

Міністерство освіти і науки України
Університет митної справи та фінансів

Факультет інноваційних технологій
Кафедра комп'ютерних наук та інженерії програмного забезпечення

Кваліфікаційна робота бакалавра

на тему: "Розробка веб-сайту для кадрового бюро"

Виконав: студентка групи ІІЗ20-2

Спеціальність 121 «Інженерія програмного
забезпечення»

Дитюк Владислава Віталіївна

(прізвище та ініціали)

Керівник к.т.н., доцент Чупілко Т.А.

(науковий ступінь, вчене звання, прізвище та ініціали)

Рецензент Університет митної справи та фінансів,

(місце роботи)

доцент кафедри кібербезпеки та інформаційних
технологій

(посада)

к.т.н. Савченко Ю.В.

Дніпро – 2024

АНОТАЦІЯ

Дитюк В.В. Розробка веб-сайту для кадрового бюро.

Кваліфікаційна робота на здобуття освітнього ступеня бакалавр за спеціальністю 121 «Інженерія програмного забезпечення». – Університет митної справи та фінансів, Дніпро, 2024.

Дана кваліфікаційна робота присвячена розробці веб-сайту для кадрового бюро, який розроблений в текстовому редакторі Visual Studio Code (VS Code) на мові програмування HTML/CSS та JavaScript. Було розглянуто методи та технічні засоби, необхідні для реалізації сайту, сформульовані вимоги до програмного забезпечення. Дизайн сайту є сучасним та інтерактивним.

Даний веб-сайт був розроблений для поліпшення та полегшення роботи кадрового бюро та управління людськими ресурсами. Розробка веб-сайту є актуальною в сучасному світі, оскільки Інтернет став ключовим джерелом інформації, засобом спілкування та ведення бізнесу. Отримані результати мають практичне значення, оскільки сприяють автоматизації та поліпшенню швидкості роботи підприємства та пошуку роботи клієнтам. Розроблений веб-сайт є надійним та ефективним.

Статичні сайти на HTML, CSS та JavaScript мають свої обмеження, але є простим і швидким рішенням для багатьох завдань. Вони підходять для проєктів з простими вимогами до функціональності та контенту. Розроблений сайт має інтерактивні елементи, анімації та ефекти за допомогою CSS та JavaScript, інтеграцію з API для отримання даних без необхідності використовувати серверний бекенд, використання сторонніх сервісів для додаткових функцій (наприклад, форми з Google Forms, коментарі з Disqus)..

Ключові слова: кадрове бюро, фірма, обслуговування, Visual Studio Code (VS Code), HTML, CSS, JavaScript.

ABSTRACT

Dityuk V.V. Development of a website for a personnel office.

Qualification work for obtaining a bachelor's degree in specialty 121 "Software engineering". – University of Customs and Finance, Dnipro, 2024.

This qualifying work is devoted to the development of a website for a personnel office, which is developed in the text editor Visual Studio Code (VS Code) in the programming language HTML/CSS and JavaScript. The methods and technical means necessary for the implementation of the site were considered, and the requirements for the software were formulated. The design of the site is modern and interactive.

This website was developed to improve and facilitate the work of the personnel office and human resources management. Website development is relevant in today's world, as the Internet has become a key source of information, a means of communication and doing business. The obtained results are of practical importance, as they contribute to the automation and improvement of the speed of the enterprise's work and job search for clients. The developed website is reliable and effective.

Static HTML, CSS, and JavaScript sites have their limitations, but are a simple and fast solution for many tasks. They are suitable for projects with simple functionality and content requirements. The developed site has interactive elements, animations and effects using CSS and JavaScript, integration with API to receive data without the need to use a server backend, use of third-party services for additional functions (for example, forms from Google Forms, comments from Disqus).

Keywords: personnel office, company, service, Visual Studio Code (VS Code), HTML, CSS, JavaScript.

ЗМІСТ

ВСТУП.....	5
РОЗДІЛ 1 ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ. ПОСТАНОВКА ЗАВДАННЯ ДОСЛІДЖЕННЯ.....	9
1.1. Аналіз публікацій щодо розробки веб-сайту для кадрового бюро	9
1.2. Аналіз методів розробки сайту для кадрового бюро	14
1.3. Висновки до першого розділу	25
РОЗДІЛ 2 АНАЛІЗ ЗАСОБІВ РЕАЛІЗАЦІЇ ВЕБ-САЙТУ ДЛЯ КАДРОВОГО БЮРО.....	27
2.1. Вибір програмних засобів для реалізації проекту	27
2.2. Засоби для реалізації веб-сайту для кадрового бюро	36
2.3. Висновки до другого розділу	41
РОЗДІЛ 3 РОЗРОБКА ВЕБ-САЙТУ ДЛЯ КАДРОВОГО БЮРО В ТЕКСТОВОМУ РЕДАКТОРІ VISUAL STUDIO CODE (VS CODE).....	43
3.1. Розробка загальної архітектури веб-сайту для кадрового бюро в текстовому редакторі Visual Studio Code (VS Code).....	43
3.2. Тестування роботи системи	49
3.3. Висновки до третього розділу	54
ВИСНОВКИ.....	56
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	58
ДОДАТОК А	60

ВСТУП

Актуальність дослідження. У сучасному світі потрібно йти нога в ногу зі змінами ринку праці та різноманітними тенденціями, які вимагають від працівників бути конкурентоспроможними, відтак важливо враховувати свої навички та здібності.

Люди є важливою опорою підприємства. Компанії повинні це враховувати, адже їхні навички, ефективність і відданість важливі не тільки для їх розвитку, але й для дій кожного. Тому важливо правильно керувати людьми. Практично в кожній компанії є відділ кадрів (HR-відділ), який є своєрідною ланкою між керівництвом і співробітниками.

Відділ кадрів відповідає не лише за підбір кандидатів на роботу. У завдання його працівників також входить відповідна мотивація працівників до виконання службових обов'язків, підготовка та проведення різноманітних тренінгів, визнання людей, які є надзвичайно цінними для колективу, і тих, хто гальмує його роботу і підлягає звільненню.

Інтегроване середовище Visual Studio Code, яке використовується для створення веб-додатків, пропонує широкий спектр інструментів та розширень, що дозволяють реалізувати сучасний та функціональний веб-сайт для кадрового бюро. Такий сайт може стати потужним інструментом для оптимізації процесів управління кадрами та сприяти підвищенню ефективності рекрутингових процесів, а також розширенню можливостей для професійного зростання працівників.

Метою роботи є створення сучасного та зручного веб-сайту для кадрового бюро в текстовому редакторі Visual Studio Code.

Методи дослідження: метод теорії інформації, обробка та аналіз інформації, методи проектування та розробки програмного забезпечення, методи проектування баз даних.

У відповідності до поставленої мети в кваліфікаційній роботі ставились та вирішувались наступні завдання дослідження:

1. Провести аналіз існуючих рішень та підходів для розробки сайту, виявити необхідні компоненти, переваги та недоліки;
2. Взяти за основу вимоги клієнтів та з їх допомогою структурувати функції майбутнього сайту;
3. Розробити дизайн і взаємодію, вивчення користувацького досвіду (UX) та взаємодії з сайтом для забезпечення зручності користування;
4. Визначити технічні аспекти розробки, таких як вибір платформи, мов програмування, хостингу, забезпечити високу продуктивність та безпеку сайту.
5. Визначити стратегії просування та рекламування сайту;
6. Розробити план тестування для перевірки роботи сайту на різних пристроях та браузерах, забезпечити високу якість веб-сайту через тестування взаємодії та виправлення помилок.

Об'єктом дослідження є процес розробки веб-сайтів для кадрових бюро.

Предметом дослідження є програмний комплекс, реалізований у вигляді веб-сайту для кадрового бюро, розроблений в текстовому редакторі Visual Studio Code.

Практичне значення одержаних результатів – покращити та автоматизувати роботу кадрового бюро за рахунок відділу кадрів.

Структура роботи:

Розділ 1 Дослідження предметної області з обґрунтуванням актуальності. Постановка завдань дослідження. У даному розділі було виконано пошук та аналіз публікацій стосовно теми даного веб-сайту та методи для розробки бази даних.

Розділ 2 Аналіз засобів реалізації підтримки прийняття рішень. В даному розділі було проаналізовано та обрано технології для реалізації веб-сайту.

Розділ 3 Розробка веб-сайту. В даному розділі було спроектовано та розроблено професійний сайт для кадрового бюро.

В результаті виконання кваліфікаційної роботи було набуто фахові компетентності та підтверджені наступні результати навчання, які відповідають

Освітньо-Професійній програмі за спеціальністю 121«Інженерія програмного забезпечення»:

K01. Здатність до абстрактного мислення, аналізу та синтезу.

K02. Здатність застосовувати знання у практичних ситуаціях.

K03. Здатність спілкуватися державною мовою як усно, так і письмово.

K05. Здатність вчитися і оволодівати сучасними знаннями.

K06. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

K13. Здатність ідентифікувати, класифікувати та формулювати вимоги до програмного забезпечення.

K14. Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування.

K15. Здатність розробляти архітектури, модулі та компоненти програмних систем.

K16. Здатність формулювати та забезпечувати вимоги щодо якості програмного забезпечення у відповідності з вимогами замовника, технічним завданням та стандартами.

K19. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних.

K22. Здатність накопичувати, обробляти та систематизувати професійні знання щодо створення і супроводження програмного забезпечення та визнання важливості навчання протягом всього життя.

K26. Здатність до алгоритмічного та логічного мислення.

ПР01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.

ПР05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.

ПР06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.

ПР08. Вміти розробляти людино-машинний інтерфейс.

ПР12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.

ПР13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.

ПР14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.

ПР21. Знати, аналізувати, вибирати, кваліфіковано застосовувати засоби забезпечення інформаційної безпеки (в тому числі кібербезпеки) і цілісності даних відповідно до розв'язуваних прикладних завдань та створюваних програмних систем.

РОЗДІЛ 1

ДОСЛІДЖЕННЯ ПРЕДМЕТНОЇ ОБЛАСТІ. ПОСТАНОВКА ЗАВДАННЯ ДОСЛІДЖЕННЯ

1.1. Аналіз публікацій щодо розробки веб-сайту для кадрового бюро

У сучасному світі наявність веб-сайту стає все більш необхідною. Користувачі очікують, що кожна компанія має свій веб-сайт, і якщо його немає, компанія виглядає менш надійною в їх очах. Отже, веб-сайт є ключовим елементом інтернет-маркетингу. Він має відповідати потребам сучасних користувачів.

Існує кілька ключових характеристик, якими має володіти зразковий корпоративний сайт. Багато з них залежать не лише від підрядника і проектувальника, але й від самих підприємців, які часто несуть відповідальність за численні помилки. Перш за все, сайт компанії повинен слугувати джерелом інформації для потенційних клієнтів.

У статті "Створення веб-додатку: основні аспекти ефективної розробки веб-сайту" [\[1\]](#) було проведено аналіз і оцінку існуючих веб-сайтів за 2023 рік. 3 серпня 2022 року в Інтернеті налічується майже 2 мільярди веб-сайтів і понад 8,9 мільйонів мобільних додатків, доступних для завантаження.

Згідно з прогнозами Бюро статистики праці (BLS) [\[2\]](#), очікується, що попит на веб-розробників у національному масштабі зросте на 23% з 2021 по 2031 рік. Цей темп зростання значно перевищує середній показник для всіх професій. Це зростання обумовлено стрімким розвитком інтернет-технологій і зростаючою потребою у створенні та підтримці сучасних, функціональних і безпечних веб-сайтів та веб-додатків. Оскільки все більше компаній переходять на цифрові платформи для ведення бізнесу, необхідність у кваліфікованих веб-розробниках стає все більш актуальною. Високий попит також підкріплюється зростанням мобільного інтернету та збільшенням кількості користувачів, які використовують

різні гаджети для доступу до веб-ресурсів. У зв'язку з цим, веб-розробка стає однією з найперспективніших та найдинамічніших галузей сучасної економіки.

Розмір світового ринку веб-розробки у 2022 році оцінювався в 60,7 мільйонів доларів, і, за прогнозами, він зросте на 8,08% протягом прогнозованого періоду, досягнувши 96,5 мільйонів доларів у 2028 році. Що стосується світового ринку програмного забезпечення для розробки додатків, його розмір становив 142,5 мільярдів доларів у 2021 році, і очікується, що він збільшиться до 1,160 мільярдів доларів у 2031 році, з середньорічним темпом зростання 23,5% з 2022 по 2031 рік [\[1\]](#).

Також у статті "Створення веб-додатку: основні аспекти ефективної розробки веб-сайту" [\[1\]](#) автори відзначають основні типи веб-додатків, такі як:

1. Статичний сайт - це економічний варіант для представлення портфоліо або статичних даних, іншими словами, це сайт без можливості взаємодії з користувачем. Час розробки такого сайту зазвичай становить приблизно від 190 до 260 годин.

2. Динамічний веб-додаток дозволяє змінювати, редагувати або видаляти вміст, який буде відображатися відповідно до пристрою та розташування користувача. Тривалість розробки такого додатку становить приблизно від 220 до 300 годин.

3. Веб-додаток електронної комерції включає у себе систему керування вмістом (CMS) разом з платіжними шлюзами, макетами, розробленими з урахуванням оптимізації для пошукових систем (SEO). Він також може включати push-сповіщення і чат-боти для покращення взаємодії з користувачами. Тривалість розробки такого додатку зазвичай становить від 350 до 700 годин.

4. Розробка веб-порталів, що призначені для бізнесу до бізнесу (B2B), бізнесу до споживача (B2C) або освітньої спільноти, вимагає значно більшого часу - від 600 до 1000 годин.

5. CMS веб-розробка робить керування вашим сайтом більш простим, не потребуючи постійного залучення розробників. Час розробки такого сайту

залежить від обраної системи управління вмістом і може становити від 100 до 300 годин.

У статті “Веб-сайт підприємства як ефективний інструмент маркетингових комунікацій в мережі інтернет” [3] автори Л.С. Васильченко, О.В. Якушев та С.В. Литвин проводять аналіз та характеристику різних видів веб-сайтів.

Згідно зі статтею, офіційної класифікації веб-сайтів не існує, однак на основі характеристик, на які можна поділити сучасні сайти, їх можна класифікувати так:

1. Сторінки-візитки: часто використовуються компаніями для представлення своїх послуг або продуктів у вигляді онлайн-брошури. Вони дозволяють компаніям ефективно комунікувати свої цінності та переваги своїм клієнтам і партнерам.

2. Лендінги, або односторінкові сайти, зазвичай створюються для рекламних кампаній або просування конкретного продукту чи послуги. Вони забезпечують детальний опис товару або послуги та створюють простий і зрозумілий маршрут для клієнтів, щоб вони могли зробити покупку або здійснити конверсію. Інтернет-магазини: включають каталог товарів і послуг, інформацію про компанію, можуть містити безліч сторінок з картками товарів, а також функції пошуку і фільтрації.

3. Веб-сервіси: платформами для взаємодії користувачів у різних галузях, таких як освіта, розваги, фінанси тощо. Вони надають різноманітні функції та можуть бути спрямовані на навчання, розваги, спілкування або роботу.

4. Пошукові системи: відіграють важливу роль у веб-серфінгу, допомагаючи користувачам знаходити необхідну інформацію в Інтернеті. Вони забезпечують швидкий та зручний доступ до різноманітних джерел інформації, що дозволяє користувачам ефективно виконувати пошукові запити (Google, Bing, Yahoo).

5. Поштові сервіси: надають користувачам можливість відправляти та отримувати електронні листи. Вони є важливим інструментом для бізнесу та

особистого спілкування, дозволяючи ефективно обмінюватися інформацією та спілкуватися з колегами, клієнтами та друзями.

6. Соціальні мережі: популярні онлайн-платформами для спілкування та обміну інформацією між користувачами. Вони дозволяють створювати профілі, додавати друзів, публікувати повідомлення та фотографії, а також взаємодіяти з іншими користувачами. Некомерційні сайти: сторінки благодійних організацій, особисті блоги, тематичні веб-ресурси.

7. Відеохостинги: платформами для розміщення та перегляду відео контенту в Інтернеті. Вони надають можливість користувачам ділитися відео зі світом та споживати цікавий та розважальний контент. (YouTube, Vimeo).

Варто зауважити, що деякі веб-сайти можуть поєднувати кілька елементів із цього списку. Наприклад, сайт з онлайн-курсами може виконувати роль інтернет-магазину, пропонувати лендинги для своїх продуктів, а також містити блог із корисними статтями для користувачів.

Зазвичай маркетологи використовують аналіз трафіку для визначення ефективності джерел, які приводять відвідувачів на сайт компанії. Цей аналіз дозволяє передбачити поведінку відвідувачів та оптимізувати шляхи конверсії, враховуючи особливості цільової аудиторії. Такий підхід дозволяє виявити найбільш ефективні рекламні стратегії для просування сайту.

Маркетологи виділяють кілька видів джерел трафіку через різні канали, такі як:

1. Пошуковий трафік, або органічний, є особливо цінним, оскільки він відображає природну видачу сайту за ключовими словами. Оптимізація під пошукові системи (SEO) допомагає збільшити цей вид трафіку, що сприяє покращенню позицій сайту у пошукових системах і збільшенню конверсії.

2. Платний трафік включає відвідувачів, які переходять на сайт через різні види реклами, такі як контекстна, таргетована, медійна реклама та інші. Цей вид трафіку може бути швидко запусканий, але його вартість може бути високою.

3. Прямий трафік складається з відвідувачів, які переходять на сайт безпосередньо, введенням URL або використанням закладок. Це важливий показник лояльності аудиторії і впізнаваності бренду.

4. Реферальний трафік приходить зовнішніх сайтів-донорів, а соціальний трафік — з соціальних мереж. Обидва ці джерела можуть бути важливими для привертання аудиторії, але їх успішність залежить від активності маркетингових комунікацій у відповідних каналах.

Також важливо зрозуміти, що веб-сайти відрізняються не лише за їхнім змістом, але й за такими характеристиками, як кількість сторінок, використання інтерактивних елементів, наявність інструментів для взаємодії з користувачами тощо. Наприклад, інформаційний портал Вікіпедія дозволяє кожному користувачеві редагувати зміст сторінок завдяки спеціальному функціоналу ресурсу.

Основними компонентами веб-сайту є текст, зображення, відео та аудіо. Також елементи можуть бути інтерактивними. Раніше для взаємодії з користувачами використовувалися flash-елементи, а тепер частіше за все використовуються різноманітні плагіни для CMS (Content Management System) — систем керування вмістом веб-сайтів [3].

У свою чергу, можна відмітити такі навички для у веб-розробці [4]:

8. Front-end розробники фокусуються на створенні користувацького інтерфейсу, щоб зробити його привабливим та зручним для використання.

9. Back-end розробники відповідають за обробку даних на сервері та забезпечують функціональність веб-додатків.

10. Full-stack розробники мають навички як у front-end, так і в back-end розробці, що дозволяє їм керувати усім процесом веб-розробки.

11. Веб-дизайнери концентруються на естетиці та макеті веб-сторінок, співпрацюючи з розробниками для досягнення бажаного вигляду.

Планування та розгляд при створенні веб-програми є критичними етапами, що визначають подальший успіх проекту. Вони вимагають уважного аналізу та

вивчення різних аспектів, починаючи від визначення цілей і закінчуючи вибором відповідних інструментів і фреймворків.

Першим кроком у плануванні є чітке визначення цілей проекту. Важливо розуміти, яку проблему веб-додаток повинен вирішити та які потреби користувачів він має задовольняти. Це визначає його функціональність, дизайн і архітектуру.

Після цього слід провести дослідження ринку та аналіз конкурентів, щоб зрозуміти, що вже існує на ринку, які переваги мають конкуренти і як можна покращити свій продукт.

Далі вибирається найбільш відповідний технологічний стек для веб-додатка. Це включає в себе вибір мов програмування, баз даних, фреймворків і інших інструментів, які будуть використовуватися для розробки.

Одним з ключових аспектів є також вивчення останніх тенденцій та технологій у сфері веб-розробки. Швидкі зміни в цьому сегменті галузі вимагають постійного оновлення знань та використання передових практик.

На останньому етапі планування визначаються стратегії тестування і впровадження, а також план монетизації (якщо це необхідно).

Враховуючи всі ці аспекти і критерії, розробники можуть створити веб-додаток, який відповідає потребам бізнесу і забезпечить задоволення його користувачів.

1.2. Аналіз методів розробки сайту для кадрового бюро

Розробка веб-сайту для кадрового бюро потребує ретельного вибору методів та технологій, щоб забезпечити високу якість, ефективність та зручність використання. У цьому розділі розглянуто різні підходи та інструменти, які можуть бути використані для створення такого сайту.

Першим кроком у розробці веб-сайту є ретельний аналіз існуючих рішень та методів, щоб визначити найбільш ефективний підхід для реалізації проекту.

Це включає вивчення різних технологій, інструментів та фреймворків, які можуть бути використані для створення сайту. Наприклад:

1. Використання бібліотек та фреймворків: Використання готових бібліотек та фреймворків, таких як Django для Python або Express для Node.js, значно спрощує розробку серверної частини. Вони забезпечують швидку інтеграцію необхідної функціональності, що дозволяє зосередитися на бізнес-логіці та унікальних вимогах проекту.

2. Інструменти автоматизації: Інструменти автоматизації, такі як Selenium, дозволяють автоматизувати тестування та взаємодію з веб-інтерфейсами. Це допомагає створювати скрипти для імітації дій користувачів, забезпечуючи стабільність та надійність роботи сайту.

3. Використання API: Використання API для доступу до баз даних кандидатів та вакансій надає потужні можливості для автоматизації обробки інформації та управління заявками. Наприклад, створення проекту у Google Cloud Console, активація API та налаштування OAuth 2.0 для авторизації користувачів забезпечує надійний та безпечний доступ до даних.

4. Вибір мови програмування: Вибір мови програмування залежить від досвіду розробника, потреб проекту та технічних вимог. Серед популярних мов для створення веб-сайтів можна виділити Python, JavaScript (Node.js), Java, C#, PHP, Ruby та Go. Кожна з них має свої переваги та бібліотеки для інтеграції з різними API та автоматизації процесів.

5. Використання Docker: Для розробки програми доцільно використовувати Docker, який забезпечує ізоляцію середовища, спрощує розгортання та масштабування додатків, а також полегшує управління залежностями та конфігураціями. Docker дозволяє швидко створювати, тестувати та розгортати додатки в стандартизованих контейнерах.

6. Використання React.js: Для створення сайту з сучасним та інтерактивним інтерфейсом пропонується використовувати React.js, популярний фронтенд-фреймворк для веб-додатків. React.js забезпечує простоту у вивченні та застосуванні, велике спільнотне екосередовище та стабільний код.

7. Генерація тестових даних: Для тестування програми потрібні тестові дані, які можна швидко створити за допомогою спеціальних онлайн-інструментів для генерації фіктивних даних. Це дозволяє ефективно перевірити функціональність сайту та забезпечити його відповідність вимогам користувачів.

Також веб-сайти можна розробляти на різних операційних системах, таких як Windows, macOS та різні дистрибутиви Linux. Наприклад:

1. Windows: Windows - це популярна операційна система зі зручним інтерфейсом та багатим набором розробницьких інструментів. Вона підходить для більшості веб-розробників, особливо якщо вони звикли до середовища Windows або використовують інструменти, що працюють тільки під цю ОС.

2. macOS: macOS - це операційна система, яка популярна серед розробників, особливо серед тих, хто працює з продукцією Apple, такою як iPhone або iPad. Вона також підтримує багато інструментів, що використовуються в розробці веб-сайтів.

3. Linux: Linux - це операційна система з відкритим вихідним кодом, яка часто використовується для серверного розвитку. Вона є популярним вибором для веб-розробників, оскільки вона надає доступ до широкого спектру інструментів та бібліотек, а також є дуже стабільною і безпечною.

Загалом, кращий вибір операційної системи залежить від вашої особистої вподобаності та досвіду в роботі з нею.

На платформах Windows, macOS є декілька популярних інтегрованих середовищ розробки (IDE) та редакторів коду, які можна використовувати для розробки веб-сайтів:

1. Visual Studio: Це потужне інтегроване середовище розробки, яке підтримує розробку веб-сайтів на різних мовах програмування, таких як HTML, CSS, JavaScript, ASP.NET та багатьох інших. Visual Studio має різні версії, включаючи безкоштовну версію Community.

2. Visual Studio Code: Це легкий, але потужний редактор коду, який підтримує багато мов програмування, включаючи HTML, CSS, JavaScript, а також плагіни для роботи з різними фреймворками, такими як React, Angular та Vue.js.

3. Sublime Text: Це швидкий та легкий редактор коду, який також підтримує розробку веб-сайтів. Він має велику кількість плагінів для полегшення роботи з HTML, CSS та JavaScript.

4. Atom: Це ще один відкритий редактор коду, який підтримує розробку веб-сайтів. Atom має багато розширень, які допомагають полегшити роботу з різними мовами програмування та фреймворками.

5. Brackets: Це редактор коду, розроблений Adobe, який спеціалізується на розробці веб-сайтів. Він має вбудовані інструменти для роботи з HTML, CSS та JavaScript, а також плагіни для роботи з іншими мовами та фреймворками.

Для розробки веб-сайтів найбільш поширеним інструментом є Visual Studio Code. Він відомий своєю швидкістю, розширюваністю та підтримкою великої кількості мов програмування та технологій веб-розробки.

Ось кілька причин, чому Visual Studio Code так популярний серед веб-розробників:

1. Багатофункціональність: VS Code має широкі можливості, такі як автодоповнення коду, інтеграція з системами контролю версій, вбудована термінал, налаштованість та багато іншого.

2. Розширюваність: Є можливість розширити функціональність VS Code за допомогою розширень, які дозволяють вам додавати нові функції, мови програмування, теми тощо.

3. Підтримка мов програмування та фреймворків: VS Code підтримує широкий спектр мов програмування, включаючи HTML, CSS, JavaScript, TypeScript та багато інших. Він також має розширення для роботи з популярними веб-фреймворками, такими як React, Angular, і Vue.js.

4. Кросплатформенність: VS Code доступний для Windows, macOS та Linux, що робить його доступним для розробки на будь-якій платформі.

5. Окрім відкриття репозиторіїв, розгалужень і запитів на отримання від постачальників засобів керування джерелами, таких як GitHub і Azure Repos, VS Code надає можливість працювати з кодом, який зберігається на локальній машині.

Загалом, Visual Studio Code є потужним та універсальним інструментом для розробки веб-сайтів, який використовується мільйонами розробників по всьому світу.

Вибір мови програмування для розробки веб-сайту залежить від багатьох факторів. Однак, деякі мови програмування виявляються особливо популярними та ефективними у веб-розробці. Ось деякі з них:

1. JavaScript: JavaScript є однією з найпоширеніших мов програмування для веб-розробки. Вона використовується для розробки фронтенду (за допомогою фреймворків і бібліотек, таких як React, Angular або Vue.js) та бекенду (за допомогою Node.js). JavaScript також використовується для реалізації динамічного контенту, взаємодії з користувачем та анімації.

2. Python: Python - це високорівнева, динамічна мова програмування, яка використовується для розробки веб-додатків, особливо у великих компаніях і стартапах. Фреймворки, такі як Django і Flask, роблять Python популярним для розробки веб-додатків.

3. HTML/CSS: Хоча HTML і CSS не є мовами програмування в суворому розумінні, вони є основними технологіями для створення веб-сторінок і визначення їх вигляду та макету.

4. PHP: PHP - це серверна мова програмування, яка використовується для розробки динамічних веб-сайтів та веб-додатків. Фреймворки, такі як Laravel і Symfony, допомагають розробникам швидше створювати складні веб-додатки.

5. Java: Java застосовується для розробки веб-додатків за допомогою фреймворків, таких як Spring та JavaServer Faces (JSF). Java також використовується для створення веб-служб та бекенду великих корпоративних систем.

Важливо також вивчити особливості кожної мови та їх використання у веб-розробці перед тим, як приймати остаточне рішення.

Веб-сайт має бути простим у використанні. Наша планета переповнена мережею з понад мільярдом веб-сайтів, доступних для відвідування потенційними користувачами. Ось чому особливо важливо, щоб дійсно хороший

дизайн був простим у використанні. У той же час безпосередній досвід користувача повинен бути позитивним. Відвідування веб-сайту має бути приємним і, залежно від його змісту, можливо, навіть веселим.

З точки зору зручності використання та взаємодії з користувачем, простота — найкраще рішення. Цей аспект можна реалізувати різними способами. Ось кілька прикладів:

1. Кольори: Довідник із взаємодії між комп'ютером і людиною рекомендує використовувати максимум п'ять (плюс/мінус два) різних кольорів у дизайні веб-сайту.

2. Шрифти: Шрифти веб-сайті мають бути легкими для читання, і, як і з кольорами, не повинно використовуватися занадто багато різних шрифтів. Зазвичай рекомендується використовувати максимум три різні шрифти максимум трьох різних розмірів.

3. Графіка: графіку слід використовувати, лише якщо вона допомагає користувачам виконати певну дію або якщо вона має спеціальну функцію.

Принцип візуальної ієрархії тісно пов'язаний з принципом простоти. Він включає в себе розташування та організацію елементів веб-сайту таким чином, щоб природним, інтуїтивно зрозумілим способом направляти відвідувачів веб-сайту до його найважливіших елементів. Коли справа доходить до оптимізації зручності використання та позитивного досвіду користувача, мета полягає в тому, щоб змусити людей виконувати певні дії у спосіб, який їм здається природним і зручним. Наприклад, регулюючи положення, колір або розмір певних елементів, можна структурувати свою сторінку таким чином, щоб відвідувачі природним чином привертали увагу до її найважливіших елементів.

Інтуїтивно зрозуміла навігація на сайті допомагає відвідувачам швидко знайти те, що вони шукають. В ідеалі відвідувачі відвідуватимуть сайт і не будуть думати, чи йти далі, куди та як. На веб-сайті має бути легко дістатися з пункту А в пункт Б. Ось декілька прикладів оптимізації навігації тексту:

1. Структура основної навігації має бути простою (бажано розміщувати її у верхній частині сторінки).

2. Нижній колонтитул веб-сайту також може містити функцію навігації.
3. На кожній сторінці, крім домашньої, потрібно використовувати навігаційні підказки, щоб повідомляти відвідувачам, де вони знаходяться та як вони туди потрапили.
4. У верхній частині сторінки повинна бути функція пошуку, щоб користувачі могли легко шукати терміни.
5. Забагато варіантів навігації є контрпродуктивним.
6. У більшості випадків найкраще, якщо навігація буде не глибше трьох рівнів.

Не лише навігація має бути узгодженою, але й загальний вигляд кожної підсторінки, яку відвідує користувач, має бути узгодженим. Фони, колірні схеми, шрифти, а також стиль у текстах – це аспекти, де єдиний дизайн позитивно впливає на зручність використання та досвід користувача. Сторінки не повинні мати однаковий макет. Потрібно створити різні макети для конкретних типів сторінок (наприклад, інший для цільових сторінок, трохи інший для інформаційних сторінок тощо). Однак потрібно використовувати ці макети послідовно, щоб відвідувачі могли легко бачити, яка інформація їм надається.

Згідно з останніми дослідженнями американської компанії ComScore, використання планшетів для ведення онлайн-бізнесу в США зросло на 30%. Використання Інтернету зі смартфонів за той же період зросло на 78%. Щоб веб-сайт був зручним для користувачів, він має бути сумісним з різними пристроями (їхніми операційними системами та браузерами), якими користуються користувачі. У ширшому розумінні це означає інвестиції в дуже гнучку структуру веб-сайту, таку як адаптивний дизайн і стандарти розробки веб-сайтів. На адаптивному веб-сайті вміст і його формат автоматично адаптуються до розміру пристрою, який використовує користувач.

Зрештою, важливіше щоб сайт забезпечував задовольняючу взаємодію з користувачами на різних платформах, а не щоб він виглядав однаково на всіх платформах.

Існують певні правила веб-дизайну, які з роками стали знайомими для більшості користувачів Інтернету. До них належать, наприклад:

- 1) Навігація на головній сторінці вгору або вліво.
- 2) Логотип у верхньому лівому куті (або в центрі) сторінки.
- 3) Логотип, який можна натиснути, повертає користувача на початкову сторінку.
- 4) Посилання, які змінюють колір або вигляд, коли ви наводите на них вказівник миші.

Конвенції веб-дизайну в елементах/стратегії дизайну, які вже знайомі відвідувачам, завжди можуть допомогти. Щоб надати сайту більше довіри. Один із найкращих способів підвищити довіру до свого веб-сайту – описати продукт або послугу надійно та зрозуміло.

Важливим елементом, який повинен містити веб-сайт, є сертифікат SSL [\[5\]](#).

Аналіз методів розробки сайту для кадрового бюро може включати різноманітні аспекти, спрямовані на забезпечення ефективної присутності в Інтернеті та задоволення потреб цільової аудиторії. Ключові методи, які потрібно врахувати:

1. Аналіз вимог клієнтів: Ретельне вивчення потреб цільової аудиторії кадрового бюро. Важливо враховувати їхні очікування та потреби в онлайн-послугах. Це включає проведення опитувань, інтерв'ю та аналіз відгуків клієнтів для розуміння їхніх потреб. Результати цього аналізу допоможуть визначити пріоритети функціональності та дизайну сайту.

2. Створення зручного інтерфейсу користувача (UI): Розробка зручного інтерфейсу, який дозволяє користувачам легко знаходити інформацію, реєструватися та використовувати послуги бюро. Це передбачає використання сучасних принципів UX/UI дизайну, таких як мінімалістичний дизайн, чітка навігація, швидкий доступ до ключової інформації та адаптивний дизайн для різних пристроїв.

3. Розробка функціональності: Визначення необхідних функцій, таких як пошук вакансій, подача заявок на роботу, перегляд профілів кандидатів,

інтерактивні форми для зв'язку тощо. Крім того, важливо передбачити можливості для реєстрації та створення профілю користувачів, управління резюме, сповіщення про нові вакансії, інтеграцію з соціальними мережами та платформами професійних спільнот.

4. Оптимізація для пошукових систем (SEO): Впевнене позиціонування веб-сайту в пошукових системах для забезпечення високого рівня видимості та привертання цільового трафіку. Це включає оптимізацію контенту для ключових слів, поліпшення швидкості завантаження сторінок, створення мета-тегів та описів, розробку XML-карти сайту, використання структурованих даних для покращення видимості у пошукових системах.

5. Мобільна сумісність: Розробка сайту з урахуванням мобільних пристроїв, оскільки все більше користувачів використовують смартфони та планшети для доступу в Інтернет. Це включає адаптивний дизайн, який автоматично підлаштовується під різні розміри екранів, та оптимізацію швидкості завантаження для мобільних пристроїв.

6. Аналіз конкурентів: Вивчення сайтів конкурентів для ідентифікації їхніх сильних та слабких сторін, а також можливостей для покращення власного веб-проекту. Це допоможе визначити унікальні продажні пропозиції (USP) і створити сайт, який буде вигідно відрізнятися від конкурентів. Аналіз конкурентів також може надати ідеї для впровадження нових функцій і сервісів, які будуть привабливими для користувачів.

7. Забезпечення безпеки: Застосування заходів забезпечення безпеки для захисту конфіденційної інформації користувачів, такої як особисті дані та резюме. Це включає використання SSL-сертифікатів для шифрування даних, регулярні оновлення програмного забезпечення, впровадження політик безпеки, захист від DDoS-атак та інших загроз, а також навчання персоналу щодо безпеки даних.

8. Аналітика та відстеження результатів: Використання аналітичних інструментів для відстеження та аналізу поведінки користувачів, що дозволяє вдосконалювати функціонал сайту та стратегії маркетингу. Інструменти на

кшталт Google Analytics, Hotjar або Mixpanel можуть надавати цінну інформацію про відвідуваність сайту, шляхи користувачів, коефіцієнти конверсії та інші метрики, що допоможуть у прийнятті обґрунтованих рішень для покращення сайту.

9. Соціальні мережі та маркетинг: Використання соціальних мереж для просування сайту та взаємодії з користувачами. Важливо створити профілі на основних платформах (Facebook, LinkedIn, Twitter) та регулярно публікувати контент, що сприятиме залученню нових користувачів та підвищенню лояльності існуючих.

Ці методи допоможуть створити веб-застосунок кадрового бюро, який відповідає сучасним вимогам та потребам користувачів.

Також для більшої зручності можна користуватися інструментами для розробки дизайну веб-сайтів. Використання інструментів для розробки дизайну веб-сайтів може значно полегшити процес розробки та допомогти створити професійний та ефективний дизайн для веб-проекту.

Існує безліч інструментів для розробки дизайну веб-сайтів. Ось декілька популярних інструментів, які можна використовувати:

1. Adobe XD: Adobe XD є потужним інструментом для дизайну веб-сайтів та мобільних додатків, що має вбудовані функції прототипування та спільної роботи.

2. Sketch: Sketch - це інструмент дизайну, який особливо популярний серед дизайнерів веб-сайтів та мобільних додатків на macOS.

3. Figma: Figma - це веб-платформа для дизайну, яка дозволяє створювати та редагувати дизайн в реальному часі та спільно працювати над проектами [\[15\]](#).

4. Adobe Photoshop: Photoshop є класичним інструментом для дизайну, який використовується для створення графіки та макетів веб-сайтів.

5. Canva: Canva - це інструмент для дизайну з багатою колекцією шаблонів, ідеальний для швидкого створення простих дизайнів веб-сайтів.

6. Adobe Illustrator: Illustrator є іншим інструментом Adobe, призначеним для створення векторної графіки, і часто використовується для створення елементів дизайну веб-сайтів.

Користування інструментами для розробки дизайну веб-сайтів має безліч переваг, серед яких:

1. Швидкість і продуктивність: Інструменти для розробки дизайну, такі як Adobe XD, Sketch, Figma та інші, надають широкий набір функцій та інструментів, що дозволяє швидко створювати прототипи та макети веб-сайтів.

2. Прототипування: Багато інструментів дозволяють створювати прототипи веб-сайтів, що дозволяє вам протестувати функціональність та навігацію перед початком реального розробки.

3. Спільна робота: Багато інструментів мають можливість спільної роботи, що дозволяє дизайнерам працювати разом над проектами в реальному часі та обмінюватися коментарями та відгуками.

4. Висока якість: Завдяки потужним інструментам і можливостям редагування, ви можете створювати дизайни високої якості з професійним виглядом.

5. Ефективність: Інструменти для розробки дизайну дозволяють швидко та ефективно вносити зміни, редагувати дизайн та працювати з різними елементами веб-сайту.

6. Шаблони і ресурси: Багато інструментів мають вбудовані шаблони та ресурси, що допомагають вам швидше створювати дизайни та заощаджувати час.

7. Адаптивність та респонсивний дизайн: Багато інструментів надають можливості для створення адаптивних і респонсивних дизайнів, які оптимізовані для різних типів пристроїв і розмірів екранів. Це важливо для забезпечення зручного перегляду веб-сайту на будь-яких пристроях, включаючи комп'ютери, планшети та смартфони.

8. Адаптивність та респонсивний дизайн: Багато інструментів надають можливості для створення адаптивних і респонсивних дизайнів, які оптимізовані для різних типів пристроїв і розмірів екранів. Це важливо для забезпечення

зручного перегляду веб-сайту на будь-яких пристроях, включаючи комп'ютери, планшети та смартфони.

9. Аналітика та відстеження: Деякі інструменти можуть інтегруватися з аналітичними сервісами, що дозволяє відстежувати поведінку користувачів на веб-сайті, збирати дані про взаємодію та використання, а також здійснювати аналіз та оптимізацію сайту на основі цих даних.

Загалом, створення ефективного веб-сайту для кадрового бюро вимагає уважного підходу до вибору та інтеграції різних методів, технологій та інструментів. Оптимальна архітектура, тестування, оптимізація, забезпечення доступності та локалізація гарантують створення веб-сайту, який буде корисним, ефективним та зручним для всіх зацікавлених сторін.

1.3. Висновки до першого розділу

У першому розділі було проведено аналіз і дослідження сфери розробки веб-сайту для кадрового бюро. Розглянуто наявні публікації, методи та підходи до створення таких сайтів. Аналіз матеріалів показав зростаючий попит на веб-сайти для кадрових бюро, що підкреслює потребу в автоматизації та оптимізації кадрових процесів.

Для реалізації проекту пропонується використовувати сучасні технології, такі як HTML, CSS та JavaScript, для забезпечення функціональності та привабливого дизайну. Розробку веб-сайту запропоновано виконувати в текстовому редакторі Visual Studio Code, який надає зручні інструменти для написання та редагування коду. Visual Studio Code підтримує розширення для інтеграції з різними фреймворками та бібліотеками, що може значно прискорити процес розробки. Крім того, Visual Studio Code має вбудовані інструменти для відладки та тестування, що дозволяє швидко й ефективно виявляти та виправляти помилки в коді. Використання таких сучасних технологій і інструментів

сприятиме створенню високоякісного веб-сайту, який буде відповідати вимогам користувачів і забезпечить позитивний користувацький досвід.

Обраний підхід до розробки веб-сайту для кадрового бюро відіграє ключову роль у забезпеченні успішної реалізації проекту, враховуючи вимоги клієнтів і відповідність професійним стандартам. Ретельний аналіз факторів та ефективне поєднання різних технологій дозволить створити високоякісний веб-сайт, що сприятиме оптимізації та автоматизації процесів управління кадрами для кадрового бюро.

Окрім технічних аспектів, важливо приділити увагу дизайну та користувацькому досвіду (UX/UI). Створення інтуїтивно зрозумілого та естетично привабливого інтерфейсу допоможе залучити та утримати користувачів. Регулярне тестування з користувачами та збір відгуків допоможуть виявити можливі проблеми й покращити загальну якість продукту. Інструментом для роботи з дизайном було рекомендовано обрати хмарний багатоплатформовий сервіс Figma, який є потужним інструментом для створення макетів та прототипів веб-сайтів. Вона дозволяє дизайнеру працювати над візуальними елементами та інтерактивністю одночасно, забезпечуючи зручний і ефективний робочий процес. У Figma можна створювати інтерфейси, які легко коригувати і тестувати, залучаючи користувачів та отримуючи зворотний зв'язок, що сприяє поліпшенню дизайну.

Правильний підхід до розробки дозволить створити ефективний інструмент для управління кадровими процесами, що сприятиме розвитку та успіху кадрового бюро.

РОЗДІЛ 2

АНАЛІЗ ЗАСОБІВ РЕАЛІЗАЦІЇ ВЕБ-САЙТУ ДЛЯ КАДРОВОГО БЮРО

2.1. Вибір програмних засобів для реалізації проекту

Всесвітня павутина (англ. World Wide Web) є глобальним інформаційним простором, що базується на фізичній інфраструктурі Інтернету та використовує протокол передачі даних HTTP. Вона викликала значний розрив в інформаційних технологіях та спричинила стрімкий розвиток Інтернету. Часто, коли говорять про Інтернет, мають на увазі саме Всесвітню павутину [\[7\]](#).

Всесвітня павутина складається з мільйонів веб-серверів в Інтернеті, які розташовані по всьому світу. Веб-сервер — це програма, що працює на комп'ютері, підключеному до мережі, і використовує протокол HTTP для передачі даних. У найпростішому випадку веб-сервер приймає HTTP-запит на певний ресурс, знаходить відповідний файл на локальному жорсткому диску і надсилає його по мережі комп'ютеру, що зробив запит. Складніші веб-сервери можуть динамічно генерувати ресурси у відповідь на HTTP-запити. Для ідентифікації ресурсів (часто файлів або їх частин) у Всесвітній павутині використовуються уніфіковані ідентифікатори ресурсів URI (англ. Uniform Resource Identifier). Для визначення місцезнаходження ресурсів у мережі використовуються уніфіковані локатори ресурсів URL (англ. Uniform Resource Locator). URL-локатори об'єднують технологію ідентифікації URI та систему доменних імен DNS (англ. Domain Name System) — доменне ім'я (або безпосередньо IP-адреса в числовому форматі) входить до складу URL для позначення комп'ютера (точніше — одного з його мережевих інтерфейсів), на якому працює відповідний веб-сервер.

Для перегляду інформації, отриманої від веб-сервера, на клієнтському комп'ютері використовується спеціальна програма — веб-браузер. Основна функція веб-браузера — відображення гіпертексту. Всесвітня павутина тісно пов'язана з поняттями гіпертексту і гіперпосилань, оскільки велика частина

інформації у Вебі представлена саме у вигляді гіпертексту. Для спрощення створення, зберігання і відображення гіпертексту у Всесвітній павутині традиційно використовується мова HTML (англ. HyperText Markup Language), мова розмітки гіпертексту. Процес розмітки гіпертексту називається версткою, а спеціалістів з розмітки називають веб-майстрами або вебмастерами.

Після створення HTML-розмітки, гіпертекст поміщається у файл, який стає найпоширенішим ресурсом Всесвітньої павутини. Коли HTML-файл стає доступний на веб-сервері, його називають «веб-сторінкою». Набір веб-сторінок утворює веб-сайт. У гіпертекст веб-сторінок додаються гіперпосилання, які допомагають користувачам Всесвітньої павутини легко переміщатися між ресурсами (файлами), незалежно від того, чи знаходяться вони на локальному комп'ютері, чи на віддаленому сервері. Гіперпосилання у Вебі засновані на технології URL.

Загалом, можна сказати, що Всесвітня павутина базується на трьох ключових компонентах: HTTP, HTML і URL. Проте, останнім часом HTML поступово втрачає свої позиції, поступаючись сучаснішим технологіям розмітки, таким як XHTML і XML. XML (англ. eXtensible Markup Language) позиціонується як основа для інших мов розмітки. Для покращення візуального сприйняття веб-сторінок стала широко використовуватися технологія CSS, яка дозволяє задавати єдині стилі оформлення для багатьох веб-сторінок. Ще одне нововведення, на яке варто звернути увагу, — система позначення ресурсів URN (англ. Uniform Resource Name).

Популярна концепція розвитку Всесвітньої павутини — це створення семантичної павутини. Семантична павутина є надбудовою над існуючою Всесвітньою павутиною, яка має на меті зробити інформацію в мережі більш зрозумілою для комп'ютерів. Це концепція мережі, в якій кожний ресурс, написаний людською мовою, супроводжується описом, зрозумілим комп'ютеру. Семантична павутина надає доступ до чітко структурованої інформації для будь-яких додатків, незалежно від платформи та мов програмування. Програми зможуть самостійно знаходити необхідні ресурси, обробляти інформацію,

класифікувати дані, виявляти логічні зв'язки, робити висновки і навіть ухвалювати рішення на основі цих висновків. При широкому поширенні та грамотному впровадженні семантична павутина може спричинити революцію в Інтернеті. Для створення опису ресурсів, зрозумілого комп'ютеру, у семантичній павутині використовується формат RDF (англ. Resource Description Framework), який базується на синтаксисі XML та використовує ідентифікатори URI для позначення ресурсів. Новітні розробки в цій галузі включають RDFS (англ. RDF Schema) та SPARQL (англ. Protocol And RDF Query Language), нову мову запитів для швидкого доступу до даних RDF.

Одним із ключових аспектів у розвитку Всесвітньої павутини є веб-розробка — процес створення веб-сайту або веб-додатку. Цей термін охоплює розробку додатків для електронної комерції, веб-дизайн, програмування для клієнтської та серверної частин, а також налаштування веб-сервера. Основні етапи веб-розробки включають:

- проектування сайту або веб-додатку;
- створення макетів сторінок;
- наповнення контентом;
- обслуговування працюючого сайту або його програмної бази.

Веб-додаток — це додаток, у якому клієнтом виступає браузер, а сервером — веб-сервер. Браузер є різновидом так званих тонких клієнтів. Він здатний відображати веб-сторінки та зазвичай входить до складу операційної системи, а його оновлення і супровід здійснює постачальник операційної системи. Вся логіка додатку зосереджується на сервері, а браузер лише відображає інформацію, завантажену з серверу через мережу. Однією з переваг цього підходу є те, що клієнти не залежать від конкретної операційної системи, що робить веб-додатки міжплатформеними сервісами.

Суттєвою перевагою розробки веб-додатків з підтримкою стандартних функцій браузера є те, що ці функції працюють незалежно від операційної системи клієнта. Замість написання різних версій для MS Windows, Mac OS X,

GNU/Linux та інших операційних систем, додаток створюється один раз і може працювати на будь-якій платформі. Однак різні реалізації HTML, CSS, DOM та інших специфікацій у браузерах можуть створювати проблеми під час розробки та підтримки веб-додатків. Крім того, можливість користувача налаштовувати багато параметрів браузера (наприклад, розмір шрифту, кольори, відключення підтримки сценаріїв) може перешкоджати коректній роботі додатку.

Інший, менш універсальний підхід, полягає у використанні Adobe Flash або Java аплетів для повної або часткової реалізації користувацького інтерфейсу. Оскільки більшість браузерів підтримують ці технології (зазвичай за допомогою плагінів), Flash або Java-додатки можуть працювати з більшою легкістю. Завдяки тому, що вони надають програмісту більший контроль над інтерфейсом, такі додатки можуть обходити багато несумісностей у конфігураціях браузерів, хоча несумісності між реалізаціями Java або Flash на стороні клієнта можуть призводити до різних ускладнень. Через архітектурну схожість з традиційними клієнт-серверними додатками, відомими як "товсті клієнти", існують суперечки щодо коректності віднесення подібних систем до веб-додатків; для них часто використовують альтернативний термін "Багаті Інтернет-додатки" (англ. Rich Internet Application).

Веб-додаток отримує запит від клієнта, виконує необхідні обчислення, формує веб-сторінку і відправляє її клієнту через мережу з використанням протоколу HTTP. Веб-додаток може також виступати як клієнт інших служб, таких як бази даних або інші веб-додатки, розташовані на інших серверах. Яскравим прикладом веб-додатку є система управління вмістом статей Вікіпедії: численні учасники можуть створювати мережеву енциклопедію, використовуючи браузери своїх операційних систем (будь то Windows, GNU/Linux чи інша система) без необхідності завантаження додаткових виконуваних модулів для роботи з базою даних статей.

Зараз набуває популярності новий підхід до розробки веб-додатків, відомий як Аїах. З використанням Аїах сторінки веб-додатку не

перезавантажуються повністю, а лише дозавантажують необхідні дані з сервера, що робить їх більш інтерактивними і ефективними.

Для створення веб-додатків використовуються різноманітні технології та мови програмування, такі як:

1. JavaScript: сценарна мова Netscape. Його можуть використовувати інтерфейсні розробники, а потім він працює на стороні браузера. Однак, можна використовувати його на стороні серверної частини, а потім використовувати як node.js або Ringo. Існує багато фреймворків, які можна використовувати для створення різноманітних анімацій, інтерактивних карт та інших елементів сайту, які роблять його зовнішній вигляд більш привабливим [\[17\]](#).

2. Python одна з мов програмування, яка найліпше підійде для вивчення початкового рівня розробки веб-сайтів. Він сумісний з більшістю платформ: Windows, Linux, MacOS і навіть iOS і Android для розробки мобільних додатків.

3. Ruby - це сервісна мова програмування, яка дозволяє об'єктно-орієнтоване програмування. Його відмінною рисою є менеджер для управління пакетами під назвою «gems». Ruby гнучкий і простий у використанні, що робить його ідеальним для більшості проектів. Фреймворк Ruby On Rails, заснований на мові Ruby, відкриває багато додаткових можливостей для користувачів.

4. Java - це популярна об'єктно-орієнтована мова програмування, яка використовується для розробки надійних програм. За допомогою потужних веб-фреймворків, таких як Spring і Java Server Faces (JSF), Java часто використовується для створення програм корпоративного рівня. також широко використовується великими компаніями для створення веб-додатків великого масштабу. Ця мова програмування відома своєю надійною платформою Java Enterprise Edition (Java EE).

5. C# - це об'єктно-орієнтована мова програмування, яка є частиною системи автоматизації, розробленої компанією Microsoft (WPL). C# використовується в поєднанні з .NET Framework для створення різноманітних програм, включаючи веб-застосунки, API, веб-служби та інші TypeScript;

6. Perl - це високорівнева, інтерпретована мова програмування, яка славиться своєю гнучкістю та ефективністю в опрацюванні текстових даних. Вона також відома своєю здатністю вирішувати широкий спектр програмних завдань, починаючи від швидкого розроблення скриптів до створення великих веб-додатків.

7. Swift - це сучасна мова програмування, створена компанією Apple для розробки додатків на платформах iOS, macOS, watchOS і tvOS. Вона відзначається високою швидкістю, безпекою та надійністю, що робить її першим вибором для багатьох розробників, які працюють у середовищі Apple.

8. Kotlin - це високорівнева, кросплатформна мова програмування загального призначення з виведенням типів і статичною типізацією. Вона призначена для ефективної взаємодії з Java, але відрізняється більш стислим синтаксисом завдяки виведенню типів. Kotlin головним чином орієнтований на платформу JVM, але також підтримує компіляцію в JavaScript для фронтенд-веб-додатків, що використовують React, або в нативний код через LLVM для розробки мобільних додатків для iOS та Android. Розробка мови ведеться компанією JetBrains, а торгіву марку Kotlin захищає Kotlin Foundation.

Деякі з них, наприклад, PHP, Perl, Python, мають відкритий код, розповсюджуються вільно і можуть використовуватися на практично будь-яких веб-серверах. Інші, як-от ASP, ASP.NET, Java, прив'язані до конкретних веб-серверів.

Для реалізації веб-сайту для кадрового бюро можна використовувати різні інструменти та технології, залежно від вимог до функціональності, бюджету, часу на розробку і технічних навичок команди. Нижче наведено основні засоби, які можуть бути корисними для створення такого веб-сайту:

1. Системи управління контентом (CMS): Таке програмне забезпечення створюється агентством як унікальний та запатентований продукт. Перевагою такого типу систем може бути те, що вони досить прості у використанні.

2. Платформи для створення веб-сайтів: Конструктори веб-сайтів — це не що інше, як платформа, яка дозволяє створювати модульні веб-сайти. Іншими

словами, це програмне забезпечення, яке допоможе створити власне місце в Інтернеті простим і приємним способом. Щоб реалізувати свій проект, вам не обов'язково вміти програмувати - достатньо використовувати готові шаблони та елементи інтерфейсу, які пропонують сайти.

3. Фреймворки та мови програмування: Для створення веб-сайтів використовуються різні мови програмування, найважливішими з яких є HTML, CSS і JavaScript. Окрім цих трьох основних мов, існують також інші мови та інструменти, які можна використовувати для створення веб-сайтів, такі як PHP, Ruby, Python та багато інших. Залежно від потреб і цілей веб-сайту, різні мови програмування можуть бути більш придатними. Важливо розуміти переваги та обмеження кожної мови та вибрати ту, яка найкраще відповідає потребам проекту.

Кінцевим вибором для реалізації веб-сайту для кадрового бюро в рамках цієї кваліфікаційної роботи було обрано редактор вихідного коду Visual Studio Code (VS Code) та мову розмітки HTML, CSS з таких причин:

Переваги Visual Studio Code:

1. Безкоштовність та відкритий вихідний код: VS Code є безкоштовним для використання і має відкритий вихідний код, що дозволяє вносити покращення та виправлення.

2. Підтримка різних мов програмування: Вбудована підтримка великої кількості мов програмування, включаючи JavaScript, Python, Java, C++, PHP, HTML, CSS та багато інших. Плагіни дозволяють додати підтримку нових мов та покращити існуючу.

3. Розширення та плагіни: Багата екосистема розширень, які можна встановити через вбудований менеджер розширень. Можливість налаштування редактора під конкретні потреби розробника, включаючи теми, підтримку нових мов, інструменти для дебагінгу, інтеграцію з системами контролю версій і багато іншого.

4. Інтеграція з Git та іншими системами контролю версій: Вбудована підтримка Git дозволяє легко керувати репозиторіями, виконувати коміти,

переглядати історію змін і вирішувати конфлікти. Підтримка інших систем контролю версій через розширення.

5. Інтелектуальні функції автозаповнення та підказок коду: IntelliSense забезпечує автозаповнення коду, підказки функцій, документування та багато іншого, що підвищує продуктивність і точність кодування. Підтримка синтаксичного аналізу і перевірки коду в реальному часі.

6. Вбудований дебагер: Можливість налагодження коду безпосередньо з редактора. Підтримка точок зупину, спостереження за змінними, виконання кроків по коду та інші функції дебагінгу.

7. Підтримка завдань та інтеграція з терміналом: Можливість налаштування завдань для автоматизації процесів, таких як складання проекту, тестування, запуск скриптів тощо. Вбудований термінал, який дозволяє виконувати команди безпосередньо в редакторі.

8. Кросплатформеність: VS Code працює на Windows, macOS та Linux, що дозволяє розробникам використовувати його на будь-якій платформі. Консистентний інтерфейс і функціональність на всіх платформах.

9. Потужні можливості налаштування та конфігурації: Можливість налаштування ключових комбінацій, створення користувацьких команд, зміна тем інтерфейсу та інше. Підтримка користувацьких конфігурацій для різних проектів і робочих середовищ.

10. Широкі можливості розширення: VS Code дозволяє створювати власні розширення, що відкриває широкі можливості для розширення функціональності редактора.

Загальні переваги:

1. Швидкість та легкість: Незважаючи на свої потужні можливості, VS Code є досить легким і швидким, що робить його зручним для використання навіть на менш потужних машинах.

2. Інтуїтивно зрозумілий інтерфейс: Зручний та зрозумілий інтерфейс, який легко освоїти навіть новачкам.

Переваги HTML:

1. Простота синтаксису і структурування документів.
2. Підтримується всіма браузерами та платформами.
3. Використовується для створення різних типів контенту, включаючи тексти, зображення, відео та форми.
4. Стандартизація: HTML є стандартом, затвердженим W3C (World Wide Web Consortium), що забезпечує його консистентність і сумісність.
5. Постійні оновлення і поліпшення стандарту (на даний момент HTML5).
6. Семантична розмітка: HTML5 забезпечує семантичні теги, які покращують доступність та SEO (наприклад, `<header>`, `<footer>`, `<article>`, `<section>`).

Переваги CSS:

1. Розділення контенту та стилю: CSS дозволяє відокремити презентаційний стиль від структури контенту, що сприяє чистішому і більш організованому коду. Полегшує підтримку та оновлення стилів.
2. Можливості стилізації та дизайну: CSS дозволяє створювати привабливі і адаптивні дизайни, використовуючи різні властивості (кольори, шрифти, відступи, анімації тощо).
3. Адаптивний дизайн: Застосовується для створення мобільно-оптимізованих веб-сайтів.
4. Підтримка модульності та повторного використання: Використання класів та ідентифікаторів дозволяє повторно використовувати стилі на різних елементах сторінки.
5. Швидкість завантаження: CSS-файли кешуються браузерами, що дозволяє прискорити завантаження сторінок. Можливість мінімізації CSS для зменшення розміру файлів.

Загальні переваги HTML та CSS:

1. Взаємодія з іншими технологіями: Легко інтегруються з JavaScript для створення інтерактивних веб-додатків. Використовуються разом з іншими веб-технологіями та фреймворками (наприклад, React, Angular).

2. Підтримка доступності: Семантична розмітка HTML і можливості CSS дозволяють створювати доступні веб-сторінки для користувачів з обмеженими можливостями. Використання ARIA (Accessible Rich Internet Applications) атрибутів для покращення доступності.

3. Пошукова оптимізація (SEO): Правильна розмітка HTML сприяє кращому індексуванню сторінок пошуковими системами. Використання метатегів і семантичних тегів покращує видимість у пошукових результатах.

Було звернено увагу, що використання HTML, CSS та JavaScript для створення веб-сайту для кадрового бюро являється дуже ефективним рішенням. HTML (мова розмітки) використовується для створення структури та вмісту веб-сторінок, CSS (контроль каскадних таблиць стилів) відповідає за їх вигляд і оформлення, а JavaScript (мова програмування) додає інтерактивність та динаміку до сайту.

За допомогою HTML можна створити різноманітні елементи веб-сторінок, такі як форми для збору інформації про кандидатів на вакансії або списки вакансій. CSS дозволяє стилізувати ці елементи, змінювати кольори, розміри шрифтів, розташування елементів на сторінці та інше, щоб створити привабливий та професійний вигляд сайту. JavaScript додає можливості, такі як валідація форм, анімація, динамічне завантаження контенту, що може полегшити взаємодію користувачів з веб-сторінкою та покращити їх досвід.

Такий підхід дозволяє створити функціональний та естетичний веб-сайт для кадрового бюро, який забезпечить зручність користувачам та ефективність роботи з персоналом.

2.2. Засоби для реалізації веб-сайту для кадрового бюро

Під час розробки веб-сайту для кадрового бюро важливо обрати належні бібліотеки та фреймворки, які полегшать та прискорять процес розробки, а також забезпечать високу якість та продуктивність кінцевого продукту. Використання

готових рішень дозволяє уникнути написання надлишкового коду та зосередитися на основному інтерфейсі веб-сайту.

Для реалізації веб-сайту для кадрового бюро буде застосований комплексний підхід, який включає використання різноманітних технологій, фреймворків та бібліотек. Основна ціль – створити зручний, інтерактивний та ефективний сайт, який буде адаптований до потреб користувачів та забезпечить їм належний рівень обслуговування та взаємодії з кадровим бюро.

Для реалізації веб-сайту для кадрового бюро було застосовано текстовий редактор Visual Studio Code, який є офіційним інструментом від Microsoft та забезпечує потужну підтримку для веб-розробки. Visual Studio Code пропонує зручний інтерфейс, інтелектуальний редактор коду, вбудований термінал, інструменти для налагодження та профілювання, а також інтеграцію з системами контролю версій та можливості для розгортання веб-сайтів.

Для зручності використання Visual Studio Code рекомендовано використовувати такі розширення:

1. Prettier - Code formatter - популярний інструмент, який автоматично форматує код для забезпечення його читабельності та відповідності стандартам. Він підтримує різноманітні мови програмування, включаючи JavaScript, HTML, CSS, TypeScript, JSON та інші.
2. Open in Browser - розширення, яке дозволяє відкривати файли HTML та інші веб-документи безпосередньо у веб-браузері з середовища Visual Studio Code. Завдяки цьому інструменту, розробники можуть швидко переглядати зміни, внесені до коду, і тестувати веб-сторінки без необхідності вручну відкривати файли у браузері. Це підвищує ефективність робочого процесу та спрощує налагодження та перевірку веб-додатків.
3. Material Icon Theme - популярне розширення, яке додає велику бібліотеку іконок до редактора коду.
4. Live Server - розширення, яке створює локальний сервер і автоматично перезавантажує веб-сторінку при збереженні файлів.

5. HTML CSS Support - розширення, яке покращує підтримку CSS у файлах HTML. Воно забезпечує автозаповнення, підсвічування синтаксису та перевірку коду в режимі реального часу CSS, вбудованого в HTML-документи.

Мовами програмування для реалізації веб-сайту було обрано HTML, CSS та JavaScript – сучасні, безпечні та лаконічні мови, які широко використовуються для створення веб-додатків. HTML і CSS забезпечують структурне та стильове оформлення сторінок, тоді як JavaScript додає інтерактивність і динаміку. Ці технології пропонують багато переваг, таких як зручне функціональне програмування, розширення функціоналу та підвищена продуктивність розробників. Використання HTML, CSS та JavaScript дозволить створити більш читабельний, лаконічний та підтримуваний код, що спростить процес розробки та подальшу підтримку веб-сайту кадрового бюро.

Для оформлення дизайну сайту було обрано програму Figma. Figma була створена, щоб заповнити прогалину в наданні безкоштовного графічного програмного забезпечення для професійного використання всім користувачам. Програма поєднує в собі найкращі інструменти та рішення, відомі з конкурентних комерційних графічних пакетів. Він дозволяє створювати та редагувати як растрову, так і векторну графіку в спільному робочому просторі. Крім того, програмне забезпечення має настільки зрозумілий та інтуїтивно зрозумілий інтерфейс, що навіть для початківця графічного дизайнера навігація в ньому не становить жодних проблем. Засоби для реалізації веб-сайту для кадрового бюро.

З появою Figma створювати складні графічні дизайни стало легше, насамперед завдяки покращеній роботі кількох команд спеціалістів. В одному інструменті графічні дизайнери, веб-дизайнери та програмісти можуть ділитися ходом роботи над проектом у режимі реального часу, представляти його клієнту, а також спілкуватися та передавати файли між собою.

Переваги Figma:

1. Хмарна платформа: Спільна робота в реальному часі. Figma дозволяє декільком користувачам працювати над одним проектом одночасно, що сприяє ефективній співпраці.
2. Зберігання в хмарі: Усі проекти зберігаються в хмарі, що забезпечує доступ до них з будь-якого пристрою без необхідності синхронізації файлів.
3. Кросплатформеність: Доступ з будь-якого пристрою: Figma працює в браузері, а також має десктопні додатки для Windows та macOS. Відсутність необхідності в установці: Можна почати працювати з проектом, просто відкривши браузер.
4. Простий і зручний інтерфейс: Інтуїтивний дизайн Figma має зручний та зрозумілий інтерфейс, що робить його легким для освоєння навіть новачкам. Усі необхідні інструменти для дизайну знаходяться під рукою, що робить процес роботи швидким та ефективним.
5. Потужні інструменти для дизайну: Підтримка векторної графіки дозволяє створювати масштабовані елементи без втрати якості. Можливість створення та повторного використання компонентів і стилів для забезпечення консистентності дизайну.
6. Прототипування: Можливість створення інтерактивних прототипів з переходами, анімаціями та інтерактивними елементами. Легкий доступ до прототипів для клієнтів і команд розробників.
7. Інтеграції та плагіни: Велика кількість плагінів для різних задач, таких як генерування контенту, управління кольоровими палітрами, перевірка доступності та інше.
8. Інтеграція з іншими інструментами: Підтримка інтеграцій з популярними інструментами для управління проектами (Jira, Trello), розробки (Zeplin, Avocode), та хмарними сховищами (Google Drive).
9. Коментування та відгуки.
10. Відстеження змін та повернення до попередніх версій макетів при необхідності.

11. Бібліотеки компонентів: Можливість створення та спільного використання бібліотек компонентів, що забезпечує консистентність дизайну в масштабних проектах.

Figma дозволяє створювати макети сторінок з урахуванням вимог користувачів і бізнесу. Для кадрового бюро можна розробити:

- Головну сторінку із загальною інформацією про бюро.
- Сторінку з вакансіями, яка дозволяє фільтрувати і шукати вакансії за різними критеріями.
- Форми для подачі резюме і контактної інформації.
- Сторінку для компаній-роботодавців з інформацією про послуги бюро.

Отже, Figma є потужним і зручним інструментом для розробки дизайну веб-сайту кадрового бюро. Її функціональність дозволяє ефективно створювати, тестувати і співпрацювати над проектом, забезпечуючи високу якість кінцевого продукту.

Прототипи в інших операційних системах:

1. Найбільш відомий аналог Figma – векторний редактор Sketch. Програма Sketch нативно інтегрується в macOS та Windows і використовує власні засоби операційки для відображення контенту, тому працює редактор плавно, не потребує великого запасу потужностей заліза і займає мінімум місця в пам'яті. Готові скетчі можна експортувати у форматах: JPEG, PNG, TIFF, PDF, SVG.

2. Blocs Website Builder – це візуальний редактор для верстки простих адаптивних сайтів. У редакторі проектування реалізовано за принципом конструктора: вибираєте готові блоки, складаєте екрани сайту або додатки, редагуєте стилі, додаєте графіку та інтерактивні можливості. Готовий прототип дизайну зберігається у вигляді готового пакету файлів HTML + CSS, причому у розширеній версії також реалізовано експорт у форматі теми для CMS Wordpress.

3. InVision – хмарний сервіс для дизайну, прототипування, командного проектування інтерфейсів додатків та сайтів.

4. Zeplin – не заміна, а скоріше доповнення до Figma та інших згаданих у статті альтернативних редакторів прототипів. Цей сервіс встановлюється у вигляді плагіну до реакторів прототипів і дозволяє генерувати валідний програмний код для верстки та реалізації інтерактивних можливостей, які дизайнер заклав у макет.

5. Visme – онлайн-редактор графічного контенту, який переважно використовується для створення презентацій.

Яку з альтернатив Figma вибрати – залежить від розв'язуваних завдань.

2.3. Висновки до другого розділу

У другому розділі були детально розглянуті та обґрунтовані вибір програмних засобів, бібліотек, фреймворків та інтегрованого середовища розробки для реалізації веб-сайту для кадрового бюро. Після ретельного аналізу та оцінки різних альтернатив були зроблені наступні ключові вибори:

1. HTML, CSS, та JavaScript: Основні мови для створення веб-інтерфейсу, розмітки сторінок та динамічного контенту.
2. Інтегроване середовище розробки (IDE) Visual Studio Code, для зручного програмування та налагодження коду.
3. Інструменти версійного контролю Git, для збереження та керування версіями проекту.
4. Онлайн-інструмент для дизайну і прототипування Figma.

Використання вибраних технологій, фреймворків і бібліотек для створення веб-сайту для кадрового бюро є ключовим етапом у процесі розробки, оскільки вони визначають якість, продуктивність та функціональність кінцевого продукту. HTML, CSS і JavaScript виступають основними мовами для розробки веб-інтерфейсу та динамічного контенту, забезпечуючи користувачам зручний та інтуїтивний інтерфейс.

Інтегроване середовище розробки (IDE) Visual Studio Code є важливим інструментом для програмістів, оскільки воно надає зручні можливості для

редагування коду, налагодження, відлагодження та спільної роботи над проектами. Інструменти версійного контролю Git дозволяють ефективно керувати версіями проекту, спільно працювати з командою розробників та впроваджувати зміни безпечно й ефективно.

Онлайн-інструмент для дизайну і прототипування Figma є необхідним інструментом для створення привабливого та функціонального дизайну веб-сайту. Він дозволяє розробникам та дизайнерам спільно працювати над проектом, створювати прототипи, тестувати рішення та вносити необхідні зміни для оптимізації користувацького досвіду.

Завдяки ретельному аналізу і вибору оптимальних інструментів, розроблений веб-сайт для кадрового бюро стає потужним інструментом для покращення процесів пошуку роботи та управління людськими ресурсами. Він забезпечує ефективне функціонування всіх аспектів роботи з персоналом, підвищуючи ефективність та конкурентоспроможність підприємств.

Крім цього, інтеграція з іншими сервісами, такими як платіжні системи, сервіси для відправлення електронної пошти та інструменти для аналітики, дозволяє забезпечити додаткову функціональність та гнучкість. Наприклад, інтеграція з платіжними системами може дозволити автоматизувати процеси оплати послуг кадрового бюро, а сервіси для відправлення електронної пошти забезпечать зручну комунікацію з кандидатами та клієнтами.

Впровадження аналітичних інструментів дозволить керівництву кадрового бюро відстежувати ключові показники ефективності, аналізувати поведінку користувачів та приймати обґрунтовані рішення щодо подальшого розвитку сайту та послуг, які він надає. Це сприятиме підвищенню задоволеності клієнтів та допоможе адаптуватися до змін у ринку праці.

У результаті, використання сучасних технологій та інструментів для розробки веб-сайту для кадрового бюро не лише покращує процеси управління кадрами, але й сприяє загальному розвитку та успіху бізнесу, надаючи йому конкурентні переваги та підвищуючи рівень задоволеності клієнтів.

РОЗДІЛ 3

РОЗРОБКА ВЕБ-САЙТУ ДЛЯ КАДРОВОГО БЮРО В ТЕКСТОВОМУ РЕДАКТОРІ VISUAL STUDIO CODE (VS CODE)

3.1. Розробка загальної архітектури веб-сайту для кадрового бюро в текстовому редакторі Visual Studio Code (VS Code)

Мета створення веб-сайту для кадрового бюро полягає в покращенні та більш ефективного пошуку роботи для потенційних працівників та іноземців, які перебувають на території Республіки Польщі. Головним завданням даного веб-сайту є забезпечення користувачів зручним та ефективним інструментом для пошуку праці. Веб-сайт має різноманітні розділи: Шапка сайту(Головна), коротка інформація про кадрове бюро(Коротко про нас), Обсяг послуг, Новини HR – індустрії, Цікаві посилання та Контакти. Дизайн сайту є приємним та сучасним. Інтерфейс є зрозумілим та привабливим для користувачів.

Для розробки веб-сайту для кадрового бюро було обрано текстовий редактор Visual Studio Code (VS Code) з таких причин:

1. Сумісність з різними операційними системами: Visual Studio Code (VS Code) доступний для Windows, macOS і Linux, що забезпечує гнучкість і зручність для розробників, незалежно від використовуваної платформи.

2. Плагіни та розширення: Велика кількість доступних розширень дозволяє налаштовувати редактор під конкретні потреби проекту. Наприклад, розширення для підтримки мов програмування, інструментів для налагодження, інтеграції з системами контролю версій та іншого.

3. Налаштування середовища: Можливість налаштування інтерфейсу та функціональності редактора відповідно до індивідуальних уподобань розробника.

4. Широкий спектр підтримуваних мов: VS Code підтримує більшість мов програмування, включаючи JavaScript, Python, HTML, CSS, а також інші мови, які можуть знадобитися для розробки як фронтенду, так і бекенду сайту.

5. Інтеграція з системами контролю версій: Вбудована підтримка Git дозволяє легко керувати версіями коду, проводити спільну розробку та відслідковувати зміни.

6. Налагодження та тестування: Вбудовані інструменти для налагодження та тестування коду допомагають швидко знаходити та виправляти помилки.

7. Підтримка роботи з контейнерами та хмарними сервісами.

8. Легка вага: VS Code є відносно легким і швидким редактором, що забезпечує високу продуктивність навіть на менш потужних комп'ютерах.

9. Спільнота розробників: Велика та активна спільнота користувачів VS Code забезпечує наявність великої кількості ресурсів, таких як документація, форуми та навчальні матеріали.

10. Часті оновлення: Регулярні оновлення редактора додають нові функції та покращення, забезпечуючи актуальність інструменту.

У рамках цієї архітектури будуть використані наступні ключові компоненти: `index.html`, `style.css`, `reset.css`, `script.js`.

«`index.html`» — це основний файл HTML, який зазвичай використовується як головна сторінка веб-сайту. Він є першою сторінкою, яку браузер завантажує при доступі до веб-сайту. Вміст цього файлу визначає структуру та початковий вигляд веб-сторінки.

«`style.css`» — це файл каскадних таблиць стилів (CSS), який використовується для стилізації HTML-документів. Він визначає зовнішній вигляд та форматування елементів на веб-сторінці, таких як кольори, шрифти, відступи, розміри та розташування.

«`reset.css`» — це файл CSS, який використовується для скидання (reset) стилів браузера до базового рівня. Це робиться для того, щоб усунути різні стилі за замовчуванням, які різні браузери застосовують до HTML-елементів,

забезпечуючи тим самим більш уніфікований зовнішній вигляд веб-сторінки незалежно від браузера.

«script.js» — це файл JavaScript, який містить скрипти для додавання динамічної функціональності та взаємодії з користувачем на веб-сторінці. Цей файл може включати в себе функції, обробники подій, виклики API, маніпуляції з DOM (Document Object Model) і багато іншого.

Ця архітектура забезпечує чітке розділення обов'язків між компонентами, полегшуючи розуміння, тестування та підтримку коду. Такий підхід також дозволяє легко розширювати функціональність веб-сайту в майбутньому за рахунок додавання нових активностей, фрагментів або модифікації існуючих компонентів.

Головна сторінка сайту є головною та виступає в ролі вхідної точки. Після запуску сайту, відображається головний екран (див. рис. 3.1).

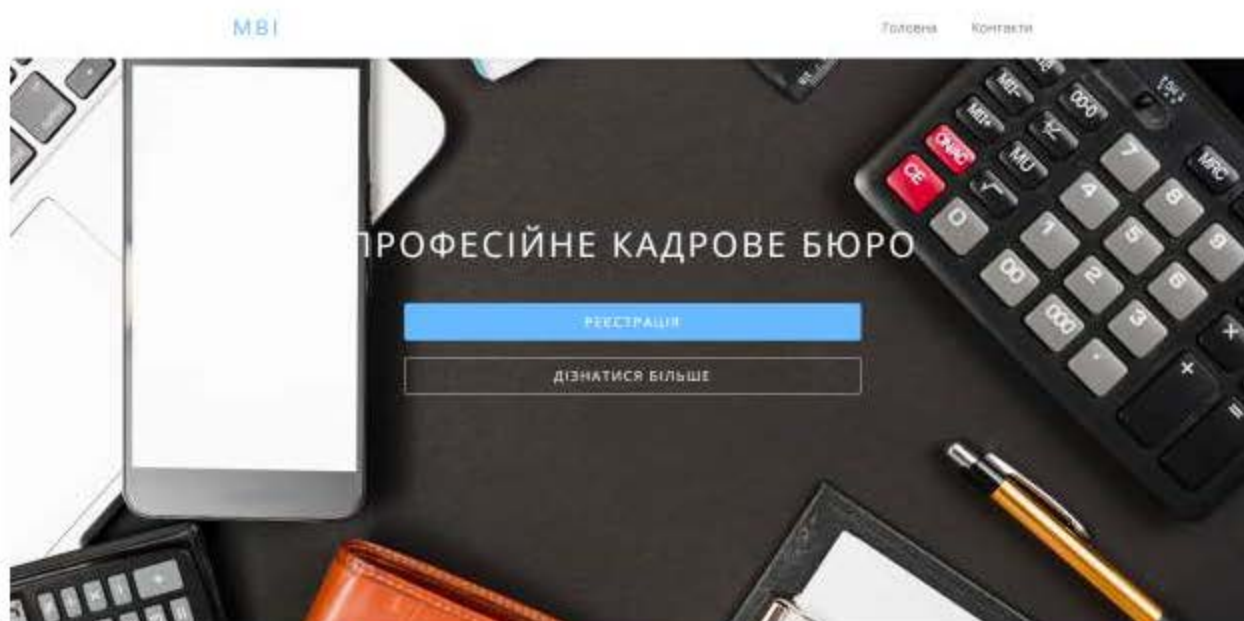


Рисунок 3.1 – Головна сторінка сайту при розмірі 1920*1080px

Головна сторінка містить в собі: логотип (див. рис. 3.2), шапку сайту, фонове зображення, заголовок і кнопки.



Рисунок 3.2 – Логотип сайту

Тематика веб-сайту була визначена відповідно до особливостей місця практики. Перед початком створення сайту було обрано шрифт «Open Sans», який буде використовуватися на всьому сайті в різних розмірах. Всі сторінки сайту пристосовуються до різних розмірів екрану пристроїв.

Сторінка «Коротко про нас» (див. рис. 3.3) містить інформацію про підприємство. Складається з заголовку, гасла та контейнерів з інформацією про підприємство.



Рисунок 3.3 – Сторінка «Коротко про нас»

Сторінка «Обсяг послуг» (див. рис. 3.4) містить перелік послуг представлені підприємством та складаються із контейнерів заголовку, гасла, фону, кнопок та переліку обов'язків кожної представленої послуги.



Рисунок 3.4 – Сторінка «Обсяг послуг»

Одна з послуг представлених підприємством (див. рис. 3.5).

Сторінка «Новини» (див. рис. 3.5) містить слайдер із кнопок, які при натисканні змінюють колір. Кнопка «Зміни в КЗпП 2022/2023» є головною. Кожен розділ показує дану йому інформацію у контейнері.



Рисунок 3.5 – Сторінка «Новини»

Слайдер містить кнопки «Декларація про платежі» (див. рис. 3.6).



Рисунок 3.6 – Кнопка «Декларація про платежі»

Остання кнопка слайдера «Підвищення зарплати у 2023 році» (див. рис. 3.7).



Рисунок 3.7 – Кнопка «Підвищення зарплати у 2023 році»

Сторінка «Контакт» (див. рис. 3.8) є компонентом «footer» та містить у собі короткий зміст про підприємство, цікавий перелік джерел та контакти

підприємства. Нижче розташовано гасло на польській мові «Biuro usług kadrowo-płacowych Marcin Figlarowicz MBI zaprasza do współpracy». Переклад: «Бюро кадрових та зарплатних послуг Марчина Фігларовича запрошує до співпраці».

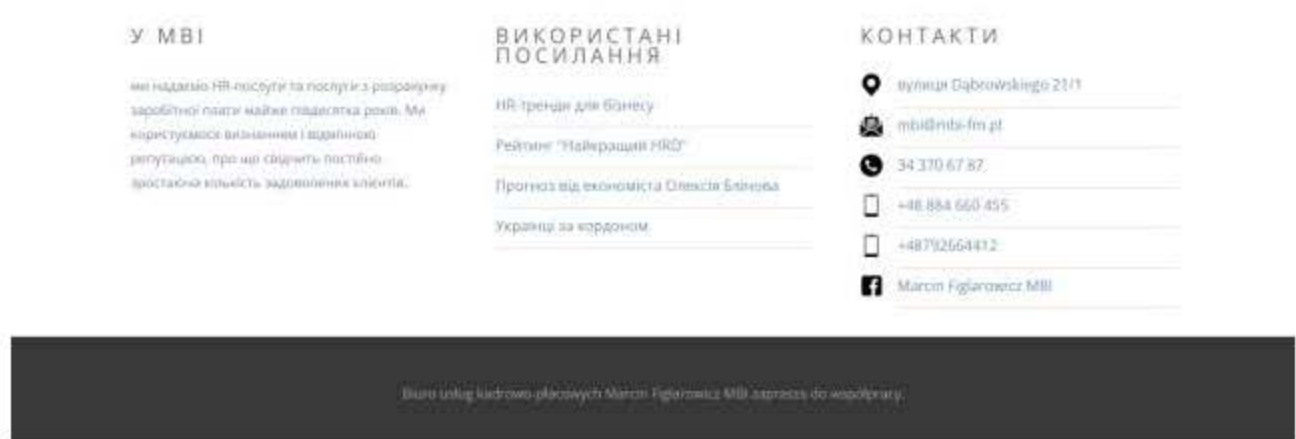


Рисунок 3.8 – Сторінка «Контакт»

3.2. Тестування роботи системи

Для вирівнювання сторінок сайту при різних розширення екрану використовується такий код (див. рис. 3.9):

```

290 @keyframes fade {
291   from {
292     opacity: 0.3;
293   }
294   to {
295     opacity: 1;
296   }
297 }
298

```

Рисунок 3.9 – Код вирівнювання сторінок сайту

Загальний вигляд сторінки при розширенні екрану Apple iPhone 5C/5S 640*1136px (див. рис. 3.10).

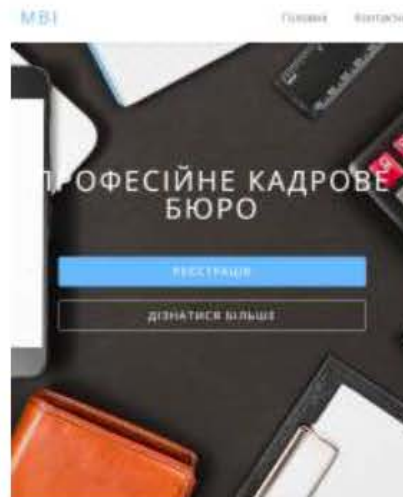


Рисунок 3.10 – Головна сторінка сайту при розмірі 1136 * 640px

Шапка сайту, яка має анімаційні ефекти має такий код (див. рис. 3.11):

```

14 // Дана функція показує переданий номер табу і робить відповідну йому кнопку активною
15
16 function showTab(index) {
17     tabsItems[index].classList.remove("hide");
18     tabsBtns[index].classList.add("active");
19 }
20
21 hideTabs();
22 showTab(0);
23
24 tabsBtns.forEach((btn, index) =>
25     btn.addEventListener("click", () => {
26         hideTabs();
27         showTab(index);
28     })
29 );

```

Рисунок 3.11 – Анімація на мові JavaScript

Reset.css, який обнуляє стилі браузера, надаючи всім елементам однакові стилі за замовчуванням має такий код (див. рис. 3.12):

```

23 h1,
24 h2,
25 h3,
26 h4,
27 h5,
28 h6,
29 p {
30     font-size: inherit;
31     font-weight: inherit;
32     line-height: inherit;
33 }
34
35 p + p {
36     margin-top: 1em;
37 }
38
39 ul {
40     list-style: none;
41 }
42
43 img,
44 svg,
45 video {
46     display: inline-block;
47     vertical-align: top;
48     max-width: 100%;
49     height: auto;
50     object-fit: cover;
51 }

```

Рисунок 3.12 – Анулювання елементів сайту

Reset.css допомагає вирішити проблему, видаляючи стилі за замовчуванням і забезпечуючи єдину основу для подальшого стилювання.

Загалом, весь код сайту можна поділити на: “index.html”, “styles.css”, “reset.css”, “script.js”.

“index.html”:

1. Структура HTML: Це скелет документа, що включає в себе елементи, такі як <html>, <head> і <body>, які визначають основну структуру та зміст сторінки.

2. Метатеги: Вони вказують на інформацію про документ, таку як кодування символів (<meta charset="UTF-8">), налаштування для адаптивності на мобільних пристроях (<meta name="viewport" content="width=device-width, initial-scale=1.0">) і заголовок документа (<title>).

3. Посилання на CSS файли: Вони підключають зовнішні стилі до документа (<link rel="stylesheet" href="reset.css"> і <link rel="stylesheet" href="styles.css">). В першому випадку, reset.css, це файл CSS, який ми

використовуємо для скидання стилів браузеру, а в другому випадку, `styles.css`, це файл, який містить власні стилі для нашого документа.

4. Вміст сторінки: Включає в себе різні елементи, такі як заголовок `<header>`, основний зміст `<main>` з різними секціями `<section>`, і підвал `<footer>` з копірайтом.

“`styles.css`”:

1. Організація стилів: Файл `styles.css` може бути організований таким чином, щоб розділити стилі на різні секції або категорії, що полегшує розуміння та підтримку коду. Наприклад, стилі для заголовків можуть бути розділені від стилів для тексту чи навігаційного меню.

2. Використання коментарів: Коментарі в CSS можуть використовуватися для пояснення різних частин коду, що полегшує розуміння і робить його більш підтримуваним.

3. Модульність: Файл `styles.css` може бути написаний з урахуванням принципів модульності, де кожен блок стилів відповідає окремому компоненту або частині веб-сторінки. Це сприяє повторному використанню та масштабованості коду.

4. Адаптивність: Стилi можуть бути адаптовані до різних розмірів екрану та пристроїв за допомогою медіа-запитів та адаптивних технік. Наприклад, ви можете встановити різні розміри шрифтів або відступи для мобільних пристроїв та настільних комп'ютерів.

5. Каскадність: CSS базується на принципі каскадності, що означає, що стилі можуть спадати або перевизначати один одного в залежності від їх специфічності та порядку визначення. Це важливо при вирішенні конфліктів у стилях.

6. Спадкоємність: Стилi можуть успадковувати властивості від батьківських елементів, що дозволяє створювати консистентний та легко змінюваний дизайн.

7. Кросбраузерність: Стили можуть бути визначені з урахуванням сумісності з різними веб-браузерами, забезпечуючи однаковий вигляд сторінки на різних платформах.

“reset.css”:

1. Нульові значення для маржів та відступів: Скидання значень `margin` та `padding` для всіх елементів, щоб уникнути внутрішніх відступів та зовнішніх маржів, які можуть встановлюватися браузером за замовчуванням.

2. Скидання стандартних значень для шрифтів: Встановлення значення `font` на `inherit` для всіх елементів, щоб уникнути будь-яких стилів шрифтів, які можуть бути встановлені за замовчуванням браузера.

3. Скидання стилів для списків: Видалення маркерів списків та скидання значень, пов'язаних зі списками, для уникнення непотрібного оформлення.

4. Скидання границь для елементів: Встановлення `border` на `none` для всіх елементів, щоб уникнути будь-яких границь, які можуть бути встановлені браузером за замовчуванням.

5. Скидання стилів для таблиць: Встановлення `border-collapse` на `collapse` та інших значень, які забезпечують однорівний вигляд таблиць на сторінці.

6. Скидання значень позиціонування: Встановлення значення `position` на `static` для всіх елементів, щоб уникнути будь-яких варіацій позиціонування, що можуть бути встановлені браузером за замовчуванням.

“script.js”:

1. Взаємодія з DOM: JavaScript може використовуватися для вибору елементів на сторінці та модифікації їх вмісту, стилю або атрибутів.

2. Обробка подій: JavaScript може додавати обробники подій до елементів сторінки, таких як клік, наведення миші, натискання клавіші тощо.

3. Асинхронні запити до сервера: JavaScript може використовувати XMLHttpRequest або Fetch API для здійснення асинхронних запитів до сервера, таких як отримання або відправлення даних.

4. Маніпулювання кукісами та локальним сховищем: JavaScript може зберігати дані на локальному комп'ютері користувача за допомогою кукісів (cookies) або локального сховища (local storage) для збереження даних між сесіями.

5. Анімація та взаємодія з CSS і SVG: JavaScript може створювати анімації та маніпулювати CSS-стилями, щоб створювати рухливі ефекти на сторінці. Також може бути використано для маніпулювання SVG-графікою на сторінці.

6. Робота з мережею: JavaScript може взаємодіяти з мережею, наприклад, для отримання даних з віддалених API або для створення WebSocket-з'єднань.

7. Створення власних функцій та об'єктів: JavaScript дозволяє створювати власні функції та об'єкти для структурування коду та виконання певних завдань.

8. Використання бібліотек та фреймворків: JavaScript може використовувати різноманітні бібліотеки та фреймворки, такі як jQuery, React, Angular або Vue.js, для спрощення розробки та забезпечення певних функцій.

9. Обробка помилок: JavaScript дозволяє обробляти помилки, які можуть виникати під час виконання коду, за допомогою конструкцій try-catch або обробки подій помилок.

3.3. Висновки до третього розділу

Сайт кадрового бюро розроблений з використанням мов програмування HTML, CSS та JavaScript, представляє собою сучасну та функціональну платформу, яка ефективно об'єднує роботодавців та шукачів роботи.

HTML забезпечує структурну основу веб-сторінки, дозволяючи організувати контент у логічний та доступний формат. CSS додає стилізацію, роблячи сайт візуально привабливим та зручним для користувачів, встановлюючи кольори, шрифти, розташування елементів та інші стилістичні

властивості. JavaScript додає динамічності та інтерактивності, дозволяючи реалізовувати складні функції, такі як валідація форм, інтерактивні карти, анімації та інші взаємодії з користувачем.

Основними характеристиками даного ресурсу є його зручність, інформативність та широкий спектр послуг. Сайт вирізняється своєю інтуїтивно зрозумілою навігацією та привабливим дизайном, що робить користування ним простим та приємним.

Важливою особливістю сайту є наявність розділу з корисними статтями та порадами щодо кар'єрного розвитку, написаними експертами у сфері HR. Це сприяє підвищенню професійного рівня користувачів та їх готовності до ринку праці.

Розробка сайту кадрового бюро на основі HTML, CSS та JavaScript також забезпечує високу сумісність з різними браузерами та пристроями, що є критично важливим у сучасному цифровому світі. Завдяки адаптивному дизайну, користувачі можуть зручно переглядати сайт як на десктопах, так і на мобільних пристроях.

Отже, використання HTML, CSS та JavaScript для створення сайту кадрового бюро є оптимальним рішенням, що дозволяє досягти високої функціональності, привабливого дизайну та зручності використання, що в сукупності сприяє успішній діяльності кадрового бюро в інтернеті.

Під час тестування роботи додатку було виявлено такі проблеми:

1. Оскільки сайт є статичним, база даних відсутня.
2. Не всі кнопки мають посилання на сторінку.

Загалом, веб-сайт кадрового бюро є ефективним інструментом для підтримки та розвитку професійної кар'єри, задовольняючи потреби як роботодавців, так і шукачів роботи. Його функціональність, зручність та широкий спектр послуг роблять його цінним ресурсом у сучасному ринку праці.

ВИСНОВКИ

У результаті виконання дипломної роботи було розглянуто та розроблено веб-сайт для кадрового бюро на текстовому редакторі Visual Studio Code (VS Code) з використанням мов програмування HTML, CSS та JavaScript. Цей сайт спрямований на автоматизацію та покращення процесу пошуку роботи для потенційних користувачів.

При створенні сайту враховувалися сучасні технології та вимоги до програмного забезпечення, такі як модульна структура, інтуїтивний та естетичний інтерфейс, підтримка різних рівнів складності, оптимізація для різних пристроїв, регулярні оновлення контенту та функціональності, а також дотримання стандартів безпеки та конфіденційності даних користувачів.

Додаток складається з декількох основних сторінок. Інтерфейс реалізовано з використанням інструментів Visual Studio Code (VS Code) та мов програмування HTML, CSS та JavaScript, що дозволило створити зручний та привабливий користувацький досвід. Під час тестування додатка було підтверджено його відповідність очікуванням щодо зручності використання та стабільної роботи на різних пристроях.

Отримані результати мають практичне значення, оскільки сприяють автоматизації та покращенню процесу пошуку роботи для широкої аудиторії користувачів. Сайт кадрового бюро стає важливим інструментом для роботодавців та шукачів роботи, полегшуючи процес найму та підвищуючи шанси знайти підходящу вакансію.

Отже, розроблений веб-сайт для кадрового бюро є важливим кроком у покращенні якості та ефективності веб-додатків. Цей додаток використовує передові технології та забезпечує надійну та зручну роботу, що в майбутньому має підвищити конкурентоспроможність користувачів на ринку праці та задовольнити їхні потреби у пошуку роботи.

Робота з людськими ресурсами та роль кадрового бюро є критично важливими аспектами управління сучасними організаціями. Успішне управління

людськими ресурсами сприяє підвищенню ефективності та конкурентоспроможності підприємств. Кадрові бюро відіграють ключову роль у цьому процесі, забезпечуючи належне функціонування всіх аспектів роботи з персоналом, від набору та відбору до навчання і розвитку співробітників.

Ефективне управління людськими ресурсами включає ключові елементи, такі як надання інформації та комунікація, набір, відбір та навчання персоналу, а також оцінка продуктивності. Кадрові бюро забезпечують працівників важливою інформацією про кадрову політику, процедури, соціальні виплати та інші аспекти, що сприяє покращенню внутрішніх комунікацій і підвищенню обізнаності співробітників. Відділи кадрів розробляють стратегії найму, залучають кандидатів, проводять відбір і організовують програми навчання, що сприяє розвитку компетенцій працівників та підвищенню їх ефективності. Системи оцінювання співробітників дозволяють точно контролювати їхні досягнення та визначати області для розвитку.

Завдяки ретельному підходу до аналізу існуючих рішень, врахуванню потреб клієнтів, конкурентному аналізу, розробці інтуїтивного дизайну та забезпеченню високої якості технічного виконання, розроблений веб-сайт кадрового бюро став важливим інструментом для покращення процесу пошуку роботи та управління людськими ресурсами. Це сприяє підвищенню ефективності та конкурентоспроможності підприємств, забезпечуючи належне функціонування всіх аспектів роботи з персоналом.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. WEB BOOK СТВОРЕННЯ ВЕБ-ДОДАТКУ: ОСНОВНІ АСПЕКТИ ЕФЕКТИВНОЇ РОЗРОБКИ ВЕБ-САЙТУ [Електронний ресурс]. - Режим доступу: <https://webbookstudio.com/ua/>
2. Бюро статистики праці (BLS) [Електронний ресурс]. - Режим доступу: <https://www.bls.gov/>
3. Герасименко К.В. Старкова Е.В. (2023) Все про створення вебсайтів — частина 1 [Електронний ресурс]. - Режим доступу: <https://www.education.ua/blog/48352/>
4. Web Development vs App Development: An Intense Analysis | 2024 [Електронний ресурс]. - Режим доступу: <https://www.guvi.com/blog/web-development-vs-app-development/>
5. Як працює SSL-сертифікат [Електронний ресурс]. - Режим доступу: <https://ssl.com.ua/info/how-ssl-works/>.
6. Програмні системи веб-сайтів [Електронний ресурс]. - Режим доступу: <http://www.znannya.org/?view=WebDev>
7. Інформація про JavaScript [Електронний ресурс]. - Режим доступу: Режим доступу до ресурсу: <https://www.seo-kueche.de/lexikon/javascript/>
8. Інформація про Node.js [Електронний ресурс]. - Режим доступу: <https://www.mittwald.de/blog/hosting/was-ist-eigentlich-node-js>
9. Офіційний сайт Figma [Електронний ресурс]. - Режим доступу: <https://www.figma.com/>.
10. Zasoby_ludzkie. [Електронний ресурс]. - Режим доступу: https://mfiles.pl/pl/index.php/Zasoby_ludzkie.
11. Czym jest system CMS i jakie są jego rodzaje Treść pochodzi ze strony: www.empressia.pl. Nasze treści monitorujemy na bieżąco, dlatego nie kopiuj ich bez naszej zgody, ponieważ to prawnie zabronione.ms-i-jakie-sa-jego-rodzaje [Електронний ресурс]. - Режим доступу: <https://www.empressia.pl/blog/62-czym-jest-system-cms-i-jakie-sa-jego-rodzaje>.

12. Tworzenie stron internetowych [Електронний ресурс]. - Режим доступу: <https://nano.komputronik.pl/n/najlepsze-kreatory-stron-www/>.

13. Jaki język jest używany do tworzenia stron WWW? [Електронний ресурс]. - Режим доступу: <https://www.seofactor.co.uk/wiedza/jaki-jezyk-jest-uzywany-do-tworzenia-stron-www#:~:text=tworzenia%20stron%20WWW%3F-Do%20tworzenia%20stron%20internetowych%20u%C5%BCywa%20si%C4%99%20r%C3%B3w%C5%BCnych%20j%C4%99zyk%C3%B3w%20programowania%2C%20z,%2C%20Python%2C%20i%20wiele%20innych..>

14. Przydatne narzędzia dla stron internetowych [Електронний ресурс]. - Режим доступу: <https://webwavecms.com/blog/narzedzia-dla-strony-www>.

15. Figma – co to jest? Jak z niej korzystać? Wszystko o popularnym oprogramowaniu do projektowania dla UX i UI [Електронний ресурс]. - Режим доступу: <https://scroll.morele.net/poradniki/figma-co-to-jest/>.

16. HTML Tutorial [Електронний ресурс]. - Режим доступу: <https://www.w3schools.com/html/>.

17. CSS Tutorial [Електронний ресурс]. - Режим доступу: <https://www.w3schools.com/css/>.

18. The Modern JavaScript Tutorial [Електронний ресурс]. - Режим доступу: <https://javascript.info>.

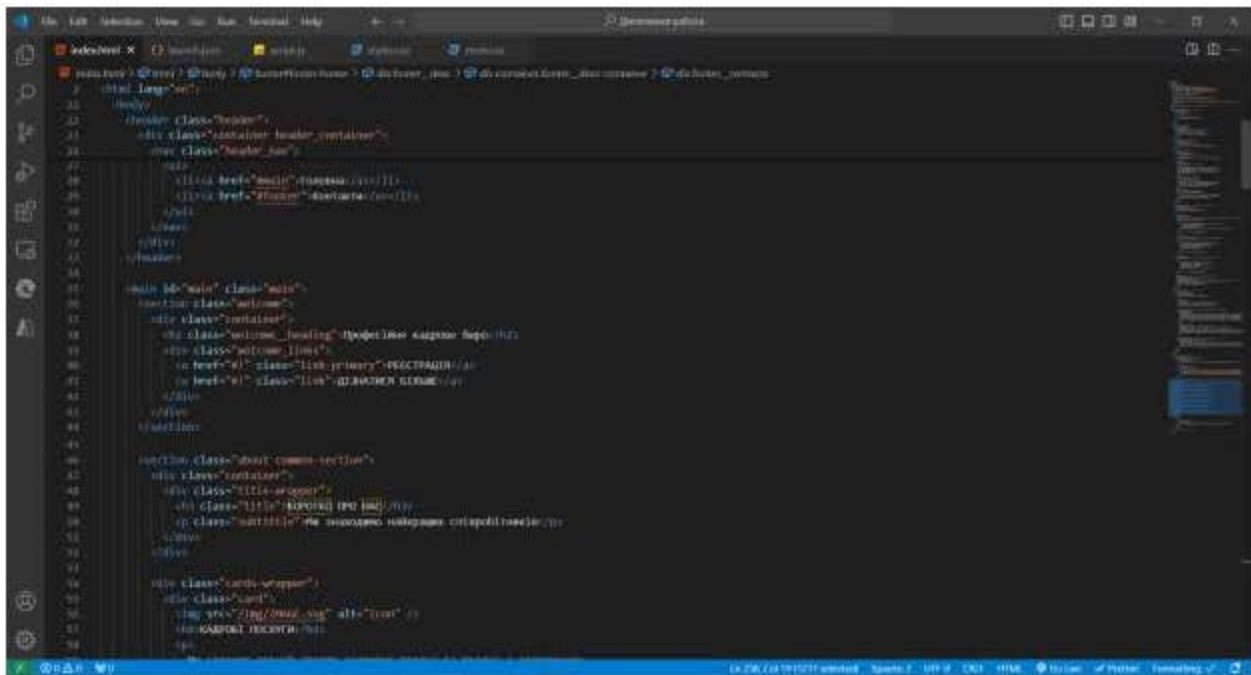
19. A Modern CSS Reset [Електронний ресурс]. - Режим доступу: <https://www.joshwcomeau.com/css/custom-css-reset/>.

20. Jak korzystać z Figmy? Krótka instrukcja – logowanie, nawigowanie, udostępnianie [Електронний ресурс]. - Режим доступу: https://scroll.morele.net/poradniki/figma-co-to-jest/#Jak_korzystac_z_Figmy_Krotka_instrukcja_%E2%80%93_logowanie_nawigowanie_udostepnianie.

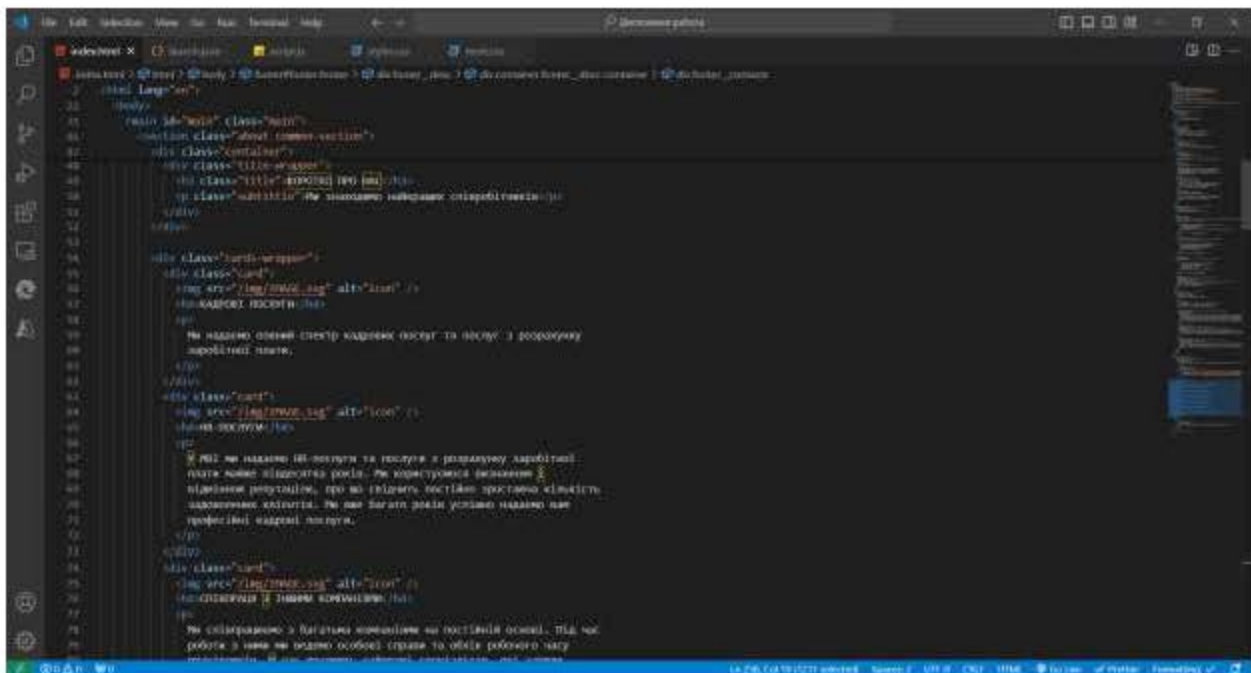
21. Statamic: A Guide to Creating a Recruitment Agency Website [Електронний ресурс]. - Режим доступу: Режим доступу до ресурсу: <https://statamic.dev/content-managers-guide>.

ДОДАТОК А

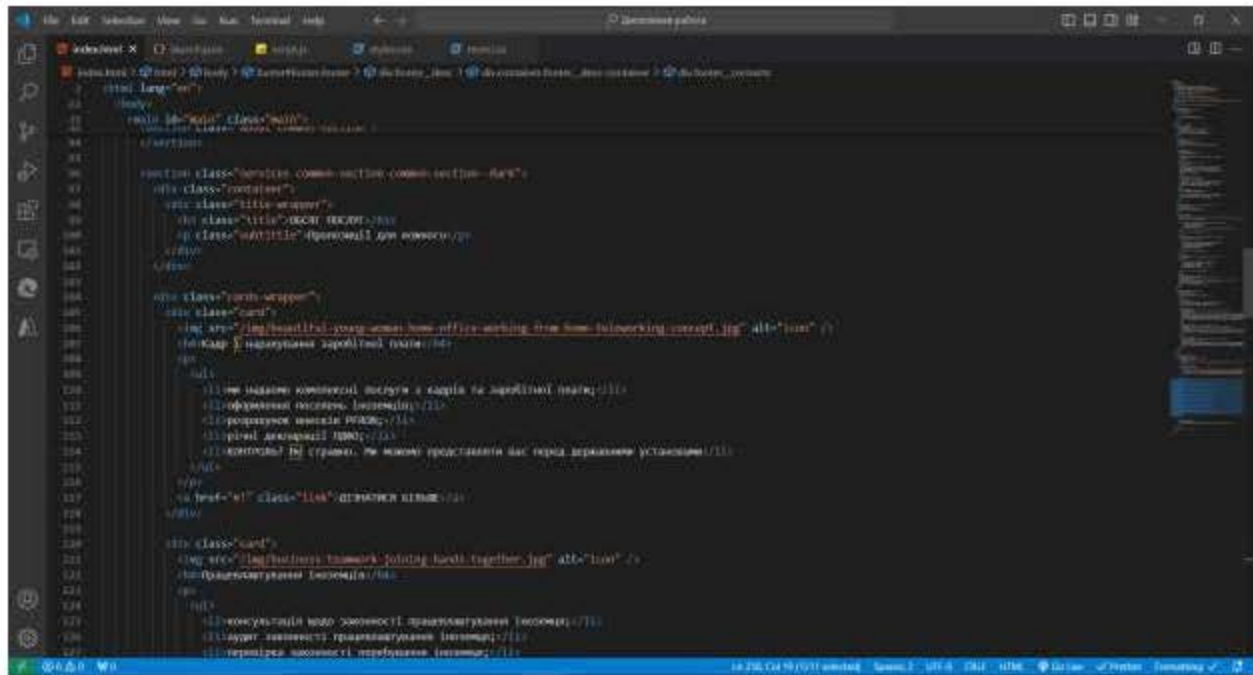
Розробка коду дизайну в Visual Studio Code. «index.html». Шапка сайту.



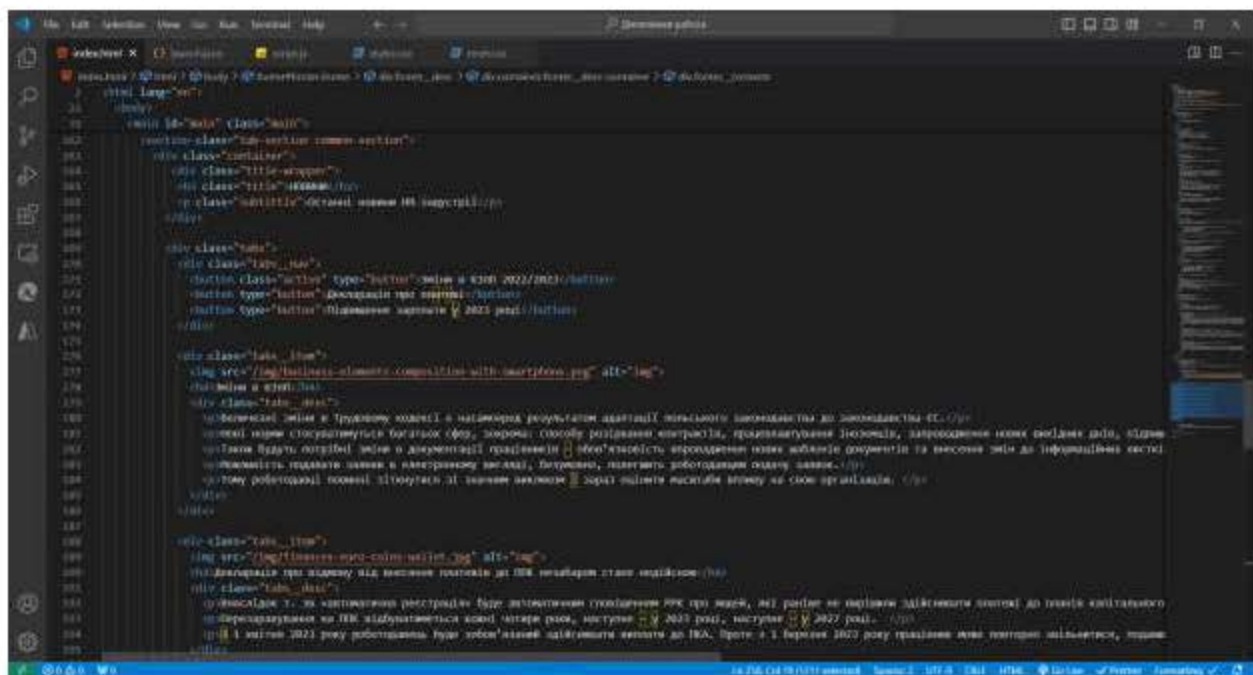
Сторінка «Коротко про нас». Інформація про підприємство.



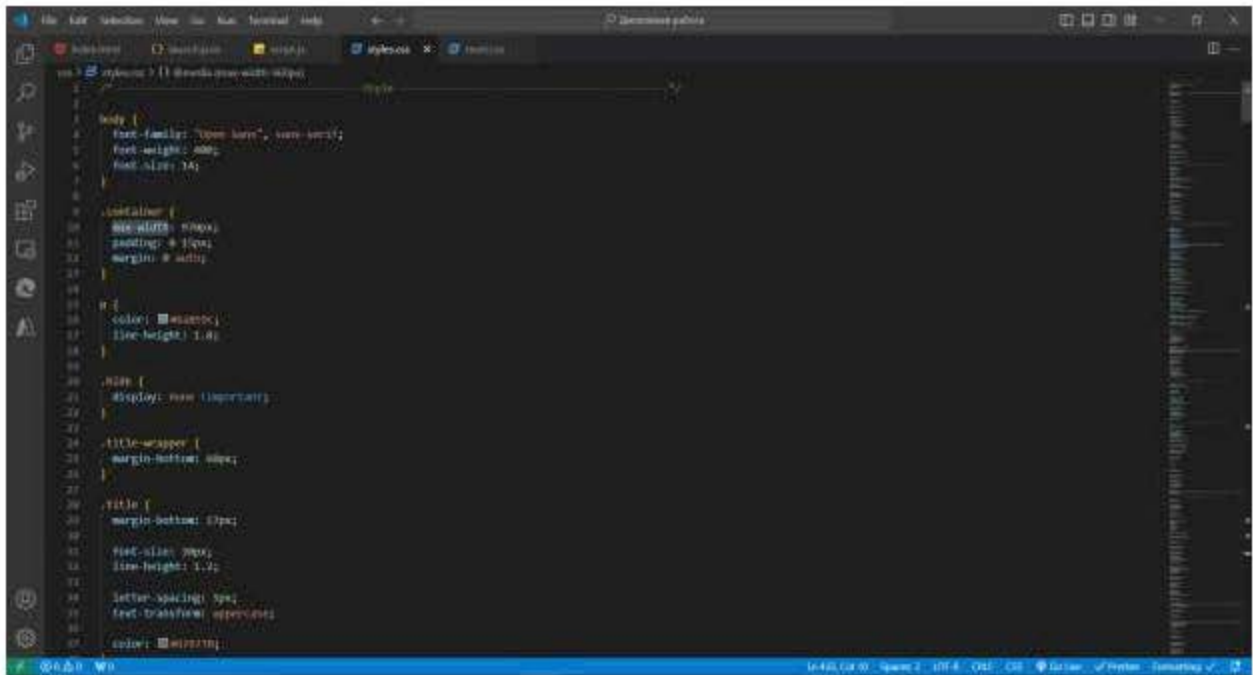
Сторінка «Обсяг послуг». Інформація про спектр послуг підприємства.



Сторінка «Новини». Інформація про останні новини HR-індустрії в Республіці Польща.



Сторінка «Контакти». Контактна інформація підприємства та соціальні мережі.



«reset.css» допомагає забезпечити послідовність і передбачуваність стилів, надаючи стабільну основу для подальшої роботи над дизайном веб-сторінки.

