Гнучкий графік роботи.

- Можливість працювати вдома, що особливо актуально для молодих батьків та маломобільних людей.
- Комфортні умови праці та зниження витрат на оренду офісу та особливо транспорт.
- Виконання лише власної роботи і можливість обирати цікаві проекти.
- Можливість збалансувати роботу і особисте життя.
- Шанс співпрацювати з клієнтами з інших країн і культур.
- Самостійний вибір партнерів і колег.

Роботодавці також отримують переваги в співпраці з фрілансерами:

- Оплата лише за результат виконаної роботи, а не за години присутності на робочому місці.
- Можливість залучати висококваліфікованих фахівців з різних регіонів.
- Зниження витрат на дорогоцінні робочі місця в офісі.
- Відсутність соціальних витрат і відповідальності за відпустки та лікарняні.
- Зменшення документообігу і звітності.
- Простота припинення співпраці з фрілансером без необхідності прокладати складні юридичні шляхи звільнення штатного співробітника.

І хоча фріланс має свої плюси і мінуси, ця форма праці стає все більш популярною, особливо серед молодого покоління, яке цінує свободу і гнучкість, яку надає фріланс.

Проблемами (недоліками) у роботі фрілансера можна виділити такі аспекти:

- 1) Нестабільність доходів, оскільки немає гарантії постійного замовлення.
- 2) Ризик неотримання оплати за виконану роботу.
- 3) Відсутність соціального пакету та пільг, які зазвичай надаються в офіційних місцях роботи.

					КНУ.РМ.123.23.05.Р	Арк.
	Арк.	№ документа	Підпис	Дата		

- 4) Обов'язок самостійно вести бухгалтерію та сплачувати податки.
- 5) Не завжди легко знаходити підходящі пропозиції проектів.
- 6) Потреба у додаткових витратах на обладнання та робоче місце.
- 7) Можливі проблеми з отриманням кредиту для особистих потреб.
- 8) Обмежені можливості для кар'єрного росту порівняно з традиційними робочими місцями.

Що стосується України, то тенденція до фрілансу тут розвивається зі затримкою порівняно з іншими країнами. Фахівці довго вагалися перейти на такий спосіб праці, оскільки посткризовий ринок не демонстрував значного зростання. Однак, нині все більше українців обирають фріланс, побачивши в ньому можливість отримати вільний заробіток. Прогнозується, що кількість фрілансерів з України на світовому ринку зростатиме щорічно від 10 до 25%. Проте, деякі люди бояться ризиків і небезпек, пов'язаних з фрілансом, і залишаються на традиційних робочих місцях, хоча вже давно мріють працювати на себе.

Кожен рік у країні отримує дипломи багато молодих фахівців, проте не всі з них можуть знайти робочі місця. У той же час фріланс приймає всіх, незалежно від віку або досвіду роботи, головне - якісно та вчасно виконувати завдання. Тому кількість фрілансерів в Україні постійно зростає.

Сьогодні дистанційна робота стає популярною в Україні, існує ініціатива з легалізації такого типу зайнятості, але деякі негативні моменти, зокрема, нелегальна праця, стають проблемою для дистанційних працівників, яким не надаються соціальні гарантії. Такою ініціативою є законопроект від Міністерства соціальної політики спрямований на забезпечення соціального захисту для дистанційних працівників і регулювання питань оплати праці.

Фріланс має велике значення для розвитку України, оскільки фрілансери творять нові товари та надають важливі послуги з використанням інформаційно-комунікаційних технологій, що сприяє зростанню ВВП, зниженню безробіття та збільшенню доходів бюджету.

Фріланс - це нова модель працевлаштування, що випереджає класичну найману працю в офісі. Чим більше молодих спеціалістів обирають фріланс, тим більше вони стають конкурентоспроможними.

Варто пам'ятати, що фріланс - це свобода, але також вимагає постійних зусиль і здібності конкурувати. Компромісним варіантом може бути поєднання фрілансу з постійною офіційною роботою для отримання переваг обох форм зайнятості. Важливо зберігати "запасний парашут" і зважено розглядати всі переваги та недоліки такої форми зайнятості.

1.2. Огляд JavaScript для розробки веб-сайтів

JavaScript - найпоширеніша мова сценаріїв для створення скриптів у веб-браузерах та є вбудованою в веб-сторінки. JavaScript був початково названий "Живим скриптом", але через популярність мови Java, його перейменували на JavaScript, щоб сприяти асоціації з Java. Проте, JavaScript розвинувся в окрему мову зі своїм стандартом ECMAScript і втратив зв'язок з Java.

Зазвичай JavaScript використовується як мова сценаріїв клієнтської сторони, і код JavaScript розміщується безпосередньо на сторінках HTML. Це дозволяє будь-кому, хто відвідує сторінку, переглядати і копіювати сценарії. Однак ця відкритість має свої переваги, оскільки дозволяє вивчати та використовувати JavaScript-код, знайдений на веб-сторінках.

Крім веб-браузерів, JavaScript також може використовуватися на серверних платформах, наприклад, у вигляді мови CGI на стороні сервера Netscape. Таким чином, він може виконувати різноманітні завдання, такі як Perl або ASP. Проте на цьому сайті акцентується саме на використанні JavaScript у веб-браузерах.

Сьогодні JavaScript може використовуватись не лише в браузері, але й на сервері чи будь-якому іншому пристрої завдяки спеціальній програмі, відомій як JavaScript Mechanical. У браузерах JavaScript працює за допомогою вбудованого "движка", який часто називають віртуальною машиною

JavaScript. Різні браузери використовують різні двигуни, такі як V8 у Chrome і Opera або SpiderMonkey у Firefox.

Сучасний JavaScript є безпечною мовою програмування, що означає, що він не надає низькорівневий доступ до пам'яті або ЦП, оскільки призначений для використання у браузерах, де це не потрібно.

Можливості JavaScript значною мірою залежать від середовища, у якому він застосовується. Наприклад, в середовищі Node.js JavaScript підтримує функції, які дозволяють читати/записувати випадкові файли, надсилати мережеві запити тощо.

У браузері JavaScript може виконувати різноманітні дії, пов'язані з шахрайством на веб-сайті, взаємодією з користувачем та веб-сервером. Наприклад, JavaScript може:

- 1) Додавати новий HTML на сторінку, змінювати існуючий вміст та стилі.
- 2) Реагувати на дії користувача, такі як натискання миші, рух вказівника та натискання клавіш.
- 3) Надсилати запити по мережі на віддалені сервери, зкачувати та завантажувати файли за допомогою технологій AJAX та COMET.
- 4) Отримувати та встановлювати файли cookie, задавати питання відвідувачеві та показувати повідомлення.
- 5) Запам'ятовувати дані на стороні клієнта за допомогою "локального зберігання".

Angular - це одне з найпотужніших середовищ JavaScript, використовуване Google для розробки односторінкових програм (SPA). Це відоме середовище розробки, яке дозволяє розробникам найкращим чином поєднати JavaScript з HTML та CSS. Понад півмільйона сайтів, включаючи google.com, youtube.com і багато інших, використовують Angular. Це також основне інтерфейсне середовище для розробників додатків Google.

Angular має компонентну структуру, аналогічну React, що дозволяє маніпулювати, вкладати та використовувати компоненти за потреби. Для

написання програм в Angular потрібно використовувати TypeScript - розширений набір JavaScript, який підтримує статичну типізацію та класи. TypeScript надає модифікатори доступу, перерахування, узагальнення, гібридні типи та багато інших корисних можливостей.

Angular має кілька переваг:

- Підтримка створення прогресивних програм (PWA);
- Зручність маніпулювання DOM-елементами;
- Висока швидкість та продуктивність;
- Доступна крива навчання;
- Вбудований механізм застосування залежностей;
- Підтримка Google та потужна екосистема.

Таким чином, Angular є фантастичною платформою, яку варто розглядати, особливо для нових розробників.

Наразі, лідером у сфері інфраструктури для JavaScript UI є React. Цей інструмент спочатку був розроблений командою Facebook з метою спростити їх роботу над програмою Facebook Ads, яка швидко зростала, що призводило до складнощів в управлінні і підтримці. Задля ефективності, команда створила структуру, яка розвивалась від раннього прототипу у 2011 році до відкритого вихідного коду доступного для громадськості через два роки. Нині React широко використовують великі компанії, такі як AirBNB, PayPal, Netflix і інші.

Основним принципом React є використання компонентів багаторазового використання. Компоненти представляють собою блоки коду, які можуть бути класифіковані як класи або функції, і кожен з них представляє певну частину сторінки, наприклад, логотип, кнопку або поле введення. Використані параметри називаються реквізитами або властивостями. Важливо зазначити, що вивчення React стає простішим, якщо у вас вже є досвід роботи з JavaScript.

React також використовує JSX, який представляє собою синтаксис XML, який поєднує JavaScript і HTML. Це не чистий шаблон JavaScript, а повний JavaScript. Новим розробникам може спочатку здатися, що JSX дещо

заплутаний, але з практикою вони виявлять його корисність. React - це бібліотека, на яку слід звернути увагу при розробці веб-інтерфейсів.

Серед переваг React можна виділити:

- 1) Можливість створювати прогресивні програми (PWA).
- 2) Зручність маніпулювання DOM-елементами.
- 3) Висока швидкість та продуктивність.
- 4) Зручна крива навчання для новачків.
- 5) Вбудований механізм застосування залежностей.
- 6) Підтримка зі сторони Google та потужна екосистема.

Aurelia, створена Робом Айзенбергом, є відкритим інтерфейсним середовищем з відкритим кодом. Перша версія, Aurelia 1.0, була випущена у 2016 році. Це інструмент, що складається з функціонально-орієнтованих модулів, таких як плагіни, маршрутизація, тестування, впровадження залежностей та інші.

За допомогою Aurelia JS можна створювати веб-додатки та програми для мобільних пристроїв та комп'ютерів. Його модульність дозволяє створювати програми різних розмірів. Aurelia підтримує як Babel, так і TypeScript, а також повністю сумісний з майбутніми версіями JavaScript.

Одна з переваг Aurelia полягає у тому, що вона використовує компонентну модель, яка відповідає найкращим веб-стандартам і використовує нові функції ECMAScript для написання коду. Це дозволяє уникнути вивчення власного синтаксису, що працює лише з Aurelia.

Основні переваги Aurelia включають високу продуктивність, наявність великої спільноти користувачів, адаптивну прив'язку даних та ефективність використання пам'яті.

Vue - це відкрито-вихіднокодовий JavaScript-фреймворк, призначений для створення інноваційних інтерфейсів. Він легко інтегрується з проектами, які використовують інші JavaScript-бібліотеки, завдяки його гнучкості та адаптивності. Наразі більш ніж 36 000 веб-сайтів використовують Vue. Відомі

компанії, такі як Stackoverflow та PlayStation, довіряють Vue для створення своїх користувацьких інтерфейсів.

Vue.js має простий вивчення, оскільки вимагає лише знань JavaScript та HTML. Його командний рядок (CLI) є ще однією сильною стороною, що спрощує розробку за допомогою широкого спектру плагінів, пресетів, миттєвого прототипування та інтерактивних інструментів проектування.

Vue.js надає такі функції, як компоненти, шаблони, переходи та двостороннє зв'язування даних, а також відзначається сильним акцентом на реактивність. Зміни або оновлення будь-якого об'єкта JavaScript у Vue автоматично відображаються у веб-інтерфейсі завдяки механізму реактивності. Vue.js також використовує технологію Shadow DOM, що забезпечує швидке відтворення сторінок.

Серед основних переваг Vue можна виділити:

- 1) Високий рівень налаштованості;
- 2) Легке освоєння;
- 3) Підтримка CSS переходів та анімацій;
- 4) Гнучкість та модульність.

jQuery, імовірно, є найбільш популярною бібліотекою JavaScript з багатьма корисними функціями для сучасної веб-розробки. Вона була створена Джоном Резігом у 2006 році та зарекомендувала себе як швидка та ефективна бібліотека JavaScript для спрощення HTML-скриптіну на стороні клієнта. На сьогоднішній день більш ніж 19 мільйонів веб-сайтів використовують jQuery! Вона є популярним вибором для компаній, таких як WordPress, Facebook, Google, IBM та багатьох інших, які сподіваються на неї для покращення веб-дизайну своїх сторінок.

API jQuery дозволяє легко обробляти, анімувати та змінювати події в HTML-документах (DOM). Крім того, jQuery успішно використовується зі збудованими інструментами Angular та React App. Одним словом, вона є однією з найважливіших бібліотек JavaScript для веб-розробки.

Ember.js - це відкритий JavaScript-фреймворк, створений Єгудою Кацем у 2011 році. Початково проект називався SproutCore 2.0, перш ніж отримати назву Ember.js. Робота над ним розпочалася у 2011 році, а версія 1.0 була випущена через два роки. Ember добре масштабується і придатний для великих проектів. Прикладом використання фреймворку є Apple Music, з понад 60 мільйонами передплатників у всьому світі, що була побудована з використанням Ember.js. Для огляду та аналізу проектів Ember в Інтернеті існує інструмент Ember Inspector.

Ember має потужний інструмент збірки під назвою Ember CLI, який позичає вдалих рішень з інших середовищ SPA. Цей інструмент забезпечує все необхідне для швидкого початку роботи. Наприклад, він надає вбудований роутер, вбудоване тестування та можливість роботи з внутрішніми даними. Ember.js використовує багато з тих самих принципів, що й Ruby on Rails, і це робить його дуже продуманим та гнучким фреймворком.

Переваги Ember.js:

- 1) Висока продуктивність.
- 2) Адаптивність та гнучкість.
- 3) Дозволяє писати розширені HTML-теги.
- 4) Можна виконувати тести прямо у браузері.
- 5) Швидко зростаюче співтовариство.

Наприкінці 2011 року було випущено веб-фреймворк під назвою Skybreak, який згодом був перейменований на Meteor. Завдяки простому JavaScript, ви можете створювати програми, які працюватимуть на різних платформах, включаючи Android та iOS. Meteor JS забезпечує повний стек рішень, охоплюючи важливі аспекти, такі як внутрішня розробка, управління базами даних, бізнес-логіка та зовнішній рендеринг.

Meteor вважає себе найшвидшим способом створення програм на JavaScript і здобуває популярність на ринку з більш ніж 13 000 веб-сайтів, які використовують цей фреймворк. Серед таких сайтів є mtv.com,

meteofrance.com і інші, які використовують Meteor для створення своїх інтерфейсів користувача.

До переваг Meteor відносяться:

- Екосистема ізоморфного розвитку (IDevE).
- Вбудоване перезавантаження браузера.
- Користувальницький менеджер пакетів.
- Потужна хмарна платформа для розгортання, масштабування та моніторингу клієнтських програм.
- Реактивні візерунки.

JavaScript є однією з найпопулярніших мов програмування, використовуваних для розробки веб-сайтів. Вона дозволяє створювати інтерактивні та динамічні веб-сторінки, що реагують на дії користувачів без необхідності перезавантаження сторінки. Нижче перелічені основні характеристики JavaScript та те, що варто знати про неї при розробці веб-сайтів:

- Синтаксис: JavaScript має синтаксис, подібний до інших мов програмування, таких як C, C++ і Java. Це легко вивчити, особливо для тих, хто вже знайомий з іншими мовами програмування.

- Клієнтський і серверний JavaScript: JavaScript може використовуватись як на стороні клієнта, так і на стороні сервера. На стороні клієнта, JavaScript використовується для управління поведінкою сторінки в браузері користувача. На стороні сервера (за допомогою платформи Node.js), він використовується для створення серверних додатків.

- DOM (Document Object Model): DOM - це інтерфейс, який дозволяє JavaScript змінювати контент, структуру і стилі HTML-документів. Він дозволяє робити взаємодію з елементами сторінки, створювати, видаляти та змінювати їх зміст.

— Асинхронність: JavaScript підтримує асинхронну обробку подій, що дозволяє виконувати довгочасні операції, такі як завантаження даних, без заблокування основного потоку виконання коду.

— Бібліотеки та фреймворки: Існує багато бібліотек і фреймворків JavaScript, таких як React, Angular, Vue.js, jQuery, які допомагають спростити розробку та забезпечують високий рівень перевикористання коду.

— AJAX (Asynchronous JavaScript and XML): AJAX дозволяє здійснювати асинхронний обмін даними між веб-сервером і браузером без необхідності перезавантаження сторінки. Це дозволяє створювати більш інтерактивні користувацькі інтерфейси.

— Робота з формами: JavaScript дозволяє перевіряти, валідувати та обробляти дані з форм на сторінках, що забезпечує більшу контроль над даними, які передаються на сервер.

— Локальне сховище: JavaScript може звертатись до локального сховища (Local Storage) браузера, що дозволяє зберігати дані на боці клієнта для подальшого використання без необхідності звертатися на сервер.

— Робота з API: JavaScript дозволяє здійснювати запити до зовнішніх API, що дозволяє отримувати дані з інших джерел і використовувати їх на веб-сайті.

Щоб стати успішним розробником веб-сайтів, важливо глибоко освоїти JavaScript та вивчити основи HTML та CSS, які використовуються для створення структури та оформлення веб-сторінок. Крім того, корисно вивчити популярні бібліотеки та фреймворки, такі як React, Angular або Vue.js, які забезпечують ефективнішу розробку та реактивні інтерфейси.

1.3. Аналіз потреб цільової аудиторії та вимог до сайту

Проведемо аналіз існуючих сайтів з фрілансу.

Fiverr є однією з найвідоміших бірж фрілансерів, що об'єднує тисячі фрілансерів у більш ніж 200 категоріях. Однак те, що робить його особливим серед конкурентів, - це доступні ціни, які починаються від всього 5 доларів за виконання певної роботи. Якщо ви бажаєте вищої якості виконання проекту і

					KNU.PM.123.23.05.P	Арк.
	Арк.	№ документа	Підпис	Дата		

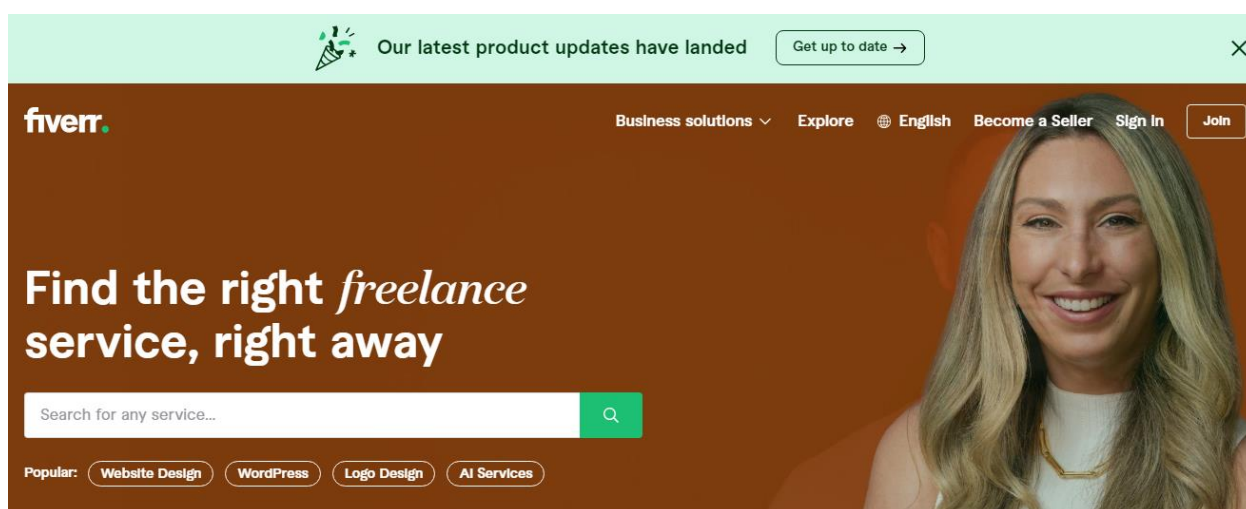
готові витратити більше коштів, ви також зможете знайти досвідчених фрілансерів з чудовими відгуками.

Один із важливих аспектів Fiverr - це безпечність платежів. Ви можете спокійно сплачувати за проекти в межах роботи, оскільки загальна ціна визначається при наймі фрілансера. Якщо виконавець не реагує протягом 24 годин після початку проекту, ви зможете скасувати проект без додаткових витрат.

Ще один перевагою Fiverr - це легка інтеграція системи відгуків і рейтингів фрілансерів. Ця система враховує час виконання роботи, задоволеність клієнтів і якість обслуговування, допомагаючи легко знаходити приховані таланти.

Сервіс Fiverr Pro є ще однією з сильних переваг платформи, дозволяючи вам звужити пошук до найкращих кандидатів у своїй галузі. За це слід доплатити, але результат вартий витрачених коштів, оскільки кожен фрілансер, який отримав позначку Pro Verified, проходить ретельну перевірку їхньої освіти, досвіду, відгуків клієнтів та портфоліо.

Серед найпоширеніших категорій фрілансерів на Fiverr можна виділити такі: графічний дизайн та дизайн логотипів, веб-розробка, написання статей, озвучування записів та робота з соціальними мережами.



Рисисунок 1.1 - Головна сторінка Fiverr

Upwork - це універсальний ринок фрілансерських послуг, який має мільйони активних профілів. Тут ви легко зможете знайти фрілансера для довготривалого співробітництва, опублікувавши своє оголошення про роботу або переглянувши тисячі пропозицій від інших фрілансерів.

Відбір, співбесіди та найм фрілансерів ще ніколи не були такими простими. Зареєструвавшись на платформі, ви можете опублікувати свою пропозицію роботи менше ніж за 30 хвилин. Якщо вам подобається ретельно ознайомлюватися з профілями кандидатів, вам сподобається велика кількість доступних фільтрів, які допоможуть звужити ваш пошук. Особливо мені подобається можливість фільтрувати результати на основі успішності роботи фрілансерів.

На Upwork мільйони фрілансерів пропонують свої послуги, і завдяки цій платформі ви зможете легко зв'язатися з будь-яким з них. Просто натиснувши кнопку, ви можете розпочати спілкування в чатах, планувати відеозустрічі, обмінюватися файлами і навіть контролювати процес виконання роботи, стежачи за робочими щоденниками виконавців.

Система захисту платежів на Upwork суттєво підвищує безпеку користувачів. Оскільки оплата здійснюється тільки після затвердження клієнтом результату роботи, фрілансери мають зацікавленість виконувати завдання згідно з обговореними умовами.

Серед найпопулярніших категорій фрілансерів на Upwork можна виділити: копірайтинг, графічний дизайн, розробка програмного забезпечення, ІТ-підтримка, аналіз даних та методи дослідження.

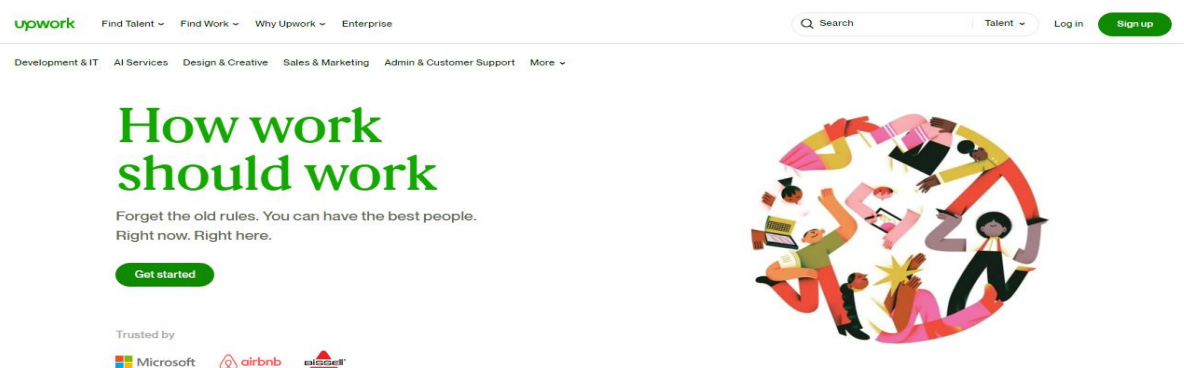


Рисунок 1.2 - Головна сторінка Upwork

					KHUY.PM.123.23.05.P	Арк.
	Арк.	№ документа	Підпис	Дата		

Toptal позиціонує себе як глобальна мережа, що збирає разом таланти світового рівня. Щороку вони обирають лише 3% з приблизно 100 000 заявок фрілансерів, які бажають зареєструватися на платформі Toptal. Незважаючи на те, що Toptal має лише п'ять основних категорій фрілансерів - дизайнери, розробники, фінансові експерти, менеджери продуктів і менеджери проектів, кожна з них детально поділена на багато дрібніших сегментів. Навіть у категорії розробників, ви зможете знайти понад 200 різних позицій.

Toptal має суворий процес відбору. Кандидат повинен пройти п'ять етапів, що може зайняти до п'яти тижнів. Якщо фрілансер не проходить перший раунд, йому буде відмовлено в співпраці з Toptal. Очевидно, тут працюють лише з найкращими професіоналами.

Кожного разу, коли ви обираєте нового фрілансера, Toptal дає вам пробний період до двох тижнів, щоб ви могли оцінити, чи відповідає він вашим вимогам. Якщо ви не повністю задоволені результатами його роботи, вам не буде виставлено рахунок за цей час, і ви зможете перейти до співпраці з іншим фрілансером.

Найпопулярніші категорії фрілансерів на Toptal: розробники програмного забезпечення, дизайнери, фінансові експерти, менеджери продуктів і менеджери проектів.

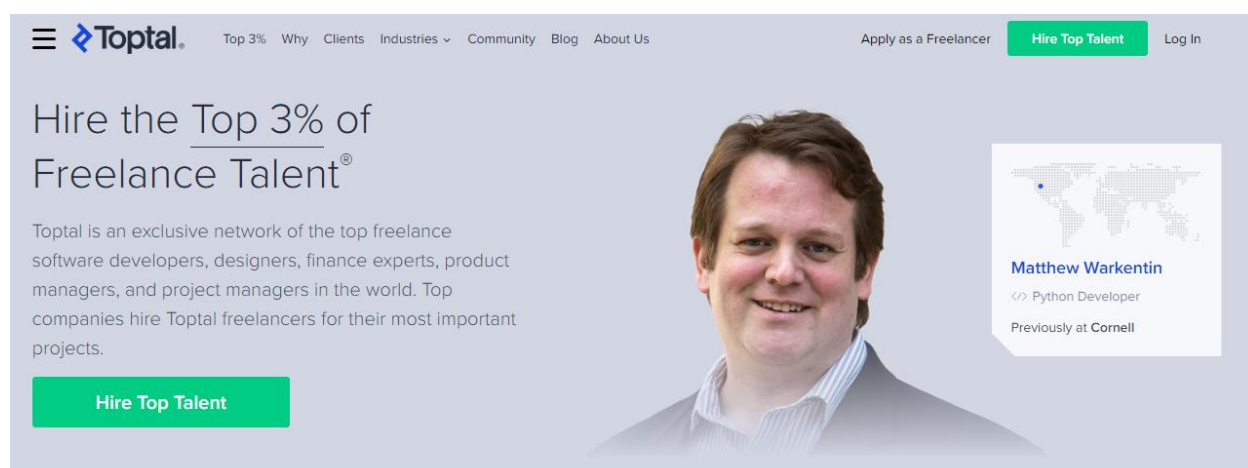


Рисунок 1.3 - Головна сторінка Toptal

Біржа DesignCrowd має понад 750 000 дизайнерів, тому ви зможете знайти того, хто працює в стилі, який вам потрібен. Перед тим як почати, вам

					KHUY.PM.123.23.05.P	Арк.
	Арк.	№ документа	Підпис	Дата		

потрібно обрати тариф, враховуючи кількість варіантів дизайну, які ви хочете отримати, і потім розмістити технічне завдання на 3-10 днів. Завдяки доступним цінам, DesignCrowd є прекрасним вибором для невеликих компаній, які шукають оригінальних дизайнерських рішень. Біржа пропонує широкий спектр категорій дизайну, від візитівок до веб-дизайну.

Процес найму на DesignCrowd схожий на 99designs, але він дешевший і менш ретельний. Ваше технічне завдання складатиметься лише з однієї сторінки, і ви не зможете переглянути відгуки про конкретних дизайнерів або їхній успіх. Проте для тих, хто має обмежений бюджет, ця біржа може стати досить гідним варіантом.

Мені сподобалося, що ви можете співпрацювати з дизайнером безпосередньо або провести конкурс, щоб оцінити дизайни, запропоновані кандидатами. Після вибору підходящого дизайну, ви можете залишити свій відгук і зробити необмежену кількість безкоштовних правок до свого проєкту.

Найпоширеніші категорії фрілансерів на DesignCrowd: проєкти графічного дизайну, такі як розробка логотипів, веб-дизайну та проєкти для друку.

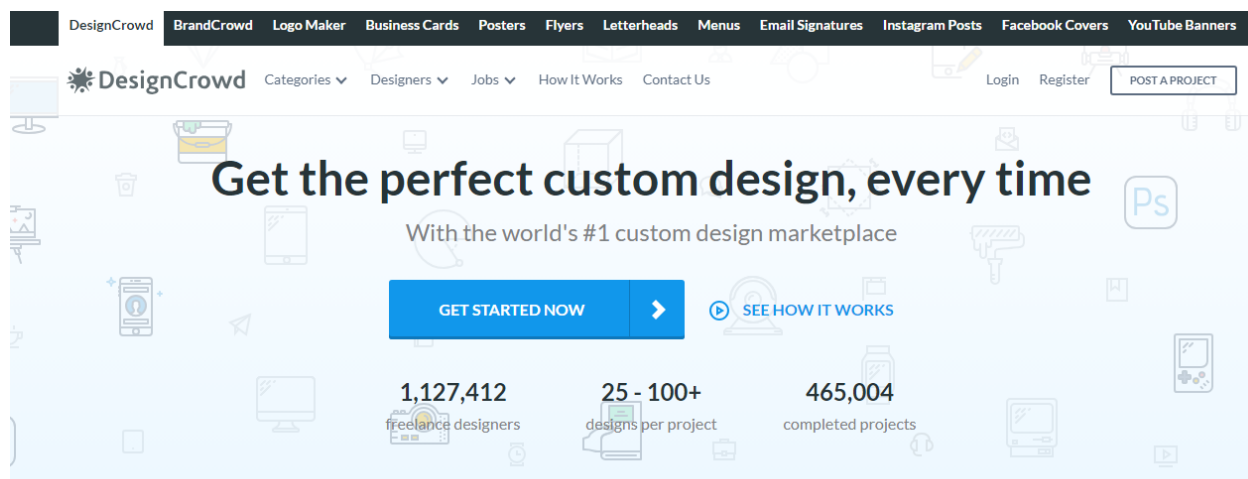


Рисунок 1.4 - Головна сторінка DesignCrowd

99Designs, безумовно, не є найдешевшою біржею (але й не найдорожчою), але вона гарантує роботу з кваліфікованими дизайнерами. Під час пошуку відповідного кандидата особливо корисною буде можливість

провести конкурс серед потенційних претендентів, оскільки це дозволить отримати кілька варіантів на вибір.

Платформа 99designs дуже проста у користуванні, а дизайнери, з якими я працювала там, залишили на мене дуже позитивне враження. Я легко організувала конкурс для свого завдання. Також мені сподобалося, що спочатку мене попросили відповісти на декілька питань, щоб краще зрозуміти, який саме стиль я переважаю. Це допомагає дизайнерам краще зрозуміти, який логотип вам потрібен. Це дійсно корисний підхід для тих, хто не дуже розбирається в дизайні!

Ви оплачуєте проект наперед, і як тільки ви затверджуєте отриманий дизайн, гроші автоматично перераховуються виконавцю платформою. Ви отримуєте всі необхідні права на ваш дизайн та проектні файли, як для друку, так і в цифровому форматі. Коли в процесі проведення конкурсу у мене виникли невеликі ускладнення, служба підтримки клієнтів дуже швидко відреагувала та допомогла вирішити їх в найкоротші терміни.

Найпопулярніші категорії фрілансерів: Розробка дизайну - від логотипів до одягу.

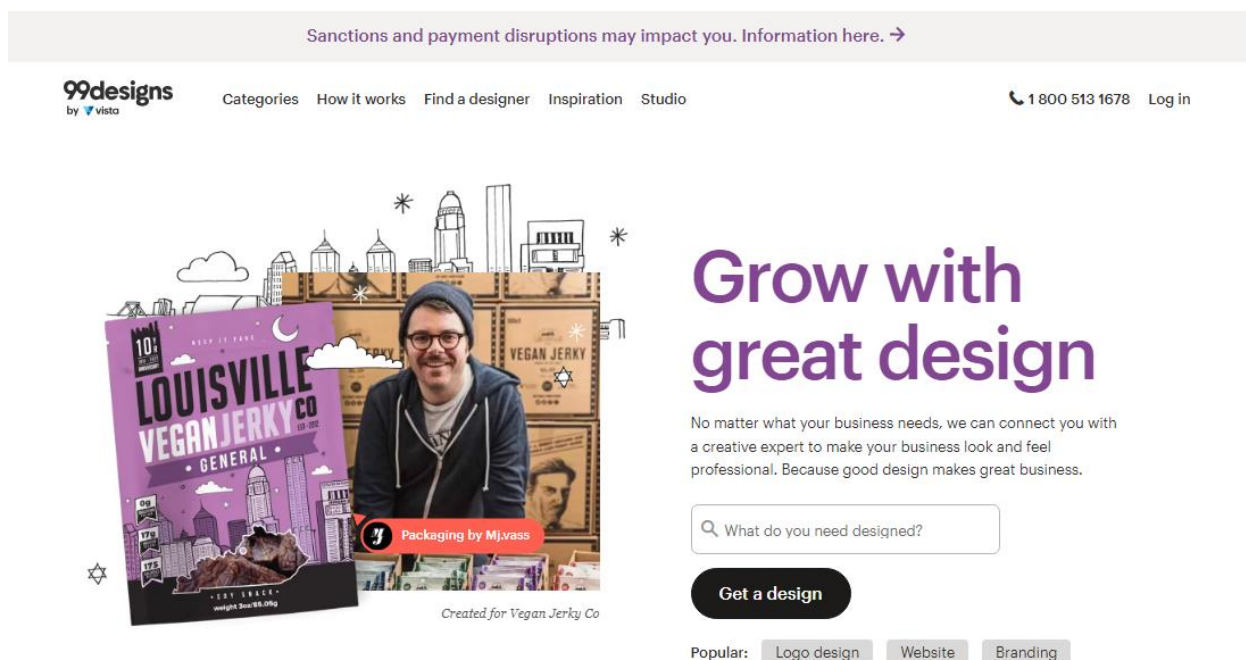


Рисунок 1.5 - Головна сторінка 99designs

					KHUY.PM.123.23.05.P	Арк.
	Арк.	№ документа	Підпис	Дата		

Freelancer.com став домівкою для великої кількості фрілансерів з більш ніж 250 країн. Хоча їх спеціалізація не така широка, як на Upwork та Fiverr, ця платформа, на моїй думці, має більш інтуїтивний та зручний у використанні інтерфейс і навігацію. Окрім того, вона підтримує багато різних мов, включаючи українську, що є чудовим.

Вартість послуг на Freelancer.com може бути вищою, порівняно з Fiverr та Upwork, через додаткові функції, за які потрібно платити. Наприклад, виділення проєкту, перехід у приватний режим, залучення спеціалістів для найму та укладання угоди про конфіденційність (NDA) — все це передбачає додаткові витрати, які можуть швидко накопичуватися. Однак, якщо вам не потрібні ці додаткові функції, вам це не створить проблем.

Мені також сподобалося, що Freelancer.com дуже простий у використанні порівняно з іншими сайтами. Створення нового проєкту на деяких інших сайтах може стати довгим та виснажливим процесом, але Freelancer.com дозволяє зробити це без зайвих зусиль. Вам потрібно лише вибрати проєкт, який найкраще відповідає вашим вимогам, і використати його як шаблон для створення свого проєкту. Це не лише зекономить ваш час, але й пропонує умови та навички, про які ви можливо не замислювалися раніше. І найкраще, за публікацію вашого проєкту не стягується жодної комісії.

Найпопулярніші категорії фрілансерів на Freelancer.com: Графічний дизайн, маркетинг, написання статей та дизайн застосунків.

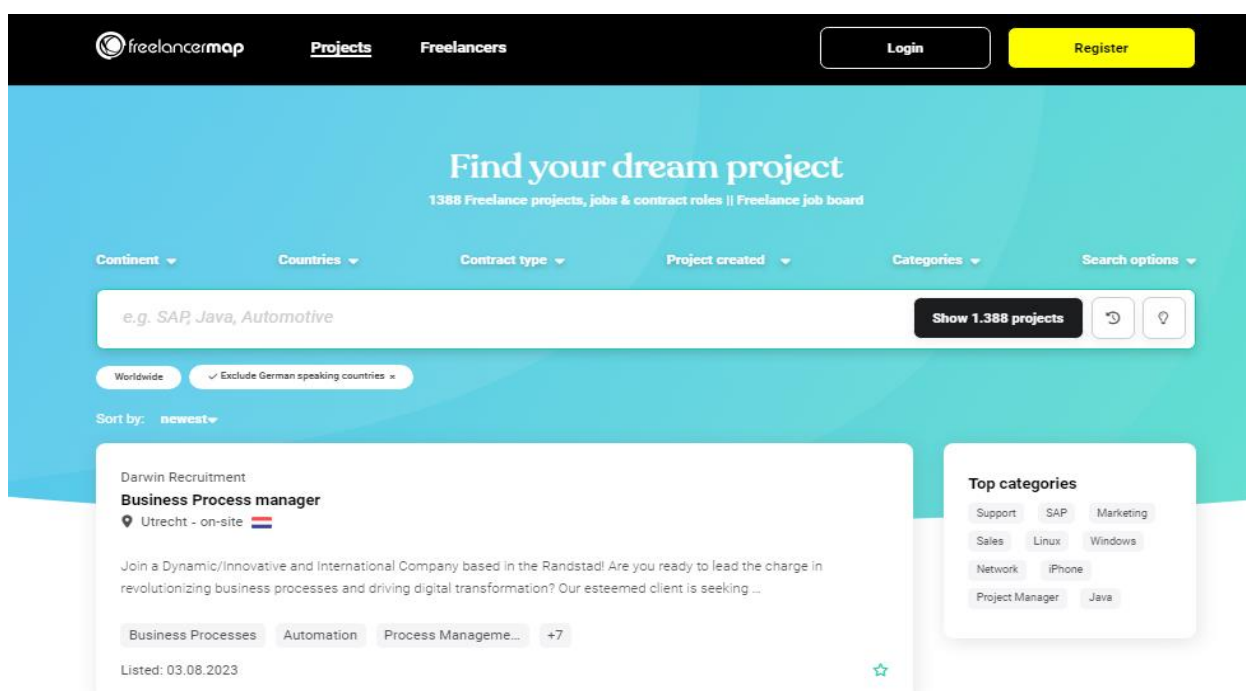


Рисунок 1.6. - Головна сторінка Freelancer.com

Отже, на основі проведеного аналізу існуючих сайтів з фрілансу створимо портрет цільової аудиторії та вимоги до сайту з фрілансу.

Цільова аудиторія та вимоги до сайту з фрілансу можуть значно варіюватися залежно від конкретної ніші та типу фрілансу. Однак, я надам загальні приклади цільових аудиторій та вимог до сайту для платформи фрілансу.

Цільова аудиторія:

1. Фрілансери: Люди, які пропонують свої послуги в різних галузях, таких як письменництво, програмування, дизайн, маркетинг, переклад, графічний дизайн та інше.
2. Замовники: Компанії та особи, які шукають фрілансерів для виконання певних проектів або завдань.

Вимоги до сайту:

1. Реєстрація та профілі: Користувачам повинно бути зручно реєструватися як фрілансери чи замовники. Профілі мають містити інформацію про навички, досвід, портфоліо та рейтинг користувачів.

2. Пошук та фільтрація: Ефективний механізм пошуку фрілансерів та проектів за ключовими словами, навичками, локацією, бюджетом тощо.

3. Повідомлення та комунікація: Вбудована система обміну повідомленнями, щоб фрілансери та замовники могли зв'язуватися та обговорювати деталі проектів.

4. Оголошення проектів: Можливість для замовників публікувати деталі проектів та запрошувати фрілансерів зробити пропозиції.

5. Управління платежами: Безпечний і зручний механізм оплати за виконані проекти, який включає систему гарантійних депозитів або ескроу.

6. Відгуки та рейтинги: Функція, що дозволяє користувачам залишати відгуки та оцінки один одного після виконання проекту.

7. Мобільна додаток: Зручна мобільна версія або додаток для доступу до платформи з різних пристроїв.

8. Безпека та захист даних: Забезпечення високого рівня безпеки для уникнення шахрайства та захисту конфіденційної інформації.

9. Підтримка користувачів: Механізм надання підтримки користувачам, які мають питання або зіштовхнулися з проблемами.

II. ПРАКТИЧНА ЧАСТИНА

2.1. Розробка архітектури сайту та його інтерфейсу

Архітектура сайту "FreeLife" включає наступні елементи:

1) Заголовок (Head):

- Основні мета-теги, включаючи кодування (UTF-8), сумісність з ІЕ, опис, ключові слова, автора та в'юпорт.

Основні мета-теги, які використовуються у коді HTML цього сайту, включають:

Кодування (Charset UTF-8):

```
<meta charset="UTF-8">
```

Визначає, що кодування символів для сторінки є UTF-8, яке підтримує більшість символів та емодзі, використовуваних у всіх сучасних мовах.

Сумісність з Internet Explorer (IE):

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
```

Забезпечує, що сайт коректно відображатиметься в браузері Internet Explorer, використовуючи найсучасніші доступні функції браузера.

Опис (Description):

```
<meta name="description" content="...">
```

Надає короткий опис сторінки, який часто використовується пошуковими системами для відображення у видачі результатів пошуку.

Ключові Слова (Keywords):

```
<meta name="keywords" content="...">
```

Перелічує ключові слова, пов'язані з вмістом сторінки. Цей тег раніше мав велике значення для SEO, але наразі його значення зменшилось через зловживання та спам.

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Арк.	№ документа	Підпис	Дата	ПРАКТИЧНА ЧАСТИНА	Літера	Аркуш	Аркушів
Перевірив	Арк.	№ документа	Підпис	Дата				
Н.контроль	Арк.	№ документа	Підпис	Дата				
Затвердив	Арк.	№ документа	Підпис	Дата		ЗКІ-22м		

Автор (Author):

```
<meta name="author" content="...">
```

Вказує ім'я автора або власника вмісту сторінки.

В'юпорт (Viewport):

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Контролює, як веб-сторінка відображається на мобільних пристроях, забезпечуючи правильне масштабування та розмір вмісту в залежності від розміру екрану.

Ці мета-теги є важливими для забезпечення правильної індексації та представлення сайту в пошукових системах, а також для забезпечення сумісності з різними браузерами та пристроями.

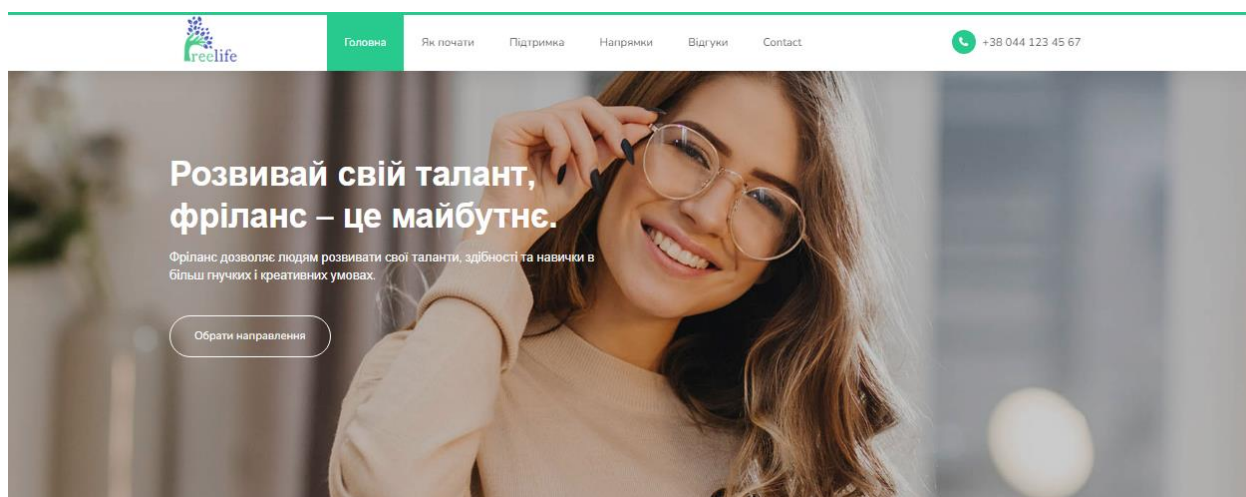


Рисунок 2.1 - Заголовок (Head)

- Підключені стилі CSS, включаючи Bootstrap, Font Awesome, Owl Carousel і власні стилі сайту.

У HTML-кодi сайту "FreeLife" використовуються наступні CSS-стилі:

Bootstrap:

```
<link rel="stylesheet" href="css/bootstrap.min.css">
```

Bootstrap — це популярний фреймворк для швидкого розроблення відзивчивих та мобільно-адаптивних веб-сайтів. Він включає систему сітки, компоненти для навігації, форм, кнопок, модальних вікон та багато іншого.

Font Awesome:

```
<link rel="stylesheet" href="css/font-awesome.min.css">
```

Font Awesome надає велику колекцію іконок, які виконані у форматі шрифту. Це дозволяє легко вставляти різні іконки на веб-сторінки без необхідності використання зображень.

Owl Carousel:

```
<link rel="stylesheet" href="css/owl.carousel.css">
```

Owl Carousel — це плагін jQuery, який дозволяє створювати адаптивні каруселі або слайдери. Використовується для демонстрації кількох елементів у слайд-шоу на веб-сторінці.

Власні стилі сайту:

```
<link rel="stylesheet" href="css/style.css">
```

Власні стилі, розроблені спеціально для цього сайту. Ці стилі можуть включати унікальний дизайн, кольорні схеми, розміри та інші елементи, які не входять до стандартних бібліотек, таких як Bootstrap чи Font Awesome.

Ці стилі разом формують зовнішній вигляд сайту, забезпечуючи його привабливість, функціональність та адаптивність до різних пристроїв та розмірів екрану. Bootstrap спрощує розробку макету та компонентів інтерфейсу, Font Awesome додає візуально привабливі іконки, а Owl Carousel використовується для створення інтерактивних слайдерів. Власні стилі додають унікальний шарм та індивідуальність дизайну сайту.

2) Тіло (Body):

- Прелоадер: Анімований екран завантаження.

Прелоадер на сайті "FreeLife" — це анімований екран завантаження, який з'являється, поки весь контент сайту завантажується. Ось детальніше про його реалізацію та функції:

Реалізація Прелоадера

HTML-структура:

Прелоадер зазвичай містить HTML-елемент (наприклад, div), який відображає анімацію або статичне зображення.

В кодї сайту "FreeLife" може бути структура типу `<div class="preloader"></div>`, де `path/to/loader.gif` - це шлях до анімаційного зображення прелоадера.

CSS-стилі:

Стилі CSS використовуються для позиціонування, розміру та інших візуальних характеристик прелоадера.

Можуть включати анімації, щоб зробити візуальний ефект більш привабливим.

JavaScript-функціональність:

JavaScript використовується для контролювання процесу відображення прелоадера.

Скрипт може визначати, коли весь контент сайту завантажено, і відповідно приховувати прелоадер, відкриваючи доступ до контенту сайту.

Функції Прелоадера

Покращення Користувацького Досвіду:

Прелоадери забезпечують користувачам зворотний зв'язок, що сайт завантажується, замість того, щоб вони бачили порожній екран або неповністю завантажену сторінку.

Естетичний Ефект:

Анімовані прелоадери можуть додати естетичного шарму, особливо коли вони добре інтегровані з загальним дизайном сайту.

Технічна Функція:

Забезпечують час для завантаження всіх ресурсів на сторінці, особливо якщо сайт містить багато візуального контенту, скриптів або сторонніх ресурсів.

Прелоадери є важливою частиною дизайну веб-сайтів, що покращують загальне сприйняття сайту користувачами та забезпечують більш гладке переходження до повноцінного контенту після завантаження сторінки.

- Меню навігації: Фіксоване меню з опціями для переходу до різних розділів сайту (Головна, Як почати, Підтримка, Напрямки, Відгуки, Контакт). Також включає контактний номер.

Меню навігації на сайті "FreeLife" виконує ключову роль у забезпеченні зручності перегляду та доступності інформації для користувачів. Ось докладний опис його особливостей та функціональності:

Особливості Меню Навігації

1) Фіксоване Розташування:

- Меню залишається на одному місці (зазвичай у верхній частині екрану), незалежно від прокрутки сторінки. Це означає, що воно завжди доступне для користувача, навіть коли він переміщається вниз по сторінці.

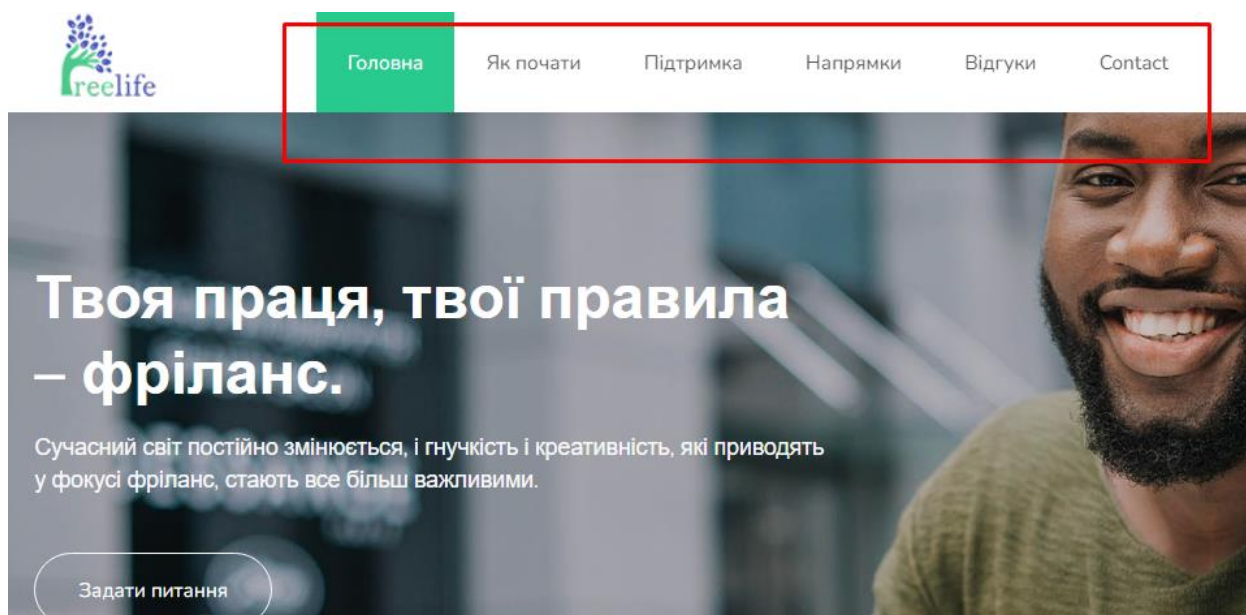


Рисунок 2.2 - Меню навігації

2) Опції для Переходу:

- Меню включає різні опції, такі як "Головна", "Як почати", "Підтримка", "Напрямки", "Відгуки", "Контакт". Кожна з цих опцій веде до відповідного розділу на сайті, забезпечуючи легкий доступ до ключових інформаційних ресурсів.

3) Контактний Номер:

- Включення контактного номеру телефону в меню надає додаткову зручність для користувачів, які бажають швидко зв'язатися з організацією.

Реалізація

1. HTML-структура:

- Меню, зазвичай, реалізується за допомогою тегів `nav` і `ul` в HTML. Наприклад, `<nav><ul><li><a href="#home">Головна</a></li>...<li><a href="#contact">Контакт</a></li></ul></nav>`.

- Кожен пункт меню (`li`) містить гіперпосилання (`a`), яке веде до відповідного розділу сторінки.

2. CSS-стилі:

- CSS використовується для стилізації меню, включаючи шрифти, кольори, фон та поведінку при наведенні курсору.

- Фіксоване положення меню реалізується за допомогою властивостей, таких як `position: fixed;` та `top: 0;`.

3. JavaScript-функціональність:

- У деяких випадках JavaScript може використовуватися для додаткової інтерактивності, наприклад, для анімацій при наведенні або для створення мобільної версії меню (гамбургер-меню).

Функціональність

- Покращення Навігації:
- Дозволяє користувачам швидко знайти та перейти до потрібного їм розділу сайту.

- Збереження Простору:

- Фіксоване меню зберігає екранний простір, забезпечуючи одночасний доступ до контенту та навігаційних елементів.

- Зручність Контакту:

- Включення контактного номеру робить зв'язок з організацією більш простим і доступним.

Таким чином, меню навігації на сайті "FreeLife" є важливою частиною користувацького інтерфейсу, дозволяючи користувачам легко орієнтуватися та знаходити потрібну інформацію.

- Головний розділ (Home): Слайдер з різними вкладками, що представляють сайт.

Головний розділ (Home) сайту "FreeLife" включає слайдер, який є важливим елементом візуального представлення та користувацького досвіду. Слайдер зазвичай використовується для демонстрації важливих інформаційних блоків, пропозицій, або особливостей сайту. Ось докладний опис цього компонента:

Особливості Слайдера

1) Вкладки:

- Слайдер містить кілька вкладок або слайдів, кожен з яких представляє різні аспекти або пропозиції сайту. Наприклад, одна вкладка може зосереджуватися на вступній інформації про сайт, інша — на спеціальних пропозиціях або послугах тощо.

2) Візуальний Дизайн:

- Кожен слайд часто включає як текстову інформацію, так і візуальні елементи, такі як зображення або графіка, створюючи привабливий і зрозумілий інтерфейс.

3) Інтерактивність:

- Слайдер може включати елементи управління, такі як кнопки для переходу до наступного або попереднього слайду, а також індикатори для показу кількості слайдів.

Реалізація Слайдера

1) HTML-структура:

- Слайдер реалізується за допомогою HTML, зазвичай через використання div елементів для кожного слайду.

- Наприклад: `<div class="slider"><div class="slide">...</div><div class="slide">...</div></div>`.

2) CSS-стилі:

- CSS використовується для стилізації слайдів, включаючи розміри, фон, шрифти тощо.

- Можуть бути додаткові стилі для анімацій або переходів між слайдами.

3) JavaScript або jQuery:

- Для додавання інтерактивності та контролю над переходами між слайдами часто використовується JavaScript або jQuery.

- Може включати функціональність автоматичної зміни слайдів через певні інтервали часу.

Функціональність Слайдера

- Презентація Інформації:
- Дозволяє ефективно презентувати ключову інформацію, пропозиції або функції сайту в компактному та привабливому форматі.

- Залучення Користувачів:
- Інтерактивні та візуально привабливі слайди можуть збільшити залученість користувачів та їхню зацікавленість у вмісті сайту.

- Легкість Навігації:
- Наявність кнопок навігації та індикаторів дозволяє користувачам легко переміщуватися між різними слайдами.

Слайдер у головному розділі сайту "FreeLife" є значним елементом в дизайні сторінки, який допомагає відразу зацікавити відвідувачів та надати їм корисну інформацію про сайт та його послуги.

- Функціональні особливості (Feature): Розділ з кроками для початку роботи на сайті.

Розділ "Функціональні Особливості" (Feature) на сайті "FreeLife" має велике значення, оскільки він надає користувачам інформацію про те, як почати користуватися сайтом. Це може включати кроки для реєстрації, огляд ключових функцій сайту, або інструкції щодо використання певних послуг. Ось докладніший опис цього розділу:

Особливості Розділу "Функціональні Особливості"

1) Покрокові Інструкції:

- Розділ може включати послідовні кроки, які пояснюють процес реєстрації на сайті або як почати використовувати його послуги.

- Ці кроки можуть бути представлені у формі списку, інфографіки, або через комбінацію тексту і зображень.

2) Огляд Особливостей Сайту:

- Ця секція може демонструвати унікальні функції або переваги, які відрізняють сайт від інших.

- Може включати інформацію про особливі інструменти, ресурси, або послуги, які пропонує сайт.

3) Візуальний Компонент:

- Застосування візуальних елементів, таких як іконки, зображення або анімації, щоб зробити інформацію більш привабливою та зрозумілою.

Реалізація

1) HTML-структура:

- Використання HTML для створення структури розділу, включаючи різні секції або блоки для кожного кроку або особливості.

2) CSS-стилі:

- Стилiзація кожного блоку або елемента через CSS, включаючи шрифти, кольори, відступи, та інші дизайнерські аспекти.

3) JavaScript-інтерактивність:

- Можливе використання JavaScript для додавання інтерактивних елементів, таких як вкладки, слайдери або модальні вікна.

Функціональність

- Інформативність:
- Надання важливої інформації новим користувачам, щоб допомогти їм легко та ефективно почати користуватися сайтом.

• Залучення Користувачів:

- Використання візуальних та інтерактивних елементів для збільшення залученості та інтересу відвідувачів.

• Зручність та Доступність:

- Забезпечення легкого доступу до інструкцій та рекомендацій, які допомагають користувачам зрозуміти, як ефективно використовувати сайт.

Розділ "Функціональні Особливості" є важливим для забезпечення позитивного першого враження користувачів від сайту та для сприяння їхньому залученню і задоволенню.

- Про нас (About): Інформація про переваги сайту та форма для реєстрації.

Розділ "Про нас" (About) на сайті "FreeLife" має за мету дати відвідувачам глибше розуміння організації, її цінностей, місії, та особливостей, які роблять її унікальною. Також в цьому розділі може бути розміщена форма для реєстрації. Ось більш детальний огляд цього розділу:

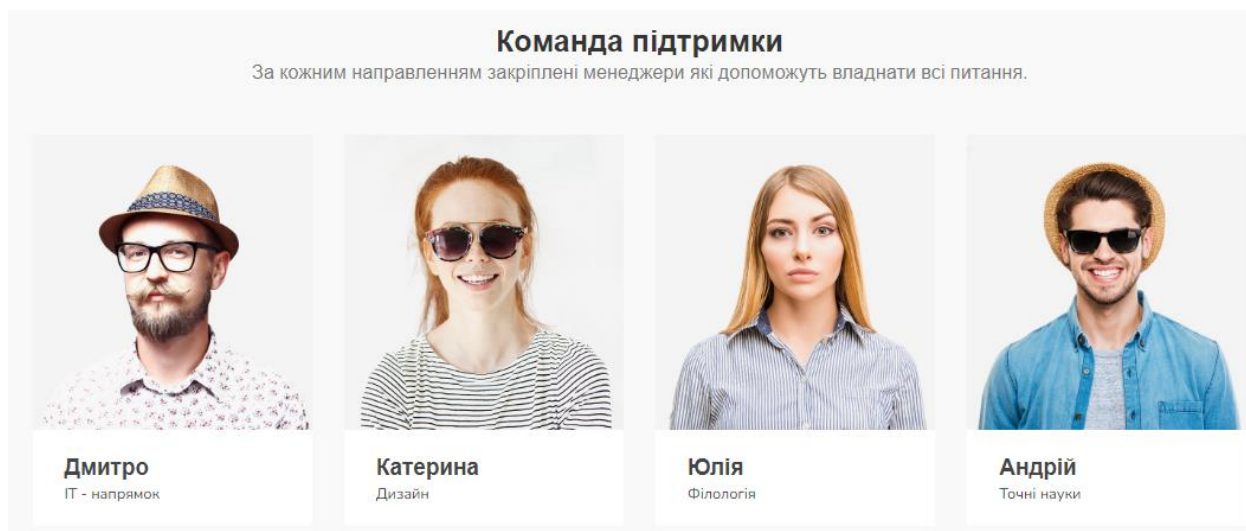


Рисунок 2.3 - Розділ "Про нас" (About)

Особливості Розділу "Про нас"

1) Інформація про Компанію:

- Включає опис історії компанії, її місії, візії та цінностей. Це допомагає створити більш особистий зв'язок з відвідувачами.
- Може також включати досягнення, відзнаки або важливі моменти в історії компанії.

2) Переваги Сайту:

- Опис основних переваг та унікальних особливостей, які пропонує сайт. Це може включати інформацію про особливі послуги, продукти або підходи до роботи.
- Цей розділ допомагає відвідувачам зрозуміти, чому варто обрати саме цей сайт.

3) Візуальні Елементи:

- Використання фотографій, відео або інфографіки, щоб зробити розділ більш привабливим та легким для сприйняття.
- Може включати фотографії команди, офісу, або інші візуальні елементи, які демонструють сутність компанії.

4) Форма Реєстрації:

- Форма для реєстрації або зворотного зв'язку, яка дозволяє відвідувачам легко вступити в контакт з компанією.
- Може включати поля для імені, електронної адреси, телефону, та інших важливих даних.

Реалізація

1) HTML-структура:

- Створення розділу за допомогою HTML, розміщуючи текстовий контент, зображення та форму в структурованому форматі.

2) CSS-стилі:

- Використання CSS для стилізації розділу, включаючи шрифти, кольори, розмітку та адаптацію під різні розміри екранів.

3) JavaScript для Форми:

- Додавання інтерактивності до форми реєстрації, таких як перевірка даних на стороні клієнта або анімації при взаємодії.

Функціональність

- Будівництво Довіри та Репутації:
 - Надання інформації про компанію допомагає побудувати довіру та визнання в очах відвідувачів.
- Залучення Користувачів:

- Візуальні елементи та переконливий текст сприяють більшій залученості користувачів.
- Сприяння Взаємодії:
 - Форма для реєстрації або зворотного зв'язку дозволяє встановити прямий канал спілкування з потенційними клієнтами або користувачами.

Розділ "Про нас" на сайті "FreeLife" відіграє важливу роль у представленні компанії, її цінностей та переваг, а також у встановленні зв'язку з відвідувачами сайту.

- Команда (Team): Інформація про команду підтримки з фотографіями і деталями.

Розділ "Команда" (Team) на сайті "FreeLife" призначений для представлення команди підтримки, включаючи фотографії членів команди та їхні детальні описи. Це допомагає створити більш особисте та довірливе сприйняття компанії з боку відвідувачів сайту. Ось детальний опис цього розділу:

Особливості Розділу "Команда"

1) Фотографії Членів Команди:

- Кожен член команди представлений фотографією, яка допомагає відвідувачам сайту встановити візуальний зв'язок з людьми за компанією.
- Фотографії можуть бути професійними портретами, що підсилюють професійний імідж компанії.

2) Описи та Ролі:

- Для кожного члена команди може бути наданий короткий опис, що включає інформацію про їхню роль у компанії, досвід роботи, освіту, спеціалізацію, досягнення тощо.
- Це допомагає відвідувачам сайту краще зрозуміти, хто працює в компанії і які їхні компетенції.

3) Візуальне оформлення:

					КНУ.РМ.123.23.05.Р	Арк.
	Арк.	№ документа	Підпис	Дата		

- Розділ може бути візуально оформлений в єдиному стилі, що відображає бренд та культуру компанії.
- Може включати елементи дизайну, такі як кольорова схема, шрифти та макет, які узгоджуються з загальним стилем сайту.

Реалізація

1) HTML-структура:

- Використання HTML для створення структури розділу, розташування фотографій та текстових описів.
- Наприклад, використання тегів div для кожного профілю члена команди.

2) CSS-стилі:

- Стилiзація кожного елемента профілю, включаючи фотографії та текст, за допомогою CSS.
- Використання відступів, рамок, тіней, та інших стилів для підкреслення кожного профілю.

3) Адаптивний дизайн:

- Забезпечення того, щоб розділ був оптимізований для перегляду на різних пристроях, включаючи настільні комп'ютери, планшети та смартфони.

Функціональність

- Будівництво Довіри:
 - Представлення реальних людей за компанією допомагає побудувати довіру та створити більш особисті відносини з клієнтами.
- Професійний Імідж:
 - Висвітлення кваліфікації та досвіду членів команди підсилює професійний імідж компанії.
- Прозорість:
 - Надання інформації про членів команди забезпечує прозорість та показує відкритість компанії.

Розділ "Команда" на сайті "FreeLife" відіграє ключову роль у представленні компанії, демонструючи внутрішній склад та різноманітність експертизи своїх співробітників, а також допомагає встановлювати сильніші зв'язки з відвідувачами та потенційними клієнтами.

- Курси (Courses): Популярні напрямки роботи з деталями та цінами.

Розділ "Курси" (Courses) на сайті "FreeLife" призначений для представлення різних курсів або програм навчання, які пропонує компанія. Цей розділ надає детальну інформацію про кожен курс, включаючи його зміст, цілі, ціни та інші важливі деталі. Ось докладний опис цього розділу:

Особливості Розділу "Курси"

1) Детальний Опис Курсів:

- Кожен курс представлений з детальним описом, що може включати його мету, структуру, тривалість, теми, які будуть вивчатися, та інші освітні матеріали.
- Опис допомагає потенційним студентам зрозуміти, що вони отримають від курсу.

Популярні напрямлення

Ви гарантовано отримаєте замовлення з цих тем:

 Програмування Python; JavaScript; Java; C; C++; PHP; Ruby; TypeScript; Swift;  від 35 USD	 Графіка та Дизайн Робота з векторною та растровою графікою. Створення візуального контенту.  від 25 USD	 Філологія Допомога в написанні текстів: копірайт, рерайт, публікації.  від 20 USD
---	--	---

Рисунок 2.4 - Направлення

2) Інформація про Викладачів:

- Може містити інформацію про кваліфікації та досвід викладачів, які проводять курси.

3) Ціни:

- Вказується вартість кожного курсу, що дозволяє потенційним студентам оцінити фінансові аспекти перед зарахуванням.

4) Модульна Структура:

- Може бути представлена модульна структура курсу, що дозволяє зрозуміти організацію та послідовність навчання.

5) Візуальні Елементи:

- Використання зображень, іконок або інфографіки для візуального представлення курсів.

Реалізація

1) HTML-структура:

- Використання HTML для створення блоків кожного курсу, включаючи текстові описи та візуальні елементи.

2) CSS-стилі:

- Стилiзація кожного блоку курсу для забезпечення єдності дизайну та легкості сприйняття інформації.

3) Адаптивний дизайн:

- Забезпечення того, щоб інформація була зручною для перегляду на різних пристроях і розмірах екранів.

Функціональність

• Інформативність:

- Надання повної та зрозумілої інформації про курси, сприяючи ухваленню обґрунтованих рішень потенційними студентами.

• Залучення Студентів:

- Привабливе візуальне представлення та чіткий опис допомагають залучити увагу та зацікавленість студентів.

• Легкість Реєстрації:

- Можливість надання зручних опцій для реєстрації або отримання додаткової інформації про курси.

Розділ "Курси" на сайті "FreeLife" відіграє важливу роль у представленні навчальних можливостей, які пропонує компанія, допомагаючи залучити потенційних студентів та надаючи їм усю необхідну інформацію для прийняття рішення про зарахування на курси.

- Відгуки (Testimonial): Розділ з відгуками клієнтів.

Розділ "Відгуки" (Testimonials) на сайті "FreeLife" має велике значення для створення довіри та позитивного іміджу компанії. Відгуки клієнтів демонструють реальний досвід користувачів з продуктами або послугами, що може значно вплинути на рішення потенційних клієнтів. Ось детальніше про цей розділ:

Особливості Розділу "Відгуки"

1) Реальні Відгуки Клієнтів:

- Зазвичай включає цитати або висловлювання реальних клієнтів, які використовували послуги або продукти компанії.
- Відгуки можуть описувати позитивний досвід користувачів, ефективність продуктів, якість обслуговування тощо.

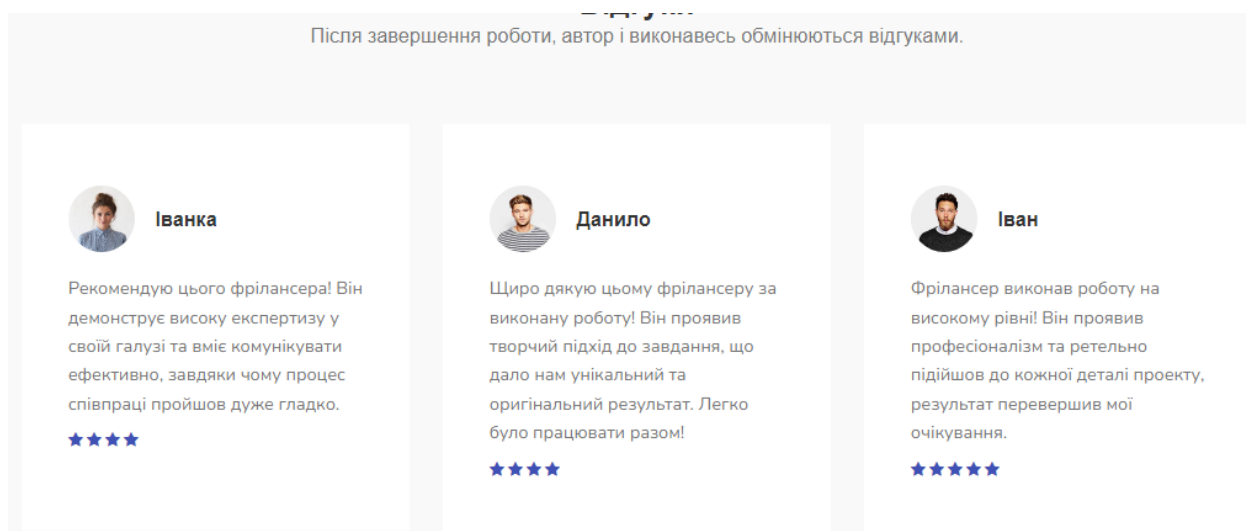


Рисунок 2.5 - Відгуки

2) Фотографії або Відео:

- Фотографії клієнтів або відеовідгуки можуть бути включені для додання більшої переконливості та автентичності.

- Візуальні елементи допомагають створити більш особистий зв'язок з відвідувачами сайту.

3) Позитивний Тон:

- Відгуки, зазвичай, підкреслюють позитивні аспекти досвіду клієнтів, висвітлюючи переваги продуктів або послуг.

Реалізація

1) HTML-структура:

- Використання HTML для створення блоків кожного відгуку, можливо з використанням тегів blockquote для цитат.

2) CSS-стилі:

- Стилiзація блоків відгуків, включаючи шрифти, кольори, та інші візуальні елементи, щоб зробити їх привабливими та легкими для читання.

3) Адаптивний дизайн:

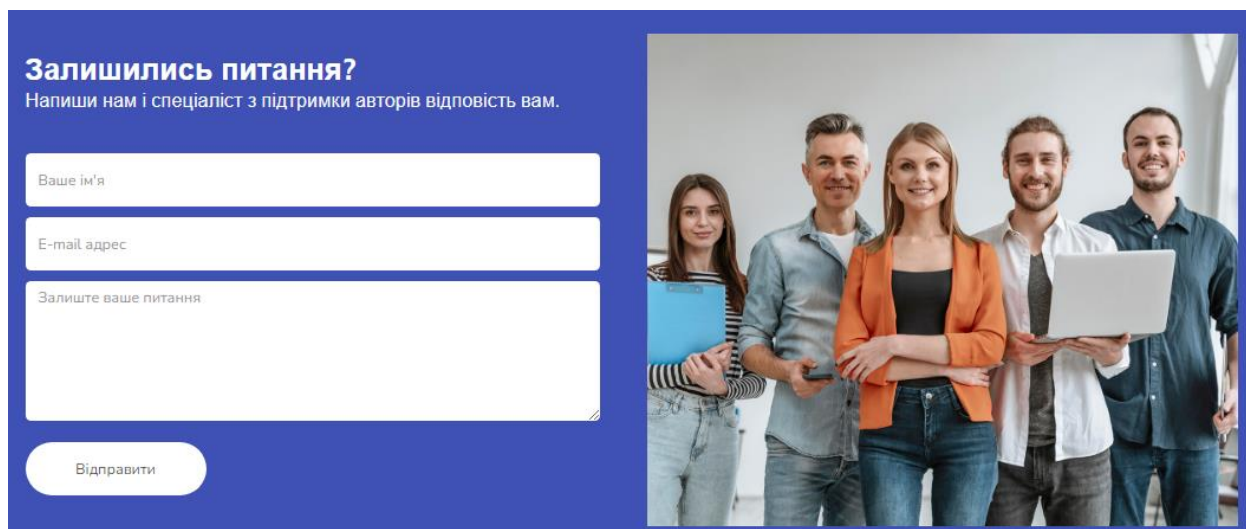
- Оптимізація для перегляду на різних пристроях, щоб забезпечити зручний перегляд відгуків на будь-якому пристрої.

Функціональність

- Будівництво Довіри:
 - Відгуки допомагають будувати довіру до компанії, оскільки потенційні клієнти бачать позитивний досвід реальних людей.
- Залучення Нових Клієнтів:
 - Позитивні відгуки можуть переконати потенційних клієнтів у якості послуг або продуктів.
- Підтримка Позитивного Іміджу:
 - Розділ з відгуками підкреслює задоволеність клієнтів та сприяє позитивному сприйняттю бренду.

Розділ "Відгуки" є важливим елементом сайту "FreeLife", який сприяє підвищенню довіри та залученню нових клієнтів через демонстрацію реальних позитивних досвідів існуючих користувачів.

- **Контакти (Contact):** Форма зворотного зв'язку та контактна інформація.



Залишилися питання?
Напиши нам і спеціаліст з підтримки авторів відповість вам.

Ваше ім'я

E-mail адрес

Залиште ваше питання

The image shows a contact form on a blue background with white text and input fields. To the right of the form is a photograph of five young adults (three men and two women) standing together, smiling, in a modern office or studio setting. One man is holding a laptop.

Рисунок 2.6 - Контакти

Розділ "Контакти" (Contact) на сайті "FreeLife" є ключовим для забезпечення взаємодії з потенційними та існуючими клієнтами. Цей розділ надає всі необхідні засоби для зв'язку з компанією, включаючи форму зворотного зв'язку та різні варіанти контактної інформації. Ось детальніший опис цього розділу:

Особливості Розділу "Контакти"

1) Форма Зворотного Зв'язку:

- Дозволяє відвідувачам сайту легко надіслати запитання, коментарі або повідомлення безпосередньо з сайту.
- Зазвичай включає поля для введення імені, електронної пошти, теми запиту, та тексту повідомлення.

2) Контактна Інформація:

- Надання докладних даних для зв'язку, таких як адреса електронної пошти, номер телефону, факсу, фізична адреса офісу.
- Може включати години роботи для вказівки найкращого часу для контакту.

3) Соціальні Медіа та Інші Канали:

- Посилання на соціальні медіа та інші платформи, де клієнти можуть зв'язатися з компанією або слідкувати за її діяльністю.

4) Карта Місцезнаходження:

- Інтеграція інтерактивної карти, наприклад, Google Maps, для візуального вказівника фізичного розташування офісу або представництва.

Реалізація

1) HTML-структура:

- Використання HTML для створення форми зворотного зв'язку та розділів з контактною інформацією.

2) CSS-стилі:

- Стилiзація форми та контактних даних для забезпечення легкого та зрозумілого читання.

3) JavaScript для Форми:

- Можливе використання JavaScript для додавання функціональності перевірки введених даних та відправлення форми.

Функціональність

- Легкість Комунікації:
 - Надання різних способів для зв'язку спрощує процес комунікації для клієнтів.
- Доступність Інформації:
 - Чітка та зрозуміла контактна інформація допомагає клієнтам легко знаходити необхідні дані для зв'язку.
- Забезпечення Зворотного Зв'язку:
 - Форма зворотного зв'язку дає можливість відвідувачам сайту легко залишати запитання або коментарі.

Розділ "Контакти" є важливим для забезпечення ефективної взаємодії між "FreeLife" та її клієнтами, надаючи всі необхідні інструменти для легкого та зручного зв'язку.

					KHY.PM.123.23.05.P	Арк.
	Арк.	№ документа	Підпис	Дата		

- Футер (Footer): Інформація про офіс, контактні дані підтримки, форма для підписки на розсилку та соціальні мережі.

Футер (Footer) веб-сайту "FreeLife" є важливою частиною дизайну, оскільки він містить ключову інформацію та ресурси, доступні користувачам на будь-якій сторінці сайту. Цей елемент часто використовується для надання контактної інформації, посилань на соціальні мережі, форми підписки та інших корисних ресурсів. Ось детальний опис елементів, які можуть бути включені в футер сайту:

Особливості Футера

1) Інформація про Офіс:

- Може включати фізичну адресу головного офісу компанії, що допомагає відвідувачам знати, де розташована компанія.

2) Контактні Дані Підтримки:

- Надання електронної пошти, телефонних номерів або форми зворотного зв'язку для підтримки, щоб клієнти могли легко зв'язатися з компанією.

3) Форма для Підписки на Розсилку:

- Можливість для відвідувачів підписатися на новини або оновлення від компанії, вказавши свою електронну адресу.

4) Посилання на Соціальні Мережі:

- Іконки або посилання на сторінки компанії в соціальних мережах, таких як Facebook, Twitter, LinkedIn тощо, дозволяють відвідувачам слідкувати за активністю компанії в соцмережах.

Реалізація

1) HTML-структура:

- Використання HTML для структурування різних елементів футера, таких як розділи контактної інформації, соціальних мереж, та форми підписки.

2) CSS-стилі:

- Стилізація футера для забезпечення чіткості та консистенції з іншими частинами сайту. Може включати стилізацію тексту, списків, форм, іконок соціальних мереж тощо.

3) JavaScript для Форми Підписки:

- Використання JavaScript для функціональності форми підписки, наприклад, для перевірки валідності електронної адреси.

Функціональність

- Забезпечення Доступу до Важливої Інформації:
 - Футер забезпечує легкий доступ до важливих контактних даних та інших ресурсів.
- Залучення та Утримання Відвідувачів:
 - Форма підписки та посилання на соціальні мережі сприяють залученню відвідувачів та їх утриманню, надаючи їм можливості для постійного зв'язку з компанією.
- Зручність для Користувачів:
 - Наявність всієї необхідної контактної інформації в одному місці робить навігацію по сайту більш зручною для користувачів.

Футер сайту "FreeLife" є важливим елементом, що забезпечує відвідувачам сайту легкий доступ до ключової інформації та ресурсів компанії, а також сприяє підвищенню залученості та задоволеності користувачів

3) Скрипти:

- jQuery, Bootstrap JS, Owl Carousel, Smooth Scroll, та власні скрипти сайту.

У веб-розробці використання різних JavaScript бібліотек та фреймворків, таких як jQuery, Bootstrap JS, Owl Carousel, Smooth Scroll, та власних скриптів, значно покращує функціональність та інтерактивність сайту. Ось докладніше про кожен з цих інструментів, які можуть бути використані на сайті "FreeLife":

1) jQuery

- **Опис:** jQuery — це швидка, невелика та багатофункціональна JavaScript бібліотека. Вона спрощує обробку HTML-документів, обробку подій, анімацію та взаємодію з Ajax, значно зменшуючи необхідність написання JavaScript-коду.

- **Застосування:** На сайті "FreeLife", jQuery може використовуватися для створення анімованих ефектів, обробки подій (наприклад, кліків по кнопках), маніпуляцій з DOM-елементами, та асинхронних запитів.

2) Bootstrap JS

- **Опис:** Bootstrap — це потужний фронтенд фреймворк для розробки відзивчивих та мобільно-адаптивних веб-сайтів. Bootstrap JS включає в себе різноманітні плагіни на основі jQuery.

- **Застосування:** Використовується для додавання інтерактивних компонентів, таких як модальні вікна, випадаючі меню, каруселі та інші елементи інтерфейсу, що підвищують зручність користування сайтом.

3) Owl Carousel

- **Опис:** Owl Carousel — це плагін jQuery, який дозволяє створювати адаптивні каруселі або слайдери. Він легко налаштовується та підтримує сенсорні жести.

- **Застосування:** Може бути використаний на сайті для створення привабливих слайдерів зображень, відгуків клієнтів, логотипів партнерів тощо.

4) Smooth Scroll

- **Опис:** Smooth Scroll — це функція, яка забезпечує плавну прокрутку сторінки до визначеного елемента або розділу, замість миттєвого переходу.

- **Застосування:** Використовується для покращення користувацького досвіду при переходах між розділами сайту, особливо на односторінкових сайтах або при навігації за допомогою якорних посилань.

5) Власні Скрипти Сайту

- **Опис:** Власні скрипти сайту — це унікальний JavaScript-код, написаний спеціально для певних потреб сайту.
- **Застосування:** Може включати специфічну логіку, таку як валідація форм, індивідуальні анімації, взаємодію з користувацькими елементами інтерфейсу тощо.

Комбінування цих інструментів дозволяє створити більш багатофункціональний, інтерактивний та візуально привабливий веб-сайт. Вони сприяють покращенню загального користувацького досвіду та забезпечують більшу гнучкість та можливості для розробників.

2.2. Використання JavaScript для створення динамічного контенту

У коді веб-сайту JavaScript використовується для додавання інтерактивності та динамічних елементів на сторінку. Ось кілька прикладів, як це реалізовано:

1) **Підключення бібліотек JavaScript:** В коді підключаються кілька JavaScript бібліотек та скриптів, таких як `jquery.js`, `bootstrap.min.js`, `owl.carousel.min.js`, `smoothscroll.js`, і `custom.js`. Ці скрипти дозволяють використовувати готові функції для різних цілей, наприклад, для анімації слайдерів, плавного прокручування сторінки тощо.

Ось приклади їхнього використання у коді (Додаток А):

— **jQuery (`jquery.js`):** jQuery — це широко використовувана JavaScript бібліотека, що дозволяє легко маніпулювати DOM-елементами, обробляти події, анімувати елементи та взаємодіяти з сервером через AJAX. У вашому коді вона може використовуватися в таких місцях, як `custom.js`, для забезпечення різних динамічних функцій на сторінці.

— **Bootstrap (`bootstrap.min.js`):** Bootstrap — це популярний фреймворк, який дозволяє швидко розробляти адаптивні та мобільно-дружні веб-сайти. JavaScript компонент Bootstrap використовується для реалізації таких компонентів, як модальні вікна, випадаючі меню, каруселі тощо.

Наприклад, кнопка `navbar-toggle` у коді використовується для створення гамбургер-меню на мобільних пристроях.

— Owl Carousel (`owl.carousel.min.js`): Owl Carousel — це плагін jQuery для створення гнучких каруселей. У вашому коді він, швидше за все, використовується в розділі з класом `owl-carousel` для створення анімованого слайдера зображень або вмісту.

— Smooth Scroll (`smoothscroll.js`): Цей скрипт дозволяє реалізувати плавне прокручування до різних секцій веб-сторінки. Зазвичай, це використовується для посилань з класом `smoothScroll`, які ви використовуєте в навігаційному меню для плавного переходу до різних частин сторінки.

— Індивідуальний JavaScript (`custom.js`): Це файл індивідуальних скриптів, створений для ваших специфічних потреб. Можливо, він містить додаткову логіку або налаштування для елементів, використаних на сайті.

Ці скрипти підключаються до HTML-файлу зазвичай перед закриваючим тегом `</body>` для оптимізації часу завантаження сторінки.

2) Анімація слайдерів: Використання бібліотеки `owl.carousel.min.js` сприяє створенню інтерактивних слайдерів на сторінці. Це можна побачити в розділі з класом `owl-carousel`.

Ось як це виглядає в коді:

```
<div class="owl-carousel owl-theme home-slider">
  <div class="item item-first">
    <!-- Вміст першого слайду -->
  </div>

  <div class="item item-second">
    <!-- Вміст другого слайду -->
  </div>

  <div class="item item-third">
```

```
<!-- Вміст третього слайду -->
```

```
</div>
```

```
</div>
```

У цьому фрагменті коду:

- owl-carousel і owl-theme: Це класи, які визначають елемент як частину Owl Carousel. owl-carousel використовується для ініціалізації каруселі, а owl-theme дає можливість використовувати стандартні стилі Owl Carousel.
- Кожен div з класом item представляє окремий слайд у каруселі. Ви можете помістити в ці контейнери будь-який вміст, наприклад, зображення, текст, кнопки тощо.

Щоб активувати і налаштувати карусель, ви, ймовірно, використовуєте JavaScript або jQuery у файлі custom.js або в іншому скрипті. Типовий код для ініціалізації Owl Carousel може виглядати так:

```
$(".owl-carousel").owlCarousel({
  loop: true,
  autoplay: true,
  items: 1, // Кількість елементів для показу
  // Інші параметри
});
```

Цей код встановлює основні параметри каруселі, такі як автопрокрутка (autoplay), кількість елементів, що відображаються одночасно (items), і чи буде карусель зациклена (loop).

3) Плавний скролінг: Скрипт smoothscroll.js використовується для реалізації плавного прокручування до різних секцій сайту при кліку на посилання в навігаційному меню.

Ось приклади з коду:

```
<ul class="nav navbar-nav navbar-nav-first">
  <li><a href="#top" class="smoothScroll">Головна</a></li>
  <li><a href="#about" class="smoothScroll">Як почати</a></li>
```

```

<li><a href="#team" class="smoothScroll">Підтримка</a></li>
<li><a href="#courses" class="smoothScroll">Напрямки</a></li>
<li><a href="#testimonial" class="smoothScroll">Відгуки</a></li>
<li><a href="#contact" class="smoothScroll">Contact</a></li>
</ul>

```

У цьому фрагменті:

- Кожен `<a>` тег з класом `smoothScroll` вказує на ідентифікатор секції (`id`) сторінки (наприклад, `href="#about"` вказує на секцію з `id="about"`).
- Коли користувач клікає на одне з цих посилань, `smoothscroll.js` перехоплює стандартну поведінку переходу до якоря та заміняє її плавним прокручуванням до відповідної секції.
- JavaScript або jQuery код, який використовує `smoothscroll.js`, імовірно, ініціалізує цю функцію та визначає тривалість анімації прокручування.

Цей код може виглядати приблизно так:

```

$('.smoothScroll').on('click', function(event) {
    // Ініціалізація плавного прокручування
});

```

Цей код встановлює обробник подій для кліка по посиланнях з класом `smoothScroll`, який активує плавне прокручування до вказаного розділу сторінки.

4) Індивідуальні скрипти: Файл `custom.js` може містити специфічні для сайту JavaScript-функції, які відповідають за унікальну поведінку інтерактивних елементів на сайті.

Загалом, JavaScript на цьому сайті використовується для покращення користувацького досвіду, реалізації анімацій, інтерактивності елементів і адаптації сайту під різні пристрої та розміри екранів.

2.3. Адаптивність сайту для різних пристроїв

У HTML-кодї, адаптивне мобільне меню реалізовано за допомогою Bootstrap (bootstrap.min.js). Ось як це виглядає в кодї:

```
<div class="navbar-header">
    <button    class="navbar-toggle"    data-toggle="collapse"    data-
target=".navbar-collapse">
        <span class="icon icon-bar"></span>
        <span class="icon icon-bar"></span>
        <span class="icon icon-bar"></span>
    </button>

    <!-- LOGO TEXT HERE -->
    <a href="" class="navbar-brand"></a>
    <img src='images/logo.png' alt='My Cool Image!'">
</div>

<!-- MENU LINKS -->
<div class="collapse navbar-collapse">
    <ul class="nav navbar-nav navbar-nav-first">
        <!-- Навігаційні посилання -->
    </ul>
</div>
```

У цьому фрагменті:

- Клас navbar-header містить логотип та кнопку для розкриття меню в мобільному режимі (navbar-toggle). Ця кнопка відображається лише на малих екранах.
- Кнопка navbar-toggle містить три span елементи, які візуально представляють іконку "гамбургера" (три горизонтальні смужки). Атрибути data-toggle="collapse" та data-target=".navbar-collapse" вказують Bootstrap, що

ця кнопка призначена для управління станом колапсу (згорання/розгорання) меню.

- Коли користувач натискає на `navbar-toggle`, клас `collapse navbar-collapse` (який містить основні посилання меню) перемикається між станами згорання та розгорання, що робить меню доступним або прихованим на мобільних пристроях.

Bootstrap автоматично обробляє цю поведінку за допомогою JavaScript, включеного в `bootstrap.min.js`, що дозволяє меню адаптуватися до різних розмірів екранів.

ІІІ. РОЗРАХУНОК ВАРТОСТІ ПРОЕКТУ

У загальному випадку розробка сайту містить у собі наступні етапи:

- початковий етап, на якому формулюються основні вимоги, пропоновані до сайту, описуються основні цілі й розробляються специфікації, тобто виявляються основні властивості, що й характеризують їхні показники;
- етап зовнішнього проєктування, де необхідно розробити архітектуру й структуру сайту, розробити зображення, сценарій та необхідний музикальний супровід, визначити алгоритм побудови, а також розробити зовнішній інтерфейс користувача;
- етап проєктування і кодування компонентів (у ході виконання даного етапу відбувається проєктування і кодування обраною мовою програмування окремих модулів сайту);
- на основному етапі необхідно зробити налагодження і тестування окремих програмних модулів, потім – комплексне налагодження всього проєкту в цілому;
- заключний етап, на якому проводиться остаточна корекція сайту і готується необхідна супровідна документація. Здійснимо розрахунок собівартості і ціни розробки сайту.

У собівартість розробки сайту входять наступні статті витрат:

- основна заробітна плата;
- додаткова заробітна плата;
- єдиний соціальний внесок;
- інші витрати.

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата	РОЗРАХУНОК ВАРТОСТІ ПРОЕКТУ			
Розробив	Анцигін							
Перевірив	Купін							
Н.контроль	Кузнецов							
Затвердив	Купін				Літера Аркуш Аркушів 3КІ-22м			

Розробку сайту проводять, щонайменше, чотири фахівці: сценарист, дизайнер, програміст-інженер та звукорежисер. Зарплата сценариста (інженер по технічним вимогам) становить 100 грн./год, програміста – 150 грн./год, дизайнера – 120 грн./год, та звукорежисер – 100 грн/год. При цьому тривалість робочого дня кожного з них становить 8 годин. Комп’ютерна гра розроблялась 1 місяць [14].

Розрахунок основної заробітної плати наведено в таблиці 4.1. Додаткова заробітна плата – це винагорода за працю понад установлені норми, за трудові успіхи та винахідливість і за особливі умови праці. Вона включає доплати, надбавки, гарантійні та компенсаційні виплати, передбачені чинним законодавством; премії, пов’язані з виконанням виробничих завдань і функцій. У даному випадку додаткова заробітна плата становить 20 % від основної: $10000 * 0,2 = 2000$ грн.

У таблиці 4.1 показані розрахунки витрат на заробітну плату, а саме декілька етапів: початковий, графічна частина, Розробка й кодування компонентів, основні етапи, заключні етапи та розрахунок усього разом із додатковою заробітною платою. Ставка єдиного соціального внеску становить 22 % від величини основної та додаткової заробітної плати: $12000 * 0,22 = 2640$ грн

Таблиця 5.1 – Розрахунок витрат на заробітну плату

Етап	Вид робіт	Виконавець	Годинна ставка	Тривалість виконання, дні	Заробітна плата, грн.	Кількість Посада
1	Початковий	інженер	40	1	1500	1
2	Графічна частина	дизайнер	35	3	2000	1
3	Розробка й кодування компонентів	програміст	40	15	3000	1
4	Розробка музичного супроводу	звукорежисер	35	5	2000	1
5	Основний етап	програміст	40	3	1000	1
6	Основний етап	програміст	40	1	800	1

7	Основний етап	програміст	40	1	500	1
8	Заклучний етап	інженер	40	1	200	1
Разом					10000	30
Додаткова заробітна плата (20 %)					2000	
Усього					12000	

До інших витрат слід віднести витрати на обслуговування ЕОМ і плату за електроенергію. Витрати на електроенергію розраховуються виходячи зі споживаної потужності устрою і тарифу на електроенергію. У даному випадку передбачається використання 3 комп'ютерів з потужністю 0,7 кВт/год. Вартість однієї кВт/год електроенергії прийнято у розмірі 1,619 грн. Час використання електроенергії в процесі розробки: $30 * 8 = 240$ години.

Отже, плата за електроенергію складе: $0,7 * 1,619 * 240 * 3 = 815,98$ грн. Витрати на обслуговування ЕОМ визначаються з вартості ЕОМ і часу її експлуатації, після закінчення якого, вона підлягає заміні (звичайно цей час не перевищує 3-х років), протягом року ЕОМ використовує 254 робочих дні:

$$(7200 / (3 * 30 * 254)) * 240 = 75,60 \text{ грн.}$$

Проект впроваджується на 1 компанію, тому собівартість розробки становить 15531,58 грн.:

$$15531,58/1=15531,58 \text{ грн.}$$

Розрахуємо суму прибутку від реалізації розробки (виходячи з рівня рентабельності 30 %): $15531,58 * 0,3 = 4659,47$ грн. Розрахуємо ціну розробки гри без податку на додану вартість (ПДВ):

$$15531,58 + 4659,47 = 20191,05 \text{ грн.}$$

Розрахуємо суму ПДВ, вона становить 20 % від ціни: $20191,05 * 0,2 = 4038,21$ грн. Розрахуємо ціну гри з урахуванням ПДВ:

$$20191,05 + 4038,21 = 24229,26 \text{ грн.}$$

Результати розрахунків наведено у таблиці 4.2

Таблиця 5.2 – Розрахунок витрат на розробку та ціни

					КНУ.РМ.123.23.05.Р	Арк.
	Арк.	№ документа	Підпис	Дата		

Стаття витрат	Сума, грн.
Основна заробітна плата	10000,00
Додаткова заробітна плата	2000,00
Єдиний соціальний внесок	2640,00
Витрати на обслуговування ЕОМ	75,60
Витрати на електроенергію	815,98
Собівартість розробки комп'ютерної гри	15531,58
Прибуток (рівень рентабельності 30 %)	4659,47
Ціна без ПДВ	20191,05
Податок на додану вартість (ПДВ)	4038,21
Ціна з урахуванням ПДВ	24229,26

Таким чином, виходячи з виконаних розрахунків, повна вартість розробки складатиме 24229,26 грн. Термін виконання всіх етапів розробки становить 30 днів (1 місяць) для команди з одного програміста, дизайнера, звукорежисера та сценариста. Очікувана сума прибутку складе 4659,47 грн.

IV. ТЕСТУВАННЯ ТА НАЛАГОДЖЕННЯ САЙТУ

Тестування та налагодження вебсайту, як у вашому випадку, є важливими кроками для забезпечення його ефективної роботи та відмінного користувацького досвіду. Ось кілька кроків, які ви можете вжити для тестування та налагодження сайту:

– Функціональне тестування:

- 1) Перевірка Посилань: Забезпечте, що всі посилання на сайті працюють правильно (включаючи внутрішні та зовнішні посилання).
- 2) Тестування Форм: Переконайтеся, що всі форми (наприклад, форма реєстрації) працюють належним чином і дані відправляються коректно.
- 3) Перевірка Скриптів: Тестування всіх клієнтських та серверних скриптів (наприклад, JavaScript).

– Інтерфейсне тестування:

- 1) Адаптивний Дизайн: Переконайтеся, що ваш сайт правильно відображається та адаптується до різних розмірів екранів і пристроїв.
- 2) Крос-браузерне Тестування: Перевірте, чи сайт однаково працює у різних веб-браузерах.

– Тестування продуктивності:

- 1) Швидкість Завантаження: Використовуйте інструменти, такі як Google PageSpeed Insights, для оцінки та оптимізації швидкості завантаження сторінок.
- 2) Навантажувальне Тестування: Тестуйте, як сайт витримує велике навантаження (багато користувачів одночасно).

– Безпекове Тестування:

- 1) Перевірка на Вразливості: Використовуйте спеціалізовані інструменти для виявлення потенційних слабких місць безпеки.

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата	ТЕСТУВАННЯ ТА НАЛАГОДЖЕННЯ САЙТУ			
Розробив	Анцигін							
Перевірив	Купін							
Н.контроль	Кузнецов							
Затвердив	Купін				3КІ-22м			

2) SSL/HTTPS: Переконайтеся, що з'єднання захищене та використовується HTTPS.

– Тестування Сумісності:

1) Плагіни: Переконайтеся, що всі плагіни або зовнішні бібліотеки сумісні з вашим сайтом та оновлені.

2) Залежності: Перевірте залежності вашого сайту, такі як версії PHP, бази даних тощо.

– Удосконалення SEO:

1) Оптимізація Метаданих: Впевніться, що всі сторінки мають відповідні мета-теги (title, description, keywords).

2) Перевірка зображень: Всі зображення мають мати атрибути 'alt' для кращого індексування пошуковими системами.

– Зворотній Зв'язок від Користувачів:

1) Тестування Користувачами: Отримайте відгуки від реальних користувачів щодо їх досвіду взаємодії з сайтом.

2) Аналітика: Використовуйте Google Analytics або аналогічні інструменти для відстеження поведінки користувачів на сайті.

– Документація та Підтримка:

1) Помилки та Проблеми: Записуйте усі знайдені помилки та проблеми, а також способи їх вирішення.

2) Регулярне Оновлення: Підтримуйте сайт оновленим, регулярно перевіряючи та оновлюючи контент та технічну частину.

– Заключні Кроки:

1) Бекап: Регулярно робіть резервні копії сайту.

2) Моніторинг: Використовуйте інструменти для моніторингу стану сайту (наприклад, uptime monitoring).

Тестування та налагодження сайту - це постійний процес, який вимагає регулярних оновлень та уваги.

ВИСНОВКИ

Дослідження показало, що JavaScript є важливим інструментом у розробці сучасних веб-сайтів, зокрема для сфери фріланс-послуг. Аналіз сфери фрілансу виявив особливі потреби цільової аудиторії, такі як легкість доступу до інформації, швидкість роботи сайту та наявність інтерактивних елементів.

Використання JavaScript дозволило ефективно реалізувати архітектуру та інтерфейс сайту, забезпечуючи його функціональність та привабливість. Динамічний контент, створений за допомогою JavaScript, значно покращив взаємодію користувача з сайтом, забезпечуючи високу адаптивність під різні пристрої та екрани.

Розробка сайту з використанням JavaScript виявилася економічно вигідною, оскільки це дозволило скоротити витрати на розробку і підтримку, при цьому забезпечуючи високий рівень задоволення користувачів.

Ретельне тестування сайту виявило деякі недоліки, які були усунені за допомогою налагодження коду. Після цього сайт показав високу надійність та продуктивність.

У підсумку, використання JavaScript для розробки сайту для фріланс-послуг виявилось ефективним рішенням, що сприяє залученню більшої аудиторії, підвищує задоволеність користувачів та забезпечує високу економічну ефективність проекту.

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Анцигін				ВИСНОВКИ	Літера	Аркуш	Аркушів
Перевірив	Купін							
Н.контроль	Кузнецов					ЗКІ-22м		
Затвердив	Купін							

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Кузьмін О.Є. Фріланс та загальна характеристика фрілансера / О.Є. Кузьмін, Н.Ю. Солярчук // Науковий вісник НЛТУ України. НУ “Львівська політехніка, 2012. – Вип. 22.7. – С. 375–380.
2. Мартеріал з вікіпедії. URL: <https://uk.wikipedia.org/wiki/Фрілансер> (дата звернення 07.03.2023).
3. Стаття у словнику термінів: значення слова. URL: <http://www.businessdictionary.com/definition/freelance.html> (дата звернення 16.06.2023).
4. Архів Одеської юридичної академії: Андропова В.А. Правове регулювання праці фрілансерів: юридична довідка. URL: <http://dspace.onua.edu.ua> (дата звернення 19.06.2023).
5. Гніденко В.І. Правове регулювання гнучких форм зайнятості : дис.. ... канд. юрид. наук : 12.00.05. Східноукраїнський нац. ун-т ім. Володимира Даля, Національний юрид. ун-т ім. Ярослава Мудрого. Харків. 2021. 283 с. С. 1.
6. Колот А.М. Соціально-трудова сфера: стан відносин, нові виклики, тенденції розвитку : монографія. Київ : КНЕУ, 2010. 251 с. С. 73.
7. Котерлін І.Б., Штих А.Т. Нормативно-правове врегулювання нестандартної зайнятості населення в Україні через впливи пандемічного обмеження. Актуальні проблеми вітчизняної юриспруденції. 2021. № 1. С. 69–73.
8. Лєскова Л.Ф. Фріланс та його перспективи на українському ринку праці. Topical Issues of Science and Education. 2017. Vol. 3. С. 22–25.

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата				
Розробив	Анцигін				СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ		Літера	Аркуш
Перевірив	Купін							
Н.контроль	Кузнецов						ЗКІ-22м	
Затвердив	Купін							

9. Літорович О.В. Аналіз ринку фрілансу в Україні. Вісник Національного університету «Львівська політехніка». Серія «Проблеми економіки та управління». 2020. № 1. Т. 4. С. 43–53.

10. Про внесення змін до деяких законодавчих актів України щодо удосконалення правового регулювання дистанційної, надомної роботи та роботи із застосуванням гнучкого режиму робочого часу : Закон України від 04.02.2021. Відомості Верховної Ради України (ВВР), 2021, № 20, ст. 178.

11. Трофименко М.С. Міжнародний досвід правового регулювання трудових відносин у сфері інформаційних технологій. Право та інновації. 2016. № 3 (15). С. 117–122.

12. Щодо проблем застосування актів законодавства про працю : Лист Міністерства праці та соціальної політики України № 105/13/116-08 від 15.05.2008 р. URL: http://search.ligazakon.ua/l_doc2.nsf/link1/FIN43309.html (дата звернення 24.06.2023).

13. Сайт пошуку роботи та надання фріланс послуг: Fiverr: приклад оформлення інтерфейсу. URL:

14. https://www.fiverr.com/?utm_source=20435&utm_medium=cx_affiliate&utm_campaign=&afp=eyJwaWQiOiIzZDE2OTEwNzAwMDk4ODkiLCJjaWQiOiIzZDE2OTEwNzAwMDk4ODI8MXw2NzI1MyIsICJzIjoiMyJ9__cor467733323&cxd_token=20435_16349140_eyJwaWQiOiIzZDE2OTEwNzAwMDk4ODkiLCJjaWQiOiIzZDE2OTEwNzAwMDk4ODI8MXw2NzI1MyIsICJzIjoiMyJ9__cor467733323&show_join=true (дата звернення 28.06.2023).

15. Сайт пошуку роботи та надання фріланс послуг: Upwork: приклад оформлення інтерфейсу. URL: <https://www.upwork.com/> (дата звернення 28.06.2023).

16. Сайт пошуку роботи та надання фріланс послуг: Toptal: приклад оформлення інтерфейсу. URL: <https://www.toptal.com/> (дата звернення 04.07.2023).

17. Сайт пошуку роботи та надання фріланс послуг: DesignCrowd: приклад оформлення інтерфейсу. URL: <https://www.designcrowd.com/> (дата звернення 04.07.2023).

18. Сайт пошуку роботи та надання фріланс послуг: 99Designs: приклад оформлення інтерфейсу. URL: <https://99designs.com/> (дата звернення 06.07.2023).

19. Сайт пошуку роботи та надання фріланс послуг: Freelancer.com приклад оформлення інтерфейсу. URL: https://www.freelancermap.com/it-projects.html?campaignid=9250790313&adgroupid=95525698484&adid=416512861353&gclid=Cj0KCQjwoK2mBhDzARIsADGbjeo3W9Olwxe7XtNVqm7CtdKSсра4fhmiz_WFHCdMJtifddOiW60sVa4aAvUfEALw_wcB (дата звернення 08.07.2023).

20. Сайт що містить актуальну інформацію та тренди щодо розвитку фріланс послуг: N26 інформація про особливості розвитку. URL: <https://n26.com/en-eu/blog/what-is-freelancing> (дата звернення 20.08.2023).

ДОДАТКИ

```
<!DOCTYPE
html>
```

```
<html lang="en">
```

```
<head>
```

```
<title>FreeLife - Кращий сайт для вільної творчості</title>
```

```
<meta charset="UTF-8">
```

```
<meta http-equiv="X-UA-Compatible" content="IE=Edge">
```

```
<meta name="description" content="">
```

```
<meta name="keywords" content="">
```

```
<meta name="author" content="">
```

```
<meta name="viewport" content="width=device-width, initial-
scale=1, maximum-scale=1">
```

```
<link rel="stylesheet" href="css/bootstrap.min.css">
```

```
<link rel="stylesheet" href="css/font-awesome.min.css">
```

```
<link rel="stylesheet" href="css/owl.carousel.css">
```

```
<link rel="stylesheet" href="css/owl.theme.default.min.css">
```

```
<!-- MAIN CSS -->
```

```
<link rel="stylesheet" href="css/templatemo-style.css">
```

```
</head>
```

```
<body id="top" data-spy="scroll" data-target=".navbar-collapse"
data-offset="50">
```

```
<!-- PRE LOADER -->
```

					КНУ.РМ.123.23.05.Р			
Змн.	Арк.	№ документа	Підпис	Дата	ДОДАТКИ			
Розробив	Анцигін							
Перевірів	Купін							
Н.контроль	Кузнецов							
Затвердив	Купін				3КІ-22м			

```

<section class="preloader">

<div class="spinner">

<span class="spinner-rotate"></span>

</div>

</section>

<!-- MENU -->

<section class="navbar custom-navbar navbar-fixed-top"
role="navigation">

<div class="container">

<div class="navbar-header">

<button class="navbar-toggle" data-toggle="collapse" data-
target=".navbar-collapse">

<span class="icon icon-bar"></span>

<span class="icon icon-bar"></span>

<span class="icon icon-bar"></span>

</button>

<!-- LOGO TEXT HERE -->

<a href="" class="navbar-brand"></a>

<img src='images/logo.png' alt='My Cool Image!'">

</div>

<!-- MENU LINKS -->

<div class="collapse navbar-collapse">

<ul class="nav navbar-nav navbar-nav-first">

<li><a href="#top" class="smoothScroll">Головна</a></li>

<li><a href="#about" class="smoothScroll">Як почати</a></li>

```

```

<li><a href="#team" class="smoothScroll">Підтримка</a></li>

<li><a href="#courses" class="smoothScroll">Напрямки</a></li>

<li><a href="#testimonial"
class="smoothScroll">Відгуки</a></li>

<li><a href="#contact" class="smoothScroll">Contact</a></li>

</ul>

<ul class="nav navbar-nav navbar-right">

<li><a href="#"><i class="fa fa-phone"></i> +38 044 123 45
67</a></li>

</ul>

</div>

</div>

</section>

<!-- HOME -->

<section id="home">

<div class="row">

<div class="owl-carousel owl-theme home-slider">

<div class="item item-first">

<div class="caption">

<div class="container">

<div class="col-md-6 col-sm-12">

<h1>FreeLife: де талант знаходить своє застосування.</h1>

<h3>Фріланс - це спосіб працювати на відстані або
безпосередньо замовляти різні проекти. </h3>

<a href="#feature" class="section-btn btn btn-default
smoothScroll">Кроки</a>

```

</div>

</div>

</div>

</div>

<div class="item item-second">

<div class="caption">

<div class="container">

<div class="col-md-6 col-sm-12">

<h1>Розвивай свій талант, фріланс – це майбутнє.</h1>

<h3>Фріланс дозволяє людям розвивати свої таланти, здібності та навички в більш гнучких і креативних умовах.</h3>

Обрати напрямлення

</div>

</div>

</div>

</div>

<div class="item item-third">

<div class="caption">

<div class="container">

<div class="col-md-6 col-sm-12">

<h1>Твоя праця, твої правила – фріланс.</h1>

<h3>Сучасний світ постійно змінюється, і гнучкість і креативність, які приводять у фокусі фріланс, стають все більш важливими. </h3>

Задати питання

					КНУ.РМ.123.23.05.Р	Арк.
	Арк.	№ документа	Підпис	Дата		

</div>

</div>

</div>

</div>

</div>

</div>

</section>

<!-- FEATURE -->

<section id="feature">

<div class="container">

<div class="row">

<div class="col-md-4 col-sm-4">

<div class="feature-thumb">

01

<h3>Реєстрація</h3>

<p>Щоб взяти своє перше замовлення вам необхідно пройти швидко але надійну реєстрацію на сайті</p>

</div>

</div>

<div class="col-md-4 col-sm-4">

<div class="feature-thumb">

02

<h3>Вибір напрямку</h3>

<p>Визначте свої інтереси та навички: Подумайте про те, що вам подобається робити і в чому ви відчуваєте себе сильним.</p>

```
</div>
```

```
</div>
```

```
<div class="col-md-4 col-sm-4">
```

```
<div class="feature-thumb">
```

```
<span>03</span>
```

```
<h3>Перше замовлення</h3>
```

```
<p>Оберіть замовлення яке вам підходить і запропонуйте себе  
як автора.</p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- ABOUT -->
```

```
<section id="about">
```

```
<div class="container">
```

```
<div class="row">
```

```
<div class="col-md-6 col-sm-12">
```

```
<div class="about-info">
```

```
<h2>Наши переваги:</h2>
```

```
<figure>
```

```
<span><i class="fa fa-users"></i></span>
```

```
<figcaption>
```

```
<h3>Цілодобова підтримка</h3>
```

<p>Ви можете звернутись в будьякий час за допомогою до нашого менеджера.</p>

</figcaption>

</figure>

<figure>

<i class="fa fa-certificate"></i>

<figcaption>

<h3>Міжнародні замовлення</h3>

<p>На нашому ресурсі є замовники як з України так і з країн ЄС.</p>

</figcaption>

</figure>

<figure>

<i class="fa fa-bar-chart-o"></i>

<figcaption>

<h3>Без комісії</h3>

<p>Ви отримуєте роботу і за неї 100% гонорару ваші.</p>

</figcaption>

</figure>

</div>

</div>

<div class="col-md-offset-1 col-md-4 col-sm-12">

<div class="entry-form">

<form action="#" method="post">

<h2>Реєструйся прямо зараз:</h2>

```
<input type="text" name="full name" class="form-control"
placeholder="ПІБ" required="">
```

```
<input type="email" name="email" class="form-control"
placeholder="Особистий e-mail" required="">
```

```
<input type="password" name="password" class="form-control"
placeholder="Надійний пароль" required="">
```

```
<button class="submit-btn form-control" id="form-
submit">Зареєструватись</button>
```

```
</form>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- TEAM -->
```

```
<section id="team">
```

```
<div class="container">
```

```
<div class="row">
```

```
<div class="col-md-12 col-sm-12">
```

```
<div class="section-title">
```

```
<h2>Команда підтримки <small>За кожним направленням
закріплені менеджери які допоможуть владнати всі
питання.</small></h2>
```

```
</div>
```

```
</div>
```

```
<div class="col-md-3 col-sm-6">
```

```

<div class="team-thumb">

<div class="team-image">



</div>

<div class="team-info">

<h3>Дмитро</h3>

<span>ІТ - напрямок</span>

</div>

</div>

</div>

<div class="col-md-3 col-sm-6">

<div class="team-thumb">

<div class="team-image">



</div>

<div class="team-info">

<h3>Катерина</h3>

<span>Дизайн</span>

</div>

</div>

</div>

<div class="col-md-3 col-sm-6">

<div class="team-thumb">

```

```

<div class="team-image">



</div>

<div class="team-info">

<h3>Юлія</h3>

<span>Філологія</span>

</div>

</div>

</div>

<div class="col-md-3 col-sm-6">

<div class="team-thumb">

<div class="team-image">



</div>

<div class="team-info">

<h3>Андрій</h3>

<span>Точні науки</span>

</div>

</div>

</div>

</div>

</div>

</section>

```

```

<!-- Courses -->

<section id="courses">

<div class="container">

<div class="row">

<div class="col-md-12 col-sm-12">

<div class="section-title">

<h2>Популярні напрямлення <small>Ви гарантовано отримаєте
замовлення з цих тем:</small></h2>

</div>

<div class="owl-carousel owl-theme owl-courses">

<div class="col-md-4 col-sm-4">

<div class="item">

<div class="courses-thumb">

<div class="courses-top">

<div class="courses-image">



</div>

</div>

<div class="courses-detail">

<h3><a href="#">Програмування</a></h3>

<p>Python; JavaScript; Java; C; C++; PHP; Ruby; TypeScript;
Swift;</p>

</div>

<div class="courses-info">

<div class="courses-author">

```

```

```

```
<span></span>
```

```
</div>
```

```
<div class="courses-price">
```

```
<a href="#"><span> від 35 USD</span></a>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="col-md-4 col-sm-4">
```

```
<div class="item">
```

```
<div class="courses-thumb">
```

```
<div class="courses-top">
```

```
<div class="courses-image">
```

```

```

```
</div>
```

```
</div>
```

```
<div class="courses-detail">
```

```
<h3><a href="#">Графіка та Дизайн</a></h3>
```

```
<p>Робота з векторною та растровою графікою. Створення
візуального контенту.</p>
```

```
</div>
```

```
<div class="courses-info">
```

```

<div class="courses-author">



<span></span>

</div>

<div class="courses-price">

<a href="#"><span>від 25 USD</span></a>

</div>

</div>

</div>

</div>

</div>

<div class="col-md-4 col-sm-4">

<div class="item">

<div class="courses-thumb">

<div class="courses-top">

<div class="courses-image">



</div>

</div>

<div class="courses-detail">

<h3><a href="#">Філологія</a></h3>

<p>Допомога в написанні текстів: копірайт, рерайт,
публікації.</p>

</div>

```

```

<div class="courses-info">

<div class="courses-author">



<span></span>

</div>

<div class="courses-price">

<a href="#"><span>від 20 USD</span></a>

</div>

</div>

</div>

</div>

</div>

<div class="col-md-4 col-sm-4">

<div class="item">

<div class="courses-thumb">

<div class="courses-top">

<div class="courses-image">



</div>

</div>

<div class="courses-detail">

<h3><a href="#">Маркетинг</a></h3>

<p>Будування стратегій розвитку підприємств. SMM -
спеціалісти та маркетологи.</p>

```

```
</div>

<div class="courses-info">

<div class="courses-author">



<span></span>

</div>

<div class="courses-price">

<a href="#"><span>від 20 USD</span></a>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

</section>

<!-- TESTIMONIAL -->

<section id="testimonial">

<div class="container">

<div class="row">

<div class="col-md-12 col-sm-12">

<div class="section-title">
```

<h2>Відгуки <small>Після завершення роботи, автор і виконавесь обмінюються відгуками.</small></h2>

</div>

<div class="owl-carousel owl-theme owl-client">

<div class="col-md-4 col-sm-4">

<div class="item">

<div class="tst-image">

</div>

<div class="tst-author">

<h4>Іван</h4>

</div>

<p>Фрілансер виконав роботу на високому рівні! Він проявив професіоналізм та ретельно підійшов до кожної деталі проекту, результат перевершив мої очікування.</p>

<div class="tst-rating">

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

</div>

</div>

</div>

<div class="col-md-4 col-sm-4">

```

<div class="item">

<div class="tst-image">



</div>

<div class="tst-author">

<h4>Марина</h4>

<span></span>

</div>

<p>Дуже задоволений співпрацею з цим фрілансером! Він
виявився дуже відповідальним та вміє оперативно вирішувати
завдання, завдяки йому проект був завершений швидко та
бездоганно.</p>

<div class="tst-rating">

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

</div>

</div>

</div>

<div class="col-md-4 col-sm-4">

<div class="item">

<div class="tst-image">



</div>

```

```
<div class="tst-author">
```

```
<h4>Іванка</h4>
```

```
<span></span>
```

```
</div>
```

```
<p>Рекомендую цього фрілансера! Він демонструє високу експертизу у своїй галузі та вміє комунікувати ефективно, завдяки чому процес співпраці пройшов дуже гладко.</p>
```

```
<div class="tst-rating">
```

```
<i class="fa fa-star"></i>
```

```
<i class="fa fa-star"></i>
```

```
<i class="fa fa-star"></i>
```

```
<i class="fa fa-star"></i>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="col-md-4 col-sm-4">
```

```
<div class="item">
```

```
<div class="tst-image">
```

```

```

```
</div>
```

```
<div class="tst-author">
```

```
<h4>Данило</h4>
```

```
<span></span>
```

```
</div>
```

<p>Щиро дякую цьому фрілансеру за виконану роботу! Він проявив творчий підхід до завдання, що дало нам унікальний та оригінальний результат. Легко було працювати разом!</p>

<div class="tst-rating">

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

<i class="fa fa-star"></i>

</div>

</div>

</div>

</div>

</div>

</div>

</section>

<!-- CONTACT -->

<section id="contact">

<div class="container">

<div class="row">

<div class="col-md-6 col-sm-12">

<form id="contact-form" role="form" action="" method="post">

<div class="section-title">

<h2>Залишилися питання? <small>Напиши нам і спеціаліст з підтримки авторів відповість вам.</small></h2>

</div>

<div class="col-md-12 col-sm-12">

```
<input type="text" class="form-control" placeholder="Ваше ім'я"
name="name" required="">
```

```
<input type="email" class="form-control" placeholder="Е-mail
адрес" name="email" required="">
```

```
<textarea class="form-control" rows="6" placeholder="Залиште
ваше питання" name="message" required=""></textarea>
```

```
</div>
```

```
<div class="col-md-4 col-sm-12">
```

```
<input type="submit" class="form-control" name="send message"
value="Відправити">
```

```
</div>
```

```
</form>
```

```
</div>
```

```
<div class="col-md-6 col-sm-12">
```

```
<div class="contact-image">
```

```

```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- FOOTER -->
```

```
<footer id="footer">
```

```
<div class="container">
```

```
<div class="row">
```

```
<div class="col-md-4 col-sm-6">
```

```
<div class="footer-info">
```

```
<div class="section-title">
```

```
<h2>Наш офіс</h2>
```

```
</div>
```

```
<address>
```

```
<p>вулиця Хрещатик, 22, Київ, 01001<br> Khreschatyk street,  
22, Kiev, 01001</p>
```

```
</address>
```

```
<ul class="social-icon">
```

```
<li><a href="#" class="fa fa-facebook-square" attr="facebook  
icon"></a></li>
```

```
<li><a href="#" class="fa fa-twitter"></a></li>
```

```
<li><a href="#" class="fa fa-instagram"></a></li>
```

```
</ul>
```

```
<div class="copyright-text">
```

```
<p>FreeLife &copy; 2013 Company</p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="col-md-4 col-sm-6">
```

```
<div class="footer-info">
```

```
<div class="section-title">
```

```
<h2>Підтримка:</h2>
```

```
</div>
```

```

<address>

<p>+38 044 123 45 67</p>

<p><a href="mailto:youremail.com">help@freelife.com</a></p>

</address>

</div>

</div>

<div class="col-md-4 col-sm-12">

<div class="footer-info newsletter-form">

<div class="section-title">

<h2>Россилка замовлень</h2>

</div>

<div>

<div class="form-group">

<form action="#" method="get">

<input type="email" class="form-control" placeholder="ваш
email" name="email" id="email" required="">

<input type="submit" class="form-control" name="submit"
id="form-submit" value="Підписатись">

</form>

<span><sup>*</sup> Завжди перевіряйте папку "спам"</span>

</div>

</div>

</div>

</div>

</div>

```

```

</div>

</footer>

<!-- SCRIPTS -->

<script src="js/jquery.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/owl.carousel.min.js"></script>

<script src="js/smoothscroll.js"></script>

<script src="js/custom.js"></script>

</body>

</html>

<div class="domain">
<div class="domain-box">
<h3>Domain 3</h3>
<p class="price">$300</p>
</div>
</div>

</div>

<div class="lerom-epson">
<p>Lerom Epson</p>
</div>

<div class="footer">
<button class="purchase-btn">Purchase</button>
<button class="consultation-btn">Consultation</button>
</div>
</div>

```

```

<div class="popup-form">
  <form id="purchase-form" action="">
    <label for="name">Name:</label>
    <input type="text" id="name" name="name"><br><br>
    <label for="phone">Phone Number:</label>
    <input type="text" id="phone" name="phone"><br><br>
    <input type="submit" value="Submit">
  </form>
</div>

```

```

<script src="script.js"></script>
</body>
</html>

```

```

* {
  box-sizing: border-box;
}

```

```

body {
  margin: 0;
  font-family: Arial, sans-serif;
}

```

```

.container {
  max-width: 960px;
  margin: 0 auto;
  padding: 20px;
}

```

```
.header {
  text-align: center;
  margin-bottom: 20px;
}
```

```
.domains {
  display: grid;
  gap: 20px;
  grid-template-columns: repeat(5, 1fr);
}
```

```
.domain {
  display: flex;
  flex-direction: column;
  align-items: center;
}
```

```
.domain-box {
  width: 100%;
  padding: 10px;
  border: 1px solid #ccc;
  border-radius: 5px;
  text-align: center;
  cursor: pointer;
  transition: all 0.3s ease-in-out;
}
```

```
.domain-box:hover {
```

```
transform: scale(1.1);
}
```

```
.lerom-epson {
  text-align: center;
  margin: 20px 0;
  font-weight: bold;
  font-size: 20px;
}
```

```
.footer {
  display: flex;
  justify-content: center;
  margin-top: 20px;
}
```

```
.purchase-btn,
.consultation-btn {
  margin: 0 10px;
  padding: 10px 20px;
  font-size: 16px;
  cursor: pointer;
  background-color: #337ab7;
  color: #fff;
  border: none;
  border-radius: 5px;
  transition: all 0.3s ease-in-out;
}
```

```
.purchase-btn:hover,
.consultation-btn:hover {
    background-color: #23527c;
}
```

```
.popup-form {
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background-color: rgba(0, 0, 0, 0.5);
    display: flex;
    align-items: center;
    justify-content: center;
    z-index: 999;
    visibility: hidden;
    opacity: 0;
    transition: all 0.3s ease-in-out;
}
```

```
.popup-form.show {
    visibility: visible;
    opacity: 1;
}
```

```
#purchase-form {
    background-color: #fff;
```

```
padding: 20px;
border-radius: 5px;
}
```

```
#purchase-form label,
#purchase-form input {
display: block;
margin-bottom: 10px;
}
```

```
#purchase-form input[type="submit"] {
background-color: #337ab7;
color: #fff;
border: none;
padding: 10px 20px;
font-size: 16px;
cursor: pointer;
border-radius: 5px;
transition: all 0.3s ease-in-out;
}
```

```
#purchase-form input[type="submit"]:hover {
background-color: #23527c;
}
```

```
/* Media Queries */
```

```
@media screen and (max-width: 768px) {
```

```

.domains {
  grid-template-columns: repeat(3, 1fr);
}

@media screen and (max-width: 480px) {
  .domains {
    grid-template-columns: repeat(2, 1fr);
  }

  .lerom-epson {
    font-size: 16px;
  }
}

/* Animations */

@keyframes modalAnim {
  from {
    transform: scale(0);
  }
  to {
    transform: scale(1);
  }
}

@keyframes closeAnim {
  from {

```

```

        right: -50px;
    }
    to {
        right: 20px;
    }
import java.util.Arrays;

public class Method2DArrayGrades {

    public static void main(String[] args) {
        int[][] array = {{86, 90}, {90, 86}, {88, 84}, {92, 88}, {88, 88}};

        System.out.println("The 2d Array values are:");
        for (int i = 0; i < 5; i++) {
            for (int j = 0; j < 2; j++) {
                System.out.println(array[i][j]);
            }
        }

        int sum = getSum(array);
        int average = getAverage(array);
        int highest = getHighestValue(array);
        int lowest = getLowestValue(array);

        System.out.println("The sum is: " + sum);
        System.out.println("The average is: " + average);
        System.out.println("The highest value is: " + highest);
        System.out.println("The lowest value is: " + lowest);
    }
}

```

```
}
```

```
public static int getSum(int[][] array) {
```

```
    int sum = 0;
```

```
    for (int i = 0; i < 5; i++) {
```

```
        for (int j = 0; j < 2; j++) {
```

```
            sum += array[i][j];
```

```
        }
```

```
    }
```

```
    return sum;
```

```
}
```

```
public static int getAverage(int[][] array) {
```

```
    int sum = getSum(array);
```

```
    int average = sum / 10;
```

```
    return average;
```

```
}
```

```
public static int getHighestValue(int[][] array) {
```

```
    int[] flatArray = new int[10];
```

```
    int count = 0;
```

```
    for (int i = 0; i < 5; i++) {
```

```
        for (int j = 0; j < 2; j++) {
```

```
            flatArray[count] = array[i][j];
```

```
            count++;
```

```
}  
}
```

```
Arrays.sort(flatArray);
```

```
return flatArray[9];  
}
```

```
public static int getLowestValue(int[][] array) {  
    int[] flatArray = new int[10];
```

```
    int count = 0;  
    for (int i = 0; i < 5; i++) {  
        for (int j = 0; j < 2; j++) {  
            flatArray[count] = array[i][j];  
            count++;  
        }  
    }
```

```
Arrays.sort(flatArray);
```

```
return flatArray[0];  
}
```